IBM Cloud Application Performance Management

2019 年 6 月

ユーザーズ・ガイド

注記

本書および本書で紹介する製品をご使用になる前に、1509 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM® Cloud Application Performance Management の 2019 年 6 月のバージョン、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに 適用されます。

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

本書は下記原典を翻訳したものです。

原典：
IBM Cloud Application Performance Management
June 2019
User's Guide

発行：
日本アイ・ビー・エム株式会社

担当：
トランスレーション・サービス・センター

© Copyright International Business Machines Corporation 2014, 2019.

目次

第 1 章新機能 1

第 2 章 PDF 資料 43

第 3 章製品の概要 45
アーキテクチャーの概要 45
ユーザー・インターフェース 46
オファリングとアドオン 47
オファリングの詳細 50
エージェントおよびデータ・コレクター 52
変更履歴 52
機能 54
説明 58
機能 72
統合 80
資料 81
資料の表記上の規則 82

第 4 章デプロイメントの計画 85
システム要件 85
エージェントおよびデータ・コレクターによって 使用されるデフォルト・ポート 85
シナリオ 90
シナリオ: IBM API Connect のモニター 90
シナリオ: IBM Java アプリケーション・スタック のモニター 91
シナリオ: IBM 統合スタックのモニター 99
エージェントのダウンロードおよびデータ・コレクター 105
チュートリアル: エージェントのダウンロードとインストール 106
チュートリアル: データ・コレクターのダウンロードと構成 110

第 5 章エージェント とデータ・コレクターのデプロイメント 115

第 6 章エージェントのインストール 123
UNIX システムでのエージェントのインストール 124
AIX システムでのプリインストール 125
Solaris システムでのプリインストール 127
エージェントのインストール 128
Linux システムでのエージェントのインストール 130
Linux システムでのプリインストール 132
エージェントのインストール 137
Windows システムでのエージェントのインストール 139
Windows システムでのプリインストール 140
エージェントのインストール 142
非 root ユーザーとしてのエージェントのインストール 145
エージェント・インストール・ファイルの保護 146
エージェントのサイレント・インストール 147
前提条件スキャナーのバイパス 149
エージェントのアンインストール 150
WebSphere Applications エージェント: データ・コレクターの構成解除 152
Node.js エージェント: モニター・プラグインの削除 161
Microsoft .NET エージェント: .NET データ・コレクターの削除 162

第 7 章環境の構成..... 163
共通トピック.....163
ネットワーク接続.....163
管理対象システム名.....171
エージェントの管理対象システム名の変更.....174
エージェントの構成.....180
データ・コレクターの一般的な構成手順.....189
Amazon EC2 モニターの構成.....193
Windows システムでのエージェントの構成.....195
プロンプトへの応答によるエージェントの構成.....198
サイレント応答ファイルを使用したエージェントの構成.....199
Amazon EC2 エージェントの構成パラメーター.....200
AWS Elastic Load Balancer モニターの構成.....201
Windows システムでのエージェントの構成.....202
プロンプトへの応答によるエージェントの構成.....204
サイレント応答ファイルを使用したエージェントの構成.....205
Amazon ELB エージェントの構成パラメーター.....206
Azure Compute モニター構成.....206
Azure Compute の構成に関する情報.....208
Windows システムでのエージェントの構成.....208
プロンプトへの応答によるエージェントの構成.....211
サイレント応答ファイルを使用したエージェントの構成.....213
Azure Compute エージェントの構成パラメーター.....215
Cassandra モニターの構成.....217
Windows システムでのエージェントの構成.....218
Linux システムでのエージェントの構成.....218
サイレント応答ファイルを使用したエージェントの構成.....219
エージェントの構成パラメーター.....219
Cisco UCS モニターの構成.....220
Windows システムでのエージェントの構成.....221
サイレント応答ファイルを使用したエージェントの構成.....222
プロンプトへの応答によるエージェントの構成.....223
エージェントの構成パラメーター.....224
データ・プロバイダーの構成パラメーター.....225
Cisco UCS データ・ソースとの SSL 通信の使用可能化.....225
Java ヒープ・サイズの増大.....226
Citrix Virtual Desktop Infrastructure モニターの構成.....227
Citrix 読み取り専用管理者特権の有効化.....228
Windows システムでのエージェントの構成.....229
プロンプトへの応答によるエージェントの構成.....232
サイレント応答ファイルを使用したエージェントの構成.....233
Citrix VDI エージェントの構成パラメーター.....234
Windows イベントおよび PowerShell メトリックのモニターの有効化.....236
DataPower モニターの構成.....237
DataPower アプライアンスの構成.....237
DataPower エージェントの構成.....246
Db2 モニターの構成.....249
Windows システムでのエージェントの構成.....251
Linux または UNIX システム上でのエージェントの構成.....252
サイレント応答ファイルを使用したエージェントの構成.....253
Db2 メトリックを表示するための特権の付与.....255
ローカル環境変数の構成.....256
リモート・モニターの前提条件.....259
Hadoop モニターの構成.....260
Windows システムでのエージェントの構成.....261
Linux および AIX システム上でのエージェントの構成.....264

サイレント応答ファイルを使用したエージェントの構成 267
Hadoop イベントを表示するためのダッシュボードの構成 269
非管理者ユーザーへの許可の付与 269
HMC Base モニターの構成 269
SSH 接続のセットアップ 271
HMC 用の SDK の準備 272
仮想入出力をモニターするための HMC コンソール・サーバーの構成 274
CPU およびメモリー使用率のモニターの有効化 274
HTTP Server モニターの構成 275
IBM HTTP Server 応答時間モジュール 277
HTTP Server エージェントのコード・サンプル 277
IBM Cloud Monitoring の構成 279
Windows システムでのエージェントの構成 280
プロンプトへの応答によるエージェントの構成 280
サイレント応答ファイルを使用したエージェントの構成 281
IBM Cloud エージェントの構成パラメーター 282
IBM Integration Bus のモニターの構成 282
IBM Integration Bus エージェントの構成 283
データ使用可能化のための IBM Integration Bus の構成 287
エージェントに対するスナップショット・データ収集の無効化 294
IBM Integration Bus エージェントのトランザクション・トラッキングの構成 295
IBM Integration Bus エージェントに対する固有の管理対象システム名の指定 295
KQIUserExit ユーザー出口の削除 297
IBM MQ Appliances モニターの構成 297
プロンプトへの応答によるエージェントの構成 298
サイレント応答ファイルを使用したエージェントの構成 299
MQ Appliance エージェントの構成パラメーター 300
InfoSphere DataStage モニターの構成 302
Windows システムでのエージェントの構成 303
Linux システムでのエージェントの構成 303
環境変数の構成 303
サイレント応答ファイルを使用したエージェントの構成 304
エージェントの構成パラメーター 305
Internet Service Monitor の構成 306
ユーザー・インターフェースを使用した Internet Service Monitoring の構成 307
Windows システムでのエージェントの構成 451
Netcool/OMNIbus の有効化 454
J2SE のモニターの構成 455
トランザクション・トラッキングおよび診断データ収集のステータスの確認 460
トランザクション・トラッキングおよび診断データ収集のステータスの変更 461
JBoss モニターの構成 461
JMX MBean サーバー接続の有効化 464
JBoss サーバー管理ユーザーの追加 465
Web/HTTP 統計収集の有効化 466
Windows システムでのエージェントの構成 467
プロンプトへの応答によるエージェントの構成 469
サイレント応答ファイルを使用したエージェントの構成 470
JBoss エージェントの構成パラメーター 471
JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ 473
Linux KVM のモニターの構成 477
ユーザーの作成および必要な権限の付与 478
プロトコルの構成 478
RHEVM サーバーへの接続の構成 483
RHEVH サーバーへの接続の構成 484
RHEVM サーバーに接続するための構成パラメーター 485
RHEVH サーバーに接続するための構成パラメーター 486
MariaDB モニターの構成 488
Windows システムでのエージェントの構成 489

Linux システムでのエージェントの構成 490
サイレント応答ファイルを使用したエージェントの構成 490
Microsoft Active Directory モニターの構成 491
管理者ユーザーとしての Microsoft Active Directory エージェントの実行 492
ローカル環境変数の構成 492
Microsoft Active Directory エージェントの非管理者ユーザーとしての実行 494
属性グループ AD_Services_Status のドメイン・グループの構成 496
Microsoft Active Directory エージェントのアップグレード 497
Microsoft Cluster Server のモニターの構成 498
Windows Server 2008、2012、2016、および 2019 システムでの汎用サービス・クラスター・リソースの作成 498
サイレント応答ファイルを使用したエージェントの構成 499
ユーザー・アカウントの変更 499
Microsoft Exchange のモニターの構成 500
ユーザーの作成 500
Exchange Server ユーザーへの管理者権限の割り当て 503
Exchange Server ユーザーをローカル管理者にする 505
Exchange Server の到達可能性の構成 507
ドメイン・ユーザーの下で実行するためのエージェントの構成 508
ローカルでのエージェントの構成 508
サイレント応答ファイルを使用したエージェントの構成 512
エージェント用のローカル環境変数の構成 513
Microsoft Hyper-V のモニターの構成 513
Windows 上の Monitoring Agent for Microsoft Hyper-V Server を非管理者ユーザーによって実行するためのローカル・セキュリティー・ポリシーの指定 514
「ローカル・セキュリティー・ポリシー」許可の付与 515
DCOM 許可の変更 516
Hyper-V 管理者ユーザー・グループへの非管理者ユーザーの追加 516
Performance Business Monitor Users グループへの非管理者ユーザーの追加 517
Microsoft IIS モニターの構成 517
Windows システムでのエージェントの構成 518
サイレント応答ファイルを使用したエージェントの構成 518
ユーザー・アカウントの変更 519
Skype for Business Server (以前の Microsoft Lync Server) のモニターの構成 520
非管理者ユーザーの権限およびアクセス権 520
Windows システムでのエージェントの構成 521
サイレント応答ファイルを使用したエージェントの構成 522
ユーザー・アカウントの変更 523
エージェントの構成パラメーター 523
Microsoft .NET のモニターの構成 525
ローカル・アカウントまたはドメイン・アカウントを使用してエージェントを実行するための権限 526
データ・コレクターの登録 527
.Net エージェントの IIS 応答時間モジュールの使用 528
トランザクション・トラッキング・データおよび診断データの収集の有効化 530
configdc コマンドを使用した診断データの収集の有効化 530
エージェントの共存環境におけるトランザクション・トラッキングの有効化 531
構成更新のアクティブ化 532
データ・コレクターのパフォーマンス調整 533
Microsoft Office 365 のモニターの構成 536
構成されているユーザーの到達可能性の検証 536
Windows システムでのエージェントの構成 537
サイレント応答ファイルを使用したエージェントの構成 538
ユーザー・アカウントの変更 539
Skype QoS のモニター 539
ローカル環境変数の構成 540
Microsoft SharePoint Server モニターの構成 541
ユーザー・アカウントの変更 542

非管理者ユーザーによる Monitoring Agent for Microsoft SharePoint Server の実行 543
ローカル・セキュリティー・ポリシーの許可 544
Microsoft SQL Server のモニターの構成 545
ユーザーの作成および権限の付与 545
ローカル環境変数 550
エージェントの構成パラメーター 555
Windows システムでのエージェントの構成 556
Linux システムでのエージェントの構成 569
サイレント応答ファイルを使用したエージェントの構成 570
クラスター環境でのエージェントの実行 571
クラスター・ユーティリティーを 使用したエージェントの構成 574
ERRORLOG ファイルの複数の照合の構成 575
MongoDB のモニターの構成 577
デフォルト設定を使用したエージェントの構成 579
サイレント応答ファイルを使用したエージェントの構成 580
プロンプトへの応答によるエージェントの構成 581
MySQL のモニターの構成 582
Windows システムでのエージェントの構成 582
Linux システムでのエージェントの構成 583
サイレント応答ファイルを使用したエージェントの構成 584
NetApp ストレージのモニターの構成 585
NetApp Manageability SDK JAR ファイルのダウンロードおよびインストール 586
Windows システムでのエージェントの構成 586
サイレント応答ファイルを使用したエージェントの構成 587
プロンプトへの応答によるエージェントの構成 588
データ・プロバイダーの構成パラメーター 589
OnCommand Unified Manager の構成パラメーター 590
OnCommand API Service の構成パラメーター 591
Node.js モニターの構成 591
Node.js エージェントの構成 593
IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成 598
オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成 603
Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターの構成 608
OpenStack モニターの構成 612
OpenStack エージェントの構成 613
プロセス関連情報の収集および SSH 接続の有効化 615
構成値の追加 616
Oracle データベースのモニターの構成 618
Windows システムでのエージェントの構成 619
プロンプトへの応答によるエージェントの構成 623
サイレント応答ファイルを使用したエージェントの構成 627
Oracle Database エージェントのユーザーへの特権の付与 630
OS のモニターの構成 633
root 以外のユーザーとしての OS エージェントの実行 633
OS エージェント・ログ・ファイル・モニターの構成 635
OS エージェントのカスタム・スクリプトの構成 659
Linux OS エージェント・ファイル・システム・データ収集の構成 666
PHP モニターの構成 666
PostgreSQL のモニターの構成 668
Windows システムでのエージェントの構成 670
Linux システムでのエージェントの構成 670
サイレント応答ファイルを使用したエージェントの構成 671
Python モニターの構成 672
IBM Cloud アプリケーション用の Python データ・コレクターの構成 673
オンプレミス・アプリケーション用の Python データ・コレクターの構成 678
RabbitMQ モニターの構成 683
Windows システムでのエージェントの構成 683

Linux システムでのエージェントの構成 684
サイレント応答ファイルを使用したエージェントの構成 684
エージェントの構成パラメーター 685
Response Time Monitoring の構成 686
トランザクション・ダッシュボードの表示 687
Response Time Monitoring コンポーネント 687
インストールの計画 688
構成の計画 689
JavaScript インジェクション 690
Windows での Response Time Monitoring の再構成 691
AIX および Linux での Response Time Monitoring の再構成 692
「エージェント構成」ページを使用した構成 693
アプリケーションの追加 694
IBM HTTP Server 応答時間モジュールの構成 695
Packet Analyzer のロードマップ 705
IBM HTTP Server 応答時間モジュールから Packet Analyzer への再構成 713
エンド・ユーザー・トランザクションのロケーション値のカスタマイズ 714
追加の Web アプリケーションのトラッキング 715
Response Time Monitoring エージェントの固有の管理対象システム名の指定 718
Ruby モニターの構成 718
Ruby エージェントの構成 719
IBM Cloud アプリケーション用の Ruby データ・コレクターの構成 727
SAP モニターの構成 730
Windows システムでのエージェントの構成 731
Linux または AIX システム上でのエージェントの構成 732
サイレント応答ファイルを使用したエージェントの構成 733
エージェントの構成パラメーター 734
SAP ホスト名が管理対象システム名の長さ制限に従ってトリムされる 737
SAP システムでの ABAP トランスポートのインポート 737
SAP システムからの ABAP トランスポートの削除 743
エージェント構成の検証 744
データベース通信ポート番号の追加 748
SAP エージェントの拡張インストールおよび構成 748
SAP HANA データベースのモニターの構成 762
SAP NetWeaver Java スタックのモニターの構成 764
Windows システムでのエージェントの構成 765
Linux または AIX システム上でのエージェントの構成 766
サイレント応答ファイルを使用したエージェントの構成 767
データ・コレクターの構成 768
トランザクション・トラッキング・データおよび診断データの収集の有効化 769
データ・コレクター構成の削除 770
SAP NetWeaver アプリケーション・サーバー・インスタンスの復元 770
エージェントの構成パラメーター 771
Siebel モニターの構成 771
Siebel ユーザー・アカウントの確認 772
コンポーネントごとの統計モニターの有効化 773
Windows システムでのエージェントの構成 774
プロンプトへの応答によるエージェントの構成 778
サイレント応答ファイルを使用したエージェントの構成 779
Siebel エージェントの構成パラメーター 780
常にモニターされる Siebel コンポーネント・ログ 783
Sterling Connect Direct モニターの構成 783
Windows システムでのエージェントの構成 784
Linux システムでのエージェントの構成 784
サイレント応答ファイルを使用したエージェントの構成 785
エージェントの構成パラメーター 785
Sterling File Gateway モニターの構成 786
B2B REST API のインストール 786

Windows システムでの Sterling File Gateway エージェントの構成 787
Linux システムでの Sterling File Gateway エージェントの構成 787
サイレント応答ファイルを使用した Sterling File Gateway エージェントの構成 788
Linux でのデータ・プロバイダーのエージェント環境変数の構成 789
Windows でのデータ・プロバイダーのエージェント環境変数の構成 789
データ・プロバイダーの環境変数 790
B2B API 詳細の構成パラメーター 791
データベース詳細の構成パラメーター 792
Java API の構成パラメーター 792
Sybase サーバーのモニターの構成 792
権限の付与 793
コマンド行インターフェースを使用したエージェントの構成 795
サイレント応答ファイルを使用したエージェントの構成 796
照会のダーティー読み取り (dirty reads) の無効化 798
Synthetic Playback モニターの構成 799
Synthetic Playback エージェントに対するアップストリーム・プロキシー・サポートの有効化 800
Tomcat モニターの構成 801
デフォルト設定を使用した Tomcat エージェントの構成 802
Windows システムでのエージェントの構成 802
Linux システムでの Tomcat エージェントの構成 806
サイレント応答ファイルを使用した Tomcat エージェントの構成 806
トランザクション・トラッキング・データおよび診断データの収集の有効化 807
Tomcat アプリケーション・サーバーの更新または変更 808
VMware VI のモニターの構成 809
VMware VI エージェントのデプロイメントのサイズ設定と計画 810
VMware VI データ・ソースとの SSL 通信の使用可能化 810
Windows システムでのエージェントの構成 812
サイレント応答ファイルを使用したエージェントの構成 813
プロンプトへの応答によるエージェントの構成 814
データ・ソースの構成パラメーター 815
データ・プロバイダーの構成パラメーター 815
Java ヒープ・サイズの増大 816
WebLogic モニターの構成 817
Windows システムでのエージェントの構成 819
プロンプトへの応答によるエージェントの構成 823
サイレント応答ファイルを使用したエージェントの構成 824
WebLogic エージェントの構成パラメーター 825
WebLogic エージェントのトランザクション・トラッキングの構成 826
WebLogic エージェントのトランザクション・トラッキング・データを表示するための Application Performance ダッシュボードの構成 832
WebSphere Applications モニターの構成 833
WebSphere Applications エージェント 834
オンプレミス・アプリケーション用の Liberty データ・コレクターの構成 881
IBM Cloud アプリケーション用の Liberty データ・コレクターの構成 886
拡張データ・コレクター構成 892
WebSphere Extreme Scale をモニターするように WebSphere Applications エージェントを構成 922
WebSphere Infrastructure Manager モニターの構成 931
WebSphere MQ のモニターの構成 931
エージェントを実行するためのユーザー ID の許可 932
データ使用可能化のための IBM MQ (WebSphere MQ) の構成 934
WebSphere MQ エージェントの構成 936
複数のキュー・マネージャーに対する固有の管理対象システム名の指定 939
WebSphere MQ エージェントのトランザクション・トラッキングの構成 941
キューおよびチャネルの長期間ヒストリーのためのデータ収集の有効化 941
IBM MQ のキュー・マネージャーのキュー統計モニターの有効化 943
MQ アプライアンス上のキュー・マネージャーのリモートでのモニター 943
MQ アプライアンス上の HA キュー・マネージャーのリモートでのモニター 944

第 8 章他の製品およびコンポーネントとの統合..... 949
Cloud Event Management との統合..... 949
IBM Tivoli Monitoring V6.3 との統合..... 949
エージェントの共存..... 950
ハイブリッド・ゲートウェイ..... 953
OMEGAMON との統合..... 965
Netcool/OMNIbus との統合..... 966
Integration Agent for Netcool/OMNIbus のインストールおよび構成..... 966
Netcool/OMNIbus の統合の構成..... 968
Operations Analytics - Log Analysis との統合..... 970
Operations Analytics - Predictive Insights との統合..... 971
Alert Notification との統合..... 972
Control Desk との統合..... 973
IBM Cloud との統合..... 974
IBM Agent Builder との統合..... 974

第 9 章管理..... 977
Cloud APM コンソールの始動..... 977
しきい値とリソース・グループ..... 978
背景情報..... 978
リソース・グループ・マネージャー..... 982
チュートリアル: しきい値の定義..... 984
チュートリアル: 管理対象リソースでコマンドを実行するためのしきい値の定義..... 986
しきい値マネージャー..... 987
EIF 受信側に転送するためのイベントのカスタマイズ..... 992
イベントに対する応答としての E メールの送信..... 999
リソース・グループ管理サービス API の使用..... 999
しきい値管理サービス API の使用..... 1001
ユーザー・アクセスの管理..... 1003
役割と権限..... 1003
役割ベースのアクセス制御サービス API へのアクセスとその使用..... 1012
エージェントの管理..... 1013
非 root ユーザーとしてのエージェントの開始..... 1013
トランザクション・モニターのイベントしきい値..... 1015
OS エージェント・イベントの管理..... 1018
Website Monitoring でのシンセティック・トランザクションおよびシンセティック・イベントの管理..... 1028
ログ・ファイル・モニターについてエージェントとサーバーのパフォーマンスを最大化するためのガイドライン..... 1043
Availability Monitoring..... 1047
Availability Monitoring について..... 1047
Availability Monitoring へのアクセス..... 1048
テストの作成および構成..... 1049
モニター・ダッシュボードでのアプリケーションの可用性とパフォーマンスの表示..... 1060
Availability Monitoring の使用量..... 1073
API の解説..... 1073
拡張構成..... 1074
イベント E メール..... 1076

第 10 章ダッシュボードの使用..... 1079
すべてのマイ・アプリケーション - Application Performance ダッシュボード..... 1079
ログ・ファイルの検索..... 1081
アプリケーション - Application Performance ダッシュボード..... 1082
集約トランザクション・トポロジー・ウィジェットの操作..... 1086
グループおよびインスタンス - Application Performance ダッシュボード..... 1087
コンポーネント・ダッシュボードのグループ・ウィジェットの編集..... 1090

メトリックの調整および時間変化の比較....1091
カスタムのグラフおよび表の表示と管理....1092
アプリケーションの管理....1098
アプリケーションの追加....1099
アプリケーションの編集....1102
アプリケーションの削除....1103
オフライン・エージェントの表示および削除....1104
イベント状況....1109
Operations Analytics - Predictive Insights を使用した異常の調査....1111
カスタム・ビュー....1112
カスタム・ページの作成および管理....1113
カスタム・ページの表示....1120
ダッシュボード・ユーティリティー....1122
ダッシュボード URL のコピー....1122
トレースの設定....1123
Cloud APM コンソールのロック....1124
レポート....1124
Response Time Monitoring エージェント レポート....1125
Synthetic Playback エージェントのレポートの生成....1129
WebSphere Applications エージェント レポート....1134

第 11 章アップグレード....1139
エージェントのアップグレード....1139
エージェント構成変更の保持....1141
AIX 上のエージェント: アップグレード前のエージェントの停止と **slibclean** の実行....1142
AIX 上の HMC Base エージェント: アップグレード前の root 以外のユーザーとしてのエージェントの停止および **slibclean** の実行....1143
Node.js エージェント: アップグレード前にデータ・コレクター・プラグインを削除する....1143
Response Time Monitoring エージェント: IBM HTTP Server 応答時間モジュールのアップグレード....1144
Microsoft .NET エージェント: アップグレード前に .NET データ・コレクターを削除する....1146
OpenStack エージェント: OpenStack Identity API v3 を使用するためのエージェント・インスタンスの再構成....1146
Ruby エージェント: アップグレード前にデータ・コレクター・プラグインを削除する....1147
WebSphere Applications エージェント: データ・コレクターのマイグレーション....1147
Tomcat エージェント: Windows での TEMA コア・フレームワークのアップグレード....1152
データ・コレクターのアップグレード....1152

第 12 章トラブルシューティングとサポート....1155
エージェントのトラブルシューティング....1155
Internet Service Monitoring....1155
Microsoft Active Directory のモニター....1156
Microsoft IIS のモニター....1156
Microsoft .NET のモニター....1157
Microsoft SharePoint Server のモニター....1157
PostgreSQL のモニター....1158
IBM サポートに提供するモニター・エージェント・ログの収集....1158

第 13 章 Agent Builder....1161
Agent Builder の概要....1161
Agent Builder の一般的な手順....1162
データ・ソースおよびデータ・セット....1163
複数のサーバーまたはサーバーの複数のインスタンスのモニター....1164
エージェントのテスト、インストール、および構成....1165
オペレーティング・システム要件....1166
IBM Tivoli Monitoring に固有の機能....1166
Agent Builder のインストールおよび開始....1167

Agent Builder をインストールおよび実行するための前提条件 1167
Agent Builder のインストール 1167
Agent Builder の開始 1170
Agent Builder でのデフォルトのブラウザーの設定 1170
Agent Builder でのデフォルトの時刻認証局の設定 1171
Agent Builder のアンインストール 1171
サイレント・アンインストール 1172
エージェントの作成 1172
エージェントの命名および構成 1172
初期データ・ソースの定義 1174
Agent Editor を使用したエージェントの変更 1175
デフォルト・オペレーティング・システム 1176
自己記述型エージェント 1177
環境変数 1177
Watchdog 情報 1187
Cognos 情報 1188
「エージェントの生成ウィザード」リンク 1188
「データ・ソース定義」ページ 1189
「ランタイム構成情報」ページ 1190
「エージェント XML エディター」ページ 1190
編集内容および変更内容の保存 1190
エージェントのバージョンのコミット 1191
エージェントの新しいバージョン番号の設定 1192
製品コードの変更 1192
データ・ソースおよび属性のプロパティーの編集 1192
属性の作成、変更、および削除 1194
属性グループのフィルタリング 1202
式エディター 1202
数式演算子および関数 1206
オペレーティング・システムの指定 1213
データ収集の構成およびチューニング 1213
データ・ソースの定義およびテスト 1218
プロセスのモニター 1219
Windows サービスのモニター 1223
Windows Management Instrumentation (WMI) からのデータのモニター 1224
Windows パフォーマンス・モニター (Perfmon) のモニター 1226
Simple Network Management Protocol (SNMP) サーバーのデータのモニター 1229
Simple Network Management Protocol イベント送信側からのイベントのモニター 1233
Java Management Extensions (JMX) MBean のモニター 1239
Common Information Model (CIM) のデータのモニター 1258
ログ・ファイルのモニター 1260
AIX バイナリー・ログのモニター 1272
Windows イベント・ログのモニター 1273
コマンドの戻りコードのモニター 1276
スクリプトからのアウトプットのモニター 1280
Java Database Connectivity (JDBC) のデータのモニター 1285
ping を使用したシステム可用性のモニター 1293
HTTP 可用性および応答時間のモニター 1295
SOAP データ・ソースまたは他の HTTP データ・ソースのデータのモニター 1303
ソケットを使用したデータのモニター 1312
Java API を使用したデータのモニター 1322
既存のソースからのデータ・セットの作成 1335
2 つの属性グループの結合 1336
結合された属性グループの属性の操作 1339
結合された属性 1340
フィルタリング属性グループの作成 1341
ナビゲーター・グループの作成 1342
サブノードの使用 1344

サブノードの作成 1350
サブノード構成 1350
エージェント構成のカスタマイズ 1360
Agent Editor を使用した構成プロパティーの変更 1363
Windows リモート接続の構成 1363
Windows Management Instrumentation (WMI) 許可を持つユーザーの作成 1364
Secure Shell (SSH) リモート接続の構成 1366
ワークスペース、アクション実行コマンド、およびシチュエーションの作成 1367
シチュエーション、アクション実行コマンド、および照会の作成 1367
ワークスペースの 作成 1368
Cloud APM 用のエージェントの準備 1373
Agent Builder でのエージェントのテスト 1376
属性グループのテスト 1376
エージェントの完全テスト 1380
テスト環境変数 1384
テストおよび使用のためのモニター・インフラストラクチャーへのエージェントのインストール 1385
エージェントのインストール 1385
エージェントの生成およびインストール後の結果 1393
エージェントのアンインストール 1401
アプリケーション・サポート・ファイルのインポート 1403
Tivoli Enterprise Monitoring Agents のファイルのエクスポートおよびインポート 1403
Tivoli System Monitor Agents のファイルのエクスポートおよびインポート 1404
イベント・フィルタリングおよび要約 1405
重複イベントの制御 1406
Tivoli Enterprise Portal でのイベント・フィルタリングおよび要約の表示 1406
トラブルシューティングとサポート 1412
プロジェクト・ファイルの共有 1413
ソリューション・インストーラー・プロジェクトの共有 1413
コマンド行オプション 1413
コマンド - **generatelocal** 1415
コマンド - **generatemappingfile** 1415
コマンド - **generatezip** 1416
属性のリファレンス 1417
可用性ノード 1417
「パフォーマンス・オブジェクトの状況」ノード 1421
「スレッド・プールの状況」属性グループ 1427
「イベント・ログ」属性ノード 1431
ログ・ファイルの要約 1433
「AIX バイナリー・ログ」属性グループ 1435
モニターと通知の属性グループ 1438
SNMP イベントの属性グループ 1448
JMX イベントの属性グループ 1449
ping 属性グループ 1450
HTTP 属性グループ 1453
ディスカバリー属性グループ 1458
「アクション実行の状況」属性グループ 1460
「ログ・ファイル状況」属性グループ 1463
「ログ・ファイル正規表現統計」属性グループ 1466
既存エージェント用のアプリケーション・サポート拡張の作成 1471
アプリケーション・サポート拡張プロジェクトの作成 1471
プロジェクトへのサポート・ファイルの追加 1471
アプリケーション・サポート拡張インストール・イメージの生成 1472
アプリケーション・サポート拡張のインストール 1473
「ソリューション・インストール・プロジェクト」のアプリケーション・サポート拡張プロジェクトへの変換 1473
Cognos データ・モデルの生成 1473
Cognos データ・モデルを生成するための前提条件 1474

レポートの作成 1478
ICU 正規表現 1489
非エージェント・ファイル・バンドルの作成 1494
Remote Deploy Bundle Editor 1495
バンドルへのコマンドの追加 1496
バンドルへの前提条件の追加 1496
バンドルへのファイルの追加 1496
バンドルの生成 1497
Tivoli Netcool/OMNIbus プローブのデプロイ可能バンドルの作成 1498
動的ファイル名のサポート 1498
SNMP トラップ構成 1501
アクション実行コマンド参照 1504
SSHEXEC アクション 1505

アクセシビリティー機能 1507

特記事項 1509
商標 1510
製品資料に関するご使用条件 1511
IBM オンラインでのプライバシー・ステートメント 1511

第 1 章 新機能

最新リリースでは、新しいフィーチャー、機能、および対象が使用可能になっています。

- 各リリースまたはリフレッシュでのエージェント・バージョンに関する情報については、52 ページの『変更履歴』を参照してください。

2019 年 12 月

新規エージェント

MariaDB エージェント

Monitoring Agent for MariaDB は、MariaDB 環境またはアプリケーションを一元的に管理できるようにします。本製品は、問題を早期に検出して回避するために必要な情報を収集するための包括的な手段を提供します。情報は、システム全体で標準化されます。単一のコンソールから複数のサーバーをモニターできます。Monitoring Agent for MariaDB を使用することで、MariaDB 固有の情報を簡単に収集して分析できます。
インストール後のエージェントの構成については、488 ページの『MariaDB モニターの構成』を参照してください。

エージェントのプラットフォーム・サポートの拡張

以下のエージェントおよびプラットフォームがサポートされるようになりました。

Solaris X86-64

- Oracle Database エージェント
- WebLogic エージェント

エージェントの機能拡張

Cassandra エージェント

新しい 2 つの属性「エージェント・ホスト名」および「エージェント・インスタンス名」が、「クラスターの詳細」、「ノード統計」、および「キースペース 詳細」属性グループに追加されました。

Db2® エージェント

現在実行中の SQL をモニターするためのサポートが追加されました。

IBM Integration Bus エージェント

新しい 2 つのグループ・ウィジェット「TCPIP クライアント接続 (TCPIP Client Connections)」および「TCPIP サーバー接続 (TCPIP Server Connections)」が「統合サーバー状況 - 詳細 (Integration Server Status - Detail)」ページに追加されました。

Internet Service Monitoring

- 以下の新しい 2 つの構成パネル変数を追加しました。
  - *Active*: プロファイル・エレメントの状態をアクティブまたは非アクティブとして選択します。
  - *sniServerName*: SNI 対応 Web サーバーからの証明書が必要なホストやサーバーの名前を示します。
- HTTP、HTTPS、および DNS モニターの場合に、「**データ検証**」タブの下の「**デフォルト設定 (Default settings)**」を編集できるようになりました。
- HTTP および HTTPS モニターの場合に、エージェントが「**ページ**」フィールドで & 文字をサポートするようになりました。
- エージェントが、HTTP および HTTPS モニターの「**正規表現 (regex)**」フィールドでデンマーク語文字をサポートするようになりました。

  注 : この機能を使用するには、エージェントをインストールする前に、Linux プラットフォームでロケールを da_DK に設定してください。

© Copyright IBM Corp. 2014, 2019

Microsoft Active Directory エージェント

- 新しいウィジェット「KCC の詳細 (KCC details)」を「状況の概要」ページに追加しました。
- 以下の新しい属性グループを「属性の詳細」タブに追加しました。
  - ディレクトリー・サービス
  - Kerberos 整合性チェッカー
  - Kerberos 鍵配布センター
  - ネーム・サービス・プロバイダー
  - ディレクトリー交換サービス

Microsoft .NET エージェント
新しい属性「要求名」を「データベース呼び出しの詳細」属性グループに追加しました。この属性は、データベース照会を起動する要求の名前を表示します。

Microsoft Exchange Server エージェント

- 新しいウィジェット「トランスポート SMTP 受信 (Transport SMTP Recive)」を「状況の概要」ページに追加しました。
- 以下の新しい属性グループを「属性の詳細」タブに追加しました。
  - MS Exchange AB
  - MS Exchange ADAccess プロセス
  - MS Exchange ADAccess キャッシュ
  - MS Exchange ADAccess ドメイン・コントローラー
  - MS Exchange ADAccess フォレスト探索

Microsoft Hyper-V Server エージェント
Windows Server 2019 に対するサポートが追加されました。

Microsoft IIS エージェント

- 以下の新しいグループ・ウィジェットを追加しました。
  - システム - メイン・メモリーの統計情報
  - IIS サーバー - 割り当てられたメモリー使用量
  - IIS サーバー - 割り当てられた CPU 使用量
  - ワーカー・プロセスの詳細
  - .Net メモリー管理
- 「ワーカー・プロセスの詳細」グループ・ウィジェットの各アプリケーション・プール名に対して、1 秒あたりの要求処理数、経過時間、キュー内の要求数、メモリーおよび CPU の使用量のヒストリカル・トレンドを示すページが作成されました。
- 「.Net メモリー管理」グループ・ウィジェットの各アプリケーション・プール名に対して、GC の時間の割合 (%) のヒストリカル・トレンドを示すポップアップが追加されました。

Microsoft SharePoint Server エージェント

- 重大度の高いログの情報を提供する Trace_Log という新しい属性グループが追加されました。
- 新しい 2 つのグループ・ウィジェット「トレース・ログの詳細」および「直近 1 時間のトレース・ ログ・カウント」を概要ページに追加しました。これらのウィジェットは、最新の 100 件のトレース・ログ・イベントの詳細、および直近 1 時間における予期しないレベルと高レベルのトレース・ログのカウントの詳細を表示します。

MySQL エージェント
エージェントは、一貫してサーバーの再始動後にデータを収集します。

NetApp Storage エージェント
エージェントが、ボリュームにマップされた Qtree の正確なリストを表示するようになりました。

PostgreSQL エージェント

エージェントが PostgreSQL サーバー バージョン 12 をサポートするようになりました。

Response Time Monitoring エージェント

- 新しい構成パラメーター KT5AARIPTOUSERID が追加されました。これにより、AAR 生データのユーザー名プロパティーにクライアント IP アドレスを保存できます。デフォルトでは NO に設定されています。この設定を変更するには、Response Time Monitoring エージェントを再始動する必要があります。
  - KT5AARIPTOUSERID=NO: 値が NO の場合、Response Time Monitoring エージェントは、トランザクションのユーザー名を AAR の userID プロパティーに保存します。
  - KT5AARIPTOUSERID=YES: 値が YES の場合、Response Time Monitoring エージェントは、トランザクションの送信元 IP アドレスを AAR の userID プロパティーに保存します。
- Response Time Monitoring エージェントが、サイレント構成における KT5AARIPTOUSERID 値の指定をサポートするようになりました。
- 既存のグループ・ウィジェット「下位のパフォーマンス（ユーザー別） - 上位 5 件」のタイトルを「下位のパフォーマンス（ユーザー別） - 上位 20 件（Worst By User - Top 20)」に変更されました。このグループ・ウィジェットは、選択された期間におけるトランザクション失敗率が高いユーザーの上位 20 件を表示するように変更されました。

VMware

- エージェントが、構成パネルからの場合のように「構成されているアドレス（Configured Address)」を表示するのではなく、vSphere API 呼び出しから vCenter の IP アドレスまたはホスト名を取り出す操作をサポートするようになりました。この機能をアクティブにするには、エージェント環境で所定のフラグを Y に設定します。例えば KVM_RETRIEVE_HOSTNAME_FROM_API=Y とします。
- 再試行回数に制限を指定し、データ・ソースとの接続の再試行を制限できるようにしました。例えば、KVM_DATA_PROVIDER_CONNECTION_RETRY_COUNT=1000 という変数をエージェントの環境ファイルに追加すると、vCenter との接続が失敗した場合に、接続の再試行回数にロックが適用されます。「1000」は、その後もエージェントが再試行を続け、接続の再試行が 1000 回失敗したら、データ・プロバイダー・プロセスを停止してログ・メッセージ「NO MORE ATTEMPTS OF CONNECTION;STOPPING THE DATA COLLECTION, TO RESUME MONITORING PLEASE RESTART THE AGENT. TO HAVE MORE ATTEMPT OF CONNECTIONS, RESET THE VALUE OF THE VARIABLE KVM_DATA_PROVIDER_CONNECTION_RETRY_COUNT」を出力することを示します。接続の再試行回数のデフォルト値は 6 であり、必要に応じてユーザーが任意のしきい値を設定できます。
- エージェントがインスタンス固有のヒープ・サイズの構成をサポートし、システムに割り振られたメモリーを効率的に利用するようになりました。例えば、KVM_CUSTOM_JVM_ARGS= -Xmx512m という変数をインスタンスの環境ファイルで設定すると、インスタンスが 512 MB のヒープ・メモリーを使用するように構成されます。このサイズは、インスタンスがモニターする vCenter オブジェクトの総数に応じて変更できます。

2019 年 9 月

エージェントのプラットフォーム・サポートの拡張

以下のエージェントおよびプラットフォームがサポートされるようになりました。

Solaris X86-64

- Db2 エージェント
- SAP エージェント
- Sybase エージェント
- UNIX OS エージェント
- WebSphere® Applications エージェント
- WebSphere MQ エージェント

- IBM Integration Bus エージェント

x86-64 版 RHEL (64 ビット)

- Internet Service Monitoring エージェント
- Microsoft SQL Server エージェント
- Sybase エージェント

POWER リトル・エンディアン版 RHEL (ppc64le)

- RabbitMQ エージェント

エージェントの機能拡張

Db2 エージェント

- エージェントは、Db2 サーバーのバージョン 11.5 をサポートするようになりました。

Hadoop エージェント

- 新規構成パラメーター**「固有のクラスター名 (Unique Cluster Name)」**が構成パネルに追加されました。このパラメーターは、バージョンとフレーバーを示す、Hadoop クラスターの固有の名前です。
- Hadoop エージェントに、Ambari サービスで作成されたしきい値の表示項目が表示されるようになりました。
- Hadoop エージェントは、Hadoop クラスター内の Streaming Analytics Manager サービスのモニターをサポートするようになりました。
- Hadoop エージェントは、Hadoop クラスター内の Schema Registry サービスのモニターをサポートするようになりました。

HTTP Server エージェント

- エージェントは、Solaris Sparc 上の Oracle HTTP Server をサポートするようになりました。

Internet Service Monitoring

- エージェントは、IBM Tivoli® Netcool/OMNIbus をサポートするようになりました。
- エージェントが拡張され、既存のプロファイルからのプロファイルの削除、および既存のプロファイルの名前変更ができるようになりました。

MongoDB エージェント

- エージェントは、MongoDB データベース・バージョン 4.x をサポートするようになりました。

MySQL エージェント

- 「ヘルプ」セクションの「アプリケーションの可用性」に FQDN 属性が追加されました。
- 「IP アドレス」のエージェント構成パラメーターに対するツールチップ表示を修正しました。
- IBM Cloud App Management でのモニター用に、以下の新規属性が追加されました。
  - データベース・サイズの情報
  - エラー情報
  - Db インスタンス・ロックの数
  - ユーザーの接続の詳細
  - プロセス・リストの詳細
  - イベント情報

PostgreSQL エージェント

- データベースでのデッドロックの数をモニターするために、`Deadlocks_Count_Crit` および `Deadlocks_Count_Warn` という 2 つの新規シチュエーションが追加されました。これにより、デッドロックの正確さの問題に対処できます。

- デッドロックの詳細を確認するために、Deadlocks_Info という新規属性グループが 1 つ追加されました。

Sybase エージェント

- 「ヘルプ」セクションの「アプリケーションの可用性」に FQDN 属性が追加されました。

Synthetic Playback エージェント

- Firefox V68.0 ESR がサポートされるようになりました。
- システム・プロキシー構成、PAC プロキシー構成、およびプロキシーなしの構成がサポートされるようになりました。

Tomcat エージェント

- エージェントが拡張され、JVM のヒープ/非ヒープ・メモリー・プール使用量をモニターするためのメトリックおよび UI ビューが使用できるようになりました。
- エージェントが拡張され、スレッドをモニターするためのメトリックと UI ビュー、および JVM のクラス・ロード情報が使用できるようになりました。
- エージェント UI の「サーバー情報」ビューに FQDN が表示されるようになりました。

UNIX OS エージェント

- エージェントが更新され、カスタム・スクリプト機能が追加されました。シェル・スクリプト、PERL スクリプト、およびその他のタイプのスクリプトを使用することができます。

VMware VI エージェント

- 新規構成フィールド **KEY_STORE_PASSWORD** が追加されました。このフィールドを使用すると、エージェント JRE 用に設定された新しい鍵ストア・パスワードを使用してエージェントを構成できます。

2019 年 6 月

エージェントのプラットフォーム・サポートの拡張

以下のエージェントおよびプラットフォームがサポートされるようになりました。

Red Hat Enterprise Linux (RHEL) 8

以下のエージェントおよびデータ・コレクターで RHEL 8 がサポートされるようになりました。エージェントを RHEL 8 にインストールする前に、132 ページの『Linux システムでのプリインストール』 のセクション 133 ページの『特定のオペレーティング・システム』を確認してください。

RHEL 8 on x86-64 (64 ビット)

- Cassandra エージェント
- Cisco UCS エージェント
- DataPower® エージェント
- DataStage® エージェント
- Db2 エージェント
- Hadoop エージェント
- HTTP Server エージェント
- Integration Agent for Netcool®/OMNIbus
- Internet Service Monitoring エージェント
- J2SE データ・コレクター
- Linux KVM エージェント
- Linux OS エージェント
- MongoDB エージェント
- MQ Appliance エージェント
- MySQL エージェント

- NetApp Storage エージェント
- Node.js データ・コレクター
- PHP エージェント
- Python データ・コレクター
- PostgreSQL エージェント
- RabbitMQ エージェント
- Response Time Monitoring エージェント
- Ruby エージェント
- SAP エージェント
- SAP HANA データベース・エージェント
- SAP NetWeaver Java™ Stack エージェント
- Sterling Connect Direct エージェント
- Sterling File Gateway エージェント
- Sybase エージェント
- Tomcat エージェント
- VMware VI エージェント
- WebSphere Applications エージェント
- WebSphere MQ エージェント

RHEL 8 on System z

- Db2 エージェント
- Hadoop エージェント
- Linux OS エージェント
- MySQL エージェント
- Node.js データ・コレクター
- Python データ・コレクター
- Response Time Monitoring エージェント
- WebSphere Applications エージェント
- WebSphere MQ エージェント

RHEL 8 on POWER Little Endian (ppc64le)

- Db2 エージェント
- Hadoop エージェント
- J2SE データ・コレクター
- Linux OS エージェント
- MySQL エージェント
- Node.js データ・コレクター
- SAP NetWeaver Java Stack エージェント
- WebSphere Applications エージェント
- WebSphere MQ エージェント

Solaris Sparc 10 および 11

- JBoss エージェント
- Oracle Database エージェント
- WebLogic エージェント

Windows Server 2019

- WebSphere Applications エージェント

エージェントの機能拡張

Hadoop エージェント

- Hadoop エージェントは、HDF 3.3 (HDP 3.1.0 を含む) Ambari サービスの Big SQL 6.0 のモニターをサポートするようになりました。
- Hadoop エージェントは、x86-64 プラットフォーム版の SUSE Linux Enterprise Server (SLES) 15 をサポートするようになりました。

HMC Base エージェント
HMC Base エージェントは HMC V9.1 をサポートしています。

Integration Agent for Netcool/OMNIbus
エージェントが更新され、Red Hat Enterprise Linux (RHEL) 8 および SUSE Linux Enterprise Server (SLES) 15 をサポートするようになりました。

Internet Service Monitoring エージェント
エージェントに サービス保障エージェント モニターが搭載され、Cisco サービス保証エージェントのプローブをモニターできるようになりました。

Microsoft IIS エージェント
エージェントが拡張され、Windows Server 2019 に対する許容度のプロビジョンが備わりました。この機能拡張により、Windows Server 2019 にインストールされているエージェントの FTP サイト・データが表示されます。

MongoDB エージェント
エージェントは、x86-64 (64 ビット) プラットフォーム版の Red Hat Enterprise Linux (RHEL) 8 をサポートするようになりました。

Python データ・コレクター ifix02
データ・コレクターは Django 1.10 以上をサポートするようになりました。

SAP エージェント

SAP エージェントは、以下のプラットフォームをサポートするようになりました。

- x86-64 (64 ビット) 版の Red Hat Enterprise Linux (RHEL) 8
- SAP NetWeaver Application Server 7.52 (SAP Basis 752)

SAP HANA データベース・エージェント

SAP HANA データベース・エージェントは、以下の機能によって拡張されています。

- SAP HANA データベース・エージェントの :HDB Subnode ノードにホスト名が追加され、一意的に識別できるようになりました。
- エージェントがスケールアウト・アーキテクチャーをサポートするようになりました。
- x86-64 (64 ビット) プラットフォーム版および Linux ppc64le プラットフォーム版の Red Hat Enterprise Linux (RHEL) 8。
- x86-64 (64 ビット) プラットフォーム版の SUSE Linux Enterprise Server (SLES) 15。
- 新規属性「トリム対象ホスト (Trimmed Host)」が「システム・データベース」属性グループの下に追加されました。

SAP NetWeaver Java Stack エージェント

SAP NetWeaver Java Stack エージェントは、以下のプラットフォームをサポートします。

- x86-64 (64 ビット) 版の Red Hat Enterprise Linux (RHEL) 8
- x86-64 (64 ビット) 版の SUSE Linux Enterprise Server (SLES) 15。
- Windows Server 2019 DE および SE
- Windows Server 2016 DE および SE

Synthetic Playback エージェント

- Selenium IDE 3.2.X、3.3.X、または 3.5.X によって記録された .side スクリプトをサポートします。
- Firefox ESR 60.5.1 による再生をサポートします。
- **待機**、**フロー制御**、および **linkText ロケーター・タイプ**の Selenium IDE コマンドをサポートします。

Skype for Business Server エージェント

Skype for Business Server エージェントは、以下の機能によって拡張されています。

- エージェントが Skype for Business Server 2019 をサポートするようになりました。
- 2 つの新しいグループ・ウィジェット (`Database-Throttled Requests(DBStore)` および `Database-Throttled Requests(SHAREDDBStore)`) が「概要」ページに追加されました。「概要」ページには、DBstore および共有 DBstore のデータベース・キューの待ち時間が長いために Skype for Business Server によって制限された要求の数が表示されます。

前提条件スキャナー

SKIP_PRECHECK コマンドの代わりに **IGNORE_PRECHECK_WARNING** コマンドを使用できるようになりました。詳しくは、149 ページの『前提条件スキャナーのバイパス』を参照してください。

資料の強化

各エージェントおよびデータ・コレクターのバージョン情報と変更履歴が素早く見つかるように、ページが作成されています。52 ページの『変更履歴』を参照してください。

2019 年 3 月

エージェントのプラットフォーム・サポートの拡張

以下のエージェントおよびプラットフォームがサポートされるようになりました。

Windows Server 2019

- Cassandra エージェント
- DataStage エージェント
- Db2 エージェント
- Hadoop エージェント
- Internet Service Monitoring
- Microsoft Active Directory エージェント
- Microsoft Cluster Server エージェント
- Microsoft IIS エージェント
- Microsoft Exchange Server エージェント
- Microsoft SQL Server エージェント
- MySQL エージェント
- PostgreSQL エージェント
- RabbitMQ エージェント
- SAP エージェント
- SAP HANA データベース・エージェント
- Sybase エージェント
- Tomcat エージェント
- Windows OS エージェント

Solaris SPARC 10 および 11

- Db2 エージェント
- HTTP Server
- MySQL エージェント
- SAP エージェント
- Sybase エージェント
- UNIX OS エージェント
- WebSphere Applications エージェント

Monitoring Agent for Cassandra

Cassandra エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 オペレーティング・システムに対するサポートが追加されました。
- トラブルシューティング用の詳細ロギングが追加されました。

Monitoring Agent for Db2

Db2 エージェントは、以下の機能によって拡張されています。

- Db2 エージェントが Windows Server 2019 をサポートするようになりました。
- Db2 エージェントが Solaris SPARC 10/11 プラットフォームをサポートするようになりました。

Monitoring Agent for Hadoop

Hadoop エージェントは、以下の機能によって拡張されています。

- SSL 対応の Hadoop BigInsights® クラスター、Hortonworks クラスター、および Cloudera クラスターをモニターするためのサポートが追加されました。
- SSL 対応の Hadoop クラスターへの接続をテストするためのサポートが追加されました。
- Windows Server 2019 オペレーティング・システム (Datacenter エディションおよび Standard エディション) のサポートが追加されました。
- Hadoop オファリングの Cloudera 6.1.1 (CDH 6.1.1) をモニターするためのサポートが追加されました。
- Hadoop オファリングの Hortonworks 3.1.0 (HDP 3.1.0) をモニターするためのサポートが追加されました。

Monitoring Agent for IBM Integration Bus

IBM Integration Bus エージェントは、以下の機能によって拡張されています。

- IBM App Connect Enterprise V11 をモニターするための許容性サポートが追加されました。詳しくは、283 ページの『IBM Integration Bus エージェントの構成』を参照してください。

Monitoring Agent for Microsoft Internet Information Services

Microsoft IIS エージェントは、Windows Server 2019 オペレーティング・システムをサポートするようになりました。

Monitoring Agent for InfoSphere® DataStage

DataStage エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 オペレーティング・システムに対するサポートが追加されました。
- エージェントのパフォーマンスを向上させるために、データ収集照会に照会タイムアウトが追加されました。

Monitoring Agent for Microsoft Active Directory

Microsoft Active Directory エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 に対するサポートが追加されました。

- Active Directory サーバーに関連するサービス状態を提供する新しい属性グループ AD_Services_Status が追加されました。
  このサービス状態に基づいて、Active Directory のサーバー状況が判別されます。
- Active Directory のサーバー状況をモニターする新規シチュエーション AD_Server_Status が追加されました。
- アクティブなバージョンとモニター対象の OS の名前を提供する新しい属性グループ Root_Directory_server が追加されました。

Monitoring Agent for Microsoft Cluster Server

Microsoft Cluster Server エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 オペレーティング・システムに対するサポートが追加されました。
- CLUSTER_SERVICE_VERSION 属性が追加されました。

Monitoring Agent for Microsoft Exchange Server

Microsoft Exchange Server エージェントは、以下の機能によって拡張されています。

- MS Exchange Server 2019 に対するサポートが追加されました。
- MAPI over HTTP プロトコルの統計に関する情報を提供する新しい属性グループ MSExchange MAPIoverHTTP が追加されました。

Monitoring Agent for Internet Services

Internet Service Monitoring エージェントは、以下の機能によって拡張されています。

- LDAP、NTP、NNTP、SOAP、SNMP、SIP、RTSP、RPING、RADIUS、および TFTP のモニターに対するサポートが追加されました。
- Windows 2008 R2 Server OS および Windows Server 2019 に対するサポートが追加されました。

Monitoring Agent for Microsoft SQL Server

Microsoft SQL Server エージェントは、Windows Server 2019 をサポートするようになりました。

Monitoring Agent for MySQL

Monitoring Agent for MySQL は、以下の機能によって拡張されています。

- Windows Server 2019 に対するサポートが追加されました。
- Solaris SPARC 10/11 プラットフォームに対するサポートが追加されました。
- エージェントが開始した MySQL サーバーとの JDBC 接続に追加のプロパティーを設定する機能が追加されました。

Monitoring Agent for PostgreSQL

PostgreSQL エージェントは、Windows Server 2019 オペレーティング・システムをサポートするようになりました。

Monitoring Agent for RabbitMQ

RabbitMQ エージェントは、Windows Server 2019 オペレーティング・システムをサポートするようになりました。

Monitoring Agent for Skype for Business Server

Skype for Business Server エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 オペレーティング・システムに対するサポートが追加されました。
- KQL_Server という新しい属性グループが追加され、Skype for Business Server 製品の関連情報が表示されるようになりました。
- サーバーのフロントエンドおよび IM 会議サービス状況に基づいて Skype for Business Server 状況をモニターするために、Skype_Server_Down という新規シチュエーションが追加されました。

Monitoring Agent for SAP Applications

SAP エージェントは、以下の機能によって拡張されています。

- SAP エージェントと SAP サーバーの間で使用されているアプリケーション・ユーザーのパスワードにおいて大/小文字を区別する機能が追加されました。
- Windows Server 2019 オペレーティング・システム (Datacenter エディションおよび Standard エディション) に対するサポートが追加されました。
- SAP システムに 24 時間を超えて存在している長時間実行ジョブを表示するサポートが追加されました。
- /IBMMON/ITM_MAIALRT_INX 汎用モジュールのパフォーマンスを改善しました。
- Solaris v10 および v11 SPARC オペレーティング・システムに対するサポートが追加されました。
- 「管理対象システム名」の最大 32 文字という制限に一致させるための、SAP ホスト名に対するトリム機能が追加されました。

Monitoring Agent for SAP HANA Database

SAP HANA データベース・エージェントは、以下の機能によって拡張されています。

- テナント DB 名と HANA システム SID が同じ場合にテナント・データベースを検出するためのサポートが追加されました。
- Windows Server 2019 オペレーティング・システム (Datacenter エディションおよび Standard エディション) に対するサポートが追加されました。

Monitoring Agent for Sybase Server

Sybase エージェントは、以下の機能によって拡張されています。

- Windows Server 2019 に対するサポートが追加されました。
- Solaris SPARC 10/11 プラットフォームに対するサポートが追加されました。
- Sybase 照会を拡張して並行性を向上させ、ロックを削減しました。

Monitoring Agent for Tomcat

Tomcat エージェントは、Windows Server 2019 オペレーティング・システム (Datacenter エディションおよび Standard エディション) をサポートするようになりました。

Monitoring Agent for UNIX OS

UNIX OS エージェントは、以下の機能によって拡張されています。

- Solaris SPARC 10 および 11 に対するサポートが追加されました。

Monitoring Agent for VMware VI

VMware VI エージェント が更新され、すべての複数行グラフに関するグラフの平均のトレンドを表示するときに、使用不可の値 (-1) が無視されるようになりました。

Monitoring Agent for WebSphere Applications

WebSphere Applications エージェントは、以下の機能によって拡張されています。

- Solaris SPARC 10 および 11 に対するサポートが追加されました。
- WebSphere® Extreme Scale をモニターするためのサポートが追加されました。Extreme Scale の 1 つ以上のゾーンのモニターは、そのゾーンに属している任意のサーバーのノードで構成することができます。1 つ以上のゾーン内のさまざまなサーバー、マップ・セット、およびパーティションの情報は、ドリルダウンすることにより表示可能です。詳しくは、922 ページの『WebSphere Extreme Scale をモニターするように WebSphere Applications エージェントを構成』を参照してください。

Monitoring Agent for WebSphere MQ

WebSphere MQ エージェントは、以下の機能によって拡張されています。

- SLES 15 xLinux に対するサポートが追加されました。

- キュー・マネージャーの統計を収集し、収集したデータを表示するためのサポートが追加されました。詳しくは、943 ページの『IBM MQ のキュー・マネージャーのキュー統計モニターの有効化』 を参照してください。

データ・コレクターの機能拡張

J2SE データ・コレクター

J2SE データ・コレクターは、以下の機能によって拡張されています。

- OpenJDK バージョン 9、10、および 11 に対するサポートが追加されました。
- Windows Server 2019 オペレーティング・システム (Datacenter エディションおよび Standard エディション) のサポートが追加されました。
- トランザクション・トラッキングおよび診断データのモニター用の、J2SE アプリケーション固有のクラスおよびメソッドを自動検出する機能が追加されました。

Selenium IDE 3.2.X および 3.3.X (シンセティック・スクリプト用)

サブスクリプションに IBM Website Monitoring on Cloud アドオンが含まれる場合に、Selenium IDE バージョン 3.2X および 3.3.X がサポートされるようになりました。スクリプトおよびテスト・スイートは、古いバージョンの Selenium IDE で使用される `.html` 形式ではなく、`.side` 形式で保存されます。既存の `.html` スクリプトがある場合は、引き続きそれを使用できます。場合によっては、`.html` スクリプトを編集するか、新しい `.side` 形式で再記録する必要があります。

詳しくは、1028 ページの『Website Monitoring でのシンセティック・トランザクションおよびシンセティック・イベントの管理』のサブトピック (1029 ページの『シンセティック・スクリプトの記録』、1031 ページの『複雑なスクリプトの構造化』、および 1032 ページの『旧バージョンの Selenium IDE からのスクリプトの更新』) を参照してください。

2018 年 12 月

新規エージェント

Monitoring Agent for IBM Cloud

Monitoring Agent for IBM Cloud は、IBM Cloud (Softlayer) アカウントから仮想マシンのインベントリーおよびメトリックを収集します。IBM Cloud で構成して実行している仮想デバイスの数を追跡するには、IBM Cloud エージェント を使用します。詳細なダッシュボード・ページで各仮想デバイスに割り振られているリソースを確認できます。このページには、デバイスが置かれているデータ・センター、オペレーティング・システム、およびその月の予想されるパブリック・ネットワーク帯域幅などの情報も表示されます。

エージェントの機能拡張

Monitoring Agent for Cassandra

Cassandra エージェントは、以下の機能によって拡張されています。

- モニター対象インスタンスの状態をモニターする `Cassandra_Cluster_Down` という新しいしきい値が追加されました。
- Ubuntu 18.04 オペレーティング・システムのサポートが追加されました。
- SUSE Linux Enterprise Server 15 プラットフォームに対するサポートが追加されました。

Monitoring Agent for Db2

Db2 エージェントは、以下の機能によって拡張されています。

- Db2 エージェントが複数のスタンバイの HADR モニター機能をサポートするようになりました。
- Db2 エージェントがデータベース状況属性の新しい値 `Stopped` をサポートするようになりました。
  `Stopped` 状況は、データベースがアクティブではなく、アクティブな接続はないが、正常で、新規接続を受け入れる準備ができていることを示します。
- Db2 サーバーの詳細を表示するために新しいウィジェット「`Db2 サーバー情報 (Db2 Server Information)`」が追加されました。

- 以下の新しいウィジェットでパートナー・データベースに関する情報を表示するための新しいページ「HADR 状況」–「ローカル・データベース (Local Databases)」 が追加されました。
  - 「HADR データベースの詳細 (HADR Databases Details)」は、パートナー・データベースの重要な属性値を表示する表ウィジェットです。
  - 「ログ・ギャップ (ヒストリー) (Log Gap (History))」は、ログ・ギャップのトレンドと時間を比較して表示するグラフ・ウィジェットです。
  - 「スタンバイ・フラグ状況 (Standby Flag Status)」は、スタンバイのフラグ状況値を表示する表ウィジェットです。
- HADR 環境の 2 次スタンバイ・データベースをモニターするための新しい事前定義しきい値 UDB_HADR_Aux_Standby_Disconnect が追加されました。
- 「上位 5 件のデータベース・メモリー使用量 (Top 5 database memory usage)」ウィジェットは、正確な値を表示するように更新されました。
- Db2 エージェントが以下のプラットフォームをサポートするようになりました。
  - Ubuntu zLinux 18.04
  - x86-64 (64 ビット) 版の SUSE Linux Enterprise Server 15
  - zLinux 版の SUSE Linux Enterprise Server 15
  - Power® Linux リトル・エンディアン版の SUSE Linux Enterprise Server 15

Monitoring Agent for InfoSphere DataStage

InfoSphere DataStage エージェントは、以下の機能によって拡張されています。

- 選択された属性グループのデータ収集を無効にする機能が追加されました。
- ジョブ実行属性グループのデータ収集が最適化されました。
- SUSE Linux Enterprise Server 15 プラットフォームに対するサポートが追加されました。

Monitoring Agent for Internet Services

Internet Service Monitoring エージェントは、Windows 64 ビット版および Linux 64 ビット版プラットフォームをサポートするようになりました。

Monitoring Agent for Microsoft .NET

.NET エージェントは、以下の機能によって拡張されています。

- .NET エージェントは、失敗要求を追跡するようになりました。これらの要求の状況は、「ミドルウェア・トランザクション詳細」ページの「最新要求」グループ・ウィジェットで failed として表示されます。また、「最新エラー」グループ・ウィジェットでは、最近の失敗要求が状況コードとエラーの説明とともにリストされます。
- .NET エージェントは、ASP.NET Identity セッションおよび ASP.NET セッションで利用できるユーザー・データもモニターします。ユーザー・データは、「ミドルウェア・トランザクション詳細」ページの「ユーザー - 上位 5 件」グループ・ウィジェットに表示されます。

Monitoring Agent for MongoDB

MongoDB エージェントは、SUSE Linux Enterprise Server 15 プラットフォームをサポートするようになりました。

Monitoring Agent for NetApp Storage

NetApp ストレージ・エージェントは、以下の機能によって拡張されています。

- 「イベントの詳細」ページに、検索基準に基づいてイベント・データをフィルターに掛ける新しい検索ボックスが追加されました。
- LUN の新しい「詳細」ページが追加されました。
- ユーザーは、「**詳細**」ページで各ストレージ・オブジェクトにマップされた関連デバイスの詳細を確認できるようになりました。

Monitoring Agent for OpenStack
仮想マシン・インスタンスの CPU、メモリー、ディスク、およびネットワーク・インターフェース・コントローラーの使用量など、仮想マシン・インスタンスをモニターするためのサポートが追加されました。

Monitoring Agent for PostgreSQL

- SUSE Linux Enterprise Server 15 に対するサポートが追加されました。
- CPU およびメモリーの属性グループのデータ収集が最適化されました。

Monitoring Agent for RabbitMQ
SUSE Linux Enterprise Server 15 に対するサポートが追加されました。

Monitoring Agent for SAP Applications
SAP エージェントは、以下のプラットフォームをサポートするようになりました。

- SUSE Linux Enterprise Server 15 プラットフォーム
- SAP NW RFC SDK 7.50

Monitoring Agent for Skype for Business Server
Skype for Business Server は、以下の機能によって拡張されています。

- シンセティック・トランザクション・モジュールのシンセティック・トランザクション・コマンドは、構成済みのテスト・ユーザーによって実行可能になりました。この機能を有効にするには、「エージェント構成」パネルで「エージェント構成値の使用」を無効にして、シンセティック・コマンドが実行される「プールの FQDN」の値を指定します。「エージェント構成」パネルの「プールの FQDN」フィールドに指定された ID に対して NewCsHealthMonitoringConfiguration コマンドを使用し、テスト・ユーザーが構成されていることを確認してください。
- ユーザーは、シンセティック・コマンドを無効にすることができるようになりました。特定のコマンドが実行されないようにするには、<CANDLE_HOME>¥tmaitm6 (32 ビット・バージョン) および <CANDLE_HOME>¥TMAITM6_x64 (64 ビット・バージョン) にある LyncSyntheticTrans.exe.config ファイルで、そのコマンド名に対して false を指定してください。

Monitoring Agent for Tomcat

- 新しい属性グループの「クラスター」が追加されました。この属性グループには、クラスターのプロパティーの情報が含まれます。
- 新しいウィジェットの「クラスター情報」が追加されました。このウィジェットでは、属性グループ「クラスター」の情報が表示されます。エージェントが非クラスターの Tomcat セットアップをモニターしている場合は、データは表示されません。
- 構成パネルの変数 *Tomcat Server Port* が追加されました。この変数は、Tomcat サーバーが実行されているポートを表します。この変数のデフォルト値は 8080 です。

Monitoring Agent for VMware VI

- 「コンポーネント」ページは、構成済みの vCenter の IP アドレスまたはホスト名と、エージェントとの接続を表示するように拡張されました。

WebSphere Infrastructure Manager エージェント
WebSphere Infrastructure Manager エージェントは、AIX® をサポートするようになりました。

WebSphere MQ エージェント
WebSphere MQ エージェントは、IBM WebSphere MQ 9.1 でサポートされるようになりました。

データ・コレクターの機能拡張

J2SE データ・コレクター
J2SE データ・コレクターは、以下の機能によって拡張されています。

- Power Linux ビッグ・エンディアン (64 ビット) 版の SUSE Linux Enterprise Server 11 に対するサポートが追加されました。
- Power Linux ビッグ・エンディアン (pLinux BE) (64 ビット) に対するサポートが追加されました。

- Power Linux リトル・エンディアン (pLinux LE) (64 ビット) に対するサポートが追加されました。
- 構成モジュールと Jetty サーバーのモニターが追加されました。

エージェントのプラットフォーム・サポートの拡張
以下のエージェントおよびプラットフォームがサポートされるようになりました。

SUSE Linux Enterprise Server 15 プラットフォーム

- Cassandra エージェント
- DataPower エージェント
- DataStage エージェント
- Db2 エージェント
- HTTP Server
- IBM Integration Bus エージェント
- Linux OS エージェント
- MongoDB エージェント
- OpenStack エージェント
- PostgreSQL エージェント
- Monitoring Agent for RabbitMQ
- SAP エージェント
- WebSphere MQ エージェント

Power Linux

- J2SE データ・コレクター

Ubuntu 18.04

- Cassandra エージェント
- IBM Integration Bus エージェント
- OpenStack エージェント
- Linux OS エージェント
- RabbitMQ エージェント
- WebSphere MQ エージェント

Power 9 のサポート
すべてのエージェントで Power 9 がサポートされるようになりました。

V8.1.4 の 2018 年 10 月の更新の新機能

Cloud Event Management との統合
Cloud Event Management は、サービス、アプリケーション、およびインフラストラクチャーのリアルタイム・インシデント管理を提供します。Cloud Event Management と IBM Cloud Application Performance Management の間での統合により、Cloud APM で生成されるイベントはすべて、Cloud Event Management に送信されるようになりました。

2018 年 9 月

新規エージェントの提供
Monitoring Agent for MQ Appliance
MQ アプライアンス・エージェントは、「MQ アプライアンス」で MQ アプライアンス・レベルに固有のモニター情報 (例えば、CPU、メモリー、ストレージ、センサー、およびキュー・マネージャーの要約情報) を提供します。

エージェントの機能拡張

Monitoring Agent for Db2
Db2 エージェントは、Power Linux ビッグ・エンディアン・オペレーティング・システムをサポートするようになりました。

Monitoring Agent for Hadoop

- Hadoop エージェントは、さらに SmartSense および Druid という 2 つのサービスの状況をモニターするようになりました。
- Hadoop エージェントは、Hortonworks Data Platform (HDP) 3.0.0 をサポートするようになりました。

Internet Service Monitoring エージェントエージェント
Internet Service Monitoring エージェント エージェントの編集機能が拡張されています。構成可能なパラメーターを持つすべてのモニターを編集できます。

Monitoring Agent for MySQL
MySQL エージェントは、MySQL v8.0.11 のモニターをサポートするようになりました。

Monitoring Agent for NetApp Storage
NetApp ストレージ・エージェントは、以下の機能によって拡張されています。

- 「イベントの全体的な要約 (Overall Events Summary)」という新しいウィジェットが「NetApp ストレージ・インスタンス」ページに追加されました。イベントの累積数が表示されます。「イベントの合計」として表されるステータス・バーをクリックすると、重大度やオブジェクトに関係なく、環境全体で発生したすべてのイベントを表示できます。

  また、すべてのオブジェクト・テーブルに「イベント状況」列が追加され、時間と、その次に重大度のレベルに基づいて優先順位が付けられたイベント状況が表示されます。

  「NetApp ストレージ・インスタンス」ページには、グラフではなく「イベントの要約」テーブルが表示されるようになりました。
- 「集約詳細」ページが更新され、選択された集約に関連するデバイスが表示されるようになりました。

Monitoring Agent for SAP Applications
SAP NW RFC SDK 750 に対するサポートが追加されました。

Monitoring Agent for SAP HANA Database
以下の 2 つの機能が追加されました。

- SAP HANA データベースをスタンバイ・モードでモニターできます。
- SAP HANA データベース・エージェントは、Power System 向けのビッグ・エンディアン・プラットフォームをサポートします。

Monitoring Agent for VMware VI

- **HostVFlashManager** のモニターがサポートされるようになりました。
- 「ESX サーバー」ダッシュボードに、CPU 使用率に関する状況が「クリティカル」、「警告」、および「正常」になっている仮想マシンの数が表示されるようになりました。

新規プラットフォーム: Linux on POWER Big Endian
新しいプラットフォームを使用できます。以下のエージェントが Linux on POWER Big Endian でサポートされるようになりました。

- Db2 エージェント
- IBM Integration Bus エージェント
- Linux OS エージェント
- SAP HANA データベース・エージェント
- WebSphere MQ エージェント
- WebSphere Applications エージェント

2018 年 7 月

使用可能な新規エージェント

Monitoring Agent for Sybase Server

Sybase エージェント は、分散データベースの一元的な管理を可能にします。データベース管理者およびシステム 管理者が Sybase サーバー・システムのパフォーマンスを 調べるために必要な情報を収集し、問題を早期に検出し、問題を予防します。

エージェントの機能拡張

Monitoring Agent for Hadoop

- Hadoop サービス (Mahout、Atlas、Falcon など) をモニターするためのサポートが追加されました。
- Hadoop オファリングの Cloudera CDH 5.13 をモニターするためのサポートが追加されました。
- Hadoop オファリングの Hortonworks HDP 2.6.4 をモニターするためのサポートが追加されました。

Monitoring Agent for HMC Base

HMC V8 R8.7.0 がサポートされるようになりました。

Monitoring Agent for HTTP Server

Windows での 64 ビット Apache HTTP Server のサポートが追加されました。

Monitoring Agent for Microsoft .NET

Monitoring Agent for Microsoft .NET が以下のように機能強化されています。

- IIS 応答時間モジュールが、以下の条件を満たす ASP.NET Web フォーム (.aspx ページ) および ASP.NET MVC Razor ビューに対する JavaScript インジェクションを通じて、サブトランザクションおよびレンダリング時間明細をモニターするようになりました。
  - ページが W3C HTML 規格を満たす。
  - 応答ヘッダーに `Content-Type: text/html, application/xml,application/json` がある。
  - 応答コンテンツに *<head>* 要素がある。
- .NET エージェントが APM サーバーの Diagnostic Query Engine (DQE) サービスに詳細なデータをアップロードします。DQE サービスの詳細ダッシュボードが迅速にデータをロードして表示します。
- 応答時間が 500 ミリ秒を超える要求が「低速要求 - 上位 10 件」ウィジェットに表示されたときにトリガーされる新規しきい値の **NET_Slow_IIS_Request_Crit** が追加されました。
- 選択フィルター・ツールが更新され、検索ボックスでアプリケーション・プールのリストからアプリケーション・プールを検索できます。
- .NET エージェント CLR プロファイラー (`CorProfLog.dll`) をロードしたプロセスのリストを提供する、**ProcListCaller** ユーティリティーが 追加されました。

Monitoring Agent for Microsoft SQL Server

- Microsoft SQL Server エージェントが、`koqErrConfig.ini` ファイルの照合設定に基づき、ERRORLOG 構文解析で複数照合をサポートするようになりました。`koqErrConfig.ini` ファイルに有効な照合設定が指定されていないと、デフォルトの重大度レベルを超える重大度レベルのエラーがある場合、デフォルトの英語のエラー・メッセージのみが表示されます。デフォルトの重大度レベルは 17 です。ERRORLOG ファイルを構文解析するときは、`koqErrConfig.ini` ファイルに存在するすべての照合が考慮されます。そのため、`koqErrConfig.ini` ファイルには使用する照合のみを追加してください。ERRORLOG の構文解析時には大文字と小文字が区別されるため、`koqErrConfig.ini` ファイルの照合キーワードの値は、ERRORLOG ファイルや参照 `koqErrConfigSample.ini` ファイルに出現するキーワード値とまったく同じでなければなりません。`koqErrConfig.ini` ファイルに対する変更は、エージェントをアップグレードすると失われるため、必ずバックアップを 行ってからエージェントをアップグレードしてください。

- エージェントに、ユーティリティー・ツール **koqVerifyPermissions.exe** も備わりました。このツールは、Microsoft SQL Server をモニターするための十分な権限を既存の SQL Server ユーザーが持っているかどうかを検査します。既存の SQL Server ユーザーが十分な権限を持っていない場合は、データ収集のための最小限の権限を既存の SQL Server ユーザーに付与するための代替方法として、ユーティリティー・ツール **permissions.cmd** を使用できます。

Monitoring Agent for NetApp Storage

Monitoring Agent for NetApp Storage が以下のように機能強化されています。

- データ・プロバイダーが稼働しているかダウンしているかのエージェント接続状態の詳細を、モニター対象データ・ソースの IP アドレスとともに表示する新しい「コンポーネント」ページが追加されています。個別のステータス・バーは、クリティカル、正常、警告、または不明状態にあるノード、集約、ボリューム、およびディスクの数を表します。
- クラスター、集約、ボリューム、ディスク、および Vserver の主要な特性に焦点を当てて表示する新しい「NetApp ストレージ・インスタンス」ページが追加されています。このページには、「イベントの要約」グラフも表示されます。このグラフには、環境全体の使用可能な各ストレージ・エンティティーまたはオブジェクトで 発生したイベントの数が表示されます。例えば、12 個のボリュームが構成されていて、各ボリュームに重大度が「クリティカル」のイベントが 2 つずつ存在する場合、「イベントの要約」グラフには環境内の使用可能なすべてのボリュームで発生したイベントの総数が示されます。この例の場合、グラフには、X 軸に 1 つのエンティティーとしてプロットされるボリュームに対して、24 個のクリティカル・イベントを示す 1 つの棒が表示されます。
- ネットワーク・ポートの詳細を表示するように「ノードの詳細」ページが更新されています。
- 選択された各ボリュームの関連するスナップ・ミラーの詳細および LUN 数を表示するように「ボリューム詳細」ページが更新されています。
- ネットワーク論理インターフェースに 関する情報を表示するように「Vserver 詳細」ページが更新されています。

Monitoring Agent for Tomcat

Tomcat V9.0.5 サーバーがサポートされるようになりました。

Monitoring Agent for WebSphere MQ

リモート・モニターがサポートされます。エージェントがリモート・キュー・マネージャーのモニター・データを収集できるように 2 つの構成パラメーターが追加されています。ただし、これらの構成パラメーターはローカル・キュー・マネージャーには影響しません。ローカル・キュー・マネージャーをモニターするようにエージェントを 構成する場合は、Enter キーを押してこれらのパラメーターの指定をスキップできます。

エージェントの構成について詳しくは、936 ページの『WebSphere MQ エージェントの構成』を参照してください。

EIF スロットのカスタマイズの機能強化

EIF スロットに複数の属性値およびリテラル値を追加できるようになりました。例えば、使用可能ディスク・スペースが少ないかどうかをテストするしきい値の「空きディスクの割合 (%) は **Disk_Free_Percent** です (Disk free percent is **Disk_Free_Percent**)」というメッセージの代わりに、「空きディスクの割合 (%) は **Disk_Free_Percent**、空き i ノードの割合 (%) は **Inodes_Free_Percent** です (Disk free percent is **Disk_Free_Percent** and inodes free percent is **Inodes_Free_Percent**)」を使用できます。転送メッセージは「空きディスクの割合 (%) は 13、空き i ノードの割合 (%) は 9 です (Disk free percent is 13 and inodes free percent is 9)」のようになります。詳しくは、992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』を参照してください。

2018 年 4 月

新規エージェントの提供

Monitoring Agent for AWS Elastic Load Balancer

Amazon ELB エージェントは、AWS Elastic Load Balancer の正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、AWS Elastic Load Balancer 環境に関して十分な情報に基づいて意思決定するために役立つ、各ロード・バランサー・タイプ (アプリケーション、ネットワーク、およびクラシック) に関する包括的な一連のメトリックが表示されます。

エージェントの機能拡張

Response Time Monitoring エージェント
IBM HTTP Server 応答時間モジュールで、Windows 上の IBM HTTP Server バージョン 7、8、および 9 がサポートされるようになりました。

Monitoring Agent for Node.js
デフォルトで、Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報が Node.js データ・コレクターにより収集されなくなりました。新しい環境変数 *SECURITY_OFF* を指定することで、このデフォルトの動作を変更できます。

Monitoring Agent for Amazon EC2
コンポーネント名にエージェント名が反映されるようになりました。

拡張データ保存のサポートが追加されました。

Monitoring Agent for WebLogic
トランザクション・トラッキングおよび詳細診断が AIX で使用可能になっています。以前は、これらの機能は Linux および Windows でのみ使用可能でした。

トランザクション・トラッキングおよび詳細診断のアノテーションを使用して実装されたサーブレット用の「要求の要約」ドリルダウンが拡張されました。

Monitoring Agent for Skype for Business Server
Windows Server 2016 のサポート。

Monitoring Agent for Sterling File Gateway
エージェントはデフォルトの動作として、失敗したファイル転送のイベントをフェッチします。新しい環境変数 **KFG_ALL_FGEVENTS** に適切な値を指定することで、このデフォルトの動作を変更できます。

Monitoring Agent for Sterling Connect Direct
エージェントのロギング機能が改善されました。詳しくは、トラブルシューティングのセクションを参照してください。

データ・コレクターの機能拡張

Node.js データ・コレクター
デフォルトで、Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報が Node.js データ・コレクターにより収集されなくなりました。新しい環境変数 *SECURITY_OFF* を指定することで、このデフォルトの動作を変更できます。

要確認 : この拡張機能を入手するには、TIBM Cloud Application Performance Management Node.js データ・コレクター 暫定修正 1 を IBM Fix Central からダウンロードして適用する必要があります。詳しくは、暫定修正 1 の README ファイルを参照してください。

J2SE データ・コレクター
J2SE アプリケーションのエントリー・ポイント・クラス (メイン・クラス) および別名のオートディスカバリーのサポートが追加されました。

構成スクリプトを使用して、ローカルでトランザクション・トラッキングおよび詳細診断を有効または無効にできます。

資料の拡充

Application Performance Management Developer Center に作成された Web ページで、各リフレッシュまたはリリースでのエージェントのレベルを調べることができます。詳しくは、Cloud APM リリースでのエージェントのバージョンを参照してください。

「各オファリングのエージェントおよびデータ・コレクターの機能」の表は、読みやすくするために簡潔にしてあります。詳しくは、54 ページの『機能』を参照してください。

2018 年 2 月

新規エージェントの提供

Monitoring Agent for Azure Compute

Azure Compute エージェントは、Azure Compute インスタンスの正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、Azure Compute 環境に関して十分な情報に基づいて意思決定する際に役立つ、包括的な一連のメトリックが表示されます。これらのメトリックには、CPU 使用率、ネットワーク使用状況、およびディスク・パフォーマンスが含まれます。

Monitoring Agent for Sterling Connect Direct

Sterling Connect Direct エージェントを使用して、Sterling Connect Direct サーバーの正常性とパフォーマンスをモニターできます。Sterling Connect Direct サーバーの機能 (ファイル転送アクティビティー、スケジュールされたプロセス、保留キューと待機キューのプロセスなど) をモニターします。このエージェントはリモート・モニターをサポートしており、マルチインスタンスです。

Monitoring Agent for Sterling File Gateway

Sterling File Gateway エージェントは、Sterling File Gateway アプリケーションをモニターします。このアプリケーションは、さまざまなプロトコル、さまざまなファイル命名規則、およびさまざまなファイル形式を使用した内部パートナーと外部パートナーの間のファイル転送に使用されます。また、リモート・モニター機能もサポートします。

エージェントの機能拡張

Monitoring Agent for DataPower

WebSphere MQ エージェントと DataPower エージェントの間のトランザクション・トラッキングがサポートされます。

Monitoring Agent for Db2

リモート・モニターに対するサポートが追加されました。

Monitoring Agent for Hadoop

Hadoop サービス (HBase、MapReduce2、Tez、Ranger など) の状況をモニターするためのサポートが追加されました。

Hadoop オファリングの Cloudera CDH 5.12 をモニターするためのサポートが追加されました。

Monitoring Agent for InfoSphere DataStage

メタデータ・リポジトリーとしての MS SQL に対するサポートが追加されました。

Windows オペレーティング・システムに対するサポートが追加されました。

Monitoring Agent for Tomcat

エージェント・フレームワークを 8.1.4.0-IBM-APM-SERVER-IF0001 パッチでアップグレードすることによって、PLinux のトランザクション・トラッキングおよび詳細サポートが提供されます。

Monitoring Agent for SAP Applications

CCMS 機能の拡張: `idx` ファイル削除の自動化。この自動化は、SAP システムの再始動時にのみ機能します。

Monitoring Agent for Microsoft .NET

IIS 応答時間モジュールを使用することにより、エンド・ユーザー・トランザクションに対するサポートが追加されました。

Monitoring Agent for Skype for Business Server
エージェントの名前が、Monitoring Agent for Microsoft Lync Server から Monitoring Agent for Skype for Business Server に変更されています。

Monitoring Agent for Linux KVM
RHEV-M 4.x に対するサポートが追加されました。

Monitoring Agent for Linux OS
メモリー・アップロード間隔は 1 分に変更されます。

ネットワーク・インターフェースに関連付けられている IP アドレスが、Linux OS のダッシュボードおよび「システム情報」ウィジェットに表示されます。

Monitoring Agent for UNIX OS
メモリー・アップロード間隔は 1 分に変更されます。

データ・コレクターの機能拡張

J2SE データ・コレクター
Spring Boot アプリケーションに対するサポートが追加されました。

Availability Monitoring の機能拡張
Availability Monitoring アドオンを使用して、テストでアクセスできる URL とアクセスできない URL を指定するためのホワイトリストとブラックリストを作成できるようになりました。ホワイトリストとブラックリストは、サード・パーティーのメトリックなど、テスト対象の Web アプリケーションの応答時間に寄与する依存関係およびリソースを制御します。ワイルドカード文字を使用することで、URL をスキーム、ドメイン、またはファイル・タイプでフィルタリングします。

2017 年 12 月

新規エージェントの提供

Monitoring Agent for InfoSphere DataStage

DataStage エージェントを使用すると、DataStage サーバーのリソースの正常性とパフォーマンス (エンジン・サービス、エンジン・システム、ジョブ・アクティビティー、ジョブ実行状況、ジョブ実行の詳細など) をモニターできます。このエージェントではリモート・モニターがサポートされています。

エージェントの機能拡張

Monitoring Agent for Hadoop

Active Directory 鍵配布センター (KDC) を使用する Kerberos SPNEGO ベースの認証で保護される Hadoop クラスターをモニターするためのサポートが追加されました。

鍵配布センター (KDC) として MIT または Active Directory を使用する Kerberos SPNEGO ベースの認証で保護される Hadoop クラスターのホストへの接続をテストするためのサポートが追加されました。

Hadoop オファリングの CDH 5.10 および CDH 5.11 をモニターするためのサポートが追加されました。

Hadoop サービス (Flume、Kafka、Titan、Spark、Knox、Pig、Slider、Solr など) の状況をモニターするためのサポートが追加されました。

Monitoring Agent for HTTP Server
Windows 32 ビットの IBM HTTP Server および Apache HTTP Server に対するサポートが追加されました。

Linux for System z® に対するサポートが追加されました (トランザクション・トラッキングはサポートされていません)。

Linux for System x 上の Oracle HTTP サーバーに対するサポートが追加されました。

Monitoring Agent for IBM Integration Bus
Linux for Power Systems (リトル・エンディアン) のサポートが追加されました。

Monitoring Agent for Microsoft .NET
ODP.NET に対するモニターのサポートが追加されました。

HttpWebRequest.GetResponse() メソッドのメソッド・トレースの詳細が追加されました。

Monitoring Agent for Microsoft SQL Server

SQL Server 2017 に対する許容性サポートが追加されました。

SQL Server Developer エディションの Always On 機能に対するサポートが追加されました。

Monitoring Agent for MySQL
情報スキーマ・テーブルをパフォーマンス・スキーマにマイグレーションするための許容性サポートが追加されました。

情報スキーマの非推奨テーブルに対して、パフォーマンス・スキーマを通じたサポートが追加されました。

Monitoring Agent for Microsoft Internet Information Services
FTP Web サイトのモニターに対するサポートが追加されました。

Monitoring Agent for MongoDB
リモート・モニターに対するサポートが追加されました。

メモリー内ストレージ・エンジンのモニターに対するサポートが追加されました。

Monitoring Agent for OpenStack
OpenStack 認証 V3 API に対するサポートが追加されました。

Monitoring Agent for Oracle Database
エージェントのバージョンが 8.0 に変更されました。

構成パラメーター**「Oracle JDBC JAR ファイル (Oracle JDBC jar file)」**が追加され、構成パラメーター**「Oracle ホーム・ディレクトリー」**および**「Oracle インスタント・クライアントのインストール・ディレクトリー」**が削除されました。

Monitoring Agent for PostgreSQL
リモート・モニターに対するサポートが追加されました。

Monitoring Agent for SAP Applications
SNC 通信に対するサポートが追加されました。

SAP システムのダウンを表す新しいしきい値が追加されました。

Monitoring Agent for SAP NetWeaver Java Stack
SAP NetWeaver アプリケーション・サーバー・インスタンスを復元する機能が追加されました。

Monitoring Agent for Tomcat
Linux for Power Systems (リトル・エンディアン) に対するサポートが追加されました (リソースのモニターのみ)。

Monitoring Agent for VMware VI
ESX サーバーの概要ページにネットワーク・サマリーとディスク・カウントが追加されました。

クラスターのサマリー・ページにイベント・グループ・ウィジェットが追加されました。

Monitoring Agent for WebSphere Applications
Linux for Power Systems (リトル・エンディアン) および Linux for System z のトランザクション・トラッキングに対するサポートが追加されました。

要確認 : Linux for Power Systems (リトル・エンディアン) および Linux for System z でトランザクション・トラッキングがサポートされるようにするには、以下のステップを実行します。

1. エージェントのインストール・イメージをダウンロードします。
2. WebSphere Applications エージェント をインストールします。
3. WebSphere Applications エージェント暫定修正 2 を Fix Central からダウンロードします。
4. 暫定修正の readme ファイルに従って修正を適用します。

Monitoring Agent for WebSphere MQ

「MQ サービス状況」グループ・ウィジェットが追加され、MQ サービスの詳細を確認できるようになりました。

Linux for Power Systems (リトル・エンディアン) のサポートが追加されました。

データ・コレクターの機能拡張

Liberty データ・コレクター

Liberty データ・コレクターによって登録される管理対象システム名 (MSN) が変更され、ホスト名と Liberty サーバー名が反映されるようになりました。このデータ・コレクターの新しい MSN は、BI:*servername_hostname_md5*:BLP (*md5* は MD5 に基づくローカル・アプリケーション GUID) です。*servername_hostname_md5* の長さは 25 文字です。

要確認 : この拡張機能を入手するには、IBM Cloud Application Performance Management Liberty データ・コレクター暫定修正 1 を Fix Central からダウンロードして適用する必要があります。

J2SE データ・コレクター

J2SE アプリケーションのトランザクション・トラッキングに対するサポートが追加されました。

資料の拡充

環境の準備を進めやすくするために、エージェントおよびデータ・コレクターが使用するデフォルト・ポートに関する情報が提供されています。85 ページの『エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート』を参照してください。

Cloud APM エージェントの管理対象システム名 (MSN) に関する情報が提供されています。MSN のホスト名ストリングを変更する方法についての説明も記載されています。171 ページの『管理対象システム名』を参照してください。

非管理者ユーザーとしてエージェントを実行する方法、および非管理者ユーザーとしてエージェントを実行するために必要な権限に関する情報が、以下のエージェントの構成に関するトピックで提供されています。

- Microsoft .NET エージェント
- Microsoft Active Directory エージェント
- Microsoft Exchange Server エージェント
- Skype for Business Server エージェント
- Microsoft SharePoint Server エージェント
- Microsoft SQL Server エージェント
- Tomcat エージェント

2017 年 8 月

IBM Cloud Application Performance Management, Availability Monitoring

Availability Monitoring アドオンは、世界の複数のポイント・オブ・プレゼンス (Point of Presence: PoP) から Web アプリケーションの拡張シンセティック・モニターを実現します。一定の間隔でユーザーの振る舞いを模倣するシンセティック・テストを作成します。パブリック・ポイント・オブ・プレゼンスからテストを実行するか、ローカル・サーバーまたはプライベート・サーバーに独自のカスタム・ポイント・オブ・プレゼンスをダウンロードしてデプロイします。Availability Monitoring ダッシュボードを使用して、アプリケーションの可用性、パフォーマンス、およびアラートをグラフ、明細表、およびマップ・ビューを使用してモニターします。ウォーターフォール式分析を使用して、パフォーマンスおよび可用性の問題がいつ発生したかを識別し、それらの問題の理由を特定します。

IBM API Connect® のモニター

Cloud APM エージェントおよびデータ・コレクターは、IBM API Connect 環境のモニターをサポートするようになりました。対応するエージェントおよびデータ・コレクターをデプロイして、環境内のコンポーネントの正常性とパフォーマンスを表示できます。リソース・モニターと詳細診断のデータに加えて、トランザクション・トラッキングのデータも使用できます。これにより、IBM API Connect 環境に関するトポロジー情報を表示できます。詳しくは、90 ページの『シナリオ: IBM API Connect のモニター』を参照してください。

OS サポート

Linux for System z
特定のモニター・エージェント (Linux OS、WebSphere アプリケーション、Db2、WebSphere MQ、IBM Integration Bus、Tomcat、および Response Time Monitoring) に対する Linux for System z のサポートが追加されました。

Linux for Power Systems (リトル・エンディアン)
Linux OS、WebSphere アプリケーション、および Db2 の各モニター・エージェントについて、Linux for Power Systems (リトル・エンディアン) に対するサポートが追加されました。

Linux for System x
Liberty データ・コレクターをサポートするために Linux on System x が追加されました。

IBM i OS エージェント・サポート
IBM i OS エージェントに関するデータを Cloud APM コンソールに表示できるようになりました。このエージェントは IBM Tivoli Monitoring V6 エージェントであり、V8.1.4 リリースでも V6 エージェントとして残されています。ハイブリッド・ゲートウェイを使用してこのエージェントのデータを取得し、Cloud APM サーバーに送信できます。これにより、このエージェントのモニター・データとイベントを Cloud APM コンソールに表示できます。IBM i OS エージェントについて詳しくは、Hybrid Gateway supported agents (APM Developer Center) を参照してください。

使用可能な新規エージェント

Monitoring Agent for OpenStack

OpenStack エージェントを使用して、OpenStack アプリケーションの正常性とパフォーマンスをモニターし、API エンドポイント、SSH サーバー接続、プロセス、ハイパーバイザーに関する情報などを表示できます。

使用可能な新規のデータ・コレクターと拡張されたデータ・コレクター
これらのデータ・コレクターを使用して、IBM Cloud、オンプレミス、またはこの両方で以下のアプリケーションの正常性とパフォーマンスをモニターできます。

J2SE データ・コレクター
J2SE データ・コレクターを使用して、Java アプリケーションの正常性とパフォーマンスをモニターし、応答時間、スループット、要求コンテキスト、要求のメソッド・トレースなどの診断データを表示できます。

Liberty データ・コレクター
Liberty データ・コレクターは、IBM Cloud 環境内の Liberty プロファイルおよび Linux for System x 内のローカル Liberty プロファイルの両方をモニターします。

Node.js データ・コレクター
Node.js データ・コレクターは、IBM Cloud アプリケーションおよびオンプレミス・アプリケーションの両方をモニターします。リソース・データと診断モニター・データの両方 (リソース使用状況、スループット、要求とメソッドに関する詳細情報など) を表示できます。

Python データ・コレクター
Python データ・コレクターは、IBM Cloud アプリケーションをモニターします。リソース・データと診断モニター・データの両方 (リソース使用状況、スループット、要求とメソッドに関する詳細情報など) を表示できます。

Python エージェントは、Cloud APM V8.1.4 のエージェント・インストール・パッケージから削除されました。Python データ・コレクターのみを使用して、Python アプリケーションをモニターできます。

Ruby データ・コレクター
Ruby データ・コレクターは、IBM Cloud アプリケーションのみをモニターします。リソース・データと診断モニター・データの両方 (リソース使用状況、スループット、要求とメソッドに関する詳細情報など) を表示できます。

エージェントの機能拡張

Monitoring Agent for Amazon EC2

- エージェントは、スケジュールされたイベントの NULL 終了日を正しく処理します。

- Amazon EC2 エージェントと Amazon Web サービスの間のフォワード・プロキシーに対するサポートが追加されました。

Monitoring Agent for Citrix Virtual Desktop Infrastructure

- エージェントが Linux システムにインストールされている場合でも Windows のイベントと PowerShell のメトリックをモニターする機能が追加されました。
- 「**VDA マシンの詳細**」ページからアクセス可能な「**VDA セッション**」ページが追加されました。
- 「**VDA マシンの詳細**」ページに「マシン・メトリック (Machine Metrics)」ウィジェットが追加されました。
- エージェントが分散環境で Desktop Delivery Controller (DDC) フェイルオーバーを処理できるよう、DDC 構成が拡張されました。

Monitoring Agent for Db2

- Linux for System z のサポートが追加されました。

Monitoring Agent for Hadoop

- Hadoop オファリングの Hortonworks HDP 2.6 および Cloudera CDH 5.9、5.10、5.11 をモニターするためのサポートが追加されました。
- ZooKeeper、Sqoop、Hive、HDFS、YARN、Ambari Metrics、Oozie などの Hadoop サービスの状況をモニターするためのサポートが追加されました。
- MIT Kerberos V5 鍵配布センター (KDC) のみを使用する Kerberos SPNEGO ベースの認証で保護される Hadoop クラスターをモニターするためのサポートが追加されました。

Monitoring Agent for IBM Integration Bus

Linux for System z に対するサポートが追加されました (トランザクション・トラッキングはサポートされていません)。

Monitoring Agent for JBoss

- Advanced Agents オファリングでの JBoss エージェントのトランザクション・トラッキングと詳細診断構成プロセスが簡素化されました。
- 「**ガーベッジ・コレクションの詳細 (Garbage Collection Detail)**」ページに 2 つのダッシュボード・ウィジェットが追加されました。1 つのウィジェットは、最後のガーベッジ・コレクション以降に解放されたヒープ・メモリーの量を示し、もう 1 つのウィジェットは、Eden/Survivor/Tenured (Old Gen) ヒープ・メモリー・プール・サイズの履歴を示します。

Monitoring Agent for Linux OS

Linux for Power Systems (リトル・エンディアン) のサポートが追加されました。

Monitoring Agent for Skype for Business Server

- 「Lync Server の概要 (Lync Server Overview)」ダッシュボードに、フロントエンド登録状況と低品質通話の質を表示する「Lync 使用状況の要約 (Lync Usage Summary)」グループ・ウィジェットが追加されました。
- Microsoft Lync Server の使用状況の詳細を表示するダッシュボードが追加されました。

Monitoring Agent for SAP NetWeaver Java Stack

SAP NetWeaver Java Stack エージェントのダッシュボードに以下の機能拡張が追加されています。

- トランザクション・トラッキング・データと診断データを収集して表示するためのデータ・セット、グループ・ウィジェット、およびページが追加されました。
- Windows 2016 システムでエージェントをインストールおよび 構成するためのサポートが追加されました。
- ユーザーによってアプリケーションに対して行われた要求のうち、応答時間が長い上位 5 個の要求に関する情報を提供するために、「SAP NW Java インスタンス」ダッシュボードに「応答時間別の上位 5 個の最低速要求」グループ・ウィジェットが追加されました。

- 「応答時間別の上位 5 個の最低速要求」グループ・ウィジェットに表示される要求に関する診断情報は、「要求インスタンス」ページで要求をクリックして表示できます。「応答時間別の上位 5 個の最低速要求」グループ・ウィジェット内の要求に関する診断情報を表示するためのサポートが追加されました。
- 「応答時間別の上位 5 個のユーザー・セッション」グループ・ウィジェットは削除されました。

Monitoring Agent for MongoDB

プライマリー・ノードの障害発生時に MongoDB クラスターまたは複製のセットアップをモニターするためのサポートが追加されました。

Monitoring Agent for MySQL

MySQL リソースをリモートでモニターするためのデータ・セットおよび構成パラメーターが追加されました。

Monitoring Agent for NetApp Storage

- クラスターおよび Vserver の要約情報を表示するように「コンポーネント」ページが更新されました。
- クラスターに関する情報を表示するように「NetApp ストレージ・インスタンス」ページが更新されました。

Monitoring Agent for Node.js

Node アプリケーションのメトリック (Appmetrics) を活用するために、Node.js エージェントに以下の機能拡張が追加されています。

- ガーベッジ・コレクションの詳細を表示するための新しいダッシュボードとグループ・ウィジェット
- イベント・ループの詳細を表示するための新しいダッシュボードとグループ・ウィジェット

Monitoring Agent for PostgreSQL

- Windows システムでエージェントをインストールおよび 構成するためのサポート。
- PostgreSQL V9.6 をモニターするためのサポート。
- バッファーのヒット率がゼロのときに状況がクリティカルにならないように、「**状況の概要**」ページが更新されました。

Monitoring Agent for SAP HANA Database

ライセンス有効期限までの残りの日数をモニターするために、「ライセンス有効期限までの日数」属性と HANA_License_Expiry_Crit_SYS しきい値および HANA_License_Expiry_Warn_SYS しきい値が追加されました。

Monitoring Agent for Tomcat

- Linux for System z のサポートが追加されました。
- トランザクション・トラッキングと詳細診断用にデータ・セット、ダッシュボード、およびグループ・ウィジェットが追加されました。

Monitoring Agent for VMware VI

SSH 状況を表示するように、「サーバーの要約」ダッシュボードの「ESX サーバー」グループ・ウィジェットが更新されました。

Monitoring Agent for WebLogic

Advanced Agents オファリングでのエージェントにトランザクション・トラッキングと詳細診断が追加されました。

Monitoring Agent for WebSphere Applications

Linux for System z に対するサポートが追加されました (トランザクション・トラッキングはサポートされていません)。

Monitoring Agent for WebSphere MQ

Linux for System z に対するサポートが追加されました (トランザクション・トラッキングはサポートされていません)。

チャネルおよびキューの長期間ヒストリー・データがサポートされています。チャネルまたはキューの統計データを収集するようにキュー・マネージャーを構成した後で、チャネルまたはキューの長期間ヒストリー・データ収集を有効にするようにエージェントを構成できます。収集された長期間ヒストリー・データを表示するための事前定義されたダッシュボードまたはウィジェットはありませんが、「**属性の詳細**」タブを使用して、カスタム表にある収集されたデータを照会できます。

Response Time Monitoring エージェント

- Windows 32 ビットの IBM HTTP Server および Apache HTTP Server に対するサポートが追加されました。
- 「**エージェント構成**」ページでアプリケーションのユーザー・トラッキングを構成するためのサポートが追加されました。
- 「**エージェント構成**」ページでアプリケーションのセッション・トラッキングを構成するためのサポートが追加されました。

可視化の向上

カスタム・ビュー

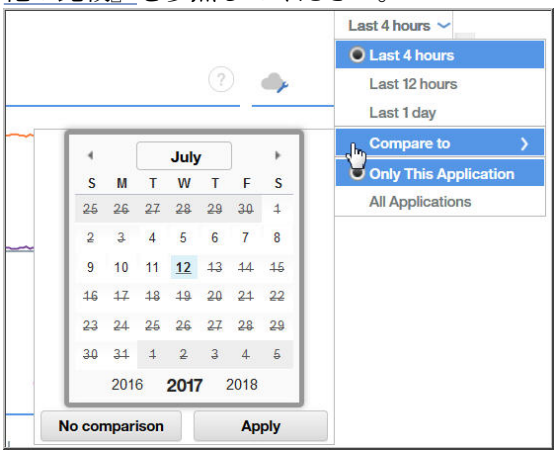
IBM Cloud Application Business Insights Universal View を使用して、モニターしているアプリケーションのカスタマイズされたページを作成できます。「カスタム・ビュー」タブでは、ページ用に既存のテンプレートを使用するか、カスタマイズされたテンプレートを作成することができます。さまざまなグラフおよびメトリックのオプションから選択して、要件に従ってデータをモニターするウィジェットを作成できます。

Universal View を使用して、さまざまなエージェントからのデータをモニターするダッシュボードを作成できます。カスタマイズされたページ・データを生データ・ファイルにエクスポートできます。

詳しくは、1112 ページの『カスタム・ビュー』を参照してください。

過去の日付のデータを比較するためのカレンダー

履歴データを示す折れ線グラフを表示しているときに、過去の日付からの時間範囲を比較するための時間セレクター・オプションを選択すると、カレンダーが開きます。比較対象として選択できない日には取り消し線が引かれています。詳しくは、1091 ページの『メトリックの調整および時間変化の比較』を参照してください。



Agent Builder の機能拡張

Java Database Connectivity (JDBC) データベースのデータをモニターするエージェントを作成する際に、そのデータベース内の正当な値とのオーバーラップを回避するために、「エラー」、「欠落データ」、および「値なし」に設定されている列挙値を変更できます。

Agent Builder の「**設定**」ウィンドウで、JAR ファイルの時刻認証局を設定できます。デフォルトの時刻認証局の署名証明書が期限切れになった場合、新しい認証局を設定することで、JAR ファイルの検証を続行できます。

統合の拡張

イベント用にカスタマイズ可能な EIF スロット
イベント転送を構成した場合は、Netcool/OMNIbus などの受信側に送信されるイベントのために、基本 EIF スロット・メッセージをカスタマイズし、カスタム EIF スロットを作成することができます。しきい値エディターには、転送されるイベントにイベントをマップする方法をカスタマイズするための新しい「**EIF イベントを転送しますか?**」フィールドと「**EIF スロット・カスタマイズ**」ボタンがあります。詳しくは、992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』を参照してください。

複数のハイブリッド・ゲートウェイ
前のリリースでは、ハイブリッド・ゲートウェイを、1 つのハブ Tivoli Enterprise Monitoring Server を持つ単一の IBM Tivoli Monitoring ドメインにインストールできました。現行リリースでは、ハイブリッド・ゲートウェイを複数の Tivoli Monitoring ドメインにインストールできるようになりました。Cloud APM コンソールの「**拡張構成**」ページにあるハイブリッド・ゲートウェイ・カテゴリーは、専用の「**ハイブリッド・ゲートウェイ・マネージャー**」ページに移動されました。ここで、複数の Tivoli Monitoring ドメインからの管理対象システムをモニターするための ハイブリッド・ゲートウェイ・プロファイル (各ドメインに 1 つのプロファイル) を作成および編集できます。詳しくは、953 ページの『ハイブリッド・ゲートウェイ』を参照してください。

スケーラビリティーの向上
Cloud APM からモニターできる管理対象システムの最大数が 4,000 から 10,000 に増加しました。

以前のリリース

以前のリリースでの新しいフィーチャー、機能、および対象については、以下の新機能 のトピックを参照してください。

- 28 ページの『新機能 (2017 年 4 月)』
- 35 ページの『新機能 (2016 年 9 月)』
- 新機能 (2016 年 4 月)

新機能 (2017 年 4 月)

Cloud APM の 2017 年 4 月リリースで、新しいフィーチャー、機能、および対象が使用可能になりました。

新しい Application Performance Management Developer Center
APM Developer Center は、APM 製品のリソース (ブログ、ビデオ、資料、サポート、イベント、IBM Marketplace など) にアクセスできる中央の場所です。Cloud APM コンソールの ? 「**ヘルプ**」メニューに、Application Performance Management Developer Center への便利なリンクがあります。

製品名の変更と簡素化

IBM Performance Management on Cloud は、IBM Cloud Application Performance Management に名前変更されました。コンポーネント名も変更されました。例えば、Cloud APM コンソールおよび Cloud APM サーバーは、以前のリリースでは Performance Management コンソールおよび Performance Management サーバーという名前でした。

IBM Performance Management on Cloud サブスクリプションのオファリングが、以下のように統合および名前変更されました。

2016 年 10 月以前のリリースでのオファリング名	**2017 年 3 月 以降のオファリング名**
Monitoring on Cloud	Cloud APM, Base
Application Performance Management Advanced on Cloud	Cloud APM, Advanced

以下のように、一部の製品拡張が統合および名前変更されました。

2016 年 10 月以前のリリースでの拡張名	2017 年 3 月 以降の拡張名
Base Extension Pack (Hadoop エージェント)	Base Extension Pack (新たに Cassandra エージェントおよび Microsoft Office 365 エージェントが追加)
Advanced 拡張パック (SAP HANA データベース・エージェントおよび SAP NetWeaver Java Stack エージェント)	Advanced Extension Pack (新たに RabbitMQ エージェントが追加)

OS サポート

Windows 2016 オペレーティング・システム

Windows 2016 オペレーティング・システムのサポートが追加されました。詳しくは、すべてのエージェントの Software Product Compatibility Report (SPCR) (http://ibm.biz/agents-pm-systemreqs) を参照してください。
レポートの Windows セクションでご使用のオペレーティング・システムを見つけ、コンポーネント列のアイコンをクリックしてサポートされるエージェントのリストを 表示します。

使用可能な新規拡張パック

IBM Cloud Application Performance Management z Systems® Extension Pack

z Systems Extension Pack により、Cloud APM オファリング内の IBM OMEGAMON® エージェントがサポートされるようになります。OMEGAMON エージェントのデータは、ハイブリッド・ゲートウェイによって Cloud APM サーバーに送信されます。ハイブリッド・ゲートウェイは、OMEGAMON エージェントが接続されている IBM Tivoli Monitoring インフラストラクチャーから、OMEGAMON エージェントのデータとイベントを取得します。結果として、Cloud APM コンソールで OMEGAMON エージェントのモニター・データおよびイベントを表示できます。

Cloud APM z Systems Extension Pack は、いずれかの Cloud APM オファリングを所有している場合に入手できます。

この拡張パックを Cloud APM と統合するには、965 ページの『OMEGAMON との統合』のステップを実行します。

新しく使用可能なエージェントとデータ・コレクター

Monitoring Agent for Cassandra

Cassandra エージェントを使用して、Cassandra クラスター・リソース (ノード、キースペース、カラム・ファミリーなど) の正常性とパフォーマンスをモニターできます。

Monitoring Agent for Microsoft Office 365

Microsoft Office 365 エージェントを使用して、Office 365 リソース (Office 365 サブスクライブ済みサービス、Office 365 ポータル、メールボックス・ユーザー、SharePoint サイト、OneDrive ストレージなど) の正常性およびパフォーマンスをモニターできます。

Monitoring Agent for NetApp Storage

NetApp Storage エージェントを使用して、NetApp OnCommand Unified Manager (OCUM) によって NetApp ストレージ・システムの正常性、可用性、およびパフォーマンスをモニターできます。モニター・エージェントは、以下のタスクを実行します。

- パフォーマンスの低いストレージ・システム・オブジェクトの識別
- フォーカル・ポイントでの OCUM サーバーを使用した、ディスカバリーおよびモニターの実行

Monitoring Agent for RabbitMQ

RabbitMQ エージェントを使用して、RabbitMQ クラスター・リソース (クラスターのノード、キュー、チャネルなど) の正常性およびパフォーマンスをモニターできます。

Bluemix® アプリケーション用のデータ・コレクター

Bluemix アプリケーション用のデータ・コレクターを使用して、Bluemix 上の次のタイプのアプリケーションの正常性とパフォーマンスをモニターできます。

- Liberty アプリケーション
- Node.js アプリケーション
- Python アプリケーション
- Ruby アプリケーション

リソース・データと診断モニター・データの両方 (リソース使用状況、スループット、要求とメソッドに関する詳細情報など) を表示できます。

Monitoring Agent for Siebel

Siebel エージェントを使用して、Siebel リソース (Siebel 統計、ユーザー・セッション、コンポーネント、タスク、アプリケーション・サーバー、Siebel ゲートウェイ・ネーム・サーバー、プロセス CPU とメモリーの使用量、ログ・イベントのモニターなど) の正常性とパフォーマンスをモニターできます。

エージェントの機能拡張

Monitoring Agent for Amazon EC2

Amazon EC2 エージェントに以下の機能拡張が追加されています。

- タグ名が使用できるときにインスタンス ID をタグ名に置換
- タグ名に基づくデータのフィルター処理とグループ化を許可

Monitoring Agent for Db2

Hadoop エージェントで以下の機能拡張が追加されています。

- Linux on Power Little Endian (pLinux LE) のサポート
- モニター対象インスタンスの Db2 エージェントのすべての属性のデータを表示するために、Db2 ユーザーに特権を付与するためのスクリプト・ファイルを追加

Monitoring Agent for Hadoop

Hadoop エージェントで以下の機能拡張が追加されています。

- Windows 2016 および AIX 7.2 システムでエージェントをインストールおよび 構成するためのサポートを追加
- Hadoop オファリングの Hortonworks HDP 2.5、Cloudera CDH 5.6、5.7、5.8、および IBM BigInsights 4.2 をモニターするためのサポートを追加
- エージェントの構成時に指定する Hadoop デーモンへの接続を検証するための「接続のテスト」ボタンを追加
- 構成の時間と複雑性を低減するためにエージェント構成プロセスを改善。以下の前提条件および構成タスクが必要なくなったため、構成が簡素化されました。
  - Hadoop クラスターの各ノードでのプラグインのインストール
  - `hadoop-metrics2.properties` ファイルの構成と更新
  - `hadoop-metrics2.properties` ファイルの構成後の Hadoop デーモンの再始動
  - クラスター内のすべての DataNode および NodeManager の構成
  - ノードをクラスターに追加したときのエージェントの再始動

Monitoring Agent for JBoss

JBoss エージェントに以下の機能拡張が追加されています。

- Advanced Agents オファリングにトランザクション・トラッキングと詳細モニターを追加
- データ・ソースのメトリックをモニターするためのダッシュボード・ページを追加
- JBoss オファリング(WildFly 8.x/9.x/10.x、JBoss EAP 7.x、JBoss AS 7.x) をモニターするためのサポートを追加

- Windows オペレーティング・システム上でエージェントを実行するためのサポートを追加

Monitoring Agent for Linux KVM
Linux KVM エージェントのダッシュボードに以下の機能拡張が追加されています。

- 「最大スケジューリング・メモリー (GB)」KPI および「ライブ・スナップショット」KPI が表示されるように、「ホスト、クラスター、およびストレージ」ページの「ホスト」グループ・ウィジェットを更新
- ストレージ・プール内のディスクおよびディスク・スナップショットに関する詳細を表示する「ストレージの詳細」ページを追加
- ネットワークで送受信された合計データ (GB) の履歴情報を表示するために、「ホスト詳細」ページに「ネットワーク送信/受信データ (GB)」グループ・ウィジェットを追加

Monitoring Agent for Linux OS
Linux OS エージェントに以下の機能拡張が追加されています。

- Linux on Power Little Endian (pLinux LE) のサポート

Monitoring Agent for Microsoft Exchange Server
Microsoft Exchange Server エージェントのダッシュボードで以下の機能拡張が追加されています。

- 「到達可能性」データ・セットでインバウンド時およびアウトバウンド時の属性を追加
- 到達可能性の詳細を表示するページおよびグループ・ウィジェットを追加
- 到達可能性のイベントしきい値を追加
- エージェントを Exchange Server 2016 と Windows Server 2016 システムでインストールおよび構成するためのサポートを追加

Monitoring Agent for Microsoft Internet Information Services
Microsoft IIS エージェントに以下の機能拡張が追加されています。

- エージェントを Microsoft Windows Server 2016 システムでインストールおよび 構成するためのサポートを追加

Monitoring Agent for Microsoft Active Directory
Microsoft Active Directory エージェントで以下の機能拡張が追加されています。

- グループ・ポリシー・オブジェクト、Netlogon、ローカル・セキュリティー機関、および LDAP の詳細を表示するためのグループ・ウィジェットとページを追加
- 「**属性の詳細**」タブで表示できる次のデータ・セットを追加
  - サービス・データ・セット
  - 複製
  - ファイル複製サービス
  - 移動または削除した組織単位
  - LDAP 属性
  - セキュリティー・アカウント・マネージャー
  - DFS
  - アドレス帳
  - イベント・ログ
  - パスワード設定オブジェクト
- ADFS、ADFS プロキシー、および非同期スレッド・キューのデータ・セットを追加
- ADFS と ADFS プロキシーの詳細を表示するためのグループ・ウィジェットとページを追加
- エージェントを Windows Server 2016 システムでインストールおよび 構成するためのサポートを追加しました。

Monitoring Agent for Microsoft .NET
Microsoft .NET エージェントのダッシュボードで以下の機能拡張が追加されています。

- データベース呼び出しの応答時間、スレッド・カウント (高) の .NET プロセスの状況、およびジャストインタイム (JIT) コンパイルの失敗を表示するように、「コンポーネント」ページの MS .NET 状況グループ・ウィジェットを更新
- JIT コンパイルの詳細、データベース呼び出しの詳細、選択した .NET プロセスの GC の処理と固定オブジェクト・コレクション、スレッドの競合率、およびスレッド・キューの長さを表示するために、データ・セット、ページ、およびグループ・ウィジェットを追加
- JIT の失敗、.NET 要求の失敗、低速コマンド、ガーベッジ・コレクション、および .NET プロセス内のアクティブ・スレッド数に関するイベントしきい値を追加

Monitoring Agent for Microsoft SQL Server
Microsoft SQL Server エージェントのダッシュボードに以下の機能拡張が追加されています。

- Microsoft SQL Server のパフォーマンス統計に従って、上位 10 個のキャッシュされた照会プランを表示するために、「サーバー・パフォーマンス - 詳細」ページに「負荷が大きな照会 (Expensive Queries)」グループ・ウィジェットを追加
- Microsoft SQL Server 2016 をモニターするためのサポートを追加
- Microsoft SQL Server エージェントを Microsoft Windows Server 2016 システムでインストールおよび構成するためのサポートを追加
- 「MS SQL エラー・イベントの詳細」属性グループのデータを収集する前にエージェントが待機する時間間隔 (秒) を設定するために、新規 *COLL_ERRORLOG_RECYCLE_WAIT* 環境変数を追加

Monitoring Agent for MongoDB
MongoDB エージェントのダッシュボードに以下の機能拡張が追加されています。

- MongoDB インスタンスの数およびその状況を表示するように、「コンポーネント」ページを更新
- MMAPv1 および WiredTiger ストレージ・エンジンの詳細を表示するページを追加
- 選択したホストのキューに入れられた操作、アクティブ接続、データ・フロー、およびデータ・アクセスに関するカーソルの詳細および履歴データを表示するために、「入出力情報」ページを追加
- バージョン 2.x およびバージョン 3.x 以降のロックの詳細を表示するページを追加
- 複製メンバー、オペレーション・ログ、および複製の遅延の履歴データに関する詳細と、オペレーション・ログによって使用されているスペースを表示する「複製の詳細 (Replication Details)」ページを追加

Monitoring Agent for Node.js
Node アプリケーションのメトリック (Appmetrics) を活用するために、Node.js エージェントに以下の機能拡張が追加されています。

- ガーベッジ・コレクションの詳細を表示するための新しいダッシュボードとグループ・ウィジェットを追加
- イベント・ループの詳細を表示するための新しいダッシュボードとグループ・ウィジェットを追加

Monitoring Agent for PostgreSQL
PostgreSQL エージェントに以下の機能拡張が追加されています。

- エージェントを Windows システムでインストールおよび 構成するためのサポートを追加しました。
- PostgreSQL V9.6 をモニターするためのサポートを追加
- バッファーのヒット率がゼロのときに状況がクリティカルにならないように、「状況の概要」ページを更新

Monitoring Agent for SAP NetWeaver Java Stack
SAP NetWeaver Java Stack エージェントに以下の機能拡張が追加されています。

- トランザクション・トラッキング・データと診断データを収集して表示するためのデータ・セット、グループ・ウィジェット、およびページを追加

- エージェントを Windows 2016 システムでインストールおよび構成するためのサポートを追加しました。

Monitoring Agent for Synthetic Playback
Synthetic Playback エージェントに以下の機能拡張が追加されています。

- Synthetic Playback エージェントには、シンセティック・トランザクションのための新しいフィルター機能が含まれています。Synthetic Script Manager で、指定した URL とドメインへの要求を除外するか、含める、シンセティック・トランザクションのブラックリストとホワイトリストを構成します。ブラックリストとホワイトリストを使用することで、アプリケーションの応答時間に影響する依存関係 (サード・パーティーのメトリックなど) をフィルターに掛けて、除外するか含めます。

Monitoring Agent for Tomcat
Tomcat エージェントに以下の機能拡張が追加されています。

- Tomcat エージェントを Windows システムと SUSE Linux Enterprise 12 システムでインストールおよび構成するためのサポートを追加

Monitoring Agent for WebSphere Applications
WebSphere Applications エージェント に以下の機能拡張が追加されています。

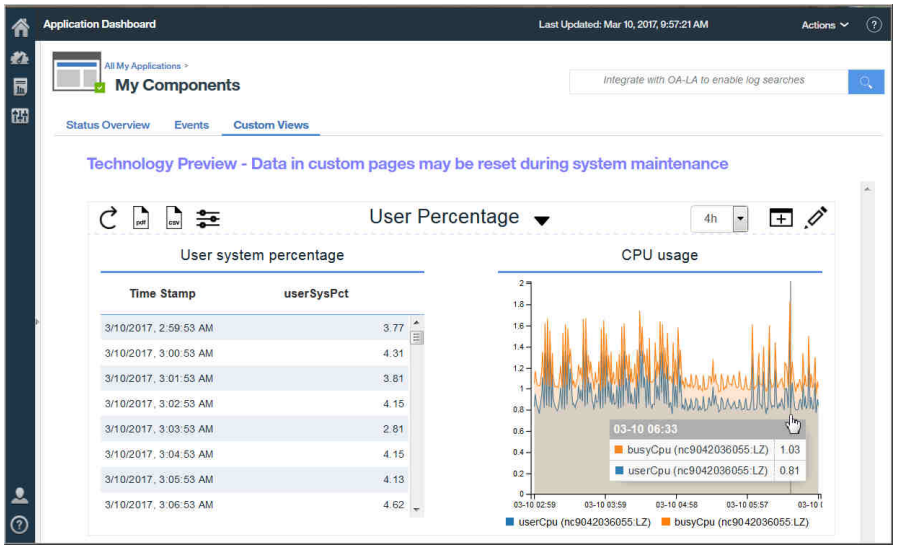
- Linux on Power Little Endian (pLinux LE) がサポートされます。(トランザクション・トラッキングは、pLinux LE システムではサポートされません。)
- IBM WebSphere Application Server traditional V9 のサポートを追加
- 各ヒープ・ダンプのヒープ使用量情報を確認することにより、潜在的なメモリー・リークを診断するのに役立つ「メモリー分析」ダッシュボードを追加。このダッシュボードにデータを含めるには、診断モードを有効にする必要があります。
- アプリケーション状況モニター用のイベントしきい値を作成するためのアプリケーション正常性状況データ・セットを使用するサポートを追加。この使用量のためデータ収集は、デフォルトでは無効になっています。イベントしきい値の作成前に、データ・コレクターのプロパティー・ファイルを変更して有効にする必要があります。
- データ・コレクターの手動構成の簡素化。WebSphere Applications Server の場合、WebSphere 管理コンソールで、アプリケーション・サーバー用にいくつか JVM 引数と変数を追加する必要があるのみです。Liberty の場合、サーバー用の 3 つのファイルを変更する必要があるのみです。

Response Time Monitoring エージェント
Response Time Monitoring エージェントに以下の機能拡張が追加されています。

- 「**エージェント構成**」ページでアプリケーションのユーザー・トラッキングを構成するためのサポートを追加
- 「**エージェント構成**」ページでアプリケーションのセッション・トラッキングを構成するためのサポートを追加

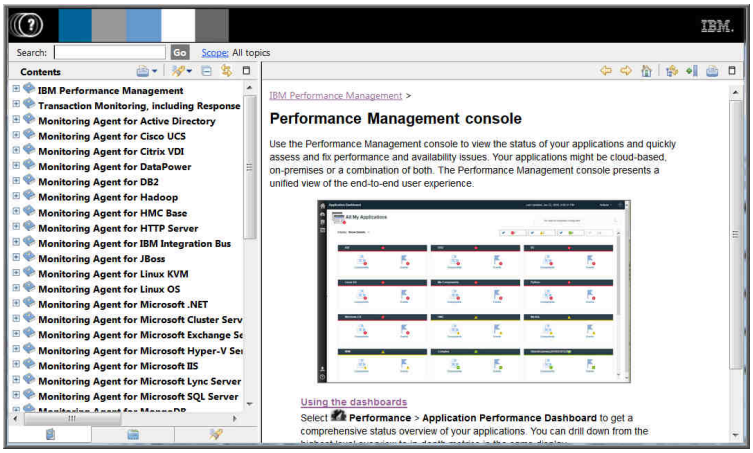
Cloud APM コンソールの機能拡張
次のコンソール機能拡張に加えて、エージェントのインストールおよび構成のインターフェースに 対してさまざまな機能強化が行われました。

- テクノロジー・プレビュー: アプリケーション・ダッシュボード・ページで、新しい「**カスタム・ビュー (Custom Views)**」タブを使用できます。管理対象リソースのメトリックを報告するためにさまざまなビューを作成し、平均やカウントなどの機能を適用できます。ページの保存後、異なるリソースのデータでのページの最新表示や、ページ・メトリックの PDF ファイルまたは CSV ファイルとしてのダウンロードを実行できます。詳しくは、1112 ページの『カスタム・ビュー』を参照してください。



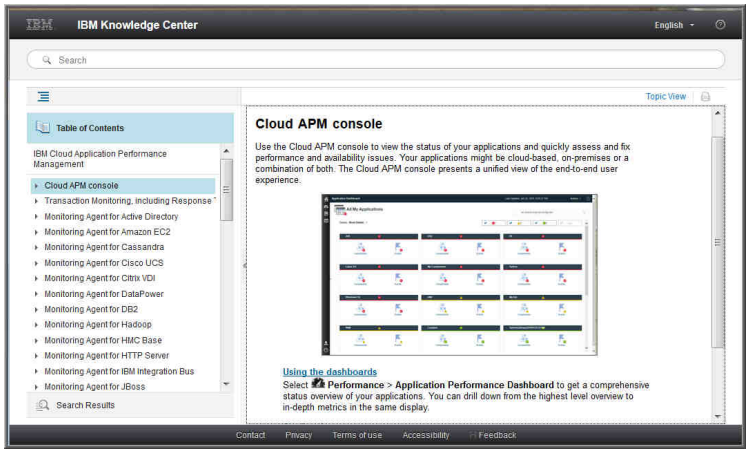
ご使用の Cloud APM サブスクリプションに「**カスタム・ビュー (Custom Views)**」が表示されず、この新機能を試す場合は、「**カスタム・ビュー (Custom Views)**」テクノロジー・プレビューを有効にするサービス・リクエストを、IBM サポートで作成してください。それらにデータを設定するカスタム・ダッシュボード・ページや履歴データは、システム保守中に保存されないことに注意してください。

- Cloud APM コンソールのヘルプ・システムを開くと、これが IBM Knowledge Center によってホストされていることがわかります。「**目次の非表示**」ツール、検索と印刷の機能、およびサポート情報とフィードバック・オプションへのリンクがあります。



前

後



Agent Builder の機能拡張

Agent Builder エージェント用に Cloud APM 要約ダッシュボードを作成するためのサポートが強化されました。単一行のデータ・セットを使用して、要約ダッシュボードのデータを指定する必要がありま

す。このようなデータ・セットは、ログ・ファイル全体から指定することも、フィルターに掛けて単一行にできる任意のデータのセットから指定することもできます。

新機能 (2016 年 9 月)

Performance Management on Cloud の 2016 年 9 月リリースで、新しいフィーチャー、機能、および対象が使用可能になりました。

使用可能な新規エージェント

Monitoring Agent for Amazon EC2

Amazon EC2 エージェントを使用して、Amazon Elastic Compute Cloud (EC2) インスタンス・リソースの正常性、可用性、およびパフォーマンスをモニターできます。以下のリソースをモニターできます。

- CPU 使用率
- Elastic Block Store (EBS) 使用率
- ネットワーク使用状況
- Amazon Web Services (AWS) の保守更新
- ディスク・パフォーマンス

このエージェントは Infrastructure Extension Pack に含まれており、IBM Monitoring、IBM Application Performance Management、および IBM Application Performance Management Advanced のいずれかのオファリングに使用可能です。

Monitoring Agent for SAP NetWeaver Java Stack

SAP NetWeaver Java Stack エージェントを使用して、SAP NetWeaver Java Stack クラスターとインスタンス・リソースの正常性、可用性、およびパフォーマンスをモニターできます。このエージェントを使用して、クラスター・リソース (ヒープ・ダンプ、JVM インスタンス、ユーザー・セッションの応答時間、トランザクション詳細、システム情報、ライセンス詳細など) をモニターできます。このエージェントを使用して、インスタンス・リソース (CPU 使用率、ディスク使用率、メモリー使用率、データベース・コレクション、ガーベッジ・コレクション、ヒープ・ダンプ、失敗したアプリケーション、Web コンテナー、セッション情報など) をモニターできます。このエージェントは Advanced 拡張パック に含まれており、IBM Application Performance Management および IBM Application Performance Management Advanced のいずれかのオファリングがある場合に使用可能です。

エージェントの機能拡張

Monitoring Agent for Citrix Virtual Desktop Infrastructure

Virtual Delivery Agent (VDA) マシンと Desktop Delivery Controller (DDC) マシンの Windows イベント・ログ・イベントを取得する機能が追加されました。

Monitoring Agent for Linux KVM

Linux カーネル・ベースの仮想マシンのデプロイメント環境をモニターするエージェントに対してダッシュボードを使用できます。ダッシュボードには、次のモニター機能が用意されています。

- 要約ダッシュボードには、Linux カーネル・ベースの仮想マシン環境またはアプリケーションの CPU 使用率とメモリー使用率に基づく、ホストの全体的な状況が表示されます。
- 「ホスト詳細」ダッシュボードには、選択したホストに関する詳細が表示されます。
- 「ホスト、クラスター、およびストレージ」ダッシュボードには、モニター対象の仮想マシンに関する詳細が表示されます。
- 「仮想マシン詳細」ダッシュボードには、「ホスト詳細」ページで選択した仮想マシンに関する詳細が表示されます。

Monitoring Agent for Linux OS

Docker V1.8.0 以降がサポート対象です。Linux OS エージェントが Docker モニター機能を提供できるようにするために、新しい属性グループとウィジェットが追加されました。

Monitoring Agent for Oracle Database

Oracle Database エージェントのダッシュボードには、「インスタンスの詳細」ページ上の以下の新機能が組み込まれています。

- 選択したインスタンスのロック競合に関する情報を表示する表。
- Oracle Real Application Clusters の GCS と GES に関する情報を表示する表。
- 選択したインスタンスに接続されている自動ストレージ管理 (ASM) ディスク・グループの詳細を表示する表。
- 表領域ごとに詳細情報を表示するビュー。このビューは、「**空き表領域 - 下位 5 件**」をクリックすると表示されます。
- 選択したインスタンスに接続されているフォアグラウンド・プロセスとバックグラウンド・プロセスの履歴の詳細を表示する表。この表内のエンティティーをクリックすると 、そのインスタンスのすべてのプロセスの詳細表を表示できます。
- 選択したインスタンスの (実行時間に基づく) 上位 5 件の最長 SQL 照会を表示する表。この表内をクリックすると、そのインスタンスの上位 50 件の最長 SQL 照会の詳細表を表示できます。

Monitoring Agent for Synthetic Playback

Synthetic Playback エージェントには、新しいセキュリティー機能が組み込まれています。シンセティック・スクリプトに格納されているパスワードが Synthetic Script Manager に表示されるのを防止できます。

Monitoring Agent for VMware VI

エージェント分離機能の追加により、エージェント・ノードとそのサブノードを同じビューで表示および選択できるようなりました。

「コンポーネントの選択」ウィンドウで VMware 仮想インフラストラクチャー・コンポーネントを選択すると、コンポーネント・エディターにエージェント・ノードとそのすべてのサブノードのツリー構造が表示されます。

- ツリーを展開してエージェント・ノードを選択すると、すべてのサブノードが自動的に選択されます。ツリーを展開して、モニターするサブノードを 個別に選択することもできます。
- ツリーが縮小表示されているときにエージェント・ノードを選択すると、すべてのサブノードが自動的に除外されます。

「コンポーネントの選択」ウィンドウで ESX サーバー・コンポーネントを選択すると、サブノードとともに、スタンドアロンの ESX サーバーもコンポーネント・エディターに表示されます。サブノードとともに、モニター用にスタンドアロンの ESX サーバーも選択できます。

アプリケーションを作成すると、「APM UI」ダッシュボードに、親としてのエージェント・インスタンスと、子としてのそのノードのツリー構造が表示されます。

Response Time Monitoring エージェント

エンド・ユーザー・トランザクション・ダッシュボードで特定の IP アドレスまたはアドレス範囲に適用されるロケーションを、特定の環境に合わせてカスタマイズできます。ロケーション値をカスタマイズするには、「**エージェント構成**」の「**地理位置情報**」タブを使用します。

Cloud APM コンソールの機能拡張

- エージェントのインストールおよび構成のインターフェースに 対してさまざまな改良が加えられました。
- 直近のサーバー再始動以降に更新されたエージェント・ダッシュボードのリストを確認するための「**ダッシュボードのログ**」オプションが「**アクション**」メニューに追加されました。詳しくは、1079 ページの『すべてのマイ・アプリケーション - Application Performance ダッシュボード』を参照してください。
- より見やすくするために、選択したアプリケーションの Application Performance ダッシュボード・ページが簡素化されました。 クリティカルおよび警告の重大度のイベントの数が「イベント」タブのタイトルに表示され、それにより「**イベント重大度要約**」棒グラフが置き換えられます。トポロジー・ビューが使用可能になっているアプリケーションの場合、「現在のコンポーネント状況」棒グラフに切り替えるためのトグル・ボタンが「**集約アプリケーション・トポロジー (Aggregate Application Topology)**」ビューに含まれます。詳しくは、1082 ページの『状況の概要』を参照してください。

- 以前のリリースでは、コンポーネント・インスタンスの場合でなければ Application Performance ダッシュボードの「**属性の詳細**」タブを使用できませんでした。「**属性の詳細**」タブが、Response Time Monitoring エージェントおよび Synthetic Playback エージェントのトランザクション・インスタンスのヒストリカル表を作成する目的にも使用できます。視覚障害があるユーザーに対して、ヒストリカル表を作成する機能で、折れ線グラフの代わりとなるものを提供します (スクリーン・リーダー・ソフトウェアなどの支援技術では折れ線グラフを認識できません)。詳しくは、1092 ページの『カスタムのグラフおよび表の表示と管理』を参照してください。

API

API を使用して Performance Management 環境の準備を自動化するためのスクリプトを作成できます。詳しくは、1073 ページの『API の解説』を参照してください。

新機能 (2016 年 4 月)

Performance Management on Cloud の 2016 年 4 月リリースで、新しいフィーチャー、機能、および対象が使用可能になりました。

IBM Marketplace

IBM Performance Management on Cloud オファリングが IBM Marketplace から入手可能です。無料トライアルまたはサブスクリプション・アカウントにサインアップしてください。詳しくは、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

使用可能な新規エージェント

Monitoring Agent for Citrix Virtual Desktop Infrastructure

Citrix VDI エージェントを使用して、Citrix XenDesktop または XenApp のリソース (サイト、マシン、アプリケーション、デスクトップ、セッション、ユーザーなど) の正常性、可用性、およびパフォーマンスをモニターできます。このエージェントは Infrastructure Extension Pack に含まれており、IBM Monitoring、IBM Application Performance Management、および IBM Application Performance Management Advanced のいずれかのオファリングに使用可能です。

Monitoring Agent for Skype for Business Server

Skype for Business Server エージェントを使用して、Microsoft Lync Server のリソース (データベース、メディエーション・サーバー、シンセティック・トランザクション、インスタント・メッセージ、CDR サービス書き込み操作、SIP ピアなど) の正常性、可用性、およびパフォーマンスをモニターできます。

Monitoring Agent for WebLogic

WebLogic エージェントを使用して、WebLogic Server のリソース (Java 仮想マシン (JVM)、Java Messaging Service (JMS)、Java Database Connectivity (JDBC) など) の正常性、可用性、およびパフォーマンスをモニターできます。

統合の機能拡張

エージェントの共存

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェントをインストールすることはできません。950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。

IBM Alert Notification

Alert Notification に、iOS デバイスと Android デバイス上で、Alert Notification の機能のサブセットを提供するモバイル・アプリケーションが含まれます。

IBM 統合スタック・モニター

IBM 統合スタックをモニターし、IBM MQ、IBM Integration Bus、および DataPower アプライアンスのミドルウェア製品とそれらの製品が公開するサービスのトランザクション・トラッキング情報を確認し、問題が発生した場合にトラブルシューティングできます。99 ページの『シナリオ: IBM 統合スタックのモニター』を参照してください。

エージェントの機能拡張

Monitoring Agent for Db2
デフォルト・ユーザー (Windows システムの場合) およびインスタンス所有者のユーザー (Linux システムおよび AIX システムの場合) に対して、Db2 エージェントの一部の属性のデータを表示するための特権を付与するためのコマンドが追加されました。

Monitoring Agent for Hadoop
Hadoop エージェントが Linux、Windows、および AIX の各オペレーティング・システムでサポートされます。

Monitoring Agent for HMC Base
仮想入出力およびハードウェア・イベントのためのモニター機能が備わっています。

Monitoring Agent for IBM Integration Bus

Linux システムおよび AIX システムで、エージェントの構成中に最新バージョンの IBM MQ (WebSphere MQ) のライブラリー・パスを自動検出できます。

Monitoring Agent for Microsoft Cluster Server
Microsoft Cluster Server エージェント はインストール後に自動的に構成されます。

Monitoring Agent for Microsoft Exchange Server
「エージェント構成」ウィンドウの「**Exchange サービス**」タブにいくつかのサービスが追加され、Exchange Server の状況を確認できます。

Monitoring Agent for Microsoft Hyper-V Server
エージェント構成パネルが削除されました。エージェント構成は不要です。

Monitoring Agent for SAP HANA Database
「**SAP HANA データベース詳細**」ダッシュボードに「キャッシュ情報詳細」グループ・ウィジェットが追加され、モニター対象データベースについて、使用済みメモリーの割合、使用可能メモリーの割合、およびキャッシュのヒット率に関する情報を提供します。

Monitoring Agent for Synthetic Playback
Synthetic Playback エージェントに含まれる機能は次のとおりです。

- Synthetic Playback エージェントをインストールおよび構成して、外部向けパブリック・アプリケーションに加えて、内部向けプライベート・アプリケーションのパフォーマンスおよび 可用性を Application Performance ダッシュボードでモニターできます。
- Synthetic Script Manager を使用して、アプリケーションの可用性およびパフォーマンスをテストするための単純なスクリプトを生成します。
- 各ロケーションでのシンセティック・トランザクションの同時再生またはずらし再生を構成します。
- Synthetic Script Manager で月次再生使用量をモニターします。
- HTTP メトリックおよび可用率を Synthetic Playback エージェント・レポートに表示します。
- 2 つの新しいレポート (トランザクションのトレンドとサブトランザクションのトレンド) が表示されます。
- シンセティック・トランザクションをリソース・グループに編成し、そのリソース・グループ内のすべてのトランザクションにしきい値を適用します。
- 関連するシンセティック・トランザクションを含むアプリケーションを作成しなくても、Application Performance ダッシュボードの「**マイ・トランザクション**」ウィンドウにシンセティック・トランザクション・データが表示されます。
- シンセティック・スクリプトを Synthetic Script Manager からダウンロードします。

Monitoring Agent for VMware VI
VMware VI エージェント・ダッシュボードが拡張され、以下の新機能が組み込まれています。

- 「**コンポーネント**」ページには、「クリティカル」状況または「警告」状態でトリガーされたアラームの数も表示されます。

- 「**クラスターの要約**」ページの新しい表には、予防アラームおよび障害アラームに関する情報が表示されます。この表で、トリガーされたエンティティーをクリックして 、その詳細ページを表示できます。
- 「**クラスターの詳細**」ページの新しい表には、選択したクラスターに属する ESX サーバーの詳細が表示されます。ESX サーバーをクリックして、その詳細ページを表示できます。
- 「**クラスターの詳細**」ページの「データ・ストア」表には、データ・ストアのオーバー・コミット・メトリックが表示されます。
- 「仮想マシン詳細」ページの「仮想マシン」表には、追加のパフォーマンス・メトリック (メモリー・サイズ、NIC、ディスクなど) が表示されます。これらのメトリックをクリックして、その詳細ページを表示できます。
- 選択した仮想マシンの重要なパフォーマンス・メトリック (メモリー、ディスク、およびネットワークなど) を表示するために、新しいウィジェットとページが追加されました。
- 「**ESX Server 詳細 (ESX Server Detail)**」ページの新しい表には、サーバー・ネットワークのネットワーク・パフォーマンスが表示されます。
- 「**仮想マシン詳細**」ページと「**ESX Server 詳細 (ESX Server Detail)**」ページの「データ・ストア」表には、データ・ストアの待ち時間メトリックが表示されます。
- 「**データ・ストア詳細**」ページの新しい表には、当該データ・ストアに関連付けられた仮想マシンに関する情報が表示されます。この表で仮想マシンをクリックして、「**仮想マシン詳細**」ページを表示できます。
- 「**仮想マシン詳細**」ページの「メモリー (%) (ヒストリー)」グラフのタイトルを「ゲスト・メモリー (%) (ヒストリー)」に変更しました。

Monitoring Agent for WebSphere Applications

処理中要求の要約ダッシュボードに、現在低速またはハング状態になっている要求インスタンスを識別する機能が備わっています。このダッシュボードの処理中要求ウィジェットで要求を選択して「**スレッドのキャンセル**」をクリックすることによって、処理中要求のソフト・キャンセル操作を実行できます。

事前定義のすべてのイベントしきい値が調整され、ユーザー・エクスペリエンスが向上しています。機能拡張および更新には、アラートをトリガーする条件、サンプリング間隔、およびしきい値の重大度も含まれます。

Monitoring Agent for WebSphere Applications のユーザー・インターフェースは 、身体障がいのあるユーザーに対応しています。

お客様からのフィードバックおよびテクニカル ・レビューに基づいて構成プロセスが調整され、ユーザー・エクスペリエンスが向上しました。

Monitoring Agent for WebSphere MQ

事前定義のイベントしきい値に対していくつかの変更を加えました。

- すべての事前定義しきい値に接頭部 MQ_ が付いています (旧バージョンでは MQSeries_ でした)。
- チャネル・イニシエーター・サーバーの状況およびキュー・マネージャーの状況についてクリティカル・アラートをトリガーするために、2 つのしきい値 MQ_Channel_Initiator_Crit と MQ_Queue_Manager_Crit が追加されました。
- MQ_Queue_Depth_High イベントのトリガー条件が、静的な 80% からキューの項目数上限に変更されました。

「**読み取られていないキュー - 上位 5 件**」ウィジェットの名前が「**読み取られていない使用中のキュー - 上位 5 件**」に変更されました。このウィジェットは、メッセージが存在し、キューにメッセージを書き込むために 1 つ以上のアプリケーションによって接続されているが、いずれのアプリケーションでも読み取られていない、上位 5 件のキューのリストを示します。

エージェントの構成中に、最新バージョンの IBM MQ (WebSphere MQ) のライブラリー・パスを自動検出できます。サイレント応答ファイルの **WMQLIBPATH** パラメーターを空のままにしても、対話式にエージェントを構成する際にデフォルト値をそのまま使用してもかまいません。

OS エージェント

OS エージェントに、アプリケーション・ログ・ファイルをモニターするための新機能を組み込んでいます。この機能には、正規表現に基づいたログ・ファイル・モニターを構成する機能が含まれます。

互換性を確保するために、OS エージェントは以下の情報およびフォーマットを使用します。

- IBM Tivoli Monitoring 6.x Log File Agent で使用された構成情報およびフォーマット・ファイル
- Tivoli Event Console ログ・ファイル・アダプターで 使用された構成情報および書式制御ストリング

これらの書式制御ストリングを使用することにより、エージェントはフォーマット・ファイル内のパターンに従ってログ・データをフィルタリングし、関係のあるデータのみをイベント・コンシューマーに送信できます。OS エージェントはデータを、Performance Management サーバーにも、Event Integration Facility (EIF) を介して OMNIbus EIF プローブなどの任意の EIF 受信側にも送信します。

Response Time Monitoring エージェント

エンド・ユーザー・トランザクション・ダッシュボードにユーザーおよびデバイスの情報が含まれます。以前は、これらの情報はユーザー・グループの認証ユーザー・ダッシュボードおよびモバイル・デバイス・ユーザー・ダッシュボードに表示されていました。ユーザー、セッション、およびデバイスの情報は、ユーザーの IP アドレスに基づいた場所 (国、都道府県、および都市) でソートされます。この新規および更新版のダッシュボードを使用して、ユーザー数、および特定のユーザー・グループに対して問題が切り分けられているかどうかを把握できます。

「エンド・ユーザー・トランザクション」ダッシュボードで、特定の IP アドレスまたはアドレス範囲に適用されるロケーションを、特定の環境に合わせてカスタマイズします。ロケーション値をカスタマイズするには、「**エージェント構成**」の「**地理位置情報**」タブを使用します。

トランザクション・トラッキング

「トランザクション要約」ページには、サービスの依存関係トポロジーが表示されます。このトポロジーは、選択したリソース・ノード (IBM Integration Bus など) およびそのノードが依存するサービスを表示します。「トランザクションの詳細」ページには、トランザクションの依存関係トポロジーが表示されます。このトポロジーは、各コンポーネント・インスタンスのトランザクション・ノードおよび各依存サービスの非インスツルメント・ノードをトランザクション・レベルで表示します (例えば、IBM Integration Bus およびそのサービス・トランザクション)。また、「トランザクションの詳細」ページは、選択したアプリケーションのユーザーの中で最も遅い応答時間に遭遇しているユーザー、およびトランザクションのボリュームが最も高いホストを強調表示します。

エージェントの汎用的な機能拡張

エージェントのインストールおよび構成について、以下の汎用的な機能拡張を行いました。

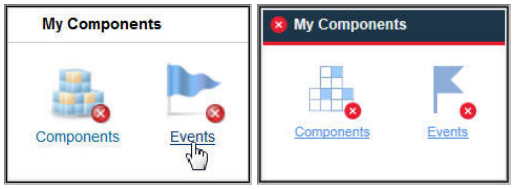
- インストールを開始する前に、エージェント・インストール・スクリプトで権限検査を実行します。十分な権限を持っていない場合は、メッセージが表示されます。
- エージェント状況コマンドは、エージェントと Performance Management コンソールの間の状況を検査します。
- Windows システムでサポートされるエージェントには、エージェント構成の実行や接続状況の検査に使用できる GUI ユーティリティーが用意されています。
- 新しいコマンドを使用することで、エージェントをアンインストールせずにエージェント ・インスタンスを削除できます。

Performance Management サーバーの機能拡張

Performance Management ユーザー認証は、IBMid OpenID Connect プロバイダーを介して管理されます。

Performance Management コンソールの機能拡張

- Performance Management コンソールの外観が、IBM Bluemix ユーザー・インターフェースと 整合するように更新されました。例えば、以下のように「**すべてのマイ・アプリケーション**」ダッシュボードの要約ボックスで、V8.1.2 と現行バージョンの間の違いがあります。



- 「拡張構成」ページに新しいオプションが追加され、上級者はすべてのシステム・グループに対してすべての事前定義しきい値を容易に有効または無効にできるようになりました。978 ページの『背景情報』を参照してください。
- Application Performance ダッシュボードの自動最新表示の頻度を制御するための新しいオプションが「拡張構成」ページに追加されました。詳しくは、1074 ページの『UI 統合』を参照してください。
- エージェントのインストールおよび構成のインターフェースに 対してさまざまな改良が加えられました。
- Performance Management コンソールのアクセシビリティーが向上しました。ユーザー・インターフェースのアクセシビリティー 機能については、1507 ページの『アクセシビリティー機能』を参照してください。

API

API を使用して Performance Management 環境の準備を自動化するためのスクリプトを作成できます。詳しくは、1073 ページの『API の解説』を参照してください。

Agent Builder の機能拡張

Agent Builder には、拡張データ・セット・フィルター機能が含まれます。フィルター機能を使用すると、可用性データ・セットなどの複数行データ・セットに基づいて単一行を返すデータ・セットを作成できます。この機能は、要約ダッシュボードに情報を表示するために使用します。

第 2 章 PDF 資料

この IBM Knowledge Center コレクション内のトピックおよびエージェントのリファレンスについては、PDF 資料を入手できます。

IBM Knowledge Center (PDF 形式)

このユーザーズ・ガイドに加えて、「IBM Agent Builder ユーザーズ・ガイド」をダウンロードできます。

エージェントのリファレンス (PDF 版)

エージェントのリファレンスをダウンロードするには、 Application Performance Management Developer Center の「Agent metrics/Reference PDFs」を参照してください。リファレンスには、ダッシュボード、イベントしきい値、およびデータ・セットに関する情報が記載されています。データ・セットには、エージェントによって報告され、重要パフォーマンス指標 (KPI) を構成するメトリックである属性が含まれています。 PDF ファイルのタイトル・ページでエージェント・バージョンを確認できます。

© Copyright IBM Corp. 2014, 2019

第 3 章 製品の概要

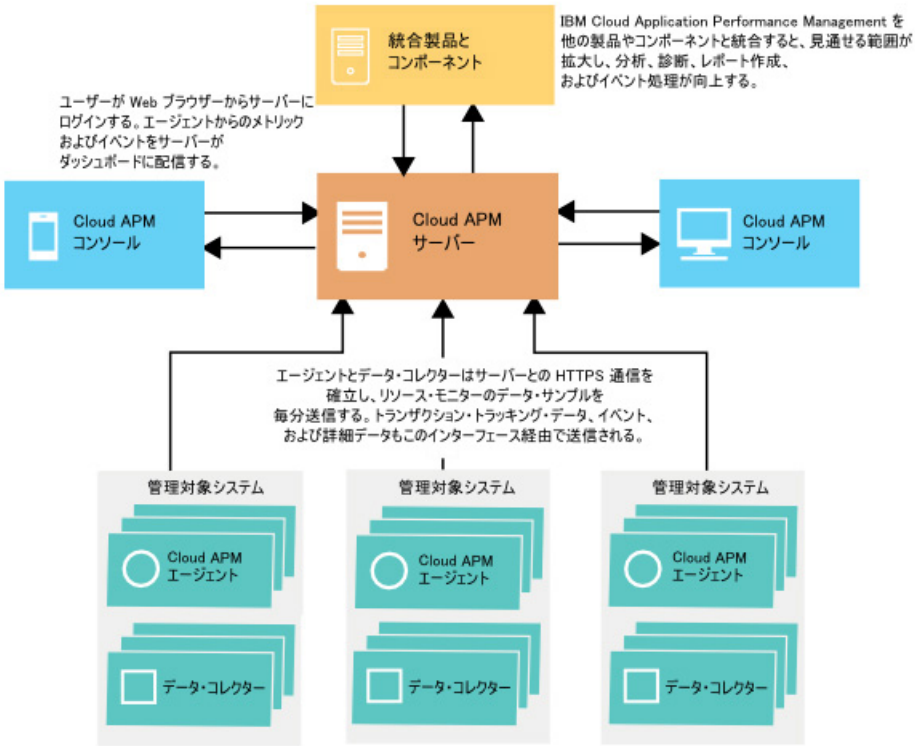
IBM Cloud Application Performance Management (Cloud APM) は、オンプレミス (専用)、公開クラウド、または混成環境にデプロイされているアプリケーションのパフォーマンスと 可用性の管理を支援する包括的なソリューションです。このソリューションはアプリケーションの 可視性、制御、および自動化の機能を提供し、最適なパフォーマンスとリソースの 効率的な使用を実現します。

このソリューションを使用して、データ・センター、クラウド・インフラストラクチャー、およびワークロードを認識インテリジェンスによって管理します。パフォーマンス上の問題を識別してから、ビジネスへの影響が生じる前に問題の発生箇所を分離して問題を診断するまで Cloud APM が支援するため、ハイブリッド・アプリケーション環境での停止およびスローダウンの発生を常に低下および回避できます。

Cloud APM エージェントおよびデータ・コレクターによって収集されたデータを処理するには、主要機能を使用します (主要機能はオファリングによって異なります)。他の製品やコンポーネントと統合すると、さらに多くの機能を使用できます。

アーキテクチャーの概要

IBM Cloud Application Performance Management は、エージェント およびデータ・コレクター を使用してモニター対象ホスト上のデータを収集します。エージェントおよびデータ・コレクターはデータを Cloud APM サーバーに渡し、このサーバーは渡されたデータを照合して Cloud APM コンソールに送ります。Cloud APM サーバーは IBM クラウドでホストされています。



データ収集

エージェントおよびデータ・コレクターはシステム、サブシステム、またはアプリケーションをモニターし、データを収集します。エージェントまたはデータ・コレクターは、単一のリソース (例えば、1 つのシステムまたはアプリケーション) と対話し、ほとんどの場合、システムまたはアプリケーションが 実行されているのと同じコンピューターまたは仮想マシンにあります。例えば、Linux OS エージェントは Linux ホスト上のオペレーティング・システムのパフォーマンス 指標を収集し、WebSphere Applications エージェ

© Copyright IBM Corp. 2014, 2019

ントは WebSphere Application Server のパフォーマンス指標をモニターします。また、異なるリソース間のトランザクションをトラッキングするエージェントもあります。

重要パフォーマンス指標 (KPI) にしきい値を設定することができます。指標がしきい値を超えたり、しきい値に達しなかったりすると、エージェントまたはデータ・コレクターはアラートを生成します。その後、そのアラートはサーバーによって処理されます。Netcool/OMNIbus Probe for Tivoli EIF や SMTP サーバーなどのターゲットにイベントを転送するように構成したり、Alert Notification を使用してイベントの E メール通知を構成したりすることもできます。

各エージェントおよびデータ・コレクターは、Cloud APM サーバーと通信するように事前構成されています。

サーバーとエージェントまたはデータ・コレクターとの間の通信

すべてのモニター対象ホスト上の各エージェントおよびデータ・コレクターは、IBM クラウドにある Cloud APM サーバーとの HTTPS 通信を確立します。エージェントまたはデータ・コレクターは、接続のクライアント・サイドになります。

エージェントおよびデータ・コレクターは、データをサーバーに送信するのにインターネット接続を必要とします。エージェントまたはデータ・コレクターがインターネットを介して直接データを送信できない場合は、フォワード・プロキシーが必要になる可能性があります。詳しくは、163 ページの『ネットワーク接続』を参照してください。

サーバーで保管されるデータ

エージェントおよびデータ・コレクターは、データのタイプに応じて、1 分から 8 分の間隔で、データを Cloud APM サーバーにプッシュします。サーバーは、デフォルトで、エージェントおよびデータ・コレクターによって送信されたすべての値を 8 日間保管します。要約されたトランザクション・データは、これより長く保管されます。

保存されたモニター・データは「ヒストリカル・データ」と呼ばれます。サーバーはヒストリカル・データを使用して、環境内の傾向分析に使用できる表やグラフを表示します。

特定のエージェントではヒストリカル・レポートも使用できます。詳しくは、1124 ページの『レポート』を参照してください。

スケーラビリティー

Cloud APM からは、最大 10,000 の管理対象システムをモニターできます。 管理対象システムは、エンタープライズ内の、エージェントが監視している単一のオペレーティング・システム、サブシステム、またはアプリケーションです。

Cloud APM は、1 秒当たり 150 から 400 の管理対象ユーザー・トランザクションをサポートします。

Integration

IBM Cloud Application Performance Management は、他の製品およびコンポーネントが Cloud APM サーバーとの通信用に構成されている場合に、その製品およびコンポーネントと統合します。

統合できる製品には、IBM Control Desk、Netcool/OMNIbus、Tivoli Monitoring、OMEGAMON、Operations Analytics - Log Analysis、Operations Analytics - Predictive Insights、IBM Alert Notification、および IBM Cloud があります。

Agent Builder は、カスタム・エージェントを作成するために使用可能な 1 つのコンポーネントです。

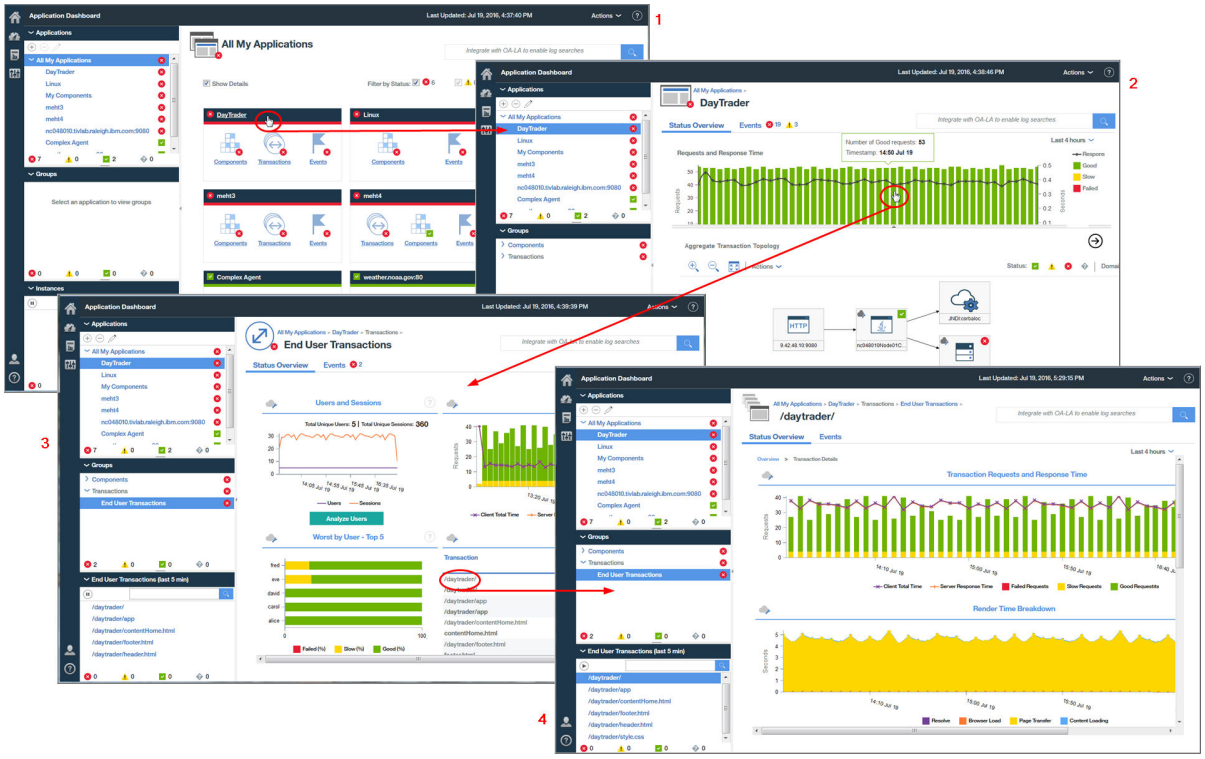
ユーザー・インターフェース

Cloud APM コンソールは、Cloud APM のユーザー・インターフェースです。この統合ユーザー・インターフェースは、すべてのハイブリッド・アプリケーションに対する単一のビューを提供します。アプリケーションの状況を表示し、パフォーマンスおよび可用性の問題を迅速に評価および修正するには、コンソールを使用します。

コンソール内のダッシュボードにより問題の識別が容易になるため、アプリケーションのパフォーマンスに影響を及ぼすボトルネックを切り分けることができます。わかりやすいダッシュボード・ナビゲーションを使用して、アプリケーション状況のビューからコード・レベルの詳細に移動します。問題の発生時点にソース・コードの問題があることを確認できます。統合検索分析を使用して問題を検索および診断できます。

コンソールでの Application Performance ダッシュボードのナビゲーターは階層型であり、使用しているアプリケーションの状況の概要を確認できるほか、それらのコンポーネントの正常性やユーザー・エクスペリエンスの品質についても確認することができます。 ダッシュボード・ビューのナビゲーター項目やリンクをクリックすると、モニター対象リソースの詳細を確認できます。例えば、アプリケーションの応答時間が長いとします。この問題は、ダッシュボードで明らかになります。ダッシュボードから順にリンクをクリックして問題をその発生源まで追跡し、その原因 (制御不能なプロセスによってシステムの CPU 使用量が高くなっていること) を見つけることができます。

Cloud APM コンソールでのダッシュボードの使用方法について詳しくは、1079 ページの『第 10 章 ダッシュボードの使用』を参照してください。



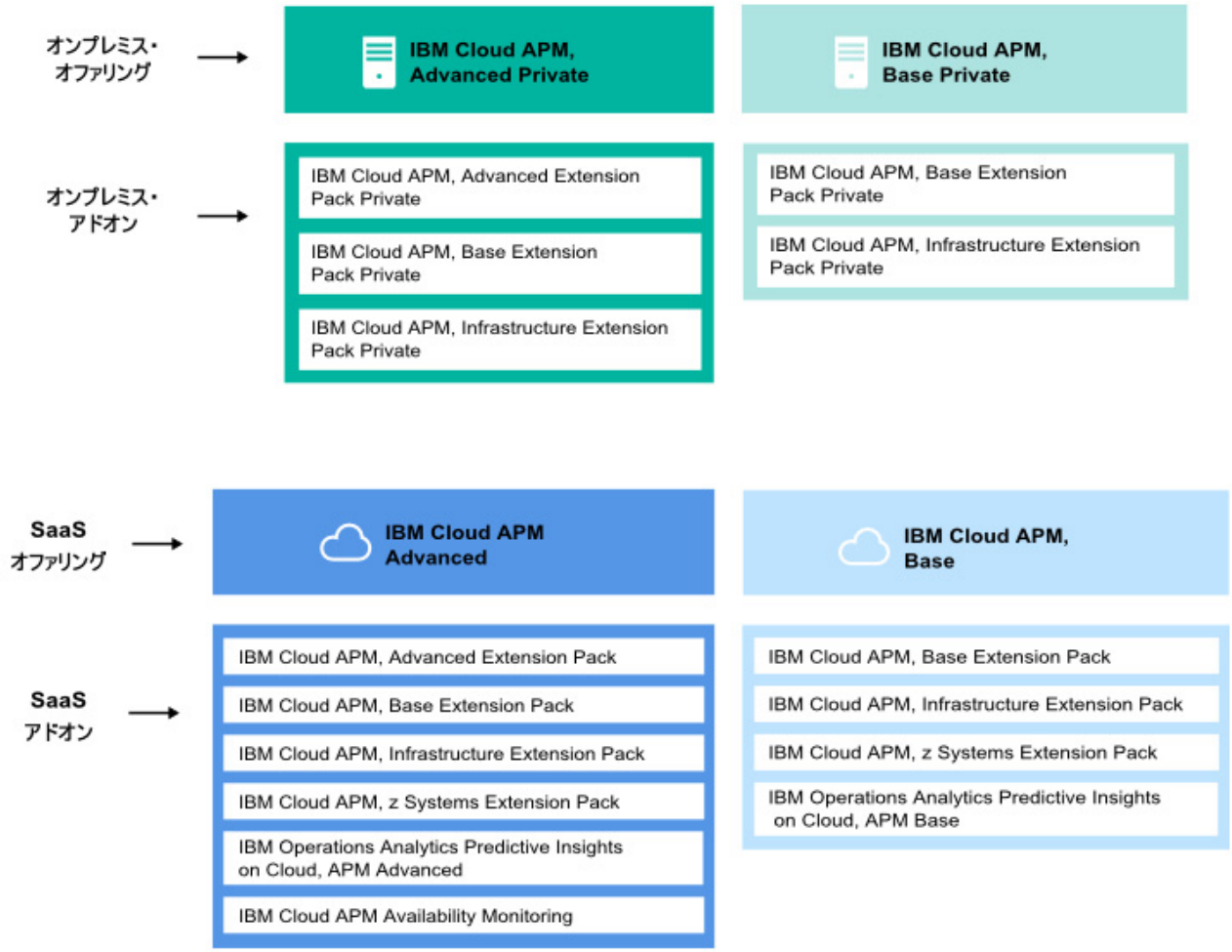
オファリングとアドオン

IBM Cloud Application Performance Management には 2 つのオファリングと複数のアドオンが含まれています。 これらのオファリングおよびアドオンには、エージェントとデータ・コレクターが含まれています。オファリングごとに、特定のアドオンを組み合わせて使用できます。

オファリングまたはアドオンに含まれるエージェント、およびエージェントとデータ・コレクターの機能を確認するには、54 ページの『機能』を参照してください。

各オファリングに対して、IBM Marketplace からアドオンを入手できます。 IBM Cloud Application Performance Management, Advanced は最も包括的なオファリングであり、すべてのエージェント、データ・コレクター、およびダッシュボード・ページが含まれています。IBM Cloud Application Performance Management, Base は Cloud APM, Advanced のサブセットです。いつでも、Cloud APM, Base を Cloud APM, Advanced に置き換えることができます。この置き換え後の最終的なインストール済みオファリングは、Cloud APM, Advanced です。各オファリングで使用できるアドオンを、図に示します。

すべてのオファリングでアドオンは同じです。ただし、Cloud APM, Advanced オファリング専用アドオンである Availability Monitoring は除きます。



オファリング

IBM Cloud Application Performance Management, Advanced

このオファリングは、すべてのアプリケーション・コンポーネントのエンド・ユーザー・エクスペリエンス、トランザクション・トラッキング、およびリソース・モニターに対応します。アプリケーション、およびアプリケーション・サーバーの正常性をコード・レベルで可視化できます。診断ダッシュボードを使用して、アプリケーション・コードのパフォーマンス・ボトルネックを検出したり、実動の重要なアプリケーションを管理したりできます。

このオファリングには、IBM Cloud Application Performance Management, Base が含まれていて、企業にインストールされているアプリケーション、トランザクション、およびその他のリソースをモニターするために使用するエージェントとデータ・コレクターが含まれています。このオファリングに含まれるエージェントおよびデータ・コレクターのリストについては、54 ページの『機能』を参照してください。

DevOps は、このオファリングにより、アプリケーションとインフラストラクチャーをすべて 見通して制御する完全なソリューションを取得します。基幹業務の所有者は、実動環境でのクリティカル・アプリケーションとエンド・ユーザー・エクスペリエンスを管理できます。アプリケーション開発者は、トランザクションの詳細を表示し、アプリケーションの問題を診断できます。

IBM Cloud Application Performance Management, Base

このオファリングは、インフラストラクチャー、アプリケーション・コンポーネント、およびクラウド・ワークロードのリソース・モニターに対応します。リソース・モニターは、低速なトランザクション、容量の問題、および障害を識別して対処するのに役立ちます。このオファリングには、企業にインストールされているアプリケーションおよびその 他のリソースをモニターする際に使用するエージェントおよびデータ・コレクターが含まれています。このオファリングに含まれるエ

ージェントおよびデータ・コレクターのリストについては、54 ページの『機能』を参照してください。

IT オペレーターは、このオファリングにより、低速のトランザクション、キャパシティーの問題、および障害に対処できます。

アドオン

Advanced Extension Pack

この拡張パックには Monitoring Agent for SAP HANA Database、SAP NetWeaver Java Stack エージェント、および Monitoring Agent for RabbitMQ が含まれています。

SAP HANA データベース・エージェントを使用して、SAP HANA データベースをモニターします。SAP NetWeaver Java Stack エージェントを使用して、SAP NetWeaver Java スタックをモニターします。 RabbitMQ エージェントを使用して、RabbitMQ メッセージングをモニターします。 この拡張パックは、IBM Cloud Application Performance Management, Advanced オファリングがある場合に使用可能です。

Base Extension Pack

この拡張パックには、以下のエージェントが含まれています。

- Monitoring Agent for Cassandra
- Monitoring Agent for InfoSphere DataStage
- Monitoring Agent for Hadoop
- Monitoring Agent for Microsoft Office 365
- Monitoring Agent for Sterling Connect Direct
- Monitoring Agent for Sterling File Gateway

これらのエージェントを使用して、Cassandra データベース、Hadoop クラスター、DataStage サーバーのリソース、Microsoft Office 365 アプリケーション、Connect Direct サーバー、および Sterling File Gateway アプリケーションをモニターします。この拡張パックは、いずれかの Cloud APM オファリングを所有している場合に入手できます。

Infrastructure Extension Pack

この拡張パックには、以下のエージェントが含まれています。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS Elastic Load Balancer
- Monitoring Agent for Azure Compute
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for IBM Cloud

Amazon EC2 エージェントを使用して、Amazon EC2 インスタンスをモニターします。Amazon ELB エージェント を使用して、AWS Elastic Load Balancer をモニターします。 Azure Compute エージェント を使用して、Azure Compute 仮想マシンをモニターします。Citrix VDI エージェントを使用して、Citrix 仮想デスクトップ・インフラストラクチャーをモニターします。

この拡張パックは、いずれかの Cloud APM オファリングを所有している場合に入手できます。

z Systems Extension Pack

z Systems Extension Pack を使用して、Cloud APM コンソールで OMEGAMON アプリケーション・コンポーネントのモニター・データおよびイベントを表示できます。この拡張パックは、いずれかの Cloud APM オファリングを所有している場合に入手できます。

Operations Analytics - Predictive Insights

このアドオンは、Cloud APM によって収集されたメトリック・データを分析し、異常が検出された場合にアラームを生成するためのものです。このアドオンは、いずれかの Cloud APM オファリングを所有している場合に入手できます。

Availability Monitoring

このアドオンは、地理的に分散した複数のポイント・オブ・プレゼンス (Point of Presence: PoP) から Web アプリケーションの可用性とパフォーマンスをモニターするためのものです。このアド

オンはスタンドアロン・オファリングとして機能しませんが、IBM Cloud Application Performance Management, Advanced オファリングがある場合に使用可能です。

各オファリングの機能の概要については、50 ページの『オファリングの詳細』を参照してください。

それぞれのエージェントおよびデータ・コレクターの説明と、それぞれに固有の情報へのリンクについては、58 ページの『説明』を参照してください。

オファリングの詳細

すべてのオファリングで使用可能な機能もあれば、特定のオファリングのみで使用可能な機能もあります。

各オファリングで使用できる主要な機能の一覧を 50 ページの表 1 に示します。

表 *1.* 各オファリングの機能

機能	Cloud APM, Advanced (DevOps、開発者、および基幹業務用)	Cloud APM, Base (操作用)
アプリケーション・リソース・モニター: 言語、ミドルウェア (適応範囲はオファリングによって異なります)。	✓	✓
オペレーティング・システム・モニター: Linux、UNIX、Windows の各システム	✓	✓
ログ・ファイル・モニター: OS エージェントを使用して、アプリケーション・ログ・ファイルをモニターします。	✓	✓
ダッシュボード: • Tivoli Monitoring と Cloud APM の KPI を同じダッシュボードに表示します。 • 履歴メトリック • カスタマイズ可能なダッシュボード	✓	✓
API: API を使用して環境を管理します。	✓	✓
役割ベースのアクセス制御: IBM Cloud Application Performance Management ユーザーのアクセス権限と特権を管理します。	✓	✓
ヒストリカル・レポート: アプリケーションのパフォーマンスと 応答時間に関するレポートをトランザクション、デバイス、ブラウザーなどごとに明細化して生成します (適応範囲はオファリングによって異なります)。	✓	✓
IBM Agent Builder: カスタム・エージェントを作成して、任意のプラットフォームまたはテクノロジーをモニターします。	✓	✓
データベース・リソース・モニター: (適応範囲はオファリングによって異なります)	✓	✓
インフラストラクチャー・リソース・モニター: ハイパーバイザー、ストレージ、およびネットワーク (適応範囲はオファリングによって異なります)。	✓	✓

表 1. 各オファリングの機能 (続き)

機能	Cloud APM, Advanced (DevOps、開発者、および基幹業務用)	Cloud APM, Base (操作用)
商業アプリケーションのリソース・モニター: ビジネス・アプリケーションおよびコラボレーション・アプリケーション (適応範囲はオファリングによって異なります)。	✓	✓
Response Time Monitoring: アプリケーションのパフォーマンスがユーザーにどのように影響しているかを表示します。	✓	✓
探索分析との統合: 問題の切り分け、診断、解決を素早く行うための情報を収集します。	✓	✓
Operations Analytics - Predictive Insights (アドオン): アプリケーションのパフォーマンス 異常を、ユーザーに影響する前に判別します。	✓	✓
実際のエンド・ユーザー・エクスペリエンスのモニター: インフラストラクチャーからデバイスに至るまでのユーザー・エクスペリエンスを表示します。	✓	–
トランザクション・トラッキング: アプリケーション環境全体にわたってエンドツーエンド・トランザクションを追跡します。 • アプリケーション・トポロジー: アプリケーション環境で、すべてのコンポーネントがどのように接続されているかを表示します。 • トランザクション・インスタンス・トポロジー: トランザクションのインスタンスごとに、環境内でたどったパスを表示します。	✓	–
詳細診断: • 要約ダッシュボードからドリルダウンして、特定のエージェントのコード・レベル、スタック・トレース、および SQL 照会の詳細を表示します。 • 進行中にハングしたトランザクションまたは処理速度の遅いトランザクションを検出、診断、および強制終了します。	✓	–
しきい値: アクティブにモニターされる 定義に基づいて、特定のアプリケーションの動作および状態を検出します。	✓	✓
リソース・グループ: モニター対象エンタープライズの 管理対象システムを目的に応じて分類します。	✓	✓

以下のような追加の機能は、他の製品やコンポーネントとの統合によりすべてのオファリングで使用できます。80 ページの『統合』を参照してください (さらに詳しい説明は、949 ページの『第 8 章 他の製品およびコンポーネントとの統合』を参照してください)。

- Tivoli Monitoring エージェントと OMEGAMON エージェント: ハイブリッド・ゲートウェイを使用して、モニター・データとイベントを取得し、この情報が Cloud APM コンソールに表示されるようにします。

- エージェントの共存: Tivoli Monitoring エージェントがインストールされているのと同じコンピューターに Cloud APM エージェントをインストールします。
- Netcool/OMNIbus と他の EIF 受信側: IBM Tivoli Netcool/OMNIbus にイベントを転送します。
- Alert Notification: アプリケーション・パフォーマンスがしきい値を超えたときに通知を受信します。
- IBM Control Desk: Control Desk で自動的にチケットをオープンします。
- IBM Cloud: IBM Cloud アプリケーションをモニターします。

エージェントおよびデータ・コレクター

IBM Cloud Application Performance Management エージェントおよびデータ・コレクターは、さまざまなオファリングとアドオンの両方で使用できます。

環境の多数のリソースをエージェントによってモニターできます。データ・コレクターは、IBM Cloud 上およびオンプレミスの一部のリソースをモニターできます。J2SE データ・コレクターと Python データ・コレクターを除き、すべてのデータ・コレクターに対応するエージェントがあります。エージェントおよびデータ・コレクターのリストとその説明については、58 ページの『説明』を参照してください。エージェントまたはデータ・コレクターが各オファリングで提供できる機能を確認するには、54 ページの『機能』を参照してください。各エージェントおよびデータ・コレクターの変更履歴については、52 ページの『変更履歴』を参照してください。

ご使用の環境と要件に応じて、これらのエージェントまたはデータ・コレクターをインストールできます。データ・コレクターは、データを Cloud APM サーバーに直接送信します。エージェントが構成されている場合、データ・コレクターはデータをエージェントに送信して、エージェントがそのデータをサーバーに転送します。データ・コレクターはアプリケーションのプロセス・スペース内で作動します。一方、エージェントはアプリケーションのプロセス・スペースの外部の別のプロセスとして実行されます。

以下の状況で、データ・コレクターをインストールします。

- よりシンプルなインストール・プロセスが必要である。
- コンテナーを使用する。

以下の状況で、エージェントをインストールします。

- より優れたスケーラビリティーが必要である。
- エンドポイントからサーバーへのソケットを制限する必要がある。
- しきい値エディターでしきい値を追加するときに、モニターする環境の属性のみが含まれた明確なリストが必要である。データ・コレクターを使用する場合は、いくつかのデータ・コレクターの属性から選択する必要があります。
- UI で、診断、トランザクション・トラッキング、またはメソッド・トレースなどの一部のデータ収集機能をオンまたはオフにする必要がある。
- インフライト要求および現在時刻のヒープ・ダンプなど、オンデマンドの診断データを表示する必要がある。

変更履歴

各エージェントやデータ・コレクターのバージョンと変更履歴に関する情報を確認します。

以下の表に、エージェント名とデータ・コレクター名、および変更履歴の技術情報へのリンクを記載しています。 リンクをクリックすると、変更履歴の詳細が表示されます。

表 2. エージェントおよびデータ・コレクターの変更履歴

エージェントおよびデータ・コレクター	**リンク**
Amazon EC2 エージェント	変更履歴
Amazon ELB エージェント	変更履歴
Azure Compute エージェント	変更履歴

表 2. エージェントおよびデータ・コレクターの変更履歴 (続き)	
エージェントおよびデータ・コレクター	**リンク**
Cassandra エージェント	変更履歴
Cisco UCS エージェント	変更履歴
Citrix VDI エージェント	変更履歴
DataPower エージェント	変更履歴
DataStage エージェント	変更履歴
Db2 エージェント	変更履歴
Hadoop エージェント	変更履歴
HMC Base エージェント	変更履歴
HTTP Server エージェント	変更履歴
IBM Cloud エージェント	変更履歴
IBM Integration Bus エージェント	変更履歴
Internet Service Monitoring	変更履歴
J2SE データ・コレクター	変更履歴
JBoss エージェント	変更履歴
Liberty データ・コレクター	変更履歴
Linux KVM エージェント	変更履歴
Linux OS エージェント	変更履歴
MariaDB エージェント	変更履歴
Microsoft Active Directory エージェント	変更履歴
Microsoft Cluster Server エージェント	変更履歴
Microsoft Exchange Server エージェント	変更履歴
Microsoft Hyper-V Server エージェント	変更履歴
Microsoft IIS エージェント	変更履歴
Microsoft .NET エージェント	変更履歴
Microsoft Office 365 エージェント	変更履歴
Microsoft SharePoint Server エージェント	変更履歴
Microsoft SQL Server エージェント	変更履歴
MongoDB エージェント	変更履歴
MQ Appliance エージェント	変更履歴
MySQL エージェント	変更履歴
NetApp Storage エージェント	変更履歴
Node.js エージェント	変更履歴
Node.js データ・コレクター	変更履歴
OpenStack エージェント	変更履歴

表 2. エージェントおよびデータ・コレクターの変更履歴 (続き)	
エージェントおよびデータ・コレクター	**リンク**
Oracle Database エージェント	変更履歴
PHP エージェント	変更履歴
PostgreSQL エージェント	変更履歴
Python データ・コレクター	変更履歴
RabbitMQ エージェント	変更履歴
Response Time Monitoring エージェント	変更履歴
Ruby エージェント	変更履歴
Ruby データ・コレクター	変更履歴
SAP エージェント	変更履歴
SAP HANA データベース・エージェント	変更履歴
SAP NetWeaver Java Stack エージェント	変更履歴
Siebel エージェント	変更履歴
Skype for Business Server エージェント	変更履歴
Sterling Connect Direct エージェント	変更履歴
Sterling File Gateway エージェント	変更履歴
Sybase エージェント	変更履歴
Synthetic Playback エージェント	変更履歴
Tomcat エージェント	変更履歴
UNIX OS エージェント	変更履歴
VMware VI エージェント	変更履歴
WebLogic エージェント	変更履歴
WebSphere Applications エージェント	変更履歴
WebSphere Infrastructure Manager エージェント	変更履歴
WebSphere MQ エージェント	変更履歴
Windows OS エージェント	変更履歴

機能

エージェントおよびデータ・コレクターの機能はオファリングによって異なります。エージェントおよびデータ・コレクターの主要機能は、リソース・モニター、トランザクション・トラッキング、および診断です。IBM Cloud Application Performance Management の任意のオファリングおよびアドオンにサブスクライブすることができます。 アドオンには、特定のオファリングが必要です。

各エージェントおよびデータ・コレクターは、そのエージェントまたはデータ・コレクターの名前に含められているリソースをモニターします。例えば、Monitoring Agent for Cisco UCS は Cisco UCS のリソースをモニターします。

開発者であるか、運用しているのか、基幹業務の所有者であるかに応じて、異なる Cloud APM 機能を使用します。

- リソース・モニター機能には、Response Time Monitoring、アプリケーション・リソース・モニター、およびインフラストラクチャー・リソース・モニターがあります。すべてのエージェントとデータ・コレクターが、リソース・モニター機能を提供できます。
- トランザクション・トラッキング機能は、トランザクション・インスタンスとトポロジーの情報を提供します。
- 診断機能には、個別の要求とメソッド呼び出し (必要な場合) のトレースおよび分析が含まれます。

要確認 : リソース・モニター機能は、すべてのオファリングとアドオンで共通です。診断機能とトランザクション・トラッキング機能は、Cloud APM, Advanced オファリングおよび アドオンでのみ使用可能です。

モニターするアプリケーション用のエージェントおよびデータ・コレクターは、「**製品およびサービス**」からダウンロードできます。エージェントのインストールには数分かかります。データ・コレクターはインストールを必要とせず、ダウンロード完了後に構成する必要があるのみです。エージェントのインストール手順については、123 ページの『第 6 章 エージェントのインストール』を参照してください。

55 ページの表 3 に、エージェントとデータ・コレクターの包括的なリスト、エージェントまたはデータ・コレクターが含まれているオファリングまたはアドオン、およびエージェントまたはデータ・コレクターの機能を示します。アドオン (Infrastructure Extension Pack など) がエージェントまたはデータ・コレクターに対して記載されている場合、そのアドオンは必須です。トランザクション・トラッキング機能または診断機能 (あるいはその両方) をサポートするエージェントおよびデータ・コレクターは、Cloud APM, Advanced 列にも示されています。

✔ は、エージェントまたはデータ・コレクターがこのオファリングで使用可能であり、リソース・モニター機能を提供できることを示します。
— は、このオファリングではデータまたは機能が使用できない、またはエージェントまたはデータ・コレクターに対してアドオンが不要であることを示しています。
TT はトランザクション・トラッキングを示します。
DD は診断を示します。

表 3. 各オファリングのエージェントおよびデータ・コレクターの機能

エージェントおよびデータ・コレクター	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	アドオン (必要な場合)
Amazon EC2 エージェント	✔	✔	Infrastructure Extension Pack
Amazon ELB エージェント	✔	✔	Infrastructure Extension Pack
Azure Compute エージェント	✔	✔	Infrastructure Extension Pack
Cassandra エージェント	✔	✔	Base Extension Pack
Cisco UCS エージェント	✔	✔	—
Citrix VDI エージェント	✔	✔	Infrastructure Extension Pack
Db2 エージェント	✔	✔	—
DataPower エージェント	✔	✔ TT	—

表 3. 各オファリングのエージェントおよびデータ・コレクターの機能 (続き)			
エージェントおよびデータ・コレクター	**Cloud APM, BaseCloud APM, Base Private**	**Cloud APM, Advanced**	**アドオン (必要な場合)**
DataStage エージェント	✓	✓	Base Extension Pack
Hadoop エージェント	✓	✓	Base Extension Pack
HMC Base エージェント	✓	✓	—
HTTP Server エージェント	✓	✓ TT	—
IBM Cloud エージェント	✓	✓	Infrastructure Extension Pack
IBM Integration Bus エージェント	—	✓ TT	—
Internet Service Monitoring	—	—	Base Extension Pack
オンプレミス・アプリケーション用の J2SE データ・コレクター	—	✓ TT DD	—
JBoss エージェント	✓	✓ TT DD	—
Liberty データ・コレクター (IBM Cloud アプリケーションおよびオンプレミス・アプリケーション用)	—	✓ TT DD	—
Linux KVM エージェント	✓	✓	—
Linux OS エージェント	✓	✓	—
Microsoft Active Directory エージェント	✓	✓	—
Microsoft Cluster Server エージェント	✓	✓	—
Microsoft Exchange Server エージェント	✓	✓	—
Microsoft Hyper-V Server エージェント	✓	✓	—
Microsoft IIS エージェント	✓	✓	—
Microsoft .NET エージェント	✓	✓ TT DD	—
Microsoft Office 365 エージェント	✓	✓	Base Extension Pack

表 3. 各オファリングのエージェントおよびデータ・コレクターの機能 (続き)

エージェントおよびデータ・コレクター	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	アドオン (必要な場合)
Microsoft SharePoint Server エージェント	✓	✓	—
Microsoft SQL Server エージェント	✓	✓	—
MongoDB エージェント	✓	✓	—
MQ Appliance エージェント	—	✓	—
MySQL エージェント	✓	✓	—
NetApp Storage エージェント	✓	✓	—
Node.js エージェント	✓	✓ DD	—
Node.js データ・コレクター (IBM Cloud アプリケーションおよびオンプレミス・アプリケーション用)	—	✓ TT DD	—
OpenStack エージェント	✓	✓	—
Oracle Database エージェント	✓	✓	—
PHP エージェント	✓	✓	—
PostgreSQL エージェント	✓	✓	—
Python データ・コレクター (IBM Cloud アプリケーションおよびオンプレミス・アプリケーション用)	✓	✓ DD	—
RabbitMQ エージェント	—	✓	Advanced Extension Pack
Response Time Monitoring エージェント	✓	✓ TT	—
Ruby エージェント	✓	✓ DD	—
Ruby データ・コレクター (IBM Cloud アプリケーション用)	—	✓ DD	—
SAP エージェント	—	✓	—
SAP HANA データベース・エージェント	—	✓	Advanced Extension Pack
SAP NetWeaver Java Stack エージェント	—	✓ TT DD	Advanced Extension Pack
Siebel エージェント	✓	✓	—

表 3. 各オファリングのエージェントおよびデータ・コレクターの機能 (続き)			
エージェントおよびデータ・コレクター	**Cloud APM, BaseCloud APM, Base Private**	**Cloud APM, Advanced**	**アドオン (必要な場合)**
Skype for Business Server エージェント (以前の Microsoft Lync Server エージェント)	✓	✓	—
Sterling Connect Direct エージェント	✓	✓	Base Extension Pack
Sterling File Gateway エージェント	✓	✓	Base Extension Pack
Sybase エージェント	✓	✓	—
Tomcat エージェント	✓	✓ TT	—
UNIX OS エージェント	✓	✓	—
VMware VI エージェント	✓	✓	—
WebLogic エージェント	✓	✓ TT DD	—
WebSphere Applications エージェント	✓	✓ TT DD	—
WebSphere Infrastructure Manager エージェント	✓	✓	—
WebSphere MQ エージェント	—	✓ TT	—
Windows OS エージェント	✓	✓	—

エージェントまたはデータ・コレクターで、トランザクション・トラッキングまたは診断がデフォルトで有効化されているかどうかについて詳しくは、エージェントおよびデータ・コレクターのトランザクション・トラッキングの有効化を参照してください。事前定義の診断ダッシュボードについて詳しくは、エージェントおよびデータ・コレクターの診断ダッシュボードを参照してください。

説明

エージェントおよびデータ・コレクターの説明には、これらの各コンポーネントのモニター対象に関する情報と、各コンポーネントの詳細情報へのリンクが記載されています。

各エージェントおよびデータ・コレクターにはバージョン番号があり、エージェントまたはデータ・コレクターが更新されるたびに変更されます。いずれのリリースでも、新しいエージェントとデータ・コレクターの追加や、既存のエージェントとデータ・コレクターの更新が行われる可能性があります。最新バージョンのエージェントまたはデータ・コレクターがない場合、エージェントの更新を検討してください。ご使用の環境内のエージェントまたはデータ・コレクターのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。

それぞれのエージェントおよびデータ・コレクターの説明には、これらのコンポーネントに関する次の種類の詳細へのリンクが用意されています。

- 特定のエージェントまたはデータ・コレクター機能に関するエージェントまたはデータ・コレクター構成およびその他の情報

- Cloud APM のエージェントまたはデータ・コレクター・ダッシュボード、グループ・ウィジェット、しきい値、データ・セット、および属性 (メトリックと KPI) の説明が記載されたリファレンス PDF

Cloud APM V8 エージェントおよびデータ・コレクターと共存できる IBM Tivoli Monitoring V6 および V7 エージェントの資料へのリンクについては、951 ページの表 236 を参照してください。

Amazon EC2 モニター

Monitoring Agent for Amazon EC2 は、Amazon Elastic Compute Cloud (EC2) インスタンスの正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。このエージェントは、CPU 使用率、Elastic Block Store (EBS) 使用率、ネットワーク使用状況、Amazon Web Services (AWS) の保守更新、およびディスク・パフォーマンスなどの、EC2 環境に関して十分な情報を元に意思決定するのに役立つ包括的なメトリック・セットを表示します。

- インストール後のエージェントの構成については、193 ページの『Amazon EC2 モニターの構成』 を参照してください。
- ダッシュボード、しきい値、および属性については、「Amazon EC2 エージェント Reference」を参照してください。

AWS Elastic Load Balancer モニター

Amazon ELB エージェントは、AWS Elastic Load Balancer の正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、AWS Elastic Load Balancer 環境に関して十分な情報に基づいて意思決定するために役立つ、各ロード・バランサー・タイプ (アプリケーション、ネットワーク、およびクラシック) に関する包括的な一連のメトリックが表示されます。

- インストール後のエージェントの構成については、201 ページの『AWS Elastic Load Balancer モニターの構成』 を参照してください。
- ダッシュボード、しきい値、および属性については、「Amazon ELB エージェント Reference」を参照してください。

Azure Compute モニター

Azure Compute エージェントは、Azure Compute インスタンスの正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、Azure Compute 環境に関して十分な情報に基づいて意思決定する際に役立つ、包括的な一連のメトリックが表示されます。これらのメトリックには、CPU 使用率、ネットワーク使用状況、およびディスク・パフォーマンスが含まれます。

- インストール後のエージェントの構成については、206 ページの『Azure Compute モニター構成』 を参照してください。
- ダッシュボード、しきい値、および属性については、「Azure Compute エージェント Reference」を参照してください。

Cassandra モニター

Monitoring Agent for Cassandra には、Cassandra クラスターをモニターする機能が用意されています。Cassandra クラスターのノード、キースペース、およびカラム・ファミリーに関する情報を収集し、分析できます。

- インストール後のエージェントの構成については、217 ページの『Cassandra モニターの構成』 を参照してください。
- ダッシュボード、しきい値、および属性については、「Cassandra エージェント Reference」を参照してください。

Cisco UCS のモニター

Monitoring Agent for Cisco UCS には、Cisco UCS の正常性、ネットワーク、およびパフォーマンスをモニターする環境が用意されています。Cisco UCS エージェントには、問題の早期検出と防止に必要な Cisco UCS 固有情報を収集および分析するための包括的な方法が用意されています。

- インストール後のエージェントの構成については、220 ページの『Cisco UCS モニターの構成』 を参照してください。
- ダッシュボード、しきい値、および属性については、「Cisco UCS エージェント Reference」を参照してください。

Citrix Virtual Desktop Infrastructure モニター
Monitoring Agent for Citrix Virtual Desktop Infrastructure は、Citrix 仮想デスクトップ・インフラストラクチャーの正常性、可用性、およびパフォーマンスについてモニターする 中央ポイントを提供します。エージェントでは、サイト、マシン、アプリケーション、デスクトップ、セッション、ユーザーなどの XenDesktop または XenApp のリソースに関して十分な情報に基づいて意思決定する際に役立つ、包括的な一連のメトリックが表示されます。

- インストール後のエージェントの構成については、227 ページの『Citrix Virtual Desktop Infrastructure モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Citrix VDI エージェント Reference」を参照してください。

DataPower モニター
Monitoring Agent for DataPower には、エンタープライズ環境内の DataPower アプライアンスをモニターする中央ポイントが用意されています。アプライアンスに 共通の問題に関する通知を識別および受信できます。エージェントは、アプライアンスのパフォーマンス 、リソース、およびワークロードに関する情報も提供します。

- インストール後のエージェントの構成については、246 ページの『DataPower エージェントの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「DataPower agent Reference」を参照してください。
- IBM 統合スタックの一部として DataPower アプライアンスをモニターする 方法については、99 ページの『IBM 統合スタックのモニター』を参照してください。

Db2 のモニター
Monitoring Agent for Db2 には、Db2 環境をモニターする中央ポイントが用意されています。単一の IBM Performance Management console から、多数のサーバーをモニターできます。各サーバーは、1 つの Db2 エージェントによってモニターされます。アプリケーション、データベース、およびシステム・リソースに関係する情報を収集および分析できます。

- 新規バージョンのエージェントにアップグレードする 前の情報については、1142 ページの『AIX 上のエージェント: アップグレード前のエージェントの停止と slibclean の実行』を参照してください。
- インストール後のエージェントの構成については、249 ページの『Db2 モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Db2 エージェント Reference」を参照してください。
- IBM Java アプリケーション・スタックの一部としてデータベース・トランザクションをモニターする方法については、92 ページの『IBM Java アプリケーション・スタックのモニター』を参照してください。

Hadoop モニター
Monitoring Agent for Hadoop には、組織内の Hadoop クラスターをモニターする機能があります。このエージェントを使用して、データ・ノードおよび Java 仮想マシンの状況、ヒープ・メモリーおよび非ヒープ・メモリーの情報、Hadoop ノード、ファイル・システム、キューに関する情報など、Hadoop クラスターの情報を収集および分析できます。

- インストール後のエージェントの構成については、260 ページの『Hadoop モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Hadoop エージェント Reference」を参照してください。

HMC Base モニター
Monitoring Agent for HMC Base には、ハードウェア管理コンソール (HMC) をモニターする機能が用意されています。このエージェントは、HMC リソース (CPU、メモリー、ストレージ、ネットワーク) の可用性と正常性をモニターします。また、このエージェントは、Power サーバー、CPU プール、および LPAR の HMC インベントリーおよび構成について報告します。Power サーバー、LPAR、およびプールの CPU 使用率は、HMC パフォーマンス・サンプル・データを使用してモニターされます。

- インストール後のエージェントの構成については、269 ページの『HMC Base モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「HMC Base エージェント Reference」を参照してください。

HTTP Server のモニター

Monitoring Agent for HTTP Server は、IBM HTTP Server に関するパフォーマンス・データを収集します。例えば、サーバーの状況およびタイプ、サーバー・エラーの数、サーバーに対する正常なログインと失敗したログインの数などのサーバー情報が表示されます。 データ・コレクターは、HTTP Server エージェントに送信されたデータを収集します。エージェントは、モニター対象の IBM HTTP Server と同じシステム上で実行されます。モニター対象の各サーバーは、サブノードとして登録されます。IBM HTTP Server Response Time モジュールは、HTTP Server エージェントとともにインストールされます。 Response Time Monitoring エージェント、WebSphere Application エージェント、およびデータベース・エージェントとともに HTTP Server エージェントを使用した場合、ブラウザーから IBM Java アプリケーション・スタックのデータベースへのトランザクション・モニター情報を表示できます。

- エージェントのインストールを開始する前に、『AIX システムでのプリインストール - HTTP Server エージェント』および『Linux システムでのプリインストール - HTTP Server エージェント』を参照してください。
- データ・コレクターの設定を確認し、エージェントのインストール後にデータ・コレクターをアクティブにする方法については、275 ページの『HTTP Server モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「HTTP Server エージェント Reference」を参照してください。
- IBM Java アプリケーション・スタックの一部として HTTP サーバーのトランザクションをモニターする方法については、92 ページの『IBM Java アプリケーション・スタックのモニター』を参照してください。

IBM Cloud Monitoring

Monitoring Agent for IBM Cloud は、IBM Cloud (Softlayer) アカウントから仮想マシンのインベントリーおよびメトリックを収集します。IBM Cloud で構成して実行している仮想デバイスの数を追跡するには、IBM Cloud エージェントを使用します。詳細なダッシュボード・ページで各仮想デバイスに割り振られているリソースを確認できます。このページには、デバイスが置かれているデータ・センター、オペレーティング・システム、およびその月の予想されるパブリック・ネットワーク帯域幅などの情報も表示されます。

- インストール後のエージェントの構成については、『IBM Cloud Monitoring の構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「IBM Cloud エージェント Reference」を参照してください。

IBM Integration Bus のモニター

Monitoring Agent for IBM Integration Bus は、IBM WebSphere Message Broker 製品と IBM Integration Bus 製品に関連付けられているメッセージ・ブローカー・トポロジーを検証、分析、および調整する手段が用意されているモニターおよび管理のツールです。

- インストール後のエージェントの構成については、282 ページの『IBM Integration Bus のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「IBM Integration Bus エージェント Reference」を参照してください。
- IBM 統合スタックの一部として IBM Integration Bus ブローカーをモニターする方法については、99 ページの『IBM 統合スタックのモニター』を参照してください。

InfoSphere DataStage のモニター

InfoSphere DataStage のモニター・エージェントは、DataStage サーバーの使用可能性、リソースの使用状況、およびパフォーマンスをモニターします。このエージェントは、エンジンのノードとジョブの正常性状況をモニターします。ユーザーは、エージェントが収集した情報を分析し、DataStage サーバー内の問題を解決するために適切なアクションを取ることができます。

- インストール後のエージェントの構成については、『InfoSphere DataStage のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「DataStage エージェント Reference」を参照してください。

Internet Service Monitoring

Internet Service Monitoring は、特定のサービスが適切に実行されているかどうかを判別し、問題の領域を特定し、サービス・レベル・アグリーメントに照らして測定したサービス・パフォーマンスを報告します。Internet Service Monitoring エージェント は、実ユーザーのアクションをエミュレートすることによって機能します。インターネット・サービスを定期的にポーリングまたはテストして、状況およびパフォーマンスを検査します。

- インストール後のエージェントの構成については、451 ページの『Windows システムでのエージェントの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Internet Service Monitoring agent Reference」を参照してください。

J2SE データ・コレクターのモニター

J2SE データ・コレクターは、Java アプリケーションに関するリソース・モニターと詳細診断のデータを収集します。詳細診断データは、要求および集約された情報に基づいてダッシュボードに表示され、さまざまなドリルダウン・ビューをサポートします。リソース・モニターと詳細診断の両方がサポートされており、Java アプリケーションの問題の検出、切り分け、および診断に役立ちます。低速要求を診断するようにデータ・コレクターを構成できます。

- データ・コレクターの構成方法については、J2SE データ・コレクター・モニターの構成を参照してください。
- ダッシュボード、しきい値、および属性については、「J2SE data collector Reference」を参照してください。

JBoss モニター

Monitoring Agent for JBoss は、JBoss Application Server および JBoss Enterprise Application Platform のリソースをモニターします。JBoss エージェントに付属のダッシュボードを使用して、JBoss アプリケーション・サーバー上の最も低速のアプリケーション、最も低速の要求数、スレッド・プールのボトルネック、JVM ヒープ・メモリーとガーベッジ・コレクションの問題、最もビジーなセッション、その他のボトルネックを特定します。

- インストール後のエージェントの構成については、461 ページの『JBoss モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「JBoss エージェント Reference」を参照してください。

Linux KVM モニター

Monitoring Agent for Linux KVM は、複数インスタンスの複数接続エージェントであり、Enterprise Linux ベースの KVM ハイパーバイザー環境と Red Hat Enterprise Virtualization Manager (RHEV-M) 環境への接続をサポートします。RHEV-M ハイパーバイザー環境、または KVM ハイパーバイザー環境の複数のハイパーバイザーをモニターするために、このエージェントを複数インスタンス作成できます。さまざまな仮想マシンにまたがって、仮想化ワークロードをモニターし、リソース・キャパシティーを分析できます。エージェントを KVM ハイパーバイザー環境の仮想マシンに接続するには、前提条件である `libvirt*.rpm` および Korn シェル・インタープリター (pdksh) をインストールする必要があります。このエージェントは、仮想マシンを管理する libvirt ハイパーバイザーにリモートで接続することで、メトリックを収集します。

- インストール後のエージェントの構成については、477 ページの『Linux KVM のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Linux KVM エージェント Reference」を参照してください。

Linux OS モニター

Monitoring Agent for Linux OS には、Linux OS 環境の可用性、パフォーマンス、およびリソースの使用状況をモニターする機能が用意されています。このエージェントでは、Docker コンテナーのモニターがサポートされます。例えば、CPU 使用率、メモリー、ネットワーク、入出力の使用情報など、Docker コンテナーに関連する詳細情報が表示されます。Docker ID やインスタンス名など、サーバーで実行されている Docker コンテナーに関する一般情報も表示されます。また、アプリケーション・ログ・ファイルをモニターするようにログ・ファイル・モニターを構成できます。オペレーティング・システムと CPU パフォーマンス、Linux のディスク情報とパフォーマンス分析、プロセス状況分析、ネットワーク・パフォーマンスなどのサーバー固有の情報を収集および分析できます。

- インストール後のログ・ファイル・モニターの構成については、635 ページの『OS エージェント・ログ・ファイル・モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Linux OS エージェント Reference」を参照してください。

MariaDB のモニター

Monitoring Agent for MariaDB は、MariaDB 環境またはアプリケーションを一元的に管理できるようにします。本製品は、問題を早期に検出して回避するために必要な情報を収集するための包括的な手段を提供します。情報は、システム全体で標準化されます。単一のコンソールから複数のサーバーをモニターできます。Monitoring Agent for MariaDB を使用することで、MariaDB 固有の情報を簡単に収集して分析できます。
インストール後のエージェントの構成については、488 ページの『MariaDB モニターの構成』を参照してください。

Microsoft Active Directory モニター

Monitoring Agent for Microsoft Active Directory には、組織内の Active Directory をモニターする機能があります。このエージェントを使用して、Active Directory に固有の情報 (ネットワーク状況、Sysvol 複製、アドレス帳のパフォーマンス、ディレクトリーのシステム 使用量など) を収集および分析できます。

- インストール後のエージェントの構成については、491 ページの『Microsoft Active Directory モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft Active Directory エージェント Reference」を参照してください。

Microsoft Cluster Server モニター

Monitoring Agent for Microsoft Cluster Server には、組織内の Microsoft Cluster Server をモニターする機能が用意されています。Microsoft Cluster Server エージェントを使用して、クラスター・リソースの可用性に関連する情報 (クラスター・レベル、クラスター・ノード、クラスター・リソース・グループ、クラスター・リソース、クラスター・ネットワークなど) を収集できます。このエージェントには、クラスター・リソースの使用状況 (プロセッサー使用率、メモリー使用量、ディスク使用率、ネットワーク使用状況など) の統計機能も用意されています。

- インストール後のエージェントの構成については、498 ページの『Microsoft Cluster Server のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft Cluster Server エージェント Reference」を参照してください。

Microsoft Exchange Server のモニター

Monitoring Agent for Microsoft Exchange Server には、組織内の Exchange Server の正常性、可用性、およびパフォーマンスをモニターする 機能が用意されています。Microsoft Exchange Server エージェントを使用して、メール・トラフィック、メールボックス・データベースの状態、クライアントのアクティビティーなどのサーバー 固有の情報を収集できます。また、このエージェントは、Exchange Server のパフォーマンスを分析するのに役立つ、キャッシュ使用量、メールの使用状況、データベースの使用状況、およびクライアント・アクティビティーの 統計を提供します。

- インストール後のエージェントの構成については、500 ページの『Microsoft Exchange のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft Exchange Server エージェント Reference」を参照してください。

Microsoft Hyper-V Server モニター

Monitoring Agent for Microsoft Hyper-V Server には、組織内のすべての Hyper-V システムの可用性およびパフォーマンスをモニターする 機能が用意されています。Microsoft Hyper-V Server エージェントは、仮想マシンの数、仮想マシンの状態、割り振られている仮想ディスクの数、割り振られている仮想メモリー、割り振られている仮想プロセッサーの数などの構成情報を提供します。また、このエージェントは、物理プロセッサー使用率、メモリー使用量、ネットワーク使用状況、論理プロセッサー使用率、および仮想プロセッサー使用率の統計を提供します。

- インストール後のエージェントの構成については、513 ページの『Microsoft Hyper-V のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft Hyper-V Server エージェント Reference」を参照してください。

Microsoft Internet Information Services モニター

Monitoring Agent for Microsoft Internet Information Services には、Microsoft Internet Information Server の可用性およびパフォーマンスをモニターする 機能が用意されています。Microsoft Internet Information Server エージェントを使用して、要求率、データ転送速度、エラー統計、接続統計などの Web サイトの詳細をモニターできます。

- インストール後のエージェントの構成については、517 ページの『Microsoft IIS モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft IIS エージェント Reference」を参照してください。

Microsoft .NET モニター

Monitoring Agent for Microsoft .NET は、インターネット・インフォメーション・サービス (IIS) および Microsoft .NET Framework のリソースを基盤とする Microsoft .NET アプリケーションをモニターします。データ・コレクター・コンポーネントは、着信 HTTP 要求のデータを収集します。データ・コレクターは、メソッド呼び出しを収集し、呼び出しツリーを構築して、要求コンテキストおよびスタック・トレース・データを収集します。Microsoft .NET エージェントに付属のダッシュボードを使用して、Microsoft .NET Framework に関連付けられた問題を特定します。また、最も低速の HTTP 要求を特定し、そこからスタック・トレース情報にドリルダウンして、問題を切り分けることもできます。

- インストール後のエージェントの構成については、527 ページの『データ・コレクターの登録』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft .NET エージェント Reference」を参照してください。

Microsoft Office 365 のモニター

Monitoring Agent for Microsoft Office 365 には、Microsoft Office 365 をモニターする機能が用意されています。Microsoft Exchange Online、SharePoint Online、Skype for Business、および OneDrive for Business に関する情報を収集し、分析できます。

- インストール後のエージェントの構成については、536 ページの『Microsoft Office 365 のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft Office 365 エージェント Reference」を参照してください。

Microsoft SharePoint Server のモニター

Monitoring Agent for Microsoft SharePoint Server には、Microsoft SharePoint Server の可用性、イベント、およびパフォーマンスをモニターする 環境が用意されています。このエージェントを使用して、Microsoft SharePoint Server からデータを収集し、操作を管理します。

- インストール後のエージェントの構成については、541 ページの『Microsoft SharePoint Server モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft SharePoint Server エージェント Reference」を参照してください。

Microsoft SQL Server モニター

Monitoring Agent for Microsoft SQL Server には、Microsoft SQL Server をモニターする機能が用意されています。Microsoft SQL Server エージェントは、分散データベースの一元的な管理を可能にします。

Microsoft SQL Server エージェント・ダッシュボードを使用して、モニター対象のすべての SQL Server インスタンスの可用性、パフォーマンス、リソースの使用状況、および全体的な状況をモニターします。

- インストール後のエージェントの構成については、545 ページの『Microsoft SQL Server のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Microsoft SQL Server エージェント Reference」を参照してください。

MongoDB のモニター

Monitoring Agent for MongoDB には、MongoDB デプロイメントの使用状況、状況、およびパフォーマンスをモニターする機能が用意されています。データベース容量の使用率、開かれている接続のパーセンテージ、メモリー使用量、インスタンス状況、応答時間などの情報を、視覚的なダッシュボードで収集および分析できます。

- インストール後のエージェントの構成については、577 ページの『MongoDB のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「MongoDB エージェント Reference」を参照してください。

MQ Appliances モニター

Monitoring Agent for MQ Appliance は、「MQ アプライアンス」で MQ アプライアンス・レベルに重点を置いたモニター情報 (例えば、CPU、メモリー、ストレージ、センサー、およびキュー・マネージャーの要約情報) を提供します。

- インストール後のエージェントの構成については、297 ページの『IBM MQ Appliances モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「MQ Appliance agent Reference」を参照してください。

MySQL のモニター

Monitoring Agent for MySQL には、MySQL デプロイメントの状況、使用状況、およびパフォーマンスをモニターする機能が用意されています。受信バイトと送信バイト、InnoDB バッファー・プール・ページ、パフォーマンス履歴などの情報を収集および分析できます。

- エージェントのインストールを開始する前に、『Linux システムでのプリインストール - MySQL エージェント』または『Windows システムでのプリインストール - MySQL エージェント』を参照してください。
- インストール後のエージェントの構成については、582 ページの『MySQL のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「MySQL エージェント Reference」を参照してください。

NetApp ストレージ・モニター

Monitoring Agent for NetApp Storage には、NetApp OnCommand Unified Manager (OCUM) を使用して NetApp ストレージ・システムをモニターする 機能が用意されています。NetApp ストレージ・システムの集約、ノード、ディスク、およびボリュームに関する情報を収集および分析できます。

- インストール後のエージェントの構成については、585 ページの『NetApp ストレージのモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「NetApp Storage エージェント Reference」を参照してください。

Node.js のモニター

Monitoring Agent for Node.js またはスタンドアロンの Node.js データ・コレクターを使用して、Node.js アプリケーションのパフォーマンスに関するデータを測定および収集できます。例えば、HTTP 要求のスループットや応答時間のほか、リソースの使用状況に関する測定値をモニターおよび格納して、表示や分析を行うことができます。Node.js エージェントおよび Node.js データ・コレクターのどちらかを選択するには、591 ページの『Node.js モニターの構成』の説明を参照してください。

Node.js エージェント

- インストールを開始する前に、『Linux システムでのプリインストール - Node.js エージェント』を参照してください。
- インストール後のエージェントの構成については、593 ページの『Node.js エージェントの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Node.js エージェント Reference」を参照してください。

Node.js データ・コレクター (スタンドアロン)

Node.js データ・コレクターは、IBM Cloud アプリケーションおよびオンプレミス・アプリケーションをモニターします。リソース・モニターおよび詳細診断がサポートされていて、アプリケーションの問題の検出、切り分け、および診断に役立ちます。個々の要求とメソッド呼び出しのパフォーマンスを追跡するようにデータ・コレクターを構成し、その情報を使用して低速の要求を診断し、それに応じてアクションを実行できます。

IBM Cloud アプリケーション

- データ・コレクターの構成方法については、598 ページの『IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Data collectors Reference」を参照してください。

オンプレミス・アプリケーション

- データ・コレクターの構成方法については、603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Data collectors Reference」を参照してください。

OpenStack モニター

Monitoring Agent for OpenStack には、OpenStack アプリケーションをモニターする 機能が用意されています。API エンドポイント、SSH サーバー接続、プロセス、およびハイパーバイザーに関する情報など、OpenStack アプリケーションのパフォーマンスを 表示するには、ダッシュボードを使用します。

- インストール後のエージェントの構成については、613 ページの『OpenStack エージェントの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「OpenStack エージェント Reference」を参照してください。

Oracle データベースのモニター

Monitoring Agent for Oracle Database には、Oracle データベースの可用性、パフォーマンス、およびリソースの使用状況をモニターする機能が用意されています。さまざまな Oracle Database エージェントをモニターするために、複数の Oracle Database インスタンスを構成できます。このエージェントには、リモート・モニター機能も用意されています。

- エージェントのインストールを開始する前に、『AIX システムでのプリインストール - Oracle Database エージェント』、『Linux システムでのプリインストール - Oracle Database エージェント』、または『Windows システムでのプリインストール - Oracle Database エージェント』(Windows) を参照してください。
- インストール後のエージェントの構成については、618 ページの『Oracle データベースのモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Oracle Database エージェント Reference」を参照してください。
- IBM Java アプリケーション・スタックの一部としてデータベース・トランザクションをモニターする方法については、92 ページの『IBM Java アプリケーション・スタックのモニター』を参照してください。

PHP モニター

Monitoring Agent for PHP は、Apache Web サーバーを通じて Web アクセス・メトリックを、MySQL からはパフォーマンス統計データを収集することで、PHP Web アプリケーションをモニターします。このエージェントは、Apache サーバー上のすべての WordPress アプリケーションをディスカバーし、WordPress アプリケーションの統計情報を提供します。PHP エージェントを使用して、Web サーバーの可用性、Apache サーバーの状況、および GET/POST 要求をモニターします。エージェントは、WordPress アプリケーションでの PHP 要求のパフォーマンスのみを評価します。CSS および JS のロードは評価されません。エージェントは、URL を識別するために URL 引数を使用しません。

- インストール後のエージェントの構成については、666 ページの『PHP モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「PHP エージェント Reference」を参照してください。

PostgreSQL のモニター

Monitoring Agent for PostgreSQL は、JDBC ドライバーを通じて PostgreSQL メトリックを収集することで、PostgreSQL データベースをモニターします。このエージェントは、システム・リソースの使用状況、データベース容量、使用された接続、実行中インスタンスの個々の状況、操作の統計、SQL 照会ステートメントの応答時間、データベース・サイズの詳細、およびロック情報に関するデータを提供します。

- インストール後のエージェントの構成については、668 ページの『PostgreSQL のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「PostgreSQL エージェント Reference」を参照してください。

Python モニター

Python データ・コレクターは、オンプレミスと IBM Cloud の両方の Python アプリケーションをモニターします。リソース・モニターと詳細診断の両方がサポートされていて、CPU とメモリーの使用量、ガーベッジ・コレクション、およびスレッドなどのモニター・データが提供されます。個々の要求とメソッド呼び出しのパフォーマンスを追跡するようにデータ・コレクターを構成し、その情報を使用して低速の要求を診断し、それに応じてアクションを実行できます。

IBM Cloud アプリケーション

- データ・コレクターの構成方法については、673 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、Data collectors Reference を参照してください。

オンプレミス・アプリケーション

- データ・コレクターの構成方法については、678 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Data collectors Reference」を参照してください。

RabbitMQ モニター

Monitoring Agent for RabbitMQ には、RabbitMQ クラスターをモニターする機能が用意されています。RabbitMQ クラスターのノード、キュー、およびチャネルに関する情報を収集し、分析できます。

- インストール後のエージェントの構成については、683 ページの『RabbitMQ モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「RabbitMQ エージェント Reference」を参照してください。

Response Time Monitoring

Response Time Monitoring エージェントは、ネットワーク・モニターを使用して、応答時間や状況コードなどの HTTP および HTTPS トランザクション・データを収集します。Response Time Monitoring エージェントを使用して、ユーザーに対する Web アプリケーションのパフォーマンスおよび 可用性 (トラ

ンザクション要求、アプリケーション、およびサーバー情報を含む) をモニターします。また、このエージェントを使用して、デバイス情報およびセッション情報をモニターします。

- Response Time Monitoring エージェントのインストールを開始する前に、『AIX システムでのプリインストール - Response Time Monitoring エージェント』、『Linux システムでのプリインストール - Response Time Monitoring エージェント』、または『Windows システムでのプリインストール - Response Time Monitoring エージェント』を参照してください。
- インストール後のエージェントの構成については、690 ページの『JavaScript インジェクション』を参照してください。
- ダッシュボード、しきい値、および属性については、「Transaction Monitoring Reference」を参照してください。
- IBM Java アプリケーション・スタックの一部として Response Time Monitoring を使用する方法については、92 ページの『IBM Java アプリケーション・スタックのモニター』を参照してください。

Ruby モニター

Monitoring Agent for Ruby またはスタンドアロンの Ruby データ・コレクターは、Ruby on Rails アプリケーションのパフォーマンス (要求トラフィックおよび構成統計を含む) をモニターします。診断機能を使用して、各アプリケーションをさらに深く調べることもできます。

スタンドアロン Ruby データ・コレクターは、IBM Cloud アプリケーションのみをモニターします。

Ruby エージェント

- インストール後のエージェントの構成については、718 ページの『Ruby モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Ruby エージェント Reference」を参照してください。

Ruby データ・コレクター (スタンドアロン)

Ruby データ・コレクターを使用して IBM Cloud アプリケーションをモニターできます。リソース・モニターと詳細診断の両方がサポートされていて、アプリケーションの問題の検出、切り分け、および診断に役立ちます。個々の要求とメソッド呼び出しのパフォーマンスを追跡するようにデータ・コレクターを構成し、その情報を使用して低速の要求を診断し、それに応じてアクションを実行できます。

IBM Cloud アプリケーション

- データ・コレクターの構成方法については、データ・コレクターの構成を参照してください。
- ダッシュボード、しきい値、および属性については、Data collectors Reference を参照してください。

SAP アプリケーション・モニター

Monitoring Agent for SAP Applications には、Advanced Business Application Programming (ABAP) スタック上で稼働する SAP アプリケーションをモニターする 機能が用意されています。エージェントは、SAP のライフサイクル管理ツールである SAP Solution Manager と、SAP 用のエンタープライズ統合ソフトウェアである SAP NetWeaver プロセス統合 (SAP PI) もモニターします。これには、問題を早期に検出し、再発を防止するための手段を講じるのに必要な情報を収集するための管理の中央ポイントが用意されています。これにより、複数の SAP リリース、アプリケーション、およびコンポーネント、さらに基盤になるデータベース、オペレーティング・システム、および外部インターフェースにまたがる効果的なシステム管理が可能になります。

- インストール後のエージェントの構成については、730 ページの『SAP モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「SAP エージェント Reference」を参照してください。

SAP HANA データベース・モニター

Monitoring Agent for SAP HANA Database は、SAP HANA データベースの可用性、リソースの使用状況、およびパフォーマンスをモニターします。このエージェントは、単一ホストから単一データベース、単一ホストから複数テナント・データベース、複数ホストから単一データベース、および複数ホストから複数テナント・データベースへの、HANA デプロイメント・シナリオをモニターできます。エー

ジェントが収集した情報を分析し、SAP HANA データベースでの問題を解決するために適切なアクションを取ることができます。

- エージェントのインストールを開始する前に、『AIX システムでのプリインストール - SAP HANA データベース・エージェント』、『Linux システムでのプリインストール - SAP HANA データベース・エージェント』、または『Windows システムでのプリインストール - SAP HANA データベース・エージェント』を参照してください。
- インストール後のエージェントの構成については、762 ページの『SAP HANA データベースのモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「SAP HANA データベース・エージェント Reference」を参照してください。

SAP NetWeaver Java Stack モニター

Monitoring Agent for SAP NetWeaver Java Stack は、SAP NetWeaver Java スタックの可用性、リソースの使用状況、およびパフォーマンスをモニターします。このエージェントは、単一ホストから単一インスタンス、単一ホストから複数インスタンス、複数ホストから単一インスタンス、および複数ホストから複数インスタンスへの、SAP NetWeaver Java Stack のデプロイメント・シナリオをモニターできます。エージェントが収集した情報を分析し、SAP NetWeaver Java Stack での問題を解決するために適切なアクションを取ることができます。

- インストール後のエージェントの構成については、764 ページの『SAP NetWeaver Java スタックのモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「SAP NetWeaver Java Stack エージェント Reference」を参照してください。

Siebel モニター

Monitoring Agent for Siebel は、Siebel 統計、ユーザー・セッション、コンポーネント、タスク、アプリケーション・サーバー、Siebel Gateway Name Server、プロセス CPU およびメモリー使用量、ログ・イベントのモニターなど、Siebel リソースに対するモニターの中心点となります。

- インストール後のエージェントの構成については、771 ページの『Siebel モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Siebel エージェント Reference」を参照してください。

Skype for Business Server (以前の Microsoft Lync Server) のモニター

Monitoring Agent for Skype for Business Server には、Skype for Business Server の正常性、可用性、およびパフォーマンスをモニターする 機能が用意されています。Skype for Business Server エージェントを使用して、待ち時間、シンセティック・トランザクション、発着信詳細記録 (CDR) サービス書き込み操作、スロットル要求の状態、セッション開始プロトコル (SIP) ピアなどのサーバー固有の情報を収集できます。また、このエージェントは、Lync Server または Skype for Business Server のパフォーマンスの分析に役立つ、インスタント・メッセージングおよびメディエーション ・サーバーの履歴使用統計を提供します。

- インストール後のエージェントの構成については、520 ページの『Skype for Business Server (以前の Microsoft Lync Server) のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Skype for Business Server エージェント Reference」を参照してください。

Sterling Connect Direct モニター

Monitoring Agent for Sterling Connect Direct は、Connect Direct サーバーをモニターします。これにより、サーバーの正常性およびパフォーマンスの情報が提供されます。また、ファイル転送アクティビティーの分析も行います。

- インストール後のエージェントの構成については、783 ページの『Sterling Connect Direct モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Sterling Connect Direct エージェント Reference」を参照してください。

Sterling File Gateway モニター
Monitoring Agent for Sterling File Gateway は Sterling File Gateway アプリケーションをモニターします。このアプリケーションは、さまざまなプロトコル、さまざまなファイル命名規則、およびさまざまなファイル形式を使用した内部パートナーと外部パートナーの間のファイル転送に使用されます。また、リモート・モニター機能もサポートします。

- インストール後のエージェントの構成については、786 ページの『Sterling File Gateway モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Sterling File Gateway エージェント Reference」を参照してください。

Sybase サーバーのモニター
Monitoring Agent for Sybase Server は、分散データベースの一元的な管理を可能にします。データベース管理者およびシステム管理者が Sybase サーバー・システムのパフォーマンスを 調べるために必要な情報を収集し、問題を早期に検出し、問題を予防します。

- インストール後のエージェントの構成については、792 ページの『Sybase サーバーのモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、Sybase エージェントリファレンスを参照してください。.

Tomcat モニター
Monitoring Agent for Tomcat は、Tomcat アプリケーション・サーバーのリソースをモニターします。Tomcat エージェントに付属のダッシュボードを使用して、Tomcat アプリケーション・サーバー上の最も低速のアプリケーション、最も低速の要求数、スレッド・プールのボトルネック、JVM ヒープ・メモリーとガーベッジ・コレクションの問題、最もビジーなセッション、その他のボトルネックを特定します。

- インストール後のエージェントの構成については、801 ページの『Tomcat モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Tomcat エージェント Reference」を参照してください。

UNIX OS モニター
Monitoring Agent for UNIX OS には、UNIX OS 環境の可用性、パフォーマンス、およびリソースの使用状況をモニターする機能が用意されています。(AIX および Solaris オペレーティング・システムのみ。APM Developer Center の「System requirements」を参照してください。) また、アプリケーション・ログ・ファイルをモニターするようにログ・ファイル・モニターを構成できます。オペレーティング・システムと CPU パフォーマンス、UNIX のディスク情報とパフォーマンス分析、プロセス状況分析、ネットワーク・パフォーマンスなどのサーバー固有の情報を収集および分析できます。

- インストール後のログ・ファイル・モニターの構成については、635 ページの『OS エージェント・ログ・ファイル・モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「UNIX OS エージェント Reference」を参照してください。

VMware VI のモニター
Monitoring Agent for VMware VI は、VMware Virtual Center に接続することにより、VMware Virtual Infrastructure をモニターします。VMware VI エージェントを使用して、単一のコンソールから、クラスターの状況要約を表示し、複数のコンポーネント (クラスター、仮想マシン、データ・ストア、ESX サーバーなど) をモニターできます。

- インストール後のエージェントの構成については、809 ページの『VMware VI のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「VMware VI エージェント Reference」を参照してください。

WebLogic モニター
Monitoring Agent for WebLogic は、WebLogic サーバー環境の正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。このエージェントでは、Java 仮想マシン (JVM)、

Java メッセージング・サービス (JMS)、Java Database Connectivity (JDBC) などの WebLogic リソースに関して十分な情報に基づく意思決定をするのに役立つ、包括的な一連のメトリックが表示されます。

- インストール後のエージェントの構成については、817 ページの『WebLogic モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「WebLogic エージェント Reference」を参照してください。

WebSphere アプリケーションのモニター

組み込みデータ・コレクターまたはスタンドアロンの Liberty データ・コレクターを備えた Monitoring Agent for WebSphere Applications は、WebSphere Application Server のリソースをモニターします。これらのモニター・コンポーネントは、以下を実行するように構成できます。

- アプリケーション・サーバー上の JMX インターフェースを 通じて、リソース・モニターのための PMI メトリックを収集する。
- 要求パフォーマンス・メトリックの総計を収集する。
- 個々の要求とメソッド呼び出しのパフォーマンスを追跡する。

モニター・データはダッシュボードに表示されます。付属のダッシュボードを使用して、アプリケーション・サーバーの具体的な問題領域を切り分けることができます。ドリルダウンして、問題が基盤になるリソースにあるか、アプリケーションのコードに関連するかを判別します。

エージェントといずれかのデータ・コレクターのどちらを使用するかについては、833 ページの『WebSphere Applications モニターの構成』を参照してください。

WebSphere Applications エージェントおよび組み込みデータ・コレクター

- インストール後のエージェントの構成については、834 ページの『WebSphere Applications エージェント 』を参照してください。
- ダッシュボード、しきい値、および属性については、「WebSphere Applications エージェント Reference」を参照してください。
- IBM Java アプリケーション・スタックの一部として WebSphere Application Server のトランザクションをモニターする方法については、92 ページの『IBM Java アプリケーション・スタックのモニター』を参照してください。

Liberty データ・コレクター (スタンドアロン)

Liberty データ・コレクターを使用して、IBM Cloud 上の WebSphere Liberty プロファイルをモニターしたり、Linux for System x 上の WebSphere Application Server Liberty をモニターしたりすることができます。リソース・モニター、診断、およびトランザクション・トラッキングがすべてサポートされており、アプリケーションの問題の検出、切り分け、および診断に役立ちます。個々の要求とメソッド呼び出しのパフォーマンスを追跡するようにスタンドアロンのデータ・コレクターを構成し、その情報を使用して低速の要求を診断し、それに応じてアクションを実行できます。

IBM Cloud アプリケーション

- データ・コレクターの構成方法については、886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、Data collectors Reference を参照してください。

オンプレミス・アプリケーション (Linux for System x のみ)

- データ・コレクターの構成方法については、881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、Data collectors Reference を参照してください。

WebSphere Infrastructure Manager モニター

Monitoring Agent for WebSphere Infrastructure Manager には、WebSphere Application Server デプロイメント・マネージャーとノード・エージェントのサーバー状況、リソース、トランザクションなどをモニターする機能が用意されています。WebSphere Infrastructure Manager エージェントによって収

集されたデータを使用して、デプロイメント・マネージャーとノード・エージェントのパフォーマンス、および問題が発生したかどうかを分析できます。

- インストール後のエージェントの構成については、931 ページの『WebSphere Infrastructure Manager モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「WebSphere Infrastructure Manager エージェント Reference」を参照してください。

WebSphere MQ のモニター

Monitoring Agent for WebSphere MQ を使用すると、キュー・マネージャーについての WebSphere MQ 固有のデータを、単一の分かりやすい視点から簡単に収集および分析できます。これで、収集したデータの傾向を追跡し、事前定義ダッシュボードを使用して、システムの問題をトラブルシューティングできます。

- インストール後のエージェントの構成については、931 ページの『WebSphere MQ のモニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「WebSphere MQ エージェント Reference」を参照してください。
- IBM 統合スタックの一部としてメッセージ・キューをモニターする方法については、99 ページの『IBM 統合スタックのモニター』を参照してください。

Windows OS のモニター

Monitoring Agent for Windows OS には、Windows OS 環境の可用性、パフォーマンス、およびリソースの使用状況をモニターする機能が用意されています。また、アプリケーション・ログ・ファイルをモニターするようにログ・ファイル・モニターを構成できます。オペレーティング・システムと CPU パフォーマンス、ディスク情報とパフォーマンス分析、プロセス状況分析、インターネット・セッション・データ、モニター対象のログ情報、インターネット・サーバー統計、メッセージ・キューイング統計、プリンターとジョブの状況データ、リモート・アクセス・サービス統計、サービス情報などのサーバー固有の情報を収集および分析できます。KNTCMA_FCProvider サービスはエージェントとともにインストールされます。

- インストール後のログ・ファイル・モニターの構成については、635 ページの『OS エージェント・ログ・ファイル・モニターの構成』を参照してください。
- ダッシュボード、しきい値、および属性については、「Windows OS エージェント Reference」を参照してください。

機能

主要機能はオファリングごとに異なります。機能の中には、1 つまたは両方のオファリングで使用できるもの、アドオンに含まれているもの、他の製品やコンポーネントとの統合により使用できるようになるものがあります。

アプリケーション・リソース・モニター

リソース・モニター・エージェントを使用して、言語およびミドルウェアをモニターします。適応範囲はオファリングによって異なります。54 ページの『機能』を参照してください。

オペレーティング・システム・モニター

リソース・モニター・エージェントを使用して、Linux、UNIX、および Windows の各オペレーティング・システムをモニターします。54 ページの『機能』を参照してください。

ログ・ファイル・モニター

OS エージェントには、アプリケーション・ログ・ファイルをモニターするための機能が組み込まれています。この機能には、正規表現に基づいたログ・ファイル・モニターを構成する機能が含まれます。

互換性を確保するために、OS エージェントは以下の情報およびフォーマットを使用します。

- IBM Tivoli Monitoring Log File Agent V6.x で使用された構成情報およびフォーマット・ファイル
- Tivoli Event Console ログ・ファイル・アダプターで 使用された構成情報および書式制御ストリング

これらの書式制御ストリングを使用することにより、エージェントはフォーマット・ファイル内のパターンに従ってログ・データをフィルタリングし、関係のあるデータのみをイベント・コンシューマーに

送信できます。OS エージェントはデータを、Cloud APM サーバーにも、Event Integration Facility (EIF) を介して Netcool/OMNIbus Probe for Tivoli EIF などの任意の EIF 受信側にも送信します。

ダッシュボード

Application Performance ダッシュボードは、ご使用の環境におけるアプリケーションの状況の概要を表示します。興味のある分野を表示するには、ナビゲーターからそれらを選択するか、要約ボックスをクリックして次のレベルにドリルダウンします。

各ダッシュボードのレベルで使用可能な機能については、1079 ページの『すべてのマイ・アプリケーション - Application Performance ダッシュボード』、1082 ページの『アプリケーション - Application Performance ダッシュボード』、および 1087 ページの『グループおよびインスタンス - Application Performance ダッシュボード』を参照してください。

Tivoli Monitoring ドメインと Cloud APM ドメインの KPI を同じダッシュボードに表示

IBM Tivoli Monitoring 製品と IBM Cloud Application Performance Management 製品の両方を含む環境では、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールして、 両方のドメインの管理対象システムを統合したビューを利用することができます。Cloud APM コンソールにハイブリッド環境を表示するには、 管理対象システム・グループを作成し、Tivoli Monitoring 環境にハイブリッド・ゲートウェイをインストールし、ハイブリッド・ゲートウェイとの通信を構成する必要があります。

詳しくは、949 ページの『IBM Tivoli Monitoring V6.3 との統合』を参照してください。

履歴メトリック

最大 24 時間の履歴データを Application Performance ダッシュボード上に可視化します。ダッシュボードの「**状況の概要**」タブに時間セレクターが表示されている場合は、履歴データのサンプルから値が導出されているグラフおよび表の時刻範囲を調整できます。折れ線グラフの場合、過去 24 時間までの現行データを最大 8 日間の履歴データと比較して、異常を特定することもできます。

詳しくは、1091 ページの『メトリックの調整および時間変化の比較』を参照してください。

IBM Cloud Application Business Insights Universal View

Universal View を使用して、モニターしているアプリケーションのカスタマイズされたページを作成できます。さまざまなグラフおよびメトリックのオプションから選択して、要件に従ってデータをモニターするウィジェットを作成します。Universal View を使用すると、ダッシュボードをカスタマイズして、複数のエージェントからの統合されたデータを表示できます。

ダッシュボードでデータを表示しているときに、グラフ・タイプを動的に変更できます。グリッド・ウィジェットで、データを動的にフィルタリングできます。

カスタマイズされたページ・データを生データ・ファイルにエクスポートできます。

詳しくは、1112 ページの『カスタム・ビュー』を参照してください。

アプリケーションの詳細

「**すべてのマイ・アプリケーション**」ダッシュボードから管理対象システム・インスタンスの詳細なダッシュボードにドリルダウンすると、「属性の詳細」タブが表示され、カスタムの履歴折れ線グラフと表を作成して管理し、保存できます。これに、追加のグラフや表のページを保存できます。このページは、自分の表示専用にすることも、同じ環境内のすべてのユーザーと共有することもできます。

詳しくは、1093 ページの『カスタムのグラフまたは表のページの作成』を参照してください。

API

Cloud APM の API は、ユーザーに役割を割り当てたりしきい値を作成したりするなど、環境を管理するために使用できます。詳しくは、1073 ページの『API の解説』を参照してください。

役割ベースのアクセス制御

Cloud APM での役割とは、ユーザーが実行できるアクションを制御する複数の権限から成るグループです。「役割ベースのアクセス制御」機能を使用して、カスタマイズされた役割を作成します。これはセキュリティーの基礎になります。また、事前定義の役割として、役割管理者、モニター管理者、 システム管理者、およびモニター・ユーザーの 4 つが用意されています。カスタマイズした役割と事前定義の役割の両方にユーザーを割り当てることができます。 ユーザーを複数の役割に割り当てることもできます。カスタマイズした役割に権限を割り当てたり、 既存のデフォルトの役割に追加の権限を割り

当てたりすることができます。許可は累積されます。 ユーザーには、割り当てられているすべての役割のすべての許可が割り当てられます。

個々のアプリケーション、システム・リソース・グループ、およびカスタム・リソース・グループに対する表示権限および変更権限を割り当てることができます。例えば、あるアプリケーションの 表示権限を持つ役割のメンバーは、そのアプリケーション内のサポートしているすべてのコンポーネントを表示できます。

システム管理タスクに対する表示権限および変更権限を割り当てることができます。例えば、「拡張構成」の表示権限を持つ役割のメンバーは、「**拡張構成**」ウィンドウで変更を行い、その変更内容を保存することができます。

詳しくは、1003 ページの『役割と権限』を参照してください。

ヒストリカル・レポート
WebSphere Applications エージェント、Response Time Monitoring エージェント、および Synthetic Playback エージェントによって収集されたデータのレポートを利用できます。Response Time Monitoring エージェント・レポートには、トランザクション・トラッキングが必要です (Cloud APM, Base にはありません)。 レポートの説明については、1124 ページの『レポート』を参照してください。

Agent Builder
カスタム・エージェントを作成して、任意のプラットフォームまたはテクノロジーをモニターします。https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/agent_builder_guide.htm を参照してください。

データベース・リソース・モニター
適応範囲はオファリングによって異なります。モニター可能なデータベースの名前については、54 ページの『機能』を参照してください。

インフラストラクチャー・リソース・モニター
リソース・モニター・エージェントを使用して、ハイパーバイザー、ストレージ、およびネットワークをモニターします。適応範囲はオファリングによって異なります。54 ページの『機能』を参照してください。

商業アプリケーションのリソース・モニター
リソース・モニター・エージェントを使用して、ビジネス・アプリケーションおよびコラボレーション・アプリケーションをモニターします。適応範囲はオファリングによって異なります。54 ページの『機能』を参照してください。

応答時間およびエンド・ユーザー・エクスペリエンスのモニター
インフラストラクチャーからデバイスに至るまでのユーザー・エクスペリエンスを表示します。ブラウザーからデータベースに至るまでの Web サイトおよび Web アプリケーションのパフォーマンスおよび可用性や、モバイル・デバイスをモニターするには、Response Time Monitoring を使用します。モニターする任意の Web サーバーに Response Time Monitoring エージェントをインストールすると、それ以上の構成をほとんど (またはまったく) 行わなくても、そのエージェントによって収集されたデータが Application Performance ダッシュボードに表示されます。Response Time Monitoring エージェントからのデータは、「**エンド・ユーザー・トランザクション**」ダッシュボードで使用されます。Cloud APM, Advanced でブラウザーからの応答時間を測定することができ、また Response Time Monitoring エージェントからのデータは「**集約トランザクション・トポロジー**」でも使用されます。詳しくは、91 ページの『シナリオ: IBM Java アプリケーション・スタック のモニター』を参照してください。

トランザクション・トラッキング
この機能は、Cloud APM, Advanced で使用できます。トランザクション・トラッキング機能により、トポロジー・ビューおよびインスタンス・レベルでのトランザクション・モニターが使用可能になります。トランザクション・トラッキングは Cloud APM サーバーの一部としてインストールされます。トランザクション・トラッキングは一部のエージェントでは自動的に有効になりますが、それ以外のエージェントでは手動で有効化する必要があります。75 ページの表 4 に、トランザクション・トラッキングをサポートするエージェントに関する詳細を示します。

表 4. エージェントおよびデータ・コレクターのトランザクション・トラッキングの有効化		
エージェント またはデータ・コレクター	**デフォルトで有効**	**有効にする方法**
DataPower エージェント	✓	249 ページの『DataPower エージェントのトランザクション・トラッキングの構成』
IBM Integration Bus エージェント	—	295 ページの『IBM Integration Bus エージェントのトランザクション・トラッキングの構成』 **注 :** このエージェントを Solaris X86 にデプロイする場合、TT はサポートされません。
J2SE データ・コレクター	✓	455 ページの『J2SE のモニターの構成』
JBoss エージェント	—	473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』
Liberty データ・コレクター	✓	881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』
Microsoft .NET エージェント	—	530 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
Node.js データ・コレクター	—	600 ページの『IBM Cloud アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ』 605 ページの『オンプレミス・アプリケーション用の Node.js データ・コレクターのカスタマイズ』
Response Time Monitoring エージェント + HTTP Server エージェント	—	688 ページの『インストールの計画』
SAP NetWeaver Java Stack エージェント	—	769 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
Tomcat エージェント	—	807 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
WebLogic エージェント	—	817 ページの『WebLogic モニターの構成』
WebSphere Applications エージェント	—	841 ページの『対話式でのデータ・コレクターの構成』 **注 :** このエージェントを Solaris X86 にデプロイする場合、TT はサポートされません。
WebSphere MQ エージェント	—	941 ページの『WebSphere MQ エージェントのトランザクション・トラッキングの構成』 **注 :** このエージェントを Solaris X86 にデプロイする場合、TT はサポートされません。

トランザクション・トラッキングをサポートするすべてのエージェントの**集約トランザクション・トポロジー**および**トランザクション・インスタンス・トポロジー**の両方のビューにデータが表示されます。

アプリケーション・トポロジー
アプリケーション環境で、すべてのコンポーネントがどのように接続されているかを表示します。詳しくは、1082 ページの『アプリケーション - Application Performance ダッシュボード』を参照してください。

トランザクション・インスタンス・トポロジー
トランザクションのインスタンスごとに、環境内をたどったパスを視覚化します。詳しくは、97 ページの『トランザクション・インスタンス・トポロジー』を参照してください。

Availability Monitoring
IBM Cloud Availability Monitoring は、世界の複数のポイント・オブ・プレゼンス (Point of Presence: PoP) から Web アプリケーションの拡張シンセティック・モニターを実現します。一定の間隔でユーザーの振る舞いを模倣するシンセティック・テストを作成します。パブリック・ポイント・オブ・プレゼンスからテストを実行するか、ローカル・サーバーまたはプライベート・サーバーに独自のカスタム・ポイント・オブ・プレゼンスをダウンロードしてデプロイします。Availability Monitoring ダッシュボードを使用して、アプリケーションの可用性、パフォーマンス、およびアラートをグラフ、明細表、およびマップ・ビューを使用してモニターします。ウォーターフォール式分析を使用して、パフォーマンスおよび可用性の問題がいつ発生したかを識別し、それらの問題の理由を特定します。

シンセティック・テストの使用について詳しくは、1047 ページの『Availability Monitoring』を参照してください。

詳細診断
特定のエージェントについては、要約ダッシュボードから詳細診断ダッシュボードにドリルダウンして、個々の要求に関する情報を表示できます。要約ダッシュボードからドリルダウンして、コード・レベル、スタック・トレース、および SQL 照会の詳細を表示します。診断ダッシュボードを使用して、問題がある要求を特定し、問題のトランザクションをデバッグします。まだ進行中のハングしたトランザクションや処理速度の遅いトランザクションを検出、診断、および強制終了することもできます (「WebSphere Applications エージェント Reference」を参照してください)。76 ページの表 5 に、診断エージェントに関する詳細情報を示します。

表 5. エージェントおよびデータ・コレクターの診断ダッシュボード

エージェントまたはデータ・コレクター	デフォルトで構成されている診断データ	使用できる診断ダッシュボード	診断ダッシュボードのアクセス方法	エージェントまたはデータ・コレクターが診断データを収集するように構成する方法
J2SE データ・コレクター	✓	詳細、Web モジュール、要求インスタンス、要求の要約、および要求トレース	「**概要**」ダッシュボードまたは「**Web モジュール**」ダッシュボードで「**診断**」をクリックします。	455 ページの『J2SE のモニターの構成』
JBoss エージェント	—	診断ダッシュボード、インフライト要求の要約、インフライト要求スタック・トレース・ダッシュボード、JVM ガーベッジ・コレクション、ヒープ・ダンプ、ヒープ・ダンプの比較	「**概要**」ダッシュボードで、「**診断**」、「**処理中要求 (Inflight Requests)**」、「**詳細**」、または「**ヒープ・ダンプ**」をクリックします。	473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』

表 5. エージェントおよびデータ・コレクターの診断ダッシュボード (続き)

エージェントまたはデータ・コレクター	デフォルトで構成されている診断データ	使用できる診断ダッシュボード	診断ダッシュボードのアクセス方法	エージェントまたはデータ・コレクターが診断データを収集するように構成する方法
Liberty データ・コレクター	✓	詳細、ヒープ・ダンプ、ヒープ・ダンプの比較、メモリー分析	「**概要**」ダッシュボードで、「**診断**」、「**ヒープ・ダンプの表示**」、または「**メモリー分析の表示**」をクリックします。	• 886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』 • 881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』
Microsoft .NET エージェント	–	要求インスタンス、要求の要約、および要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	530 ページの『configdc コマンドを使用した診断データの収集の有効化』
Node.js エージェント	✓	GC の詳細、要求インスタンス、要求の要約、要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	593 ページの『Node.js エージェントの構成』
Node.js データ・コレクター	✓	GC の詳細、最低速要求の詳細、要求インスタンス、要求トレース	「概要」ダッシュボードで、「**診断**」または「**GC の詳細**」をクリックします。	• 598 ページの『IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』 • 603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』

表 5. エージェントおよびデータ・コレクターの診断ダッシュボード (続き)

エージェントまたはデータ・コレクター	デフォルトで構成されている診断データ	使用できる診断ダッシュボード	診断ダッシュボードのアクセス方法	エージェントまたはデータ・コレクターが診断データを収集するように構成する方法
Python データ・コレクター	✓	最低速要求の詳細、要求インスタンスの詳細、要求トレースの詳細、Python スレッド詳細、Python ガーベッジ・コレクション、Python ヒープ詳細	「**概要**」ダッシュボードで、「**診断**」、「**スレッドの詳細**」、または「**メモリーの詳細**」をクリックします。	• 673 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成』 • 678 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成』
Ruby エージェント	–	要求の要約/詳細、サンプル要求インスタンス、および要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	718 ページの『Ruby モニターの構成』
Ruby データ・コレクター	✓	要求インスタンス、要求の要約、および要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	727 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクターの構成』
SAP NetWeaver Java Stack エージェント	✓	要求インスタンス、要求の要約、および要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	769 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
Tomcat エージェント	–	要求インスタンス、要求の要約、および要求トレース	「**概要**」ダッシュボードで、「**診断**」をクリックします。	807 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
WebLogic エージェント	–	診断ダッシュボード、インフライト要求の要約、インフライト要求スタック・トレース・ダッシュボード、JVM GC の詳細、ヒープ・ダンプ、ヒープ・ダンプの比較	「**概要**」ダッシュボードで、「**診断**」、「**要求の表示**」、「**詳細**」、または「**ヒープ・ダンプ**」をクリックします。	817 ページの『WebLogic モニターの構成』

表 5. エージェントおよびデータ・コレクターの診断ダッシュボード (続き)				
エージェントまたはデータ・コレクター	**デフォルトで構成されている診断データ**	**使用できる診断ダッシュボード**	**診断ダッシュボードのアクセス方法**	**エージェントまたはデータ・コレクターが診断データを収集するように構成する方法**
WebSphere Applications エージェント	—	診断、要求インスタンス、要求シーケンス、インフライト要求の要約、インフライト要求スタック・トレース、ヒープ・ダンプ、ヒープ・ダンプの比較、メモリー分析	「**概要**」ダッシュボードで、「**診断**」、「**要求の表示**」、「**ヒープ・ダンプの表示 (View Heap Dump)**」、または「**メモリー分析の表示 (View Memory Analysis)**」をクリックします。 「**メモリー分析の表示 (View Memory Analysis)**」ボタンが機能するのは、メモリー・リーク・モニターが有効になった後のみです。	• 838 ページの『簡易構成ユーティリティーを使用したデータ・コレクターの構成』 • 878 ページの『メモリー・リーク・モニターの有効化』

「**診断**」ボタンが有効になるのは、エージェントに対して詳細エージェントが構成されていて、ユーザーが役割管理者、モニター管理者、または診断ダッシュボードに対する表示権限があるその他のカスタムの役割のメンバーである場合のみです。

しきい値
しきい値を使用すると、アクティブにモニターする 定義に基づいて、アプリケーションの特定の動作や条件を検出できます。事前定義しきい値はエージェントごとに使用でき、モニター用の新しいしきい値を定義できます。詳しくは、987 ページの『しきい値マネージャー』を参照してください。

イベント転送を構成した場合は、イベントが EIF 受信側に送信されます。しきい値と、イベント・サーバーに転送されるイベントとの間のデフォルト・マッピングを使用するか、しきい値をマップする方法をカスタマイズすることができます。詳しくは、992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』を参照してください。

「**Application Performance ダッシュボード**」でアプリケーションを選択すると、「**イベント**」タブが表示されます。「**イベント**」タブには、現在のアプリケーションに対してオープンされているイベントが表示されます。パフォーマンス・メトリックを示す詳細なダッシュボードにドリルダウンして、イベントの原因を判別するのに役立てることができます。詳しくは、1109 ページの『イベント状況』を参照してください。

リソース・グループ
モニター対象エンタープライズの 管理対象システムを目的に応じて分類できます。そのような管理対象システムには通常、同じしきい値要件が存在します。リソース・グループ・マネージャーを使用してモニター対象システムをグループに 編成し、イベントしきい値を割り当てることができます。詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。

「はじめに」ページ
Cloud APM コンソール にログインすると、「はじめに」ページが表示されます。シナリオ・ベースのツアーまたはビデオ・デモンストレーションへのリンクを使用するには、「**ユーザー・タスク**」または「**管理者タスク**」のいずれかをクリックします。「今すぐ開始」リンクをクリックすると、「しきい値マネー

ジャー」などの機能が直接表示されます。「**コミュニティー・リソース**」リンクは 「**よくある質問**」や Cloud APM フォーラムなどに移動します。

他の製品やコンポーネントとの統合により、追加の機能を使用できます。詳しくは、80 ページの『統合』を参照してください。さらに詳しい説明は、949 ページの『第 8 章 他の製品およびコンポーネントとの統合』を参照してください。

統合

以下のような他の製品やコンポーネントとの統合により、追加の機能を使用できます。 Tivoli Monitoring、OMEGAMON、Netcool/OMNIbus、Operations Analytics - Log Analysis、Operations Analytics - Predictive Insights、Alert Notification、Control Desk、 IBM Cloud、および Agent Builder。

IBM Tivoli Monitoring

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Cloud Application Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェント・タイプをインストールすることはできません。詳しくは、950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。

IBM Tivoli Monitoring 製品と Cloud APM 製品 (クラウドまたはオンプレミス、あるいはオンプレミスとクラウド) が両方とも存在する環境では、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールして、両方の環境の管理対象システムの統合ビューを利用することができます。詳しくは、953 ページの『ハイブリッド・ゲートウェイ』を参照してください。サポートされる Tivoli Monitoring エージェントのリストについては、Hybrid Gateway supported agents (APM Developer Center)を参照してください。

IBM OMEGAMON

z Systems Extension Pack は、z Systems メインフレームで実行されている 1 つ以上の OMEGAMON エージェントを Cloud APM に接続します。z Systems Extension Pack と ハイブリッド・ゲートウェイを使用して、デプロイされている OMEGAMON エージェントを Cloud APM に接続することで、OMEGAMON アプリケーション・コンポーネントのモニター・データとイベントを Cloud APM コンソールに表示できます。

詳しくは、965 ページの『OMEGAMON との統合』を参照してください。

IBM Netcool/OMNIbus

Cloud APM からオンプレミスの Netcool/OMNIbus イベント・マネージャーにイベントを転送することができます。詳しくは、966 ページの『Netcool/OMNIbus との統合』を参照してください。

IBM Operations Analytics - Log Analysis

ご使用の環境に IBM Operations Analytics - Log Analysis が含まれている場合、アプリケーションのログ・データとパフォーマンス・データをまとめて、アプリケーションが遭遇している問題の根本原因を調べることができます。 アプリケーションに関連するログ・データを検索して、問題 (処理速度が遅い、障害が発生するなど) の原因を調べることができます。詳しくは、970 ページの『Operations Analytics - Log Analysis との統合』を参照してください。

IBM Operations Analytics - Predictive Insights

Operations Analytics - Predictive Insights は、データを分析して、システムの正常な動作を学習します。また、パフォーマンス・モデルを作成し、それを使用することでモデルの範囲外の動作を検出または予測して、異常な動作が発生したときにアラームを生成します。ご使用の Cloud APM サブスクリプションに Operations Analytics - Predictive Insights を追加できます。 これで、Application Performance ダッシュボードで異常を表示し、Operations Analytics - Predictive Insights ユーザー・インターフェースにドリルダウンして 、さらなる詳細を表示できます。詳しくは、971 ページの『Operations Analytics - Predictive Insights との統合』を参照してください。

IBM Cloud

スタンドアロンのデータ・コレクターを使用して、IBM Cloud 環境内のアプリケーションのモニター情報を表示できます。データ・コレクターは、ご使用の IBM Cloud アプリケーションに関するリソース・データと詳細診断モニター・データを Cloud APM サーバーに転送することによって、モニター機能を IBM Cloud と統合できるようにします。Cloud APM サーバーは、データ・コレクターによって収集され

たモニター情報を受信し、処理します。モニターできる IBM Cloud アプリケーションのタイプは、以下のとおりです。

- Liberty アプリケーション
- Node.js アプリケーション
- Python アプリケーション
- Ruby アプリケーション

データ・コレクターを構成した後、Cloud APM コンソールでモニター・データを表示できます。詳しくは、189 ページの『データ・コレクターの一般的な構成手順』を参照してください。

IBM Alert Notification
IBM Cloud Application Performance Management を使用している場合、IBM Alert Notification は、自動的に統合されます。 Alert Notification は、IT 運用環境で問題が発生した場合にただちに IT スタッフにアラートを通知する、使いやすいシンプルな通知システムです。アラートのソースは、エージェントから受信したデータによって提供されます。Alert Notification をアクティブ化した後に、Cloud APM インスタンスに接続します。スタンドアロン・パッケージとして、REST API を実装して開始することができる任意のオンプレミス・モニター・ツールに Alert Notification を統合することができます。サポートされるツールには、IBM Tivoli Netcool/OMNIbus があります。詳しくは、『Alert Notification との統合』を参照してください。

IBM Control Desk
IBM Control Desk との統合は、サポート・チケットを IBM サポートに送信することによって利用可能になります。Cloud APM イベントが IBM Control Desk で自動的にチケットをオープンするように 構成することができます。IBM サポートにアクセスして「**サブスクリプション**」を選択します。この統合の構成を可能にするためにサポートが必要とする詳細について詳しくは、973 ページの『Control Desk との統合』を参照してください。

IBM Agent Builder
Agent Builder を使用して任意のテクノロジー用のカスタム・エージェントを作成できます。 詳しくは、https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/agent_builder_guide.htm を参照してください。

資料

IBM Cloud Application Performance Management に関する情報は、IBM Knowledge Center、Cloud APM コンソール、および Application Performance Management Developer Center にあります。

IBM Knowledge Center
IBM Knowledge Center 内の Cloud APM は、製品の技術情報の公式な情報源です。

ユーザー・インターフェースのヘルプ
Cloud APM コンソールにログインしているか、ガイド付きデモを閲覧しているときは、次の方法でヘルプ・システムにアクセスできます。

- ナビゲーション・バーの ⓘ「ヘルプ」メニューにある「**ヘルプ・コンテンツ**」をクリックします。
- Application Performance ダッシュボードのバナーの ⓘ をクリック。
- システム構成ページの「**詳細情報**」リンクをクリック。
- ダッシュボード・ウィジェットの ⓘ をクリック。

IBM Application Performance Management Developer Center
Application Performance Management Developer Center には、APM の広範囲のユーザーに適用される APM コンテンツが 1 カ所にまとめられています。資料、ブログ、ビデオ、追加リソースへのリンクなどのコンテンツがあります。

IBM Cloud Application Performance Management フォーラムおよび dwAnswers
Cloud Application Performance Management フォーラムおよび dwAnswers では、問題のトラブルシューティングや解決方法など、製品の問題についての技術的なディスカッションが行われます。

以下の Web サイトからも情報を入手できます。

Software Product Compatibility Reports (SPCR) ツール
SPCR ツールを使用して、オファリングおよびコンポーネント要件に関連する各種レポートを生成できます。いずれかの Cloud Application Performance Management オファリング名または IBM Cloud Application Performance Management - Agents を検索します。

IBM Marketplace
ビデオ・デモンストレーションや FAQ などのリソースを、IBM Marketplace で利用できます。

IBM API Explorer
Cloud APM API の資料については、1073 ページの『API の解説』を参照してください。

IBM 用語
IBM Terminology Web サイトには、IBM 製品に関する用語集が、アクセスしやすいように 1 カ所にまとめてあります。

IBM Redbooks®
IBM Redbooks Web サイトで提供される Redbooks 資料、Redpaper、および Redbooks 技術情報には、プラットフォームおよびソリューションの 視点からの製品情報を記載しています。

資料の表記上の規則

この資料では、特殊な用語、アクション、コマンド、ご使用のオペレーティング・システムに依存するパス、およびプラットフォーム固有の情報と製品固有の情報に関して、いくつかの規則を使用します。

書体規則

本書では、以下のような書体の規則が適用されています。

太字

- 太字にしなければ周囲のテキストと区別するのが難しい小文字のコマンド、大/小文字混合コマンド、パラメーター、および環境変数
- インターフェース・コントロール (チェック・ボックス、プッシュボタン、ラジオ・ボタン、スピン・ボタン、フィールド、フォルダー、アイコン、リスト・ボックス、リスト・ボックス内の項目、マルチカラム・リスト、コンテナー、メニュー選択、メニュー名、タブ、プロパティー・シート)、ラベル (**ヒント**など)
- テキスト内のキーワードおよびパラメーター

イタリック

- 引用 (資料、ディスケット、および CD の名称など)
- 本文内で定義されている単語および句 (例: 非交換回線のことを *Point-to-Point* 回線 と呼びます。)
- 強調する単語および文字 (例: LUN アドレスは英字 *L* から始まる必要があります。)
- テキスト中の新規用語 (定義リスト内を除きます) (例: ビュー は、データが入っているワークスペース内のフレームです。)
- ユーザーが指定する必要がある変数および値 (例: ここで、*myname* は ... を表します。)

モノスペース

- 例およびコード例
- ファイル名、ディレクトリー名、パス名、プログラミング・キーワード、プロパティー、および周囲のテキストと区別するのが難しいその他の要素
- メッセージ・テキストおよびプロンプト
- 入力する必要のあるテキスト
- 引数またはコマンド・オプションの値

太字モノスペース

- コマンド名、およびコマンドとして入力できるマクロおよびユーティリティーの 名前
- テキスト中の環境変数名
- キーワード

- テキスト中のパラメーター名: API 構造パラメーター、コマンドのパラメーターと引数、および構成パラメーター
- プロセス名
- テキスト中のレジストリー変数名
- スクリプト名

オペレーティング・システムごとに異なる変数およびパス

ディレクトリー・パスのスラッシュの方向は、資料内で異なる場合があります。資料内での表記に関わらず、以下のガイドラインに従ってください。

- Linux AIX スラッシュ (**/**) を使用します。
- Windows 円記号 (**¥**) を使用します。

環境変数の名前は、Windows と AIX で必ずしも同じになるとは限りません。例えば、Windows での %TEMP% は、AIX または Linux での $TMPDIR に相当します。

環境変数については、次のガイドラインに従ってください。

- Linux AIX **$***variable* を使用します。
- Windows **%***variable***%** を使用します。

Windows Windows システムで bash シェルを使用する場合は、AIX の規則を使用できます。

エージェントのインストール・ディレクトリーの変数およびパス

install_dir はエージェントのインストール・ディレクトリーです。デフォルトの場所は、次のようにオペレーティング・システムによって異なります。

- Windows `C:¥IBM¥APM`
- Linux `/opt/ibm/apm/agent`
- AIX `/opt/ibm/apm/agent`

第 4 章 デプロイメントの計画

IBM Cloud Application Performance Management のデプロイメントを確実に正常に行うためには、計画が重要です。

システム要件

IBM Cloud Application Performance Management のエージェントおよびデータ・コレクターでは、さまざまなオペレーティング・システムがサポートされており、各コンポーネントに固有の要件があります。

タイム・ゾーン

管理対象システムで Network Time Protocol (NTP) を使用して、時刻が正確になるようにします。サーバーの物理的な場所に合う時刻 (例えば、ブラジリアであれば UTC-03:00、ヤンゴンであれば UTC +06:30) を設定すると、イベントおよびトランザクションの正確なタイム・スタンプを得られます。エージェントは、ユーザーの現地時間を使用して、APM UI にデータをレポートします。

Cloud APM のエージェントおよびデータ・コレクターの要件

インストール予定の各モニター・エージェントおよびスタンドアロンのデータ・コレクターの要件に関する情報を入手してください。

Cloud APM エージェントおよびデータ・コレクターは、一般にハイパーバイザー・トランスペアレントです。つまり、オペレーティング・システムがホストされているハイパーバイザー (Hyper-V、IBM PowerVM、KVM、VMWare ESX など) に関係なく、サポートされるすべてのオペレーティング・システムにインストールしてデプロイすることができます。

エージェントおよびデータ・コレクターの要件については、System requirements (APM Developer Center) の「Component reports」セクション内のリンクを使用してください。

サポートされるブラウザーに関する最新情報については、IBM Cloud Application Performance Management の詳細なシステム要件のレポートを参照してください。

Software Product Compatibility Reports ツールで IBM Cloud Application Performance Management を検索することもできます。

エージェントがグローバル化データを Cloud APM サーバーに送信する場合、エージェントがインストールされているローカル・コンピューター・システムで UTF-8 エンコードがサポートされている必要があります。

エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート

Cloud APM コンポーネントと、モニターされているアプリケーションやシステム (ローカルまたはリモート) との間の通信には、さまざまなポートが使用されます。ほとんどの場合、構成を容易にするためにデフォルト・ポートが用意されています。大部分のデフォルトは、構成パラメーターを使用してカスタマイズできます。

86 ページの表 6 には、Cloud APM エージェントとデータ・コレクターが、モニターしているアプリケーションやシステムと通信するために使用するデフォルト・ポートがリストされています。表内の N/A は、以下のいずれかのシチュエーションを示します。

- エージェントおよびデータ・コレクターが、モニター対象のアプリケーションおよびシステムとの 通信にいずれのポートも使用していない。
- 通信に使用されるポートが、モニター対象アプリケーションの構成によって決定されている。
- エージェントまたはデータ・コレクターによって使用されるポートが動的に割り当てられており、静的デフォルトが提供されない。
- 使用されるすべてのポートをユーザーが指定する必要があり、デフォルトが提供されない。

© Copyright IBM Corp. 2014, 2019

表 6. エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート

エージェントおよびデータ・コレクター	デフォルト・ポート	構成可能	ローカル	リモート
Amazon EC2 エージェント	• TCP ポート 80 (HTTP 用) • TCP ポート 443 (HTTPS 用)	N/A	はい	いいえ
Amazon ELB エージェント	• TCP ポート 80 (HTTP 用) • TCP ポート 443 (HTTPS 用)	N/A	いいえ	いいえ
Azure Compute エージェント	• TCP ポート 80 (HTTP 用) • TCP ポート 443 (HTTPS 用)	いいえ	いいえ	いいえ
Cassandra エージェント	7199 (JMX サーバー用。ローカルおよびリモート)	はい	はい	はい
Cisco UCS エージェント	• TCP ポート 80 (HTTP 用) • TCP ポート 443 (HTTPS 用)	いいえ	はい	いいえ
Citrix VDI エージェント	PowerShell 呼び出し用: • 5985 (HTTP 用) • 5986 (HTTPS 用)	はい	はい	はい
Db2 エージェント	• 50000 (Db2 サーバーのデフォルト・ポート) • サポートされるリモート・モニター: リモート・サーバー・インスタンスのカタログ中にユーザーが指定したポート番号を使用します。	はい	はい	はい
DataPower エージェント	5550 (リモート DataPower アプライアンスへの接続用)	はい	いいえ	はい
Hadoop エージェント	• ローカル・モニター: **CP_PORT** 環境変数値 • リモート・モニター: – 50070 (スタンバイ Namenode) – 50090 (2 次 Namenode) – 8088 (ResourceManager) – 19888 (JobHistory Server) – 8080 (Ambari)	はい	はい	はい
HMC Base エージェント	12443 (HMC からの SDK のダウンロード用)	いいえ	はい	いいえ
HTTP Server エージェント	HTTP サーバーは別のポートに構成されている可能性があります。ただし、エージェント自体にはデフォルト・ポートはありません。	N/A	はい	いいえ
IBM Cloud エージェント	api.softlayer.com ポート 443 への発信接続。	N/A	いいえ	はい

表 6. エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート (続き)

エージェントおよびデータ・コレクター	デフォルト・ポート	構成可能	ローカル	リモート
IBM Integration Bus エージェント	N/A	N/A	はい	いいえ
Internet Service Monitoring	データ・ブリッジ用: • 9510 • 9520	はい	いいえ	はい
DataStage エージェント	• 9443 (WAS HTTPS ポート) • 50000 (データベース JDBC ポート) • 1433 (Microsoft SQL) • 1521 (Oracle)	はい	はい	はい
J2SE データ・コレクター	N/A	N/A	いいえ	いいえ
JBoss エージェント	JBoss サーバーのバージョンによって異なります。 • 9990 • 9994 • 9999	いいえ	はい	いいえ
Liberty データ・コレクター	N/A	N/A	いいえ	いいえ
Linux KVM エージェント	• 8080 (HTTP 用) • 8443 (HTTPS 用)	はい	はい	いいえ
Linux OS エージェント	22 (SSH を使用したリモート・ログ・モニター用)	はい	はい	いいえ
MariaDB エージェント	3306	はい	はい	はい
Microsoft Active Directory エージェント	ポート番号は、使用状況をモニターするためのリスナー設定によって異なります。	N/A	はい	はい
Microsoft Cluster Server エージェント	N/A	N/A	いいえ	いいえ
Microsoft Exchange Server エージェント	N/A	N/A	いいえ	いいえ
Microsoft Hyper-V Server エージェント	N/A	N/A	いいえ	いいえ
Microsoft IIS エージェント	N/A	N/A	いいえ	いいえ
Microsoft .NET エージェント	トランザクション・トラッキング・データの送信には、デフォルトでポート 5456 が使用されます。	はい	はい	いいえ
Microsoft Office 365 エージェント	7799 (Skype シンセティック・トランザクション用)	はい	はい	いいえ
Microsoft SharePoint Server エージェント	1433 (SQL サーバー用)	いいえ	はい	はい

表 6. エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート (続き)

エージェントおよびデータ・コレクター	デフォルト・ポート	構成可能	ローカル	リモート
Microsoft SQL Server エージェント	1433 (SQL サーバーのデフォルト)	はい (**COLL_PORT** による)	はい	いいえ
MQ Appliance エージェント	• 162 (SNMP イベントの受信用) • 5554 (MQ Appliances への接続用)	はい	はい	はい
MongoDB エージェント	• 27017 (単一インスタンス用) • 27019 (クラスター用)	はい	はい	いいえ
MySQL エージェント	3306 (JDBC 接続用)	はい	はい	はい
NetApp Storage エージェント	リモート接続用: • 8088 • 8488 • 443 • 8443	いいえ	いいえ	はい
Node.js エージェント	63336	はい	はい	いいえ
Node.js データ・コレクター	N/A	N/A	いいえ	いいえ
OpenStack エージェント	5000 (OpenStack ID サービス用)	はい	いいえ	はい
Oracle Database エージェント	1521 (SQL 接続用)	はい	はい	いいえ
PHP エージェント	• Apache 接続 • ポート番号は Apache の構成に基づきます。	はい	はい	いいえ
PostgreSQL エージェント	5432 (JDBC 接続用)	はい	はい	はい
Python データ・コレクター	N/A	N/A	いいえ	いいえ
RabbitMQ エージェント	RabbitMQ 管理プラグインが有効になっているポート番号 (ローカルおよびリモート): 15672	はい	はい	はい
Response Time Monitoring エージェント	• パッケージ・アナライザー・モデルは、ポート 80 で HTTP トランザクションをモニターします。 • HTTP サーバー・モデルは、すべてのポートをモニターします。	はい	はい	いいえ
Ruby エージェント	動的に生成	N/A	はい	いいえ
Ruby データ・コレクター	N/A	N/A	いいえ	いいえ
SAP エージェント	33*nn* (*nn* は SAP インスタンス番号)	いいえ	はい	いいえ
SAP HANA データベース・エージェント	デフォルト: 30013。範囲: 30013 から 39913。	はい	はい	いいえ

表 6. エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート (続き)

エージェントおよびデータ・コレクター	デフォルト・ポート	構成可能	ローカル	リモート
SAP NetWeaver Java Stack エージェント	デフォルト: 50004。範囲: 50004 から 59904。	はい	はい	いいえ
Siebel エージェント	N/A	N/A	はい	いいえ
Skype for Business Server エージェント (以前の Microsoft Lync Server エージェント)	• Business server のデフォルト・ポート 5061 • SQL Server のポート 1433 (環境に応じてローカルまたはリモート)	いいえ	はい	いいえ
Sterling Connect Direct エージェント	1363	はい	いいえ	はい
Sterling File Gateway エージェント	50000 IBM B2B Integrator REST API のポート番号およびデータベース・ポート番号は両方とも必須であり、構成可能です。	はい	はい	はい
Sybase エージェント	5000	N/A	はい	いいえ
Synthetic Playback エージェント	• 4444 (内部 Selenium サーバーの接続用) • リモート・ポートはモニター対象の Web サイトの HTTP URL で指定されます (通常は HTTP 80 および HTTPS 443)。	いいえ	はい	いいえ
Tomcat エージェント	8686 (Tomcat MBean サーバー用)	はい (JMX ポートによる)	はい	いいえ
UNIX OS エージェント	22 (SSH を使用したリモート・ログ・モニター用)	はい	はい	いいえ
VMware VI エージェント	• 443 (リモート・モニター用) • 80 (ローカル・モニター用)	いいえ	はい	はい
WebLogic エージェント	7003 (JMX 管理 HTTP トラフィック)	はい	はい	いいえ
WebSphere Applications エージェント	• 63335 (V8 モニター・エージェント用) • 63336 (V6 モニター・エージェント用) • 63355 (リソース・モニター用) • 5457 (Transaction Framework Extension 用)	はい	はい	いいえ
WebSphere Infrastructure Manager エージェント	N/A	N/A	はい	いいえ
WebSphere MQ エージェント	ポート番号は、使用状況をモニターするためのリスナー設定によって異なります。	N/A	いいえ	はい

表 6. エージェントおよびデータ・コレクターによって使用されるデフォルト・ポート (続き)				
エージェントおよびデータ・コレクター	**デフォルト・ポート**	**構成可能**	**ローカル**	**リモート**
Windows OS エージェント	22 (SSH を使用したリモート・ログ・モニター用)	はい	はい	いいえ

シナリオ

ご使用の環境の複雑さに応じて、モニター対象のさまざまなコンポーネントに対応するエージェントをインストールする必要があります。これらのデプロイメント・シナリオを参照して、IBM Cloud Application Performance Management から最良の結果を得るためには何をどこにインストールするべきかを理解してください。

シナリオ: IBM API Connect のモニター

APM エージェントおよびデータ・コレクターを使用して、IBM API Connect 環境をモニターおよびトラブルシューティングできます。

Cloud APM 製品は、API Connect 環境のパフォーマンスおよび可用性の管理を支援します。Cloud APM エージェントおよびデータ・コレクターを使用すると、API Connect インフラストラクチャーとアプリケーション API の両方を表示および制御して、パフォーマンスを最適化し、リソースの使用を効率化することができます。API Connect 環境内でパフォーマンスの問題が発生した場合、Cloud APM 製品は、問題の検出、診断、および切り分けを支援できます。

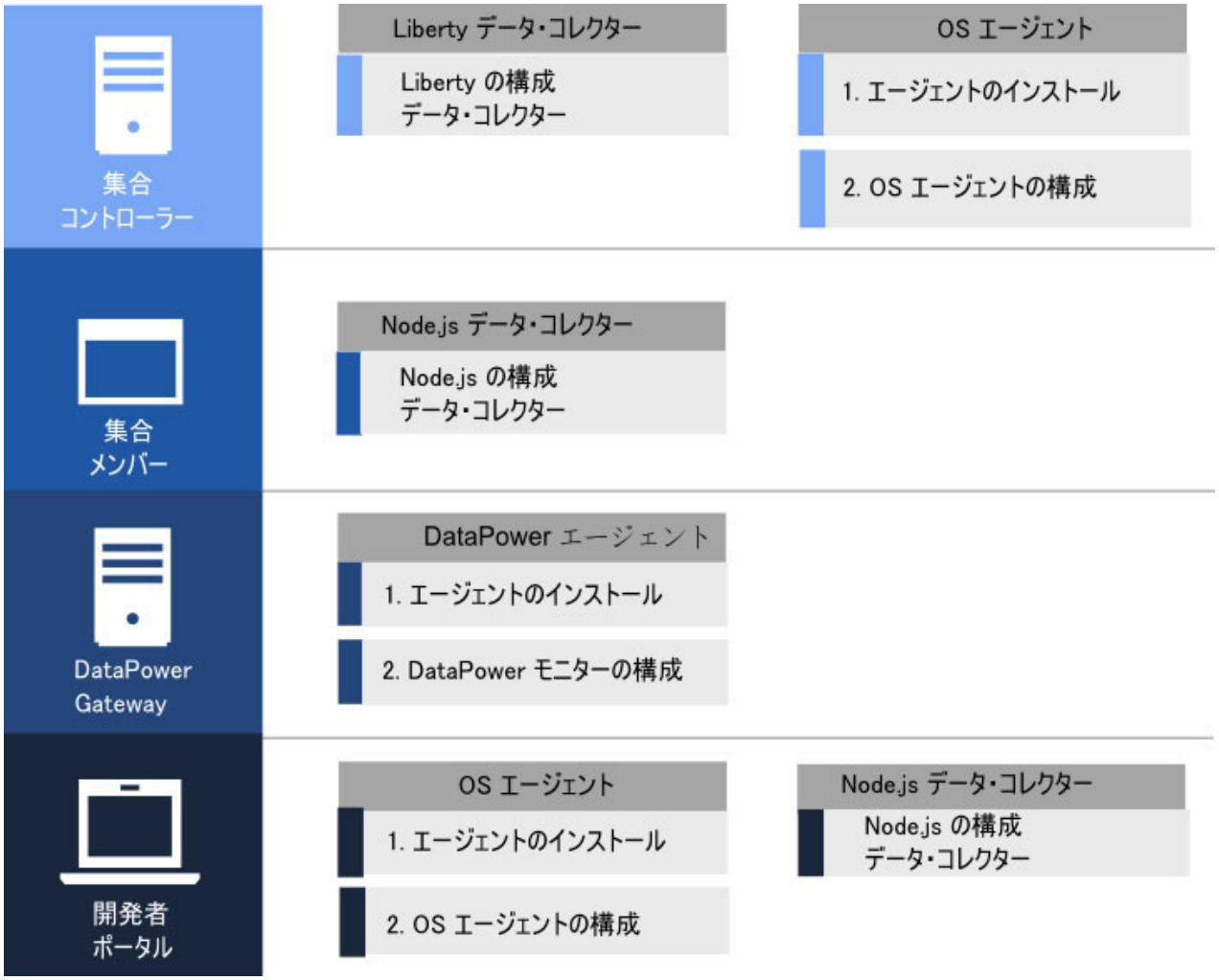
例えば、すべての利用可能なシステムに OS エージェントをインストールできます。OS エージェントを使用すると、CPU パフォーマンス、ディスクの入出力と使用状況、プロセスの可用性とパフォーマンス、ネットワークのパフォーマンスなど、サーバー固有のパフォーマンスを収集し、分析できます。また、主要な API Connect ログおよびシステム・ログをモニターするように OS エージェントを構成できます。

他のミドルウェア製品がデプロイされている場合、Cloud APM サーバーの一部としてインストールされるトランザクション・トラッキング機能により、トポロジー・ビューが提供されます。このトポロジー・ビューで、ミドルウェア製品とそれらの製品が公開するサービスのトランザクション・トラッキング情報を確認することや、問題が発生した場合にトラブルシューティングを 行うことができます。

以下の図は、API Connect コンポーネントと、各コンポーネントに対応してそのコンポーネントをモニターできる Cloud APM エージェントおよびデータ・コレクターを示しています。これらのエージェントおよびデータ・コレクターを有効にするには、エージェントおよびデータ・コレクターの名前の下にリストされているインストール・タスクと構成タスクを実行します。インストール・タスクまたは構成タスクに進むには、図の中でタスク名が表示されている長方形のボックスをクリックします。

注 :

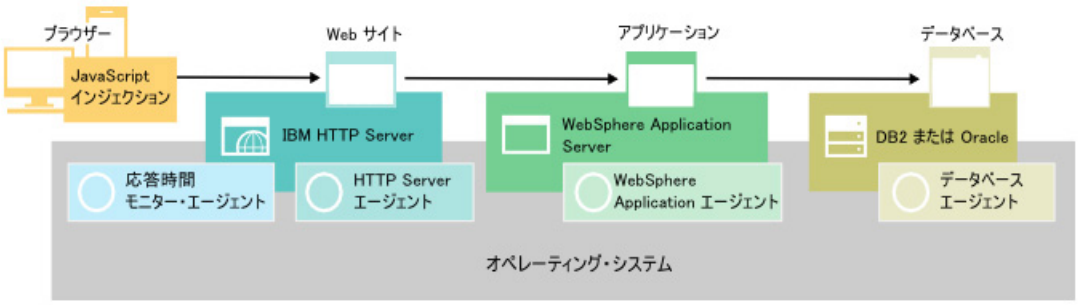
- 集合メンバーの公開済みの各 IBM API Connect アプリケーションに対して Node.js データ・コレクターをインストールします。
- DataPower Gateway をモニターする場合、DataPower エージェントが DataPower アプライアンスからリモートで実行されます。



1. 881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』
2. 123 ページの『第 6 章 エージェントのインストール』
3. 633 ページの『OS のモニターの構成』
4. 603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』
5. 123 ページの『第 6 章 エージェントのインストール』
6. 633 ページの『OS のモニターの構成』
7. 123 ページの『第 6 章 エージェントのインストール』
8. 237 ページの『DataPower モニターの構成』
9. 123 ページの『第 6 章 エージェントのインストール』
10. 633 ページの『OS のモニターの構成』
11. 603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』

シナリオ: IBM Java アプリケーション・スタック のモニター

IBM Java アプリケーション・スタック のモニターとトラブルシューティングを 行って、個々のコンポーネントからのリソース・モニターなど、ブラウザーからデータベースへのトランザクション・モニター情報を把握できます。IBM Java アプリケーション・スタックには、IBM HTTP Server、WebSphere Application Server、および IBM Db2 データベースまたは Oracle データベースが含まれます。



IBM Java アプリケーション・スタックのモニター
IBM Java アプリケーション・スタックをモニターするには、各コンポーネントにリストされているエージェントを記載されている順にインストールします。

オプションとしてシステムもモニターする 場合は、すべてのコンポーネントに OS エージェントをインストールします。

Web サーバーの場合は、以下のステップを実行します。

1. HTTP Server エージェント をインストールします。

   ファースト・パス : このインストールによって IBM HTTP Server 応答時間モジュールもインストールされ、JavaScript インジェクションが自動的に構成されます。
2. HTTP Server エージェントのインストール済み環境を構成します。
3. Response Time Monitoring エージェントをインストールします。

アプリケーション・サーバーの場合は、WebSphere Applications エージェント をインストールします。

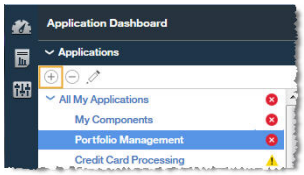
データベースの場合は、ご使用のデータベースに応じて Oracle データベース・エージェントまたは Db2 エージェントをインストールします。

Application Performance ダッシュボードへの Web アプリケーションの追加
モニターしたい Web アプリケーションを Application Performance ダッシュボードに追加します。

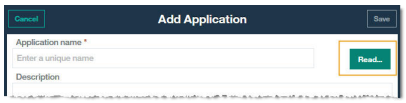
手順

Web アプリケーションを追加するには、以下のステップを実行します。

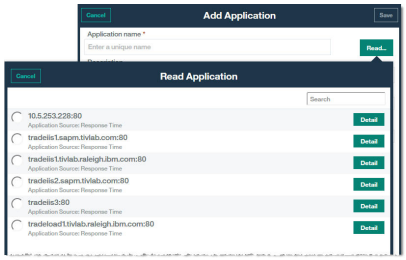
1. Application Performance ダッシュボードで、「**アプリケーションの追加**」をクリックします。



2. 「**読み取り**」をクリックして、検出されたアプリケーションのリストを開きます。



3. モニターしたい Web アプリケーションを選択します。



4. 「**保存**」をクリックします。

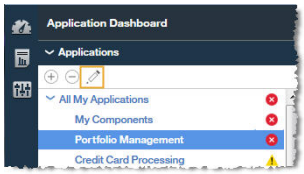
IBM Java アプリケーション・スタックと Web アプリケーションとの関連付け

WebSphere Application Server と、それを使用してモニターするデータベース・コンポーネントを関連付けるには、Web アプリケーションを編集します。

手順

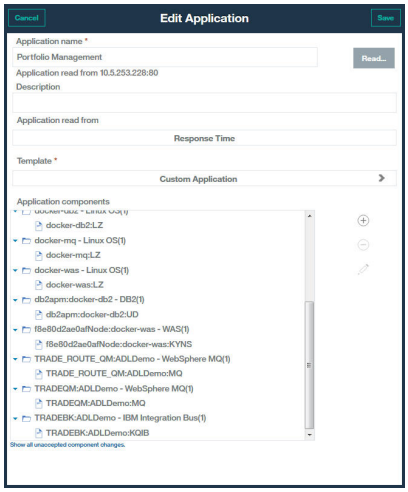
Java アプリケーション・スタック内のコンポーネントを表示するには、Application Performance ダッシュボードで以下のステップを実行します。

1. Web サーバーを選択して「**アプリケーションの編集**」をクリックします。



2. 「**アプリケーションの編集**」ウィンドウで、「**コンポーネントの追加**」 + をクリックします。
3. 「**コンポーネントの選択**」ウィンドウで WebSphere Application Server を選択します。
4. 「**コンポーネント・エディター**」で、必要なコンポーネント・インスタンスを選択し、「**追加**」をクリックします。

   検出された WebSphere Application Server インスタンスはこのリストに自動的に追加されます。
5. 「**戻る**」をクリックし、ご使用のデータベースに対してステップ 93 ページの『3』 から 93 ページの『4』 を繰り返します。WebSphere Application Server とデータベース・インスタンスの追加を、Java アプリケーション・スタック が完全なものになるまで続行します。



6. 「**閉じる**」をクリックしてから「**保存**」をクリックして、Application Performance ダッシュボードに戻ります。

タスクの結果

ヒント: 集約トランザクション・トポロジーに予期したトポロジーが最初に表示されない場合は、最新表示されるのを待ってから数分後に再度確認してください。それでもトポロジーが予期したものと異なる場合は、アプリケーションが所定のコンポーネントと通信していない可能性があります。ご使用の環境を確認してください。

IBM Java アプリケーション・スタックのモニター結果の表示

IBM Java アプリケーション・スタックのモニター結果をトポロジーに表示できます。

このタスクについて

トポロジーには、個々のコンポーネントからのリソース・モニターなど、ブラウザーからデータベースへのトランザクション・モニター情報が表示されます。集約トランザクション・トポロジーおよびトランザクション・インスタンス・トポロジーには以下のノードが表示されます。

- ブラウザー。JavaScript インジェクションが有効になっている場合のみ表示されます。
- HTTP サーバー
- WebSphere Application Server
- データベース

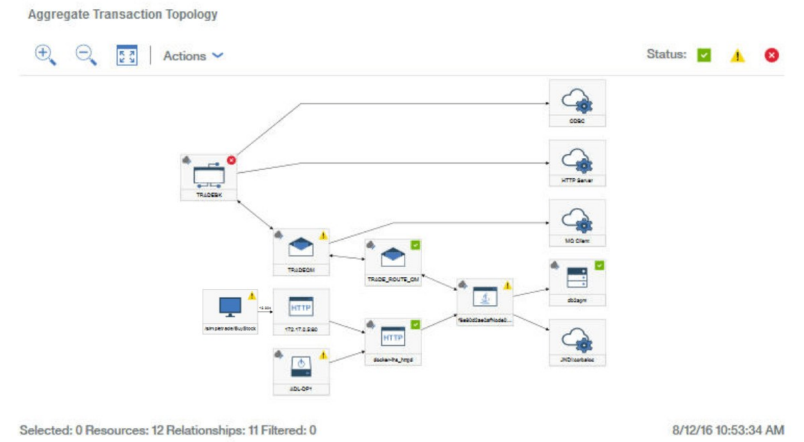
手順

トポロジー内のノードから、そのノードに関する詳細に以下のようにしてリンクできます。

1. ノードにマウス・ポインターを合わせると、追加情報が表示されたウィンドウが表示されます。
2. ノードについてより詳細なダッシュボードにドリルダウンするには、ノードを右クリックし、リンクを選択します。

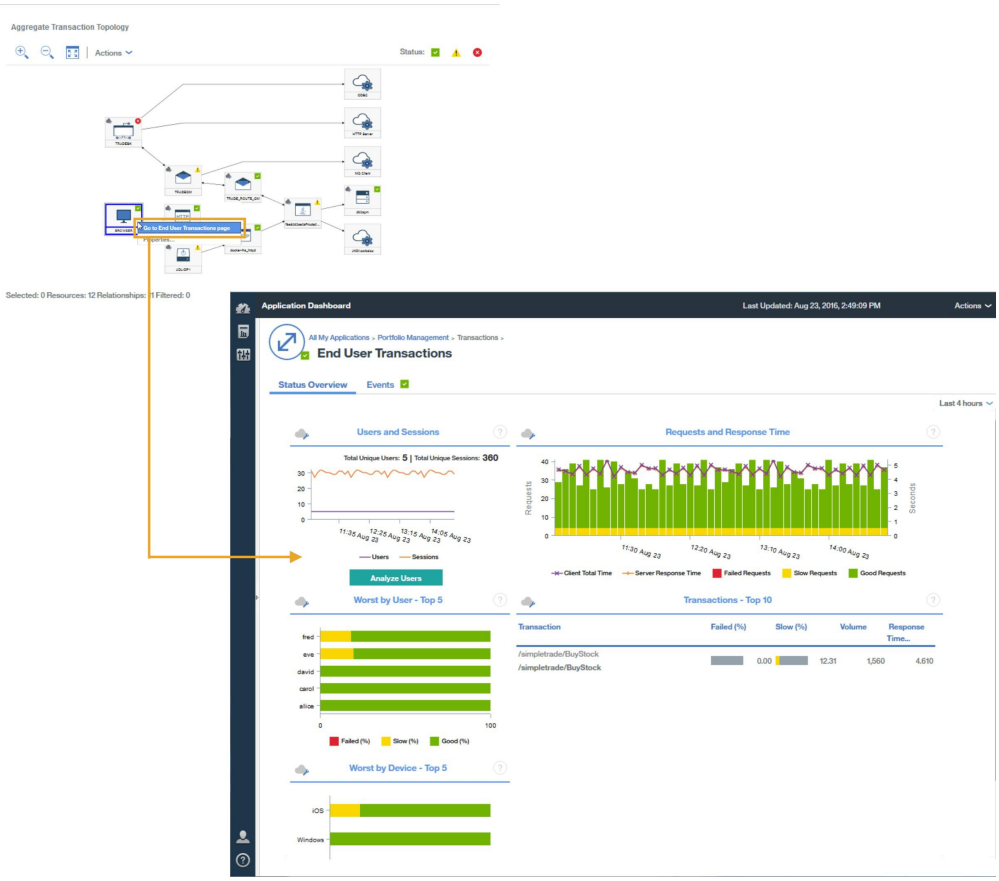
集約トランザクション・トポロジー

集約トランザクション・トポロジーはアプリケーション要約ダッシュボードに表示されます。



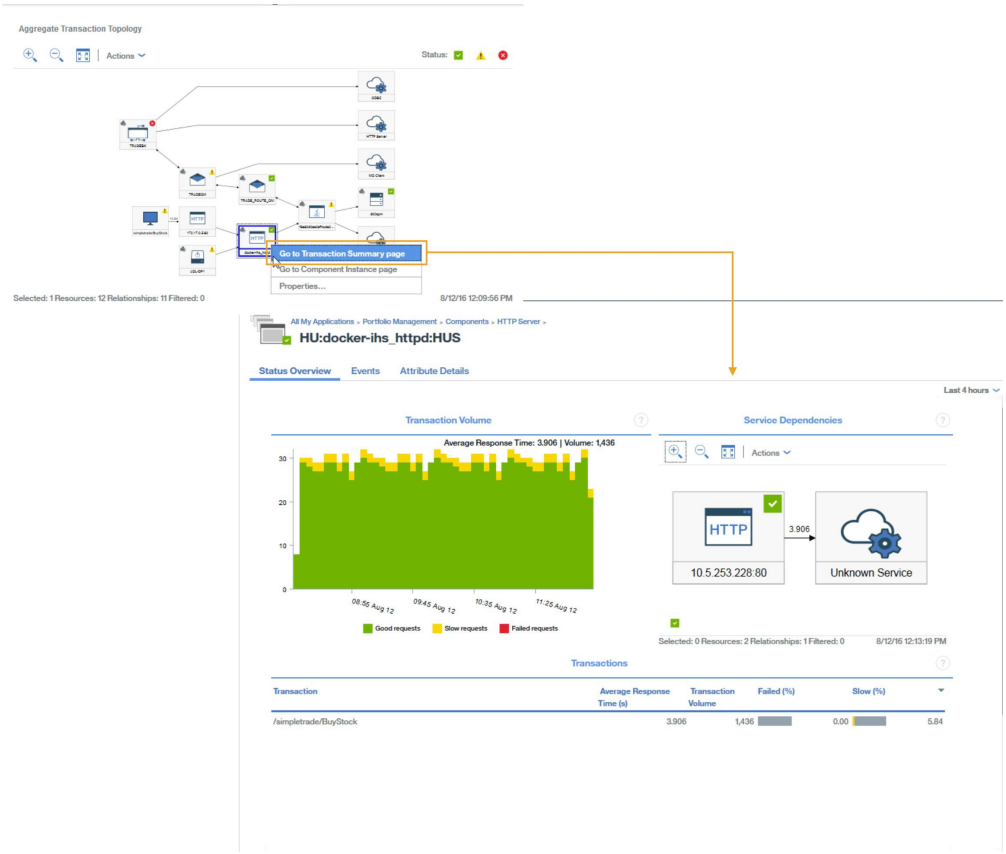
「集約トランザクション・トポロジー」では以下の情報が表示されます。

- ブラウザー・ベース・クライアントのノード。エンド・ユーザー・エクスペリエンスへのドリルダウン



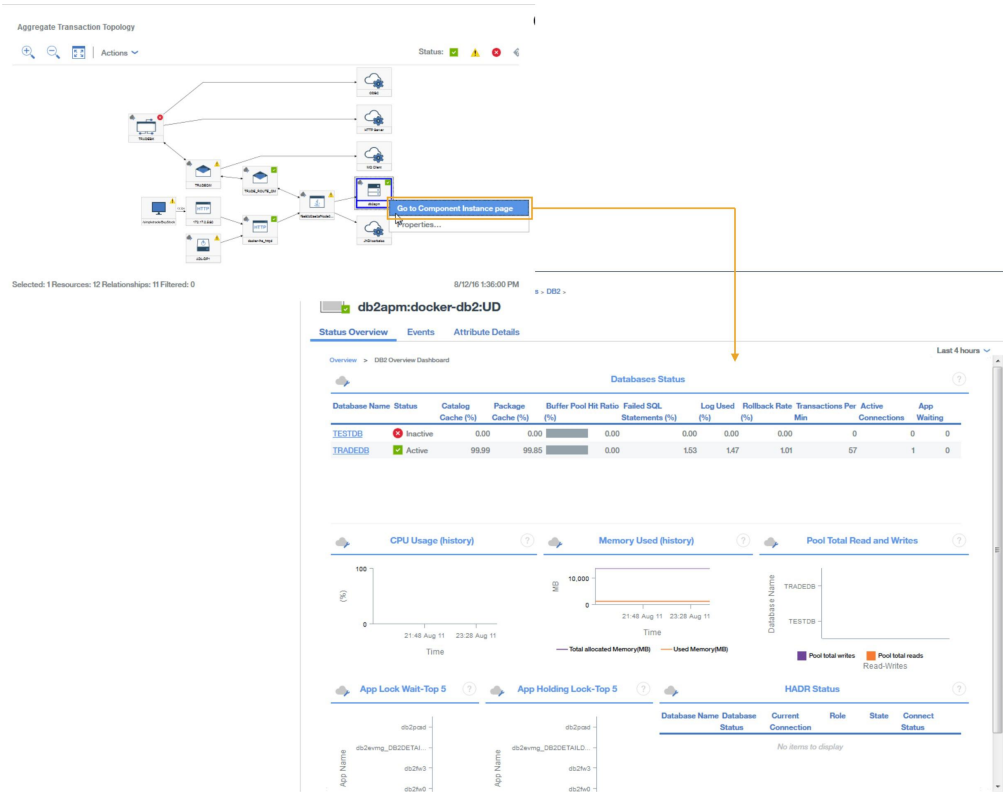
要確認： このノードが表示されるのは、自動 JavaScript インジェクションがブラウザーからのデータを測定している場合のみです。

- HTTP ベースのアプリケーションのノード。Web サーバー・リソース・ページまたはトランザクション要約ページへのドリルダウン



- WebSphere Application Server ベースのアプリケーションのノード。アプリケーション・リソース・ページまたはトランザクション要約ページへのドリルダウン

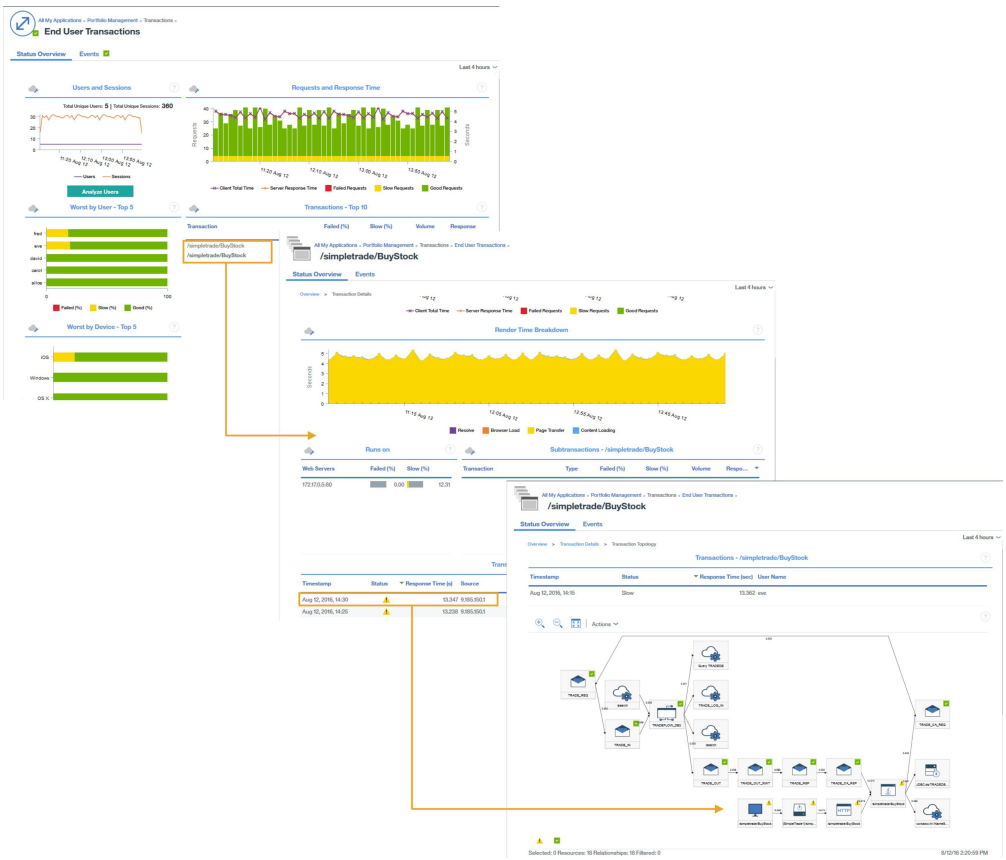
- 特定のデータベース・サーバーのノード。データベース・リソース・ページへのドリルダウン (当該ページを使用できる場合)



トランザクション・インスタンス・トポロジー
トランザクション・インスタンス・トポロジーは実際のエンド・ユーザー・トランザクションに対して表示されます。

「**エンド・ユーザー・トランザクション**」の要約からドリルダウンするには以下のウィジェットを使用します。

1.「トランザクション - 上位 10 件」表でトランザクションを選択
2.「トランザクション・インスタンス」表でインスタンスを選択



Java アプリケーション・スタックのトランザクション・インスタンス・トポロジーには以下のノードが表示されます。ノードをクリックすると、ノードに関する情報が表示されます。

- ブラウザー・ベース・クライアントのノード

  要確認 : このノードが表示されるのは、自動 JavaScript インジェクションがブラウザーからのデータを測定している場合のみです。
- HTTP ノード。ブラウザーからの応答時間を含む
- DataPower ノード (インスツルメントされている場合)
- WebSphere Application Server ノード。ここからアプリケーション・リソース・ページにドリルダウンできます
- データベース・サーバーのノード。ここから、データベースのリソース状況、および JDBC 要求の SQL ステートメント診断情報にドリルダウンできます

  ヒント : 応答時間の大半がデータベースで費やされていることをトポロジーが示している場合は、「**診断**」をクリックすると、SQL ステートメント情報が直接開きます。

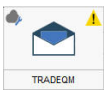
ガント・チャートも表示されます。このチャートにはインスタンスのタイミングが要約表示されます。

環境における問題の診断

環境内のいずれかのコンポーネントのトランザクション・インスタンスが低速になっているかトランザクション・インスタンスに障害が発生している場合、影響を受けているコンポーネントにはそれに応じた状況が割り当てられます。

ノードの状況は以下のいずれかになります。

-

正常: 緑色の正方形で囲まれたチェック・マークがノードの右上隅に表示されます。
- 

警告: 黄色の三角で囲まれた感嘆符がノードの右上隅に表示されます。

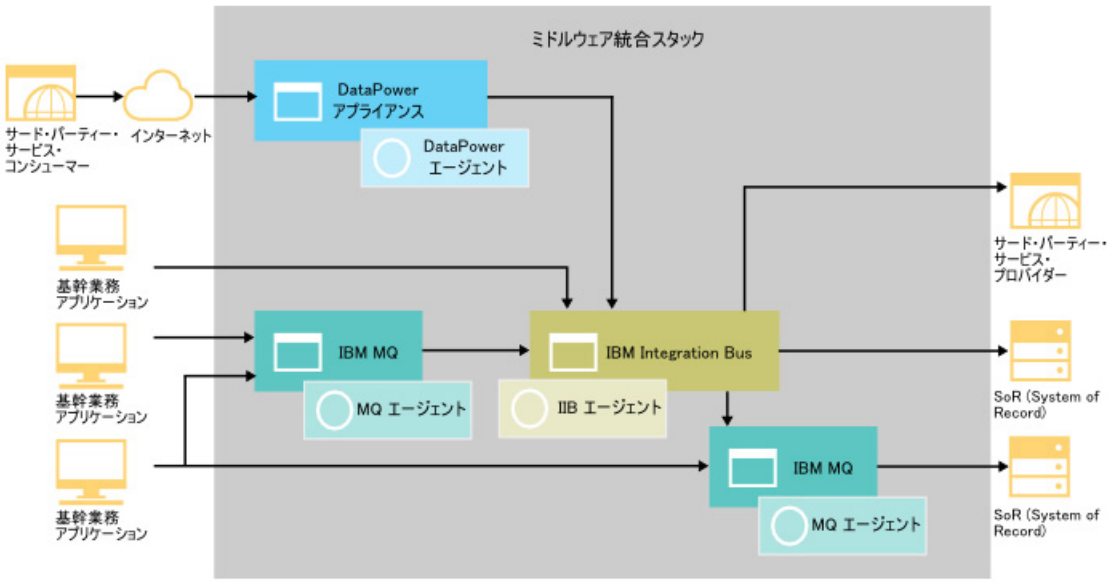
-

クリティカル: ノードの背景が赤色になり、赤色の丸で囲まれた×印がノードの右上隅に表示されます。

状況が警告またはクリティカルになっているこれらの コンポーネントの問題の原因を特定するには、ノードを右クリックしてドリルダウンし、障害の潜在的な原因に関する詳細情報を確認します。

シナリオ: IBM 統合スタックのモニター

IBM 統合スタックをモニターして、ミドルウェア製品とミドルウェア製品が公開するサービスのトランザクション・トラッキング情報を確認することができ、また問題が発生した場合にトラブルシューティングを行うことができます。IBM 統合スタックには、IBM MQ、IBM Integration Bus、および DataPower アプライアンスが含まれています。



IBM 統合スタックのモニター

IBM 統合スタックをモニターするには、コンポーネントごとにリストされるエージェントを、記載されている順にインストールします。

オプションとして、システムもモニターする 場合は、そのシステムに OS エージェントをインストールします。

IBM MQ の場合、以下の手順を実行します。

1. Monitoring Agent for WebSphere MQ をインストールします。
2. WebSphere MQ エージェントを、キュー・マネージャーに接続するように構成します。
3. キュー・マネージャーで、**MQ アプリケーションのアクティビティー ・トレース**を有効にします。

IBM Integration Bus の場合、以下の手順を実行します。

1. Monitoring Agent for IBM Integration Bus をインストールします。
2. トランザクション・トラッキング用に IBM Integration Bus を有効化します。
3. 必要な IBM Integration Bus エージェント・インスタンスのトランザクション・トラッキングを構成します。

DataPower アプライアンス の場合、以下の手順を実行します。

1. Monitoring Agent for DataPower をインストールします。
2. DataPower エージェントが DataPower アプライアンスに接続するように構成します。
3. 必要な DataPower エージェント・インスタンスに対してトランザクション・トラッキングが有効になっていることを確認します。

4. DataPower アプライアンスをセットアップします。

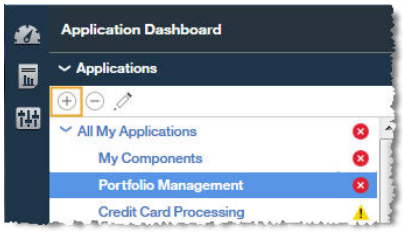
Application Performance ダッシュボードへの ミドルウェア・アプリケーションの追加

IBM 統合スタック・アプリケーションを作成して、そのアプリケーションに、モニターする IBM MQ インスタンス、IBM Integration Bus インスタンス、および DataPower アプライアンス・インスタンスを追加します。

手順

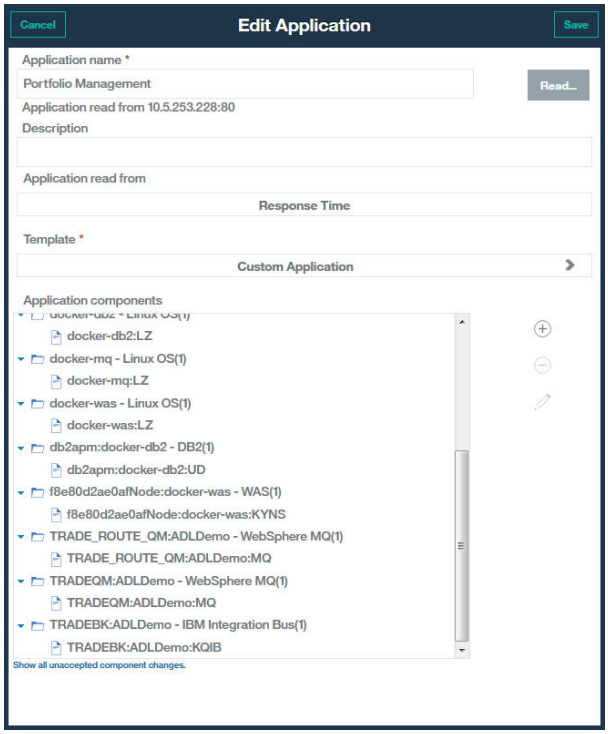
IBM 統合スタック内のコンポーネントを表示するには、Application Performance ダッシュボードで以下のステップを実行します。

1. Application Performance ダッシュボードで、「**アプリケーションの追加**」をクリックします。



2. 「**アプリケーションの編集**」ウィンドウで、アプリケーションの名前を追加し、「**コンポーネントの追加**」 + をクリックします。
3. 「**コンポーネントの選択**」ウィンドウで、「**IBM Integration Bus**」を選択します。
4. 「**コンポーネント・エディター**」で、必要なコンポーネント・インスタンスを選択し、「**追加**」をクリックします。

   検出された IBM Integration Bus インスタンスはこのリストに自動的に追加されます。
5. 「**戻る**」をクリックして、ステップ 3 と 4 を「**WebSphere MQ**」および「**DataPower アプライアンス**」に対して繰り返します。IBM Integration Bus インスタンス、IBM MQ インスタンス、および DataPower アプライアンス・インスタンスの追加を、IBM 統合スタックが完全なものになるまで続行します。



6. 「**閉じる**」をクリックしてから「**保存**」をクリックして、Application Performance ダッシュボードに戻ります。

タスクの結果

ヒント：集約トランザクション・トポロジーに予期したトポロジーが最初に表示されない場合は、最新表示されるのを待ってから数分後に再度確認してください。それでもトポロジーが予期したものと異なる場合は、アプリケーションが所定のコンポーネントと通信していないという問題がある可能性があります。ご使用の環境を確認してください。

IBM 統合スタックのモニター結果の表示

IBM 統合スタックのモニター結果をトポロジーおよびミドルウェアのページに表示することができます。トランザクションが定義されたしきい値に違反したときに生成されるイベントを表示することもできます。

このタスクについて

トポロジーでは、ミドルウェア・コンポーネント間の対話を表示できます。集約トランザクション・トポロジーおよびトランザクション・インスタンス・トポロジーには以下のミドルウェア・ノードが表示されます。

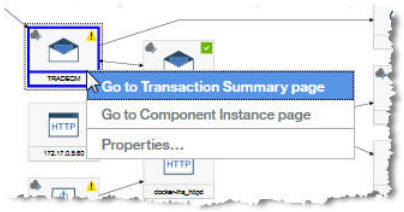
- IBM Integration Bus
- IBM MQ
- DataPower アプライアンス

ノードにマウス・ポインターを合わせると、ノードが特定の状況にある理由を説明した情報を示す「プロパティー」ウィンドウが表示されます。状況はシチュエーションによって決まります。不良の状況を伴うシチュエーションが表示されます。

手順

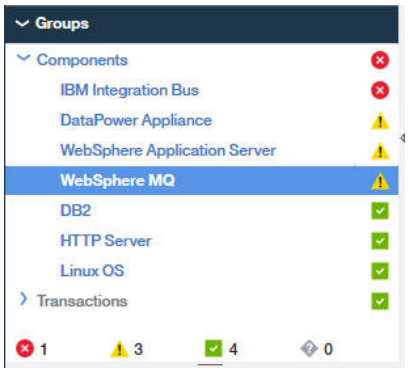
トポロジー内のノードから、そのノードに関する詳細に以下のようにしてリンクできます。

1. ノードを右クリックします。



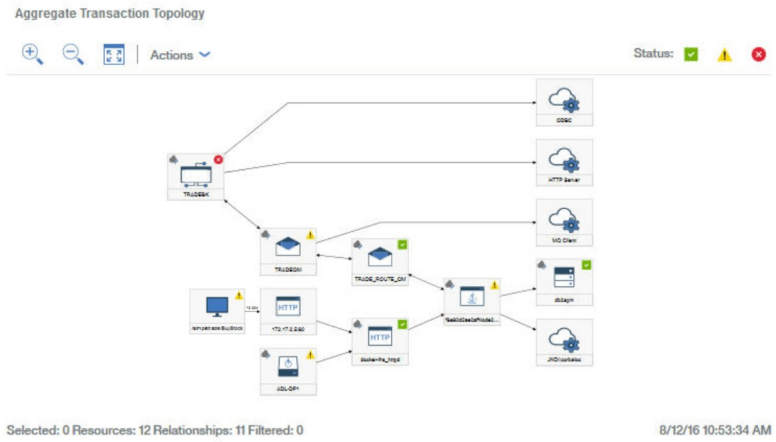
2. 「**「コンポーネント」ページに移動 (Go to Component page)**」を選択して、コンポーネントに関する情報を表示します。
3. 「**「トランザクション要約」ページに移動**」を選択して、ミドルウェアのトランザクションに関する情報を表示します。

   ヒント：ナビゲーターで「**グループ**」 > 「**コンポーネント**」 > 「***middleware component***」を選択し、ボリューム・ウィジェットで要求期間を選択して、同じダッシュボードにアクセスすることもできます。



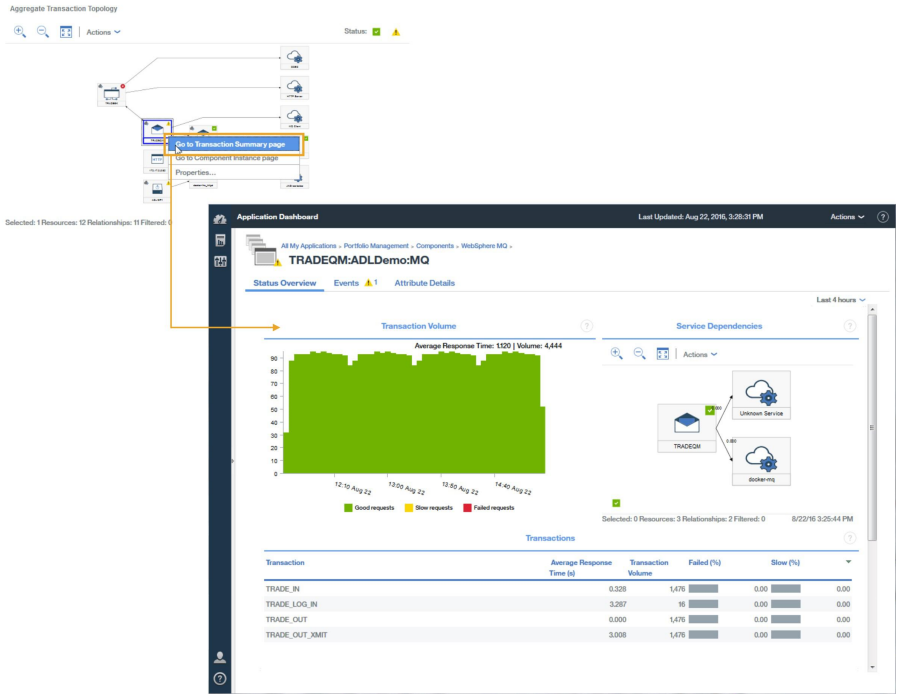
集約トランザクション・トポロジー

集約トランザクション・トポロジーはアプリケーション要約ダッシュボードに表示されます。



集約トランザクション・トポロジーは、IBM MQ ノード、IBM Integration Bus ノードおよび DataPower アプライアンス・ノードを表示できます。これらのノードからドリルダウンして、ミドルウェア統合スタックについて詳しく調べることができます。

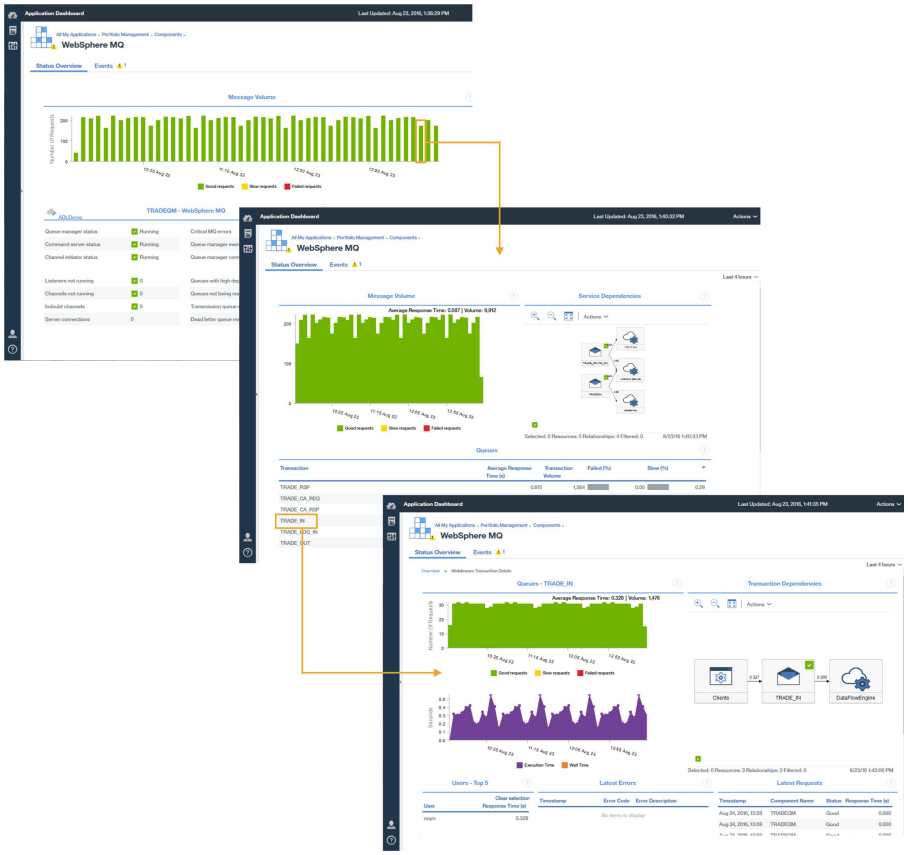
ドリルダウンするには、集約トランザクション・トポロジー内のミドルウェア・ノードを右クリックし、「**「トランザクション要約」ページに移動**」を選択します。あるいは、ナビゲーターで「**グループ**」 > 「**コンポーネント**」 > 「***middleware component***」を選択し、ボリューム・ウィジェットで要求期間を選択して、同じ情報にアクセスすることもできます。



ミドルウェアのトランザクション詳細

ミドルウェアの「トランザクション要約」ページから、ミドルウェアのトランザクションの詳細情報にドリルダウンすることができます。

コンポーネントについてのミドルウェアのトランザクションの詳細にドリルダウンするには、以下の手順を実行します。



1. コンポーネントについてのミドルウェアの「トランザクション要約」ページの「**メッセージまたはボリューム (Message or Volume)**」グラフで、モニター間隔を選択します。
2. 「**ミドルウェア・トランザクション要約**」の「**キュー**」、「**ブローカー**」、または「**アプライアンス**」ウィジェットで、キュー、ブローカー、またはアプライアンスを選択します。

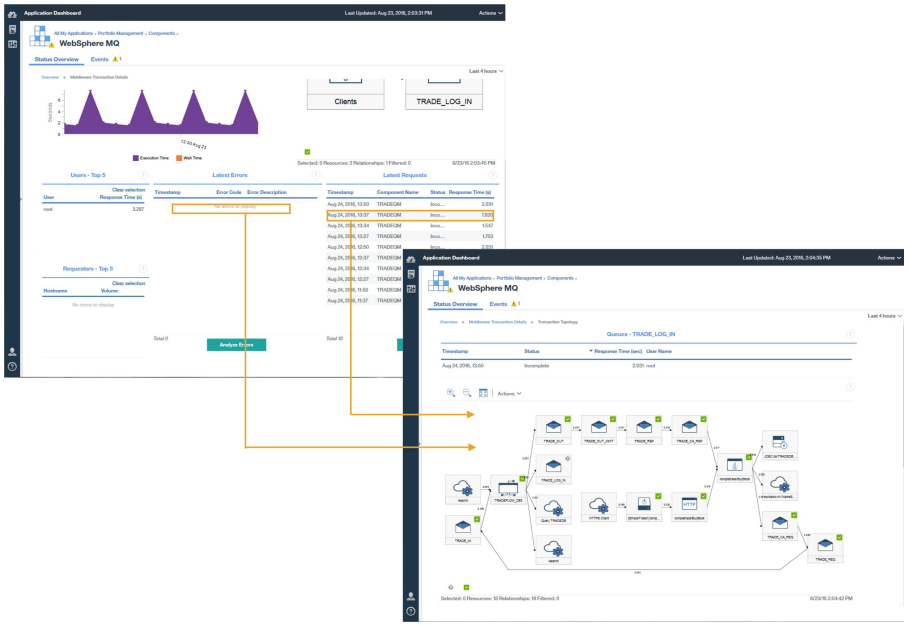
エラーおよびインスタンスの分析

ミドルウェアの「トランザクションの詳細」ページからさらにドリルダウンして、エラーやインスタンスの分析に役立つ情報を表示し、トランザクション・インスタンス・トポロジーにアクセスすることができます。

ミドルウェア・コンポーネントのエラーやインスタンス、さらに、トランザクション・インスタンス・トポロジーにドリルダウンするには、「**トランザクションの詳細**」 ページで、以下のいずれかの手順を実行します。

- 「**エラーの分析**」をクリックして「**エラー分析**」ページを表示してから、エラーを選択します。
- 「**要求の分析**」をクリックして「**インスタンス分析**」ページを表示してから、インスタンスを選択します。

あるいは、「**トランザクションの詳細**」ページでエラーまたはインスタンスを選択して、トランザクション・インスタンス・トポロジーに直接移動することもできます。



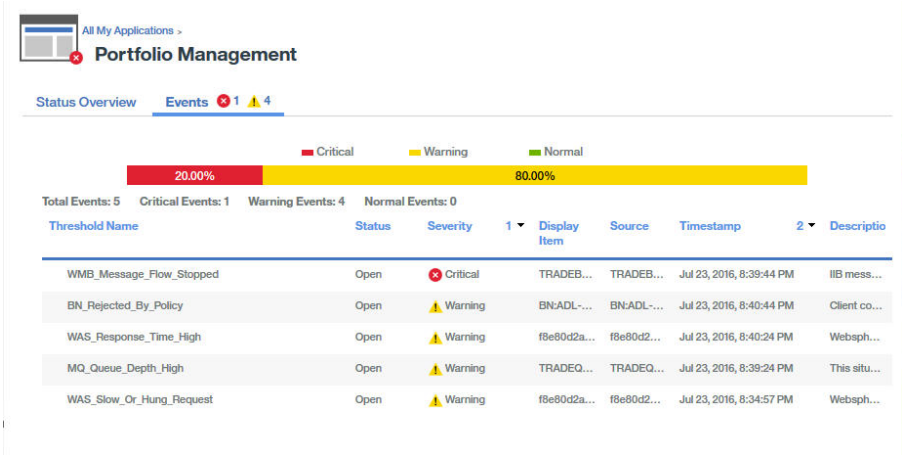
トランザクション・インスタンス・トポロジーには、以下のミドルウェア・ノードが表示されます。

- メッセージ・キュー・マネージャー
- IBM Integration Bus ブローカー
- DataPower アプライアンス

ノードを選択すると、ノードが特定の状況にある理由を説明するそのノードのプロパティーが表示されます。

選択したキューまたはブローカーに対するトランザクション・ガント・チャートが表示されます。ガント・チャートによって、トランザクション全体の応答時間に対する最も重要な要因を探し出すことができます。

イベント
IBM 統合スタックでは、他のエージェントに加え、デフォルトのトランザクション・トラッキングのしきい値によってイベントが生成されます。



トランザクション・トラッキングのデフォルト・イベントについて詳しくは、1015 ページの『トランザクション・モニターのイベントしきい値』を参照してください。

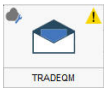
さらにイベントを生成するために (例えば、トランザクション速度が低速の場合や一定のしきい値を下回った場合のイベントなど)、しきい値を追加することができます。

イベントの追加について詳しくは、1017 ページの『トランザクション・モニターのイベントを生成するためのしきい値の作成』を参照してください。

環境における問題の診断
環境内のいずれかのコンポーネントのトランザクション・インスタンスが低速になっているかトランザクション・インスタンスに障害が発生している場合、影響を受けているコンポーネントにはそれに応じた状況が割り当てられます。

ノードの状況は以下のいずれかになります。

-

正常: 緑色の正方形で囲まれたチェック・マークがノードの右上隅に表示されます。
- 

警告: 黄色の三角で囲まれた感嘆符がノードの右上隅に表示されます。
-

クリティカル: ノードの背景が赤色になり、赤色の丸で囲まれた×印がノードの右上隅に表示されます。

状況が警告またはクリティカルになっているこれらの コンポーネントの問題の原因を特定するには、ノードを右クリックしてドリルダウンし、障害の潜在的な原因に関する詳細情報を確認します。

エージェントのダウンロードおよびデータ・コレクター

Cloud APM サブスクリプションには IBM Marketplace Web サイトからもアクセスできます。ご使用のアカウントでサインインし、インストール・アーカイブ・ファイルをダウンロードします。インストール・アーカイブ・ファイルには、エージェントおよびデータ・コレクターのインストール・ファイルと構成ファイルが含まれます。

以下のチュートリアルの手順を実行することにより、エージェントおよびデータ・コレクターのダウンロードの詳細を学習できます。

- 106 ページの『チュートリアル: エージェントのダウンロードとインストール』
- 110 ページの『チュートリアル: データ・コレクターのダウンロードと構成』

IBM Marketplace から、いずれかの IBM Cloud Application Performance Management オファリングのアクティブな使用版、または購入したサブスクリプションにサインアップできます。

IBM Marketplace

「IBM Marketplace」>「Cloud APM」>「Free Trial」でトライアルにサインアップします。IBM マーケットプレイス」>「Cloud APM」>「購入・お問い合わせ」でサブスクリプションを購入します。製品およびサービス ページにサインインし、エージェントとデータ・コレクターをダウンロードします。

「**Products and Services**」ページは、アクティブなサブスクライバーが 使用できます。問題がある場合は、『Marketplace サポート』を参照してください。

接続のテスト

パッケージをダウンロードするために使用する Cloud APM サーバー への接続のテストについては、ネットワーク接続を参照してください。

チュートリアル: エージェントのダウンロードとインストール

このチュートリアルでは、IBM Marketplace からの Cloud APM Windows OS エージェントのダウンロードとインストールを実際に体験することができます。インストールした後で Cloud APM コンソールを開始し、ダッシュボードに重要パフォーマンス指標 (KPI) を表示して、モニター対象リソースの正常性を確認することができます。

このタスクについて

このチュートリアルでは、IBM Marketplace の「**製品およびサービス**」ページから Windows のインストール・パッケージをダウンロードし、インストール・ファイルを解凍して、Windows OS エージェントをインストールします。「**製品およびサービス**」に戻り、Cloud APM コンソールを起動して Application Performance ダッシュボードを開き、Windows システムの正常性を確認します。

手順

1. IBM Marketplace にサインインしていない場合は、IBMid とパスワードでサインインして、「**製品およびサービス**」に移動します。

   「**製品およびサービス**」ページは、アクティブなサブスクライバーが 使用できます。問題がある場合は、Cloud Application Performance Management フォーラム または Marketplace サポート に移動してください。
2. Windows のインストール・アーカイブ・ファイルを以下の手順でダウンロードします。
   a) Cloud APM のサブスクリプション・ボックスで、「**管理**」 > 「**ダウンロード**」をクリックします。
   b) Windows オペレーティング・システムを選択します。

      IBM Cloud Application Performance Management エージェント 64 ビット・パッケージを選択します。 Windows 32 ビット・バージョンを使用している場合は、32 ビット・エージェント・パッケージを選択します。
   c)「**ダウンロード (Download)**」をクリックし、ご使用のシステムにエージェント・インストール・アーカイブ・ファイルを保存します。
   例:

   ```
   C:¥Users¥MY_ADMIN¥Downloads¥IAPM_Agent_Install.zip
   ```
3. ローカル Windows システムで、ダウンロードしたアーカイブ・ファイルの保存先のディレクトリーに移動して、そのファイルを解凍します。
   例えば、Windows エクスプローラーで「**ダウンロード**」ディレクトリーを開き、IAPM_Agent_Install.zip を右クリックして、「**すべて抽出**」を選択します。
4. 以下のように、管理者としてコマンド・プロンプトを開きます。
   a) Windows の「**スタート**」メニューから、検索ボックスに「**コマンド**」と入力します。

b) 表示されるリストから「**コマンド プロンプト**」を右クリックし、「**管理者として実行**」を選択します。

5. インストール・ファイルを解凍したディレクトリーに移動します。
例:

```
cd C:¥Users¥MY_ADMIN¥Downloads¥IAPM_Agent_Install¥IAPM_Agent_Install_8.1.4
```

6. いかのように、インストール・スクリプトを実行して Windows OS エージェント をインストールします。

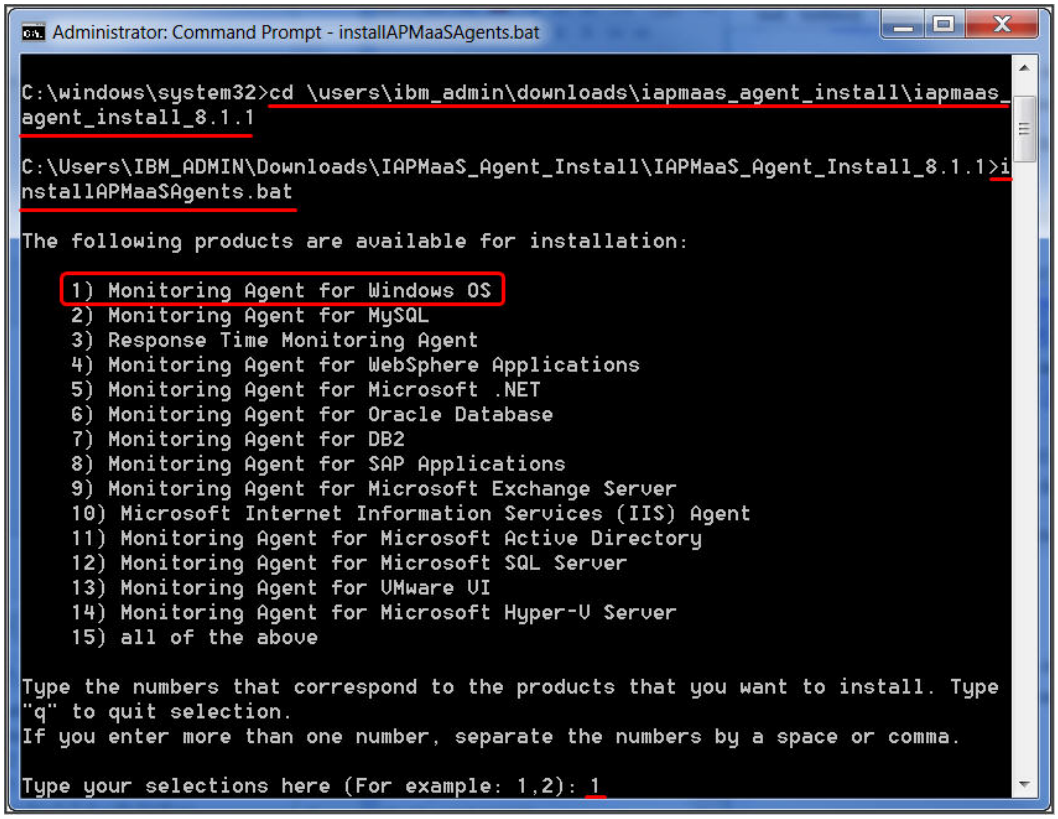
a) 以下のコマンドを入力します。

```
installAPMAgents.bat
```

b) 使用可能なエージェントのリストから、Windows OS エージェントに対応する番号を入力します。

c) プロンプトに応答して、Windows OS エージェントをインストールすることを確認し、ご使用条件に同意します。

ご使用の環境の前提条件スキャンが開始されます。このスキャンが完了するまでにはしばらく時間がかかります。要件が満たされていない場合は、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ディスク・スペースの不足などの前提条件が原因で、インストールは停止します。失敗に対処してから、インストール・スクリプトを再度開始する必要があります。問題がある場合は、Cloud Application Performance Management フォーラムに移動するか `info@ibmserviceengage.com` に E メールを送信してください。



Windows オペレーティング・システム要件レポートを表示するには、System requirements (APM Developer Center)を参照してください。

インストールが正常に完了したら、Windows OS エージェントが自動的に開始されます。ユーザーは、Cloud APM コンソールを開始して、Windows システムのモニターを開始することができます。

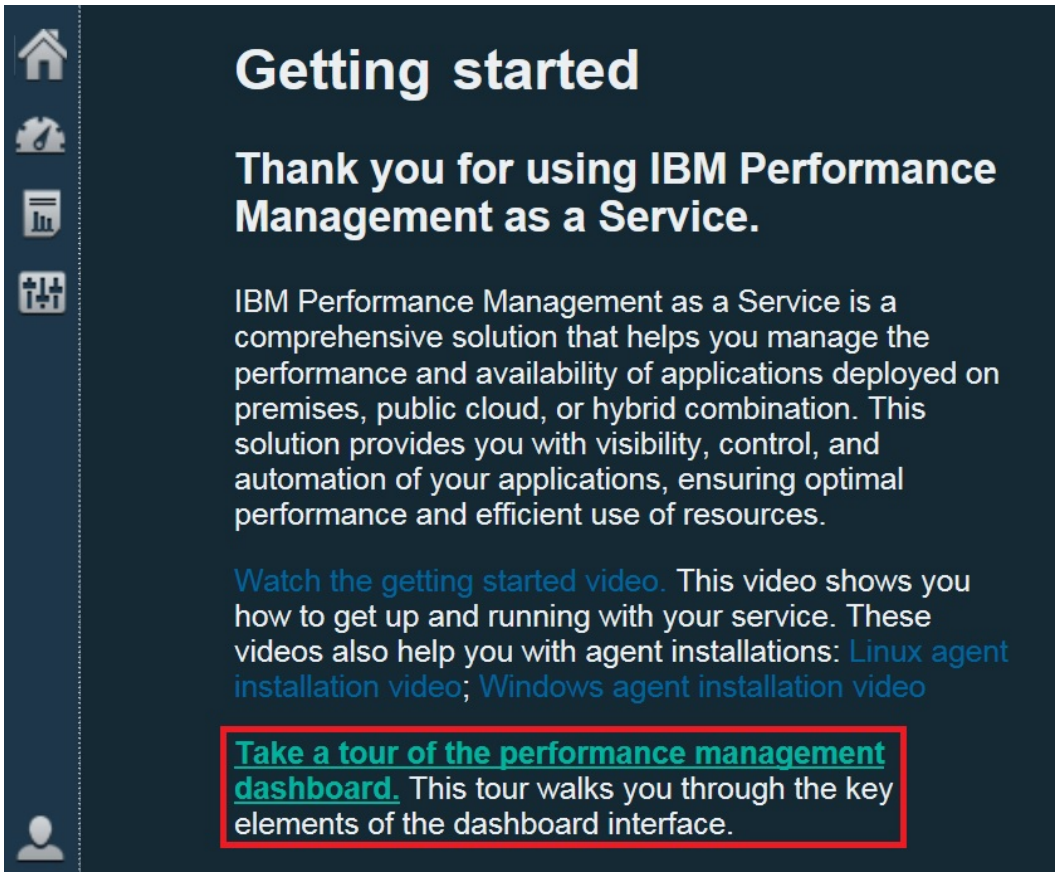
注: ご使用の環境に外部ホストへの透過的アウトバウンド HTTPS 接続を許可しないファイアウォールが含まれている場合は、エージェントと Cloud APM サーバー間の通信にフォワード・プロキシーをセットアップする必要があります。フォワード・プロキシーをセットアップすることにより、ネットワーク上の特定のポイントにすべてのトラフィックを転送して、ファイアウォールで単一の接続のみを許可す

ることが可能になります。詳しくは、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。

7. IBM Marketplace の「**Products and Services**」に戻り、Cloud APM のサブスクリプション・ボックスから「**起動**」をクリックします。

   Cloud APM コンソールが開き、「**はじめに**」ページが表示されます。このページから、各機能について学んだり、さまざまなユーザー・シナリオのビデオを見たり、関連するコンソール・ページを開いたりすることができます。

8. 「**はじめに**」ページで「Performance Management ダッシュボードのツアーをお試しください」をクリックして、ナビゲーション・エレメントのクイック・ツアーを表示します。



9. 以下の手順で Windows OS の要約ダッシュボードを開きます。

   a) ナビゲーション・バーで、「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックします。

   - 「**すべてのマイ・アプリケーション**」ダッシュボードが表示され、環境内で定義されているアプリケーションごとの要約状況ボックスが示されます。最初に表示されるアプリケーションは、「**マイ・コンポーネント**」という事前定義のアプリケーションのみです。
   - 「**マイ・コンポーネント**」の代わりに「**アプリケーションの追加**」ウィンドウが表示される場合は、以下の手順で、モニター対象リソースを表示するためのアプリケーションを作成します。

     1) このアプリケーションの名前を入力します (例えば、「My Apps」など)。

     2) + をクリックします。

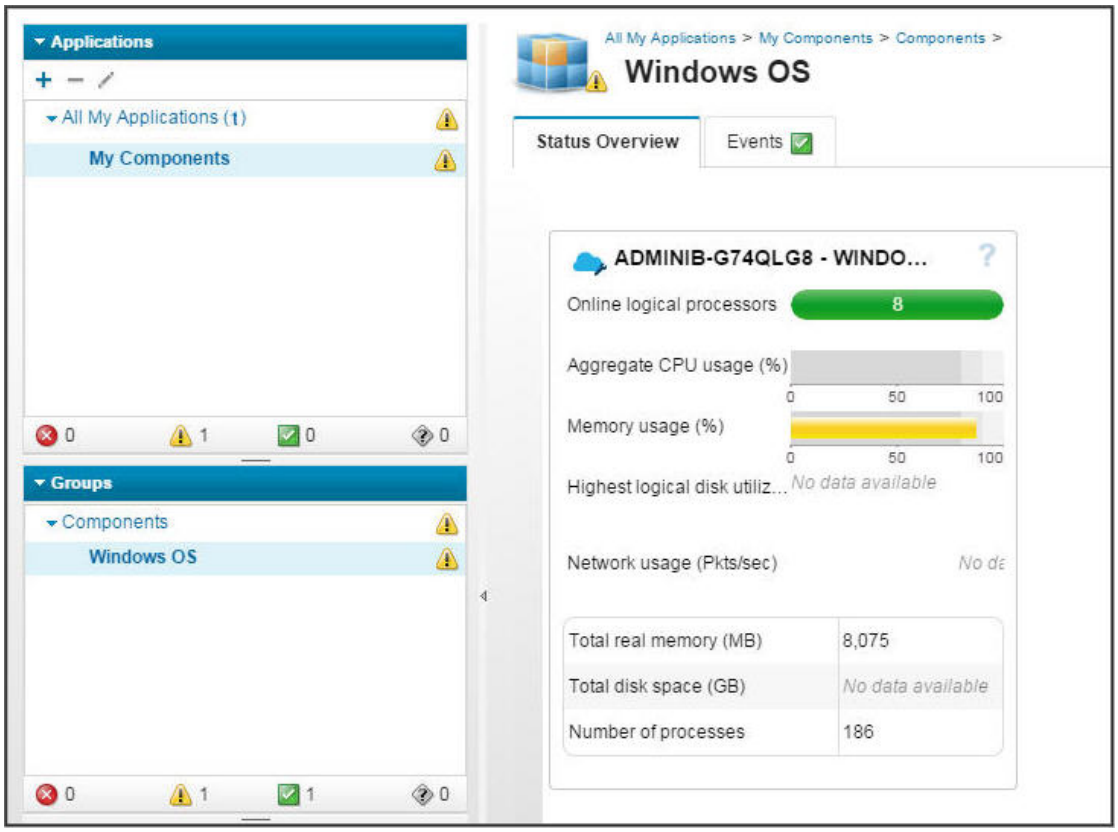
     3) 「**コンポーネントの選択**」リストを最後までスクロールして、「**Windows OS**」をクリックします。

     4) 「コンポーネント・エディター」で、「`Primary:`*`Host_Name`*`:NT`」をクリックし、「**追加**」をクリックした後「**戻る**」をクリックして、このアプリケーションにエージェントを追加します。

     5) 「**保存**」をクリックしてウィンドウを閉じ、ダッシュボードに新しく作成したアプリケーションの要約状況ボックスを表示します。

   b) 要約ボックスで、「**コンポーネント**」をクリックします。

- Windows OS の管理対象システム用の要約ダッシュボードが表示されます。このダッシュボードから、状況要約グループ・ウィジェット内の任意の場所をクリックすると、Windows OS エージェントから報告された KPI を示す詳細なダッシュボードにドリルダウンすることができます。
- 新しく開始されたエージェントがモニター・インフラストラクチャーと通信して、コンソールに KPI を送信するのには数分かかる場合があります。



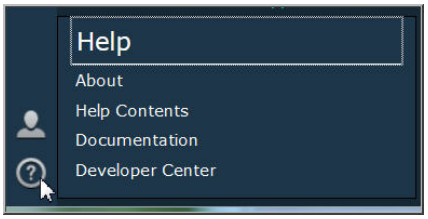
エージェントが Cloud APM サーバーと通信していない場合、またはエージェントが開始されていない場合は、要約ダッシュボードに KPI は表示されず、状況が 「不明」として示されます。必要な場合は、**os-agent** コマンドを使用して状況を確認し、エージェントを開始することができます。コマンド・プロンプトを管理者として開き、`C:¥¥IBM¥APM¥bin` フォルダーから **os-agent status** コマンドを入力します。エージェントが開始されていない場合は、**os-agent start** コマンドを入力します。

タスクの結果

Cloud APM エージェントをインストールし、Application Performance ダッシュボードに送信されたモニター・データを観察しました。

次のタスク

- コンソールを探索します。Cloud APM コンソールを使用しながら、さまざまな機能を調べてみます。ウィンドウ・バナーにある ⓘ をクリックすると、現在のダッシュボードについての情報を入手できます。ナビゲーション・バーの ⓘ 「**ヘルプ**」メニューから、IBM Knowledge Center のヘルプ・システムまたは Cloud APM トピック・コレクションを開くことができます。



- 他のエージェントをインストールします。ご使用の環境のモニター用の他のタイプの Windows エージェントをインストールするために必要なすべてのインストール・ファイルが用意されています。環境内の

他のシステムにエージェントをインストールすることもできます。AIX システムや Linux システムを使用している場合は、それに関連付けられたインストール・アーカイブ・ファイルをダウンロードします。一部のエージェント・タイプでは、エージェントをインストールする前に完了しておく必要がある前提条件があります。また、ほとんどのエージェント・タイプでは、インストールした後に何らかの構成が必要です。詳しくは、125 ページの『AIX システムでのプリインストール』、132 ページの『Linux システムでのプリインストール』、140 ページの『Windows システムでのプリインストール』、および 163 ページの『第 7 章 環境の構成』を参照してください。

チュートリアル: データ・コレクターのダウンロードと構成

このチュートリアルでは、IBM Marketplace からの Cloud APM Bluemix Ruby データ・コレクターのダウンロードと構成を実際に体験することができます。インストールした後で Cloud APM コンソールを開始し、ダッシュボードに重要パフォーマンス指標 (KPI) を表示して、モニター対象リソースの正常性を確認することができます。

このタスクについて

このチュートリアルでは、IBM Marketplace の「**製品およびサービス**」ページから Bluemix アプリケーション用のデータ・コレクター・パッケージをダウンロードし、構成ファイルを解凍して、Linux システム上で Bluemix Ruby データ・コレクターを構成します。「**製品およびサービス**」に戻り、Cloud APM コンソールを起動して Application Performance ダッシュボードを開き、Bluemix Ruby アプリケーションの正常性を確認します。

手順

1. IBM Marketplace にサインインしていない場合は、IBMid とパスワードでサインインして、「**製品およびサービス**」に移動します。

   「**製品およびサービス**」ページは、アクティブなサブスクライバーが 使用できます。問題がある場合は、Cloud Application Performance Management フォーラム または Marketplace サポート に移動してください。
2. Bluemix アプリケーション用のデータ・コレクター・パッケージ `IBM_Bluemix_Data_Collectors_Install.tgz` をダウンロードします。
3. ローカル・システムで、ダウンロードしたアーカイブ・ファイルの保存先のディレクトリーに移動し、以下のコマンドを実行して、そのファイルを解凍します。

   ```
   tar -zxvf IBM_Bluemix_Data_Collectors_Install.tgz
   ```

   4 つの圧縮ファイルが得られます。各ファイルは、1 つのタイプの Bluemix アプリケーション用のデータ・コレクターを表しています。Bluemix Ruby アプリケーションのデータ・コレクター・パッケージは `ruby_datacollector.tgz` です。
4. 例えば、以下のコマンドを実行して、`ruby_datacollector.tgz` 内のファイルを解凍します。

   ```
   tar -zxvf ruby_datacollector.tgz
   ```

   `ibm_ruby_dc` フォルダーが得られます。
5. 例えば、以下のコマンドを実行して、Ruby アプリケーションのルート・フォルダーに `ibm_ruby_dc` 内の `etc` フォルダー全体をコピーします。

   ```
   cp –r directory to the etc folder  home directory of your Ruby application
   ```

   データ・コレクターを `/opt/ibm/ccm/ibm_ruby_dc/etc` ディレクトリーに解凍し、ご使用の Ruby アプリケーションのホーム・ディレクトリーが `/root/ruby_app/` である場合は、コマンドは次のようになります。

   ```
   cp –r /opt/ibm/ccm/ibm_ruby_dc/etc /root/ruby_app/
   ```
6. Bluemix Ruby アプリケーションのホーム・フォルダーにある `Gemfile` に次のセクションを追加します。

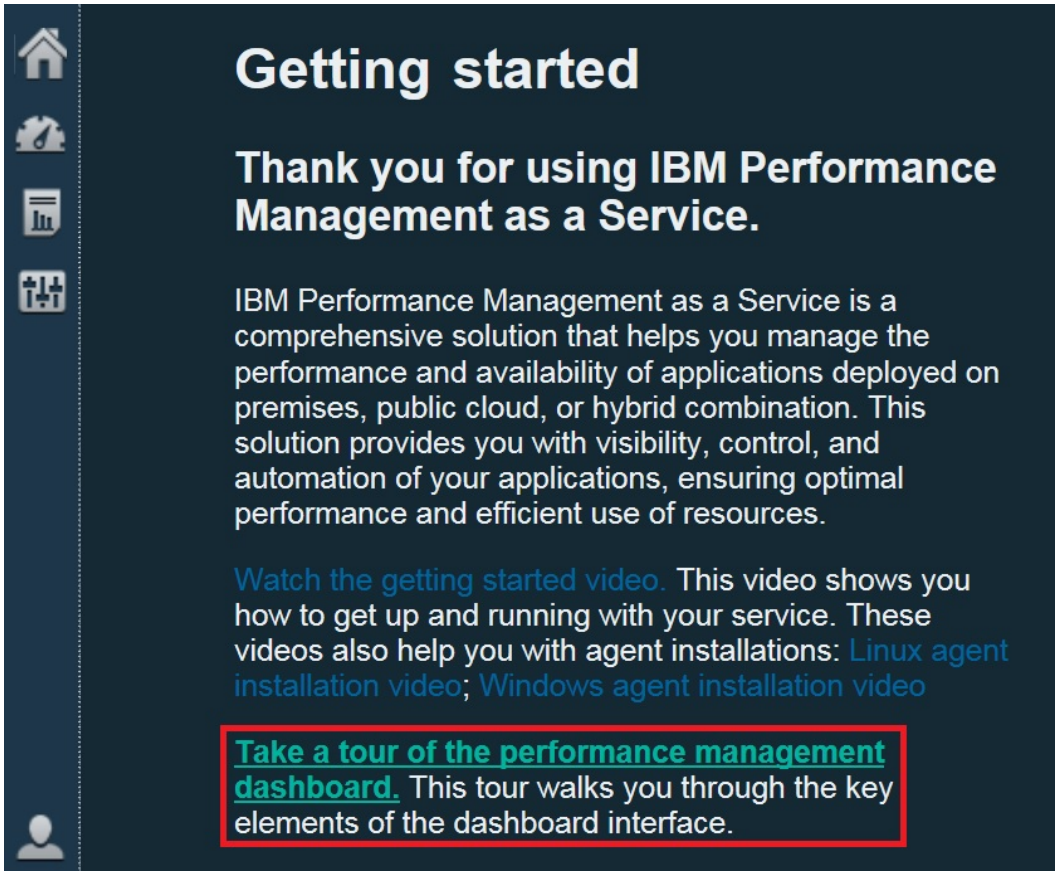
```
gem 'logger',  '>= 1.2.8'
source 'https://managemserver.ng.bluemix.net' do
  gem 'ibm_resource_monitor'
  gem 'stacktracer'
end
```

7. `bundle lock` コマンドを実行して、`Gemfile.lock` ファイルを再生成します。
8. Ruby アプリケーションのホーム・ディレクトリーから、以下のコマンドを実行します。

```
cf push
```

9. IBM Marketplace の「**Products and Services**」に戻り、Cloud APM のサブスクリプション・ボックスから「**起動**」をクリックします。

   Cloud APM コンソールが開き、「**はじめに**」ページが表示されます。このページから、各機能について学んだり、さまざまなユーザー・シナリオのビデオを見たり、関連するコンソール・ページを開いたりすることができます。
10. 「**はじめに**」ページで「Performance Management ダッシュボードのツアーをお試しください」をクリックして、ナビゲーション・エレメントのクイック・ツアーを表示します。



11. 次のように Bluemix Ruby の要約ダッシュボードを開きます。

    a) ナビゲーション・バーで、「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックします。

    - 「**すべてのマイ・アプリケーション**」ダッシュボードが表示され、環境内で定義されているアプリケーションごとの要約状況ボックスが示されます。最初に表示されるアプリケーションは、「**マイ・コンポーネント**」という事前定義のアプリケーションのみです。
    - 「**マイ・コンポーネント**」の代わりに「**アプリケーションの追加**」ウィンドウが表示される場合は、以下の手順で、モニター対象リソースを表示するためのアプリケーションを作成します。

      1) このアプリケーションの名前を入力します (例えば、「My Apps」など)。

      2) + をクリックします。

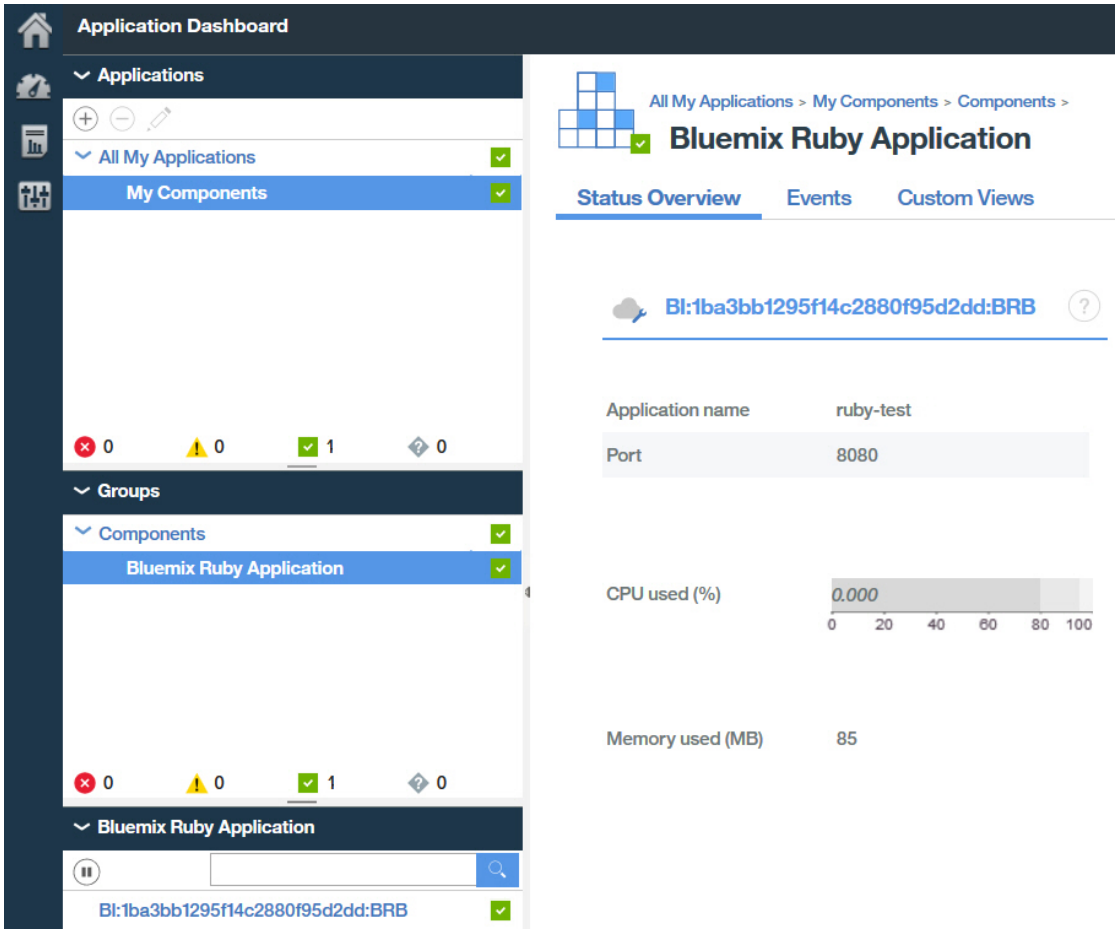
      3) 「**Bluemix Ruby アプリケーション**」をクリックします。

4)「コンポーネント・エディター」で、インスタンスを選択し、「**追加**」をクリックした後「**戻る**」をクリックして、このアプリケーションにデータ・コレクターを追加します。

5)「**保存**」をクリックしてウィンドウを閉じ、ダッシュボードに新しく作成したアプリケーションの要約状況ボックスを表示します。

b) 要約ボックスで、「**コンポーネント**」をクリックします。

- Bluemix Ruby アプリケーション用の要約ダッシュボードが表示されます。このダッシュボードから、状況要約グループ・ウィジェット内の任意の場所をクリックすると、Bluemix Ruby データ・コレクターから報告された KPI を示す詳細なダッシュボードにドリルダウンすることができます。
- 新しく開始されたデータ・コレクターがモニター・インフラストラクチャーと通信して、コンソールに KPI を送信するのには数分かかる場合があります。

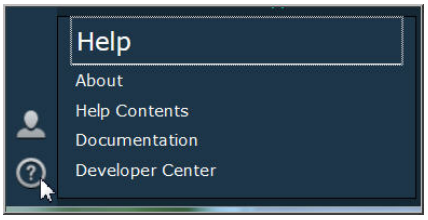


タスクの結果

Cloud APM データ・コレクターをインストールし、Application Performance ダッシュボードに送信されたモニター・データを観察しました。

次のタスク

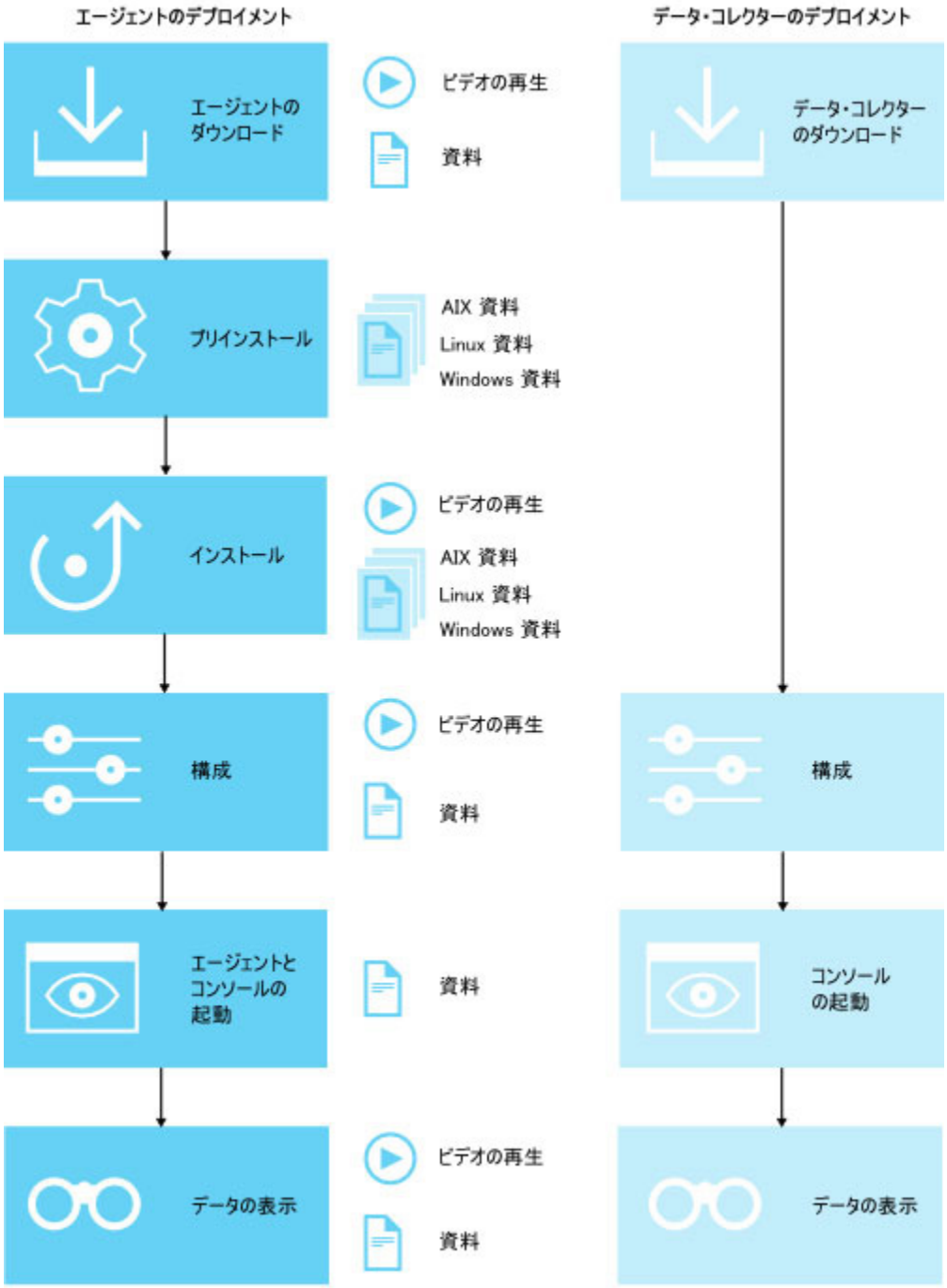
- コンソールを探索します。Cloud APM コンソールを使用しながら、さまざまな機能を調べてみます。ウィンドウ・バナーにある ? をクリックすると、現在のダッシュボードについての情報を入手できます。ナビゲーション・バーの ? 「**ヘルプ**」メニューから、IBM Knowledge Center のヘルプ・システムまたは Cloud APM トピック・コレクションを開くことができます。



- 他のデータ・コレクターをインストールします。ご使用の環境のモニター用の他のタイプのデータ・コレクターをインストールするために必要なすべてのインストール・ファイルが用意されています。環境内の他のシステムにデータ・コレクターをインストールすることもできます。詳しくは、163 ページの『第 7 章 環境の構成』を参照してください。

第 5 章 エージェント とデータ・コレクターのデプロイメント

エージェントでは、インストールに必要なタスクと、そのエージェントが収集するデータの表示に必要なタスクは異なります。タスクには、自動で行われるものと、手動で行われるものがあります。データ・コレクターのダウンロード後、各データ・コレクターを手動で構成する必要があります。



1. ダウンロードのビデオの再生

© Copyright IBM Corp. 2014, 2019

2. 資料のダウンロード
3. AIX のプリインストールの資料
4. Linux のプリインストールの資料
5. Windows のプリインストールの資料
6. インストールのビデオの再生
7. AIX のインストールの資料
8. Linux のインストールの資料
9. Windows のインストールの資料
10. 構成のビデオの再生
11. 構成の資料
12. 起動の資料
13. データの表示のビデオの再生
14. 機能の資料

1. ダウンロードのビデオの再生
2. 資料のダウンロード
3. AIX のプリインストールの資料
4. Linux のプリインストールの資料
5. Windows のプリインストールの資料
6. インストールのビデオの再生
7. AIX のインストールの資料
8. Linux のインストールの資料
9. Windows のインストールの資料
10. 構成のビデオの再生
11. 構成の資料
12. 起動の資料
13. データの表示のビデオの再生
14. 機能の資料

インストール後に、一部のエージェントは自動的に構成されて開始されます
開始されるエージェントはいずれも、デフォルト設定を使用して構成されます。手動で構成および開始されるエージェントを判別するには、117 ページの表 7 を参照してください。

インストール後に、一部のエージェントは手動構成が必要ですが、開始は自動的に行われます
エージェントの構成方法については、163 ページの『第 7 章 環境の構成』 を参照してください。 手動で構成され、自動的に開始されるエージェントを判別するには、117 ページの表 7 を参照してください。

インストール後に、一部のエージェントは手動で構成および開始する必要があります
自動的に開始されないエージェントについては、エージェントを開始するために事前にエージェントを構成する必要があります。手動で構成および開始されるエージェントを判別するには、117 ページの表 7 を参照してください。

複数インスタンス・エージェントでは、最初のインスタンスを作成し、手動で開始する必要があります
最初のインスタンスを作成し、エージェントを手動で開始する必要があります。複数インスタンスのエージェントの場合は、エージェントの単一のインストールによって、固有のアプリケーション・インスタンスごとに固有のモニター・インスタンスがインスタンス化されることになります。結果として Cloud APM コンソール からそのようなインスタンスを表示できます。どのエージェントが複数インスタンス・エージェントなのかを判別するには、117 ページの表 7 を参照してください。

OS エージェントおよびログ・ファイル・モニター

Linux OS エージェント、UNIX OS エージェント、および Windows OS エージェントは自動的に構成されて開始されます。ただし、アプリケーション・ログ・ファイルをモニターできるように、OS エージェントに対してログ・ファイル・モニターを構成できます。詳しくは、635 ページの『OS エージェント・ログ・ファイル・モニターの構成』を参照してください。

エージェント とデータ・コレクターの構成、開始、およびインスタンスの特性

表 7. ポストインストール・チェックリスト

エージェント またはデータ・コレクター	自動的に構成および開始	手動で構成され、自動で開始	手動で構成および開始	複数インスタンス(手動で開始)
Amazon EC2 エージェント	–	–	–	✓
Amazon ELB エージェント	–	–	–	✓
Azure Compute エージェント	–	–	–	✓
Cassandra エージェント	–	–	✓	✓
Cisco UCS エージェント	–	–	✓	✓
Citrix VDI エージェント	–	–	–	✓
DataPower エージェント	–	–	–	✓
DataStage エージェント	–	–	✓	✓
Db2 エージェント	–	–	–	✓
Hadoop エージェント	–	–	✓	–
HMC Base エージェント	–	–	–	✓
HTTP Server エージェント	✓ エージェントによって HTTP Server 用に作成された構成ファイルを確認する必要があります。次に、 手動でデータ・コレクターの構成をサーバー構成ファイルに追加する必要があります。	–	–	–
IBM Cloud エージェント	–	–	–	✓
IBM Integration Bus エージェント	–	–	–	✓
J2SE データ・コレクター	–	✓	–	–
JBoss エージェント	–	–	–	✓
Liberty データ・コレクター	–	✓	–	–
Linux KVM エージェント	–	–	✓	✓

表 7. ポストインストール・チェックリスト (続き)

エージェント またはデータ・コレクター	自動的に構成および開始	手動で構成され、自動で開始	手動で構成および開始	複数インスタンス(手動で開始)
Linux OS エージェント	✓	–	–	–
Microsoft Active Directory エージェント	–	✓ エージェントは自動的に開始されます。ただし、一部の属性のデータを表示するには、エージェントを構成する必要があります。	–	–
Microsoft Cluster Server エージェント	–	–	✓	–
Microsoft Exchange Server エージェント	–	✓ エージェントは自動的に開始されます。ただし、すべての属性のデータを表示するには、エージェントを構成する必要があります。	–	–
Microsoft Hyper-V Server エージェント	✓	–	–	–
Microsoft IIS エージェント	✓	–	–	–
Skype for Business Server エージェント (以前の Microsoft Lync Server エージェント)	✓	✓ エージェントは自動的に開始されます。ただし、一部の属性のデータを表示するには、エージェントを構成する必要があります。	–	–
Microsoft Office 365 エージェント	–	–	✓	–
Microsoft .NET エージェント	–	✓ データのレポートが作成される前にデータ・コレクターを構成する必要があります。	–	–
Microsoft SharePoint Server エージェント	✓	–	–	–

表 7. ポストインストール・チェックリスト (続き)

エージェント またはデータ・コレクター	自動的に構成および開始	手動で構成され、自動で開始	手動で構成および開始	複数インスタンス(手動で開始)
Microsoft SQL Server エージェント	–	–	–	✓ 各エージェント・インスタンスを手動で構成して開始する必要があります。
MQ Appliance エージェント	–	–	–	✓
MongoDB エージェント	–	–	–	✓
MySQL エージェント	–	–	–	✓
NetApp Storage エージェント	–	–	✓	✓
Node.js エージェント	–	✓ データのレポートが作成される前にエージェントを構成する必要があります。Node.js アプリケーションにモニター・プラグインを追加する必要があります。	–	–
Node.js データ・コレクター	–	✓	–	–
OpenStack エージェント	–	–	✓	✓
Oracle Database エージェント	–	–	–	✓
PHP エージェント	–	–	–	✓
PostgreSQL エージェント	–	–	–	✓
Python データ・コレクター	–	✓	–	–
RabbitMQ エージェント	–	–	✓	✓
Response Time Monitoring エージェント	✓	–	–	–

表 7. ポストインストール・チェックリスト (続き)

エージェント またはデータ・コレクター	自動的に構成および開始	手動で構成され、自動で開始	手動で構成および開始	複数インスタンス(手動で開始)
Ruby エージェント	–	–	–	✓ 詳細診断の場合は、データのレポートが作成される前にエージェントを構成する必要があります。診断ダッシュボードを有効にするには、診断データ・コレクターをインストールして構成する必要があります。
Ruby データ・コレクター	–	✓	–	–
SAP エージェント	–	–	✓	✓
SAP HANA データベース・エージェント	–	–	✓	✓
SAP NetWeaver Java Stack エージェント	–	–	✓	✓
Siebel エージェント	–	–	–	✓
Sterling Connect Direct エージェント	–	–	–	✓
Sterling File Gateway エージェント	–	–	✓	✓
Sybase エージェント	–	–	✓	✓
Synthetic Playback エージェント	✓ エージェントは外部向けパブリック・アプリケーション用に自動的に構成されて開始されますが、データのレポートが作成される前に、Synthetic Script Manager でトランザクションを作成する必要があります。	✓ エージェントは自動的に開始されますが、内部向けプライベート・アプリケーション用にエージェントを構成する必要があります。データのレポートが作成される前に、Synthetic Script Manager でトランザクションを作成する必要があります。	–	✓
Tomcat エージェント	–	–	–	✓
UNIX OS エージェント	✓	–	–	–

表 7. ポストインストール・チェックリスト (続き)

エージェント またはデータ・コレクター	自動的に構成および開始	手動で構成され、自動で開始	手動で構成および開始	複数インスタンス(手動で開始)
VMware VI エージェント	–	–	✓	✓
WebLogic エージェント	–	–	–	✓
WebSphere Applications エージェント	–	✓ エージェントは自動的に開始されますが、データのレポートが作成される前にデータ・コレクターを構成する必要があります。	–	–
WebSphere Infrastructure Manager エージェント	–	–	–	✓
WebSphere MQ エージェント	–	–	–	✓
Windows OS エージェント	✓	–	–	–

第 6 章 エージェントのインストール

IBM Cloud Application Performance Management インフラストラクチャーは、IBM によってインストールおよび管理されます。アプリケーションをモニターするには、モニター対象のアプリケーション用にモニター・エージェントを選択してインストールします。エージェントは、Linux、AIX、または Windows の各オペレーティング・システムにインストールすることができます。スタンドアロン・データ・コレクターはインストールする必要がありません。

アプリケーションのモニターにスタンドアロン・データ・コレクターを使用する場合は、インストール手順を省略できます。163 ページの『第 7 章 環境の構成』に進んで、アプリケーションをモニターするためのデータ・コレクターをデプロイする手順を確認してください。

リモート・モニター

一部のエージェントは、モニターするリソースからリモートにインストールできます。リモート・モニターは以下のエージェントでサポートされます。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS Elastic Load Balancer
- Monitoring Agent for Azure Compute
- Monitoring Agent for Cassandra
- Monitoring Agent for Cisco UCS
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for DataPower - このエージェントは、リモート・マシン上にのみインストールできます。
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HMC Base
- Monitoring Agent for IBM Cloud
- Monitoring Agent for InfoSphere DataStage
- Monitoring Agent for JBoss - リソース・モニターにこのエージェントを使用する場合は、リモートまたはローカルにインストールします。トランザクション・トラッキングおよび詳細診断にエージェントを使用する場合は、ローカルにインストールします。
- Monitoring Agent for Linux KVM
- Monitoring Agent for MariaDB
- Monitoring Agent for Microsoft Cluster Server
- Monitoring Agent for Microsoft Exchange Server
- Monitoring Agent for Microsoft Office 365
- Monitoring Agent for Microsoft SharePoint Server
- Monitoring Agent for MongoDB
- Monitoring Agent for MySQL
- Monitoring Agent for NetApp Storage
- Monitoring Agent for OpenStack
- Monitoring Agent for Oracle Database
- Monitoring Agent for PostgreSQL
- Monitoring Agent for RabbitMQ
- Monitoring Agent for SAP Applications

© Copyright IBM Corp. 2014, 2019

- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack - リソース・モニターにこのエージェントを使用する場合は、リモートまたはローカルにインストールします。トランザクション・トラッキングおよび詳細診断にエージェントを使用する場合は、ローカルにインストールします。
- Monitoring Agent for Sterling Connect Direct
- Monitoring Agent for Sterling File Gateway
- Monitoring Agent for VMware VI
- Monitoring Agent for WebLogic - リソース・モニターにこのエージェントを使用する場合は、リモートまたはローカルにインストールします。トランザクション・トラッキングおよび詳細診断にエージェントを使用する場合は、ローカルにインストールします。
- Response Time Monitoring エージェント - Packet Analyzer コンポーネントを使用する場合は、エージェントをリモートまたはローカルにインストールできます。IBM HTTP Server Response Time モジュールを使用する場合は、HTTP Server と同じマシンにエージェントをインストールする必要があります。

UNIX システムでのエージェントのインストール

管理対象リソースの AIX システムまたは Solaris システムにモニター・エージェントをインストールします。

AIX にインストールできるエージェント・リスト

- Monitoring Agent for DataPower
- Monitoring Agent for Cassandra
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HMC Base
- Monitoring Agent for HTTP Server
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for MQ Appliance
- Monitoring Agent for Oracle Database
- Monitoring Agent for SAP Applications
- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack
- Monitoring Agent for Siebel
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebLogic
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ
- Response Time Monitoring エージェント

Solaris Sparc にインストールできるエージェント・リスト

- Monitoring Agent for Db2
- Monitoring Agent for HTTP Server
- Monitoring Agent for JBoss
- Monitoring Agent for MySQL
- Monitoring Agent for Oracle Database

- Monitoring Agent for SAP Applications
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebLogic

Solaris X86 にインストールできるエージェント・リスト

- Monitoring Agent for Db2
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for SAP Applications
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ

AIX システムでのプリインストール

AIX システムにエージェントをインストールする 前に、必要なプリインストール・タスクを完了する必要があります。一部のプリインストール・タスクはエージェント固有であり、その他のタスクは複数のエージェントに適用されます。

注 : これらの要件は、Software Product Compatibility Reports で示された要件を補足するものです。

ご使用のエージェントの現行バージョンの要件および依存関係については、System requirements (APM Developer Center) にある Software Product Compatibility Reports へのリンクを参照してください。

すべてのエージェント

以下のプリインストール・タスクは、すべてのエージェントに適用されます。

接続のテスト
エージェントをインストールする 前に、システムが Cloud APM サーバーと通信可能であることを確認してください。Cloud APM サーバーへの接続を検査する方法については、『ネットワーク接続』を参照してください。

root 以外のユーザーのインストール
インストール・ディレクトリーに対する読み取り、書き込み、および実行の許可を持っている必要があります。これらの許可がない場合、インストールがキャンセルされます。root 以外のユーザーのインストールについて詳しくは、145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください。

インストール・パスの 70 文字の制限
インストール・ディレクトリーとそのパスは 70 文字以下である必要があります。

.tar ファイル名の 100 文字の制限
AIX システムのデフォルトの **tar** コマンドは、100 文字を超えるファイル名を処理できません。インストールの問題を回避するには、以下のステップを実行します。

1. AIX Toolbox for Linux Applications の Web サイトから GNU 版の **tar** コマンドをダウンロードしてインストールします。
2. GNU 版をデフォルトの **tar** コマンドにします。以下のいずれかのステップを実行します。
   - 現在の *PATH* 環境変数の先頭に /opt/freeware/bin を追加します。例:

     ```
     export PATH=/opt/freeware/bin:$PATH
     ```

     ここで、/opt/freeware/bin は GUN bin のディレクトリーです。

- 以下に示すように /bin/tar を /opt/freeware/bin/tar に対するシンボリック・リンクで置き換えます。

```
ln -s /opt/freeware/bin/tar /bin/tar
```

あるいは、 最新バージョンの AIX にアップグレードし、100 文字を超えるファイル名を処理するための修正コードを入手してください。詳しくは、TAR コマンドの技術情報 (AIX V6.1)または TAR コマンドの技術情報 (AIX V7.1)を参照してください。

***CANDLEHOME* 環境変数の設定**

以前に ITM Agent Converter を使用して同じ管理対象システムにエージェントをインストールして 構成した場合は、*CANDLEHOME* 環境変数が、Agent Converter でエージェントをインストールしたディレクトリーに変更されています。ネイティブ Cloud APM エージェントをインストールして構成する前に、*CANDLEHOME* 環境変数を別のディレクトリーに設定する必要があります。そうしないと、ネイティブ Cloud APM エージェントを開始できません。

特定のエージェント

以下のプリインストール・タスクは、指定されたエージェントに適用されます。

DataPower エージェント

エージェントのインストール前に、AIX 上で *ulimit* が **unlimited** に設定されていることを前提条件チェッカーが検査します。*max data segment size* システム環境変数が確実に **unlimited** に設定されるように、**ulimit -d unlimited** コマンドを実行する必要があります。 このエージェントは、モニター対象の DataPower アプライアンスと同じマシンにインストールすることはできません。

HMC Base エージェント

エージェントを root ユーザーとしてインストールする場合は、システム TL07 がインストールされていることを確認する必要があります。エージェントを root 以外のユーザーとしてインストールする場合は、システム TL08 が AIX バージョン 6 専用にインストールされていることを確認する必要があります。

HTTP Server エージェント

このエージェントは、root ユーザーとしてインストールおよび実行してください。エージェントのインストールおよび実行には同じユーザー ID を使用してください。root 以外のユーザーとしてエージェントをインストールおよび実行する場合は、その非 root ユーザーが、IBM HTTP Server を始動したユーザーと同じユーザー ID を持っている必要があります。そうでない場合は、エージェントが IBM HTTP Server をディスカバーするときに問題が発生します。

AIX システムでデフォルトの **.tar** コマンドによって長いパスが切り捨てられたことにより、AIX でのインストールは失敗します。詳しくは、このトピックの『.tar ファイル名の 100 文字の制限』セクションを参照してください。

AIX のみ: lynx ユーティリティーまたは curl アプリケーションをインストールします。

Oracle Database エージェント

Red Hat Enterprise Linux バージョン 5 とバージョン 6 および SUSE Linux Enterprise Server バージョン 11 とバージョン 12 x64 では、Oracle Database エージェントが Oracle データベースをリモートでモニターする場合は、最初に Oracle インスタント・クライアントをインストールする 必要があります。Oracle インスタント・クライアントは、Oracle Technology Network - Instant Client のダウンロード からインストールしてください。

インスタント・クライアント v10.x、v11.x、および v12.x は、Oracle Database エージェント によってサポートされます。

Response Time Monitoring エージェント

Response Time Monitoring エージェントをインストールする前に、インストール計画のセクション (688 ページの『インストールの計画』) を確認してください。

SAP HANA データベース・エージェント

1. AIX システムに SAP HANA データベース・クライアント HDBSQL バージョン 1.00.102.06 以降をインストールします。

2. 以下のコマンドを実行して、インストール・ディレクトリーのパスを **LIBPATH** 環境変数に追加します。

   `export LIBPATH=$LIBPATH:`*`install_directory_path`*

   例: `export LIBPATH=$LIBPATH:/usr/sap/hdbclient`。ここで、`/usr/sap/hdbclient` は SAP HANA データベース・クライアントのインストール・パスを示します。

重要 :

SAP HANA データベース・クライアントのインストール・パスが **LIBPATH** 環境変数に追加されていない場合、前提条件スキャナーによって FAIL の結果が返されます。

export コマンドを使用して追加された環境変数は、端末の特定のセッションに対してのみ保持されます。そのため、環境変数の追加に使用されたのと同じ端末からエージェント・インストール・スクリプトを実行するようにします。

WebSphere Applications エージェント
エージェントのインストール前に、AIX システム上で *ulimit* が **524000** に設定されていることを前提条件チェッカーが検査します。*max data segment size* システム環境変数が確実に **524000** に設定されるように、**ulimit -d 524000** コマンドを実行する必要があります。

Solaris システムでのプリインストール

Solaris システムにエージェントをインストールする 前に、必要なプリインストール・タスクを完了する必要があります。一部のプリインストール・タスクはエージェント固有であり、その他のタスクは複数のエージェントに適用されます。

注 : これらの要件は、Software Product Compatibility Reports で示された要件を補足するものです。

ご使用のエージェントの現行バージョンの要件および依存関係については、System requirements (APM Developer Center) にある Software Product Compatibility Reports へのリンクを参照してください。

すべてのエージェント

以下のプリインストール・タスクは、すべてのエージェントに適用されます。

接続のテスト
エージェントをインストールする前に、システムが Cloud APM サーバーと通信可能であることを確認してください。Cloud APM サーバーへの接続を検査する方法については、『ネットワーク接続』を参照してください。

root 以外のユーザーのインストール
インストール・ディレクトリーに対する読み取り、書き込み、および実行の許可を持っている必要があります。これらの許可がない場合、インストールがキャンセルされます。root 以外のユーザーのインストールについて詳しくは、145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください。

インストール・パスの 70 文字の制限
インストール・ディレクトリーとそのパスは 70 文字以下である必要があります。

.tar ファイル名の 100 文字の制限
Solaris システムのデフォルトの **tar** コマンドは、100 文字を超えるファイル名を処理できません。@LongLink エラーの問題を回避するには、以下のステップを実行します。

1. http://www.gnu.org の Web サイトから GNU 版の **tar** コマンドをダウンロードしてインストールします。
2. GNU 版をデフォルトの **tar** コマンドにします。以下のいずれかのステップを実行します。
   - *PATH* 環境変数で、先頭に以下の変数を指定します。

     ```
     export PATH=/opt/freeware/bin:$PATH
     ```

   - `/bin/tar` を `/opt/freeware/bin/tar` へのシンボリック・リンクで置き換えます。

***CANDLEHOME* 環境変数の設定**

以前に ITM Agent Converter を使用して同じ管理対象システムにエージェントをインストールして 構成した場合は、*CANDLEHOME* 環境変数が、Agent Converter でエージェントをインストールしたディレクトリーに変更されています。ネイティブ Cloud APM エージェントをインストールして構成する前に、*CANDLEHOME* 環境変数を別のディレクトリーに設定する必要があります。そうしないと、ネイティブ Cloud APM エージェントを開始できません。

特定のエージェント

以下のプリインストール・タスクは、指定されたエージェントに適用されます。

HTTP Server エージェント

このエージェントは、root ユーザーとしてインストールおよび実行してください。エージェントのインストールおよび実行には同じユーザー ID を使用してください。root 以外のユーザーとしてエージェントをインストールおよび実行する場合は、その非 root ユーザーが、IBM HTTP Server を始動したユーザーと同じユーザー ID を持っている必要があります。そうでない場合は、エージェントが IBM HTTP Server をディスカバーするときに問題が発生します。

エージェントのインストール

管理対象システムにはモニター・エージェントを自由に組み合わせてインストールできます。例えば、Ruby On Rails アプリケーションをモニターするために Ruby エージェントをインストールする場合に、Response Time Monitoring エージェント、Linux OS エージェント、またはその両方のエージェントもインストールしたい場合があります。Response Time Monitoring エージェントでは、Ruby アプリケーションの応答時間についてさらに多くの情報を収集できます。Linux OS エージェントでは、システムのその他の状況 (CPU 全体、メモリー、ディスクなど) をモニターすることができます。

インストール可能なモニター・エージェントの種類はオファリングによって決まります。それぞれのオファリングに含まれるエージェントのリストについては、54 ページの『機能』を参照してください。

AIX システムおよび Solaris システムで稼働するエージェントのリストについては、124 ページの『UNIX システムでのエージェントのインストール』を参照してください。

始める前に

エージェントをダウンロードします。105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

85 ページの『システム要件』の情報を参照して、インストールする予定のエージェントの要件を必ず満たしてください。

エージェントをインストールする前に、エージェントのプリインストール・タスクを確認してください。

- AIX システムの場合は、125 ページの『AIX システムでのプリインストール』を参照してください。
- Solaris システムの場合は、127 ページの『Solaris システムでのプリインストール』を参照してください。

重要 : Java ランタイムは、エージェントがそれを必要としている場合にのみインストールされるものであり、使用可能でない場合があります。また、ksh はエージェントのインストールを実行するためには不要になり、enforcing モードの SELinux がサポートされています。

このタスクについて

モニター・エージェントは、root ユーザーとしてインストールすることも、非 root ユーザーとしてインストールすることもできます。root 特権を持たないユーザーがモニター・エージェントをインストールしたい場合には、非 root ユーザーとしてエージェントをインストールできます。145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください。また、ホスト管理者が root ユーザーとしてモニター・エージェントを実行したくない場合にも、非 root ユーザーとしてエージェントをインストールできます。インストール・フローは root ユーザーの場合と同じです。

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Cloud Application Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェント・タイプをインストールすることはできませ

ん。 エージェントの共存について詳しくは、950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。

手順

1. AIX システムまたは Solaris システムで端末シェル・セッションを開きます。
2. ご使用のシステムで、.tar ファイルをダウンロードしたディレクトリーに移動します。

   エージェントは、モニターするアプリケーションがインストールされているシステムにインストールする必要があります。必要に応じて、モニター対象のシステムにインストール・アーカイブ・ファイルを転送します。アーカイブ・ファイルにはエージェントとインストール・スクリプトが含まれています。

   要確認 : 旧バージョンのアーカイブ・ファイルがディレクトリーに存在しないことを確認してください。
3. 次のコマンドを使用して、 インストール・ファイルを抽出します。

   ```
   tar -xf ./installation files
   ```

   ここで、*installation files* はご使用のオファリングのインストール・ファイル名です。

   インストール・スクリプトは、このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーに解凍されます。例えば、*offering*_Agent_Install_8.1.4.0 などです。エージェント・バイナリーおよび構成関連ファイルは、そのディレクトリー内のサブディレクトリーに解凍されます。
4. オプション: このステップは、Solaris 10 の場合のみ必要です。Solaris 10 でインストール・スクリプトを実行する前に、ksh へのソフト・リンクを作成しておく必要があります。

   a) /bin/sh コマンドをバックアップします。

   ```
   mv /bin/sh /bin/sh.bkup_origin
   ```

   b) ksh コマンドへのソフト・リンクを作成します。

   ```
   ln -s /bin/ksh /bin/sh
   ```

   c) 結果が ksh を指し示すことを確認します。

   ```
   ls -l /bin/sh
   ```
5. このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーから、以下のインストール・スクリプトを実行します。

   ```
   ./installAPMAgents.sh
   ```

   エージェントをサイレント・モードでインストールするには、147 ページの『エージェントのサイレント・インストール』を参照してください。
6. 個々のエージェント、エージェントの組み合わせ、またはすべてのエージェントのいずれをインストール対象とするかを指定します。
7. エージェントをインストールしているのか、アップグレードしているのかに応じて、以下のいずれかの手順を実行します。

   - エージェントをインストールしている場合、エージェント・インストール済み環境の異なるホーム・ディレクトリーを指定するか、以下の該当するデフォルト・ディレクトリーを使用します。

     – /opt/ibm/apm/agent
   - エージェントをアップグレードしている場合に、エージェントのインストール済み環境のホーム・ディレクトリーの 入力を求めるプロンプトが出されたときは、以前のバージョンのエージェントのインストール・ディレクトリーを入力してください。
8. ご使用条件への同意を確認するメッセージが表示されたら、1 (ご使用条件に同意して続行する場合) または 2 (同意しない場合) を入力します。

1 (同意) を入力すると、環境の前提条件スキャンが開始され、しばらくすると完了します。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ライブラリーの欠落やディスク・スペースの不足といった前提条件が原因で、インストールが停止します。失敗に対処してから、インストール・スクリプトを再度開始する必要があります。

9. 非 root ユーザー ID を使用してエージェントをインストールした場合、システム起動スクリプトを更新する必要があります (145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください)。
10. インストールが完了してコマンド・ラインが使用可能になったら、この手順のステップを繰り替えし、管理対象システムにモニター・エージェントを追加でインストールできます。

次のタスク

必要に応じて、以下のようにエージェントを構成します。モニター・エージェントで、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』に記載されている構成が必要な場合、またはデフォルト設定を確認する必要がある場合は、163 ページの『第 7 章 環境の構成』を参照してください。

- ご使用のファイアウォールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていないために、フォワード・プロキシーを使用する場合、エージェントの環境構成ファイルを編集する必要があります。説明については、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。
- 以前のバージョンからエージェントをアップグレードした 場合は、Cloud APM コンソールにログインする前に実行する必要がある再構成タスクまたはマイグレーション・タスクを特定してください。それらのタスクについては、1139 ページの『エージェントのアップグレード』を参照してください。アップグレードした後、インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。

エージェントを始動するには、次のコマンドを実行します。

```
./name-agent.sh start
```

name の使用法を含む、モニター・エージェント・コマンドの詳細については、181 ページの『エージェント・コマンドの使用』を参照してください。自動で開始されるエージェントおよび手動で開始されるエージェントについて詳しくは、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。

アップグレードした後、インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。

エージェントを構成して開始した後で、エージェントが収集しているデータを表示します。

- ログインしていない場合は、977 ページの『Cloud APM コンソールの始動』の手順に従ってください。
- IBM Tivoli Monitoring ドメインの管理対象システムを Application Performance ダッシュボードに表示する場合、949 ページの『IBM Tivoli Monitoring V6.3 との統合』で説明しているタスクを実行します。

Linux システムでのエージェントのインストール

管理対象リソースの Linux システムにモニター・エージェントをインストールします。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS Elastic Load Balancer
- Monitoring Agent for Azure Compute
- Monitoring Agent for Cassandra
- Monitoring Agent for Cisco UCS
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for DataPower
- Monitoring Agent for Db2

- Monitoring Agent for Hadoop
- Monitoring Agent for HTTP Server
- Monitoring Agent for IBM Cloud
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for Internet Services
- Monitoring Agent for MQ Appliance
- Monitoring Agent for InfoSphere DataStage
- Monitoring Agent for JBoss
- Monitoring Agent for Linux OS
- Monitoring Agent for Linux KVM
- Monitoring Agent for MariaDB
- Monitoring Agent for Microsoft SQL Server
- Monitoring Agent for MongoDB
- Monitoring Agent for MySQL
- Monitoring Agent for NetApp Storage
- Monitoring Agent for Node.js
- Monitoring Agent for OpenStack
- Monitoring Agent for Oracle Database
- Monitoring Agent for PHP
- Monitoring Agent for PostgreSQL
- Monitoring Agent for RabbitMQ
- Monitoring Agent for Ruby
- Monitoring Agent for SAP Applications
- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack
- Monitoring Agent for Siebel
- Monitoring Agent for Sterling Connect Direct
- Monitoring Agent for Sterling File Gateway
- Monitoring Agent for Sybase Server
- Monitoring Agent for Synthetic Playback
- Monitoring Agent for Tomcat
- Monitoring Agent for VMware VI
- Monitoring Agent for WebLogic
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere Infrastructure Manager
- Monitoring Agent for WebSphere MQ
- Response Time Monitoring エージェント

Linux on Power Little Endian (pLinux LE) システムでは、以下のエージェントがサポートされます。

- Monitoring Agent for Db2
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for Linux OS
- Monitoring Agent for Tomcat - リソース・モニターがサポートされます。
- Monitoring Agent for WebSphere Applications

- Monitoring Agent for WebSphere MQ

以下のエージェントが Linux for System z システムでサポートされています。

- Monitoring Agent for Db2
- Monitoring Agent for HTTP Server - トランザクション・トラッキングはサポートされません。
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for Linux OS
- Response Time Monitoring エージェント
- Monitoring Agent for Tomcat
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ

以下のエージェントが Linux for System x システムでサポートされています。

- Monitoring Agent for HTTP Server - トランザクション・トラッキングはサポートされません。

Linux システムでのプリインストール

Linux システムにエージェントをインストールする 前に、必要なプリインストール・タスクを完了する必要があります。一部のプリインストール・タスクはエージェント固有であり、その他のタスクは複数のエージェントに適用されます。

注 : これらの要件は、Software Product Compatibility Reports で示された要件を補足するものです。

ご使用のエージェントの現行バージョンの要件および依存関係については、System requirements (APM Developer Center) にある Software Product Compatibility Reports へのリンクを参照してください。

すべてのエージェント

以下のプリインストール・タスクは、すべてのエージェントに適用されます。

接続のテスト
エージェントをインストールする 前に、システムが Cloud APM サーバーと通信可能であることを確認してください。Cloud APM サーバーへの接続を検査する方法については、『ネットワーク接続』を参照してください。

root 以外のユーザーのインストール
インストール・ディレクトリーに対する読み取り、書き込み、および実行の許可を持っている必要があります。これらの許可がない場合、インストールがキャンセルされます。root 以外のユーザーのインストールについて詳しくは、145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください。

インストール・パスの 70 文字の制限
インストール・ディレクトリーとそのパスは 70 文字以下である必要があります。

***CANDLEHOME* 環境変数の設定**
以前に ITM Agent Converter を使用して同じ管理対象システムにエージェントをインストールして 構成した場合は、*CANDLEHOME* 環境変数が、Agent Converter でエージェントをインストールしたディレクトリーに変更されています。ネイティブ Cloud APM エージェントをインストールして構成する前に、*CANDLEHOME* 環境変数を別のディレクトリーに設定する必要があります。そうしないと、ネイティブ Cloud APM エージェントを開始できません。

特定のオペレーティング・システム

Red Hat Enterprise Linux (RHEL) 8

RHEL 8 には libnsl.so.1 パッケージが必要です。

デフォルトでは、Red Hat Enterprise Linux リリース 8.0 には `libnsl.so.1` がインストールされていません。このパッケージがないと、エージェントを正常にインストールできません。管理者に yum リポジトリーのセットアップを依頼し、以下のコマンドを実行してください。

```
yum install libnsl
```

インストールが正常に完了すると、`/usr/lib64/libnsl.so.1` が表示されます。

注: `libnsl.so.1` パッケージが必要になるのは、対象がエージェントの場合のみです。対象がデータ・コレクターの場合、このステップを行う必要はありません。

一部のエージェントを対象とする前提条件スキャナーのバイパス

最新のリリースとの互換性を保つように前提条件スキャナーを更新する前に、一部のエージェントを対象に、前提条件スキャナーをバイパスできます。適したシナリオおよび説明については、149 ページの『前提条件スキャナーのバイパス』を参照してください。

注: 対象がデータ・コレクターの場合、このステップを行う必要はありません。

特定のエージェント

以下のプリインストール・タスクは、指定されたエージェントに適用されます。

DataPower エージェント

max data segment size システム環境変数が確実に **unlimited** に設定されるように、**ulimit -d unlimited** コマンドを実行する必要があります。 このエージェントは、モニター対象の DataPower アプライアンスと同じマシンにインストールすることはできません。

DataStage エージェント

1. `DSODBConfig.cfg` ファイル内でパラメーターを有効にします。以下のステップを実行してください。
   a. 以下の場所にある `DSODBConfig.cfg` ファイルをエディターで開きます。

      `infosphere_information_server_install_dir/Server/DSODB`

   b. # 記号を削除して、以下のパラメーターのコメントを外します。

      MonitorLinks=1
      JobRunUsage=1
      ResourceMonitor=1
      DSODBON=1

   c. これらのパラメーターの値を編集して、1 に設定します。
2. エージェント・コンピューター上でメタデータ・リポジトリー構成に使用されているデータベースの JDBC ドライバーをコピーします。
   a. タイプ 4 JDBC 4 以降。例: `db2jcc4.jar`
   b. Oracle 用タイプ 4 JDBC ドライバー。例: `ojdbc6.jar`
   c. 以下の MS SQL 用 JDBC ドライバー。
      - `Sqljdbc41.jar` は、JRE 7 を必要とし、JDBC 4.1 API をサポートします。
      - `Sqljdbc42.jar` は、JRE 8 を必要とし、JDBC 4.2 API をサポートします。

HTTP Server エージェント

このエージェントを root ユーザーとしてインストールする場合は、同じユーザー ID を使用してエージェントを実行および構成する必要があります。

root 以外のユーザーとしてエージェントをインストールおよび実行する場合は、その非 root ユーザーが、IBM HTTP Server を始動したユーザーと同じユーザー ID を持っている必要があります。そうでない場合は、エージェントが IBM HTTP Server をディスカバーするときに問題が発生します。同じユーザー ID を使用してエージェントを実行および構成できます。

Linux KVM エージェント

Monitoring Agent for Linux KVM は、複数インスタンスの複数接続エージェントであり、Enterprise Linux ベースの KVM ハイパーバイザー環境と Red Hat Enterprise Virtualization Manager (RHEV-M) 環境への接続をサポートします。RHEV-M ハイパーバイザー環境、または KVM ハイパーバイザー環境の複数のハイパーバイザーをモニターするために、このエージェントを複数インスタンス作成できます。さまざまな仮想マシンにまたがって、仮想化ワークロードをモニターし、リソース・キャパシティーを分析できます。エージェントを KVM ハイパーバイザー環境の仮想マシンに接続するには、前提条件である `libvirt*.rpm` および Korn シェル・インタープリター (pdksh) をインストールする必要があります。このエージェントは、仮想マシンを管理する libvirt ハイパーバイザーにリモートで接続することで、メトリックを収集します。

Microsoft SQL Server エージェント

Microsoft SQL 環境をモニターするには、Monitoring Agent for Microsoft SQL Server をインストールする前に、Microsoft SQL Server および Microsoft SQL ODBC ドライバーがインストールされている必要があります。例えば、ODBC ドライバーを Red Hat Enterprise Linux にインストールするには、以下のコマンドを使用します。

```
sudo yum install unixODBC
sudo yum install msodbcsql17
```

前提条件チェッカーの実行を完了するには、Cloud Application Performance Management バージョン 8.1.4.0 サーバー暫定修正 15 (8.1.4.0-IBM-APM-SERVER-IF0015.tar) 以降でエージェントを構成する必要があります。

MongoDB エージェント

MongoDB データベース・サーバーがインストールされているシステムに、MongoDB エージェントをインストールして構成する必要があります。

MySQL エージェント

MySQL 環境をモニターするには、Monitoring Agent for MySQL をインストールする前に MySQL サーバーおよび MySQL JDBC ドライバーがインストールされている必要があります。例えば、JDBC ドライバーを Red Hat Enterprise Linux にインストールするには、以下のコマンドを使用します。

```
yum install mysql-connector-java
```

エージェントのインストールを開始し、MySQL パッケージ名の前提条件検査を実行する際に、Red Hat 以外のプロバイダー (Oracle など) を使用すると、警告が表示される場合があります。MySQL サーバーと JDBC ドライバーが使用可能な場合、この警告が原因でインストールが失敗することはないため、この警告メッセージは無視してかまいません。 出力例を以下に示します。

```
Scenario: Prerequisite Scan
KSE - Monitoring Agent for MySQL Prerequisite Checker [version 01000005]:
Property                       Result Found       Expected
========                       ====== =====       ========
os.package.mysql-server          WARN Unavailable mysql-server-5.1.12+
os.package.mysql-connector-java WARN Unavailable mysql-connector-java-5.
1.12+
```

IBM Node.js エージェント

モニター対象アプリケーションを実行するために使用する Node.js のバージョンは、デフォルトのインストール済みバージョンと同じである必要があります。

現在、Node.js v5 はサポートされません。

OpenStack エージェント

OpenStack エージェントを使用する前に、エージェントをインストールしたサーバーに以下のソフトウェアが必要です。

- Python 2.6.0 以降または Python 2.7.0 以降
- 最新の OpenStack クライアント:

- OpenStack
- Keystone
- Neutron
- Swift

OpenStack コマンド行クライアントをインストールするには、『OpenStack コマンドラインクライアントのインストール』を参照してください。

- Python でのリモート・アクセス用の Paramiko ライブラリー。

注 : 新規の RedHat Linux サーバーに OpenStack エージェントをインストールする場合は、Paramiko ライブラリーをインストールする前に、以下のコマンドを実行して必須ソフトウェアをインストールします。

```
wget https://ftp.dlitz.net/pub/dlitz/crypto/pycrypto/pycrypto-2.6.1.tar.gz
yum install gcc/openssl-devel/libffi-devel
```

- KornShell

Oracle Database エージェント

Red Hat Enterprise Linux バージョン 5 とバージョン 6 および SUSE Linux Enterprise Server バージョン 11 とバージョン 12 x64 では、Oracle Database エージェントが Oracle データベースをリモートでモニターする場合は、最初に Oracle インスタント・クライアントをインストールする 必要があります。Oracle インスタント・クライアントは、Oracle Technology Network - Instant Client のダウンロード からインストールしてください。

インスタント・クライアント v10.x、v11.x、および v12.x は、Oracle Database エージェント によってサポートされます。

PHP エージェント

PHP アプリケーションをデプロイするために root ユーザーを使用した場合は、エージェントをインストール、構成、開始、または停止するために root ユーザーを使用する必要があります。PHP アプリケーションをデプロイするために root 以外のユーザーを使用した場合は、エージェントをインストール、構成、開始、または停止するために root ユーザーまたは同じ root 以外のユーザーを使用できます。

既存の WordPress のアプリケーションがインストールされていなければなりません。PHP エージェントがモニターするのは WordPress V3.7.1 以降です。

エージェントは、WordPress アプリケーションでの PHP 要求のパフォーマンスのみを評価します。CSS および JS のロードは評価されません。

エージェントは、URL を識別するために URL 引数を使用しません。

Python データ・コレクター

Python データ・コレクター は、Django アプリケーションをモニターします。

Response Time Monitoring エージェント

Response Time Monitoring エージェントをインストールする前に、インストール計画のセクション (688 ページの『インストールの計画』) を確認してください。

SAP HANA データベース・エージェント

1. Linux システムに SAP HANA データベース・クライアント HDBSQL バージョン 1.00.102.06 以降をインストールします。

   重要 : RHEL 5.x 64 ビット・オペレーティング・システムの場合は、Linux on x86_64 64bit ではなく Linux SUSE 9 on x86_64 64bit SAP HANA データベース・クライアントをインストールします。RHEL 6.x 以降の 64 ビット・オペレーティング・システムの場合は、Linux on x86_64 64bit SAP HANA データベース・クライアントをインストールします。

2. 以下のコマンドを実行して、インストール・ディレクトリーのパスを **LD_LIBRARY_PATH** 環境変数に追加します。

   `export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:`*`install_directory_path`*

例: `export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/sap/hdbclient`。ここで、`/usr/sap/hdbclient` は SAP HANA データベース・クライアントのインストール・パスを示します。

重要 :

SAP HANA データベース・クライアントのインストール・パスが **LD_LIBRARY_PATH** 環境変数に追加されていない場合、前提条件スキャナーによって FAIL の結果が返されます。

export コマンドを使用して追加された環境変数は、端末の特定のセッションに対してのみ保持されます。そのため、環境変数の追加に使用されたのと同じ端末からエージェント・インストール・スクリプトを実行するようにします。

Synthetic Playback エージェント

Synthetic Playback エージェントをインストールするには、オペレーティング・システム・ユーザーに以下の権限が必要です。

- インストール・イメージの読み取りと実行の権限が有効になっている
- エージェント・ホームに対する書き込み権限が有効になっている

Synthetic Playback エージェントを実行するには、オペレーティング・システム・ユーザーに以下の権限が必要です。

- エージェントのインストール・ロケーションおよびそのサブディレクトリーとファイルに 対する読み取り、書き込み、および実行の権限が有効になっている。
- Mozilla Firefox を実行する権限が有効になっている。
- Mozilla Firefox の実行バイナリーが、ユーザーのプロファイルの PATH 環境変数のパスに確実に存在する。

Synthetic Playback エージェントをインストールする前に、以下のステップを実行する必要があります。

1. Cloud APM コンソールでエージェントのインストール場所を同期します。
2. Mozilla Firefox および Xvfb 表示サーバーをインストールします。
3. Xvfb 表示サーバーが機能していることを確認します。次のコマンドを実行します。

   ```
   # Xvfb -ac
   ```

   エラーが出力されないはずです。
4. Xvfb プロセスが実行中であることを確認します。次のコマンドを実行します。

   ```
   # ps -ef|grep Xvfb
   ```

   出力例を以下に示します。

   ```
   root 7192 1 0 Jan14 ? 00:00:14 Xvfb -ac
   root 20393 17900 0 02:05 pts/0 00:00:00 grep -i xvfb
   ```
5. Xvfb プロセスを停止します。次のコマンドを実行します。

   ```
   # kill -9 7192
   ```
6. *install_dir*/etc/hosts にナビゲートし、hosts ファイルの先頭を編集して、以下のパラメーターを組み込みます。

   ```
   127.0.0.1 localhost
   ```

   次に、hosts ファイルを保存して閉じます。

WebSphere Applications エージェント
エージェントのインストール前に、Linux システム上で *ulimit* が **524000** に設定されていることを前提条件チェッカーが検査します。*max data segment size* システム環境変数が確実に **524000** に設定されるように、**ulimit -d 524000** コマンドを実行する必要があります。

エージェントのインストール

管理対象システムにはモニター・エージェントを自由に組み合わせてインストールできます。例えば、Ruby On Rails アプリケーションをモニターするために Ruby エージェントをインストールする場合に、Response Time Monitoring エージェント、Linux OS エージェント、またはその両方のエージェントもインストールしたい場合があります。Response Time Monitoring エージェントでは、Ruby アプリケーションの応答時間についてさらに多くの情報を収集できます。Linux OS エージェントでは、システムのその他の状況 (CPU 全体、メモリー、ディスクなど) をモニターすることができます。

インストール可能なモニター・エージェントの種類はオファリングによって決まります。それぞれのオファリングに含まれるエージェントのリストについては、54 ページの『機能』を参照してください。

Linux システムで稼働するエージェントのリストについては、130 ページの『Linux システムでのエージェントのインストール』を参照してください。

始める前に

エージェントをダウンロードします。105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

85 ページの『システム要件』の情報を参照して、インストールする予定のエージェントの要件を必ず満たしてください。

エージェントをインストールする前に、エージェントのプリインストール・タスクを確認してください。詳しくは、132 ページの『Linux システムでのプリインストール』を参照してください。

注 : Java ランタイムは、エージェントがそれを必要としている場合にのみインストールされるものであり、使用可能でない場合があります。また、ksh はエージェントのインストール用には不要になりましたが、Cloud APM サーバーのインストール中にインストールされる 要約およびプルーニング・エージェントのインストールについては例外です。enforcing モードの SELinux がサポートされています。

このタスクについて

モニター・エージェントは、root ユーザーとしてインストールすることも、非 root ユーザーとしてインストールすることもできます。root 特権を持たないユーザーがモニター・エージェントをインストールしたい場合には、非 root ユーザーとしてエージェントをインストールできます。145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください。また、ホスト管理者が root ユーザーとしてモニター・エージェントを実行したくない場合にも、非 root ユーザーとしてエージェントをインストールできます。インストール・フローは root ユーザーの場合と同じです。

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Cloud Application Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェント・タイプをインストールすることはできません。 エージェントの共存について詳しくは、950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。

手順

1. Red Hat Enterprise Linux システムで端末シェル・セッションを開きます。
2. ご使用のシステムで、.tar ファイルをダウンロードしたディレクトリーに移動します。

   エージェントは、モニターするアプリケーションがインストールされているシステムにインストールする必要があります。必要に応じて、モニター対象のシステムにインストール・アーカイブ・ファイルを転送します。アーカイブ・ファイルにはエージェントとインストール・スクリプトが含まれています。

   要確認 : 旧バージョンのアーカイブ・ファイルがディレクトリーに存在しないことを確認してください。
3. ご使用のオファリングに応じて以下のコマンドを使用して、 インストール・ファイルを解凍します。

```
tar -xf ./installation files.tar
```

ここで、*installation files* はご使用のオファリングのインストール・ファイル名です。

インストール・スクリプトは、このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーに解凍されます。例えば、*offering*_Agent_Install_8.1.4.0 です。エージェント・バイナリーおよび構成関連ファイルは、そのディレクトリー内のサブディレクトリーに解凍されます。

4. このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーから、以下のインストール・スクリプトを実行します。

```
./installAPMAgents.sh
```

エージェントをサイレント・モードでインストールするには、147 ページの『エージェントのサイレント・インストール』を参照してください。

5. 個々のエージェント、エージェントの組み合わせ、またはすべてのエージェントのいずれをインストール対象とするかを指定します。
6. エージェントをインストールしているのか、アップグレードしているのかに応じて、以下のいずれかの手順を実行します。
   - エージェントをインストールしている場合、エージェント・インストール済み環境の異なるホーム・ディレクトリーを指定するか、以下の該当するデフォルト・ディレクトリーを使用します。
     - /opt/ibm/apm/agent
   - エージェントをアップグレードしている場合に、 エージェントのインストール済み環境のホーム・ディレクトリーの 入力を求めるプロンプトが出されたときは、 以前のバージョンのエージェントのインストール・ディレクトリーを入力してください。
     a. 以前のバージョンのエージェントが /opt/ibm/apm/agent ディレクトリーに存在する場合は、新しいインストール・ディレクトリーを指定する必要があります。次のステップで、エージェント構成を /opt/ibm/apm/agent ディレクトリーからマイグレーションするかどうかの 確認を求められます。
     b. 古いインストール・ディレクトリー (/opt/ibm/ccm/agent) から新しいインストール・ディレクトリー (例えば、/opt/ibm/apm/agent) にエージェント構成をマイグレーションすることを確認した場合、新しいインストール・ロケーションでエージェントを開始する必要があります。

        制約事項 : 前のバージョンのエージェントは、前のインストール・ロケーションで自動的に停止されますが、新規インストール・ロケーションで自動的に開始されることはありません。
     c. インストールが完了し、新規インストール・ディレクトリーでエージェントが 機能することを確認したら、前のバージョンのエージェントを /opt/ibm/ccm/agent ディレクトリーからアンインストールする必要があります。すべてのエージェントを削除する場合は、**/opt/ibm/ccm/agent/bin/smai-agent.sh uninstall_all** コマンドを実行します。
7. ご使用条件への同意を確認するメッセージが表示されたら、1 (ご使用条件に同意して続行する場合) または 2 (同意しない場合) を入力します。

   1 (同意) を入力すると、環境の前提条件スキャンが開始され、しばらくすると完了します。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ライブラリーの欠落やディスク・スペースの不足といった前提条件の欠落が原因で、インストールが停止します。失敗に対処してから、インストール・スクリプトを再度開始する必要があります。

   注 : 以下のメッセージが表示されてインストールが終了する場合、Server サービスが開始されているかどうかを確認してください (「スタート」->「管理ツール」->「サービス」)。開始されていない場合は、Server サービスを開始して、installAPMAgents.bat を再実行します。

```
This script [installAPMAgents.bat] must be run as Administrator.
```

8. 非 root ユーザー ID を使用してエージェントをインストールした場合、システム起動スクリプトを更新する必要があります (145 ページの『非 root ユーザーとしてのエージェントのインストール』を参照してください)。

9. インストールが完了してコマンド行が使用可能になったら、この手順のステップを繰り替えし、管理対象システムにモニター・エージェントを追加でインストールできます。

次のタスク

必要に応じて、以下のようにエージェントを構成します。モニター・エージェントで、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』に記載されている構成が必要な場合、またはデフォルト設定を確認する必要がある場合は、163 ページの『第 7 章 環境の構成』を参照してください。

- ご使用のファイアウォールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていないために、フォワード・プロキシーを使用する場合、エージェントの環境構成ファイルを編集する必要があります。説明については、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。
- 以前のバージョンからエージェントをアップグレードした場合は、Cloud APM コンソールにログインする前に実行する必要がある再構成タスクまたはマイグレーション・タスクを特定してください。それらのタスクについては、1139 ページの『エージェントのアップグレード』を参照してください。アップグレードした後、インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。

エージェントを始動するには、次のコマンドを実行します。

```
./name-agent.sh start
```

モニター・エージェント・コマンドの詳細 (使用する名前を含む) については、181 ページの『エージェント・コマンドの使用』を参照してください。自動で開始されるエージェントおよび手動で開始されるエージェントについて詳しくは、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。

アップグレードした後、インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。

エージェントを構成して開始した後で、エージェントが収集しているデータを表示します。

- ログインしていない場合は、977 ページの『Cloud APM コンソールの始動』の手順に従ってください。
- IBM Tivoli Monitoring ドメインの管理対象システムを Application Performance ダッシュボードに表示する場合、949 ページの『IBM Tivoli Monitoring V6.3 との統合』で説明しているタスクを実行します。

Windows システムでのエージェントのインストール

一部の Cloud APM モニター・エージェントは Windows システムにインストールできます。

以下のモニター・エージェントが Windows 64 ビット・システムでサポートされています。記載がある場合、これらのエージェントは Windows 32 ビット・システムでもサポートされています。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS Elastic Load Balancer
- Monitoring Agent for Azure Compute
- Monitoring Agent for Cassandra
- Monitoring Agent for Cisco UCS
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HTTP Server*
- Monitoring Agent for IBM Cloud
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for Internet Services*

- Monitoring Agent for MQ Appliance
- Monitoring Agent for JBoss
- Monitoring Agent for MariaDB
- Monitoring Agent for Microsoft Active Directory*
- Monitoring Agent for Microsoft Cluster Server*
- Monitoring Agent for Microsoft Exchange Server
- Monitoring Agent for Microsoft Hyper-V Server
- Monitoring Agent for Microsoft Internet Information Services
- Monitoring Agent for Skype for Business Server (以前の Microsoft Lync Server)*
- Monitoring Agent for Microsoft .NET
- Monitoring Agent for Microsoft Office 365
- Monitoring Agent for Microsoft SharePoint Server
- Monitoring Agent for Microsoft SQL Server*
- Monitoring Agent for MySQL
- Monitoring Agent for NetApp Storage
- Monitoring Agent for Oracle Database
- Monitoring Agent for PostgreSQL
- Monitoring Agent for RabbitMQ
- Monitoring Agent for SAP Applications
- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack
- Monitoring Agent for Siebel
- Monitoring Agent for Sterling Connect Direct
- Monitoring Agent for Sterling File Gateway
- Monitoring Agent for Sybase Server
- Monitoring Agent for Tomcat
- Monitoring Agent for VMware VI
- Monitoring Agent for WebLogic
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ
- Monitoring Agent for Windows OS*
- Response Time Monitoring エージェント*

* 64 ビットおよび 32 ビットの Windows システムのサポート対象。

Windows システムでのプリインストール

Windows システムにエージェントをインストールする 前に、必要なプリインストール・タスクを完了する必要があります。一部のプリインストール・タスクはエージェント固有であり、その他のタスクは複数のエージェントに適用されます。

注 : これらの要件は、Software Product Compatibility Reports で示された要件を補足するものです。

ご使用のエージェントの現行バージョンの要件および依存関係については、System requirements (APM Developer Center) にある Software Product Compatibility Reports へのリンクを参照してください。

すべてのエージェント

以下のプリインストール・タスクは、すべてのエージェントに適用されます。

接続のテスト

エージェントをインストールする前に、システムが Cloud APM サーバーと通信可能であることを確認してください。Cloud APM サーバーへの接続を検査する方法については、『ネットワーク接続』を参照してください。

ローカル・ドライブ上のコマンド・プロンプトからのインストール

インストール・スクリプトの開始には Windows コマンド・プロンプトを使用してください。インストール・スクリプトの開始に Windows PowerShell を使用しないでください。

インストール・ファイルをローカル・ディスクまたはマップされたネットワーク・ドライブにコピーしてから インストール・スクリプトを開始してください。インストール・スクリプトをネットワーク・ロケーションから開始しないでください。

インストール・スクリプトは新規のコマンド・プロンプトから開始してください。既存のコマンド・プロンプトでは環境変数の内容が更新されていない可能性があるため、 既存のコマンド・プロンプトからインストール・スクリプトを開始しないでください。

***CANDLEHOME* 環境変数の設定**

以前に ITM Agent Converter を使用して同じ管理対象システムにエージェントをインストールして 構成した場合は、*CANDLEHOME* 環境変数が、Agent Converter でエージェントをインストールしたディレクトリーに変更されています。ネイティブ Cloud APM エージェントをインストールして構成する前に、*CANDLEHOME* 環境変数を別のディレクトリーに設定する必要があります。そうしないと、ネイティブ Cloud APM エージェントを開始できません。

特定のエージェント

以下のプリインストール・タスクは、指定されたエージェントに適用されます。

DataStage エージェント

1. DSODBConfig.cfg ファイル内でパラメーターを有効にします。以下のステップを実行してください。
   a. 以下の場所にある DSODBConfig.cfg ファイルをエディターで開きます。

      ```
      infosphere_information_server_install_dir¥Server¥DSODB
      ```

   b. # 記号を削除して、以下のパラメーターのコメントを外します。

      ```
      MonitorLinks=1
      JobRunUsage=1
      ResourceMonitor=1
      DSODBON=1
      ```

   c. これらのパラメーターの値を編集して、1 に設定します。
2. エージェント・コンピューター上でメタデータ・リポジトリー構成に使用されているデータベースの JDBC ドライバーをコピーします。
   a. タイプ 4 JDBC 4 以降。例: db2jcc4.jar
   b. Oracle 用タイプ 4 JDBC ドライバー。例: ojdbc6.jar
   c. 以下の MS SQL 用 JDBC ドライバー。
      - Sqljdbc41.jar は、JRE 7 を必要とし、JDBC 4.1 API をサポートします。
      - Sqljdbc42.jar は、JRE 8 を必要とし、JDBC 4.2 API をサポートします。

IBM Integration Bus エージェント

IBM Integration Bus エージェントをインストールするユーザー ID は mqbrkrs ユーザー・グループに属している必要があります。

Internet Service Monitoring

Internet Service Monitoring について、APM サーバーに対してここから IBM Cloud Application Performance Management 8.1.4.0 コア・フレームワーク暫定修正 3 を適用し、エージェントを事前構成する必要があります。エージェントとブリッジ・モジュールは、ポート 9510 と 9520 を使用します。これらのポートが既に使用中の場合、インストールは進行しません。

注:

- 既存のユーザーの場合は、Windows 32 ビット・プラットフォーム上のエージェントを新しいバージョンにアップグレードするのではなく、Windows または Linux の 64 ビット・プラットフォームに Internet Service Monitoring エージェントをインストールすることをお勧めします。
- Internet Service Monitoring エージェントは、Windows 64 ビット・プラットフォーム上の Windows 2008 R2 をサポートしません。

MySQL エージェント

Monitoring Agent for MySQL の場合、対象のシステムに MySQL エージェント をインストールする前に、MySQL サーバーと MySQL JDBC ドライバーをインストールする必要があります。JDBC ドライバーをインストールするには、MySQL Connector/J JDBC ドライバーを参照してください。

Oracle Database エージェント

以下のシステムでは、Oracle Database エージェントが Oracle データベースをリモートでモニターする場合は、最初に Oracle インスタント・クライアントを Oracle Technology Network - Instant Client のダウンロードからインストールする必要があります。

- Windows Server 2012 64 ビット
- Windows Server 2012 R2 64 ビット
- Windows Server 2008 R2 Datacenter 64 ビット
- Windows Server 2008 R2 Enterprise 64 ビット
- Windows Server 2008 R2 Standard 64 ビット

インスタント・クライアント v10.x、v11.x、および v12.x は、Oracle Database エージェント によってサポートされます。

Response Time Monitoring エージェント

Response Time Monitoring エージェントをインストールする前に、インストール計画のセクション (688 ページの『インストールの計画』) を確認してください。

SAP HANA データベース・エージェント

1. Windows システムに SAP HANA データベース・クライアント HDBSQL バージョン 1.00.102.06 以降をインストールします。
2. SAP HANA クライアントのインストール・パスを **PATH** 環境変数に追加します。

   例: `C:¥Program Files¥sap¥hdbclient` を **PATH** 環境変数に追加します。ここで、`C:¥Program Files¥sap¥hdbclient` は SAP HANA データベース・クライアントのインストール・パスを示します。

Tomcat エージェント

1. エージェントがインストールされている Tomcat サーバーに Java SDK がインストールされている。
2. SDK パスが *PATH* 変数に直接追加されているか、エージェントをインストールする前に **set path** コマンドを使用して追加されている。
3. **JAR** コマンドが動作する。

エージェントのインストール

管理対象システムにはモニター・エージェントを自由に組み合わせてインストールできます。例えば、MySQL サーバーをモニターするために Monitoring Agent for MySQL をインストールする場合、Ruby アプリケーションの応答時間の詳細情報を収集するために、Response Time Monitoring エージェントをインストールすることもできます。また、システムのその他の状況 (CPU 全体、メモリー、ディスクなど) をモニターするために、Monitoring Agent for Windows OS をインストールすることもできます。

インストール可能なモニター・エージェントの種類はご使用のオファリングによって決まります。それぞれのオファリングに含まれるエージェントのリストについては、54 ページの『機能』を参照してください。

Windows システムで稼働するエージェントのリストについては、140 ページの『Windows システムでのプリインストール』を参照してください。

始める前に

エージェントをダウンロードします。105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

85 ページの『システム要件』の情報を参照して、インストールする予定のエージェントの要件を必ず満たしてください。

エージェントをインストールする前に、エージェントの前提条件を確認してください。詳しくは、140 ページの『Windows システムでのプリインストール』を参照してください。

このタスクについて

エージェント・インストール・スクリプトおよびエージェント・コマンドを実行するための十分な権限を備えていることを確認します。以下のいずれかのユーザー・アカウント・タイプを使用してログインする必要があります。

- デフォルトの Windows 管理者ユーザー・アカウント
- 管理者ユーザー・アカウント
- 管理者グループのメンバーであるユーザー ・アカウント
- ユーザー・アカウント (Active Directory サービスで管理者として登録されたもの)

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Cloud Application Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェント・タイプをインストールすることはできません。エージェントの共存について詳しくは、950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。

手順

Windows オペレーティング・システムがインストールされている VM およびシステムにモニター・エージェントをインストールするには、以下の手順を実行します。

1. ご使用のシステムで、圧縮ファイルをダウンロードしたディレクトリーに移動します。
2. モニター・エージェント・ソフトウェアをインストールする場所に、ご使用のオファリングのエージェント・インストール・ファイルを解凍します。
   .bat インストール・スクリプトは、このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーに解凍されます。例えば、*offering*_Agent_Install_8.1.4.0 などです。エージェント・バイナリーおよび構成関連ファイルは、そのディレクトリー内のサブディレクトリーに解凍されます。
3. 管理者としてコマンド・プロンプトを開きます。
   a)「**スタート**」メニューで、検索ボックスに「コマンド」と入力します。
   b) 表示されるリストから「**コマンド プロンプト**」を右クリックし、「**管理者として実行**」を選択します。
4. アーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーで、コマンド・プロンプトから管理者特権を使用してインストール・スクリプトを実行します。

   ```
   cd offering_Agent_Install_version
   installAPMAgents.bat
   ```

   制約事項 : WebSphere Applications エージェントの場合、管理者特権は、WebSphere Application Server のインストールに使用された特権と同じである必要があります。

   エージェントをサイレント・モードでインストールするには、147 ページの『エージェントのサイレント・インストール』を参照してください。
5. エージェントをインストールする場合、インストール・ディレクトリーの名前を指定します。

デフォルトのインストール・パスは、C:¥IBM¥APM です。インストール・ディレクトリーの名前は、80 文字以内にする必要があり、非 ASCII 文字、特殊文字、2 バイト文字を含めることはできません。パス内のディレクトリー名に使用できる文字は abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ _¥:0123456789()~-./ のみです。

注: 短いファイル名の作成 (*8dot3Name*) が無効のときに、パスのディレクトリー名にスペースが含まれている場合、インストールはサポートされません。

エージェントをアップグレードする 場合は、このステップをスキップしてください。 エージェントは以前のインストール・ディレクトリーにインストールされます。

6. ご使用条件への同意を確認するメッセージが表示されたら、1 (ご使用条件に同意して続行する場合) または 2 (同意しない場合) を入力します。

   1 (同意) を入力すると、環境の前提条件スキャンが開始され、しばらくすると完了します。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ライブラリーの欠落やディスク・スペースの不足といった前提条件の欠落が原因で、インストールが停止します。失敗に対処してから、インストール・スクリプトを再度開始する必要があります。

   注: 以下のメッセージが表示されてインストールが終了する場合、Server サービスが開始されているかどうかを確認してください (「スタート」->「管理ツール」->「サービス」)。開始されていない場合は、Server サービスを開始して、installAPMAgents.bat を再実行します。

   ```
   This script [installAPMAgents.bat] must be run as Administrator.
   ```

7. インストールが完了してコマンド・プロンプトが使用可能になったら、上記のステップを繰り返して、追加のモニター・エージェントをインストールします。

次のタスク

必要に応じてエージェントを構成します。モニター・エージェントの手動構成が必要かどうかを確認するには、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。構成の説明について、またはデフォルトの構成設定を検討する場合は、163 ページの『第 7 章 環境の構成』を参照してください。

新規エージェントをインストールする 前に、Windows インストーラーにより、製品のインストール場所で現在実行されているすべてのエージェントが一時的に停止されます。インストールが完了したら、インストーラーは停止されたすべてのエージェントを再始動します。インストーラーによる自動開始が行われないモニター・エージェントがある場合、それらをすべて手動で再始動する必要があります。

- ご使用のファイアウォールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていないために、フォワード・プロキシーを使用する場合、エージェントの環境構成ファイルを編集する必要があります。説明については、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。
- 以前のバージョンからエージェントをアップグレードした 場合は、Cloud APM コンソールにログインする前に実行する必要がある再構成タスクまたはマイグレーション・タスクを特定してください。それらのタスクについては、1139 ページの『エージェントのアップグレード』を参照してください。

エージェントを開始するには、以下のいずれかの方法に従ってください。

- 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Cloud Application Performance Management**」をクリックします。エージェントを右クリックしてから、「**スタート**」をクリックします。
- 以下のコマンドを実行します。

  ```
  name-agent.bat start
  ```

モニター・エージェント・コマンドの詳細 (使用する名前を含む) については、181 ページの『エージェント・コマンドの使用』を参照してください。自動で開始されるエージェントおよび手動で開始されるエージェントについて詳しくは、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。

アップグレードした後、インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。

エージェントを構成して開始した後で、エージェントが収集しているデータを表示します。

- ログインしていない場合は、977 ページの『Cloud APM コンソールの始動』の手順に従ってください。
- IBM Tivoli Monitoring ドメインの管理対象システムを Application Performance ダッシュボードに表示する場合、949 ページの『IBM Tivoli Monitoring V6.3 との統合』で説明しているタスクを実行します。

非 root ユーザーとしてのエージェントのインストール

root 特権を持たないユーザーがモニター・エージェントをインストールしたい場合には、非 root ユーザーとしてエージェントをインストールできます。また、ホスト管理者が root ユーザーとしてモニター・エージェントを実行したくない場合にも、非 root ユーザーとしてエージェントをインストールできます。インストール・フローは root ユーザーの場合と同じです。 非 root ユーザーとしてのインストールの後に、root ユーザーまたは sudo ユーザー・アクセス権限で **UpdateAutoRun.sh** スクリプトを実行します。

始める前に

コンピューター・システムを一意的に識別するために、Linux OS エージェントは、コンピューター・システム・ボードの Universal Unique Identifier (UUID)、メーカー、モデル、およびシリアル番号を識別する必要があります。この情報は、エージェントが Cloud APM コンソールでアプリケーションに追加される前に必要です。

コンピューター・システムの情報を取得するには、以下のエンティティーがコンピューター・システム上に存在するかどうかを確認します。

1. **/usr/bin/hal-get-property** コマンドがコンピューター・システムにインストールされ、hald プロセス (HAL デーモン) が実行されているかどうかを確認します。このコマンドがインストールされていない場合は、ステップ 2 に進みます。コマンドがインストールされている場合は、ステップ 2 およびステップ 3 をスキップします。注: OS のバージョンが Red Hat 7 である場合、hald プロセスは使用できません。
2. **/usr/bin/hal-get-property** コマンドがコンピューター・システムにインストールされていない場合は、`/sys/class/dmi/id/product_uuid` ファイルが存在していて、コンピューター・システム UUID を含んでいること、および Linux OS エージェントをインストールするユーザーにこのファイルへの読み取り権限があることを確認します。このファイルが存在しない場合、ステップ 3 に進みます。このファイルが存在する場合、ステップ 3 をスキップします。
3. **/usr/bin/hal-get-property** コマンドがインストールされず、`/sys/class/dmi/id/product_uuid` ファイルが存在しない場合、**hostname** コマンドまたは **hostnamectl** コマンドが完全修飾ホスト名を返すことを確認する必要があります。これらのコマンドが、ドメインのない短いホスト名を返す場合、**"hostname <fqhn>"** コマンドまたは **"hostnamectl set-hostname <fqhn>"** コマンドを入力して完全修飾ホスト名を設定する必要があります。ここで、*<fqhn>* は、完全修飾ホスト名で置き換えられなければなりません。

注 : Linux OS エージェントは、この情報を定期的に取得します。そのため、前のステップにおけるコマンドまたはファイルは、インストール後も、配置されたままになっている必要があります。

注 : 非 root として実行されている Linux OS エージェントでは、Docker のモニタリングはサポートされません。

手順

1. 130 ページの『Linux システムでのエージェントのインストール』および 124 ページの『UNIX システムでのエージェントのインストール』の説明に従って、モニター・エージェントを Linux システムまたは UNIX システムにインストールします。
2. オプション: 特定のユーザーとしてエージェントをインストールした場合に、それとは異なるユーザーとしてエージェントを構成したいときは、`./secure.sh` スクリプトを実行します。

./secure.sh スクリプトについて詳しくは、187 ページの『非 root ユーザーとしてのエージェントの構成』および『エージェント・インストール・ファイルの保護』を参照してください。

例: **./secure.sh -g db2iadm1**

3. オプション: 必要に応じて、モニター・エージェントを Linux または UNIX で構成します。163 ページの『第 7 章 環境の構成』を参照してください。
4. システム始動スクリプトを更新するには、以下のスクリプトを root ユーザー または sudo ユーザー・アクセス権限で実行します。*install_dir*/bin/UpdateAutoRun.sh

次のタスク

非 root ユーザーとしてエージェントをインストールした後で、同じユーザーとしてエージェントを構成する場合は、特別なアクションは必要ありません。選択されたユーザーとしてエージェントをインストールした後で、別のユーザーとしてエージェントを構成する場合は、187 ページの『非 root ユーザーとしてのエージェントの構成』を参照してください。

エージェントのインストールと構成を非 root ユーザーとして行ったときに、その同じユーザーとしてエージェントを開始する場合は、特にアクションは必要ありません。選択されたユーザーとしてエージェントをインストールおよび構成した後で、別のユーザーとしてエージェントを開始する場合は、1013 ページの『非 root ユーザーとしてのエージェントの開始』を参照してください。

エージェントのインストールとアップグレードには同じユーザー ID を使用してください。

UpdateAutoRun.sh スクリプトを root ユーザーとして実行すると、エージェントはオペレーティング・システムの再始動後に自動的に開始するように構成されます。エージェントをこのように動作させたくない場合、エージェントの自動開始を無効にできます。詳しくは、188 ページの『UNIX システムおよび Linux システムでのエージェントの 自動開始の無効化』を参照してください。

エージェント・インストール・ファイルの保護

非 root ユーザーとして Linux システムまたは UNIX システムにモニター・エージェントをインストールしたら、secure.sh スクリプトを実行して world 書き込み許可を削除し、正しいファイル所有権を設定することにより、エージェントのインストール済み環境を保護することができます。

始める前に

- インストール・ディレクトリーに対する読み取り、書き込み、および実行の許可を持っている必要があります。
- モニター・エージェントのインストールとエージェントの 構成処理がシステムで完了し、エージェントが正常に開始されている必要があります。
- 異なる複数のユーザー・アカウントでエージェントを実行する場合、それらは同じグループのメンバーでなければなりません (オプション -g を参照)。

このタスクについて

インストール済み環境内のファイル許可をロックするには、以下のステップを実行します。root パスワードを使用しないオプション、グループ名を指定するオプション、コマンドのヘルプを表示するオプションが用意されています。

手順

- *install_dir*/bin ディレクトリーで以下のコマンドを実行します。 使用法:

```
secure.sh [-g common_group] [-n] [-h]
```

  - 最も単純なモードで **./secure.sh** スクリプトを実行します。このスクリプトにより、world 書き込み許可が削除され、現在のユーザーとユーザー・グループがファイル所有者として設定されます。非 root ユーザーがこのスクリプトを実行した場合は、root パスワードの入力を求めるプロンプトが表示されます。

- 非 root ユーザーが **./secure.sh** スクリプトを -n オプション付きで実行した場合、root パスワードの入力を求めるプロンプトは表示されません。この場合は、このユーザーの特権を使用して、ファイル許可と所有権が変更されます。他のユーザーによって所有されているファイルがインストール・ディレクトリーに格納されている場合は、他のユーザーのファイルの許可と所有権を変更する特権を現行ユーザーが持っていないと、このモードは失敗することがあります。
- 特定のグループをグループ所有者として設定する場合、所有者は、-g オプションを指定し、そのオプションへの引数として有効なグループ名を指定する必要があります (『例』を参照)。
  secure.sh -g *common_group* を実行します。
  このコマンドにより、ファイルとディレクトリーの所有権が再帰的に変更されます。

  common_group グループがユーザーの 1 次グループでない場合、以下のコマンドを実行することで、ディレクトリー内に作成された新規ファイルのグループ所有者になるように *common_group* グループを設定できます。

  ```
  chmod g+s install_dir/bin/sub_dir
  ```

  ここで、*sub_dir* は任意のサブディレクトリーです (例えば、/opt/ibm/ccm/agent)。
- スクリプトのヘルプ情報を表示するには、–h オプションを指定して **./secure.sh** スクリプトを実行します。

タスクの結果

スクリプトを実行したユーザーまたは指定されたグループに属するユーザーだけが、インストール・ディレクトリーにアクセスできるようになります。

例

例えば、Alice というユーザーが「apmgroup」というシステム・グループのメンバーである 場合、Alice は以下のコマンドでこのグループを使用して、ファイルのグループ所有権を設定することができます。

```
./secure.sh -g apmgroup
```

このスクリプトを実行すると、グループの *install_dir* 内のすべてのファイルでグループが「apmgroup」と設定されます。

次のタスク

./secure.sh スクリプトを実行すると、次の許可がエージェントに設定されます。

```
rwx rwx ---
```

スクリプトを実行したら、エージェント・ファイルの許可を確認してください。例えば、Response Time Monitoring の場合、*install_dir*/*arch*/hu/lib/mod_wrt.so にあるファイルを確認します。これらのファイルの許可が正しく設定されていない場合は、許可を手動で更新してください。例えば、Response Time Monitoring エージェントの場合、次のようにします。

1. 次のように実行して許可を設定します。

   ```
   chmod g+rx $AGENT_HOME/bin/rt-agent.sh
   ```

2. 次のように実行してユーザーとグループを設定します。

   ```
   chown newuser:newgroup $AGENT_HOME/bin/rt-agent.sh
   ```

エージェントのサイレント・インストール

エージェントのサイレント・インストールでインストール時間を短縮します。サイレント・モードでモニター・エージェントをインストールするには、エージェント・インストール・イメージのアーカイブ・ファイルを IBM Marketplace Web サイトからダウンロードし、エージェント・インストール・ファイルを解

凍してから、サイレント応答ファイルを準備し、インストール・スクリプトをサイレント・モードで実行する必要があります。

始める前に

1. モニター・エージェントをインストールするための前提条件を確認し、エージェント・インストール・ファイルをダウンロードして解凍します。詳しくは、『UNIX システムでのエージェントのインストール』、『Linux システムでのエージェントのインストール』、または『Windows システムでのエージェントのインストール』を参照してください。
2. エージェントをインストールするためのサイレント 応答ファイルを準備するには、以下のステップを実行します。
   a. オファリングに応じたサイレント・インストール・ファイル *offering*_silent_install.txt を見つけ、そのファイルのコピーを作成してテキスト・エディターで開きます。
   b. ご使用条件のコメントを外します。
   c. 以下のいずれかのステップを実行して、インストールするエージェントを指定します。
      - インストールする個々のエージェントのコメントを外します。例:

        ```
        INSTALL_AGENT=os
        ```

        ```
        INSTALL_AGENT=ruby
        ```

      - INSTALL_AGENT=all のコメントを外して、すべてのエージェントをインストールします。
   d. AGENT_HOME のコメントを外して、エージェントをインストールするディレクトリーを 指定します。

      要確認 : Linux Linux システムでエージェントをアップグレードする 場合は /opt/ibm/apm/agent ディレクトリーを指定しないでください。
   e. Linux Linux システムでエージェントをアップグレードする 場合は、MIGRATE_CONF=yes のコメントを外します。
   f. このファイルを保存します。

手順

1. コマンド・ラインで、インストール・スクリプトを解凍したディレクトリーに移動して、以下のコマンドを実行します。

   ```
   cd offering_Agent_Install_version
   ```

2. オプション: このステップは、Solaris 10 の場合のみ必要です。Solaris 10 でインストール・スクリプトを実行する前に、ksh へのソフト・リンクを作成しておく必要があります。
   a) /bin/sh コマンドをバックアップします。

   ```
   mv /bin/sh /bin/sh.bkup_origin
   ```

   b) ksh コマンドへのソフト・リンクを作成します。

   ```
   ln -s /bin/ksh /bin/sh
   ```

   c) 結果が ksh を指し示すことを確認します。

   ```
   ls -l /bin/sh
   ```

3. 以下のインストール・コマンドを実行します。
   - Linux AIX

     ```
     ./installAPMAgents.sh -p path_to_silent_response_file
     ```

   - Windows

```
installAPMAgents.bat -p path_to_silent_response_file
```

Windows では、前提条件スキャナーがエージェントのインストール先のディスクのタイプを取得できない場合、エージェントのインストールは失敗します。この状態が発生した場合、インストール・ログ・ファイルの validDestLocation プロパティーに失敗の結果が示されます。この場合、インストール・コマンドに SKIP_PRECHECK=1 を追加して、前提条件スキャンをスキップすることができます。例:

```
installAPMAgents.bat -p path_to_silent_response_file SKIP_PRECHECK=1
```

注 : Windows 短いファイル名の作成 (*8dot3Name*) が無効のときに、パスのディレクトリー名にスペースが含まれている場合、インストールはサポートされません。

タスクの結果

エージェントがインストールされます。

次のタスク

エージェントを構成します。Linux システムと UNIX システムおよび Windows システムについて手順とコマンド表を参照してください。

前提条件スキャナーのバイパス

モニター・エージェントをインストールすると、ご使用の環境の前提条件スキャンが開始されます。このスキャンが完了するまでにはしばらく時間がかかります。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。一部のインストール・シナリオでは、警告メッセージを無視したり、前提条件検査を完全にバイパスしたりすることができます。

このタスクについて

失敗メッセージには WARN と FAIL の 2 つのレベルがあり、バイパスには以下の 2 つのレベルがあります。

- **IGNORE_PRECHECK_WARNING** 変数を設定すると、インストーラーが警告 (WARN) メッセージを無視します。
- **SKIP_PRECHECK** 変数を設定すると、インストーラーがすべての失敗メッセージを無視します。

エージェントのインストールが失敗し、前提条件チェッカーから警告 (WARN) を受け取った場合は、その警告を確認してください。インストールを続行する場合は、**IGNORE_PRECHECK_WARNING** を設定してインストールを再実行します。

テンプレートとして機能する仮想マシン・イメージが存在する環境では、インストールを開始する前に実行された前提条件スキャンは、最初のテンプレート・イメージ上でのみ実行することができます。ある VM イメージがスキャンをパスすると、そのイメージから作成された他の VM もスキャンをパスします。 同じイメージから作成された他の VM の前提条件検査をバイパスすると、時間を節約することができます。**SKIP_PRECHECK** 変数を設定して、インストールを再実行します。

SKIP_PRECHECK 設定は、IBM サポートまたは Software Product Compatibility Report でサポート対象と示されている新しいオペレーティング・システムを使用しているが、前提条件チェッカーがまだ更新されていない場合にも使用できます。まずエージェントのインストールを試行してログを確認し、失敗している項目がこの新規 OS だけであり、バイパスしようとしているのがその OS の項目だけであることを確認してください。これは、**SKIP_PRECHECK** を指定すると、前提条件チェックリストのすべての項目がバイパスされるためです。

警告メッセージを無視するか、前提条件スキャンをバイパスするには、インストール・ファイルをダウンロードして解凍した後に、以下の手順を実行します。

手順

モニター・エージェントのインストール先のシステムで、以下のいずれかのコマンドを入力します。

- 前提条件検査で発生した警告 (WARN) メッセージを無視する場合
  - Linux AIX `export IGNORE_PRECHECK_WARNING=1`
  - Windows `set IGNORE_PRECHECK_WARNING=1`
- 前提条件スキャンをバイパスする場合
  - Linux AIX `export SKIP_PRECHECK=1`
  - Windows `set SKIP_PRECHECK=1`

次のタスク
次回エージェントと前提条件スキャナーをインストールするときにデフォルト設定を復元するには、**IGNORE_PRECHECK_WARNING** または **SKIP_PRECHECK** 変数をオフにします。

- Linux AIX `unset IGNORE_PRECHECK_WARNING`
- Windows `set IGNORE_PRECHECK_WARNING=`

または

- Linux AIX `unset SKIP_PRECHECK`
- Windows `set SKIP_PRECHECK=`

エージェントのアンインストール

管理対象システムから 1 つのエージェントまたはすべてのエージェントをアンインストールします。

始める前に

複数インスタンス・エージェントの場合は、エージェントをアンインストールする前に、すべてのエージェント・インスタンスを削除する必要があります。そうしないと、エージェントの項目がレジストリーからクリアされません。インスタンスを削除するには、以下のコマンドを実行します。

- Windows *name*`-agent.bat remove` *instance_name*
- Linux AIX `./`*name*`-agent.sh remove` *instance_name*

ここで、*name* はエージェントの名前であり、*instance_name* はインスタンス名です。詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。複数インスタンス・エージェントのリストについては、117 ページの表 7 を参照してください。

以下のエージェントの場合、アンインストール手順を実行する前にエージェント固有のタスクを完了する必要があります。

- Monitoring Agent for HTTP Server の場合、`http.conf` ファイルの `Include` ステートメント (`"Include "/opt/ibm/apm/agent/tmp/khu/kvm65s2_8044.conf"` など) を削除してから IBM HTTP Server を再始動する必要があります。
- Monitoring Agent for Python の場合は、 エージェントをアンインストールする 前に *install_dir*`/lx8266/pg/bin/uninstall.sh` を実行してインジェクション・コードを削除してください。
- Monitoring Agent for PHP の場合は、 エージェントをアンインストールする 前に *install_dir*`/bin/lx8266/pj/lib/uninstall.`*instance_name*`.sh` を実行してインジェクション・コードを移動してください。
- Monitoring Agent for WebSphere Applications では、エージェントをアンインストールする 前に、モニター対象のすべてのサーバー・インスタンスに対するデータ・コレクターを構成解除する必要があります。152 ページの『WebSphere Applications エージェント: データ・コレクターの構成解除』の説明に従ってください。

  WebSphere Applications エージェントの場合は、エージェントのアンインストールに 使用されるユーザー ID が、データ・コレクターのホーム・ディレクトリー内の `logs` ディレクトリーと `runtime` ディレ

クトリー、およびそこに含まれているすべてのサブディレクトリーとファイルに対する完全な読み取りおよび書き込み許可を備えていることを確認してください。データ・コレクターのホーム・ディレクトリーは以下のとおりです。

  - Windows *install_dir*¥dchome¥7.3.0.14.08
  - Linux AIX *install_dir*/yndchome/7.3.0.14.08

- Node.js エージェントの場合、エージェントをアンインストールする 前に、モニター・プラグインを Node.js アプリケーションから削除する必要があります。161 ページの『Node.js エージェント: モニター・プラグインの削除』の説明に従ってください。
- Microsoft .NET エージェントの場合、エージェントをアンインストールする 前に、.NET アプリケーションからデータ・コレクターを削除する必要があります。162 ページの『Microsoft .NET エージェント: .NET データ・コレクターの削除』の説明に従ってください。
- IBM Integration Bus エージェントで、エージェント提供のユーザー出口を使用してブローカーに対してトランザクション・トラッキングを構成していた場合は、エージェントをアンインストールする 前にそのユーザー出口を削除する必要があります。の説明に従ってください。297 ページの『KQIUserExit ユーザー出口の削除』.
- Internet Service Monitoring の場合は、<candle_home>¥BIN に移動し、引数として uninstall を指定して ism-agent.bat ファイルを実行します。smai-agent.bat を使用してサーバーにあるすべてのモニター・エージェントをアンインストールする 場合は、最初に引数として uninstall を指定して ism-agent.bat を実行した後、smai-agent.bat を実行します。
- Monitoring Agent for SAP NetWeaver Java Stack の場合は、エージェントをアンインストールする 前に、次のコマンドを使用してすべての SAP NetWeaver Java Stack エージェント・インスタンスを停止します。
  - Windows sap_netweaver_java_stack-agent.bat stop instance_name

このタスクについて

Windows システム上の Oracle エージェントをアンインストールするには、必ずコマンド・プロンプトを使用する必要があります。

手順

1. 1 つ以上のモニター・エージェントがインストールされている VM またはシステムで、コマンド行を開始して、以下のバイナリー・ディレクトリーに移動します。
   - Linux AIX *install_dir*/bin
   - Windows *install_dir*¥BIN

   ここで、*install_dir* は 1 つ以上のモニター・エージェントのインストール・ディレクトリーです。
2. 特定のモニター・エージェントをアンインストールするには、エージェント・スクリプト名とアンインストール・オプションを入力します。*name* はエージェント・スクリプト名です。
   - Linux AIX ./*name*-agent.sh uninstall
   - Windows *name*-agent.bat uninstall

   エージェント・スクリプト名のリストについては、181 ページの『エージェント・コマンドの使用』を参照してください。

   要確認 : Monitoring Agent for Microsoft .NET の場合、管理者特権でコマンドを実行する必要があります。

   管理対象システムからモニター・エージェントがアンインストールされます。

   すべてのモニター・エージェントを個別にアンインストールした場合は、続いてフレームワークのファイルを削除します。次の処置を参照してください。
3. 確認プロンプト付きで管理対象システムからすべてのモニター・エージェントをアンインストールするには、 次のようにスクリプト名と全アンインストール・オプションを入力します。

- Linux AIX `./smai-agent.sh uninstall_all`
- Windows `smai-agent.bat uninstall_all`

確認プロンプトが表示されます。続行するには 1 を入力し、キャンセルするには 2 を入力します。
システムまたは VM からすべてのモニター・エージェントがアンインストールされます。

4. Linux AIX
Linux および UNIX で確認プロンプトなしで強制的にすべてのモニター・エージェントをアンインストールするには、次のようにスクリプト名および強制的全アンインストール・オプションを入力します。
`./smai-agent.sh uninstall_all force`

次のタスク

Monitoring Agent for HTTP Server の場合は、エージェントをアンインストールした後に以下のファイルを手動で削除する必要があります。

- `/tmp/khu_cps.properties`
- `/tmp/httpserver-disc.error`

Monitoring Agent for Python の場合:

1. pyc 構成ファイルを削除して、リストアした pyc ファイルが確実にバイナリーを生成するようにします。
2. Apache サーバーを再始動して、Apache プロセスでロードされたミドルウェアを削除します。

Monitoring Agent for Ruby の場合、診断データ・コレクターをアンインストールするには以下の手順を実行します。

1. アプリケーションのホーム・ディレクトリーにナビゲートして、その `Gemfile` を開き、ファイルから `gem 'stacktracer'` という行を削除します。
2. Ruby on Rails アプリケーションを再始動します。
3. 診断データ・コレクターをアンインストールします。以下を入力します。`gem uninstall` *`Gemfile`*
4. データ・コレクターの runtime ディレクトリーを削除します。このディレクトリーのデフォルトの場所は、*`install_dir`*`/install-images/kkm/dchome` です。

Monitoring Agent for Microsoft .NET の場合は、以下のステップを実行します。

1. 以下のオプションのいずれかを使用して、データ・コレクターの `dll` ファイルを削除します。
   - オペレーティング・システムをリブートする。
   - ファイル *`install_dir`*`¥qe¥bin64¥CorProfLog.dll` の削除を試みます。
     「使用中のファイル」ダイアログが表示されます。 そこに、現在実行中の .NET プロセスが示されます。
   - 各 .NET プロセスを再始動します。
2. .NET アプリケーションを再始動します。

WebSphere Applications エージェント: データ・コレクターの構成解除

データ・コレクターを構成解除する前に WebSphere Applications エージェントをアンインストールすると、エージェントのアンインストールに失敗します。アプリケーション・サーバー・インスタンスからデータ・コレクターを削除するには、手動、対話式ユーティリティー、サイレント構成解除プロセスのいずれかの方法で行います。

PMI リソース・モニターを使用してインスタンスをモニターしている場合、構成解除は使用可能になりません。 サーバーが使用可能な間、PMI データのモニターは続行します。

対話式でのデータ・コレクターの構成解除

データ・コレクターでモニターしなくなった アプリケーション・サーバー・インスタンスが 1 つ以上ある場合、 それらについてデータ・コレクターを構成解除できます。

始める前に

データ・コレクターの構成用のユーザー ID を使用して、データ・コレクターを構成解除します。このユーザー ID は、アプリケーション・サーバーのインストール用のユーザー ID でもあります。このユーザー ID が、データ・コレクター・ホーム・ディレクトリーおよびそのすべてのサブディレクトリーに 対する読み取り許可と書き込み許可を備えていることを確認してください。データ・コレクター・ホーム・ディレクトリーは以下のとおりです (ここで、*install_dir* は WebSphere Applications エージェントのインストール・ディレクトリーです)。

- Windows *install_dir*¥dchome¥7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

このタスクについて

構成解除ユーティリティー (unconfig.sh または unconfig.bat) は、データ・コレクターを構成解除するための、メニュー方式のコマンド・ラインユーティリティーです。

手順

1. データ・コレクターの構成に使用するユーザー ID としてシステムにログインします。
2. 以下の bin ディレクトリーにナビゲートします。
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
3. オプション: ユーティリティーを 開始する前に、Java ホーム・ディレクトリーのロケーションを 設定します。
   例:

   Linux AIX
   ```
   export JAVA_HOME=/opt/IBM/AppServer80/java
   ```

   Windows
   ```
   set JAVA_HOME=C:¥Progra~1¥IBM¥WebSphere¥AppServer80¥java
   ```
4. 以下のコマンドを実行して構成解除ユーティリティーを 開始します。

   Linux AIX
   ```
   ./unconfig.sh
   ```

   Windows
   ```
   unconfig.bat
   ```
5. このユーティリティーは、データ・コレクターがモニターしているすべてのサーバー・インスタンスを検索します。当該のアプリケーション・サーバー・インスタンスに対応する番号を入力してデータ収集を構成解除するか、 アスタリスク (*) を入力して すべてのアプリケーション・サーバー・インスタンスについてのデータ収集を構成解除します。サーバーのサブセットを指定するには、当該のサーバーを表す番号をコンマで区切って入力します。 例えば、1,2,3 です。

   要確認 :

   - スタンドアロン環境の場合、構成中はアプリケーション・サーバー・インスタンスが実行中である必要があります。(WebSphere Application Server Liberty のインスタンスは稼働している必要はありません)。
   - ネットワーク・デプロイメント環境では、ノード・エージェントとデプロイメント・マネージャーが稼働している必要があります。
6. ユーティリティーからは、WebSphere Application Server の現在の構成のバックアップを作成するかどうかを指定するよう、プロンプトが出されます。現在の構成のバックアップを作成するには、1 を入力します。 そうしない場合は、2 を入力して、ステップ 8 へスキップします。
7. 構成のバックアップを格納するディレクトリーを指定するよう、ユーティリティーからプロンプトが出されます。 構成のバックアップを格納するディレクトリーを指定するか、デフォルトのディレクトリーを受け入れます。

   ユーティリティーは、WebSphere ホーム・ディレクトリーと、バックアップが作成される対象の WebSphere プロファイルについて、その名前を表示します。

8. ユーティリティーは、指定した WebSphere Application プロファイルについて WebSphere グローバル・セキュリティーが有効かどうかを、以下のように示します。グローバル・セキュリティーが有効になっていない場合は、ステップ 10 までスキップします。
9. ユーティリティーから、クライアント・プロパティー・ファイルからセキュリティー設定を取得するかどうかの指定を求めるプロンプトが出されます。ユーティリティーが、該当するクライアント・プロパティー・ファイルからユーザー名とパスワードを取得できるようにするには 1 を入力し、ステップ 154 ページの『10』 へスキップします。 そうしない場合は、2 を入力して、ユーザー名とパスワードを入力します。

   データ・コレクターは、 RMI または SOAP プロトコルを使用して WebSphere 管理サービスと通信します。プロファイルでグローバル・セキュリティーが有効な場合は、 そのプロファイルについて IBM WebSphere Application Server 管理コンソールにログインする権限があるユーザーのユーザー ID とパスワードを指定する必要があります。あるいは、ユーザー名とパスワードを暗号化して、それらをクライアント・プロパティー・ファイルに保存してから、データ・コレクターを構成することもできます。 `sas.client.props` ファイル (RMI 接続の場合) または `soap.client.props` ファイル (SOAP 接続の場合) を使用する必要があります。

   現在の WebSphere 構成をバックアップするオプションを 選択した場合、ユーティリティーは 構成のバックアップを開始します。
10. ユーティリティーは、指定したアプリケーション・サーバー・インスタンスについて、 データ・コレクターの構成を解除します。データ・コレクターが正常に構成解除されたことを示す状況メッセージが表示されます。
11. データ・コレクターの構成解除が完了した後に、アプリケーション・サーバー・インスタンスを再始動します。

    データ・コレクターの構成は、アプリケーション・サーバー・インスタンスが再始動されると有効になります。 サーバー・インスタンスの PMI リソース・モニターは引き続き使用可能です。
12. オプション: データ・コレクターを構成解除した後にサーバー・インスタンスのリソース・モニターを使用する場合は、以下のコマンドを実行して、モニター・エージェントを再始動します。
    - Windows

      ```
      cd install_dir¥bin
      was-agent.bat stop
      was-agent.bat start
      ```
    - Linux AIX

      ```
      cd install_dir/bin
      ./was-agent.sh stop
      ./was-agent.sh start
      ```

タスクの結果
指定したアプリケーション・サーバー・インスタンスに対するデータ・コレクターが構成解除されます。

サイレント・モードでのデータ・コレクターの構成解除
データ・コレクターは、構成解除ユーティリティーをサイレント ・モードで使用して構成解除することができます。

始める前に

データ・コレクターの構成用のユーザー ID を使用して、データ・コレクターを構成解除します。このユーザー ID は、アプリケーション・サーバーのインストール用のユーザー ID でもあります。このユーザー ID が、データ・コレクター・ホーム・ディレクトリーおよびそのすべてのサブディレクトリーに 対する読み取り許可と書き込み許可を備えていることを確認してください。データ・コレクター・ホーム・ディレクトリーは以下のとおりです (ここで、*install_dir* は WebSphere Applications エージェントのインストール・ディレクトリーです)。

- Windows *install_dir*¥dchome¥7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

このタスクについて

データ・コレクターをサイレント・モードで構成解除する場合、まず構成オプションをプロパティー・ファイルに指定します。 サンプル・プロパティー・ファイル sample_silent_unconfig.txt が、構成解除ユーティリティーのパッケージに 付属しています。 このファイルは、データ・コレクター・ホーム・ディレクトリーの bin ディレクトリー内にあります。

手順

1. データ・コレクターの構成に使用するユーザー ID でシステムにログインします。
2. properties.txt ファイルに構成オプションを指定します。

   サイレント・モードでデータ・コレクターを構成解除するために、以下のプロパティーが使用可能です。

   WebSphere Application Server の接続の設定

   was.wsadmin.connection.host
   wsadmin ツールの接続先のホストの名前を指定します。

   WebSphere Application Server のグローバル・セキュリティー設定

   was.wsadmin.username
   IBM WebSphere Application Server 管理コンソールにログオンする権限があるユーザーのユーザー ID を指定します。このユーザーは、アプリケーション・サーバー上でエージェントの役割を持っている必要があります。

   was.wsadmin.password
   was.wsadmin.username プロパティーに指定されたユーザーに対応するパスワードを指定します。

   WebSphere Application Server の設定

   was.appserver.profile.name
   構成解除するアプリケーション・サーバー・プロファイルの名前を指定します。

   was.appserver.home
   WebSphere Application Server ホーム・ディレクトリーを指定します。

   was.appserver.cell.name
   WebSphere Application Server のセル名を指定します。

   was.appserver.node.name
   WebSphere Application Server のノード名を指定します。

   WebSphere Application Server 構成のバックアップ

   was.backup.configuration
   データ・コレクターの構成を解除する前に、WebSphere Application Server データ・コレクターの、現在の構成をバックアップするかどうかを指定します。 有効な値は、True と False です。

   was.backup.configuration.dir
   バックアップ・ディレクトリーのロケーションを指定します。

   WebSphere Application Server のランタイム・インスタンス設定

   was.appserver.server.name
   データ・コレクターを構成解除するアプリケーション・サーバー・プロファイル内のアプリケーション・サーバー・インスタンスを指定します。

   ヒント: サイレント応答ファイルには、このプロパティーの複数のインスタンスを含めることができます。
3. 以下のディレクトリーにナビゲートします。
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
4. 次のコマンドを実行します。

- Windows

```
unconfig.bat -silent path_to_silent_file
```

- Linux AIX

```
unconfig.sh -silent path_to_silent_file
```

5. データ・コレクターの構成解除が完了した後に、アプリケーション・サーバー・インスタンスを再始動します。

   データ・コレクターの構成は、アプリケーション・サーバー・インスタンスが再始動されると有効になります。 サーバー・インスタンスの PMI リソース・モニターは引き続き使用可能です。
6. オプション: データ・コレクターを構成解除した後にサーバー・インスタンスのリソース・モニターを使用する場合は、以下のコマンドを実行して、モニター・エージェントを再始動します。

   - Windows

   ```
   cd install_dir¥bin
   was-agent.bat stop
   was-agent.bat start
   ```

   - Linux AIX

   ```
   cd install_dir/bin
   ./was-agent.sh stop
   ./was-agent.sh start
   ```

手動によるアプリケーション・サーバー・インスタンスからのデータ・コレクター構成の削除

アプリケーション・サーバー・インスタンスからデータ・コレクター構成を手動で削除するには、wsadmin ツールを使用してアプリケーション・サーバーに接続できなければなりません。これは、WebSphere Application Server Network Deployment を使用し、かつ Deployment Manager が稼動している場合にのみ可能です。WebSphere Application Server が開始できない場合は、構成ユーティリティーの 実行時に取ったバックアップから WebSphere Application Server をリストアする必要があります。

このタスクについて

以下のいずれかの条件に適合する場合は、アプリケーション・サーバー・インスタンスからデータ・コレクター構成を手動で削除できます。

- Network Deployment 環境以外の環境で、アプリケーション・サーバー・インスタンスにデータ・コレクター構成を手動で追加してあり、データ収集を構成解除する場合。アプリケーション・サーバー・インスタンスが稼働している必要があります。
- Network Deployment 環境で、データ・コレクター構成を手動でアプリケーション・サーバー・インスタンスに追加してあり、データ収集を構成解除する場合。アプリケーション・サーバー上でノード・エージェントとデプロイメント・マネージャーが稼働している必要があります。
- Network Deployment 環境で、アプリケーション・サーバー・インスタンスをデータ収集用に手動で構成してあり、アプリケーション・サーバーが始動しない場合。アプリケーション・サーバー上でノード・エージェントとデプロイメント・マネージャーが稼働している必要があります。

手動で、あるいは構成ユーティリティーまたはマイグレーション・ユーティリティーで、スタンドアロン・アプリケーション・サーバー・インスタンスをデータ収集用に構成した場合で、アプリケーション・サーバーが始動しないときは、バックアップ構成を使用して WebSphere Application Server 構成をリストアする必要があります。詳しくは、880 ページの『バックアップからのアプリケーション・サーバー構成のリストア』を参照してください。

要確認 :

- WebSphere 管理ユーザーとして、データ・コレクターに関する WebSphere Application Server の構成を手動で変更する必要があります。

- データ収集のための WebSphere Application Server の手動変更は、経験豊かな WebSphere 管理者のみが行う必要があります。手動による構成変更で何らかのエラーが発生すると、アプリケーション・サーバーは開始されないことがあります。
- アプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターを手動で構成する場合は、構成解除ユーティリティーを 使用してデータ・コレクターを構成解除することはできません。

手順

手動でデータ・コレクター構成を削除するには、以下の手順を実行してください。

1. WebSphere 管理サーバー・コンソールにログインします。
2. 「**サーバー**」をクリックします。
3. 「**サーバー・タイプ**」を展開し、「**WebSphere Application Server**」を選択します。
4. サーバーの名前をクリックします。
5. 「構成」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**Java 仮想マシン**」 > 「**追加プロパティー: カスタム・プロパティー**」に移動します。
6. 以下の JVM カスタム・プロパティーをすべて削除します (存在する場合)。
   - `am.home`
   - `ITCAM.DC.ENABLED`
   - `TEMAGCCollector.gclog.path`
   - `com.ibm.tivoli.itcam.toolkit.ai.runtimebuilder.enable.rebuild`
   - `com.ibm.tivoli.jiti.injector.ProbeInjectorManagerChain. primaryInjectorFile`
7. データ・コレクターのために追加された JVM 引数を特定します。
   a) ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックします。
   b) 手動でアプリケーション・サーバーをデータ収集用に構成した場合は、手動で追加した JVM 引数を見つけます。
   構成ユーティリティーを 使用してデータ収集用にアプリケーション・サーバーを構成した場合は、引数 **AM_OLD_ARGS** と **AM_CONFIG_JVM_ARGS** の値を比較して、どちらの引数が構成ユーティリティーによって追加されたのかを判別します。
8. 「**サーバー**」 > 「**アプリケーション・サーバー**」をクリックし、該当するサーバー名を選択します。
9. 「構成」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**Java 仮想マシン**」に移動します。
10. 「**汎用 JVM 引数**」フィールドで、ステップ 7 で特定したデータ・コレクター用の JVM 引数を削除します。
11. 「**適用**」または「**OK**」をクリックします。
12. 「**メッセージ**」ダイアログ・ボックスで、「**保存**」をクリックします。
13. 「**マスター構成に保存**」ダイアログ・ボックスで、以下のステップのいずれかを行います。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」チェック・ボックスを選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
14. データ・コレクター用に追加された環境エントリーを削除します。
    a) 「構成」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**環境エントリー**」に移動します。
    b) オペレーティング・システムに応じて、以下の環境エントリーを削除します。
       - UNIX **LIBPATH**
       - Linux **LD_LIBRARY_PATH**
       - Windows **PATH**
    c) **NLSPATH** 環境エントリーを削除します。

15.「**適用**」または「**OK**」をクリックします。
16.「**メッセージ**」ダイアログ・ボックスで、「**保存**」をクリックします。
17.「**マスター構成に保存**」ダイアログ・ボックスで、以下のステップのいずれかを行います。
    - Network Deployment 環境の場合は、必ずチェック・ボックス「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
18. ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックします。
19. 以下の変数を削除します。
    - **AM_CONFIG_JVM_ARGS**
    - **AM_OLD_JVM_ARGS**
    - **ITCAMDCHOME**
    - **ITCAMDCVERSION**
20.「**メッセージ**」ダイアログ・ボックスで、「**保存**」をクリックします。
21.「**マスター構成に保存**」ダイアログ・ボックスで、以下のステップのいずれかを行います。
    - Network Deployment 環境の場合は、必ずチェック・ボックス「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
22. データ収集用のサーバー・インスタンスを、手動ではなくデータ・コレクター構成ツールで構成した場合には、以下の手順を実行します。
    a) *dc_home*/runtime ディレクトリーにナビゲートします。
    b) $profile.$cell.$node.$server.input.properties ファイルを $profile.$cell.$node.$server.input.properties.bak に名前変更します。
23. プロファイル内のすべてのアプリケーション・サーバー・インスタンスからデータ・コレクター構成を手動で削除する場合は、以下の手順を実行します。
    a) $appserverhome/bin ディレクトリーにナビゲートします。
    b) **osgiCfgInit.sh/bat -all** コマンド (Windows システムの場合) または **osgiCfgInit.sh -all** コマンド (UNIX システムおよび Linux システムの場合) を実行します。
24. データ・コレクターのモニター対象であるアプリケーション・サーバー・インスタンスを再始動します。

データ・コレクターの手動による構成解除

WebSphere Applications エージェント用にデータ・コレクターを手動で構成した後には、データ・コレクターを手動で構成解除する必要があります。

このタスクについて

以下の手順が適用されるのは、856 ページの『構成ユーティリティーが 失敗した場合のデータ・コレクターの手動構成』の説明に従ってデータ・コレクターを手動で構成した後のみです。構成ユーティリティーを使用してデータ・コレクターを構成した場合は、データ・コレクターを構成解除するには、同様に構成解除ユーティリティーを 使用する必要があります。説明については、152 ページの『対話式でのデータ・コレクターの構成解除』または154 ページの『サイレント・モードでのデータ・コレクターの構成解除』を参照してください。

手順

- WebSphere Application Server 用のデータ・コレクターを手動で構成解除するには、159 ページの『WebSphere Application Server traditional 用のデータ・コレクターの手動による構成解除』を参照してください。
- Liberty サーバー用のデータ・コレクターを手動で構成解除するには、160 ページの『WebSphere Application Server Liberty 用のデータ・コレクターの手動による構成解除』を参照してください。

WebSphere Application Server traditional 用のデータ・コレクターの手動による構成解除

手順

1. WebSphere の管理コンソールにアドミニストレーターとしてログインします。
2. ナビゲーション・ペインで、「**サーバー**」をクリックし、「**サーバー・タイプ**」を展開して、「**WebSphere Application Server**」を選択します。
3. アプリケーション・サーバーの名前をクリックします。
4. 「構成」タブの「**サーバー・インフラストラクチャー**」セクションで、「**Java 仮想マシン**」を展開し、「**プロセス定義**」をクリックします。
5. 「**追加プロパティー**」のセクションで、「**Java 仮想マシン**」をクリックします。
6. 「**汎用 JVM 引数**」フィールドの内容から以下の項目を削除します。

   ```
   -agentlib:am_ibm_16=${WAS_SERVER_NAME} -Xbootclasspath/p:${ITCAMDCHOME}/
   toolkit/lib/bcm-bootstrap.jar -Djava.security.policy=${ITCAMDCHOME}/itcamdc/
   etc/datacollector.policy -verbosegc
   ```

7. 「**適用**」をクリックし、「**保存**」をクリックします。 「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
   - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
   - Network Deployment 環境でない場合は、「**保存**」をクリックします。
8. ナビゲーション・ペインで「**サーバー**」をクリックし、「**サーバー・タイプ**」を展開し、「**WebSphere Application Server**」をクリックしてから、サーバー名をクリックします。
9. 「構成」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**環境エントリー**」に移動します。
10. オペレーティング・システム、ハードウェア・プラットフォーム、およびアプリケーション・サーバー JVM に応じて、以下の環境エントリーを削除します。
    - UNIX LIBPATH
    - Linux LD_LIBRARY_PATH
    - Windows PATH
11. ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックします。
12. *ITCAMDCHOME* 変数が存在する場合は削除します。
13. 「**適用**」をクリックし、「**保存**」をクリックします。 「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
14. アプリケーション・サーバー・インスタンスを再始動します。
15. エージェント・インストール・ディレクトリーの runtime ディレクトリーに移動し、*profile_name.cell_name.node_name.server_name*.manual.input.properties ファイルを削除します。
    - Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime/*profile_name.cell_name.node_name.server_name*.manual.input.properties
    - Windows *install_dir*¥dchome¥7.3.0.14.08¥runtime¥*profile_name.cell_name.node_name.server_name*.manual.input.properties

WebSphere Application Server Liberty 用のデータ・コレクターの手動による構成解除

手順

1. Liberty サーバー・ディレクトリーにナビゲートし、Liberty サーバー・インストール・ディレクトリー内の *server_name* ディレクトリーにある jvm.options ファイルを開きます。例えば、/opt/ibm/wlp/usr/servers/defaultServer です。
2. 以下のパラメーターを jvm.options ファイルから削除します。

```
-agentlib:am_ibm_16=server_name
–Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
-verbosegc
```

   ここで、*server_name* は Liberty サーバーの名前で、*dc_home* はデータ・コレクター・ホーム・ディレクトリーです。
3. server.xml ファイルを開き、以下の行を削除します。

```
<feature>webProfile-7.0</feature>
<feature>monitor-1.0</feature>
<feature>usr:itcam-730.140</feature>
```

4. server.env ファイルを開き、オペレーティング・システムに従って環境エントリーから以下のエントリー値を削除します。

表 *8.* 環境エントリー

プラットフォーム	環境エントリー名	環境エントリーの値
AIX R6.1 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
AIX R7.1 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
Solaris 10 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Solaris 11 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Linux x86_64 R2.6 (64 ビット JVM)	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/lx8266
Linux Intel R2.6 (32 ビット JVM)	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/li6263
Windows (32 ビット JVM)	PATH	/lib;*dc_home*/toolkit/lib/win32
Windows (64 ビット JVM)	PATH	/lib;*dc_home*/toolkit/lib/win64

5. Liberty サーバーを再始動します。
6. WebSphere Applications エージェント・インストール・ディレクトリー内の runtime ディレクトリーに移動し、*cell_name.node_name.server_name*.manual.input.properties ファイルを削除します。
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime/*cell_name.node_name.server_name*.manual.input.properties
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥runtime¥*cell_name.node_name.server_name*.manual.input.properties

Node.js エージェント: モニター・プラグインの削除

Node.js エージェント をアンインストールする 前に、Node.js アプリケーションからモニター・プラグインを削除する必要があります。

手順

1. Node.js アプリケーション・ファイルの先頭からデータ・コレクター・プラグインを削除します。
   - Node.js エージェントを V01.00.12.00 から V01.00.13.00 にアップグレードする 場合、以下の手順を実行します。
     - リソース・データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_index.js');
       ```

       ここで、*KNJ_NPM_LIB_LOCATION* は、npm パッケージのグローバル・インストール・ディレクトリーの `lib` フォルダーへのディレクトリーです。デフォルトのディレクトリーは、`/usr/local/lib` です。
     - リソース・データの収集および詳細診断データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_deepdive.js');
       ```

     - リソース・データの収集、詳細診断データの収集、およびメソッド・トレースの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_methodtrace.js');
       ```

   - Node.js エージェントを V01.00.10.00 から V01.00.13.00 にアップグレードする 場合、以下の手順を実行します。
     - リソース・データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_index.js');
       ```

       ここで、*install_dir* は Node.js エージェント のインストール・ディレクトリーです。
     - リソース・データの収集および詳細診断データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_deepdive.js');
       ```

     - リソース・データの収集、詳細診断データの収集、およびメソッド・トレースの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_methodtrace.js');
       ```

2. Node.js アプリケーションを再始動して、データ・コレクター・プラグインを無効にします。
   - 現在の Node.js エージェントのバージョンが V01.00.10.00 の場合、ここまでで、データ・コレクター・プラグインが正常に削除されました。
   - 現在の Node.js エージェントのバージョンが V01.00.12.00 の場合、次のステップに進みます。
3. *install_dir*`/lx8266/nj/bin` ディレクトリーから `./uninstall.sh` コマンドを実行して、以前のエージェント設定を削除します。

次のタスク

Node.js エージェント のアンインストールの 詳細については、150 ページの『エージェントのアンインストール』 を参照してください。

Microsoft .NET エージェント: .NET データ・コレクターの削除

Microsoft .NET エージェントのアンインストールの 前に、.NET アプリケーションから .NET データ・コレクターを削除する必要があります。

手順

1. データ・コレクターを登録抹消します。
   管理者として、以下を入力します。

   ```
   cd install_dir¥qe¥bin configdc unregisterdc
   ```

   ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。
2. .NET アプリケーションをすべて停止して、データ・コレクターを無効にします。
   `net stop was /y` と入力します
3. アンインストール後に .NET データ・コレクターが完全にクリーンアップされるように、以下の手順を実行します。
   a) コマンド・プロンプトで <APM_HOME>¥qe¥bin ディレクトリーに移動します。
   b) ProcListCaller.bat ファイルを実行します。
   c) <APM_HOME>¥qe¥logs ディレクトリーの CorProfAttach.Log ログ・ファイルを確認します。このログ・ファイルには、.NET DC プロファイラー・コンポーネントの接続先プロセスがリストされます。
   d) エージェントをアンインストールする 前に、CorProfAttach.Log ファイルにあるプロセスを停止します。
   e) プロセスがリストされない場合は、エージェントのアンインストールを 続行します。

次のタスク

Microsoft .NET エージェントをアンインストールします。 150 ページの『エージェントのアンインストール』を参照してください。

第 7 章 環境の構成

ご使用のモニター・エージェントで構成が必要な場合、またはエージェントのデフォルト設定を確認する必要がある場合は、そのエージェント用に用意されたステップを実行してください。

共通トピック

一部のトピックは、エージェントの構成とデータ・コレクターの構成に共通しています。

ネットワーク接続

エージェント・サーバー通信を確立するために、システムの Cloud APM サーバーへの接続をテストします。

通信を検証するには、Cloud APM データ・センターへの接続をテストします。ファイアウォール・ルールで 3 つの特定の IP アドレスおよびポート 443 からの戻りトラフィックが許可されることを確認するには、接続を検証する必要がある 3 つのデータセンターの IP アドレスを検索します。詳しくは、Application Performance Management Developer Center の「Data Center IP addresses (SaaS only)」を参照してください。**openssl** コマンドを使用して、エージェントがこれら 3 つの IP アドレスに接続できることを確認します。**openssl** コマンドの使用方法について詳しくは、『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。エージェントが接続できない場合は、ローカルの IT チームにお問い合わせください。チームでは、ファイアウォール・ルールを調整し、ポート 443 を有効にし、サーバーからの TLS 1.2 トラフィックを有効にしたり、Cloud APM サーバーに接続するようにプロキシー・サーバーを構成したりできます。

ご使用のファイアウォール・ルールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていない場合は、フォワード・プロキシーにトラフィックを送信するようにエージェントを構成できます。詳しくは、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。

ブラウザー接続

Cloud APM コンソールへのブラウザー接続を確認するには、サブスクリプションが用意されたときに IBM から提供された**起動** URL を見つけます。アカウントにサインインしてコンソールを開始することもできます。IBM Marketplace サブスクリプションの詳細を使用して、「**Products and Services**」 (http://ibm.biz/my-prodsvcs) ページにサインインします。「**起動**」をクリックしてコンソールを開始し、URL (例: `8b68ba1b9.agents.na.apm.ibmserviceengage.com`) を表示します。この URL を使用してコンソールにログインできることを確認してください。

セキュア通信

エージェントと Cloud APM サーバーの間のセキュア通信には TLS 1.2 が必要です。

エージェントと IBM クラウド内の Cloud APM サーバー の間の通信には、TLS 1.2 および FIPS Suite-B 暗号スイートによる HTTPS を使用します。以下の暗号を使用します。

```
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
```

ブラウザーと Cloud APM サーバーとの間の通信にも、TLS 1.2 が必要です。一部のブラウザーでは TLS 1.2 がデフォルトで有効になっていないため、手動で有効にする必要があります。

フォワード・プロキシーを使用して通信するためのエージェントの構成

ご使用のファイアウォール・ルールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていない場合は、トラフィックをフォワード・プロキシーに送信するように IBM モニター・エージェントを構成できます。KDH_FORWARDPROXY 環境変数を編集して、エージェントがフォワード・プロキシーを介して通信するように構成します。

© Copyright IBM Corp. 2014, 2019

始める前に

エージェントが接続する Cloud APM データ・センターの IP アドレスを判別するには、Data Center IP addresses (APM Developer Center) を参照してください。. その後、ファイアウォール・ルールを調整して、確認した IP アドレスにフォワード・プロキシーから要求を送信できるようにしてください。

openssl コマンドを使用すると、エージェントがインストールされているコンピューター・システムが、Cloud APM データ・センターの各サーバーと接続できるかどうかを確認することができます。**openssl** コマンドを使用して、Cloud APM で使用されている暗号スイートがネットワークでサポートされているかどうかを確認することもできます。**openssl** コマンドの結果、コンピューター・システムが接続できないことが示された場合は、フォワード・プロキシーをセットアップしなければならない可能性があります。コマンドの結果が Cloud APM サーバー証明書を取得できなかったことを示している場合は、ネットワーク・チームと協力して、必要な暗号スイートがサポートされていない理由を判別してください。Cloud APM で使用される暗号スイートのリストについては、163 ページの『セキュア通信』を参照してください。

以下の例に示すように、**openssl** コマンドを実行します。

```
echo quit | openssl s_client
-state -connect <domain-name>:443
-tls1_2 -cipher
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384
```

ここで、*domain-name* は Cloud APM のサブスクリプションのドメイン名 (例: 8b68ba1b9.agents.na.apm.ibmserviceengage.com) です。

サブスクリプションのドメイン名を調べるには、以下の手順を実行します。

1. テキスト・エディターで、以下のエージェント環境構成ファイルを開きます。

   Linux AIX /opt/ibm/apm/agent/config/global.environment

   Windows install_dir¥TMAITM6_x64¥K*pc*ENV (64 ビットの Windows システムの場合)、または install_dir¥TMAITM6¥K*pc*ENV (32 ビットの Windows システムの場合)。ここで、*pc* はエージェントの製品コードを示します。

   製品コードのリストについては、181 ページの『エージェント・コマンドの使用』を参照してください。

2. *IRA_ASF_SERVER_URL* 変数を探します。この値は、https://domain-name/ccm/asf/request の形式になっています。**openssl** コマンドでは、この値のドメイン名の部分を使用します。

接続が成功すると、次の例のようなメッセージが表示されます。

```
CONNECTED(00000003)
SSL_connect:before/connect initialization
SSL_connect:SSLv3 write client hello A
SSL_connect:SSLv3 read server hello A
depth=2 C = US, O = IBM Service Engage,
CN = ca_ec_384.ibmserviceengage.com
verify error:num=19:self signed certificate in certificate chain
verify return:0
SSL_connect:SSLv3 read server certificate A
SSL_connect:SSLv3 read server key exchange A
SSL_connect:SSLv3 read server certificate request A
SSL_connect:SSLv3 read server done A
SSL_connect:SSLv3 write client certificate A
SSL_connect:SSLv3 write client key exchange A
SSL_connect:SSLv3 write change cipher spec A
SSL_connect:SSLv3 write finished A
SSL_connect:SSLv3 flush data
SSL_connect:SSLv3 read finished A
---
Certificate chain
0 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
```

```
i:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN =ca_ec_384.apm.ibmserviceengage.com
1 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
2 s:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICkjCCAhegAwIBAgIIXlr284nLPaMwDAYIKoZIzj0EAwMFADCBhDELMAkGA1UE
BgwCVVMxGzAZBgNVBAoMEklCTSBTZXJ2aWNlIEVuZ2FnZTErMCkGA1UECwwiQXBw
bGljYXRpb24gUGVyZm9ybWFuY2UgTWFuYWdlbWVudDErMCkGA1UEAwwiY2FfZWNf
Mzg0LmFwbS5pYm1zZXJ2aWNlZW5nYWdlLmNvbTAeFw0xMzEyMDIxNjM2MDlaFw0y
MzEyMDExNjM2MDlaMIGGMQswCQYDVQQGDAJVUzEbMBkGA1UECgwSSUJNIFNlcnZp
Y2UgRW5nYWdlMSswKQYDVQQLDCJBcHBsaWNhdGlvbiBQZXJmb3JtYW5jZSBNYW5h
Z2VtZW50MS0wKwYDVQQDDCQqLmFnZW50cy5uYS5hcG0uaWJtc2VydmljZWVuZ2Fn
ZS5jb20wdjAQBgcqhkjOPQIBBgUrgQQAIgNiAAQmrGoCkAMoNAC3F6MIo1zR8fcO
mczYXtUux2bhlOibn3jQdxamhDR91nr2RBerGjMIITKNXd2MaOr3b6m8euk1BAL3
KsbN9lqvw94kXg0BTO1IHAcdsZQB+AuEVVhmDVGjUDBOMAwGA1UdEwEB/wQCMAAw
HwYDVR0jBBgwFoAU/zpE5TOnQ8LSuvbSWRfpbiGea08wHQYDVR0OBBYEFHL0At4O
GUdcOHVGg4Tfo4hl7LLGMAwGCCqGSM49BAMDBQADZwAwZAIwDWPHo5I04ZFVrkfk
St6gwH2UNF37jBscRN1lOE4SIwezZAqVs42BNMkWRjJBgiHzAjBm4m3zOjsXzNL8
+u8ALjQQCpBDT6dUHujzY5CRxG0xEHi5IXsXf4QwbctnjjvTeYA=
-----END CERTIFICATE-----
subject=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
issuer=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
---
Acceptable client certificate CA names
/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
/C=US/O=DigiCert Inc/OU=www.digicert.com/CN=DigiCert
Global Root CA/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
Server Temp Key: ECDH, prime256v1, 256 bits
---
SSL handshake has read 2659 bytes and written 261 bytes
---
New, TLSv1/SSLv3, Cipher is ECDHE-ECDSA-AES128-GCM-SHA256
Server public key is 384 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
Protocol : TLSv1.2
Cipher : ECDHE-ECDSA-AES128-GCM-SHA256
Session-ID:
A18C31D0B45A1166357C917E1CFCD86A9FBEDB4A0EB768EF5390AC28C95CB7EF
Session-ID-ctx:
Master-Key:
252B8FE2731E51AC0B79A27C7BED33CA8B15AF4CFD015C98DBACA46EA01DC40B
9E6B56E62E0F332FF6B56266B5ADD7B0
Key-Arg : None
Krb5 Principal: None
PSK identity: None
```

```
PSK identity hint: None
Start Time: 1510772474
Timeout : 7200 (sec)
Verify return code: 19 (self signed certificate in certificate chain)
---
DONE
SSL3 alert write:warning:close notify
```

コンピューター・システムが Cloud APM サーバーに接続できない場合、次の例のようなメッセージが表示されます。

```
getaddrinfo: Name or service not known
connect:errno=2
```

ネットワーク内のどこかで暗号スイートがブロックされているためにコンピューター・システムがサーバー証明書を取得できない場合、次の例のようなメッセージが表示されます。

```
SSL_connect:failed
---
no peer certificate available
---
No client certificate CA names sent
```

このタスクについて

フォワード・プロキシーが使用された場合、エージェントは最初にそのプロキシーとの TCP 接続を開きます。エージェントは、HTTP CONNECT 要求、およびターゲット・エンドポイント (Cloud APM サーバー) の URL をフォワード・プロキシーに送信します。次に、フォワード・プロキシーがターゲット・エンドポイントとの間に TCP 接続を確立し、エージェントと Cloud APM サーバーとの間の HTTPS トンネリング・セッションをセットアップします。

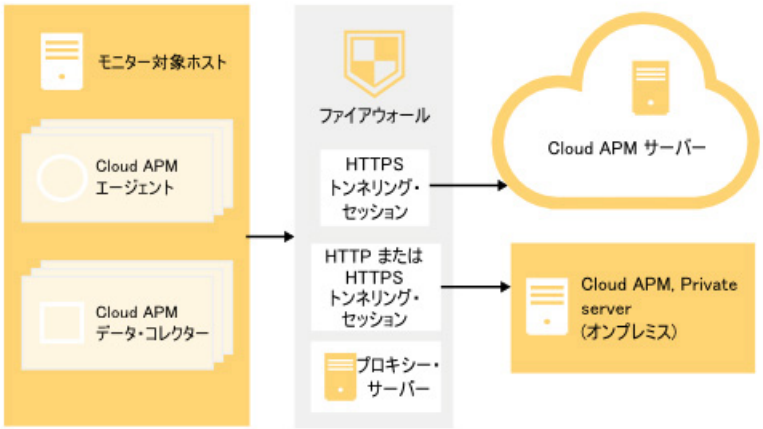


図 *1.* フォワード・プロキシー使用時の接続図

モニター・エージェントはプロキシーの認証はサポートしていません。つまり、エージェントは、構成されているプロキシー・ユーザー ID およびパスワードを使用することによる、フォワード・プロキシーへのログオンはサポートしていません。

手順

1. テキスト・エディターで、以下のエージェント環境構成ファイルを開きます。

   Linux AIX *install_dir*/config/global.environment ファイル。ここで、*install_dir* はエージェントのインストール済み環境のホーム・ディレクトリーです。global.environment ファイルで、インストール・ディレクトリーにあるすべてのエージェントの 構成を行います。

.global.environment ファイル内のカスタマイズされた設定は、エージェントのアップグレード後に失われます。設定した値を保持するには、global.environment ファイルでカスタマイズ変更を行います。このファイル内の設定値は、エージェントのアップグレードによって上書きされません。

Windows *install_dir*¥TMAITM6_x64¥K*pc*ENV ファイル (64 ビット・エージェントの場合)、および *install_dir*¥TMAITM6¥K*pc*ENV (32 ビット・エージェントの場合)。ここで、*pc* はエージェント製品コードです。エージェントごとに K*pc*ENV ファイルを構成します。

製品コードのリストについては、181 ページの『エージェント・コマンドの使用』を参照してください。

2. KDH_FORWARDPROXY 環境変数を編集して、プロキシーのアドレスとポートを指定します。

```
KDH_FORWARDPROXY=http://proxy-address:proxy-port-number
```

例:

```
KDH_FORWARDPROXY=http://HostA:8085
```

3. エージェントを再始動して変更内容を実装します。181 ページの『エージェント・コマンドの使用』を参照してください。

フォワード・プロキシーを使用して通信するためのデータ・コレクターの構成

ご使用のファイアウォール・ルールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていない場合は、フォワード・プロキシーにトラフィックを送信するようにデータ・コレクターを構成できます。APM_GW_PROXY_CONNECTION 環境変数を編集して、データ・コレクターがフォワード・プロキシーを介して通信するように構成します。

始める前に

データ・コレクターが接続する Cloud APM データ・センターの IP アドレスを判別するには、Data Center IP addresses (APM Developer Center) を参照してください。その後、ファイアウォール・ルールを調整して、確認した IP アドレスにフォワード・プロキシーから要求を送信できるようにしてください。

openssl コマンドを使用すると、データ・コレクターがインストールされているコンピューター・システムが、Cloud APM データ・センターの各サーバーと接続できるかどうかを確認することができます。Cloud APM で使用されている暗号スイートがネットワークでサポートされているかどうかを確認することもできます。**openssl** コマンドの結果、コンピューター・システムが接続できないことが示された場合は、フォワード・プロキシーをセットアップしなければならない可能性があります。コマンドの結果が Cloud APM サーバー証明書を取得できなかったことを示している場合は、ネットワーク・チームと協力して、必要な暗号スイートがサポートされていない理由を判別してください。Cloud APM で使用される暗号スイートのリストについては、163 ページの『セキュア通信』を参照してください。

以下の例に示すように、**openssl** を実行します。

```
echo quit | openssl s_client
-state -connect <domain-name>:443
-tls1_2 -cipher
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384
```

ここで、*domain-name* は Cloud APM のサブスクリプションのドメイン名です。.

サブスクリプションのドメイン名の確認方法については、163 ページの『フォワード・プロキシーを使用して通信するためのエージェントの構成』を参照してください。

接続が成功すると、次の例のようなメッセージが表示されます。

```
CONNECTED(00000003)
SSL_connect:before/connect initialization
SSL_connect:SSLv3 write client hello A
SSL_connect:SSLv3 read server hello A
depth=2 C = US, O = IBM Service Engage,
CN = ca_ec_384.ibmserviceengage.com
verify error:num=19:self signed certificate in certificate chain
verify return:0
```

```
SSL_connect:SSLv3 read server certificate A
SSL_connect:SSLv3 read server key exchange A
SSL_connect:SSLv3 read server certificate request A
SSL_connect:SSLv3 read server done A
SSL_connect:SSLv3 write client certificate A
SSL_connect:SSLv3 write client key exchange A
SSL_connect:SSLv3 write change cipher spec A
SSL_connect:SSLv3 write finished A
SSL_connect:SSLv3 flush data
SSL_connect:SSLv3 read finished A
---
Certificate chain
0 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN =ca_ec_384.apm.ibmserviceengage.com
1 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
2 s:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICkjCCAhegAwIBAgIIXlr284nLPaMwDAYIKoZIzj0EAwMFADCBhDELMAkGA1UE
BgwCVVMxGzAZBgNVBAoMEklCTSBTZXJ2aWNlIEVuZ2FnZTErMCkGA1UECwwiQXBw
bGljYXRpb24gUGVyZm9ybWFuY2UgTWFuYWdlbWVudDErMCkGA1UEAwwiY2FfZWNf
Mzg0LmFwbS5pYm1zZXJ2aWNlZW5nYWdlLmNvbTAeFw0xMzEyMDIxNjM2MDlaFw0y
MzEyMDExNjM2MDlaMIGGMQswCQYDVQQGDAJVUzEbMBkGA1UECgwSSUJNIFNlcnZp
Y2UgRW5nYWdlMSswKQYDVQQLDCJBcHBsaWNhdGlvbiBQZXJmb3JtYW5jZSBNYW5h
Z2VtZW50MS0wKwYDVQQDDCQqLmFnZW50cy5uYS5hcG0uaWJtc2VydmljZWVuZ2Fn
ZS5jb20wdjAQBgcqhkjOPQIBBgUrgQQAIgNiAAQmrGoCkAMoNAC3F6MIo1zR8fcO
mczYXtUux2bhlOibn3jQdxamhDR91nr2RBerGjMIITKNXd2MaOr3b6m8euk1BAL3
KsbN9lqvw94kXg0BTO1IHAcdsZQB+AuEVVhmDVGjUDBOMAwGA1UdEwEB/wQCMAAw
HwYDVR0jBBgwFoAU/zpE5TOnQ8LSuvbSWRfpbiGea08wHQYDVR0OBBYEFHL0At4O
GUdcOHVGg4Tfo4hl7LLGMAwGCCqGSM49BAMDBQADZwAwZAIwDWPHo5I04ZFVrkfk
St6gwH2UNF37jBscRN1lOE4SIwezZAqVs42BNMkWRjJBgiHzAjBm4m3zOjsXzNL8
+u8ALjQQCpBDT6dUHujzY5CRxG0xEHi5IXsXf4QwbctnjjvTeYA=
-----END CERTIFICATE-----
subject=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
issuer=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
---
Acceptable client certificate CA names
/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
/C=US/O=DigiCert Inc/OU=www.digicert.com/CN=DigiCert
Global Root CA/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
Server Temp Key: ECDH, prime256v1, 256 bits
---
SSL handshake has read 2659 bytes and written 261 bytes
---
New, TLSv1/SSLv3, Cipher is ECDHE-ECDSA-AES128-GCM-SHA256
Server public key is 384 bit
Secure Renegotiation IS supported
```

```
Compression: NONE
Expansion: NONE
SSL-Session:
Protocol : TLSv1.2
Cipher : ECDHE-ECDSA-AES128-GCM-SHA256
Session-ID:
A18C31D0B45A1166357C917E1CFCD86A9FBEDB4A0EB768EF5390AC28C95CB7EF
Session-ID-ctx:
Master-Key:
252B8FE2731E51AC0B79A27C7BED33CA8B15AF4CFD015C98DBACA46EA01DC40B
9E6B56E62E0F332FF6B56266B5ADD7B0
Key-Arg : None
Krb5 Principal: None
PSK identity: None
PSK identity hint: None
Start Time: 1510772474
Timeout : 7200 (sec)
Verify return code: 19 (self signed certificate in certificate chain)
---
DONE
SSL3 alert write:warning:close notify
```

コンピューター・システムが Cloud APM サーバーに接続できない場合、次の例のようなメッセージが表示されます。

```
getaddrinfo: Name or service not known
connect:errno=2
```

ネットワーク内のどこかで暗号スイートがブロックされているためにコンピューター・システムがサーバー証明書を取得できない場合、次の例のようなメッセージが表示されます。

```
SSL_connect:failed
---
no peer certificate available
---
No client certificate CA names sent
```

このタスクについて

フォワード・プロキシーが使用された場合、データ・コレクターは最初にそのプロキシーとの TCP 接続を開きます。データ・コレクターは、接続要求およびターゲット・エンドポイント (Cloud APM サーバー) の URL をフォワード・プロキシーに送信します。次に、フォワード・プロキシーがターゲット・エンドポイントとの間に TCP 接続を確立し、データ・コレクターと Cloud APM サーバーとの間の HTTPS トンネリング・セッションをセットアップします。

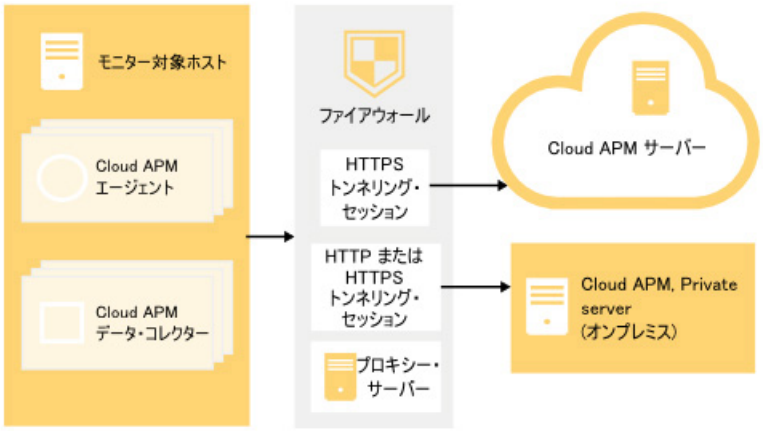


図 2. フォワード・プロキシー使用時の接続図

一部のデータ・コレクターは、プロキシー (例えば、Node.js、Liberty データ・コレクター) の認証をサポートします。データ・コレクターは、構成されているプロキシー・ユーザー ID およびパスワードを使用することによる、フォワード・プロキシーへのログオンはサポートしていません。

手順

1. Python データ・コレクターのフォワード・プロキシー通信を構成するには、以下のいずれかのステップを実行します。
   - テキスト・エディターで、データ・コレクターのプロパティー・ファイルである *<dc home>*/config.properties を開きます。ここで *<dc home>* は、データ・コレクターのインストール・ホーム・ディレクトリーです (例: /usr/lib/python2.7/site-packages/ibm_python_dc)。プロキシー・ホストおよびポート番号を使用して 変数を更新します (例えば、APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085)。このファイルで変数を編集すると、Python データ・コレクターが有効になっているすべてのアプリケーションが影響を受けます。

     注: 単一アプリケーションに対してフォワード・プロキシー通信を構成するには、*<dc home>*/config.properties ファイルをその単一アプリケーションのディレクトリーにコピーします 。アプリケーション・ディレクトリーで 変数を更新します。
   - Linux システム上で以下のコマンドを実行します。

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy host>:<http proxy port>
     ```

     以下に例を示します。

     ```
     export APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085
     ```
2. Node.js データ・コレクターに対してフォワード・プロキシー通信を構成するには、以下のいずれかのステップを実行します。
   - Linux システム上で以下のコマンドを実行します。

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy host>:<http proxy port>
     ```

     以下に例を示します。

     ```
     export APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085
     ```
   - Node.js データ・コレクターのフォワード・プロキシー・サーバーにアクセスするのにユーザー名とパスワードが必要な場合、Linux システムで以下のコマンドを実行します。

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy user>:
     <http proxy password>@<http proxy host>:<http proxy port>
     ```

     以下に例を示します。

     ```
     export APM_GW_PROXY_CONNECTION =http://Joe:passw0rd@9.181.138.247:8085
     ```
3. Liberty データ・コレクターに対してフォワード・プロキシー通信を構成するには、*<Liberty server home>*/jvm.options ファイルを編集します。ここで、*<Liberty server home>* は Liberty サーバーのホーム・ディレクトリーです (例えば、/opt/ibm/wlp/usr/servers/defaultServer/jvm.options)。以下のいずれかのステップを実行します。
   - 認証が不要の場合、jvm.options ファイルに以下のコードを追加します。

     ```
     -Dhttp.proxyHost=<http proxy host>
     -Dhttp.proxyPort=<http proxy port>
     -Dhttps.proxyHost=<https proxy host>
     -Dhttps.proxyPort=<https proxy port>
     -Djava.net.useSystemProxies=true
     ```

- フォワード・プロキシー・サーバーにアクセスするのにユーザー名とパスワードが必要な場合、jvm.options ファイルに以下のコードを追加します。

```
-Dhttp.proxyHost=<http proxy host>
-Dhttp.proxyPort=<http proxy port>
-Dhttp.proxyUser=<http proxy user>
-Dhttp.proxyPassword=<http proxy password>
-Dhttps.proxyHost=<https proxy host>
-Dhttps.proxyPort=<https proxy port>
-Dhttps.proxyUser=<https proxy user>
-Dhttps.proxyPassword=<https proxy password>
-Djava.net.useSystemProxies=true
```

4. 変更を適用するために、ローカル・アプリケーションを再始動します。

タスクの結果
フォワード・プロキシーを介して通信するようにデータ・コレクターを構成しました。

管理対象システム名

管理対象システム名 (MSN) は、ご使用の環境内で各 Cloud APM エージェントを一意的に識別するために使用されます。また、ナビゲーターの「**グループ**」セクションから各管理対象システムのグループを 選択するときに、Application Performance ダッシュボードに表示されるインスタンス名でもあります。 ご使用の環境内での競合を避けるために、エージェントに固有の MSN を割り当ててください。

エージェントの MSN の形式は、エージェント・タイプによって異なります。以下のいずれかのカテゴリーに分類されます。

- 171 ページの『単一インスタンス・エージェントの一般的な MSN の形式』
- 171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』
- 172 ページの『特殊な MSN の形式』

単一インスタンス・エージェントの一般的な MSN の形式

ほとんどの単一インスタンス・エージェントの場合、MSN の共通形式は次の形式になります。

```
hostname:pc
```

ここで、

- *hostname* は、エージェントのインストール先コンピューターの名前です。この部分は、必要に応じて変更できます。
- *pc* は、大文字の 2 文字のエージェント・コードであり、変更することはできません。エージェント・コードについて詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。
- : は、分離文字であり、変更することはできません。

例: `linuxhost01:LZ` は、Linux OS エージェントの MSN です。

この MSN の形式に従わない一部の単一インスタンス・エージェントは、172 ページの表 9 にリストされています。

MSN は 32 文字に制限されています。この MSN カテゴリーでは、エージェント・コードと分離文字を変更できないため、ホスト名に 29 文字を使用できます。

重要 : MSN の長さが 32 文字を超える場合は、MSN の一部が切り捨てられ、Cloud APM コンソールに正しく表示されません。例えば、ご使用のホスト名が `VeryLongSalesDivisionServerName03` の場合、管理対象システム名は `VeryLongSalesDivisionServerName03:PC` になります。ただし、これは `VeryLongSalesDivisionServerName0` に切り捨てられます。

複数インスタンス・エージェントの一般的な MSN の形式

ほとんどの複数インスタンス・エージェントの場合、MSN の共通形式は次の形式になります。

```
instancename:hostname:pc
```

ここで、

- *instancename* は、エージェントの構成中に指定するエージェント・インスタンス名です。この変数を使用して、各エージェント・ホスト・コンピューター上の各エージェント・タイプの各インスタンスに対して固有の MSN が割り当てられるようにします。

  要確認 :

  – エージェント・インスタンス名の作成には、英字 (a-z、A-Z)、アラビア数字 (0-9)、およびハイフンマイナス文字 (-) を使用できます。

  – エージェント・インスタンス名に下線文字 (_) は使用できません。

  – 指定するインスタンス名は、以下のように制限されます。

  - Linux AIX Linux システムまたは AIX システムでは、28 文字からホスト名を引いた長さ。

  - Windows Windows システムでの構成用にサイレント応答ファイルを使用する場合、28 文字からホスト名を引いた長さ。例えば、`Server-Name` の長さは 11 文字です。したがって、`Server-Name` ホスト上のエージェント・インスタンスは、長さが 17 文字以下でなければなりません。

  - Windows Windows システムで Cloud APM コンソール構成を使用する場合は、ホスト名が 8 文字を超えた分を 20 文字から引いた長さ。例えば、`TestServer` は長さ 10 文字であり、8 文字を 2 文字上回っています。 したがって、`TestServer` ホスト上のエージェント・インスタンスは、長さが 18 文字以下でなければなりません。

- *hostname* は、エージェントのインストール先コンピューターの名前です。必要に応じて、MSN のホスト名コンポーネントを変更できます。
- *pc* は、大文字の 2 文字のエージェント・コードであり、変更することはできません。エージェント・コードについて詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。
- `:` は、分離文字であり、変更することはできません。

例: `jboss1:win2016:JE` は、JBoss エージェントの MSN です。

この MSN の形式に従わない一部のマルチインスタンス・エージェントは、172 ページの表 9 にリストされています。

MSN は 32 文字に制限されています。この MSN カテゴリーでは、エージェント・コードと分離文字を変更できないため、インスタンス名とホスト名の間に 28 文字を使用できます。

重要 : MSN の長さが 32 文字を超える場合は、MSN の一部が切り捨てられ、Cloud APM コンソールに正しく表示されません。例えば、インスタンス名として `VeryLongInstanceName` を指定し、ご使用のサーバー名が `Production09` である場合、管理対象システム名は `VeryLongInstanceName:Production09:`*PC* になります。ただし、これは `VeryLongInstanceName:Production0` に切り捨てられます。

特殊な MSN の形式

特殊な MSN の形式は、MSN が上記の 2 つの一般的な MSN の形式に従わないエージェントに適用されます。これらのエージェントは、172 ページの表 9 にリストされます。

特殊な MSN は 32 文字に制限されています。172 ページの表 9 で、「MSN の形式」列のイタリックのストリングのみを変更できます。

表 *9.* 特殊な *MSN* の形式

エージェント	MSN の形式	MSN の例
Amazon EC2 エージェント	B5:*ec2subnodename*:INS	B5:sales:INS

表 9. 特殊な *MSN* の形式 (続き)

エージェント	MSN の形式	MSN の例
Amazon ELB エージェント	• AL:*instancename*A:APP • AL:*instancename*C:CLA • AL:*instancename*N:NET	• AL:elb-inst3A:APP • AL:elb-inst3C:CLA • AL:elb-inst3N:NET
Azure Compute エージェント	AK:*azure_compute_subnode_name*:AVM	AK:azc-inst3:AVM
Citrix VDI エージェント	VD:*citrixsitename*:XDS	VD:xds1:XDS
DataPower エージェント	BN:*datapowersystemname*:DPS	BN:datapower23:DPS
HTTP Server エージェント	HU:*hostname_alias*:HUS	HU:docker-ihs_httpd:HUS
IBM Integration Bus エージェント	*monitoredbrokername*:*agentID*:KQIB	TRADEBRK:AGT1:KQIB
MQ Appliance エージェント	MK:*hostname_sectionname*:ARM	MK:bvtmin_linux150:ARM
Node.js エージェント	NJ:*hostname_port*:NJA	NJ:KVM-014179_3000:NJA
Oracle Database エージェント	• RZ:*dbconnection-instancename-hostname*:ASM • RZ:*dbconnection-instancename-hostname*:DG • RZ:*dbconnection-instancename-hostname*:RDB	RZ:11g-oracledbdemo-GVT-1BL:RDB
Ruby エージェント	KM:*hostname_appname*:RAP	KM:nc9098036112_Blog:RAP
SAP エージェント	• SAP インスタンス: *instancename-hostname_sid_instancenumber*:Ins • SAP プロセス統合: *instancename-hostname*:PI • SAP Solution Manager: *instancename-hostname*:Slm • SAP システム: *instancename-hostname*:Sys	• PS5-IBMSAP3V1_PS5_11:Ins • PS5-IBMSAP3V1:PI • PS8-IBMSAP3V3:Slm • PS5-IBMSAP3V1:Sys
SAP HANA データベース・エージェント	• SAP Hana データベース: S7:*dbname-systemsid*:HDB • SAP Hana システム: *instancename*:*hostname*:S7	• S7:HNA-HNA:HDB • HNA:PS8760:S7
SAP NetWeaver Java Stack エージェント	• SAP NW Java AS クラスター: *instancename*:*hostname*:SV • SAP NW Java AS インスタンス: SV:*systemsid-jvmid*:NWJ	• J01:VPT02F17:SV • SV:J01-83309750:NWJ

表 9. 特殊な *MSN* の形式 (続き)		
エージェント	**MSN の形式**	**MSN の例**
UNIX OS エージェント	*hostname*:KUX	worklight17:KUX
WebLogic エージェント	WB:*instancename*:WLS	WB:Server1:WLS
WebSphere Applications エージェント	• WebSphere Application Server: *serveralias*:*hostname*:KYNS • WebSphere Portal Server: *serveralias*:*hostname*:KYNR • WebSphere Process Server: *serveralias*:*hostname*:KYNP	simpletrade:worklight17:KYNS
WebSphere MQ エージェント	*monitoredqmgrname*:*agentname*:MQ	TRADEQM:PoC:MQ
Windows OS エージェント	Primary:*hostname*:NT	Primary:TRADEIIS1:NT

エージェントの管理対象システム名の変更

管理対象システム名を変更する手順は、Cloud APM エージェントごとに異なります。エージェントによっては、管理対象システム名の変更は、その管理対象システム名に含まれるホスト名、エージェント・インスタンス名、またはそれら両方の変更を意味します。管理対象システム名を変更するために特定の手順が必要なエージェントもあります。

始める前に

171 ページの『管理対象システム名』で説明されている管理対象システム名の形式および命名上の制約について理解してください。

このタスクについて

ほとんどの Cloud APM エージェントでは、**CTIRA_HOSTNAME** パラメーターを使用して、管理対象システム名で使用されるホスト名を変更できます。マルチインスタンス・エージェントの管理対象システム名に含まれるエージェント・インスタンス名を変更するには、エージェント構成パラメーターを使用できます。エージェントを構成済みの場合は、再構成して、異なるエージェント・インスタンス名を割り当てる必要があります。エージェントを再構成した後は、以前のエージェント・インスタンスによって収集されたデータを取得できなくなります。

管理対象システム名のどの部分を変更するかによって、1 つの手順では管理対象システム名を変更できない場合があります。

対象のエージェントの管理対象システム名の変更方法を確認するには、174 ページの表 10 を参照してください。

例外 : 管理対象システム名の変更は、HTTP Server エージェント、Node.js エージェント、および Synthetic Playback エージェント ではサポートされません。

表 10. *Cloud APM* エージェントの管理対象システム名の変更方法	
エージェント	**管理対象システム名の変更方法**
Amazon EC2 エージェント	エージェント構成パラメーターを使用して管理対象システム名の EC2 サブノード名を変更します (200 ページの『Amazon EC2 エージェントの構成パラメーター』を参照)。
Amazon ELB エージェント	管理対象システム名を変更するには、新しいインスタンス名を使用して新規エージェント・インスタンスを作成します。

表 *10. Cloud APM* エージェントの管理対象システム名の変更方法 *(*続き*)*	
エージェント	**管理対象システム名の変更方法**
Azure Compute エージェント	エージェント構成パラメーターを使用して管理対象システム名のサブノード名を変更します (215 ページの『Azure Compute エージェントの構成パラメーター』を参照)。
Cassandra エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Cisco UCS エージェント	エージェント構成パラメーターを使用して、エージェント・インスタンス名を変更します。224 ページの『エージェントの構成パラメーター』を参照してください。
Citrix VDI エージェント	エージェント構成パラメーターを使用して、Citrix サイト名を変更します。234 ページの『Citrix VDI エージェントの構成パラメーター』を参照してください。
Db2 エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
DataPower エージェント	エージェント構成パラメーターを使用して、管理対象システム名を変更します。246 ページの『DataPower エージェントの構成』を参照してください。
DataStage エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Hadoop エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
HMC Base エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
IBM Cloud エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
IBM Integration Bus エージェント	295 ページの『IBM Integration Bus エージェントに対する固有の管理対象システム名の指定』

表 *10. Cloud APM* エージェントの管理対象システム名の変更方法 *(*続き*)*

エージェント	管理対象システム名の変更方法
JBoss エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Linux KVM エージェント	エージェントの構成パラメーターを使用します (477 ページの『Linux KVM のモニターの構成』を参照)。
Linux OS エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft .NET エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft Active Directory エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft Exchange Server エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft Hyper-V Server エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft IIS エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft Office 365 エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Microsoft SQL Server エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Microsoft SharePoint Server エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。

表 *10. Cloud APM* エージェントの管理対象システム名の変更方法 *(*続き*)*

エージェント	管理対象システム名の変更方法
MongoDB エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
MySQL エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
NetApp Storage エージェント	エージェントの構成パラメーターを使用します (585 ページの『NetApp ストレージのモニターの構成』を参照)。
OpenStack エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Oracle Database エージェント	エージェントの構成パラメーターを使用します (618 ページの『Oracle データベースのモニターの構成』を参照)。
PHP エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
PostgreSQL エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
RabbitMQ エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Response Time Monitoring エージェント	718 ページの『Response Time Monitoring エージェントの固有の管理対象システム名の指定』
Ruby エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。

表 *10. Cloud APM* エージェントの管理対象システム名の変更方法 *(*続き*)*

エージェント	管理対象システム名の変更方法
SAP エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
SAP HANA データベース・エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
SAP NetWeaver Java Stack エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Siebel エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Skype for Business Server エージェント (以前の Microsoft Lync Server エージェント)	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
Sterling File Gateway エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Sterling Connect Direct エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
Tomcat エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。

表 10. Cloud APM エージェントの管理対象システム名の変更方法 (続き)	
エージェント	**管理対象システム名の変更方法**
UNIX OS エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。
WebSphere Applications エージェント	管理対象システム名のホスト名を変更するには、How to change the host name used in managed system name for the WAS agent instance? を参照してください。 管理対象システム名のサーバー別名を変更するには、エージェントを再構成します。846 ページの『対話式でのデータ・コレクターの再構成』を参照してください。
WebSphere Infrastructure Manager エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。 エージェント構成パラメーターを使用して管理対象システム名のインスタンス名を変更します。
WebSphere MQ エージェント	939 ページの『複数のキュー・マネージャーに対する固有の管理対象システム名の指定』
Windows OS エージェント	**CTIRA_HOSTNAME** パラメーターを使用して管理対象システム名のホスト名を変更します。179 ページの『管理対象システム名のホスト名の変更』を参照してください。

管理対象システム名のホスト名の変更

このタスクについて

管理対象システム名のホスト名を変更することは、あまりありません。ホスト名は、エージェントの構成時に自動的に検出され、設定されます。管理対象システム名のホスト名は必要な場合にのみ変更し、指定する値が、管理対象システム名の命名上の制約によって切り捨てられないことを確認してください。

手順

1. エージェントの既存のインスタンスをすべて停止し、そのエージェントまたはそのサブノードがオフラインであると Cloud APM コンソールに表示されるまで待ちます。既存のエージェント・インスタンスがない場合は、次のステップに進みます。

   エージェント・インスタンスの停止について詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。
2. エージェントが単一インスタンス・エージェントの場合は、以下のステップを実行して **CTIRA_HOSTNAME** パラメーターを変更します。**CTIRA_HOSTNAME** パラメーターに指定する値は、すべての新規エージェント・インスタンスに適用される値です。

   a) 以下のファイルのバックアップ・コピーを作成します。

   - Linux AIX *install_dir*/config/*pc*.environment
   - Windows *install_dir*\TMAITM6_x64\k*pc*cma.ini

   ここで、

   - *install_dir* は、エージェントのインストール・ディレクトリーです。
   - *pc* は、2 文字のエージェント・コードです。エージェント名およびエージェント・コードの表を参照してください。

b) 以下のように **CTIRA_HOSTNAME** パラメーター値を変更して、ファイルを編集します。 ここで、*newhostname* は、エージェントがインストールされているコンピューターの実際のホスト名の代わりに使用されるカスタム・ストリングです。

- Linux AIX **CTIRA_HOSTNAME**=*newhostname*
- Windows **CTIRA_HOSTNAME**=*newhostname* .TYPE=REG_EXPAND_SZ

c) 変更を保存します。

3. エージェントがマルチインスタンス・エージェントの場合は、以下のステップを実行して **CTIRA_HOSTNAME** パラメーターを変更します。通常は、コンピューター上のすべてのエージェント・インスタンスが同じホスト名の値を使用します。エージェント・インスタンスごとに異なる値を使用する必要がある場合は、このステップの実行時に **CTIRA_HOSTNAME** に異なる値を割り当ててください。

a) 以下のファイルのバックアップ・コピーを作成します。

- Linux AIX *install_dir*/config/*pc_instance*.environment
- Windows *install_dir*/TMAITM6_x64/kpccma_*instance*.ini

b) ファイルを編集して、**CTIRA_HOSTNAME** パラメーター値を以下のように変更します。

- Linux AIX **CTIRA_HOSTNAME**=*newhostname*
- Windows **CTIRA_HOSTNAME**=*newhostname* .TYPE=REG_EXPAND_SZ

c) 変更を保存します。

4. Windows
既存のエージェント・インスタンスを再構成します。
5. すべてのエージェント・インスタンスを開始します。

次のタスク

エージェントの管理対象システム名を変更したら、Cloud APM コンソールを開始し、以前の管理対象システム名をアプリケーションから削除し、その代わりに新しいエージェント・インスタンスを追加することで、アプリケーションを変更します。

エージェントの構成

インストール後に、一部のエージェントは、自動的に構成されて開始されます。また、手動構成を必要とするが、自動的に開始されるエージェントもあります。一部のエージェントは手動で構成および開始する必要があります。複数インスタンス・エージェントでは、最初のインスタンスを作成し、手動で開始する必要があります。

始める前に

エージェントをインストールすると、/opt/ibm/apm/agent/samples ディレクトリーにサンプル・サイレント構成ファイル (例えば、ynv_silent_config_agent.txt、datapower_silent_config.txt) が配置されます。

注 : 一部のエージェント (例えば、Monitoring Agent for WebSphere Applications)には、データ・コレクターの構成などの各種タスクのための複数のサイレント構成ファイルがあります。

このタスクについて

エージェントの特定のデプロイメント詳細については、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。

エージェントを構成するために、この手順で説明するように、コマンド行またはサイレント応答ファイルを使用できます。

構成方法は、エージェントによって異なるため、ご使用のエージェント用に記載されている手順を使用してください。

手順

- *agent-name*.sh config コマンドを実行する。
  その他のコマンドについては、183 ページの表 12 と 185 ページの表 13 を参照してください。
- サイレント応答ファイルを編集してから、以下のいずれかのコマンドを実行する。
  - 単一インスタンス・エージェントの場合、以下のコマンドを実行します。

    ```
    agent-name.sh config response_file
    ```

  - 複数インスタンス・エージェントの場合、以下のコマンドを実行します。

    ```
    agent-name.sh config instance_name response_file
    ```

  - ここで、*instance_name* はインスタンス名です。これは、何をモニターするか示すために割り当てることができます。
- Windows
  Windows システムでサポートされるエージェントでは、「IBM Cloud Application Performance Management」ウィンドウを使用して特定の構成タスクを実行できます。「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。詳しくは、186 ページの『Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用』を参照してください。
- トランザクション・トラッキングまたはデータ収集の構成や診断データの有効化などの、特定のエージェントの拡張構成を実行するには、「**エージェント構成**」ウィンドウを使用します。詳しくは、186 ページの『「エージェント構成」ページ』を参照してください。

エージェント・コマンドの使用
モニター・エージェントをインストールするために使用するのと同じスクリプトを、インストール済みエージェントの状況の確認、エージェントの停止または開始、エージェントのアンインストールにも 使用できます。

このタスクについて
リファレンスとして、エージェント名とエージェント・コードを示します。

エージェント名は、次のコマンドで使用します。

Linux AIX *name*-agent.sh
Windows *name*-agent.bat

ここで、*name* は 181 ページの表 11 に示されているエージェントの名前です。

表 *11.* エージェント名およびエージェント・コード

モニター・エージェント	***name***	**2 文字のエージェント・コード**
Monitoring Agent for Amazon EC2	amazon_ec2	b5
Monitoring Agent for Azure Compute	azure_compute	ak
Monitoring Agent for Cassandra	cassandra	zc
Monitoring Agent for Cisco UCS	cisco_ucs	v6
Monitoring Agent for Citrix Virtual Desktop Infrastructure	citrix_vdi	vd
Monitoring Agent for DataPower	datapower	bn
Monitoring Agent for Db2	db2	ud
Monitoring Agent for Hadoop	hadoop	h8

表 *11.* エージェント名およびエージェント・コード *(*続き*)*

モニター・エージェント	***name***	**2** 文字のエージェント・コード
Monitoring Agent for HMC Base	hmc_base	ph
Monitoring Agent for HTTP Server	http_server	hu
Monitoring Agent for IBM Cloud	ibm_cloud	fs
Monitoring Agent for IBM Integration Bus	iib	qi
Monitoring Agent for MQ Appliance	ibm_mq_appliances	mk
Monitoring Agent for InfoSphere DataStage	datastage	dt
Monitoring Agent for JBoss	jboss	je
Monitoring Agent for Linux KVM	linux_kvm	v1
Monitoring Agent for Linux OS	os	lz
Monitoring Agent for MariaDB	mariadb	mj
Monitoring Agent for Microsoft Active Directory	msad	3z
Monitoring Agent for Microsoft Cluster Server	mscs	q5
Monitoring Agent for Microsoft Exchange Server	msexch	ex
Monitoring Agent for Microsoft Hyper-V Server	microsoft_hyper-v_server	hv
Monitoring Agent for Microsoft Internet Information Services	msiis	q7
Monitoring Agent for Skype for Business Server (以前の Microsoft Lync Server)	skype_for_business_server	ql
Monitoring Agent for Microsoft .NET	dotnet	qe
Monitoring Agent for Microsoft Office 365	microsoft_office365	mo
Monitoring Agent for Microsoft SharePoint Server	ms_sharepoint_server	qp
Monitoring Agent for Microsoft SQL Server	mssql	oq
Monitoring Agent for MongoDB	mongodb	kj
Monitoring Agent for MySQL	mysql	se
Monitoring Agent for NetApp Storage	netapp_storage	nu
Monitoring Agent for Node.js	nodejs	nj
Monitoring Agent for OpenStack	openstack	sg
Monitoring Agent for Oracle Database	oracle_database	rz
Monitoring Agent for PHP	php	pj
Monitoring Agent for PostgreSQL	postgresql	pn
Monitoring Agent for Python	python	pg
Monitoring Agent for RabbitMQ	rabbitMQ	zr
Monitoring Agent for Ruby	ruby	km
Monitoring Agent for SAP Applications	sap	sa

表 11. エージェント名およびエージェント・コード (続き)

モニター・エージェント	*name*	2 文字のエージェント・コード
Monitoring Agent for SAP HANA Database	sap_hana_database	s7
Monitoring Agent for SAP NetWeaver Java Stack	sap_netweaver_java_stack	sv
Monitoring Agent for Siebel	siebel	uy
Monitoring Agent for Sterling Connect Direct	sterling_connect_direct-agent	FC
Monitoring Agent for Sterling File Gateway	file_gateway	fg
Monitoring Agent for Sybase Server	sybase	oy
Monitoring Agent for Synthetic Playback	synthetic_transactions	sn
Monitoring Agent for Tomcat	tomcat	ot
Monitoring Agent for UNIX OS	os	ux
Monitoring Agent for VMware VI	vmware_vi	vm
Monitoring Agent for WebLogic	oracle_weblogic	wb
Monitoring Agent for WebSphere Applications	was	yn
Monitoring Agent for WebSphere Infrastructure Manager	wim	d0
Monitoring Agent for WebSphere MQ	mq	mq
Monitoring Agent for Windows OS	os	nt
Response Time Monitoring エージェント	rt	t5

手順

- Linux AIX

  モニター・エージェントにコマンドを送信するシステム上で、*install_dir*/bin ディレクトリーに移動します。 183 ページの表 12 の任意のコマンドを入力します。ここで、*name* は 181 ページの表 11 に示されているエージェントの名前です。

  表 12. *UNIX* システムおよび *Linux* システム用のコマンド

コマンド	説明
./*name*-agent.sh status	エージェントの状況を確認します。状況は、実行中か実行されていないかのいずれかです。エージェントが実行中の場合、エージェントと Cloud APM サーバー間の接続状況も確認されます。良好ではない接続状況には、「接続に失敗しました」、「エラーが検出されました」、「切断されました - エラー」があります。良好な状況には、予期される状況である「接続されました」があります。遷移中の状況は「接続中」です。「不明」状況は、ファイル・システムやエージェント・ログ・ファイルのエラーなどのため、エージェント状況を認識できなかったことを意味します。

表 12. *UNIX* システムおよび *Linux* システム用のコマンド (続き)

コマンド	説明
./*name*-agent.sh start	モニター・エージェントを開始します。エージェントにインスタンスがある場合は、コマンドの後ろにインスタンス名を入力してください。
./*name*-agent.sh stop	エージェントを停止します。エージェントにインスタンスがある場合は、コマンドの後ろにインスタンス名を入力してください。
./*name*-agent.sh prereqcheck	前提条件スキャンを実行します。このコマンド・オプションは、ほとんどのエージェントで使用可能です。
./*name*-agent.sh install	モニター・エージェントをインストールします。詳しくは、124 ページの『UNIX システムでのエージェントのインストール』および 130 ページの『Linux システムでのエージェントのインストール』を参照してください。
./*name*-agent.sh config *instance_name path_to_silent_config_file*	モニター・エージェントを構成します。*install_dir*/bin ディレクトリーからコマンドを実行し、必要に応じて応答ファイル・パスを追加します。 エージェントにインスタンスがある場合は、インスタンス名を入力してください。どのエージェントが複数インスタンス・エージェントであるかについて詳しくは、117 ページの表 7 を参照してください。 *silent_config_file* はオプションです。サイレント構成用のファイルを指定しない場合、プロンプトに従って対話式にモニター・エージェントを構成できます。
./*name*-agent.sh uninstall	モニター・エージェントをアンインストールします。詳しくは、150 ページの『エージェントのアンインストール』を参照してください。
./smai-agent.sh uninstall_all	管理対象システムからすべてのモニター・エージェントをアンインストールします。
./*name*-agent.sh remove *instance_name*	複数インスタンス・エージェントのインスタンスを削除します。
./*name*-agent.sh	このスクリプトで使用可能な関数の説明を表示します。

- Windows
モニター・エージェントにコマンドを送信するシステム上で、コマンド・プロンプトで *install_dir*¥BIN ディレクトリーに移動します。例: C:¥IBM¥APM¥bin。185 ページの表 13 の任意のコマンドを入力します。ここで、*name* は 181 ページの表 11 に示されているエージェントの名前です。

表 13. Windows システム用のコマンド

コマンド	説明
name-agent.bat status	エージェントの状況を確認します。 エージェントと Cloud APM サーバーの間の接続状況を確認します。良好ではない接続状況には、「接続に失敗しました」、「エラーが検出されました」、「切断されました - エラー」があります。良好な状況には、予期される状況である「接続されました」があります。遷移中の状況は「接続中」です。「不明」状況は、ファイル・システムやエージェント・ログ・ファイルのエラーなどのため、エージェント状況を認識できなかったことを意味します。
name-agent.bat start	モニター・エージェントを開始します。エージェントにインスタンスがある場合は、コマンドの後ろにインスタンス名を入力してください。
name-agent.bat stop	エージェントを停止します。エージェントにインスタンスがある場合は、コマンドの後ろにインスタンス名を入力してください。
name-agent.bat prereqcheck	前提条件スキャンを実行します。このコマンド・オプションは、ほとんどのエージェントで使用可能です。
name-agent.bat install	モニター・エージェントをインストールします。詳しくは、142 ページの『エージェントのインストール』を参照してください。
name-agent.bat config *instance_name path_to_silent_config_file*	モニター・エージェントを構成します。*install_dir*¥bin ディレクトリーからコマンドを実行し、必要に応じて応答ファイル・パスを追加します。 エージェントにインスタンスがある場合は、インスタンス名を入力してください。どのエージェントが複数インスタンス・エージェントであるかについて詳しくは、117 ページの表 7 を参照してください。 *silent_config_file* はオプションです。サイレント構成用のファイルを指定しない場合、プロンプトに従って対話式にモニター・エージェントを構成できます。
name-agent.bat uninstall	モニター・エージェントをアンインストールします。詳しくは、150 ページの『エージェントのアンインストール』を参照してください。
smai-agent.bat uninstall_all	管理対象システムからすべてのモニター・エージェントをアンインストールします。
name-agent.bat remove *instance_name*	複数インスタンス・エージェントのインスタンスを削除します。
name-agent.bat	このスクリプトで使用可能な関数の説明を表示します。

エージェント・バージョン・コマンド

- 環境内のエージェントのバージョンを表示するには、以下のコマンドを実行します。
  - Linux AIX

    ```
    install_dir/bin/cinfo
    ```

    バージョンを表示するには、1 を入力します。
  - Windows

    ```
    install_dir/InstallITM/kincinfo
    ```

関連タスク
186 ページの『Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用』

Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用
Windows でサポートされるエージェントには、エージェントの構成の実行および接続状況の確認に使用できる GUI ユーティリティーが用意されています。

Monitoring Agent for WebSphere MQ および Monitoring Agent for IBM Integration Bus では、GUI 構成ユーティリティーは使用できません。

手順

- 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Cloud Application Performance Management**」をクリックします。

タスクの結果
「IBM Cloud Application Performance Management」ウィンドウが表示されます。インストールされている各エージェント・コンポーネントが、その構成状況、開始しているのか停止しているのか、接続状況、バージョン番号などの情報とともにリストされます。

次のタスク
エージェントを右クリックし、オプションを選択することにより、エージェントを開始または停止したり、パラメーターを構成したりします。

「エージェント構成」ページ
Response Time Monitoring エージェントおよび WebSphere Applications エージェントなどのエージェントの設定を一元的に構成するには、「**エージェント構成**」ページを使用します。

一般的な使用法
ナビゲーション・バーから 「**システム構成**」 > 「**エージェント構成**」をクリックすると、タブ付きダッシュボードが表示されます。このダッシュボードには、構成可能なモニター・エージェントごとにタブが 1 つあります。表の列に、各管理対象システムの名前や IP アドレスなどの構成情報が、管理対象システムごとに 1 行表示されます。

アクション
「アクション」オプションは、トランザクション・トラッキングやデータ収集などの機能を有効または無効にするために使用します。

列のサイズ変更
列見出しの枠をドラッグして列幅を調整します。

列のソート
列見出し内をクリックすると、その列を基準にしたソートが行われます。同じ 列見出し内を再度クリックすると、ソートの昇順と降順が 切り替わります。

表フィルター
フィルター・テキスト・ボックス の内側をクリックして、表をフィルタリングするための 値の先頭部分を入力します。入力していくと、基準に合わない表の行がフィルターで除外され、検出行数に対する行の「**合計**」が更新されます。

フィルター・ボックス の「x」をクリックするか、Backspace キーを押すと、フィルターがクリアされます。

エージェント構成

特定のエージェントの設定について詳しくは、以下のトピックを参照してください。

- DataPower エージェント: 237 ページの『DataPower モニターの構成』
- IBM Integration Bus エージェント: 295 ページの『IBM Integration Bus エージェントのトランザクション・トラッキングの構成』
- Internet Service Monitoring451 ページの『Windows システムでのエージェントの構成』
- JBoss エージェント: 473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』
- Microsoft .NET エージェント: 530 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
- OS エージェントのログ・ファイル・モニター: 636 ページの『OS エージェント用のログ・ファイル・モニター構成の追加または削除』
- Response Time Monitoring エージェント: 693 ページの『「エージェント構成」ページを使用した構成』
- 地理位置情報: 714 ページの『エンド・ユーザー・トランザクションのロケーション値のカスタマイズ』
- Ruby エージェント: 726 ページの『Ruby アプリケーション用の診断データの無効化または有効化』
- SAP NetWeaver Java Stack エージェント: 769 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
- Tomcat エージェント: 807 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』
- WebLogic エージェント: 826 ページの『WebLogic エージェントのトランザクション・トラッキングの構成』
- WebSphere Applications エージェント: 877 ページの『「エージェント構成」ページでのデータ収集の動的な構成』
- WebSphere MQ エージェント: 941 ページの『WebSphere MQ エージェントのトランザクション・トラッキングの構成』

非 root ユーザーとしてのエージェントの構成

非 root ユーザーとしてエージェントを構成する場合は、システムに共通グループを作成し、各ユーザーをこのグループのメンバーにします。

始める前に

root ユーザーまたは非 root ユーザーとしてエージェントをインストールした後で、同じユーザーとしてエージェントを構成する場合は、特別なアクションは必要ありません。

選択されたユーザーとしてエージェントをインストールした後で、別のユーザーとしてエージェントを構成する場合は、システムに共通グループを作成します。すべてのエージェント管理ユーザーを、この共通グループのメンバーにします。すべてのエージェント・ファイルおよびディレクトリーの所有権をこのグループに移転します。

注 :

- HTTP Server エージェントの場合、非 root ユーザーとしてエージェントを構成する場合は、その非 root ユーザーが、IBM HTTP Server を始動したユーザーと同じユーザー ID を持っている必要があります。そうでない場合は、エージェントが IBM HTTP Server をディスカバーするときに問題が発生します。
- IBM Integration Bus エージェントの場合、IBM Integration Bus インストール済み環境が単一ユーザー・デプロイメントの場合は、IBM Integration Bus をインストールしたユーザーと同じユーザー ID を使用してエージェントを構成します。 エージェントを構成する前に、このユーザー ID に対して以下のステップを実行します。

手順

1. 130 ページの『Linux システムでのエージェントのインストール』および 124 ページの『UNIX システムでのエージェントのインストール』の説明に従って、モニター・エージェントを Linux または UNIX にインストールします。
2. 非 root ユーザーのグループ名を指定して `./secure.sh` スクリプトを実行することにより、ファイルを保護し、ファイルに対するファイル・グループ所有権を設定します。
   例: **`./secure.sh -g db2iadm1`**
3. 必要に応じて、モニター・エージェントを Linux または AIX で構成します。163 ページの『第 7 章 環境の構成』を参照してください。
4. システム始動スクリプトを更新するには、以下のスクリプトを root ユーザーまたは sudo ユーザー・アクセス権限で実行します。*`install_dir`*`/bin/UpdateAutoRun.sh`

次のタスク

`./secure.sh` スクリプトについて詳しくは、エージェント・インストール・ファイルの保護を参照してください。

エージェントのインストールとアップグレードには同じユーザー ID を使用してください。

UNIX システムおよび Linux システムでのエージェントの自動開始の無効化

UNIX および Linux システムでは、オペレーティング・システムの再始動後にエージェントを自動的に開始できます。システム再始動後にエージェントを自動的に開始したくない場合、エージェントの自動開始を無効にできます。

このタスクについて

UNIX または Linux システムでエージェントを root ユーザーとしてインストールする場合、システムの再始動後にエージェントを自動的に開始できます。また、非 root ユーザーとしてエージェントをインストールするが、インストール後に **`UpdateAutoRun.sh`** スクリプトを root として実行する場合も、システムの再始動後にエージェントを自動的に開始できます。

手順

1. 一部のエージェントの自動開始を無効にするには、以下のステップを実行します。
   a. Linux OS エージェントおよび WebSphere® Applications エージェントの場合は、`agent_install_dir/registry/kcirunas.cfg` ファイルに以下のコードを追加します。

```
<productCode>lz</productCode>
 <default>
   <autoStart>no</autoStart>
 </default>
 <productCode>yn</productCode>
 <default>
   <autoStart>no</autoStart>
 </default>
```

   b. **`agent_install_dir/bin/UpdateAutoRun.sh`** コマンドを実行します。
2. 一部のエージェントの自動開始を有効にするには、以下のステップを実行します。
   a. Linux OS エージェントおよび WebSphere® Applications エージェントの場合は、`agent_install_dir/registry/kcirunas.cfg` ファイルで *<autoStart>* タグの値を **yes** に変更します。
   b. `agent_install_dir/registry/AutoStart` ファイルを開き、その内容を調べます。
   c. `/etc/init.d/ITMAgents{$Num}` ファイルを削除します。ここで、*{$Num}* は、`agent_install_dir/registry/AutoStart` ファイルの正数です。値が 1 の場合は、`/etc/init.d/ITMAgents1` ファイルを削除する必要があります。
   d. **`agent_install_dir/bin/UpdateAutoRun.sh`** コマンドを実行します。

タスクの結果

システム再始動後は、エージェントを開始するエージェント・スクリプトは自動的に実行されなくなります。

データ・コレクターの一般的な構成手順

データ・コレクターを使用してアプリケーションの Cloud APM コンソールでモニター・データを表示するには、いくつかの構成タスクを実行する必要があります。

このタスクについて

以下の手順は、アプリケーションのモニターを構成するためのロードマップであり、必須のステップ、条件付きステップ、オプションのステップが共に含まれています。ニーズに従って、必要なステップを実行します。

手順

1. データ・コレクター・パッケージをダウンロードして解凍します。説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。
2. IBM Cloud アプリケーションとオンプレミス・アプリケーションに関するモニター・データを収集し、そのデータを Cloud APM サーバーに送信するようにデータ・コレクターを構成します。ご使用のアプリケーションのタイプに応じて、以下のタスクを 1 つ以上実行します。

   Liberty アプリケーション

   - 881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』
   - 886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』

   Node.js アプリケーション

   - 598 ページの『IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』
   - 603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』

   Python アプリケーション

   - 673 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成』
   - 678 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成』

   Ruby アプリケーション

   - 727 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクターの構成』

   Java アプリケーション

   - 455 ページの『J2SE のモニターの構成』
3. 鍵ファイルまたは Cloud APM サーバーが変更された場合は、データ・コレクターを Cloud APM サーバーに再接続します。説明については、190 ページの『Cloud APM サーバーへのデータ・コレクターの再接続』を参照してください。

次のタスク

必要な構成タスクをすべて完了すると、IBM Cloud アプリケーションのモニター・データが Cloud APM コンソールに表示されることを確認できます。

Cloud APM サーバーへのデータ・コレクターの再接続

Cloud APM サーバー、鍵ファイル、または鍵ファイルのパスワードが変更された場合は、データ・コレクターを Cloud APM サーバーに再接続するために複数の環境変数を設定する必要があります。

始める前に

鍵ファイルが変更された場合は、まず Base64 を使用して鍵ファイルのプレーン・テキスト・パスワードを暗号化します。Linux ユーザーの場合は、以下のコマンドを実行します。

```
echo -n keyfile_password | base64
```

コマンド出力は、暗号化されたパスワードです。例えば、プレーン・テキスト・パスワードが password の場合、コマンド出力 cGFzc3dvcmQ= が暗号化されたパスワードです。この暗号化されたパスワードを使用して、以下の構成で APM_KEYFILE_PSWD: *encrypted_keyfile_password* と APM_KEYFILE_PSWD=*encrypted_keyfile_password* を設定します。

手順

- IBM Cloud アプリケーション用の Cloud APM サーバーにデータ・コレクターを再接続するには、190 ページの『IBM Cloud アプリケーション用のデータ・コレクターへの再接続』を参照してください。
- オンプレミス用の Cloud APM サーバーにデータ・コレクターを再接続するには、191 ページの『オンプレミス・アプリケーション用のデータ・コレクターへの再接続』を参照してください。

IBM Cloud アプリケーション用のデータ・コレクターへの再接続

このタスクについて

データ・コレクターを Cloud APM サーバーに再接続するには、以下の 2 つのオプションがあります。

- アプリケーションの manifest.yml を編集して変数を設定する。
- IBM Cloud UI 上で変数を設定する。

手順

- IBM Cloud アプリケーションの manifest.yml ファイルを使用してデータ・コレクターを再接続するには、以下の手順を実行します。
  a) 変更内容に応じて、IBM Cloud アプリケーションの manifest.yml ファイル内の変数を編集します。
  - *HTTP* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

    ```
    APM_BM_GATEWAY_URL: http://server_ip_or_hostname:80
    ```

  - *HTTPS* を使用するようにゲートウェイを構成するには、以下の 3 つの変数を設定します。

    ```
    APM_BM_GATEWAY_URL: https://server_ip_or_hostname:443
    APM_KEYFILE_PSWD: encrypted_keyfile_password
    APM_KEYFILE_URL: http://hosted_http_server:port/keyfile_name
    ```

    ヒント: Liberty データ・コレクターの鍵ファイルは、.jks ファイルです。Python データ・コレクター、Node.js データ・コレクター、および Liberty データ・コレクターについては、鍵ファイルは .p12 ファイルです。

  b) IBM Cloud アプリケーションのディレクトリーに移動し、以下のコマンドを実行します。

  ```
  cf push
  ```

- IBM Cloud UI を使用してデータ・コレクターを再接続するには、以下の手順を実行します。
  a) IBM Cloud UI にログインします。

b) IBM Cloud アプリケーションをクリックします。

c) 左パネル上で「**ランタイム**」をクリックします。

d)「**環境変数**」タブに切り替えます。

e)「**ユーザー定義**」 セクションで、ニーズに応じて、以下のいずれかの方法で変数を定義します。

– *HTTP* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

```
APM_BM_GATEWAY_URL: http://server_ip_or_hostname:80
```

– *HTTPS* を使用するようにゲートウェイを構成するには、以下の 3 つの変数を設定します。

```
APM_BM_GATEWAY_URL: https://server_ip_or_hostname:443
APM_KEYFILE_PSWD: encrypted_keyfile_password
APM_KEYFILE_URL: http://hosted_http_server:port/keyfile_name
```

ヒント : Liberty データ・コレクターの鍵ファイルは、 `.jks` ファイルです。Python データ・コレクター、Node.js データ・コレクター、および Liberty データ・コレクターについては、鍵ファイルは `.p12` ファイルです。

f) **`cf push`** コマンドを実行してアプリケーションをプッシュしたディレクトリーで 、以下のコマンドを実行して変更を有効にします。

```
cf restage <app_name>
```

タスクの結果

データ・コレクターを Cloud APM サーバーに接続するように、変数の値が正しく設定されました。

オンプレミス・アプリケーション用のデータ・コレクターへの再接続

このタスクについて

`global.environment` ファイルまたは `dc.java.properties` ファイルを変更すると、データ・コレクターと Cloud APM サーバーの間の接続をカスタマイズできます。

手順

1. 接続変数を含む該当するファイルを特定します。

   a) Liberty データ・コレクター、Node.js データ・コレクター、および Python データ・コレクターの場合は、以下の表の情報に従って `global.environment` ファイルを特定します。

データ・コレクター名	**global.environment ファイルのディレクトリー**
Liberty データ・コレクター	Liberty データ・コレクターがインストールされている `itcamdc/etc/global.environment` フォルダー。
Node.js データ・コレクター	Node.js データ・コレクターがインストールされている `ibmapm/etc` フォルダー。
Python データ・コレクター	Python データ・コレクターがインストールされている `etc` フォルダー。

   b) J2SE データ・コレクターの場合は、*DC_HOME*`/itcamdc/etc` フォルダーの `dc.java.properties` ファイルを特定します。*DC_HOME* は、J2SE データ・コレクターのインストール先ディレクトリーです。

2. 変更内容に応じて、対応するファイル内の変数を編集します。

a) Liberty データ・コレクター、Node.js データ・コレクター、および Python データ・コレクターの場合は、以下の手順に従って `global.environment` ファイルを編集します。

- *HTTP* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

```
APM_BM_GATEWAY_URL=http://server_ip_or_hostname:80
```

- *HTTPS* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

```
APM_BM_GATEWAY_URL=https://server_ip_or_hostname:443
APM_KEYFILE_PSWD=encrypted_keyfile_password
APM_KEYFILE_URL=http://hosted_http_server_:port/keyfile_name
```

ヒント: Liberty データ・コレクターの鍵ファイルは `.jks` ファイルです。Python データ・コレクター、Node.js データ・コレクター、および Liberty データ・コレクターについては、鍵ファイルは `.p12` ファイルです。

b) J2SE データ・コレクターの場合は、以下の手順に従って `dc.java.properties` ファイルを編集します。

- *HTTP* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

```
apm.http.type=http
```

この変数の値を空のままにすると、`http` がデフォルト値となります。

- *HTTPS* を使用するようにゲートウェイを構成するには、以下の変数を設定します。

```
apm.ssl.password=encrypted_keyfile_password
apm.http.type=https
```

重要: パスワードが変更された場合は、*DC_HOME*`/itcamdc/etc/keyfile.jks` ファイルを Cloud APM サーバーの `/opt/ibm/ccm/keyfiles/default.agent/keyfiles/keyfile.jks` ファイルに置き換えます。ここで *DC_HOME* は、J2SE データ・コレクターのホーム・ディレクトリーです。

3. オプション: Node.js データ・コレクターでデフォルトの鍵ファイルを使用しない場合は、以下の変数を設定します。

```
APM_SNI=owner_host_in_the_key_file
```

ヒント: *owner host* 変数の値を調べるには、使用している鍵ファイルを開いて `owner` を検索します。次に、*APM_SNI* 変数を *owner* と同じ値に設定します。

4. アプリケーションを再始動して、変更を有効にします。

タスクの結果

データ・コレクターを Cloud APM サーバーに接続するように、変数の値が正しく設定されました。

サンプル manifest.yml ファイル

IBM Cloud アプリケーションの `manifest.yml` ファイルの内容については、以下の行を参照してください。

```
applications:

- disk_quota: 1024M
  host: myBluemixApp
  name: myBluemixApp
  path: .
  domain: mybluemix.net

  instances: 1
  memory: 512M

  env:
```

```
    KNJ_ENABLE_TT: "true"
    KNJ_SAMPLING: 1
```

Cloud APM コンソールからのデータ・コレクターの削除

データ・コレクターを構成解除した後は、データ・コレクターを追加したアプリケーションおよびリソース・グループからそのデータ・コレクターを削除する必要もあります。そうしないと、Cloud APM コンソールに、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。

手順

1. データ・コレクターを追加したアプリケーションからデータ・コレクターを削除するには、アプリケーションを手動で編集します。

   この処理は、アプリケーションからオフライン・エージェントを削除する処理に似ています。1104 ページの『オフライン・エージェントの表示および削除』を参照してください。
2. データ・コレクターを追加したすべてのカスタム・リソース・グループからデータ・コレクターを削除します。

   詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。
3. Cloud APM 運用チームのチケットをオープンし、Cloud APM サーバーで以下の手順を実行します。
   a) *install_dir*/serveragents/config/hostname_bi.cfg ファイルを編集し、構成解除したデータ・コレクターの行を削除します。
   b) root ユーザーとして以下のコマンドを実行することによって、データ・コレクターのサーバー・コンポーネントを再始動します。

   ```
   apm restart biagent
   ```

タスクの結果

数分後に、Cloud APM コンソールでデータ・コレクターのシステム・リソース・グループを選択すると、「**マイ・コンポーネント**」アプリケーションおよび「**リソース・グループ・マネージャー**」 UI にデータ・コレクターがオフラインであると表示されます。

「**拡張構成**」ページの「**オフライン・システム除去の遅延**」構成プロパティーによって指定された時間が経過した後に、自動的にデータ・コレクターが「**マイ・コンポーネント**」およびそのシステム・リソース・グループから削除されます。

ヒント: 「**拡張構成**」ページの「**オフライン・システム除去の遅延**」設定を調整して、オフライン・エージェントをビューに表示しないようにするまでの待機時間を増減できます。詳しくは、1075 ページの『エージェント・サブスクリプション・ファシリティー』を参照してください。

要確認: データ・コレクターがトランザクション・トラッキング・データを Cloud APM サーバーに提供していた場合は、引き続き Cloud APM コンソールの「**マイ・コンポーネント**」アプリケーションにデータ・コレクターが表示され、「**オフライン・システム除去の遅延**」設定で指定された時間が経過しても、そのデータ・コレクターに対して「エージェントが無効です」というメッセージが表示される場合があります。Cloud APM 8.1.4.0 Server 暫定修正 3 以降をインストールした場合は、最終的に、データ・コレクターからのトランザクション・トラッキング・データの受信が停止してから 8 日後に、無効なデータ・コレクターが「**マイ・コンポーネント**」アプリケーションから削除されます。

Amazon EC2 モニターの構成

Amazon EC2 エージェントは、Amazon Elastic Compute Cloud (EC2) インスタンスの正常性、可用性、およびパフォーマンスについてモニターする 中央ポイントを提供します。このエージェントは、EC2 環境に関して十分な情報を元に意思決定するのに役立つ包括的なメトリック・セットを表示します。これらのメトリックには、CPU 使用率、Elastic Block Store (EBS) 使用率、ネットワーク使用状況、Amazon Web Services (AWS) の保守更新、およびディスク・パフォーマンスなどが含まれます。

始める前に

- 193 ページの『Amazon EC2 モニターの構成』のトピック全体を読んで、構成を完了するために何が必要かを判断します。
- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、Amazon EC2 エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Amazon EC2 エージェント を参照してください。
- 次の情報が使用可能であることを確認します。
  - モニター対象の EC2 インスタンスが含まれている AWS リージョン名のリスト。
  - 各 AWS リージョンへのアクセス許可が含まれる AWS セキュリティー資格情報 (アクセスキー ID およびシークレットアクセスキー)。
- 各 AWS リージョンに使用される AWS セキュリティー資格情報が、少なくとも *AmazonEC2ReadOnlyAccess* ポリシーを含むグループのメンバーであることを確認します。

このタスクについて

Amazon EC2 エージェントは複数インスタンス・エージェントであり、サブノード・エージェントでもあります。複数のサブノードを含む 1 つのエージェント・インスタンスを作成できます (各 Amazon EC2 リージョンが 1 つのサブノードに対応します)。または、各 Amazon EC2 リージョンに対して 1 つのエージェント・インスタンスを作成できます (そのリージョンに対応する 1 つのサブノードが含まれます)。または、各タイプの構成の組み合わせを作成できます。エージェント・インスタンスを構成した後で、各エージェント・インスタンスを手動で開始する必要があります。Amazon EC2 リージョンごとのリソース数が 50 を超える場合は、リージョンごとにエージェント・インスタンスを作成するか、EC2 インスタンスへのタグ付けを使用して、エージェントのフィルター条件パラメーターを使用して作成したタグでエージェント・インスタンスをフィルタリングすることをお 勧めします。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント 応答ファイルを使用して Windows システムでエージェントを構成します。
   - 195 ページの『Windows システムでのエージェントの 構成』.
   - 199 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. 応答を要求するプロンプトを出すスクリプト、またはサイレント 応答ファイルを使用して、Linux システムでエージェントを構成します。
   - 198 ページの『プロンプトへの応答によるエージェントの構成』.
   - 199 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Amazon EC2 エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for Amazon EC2**」 テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。例えば、`ec2-inst3` です。

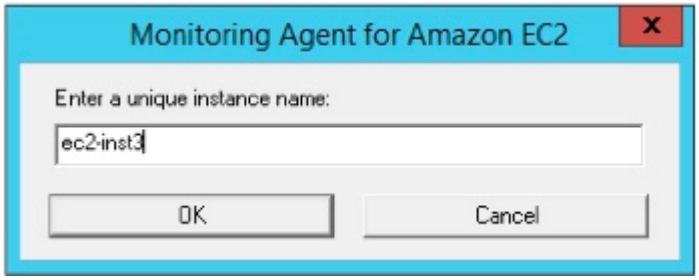


図 *3.* 固有のインスタンス名を入力するウィンドウ

4. エージェント・インスタンス名ウィンドウで「**次へ**」をクリックします。

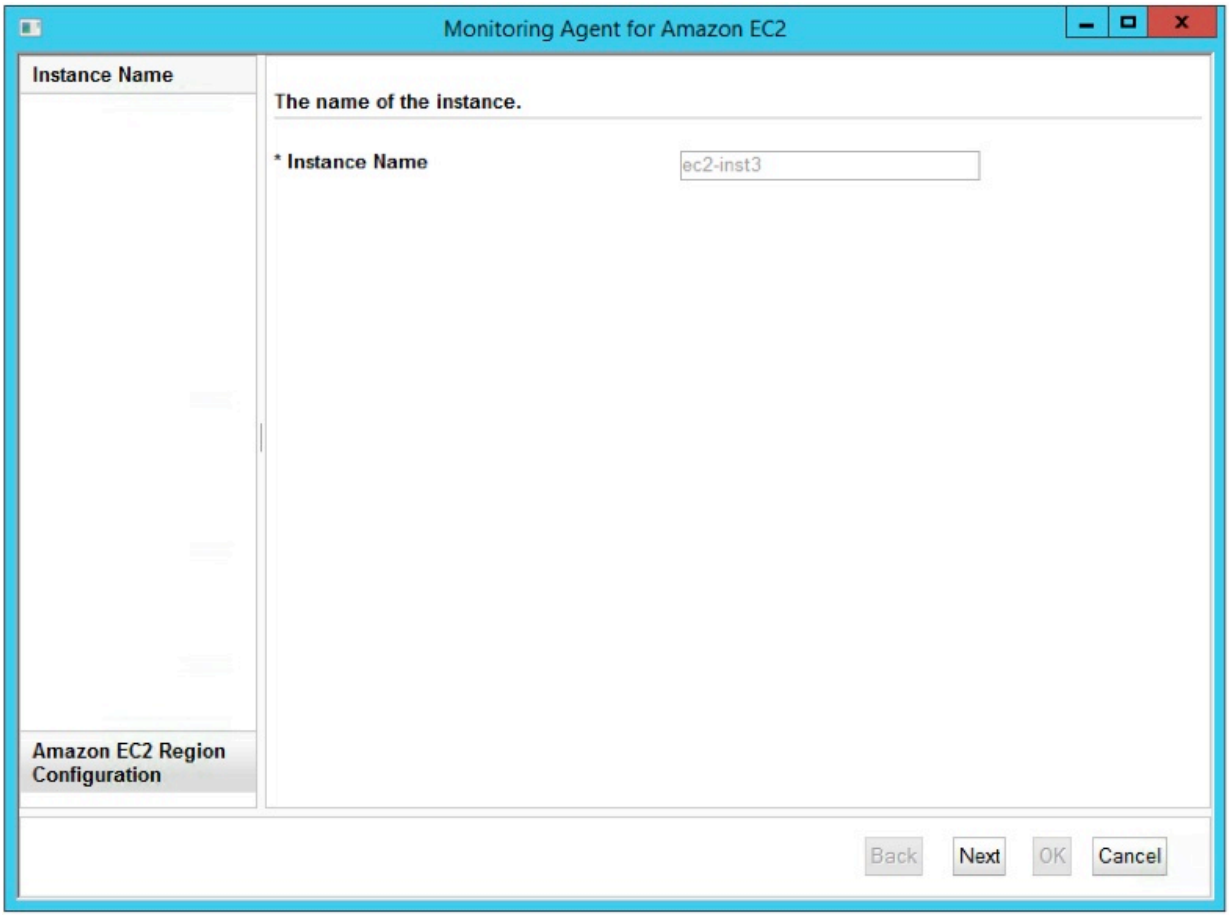


図 *4.* エージェント・インスタンス名ウィンドウ。

5. 「**Amazon EC2 リージョンの構成**」インスタンスのテンプレート設定を入力します。

   注 : このセクションは、Amazon EC2 リージョン・インスタンスの構成ではありません。ステップ 6 で実際の Amazon EC2 リージョン・インスタンス構成を追加する際にデフォルト値として使用する内容を設定するためのテンプレート・セクションです。

   各構成パラメーターの説明については、200 ページの表 14 を参照してください。

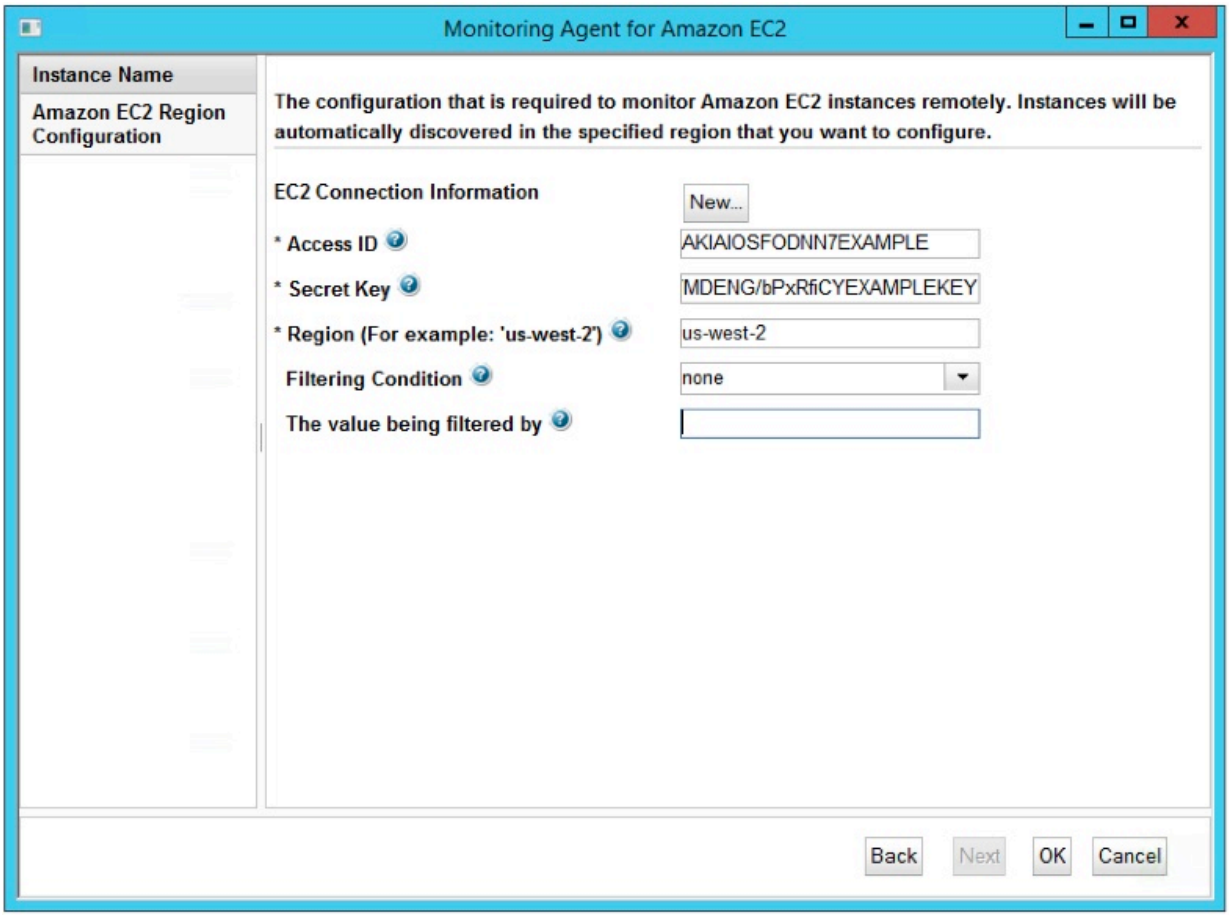


図 *5. Amazon EC2* リージョン・インスタンスのテンプレート設定を指定するためのウィンドウ。

6.「**新規**」をクリックし、Amazon EC2 リージョン・インスタンスの設定を入力し、「**次へ**」をクリックします。

各構成パラメーターの説明については、200 ページの表 14 を参照してください。

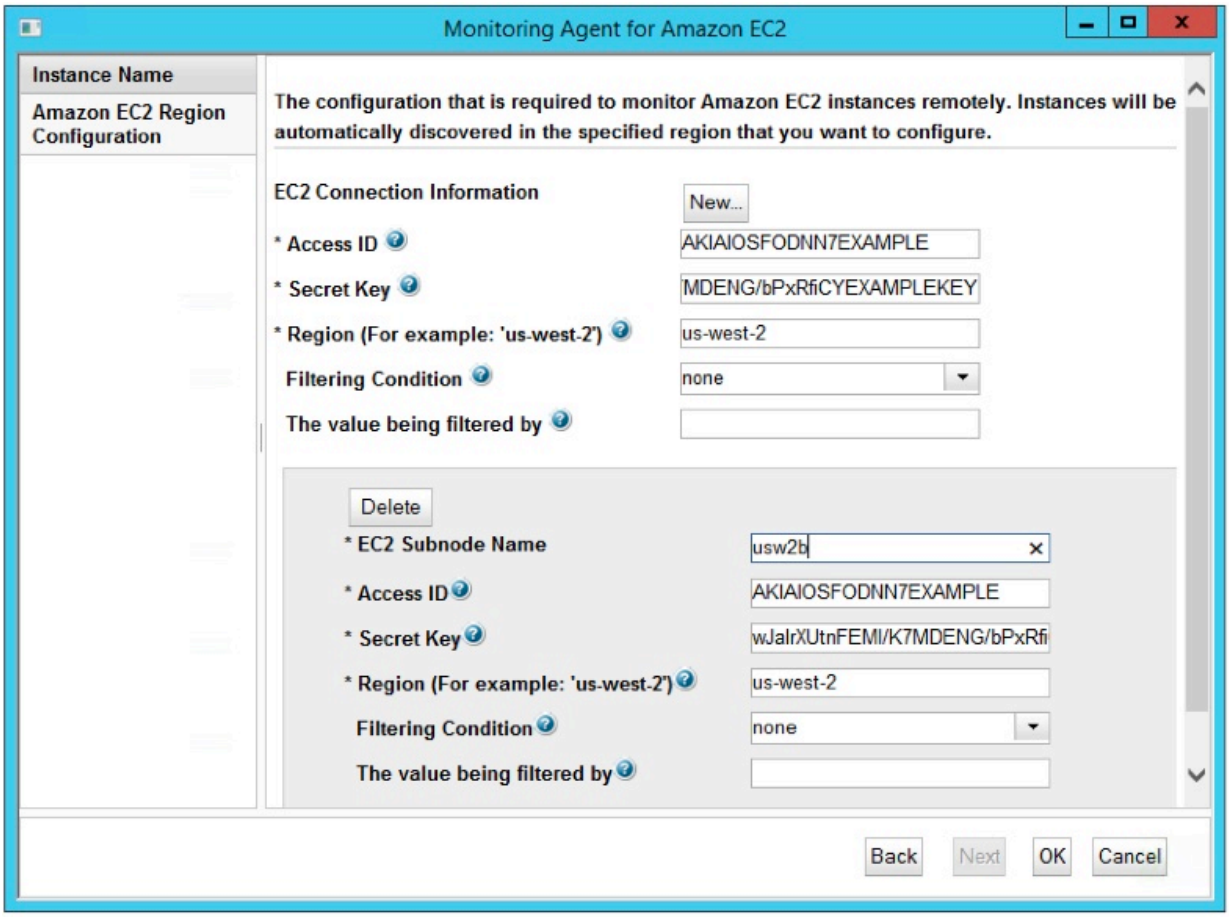


図 6. *Amazon EC2* リージョン・インスタンスの設定を指定するためのウィンドウ。

7. 「**OK**」をクリックして、構成を完了します。
8. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

Amazon EC2 エージェントのインストール後、エージェントを開始する前に構成する必要があります。Amazon EC2 エージェントがローカル Linux コンピューターにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで Amazon EC2 エージェントを構成します。

1. 次のコマンドを実行します。

```
install_dir/bin/amazonec2-agent.sh config instance_name
```

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

例

```
/opt/ibm/apm/agent/bin/amazonec2-agent.sh config ec2-inst3
```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、200 ページの『Amazon EC2 エージェントの構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/amazonec2-agent.sh start instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/amazonec2-agent.sh start ec2-inst3
   ```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- 以下の手順を実行して、Amazon EC2 エージェントをサイレント・モードで構成します。

  a) テキスト・エディターで、以下のいずれかのパスにある amazonec2_silent_config.txt ファイルを開きます。

  – Linux *install_dir*/samples/amazonec2_silent_config.txt

    例: /opt/ibm/apm/agent/samples/amazonec2_silent_config.txt

  – Windows *install_dir*¥samples¥amazonec2_silent_config.txt

    例: C:¥IBM¥APM¥samples¥amazonec2_silent_config.txt

  ここで、*install_dir* はエージェントのインストール先パスです。

  b) amazonec2_silent_config.txt ファイルで、すべての必須パラメーターの値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

  各構成パラメーターの説明については、200 ページの『Amazon EC2 エージェントの構成パラメーター』を参照してください。

  c) amazonec2_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。

  – Linux *install_dir*/bin/amazonec2-agent.sh config **instance_name** *install_dir*/samples/amazonec2_silent_config.txt

    例: **/opt/ibm/apm/agent/bin/amazonec2-agent.sh config ec2-inst3 /opt/ibm/apm/agent/samples/amazonec2_silent_config.txt**

  – Windows *install_dir*¥bin¥amazonec2-agent.bat config **instance_name** *install_dir*¥samples¥amazonec2_silent_config.txt

例: **C:¥IBM¥APM¥bin¥amazonec2-agent.bat config ec2-inst3 C:¥IBM¥APM ¥samples¥amazonec2_silent_config.txt**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

– Linux *install_dir*/bin/amazonec2-agent.sh start **instance_name**

例: **/opt/ibm/apm/agent/bin/amazonec2-agent.sh start ec2-inst3**

– Windows *install_dir*¥bin¥amazonec2-agent.bat start **instance_name**

例: **C:¥IBM¥APM¥bin¥amazonec2-agent.bat start ec2-inst3**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

Amazon EC2 エージェントの構成パラメーター

Amazon EC2 エージェントの構成パラメーターを表に表示します。

1. Amazon EC2 リージョンの構成 - Amazon EC2 インスタンスをリモートでモニターするための 設定。構成対象の指定のリージョンで、インスタンスが自動的にディスカバーされます。

表 *14. Amazon EC2* リージョンの構成

パラメーター名	説明	サイレント構成ファイルのパラメーター名
EC2 サブノード名 (EC2 Subnode Name)	データ収集のための EC2 サブノードの名前。例えば、*usw2a* です。 この別名は、管理対象システム名 (MSN) の一部であり、Cloud APM コンソール内でモニター対象環境を視覚的に識別するために使用されます。 **注 :** この別名は、Amazon EC2 サブノード・インスタンスを表す任意のものにすることができますが、以下の制限があります。エージェント・サブノード・インスタンス名の作成には、英字 (a-z、A-Z)、アラビア数字 (0-9)、ハイフンマイナス文字 (-)、および下線文字 (_) を使用できます。EC2 サブノード名の最大長は 25 文字です。	以下の各パラメーターには、サブノード名サフィックスを付ける必要があります。このサフィックスは、1 つのエージェント・サブノード・インスタンスのどのパラメーターでも同じです。新規エージェント・サブノード・インスタンスのパラメーターのセットでは、固有の名前を使用する必要があります。例えば、以下のパラメーター名の *.subnode_name* の代わりに、あるサブノード・インスタンスでは *.usw2a* を使用し、別のエージェント・インスタンスでは *.usw2b* を使用できます。
アクセス ID	指定された Amazon リージョンでの認証に使用する AWS セキュリティー資格情報のアクセスキー ID。例えば、「AKIAxxxxxxxxxxxxxxx」です。	**KB5_INS_ACCESS_ID.subnode_name**
秘密鍵 (Secret Key)	指定された Amazon リージョンでの認証に使用する AWS セキュリティー資格情報のシークレットアクセスキー。例えば、「kK7txxxxxxxxxxxxxxxxxxxxxxxxx」です。	**KB5_INS_SECRET_KEY.subnode_name**
リージョン	モニター対象の AWS リージョン。例えば、「us-west-2」です。	**KB5_INS_REGION.subnode_name**

表 14. *Amazon EC2* リージョンの構成 (続き)

パラメーター名	説明	サイレント構成ファイルのパラメーター名
フィルター条件 (Filtering Condition)	実行するフィルター処理のタイプ。 EC2 インスタンスに対してカスタム・タグを使用して、エージェントによってモニターされる EC2 インスタンスを制限することができます。詳しくは、Amazon EC2 リソースにタグを付けるを参照してください。 フィルター・オプション: **なし** モニター対象のリージョン内のすべての EC2 インスタンス。**「フィルター値 (Filter Value)」**は無視されます。 **tagName** **「フィルター値 (Filter Value)」**に指定されたタグ・キーを持つ EC2 インスタンスが、対応する EC2 インスタンス・タグ値の実際の値に関係なくモニターされます。例えば、タグ・キー *Stack* を持つすべての EC2 インスタンスを、そのタグ値での値に関係なくモニターするには、**「フィルター値 (Filter Value)」**に Stack と指定します。 **tagName|tagValue** 縦棒 (|) で区切られ、**「フィルター値 (Filter Value)」**に指定されたタグ・キーとタグ値のペアを持つ EC2 インスタンスがモニターされます。例えば、タグ・キー *Stack* とタグ値 *Production* を持つすべての EC2 インスタンスをモニターするには、**「フィルター値 (Filter Value)」**に Stack|Production と指定します。 **monitoring-tag** 1 つ以上のタグを持つ EC2 インスタンスがモニターされます。**「フィルター値 (Filter Value)」**は無視されます。	**FILTER_CONDITION.subnode_name** 有効な値を以下に示します。 **なし** なし **tagName** tagName **tagName|tagValue** tagName|tagValue **monitoring-tag** monitoring-tag
フィルター値 (Filter Value)	**「フィルター条件 (Filtering Condition)」**で tagName または tagName|tagValue が選択されている場合に EC2 インスタンスのフィルタリングに使用されるタグの値。	**FILTER_VALUE.subnode_name**

AWS Elastic Load Balancer モニターの構成

Amazon ELB エージェントは、AWS Elastic Load Balancer の正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、AWS Elastic Load Balancer 環境に関して十分な情報に基づいて意思決定するために役立つ、各ロード・バランサー・タイプ (アプリケーション、ネットワーク、およびクラシック) に関する包括的な一連のメトリックが表示されます。

始める前に

- 201 ページの『AWS Elastic Load Balancer モニターの構成』のトピック全体を読んで、構成を完了するために何が必要かを判断します。
- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、Amazon ELB エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Amazon ELB エージェント を参照してください。
- 次の情報が使用可能であることを確認します。
  - Elastic Load Balancer での各 AWS リージョンへのアクセス許可が含まれる AWS セキュリティー資格情報 (アクセスキー ID およびシークレットアクセスキー)。

このタスクについて

Amazon ELB エージェントは複数インスタンス・エージェントであり、サブノード・エージェントでもあります。サブノードは、AWS 環境で使用可能な各タイプの Elastic Load Balancer 用に自動的に作成されます。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用して Windows システムでエージェントを構成します。
   - 202 ページの『Windows システムでのエージェントの 構成』.
   - 205 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. 応答を要求するプロンプトを出すスクリプト、またはサイレント応答ファイルを使用して、Linux システムでエージェントを構成します。
   - 204 ページの『プロンプトへの応答によるエージェントの構成』.
   - 205 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Amazon ELB エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for AWS Elastic Load Balancer**」テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。

3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。例えば、`elb-inst3` です。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

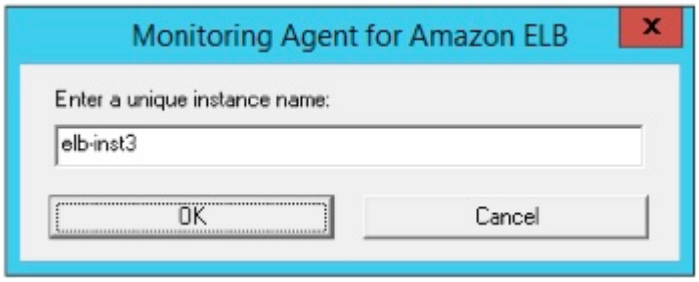


図 7. 固有のエージェント・インスタンス名を入力するウィンドウ

4. 「**Amazon ELB サブスクリプション資格情報**」を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、206 ページの『Amazon ELB エージェントの構成パラメーター』を参照してください。

   重要 : Windows **「秘密鍵/パスワード」**に等号 (=) が含まれている場合、エージェントを再構成するたびに再入力する必要があります。

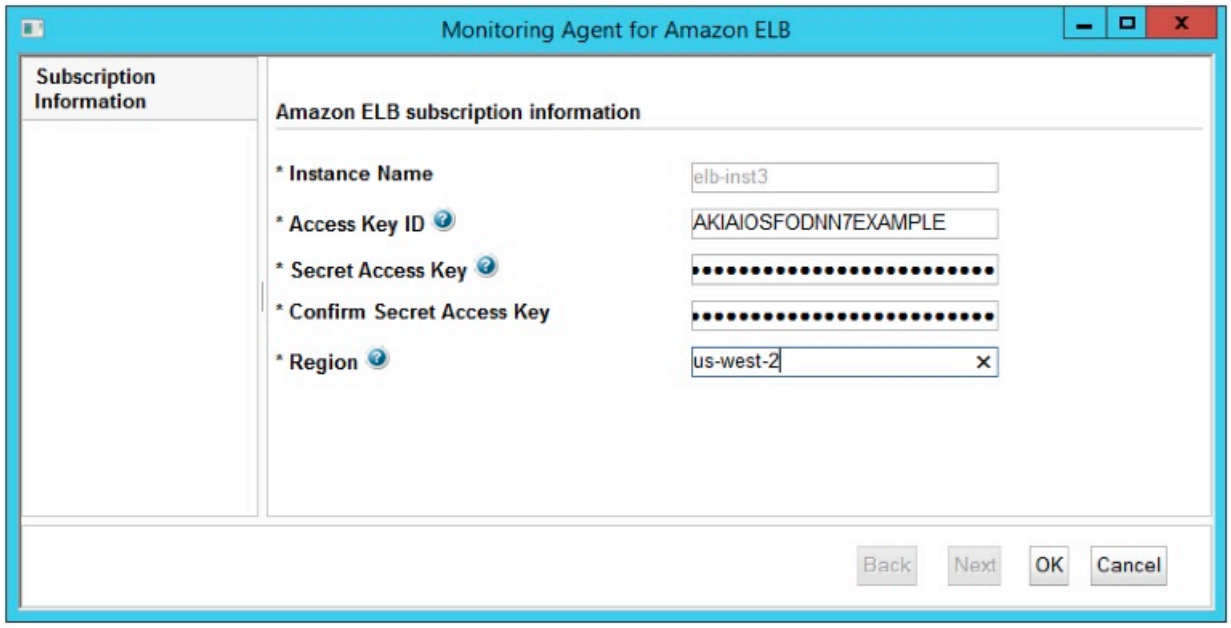


図 *8.* 「*Amazon ELB* サブスクリプション資格情報」ウィンドウ。

5. 「**OK**」をクリックして、構成を完了します。
6. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

Amazon ELB エージェントのインストール後、エージェントを開始する前に構成する必要があります。Amazon ELB エージェントがローカル Linux コンピューターにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで Amazon ELB エージェントを構成します。

1. 次のコマンドを実行します。

   ```
   install_dir/bin/amazon_elb-agent.sh config instance-name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスに指定する名前です。*instance-name* では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

   例

   ```
   /opt/ibm/apm/agent/bin/amazon_elb-agent.sh config elb-inst3
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、206 ページの『Amazon ELB エージェントの構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/amazon_elb-agent.sh start instance-name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/amazon_elb-agent.sh start elb-inst3
   ```

例

elb-inst3 というエージェント・インスタンスを作成します。

```
# ./amazon_elb-agent.sh config elb-inst3
Configuring Monitoring Agent for Amazon ELB

Edit 'Monitoring Agent for Amazon ELB' settings? [1=はい、2=いいえ] (デフォルト: 1): 1

Subscription Information :
Amazon ELB subscription information

The access ID that is used to authenticate with the specified Amazon Region.
例えば、「AKIAxxxxxxxxxxxxxx」です。
Access Key ID (default is: ): AKIAIOSFODNN7EXAMPLE

The secret access key that is used to authenticate with the specified Amazon
Region. 例えば、「kK7txxxxxxxxxxxxxxxxxxxxxxxxx」です。
Enter Secret Access Key (default is: ): hidden

Re-type : Secret Access Key (default is: ): hidden
```

```
The Amazon region where the load balancers are located. 例えば、「us-west-2」です。
Region (default is: ): us-west-2

構成が正常に完了しました。
Automatic start at system initialization has been configured.
Automatic stop at system shutdown has been configured.
```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

Amazon ELB エージェントをサイレント・モードで構成するには、以下のステップを実行します。

1. テキスト・エディターで、以下のいずれかのパスにある amazon_elb_silent_config.txt ファイルを開きます。
   - Linux *install_dir*/samples/amazon_elb_silent_config.txt

     例: /opt/ibm/apm/agent/samples/amazon_elb_silent_config.txt
   - Windows *install_dir*¥samples¥amazon_elb_silent_config.txt

     例: C:¥IBM¥APM¥samples¥amazon_elb_silent_config.txt

   ここで、*install_dir* はエージェントのインストール先パスです。
2. amazon_elb_silent_config.txt ファイルで、すべての必須パラメーターの値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

   各構成パラメーターの説明については、206 ページの『Amazon ELB エージェントの構成パラメーター』を参照してください。
3. amazon_elb_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
   - Linux *install_dir*/bin/amazon_elb-agent.sh config **instance-name** *install_dir*/samples/amazon_elb_silent_config.txt

     例: **/opt/ibm/apm/agent/bin/amazon_elb-agent.sh config elb-inst3 /opt/ibm/apm/agent/samples/amazon_elb_silent_config.txt**
   - Windows *install_dir*¥bin¥amazon_elb-agent.bat config **instance-name** *install_dir*¥samples¥amazon_elb_silent_config.txt

     例: **C:¥IBM¥APM¥bin¥amazon_elb-agent.bat config elb-inst3 C:¥IBM¥APM¥samples¥amazon_elb_silent_config.txt**

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスに指定する名前です。*instance-name* では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

   重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。
4. 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/amazon_elb-agent.sh start **instance-name**

  例: **/opt/ibm/apm/agent/bin/amazon_elb-agent.sh start elb-inst3**
- Windows *install_dir*¥bin¥amazon_elb-agent.bat start **instance-name**

  例: **C:¥IBM¥APM¥bin¥amazon_elb-agent.bat start elb-inst3**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスの名前です。

例

編集済みの amazon_elb_silent_config.txt。

```
#
# This is a sample configuration response file for agent Amazon ELB.
#
# It contains an entry for every configuration property.
# Entries for optional properties that have no default value are included
# in comments.
# Ensure that all uncommented properties have a value before configuring
# the agent.
#

# Access Key ID: The access ID that is used to authenticate with the
# specified Amazon Region. 例えば、「AKIAxxxxxxxxxxxxxxx」です。
KAL_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE

# Secret Access Key: The secret access key that is used to authenticate with
# the specified Amazon Region. 例えば、「kK7txxxxxxxxxxxxxxxxxxxxxxxxxxx」です。
KAL_SECRET_ACCESS_KEY_PASSWORD=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

# Region: The Amazon region where the load balancers are located. For
# example, 'us-west-2'.
KAL_REGION=us-west-2
```

Amazon ELB エージェントの構成パラメーター

Amazon ELB エージェントの構成パラメーターを表に表示します。

1. 206 ページの表 15 - モニター対象の AWS Elastic Load Balancer を含む Amazon サブスクリプションへのアクセスに必要な資格情報。

表 *15.* サブスクリプション情報

パラメーター名	説明	サイレント構成ファイルのパラメーター名
アクセスキー ID	The access ID that is used to authenticate with the specified Amazon Region. 例えば、「AKIAxxxxxxxxxxxxxxx」です。	**KAL_ACCESS_KEY_ID**
シークレットアクセスキー	The secret access key that is used to authenticate with the specified Amazon Region. 例えば、「kK7txxxxxxxxxxxxxxxxxxxxxxxxxxx」です。	**KAL_SECRET_ACCESS_KEY_PASSWORD**
リージョン	The Amazon region where the load balancers are located. 例えば、「us-west-2」です。	**KAL_REGION**

Azure Compute モニター構成

Azure Compute エージェントは、Azure Compute インスタンスの正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。エージェントにより、Azure Compute 環境に関して十

分な情報に基づいて意思決定する際に役立つ、包括的な一連のメトリックが表示されます。これらのメトリックには、CPU 使用率、ネットワーク使用状況、およびディスク・パフォーマンスが含まれます。

始める前に

- 206 ページの『Azure Compute モニター構成』のトピック全体を読んで、構成を完了するために何が必要かを判断します。
- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、Azure Compute エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Azure Compute エージェント を参照してください。
- 次の情報が使用可能であることを確認します。
  - Azure Compute インスタンスにアクセスしてモニターするための権限がある Azure サブスクリプション資格情報。詳しくは、208 ページの『Azure Compute の構成に関する情報』を参照してください。

このタスクについて

Azure Compute エージェントは複数インスタンス・エージェントであり、サブノード・エージェントでもあります。Azure Compute エージェントの各サブノードは、定義するフィルターに従って、Azure Compute 仮想マシンのグループをモニターします。複数のサブノードを持つ 1 つのエージェント・インスタンスを作成できます (各仮想マシン・グループに対して 1 つのサブノード)。または、各仮想マシン・グループに対して、そのグループに対応する 1 つのサブノードを持つ 1 つのエージェント・インスタンスを作成できます。または、各タイプの構成の組み合わせを作成できます。エージェント・インスタンスを構成した後で、各エージェント・インスタンスを手動で開始する必要があります。Azure Compute 仮想マシン・グループあたりのリソース数は 50 以下にすることが推奨されます。各 Azure Compute エージェントのサブノード名は環境内で固有である必要があります。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用して Windows システムでエージェントを構成します。
   - 208 ページの『Windows システムでのエージェントの 構成』.
   - 213 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. 応答を要求するプロンプトを出すスクリプト、またはサイレント応答ファイルを使用して、Linux システムでエージェントを構成します。
   - 211 ページの『プロンプトへの応答によるエージェントの構成』.
   - 213 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Azure Compute の構成に関する情報

Azure Compute エージェントは、Azure Compute 環境でいくつかの追加のセットアップを必要とします。

このタスクについて

以下のステップを実行するには、Microsoft Azure コンソールにログインする必要があります。

手順

1. サブスクリプション ID
   - 左側で「サブスクリプション」を選択し、このエージェントで使用するサブスクライブを選択します。
   - 「概要」を選択し、サブスクリプション ID をコピーします。これはエージェントの構成パラメーターの 1 つとして使用します。
2. テナント ID
   - 「Azure Active Directory」にナビゲートします。
   - 「プロパティ」を選択し、テナント ID をコピーします。
3. アプリケーションを登録する
   - 「すべてのサービス」に移動し、「アプリの登録」を探します。
   - 「新しいアプリケーションの登録」をクリックします。
   - 名前を入力し、アプリケーションの種類に「Web App/API」を選択して、サインオン URL を選択します (この URL は使用されないため、何でもお好きなものを選択してください)。
   - 「作成」をクリックします。
   - アプリケーション ID をコピーします。これはエージェントの「クライアント ID」フィールドで使用します。
4. アプリケーション・キーを作成する
   - 先ほど作成したアプリをクリックし、「設定」、「鍵」の順に移動します。
   - 説明 (例:「IBM Key」) と期間 (例:「有効期限なし」) を入力し、「保存」をクリックします。
   - 秘密鍵をコピーし、安全な場所に格納します。この鍵は一度しか表示されないため、失くした場合は新しいものを生成する必要があります。
5. アプリケーションに許可を与える
   - 「サブスクリプション」に移動し、モニターするサブスクリプションを選択します。
   - 「アクセス制御 (IAM)」に移動し、「追加」をクリックします。
   - モニター用に「閲覧者」かそれ以上の役割を選択します。
   - 「選択」から先ほど登録したアプリを見つけて選択し、「保存」をクリックします。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Azure Compute エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for Azure Compute**」テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。

3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、ハイフンマイナス文字のみを使用してください。例えば、`azc-inst3` です。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

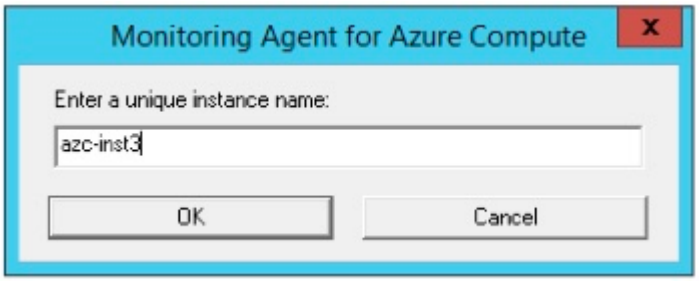


図 *9.* 固有のエージェント・インスタンス名を入力するウィンドウ

4. 「**Azure サブスクリプション資格情報**」を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、215 ページの表 16 を参照してください。

   重要 : Windows **「秘密鍵/パスワード」**に等号 (=) が含まれている場合、エージェントを再構成するたびに再入力する必要があります。

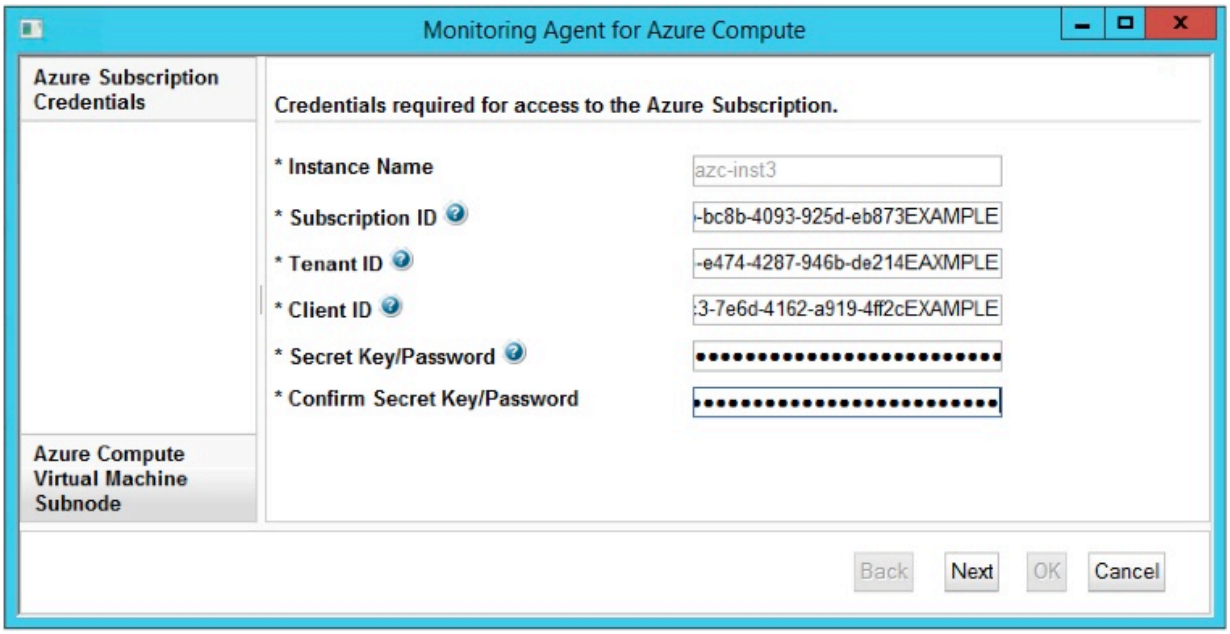


図 *10.* 「*Azure* サブスクリプション資格情報」ウィンドウ。

5. 「**Azure Compute 仮想マシン・サブノード**」テンプレート設定を入力します。
   各構成パラメーターの説明については、216 ページの表 17 を参照してください。

   注 : このセクションは、Azure Compute 仮想マシン・サブノードのインスタンス構成ではありません。ステップ 6 で実際の Azure Compute 仮想マシン・サブノードのインスタンス構成を追加するときにデフォルト値として使用される内容を設定するためのテンプレート・セクションです。

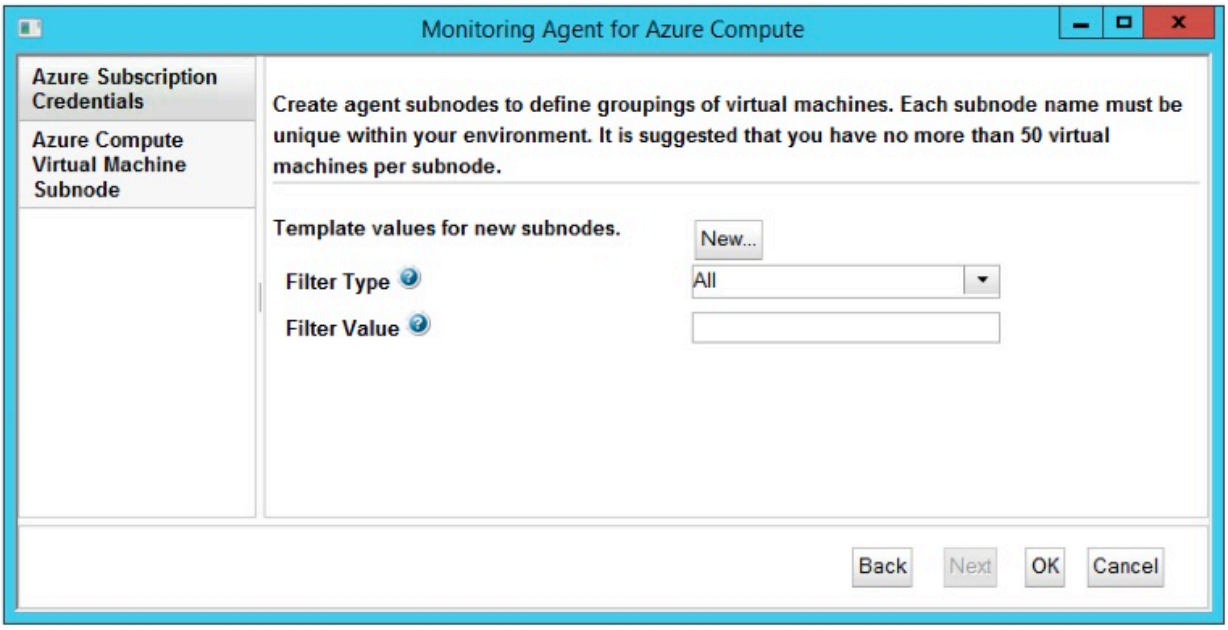


図 *11. Azure Compute* 仮想マシン・サブノードのテンプレート設定を指定するためのウィンドウ。

6. 「**新規**」をクリックし、**Azure Compute 仮想マシン・サブノード** のインスタンス設定を入力し、「**次へ**」をクリックします。

   各構成パラメーターの説明については、216 ページの表 17 を参照してください。

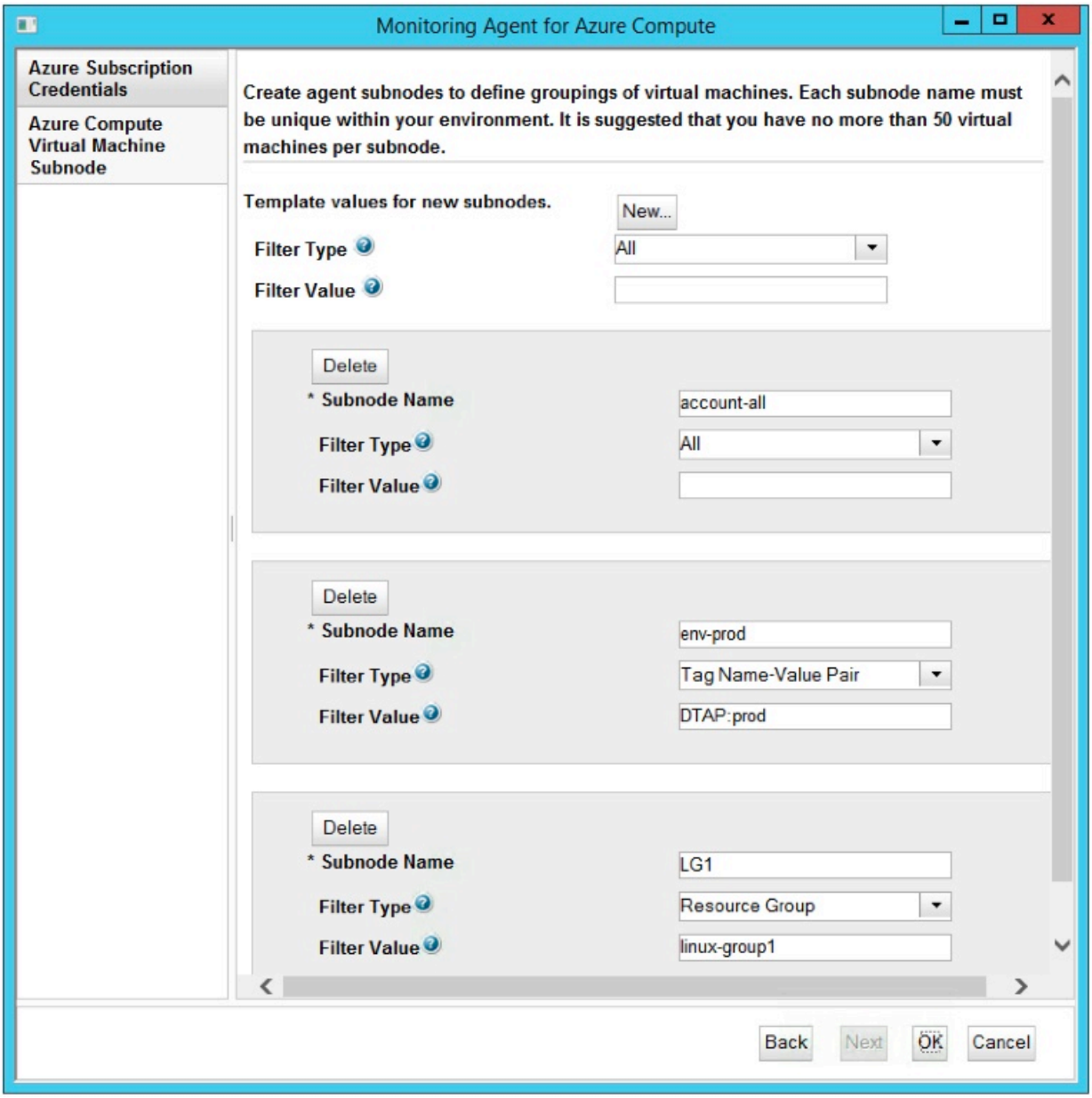


図 *12. Azure Compute* 仮想マシン・サブノードのインスタンス設定を指定するためのウィンドウ。

7. 「**OK**」をクリックして、構成を完了します。
8. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

Azure Compute エージェントのインストール後、エージェントを開始する前に構成する必要があります。Azure Compute エージェントがローカル Linux コンピューターにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで Azure Compute エージェントを構成します。

1. 次のコマンドを実行します。

   ```
   install_dir/bin/azure_compute-agent.sh config instance-name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスに指定する名前です。*instance-name* では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

   例

   ```
   /opt/ibm/apm/agent/bin/azure_compute-agent.sh config azc-inst3
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、215 ページの『Azure Compute エージェントの構成パラメーター』を参照してください。

   要確認 : エージェント・インスタンスを最初に構成するときに、「**Edit 'Azure Compute Virtual Machine Subnode' settings**」というプロンプトが出されたら、少なくとも 1 つのサブノードを追加する必要があります。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/azure_compute-agent.sh start instance-name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/azure_compute-agent.sh start azc-inst3
   ```

例

azc1 という 1 つのサブノード・インスタンスを持つ、azc-inst3 というエージェント・インスタンスを作成します。

```
# ./azure_compute-agent.sh config azc-inst3
Configuring Monitoring Agent for Azure Compute

Edit 'Monitoring Agent for Azure Compute' settings? [1=はい、2=いいえ] (デフォルト: 1): 1

Azure Subscription Credentials :
Credentials required for access to the Azure Subscription.

The ID assigned by Azure for the Subscription that is monitored.
Subscription ID (default is: ): 09x73b6b-bcxb-40x3-92xd-ebx7-EXAMPLE

The tenant ID that is assigned by Azure. Used to log in to the Azure service API.
Tenant ID (default is: ): 75x2e745-e4x4-42x7-94xb-dex1-EXAMPLE

The client ID that is assigned by Azure to identify this agent as an external
application that monitors the Azure compute services.
Client ID (default is: ): 79x2e6c3-7exd-41x2-a9x9-4fx2-EXAMPLE

The secret access key or password that is created by Azure for the client application.
Enter Secret Key/Password (default is: ): hidden

Re-type : Secret Key/Password (default is: ): hidden

Azure Compute Virtual Machine Subnode :

Create agent subnodes to define groupings of virtual machines. Each subnode name
must be unique within your environment. It is suggested that you have no more
than 50 virtual machines per subnode.
```

```
No 'Azure Compute Virtual Machine Subnode' settings available.
Edit 'Azure Compute Virtual Machine Subnode' settings, [1=Add, 2=Edit, 3=Del,
4=Next, 5=Exit] (default is: 5): 1
Subnode Name (default is: ): azc1
The type of filter to be applied.
Filter Type [ 1=All, 2=Tag Name-Value Pair, 3=Resource Group ] (default is: 1): 2

The filter value corresponding to the selected Filter Type. This value can be a
Resource Group or Tag Name-Value Pair, for example Environment¥:Production.
A backslash might appear in the example, do not enter a backslash in the value
you provide.
Filter Value (default is: ): Environment:Production



Azure Compute Virtual Machine Subnode settings: Subnode Name=azc1
Edit 'Azure Compute Virtual Machine Subnode' settings, [1=Add, 2=Edit, 3=Del,
4=Next, 5=Exit] (default is: 5): 5
構成が正常に完了しました。
Automatic start at system initialization has been configured.
Automatic stop at system shutdown has been configured.
You have new mail in /var/spool/mail/root
```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

Azure Compute エージェントをサイレント・モードで構成するには、以下のステップを実行します。

1. テキスト・エディターで、以下のいずれかのパスにある azure_compute_silent_config.txt ファイルを開きます。
   - Linux *install_dir*/samples/azure_compute_silent_config.txt

     例: /opt/ibm/apm/agent/samples/azure_compute_silent_config.txt
   - Windows *install_dir*¥samples¥azure_compute_silent_config.txt

     例: C:¥IBM¥APM¥samples¥azure_compute_silent_config.txt

   ここで、*install_dir* はエージェントのインストール先パスです。
2. azure_compute_silent_config.txt ファイルで、すべての必須パラメーターの値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

   各構成パラメーターの説明については、215 ページの『Azure Compute エージェントの構成パラメーター』を参照してください。

   重要: 少なくとも 1 つのサブノード名に対して、「フィルター・タイプ」と「フィルター値」のパラメーターを有効にして指定する必要があります。
3. azure_compute_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
   - Linux *install_dir*/bin/azure_compute-agent.sh config **instance-name** *install_dir*/samples/azure_compute_silent_config.txt

例: **/opt/ibm/apm/agent/bin/azure_compute-agent.sh config azc-inst3 /opt/ibm/apm/agent/samples/azure_compute_silent_config.txt**

- Windows *install_dir*¥bin¥azure_compute-agent.bat config **instance-name** *install_dir*¥samples¥azure_compute_silent_config.txt

  例: **C:¥IBM¥APM¥bin¥azure_compute-agent.bat config azc-inst3 C:¥IBM¥APM¥samples¥azure_compute_silent_config.txt**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスに指定する名前です。*instance-name* では、英字、アラビア数字、ハイフンマイナス 文字のみを使用してください。詳しくは、171 ページの『複数インスタンス・エージェントの一般的な MSN の形式』の *instancename* を参照してください。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

4. 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/azure_compute-agent.sh start **instance-name**

  例: **/opt/ibm/apm/agent/bin/azure_compute-agent.sh start azc-inst3**

- Windows *install_dir*¥bin¥azure_compute-agent.bat start **instance-name**

  例: **C:¥IBM¥APM¥bin¥azure_compute-agent.bat start azc-inst3**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance-name* はエージェント・インスタンスの名前です。

例

account-all、env-prod、および LG1 という 3 つのサブノードで編集した azure_compute_silent_config.txt。

```
#
# This is a sample configuration response file for agent Azure Compute.
#
# It contains an entry for every configuration property.
# Entries for optional properties that have no default value are included in
# comments.
# Entries for subnode AVM are given a sample subnode instance name of avm1.
# Ensure that all uncommented properties have a value before configuring the
# agent.
#

# Subscription ID: The ID assigned by Azure for the Subscription that is
# monitored.
KAK_SUBSCRIPTION_ID=09x73b6b-bcxb-40x3-92xd-ebx7-EXAMPLE
# Tenant ID: The tenant ID that is assigned by Azure. Used to log in to the
# Azure service API.
KAK_TENANT_ID=75x2e745-e4x4-42x7-94xb-dex1-EXAMPLE
# Client ID: The client ID that is assigned by Azure to identify this agent
# as an external
# application that monitors the Azure compute services.
KAK_CLIENT_ID=79x2e6c3-7exd-41x2-a9x9-4fx2-EXAMPLE
# Secret Key/Password: The secret access key or password that is created by
# Azure for the client application.
KAK_SECRET_PASSWORD=hZxWPq/IOxlnvg/wdxLwTf2Fs3x2sWQV/sCE-EXAMPLE

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
#KAK_FILTER_TYPE.avm1=ALL
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
#KAK_FILTER_VALUE.avm1=

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.account-all=ALL
```

```
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.account-all=

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.env-prod=TAG_NAME_VALUE
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.env-prod=DTAP:prod

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.LG1=RESOURCE_GROUP
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.LG1=linux-group1
```

Azure Compute エージェントの構成パラメーター

Azure Compute エージェントの構成パラメーターを、セクションに従ってグループ化して表に表示します。

1. 215 ページの表 16 - モニター対象の Azure Compute リソースを含む Azure サブスクリプションへのアクセスに必要な資格情報。
2. 216 ページの表 17 - 仮想マシンのグループを定義するためのエージェント・サブノードを作成します。各サブノード名は環境内で固有である必要があります。 サブノードあたりの仮想マシン数は 50 以下にすることが推奨されます。

表 *16. Azure* サブスクリプション資格情報

パラメーター名	説明	サイレント構成ファイルのパラメーター名
サブスクリプション ID	The ID assigned by Azure for the Subscription that is monitored.	**KAK_SUBSCRIPTION_ID**
テナント ID	The tenant ID that is assigned by Azure. Used to log in to the Azure service API.	**KAK_TENANT_ID**
クライアント ID	このエージェントを Azure Compute サービスをモニターする外部アプリケーションとして識別するために Azure により割り当てられるクライアント ID。	**KAK_CLIENT_ID**
秘密鍵/パスワード	The secret access key or password that is created by Azure for the client application. **重要 :** Windows **「秘密鍵/パスワード」**に等号 (=) が含まれている場合、エージェントを再構成するたびに再入力する必要があります。	**KAK_SECRET_PASSWORD**

表 *17. Azure Compute* 仮想マシン・サブノード

パラメーター名	**説明**	**サイレント構成ファイルのパラメーター名**
サブノード名	データ収集のための Azure Compute サブノードの名前。例えば、*azc1* です。サブノード名は環境内で固有である必要があります。 この別名は、管理対象システム名 (MSN) の一部であり、Cloud APM コンソール内でモニター対象環境を視覚的に識別するために使用されます。 **注 :** この別名は、Azure Compute サブノード・インスタンスを表す任意のものにすることができますが、以下の制限があります。エージェント・サブノード・インスタンス名の作成には、英字 (a-z、A-Z)、アラビア数字 (0-9)、ハイフンマイナス文字 (-)、および下線文字 (_) を使用できます。Azure Compute サブノード名の最大長は 25 文字です。	以下の各パラメーターには、ピリオド (.) に続けてエージェントの**サブノード名**をサフィックスとして使用する必要があります。サブノードの各パラメーターでは、**サブノード名**が同じでなければなりません。新規エージェント・サブノード・インスタンスのパラメーターのセットでは、固有の**サブノード名**を使用する必要があります。例えば、以下のパラメーター名の *.subnode_name* の代わりに、あるサブノード・インスタンスでは *.azc1* を使用し、別のエージェント・インスタンスでは *.azc2* を使用できます。
フィルター・タイプ	The type of filter to be applied.	**KAK_FILTER_TYPE.subnode_name** 有効な値を以下に示します。 **ALL** すべて **TAG_NAME_VALUE** タグ名と値のペア **RESOURCE_GROUP** リソース・グループ

表 17. *Azure Compute* 仮想マシン・サブノード (続き)		
パラメーター名	**説明**	**サイレント構成ファイルのパラメーター名**
フィルター値 (Filter Value)	選択されたフィルター・タイプに対応する**フィルター値**。この値は「**リソース・グループ**」または「**タグ名と値のペア**」にできます。**「フィルター・タイプ」**が「**すべて**」の場合は、値を空にします。コマンド・ライン構成の場合、表示される例に円記号 (¥) が表示される場合があります。指定する値に円記号は入力しないでください。 フィルター・タイプとフィルター値のペアの例: • すべての仮想マシンをモニターする Azure Compute サブノード。フィルター値は空のままにします。「フィルター・タイプ」が「**すべて**」の場合、フィルター値は不要のため無視されます。 – フィルター・タイプ: **すべて** – フィルター値: • タグ名が DTAP であり、タグ値がストリング prod と一致するすべての仮想マシンをモニターする Azure Compute サブノード。 – フィルター・タイプ: **タグ名と値のペア** – フィルター値: DTAP:prod • ストリング linux-group1 と一致するリソース・グループ・プロパティーを持つすべての仮想マシンをモニターする Azure Compute サブノード。 – フィルター・タイプ: **リソース・グループ** – フィルター値: linux-group1	**KAK_FILTER_VALUE.subnode_name**

Cassandra モニターの構成

Cassandra エージェントを構成して、Cassandra データベースの正常性をモニターするためにクラスター内のノードからエージェントがデータを収集できるようにする必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for Cassandra agent を参照してください。

このタスクについて

Cassandra エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェ

ント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

- Windows システム上でエージェントを構成するには、「IBM Cloud Application Performance Management」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**デフォルトを使用して構成**」をクリックします。

   「**Monitoring Agent for Cassandra**」ウィンドウが開きます。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。
4. 「**Monitoring Agent for Cassandra**」ウィンドウで、構成パラメーターの値を指定し、「**OK**」をクリックします。

   構成パラメーターについては、219 ページの『エージェントの構成パラメーター』を参照してください。
5. 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks® の IBM Cloud APM Forum を参照してください。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
   例: `/opt/ibm/apm/agent/bin`
2. 以下のコマンドを実行します。ここで、instance_name は、インスタンスに付ける名前です。
   `./cassandra-agent.sh config instance_name`
3. コマンド行に以下のメッセージが表示されたら、1 と入力して Enter キーを押します。

   ```
   Edit 'Monitoring Agent for Cassandra' setting? [1=Yes, 2=No]
   ```
4. プロンプトが出されたら、構成パラメーターの値を指定します。

   構成パラメーターについては、219 ページの『エージェントの構成パラメーター』を参照してください。
5. 以下のコマンドを実行して、エージェントを開始します。

   `./cassandra-agent.sh start instance_name`

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システムおよび Windows システム上で Cassandra エージェントを構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. テキスト・エディターで、次の場所にあるサイレント構成ファイルを開き、すべてのパラメーターの値を指定します。

   Windows *install_dir*¥samples¥cassandra_silent_config_windows.txt
   Linux *install_dir*¥samples¥cassandra_silent_config_UNIX.txt

   Windows C:¥IBM¥APM¥samples

   Linux /opt/ibm/apm/agent/samples

   構成パラメーターについては、219 ページの『エージェントの構成パラメーター』を参照してください。
2. コマンド行で、パスを *install_dir*¥bin に変更します。
3. 次のコマンドを実行します。

   Windows cassandra-agent.bat config *instance_name install_dir*¥samples¥cassandra_silent_config_windows.txt
   Linux cassandra-agent.sh config *instance_name install_dir*¥samples¥cassandra_silent_config_UNIX.txt
4. エージェントを開始します。

   Windows 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。
   Linux 次のコマンドを実行します。./cassandra-agent.sh start *instance_name*

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

エージェントの構成パラメーター

Cassandra エージェントの構成中に、IP アドレスおよび JMX_PORT などのパラメーターのデフォルト値を変更できます。

以下の表で、Cassandra エージェントの構成パラメーターについて詳しく説明します。

表 18. 構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
インスタンス名	このフィールドのデフォルト 値は、「**固有のインスタンス名を入力してください**」フィールドに指定する値と同じです。	はい
IP アドレス	クラスター内の任意のノードの IP アドレス。	はい
JMX_PORT	モニターを有効にする JMX ポート番号。 **重要 :** JMX_Port、JMX_Username、および JMX_Password は、クラスター全体で指定してください。エージェントをクラスターに接続する際に使用するノードが機能していない場合、エージェントは、同じパラメーターを使用して、クラスター内の別のノードを介してデータを収集できます。	はい
JMX_Username	JMX にアクセスするためのユーザー名。	いいえ
JMX_Password	JMX にアクセスするためのパスワード。	いいえ

Cisco UCS モニターの構成

Monitoring Agent for Cisco UCS は、Cisco UCSM に接続することにより、Cisco UCS Virtual Infrastructure をモニターします。エージェントが Cisco UCS データを収集できるように、Cisco UCS エージェントを構成する必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Cisco UCS エージェント を参照してください。
- Cisco UCSM インフラストラクチャーに接続するユーザーが確実に aaa 特権または管理者特権を持つようにします。aaa 特権または管理者特権を持つ既存のユーザー ID を使用するか、新規ユーザー ID を作成してください。
- Cisco UCS エージェントが、SSL エージェントを使用する Cisco UCS データ・ソースと通信するように構成されている場合は、エージェントの証明書トラストストアに各データ・ソースの SSL 証明書を追加してください。Cisco UCS データ・ソースとの SSL 通信の使用可能化について詳しくは、225 ページの『Cisco UCS データ・ソースとの SSL 通信の使用可能化』を参照してください。

このタスクについて

Cisco UCS エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で始動する必要があります。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン』を参照してください。以前のエージェント・リリースに関する資料にアクセスするには、次の表を参照してください。

表 19. エージェントのバージョンおよび資料	
Cisco UCS エージェントのバージョン	**資料**
7.2.0.4, 7.2.0.3	IBM Cloud Application Performance Management **注 :** リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。

表 19. エージェントのバージョンおよび資料 (続き)	
Cisco UCS エージェントのバージョン	**資料**
7.2.0.2	IBM Performance Management 8.1.3 **注 :** リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。
7.2.0.1	IBM Performance Management 8.1.2 **注 :** リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。

構成属性は、モニター対象となる Cisco UCS Infrastructure を定義します。属性は、Cisco UCSM 1.4 以降への接続を定義します。リモートのモニター・ホスト・システム上にモニター・エージェントの複数のインスタンスを構成できます。特定の Cisco UCS Infrastructure をモニターするために個別のインスタンスを作成することもできます。

Cisco UCS エージェント がインストールされたら、エージェントを開始できます。ただし、すべてのエージェント属性のデータを表示するには、エージェントを手動で構成する必要があります。

- Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

Cisco UCS エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Cisco UCS**」を右クリックしてから「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェントを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェントを再度構成するには、「**再構成**」をクリックします。
3. 「Monitoring Agent for Cisco UCS」ウィンドウで、以下のステップを実行します。

   a) Cisco UCS エージェント・インスタンスの固有の名前を入力し、「**OK**」をクリックします.

   b) 「**CONFIG**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

   c) 「**LOG_CONFIG**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

   「Monitoring Agent for Cisco UCS」ウィンドウの各タブにある構成パラメーターについて詳しくは、以下のトピックを参照してください。

   - 224 ページの『エージェントの構成パラメーター』
   - 225 ページの『データ・プロバイダーの構成パラメーター』
4. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Cisco UCS**」を右クリックし、「**開始**」をクリックします。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 大規模な Cisco UCS 環境をモニターする場合、Java™ データ・プロバイダー用のヒープ・サイズを増やす必要がある場合があります。詳しくは、226 ページの『Java ヒープ・サイズの増大』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで Cisco UCS エージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで、cisco_ucs_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
     - Linux *install_dir*/samples/cisco_ucs_silent_config.txt

       例: /opt/ibm/apm/agent/samples/cisco_ucs_silent_config.txt
     - Windows *install_dir*¥samples¥cisco_ucs_silent_config.txt

       例: C:¥IBM¥APM¥samples¥cisco_ucs_silent_config.txt

  b) cisco_ucs_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

     構成パラメーターについては、以下のトピックを参照してください。
     - 224 ページの『エージェントの構成パラメーター』
     - 225 ページの『データ・プロバイダーの構成パラメーター』

  c) cisco_ucs_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
     - Linux *install_dir*/bin/cisco_ucs-agent.sh config *instance_name* *install_dir*/samples/cisco_ucs_silent_config.txt

       例: **/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh config instance_name /opt/ibm/apm/agent/samples/cisco_ucs_silent_config.txt**
     - Windows *install_dir*¥bin¥cisco_ucs-agent.bat config *instance_name* *install_dir*¥samples¥cisco_ucs_silent_config.txt

       例: **C:¥IBM¥APM¥bin¥cisco_ucs-agent.bat config instance_name C:¥IBM¥APM¥samples¥cisco_ucs_silent_config.txt**

       表の見方

instance_name
インスタンスに指定する名前。

install_dir
エージェントのインストール先パス。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/cisco_ucs-agent.sh start *instance_name*

  例: **/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh start instance_name**

- Windows *install_dir*¥bin¥cisco_ucs-agent.bat start *instance_name*

  例: **C:¥IBM¥APM¥bin¥cisco_ucs-agent.bat start instance_name**

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 大規模な Cisco UCS 環境をモニターする場合、Java™ データ・プロバイダー用のヒープ・サイズを増やす必要がある場合があります。詳しくは、226 ページの『Java ヒープ・サイズの増大』を参照してください。

プロンプトへの応答によるエージェントの構成

Linux オペレーティング・システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

- スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

  a) コマンド行に以下のコマンドを入力します。

  install_dir/bin/cisco_ucs-agent.sh config *instance_name*

  例: **/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh config instance_name**

  表の見方

  instance_name
  インスタンスに指定する名前。

  install_dir
  エージェントのインストール先パス。

  b) 以下のトピックを参照して、プロンプトに応答します。

  - 224 ページの『エージェントの構成パラメーター』
  - 225 ページの『データ・プロバイダーの構成パラメーター』

  c) 以下のコマンドを実行して、エージェントを開始します。

  install_dir/bin/cisco_ucs-agent.sh start *instance_name*

  例: **/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh start instance_name**

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 大規模な Cisco UCS 環境をモニターする場合、Java™ データ・プロバイダー用のヒープ・サイズを増やす必要がある場合があります。詳しくは、226 ページの『Java ヒープ・サイズの増大』を参照してください。

エージェントの構成パラメーター

Cisco UCS エージェントを構成する際に、インスタンス名や SSL 検証証明書などのパラメーターのデフォルト値を変更できます。

以下の表に、Cisco UCS エージェント の構成パラメーターの詳細な説明を示します。

表 *20. Cisco UCS* エージェントの構成パラメーターの名前と説明

パラメーター名	**説明**	**必須フィールド**
インスタンス名	The name of the instance. **制約事項 :** 「**インスタンス名**」フィールドでは、初めてエージェントを構成したときに指定したインスタンスの名前が表示されます。エージェントを再構成する場合に、エージェントのインスタンス名を変更することはできません。	はい
URL	Cisco UCS Manager の URL。 Cisco UCS マネージャーの URL を設定するには、http://ip_address/nuova の形式で URL を入力します。	はい
ユーザー名	Cisco UCS Manager の 管理者ユーザー名。	はい
パスワード	Cisco UCS Manager の 管理者パスワード。	はい
パスワードの確認入力	「**パスワード**」フィールドに入力したものと同じパスワード。	はい
SSL トラストストア・ファイル・パス (SSL truststore filepath)	Secure Sockets Layer (SSL) トラストストア・ファイルのパス。 ネットワークを介した通信に SSL を使用する際にエージェントが SSL 証明書を検証するようにする場合は、Secure Sockets Layer (SSL) トラストストア・ファイルがある場所を指定します。	はい
SSL 証明書の検証	エージェントが SSL を使用してネットワークで通信するときに、エージェントが SSL 証明書を検証するかどうかを示すブール値。 ネットワーク上で SSL を使用して通信するときにエージェントが SSL 証明書を検証する必要がある場合は、値を Yes に設定します。エージェントが SSL 証明書を検証しないようにする場合は、値を No に設定します。 **ヒント :** Cisco UCS データ・ソースとの SSL 通信の使用可能化について詳しくは、225 ページの『Cisco UCS データ・ソースとの SSL 通信の使用可能化』を参照してください。	はい

データ・プロバイダーの構成パラメーター

Cisco UCS エージェントを構成する際に、データ・プロバイダーのログ・ファイルの最大数、ログ・ファイルの最大サイズ、ログ・ファイルに含まれる詳細レベルなど、データ・プロバイダーのパラメーターのデフォルト値を変更できます。

以下の表に、データ・プロバイダーの構成パラメーターの詳細な説明を示します。

表 *21.* データ・プロバイダーの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
データ・プロバイダー・ログ・ファイルの最大数	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。	はい
各データ・プロバイダー・ログの最大サイズ (KB)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。	はい
データ・プロバイダー・ログの詳細レベル	データ・プロバイダーによって作成されるログ・ファイルに含めることができる詳細のレベル。デフォルト値は INFO です。有効な値は、OFF、SEVERE、WARNING、INFO、FINE、FINER、FINEST、および ALL です。	はい

Cisco UCS データ・ソースとの SSL 通信の使用可能化

Cisco UCS エージェントは、SSL を使用して Cisco UCS データ・ソースとセキュアな通信ができるように構成できます。この構成では、データ・ソースの SSL 証明書をエージェントの証明書トラストストアに追加する必要があります。

このタスクについて

重要 : 以下の情報は、SSL 証明書を検証するようにエージェントを構成する場合にのみ当てはまります。

SSL 証明書の検証がオフになっていると、Cisco UCS エージェントは、Cisco UCS データ・ソースの SSL 証明書が期限切れである、信頼されていない、または無効である場合でも Cisco UCS データ・ソースに接続できます。ただし、SSL 証明書の検証をオフにするとセキュリティーで保護されなくなる可能性があるため、実行には注意が必要です。

Cisco UCS データ・ソースで、共通認証局 (Verisign、Entrust、Thawte など) によって署名された SSL 証明書を使用する場合は、Cisco UCS エージェントの証明書トラストストアに証明書を追加する必要はありません。しかし、データ・ソースで、共通認証局によって署名されていない証明書を使用する場合 (これがデフォルトです) は、エージェントが正常に接続し、データを収集するためには、トラストストアに証明書を追加する必要があります。

手順

1. データ・ソースからエージェント・コンピューターに証明書ファイルをコピーします。
2. エージェント・コンピューターで、選択したディレクトリーに証明書ファイルを配置します。証明書ファイルを上書きしないでください。追加する証明書ごとに固有のファイル名およびラベルを使用してください。
3. keytool コマンドを使用して、データ・ソース証明書をエージェントの証明書トラストストアに追加します。

```
keytool -import -noprompt -trustcacerts -alias CertificateAlias -file
CertificateFile -keystore Truststore -storepass TruststorePassword
```

表の見方

CertificateAlias

エージェントの証明書トラストストアに追加される各証明書の固有の参照です。例えば、*datasource.example.com* からの証明書に対する適切な別名は *datasource* です。

CertificateFile

トラストストアに追加する Cisco UCS データ・ソース証明書への完全パスおよびファイル名。

トラストストア

Complete path and file name to the Cisco UCS エージェント certificate database. 次のパスおよびファイル名を使用します。

- Windows (64 ビット) *install_dir*¥tmaitm6_x64¥kv6.truststore
- Linux (64 ビット) install_dir/lx8266/vm/etc/kv6.truststore

TruststorePassword

Cisco UCS エージェント・トラストストアのデフォルトのパスワードは ITMFORVE です。このパスワードを変更するには、使用するツールについての Java ランタイムの資料を参照してください。

重要: keytool コマンドを使用するには、Java ランタイムの bin ディレクトリーが PATH になければなりません。以下のコマンドを使用します。

- Windows (64 ビット) set PATH=%PATH%;install_dir¥java¥java70_x64¥jre¥bin
- Linux (64 ビット) PATH="$PATH":/opt/ibm/apm/agent/JRE/lx8266/bin

4. すべてのデータ・ソース証明書を追加した後で、モニター・エージェントを開始します。

Java ヒープ・サイズの増大

大規模な Cisco UCS 環境をモニターする場合は、Cisco UCS エージェントを構成した後で、Java™ データ・プロバイダーのヒープ・サイズを増やす必要がある場合があります。

このタスクについて

Java データ・プロバイダーのデフォルトのヒープ・サイズは 256 メガバイトです。大規模な Cisco UCS 環境では、以下の問題が発生した場合に、ヒープ・サイズを増やす必要がある場合があります。

- javacore の問題が原因で Java データ・プロバイダーが停止し、javacore.*date.time.number*.txt という名前のファイルが CANDLEHOME¥tmaitm6_x64 ディレクトリー内に作成される。
- javacore.*date.time.number*.txt ファイルに java/lang/OutOfMemoryError というストリングが含まれている。

手順

- Windows
  ヒープ・サイズとして 1 GB の値を設定するには、以下の手順を実行します。
  1. %CANDLE_HOME%¥TMAITM6_x64¥kv6_data_provider.bat ファイルを開きます。
  2. KV6_JVM_ARGS="$KV6_CUSTOM_JVM_ARGS... で始まる行の前に次の行を追加します。

     ```
     SET KV6_CUSTOM_JVM_ARGS=-Xmx1024m
     ```

  3. エージェントを再始動します。
- Linux
  ヒープ・サイズとして 1 GB の値を設定するには、以下の手順を実行します。
  1. $CANDLEHOME/lx8266/vm/bin/kv6_data_provider.sh ファイルを開きます。
  2. KV6_JVM_ARGS="$KV6_CUSTOM_JVM_ARGS... で始まる行の前に次の行を追加します。

     ```
     KV6_CUSTOM_JVM_ARGS=-Xmx1024m
     ```

3. エージェントを再始動します。

Citrix Virtual Desktop Infrastructure モニターの構成

Citrix VDI エージェントには、配信グループ、カタログ、アプリケーション、デスクトップ、ユーザー、セッションなどの Citrix XenDesktop リソースまたは XenApp リソースについてモニターする中央ポイントが用意されています。エージェントを使用する前に、配信コントローラーを介してデータを収集するようにエージェントを構成する必要があります。

始める前に

- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、Citrix VDI エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Citrix VDI エージェント を参照してください。
- 次の情報が使用可能であることを確認します。
  - 接続先にする配信コントローラーのホスト名。
  - OData ユーザー名、パスワード、およびドメイン。
  - Windows イベント・ログ・イベントおよび PowerShell メトリックの取得を有効にする場合は、PowerShell ユーザー名、パスワード、ドメイン、PowerShell ポート、SSL 検証タイプ、および認証メカニズム。
- エージェント・オペレーター・ユーザーのアカウントが、少なくとも Citrix 読み取り専用管理者特権を持つようにします。『Citrix 読み取り専用管理者特権の有効化』を参照してください。
- Citrix VDI エージェント バージョン 8.1.3.1 以降では、Windows イベント・ログ・イベントを取得するための機能が使用可能になっています。すべての Desktop Delivery Controller (DDC) マシンおよび Virtual Delivery Agent (VDA) マシンから Windows イベント・ログ・イベントを取得するには、エージェント・インスタンスの構成中に指定したユーザー・アカウントに対してリモート PowerShell のアクセスを有効にする必要があります。以下のステップに従い、エージェントがこの機能を実行できるようにします。
  1. エージェント・インスタンスの構成で指定されているユーザーとして Windows コンピューターにログインする。
  2. 以下の PowerShell コマンドを実行します (*vda_system* は電源がオンになっている VDA マシンの名前)。
     ```
     Get-WinEvent -FilterHashtable
     @{ProviderName='Citrix*';LogName='Citrix*';StartTime=((Get-
     Date).AddDays(-10))} -ComputerName vda_system
     ```
- モニター対象の環境に対して以下のロード・バランシング・ポリシーを必ず有効にしてください。
  - CPU 使用量
  - ディスク使用量
  - メモリー使用量

  これらのポリシーは Citrix Studio アプリケーションから構成できます。

このタスクについて

Citrix VDI エージェントは複数インスタンス・エージェントです。少なくとも 1 つのインスタンスを作成し、エージェント・インスタンスを手動で開始する必要があります。

XenApp サーバーの構成は、XenDesktop サーバーと同じです。構成パラメーター名または説明で「XenDesktop」についてのみ言及されている場合、それは XenApp にも適用されます。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用して Windows システムでエージェントを構成します。

- 229 ページの『Windows システムでのエージェントの 構成』.
- 233 ページの『サイレント応答ファイルを使用したエージェントの構成』.

2. 応答を要求するプロンプトを出すスクリプト、またはサイレント応答ファイルを使用して、Linux システムでエージェントを構成します。
   - 232 ページの『プロンプトへの応答によるエージェントの構成』.
   - 233 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Citrix 読み取り専用管理者特権の有効化

Citrix VDI エージェントでは、エージェント・オペレーター・ユーザーのアカウントが、少なくとも Citrix 読み取り専用管理者特権を持っている必要があります。

このタスクについて

Citrix Delegated Admin PowerShell スナップインがインストールされているコンピューターからリモートでこれらのステップを実施するには、`AdminAddress` パラメーターを使用します。例えば、ステップ 2 のコマンドは以下のようになります。
`New-AdminAdministrator -Name "YOURDOMAIN¥NewAdmin" -AdminAddress "controller1.YOURDOMAIN.com"`。ここで、*YOURDOMAIN* はネットワークのドメイン・ネーム、*NewAdmin* は Citrix の管理権限を付与されるユーザー・アカウント、*controller1.YOURDOMAIN.com* は Citrix サイト・サーバーの完全修飾ドメイン・ネームです。

手順

1. 既存の Citrix 管理者アカウントで PowerShell のセッションを開始します。
2. Citrix XenApp または XenDesktop のサイトを管理するための Delegated Admin PowerShell スナップインをロードします。

   `(Add-PSSnapin Citrix.DelegatedAdmin.Admin.V1)`
3. エージェント・オペレーター・ユーザーのアカウントを Citrix サイト管理者として追加します。

   `New-AdminAdministrator -Name "YOURDOMAIN¥NewAdmin"`

   ここで、*YOURDOMAIN* はネットワークのドメイン・ネーム、*NewAdmin* は Citrix の管理権限を付与されるユーザー・アカウントです。
4. *NewAdmin* に割り当てることができる役割およびスコープを照会します。

```
Get-AdminRole
Get-AdminScope
```

5. 読み取り専用の管理者権限を含む役割をエージェント・オペレーター・ユーザーのアカウントに割り当てます。

   `Add-AdminRight -Administrator "YOURDOMAIN¥NewAdmin" -Role "Read Only Administrator" -Scope "All"`

構文の内容は次のとおりです。

- *YOURDOMAIN* はネットワークのドメイン・ネームです。
- *NewAdmin* は Citrix の管理権限を付与されるユーザー・アカウントです。
- *Read Only Administrator* は割り当てる Citrix サイト管理者の役割です。
- *All* は割り当てる Citrix サイト管理者スコープです。

6. 管理者の追加および変更を確認します。

   ```
   Get-AdminAdministrator -Name "YOURDOMAIN¥NewAdmin"
   ```

   ここで、*YOURDOMAIN* はネットワークのドメイン・ネーム、*NewAdmin* は Citrix の管理権限を付与されるユーザー・アカウントです。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Citrix VDI エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for Citrix Virtual Desktop Infrastructure**」 テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、下線文字、およびマイナス文字のみを使用してください。例: `vdi_inst2`

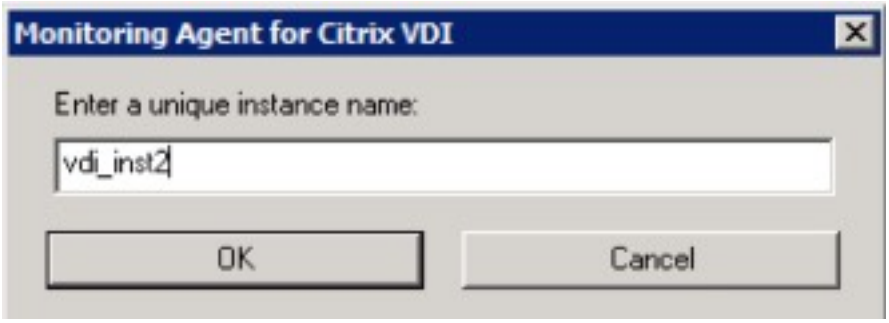


図 *13.* 固有のインスタンス名を入力するウィンドウ

4. エージェント・インスタンス名ウィンドウで「**次へ**」をクリックします。

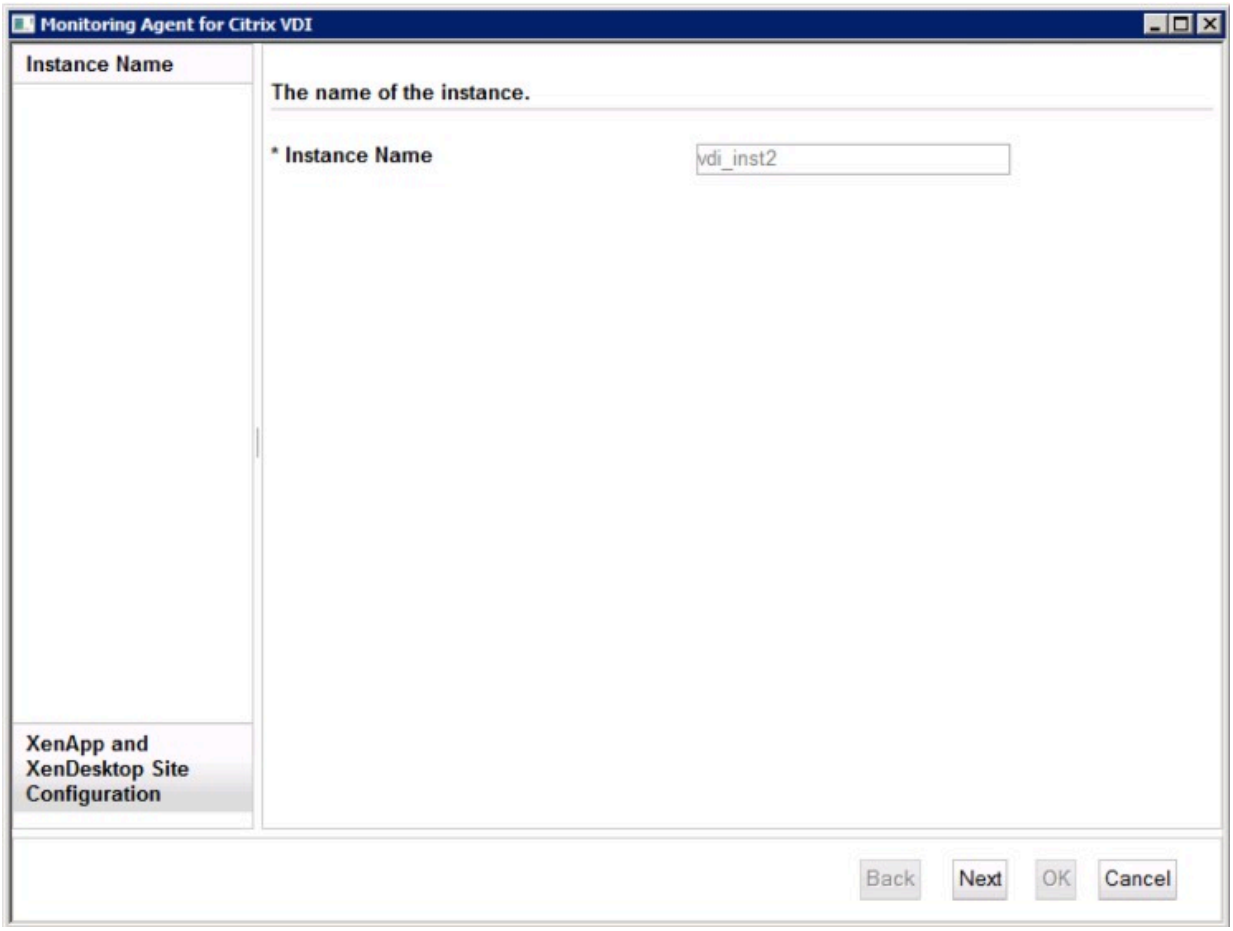


図 *14.* エージェント・インスタンス名ウィンドウ。

5. 「**XenApp および XenDesktop サイト構成 (XenApp and XenDesktop Site Configuration)**」インスタンス・テンプレート設定を入力します。

   注 : このセクションは、XenApp および XenDesktop サイト・インスタンス構成ではありません。ステップ 6 で実際の XenApp または XenDesktop サイト・インスタンス構成を追加するときにデフォルト値として使用される内容を設定するためのテンプレート・セクションです。

   各構成パラメーターの説明については、235 ページの表 22 を参照してください。

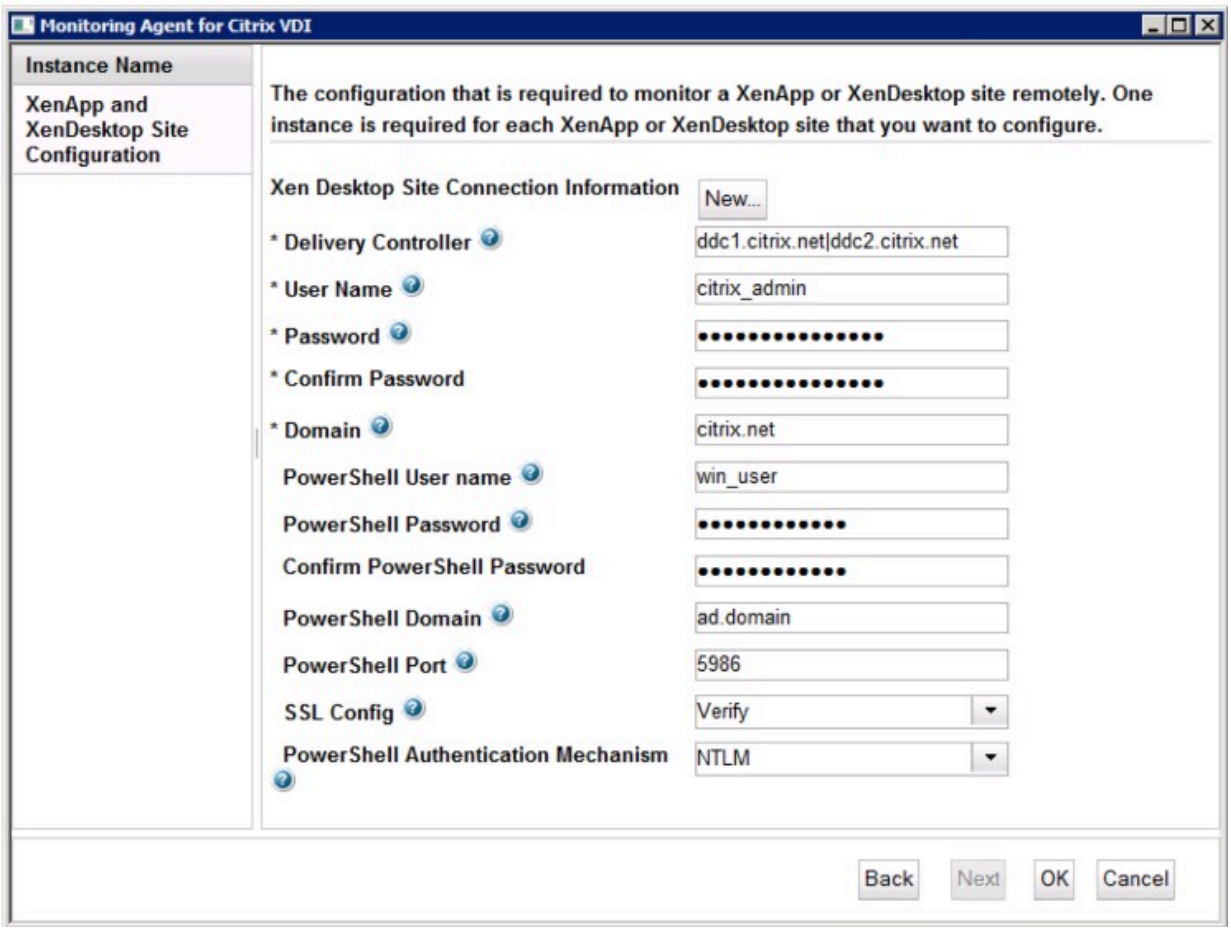


図 *15. XenApp* サイトまたは *XenDesktop* サイトのインスタンス・テンプレート設定を指定するウィンドウ。

6. 「**新規**」をクリックし、XenApp または XenDesktop サイト・インスタンス設定を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、235 ページの表 22 を参照してください。

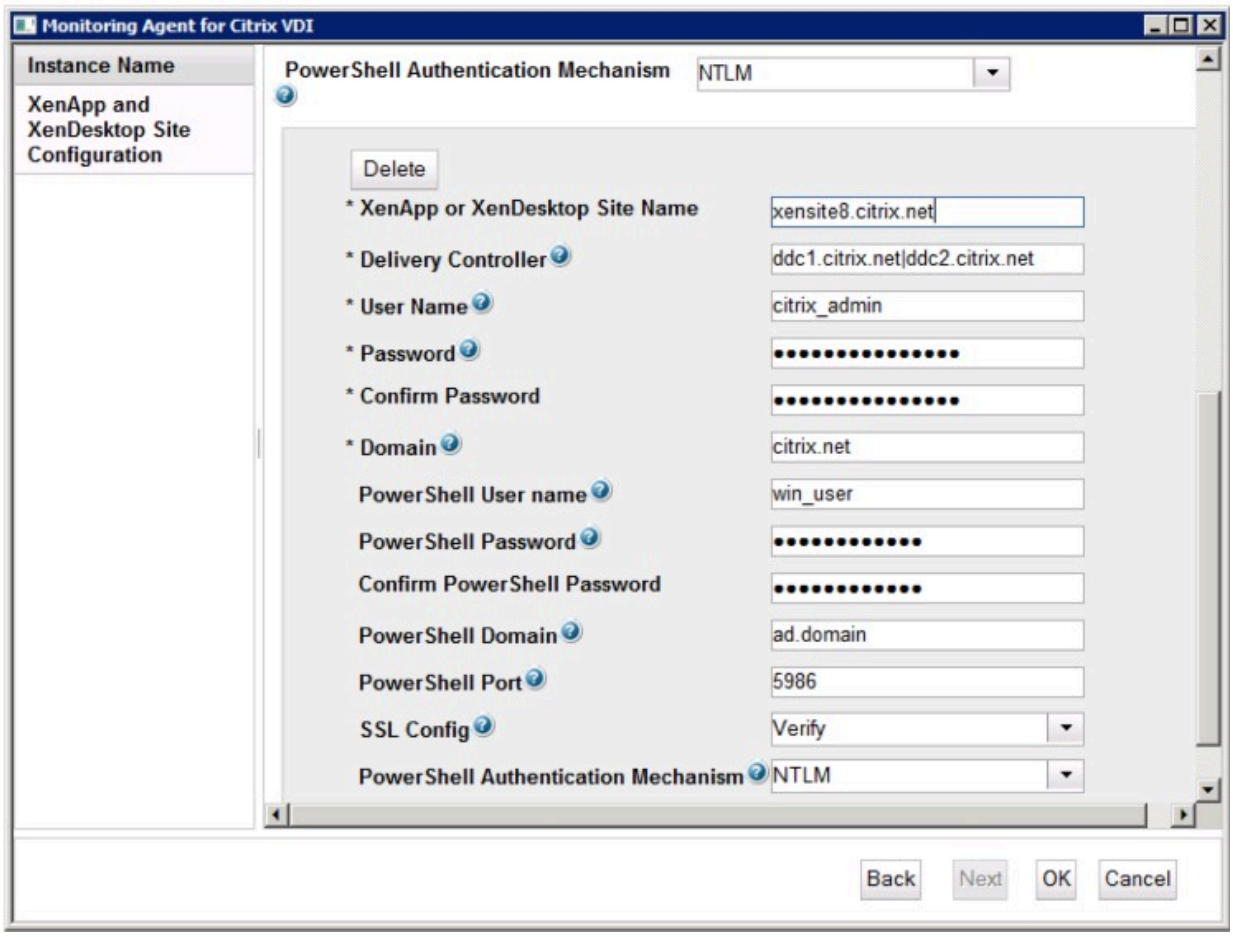


図 *16. XenApp* サイトまたは *XenDesktop* サイトのインスタンス設定を指定するウィンドウ。

注 :「PowerShell ユーザー名 (PowerShell User name)」パラメーターおよび以降のすべての PowerShell パラメーターは、236 ページの『Windows イベントおよび PowerShell メトリックのモニターの有効化』を行う場合に限り必要です。これらの拡張環境変数は、モニター対象システムに大きな負荷をかけるため、デフォルトではオフです。

注 : 新規の各 XenApp サイト・インスタンスまたは XenDesktop サイト・インスタンスに対して**「SSL 構成 (SSL Config)」**パラメーターおよび**「PowerShell 認証メカニズム (PowerShell Authentication Mechanism)」**パラメーターが正しく設定されていることを確認します。問題がある場合、テンプレート値の代わりにデフォルト値が設定されます。

7. 「**OK**」をクリックして、構成を完了します。
8. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

Citrix VDI エージェントのインストール後、エージェントを開始する前に構成する必要があります。Citrix VDI エージェントがローカル Linux マシンにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで Citrix VDI エージェントを構成します。

1. 次のコマンドを実行します。

   ```
   install_dir/bin/citrixvdi-agent.sh config instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/citrixvdi-agent.sh config vdi_inst01
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、234 ページの『Citrix VDI エージェントの構成パラメーター』を参照してください。

   注：「PowerShell ユーザー名 (PowerShell User name)」パラメーターおよび以降のすべての PowerShell パラメーターは、236 ページの『Windows イベントおよび PowerShell メトリックのモニターの有効化』を行う場合に限り必要です。これらの拡張環境変数は、モニター対象システムに大きな負荷をかけるため、デフォルトではオフです。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/citrixvdi-agent.sh start instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/citrixvdi-agent.sh start vdi_inst01
   ```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- 以下の手順を実行して、Citrix VDI エージェントをサイレント・モードで構成します。
  a) テキスト・エディターで、以下のいずれかのパスにある citrixvdi_silent_config.txt ファイルを開きます。
  - Linux *install_dir*/samples/citrixvdi_silent_config.txt

    例: /opt/ibm/apm/agent/samples/citrixvdi_silent_config.txt
  - Windows *install_dir*¥samples¥citrixvdi_silent_config.txt

    例: C:¥IBM¥APM¥samples¥citrixvdi_silent_config.txt

ここで、*install_dir* はエージェントのインストール先パスです。

b) `citrixvdi_silent_config.txt` ファイルで、すべての必須パラメーターの値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

各構成パラメーターの説明については、234 ページの『Citrix VDI エージェントの構成パラメーター』を参照してください。

注：「PowerShell ユーザー名 (PowerShell User name)」パラメーターおよび以降のすべての PowerShell パラメーターは、236 ページの『Windows イベントおよび PowerShell メトリックのモニターの有効化』を行う場合に限り必要です。これらの拡張環境変数は、モニター対象システムに大きな負荷をかけるため、デフォルトではオフです。

c) `citrixvdi_silent_config.txt` ファイルを保存して閉じ、次のコマンドを実行します。

- Linux *install_dir*`/bin/citrixvdi-agent.sh config` **instance_name** *install_dir*`/samples/citrixvdi_silent_config.txt`

  例: **/opt/ibm/apm/agent/bin/citrixvdi-agent.sh config vdi_inst01 /opt/ibm/apm/agent/samples/citrixvdi_silent_config.txt**

- Windows *install_dir*`¥bin¥citrixvdi-agent.bat config` **instance_name** *install_dir*`¥samples¥citrixvdi_silent_config.txt`

  例: **C:¥IBM¥APM¥bin¥citrixvdi-agent.bat config vdi_inst01 C:¥IBM¥APM¥samples¥citrixvdi_silent_config.txt**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

重要：サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*`/bin/citrixvdi-agent.sh start` **instance_name**

  例: **/opt/ibm/apm/agent/bin/citrixvdi-agent.sh start vdi_inst01**

- Windows *install_dir*`¥bin¥citrixvdi-agent.bat start` **instance_name**

  例: **C:¥IBM¥APM¥bin¥citrixvdi-agent.bat start vdi_inst01**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

Citrix VDI エージェントの構成パラメーター

Citrix VDI エージェントの構成パラメーターを表に表示します。

1. Citrix VDI エージェント設定 - Citrix VDI エージェント環境設定。

表 *22. Citrix VDI* エージェント設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
XenApp または XenDesktop サイト名	XenApp または XenDesktop サイト・エージェント・インスタンスを識別する名前を指定します。 例えば、*vdi_inst2* です。 **注 :** この別名は、WebLogic サーバー・エージェント・インスタンスを表す任意のものにすることができますが、以下の制限があります。接続名に使用できるのは、文字、アラビア数字、下線文字、およびマイナス文字のみです。 接続名の最大長は 25 文字です。	以下の各パラメーターには、インスタンス名サフィックスを付加する必要があります。このサフィックスは、エージェント・インスタンスのどのパラメーターでも同じです。新規エージェント・インスタンスのパラメーターのセットでは、固有のインスタンス名を使用する必要があります。例えば、以下のパラメーター名で *.instance_name* の代わりに、あるエージェント・インスタンスでは *.vdi1* を使用し、別のエージェント・インスタンスでは *.vdi2* を使用する場合があります。
配信コントローラー	配信コントローラーのホスト名または IP アドレス。複数の DDC が 1 つのクラスターでセットアップされている場合、配信コントローラーの「|」で区切ったリストを指定できます。	**KVD_XDS_DELIVERY_CONTROLLER.instance_name**
ユーザー名	指定の XenApp または XenDesktop 配信コントローラーの OData API による認証で使用するユーザー名。	**KVD_XDS_ODATA_USERNAME.instance_name**
パスワード	指定の XenApp または XenDesktop 配信コントローラーの OData API による認証で使用するパスワード。	**KVD_XDS_ODATA_PASSWORD.instance_name**
ドメイン	指定の XenApp または XenDesktop 配信コントローラーの OData API による認証で使用するドメイン。	**KVD_XDS_ODATA_DOMAIN.instance_name**
PowerShell ユーザー名	リモート VDA および DDC マシンに対する PowerShell 呼び出しのための認証で使用するユーザー名。 **注 :** これ以降のすべての PowerShell パラメーターは、236 ページの『Windows イベントおよび PowerShell メトリックのモニターの有効化』を行う場合に限り必要です。これらの拡張環境変数は、モニター対象システムに大きな負荷をかけるため、デフォルトではオフです。	**KVD_XDS_POWERSHELL_USERNAME.instance_name**
PowerShell パスワード (PowerShell Password)	指定の PowerShell ユーザー名に関連付けられているパスワード。	**KVD_XDS_POWERSHELL_PASSWORD.instance_name**
PowerShell ドメイン (PowerShell Domain)	指定の PowerShell ユーザー名に関連付けられているドメイン。	**KVD_XDS_POWERSHELL_DOMAIN.instance_name**

表 22. *Citrix VDI* エージェント設定 *(続き)*		
パラメーター名	**説明**	**サイレント構成ファイルのパラメーター名**
PowerShell ポート (PowerShell Port)	WinRm による使用のために開く SSL ポート。 PowerShell のデフォルト・リモート接続ポートは、HTTP の場合は 5985 で、HTTPS の場合は 5986 です。	**KVD_XDS_POWERSHELL_PORT.instance_name**
SSL 要件 (SSL Requirement)	環境に必要な SSL オプションを選択します。	**KVD_XDS_SSL_CONFIG.instance_name** 有効な値を以下に示します。 **KVD_XDS_SSL_CONFIG_VERIFY** 検証 **KVD_XDS_SSL_CONFIG_NOVERIFY** 検証なし **KVD_XDS_SSL_CONFIG_NOSSL** SSL なし
PowerShell 認証メカニズム (PowerShell Authentication Mechanism)	PowerShell でリモート・システムから情報を取得するための資格情報の作成に使用する認証タイプを定義します。	**KVD_XDS_POWERSHELL_AUTH_MECH.instance_name** 有効な値を以下に示します。 **KVD_XDS_POWERSHELL_BASIC** 基本 **KVD_XDS_POWERSHELL_CREDSSP** CredSSP **KVD_XDS_POWERSHELL_NTLM** NTLM **KVD_XDS_POWERSHELL_DEFAULT** デフォルト **KVD_XDS_POWERSHELL_DIGEST** ダイジェスト **KVD_XDS_POWERSHELL_KERBEROS** Kerberos **KVD_XDS_POWERSHELL_NEGOTIATE** ネゴシエーション

Windows イベントおよび PowerShell メトリックのモニターの有効化

この手順を使用して、Windows イベントおよび PowerShell メトリックのモニターを有効にします。このデータをモニターすると、モニター対象システムのパフォーマンスに大きな影響を与える場合があります。

始める前に

エージェントの PowerShell 構成パラメーターが設定されていることを確認します。

このタスクについて

エージェントが Windows イベントおよび PowerShell メトリックをモニターできるようにするために、以下の拡張環境変数を 1 つ以上有効にする必要があります。

GET_SESSION_LATENCY
セッションの待ち時間および往復時間を、PowerShell からの接続済み VDA からリモートで取得するかどうかを指定します。

GET_VDA_MACHINE_METRICS_REMOTELY
VDA マシン・メトリックを PowerShell からリモートで取得するかどうかを指定します。

RETRIEVE_WINDOWS_EVENTS
Windows イベント・ログ・イベントを、Window VDA および DDC からの PowerShell から取得するかどうかを指定します。

手順

1. 以下の Citrix VDI エージェントのエージェント・インストール・ディレクトリーに移動します。
   - Linux *install_dir*/config
   - Windows *install_dir*¥TMAITM6_x64

   ここで、*install_dir* はエージェントのインストール先パスです。
2. Citrix VDI エージェント構成ファイルを編集して、*GET_SESSION_LATENCY*、*GET_VDA_MACHINE_METRICS_REMOTELY*、および *RETRIEVE_WINDOWS_EVENTS* の 1 つ以上の変数を true に設定します。
   - Linux vd.environment
   - Windows KVDENV_*instance_name*

   ここで *instance_name* はエージェント・インスタンスの名前です。
3. エージェントを再始動します。

   重要: これらの設定を、すべての新規エージェント・インスタンスに対してデフォルトにするには、構成テンプレート・ファイルでこれらの設定を true に設定します。
   - Linux この設定は、ステップ 2 で vd.environment を編集することで、新規エージェント・インスタンスのデフォルトに既になっています。
   - Windows KVDENV

例

```
GET_SESSION_LATENCY=true
GET_VDA_MACHINE_METRICS_REMOTELY=true
RETRIEVE_WINDOWS_EVENTS=true
```

DataPower モニターの構成

DataPower アプライアンスをモニターするには、まずアプライアンスでいくつかの構成タスクを実行し、次に Monitoring Agent for DataPower を構成する必要があります。

ヒント: APM v8: Configuring DataPower monitoring in IBM APM をクリックして、DataPower モニターの基本構成プロセスに関するビデオを閲覧してください。

DataPower アプライアンスの構成

Monitoring Agent for DataPower を構成する前に、アプライアンスでいくつかの構成タスクを実行する必要があります。

ヒント: サポートされる DataPower アプライアンスについては、Software Product Compatibility Reports の「Prerequisites」タブを参照してください。

3 つの異なるレベルで DataPower アプライアンスをモニターできます。Cloud APM コンソールに DataPower アプライアンス・データを表示するためにモニターする DataPower アプライアンス ごとに、要件に基づいて 3 つのレベルを構成します。

1. リソース・モニター

   リソース使用状況、スループット、接続統計などのモニター・データを表示するには、リソース・モニターを有効にします。説明については、238 ページの『リソース・モニター』を参照してください。

2. ミドルウェア・トランザクション・トラッキング

   トランザクション詳細情報、ボリューム、依存関係など、トランザクションのデータをモニターするには、ミドルウェア・トランザクション・トラッキングを有効にします。説明については、239 ページの『ミドルウェア・トランザクション・トラッキング』を参照してください。

3. DataPower アプライアンスのインスタンス・レベルのトランザクション・トラッキング

   インスタンス・トポロジー内のトランザクションのモニター・データを表示するには、DataPower アプライアンスのインスタンス・レベルのトランザクション・トラッキングを構成します。説明については、240 ページの『DataPower アプライアンスのインスタンス・レベルのトランザクション・トラッキング』を参照してください。

重要 : ユーザー ID が DataPower アプライアンスを構成するための適切な許可を備えていることを確認してください。DataPower アプライアンスの構成に使用するユーザー ID の「**アクセス・プロファイル (Access profile)**」フィールドに`*/*/*?Access=r` を入力できます。 その後、そのユーザー ID を使用して DataPower アプライアンスを構成します。

パブリック証明書のエクスポート
DataPower アプライアンスの XML 管理インターフェースで SSL プロキシー・プロファイルが有効になっている場合は、DataPower アプライアンスの XML 管理インターフェースによって 使用されるパブリック証明書を、DataPower エージェントを実行するマシンにエクスポートしておく必要があります。

手順

1. DataPower アプライアンスの XML 管理インターフェースによって 使用される暗号証明書 (`pubcert:///mycert.pem` など) をダウンロードするには、「**管理**」 > 「**メイン**」 > 「**ファイル管理**」をクリックし、DataPower エージェントを実行するマシンに証明書を保存します。
2. DataPower エージェントを構成するときに、「**SSL プロキシー・プロファイル**」フィールドを指定するオプションがあります。パブリック証明書の絶対パスを入力します。

   注 : マルチプロトコル・ゲートウェイを追加した場合、上記ステップを繰り返す必要があります。

リソース・モニター
DataPower アプライアンスで使用可能なモニターの第 1 レベルは、SOAP 管理、統計、トランザクション速度などのリソース・モニターの有効化です。

以下の構成タスクにおける DataPower Gateway ユーザー・インターフェース (UI) での操作は、DataPower Gateway バージョン 7.5.1 およびそれ以前のバージョンに適用されます。使用している DataPower Gateway のバージョンが V 7.5.1 より新しい場合は、UI の右上にある疑問符 (?) をクリックし、「**WebGUI**」を選択して以前のバージョンの UI に戻すことができます。その後、DataPower アプライアンスの 構成タスクを実行するための指示に従います。

***SOAP* 管理の有効化**
DataPower エージェントが DataPower アプライアンスからデータを収集するようにする場合、SOAP 管理を有効にするように XML 管理インターフェースを 構成する必要があります。

手順

SOAP を使用可能にするには、以下の手順を実行します。

1. モニター対象の DataPower アプライアンスの WebGUI にログオンします。
2. 「**オブジェクト**」 > 「**デバイス管理**」 > 「**XML 管理インターフェース**」をクリックします。

   注 : 管理状態が有効であることを確認してください。
3. 「**ポート番号**」に、DataPower エージェントが通知レポートを listen するポート番号を入力します。デフォルトで、ポート番号は 5550 です。

4.「**使用可能サービス (Enabled Services)**」で「**SOAP 管理 (SOAP Management)**」が選択されていることを確認します。

統計の有効化
DataPower エージェントで DataPower アプライアンスからデータを収集する必要がある場合は、統計を有効にしなければなりません。

手順

統計を有効にするには、以下の手順に従います。

1. モニター対象の DataPower アプライアンスの WebGUI にログオンします。
2.「**管理**」 >「**デバイス**」 >「**統計設定**」をクリックします。
3.「**統計設定**」を有効にし、「**適用**」をクリックします。

トランザクション率の有効化
DataPower エージェントで DataPower アプライアンスからデータを収集するには、トランザクション率が有効になっている必要があります。

手順

トランザクション率を有効にするには、以下の手順を実行します。

1. モニター対象の DataPower アプライアンスの WebGUI にログオンします。
2. `default` ドメインを選択してください。
3.「**状況**」 >「**接続**」 >「**トランザクション率**」をクリックします。
4.「**統計は、現在無効です**」と表示された場合、「**無効**」をクリックし、「統計設定」で「**管理状態**」を「**有効**」に設定します。
5. 複数のドメインがある場合は、「**すべてのドメインの表示**」をクリックし、ステップ 3 から 4 を繰り返して、該当するすべてのドメインのトランザクション率を有効にします。
6.「**適用**」をクリックします。

ミドルウェア・トランザクション・トラッキング
DataPower アプライアンスで使用可能なモニターの第 2 レベルは、ワークスペースでのミドルウェア・トランザクション・トラッキングの表示です。

以下の構成タスクにおける DataPower Gateway ユーザー・インターフェース (UI) での操作は、DataPower Gateway バージョン 7.5.1 およびそれ以前のバージョンに適用されます。使用している DataPower Gateway のバージョンが V 7.5.1 より新しい場合は、UI の右上にある疑問符 (?) をクリックし、「**WebGUI**」を選択して以前のバージョンの UI に戻すことができます。その後、DataPower アプライアンスの 構成タスクを実行するための指示に従います。

DataPower アプライアンスを 介した SOAP トラフィックと REST トラフィックの両方のトランザクション・トラッキングがサポートされます。DataPower のトランザクション・トラッキングでは、ファイル `store:///soapreq.xsl`、`store:///soaprsp.xsl`、および `store:///soaperror.xsl` を使用して SOAP がサポートされます。これらの XSL ファイルは、SOAP エンベロープ内に `kd4:KD4SoapHeaderV2` を追加し、これについてレポートするように Web サービス・プロキシーを装備します。

`soap*.xsl` ファイルに加え、DataPower トランザクション・トラッキングには、`apm_req.xsl`、`apm_rsp.xsl`、および `apm_error.xsl` も含まれています。これらのファイルでは、`ARM_CORRELATOR: HTTP Header` が含まれている着信 HTTP 要求と、`ITCAMCorrelator` または `kd4:KD4SoapHeaderV2` が含まれている SOAP エンベロープがサポートされます。Web サービス・プロキシーは、`ARM_CORRELATOR: HTTP Header` が含まれるように発信要求を更新または設定し、SOAP 相関関係子を削除します。

注 : DataPower アプライアンスをビジネス・アプリケーションに追加し、そのアプリケーションが複数のアプリケーションのトラフィックを 伝送する場合、トランザクション・トラッキングを有効にすると、それらのビジネス・アプリケーションについて表示されるアプリケーション・トポロジーには、すべてのアプリケーションのノードへのパスが 含められます。

Web サービス管理の構成

トラッキング・データを表示する DataPower アプライアンスごとに、以下の手順を実行します。

1. モニター対象の DataPower アプライアンスの WebGUI にログオンします。
2. default ドメインを選択してください。
3.「XML 管理インターフェース」を検索します。以下の値を設定し、「**適用**」をクリックします。
   - 「**メイン**」タブの「**有効なサービス**」セクションで、「**WS 管理エンドポイント**」を有効にします。
4.「Web サービス管理エージェント」を検索します。以下の値を設定し、「**適用**」をクリックします。
   - 「**管理状態**」を「**有効**」に設定します。
   - 「**キャプチャー・モード**」を「**なし**」に設定します。
   - 「**バッファリング・モード (非推奨) (Buffering Mode (deprecated))**」を「**廃棄**」に設定します。
5. 次の各トピックの説明どおりに、Web サービス・プロキシーまたはマルチプロトコル・ゲートウェイを構成します。

Web サービス・プロキシーの構成

トラッキング・データを表示する Web サービス・プロキシーごとに、以下の手順を実行します。

手順

1. Web サービス・プロキシーが属しているドメインを選択します。
2. 「**プロキシー設定**」タブで、次の値を設定し、「**適用**」をクリックします。
   - 「**Web サービス管理エージェントを経由したモニター**」を「**オン**」に設定します。
3. Cloud APM コンソール で、SOAP 障害を報告し、エラー処理を無効にして、エラー報告を有効にするには、「**詳細なプロキシー設定**」タブで、「**HTTP エラーの処理**」を「オフ」に設定し、「**適用**」をクリックします。

マルチプロトコル・ゲートウェイの構成

トランザクション・トラッキング・データを表示するマルチプロトコル・ゲートウェイごとに、以下の手順を実行します。

手順

1. マルチプロトコル・ゲートウェイが属しているドメインを選択します。
2. マルチプロトコル・ゲートウェイの「**拡張**」タブで、次の値を設定し、「**適用**」をクリックします。
   - 「**Web サービス管理エージェントを経由したモニター**」を「**オン**」に設定します。
   - Web サーバーがリダイレクトを使用する場合は、「**リダイレクトに従う**」を「**オフ**」に設定します。次に、「**ロケーション URL の再書き込み**」を「**オン**」に設定します。
3. 応答タイプまたは要求タイプが 非 XML のマルチプロトコル・ゲートウェイをモニターする場合は、「クライアントからサーバー」と「サーバーからクライアント」の両方向をカバーするルールを使用してマルチプロトコル・ゲートウェイ・ポリシーを定義する必要があります。非 XML マルチプロトコル・ゲートウェイのポリシーにルールがない場合、Web サービス管理エージェントも DataPower の「プローブのデバッグ」(有効になっている場合) もトラフィックを収集しません。
4. バックエンド・サーバーから HTTP 応答コードを伝搬し、SOAP 障害を報告するには、「**拡張設定**」タブで、「**バックエンド・エラーの処理**」を「オフ」に設定し、「**適用**」をクリックします。

DataPower アプライアンスのインスタンス・レベルのトランザクション・トラッキング

DataPower アプライアンスで使用可能なモニターの第 3 レベルは、インスタンス・トポロジーでのデータの表示です。

以下の構成タスクにおける DataPower Gateway ユーザー・インターフェース (UI) での操作は、DataPower Gateway バージョン 7.5.1 およびそれ以前のバージョンに適用されます。使用している DataPower Gateway のバージョンが V 7.5.1 より新しい場合は、UI の右上にある疑問符 (?) をクリックし、「**WebGUI**」を選択して以前のバージョンの UI に戻すことができます。その後、DataPower アプライアンスの 構成タスクを実行するための指示に従います。

変換の構成
インスタンス・トポロジーで表示する各 DataPower アプライアンスで以下の作業を完了します。

このタスクについて

IBM Performance Management V8.1.2 Fix Pack 1 では、DataPower アプライアンスを使用した SOAP トラフィックのトランザクション・トラッキングがサポートされます。DataPower のトランザクション・トラッキングでは、ファイル `store:///soapreq.xsl`、`store:///soaprsp.xsl`、および `store:///soaperror.xsl` を使用して SOAP がサポートされます。これらの XSL ファイルは、SOAP エンベロープ内に `kd4:KD4SoapHeaderV2` を追加し、これについてレポートするように Web サービス・プロキシーを装備します。

IBM Performance Management V8.1.3 以降では、DataPower アプライアンスを使用した REST トラフィックのトランザクション・トラッキングもサポートされます。 `soap*.xsl` ファイルに加え、DataPower トランザクション・トラッキングには、`apm_req.xsl`、`apm_rsp.xsl`、および `apm_error.xsl` も含まれています。これらのファイルでは、`ARM_CORRELATOR: HTTP Header` が含まれている着信 HTTP 要求と、`ITCAMCorrelator` または `kd4:KD4SoapHeaderV2` が含まれている SOAP エンベロープがサポートされます。 Web サービス・プロキシーは、`ARM_CORRELATOR: HTTP Header` が含まれるように発信要求を更新または設定し、SOAP 相関関係子を削除します。

DataPower エージェントは、DataPower アプライアンスを介した SOAP トラフィック、DataPower アプライアンスを介した REST トラフィック、および DataPower と WebSphere MQ と間のトラフィックのトランザクション・トラッキングをサポートします。

- DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックのトランザクション・トラッキングを有効にするには、`ARM_CORRELATOR: HTTP Header` を含む着信 HTTP 要求をサポートする `apm_req.xsl`、`apm_rsp.xsl`、および `apm_error.xsl` を適用するか、`ITCAMCorrelator` または `kd4:KD4SoapHeaderV2` が含まれている SOAP エンベロープを適用します。Web サービス・プロキシーは、`ARM_CORRELATOR: HTTP Header` が含まれるように発信要求を更新または設定し、SOAP 相関関係子を削除します。
- DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトランザクション・トラッキングを有効にするには、`apm_req_MQ.xsl`、`apm_rsp_MQ.xsl`、および `apm_error_MQ.xsl` の各ファイルを適用します。これらのファイルを適用すると、SOAP トラフィックと REST トラフィックのトランザクション・トラッキングも自動的に有効になります。

手順

REST トラフィックをトラッキングして、DataPower と WebSphere MQ の間のトランザクション・トラッキングを有効にするには、以下のステップを実行します。

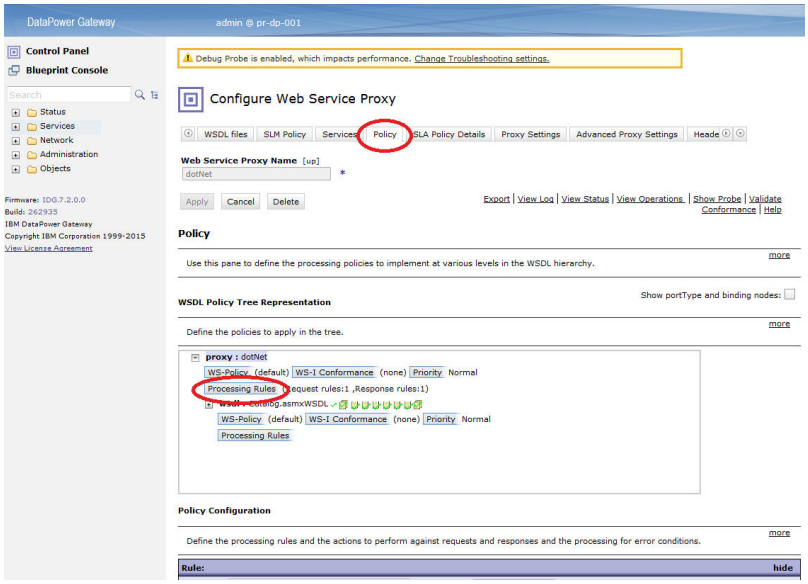
1. 以下の場所からファイルをダウンロードします。
   - Linux システムの場合: `/opt/ibm/apm/agent/`*`lx8266`*`/bn/bin`
   - AIX システムの場合: `/opt/ibm/apm/agent/`*`aix536`*`/bn/bin`
2. IBM 統合スタックの一部としてモニターする各 DataPower アプライアンスに XSL ファイルをアップロードします。
3. 次の各トピックの説明どおりに、Web サービス・プロキシーまたはマルチプロトコル・ゲートウェイを構成します。
4. モニターするドメインごとに、以下のステップを使用して構成します。
   a) DataPower Gateway ヘッダーのドロップダウン・リストでドメインを選択します。
   b)「コントロール・パネル」ナビゲーターで「**オブジェクト**」>「**デバイス管理**」>「**Web サービス管理エージェント**」を選択します。
   c)「**バッファリング・モード (非推奨) (Buffering Mode (deprecated))**」を「廃棄」に設定します。
   d)「**適用**」をクリックします。

Web サービス・プロキシーの構成
インスタンス・トポロジーで表示する各 Web サービス・プロキシーで以下の作業を完了します。

手順

WebGUI で、モニターする Web サービス・プロキシーごとに以下の手順を実行します。

1. 「**Web サービス・プロキシーの構成 (Configure Web Service Proxy)**」ページで、構成する Web サービス・プロキシーの名前を選択します。
2. 「**ポリシー**」タブで「**プロキシー: *domain* (proxy :*domain*)**」を展開し、「**処理ルール**」をクリックします。

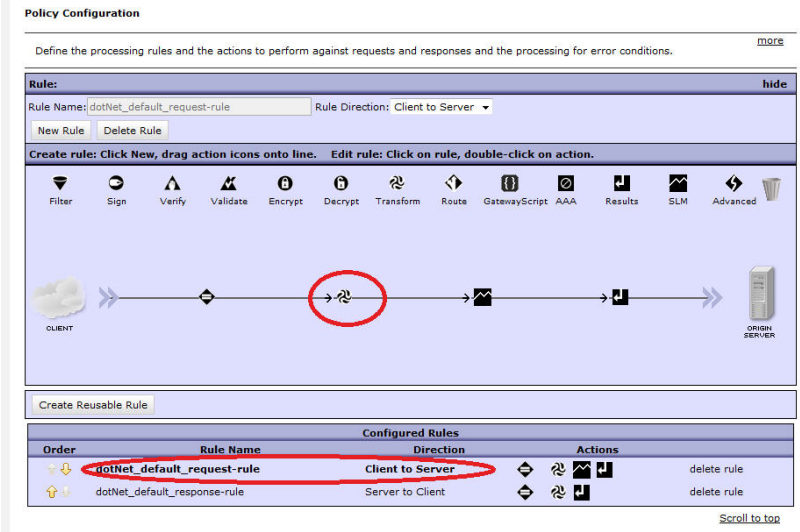


3. 「**ポリシー構成**」セクションで、既存の「クライアントからサーバー」ルールを選択するか、「**新規ルール**」をクリックしてルールを作成します。
   a. 変換をタイムラインにドラッグします。

      注:

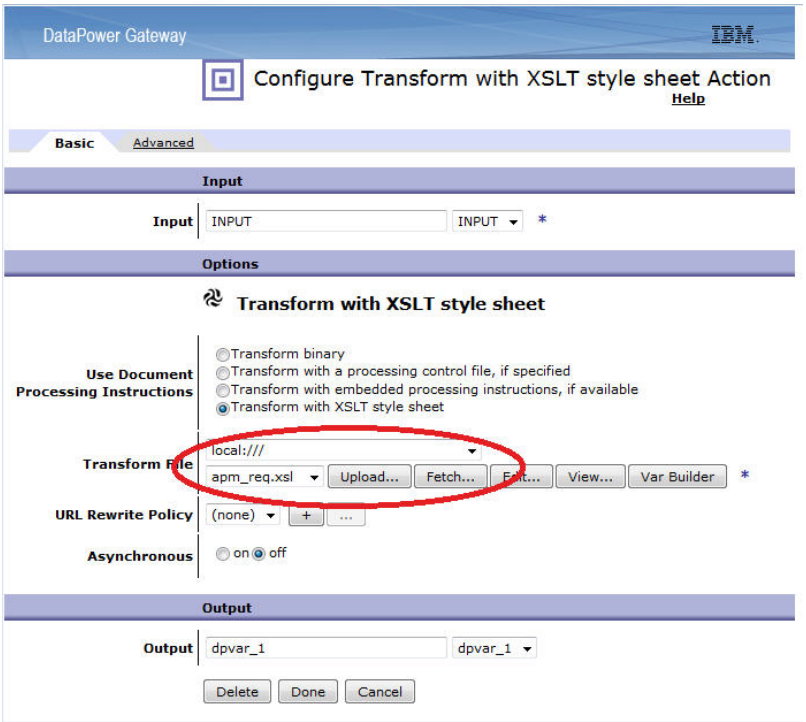
      1) 「クライアントからサーバー」ルールが既に存在する場合は、それに変換ノードを追加します。
      2) 「クライアントからサーバー」ルールに、認証、許可、および監査 (AAA) ノードがある場合、DataPower エージェントの XSLT ファイルを格納する変換ノードを、AAA ノードより前に配置します。

   b. 変換をダブルクリックして編集します。



c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの `apm_req.xsl` を選択します。 例: `local:///`

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。



ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「クライアントからサーバー」ルールを構成する場合は、このステップで、`apm_req.xsl` ファイルの代わりに `apm_req_MQ.xsl` ファイルを適用します。

d.「**完了**」をクリックします。

4.「**ポリシー構成**」セクションに戻って、ステップ 3 を繰り返し、「サーバーからクライアント」ルールを構成するか、「**新規ルール**」をクリックしてルールを作成します。

a. 変換をタイムラインにドラッグします。

b. 変換をダブルクリックして編集します。

c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの `apm_rsp.xsl` を選択します。 例: `local:///`

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。

ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「サーバーからクライアント」ルールを構成する場合は、このステップで、`apm_rsp.xsl` ファイルの代わりに、`apm_rsp_MQ.xsl` ファイルを適用します。

d.「**完了**」をクリックします。

5.「**ポリシー構成**」セクションに戻って、ステップ 3 を繰り返し、「エラー」ルールを構成するか、「**新規ルール**」をクリックしてルールを作成します。

a. 変換をタイムラインにドラッグします。

b. 変換ルールをダブルクリックして編集します。

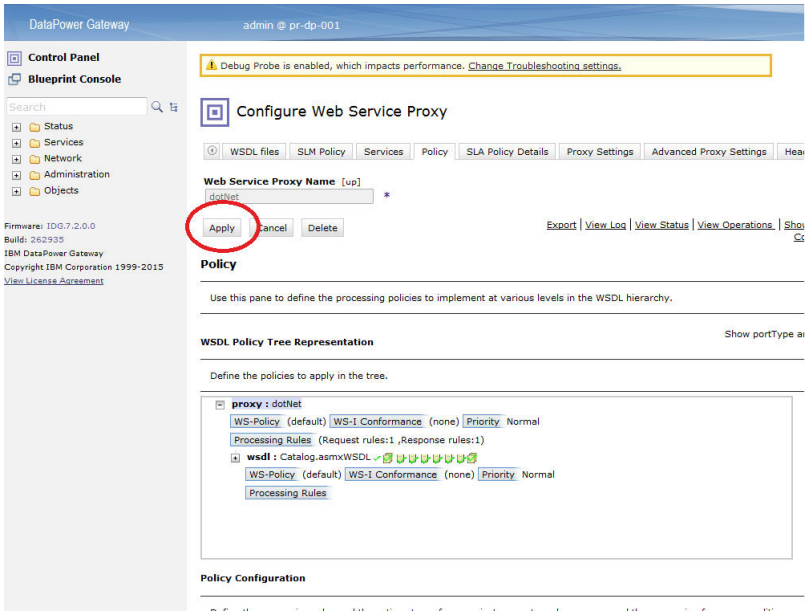
c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの `apm_error.xsl` を選択します。 例: `local:///`

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。

ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「エラー」ルールを構成する場合は、このステップで、`apm_error.xsl` ファイルの代わりに `apm_error_mq.xsl` ファイルを適用します。

d.「**完了**」をクリックします。

6.「**Web サービス・プロキシーの構成 (Configure Web Service Proxy)**」ページに戻って、「**適用**」をクリックします。



マルチプロトコル・ゲートウェイの構成
インスタンス・トポロジーで表示する各マルチプロトコル・ゲートウェイで以下の作業を完了します。

手順

WebGUI で、モニターするマルチプロトコル・ゲートウェイごとに以下の手順を実行します。

1.「**マルチプロトコル・ゲートウェイの構成**」ページで、構成するマルチプロトコル・ゲートウェイの名前をクリックします。
2.「**マルチプロトコル・ゲートウェイ・ポリシー**」ページで、ポリシーを構成します。**...**をクリックします。
3.「**マルチプロトコル・ゲートウェイ・スタイル・ポリシーの構成**」ページで、既存の「クライアントからサーバー」ルールを選択するか、「**新規ルール**」をクリックしてルールを作成します。

a. 変換をタイムラインにドラッグします。

注:

1)「クライアントからサーバー」ルールが既に存在する場合は、それに変換ノードを追加します。
2)「クライアントからサーバー」ルールに、認証、許可、および監査 (AAA) ノードがある場合、DataPower エージェントの XSLT ファイルを格納する変換ノードを、AAA ノードより前に配置します。

b. 変換ルールをダブルクリックして編集します。

c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの apm_req.xsl を選択します。 例: local:///

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。

ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「クライアントからサーバー」ルールを構成する場合は、このステップで、apm_req.xsl ファイルの代わりに apm_req_MQ.xsl ファイルを適用します。

d.「**完了**」をクリックします。

4.「**マルチプロトコル・ゲートウェイ・スタイル・ポリシーの構成**」ページに戻って、既存の「サーバーからクライアント」ルールを選択するか、「**新規ルール**」をクリックしてルールを作成します。

a. 変換をタイムラインにドラッグします。

b. 変換ルールをダブルクリックして編集します。

c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの apm_rsp.xsl を選択します。 例: local:///

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。

ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「サーバーからクライアント」ルールを構成する場合は、このステップで、apm_rsp.xsl ファイルの代わりに、apm_rsp_MQ.xsl ファイルを適用します。

d.「**完了**」をクリックします。

5.「**マルチプロトコル・ゲートウェイ・スタイル・ポリシーの構成**」ページに戻って、既存の「エラー」ルールを選択するか、「**新規ルール**」をクリックしてルールを作成します。

a. 変換をタイムラインにドラッグします。

b. 変換ルールをダブルクリックして編集します。

c.「**XSLT スタイル・シートを使用して変換するアクションの構成 (Configure Transform with XSLT style sheet Action)**」ウィンドウの「変換ファイル」の横で、アップロードしたデータ・ストアの apm_error.xsl を選択します。 例: local:///

ファイルが存在しない場合は、「**アップロード**」をクリックして、インストールした場所から取得します。

ヒント: DataPower アプライアンスを介した SOAP トラフィックと REST トラフィックに加えて、DataPower と WebSphere MQ の間のトラフィックをモニターする「エラー」ルールを構成する場合は、このステップで、apm_error.xsl ファイルの代わりに apm_error_mq.xsl ファイルを適用します。

d.「**完了**」をクリックします。

6.「**マルチプロトコル・ゲートウェイ・スタイル・ポリシーの構成**」ページに戻って、「**拡張**」タブで「**Web サービス管理エージェントを経由したモニター**」を「**オン**」に設定し、「**適用**」をクリックします。

7.「**適用**」をクリックします。

次のタスク

場合によっては、トランザクション・トラッキングに変換を追加すると、DataPower によって HTTP の Content-Type ヘッダーの値が変更される場合があります。イメージがロードされない Web ページや、HTML テキストが文字化けしてレンダリングされるバイナリー・ファイルが表示される場合があります。

DataPower の動作は、XSL 変換がないルールと、1 つ以上の XSL 変換を持つルールとを比較するときに変更されます。サービスが MIME、MTOM、XOP などでエンコードされたメッセージを処理する場合、この動

作が適切なことがありますが、それ以外の場合には、この動作を防止するように DataPower 構成を変更してください。

DataPower が HTTP の Content-Type ヘッダーを変更するのを防止するには、次のように、影響がある各ルールに var://service/mpgw/proxy-content-type 変数を設定します。

1. ルールに拡張オブジェクトをドラッグします。
2. 拡張オブジェクトをダブルクリックして 編集します。
3. 「**変数の設定**」を選択し、「**次へ**」をクリックします。
4. 変数名 `service/mpgw/proxy-content-type` と変数値 1 を入力し、「**完了**」をクリックします。
5. ポリシーとサービス構成の変更を適用します。
6. 影響がある各ルールについて、1 から 5 の手順を繰り返します。

DataPower エージェントの構成

Monitoring Agent for DataPower には、エンタープライズ環境内の DataPower アプライアンスをモニターする中央ポイントが用意されています。アプライアンスに共通の問題に関する通知を識別および受信できます。エージェントは、アプライアンスのパフォーマンス、リソース、およびワークロードに関する情報も提供します。

このタスクについて

DataPower エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で始動する必要があります。管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は 32 文字に制限されています。

指定するインスタンス名は 28 文字 (ホスト名の字数は除く) までに制限されています。例えば、インスタンス名に `DataPower` を指定すると、管理対象システム名は `DataPower:hostname:BN` になります。

重要 : 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが正しく表示されません。

注 : DataPower エージェントの XSLT では、メインフレーム・アプリケーションに使用されている BLOB 文字が解析されません。

各実動 DataPower アプライアンスに対して、1 つのインスタンスを構成します。DataPower アプライアンスが非実稼働アプライアンスまたは小規模アプライアンスの場合、すべてのアプライアンスをモニターするように単一のエージェント・インスタンスのみを構成できます。複数のインスタンスが同じマシン上で実行できます。インスタンスを作成し、構成設定を変更する場合は、構成スクリプトを実行します。必要のないプロンプト表示と応答を迂回するには、スクリプトを実行する前に、エージェントのサイレント応答ファイルを編集できます。

手順

- DataPower エージェントを構成するためには、以下のいずれかの手順を行います。
  - Linux AIX プロンプトに応答を入力してエージェントを構成する場合は、以下のステップを実行します。
    1. *install_dir*/bin ディレクトリーに移動します。ここで、*install_dir* は DataPower エージェントのインストール・ディレクトリーです。
    2. `./datapower-agent.sh config` *instance_name* コマンドを実行します。

       サーバーに固有の *instance_name* を選択します。
    3. DataPower エージェント設定を編集するプロンプトが出されたら、1 を入力して続行します。
    4. **管理対象システムの詳細**を編集するプロンプトが出されたら、以下のオプションのいずれかを入力します。

       - 1=追加

- 2=編集
- 3=削除
- 4=次へ
- 5=終了

ご使用のシステム上の DataPower エージェント・インスタンスを初めて構成する場合は、「使用可能な DataPower アプライアンス設定がありません」というメッセージが表示されます。1 を入力して、DataPower アプライアンス 設定を追加します。デフォルトのオプションは 5=終了です。

5. DataPower アプライアンスのプロパティーを 以下のように入力します。

   管理対象システム名
   「管理対象システム名」には、エージェントの管理対象システム 名を入力します。

   エージェントのすべてのインスタンスで固有であり、アプライアンスを容易に識別するために使用できる**「管理対象システム名」**を選択してください。名前には、英数字のみを使用できます。例えば、DataPower アプライアンスのホスト名です。

   デバイス・ホスト
   「デバイス・ホスト」には、モニター対象の DataPower アプライアンスの IP アドレスを入力します。デフォルトの IP アドレスは *9.123.109.139* です。

   XML 管理インターフェースのポート
   「XML 管理インターフェースのポート」に、XML 管理インターフェースのポート 番号を入力します。デフォルトの番号は、5550 です。

   ユーザー ID
   「ユーザー ID」に、モニター対象の DataPower アプライアンスにログインするためのユーザー ID を入力します。 デフォルト値は admin です。

   パスワード
   「パスワード」に、モニター対象の DataPower アプライアンスにログインするためのパスワードを入力してから、パスワードを確認します。

   SSL プロキシー・プロファイル
   デバイスの XML 管理インターフェースがプロファイルを 使用するように構成されている場合は、**「SSL プロキシー・プロファイル」**に、SSL プロキシー・プロファイルのパブリック証明書の絶対パスを入力します。例えば、次の場合があります。

   ```
   the location of the .pem file exported from datapower appliances/mycert.pem
   ```

   ここで、*the location of the .pem file exported from datapower appliances* は、公開証明書の絶対パスです。公開証明書をエクスポートするには、『公開証明書のエクスポート』を参照してください。

   SSL プロキシー・オプション (SSL Proxy Option)
   モニター対象デバイスの XML 管理インターフェースがカスタムの SSL プロキシー・プロファイルを使用するように構成されている場合は、**「SSL プロキシー・オプション（SSL Proxy Option)」**を「はい」に設定します。 そうではない場合は、「いいえ」に設定してください。

6. 複数の DataPower アプライアンスをモニターするには、246 ページの『4』 と 247 ページの『5』を繰り返して、DataPower アプライアンスごとに 1 つのエージェント・インスタンスを構成します。その他の場合は、5 を入力して **Enter** キーを押し、構成を完了します。
7. 以下のコマンドを実行して、エージェントを開始します。

   ```
   ./datapower-agent.sh start instance_name
   ```

- サイレント構成

1. サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。

- Linux AIX テキスト・エディターで *install_dir*/samples/datapower_silent_config.txt を開きます。
- Windows テキスト・エディターで *install_dir*/samples/datapower_silent_config.txt を開きます。

2. DataPower エージェントがアプライアンスをモニターするように 構成するために、以下のプロパティーを入力します。

   デバイス・ホスト
   デバイスのホスト名または IP アドレスを入力します。例えば、**SOAP_HOST.ManageSystemName=** *datapower01* です。

   XML 管理インターフェースのポート
   XML 管理インターフェースのポート 番号を入力します。デフォルト値は 5550 です。例えば、**DP_PORT.ManageSystemName=** *5550* です。

   ユーザー ID
   デバイスへの接続に使用するユーザー ID を入力します。デフォルト値は admin です。例えば、**DP_UID.ManageSystemName=** *admin* です。

   パスワード
   ユーザー ID のパスワードを入力します。例えば、**DP_PASSWORD.ManageSystemName=** *password* です。

   SSL プロキシー・プロファイル
   デバイスの XML 管理インターフェースがプロファイルを 使用するように構成されている場合は、SSL プロキシー・プロファイルのパブリック 証明書の絶対パスを入力します。例えば、次の場合があります。

   ```
   the location of the .pem file exported from datapower appliances/mycert.pem
   ```

   ここで、*the location of the .pem file exported from datapower appliances* は、公開証明書の絶対パスです。公開証明書をエクスポートするには、『公開証明書のエクスポート』を参照してください。

   SSL プロキシー・オプション (SSL Proxy Option)
   モニター対象デバイスの XML 管理インターフェースがカスタムの SSL プロキシー・プロファイルを使用するように構成されている場合は、**「SSL プロキシー・オプション（SSL Proxy Option)」**を「はい」に設定します。 それ以外の場合は「いいえ」に設定します。例: **DP_SSL_OPTION.ManageSystemName1=** はい。

   重要 : ManageSystemName は固有です。必ずすべての項目で、ご使用のシステム名に置き換えてください。複数のアプライアンスをモニターする 場合は、 アプライアンスをモニターするために示したステップを繰り返します。必ず適切な ManageSystemName および DataPower アプライアンスのパラメーターを設定してください。

3. エージェントのインストール・ディレクトリーに移動し、以下のコマンドを実行してエージェントを開始します。

   ```
   ./datapower-agent.sh start instance_name
   ```

次のタスク

- 構成済みのエージェント・インスタンスの名前および設定を確認するために、コマンド **./cinfo -s bn** を実行します。
- DataPower エージェントのデータが Cloud APM コンソールで表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

- Cloud APM コンソールでトランザクション・トラッキング・データを表示するには、DataPower エージェントのトランザクション・トラッキングを構成します。説明については、『DataPower エージェントのトランザクション・トラッキングの構成』を参照してください。
- 異なるレベルでモニターを表示するには、DataPower アプライアンスを適宜構成します。手順については、DataPower アプライアンスのリソース・モニター、ミドルウェア・トランザクション・トラッキング、およびインスタンス・レベルのトランザクション・トラッキングを参照してください。

DataPower エージェントのトランザクション・トラッキングの構成

ミドルウェアおよびトポロジー・ダッシュボードで DataPower アプライアンスのトランザクション・トラッキング・データを表示するには、DataPower エージェントのトランザクション・トラッキングを有効にする必要があります。

始める前に

- DataPower エージェントをインストールし、DataPower アプライアンス に接続するように構成します。
- DataPower アプライアンスで SOAP または ARM のモニターを有効にします。

手順

DataPower エージェント用のトランザクション・トラッキングを有効にするには、以下の手順を実行します。

1. ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。
2. 「**DataPower**」タブで、トランザクション・トラッキングを有効にするエージェント・インスタンスを選択します。
3. 「**アクション**」 > 「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」を選択し、トランザクション・トラッキングを有効にします。エージェントの「**トランザクション・トラッキング**」 列の状況が「**有効**」に更新されます。

タスクの結果

選択されたエージェント・インスタンスのトランザクション・トラッキングを有効にしました。

次のタスク

ミドルウェアおよびトポロジー・ダッシュボードで DataPower アプライアンス のデータを表示するには、次に、モニターするアプライアンスを Application Performance ダッシュボードに追加する必要があります。Application Performance ダッシュボードへの DataPower アプライアンスの追加について詳しくは、100 ページの『Application Performance ダッシュボードへの ミドルウェア・アプリケーションの追加』を参照してください。

注 : 統合サービスを使用していて、IBM Integration Bus と DataPower 間で送信されるデータをモニターする場合は、正確な集約トランザクション・トポロジーを表示するために追加の構成が必要になります。IBM Integration Bus エージェントには、SOAP エンベロープがない SOAP メッセージの相関サポートを組み込むことはできません。SOAPRequest ノード、SOAPAsyncRequest ノードおよび SOAPReply ノードは、SOAP エンベロープを使用しないメッセージを入力メッセージとして受け入れることができます。これらのノードについては、メディエーションからダウンストリームのメディエーションまたはアプリケーション・サーバーへの関係がトポロジー・ビューに表示されません。この問題を回避するには、IBM Integration Bus メッセージ・フロー内で SOAPRequest、SOAPAsyncRequest、または SOAPReply ノードの直前に SOAPEnvelope ノードを挿入し、SOAPEnvelope ノードの「**新規エンベロープの作成**」オプションを選択して、SOAP メッセージの SOAP エンベロープを追加します。

Db2 モニターの構成

Monitoring Agent for Db2 は、Db2 サーバー の可用性およびパフォーマンスをモニターします。Cloud APM コンソールで、複数のサーバーをモニターすることができます。各サーバーは、Db2 エージェント・インスタンスによってモニターされます。リモート・モニターは Db2 エージェントでもサポートされています。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Db2 エージェント を参照してください。

このタスクについて

Db2 エージェントは、複数インスタンス・エージェントです。最初にインスタンスを作成し、エージェントを手動で開始する必要があります。

管理対象システム名には、指定するエージェント・インスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。

各部の意味は次のとおりです。

- *pc* は、2 文字の製品コードです。
- *instance_name* は、エージェント・インスタンス名です。これはモニター対象の Db2 インスタンス名と同じでなければなりません。

管理対象システム名は最大で 32 文字です。 指定するエージェント・インスタンス名は 8 文字 (ホスト名の字数は除く) まで入力できます。例えば、エージェント・インスタンス名に `DB2inst1` を指定すると、管理対象システム名は `DB2inst1:hostname:ud` になります。

重要 : 長いエージェント・インスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが完全には表示されません。

エージェントの構成時に権限の問題が発生しないようにするために、必ず、エージェントのインストールに使用したのと同じ root ユーザー ID または非 root ユーザー ID を使用してください。選択されたユーザーとしてエージェントをインストールした後で、別のユーザーとしてエージェントを構成する場合は、187 ページの『非 root ユーザーとしてのエージェントの構成』を参照してください。選択されたユーザーとしてエージェントをインストールおよび構成した後で、別のユーザーとしてエージェントを開始する場合は、1013 ページの『非 root ユーザーとしてのエージェントの開始』を参照してください。

インスタンスを作成して構成設定を変更するには、構成スクリプトを実行します。構成スクリプトを実行する前に、Db2 サイレント応答ファイルを編集して、編集しないと必要になるプロンプトおよび応答を迂回できます。

Db2 エージェントを構成したら、モニター対象のインスタンスの Db2 SYSADM 権限を持つユーザー ID を使用してエージェントを開始してください。 エージェントですべてのモニター・スイッチをオンにしてモニター・データを収集するには、SYSADM 権限が必要です。そのため、SYSADM 権限を持つユーザーがエージェントを開始する必要があります。インスタンス所有者ユーザー (SYSADM 権限を持つ) を使用してエージェントを開始してください。

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

手順

デフォルト設定を使用してエージェントを構成するには、以下の手順を実行します。

1. 以下のコマンドを実行します。*instance_name* は、インスタンスに付ける名前です。

   ```
   install_dir/bin/db2-agent.sh config instance_name
    install_dir/samples/db2_silent_config.txt
   ```

   エージェント・インスタンス名 *instance_name* は、常にモニター対象の Db2 インスタンス名と同じです。既存のエージェント・インスタンスについて詳しくは、186 ページの『「エージェント構成」ページ』を参照してください。
2. 次のコマンドを実行して、Db2 エージェントを始動します。

   ```
   install_dir/bin/db2-agent.sh start instance_name
   ```

次のタスク

- Db2 の一部の属性のデータを表示するために、Db2 ユーザーに特権を付与します。これらの特権の付与については、255 ページの『Db2 メトリックを表示するための特権の付与』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

始める前に

ローカル・モニターおよびリモート・モニター用の Db2 エージェントの構成を開始する前に、リモート・モニターに関する以下のタスクが完了していることを確認してください。

- リモート・モニター用にクライアント/サーバー環境をセットアップします。259 ページの『リモート・モニターの前提条件』を参照してください。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Cloud Application Performance Management**」をクリックします。
2. 「IBM Cloud Application Performance Management」ウィンドウで、「**Monitoring Agent for DB2**」を右クリックしてから「**エージェントを構成します**」をクリックします。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。

   重要 : ローカル・モニターの場合、エージェント・インスタンス名は、モニター対象の Db2 インスタンスの名前と一致している必要があります。

   リモート・モニターの場合、エージェント・インスタンス名は固有のカタログ・ノード名でなければなりません。
4. 「**Monitoring Agent for DB2**」ウィンドウで、次の手順を実行します。
   a) 「**ユーザー名**」に、Db2 インスタンスのユーザー名を入力します。

      ローカル Db2 の場合、Db2 インスタンス所有者の名前を入力します。

      リモート Db2 の場合、リモート Db2 マシンから、実際の Db2 インスタンス所有者の名前を入力します。

      重要 : リモート Db2 インスタンスをモニターする場合、このパラメーターは必須です。

   b) 「**パスワード**」に、Db2 インスタンスのパスワードを入力します。

      ローカル Db2 の場合、Db2 インスタンス所有者のパスワードを入力します。

      リモート Db2 の場合、リモート Db2 マシンから、実際の Db2 インスタンス所有者のパスワードを入力します。

      重要 : リモート Db2 インスタンスをモニターする場合、このパラメーターは必須です。

   c) 「**DB2 カスタマイズ SQL 定義ファイル**」フィールドに、SQL 定義ファイルの絶対ファイル・パス名を入力します。SQL 定義ファイルがデフォルト・ディレクトリー内にある場合、このフィールドをブランクのままにします。その他の場合は、ファイルの絶対ファイル・パス名を入力します。デフォルトのファイル名とそのパスは以下のとおりです。

      Linux AIX `CANDLEHOME/config/kudcussql.properties`

      Windows `CANDLEHOME¥TMAITM6_x64¥kudcussql.properties`

   d) 「**db2diag ログ・ファイル・パス**」フィールドに、db2diag ログ・ファイルのディレクトリー・パスを入力します。db2diag ログ・ファイルがデフォルト・ディレクトリー内にある場合、このフィール

ドをブランクのままにします。 それ以外の場合は、ディレクトリーのパスを入力します。デフォルトのディレクトリー・パスは以下のとおりです。

Linux AIX /home/*DB2owner_home_dir*/sqllib/db2dump

Windows C:¥ProgramData¥IBM¥DB2¥DB2COPY¥*DB2INSTANCENAME*

注 : このパラメーターはリモート・モニターでは適用されません。

e)「**正規表現での MSGID フィルター**」フィールドに、診断ログをフィルターに掛けるための *MSGID* を入力します。MSGID は、メッセージ・タイプ、メッセージ番号、および重大度レベルを組み合わせたものです。正規表現を使用して、メッセージ・タイプ、メッセージ番号、または重大度レベルに基づいてログをフィルタリングします (例: ADM1¥d*1E|ADM222¥d2W)。

f)「**リモート・ホストのパーティションのモニタリングを有効化**」リストで、「はい」を選択して、Db2 エージェントがリモート・ホストのパーティションをモニターできることを指定します。

g)「**すべてのデータベースのモニタリングを有効化**」リストで、「はい」を選択して、Db2 エージェントがすべてのデータベースをモニターできることを指定します。

h)「**OK**」をクリックします。

「IBM Cloud Application Performance Management」ウィンドウに、エージェント・インスタンスが表示されます。

5. リモート・モニターを構成するには、以下のステップを実行します。

a) *install_dir*¥TMAITM6_x64¥KUDENV_<instanceName> を開きます。

b) *KUD_DB2_CLIENT_INST* を、リモート Db2 サーバー・インスタンスがカタログされる Db2 クライアント・インスタンス名に設定します。

6.「**Monitoring Agent for DB2**」インスタンスを右クリックし、「**開始**」をクリックします。

次のタスク

- Db2 エージェントの一部の属性のデータを表示するために、Db2 ユーザーに特権を付与します。これらの特権の付与については、255 ページの『Db2 メトリックを表示するための特権の付与』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Linux または UNIX システム上でのエージェントの構成

構成スクリプトを実行して、Linux システム上でエージェントを構成します。

始める前に

ローカル・モニターおよびリモート・モニター用の Db2 エージェントの構成を開始する前に、リモート・モニターに関する以下のタスクが完了していることを確認してください。

- リモート・モニター用にクライアント/サーバー環境をセットアップします。259 ページの『リモート・モニターの前提条件』を参照してください。

手順

1. コマンド *install_dir*/bin/db2-agent.sh config *instance_name* を実行します。

ここで、*instance_name* は、インスタンスに指定する名前です。

重要 : ローカル・モニターの場合、エージェント・インスタンス名は、モニター対象の Db2 インスタンスの名前と一致している必要があります。

リモート・モニターの場合、モニター対象のリモート Db2 サーバー・インスタンスのローカル・カタログ・ノードです。

2. 以下のパラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、値を指定してから Enter キーを押します。

- ユーザー名
- パスワード
- DB2® SQL パス
- 診断ログのパス
- ダイアログ・メッセージ ID フィルター
- リモート・パーティションのモニター
- すべてのデータベースのモニター

3. 以下のコマンドを実行して、エージェントを開始します。

   ローカル・モニターの場合、Db2 インスタンス所有者ユーザーが *install_dir*/bin/db2-agent.sh start *instance_name* を実行します。

   リモート・モニターの場合、リモート Db2 サーバー・インスタンスがカタログされる Db2 クライアント・インスタンスのインスタンス所有者が *install_dir*/bin/db2-agent.sh start *node_name* を実行します。

次のタスク

- Db2 エージェントの一部の属性のデータを表示するために、Db2 ユーザーに特権を付与します。これらの特権の付与については、255 ページの『Db2 メトリックを表示するための特権の付与』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

構成スクリプトを実行するときに、プロンプトに応答せずにエージェントを構成するには、サイレント応答ファイルを使用します。サイレント応答ファイルは、Windows と Linux のいずれのシステム上でのエージェントの構成にも使用できます。

始める前に

ローカル・モニターおよびリモート・モニター用の Db2 エージェントの構成を開始する前に、リモート・モニターに関する以下のタスクが完了していることを確認してください。

- リモート・モニター用にクライアント/サーバー環境をセットアップします。259 ページの『リモート・モニターの前提条件』を参照してください。

このタスクについて

サイレント応答ファイルには、構成パラメーターがあります。応答ファイル内のパラメーター値を編集し、構成スクリプトを実行して、エージェント・インスタンスを作成し、構成値を更新します。

手順

1. テキスト・エディターで、db2_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。

   Linux AIX *install_dir*/samples/db2_silent_config.txt

   Windows *install_dir*¥tmaitm6_x64¥samples¥db2_silent_config.txt

2. 応答ファイルで、以下のパラメーターの値を指定します。
   - 「**ユーザー名**」に、Db2 インスタンスのユーザー名を入力します。

     ローカル Db2 の場合、Db2 インスタンス所有者の名前を入力します。

     リモート Db2 の場合、リモート Db2 マシンから、実際の Db2 インスタンス所有者の名前を入力します。

重要 : リモート Db2 インスタンスをモニターする場合、このパラメーターは必須です。

- 「**パスワード**」に、Db2 インスタンスのパスワードを入力します。

  ローカル Db2 の場合、Db2 インスタンス所有者のパスワードを入力します。

  リモート Db2 の場合、リモート Db2 マシンから、実際の Db2 インスタンス所有者のパスワードを入力します。

  重要 : リモート Db2 インスタンスをモニターする場合、このパラメーターは必須です。

- 「**DB2 SQL パス**」パラメーターでは、SQL 定義ファイルがデフォルト・ディレクトリーにある場合は、このフィールドをブランクのままにします。そうでない場合は、正しいディレクトリー・パスを入力します。SQL 定義ファイルは以下のデフォルト・パスにあります。

  Linux AIX CANDLEHOME/config/kudcussql.properties 例: **KUD_DB2_SQL_PATH=** /opt/ibm/apm/agent/config/kudcussql.properties

  Windows CANDLEHOME¥TMAITM6_x64¥kudcussql.properties 例: **KUD_DB2_SQL_PATH=** C:¥IBM¥ITM¥TMAITM6_x64¥kudcussql.properties

- 「**ダイアログ・パス**」パラメーターでは、db2diag ログ・ファイルがデフォルト・ディレクトリーにある場合は、このフィールドをブランクのままにします。そうでない場合は、正しいディレクトリー・パスを入力します。ログ・ファイルは以下のデフォルト・パスにあります。

  Linux AIX /home/DB2owner_home_dir/sqllib/db2dump
  例えば、**KUD_DIAGLOG_PATH=** /home/db2inst1/sqllib/db2dump です。

  Windows Windows Install_Driver:#xa5;ProgramData#xa5;IBM#xa5;DB2#xa5;DB2COPY#xa5;DB2INSTANCENAME
  例えば、**KUD_DIAGLOG_PATH=** C:¥ProgramData¥IBM¥DB2¥DB2COPY1¥DB2 です。

  注 : このパラメーターはリモート・モニターでは適用されません。

- 「**ダイアログ・メッセージ ID フィルター**」パラメーターでは、診断ログをフィルターに掛けるための *MSGID* を指定します。MSGID は、メッセージ・タイプ、メッセージ番号、および重大度レベルを組み合わせたものです。また、正規表現を使用することもできます (例えば、**KUD_DIAGLOG_MSGID_FILTER=** ADM1¥d*1E|ADM222¥d2W)。

- 「**リモート・パーティションのモニター**」パラメーターでは、Db2 エージェントがリモート・ホスト内のパーティションをモニターすることを指定する Yes を入力します。例えば、**KUD_MONITOR_REMOTE_PARTITIONS=** *Yes* です。

- 「**すべてのデータベースのモニター**」パラメーターでは、Db2 エージェントがすべてのデータベースをモニターすることを指定する Yes を入力します。例えば、**KUD_MONITOR_ALL_DATABASES=** *Yes* です。

3. db2_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。

   Linux AIX *install_dir*/bin/db2-agent.sh config *instance_name* *install_dir*/samples/db2_silent_config.txt

   Windows *install_dir*¥bin¥db2-agent.bat config*instance_name* ¥tmaitm6_x64¥samples¥db2_silent_config.txt

   <instance_name> は以下のとおりです。

   - ローカル Db2 サーバーをモニターする場合: モニター対象の Db2 サーバー・インスタンス名。
   - リモート Db2 サーバーをモニターする場合: リモート Db2 サーバー・インスタンスのカタログ・ノード名。

   重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

4. Windows の場合は、CANDLEHOME¥TMAITM6_x64¥KUDENV_<instance_name> ファイルを開きます。次に、行 KUD_DB2_CLIENT_INST を KUD_DB2_CLIENT_INST=<client instance name under which remote Db2 server instance is cataloged> に編集します。
5. 以下のコマンドを実行して、エージェントを開始します。

   Linux AIX *install_dir*/bin/db2-agent.sh start *instance_name*

   Windows *install_dir*¥bin¥db2-agent.bat start instance_name

   要確認 : UNIX または Linux からリモート Db2 サーバー・インスタンスをモニターする場合、リモート・サーバー・インスタンスがカタログされるクライアント ・インスタンスの所有者がコマンドを実行する必要があります。

次のタスク

- Db2 エージェントの一部の属性のデータを表示するために、Db2 ユーザーに特権を付与します。これらの特権の付与については、255 ページの『Db2 メトリックを表示するための特権の付与』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Db2 メトリックを表示するための特権の付与

Db2 リソースをモニターするには、Db2 ユーザーは Db2 エージェントの一部の属性のデータを表示するために、Db2 の SYSADM、SYSCTRL、SYSMAINT、および SYSMON の権限が必要です。

このタスクについて

エージェントがすべての属性について収集するモニター・データをダッシュボードに表示するには、Db2 ユーザーに特定の特権が必要です。Db2 ユーザーにこれらの特権を割り当てるには、以下の場所にあるスクリプト・ファイルを実行します。

Linux AIX *install_dir*/config/KudGrantUserPermissions.sh

Windows *install_dir*¥TMAITM6_x64¥KudGrantUserPermissions.bat

SYSADM 権限を持つ Db2 ユーザーがこのスクリプトを実行して、自分自身またはその他の Db2 ユーザーに特権を付与できます。Db2 インスタンスの場合、既に SYSADM 権限を持っているインスタンス所有者を使用してこのスクリプトを実行し、その他の権限を自分自身に付与するか、すべての権限を他の Db2 ユーザーに付与します。

手順

1. ローカル・モニターの場合は、以下のステップを参照してください。
   a) Db2 エージェントがインストールされているシステムで、Db2 コマンド行インターフェースを 開きます。
   b) 以下のコマンドを実行します。ここで、*instance_name* は Db2 インスタンスの名前、*username* は Db2 ユーザーの名前です。

      Linux AIX *install_dir*/config/KudGrantUserPermissions.sh *instance_name username*

      Windows *install_dir*¥TMAITM6_x64¥KudGrantUserPermissions.bat *instance_name username*

      注 : Windows システムの場合、コマンドの *username* はオプションです。コマンドにユーザー名が指定されなかった場合、特権はデフォルト・ユーザー (system) に割り当てられます。
2. リモート・モニターの場合は、以下のステップを参照してください。

a) エージェント・ワークステーションの *install_dir*/TMAITM6_x64/ から Db2 サーバー・マシンに、KudGrantUserPermissions.sh (Unix または Linux 用) および KudGrantUserPermissions.bat (Windows 用) をコピーします。

b) Db2 インスタンス所有者ユーザーから、以下のコマンドを実行します。ここで、*instance_name* は Db2 インスタンスの名前、*username* は Db2 ユーザーの名前です。

Linux AIX `./KudGrantUserPermissions.sh instance_name username`

Windows `KudGrantUserPermissions.bat instance_name username`

要確認 : Windows でのリモート Db2 モニターの場合、*username* は、クライアント・ワークステーションでの Db2 エージェントの構成中に指定されるユーザー名でなければなりません。

ローカル環境変数の構成

ローカル環境変数を構成して、Db2 エージェントの動作を変更することができます。

手順

1. Windows システムでは、「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」 >「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**アクション**」メニューから「**拡張**」>「**ENV ファイルの編集**」をクリックします。
3. Linux システムまたは AIX システムでは、コマンド行に移動し、install_dir/config ディレクトリーの ud.environment ファイルを編集します。ここで、install_dir はエージェント・インストール・ディレクトリーです。

   注 : ud.environment ファイルは、隠しファイルです。
4. 環境変数ファイルに、環境変数の値を入力します。

   構成可能な環境変数については、256 ページの『ローカル環境変数』を参照してください。

ローカル環境変数

Db2 エージェントの動作は、ローカル環境変数を構成することで変更できます。

表スペース・データ・セットのデータ収集方式を定義するための変数

表スペース・データ・セットのデータ収集方式を設定するには、以下の環境変数を使用します。

- **KUD_T1_BY_SQL**: この変数を使用して、SQL 照会を使用した表スペース・データ・セットのデータ収集方式を設定します。SQL 照会を使用したデータ収集を有効にするには、この変数の値を Y に設定します。スナップショット方式を使用して表スペース・データ・セットのデータを収集するには、この変数の値を N に設定します。この変数のデフォルト値は N です。

  重要 : SQL 照会を使用してデータを収集するには、Db2 のバージョンが 9.7 以降である必要があります。また、Db2 エージェントを開始するユーザーは、すべてのデータベースに対する SYSADM 権限がなければなりません。
- **KUD_T1_DISABLE**: この変数を使用して、表スペース・データ・セットのデータ収集を無効にします。表スペース・データ・セットのデータ収集を有効にするには、この変数の値を N に設定します。表スペース・データ・セットのデータ収集を無効にするには、この変数の値を Y に設定します。この変数のデフォルト値は N です。

データ収集からキャッシング・ファシリティー (CF) ノードを除外するための変数

pureScale® 環境のデータ収集アルゴリズムからキャッシング・ファシリティー CF ノードを除外するには、**DB2_CF_PARTITION_NUMS** 変数を使用します。エージェントの環境ファイルで、**DB2_CF_PARTITION_NUMS** 変数を DB2_CF_PARTITION_NUMS=<CF node number> に設定します。例えば、DB2_CF_PARTITION_NUMS=1 です。複数の CF ノードの場合は、**DB2_CF_PARTITION_NUMS** 変数の値を、区切り文字として # . : , ; | @ のいずれかの特殊記号を使用するリストとして設定します。例えば、DB2_CF_PARTITION_NUMS=12,13,23,34 です。この変数にはデフォルト値は設定されません。

Db2 表データ・セットのデータ収集を制限する変数

Db2 エージェントが Db2 表データ・セットのデータの収集中に戻す必要のある行の最大数を設定するには、**KUD_TABLE_NUMBER** 環境変数を使用します。デフォルト値は 10000 です。

カスタマイズ SQL プロパティー・ファイルの再ロード間隔を設定するための変数

カスタマイズ SQL プロパティー・ファイルの再ロード時間間隔 (秒単位) を設定するには、**KUD_CUS_SQL_INTERVAL** 変数を使用します。デフォルト値は 20 秒です。

エージェント・イベント・データ・セットのデータ収集の行数を制限するための変数

エージェント・イベント・データ・セットのデータ収集の行数を設定するには、**KUD_AGENT_EVENT_CACHE** 変数を使用します。エージェント・イベント・データ・セットは、事前定義イベントおよびトリガーされるイベントに関する詳細情報を提供し、モニター対象データベースの正常性に関する問題を判別します。デフォルト値は 50 です。

Db2 ログ・レコード・データ・セットのデータ収集の行を制限するための変数

Db2 ログ・レコード・データ・セットのデータ収集の行数を設定するには、**KUD_DBHISTORY_MAXROW** 変数を使用します。Db2 ログ・レコード・データ・セットは、Db2 アーカイブ・ログに関するヒストリカル情報を提供します。デフォルト値は 500 です。

Db2 診断ログ・データ・セットのデータ収集を定義するための変数

Db2 診断ログ・データ・セットのデータ収集の方式を設定するには、次の環境変数を使用します。

- **KUD_DIAGLOG_BY_TABLE**: この変数を使用して、Db2 診断ログ・データ・セットのデータ収集方式を設定します。この変数の値が Y に設定されている場合、Db2 診断ログ・データ・セットのデータは SQL 照会を使用して収集されます。この変数の値が N に設定されている場合、Db2 診断ログ・データ・セットのデータは db2diag.log を解析することによって収集されます。この変数のデフォルト値は、Y です。

  重要 : SQL 照会を使用してデータを収集するには、Db2 のバージョンが 10 以降である必要があります。
- **KUD_DIAGLOG_TAILCOUNT**: この変数を使用して、Db2 エージェントが DB2 診断ログ・データ・セットのデータを収集するために解析する db2diag.log ファイルの行数を定義します。この変数は、最新のメッセージおよびイベントのみがモニターされるように Db2 エージェント が Db2 エージェント・ログ・ファイルを処理するように制限します。この変数のデフォルト値は、1000 です。
- **KUD_DIAGLOG_CACHE**: この変数を使用して、Db2 診断ログ・データ・セットのダッシュボードに表示されるログ・レコードの数を制限します。この変数のデフォルト値は、20 です。
- **KUD_DIAGLOG_INTERVAL**: この変数を使用して、Db2 診断ログ・データ・セットのデータ収集用の db2diag.log ファイルの再ロード時間間隔 (秒単位) を定義します。この変数のデフォルト値は、30 秒です。
- **KUD_DISABLE_DIAGLOG**: この変数を使用して、Db2 診断ログ・データ・セットのデータ収集を無効にします。Db2 診断ログ・データ・セットのデータ収集を有効にするには、この変数の値を N に設定します。Db2 診断ログ・データ・セットのデータ収集を無効にするには、この変数の値を Y に設定します。この変数のデフォルト値は N です。

照会タイムアウト間隔を設定するための変数

完了までに非常に長い時間を要する SQL 照会がある場合、これは Db2 エージェントのパフォーマンスに影響します。Db2 エージェントの照会タイムアウト間隔を設定するには、**KUD_QUERY_TIMEOUT** 変数を使用します。この変数を使用して、Db2 エージェントに送信された照会に対する応答の受信を Db2 サーバー が待機する最大時間 (秒数) を定義します。この変数の値は、300 秒より少なくする必要があります。この変数のデフォルト値は、45 秒です。

DB2 Database01 (旧) データ・セットのデータ収集を定義するための変数

エージェントは、ASN スキーマが存在しない場合に、DB2 Database01 (旧) データ・セットのデータを収集する ASN 照会をトリガーしてはなりません。ASN 照会の実行を有効にするには、**KUD_REPLICATION_ON** 変数を使用します。この変数の値が Y に設定されている場合、Db2 エージェントは ASN スキーマが存在しなくても ASN 照会を実行します。この変数の値が N に設定されている場合、Db2 エージェントは ASN 照会を実行しません。この変数のデフォルト値は、Y です。

スナップショット方式を使用してデータを収集する際のモニター・スイッチを構成するための変数

スナップショット方式を使用して Db2 エージェント・モニター・データを収集する場合は、データ・セットの Db2 モニター・スイッチを有効にします。Db2 モニター・スイッチを有効にするには、**KUD_MON_SWITCH_OVERRIDE** 変数を使用します。Db2 モニター・スイッチのリストは次のとおりです。

LOCK
ロック情報

SORT
ソート情報

STATEMENT
SQL ステートメント情報

TABLE
表のアクティビティー 情報

TIMESTAMP
タイム・スタンプ取得情報

UOW
作業単位情報

この変数の値が Y に設定されている場合、Db2 エージェントは Db2 モニター・スイッチの構成設定を維持します。この変数の値が N に設定されている場合、Db2 エージェントはすべてのモニター・スイッチを有効にしてデータを収集します。この変数のデフォルト値は、N です。

データ・セットの Db2 スナップショット・バッファー・データをトレースするための変数

データ・セットに対してスナップショット方式を使用して収集されたデータを表示するには、**KUD_SNAPSHOT_DUMPOUT** 変数を使用します。この変数の値が Y に設定されている場合、Db2 エージェントは属性グループのスナップショット・バッファー・データをエージェント・ログ・ファイルにダンプします。この変数の値が N に設定されている場合、Db2 エージェントはスナップショット・バッファー・データをエージェント・ログ・ファイルにダンプしません。この変数のデフォルト値は、N です。

データ・セットのスナップショット・バッファー・データ使用して Db2 エージェントをトレースするための変数

データ・セットについて収集されるスナップショット・バッファー・データ使用して Db2 エージェントをトレースするには、 **KUD_SNAPSHOT_READIN** 変数を使用します。Db2 エージェントのトレースを有効にするには、この変数の値を Y に設定します。Db2 エージェントのトレースを無効にするには、この変数の値を N に設定します。

ロック競合データ・セットのデータ収集方式を定義するための変数

ロック競合データ・セットのデータ収集方式を設定するには、**KUD_LOCKCONFLICT_BY_SQL** 変数を使用します。SQL 照会を使用してロック競合データ・セットのデータを収集するには、この変数の値を Y に設定します。スナップショット方式を使用してロック競合データ・セットのデータを収集するには、この変数の値を N に設定します。この変数のデフォルト値は Y です。

重要 : SQL 照会を使用してデータを収集するには、Db2 のバージョンが 9.7 FP1 以降である必要があります。また、Db2 エージェントを開始するユーザーは、すべてのデータベースに対する SYSADM 権限がなければなりません。

Windows 上のリモート Db2 サーバー をモニターするための変数

KUD_DB2_CLIENT_INST: この変数を、リモート Db2 サーバー・インスタンスがカタログされている Db2 クライアント・インスタンス名に設定します。この変数を設定する必要があるのは、エージェントが Windows 上にあるリモート・モニターを使用する場合のみです。

リモート・モニターの前提条件

リモート・モニターには、Monitoring Agent for Db2 を使用することができます。Db2 のリモート・モニターの前提条件に関するトピックを参照してください。

このタスクについて

Db2 のリモート・モニターの場合は、最初に基本的な Db2 クライアント/サーバー環境のセットアップを行う必要があります。 このセットアップを Windows および UNIX または Linux 向けに行います。

このセットアップでは、ユーザーは Db2 SYSADM または SYSCTRL 権限を持っている必要があります。

要確認 : エージェント・ワークステーション上で、ステップ 2 を除くすべてのステップを実行します。

手順

1. Db2 エージェント・ワークステーションで、Db2 クライアントをインストールします。このクライアントのバージョンは、モニター対象の Db2 サーバー・インスタンスのバージョン以上でなければなりません。
2. Db2 インスタンスの通信プロトコルが TCPIP であることを確認します。

   a) 確認するには、Db2 コマンド・ラインでコマンド **db2set** を実行します。

   b) TCPIP に設定されていない場合は、Db2 コマンド・ラインで **db2set DB2COMM=tcpip** を実行します。

   重要 : このステップはサーバー・サイドで実行します。
3. 次のコマンドを使用して、Db2 エージェント・ワークステーションでリモート・サーバー・インスタンスをカタログします。

   重要 : サーバー・インスタンスは、クライアント・インスタンス下にカタログされます。そのため、クライアント・インスタンス上で次のコマンドを実行します。

   db2=>CATALOG TCPIP NODE<node_name> REMOTE <hostname/ip_address> SERVER <service_name/port_number>

   Db2 の各部の意味は次のとおりです。

   a. *<node_name>* は、クライアント・コンポーネント上の Db2 インスタンスのローカル・ニックネームを示します。

      注 : UNIX または Linux の場合、*<node_name>* は、同じワークステーション上で使用可能ないずれの Db2 クライアントおよび Db2 サーバー・インスタンスとも同じ名前であってはなりません。

   b. <hostname/ip_address> は、Db2 サーバーのワークステーションのホスト名または IP アドレスを示します。

   c. <service_name/port_number> は、Db2 の TCPIP の構成場所を示します。

   リモート・サーバー「**myserver**」のポート番号 50000 で実行中の Db2 サーバー・インスタンスをノード「db2node」としてカタログするには、Db2 コマンド・ラインから次のコマンドを入力します。

   db2 => CATALOG TCPIP NODE db2node REMOTE myserver SERVER 50000

   カタログ・ノードについて詳しくは、https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.qb.client.doc/doc/t0005621.html を参照してください。
4. Db2 エージェント・ワークステーションが UNIX/Linux の場合は、以下を実行します。

   - カタログ・コマンドで使用されているノード名を使用して、ユーザーを作成します。

     以下のコマンドを実行します。

useradd -g <group> -m -d <home_dir> <user> -p <password>

ここで、

– **<group>** は、DB2 UDB インスタンス所有者のグループを表します。

– **<user>** は、クライアント・ワークステーション上のローカルの **username** を表します。**username** は、エージェント・マシン上でサーバー・インスタンスのカタログに使用されているノード名と同じでなければなりません。

- リモート Db2 サーバー・インスタンスがカタログされている Db2 クライアントのインスタンス 名を確認し、このインスタンスの所有者に、新しく作成されたユーザーのホーム・ディレクトリーの読み取り、書き込み、実行の各権限を割り当てます。このステップは、クライアント Db2 環境をリモート・ノード上で操作できるようにするために必要です。
- 以下のコマンドを実行します。

  chmod -R 775 /home/<nodename>

  ここで、

  – **<nodename>** は、クライアント・コンポーネント上の Db2 インスタンスのローカルユーザー名を表します。

5. Db2 エージェント・ワークステーションに存在するクライアント・インスタンスで、モニター対象のすべてのデータベースをカタログします。

   Db2 CLP で以下のコマンドを発行して、データベースをカタログします。

   CATALOG DATABASE <db_name> AS <db_alias> AT NODE <node_name>authentication server

   a. <db_name> は、サーバー・データベース名を示します。

   b. <db_alias> は、Db2 クライアントのデータベースのローカル・ニックネームを示します。

   c. <node_name> は、データベースがカタログされるクライアント・コンポーネント上の Db2 インスタンスのローカル・ニックネームを示します。

   カタログ・ノード「db2node」上に「sample」というデータベースを「dbAlias1」という別名を使用してカタログするには、Db2 プロンプトから以下のコマンドを入力します。

   db2 => CATALOG DATABASE sample AS dbAlias1 AT NODE db2node authentication server

Hadoop モニターの構成

エージェントがモニター対象の Hadoop クラスターのデータを収集できるように、Monitoring Agent for Hadoop を構成する必要があります。エージェントでは、単一ノードの Hadoop クラスターおよびマルチノードの Hadoop クラスターをモニターできます。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Hadoop エージェント を参照してください。

Hadoop エージェントがインストールされたコンピューターから以下のホストが解決できることを確認します。

- NameNode、ResourceManager など、構成するすべての Hadoop ホスト
- NodeManager ロールのみを持つ Hadoop ホスト

例えば、以下の手順を実行してホストを解決できます。

- すべての Hadoop ホストの IP アドレス、ホスト名、および完全修飾ドメイン・ネームを、以下のパスにある hosts ファイルに追加します。

  – Windows `C:¥Windows¥System32¥drivers¥etc¥hosts`

– Linux AIX /etc/hosts

- Hadoop ホストのドメインと同じドメイン内の Hadoop エージェントがインストールされているコンピューターを追加します。

要確認 : Kerberos SPNEGO ベースの認証で保護されている Hadoop クラスターをモニターするには、Hadoop エージェントがインストールされているコンピューターからすべてのホストを解決できることを確認します。

このタスクについて

Hadoop エージェント は、単一インスタンス・エージェントです。 エージェントをインストールした後にエージェントを手動で構成する必要があります。Hadoop エージェントは Windows システム、Linux システム、および AIX システムで構成できます。

要確認 :

- 単一ノードの Hadoop クラスターでは、同じノードがすべての役割 (NameNode、ResourceManager、2 次 NameNode など) を Hadoop クラスターの構成に従って実行します。しかし、マルチノードの Hadoop クラスターでは、複数の Hadoop ノードでこれらの役割を実行します。
- エージェントを構成すると、エージェントは、モニター対象の Hadoop クラスター内にある DataNode および NodeManager を自動的に検出します。

ソケット・ベースのエージェント (8.1.2 フィックスパック 2 以前) から REST API ベースのエージェント (8.1.3 以降) にアップグレードする際に、後述のトピックに記載されている構成手順を実行します。その際、必ず以下のガイドラインに従ってエージェントの構成時にホスト名を指定してください。

- 指定するさまざまなデーモン・プロセス (NameNode、ResourceManger など) のホスト名は、ソケット・ベースのエージェント用に構成されたホスト名と同じ (大/小文字およびフォーマット) でなければなりません。
- ホスト名を指定する際には、完全修飾ドメイン・ネーム (FQDN) を使用する必要があります。例えば、`hos1.ibm.com` です。FQDN の長さが 25 文字を超えている場合は、ドメイン名なしでショート・ホスト名のみを指定します。例えば、ホストの FQDN が *myhadoopclustersetupnode.ibm.com* の場合、ショート・ホスト名は `myhadoopclustersetupnode` です。

アップグレードされるエージェントを 構成し、Cloud APM コンソールでデータを表示した後、Hadoop エージェント用の `hadoop-metrics2.properties` ファイルに加えた変更を元に戻します。詳しくは、1139 ページの『エージェントのアップグレード』を参照してください。

Windows システムでは、非管理者ユーザーで Hadoop エージェントを実行できます。ただし、そのようなユーザーの場合、ダッシュボードでデータを表示するための特定の許可が必要になります。この許可を付与する方法については、269 ページの『非管理者ユーザーへの許可の付与』を参照してください。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows システムで構成することができます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Hadoop**」を右クリックします。
3. 「**エージェントを構成します**」をクリックします。

重要：「**エージェントを構成します**」が無効になっている場合は「**再構成**」をクリックします。

「**Monitoring Agent for Hadoop の構成**」ウィンドウが開きます。

4. Kerberos SPNEGO ベースの認証が有効になっている Hadoop クラスターをモニターするには、以下のステップを実行します。
   a)「**Hadoop クラスターで、HTTP ベースの Hadoop サービスに対する Kerberos SPNEGO ベースの認証が有効になっているかどうか**」で「**はい**」をクリックします。

   Hadoop クラスター内の HTTP ベースの Hadoop サービスの REST エンドポイントを保護するために、Kerberos SPNEGO ベースの認証を使用しない場合は、「**いいえ**」をクリックします。「**レルム名**」、「**KDC ホスト名**」、「**SPNEGO プリンシパル名**」、および「**SPNEGO キータブ・ファイル**」の各フィールドの値は空白のままでかまいません。

   b)「**レルム名**」フィールドに、サービス・プリンシパルの作成に使用する Kerberos レルムの名前を入力します。
   通常、レルム名はドメイン名と同じです。例えば、コンピューターが `tivoli.ibm.com` ドメインにある場合、Kerberos レルム名は `TIVOLI.IBM.COM` です。この名前では、大/小文字が区別されます。

   c)「**KDC ホスト名**」フィールドに、指定したレルムの鍵配布センター (KDC) ホストの完全修飾ドメイン名 (FQDN) を入力します。

   FQDN の代わりに、KDC ホストの IP アドレスを指定することもできます。Active Directory KDC の場合、ドメイン・コントローラーは KDC ホストです。

   d)「**SPNEGO プリンシパル名**」フィールドに、HTTP ベース・サービスの SPNEGO 認証済み REST エンドポイントへのアクセスに使用する Kerberos プリンシパルの名前を入力します。

   この名前では、大/小文字が区別されます。名前の形式は HTTP/*fully_qualified_host_name*@*kerberos_realm* です。

   e)「**SPNEGO キータブ・ファイル**」フィールドに、SPNEGO サービスのキータブ・ファイルの名前とその絶対パスを入力するか、「**参照**」をクリックして、これを選択します。

   キータブ・ファイルには、Kerberos サービス・プリンシパルの名前とキーが含まれます。このファイルにより、各サービスのパスワードなしで Hadoop サービスに直接アクセスできます。このファイルは、パス `etc/security/keytabs/` で見つけることができます。
   SPNEGO プリンシパル名およびキータブ・ファイルが同じホストに属していることを確認します。例えば、プリンシパル名が *HTTP/abc.ibm.com@IBM.COM* の場合、使用するキータブ・ファイルは *abc.ibm.com* ホストに属している必要があります。
   エージェントがリモート・コンピューターにインストールされている場合、プリンシパルのキータブ・ファイルをリモート・コンピューターの任意のパスにコピーし、「**SPNEGO キータブ・ファイル**」フィールドでこのパスを指定します。

   f)「**次へ**」をクリックします。
5. HTTPS/SSL が有効になっている Hadoop クラスターをモニターするには、以下のステップを実行します。
   a)「**Hadoop クラスターで SSL が有効になっていますか (Is Hadoop Cluster SSL enabled)**」で、「**はい**」をクリックします。

   SSL が有効になっている Hadoop クラスターを使用しない場合は、「**いいえ**」を選択します。「**トラストストア・ファイル・パス (TrustStore file path)**」、「**トラストストア・パスワード (TrustStore Password)**」の各フィールドの値は空白のままでかまいません。

   b)「**トラストストア・ファイル・パス (TrustStore file path)**」には、ローカル・マシンに保管されているトラストストア・ファイルを選択します。

   このファイルは、Hadoop クラスターからローカル・マシンにコピーして、構成に使用できます。

   c)「**トラストストア・パスワード (TrustStore Password)**」には、トラストストア・ファイルの構成時に作成したパスワードを入力します。
6. Hadoop クラスターのパラメーター値を指定するために、以下のステップを実行します。
   a) **Unique Hadoop Cluster Name** フィールドに、Hadoop のバージョンとフレーバーを示す、Hadoop クラスターの固有の名前を入力します。このフィールドの最大文字数は 12 文字です。

b)「**NameNode ホスト名 (NameNode Hostname)**」フィールドに、NameNode のデーモン・プロセスが実行されるノードのホスト名を入力します。

c)「**NameNode ポート (NameNode Port)**」フィールドに、NameNode のデーモン・プロセスに関連付けられているポート番号を入力します。デフォルトのポート番号は 50070 です。

d)「**ResourceManager ホスト名 (ResourceManager Hostname)**」フィールドに、ResourceManager のデーモン・プロセスが実行されるノードのホスト名を入力します。

e)「**ResourceManager ポート (ResourceManager Port)**」フィールドに、ResourceManager のデーモン・プロセスに関連付けられているポート番号を入力します。デフォルトのポート番号は 8088 です。

f) オプション:「**JobHistoryServer ホスト名 (JobHistoryServer Hostname)**」フィールドに、JobHistoryServer のデーモン・プロセスが実行されるノードのホスト名を入力します。

g) オプション:「**JobHistoryServer ポート (JobHistoryServer Port)**」フィールドに、JobHistoryServer のデーモン・プロセスに関連付けられているポート番号を入力します。デフォルトのポート番号は 19888 です。

h) オプション:「**追加 NameNode ホスト名 (Additional NameNode Hostname)**」フィールドに、スタンバイ NameNode または 2 次 NameNode のデーモン・プロセスが実行されるホスト名を入力します。

i) オプション:「**追加 NameNode ポート (Additional NameNode Port)**」フィールドに、スタンバイ NameNode または 2 次 NameNode のデーモン・プロセスに関連付けられたポート番号を入力します。

要確認 : 追加の NameNode がスタンバイ NameNode の場合、スタンバイ NameNode デーモン・プロセスに関連付けられるデフォルト・ポート番号は 50070 です。追加の NameNode が 2 次 NameNode の場合、2 次 NameNode デーモン・プロセスに関連付けられるデフォルト・ポート番号は 50090 です。

j)「**接続のテスト**」をクリックして、指定したホスト名とポートへの接続を検証します。
「**接続のテスト**」をクリックした後、以下の場合に該当する検証メッセージが表示されます。

- 指定したホスト名およびポートへの接続が成功した、または失敗した。
- ホスト名の値がブランクのままになっている。
- ポートの値がブランクのままになっている。
- ポート番号に非整数値が指定されている。

検証メッセージの推奨に従って構成値を更新して、再度接続を確認してください。

k) オプション: Hadoop クラスターのスタンバイ ResourceManager を追加するには、「**Hadoop クラスター内のスタンバイ ResourceManager**」で「**はい**」をクリックします。
スタンバイ ResourceManager の詳細を追加するよう要求するプロンプトが後で出されます。

l) オプション: Apache Ambari によって管理される Hadoop クラスターで Hadoop サービスをモニターするには、「**Ambari ベースの Hadoop インストール済み環境に対する Hadoop サービスのモニター**」で「**はい**」をクリックし、「**次へ**」をクリックします。

7. オプション: Hadoop サービスをモニターする Ambari サーバーの詳細を指定するには、以下のステップを実行します。

a)「**Ambari サーバーのホスト名**」フィールドに、Ambari サーバーが実行されるホストの名前を入力します。

b)「**Ambari サーバーのポート**」フィールドに、Ambari サーバーに関連付けられているポート番号を入力します。
デフォルトのポート番号は 8080 です。

c)「**Ambari ユーザーのユーザー名**」フィールドに、Ambari ユーザーの名前を入力します。

d)「**Ambari ユーザーのパスワード**」フィールドに、Ambari ユーザーのパスワードを入力します。

e)「**次へ**」をクリックします。

8. Java パラメーターの値を指定するために、以下のステップを実行します。

a)「**Java トレース・レベル**」から、Java プロバイダーで使用されるトレース・レベルの値を選択します。

b) オプション:「**JVM 引数**」フィールドに、Java 仮想マシンの引数のリストを指定します。
引数のリストは、エージェントとともにインストールされている Java のバージョンと互換性がなければなりません。

c)「**次へ**」をクリックします。

9. オプション: スタンバイ ResourceManager を追加するために、以下のステップを実行します。

a)「**新規**」をクリックします。

b)「**スタンバイ ResourceManager ホスト名 (Standby ResourceManager Hostname)**」フィールドに、スタンバイ ResourceManager のデーモン・プロセスが実行されるノードのホスト名を入力します。

c)「**スタンバイ ResourceManager ポート (Standby ResourceManager Port)**」フィールドに、スタンバイ ResourceManager のデーモン・プロセスに関連付けられているポート番号を入力します。デフォルトのポート番号は 8088 です。

d)「**接続のテスト**」をクリックして、指定したホスト名とポート番号への接続を検証します。
「**接続のテスト**」をクリックした後、以下の場合に該当する検証メッセージが表示されます。

- 指定したホスト名およびポートへの接続が成功した、または失敗した。
- ホスト名の値がブランクのままになっている。
- ポートの値がブランクのままになっている。
- ポート番号に非整数値が指定されている。

検証メッセージの推奨に従って構成値を更新して、再度接続を確認してください。

e) スタンバイ ResourceManager をさらに追加するには、ステップ a、b、および c を繰り返します。
スタンバイ ResourceManager のいずれかを削除する場合は、削除するスタンバイ ResourceManager に対応する「**削除**」をクリックします。

f)「**次へ**」をクリックします。

10.「**外部 JAR のクラスパス**」フィールドに、JAR ファイルのクラスパスを指定します。
このクラス・パスは、エージェントによって生成されるクラスパスに追加されます。このフィールドは、ブランクのままでもかまいません。

11.「**OK**」をクリックします。
指定した構成設定が保存されました。

12.「**Monitoring Agent for Hadoop**」を右クリックし、「**開始**」をクリックします。

次のタスク

1. Hadoop エージェントのイベントしきい値を表示するために、サブノード・イベントを有効にします。サブノード・イベントの有効化については、269 ページの『Hadoop イベントを表示するためのダッシュボードの構成』を参照してください。
2. Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Linux および AIX システム上でのエージェントの構成

Linux システムおよび AIX システムでエージェントを構成するには、構成スクリプトを実行してプロンプトに応答します。

手順

1. コマンド行で、コマンド *install_dir*/bin/hadoop-agent.sh config を実行します。
ここで *install_dir* は Hadoop エージェントのインストール・ディレクトリーです。
エージェントは以下のデフォルトのインストール・ディレクトリーにインストールされます。
/opt/ibm/apm/agent

2. コマンド行に以下のメッセージが表示されたら、1 と入力して Enter キーを押し、構成ステップを続行します。
   Monitoring Agent for Hadoop の設定を編集しますか? (Edit 'Monitoring Agent for Hadoop' setting?) [1= はい、2= いいえ] ([1= yes, 2= No])
3. コマンド行に以下のメッセージが表示されたら、1 と入力し、Kerberos SPNEGO ベースの認証が有効になっている Hadoop クラスターをモニターするための値を指定し、Enter キーを押します。 それ以外の場合、2 と入力し、Enter キーを押します。「**レルム名**」、「**KDC ホスト名**」、「**SPNEGO プリンシパル名**」、および「**SPNEGO キータブ・ファイル**」の各フィールドの値はブランクのままでかまいません。
   Hadoop クラスター内の HTTP ベースの Hadoop サービスに対して Kerberos SPNEGO ベースの認証が有効になっていますか¥: [1= はい、2= いいえ (デフォルトは 2 です) (Is Kerberos SPNEGO-based authentication for HTTP based Hadoop services in Hadoop cluster enabled¥: [1=Yes, 2=No (default is: 2))
   a) 「**レルム名**」パラメーターに、サービス・プリンシパルの 作成に使用する Kerberos レルムの名前を入力します。
      通常、レルム名はドメイン名と同じです。例えば、コンピューターが tivoli.ibm.com ドメインにある場合、Kerberos レルム名は TIVOLI.IBM.COM です。この名前では、大/小文字が区別されます。
   b) 「**KDC ホスト名**」フィールドに、指定したレルムの鍵配布センター (KDC) ホストの完全修飾ドメイン名 (FQDN) を入力します。FQDN の代わりに、KDC ホストの IP アドレスを指定することもできます。Active Directory KDC の場合、ドメイン・コントローラーは KDC ホストです。
   c) 「**SPNEGO プリンシパル名**」パラメーターに、HTTP ベース・サービスの SPNEGO 認証済み REST エンドポイントへのアクセスに使用する Kerberos プリンシパルの 名前を入力します。
      この名前では、大/小文字が区別されます。名前の形式は HTTP/*fully_qualified_host_name*@*kerberos_realm* です。
   d) 「**SPNEGO キータブ・ファイル**」パラメーターに、SPNEGO サービスのキータブ・ファイルの名前とその絶対パスを入力します。
      キータブ・ファイルには、Kerberos サービス・プリンシパルの 名前とキーが含まれます。このファイルにより、各サービスのパスワードなしで Hadoop サービスに直接アクセスできます。このファイルは、パス etc/security/keytabs/ で見つけることができます。
      SPNEGO プリンシパル名およびキータブ・ファイルが同じホストに属していることを確認します。例えば、プリンシパル名が *HTTP/abc.ibm.com@IBM.COM* の場合、使用するキータブ・ファイルは *abc.ibm.com* ホストに属している必要があります。
      エージェントがリモート・コンピューターにインストールされている場合、プリンシパルのキータブ・ファイルをリモート・コンピューターの任意のパスにコピーし、「**SPNEGO キータブ・ファイル**」パラメーターにこのパスを指定します。
4. コマンド行に以下のメッセージが表示されたら、1 と入力し、SSL が有効になっている Hadoop クラスターをモニターするための値を指定し、**Enter** キーを押します。それ以外の場合は、2 と入力し、**Enter** キーを押します。「**トラストストア・ファイル・パス (TrustStore file path)**」および「**トラストストア・パスワード (TrustStore Password)**」の各フィールドの値はブランクのままでかまいません。
   Hadoop クラスターで SSL が有効になっていますか [1= はい、2= いいえ] (デフォルトは 2 です) (Is Hadoop Cluster SSL enabled [1=Yes, 2=No (default is: 2))
   a) **「トラストストア・ファイル・パス (TrustStore file path)」**には、ローカル・マシンに保管されているトラストストア・ファイルのパスを指定します。
      このファイルは、Hadoop クラスターからローカル・マシンにコピーして、構成に使用できます。
   b) **「トラストストア・パスワード (TrustStore Password)」**には、トラストストア・ファイルの構成時に作成したパスワードを指定します。
5. Hadoop クラスターの詳細を入力するよう求めるプロンプトが出されたら、以下の各パラメーターに適切な値を指定し、Enter キーを押します。
   a) **Unique Hadoop Cluster Name** には、Hadoop のバージョンとフレーバーを示す、Hadoop クラスターの固有の名前を指定します。このフィールドの最大文字数は 12 文字です。
   b) **NameNode Hostname** パラメーターには、NameNode のデーモン・プロセスが実行されるノードのホスト名を指定し、Enter キーを押します。

重要 : ホスト名を指定せずに Enter キーを押すと、ホスト名を入力するよう要求するプロンプトが出されます。

c) **NameNode Port** パラメーターには、NameNode のデーモン・プロセスに関連付けられたポート番号を指定し、Enter キーを押します。デフォルトのポート番号は 50070 です。

d) **ResourceManager Hostname** パラメーターには、ResourceManager のデーモン・プロセスが実行されるノードのホスト名を指定し、Enter キーを押します。

重要 : ホスト名を指定せずに Enter キーを押すと、ホスト名を入力するよう要求するプロンプトが出されます。

e) **ResourceManager Port** パラメーターには、ResourceManager のデーモン・プロセスに関連付けられたポート番号を入力します。デフォルトのポート番号は 8088 です。

6. オプション: Hadoop クラスターの以下のパラメーターの詳細を追加するよう要求するプロンプトが出されたら、デフォルト値をそのまま使用するか、以下の各パラメーターに適切な値を指定し、Enter キーを押します。

a) **JobHistoryServer Hostname** パラメーターには、JobHistoryServer のデーモン・プロセスが実行されるノードのホスト名を入力します。

b) **JobHistoryServer Port** パラメーターには、JobHistoryServer のデーモン・プロセスに関連付けられたポート番号を入力します。デフォルトのポート番号は 19888 です。

c) **Additional Hostname** パラメーターには、2 次 NameNode またはスタンバイ NameNode のデーモン・プロセスが実行されるノードのホスト名を入力します。

d) **Additional Port** パラメーターには、2 次 NameNode またはスタンバイ NameNode のデーモン・プロセスに関連付けられたポート番号を入力します。2 次 NameNode のデフォルトのポート番号は 50090 です。スタンバイ NameNode の場合、デフォルトのポート番号は 50070 です。

7. オプション: コマンド行に以下のメッセージが表示されたら、高可用性クラスター用のスタンバイ ResourceManger の詳細を追加するために 1 と入力して Enter キーを押します。
`Hadoop クラスターのスタンバイ ResourceManager [1= はい、2= いいえ] (デフォルト: 2) (Standby ResourceManager(s) in Hadoop Cluster [ 1=Yes, 2=No ] (default is: 2)):`

8. コマンド行に以下のメッセージが表示されたら、Ambari によって管理される Hadoop クラスター内の Hadoop サービスをモニターするために、1 を指定し、Enter キーを押します。
`Ambari ベースの Hadoop インストール環境の Hadoop サービスのモニター [ 1= はい、2= いいえ ] (デフォルトは 2 です): (Monitoring of Hadoop services for Ambari based Hadoop installations [ 1=Yes, 2=No ] (default is: 2):)`
有効にしない場合、デフォルト値の 2 をそのまま使用し、Enter キーを押します。Hadoop サービスのモニターを有効にする場合、Ambari サーバーの以下の各パラメーターに値を指定し、Enter キーを押します。

a)「**Ambari サーバーのホスト名**」パラメーターに、Ambari サーバーが実行されているホストの名前を入力します。

b)「**Ambari サーバーのポート**」パラメーターに、Ambari サーバーに関連付けられているポート番号を入力します。
デフォルトのポート番号は 8080 です。

c)「**Ambari ユーザーのユーザー名**」パラメーターに、Ambari ユーザーの名前を入力します。

d)「**Ambari ユーザーのパスワード**」パラメーターに、Ambari ユーザーのパスワードを入力します。

9. コマンド行に以下のメッセージが表示されたら、適切な Java トレース・レベルを選択して Enter キーを押します。
`このパラメーターでは、Java プロバイダーで使用されるトレース・レベルを指定できます Java トレース・レベル [ 1=オフ、2=エラー、3=警告、4=情報、5=デバッグ詳細度 - 低、6=デバッグ詳細度 - 中、7=デバッグ詳細度 - 高、8=すべて ] ([ 1=Off, 2=Error, 3=Warning, 4=Information, 5=Minimum Debug, 6=Medium Debug, 7=Maximum Debug, 8=All ]) (デフォルトは 2)`

10. オプション: コマンド行に以下のメッセージが表示されたら、Java 仮想マシンの引数を指定して Enter キーを押します。引数のリストは、エージェントと一緒にインストールされている Java のバージョンと互換性がなければなりません。
    このパラメーターでは、Java 仮想マシンの引数のリストを指定できます（任意指定） JVM 引数（デフォルト:）（default is:）
11. オプション: コマンド行に以下のメッセージが表示されたら、スタンバイ ResourceManager の以下の詳細を追加するために 1 と入力して Enter キーを押します。
    「スタンバイ ResourceManager を使用する Hadoop 高可用性（HA）クラスター」設定の編集、[1=追加、2=編集、3=削除、4=次へ、5=終了]（デフォルト: 5): 1（Edit 'Hadoop High Availability(HA) Cluster with Standby ResourceManagers' settings, [1=Add, 2=Edit, 3=Del, 4=Next, 5=Exit] (default is: 5): 1）
    a) **Standby ResourceManager Hostname** パラメーターには、スタンバイ ResourceManger のデーモン・プロセスが実行されるノードのホスト名を入力します。
    b) **Standby ResourceManager Port** には、スタンバイ ResourceManager のデーモン・プロセスに関連付けられているポート番号を入力します。デフォルトのポート番号は 8088 です。
    c) プロンプトが出されたら、スタンバイ ResourceManager をさらに追加するために 1 を入力してステップ a および b を繰り返すか、または 5 を入力して次のステップに進みます。

- 特定のスタンバイ ResourceManager の構成設定を編集するには、必要なスタンバイ ResourceManager のホスト名が表示されるまで 4 を入力して Enter キーを押します。
- スタンバイ ResourceManager を削除するには、削除するスタンバイ ResourceManger のホスト名が表示された後に 3 と入力し、Enter キーを押します。

12. プロンプトが表示されたら、Java API データ・プロバイダーで必要となる JAR ファイルのクラスパスを入力し、Enter を押します。
    指定された構成値が保存され、確認メッセージが表示されます。
13. エージェントを開始するには、コマンド *install_dir*/bin/hadoop-agent.sh start を実行します

次のタスク

1. Hadoop エージェントのイベントしきい値を表示するために、サブノード・イベントを有効にします。サブノード・イベントの有効化については、269 ページの『Hadoop イベントを表示するためのダッシュボードの構成』を参照してください。
2. Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。一部のパラメーターでは、デフォルト値がコメントで示されています。これらのパラメーターに別の値を指定したり、パラメーターの先頭にあるコメント・タグを削除したりすることができます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システム、AIX システム、および Windows システム上で Hadoop エージェントを構成できます。

手順

1. パス *install_dir*¥samples¥hadoop_silent_config.txt にあるサイレント応答ファイルを開きます。
2. 応答ファイルで、以下の手順を実行します。
    a) Kerberos SPNEGO ベースの認証が有効になっている Hadoop クラスターをモニターする場合、yes を指定し、以下のパラメーターの値を入力します。

```
HADOOP_REALM_NAME
HADOOP_KDC_HOSTNAME
HADOOP_PRINCIPAL_NAME
HADOOP_SPNEGO_KEYTAB
```

b) SSL が有効になっている Hadoop クラスターをモニターする場合、yes と入力し、以下のパラメーターの値を入力します。

```
HADOOP_TRUSTSTORE_PATH
HADOOP_TRUSTSTORE_PASSWORD
```

c) Cluster、NameNode (NN)、ResourceManager (RM)、および Job History Server (JHS) の以下のパラメーターに対して値を入力します。

```
HADOOP_CLUSTER_NAME (optional)
HADOOP_NN_HOSTNAME
HADOOP_NN_PORT
HADOOP_RM_HOSTNAME
HADOOP_RM_PORT
HADOOP_JHS_HOSTNAME (optional)
HADOOP_JHS_PORT (optional)
```

d) オプション: **HADOOP_ADDITIONAL_NN_HOSTNAME** パラメーターに、スタンバイまたは 2 次 NameNode のホスト名を指定します。

e) オプション: **HADOOP_ADDITIONAL_NN_PORT** パラメーターに、スタンバイまたは 2 次 NameNode のポート番号を指定します。

要確認 : 追加の NameNode がスタンバイ NameNode の場合、スタンバイ NameNode デーモン・プロセスに関連付けられるデフォルト・ポート番号は 50070 です。追加の NameNode が 2 次 NameNode の場合、2 次 NameNode デーモン・プロセスに関連付けられるデフォルト・ポート番号は 50090 です。

f) オプション: **Hadoop_SRM** パラメーターに、高可用性クラスターのスタンバイ ResourceManager を追加するために Yes を指定し、ステップ g に進みます。

g) オプション: Ambari によって管理される Hadoop クラスターで Hadoop サービスをモニターするには、以下の各パラメーターの値を入力し、Enter キーを押します。

```
AMBARI_SERVER_HOSTNAME
AMBARI_SERVER_PORT
USERNAME_OF_AMBARI_USER
PASSWORD_OF_AMBARI_USER
```

h) **JAVA_TRACE_LEVEL** パラメーターに、適切なトレース・レベルを指定します。

i) オプション: **JAVA_JVM_ARGS** パラメーターに Java™ 仮想マシンの引数を指定します。

j) オプション: スタンバイ ResourceManager のホスト名およびポート番号を *hadoop_srm_config_sec_1*=8088 の形式で追加します。

ここで、*hadoop_srm_config_sec_1* は、スタンバイ ResourceManager のデーモン・プロセスが実行されるノードのホスト名で、8088 はデフォルトのポート番号です。さらにスタンバイ ResourceManager を追加するには、その他のスタンバイ ResourceManager のホスト名とポート番号を新しい行に同じ形式で追加します。

3. 応答ファイルを保存し、以下のコマンドを実行します。

Linux AIX *install_dir*/bin/hadoop-agent.sh config *install_dir*/samples/hadoop_silent_config.txt

Windows *install_dir*/bin/hadoop-agent.bat config *install_dir*/samples/hadoop_silent_config.txt

4. 以下のようにして、エージェントを開始します。

Linux AIX コマンド *install_dir*¥bin¥hadoop-agent.sh start を実行します。

Windows 「**Monitoring Agent for Hadoop**」を右クリックし、「**開始**」をクリックします。

次のタスク

1. Hadoop エージェントのイベントしきい値を表示するために、サブノード・イベントを有効にします。サブノード・イベントの有効化については、269 ページの『Hadoop イベントを表示するためのダッシュボードの構成』を参照してください。
2. Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Hadoop イベントを表示するためのダッシュボードの構成

「**イベント**」タブに Hadoop イベントを表示できるようにするには、ダッシュボードを構成してサブノード・イベントを有効にする必要があります。

このタスクについて

「**サブノード・イベントの有効化**」のデフォルト値は false です。Hadoop イベントを表示するには、この値を true に変更します。

手順

1. Cloud APM コンソール を開いて、「**システム 構成**」に移動します。
2. 「**拡張構成**」ページで、「**構成カテゴリー**」の下にある「**UI 統合**」をクリックします。
3. 「**サブノード・イベントの有効化**」リストから「**True**」を選択します。
4. 「**保存**」をクリックします。

非管理者ユーザーへの許可の付与

Windows システムでは、Hadoop エージェントを実行するために、「プログラムのデバッグ」許可を非管理者ユーザーに付与します。この許可は、Hadoop エージェントのダッシュボードでデータを表示するために必要です。

手順

Hadoop エージェントがインストールされているコンピューターで以下のステップを実行します。

1. 「**スタート**」>「**コントロール パネル**」>「**管理ツール**」をクリックします。
2. 「**ローカル セキュリティ ポリシー**」をダブルクリックします。
3. 「セキュリティの設定」ペインで、「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。
4. 「**プログラムのデバッグ**」を右クリックして「**プロパティ**」をクリックします。
5. 「**ユーザーまたはグループの追加**」をクリックし、この許可を付与する非管理者ユーザー名を追加します。
6. 「**OK**」をクリックします。

次のタスク

非管理者ユーザーで Hadoop エージェントを構成して実行します。

HMC Base モニターの構成

Monitoring Agent for HMC Base には、ハードウェア管理コンソール (HMC) をモニターする機能が用意されています。このエージェントは、HMC リソース (CPU、メモリー、ストレージ、ネットワーク) の可用性と正常性をモニターします。また、このエージェントは、Power サーバー、CPU プール、および LPAR の HMC インベントリーおよび構成について報告します。Power サーバー、LPAR、およびプールの CPU 使用率は、HMC パフォーマンス・サンプル・データを使用してモニターされます。

始める前に

HMC Base エージェントを構成する前に、以下のタスクを完了する必要があります。

- エージェントを実行するシステムと HMC の間で SSH 接続をセットアップします。詳しくは、271 ページの『SSH 接続のセットアップ』を参照してください。
- 最初のエージェント・インスタンスを開始する前に HMC SDK を準備します。詳しくは、272 ページの『HMC 用の SDK の準備』を参照してください。

手順

- サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。
  1. テキスト・エディターで hmc_base_silent_config.txt ファイルを開きます。
     - AIX *install_dir*/samples/hmc_base_silent_config.txt。
  2. **「HMC ホスト名」**には、IP アドレスまたはホスト名を指定できます。
  3. **「HMC ユーザー名」**には、HMC のログオン・ユーザー名を入力する必要があります (例: **HMC_USERNAME=** *hscroot*)。

     注 : HMC に割り当てるログオン・ユーザー名には、最低でも hscviewer 権限が必要です。
  4. **「HMC パスワード」**に、ユーザーのパスワードを入力する必要があります。
  5. **「データ・プロバイダー・ログ・ファイルの最大数」**には、作成されるデータ・プロバイダー・ログ・ファイルの最大数を指定する必要があります。例えば、**KPH_LOG_FILE_MAX_COUNT=***10* です。
  6. **「各データ・プロバイダー・ログの最大サイズ (KB)」**には、新規ログ・ファイルが作成されるまでにデータ・プロバイダー・ログ・ファイルが到達できる最大サイズ (KB) を入力する必要があります (例: **KPH_LOG_FILE_MAX_SIZE=** *5190*)。
  7. **「データ・プロバイダー・ログの詳細レベル」**には、データ・プロバイダーがデータ・プロバイダー・ログ・ファイルに含める詳細のレベルを入力する必要があります (例: **KPH_LOG_LEVEL=***Fine*)。 以下の値のいずれかを指定する必要があります。
     - 1= Off
     - 2=Severe
     - 3=Warning
     - 4=Info
     - 5=Fine
     - 6=Finer
     - 7=Finest
     - 8=All

     重要 : デフォルト値は 4 です。
  8. hmc_base_silent_config.txt ファイルを保存して閉じ、./hmc_base-agent.sh config *instance_name install_dir*/samples/hmc_base_silent_config.txt と入力します。ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は HMC Base エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
- AIX プロンプトに応答を入力してエージェントを構成する場合は、以下のステップを実行します。
  1. *install_dir*/bin ディレクトリーを開きます。ここで、*install_dir* は HMC Base エージェントのインストール・ディレクトリーです。
  2. HMC Base エージェントを構成するには、コマンド **./hmc_base-agent.sh config instance_name** を実行します。
  3. **「Monitoring Agent for HMC Base 設定の編集 (edit the Monitoring Agent for HMC Base settings)」**のプロンプトが出されたら、**Enter** キーを押します。デフォルト値は「はい」です。
  4. HMC 構成情報を入力するには、以下のステップを実行します。

a. **HMC ホスト名**を求めるプロンプトが出されたら、ホスト名または IP アドレスを入力し、**Enter** キーを押します。
b. **「HMC ユーザー名」**の入力を求めるプロンプトが出されたら、HMC に関連付けられたログオン・ユーザー名を入力して、**Enter** キーを押します。

5. **「HMC パスワード」**を求めるプロンプトが出されたら、ユーザーのパスワードを入力します。
6. データ・プロバイダー情報を入力するには、以下のステップを実行します。
   a. **「データ・プロバイダー・ログ・ファイルの最大数」**を求めるプロンプトが出されたら、ログ・ファイルの量を入力し、**Enter** キーを押します。

      データ・プロバイダー・ログ・ファイルの最大数のデフォルトは 10 です。
   b. **「各データ・プロバイダー・ログの最大サイズ (KB)」**を求めるプロンプトが出されたら、サイズを入力し、**Enter** キーを押します。

      デフォルトの最大サイズ (KB) は 5190 です。
   c. **「データ・プロバイダー・ログの詳細レベル」**を求めるプロンプトが出されたら、以下のいずれかのレベルを入力し、**Enter** キーを押します。
      - 1= Off
      - 2=Severe
      - 3=Warning
      - 4=Info
      - 5=Fine
      - 6=Finer
      - 7=Finest
      - 8=All

次のタスク

- エージェントを開始するには、`./hmc_base-agent.sh start` *InstanceName* と入力します。
- 仮想入出力をモニターするために、274 ページの『仮想入出力をモニターするための HMC コンソール・サーバーの構成』の説明に従って HMC コンソール・サーバーを構成します。
- 274 ページの『CPU およびメモリー使用率のモニターの有効化』の説明に従って CPU およびメモリーの使用状況のモニターを有効にします。

SSH 接続のセットアップ

エージェントにデータを収集させるには、エージェントが稼動しているシステムと HMC の間の SSH 接続をセットアップする必要があります。

このタスクについて

CLI コマンドを実行することで、エージェント・データ・プロバイダーが SSH 経由で管理コンソールからデータを収集します。デフォルトでは、データ・プロバイダーは CLI コマンドが実行を完了するまで最大で 1 分間待機します。この時間が過ぎると、データ・プロバイダーは、CLI コマンドを実行中の SSH セッションを閉じます。このコマンドが正常に実行されるまでは、このコマンドでのデータをエージェント・データ・セットで使用することはできません。SSH コマンドのデフォルト・パスは `/usr/bin/ssh` です。SSH を別の場所にインストールした場合は、**KPH_SSH_PATH** 環境変数を使用してパスを指定する必要があります。

手順

SSH 接続をセットアップするには、以下のいずれかの方法を実行します。

- `setup_hmc_key.pl` スクリプトを使用して SSH 接続をセットアップします。

  a) エージェントのインストール先サーバーにログオンします。

b) install_dir/aix526/ph/bin ディレクトリーを開きます。ここで、install_dir は、HMC Base エージェントのインストール・ディレクトリーです。

c) perl setup_hmc_key.pl コマンドを実行します。

d) プロンプトに応答して、HMC ホスト名または IP アドレス、HMC ユーザー名 (hscviewer 権限と同等の権限を持つもの)、および鍵ペア作成用のパスワードを指定します。

e) 鍵ペアを作成した後、ssh hscroot@hmchost lshmc -V などのコマンドを実行して接続をテストします。

SSH がこの HMC に初めて接続する場合は、以下のメッセージに yes と応答して、HMC を ssh known_hosts ファイルに追加します。

```
The authenticity of host 'hmchost (3.3.333.333)' can't be established.
RSA key fingerprint is 4c:b4:26:27:38:f3:ec:58:01:92:26:f9:61:32:bb:4d.
Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'hmchost,3.3.333.333' (RSA) to the list of known hosts.
```

これで、エージェントが SSH を使用して HMC からデータを収集できるようになります。

- ssh-keygen ユーティリティーを 使用して鍵を生成し、SSH 接続をセットアップします。

a) エージェントのインストール先サーバーにログオンします。

b) ssh-keygen ユーティリティーを 使用して、パスフレーズを持たない公開鍵および秘密鍵を生成します。
例えば、次のコマンドによって公開鍵および秘密鍵のセットが生成されます。

```
ssh-keygen -t rsa -f /.ssh/id-rsa
```

パスフレーズを求めるプロンプトが出されたら Enter キーを押します。生成された公開鍵は /.ssh/id-rsa.pub ファイルに保管されます。秘密鍵は /.ssh/id-rsa ファイルに保管されます。

c) scp などのユーティリティーを 使用して、公開鍵を含むファイルを HMC コンピューターに転送します。

d) HMC コンピューターで、HMC に保管されている鍵のコレクションに公開鍵ファイルを追加します。
保管されている鍵は /.ssh/authorized_keys2 ファイルにあります。

e) HMC のホスト名および鍵を known_hosts ファイルに追加します。
このファイルは /.ssh ディレクトリーにあります。

a. ssh "user"@"hmc_hostname" -i "private_keyfile" date コマンドを実行します。

b. 鍵をキャッシュに入れることを求めるプロンプトが出されたら、yes と入力します。このコマンドにより、known_hosts ファイルに項目が追加されて、今後の接続で使用されるようになります。

f) ssh "user"@"hmc_hostname" date コマンドを実行します。
パスワード・プロンプトが出されずに日付が返された場合、SSH 鍵は正常に設定されています。

HMC 用の SDK の準備

エージェント・インスタンスの初回始動の前に、HMC 用の SDK を準備する必要があります。

このタスクについて

最初のエージェント・インスタンスを始動する前に、ご使用の HMC に対応するバージョンの SDK を準備する必要があります。準備が完了した後、同じバージョンの HMC 用に作成する他の HMC Base エージェント インスタンスに対して、このタスクを繰り返す必要はありません。別のバージョンの HMC をモニターする場合は、このタスクを繰り返して SDK を再準備します。

手順

- HMC から SDK を自動的に準備するために、*agent_dir*/aix526/ph/bin ディレクトリーから **prepareSDK.sh** スクリプト・ツールを実行します。
  - 「SDK is ready for HMC」というメッセージが表示されたら、準備は完了です。
  - 「SDK is ready for HMC」というメッセージが表示されない場合、HMC 用の SDK を手動で準備します。

    HMC V8.5.0 の場合、以下のステップを実行します。

    1. ブラウザーで以下の URL を使用して、HMC から SDK を直接ダウンロードします。

       ```
       https://HMC_IP:12443/rest/api/web/sdk
       ```

       プロンプトが出されたら、hscroot アカウントのユーザー名とパスワードを入力します。SDK のファイル名は、pmc_sdk_*.zip という形式です。
    2. . SDK zip ファイルを unzip し、IBM HMC REST Web Services SDK Runtime/lib/ibm3 ディレクトリーに移動します。
    3. まだ存在していない場合は、<agent_dir>/aix526/ph/lib/*my_hmc_version* サブディレクトリーを作成します。ここで、*my_hmc_version* は HMC 環境のバージョン (例えば、8502) です。 ご使用の HMC 環境のバージョンを判別するには、次のコマンドを実行します。

       ```
       ssh hscroot@<HMC_IP> 'lshmc -v' | grep RM |
       awk -FR '{print $3}' | tr -d '.'
       ```
    4. HMC SDK の IBM HMC REST Web Services SDK Runtime/lib/ibm3 フォルダー内のすべての .jar ファイルを *agent_dir*/aix526/ph/lib/*HMC_version* ディレクトリーにコピーします。

    HMC V8.6.0 または V8.7.0 の場合、以下のステップを実行します。

    1. ブラウザーで以下の URL を使用して、HMC から SDK を直接ダウンロードします。

       ```
       https://HMC_IP:12443/rest/api/web/sdk
       ```

       プロンプトが出されたら、hscroot アカウントのユーザー名とパスワードを入力します。SDK のファイル名は、pmc-rest-sdk*.zip という形式です。
    2. SDK zip ファイルを unzip し、lib サブディレクトリーに移動します。
    3. まだ存在していない場合は、<agent_dir>/aix526/ph/lib/*my_hmc_version* サブディレクトリーを作成します。ここで、*my_hmc_version* は HMC 環境のバージョン (例えば、8602 または 87012) です。ご使用の HMC 環境のバージョンを判別するには、次のコマンドを実行します。

       ```
       ssh hscroot@<HMC_IP> 'lshmc -v' | grep RM |
       awk -FR '{print $3}' | tr -d '.'
       ```
    4. HMC SDK の lib フォルダー内のすべての .jar ファイルを *agent_dir*/aix526/ph/lib/*my_hmc_version* ディレクトリーにコピーします。

タスクの結果

HMC 用の SDK を正しく準備しました。

次のタスク

269 ページの『HMC Base モニターの構成』の指示に従って、HMC Base エージェントを構成します。

仮想入出力をモニターするための HMC コンソール・サーバーの構成

HMC Base エージェントによって仮想入出力の状況をモニターするには、その前に HMC コンソール・サーバーを構成する必要があります。

手順

以下の手順に従って、HMC Base エージェントによって仮想入出力をモニターするための前提条件として HMC コンソール・サーバーを構成します。

- HMC コンソール・サーバーおよび仮想入出力サーバーの PMC 機能を有効にします。
  a) クラシック・モードでブラウザーを使用して、HMC コンソール・サーバーにログオンします。

  ```
  https://hmc_hostname
  ```

  b)「**HMC 管理**」 >「**パフォーマンス・モニター設定の変更**」をクリックします。
  「**パフォーマンス・モニター設定の変更**」ウィンドウが表示されます。
  c)「**管理対象サーバーのパフォーマンス・モニター・データ収集**」セクションで、対応するサーバーの「**収集**」機能をオンにします。
  d) 各仮想入出力サーバーをクリックし、そのサーバーの「**パーティションの属性**」ウィンドウを表示します。
  e)「**一般**」タブで、「**パフォーマンス情報の収集を許可する**」オプションのチェック・ボックスが選択されていることを確認します。
  「**OK**」をクリックして設定を保存します。
  数分後、「**パフォーマンス・モニター**」ページに、対応するサーバーのネットワークおよびストレージ・トラフィックが表示されます。
- HMC Base エージェントの HMC ユーザーが正しい特権を備えていることを確認します。
  a) ユーザーを追加または編集する際には、そのユーザーが **hmcviewer** 役割を備えていること、およびそのユーザーの **AllSystemResource** オプションが有効になっていることを確認します。
  b)「**ユーザー・プロパティー**」ウィンドウで、「**Web を介したリモート・アクセスを許可**」オプションを有効にします。

CPU およびメモリー使用率のモニターの有効化

CPU およびメモリー使用率のデータ収集が無効になっている場合、UI で各 Power サーバーの CPU およびメモリー使用率データは表示されません。

手順

以下のいずれかの方法を使用して、CPU およびメモリー使用率のモニターを有効にします。

- 以下の HMC 管理コマンド `chlparutil` を実行して、CPU およびメモリー使用率のモニターを有効にします。

  ```
  chlparutil-r config -m <CECname> -s <the sample rate in seconds, always 60>
  ```

- HMC コンソール・サーバーで CPU およびメモリー使用率のモニターを有効にします。
  a) クラシック・モードで HMC コンソール・サーバーにログオンします。
  b) ナビゲーション・ツリーで「**サーバー**」ノードをクリックします。
  c) サーバーを選択し、「**操作**」 >「**使用状況データ**」 >「**サンプル・レートの変更 (Change the Sample rate)**」に移動します。
  d) サンプリング・レートを設定します。
  サンプリング・レートはデフォルトで無効になっています。適切な値 (例えば、30 分) を使用してレートを設定できます。

HTTP Server モニターの構成

インストールが完了すると、Monitoring Agent for HTTP Server が自動的に開始されます。データ収集を有効にするには、HTTP Server が実行されていることを確認し、HTTP Server 構成ファイルを編集して HTTP Server エージェント・データ・コレクター構成ファイルへの参照が含まれるようにします。

始める前に

HTTP Server エージェントの構成に関わるファイルは 2 つあります。これらのファイルのサンプルを表示するには、『サンプル』を参照してください。以下のファイルを見つけて確認します。

HTTP Server エージェント・データ・コレクター構成ファイル

HTTP Server エージェントをインストールすると、HTTP Server が検出され、*install_dir* /tmp/khu ディレクトリーにデータ・コレクター構成ファイルが生成されます (ここで、*install_dir* は、HTTP Server エージェントがインストールされているディレクトリーです)。

環境内に複数の HTTP Server がある場合は、HTTP Server ごとに 1 つの HTTP Server エージェント構成ファイルが生成されます。

HTTP Server エージェント構成ファイル名は 2 つの部分で構成され、そのフォーマットは以下のとおりです。

```
khu.full path of the HTTP Server configuration file name.conf
```

エージェント構成ファイル名の最初の部分は khu であり、その hu は HTTP Server エージェント・コードです。
エージェント構成ファイル名の 2 番目の部分は、HTTP Server 構成ファイルの絶対パスと名前を使用して作成され、/ が . に置き換えられます。例えば、ファイル名は以下のようになります。

Linux AIX
```
khu.usr.local.apache24.conf.httpd.conf
```

Windows
```
khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf
```

HTTP Server エージェント・データ・コレクター構成ファイルには、以下の要素が含まれています。

- HTTP Server が使用する httpd.conf ファイルのパスに関する詳細 (KhuShmemPath "/IBM/HTTPServer/conf/httpd.conf" など)。
- ロードするライブラリーの場所
- 共用メモリーに関連付けられた許可

HTTP Server 構成ファイル

各 HTTP Server には、*http_server_install_dir* /conf/httpd.conf というデフォルト名の構成ファイルがあります (ここで、*http_server_install_dir* は、HTTP Server がインストールされているディレクトリーです)。一部の環境では、このファイル名がカスタマイズされていることがあります。正確なファイル名については、HTTP Server 管理者に確認してください。

このタスクについて

以下の状況で、HTTP Server エージェントでデータ収集を有効にします。

- HTTP Server エージェントのインストール後
- HTTP Server エージェントのバージョン 1.0.0.4 にアップグレードした後。新規別名が原因で、既存の HTTP Server エージェント・ノードが Cloud APM コンソールでオフラインになります。
- バージョン 1.0.0.4 からアップグレードした後。既存の HTTP Server エージェント・ノードが Cloud APM コンソールでオフラインになることがあります。これが発生する可能性があるのは、同じようなエージェント構成ファイル名 (例えば、httpd と httpd01 など) を使用している複数の HTTP Server インスタンスが存在する場合です。

netstat コマンド行ネットワーク・ユーティリティー・ツールは、実行中の HTTP サーバーを HTTP Server エージェントが正常に検出するために必要です。

重要 : アップグレード後に生じるエージェント・ノードのオフラインに関する問題を解決するには、このタスクを実行した後に、Cloud APM コンソールで新規 HTTP Server インスタンスを追加する必要があります。

手順

1. データ収集をアクティブ化するには、HTTP Server 構成ファイルで Include ステートメントを使用してデータ・コレクター構成ファイルを参照する必要があります。以下のステートメントを HTTP Server 構成ファイルの末尾に追加します。

   ```
   Include “install_dir/tmp/khu/khu.HTTP サーバー構成ファイル名の絶対パス.conf”
   ```

   例えば、次の場合があります。

   Linux AIX IBM HTTP Server が /opt/IBM/HTTPServer ディレクトリーにインストールされていて、データ・コレクター構成ファイルが以下のディレクトリーにある場合を考えます。

   ```
   /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
   ```

   以下のステートメントを /opt/IBM/HTTPServer/conf/httpd.conf HTTP Server 構成ファイルに追加します。

   ```
   Include “/opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf”
   ```

   Windows IBM® HTTP Server が C:¥ProgramFiles¥IBM¥HTTPServer ディレクトリーにインストールされていて、データ・コレクター構成ファイルが以下のディレクトリーにある場合を考えます。

   ```
   C:¥IBM¥APM¥tmp¥khu¥khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf
   ```

   以下のステートメントを C:¥Program Files¥IBM¥HTTPServer¥conf¥httpd.conf HTTP Server 構成ファイルに追加します。

   ```
   Include “C:¥IBM¥APM¥tmp¥khu¥khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf”
   ```

2. 以下のディレクトリーに移動します。

   ```
   HTTP_server_installation_directory/bin
   ```

3. HTTP Server を再始動します。例えば、次のようにします。

   Linux AIX

   ```
   ./apachectl -k stop
   ./apachectl -k start
   ```

   Windows

   ```
   httpd.exe -k stop
   httpd.exe -k start
   ```

タスクの結果

エージェントが正常に構成されました。

次のタスク

これで、HTTP Server エージェントのデータが Cloud APM コンソールで表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

注 : HTTP Server でトラフィックがない場合は、Cloud APM コンソールでデータは表示されません。

IBM HTTP Server 応答時間モジュール

IBM HTTP Server 応答時間モジュールと連携する Response Time Monitoring エージェントをインストールすると、HTTP 要求および HTTPS 要求のすべてのポートがモニターされます。

IBM HTTP Server 応答時間モジュール

IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントの一部です。HTTP Server エージェントが、AIX、Linux または Windows 上の Apache HTTP Server または IBM HTTP Server 用の Response Time Monitoring エージェントよりも前またはこれと同時にインストールおよび構成された場合、IBM HTTP Server 応答時間モジュールは自動的に有効になります。IBM HTTP Server 応答時間モジュールの機能の説明については、695 ページの『IBM HTTP Server 応答時間モジュールの構成』を参照してください。

データ・コレクター構成ファイル

HTTP Server エージェントをインストールすると、HTTP Server が検出され、*install_dir* /tmp/khu ディレクトリーにデータ・コレクター構成ファイルが生成されます (ここで、*install_dir* は、HTTP Server エージェントがインストールされているディレクトリーです)。

Apache HTTP Server の場合、データ・コレクター構成ファイルは以下のとおりです。
`khu.usr.local.apache24.conf.httpd.conf`
IBM HTTP Server の場合、データ・コレクター構成ファイルは以下のとおりです。
`khu.opt.IBM.HTTPServer.conf.httpd.conf`

プラグイン

HTTP Server エージェントは、以下の 2 つのプラグインで構成されています。

1. `khu_module` - これは HTTP Server エージェントです。このプラグインは、HTTP Server エージェント関連付けられたすべてのダッシュボードを処理します。詳しくは、「HTTP Server エージェント Reference」を参照してください。
2. `wrt_module` - これは IBM HTTP Server 応答時間モジュールです。

これらの 2 つのプラグインは、データ・コレクター構成ファイルに次のように示されます。

```
LoadModule khu_module

LoadModule wrt_module
```

データ収集のアクティブ化

データ収集をアクティブ化するには、HTTP Server 構成ファイルで `Include` ステートメントを使用してデータ・コレクター構成ファイルを参照する必要があります。以下のステートメントを HTTP Server 構成ファイルの末尾に追加します。

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
```

詳しくは、275 ページの『HTTP Server モニターの構成』を参照してください。

データ収集がアクティブ化されると、エンド・ユーザー・ダッシュボードにデータが取り込まれます。

HTTP Server エージェントのコード・サンプル

HTTP Server エージェントの構成に関わるファイルは 2 つあります。HTTP Server エージェント・データ・コレクター構成ファイルと HTTP サーバー構成ファイルです。別名の仕組みの説明に役立つように、インスタンス別名マッピング・ファイルのサンプルも用意されています。

HTTP Server エージェント・データ・コレクター・ファイルのサンプル

IBM HTTP Server バージョン 8 以降 (64 ビット) の場合、HTTP Server エージェント・データ・コレクター構成ファイルには、以下の情報が含まれています。

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache22dc_64.so"
```

```
<IfModule mod_khu.c>
   KhuShmemPerm 660
   KhuShmemPath "/opt/IBM/IHS/conf/httpd.conf"
   KhuCpsPath "/tmp/ihs/tmp/khu/khu_cps.properties"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  #Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap22_64.so
WrtOriginID HU:tivvm09_httpd:HUS
```

IBM HTTP Server バージョン 7 (32 ビット) の場合、構成ファイルには以下の情報が含まれています。

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache22dc_32.so"

<IfModule mod_khu.c>
 KhuShmemPerm 660
 KhuShmemPath "/opt/IBM/HTTPServer/conf/httpd.conf"
 KhuCpsPath "/tmp/ihs/tmp/khu/khu_cps.properties"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  #Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap22.so
WrtOriginID HU:linux_httpd:HUS
```

Apache バージョン 2.4 (64 ビット) の場合、HTTP Server エージェント構成ファイルには、以下の情報が含まれています。

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache24dc_64.so"

<IfModule mod_khu.c>
 KhuShmemPerm 660
 KhuShmemPath "/usr/local/apache24/conf/httpd.conf"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap24_64.so
WrtOriginID HU:linux-tzsi_httpd:HUS
```

インスタンス別名マッピング・ファイルのサンプル

```
# Monitoring Agent for HTTP Server instance alias mapping
# INSTANCE: auto discovered by agent. Please do NOT modify.
# ALIAS: alias name for the instance. The name will be displayed in APM UI dashboard. It
must be unique
# among all instances and it must be less than 10 characters and consist of only
alphanumeric characters.
#
INSTANCE.1=/usr/local/apache24/conf/httpd.conf
```

```
ALIAS.1=httpd

INSTANCE.1=/usr/local/apache24/conf/admin.conf
ALIAS.1=admin
```

IBM Cloud Monitoring の構成

Monitoring Agent for IBM Cloud は、IBM Cloud (SoftLayer) アカウントから仮想マシンのインベントリーおよびメトリックを収集します。IBM Cloud で構成して実行している仮想デバイスの数を追跡するには、IBM Cloud エージェントを使用します。詳細なダッシュボード・ページで各仮想デバイスに割り振られているリソースを確認できます。このページには、デバイスが置かれているデータ・センター、オペレーティング・システム、およびその月の予想されるパブリック・ネットワーク帯域幅などの情報も表示されます。

始める前に

- 279 ページの『IBM Cloud Monitoring の構成』のトピック全体を読んで、構成を完了するために何が必要かを判断します。
- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、IBM Cloud エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the IBM Cloud エージェント を参照してください。
- 次の情報が使用可能であることを確認します。
  - 少なくとも監査員権限を持つユーザーのユーザー名。
  - その関連するユーザーの IBM Cloud の API キー。

このタスクについて

IBM Cloud エージェントは複数インスタンス・エージェントであり、サブノード・エージェントでもあります。エージェント・インスタンスを構成した後で、各エージェント・インスタンスを手動で開始する必要があります。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用して Windows システムでエージェントを構成します。
   - 280 ページの『Windows システムでのエージェントの構成』.
   - 281 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. 応答を要求するプロンプトを出すスクリプト、またはサイレント応答ファイルを使用して、Linux システムでエージェントを構成します。
   - 280 ページの『プロンプトへの応答によるエージェントの構成』.
   - 281 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、IBM Cloud エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for IBM Cloud**」テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは使用できません。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成...**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、ハイフンマイナス文字のみを使用してください。例: `icloud-inst`
4. エージェント・インスタンス名ウィンドウで「**次へ**」をクリックします。
5. 「**新規**」をクリックし、IBM Cloud SoftLayer のユーザー名と API キー設定を入力し、「**次へ**」をクリックします。
6. 「**OK**」をクリックして、構成を完了します。
7. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

IBM Cloud エージェントのインストール後、エージェントを開始する前に構成する必要があります。IBM Cloud エージェントがローカル Linux コンピューターにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで IBM Cloud エージェントを構成します。

1. 次のコマンドを実行します。

   ```
   install_dir/bin/ibm_cloud-agent.sh config instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パス、*instance_name* はエージェント・インスタンスに指定する名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/ibm_cloud-agent.sh config icloud-inst
   ```
2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、282 ページの『IBM Cloud エージェントの構成パラメーター』を参照してください。
3. 以下のコマンドを実行して、エージェントを開始します。

```
install_dir/bin/ibm_cloud-agent.sh start instance_name
```

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

例

```
/opt/ibm/apm/agent/bin/ibm_cloud-agent.sh start icloud-inst
```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- 以下の手順を実行して、IBM Cloud エージェントをサイレント・モードで構成します。
  a) テキスト・エディターで、以下のいずれかのパスにある ibm_cloud_silent_config.txt ファイルを開きます。
  - Linux *install_dir*/samples/ibm_cloud_silent_config.txt

    例: /opt/ibm/apm/agent/samples/ibm_cloud_silent_config.txt
  - Windows *install_dir*¥samples¥ibm_cloud_silent_config.txt

    例: C:¥IBM¥APM¥samples¥ibm_cloud_silent_config.txt

  ここで、*install_dir* はエージェントのインストール先パスです。
  b) ibm_cloud_silent_config.txt ファイルで、すべての必須パラメーターの値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

  各構成パラメーターの説明については、282 ページの『IBM Cloud エージェントの構成パラメーター』を参照してください。
  c) ibm_cloud_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
  - Linux *install_dir*/bin/ibm_cloud-agent.sh config **instance_name** *install_dir*/samples/ibm_cloud_silent_config.txt

    例: **/opt/ibm/apm/agent/bin/ibm_cloud-agent.sh config icloud-inst /opt/ibm/apm/agent/samples/ibm_cloud_silent_config.txt**
  - Windows *install_dir*¥bin¥ibm_cloud-agent.bat config **instance_name** *install_dir*¥samples¥ibm_cloud_silent_config.txt

    例: **C:¥IBM¥APM¥bin¥ibm_cloud-agent.bat config icloud-inst C:¥IBM¥APM¥samples¥ibm_cloud_silent_config.txt**

  ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

  重要: サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/ibm_cloud-agent.sh start **instance_name**

  例: **/opt/ibm/apm/agent/bin/ibm_cloud-agent.sh start icloud-inst**

- Windows *install_dir*¥bin¥ibm_cloud-agent.bat start **instance_name**

  例: **C:¥IBM¥APM¥bin¥ibm_cloud-agent.bat start icloud-inst**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

IBM Cloud エージェントの構成パラメーター

IBM Cloud エージェントの構成パラメーターを表に表示します。

1. IBM Cloud の構成 - IBM Cloud インスタンスをリモートでモニターするための 設定。構成対象の API キーのインスタンスが自動的にディスカバーされます。

表 *23. IBM Cloud* の構成

パラメーター名	説明	サイレント構成ファイルのパラメーター名
ユーザー名	IBM Cloud API からメトリックを取得するために使用される IBM SoftLayer アカウントのユーザー名。	**KFS_USERNAME**
API キー	認証の実行に必要なユーザー固有の API キー。API キーは、IBM SoftLayer カスタマー・ポータルで生成され、このポータルから取得できます。	**KFS_API_KEY_PASSWORD**

IBM Integration Bus のモニターの構成

IBM Integration Bus エージェントは複数インスタンス・エージェントです。最初のエージェント・インスタンスを作成し、手動で開始する必要があります。

始める前に

- 以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。
- ご使用の環境で、IBM Integration Bus エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Detailed System Requirements Report for the IBM Integration Bus エージェント を参照してください。

このタスクについて

以下の手順は、IBM Integration Bus エージェントを構成するためのロードマップであり、必須のステップとオプションのステップの両方が含まれています。ニーズに従って、必要なステップを実行します。

手順

1. IBM Integration Bus エージェントの開始および停止に使用するユーザー ID が、**mqm** ユーザー・グループおよび **mqbrkrs** ユーザー・グループに属していることを確認します。
2. Windows

IBM MQ (WebSphere MQ) が Windows システムにインストールされている場合、IBM MQ (WebSphere MQ) ライブラリー・パスを **PATH** 環境変数に追加します。それにより、IBM Integration Bus エージェントが開始するために必要な IBM MQ (WebSphere MQ) ライブラリーをロードできます。

a) IBM MQ (WebSphere MQ) ライブラリー・パスを **PATH** 環境変数の先頭に追加します。

例えば、IBM MQ (WebSphere MQ) のインストール・パスが `C:¥IBM¥WMQ75` の場合は、`C:¥IBM¥WMQ75¥bin` を Windows システムの **PATH** 環境変数の先頭に追加します。

b) Windows システムを再始動して、変更を有効にします。

3. 以下の構成パラメーターを指定して、IBM Integration Bus エージェントを構成します。エージェントに指定できるいくつかのオプションの構成パラメーターもあります。説明については、283 ページの『IBM Integration Bus エージェントの構成』を参照してください。
   - エージェント ID
   - モニター対象の統合ノード (ブローカー) のインストール・ディレクトリー
   - IBM MQ (WebSphere MQ) の 64 ビットのライブラリー・パス
4. IBM Integration Bus を構成して、モニターするデータを使用可能にします。287 ページの『データ使用可能化のための IBM Integration Bus の構成』を参照してください。
5. 統合ノード (ブローカー) に対するスナップショット・データ収集を有効にした場合は、IBM Integration Bus エージェントがスナップショット・データを格納しないように構成してください。説明については、294 ページの『エージェントに対するスナップショット・データ収集の無効化』を参照してください。
6. オプション: IBM Integration Bus エージェントを構成してトランザクション・トラッキングを有効にするには、「**エージェント構成**」ページを使用します。説明については、295 ページの『IBM Integration Bus エージェントのトランザクション・トラッキングの構成』を参照してください。
7. オプション: トランザクション・トラッキング機能が必要なくなった場合、または IBM Integration Bus エージェントをアンインストールする場合は、IBM Integration Bus のトランザクション・トラッキングを無効にし、エージェント提供ユーザー出口を削除します。説明については、293 ページの『トランザクション・トラッキングの無効化』および 297 ページの『KQIUserExit ユーザー出口の削除』を参照してください。

IBM Integration Bus エージェントの構成

IBM Integration Bus エージェントで IBM Integration Bus 環境のモニターを開始するには、事前にこのエージェントにインスタンス名を割り当て、エージェントを構成する必要があります。

始める前に

- エージェントの開始および停止に使用するユーザー ID が **mqm** および **mqbrkrs** ユーザー・グループに属していることを確認します。
- Windows IBM MQ (WebSphere MQ) が Windows システムにインストールされている場合、IBM MQ (WebSphere MQ) ライブラリー・パスを **PATH** 環境変数に追加します。それにより、IBM Integration Bus エージェントが開始するために必要な IBM MQ (WebSphere MQ) ライブラリーをロードできます。

  1. IBM MQ (WebSphere MQ) ライブラリー・パスを **PATH** 環境変数の先頭に追加します。

     例えば、IBM MQ (WebSphere MQ) のインストール・パスが `C:¥IBM¥WMQ75` の場合は、`C:¥IBM¥WMQ75¥bin` を Windows システムの **PATH** 環境変数の先頭に追加します。

  2. Windows システムを再始動して、変更を有効にします。
- エージェント構成時に、ご使用の環境に従って、以下の情報を指定する必要が生じることがあります。指定する適切な構成値が分からない場合は、IBM MQ (WebSphere MQ) および IBM Integration Bus の管理者から情報を収集します。
  - IBM MQ (WebSphere MQ) が IBM Integration Bus エージェントと同じシステムにインストールされている場合は、IBM MQ (WebSphere MQ) の 64 ビット・ライブラリー・パスを指定する必要があります。
  - IBM Integration Bus V10 または IBM App Connect Enterprise V11 の統合ノードをモニターするように IBM Integration Bus エージェントを構成する場合は、IBM Integration Bus V10 または IBM App Connect Enterprise V11 のインストール・ディレクトリーを指定する必要があります。

– IBM Integration Bus エージェントが同じシステム上のすべてのものではなく特定の統合ノード (ブローカー) をモニターするようにする場合は、各統合ノード (ブローカー) の名前とインストール・パスを指定する必要があります。

このタスクについて

IBM Integration Bus エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

UNIX または Linux システムでは、対話式または非対話式でエージェントを構成することを選択できます。Windows システムでは、非対話式でのみ、エージェントを構成できます。

- 対話式でエージェントを構成するには、構成スクリプトを実行してプロンプトに応答します。284 ページの『対話式の構成』を参照してください。
- 対話操作なしでエージェントを構成するには、サイレント応答ファイルを編集してから構成スクリプトを実行します。285 ページの『サイレント構成』を参照してください。

重要 : Cloud APM で提供される IBM Integration Bus エージェントと同じシステムに、ITCAM for Applications 製品の 1 つとして提供される ITCAM Agent for WebSphere Message Broker をインストールした場合には、これらをシステム上の同じ統合ノード (ブローカー) をモニターするために使用しないでください。

対話式の構成

手順

スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

1. 以下のコマンドを入力します。

   ```
   install_dir/bin/iib-agent.sh config instance_name
   ```

   ここで、*instance_name* は、エージェント・インスタンスに指定する名前です。

   重要 : 対話式の構成は、Windows システムではサポートされていません。
2. IBM Integration Bus エージェントを構成することを確認した後に、一般エージェント設定の構成値を指定します。
   a) **「Agent Id」** パラメーターの入力を求めるプロンプトが出されたら、最大 8 文字の長さの固有の英数字ストリングを指定します。

      要確認 : エージェント ID の最大長は IBM Integration Bus エージェントのバージョン 7.3.0.1 からは 8 文字です。それ以前のバージョンでは、エージェント ID の最大長は 4 文字です。

      管理対象システム名には、指定したエージェント ID が組み込まれます。例えば、*monitoredbrokername*:*agentID*:KQIB のようになります。ここで、*monitoredbrokername* は、モニター対象統合ノード (ブローカー) の名前です。

   b) IBM Integration Bus V10 または IBM App Connect Enterprise V11 の統合ノードをモニターする場合に、**IIB version 10 or ACE version 11** Install Directory パラメーターの入力を求めるプロンプトが出されたら、IBM Integration Bus V10 または IBM App Connect Enterprise V11 のインストール・ディレクトリーを指定します (例: /opt/ibm/mqsi/ace-11.0.0.3)。IBM Integration Bus V10 および IBM App Connect Enterprise V11 をモニターしない場合は、Enter を押してデフォルトを受け入れます。

      要確認 : IIB version 10 or ACE version 11 Install Directory パラメーターに指定できるインストール・ディレクトリーは 1 つだけです。IBM Integration Bus V10 または IBM App Connect Enterprise V11 を異なる複数のディレクトリーにインストール済みで、それらをすべてモニターしたい場合は、複数のエージェント・インスタンスを作成し、各エージェント・インスタンスに対して IBM Integration Bus V10 または IBM App Connect Enterprise V11 のインストール・ディレクトリーを 1 つ指定します。

3. オプション: **Monitored Broker Settings** セクションを使用して、このエージェントを使用して特定のいくつかの統合ノード (ブローカー) のみをモニターするかどうかを指定します。

   デフォルトでは、IBM Integration Bus エージェントと同じホスト・システム上で実行されているすべての統合ノード (ブローカー) がモニターされます (セルフ・ディスカバリーで判別)。エージェントでいくつかの特定の統合ノード (ブローカー) をモニターする場合は、モニターする統合ノード (ブローカー) の名前を指定し、**Monitored Broker Settings** セクションで **Collect Node Data** 設定を No (デフォルト値) に設定します。**Monitored Broker Settings** セクションは複数存在することが可能です。各セクションは 1 つの統合ノード (ブローカー) のモニター設定を制御します。

   ヒント : 複数の **Monitored Broker Settings** セクションを指定できます。**Monitored Broker Settings** セクションを編集する場合、以下のオプションを指定できます。

   - Add: 他の統合ノード (ブローカー) 用に構成するために **Monitored Broker Settings** セクションを作成します。
   - Edit: 現在の **Monitored Broker Settings** セクションの設定を変更します。
   - Del: 現在の **Monitored Broker Settings** セクションを削除します。
   - Next: 次の **Monitored Broker Settings** セクションに移動します。
   - Exit: **Monitored Broker Settings** の構成を終了します。
4. IBM MQ (WebSphere MQ) が同じシステムにインストールされていることを確認した場合、**「WebSphere MQ 64-bit library path」**パラメーターを求めるプロンプトが出されます。Enter キーを押してデフォルト値を受け入れます。このデフォルト値は、エージェントによって自動的にディスカバーされた IBM MQ (WebSphere MQ) の 64 ビット・ライブラリー・パスです。デフォルト値が表示されない場合は、必ず IBM MQ (WebSphere MQ) の 64 ビット版ライブラリーのパスを指定してから次のステップに進んでください。例えば、`/opt/mqm8/lib64` です。

   要確認 : ご使用の統合ノード (ブローカー) が複数の異なるバージョンのキュー・マネージャーを使用する場合、このパラメーターには、最新バージョンの IBM MQ (WebSphere MQ) の 64 ビットのライブラリー・パスを指定してください。
5. 構成の完了後に、以下のコマンドを入力してエージェントを開始します。

   ```
   install_dir/bin/iib-agent.sh start instance_name
   ```

サイレント構成

手順

サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。

1. テキスト・エディターで以下に示すエージェントのサイレント 応答ファイルを開きます。

   - Linux AIX *install_dir*/samples/iib_silent_config.txt
   - Windows *install_dir*¥tmaitm6_x64¥samples¥qi_silent_config.txt

   ここで、*install_dir* は IBM Integration Bus エージェント のインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは次のとおりです。

   - Linux AIX /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM
2. **「agentId」** パラメーターに、エージェントの短い ID として、最大 8 文字の長さの固有の英数字ストリングを指定します。

   要確認 : エージェント ID の最大長は IBM Integration Bus エージェントのバージョン 7.3.0.1 からは 8 文字です。それ以前のバージョンでは、エージェント ID の最大長は 4 文字です。

   管理対象システム名には、指定したエージェント ID が組み込まれます。例えば、*monitoredbrokername*:*agentID*:KQIB のようになります。ここで、*monitoredbrokername* は、モニター対象統合ノード (ブローカー) の名前です。

3. IBM Integration Bus V10 または IBM App Connect Enterprise V11 の統合ノードをモニターする場合は、**defaultWMBInstallDirectory** パラメーターに IBM Integration Bus V10 または IBM App Connect Enterprise V11 のインストール・ディレクトリーを指定します。例えば、Windows システムの場合は `C:¥Program Files¥IBM¥ACE¥11.0.0.3¥`、Linux システムの場合は `/opt/ibm/mqsi/ace-11.0.0.3` のようにします。IBM Integration Bus V10 および IBM App Connect Enterprise V11 をモニターしない場合は、このパラメーターは必要ありません。これは、IBM Integration Bus エージェントはそれより前のバージョンの統合ノード (ブローカー) を自動的にディスカバーできるからです。

   要確認 : defaultWMBInstallDirectory パラメーターには、インストール・ディレクトリーは 1 つしか指定できません。IBM Integration Bus V10 または IBM App Connect Enterprise V11 を異なる複数のディレクトリーにインストール済みで、それらをすべてモニターしたい場合は、複数のエージェント・インスタンスを作成し、各エージェント・インスタンスに対して IBM Integration Bus V10 または IBM App Connect Enterprise V11 のインストール・ディレクトリーを 1 つ指定します。
4. オプション: このエージェントを使用して特定のいくつかの統合ノード (ブローカー) のみをモニターするかどうかを指定します。

   デフォルトでは、IBM Integration Bus エージェントと同じホスト・システム上で実行されているすべての統合ノード (ブローカー) がモニターされます (セルフ・ディスカバリーで判別)。特定の統合ノード (ブローカー) をモニターするには、モニターする統合ノード (ブローカー) ごとに、**collectNodeData** および **WMBInstallDirectory** パラメーターを設定します。

   collectNodeData
   モニター対象統合ノード (ブローカー) のノード定義データを収集するかどうかを指定します。 構文は、`collectNodeData.`*`brkr_name`*`=NO|YES` です。ここで、*brkr_name* は統合ノード (ブローカー) の名前です。

   デフォルト値は NO です。Cloud APM コンソールではノード定義データはサポートされないため、デフォルト値を使用することをお勧めします。

   WMBInstallDirectory
   モニターする統合ノード (ブローカー) のインストール・ディレクトリー。構文は、`WMBInstallDirectory.`*`brkr_name=broker_install_dir`* です。ここで、*broker_install_dir* はモニターする統合ノード (ブローカー) のインストール・ディレクトリーです。

   要確認 : バージョン 10 の統合ノードの場合、**WMBInstallDirectory** パラメーターにより、前のステップで設定した **defaultWMBInstallDirectory** パラメーターをオーバーライドできます。

   例えば名前が BK1 と BK2 の 2 つの統合ノード (ブローカー) のみをモニターするには、以下のようにパラメーターを設定します。

```
collectNodeData.BK1=NO
collectNodeData.BK2=NO
WMBInstallDirectory.BK1=BK1_install_dir
WMBInstallDirectory.BK2=BK2_install_dir
```

5. IBM Integration Bus V10 より前のブローカーをモニターするには、**WMQLIBPATH** パラメーターに IBM MQ (WebSphere MQ) の 64 ビット版ライブラリーのパスを指定します。例えば、`C:¥Program Files¥IBM¥WebSphere MQ¥bin64` (Windows システムの場合) や `/opt/mqm8/lib64` (Linux システムの場合) などです。

   要確認 : ご使用の統合ノード (ブローカー) が複数の異なるバージョンのキュー・マネージャーを使用する場合、このパラメーターには、最新バージョンの IBM MQ (WebSphere MQ) の 64 ビットのライブラリー・パスを指定してください。
6. エージェントのサイレント応答ファイルを保存して閉じ、次のコマンドを入力します。
   - Linux AIX *install_dir*`/bin/iib-agent.sh config` *instance_name path_to_responsefile*
   - Windows *install_dir*`¥BIN¥iib-agent.bat config "`*instance_name path_to_responsefile*`"`

   ここで、*instance_name* は構成するインスタンスの名前であり、*path_to_responsefile* はサイレント応答ファイルの絶対パスです。

警告 : Windows システムでは、サイレント応答ファイルの絶対パスを囲む二重引用符 ("") を含めないでください。構成エラーの原因となるためです。

7. 構成の完了後に、以下のコマンドを入力してエージェントを開始します。

- Linux AIX

```
install_dir/bin/iib-agent.sh start instance_name
```

- Windows

```
install_dir¥bin¥iib-agent.bat start instance_name
```

タスクの結果

これで、Cloud APM コンソールにログインし、アプリケーション・エディターを使用して IBM Integration Bus エージェント・インスタンスを Application Performance ダッシュボードに追加できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

要確認 : モニター対象統合ノード (ブローカー) を更新またはマイグレーションするときは必ず、統合ノード (ブローカー) のアップグレードまたはマイグレーション 後に IBM Integration Bus エージェントを再始動する必要があります。

次のタスク

次のステップでは、データ使用可能化のために IBM Integration Bus を構成します。以下のデータが Application Performance ダッシュボードで使用可能になるのは、IBM Integration Bus でそれらを使用可能にした後のみです。

- アーカイブ・アカウンティングおよび統計
- JVM リソース統計
- トランザクション・トラッキング

説明については、287 ページの『データ使用可能化のための IBM Integration Bus の構成』を参照してください。

データ使用可能化のための IBM Integration Bus の構成

一部のデータを Cloud APM コンソールで使用可能にするために、IBM Integration Bus を構成して必要なデータ収集を有効にする必要があります。

始める前に

IBM Integration Bus エージェントが構成されていることを確認します。

要確認 : トランザクション・トラッキングを有効にするには、統合ノード (ブローカー) を再始動する必要があります。

このタスクについて

アーカイブ統計およびリソース統計を IBM Integration Bus エージェントによってモニターできるのは、統合ノード (ブローカー) でデータ収集が有効になった後のみです。同様に、ミドルウェアおよびトポロジー・ダッシュボードでトランザクション・トラッキングを確認する場合は、IBM Integration Bus エージェントでトランザクション・トラッキングを有効にする前に、統合ノード (ブローカー) 内でトランザクション・トラッキングを有効にする必要があります。

IBM Integration Bus エージェントでモニターするデータのタイプを決定し、ニーズに応じて以下のステップを実行します。

統合ノードによって所有される統合サーバーには、統合ノード・ディレクトリーのサブディレクトリーに保管されている統合サーバーごとに、デフォルトの server.conf.yaml 構成ファイルがあります。node.conf.yaml ファイル内にある統合ノード に設定したプロパティーはすべて、統合ノードが所有する統合サーバーによって継承されます。それでも、server.conf.yaml ファイルで変更することにより、統合サーバーのプロパティーを 変更できます。(詳細については、IBM App Connect Enterprise 資料の Configuring an integration node by modifying the node.conf.yaml file を参照してください。)

手順

- 統合ノード (ブローカー) のアーカイブ統計データ収集を有効にする場合は、288 ページの『アーカイブ・アカウンティングおよび統計データ収集の有効化』を参照してください。
- 統合ノード (ブローカー) のリソース統計データを有効にする場合は、291 ページの『JVM リソース統計の有効化』を参照してください。
- 統合ノード (ブローカー) 内のメッセージ・フローのトランザクション・トラッキングを有効にする場合は、292 ページの『トランザクション・トラッキングの有効化』を参照してください。
- トランザクション・トラッキング・データが必要なくなった場合、有効にされた統合ノード (ブローカー) のトランザクション・トラッキングを必ず無効にしてください。293 ページの『トランザクション・トラッキングの無効化』を参照してください。

アーカイブ・アカウンティングおよび統計データ収集の有効化

このタスクについて

統合ノード (ブローカー) に属しているメッセージ・フローのアーカイブ・アカウンティングおよび統計収集を有効にするには、統合ノード (ブローカー) インストール・ディレクトリーの bin ディレクトリーから **mqsichangeflowstats** コマンドを発行します。

要確認 : データのモニタリングの要件に従って **mqsichangeflowstats** コマンドを統合ノード (ブローカー) に対して発行します。必要な統計のみを有効にすることをお勧めします。大量のメッセージ・フローを使用している場合、データと処理が大量になる可能性があります。**mqsichangeflowstats** コマンドについて詳しくは、IBM Integration Bus の資料を参照してください。

重要 : IBM Cloud Application Performance Management では、設定された 20 秒のスナップショット間隔で必要となるデータ量と処理量の理由から、スナップショットのアカウンティング・データと統計データをサポートしません。アーカイブ・データは、スナップショット・データと完全に同じ属性を提供するため、IBM Cloud Application Performance Management により提供される通常の実動モニタリングにより適しています。統合ノード (ブローカー) に対するスナップショット・データ収集を有効にした場合は、必ず IBM Integration Bus エージェントがスナップショット・データを格納しないように構成してください。説明については、294 ページの『エージェントに対するスナップショット・データ収集の無効化』を参照してください。

手順

- メッセージ・フローのほとんどのデータを取得するには、以下のコマンドを発行します。このコマンドは、ノードあたりの端末ごとの起動カウントを提供する最も詳細な端末統計を有効にしないため、推奨されます。端末レベルは大量のストレージを消費します。

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n basic -o xml
```

- ACE バージョン 11 で、メッセージ・フローのほとんどのデータを取得するには、node.conf.yaml ファイルまたは server.conf.yaml ファイルを以下のように変更します。これらのプロパティーでは、ノードあたりの端末ごとの起動カウントを提供する最も詳細な端末統計が有効にならないため、これらのプロパティーを使用することをお勧めします。端末レベルは大量のストレージを消費します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
```

```
                              # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               #csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'     # choose 1 of : active|inactive,
                             # default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60         # Sets the interval in minutes at which
                              #archive statistics are published
    nodeDataLevel: 'basic'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml'  # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
```

注 : この設定を無効にするには、**archivalOn: ' active'**、**nodeDataLevel: ' basic'**、および **outputFormat: ' xml'** の各行をコメント化します。

- IBM Integration Bus エージェントでサポートされるすべてのデータを取得するには、以下のコマンドを発行します。

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n advanced -o xml
```

- ACE バージョン 11 で、IBM Integration Bus エージェントでサポートされるすべてのデータを取得するには、node.conf.yaml ファイルまたは server.conf.yaml ファイルを以下のように変更します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60       # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none' # choose 1 of : none|basic
```

注 : この設定を無効にするには、**archivalOn: ' active'**、**nodeDataLevel: ' advanced'**、および **outputFormat: ' xml'** の各行をコメント化します。

- データ量を削減するが、追加の詳細を必要とせずにすべてのメッセージ・フローをある程度モニターするには、以下のコマンドを発行します。

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n none -o xml
```

- ACE バージョン 11 で、データ量は削減するが、追加の詳細を収集せずにすべてのメッセージ・フローを無駄なくモニターするには、node.conf.yaml ファイルまたは server.conf.yaml ファイルを以下のように変更します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  #set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
```

```
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60       # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'none'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none' # choose 1 of : none|basic
```

注 : この設定を無効にするには、**archivalOn: ' active'**、**nodeDataLevel: ' none'**、および **outputFormat: ' xml'** の各行をコメント化します。

- 大量のメッセージ・フローがあり、データ量を削減する必要がある場合、上記コマンドの `-g` または `-j` オプションを置き換えることにより、どのメッセージ・フローをモニターするかを指定できます。
  - 特定の統合サーバー (実行グループ) を有効にすることを指定するには、`-g` を `-e` *IntegrationServerName* で置き換えます。
  - 特定のメッセージ・フローを有効にすることを指定するには、`-j` を `-f` *MessageFlowName* で置き換えます。
  - メッセージ・フローをアプリケーションにグループ化している場合、特定のアプリケーションを有効にすることを指定するには、`-k` *ApplicationName* を `-j` オプションに追加します。
- IBM Integration Bus エージェントは、5 分間隔でアーカイブ・アカウンティングおよび統計データを収集します。統合ノード (ブローカー) がアーカイブ・アカウンティングおよび統計データを生成する間隔を同じ間隔に設定するには、統合ノード (ブローカー) を停止した状態で以下のコマンドを発行してから、統合ノード (ブローカー) を再始動します。

```
mqsichangebroker BrokerName -v 5
```

- ACE バージョン 11 では、IBM Integration Bus エージェントは 5 分間隔でアーカイブ・アカウンティングおよび統計データを収集します。統合ノード (ブローカー) がアーカイブ・アカウンティングおよび統計データを生成する間隔を同じ間隔に設定するには、`node.conf.yaml` ファイルまたは `server.conf.yaml` ファイルを以下のように変更します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'     # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'none'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none'       # choose 1 of : none|basic
```

タスクの結果

IBM Integration Bus エージェントが構成されて開始されると、以下のグループ・ウィジェットにメッセージ・フローのアカウンティングおよび統計データが表示されます。

- メッセージ・フロー・ダッシュボード
  - コミットおよびバックアウト
  - CPU マイクロ秒

  - 経過マイクロ秒
  - 入力バイトの比率
  - 入力メッセージの比率
  - 入力メッセージ・サイズ
  - 入力メッセージ待機 CPU マイクロ秒
  - 入力メッセージ待機経過マイクロ秒
  - メッセージ・フロー・エラー
  - メッセージ処理ノード統計
- 処理ノード・ダッシュボード
  - CPU マイクロ秒
  - 経過マイクロ秒
  - 呼び出し数
  - 処理ノード状況
  - 端末統計

JVM リソース統計の有効化

このタスクについて

統合ノード (ブローカー) に属している統合サーバーの JVM リソース統計を有効にするには、統合ノード (ブローカー) インストール・ディレクトリーの bin ディレクトリーから **mqsichangeresourcestats** コマンドを発行します。

要確認 : JVM リソース統計はオプションと見なされます。これは、エージェントがこのデータを 20 秒ごとに処理する高いコストに対してデータのほんの一部の属性しか表示されないためです。JVM リソースの統計データが必要かどうかを慎重に検討してください。

手順

- 統合ノード (ブローカー) 内のすべての統合サーバーにまたがって統計を有効にするには、以下のコマンドを発行します。

```
mqsichangeresourcestats BrokerName -c active
```

- ACE バージョン 11 で、統合ノード (ブローカー) 内のすべての統合サーバーにまたがって統計を有効にするには、node.conf.yaml ファイルを以下のように変更します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    threadDataLevel: 'basic'      # choose 1 of : none|basic
   Resource:
     reportingOn: true       # choose 1 of : true|false, default false
......
```

注 : この設定を無効にするには、**reportingOn: true** をコメント化します。

- 統合ノード (ブローカー) 内の特定の統合サーバーの統計を有効にするには、以下のコマンドを発行します。

```
mqsichangeresourcestats BrokerName -e IntegrationServerName -c active
```

- ACE バージョン 11 で、統合ノード (ブローカー) 内の特定の統合サーバーの統計を有効にするには、server.conf.yaml ファイルを以下のように変更します。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    threadDataLevel: 'basic'   # choose 1 of : none|basic
  Resource:
    reportingOn: true        # choose 1 of : true|false, default false
```

注 : この設定を無効にするには、**reportingOn: true** をコメント化します。

タスクの結果

以下のグループ・ウィジェットに JVM リソース統計データが表示されます。

- ガーベッジ・コレクション数
- ガーベッジ・コレクションの期間
- JVM 非ヒープ・メモリー
- JVM ヒープ・メモリー

トランザクション・トラッキングの有効化

始める前に

1. IBM Integration Bus エージェントがインストールされていることを確認します。IBM Integration Bus でトランザクション・トラッキングを有効にするために、KQIUserExit という名前のユーザー出口が用意されています。
2. 統合ノード (ブローカー) を開始するユーザーが KQI ユーザー出口モジュールのディレクトリーに確実にアクセスできるようにしてください。つまり、統合ノード (ブローカー) の開始に使用されるユーザー ID を、IBM Integration Bus エージェントをインストールしたグループに追加してください。

このタスクについて

KQIUserExit ユーザー出口を統合ノード (ブローカー) にデプロイする必要があります。 そのようにしないと、IBM Integration Bus エージェントでトランザクション・トラッキングを有効にした後でも、ミドルウェア・ダッシュボードおよびトポロジー・ダッシュボードにデータは表示されません。

ヒント : 以下の IBM Integration Bus ノードが、KQIUserExit ユーザー出口によって、非インスツルメント・サービスとしてミドルウェアおよびトポロジー・ダッシュボードに組み込まれます。

- ODBC データ・ソースが指定されているデータベース・ノードおよび計算ノード

- TCP/IP ノード
- リモート FTP/FTPS サーバーのファイル・ノード
- MQ ノード (まだインスツルメントされていない場合)

手順

IBM Integration Bus のトランザクション・トラッキングを有効にするには、以下の手順を実行します。

1. Linux AIX
   MQSI 環境をロードしたすべてのブローカー・シェルを閉じます。
2. 以下のいずれかの方法で IBM Integration Bus コマンド・コンソールを開きます。複数のバージョンの統合ノード (ブローカー) をインストールしている場合は、必ず正しいバージョン用のコマンド・コンソールを開始してください。
   - Windows 「**スタート**」 > 「**IBM Integration Bus**」 > 「**IBM Integration Console**」をクリックします。
   - Linux AIX 統合ノード (ブローカー) のインストール・ディレクトリーの bin ディレクトリーで、**mqsiprofile** コマンドを発行します。
3. **mqsistop** コマンドを使用して、構成する統合ノード (ブローカー) を停止します。
4. **mqsichangebroker** コマンドを使用して KQIUserExit ユーザー出口を追加することにより、統合ノード (ブローカー) 内のメッセージ・フローのトランザクション・トラッキングを有効にします。
   - 統合ノード (ブローカー) 内のすべてのメッセージ・フローのトランザクション・トラッキングを有効にするには、次のコマンドを実行します。

     ```
     mqsichangebroker broker_name -e "KQIUserExit"
     ```

   - 統合ノード (ブローカー) 内の特定のメッセージ・フローのトランザクション・トラッキングを有効にするには、次のコマンドを実行します。

     ```
     mqsichangeflowuserexits broker_name -e execution_group_name -k application_name -f
     message_flow_name -a "KQIUserExit"
     ```

5. または、Ace バージョン 11 で、KQIUserExit を node.conf.yaml ファイルまたは server.conf.yaml ファイルに追加することにより、統合ノード (ブローカー) 内のメッセージ・フローのトランザクション・トラッキングを有効にします。

   ```
   UserExits:
   activeUserExitList: 'KQIUserExit' # Specify the name
   #of an installed user exit to activate.
   ```

   注: トランザクション・トラッキングを無効にするには、**activeUserExitList: 'KQIUserExit'** をコメント化します。
6. **mqsistart** コマンドを使用して統合ノード (ブローカー) を再始動します。

トランザクション・トラッキングの無効化

手順

IBM Integration Bus のトランザクション・トラッキングを無効にするには、以下の手順を実行します。

1. 以下のいずれかの方法で IBM Integration Bus コマンド・コンソールを開きます。複数のバージョンの統合ノード (ブローカー) をインストールしている場合は、必ず正しいバージョン用のコマンド・コンソールを開始してください。
   - Windows 「**スタート**」 > 「**IBM Integration Bus**」 > 「**IBM Integration Console**」をクリックします。
   - Linux AIX 統合ノード (ブローカー) のインストール・ディレクトリーの bin ディレクトリーで、**mqsiprofile** コマンドを発行します。

2. 以下のいずれかの方法で、統合ノード (ブローカー) 内のメッセージ・フローのトランザクション・トラッキングを無効にします。
   - 特定のメッセージ・フローのトランザクション・トラッキングを無効にするには、**mqsichangeflowuserexits** コマンドを次のように使用します。

     ```
     mqsichangeflowuserexits broker_name -e execution_group_name
     -f message_flow_name -a ""
     ```

   - 統合ノード (ブローカー) 内のすべてのメッセージ・フローのトランザクション・トラッキングを無効にするには、まず **mqsistop** コマンドを使用して統合ノード (ブローカー) を停止してから、次のように **mqsichangebroker** コマンドを発行します。

     ```
     mqsichangebroker broker_name -e ""
     ```

次のタスク

- トランザクション・トラッキングについては、IBM Integration Bus のトランザクション・トラッキングを有効にした後に、エージェントのトランザクション・トラッキングも有効にする必要があります。説明については、295 ページの『IBM Integration Bus エージェントのトランザクション・トラッキングの構成』を参照してください。
- 統合ノード (ブローカー) に対するスナップショット・データ収集を有効にした場合は、IBM Integration Bus エージェントがスナップショット・データを格納しないように構成してください。Cloud APM では、設定された 20 秒のスナップショット間隔で必要となるデータ量と処理量の理由から、スナップショットのアカウンティング・データと統計データをサポートしません。説明については、294 ページの『エージェントに対するスナップショット・データ収集の無効化』を参照してください。

エージェントに対するスナップショット・データ収集の無効化

Cloud APM では、設定された 20 秒のスナップショット間隔で必要となるデータ量と処理量の理由から、スナップショットのアカウンティング・データと統計データをサポートしません。ブローカーに対するスナップショット・データ収集を有効にした場合は、必ず IBM Integration Bus エージェントがスナップショット・データを格納しないように構成してください。

手順

1. エージェントの構成ファイルをテキスト・エディターで開きます。エージェント構成ファイルは、オペレーティング・システムに応じて、以下のいずれかのディレクトリーにあります。
   - Linux AIX *install_dir*/config/<*hostname*>_qi_<*instance_name*>.cfg
   - Windows *install_dir*¥TMAITM6_x64¥<*hostname*>_qi_<*instance_name*>.cfg

   ここで、*install_dir* はエージェントのインストール・ディレクトリーであり、*hostname* はオペレーティング・システムのホスト名であり、*instance_name* はエージェント・インスタンスの名前です。
2. ファイルを編集して、以下のパラメーターを KqiAgent セクションに追加します。

   ```
   defaultRetainRecentSnapshotSamples=0
   ```

   例:

   ```
   INSTANCE=inst1 [
   SECTION=KqiAgent [ { agentId=inst1 } { instName=inst1 }
   {defaultRetainRecentSnapshotSamples=0}]
   SECTION=MonitorBroker:BRK1 [ { collectNodeData=NO } ]
   SECTION=MonitorBroker:BRK2 [ { collectNodeData=NO } ]
   ]
   ```

3. ファイルを保存して閉じます。
4. IBM Integration Bus エージェント を再始動して、変更を有効にします。

IBM Integration Bus エージェントのトランザクション・トラッキングの構成

IBM Integration Bus のトランザクション・トラッキング・データは、IBM Integration Bus エージェントの「**エージェント構成**」ページでデータ収集を有効にした後にミドルウェアおよびトポロジー・ダッシュボードに表示できます。

始める前に

- KQIUserExit という名前のエージェント提供ユーザー出口を使用して IBM Integration Bus に対してトランザクション・トラッキングが有効になっていることを確認します。まだそうなっていない場合は、292 ページの『トランザクション・トラッキングの有効化』の説明に従います。
- IBM Integration Bus エージェントが適切に構成されていることを確認します。まだそうなっていない場合は、283 ページの『IBM Integration Bus エージェントの構成』の説明に従います。

手順

IBM Integration Bus エージェント 用のトランザクション・トラッキングを構成するには、以下の手順を実行します。

1. ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
2. 「**IBM Integration Bus**」タブをクリックします。
3. エージェント・インスタンスのチェック・ボックスを選択し、「**アクション**」リストから以下のいずれかのアクションを実行します。
   - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
   - トランザクション・トラッキング・データを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。

タスクの結果

選択されたエージェント・インスタンスのトランザクション・トラッキングを構成しました。トランザクション・トラッキング・データは、データ収集を有効にした後に、ミドルウェア・ダッシュボードおよびトポロジー・ダッシュボードに表示できます。詳しくは、100 ページの『Application Performance ダッシュボードへの ミドルウェア・アプリケーションの追加』を参照してください。

IBM Integration Bus エージェントに対する固有の管理対象システム名の指定

Cloud APM コンソールに表示されている IBM Integration Bus エージェントのインスタンス名は管理対象システム名 (MSN) とも呼ばれます。エージェント構成パラメーターを使用して、各エージェント・インスタンスに固有の MSN を指定できます。

このタスクについて

IBM Integration Bus エージェントが開始されると、MSN をエージェント・インスタンスごとに *monitoredbrokername*:*agentID*:KQIB の形式で登録します。ここで、*monitoredbrokername* はモニター対象ブローカーの名前、*agentID* はエージェント構成パラメーターによって設定されるエージェント ID です。MSN の最大長は 32 文字です。MSN の長さが 32 文字を超えると、切り捨てられます。

以下の状況では固有の MSN が必要なことがあります。

- 同じシステムで複数の IBM Integration Bus エージェントを実行している。
- 別の複数のシステムで、同じ名前の複数のモニター対象ブローカーを実行している。

固有の MSN を取得するためにエージェント ID を指定するには、対話式の構成中に **Agent Id** オプションを使用するか、サイレント応答ファイルで **agentId** パラメーターを使用します。

要確認 : IBM Integration Bus エージェントをインストール後にまだ構成していない場合、283 ページの『IBM Integration Bus エージェントの構成』に記載されている手順に従ってください。

手順

- 対話式の構成中に **Agent Id** オプションを使用するには、以下の手順を実行します。
  a) 以下のコマンドを入力します。

  ```
  install_dir/bin/iib-agent.sh config instance_name
  ```

  ここで、*instance_name* はエージェント ID を指定する対象のエージェント・インスタンス名です。
  b) オプションに従って、エージェント・インスタンスを構成します。
  オプションを変更する必要がない場合はデフォルト値を使用します。
  c) Agent ID オプションが表示されたら、管理対象システム名の中間修飾子を指定します。
  有効な形式は、最大 8 文字の長さの英数字ストリングです。
- サイレント応答ファイルで **agentId** パラメーターを使用するには、以下の手順を実行します。
  a) テキスト・エディターで以下に示すエージェントのサイレント応答ファイルを開きます。
  – Linux AIX *install_dir*/samples/iib_silent_config.txt
  – Windows *install_dir*¥tmaitm6_x64¥samples¥qi_silent_config.txt
  b) **agentId** パラメーターにエージェント ID を指定します。
  有効な形式は、最大 8 文字の長さの英数字ストリングです。
  c) サイレント応答ファイルを保存して閉じてから、コマンド行で以下のコマンドを実行します。
  – Linux AIX *install_dir*/bin/iib-agent.sh config *instance_name path_to_responsefile*
  – Windows *install_dir*¥BIN¥iib-agent.bat config "*instance_name path_to_responsefile*"

  ここで、*instance_name* は構成するインスタンスの名前であり、*path_to_responsefile* はサイレント応答ファイルの絶対パスです。

  警告 : Windows システムでは、サイレント応答ファイルの絶対パスを囲む二重引用符 ("") を含めないでください。構成エラーの原因となるためです。
  d) 構成の完了後に、以下のコマンドを入力してエージェントを開始します。
  – Linux AIX

  ```
  install_dir/bin/iib-agent.sh start instance_name
  ```

  – Windows

  ```
  install_dir¥bin¥iib-agent.bat start instance_name
  ```

次のタスク

Cloud APM コンソール にログインします。以前の MSN が設定されたエージェント・インスタンスがオフラインとして表示されている場合、アプリケーションを編集してエージェント・インスタンスを削除し、割り当てられたエージェント ID が設定された新しいエージェント・インスタンスを追加します。

KQIUserExit ユーザー出口の削除

IBM Integration Bus エージェントをアンインストールする 前に、まず KQIUserExit ユーザー出口を削除する必要があります。

手順

トランザクション・トラッキングのために IBM Integration Bus にデプロイした KQIUserExit ユーザー出口を削除するには、以下の手順を実行します。

1. IBM Integration Bus エージェントの bin ディレクトリーに移動します。
   - Windows *agent_install_dir*¥*arch*¥qi¥bin
   - Linux AIX *agent_install_dir*/*arch*/qi/bin

   ここで、
   - *agent_install_dir* は、エージェントのインストール・ディレクトリーです。デフォルトは、Windows システムの場合は C:¥IBM¥APM で、Linux システムおよび AIX システムの場合は /opt/ibm/apm/agent です。
   - *arch* はプラットフォームのアーキテクチャー・コードです。例えば、lx8266 は、Linux Intel v2.6 (64 ビット) を表します。アーキテクチャー・コードの完全なリストについては、*agent_install_dir*/archdsc.tbl ファイルを参照してください。
2. **configDC** スクリプトを実行し、ユーザー出口ライブラリーを対話式に削除します。
   - Windows

     ```
     configDC.bat -disable iib_install_dir
     ```
   - Linux AIX

     ```
     ./configDC.sh -disable iib_install_dir
     ```

   ここで、*iib_install_dir* は、IBM Integration Bus のインストール・ディレクトリーです。

例

次の例では、AIX システムにインストールされているバージョン 9.0 のブローカーのエージェント提供ユーザー出口を削除します。

```
cd /opt/IBM/ITM/aix513/qi/bin
./configDC.sh -disable /opt/IBM/mqsi/9.0
```

IBM MQ Appliances モニターの構成

MQ Appliance エージェントは複数インスタンス・エージェントです。インストール後に、エージェントを使用してモニターを開始するには、エージェント・インスタンスを作成してエージェントを構成しておく必要があります。

始める前に

- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。

手順

- Linux システムおよび UNIX システムでは、応答を要求する構成スクリプトまたはサイレント応答ファイルを使用してエージェントを構成できます。
  - 298 ページの『プロンプトへの応答によるエージェントの構成』
  - 299 ページの『サイレント応答ファイルを使用したエージェントの構成』

- Windows システムでは、サイレント応答ファイルでのみエージェントを構成できます。
  - 299 ページの『サイレント応答ファイルを使用したエージェントの構成』

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログへのデフォルト・パスを以下にリストします。

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

プロンプトへの応答によるエージェントの構成

MQ Appliance エージェントで IBM® MQ Appliance のモニターを開始するには、事前にこのエージェントにインスタンス名を割り当てて、エージェントを構成する必要があります。

手順

スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

1. 次のコマンドを実行します。

   ```
   install_dir/bin/mq_appliance-agent.sh config instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

   例:

   ```
   /opt/ibm/apm/agent/bin/mq_appliance-agent.sh config AQM904
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、300 ページの『MQ Appliance エージェントの構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/mq_appliance-agent.sh start instance_name
   ```

   例:

   ```
   /opt/ibm/apm/agent/bin/mq_appliance-agent.sh start AQM904
   ```

タスクの結果

これで、Cloud APM コンソールにログインし、アプリケーション・エディターを使用して MQ Appliance エージェント・インスタンスを Application Performance ダッシュボードに追加できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。

1. 以下のいずれかのディレクトリーにある mq_appliance_silent_config.txt ファイルをテキスト・エディターで開きます。
   - Linux AIX *install_dir*/samples/mq_appliance_silent_config.txt
   - Windows *install_dir*¥samples¥mq_appliance_silent_config.txt

   ここで、*install_dir* はエージェント・インストール・ディレクトリーです。例えば、/opt/ibm/apm/agent です。
2. mq_appliance_silent_config.txt ファイルで、すべての必須パラメーターに値を指定し、必要に応じて他のパラメーターのデフォルト値を変更します。

   各構成パラメーターの説明については、300 ページの『MQ Appliance エージェントの構成パラメーター』を参照してください。
3. mq_appliance_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
   - Linux AIX

     ```
     install_dir/bin/mq_appliance-agent.sh config instance_name path_to_silent_file
     ```
   - Windows

     ```
     install_dir¥bin¥mq_appliance-agent.bat config instance_name path_to_silent_file
     ```

   ここで、
   - *instance_name* は、エージェント・インスタンスに指定する名前です。例えば、AQM904 です。
   - *path_to_silent_file* は mq_appliance_silent_config.txt ファイルへのパスです。例えば、/opt/ibm/apm/agent/samples/mq_appliance_silent_config.txt です。
4. 構成の完了後に、以下のコマンドを実行してエージェントを開始します。
   - Linux AIX

     ```
     install_dir/bin/mq_appliance-agent.sh start instance_name
     ```
   - Windows

     ```
     install_dir¥bin¥mq_appliance-agent.bat start instance_name
     ```

タスクの結果

これで、Cloud APM コンソールにログインし、アプリケーション・エディターを使用して MQ Appliance エージェント・インスタンスを Application Performance ダッシュボードに追加できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

MQ Appliance エージェントの構成パラメーター

MQ Appliance エージェントの構成パラメーターを、セクションに従ってグループ化して表に表示します。

- 300 ページの表 24: SNMP イベントを受信し、V3 イベントをデコードするためのプロパティー。
- 301 ページの表 25: Java 設定のためのプロパティー。
- 301 ページの表 26: HTTP プロバイダーが使用するプロキシー・サーバーのプロパティー。
- 301 ページの表 27: HTTP サーバーのプロパティー。
- 302 ページの表 28: MQ アプライアンスに接続するためのプロパティー。

表 *24. SNMP* イベント構成パラメーター

パラメーター名	説明	サイレント構成ファイル内のパラメーター名
ポート番号	SNMP イベントを listen するために使用されるポート番号。デフォルトは 162 です。	KQZ_SNMPEVENT_PORT
セキュリティー・レベル	SNMP イベントに接続する際に使用するセキュリティー・レベル。以下のいずれかの値を指定できます。 • 1=noAuthNoPriv • 2=authNoPriv • 3=authPriv デフォルトは 2 です。	KQZ_SNMPEVENT_ SECURITY_LEVEL
ユーザー名	SNMP エージェントに接続するために使用するユーザー名。デフォルトは、`snmpuser` です。	KQZ_SNMPEVENT _USER_NAME
認証プロトコル	SNMP エージェントに接続するために使用する許可プロトコル。以下のいずれかの値を指定できます。 • 1=MD5 • 2=SHA デフォルトは 2 です。	KQZ_SNMPEVENT_AUTH _PROTOCOL
認証パスワード	SNMP エージェントに接続するために使用する許可パスフレーズ。	KQZ_SNMPEVENT_AUTH _PASSWORD
プライバシー・パスワード	SNMP エージェントに接続するために使用するプライバシー・パスフレーズ。	KQZ_SNMPEVENT_PRIV _PASSWORD
トラップ構成ファイル	トラップ構成ファイルの場所。	KQZ_SNMPEVENT_ TRAPCNFG_FILE

表 25. *Java* 構成パラメーター

パラメーター名	説明	サイレント構成ファイル内のパラメーター名
Java トレース・レベル	Java プロバイダーで使用されるトレース・レベル。以下のいずれかの値を指定できます。 • 1=Off • 2=Error • 3=Warning • 4=Information • 5=Minimum Debug • 6=Medium Debug • 7=Maximum Debug • 8=All デフォルトは 2 です。	JAVA_TRACE _LEVEL

表 26. プロキシー・サーバーの構成パラメーター

パラメーター名	説明	サイレント構成ファイル内のパラメーター名
プロキシー・ホスト名	プロキシー・サーバーのホスト名。	KQZ_HTTP _PROXY_HOSTNAME
プロキシー・ポート	プロキシー・サーバーのポート番号。デフォルトは 80 です。	KQZ_HTTP_PROXY_PORT
プロキシー・ユーザー名	プロキシー・サーバーのユーザー名。	KQZ_HTTP _PROXY_USER
プロキシー・パスワード	プロキシー・サーバーのパスワード。	KQZ_HTTP _PROXY_PASSWORD

表 27. *HTTP* サーバーの構成パラメーター

パラメーター名	説明	サイレント構成ファイル内のパラメーター名
HTTP ユーザー名	MQ アプライアンス REST 管理インターフェースにアクセスするためのユーザー名。	KQZ_HTTP _USER
HTTP パスワード	MQ アプライアンス REST 管理インターフェースにアクセスするためのパスワード。	KQZ_HTTP _PASSWORD
証明書検証が有効	証明書検証を有効にするかどうか。以下のいずれかの値を指定できます。 • 1=true • 2=false デフォルトは 2 です。	KQZ_HTTP_CERTIFICATE _VALIDATION

表 28. MQ アプライアンス接続構成パラメーター		
パラメーター名	**説明**	**サイレント構成ファイル内のパラメーター名**
アプライアンスのホストまたは IP アドレス (Appliance Host or IP Address)	MQ アプライアンスのホスト名または IP アドレス。デフォルトは `https://hostnameoripaddress:` `https://9.123.123.123` です。	KMK_APPLIANCE_HOST_OR_IP_ADDRESS.arm1
アプライアンスのポート番号	MQ アプライアンスへの HTTPS 接続用のポート番号。デフォルトは 5554 です。	KMK_APPLIANCE_PORT_NUMBER.arm1
アプライアンス・ユーザー名	MQ アプライアンスに接続するために使用するユーザー名。	KMK_APPLIANCE_USER_NAME.arm1
アプライアンス・ユーザー・パスワード	MQ アプライアンス・ユーザーのパスワード。	KMK_APPLIANCE_USER_PASSWORD.arm1
エージェント・ホスト ID (Agent Host Identification)	MQ Appliance エージェントが実行されているシステムのホスト名。デフォルトは `9.123.123.111` です。	KMK_APM_AGENT_IDENTIFICATION.arm1
証明書検証が有効	HTTP 接続の証明書検証を有効にするかどうか。 • 1=true • 2=false デフォルトは 2 です。	KMK_CERTIFICATE_VALIDATION_ENABLED.arm1

InfoSphere DataStage モニターの構成

エージェントが DataStage サーバーのリソースの正常性およびパフォーマンスをモニターするためのデータを収集できるように、DataStage エージェントを構成する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for DataStage agent を参照してください。

このタスクについて

DataStage エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

製品バージョンとエージェント・バージョンは、通常、異なります。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**デフォルトを使用して構成**」をクリックします。

   「**Monitoring Agent for DataStage**」ウィンドウが開きます。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。
4. 「**Monitoring Agent for DataStage**」ウィンドウで、構成パラメーターの値を指定し、「**OK**」をクリックします。

   構成パラメーターについては、305 ページの『エージェントの構成パラメーター』を参照してください。
5. 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックしてエージェントを開始します。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
   例: `/opt/ibm/apm/agent/bin`
2. 以下のコマンドを実行します。ここで、instance_name は、インスタンスに付ける名前です。

   ```
   ./datastage-agent.sh config instance_name
   ```
3. コマンド行に以下のメッセージが表示されたら、1 と入力して Enter キーを押します。

   ```
   Edit 'Monitoring Agent for DataStage' setting? [1=Yes, 2=No]
   ```
4. プロンプトが出されたら、構成パラメーターの値を指定します。

   構成パラメーターについては、305 ページの『エージェントの構成パラメーター』を参照してください。
5. 以下のコマンドを実行して、エージェントを開始します。

   ```
   ./datastage-agent.sh start instance_name
   ```

環境変数の構成

環境変数を構成して、DataStage エージェントの動作を変更することができます。

手順

1. テキスト・エディターで以下のファイルを開きます。
   a) Windows *install_dir*¥TMAITM6_x64¥KDTENV_*instance_name*
   b) Linux *install_dir*/config/.dt.environment
2. 以下の環境変数を編集します。
   - **KDT_FIRST_COLLECTION_INTERVAL**: 最初のデータ収集の時間間隔 (秒)。この時間間隔は、エージェントが開始するまでに指定された時刻にエージェントが前回のジョブ実行データを収集する期間に設定します。デフォルト値は 300 秒 (5 分) です。したがって、エージェントが 2:00 PM に開始する場

合は、1:55 PM から 2:00 PM までのジョブ実行データを収集します。これは、エージェントがデータ収集を開始するときのヒストリカル・ジョブ実行のデータ・ストームを回避するためです。その後のジョブ実行のすべてのエージェント・データ収集では、最後の収集以降に実行される、新しく追加されたジョブ実行のみがフェッチされます。

- **KDT_SSL_CONTEXT**: サービス層 (WebSphere Application Server) で有効になっている SSL プロトコル。デフォルト値は TLS です。
- **KDT_META_SCHEMA_NAME**: メタデータ・リポジトリー用に作成されるデータベース・スキーマの名前。デフォルト値は、Db2 の場合は DSODB、MSSQL および Oracle データベースの場合は xmeta です。
- **KDT_DATABASE_SERVICE_NAME**: エージェントがメタデータ・リポジトリーに接続するために使用するデータベース名またはサービス名。デフォルト値は、Db2 の場合は XMETA、MSSQL の場合は xmeta、Oracle データベースの場合は ORCL です。
- **KDT_DISABLED_ATTRIBUTEGROUP**: データ収集を使用不可にする必要がある属性グループのコンマ区切りリスト。それぞれの属性グループに対して以下の値を単一値または複数値として設定できます。JobRuns、JobProperties、JobRunLog、JobStages、JobParameters、EngineSystemConfiguration、EngineSystemResources、EngineServiceStatus、EngineStatusSummary、JobActivity、AgentConfiguration、および JobConfiguration。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システムおよび Windows システム上で DataStage エージェントを構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. テキスト・エディターで、次の場所にあるサイレント構成ファイルを開き、すべてのパラメーターの値を指定します。

   Windows *install_dir*¥samples¥datastage_silent_config.txt
   Linux *install_dir*¥samples¥datastage_silent_config_UNIX.txt

   Windows C:¥IBM¥APM¥samples

   Linux /opt/ibm/apm/agent/samples

   構成パラメーターについては、305 ページの『エージェントの構成パラメーター』を参照してください。
2. コマンド行で、パスを *install_dir*¥bin に変更します。
3. 次のコマンドを実行します。

   Windows datastage-agent.bat config *instance_name install_dir*¥samples¥datastage_silent_config.txt
   Linux datastage-agent.sh config *instance_name install_dir*¥samples¥datastage_silent_config_UNIX.txt
4. エージェントを開始します。

   Windows 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。
   Linux 次のコマンドを実行します。./datastage-agent.sh start *instance_name*

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

エージェントの構成パラメーター

DataStage エージェントの構成中に、サービス層、メタデータ・リポジトリー、および拡張構成パラメーターを変更できます。

サービス層の構成パラメーター

エージェントがサービス層に接続するために必要な構成パラメーター。

以下の表に、DataStage エージェントのサービス層の構成パラメーターの詳細な説明を示します。

表 *29.* サービス層の構成パラメーターの名前および説明

パラメーター名	説明	必須フィールド
ホスト名	サービス層がインストールされているコンピューターのホスト名。コンピューターがドメインの一部である場合は、完全修飾ドメイン名 (FQDN) を指定します。デフォルト値は localhost です。	はい
HTTPS ポート	サービス層がインストールされているコンピューターの REST インターフェース 用の HTTPS ポート。 デフォルト値は 9443 です。	はい
WAS ユーザー名	WebSphere Application Server に接続するためのユーザー名。デフォルト値は wasadmin です。	はい
WAS パスワード	WebSphere Application Server に接続するためのパスワード。	はい
WAS パスワードの確認 (Confirm WAS Password)	「**WAS パスワード**」フィールドに指定したパスワード。	はい

メタデータ・リポジトリーの構成パラメーター

エージェントがメタデータ・リポジトリーに接続するために必要な構成パラメーター。

以下の表に、DataStage エージェントのメタデータ・リポジトリーの構成パラメーターの詳細な説明を示します。

表 *30.* メタデータ・リポジトリーの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
データベース・タイプ	メタデータ・リポジトリーのデータベース・タイプ。Db2 デフォルト値は 1 です。	はい
ホスト名	メタデータ・リポジトリーがインストールされているコンピューターのホスト名。コンピューターがドメインの一部である場合は、完全修飾ドメイン名 (FQDN) を指定します。デフォルト値は localhost です。	はい
データベース・ポート	JDBC 接続用のメタデータ・リポジトリー上のデータベース・ポート。 デフォルト値は 50000 です。	はい
データベース・ユーザー名	オペレーション・データベースに接続するためのユーザー名。デフォルト値は dsodb です。	はい

表 30. メタデータ・リポジトリーの構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
データベースのパスワード	オペレーション・データベースに接続するためのパスワード。	はい
データベース・パスワードの確認 (Confirm Database Password)	「**データベースのパスワード**」フィールドに指定したパスワード。	はい
JDBC ドライバー・パス	jar ファイルを含む JDBC ドライバーのパス。例えば、Linux の場合は /home/jars/db.jar です。	はい

拡張構成パラメーター

表 31. 拡張構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
Java トレース・レベル	Java カスタム・プロバイダーで使用するトレース・レベル。デフォルト値は 2 です。	はい

Java API クライアント構成パラメーター

表 32. *Java API* クライアント構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
外部 JAR のクラスパス		いいえ

Internet Service Monitor の構成

Internet Service Monitoring エージェントは、特定のサービスが適切に実行されているかどうかを判別し、問題の領域を特定し、サービス・レベル・アグリーメントに照らして測定したサービス・パフォーマンスを報告する機能を提供します。Internet Service Monitoring エージェント は、実ユーザーのアクションをエミュレートすることによって機能します。インターネット・サービスを定期的にポーリングまたはテストして状況およびパフォーマンスを検査します。

概要

インターネット・サービスをモニターする際、モニターする対象、そのユーザー、および時期を定義します。Internet Service Monitoring エージェント の構成ユーザー・インターフェースを 使用して、モニターを構成できます。

モニターは、特定のインターネット・サービスをテストして、テストの結果を Databridge に転送します。モニターは、サービスの実ユーザーのアクションをエミュレートします。

例えば、HTTP モニターは、ユーザーがページにアクセスするときに Web ブラウザーが通常送信する要求をエミュレートすることによって、定期的に Web ページへのアクセスを試みます。モニターはテストの結果を記録し、結果はデータ・ブリッジに送信されます。

Internet Service Monitoring

各モニターは、あるタイプのプロトコルまたはサービスをテストするように 設計されています。 例えば、HTTP モニターは、Hypertext Transfer Protocol (HTTP) を介して Web ページなどのリソースの可用性をテストし、FTP モニターは、File Transfer Protocol (FTP) を実行するホスト間のファイルの転送をテストします。

モニターは、さまざまなホストによってサービスを提供される一連の Web ページなど、同じサービスの多くの異なるインスタンスをテストすることができます。

Web サービス・モニタリング

一連の Internet Service Monitoring モニターを使用すると、Web ページの可用性をテストする基本の Internet Service Monitoring からテストの手順の結合まで、提供する Web サービス・モニターのタイプを調整することができます。

Internet Service Monitoring は、大容量の複雑度が低いポーリングを使用して Web サービスの可用性をテストします。例えば、Web サイトの一般的な可用性をモニターするには、HTTP モニターを使用して定期的に多数の URL をポーリングします。

モニターの組み合わせを使用することにより、ユーザーの要件に適切なレベルのサービス・モニターを作成することができます。

- HTTP および HTTPS モニター

  基本的な単一要求のテストを大量に実行することによって、HTTP または HTTPS を介してリソースの可用性をモニターします。

- トランザクション・モニター (TRANSX)

  モニターのグループによって実行されたテストの手順を結合して、実ユーザーのアクションをシミュレートします。例えば、サービスをダイヤルアップして、いくつかの Web サイト上の多数のページにアクセスしてから、電子メール・サービスにアクセスします。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

ユーザー・インターフェースを使用した Internet Service Monitoring の構成

インターネット・サービスをモニターするには、ユーザー・プロファイル、プロファイル・エレメント、およびモニター・スケジュールを作成します。Internet Service Monitoring ユーザー・インターフェースを使用して、ユーザー・プロファイル、プロファイル・エレメント、およびモニター・スケジュールを構成します。

このタスクについて

ユーザー・プロファイルは、インターネットまたは Web サービスのモニター対象となる顧客、部署、またはサービス・グループを表します。ユーザーは、ユーザー・プロファイルごとに 1 つ以上のプロファイル・エレメントを定義する必要があります。 例えば、ユーザーは HTTP サービス経由で配信される Web ページをモニターするプロファイル・エレメントを定義したり、FTP サービスの可用性をモニターするプロファイル・エレメントを定義したりします。通常、ユーザー・プロファイルには、複数のプロファイル・エレメントが含まれており、各プロファイル・エレメントによって、そのユーザーに提供されるサービスの 1 つがテストされます。各ユーザー・プロファイルには、関連するモニター・スケジュールも含まれています。このスケジュールによって、プロファイルに定義されたテストの実行日時が決定されます。

IBM Application Performance Management ダッシュボードを介して Internet Service Monitoring エージェントの構成ウィンドウにアクセスするには、以下の方法を使用します。

手順

1. Application Performance Management ダッシュボードで、 アイコンをクリックします。「**エージェント構成**」をクリックします。

   「エージェント構成」ウィンドウが開きます。
2. 「**ISM**」タブをクリックして、Internet Service Monitoring エージェントを構成します。

   ユーザー・プロファイルを作成、編集、削除、最新表示、スケジュール、およびフィルタリングすることができます。選択された管理対象システムに、作成されたユーザー・プロファイルをデプロイします。表示されるバージョンは、管理対象システムのバージョンです。プロファイル名は、選択した

管理対象システムにデプロイされているユーザー・プロファイルを示します。プロファイルを構成し、管理対象システムにデプロイするには、以下の手順を実行します。

3. プロファイルを追加するには、 アイコンをクリックします。ダイアログ・ボックスで「**プロファイル名**」および「**説明**」を入力します。
4. 「**次へ**」をクリックします。
5. モニター・ドロップダウン・リストからモニターを選択して、「**次へ**」をクリックします。
6. フィールドの値を指定し、「**追加**」をクリックします。

   1 つのプロファイルに対して複数のモニターを選択できます。使用可能なモニターについては、311 ページの『使用可能な Internet Service Monitoring モニター』を参照してください。
7. 「**完了**」をクリックします。
8. アイコンをクリックします。
9. 「**フィルター**」フィールドで、ユーザー・プロファイルをその名前で検索します。
10. 作成したプロファイルを管理対象システムでデプロイするには、構成する作成済みプロファイルのチェック・ボックスを選択し、管理対象システムを選択します。「**デプロイ**」をクリックして、選択した管理対象システム名に対してプロファイルをデプロイします。

プロファイルの編集

作成したユーザー・プロファイルはすべて編集可能です。

このタスクについて

プロファイルを編集するには、以下の手順を使用します。

手順

1. 「**プロファイル名**」を選択して、 をクリックします。
2. 編集するサービスを選択して、「**編集**」をクリックします。

   a. アイコンを 使用してモニターを追加し、 アイコンを 使用してモニターを削除します。

   b. プロファイル名を変更するには、「**プロファイル名**」テキスト・フィールドをダブルクリックし、プロファイル名を変更し、「**プロファイル名の変更**」をクリックします。
3. 選択したサービスの値を編集します。

   注 :

   - パスワード・フィールドを編集できるようにするには、「**ユーザー名**」テキスト・フィールドをダブルクリックします。 ユーザーは、ユーザー名の編集または追加、およびパスワードの変更を実行できます。
   - 「**sslkeypassword**」フィールドを編集できるようにするには、そのテキストをダブルクリックして、秘密鍵を変更します。
4. 「**保存**」をクリックし、 アイコンをクリックして 最新表示します。

プロファイルのスケジューリング

作成したプロファイルは、特定の日時にデプロイするようにスケジュールできます。

このタスクについて

プロファイルをスケジュールするには、以下の手順を使用します。

手順

1. 「**プロファイル名**」を選択します。
2. 「**スケジュール**」ボタンをクリックします。
3. 時刻に対する日付を選択してプロファイルをスケジュールします。ユーザーはグリッドをドラッグして任意の目的の時刻を選択できます。

4.「**保存**」をクリックします。
5. アイコンをクリックして 最新表示します。

プロファイルの削除
作成したプロファイルを完全に削除できます。

このタスクについて

プロファイルを削除するには、以下の手順を使用してください。

手順

1.「**プロファイル名**」を選択します。
2. プロファイルを削除するには、 アイコンをクリックします。

OID グループ
オブジェクト ID (OID) グループは、オプションのモニター固有パラメーターです。デバイスの管理情報ベース (MIB) オブジェクトの 1 つ以上の OID が含まれるセットを定義します。SNMP モニターは、OID グループを使用して、指定された OID グループに含まれる OID を持つ MIB オブジェクトからデータを取り出します。

モニターがデータを抽出する MIB オブジェクトの詳細を以下に示します。

- OID 値

  ASN.1 表記 (例えば、.1.3.6.1.2.1.1.2.0) またはオブジェクトの名前 (例えば、sysObjectID.0) を使用して表現された MIB オブジェクト・インスタンスの数値 ID

  注: ASN.1 表記を使用するときは、OID の先頭に . 文字を含める必要があります。

  注: オブジェクトのインスタンス 名を定義する MIB 文書にモニターがアクセスできる場合は、OID 値を指定するために、そのインスタンス名のみを使用します。MIB 文書のデフォルト・ディレクトリーは $ISHOME/mibs です。
- OID 名

  MIB オブジェクトの名前。例えば、sysObjectID。この名前は、サービス・レベル分類と $oidName*n* モニター・エレメントで使用されます。
- OID 単位

  MIB オブジェクトに含まれるデータの単位。秒、バイト、ビット/秒 (BPS) など。 BPS に設定すると、OID に対してビット/秒計算が有効になります。 ビット/秒の値は、以下のように計算されます。

  ```
  current_poll_value - prev_poll_value) / poll_interval * 8
  ```
- セレクター

  MIB オブジェクトの指標値。以下の表に、セレクターがすべての ifDescr 行で値 FastEthernet0/1 を検索し、行のインデックスが 2 という結果になる例を示します。次に、行 ifPhysAddress.2 が照会され、値 0:6:53:34:d2:a1 が返されます。このように、2 というインデックスは直接指定されないため、FastEthernet0/1 のインデックスが変わった場合に、OID グループを再構成する必要はありません。

表 *33.* インデックス値の使用

MIB オブジェクト	**MIB オブジェクト値**
OID 値	ifPhysAddress
OID 名	FastEthernet0/1PhysicalAddress
OID 単位	string

表 33. インデックス値の使用 (続き)	
MIB オブジェクト	**MIB オブジェクト値**
セレクター	ifDescr=FastEthernet0/1

OID グループおよび *MIB* オブジェクトの作成

OID グループは、グローバルに作成されるため、SNMP 対応デバイスをモニターするすべてのユーザー・プロファイルで使用することが可能です。

手順

OID グループおよび MIB オブジェクトを作成するには、以下のステップを実行します。

1. Internet Service Monitoring エージェントのダッシュボードで、「**OID**」ボタンをクリックして OID グループを作成します。
2. ⊕ アイコンをクリックし、「**OID グループ名**」フィールドに OID グループ名を入力します。
3. ⊕ アイコンをクリックして MIB オブジェクトを追加します。
   a. MIB オブジェクトの「**値**」、「**名前**」、「**単位**」、および「**セレクター**」を入力します。
   b.「**追加**」をクリックします。

   MIB オブジェクトが正常に作成されます。
4. ↻ アイコンをクリックして最新表示します。
   OID グループが正常に作成されています。
5.「**OID グループ名**」を選択して「**表示**」をクリックすると、選択した 「**OID グループ**」の下に作成されたすべての MIB オブジェクトのリストが表示されます。
6.「**閉じる**」をクリックします。

OID グループおよび *MIB* オブジェクトの編集

OID グループは編集できます。OID グループの作成中、または OID グループの作成後に MIB オブジェクトを編集することもできます。

手順

OID グループを編集するには、以下のステップを実行します。

1. Internet Service Monitoring エージェントのダッシュボードで、「**OID**」ボタンをクリックして OID グループを編集します。
2.「**OID グループ名**」リストから OID グループ名を選択し、✎ アイコンをクリックします。
3.「**OID グループの編集**」から値を選択し、✎ アイコンをクリックします。
4. 必要に応じて MIB オブジェクト・フィールドを変更し、「**保管**」ボタンをクリックします。
5.「**OID グループ**」ポップアップ・ページで「**保管**」をクリックします。
6.「**閉じる**」をクリックします。

OID グループの削除

MIB オブジェクトは、OID グループに含まれており、SNMP モニターがデータを取得するために使用されます。OID グループから MIB オブジェクトを個々に削除することも、OID グループ全体を削除して MIB オブジェクトをすべて削除することもできます。

手順

OID グループを削除するには、以下のステップを実行します。

1. ODI グループを削除するには、Internet Service Monitoring エージェントのダッシュボードにある「**OID**」ボタンをクリックします。
2.「**OID グループ名**」リストから OID グループ名を選択し、⊖ アイコンをクリックします。

OID グループが MIB オブジェクトとともに削除されます。
3.「**閉じる**」をクリックします。

MIB オブジェクトの削除

手順

MIB グループを削除するには、以下のステップを実行します。

1. MIB オブジェクトを削除するには、Internet Service Monitoring エージェントのダッシュボードにある「**OID**」ボタンをクリックします。
2.「**OID グループ名**」リストから OID グループ名を選択し、 アイコンをクリックします。
3. MIB オブジェクトの値を選択し、 アイコンをクリックします。
   MIB オブジェクトが削除されます。
4.「**OID グループの編集**」ポップアップ・ページで「**保管**」をクリックします。
5.「**閉じる**」をクリックします。

使用可能な Internet Service Monitoring モニター

Internet Service Monitoring エージェント モニター・スイートは、幅広いインターネット・サービスを対象としています。

次の表に、Internet Service Monitoring エージェント で使用できるモニター、およびそれらがモニターするサービスのタイプをリストします。

表 *34.* 使用可能なインターネット・サービス・モニター

モニター名	**モニター対象のサービスのタイプ**
DHCP	動的ホスト構成プロトコル。 DHCP を構成するには、324 ページの『DHCP モニター』を参照してください。
DNS	ドメイン・ネーム・サービス。 DNS を構成するには、326 ページの『DNS モニター』を参照してください。
FTP	File Transport Protocol。 FTP を構成するには、331 ページの『FTP モニター』を参照してください。
HTTP	HyperText Transport Protocol。 HTTP を構成するには、335 ページの『HTTP モニター』を参照してください。
HTTPS	HyperText Transport Protocol (Secure)。 HTTPS を構成するには、345 ページの『HTTPS モニター』を参照してください。
ICMP	Internet Control Message Protocol。 ICMP を構成するには、350 ページの『ICMP モニター』を参照してください。
LDAP	Lightweight Directory Access Protocol。 LDAP を構成するには、355 ページの『LDAP モニター』を参照してください。
IMAP4	Internet Message Access Protocol。 IMAP4 を構成するには、361 ページの『IMAP4 モニター』を参照してください。

表 34. 使用可能なインターネット・サービス・モニター (続き)

モニター名	モニター対象のサービスのタイプ
NTP	Network Time Protocol。 NTP を構成するには、366 ページの『NTP モニター』を参照してください。
NNTP	Network News Transport Protocol。 NNTP を構成するには、369 ページの『NNTP モニター』を参照してください。
POP3	Post Office Protocol。 POP3 を構成するには、374 ページの『POP3 モニター』を参照してください。
RADIUS	Remote Authentication Dial-In User Service。 RADIUS を構成するには、379 ページの『RADIUS モニター』を参照してください。
RPING	リモート Ping (Cisco、Juniper、および RFC2925)。 RPING を構成するには、384 ページの『RPING モニター』を参照してください。
RTSP	Real-Time Streaming Protocol。 RTSP を構成するには、389 ページの『RTSP モニター』を参照してください。
SAA	Cisco Service Assurance Agent。 SAA を構成するには、394 ページの『SAA モニター』を参照してください。
SIP	Session Initiation Protocol。 SIP を構成するには、410 ページの『SIP モニター』を参照してください。
SMTP	Simple Mail Transport Protocol。SMTP を構成するには、414 ページの『SMTP モニター』を参照してください。
SNMP	Simple Network Management Protocol。 SNMP を構成するには、419 ページの『SNMP モニター』を参照してください。
SOAP	XML ベースのメッセージング・プロトコル。 SOAP を構成するには、423 ページの『SOAP モニター』を参照してください。
TCPPort	Transmission Control Protocol。 TCPPort を構成するには、428 ページの『TCPPort モニター』を参照してください。
TFTP	Trivial File Transfer Protocol。 TFTP を構成するには、432 ページの『TFTP モニター』を参照してください。
TRANSX	トランザクション。 TRANSX を構成するには、437 ページの『TRANSX モニター』を参照してください。

ファイル

実行可能ファイル

各インターネット・サービス・モニターは、実行可能ファイル、プロパティー・ファイル、ルール・ファイル、およびログ・ファイルで構成されます。

モニター実行可能ファイルは、$ISHOME/platform/arch/bin ディレクトリーにあります。arch の値は、Windows オペレーティング・システムのアーキテクチャー・コードで、win 32 です。

プロパティー・ファイル

プロパティー・ファイルは、テキスト・ファイルであり、ハッシュ・シンボルが前に付いたデフォルトの設定値が入っています。

設定値を変更するには、デフォルトの設定値を変更してハッシュ・シンボルを除去するか、あるいはデフォルトの設定値が入った行をコピーして貼り付け、変更を行って、ハッシュ・シンボルを除去します。 こうすると、後でデフォルトを復元することができます。モニター・プロパティー・ファイルは、$ISHOME/etc/props ディレクトリーにあります。

ルール・ファイル

ルール・ファイルは、IBM Application Performance Management Netcool/OMNIbus プローブ・ルール・ファイルに似ています。これらの構文については、「*IBM Application Performance Management Netcool/OMNIbus Probe and Gateway Guide*」を参照してください。

モニター・ルール・ファイルは、$ISHOME/etc/rules ディレクトリーにあります。

ログ・ファイル

ログ・ファイルは、モニターの操作に関するメッセージを保管します。

モニター・ログ・ファイルは、$ISHOME/log ディレクトリーにあります。MessageLog プロパティーは、ログ・ファイルの場所と名前を決定します。MessageLevel プロパティーは、ログ・ファイルに書き込まれる情報のレベルを選択します。例えば、詳細なデバッグ・メッセージまたは復旧不能エラー・メッセージなどです。MaxLogFileSize プロパティーは、ロールオーバー前のログ・ファイルのサイズを決定します。

ログ・ファイルのデフォルトの名前は、name.log です。ここで、*name* はモニターの名前です。

共通機能

すべてのインターネット・サービス・モニターに共通する機能は数多くあります。これらの機能は、プロパティー、モニターによって作成される結果、および状況メッセージで構成されます。

このセクションでは、すべてのモニターのプロパティーについて説明します。モニター固有のプロパティーについては、個々のモニターのセクションで説明しています。

次の表で、デフォルトのプロパティー・パラメーターには下線が付いています (該当する場合)。

表 *35.* 共通のプロパティー

プロパティー名	プロパティー・パラメーター	説明
AddRoute	0|1	モニターが使用するネットワーク・インターフェースの IP アドレスからモニター対象のホストの IP アドレスへの経路を作成します。 0 - 無効 1 - 有効 (モニターはプロファイル・エレメントで指定された経路を使用して、別のネットワーク・インターフェースを 経由しません)。 **注:** このプロパティーは、AIX プラットフォームおよび HP-UX プラットフォームではサポートされていません。

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
BridgeIPAddress	適用外	データ・ブリッジの IP アドレスを指定します。このプロパティーは構成可能ではありません。データ・ブリッジは常にローカル・ホスト上にあります。
BridgePort	整数	データ・ブリッジによって使用されるポート番号。このプロパティーは、データ・ブリッジの SocketPort プロパティーと同じ値に設定します。 デフォルト: 9510
BridgeSSLAuthenticatePeer	0|1	モニターとブリッジ間、またはブリッジとエージェント間で SSL 認証を構成する必要がある場合、BridgeSSLAuthenticatePeer を 1 に設定し、ブリッジを再始動します。このアクションにより、サーバーからの証明書が認証されます。証明書は BridgeSSLTrustStore に保管されます。 0 - 無効 1 - 有効
BridgeSSLCertificateFile	ストリング	デジタル・ブリッジ SSL 証明書のパスおよびファイル名を指定します。 デフォルト: $ISHOME/certificates/monitorCert.pem
BridgeSSLCipherSet	ストリング	暗号セットを指定します。この値を更新する場合は、OpenSSL 資料で定義されている暗号構文を使用してください。 **注 :** Internet Service Monitoring エージェント、すべてのモニター、および Databridge で同じ値を設定してください。 デフォルト: RC4:3DES:DES:+EXP
BridgeSSLDisableSSLv2	0|1	どのタイプのソケットが受け入れられるかを決定します。 • 0 に設定すると、SSLv2 と SSLv3 の両方が受け入れられます。 • 1 に設定すると、SSLv3 でのみソケットが開きます。 **制約事項 :** Internet Service Monitoring エージェント、すべてのモニター、および Databridge で同じ値を設定してください。

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
BridgeSSLEncryption	0|1	ブリッジ SSL 暗号化を使用可能にします。このプロパティーは、対応するデータ・ブリッジのプロパティーと同じ値に設定します。 0 - 無効 1 - 有効 **注:** すべてのモニターに同じ値を設定します。
BridgeSSLKeyFile	ストリング	ブリッジ SSL 秘密鍵ファイルのパスおよびファイル名。 デフォルト: $ISHOME/certificates/monitorKey.pem
BridgeSSLKeyPassword	ストリング	ブリッジ SSL 秘密鍵の暗号化に使用されるパスワード。 デフォルト: tivoli
BridgeSSLTruststore	ストリング	認証用のトラステッド証明書ファイルのパスおよびファイル名。これは、BridgeSSLAuthenticatePeer 設定を使用する場合にのみ必要です。 モニターとブリッジ間、またはブリッジとエージェント間で SSL 認証を構成する必要がある場合、BridgeSSLAuthenticatePeer を 1 に設定し、ブリッジを再始動します。このアクションにより、サーバーからの証明書が認証されます。証明書は、SSLTrustStoreFile と SSLTrustStorePath の両方に保管できます。 デフォルト: • SSLTrustStoreFile、$ISHOME/certificates/trust.pem • SSLTrustStorePath、$ISHOME/certificates/ 新規証明書を追加するためには、以下のステップのいずれかを実行します。 • SSLTrustStoreFile テキスト・ファイルで証明書をリストの終わりに追加します。 • 証明書を SSLTrustStorePath ディレクトリーに追加し、OpenSSL c_rehash *certificate_dir* コマンドを実行して証明書をハッシュ化します。
BridgeTimeout	整数	モニターがデータ・ブリッジからの応答を待つ時間 (秒単位)。
ConfigFile	ストリング	モニター構成ファイルを指す場合に使用します。 デフォルト: ブランク (空ストリング)。

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
ConfigurationCheckInterval	整数	モニターがプロファイルの変更を検査する間隔 (秒単位)。 デフォルト: 1
Datalog	0|1	モニターが強制的にデータ・ログ・ファイルにパフォーマンス・データを記録するようにします。パフォーマンス・データは次の場所に記録されます。 $ISHOME/datalogs/*userprofile* 0 - 無効 1 - 有効
DatalogFormat	ストリング	データ・ログ・ファイルのフォーマットを定義します。パラメーターはスペースで区切られたエレメントのリストで、その値はデータ・ログ・ファイルに保管される必要があります。データ・ログ・ファイルに書き込まれるポーリング結果ごとに、現在時刻 ($time) および要した時間 ($totalTime) が記録され、その後にこのプロパティーで定義されたすべてのエレメントが続きます。
DatalogNameFormat	ストリング	データ・ログ・ファイル名のフォーマット。
Domain	ストリング	モニターを実行しているホストのドメイン名を指定します。 このプロパティーが設定されない場合、モニターは NIS および DNS 構成を使用してドメイン名を推測しようとします。
DumpProps	適用外	モニターのすべてのプロパティーのリストを 表示します。
FullHostInfo	0|1	$host エレメントを IP アドレス・エレメント $hostIP ($host が DNS 名である場合) にマップするか、DNS 名エレメント ($host が IP アドレスである場合) にマップするかを指定します。 0 - 無効 1 - 有効 **注:** これは、TRANSX モニターでは使用できません。
GroupID	ストリング	モニターを実行する際に使用する必要があるグループ ID。

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
Help	0|1	モニターを実行せずにコマンド行オプションのヘルプを表示します。 0 - 無効 1 - 有効
IdentifierChecksumFields	ストリング	非推奨。
IgnoreUnmatchedDVC	0|1	特定のサービス・レベル分類が一致しない (エレメントがモニターによって作成されていない) 場合に、サービス・レベルの計算でそのエレメントを無視します。 **ヒント:** 以前のリリースの Internet Service Monitoring エージェント では、サービス・レベル分類は Discrete Value Classifications (DVC) と呼ばれていました。 0 - 無効 1 - 有効
IpAddress	ストリング	モニターがテスト中に使用するネットワーク・インターフェースの IP アドレスを指定します。 このプロパティーが設定されない場合、モニターはホスト名ルックアップを使用してホスト・マシンの IP アドレスを判別しようとします。この試行は、ホスト・マシンに複数のネットワーク・インターフェースがある 場合は失敗します。
Manager	ストリング	ObjectServer イベントの重複排除で使用される管理アプリケーションの名前を指定します。
MaxCCA	整数	モニターがどの時点でも持つことができる同時接続の最大数を設定します。この値の設定が高すぎる場合、モニターのパフォーマンスが著しく影響を受ける可能性があることに注意してください。 このプロパティーは、ICMP モニターでは使用できません。 デフォルト: 10
MaxLogFileSize	整数	ログ・ファイルの最大サイズ (バイト単位)。 デフォルト: 1048576
MessageLevel	ストリング	メッセージ・ログに送信されるメッセージの最低レベル。値は、重大度の昇順で、debug、info、warn、error、および fatal です。 デフォルト: warn

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
MessageLog	ストリング	ログ・ファイルのロケーション。 デフォルト: $ISHOME/log/*monitor*.log
MinPoll	整数	許可される最小ポーリング間隔を定義します。モニター構成ファイルのいずれかにこの値より小さいポーリング間隔が設定されている場合、構成ファイルの値は指定変更されます。 デフォルト: 60
MsgDailyLog	整数	日次ログ・ファイルの生成を可能にします。 デフォルト: 0 - 日次ログは使用不可になります
MsgTimeLog	ストリング	MsgDailyLog が使用可能にされている場合に、モニターが日次ログを生成する時間 (24 時間形式の HHMM) を指定します。 デフォルト: 0000 - 12 (深夜)
Name	ストリング	モニターの名前。 このプロパティーを設定すると、PropsFile、RulesFile、および MessageLog プロパティーがデフォルトにリセットされます。
NewProfileCheckMultiple	整数	プロファイルの変更の検査時にモニターが新しい構成ファイルがあるか検査する頻度を示す複数。 デフォルト: 10
NoRecover	整数	モニターに、ストア・アンド・フォワード・ファイルをリカバリーしないように指示します。 デフォルト: 0 - リカバリーは抑止されません
Pause	整数	モニターがスレッドを spawn する間隔 (秒) を設定します。このプロパティーをより高い値に設定すると (100 以上など)、モニターは強制的に低速でスレッドを spawn するようになります。通常、値を増やす必要があるのは、低速システムに限られます。 このプロパティーは、ICMP モニターではサポートされません。 デフォルト: 50
PreviousFields	ストリング	このプロパティーによって指定されたエレメントは ("<element>, <element>, …" の形式を使用して)、1 つのポーリング用に保管され、ストリング previous の接頭部が付けられます。

表 35. 共通のプロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
Profile	ストリング	使用する顧客プロファイル (複数可) の名前。 ストリングは、単一のプロファイル名、スペースで区切られたプロファイル名のリスト、または * (モニターは強制的にすべての使用可能なプロファイルを使用します) にすることができます。 デフォルト: *
ProfileUpdateTimeout	整数	プロファイル・ファイルが静的でなければならない時間 (ミリ秒単位)。この時間を過ぎれば、プロファイル・ファイルをモニターで読み取り、更新することができます。許容範囲は 1 から 20000 ミリ秒です。 デフォルト: 100
PropsFile	ストリング	プロパティー・ファイルの名前。 デフォルト: $ISHOME/etc/props/*monitor*.props
QFile	ストリング	ストア・アンド・フォワード・ファイルの名前を設定します。 デフォルト: $ISHOME/var/*monitor*.saf。
QSize	整数	ストア・アンド・フォワード・ファイルの予約済みサイズ (バイト単位) を設定します。 デフォルト: 10240000
UserID	ストリング	モニターを実行する際に使用する必要があるユーザー ID。 **注:** このプロパティーは、DHCP モニターで使用しないでください。
Version	適用外	モニターを実行せずにモニター・バージョンを印刷します。

共通モニター・エレメント

このセクションでは、すべてのモニターによって作成されるエレメントについて説明します。モニター固有のエレメントについては、個々のモニターのセクションで説明しています。作成されるエレメントは、Internet Service Monitoring エージェント・ダッシュボードで表示できます。

IBM Application Performance Management を使用する場合、エージェントのダッシュボードで属性として表示できるエレメントは、Internet Service Monitoring エージェントが生成するマッピング・ファイルによって決定されます。このマッピング・ファイルは構成可能ではありません。

320 ページの表 36 に、すべてのモニターによって作成されるエレメントをリストしています。アスタリスク (*) で示すエレメントは、ワークスペース属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等のワークスペース属性はありません。括弧内に属性が示されているが、エレメントがないものは、ワークスペース属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 36. 共通モニター・エレメント

エレメント名	エレメントの説明
$consecutiveFailures	$failureRetests がゼロ以外で、サービス・レベル分類に従ってテストが失敗した場合、このエレメントが作成されて、値 1 で始まります。この値は、テストが失敗しなくなって $consecutiveFailures が 0 に設定される時点まで、または後続のポーリングまで増え続けます。 このポーリングで、サービス・レベルが渡されるか、再び増加を開始すると、このエレメントは作成されなくなります。このエレメントの値が $failureRetests の値を超えると、$consecutiveFailures の値は 1 にリセットされます。 **注 :** TRANSX モニターは、このエレメントを生成しません。
$datalogPath* (GUID)	モニターが使用するデータ・ログ・ファイルへのパス。ワークスペース属性は、パスの最後の 100 文字を使用します。
$description* (Description)	モニター・プロファイル・エレメントの「**説明**」フィールドで指定されたテキスト記述が入っています。
$failureRetestInterval	再テストの失敗で使用されるポーリング間隔。これは、$failureRetests が 0 より大きい場合にのみ有効です。再テスト間隔が通常のポーリング間隔より大きい場合、通常のポーリング間隔と等しく設定されます。 **注 :** TRANSX モニターは、このエレメントを生成しません。
$failureRetests	失敗イベントが記録されて ObjectServer に送信される前に超える必要があるサービス・レベル障害の数。 **注 :** TRANSX モニターは、このエレメントを生成しません。
$host* (Host)	ホストまたはサーバーの名前。 構成ファイルに保管されています。
$hostName	$host エレメントのホスト名が含まれています ($host が IP アドレスである場合)。
$hostIP	$host のホスト IP が含まれています ($host が DNS 名である場合)。
$identchecksum* (Identchecksum)	プロファイル・エレメントの ID。
$lastServiceLevel* (LastServiceLevel)	直前のポーリングのサービス・レベル番号。 プロファイルが変更された場合、これはクリアされます。
$lastServiceLevelCounter	直前のポーリングの serviceLevelCounter。プロファイルが変更された場合、これはリセットされます。
$monitorDNSdomain	モニターを実行しているマシンの、DNS が使用するドメイン名。

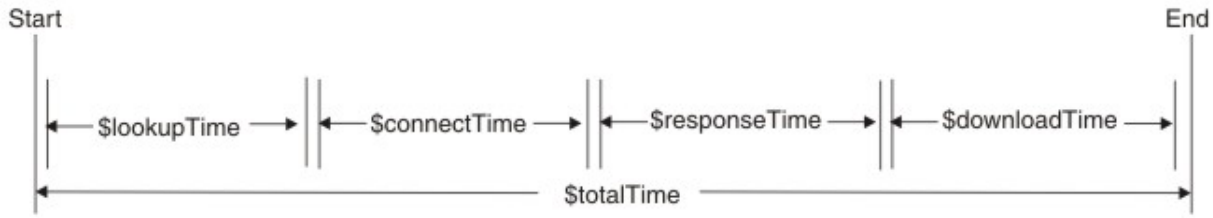
表 36. 共通モニター・エレメント (続き)

エレメント名	エレメントの説明
$monitorHost* (MonitorLocation)	モニターを実行しているホストの名前。
$monitorNISdomain	モニターを実行しているホストの、NIS (ネットワーク情報サービス) が使用するドメイン名。
$monitorDomain	$monitorDNSdomain および $monitorNISdomain の設定を指定変更します。
$message* (結果メッセージ)	ポーリングの結果を説明するテキスト・ストリング。例えば、Connection failed、OK、または Success などです。
(Node)	Internet Service Monitoring が稼働しているシステムの名前。この属性は、Internet Service Monitoring エージェントによって追加されます。
$pollInterval	各モニターによって指定されたポーリング間隔。
$resultString* (ResultString)	ポーリングの結果に適用されたサービス・レベル分類を示すテキスト・ストリング。例えば、TotalTime > 20 などです。
$service* (Service)	モニターされているサービスの名前。 例えば、FTP または HTTP などです。
$serviceLevel* (ServiceLevel)	ポーリングのサービス・レベル番号。サービス・レベル分類で次のように定義されています。 0 - 不明 (Unknown) 1 - 良好 (Good) 2 - 限界 (Marginal) 3 - 失敗 (Failed)
$serviceLevelCounter	サービス・レベル番号が変更されないままでいる回数。
(ServiceLevelString)	返されたサービス・レベルに関連付けられたストリング (Unknown、Good、Marginal、または Failed)。
$startTimePoll	ポーリングが開始された時刻。
$time	ポーリングが行われた UNIX 時刻 (秒単位)。
$timeStamp* (Timestamp)	テストが実行された日時。タイム・スタンプの形式では、ローカルの設定が使用されます。
$transxName	トランザクションの名前。 モニターがトランザクションで使用される場合、これはモニターによって作成されます。

表 36. 共通モニター・エレメント (続き)	
エレメント名	**エレメントの説明**
プロファイルの詳細	
$profile* (IsmProfile)	ユーザー・プロファイルの名前。
タイミング - タイミングの測定方法については、322 ページの『時間計算』を参照してください。	
$timeout	サーバーがそれまでに応答する必要がある秒数。 構成ファイルから取られます。
$totalTime* (TotalTime)	操作の実行にかかった合計時間 (秒)。これには、すべてのルックアップ、接続、ダウンロード時間 (該当する場合)、および中間処理の時間が含まれます。

時間計算

モニターは、ポーリングを完了するのに要する時間をさまざまなタイミングのステージに分割しようとします。例えば、ホストの IP アドレスを取得するのに要する時間、またはホストに正常に接続するために要する時間が含まれます。



$totalTime は、その他の時間の合計よりも若干長くなります。モニターの受信データの処理およびシステム呼び出しの実行といったアクティビティーによって 生じるオーバーヘッドが含まれるためです。$totalTime は、秒単位で測定されます。

状況メッセージ
モニターは、各サービス・テストの後で生成された状況メッセージを返します。状況メッセージは、テストの結果を示します。

メッセージは、通常、モニター対象サービスまたはモニター外部のネットワーク環境から発信されます。323 ページの表 37 に、IBM Application Performance Management を使用している場合に ResultMessage 属性でモニターによって返される共通状況メッセージを示します。モニター固有の状況メッセージについては、個々のモニターのセクションで説明しています。

個々のモニターによって示されるメッセージの他に、HTTP モニターなどの一部のモニターは、基礎となるオペレーティング・システムのメッセージを 報告します。例えば、TCP 接続に障害が発生した場合、Internet Service Monitoring エージェント はオペレーティング・システムによって定義されたストリング (**connection refused**、**timeout**、**network unreachable** およびその他のストリング) を使用します。

表 37. 共通状況メッセージ

メッセージ	説明
OK	モニターからの要求は正常に実行されました。 モニターから、プロセスが正常に実行されたことを示すその他の状況メッセージが出される場合があります。 各モニターの『状況メッセージ』セクションを参照してください。
Received response to request not originating from this monitor - ignored	指定モニターから発信されたのではないメッセージに対するサーバーからの応答を受信しました。
Connection failed Connect to server failed	モニターはサーバーへの接続に失敗しました。詳しくは、ログ・ファイルを参照してください。
Connection closed unexpectedly	サーバーへの接続は失敗しました。
Connection timed out	接続は正常に行われましたが、サーバーが応答を停止しました。
Connection closed by foreign host	リモート・ホストは、モニターが予期するより早く接続を閉じました。
Timed out waiting to read/ write	モニター対象のサーバーへのデータ接続は確立されましたが、応答が停止しました。
No Response from server	要求はタイムアウトになりました。
Format Error Server Failure No such host or domain Not Implemented Request refused Unknown Error	モニター対象のサーバーからエラーが返されました。

表 37. 共通状況メッセージ (続き)	
メッセージ	**説明**
Network is down Network is unreachable Network dropped connection on reset Software caused connection abort Connection reset by peer Connection timed out Connection refused Host is down No route to host Remote peer released connection	ネットワークに問題があります。

DHCP モニター

DHCP モニターは、DHCP サーバーの可用性と応答時間を検査します。

DHCP サーバーが DHCP モニターからの要求に応答する時間に従って、合計時間、ルックアップ時間、または応答時間を使用して、サービス・レベル分類を割り当てます。

表 38. *DHCP* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_dhcp
プロパティー・ファイル	$ISMHOME/etc/props/dhcp.props
ルール・ファイル	$ISMHOME/etc/rules/dhcp.rules
ログ・ファイル	$ISMHOME/log/dhcp.log

DHCP モニターを構成するためのガイドライン

DHCP モニターは、制限された DHCP クライアントとして動作することによって DHCP サービスをテストします。確立された接続でトランスポート・プロトコルとして UDP を使用して、同じネットワーク上のターゲット DHCP サーバーに DHCP INFORM 要求を送信し、サーバーからの対応する DHCP ACK を待機します。モニターは、IP アドレスを要求せず、既存の IP アドレスの有効期限に影響することもありません。

注 : モニター対象の DHCP サーバーは、DHCP INFORM 要求をサポートし、RFC2131 に準拠していなければなりません。

DHCP モニターは、1024 より小さいポートにバインドするため、root として実行する必要があります。

制約事項

DHCP モニターでは、DHCP クライアントを使用して構成したネットワーク・インターフェースは 使用できません。代わりに、IP アドレスが動的に割り当てられないネットワーク・インターフェースを 使用するように、モニターを構成してください。

DHCP モニター・サービス・テストの構成

表 *39. DHCP* モニター構成	
フィールド	**説明**
server	DHCP サーバーのホスト名。例: dhcp1.mycompany.com
ローカル IP	モニターがテストの実行に使用する IP アドレス・ネットワーク・インターフェース。例: 192.168.n.n
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: DHCP モニター
ポート	DHCP サーバーのポート番号。デフォルト値は 67 です。
ローカル・ポート	モニターがテストの実行に使用するポート番号。デフォルト値は 68 です。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。デフォルト値は 30 です。
再試行	モニターが終了するまでに DHCP サーバーへの接続を再試行する回数。デフォルト値は 0 です。例: 2
ポーリング	現在のプロファイル・エレメントを使用することによるサーバーのポーリングの間隔 (秒単位)。デフォルト値は 300 です。
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。デフォルト値は 0 です。
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。デフォルト値は 30 です。

すべてのエレメントに共通するテスト結果に加えて、DHCP モニターは、DHCP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

表 *40. DHCP* モニター・エレメント	
エレメント	**説明**
`$clientIP*ClientIp`	モニターが実行されているホストの IP アドレス。
`$lookupTime*(LookupTime)`	ホスト・サーバーの IP アドレスを取得するために要した時間。
`$responseTime*ResponseTime`	接続が確立されてからデータの最初のバイトが受信されるまでの時間。
`$retries`	モニター構成時に指定された再試行の最大回数。
`$router`	DHCP サーバーによって返されるルーターの IP アドレス。

状況メッセージ

DHCP モニターは、IBM Application Performance Management を使用している場合は `ResultMessage` 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 *41. DHCP* モニターの状況メッセージ	
メッセージ	**説明**
Received DHCPACK Received DHCPNAK	DHCP サーバーは、モニターによって送信された DHCP 通知要求に応答します。
This monitor requires root privileges to run	root としてログインします。
Did not receive valid DHCP MESSAGE	DHCP サーバーからの応答が認識されません。
Did not receive valid DHCP MESSAGE TYPE	DHCP サーバーからの応答が認識されません (DHCPACK または DHCPNAK が予期されます)。
Invalid transaction ID Received response to request not originating from this monitor - ignored	このモニターから発信されていないメッセージに対する DHCP サーバーからの応答を受信しました。
Unexpected op-code returned	このポートで予期しないメッセージを受信しました。
Connection failed	指定されたサーバー名は無効です。
Failed to send request to DHCP server	オペレーティング・システムは、要求をサーバーに送信できなかった理由を具体的に識別できないため、ネットワークに問題があることを示すこの状況メッセージを返します。
No Response from server	DHCP サーバーが応答していません。

DNS モニター

DNS モニターは、DNS (ドメイン・ネーム・システム) サービスを使用して、1 つ以上のホストに関する情報を検索します。

DNS モニターは、ホストの IP アドレスを使用してホスト名を検索するか、ホスト名を使用して IP アドレスを検索します。モニターは、検索の結果および応答時間を記録することによって、サービスのパフォーマンスを測定します。 また、モニターは、サーバーに送信された各照会に関する詳細も記録します。

表 *42. DNS* モニター・ファイルの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_dns
プロパティー・ファイル	$ISMHOME/etc/props/dns.props
ルール・ファイル	$ISMHOME/etc/rules/dns.rules
ログ・ファイル	$ISMHOME/log/dns.log

DNS モニターを構成するためのガイドライン

DNS モニターは、ターゲット・ホストの IP アドレスまたはホスト名をルックアップするように構成することができます。ルックアップのタイプによっては、モニターは異なる方法で DNS サーバーと通信します。

IP アドレス・ルックアップ

IP アドレス・ルックアップ・テストの実行時に、モニターに対してホスト名が指定され、モニターはそのホスト名を使用して IP アドレスを見つけます。

モニターは、以下のようにして DNS をテストします。

1. モニターは、HostA の完全修飾ホスト名 (hosta.dev.net) を使用してその IP アドレスを要求することにより、DNS サーバーを照会します。

   DNS サーバーは、ホストの IP アドレスが見つかった場合、その IP アドレスをモニターに返します。DNS サーバーは、ホストの IP アドレスが見つからなかった場合には、検索の失敗の詳細を含むメッセージをモニターに返します。

   要求がタイムアウトになった場合、モニターは再試行します (再試行するように構成されている場合)。 それ以上再試行しない場合に、モニターは失敗イベントを作成します。

   構成で指定されたホスト名が、hostx.mycompany.com のような完全修飾ホスト名ではなく、mycompany.com などのドメイン名である場合、モニターはドメイン全体に関する情報を取り出します。この情報は、$domainNameServer および $domainNameAddr という 2 つの追加エレメントに格納されます。

2. モニターに返されたメッセージに正規名が含まれている場合、モニターは、構成ファイルで指定された名前は別名であったものと判断します。モニターは、正規名を DNS サーバーに送信して、ホストの IP アドレスを要求します。

   DNS サーバーは、正規名を使用してホストの IP アドレスを見つけた場合、その IP アドレスをモニターに返します。DNS サーバーは、ホストの IP アドレスが見つからなかった場合には、検索の失敗の詳細を含むメッセージをモニターに返します。

3. DNS サーバーを照会する試みが最初に 2 回失敗した場合、モニターは DNS サーバーの IP アドレス (192.168.n.n) を DNS サーバーに送信し、その完全修飾ホスト名を要求します。

   DNS サーバーは、自身の完全修飾ホスト名が見つかった場合、そのホスト名をモニターに返します。DNS サーバーは、自身の完全修飾ホスト名が見つからなかった場合には、検索の失敗の詳細を含むメッセージを返します。サーバーの完全修飾ホスト名の要求 (リバース DNS ルックアップ要求) は、すべてのタイプの DNS サーバーでサポートされているわけではありません。ターゲット DNS サーバーがリバース・ルックアップをサポートしない場合は、LookupServerName プロパティーを 0 に設定することによって、DNS モニターがこの要求を送信しないようにすることができます。

再帰的ルックアップ

非再帰的ルックアップによって、DNS サーバーのパフォーマンスをより正確に把握することができますが、再帰的ルックアップの方が、インターネット・アプリケーション (すなわちユーザー) によって得られる DNS のパフォーマンスを分かりやすく示します。 DNS モニターは、再帰的ルックアップと非再帰的ルックアップの両方をサポートします。

これは、インターネット・アプリケーションが DNS 照会作業を行う一般的な方法です。例えば、Web ブラウザーは、URL のホスト部分を解決しようとする場合、常に再帰的ルックアップを指定します。

データベース内にホストのエントリーが含まれていないために DNS サーバーが照会に応答できない場合、DNS サーバーは階層内の上位の DNS サーバーを再帰的に照会することができます。

DNS 照会タイプ

DNS モニターは、一連の DNS 照会タイプをサポートします。DNS 照会のタイプを指定するときに照会コードを使用します。

表 *43. DNS* 照会タイプ

照会コード	**照会タイプ**
A	ホスト・アドレス
NS	権威あるネーム・サーバー
MD	メール宛先
MF	メール転送プログラム
CNAME	別名に対応する正規名
SOA	権限のゾーンの開始

表 *43. DNS* 照会タイプ *(*続き*)*	
照会コード	**照会タイプ**
MB	メールボックス・ドメイン名
MG	メール・グループ・メンバー
MR	メール名前変更ドメイン名。
NULL	ヌル RR
WKS	既知のサービス記述
PTR	ドメイン名ポインター
HINFO	ホスト情報
MINFO	メールボックスまたはメール・リスト情報
MX	メール交換
TXT	テキスト・ストリング
AXFR	ゾーン全体の転送
MAILB	メールボックス関連レコード
MAILA	メール・エージェント RR
ANY	すべてのレコード

DNS モニター・サービス・テストの構成

DNS モニター構成パラメーターを使用して、DNS サービス・テストを定義します。

表 *44.* 表 *3. DNS* モニター構成	
フィールド	**説明**
サーバー	1 次 DNS サーバーの IP アドレス。例: 192.168.n.n
host	ターゲット・ホストのホスト名。例: www.myconpany.com。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: DNS モニター。
再帰ルックアップ	再帰的ルックアップを有効または無効にします。 • recurse (ismbatch で true を使用します)。 • norecurse (ismbatch で false を使用します)。 デフォルト: recurse
ポート	モニターが listen する DNS サーバー上のポート。デフォルト値は 53 です。
ローカル IP	モニターがテストの実行時にバインドする、ホスト・マシン上のネットワーク・インターフェースの IP アドレスを指定します。モニターの IpAddress プロパティーを設定すると、このフィールドの値がオーバーライドされます。
照会タイプ	テストで使用する DNS 照会タイプ。サポートされる照会タイプのリストについては、327 ページの表 43 を参照してください。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 10。

表 44. 表 3. DNS モニター構成 (続き)	
フィールド	**説明**
再試行	モニターが DNS サーバーへの接続を再試行する回数。この回数だけ再試行すると、モニターは終了します。
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、DNS モニターは、DNS サービス・テストに固有のデータを含む一連のテスト結果を生成します。

表 45. 表 4. DNS モニター・エレメント	
エレメント	**説明**
$authoritative	このエレメントは、権限を持つ DNS サーバーから情報が取り出された場合にのみ作成されます。DNS サーバーに権限がない場合、このエレメントは作成されません。
$domainEmailAddr	ターゲット・ドメインの連絡先アドレス。
$domainNameServer	ターゲット・ドメインの DNS サーバーの名前。
$fromAliasTime	前回の照会から受け取った正規名の要求を発行してから IP アドレスを受け取るまでの時間。
$localIP	モニターが使用するように構成されているローカル IP アドレス。 インターフェースが 1 つのみのマシンではブランクにすることができます。
$lookup*(HostLookup)	モニターが見つけようとしているターゲット・ホストのホスト名または IP アドレス。
$lookupCName	ターゲット・ホストの正式なホスト名。このエレメントは、正式なホスト名が $lookupName のホスト名と異なる場合にのみ作成されます。
$lookupIP*(HostIp)	ターゲット・ホストの IP アドレス。
$lookupName*(Host)	ターゲット・ホストの完全なホスト名。
$mxRecords	検出された MX レコードの数。
$port	サービスがモニターされているポート。
$queryType	テストで使用する DNS 照会のタイプ。サポートされる照会タイプのリストについては、327 ページの表 43 を参照してください。
$responseTime*(ResponseTime)	モニターが DNS サーバーへ要求を送出してから応答を受け取るまでの時間。
$retries	プロファイル・エレメントで指定された再試行の最大回数。

表 45. 表 4. DNS モニター・エレメント (続き)	
エレメント	**説明**
$serverIP	DNS サーバーの IP アドレス。
$serverName	DNS サーバーのホスト名。
$serverTime	サーバーがその名前を解決する時間。

MX レコード処理

DNS モニターによって検出された MX レコードごとに、$mxHostn および $mxPreferencen の 2 つのエレメントが作成されます。

$mxHostn は、MX レコードのホスト名を保管します。$mxPreferencen には、ホストの優先順位の重み付けが含まれます。n は、レコード・ペアごとに増分して、それらを区別します。モニターは、特定ホストの MX レコードの総数をエレメント $mxRecords に保管します。 レコード・ペアは、MX 優先順位の降順でソートされます。

状況メッセージ

DNS モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 46. 表 5. DNS モニターの状況メッセージ	
メッセージ	**説明**
Domain information received	ドメイン名の要求は正常に処理されました。
Success	要求は正常に処理されました。
Invalid Response	DNS サーバーからの応答が認識されません。
Connection failed	指定されたサーバー名は無効です。
No Response from server	要求はタイムアウトになりました。
Failed to send DNS request	ネットワークに問題があります。
No such domain (no recursion)	ドメイン名が間違っています。

プロパティー

次の表では、DNS モニターに固有のプロパティーについて説明しています。

表 47. DNS モニター・プロパティー		
プロパティー名	**プロパティー・パラメーター**	**説明**
AcceptCNAME	0|1	これを有効にすると、DNS モニターは DNS 応答内の正規名を受け入れ、さらにルックアップを行うことはありません。
DNSQueryType	ストリング	テストで使用する DNS 照会タイプ。サポートされる照会タイプのリストについては、327 ページの表 43 を参照してください。 デフォルト: ANY

表 47. *DNS* モニター・プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
LookupServerName	0|1	DNS サーバーの IP アドレスに対するリバース DNS ルックアップを有効にします。 0 - 無効 1 - 有効

FTP モニター

FTP モニターは、FTP サーバーとの間でファイルをアップロードまたはダウンロードすることによって、FTP サービスをテストします。応答時間およびデータ転送速度を記録することによってサービスのパフォーマンスをモニターし、ディスク・スペースおよびファイルの整合性をモニターします。

表 48. *FTP* モニターの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_ftp
プロパティー・ファイル	$ISHOME/etc/props/ftp.props
ルール・ファイル	$ISHOME/etc/rules/ftp.rules
ログ・ファイル	$ISHOME/log/ftp.log

FTP モニターを構成するためのガイドライン

FTP モニターは、FTP STOR コマンドを使用してサーバーへファイルをアップロードするか、FTP RETR コマンドを使用してサーバーからファイルをダウンロードすることによって、FTP サーバーの可用性をテストします。

FTP モニター・サービス・テストの構成

FTP モニターの構成パラメーターについて、次の表で説明します。

表 49. *FTP* モニター構成	
フィールド	**説明**
サーバー	ターゲット FTP サーバー (FTP によって取得しようとするファイルのあるマシン) の IP アドレス。例: ftp.mycompany.com 。
localfile	FTP GET 操作の場合、このフィールドはファイルのダウンロード先の名前とパスを指定します。 FTP PUT 操作の場合、このフィールドは FTP サーバーにアップロードされるファイルの名前とパスを指定します。 デフォルト値は FULL PATHNAME です。例: $ISMHOME/etc/ism/downloads/ftp-test.tar.Z
remotefile	FTP GET 操作の場合、このフィールドはサーバーからダウンロードされるファイルの名前とパスを指定します。 FTP PUT 操作の場合、このフィールドは、アップロードされるファイルの、FTP サーバーでの名前とパスを指定します。 デフォルト値は FULL PATHNAME です。例: /sales/prodlist.tar.Z

表 *49. FTP* モニター構成 *(*続き*)*	
フィールド	**説明**
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	FTP サーバーが使用するデフォルトのポート。 デフォルト: 21
ユーザー名	ターゲット FTP サーバーにログオンするために使用されるユーザー名。
パスワード	ターゲット FTP サーバーにログオンするために使用されるパスワード。FTP アカウントでパスワードが不要である場合は、これをブランクのままにします。
コマンド	モニターが使用する FTP コマンド: • GET または RECV - ターゲット FTP サーバーからファイルをダウンロードします。 • SEND または PUT - ターゲット FTP サーバーにファイルをアップロードします。 デフォルト: GET
接続タイプ	モニターがファイルの転送を試みるときにサーバーとの間に確立する接続のタイプを指定します。 • Active • Passive デフォルト: Active
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30。
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行できます。 FTP モニターは、検索された内容を各正規表現に突き合わせようとします。

指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

正規表現は、サービスのテスト時にダウンロードされたコンテンツに対してストリングの突き合わせを実行します。これらの表現には 1 つ以上の正規表現演算子が含まれる場合があり、それらの演算子は、表現がどのようなコンテンツと合致するかを決定します。

注 : 正規表現構文を使用して、単一行のみでストリングを突き合わせることができます。Internet Service Monitoring では、改行や復帰を含むストリングの突き合わせはできません。複数の行にわたるストリングを突き合わせるには、複数の正規表現を使用します。また、SLC ルールを使用して、複数の正規表現の結果に基づいてアラームを生成することもできます。

表 *50.* 正規表現演算子

文字	説明
.	任意の単一の文字に合致します。 例えば、正規表現 `r.t` は、ストリング `rat`、`rut`、`r t` に合致しますが、`root` には合致しません。
$	行の末尾に合致します。 例えば、正規表現 `dog$` は、ストリング `it's a dog` の末尾に合致しますが、ストリング `There are a lot of dogs` には合致しません。
^	行の先頭に合致します。 例えば、正規表現 `^When in` は、ストリング `When in the course of human events` の先頭に合致しますが、`What and When in the` には合致しません。
*	すぐ前にある文字の 0 個以上の繰り返しに合致します。 例えば、正規表現 `.*` は、任意の文字が任意の数並んでいる箇所に合致します。
\	後続の文字を通常の文字として扱います。 例えば、`¥$` は、行の末尾ではなくドル記号 ($) に合致します。同様に、`¥.` という正規表現は、任意の単一の文字ではなく、ピリオド文字に合致します。
[]	大括弧内の文字のうちの任意の 1 つに合致します。 例えば正規表現 `r[aou]t` は、`rat`、`rot`、および `rut` に合致しますが、`rit` には合致しません。 文字の範囲を指定するには、ハイフンを使用します。 例えば、正規表現 `[0-9]` は、任意の数字に合致します。 複数の範囲を指定することもできます。 例えば、正規表現 `[A-Za-z]` は、大文字または小文字の任意の文字に合致します。
|	指定された条件のいずれかを含む句に合致します。 例えば `him|her` は、`it belongs to him` という行と `it belongs to her` という行に一致しますが、`it belongs to them` という行には一致しません。

注 : 中括弧 {} または二重引用符 "" が含まれる出力データ・ストリングを使用する場合は、正規表現ですべての中括弧および二重引用符の前にエスケープ文字の円記号 ¥ を追加する必要があります。

例えば、データ・ストリングが `{"templates":true,"mongodb":true,"ldap":true,"ucd":true,"github":true}` である場合、正規表現は `¥{¥"templates¥":true,¥"mongodb¥":true,¥"ldap¥":true,¥"ucd ¥":true,¥"github¥":true¥}` として表示されます。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、FTP モニターは、FTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

表 *51. FTP* モニター・エレメント

エレメント	説明
$bytesPerSec*(BytesPerSec)	1 秒当たりの平均転送バイト数。
$bytesTransfered*(BytesTransferred)	アップロードまたはダウンロードされたバイト数。
$checksum	チェックサム・エレメントは、通常、サービス・レベル分類の有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、 checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$checksum および $previousChecksum モニター・エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$command*(FtpCommand)	モニターが発行した FTP コマンド。
$connectionType*(FtpConnection)	使用されたデータ接続のタイプ。 これには ACTIVE または PASSIVE が入ります。
$connectTime*(ConnectTime)	FTP サーバーに接続するために要した時間。
$downloadTime	ファイルをダウンロードするために要した時間。
$localFile*(FtpLocalFile)	ローカル・ホストに保管されているファイルの絶対パス名。 このエレメントは構成ファイルから取られます。
$lookupTime*(LookupTime)	FTP サーバーの IP アドレスをルックアップするために要した時間。
$previousChecksum	PreviousChecksum エレメントは、通常、サービス・レベル分類の有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$previousChecksum および $checksum モニター・エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$regexp*n*	正規表現。
$regexpMatch*n*	正規表現と合致する行の内容。
$regexpStatus*n*	正規表現の合致の状況: NONE - 正規表現の検査が構成されていません MATCHED - 正規表現に合致するものが見つかりました FAILED - 正規表現に合致するものが見つかりませんでした
$remoteFile*(FtpRemoteFile)	リモート・ホスト (FTP サーバー) に保管されているファイルの絶対パス名。 このエレメントは構成ファイルから取られます。
$responseTime*(ResponseTime)	接続が作成されてからターゲット・ファイルの最初のバイトが受信されるまでに要した時間。
$status	FTP サーバーによって返された状況コード。
$transferTime*(TransferTime)	値を $uploadTime または $downloadTime に設定します。

表 *51. FTP* モニター・エレメント *(*続き*)*	
エレメント	**説明**
$uploadTime	ファイルをアップロードするために要した時間。
$username	モニターがターゲット・ホストにログインするために使用するユーザー名 (アカウント名)。$message に OK が含まれる場合、このエレメントは構成ファイルから取られます。

状況メッセージ

FTP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 *52. FTP* モニターの状況メッセージ	
メッセージ	**説明**
OK	FTP 要求は正常に処理されました。
Unable to open local file for reading/writing	詳しくは、FTP モニターのログ・ファイルを参照してください。
Unable to read from/write to local file	
Unable to read from data connection	FTP サーバーへのデータ接続は確立されましたが、問題が発生しました。
Unable to upload to ftp server	
Timed out waiting to read/write	
Connection closed by foreign host	FTP サーバーへの接続は失敗しました。
Connection closed unexpectedly	
Connection failed	モニターは FTP サーバーへの接続に失敗しました。詳しくは、FTP モニターのログ・ファイルを参照してください。

***HTTP* モニター**

HTTP モニターは、Web サーバーの可用性と応答時間を検査します。

通常はユーザーがデータをフィールドに入力する必要がある、CGI を使用するものも含めて個別の Web ページをモニターすることができます。また、Web ページ上のイメージなどのエレメントのダウンロード時間もモニターすることができます。

表 *53. HTTP* モニター・ファイルの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_http
プロパティー・ファイル	$ISHOME/etc/props/http.props
ルール・ファイル	$ISHOME/etc/rules/http.rules
ログ・ファイル	$ISHOME/log/http.log

HTTP モニターを構成するためのガイドライン

HTTP モニターと HTTPS モニターは、Web サーバーの可用性と応答時間を検査します。以下のシチュエーションでは HTTP モニターを使用してください。

- ターゲット Web サイトが静的である。

動的な Web サイトの場合は、TRANSX モニターを使用してください。

- ターゲット Web サイトのサービスが HTTP プロトコルを介して提供される。

  HTTPS プロトコルを介してコンテンツを配信する Web サイトの場合は、HTTPS モニターを選択してください。
- 複数のプラットフォームにまたがってモニターを実行する。
- 速度が決定要因である (HTTP モニターは高いパフォーマンスを提供します)。

HTTP 要求タイプ

HTTP モニターは、HTTP/1.0 プロトコルをサポートする Web ブラウザーをエミュレートします。このモニターは、Web サーバーをテストするため、以下の HTTP 要求タイプのいずれかを使用して Web ページ要求を Web サーバーに送信します。

- HEAD

  HEAD コマンドは、Web ページにアクセスし、HTTP ヘッダーを返そうとします。HEAD コマンドを発行することで、Web ページがアクセス可能であることを短時間で検査できます。
- GET

  GET コマンドは、Web ページにアクセスし、HTTP ヘッダーを含むページ全体を返そうとします。ページに関連付けられたファイル (イメージなど) を返すことは試みません。
- GETALL

  GETALL コマンドは、Web ページにアクセスし、HTTP ヘッダー、背景、イメージ、アプレット、フレーム、カスケーディング・スタイルシート (CSS) ファイル、およびスクリプトを含むページ全体を返そうとします。HEAD および GET コマンドと同様、このコマンドも Web ページがアクセス可能であることを検査しますが、GETALL コマンドはページ全体とすべての関連ファイルを返すため、ページにアクセスするためにかかる時間が、より実際に近い状態で分かります。また、モニターは、GETALL コマンド実行中に複数のスレッドを使用して、より正確に Web ブラウザーの動作に合致するようにします。
- POST

  POST コマンドは、HTTP フォームを含む Web ページにアクセスし、そのフォームのフィールドに入力しようとします。POST 要求の本文テキストを Internet Service Monitoring エージェント 構成の「**本文**」タブに追加するか、または Internet Service Monitoring エージェント 構成または ismbatch で @Body グループを使用します。あるいは、フォーム・パラメーターを使用できます。本文テキストとフォーム・パラメーターの両方を POST 要求で使用することはできません。

プロキシー・サーバーの使用

プロキシー・サーバーを通じて Web ページの可用性をテストできます。プロキシーを使用するようにモニターを構成すると、モニターはプロキシーを通じて HTTP 要求を送信します。必要な場合は、プロキシー・キャッシュを迂回できます。「**プロキシーの詳細**」タブでプロキシー・サーバーのパラメーターを構成します。 HTTP モニターは、プロキシー・サーバーへの認証されたアクセスをサポートします。この認証は、ターゲット Web ページで要求される認証とは関係ありません。

プロキシー・サーバー・エレメント

V7.2.0.1 より前のバージョンで、プロキシー・サーバーを使用するようにプロファイル・エレメントを構成した場合は、デフォルトで、HTTP モニターは、目的の宛先サーバーの名前およびポートではなく、プロキシー・サーバーの名前およびポートを $server エレメントおよび $port エレメントに挿入します。 V7.2.0.1 より前のバージョンで目的の宛先サーバーの名前およびポートの値を保持するには、generateProxyTokens プロパティーを 1 に設定するか、-generateproxytokens コマンド行パラメーターを指定してモニターを開始します。

このプロパティーまたはコマンド行パラメーターが設定された場合、$server および $port エレメントの値を保存するほか、モニターは、$proxyServer、$proxyPort、$proxyAuthType、$proxyUsername、および $proxyCache エレメントを生成します。

認証

モニターする Web ページまたはテストするプロキシー・サーバーで認証を必要とする場合は、ページにアクセスするための資格情報を、「拡張」タブまたは「プロキシーの詳細」タブの「authenticationtype」、「username」、および「password」の各パラメーター・フィールドに指定します。

認証を使用不可にするには、「authenticationtype」を NONE に設定します。

基本認証を選択するには、以下のようにします。

1. 「authenticationtype」を BASIC に設定します。
2. 「username」および「password」を、Web ページまたはプロキシー・サーバーで必要とされる値に設定します。

NTLM を選択するには、以下のようにします。

1. 「authenticationtype」を NTLMv1 または NTLMv2 に設定します。
2. 「username」および「password」を、Web ページまたはプロキシー・サーバーで必要とされる値に設定します。

注:

モニターは、HTTP 要求の長さを 4096 文字までに制限しています。 追加フォーム・データを含めると要求の長さがこの制限を超えることになる場合、モニターは追加フォーム・データを要求に組み込みません。

HTTP モニター・サービス・テストの構成

HTTP モニター構成パラメーターを使用して、HTTP サービス・テストを定義します。

表 *54. HTTP* モニター構成

フィールド	説明
サーバー	モニターされるサーバーのホスト名。例: www.mycompany.com
ページ	モニターされるページの URL。例: index.html
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: プロキシー・サーバーを経由するモニター
ポート	使用する HTTP サーバー上のポート。 デフォルト: 80
ローカル IP	モニターがテストに使用するネットワーク・インターフェースの IP アドレスを指定します。このフィールドが空の場合、モニターは IpAddress プロパティーによって指定されたインターフェースを 使用します。
バージョン	使用される HTTP プロトコル・バージョン: • 1.0 • 1.1 デフォルト: 1.0

表 54. *HTTP* モニター構成 *(続き)*	
フィールド	**説明**
コマンド	HTTP 要求タイプ: • HEAD • GET • GETALL • POST デフォルト: GET
フォーム名	トランザクションで使用された場合、HTTP モニターは指定されたフォームをスキャンしてデフォルト値を見つけます。検出された値はすべて、トランザクションの次の HTTP ステップで自動的に完了されます。
認証タイプ	ネットワーク・ユーザーの認証用の Challenge/Response 認証メカニズムを指定します。 • NONE - 認証を行わない • BASIC • NTLMv1 - Windows NTLM バージョン 1 の Challenge/Response 認証 • NTLMv2 - Windows NTLM バージョン 2 デフォルト: NONE
ユーザー名	モニターがサーバーにログインするために使用するユーザー名 (アカウント名)。
パスワード	モニターがサーバーにログインするために使用するユーザー名に対応するパスワード。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300 .
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
証明書の検証	デフォルトでは無効です。
プロキシーの詳細	
サーバー	プロキシー・サーバーのホスト名。
ポート	使用するプロキシー・サーバー上のポート。 デフォルト: 8080

表 54. HTTP モニター構成 (続き)	
フィールド	**説明**
認証タイプ	プロキシー・サーバーのサーバー認証タイプ。 詳しくは、前の authenticationtype を参照してください。 デフォルト: NONE
ユーザー名	モニターがプロキシー・サーバーにログインするためにパスワードと共に使用します。
パスワード	モニターがプロキシー・サーバーにログインするためにユーザー名と共に使用します。ラベルはパスワードです。
プロキシーの使用	プロキシー・サーバーを使用して要求を実行するようにモニターを構成します。 • proxy (ismbatch で true を使用します) • noproxy (ismbatch で false を使用します) デフォルト: noproxy
ホスト名ルックアップ設定	どちらの IP バージョン (IPv6 または IPv4) が、提供されたホスト名に適用されるか判別します。オプションは以下のとおりです。 • default を指定すると、モニター全般用に設定されているプロパティーを使用するように、モニターが設定されます。これはデフォルトです。 • 4Then6 を指定すると、IPv4 の次に IPv6 が選択されます。IPv4 アドレスが使用可能な場合は、IPv4 アドレスが使用されます。IPv4 アドレスが見つからない場合は、IPv6 アドレスが使用されます。 • 6Then4 を指定すると、IPv6 の次に IPv4 が選択されます。IPv6 アドレスが使用可能な場合は、IPv6 アドレスが使用されます。IPv6 アドレスが見つからない場合は、IPv4 アドレスが使用されます。 • 4Only を指定すると、IPv4 のみが選択されます。IPv4 アドレスのみが使用されます。IPv4 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Only を指定すると、IPv6 のみが選択されます。IPv6 アドレスのみが使用されます。IPv6 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Or4 を指定すると、IPv4 または IPv6 が選択されます。ホスト名から最初に返されたアドレスが使用されます。
キャッシュなし	デフォルトでは cache に設定されます。

正規表現

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行できます。 HTTP モニターは、検索された内容を各正規表現に突き合わせようとします。指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。正規表現構文については、333 ページの表 50 を参照してください。

ヘッドおよびフォームのパラメーター

HTTP モニターは、HTTP 要求のヘッダー・フィールドおよびメッセージ本体に追加データを入れて送信することができます。

「パラメーター」タブでこの追加データのパラメーターを構成します。 パラメーターは、「名前」、「値」、および「タイプ」であり、これらは以下のように機能します。

- タイプが HEAD の「名前」と「値」のペアは、User-Agent および Referer などの追加ヘッダー・フィールドを指定します。これらのフィールドはモニターが送信するすべての HTTP 要求に組み込まれます。ヘッダー・フィールドは、どのタイプの HTTP メソッド (GET、GETALL、HEAD、または POST) に対しても指定できます。

  ITCAM for Transactions V7.4.0.1 以降では、デフォルトのユーザー・エージェント HEAD パラメーターの Mozilla/5.0 (ISM-MONITOR) が、すべての新規 HTTP エレメントまたは HTTPS エレメントに対して追加されます。デフォルト User-Agent ヘッダーは、HTTP モニターおよび HTTPS モニターを、ブラウザー・クライアントに基づいてコンテンツを切り替える Web サイトに使用できるようにするためのものです。
- タイプが FORM の名前と値のペアは、モニターが送信する HTTP POST 要求のメッセージ本体に組み込まれる追加データを指定します。ターゲット・ページに、「formname」フィールドで指定された名前に合致するフォームが含まれる場合、モニターはフォーム内のすべての名前と値のペアを、プロファイル・エレメントで構成されているかのように扱います。

注:

モニターは、HTTP 要求の長さを 4096 文字までに制限しています。 追加フォーム・データを含めると要求の長さがこの制限を超えることになる場合、モニターは追加フォーム・データを要求に組み込みません。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、HTTP モニターは、HTTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 *55. HTTP* モニター・エレメント

エレメント	説明
$bytesPerSec*(BytesPerSec)	1 秒当たりの平均転送バイト数。
$bytesTransfered*(BytesTransferred)	アップロードまたはダウンロードされたバイト数。
$checksum	チェックサム・エレメントは、通常、サービス・レベル分類の 有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、 checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を 計算します)。$checksum および $previousChecksum モニター・ エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$command	モニターが発行した HTTP コマンド。例えば、HEAD、GET、GETALL、または POST です。
$connectTime*(ConnectTime)	サーバーに接続するために要した時間。
$downloadTime*(DownloadTime)	ファイルをダウンロードするために要した時間。
(Elements)	受信したページ・エレメントの数。
$formname	POST アクションで使用するフォームの名前。

表 55. *HTTP* モニター・エレメント *(続き)*	
エレメント	**説明**
$lastStatus*(PageStatus)	プロファイル・エレメントが複数のページを取得する場合、このエレメントには取得された最後のページの 結果ストリングが格納されます。この値は、$urlResult*n* (*n* は $pageCount の値に等しい) の値と同じです。
$lastModified	取得された最初のページの Last-Modified HTTP ヘッダー・フィールドの値。
$page*(Page)	HTTP サーバーでアクセスされているページ。
$pageCount	GETALL テスト中にダウンロードされたリソースの総数 (テスト・ページ自体は除く)。テスト・ページが他のリソースを参照しない場合、 このエレメントは生成されません。
$port*(Port)	HTTP サーバーへのアクセスに使用されたポート。 テストでプロキシー・サーバーを使用した場合、これは、 要求が送信されたプロキシー・サーバー上のポートの 値です。目的の宛先サーバーのポートを保存するには、 generateProxyTokens プロパティーを 1 に設定するか、 -generateproxytokens コマンド行パラメーターを使用してモニターを開始します。
$previousChecksum	PreviousChecksum エレメントは、通常、サービス・レベル分類の 有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、 checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$previousChecksum および $checksum モニター・ エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$proxyAuthType	プロキシー・サーバーのサーバー認証タイプ。
$proxyCache	値 true は、プロキシー・サーバーが独自のキャッシュではなくサーバーから Web ページを取得したことを示します。
$proxyPort	要求が実行依頼されるプロキシー・サーバーのポート番号。
$proxyServer	プロキシー・サーバーのホスト名。
$proxyUsername	モニターがプロキシー・サーバーにログインするためにパスワードと共に使用します。
$regexpMatch*n*	正規表現と合致する行の内容。
$regexp*n*	正規表現。
$regexpMatch*n*	正規表現と合致する行の内容。
$regexpStatus*n*	正規表現の合致の状況: NONE - 正規表現の検査が構成されていません MATCHED - 正規表現に合致するものが見つかりました FAILED - 正規表現に合致するものが見つかりませんでした
$responsetime*(ResponseTime)	接続が作成されてからページの最初のバイトが受信されるまでに要した時間。

表 55. *HTTP* モニター・エレメント (続き)	
エレメント	**説明**
$timeSinceModification	ページの最終変更日時から経過した時間。 これは、テストの時刻と、取得された最初のページの Last-Modified HTTP ヘッダー・フィールドの値の差です。
$urlDownloadTimes*n** (UrlDownloadTime)	GETALL 要求の各エレメントの URL ダウンロード時間。 各エレメントには、000 から始まる番号が付けられます ($urlDownloadTime000、$urlDownloadTime001、$urlDownloadTime002 など)。
$url*n**(Url)	GETALL テストの各ページの URL。 各ページには、000 から始まる番号が付けられます ($url000、$url001、$url002 など)。
$urlResult*n** (UrlResultString)	GETALL 要求でダウンロードされた各ページの結果ストリング。 各結果には、000 から始まる番号が付けられます ($urlResult000、$urlResult001、$urlResult002 など)。
$username	ユーザーを認証する必要があるページにアクセスするために使用する名前。

状況メッセージ

HTTP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 56. 表 4. *HTTP* モニターの状況メッセージ	
メッセージ	**説明**
OK	モニターからの要求は正常に実行されました。
Form not found	要求されたページが見つかりません。
Initialise Fetch Page Failed	HTTP ページのフェッチ・メカニズム用にスペースを割り振るのに十分なメモリーがありません。直前の行メッセージに詳細情報が記載されています。
Connection Failed	リンクがダウンした、接続がリセットされた、リンクが到達不能、接続がタイムアウトになった、接続が終了した、またはホストがダウンしたという理由以外の理由で、モニターが接続に失敗しました。詳しくは、HTTP モニターのログ・ファイルを参照してください。

フォーム・パラメーターおよび正規表現の突き合わせ

フォーム・パラメーター search=ism と共に HTTP POST 要求を送信することによって、フォーム http://support.mycompany.com/cgi-bin/search.cgi の操作をモニターし、正規表現を使用して応答内のストリング Your search was successful と突き合わせます。応答内にこのストリングが返されている場合は、サービス・レベルを「Good」として分類し、それ以外の場合は「Failed」と分類します。

新しい HTTP モニター・プロファイル・エレメントを作成し、各フィールドを次の表に示すように設定します。

表 57. *HTTP* フォーム・プロファイル・エレメントの例	
プロファイル・エレメント構成フィールド	**値**
サーバー	support.mycompany.com
ページ	/cgi-bin/search.cgi
説明	Example - form parameters and regular expressions
正規表現の詳細	
match 1	Your search was successful
サービス・レベル分類の詳細	
statement	Regexp Status 1 = MATCHED then status GOOD
ヘッドおよびフォームの詳細	
名前	search
value	ism
type	FORM

プロパティー

次の表では、HTTP モニターに固有のプロパティーおよびコマンド行オプションについて説明しています。

表 58. *HTTP* モニター・プロパティー		
プロパティー名	**プロパティー・パラメーター**	**説明**
AllowDuplicateDownload	0|1	ページが検出されるたびに強制的にダウンロードされるようにします。 0 - 無効 (1 回のみダウンロード) 1 - 有効
ForceHTMLParse	0|1	コンテンツ・タイプ text/html がないページが強制的に HTML として構文解析されるようにします。 0 - 無効 1 - 有効
GenerateProxyTokens	0|1	テストでプロキシー・サーバーが使用される場合に、モニターがプロキシー・サーバーに関する情報が入った追加エレメントを生成するかどうかを指定します。 0 - 無効 1 - 有効 (追加エレメントの $server および $port にはプロキシー・サーバーの値が含まれます)
GETALLThreadNum	1|2|3|4|5	GETALL 要求で使用する個別のスレッドの数を指定します。

表 58. *HTTP* モニター・プロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
GetLinkTags	0|1	GETALL 要求のリンク・スタイルシートのダウンロードを活動化します。 0 - 無効 1 - 有効 (ターゲット・ページに link タグが含まれ、属性値 rel=stylesheet が指定されている場合、モニターは link タグの href 属性によって参照されるリソースをダウンロードしようとします)
HostnameLookupPreference	ストリング	提供されたホスト名に適用する IP バージョン (IPv6 または IPv4) を判別します。指定できる値は、以下のとおりです。 • 4Then6 を指定すると、IPv4 の次に IPv6 が選択されます。IPv4 アドレスが使用可能な場合は、IPv4 アドレスが使用されます。IPv4 アドレスが見つからない場合は、IPv6 アドレスが使用されます。 • 6Then4 を指定すると、IPv6 の次に IPv4 が選択されます。IPv6 アドレスが使用可能な場合は、IPv6 アドレスが使用されます。IPv6 アドレスが見つからない場合は、IPv4 アドレスが使用されます。 • 4Only を指定すると、IPv4 のみが選択されます。IPv4 アドレスのみが使用されます。IPv4 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Only を指定すると、IPv6 のみが選択されます。IPv6 アドレスのみが使用されます。IPv6 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Or4 を指定すると、IPv4 または IPv6 が選択されます。ホスト名から最初に返されたアドレスが使用されます。 デフォルト: 4Then6
Ipv6Address	整数	HTTP IPv6 を使用する場合に HTTP 要求の発信元としてバインドするローカル・アドレス。 デフォルト: アドレスなし
NoParseExtensions	ストリング	モニターが構文解析せずに単にダウンロードするファイル・タイプを示すファイル拡張子のコンマ区切りリスト。
OutputDirectory	ストリング	OutputResult が真である (1 に設定される) 場合に使用する出力ディレクトリーを指定します。 デフォルト: $ISHOME/var

表 58. *HTTP* モニター・プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
OutputResult	0|1	モニターがサービスから受信するデータを保管するかどうかを指定します。 0 - 無効 1 - 有効
RelativeRedirects	0|1	HTTP 301 および HTTP 302 の状況コードのロケーション・フィールドで絶対 URL の代わりに相対 URL を使用することを許可します。 0 - 絶対 URL 1 - 相対 URL
RFCPOST	0|1	モニターが RFC1945 および RFC2616 に従い、リダイレクトの後で 2 番目の POST を送信することを指定します。多くの Web サーバーはリダイレクトの後で POST を予期せず、大半のブラウザーは RFC に従っていません。 0 - 無効 1 - 有効

HTTPS モニター

HTTPS モニターは、Web サーバーの可用性と応答時間を検査します。通常はユーザーがデータをフィールドに入力する必要がある、HTML 形式を使用するものも含めて個別の Web ページをモニターすることができます。

注 : HTTPS モニターは HTTP モニターと同じ方法で機能しますが、HTTP サーバーとの通信に、サーバーとモニターの間のすべての通信を暗号化する SSL (Secure Sockets Layer) プロトコルのバージョン 2 またはバージョン 3 を使用します。

表 59. *HTTPS* モニター・ファイルの概要	
モニター・ファイル。	**名前またはロケーション**
モニター実行可能ファイル	nco_m_https
プロパティー・ファイル	$ISHOME/etc/props/https.props
ルール・ファイル	$ISHOME/etc/rules/https.rules
ログ・ファイル	$ISHOME/log/https.log

HTTPS モニターを構成するためのガイドライン

HTTPS モニターは、Web サーバーの可用性と応答時間を検査します。以下のシチュエーションでは HTTPS モニターを使用してください。

- ターゲット Web サイトが静的である。

  動的な Web サイトの場合は、TRANSX モニターを使用してください。
- ターゲット Web サイトのサービスが HTTPS プロトコルを介して提供される。

  HTTP プロトコルを介してコンテンツを配信する Web サイトの場合は、HTTP モニターを選択してください。

- 複数のプラットフォームにまたがってモニターを実行する。
- 速度が決定要因である (HTTPS モニターは高いパフォーマンスを提供します)。

クライアント・サイド証明書

このモニターを使用すると、相互認証のためにクライアント・サイド証明書を必要とするサーバーをモニターできます。

プロファイル・エレメントの作成時に、SSL 証明書ファイル、鍵格納ファイル、および鍵パスワードを指定します。

証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。証明書が別の形式である場合は、PEM 形式に変換する必要があります。証明書は openSSL などのソフトウェアを使用して変換できます。openSSL は、http://www.openssl.org から入手できます。

注 : すべてのプロファイル・エレメントで常に同じ証明書、鍵、およびパスワードを使用する場合は、作成するプロファイル・エレメントごとにそれらを定義するのではなく、モニター・プロパティーを使用して指定してください。

HTTPS モニター・サービス・テストの構成

HTTPS モニター構成パラメーターを使用して、HTTPS サービス・テストを定義します。

表 *60. HTTPS* モニター構成

フィールド	説明
サーバー	モニターされるサーバーのホスト名。例: `www.myconpany.com`
ページ	モニターされるページの URL。例: `/secure/`
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	使用するサーバー上のポート。 デフォルト: 443
ローカル IP	モニターがテストに使用するネットワーク・インターフェースの IP アドレスを指定します。このフィールドが空の場合、モニターは `IpAddress` プロパティーによって指定されたインターフェースを使用します。
バージョン	使用される HTTPS プロトコル・バージョン: • `1.0` • `1.1` デフォルト: `1.0`
コマンド	要求タイプ: • `HEAD` • `GET` • `GETALL` • `POST` デフォルト: `GET`
フォーム名	トランザクションで使用された場合、HTTPS モニターは指定されたフォームをスキャンしてデフォルト値を見つけます。検出された値はすべて、トランザクションの次の HTTPS ステップで自動的に完了されます。

表 60. HTTPS モニター構成 (続き)	
フィールド	**説明**
認証タイプ	ネットワーク・ユーザーの認証用のチャレンジ応答認証メカニズムを指定します。 • NONE - 認証を行わない。 • BASIC • NTLMv1 - Windows NTLM バージョン 1 の Challenge/Response 認証。 • NTLMv2 - Windows NTLM バージョン 2。 デフォルト: NONE
ユーザー名	モニターが HTTPS サーバーにログインするために使用するユーザー名 (アカウント名)。
パスワード	モニターが HTTPS サーバーにログインするために使用するユーザー名に対応するパスワード。
SSL 証明書ファイル	モニター・エレメントで使用されるデジタル証明書ファイルのパスおよびファイル名。パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー (`$ISMHOME/platform/arch/bin`) からの相対位置にあるものと解釈します。 証明書ファイルを指定しない場合、モニターは、モニター・プロパティー `SSLCertificateFile` によって指定された証明書を使用します。
SSL 鍵ファイル	サーバーを識別して SSL メッセージに署名するために使用される SSL 秘密鍵を含むファイルの、パスおよびファイル名。
SSL 鍵パスワード	SSL 秘密鍵の暗号化に使用されるパスワード。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
プロキシーの詳細	
サーバー	プロキシー・サーバーのホスト名。
ポート	使用するプロキシー・サーバー上のポート。
認証タイプ	プロキシー HTTPS サーバーのサーバー認証タイプ。 詳しくは、authenticationtype を参照してください。

表 60. HTTPS モニター構成 (続き)	
フィールド	**説明**
ユーザー名	モニターがプロキシー HTTPS サーバーにログインするために使用するユーザー名。
パスワード	モニターがプロキシー HTTPS サーバーにログインするために使用するパスワード。
プロキシーの使用	プロキシー・サーバーを使用して要求を実行するようにモニターを構成します。 • proxy (ismbatch で true を使用します) • noproxy (ismbatch で false を使用します) デフォルトは noproxy です。
ホスト名ルックアップ設定	どちらの IP バージョン (IPv6 または IPv4) が、提供されたホスト名に適用されるか判別します。オプションは以下のとおりです。 • default を指定すると、モニター全般用に設定されているプロパティーを使用するように、モニターが設定されます。これはデフォルトです。 • 4Then6 を指定すると、IPv4 の次に IPv6 が選択されます。IPv4 アドレスが使用可能な場合は、IPv4 アドレスが使用されます。IPv4 アドレスが見つからない場合は、IPv6 アドレスが使用されます。 • 6Then4 を指定すると、IPv6 の次に IPv4 が選択されます。IPv6 アドレスが使用可能な場合は、IPv6 アドレスが使用されます。IPv6 アドレスが見つからない場合は、IPv4 アドレスが使用されます。 • 4Only を指定すると、IPv4 のみが選択されます。IPv4 アドレスのみが使用されます。IPv4 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Only を指定すると、IPv6 のみが選択されます。IPv6 アドレスのみが使用されます。IPv6 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Or4 を指定すると、IPv4 または IPv6 が選択されます。ホスト名から最初に返されたアドレスが使用されます。

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行できます。 HTTPS モニターは、検索された内容を各正規表現に突き合わせようとします。

指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

詳しくは、333 ページの表 50 を参照してください。

ヘッドおよびフォームのパラメーター

HTTPS モニターは、HTTP モニターと同様に、HTTP 要求のヘッダー・フィールドおよびメッセージ本文に追加データを入れて送信することができます。

ヘッド・パラメーターとフォーム・パラメーターについて詳しくは、『ヘッド・パラメーターおよびフォーム・パラメーター』を参照してください。

モニター・エレメント

表 *61. HTTPS SSL* モニター・エレメント

エレメント	説明
$SSLcertificateSerialNumber	サーバーによって提示される X509 証明書のシリアル番号。
$SSLcipherSuiteCount	接続で使用できる暗号スイートの数。
$SSLcipherSuiteList	接続で使用できる暗号スイートのリスト。
$SSLcipherSuiteName	接続用に選択された暗号スイート。
$SSLeffectiveSessionKeyBits	セッション鍵のビット数。通常は 128 または 168 で、エクスポート・バージョンの場合は 40 です。
$SSLHandshakeTime* (SslHandshakeTime)	SSL 接続を確立するために要した時間。
$SSLissuerName	サーバーの X509 形式の認証の発行者名。
$SSLprotocolVersion	使用されている SSL のバージョン (v2 または v3)。
$SSLpublicKeyLengthBits	サーバーの公開鍵のサイズ。通常は 1024 ビットですが、エクスポート仕様の暗号スイートが使用される場合は除きます。
$SSLserverCertificateValidFrom	サーバー証明書が有効になる日付。
$SSLserverCertificateValidTo	サーバー証明書が有効である最後の日付。
$SSLserverName	SSL サーバー名。
$SSLsubjectName	X509 形式の認証のサブジェクト名。 通常は、サーバーを制御する組織の名前です。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

HTTPS モニターは、HTTP モニターと同じ追加エレメントを作成します (340 ページの表 55 を参照)。さらに、クライアント・サイド証明書がテストで使用される場合は SSL に関連するエレメントも作成します (349 ページの表 61 を参照)。

すべてのエレメントに共通するテスト結果に加えて、HTTPS モニターは、HTTPS サービス・テストに固有のデータを含む一連のテスト結果を生成します。

状況メッセージ

HTTPS モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。 これらのメッセージは、テストの結果を示します。

HTTP 状況メッセージに加えて、HTTPS モニターは 349 ページの表 62 にリストされているメッセージも生成します。

表 *62. HTTPS* モニターの状況メッセージ

メッセージ	説明
OK	モニターは正常にサーバーに接続しました。
SSL handshake failed	モニターは、サーバーへの接続を確立した後で SSL 接続を初期化できませんでした。

表 62. *HTTPS* モニターの状況メッセージ (続き)	
メッセージ	**説明**
Connection failed	リンクがダウンした、接続がリセットされた、リンクが到達不能、接続がタイムアウトになった、接続が終了した、またはホストがダウンしたという理由以外の理由で、モニターが接続に失敗しました。詳しくは、HTTP モニターのログ・ファイルを参照してください。

プロパティー

HTTPS モニターのプロパティーは、HTTP モニターと同じです。

HTTP モニターと同じプロパティー・オプションについて詳しくは、343 ページの表 58 を参照してください。表 5 に、その他の HTTPS に固有のプロパティーをリストしています。

表 63. *HTTPS* モニター固有のプロパティー		
プロパティー名	**プロパティー・パラメーター**	**説明**
SSLCertificate File	ストリング	HTTPS エレメントの作成時に証明書が明示的に指定されなかった場合に使用されるデジタル証明書のパスおよびファイル名。 パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー (`$ISHOME/platform/arch/bin`) からの相対位置にあるものと解釈します。
SSLCipherSuite	ストリング	SSL 操作で使用する暗号スイート。 デフォルト: `RC4:3DES:DES:+EXP`
SSLDisableTLS	整数	ヘリテージ・サポートで TLSv1 を使用不可にします。 デフォルト: `0` - TLSv1 を使用可能にします。 `1` - TLSv1 を使用不可にします。
SSLKeyFile	ストリング	SSL 秘密鍵が入ったファイル。
SSLKeyPassword	ストリング	SSL 秘密鍵の暗号化に使用されるパスワード。

暗号スイート

SSLCipherSuite プロパティーは、HTTPS モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』 を参照してください。

ICMP モニター

ICMP モニターは、ネットワーク上で実行されている Internet Control Message Protocol サービスのパフォーマンスをテストします。そのために、モニターは ICMP エコー・コマンドを使用します。

以下の表に、ICMP モニター・ファイルをリストしています。

表 64. *ICMP* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	`nco_m_icmp`
プロパティー・ファイル	`$ISHOME/etc/props/icmp.props`
ルール・ファイル	`$ISHOME/etc/rules/icmp.rules`

表 *64. ICMP* モニター・ファイル (続き)	
モニター・ファイル	**名前またはロケーション**
ログ・ファイル	$ISHOME/log/icmp.log

ICMP モニターを構成するためのガイドライン

ICMP モニターは、ICMP エコー要求 (一般に ping と呼ばれる) をターゲット・ホストに対して発行し、エコー応答を待機します。ICMP モニターは、ネットワークのパフォーマンスを示すルックアップ時間、往復時間、および成功率のメトリックを記録します。モニターがエコー要求を発行すると、要求は、ターゲット・ホストに到達する前に 1 つ以上のルーターを通過する場合があります。これらのルーターは、ターゲット・ホストがエコー要求を受信する前にモニターに応答することができます。モニターが発行したエコー要求がルーターを通過した場合、ルーターがモニターへの応答を発行することがあります。この応答は、ルーターがターゲット・ホストを見つけることができないこと、またはルーターがビジーであるため要求を処理できないことを示します。モニターがターゲット・ホストからエコー応答を受信する前に、複数のルーターから応答を受信する可能性があります。モニターがターゲット・ホストから正常にエコー応答を受信した場合、モニターはかかった時間を記録します。モニターが、指定されたタイムアウト期間内にターゲット・サーバーからの応答を受信しなかった場合、要求は失敗したものとして記録されます。各テストにおいて、同じターゲットに複数の ICMP エコー要求を送信するようにモニターを構成することができます。モニターは、送信された各要求の統計を記録します。

注 : ICMP モニターは、ロー・ソケットを開いて ICMP パケットを送信するため、root として実行します。

ICMP モニター・サービス・テストの構成

ICMP モニター構成パラメーターを使用して、サービス・テストを定義します。モニターの構成時に、タイムアウト・パラメーターとポーリング間隔パラメーターのデフォルト値が表示されます。タイムアウトのデフォルトは 30 秒、ポーリング間隔のデフォルトは 300 秒です。以下の表にリストされたその他のデフォルトは、構成時には表示されませんが、値を指定しなかった場合に構成の詳細を保存した時点で適用されます。

以下の表に、ICMP モニター構成をリストしています。

表 *65. ICMP* モニター構成	
フィールド	**説明**
サーバー	エコー要求の送信先となるサーバーのホスト名または IP アドレス。例: test.myconpany.com
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
timeout	サーバーが各エコー要求に応答するのを待機する時間 (秒単位)。 デフォルト: 10
numberofpings	送信するエコー要求の数。 デフォルト: 5
packetinterval	パケットを送信してから次にエコー要求を送信するまで待機する時間 (秒単位)。 デフォルト: 1
packetsize	送信された各エコー要求のサイズ (バイト単位)。 デフォルト: 64

表 65. ICMP モニター構成 (続き)	
フィールド	**説明**
サービス・タイプ	IP レイヤーで Type of Service フィールドを設定します。IPv4 スタイルの Type Of Service (TOS) の値と、DSCP の DiffServ フィールドの値の両方を入力できます。有効な値は、0 から 255 です。
再試行	モニターが各エコー要求を再試行する回数。この回数だけ再試行すると、モニターは終了します。 デフォルト: 0
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
ホスト名ルックアップ設定	どちらの IP バージョン (IPv6 または IPv4) が、提供されたホスト名に適用されるか判別します。オプションは以下のとおりです。 • default を指定すると、モニター全般用に設定されているプロパティーを使用するように、モニターが設定されます。これはデフォルトです。 • 4Then6 を指定すると、IPv4 の次に IPv6 が選択されます。IPv4 アドレスが使用可能な場合は、IPv4 アドレスが使用されます。IPv4 アドレスが見つからない場合は、IPv6 アドレスが使用されます。 • 6Then4 を指定すると、IPv6 の次に IPv4 が選択されます。IPv6 アドレスが使用可能な場合は、IPv6 アドレスが使用されます。IPv6 アドレスが見つからない場合は、IPv4 アドレスが使用されます。 • 4Only を指定すると、IPv4 のみが選択されます。IPv4 アドレスのみが使用されます。IPv4 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Only を指定すると、IPv6 のみが選択されます。IPv6 アドレスのみが使用されます。IPv6 アドレスがない場合は、ポーリングでエラーが返されます。 • 6or4 を指定すると、IPv4 または IPv6 が選択されます。ホスト名から最初に返されたアドレスが使用されます。

注: 10 分間隔で応答時間をチェックすることによって、ホスト test.mycompany.com の可用性をモニターします。30 秒以内にサーバーへの接続を試み、タイムアウトした場合は、さらに 2 回再試行します。それでも失敗する場合は、再試行の間隔を 5 秒にしてテストを 3 回繰り返します。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、ICMP モニターは、ICMP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、ICMP モニターの追加エレメントについて説明します。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、エレメントの下に括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 66. *ICMP* モニター・エレメント	
エレメント	**説明**
$averageRTT*(Average RTT)	平均往復時間 (秒)。
$endTime	応答を受信した UNIX 時刻。
$jitter	2 つの隣接 ICMP エコー要求の着信時刻の差異からそれらの発信時刻を引いた絶対値。この値は、RFC2598 で指定された数式に従って計算されます。エレメントは、エコー要求数が 1 より大きい場合にのみ作成されます。3 つ以上のエコー要求が使用される場合、この値は、すべてのエコー要求ペアの平均ジッターになります。
$lookupTime*(LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$maxRTT*(MaxRTT)	最大往復時間 (秒)。
$minRTT* (MinRTT)	最小往復時間 (秒)。
$numberPackets	プロファイル・エレメントで指定された送信済み ICMP エコー要求数。
$packetInterval	プロファイル・エレメントで指定された、各 ICMP エコー要求を送信する間隔の時間。
$packetRetries	モニターが終了する前に ICMP エコー要求の再送信を試行する回数。
$packetSize	プロファイル・エレメントで指定された各 ICMP エコー要求のサイズ (バイト単位)。
$pingAttempts Failed	最初に失敗した ICMP エコー要求について行われた再試行回数。
$pingAttempts Responded	最初に正常終了した ICMP エコー要求について行われた再試行回数。
$pingMessageFailed	最初に失敗した ICMP エコー要求について返されたメッセージ。
$pingMessage Responded	最初に正常終了した ICMP エコー要求について返されたメッセージ。
$pingReceivedTime Failed	最初に失敗したエコー応答を受信した UNIX 時刻。
$pingReceivedTime Responded	最初に正常終了したエコー応答を受信した UNIX 時刻。
$pingRespondIP Failed	最初に失敗した ICMP エコー要求に応答した IP アドレス。
$pingRespondIP Responded	最初に正常終了した ICMP エコー要求に応答した IP アドレス。
$pingRTTFailed	最初に失敗した ICMP エコー要求の往復時間 (秒単位)。
$pingRTTResponded	最初に正常終了した ICMP エコー要求の往復時間 (秒単位)。
$pingSentTime Failed	最初に失敗した ICMP エコー要求が送信された UNIX 時間。
$pingSentTime Responded	最初に正常終了した ICMP エコー要求が送信された UNIX 時間。
$pingsFailed	エコー応答がなかった場合に送信された ICMP エコー要求の数。

表 66. *ICMP* モニター・エレメント (続き)	
エレメント	**説明**
$pingsResponded	エコー受信した有効な応答の数。
$pingTime	ICMP エコー要求を送信してからエコー応答を受信するまでの時間。
$respondPercent* (RespondPercent)	送信された ICMP エコー要求で応答があったものの割合 (パーセント)。
$responseTime	ターゲット・ホストが ICMP エコー要求に応答するために要した時間。
$sentTime	ICMP エコー要求が送信された UNIX 時刻。
$spreadRTT	$maxRTT と $minRTT の差異。
$startTime	テストが開始された UNIX 時刻。
$totalHostTime	テストを開始してからエコー応答を受信するまでの時間。
$typeOfService	新規 ICMP エレメントの追加時に指定された、IP レイヤーの Type of Service フィールド。詳しくは、350 ページの『ICMP モニター』を参照してください。

ICMP モニターは、テスト中に送信された各 ICMP エコー要求の結果を記録するために、別々の $pingname エレメント・セットを作成します。送信された要求数は、$numberPackets で示されます。例えば、$pingRTT エレメントで、$numberPackets が 3 である場合、3 つの送信済み ICMP エコー要求の往復時間測定が入った 3 つのエレメント ($pingRTT1、$pingRTT2、および $pingRTT3) を作成します。

状況メッセージ

ICMP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、ICMP 状況メッセージについて説明します。

表 67. *ICMP* モニターの状況メッセージ	
メッセージ	**説明**
Pings Complete	ICMP エコー要求は正常に処理されました。
ICMP echo failed	モニター・ホストまたはネットワークへの接続に問題があるため、モニターは ICMP エコー要求を発行できませんでした。
Timed out	ICMP エコー要求はタイムアウトになりました。
Unreachable	このメッセージはルーターから返され、必ずしも正確ではありません。
Source quench	ルーターがビジーであるため、ICMP エコー要求を処理できません。
Time exceeded	このメッセージはルーターから返されます。ICMP エコー要求がネットワーク経由で転送された回数が多すぎることを示しています。
Parameter problem	このメッセージはルーターから返されます。ルーターが ICMP エコー要求を処理できないことを示しています。メッセージが壊れていることが原因として考えられます。

プロパティー

以下の表では、ICMP モニターに固有のプロパティーについて説明します。

表 *68. ICMP* プロパティー

プロパティー名	プロパティー・パラメーター	説明
EventsPerSec	適用外	このプロパティーはサポートされていません。
IntraPingWait	整数	ICMP モニターによって送信されたすべての ping 間の最小時間間隔 (ミリ秒単位)。長期間にわたるネットワーク・トラフィックを分散するようにシステムを調整する場合に使用します。例えば、多数のターゲット ICMP ホストが存在する環境では、IntraPingWait を 3 に設定します。 デフォルト: 0
Ipv6Address	整数	ICMP IPv6 を使用する場合に ICMP エコー要求の発信元としてバインドするローカル・アドレス。 デフォルト: アドレスなし
MaxDNSResolvingThreads	整数	DNS リゾルバーが使用するスレッドの最大数。 デフォルト: 20
MaxPacketSize	整数	最大 ICMP パケット・サイズ (バイト単位)。
PingsPerSec	整数	モニターが 1 秒当たりに送信しようとするエコー要求の数。実際に送信された要求の数は、CPU およびネットワーク・ロードによって異なります。 デフォルト: 100
SocketBufferSize	整数	受信するソケット・バッファーのサイズ (キロバイト単位)。 デフォルト: 32

***LDAP* モニター**

LDAP モニターは、Lightweight Directory Access Protocol (LDAP) サーバーの操作をテストします。

以下の表に、LDAP モニター・ファイルをリストしています。

表 *69. LDAP* モニター・ファイル

モニター・ファイル	名前またはロケーション
モニター実行可能ファイル	nco_m_ldap
プロパティー・ファイル	$ISHOME/etc/props/ldap.props
ルール・ファイル	$ISHOME/etc/rules/ldap.rules
ログ・ファイル	$ISHOME/log/ldap.log

LDAP モニターを構成するためのガイドライン

LDAP モニターは、LDAP サーバーに接続し、特定のエントリーを見つけようとすることによって、LDAP サービスをテストします。サーバーは、エントリーを見つけることに成功すると、エントリーの内容をモニターに返します。LDAP モニターは、SSL を使用して認証を行い、LDAP サーバーに接続できます。

LDAP モニターを構成するには、LDAP プロトコルとモニター対象のディレクトリー・サービスがどのように機能するかを理解する必要があります。LDAP は、ディレクトリー・サービスにアクセスして管理するためのインターネット・プロトコルです。ディレクトリー・サービスは、分散データベース・アプリケーションです。ディレクトリーは、エントリーから構成されています。例えば、組織の従業員またはリソースに関連するエントリーを、ディレクトリーに格納することができます。各エントリーには、属性セットが含まれています。例えば、従業員のディレクトリー内のエントリーには、従業員の名前、電話番号、および住所が含まれている場合があります。

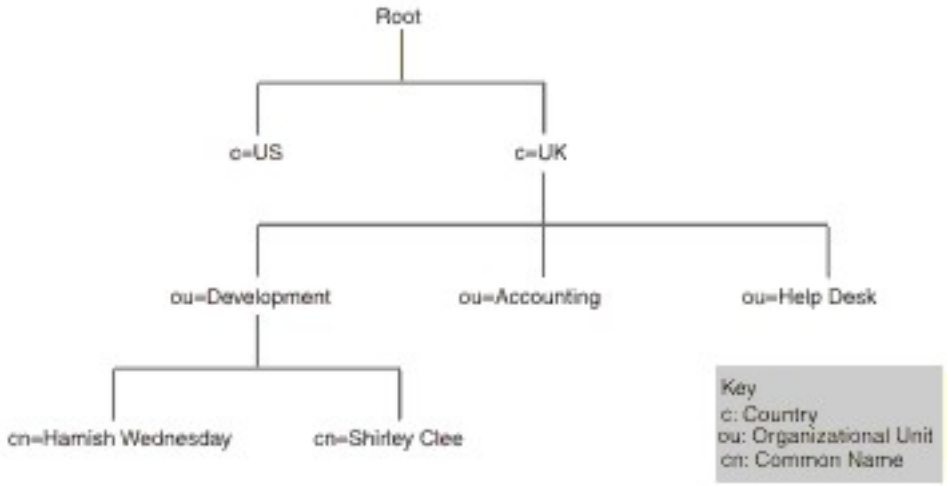
個々のディレクトリー・サービスは、異なった構成になっている場合があるため、モニター手順も異なることがあります。

LDAP バージョン

LDAP モニターは、LDAP のバージョン 2 および 3 の両方をサポートします。デフォルトでは、モニターはバージョン 3 を使用してターゲット LDAP サーバーに接続を試みます。それが失敗した場合は、自動的にバージョン 2 に戻ります。**NOLDAPV3** プロパティーを設定することによって、モニターが常にバージョン 2 を使用するように強制することができます。

サンプル・ディレクトリー・サービス

このサンプル・ディレクトリー・サービスには、すべての従業員の詳細な個人情報が格納されています。ディレクトリーは、国別に分けられ、次に部門別に分けられます。従業員とその属性は、各部門の下に格納されています。



ディレクトリー階層例の画像に、サンプル・ディレクトリーからの抜粋を示します。この図は、ディレクトリー構造を示しています。最上位は root です。2 つのサブディレクトリーは国を表し、UK と US のラベルが付いています。UK サブディレクトリーは、組織単位を表す他の 3 つのサブディレクトリーにさらに分けられます。それらのサブディレクトリーには、Development、Accounting、および Help Desk のラベルが付いています。Development 組織単位内には、共通名が Shirley Clee と Hamish Wednesday である 2 つのサブディレクトリーがあります。

エンティティーは、その識別名によって参照されます。識別名は、エンティティーへの 経路です。例えば、経理部と Hamish Wednesday の識別名は次のようになります。

```
dn="ou=accounting, c=UK"
dn="cn=Hamish Wednesday, ou=Development, c=UK"
```

各従業員のエントリーには、複数の属性があります。例えば、Hamish Wednesday のエントリーには、次の詳細が含まれます。

```
cn: Hamish Wednesday
uid: ham
mail: HWednesday@development.mycompany.com
telephoneNumber: 88 88 55 44
```

ディレクトリー階層の各エンティティーは、ユーザー名 (LDAP ではこれが識別名) およびパスワードで保護することができます。モニターは、このユーザー名とパスワードを使用して LDAP サーバーにアクセスします。

モニターは、サーバーにアクセスするときに、ディレクトリー階層内のどの場所からターゲット・エンティティーの 検索を始めるかを指示します。これは、searchBase フィールドで識別名として指定されます。例えば、以下のように検索を部門レベルで開始できます。

```
ou=Accounting, c=UK
```

注 : 識別名を構成するエンティティーは、逆順に並んでいます。すなわち、階層内の最も低い点から始まり、その上の各エンティティーをリストします。

ターゲット・エンティティーは、「filter」フィールドに入れられてサーバーに渡されます。このフィールドには、ターゲット・エンティティーの 属性が含まれます。例えば、Hamish Wednesday のエンティティーを検索する場合、「filter」フィールドの内容を次のように設定できます。

```
(uid=ham)
```

LDAP サーバーは、モニターが提供するフィールドを 使用して、ターゲット・エンティティーを 検索します。検索の結果がモニターに返されます。

検索が成功した場合、サーバーは、ターゲット・エンティティーの 属性も返します。モニターは、その属性を、動的に作成された名前を持つエレメントに変換します。例えば、モニターは Hamish Wednesday のエントリーを以下のように変換します。

```
$dnMatched = "cn=Hamish Wednesday, ou=Development, c=UK"
$cn = "Hamish Wednesday"
$uid = "ham"
$mail = "HWednesday@development.mycompany.com"
$telephoneNumber = "88 88 55 44"
```

LDAP 認証

SSL LDAP サーバー認証では、Verisign や Thawte などの認証局によって署名された公開鍵証明書と秘密鍵証明書を利用します。SSL 認証の場合、LDAP モニターは公開証明書の Netscape cert7db データベースを使用して、認証局が発行した LDAP サーバー証明書の署名を検証します。

Verisign または Thawte などの Netscape が認識する認証局によって署名された証明書を使用している場合、LDAP モニターはそれらを自動的に認識します。自分が所属する組織または Netscape データベース内に含まれない組織が署名した証明書を使用する場合は、それらを cert7db データベースに追加する必要があります。

Netscape から提供される certutil ユーティリティーを 使用して、データベースに証明書を追加します。LDAP モニターの cert7db データベースは、`$ISHOME/certificates/cert7.db` ファイル内にあります。

SSL 暗号化または TLS 暗号化で保護されている LDAP サーバーをモニターするには、次の表に示されている環境変数を設定します。

表 *70.* セキュアな *LDAP* サーバーをモニターするために必要な環境変数

変数	説明	設定
LDAPTLS_CACERT	CA 証明書が入っているファイルを指定します。	サーバー証明書が入っているファイル。 例えば、`cacert.pem` のようになります。
LDAPTLS_REQCERT	サーバー証明書に対して実行するチェックを指定します。	`never|allow|try|demand` から選択します。

詳しくは、http://www.openldap.org を参照してください。

プロパティー

次の表では、LDAP モニターに固有のプロパティーについて説明しています。

表 *71. LDAP* モニターのプロパティー・オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
NOLDAPV3	0|1	モニターが強制的に LDAP v3 ではなく LDAP v2 を使用するようにします。 0 - LDAP v3 を使用します 1 - LDAP v2 を使用します

暗号スイート

SSLCipherSuite プロパティーは、LDAP モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

LDAP モニター・サービス・テストの構成

LDAP モニター構成パラメーターを使用して、サービス・テストを定義します。

モニターの構成時に、タイムアウト・パラメーターに表示されるデフォルト値は 30 秒、ポーリング間隔パラメーターのデフォルト値は 300 秒です。表にリストされたその他のデフォルトは、 構成中には表示されませんが、値を指定しなかった場合に構成の詳細を保存した時点で適用されます。

次の表では、LDAP モニター構成について説明します。

表 *72. LDAP* モニター構成	
フィールド	**説明**
サーバー	モニター対象の LDAP サーバーの名前または IP アドレス。例: `ldap.mycompany.in`
searchbase	検索を開始するロケーションの識別名。 例: `ou=Accounting, c=UK`
filter	検索するターゲット・エンティティーの属性。例: `(uid=ham)`
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: `LDAP monitor`
アクティブ	プロファイル・エレメントを作成後にアクティブ化するかどうかを選択します。例: `Selected`
ポート	接続する LDAP サーバー上のポート。SSL 認証を使用している場合は、SSL ポートを指定する必要があります。 デフォルト: 389
ユーザー名	ディレクトリー・サービスにログインするために使用するユーザー名。ユーザー名の形式は認証タイプの設定によって異なります。 Windows ドメインを指定できます (すなわち `DOMAIN¥username`)。例: `jbloggs`
パスワード	ディレクトリー・サービスにログインするために使用されるパスワード (必要な場合)。例: `secret9`

表 72. *LDAP* モニター構成 *(続き)*	
フィールド	**説明**
認証タイプ	使用する LDAP 認証方式: • SIMPLE (匿名または非暗号化テキストのパスワード) • SSL-SIMPLE • SASL-DIGEST-MD5 **注 :** SASL-DIGEST-MD5 認証は、Linux オペレーティング・システムでは使用できません。 authenticationtype を SIMPLE または SSL-SIMPLE に設定した場合は、ユーザー名を識別名形式で入力してください。authenticationtype を SASL-DIGEST-MD5 に設定した場合は、ユーザー名を SASL バインド ID として入力してください。anonymous ユーザーとして LDAP サーバーにログインするには、authenticationtype を SIMPLE に設定し、ユーザー名フィールドとパスワード・フィールドをブランクのままにします。 デフォルト: SIMPLE
saslrealm	LDAP サーバーの認証レルム。通常はサーバーの完全修飾ドメイン名です。複数のシステムでパスワードを共有する場合、ドメイン名を使用することができます。例: `mycompany.com`
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

サービス・レベル分類

LDAP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
connectTime
searchTime
initTime
dnMatched
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、`message` は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、LDAP モニターは、LDAP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、LDAP モニターの追加エレメントをリストしています。

表 *73. LDAP* モニター・エレメント

エレメント	**説明**
$authentication	LDAP サーバーで必要とされるユーザー認証方式のタイプ (標準または CRAM-MD5)。
$connectTime* (ConnectTime)	LDAP サーバーに接続するために要した時間。
$distinguishedName* (UserName)	ディレクトリー・サービスにログインするために使用された識別名。
$dnMatched	検索で合致したエンティティー。
$filter* (SrchFilter)	ターゲット・エンティティーを見つけるために使用された属性。
$initTime* (InitTime)	LDAP クライアントを初期化するために要した時間。
$port* (Port)	モニターが接続されている LDAP サーバー上のポート。
$saslRealm	新規 LDAP エレメントの追加後に指定した SASL レルム。
$searchBase* (SearchBase)	検索が開始されたエンティティーの 識別名。
$searchTime* (SearchTime)	検索を完了するために要した時間。

状況メッセージ

LDAP モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、LDAP 状況メッセージについて説明します。

表 *74. LDAP* モニターの状況メッセージ

メッセージ	**説明**
`Search successful`	要求は正常に処理されました。
`Search failed`	要求は失敗しました。

表 74. LDAP モニターの状況メッセージ (続き)	
メッセージ	**説明**
No match	サーバーでは、検索基準で一致する項目が見つからなかった可能性があります。
Connection timed out	接続は正常に行われましたが、サーバーが応答を停止しました。
Initialisation failed - an unrecognised authentication type was specified	LDAP モニターがサポートしない認証タイプが使用された場合に発生します。
Client initialisation failed	メモリーが不十分であるため、LDAP 構造の初期化は失敗しました。
Bind (authentication) failed	バインドの完了を待機しているサーバーがタイムアウトになりました。
SASL bind is not possible because server does not support LDAPv3	SASL バインドを作成するには、サーバーが LDAPv3 をサポートしている必要があります。
SASL bind is not possible because one of 'bind_id' (username), password or sasl_realm is blank	バインドが行われるようにするには、すべての認証フィールドに値が指定されている必要があります。そのため、ユーザーが SIMPLE 認証タイプを使用して匿名 (非暗号化テキスト) でログインする場合は、SASL バインドは不可能になります。
SASL bind error	SASL バインドが失敗した理由を特定できません。
SASL bind authorisation error	権限証明書が誤っていたため、SASL バインドは失敗しました。

IMAP4 モニター

IMAP4 モニターは、SMTP モニターと連動して、IMAP4 電子メール・サービスの可用性と応答時間をテストします。

以下の表に、IMAP4 モニター・ファイルをリストしています。

表 75. IMAP4 モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_imap4
プロパティー・ファイル	$ISHOME/etc/props/imap4.props
ルール・ファイル	$ISHOME/etc/rules/imap4.rules
ログ・ファイル	$ISHOME/log/imap4.log

IMAP4 モニターを構成するためのガイドライン

IMAP4 モニターは、SMTP モニターがテスト・メッセージを送信する宛先のメールボックスをモニターして、それらのメッセージの配信に要した時間を測定することにより、SMTP モニターと連動します。

注 : 配信時間の計算を正しく機能させるには、モニター・ホスト・コンピューターとメール・サーバーのシステム・クロックが同期していることを確認してください。

IMAP4 モニターは、メールボックスの内容を読み取ると、2 つの異なるタイプのイベントを生成します。

- メッセージ固有のイベント

  IMAP4 モニターは、メールボックスからダウンロードした電子メール・メッセージごとに、メッセージ固有のイベントを作成します。このタイプのイベントでは、モニターは $message エレメントを Message Successfully Downloaded に設定します。$timeToDeliver エレメントは、メッセージを送出した SMTP モニターとそれを受信したメールボックスの間でメッセージの移動に要した時間として計算されます。$hopCount エレメントは、メールボックスに到達するためにメッセージがホストを通過したホップ数を示します。

- 要約イベント

  モニターは、メールボックス内のすべてのメッセージの処理を完了すると、要約イベントを作成します。このタイプのイベントでは、$message エレメントはメールボックスから正常にダウンロードされたメッセージの合計数を示し、$totaltime エレメントは要求の完了に要した時間を示します。$totaltime は、秒単位です。

セキュア・メール

IMAP4 モニターは、機密保護機能のあるメール・サービスへの接続をサポートします。 SSL/TLS か、または STARTTLS コマンドを使用して接続できます。プロファイル・エレメントを定義する際に、「securitytype」フィールドを使用して適切なセキュリティーを選択してください。メール・サーバーで、SSL 暗号化のクライアント・サイド証明書が必要な場合は、SSL プロパティーを使用して、証明書ファイル、鍵ファイル、鍵パスワード、および暗号スイートを指定してください。

クライアント・サイド証明書

IMAP4 モニターを使用すると、相互認証のためにクライアント ・サイド証明書を必要とするサーバーをモニターできます。

プロファイル・エレメントの作成時に、SSL 証明書ファイル、鍵格納ファイル、および鍵パスワードを指定します。

証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。証明書が別の形式である場合は、PEM 形式に変換する必要があります。証明書は openSSL などのソフトウェアを使用して変換できます。openSSL は、http://www.openssl.org から入手できます。

注 : すべてのプロファイル・エレメントで常に同じ証明書、鍵、およびパスワードを使用する場合は、作成するプロファイル・エレメントごとにそれらを定義するのではなく、モニター・プロパティーを使用して指定してください。

メールボックス

IMAP4 モニターは、SMTP モニターが送信した電子メール・メッセージに含まれる情報を処理した後で、メールボックスからそれを削除します。任意の既存のメールボックスを使用して、2 つのモニター間の電子メール・メッセージを保管できます。そのメールボックスが実際のユーザーに属するものであってもかまいません。ただし、サービス・テスト用の特別なメールボックス・アカウントを作成することをお勧めします。

IMAP4 モニター・サービス・テストの構成

IMAP4 モニター構成パラメーターを使用して、サービス・テストを定義します。

モニターの構成時に、タイムアウト・パラメーターとポーリング間隔パラメーターのデフォルト値が表示されます。タイムアウトのデフォルトは 30 秒、ポーリング間隔のデフォルトは 300 秒です。以下の表にリストされたその他のデフォルトは、構成時には表示されませんが、値を指定しなかった場合に構成の詳細を保存した時点で適用されます。

表 *76. IMAP4* モニター構成

フィールド	**説明**
サーバー	メール・サーバーの IP アドレス。例: test.mycompany.com

表 76. *IMAP4* モニター構成 (続き)	
フィールド	**説明**
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	IMAP4 サーバーの IP ポート。 デフォルト: 143
セキュリティー・タイプ	メール・サーバーとの間で開かれたセキュア接続のタイプ: • NONE - セキュリティーを使用せずに接続します。 • SSL - SSLv2 hello を送信してから、SSLv2、SSLv3、または TLSv1 をネゴシエーションします。 • STARTTLS - セキュリティーを使用せずに接続し、STARTTLS コマンドを実行してから、TLSv1 を介して接続を確立します。 デフォルト: NONE
ユーザー名	メールボックスの名前。
パスワード	メールボックスにログインするために使用されるパスワード (必要な場合)。
認証タイプ	使用する認証方式 (STANDARD または CRAM_MD5)。 デフォルト: STANDARD
共有秘密鍵	CRAM_MD5 認証の共有秘密鍵 (該当する場合)。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行できます。 モニターは、検索された内容を各正規表現に突き合わせようとします。

指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

正規表現について詳しくは、333 ページの表 50 を参照してください。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、IMAP4 モニターは、IMAP4 サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、IMAP4 モニターの追加エレメントについて説明します。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 77. *IMAP4* モニター・エレメント

エレメント	説明
$authentication	IMAP4 サーバーで必要とされるユーザー認証方式のタイプ (標準または CRAM-MD5)。
$bytesPerSec	1 秒当たりの平均転送バイト数。
$bytesTransferred	アップロードまたはダウンロードされたバイト数。
$connectTime	IMAP4 サーバーに接続するために要した時間。
$downloadTime* (DownloadTime)	ファイルをダウンロードするために要した時間。
$hopCount	メッセージがメールボックスに到達するまでにホストを通過したホップ数。
$inEvent	このイベントが、一連のイベントの一部であることを示します。 1 は、それが最終イベントではないことを示します。 0 は、最終イベントであることを示します。
$lookupTime*(LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$port*(Port)	サービスがモニターされているポート。
$responseTime* (ResponseTime)	接続が確立されてからデータの最初のバイトが受信されるまでの時間。
$security	IMAP エレメントを追加するときに指定した、 メール・サーバーとの間で開かれたセキュア接続のタイプ (NONE、STARTTLS、または SSL)。
$sentTo*(SentTo)	SMTP モニターが元のメッセージを送信するために使用する電子メールメール・アドレス。
$smtpServer	電子メールの送信元である SMTP サーバーの名前。
$SSLHandshakeTime* (SslHandshakeTime)	SSL 接続を確立するために要した時間。
$timeToDeliver	電子メール・メッセージが SMTP モニターとその宛先メールボックスの間で移動するために要した時間。
$user*(ImapUser)	モニターが IMAP4 サーバーにログインするために使用するユーザー名 (アカウント名)。

状況メッセージ

IMAP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、IMAP4 状況メッセージについて説明します。

表 *78. IMAP4* モニターの状況メッセージ

メッセージ	説明
Message successfully downloaded	メッセージは正常にダウンロードされました。
Downloaded x messages	メールボックスから正常にダウンロードされたメッセージの数を示します。
Server not IMAP4rev1 compliant	IMAP4 サーバーが IMAP4 仕様 (RFC2060) に準拠していません。
Server does not support STARTTLS capability	サーバーは正しく構成されていません。
Unable to log into server	モニターは IMAP サーバーにログインできません。
Unrecognised response to STATUS command	モニターが、サーバーによって返された値を認識しません。
Unrecognised response to FETCH INTERNALDATE command	
Failed to obtain Actual-Time-Sent header	モニターは、サーバーから予期する応答を取得しませんでした。
Failed to obtain Actually-To header	
Failed to obtain SMTP-Server header	

プロパティー

次の表では、IMAP4 モニターに固有のプロパティーおよびコマンド行オプションについて説明しています。

表 *79. IMAP4* モニター・プロパティー

プロパティー名	プロパティー・パラメーター	説明
Originator	ストリング	SMTP モニターによって送信されたテスト電子メール・メッセージを取得する場合に突き合わせる From フィールドを指定します。モニターは、From フィールドが発信元のストリングと合致するメッセージのみを取得します。IMAP4 の発信元は、SMTP モニターの発信元と合致する必要があります。 デフォルト: SMTP-Monitor
SSLCertificate File	ストリング	HTTPS エレメントの作成時に証明書が明示的に指定されなかった場合に使用されるデジタル証明書のパスおよびファイル名。 パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー ($ISHOME/platform/arch/bin) からの相対位置にあるものと解釈します。

表 79. *IMAP4* モニター・プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
SSLCipherSuite	ストリング	SSL 操作で使用する暗号スイート。指定可能な値の説明については、『暗号スイート』を参照してください。 デフォルト: RC4:3DES:DES:+EXP
SSLDisableTLS	整数	ヘリテージ・サポートで TLSv1 を使用不可にします。 デフォルト: 0 - TLSv1 を使用可能にします。 1 - TLSv1 を使用不可にします。
SSLKeyFile	ストリング	SSL 秘密鍵が入ったファイル。
SSLKeyPassword	ストリング	SSL 秘密鍵の暗号化に使用されるパスワード。

暗号スイート

SSLCipherSuite プロパティーは、IMAP4 モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

NTP モニター

Network Time Protocol (NTP) モニターは、UDP (ユーザー・データグラム・プロトコル) を使用して NTP サーバーを照会し、サーバーが正しい時刻を提示しているかどうかを判別します。

NTP では、協定世界時を使用して、コンピューターのクロックをミリ秒単位に同期化します。

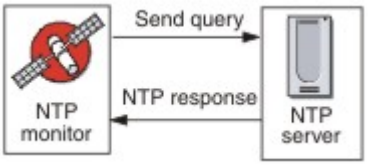
以下の表に、NTP モニター・ファイルをリストしています。

表 80. *NTP* モニター・ファイル	
モニター・ファイル	**名前およびロケーション**
モニター実行可能ファイル	nco_m_ntp
プロパティー・ファイル	$ISHOME/etc/props/ntp.props
ルール・ファイル	$ISHOME/etc/ntp.rules
ログ・ファイル	$ISHOME/log/ntp.log

NTP モニターを構成するためのガイドライン

NTP モニターは、NTP サーバーに照会を送信することによって、データを獲得します。NTP サーバーは、現在時刻 (NTP サーバーから見て) の UDP 応答パケットを返します。

次の図は、モニターと NTP サーバーとの間で交換されるメッセージの例を示します。



NTP モニター・サービス・テストの構成

NTP モニター構成パラメーターを使用して、サービス・テストを定義します。

次の表では、NTP 構成について説明します。

表 *81. NTP* 構成	
フィールド	**説明**
サーバー	NTP サーバーのホスト名。例: `ntp.mycompany.com`
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: `NTP`
ポート	使用する NTP サーバー上のポート。 デフォルト: `123`
ローカル IP	モニターがテストの実行時にバインドする、ホスト・システム上のネットワーク・インターフェースの IP アドレスを指定します。 モニターの `IpAddress` プロパティーを設定すると、このフィールドの値がオーバーライドされます。例: `102.168.n.n`
バージョン	使用する NTP サーバーのバージョン (1、2、3、または 4)。 デフォルト: `1`
timeout	サーバーの応答を待機する時間 (秒単位)。 デフォルト: `10`
再試行	モニターが NTP サーバーへの接続を再試行する回数。 デフォルト: `0`
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: `300`
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: `0`
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: `10`

サービス・レベル分類

サービス・レベル分類は、NTP を介して提供されるサービス・レベルを決定するためのルールを定義します。

NTP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
responseTime
lookupTime
offset
adjustedOffset
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。

- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して $message エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、NTP モニターは、NTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、NTP モニターの追加エレメントについて説明します。

表 *82. NTP* モニター・エレメント

エレメント	説明
$adjustedOffset	サーバーからの時間オフセット (秒単位)。
$localIP	モニターが使用するように構成されているローカル IP アドレス。 インターフェースが 1 つのみのシステムではブランクにすることができます。
$lookupTime* (LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$ntpVersionIn	サーバーからの応答に使用されるプロトコル・バージョン。
$ntpVersionOut	送信に使用されるプロトコル・バージョン。
$offset	NTP サーバーと、モニターを実行中のシステムとの時間差 (秒単位)。
$port* (Port)	使用する NTP サーバー上のポート。
$responseTime* (ResponseTime)	モニターが NTP サーバーに接続してから応答を受信するまでの時間。
$retries	応答 ID を受信しなかった場合に要求を再送信する回数。

状況メッセージ

NTP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、NTP 状況メッセージについて説明します。

表 *83. NTP* モニターの状況メッセージ

メッセージ	説明
Successful query	NTP サーバーは予期される応答を返しました。
Connection failed	UDP ソケットを初期化できません。
Failed to send request to NTP server	UDP ソケットに書き込むことができません。
No response from server	NTP サーバーは応答しませんでした。

NNTP モニター

NNTP モニターは、ニュースグループとの間で読み取りおよび投稿を行って、NNTP サービスの可用性をテストします。

以下の表に、NNTP モニター・ファイルをリストしています。

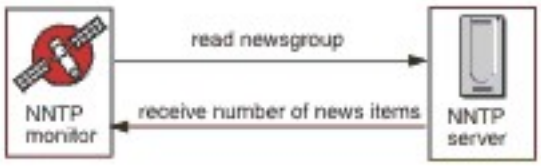
表 *84. NNTP* モニター・ファイル

モニター・ファイル	名前またはロケーション
モニター実行可能ファイル	nco_m_nntp
プロパティー・ファイル	$ISHOME/etc/props/nntp.props
ルール・ファイル	$ISHOME/etc/rules/nntp.rules
ログ・ファイル	$ISHOME/log/nntp.log

NNTP モニターを構成するためのガイドライン

NNTP モニターは、NNTP サーバーに投稿したり、NNTP サーバーからの読み取りを行って、NNTP サービスをテストします。モニター用に作成した各プロファイル・エレメントは、読み取り操作または投稿操作を実行します。

読み取り操作では、モニターは、NNTP サービスに接続して、特定のインターネット・ニュースグループが存在するかどうかをチェックします。ニュースグループが存在する場合、モニターはそのニュースグループのニュース項目の数を記録します。また、そのニュースグループに最後に追加されたニュース項目も記録しようとします。次の図は、読み取り操作を示しています。



投稿操作では、モニターはニュースグループが存在することをチェックしてから、そのニュースグループへのテスト・メッセージの書き込みを試みます。テスト・メッセージの件名は、NNTP Monitor Test Message です。次の図は、投稿操作を示しています。

各プロファイル・エレメントは、NNTP サーバーへのアクセス時にモニターが提供する username および password を指定します。モニターは、非暗号化テキスト認証システムを使用します。

```
AUTHINFO USER username
AUTHINFO PASS password
```

ここで username および password は、モニター・プロファイル・エレメントで指定されたものです。

プロパティー

次の表では、NNTP モニターに固有のプロパティー・オプションについて説明しています。

表 *85. NNTP* モニターのプロパティー・オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
OutputDirectory	ストリング	**OutputResult** が真である (1 に設定される) 場合に使用する出力ディレクトリーを指定します。 デフォルト: `$ISHOME/var`
OutputResult	0|1	モニターがサービスから受信するデータを保管できることを指定します。 0 - 無効 1 - 有効

NNTP モニター・サービス・テストの構成

NNTP モニター構成パラメーターを使用して、サービス・テストを定義します。

次の表では、NNTP モニター構成をリストしています。

表 *86. NNTP* モニター構成	
フィールド	**説明**
サーバー	NNTP サーバーの IP アドレス。例えば、`news.mycompany.com` です。
newsgroup	モニターがテスト・メッセージの投稿および読み取りに使用するニュースグループの名前。例: `mycompany.test`
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: `READ`
ポート	NNTP サーバーのポート番号。 デフォルト: `119`
ユーザー名	NNTP サーバーでの認証に使用されたユーザー名。
パスワード	NNTP サーバーでの認証に使用されたユーザー名のパスワード。
action	記事を投稿するか取得するかを示します。これは `READ` または `POST` になります。 デフォルト: `POST`
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: `30`
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: `300`
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: `0`
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: `10`

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされている情報で正規表現検索を実行できます。 NNTP モニターは、検索された内容を各正規表現に突き合わせようとします。

指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

詳しくは、333 ページの表 50 を参照してください。

サービス・レベル分類

サービス・レベル分類は、NNTP を介して提供されるサービス・レベルを決定するためのルールを定義します。

NNTP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
lookupTime
connectTime
transferTime
responseTime
status
bytesTransferred
bytesPerSec
newsItems
expected
lastLineReceived
checksum
previousChecksum
regexpMatch1 to 3
regexpStatus1 to 3
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して $message エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。
- status コード 220 および 240 は正常終了を示します。操作によって返されるその他の状況コードについては、NNTP プロトコルを参照してください。
- egexpStatus*n* が取り得る値を以下に示します。
  - NONE: 正規表現の検査が構成されていません
  - MATCHED: 正規表現に合致するものが見つかりました
  - FAILED: 正規表現に合致するものが見つかりませんでした
- 次の形式のテスト式を使用して正規表現の合致を評価します。

```
regexpMatchn [contains|!contains] expression
```

contains および !contains 演算子を = および != の代わりに使用します。regexpMatch*n* には、通常、合致部分だけでなく、正規表現と合致する行全体が含まれ、= および != が正規表現と合致しないことが多いためです。

- Checksum および PreviousChecksum エレメントは、通常、サービス・レベル分類の有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$checksum および

$previousChecksum モニター・エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、NNTP モニターは、NNTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、NNTP モニターの追加エレメントについて説明します。

表 *87. NNTP* モニター・エレメント

エレメント	**説明**
$action* (NntpAction)	モニターが実行したアクション。 これは READ または POST になります。
$bytesPerSec	1 秒当たりの平均転送バイト数。
$bytesTransferred	アップロードまたはダウンロードされたバイト数。
$checksum	Checksum エレメントは、通常、サービス・レベル分類の有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$checksum および $previousChecksum モニター・エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$connectTime* (ConnectTime)	NNTP サーバーへの接続を確立するために要した時間。
$downloadTime	ファイルをダウンロードするために要した時間。
$group* (NntpGroup)	モニター対象ニュースグループの 名前。
$lastLineReceived	このエレメントは、$message エレメントにメッセージ「Expect Failed」が含まれる場合にのみ設定されます。設定される場合は、NNTP サーバーの応答が含まれます。
$lookupTime* (LookupTime)	サーバーの IP アドレスをルックアップするために要した時間。
$newsItems	新規グループの新規項目の数。
$password	モニターの認証に使用されたパスワード。

表 87. NNTP モニター・エレメント (続き)	
エレメント	**説明**
$previousChecksum	PreviousChecksum エレメントは、通常、サービス・レベル分類の有意義な値を提供しません。これは、プロファイル・エレメントを作成するときに、checksum 値が不明であるためです (モニターは、テストの進行中に checksum 値を計算します)。$previousChecksum および $checksum モニター・エレメントは、モニターのルール・ファイルを使用するアラート機能強化を目的としています。
$responseTime* (ResponseTime)	接続が作成されてからターゲット記事の最初のバイトが受信されるまでに要した時間。
$status	NNTP サーバーによって返された状況コード。
$transferTime* (TransferTime)	値を $uploadTime または $downloadTime に設定します。
$uploadTime	ファイルをアップロードするために要した時間。
$username	モニターの認証に使用されたユーザー名。
$message に $ExpectFailed が含まれる場合	
$expected	接続失敗時にモニターが待機していた接続のテキスト。
$lastLineReceived	接続でモニターが NNTP サーバーから受信したテキストの最終行。

状況メッセージ

NNTP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、NNTP 状況メッセージについて説明します。

表 88. NNTP モニターの状況メッセージ	
メッセージ	**説明**
Article Posted	NNTP POST アクションは正常終了しました。
Article Retrieved	NNTP READ アクションは正常終了しました。
Not Found	記事を見つけることができませんでした。
Expect failed	NNTP 要求は失敗しました。
Timed out waiting to read	サーバーへのデータ接続は確立されましたが、応答が停止しました。

表 88. NNTP モニターの状況メッセージ (続き)	
メッセージ	**説明**
Connection failed	モニターはサーバーへの接続に失敗しました。詳しくは、ログ・ファイルを参照してください。
Connection closed by foreign host	リモート・ホストは、モニターが予期するより早く接続を閉じました。

POP3 モニター

POP3 モニターは、SMTP モニターと連動して、POP3 電子メール・サービスの可用性と応答時間をテストします。

以下の表に、POP3 モニター・ファイルをリストしています。

表 89. POP3 モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_pop3
プロパティー・ファイル	$ISHOME/etc/props/pop3.props
ルール・ファイル	$ISHOME/etc/rules/pop3.rules
ログ・ファイル	$ISHOME/log/pop3.log

POP3 モニターを構成するためのガイドライン

POP3 モニターは、SMTP モニターがテスト・メッセージを送信する宛先のメールボックスをモニターし、それらのメッセージの配信に要した時間を測定することにより、SMTP モニターと連動します。

注 : 配信時間の計算を正しく機能させるには、モニター・ホスト・コンピューターとメール・サーバーのシステム・クロックが同期していることを確認してください。

POP3 モニターは、メールボックスの内容を読み取ると、2 つの異なるタイプのイベントを生成します。

- メッセージ固有のイベント

  POP3 モニターは、メールボックスからダウンロードした電子メール・メッセージごとに、メッセージ固有のイベントを作成します。このタイプのイベントでは、モニターは $message エレメントを Message Successfully Downloaded に設定します。$timeToDeliver エレメントは、メッセージを送出した SMTP モニターとそれを受信したメールボックスの間でメッセージの移動に要した時間として計算されます。$hopCount エレメントは、メールボックスに到達するためにメッセージがホストを通過したホップ数を示します。

- 要約イベント

  モニターは、メールボックス内のすべてのメッセージの処理を完了すると、要約イベントを作成します。このタイプのイベントでは、$message エレメントはメールボックスから正常にダウンロードされたメッセージの合計数を示し、$totaltime エレメントは要求の完了に要した時間を示します。$totaltime は、秒単位です。

セキュア・メール

POP3 モニターは、機密保護機能のあるメール・サービスへの接続をサポートします。SSL/TLS か、または STARTTLS コマンドを使用して接続できます。POP3 モニター・エレメントを定義する際に、「セキュリティー・タイプ (Security Type)」フィールドを使用して適切なセキュリティーを選択してください。メール・サーバーで、SSL 暗号化のクライアント・サイド証明書が必要な場合は、SSLname プロパティーまたはコマンド行オプションを使用して、証明書ファイル、鍵格納ファイル、鍵パスワード、および暗号スイートを指定してください。

クライアント・サイド証明書

POP3 モニターを使用すると、相互認証のためにクライアント・サイド証明書を必要とするサーバーをモニターできます。プロファイル・エレメントの作成時に、SSL 証明書ファイル、鍵格納ファイル、および鍵パスワードを指定します。証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。証明書が別の形式である場合は、PEM 形式に変換する必要があります。証明書は openSSL などのソフトウェアを使用して変換できます。openSSL は、http://www.openssl.org から入手できます。

注 : すべてのプロファイル・エレメントで常に同じ証明書、鍵、およびパスワードを使用する場合は、作成するプロファイル・エレメントごとにそれらを定義するのではなく、モニター・プロパティーを使用して指定してください。

POP3 モニター・テストの構成

注 : テスト・メールボックスにメッセージを送信するように SMTP モニターを構成し、メッセージを取り出すように POP3 モニターを構成することによって、メール・サーバー `mail.mycompany.com` の操作をモニターします。テスト・メールボックスのアドレスは `ismtest@mycompany.com`、資格情報は `ismtest/secret1` です。接続タイムアウトは 20 秒、障害発生時の再テストは 2 回、それぞれのサイドでの再テスト間隔は 5 秒とし、10 分ごとにサービスをテストします。プロファイル・エレメントが提供するデフォルトのサービス・レベル分類を使用します。

表 *90. POP3* モニター構成

フィールド	説明
サーバー	メール・サーバーの IP アドレス。例: `mail.mycompany.com`
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	メール・サーバーのポート番号。 デフォルト: `110`
セキュリティー・タイプ	メール・サーバーとの間で開かれたセキュア接続のタイプ: • NONE - セキュリティーを使用せずに接続します。 • SSL - SSLv2 `hello` を送信してから、SSLv2、SSLv3、または TLSv1 をネゴシエーションします。 • STARTTLS - セキュリティーを使用せずに接続し、STLS コマンドを実行してから、TLSv1 を介して接続を確立します。これが最も安全なセキュリティー・タイプです。 • NONE - セキュリティーを使用せずに接続します。 デフォルト: NONE
ユーザー名	メールボックスの名前。
パスワード	メールボックスにログインするために使用されるパスワード (必要な場合)。
認証タイプ	使用する認証方式。ラベルは「認証タイプ」です。 • STANDARD - ユーザー名/パスワード交換を使用します。パスワードは暗号化されません。これは、POP3 を断続的に使用する場合に適しています。 • APOP - POP3 クライアントが定期的にサーバーに接続する場合に使用します。これで、標準よりも高いレベルのセキュリティーが提供されます。APOP を選択した場合、必ず APOP 共有秘密鍵を指定してください。 すべてのサーバーが APOP をサポートするわけではないことに注意してください。 デフォルト: STANDARD。

表 *90. POP3* モニター構成 *(続き)*	
フィールド	**説明**
共有秘密鍵	APOP 認証の共有秘密鍵。APOP 認証タイプを使用している場合のみ適用されます。ストリングは 8 文字以上とする必要があり、ユーザー・インターフェースでは隠されています。
timeout	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
証明書の検証	サーバーの検証証明書。 デフォルト: Disabled

POP3 モニター構成パラメーターを使用して、サービス・テストを定義します。

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行できます。 モニターは、検索された内容を各正規表現に突き合わせようとします。

指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

正規表現について詳しくは、333 ページの表 50 を参照してください。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、POP3 モニターは、POP3 サービス・テストに固有のデータを含む一連のテスト結果を生成します。

表 1 では、POP3 モニターの追加エレメントについて説明しています。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 *91. IMAP4* モニター・エレメント	
エレメント	**説明**
$authentication	IMAP4 サーバーで必要とされるユーザー認証方式のタイプ (標準または CRAM-MD5)。
$bytesPerSec	1 秒当たりの平均転送バイト数。

表 *91. IMAP4* モニター・エレメント *(*続き*)*	
エレメント	**説明**
$bytesTransferred	アップロードまたはダウンロードされたバイト数。
$connectTime	IMAP4 サーバーに接続するために要した時間。
$downloadTime* (DownloadTime)	ファイルをダウンロードするために要した時間。
$hopCount	メッセージがメールボックスに到達するまでにホストを通過したホップ数。
$inEvent	このイベントが、一連のイベントの一部であることを示します。1 は、それが最終イベントではないことを示し、 0 は、最終イベントであることを示します。
$lookupTime*(LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$port*(Port)	サービスがモニターされているポート。
$responseTime* (ResponseTime)	接続が確立されてからデータの最初のバイトが受信されるまでの時間。
$security	IMAP エレメントを追加するときに指定した、 メール・サーバーとの間で開かれたセキュア接続のタイプ (NONE、 STARTTLS、または SSL)。
$sentTo*(SentTo)	SMTP モニターが元のメッセージを送信するために使用する電子メールメール・アドレス。
$smtpServer	電子メールの送信元である SMTP サーバーの名前。
$SSLHandshakeTime* (SslHandshakeTime)	SSL 接続を確立するために要した時間。
$timeToDeliver	電子メール・メッセージが SMTP モニターとその宛先メールボックスの間で移動するために要した時間。
$user*(ImapUser)	モニターが IMAP4 サーバーにログインするために使用するユーザー名 (アカウント名)。

状況メッセージ

POP3 モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、POP3 状況メッセージについて説明します。

表 *92. POP3* モニターの状況メッセージ	
メッセージ	**説明**
Message successfully downloaded	POP3 要求は正常に処理されました。
Downloaded *x* messages	メールボックスからダウンロードされたメッセージの数を示します。

表 92. POP3 モニターの状況メッセージ (続き)	
メッセージ	**説明**
Timed out waiting to read/write	サーバーへのデータ接続は確立されましたが、応答が停止しました。
Connection closed by foreign host	リモート・ホストは、モニターが予期するより早く接続を閉じました。
Connection failed	モニターはサーバーへの接続に失敗しました。詳しくは、ログ・ファイルを参照してください。
APOP not supported by the server	APOP 認証方式はサーバーによってサポートされていません。代わりに標準認証タイプを使用してください。
APOP service not available	APOP サーバー・インプリメンテーションはモニターによってサポートされていません。代わりに標準認証タイプを使用してください。
Server does not support STLS capability	サーバーは STARTTLS をサポートしていません。別のセキュリティー・タイプを使用してください。

プロパティー

以下の表では、POP3 モニターに固有のプロパティーについて説明します。

表 93. POP3 モニターのプロパティーおよびコマンド行オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
SSLCertificate File	ストリング	POP3 エレメントの作成時に証明書が明示的に指定されなかった場合に使用されるデジタル証明書のパスおよびファイル名。 パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー (`$ISHOME/platform/arch/bin`) からの相対位置にあるものと解釈します。
SSLCipherSuite	ストリング	SSL 操作で使用する暗号スイート。デフォルト: `RC4:3DES:DES:+EXP` 指定可能な値の説明については、『暗号スイート』を参照してください。
SSLDisableTLS	整数	レガシー・サポートで TLSv1 を使用不可にします。 デフォルト: `0` - TLSv1 を使用可能にします。TLSv1 を使用不可にするには、`1` に設定します。
SSLKeyFile	ストリング	SSL 秘密鍵が入ったファイル。
SSLKeyPassword	ストリング	SSL 秘密鍵の暗号化に使用されるパスワード。

暗号スイート

SSLCipherSuite プロパティーは、POP3 モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

RADIUS モニター

Remote Authentication Dial In User Service (RADIUS) は、サービスへのリモート・アクセスの認証を行います。RADIUS モニターは、RADIUS サービスにアクセスするクライアント・システムをシミュレートして、サービスのパフォーマンスに関するデータを返します。

以下の表に、RADIUS モニター・ファイルをリストしています。

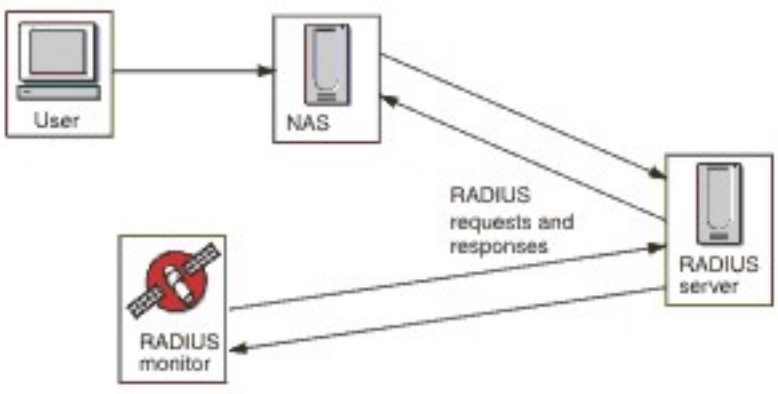
表 *94. RADIUS* モニター・ファイル

モニター・ファイル	**名前およびロケーション**
モニター実行可能ファイル	`nco_m_radius`
プロパティー・ファイル	`$ISHOME/etc/props/radius.props`
ルール・ファイル	`$ISHOME/etc/rules/radius.rules`
ログ・ファイル	`$ISHOME/log/http.log`

RADIUS モニターを構成するためのガイドライン

RADIUS モニターは、RADIUS サーバーへ要求を送信して、Network Access Server (NAS) の操作をシミュレートします。

RADIUS モニターは、UDP を使用して RADIUS サーバーに要求を送信してから、それらの要求の結果とサーバー・パフォーマンスに関するデータを含むイベントを生成します。次の図は、モニターの操作を示します。



モニターは、RADIUS サーバーの認証操作とアカウンティング操作の両方をテストできます。

- パスワード認証手順 (PAP) を使用した Access-Request
- チャレンジ・ハンドシェーク認証プロトコル (CHAP) を使用した Access-Request
- Accounting-Requests: Start、Stop、Accounting On、および Accounting Off

プロパティー

次の表では、RADIUS モニターに固有のプロパティー・オプションについて説明しています。

表 *95. RADIUS* モニターのプロパティー・オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
FramedServiceRequest	0|1	このプロパティーが 1 に設定される場合、モニターは、Access-Request で設定された Framed サービス・タイプを選択します。 0 - 無効 1 - 有効

RADIUS モニター・サービス・テストの構成

RADIUS モニター構成パラメーターを使用して、サービス・テストを定義します。

次の表では、Radius モニター構成について説明します。

表 *96. RADIUS* モニター構成	
フィールド	**説明**
サーバー	RADIUS サーバーの IP アドレス。
共有秘密鍵	モニターの認証に使用された共有秘密鍵。
ユーザー名	RADIUS サーバーを認証するためにモニターが提供したユーザー名。
パスワード	RADIUS サーバーを認証するためにモニターが提供したパスワード。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
requesttype	RADIUS サーバーに送信される要求のタイプを指定します。 • Authenticate (CHAP) • Authenticate (PAP) • Accounting デフォルト: Authenticate(CHAP)
ポート	RADIUS サーバーに接続するために使用するポート。 デフォルト: 1812
ローカル IP	モニターがテストの実行時にバインドする、ホスト・システム上のネットワーク・インターフェースの IP アドレスを指定します。 モニターの IpAddress プロパティーを設定すると、このフィールドの値がオーバーライドされます。
loginhost	Access-Request 内の Login-IP-Host 属性の値を設定します。
calledstation	Access-Request 内の Called-Station-Id 属性の値を設定します。
callingstation	Access-Request 内の Calling-Station-Id 属性の値を設定します。

表 96. *RADIUS* モニター構成 (続き)	
フィールド	**説明**
accountsessionid	アカウント・サーバーに送信される Accounting-Request パケット内の Acct-Session-Id 属性の値を設定します。 **注:** このフィールドは、Accounting 要求タイプのみに適用されます。
accountstatustype	アカウント・サーバーに送信される Accounting-Request パケット内の Acct-Status-Type 属性の値を設定します。 • Start • Stop • Accounting On • Accounting Off **注:** このフィールドは、Accounting 要求タイプのみに適用されます。 デフォルト: Start
accountsessiontime	アカウント・サーバーに送信される Accounting-Request パケット内の Acct-Session-Time 属性の値 (秒単位) を設定します。 **注:** このフィールドは、Accounting 要求タイプのみに適用されます。
nasip	RADIUS モニターによって Access-Request パケットの一部として送信された NAS-IP-Address 属性。
nasport	RADIUS モニターによって Access-Request パケットの一部として送信された NAS-Port 属性。
timeout	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 10
再試行	問題がある場合に RADIUS サーバーへの接続を再試行する回数。 デフォルト: 0
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

サービス・レベル分類

サービス・レベル分類は、RADIUS サービスによって提供されるサービス・レベルを決定するためのルールを定義します。

RADIUS モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
lookupTime
```

```
responseTime
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して $message エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、RADIUS モニターは、RADIUS サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、RADIUS モニターの追加エレメントについて説明します。

表 *97. RADIUS* モニター・エレメント

エレメント	説明
$accountSessionId	開始および停止レコードを突き合わせるために使用する固有 ID。
$accountSessionTime	accountStatusType が Stop に設定されている場合、このフィールドには、ユーザーがサービスを受信する時間が秒単位で表示されます。
$accountStatusType	ユーザーのサービスの開始 (start) であるか、終了 (stop) であるかを示します。
$calledStationId	RADIUS モニターは、calledStationId を Access-Request パケットの一部として送信します。これは、RADIUS サーバーによって要求される場合に使用され、callingStationId が使用される場合は使用されません。
$callingStationId	RADIUS モニターは、callingStationId を Access-Request パケットの一部として送信します。これは、RADIUS サーバーによって要求される場合に使用され、calledStationId が使用される場合は使用されません。
$localIP	モニターが使用するように構成されているローカル IP アドレス。 インターフェースが 1 つのみのシステムではブランクにすることができます。
$loginIPHost* (LoginIpHost)	RADIUS モニターは、loginIPHost を Access-Request パケットの一部として送信します。モニターされているサーバーで必要とされる可能性があります。
$lookupTime* (LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$nasPort* (NasPort)	RADIUS モニターによって Access-Request パケットの一部として送信された NAS Port パラメーター。デフォルト: 0
$password	モニターの認証に使用されたパスワード。

表 97. *RADIUS* モニター・エレメント *(続き)*	
エレメント	**説明**
$port* (Port)	サービスがモニターされているポート。
$requestType	エレメントのために選択された要求タイプ (PAP、CHAP、または Accounting)。
$responseTime	要求を RADIUS サーバーに送信してから応答を受け取るまでの時間。
$retries	最大試行回数。
$secret	構成ファイルから取られた共有秘密鍵パスワード。
$username* (RadiusUser)	モニターの認証に使用されたユーザー名。

状況メッセージ

RADIUS モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、RADIUS 状況メッセージについて説明します。

表 98. *RADIUS* モニターの状況メッセージ	
メッセージ	**説明**
CHAP authentication - Access granted	モニターは認証されました (CHAP を使用)。これは、CHAP 要求タイプが使用された場合にのみ返されます。
PAP authentication - Access granted	モニターは認証されました (PAP を使用)。これは、PAP 要求タイプが使用された場合にのみ返されます。
Accounting response received	サーバーからアカウンティング応答を受信しました。トランザクションは続行されます。
Connection failed	指定されたサーバー名は無効です。
Failed to send request to RADIUS server	UDP パケットをネットワークに書き込むことができませんでした。これ以上のエラー情報はありません。
No response from server	RADIUS サーバーが応答していません。
Incorrect identifier returned	モニターから送信されていない要求に対してサーバーが応答しました。
Invalid response authenticator	応答に、予期しない許可が含まれていました。共有秘密鍵またはパスワードの誤りが原因として考えられます。
Unrecognized response	サーバーは送信されたパケットを認識しませんでした。

表 98. *RADIUS* モニターの状況メッセージ (続き)	
メッセージ	**説明**
PAP authentication - Access denied	モニターは認証されませんでした (PAP を使用)。
CHAP authentication - Access denied	モニターは認証されませんでした (CHAP を使用)。

RPING モニター

RPING モニターは、ルーターからリモート側でネットワーク・デバイスを ping することによって、それらの可用性をテストします。最大、最小、および平均の往復時間のパフォーマンス・データを提供します。

モニターは、Cisco、Juniper ルーター、および RFC2925 準拠のルーターをサポートします。

次の表に、RPING モニター・ファイルをリストしています。

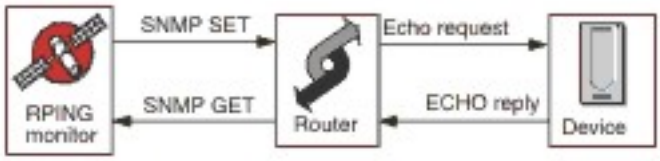
表 99. *RPING* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_rping
プロパティー・ファイル	$ISHOME/etc/ims/props/rping.props
ルール・ファイル	$ISHOME/etc/ims/rules/rping.rules
ログ・ファイル	$ISHOME/log/rping.log
スクリプト・ファイル	$ISHOME/scripts/rping/cisco.s (Cisco ルーター用の SNMP スクリプト) $ISHOME/scripts/rping/juniper.s (Juniper ルーター用の SNMP スクリプト) $ISHOME/scripts/rping/rfc2925.s (RFC2925 準拠のルーター用の SNMP スクリプト)

RPING モニターを構成するためのガイドライン

RPING モニターは、ネットワーク・デバイスを ping するようにルーターを構成してから、定期的にルーターをポーリングして ping の結果を取得することによって、データを獲得します。

モニターは、SNMP SET コマンドを使用してルーターの ping MIB にコントロール行を作成し、ping テストを構成してから、SNMP GET コマンドを使用して MIB から ping データを取り出します。ルーターとのすべての通信は SNMP を介して行われます。

次の図は、モニターとネットワーク・デバイスとの間で交換されるメッセージの例を示します。



Cisco ルーターでのリモート ping 要求の使用可能化

デフォルトでは、Cisco ルーターでのリモート ping SNMP 要求は使用不可になっています。ただし、RPING モニターが SNMP SET 要求を実行し、ping を開始するには、この要求を使用可能にする必要があります。

要求を使用可能にするには、Cisco ルーターにログインし、以下のコマンドを入力します。

```
enable
config terminal
snmp-server community communitystring rw
write mem
logout
```

ルーターで構成された communitystring は、そのルーター用に作成された任意の RPING プロファイル・エレメントの「communitystring」フィールドで入力したストリングと一致していなければなりません。write mem 行によって、ルーターのリブート時に設定が保管されるようになります。

Juniper ルーターでのリモート ping 要求の使用可能化

デフォルトでは、Juniper ルーターでのリモート ping SNMP 要求は使用不可になっています。RPING モニターが Juniper ルーターを使用して作動するためには、SNMP 要求を使用可能にする必要があります。

ルーターで SNMP 要求を使用可能にするには、JUNOS 構成の SNMP セクションが以下の内容と一致するようにしてください。

```
[edit snmp]
view ping-mib-view {
  oid .1.3.6.1.2.1.80 include; # pingMIB
  oid jnxPingMIB include; # jnxPingMIB
}
community communitystring {
  authorization read-write;
  view ping-mib-view;
}
```

ルーターで構成された communitystring は、そのルーター用に構成された任意の RPING プロファイル・エレメントの「communitystring」フィールドで入力したストリングと一致していなければなりません。

プロパティー

次の表では、RPING モニターに固有のプロパティー・オプションについて説明しています。

表 *100. RPING* のプロパティー・オプション

プロパティー名	プロパティー・パラメーター	説明
MibDir	ストリング	モニターが使用する MIB ファイルが入っているディレクトリー。 デフォルト: `$ISHOME/mibs`。

RPING モニター・サービス・テストの構成

RPING モニター構成パラメーターを使用して、サービス・テストを定義します。

表 *101. RPING* モニター構成

フィールド	説明
サーバー	ルーターの名前または IP アドレス。 例えば、`rt1.mycompany.com` です。

表 *101. RPING* モニター構成 *(*続き*)*	
フィールド	**説明**
routertype	ルーターのタイプ: • CISCO • Juniper • RFC2925
host	ルーターの ping 先とするサーバーの名前または IP アドレス。
communitystring	ルーターとの通信に使用される SNMP コミュニティー・ストリングを指定します。例えば、`server1.mycompany.com` です。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例えば、`RPING monitor` のようになります。
vpn	ping の送信に使用する VPN の名前 (オプション)。 ルーターは、構成されたデフォルトの経路ではなく、指定された VPN を使用します。
バージョン	使用する SNMP バージョン: 1 - SNMPv1 2 - SNMPv2c 3 - SNMPv3 デフォルト: 2
numberofpings	送信する ping の数。 デフォルト: 5
packetsize	送信するパケットのサイズ (バイト単位)。 デフォルト: 64
packettimeout	ping から次の ping までの待ち時間 (秒単位)。 デフォルト: 500
securityname†	SNMP セッションのユーザー名。
authenticationphrase†	ユーザーの認証パスワード。
privacyphrase†	ユーザーのプライバシー・パスワード。
authenticationprotocol†	ユーザーの認証に使用するプロトコル。 • MD5 • SHA1 デフォルト: MD5
privacyprotocol†	セッションの暗号化に使用するプロトコル。 デフォルト: DES

表 *101. RPING* モニター構成 *(続き)*	
フィールド	**説明**
timeout	ポーリングの間隔 (秒単位)。 デフォルト: 10
再試行	モニターがサーバーへの接続を再試行する回数。 デフォルト: 3
ポーリング	ping から次の ping までの待ち時間 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
† SNMPv3 のみで適用されます。	

サービス・レベル分類

サービス・レベル分類は、RPING を介して提供されるサービス・レベルを決定するためのルールを定義します。

RPING モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
lookupTime
numPacketSent
numPacketsRecv
maxRTT
minRTT
averageRTT
respondPercent
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、RPING モニターは、RPING サービス・テストに固有のデータを含む一連のテスト結果を生成します。

次の表では、RPING モニターの追加エレメントをリストしています。

表 *102. RPING* モニター・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。

表 *102. RPING* モニター・エレメント *(続き)*	
エレメント	**説明**
(AverageRTT)	平均往復時間 (秒)。
$community	ルーターの SNMP コミュニティー・ストリング。
$communityString	ルーターとの通信に使用される SNMP コミュニティー・ストリング。
(MaxRTT)	最大往復時間 (秒)。
(MinRTT)	最小往復時間 (秒)。
$numPacketSent	モニターによって送信されたパケット数。
$numPings	RPING エレメントの追加時に指定された送信済み ping の数。
$packetSize	送信するパケットのサイズ。
$packetTimeout	パケットを送信してから次のパケットを送信するまでの待ち時間。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$remoteHost* (RemoteHost)	ルーターの ping 先とするサーバーの名前または IP アドレス。
(RespondPercent)	送信された ping で応答があったものの割合 (パーセント)。
$routerMan* (RouterName)	RPING エレメントの追加時に選択されたルーター・タイプ。 • CISCO • Juniper • RFC2925
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
(SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	ルーターの名前または IP アドレス。
$timeout	サーバーがそれまでに応答する必要がある秒数。 構成ファイルから取られます。
$vpn* (Vpn)	RPING プロファイル・エレメントの「**vpn**」フィールドで指定された VPN の名前。

状況メッセージ

RPING モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、RPING 状況メッセージについて説明します。

表 *103. RPING* モニターの状況メッセージ	
メッセージ	**説明**
Got Response	モニターが Cisco デバイスから応答を受信しました。
Error in packet - exiting thread	パケットの 1 つでエラーが発生しました。
Timed out while trying initial sets	rowEntry フィールドを作成しようとしたときにルーターから応答がありませんでした。
Internal Error	ルーターでエラーが発生しました。
Host poll did not finish	ネットワーク・デバイスが ping を終了しませんでした。
Response Failed Operation Failed	ルーターはネットワーク・デバイスを ping できませんでした。
Timed out on Get requests	ルーターから結果を取得しようとしたときにモニターはタイムアウトになりました。

RTSP モニター

Real Time Streaming Protocol (RTSP) モニターは、ストリーム・サーバー上のオーディオ・ストリームおよびビデオ・ストリームのプレイバックをテストします。メディア・ファイルに関する情報を収集して、ストリーミング・セッションのストリームのプレイバック、休止、および終了を開始します。

以下の表に、RTSP モニター・ファイルをリストしています。

表 *104. RTSP* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_rtsp
プロパティー・ファイル	$ISHOME/etc/props/rtsp.props
ルール・ファイル	$ISHOME/etc/rules/rtsp.rules
ログ・ファイル	$ISHOME/log/rtsp.log

RTSP モニターを構成するためのガイドライン

RTSP モニターは、DESCRIBE モードまたは PLAY モードのいずれかでストリーミング・サーバーに接続します。このモニターは、Darwin などの純粋な RTSP サーバーによって配信された情報または統計をダウンロードします。

DESCRIBE モード

DESCRIBE モードでは、RTSP モニターは、ストリーミング・サーバーに接続し、オーディオ・ファイル、ビデオ・ファイル、およびストリームに関する情報を要求します。

サーバーは状況コードを返します。状況コードの値が 200 の場合は、ダウンロード可能ファイルであることを示し、その他の値の場合は、要求されたファイルを再生できない理由を示します。

再生に関連する統計は、このモードでは報告されませんが、RTSP をサポートするサーバーの基本的な機能をテストすることはできます。

PLAY モード

PLAY モードでは、RTSP モニターは DESCRIBE モードと同じ方法でストリーミング・サーバーに接続し、ファイルのストリーミングを実行して、要求されたダウンロードの統計を提供します。

プロパティー

次の表では、RTSP モニターに固有のプロパティー・オプションについて説明しています。

表 *105. RTSP* モニターのプロパティー・オプション

プロパティー名	プロパティー・パラメーター	説明
StreamingSocket BufferSize	整数	ストリーミング・ソケット・バッファーのサイズ (8 から 64 KB の範囲)。 デフォルト: 8

RTSP モニター・サービス・テストの構成

RTSP モニター構成パラメーターを使用して、サービス・テストを定義します。

次の表では、RTSP モニター構成をリストしています。

表 *106. RTSP* モニター構成

フィールド	説明
サーバー	ストリーミング・サーバーを実行しているターゲット・システム。例えば、`rtsp.mymusic.com` です。
remotefile	ダウンロードされるファイル。例えば、`singalong.mp3` です。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例えば、`RTSP monitor` です。
ポート	ターゲット・システム上の、モニターが接続するポート。 デフォルト: 554
action	ストリームでサーバーが実行するアクション: • DESCRIBE • PLAY デフォルト: DESCRIBE
duration	ストリームのうちの、サーバーが再生する部分 (秒単位)。 デフォルト: 5

表 *106. RTSP* モニター構成 *(続き)*	
フィールド	**説明**
maxbandwidth	ストリーミングに使用される最大帯域幅 (ビット/秒)。 デフォルト: 1500000
タイムアウト	RTSP サーバーの応答を待機する時間 (秒単位)。 デフォルト: 10
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

サービス・レベル分類

サービス・レベル分類は、RTSP を介して提供されるサービス・レベルを決定するためのルールを定義します。

RTSP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
lookupTime
connectTime
responseTime
sdpDownloadTime
playbackTime
status
percentPacketsLost
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。
- 状況コード 200 は正常終了を示します。操作によって返されるその他の状況コードについては、RTSP プロトコルを参照してください。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、RTSP モニターは、RTSP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、RTSP モニターの追加エレメントについて説明します。

表 *107. RTSP* モニター・エレメント	
エレメント	**説明**
$action	モニターが実行したアクション。

表 *107. RTSP* モニター・エレメント *(*続き*)*	
エレメント	**説明**
$averageBandwidth	平均合計帯域幅 (ビット単位)。
$bytesReceived	受信したバイト数の合計。
$connectTime* (ConnectTime)	ターゲット・サーバーとの接続を確立するために要した時間。
$describeStageStatus	RTSP 会話のステージの状況コード。
$filename	メディア・ファイルの名前。
$lookupTime* (LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$maxBandwidth	構成インターフェースを 使用する最大帯域幅。
$mediaResponseTime	サーバーが要求ファイルのストリーミングを開始するのに要した時間。
$numberOfStreams	メディアに埋め込まれているストリームの数。
$percentPacketsLost	失われたパケットの割合。
$playbackTime* (PlaybackTime)	setupResponseTime と mediaResponseTime の合計を表す時間。
$playStageStatus	RTSP 会話のステージの状況コード。
$port	モニターのサーバーへのアクセスに使用されたポート。
$responseTime* (ResponseTime)	接続が確立されてからデータの最初のバイトが受信されるまでの時間。
$sdpDownloadTime* (SdpDownloadTimed)	メディア・ファイルに関するデータをダウンロードするために要した時間。
$setupResponseTime	playbackTime の一部を表す時間。 **注 :** このエレメントは、RTSP モニターが PLAY モードで作動している場合にのみ生成されます。
$setupStageStatus	RTSP 会話のステージの状況コード。
$status	RTSP サーバーによって返された状況コード。
$streamingTime	サーバーが要求ファイルのストリーミングを完了するのに要した時間。
$streamLength	メディア・ファイルの最長ストリームの長さ。

表 *107. RTSP* モニター・エレメント *(*続き*)*	
エレメント	**説明**
$teardownStageStatus	RTSP 会話のステージの状況コード。
$totalBandwidthRequired	合計帯域幅 (K ビット/秒)。
$totalPacketsLost	失われたパケットの合計数。
$totalPacketsReceived	受信したパケットの数。

状況メッセージ

RTSP モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 *108. RTSP* モニターの状況メッセージ	
メッセージ	**説明**
OK	要求は正常に処理されました。
Connection failed	モニターはサーバーへの接続に失敗しました。詳しくは、ログ・ファイルを参照してください。
Connection closed by foreign host	RTSP サーバーへの接続は失敗しました。
Timed out waiting to read/write	RTSP サーバーへのデータ接続は確立されましたが、問題が発生しました。
Play failed - no streams	モニターは応答を受信しましたが、再生に使用できるオーディオまたはビデオはありませんでした。
select() failed on RTSP socket (PLAY stage)	ソケットがリモート・サーバーから閉じられたか、応答を待っている間にタイムアウトになりました。
RTSP Server response not in expected format	サーバーからの応答は、モニターがサポートしない形式でした。
Redirection requested by server not supported by client	サーバーからの応答はクライアントによってサポートされていません。
Server cannot fulfill client request	要求は失敗しました。これ以上の情報はありません。
Server Error	サーバーで問題が発生し、要求は失敗しました。 サーバーによって 500 以上のコードが返されました。 詳しくは、RTSP プロトコル (RFC 2326) を参照してください。

表 *108. RTSP* モニターの状況メッセージ *(続き)*	
メッセージ	**説明**
RTSP response header CSeq doesn't match request CSeq	RTSP サーバーは、誤って構成されており、正しく機能していません。
Corrupted RTSP server response	
Corrupted session description	
RTSP SETUP response CSeq doesn't match request CSeq	
RTSP SETUP response, incomplete Session string	
RTSP SETUP response, Session ID has changed within the same session	
RTSP SETUP response does not contain server ports to connect to	
RTSP SETUP response does not contain server port pair to connect to	
RTSP PLAY response CSeq doesn't match request CSeq	
RTSP PLAY response, incomplete Session string	
RTSP PLAY response, Session ID has changed within the same session	
RTSP PLAY response, incomplete RTP-Info string	
RTSP PLAY response does not valid RTP seqnum in RTP-Info response	
RTSP PLAY response does not valid RTP time in RTP-Info response	

SAA モニター

Cisco サービス保障エージェント (SAA) は、IOS バージョン 12.2(2) 以上の Cisco 製品のパフォーマンス・モニター・エージェントです。

SAA モニターは、Cisco サービス保障エージェント 機能を使用して、Cisco ルーター間のさまざまなタイミングをテストします。

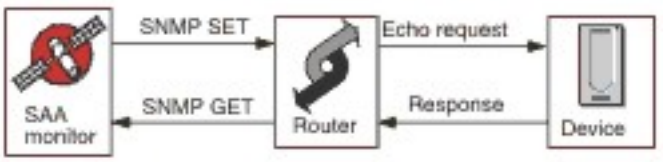
表 *109. SAA* モニターの概要	
モニター・ファイル	**名前またはロケーション**
実行可能ファイル名	nco_m_saa

表 109. SAA モニターの概要 (続き)	
モニター・ファイル	**名前またはロケーション**
プロパティー・ファイル	$ISHOME/etc/props/saa.props
ルール・ファイル	$ISHOME/etc/rules/saa.rules
ログ・ファイル	$ISHOME/log/saa.log
スクリプト・ディレクトリー	$ISHOME/scripts/saa/

SAA モニターを構成するためのガイドライン

SAA モニターは、Cisco Response Time Monitor の管理情報ベース (MIB) で定義されたエコー要求または応答 (指定時刻に作動する) を使用して別のネットワーク・デバイスまたはサービスの可用性をテストするように、ルーターの SAA を構成します。モニターは、Simple Network Management Protocol を使用して サービス保障エージェント と通信します。

次の図は、SAA モニターの操作を示します。



操作

SAA モニターは、他のネットワーク・デバイスでエコー・テスト (プローブと呼ばれます) を実行するように サービス保障エージェント を構成します。さまざまなプローブを構成することができ、それぞれが異なるプロトコルを使用します。

すべてのプローブは、IP 対応の任意のターゲット (Jitter を除く) に対して作動することができます。Jitter では、別の SAA 対応 Cisco 応答側ルーターが必要です。

各モニター・プロファイル・エレメントは、始動時にルーター上で サービス保障エージェント プローブを開始し、ポーリングが成功するごとに、結果情報を収集し、プローブのスケジュールを変更します。プローブが予期せぬタイミングで停止した場合、モニターは即時にプローブを再開します。プローブ・テストが完了したら、次のモニター・ポーリングまで非アクティブ状態になります。次のモニター・ポーリングで、結果データが収集され、別のテスト・サイクルが開始します。各回のポーリングで、モニターはプローブの状態をチェックします。プローブがまだ実行中の場合、モニターはプローブを停止してから、最後のサイクルの結果データとエラー情報を得るために管理情報ベース (MIB) をポーリングします。 次に、プローブのスケジュールを変更し、統計データをリセットして、プローブを再度活動化します。プローブは次のモニター・ポーリングまで無人で実行されます。

制御されないプロセスがルーターに残る可能性を排除するため、モニターは、各モニター・ポーリングでの存続期間が事前に定義されたプローブを開始します。モニターが強制終了された場合、モニターは存続期間が終わるまで実行を継続します。その後、エージアウト時刻に達するまで非アクティブ状態に移行し、ルーターはプロセスを終了させます。

IOS および サービス保障エージェント を事前に構成する必要はありません。モニターが実行時に、自動的に構成、制御、およびプローブ後のクリーンアップを行うためです。これには、一部のプローブ・タイプで必要な応答側ルーターの構成も含まれます。

プローブの永続性

ProbePersist モニター・プロパティーは、複数回のモニター・ポーリングにまたがるプローブの永続性を制御します。プローブの永続性が有効になっていない場合、プローブはポーリングのたびに開始され、テスト結果が生成されると直ちに終了します。

ルーターの負荷

時には、プローブ操作がルーターの負荷に影響を受ける可能性があります。StatusWait プロパティーは、プローブに対して、ある状態から別の状態に変わるためにどれだけの時間がかかると操作が失敗したと見なされるかを指定します。

プローブのタイプ

SAA モニターで使用可能なプローブのタイプを以下に示します。

- DHCP
- DLSW
- DNS
- FTP
- HTTP Get 要求
- ICMP エコー
- ICMP パス・エコー
- Jitter
- UDP エコー
- SNA エコー
- VOIP

エコー・プローブは、時間フレームに基づいてテストを実行しますが、Jitter プローブ、VOIP プローブ、および HTTP プローブは、単一の操作によってテストを実行します。

SAA プロパティー

SAA モニターのプロパティーを設定する必要があります。

次の表では、SAA モニターのプロパティーについて説明します。

表 *110. SAA* モニターのプロパティー

プロパティー名	**プロパティー・パラメーター**	**説明**
AgeOut	整数	プローブが停止するまで非アクティブのままである最大秒数。 デフォルトは 600 です。
MibDir	ストリング	MIB ファイルに使用されるディレクトリー。 デフォルトのパスは `$ISHOME/mibs` です。
ProbeLife	整数	プローブが無人の状態でアクティブのままである最大秒数。 デフォルトは 600 です。
ProbePersist	0|1	プローブは 2 つのモードで実行できます。モニター・ポーリングごとに単一テスト・サイクルで実行されるか、1 回開始されてからポーリングごとにスケジュールを変更されます。 • 0 は単一テスト・サイクルを示します • 1 はポーリングごとのスケジュール変更を示します
StatusWait	整数	プローブが失敗するまでアクションを実行するのをモニターが待機する秒数。

SAA モニター・サービス・テストの構成

SAA モニター・パラメーターを構成して、サービス・テストを定義する必要があります。

次の表では、SAA モニター構成のフィールドについて説明しています。

表 *111. SAA* モニター構成

フィールド	**説明**
サーバー	Cisco ルーターの名前または IP アドレス。

表 *111. SAA* モニター構成 *(*続き*)*	
フィールド	**説明**
communitystring	ルーターの SNMP コミュニティー・ストリング。
probetype	プロファイル・エレメントに該当する SAA プローブ・タイプ。
説明	エレメントの記述情報を提供するテキスト・フィールド。
アクティブ	プロファイル・エレメントがアクティブであるかどうかを示します。
ポート	ルーターへのアクセスに使用されたポート。 デフォルト・ポートは 161 です。
バージョン	使用する SNMP バージョン: • 1 は SNMPv1 に使用されます • 2 は SNMPv2c に使用されます • 3 は SNMPv3 に使用されます デフォルトは 1 です。
プローブ ID	プローブ制御の行のインデックスを生成するために使用される値を指定します。
securityname†	SNMP セッションのユーザー名。
authenticationphrase†	ユーザーの認証パスワード。
privacyphrase†	ユーザーのプライバシー・パスワード。
authenticationprotocol†	ユーザーの認証に使用されるプロトコルは次のとおりです。 • MD5 • SHA1 デフォルトは MD5 です。
privacyprotocol†	セッションの暗号化に使用するプロトコル。これは DES です。
timeout	ルーターの応答を待機する時間 (秒単位)。 デフォルトは 5 です。
再試行	モニターがルーターへの接続を再試行する回数。この回数だけ再試行すると、モニターは終了します。 デフォルトは 0 です。
ポーリング	ポーリングの間隔 (秒単位)。 デフォルトは 300 です。
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルトは 0 です。
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルトは 10 です。

注 : † SNMPv3 のみで適用されます。

プローブ・タイプの構成

プローブ構成は、プローブのタイプごとに異なり、Internet Service Monitoring エージェントでは、それぞれのタイプに固有の構成フィールドのセットを用意しています。プロファイル・エレメントを作成するには、プローブ・タイプを選択してから、そのタイプに適した構成を入力します。個々の構成項目について詳しくは、Cisco Response Time Monitor MIB の資料を参照してください。

サービス・レベル分類

サービス・レベル分類は、ネットワーク・デバイスによって提供されるサービス・レベルを決定するためのルールを定義します。

SAA モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
errTotal
numRTT
minRTT
maxRTT
avgRTT
minPosJitterSD
maxPosJitterSD
minNegJitterSD
maxNegJitterSD
minPosJitterDS
maxPosJitterDS
minNegJitterDS
maxNegJitterDS
packetLossSD
packetLossDS
packetOutOfSequence
packetMIA
packetLateArrival
minDelaySD
maxDelaySD
minDelayDS
maxDelayDS
avgPosJitterSD
avgPosJitterDS
avgNegjitterSD
avgNegJitterDS
avgDelaySD
avgDelayDS
devPosJitterSD
devPosJitterDS
devNegJitterSD
devNegJitterDS
devDelaySD
devDelayDS
MOS
ICPIF
mMinRTT
httpRTT
dnsRTT
tcpConnectRTT
transactionRTT
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 有効値のリストについては、409 ページの『状況メッセージ』を参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント
すべてのエレメントに共通するテスト結果に加えて、SAA モニターは、使用中のプローブのタイプに固有のデータを含む一連のテスト結果を生成します。

DHCP プローブ
DHCP プローブは複数のエレメントを生成します。

次の表では、DHCP プローブ・エレメントについて説明します。

表 *112. DHCP* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	dhcp
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	DHCP 要求の送信に使用されたルーターの名前。
$totalRTT † (TotalRTT)	DHCP サーバーから IP を取得するために要した合計往復時間 (秒単位)。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

DLSW プローブ
DLSW プローブは複数のエレメントを生成します。

次の表では、DLSW プローブ・エレメントについて説明します。

表 *113. DLSW* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT* † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errTotal* † (ErrorTotal)	エラーのあるパケットの合計数。
$maxRTT* † (MaximumRTT)	最大往復時間 (秒)。
$minRTT* † (MinimumRTT)	最小往復時間 (秒)。
$numRTT†	正常終了した往復の数。

表 *113. DLSW* プローブ・エレメント *(*続き*)*	
エレメント	**説明**
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType †	dlsw
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	SAA テストの実行に使用されたルーター。
$sumOfRTT (TotalRTT)	合計往復時間 (秒単位)。
(TargetHost)	ターゲット SAA が実行されているホストの名前または IP アドレス。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

DNS プローブ
DNS プローブは複数のエレメントを生成します。

次の表では、DNS プローブ・エレメントについて説明します。

表 *114. DNS* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$dnsHost (Host)	サーバーから解決されるホスト。
$dnsServer	DNS サーバーの IP。
(HostLookup)	IP アドレスのホスト。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	dns
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	DNS 要求の送信に使用されたルーターの名前。
$totalRTT † (TotalRTT)	DNS ルックアップの合計往復時間 (秒単位)。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

FTP プローブ
FTP プローブは複数のエレメントを生成します。

次の表では、FTP プローブ・エレメントについて説明します。

表 *115. FTP* プローブ・エレメント	
エレメント	**説明**
$activePassive	テストで使用される接続タイプ (`Active` または `Passive`)。 デフォルト: `Passive`
$authProto	エレメントの作成時に指定された認証プロトコル。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errorStatus	テストの状況を示す結果ストリング (`rttMonLatestRttOperSense` MIB オブジェクトから取得したもの)。
$ftpFile	テスト中に受信したテスト・ファイルの名前。
$ftpUrl (FtpUrl)	FTP テストで使用した URL。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	FTP 要求の送信に使用されたルーターの名前。
$totalRTT (TotalRTT)	テストの完了時刻 (`rttMonLatestRttOperCompletionTime` MIB オブジェクトから取得したもの) (秒単位)。

HTTP-Get プローブ
HTTP-Get プローブは複数のエレメントを生成します。

次の表では、HTTP-Get プローブ・エレメントについて説明します。

表 *116. HTTP-Get* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$dnsRTT † (DnsRTT)	DNS 照会を実行するための往復時間 (秒単位)。
$httpRTT † (HttpRTT)	HTTP 操作を実行するための往復時間 (秒単位)。
(HttpUrl)	モニターされている URL。
$messageBodyBytes	受信したメッセージ本文のサイズ。
$numRTT†	正常終了した往復の数。

表 116. *HTTP-Get* プローブ・エレメント *(続き)*	
エレメント	**説明**
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	http-get
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
(SourceRouter)	HTTP 要求の送信に使用されたルーターの名前。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
$targetHost	テストされているサービスのホストの名前。
$tcpConnectRTT † (TcpConnectRTT)	HTTP サーバーに接続するための往復時間 (秒単位)。
$transactionRTT † (TransactionRTT)	URL によって指定されたオブジェクトをダウンロードするための往復時間 (秒単位)。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

ICMP-Echo プローブ
ICMP-Echo プローブは複数のエレメントを生成します。

次の表では、ICMP-Echo プローブ・エレメントについて説明します。

表 117. *ICMP-Echo* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errBusies	前回の ping が完了しなかったために失敗した ping の数。
$errDisconnects	切断によって失敗した ping の数。
$ErrDrops	内部リソースが使用不可であったために失敗した ping の数。
$errNoConnects	ターゲットへの接続を確立できなかったために失敗した ping の数。
$errSequences	予期しないシーケンス ID を受け取ったために失敗した ping の数。
$errTimeouts	タイムアウトによって失敗した ping の数。
$errTotal † (ErrorTotal)	エラーのあるパケットの合計数。
$errVerifies	受信したデータが予期されるデータと異なっていたために失敗した ping の数。

表 *117. ICMP-Echo* プローブ・エレメント (続き)	
エレメント	**説明**
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$numRTT †	正常終了した往復の数。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	プローブ・タイプは以下のとおりでなければなりません。 • icmp-echo • icmp-echo-path • udp-echo
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	ICMP 要求の送信に使用されたルーターの名前。
$sumOfRTT	合計往復時間 (秒単位)。
$targetHost (Host)	モニターされているサービスのホスト名。
$tos (Tos)	Type of Service 値。
$vpn (Vpn)	VPN の名前。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

ICMP-Path-Echo プローブ
ICMP-Patch-Echo プローブは複数のエレメントを生成します。

次の表では、ICMP-Patch-Echo プローブ・エレメントについて説明します。

表 *118. ICMP-Path-Echo* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
(HopHost*One to Eight*)	ICMP エコー・パスを使用してアクセスする 1 番目から 8 番目のホスト。

表 *118. ICMP-Path-Echo* プローブ・エレメント *(続き)*	
エレメント	**説明**
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$numRTT †	正常終了した往復の数。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	プローブ・タイプは以下のとおりです。 • icmp-echo • icmp-echo-path
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	ICMP 要求の送信に使用されたルーターの名前。
$sumOfRTT	合計往復時間 (秒単位)。
$targetHost	テストされているサービスのホストの名前。
$tos (Tos)	Type of Service 値。
$vpn (Vpn)	VPN の名前。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

Jitter プローブ
Jitter プローブは複数のエレメントを生成します。

次の表では、Jitter プローブ・エレメントについて説明します。

表 *119. Jitter* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgDelayDS†	宛先からソースへの平均遅延 (秒単位)。
$avgDelaySD†	ソースから宛先への平均遅延 (秒単位)。
$avgNegJitterDS†	宛先からソースへの負の平均ジッター (秒単位)。
$avgNegJitterSD†	ソースから宛先への負の平均ジッター (秒単位)。
$avgPosJitterDS†	宛先からソースへの正の平均ジッター (秒単位)。
$avgPosJitterSD†	ソースから宛先への正の平均ジッター (秒単位)。

表 *119. Jitter* プローブ・エレメント *(*続き*)*	
エレメント	**説明**
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$devDelayDS†	宛先からソースへの遅延の標準偏差。
$devDelaySD†	ソースから宛先への遅延の標準偏差。
$devNegJitterDS†	宛先からソースへの負のジッターの標準偏差。
$devNegJitterSD†	ソースから宛先への負のジッターの標準偏差。
$devPosJitterDS†	宛先からソースへの正のジッターの標準偏差。
$devPosJitterSD†	ソースから宛先への正のジッターの標準偏差。
$errDescription	エラーの説明。
$errTotal (ErrorTotal)	エラーのあるパケットの合計数。
$maxDelayDS †	宛先からソースへの最大遅延 (秒単位)。
$maxDelaySD †	ソースから宛先への最大遅延 (秒単位)。
$maxNegJitterDS †	宛先からソースへの負の最大ジッター値 (秒単位)。
$maxNegJitterSD †	ソースから宛先への負の最大ジッター値 (秒単位)。
$maxPosJitterDS †	宛先からソースへの正の最大ジッター値 (秒単位)。
$maxPosJitterSD †	ソースから宛先への正の最大ジッター値 (秒単位)。
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minDelayDS †	宛先からソースへの最小遅延 (秒単位)。
$minDelaySD †	ソースから宛先への最小遅延 (秒単位)。
$minNegJitterDS †	宛先からソースへの負の最小ジッター値 (秒単位)。
$minNegJitterSD †	ソースから宛先への負の最小ジッター値 (秒単位)。
$minPosJitterDS †	宛先からソースへの正の最小ジッター値 (秒単位)。
$minPosJitterSD †	ソースから宛先への正の最小ジッター値 (秒単位)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$numNegJitterDS	宛先からソースへの負のジッター値の数。
$numNegJitterSD	ソースから宛先への負のジッター値の数。
$numOW	遅延に対する片方向操作の数。
$numPosJitterDS	宛先からソースへの正のジッター値の数。

表 119. Jitter プローブ・エレメント (続き)	
エレメント	**説明**
$numPosJitterSD	ソースから宛先への正のジッター値の数。
$numRTT †	正常終了した往復の数。
$packetLateArrival †	タイムアウトの後で到着したパケット数。
$packetLossDS †	宛先からソースの間で失われたパケットの数。
$packetLossSD †	ソースから宛先の間で失われたパケットの数。
$packetMIA†	宛先が不明な場合に失われたパケットの数。
$packetOutOfSequence†	正しくない順序で返されたパケットの数。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType†	Jitter
(ResponderRouter)	Jitter 要求に応答するために使用されるルーターの名前。
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
(SourceRouter)	ジッター要求を送信するために使用されるルーターの名前。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
$sum2DelayDS	宛先からソースへの遅延の二乗の合計。
$sum2DelaySD	ソースから宛先への遅延の二乗の合計。
$sum2NegJitterDS	すべての負のジッター値の二乗の合計。
$sum2NegJitterSD	すべての負のジッター値の二乗の合計。
$sum2PosJitterDS	すべての正のジッター値の二乗の合計。
$sum2PosJitterSD	すべての正のジッター値の二乗の合計。
$sum2Rtt†	往復値の二乗の合計 (秒単位)。
$sumDelayDS	宛先からソースへの遅延の合計 (秒単位)。
$sumDelaySD	ソースから宛先への遅延の合計 (秒単位)。
$sumNegJitterDS	すべての負のジッター値の合計 (秒単位)。
$sumNegJitterSD	負のジッター値の合計 (秒単位)。
$sumPosJitterDS	すべての正のジッター値の合計 (秒単位)。
$sumPosJitterSD	すべての正のジッター値の合計 (秒単位)。
$sumRTT	すべての往復の合計 (秒単位)。
$targetHost	テストされているサービスのホストの名前。
$tos (Tos)	Type of Service 値。

表 *119. Jitter* プローブ・エレメント (続き)	
エレメント	**説明**
$vpn (¥Vpn)	VPN の名前。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

SNA-Echo プローブ
SNA-Echo プローブ (SNA-RU-Echo、SNA-LU0-Echo、SNA-LU2-Echo、SNA-LU62-Echo、および SNA-LU62Native-Echo) は、次の表にリストされているエレメントを生成します。

次の表では、JSNA-Echo プローブ・エレメントについて説明します。

表 *120. SNA-Echo* プローブ・エレメント	
エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errTotal †	エラーのあるパケットの合計数。
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$numRTT†	正常終了した往復の数。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType (ProbeType)	sna-*name*-echo
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
(SourceRouter)	SNA 要求を送信するために使用されるルーターの名前。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
$sumOfRTT (TotalRTT)	合計往復時間 (秒単位)。
(TargetHost)	SNA エコー要求のホスト・ターゲット。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

UDP-Echo プローブ
UDP-Echo プローブは、次の表にリストされているエレメントを生成します。

次の表では、UDP-Echo プローブ・エレメントについて説明します。

表 *121. UDP-Echo* プローブ・エレメント

エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errBusies	前回の ping が完了しなかったために失敗した ping の数。
$ErrDrops	内部リソースが使用不可であったために失敗した ping の数。
$errTimeouts	タイムアウトによって失敗した ping の数。
$errTotal † (ErrorTotal)	エラーのあるパケットの合計数。
$errVerifies	受信したデータが予期されるデータと異なっていたために失敗した ping の数。
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$numRTT †	正常終了した往復の数。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType	udp-echo
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion* (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
$sumOfRTT	合計往復時間 (秒単位)。
$targetHost (Host)	モニターされているサービスのホスト名。
$tos (Tos)	Type of Service 値。
$vpn (Vpn)	VPN の名前。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

VOIP プローブ
VOIP プローブは、Jitter プローブと同じエレメントを生成します。その他に、次の表にリストされているエレメントも生成します。

以下の表では、VOIP プローブ・エレメントについて説明します。

表 *122. VOIP* プローブ・エレメント

エレメント	**説明**
$authProto	エレメントの作成時に指定された認証プロトコル。
$avgRTT † (AverageRTT)	平均往復時間 (秒)。
$community	SNMP 要求を SAA に送信するために使用されたコミュニティー。
$errTotal † (ErrorTotal)	エラーのあるパケットの合計数。
$ICPIF †	ICPIF 値。
$maxRTT † (MaximumRTT)	最大往復時間 (秒)。
$minRTT † (MinimumRTT)	最小往復時間 (秒)。
$MOS †	テストの平均オピニオン評点 (MOS) の値。
$port	SAA への接続に使用されたポート。
$privProto	エレメントの作成時に指定されたプライバシー・プロトコル。
$probeType †	voip
(ResponderRouter)	VOIP 要求に応答するために使用されるルーターの名前。
$securityName	エレメントの作成時に指定されたセキュリティー・ユーザー名。
$snmpVersion (SnmpVersion)	SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。
(SourceRouter)	VOIP 要求を送信するために使用されるルーターの名前。
(Tos)	Type of Service 値。
(Vpn)	VPN の名前。

注 : † は、そのエレメントをサービス・レベル分類で使用できることを示しています。

状況メッセージ
SAA モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

次の表では、SAA モニターの状況メッセージについて説明します。

表 *123. SAA* モニターの状況メッセージ

メッセージ	**説明**
Success	プローブ操作は正常終了しました。

表 123. SAA モニターの状況メッセージ (続き)	
メッセージ	**説明**
Operation failed	プローブ操作は失敗しました。
Invalid status	プローブ操作は、無効な状況で失敗しました。

SIP モニター

SIP モニターは、エンドポイントの登録および認証に要する時間など、Session Initiation Protocol (SIP) サーバーの可用性を検査します。 モニターは、SIP 要求および SIP 応答をモニターできるように SIP セッションを開始します。

次の表に、SIP モニター・ファイルをリストしています。

表 124. *SIP* モニター・ファイルの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_sip
プロパティー・ファイル	$ISHOME/etc/props/sip.props
ルール・ファイル	$ISHOME/etc/rules/sip.rules
ログ・ファイル	$ISHOME/log/sip.log

SIP モニターを構成するためのガイドライン

SIP モニターは、SIP サーバーを介して SIP 対応のデバイスの URI に要求を送信し、SIP デバイスからの応答をやはり SIP サーバーを介して受信することにより、SIP サーバーの可用性をテストします。

SIP モニターは、ユーザー・エージェント・クライアント (UAC) として機能し、SIP サービスをテストするために使用される接続を開始します。ソフト・フォンを実行するコンピューターまたはメッセージ・バンクなどの任意の SIP 対応デバイスを、受信側または呼び出しのターゲットであるユーザー・エージェント・サーバー (UAS) として使用できます。

SIP サーバーのテスト時に、モニターは以下の一連のアクションを実行します。

1. プロファイル・エレメントで提供された資格情報を使用して SIP サーバーに登録します。
2. UAS に OPTIONS 要求を送信します。
3. UAS に INVITE 要求を送信します。

   UAS が要求を受け入れた場合、成功というテスト結果を登録します。
4. UAS に BYE 要求を送信し、UAS との接続を終了させます。
5. SIP サーバーの登録を抹消します (即時に失効します)。

モニターは、テストで実行されたそれぞれのアクションの所要時間を記録します。

プロパティー

次の表では、SIP モニターに固有のプロパティー・オプションについて説明しています。

表 125. *SIP* モニターのプロパティー・オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
ShowZeroes	0|1	ゼロの値を持つ SIP 統計の表示を指定します。 0 - 無効 1 - 有効

表 125. SIP モニターのプロパティー・オプション (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
Transports	ストリング	ローカル・プロトコル・ポート・トランスポートをスペースで区切ってリストします。プロトコルの場合、これは TCP または UDP です。ワイルドカードのポート番号が許可されます。 デフォルト: UDP:*

暗号スイート

SSLCipherSuite プロパティーは、SIP モニターによって使用される暗号スイートを指定します。詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

SIP モニター・サービス・テストの構成

SIP モニター構成パラメーターを使用して、サービス・テストを定義します。

表 126. SIP モニター構成	
フィールド	**説明**
サーバー	テストされるサーバーの名前を指定します。例: `sip1.mycompany.com`
serverport	SIP モニターがテスト対象のサーバーにアクセスする場合に使用するポート。
ユーザー名	呼び出しを行う SIP モニターの内線番号またはアカウント ID を指定します。例: `jblogg`
target	呼び出しを行うために使用される SIP 対応のデバイスの内線番号を指定します。例: `5551234`
パスワード	ユーザー名のパスワードを指定します。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。例: `SIP monitor`
proxy	プロキシー・サーバーのホスト名。例: `proxy.mycompany.com`
proxyport	SIP モニターがプロキシー・サーバーにアクセスする場合に使用するポート。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

サービス・レベル分類

サービス・レベル分類は、SIP を介して提供されるサービス・レベルを決定するためのルールを定義します。

SIP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、SIP モニターは、SIP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

次の表では、SIP モニターの追加エレメントについて説明します。

表 *127. SIP* モニター・エレメント

エレメント	**説明**
$AcceptReg	受け入れられた SIP 登録要求の数。
$AuthTime* (認証時間)	SIP モニターと SIP 対応のデバイスの両方を許可するために要した時間。
$authAttempts	モニターがその資格情報を組み込むために要求を再送信する必要があった回数。
(呼び出しセットアップ時間)	呼び出しをセットアップするために要した時間。
$Invalid	送信および受信された無効な要求の数。
$InvalidReg	無効な SIP 登録要求の数。
$lastMethod	モニターが最後に認識したメソッド (BYE または ACK を除く)。
$lastSequence [METHOD]	メソッドで受信した最後のシーケンス。
$lastStatus [METHOD]	メソッドまたは全体で受信した最後の状況。
$method *METHOD*	メソッドで表示されるメッセージの集計。
$optionsTime* (オプション時間)	オプション変更 (OPTIONS から 200 OK) のネゴシエーションに要した時間。
$postDialTime* (ダイヤル後時間)	ダイヤル後にリンギング・シグナルを受信するために要した時間 (INVITE から 180 Ringing)。

表 *127. SIP* モニター・エレメント *(続き)*	
エレメント	**説明**
$RegTime* (登録時間)	SIP モニターと SIP 対応のデバイスの両方を登録するために要した時間。
$registrationTime	サーバーに登録するために要した時間 (REGISTER から 200 OK)。
$Requests	受信および送信された SIP 要求メッセージの数。
($RequestsSent)	送信された SIP 要求メッセージの数。
`${request|response}[Sent|Received|Transmitted|Total][METHOD][STATUS]`	さまざまなカテゴリーで表示されるメッセージの集計。例えば、`requestSentINVITE = 1,responseReceived = 10`、および `responseReceivedBYE200 = 1` です。
$Responses	受信および送信された SIP 応答メッセージの数。
($ResponseReceived)	受信した SIP 応答メッセージの数。
$sessionAnswered	• 1 - 呼び出しが応答された場合 • 0 - 呼び出しに応答がない場合
$sessionCreated	• 1 - セッションが確立された場合 • 0 - セッションが確立されなかった場合
$sessionTerminated	• 1 - セッションが終了する場合 • 0 - セッションが終了しない場合
$shutdownTime* (ShutdownTime)	接続を終了するために要した時間 (BYE から 200 OK)。
$terminatedReason* (終了の原因)	接続が閉じられた原因。
(ユーザー名)	SIP サーバーにログインするために使用するユーザー名。
(ターゲット)	セッションを開くターゲット。

状況メッセージ

SIP モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

次の表では、SIP モニターの状況メッセージについて説明します。

表 *128. SIP* モニターの状況メッセージ	
メッセージ	**説明**
`Register timed out`	モニターはサーバーへの登録に失敗しました。
`Invite timed out`	INVITE メッセージがタイムアウトになりました。
`OK`	要求および応答は正常終了しました。
`n operation status description`	• *n* はメッセージ番号の数列です。 • operation はメッセージのタイプです。 • status は状況コードです。 • description は、非暗号化テキストによる状況の説明です。 例えば、`1 INVITE 200 OK` などです。

SIP 応答

SIP モニターは以下のタイプの応答をサポートします。それぞれの応答には 3 桁のコードが指定されています。

- 通知応答 (100 から 199)
- 成功応答 (200 から 299)
- リダイレクト応答 (300 から 399)
- クライアント障害応答 (400 から 499)
- サーバー障害応答 (500 から 599)
- グローバル障害応答 (600 から 699)

次の表では、共通の SIP 応答をリストしています。

表 *129.* 共通の *SIP* 応答	
応答	**説明**
100 試行中 (Trying)	メッセージは、SIP 対応のデバイスによって受信されましたが、まだ処理されていません。
180 呼び出し中 (Ringing)	メッセージは、SIP 対応のデバイスによって受信および処理されました。デバイスはリンギングによってユーザーにアラートを出します。
200 OK	このコードは、メソッドの正常終了時に返されます。例えば、呼び出しがサーバーに登録されたり、ユーザーが呼び出しに応答した場合です。
401 未許可	ユーザーは許可されていません。
407 プロキシー認証が必要です	このコードは 401 に似ていますが、最初にユーザーの認証が必要であることを示しています。
408 要求タイムアウト	ユーザーは呼び出しに応答しませんでした。

SIP 応答の完全なリストについては、RFC3261 を参照してください。

***SMTP* モニター**

SMTP モニターは、IMAP4 モニターまたは POP3 モニターと連動して、電子メール・サービスのパフォーマンスをテストします。

以下の表に、SMTP モニター・ファイルをリストしています。

表 130. SMTP モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_smtp
プロパティー・ファイル	$ISHOME/etc/props/smtp.props
ルール・ファイル	$ISHOME/etc/rules/smtp.rules
ログ・ファイル	$ISHOME/log/smtp.log

SMTP モニターをモニターするためのガイドライン

SMTP モニターは、POP3 モニターまたは IMAP4 モニターと連動して作動します。定期的にターゲット・サーバー上のメールボックスへ電子メールを送信し、電子メールの送信要求の発行にかかった時間を記録します。その後、POP3 モニターまたは IMAP4 モニターがメールボックスからメッセージを読み取り、それらを使用して電子メール・サービスの応答時間と可用性を計算します。

注 : SMTP モニターは、POP3 モニターまたは IMAP4 モニターと連動して作動します。定期的にターゲット・サーバー上のメールボックスへ電子メールを送信し、電子メールの送信要求の発行にかかった時間を記録します。その後、POP3 モニターまたは IMAP4 モニターがメールボックスからメッセージを読み取り、それらを使用して電子メール・サービスの応答時間と可用性を計算します。

メールボックス

任意の既存のメールボックスに電子メール・メッセージを送信するようにモニターを構成できます。そのメールボックスが実際のユーザーに属するものであってもかまいません。ただし、サービス・テスト用の特別なメールボックス・アカウントを作成することをお勧めします。email パラメーターでは、受信側メールボックスを指定します。 デフォルトでは、件名の行が SMTP Monitor Test Message であるテスト・メッセージがモニターによって送信されます。必要な場合は、メールボックス名を使用せずに SMTP プロファイル・エレメントを構成できます。この構成では、モニターは、SMTP サービスが接続を受け入れていることのみをチェックします。

機密保護機能のあるメール

SMTP モニターは、機密保護機能のあるメール・サービスへの接続をサポートします。 SSL/TLS か、または STARTTLS コマンドを使用して接続できます。SMTP モニター・エレメントを定義する際に、「セキュリティー・タイプ (Security Type)」フィールドを使用して適切なセキュリティーを選択してください。メール・サーバーで、SSL 暗号化のクライアント・サイド証明書が必要な場合は、SSLname プロパティーまたはコマンド行オプションを使用して、証明書ファイル、鍵格納ファイル、鍵パスワード、および暗号スイートを指定してください。

クライアント・サイド証明書

SMTP モニターを使用すると、相互認証のためにクライアント・サイド証明書を必要とするサーバーをモニターできます。プロファイル・エレメントの作成時に、SSL 証明書ファイル、鍵格納ファイル、および鍵パスワードを指定します。証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。証明書が別の形式である場合は、PEM 形式に変換する必要があります。証明書は openSSL などのソフトウェアを使用して変換できます。openSSL は、http://www.openssl.org から入手できます。

注 : すべてのプロファイル・エレメントで常に同じ証明書、鍵、およびパスワードを使用する場合は、作成するプロファイル・エレメントごとにそれらを定義するのではなく、モニター・プロパティーを使用して指定してください。

SMTP モニター・サービス・テストの構成

SMTP モニター構成パラメーターを使用して、サービス・テストを定義します。モニターの構成時に、タイムアウト・パラメーターとポーリング間隔パラメーターのデフォルト値が表示されます。タイムアウトのデフォルトは 30 秒、ポーリング間隔のデフォルトは 300 秒です。値が指定されない場合、以下の表にリストされたその他のデフォルトは、構成時には表示されませんが、構成の詳細を保存した時点で適用されます。

表 131. SMTP モニター構成	
フィールド	**説明**
サーバー	メール・サーバーの IP アドレス。例: mail.mycompany.com
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	メール・サーバーのポート番号。 デフォルト: 25 SMTP サーバー以外のサーバーを使用する場合は、サーバーに接続するポートを更新します。 例えば、Microsoft Exchange 用に SSL を介して IMAP4 サーバーを使用する場合、ポート 465 を指定します。
セキュリティー・タイプ	メール・サーバーとの間で開かれたセキュア接続のタイプ: • NONE - セキュリティーを使用せずに接続します。 • SSL - SSLv2 hello を送信してから、SSLv2、SSLv3、または TLSv1 をネゴシエーションします。 • STARTTLS - セキュリティーを使用せずに接続し、STARTTLS コマンドを実行してから、TLSv1 を介して接続を確立します。 デフォルト: NONE
ユーザー名	SMTP サーバーにログインするために使用されたユーザー名。 PLAIN または CRAM-MD5 認証で使用されます。
パスワード	SMTP サーバーにログインするために使用されたパスワード。 PLAIN または CRAM-MD5 認証で使用されます。
認証タイプ	SMTP サーバーに対してモニターを認証する方式。 使用可能なオプションは以下のとおりです。 • NONE - 認証は試行されない • PLAIN - 非暗号化テキストのユーザー名とパスワードによる認証 • CRAM-MD5- CRAM-MD5 認証を使用 デフォルト値は NONE です。
共有秘密鍵	CRAM-MD5 認証の共有秘密鍵。
E メール	SMTP モニターおよび POP3 モニターが使用するメールボックスの電子メールアドレス。
タイムアウト	SMTP サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	再テストの回数。この回数だけ再テストされると、失敗であることが示されます。 デフォルト: 0

表 131. SMTP モニター構成 (続き)	
フィールド	**説明**
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

注: 10 分間隔でメール・サーバー mail.mycompany.com への接続を試みることによって、このメール・サーバーの可用性をモニターします。接続タイムアウトは 30 秒とし、接続が失敗した場合は、5 秒ずつ間隔をあけて 3 回再試行します。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、SMTP モニターは、SMTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、SMTP モニターの追加エレメントについて説明します。

表 132. SMTP モニター・エレメント	
エレメント	**説明**
$authentication	SMTP サーバーで必要とされるユーザー認証方式のタイプ (Standard または APOP)。
$bytesPerSec	1 秒当たりの平均転送バイト数。
$bytesTransferred	アップロードまたはダウンロードされたバイト数。
$connectTime* (ConnectTime)	SMTP サーバーに接続するために要した時間。
$email* (EmailAddress)	モニターがテスト電子メールを送信する宛先のメールボックスの電子メール・アドレス。
$lookupTime* (LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$port* (Port)	サービスがモニターされているポート。
$responseTime* (ResponseTime)	接続が作成されてからテスト電子メールの最初のバイトを SMTP サーバーに送信できるようになるまでの時間。
$security	プロファイル・エレメントの「**securitytype**」フィールドで設定された、メール・サーバーとの間で開かれたセキュア接続のタイプ (NONE、STARTTLS、または SSL)。
$SSLHandshakeTime* (SslHandshakeTime)	SSL 接続を確立するために要した時間。
$status* (ResultStatus)	SMTP サーバーによって返された状況コード。

表 132. SMTP モニター・エレメント (続き)	
エレメント	**説明**
$uploadTime* (UploadTime)	ファイルをアップロードするために要した時間。
$user* (SmtpUser)	モニターが SMTP サーバーにログインするために使用するユーザー名 (アカウント名)。

状況メッセージ

SMTP モニターは、IBM Application Performance Management を使用している場合は $message エレメントで状況メッセージを示します。これらのメッセージは、テストの結果を示します。

プロパティー

以下の表では、SMTP モニターに固有のプロパティーについて説明します。

表 133. SMTP モニターのプロパティーおよびコマンド行オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
MailMessage Path	ストリング	テスト電子メールで送信するテキストが入ったファイルへのパス。これが設定されない場合、デフォルトのメッセージが送信されます。
Originator	ストリング	テスト電子メールの送信時に設定する From フィールドを指定します。これが IMAP4 モニターの対応するストリングと合致していることを確認してください。 デフォルト: SMTP-Monitor
SSLCertificate File	ストリング	SMTP エレメントの作成時に証明書が明示的に指定されなかった場合に使用されるデジタル証明書のパスおよびファイル名。 パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー ($ISHOME/platform/arch/bin) からの相対位置にあるものと解釈します。
SSLCipherSuite	ストリング	SSL 操作で使用する暗号スイート。指定可能な値の説明については、『暗号スイート』を参照してください。 デフォルト: RC4:3DES:DES:+EXP
SSLDisableTLS	整数	レガシー・サポートで TLSv1 を使用不可にします。 デフォルト: 0 - TLSv1 を使用可能にします。 1 - TLSv1 を使用不可にします。
SSLKeyFile	ストリング	SSL 秘密鍵が入ったファイル。
SSLKeyPassword	ストリング	SSL 秘密鍵の暗号化に使用されるパスワード。
UseBody	整数	モニターがメール・メッセージにトラッキング情報を書き込む場所 (メール・ヘッダーまたはメール本文) を指定します。 デフォルト値は 0 です。この場合、情報はメール・ヘッダーに書き込まれます。 1 - 情報はメール本文に書き込まれます。

暗号スイート

SSLCipherSuite プロパティーは、SMTP モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

SNMP モニター

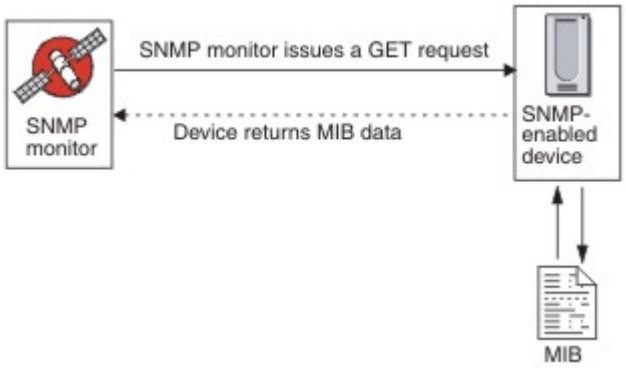
SNMP モニターは、パフォーマンスおよび障害データを調べるために SNMP 対応デバイスをテストします。

以下の表に、SNMP モニター・ファイルをリストしています。

表 *134. SNMP* モニターの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_snmp
プロパティー・ファイル	$ISHOME/etc/props/snmp.props
ルール・ファイル	$ISHOME/etc/rules/snmp.rules
ログ・ファイル	$ISHOME/log/snmp.log

SNMP モニター構成のガイドライン

SNMP モニターは、デバイスの MIB に格納された 1 つ以上のオブジェクトに対する SNMP GET 要求を送信することによって、SNMP 対応のデバイスからデータを取得します。するとデバイスは、SNMP モニターに MIB データを返します。SNMP モニターは、SNMP バージョン 1、2c、および 3 をサポートします。



プロパティー

次の表では、SNMP モニターに固有のプロパティー・オプションについて説明しています。

表 *135. SNMP* モニターのプロパティー・オプション		
プロパティー名	**プロパティー・パラメーター**	**説明**
InvalidBps Value	整数	1 つのデータ・ポイントのみが使用可能である場合にビット/秒 (Bps) 値の計算のために置換される整数値を指定します。
MibDir	ストリング	モニターが使用する MIB 文書が入っているディレクトリーを指定します。 デフォルト: $ISHOME/mibs。

表 *135. SNMP* モニターのプロパティー・オプション (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
StripQuotes	0|1	整数データから引用文字をストリップします。 0 - 無効 1 - 有効
Rollover Threshold	整数	ロールオーバーが生じた場合、ルーターのリセットが行われるまでにデルタが一致または超過している必要がある値。 デフォルト: 0 (ロールオーバーは行われません)

SNMP モニター・サービス・テストの構成

SNMP モニター構成パラメーターを使用して、サービス・テストを定義します。

表 *136. SNMP* モニター構成	
フィールド	**説明**
サーバー	SNMP GET 要求を送信する宛先のサーバー。
objectgroupname	GET 要求に組み込む OID のグループのテキスト名。
communitystring	クライアント上の SNMP サーバーの SNMP 読み取り/書き込みコミュニティー・ストリング。 **注 :** コミュニティー名では脱字記号文字 (^) を慎重に使用してください。詳しくは、コミュニティー名を参照してください。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポート	使用するサーバー上のポート。 デフォルト: 161
バージョン	使用する SNMP バージョン: 1 - SNMPv1 2 - SNMPv2c 3 - SNMPv3 デフォルト: 1
securityname†	SNMP セッションのユーザー名。
authenticationphrase†	ユーザーの認証パスワード。
privacyphrase†	ユーザーのプライバシー・パスワード。
authenticationprotocol†	ユーザーの認証に使用するプロトコル。 • MD5 • SHA1 デフォルト: MD5

表 136. SNMP モニター構成 (続き)

フィールド	説明
privacyprotocol†	セッションの暗号化に使用するプロトコル。 デフォルト: DES
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 20
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
再試行	モニターがサーバーへの接続を再試行する回数。この回数だけ再試行すると、モニターは終了します。 デフォルト: 0
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10
ホスト名ルックアップ設定	どちらの IP バージョン (IPv6 または IPv4) が、提供されたホスト名に適用されるか判別します。オプションは以下のとおりです。 • default を指定すると、モニター全般用に設定されているプロパティーを使用するように、モニターが設定されます。これはデフォルトです。 • 4Then6 を指定すると、IPv4 の次に IPv6 が選択されます。IPv4 アドレスが使用可能な場合は、IPv4 アドレスが使用されます。IPv4 アドレスが見つからない場合は、IPv6 アドレスが使用されます。 • 6Then4 を指定すると、IPv6 の次に IPv4 が選択されます。IPv6 アドレスが使用可能な場合は、IPv6 アドレスが使用されます。IPv6 アドレスが見つからない場合は、IPv4 アドレスが使用されます。 • 4Only を指定すると、IPv4 のみが選択されます。IPv4 アドレスのみが使用されます。IPv4 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Only を指定すると、IPv6 のみが選択されます。IPv6 アドレスのみが使用されます。IPv6 アドレスがない場合は、ポーリングでエラーが返されます。 • 6Or4 を指定すると、IPv4 または IPv6 が選択されます。ホスト名から最初に返されたアドレスが使用されます。
† SNMPv3 のみで適用されます。	

コミュニティー名

Internet Service Monitoring は、ターゲット・デバイスへの情報の送信時に脱字記号 (^) をエスケープ文字として使用します。コミュニティー名に脱字記号が含まれる場合、ルーターで正しい名前が使用されるためには、2 つの脱字記号を並べて入力する (^^) 必要があります。例えば、コミュニティー名 a $^&b がデバイスに送信されたときに正しくなるようにするには、a$^^&b を使用します。

サービス・レベル分類

サービス・レベル分類は、サービス・レベルを決定するためのルールを定義します。

SNMP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して **$message** エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。
- oidName は、OID グループで定義された「OID 名」フィールドの MIB オブジェクトに割り当てられた名前です。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、SNMP モニターは、SNMP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

以下の表では、SNMP モニターの追加エレメントについて説明します。

表 *137. SNMP* モニター・エレメント

エレメント	**説明**
$community	クライアント上の SNMP サーバーの SNMP コミュニティー・ストリング。
$numOids	照会で使用される OID の数。
$oidGroupName* (OidGroup)	OID グループの名前。 OID グループには、モニターがポーリングしている OID が含まれています。
$oidName*0 to n* (OIDName*Zero to Nine*)	OID グループ内の最初から最後の MIB オブジェクトの名前。Netcool/OMNIbus を使用している場合には番号で示され、IBM Application Performance Management を使用している場合には英字テキスト (zero から nine) で示されます。
$oidNames	縦線 (|) で区切られた各 OID の名前。
$oidReturnValues*0 to n* (snmpResult*Zero to Nine*)	OID グループ内の最初から最後の MIB オブジェクトに対する SNMP GET コマンドによって返されたデータ。これは、Netcool/OMNIbus を使用している場合には番号で示され、IBM Application Performance Management を使用している場合には英字テキスト (zero から nine) で示されます。
$oidUnit*0 to n*	数値によって示される、OID グループ内の最初から最後の MIB オブジェクトの単位。
$oidUnits	縦線文字 (|) によって区切られた各 OID の単位。
$port	サービスがモニターされているポート。

表 *137. SNMP* モニター・エレメント *(続き)*	
エレメント	**説明**
$snmpVersion* (SnmpVersion)	プロファイルで構成された SNMP パケットの送信に使用された SNMP のバージョン (バージョン 1、2c、または 3)。

状況メッセージ

SNMP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、SNMP 状況メッセージについて説明します。

表 *138. SNMP* モニターの状況メッセージ	
メッセージ	**説明**
Successful Get	SNMP エージェントの照会は正常に実行されました。
Failed to open snmp sessions SNMP session - start failed	SNMP セッションを初期化できません。
Error in packet	有効な SNMP パケットを作成できません。
Timed out while waiting for response	SNMP エージェントから応答が受け取られていません。
Internal Error	モニターで内部エラーが発生しました。詳細については、IBM テクニカル・サポートにお問い合わせください。
Error Processing OID	OIDs の 1 つの処理中にエラーが発生しました。
ERROR: Too Many OIDs	モニターが一度に要求するように設定された OID の数が多すぎます。最大は 100 です。
ERROR: PDU received mismatch with PDU sent	モニターが受信したプロトコル・データ単位 (PDU) が、サーバーに送信された PDU と合致しませんでした。

SOAP モニター

SOAP モニターは、SOAP インターフェース (SOAP 1.0 および 1.1) の可用性と応答時間を検査します。また、SOAP 入力 (要求) および SOAP 出力 (応答) の妥当性もモニターします。

SOAP モニターでは、次のメッセージ・エンコード・スタイルがサポートされています。

- RPC - エンコード
- ドキュメント - リテラル Unwrapped
- ドキュメント - リテラル Wrapped

以下の表に、SOAP モニター・ファイルをリストしています。

表 139. SOAP モニター・ファイルの概要	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_soap
プロパティー・ファイル	$ISHOME/etc/props/soap.props
ルール・ファイル	$ISHOME/etc/rules/soap.rules
ログ・ファイル	$ISHOME/log/soap.log

SOAP モニターを構成するためのガイドライン

SOAP モニターは、ターゲット SOAP インターフェースに 1 組の入力が入った要求を送信して、インターフェースから応答を受信し、受信した応答に含まれる出力を分析することによって、SOAP サービスの操作をテストします。要求が SOAP インターフェースに送信されると、要求は成功または失敗のいずれかになります。応答が受信されて、応答メッセージの値が指定された出力値と一致する場合、要求は成功します。応答が受信されないか、応答が受信されてもメッセージの値が出力値と一致しない場合、要求は失敗します。

要求と応答に含まれる SOAP 入力および出力は、テストされている SOAP サービスの機能によって異なるため、SOAP サービスのテストを設計する際には、そのサービスに合った入力と出力を指定してください。 入力は、送信されるデータの名前と、それらに割り当てられた入力値からなります。出力は、受信されるデータの名前と、それらの予期される出力値からなります。これらのデータ名は、SOAP モニターを構成したときに指定したローカルの Web サービス記述言語 (WSDL) ファイルから取得したものです。入力データ名および出力データ名は、WSDL ファイル内の名前とデータ型に一致している必要があります。また、データ名の順序も WSDL ファイルと同じでなければなりません。名前が一致しないか、順序が正しくない場合、モニターが SOAP インターフェースのポーリングを 試みると、エラー・メッセージが生成されます。

入力のフォーマットは以下のとおりです。

```
dataname:datatype=assigned_value, dataname:datatype=assigned_value, ...
```

出力のフォーマットは以下のとおりです。

```
dataname:datatype=expected_value, dataname:datatype=expected_value, ...
```

SOAP データ型

SOAP モニターは、単純データ型、配列データ型、およびユーザー定義のデータ型をサポートします。単純データ型には、整数、ストリング、およびブール値があります。配列には、単純データ型が含まれることも、その他の配列やユーザー定義のデータ型が含まれることもあります。

表 140. 単純データ型			
単純データ型			
anyURI	float	language	Qname
boolean	gDay	long	short
byte	gMonth	Name	ストリング
date	gMonthDay	NCName	time
dateTime	gYear	negativeInteger	token
decimal	gYearMonth	NMTOKEN	unsignedByte

表 *140.* 単純データ型 *(続き)*			
単純データ型			
double	ID	NMTOKENS	unsignedInt
duration	IDREFS	nonNegativeInteger	unsignedLong
ENTITIES	int	nonPostiveInteger	unsignedShort
ENTITY	整数	normalizedString	

SOAP 認証

モニター対象の SOAP インターフェースに 基本 HTTP 認証が必要な場合は、Internet Service Monitoring 構成ツールを使用する際に SOAP プロファイル・エレメントのインターフェースにアクセスするための資格情報を指定します。

必要な SOAP 認証パラメーターを設定するには、以下のようにします。

1. Internet Service Monitoring 構成ツールで、認証情報を追加するプロファイル・エレメントを選択します。
2. 「**拡張**」タブで、username パラメーターの「**値**」フィールドをクリックし、必要な値を入力します。
3. password パラメーターの「**値**」フィールドをクリックし、必要な値を入力します。パスワードは暗号化されます。
4. 「**OK**」をクリックします。

認証が必要なくなったら、**username** および **password** パラメーターの値を削除してください。

プロパティー

次の表では、SOAP モニターに固有のプロパティー・オプションについて説明しています。

表 *141. SOAP* モニターのプロパティー・オプション			
プロパティー名	**プロパティー・パラメーター**	**説明**	**デフォルト**
SoapParser	ストリング	XML 構文解析ライブラリー。	$ISHOME/platform/$ARCH/bin/AxisXMLParserXerces.dll
SoapTransport	ストリング	SOAP トランスポート・ライブラリー。	$ISHOME/platform/$ARCH/bin/HTTPTransport.dll
SoapChannel	ストリング	SOAP チャネル・ライブラリー	$ISHOME/platform/$ARCH/bin/HTTPChannel.dll
SoapSecureChannel	ストリング	SOAP セキュア・チャネル・ライブラリー。	$ISHOME/platform/$ARCH/bin/HTTPSSLChannel.dll
SoapClientLog	ストリング	追加の SOAP クライアント・ログ・ファイルの名前。	$ISHOME/log/SoapClient.log

暗号スイート

SSLCipherSuite プロパティーは、SOAP モニターによって使用される暗号スイートを指定します。

詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

SOAP モニター・サービス・テストの構成

SOAP モニター構成パラメーターを使用して、サービス・テストを定義します。

表 *142. SOAP* モニター構成

エレメント	**説明**
wsdl	WSDL ファイルのローカル・コピーへのパス。
operation	SOAP 操作の名前。
operationnamespace	SOAP 操作の名前空間。
location	モニターされる SOAP サービスの URL。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
タイムアウト	SOAP サービスの応答を待機する時間 (秒単位)。 デフォルト: `10`
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: `300`
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: `0`
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: `10`
Soap パラメーター	

inputs	SOAP 入力用の、名前、タイプ、および値のフィールド (属性を含む) にアクセスできます。単純、複合、または配列 Soap パラメーターを使用します。例:
	• 単純: `symbol:string="IBM"`
	• 複合:
	`outer:{item1:string,item2:string}(aaa:string='bbb')`
	`={item1(attr:string='ccc')='',`
	`item2(attr:string='ddd',attr2:string='eee')='fff'}`
	この例で、小括弧で囲まれた属性 (太字) はオプションです。
	• 配列: `input:int[]=[1,2,3,4]`

outputs	SOAP 出力用の、名前、タイプ、および値のフィールド (属性を含む) にアクセスできます。単純、複合、または配列の SOAP パラメーターを使用します。 構文について詳しくは、SOAP パラメーター入力の例を参照してください。

サービス・レベル分類

サービス・レベル分類は、SOAP インターフェースによって 提供されるサービス・レベルを決定するためのルールを定義します。

SOAP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対する **$message** エレメントの任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、SOAP モニターは、SOAP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

次の表では、SOAP モニターの追加エレメントをリストしています。

表 *143. SOAP* モニター・エレメント

エレメント	**説明**
(Location)	モニターされる SOAP サービスの URL。
(Operation)	モニターされている SOAP サービスの名前。
$outputMatch	戻り値が出力値と一致する場合は Success、それ以外の場合は Failure。
$responseValueName	SOAP 応答で受信した値の名前。
$soapname	SOAP 応答のコンテナー名。配列またはユーザー定義の複素数データ型にのみ適用されます。
$soaptype	SOAP 応答のコンテナー・タイプ。配列またはユーザー定義の複素数データ型にのみ適用されます。
(WSDL)	WSDL ファイルのローカル・コピーへのパス。

状況メッセージ

SOAP モニターは、IBM Application Performance Management を使用している場合は **ResultMessage** 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

メッセージは、戻り値が出力値と合致する場合は Success、またはエラー・メッセージのいずれかです。エラー・メッセージには、エラーの説明が含まれます。

例

5 分間隔で SOAP インターフェースの 可用性をモニターします。SOAP インターフェースを 使用できない場合は、5 秒間隔でテストを 2 回まで繰り返します。1 と 2 を加算する要求を送信し、応答に 3 という値が含まれていることを確認します。

SOAP プロファイル・エレメントを作成し、次の表に示すフィールドを設定します。

表 *144. SOAP* プロファイル・エレメントの例

構成フィールド	**値**
wsdl	c:¥%ISMHOME%¥etc¥SOAP.wsdl

表 *144. SOAP* プロファイル・エレメントの例 (続き)	
構成フィールド	**値**
operation	add
operationnamespace	http://localhost/SOAP/Calculator
location	http://serverA/SOAP/Calculator
説明	基本的な電卓 SOAP モニター
アクティブ	選択済み
タイムアウト	30
ポーリング	300
失敗時の再テスト	2
再テスト間隔	5
inputs	[in0=1,in1=2]
outputs	[addReturn=3]

TCPPort モニター

TCPPort モニターは、その他のモニターによってテストされないサービスを対象とします。TCP ポートでコマンドまたはストリングを検出して応答します。このモニターは、特にカスタム・サービスのモニターに有用です。

以下の表に、TCPPort モニター・ファイルをリストしています。

表 *145. TCPPort* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_tcpport
プロパティー・ファイル	$ISHOME/etc/props/tcpport.props
ルール・ファイル	$ISHOME/etc/rules/tcpport.rules
ログ・ファイル	$ISHOME/log/tcpport.log

TCPPort モニターを構成するためのガイドライン

TCPPort モニターは、サービスに接続し、サービスから受信したメッセージをモニターし、それに対する応答を送信することによって、TCP ベースのサービスをテストします。

テストを構成するには、そのサービスの通常の対話を構成する、予期されるメッセージおよび応答のシーケンスを定義します。

例えば、telnet サービスの標準的な対話には、以下のシーケンスが含まれます。

- telnet サービスが、ユーザー名の入力を求めるログイン・メッセージを送信します。
- クライアントが、ユーザー名を含む応答を送信します。
- telnet サービスが、パスワードの入力を求めるメッセージを送信します。
- クライアントが、パスワードを含む応答を送信します。
- ログインの試行が成功すると、telnet サービスが、何らかのあいさつメッセージを送信します。

指定されたモニターの **WaitForn** および **Sendn** プロパティーは、予期されるメッセージと、それらのメッセージに対する応答を定義します。モニター・プロパティー・ファイル内のこれらのプロパティーは、モニターが TCP サービスとどのように対話するかを定義します。

- **WaitForn** プロパティーは正規表現です。 モニターは、これを使用して、モニター対象ポートで受信したメッセージと突き合わせます。
- **Sendn** プロパティーは、モニターがポートに書き込むリテラル・ストリングです。

注: 必要な場合は、制御文字の挿入をサポートするテキスト・エディターを使用して、これらのプロパティーに制御文字を挿入できます。

WaitForn プロパティーおよび Sendn プロパティーを定義するための形式は次のとおりです。

```
WaitFor1: 1st received message
Send1: 1st response
WaitFor2: 2nd received message
Send2: 2nd response
...
WaitFor5: 5th received message
Send5: 5th response
```

モニターは、最初に設定解除された **WaitFor** プロパティーに達すると、送受信を停止します。**MonitorDisconnect** プロパティーが 0 に設定されている場合、モニター対象のサービスは、モニターが開いた接続を閉じる必要があります。接続を閉じない場合、モニターは **$message** エレメントで `Timed out waiting to read` というメッセージを報告します。 多くのサービスでは、`quit` コマンドを送信することで接続を閉じることができます。**MonitorDisconnect** が 1 に設定されている場合、最後の Send または WaitFor コマンドが完了した後か、またはタイムアウトに達した後に (どちらが先に発生した場合でも)、モニターは接続を切断します。

TCPPort モニター・サービス・テストの構成

TCPPort モニター構成パラメーターを使用して、サービス・テストを定義します。

表 *146. TCPPort* 構成

フィールド	説明
サーバー	ターゲット・サービスが実行されているシステムの IP アドレス。 例: `server.mycompany.com`
ポート	ターゲット・サービスへの接続に使用するポート。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
タイムアウト	サーバーの応答を待機する時間 (秒単位)。 デフォルト: 30
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300
失敗時の再テスト	失敗が示されるまで再テストする回数。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

注: ホスト `server.mycompany.com` でポート 23 を使用して実行されている `telnet` サービスの可用性をモニターします。資格情報 `user` または `guest` を使用して、サーバーにログインし、ログイン後直ちに接続を閉じます。5 分間隔でテストを実行し、接続の試行に 10 秒のタイムアウトを設定します。

1. TCPPort プロパティー・ファイルに以下のエントリーを追加します。

```
WaitFor1: ".*[Ll]ogin:"
Send1: "user"
WaitFor2: ".*[Pp]assword:"
Send2: "guest"
WaitFor3: ".*%"
Send3: "exit"
```

2. TCPPort モニターを開始または再開します。

正規表現の突き合わせ

最大 50 個の異なる正規表現を入力することによって、ダウンロードされた情報で正規表現検索を実行します。TCPPort モニターは、検索された内容を各正規表現に突き合わせようとします。指定した正規表現に合致するものが見つかった場合は、合致した行 (または、モニターの内部バッファーに収容できるだけの行) が、対応する $regexpMatchn エレメントに戻されます。 正規表現がダウンロードされた情報内で複数回合致した場合は、最初に合致したものだけが戻されます。各正規表現テストの状況は、$regexpStatusn エレメントによって示されます。 正規表現の突き合わせとそれらの状況情報は、サービス・レベル分類の基準として使用できます。

詳しくは、333 ページの表 50 を参照してください。

モニター・エレメント

以下の表では、TCPPort モニターの追加エレメントについて説明します。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

すべてのエレメントに共通するテスト結果に加えて、TCPPort モニターは、TCPPort サービス・テストに固有のデータを含む一連のテスト結果を生成します。

表 *147. TCPPort* モニター・エレメント

エレメント	説明
$bytesPerSec	1 秒当たりの平均転送バイト数。
$bytesTransferred	アップロードまたはダウンロードされたバイト数。
$connectTime*(ConnectTime)	ターゲット・サーバーとの接続を確立するために要した時間。
$downloadTime*(DownloadTime)	データをダウンロードするために要した時間。
$lastlineThere's	ターゲット・サーバーから受信した最終行の内容。
$lookupTime*(LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$networkError	接続中のネットワーク・エラーが含まれます。
$port* (Port)	モニターが接続を試行したターゲット・サーバー上のポート。
$waitingFor	モニターが一連の待機と送信を完了する前に接続が終了した場合、このエレメントには、最後の WaitFor プロパティーの内容が含まれます。

状況メッセージ

TCPPort モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、TCPPort 状況メッセージについて説明します。

表 *148. TCPPort* モニターの状況メッセージ	
メッセージ	**説明**
OK	要求は正常に処理されました。
Timed out waiting to read/ write	サーバーへのデータ接続は確立されましたが、応答しません。
Connection closed unexpectedly	サーバーへの接続は失敗しました。
Connection failed	モニターはサーバーへの接続に失敗しました。詳しくは、ログ・ファイルを参照してください。
Network connect error	ネットワークに問題があります。
Network error whilst reading	

プロパティー

以下の表では、TCPPort モニターに固有のプロパティー・オプションについて説明します。

表 *149. TCPPort* プロパティー		
プロパティー名	**プロパティー・パラメーター**	**説明**
Monitor Disconnect	0|1	モニターが最後の Send または WaitFor コマンドの後で切断することを指定します。 最後のコマンドが Send である場合、モニターはストリングが送信された後で即時に切断します。最後のコマンドが WaitFor である場合、モニターが合致を受信するか、ポーリングのタイムアウトを超過すると、すぐにモニターは切断します。 0 - 無効 (モニターは接続しない) 1 - 有効
OutputDirectory	ストリング	OutputResult が保管される場合に使用する出力ディレクトリーを指定します。 デフォルト: $ISHOME/var
OutputResult	0|1	モニターがサービスから受信するデータを保管することを指定します。 0 - 無効 1 - 有効
Send	*n*	モニターがポートに書き込むリテラル・ストリング。TCPPort モニターを構成するためのガイドラインを参照してください。 n は、1 から 30 の範囲内の番号です。

表 *149. TCPPort* プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
singleLineMatch	0|1	正規表現が合致した場合にモニターが単一行の合致を返すことを指定します。 0 - 無効 (複数行の合致) 1 - 有効 (単一行の合致)
WaitFor	*n*	モニター対象のポートでコマンドまたはストリングを突き合わせるために使用される正規表現。詳しくは、TCPPort モニターを構成するためのガイドラインを参照してください。 n は、1 から 30 までの番号です。

暗号スイート

SSLCipherSuite プロパティーは、TCPPORT モニターによって使用される暗号スイートを指定します。SSL 設定について詳しくは、442 ページの『Internet Service Monitoring での SSL 設定』を参照してください。

TFTP モニター

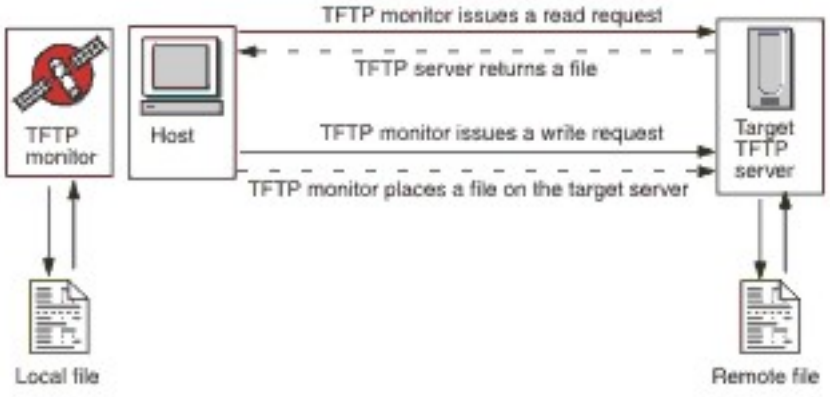
TFTP モニターは、2 つのシステム間の Trivial File Transfer Protocol (TFTP) サービスのパフォーマンスを測定します。

次の表に、TFTP モニター・ファイルをリストしています。

表 *150. FTP* モニターの概要	
モニター・ファイル	**名前またはロケーション**
実行可能ファイル名	nco_m_tftp
プロパティー・ファイル	$ISHOME/etc/props/tftp.props
ルール・ファイル	$ISHOME/etc/rules/tftp.rules
ログ・ファイル	$ISHOME/log/tftp.log

TFTP モニターを構成するためのガイドライン

TFTP モニターは、TFTP READ または WRITE 要求を使用してホスト・システムとターゲット ・サーバーの間でファイルを転送してから、応答時間とデータ転送速度を記録します。このモニターを使用して、TFTP サーバーが稼働しており、許容できる速度でファイルを転送していることを確認してください。



モニターは、ファイルをアップロードする場合には TFTP WRITE 要求 (WRQ) を送信し、ファイルをダウンロードする場合には TFTP READ 要求 (RRQ) を送信します。 TFTP クライアントでは、アップロード操作は PUT、ダウンロード操作は GET です。

TFTP モニターは octet (バイナリー) ファイル転送モードおよび netascii ファイル転送モードをサポートします。

TFTP モニター・サービス・テストの構成

TFTP モニター構成パラメーターを使用して、サービス・テストを定義します。

表 *151. TFTP* モニター構成

フィールド	**説明**
サーバー	ターゲット TFTP サーバー (ファイルの転送先または転送元とするシステム) の IP アドレス。
localfile	GET 操作の場合、このフィールドはファイルのダウンロード先の名前とパスを指定します。 PUT 操作の場合、このフィールドはサーバーにアップロードされるファイルの名前とパスを指定します。
remotefile	GET 操作の場合、このフィールドはサーバーからダウンロードされるファイルの名前とパスを指定します。 PUT 操作の場合、このフィールドは、アップロードされるファイルの、サーバーでの名前とパスを指定します。
説明	TFTP モニターに関する記述情報を提供するテキスト・フィールド。
ポート	TFTP サーバーが使用するポート。 デフォルト: 69
ローカル IP	モニターが TFTP 接続を開くホストのネットワーク・インターフェースの IP アドレス。このフィールドが空の場合、モニターは `IpAddress` プロパティーによって指定されたインターフェースを使用します。
ローカル・ポート	モニターが TFTP 接続を確立するために使用するポート。このフィールドの値が `0` の場合は、モニターが適切なポートを選択します。

表 *151. TFTP* モニター構成 *(*続き*)*	
フィールド	**説明**
コマンド	モニターが使用する TFTP コマンド: • GET - ターゲット・サーバーからモニター・ホストへファイルをダウンロードします。 • PUT - モニター・ホストからターゲット・サーバーへファイルをアップロードします。 デフォルト: GET
transfermode	モニターがファイルを転送するときの形式を指定します。 • OCTET (8 ビット) • NETASCII デフォルト: OCTET
タイムアウト	TFTP サーバーの応答を待機する時間 (秒単位)。 デフォルト: 10
再試行	モニターがファイルの転送を試行する回数。この回数だけ試行すると、モニターは終了します。 デフォルト: 3
ポーリング	ポーリングの間隔 (秒単位)。これにはあまり小さい値を設定しないでください。絶えずポーリングが行われると、サービスよりもポーリングの負荷の方が大きくなる可能性があるためです。 デフォルト: 300
失敗時の再テスト	モニターが最初にテストを失敗した後に TFTP サーバーを再テストする回数。この回数だけ再テストすると、失敗であることが示されます。 デフォルト: 0
再テスト間隔	再テストが失敗してから次の再テストまで待機する時間 (秒単位)。 デフォルト: 10

サービス・レベル分類

サービス・レベル分類は、TFTP サーバーによって提供されるサービス・レベルを決定するためのルールを定義します。

TFTP モニターで選択可能なサービス・レベル分類オプションは、次のとおりです。

```
totalTime
lookupTime
responseTime
transferTime
bytesTransferred
bytesPerSec
checksum
message
```

サービス・レベル分類では、下記のようになります。

- モニター・エレメントの名前を手動で入力することにより、追加のサービス・レベル分類を指定します。この名前は、『モニター・エレメント』セクションのエレメントで示されている名前と一致している必要があります。
- ウィジェットで使用される場合、message は、IBM Application Performance Management サーバーに対して $message エレメントで転送される任意のメッセージにすることができます。 指定可能な値のリストについては、状況メッセージを参照してください。
- オペランドは、ストリングまたは正数です。
- checksum エレメントは、通常、サービス・レベル分類に対して有意義な結果を提供しません。プロファイル・エレメントを作成するときに、その値が不明であるためです。モニターは、テストの進行中に checksum 値を計算します。このエレメントは、ルール・ファイルを使用することによるアラートの機能強化を目的としています。

モニター・エレメント

すべてのエレメントに共通するテスト結果に加えて、TFTP モニターは、TFTP サービス・テストに固有のデータを含む一連のテスト結果を生成します。

次の表では、TFTP モニターの追加エレメントについて説明します。

表 *152. TFTP* モニター・エレメント

エレメント	**説明**
$bytesPerSec* (BytesPerSec)	1 秒当たりの平均転送バイト数。
$bytesTransferred* (BytesTransferred)	アップロードまたはダウンロードされたバイト数。
$checksum	ダウンロードしたデータのチェックサム値。これは、モニターによって生成され、ルール・ファイルを使用してさらに処理するために提供されています。
$command* (TftpCommand)	モニターが発行した TFTP コマンド (GET または PUT)。
$localFile* (TftpLocalFile)	ローカル・ホストに保管されているファイルの絶対パス名。このエレメントは構成ファイルから取られます。
$localIP	モニターが使用するように構成されているローカル IP アドレス。 インターフェースが 1 つのみのシステムではブランクにすることができます。
$lookupTime* (LookupTime)	ホスト・サーバーの IP アドレスを取得するために要した時間。
$remoteFile* (TftpRemoteFile)	リモート・ホスト (FTP サーバー) に保管されているファイルの絶対パス名。 このエレメントは構成ファイルから取られます。
(TransferTime)	ファイルを転送するために要した時間。
(TftpConnection)	モニターがファイルを転送したときに使用した形式。 これは OCTET (8 ビット) または NETASCII です。

状況メッセージ

TFTP モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

表 *153. TFTP* モニターの状況メッセージ	
メッセージ	**説明**
OK	TFTP 要求は正常に処理されました。
FAILED: connect failed	モニターはサーバーへの接続に失敗しました。サーバーが稼働中であることを確認してください。
FAILED: internal tftp monitor error	モニターに問題があり、原因としてメモリー不足が考えられます。
FAILED: A send/wait timed out	TFTP 要求は失敗しました。ネットワークに問題がある可能性があります。
FAILED: An unspecific error condition. The transfer should be aborted	
FAILED: Received a short or malformed packet	
FAILED: local file open/read/write failed	
FAILED: unrecognized status from transfer attempt	

例

ファイル $ISHOME/etc/testfiles/upload.txt を /ism/test/upload_result.txt にアップロードすることによって、TFTP サーバー tftp.mycompany.com の可用性をテストします。netascii モードを使用して、20 分間隔でファイルをアップロードします。

以下の基準を使用してサービス・レベルを分類します。

- アップロードが正常に完了しなかった場合、サービス・レベルは「Failed」です。
- 転送の合計時間が 10 秒を超えた場合、サービス・レベルは「Marginal」です。
- それ以外の場合、サービス・レベルは「Good」です。

TFTP モニター・プロファイル・エレメントを作成し、次の表に示すように構成を設定します。

表 *154. TFTP* プロファイル・エレメントの例	
構成フィールド	**値**
サーバー	tftp.mycompany.com
localfile	$ISHOME/etc/ism/testfiles/upload.txt
remotefile	/ism/test/upload_result.txt
説明	TFTP テスト
アクティブ	選択済み

表 *154. TFTP* プロファイル・エレメントの例 *(続き)*	
構成フィールド	**値**
コマンド	PUT
transfermode	NETASCII
ポーリング	1200

statement	`If (Message != OK) then status Failed`
	`else if (TotalTime > 10) then status Marginal`
	`else status Good`

TRANSX モニター

TRANSX モニターは、他のインターネット・サービス・モニターを使用して実行する一連のアクティビティーを実行することによって、実際のインターネット・ユーザーのアクションをシミュレートします。

例えば、HTTP モニターを使用して Web サイトのページにアクセスし、いくつかのファイルをダウンロードし、POP3 モニターおよび SMTP モニターを使用して送信または受信するように TRANSX を構成します。

以下の表に、TRANSX モニター・ファイルをリストしています。

表 *155. TRANSX* モニター・ファイル	
モニター・ファイル	**名前またはロケーション**
モニター実行可能ファイル	nco_m_transaction
プロパティー・ファイル	$ISHOME/etc/props/transx.props
ルール・ファイル	$ISHOME/etc/rules/transx.rules
ログ・ファイル	$ISHOME/log/transx.log

プロパティー

表 *156. TRANSX* モニター・プロパティー		
プロパティー名	**プロパティー・パラメーター**	**説明**
CompleteTransax	0|1	ステップが失敗した場合でもトランザクションが続行されることを指定します。 • 0 - 無効 (続行しません) • 1 - 有効 (続行します)
DetailedTimings	0|1	TRANSX モニターが各ステップの詳細なタイミングが含まれるデータ・ログを作成することを指定します。データ・ログ内に作成される詳細なタイミングは、事前設定されており、変更できません。 • 0 - 無効 • 1 - 有効

表 156. TRANSX モニター・プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
MultipleEvents	0|1	モニターがトランザクション結果の複数イベントを生成するかどうかを指定します。 • 0 - 無効 (モニターは、すべてのステップの結果および要約の結果を含む 1 つのイベントのみを生成します) • 1 - 有効 (モニターは、トランザクションのステップごとに 1 つのイベント、および最終的な要約イベントを生成します)
StepPause	整数	各トランザクション・ステップ実行の間の一時停止の長さを秒単位で指定します。 一時停止の長さは、トランザクションの $totalTime エレメントの値に影響しません。$totalTime は、トランザクションの $stepXTime エレメントの合計を表します。 デフォルト: 0

以下の表では、SMTP モニターに固有のプロパティーについて説明します。

TRANSX モニターを構成するためのガイドライン

TRANSX モニターは、標準的なユーザー・エクスペリエンスからなるアクティビティーのセットをシミュレートすることによって、サービスをテストします。アクティビティーのセットは **トランザクション** と呼ばれ、トランザクション内の各アクティビティーは **トランザクション・ステップ** と呼ばれます。

TRANSX プロファイル・エレメントはトランザクションを定義します。各トランザクション・ステップは、HTTP などのインターネット・サービス・モニターを、そのステップの操作を実行するように構成します。TRANSX プロファイル・エレメントの「ステップ」タブの「編集」ボタンによってトランザクション・ステップを構成します。

ステップは、その他すべてのプロファイル・エレメントを構成するのと同じ方法で構成されます。例えば、HTTP モニターが関与するステップの構成の詳細には、ヘッド/フォーム・パラメーター、プロキシー・サーバー・パラメーター、正規表現、およびサービス・レベル分類を含めることができます。

TRANSX モニターは、トランザクション内のステップをテストするとき、かかった時間と、ステップのサービスのレベルを記録します。

注 : いずれかのトランザクション・ステップで root 特権を必要とする別のモニター (ICMP など) を使用する場合、TRANSX モニターには root 特権が必要です。

HTTP モニターおよび HTTPS モニターを使用した動的コンテンツの処理

多くの Web サイトでは、動的コンテンツを使用して、セッション・ベースまたは地域ベースの対話などの機能を提供します。TRANSX モニターと共に使用した場合、HTTP モニターと HTTPS モニターは、リンクに組み込まれたセッション ID、地域コード、または日時などの動的コンテンツを含む Web ページをテストできます。動的コンテンツの値は、トランザクションがテストされるごとに異なる場合があります。

トランザクション・モードで稼働する場合に HTTP モニターおよび HTTPS モニターが提供する動的コンテンツ機能を使用すると、URL に組み込まれるか HTML `form` エレメントで定義された動的ページ・エレメントと呼ばれる名前と値のペアの形で、動的コンテンツを識別できます。次に、モニターが各テストの実行時にページから動的ページ・エレメントを抽出し、トランザクションがテストされるたびに適切な動的な値が使用されるようにします。

例えば、Web サイトにログオンするためのリンクを含む、Web サイトのログイン・ページ `http://www.mycompany.com/login` について考えます。ログイン・アクション用のリンク URL `http://`

www.mycompany.com/doLogin?sessionID=id には、ログイン・トランザクションのセッション ID が含まれています。この例では、名前と値のペア sessionID=id が、動的ページ・エレメントです。id の値は、ログイン・ページにアクセスされるたびに変わります。トランザクションの一部としてログイン・ページをテストするには、トランザクションがテストされるたびに sessionID の値を取得して使用し、その値をログイン・アクション URL に挿入するようにトランザクションを構成します。

TRANSX モニターは、動的ページ・エレメントをトランザクション・ステップから次のトランザクション・ステップへ渡します。 Web サイトのログインの例では、トランザクションの最初のステップは、ログイン・ページにアクセスしてセッション ID を取得し、それを 2 番目のステップに渡すというものです。2 番目のステップで、その ID を含むログイン要求が実行依頼されます。これらのステップで実行される操作は以下のとおりです。

1. 動的ページ・エレメントを含むページにアクセスします。例えば http://www.mycompany.com/login などです。
2. 動的ページ・エレメントを使用するアクションを実行依頼します。例えば http://www.mycompany.com/doLogin?sessionID=@@030671 などです。

トランザクション・ステップで動的ページ・エレメントを識別すると、それが次のトランザクション・ステップに渡され、そこで名前と値のペアが要求に挿入されます。エレメントは、明示的に削除されるまで、後続の各 HTTP または HTTPS トランザクションへ順に送られます。

動的ページの追加および削除

各 HTTP または HTTPS トランザクション・ステップは要求で構成されており、HTML ページを返します。動的ページ・エレメントを含むトランザクション・ステップの実行時に、モニターは HTML ページを構文解析して、各エレメントの名前と値のペアを見つけて、次のトランザクション・ステップに渡します。そこで、渡されたペアが HTTP 要求または HTTPS 要求に挿入されます。

ステップが動的ページ・エレメントを使用するように指定するには、エレメントのパラメーター・タイプを DYNAMIC に設定します。次に、抽出して後のステップに渡す各動的エレメントを指定します。sessionID などの名前を入力して、各動的エレメントを識別し、値として「追加先」を選択します。「追加先」は、エレメントが後のステップに渡されることを示します。

注 : 動的ページ・エレメントの名前を取得するには、その動的ページ・エレメントが存在するページの HTML ソースを表示します。

動的ページ・エレメントは、トランザクション・ステップから次のトランザクション・ステップへ渡されます。 ページ・エレメントを次のステップに渡す必要がなくなった場合は、エレメントの「値」を「除去先」に設定します。 手動で後続のトランザクション・ステップを更新し、正しいページ・エレメントが処理されるようにします。

動的ページ・エレメントの追加および削除には、以下のガイドラインを使用します。

- ステップが動的ページ・エレメントを使用しない場合は、パラメーター・タイプとして DYNAMIC を選択しないでください。
- ステップで動的ページ・エレメントを必要とする場合、動的エレメントが出現するページを取得するステップで、そのエレメントの名前と「追加」という値を指定する必要があります。
- ステップで、前のステップから渡される動的エレメントを必要としない場合は、直前のステップの値を「除去」に設定します。

GET および POST

GET メソッドでは、すべての動的ページ・エレメントが自動的に要求 URL に挿入されます。POST メソッドでは、「パラメーター」タブで、各動的エレメントを FORM パラメーターとして指定する必要があります。

モニターは、トランザクション・ステップを実行するときに、自動的に各フォームの動的な値を挿入します。

トランザクションの作成

Internet Service Monitoring ユーザー・インターフェースを 使用して TRANSX プロファイル・エレメントおよびトランザクション・ステップを作成することにより、トランザクションを定義します。詳しくは、441 ページの『トランザクションの作成』 を参照してください。

TANSX モニター・サービス・テストの構成

表 *157. TRANSX* モニター構成

フィールド	説明
transxname	トランザクションの名前。
説明	エレメントに関する記述情報を提供するテキスト・フィールド。
ポーリング	ポーリングの間隔 (秒単位)。 デフォルト: 300

注 : Web のブラウズ、ファイルのダウンロード、および電子メール・メッセージの送信から成るシーケンスを使用して、Web サイトの可用性をモニターします。

1. TRANSX プロファイル・エレメントを作成します。
2. Web サイトの可用性をモニターするための HTTP トランザクション・ステップを作成します。
3. ファイルのダウンロードをモニターするための FTP トランザクション・ステップを作成します。
4. 電子メールをモニターするための POP3 または SMTP トランザクション・ステップを作成します。

詳しくは、各モニターの資料を参照してください。

モニター・エレメント

TRANSX モニターは、各トランザクションの結果が含まれるイベントを生成します。これらのイベントには、トランザクション全体の結果および個々のトランザクション・ステップの結果が含まれます。

デフォルトで、モニターはすべてのトランザクションおよびステップの結果を単一イベントの中に入れます。ただし、`MultipleEvents` プロパティーを使用すると、各トランザクション・ステップ用の個別のイベントおよびトランザクション全体用の要約イベントを作成するようにモニターを構成することができます。表 1 に、TRANSX 要約エレメントをリストします。

アスタリスク (*) で示すエレメントは、属性として使用可能です。属性の名前は、括弧で囲んで記載しています。アスタリスクが付いていないものには、同等の属性はありません。括弧内に属性が示されているが、エレメントがないものは、属性としてのみ使用可能で、同等のエレメントがないことを示しています。

表 *158. TRANSX* 要約モニター・エレメント

エレメント	説明
`$numberOfSteps*(NumberOfSteps)`	トランザクションのステップ数。
`$stepDescriptions`	縦線文字 (|) によって区切られた各ステップの説明のリスト。
`$stepTimes*(Step1 to 10TotalTime)`	各ステップ (1 から 10) によってデータが返されたタイミング。
`$stepUnits`	縦線文字 (|) で区切られた各ステップの単位 (通常は秒単位) のリスト。
`(TransName)`	トランザクションの構成時に指定されたトランザクションの名前。
`(TransStepDescription)`	ステップの構成時に指定されたトランザクション・ステップの説明。

状況メッセージ

TRANSX モニターは、IBM Application Performance Management を使用している場合は ResultMessage 属性で状況メッセージを示します。これらのメッセージは、テストの結果を示します。

以下の表では、状況メッセージについて説明します。

表 *159. TRANSX* モニターの状況メッセージ

メッセージ	説明
Successfully completed transaction	トランザクションは正常に完了しました。
Error in transaction	トランザクションのステップの 1 つで障害が発生しました。
Service Level Failed, ending transactionService Level Failed	ステップの 1 つのサービス・レベルが失敗の応答を返したため、トランザクションは停止しました。

トランザクションの作成

Internet Service Monitoring エージェント ユーザー・インターフェースを 使用して、TRANSX プロファイル・エレメントおよびトランザクション・ステップを作成することにより、トランザクションを定義できます。

手順

インターフェースを 使用してトランザクションを作成するには、以下のようにします。

1. 「**システム構成 (System Configuration)**」アイコンをクリックします。その下の「**エージェント構成**」をクリックします。

   「エージェント構成」ウィンドウが開きます。
2. 「**ISM**」タブをクリックして、Internet Service Monitoring エージェントを構成します。
3. **(+)** プラス・アイコンをクリックして、新規プロファイルを作成します。「**プロファイル名**」と「**説明**」を入力します。
4. 「**次へ**」をクリックします。
5. モニター・ドロップダウン・リストから「**TRANSX**」モニターをクリックして、TRANSX モニターを選択します。
6. 「**次へ**」をクリックします。
7. 必須パラメーターを入力してください。
8. 「**拡張**」タブで、「**ポーリング間隔**」を指定します。
9. 「ステップ」タブで **(+)** プラス・アイコンをクリックします。
10. モニター・ドロップダウン・リストから、選択する必要があるモニターをクリックします。
11. 「**選択**」をクリックして、トランザクション・ステップを構成します。
    a) プロファイル・エレメントを構成するために以前に入力したのと同様に、必須パラメーターとオプション・パラメーターを指定します。
    b) HTTP または HTTPS モニター用の動的ステップを作成する場合は、「パラメーター」タブで「名前」と「値」のペアを設定し、パラメーター・タイプとして「DYNAMIC」を選択します。
12. 「**追加**」をクリックします。
13. ステップ・グリッドで「リフレッシュ」アイコンをクリックします。
14. 追加のトランザクション・ヘルプごとに、ステップ 1 からステップ 13 までを繰り返します。
15. 「**追加**」をクリックして終了します。
16. 「**完了**」をクリックして保存します。

タスクの結果

注 : トランザクション・ステップごとに一時停止するように指定するには、StepPause プロパティーを使用します。

Internet Service Monitoring での SSL 設定

Internet Service Monitoring は、さまざまなモニターを使用する (通常はリモートの) インターネット・サービスと安全に通信するために OpenSSL を使用します。例えば、HTTPS モニターは保護された HTTPD と通信します。Internet Service Monitoring エージェントはまた、モニターと Databridge の間、および Internet Service Monitoring エージェント (KIS) と Databridge の間でも OpenSSL を使用します。ご使用のアプリケーションで使用する暗号スイートを SSLCipherSuite プロパティーに指定してください。

Databridge は、各モニターおよび Internet Service Monitoring エージェント と安全に通信するように構成して、すべてのモニターが、Databridge とのセキュアな通信を管理するための Databridge 関連のプロパティーの共通セットを共有するようにする必要があります。また、一部のモニターでは、テスト中のそれぞれのインターネット・サービスとのセキュア通信を管理するための類似の (ただし異なる) セットの関連プロパティーも共有します。

以下のモニターは、保護されたインターネット・サービスのモニターをサポートしています。

- HTTPS
- IMAP4
- POP3
- SMTP

これらのモニターは証明書を使用します。すべての証明書は、$ISMHOME/certificates にある Privacy Enhanced Mail .Pemfiles に X509 形式で保管されます。Databridge の証明書も同じ場所に保管されます。このため、以下のプロパティーは、すべてのモニター、Databridge、および Internet Service Monitoring エージェント によって共有されます。

- SSLTrustStore (デフォルト: $ISMHOME/certificates/trust.pem)
- SSLTrustStorePath (デフォルト: $ISMHOME/certificates/)

モニターと Databridge の間のすべての通信、および選択されたモニターと保護されたインターネット・サービスの間のすべての通信は、同じバージョンの OpenSSL に基づいて構築されるため、特性を共有します。例えば、Internet Service Monitoring が提供できる最高レベルのセキュリティーは、基礎となる OpenSSL によって提供される最高レベルの機能です。提供される最低レベルのセキュリティーも同様に、基礎となる OpenSSL に同様に依存します。

Internet Service Monitoring エージェント が更新され、その更新に基礎となる OpenSSL の更新が含まれている場合、モニター対象のインターネット・サービスが影響を受ける可能性があります。例:

1. Internet Service Monitoring エージェント V7.x.1 の HTTPS モニターは、保護された HTTPD サーバーをモニターしています。
2. 更新バージョンの OpenSSL を含む新規バージョンの Internet Service Monitoring エージェント を適用します。このことは、HTTPS モニターが V7.x.2 になったことを意味します。
3. HTTPS モニターが保護された HTTPD をモニターできない状態になっていることに気付きます。

HTTPD サーバーのセキュリティー・レベルが、新たに更新された Internet Service Monitoring エージェント V7.x.2 でサポートされる最小レベルを下回っています。HTTPS モニターの構成は変更されていませんが、その動作は変更されています。これは、動作が基礎となる OpenSSL レイヤーに依存しているためです。新しい Internet Service Monitoring エージェント/HTTPS モニター/OpenSSL の組み合わせは、以前の組み合わせよりもセキュアであるため、ここでリモート HTTPD サーバーのセキュリティー・レベルを上げる必要があります。

保護されたインターネット・サービスのモニターには、ジレンマがあります。Internet Service Monitoring エージェント のセキュリティー・レベルを、弱く保護されたインターネット・サービスをモニターできる程度に低くすべきか、または、現在推奨されている最小の設定と同じレベルにすべきか検討します。前者を選択した場合、弱いセキュリティーの Internet Service Monitoring エージェント により、おそらく両端でセキュリティーが損なわれる可能性があります。

すべてのモニターで、同じバージョンの OpenSSL が使用されます。これらのモニターはすべて、基礎となる OpenSSL を構成するためのモニター・プロパティーの共通セットを共有します。これらのプロパティーについて、次の表で説明します。

表 *160. OpenSSL* 関連モニター・プロパティー

プロパティー名	**プロパティー・パラメーター**	**説明**
SSLCipherSuite	ストリング	モニターとモニター対象のインターネット・サービスの間の SSL 操作に使用する暗号スイートを指定します。このプロパティーの値は、OpenSSL で推奨される形式にする必要があります。 デフォルト: AES:3DES:DES:!EXP:!DHE:!EDH
SSLDisableSSLv2	0|1	保護されたインターネット・サービスをモニターする際に行うセキュア接続のタイプを決定します。 0 – SSLv2 は許可されます 1 – SSLv2 は許可されません デフォルト: 1 (SSLv2 は許可されません)。
SSLDisableSSLv3	0|1	保護されたインターネット・サービスをモニターする際に行うセキュア接続のタイプを決定します。 0 – SSLv3 は許可されます 1 – SSLv3 は許可されません デフォルト: 1 (SSLv3 は許可されません)。
SSLDisableTLS	0|1	保護されたインターネット・サービスをモニターする際に行うセキュア接続のタイプを決定します。 0 – TLSv1.0 は許可されます 1 – TLSv1.0 は許可されません デフォルト: 0 (TLSv1.0 は許可されます)。
SSLDisableTLS11	0|1	保護されたインターネット・サービスをモニターする際に行うセキュア接続のタイプを決定します。 0 – TLSv1.1 は許可されます 1 – TLSv1.1 は許可されません デフォルト: 0 (TLSv1.1 は許可されます)。
SSLDisableTLS12	0|1	保護されたインターネット・サービスをモニターする際に行うセキュア接続のタイプを決定します。 0 – TLSv1.2 は許可されます 1 – TLSv1.2 は許可されません デフォルト: 0 (TLSv1.2 は許可されます)。

表 *160. OpenSSL* 関連モニター・プロパティー *(続き)*

プロパティー名	プロパティー・パラメーター	説明
SSLCertificateFile	ストリング	モニターで使用される公開デジタル証明書ファイルのパスとファイル名。モニターがインターネット・サービスへのセキュア接続をセットアップしようとしたときに、インターネット・サービスはオプションで、モニターによるクライアント・サイド証明書の提供を要求し、当該インターネット・サービスがモニターまたはクライアントを検証できるようにすることができます (クライアント・サイド証明書の検証)。 証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。 HTTPS モニターでは、この値は作成時に HTTPS エレメントごとに指定できます。ただし、HTTPS モニターがすべてのエレメントに同じ証明書を使用する場合は、HTTPS.props ファイルの値が使用されます。 IMAP、LDAP、POP3、SIP、SMTP、および SOAP モニターの場合は、値はモニター全体に設定されます。 パスが絶対パスでない場合、モニターは、そのパスを作業ディレクトリー ($ISMHOME/certificates) からの相対位置にあるものと解釈します。 デフォルト: “”
SSLKeyFile	ストリング	モニターで使用される秘密鍵が入っているファイルのパスとファイル名。モニターはこのファイルを使用して、他のユーザーに送信するメッセージを暗号化します。受信側は、モニターの公開デジタル証明書を使用してメッセージを復号します。 デフォルト: monitoryKey.pem
SSLKeyPassword	ストリング	SSL 秘密鍵の暗号化に使用されるパスワード。 デフォルト: “”
SSLTrustStoreFile	ストリング	モニターされているインターネット・サービスのすべての X509 公開証明書を連結リストとして保管するファイルの完全修飾名。 失効証明書 (CRL) も連結リストとしてここに保管されます。 Databridge では、その公開証明書もここに保管できます。このプロパティーは、bridge.props ファイルに表示されます。 証明書は、Privacy Enhanced Mail (PEM) 形式で格納されます。他の形式で取得した証明書は、http://www.openssl.org から入手できる OpenSSL ソフトウェアを使用して PEM 形式に変換します。 デフォルト: **“$ISMHOME/certificates/trust.pem”**

表 160. *OpenSSL* 関連モニター・プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
SSLTrustStorePath	ストリング	モニター対象のセキュア・インターネット・サービスの X509 証明書を含む .pem ファイルの場所。 失効証明書 (CRL) もここに保管されます。 Databridge では、その公開証明書もここに保管できます。このプロパティーは、bridge.props ファイルに表示されます。 新しい証明書がこのディレクトリーに追加された場合は、openssl rehash コマンドを実行してディレクトリーをスキャンし、各証明書のハッシュを計算します。 SSLTrustStoreFile プロパティーと SSLTrustStorePath プロパティーの両方を使用する場合、OpenSSL は両方のプロパティーを使用してトラステッド証明書を見つけます。 デフォルト: "$ISMHOME/certificates/"
VerifyCertificate Preference	0|1	証明書失効リスト (CRL) に対してモニターされているインターネット・サービスによって提供される証明書の検証を有効または無効にします。 デフォルト: 0 - 無効

暗号スイート

Internet Service Monitoring で使用可能な暗号スイートは、OpenSSL で許可される暗号スイートのサブセットです。OpenSSL によって許可される暗号スイートのセットは、時間とともに変化します。新しい脆弱性が発見され、ベスト・プラクティスが進化するにつれて、特定または一般的なタイプの暗号スイートへのアクセスが、OpenSSL によって制限または完全に削除される場合があります。新しいバージョンの ISM にはこれらの新しいバージョンの OpenSSL が含まれているため、モニターの構成と操作に影響を与える可能性があるフローオン効果 (副次的効果) があります。

モニター全体用の SSLCipherSuite プロパティーを使用して、モニターによって許可される暗号スイートを、キーワードを使用して利用できるすべての暗号スイートから選択します。複数のスイートを指定するには、コロンで区切られたキーワードのリストを使用します。例えば、デフォルトの SSLCipherSuite プロパティーは AES:3DES:DES:!EXP:!DHE:!EDH です。この選択では、AES、3DES、および DES を含む暗号スイートは許可されますが、EXP (エクスポート (短いキー長))、DHE (Diffie Hellman Exchange)、または EDH (一時 Diffie Hellman) 鍵交換を使用する暗号スイートはすべて除外されます。また、モニターとインターネット・サービスの間でセキュア接続が行われるときに、最初に AES が使用され、続いて 3DES が使用され、その後、必要に応じて DES が使用されます。Internet Service Monitoring エージェント の暗号スイート・リストの構文は、OpenSSL の場合と同じです。

モニター用の正しい暗号スイート・セットを選択するには、基礎となる OpenSSL がサポートしている内容、モニター対象のインターネット・サービスがサポートする暗号の範囲、および組織のセキュリティー標準を考慮してください。Internet Service Monitoring または OpenSSL で許容されるレベルよりも低いレベルのセキュリティーを持つセキュア外部サイトをモニターできない場合があります。場合によっては、あるインターネット・サービスについて、以前はモニターで監視できたが、Internet Service Monitoring のアップグレード後に監視できなくなることがあります。これはセキュリティー・レベルに互換性がないためです。

以下の表に、AES:3DES:DES:!EXP:!DHE:!EDH の SSLCiperSuite のデフォルト値に相当する暗号スイートのサブセットとそのプロパティーをリストします。この表では、以下の用語が使用されています。

- 暗号スイート名: キーワードから構成される名前を使用して、暗号スイートを示します。

- プロトコル: サポートされるプロトコルのバージョンを示します。
- 鍵交換: 暗号化および復号に使用される鍵交換システムを示します。
- 暗号化および鍵の長さ: 使用される暗号化アルゴリズムのタイプと、使用される鍵の長さ (ビット単位) を示します。
- MAC: データが改ざんされていないことを確認するために使用されるメッセージ認証コードを示します。

表 *161. Cipher suite name and property values AES:3DES:DES:!EXP:!DHE:!EDH*

暗号スイート名	プロトコル	鍵交換	認証	暗号化および鍵の長さ	メッセージ認証コード
ECDHE-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH	RSA	AESGCM(256)	AEAD
ECDHE-ECDSA-AES256-GCM-SHA384	TLSv1.2	ECDH	ECDSA	AESGCM(256)	AEAD
ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
ECDHE-ECDSA-AES256-SHA384	TLSv1.2	ECDH	ECDSA	AES(256)	SHA384
ECDHE-RSA-AES256-SHA	SSLv3	ECDH	RSA	AES(256)	SHA1
ECDHE-ECDSA-AES256-SHA	SSLv3	ECDH	ECDSA	AES(256)	SHA1
SRP-DSS-AES-256-CBC-SHA	SSLv3	SRP	DSS	AES(256)	SHA1
SRP-RSA-AES-256-CBC-SHA	SSLv3	SRP	RSA	AES(256)	SHA1
SRP-AES-256-CBC-SHA	SSLv3	SRP	SRP	AES(256)	SHA1
DH-DSS-AES256-GCM-SHA384	TLSv1.2	DH/DSS	DH	AESGCM(256)	AEAD
... さらに 61 個の行が続きます					

以下の表に、AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3 の SSLCiperSuite の値に相当する暗号スイートのサブセットとそのプロパティーをリストします。現在、一部のプロトコルが除去されて、暗号スイートのセット全体が 71 から 31 に削減されました。

表 *162. Cipher suite name and property values **AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3***

暗号スイート名	プロトコル	鍵交換	認証	暗号化および鍵の長さ	メッセージ認証コード
ECDHE-ECDSA-AES256-GCM-SHA384	TLSv1.2	ECDH	ECDSA	AESGCM(256)	AEAD
ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
ECDHE-ECDSA-AES256-SHA384	TLSv1.2	ECDH	ECDSA	AES(256)	SHA384
DH-DSS-AES256-GCM-SHA384	TLSv1.2	DH/DSS	DH	AESGCM(256)	AEAD

表 *162. Cipher suite name and property values* ***AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3*** *(続き)*					
暗号スイート名	**プロトコル**	**鍵交換**	**認証**	**暗号化および鍵の長さ**	**メッセージ認証コード**
DH-RSA-AES256-GCM-SHA384	TLSv1.2	DH/RSA	DH	AESGCM(256)	AEAD
DH-RSA-AES256-SHA256	TLSv1.2	DH/RSA	DH	AES(256)	SHA256
DH-DSS-AES256-SHA256	TLSv1.2	DH/DSS	DH	AES(256)	SHA256
ADH-AES256-GCM-SHA384	TLSv1.2	DH	なし	AESGCM(256)	AEAD
ADH-AES256-SHA256	TLSv1.2	DH	なし	AES(256)	SHA256
ECDH-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH/RSA	ECDH	AESGCM(256)	AEAD
... さらに 21 個の行が続きます					

脆弱性の削減

将来のリリースでは、DHE 暗号と EDH 暗号は、脆弱性が存在するためにデフォルトで無効になります。以前のバージョンの Internet Service Monitoring エージェント では、すべてのモニターで DHE 暗号と EDH 暗号の無効化が必要になる場合があります。DHE 暗号と EDH 暗号を無効にするには、`SSLCipherSuite` と `BridgeSSLCipherSet` のモニター・プロパティーを更新します。

例えば、HTTPS モニターで DHE 暗号と EDH 暗号を無効にするには、以下のプロパティーを組み込むように `https.props` ファイルを更新します。

```
SSLCipherSuite: AES:3DES:DES:!DES-CBC-SHA:!EXP:!DHE:!EDH
BridgeSSLCipherSet: AES:3DES:DES:!DES-CBC-SHA:!EXP:!DHE:!ED
```

この構成変更により互換性の問題が発生しないことを必ず確認してください。この修正の適用後にデフォルトの設定を変更すると、セキュリティー上の脆弱性にさらされる可能性があります。環境全体を確認して、TLS で使用される Diffie-Hellman 鍵交換プロトコルを有効にした他の領域を特定し、適切な緩和処置および修復処置を行う必要があります。

プロトコルの選択

過去および現在の広範囲なセキュア通信プロトコルから選択できます。これらのプロトコルは、以下のブール・モニター・プロパティーのセットを使用して個別に選択できます。

- SSLDisableSSLv2
- SSLDisableSSLv3
- SSLDisableTLS
- SSLDisableTLS11
- SSLDisableTLS12
- BridgeSSLDisableSSLv2
- BridgeSSLDisableSSLv3

SSLv2 と SSLv3 は無効にする必要があります。これらのプロトコルは不正に使用され、いくつかの既知の脆弱性があります。これらはデフォルトでは無効であり、既存システム用にのみ提供されています。

Internet Service Monitoring では、TLS はデフォルトで有効化されます。モニター中のインターネット・サービスが TLS 1.0 を使用しておらず、既に TLS 1.1 または TLS 1.2 に移行していることが分かっている場合は、Internet Service Monitoring の未使用のプロトコルを無効にする必要があります。

Databridge コンポーネントは、Internet Service Monitoring エージェントおよび各モニターと通信します。デフォルトでは、この通信は暗号化されており、TLS が優先プロトコルです。

鍵トラストストアおよび証明書

Internet Service Monitoring は、ユーザー定義のロケーションにあるユーザー定義ファイルに証明書を保管します。すべての証明書は、Privacy Enhanced Mail (PEM) 形式でなければなりません。他の組織から取得した公開証明書が PEM 形式に変換されていることを確認してください。変換ソフトウェアは、http://www.openssl.org で入手できます。

SSLTrustStoreFile プロパティーを使用して指定されたトラステッド証明書は、連結リストとしてファイルに保管されます。

証明書の検証に使用できるトラストストアに、証明書失効リスト (CRL) を保管することをお勧めします。認証局には、失効した証明書のリストを生成するシステム、およびそれらのリストを公開して使用可能にする配布システムが用意されています。その後、証明書が危殆化されると、その証明書は失効します。

Databridge のセキュリティー設定

すべてのモニターは Databridge と通信するため、すべてのモニターには、モニターと Databridge 間の通信を管理するために設定する必要がある共通のプロパティー・セットがあります。デフォルトでは、通信は暗号化されています。デフォルトの暗号化プロトコルは TLS です。モニター・プロパティーとは異なり、特定のバージョンの TLS を有効または無効にするかどうかを制御するメカニズムはありません。すべてのモニターで Databridge プロパティーの値が同じでなければなりません。そうでない場合、通信の問題が発生します。同様に、Databridge .props ファイルに設定されているプロパティーは、モニターのプロパティーと整合している必要があります。Databridge は、独自の .props ファイルを持つ Internet Service Monitoring エージェントとも通信します。エージェント .props のいくつかの値は、Databridge に関連しており、モニターと同様に、Databridge .props ファイルのプロパティーの値と整合している必要があります。

表 *163. OpenSSL* 関連の *Databridge* プロパティー

プロパティー名	プロパティー・パラメーター	説明
BridgeSSLEncryption	0|1	Databridge との通信が暗号化されるかどうかを決定します。これは、Databridge から Monitors および Internet Service Monitoring エージェントへのすべての通信を対象としています。 0 – 非暗号化 1 – 暗号化 **制約事項 :** Internet Service Monitoring エージェント、すべてのモニター、および Databridge で同じ値を設定してください。
BridgeSSLCipherSet	ストリング	Databridge との間の SSL 操作に使用する暗号スイートを指定します。このプロパティーの値は、OpenSSL で推奨される形式にする必要があります。 **制約事項 :** Internet Service Monitoring エージェント、エージェント、すべてのモニター、および Databridge で同じ値を設定してください。 デフォルト: AES:3DES:DES:!EXP:!DHE:!EDH

表 163. *OpenSSL* 関連の *Databridge* プロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
BridgeSSLDisableSSLv2	0|1	Databridge との間で行うセキュア接続のタイプを決定します。 0 – SSLv2 と SSLv3 が許可されます 1 – SSLv2 は許可されません **制約事項 :** Internet Service Monitoring エージェント、すべてのモニター、および Databridge で同じ値を設定してください。 デフォルト: 1 (SSLv2 は許可されません)。
BridgeSSLDisableSSLv3	0|1	Databridge との間で行うセキュア接続のタイプを決定します。 0 – SSLv3 は許可されます 1 – SSLv3 は許可されません **制約事項 :** Internet Service Monitoring エージェント、すべてのモニター、および Databridge で同じ値を設定してください。 デフォルト: 1 (SSLv3 は許可されません)。
BridgeSSLCertificateFile	ストリング	デジタル Databridge SSL 証明書のパスおよびファイル名。 デフォルト: `$ISMHOME/certificates/bridgeCert.pem`
BridgeSSLKeyFile	ストリング	Databridge SSL 秘密鍵ファイルのパスおよびファイル名。 デフォルト: `$ISMHOME/certificates/bridgeKey.pem`
BridgeSSLKeyPassword	ストリング	Databridge SSL 秘密鍵の暗号化に使用されるパスワード。 デフォルト: **Tivoli**

表 163. *OpenSSL* 関連の *Databridge* プロパティー (続き)

プロパティー名	プロパティー・パラメーター	説明
BridgeSSLTrustStore	ストリング	認証用のトラステッド証明書ファイルのパスおよびファイル名。これは、**BridgeSSLAuthenticatePeer** プロパティーを使用する場合にのみ必要です。 デフォルト: `$ISMHOME/certificates/trust.pem` モニターと Databridge の間、または Databridge とエージェント間の SSL 認証を構成する場合は、`BridgeSSLAuthenticatePeer` を `1` に設定して、Databridge を再始動します。このアクションにより、サーバーからの証明書が認証されます。証明書は、`SSLTrustStoreFile` と `SSLTrustStorePath` の両方に保管できます。 デフォルト: • `SSLTrustStoreFile`、`$ISMHOME/certificates/trust.pem` • `SSLTrustStorePath`、`$ISMHOME/certificates/` 新規証明書を追加するためには、以下のステップのいずれかを実行します。 • `SSLTrustStoreFile` テキスト・ファイルで証明書をリストの終わりに追加します。 • 新しい証明書を `SSLTrustStorePath` ディレクトリーに追加して、OpenSSL の **c_rehash certificate_dir** コマンドを実行して、証明書をハッシュします。
SSLTrustStoreFile	ストリング	このプロパティーは、セキュア・モニターおよび Databridge によって使用されます。詳細については、443 ページの表 160 を参照してください。
SSLTrustStorePath	ストリング	このプロパティーは、セキュア・モニターおよび Databridge によって使用されます。詳細については、443 ページの表 160 を参照してください。

表 163. *OpenSSL* 関連の *Databridge* プロパティー (続き)		
プロパティー名	**プロパティー・パラメーター**	**説明**
BridgeSSLAuthenticatePeer	0|1	Databridge が他の Internet Service Monitoring コンポーネントと相互認証するかどうかを指定します。 0 - 無効 1 – 有効 モニターが Databridge に接続する場合、モニターは Databridge で認証する必要があり、Databridge はモニターで認証する必要があります。 Internet Service Monitoring エージェントが Databridge に接続する場合、エージェントは Databridge で認証する必要があり、Databridge はエージェントで認証する必要があります。 Databridge の証明書は、BridgeSSLTrustStore に保管されます。 デフォルト: 0 - 無効

Internet Service Monitoring エージェントのプロパティー

Internet Service Monitoring エージェント には、セキュリティーのプロパティーと 設定のセットを含む独自のプロパティー・ファイルがあります。エージェント・プロパティー・ファイルは、モニターと通信しませんが、Databridge との通信を行います。そのため、エージェントの .props ファイルのセキュリティー設定により、エージェントと Databridge の間の通信が管理されます。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows システムで構成することができます。

手順

次のようにして、Internet Service Monitoring エージェント エージェントをユーザー・システムで構成します。

ユーザーのシステムで Internet Service Monitoring エージェントを手動で構成するには、次の手順を実行します。

1. 「**IBM Performance Management ダッシュボード**」で、「**システム構成 (System configuration)**」>「**エージェント構成**」(「**システム構成 (System Configuration)**」の下にリストされている) をクリックします。
2. 「**ISM**」タブをクリックして、Internet Service Monitoring エージェント・ダッシュボードを開きます。

Databridge の構成

Databridge を構成する過程には、その動作 (コンポーネント・モジュールとインターネット・サービス・モニターとの接続など) を制御する Databridge のプロパティーの設定手順が含まれます。

操作および構成

Databridge とそのコンポーネント・モジュールは、プロパティー・ファイルを使用して構成します。

プロパティーは、Internet Service Monitoring エージェント・ダッシュボードでのレポート作成のためにテスト結果を IBM Cloud Application Performance Management に送信する Databridge とそのコンポーネント・モジュールの動作を決定します。

Databridge の構成

インターネット・サービス・モニターからデータを受信したり、そのデータをさらに処理するためにコンポーネント・モジュールに転送するためには、Databridge を構成する必要があります。

次の表に、Databridge に関連付けられたファイルのリストを示します。**プロパティー・ファイル**、**ストア・アンド・フォワード・ファイル**、および**ログ・ファイル**については、該当のセクションでさらに詳しく説明します。

表 *164. Databridge* ファイルとそのロケーション

Databridge ファイル	**ロケーションまたは名前**
実行可能ファイル	`$ISHOME/platform/arch/bin/nco_m_bridge`
プロパティー・ファイル	`$ISHOME/etc/props/bridge.props`
ストア・アンド・フォワード・ファイル	名前およびロケーションは、`bridge.props` ファイル内のプロパティーによって指定されます。デフォルトの名前およびロケーションは、`$ISHOME/var/sm_bridge.saf` です。
ログ・ファイル	`$ISHOME/log/bridge.log`
エラー・ログ・ファイル	`$ISHOME/log/bridge.err`

ストア・アンド・フォワード・ファイル

Databridge は、Netcool/OMNIbus にデータを転送できない場合、通常なら送信するすべてのデータをストア・アンド・フォワード (SAF) ファイルに保管します。Netcool/OMNIbus が再び使用可能になると、SAF ファイルに保管したすべてのイベントを処理します。

Databridge のプロパティー・ファイル内の QFile および QSize プロパティーは、ストア・アンド・フォワード処理の名前、ロケーションおよび動作を決定します。

ログ・ファイル

Databridge は、その動作に関する毎日のメッセージをメッセージ・ログ・ファイルに送ります。デフォルトでは、このファイルの名前は `$ISHOME/log/bridge.log` です。このファイルは、深夜 0 時に更新されます。メッセージ・ロギングの動作は、Databridge のプロパティー MsgDailyLog および MsgTimeLog によって制御されます。

Databridge の開始

Windows Services コンソールを使用して Databridge を開始します。

手順

注 : ObjectServer モジュールが Databridge に接続されている場合は、Databridge を開始する前に、必ずターゲット・システムを稼働状態にしておいてください。Databridge モジュールの中に 1 つでも正しく初期化されていないものがあると、Databridge は開始されません。

1. Windows デスクトップから、「**スタート**」 > 「**管理ツール**」 > 「**サービス**」 をクリックします。
2. サービスのリストから「NCO BRIDGE Internet Service Monitor」というサービスを選択し、メニューの「**開始**」をクリックします。

モジュールの接続

Databridge プロパティー・ファイルは、Databridge に接続するモジュールを定義します。

このタスクについて

Module n SharedLib と Module n PropFile のプロパティーのペアは、それぞれのペアが 1 つのモジュールの接続を定義します。モジュールは、Module0 から始まって、定義と同じ順序でロードされます。

手順

1. 個々のモジュールを Databridge に接続するには、次の手順を実行します。

a) Databridge プロパティー・ファイルで、次に使用可能な Module n SharedLib と Module n PropFile のプロパティー・ペアを識別します。

b) Module n SharedLib は、モジュールの共有ライブラリー (そのバイナリー実装) の名前に設定します。

c) Module n PropFile は、モジュールのプロパティー・ファイルの絶対パスに設定します。

この例では、回線 1 および 2 が ObjectServer モジュールに接続し、回線 3 および 4 が Datalog モジュールに接続し、回線 5 および 6 が IBM Application Performance Management(pipe) モジュールに接続しています。Datalog モジュールにはプロパティーはありません。したがって、プロパティー・ファイル用の項目の値は "" です。

2. モジュールを使用不可にするには、以下のようにします。

a) 該当の Module n SharedLib プロパティーを "NONE" に設定し、Module n PropFile プロパティーを "" に設定します。n より大きい値を持つ他のすべてのモジュールも無視されます。

モニターの接続
インターネット・サービス・モニターは、TCP を介して Databridge に接続されます。各モニターには、Databridge への接続を構成する一連のプロパティーがあります。

このタスクについて
Databridge にモニターを接続するには、モニターのプロパティー・ファイルに定義されている BridgePort プロパティーの値を、Databridge プロパティー・ファイルに定義されている SocketPort プロパティーの値に設定します。各モニターの BridgePort プロパティーおよび Databridge の SocketPort プロパティーのデフォルト値は 9510 です。

Databridge は、モニターから受け取ったテスト結果の SSL 暗号化をサポートしています。モニターのテスト結果を暗号化するには、モニターのプロパティー・ファイルに定義されている BridgeSSL プロパティーの値を、Databridge プロパティー・ファイルに定義されている BridgeSSL プロパティーの値に設定します。すべてのモニターのテスト結果を暗号化するには、すべてのモニターが同一の BridgeSSL プロパティーを持っている必要があります。

***Databridge* モジュールの構成**
Databridge は、テスト結果を Internet Service Monitoring エージェント宛に送ります。モニター・エージェントは、このデータを必要なフォーマットに変換して、それを IBM Application Performance Management サーバーに配布します。Databridge モジュールおよび Internet Service Monitoring エージェントは、それぞれのプロパティー・ファイルを使用して構成します。

Databridge の動作を構成するには、モジュールのプロパティー・ファイルに定義されているプロパティー値を変更します。

モジュールのプロパティー・ファイルの名前は pipe_module.props です。このファイルは、$ISHOME/etc/props/ ディレクトリーの中にあります。

以下の表に、モジュールに使用可能なプロパティーのリストを示します。プロパティーを変更した場合、変更を有効にするためには Databridge を再始動する必要があります。

表 *165. Databridge* モジュールのプロパティー

プロパティー名	タイプ	説明
TEMAHOST	ストリング	モニター・エージェントを実行しているホストの名前。 デフォルト: localhost
TEMAPORT	整数	ホストで使用されるポート番号。 デフォルト: 9520

Internet Service Monitoring エージェントの動作を構成するには、モニター・エージェント・プロパティー・ファイルに定義されているプロパティー値を変更します。

モニター・エージェントのプロパティー・ファイルの名前は kisagent.props です。このファイルは、$ISMHOME/etc/props/ ディレクトリーの中にあります。

以下の表に、モニター・エージェントに使用可能なプロパティーのリストを示します。

表 *166.* モニター・エージェントのプロパティー

プロパティー名	タイプ	説明
TEMAPORT	整数	ホストで使用されるポート番号。これは、モジュール・プロパティー・ファイルにリストされている TEMAPORT プロパティーのポート番号と同一でなければなりません。 デフォルト: 9520
ObsoleteDuration	整数	更新されなかったデータをモニター・エージェントのメモリーから削除するまでの時間 (秒数)。データが更新されない可能性があるのは、例えば、プロファイル・エレメントが停止したり、ネットワーク障害が発生したりした場合です。 **注:** ObsoleteDuration 時間をポーリング間隔より小さい値に設定しないでください。小さい値にすると、ポーリング間隔の間でデータが消失することになるからです。 デフォルト: 900
AggDuration	整数	モニター・エージェントがエージェント・ダッシュボードでデータの集約とレポート作成を停止するまでの時間 (秒数)。指定された時間より古いデータは、モニター・エージェントのメモリーから削除されます。 古いデータは、開始時刻と現在時刻との間隔を集約所要時間と比較して計算されます。この間隔が集約所要時間より大きければ、古いデータの 10% が削除され、開始時刻が間隔の 1/10 だけ増加されます。モニター・エージェントは、この計算を 5 分ごとに実行します。 デフォルト: 3600
ManageServices	0|1	Internet Service Monitoring エージェントの開始/停止に合わせて、すべてのモニターおよび Databridge を開始/停止します。1 は使用可能、0 は使用不可です。 デフォルト: 1

Internet Service Monitoring をインストールすると、Internet Service Monitoring エージェントと Databridge モジュールの間に接続が作成されます。

Netcool/OMNIbus の有効化

Tivoli Netcool/OMNIbus が Internet Service Monitoring エージェントから Netcool/OMNIbus にイベントを送信できるようにするには、以下の手順を実行します。

始める前に

IBM Tivoli Netcool/OMNIbus がインストールされていることを確認してください。

手順

以下の手順を実行して、Netcool/OMNIbus を有効にします。

1. 次のコマンドを使用して、Internet Service Monitoring エージェントを停止します。

```
$CANDLEHOME/bin/ism-agent.sh stop
```

2. $ISMHOME/etc/props パスにある bridge.props ファイルを開き、そのファイルを以下のコード・スニペットで更新します。

```
Module0SharedLib : "libSMModulePipe"
Module0PropFile : "$ISMHOME/etc/props/pipe_module.props"
Module1SharedLib : "libSMModuleObjectServer"
Module1PropFile : "$ISMHOME/etc/props/objectserver.props"
```

3. $ISMHOME/objectserver パスにある 8.1.0 ディレクトリーの許可を次のように変更します。

```
cd  $ISMHOME/objectserver/
chmod -R 777 8.1.0
```

注: chmod -R 777 <file-name> コマンドを使用して、8.1.0 ディレクトリー内のすべてのファイルの許可を変更します。ここで、<file-name> は、8.1.0 ディレクトリー内のファイルの名前です。

4. $ISMHOME/objectserver/8.1.0/etc パスにある omni.dat ファイルを変更して、Netcool/OMNIbus サーバー・アドレスを構成します。
5. 次の場所から nco_igen を実行します。

```
cd $ISMHOME/objectserver/8.1.0/bin
./nco_igen
```

6. 次のコマンドを使用して、Internet Service Monitoring エージェントを開始します。

```
$CANDLEHOME/bin/ism-agent.sh start
```

7. Internet Service Monitoring エージェント、Databridge およびすべてのモニターが実行状態にあることを確認します。

   Databridge とモニターの状況を確認するには、次のコマンドを実行します。

```
ps -aef|grep -i nco_*
```

   Internet Service Monitoring エージェントの状況を確認するには、次のコマンドを実行します。

```
ps -aef|grep -i kis
```

8. IBM Tivoli Netcool/OMNIbus ユーザー・インターフェースを 使用して、Databridge がデータを Netcool/OMNIbus サーバーに送信することを確認します。

   Internet Service Monitoring エージェントのデータが IBM Application Performance Management ユーザー・インターフェースに 表示されなければなりません。

J2SE のモニターの構成

モニター対象のオンプレミス Java アプリケーションからリソースのモニターおよび診断データを収集するには、J2SE データ・コレクターを構成する必要があります。

始める前に

以下のいずれかのサポート対象 Java ランタイムをインストールします。

- Oracle Java Platform Standard Edition 7 (Java SE Development Kit 7)

  要確認: この Java ランタイムは、HTTPS プロトコルを使用して構成された J2SE データ・コレクター・イメージをサポートしません。
- Oracle Java Platform Standard Edition 7 (Java SE Runtime Environment 7)

要確認 : この Java ランタイムは、HTTPS プロトコルを使用して構成された J2SE データ・コレクター・イメージをサポートしません。

- Oracle Java Platform Standard Edition 8 (Java SE Development Kit 8)
- Oracle Java Platform Standard Edition 8 (Java SE Runtime Environment 8)
- IBM SDK, Java Technology Edition, Version 7
- IBM SDK, Java Technology Edition, Version 8

重要 : Windows Server 2016 の場合は、JDK 8 Update 131 (Java SE Development Kit 8u131) か、Java SE Development Kit 7 と Update 80 (JDK 7u80) をインストールしてください。

システム要件について詳しくは、Software Product Compatibility Reports for J2SE data collector を参照してください。

このタスクについて

J2SE データ・コレクターは、Windows、Linux、および AIX の各システム上で構成できます。

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントとデータ・コレクターのバージョン・リストと、各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

手順

1. 以下のファイルを APM インストーラーからディレクトリーにコピーします。

   重要 : ディレクトリー・パスにスペースを含めないでください。

   - Windows gdc.zip ファイルを APM インストーラーからディレクトリーにコピーし、解凍します。
   - Linux AIX gdc-apd.tar.gz ファイルを APM インストーラーからディレクトリーにコピーし、解凍します。
   - Linux AIX j2se_dc フォルダーに対する読み取り/書き込みおよび実行許可をユーザーに付与します。実行許可は、フォルダー内のスクリプトおよび JAR ファイルを実行するために付与します。読み取り/書き込み許可を付与するのは、詳細診断ファイルがこのフォルダー内に生成されるためです。
2. コマンド行で、*DCHOME*¥.gdc¥<toolkit_version>¥bin に移動します。

   ここで、*toolkit_verion* は以下のとおりです。

   - V8.1.4.0 以前の場合、*toolkit_verion* は 7.3.0.5.0 です。
   - V8.1.4.0.1 以降の場合、*toolkit_verion* は 7.3.0.14.0 です。
3. 次のコマンドを実行します。

   Windows `config.bat`

   Linux AIX `config.sh`
4. プロンプトが出されたら、Java ホームのパスを指定し、**Enter** キーを押します。
   例えば、次の場合があります。

   Windows `C:¥Program Files¥jre7`

   Linux AIX `/opt/ibm/java`
5. ご使用のエージェント・バージョンで以下のステップを実行します。

   - V8.1.4.0.2 以前の場合、以下のステップを実行します。

     a. プロンプトが出されたら、アプリケーションのメイン・クラスのフルネーム (修飾名) を入力して、**Enter** キーを押します。メイン・クラスは、モニターする必要があるアプリケーションのエントリー・ポイントです。例: `testapp.TemperatureConveter`

b. プロンプトが出されたら、アプリケーション固有の別名を入力し、**Enter** キーを押します。ここで入力した名前は、APM ダッシュボードでインスタンス名を作成するために使用されます。

Windows dcstartup.bat ファイルが *DCHOME*¥.gdc¥*toolkit_verion*¥runtime¥j2se*application_alias*.*hostname*.*application_alias* に生成されます。 このファイルは、データ・コレクターとともにアプリケーションを実行するスクリプトです。

Linux AIX dcstartup.sh ファイルが *DCHOME*/.gdc/*toolkit_verion*/runtime/j2se*application_alias*.*hostname*.*application_alias* に生成されます。 このファイルは、データ・コレクターとともにアプリケーションを実行するスクリプトです。

- V8.1.4.0.3 から V8.1.4.0.5 の場合、以下のステップを実行します。

a. プロンプトが出されたら、Java アプリケーションのホーム・ディレクトリーを入力します。例えば、/root/J2seApp/ です。

b. 表示されたリストから Java アプリケーションを選択し、終了します。

- com.ibm.SampleApplication
- com.ibm.DBApplication
- com.ibm.SpringBootApplication

リストに表示されたアプリケーションを選択するか、リストにない他のアプリケーションを選択するには 0 を指定します。

1) 0 を指定した場合、その他のアプリケーションのメイン・クラスのフルネームを入力します。例えば、com.ibm.testApp.Main です。

2) 表示されたリストからオプションを選択した場合、クラス名に基づいて別名が作成されます。この別名が文字の長さ制限を超過する場合は、文字の長さ制限内の別名を指定してください。

重要 : 別名の最大文字制限は、別名 + ホスト名が 24 文字を超えないように計算されます。

c. トランザクション・トラッキングを有効または無効にするオプションを選択します。デフォルト値は「はい」です。

d. 診断データ収集を有効または無効にするオプションを選択します。デフォルト値は「はい」です。

1) *Yes* を選択する場合、メソッド・トレース・モードのオプションを選択します。デフォルト値は *No* です。

e. ステップ b で表示されたリストからオプションを選択した場合、開始スクリプト <DCHOME>/j2se_dc/.gdc/*toolkit_version*/runtime/j2se<application_alias>.<hostname>.<application_alias> を任意の場所にコピーします。

- V8.1.4.0.6 以降の場合、以下のステップを実行します。

a. プロンプトが出されたら、Java アプリケーションのホーム・ディレクトリーを入力します。例えば、/root/J2seApp/ です。

b. モニターするアプリケーション・タイプを選択します。

- Java アプリケーション
- Jetty サーバー

c. アプリケーション・タイプとして *Java Application* を選択する場合は、ステップ 5 のセクション V8.1.4.0.3 から V8.1.4.0.5 に記載されているステップに従います。

d. アプリケーション・タイプとして *Jetty Server* を選択する場合は、以下のステップに従います。

1) Jetty ホーム・ディレクトリーを入力します。例えば、/home/jetty/jetty-distribution-9.4.12.v20180830 です。

2) 別名を入力します。この別名が文字の長さ制限を超過する場合は、文字の長さ制限内の別名を指定してください。

重要 : アプリケーション・タイプとして *Java* アプリケーションを選択し、V8.1.4.0.3 から V8.1.4.0.5 のセクションのステップ b で表示されたリストからオプションを選択した場合は、開始スクリプト <DCHOME>/j2se_dc/.gdc/toolkit_version/runtime/j2se<application_alias>.<hostname>.<application_alias> を任意の場所にコピーします。

e. 選択したタイプが *Jetty Server* である場合は、dcstartup.bat/dcstartup.sh は、指定された Jetty ホーム・ディレクトリーにコピーされます。

- V8.1.4.0.7 の場合、以下の手順を実行します。
  - Open JDK バージョン 9 以降を使用して J2SE データ・コレクターを構成する場合、Java ホームのパスを入力すると、以下の内容で警告が表示されます。

警告 :

```
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by jnr.posix.JavaLibCHelper
(file:/root/testopen/preconf-13march/j2se_dc/.gdc/7.3.0.14.0/
bin/lib/jython.jar)to method sun.nio.ch.SelChImpl.getFD()
WARNING: Please consider reporting this to the maintainers of
jnr.posix.JavaLibCHelper
WARNING: Use --illegal-access=warn to enable warnings of next
illegal reflective access operations
WARNING: All illegal access operations will be denied in next
release Mar 15, 2019 11:35:06 AM
org.python.netty.util.internal.PlatformDependent <clinit>
INFO: Your platform does not provide complete lowlevel API for
accessing direct buffers reliably.
Unless explicitly requested, heap buffer is always preferred
to avoid potential system unstability.
```

しかしながら、J2SE データ・コレクターは正常に動作しているので、この警告は無視して構いません。

V8.1.4.0.2 以前のバージョンの場合、ステップ 6 に従って Windows dcstartup.bat ファイルまたは Linux AIX dcstartup.sh ファイルを変更します。

6. Windows dcstartup.bat または Linux AIX dcstartup.sh ファイルを変更するには、以下のステップを実行します。
   - アプリケーション・クラスおよび JAR ファイルが単一の JAR ファイルにバンドルされている場合は、以下のステップを実行します。
     a. 以下のファイルを開きます。
        - Windows dcstartup.bat
        - Linux AIX dcstartup.sh
     b. -cp .:$classpath:$Classpath $ITCAM_JVM_OPTS メイン・クラスの絶対パス名 を $ITCAM_JVM_OPTS -jar アプリケーション *jar* ファイル に置き換え、ファイルを保存します。
   - アプリケーションで複数の JAR ファイルを使用している場合は、以下のステップを実行します。
     a. 以下のファイルを開きます。
        - Windows dcstartup.bat
        - Linux AIX dcstartup.sh
     b. *CLASSPATH* 変数を JAR ファイルに設定します。
     c. -cp .:$classpath:$Classpath $ITCAM_JVM_OPTS メイン・クラスの絶対パス名 を -cp .:$classpath:$Classpath $ITCAM_JVM_OPTS -jar アプリケーション *jar* ファイル に置き換え、ファイルを保存します。

アプリケーション JAR ファイルには、アプリケーションのメイン・クラスが含まれている必要があります。

注: Windows dcstartup.bat ファイルまたは Linux AIX dcstartup.sh ファイルを V8.1.4.0.3 から V8.1.4.0.5、および V8.1.4.0.6 からそれ以降のバージョンに変更するには (アプリケーション・タイプとして *Java Application* が選択されている場合)、ステップ 7 に従います。

7. アプリケーションで複数の JAR ファイルを使用している場合は、以下のステップを実行します。
   a) 以下のファイルを開きます。
      - Windows dcstartup.bat
      - Linux AIX dcstartup.sh
   b) *CLASSPATH* 変数を JAR ファイルに設定します。
   c) V8.1.4.0.7 では、J2SE データ・コレクターを Java 9 か 10 で構成し、APM への接続に SSL 接続を使用する場合、トランザクション・トラッキングのデータが表示されません。この問題を解決するには、dcstartup.bat ファイルか dcstartup.sh ファイルの最終行に **--add-modules java.xml.bind** フラグを追加します。

      例えば、次の場合があります。
      - アプリケーションが JAR ファイルである場合は、最終行を以下のように更新します。

```
PathToJava --add-modules java.xml.bind --add-opens=
jdk.management/com.sun.management.internal=
ALL-UNNAMED -jar $Classpath $ITCAM_JVM_OPTS AppJarName.jar
```

      - アプリケーションが JAR ファイルにバンドルされていない場合は、最終行を以下のように更新します。

```
PathToJava --add-modules java.xml.bind --add-opens=
jdk.management/com.sun.management.internal=ALL-UNNAMED -cp
.:$classpath:$Classpath $ITCAM_JVM_OPTS FullyQualifiedClassName
```

8. 詳細診断モニターを有効にするには、モニターする必要のある J2SE 固有のクラスとメソッドで custom_request.xml を編集します。これは手動プロセスと自動プロセスの 2 とおりの方法で行うことができます。

   custom_request.xml に J2SE アプリケーション固有のクラスとメソッドを自動で取り込むには、以下を実施します。
   a) <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/ ディレクトリーに移動し、dc.properties ファイルを開きます。
   b) is.auto.update.custom_requests.xml プロパティーの値を *true* に設定して有効にし、ファイルを保存します。
   c) ステップ 9 のコマンドを実行します。
   d) 10 から 15 分後にデータ・コレクターを停止します。
   e) DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/custom/custom_requests.xml にカスタムのメソッドとクラスが取り込まれているかどうかを確認します。
   f) 不要なエントリーを削除し、dc.properties ファイルを再度開きます。
   g) is.auto.update.custom_requests.xml プロパティーの値を *false* に設定して無効にし、ファイルを保存します。
   h) ステップ 9 のコマンドを実行します。

注: アプリケーションの一部のカスタム・メソッドが自動検出されない場合、そのカスタム・メソッドを手動で追加する必要があります。

custom_request.xml ファイルへの取り込みを手動で行うには、以下を実施します。

a) <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/custom/custom_requests.xml に移動し、custom_request.xml を編集します。

例えば、次の場合があります。

```
<edgeRequest>
<requestName>truncateDb</requestName>
<Matches>testApp.JDBC.DBManager</Matches>
<type>application</type>
<methodName>truncateDb</methodName>
</edgeRequest>
```

b) アプリケーション固有のクラスとメソッドを追加します。

9. 次のコマンドを実行します。

Windows dcstartup.bat

Linux AIX dcstartup.sh

注: 選択したアプリケーション・タイプが *Jetty Server* である場合は、Jetty ホーム・ディレクトリーにある dcstartup.bat/dcstartup.sh を実行します。

J2SE アプリケーションが、構成されているデータ・コレクターとともに開始されます。

次のタスク

Cloud APM コンソールにログインして、データ・コレクターによって収集されたデータをダッシュボードで表示します。コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

トランザクション・トラッキングおよび診断データ収集のステータスの確認

V8.1.0.3 以降の場合、エージェント構成ページでトランザクション・トラッキングおよび診断データのステータスを確認できます。

このタスクについて

トランザクション・トラッキングおよび診断データ収集のステータスは、2 つのコマンドを使用して確認できます。これらのコマンドについて、手順を説明します。

手順

1. **config status** コマンドを使用します。

a) bin ディレクトリーを開きます。以下のコマンドを実行します。

Linux AIX **cd <DCHOME>/j2se_dc/.gdc/toolkit_version/bin/**

Windows **cd <DCHOME>¥j2se_dc¥.gdc¥toolkit_version¥bin¥**

ここで、*toolkit_version* は以下のとおりです。

- V8.1.4.0 以前の場合、*toolkit_verion* は 7.3.0.5.0 です。
- V8.1.4.0.1 以降の場合、*toolkit_verion* は 7.3.0.14.0 です。

b) 以下のコマンドを入力して、ステータスを確認します。

Linux AIX **config.sh status**

Windows **config.bat status**

c) 識別された Java アプリケーションをリストから別名で選択してそのステータスを確認するか、または終了を選択します。

1) ddperf

2) Main

3) Exit

2. **config status <application_alias_name>** コマンドを使用します。

a) 以下のコマンドを入力して、ディレクトリーを開きます。

Linux AIX **cd <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/bin/**

Windows **cd <DCHOME>¥j2se_dc¥.gdc¥7.3.0.14.0¥bin¥**

b) 以下のコマンドを入力して、ステータスを確認します。

Linux AIX **config.sh status <application_alias_name>**

Windows **config.bat status <application_alias_name>**

トランザクション・トラッキングおよび診断データ収集のステータスの変更

V8.1.0.3 以降の場合、エージェント構成ページでトランザクション・トラッキングおよび診断データのステータスを変更できます。

このタスクについて

トランザクション・トラッキングおよび診断データ収集のステータスは、コマンド・プロンプトを使用して変更できます。これらのコマンドについては、以下の手順で説明します。

手順

1. bin ディレクトリーを開き、以下のコマンドを実行します。

Linux AIX **cd <DCHOME>/j2se_dc/.gdc/toolkit_version/bin/**

Windows **cd <DCHOME>¥j2se_dc¥.gdc¥toolkit_version¥bin¥**

ここで、*toolkit_version* は以下のとおりです。

- V8.1.4.0 以前の場合、*toolkit_verion* は 7.3.0.5.0 です。
- V8.1.4.0.1 以降の場合、*toolkit_verion* は 7.3.0.14.0 です。

2. ステータスを確認するには、以下のコマンドを入力します。

Linux AIX **config.sh <application_alias_name>**

Windows **config.bat <application_alias_name>**

3. プロンプトが出されたら、トランザクション・トラッキングを有効または無効にするオプションを選択します。デフォルトは Yes です。
4. プロンプトが出されたら、診断データ収集を有効または無効にするオプションを選択します。デフォルトは Yes です。

a)「Yes」の場合、メソッド・トレースを有効/無効にするオプションを選択します。デフォルトは NO です。

JBoss モニターの構成

Monitoring Agent for JBoss は、JBoss Application Server および JBoss Enterprise Application Platform のリソースをモニターします。JBoss エージェントに付属のダッシュボードを使用して、JBoss アプリケーション・サーバー上の最も低速のアプリケーション、最も低速の要求数、スレッド・プールのボトルネック、

JVM ヒープ・メモリーとガーベッジ・コレクションの問題、最もビジーなセッション、その他のボトルネックを特定します。

始める前に

- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、JBoss エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the JBoss エージェント を参照してください。
- JBoss エージェントを構成する前に、以下のタスクを実行して、JBoss サーバーを事前に構成しておく必要があります。
  1. 464 ページの『JMX MBean サーバー接続の有効化』.
  2. 465 ページの『JBoss サーバー管理ユーザーの追加』.
  3. 466 ページの『Web/HTTP 統計収集の有効化』. この手順は、JBoss EAP バージョン 7.x と WildFly バージョン 8.x、9.x、および 10.x 用です。

このタスクについて

管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は 32 文字に制限されています。

指定するインスタンス名は 28 文字 (ホスト名の字数は除く) までに制限されています。例えば、インスタンス名に JBoss を指定すると、管理対象システム名は JBoss:hostname:JE になります。

注 : 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが正しく表示されません。

JBoss エージェントは複数インスタンス・エージェントです。モニターする JBoss サーバーごとにエージェント・インスタンスを作成し、各エージェント・インスタンスを手動で開始する必要があります。

トランザクション・トラッキング機能は、Cloud APM, Advanced オファリングの JBoss エージェント で使用できます。

- 新規エージェント・インスタンスに対してトランザクション・トラッキングを有効にするには、以下の手順のステップ 1 またはステップ 2 を実行してから、473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』に記載された手順を実行します。
- 基本モニター用に既に構成されているエージェント・インスタンスに対してトランザクション・トラッキングを有効にするには、473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』の手順を実行します。
- エージェント・インスタンスに対してトランザクション・トラッキングを無効にするには、475 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターの無効化』の手順を実行します。
- すべてのエージェント・インスタンスのトランザクション・トラッキングをアンインストールして、トランザクション・トラッキング・ツールキットを削除するには、476 ページの『すべての JBoss エージェントのトランザクション・トラッキングのアンインストール』の手順を実行します。

手順

1. 「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用して、Windows システム上でエージェントを構成します。
   - 467 ページの『Windows システムでのエージェントの 構成』.
   - 470 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. コマンド・ライン・スクリプトを実行してプロンプトに応答するか、サイレント応答ファイルを使用して、Linux システム上でエージェントを構成します。
   - 469 ページの『プロンプトへの応答によるエージェントの構成』.

- 470 ページの『サイレント応答ファイルを使用したエージェントの構成』.

3. オプション: トランザクション・トラッキング・データを提供するように個別のエージェント・インスタンスを構成し、トランザクション・トラッキング・データを表示するように Application Performance ダッシュボードを構成して、トランザクション・トラッキングを構成します。

   a) 473 ページの『JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ』の手順に従ってください。

   b) Application Performance ダッシュボードで JBoss エージェントのトランザクション・トラッキング・データを有効化します。

      1) ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。「**エージェント構成**」ページが表示されます。

      2)「**JBoss**」タブを選択します。

      3) モニターする JBoss サーバー・エージェントのチェック・ボックスを選択し、「**アクション**」リストから次のいずれかのアクションを実行します。

         - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
         - トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。

   c) JBoss エージェントのトランザクション・トラッキング・データのダッシュボードを表示するには、JBoss エージェント・インスタンスを Application Performance ダッシュボード 内のアプリケーションに追加します。

      アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

   d) ユーザー・アカウントが、JBoss エージェントのトランザクション・トラッキング・アプリケーション・ダッシュボードの、次のボタンにアクセスするための診断ダッシュボード許可が含まれるロールに割り当てられていることを確認します。
      そのように割り当てられていない場合、アプリケーション・ダッシュボード内で、そのユーザーに対してこれらのボタンが無効になります。

      1)「**最も遅い 5 応答時間**」ウィジェットの「**診断**」ドリルダウン・ボタン。

      2)「**アプリケーション**」ウィジェットの「**処理中要求**」ボタン。

   注 : トランザクション・トラッキングは、Cloud APM, Advanced オファリング内の JBoss エージェントで使用できます。Cloud APM, Base オファリングに含まれる基本的なリソース・モニター機能を持つ JBoss エージェントについては、この手順をスキップしてください。

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログのデフォルト・パスは、以下のとおりです。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

JMX MBean サーバー接続の有効化

JBoss エージェントが JBoss サーバーからデータを収集できるようにするには、Java Management Extensions (JMX) MBean サーバー接続を有効にする必要があります。

手順

ご使用の JBoss サーバーのリリースとバージョン用の手順に従います。

- EAP 5.2 の構成

  run.conf ファイルのバックアップ・コピーを作成してから、このファイルに以下の行を追加します。

```
JAVA_OPTS="$JAVA_OPTS -Djboss.platform.mbeanserver"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.ssl=false"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.authenticate=false"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.port=1090"
JAVA_OPTS="$JAVA_OPTS -Djavax.management.builder.initial=
                       org.jboss.system.server.jmx.MBeanServerBuilderImpl"
```

- AS 6.x の構成

  JBoss サーバーの始動時のパラメーターとしてバインド・アドレスを指定します。

  – Linux *jboss_server_home*/bin/run.sh **-b** Ip_address

  – Windows *jboss_server_home*¥bin¥run.bat **-b** <Ip_address>

  ここで *jboss_server_home* は JBoss サーバーのインストール・ディレクトリーです。

  例えば、バインド・アドレスが 10.77.9.250 の場合、次のようになります。

  /apps/wildfly-9.0.2.Final/bin/run.sh **-b** 10.77.9.250

- 他のすべてのサポート対象バージョンの構成

  JBoss サーバーおよび WildFly サーバーをインストールした時点では、リモート管理用の JMX ポートはデフォルトでは無効になっています。JBoss サーバーの構成を変更して、リモート管理を使用可能にする必要があります。*jboss_server_home*/standalone/configuration/standalone.xml を編集して、リモート管理を使用可能にする必要があります。

  a) *jboss_server_home*/standalone/configuration/standalone.xml ファイルのバックアップ・コピーを作成します。

  ここで *jboss_server_home* は JBoss サーバーのインストール・ディレクトリーです。

  b) リモート構成を許可します。

  urn:jboss:domain:jmx を検索し、その subsystem セクション内で、remoting-connector 項目が use-management-endpoint="true" になっていることを確認します。

  結果例は次のとおりです。

```
<subsystem xmlns="urn:jboss:domain:jmx:1.3">
            <expose-resolved-model/>
            <expose-expression-model/>
            <remoting-connector use-management-endpoint="true"/>
</subsystem>
```

  c) リモート接続を許可します。

  インターフェースが定義されている場所を見つけ、127.0.0.1 (ループバック) を、バインド先のサーバーの外部 IP に置き換えます。0.0.0.0 にはバインドしないでください。

  置換前の例

```
  <interfaces>
        <interface name="management">
            <inet-address value="${jboss.bind.address.management:127.0.0.1}"/>
        </interface>
        <interface name="public">
            <inet-address value="${jboss.bind.address:127.0.0.1}"/>
```

```
      </interface>
...
```

置換後の例 (外部 IP アドレスが 192.168.101.1 の場合)

```
<interfaces>
      <interface name="management">
          <inet-address value="${jboss.bind.address.management:192.168.101.1}"/>
      </interface>
      <interface name="public">
          <inet-address value="${jboss.bind.address:192.168.101.1}"/>
      </interface>
...
```

JBoss サーバー管理ユーザーの追加

JBoss エージェントが JBoss サーバーからデータを収集できるようにするには、管理ユーザーを追加する必要があります (存在しない場合)。

手順

JBoss の **add-user** スクリプトを使用して、管理ユーザーを追加します。

1. JBoss サーバーのインストール・ディレクトリーの下のバイナリー、つまり bin ディレクトリーに移動します。
2. **add-user** スクリプトを実行します。
   - Linux **./add-user.sh**
   - Windows **add-user.bat**
3. プロンプトに従って、管理ユーザーを生成します。

例

```
root@jboss-wf10-rh7:/apps/wildfly-10.0.0.Final/bin
] ./add-user.sh

What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
(a): a

Enter the details of the new user to add.
Using realm 'ManagementRealm' as discovered from the existing property files.
Username : MyAdmin
Password recommendations are listed below. To modify these restrictions edit the add-
user.properties
configuration file.
 - The password should be different from the username
 - The password should not be one of the following restricted values {root, admin,
administrator}
 - The password should contain at least 8 characters, 1 alphabetic character(s), 1 digit(s),
1 non-alphanumeric symbol(s)
Password :
Re-enter Password :
What groups do you want this user to belong to? (Please enter a comma separated list, or leave
blank
for none)[  ]:
About to add user 'MyAdmin' for realm 'ManagementRealm'
Is this correct yes/no? yes
Added user 'MyAdmin' to file '/apps/wildfly-10.0.0.Final/standalone/configuration/mgmt-
users.properties'
Added user 'MyAdmin' to file '/apps/wildfly-10.0.0.Final/domain/configuration/mgmt-
users.properties'
Added user 'MyAdmin' with groups  to file
          '/apps/wildfly-10.0.0.Final/standalone/configuration/mgmt-groups.properties'
Added user 'MyAdmin' with groups  to file
          '/apps/wildfly-10.0.0.Final/domain/configuration/mgmt-groups.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for
server to
server EJB calls.
yes/no? no
```

Web/HTTP 統計収集の有効化

JBoss エージェントが JBoss サーバーの Web メトリックやその他のサブシステム・メトリックを収集できるようにするには、各サブシステムに対して統計の収集を有効にする必要があります。この手順は、JBoss EAP バージョン 7.x と WildFly バージョン 8.x、9.x、および 10.x 用です。

手順

各種 JBoss サブシステムの **statistics-enabled** 属性により、統計収集が制御されます。この設定は、JBoss コマンド・ライン・インターフェースを 使用して表示および更新できます。

注 : この手順は、JBoss EAP バージョン 7.x と WildFly バージョン 8.x、9.x、および 10.x 用です。

1. JBoss サーバーのインストール・ディレクトリーの下のバイナリー、つまり bin ディレクトリーに移動します。
2. JBoss コマンド・ライン・インターフェースを 開始します。
   - Linux **./jboss-cli.sh --connect** [**--controller**=*IP*:*port*]
   - Windows **jboss-cli.bat --connect** [**--controller**=*IP*:*port*]

   ここで *IP* は JBoss サーバーの IP アドレス、*port* は JBoss サーバーのポートです。例: 192.168.10.20:9990

   ヒント : 接続試行の結果として次のエラーが生じる場合があります。「Failed to connect to the controller: The controller is not available at localhost:9990: java.net.ConnectException: WFLYPRT0053: Could not connect to http-remoting://localhost:9990. The connection failed: WFLYPRT0053: Could not connect to http-remoting://localhost:9990. The connection failed: Connection refused」。この場合、**--controller** パラメーターを使用します。

   このエラーは、管理サーバーが localhost の IP アドレス (127.0.0.1) を listen していないこと、およびコンピューターの IP アドレスを listen するように構成されていることを示します。
3. 以下のコマンドを実行して、各サブシステムの statistics-enabled 属性の現在の状態を表示します。

   注 : JBoss をドメイン・モードで実行している場合は、各コマンドの先頭に、関連するプロファイルを指定する必要があります。また、これらのコマンドをモニター対象のプロファイルごとに実行する必要があります。例: /profile=full/subsystem=ejb3:read-attribute(name=statistics-enabled)

   /subsystem=**ejb3**:read-attribute(name=enable-statistics)

   /subsystem=**transactions**:read-attribute(name=statistics-enabled)

   /subsystem=**undertow**:read-attribute(name=statistics-enabled)

   /subsystem=**webservices**:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=pool**:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=jdbc**:read-attribute(name=statistics-enabled)

   ここで *Data_Source_Name* は JBoss で使用するように構成されているデータ・ソースの名前です。

   注 : コマンド /subsystem=datasources:read-resource を使用すると、データ・ソースをリストできます。

   統計が有効ではないときの、結果例は次のとおりです。

```
{
    "outcome" => "success",
```

```
    "result" => false
}
```

4. 以下のコマンドを実行して、各サブシステムの statistics-enabled 属性の値を *true* に変更します。

   /subsystem=**ejb3**:write-attribute(name=enable-statistics, value=true)

   /subsystem=**transactions**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**undertow**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**webservices**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=pool**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=jdbc**:write-attribute(name=statistics-enabled,value=true)

   サブシステムの統計を有効にした場合、次の例のような結果になります。

```
{
    "outcome" => "success",
    "response-headers" => {
        "operation-requires-reload" => true,
        "process-state" => "reload-required"
    }
}
```

5. JBoss コマンド・ライン・インターフェースを 終了します。
6. JBoss サーバーを再始動します。

   注 : 現在実行中の、トランザクション・トラッキングが有効な JBoss エージェントは再始動する必要があります。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、JBoss エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for JBoss**」テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、下線文字、およびマイナス文字のみを使用してください。
   例: jboss01

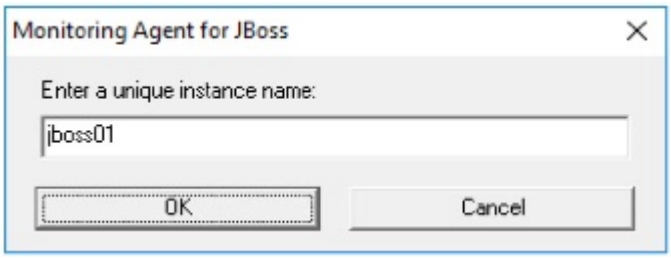


図 *17.* 固有のインスタンス名を入力するウィンドウ

4. JBoss サーバー設定を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、472 ページの表 167 を参照してください。

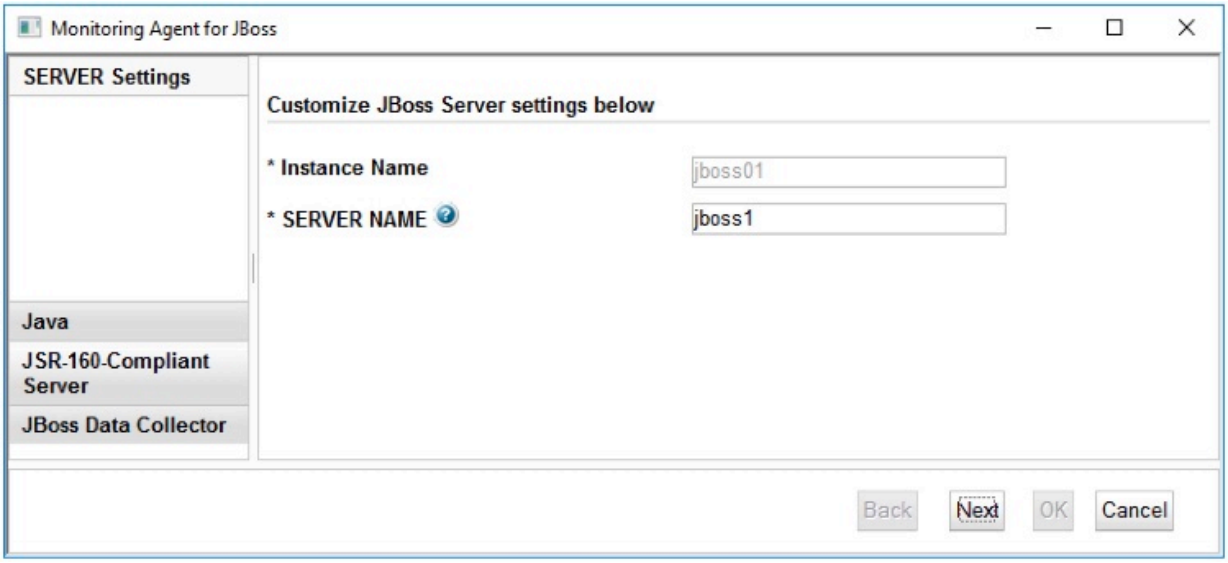


図 *18. JBoss* サーバーの構成パラメーターのウィンドウ

5. Java 設定を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、472 ページの表 167 を参照してください。

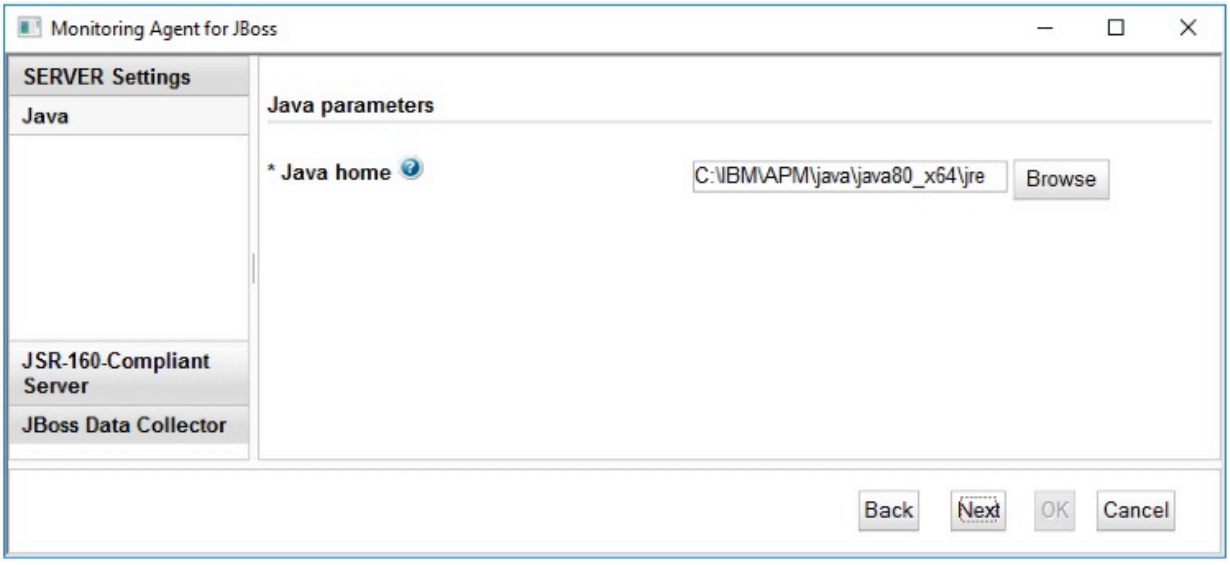


図 *19. Java* 設定を指定するウィンドウ

6. JMX 設定を入力し、「**次へ**」をクリックします。
   各構成パラメーターの説明については、472 ページの表 167 を参照してください。

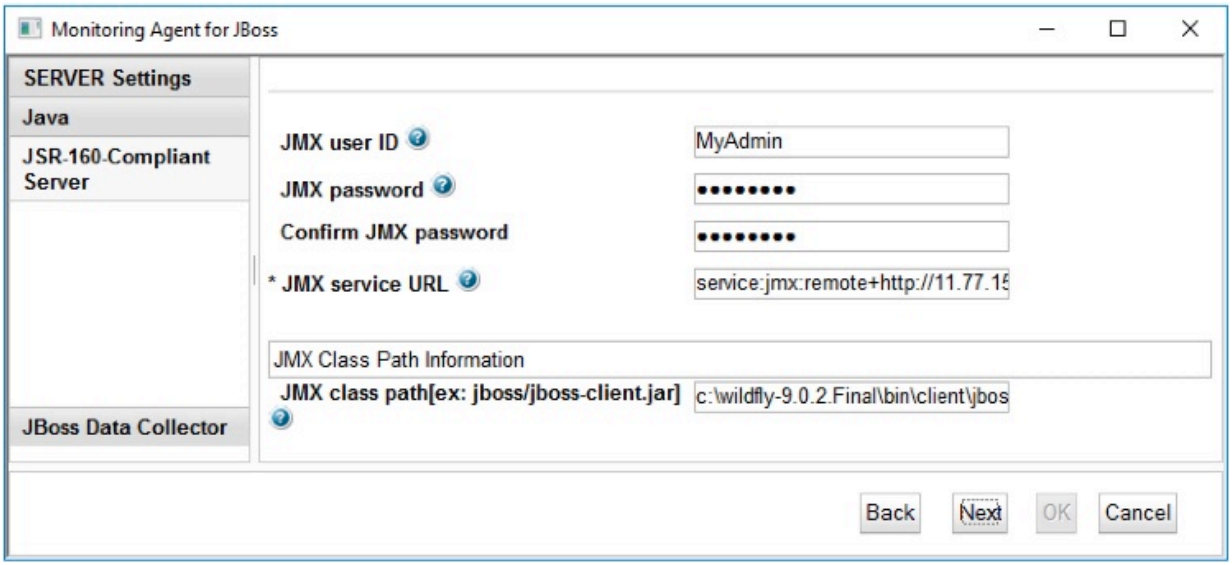


図 *20. JMX* 設定を指定するウィンドウ

7. JBoss エージェント・データ・コレクターの設定を表示します。
   エージェントの初回構成時には、**「DC 実行時ディレクトリー (DC Runtime Directory)」**を空のままにしてください。各構成パラメーターの説明については、472 ページの表 167 を参照してください。

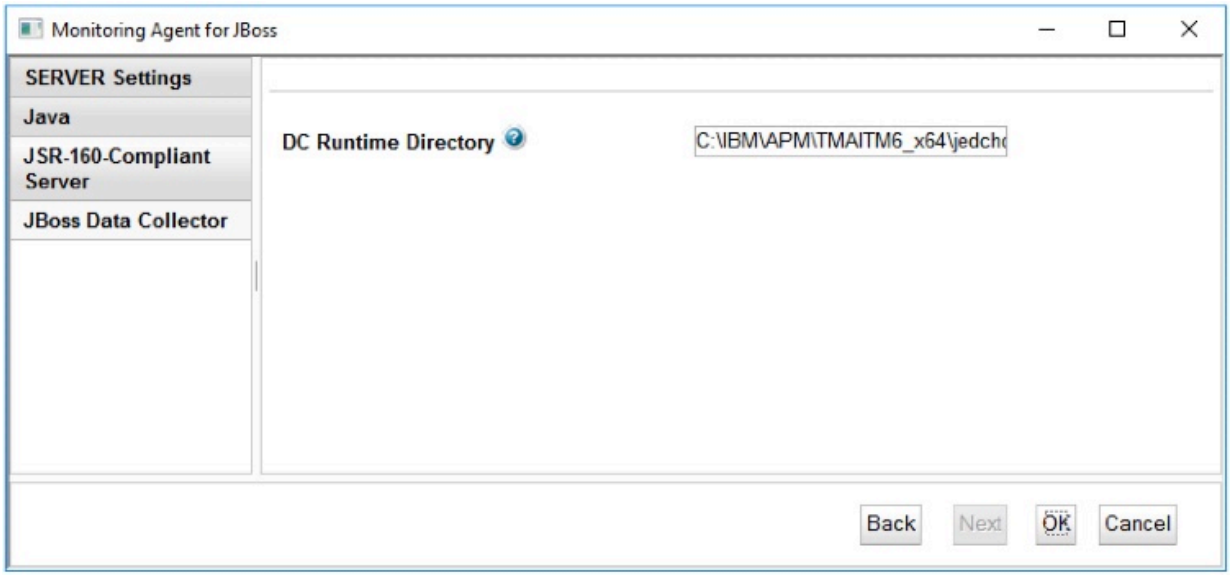


図 *21. JBoss* エージェント・データ・コレクター設定を指定するウィンドウ

8. 「**OK**」をクリックして、エージェント構成を完了します。
9. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

JBoss エージェントのインストール後、エージェントを開始する前に構成する必要があります。JBoss エージェントがローカル Linux または UNIX コンピューターにインストールされている場合、以下の説明に従って、コマンド・ライン・プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

1. コマンド・ラインで、以下のコマンドを実行します。

   ```
   install_dir/bin/jboss-agent.sh config instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/jboss-agent.sh config example-inst01
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、471 ページの『JBoss エージェントの構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/jboss-agent.sh start instance_name
   ```

   ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/jboss-agent.sh start example-inst01
   ```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで JBoss エージェント を構成するには、以下の手順を実行します。
  a) テキスト・エディターで、jboss_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
  - Linux AIX *install_dir*/samples/jboss_silent_config.txt

    例: /opt/ibm/apm/agent/samples/jboss_silent_config.txt
  - Windows *install_dir*¥samples¥jboss_silent_config.txt

  ここで、*install_dir* はエージェントのインストール先パスです。

  install_dir のデフォルト・パスは、以下のとおりです。
  - Linux /opt/ibm/apm/agent
  - Windows C:¥IBM¥APM¥TMAITM6_x64

  例

Linux AIX /opt/ibm/apm/agent/samples/jboss_silent_config.txt

Windows C:¥IBM¥APM¥samples¥jboss_silent_config.txt

b) jboss_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

各構成パラメーターの説明については、471 ページの『JBoss エージェントの構成パラメーター』を参照してください。

c) jboss_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。

- Linux AIX *install_dir*/bin/jboss-agent.sh config **instance_name** *install_dir*/samples/jboss_silent_config.txt
- Windows *install_dir*¥bin¥jboss-agent.bat config **instance_name** *install_dir*¥samples¥jboss_silent_config.txt

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

install_dir のデフォルト・パスは、以下のとおりです。

- Linux /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

重要: サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

例

Linux AIX
```
/opt/ibm/apm/agent/bin/jboss-agent.sh config example-inst01
/opt/ibm/apm/agent/samples/jboss_silent_config.txt
```

Windows
```
C:¥IBM¥APM¥bin¥jboss-agent.bat config example-inst01 C:¥IBM¥APM¥samples
¥jboss_silent_config.txt
```

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux AIX *install_dir*/bin/jboss-agent.sh start **instance_name**
- Windows *install_dir*¥bin¥jboss-agent.bat start **instance_name**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

install_dir のデフォルト・パスは、以下のとおりです。

- Linux /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

例

Linux AIX /opt/ibm/apm/agent/bin/jboss-agent.sh start example-inst01

Windows C:¥IBM¥APM¥bin¥jboss-agent.bat start example-inst01

JBoss エージェントの構成パラメーター

JBoss エージェントの構成パラメーターを表に表示します。

1. JBoss エージェント設定 - JBoss エージェントの環境設定。

2. 472 ページの表 168 - JMX サービス URL の例。

表 *167. JBoss* エージェント設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
サーバー名	JBoss/WildFly サーバーを識別するための名前を指定します。	**KJE_SERVER**
Java ホーム	Java がインストールされるパス。	**JAVA_HOME**
JMX ユーザー ID	MBean サーバーに接続するためのユーザー ID。	**KQZ_JMX_JSR160_JSR160_USER_ID**
JMX パスワード	パスワード	**KQZ_JMX_JSR160_JSR160_PASSWORD**
JMX サービス URL	MBean サーバーに接続するためのサービス URL。 例については、472 ページの表 168 を参照してください。	**KQZ_JMX_JSR160_JSR160_SERVICE_URL**
JMX クラスパス	クラスまたはリソースを見つけるために検索する JAR ファイル。JBoss サーバーの `jboss-client.jar` ファイルを見つけ、そのファイルへのパスを入力します。JBoss EAP 6 サーバーの例: `/opt/EAP-6.3.0/jboss-eap-6.3/bin/client/jboss-client.jar`	**KQZ_JMX_JSR160_JSR160_JAR_FILES**
DC ランタイム・ディレクトリー	**注:** このパラメーターは、Cloud APM, Advanced オファリングに含まれるトランザクション・トラッキング機能を持つ JBoss エージェント専用です。Cloud APM, Base オファリングに含まれる基本的なリソース・モニター機能を持つ JBoss エージェントについては、このパラメーターをスキップします。 JBoss データ・コレクターのランタイム・ディレクトリーへの絶対パスは、**simpleConfig** スクリプトによって設定されます。エージェントの初回構成時には、このパラメーターを空のままにしてください。	**KQZ_DC_RUNTIME_DIR**

表 *168. JMX* サービス *URL*

JBoss サーバーのバージョン	デフォルトのポート付き JMX サービス URL[1]
WildFly 8、9、および 10 JBoss EAP 7	`service:jmx:remote+http://ip:9990` `service:jmx:remote+https://ip:9994`
JBoss EAP 6 JBoss AS 7	`service:jmx:remoting-jmx://ip:9999`
JBoss EAP 5.2 JBoss AS 6.1	`service:jmx:rmi:///jndi/rmi://ip:1090/jmxrmi`

[1] ポートは、JBoss 構成ファイルの項目 <socket-binding name="management-native" interface="management" port="${jboss.management.native.port:*NNNN*}"/> のポートに基づいています。ポートがデフォルト値から変更されている場合、構成ファイル内のポート番号に合わせて調整してください。

JBoss エージェントのトランザクション・トラッキング・データ・コレクターのセットアップ

JBoss エージェントのトランザクション・トラッキング機能は、エージェント・インスタンス環境設定ファイル、JBoss サーバー・スタートアップ・ファイル、および DC ランタイム・ディレクトリー・エージェント構成パラメーターの変更を必要とします。これらの変更を行うためのスクリプトが提供されています。

始める前に

Linux トランザクション・トラッキング・ツールキットが適切に機能するように、オープン・ファイルのリソース制限が 5,000 より大きいことを確認します。

- 現在のオープン・ファイル制限設定を表示します。**ulimit -n**
- オープン・ファイル制限を 5,056 に設定する例: **ulimit -n 5056**

この手順に従う前に、JBoss エージェントの構成のステップ 462 ページの『1』 または 462 ページの『2』 を実行してください。

JBoss エージェントは、トランザクション・トラッキング機能によってモニターされる JBoss サーバーに対してローカル側にインストールされる必要があります。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *JBOSS_HOME* ディレクトリー。
2. *JBOSS_HOME*/bin ディレクトリーおよびファイル。
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml ファイル。
4. *install_dir*/config ディレクトリー。
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg ファイル。

ここで、

JBOSS_HOME
JBoss サーバー のインストール・ディレクトリー。

install_dir
エージェントのインストール先パス。デフォルトのパスは次のとおりです。

- Linux /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

hostname
エージェントがインストールされているホスト・コンピューターの名前。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前。

- Windows システムでのエージェントの 構成の場合は、手順 467 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 470 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 471 ページの『3』

手順

simpleConfig スクリプトを実行します。

1. JBoss エージェントがインストールされている JBoss サーバーにログインします。

2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux *install_dir*
   - Windows *install_dir*¥TMAITM6_x64

   ここで、*install_dir* はエージェントのインストール先パスです。

   install_dir のデフォルト・パスは、以下のとおりです。
   - Linux /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64
3. ディレクトリーを jedchome/7.3.0.13.0/bin に切り替えます。
4. セットアップ・スクリプトを実行します。
   - Linux ./simpleConfig.sh
   - Windows simpleConfig.bat
5. プロンプトに従って、次のようにご使用の環境に合わせてパラメーターを入力します。
   a) エージェント・インスタンス用に選択した JBoss エージェント *instance_name* を入力します。
   b) JBoss サーバーのインストール・ディレクトリーを入力します。
   JBOSS_HOME 環境変数が設定されている場合、その値がデフォルト値として提供されます。

   ここで、

   JBOSS_HOME
   JBoss サーバー のインストール・ディレクトリー。

   instance_name
   エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。
   - Windows システムでのエージェントの 構成の場合は、手順 467 ページの『3』
   - プロンプトへの応答によるエージェントの構成の場合は、手順 470 ページの『1』
   - サイレント応答ファイルを使用したエージェントの構成の場合は、手順 471 ページの『3』

   install_dir
   エージェントのインストール先パスです。デフォルトのパスは次のとおりです。
   - Linux /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64
6. JBoss サーバーおよびエージェントが稼働中の場合は再始動します。

タスクの結果

トランザクション・トラッキングの構成中に変更される JBoss サーバー・ファイル:

- *JBOSS_HOME*/bin/standalone.conf

  このファイルは、トランザクション・トラッキング機能に必要な構成設定で更新されます。トランザクション・トラッキング機能を無効にするときに使用するために、構成マーカーがファイルに挿入されます。トランザクション・トラッキング機能の変更を追加または削除する前に、バックアップ・ファイルが *JBOSS_HOME*/bak ディレクトリーに保存されます。
- *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml

  このファイルは、JAVA EE API モジュール従属関係によって更新されます。トランザクション・トラッキング機能を無効にするときに使用するために、構成マーカーがファイルに挿入されます。トランザクション・トラッキング機能の変更を追加または削除する前に、バックアップ・ファイルが *JBOSS_HOME*/bak ディレクトリーに保存されます。

トランザクション・トラッキングの構成中に変更されるエージェント・ファイル:

- エージェント・インスタンス構成ファイル

  - Linux *install_dir*/config/*hostname*_je_*instance_name*.cfg
  - Windows *install_dir*¥TMAITM6_x64¥*hostname*_JE_*instance_name*.cfg
- エージェント環境設定ファイル
  - Linux *install_dir*/config/je_*instance_name*.environment
  - Windows *install_dir*¥TMAITM6_x64¥KJEENV_*instance_name*

ここで、

JBOSS_HOME
JBoss サーバー のインストール・ディレクトリー。

install_dir
エージェントのインストール先パス。デフォルトのパスは次のとおりです。

- Linux /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

hostname
エージェントがインストールされているホスト・コンピューターの名前。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前。

- Windows システムでのエージェントの 構成の場合は、手順 467 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 470 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 471 ページの『3』

JBoss エージェントのトランザクション・トラッキング・データ・コレクターの無効化

JBoss エージェントのトランザクション・トラッキング機能は、エージェント・インスタンス環境設定ファイル、JBoss サーバー・スタートアップ・ファイル、および DC ランタイム・ディレクトリー・エージェント構成パラメーターの変更を必要とします。トランザクション・トラッキングが有効になっているエージェント・インスタンスに対するこれらの変更を削除するためのスクリプトが提供されています。

始める前に

JBoss サーバーおよび JBoss エージェントがシャットダウンされていることを確認します。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *JBOSS_HOME* ディレクトリー
2. *JBOSS_HOME*/bin ディレクトリーとファイル
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml ファイル
4. *install_dir*/config ディレクトリー
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg ファイル

手順

remove オプションを指定して **simpleConfig** スクリプトを実行します。

1. JBoss エージェントがインストールされている JBoss サーバーにログインします。
2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux *install_dir*
   - Windows *install_dir*¥TMAITM6_x64
3. ディレクトリーを jedchome/7.3.0.13.0/bin に切り替えます。

4. **remove** オプションを指定して **simpleConfig** を実行します。
   - Linux `./simpleConfig.sh remove instance_name`
   - Windows `simpleConfig.bat remove instance_name`
5. JBoss サーバーおよびエージェントを始動します。

   各部の意味は次のとおりです。

   JBOSS_HOME
   JBoss サーバー のインストール・ディレクトリー

   hostname
   エージェントがインストールされているホスト・コンピューターの名前

   instance_name
   エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。
   - Windows システムでのエージェントの 構成の場合は、手順 467 ページの『3』
   - プロンプトへの応答によるエージェントの構成の場合は、手順 470 ページの『1』
   - サイレント応答ファイルを使用したエージェントの構成の場合は、手順 471 ページの『3』

   install_dir
   エージェントのインストール先パスです。デフォルトのパスは次のとおりです。
   - Linux `/opt/ibm/apm/agent`
   - Windows `C:¥IBM¥APM¥TMAITM6_x64`

   アーキテクチャー
   IBM Application Performance Management または Cloud APM システム・アーキテクチャー ID。 例えば、lx8266 は、Linux Intel v2.6 (64 ビット) を表します。アーキテクチャー・コードの完全なリストについては、*install_dir*/registry/archdsc.tbl ファイルを参照してください。

すべての JBoss エージェントのトランザクション・トラッキングのアンインストール

JBoss エージェントのトランザクション・トラッキング機能をアンインストールできます。トランザクション・トラッキングが有効になっているすべてのエージェント・インスタンスを削除して、トランザクション・トラッキング・ツールキットも削除するためのスクリプトが提供されています。

始める前に

JBoss サーバーおよびすべての JBoss エージェント・インスタンスがシャットダウンされていることを確認します。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *JBOSS_HOME* ディレクトリー。
2. *JBOSS_HOME*/bin ディレクトリーおよびファイル。
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml ファイル。
4. *install_dir*/config ディレクトリー。
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg ファイル。

手順

uninstall オプションを指定して **simpleConfig** スクリプトを実行します。

1. JBoss エージェントがインストールされている JBoss サーバーにログインします。
2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux *install_dir*/*architecture*/je/bin。例: /opt/ibm/apm/agent/lx8266/je/bin または /opt/ibm/apm/agent/lx8266/je/bin

- Windows *install_dir*¥TMAITM6_x64

3. ディレクトリーを jedchome/7.3.0.13.0/bin に切り替えます。
4. **uninstall** オプションを指定して **simpleConfig** を実行します。
   - Linux ./simpleConfig.sh **uninstall**
   - Windows simpleConfig.bat **uninstall**
5. JBoss サーバーおよびすべてのエージェント・インスタンスを始動します。

   各部の意味は次のとおりです。

   JBOSS_HOME
   JBoss サーバー のインストール・ディレクトリー。

   hostname
   エージェントがインストールされているホスト・コンピューターの名前。

   instance_name
   エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。
   - Windows システムでのエージェントの 構成の場合は、手順 467 ページの『3』
   - プロンプトへの応答によるエージェントの構成の場合は、手順 470 ページの『1』
   - サイレント応答ファイルを使用したエージェントの構成の場合は、手順 471 ページの『3』

   install_dir
   エージェントのインストール先パスです。デフォルトのパスは次のとおりです。
   - Linux /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64

   アーキテクチャー
   IBM Application Performance Management または Cloud APM システム・アーキテクチャー ID。 例えば、lx8266 は、Linux Intel v2.6 (64 ビット) を表します。アーキテクチャー・コードの完全なリストについては、*install_dir*/registry/archdsc.tbl ファイルを参照してください。

Linux KVM のモニターの構成

Red Hat Enterprise Virtualization Hypervisor (RHEVH) サーバーと Red Hat Enterprise Virtualization Manager (RHEVM) サーバーのデータを収集するように Monitoring Agent for Linux KVM を構成する必要があります。エージェントをサーバーまたは仮想マシンにインストールした後、最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Linux KVM エージェント を参照してください。

このタスクについて

Linux KVM エージェントは、複数インスタンスかつ複数接続のエージェントです。複数インスタンスとは、複数のインスタンスを作成することができ、各インスタンスが 1 つ以上の RHEVM サーバーまたは RHEVH サーバーに対して複数の接続を行うことができることを意味します。

要確認 : RHEVM サーバーと RHEVH サーバーのモニターには、異なるインスタンスを使用してください。

製品のバージョンとエージェントのバージョンが異なっていることがよくあります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』 を参照してください。

RHEVM サーバーおよび RHEVH サーバーのインスタンスは、同じ構成スクリプトを使用して構成することができます。

- RHEVM サーバーへの接続を構成するには、「RHEVM サーバーへの接続の構成」トピックに記載されている手順を実行します。
- RHEVH サーバーへの接続を構成するには、「RHEVH サーバーへの接続の構成」トピックに記載されている手順を実行します。

ユーザーの作成および必要な権限の付与

Linux KVM エージェントを構成する前に、ユーザーを作成し、そのユーザーに RHEVM サーバーおよび RHEVH サーバーをモニターするために必要な権限を付与する必要があります。

手順

1. **Red Hat Enterprise Virtualization Manager Web 管理**ポータルを開きます。
2. 「**構成**」をクリックします。
3. 「**構成**」ウィンドウで、「**役割**」を選択します。
   a) 役割を作成するには、「**新規**」をクリックします。
   b) 「**新しい役割 (New Role)**」ウィンドウで、役割の名前を追加して、アカウント・タイプとして「**管理 (Admin)**」を選択します。
   c) 「**Check boxes to Allow Action**」ペインのチェック・ボックスが選択されていないことを確認し、「**OK**」をクリックします。
4. 「**構成**」ウィンドウで、「**システム許可 (System Permission)**」を選択します。
   a) ユーザー権限を付与するには、「**追加**」をクリックします。
   b) 「**ユーザーへのシステム許可の追加 (Add System Permission to User)**」ウィンドウで、許可を付与するユーザーを選択します。
   c) 「**ユーザーへの役割の割り当て (Assign role to user)**」リストから、作成した役割を選択し、「**OK**」をクリックします。

次のタスク

次のエージェント構成を実行します。

- 484 ページの『RHEVH サーバーへの接続の構成』
- 483 ページの『RHEVM サーバーへの接続の構成』

プロトコルの構成

エージェントは、さまざまなプロトコルを使用して RHEVH サーバーに接続します。SSH、TLS、TCP から任意のプロトコルを構成できます。

このタスクについて

Linux KVM エージェントは、QEMU-KVM 仮想マシンを管理する **virsh** ツールを使用することで、各ハイパーバイザーにリモートで接続し、メトリックを収集します。エージェント環境の `libvirt` API は、複数の異なるリモート・トランスポート・プロトコルを使用します。サポートされるプロトコルのリストについては、『Remote support』ページを参照してください。

SSH プロトコルの構成

リモートでホストをモニターするように SSH プロトコルを構成できます。

このタスクについて

前提: Linux KVM エージェントはホスト A にインストールされています。リモートでホスト B 上のハイパーバイザーをモニターします。

手順

1. Linux KVM エージェント・プロセスを実行する場合と同じユーザー ID (例: root ユーザー ID) を指定して、ホスト A にログインします。

ヒント : SSH 接続を受け入れるホスト B の ID と、ホスト A の root ユーザー ID が分かっている必要があります。

2. **ssh-keygen** ユーティリティーを 使用して、ホスト A で **id_rsa** 鍵と *id_rsa.pub* 鍵を生成します。

   鍵は ~/.ssh: $ ssh-keygen -t rsa に保存されます。

3. 次のように、ホスト B から認証鍵をコピーします。

   $ scp Id on host B@name or IP address of host B:~/.ssh/authorized_keys ~/.ssh/authorized_keys_from_B

4. ホスト A の公開鍵をホスト B の認証鍵の末尾に付加します。

   cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys_from_B

5. ホスト B に許可された鍵をコピーして戻します。

   $ scp ~/.ssh/authorized_keys_from_B Id on host B@name or IP address of host B:~/.ssh/authorizede_keys

   要確認 : 複数のホストをモニターする場合、ホストごとにステップ 479 ページの『3』、479 ページの『4』、および 479 ページの『5』 を繰り返します。

6. ホスト B でコピーした許可された鍵を削除します。

   ~/.ssh/authorized_keys_from_B

7. ホスト A の現行 ID の **~/.bash_** プロファイルに以下のコマンドを追加します。

   $ eval `ssh-agent`

   要確認 : 単一引用符 (') ではなく、必ず、US キーボードのティルド (~) の下にある逆単一引用符 (`) を使用してください。

8. ID をホスト A に追加し、ID の作成時に使用したパスワードを入力します。

   $ ssh-add ~/.ssh/id_rsa

9. 「Could not open a connection to your authentication agent」メッセージを受け取った場合、以下のコマンドを実行します。

   exec ssh-agent bash

   ヒント : bash の代わりにご使用の shell を指定して、次のコマンドを再び実行することができます。

   $ ssh-add ~/.ssh/id_rsa

10. 次のように、SSH プロトコルをテストして、SSH パスワードを入力せずにホスト A からホスト B に接続することを確認します。

    ヒント : 複数のホストをモニターする場合、次のコマンドを使用して各ホストの接続をテストしてください。

    $ ssh Id on host B@name or IP address of host B

11. 接続を確認するために、次のコマンドを実行します。

    virsh -c qemu+ssh://Id on host B@name or IP address of host B:port/system

    デフォルトの SSH ポートを変更していない場合、コマンドの **:port** セクションは省略してください。

    重要 : virsh コマンドが正常に終了すると、Linux KVM エージェントがハイパーバイザーに接続します。

12. ホスト A で Linux KVM エージェントを再開する前に、ホスト A を再始動する必要があります。再始動するためには、**ssh-add** コマンドを再度実行し、その都度パスワードを指定します。

    ヒント : SSH 鍵チェーンを使用すると、パスワードを再入力する必要がなくなります。

TLS プロトコルの構成
リモートでホストをモニターするように TLS プロトコルを構成できます。

このタスクについて
前提: Linux KVM エージェントはホスト A にインストールされています。リモートでホスト B 上のハイパーバイザーをモニターします。

手順
1. ハイパーバイザー内に認証局 (CA) 鍵と証明書を作成するために、次の手順を実行します。
   a) ホスト B にログインします。
   b) 次のように、一時ディレクトリーを作成し、この一時ディレクトリーにパスを切り替えます。

      ```
      mkdir cert_files
      cd cert_files
      ```

   c) 次のように、2048 ビットの RSA 鍵を作成します。

      ```
      openssl genrsa -out cakey.pem 2048
      ```

   d) 次のように、ローカル CA の自己署名証明書を作成します。

      ```
      openssl req -new -x509 -days 1095 -key cakey.pem -out ¥
      cacert.pem -sha256 -subj "/C=US/L=Austin/O=IBM/CN=my CA"
      ```

   e) 次のように、CA 証明書を検査します。

      ```
      openssl x509 -noout -text -in cacert.pem
      ```

2. ハイパーバイザー内のクライアントとサーバーの鍵および証明書を作成するために、次の手順を実行します。
   a) 次のように鍵を作成します。

      ```
      openssl genrsa -out serverkey.pem 2048
      openssl genrsa -out clientkey.pem 2048
      ```

   b) 次のようにサーバーの証明書署名要求を作成します。

      要確認 : サーバー証明書要求で使用される kvmhost.company.org アドレスを、ハイパーバイザー・ホストの完全修飾ドメイン名に変更してください。

      ```
      openssl req -new -key serverkey.pem -out serverkey.csr ¥
      -subj "/C=US/O=IBM/CN=kvmhost.company.org"
      ```

   c) 次のようにクライアントの証明書署名要求を作成します。

      ```
      openssl req -new -key clientkey.pem -out clientkey.csr ¥
      -subj "/C=US/O=IBM/OU=virtualization/CN=root"
      ```

   d) 次のようにクライアント証明書とサーバー証明書を作成します。

      ```
      openssl x509 -req -days 365 -in clientkey.csr -CA cacert.pem ¥
      -CAkey cakey.pem -set_serial 1 -out clientcert.pem
      openssl x509 -req -days 365 -in serverkey.csr -CA cacert.pem ¥
      -CAkey cakey.pem -set_serial 94345 -out servercert.pem
      ```

   e) 次のように鍵を検査します。

      ```
      openssl rsa -noout -text -in clientkey.pem
      openssl rsa -noout -text -in serverkey.pem
      ```

   f) 次のように証明書を検査します。

      ```
      openssl x509 -noout -text -in clientcert.pem
      openssl x509 -noout -text -in servercert.pem
      ```

3. ホスト・サーバーに鍵と証明書を配布するために、次の手順を実行します。
   a) 次のように、CA 証明書である cacert.pem ファイルをディレクトリー /etc/pki/CA にコピーします。

      cp cacert.pem /etc/pki/CA/cacert.pem

   b) /etc/pki/libvirt ディレクトリーを作成し、/etc/pki/libvirt ディレクトリーに servercert.pem サーバー証明書ファイルをコピーします。root ユーザーのみが秘密鍵にアクセスできるようにします。

      mkdir /etc/pki/libvirt

      cp servercert.pem /etc/pki/libvirt/.

      chmod -R o-rwx /etc/pki/libvirt

      要確認: 鍵または証明書の名前に誤りがある場合、あるいは正しくないディレクトリーにコピーされている場合、許可は失敗します。

   c) /etc/pki/libvirt/private ディレクトリーを作成し、/etc/pki/libvirt/private ディレクトリーに serverkey.pem サーバー鍵ファイルをコピーします。root ユーザーのみが秘密鍵にアクセスできるようにします。

      mkdir /etc/pki/libvirt/private

      cp serverkey.pem /etc/pki/libvirt/private/.

      chmod -R o-rwx /etc/pki/libvirt/private

      要確認: 鍵または証明書の名前に誤りがある場合、あるいは正しくないディレクトリーにコピーされている場合、許可は失敗します。

   d) 次のように、ファイルの配置が正しいかどうかを検査します。

      find /etc/pki/CA/*|xargs ls -l

      ls -lR /etc/pki/libvirt

      ls -lR /etc/pki/libvirt/private

      要確認: 鍵または証明書の名前に誤りがある場合、あるいは正しくないディレクトリーにコピーされている場合、許可は失敗します。

4. クライアントまたは管理ステーションに鍵と証明書を配布するために、次の手順を実行します。
   a) ホスト A にログインします。
   b) ホスト A の /etc/pki/CA ディレクトリーに、ホストの CA 証明書である cacert.pem をファイル名を変更せずにコピーします。

      scp kvmhost.company.org:/tmp/cacert.pem /etc/pki/CA/

   c) /etc/pki/libvirt ディレクトリーに、ホスト B のクライアント証明書である clientcert.pem ファイルをコピーします。デフォルトのファイル名を使用し、root ユーザーのみが秘密鍵にアクセスできるようにします。

      mkdir /etc/pki/libvirt/

      scp kvmhost.company.org:/tmp/clientcert.pem /etc/pki/libvirt/.

      chmod -R o-rwx /etc/pki/libvirt

      要確認: 鍵または証明書の名前に誤りがある場合、あるいは正しくないディレクトリーにコピーされている場合、許可は失敗します。

   d) /etc/pki/libvirt/private ディレクトリーに、ホストのクライアント鍵である clientkey.pem をコピーします。デフォルトのファイル名を使用し、root ユーザーのみが秘密鍵にアクセスできるようにします。

      mkdir /etc/pki/libvirt/private

      scp kvmhost.company.org:/tmp/clientkey.pem /etc/pki/libvirt/private/.

chmod -R o-rwx /etc/pki/libvirt/private

要確認 : 鍵または証明書の名前に誤りがある場合、あるいは正しくないディレクトリーにコピーされている場合、許可は失敗します。

e) 次のように、ファイルの配置が正しいかどうかを検査します。

ls -lR /etc/pki/libvirt

ls -lR /etc/pki/libvirt/private

5. libvirtd デーモン構成を編集するために、次の手順を実行します。
   a) ホスト B にログインします。
   b) /etc/sysconfig/libvirtd ファイルと /etc/libvirt/libvirtd.conf ファイルのコピーを作成します。
   c) /etc/sysconfig/libvirtd ファイルを編集し、**--listen** パラメーターが libvirtd デーモンに渡されるようにします。この手順により、libvirtd デーモンがネットワーク接続を listen するようになります。
   d) /etc/libvirt/libvirtd.conf ファイルを編集し、libvirtd.conf ファイル内の **tls_allowed_dn_list** ディレクティブを 使用して、許可された 1 組のサブジェクトを構成します。

   重要 : サブジェクト内のフィールドの順序は、証明書の作成に使用した順序と同じにする必要があります。

   e) 次のように、変更を反映するために libvirtd デーモン・サービスを再始動します。

/etc/init.d/libvirtd restart

6. ファイアウォール構成を変更するために、セキュリティー・レベルの構成にアクセスし、トラステッド・ポートとして TCP ポート 16514 を追加します。
7. リモート管理が機能していることを検査するために、ホスト A で次のコマンドを実行します。

virsh -c qemu+tls://kvmhost.company.org/system list --all

TCP プロトコルの構成

TCP プロトコルは、テストのみに使用します。

このタスクについて

前提: Linux KVM エージェントはホスト A にインストールされています。リモートでホスト B 上のハイパーバイザーをモニターします。

手順

1. ホスト B にログインします。
2. /etc/libvirt/libvirtd.conf ファイルを編集して、**listen_tcp** パラメーターを有効にし、**tcp_port** パラメーターの値がデフォルト値である 16509 に設定されるようにします。
3. /etc/libvirt/libvirtd.conf ファイルを編集して、**auth_tcp** パラメーターを「none」に設定します。この手順により、TCP が接続を認証しないことが指定されます。
4. **--listen** フラグを指定して **libvirt** デーモンを実行するか、/etc/sysconfig/libvirtd ファイルを編集して LIBVIRTD_ARGS="--listen" 行のコメントを外すことで、ホスト B で listen モードでデーモンを再開します。
5. 接続を確認するために、次のコマンドを実行します。

   virsh -c qemu+tcp://kvmhost.company.org:port/system

   デフォルトの TCP ポートを変更していない場合、コマンドの **:port** セクションは省略してください。

   重要 : virsh コマンドが正常に終了すると、Linux KVM エージェントがハイパーバイザーに接続します。

次のタスク

484 ページの『RHEVH サーバーへの接続の構成』で説明しているステップを実行して、エージェントを構成します。

RHEVM サーバーへの接続の構成

RHEVM サーバーへの接続を構成するには、スクリプトを実行してプロンプトに応答する必要があります。

始める前に

1. 次のパスから入手できるセキュリティー証明書をダウンロードします。

   ```
   https://RHEVM-HOST:RHEVM-PORT/ca.crt
   ```

   表の見方

   RHEVM-HOST
   ホストの名前。

   RHEVM-PORT
   RHEVM 環境で使用するポート。

2. 次のように、*keytool* ユーティリティーを 使用して、ローカル鍵ストア・ファイルを生成するためにセキュリティー証明書ファイルをインポートします。

   keytool -import -alias ALIAS -file CERTIFICATE_FILE -keystore KEYSTORE_FILE

   例: **keytool -import -alias RHEVM36vmwt9 -file hjs495-vmw-t-9.cer -keystore RHEVM36KeyStore**

   表の見方

   ALIAS
   エージェントの証明書トラストストアに追加される各証明書の固有の参照です。例えば、*datasource.example.com* からの証明書に対する適切な別名は *datasource* です。

   CERTIFICATE_FILE
   トラストストアに追加するデータ・ソース証明書の完全パスとファイル名。

   KEYSTORE_FILE
   指定したい鍵ストア・ファイルの名前。

   ヒント: *keytool* ユーティリティーは Java ランタイム環境 (JRE) で使用可能です。鍵ストア・ファイルは、コマンドを実行したのと同じ場所に保管されます。

3. RHEVM に接続するユーザーが SuperUser 役割を備えている管理者であるようにします。 既存のユーザー ID を使用することも、478 ページの『ユーザーの作成および必要な権限の付与』に記載されている手順を実行して、新しいユーザー ID を作成することもできます。

手順

1. コマンド・ラインで、以下のコマンドを実行します。

   install_dir/bin/linux_kvm-agent.sh config instance_name

   例: **/opt/ibm/apm/agent/bin/linux_kvm-agent.sh config instance_name**

   表の見方

   instance_name
   インスタンスに指定する名前です。

   install_dir
   エージェントがインストールされるパスです。

2. プロンプトに応答し、構成パラメーターの値を指定します。

構成パラメーターについては、485 ページの『RHEVM サーバーに接続するための構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   install_dir/bin/linux_kvm-agent.sh start instance_name

   例: **/opt/ibm/apm/agent/bin/linux_kvm-agent.sh start instance_name**

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

RHEVH サーバーへの接続の構成

RHEVH サーバーへの接続を構成するには、スクリプトを実行してプロンプトに応答する必要があります。

始める前に

- RHEVM に接続しているユーザーが、root ユーザーであることを確認してください。既存のユーザー ID を使用することも、478 ページの『ユーザーの作成および必要な権限の付与』に記載されている手順を実行して、新しいユーザー ID を作成することもできます。
- 478 ページの『プロトコルの構成』で説明されているステップを実行して、RHEVH サーバーへの接続に使用するプロトコルを構成します。

手順

1. コマンド・ラインで、以下のコマンドを実行します。

   install_dir/bin/linux_kvm-agent.sh config instance_name

   例: **/opt/ibm/apm/agent/bin/linux_kvm-agent.sh config instance_name**

   表の見方

   instance_name
   インスタンスに指定する名前です。

   install_dir
   エージェントがインストールされるパスです。

2. プロンプトに応答し、構成パラメーターの値を指定します。

   構成パラメーターについては、486 ページの『RHEVH サーバーに接続するための構成パラメーター』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

   install_dir/bin/linux_kvm-agent.sh start instance_name

   例: **/opt/ibm/apm/agent/bin/linux_kvm-agent.sh start instance_name**

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

RHEVM サーバーに接続するための構成パラメーター

RHEVM サーバーを使用してエージェントを接続するために使用される構成パラメーターのデフォルト値を変更できます。

以下の表に、構成パラメーターの詳細な説明を示します。

表 *169. RHEVM* サーバーに接続するための構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
Monitoring Agent for Linux KVM 設定の編集 (Edit Monitoring Agent for Linux KVM settings)	構成パラメーターのデフォルト値の編集を開始できることを示します。「1 （はい）」 (デフォルト値でもある) を入力して続行します。	はい
データ・プロバイダー・ログ・ファイルの最大数	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。	はい
各データ・プロバイダー・ログの最大サイズ (KB)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。	はい
データ・プロバイダー・ログの詳細レベル	データ・プロバイダーによって作成されるログ・ファイルに含めることができる詳細のレベル。デフォルト値は 4 (Info) です。有効な値は以下のとおりです。 • 1 = オフ: メッセージは記録されません。 • 2 = 重大: エラーのみが記録されます。 • 3 = Warning (警告): 重大レベルで記録されるすべてのエラーおよびメッセージと、望ましくない動作の原因となる可能性がある潜在的なエラー。 • 4 = 情報: 警告レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した概要的な情報メッセージ。 • 5 = 詳細レベル (低): 情報レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した詳しい情報メッセージ。 • 6 = Finer (詳細レベル (中)): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、パフォーマンスのプロファイル情報およびデバッグ・データなどの非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 7 = Finest (詳細レベル (高)): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、詳細なプログラミング・メッセージおよびデータを含む、非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 8 = すべて: すべてのエラーおよびメッセージが記録されます。	はい
ハイパーバイザー設定の編集 (Edit Hypervisor settings)	RHEVH サーバーへの接続用のパラメーターを編集するかどうかを示します。RHEVM サーバーへの接続を構成するため、「5 （次へ)」を入力します。デフォルト値は 5 （Next) です。	はい

表 169. *RHEVM* サーバーに接続するための構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
RHEVM 接続詳細設定の編集 (Edit RHEVM Connection Details settings)	RHEVM サーバーへの接続用のパラメーターを編集するかどうかを示します。1 (Add) と入力して続行します。デフォルト値は 5 (Next) です。 **重要 :** すべての構成パラメーターの値を指定した後に、パラメーターの編集を続行するかどうかを指示するように求めるプロンプトが再び出されます。5 (Exit) と入力します。	はい
RHEVM ID	接続先の RHEVM に対して指定する固有のユーザー名。	はい
ホスト	RHEVM サーバーへの接続に使用するデータ・ソースのホスト名または IP アドレス。	はい
ユーザー	RHEVM サーバーに接続するための十分な特権を持っている、データ・ソースのユーザー名。	はい
パスワード	RHEVM サーバーへの接続に使用するユーザー名のパスワード。	はい
パスワードの再入力 (Re-type password)	「**パスワード**」フィールドで指定したものと同じパスワード。	はい
ポート	RHEVM サーバーに接続するために使用されるポート番号。	はい
ドメイン	ユーザーが属するドメイン。	はい
KeyStorePath	**keytool** コマンドを使用して作成したローカル鍵ストア・ファイルのファイル・パスと名前。	はい

RHEVH サーバーに接続するための構成パラメーター

RHEVH サーバーを使用してエージェントを接続するために使用される構成パラメーターのデフォルト値を変更できます。

以下の表に、構成パラメーターの詳細な説明を示します。

表 170. ハイパーバイザーに接続するための構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
Monitoring Agent for Linux KVM 設定の編集 (Edit Monitoring Agent for Linux KVM settings)	構成パラメーターのデフォルト値の編集を開始できることを示します。「1 (はい)」 (デフォルト値でもある) を入力して続行します。	はい
データ・プロバイダー・ログ・ファイルの最大数	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。	はい
各データ・プロバイダー・ログの最大サイズ (KB)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。	はい

表 170. ハイパーバイザーに接続するための構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
データ・プロバイダー・ログの詳細レベル	データ・プロバイダーによって作成されるログ・ファイルに含めることができる詳細のレベル。デフォルト値は 4 (Info) です。有効な値は以下のとおりです。 • 1 = オフ: メッセージは記録されません。 • 2 = 重大: エラーのみが記録されます。 • 3 = Warning (警告): 重大レベルで記録されるすべてのエラーおよびメッセージと、望ましくない動作の原因となる可能性がある潜在的なエラー。 • 4 = 情報: 警告レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した概要的な情報メッセージ。 • 5 = 詳細レベル (低): 情報レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した詳しい情報メッセージ。 • 6 = Finer (詳細レベル (中)): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、パフォーマンスのプロファイル情報およびデバッグ・データなどの非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 7 = Finest (詳細レベル (高)): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、詳細なプログラミング・メッセージおよびデータを含む、非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 8 = すべて: すべてのエラーおよびメッセージが記録されます。	はい
ハイパーバイザー設定の編集 (Edit Hypervisor settings)	ハイパーバイザー接続用のパラメーターを編集するかどうかを示します。1 （Add）と入力します。デフォルト値は 5 （Next）です。	はい
ハイパーバイザー ID	接続先の RHEVH に対して指定する固有のユーザー名。	はい
ホスト	RHEVH サーバーへの接続に使用するデータ・ソースのホスト名または IP アドレス。	はい
ユーザー	RHEVM サーバーに接続するための十分な特権を持っている、データ・ソースのユーザー名。	はい
リモート・トランスポート	ローカル `libvirt` API がリモート `libvirt` API に接続するために使用するプロトコル。デフォルト値は 1 です。有効な値は以下のとおりです。 • 1 = SSH • 2 = TLS • 3 = TCP (暗号化されないため、実動での使用はお勧めしません)	はい

表 170. ハイパーバイザーに接続するための構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
ポート	トランスポート・プロトコルで libvirt API への接続に使用するポート。デフォルト値は 22 です。 **重要 :** このポートが必要になるのは、標準ポート (SSH では 22、TLS では 16514、TCP では 16509) が変更されている場合のみです。	はい
ドメイン	ユーザーが属するドメイン。	はい
接続インスタンス・タイプ	ローカル libvirt API が特権のあるシステム・ドライバーに接続するのか、またはユーザーごとの、特権のないセッション・ドライバーに接続するのかを指示します。デフォルト値は 1 です。有効な値は以下のとおりです。 • 1 = システム • 2 = セッション	はい
RHEVM 接続詳細設定の編集 (Edit RHEVM Connection Details settings)	RHEVM サーバーへの接続用のパラメーターを編集するかどうかを示します。1 (Add) と入力して続行します。デフォルト値は 5 (Next) です。 **重要 :** すべての構成パラメーターの値を指定した後に、パラメーターの編集を続行するかどうかを指示するように求めるプロンプトが再び出されます。5 (Next) と入力します。	はい

MariaDB モニターの構成

MariaDB エージェントが MariaDB サーバーのリソースの可用性とパフォーマンスをモニターするためのデータを収集できるように、このエージェントを構成する必要があります。リモートとローカルの両方のモニターについて MariaDB エージェントの構成に関する次の前提条件を参照してください。

始める前に

ご使用の環境で、MariaDB エージェントのシステム要件が満たされていることを確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the MariaDB エージェント を参照してください。

このタスクについて

MariaDB エージェントは単一インスタンス・エージェントです。エージェントをインストールした後にエージェントを手動で構成する必要があります。エージェントは、Windows オペレーティング・システムと Linux オペレーティング・システムで構成できます。エージェントは、構成のためにインスタンス名と MariaDB サーバーのユーザー資格情報を必要とします。管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は最大で 32 文字です。 指定するインスタンス名は 28 文字 (ホスト名の字数は除く) まで入力できます。例えば、インスタンス名に MariaDB を指定すると、管理対象システム名は MariaDB:hostname:MJ になります。

重要 : 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードは表示されません。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成できます。構成値を更新したら、エージェントを始動し、更新した値を適用します。

手順

Windows オペレーティング・システム上でエージェントを構成するには、以下の手順を実行します。

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで以下の手順を実行します。
   a) 「**Monitoring Agent for MariaDB**」テンプレートをダブルクリックします。
   b) 「**Monitoring Agent for MariaDB**」ウィンドウで、インスタンス名を指定して「**OK**」をクリックします。
3. 「**Monitoring Agent for MariaDB**」ウィンドウで、次の手順を実行します。
   a) 「**IP アドレス**」フィールドに、リモートでモニターする MariaDB サーバーの IP アドレスを入力します。モニター対象のサーバーにエージェントがインストールされている場合は、デフォルト値を保持します。
   b) 「**JDBC ユーザー名**」フィールドに、MariaDB サーバーのユーザー名を入力します。デフォルト値は root です。
   c) 「**JDBC パスワード**」フィールドに JDBC ユーザーのパスワードを入力します。
   d) 「**JDBC パスワードの確認 (Confirm JDBC password)**」フィールドに、再度パスワードを入力します。
   e) 「**JDBC JAR ファイル (JDBC Jar File)**」フィールドで「**参照**」をクリックし、MariaDB コネクターの Java ファイルを含むディレクトリーを探して選択します。
   f) 「**次へ**」をクリックします。
   g) 「**JDBC ポート番号**」フィールドで JDBC サーバーのポート番号を指定します。
   デフォルトのポート番号は 3306 です。
   h) 「**Java トレース・レベル**」リストから Java のトレース・レベルを選択します。
   デフォルト値は `Error` です。
   i) 「**OK**」をクリックします。
   「**IBM Performance Management**」ウィンドウにインスタンスが表示されます。
4. 「**Monitoring Agent for MariaDB**」インスタンスを右クリックし、「**開始**」をクリックします。

要確認 : 再度エージェントを構成するには、「**IBM Performance Management**」ウィンドウで以下の手順を実行します。

a. 構成するエージェント・インスタンスを停止します。
b. 「**Monitoring Agent for MariaDB**」インスタンスを右クリックし、「**再構成**」をクリックします。
c. ステップ 3 と 4 を繰り返します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Linux システムでのエージェントの構成

構成スクリプトを実行し、プロンプトに応答することで、Linux オペレーティング・システム上でエージェントを構成できます。

手順

Linux オペレーティング・システムでエージェントを構成するには、以下の手順を実行します。

1. コマンド行で、次のコマンドを実行します。

   ```
   install_dir/bin/mariadb-agent.sh config instance_name
   ```

   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は MariaDB エージェントのインストール・ディレクトリーです。
2. 以下のパラメーターの値を入力するように求めるプロンプトが表示されたら、**Enter** を押してデフォルト値を受け入れるか、異なる値を指定して **Enter** を押します。
   - IP アドレス
   - JDBC ユーザー名
   - JDBC パスワード
   - JDBC パスワードの再入力
   - JDBC JAR ファイル
   - JDBC ポート番号 (デフォルトのポート番号: 3306)
   - Java トレース・レベル (デフォルト値: `Error`)

   構成パラメーターについて詳しくは、584 ページの『サイレント応答ファイルを使用したエージェントの構成』を参照してください。
3. 以下のコマンドを実行して、エージェントを開始します。

   ```
   install_dir/bin/mariadb-agent.sh start instance_name
   ```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用して、Linux システムと Windows システム上で、MariaDB エージェントを構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

サイレント応答ファイルを使用してエージェントを構成するには、以下の手順を実行します。

要確認 : この手順では、エージェントがインストールされている以下のデフォルト・パスを想定しています。

Windows `C:¥IBM¥APM`

Linux `opt/ibm/apm/agent`

エージェントが異なるパスにインストールされている場合は、手順内のパスを置き換えます。また、サイレント応答ファイル内の **AGENT_HOME** パラメーターを編集して、エージェントがインストールされているパスを指定してください。

1. テキスト・エディターで、次のパスにある応答ファイルを開きます。

   Linux *install_dir*/samples/mariadb_silent_config.txt

   Windows *install_dir*¥samples¥mariadb_silent_config.txt

   ここで、*install_dir* は MariaDB エージェント のインストール・ディレクトリーです。
2. 応答ファイルで、以下のパラメーターの値を指定します。
   - **Server Name** パラメーターに、リモートでモニターする MariaDB サーバーの IP アドレスを指定します。指定しない場合は、デフォルト値の localhost のままにします。
   - **JDBC user name** パラメーターでは、root のデフォルト・ユーザー名の値を保持するか、INFORMATION_SCHEMA テーブルの表示権限を持つユーザーの名前を指定します。
   - **JDBC password** パラメーターに、JDBC ユーザー・パスワードを入力します。
   - **JDBC Jar File** パラメーターでは、Java JAR ファイルの MariaDB コネクターのデフォルト・パスが正しい場合は、そのパスを保持します。そうでない場合は、正しいパスを入力します。コネクターは以下のデフォルト・パスにあります。

     Linux /usr/share/java/mariadb-connector-java.jar

     Windows C:¥Program Files (x86)¥MariaDB¥mariadb-connector-java.jar
   - **JDBC port number** パラメーターでは、デフォルトのポート番号 3306 を保持するか、別のポート番号を指定します。
   - **Java trace level** パラメーターでは、デフォルト値 Error を保持するか、IBM サポートの指示に従って、別のレベルを指定します。
3. 応答ファイルを保存して閉じ、以下のコマンドを実行してエージェント構成設定を更新します。

   Linux *install_dir*/bin/mariadb-agent.sh config *instance_name install_dir*/samples/mariadb_silent_config.txt

   Windows *install_dir*¥BIN¥mariadb-agent.bat config *instance_name install_dir*¥samples¥mariadb_silent_config.txt

   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は MariaDB エージェントのインストール・ディレクトリーです。

   重要 : サイレント応答ファイルの絶対パスを含めるようにしてください。含まれていないと、ダッシュボードにエージェント・データが表示されません。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Microsoft Active Directory モニターの構成

Monitoring Agent for Microsoft Active Directory は、インストールされた後に自動的に構成されて開始されます。

始める前に

ハードウェアおよびソフトウェア前提条件を確認します Software Product Compatibility Reports for Microsoft Active Directory agent を参照してください。

すべての属性のデータをダッシュボードに表示するには、以下のタスクを実行します。

- 492 ページの『管理者ユーザーとしての Microsoft Active Directory エージェントの実行』
- 492 ページの『ローカル環境変数の構成』

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

管理者ユーザーとしての Microsoft Active Directory エージェントの実行

Microsoft Active Directory エージェントをデプロイするには、管理権限が必要です。

このタスクについて

「管理者」グループのメンバーであるユーザーは、すべてのデータ・セットを使用できます。このタスクでは、ユーザーを作成し、管理者権限をこのユーザーに割り当て、エージェントのユーザー・アカウントをこのユーザーに変更します。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**管理ツール**」 > 「**Active Directory ユーザーとコンピューター**」の順にクリックします。
2. ユーザーを作成するドメインを、ドメインの名前の横にある正符号 (+) をクリックして展開します。
3. 「**ユーザー**」を右クリックして、「**新規作成**」 > 「**ユーザー**」をクリックします。
4. 新規ユーザーを作成するため、「**新規オブジェクト - ユーザー**」ウィザードを使用します。
   デフォルトでは、新規ユーザーは Domain Users グループのメンバーです。
5. Domain Users グループに作成された新規ユーザーを右クリックして、更に「**プロパティ**」をクリックします。「**ユーザー名プロパティ (Username Properties)**」ウィンドウが表示されます。username には新規ユーザーの名前が入ります。
6. 「**ユーザー名プロパティ (Username Properties)**」ウィンドウで、以下のステップを実行します。
   a) 「**所属するグループ**」タブをクリックします。「**所属するグループ**」エリアで、「管理者」グループを追加します。
   b) 「**適用**」をクリックしてから「**OK**」をクリックします。
7. 「**スタート**」 > 「**ファイル名を指定して実行**」の順にクリックし、`services.msc` と入力します。
8. 「**サービス**」ウィンドウで、以下のステップを実行します。
   a) 「**Monitoring Agent for Active Directory**」サービスで右クリックして、「**プロパティ**」をクリックします。
   b) 「**Monitoring Agent for Active Directory プロパティ**」ウィンドウの「**ログオン**」タブで、「**次のアカウント**」をクリックします。ユーザーの資格情報を入力します。
   c) 「**適用**」をクリックしてから「**OK**」をクリックします。
9. エージェント・サービスを再始動します。

ローカル環境変数の構成

ダッシュボードに Sysvol 複製データを表示するには、環境変数の値を指定する必要があります。オプションで、キャッシュ間隔の値を更新して、キャッシングの有効/無効を切り替えることもできます。

手順

1. 「**IBM Performance Management**」ウィンドウで、「**アクション**」メニューから「**拡張**」 > 「**ENV ファイルの編集**」をクリックします。
2. K3ZENV ファイルで、次の環境変数の値を変更します。

ADO_CACHE_INTERVAL
キャッシングを開始するか停止するかを決定します。また、キャッシュ間隔の値を設定するために使用されます。キャッシュ間隔は、2 つの連続するデータ収集の間の時間 (秒数) です。キャッシングを開始するには、キャッシュ間隔に正の整数値を指定します。 キャッシングを停止するには、キャッシュ間隔にゼロの値を指定します。キャッシュ間隔値はデフォルトで 1200 に設定されており、キャッシングが開始されます。

ADO_SYSVOL_FORCE_REPLICATION_FLAG
エージェントによって開始される強制複製を有効にするか無効にするかを決定します。この変数のデフォルト値は、TRUE です。強制複製を無効にするには、この変数の値を FALSE に変更してください。

ADO_SYSVOL_REPLICATION_TEST_INTERVAL
2 つの Sysvol 複製テストの間の時間間隔 (分数) を決定します。この変数のデフォルト値は 0 分です。Sysvol 複製テストを実行するには、この変数にゼロより大きい値を指定してください。

ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL
Sysvol 複製テストの完了後、エージェントが Sysvol 複製の結果を検証するために待機する時間の長さを分単位で決定します。

ADO_SYSVOL_REPLICATION_TEST_INTERVAL 変数の値は、**ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL** 変数の値より大きくする必要があります。これらの変数には、以下の値を使用することができます。

ADO_SYSVOL_REPLICATION_TEST_INTERVAL: 1440
ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL: 30

2 つの環境変数に有効な値を割り当てると、Active Directory エージェントは、管理対象システムの Sysvol 共有フォルダーに 1 つのファイルを作成し、強制 Sysvol 複製を初期化します。この強制複製は、管理対象システムから Sysvol 複製パートナーの Sysvol 共有フォルダーまで初期化されます。複製テストの結果の検証後、エージェントは、作成および複製されたファイルを管理対象システムおよび Sysvol 複製パートナーから削除します。

3. オプション: K3ZENV ファイルで、**APM_ATTRIBUTES_ENABLE_COLLECTION** 環境変数を追加し、その値を Yes に設定して、次のデータ・セットのデータを「**属性の詳細**」タブに表示します。
   - サービス
   - 複製
   - ファイル複製サービス
   - 移動または削除した組織単位
   - LDAP
   - セキュリティー・アカウント・マネージャー
   - DFS
   - アドレス帳
   - イベント・ログ
   - パスワード設定オブジェクト

   要確認 : これらのデータ・セットのデータ収集を無効にする場合は、**APM_ATTRIBUTES_ENABLE_COLLECTION** 環境変数の値を No に設定します。
4. Microsoft Active Directory エージェントを再始動します。

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Microsoft Active Directory エージェントの非管理者ユーザーとしての実行

Log File Agent を非管理者ユーザーとして実行できます。

このタスクについて

Active Directory のモニター・エージェントは、非管理者ユーザーとして実行できます。ただし、トラスト・トポロジー属性と Sysvol 複製属性は、使用できない場合があります。これらの属性はドメイン・ユーザーのみが使用できます。

非管理者ユーザーがトラスト・トポロジー属性を表示するには、以下のレジストリー権限が必要です。

- HKEY_LOCAL_MACHINE¥SOFTWARE¥Candle ディレクトリーへの全アクセス権限を付与します。
- HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥WindowsNT¥CurrentVersion¥Perflib ディレクトリーの読み取り権限を付与します。

非管理者ユーザーが Sysvol 複製属性を表示するには、ドメイン内のすべてのドメイン・コントローラーの Sysvol フォルダーへの全アクセス権限を保持していなければなりません。

重要 : Microsoft Active Directory エージェントが非管理者ユーザーとして実行されている場合は、APM ユーザー・インターフェースで、「サービス属性」グループの一部のサービスについて、「現在の状況」属性と「開始タイプ (Start Type)」属性の値が「不明」と表示されます。

以下の表に、ドメイン・ユーザーおよびパフォーマンス・モニター・ユーザーに関するデータを表示する Active Directory エージェントの属性グループを示します。

表 *171.* ドメイン・ユーザーおよびパフォーマンス・モニター・ユーザーの属性グループ

ユーザー権限	属性グループ
ドメイン・ユーザー	• RID プールの情報 • サービス • イベント・ログ • DNS • DNS ADIntegrated の詳細 • DNS ADIntegrated • DHCP • トラスト • グループ・ポリシー・オブジェクト • 脱落および検出オブジェクト • ディレクトリー交換サービス • 複製競合オブジェクト • LDAP 属性 • ルート・ディレクトリー・サーバー • コンテナー • 複製パートナー • ドメイン・コントローラーの可用性 • 複製パートナー待ち時間 • フォレスト・トポロジー

表 171. ドメイン・ユーザーおよびパフォーマンス・モニター・ユーザーの属性グループ (続き)	
ユーザー権限	**属性グループ**
ドメイン・ユーザーおよびパフォーマンス・モニター・ユーザー	ドメイン・ユーザーで示されているすべての属性グループと以下の追加の属性グループ • アドレス帳 • 複製 • ディレクトリー・サービス • 知識整合性チェッカー • Kerberos 鍵配布センター • Lightweight Directory Access Protocol (LDAP) • ローカル・セキュリティー権限 • ネーム・サービス・プロバイダー • セキュリティー・アカウント・マネージャー • ファイル複製サービス • 分散ファイル・システム 複製 • DFS 複製接続 • DFS 複製フォルダー • DFS サービス・ボリューム • ドメイン・コントローラー・パフォーマンス • リモート・アクセス・サーバー • DirectAccess サーバー • Netlogon 属性

注 : さらに、以下の属性グループでは、管理者 グループのメンバーであるユーザーのデータが 表示されます。

- Active Directory データベース情報
- 移動または削除された組織単位
- パスワード設定オブジェクト

詳しくは、491 ページの『Microsoft Active Directory モニターの構成』を参照してください。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**管理ツール**」>「**Active Directory ユーザーとコンピューター**」の順にクリックします。
2. ドメインの名前の横にある正符号 (+) をクリックして、ユーザーを作成するドメインを展開します。
3. 「**ユーザー**」を右クリックして、「**新規作成**」>「**ユーザー**」をクリックします。
4. 「**新規オブジェクト - ユーザー (New Object - User)**」ウィザードを使用して新規ユーザーを作成します。デフォルトでは、新規ユーザーは「**domain users**」グループのメンバーです。
5. 「*domain users*」 グループに作成された新規ユーザーを右クリックして、「**プロパティ**」をクリックします。「**ユーザー名のプロパティ**」ウィンドウ (ユーザー名 は新規ユーザーの名前) が開きます。「**ユーザー名のプロパティ**」ウィンドウで以下のステップを実行します。
   a) 「**所属するグループ (Member of)**」タブをクリックします。「**所属するグループ (Member of)**」領域に「**Performance Monitor Users**」グループを追加します。
   b) 「**適用**」をクリックしてから「**OK**」をクリックします。
6. Candle_Home ディレクトリーに移動します。デフォルトのパスは C:¥IBM¥APM です。

7. 「APM」フォルダーを右クリックして「**プロパティ**」をクリックします。「**APM のプロパティ**」ウィンドウが開きます。「**APM のプロパティ**」ウィンドウで以下のステップを実行します。
   a) 「**セキュリティ**」タブで「**編集**」をクリックします。
   b) 「**追加**」をクリックして新規ユーザーを追加し、そのユーザーに全アクセス権限を付与します。
   c) 「**適用**」をクリックしてから「**OK**」をクリックします。
8. 「**スタート**」>「**ファイル名を指定して実行**」の順にクリックし、services.msc と入力します。「**サービス**」ウィンドウが開きます。「**サービス**」ウィンドウで以下のステップを実行します。
   a) 「**Monitoring Agent for Active Directory (Monitoring Agent for Active Directory)**」サービスを右クリックして「**プロパティ**」をクリックします。
   b) 「**Active Directory のプロパティ**」ウィンドウの「**ログオン**」タブで、「**アカウント**」をクリックします。ユーザーの資格情報を入力します。
   c) 「**適用**」をクリックしてから「**OK**」をクリックします。
9. エージェント・サービスを再始動します。

属性グループ AD_Services_Status のドメイン・グループの構成

Services.properties の MS Active Directory ドメイン・サービスを、サーバー状況を確認する際に使用または除外されるように構成できます。属性グループ AD_Services_Status とそのシチュエーションは、Windows Server 2012 以降で利用可能です。

このタスクについて

Services.properties ファイルには、以下のデフォルトの MS Active Directory ドメイン・サービスとその構成が含まれています。
True は、サーバー状況の値を確認する際にそのサービスが考慮されることを示します。
False は、サーバー状況の値を確認する際にそのサービスが考慮されないことを示します。

表 *172. MS Active Directory* ドメイン・サービスとその構成設定。

MS Active Directory ドメイン・サービス	**デフォルト設定**
DFS レプリケーション	true
リモート・プロシージャー・コール (RPC)	false
DNS クライアント	true
DNS サーバー	true
グループ・ポリシー・クライアント	false
サイト間メッセージ	true
Kerberos 鍵配布センター	true
NetLogon	true
Windows Time	true
DHCP クライアント	false
Active Directory Web サービス	false
Active Directory フェデレーション・サービス	false

注 : Windows Server 2012 以降で属性グループ AD_Services_Status のデータ収集を有効にするには、エージェントの再始動が必要です。

手順

1. エージェントを停止します。
2. 変更がある場合は、Services.properties ファイルを見つけます。
   32 ビット・エージェントの場合、Services.properties ファイルは *CANDLE_HOME*¥TMAITM6¥ にあります。
   64 ビット・エージェントの場合、Services.properties ファイルは *CANDLE_HOME*¥TMAITM6_x64¥ にあります。
   CANDLE_HOME はエージェントのインストール・ディレクトリーです。
3. サーバー状況を確認する際にドメイン・サービスを考慮する必要がある場合は、そのサービスの値を true に設定します。
   サーバー状況を確認する際にドメイン・サービスを除外する必要がある場合は、そのサービスの値を false に設定します。
   ファイルを保存して閉じます。
4. エージェントを開始します。

Microsoft Active Directory エージェントのアップグレード

MS Active Directory エージェントを最新バージョンにアップグレードできます。

始める前に

エージェントがインストールされているマシンで、最新リリースのインストーラーで提供されている installAPMAgents.bat ファイルを使用できるようにします。

このタスクについて

エージェントを最新バージョンにアップグレードするには、以下の手順を実施します。

手順

1. エージェントがインストールされているマシンにログオンします。
2. コマンド・プロンプトを起動し、最新リリースのインストーラーから入手できる installAPMAgents.bat ファイルを実行します。
3. 既存のエージェントがあるインストール・ディレクトリーを入力し、Enter キーを押します。
4. コマンド・プロンプトに、ベース・エージェントのバージョンと、アップグレード先のターゲット・エージェントのバージョンが表示されます。 続行するには、Enter キーを押してください。
5. エージェントのアップグレードに成功すると、アップグレードされたエージェントのバージョンが「**IBM Performance Management**」ウィンドウに表示されます。
6. 「**IBM Performance Management**」ウィンドウでエージェントを右クリックし、ドロップダウン・メニューから「**再構成**」を選択します。
7. アップグレードしたエージェント・バージョンを「**Application Performance ダッシュボード**」に反映するには、APM サーバーにログオンし、以下のコマンドを使用して APM サーバーのコンポーネントを再始動します。
   a. **apm stop_all**
   b. **apm start_all**
8. 「**IBM Performance Management**」ウィンドウでエージェントを右クリックし、ドロップダウン・メニューから「**リサイクル (Recycle)**」を選択します。

タスクの結果

アップグレードしたエージェントが 「**Application Performance ダッシュボード**」に反映されます。

注 : アップグレードしたエージェントが「**Application Performance ダッシュボード**」に表示されるまでには、最大で 30 分以上かかる場合があります。

Microsoft Cluster Server のモニターの構成

エージェントがクラスター・サーバー・データを収集できるように、Monitoring Agent for Microsoft Cluster Server を構成する必要があります。 エージェントを構成するには、サイレント応答ファイルを使用します。

始める前に

以下のタスクを確実に完了してください。

- エージェント用の空のリソース・グループを作成します。
- Windows Server 2008、2012、2016、および 2019 システム上のエージェントのリソース・グループに、汎用サービス・クラスター・リソースを作成します。
- Microsoft Cluster Server 環境またはアプリケーションに接続するユーザーが確実に管理者特権を持つようにします。管理者特権を持つ既存のユーザーを使用するか、新規ユーザーを作成してください。新規ユーザーを管理者グループに追加することにより、新規ユーザーに管理者特権を割り当てます。

  要確認 : Microsoft Cluster Server エージェントの構成には、ローカル・ユーザーまたはドメイン・ユーザーを使用できます (ただし、ユーザーが管理者特権を持っている必要があります)。

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft Cluster Server エージェント を参照してください。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

このタスクについて

Microsoft Cluster Server エージェントは、単一インスタンス・エージェントです。エージェントは、クラスター内の各ノード上で同じ方法で手動でインストールおよび構成する必要があります。エージェントを構成するには、499 ページの『サイレント応答ファイルを使用したエージェントの構成』を参照してください。

Windows Server 2008、2012、2016、および 2019 システムでの汎用サービス・クラスター・リソースの作成

エージェントがクラスター・サーバーをモニターできるように、クラスター・エージェント・サービスをリソースとして追加する必要があります。

始める前に

クラスターの各ノードでエージェントが停止していることを確認してください。

手順

汎用サービス・クラスター・リソースを作成するには、以下の手順を実行します。

1. いずれかのクラスター・ノード上で「**フェールオーバー クラスター マネージャー**」を開きます。
2. 以下のいずれかのステップを実行します。
   - Windows Server 2008 の場合:

     ナビゲーション・ペインで、「**サービスとアプリケーション**」を右クリックし、「**他の操作**」 > 「**空のサービスまたはアプリケーションの作成**」をクリックします。サービスとアプリケーションのリストに新規サービスが表示されます。新しく作成したサービスを名前変更します。
   - Windows Server 2012 の場合:

     ナビゲーション・ペインで、 「**役割**」を右クリックし、「**他の操作**」 > 「**役割の作成 (Create Roles)**」をクリックします。新しいサービスが役割のリストに表示されます。
   - Windows Server 2016 および 2019 の場合:

ナビゲーション・ペインで「**役割**」を右クリックしてから、「**役割の構成 (Configure Roles)**」をクリックします。新しいサービスが表示されます。

3. 新規サービスを右クリックし、「**リソースの追加**」 > 「**汎用サービス**」をクリックします。
4. 「**新しいリソース ウィザード**」ウィンドウで、「**Monitoring Agent for Microsoft Cluster Server**」を選択し、「**次へ**」をクリックします。
5. 「**完了**」ボタンが表示されるまで、後続のウィンドウで「**次へ**」をクリックします。
6. 「**完了**」をクリックします。

   エージェント・サービスがリソースとして追加されます。
7. 「**Monitoring Agent for Microsoft Cluster Server**」リソースを右クリックし、「**このリソースをオンラインにする**」をクリックします。

タスクの結果

希望するノード上でエージェントが開始されます。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、Microsoft Cluster Server エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集し、構成パラメーターに異なる値を指定してエージェントを構成できます。

始める前に

変更する構成パラメーターを含む応答ファイルを作成します。デフォルトの構成パラメーターを変更するには、応答ファイルを編集します。

このタスクについて

サイレント応答ファイルを使用して、エージェントを構成できます。

手順

1. パス *install_dir*¥samples¥microsoft_cluster_server_silent_config.txt にあるサイレント応答ファイルを開きます。
2. **CTIRA_HOSTNAME** 環境変数には、値としてクラスター名を指定します。
3. 各クラスター・ノードで、次のコマンドを実行します。**install_dir¥BIN ¥microsoft_cluster_server-agent.bat config install_dir¥samples ¥microsoft_cluster_server_silent_config.txt**

次のタスク

ローカル・ユーザーからドメイン・ユーザーにユーザー・アカウントを変更します。

ユーザー・アカウントの変更

Microsoft Cluster Server エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。

このタスクについて

デフォルトでは、エージェントはローカル・ユーザー・アカウントで実行されます。エージェントは、クラスター内のすべてのノードを単一ノードからモニターできるように、ドメイン・ユーザーの下で実行する必要があります。

手順

ユーザー・アカウントを変更するには、以下のステップを実行します。

1. 「**IBM Performance Management**」ウィンドウを開きます。
2. エージェントを右クリックし、「**スタートアップの変更**」をクリックします。
3. ドメインのログイン資格情報を入力します。

4. いずれかのノード上で「**フェールオーバー クラスター マネージャー**」を開き、クラスター・サービスを開始します。

タスクの結果
エージェントがそのノードで開始されます。

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Performance Management コンソールの使用法については、977 ページの『Cloud APM コンソールの始動』を参照してください。

Microsoft Exchange のモニターの構成

Exchange Server の可用性およびパフォーマンスをモニターするには、Monitoring Agent for Microsoft Exchange Server を構成する必要があります。

始める前に
エージェントを構成する前に、以下のタスクを必ず完了してください。

- 500 ページの『ユーザーの作成』
- 503 ページの『Exchange Server ユーザーへの管理者権限の割り当て』
- 505 ページの『Exchange Server ユーザーをローカル管理者にする』
- 507 ページの『Exchange Server の到達可能性の構成』
- 508 ページの『ドメイン・ユーザーの下で実行するためのエージェントの構成』
- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft Exchange Server エージェント を参照してください。

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

エージェントのインストール後に、Microsoft Exchange Server エージェントを開始できます。ただし、すべてのエージェント属性のデータを表示するには、手動構成が必要です。

- ローカルにエージェントを構成するには、508 ページの『ローカルでのエージェントの構成』を参照してください。
- サイレント応答ファイルを使用してエージェントを構成するには、512 ページの『サイレント応答ファイルを使用したエージェントの構成』を参照してください。

ユーザーの作成

Exchange Server でエージェントのユーザーを作成するには、手動で作成するか、または「新規ユーザー」ユーティリティーを 実行します。モニターする Exchange Server ごとに、ユーザーを作成する必要があります。

始める前に
Microsoft Exchange Server エージェント をインストールします。ユーザーを作成するには、Microsoft Exchange Server で完全な管理者権限を持つドメイン管理者である必要があります。

このタスクについて

以下のいずれかの手順でユーザーを作成します。

- 501 ページの『Exchange Server 2007 および 2010 でのユーザーの作成』
- 502 ページの『Exchange Server 2013 でのユーザーの作成』
- 502 ページの『新規ユーザー・ユーティリティーの 使用によるユーザーの作成』

Exchange Server 2007 および 2010 でのユーザーの作成

エージェントがモニター対象の Exchange Server と通信し、Exchange Server での認証を行うことができるように、Exchange Server 2007 および 2010 でエージェントのユーザーを作成する必要があります。

手順

ユーザーを作成するには、以下の手順を実行します。

1. 「**スタート**」>「**すべてのプログラム**」>「**Microsoft Exchange Server 2007**」>「**Exchange 管理コンソール**」をクリックします。「**Exchange 管理コンソール**」ウィンドウが開きます。
2. コンソール・ツリーで、「**受信者の構成**」の「**メールボックス**」をクリックします。
3. 操作ウィンドウで「**メールボックスの新規作成**」をクリックします。「メールボックスの新規作成」ウィザードが開きます。
4. **概要**ページで、「**ユーザー メールボックス**」をクリックします。
5. 「**ユーザーの種類**」ページで、「**新しいユーザー**」をクリックします。
6. 「**ユーザー情報**」ページで、以下の情報を指定します。

   組織単位
   デフォルトでは、Active Directory にある「ユーザー」コンテナーが表示されます。 デフォルトの組織単位を変更するには、「**参照**」をクリックします。

   名
   ユーザーの名を入力します。

   イニシャル
   ユーザーのイニシャルを入力します。

   姓
   ユーザーの姓を入力します。

   Name
   このフィールドには、デフォルトで、ユーザーの名、イニシャル、および姓が表示されます。 名前は変更できます。

   ユーザー ログオン名 (ユーザー プリンシパル名)
   ユーザーがメールボックスにログオンする際に使用するユーザー名を入力します。

   ユーザー ログオン名 (Windows 2000 以前)
   Microsoft Windows 2000 サーバー以前と互換性のあるユーザー名を入力します。

   パスワード
   ユーザーがメールボックスにログオンする際に使用するパスワードを入力します。

   パスワードの確認入力
   「**パスワード**」フィールドに入力したパスワードを再入力します。

   ユーザーは次回のログオン時にパスワード変更が必要
   ユーザーのパスワードをリセットしたい場合は、このチェック・ボックスを選択します。
7. 「**メールボックスの設定**」ページで、以下の情報を指定します。

   エイリアス
   デフォルトで、このフィールドの値は、「**ユーザー ログオン名 (ユーザー プリンシパル名)**」フィールドで指定した値と同じです。

   メールボックス データベース
   「**参照**」をクリックして「**メールボックス データベースの選択**」ウィンドウを開きます。使用するメールボックス・データベースを選択して、「**OK**」をクリックします。

管理フォルダ メールボックス ポリシー
メッセージング・レコード管理 (MRM) ポリシーを指定するには、このチェック・ボックスを選択します。「**参照**」をクリックして、このメールボックスに関連付ける MRM メールボックス・ポリシーを選択します。

Exchange ActiveSync メールボックス ポリシー
Exchange ActiveSync メールボックス・ポリシーを指定するには、このチェック・ボックスを選択します。「**参照**」をクリックして、このメールボックスに関連付ける Exchange ActiveSync メールボックス・ポリシーを選択します。

8. 「**メールボックスの新規作成**」ページで、「構成の概要」を確認します。「**新規作成**」をクリックして、メールボックスを作成します。「**完了**」ページの「概要」セクションに、メールボックスが作成されたかどうかが表示されます。
9. 「**完了**」をクリックします。

次のタスク

作成した Exchange ユーザーに管理者権限を割り当てます。

Exchange Server 2013 でのユーザーの作成

エージェントがモニター対象の Exchange Server と通信し、Exchange Server での認証を行うことができるように、Exchange Server 2013 でエージェントのユーザーを作成する必要があります。

手順

Exchange Server 2013 でユーザーを作成するには、以下の手順を実行します。

1. 管理者の資格情報を使用して Exchange 管理センターにログインします。
2. 「**Exchange 管理センター**」ページで、「**受信者 (recipients)**」をクリックしてから「**メールボックス**」をクリックします。
3. 「**メールボックス**」オプションの下にある正符号 (+) の隣の下矢印をクリックして、「**ユーザー メールボックス**」をクリックします。
4. 「ユーザー メールボックスの新規作成」ページで、「**新しいユーザー**」をクリックして、その他のフィールドの値を指定します。
5. 「**保存**」をクリックします。

次のタスク

作成した Exchange ユーザーに管理者権限を割り当てます。

新規ユーザー・ユーティリティーの使用によるユーザーの作成

新規ユーザー・ユーティリティーを 実行して、Exchange Server 2007 以降上にユーザーを作成することができます。このユーティリティーの 実行によって作成されたユーザーは、エージェントを実行するために必要な権限をすべて持っています。このユーティリティーは、エージェントのインストール時にインストールされます。

始める前に

エージェントがインストール済みであることを確認します。新規ユーザー・ユーティリティーを 実行するには、Exchange Server で完全な管理者権限を持つドメイン管理者である必要があります。

このタスクについて

このユーティリティーを 実行すると、以下の権限を持つユーザーが、Active Directory のユーザー・グループ内に作成されます。

- Exchange Server 2007 上:
  - ローカル管理者
  - リモート・デスクトップ・ユーザー
  - Exchange 受信側管理者

- Exchange Server 2010 以降:
  - ローカル管理者
  - リモート・デスクトップ・ユーザー
  - Exchange サーバーまたはパブリック・フォルダーの管理

手順

新規ユーザー・ユーティリティーを 実行するには、以下のステップを実行します。

1. kexnewuser.exe ファイルをダブルクリックします。このファイルは、以下の場所にあります。
   install_dir¥TMAITM6_x64 (*install_dir* はエージェントのインストール先パス)。
2. 「**新規ユーザー**」ウィンドウで、以下のステップを実行します。
   a) ユーザーの**姓**と**名**を入力します。

      制約事項 : 姓名の長さは、28 文字を越えることはできません。

   b) 「**ユーザー・ログオン名**」フィールドに、ユーザーがログイン時に必ず入力する必要がある名前を入力します。

      制約事項 : ユーザー・ログオン名の長さは、256 文字を超えることはできません。

   c) 「**パスワード**」フィールドにパスワードを入力します。
   d) 「**パスワードの確認**」 フィールドに再度パスワードを入力します。
   e) ユーザーに、次回のログオン時に指定したパスワードを再設定するように指示する場合は、「**ユーザーは次回のログオン時にパスワード変更が必要**」 を選択します。
   f) 「**次へ**」をクリックします。

   指定した構成値が検証され、誤った値に対してエラー・メッセージが表示されます。
3. メールボックス・データベースのリストから、必要なメールボックス・データベースを選択し、「**次へ**」をクリックします。
   構成値の要約が表示されます。
4. 「**完了**」をクリックします。

タスクの結果
設定が保存され、ユーザーが作成されます。

Exchange Server ユーザーへの管理者権限の割り当て

Microsoft Exchange Server エージェント用に作成したユーザーは、Microsoft Exchange Server での完全な管理者権限を持つドメイン管理者である必要があります。Microsoft Exchange Server エージェント・コンポーネントにアクセスするには、管理者権限が必要です。

始める前に
モニター対象の Exchange Server にメールボックスを持つ Exchange Server ユーザーを作成します。

このタスクについて
次のいずれかの手順で、ユーザーに管理者権限を割り当てます。

- 504 ページの『Exchange Server 2007 での管理者権限の割り当て』
- 504 ページの『Exchange Server 2010 での管理者権限の割り当て』
- 504 ページの『Exchange Server 2013 での管理者権限の割り当て』
- 505 ページの『Exchange Server 2016 での管理者権限の割り当て』

Exchange Server 2007 での管理者権限の割り当て
Exchange Server 2007 上のユーザーに Exchange 受信者管理者権限を割り当てる必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**Microsoft Exchange Server 2007**」>「**Exchange 管理コンソール**」をクリックします。「**Exchange 管理コンソール**」ウィンドウが開きます。
2. コンソール・ツリーで、「**組織の構成**」をクリックします。
3. 操作ウィンドウで、「**Exchange 管理者の追加**」をクリックします。
4. 「**Exchange 管理者の追加**」ページで、「**参照**」をクリックします。 作成した新規ユーザーを選択して、「**Exchange 受信者管理者**」の役割を選択します。
5. 「**追加**」をクリックします。
6. 「**完了**」ページで、「**完了**」をクリックします。

Exchange Server 2010 での管理者権限の割り当て
Exchange Server 2010 上のユーザーに Exchange サーバーまたはパブリック・フォルダーの管理権限を割り当てる必要があります。

手順

1. 管理者権限を使用して、Exchange サーバーにログオンします。
2. 「**スタート**」>「**管理ツール**」>「**サーバー マネージャ**」の順にクリックします。
3. 「**ツール**」を展開します。
4. 「**Active Directory ユーザーとコンピュータ**」をクリックします。
5. 「**ドメイン**」を展開し、「**Microsoft Exchange セキュリティ グループ**」をクリックします。
6. 「**Exchange サーバー (Exchange Servers)**」または「**パブリック・フォルダーの管理 (Public Folder Management)**」を右クリックし、「**プロパティー**」をクリックします。
7. 「**Exchange サーバーのプロパティー (Exchange Servers Properties)**」または「**パブリック・フォルダーの管理のプロパティー (Public Folder Management Properties)**」ウィンドウで、「**メンバー**」に移動し、「**追加**」をクリックします。
8. ユーザーのリストから、グループに追加するユーザーを選択し、「**OK**」をクリックします。
9. 「**OK**」をクリックします。

Exchange Server 2013 での管理者権限の割り当て
Exchange Server 2013 上のユーザーに Exchange サーバーまたはパブリック・フォルダーの管理権限を割り当てる必要があります。

手順

1. 管理者権限を使用して、Exchange サーバーにログオンします。
2. 「**スタート**」>「**管理ツール**」>「**サーバー マネージャ**」の順にクリックします。
3. 「**ツール**」を展開します。
4. 「**Active Directory ユーザーとコンピュータ**」をクリックします。
5. 「**ドメイン**」を展開し、「**Microsoft Exchange セキュリティ グループ**」をクリックします。
6. 「**Exchange サーバー (Exchange Servers)**」または「**パブリック・フォルダーの管理 (Public Folder Management)**」を右クリックし、「**プロパティー**」をクリックします。
7. 「**Exchange サーバーのプロパティー (Exchange Servers Properties)**」または「**パブリック・フォルダーの管理のプロパティー (Public Folder Management Properties)**」ウィンドウで、「**メンバー**」に移動し、「**追加**」をクリックします。
8. ユーザーのリストから、グループに追加するユーザーを選択し、「**OK**」をクリックします。
9. 「**OK**」をクリックします。

Exchange Server 2016 での管理者権限の割り当て

Exchange Server 2016 上のユーザーに Exchange サーバーまたはパブリック・フォルダーの管理権限を割り当てる必要があります。

手順

1. 管理者権限を使用して、Exchange サーバーにログオンします。
2. 「**スタート**」>「**管理ツール**」>「**サーバー マネージャ**」の順にクリックします。
3. 「**ツール**」を展開します。
4. 「**Active Directory ユーザーとコンピュータ**」をクリックします。
5. 「**ドメイン**」を展開し、「**Microsoft Exchange セキュリティ グループ**」をクリックします。
6. 「**Exchange サーバー (Exchange Servers)**」または「**パブリック・フォルダーの管理 (Public Folder Management)**」を右クリックし、「**プロパティー**」をクリックします。
7. 「**Exchange サーバーのプロパティー (Exchange Servers Properties)**」または「**パブリック・フォルダーの管理のプロパティー (Public Folder Management Properties)**」ウィンドウで、「**メンバー**」に移動し、「**追加**」をクリックします。
8. ユーザーのリストから、グループに追加するユーザーを選択し、「**OK**」をクリックします。
9. 「**OK**」をクリックします。

次のタスク

ユーザーを Exchange Server がインストールされているコンピューターのローカル管理者にします。

Exchange Server ユーザーをローカル管理者にする

Exchange Server データにアクセスするには、Microsoft Exchange Server エージェント用に作成したユーザーが、Exchange Server がインストールされているコンピューターのローカル管理者である必要があります。

始める前に

Exchange Server ユーザーを作成します。

このタスクについて

次のいずれかの手順で、ユーザーをローカル管理者にします。

- 505 ページの『Windows 2003 コンピューター上でユーザーをローカル管理者にする』
- 506 ページの『Windows 2008 コンピューター上でユーザーをローカル管理者にする』
- 506 ページの『Windows 2012 コンピューター上でユーザーをローカル管理者にする』
- 506 ページの『Windows 2016 コンピューター上でユーザーをローカル管理者にする』

Windows 2003 コンピューター上でユーザーをローカル管理者にする

Exchange Server 用に作成したユーザーを、Windows 2003 オペレーティング・システムで稼働していて Exchange Server がインストールされているコンピューターのローカル管理者にする必要があります。

手順

1. コンピューターのデスクトップにある「**マイ コンピュータ**」を右クリックし、「**管理**」をクリックします。
2. 「**ローカル ユーザーとグループ**」を展開します。
3. 「**グループ**」をクリックします。
4. 「**管理者**」をダブルクリックして、「**管理者のプロパティ**」ウィンドウを表示します。
5. 「**追加**」をクリックします。
6. 「**場所**」リストから「**ディレクトリ全体**」を選択します。
7. 作成したユーザーの名前を選択し、「**追加**」をクリックします。
8. 「**OK**」をクリックします。

9.「**OK**」をクリックします。

Windows 2008 コンピューター上でユーザーをローカル管理者にする
Exchange Server 用に作成したユーザーを、Windows Server 2008 オペレーティング・システムで稼働していて Exchange Server がインストールされているコンピューターのローカル管理者にする必要があります。

手順

1.「**スタート**」>「**管理ツール**」>「**サーバー マネージャ**」の順にクリックします。
2. ナビゲーション・ペインで「**構成**」を展開します。
3.「**ローカル ユーザーとグループ**」をダブルクリックします。
4.「**グループ**」をクリックします。
5. ユーザー・アカウントを追加するグループを右クリックして、「**グループに追加**」をクリックします。
6.「**追加**」をクリックして、ユーザー・アカウントの名前を入力します。
7.「**名前の確認**」をクリックしてから「**OK**」をクリックします。

Windows 2012 コンピューター上でユーザーをローカル管理者にする
Exchange Server 用に作成したユーザーを、Windows Server 2012 オペレーティング・システムで稼働していて Exchange Server がインストールされている、コンピューターのローカル管理者にする必要があります。

手順

1.「**スタート**」>「**サーバー マネージャ**」の順にクリックします。
2.「**サーバー マネージャ ダッシュボード**」ページで、「**ツール**」>「**コンピュータの管理**」をクリックします。
3.「**コンピュータの管理**」ページのナビゲーション・ペインで、「**ローカル ユーザーとグループ**」を展開してから、「**ユーザー**」をクリックします。
4. ユーザー・リストから、管理者権限を割り当てるユーザーを右クリックして、「**プロパティ**」をクリックします。
5.「**所属するグループ**」タブをクリックして、「**追加**」をクリックします。
6.「**グループの選択**」ページで、Administrators と入力してから、「**OK**」をクリックします。
7.「**適用**」をクリックして、「**OK**」をクリックします。

Windows 2016 コンピューター上でユーザーをローカル管理者にする
Exchange Server 用に作成したユーザーを、Windows Server 2016 オペレーティング・システムで稼働していて Exchange Server がインストールされている、コンピューターのローカル管理者にする必要があります。

手順

1.「**スタート**」>「**サーバー マネージャ**」の順にクリックします。
2.「**サーバー マネージャ ダッシュボード**」ページで、「**ツール**」>「**コンピュータの管理**」をクリックします。
3.「**コンピュータの管理**」ページのナビゲーション・ペインで、「**ローカル ユーザーとグループ**」を展開してから、「**ユーザー**」をクリックします。
4. ユーザー・リストから、管理者権限を割り当てるユーザーを右クリックして、「**プロパティ**」をクリックします。
5.「**所属するグループ**」タブをクリックして、「**追加**」をクリックします。
6.「**グループの選択**」ページで、Administrators と入力してから、「**OK**」をクリックします。
7.「**適用**」をクリックして、「**OK**」をクリックします。

Exchange Server の到達可能性の構成

到達可能性を検証するために、Microsoft Exchange Server エージェントはサーバーへ E メール・メッセージを送信し、自動応答の受信にかかる時間を測定します。エージェントを開始する前に、E メール・メッセージに自動的に応答するように Exchange Server を構成しておく必要があります。

始める前に

Exchange Server を構成する前に、以下のタスクが完了していることを確認してください。

- モニターする Exchange Server にユーザーのメールボックスが作成されている。
- エージェント用に作成したユーザーがドメイン・ユーザーである。
- ご使用の Microsoft Exchange 組織内のサーバーがサーバー間のメール・フローに合わせて構成されている。

手順

到達可能性を検証する Exchange Server ごとに、以下の手順を実行します。

1. 作成したユーザーの資格情報を指定して、Microsoft Outlook にログインします。
2. **スタートアップ**・ウィンドウで「**次へ**」をクリックします。
3. 「**はい**」を選択し、「**次へ**」をクリックします。
4. 「**Microsoft Exchange Server**」フィールドに、Exchange Server の名前を入力します。
5. 「**メールボックス**」フィールドに、作成したユーザーの名前を入力します。
6. 「**完了**」をクリックします。
7. 「**OK**」をクリックします。
8. 「**ツール**」>「**仕分けルールと通知**」>「**新しい仕分けルール**」とクリックします。
9. 「**新しい仕分けルールを作成**」を選択します。
10. 「**受信メール用に独自の仕分けルールを作成する**」を選択し、「**次へ**」をクリックします。
11. 以下のオプションを選択します。
    - 「**宛先**」**に自分の名前がある場合**
    - 「**件名**」**または本文に特定の文字が含まれる場合**
12. このウィンドウの「**ステップ 2**」の下で「**特定の文字**」をクリックします。
13. 「「**件名**」**または本文に含まれる文字**」フィールドに `AVAILABILITY CHECK` と入力します。
14. 「**追加**」をクリックします。
15. 「**OK**」をクリックしてから、「**次へ**」をクリックします。
16. 「**通知メッセージを使ってサーバーで返信する**」を選択し、「**通知メッセージ**」をクリックします。
17. E メール・メッセージ・エディターで、メッセージの件名フィールドに次のテキストを入力します。
    ```
    CHECK RECEIVED: MAILBOX AVAILABLE.
    ```
18. E メール・メッセージ・エディターを閉じ、「**はい**」をクリックしてこれらの変更を保存します。
19. 「**次へ**」をクリックします。
20. 例外について確認されたときは、制限を指定しないでください。
21. 「**次へ**」をクリックします。
22. 「**完了**」をクリックしてから、「**OK**」をクリックします。

次のタスク

Microsoft Exchange Server エージェントを構成します。

ドメイン・ユーザーの下で実行するためのエージェントの構成

デフォルトでは、Microsoft Exchange Server エージェントは、ローカル・ユーザーの下で実行されるように構成されます。エージェントは、作成したドメイン・ユーザーを使用して実行する必要があります。

始める前に

以下を確認してください。

- 作成したユーザーがローカル管理者権限を持つドメイン・ユーザーである。
- ユーザーが、エージェントがインストールされているサーバーの 管理者権限を持っている。

このタスクについて

エージェントがドメイン・ユーザーの下で実行されると、エージェントは Exchange Server のすべてのコンポーネントをモニターできます。

手順

エージェントを実行するユーザーを変更するには、以下の手順を実行します。

1. 次のコマンドを実行して、エージェントを開始するために使用されるユーザー ID を検証します。
   install_dir¥InstallITM¥KinCinfo.exe -r
2. 管理者グループに属さないユーザー ID を使用して、モニター・エージェントが開始された場合、エージェントは停止します。
3. 「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウを開きます。
4. エージェント・インスタンスを右クリックし、「**スタートアップの変更**」をクリックします。
5. 完全修飾ユーザー ID として <Domain¥Userid> を指定した後、パスワードを指定します。
6. モニター・エージェントを開始します。

ローカルでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、エージェントをローカルで構成することができます。

手順

1. 「**スタート**」> 「**すべてのプログラム**」 >「**IBM Monitoring エージェント (IBM Monitoring agents)**」 >「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Microsoft Exchange Server**」を右クリックしてから、「**エージェントを構成します**」をクリックします。

   重要 : 「**エージェントを構成します**」が無効になっている場合、「**再構成**」をクリックします。
3. 「**Monitoring Agent for Microsoft Exchange Server: エージェント拡張構成**」ウィンドウで、「**OK**」をクリックします。
4. 「**エージェント構成**」ウィンドウで、以下の手順を実行します。
   a) 「**Exchange Server のプロパティー**」タブをクリックして、構成パラメーターの値を指定します。「**OK**」をクリックすると、指定された値が検証されます。
   b) 「**Exchange サービスのモニター**」タブをクリックして、構成パラメーターの値を指定します。「**OK**」をクリックすると、指定された値が検証されます。
   c) 「**拡張構成プロパティー**」タブをクリックして、構成パラメーターの値を指定します。「**OK**」をクリックすると、指定された値が検証されます。

   「**エージェント構成**」ウィンドウの各タブにある構成パラメーターについて詳しくは、以下のトピックを参照してください。

   - 509 ページの『「Exchange Server のプロパティー」タブの構成パラメーター』
   - 510 ページの『Exchange サービスの構成パラメーター』
   - 511 ページの『到達可能性に関する構成パラメーター』

構成値の検証については、512 ページの『構成値の検証』を参照してください。

5. エージェントをリサイクルします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

制約事項 : Cloud APM ダッシュボードでは、Exchange コンポーネント・タイプのいずれか (Microsoft Exchange Server または Microsoft Exchange Server 2013) のみのインスタンスが「マイ・コンポーネント」に表示されます。

「Exchange Server のプロパティー」タブの構成パラメーター

「**エージェント構成**」ウィンドウの「**Exchange Server のプロパティー**」タブでは、サーバー名、ドメイン名、ユーザー名などの Exchange Server のプロパティーを構成することができます。

以下の表に、「**Exchange Server のプロパティー**」タブの構成設定についての詳細な説明を示します。

表 *173.* 「*Exchange Server* プロパティー」タブの構成設定の名前と説明

パラメーター名	説明	必須フィールド	例
Exchange Server 名	Exchange Server の名前。 Exchange Server のインストール時、Exchange Server 名のデフォルトは Windows Server のホスト名です。Exchange Server 名のデフォルトを変更した場合は、Exchange Server エージェントの構成時にその変更した名前を使用する必要があります。 **要確認 :** クラスター環境および分散環境の場合は、Exchange Server 2007 のメールボックス・サーバー名を指定してください。	はい **重要 :** ノード数が 2 より多い単一のコピー・クラスターを持つサーバーにエージェントがインストールされている場合は、値を指定しないでください。	Exchange Server 名が popcorn の場合は、「**Exchange Server 名**」フィールドに popcorn と入力します。
Exchange ドメイン名	Exchange Server がインストールされているドメインの名前です。	はい	Exchange Server が LAB.XYZ.com ドメインにある場合、最初のドットより前の名前、つまり LAB を入力します。
Exchange ユーザー名	Exchange Server のアクセス用に構成されるユーザーの名前。 **要確認 :** ユーザーは同じ Exchange Server 上にメールボックスを配置する必要があります。	はい	
Exchange ユーザー・パスワード	Exchange Server のアクセス用に構成されるユーザーのパスワード。	はい	
パスワードの確認入力	Exchange Server ユーザーに対して指定するのと同じパスワード。	はい	

表 173. 「*Exchange Server* プロパティー」タブの構成設定の名前と説明 *(続き)*			
パラメーター名	**説明**	**必須フィールド**	**例**
Exchange MAPI プロファイル名	MAPI プロファイルは Exchange Server へのアクセスに必要なプライマリー構成設定です。64 ビット版の Microsoft Exchange Server エージェントを使用して Exchange Server 2007 以降をモニターしている場合、 このフィールドは使用できません。	いいえ	
クラスターでの構成	クラスター環境で Microsoft Exchange Server エージェント を構成する場合は、このチェック・ボックスを選択します。	適用外	
クラスター・サーバー名	クラスター・サーバーの名前。 このフィールドは、「**クラスターでの構成**」チェック・ボックスを選択すると、使用可能になります。	フィールドが使用可能な場合、はい。	SCCCluster
Exchange サブシステム ID	Cluster Server ノードの名前 このフィールドは、「**クラスターでの構成**」チェック・ボックスを選択すると、使用可能になります。	フィールドが使用可能な場合、はい。	node1
Exchange エージェントのヒストリカル・データ・ディレクトリー	ヒストリカル・データが保管されるディスク上のロケーション。 このフィールドは、「**クラスターでの構成**」チェック・ボックスを選択すると、使用可能になります。	フィールドが使用可能な場合、はい。	c:¥history

Exchange サービスの構成パラメーター

「**エージェント構成**」ウィンドウの「**Exchange サービスのモニター**」タブで Exchange サービスを選択すると、Exchange Server の状況を把握することができます。

以下の表に、「**Exchange サービスのモニター**」タブの構成設定についての詳細な説明を示します。

表 174. 「*Exchange* サービスのモニター」タブの構成設定の名前と説明		
パラメーター名	**説明**	**必須フィールド**
Exchange サービス	使用可能なサービス・リストで Exchange サービスを選択して矢印をクリックし、選択したサービスを「**サーバー状況用に構成されたサービス**」リストに移動し、Microsoft Exchange Server エージェントが当該サービスをモニターできるようにします。 **要確認 :** 使用可能なサービスのリストは、インストールされている Exchange Server のバージョンと役割によって異なります。	適用外
サーバー状況用に構成されたサービス	このリストで既に使用可能になっているサービスにより、Exchange Server の状況が決定されます。 該当するサービスは必須であり、「**サーバー状況用に構成されたサービス**」リストから「**Exchange サービス**」リストに移動することはできません。「**Exchange サービス**」リストからサービスを移動することで、さらにサービスを「**サーバー状況用に構成されたサービス**」リストに追加できます。追加したサービスは、「**Exchange サービス**」リストに戻すことができます。	適用外

到達可能性に関する構成パラメーター

「**エージェント構成**」ウィンドウの「**拡張構成プロパティー**」タブでは、ターゲット E メール・アドレス、到達可能性の間隔など、到達可能性に関連したパラメーターを構成することができます。

以下の表に、「**拡張構成プロパティー**」タブの構成設定についての詳細な説明を示します。

表 *175.* 「拡張構成プロパティー」タブの構成設定の名前と説明

パラメーター名	説明	必須フィールド
メールボックス到達可能性のモニターを使用可能にする	エージェントで到達可能性メトリック・データを取得する場合は、このチェック・ボックスを選択します。	適用外
ターゲット E メール・アドレス	到達可能性を確認するための E メール・アドレス。複数の E メール・アドレスはセミコロン (;) で区切ります。 **制約事項 :** このフィールドの 合計文字数は 1023 以下にする必要があります。	このフィールドが使用可能な場合、はい。
E メール送信間隔 (秒)	Exchange Server エージェントが E メールを送信するまでの待機時間 (秒)。	このフィールドが使用可能な場合、はい。
E メール送信タイムアウト (秒)	メールボックス・サーバーが到達可能かどうかをテストするために送信された E メールに対する応答をエージェントが待機する間隔 (秒)。	いいえ
メールボックスの詳細のモニターを使用可能にする	メールボックスの詳細メトリックスに関するデータを収集するには、このチェック・ボックスを選択します。	適用外
メールボックスの詳細の収集開始時刻	メールボックスの詳細メトリックが収集される時刻 (hh:mm:ss 形式)。	いいえ
メールボックスの詳細の収集間隔 (秒)	メールボックスの詳細メトリックの収集間隔 (秒単位)。	いいえ
イベント・ログ収集時間 (分)	エージェントがイベント・レコードを収集する期間 (分単位)。	いいえ
イベントの最大数	収集されるイベント・レコードの最大数。イベント・レコードの収集は、収集されたイベント・レコードの数が最大数を超えると停止します。	いいえ
収集間隔 (秒)	エージェントのサイクルの 間隔 (秒単位)。	いいえ
Exchange トポロジー間隔 (秒)	トポロジー詳細情報の収集の間隔 (秒単位)。	いいえ
メッセージ追跡の収集間隔 (時間)	メッセージ追跡ログが収集される間隔 (時間単位)。 **制約事項 :** 間隔の値は 1 から 12 までの範囲内にする必要があります。12 より大きい間隔の値を指定すると、値は 12 として保存されます。英字または特殊文字を含む無効値を入力すると、値は 0 として保存されます。0 という値は、メッセージ追跡の収集が無効になっていることを示します。 以下の条件のいずれかが true の場合は、このフィールドは使用できません。 • メールボックス・サーバーの役割またはハブ・トランスポートの役割が Exchange Server にインストールされていない。 • メッセージ追跡の機能が Exchange Server で使用不可になっている。	いいえ

構成値の検証
エージェントの構成中に指定した値は検証されます。検証により、すべての必須パラメーターに対して値が指定されていることと、ユーザーのローカル管理者権限などの特定の条件が満たされていることが保証されます。

以下の表に、指定された構成値に対して実行される妥当性テストを示します。

表 *176.* 妥当性テスト

妥当性テスト	**検証の内容**
Exchange Server 名	ユーザーのメールボックス・サーバー名が、指定された Exchange Server 名に一致する。
Exchange Server 権限	必要な Exchange Server 権限をユーザーが持っている。Exchange Server 2007 では、ユーザーに受信者管理者権限が必要です。Exchange Server 2010 以降では、ユーザーに受信者管理者権限が必要です。
ローカル管理者	ユーザーがローカル管理者権限を持っている。
エージェント・サービス・ログオン	指定したユーザー・アカウントで実行されるようにエージェント・サービスが構成されている。

1 つ以上の妥当性テストが失敗した場合は、エラー・メッセージが生成されます。すべての必須パラメーターに値を指定する必要があります。そうしないと、構成値を保存できません。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集し、構成パラメーターに異なる値を指定してエージェントを構成できます。

このタスクについて
サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. *install_dir*¥samples にある msex_silent_config.txt ファイルを開き、すべての必須パラメーターの値を指定します。
   他のパラメーターのデフォルト値を変更することもできます。
2. 次のコマンドを実行します。
   install_dir¥BIN¥msexch-agent.bat config install_dir¥samples ¥msex_silent_config.txt

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

制約事項 : Cloud APM ダッシュボードでは、Exchange コンポーネント・タイプのいずれか (Microsoft Exchange Server または Microsoft Exchange Server 2013) のみのインスタンスが「マイ・コンポーネント」に表示されます。

エージェント用のローカル環境変数の構成

Microsoft Exchange Server エージェント用のローカル環境変数を構成すると、重複するイベントのイベント・スロットルを有効または無効にすることができます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**アクション**」メニューから「**拡張**」>「**ENV ファイルの編集**」をクリックします。
3. KEXENV ファイルで、次の環境変数の値を変更します。

   EX_EVENT_THROTTLE_ENABLE
   この変数を使用すると、重複するイベントをスロットルできます。デフォルト値は False です。重複するイベントに対するシチュエーションがトリガーされないようにイベント・スロットルを有効にするには、この変数の値を True に設定します。

   EX_EVENT_THROTTLE_DURATION
   この変数は、イベントをスロットルする時間を分単位で指定します。デフォルト値は 0 分です。

Microsoft Hyper-V のモニターの構成

Monitoring Agent for Microsoft Hyper-V Server をインストールすると、エージェントはデフォルトの構成設定で自動的に構成され、起動されます。デフォルトの構成設定を変更するには、サイレント応答ファイルを使用します。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft Hyper-V Server エージェント を参照してください。
- 変更する構成パラメーターを含む応答ファイルを作成します。
- 「仮想マシン」ページに仮想マシンのデータを表示するには、 統合コンポーネントおよび OS エージェントを各仮想マシンにインストールする必要があります。Linux システムで稼働する仮想マシンの場合は、以下のタスクを必ず完了してください。
  - Linux システムをアップグレードします。
  - 更新した hypervkvpd パッケージまたは hyperv-daemons rpm パッケージを仮想マシンにインストールします。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

重要 : Performance Management 8.1.3 リリースでは、エージェント構成ウィンドウは不要であるため、削除されています。エージェント構成ウィンドウは、Performance Management バージョン 8.1.2 以前で使用可能です。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

手順

エージェントを構成するには、以下のステップを実行します。

1. *install_dir*¥samples にある microsoft_hyper-v_server_silent_config.txt ファイルを開き、すべての必須パラメーターの値を指定します。

   他のパラメーターのデフォルト値を変更することもできます。
2. コマンド・プロンプトを開き、以下のコマンドを入力します。

   install_dir¥BIN¥microsoft_hyper-v_server-agent.bat config install_dir ¥samples¥microsoft_hyper-v_server_silent_config.txt

   応答ファイルには、以下のパラメーターがあります。

   - KHV_DIRECTOR_PORT
   - KHV_DIRECTOR_SERVER

   要確認 : エージェントの構成は以下のグループに編成されています。

   IBM Systems Director 構成 (IBM_DIRECTOR_CONFIGURATION)
   このグループで定義される構成要素は、常にエージェントの構成に存在します。このグループでは、エージェント全体に適用される情報を定義します。

   IBM Systems Director Server のポート番号 (KHV_DIRECTOR_PORT)
   IBM Systems Director Server のポート番号。デフォルト値は「なし」です。

   IBM Systems Director Server のホスト名 (KHV_DIRECTOR_SERVER)
   環境を管理している IBM Systems Director Server のホスト名または IP アドレス。デフォルト値は「なし」です。
3. エージェントが停止状態である場合は、エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Windows 上の Monitoring Agent for Microsoft Hyper-V Server を非管理者ユーザーによって実行するためのローカル・セキュリティー・ポリシーの指定

ローカル・セキュリティー・ポリシーを使用して、Windows 上の Monitoring Agent for Microsoft Hyper-V Server を非管理者ユーザーによって実行できるようにすることができます。

このタスクについて

以下の 2 つのローカル・セキュリティー・ポリシーを組み合わせることで、Windows 上の Microsoft Hyper-V Server エージェントを非管理者ユーザーによって実行できるようにすることができます。Microsoft Hyper-V Server エージェントの開始、停止、構成、およびデータ検証を実行するには、以下の 2 つのポリシーを使用します。

- プログラムのデバッグ
- サービスとしてログオン (Log on as Service)

また、以下の属性グループは APM ポータルでデータを取得するために管理者権限を必要とします。

- 可用性
- マイグレーション
- クラスターなしの VM マイグレーション
- VM ストレージ・マイグレーション

これらのローカル・セキュリティー許可を非管理者ユーザーに付与するには、以下の手順を実行します。

手順

1. ローカル管理者として、Microsoft Hyper-V Server エージェントをインストールします。
2. 非管理者ユーザーを install_dir ディレクトリーの下に追加し、以下の許可を付与します。

a) HKEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring レジストリーへの全アクセス権限を付与します。
b) HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows NT¥CurrentVersion¥Perflib レジストリー内の非管理者ユーザーに読み取り権限を付与します。
c) install_dir ディレクトリー内の非管理者ユーザーに全アクセス権限を付与します。

3. 「**スタート**」メニューに移動し、**secpol.msc** コマンドを実行して「ローカル セキュリティ ポリシー」を開きます。
4. 非管理者ユーザーをポリシーに追加するには、515 ページの『「ローカル・セキュリティー・ポリシー」許可の付与』を参照してください。
5. 非管理者ユーザーを Hyper-V 管理者ユーザー・グループに追加するには、516 ページの『Hyper-V 管理者ユーザー・グループへの非管理者ユーザーの追加』を参照してください。
6. 非管理者ユーザーを Performance Monitor Users グループに追加するには、517 ページの『Performance Business Monitor Users グループへの非管理者ユーザーの追加』を参照してください。
7. 非管理者ユーザーの DCOM セキュリティー許可を変更するには、516 ページの『DCOM 許可の変更』を参照してください。
8. Microsoft Hyper-V Server エージェントを再始動し、APM ポータルでデータを確認します。

「ローカル・セキュリティー・ポリシー」許可の付与

Microsoft Hyper-V Server エージェントの開始、停止、構成、およびデータ検証を実行するには、「プログラムのデバッグ」と「サービスとしてログオン (Log on as Service)」の 2 つのローカル・セキュリティー・ポリシーに対する許可を付与する必要があります。

「プログラムのデバッグ」許可の付与

このタスクについて

「プログラムのデバッグ」許可を付与するには、以下の手順を実行します。

手順

1. 「**スタート**」>「**コントロール パネル**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。 ポリシーのリストが開きます。
3. 「**プログラムのデバッグ**」ポリシーをダブルクリックします。 「**プログラムのデバッグのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**適用**」をクリックしてから「**OK**」をクリックします。

「サービスとしてログオン (Log on as Service)」許可の付与

このタスクについて

「サービスとしてログオン (Log-on as Service)」許可を付与するには、以下の手順を実行します。

手順

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。 ポリシーのリストが開きます。

3.「**サービスとしてログオン (Log-on as service)**」ポリシーをダブルクリックします。「**サービスとしてログオンのプロパティ**」ウィンドウが開きます。
4.「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5.「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6.「**適用**」をクリックしてから「**OK**」をクリックします。

DCOM 許可の変更

非管理者ユーザーのアクセス権で Microsoft Hyper-V Server エージェントを実行するには、DCOM 許可を変更する必要があります。

このタスクについて

DCOM 許可を変更するには、ユーザーが DCOM サーバーを始動するための適切な権限を持っていることを確認します。許可を変更するには、以下の手順を実行します。

手順

1. **Regedit** コマンドを使用して、HKCR¥Clsid¥*clsid value レジストリー値に移動します。

   注 : 非管理者ユーザーを使用してエージェントを構成する場合、イベント ID 10016 とともに CLSID 値がイベント・ビューアーに表示されます。
2.「レジストリ エディタ」ペインで、「**既定**」をダブルクリックします。
3.「**文字列の編集**」ダイアログ・ボックスで、値のデータ・ストリングをコピーします。
4.「**スタート**」>「**コントロール パネル**」>「**管理ツール**」>「**コンポーネント サービス**」をクリックします。
5.「**コンポーネント サービス**」ウィンドウで、「**コンポーネント サービス**」>「**コンピューター**」>「**マイコンピューター**」を展開し、「**DCOM**」をダブルクリックします。
6.「DCOM の構成」ペインで、コピーしたストリング (プログラム名) を見つけ、プログラム名を右クリックし、「**プロパティ**」をクリックします。
7.「**プロパティ**」ウィンドウで、「**セキュリティ**」タブを選択します。
8.「**起動とアクティブ化のアクセス許可**」グループ・ボックスの下で、「**カスタマイズ**」を選択してから「**編集**」をクリックします。「**起動とアクティブ化のアクセス許可**」ウィンドウが開きます。
9.「**追加**」をクリックして、許可リストに非管理者ユーザーを入力し、「**OK**」をクリックします。
10.「ローカルからの起動」および「ローカルからのアクティブ化」の「**許可**」チェック・ボックスを選択し、「**OK**」をクリックします。

Hyper-V 管理者ユーザー・グループへの非管理者ユーザーの追加

APM ポータルでデータを取得するには、非管理者ユーザーを Hyper-V 管理者ユーザー・グループに追加する必要があります。

このタスクについて

非管理者ユーザーを Hyper-V 管理者ユーザー・グループに追加するには、以下の手順を実行します。

手順

1.「**スタート**」>「**コントロール パネル**」>「**管理ツール**」>「**コンピューターの管理**」をクリックします。「**コンピューターの管理**」ウィンドウが開きます。
2.「**システム ツール**」>「**ローカル ユーザーとグループ**」>「**グループ**」をクリックします。グループのリストが開きます。
3.「**Hyper-V Administrators**」グループをダブルクリックします。「**Hyper-V Administrators のプロパティ**」ウィンドウが開きます。

4. 「**追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**適用**」をクリックしてから「**OK**」をクリックします。

Performance Business Monitor Users グループへの非管理者ユーザーの追加

APM ポータルでデータを取得するには、非管理者ユーザーを Performance Monitor Users グループに追加する必要があります。

このタスクについて

非管理者ユーザーを Performance Business Monitor Users グループに追加するには、以下の手順を実行します。

手順

1. 「**スタート**」>「**コントロール パネル**」>「**管理ツール**」>「**コンピューターの管理**」をクリックします。「**コンピューターの管理**」ウィンドウが開きます。
2. 「**システム ツール**」>「**ローカル ユーザーとグループ**」>「**グループ**」をクリックします。グループのリストが開きます。
3. 「**Performance Monitor Users**」グループをダブルクリックします。「**Performance Business Monitor Users のプロパティー (Performance Business Monitor Users Properties)**」ウィンドウが開きます。
4. 「**追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**適用**」をクリックしてから「**OK**」をクリックします。

Microsoft IIS モニターの構成

Monitoring Agent for Microsoft Internet Information Services をインストールすると、エージェントはデフォルトの構成設定で自動的に構成され、起動します。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft IIS エージェント を参照してください。
- Microsoft Internet Information Server 環境またはアプリケーションに接続するユーザーが確実に管理者特権を持つようにします。管理者特権を持つ既存のユーザーを使用するか、新規ユーザーを作成してください。新規ユーザーを管理者グループに追加することにより、新規ユーザーに管理者特権を割り当てます。

  要確認 : Microsoft IIS エージェントの構成には、ローカル・ユーザーまたはドメイン・ユーザーを使用できます (ただし、ユーザーが管理者特権を持っている必要があります)。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

エージェントを構成するには、「**IBM Performance Management**」ウィンドウを使用するか、またはサイレント応答ファイルを使用することができます。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

次のタスク

エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。ユーザー・アカウントの変更手順については、519 ページの『ユーザー・アカウントの変更』を参照してください。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、Microsoft IIS エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

Microsoft IIS エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Microsoft Internet Information Services**」を右クリックし、「**再構成**」をクリックします。
3. 「Monitoring Agent for Microsoft Internet Information Services」ウィンドウで、以下のステップを実行します。
   a) 「**HTTP エラー・ログの構成**」タブで、ログ・ファイルの保存先を指定して、「**次へ**」をクリックします。

      注 : デフォルトでは、このログ・ファイルが `C:¥WINDOWS¥system32¥LogFiles¥HTTPERR` に保存されます。管理者は、このログ・ファイルの場所を変更できます。

   b) 「**サイトのログの構成**」タブで、ログ・ファイルの保存先を指定して、「**OK**」をクリックします。

      注 : デフォルトでは、このログ・ファイルが `C:¥inetpub¥logs¥LogFiles` に保存されます。管理者は、このログ・ファイルの場所を変更できます。
4. 「**Monitoring Agent for Microsoft IIS の再始動 (Restart of Monitoring Agent for Microsoft IIS)**」ウィンドウで、「**はい**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

Microsoft IIS エージェント をインストールすると、エージェントはデフォルトの構成設定で自動的に構成され、起動します。デフォルトの構成設定を変更するには、サイレント応答ファイルを使用します。

始める前に

変更する構成パラメーターを含む応答ファイルを作成します。デフォルトの構成パラメーターを変更するには、応答ファイルを編集します。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

手順

Microsoft IIS エージェントを構成するには、以下のステップを実行します。

1. コマンド・ラインで、パスを、`msiis-agent.bat` ファイルが格納されているディレクトリーに変更します。
2. **`msiis-agent.bat config`** *absolute path to the response file* というコマンドを入力します。

   応答ファイルには、以下のパラメーターがあります。

   KQ7_SITE_LOG_FILE
   C:¥inetpub¥logs¥LogFiles

   KQ7_HTTP_ERROR_LOG_FILE
   C:¥WINDOWS¥system32¥LogFiles¥HTTPERR

   要確認 : エージェントの構成は以下のグループに編成されています。

   サイトのログの構成 (SITE_LOG)
   このグループには、サイト・ログ・ファイル (KQ7_SITE_LOG_FILE) に関連した構成パラメーターが含まれます。管理者はログ・ファイルの保存先を指定することができます。デフォルトでは、このログ・ファイルが `C:¥inetpub¥logs¥LogFiles` に保存されます。

   HTTP エラー・ログの構成 (HTTP_ERROR_LOG)
   このグループには、HTTP エラー・ログ・ファイル (KQ7_HTTP_ERROR_LOG_FILE) に関連した構成パラメーターが含まれます。管理者はログ・ファイルの保存先を指定することができます。デフォルトでは、このログ・ファイルが `C:¥WINDOWS¥system32¥LogFiles¥HTTPERR` に保存されます。
3. エージェントが停止状態の場合は、エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

ユーザー・アカウントの変更

Microsoft IIS エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。

このタスクについて

デフォルトでは、Microsoft IIS エージェント はローカル・ユーザー・アカウントで実行されます。

手順

1. 次のコマンドを実行して、エージェントを開始するために使用されるユーザー ID を検証します。

   `install_dir¥InstallITM¥KinCinfo.exe -r`
2. 管理者グループに属さないユーザー ID を使用して、モニター・エージェントが開始された場合、エージェントは停止します。
3. 「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウを開きます。
4. エージェント・インスタンスを右クリックし、「**スタートアップの変更**」をクリックします。
5. 完全修飾ユーザー ID として <Domain¥User ID> を指定した後、パスワードを指定します。
6. Microsoft IIS エージェントを始動します。

Skype for Business Server (以前の Microsoft Lync Server) のモニターの構成

Monitoring Agent for Skype for Business Server (以前の MS Lync Server) をインストールすると、エージェントは未構成の状態になります。エージェントを始動するには、そのエージェントを構成する必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Skype for Business Server エージェント を参照してください。
- Skype for Business Server エージェントの実行に使用するユーザーが、管理者特権を持つドメイン・ユーザーであり、かつ Lync Server トポロジーまたは Skype for Business Server トポロジーにリストされているすべてのリモート・サーバーへのアクセス権限を持っているようにする必要があります。管理者特権を持つ既存のドメイン・ユーザーを使用するか、新しいドメイン・ユーザーを作成してその新しいドメイン・ユーザーに管理者特権を割り当ててください。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

エージェントを構成するには、「**IBM Performance Management**」ウィンドウを使用するか、またはサイレント応答ファイルを使用することができます。

次のタスク

エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。ユーザー・アカウントの変更手順については、523 ページの『ユーザー・アカウントの変更』を参照してください。

非管理者ユーザーの権限およびアクセス権

非管理者ユーザーとして Skype for Business Server エージェント のモニター・エージェントを実行することはできますが、一部の機能にはアクセスできません。

レジストリー権限

非管理者ユーザーを作成するには、新規ユーザー (非管理者ユーザー) を作成して、以下のようにその新規ユーザーのレジストリー権限をセットアップします。

- KEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring への全アクセス権限
- CANDLE_HOME ディレクトリーへの 全アクセス権限

非管理者ユーザーは、Performance Monitor Users および Performance Log Users のメンバーである必要があります。これらの権限を非管理者ユーザーに定義した場合、Perfmon ベースの属性グループすべてのデータが表示されます。

データベースから収集した属性グループのデータを表示するには

データベースから収集された属性グループのデータを表示する場合は、非管理者ユーザーに対して以下の権限をセットアップする必要があります。

- Skype for Business Server エージェントを実行するために使用する非管理者ユーザー・アカウントが、任意のプロセスにデバッガーを追加するためのデバッグ・プログラム権限を持っている必要があります。

デフォルトでは、デバッグ・プログラム権限は管理者およびローカル・システムのアカウントにのみ割り当てられます。デバッグ・プログラム権限を付与するには、Lync Server または Skype for Business Server で以下の手順を実行する必要があります。

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。ユーザー権限のリストが開きます。
3. 「**プログラムのデバッグ**」ポリシーをダブルクリックします。「**プログラムのデバッグのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「選択するオブジェクト名を入力してください」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**OK**」をクリックします。

- サービスとしてログオン権限を付与

  サービスとしてログオン権限を付与するには、Lync Server または Skype for Business Server で以下の手順を実行する必要があります。

  1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
  2. 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。ユーザー権限のリストが開きます。
  3. 「**サービスとしてログオン**」ポリシーをダブルクリックします。 「**サービスとしてログオンのプロパティ**」ウィンドウが開きます。
  4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーまたはグループの選択**」ウィンドウが開きます。
  5. 「選択するオブジェクト名を入力してください」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
  6. 「**OK**」をクリックします。

可用性属性グループに、管理者グループのメンバーであるユーザーのデータが 表示されます。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、Skype for Business Server エージェント (以前の MS Lync Server エージェント) を Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

Skype for Business Server エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Skype for Business Server**」を右クリックしてから「**エージェントを構成します**」をクリックします。
3. 「Monitoring Agent for Skype for Business Server」ウィンドウで、以下のステップを実行します。

a)「**Skype for Business トポロジーの SQL 構成**」タブで、Microsoft Lync Server または Skype for Business Server 中央管理ストアに接続するために、構成パラメーターの値を指定し、「**次へ**」をクリックします。

注 : 「Skype for Business トポロジーの SQL 構成」は IBM Cloud Application Performance Management には適用されないため、このタブをスキップできます。

重要 : シンセティック・トランザクション構成はオプションです。シンセティック・トランザクション・データが必要な場合は、「**セットアップ情報 (Setup Information)**」タブおよび「**スケジューラー構成 (Scheduler Configuration)**」タブで構成パラメーターを指定します。

b)「**管理者のログイン資格情報 (Administrator Login Credentials)**」タブで、管理者の資格情報を指定して、「**次へ**」をクリックします。

c)「**セットアップ情報 (Setup Information)**」タブで、シンセティック・トランザクションのコマンドを実行するために、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

d)「**スケジューラー構成 (Scheduler Configuration)**」タブで、シンセティック・トランザクションをスケジュールするために、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

e)「**Skype for Business モニター役割のための SQL Server 構成**」タブで、Microsoft Lync Server または Skype for Business Server モニター役割に接続するために、構成パラメーターの値を指定し、「**次へ**」をクリックします。

構成パラメーターについては、523 ページの『エージェントの構成パラメーター』を参照してください。

4.「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Skype for Business Server**」を右クリックし、「**始動**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

Skype for Business Server エージェント (以前の MS Lync Server エージェント) をインストールすると、エージェントは未構成の状態になります。エージェントを始動するには、そのエージェントを構成する必要があります。
サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

始める前に

変更する構成パラメーターを含む応答ファイルを作成します。デフォルトの構成パラメーターを変更するには、応答ファイルを編集します。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

手順

Skype for Business Server エージェントを構成するには、以下のステップを実行します。

1. コマンド・プロンプトを開きます。
2. `skype_for_business_server-agent.bat` ファイルが格納されているディレクトリーへのパスを変更します。

3. **>skype_for_business_server-agent.bat config** *absolute path to the response file* というコマンドを入力します。

   構成パラメーターについては、523 ページの『エージェントの構成パラメーター』を参照してください。
4. エージェントが停止状態である場合は、エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

ユーザー・アカウントの変更

Skype for Business Server エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。

このタスクについて

デフォルトでは、Skype for Business Server エージェント はローカル・ユーザー・アカウントで実行されます。エージェントをドメイン・ユーザーの下で実行すると、エージェントがリモート・サーバーからデータを収集することができます。

手順

1. 次のコマンドを実行して、エージェントを開始するために使用されるユーザー ID を検証します。

   install_dir¥InstallITM¥KinCinfo.exe -r
2. 管理者グループに属さないユーザー ID を使用して、モニター・エージェントが開始された場合、エージェントは停止します。
3. 「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウを開きます。
4. エージェント・インスタンスを右クリックし、「**スタートアップの変更**」をクリックします。
5. 完全修飾ユーザー ID として <Domain¥User ID> を指定した後、パスワードを指定します。
6. Skype for Business Server エージェントを始動します。

エージェントの構成パラメーター

Skype for Business Server エージェント (以前の MS Lync Server エージェント) を構成する際に、データベース・サーバー名、データベース・インスタンス名、データベース名、その他のパラメーターなど、構成パラメーターのデフォルト値を変更できます。

以下の表に、Skype for Business Server エージェントの構成パラメーターの説明を示します。

注 : 次の表で、すべてのフィールドの中で「プールの FQDN」フィールドは必須です。

表 *177.* エージェントの構成パラメーターの名前と説明

パラメーター名	説明
データベース・サーバー名 (例: PS6877)	• 「**Skype for Business トポロジーの SQL 構成**」タブ: Lync または Skype for Business Server 中央管理ストアがインストールされているデータベース・サーバーの名前。 • 「**Skype for Business モニター役割のための SQL Server 構成**」タブ: モニター役割がインストールされているデータベース・サーバーの名前。

表 177. エージェントの構成パラメーターの名前と説明 (続き)	
パラメーター名	**説明**
データベース・インスタンス名	• 「**Skype for Business トポロジーの SQL 構成**」タブ: デフォルトのインスタンス。 • 「**Skype for Business モニター役割のための SQL Server 構成**」タブ: モニター役割がインストールされているデータベース・インスタンスの名前。
データベース名	データベースの名前。
データベース・ユーザー ID	データベースのユーザー ID。このユーザーは、必要な Microsoft SQL Server インスタンスにアクセスできなければなりません。このユーザーは Active Directory ユーザーでなくても構いません。
データベースのパスワード	モニター役割がインストールされているデータベースのパスワード。
ユーザー名 (例: skype¥administrator)	管理者のユーザー ID。このユーザーは、管理者特権と、Lync Server トポロジーまたは Skype for Business Server トポロジーにリストされているすべてのリモート・サーバーへのアクセス権限を持つドメイン・ユーザーでなければなりません。このユーザーの資格情報は、シンセティック・トランザクション機能でも使用されます。そのため、このユーザーには、タスク・スケジューラーで Windows スケジュールを作成する権限とシンセティック・トランザクション・コマンドを実行する権限が必要です。
パスワード	管理者のログイン・パスワード。
ドメイン・パスワードの確認 (Confirm Domain Password)	「ドメイン・パスワード」フィールドに指定したものと同じパスワードを入力します。
プールの完全修飾ドメイン名 (Pool FQDN)	シンセティック・コマンドが実行される Skype プールの完全修飾ドメイン名 (FQDN)。
地理的ロケーション	実動システムの地理的ロケーション。
テスト・ユーザー 1 (Test Users1) (例えば、user1@skype.com)	シンセティック・トランザクション・コマンドレットの実行中に使用できる最初のユーザー名。ユーザー名の形式は、SAMAccountName@domain.com です。 SIP アドレスを指定しないでください。
テスト・ユーザー 1 のパスワード (Test User1 PWD)	テスト・ユーザー 1 のパスワード。
テスト・ユーザー 1 のパスワードの確認 (Confirm Test User1 PWD)	**「テスト・ユーザー 1 のパスワード」**フィールドに指定したのと同じパスワードを入力します。
テスト・ユーザー 2 (Test User2) (例えば、user2@skype.com)	シンセティック・トランザクション・コマンドレットの実行中に使用できる 2 番目のユーザー名。ユーザー名の形式は、SAMAccountName@domain.com です。 SIP アドレスを指定しないでください。
テスト・ユーザー 2 のパスワード (Test User2 PWD)	テスト・ユーザー 2 のパスワード。
テスト・ユーザー 2 のパスワードの確認 (Confirm Test User2 PWD)	**「テスト・ユーザー 2 のパスワード (Test User2 PWD)」**フィールドに指定したのと同じパスワードを入力します。

表 177. エージェントの構成パラメーターの名前と説明 (続き)	
パラメーター名	**説明**
エージェント構成値の使用 (Use Agent Configuration Values)	構成パネルで指定したすべてのフィールドを使用してシンセティック・コマンドを実行する場合は、このフィールドを有効なままにします。 New-CsHealthMonitoringConfiguration で設定された値を使用する場合は、無効にします。無効にすると、**「プールの FQDN」** の値が Get-CsHealthMonitoringConfiguration の ID として使用されます。 **Test-CsMcxP2PIM** コマンドを実行するには、必ず有効なテスト・ユーザーの資格情報を指定してください。
頻度	シンセティック・トランザクションのデータをフェッチするスケジュールされたユーティリティーの 頻度。頻度には以下の値を指定できます。 • 日次 (DAY_FREQUENCY) • 週次 (WEEK_FREQUENCY) • 月次 (MONTHLY_FREQUENCY)
収集 (時) (Collection Hour)	ユーティリティーをスケジュールするために 選択する 24 時間クロック形式でのタイム・スタンプの時の部分。
収集 (分) (Collection Minute)	ユーティリティーをスケジュールするために 選択するタイム・スタンプの分の部分。
開始日 (YYYY-MM-DD)	スケジューラーがアクティブ化される時刻。
終了日 (YYYY-MM-DD)	スケジューラーが非アクティブ化される時刻。

Microsoft .NET のモニターの構成

Monitoring Agent for Microsoft .NET は、.NET アプリケーションをモニターします。エージェントは、インストールが完了すると、自動的にリソース・モニター・データの収集を開始します。ただし、トランザクション・トラッキング・データおよび診断データを収集するには、いくつかの構成タスクを実行する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。最新のシステム要件情報については、Microsoft .NET エージェントの Software Product Compatibility Reports (SPCR) を参照してください。

このタスクについて

エージェントがインストールされたら、エージェントがトランザクション・トラッキング・データおよび診断データを収集できるように、次の構成タスクを実行します。

1. データ・コレクターの登録
   データ・コレクターは、Microsoft .NET エージェントのコンポーネントです。これは、トランザクション・トラッキング・データおよび診断データを収集し、そのデータを Microsoft .NET エージェントに渡します。このデータを収集するには、データ・コレクターを登録する必要があります。詳しくは、527 ページの『データ・コレクターの登録』を参照してください。
2. トランザクション・トラッキングおよび診断データの収集の構成
   データ・コレクターを登録した後、Cloud APM コンソールでトランザクション・トラッキング・データおよび診断データの収集を有効にします。**configdc** コマンドを使用して、診断データ収集を有効にすることもできます。詳しくは、530 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』および530 ページの『configdc コマンドを使用した診断データの収集の有効化』を参照してください。
3. 構成更新のアクティブ化

configdc コマンドを使用して、診断データ収集を有効にする場合、更新が構成ファイルに保存されるように、構成をアクティブにする必要があります。構成変更のアクティブ化について詳しくは、532 ページの『構成更新のアクティブ化』を参照してください。

4. データ・コレクターのパフォーマンスの 調整
   データ・コレクターのパフォーマンスを微調整するためのタスクをいくつか実行する必要が生じることがあります。詳しくは、533 ページの『データ・コレクターのパフォーマンス調整』を参照してください。

エージェントの共存

エージェントの共存環境では、Cloud APM コンソールまたは Tivoli® Enterprise Portal からトランザクション・トラッキング・データを表示できます。エージェント共存環境におけるトランザクション・トラッキング用のデータの使用可能化については、531 ページの『エージェントの共存環境におけるトランザクション・トラッキングの有効化』を参照してください。

製品のバージョンとエージェントのバージョンが異なっていることがよくあります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン』を参照してください。以前のエージェント・リリースに関する資料にアクセスするには、次の表を参照してください。

表 *178.* エージェントのバージョンおよび資料

Microsoft .NET エージェントのバージョン	**資料**
8.1.3.2	IBM Cloud Application Performance Management
8.1.3 および 8.1.2	IBM Performance Management 8.1.3

リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。

ローカル・アカウントまたはドメイン・アカウントを使用してエージェントを実行するための権限

Microsoft .NET エージェントを実行する権限を持つのは、管理者グループのメンバーであるローカル・ユーザーまたはドメイン・ユーザーのみです。このトピックでは、ローカル・ユーザーまたはドメイン・ユーザーが管理者グループのメンバーでない 場合に満たす必要がある条件について説明します。

ユーザーに、システム・ドライブおよびエージェントのインストール先ドライブに対する以下の権限が必要

1. 読み取り
2. 書き込み
3. 実行
4. 変更

ユーザーに、HKEY_LOCAL_MACHINE レジストリー・キーに対する以下の権限が必要

- 読み取り

ユーザーが、モニター対象サーバーの以下のグループのメンバーでなければならない

1. ユーザー
2. IIS_IUSRS
3. Performance Monitor Users
4. Performance Log Users

注 : ただし、Microsoft .NET エージェントを実行するときは、管理者グループのメンバーになっているローカル・ユーザーまたはドメイン・ユーザーを使用することをお勧めします。

データ・コレクターの登録

データ・コレクターを登録し、トランザクション・トラッキングおよび診断データを収集する必要があります。リソース・モニター・データを収集するための固有の構成は不要です。

このタスクについて

データ・コレクターで収集するトランザクション・データまたは診断データ、あるいは両タイプのデータに応じて、データ・コレクターの以下のコンポーネントを登録します。

表 *179.* データ・コレクターのコンポーネントとその機能

コンポーネント名	モニター
httpmodule	ASP.NET トランザクション。要求の応答時間と CPU 時間を収集します
profiler	ADO.NET トランザクション。診断用のメソッド、スタック・トレース、および要求コンテキストのデータを収集します
isapi	ASP.NET トランザクション。要求の応答時間と CPU 時間を収集します
soap	ASMX または WCF サービス・トランザクション、および WCF サービス応答時間

要確認 :

- 64 ビットの Microsoft IIS サーバーで 32 ビットのアプリケーションをフィルターに 掛ける場合は、`isapi32` を使用します。
- すべてのトランザクションを追跡し、完全なトランザクション・トポロジーを表示するには、すべてのコンポーネントを登録します。

手順

1. エージェントがインストールされているサーバーで、管理者として以下のコマンドを実行します。

   ```
   cd install_dir¥qe¥bin
   configdc.exe registerdc [all|isapi|isapi32|profiler|httpmodule|soap]
   ```

   要確認 :

   - 登録するコンポーネントを一切指定せずに **configdc.exe registerdc** コマンドを実行したときは、httpmodule のみが登録されます。
   - すべてのコンポーネントを登録するには、**configdc.exe registerdc all** コマンドを実行します。
   - すべてのコンポーネントを一緒に登録するには、コマンド **configdc.exe registerdc** *component_name component_name* を実行します。例: **configdc.exe registerdc** httpmodule profiler
2. .NET アプリケーションを再始動します。

次のタスク

データ・コレクターを登録した後に、トランザクション・トラッキングおよび診断用のデータ収集を有効にする必要があります。データ・コレクターの有効化については、530 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』を参照してください。

.NET アプリケーションのモニターを停止する場合、データ・コレクターを登録抹消します。データ・コレクターのすべてのコンポーネントを登録抹消するには、**configdc.exe unregisterdc** コマンドを使用して、示されたステップを繰り返します。

.Net エージェントの IIS 応答時間モジュールの使用

8.1.4.0.2 リリース以降では、.NET エージェントに「IIS 応答時間モジュール」が組み込まれています。このモジュールは、応答時間エージェントと連動して、IIS サーバーのエンド・ユーザー・トランザクション・データを表示します。

応答時間モジュールの有効化

応答時間モジュールを使用する前に、有効にする必要があります。

手順

応答時間モジュールを有効にするには、以下のステップを実行します。

1. 管理者モードで、コマンド・プロンプトを開きます。
2. IIS を停止するには、以下のコマンドを実行します。

   ```
   iisreset /stop
   ```

3. コマンド・プロンプトで、install_dir¥qe¥bin ディレクトリーに移動します。
4. IIS の応答時間モジュールを登録するには、以下のコマンドを実行します。

   ```
   configdc registerdc rtmodule
   ```

5. IIS を開始するには、以下のコマンドを実行します。

   ```
   iisreset /start
   ```

タスクの結果

応答時間モジュールが有効になります。

.NET エージェントの IIS 応答時間モジュールと連動させるための応答時間エージェントの構成

応答時間エージェントが .NET エージェントの IIS 応答時間モジュールと連動するように構成する必要があります。

始める前に

応答時間エージェント (8.1.4 バージョン) をインストールします。詳しくは、123 ページの『第 6 章 エージェントのインストール』を参照してください。

手順

以下のステップを実行して、応答時間エージェントが .NET エージェントの IIS 応答時間モジュールと連動するように構成します。

1. 管理者モードでテキスト・エディターを開きます。
2. テキスト・エディターで以下のファイルを開きます。

   config_dir¥TMAITM6_x64¥*host_name*_T5.config

   ここで、*config_dir* は APM ホームであり、*host_name* はサーバーの名前です。
3. 以下のプロパティーを更新します。

   ```
   { KT5DISABLEANALYZER=YES } { KT5ENABLEWEBPLUGIN=YES }
   ```

4. SECTION=analyzerconfig [ ] セクションに、次のプロパティーを追加します。

   ```
   {KT5WEBPLUGINIPCNAME=KFC1}
   ```

5. 応答時間エージェントを再始動します。
6. Performance Management コンソールにログインして、ダッシュボードにエージェントにより収集されるデータを確認します。Performance Management コンソールの使用法については、977 ページの『Cloud APM コンソールの始動』を参照してください。

IIS 応答時間モジュールに対する JavaScript インジェクションの構成

.NET エージェントの Internet Information Services (IIS) 応答時間モジュールと連携するように JavaScript (JS) インジェクションを構成する必要があります。

手順

.NET エージェントの IIS 応答時間モジュールと連携するように JS インジェクションを構成するには、以下の手順を実行します。

1. 管理者モードでテキスト・エディターを開きます。
2. テキスト・エディターで以下のファイルを開きます。

   <APM_HOME>¥qe¥config¥dotNetDcConfig.properties.inactive
3. 応答時間モジュールに対する JS インジェクションを有効にするには、**RTModule.JSInjection.Enabled** プロパティーを **true** に設定します。
4. 応答時間モジュールに対する JS インジェクションを無効にするには、**RTModule.JSInjection.Enabled** プロパティーを **false** に設定します。
5. 管理者モードでコマンド・プロンプトを開き、<APM_HOME>¥qe¥bin ディレクトリーに移動します。
6. 以下のコマンドを実行してください。
   - configdc activateconfig
   - iisreset

IIS 応答時間モジュールの無効化

IIS サーバーのエンド・ユーザー・トランザクション・データを表示しない場合は、IIS 応答時間モジュールを無効にできます。

手順

IIS 応答時間モジュールを無効にするには、以下のステップを実行します。

1. 管理者モードで、コマンド・プロンプトを開きます。
2. IIS を停止するには、以下のコマンドを実行します。

   iisreset /stop
3. コマンド・プロンプトで、*install_dir*¥qe¥bin ディレクトリーに移動します。
4. IIS の応答時間モジュールを登録抹消するには、以下のステップを実行します。
   - IIS の応答時間モジュールを登録抹消するには、以下のコマンドを実行します。

     configdc unregisterdc rtmodule
   - 応答時間モジュールを含めて、データ・コレクターのすべてのコンポーネントを登録抹消するには、以下のコマンドを実行します。

     configdc unregisterdc all
5. IIS を開始するには、以下のコマンドを実行します。

   iisreset /start

タスクの結果

IIS 応答時間モジュールは無効です。

IIS 応答時間モジュールの制限

IIS 応答時間モジュールの制限を以下にリストします。

- ユーザー情報は、IIS 応答時間モジュールによってトラッキングされず、現在、ユーザー名は「不明」と表示されます。

トランザクション・トラッキング・データおよび診断データの収集の有効化

「**エージェント構成**」ページで、トランザクション・トラッキング・データまたは診断データの収集を有効化または無効化できます。

始める前に

データ・コレクターが登録されていることを確認します。詳しくは、527 ページの『データ・コレクターの登録』を参照してください。

手順

各管理対象システムでデータ収集を構成するには、以下のステップを実行します。

1. Cloud APM コンソール にログインします。
2. ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
3. 「**MS .NET**」タブをクリックします。
4. データ収集を構成する管理対象システムのチェック・ボックスを選択し、「**アクション**」リストから以下のいずれかのアクションを実行します。
   - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。選択した各管理対象システムの「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
   - 診断データの収集を有効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」を選択し、設定するレベルをクリックします。選択した各管理対象システムの「**診断モード**」列の状況が更新されて、指定したレベルが表示されます。
     - レベル 1: HTTP モジュールにより、要求の要約および要求インスタンスのデータが収集されます。
     - レベル 2: HTTP モジュールにより、要求の要約および要求インスタンスのデータが収集されます。プロファイラーによって、メソッド・データおよびスタック・トレース・データの収集が行われます。
   - トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。選択した各管理対象システムの「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。
   - 診断データの収集を無効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**無効**」をクリックします。選択した各管理対象システムの「**診断モード**」列の状況が「無効」に更新されます。

タスクの結果

各管理対象システムに対してデータ収集が構成されました。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるトランザクション・トラッキングおよび診断データをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

configdc コマンドを使用した診断データの収集の有効化

configdc コマンドを使用して診断データの収集を有効または無効にすることもできます。このプロセスはオプションです。

始める前に

- データ・コレクターが登録されていることを確認します。詳しくは、527 ページの『データ・コレクターの登録』を参照してください。
- 530 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』の処理を完了していることを確認します。

- qe_custom.properties ファイルが <APM_Home>¥localconfig¥qe の APM サーバーによって処理され、以下のプロパティーを持っていることを確認します。
  - transaction_tracking=ENABLED
  - diagnostic_mode=LEVEL2

手順

1. 次のコマンドを実行します。

   ```
   cd install_dir¥qe¥bin configdc deepdivedc -tracelevel trace_level
   ```

   各部の意味は次のとおりです。

   install_dir
   Microsoft .NET エージェント のインストール・ディレクトリー。

   trace_level
   .NET データ・コレクターが収集する診断データの量を示すトレース・レベル。以下のいずれかの値を指定します。

   0
   診断データの収集が無効になります。

   1
   診断データの収集が有効になります。HTTP モジュールにより、要求の要約および要求インスタンスのデータが収集されます。

   2
   診断データの収集が有効になります。HTTP モジュールにより、要求の要約および要求インスタンスのデータが収集されます。プロファイラーによって、メソッド・データおよびスタック・トレース・データの収集が行われます。

   ヒント:configdc.exe deepdivedc -tracelevel コマンドを使用してトレース・レベルを設定すると、dotNetDcConfig.properties ファイルの bci_dc.diagnose.level パラメーターの値が設定されます。
2. 構成変更をアクティブにします。
   変更のアクティブ化については、532 ページの『構成更新のアクティブ化』を参照してください。

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集される診断データをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェントの共存環境におけるトランザクション・トラッキングの有効化

エージェントの共存環境では、トランザクション・トラッキング・データを収集して、IBM Tivoli Monitoring のコンポーネントである Tivoli Enterprise Portal に渡すようにデータ・コレクターを構成できます。

始める前に
Cloud APM の一部として提供されている Microsoft .NET エージェントをインストールする必要があり、また ITCAM for Microsoft Applications で提供されている .NET データ・コレクター・コンポーネントを削除または登録抹消する必要があります。データ・コレクターのすべてのモジュールを登録抹消するには、**configdc.exe unregisterdc** コマンドを使用します。

手順

トランザクション・トラッキング・データを収集して Tivoli Enterprise Portal に渡すようにデータ・コレクターを構成するには、以下の手順に従います。

1. *install_dir*¥localconfig¥qe ディレクトリーに移動します。ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。デフォルトのパスは、C:¥IBM¥APM です。

2. qe_default.properties ファイルを開き、**transaction_tracking** パラメーターの値を ENABLED に設定します。
3. qe_default.properties ファイルを保存して閉じます。
4. *install_dir*¥qe¥config ディレクトリーに移動します。
5. テキスト・エディターで dotNetDcConfig.properties.inactive ファイルを開きます。
6. **TTDC.enabled** および **TTAS.enabled** の各パラメーターを以下のように設定します。

```
TTDC.enabled=true
TTAS.enabled=true
```

7. トランザクション・コレクターへの接続を構成するために、**TTAS.Host** パラメーターおよび **TTAS.Port** パラメーターの値を、トランザクション・コレクターの IP アドレスおよびポート番号に設定します。
8. 以下のコマンドを実行して変更内容をアクティブ化します。

```
install_dir¥qe¥bin¥configdc.exe activateconfig
```

9. .NET アプリケーションを再始動して、変更を有効にします。

タスクの結果
これで、トランザクション・トラッキング・データを収集して Tivoli Enterprise Portal で表示できるようになりました。

次のタスク

.NET データ・コレクターのトランザクション・トラッキングを無効にするには、この手順を繰り返し、以下の構成値を使用します。

- qe_default.properties ファイルでは transaction_tracking=DISABLED と設定します。
- dotNetDcConfig.properties.inactive ファイルでは TTDC.enabled=false および TTAS.enabled=false と設定します。

構成更新のアクティブ化

configdc コマンドを使用して、構成設定で行った更新をアクティブにする必要があります。アクティブ化により、更新が dotNetConfig.properties ファイルに確実に保存されます。

このタスクについて
configdc コマンドを使用して構成設定を更新した場合、dotNetConfig.properties ファイルでパラメーター値が更新されます。 ただし、このファイルが使用中で変更できない場合、構成設定の更新は dotNetDcConfig.properties.inactive ファイルに保存されます。更新が dotNetConfig.properties ファイルに保存されるように、構成をアクティブ化する必要があります。

手順

1. 以下のパスに移動します。
   install_dir¥qe¥bin
   ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。
2. 次のコマンドを実行します。
   configdc activateConfig

次のタスク

インターネット・インフォメーション・サービス (IIS) のトランザクションがモニターされていて、データ・コレクターの構成が更新された場合は、IIS を再始動して構成をアクティブにします。

ASMX または WCF Web サービスがモニターされていて、データ・コレクターの構成が更新された場合は、Web サービスをホストするプロセスを再開します。

データ・コレクターのパフォーマンス調整

トランザクション・トラッキングおよび診断データを収集するようにデータ・コレクターを構成すると、データ・コレクターのパフォーマンスが影響を受けます。パフォーマンスを向上させるために、パフォーマンス調整タスクを実行できます。

以下のタスクを実行してデータ・コレクターのパフォーマンスを向上させる必要が生じることがあります。

- モニターする ADO.NET インターフェースをフィルターに 掛けます。
- トランザクション・トラッキングおよび診断データをサンプリングします。
- トレース・ロギングを構成します。

モニター用の ADO.NET インターフェースの指定

トランザクション・トラッキング用に有効化する ADO.NET クライアント・インターフェースを 指定することができます。

始める前に

.NET データ・コレクターでサポートされる ADO.NET インターフェースを 表示する場合は、『.NET データ・コレクターでサポートされる名前空間の関数』を参照してください。

.NET データ・コレクターの構成値を表示するには、*install_dir*¥qe¥config ディレクトリーにある dotNetDcConfig.properties ファイルを参照してください。ここで、*install_dir* は Microsoft .NET エージェントのインストール・ディレクトリーです。

このタスクについて

デフォルトでは、エージェントのインストール時に、トランザクション・トラッキングでサポートされるすべての ADO.NET インターフェースのが 有効になります。データ・コレクターでモニターする必要があるインターフェースを 指定するために、特定のインターフェースのモニターを 有効または無効にします。

インターフェースのモニターを 無効にしても、関連付けられているアプリケーション・ドメイン・フィルターの設定は、データ・コレクターの構成ファイル内に残ります。インターフェースを 再び有効にすると、フィルターが保持されます。

手順

- ADO.NET インターフェースのモニターを 有効にするには、以下のステップを実行します。
  a) *install_dir*¥qe¥bin ディレクトリーから、以下のコマンドを実行します。

```
configdc enableMonitor all | adsi | db2 | ldap | odbc | oledb | oracle | sql
| http | web
[-appdomain appdomain filter list]
```

  b) 構成変更をアクティブにします。
  変更のアクティブ化については、532 ページの『構成更新のアクティブ化』を参照してください。
- ADO.NET インターフェースのモニターを 無効にするには、以下のステップを実行します。
  a) *install_dir*¥qe¥bin ディレクトリーから、以下のコマンドを実行します。

```
configdc disableMonitor all | adsi | db2 | ldap | odbc | oledb | oracle | sql
| http | web
```

  b) 構成変更をアクティブにします。
  変更のアクティブ化については、532 ページの『構成更新のアクティブ化』を参照してください。

トランザクション・トラッキング・データおよび診断データのサンプリング

トランザクション・トラッキング・データまたは診断データの収集によってシステム・パフォーマンスが影響を受ける場合は、パフォーマンスを向上させるために、収集データのサンプリングを有効にすることができます。

このタスクについて

トランザクション・トラッキング・データおよび診断データの収集が原因でシステム・パフォーマンスに悪影響が及ぶ場合は、サンプリングによって定期的にデータを収集するようにデータ・コレクターを構成することができます。サンプリングが有効になっている場合、データ・コレクターはすべての要求でデータを収集するのではなく、いくつかの要求ごとにデータを収集します。サンプリング・レートは、DotNetProfilerService プロセスの CPU 使用率に応じて動的に変更できます。

注意 : ただし、サンプリングによりシステム ・リソースは節約できるものの、サンプリングされたデータでは問題を診断するのに非効率な可能性があります。データ・サンプリングを有効にすると、トランザクション・トラッキング・トポロジーが破損したり失われたりするおそれがあります。したがって、データ・サンプリングは、パフォーマンスへの影響が深刻な場合にのみ有効にしてください。

手順

トランザクション・トラッキング・データおよび診断データの収集でサンプリングを有効にするには、以下の手順を実行します。

1. 以下のディレクトリーに移動します。
   install_dir¥qe¥config
   ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。
2. テキスト・エディターで、dotNetDcConfig.properties.inactive ファイルを開きます。
3. ファイル内で以下のパラメーターを設定します。

   bci_dc.sampling.Enabled
   データ・コレクターでトランザクション・トラッキング・データおよび診断データを定期的に収集するかどうかを指定します。有効な値は、true および false です。

   bci_dc.sampling.base
   データ・サンプリングのベースを 指定します。有効な値は正数です。例えば、**bci_dc.sampling.base** パラメーターの値を 10 に設定すると、要求 10 件ごとにデータ・コレクターがトランザクション・トラッキング・データおよび診断データを収集します。サンプリング・レートは、10 個の要求につき 1 つ (10%) です。つまり、データ・コレクターは、1 番目、11 番目、21 番目、31 番目 (以下、10 個ごと) の要求のデータを収集します。

   bci_dc.dynamic.sampling
   サンプリング・レートが定数か動的かを指定します。有効な値は、on および off です。**bci_dc.dynamic.sampling** パラメーターの値を on に設定すると、サンプリング・レートが **bci_dc.dynamic.max_cpu_usage** パラメーターの値に従って動的に調整されます。

   bci_dc.dynamic.max_cpu_usage
   DotNetProfilerService プロセスの CPU 使用率のしきい値を指定します。DotNetProfilerService プロセスの CPU の使用率が指定値の 110% を超えると、サンプリング・レートは低下します。CPU 使用率が指定値の 90% より減少すると、サンプリング・レートは増加します。 有効な値の範囲は、1 から 100 です。
4. dotNetDcConfig.properties.inactive ファイルを保存して閉じます。
5. 以下のコマンドを実行して変更内容をアクティブ化します。

   ```
   install_dir¥qe¥bin¥configdc.exe activateconfig
   ```
6. .NET アプリケーションを再始動して、変更を有効にします。

データ・コレクターのトレース・ロギングの有効化

データ・コレクターのトレース・ログの生成を有効にできます。これらのトレース・ログを使用して、トランザクション・トラッキング・データおよび診断データの収集で発生する可能性のある問題をトラブルシューティングできます。

このタスクについて

ASP.NET トランザクション、ADO.NET トランザクション、および診断データのログを収集するには、データ・コレクターの httpmodule、プロファイラー、および isapi コンポーネントのトレース・ログを有効にします。ASMX トランザクションおよび WCF トランザクションのログを収集するには、データ・コレクターの SOAP コンポーネントのトレース・ログを有効にします。

重要 : トレース・ロギングが有効になっていると、データ・コレクターのパフォーマンスに影響する場合があります。このため、必要なトレース・ログが収集されたら、トレース・ロギングを無効にしてください。

手順

1. エージェントがインストールされているサーバーで、次のパスに移動します。
   install_dir¥qe¥bin

   ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。
2. 有効にするトレース・ログに応じて、次の手順のいずれか、または両方を実行します。
   - httpmodule、プロファイラー、soap コンポーネント、および応答時間モジュールのトレース・ログを有効にするには、次の手順を実行します。
     a. 次のコマンドを実行します。
        configdc logging -tracing on
     b. IIS および .NET アプリケーションを再始動します。
   - BCI エンジンのトレース・ログを有効にするには、次の手順を実行します。
     a. 以下のパスにナビゲートします。
        <APM_HOME>¥qe¥config
     b. テキスト・エディターで、dotNetDcConfig.properties.inactive ファイルを開きます。
     c. プロパティー **bci_dc.trace.logging** の値に on を指定します。
     d. 次のコマンドを実行します。
        configdc activateconfig
     e. IIS を再始動します。

次のタスク

トレース・ログを無効にするには、次のようにします。

- httpmodule、プロファイラー、soap コンポーネント、および応答時間モジュールのトレース・ログを無効にするには、次のようにします。
  - 次のコマンドを実行します。
    configdc logging -tracing off
  - IIS および .NET アプリケーションを再始動します。
- BCI エンジンのトレース・ログを無効にするには、次のようにします。
  - 以下のパスに移動します。
    <APM_HOME>¥qe¥config
  - テキスト・エディターで、dotNetDcConfig.properties.inactive ファイルを開きます。
  - **bci_dc.trace.logging** プロパティーの値に off を指定します。
  - 次のコマンドを実行します。
    configdc activateconfig
  - IIS および .NET アプリケーションを再始動します。

Microsoft Office 365 のモニターの構成

組織の Microsoft Office 365 サブスクリプションの可用性およびパフォーマンスをモニターするには、Microsoft Office 365 エージェントを構成する必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。
- Office 365 ユーザーのデータを収集するには、エージェントがインストールされている Windows クライアントに、次のモジュールがインストールされていることが必要です。
  - PowerShell 3.0 以降
  - Microsoft Online Services サインイン アシスタント PowerShell
  - SharePoint Online 管理シェル
  - .Net Framework バージョン 4.5.2 以降

  エージェントを構成するユーザーは、PowerShell のリモート実行ポリシーを有効にする特権に加えて、管理特権を持っていることが必要です。
- Skype シンセティック・トランザクションをモニターするには、次のタスクを実行します。
  - Skype のシンセティック・トランザクションをユーザーが実行する Windows クライアントに、Skype 2013 クライアントをインストールします。
  - Lync と Skype に対して、仮想オーディオ・ビデオ・フィルターとしてデフォルトのビデオ・デバイスを設定します。
- Microsoft Office 365 を開始するユーザーに管理者特権が付与されていることを確認してください。管理者特権を持つ既存のユーザーを使用するか、新規ユーザーを作成してください。新規ユーザーを管理者グループに追加することにより、新規ユーザーに管理者特権を割り当てます。

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft Office 365 エージェント を参照してください。

このタスクについて

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『Microsoft Office 365 エージェント』を参照してください。V1.0.0 の資料にアクセスするには、IBM Cloud Application Performance Management の Knowledge Center を参照してください。

エージェントのインストール後に、Microsoft Office 365 エージェントを開始できます。ただし、すべてのエージェント属性のデータを表示するには、手動構成が必要です。

エージェントを構成するには、「IBM Cloud Application Performance Management」ウィンドウを使用するか、またはサイレント応答ファイルを使用することができます。

構成されているユーザーの到達可能性の検証

到達可能性を検証するために、Microsoft Office 365 エージェントは、構成されているユーザーに E メール・メッセージを送信し、自動応答の受信にかかる時間を測定します。エージェントを開始する前に、Office 365 エージェントのメールボックス到達可能性設定で構成されているすべてのユーザーを、E メール・メッセージに自動的に応答するように構成しておく必要があります。

始める前に

Exchange Online ユーザーの到達可能性を構成する前に、次のタスクが完了していることを確認してください。

- モニターする Exchange Online のユーザーごとに、メールボックスが作成されていること。
- エージェント用に作成したユーザーが Office 365 のグローバル・ユーザーであること。

手順

到達可能性を検証する対象の Exchange Online ユーザー・アカウントごとに、次の手順を実行します。

1. 作成したユーザーの資格情報を指定して、Microsoft Outlook にログインします。
2. 「**ツール**」>「**仕分けルールと通知**」>「**新しい仕分けルール**」とクリックします。
3. 「**自動仕分けウィザード**」ウィンドウの「**新しい仕分けルールを作成**」で「**受信メッセージにルールを適用する**」をクリックし、「**次へ**」をクリックします。
4. 以下のオプションを選択します。
   - **名前/パブリック グループ**
   - **件名に特定の文字が含まれる場合**
5. ウィンドウの「**ステップ 2**」で「**名前/パブリック グループ**」をクリックします。
6. 「**電子メール アドレスの選択**」ウィンドウで、メッセージの受信元のユーザー (グローバル管理者) を選択し、「**次へ**」をクリックします。
7. このウィンドウの「**ステップ 2**」の下で「**特定の文字**」をクリックします。
8. 「**「件名」または本文に含まれる文字**」フィールドに `Test Reachability` と入力します。
9. 「**追加**」をクリックします。
10. 「**OK**」をクリックしてから、「**次へ**」をクリックします。
11. 「**通知メッセージを使ってサーバーで返信する**」を選択し、「**通知メッセージ**」をクリックします。
12. E メール・メッセージ・エディターで、メッセージの件名フィールドに次のテキストを入力します。
    ```
    Test Reachability
    ```
13. 「**宛先**」リストで、グローバル管理者を追加します。
14. E メール・メッセージ・エディターを閉じ、「**はい**」をクリックしてこれらの変更を保存します。
15. 「**完了**」をクリックします。
16. 「**適用**」をクリックしてから、「**OK**」をクリックします。

次のタスク

Microsoft Office 365 エージェントを構成します。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Microsoft Office 365 エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

Microsoft Office 365 エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Microsoft Office 365**」を右クリックしてから、「**エージェントを構成します**」をクリックします。
3. 「**Monitoring Agent for Microsoft Office 365**」ウィンドウで、以下のステップを実行します。
   a) 「**Office365 サブスクリプションの詳細 (Office365 Subscription Details)**」タブで、Office 365 のグローバル管理者のユーザー名とパスワードを入力し、「**次へ**」をクリックします。

b)「**シンセティック・トランザクション**」タブで、「**到達可能性の E メール・アドレス**」フィールドに、E メール・アドレスのリストをセミコロンで区切って入力します。

c) Skype QoS メトリックのデータ収集を有効にするには、「**Skype QoS**」チェック・ボックスを選択し、「**次へ**」をクリックします。

d)「**メールボックスおよび OneDrive 使用モニター (Mailbox and OneDrive Usage Monitoring)**」タブで、「**収集間隔**」リストから収集間隔 (時間単位) の期間を選択し、「**次へ**」をクリックします。

4.「**Monitoring Agent for Microsoft Office 365**」ウィンドウで、「**はい**」をクリックします。

次のタスク

- Skype QoS シンセティック・トランザクションをモニターするように Skype シンセティック・トランザクション・ユーティリティーを 構成します。Skype QoS のモニター方法について詳しくは、539 ページの『Skype QoS のモニター』を参照してください。
- ローカル・ユーザーからドメイン・ユーザーにユーザー・アカウントを変更します。詳しくは、539 ページの『ユーザー・アカウントの変更』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。
- トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

Microsoft Office 365 エージェントをインストールするときは、エージェントを構成し、構成設定の指定後に手動で開始する必要があります。サイレント応答ファイルを使用して、カスタム設定を構成します。

始める前に

応答ファイルを編集して、次のデフォルトの構成設定を変更します。

KMO_USER_NAME
Office 365 グローバル管理者のユーザー名。

KMO_PASSWORD
Office 365 グローバル管理者のパスワード。

KMO_MAIL_ADDRESSES1
メールボックスの到達可能性を検証するために宛先にする E メール・アドレスのリスト。E メール・アドレスのリストはセミコロンを使用して区切る必要があります。

KMO_SKYPE
このパラメーターは、Skype QoS シンセティック・トランザクションの収集を有効にするために使用されます。

KMO_DATA_COLLECTION_DURATION
エージェントがメールボックスおよび OneDrive の使用データをフェッチする前に待機する期間 (時間数)。

応答ファイルは、次の場所から入手できます。
`<CANDLEHOME>¥samples`

このタスクについて

エージェントは、エージェントが実行状態でも停止状態でも構成できます。エージェントは、構成後も同じ状態を維持します。例えば、エージェントが実行中だった場合は、構成後も実行状態を維持します。

手順

Microsoft Office 365 エージェントを構成するには、以下のステップを実行します。

1. コマンド・プロンプトで、`microsoft_office365-agent.bat` ファイルが格納されているディレクトリーへのパスに切り替えます。

2. `microsoft_office365-agent.bat` *absolute path to the response file* というコマンドを入力します。

   応答ファイルには、以下のパラメーターがあります。
3. エージェントが停止状態の場合は、エージェントを開始します。

次のタスク

- Skype QoS シンセティック・トランザクションをモニターするように Skype シンセティック・トランザクション・ユーティリティーを 構成します。Skype QoS のモニター方法について詳しくは、539 ページの『Skype QoS のモニター』を参照してください。
- ローカル・ユーザーからドメイン・ユーザーにユーザー・アカウントを変更します。詳しくは、539 ページの『ユーザー・アカウントの変更』を参照してください。
- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。
- トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

ユーザー・アカウントの変更

Microsoft Office 365 エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更します。

このタスクについて

デフォルトでは、Microsoft Office 365 エージェント はローカル・ユーザー・アカウントで実行されます。

手順

1. 次のコマンドを実行して、エージェントを開始するために使用されるユーザー ID を検証します。

   install_dir¥InstallITM¥KinCinfo.exe -r
2. 管理者グループに属さないユーザー ID を使用して、モニター・エージェントが開始された場合、エージェントは停止します。
3.「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウを開きます。
4.「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウで、エージェント・インスタンスを右クリックし、「**スタートアップの変更**」をクリックします。
5. 完全修飾ユーザー ID として <Domain¥User ID> を指定した後、パスワードを指定します。
6. Microsoft Office 365 エージェントを始動します。

Skype QoS のモニター

Skype QoS をモニターするには、エージェントがインストールされている Windows クライアント、または Skype for Business クライアントが構成されている分散環境で、Skype シンセティック・トランザクション・ユーティリティー `kmoskypecaller.exe` と `Kmoskypereceiver.exe` をユーザーが構成する必要があります。

始める前に

シンセティック・トランザクションを実行するには、次の形式で、`<CANDLEHOME>¥tmaitm6_x64¥kmoskypecallerlist.properties` ファイル内の Skype 発信者名と Skype 受信者名を更新する必要があります。
Skype 発信者 = Skype 受信者
例: john@xyz.com = alan@xyz.com
次の形式で、単一の Skype 発信者に対して、複数の Skype 通話受信者を追加できます。
Skype 発信者 = Skype 受信者のリスト
例: john@xyz.com = alam@xyz.com;bill@xyz.com;chuk@xyz.com

要確認 : シンセティック・トランザクションを実行せずに、リアルタイム・ユーザーの Skype QoS をモニターする場合、<CANDLEHOME>¥TMAITM6_x64¥kmoskypecallerlist.properties ファイルの更新は必要ありません。

このタスクについて

Microsoft Office 365 エージェントを構成して開始すると、<CANDLEHOME>¥TMAITM6_x64¥ に次のファイルとフォルダーが作成されます。

- kmoskypecaller.properties
- kmoskypecallerlist.properties
- KMOSynthTransSkype.zip
- KMOSkypeTransReceiver.zip

また、kmoskypecaller.properties ファイルが更新されて、エージェントと kmoskypecaller ユーティリティーとの間の通信に使用されているサーバー IP とポートが反映されます。

手順

Skype 発信者と Skype 受信者を構成し、インスタント・メッセージング、オーディオとビデオの通話、アプリケーション共有セッションなどのシンセティック・トランザクションを開始するには、次の手順を実行します。

1. Office 365 エージェントを開始します。
2. エージェント・クライアントの KMOSynthTransSkype.zip ファイルを Skype 通話の開始元の Windows クライアントにコピーします。
3. KMOSynthTransSkype.zip ファイルを解凍します。
4. エージェント・クライアントの kmoskypecaller.properties ファイルを、Skype 通話の開始元となる Windows クライアント上の解凍した KMOSynthTransSkype フォルダーにコピーします。
5. Skype 通話を受信する必要があるすべての Windows クライアントにエージェント・クライアントの KMOSkypeTransReceiver.zip ファイルをコピーします。
6. Skype 通話を受信する必要があるすべての Windows クライアントで、KMOSkypeTransReceiver.zip ファイルを解凍し、KMOSkypeTransReceiver.exe を実行してメッセージの受信を開始します。
7. シンセティック・トランザクションを開始するには、Windows クライアント上の解凍した KMOSynthTransSkype フォルダーにある KMOSynthTransSkype.exe ファイルを実行します。Office 365 エージェントで、発信者クライアントからの Skype モニター・データの受信が開始されます。

タスクの結果

エージェントは、Skype QoS のモニターを開始します。

ローカル環境変数の構成

ローカル環境変数を構成して、Microsoft Office 365 エージェントの動作を変更することができます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**アクション**」メニューから「**拡張**」>「**ENV ファイルの編集**」をクリックします。
3. 環境変数ファイルに、環境変数の値を入力します。

   構成可能な環境変数については、541 ページの『ローカル環境変数』を参照してください。

ローカル環境変数

Microsoft Office 365 エージェントの動作は、ローカル環境変数を構成することで変更できます。

エージェントのデータ収集方式を定義するための変数

エージェントのデータ収集方式を設定するには、以下の環境変数を使用します。

- **CDP_DP_INITIAL_COLLECTION_DELAY**: この変数を使用して、スレッド・プールがそのデータ収集を開始するまでの時間間隔 (秒) を設定します。
- **KMO_MAILBOX_REACHABILITY_INTERVAL**: この変数を使用して、メールボックス到達可能性属性グループのデータ収集間隔 (分) を設定します。
- **KMO_SKYPE_REPORT_INTERVAL**: この変数を使用して、Skype for Business 使用統計機能のデータ収集間隔 (時間) を設定します。
- **KMO_SERVICE_API_INTERVAL**: この変数を使用して、Office 365 サービス正常性機能のデータ収集間隔 (分) を設定します。
- **KMO_NETWORK_CONNECTION_INTERVAL**: この変数を使用して、インターネット接続機能のデータ収集間隔 (分) を設定します。
- **KMO_NETWORK_PERFORMANCE_INTERVAL**: この変数を使用して、Office 365 サービス・ネットワーク・パフォーマンス機能のデータ収集間隔 (分) を設定します。
- **KMO_SITE_CONNECTION_INTERVAL**: この変数を使用して、Office 365 接続機能のデータ収集間隔 (分) を設定します。
- **KMO_SPSITE_COLLECTION_INTERVAL**: この変数を使用して、SharePoint Sites 詳細機能のデータ収集間隔 (分) を設定します。
- **KMO_UASGE_STATS_INTERVAL**: この変数を使用して、Office 365 Services 使用およびユーザー統計機能のデータ収集間隔 (時間) を設定します。
- **KMO_TENANT_INTERVAL**: この変数を使用して、Office 365 テナント機能のデータ収集間隔 (分) を設定します。
- **KMO_ONEDRIVE_CONNECTIVITY_INTERVAL**: この変数を使用して、Office 365 OneDrive 接続機能のデータ収集間隔 (分) を設定します。
- **KMO_TENANT_DOMAIN**: この変数を使用して、テナントのドメイン・ネームを設定します。

Microsoft SharePoint Server モニターの構成

Monitoring Agent for Microsoft SharePoint Server をインストールすると、エージェントはデフォルトの構成設定で自動的に構成され、起動されます。デフォルトの構成設定を変更するには、サイレント応答ファイルを使用します。

始める前に

以下のタスクを確実に完了してください。

- Microsoft SharePoint Server 環境またはアプリケーションに接続するユーザーが確実に管理者特権を持つようにします。管理者特権を持つ既存のユーザーを使用するか、新規ユーザーを作成してください。新規ユーザーを管理者グループに追加することにより、新規ユーザーに管理者特権を割り当てます。

  要確認 : Microsoft SharePoint Server エージェントの構成には、ローカル・ユーザーまたはドメイン・ユーザーを使用できます (ただし、ユーザーが管理者特権を持っている必要があります)。
- 応答ファイルを編集して、デフォルトの構成パラメーターを変更します。

  応答ファイルには、以下のパラメーターがあります。

  KQP_DB_User
  データベースのユーザー ID。

KQP_DB_Password
データベースのパスワード。

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Microsoft SharePoint Server エージェント を参照してください。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン』を参照してください。以前のエージェント・リリースに関する資料にアクセスするには、次の表を参照してください。

表 *180.* エージェントのバージョンおよび資料

Microsoft SharePoint Server エージェントのバージョン	**資料**
06.31.09.00, 06.31.10.00	IBM Cloud Application Performance Management
06.31.09.00	IBM Performance Management 8.1.3 **注 :** リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。
06.31.07.00	IBM Performance Management 8.1.2 **注 :** リンクをクリックすると、オンプレミス版の Knowledge Center のトピックが開きます。

手順

Microsoft SharePoint Server エージェントを構成するには、以下のステップを実行します。

1. コマンド・プロンプトを開きます。
2. `ms_sharepoint_server-agent.bat` ファイルが格納されているディレクトリーへのパスを変更します。
3. **ms_sharepoint_server-agent.bat config** *absolute path to the response file* というコマンドを入力します。
4. エージェントが停止状態の場合は、エージェントを開始します。

次のタスク
エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。ユーザー・アカウントの変更手順については、542 ページの『ユーザー・アカウントの変更』を参照してください。

ユーザー・アカウントの変更

Microsoft SharePoint Server エージェントを構成した後、ユーザー・アカウントをローカル・ユーザーからドメイン・ユーザーに変更できます。

このタスクについて
ドメイン・ユーザーの場合、エージェントは Microsoft SharePoint Server エージェント のすべてのコンポーネントをモニターできます。

手順

ユーザー・アカウントを変更するには、以下のステップを実行します。

1. 次のコマンドを実行して、エージェントを開始するために使用されるユーザー ID を検証します。
   install_dir¥InstallITM¥KinCinfo.exe -r

2. 管理者グループに属さないユーザー ID を使用して、モニター・エージェントが開始された場合、エージェントは停止します。
3. 「**Monitoring Services の管理 (Manage Monitoring Services)**」ウィンドウを開きます。
4. エージェント・インスタンスを右クリックし、「**スタートアップの変更**」をクリックします。
5. 完全修飾ユーザー ID として <Domain¥Userid> を指定した後、パスワードを指定します。
6. モニター・エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

非管理者ユーザーによる Monitoring Agent for Microsoft SharePoint Server の実行

ローカル・セキュリティー・ポリシーを使用して、Monitoring Agent for Microsoft SharePoint Server を非管理者ユーザーによって実行できるようにすることができます。

このタスクについて

以下の 2 つのローカル・セキュリティー・ポリシーを組み合わせることで、Microsoft SharePoint Server エージェントを非管理者ユーザーによって実行できるようにすることができます。

1. プログラムのデバッグ
2. サービスとしてログオン (Log on as Service)

これらのローカル・セキュリティー許可を非管理者ユーザーに付与するには、以下の手順を実行します。

手順

1. TEMA にアクセスして、Microsoft SharePoint Server エージェントの始動を非管理者ユーザーを指定して変更します。
2. レジストリー・キー HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Office Server ディレクトリーの下に非管理者ユーザーを追加し、読み取り権限を付与します。
3. レジストリー・キー HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Shared Tools¥Web Server Extensions の下に非管理者ユーザーを追加し、読み取り権限を付与します。
4. レジストリー・キー HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Shared Tools¥Web Server Extensions¥16.0¥Secure¥ の下に非管理者ユーザーを手動で追加し、読み取り権限を付与します。
5. レジストリー・キー HKEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring ディレクトリーの下に非管理者ユーザーを追加し、すべての権限を付与します。
6. レジストリー・キー HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows NT ¥CurrentVersion¥Perflib ディレクトリーの下に非管理者ユーザーを追加し、読み取り権限を付与します。
7. SharePoint エージェントのインストール・フォルダー (C:¥IBM¥APM などの Candle フォルダー) に非管理者ユーザーを追加し、すべての権限を付与します。
8. 「**スタート**」**メニュー**で **secpol.msc** コマンドを実行して、「**ローカル セキュリティ ポリシー**」を開きます。
9. 「ローカル セキュリティ ポリシー」で非管理者ユーザーを追加します (544 ページの『ローカル・セキュリティー・ポリシーの許可』を参照)。
10. 「SQL Server ログイン」ユーザー・グループに非管理者ユーザーを追加します。このユーザーは SQL Server 上で sysadmin SQL Server ロール権限を保持している必要があります。
11. Microsoft SharePoint Server エージェントを再始動します。
12. APM ポータルで Microsoft SharePoint Server エージェントの状況を確認し、データを検証します。
13. 以下の属性グループには、Administrators グループのメンバーであるユーザーのデータが表示されます。

a) 可用性

b) Web サービス

ローカル・セキュリティー・ポリシーの許可

ローカル・セキュリティー・ポリシーを使用して、Microsoft SharePoint Server エージェントを非管理者ユーザーによって実行できるようにすることができます。これらのポリシーを使用することで、エージェントの開始または停止、構成、およびデータ検証が可能になります。以下の 2 つのローカル・セキュリティー・ポリシーを使用すると、Microsoft SharePoint Server エージェントを非管理者ユーザーによって実行できるようになります。

「サービスとしてログオン (Log on as Service)」許可の付与

「サービスとしてログオン (Log-on as Service)」許可を付与できます。

このタスクについて

「サービスとしてログオン (Log-on as Service)」許可を付与するには、Microsoft SharePoint Server エージェントで以下の手順を実行します。

手順

1. 「**スタート**」 > 「**管理ツール**」 > 「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. ナビゲーション・ペインで、 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。 ユーザー権限のリストが開きます。
3. 「**サービスとしてログオン (Log-on as service)**」ポリシーをダブルクリックします。「**サービスとしてログオンのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**OK**」をクリックします。

「プログラムのデバッグ」許可の付与

「プログラムのデバッグ」許可を付与できます。

このタスクについて

「プログラムのデバッグ」許可を付与するには、Microsoft SharePoint Server エージェントで以下の手順を実行します。

手順

1. 「**スタート**」 > 「**管理ツール**」 > 「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル ポリシー**」を展開し、「**ユーザー権利の割り当て**」をクリックします。 ユーザー権限のリストが開きます。
3. 「**プログラムのデバッグ**」ポリシーをダブルクリックします。「**プログラムのデバッグのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「選択するオブジェクト名を入力してください」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**OK**」をクリックします。

Microsoft SQL Server のモニターの構成

エージェントがモニター対象のアプリケーションからデータを収集できるように、Monitoring Agent for Microsoft SQL Server を構成する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。最新のシステム要件情報については、Microsoft SQL Server エージェントの Software Product Compatibility Reports (SPCR) を参照してください。

コマンド・プロンプト・インターフェースを 使用して、Microsoft SQL Server エージェントをローカルでインストールおよび構成することができます。モニター対象のサーバー上にエージェントがインストールされていることを確認してください。

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

Microsoft SQL Server エージェントは複数インスタンス・エージェントです。エージェントの各インスタンスを手動で構成し開始する必要があります。

- エージェントを構成するには、以下のタスクを実行します。
  - ユーザーを作成し、必要な許可を付与する
  - モニター用のデータベースを選択する
  - ローカル環境変数を構成する
- クラスター環境でエージェントを実行する場合は、トピック『クラスター環境でのエージェントの実行』で説明されているステップを実行します。

ユーザーの作成および権限の付与

Microsoft SQL Server で、エージェントの実行元にするユーザーを作成し、そのユーザーに、Microsoft SQL Server をモニターする権限を付与する必要があります。権限を付与するプロセスは、Microsoft SQL Server 2005 以降では同じです。

始める前に

Microsoft SQL Server エージェント をインストールします。ユーザーを作成し、そのユーザーに権限を付与するには、sysdamin 許可の役割を持つデータベース管理者である必要があります。

このタスクについて

既存の SQL Server ユーザーが Microsoft SQL Server をモニターするための十分な権限を持っているかどうかを判別するには、以下の手順を使用します。

- Windows 546 ページの『既存の SQL Server ユーザーの権限の確認』

次のいずれかの手順を使用して、ユーザーを作成します。

- Windows 547 ページの『Windows 認証を使用した SQL Server ユーザー ID の作成』
- Linux Windows 547 ページの『SQL Server 認証を使用した SQL Server ユーザー ID の作成』

権限を付与するには、以下の手順を使用します。

- Windows 548 ページの『データ収集のための最小限の権限の付与』
- Windows 549 ページの『特定のデータ・セットのデータを収集するための Perflib レジストリー・キーへのアクセス権の付与』

既存の SQL Server ユーザーの権限の確認

Windows ユーティリティー・ツール **koqVerifyPerminssions.exe** を実行して、既存の SQL Server ユーザーが SQL Server データベースに関連する十分な権限を持っているかどうかを確認できます。

このタスクについて

ユーザーが **sysadmin** 役割または最小限の必要な権限を持っている場合、ユーティリティー・ツール **koqVerifyPerminssions.exe** は、メッセージ PASS を返します。詳細な検査結果は、koqVerifyPermissions_log ログに記録されています。

以下は、最小限の権限のリストです。

- サーバーの権限には、「**サーバー状態の表示**」、「**データベースの状態の表示**」、および「**定義の表示**」が含まれている必要があります。

  これらのサーバー・レベル権限は必須です。
- モニターするすべてのシステム・データベースとユーザー定義データベースについて、データベース役割メンバーシップには、**public** および **db_owner** が含まれている必要があります。

  db_owner 権限は、以下のデータ・セットのデータを収集するために必要です。
  - サーバーの詳細データ・セット
  - データベース詳細データ・セット
  - データベースのミラーリング・データ・セット
  - サーバーの要約データ・セット
  - ジョブ要約データ・セット
- **msdb** データベースについて、データベース役割メンバーシップには、**db_datareader**、**SQLAgentReaderRole**、および **SQLAgentUserRole** が含まれている必要があります。これらの権限は、ジョブの詳細データ・セットのために必要です。

手順

1. コマンド・プロンプトを起動し、以下のユーティリティー・ディレクトリーに変更します。
   - 64 ビットのエージェントの場合は、*Agent_home*¥TMAITM6_x64
   - 32 ビットのエージェントの場合は、*Agent_home*¥TMAITM6

   ここで、*Agent_home* はエージェント・インストール・ディレクトリーです。
2. 以下のパラメーターを指定して、**koqVerifyPerminssions.exe** を実行します。

   ```
   koqVerifyPermissions.exe -S Instance_name -U Username -P Password
   ```

   各部の意味は次のとおりです。
   - *Instance_name* は、SQL Server インスタンス名です。
   - *Username* は、ユーティリティー・ツールによって検証されるユーザー名です。
   - *Password* は、ユーザーのパスワードです。このパラメーターは、*username* が指定されている場合に必要です。

   注 : *username* および *password* が指定されていない場合は、システムへのログオンであるデフォルト・ユーザーが使用されます。例: NT AUTHORITY¥SYSTEM。

タスクの結果

詳細な検査結果は、以下のディレクトリーにある koqVerifyPermissions_log にあります。

- 64 ビットのエージェントの場合は、*Agent_home*¥TMAITM6_x64¥logs
- 32 ビットのエージェントの場合は、*Agent_home*¥TMAITM6¥logs

ここで、*Agent_home* はエージェント・インストール・ディレクトリーです。

Windows 認証を使用した SQL Server ユーザー ID の作成

Windows Windows 認証を使用して新しいユーザーを作成し、そのユーザーに必要な役割と権限を割り当てます。

手順

ユーザーを作成するには、以下の手順を実行します。

1. **SQL Server Management Studio** で、「**オブジェクト エクスプローラー**」を開きます。
2. 「***Server_instance_name***」>「**セキュリティ**」>「**ログイン**」をクリックします。
3. 「**ログイン**」を右クリックし、「**新規ログイン**」を選択します。
4. 「**全般**」ページの「**ログイン名**」フィールドに、Windows ユーザーの名前を入力します。
5. 「**Windows 認証**」を選択します。
6. このユーザーに割り当てる役割と権限に応じて、次のいずれかのタスクを実行します。
   - 「サーバー ロール」ページで、新しいログイン ID に *sysadmin* 役割を割り当てます。
   - ユーザーに *sysadmin* 役割を割り当てない場合は、548 ページの『データ収集のための最小限の権限の付与』で説明されている手順を実行することで、ユーザーに最小限の権限を付与します。

   重要 : デフォルトでは、新しいログイン ID には *public* 役割が割り当てられます。
7. 「**OK**」をクリックします。

タスクの結果

デフォルトの *public* 役割と、割り当てた権限を持つユーザーが作成され、「**ログイン**」リストに表示されます。

SQL Server 認証を使用した SQL Server ユーザー ID の作成

Linux Windows SQL Server 認証を使用して新しいユーザーを作成し、そのユーザーに必要な役割と権限を割り当てます。

手順

ユーザーを作成するには、以下の手順を実行します。

1. **SQL Server Management Studio** で、「**オブジェクト エクスプローラー**」を開きます。
2. 「***Server_instance_name***」>「**セキュリティ**」>「**ログイン**」をクリックします。
3. 「**ログイン**」を右クリックし、「**新規ログイン**」を選択します。
4. 「**全般**」ページの「**ログイン名**」フィールドに、新しいユーザーの名前を入力します。
5. 「**SQL Server 認証**」を選択します。
6. 「**パスワード**」フィールドに、ユーザーのパスワードを入力します。
7. 「**パスワードの確認**」フィールドに、「**パスワード**」フィールドに入力したパスワードを再入力します。
8. このユーザーに割り当てる役割と権限に応じて、次のいずれかのタスクを実行します。
   - 「サーバー ロール」ページで、新しいログイン ID に *sysadmin* 役割を割り当てます。
   - ユーザーに *sysadmin* 役割を割り当てない場合は、548 ページの『データ収集のための最小限の権限の付与』で説明されている手順を実行することで、ユーザーに最小限の権限を付与します。

   重要 : デフォルトでは、新しいログイン ID には *public* 役割が割り当てられます。
9. 「**OK**」をクリックします。

タスクの結果

デフォルトの *public* 役割と、割り当てた権限を持つユーザーが作成され、「**ログイン**」リストに表示されます。

データ収集のための最小限の権限の付与

Windows デフォルトの **public** 役割とは別に、エージェントがデータ・セットのデータを収集できるように、**sysadmin** 役割をユーザーに割り当てるか、最小限の権限をユーザーに付与することができます。

このタスクについて

権限の付与は、ユーザー・インターフェースまたはユーティリティー ・ツール **permissions.cmd** から行えます。

手順

- ユーザー・インターフェースからユーザーに 最小限の権限を付与するには、以下の手順を実行します。
  a)「**サーバー ロール**」ページで、**public** チェック・ボックスが選択されていることを確認します。
  b)「**ユーザー マッピング**」ページを開き、モニターするすべてのシステム・データベースとユーザー定義データベースについて、以下のチェック・ボックスを選択します。
  - **public**
  - **db_owner**

  msdb データベースの場合は、さらに以下のチェック・ボックスも選択します。
  - **db_datareader**
  - **SQLAgentReaderRole**
  - **SQLAgentUserRole**

  c)「**セキュリティ 保護可能なリソース**」ページを開き、モニターするサーバー・インスタンスについて、以下のチェック・ボックスを選択します。
  - データベースの状態の表示
  - 定義の表示
  - サーバー状態の表示
- ユーティリティー・ツール **permissions.cmd** を使用してユーザーに最小限の権限を付与するには、以下を実行します。
  a) Windows Explorer を起動し、ユーティリティー ・ツールの以下のディレクトリー *Agent_grant_perm_dir* を参照します。
  - 64 ビットのエージェントの場合、*Agent_grant_perm_dir* は *Agent_home*¥TMAITM6_x64¥scripts¥KOQ¥GrantPermission です。
  - 32 ビットのエージェントの場合、*Agent_grant_perm_dir* は *Agent_home*¥TMAITM6¥scripts¥KOQ¥GrantPermission です。
  - *Agent_home* はエージェントのインストール・ディレクトリーです。

  重要 : ユーティリティー ・ツール **permissions.cmd** は、デフォルトで、すべてのデータベースに対する **db_owner** を付与します。特定のデータベースを除外するには、データベース名を *Agent_grant_perm_dir*¥exclude_database.txt ファイルに追加する必要があります。データベース名は、記号別名 **@** で区切る必要があります。

  ヒント : 例えば、データベース **MyDatabase1** および **MyDatabase2** を除外する場合は、exclude_database.txt ファイルに以下の項目を追加します。

  ```
  MyDatabase1@MyDatabase2
  ```

  b) **permissions.cmd** をダブルクリックしてユーティリティー ・ツールを起動します。
  c) プロンプトが出されたら、指定するパラメーター値を入力します。

表 *181.* パラメーター	
パラメーター	**説明**
SQL Server 名または SQL Server インスタンス名	ユーザーに権限を付与する対象のターゲット SQL Server 名またはターゲット SQL Server インスタンス名を入力します。
既存の SQL Server ユーザーのログオン名	権限を変更するユーザーの名前を入力します。
権限オプション: **1 db_owner** 権限の付与 **2 db_datareader**、**SQLAgentReaderRole**、および **SQLAgentUserRole** 権限の付与 **3** すべての必要な権限の付与	必要に応じて、**1**、**2**、または **3** を入力します。
権限を付与する対象のユーザー: **1** 現在システムにログオンしているユーザー **2** 別のユーザー	**1** または **2** を入力します。 **2** を選択した場合は、プロンプトで対象ユーザー名を入力します。 **注 :** ユーザーが他のユーザーに権限を付与するには、アクセス権限が必要です。

次のタスク
エージェントを構成します。

特定のデータ・セットのデータを収集するための Perflib レジストリー・キーへのアクセス権の付与
Windows 少数のデータ・セットのデータを収集するために、ユーザーに Perflib レジストリー・キーへの読み取り権限を付与する必要があります。

このタスクについて

この権限は、エージェント・サービスの構成に使用する Windows ユーザーに付与する必要があります。Perflib アクセス権がないことにより影響を受けるデータ・セットは、MS SQL データベース詳細、MS SQL メモリー・マネージャー、MS SQL ロック・リソース・タイプ要約、MS SQL ジョブの要約、MS SQL Server トランザクションの要約、MS SQL Server の要約など多数存在します。

手順

Perflib レジストリー・キーへのアクセス権を付与するには、以下の手順を実行します。

1. レジストリー・エディターを開くために、「**スタート**」 > 「**実行**」 > 「**Regedit.exe**」と入力し、**Enter** を押します。
2. `HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows NT¥CurrentVersion¥Perflib` レジストリー・キーに進みます。
3. **Perflib** キーを右クリックし、「**アクセス許可**」をクリックします。
4. 「**追加**」をクリックし、エージェントのインストールおよび構成に使用する Windows ユーザー名を入力してから、「**OK**」をクリックします。
5. 追加したユーザーをクリックします。
6. チェック・ボックスを選択して、ユーザーに読み取り権限を許可します。
7. 「**適用**」をクリックしてから「**OK**」をクリックします。

ローカル環境変数

Microsoft SQL Server エージェントの動作は、ローカル環境変数を構成することで変更できます。

SQL Server サービスの可用性を確認するための変数

SQL Server サービスの可用性を確認するには、以下の環境変数を使用します。

- **COLL_MSSQL_RETRY_INTERVAL**: この変数は、SQL Server サービスの状況をチェックする際の再試行間隔 (分単位) を指定します。 値がゼロ以下の場合、この変数のデフォルト値は 1 分となります。
- **COLL_MSSQL_RETRY_CNT**: この変数は、SQL Server サービスが開始されているかどうかについて、SQL Server エージェントがチェックする再試行数を指定します。この変数で指定した回数の再試行の後で SQL Server サービスが開始されていない場合、コレクターは処理を停止します。この変数の値がゼロ以下の場合、この変数はデフォルト値 3 になります。

SQL Server エラー・ログ・ファイルのモニター用変数

「MS SQL エラー・イベントの詳細」データ・セットをモニターするには、以下の環境変数を使用します。

- **COLL_ERRORLOG_STARTUP_MAX_TIME**: この変数は、エージェント開始前のエラー収集の時間間隔 (T) を指定します。デフォルト値は 0 分です。この変数には、以下の値を指定できます。

  T = 0
  エージェントが開始または再始動した時間から、エラー・ログ・ファイルのモニターを開始します。エージェントが開始する前にエラー・ログ・ファイルに記録されたエラーは、読み取られません。

  T = 1
  エージェントは、**COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** 変数に設定される以下の値 (R で示されます) に従ってエラー・ログ・ファイルをモニターします。

  - R < 0 の場合、エージェントが開始または再始動した時間から、エラー・ログ・ファイルのモニターを開始します。
  - R = 1 の場合、エラー・ログ・ファイルに記録されたすべてのエラーをモニターします。
  - R > 1 で、かつエージェントが初めてインストールされた場合、R 個のエラーがモニターされるまでエラー・ログ・ファイルをモニターします。R > 1 で、かつエージェントが再始動された場合、前回モニターされなかった R 個のエラーをすべてモニターします。

  T > 1
  エージェントは、開始または再始動した時間から最大 T 分間に記録された以前のエラーを、すべてモニターします。エージェントのモニタリングは、**COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** 変数に設定された以下の値によっても異なります。

  - R ≤ 0 の場合、エージェントが開始または再始動した時間から、エラー・ログ・ファイルのモニターを開始します。
  - R = 1 の場合、最大 T 分間に記録されたすべてのエラーに対するエラー・ログ・ファイルをモニターします。
  - R > 1 の場合、最後の T 分間に記録された最大 R 個のエラーをモニターします。
- **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW**: この変数は、エージェントの開始時に処理する必要のあるエラーの最大数を指定します。デフォルト値は 0 です。この変数には、以下の値を割り当てることができます。

  R = 0
  エージェントは、開始または再始動した時間から、エラー・ログ・ファイルのモニターを開始します。エージェントが開始する前にエラー・ログ・ファイルに作成されたエラーは、読み取られません。

  R = 1
  エージェントは、開始または再始動した時間から最後の T 分間に記録されたエラーをモニターします。

  R > 1
  エージェントは、最後の T 分間に記録された R 個のエラーをモニターします。

- **COLL_ERRORLOG_MAX_EVENT_ROW**: この変数は、エラーの行数を指定します。デフォルト値は 50 です。この変数には、以下の値を割り当てることができます。

  X = 0
  エージェントは、エラー・ログを表示しません。

  X > 0
  エージェントは、X 行のエラーを表示します。
- **COLL_ERRORLOG_RECYCLE_WAIT**: この変数は、MS SQL エラー・イベントの詳細属性グループのシチュエーションがトリガーされたときにその属性グループのデータを収集する前に Microsoft SQL Server エージェントが待機する時間間隔 (秒) を示します。この変数には、1 から 30 の範囲の値を割り当てることができます。この変数の値がゼロ未満の場合、この変数はデフォルト値ゼロ (秒) になります。この変数の値が 30 より大きい場合、この変数はデフォルト値 30 (秒) になります。

照会タイムアウト間隔を設定するための変数

SQL Server エージェントの照会タイムアウト間隔を設定するには、以下の環境変数を使用します。

- **QUERY_TIMEOUT**: この環境変数は、SQL Server エージェントが SQL Server に送信された照会に対する応答の受信を待機する最大時間 (秒単位) を定義します。この変数の値は、45 秒より少なくする必要があります。ただし、この変数の値が 0 秒に設定されている場合、SQL Server エージェントは SQL Server からの応答の受信を無制限に待機します。ロックされている多数のデータベースに SQL Server エージェントがアクセスする場合は、この変数に 10 から 20 秒の範囲の値を割り当てる必要があります。設定したタイムアウト間隔内に照会が処理されない場合、SQL Server エージェントはタイムアウトした照会をスキップし、キュー内の次の照会に移動します。エージェントは、タイムアウトした照会のデータを表示しません。
- **QUERY_THREAD_TIMEOUT**: この環境変数は、SQL Server エージェントが SQL Server に送信された照会に対する応答の受信を待機する最大時間 (秒単位) を定義します。この環境変数は、スレッド収集を使用するいくつかの属性グループに適用できます。例えば、KOQDBD、KOQTBLD、KOQDEVD などです。この変数の値には、QUERY_TIMEOUT 変数とは異なり、制限はありません。その他の点では、QUERY_TIMEOUT 変数と同様に機能します。

使用可能なジョブに関する情報を表示するための変数

「MS SQL ジョブの詳細」データ・セットの使用可能なジョブに関する情報を表示するには、**COLL_JOB_DISABLED** 環境変数を使用します。この変数の値を 1 に設定すると、使用不可のジョブに関する情報は Microsoft SQL Server エージェントに表示されません。 この変数を指定しない場合は、使用可能なジョブと使用不可のジョブに関する情報を表示できます。

「MS SQL ファイル・グループの詳細」データ・セット内の行数を制限するための変数

「MS SQL ファイル・グループの詳細」データ・セットに対して、コレクター・サービスがフェッチする行数を制限するには、**COLL_KOQFGRPD_MAX_ROW** 環境変数を使用します。 この環境変数は、「MS SQL ファイル・グループの詳細」データ・セットに対して、コレクター・サービスがフェッチする最大行数を定義します。 この変数の値を指定しない場合、コレクター・サービスは「MS SQL ファイル・グループの詳細」データ・セットに対して 10,000 行をフェッチします。この環境変数を使用して、koqcoll.ctl ファイル内の最大行数のデフォルトの制限を変更します。デフォルトの制限を変更するには、以下のステップを実行します。

1. koqcoll.ctl ファイル内で KOQFGRPD に対する最大行数を指定します。
2. **COLL_KOQFGRPD_MAX_ROW** 環境変数を追加し、この変数の値が koqcoll.ctl ファイルで指定した値と同じであることを確認します。

koqcoll.ctl ファイル内の値が **COLL_KOQFGRPD_MAX_ROW** 環境変数で指定されている値より小さい場合は、koqcoll.ctl ファイル内の値が最大行数の値として処理されます。

koqcoll.ctl ファイル内の値が **COLL_KOQFGRPD_MAX_ROW** 環境変数で指定されている値より大きい場合は、**COLL_KOQFGRPD_MAX_ROW** 環境変数内の値が最大行数の値として処理されます。

「MS SQL ファイル・グループの詳細」データ・セットの収集を向上させるための変数

COLL_DBD_FRENAME_RETRY_CNT 変数を使用して、%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_TEMP ファイルを %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_PREV ファイルに移動するために可能な試行回数を指定します。

この変数の値を指定しない場合、Microsoft SQL Server エージェントはファイルの移動を 3 回試行します。

「MS SQL デバイス詳細」データ・セット内の行数を制限するための変数

「MS SQL デバイス詳細」データ・セットに対して、コレクター・サービスがフェッチする行数を制限するには、**COLL_KOQDEVD_MAX_ROW** 環境変数を使用します。この環境変数は、「MS SQL デバイス詳細」データ・セットに対して、コレクター・サービスがフェッチする最大行数を定義します。 この変数の値を指定しない場合、コレクター・サービスは「MS SQL デバイス詳細」データ・セットに対して 10,000 行をフェッチします。この環境変数を使用して、koqcoll.ctl ファイル内の最大行数のデフォルトの制限を変更します。デフォルトの制限を変更するには、以下のステップを実行します。

1. koqcoll.ctl ファイル内で KOQDEVD に対する最大行数を指定します。
2. **COLL_KOQDEVD_MAX_ROW** 環境変数を追加し、この変数の値が koqcoll.ctl ファイルで指定した値と同じであることを確認します。

koqcoll.ctl ファイル内の値が **COLL_KOQDEVD_MAX_ROW** 環境変数で指定されている値より小さい場合は、koqcoll.ctl ファイル内の値が最大行数の値として処理されます。

koqcoll.ctl ファイル内の値が **COLL_KOQDEVD_MAX_ROW** 環境変数で指定されている値より大きい場合は、**COLL_KOQDEVD_MAX_ROW** 環境変数内の値が最大行数として処理されます。

「MS SQL デバイス詳細」データ・セットの収集を向上させるための変数

「MS SQL デバイス詳細」データ・セットの収集を向上させるには、以下の環境変数を使用します。

- **COLL_KOQDEVD_INTERVAL**: この環境変数により、「MS SQL デバイス詳細」データ・セットの 2 つの連続する収集間の時間間隔 (分単位) を指定できます。

  注 : デフォルトでは、「MS SQL デバイス詳細」データ・セットのデータ収集はデマンド・ベースです。**COLL_KOQDEVD_INTERVAL** 変数を使用して、「MS SQL デバイス詳細」データ・セットのスレッド・ベースの収集を開始し、2 つのスレッド収集間の時間間隔を設定します。
- **COLL_DBD_FRENAME_RETRY_CNT**: この環境変数を使用して、%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_TEMP ファイルを %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_PREV ファイルに移動するために可能な試行回数を指定します。

この変数の値を指定しない場合、Microsoft SQL Server エージェントはファイルの移動を 1 回試行します。

「MS SQL データベース詳細」データ・セットの収集を向上させるための変数

「MS SQL データベース詳細」データ・セットの収集を向上させるには、以下の環境変数を使用します。

- **COLL_KOQDBD_INTERVAL**: この環境変数を使用して、「MS SQL データベース詳細」データ・セットの 2 つの連続するスレッド・ベースの収集間の時間間隔 (分単位) を指定します。この変数の値を指定しない場合、または指定された時間間隔が 3 分未満の場合、Microsoft SQL Server エージェントは、デフォルトの 3 分間隔を使用します。収集により長い時間がかかる場合、またはデータが頻繁に NOT_COLLECTED と表示される場合、Database Detail Collection completed in %d seconds のログを参照して収集時間を確認し、この変数値をログに示された収集時間よりも大きい値に設定できます。
- **COLL_DBD_FRENAME_RETRY_CNT**: この環境変数を使用して、%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_TEMP ファイルを %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_PREV ファイルに移動するために可能な試行回数を指定します。

この変数の値を指定しない場合、Microsoft SQL Server エージェントはファイルの移動を 1 回試行します。

「MS SQL 監査の詳細」データ・セットの収集を向上させるための変数

「MS SQL 監査の詳細」データ・セットの収集を向上させるには、以下の環境変数を使用します。

- **COLL_AUDIT_TYPE**: この変数を使用して、特定のログのモニターを使用可能または使用不可にします。この変数のデフォルト値は [AL][FL][SL] です。デフォルトでは、エージェントはアプリケーション・ログ、監査ファイル、およびセキュリティー・ログの 3 つのタイプすべてのログをモニターします。以下に示すとおり、変数の値には各ログ・タイプを表す 2 つの文字コードが含まれます。
  - [AL] はアプリケーション・ログを表します
  - [FL] は監査ファイルを表します
  - [SL] はセキュリティー・ログを表します

  変数の値を変更して、特定のログ・タイプのモニターを無効にすることができます。 例えば、変数の値を [AL][SL] と指定すると、監査ファイルはモニターされません。この変数に値を指定しない場合、監査の詳細はモニターされません。
- **COLL_AUDIT_DURATION**: この変数を使用して、この変数で指定した時間間隔中に発生した監査イベントを報告します。例えば、この変数を 7 に設定した場合、最後の 7 時間に発生した監査イベントのみが「MS SQL 監査の詳細」データ・セットによって報告されます。**COLL_AUDIT_DURATION** 変数のデフォルト値は 24 hours です。
- **COLL_AUDIT_COLLECTION_INTERVAL**:「MS SQL 監査の詳細」データ・セットのスレッド収集は、SQL Server インスタンス上にあるすべてのデータベースの仕様を提供します。この変数を使用して、このスレッド収集の間隔を設定します。例えば、この変数を 7 に設定した場合、データベース仕様の最新セットが 7 時間ごとに SQL Server インスタンスから抽出されます。**COLL_AUDIT_COLLECTION_INTERVAL** 変数のデフォルト値は 24 です。

「MS SQL プロセスの詳細」データ・セットの収集を向上させるための変数

「MS SQL プロセスの詳細」データ・セットの収集を向上させるには、**COLL_PROC_BLOCK_INTERVAL** 変数を以下の値とともに使用します。

- **COLL_PROC_BLOCK_INTERVAL** = 0 の場合、「ブロッキング・プロセスの期間」属性および「ブロッキング・リソースの期間」属性の収集は使用不可になります。
- **COLL_PROC_BLOCK_INTERVAL** = *x* の場合、「ブロッキング・プロセスの期間」属性および「ブロックしているリソースの期間」属性についての 2 つの連続するデータ収集の間隔は *x* 分です。

CANDLE_HOME ディレクトリーで **COLL_PROC_BLOCK_INTERVAL** 変数が設定されていない場合、2 つの連続するデータ収集間の間隔は 3 分になります。

ロックされたオブジェクトをデータ収集から除外するための変数

データベース詳細、 ファイル・グループの詳細、データベースのミラーリング、 およびデバイス詳細の各ワークスペースのために送信する照会の実行に長い時間がかかる場合は、**COLL_DBCC_NO_LOCK** 変数を使用して、値 WITH (NOLOCK) を指定した照会を実行してください。この変数により、照会の実行対象のオブジェクトがロックされている場合に照会がキューで待機することがなくなります。

「表の詳細」データ・セットによって返された行のソート基準を設定するための変数

「表の詳細」データ・セットによって返された行は、**COLL_TBLD_SORTBY** 変数に設定された値によって、降順にソートされます。**COLL_TBLD_SORTBY** 変数のデフォルト値は FRAG (フラグメント化の割合) です。有効な値は、ROWS (表内の行数)、SPACE (表によって使用されるスペース)、および OPTSAGE (表のオプティマイザー統計の経過時間) です。

「MS SQL 問題の詳細」データ・セットおよび「MS SQL 問題の要約」データ・セットの収集を向上させるための変数

- **COLL_ALERT_SEV**: この変数を使用して、「MS SQL 問題の詳細」データ・セットおよび「MS SQL 問題の要約」データ・セットに表示されるエラー・メッセージの重大度レベルを設定します。重大度レベルがこ

の変数で指定された値以上であるエラー・メッセージが、「MS SQL 問題の詳細」データ・セットおよび「MS SQL 問題の要約」データ・セットに表示されます。例えば、この変数の値を 10 に設定した場合、重大度レベルが 10 以上であるエラー・メッセージが「MS SQL 問題の詳細」データ・セットおよび「MS SQL 問題の要約」データ・セットに表示されます。この変数の値を指定しない場合、重大度レベルが 17 以上であるエラー・メッセージが「MS SQL 問題の詳細」データ・セットおよび「MS SQL 問題の要約」データ・セットに表示されます。

- **COLL_SINCE_ERRORLOG_RECY**: この変数を使用して、現在の ERRORLOG ファイル内の重大度が高いエラーのみをモニターします。この変数の値を指定しない場合、変数の値は 0 になります。つまり、データの収集について、「MS SQL 問題の要約」データ・セットは、前の ERRORLOG ファイルから読み取られた重大度が高いエラーも対象に入れます。現在の ERRORLOG ファイル内の重大度が高いエラーのみをモニターするには、この変数の値を 1 に設定します。

タイムアウト間隔を設定するための変数

Microsoft SQL Server エージェントのタイムアウト間隔を設定するには、以下の環境変数を使用します。

- **WAIT_TIMEOUT**: この変数を使用して、Microsoft SQL Server エージェントの待機タイムアウト間隔を指定します。データの収集に 45 秒を超える時間を要するデータ・セットがある場合、エージェントが停止するか、シチュエーションが不適切にトリガーされる可能性があります。 ログを調べて、データの収集に 45 秒を超える時間を要するデータ・セットがないか確認し、**WAIT_TIMEOUT** 変数を使用して、エージェント・プロセスとコレクター・プロセスの間の待機時間を増やします。
- **COLL_DB_TIMEOUT**: この変数を使用して、既存の SQL Server 接続上での照会の実行などの要求が完了するまで待機する間隔 (秒単位) を定義します。この値を超えると、アプリケーションに戻ります。この値を 0 に設定した場合、タイムアウトはありません。この変数の値を指定しない場合、エージェントは 15 秒待機してからアプリケーションに戻ります。

コレクター・ログ・ファイルのプロパティーを設定するための変数

コレクター・ログ・ファイルのプロパティーを設定するには、以下の環境変数を使用できます。

- **COLL_WRAPLINES**: この変数を使用して、col.out ファイル内の最大行数を指定します。この変数のデフォルト値は 90,000 行 (およそ 2 MB) です。
- **COLL_NUMOUTBAK**: この変数を使用して、コレクター・ログ・ファイルのバックアップ・コピーの作成数を指定します。デフォルトでは、コレクター・ログ・ファイルのバックアップ・コピーは 5 個作成されます。バックアップ・ファイルの名前は *.out です。バックアップ・ファイルがいっぱいになると、ファイルの名前が *.ou1 に変更され、最新のログが *.out ファイルに書き込まれます。このような方法で、バックアップ・ファイルが 5 個になると、最古のログは *.ou5 ファイルで入手でき、最新のログは *.out ファイルで入手できます。

  COLL_NUMOUTBAK 変数に以下のいずれかの値を指定すると、コレクター・ログ・ファイルのバックアップ・コピーを 6 個以上作成できます。

  - バックアップ・ファイルが 9 個以下の場合は、**COLL_NUMOUTBAK** 変数に、作成するバックアップ・ファイルの数を指定します。例えば、**COLL_NUMOUTBAK** 変数に 9 を指定した場合、9 個のバックアップ・ファイルが作成されます。
  - バックアップ・ファイルが 10 個以上 999 個以下の場合は、**COLL_NUMOUTBAK** 変数に、前にハイフンを付けてバックアップ・ファイルの数を指定します。例えば、**COLL_NUMOUTBAK** 変数に -352 を指定した場合、352 個のバックアップ・ファイルが作成されます。
- **COLL_DEBUG**: この変数を使用して、この変数の値を「ddddddddddd」 (10 個の「d」) に設定することで、コレクターの完全トレースを有効にします。

一時ファイルを削除するための変数

COLL_TMPFILE_DEL_INTERVAL: この変数を使用して、KOQ_<タイム・スタンプ> 一時ファイルを削除する必要がある時点までの間隔 (分単位) を指定します。この変数の値を指定しない場合、変数の値は 0 になります。つまり、一時ファイルを直ちに削除する必要があります。

MS SQL Server エージェントで使用されるドライバーを変更するための変数

Microsoft SQL Server エージェントで使用されるドライバーを変更するには、**KOQ_ODBC_DRIVER** 環境変数を使用します。この変数は、SQL Server に接続するために Microsoft SQL Server エージェントが使用するドライバーを指定します。この変数の値を指定しない場合、エージェントはデフォルト・ドライバーとして ODBC SQL Server Driver を使用します。

注 : Microsoft SQL Server ドライバーを指定する場合、ドライバー名が正しいこと、およびそのドライバーがデータ・ソースのドライバーのオプション (ODBC) にリストされていることを確認してください。

AlwaysOn が有効な SQL Server データベースに接続するための変数

KOQ_APPLICATION_INTENT: この変数を使用して、SQL Server への接続中の接続オプションを指定します。
KOQ_APPLICATION_INTENT オプションの詳細:

- **Readonly**: 接続は、**ApplicationIntent** を *readonly* に指定してオープンされます。
- **Readwrite**: 接続は、**ApplicationIntent** を *readwrite* に指定してオープンされます。
  これが Readwrite に設定されている場合、Microsoft SQL Server エージェントは、接続での書き込み操作を実行しません。

この変数が設定されていない場合、接続は、**ApplicationIntent** プロパティーなしで確立されます。

注 : ドライバーは、環境変数 **KOQ_ODBC_DRIVER** によって指定されます。この変数が設定されていない場合は、デフォルトの SQL Server ドライバーが使用されます。
ドライバーが **ApplicationIntent** をサポートしていない場合、接続は、**ApplicationIntent** プロパティーなしでオープンされます。

エージェントの構成パラメーター

エージェントの必須構成パラメーターを指定する必要があります。

このタスクについて

次の表に、構成パラメーターの詳細を示します。パラメーターを確認し、各パラメーターの値を決定してください。

パラメーター名	説明	デフォルト値	必須フィールド
ユーザー名	エージェントと SQL Server との間の接続を確立するために使用するユーザー名またはログイン	NA	はい
パスワード	ユーザーまたはログインのパスワード	NA	はい
データベース・バージョン	モニター対象の SQL Server データベースのバージョン	NA	はい
データベース・サーバーのホーム・ディレクトリー	SQL Server データベースのホーム・パス	NA	はい
エラー・ログ・ファイルのパス	SQL Server のエラー・ログ・ファイルがある場所	NA	はい

Windows システムでのエージェントの構成

「IBM® Cloud Application Performance Management」ウィンドウを使用して、Windows システム上でエージェントを構成できます。

始める前に

エージェントを構成する前に、以下のタスクを必ず完了してください。

- ユーザーを作成し、必要な許可を付与する
- ローカル環境変数の確認

このタスクについて

Microsoft SQL Server エージェントは複数インスタンス・エージェントです。エージェントの各インスタンスを手動で構成し開始する必要があります。

- エージェントを構成するには、以下のタスクを実行します。
  - モニター用のデータベースを選択する
  - ローカル環境変数を構成する

モニター用のデータベースの選択

「**データベース・エージェントの構成**」ウィンドウを使用して、モニターするデータベースを選択できます。

手順

1. 「**IBM Performance Management**」ウィンドウを開きます。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列をクリックし、「**テンプレート**」を右クリックし、「**デフォルトを使用して構成**」をクリックします。
3. 「**データベース・エージェントの構成**」ウィンドウで、モニターするデータベース・サーバーを「**使用可能なデータベース・サーバー**」から選択し、「**モニター対象のサーバー**」リストに移動します。
4. 「**データベース・サーバーのプロパティー**」ウィンドウで、以下のフィールドの値が自動的に設定されます。
   - サーバー名
   - データベース・バージョン
   - ホーム・ディレクトリー
   - エラー・ログ・ファイル

   「**データベース・サーバーのプロパティー**」ウィンドウの次のフィールドは、オプションです。
   - Windows 認証
   - 長期継続データベース接続のサポート
   - 拡張パラメーター
   - すべてのデータベースのモニター
   - 日単位の頻度
   - 週単位の頻度
   - 月単位の頻度
   - 収集開始時刻
   - 表の詳細の連続収集

   「**データベース・サーバーのプロパティー**」ウィンドウの構成パラメーターについて詳しくは、557 ページの『「データベース・サーバーのプロパティー」の構成パラメーター』を参照してください。
5. 「**Windows 認証**」フィールドを選択しない場合は、ASCII 文字のみを使用して、「**ログイン**」フィールドおよび「**パスワード**」フィールドにユーザー ID およびパスワードを入力してください。

6. 「**拡張パラメーター**」フィールドに、データ収集を無効にするデータ・セットの名前を入力し、「**OK**」をクリックします。

   例:

   - 表の詳細データ・セットのデータ収集を無効にするには、koqtbld を入力します。
   - データベース詳細データ・セットのデータ収集を無効にするには、koqdbd を入力します。
   - 表の詳細データ・セットおよびデータベース詳細データ・セットのデータ収集を無効にするには、koqtbld,koqdbd を入力します。
7. 「**すべてのデータベースのモニター**」チェック・ボックスを選択しない場合は、「**データベース**」グループ領域のフィールドに、モニターを有効または無効にする対象のデータベースのリストを指定します。

   要確認 : 「**すべてのデータベースのモニター**」チェック・ボックスを選択し、「**データベース**」グループ領域にデータベースを指定した場合は、「**すべてのデータベースのモニター**」チェック・ボックスの設定が優先されます。
8. MS SQL 表の詳細データ・セットの収集では、頻度を指定します。指定できる値は日単位、週単位、月単位です。
9. 「**表の詳細の連続収集**」チェック・ボックスを選択して、MS SQL 表の詳細データ・セットの連続収集を使用可能にします。「**表の詳細の連続収集**」チェック・ボックスを選択する場合は、「**2 つの連続収集間の間隔 (分単位)**」フィールドに値を入力してください。
10. 「**データベース・エージェントの構成**」ウィンドウで「**OK**」をクリックしてから、エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

「データベース・サーバーのプロパティー」の構成パラメーター

「**データベース・サーバーのプロパティー**」ウィンドウで、サーバー名、データベースのバージョン、ホーム・ディレクトリーなどのデータベース・サーバーのプロパティーを構成することができます。

以下の表で、「**データベース・サーバーのプロパティー**」ウィンドウでの構成設定について詳しく説明します。

表 *182.* **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明			
パラメーター名	**説明**	**必須フィールド**	**例**
サーバー名	モニターする Microsoft SQL Server インスタンスの名前。 デフォルト・インスタンスのインスタンス名として MSSQLSERVER を使用します。 この名前は、管理対象システム名全体に収まる長さ (2 文字から 32 文字の間) でなければなりません。	はい	モニター対象の Microsoft SQL Server インスタンスがデフォルトの Microsoft SQL Server インスタンスである場合は、このフィールドに MSSQLSERVER と入力します。 モニター対象の Microsoft SQL Server インスタンスが名前付きインスタンスであり、インスタンス名が `mysqlserver` で、ホスト名が popcorn の場合、このフィールドには `mysqlserver` と入力します。
ログイン	Microsoft SQL Server に接続するために使用する Microsoft SQL Server ユーザー ID。 ユーザー ID が必要なのは、**`Windows 認証`**パラメーターが `False` に設定されている場合のみです。 ユーザー ID には ASCII 文字のみを使用してください。 「**ログイン**」フィールドでログイン ID を指定して Microsoft SQL Server エージェントを構成すると、エージェントは、このログイン ID を使用して Microsoft SQL Server に接続します。 **重要 :** エージェントの構成中に「**Windows 認証**」チェック・ボックスを選択した上で「**ログイン**」フィールドでログイン ID を指定すると、エージェントは Windows 認証を優先します。	いいえ	
パスワード	Microsoft SQL Server ユーザー ID のパスワード。 パスワードが必要なのは、**`Windows 認証`**パラメーターが `False` に設定されている場合のみです。 パスワードには ASCII 文字のみを使用してください。	いいえ	

表 182. **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)

パラメーター名	説明	必須フィールド	例
データベース・バージョン	SQL Server インスタンスのバージョン。	はい	SQL Server インスタンスのデータベースのバージョンは以下のとおりです。 • Microsoft SQL Server 2014 - 12.0.2000.8 • Microsoft SQL Server 2012 - 11.0.2100.60 • Microsoft SQL Server 2008 R2 - 10.50.1600.1 • Microsoft SQL Server 2008 - 10.0.1600.22 • Microsoft SQL Server 2005 - 9.0.1399.06
ホーム・ディレクトリー	SQL Server のインストール・ディレクトリー。	はい	デフォルトの Microsoft SQL Server 2005 インスタンスのデフォルト・ホーム・ディレクトリー・パスは、`C:¥Program Files ¥Microsoft SQL Server¥MSSQL` です。 Microsoft SQL Server 2005 の名前付きインスタンスのデフォルト・ホーム・ディレクトリー・パスの形式は `C:¥Program Files ¥Microsoft SQL Server¥MSSQL$` *instance_name* です (*instance_name* は Microsoft SQL Server インスタンス名です)。

表 *182.* **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)

パラメーター名	説明	必須フィールド	例
エラー・ログ・ファイル	SQL Server エラー・ログの完全修飾されたロケーションおよび名前。	はい	デフォルトの Microsoft SQL Server 2005 インスタンスのデフォルト・エラー・ログ・パスは C:¥Program Files ¥Microsoft SQL Server¥MSSQL¥ LOG ¥ERRORLOG です。 Microsoft SQL Server 2005 の名前付きインスタンスのデフォルト・エラー・ログ・パスの形式は、C:¥Program Files¥Microsoft SQL Server¥MSSQL$ *instance_name* ¥LOG¥ERRORLOG です (*instance_name* は Microsoft SQL Server インスタンス名です)。

表 182. **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)			
パラメーター名	**説明**	**必須フィールド**	**例**
Windows 認証	Windows 認証は、エージェント・サービスの構成に使用される Windows アカウントであり、デフォルトの構成オプションです。 「**Windows 認証**」チェック・ボックスを選択した場合、認証に Windows 資格情報が使用されます。 Microsoft SQL Server エージェントを Windows 認証で構成した場合、エージェント・サービスは、Microsoft SQL Server へのログオンに「**ローカル・システム・アカウント (Local System account)**」または「**次のアカウント**」を使用します。 • エージェント・サービスが「**ローカル・システム・アカウント (Local System account)**」を使用してログオンするように構成されている場合、エージェントは、NT AUTHORITY¥SYSTEM というユーザー ID を使用して Microsoft SQL Server にアクセスします。 • エージェント・サービスが「**次のアカウント**」を使用してログオンするように構成されている場合、エージェントは、それぞれのユーザー ID を使用して Microsoft SQL Server にアクセスします。 **要確認 :** 「**Windows 認証**」チェック・ボックスを選択しない場合は、「**ログイン**」パラメーターおよび「**パスワード**」パラメーターの値を指定する必要があります。これらのパラメーターを指定しないで「**データベース・サーバーのプロパティー**」ウィンドウの「**OK**」をクリックすると、エラー・メッセージがポップアップ・ウィンドウに表示され、エージェント構成が終了しません。 **重要 :** 「**Windows 認証**」チェック・ボックスを選択し、さらに「**ログイン**」フィールドでログイン ID を指定してエージェントを構成した場合、エージェントは Windows 認証を優先します。	いいえ	
長期継続データベース接続のサポート	長期継続データベース接続を有効化または無効化します。以下のデータ・セットは長期継続データベース接続を使用しません。 • MS SQL テキスト • MS SQL ファイル・グループの詳細 • MS SQL Server の要約	いいえ	

表 *182.* **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)			
パラメーター名	**説明**	**必須フィールド**	**例**
拡張パラメーター	任意の属性グループのデータ収集を無効にします。	いいえ	例: 「表の詳細」データ・セットのデータ収集を無効にするには、「**拡張パラメーター**」フィールドに `koqtbld` と入力します。 「データベース詳細」データ・セットのデータ収集を無効にするには、「**拡張パラメーター**」フィールドに `koqdbd` と入力します。 「表の詳細」データ・セットと「データベース詳細」データ・セットのデータ収集を無効にするには、「**拡張パラメーター**」フィールドに `koqtbld,koqdbd` と入力します。

表 182. **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)

パラメーター名	説明	必須フィールド	例

データベース	モニターするデータベースを選択するには、このパラメーターの値を指定します。SQL サーバー・インスタンスで使用可能なすべてのデータベースのモニターを有効にするには、「**データベース**」グループ領域の「**すべてのデータベースのモニター**」チェック・ボックスを選択します。「**すべてのデータベースのモニター**」チェック・ボックスはデフォルトで選択されています。
	
	特定のデータベースのモニターを有効または無効にするには、「**すべてのデータベースのモニター**」チェック・ボックスをクリアします。
	• 特定のデータベースをモニターをするには、リストから「**含む**」を選択し、リストの横のテキスト・フィールドにデータベースの 名前を指定します。
	• 特定のデータベースをモニター対象から除外するには、リストから「**除外**」を選択し、リストの横のテキスト・フィールドにデータベースの 名前を指定します。
	テキスト・フィールドを使用して、 モニターするデータベースをフィルターに 掛けます。
	データベース・フィルターを指定するには、最初に分離文字を選択する必要があります。分離文字は、データベース名またはデータベース式を別のデータベース名またはデータベース式と区別するための文字です。
	分離文字を選択する際は、データベース名およびデータベース式に、分離文字として選択した文字が含まれていないことを確認してください。データベース名またはデータベース式でワイルドカード文字を使用する場合、一般的に T-SQL 照会で使用されるワイルドカード文字 (%、_、[ ]、^、- など) は、使用しないでください。
	データベース・フィルターを指定する際は、以下の点に注意してください。
	• データベース名は、分離文字で始める必要があります。
	• データベース式は、2 つの分離文字で始める必要があります。
	データベース式は、T-SQL 照会の LIKE 部分で使用可能な有効な式です。ただし、データベース式を指定する場合は、T-SQL ESCAPE 節を使用することはできません。次のデータ・セットは、データベース・フィルターの影響を受けます: データベース詳細、データベースの要約、デバイス詳細、表の詳細、表の要約、ファイル・グループの詳細、追加データベースの詳細。
		いいえ	フィルターの例:
			ケース 1: % の使用
			例:
			`@@%m%`
			出力: データベース名に文字 m が含まれているデータベースがすべてフィルタリングされます。
			ケース 2: _ の使用
			例:
			`@@____`
			出力: 長さが 4 文字のデータベースがすべてフィルタリングされます。
			ケース 3: [] の使用
			例:
			`@@[m]___`
			出力: 名前が文字 m で始まり、長さが 4 文字のデータベースがすべてフィルタリングされます。
			ケース 4: [^] の使用
			例:
			`@@[^m]%`
			出力: データベース名が文字 *m* で始まるものを除き、(すべての長さの) すべてのデータベースがフィルターに掛けられます。

表 182. **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)

パラメーター名	説明	必須フィールド	例

データベース (続き)	**要確認 :**
	• 「**すべてのデータベースのモニター**」チェック・ボックスを選択しない場合は、「**データベース**」グループ領域にあるテキスト・フィールドに、モニターを有効または無効にする対象のデータベースのリストを指定する必要があります。「**すべてのデータベースのモニター**」チェック・ボックスの選択およびデータベース・リストの指定を行わずに、「**データベース・サーバーのプロパティー**」ウィンドウで「**OK**」をクリックすると、エラー・メッセージがポップアップ・ウィンドウに表示され、エージェント構成が終了しません。
	• 「**すべてのデータベースのモニター**」チェック・ボックスを選択し、かつ「**データベース**」グループ領域にあるテキスト・フィールドにモニター対象データベースを指定した場合は、「**すべてのデータベースのモニター**」の値が優先されます。テキスト・フィールドに指定したデータベース・リストは無視されます。
			ケース 5: 不正な入力
			例:
			`@%m%`
			出力: データベースはフィルタリングされません。
			ケース 6: デフォルト
			例: フィールドが空の場合 (照会が入力されていない)
			出力: すべてのデータベースがフィルタリングされます。
			ケース 7: 混合パターン
			例:
			`@@[m-t]_d%`
			出力: データベース名が文字 m、n、o、p、q、r、s、t のいずれかで始まり、その後ろに任意の 1 文字が続き、3 文字目に d が付くすべてのデータベースがフィルタリングされます。

日単位の頻度	この機能を使用して、表の詳細属性のデータを収集する頻度を定義します。指定できる値は、0 から 31 です。	いいえ	
週単位の頻度	この機能を使用して、表の詳細属性のデータを収集する特定の曜日を指定します。指定できる値は、0 から 7 です。	いいえ	
月単位の頻度	この機能を使用して、表の詳細属性のデータを収集する特定の日付を定義します。指定できる値は、1、2、3 などです。	いいえ	
収集開始時刻	収集開始時刻は HH:MM の形式で入力できます。 時刻の時に指定できるのは、0 から 23 までの値です。デフォルト値は 0 です。 時刻の分に指定できるのは、0 から 59 までの値です。デフォルト値は 0 です。	いいえ	

表 182. **「データベース・サーバーのプロパティー」**ウィンドウの構成設定の名前と説明 (続き)			
パラメーター名	**説明**	**必須フィールド**	**例**
表の詳細の連続収集	「表の詳細」データの連続バックグラウンド収集を行う場合は、この機能を使用します。 「**表の詳細の連続収集**」チェック・ボックスはデフォルトで選択されています。	いいえ	
2 つの連続収集間の間隔 (分単位)	2 つの収集間の間隔の時間を分単位で指定します。最小間隔時間は 3 分です。 「**2 つの連続収集間の間隔 (分単位)**」チェック・ボックスを選択するか、「スケジューリング」を使用して「表の詳細」データ・セットの連続収集を指定することができます。「**2 つの連続収集間の間隔 (分単位)**」チェック・ボックスを選択する場合は、収集の時間間隔を指定する必要があります。「スケジューリング」を使用して「表の詳細」データ・セットの収集を指定する場合、最小時間間隔は 1 日です。 連続する 2 つの収集の間のデフォルトの間隔は 3 分です。	いいえ	

エージェントは、データ収集の頻度が高い時間間隔に合わせてデータを収集します。例えば、データ収集の頻度 (日単位、週単位、月単位) をすべて指定した場合、エージェントは以下の条件に従ってデータ収集を開始します。

- 「日単位の頻度」が 7 以下の場合、「日単位の頻度」設定が選択され、「週単位の頻度」設定および「月単位の頻度」設定は無視されます。
- 「日単位の頻度」が 7 より大きい場合、「週単位の頻度」設定が選択され、「日単位の頻度」設定および「月単位の頻度」設定は無視されます。

要確認 : 「**表の詳細の連続収集**」チェック・ボックスを選択した場合、エージェントは、日単位、週単位、月単位の頻度による収集を行うのではなく、「**2 つの連続収集間の間隔 (分単位)**」フィールドに指定された間隔でデータを収集します。

Windows システムでのローカル環境変数の構成

ローカル環境変数を構成して、Microsoft SQL Server エージェントの動作を変更することができます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**アクション**」メニューから「**拡張**」>「**変数の編集**」をクリックします。
3. 「Monitoring Agent for Microsoft SQL Server: **ローカル変数設定のオーバーライド**」ウィンドウで、「**追加**」をクリックします。
4. 「**環境設定のオーバーライドを追加します**」ウィンドウで、変数および対応する値を入力します。

   注 : 構成可能な環境変数の完全なリストについては、550 ページの『ローカル環境変数』を参照してください。

非管理者ユーザーとしての実行

Microsoft SQL Server のモニター・エージェントを非管理者ユーザーとして実行することができます。

このタスクについて

Microsoft SQL Server エージェントは「domain users」グループの非管理者ユーザーとして実行することができます。

手順

1. Windows アプリケーション「Active Directory ユーザーとコンピュータ」を開始し、ドメイン・ユーザーを作成します。
   - 新規ユーザーが「*domain users*」グループのメンバーであることを確認します。
   - SQL Server が「*domain computers*」のメンバーであることを確認します。
2. 新たに作成したドメイン・ユーザーを「*SQL Server* ログイン」ユーザー・グループに追加します。このドメイン・ユーザーは、SQL Server に対する **sysadmin** SQL Server 役割の許可を保持しているか、『https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/sql_config_agent_grant_permission_sqlserver.htm』に記載された許可を保持している必要があります。
3. ドメイン管理者として SQL Server にログオンします。
4. Microsoft SQL Server エージェントがアクセスするすべてのドライブへの「**変更**」許可を付与します。すべてのサブディレクトリーに許可を伝搬するには、以下の手順を実行します。
   a)「**マイ コンピュータ**」にアクセスします。
   b) **ドライブ**を右クリックします。
   c)「**セキュリティ**」タブをクリックします。
   d) 新たに作成したユーザーを追加します。
   e) 新たに作成したユーザーに「**変更**」許可を付与します。
   f)「**OK**」をクリックします。 この手順ですべてのサブディレクトリーに許可を適用するには数分かかります。
5. Windows レジストリーを使用して、HKEY_LOCAL_MACHINE への読み取り権限を付与し、設定を伝搬します。設定を伝搬するには、以下のステップを実行します。
   a) HKEY_LOCAL_MACHINE ディレクトリーを右クリックして「**アクセス許可**」を選択します
   b) 新たに作成したユーザーを追加します。
   c) 新たに作成したユーザーを選択します。
   d)「**読み取り**」の「**許可**」チェック・ボックスを選択します。
   e)「**OK**」をクリックします。 この手順で HKEY_LOCAL_MACHINE ツリー全体に設定を伝搬するには数分かかります。
6. Windows レジストリーを使用して、以下のリストに従ってエージェント固有のレジストリー権限を付与します。
   - 32 ビットのオペレーティング・システムに 32 ビット・エージェントをインストールした場合は、KEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring ディレクトリーへの全アクセス権限を付与して、設定を伝搬します。
   - 64 ビットのオペレーティング・システムに 32 ビット・エージェントをインストールした場合は、HKEY_LOCAL_MACHINE¥SOFTWARE¥Wow6432Node¥Candle ディレクトリーへの全アクセス権限を付与して、設定を伝搬します。
   - 64 ビットのオペレーティング・システムに 64 ビット・エージェントをインストールした場合は、KEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring ディレクトリーへの全アクセス権限を付与して、設定を伝搬します。

   設定を伝搬するには、以下のステップを実行します。
   a) 全アクセス権限を持つディレクトリーを右クリックして「**アクセス許可**」を選択します。

b) 新たに作成したユーザーを追加します。

c) 新たに作成したユーザーを選択します。

d)「**フル コントロール**」の「**許可**」チェック・ボックスを選択します。

e)「**OK**」をクリックします。 この手順で KEY_LOCAL_MACHINE¥SOFTWARE¥IBMMonitoring ツリー全体に設定を伝搬するには数分かかります。

7. 新規ドメイン・ユーザーを「**Performance Monitor Users**」グループに追加します。
8. ドメイン・ユーザーが「ユーザー」グループのメンバーであることを確認します。
9. 非管理者ユーザーとして実行するには、Windows ディレクトリーに対する以下の許可を付与します。
   - 32 ビットのオペレーティング・システムに 32 ビット・エージェントがインストールされている場合は、OS_installation_drive:¥Windows¥system32 ディレクトリーへの読み取り/書き込み権限を付与します。
   - 64 ビットのオペレーティング・システムに 32 ビット・エージェントがインストールされている場合は、OS_installation_drive:¥Windows¥SysWOW64 ディレクトリーへの読み取り/書き込み権限を付与します。

   注 : Windows Server 2008、Windows Server 2008 R2、Windows Server 2012、Windows Server 2012 R2、Windows Server 2016 では、Windows ディレクトリーに対する権限は不要です。
10. 以下に示す SQL Server のデータ・ファイルおよびログ・ファイルに対する「**変更**」許可を付与します。
    - SQL Server データ・ファイルのデフォルト・パスは *SQLServer_root_dir*¥DATA です (*SQLServer_root_dir* は SQL Server インスタンスのルート・ディレクトリー)。例えば、SQL Server インスタンスのルート・ディレクトリーが C:¥Program Files¥Microsoft SQL Server ¥MSSQL.1¥MSSQL である場合、データ・ファイルのパスは C:¥Program Files¥Microsoft SQL Server¥MSSQL.1¥MSSQL¥DATA となります。
    - SQL Server ログ・ファイルのデフォルト・パスは *SQLServer_root_dir*¥LOG です (*SQLServer_root_dir* は SQL Server インスタンスのルート・ディレクトリー)。例えば、SQL Server インスタンスのルート・ディレクトリーが C:¥Program Files¥Microsoft SQL Server ¥MSSQL.1¥MSSQL である場合、ログ・ファイルのパスは C:¥Program Files¥Microsoft SQL Server¥MSSQL.1¥MSSQL¥LOG となります。
11. Candle_Home ディレクトリーへの全アクセス権限を付与します。デフォルトのパスは C:¥IBM¥ITM です。
12. ローカル・セキュリティー権限を適用します (567 ページの『ローカル・セキュリティー・ポリシーの許可』を参照)。
13. SQL Server を再始動して、ローカル・セキュリティー許可が適用されて有効になるようにします。
14. 以下のステップを実行して、SQL Server エージェント・サービスのログオン設定を非管理者ユーザーに変更します。

    a)「**スタート**」>「**管理ツール**」>「**サービス**」をクリックします。

    b)「**Monitoring Agent For SQL Server *instance_name* (Monitoring Agent For SQL Server instance_name)**」を右クリックして「**プロパティ**」をクリックします。「**SQL サービスのプロパティ (SQL Service Properties)**」ウィンドウが開きます。

    c)「**ログオン**」タブをクリックします。

    d)「**アカウント**」をクリックしてユーザー名を入力します。

    e)「**パスワード**」フィールドと「**パスワードの確認入力**」フィールドにパスワードを入力して、「**OK**」をクリックします。

    f)「**Monitoring Agent For SQL Server Collector *instance_name* (Monitoring Agent For SQL Server Collector instance_name)**」に対してもステップ b から e を繰り返します (*instance_name* は Microsoft SQL Server インスタンスの名前)。

ローカル・セキュリティー・ポリシーの許可

ローカル・セキュリティー・ポリシーは、システムとそのセキュリティー・ポリシーを管理します。ローカル・セキュリティー・ポリシーは、エージェントとそのエージェントのインストール先システムをセキ

ュアな状態に保つ上で重要な役割を果たします。このポリシーは、アクセス権限と許可をユーザーに付与することによって機能します。Microsoft SQL Server エージェントの場合は、ローカル・セキュリティー許可ポリシーに従って、以下の許可を必ずユーザーに付与します。

「サービスとしてログオン *(Log on as Service)*」許可

このタスクについて

「サービスとしてログオン (Log-on as service)」許可を付与するには、以下の手順を実行します。

手順

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル・ポリシー (Local Policies)**」をクリックして、リストを展開します。
3. 「**ユーザー権限の割り当て (User Rights Assignment)**」をクリックします。ユーザー権限のリストが開きます。
4. 「**サービスとしてログオン (Log-on as service)**」ポリシーをダブルクリックします。「**サービスとしてログオンのプロパティ**」ウィンドウが開きます。
5. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
6. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
7. 「**OK**」をクリックします。

「プログラムのデバッグ」許可

このタスクについて

「プログラムのデバッグ」許可を付与するには、Microsoft SQL Server エージェントで以下の手順を実行します。

手順

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル・ポリシー (Local Policies)**」をクリックして、リストを展開します。
3. 「**ユーザー権限の割り当て (User Rights Assignment)**」をクリックします。ユーザー権限のリストが開きます。
4. 「**プログラムのデバッグ**」ポリシーをダブルクリックします。「**プログラムのデバッグのプロパティ**」ウィンドウが開きます。
5. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
6. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
7. 「**OK**」をクリックします。

「認証後にクライアントを偽装」

このタスクについて

「認証後にクライアントを偽装」許可を付与するには、Microsoft SQL Server エージェントで以下の手順を実行します。

手順

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ 設定**」ウィンドウが開きます。
2. 「**ローカル・ポリシー (Local Policies)**」をクリックして、リストを展開します。
3. 「**ユーザー権限の割り当て (User Rights Assignment)**」をクリックします。ユーザー権限のリストが開きます。
4. 「**認証後にクライアントを偽装**」ポリシーをダブルクリックします。「**認証後にクライアントを偽装のプロパティ**」ウィンドウが開きます。
5. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
6. 「**選択するオブジェクト名を入力してください**」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
7. 「**OK**」をクリックします。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを構成するには、スクリプトを実行してプロンプトに応答する必要があります。

始める前に

エージェントを構成する前に、以下のタスクを必ず完了してください。

- ローカル環境変数の確認

このタスクについて

Microsoft SQL Server エージェントは複数インスタンス・エージェントです。エージェントの各インスタンスを手動で構成し開始する必要があります。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
   例:
   ```
   cd /opt/ibm/apm/agent/bin
   ```
2. 以下のコマンドを実行します。*instance_name* は、インスタンスに付ける名前です。
   ```
   ./mssql-agent.sh config instance_name
   ```
3. コマンド・プロンプトに次のメッセージが表示されたら、1 と入力して Enter キーを押します。
   ```
   Edit 'Monitoring Agent for MSSQL setting? [1=Yes, 2=No]
   ```
4. プロンプトが出されたら、構成パラメーターの値を指定します。
   構成パラメーターについては、『エージェントの構成パラメーター』を参照してください。
5. 以下のコマンドを実行して、エージェントを開始します。
   ```
   ./mssql-agent.sh start instance_name
   ```
6. 以下のコマンドを実行して、エージェントを停止します。
   ```
   ./mssql-agent.sh stop instance_name
   ```

Linux システムでのローカル環境変数の構成

Linux システムでローカル環境変数を構成して、Microsoft SQL Server エージェントの動作を変更することができます。

手順

1. 端末またはシステム・ファイル・マネージャーを起動して、ディレクトリーをエージェントのインストール・ディレクトリーに変更します。
   例:
   ```
   /opt/ibm/apm/agent
   ```
2. 以下のコマンドを実行して、エージェントを停止します。
   ```
   ./mssql-agent.sh stop instance_name
   ```
   ここで、*instance_name* はエージェント・インスタンス名です。
3. 次の構成ディレクトリーにある `.oq.environment` ファイルを開きます。
   例:
   ```
   install_dir/config
   ```
   ここで、*install_dir* はエージェントのインストール・ディレクトリーです。
4. 名前と値のペアの形式に従って、必要な環境変数を `.oq.environment` ファイルの末尾に追加します。
   `export VARIABLE_NAME=VARIABLE_VALUE`
   例:
   ```
   export KOQ_ODBC_DRIVER=ODBC Driver 17 for SQL Server
   ```
   注:
   - 構成可能な環境変数の完全なリストについては、550 ページの『ローカル環境変数』を参照してください。
   - 追加したカスタム変数は、エージェントをアップグレードすると 失われます。
5. このファイルを保存します。
6. 次のエージェント・インストール・ディレクトリーからエージェントを開始します。
   ```
   cd /opt/ibm/apm/agent/bin
   ```
   ```
   ./mssql-agent.sh start instance_name
   ```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルを使用し、エージェントまたはエージェントの複数インスタンスを構成できます。

始める前に

エージェントの複数のインスタンスを構成するには、 すべてのエージェント・インスタンスの構成詳細情報をサイレント応答ファイルで指定する必要があります。

このタスクについて

構成設定を変更するには、構成スクリプトを実行します。構成スクリプトを実行する前にサイレント応答ファイルを編集できます。

手順

エージェントを構成するには、以下のステップを実行します。

1. テキスト・エディターを起動し、次の場所にあるサイレント応答ファイルを開きます。

- Windows *install_dir*¥samples/mssql_silent_config.txt
- Linux *install_dir*/samples/mssql_silent_config.txt

ここで、*install_dir* はエージェントのインストール・ディレクトリーです。

例:

- Windows C:¥IBM¥APM¥samples¥mssql_silent_config.txt
- Linux /opt/ibm/apm/agent/samples/mssql_silent_config.txt

注: エージェントの構成パラメーターについては、555 ページの『エージェントの構成パラメーター』を参照してください。

2. コマンド・プロンプトを起動して、ディレクトリーを以下に変更します。
   - Windows
     ```
     cd install_dir¥bin
     ```
   - Linux
     ```
     cd install_dir/bin
     ```
3. 次のコマンドを実行します。
   - Windows
     ```
     mssql-agent.bat config install_dir¥samples¥mssql_silent_config.txt
     ```
   - Linux
     ```
     mssql-agent.sh config  instance_name install_dir/samples/mssql_silent_config.txt
     ```
4. エージェントを開始します。
   - Windows 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「開始」をクリックします。
   - Linux 次のコマンドを実行します。
     ```
     cd /opt/ibm/apm/agent/bin
     ```
     ```
     ./mssql-agent.sh start instance_name
     ```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

クラスター環境でのエージェントの実行

Windows Microsoft SQL Server エージェントをクラスター環境で構成できます。Microsoft SQL Server と Microsoft SQL Server エージェントの複数のインスタンスを単一ノード上で実行できます。

Microsoft SQL Server エージェントをインストールして構成したら、クラスター環境でエージェントを実行するために、以下のタスクを実行します。

- 環境変数の追加
- エージェント・サービスとコレクター・サービスの開始タイプの変更

- クラスター環境へのエージェントとコレクターの追加

クラスター環境は、以下の Microsoft SQL Server のバージョン向けにセットアップできます。

- Microsoft SQL Server 2005
- Microsoft SQL Server 2008
- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2012
- Microsoft SQL Server 2014
- Microsoft SQL Server 2016

重要 : Windows システムの場合、OS エージェントがインストールされているのと同じディレクトリーにエージェントをインストールする必要があります。各クラスター・ノードのノード・システム・ディスクにエージェントをインストールします。

環境変数の追加
各クラスター・ノードにインストールされているエージェントによって使用される環境変数を構成する必要があります。

このタスクについて
以下の環境変数に対して値を指定する必要があります。

- *CTIRA_HOSTNAME*: この変数は、Microsoft SQL Server エージェントの各インスタンスを構成するために使用されます。この変数の値は 31 文字に制限されています。また、この値はすべてのモニター・エージェントに共通です。この変数の値は、Application Performance ダッシュボードで当該クラスターのすべてのモニター・エージェントにナビゲートするクラスター名に設定してください。
- *CTIRA_NODETYPE*: この変数は、エージェントを識別するために使用されます。デフォルトでは、この変数の値は Microsoft SQL Server エージェントの **MSS** に設定されています。
- *CTIRA_SUBSYSTEMID*: この変数は、Microsoft SQL Server エージェントの複数のインスタンスを区別するために使用されます。デフォルトでは、この変数の値は Microsoft SQL Server エージェントの **Microsoft SQL Virtual Server** に設定されています。
- *COLL_HOME*: この変数は、共有ロケーションで構成ファイルを使用する属性グループのデータの収集およびログ・ファイルの保管を行うために使用されます。この変数の値は、*X*:¥shared-location に設定してください。ここで、*X* は、クラスター・ノードからアクセス可能な共有ドライブです。例えば、MS SQL 表の詳細属性グループまたは MS SQL エラー・イベントの詳細属性グループの構成設定を定義する場合に、*COLL_HOME* 変数の値を設定します。
- *CTIRA_HIST_DIR*: この変数は、共有ディスク・ディレクトリーのパスを指定するために使用されます。Microsoft SQL Server エージェントのヒストリーがモニター・エージェントで保管されるように構成されている場合は、エージェントの各インスタンスを、共有ディスク・ディレクトリーを参照する共通の *CTIRA_HIST_DIR* 変数で構成する必要があります。

  要確認 : ヒストリーが Cloud APM サーバーで保管される場合は、*CTIRA_HIST_DIR* 変数の値を指定する必要はありません。Cloud APM サーバーでヒストリーを保管すると、そのサーバーの負荷が増大します。

これらの変数を追加するには、565 ページの『Windows システムでのローカル環境変数の構成』で説明しているステップを参照してください。

次のタスク
573 ページの『エージェント・サービスとコレクター・サービスの開始タイプの変更』で説明しているステップを実行して、エージェント・サービスとコレクター・サービスの開始タイプを「**手動**」に変更します。

エージェント・サービスとコレクター・サービスの開始タイプの変更

デフォルトでは、エージェント・サービスとコレクター・サービスの開始タイプは「**自動**」になっています。クラスター・リソースがモニター・エージェントの開始と停止を制御できるように、エージェント・サービスとコレクター・サービスの開始タイプを「**手動**」に変更します。

手順

エージェント・サービスの開始タイプを変更するには、以下のステップを実行します。

1. 「**スタート**」>「**ファイル名を指定して実行**」をクリックし、コマンド services.msc を入力して「**OK**」をクリックします。
2. エージェントを右クリックし、「**プロパティ**」をクリックします。
3. 「**Monitoring Agent for Microsoft SQL Server Properties**」ウィンドウで、「**スタートアップの種類**」リストから「**手動**」を選択し、「**適用**」をクリックしてから「**OK**」をクリックします。

次のタスク

- 同じ手順を使用して、コレクター・サービスの開始タイプを「**手動**」に変更します。
- 573 ページの『クラスター環境へのエージェントとコレクターの追加』で説明しているステップを実行して、エージェントとコレクターをクラスター環境に追加します。

クラスター環境へのエージェントとコレクターの追加

エージェントとコレクターをクラスター環境に追加する必要があります。

手順

1. 「**スタート**」>「**コントロール パネル**」>「**管理ツール**」>「**フェールオーバー クラスタ管理**」をクリックします。
2. 「**フェールオーバー クラスタ管理**」を展開します。
3. 「**サービスとアプリケーション**」を展開して、構成する SQL インスタンスを右クリックします。
4. 「**リソースの追加**」>「**汎用サービス**」をクリックします。「新しいリソース ウィザード」が開きます。
5. 「サービスの選択」ページで、サービス名を選択して、「**次へ**」をクリックします。

   Windows サービス名の例:

   - `Monitoring Agent for Microsoft SQL Server: SQLTEST#INSTANCE1`
   - `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST#INSTANCE1`
   - `Monitoring Agent for Microsoft SQL Server: SQLTEST2#INSTANCE2`
   - `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST2#INSTANCE2`
6. 「確認」ページで詳細を確認してから、「**次へ**」をクリックします。
7. 「概要」ページで「**終了**」をクリックします。これで、Microsoft SQL Server エージェントが追加されます。

   要確認 : 同じステップを使用して、コレクターをクラスター環境に追加します。
8. エージェントをオンラインにするには、エージェントを右クリックして、「**このリソースをオンラインにする**」をクリックします。
9. コレクターをオンラインにするには、コレクターを右クリックして、「**このリソースをオンラインにする**」をクリックします。

タスクの結果

これで、Microsoft SQL Server エージェントがクラスター環境で実行されます。

要確認 : エージェントを再度構成する場合は、最初にエージェントとコレクターをオフラインにするか、エージェントとコレクターが実行されているノードでエージェントの変数を編集する必要があります。 エージェントの構成が完了したら、エージェントとコレクターをオンラインに戻してください。

クラスター・ユーティリティーを使用したエージェントの構成

Windows クラスター・ユーティリティーを使用して、クラスター環境内のクラスター・グループに複数の Microsoft SQL Server エージェント・インスタンスを追加できます。

クラスター・ユーティリティーは、各 Microsoft SQL Server エージェント・インスタンスのエージェント・サービスとコレクター・サービスを汎用サービス・リソースとしてクラスター・グループに自動的に追加します。クラスター・ユーティリティーを使用して、以下のタスクを実行できます。

- クラスターへの SQL Server エージェント・インスタンスの追加
- クラスター内の既存の SQL Server エージェント・インスタンスの更新
- クラスターからの SQL Server エージェント・インスタンスの削除

クラスター・ユーティリティーを使用するための前提条件

ご使用のシステム環境が、クラスター・ユーティリティーを実行するための前提条件を満たしていることを確認する必要があります。

以下の前提条件が満たされていることを確認してください。

- クラスター環境内に少なくとも 1 つのグループがあるコンピューターでクラスター・ユーティリティーを実行すること。
- クラスター内のすべてのノードに対してリモート・レジストリー・サービスを開始すること。
- ユーザーがクラスター・ユーティリティーにアクセスするためのクラスター・マネージャー権限を持っていること。
- エージェントとコレクターのサービス名は、すべてのクラスター・ノードで同じであること。

  例えば、エージェント・サービス名が Monitoring Agent for Microsoft SQL Server: SQLTEST#INSTANCE1 であり、コレクター名が Monitoring Agent for Microsoft SQL Server: Collector SQLTEST#INSTANCE1 である場合、同じサービス名がクラスターのすべてのノード上に存在する必要があります。

クラスターへの Microsoft SQL Server エージェント・インスタンスの追加

クラスター・ユーティリティーを使用して、クラスター環境内のクラスター・グループに Microsoft SQL Server エージェント・インスタンスを追加できます。

手順

1. ユーティリティーを実行するためには、以下のステップのいずれかを実行します。
   - 64 ビット・エージェントの場合は、*candle_home*¥TMAITM6_x64 ディレクトリーに移動します。
   - 32 ビット・エージェントの場合は、*candle_home*¥TMAITM6 ディレクトリーに移動します。
2. クラスター・ユーティリティーを実行するために、KoqClusterUtility.exe をダブルクリックします。
3. 「**使用可能な SQL Server エージェント・インスタンス (SQL Server Agent Instances Available)**」領域で、Microsoft SQL Server エージェント・インスタンスを選択し、「**追加**」をクリックします。
4. 「**クラスター・グループ名の選択 (Select cluster group name)**」ウィンドウで、クラスター・グループを選択します。

   選択するクラスター・グループは、Microsoft SQL Server エージェントによってモニターされる SQL Server インスタンスでなければなりません。
5. 「**共有ロケーションのパスの選択 (Select Path for Shared Location)**」ウィンドウで、エージェント・ログとコレクター・ログの保管先のパスにナビゲートします。

   パスを選択しない場合、デフォルトで、エージェント・ログとコレクター・ログの保管用に CANDLEHOME/TMAITM6(_x64)/logs の場所が選択されます。
6. Microsoft SQL Server エージェント・インスタンスをクラスター環境に追加するために、「**OK**」をクリックします。

   クラスター・ユーティリティーのアクティビティー ・ログが「**ヒストリー**」ペインに表示されます。

クラスター内の既存の Microsoft SQL Server エージェント・インスタンスの更新

クラスター・ユーティリティーを 使用して、クラスター内の SQL Server インスタンスに対して、エージェント・ログとコレクター・ログが保管される場所を更新できます。

手順

1. 既存の Microsoft SQL Server エージェント・インスタンスを更新するために、「**クラスター・ユーティリティー**」ウィンドウを開きます。
2. 「**構成されている SQL Server エージェント・インスタンス (SQL Server Agent Instances Configured)**」領域で、Microsoft SQL Server エージェント・インスタンスを選択し、「**更新**」をクリックします。
3. 「**共有ロケーションのパスの設定 (Set Path for Shared Location)**」ウィンドウで、エージェント・ログとコレクター・ログの保管先のパスにナビゲートします。

   パスを選択しない場合、エージェント・ログとコレクター・ログは、クラスター内の Microsoft SQL Server エージェント・インスタンスの追加中に設定された場所に保管されます。
4. 「**OK**」をクリックします。

   クラスター・ユーティリティーのアクティビティー ・ログが「**ヒストリー**」ペインに表示されます。

クラスターからの Microsoft SQL Server エージェント・インスタンスの削除

クラスター・ユーティリティーを 使用して、クラスター・グループから Microsoft SQL Server エージェント・インスタンスを削除できます。

手順

1. 「**クラスター・ユーティリティー**」ウィンドウを開きます。
2. 「**構成されている SQL Server エージェント・インスタンス (SQL Server Agent Instances Configured)**」領域で、Microsoft SQL Server エージェント・インスタンスを選択し、「**削除**」をクリックします。
3. 「**アクションを確認してください**」ダイアログ・ボックスで、クラスターから Microsoft SQL Server エージェント インスタンスを削除するには、「**はい**」をクリックします。

   クラスター・ユーティリティーのアクティビティー ・ログが「**ヒストリー**」ペインに表示されます。

ERRORLOG ファイルの複数の照合の構成

Application Performance Management バージョン 8.1.4.0.4 用の Microsoft SQL Server エージェント バージョン 06.31.17.00 以降では、ERRORLOG ファイルで複数照合をサポートしています。「**問題の詳細**」属性グループで、ERRORLOG ファイルで複数照合を構文解析するようにエージェントを構成できます。ただし、「**エラー・イベントの詳細**」属性グループには ERRORLOG での複数照合を適用できません。

始める前に

エージェントの複数の照合を構成するには、エージェントがインストール済みであることを確認します。

このタスクについて

デフォルトの照合は英語です。他の言語の SQL Server の場合、エージェントは、構成ファイル `koqErrConfig.ini` の照合に基づいて ERRORLOG ファイルを構文解析します。そのため、`koqErrConfig.ini` ファイル内で使用されている照合を追加する必要があります。

手順

エージェントの複数の照合を構成するには、以下の手順を実行します。

1. エージェント・ディレクトリー *agent_directory* に移動します。

   Windows

   - 64 ビット・エージェントの場合、*agent_directory* は *Agent_home*¥TMAITM6_x64 です。
   - 32 ビット・エージェントの場合、*agent_directory* は *Agent_home*¥TMAITM6 です。

   Linux

- 64 ビット・エージェントの場合、*agent_directory* は *Agent_home*/TMAITM6_x64 です。

ここで、*Agent_home* はエージェント・インストール・ディレクトリーです。

2. 構成ファイル koqErrConfig.ini を開きます。
3. ファイルの末尾に移動し、新しい照合を追加します。
例えば、フランス語の照合を有効にするには、koqErrConfig.ini ファイルの末尾に **name-value** の組の形式で以下の照合設定を追加します。

```
[French]
Error = Erreur :
Severity = Gravité :
State = État :
```

注: 照合の例については *agent_directory*¥koqErrConfigSample.ini を参照してください。

構文の内容は次のとおりです。 Windows

- 64 ビット・エージェントの場合、*agent_directory* は *Agent_home*¥TMAITM6_x64 です。
- 32 ビット・エージェントの場合、*agent_directory* は *Agent_home*¥TMAITM6 です。

Linux

- 64 ビット・エージェントの場合、*agent_directory* は *Agent_home*/TMAITM6_x64 です。

ここで、*Agent_home* はエージェント・インストール・ディレクトリーです。

ターゲットの照合が koqErrConfigSample.ini にない場合は、ERRORLOG ファイルから照合キーワード値を判別できます。
koqErrConfig.ini 内の照合設定を構成するときには、以下の照合形式に従います。

```
[Section_name]
Error = Error_value
Severity = Severity_value
State = State_value
```

表の見方

- *Section_name* は SQL Server 照合名です。照合名は、必ず左大括弧「**[**」と右大括弧「**]**」で囲んでください。
- *Error_value* は、ターゲット照合の ERRORLOG ファイルにある、対応するエラー・キーワードです。
- *Severity_value* は、ターゲット照合の ERRORLOG ファイルにある、対応する重大度キーワードです。
- *State_value* は、ターゲット照合の ERRORLOG ファイルにある、対応する状態キーワードです。

重要: キーワードの値は、特殊文字も含めて、ERRORLOG ファイル内のキーワード値と同じでなければなりません。

4. 構成ファイル koqErrConfig.ini を保存します。

エージェントの再始動は不要です。

構成ファイル koqErrConfig.ini を使用できない場合、または構成ファイル koqErrConfig.ini が空である場合は、デフォルトの重大度レベルを超えていると、デフォルトの照合が英語のエラー・メッセージとして ERRORLOG ファイルに示されます。

構成ファイル koqErrConfig.ini が正しく構成されている場合は、デフォルトの重大度レベルを超えていると、対応するエラー・メッセージが ERRORLOG ファイルに示されます。

デフォルトの重大度レベルは 17 です。

重要: koqErrConfig.ini ファイルに対する変更は、エージェントをアップグレードすると失われるため、必ずバックアップを行ってからエージェントをアップグレードしてください。

次のタスク
照合設定の結果が Application Performance Management ダッシュボードの「**エラー・ログ・アラート**」ウィジェットまたは「**問題の詳細**」属性グループに反映されていることを確認します。

MongoDB のモニターの構成

Monitoring Agent for MongoDB にはインスタンス名が必要です。エージェント・インスタンスを手動で構成して開始する必要があります。MongoDB エージェント は、ローカル・モニターおよびリモート・モニターをサポートします。リモート・モニターとローカル・モニターの両方について、MongoDB エージェントの構成のための以下の前提条件を参照してください。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件の最新情報については、MongoDB エージェントの Software Product Compatibility Reports (SPCR) を参照してください。
- MongoDB エージェントを構成するユーザーが、すべての属性のデータを収集するために必要な役割を持っていることを確認します。
  - MongoDB データベース・バージョン 2.4 とバージョン 2.6 上にエージェントを構成するには、clusterAdmin、readAnyDatabase、および dbAdminAnyDatabase の各役割がユーザーに割り当てられていることが必要です。
  - MongoDB データベース・バージョン 3.x および 4.x 上にエージェントを構成するには、clusterMonitor、readAnyDatabase、および dbAdminAnyDatabase の各役割がユーザーに割り当てられていることが必要です。

  これらのユーザー役割が必要な属性グループについては、578 ページの表 183 を参照してください。
- 既存のユーザーを使用するか、管理データベースでユーザーを作成してください。

  重要 : ユーザーを作成して必要な役割をユーザーに付与する前に、MongoDB データベースに接続し、データベースを管理データベースに変更する必要があります。mongod または mongos プロセスが認証モードで実行されている場合、MongoDB データベースに接続するために必要な資格情報を入力します。

  1. 以下のコマンドを実行して、MongoDB データベースに接続します。

     ```
     mongo IP:port
     ```

     表の見方

     - *IP* は、mongod または mongos プロセスの IP アドレスです
     - *port* は、mongod または mongos プロセスのポート番号です
  2. 以下のように、データベースを管理データベースに変更します。

     ```
     use admin
     ```
  3. 以下のいずれかのコマンドを実行し、MongoDB の管理データベースでユーザーを追加して、必要な役割をユーザーに割り当てます。
     - MongoDB データベース・バージョン 2.4 の場合は、以下のコマンドを実行します。

       ```
       db.addUser({ user: "username", pwd: "password", roles: [ 'clusterAdmin',
       'readAnyDatabase', 'dbAdminAnyDatabase' ] })
       ```
     - MongoDB データベース・バージョン 2.6 の場合は、以下のコマンドを実行します。

       ```
       db.createUser({user: "username", pwd: "password", roles:
       [ 'clusterAdmin', 'readAnyDatabase', 'dbAdminAnyDatabase' ] })
       ```
     - MongoDB データベース・バージョン 3.x および 4.x の場合は、以下のコマンドを実行します。

       ```
       db.createUser({user: "username", pwd: "password", roles:
       [ 'clusterMonitor', 'readAnyDatabase', 'dbAdminAnyDatabase' ] })
       ```
  4. 以下のコマンドを実行して、管理データベースにユーザーが追加されたことを確認します。

db.auth("*username*", "*password*")

戻りコード **1** は、ユーザーが追加されたことを示します。戻りコード **0** は、ユーザーの追加に失敗したことを示します。

以下の表に、ユーザー役割およびそれらのユーザー役割が必要となる属性に関する情報を示します。

表 *183.* 属性グループおよび必要なユーザー役割

役割	**MongoDB データベース・バージョン**	**属性グループ**
dbAdminAnyDatabase	2.x、3.x、4.x	応答時間
readAnyDatabase	2.x、3.x、4.x	• Mongod リスト • 一般シャード情報 • コレクション・ストレージ • データベース名 • シャードの詳細 • コレクション・ストレージの詳細
clusterAdmin	2.x、3.x、4.x	• Mongo インスタンス情報 • Mongo インスタンス入出力情報 • APMUI の MII コピー 1 • APMUI の MII コピー 2 • Mongo インスタンス・データベース・ロック • ロック • MongoDB ロック • WiredTiger の詳細 • MMAPv1 の詳細
clusterMonitor	2.x、3.x、4.x	• Mongo インスタンス情報 • Mongo インスタンス入出力情報 • APMUI の MII コピー 1 • APMUI の MII コピー 2 • Mongo インスタンス・データベース・ロック • ロック • MongoDB ロック • WiredTiger の詳細 • MMAPv1 の詳細

- MongoDB サーバーのリモート・モニターの場合は、次の 2 つの前提条件を参照してください。
  1. MongoDB エージェントには、リモート側で MongoDB サーバーから情報を収集するために mongo シェルが必要であるため、MongoDB エージェントがインストールおよび構成されているシステムに MongoDB サーバーのインスタンスが必要です。エージェント・マシン上の MongoDB サーバーの mongo シェルは、モニター操作のためにリモート側の MongoDB サーバーに接続するのに使用されます。

2. エージェントをホストするシステムの /etc/hosts ファイルに、リモート・マシンの項目がある。

このタスクについて

管理対象システム名には、指定したインスタンス名が含まれます。例えば、インスタンス名を *instance_name*:*host_name*:*pc* (*pc* はエージェントの 2 文字の製品コード) と指定することができます。管理対象システム名は最大で 32 文字です。 インスタンス名は 28 文字 (ホスト名の字数は除く) まで入力できます。例えば、インスタンス名に Mongo2 を指定すると、管理対象システム名は Mongo2:hostname:KJ になります。

重要 : 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが完全には表示されません。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

要確認 :

- エージェントがデータを正しく収集できるようにするには、スーパー (root) ユーザーとしてエージェントを開始するか、エージェントを開始する場合と mongod プロセスを開始する場合に同じユーザー ID を使用します。
- MongoDB がクラスターとして実行されている環境では、ルーター・プロセスが実行されているのと同じコンピューターにエージェントをインストールしてください。同じコンピューターでエージェントを構成します。その際、そのコンピューターの IP アドレスとポート番号を使用し、セットアップの **TYPE** に 1 を指定します。
- MongoDB が認証モードでクラスターとして実行されている環境では、必ずクラスター内のすべてのシャードで、必要な権限を持つ同じユーザー ID を追加します。

デフォルト設定を使用するか、サイレント応答ファイルを編集するか、プロンプトに応答して、エージェントを構成できます。

デフォルト設定を使用したエージェントの構成

標準的な環境の場合は、デフォルトの設定を使用してエージェントを構成します。エージェント構成のデフォルト設定を使用した場合、エージェントは認証モードで実行されません。

手順

1. 次のコマンドを実行します。
   install_dir/bin/mongodb-agent.sh config *instance_name install_dir*/samples/mongodb_silent_config.txt

   表の見方

   - *instance_name* は、固有のアプリケーション・インスタンスに指定する名前です。
   - *install_dir* は MongoDB エージェント のインストール・ディレクトリーです。

   デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
2. 以下のコマンドを実行して、エージェントを開始します。
   install_dir/bin/mongodb-agent.sh start *instance_name*

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して構成パラメーターに異なる値を指定し、エージェントを構成できます。

始める前に

認証モードで MongoDB データベースを実行するには、必ず MongoDB データベースに対する clusterAdmin、readAnyDatabase、および dbAdminAnyDatabase の役割を持つユーザーでエージェントを構成してください。

手順

1. テキスト・エディターで、次のパスにあるサイレント応答ファイルを開きます。
   install_dir/samples/mongodb_silent_config.txt.
2. **TYPE** パラメーターに対して以下のいずれかの値を入力します。
   - 1: クラスターの場合
   - 2: レプリケーション・セットの場合
   - 3: スタンドアロン・インスタンスの場合

   デフォルトでは、エージェントはクラスターをモニターします。
3. **PORT** パラメーターには、モニター対象の MongoDB レプリケーション・セットの MongoDB クラスターまたは mongod インスタンスのルーターのポート番号を指定します。

   要確認 : ポート番号を指定しなかった場合、デフォルト・インターフェースでアクティブになっている適切な MongoDB プロセスのポート番号がエージェントによって自動的に検出されます。MongoDB プロセスがデフォルト・インターフェースでアクティブになっていない 場合、セカンダリー・インターフェースでアクティブになっている 適切な MongoDB プロセスのポート番号がエージェントによって選択されます。
4. **HOST** パラメーターに、MongoDB ホスト・システムの IP アドレスを指定します。

   要確認 : IP アドレスを指定しなかった場合、デフォルト・インターフェースでアクティブになっている適切な MongoDB プロセスの IP アドレスがエージェントによって自動的に検出されます。MongoDB プロセスがデフォルト・インターフェースでアクティブになっていない 場合、セカンダリー・インターフェースでアクティブになっている 適切な MongoDB プロセスの IP アドレスがエージェントによって検出されます。
5. **AUTHENTICATION** パラメーターには、mongoDB を認証モードで実行することを指示する場合に、YES を指定します。デフォルト値は NO であり、これはエージェントを認証モードで実行しないことを指示します。

   要確認 : MongoDB データベースが認証モードで実行されている場合、MongoDB エージェントもいずれの MongoDB クライアントも、資格情報なしで MongoDB データベースに接続できません。認証モードで実行されているデータベースに接続するには、**AUTHENTICATION** パラメーターに YES を指定します。

   YES を指定した場合、以下のステップを実行します。

   a) **User Name** パラメーターにルーターまたは mongod インスタンスのユーザー名を指定します。このユーザーに必要最低限の役割が割り当てられていることを確認してください。ユーザー役割については、578 ページの表 183 を参照してください。

   b) **Password** パラメーターにはパスワードを指定します。
6. mongodb_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
   install_dir/bin/mongodb-agent.sh config *instance_name install_dir*/samples/mongodb_silent_config.txt

   表の見方

   - *instance_name* は、インスタンスに指定する名前です。
   - *install_dir* は MongoDB エージェント のインストール・ディレクトリーです。

7. 以下のコマンドを実行して、エージェントを開始します。
   install_dir/bin/mongodb-agent.sh start *instance_name*

   重要 : エージェントを V1.0.0.9 以降にアップグレードし、エージェントを認証モードで実行する場合は、エージェントを再構成してユーザー名とパスワードを指定する必要があります。データを収集するには、構成後にエージェントを停止して再始動する必要があります。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

プロンプトへの応答によるエージェントの構成

カスタム設定を使用してエージェントを構成するには、スクリプトの実行中にプロンプトが出されたときに構成パラメーターの値を指定します。

手順

1. 次のコマンドを実行します。
   install_dir/bin/mongodb-agent.sh config *instance_name*

   表の見方

   - *instance_name* は、インスタンスに指定する名前です。
   - *install_dir* は MongoDB エージェント のインストール・ディレクトリーです。
2. **TYPE** パラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、以下の値のいずれかを指定してから、Enter キーを押します。

   - 1: クラスターの場合
   - 2: レプリケーション・セットの場合
   - 3: スタンドアロン・インスタンスの場合

   デフォルトでは、エージェントはクラスターをモニターします。
3. **PORT** パラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、MongoDB クラスターのルーターまたはモニター対象の MongoDB 複製セットの mongod インスタンスのポート番号を指定してから、Enter キーを押します。

   要確認 : ポート番号を指定しなかった場合、デフォルト・インターフェースでアクティブになっている適切な MongoDB プロセスのポート番号がエージェントによって自動的に検出されます。MongoDB プロセスがデフォルト・インターフェースでアクティブになっていない 場合、セカンダリー・インターフェースでアクティブになっている 適切な MongoDB プロセスのポート番号がエージェントによって選択されます。
4. **HOST** パラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、MongoDB ホスト・システムの IP アドレスを指定してから、Enter キーを押します。

   要確認 : IP アドレスを指定しなかった場合、デフォルト・インターフェースでアクティブになっている適切な MongoDB プロセスの IP アドレスがエージェントによって自動的に検出されます。MongoDB プロセスがデフォルト・インターフェースでアクティブになっていない 場合、セカンダリー・インターフェースでアクティブになっている 適切な MongoDB プロセスの IP アドレスがエージェントによって検出されます。
5. **AUTHENTICATION** パラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、エージェントが認証モードで実行されているかどうかを指定します。

   デフォルト値は NO であり、これはエージェントを認証モードで実行しないことを指示します。mongoDB が認証モードで実行されていることを指定する場合、YES を指定します。

要確認 : MongoDB データベースが認証モードで実行されている場合、MongoDB エージェントもいずれの MongoDB クライアントも、資格情報なしで MongoDB データベースに接続できません。認証モードで実行されているデータベースに接続するには、**AUTHENTICATION** パラメーターに YES を指定します。

YES を指定した場合、以下のステップを実行します。

a) **User Name** パラメーターにルーターまたは mongod インスタンスのユーザー名を指定します。このユーザーに必要最低限の役割が割り当てられていることを確認してください。ユーザー役割については、578 ページの表 183 を参照してください。

b) **Password** パラメーターにはパスワードを指定します。

6. 以下のコマンドを実行して、エージェントを開始します。
   install_dir/bin/mongodb-agent.sh start *instance_name*

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』 を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

MySQL のモニターの構成

Monitoring Agent for MySQL にはインスタンス名と MySQL サーバー・ユーザー資格情報が必要です。 最初のエージェント・インスタンスを作成した後に構成設定を変更できます。

始める前に

- エージェントの実行のために MySQL データベースにユーザーが作成されていることを確認します。このユーザーには、モニター対象の MySQL データベースに対する特定の特権は不要です。
- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the MySQL エージェント を参照してください。

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は最大で 32 文字です。 指定するインスタンス名は 28 文字 (ホスト名の字数は除く) まで入力できます。例えば、インスタンス名に MySQL2 を指定すると、管理対象システム名は MySQL2:hostname:SE になります。

重要 : 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが完全には表示されません。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システム上でエージェントを構成できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで以下の手順を実行します。

a)「**Monitoring Agent for MySQL**」テンプレートをダブルクリックします。

b)「**Monitoring Agent for MySQL**」ウィンドウで、インスタンス名を指定して「**OK**」をクリックします。

3.「**Monitoring Agent for MySQL**」ウィンドウで以下の手順を実行します。

a)「**IP アドレス**」フィールドに、リモートでモニターする MySQL サーバーの IP アドレスを入力します。 モニター対象のサーバーにエージェントがインストールされている場合は、デフォルト値を保持します。

b)「**JDBC ユーザー名**」フィールドに、MySQL サーバーのユーザー名を入力します。デフォルト値は root です。

c)「**JDBC パスワード**」フィールドに JDBC ユーザーのパスワードを入力します。

d)「**JDBC パスワードの確認 (Confirm JDBC password)**」フィールドに、再度パスワードを入力します。

e)「**JDBC JAR ファイル (JDBC Jar File)**」フィールドで「**参照**」をクリックし、MySQL コネクターの Java ファイルを含むディレクトリーを探して選択します。

f)「**次へ**」をクリックします。

g)「**JDBC ポート番号**」フィールドで JDBC サーバーのポート番号を指定します。
デフォルトのポート番号は 3306 です。

h)「**Java トレース・レベル**」リストから Java のトレース・レベルを選択します。
デフォルト値は `Error` です。

i)「**OK**」をクリックします。
「**IBM Performance Management**」ウィンドウにインスタンスが表示されます。

4.「**Monitoring Agent for MySQL**」インスタンスを右クリックして「**開始**」をクリックします。

要確認 : 再度エージェントを構成するには、「**IBM Performance Management**」ウィンドウで以下の手順を実行します。

a. 構成するエージェント・インスタンスを停止します。

b.「**Monitoring Agent for MySQL**」インスタンスを右クリックして「**再構成**」をクリックします。

c. ステップ 3 および 4 を繰り返します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Linux システムでのエージェントの構成

構成スクリプトを実行して、Linux システム上でエージェントを構成します。

手順

1. 次のコマンドを実行します。

```
install_dir/bin/mysql-agent.sh config instance_name
```

ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は MySQL エージェントのインストール・ディレクトリーです。

2. 以下のパラメーターの値を入力するようにプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、別の値を指定して Enter キーを押します。

- IP アドレス
- JDBC ユーザー名
- JDBC パスワード
- 再入力: JDBC パスワード (Re-type:JDBC password)

- JDBC JAR ファイル
- JDBC ポート番号 (デフォルトのポート番号: 3306)
- Java トレース・レベル (デフォルト値: Error)

構成パラメーターについては、584 ページの『サイレント応答ファイルを使用したエージェントの構成』を参照してください。

3. 以下のコマンドを実行して、エージェントを開始します。

```
install_dir/bin/mysql-agent.sh start instance_name
```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

構成スクリプトを実行するときに、プロンプトに応答せずにエージェントを構成するには、サイレント応答ファイルを使用します。サイレント応答ファイルは、Windows と Linux のいずれのシステム上でのエージェントの構成にも使用できます。

このタスクについて

サイレント応答ファイルには、構成パラメーターがあります。応答ファイル内のパラメーター値を編集し、構成スクリプトを実行して、エージェント・インスタンスを作成し、構成値を更新します。

手順

1. テキスト・エディターで、次のパスにある応答ファイルを開きます。

   Linux *install_dir*/samples/mysql_silent_config.txt

   Windows *install_dir*¥samples¥mysql_silent_config.txt

   ここで、*install_dir* は MySQL エージェント のインストール・ディレクトリーです。
2. 応答ファイルで、以下のパラメーターの値を指定します。
   - **Server Name** パラメーターに、リモートでモニターする MySQL サーバーの IP アドレスを指定します。指定しない場合は、デフォルト値の localhost のままにします。
   - **JDBC user name** パラメーターでは、root のデフォルト・ユーザー名の値を保持するか、INFORMATION_SCHEMA テーブルの表示権限を持つユーザーの名前を指定します。
   - **JDBC password** パラメーターに、JDBC ユーザー・パスワードを入力します。
   - **JDBC Jar File** パラメーターでは、Java jar ファイルの MySQL コネクターのデフォルト・パスが正しい場合は、そのパスを保持します。 そうでない場合は、正しいパスを入力します。コネクターは以下のデフォルト・パスにあります。

     Linux /usr/share/java/mysql-connector-java.jar

     Windows C:¥Program Files (x86)¥MySQL¥Connector J 5.1.26¥mysql-connector-java-5.1.26-bin.jar
   - **JDBC port number** パラメーターでは、デフォルトのポート番号 3306 を保持するか、別のポート番号を指定します。
   - **Java trace level** パラメーターでは、デフォルト値 Error を保持するか、IBM サポートの指示に従って、別のレベルを指定します。
3. 応答ファイルを保存して閉じ、以下のコマンドを実行してエージェント構成設定を更新します。

   Linux *install_dir*/bin/mysql-agent.sh config *instance_name* *install_dir*/samples/mysql_silent_config.txt

Windows *install_dir*¥BIN¥mysql-agent.bat config *instance_name install_dir* ¥samples¥mysql_silent_config.txt

ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は MySQL エージェントのインストール・ディレクトリーです。

重要 : サイレント応答ファイルの絶対パスを必ず指定してください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

NetApp ストレージのモニターの構成

Monitoring Agent for NetApp Storage は、NetApp OnCommand Unified Manager、OnCommand API Service、および OnCommand Performance Manager を使用して NetApp ストレージ・システムをモニターします。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the NetApp Storage エージェント を参照してください。
- 次のコンポーネントがマシンにインストールされていることを確認してください。
  - OnCommand Unified Manager
  - OnCommand Performance Manager
  - OnCommand API Service

  これらのコンポーネントのインストールについて詳しくは、NetApp の資料を参照してください。
- OnCommand API Service、OnCommand Unified Manager、および OnCommand Performance Manager のバージョンに互換性があることを確認してください。例えば、OnCommand API Service V1.0 を構成するには、OnCommand Unified Manager V6.2、V6.1、または V6.0 と OnCommand Performance Manager V1.1 を組み合わせます。互換性のある製品バージョンについては、Interoperability Matrix Tool を参照してください。
- OnCommand Unified Manager に接続するユーザーに、モニター対象の NetApp ストレージ・システムに対する GlobalRead 特権が付与されていることを確認してください。この特権を持つ既存のユーザー ID を使用するか、新規ユーザー ID を作成します。NetApp ストレージ・システムでのユーザー ID の作成については、NetApp 資料を参照してください。
- OnCommand API Service の構成に使用するユーザーが、管理者またはモニター担当者であることを確認してください。これらのユーザー・タイプには、REST API を実行する許可がデフォルトで付与されています。
- 586 ページの『NetApp Manageability SDK JAR ファイルのダウンロードおよびインストール』に記載されているステップを実行して、NetApp Manageability SDK JAR ファイル (`manageontap.jar`) を NetApp Web サイトからダウンロードし、そのファイルをモニター・エージェントの `lib` ディレクトリーにインストールします。

このタスクについて

NetApp Storage エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で始動する必要があります。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェ

ント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

- Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

NetApp Manageability SDK JAR ファイルのダウンロードおよびインストール

NetApp Storage エージェントは、NetApp OCUM サーバーと通信するために、NetApp Manageability SDK JAR ファイルを必要とします。

このタスクについて

NetApp Storage エージェントをインストールした後に、NetApp Web サイトから NetApp Manageability SDK JAR ファイル (`manageontap.jar`) をダウンロードし、そのファイルをモニター・エージェントの `lib` ディレクトリーにインストールします。

手順

1. Web サイト http://communities.netapp.com/docs/DOC-1152 から Jar ファイルが含まれた圧縮ファイルをダウンロードします。
2. この圧縮ファイルを解凍し、`manageontap.jar` ファイルを以下の場所にコピーします。
   - 32 ビットの Windows システムの場合は、ファイルを *install_dir*`/tmaitm6` にコピーします。
   - 64 ビットの Windows システムの場合は、ファイルを *install_dir*`tmaitm6_x64` にコピーします。
   - 32 ビット Linux システムの場合、ファイルを *install_dir*`/li6263/nu/lib` にコピーします。
   - 64 ビットの x86-64 Linux システムの場合、ファイルを *install_dir*`/lx8266/nu/lib` にコピーします。
   - 64 ビット zLinux システムの場合、ファイルを *install_dir*`/ls3266/nu/lib` にコピーします。

次のタスク

エージェント構成を完了します。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

NetApp Storage エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for NetApp Storage**」を右クリックしてから「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェントを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェントを再度構成するには、「**再構成**」をクリックします。
3. 「Monitoring Agent for NetApp Storage」ウィンドウで、以下のステップを実行します。

   a) NetApp Storage エージェント・インスタンスの固有の名前を入力し、「**OK**」をクリックします.

b)「**データ・プロバイダー**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

c)「**OnCommand Unified Manager**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

d)「**OnCommand API Service**」タブで、構成パラメーターの値を指定してから、「**OK**」をクリックします。

「Monitoring Agent for NetApp Storage」ウィンドウの各タブにある構成パラメーターについて詳しくは、以下のトピックを参照してください。

- 589 ページの『データ・プロバイダーの構成パラメーター』
- 590 ページの『OnCommand Unified Manager の構成パラメーター』
- 591 ページの『OnCommand API Service の構成パラメーター』

4.「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for NetApp Storage**」を右クリックし、「**始動**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで NetApp Storage エージェント を構成するには、以下の手順を実行します。
  a) テキスト・エディターで、netapp_storage_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
     - Linux *install_dir*/samples/netapp_storage_silent_config.txt

       例: /opt/ibm/apm/agent/samples/netapp_storage_silent_config.txt
     - Windows *install_dir*¥samples¥netapp_storage_silent_config.txt

       例: C:¥IBM¥APM¥samples¥netapp_storage_silent_config.txt

  b) netapp_storage_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

     構成パラメーターについては、以下のトピックを参照してください。
     - 589 ページの『データ・プロバイダーの構成パラメーター』
     - 590 ページの『OnCommand Unified Manager の構成パラメーター』
     - 591 ページの『OnCommand API Service の構成パラメーター』

c) netapp_storage_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。

- Linux *install_dir*/bin/netapp_storage-agent.sh config *instance_name* *install_dir*/samples/netapp_storage_silent_config.txt

  例: **/opt/ibm/apm/agent/bin/netapp_storage-agent.sh config instance_name /opt/ibm/apm/agent/samples/ netapp_storage_silent_config.txt**

- Windows *install_dir*¥bin¥netapp_storage-agent.bat config *instance_name* *install_dir*¥samples¥netapp_storage_silent_config.txt

  例: **C:¥IBM¥APM¥bin¥netapp_storage-agent.bat config instance_name C:¥IBM ¥APM¥samples¥netapp_storage_silent_config.txt**

  表の見方

  instance_name
  インスタンスに指定する名前。

  install_dir
  エージェントのインストール先パス。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/netapp_storage-agent.sh start *instance_name*

  例: **/opt/ibm/apm/agent/bin/netapp_storage-agent.sh start instance_name**

- Windows *install_dir*¥bin¥netapp_storage-agent.bat start *instance_name*

  例: **C:¥IBM¥APM¥bin¥netapp_storage-agent.bat start instance_name**

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

プロンプトへの応答によるエージェントの構成

Linux システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド行に以下のコマンドを入力します。

   install_dir/bin/netapp_storage-agent.sh config *instance_name*

   例: **/opt/ibm/apm/agent/bin/netapp_storage-agent.sh config instance_name**

   表の見方

   instance_name
   インスタンスに指定する名前。

   install_dir
   エージェントのインストール先パス。

2. 以下のトピックを参照して、プロンプトに応答します。

   - 589 ページの『データ・プロバイダーの構成パラメーター』

- 590 ページの『OnCommand Unified Manager の構成パラメーター』
- 591 ページの『OnCommand API Service の構成パラメーター』

3. 以下のコマンドを実行して、エージェントを開始します。

   install_dir/bin/netapp_storage-agent.sh start *instance_name*

   例: **/opt/ibm/apm/agent/bin/netapp_storage-agent.sh start instance_name**

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。

データ・プロバイダーの構成パラメーター

NetApp Storage エージェントを構成する際に、データ・プロバイダーのログ・ファイルの最大数、ログ・ファイルの最大サイズ、ログ・ファイルに含まれる詳細レベルなど、データ・プロバイダーのパラメーターのデフォルト値を変更できます。

以下の表に、データ・プロバイダーの構成パラメーターの詳細な説明を示します。

表 *184.* データ・プロバイダーの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
インスタンス名 (**KNU_INSTANCE_NAME**)	インスタンスの名前。 **制約事項:** 「インスタンス名」フィールドでは、初めてエージェントを構成したときに指定したインスタンスの名前が表示されます。エージェントを再構成する場合に、エージェントのインスタンス名を変更することはできません。	はい
データ・プロバイダー・ログ・ファイルの最大数 (**KNU_LOG_FILE_MAX_COUNT**)	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。	はい
各データ・プロバイダー・ログの最大サイズ (KB) (**KNU_LOG_FILE_MAX_SIZE**)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。	はい

表 *184.* データ・プロバイダーの構成パラメーターの名前と説明 *(続き)*		
パラメーター名	**説明**	**必須フィールド**
データ・プロバイダー・ログの詳細レベル (**KNU_LOG_LEVEL**)	データ・プロバイダーによって作成されるログ・ファイルに含めることができる詳細のレベル。デフォルト値は 4 (Info) です。有効な値は以下のとおりです。 • 1 (オフ): メッセージは記録されません。 • 2 (重大): エラーのみが記録されます。 • 3 (Warning (警告)): 重大レベルで記録されるすべてのエラーおよびメッセージと、望ましくない動作の原因となる可能性がある潜在的なエラー。 • 4 (Info (情報)): 警告レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した概要的な情報メッセージ。 • 5 (Fine (詳細レベル (低))): 情報レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した詳しい情報メッセージ。 • 6 (Finer (詳細レベル (中))): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、パフォーマンスのプロファイル情報およびデバッグ・データなどの非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 7 (Finest (詳細レベル (高))): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、詳細なプログラミング・メッセージおよびデータを含む、非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 8 (すべて): すべてのエラーおよびメッセージが記録されます。	はい

OnCommand Unified Manager の構成パラメーター

NetApp Storage エージェントの構成時に、OnCommand Unified Manager (OCUM) のパラメーター (OCUM サーバーの IP アドレス、ユーザー名、パスワードなど) のデフォルト値を変更できます。

以下の表に、データ・ソースーの構成パラメーターの詳細な説明を示します。

表 *185. OnCommand Unified Manager* の構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
サーバー (**KNU_DATASOURCE_HOST_ADDRESS**)	モニター対象の NetApp OCUM サーバーのホスト名または IP アドレス。	はい
ユーザー (**KNU_DATASOURCE_USERNAME**)	データを収集するための十分な特権を備えている、NetApp OCUM サーバー上のユーザー名。デフォルト値は admin です。	はい

表 185. *OnCommand Unified Manager* の構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
パスワード (**KNU_DATASOURCE_PASSWORD**)	**「ユーザー」**パラメーターで指定したユーザーのパスワード。	はい
パスワードの確認入力	**「パスワードの入力 (Enter Password)」**パラメーターに指定したのと同じパスワード。	はい
プロトコル (**KNU_DATASOURCE_PROTOCOL**)	NetApp OCUM サーバーとの通信に使用するプロトコル。デフォルト値は HTTPS です。	はい

OnCommand API Service の構成パラメーター

NetApp Storage エージェントの構成時に、ホスト・アドレス、ユーザー名、パスワードなど、OnCommand API Service のパラメーターのデフォルト値を変更できます。

以下の表に、データ・ソースーの 構成パラメーターの詳細な説明を示します。

表 186. *OnCommand API Service* の構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
ホスト・アドレス (**KNU_API_SERVICES_HOST_ADDRESS**)	OnCommand API Service のホスト名または IP アドレス。	はい
ユーザー (**KNU_API_SERVICES_USERNAME**)	OnCommand API Service に接続するための十分な特権を持っているユーザー名。デフォルト値は admin です。	はい
パスワード (**KNU_API_SERVICES_PASSWORD**)	**「ユーザー」**パラメーターで指定したユーザーのパスワード。	
パスワードの確認入力	**「パスワードの入力 (Enter Password)」**パラメーターに指定したのと同じパスワード。	はい

Node.js モニターの構成

Node.js エージェントまたはスタンドアロンの Node.js データ・コレクターを使用して、Node.js アプリケーションをモニターできます。よりシンプルなインストール・プロセスおよびトランザクション・トラッキング機能が必要な場合は、Node.js データ・コレクターを使用してください。

始める前に

- 以下の手順は、このエージェントおよびデータ・コレクターの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。
- ご使用の環境で、Node.js エージェントまたは Node.js データ・コレクターのシステム要件が満たされているか確認します。

– Node.js エージェントのシステム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Node.js エージェント を参照してください。
– Node.js データ・コレクターのシステム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Node.js データ・コレクター を参照してください。

このタスクについて

以下の手順は、Monitoring Agent for Node.js およびスタンドアロンの Node.js データ・コレクターを構成するためのロードマップであり、必須のステップとオプションのステップの 両方が含まれています。

- オンプレミス・アプリケーションをモニターする 場合は、スタンドアロンの Node.js データ・コレクターまたは Node.js エージェントを構成できます。Node.js アプリケーションのトランザクション・トラッキングを有効にする場合は、スタンドアロン・データ・コレクターを構成します。
- IBM Cloud (以前の Bluemix) アプリケーションまたは Kubernetes アプリケーションをモニターするには、スタンドアロンの Node.js データ・コレクターを構成します。

ニーズに応じて、以下のステップを実行してください。

手順

- オンプレミス・アプリケーションをモニターするように Node.js エージェントを構成します。
  a) エージェントが正常に機能するようにするために、1 つのエージェント・データ・コレクターを Node.js アプリケーションに追加します。593 ページの『Node.js エージェントの構成』を参照してください。
  b) オプション: エージェントのモニター動作を変更する場合は、『Node.js エージェントのデータ・コレクターの構成』を参照してください。
  c) オプション: 診断データの収集および表示を構成する場合は、『診断データ・コレクターの構成』を参照してください。
- IBM Cloud アプリケーションをモニターするようにスタンドアロンの Node.js データ・コレクターを構成します。
  a) スタンドアロンの Node.js データ・コレクターを構成するには、598 ページの『IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』を参照してください。
  b) スタンドアロンの Node.js データ・コレクターの動作を変更するには、600 ページの『IBM Cloud アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ』を参照してください。
- オンプレミス・アプリケーションをモニターするようにスタンドアロンの Node.js データ・コレクターを構成します。
  a) スタンドアロンの Node.js データ・コレクターを構成するには、603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』を参照してください。
  b) スタンドアロンの Node.js データ・コレクターの動作を変更するには、605 ページの『オンプレミス・アプリケーション用の Node.js データ・コレクターのカスタマイズ』を参照してください。
- Kubernetes 上のアプリケーションをモニターするようにスタンドアロンの Node.js データ・コレクターを構成します。
  a) スタンドアロンの Node.js データ・コレクターを構成するには、608 ページの『Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』を参照してください。
  b) スタンドアロンの Node.js データ・コレクターの動作を変更するには、609 ページの『Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ』を参照してください。

Node.js エージェントの構成

エージェントがアプリケーションのモニターを開始できるようにするには、その前に Node.js アプリケーションにエージェント・データ・コレクターを追加して、アプリケーションを再始動する必要があります。

始める前に

同じバージョン内でエージェント設定を再構成する前に、以下のステップを実行して、以前の構成で作成されたデータ・コレクター・ファイルをクリーンアップします。

1. *install_dir*/lx8266/nj/bin ディレクトリーに移動します。
2. ./uninstall.sh コマンドを実行して、既存のデータ・コレクター・ファイルを削除します。

このタスクについて

Node.js エージェントは単一インスタンス・エージェントです。このエージェントはモニター対象の各 Node.js アプリケーションのサブノードを 登録します。サブノードの構造は、以下のとおりです。

NJ:hostname_port:NJA

ヒント: ある Node.js アプリケーションが複数のポート番号を listen する場合、最も低いポート番号が使用されます。

エージェントがアプリケーションのモニターを開始できるようにするには、その前に Node.js アプリケーションにエージェント・データ・コレクターを追加して、アプリケーションを再始動する必要があります。エージェント・データ・コレクターは、Node.js エージェントに転送されるデータを収集します。現在、以下のエージェント・データ・コレクターが用意されています。

- リソース・データ・コレクター。Node.js アプリケーションからリソース・モニター・データを収集します。
- 診断データ・コレクター。Node.js アプリケーションから診断データおよびリソース・モニター・データを収集します。
- メソッド・トレース・データ・コレクター。Node.js アプリケーションからメソッド・トレース、診断データ、およびリソース・モニター・データを収集します。

手順

1. アプリケーション・サーバーの実行に使用するユーザー ID にエージェントの install_dir ディレクトリーに対するすべての権限が付与されていることを確認してください。
2. *install_dir*/bin ディレクトリーに移動し、次のコマンドを実行します。

```
./nodejs-agent.sh config
```

3. プロンプトに従って、以下の構成オプションの値を指定します。

KNJ_NODEJS_RUNTIME_BIN_LOCATION
Node.js ランタイムの bin フォルダーのディレクトリー。デフォルトのディレクトリーは、/usr/local/bin です。

KNJ_NPM_RUNTIME_BIN_LOCATION
npm コマンドの bin フォルダーのディレクトリー。デフォルトのディレクトリーは、/usr/local/bin です。

KNJ_NPM_LIB_LOCATION
npm パッケージのグローバル・インストール・ディレクトリーの lib フォルダーのディレクトリー。デフォルトのディレクトリーは、/usr/local/lib です。例えば、npm install -g コマンドを実行して npm パッケージをインストールした場合、パッケージは /*nodejs_home*/lib/node_modules にインストールされます。この場合、**KNJ_NPM_LIB_LOCATION** は /*nodejs_home*/lib です。

CP_PORT
ソケット・クライアントからのデータに関してエージェントが listen するポート。値 0 は、一時ポートが使用されることを示します。デフォルト値は 63336 です。

注 : システムで既に使用されているポート番号は使用しないでください。ポートが既に使用中かどうかを確認するには、`netstat -apn | grep port_number` コマンドを実行します。

4. 以下のコマンドを実行してエージェントを開始します。

```
./nodejs-agent.sh start
```

5. Node.js エージェントが正常に開始されていることを確認します。エージェントが正常に開始した場合、*KNJ_NPM_LIB_LOCATION*/node_modules/ibmapm フォルダーが生成されています。
6. 所有するオファリングと要件に基づいて、エージェント・データ・コレクターを構成するために Node.js アプリケーションの .js ファイルに次のいずれかの項目を挿入します。

   注 : エージェント・データ・コレクター機能を有効にするために Node.js アプリケーションに追加できる項目は 1 つのみです。また、オファリングに含まれていない機能を有効にした場合、不要なオーバーヘッドが生じ、アプリケーションの実行効率が低下する可能性があります。

   - リソース・モニター機能のみを所有している場合は、リソース・データ・コレクターを追加できます。これを追加するには、Node.js アプリケーション・ファイルの先頭に以下の行を挿入します。

     ```
     require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm');
     ```

     ご使用の環境の **KNJ_NPM_LIB_LOCATION** の値が /usr/local/lib の場合、この行は、以 下のとおりです。

     ```
     require('/usr/local/lib/node_modules/ibmapm');
     ```

   - リソース・レベルのモニター機能に加えて診断機能を保持している場合は、次のいずれかのエージェント・データ・コレクターを追加できます。
     - メソッド・トレース・データ・コレクターを追加するには、Node.js アプリケーション・ファイルの先頭に以下の行を挿入します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm/methodtrace.js');
       ```

     - 診断データ・コレクターを追加するには、Node.js アプリケーション・ファイルの先頭に以下の行を挿入します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm/deepdive.js');
       ```

     - リソース・モニター・データ・コレクターを追加するには、Node.js アプリケーション・ファイルの先頭に以下の行を挿入します。

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm');
       ```

     最適なパフォーマンスを保証するには、デバッグのためにメソッド・トレース・データ・コレクターのみを追加します。

   注 : プラグインのコードは、Cloud APM March, 2017 から変更されています。以前のバージョンからエージェントをアップグレードした場合は、モニター機能が正常に動作するようにするには、アプリケーション内の既存のデータ・コレクターのコードを更新する必要があります。
7. Node.js アプリケーションを再始動して、エージェント・データ・コレクターを有効にします。

タスクの結果

Node.js エージェントが正常に構成されました。

次のタスク

- これで、Node.js エージェントのデータが Cloud APM コンソールで表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

重要 : Cloud APM コンソールにアプリケーションを追加するには、アプリケーション・エディターで **Node.js** を選択します。

- Node.js エージェント・データ・コレクターの実行時の動作を変更することができます。詳しくは、『Node.js エージェント・データ・コレクターの構成』を参照してください。
- 診断データ・コレクターを構成して、診断データの収集と表示を有効にすることができます。詳しくは、『エージェント診断データ・コレクターの構成』を参照してください。

Node.js エージェント データ・コレクターの構成

各 Node.js エージェント・データ・コレクターの構成ファイルでランタイム構成を変更することにより、その動作を変更できます。

ランタイム構成ファイル

Node.js データ・コレクター・コードは、以下のディレクトリーにあります。

KNJ_NPM_LIB_LOCATION/node_modules/ibmapm

ここで、*KNJ_NPM_LIB_LOCATION* は、npm パッケージのグローバル・インストール・ディレクトリーの lib フォルダーのディレクトリーです。デフォルトのディレクトリーは、/usr/local/lib です。

同じフォルダー内に各エージェント・データ・コレクターのランタイム構成ファイルもあります。エージェント・データ・コレクターは、1 分ごとに構成ファイルを読み取ります。

ヒント : ランタイム構成ファイルの名前のフォーマットは、以下のとおりです。

plugin_*application port number*_conf.json

構成ファイルの内容を変更すると、関連するエージェント・データ・コレクターの動作が変更されます。構成ファイルでは、以下の 2 つのタイプの情報を変更できます。

- URL フィルタリング・ルール
- エージェント・データ・コレクターのロギング・パラメーター

URL フィルタリング・ルール

ランタイム構成ファイルで URL フィルタリング・ルールを変更できます。URL パス名をユーザー・カスタマイズされたパス名にマップするには、正規表現が使用されます。 カスタマイズしたパス名に URL をマップして、以下の要件を満たすことができます。

- 類似したパスを持つ URL を集約する。例えば、次のような URL パスがあるとします。

```
/demo/poll/1
/demo/poll/2
/demo/poll/3
...
```

Web サーバーで、このようなパスの要求は、共通ルーチンによって処理されることが多いと思われます。そのため、以下の例のようなフィルターを使用して、単一の URL タイプにパスを集約できます。

```
"filters":
[
    {
        "pattern": "/demo/poll/.+",
        "to": "/demo/poll/"
    }
```

このフィルターを使用すると、"/demo/poll/xxx" などの URL パスへの要求がすべて、"/demo/poll" という URL のパス・タイプにマップされます。このようにすると、このタイプの URL パスへのすべての要求の応答時間は、単一値に平均化されます。この方法で集約すれば、使用可能なリソースをさらに効率よく使用するのに役立ちます。

- 静的ファイルへの URL パスを無視したり、特定のタイプの要求をフィルターで除外したりする。例えば、個別のサーバー・ダウンロード要求を生成するイメージが Web ページに組み込まれている場合、このタイプの要求の応答時間を表示する必要がない場合があります。

特定のタイプの要求をフィルターで除外するには、以下の例の形式に従って、to の値を空に設定します。

```
"filters":
[
    {
        "pattern": "GET /css/.+¥¥.css$",
        "to": ""
    }
```

このフィルターにより、.css ファイルを取得する要求は無視されます。結果として、モニターする必要がある要求で、使用可能なリソースをより効率的に使用できるようになります。

構成ファイルで、URL フィルター・ルールは、以下のように filters という名前の JSON 配列で示されています。

```
"filters":
[
    {
        "pattern": ".+¥¥.png$",
        "to": ""
    },
    {
        "pattern": ".+¥¥.jpg$",
        "to": ""
    },
    {
        "pattern": "GET /js/.+¥¥.js$",
        "to": ""
    },
    {
        "pattern": "GET /css/.+¥¥.css$",
        "to": ""
    }
]
```

配列内の各メンバーは 1 つのフィルター・ルールです。エージェント・データ・コレクターは、HTTP 要求を受け取ると、その要求から URL パス名を抽出し、それを各 "pattern" と比較します。パス名が "pattern" と一致しない場合、元の URL パス名が保持され、測定に使用されます。

エージェント・データ・コレクターのロギング・パラメーター

KNJ_NPM_LIB_LOCATION/node_modules/ibmapm/etc ディレクトリーにある構成ファイル config.properties のパラメーターを変更することで、ロギングの動作を変更できます。変更可能なロギング・パラメーターは以下のとおりです。

ログ・レベル

構成ファイル内のログ・レベル用の項目は、KNJ_LOG_LEVEL=info です。つまり、アクションに関する要約情報がログに出力されます。ログ・レベルを設定するには、KNJ_LOG_LEVEL の値を変更します。デフォルト値は info であり、ログは標準出力に出力されます。

以下の 5 つのログ・レベル値がサポートされます。

off
ログは出力されません。

error
エラー状態に関する情報のみがログに記録されます。

info
Node.js エージェントのデータ・コレクターが正常に実行されている場合に、情報がログに記録されます。エージェントに送信された未加工のモニター・データもログに記録されます。

debug
デバッグ、通知、およびエラーの情報がログに出力されます (収集されたデータ、サーバーに送信されたデータ、サーバー応答など)。

all
すべての情報がログに出力されます。

Node.js エージェントの診断データ・コレクターの構成

診断データ収集機能のサポートは、デフォルトで無効に設定されています。診断機能を保持している場合は、特定の Node.js アプリケーションのデータ収集を設定および調整する必要があります。

手順

- 実行中の特定アプリケーションのデータ・コレクター設定を変更するには、以下のようにします。
  1. *KNJ_NPM_LIB_LOCATION*/node_modules/ibmapm ディレクトリーに移動し、テキスト・エディターで plugin_*port*_conf.json ファイルを開きます。

     ヒント: *KNJ_NPM_LIB_LOCATION* については、593 ページの『KNJ_NPM_LIB_LOCATION』のパラメーターの説明を参照してください。
  2. データ・コレクター設定の変更については、以下の表を参照してください。

表 *187.* データ・コレクター設定

診断データ・カテゴリー	説明	プロパティー	アクション
スタック・トレース・レポート作成の時刻の最小差分	要求またはメソッド呼び出しのスタック・トレースを収集するための応答時間しきい値を指定します。要求またはメソッド呼び出しの応答時間がこの値を超過した場合、データ・コレクターはそのスタック・トレースを収集します。	minClockStack	ミリ秒単位の値に設定します
要求をレポートする時刻の最小差分	要求インスタンスのメソッド・トレースを収集するための応答時間しきい値を指定します。要求インスタンスの応答時間がこのしきい値を超えた場合、データ・コレクターはそのメソッド・トレースを収集します。	minClockTrace	ミリ秒単位の値に設定します
ファイルごとのイベントの最大数	.jso ファイルに記録するイベントの最大数を指定します。 .jso ファイルは、該当するイベントの診断データを記録します。	eventsPerFile	イベントの最大数の値に設定します
1 つのファイルにレポートする最大時間	.jso ファイルで診断データを記録する最大時間を指定します。	fileCommitTime	秒単位の最大時間に設定します

表 187. データ・コレクター設定 (続き)			
診断データ・カテゴリー	**説明**	**プロパティー**	**アクション**
この数を超えると最も古いファイルが削除される、保持するファイルの最大数	保持する .jso ファイルの最大数を指定します。この数を超えると、最も古いファイルが削除されます。	maxFiles	ファイルの最大数に設定します
要求サンプリング期間	要求のサンプリング期間を指定します。	sampling	必要なサンプリング期間に設定します。デフォルト値は 10 です。値 10 は、エージェントが 10 件の要求ごとに 1 件収集することを意味します。

- オプション: Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報を診断データ・コレクターで収集する場合、*SECURITY_OFF* 環境変数を設定します。これらの情報はデフォルトでは収集されません。

  この変数を設定すると、情報の漏えいにつながる可能性があるため、使用する際は注意してください。

  Linux この環境変数を設定するには、以下のコマンドを実行します。

  ```
  export SECURITY_OFF=true
  ```

タスクの結果

指定された実行中のアプリケーションまたはすべてのアプリケーションに 対する診断データ・コレクターの構成が変更されました。

IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成

IBM Cloud 上の Node.js アプリケーションに関する情報を収集するには、スタンドアロン Node.js データ・コレクターを構成する必要があります。

始める前に

1. ご使用の Node.js アプリケーションがローカルで正常に実行できることを確認してください。スタンドアロン Node.js データ・コレクターは、Node.js V8.0.0 と今後のフィックスパック、V10.0.0 と今後のフィックスパック、および V12.0.0 と今後のフィックスパックをモニターできます。
2. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

手順

1. データ・コレクター・パッケージからファイルを抽出します。
   nodejs_datacollector_8.1.4.0.6.tgz パッケージは、解凍されたディレクトリーに含まれています。
2. nodejs_datacollector_8.1.4.0.6.tgz ファイルを解凍します。例えば、以下のコマンドを実行します。

   ```
   tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
   ```

3. 以下のコマンドを実行して、nodejs_dc フォルダー内の ibmapm.tgz ファイルを解凍します。

   ```
   tar -zxf nodejs_dc/ibmapm.tgz
   ```

ibmapm フォルダーが作成されます。

4. データ・コレクター・パッケージから解凍された ibmapm フォルダーを、アプリケーションのホーム・ディレクトリーにコピーします。例えば、以下のコマンドを実行します。

```
cp –r directory_to_the_ibmapm_folder  home_directory_of_your_Node.js_application
```

ヒント : Node.js アプリケーションのホーム・ディレクトリーは、その Node.js アプリケーションを開始するのに使用するコマンドと、メインファイルを含むディレクトリーによって決定されます。Node.js アプリケーションの開始に **node app.js** コマンドを使用し、app.js メインファイルが /root/nodejs_app ディレクトリーに存在する場合、/root/nodejs_app がアプリケーションのホーム・ディレクトリーです。

5. Node.js アプリケーションの package.json ファイルで、以下の行を dependencies セクションに追加します。

```
 “ibmapm": "./ibmapm"
```

要確認 : ファイルの各行の末尾には必ずコンマを入力し (最終行を除く)、package.json ファイルを適切な形式に保つようにしてください。

例:

```
 "dependencies": {
    "ibmapm": "./ibmapm",
    "cors": "^2.5.2",
    "helmet": "^1.3.0",
    "loopback": "^2.22.0",
    "loopback-boot": "^2.6.5",
    "loopback-datasource-juggler": "^2.39.0",
    "serve-favicon": "^2.0.1",
    "strong-error-handler": "^1.0.1"
  }
```

6. 以下の行を、Node.js アプリケーションのメインファイルの先頭に追加します。

```
require('ibmapm');
```

node app.js コマンドを実行してアプリケーションを開始する場合は、app.js がアプリケーションのメインファイルです。

7. Node.js アプリケーションの manifest.yml ファイルを含むディレクトリーから IBM Cloud にログインし、以下のコマンドを実行します。

```
cf push
```

ヒント : サンプルの manifest.yml ファイルについては、192 ページの『サンプル manifest.yml ファイル』を参照してください。

タスクの結果

データ・コレクターが構成され、Cloud APM サーバーに接続されました。

次のタスク

ご使用の IBM Cloud アプリケーションのモニター・データが Cloud APM コンソールに表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

要確認 : Cloud APM コンソールにアプリケーションを追加するには、アプリケーション・エディターで「**Node.js ランタイム (Node.js Runtime)**」を選択します。

IBM Cloud アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ

IBM Cloud ユーザー・インターフェース (UI) で環境変数を追加して、ご使用の IBM Cloud アプリケーションのモニターをカスタマイズできます。

Node.js データ・コレクター用のユーザー定義の環境変数

以下の表の情報を使用して、IBM Cloud での Node.js モニターをカスタマイズできます。

表 *188. IBM Cloud* での *Node.js* モニター用にサポートされるユーザー定義の環境変数

変数名	重要度	値	説明
KNJ_SAMPLING	オプション	要求のサンプリング数	1 件のサンプルを取得するための基になる要求数。 デフォルト値は 10 です。これは、10 件の要求ごとに 1 件モニターされることを意味します。 この変数を設定しない場合、デフォルト値 10 が使用されます。
KNJ_MIN_CLOCK_TRACE	オプション	メソッド・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのメソッド・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。
KNJ_MIN_CLOCK_STACK	オプション	スタック・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのスタック・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。
KNJ_ENABLE_METHODTRACE	オプション	• True • False	メソッド・トレースを有効または無効にします。 • この変数を true に設定すると、要求のメソッド・トレースが無効になります。 • この値を false に設定すると、要求のメソッド・トレースが有効になります。これはデフォルト値です。 この変数を設定しない場合、デフォルト値 False が使用され、要求のメソッド・トレースが有効になります。
KNJ_ENABLE_DEEPDIVE	オプション	• True • False	この変数を true に設定すると、診断データはサーバーに送信されます。デフォルトでは、この値は false に設定されています。これは、診断データがサーバーに送信されないことを意味します。

表 188. *IBM Cloud* での *Node.js* モニター用にサポートされるユーザー定義の環境変数 (続き)

変数名	重要度	値	説明
KNJ_ENABLE_TT	オプション	• `true` • `false`	AAR のトランザクション・トラッキングを有効または無効にします。 • この変数を true に設定すると、AAR のトランザクション・トラッキングが有効になります。 • この変数を false に設定すると、AAR のトランザクション・トラッキングが無効になります。 デフォルトでは、この値は設定されません。これは、トランザクション・トラッキングが無効になることを意味します。
KNJ_AAR_BATCH_FREQ	オプション	AAR データが送信される間隔 (秒)	AAR データがバッチ処理されてサーバーに送信される間隔 (秒) を指定します。 デフォルト値は 60 です。これは、AAR データが 1 分ごとにバッチ処理されてサーバーに送信されることを意味します。 **注 :** この変数は、KNJ_AAR_BATCH_COUNT と連動して、AAR データがバッチ処理されてサーバーに送信されるタイミングを決定します。2 つの変数のいずれかによって設定された条件が満たされると、AAR データがバッチ処理されて送信されます。設定よりも短い間隔で AAR データに含まれている要求が最大数 (例えば、100) に達した場合にも、データは即時にバッチ処理されて送信されます。
KNJ_AAR_BATCH_COUNT	オプション	AAR データのバッチに含まれる要求の最大数	サーバーに送信する前に AAR データのバッチに含めることができる要求の最大数を指定します。 デフォルト値は 100 です。これは、AAR データのバッチに含まれる要求の数が 100 に達すると、AAR データのこのバッチがサーバーに送信されることを意味します。

表 188. *IBM Cloud* での *Node.js* モニター用にサポートされるユーザー定義の環境変数 (続き)			
変数名	**重要度**	**値**	**説明**
KNJ_LOG_LEVEL	オプション	ログに出力する情報のレベル。	ログに出力する情報のレベルを制御します。以下のレベルを使用できます。 **off** ログは出力されません。 **error** エラー状態に関する情報のみがログに記録されます。 **info** Node.js エージェントのデータ・コレクターが正常に実行されている場合に、情報がログに記録されます。エージェントに送信された未加工モニター・データもログに記録されます。 **debug** デバッグ、通知、およびエラーの情報がログに出力されます (収集されたデータ、サーバーに送信されたデータ、サーバー応答など)。 **all** すべての情報がログに出力されます。 デフォルトでは、ログ・レベルは `info` です。この場合、データ・コレクターのアクションに関する要約情報がログに出力されます。ログは標準出力に出力されます。
SECURITY_OFF	オプション	• `true` • `false`	Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報の収集を有効または無効にします。 • この変数を true に設定すると、ユーザー機密情報が収集されます。 • この変数を false に設定すると、ユーザー機密情報は収集されません。これはデフォルト値です。 この変数を指定しない場合、デフォルト値の `false` が使用され、ユーザー機密情報は収集されません。

IBM Cloud アプリケーション用のスタンドアロン Node.js データ・コレクターの構成解除

Node.js 環境をモニターする必要がない場合、またはスタンドアロン Node.js データ・コレクターをアップグレードしたい場合は、まずスタンドアロン Node.js データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. アプリケーションのメインファイルから `require('ibmapm');` という行を削除します。

ヒント : node app.js コマンドを実行してアプリケーションを開始する場合は、app.js がアプリケーションのメインファイルです。

2. package.json ファイルで以下の依存関係を削除します。

```
“ibmapm": "./ibmapm"
```

要確認 : アプリケーションで必要な依存関係は削除しないでください。

3. アプリケーションのホーム・ディレクトリーで ibmapm フォルダーを削除します。

タスクの結果

スタンドアロン Node.js データ・コレクターが正常に構成解除されました。

次のタスク

データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成

Node.js アプリケーションをオンプレミス環境にインストールした場合は、Node.js アプリケーションに関する情報を収集するように Node.js データ・コレクターを構成する必要があります。

始める前に

1. ご使用の Node.js アプリケーションがローカルで正常に実行できることを確認してください。スタンドアロン Node.js データ・コレクター は、Node.js V8.0.0 と今後のフィックスパック、V10.0.0 と今後のフィックスパック、および V12.0.0 と今後のフィックスパックをモニターできます。
2. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

手順

1. データ・コレクター・パッケージからファイルを抽出します。
   nodejs_datacollector_8.1.4.0.6.tgz パッケージは、解凍されたディレクトリーに含まれています。
2. アプリケーションのホーム・ディレクトリーを判別します。
   - 標準的な Node.js アプリケーションでは、**node app.js** コマンドを使用して Node.js アプリケーションを開始し、app.js メインファイルが /root/nodejs_app ディレクトリーにある場合、/root/nodejs_app がアプリケーションのホーム・ディレクトリーです。
   - IBM API Connect 環境内の集合メンバーの場合、**wlpn-server list** コマンドを実行して、同じマシン上のすべての集合メンバーのリストを表示します。集合メンバーのホーム・ディレクトリーは、以下の形式です。

     ```
     user_directory/collective-member_name/package
     ```

     例えば、コマンド出力として /root/wlpn/rock-8345a96-148538-1/package が得られた場合、/root/wlpn はユーザー・ディレクトリーであり、rock-8345a96-148538-1 は集合メンバー名です。
   - IBM API Connect 環境内の Developer Portal アプリケーションの場合、**ps -ef | grep node** コマンドを実行して、ホーム・ディレクトリーを確認できます。例えば、以下のコマンド出力が得られた

場合、ホーム・ディレクトリーは /home/admin/bgsync であり、アプリケーションのメインファイルは rest_server.js です。

```
admin 19085 1 0 Jun25 ?  00:06:53 /usr/local/bin/node /home/admin/bgsync/
rest_server.js
```

3. アプリケーションのホーム・ディレクトリーから、以下のコマンドを実行して、データ・コレクター・パッケージからファイルを解凍します。

```
tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
```

4. 以下のコマンドを実行して、nodejs_dc フォルダー内の ibmapm.tgz ファイルを解凍します。

```
tar -zxf nodejs_dc/ibmapm.tgz
```

ibmapm フォルダーが作成されます。

5. 以下のコマンドを実行して、アプリケーションのデータ・コレクターをインストールします。

```
npm install ./ibmapm
```

6. 以下の行を、Node.js アプリケーションのメインファイルの 先頭に追加します。

```
require('ibmapm');
```

- **node app.js** コマンドを実行してアプリケーションを開始する場合は、app.js がアプリケーションのメインファイルです。
- IBM API Connect 環境内の集合メンバーの場合、メインファイルは、ホーム・ディレクトリーまたはそのサブフォルダー内の package.json ファイルで定義されます。デフォルトでは、メインファイルは、*home_directory*/server/server.js です。ここで、*home_directory* は、集合メンバーのホーム・ディレクトリーです。
- IBM API Connect 環境内の Developer Portal アプリケーションの場合、**ps -ef | grep node** コマンドを実行して、メインファイルを確認できます。例えば、以下のコマンド出力が得られた場合、アプリケーションのメインファイルは rest_server.js です。

```
admin 19085 1 0 Jun25 ?  00:06:53 /usr/local/bin/node /home/admin/bgsync/
rest_server.js
```

7. アプリケーションを再始動します。

ヒント:

- 集合メンバーを再始動するには、wlpn-server stop *collective_member_name* コマンドを実行します。このコマンドを実行すると、集合メンバーは自動的に再始動します。始動しない場合は、wlpn-server start *collective_member_name* コマンドを実行して手動で再始動します。
- Developer Portal アプリケーションを再始動するには、まず /etc/init.d/restservice stop コマンドを実行してアプリケーションを停止してから、/etc/init.d/restservice start コマンドを実行してアプリケーションを開始します。

タスクの結果

データ・コレクターが構成され、Cloud APM サーバーに接続されました。

次のタスク

- ご使用のアプリケーションのモニター・データが Cloud APM コンソールに表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

- API Connect 環境のトポロジー情報を表示するには、トランザクション・トラッキングを有効にします。詳しい説明については、605 ページの『オンプレミス・アプリケーション用の Node.js データ・コレクターのカスタマイズ』の *KNJ_ENABLE_TT* 変数の説明を参照してください。

要確認 : Cloud APM コンソールにアプリケーションを追加するには、アプリケーション・エディターで「**Node.js ランタイム (Node.js Runtime)**」を選択します。

オンプレミス・アプリケーション用の Node.js データ・コレクターのカスタマイズ

データ・コレクター・パッケージ内のファイルを変更して、Node.js アプリケーションのモニターをカスタマイズするための環境変数を設定できます。

環境変数をカスタマイズするか `config.properties` ファイルを編集することによって変数を設定できます。`config.properties` ファイルは、Node.js データ・コレクターがインストールされている `ibmapm/etc` フォルダーにあります。

表 *189.* サポートされる変数

変数名	重要度	値	説明
KNJ_SAMPLING	オプション	要求のサンプリング数	1 件のサンプルを取得するための基になる要求数。 デフォルト値は 10 です。これは、10 件の要求ごとに 1 件モニターされることを意味します。 この変数を設定しない場合、デフォルト値 10 が使用されます。
KNJ_MIN_CLOCK_TRACE	オプション	メソッド・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのメソッド・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。
KNJ_MIN_CLOCK_STACK	オプション	スタック・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのスタック・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。
KNJ_ENABLE_METHODTRACE	オプション	• True • False	メソッド・トレースを有効または無効にします。 • この変数を true に設定すると、要求のメソッド・トレースが無効になります。 • この値を false に設定すると、要求のメソッド・トレースが有効になります。これはデフォルト値です。 この変数を設定しない場合、デフォルト値 False が使用され、要求のメソッド・トレースが有効になります。

表 189. サポートされる変数 (続き)

変数名	重要度	値	説明
KNJ_ENABLE_DEEPDIVE	オプション	• True • False	この変数を true に設定すると、診断データはサーバーに送信されます。デフォルトでは、この値は false に設定されています。これは、診断データがサーバーに送信されないことを意味します。
KNJ_ENABLE_TT	オプション	• true • false	AAR のトランザクション・トラッキングを有効または無効にします。 • この変数を true に設定すると、AAR のトランザクション・トラッキングは有効になります。 • この変数を false に設定すると、AAR のトランザクション・トラッキングは無効になります。 デフォルトでは、この値は設定されません。これは、トランザクション・トラッキングが無効になることを意味します。
KNJ_AAR_BATCH_FREQ	オプション	AAR データが送信される間隔 (秒)	AAR データがバッチ処理されてサーバーに送信される間隔 (秒) を指定します。 デフォルト値は 60 です。これは、AAR データが 1 分ごとにバッチ処理されてサーバーに送信されることを意味します。 **注 :** この変数は、KNJ_AAR_BATCH_COUNT と連動して、AAR データがバッチ処理されてサーバーに送信されるタイミングを決定します。2 つの変数のいずれかによって設定された条件が満たされると、AAR データがバッチ処理されて送信されます。設定よりも短い間隔で AAR データに含まれている要求が最大数 (例えば、100) に達した場合にも、データは即時にバッチ処理されて送信されます。
KNJ_AAR_BATCH_COUNT	オプション	AAR データのバッチに含まれる要求の最大数	サーバーに送信する前に AAR データのバッチに含めることができる要求の最大数を指定します。 デフォルト値は 100 です。これは、AAR データのバッチに含まれる要求の数が 100 に達すると、AAR データのこのバッチがサーバーに送信されることを意味します。

表 189. サポートされる変数 (続き)

変数名	重要度	値	説明
KNJ_LOG_LEVEL	オプション	ログに出力する情報のレベル。	ログに出力する情報のレベルを制御します。以下のレベルを使用できます。 **off** ログは出力されません。 **error** エラー状態に関する情報のみがログに記録されます。 **info** Node.js エージェントのデータ・コレクターが正常に実行されている場合に、情報がログに記録されます。エージェントに送信された未加工モニター・データもログに記録されます。 **debug** デバッグ、通知、およびエラーの情報がログに出力されます (収集されたデータ、サーバーに送信されたデータ、サーバー応答など)。 **all** すべての情報がログに出力されます。 デフォルトでは、ログ・レベルは `info` です。この場合、データ・コレクターのアクションに関する要約情報がログに出力されます。ログは標準出力に出力されます。
SECURITY_OFF	オプション	• `true` • `false`	Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報の収集を有効または無効にします。 • この変数を true に設定すると、ユーザー機密情報が収集されます。 • この変数を false に設定すると、ユーザー機密情報は収集されません。これはデフォルト値です。 この変数を指定しない場合、デフォルト値の `false` が使用され、ユーザー機密情報は収集されません。

オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成解除

Node.js 環境をモニターする必要がない場合、またはスタンドアロン Node.js データ・コレクターをアップグレードしたい場合は、まずスタンドアロン Node.js データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. アプリケーションのメインファイルから `require('ibmapm');` という行を削除します。

ヒント : node app.js コマンドを実行してアプリケーションを開始する場合は、app.js がアプリケーションのメインファイルです。

2. Node.js アプリケーションの package.json ファイル内の dependencies セクションから “ibmapm": "./ibmapm" を削除します。
3. アプリケーションのホーム・ディレクトリーで node_modules フォルダーを削除します。
4. npm install コマンドを実行して、アプリケーション依存関係をインストールします。

タスクの結果
スタンドアロン Node.js データ・コレクターが正常に構成解除されました。

次のタスク
データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターの構成

Node.js アプリケーションを Kubernetes 上にインストールした場合は、Node.js アプリケーションに関する情報を収集するように Node.js データ・コレクターを構成できます。

始める前に

1. ご使用の Node.js アプリケーションが正常に実行できることを確認してください。スタンドアロン Node.js データ・コレクター は、Node.js V8.0.0 と今後のフィックスパック、V10.0.0 と今後のフィックスパック、および V12.0.0 と今後のフィックスパックをモニターできます。
2. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

手順

1. データ・コレクター・パッケージからファイルを抽出します。
   nodejs_datacollector_8.1.4.0.6.tgz パッケージは、解凍されたディレクトリーに含まれています。
2. nodejs_datacollector_8.1.4.0.6.tgz ファイルを解凍します。例えば、以下のコマンドを実行します。

   ```
   tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
   ```

3. 以下のコマンドを実行して、nodejs_dc フォルダー内の ibmapm.tgz ファイルを解凍します。

   ```
   tar -zxf nodejs_dc/ibmapm.tgz
   ```

   ibmapm フォルダーが作成されます。
4. Node.js アプリケーションの package.json ファイルで、以下の行を dependencies セクションに追加します。

   ```
   "ibmapm": "./ibmapm"
   ```

   要確認 : ファイルの各行の末尾には必ずコンマを入力し (最終行を除く)、package.json ファイルを適切な形式に保つようにしてください。
5. 以下の行を、Node.js アプリケーションのメインファイルの 先頭に追加します。

   ```
   require('./ibmapm');
   ```

node app.js コマンドを実行してアプリケーションを開始する場合は、app.js がアプリケーションのメインファイルです。

6. Docker イメージを再ビルドします。

注 : Node.js アプリケーションを他の Docker 環境 (Docker Swarm や AWS Docker のサービスなど) で実行する場合は、各ステップを Docker に対応させる必要があります。

次のタスク

モニターをカスタマイズする場合は、デプロイメントの yaml ファイルに環境変数を追加できます。詳しくは、609 ページの『Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ』を参照してください。

Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターのカスタマイズ

デプロイメントの yaml ファイルに環境変数を追加し、Kubernetes アプリケーションのモニターをカスタマイズできます。

Node.js データ・コレクター用のユーザー定義の環境変数

以下の表の情報を使用して、Kubernetes での Node.js のモニターをカスタマイズできます。

表 *190. Kubernetes* での *Node.js* のモニター用にサポートされるユーザー定義の環境変数

変数名	重要度	値	説明
KNJ_SAMPLING	オプション	要求のサンプリング数	1 件のサンプルを取得するための基になる要求数。 デフォルト値は 10 です。これは、10 件の要求ごとに 1 件モニターされることを意味します。 この変数を設定しない場合、デフォルト値 10 が使用されます。
KNJ_MIN_CLOCK_TRACE	オプション	メソッド・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのメソッド・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。
KNJ_MIN_CLOCK_STACK	オプション	スタック・トレースを収集するための応答時間しきい値 (ミリ秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのスタック・トレースを収集します。 デフォルト値は 0 です。 この変数を設定しない場合、デフォルト値 0 が使用されます。

表 190. *Kubernetes* での *Node.js* のモニター用にサポートされるユーザー定義の環境変数 (続き)

変数名	重要度	値	説明
KNJ_ENABLE_METHODTRACE	オプション	• True • False	メソッド・トレースを有効または無効にします。 • この変数を true に設定すると、要求のメソッド・トレースが無効になります。 • この値を false に設定すると、要求のメソッド・トレースが有効になります。これはデフォルト値です。 この変数を設定しない場合、デフォルト値 False が使用され、要求のメソッド・トレースが有効になります。
KNJ_ENABLE_DEEPDIVE	オプション	• True • False	この変数を true に設定すると、診断データはサーバーに送信されます。デフォルトでは、この値は false に設定されています。これは、診断データがサーバーに送信されないことを意味します。
KNJ_ENABLE_TT	オプション	• true • false	AAR のトランザクション・トラッキングを有効または無効にします。 • この変数を true に設定すると、AAR のトランザクション・トラッキングが有効になります。 • この変数を false に設定すると、AAR のトランザクション・トラッキングが無効になります。 デフォルトでは、この値は設定されません。これは、トランザクション・トラッキングが無効になることを意味します。
KNJ_AAR_BATCH_FREQ	オプション	AAR データが送信される間隔 (秒)	AAR データがバッチ処理されてサーバーに送信される間隔 (秒) を指定します。 デフォルト値は 60 です。これは、AAR データが 1 分ごとにバッチ処理されてサーバーに送信されることを意味します。 **注 :** この変数は、KNJ_AAR_BATCH_COUNT と連動して、AAR データがバッチ処理されてサーバーに送信されるタイミングを決定します。2 つの変数のいずれかによって設定された条件が満たされると、AAR データがバッチ処理されて送信されます。設定よりも短い間隔で AAR データに含まれている要求が最大数 (例えば、100) に達した場合にも、データは即時にバッチ処理されて送信されます。

表 190. *Kubernetes* での *Node.js* のモニター用にサポートされるユーザー定義の環境変数 (続き)

変数名	重要度	値	説明
KNJ_AAR_BATCH_COUNT	オプション	AAR データのバッチに含まれる要求の最大数	サーバーに送信する前に AAR データのバッチに含めることができる要求の最大数を指定します。 デフォルト値は 100 です。これは、AAR データのバッチに含まれる要求の数が 100 に達すると、AAR データのこのバッチがサーバーに送信されることを意味します。
KNJ_LOG_LEVEL	オプション	ログに出力する情報のレベル。	ログに出力する情報のレベルを制御します。以下のレベルを使用できます。 **off** ログは出力されません。 **error** エラー状態に関する情報のみがログに記録されます。 **info** Node.js エージェントのデータ・コレクターが正常に実行されている場合に、情報がログに記録されます。エージェントに送信された未加工モニター・データもログに記録されます。 **debug** デバッグ、通知、およびエラーの情報がログに出力されます (収集されたデータ、サーバーに送信されたデータ、サーバー応答など)。 **all** すべての情報がログに出力されます。 デフォルトでは、ログ・レベルは `info` です。この場合、データ・コレクターのアクションに関する要約情報がログに出力されます。ログは標準出力に出力されます。
SECURITY_OFF	オプション	• `true` • `false`	Cookie、HTTP 要求コンテキスト、データベース要求コンテキストなどのユーザー機密情報の収集を有効または無効にします。 • この変数を true に設定すると、ユーザー機密情報が収集されます。 • この変数を false に設定すると、ユーザー機密情報は収集されません。これはデフォルト値です。 この変数を指定しない場合、デフォルト値の `false` が使用され、ユーザー機密情報は収集されません。

yaml ファイルの例

```
spec:
      containers:
      - name: testapp
        image: mycluster.icp:8500/default/testapp:v1
        imagePullPolicy: Always
        ports:
        - containerPort: 3000
          protocol: TCP
        env:
        - name: KNJ_LOG_LEVEL
          value: "debug"
        - name: KNJ_ENABLE_TT
          value: "true"
        - name: KNJ_ENABLE_DEEPDIVE
          value: "true"
```

Kubernetes アプリケーション用のスタンドアロン Node.js データ・コレクターの構成解除
Node.js 環境をモニターする必要がない場合、またはスタンドアロン Node.js データ・コレクターをアップグレードしたい場合は、まずスタンドアロン Node.js データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. アプリケーションのメインファイルから `require('ibmapm');` という行を削除します。

   ヒント : node app.js コマンドを実行してアプリケーションを開始する場合は、`app.js` がアプリケーションのメインファイルです。
2. `package.json` ファイルで以下の依存関係を削除します。

   ```
   "ibmapm": "./ibmapm"
   ```

   要確認 : アプリケーションで必要な依存関係は削除しないでください。

タスクの結果
スタンドアロン Node.js データ・コレクターが正常に構成解除されました。

次のタスク
データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

OpenStack モニターの構成

Monitoring Agent for OpenStack が OpenStack エージェント環境を自動的にモニターできるようにするには、事前にこのエージェントを構成する必要があります。

手順

1. プロンプトに応答してエージェントを構成します。説明については、613 ページの『OpenStack エージェントの構成』を参照してください。
2. プロセス関連情報を収集する場合は、OpenStack エージェント用にデータ・コレクターを構成します。説明については、615 ページの『プロセス関連情報の収集および SSH 接続の有効化』を参照してください。
3. エージェントが作動するための構成値を入力します。説明については、616 ページの『構成値の追加』を参照してください。

OpenStack エージェントの構成

通常の環境では、OpenStack エージェントで OpenStack 環境を自動的にモニターする場合、まずエージェントを構成する必要があります。

始める前に

Linux システムでのプリインストール で説明されている必須ソフトウェアをすべてインストールしていることを確認します。

手順

Linux システムで OpenStack エージェントを構成するには、以下の 2 つのオプションがあります。

- スクリプトを実行してプロンプトに応答することでエージェントを構成する場合は、613 ページの『対話式の構成』を参照してください。
- サイレント応答ファイルを編集し、対話なしでスクリプトを実行することでエージェントを構成する場合は、614 ページの『サイレント構成』を参照してください。

対話式の構成

手順

1. エージェントを構成するには、以下のコマンドを実行します。

   ```
   install_dir/bin/openstack-agent.sh config instance_name
   ```

   ここで、*install_dir* は OpenStack エージェント のインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。
2. 「インスタンス名の入力 (Enter instance name)」というプロンプトが出されたら、インスタンス名を指定します。

   重要 : OpenStack エージェントは複数インスタンス・エージェントであるため、エージェント・インスタンスごとにインスタンス名が必要です。 指定したインスタンス名は、管理対象システム名 `instance_name:host_name:sg` に含まれます。指定するインスタンス名の長さは、28 - (ホスト名の長さ) 文字に制限されます。例えば、インスタンス名に OS1 を指定すると、管理対象システム名は `OS1:hostname:S` になります。
3. 「Monitoring Agent for OpenStack の編集 (Edit Monitoring Agent for OpenStack)」というプロンプトが出されたら、Enter キーを押して続行します。
4. 「OpenStack 環境の認証情報の編集 (Edit OpenStack environment authentication information)」というプロンプトが出されたら、以下の情報を指定します。

   ```
   OpenStack authentication url (default is: http://localhost:identity/v3):
   OpenStack username (default is: admin):
   Enter OpenStack password (default is: ):
   Re-type: OpenStack password (default is: ):
   OpenStack tenant name (default is: admin):
   ```
5. 「Python 実行可能ファイルの場所」を求めるプロンプトが出されたら、Python 実行可能ファイルの場所を指定します (例えば、`/usr/bin/python`)。

   完全修飾パスを確認するには、ご使用の環境で以下のコマンドを実行します。

   ```
   which python
   ```
6. 「ポート番号」を求めるプロンプトが出されたら、デフォルト値を受け入れるか、ポート番号を指定します。

   このポートは、OpenStack エージェント・データ・コレクターと OpenStack エージェント (いずれもローカル・サーバーにのみインストールされます) の間の内部通信をモニターするために使用されます。エージェントはこのポートでデータ・コレクターからのデータを listen します。デフォルト値 0 は、エージェントの開始時に一時ポートが使用されることを示します。ポートに対して厳密なセキュリティ

ー・ルールが適用されているサーバーでは、エージェントで使用する特定の単一ポートを構成できます。このポートはエージェントによって内部的に使用され、OpenStack 環境には関係しません。

7. システム上の /etc/hosts ファイルを編集して、モニター対象ノードごとにホスト・マッピングを追加します。

サイレント構成

手順

1. テキスト・エディターで sg_silent_config.txt ファイルを開きます。このファイルは *install_dir*/samples ディレクトリーにあります。ここで、*install_dir* は OpenStack エージェントのインストール・ディレクトリーです。
2. OpenStack エージェントの sg_silent_config.txt 構成ファイルを編集します。
3. ファイルに示されているパラメーターの値を指定します。応答ファイルには、使用可能なパラメーターと指定する値を定義するコメントが含まれています。
4. ファイルを保存して終了します。
5. システム上の /etc/hosts ファイルを編集して、モニター対象ノードごとにホスト・マッピングを追加します。
6. *install_dir*/samples ディレクトリーから、以下のコマンドを実行してエージェントを構成します。

```
install_dir/bin/openstack-agent.sh config instance_name path_to_response_file
```

ここで、*install_dir* は構成するインスタンスの名前であり、*path_to_responsefile* はサイレント応答ファイルの絶対パスです。このファイルの絶対パスを指定します。

例えば、応答ファイルがデフォルト・ディレクトリーにある場合は、以下のコマンドを実行します。

```
/opt/ibm/apm/agent/bin/openstack-agent.sh config instance_name
/opt/ibm/apm/agent/samples/sg_silent_config.txt
```

タスクの結果
エージェントが構成されました。

次のタスク

- エージェントの構成が完了したら、以下のコマンドを実行してエージェント・インスタンスを開始できます。

```
install_dir/bin/openstack-agent.sh start instance_name
```

ここで、*instance_name* は、構成するエージェント・インスタンスの名前です。

- OpenStack エージェントを SSL 対応の OpenStack 環境に接続するには、以下の環境変数を設定することで、OpenStack サーバーの SSL 証明書のディレクトリーを指定します。

```
OS_cert_path=directory of the certificate.crt file
```

OS_cert_path 変数は、ksg_dc_*instance_name*.cfg ファイルの OS_authentication_info セクションにあります。

- プロセス関連情報を収集する場合は、615 ページの『プロセス関連情報の収集および SSH 接続の有効化』の手順を実行して、OpenStack エージェント用にデータ・コレクターを構成します。
- トラブルシューティングの 目的でエージェントのトレース・レベルを変更する場合は、ファイルの指示に従って、*install_dir*/config/sg.environment ファイルの変数 **KBB_RAS1** の値を編集します。

プロセス関連情報の収集および SSH 接続の有効化

プロセス関連情報を収集する場合は、OpenStack エージェント用にエージェント・データ・コレクターを構成して、ターゲットの OpenStack コンポーネント・サーバーとの SSH 接続をセットアップします。

このタスクについて

OpenStack エージェントを開始する前に、プロセス情報を収集するために SSH 接続をセットアップする必要があります。接続をセットアップするには、製品によって提供され、以下の手順で説明されている **ksg_setup_key.sh** または **ksg_ssh_setup.py** 支援ツールを使用します。

SSH 接続のセットアップに精通している場合は、**ssh-keygen** および **ssh-copy-id** Linux コマンドを使用して接続をセットアップすることもできます。

手順

1. *install_dir*/config ディレクトリーに移動します。ここで、*install_dir* はエージェントのインストール・ディレクトリーです。
2. ksg_dc_*instance_name*.cfg ファイルを編集します。ここで、*instance_name* は当該エージェント・インスタンスに指定した名前です。

   このファイルは、エージェント・インスタンスの開始後に作成されます。ファイルが存在しない場合は、*install_dir*/lx8266/sg/bin/ksg_dc.cfg を *install_dir*/config ディレクトリーにコピーして、ファイル名を ksg_dc_*instance_name*.cfg に変更します。

   例えば、インスタンス名が OS1 の場合は、名前を ksg_dc_OS1.cfg に変更します。
3. ksg_dc_*instance_name*.cfg ファイル内で、パラメーター **collect_process_information** の値を YES に設定します。
4. **OS_process_collection** セクションで、以下の例の形式に従って、OpenStack コンポーネント・サーバーのユーザーとホスト名または IP アドレスを使用して、**ssh_user_host** パラメーターの値を指定します。

   ```
   ssh_user_host=root@9.112.250.248,user1@hostname
   ```
5. 設定を保存します。
6. 設定を有効にするために、以下のコマンドを実行してエージェント・インスタンスを再始動します。

   ```
   install_dir/bin/openstack-agent.sh stop instance_name
   install_dir/bin/openstack-agent.sh start instance_name
   ```
7. 以下のいずれかの方法を使用して、ターゲット・コンポーネント・サーバーとの SSH 接続をセットアップします。
   - ksg_setup_key.sh スクリプトを使用して、接続を 1 つずつセットアップします。*install_dir*/lx8266/sg/bin ディレクトリーに移動し、ホスト名または IP とユーザーを指定して ksg_setup_key.sh スクリプトを実行し、ステップ 4 で指定したコンポーネント・サーバーとの SSH 接続を構築します。ステップ 4 で示されている例に従った場合、以下のようにスクリプトを 2 回実行して、1 つずつ接続をセットアップする必要があります。

     ```
     ./ksg_setup_key.sh 9.112.250.248 root
     ./ksg_setup_key.sh hostname user1
     ```

     注 : 初めてスクリプトを実行する際には、パスワードを指定する必要があります。それ以降はパスワードを指定する必要はありません。
   - OpenStack エージェントによって *install_dir*/lx8266/sg/bin に提供されている ksg_ssh_setup.py ツールを使用して、接続を 1 つずつセットアップするか、バッチ・ジョブでセットアップします。このツールを使用する前に、Python pexpect ライブラリーをインストールする必要があります。
     - SSH 接続を 1 つずつセットアップするには、以下のコマンドを実行します。

       ```
       python ksg_ssh_setup.py -single
       ```

このコマンドは、リモート・ターゲット・サーバーへの SSH 接続をセットアップする場合に役立ちます。以下の情報を指定する必要があります。

```
Enter the remote target machine host name or IP address: (Type 'END' to end input.)
Enter the account to access the remote machine(e.g. root):
Enter the above user's password:
```

– SSH 接続をバッチ・ジョブでセットアップするには、以下のコマンドを実行します。

```
python ksg_ssh_setup.py -ssh SSH_file
```

ここで、*SSH_file* は、ターゲット・サーバー、ユーザー、およびパスワードの情報が含まれているファイルです。Python ツールと同じディレクトリーにある ksg_dc_ssh_list.txt ファイルに従ってファイルを作成し、以下の例のフォーマットに従ってファイルにホストおよびユーザーの情報を指定する必要があります。

```
hostname root passw0rd
9.112.250.248 user1 passw0rd
```

注 : 接続を再度セットアップする必要があるのは、ターゲット・サーバーのユーザー名またはパスワードが変更された場合のみです。エージェントの再始動後やエージェント構成の変更後に、接続を再度セットアップする必要はありません。

タスクの結果

データ・コレクターが構成され、SSH 接続が正しくセットアップされました。これで、Cloud APM コンソールにログインし、アプリケーション・エディターを使用して OpenStack エージェント・インスタンスを Application Performance ダッシュボードに追加できます。エージェント・インスタンスを追加する際、コンポーネント・リストから「**OpenStack 環境 (OpenStack Environment)**」を選択します。

次のタスク

- 「**サービス・タイプ別 API エンドポイントの要約 (API End Point Summary By Sevice Type)**」 > 「**API エンドポイントの詳細**」をクリックすると、グループ・ウィジェット「**API 検出失敗回数ヒストリー**」および「**API 検出失敗率ヒストリー**」に「使用可能なデータがありません」というメッセージが表示されます。「**API エンドポイントの詳細**」に表示されている API エンドポイントをクリックすると、2 つのグループ・ウィジェットでモニター・データを表示できます。
- 「**コンポーネント別のプロセスの要約**」 > 「**プロセスの詳細**」または「**SSH サーバー接続状況**」 > 「**プロセスの詳細**」をクリックすると、グループ・ウィジェット「**プロセス CPU 使用率ヒストリー**」および「**プロセス・メモリー使用率ヒストリー**」に「使用可能なデータがありません」というメッセージが表示されます。 「**プロセスの詳細**」に表示されているプロセスをクリックすると、2 つのグループ・ウィジェットでモニター・データを表示できます。

構成値の追加

ローカル構成とリモート構成の両方で、エージェントが作動するための構成値を指定します。

対話モードを使用してエージェントを構成する場合、パネルが表示されるため、それぞれの値を入力できます。デフォルト値がある場合は、その値がフィールドにあらかじめ入力されています。パスワードのフィールドの場合は、2 つの入力フィールドが表示されます。各フィールドに同じ値を入力する必要があります。 入力した値のセキュリティーを維持できるように、それらの値は表示されません。

サイレント・モードを使用してエージェントを構成する場合、*install_dir*/samples ディレクトリーの *response_file* を編集して、構成値を追加することができます。変更を保存した後、ステップ 614 ページの『6』 の指示に従い、以下のコマンドを実行して、変更を有効にします。

```
install_dir/bin/openstack-agent.sh start instance_name
```

ここで、*instance_name* は、構成するエージェント・インスタンスの名前です。

構成が完了した後、エージェント・インスタンスの .cfg ファイル (例えば、*hostname*_sg_*instance_name*.cfg) で構成した値を見つけることができます。

このエージェントの構成は、以下のグループに編成されます。

OpenStack 環境の認証情報 (OPENSTACK_CONNECTION)
OpenStack 環境の認証情報

このグループで定義される構成要素は、常にエージェントの構成に存在します。

このグループでは、エージェント全体に適用される情報を定義します。

OpenStack 認証 URL (KSG_OPENSTACK_AUTH_URL)
OpenStack 環境の auth_url

タイプはストリングです。

この値は必須です。

デフォルト値: `http://localhost:identity/v3`

OpenStack パスワード (KSG_OPENSTACK_PASSWORD)
管理者ユーザーのパスワード

タイプはパスワードです。

この値は必須です。

デフォルト値: None

OpenStack テナント名 (KSG_OPENSTACK_TENANT_NAME)
OpenStack テナント名。プロジェクト名とも呼ばれます。

タイプはストリングです。

この値は必須です。

デフォルト値: admin

OpenStack ユーザー名 (KSG_OPENSTACK_USERNAME)
OpenStack 環境にログインするための管理者ユーザー

タイプはストリングです。

この値は必須です。

デフォルト値: admin

Python (KSG_PYTHON)
Python 実行可能ファイルの場所

このグループで定義される構成要素は、常にエージェントの構成に存在します。

このグループでは、エージェント全体に適用される情報を定義します。

Python 実行可能ファイルの場所 (KSG_PYTHON_LOCATION)
OpenStack エージェント・データ・コレクターを実行するのに使用される Python 実行可能ファイル。完全修飾パスを確認するには、端末でコマンド "which python" を実行します。

タイプはストリングです。

この値は必須です。

デフォルト値: None

ソケット (KSG_SOCKET)
ソケット・データ・ソース

このグループで定義される構成要素は、常にエージェントの構成に存在します。

このグループでは、エージェント全体に適用される情報を定義します。

ポート番号 (CP_PORT)
ソケット・クライアントからのデータに関してエージェントが listen するために使用するポート。値 0 は、一時ポートが使用されることを示します。このポートは、アプリケーションで使用されるポートには対応していません。このポートは、エージェントが内部で使用するためのものです。

タイプは数値です。

この値はオプションです。

デフォルト値: 0

Oracle データベースのモニターの構成

Monitoring Agent for Oracle Database には、Oracle データベースの可用性、パフォーマンス、およびリソースの使用状況をモニターする機能が用意されています。さまざまな Oracle Database エージェントをモニターするために、複数の Oracle Database インスタンスを構成できます。このエージェントには、リモート・モニター機能も用意されています。

始める前に

- Oracle Database エージェントを構成する前に、Oracle Database エージェントによって使用される Oracle ユーザー・アカウントに特権を付与する必要があります。特権について詳しくは、『Oracle Database エージェントのユーザーへの特権の付与』を参照してください。
- Oracle データベースをリモートでモニターする 場合は、Oracle データベース・ソフトウェアまたは Oracle インスタント・クライアントがインストールされているコンピューターにエージェントをインストールする必要があります。

このタスクについて

以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン』を参照してください。

一般的な Oracle データベース・パフォーマンス・モニターでは、Oracle Database エージェントは、Oracle データベースの可用性、パフォーマンス、リソース使用、およびアクティビティーをモニターします。以下に例を示します。

- モニター対象の Oracle Database 内のインスタンスの可用性
- リソース情報 (メモリー、キャッシュ、セグメント、リソース制限、表領域、UNDO (ロールバック)、システム・メトリック、およびシステム統計など)
- アクティビティー 情報 (OS 統計、セッション、競合、アラート・ログなど)

Oracle Database エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。また、各エージェント・インスタンスは複数のデータベースをモニターできます。

Oracle Database エージェントの管理対象システム名には、指定するデータベース接続名、指定するエージェント・インスタンス名、およびエージェントがインストールされているコンピューターのホスト名が含まれます。例えば、`pc:connection_name-instance_name-host_name:SUB` です。ここで、*pc* は 2 文字の製品コードであり、*SUB* はデータベース・タイプです (指定可能な値は RDB、ASM、または DG です)。管理対象システム名は 32 文字に制限されています。指定するインスタンス名は、23 文字からホスト名およびデータベース接続の字数を除いた長さに制限されています。 例えば、データベース接続名に **dbconn**、エージェント・インスタンス名に **Oracle02** を指定し、ホスト名が *Prod204a* であると、管理対象システム名は `RZ:dbconn-oracle02-Prod204a:RDB` になります。この例では、データベース接続名、エージェント・インスタンス名、およびホスト名に使用できる 23 文字のうち 22 文字を使用しています。

- 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが正しく表示されません。
- *connection_name*、*instance_name*、および *hostname_name* の各変数の合計の長さは、23 文字を超えると切り捨てられます。
- サブノード名の切り捨てを防ぐため、環境変数 **KRZ_SUBNODE_INCLUDING_AGENTNAME**、**KRZ_SUBNODE_INCLUDING_HOSTNAME**、および **KRZ_MAX_SUBNODE_ID_LENGTH** を設定してサブノード命名規則を変更します。

- **KRZ_SUBNODE_INCLUDING_AGENTNAME** を NO に設定すると、サブノード名のサブノード ID 部分にエージェント・インスタンス名が含まれません。例えば、次の場合があります。
  - デフォルトのサブノード名: *DBConnection-Instance-Hostname*
  - 環境変数が NO に設定された場合のサブノード名: *DBConnection-Hostname*
- **KRZ_SUBNODE_INCLUDING_HOSTNAME** を NO に設定すると、サブノード名のサブノード ID 部分にホスト名が含まれません。例えば、次のとおりです。
  - デフォルトのサブノード名: *DBConnection-Instance-Hostname*
  - 環境変数が NO に設定された場合のサブノード名: *DBConnection-Instance*

手順

1. Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
   - 619 ページの『Windows システムでのエージェントの構成』.
   - 627 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. Linux システムおよび UNIX システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。
   - 623 ページの『プロンプトへの応答によるエージェントの構成』.
   - 627 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

拡張構成の場合のみ、Oracle データベース管理者は、Oracle ユーザーが `krzgrant.sql` スクリプトを実行してデータベースにアクセスできるようにする必要があります。krzgrant.sql スクリプトの実行を参照してください。

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動して、更新した値を保存します。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Oracle Database**」テンプレートを右クリックしてから「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 「Monitoring Agent for Oracle Database」ウィンドウで、以下のステップを実行します。

a) Monitoring Agent for Oracle Database・インスタンスの固有のインスタンス名を入力し、「**OK**」をクリックします。

4. 「**ITCAM Extended Agent for Oracle Database の構成**」ウィンドウの「デフォルトのデータベース構成」ペインで、以下のステップを実行します。
   a) **「デフォルト・ユーザー名」**を入力します。これは、データベース接続用のデフォルト・データベース・ユーザー ID です。

      このユーザー ID は、エージェントがモニター対象データベース・インスタンスにアクセスするときに使用する ID です。このユーザー ID には、エージェントが必要とする動的パフォーマンス・ビューおよび表に対する選択特権が必要です。

   b) **「デフォルト・パスワード」**を入力します。これは、指定したデフォルト・データベース・ユーザー ID に関連付けられたパスワードです。

   c) Oracle エージェントのバージョンが 8.0 の場合は、以下のステップを実行します。

      1) **「Oracle JDBC JAR ファイル (Oracle JDBC Jar File)」**を入力します。これは Oracle データベースとの通信に使用される Oracle JDBC ドライバーの jar ファイルの絶対パスです。Oracle エージェントのモニター対象である Oracle データベースのバージョンをサポートする Oracle Java Database Connectivity (JDBC) ドライバーが、エージェント・コンピューター上で使用可能になっていなければなりません。

   d) Oracle エージェントのバージョンが 6.3.1.10 の場合は、以下のステップを実行します。

      1) Oracle Database エージェント がモニター対象の Oracle データベース・サーバーにインストールされている場合は、「**Oracle ホームでライブラリーを使用**」を選択し、**「Oracle ホーム・ディレクトリー」**に入力します。オプションとして、ローカル・モニターでは、「Oracle ホーム・ディレクトリー」設定をブランクのままにすることができます。そうすると、*ORACLE_HOME* システム環境変数が使用されます。

      2) Oracle Database エージェント がモニター対象の Oracle データベース・サーバーのリモートに配置されている場合は、「**Oracle インスタント・クライアントでライブラリーを使用**」を選択し、**「Oracle インスタント・クライアントのインストール・ディレクトリー」**に入力します。

   e) 拡張構成オプションを設定する必要がある場合は、「**拡張オプションの表示**」にチェック・マークを付けます。それ以外の場合は、ステップ 5 に進みます。

   f) 「Net Configuration Files Directories」はブランクのままにすることができます。そうした場合、デフォルト・ディレクトリーが使用されます。Oracle エージェントのバージョンが 6.3.1.10 の場合は、セミコロン (;) を使用してディレクトリーを区切ることで、複数の Net 構成ファイル・ディレクトリーを入力できます。Oracle エージェントのバージョンが 8.0 の場合は、1 つのディレクトリーのみサポートされます。

      この設定には、Oracle Database Net 構成ファイルが含まれます。ディレクトリーは、各 Oracle Database インスタンスの *TNS_ADMIN* 環境変数によって定義されます。デフォルトのディレクトリーは、%ORACLE_HOME%¥NETWORK¥ADMIN です。この項目が構成されていない場合は、デフォルトのディレクトリーが使用されます。デフォルト・ディレクトリーの使用を無効にするには、エージェント環境変数 KRZ_LOAD_ORACLE_NET を false に設定します (KRZ_LOAD_ORACLE_NET=false)。

   g) 「カスタマイズ SQL 定義ファイル名」はブランクのままにします。これは使用されません。

   h) このワークステーションでデフォルトの動的リスナーが構成されているかどうかを選択します。

      デフォルトの動的リスナーは、(PROTOCOL=TCP)(HOST=localhost)(PORT=1521) です。このワークステーションでデフォルトの動的リスナーが構成されている場合は、 この値を Yes に設定します。

   i) 「**次へ**」をクリックします。

5. 「**ITCAM Extended Agent for Oracle Database の構成**」ウィンドウの「**インスタンス構成**」ペインで、以下のステップを実行します。

   ここで、実際のデータベース接続インスタンスを定義します。少なくとも 1 つのタイプを追加しなければなりません。ここは、データベース接続インスタンスを編集および削除する場所でもあります。複数のデータベース接続インスタンス構成が存在する場合は、「**データベース接続**」オプションを使用して、編集または削除するインスタンスを選択します。

a)**「データベース接続」**セクションの「**新規**」を押します。

b) データベースへの接続の別名として**「データベース接続名」**を入力します。

この別名は、データベース接続を表す任意のものにすることができますが、以下の制限があります。接続名に使用できるのは、文字、アラビア数字、下線文字、およびマイナス文字のみです。 接続名の最大長は 25 文字です。

c)**「接続タイプ」**を選択します。

1) (オプション) 基本

デフォルトで、最も一般的な接続タイプは、「**基本**」です。 必要な接続タイプが分からない場合は、この接続タイプを選択することをお勧めします。

a) 標準ファイル・システム・インスタンスや ASM 単一インスタンスなど、ターゲットのモニター対象データベースが単一インスタンスの場合は、「**基本**」接続タイプを選択します。

b) データベースのホスト名または IP アドレスとして「ホスト名」を入力します。

c) データベースによって使用される「ポート」番号を入力します。

d)「**サービス名**」または「**SID**」を選択します。

i.「**サービス名**」を選択した場合、データベースの論理表記であるサービスの名前 (グローバル・データベース・サービス名であるストリング) を入力します。

サービス名は、データベースの論理表記であり、データベースがクライアントに表示される表記です。1 つのデータベースは複数のサービスとして表示でき、また 1 つのサービスは複数のデータベース・インスタンスとして実装できます。サービス名はグローバル・データベース名であるストリングです。この名前は、データベース名とドメイン名で構成され、インストール時またはデータベース作成時に入力されます。グローバルなデータベース名が分からない場合は、初期化パラメーター・ファイルの SERVICE_NAMES パラメーターの値から取得できます。

ii.「**SID**」を選択した場合は、実行中のデータベースの特定のインスタンスを識別する Oracle システム ID を入力します。

これは、データベースの特定のインスタンスを識別する Oracle システム ID です。

ステップ 5d に進んでください。

2) (オプション) TNS

a)「**TNS**」接続タイプを選択するのは、*ORACLE_HOME* システム環境変数が設定されていて、ターゲットのモニター対象データベースの TNS 別名が `$ORACLE_HOME/network/admin/tnsnames.ora` ファイルに定義されている場合です。

b)**「TNS 別名」**を入力します。

ステップ 5d に進んでください。

3) (オプション) Advanced

a) ターゲットのモニター対象データベースの複数の物理ノードにまたがって複数の Oracle インスタンスが存在する場合は、「**Advanced**」接続タイプを選択します。例えば、ASM with Real Applications Cluster (RAC) データベース。

b)**「Oracle 接続文字列」**を入力します。

この属性は、以下のすべての Oracle Net ネーミング・メソッドをサポートしています。

- `//host:port/service name` という形式の SQL 接続 URL ストリング。例えば、`//dlsun242:1521/bjava21` です。
- Oracle Net のキーワードと値の対。例えば、次の場合があります。

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))
```

- **inst1** などの **TNSNAMES** エントリー (*TNS_ADMIN* または *ORACLE_HOME* 環境変数を設定し、構成ファイルを構成します)。

ステップ 5d に進んでください。

d) ステップ 4a とステップ 4b で設定したデフォルト資格情報とは異なる資格情報をこの接続で使用するには、「**別のユーザー名とパスワードを使用**」にチェック・マークを付けます。あるいは、ステップ 5g に進んでください。

e) この接続の**「データベース・ユーザー名」**を入力します。

このユーザー ID は、エージェントがモニター対象データベース・インスタンスにアクセスするときに使用する ID です。このユーザー ID には、エージェントが必要とする動的パフォーマンス・ビューおよび表に対する選択特権が必要です。

f) **「データベース・パスワード」**を入力します。指定したデータベース・ユーザー ID に関連付けられたパスワード。

g) データベース接続の資格情報に付与されている許可に一致する「**役割**」を選択します。

役割は、接続に関連付ける特権セットです。SYSDBA システム特権が付与されたユーザーの場合、その特権が含まれている役割を指定します。ASM インスタンスの場合は、**SYSDBA** 役割または **SYSASM** 役割を使用します。

h) このエージェント・インスタンスからリモート Oracle アラート・ログをモニターする場合は、「**リモート・ログ・モニタリング・オプションの表示**」にチェック・マークを付けます。それ以外の場合は、ステップ 5k に進みます。

i) パスを入力するか、または「**参照**」を使用して**「Oracle アラート・ログ・ファイルのパス」**を選択します。

このデータベース接続においてリモート・データベース・インスタンスにマップされたアラート・ログ・ファイルの絶対ファイル・パス。 エージェントは、これらのファイルを読み取ることでアラート・ログをモニターします。 通常、その場所は、`$ORACLE_BASE/diag/rdbms/`*`DB_NAME`*`/`*`SID`*`/trace/alert_`*`SID`*`.log` です。 例えば、*DB_NAME* と *SID* がともに db11g で、*ORACLE_BASE* が `/home/dbowner/app/oracle` の場合、アラート・ログは、`/home/dbowner/app/oracle/diag/rdbms/db11g/db11g/trace/alert_db11g.log` です。

Windows Oracle Database エージェントが実行され、ネットワークを通してアラート・ログ・ファイルを読み取る場合、リモート・ファイルのパスは Windows システムの汎用命名規則に従わなければなりません。例えば、`¥¥tivx015¥path¥alert_orcl.log` のようになります。

Windows

重要 : パスとアラート・ログ・ファイル名を組み合わせて入力してください。マップされたネットワーク・ドライバーは、アラート・ログのパスに対してサポートされません。

Linux AIX Oracle Database エージェントがリモート・サーバー上にある場合は、リモート・アラート・ログをモニターするには、ローカルでマウントされたファイル・システムが必要です。

Windows 各ファイルをセミコロン (;) で区切ります。

Linux AIX 各ファイルをコロン (:) で区切ります。

各ファイルは、`alert_`*`instance`*`.log` ファイル名パターンを使用してデータベース・インスタンスと照合されます。一致しなかった場合、無視されます。

ローカル・データベース・インスタンスのアラート・ログ・ファイルは、自動的にディスカバーされます。

j) **「Oracle アラート・ログ・ファイルの文字セット」**を選択するか入力します。これは、マップされたアラート・ログ・ファイルのコード・ページです。

このパラメーターが空の場合、システムの現行ロケール設定が使用されます。例えば、以下のようになります。

- ISO8859_1、ISO 8859-1 西ヨーロッパ・エンコード
- UTF-8、Unicode の UTF-8 エンコード
- GB18030、中国語 (簡体字) GB18030 エンコード

- CP950、中国語 (繁体字) エンコード
- EUC_JP、日本語エンコード
- EUC_KR、韓国語エンコード

サポートされるすべてのコード・ページの完全なリストについては、ICU でサポートされるコード・ページ (ICU supported code pages)を参照してください。

k)「**データベース接続**」セクションで、「**適用**」をクリックしてこのデータベース接続インスタンスの設定を保存します。

l) (オプション) 新規データベース接続をテストします。

1)「**データベース接続**」セクションで、新規のデータベース接続を選択します。

2)「**接続のテスト**」をクリックします。

3)「**接続のテスト**」結果ウィンドウで結果を確認します。

- 「**テスト結果**」の成功例:

```
Testing connection config1 ...
成功
```

- 「**テスト結果**」の失敗例:

```
Testing connection config1 ...
KBB_RAS1_LOG; Set MAXFILES to 1
ORA-12514: TNS:listener does not currently know of service requested in connect
descriptor
失敗
```

m)「**次へ**」をクリックします。

6.「**ITCAM Extended Agent for Oracle Database の構成**」ウィンドウの「**要約**」ペインの情報を読んでから、「**OK**」をクリックしてエージェント・インスタンスの構成を終了します。

7.「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Oracle Database**」を右クリックしてから「**スタート**」をクリックします。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

プロンプトへの応答によるエージェントの構成

Linux および UNIX オペレーティング・システムでエージェントを 構成するには、コマンド・ライン構成スクリプトを実行し、そのプロンプトに応答します。

手順

1. *install_dir*/bin ディレクトリーを開きます。ここで、*install_dir* は Oracle Database エージェントのインストール・ディレクトリーです。
2. (オプション) すべての既存の構成済みエージェント・インスタンスの名前をリストするには、コマンド **./cinfo -o rz** を実行します。
3. Oracle Database エージェントを構成するには、コマンド **./oracle_database-agent.sh config** *instance_name* を実行します。
4.「Edit 'Monitoring Agent for Oracle Database' settings」というプロンプトが出されたら、**Enter** キーを押します。デフォルト値は「はい」です。
5.「Default Database Configuration」情報を入力するには、以下のステップを実行します。

注: Default Database Configuration セクションは、データベース接続インスタンスの構成ではありません。これは、ステップ 6 から始まる手順で実際のデータベース接続インスタンスの構成を追加する際に、デフォルト値として使用される内容を設定するためのテンプレート・セクションです。

a)「Default Username」を求めるプロンプトが出されたら、データベース接続のデフォルト・データベース・ユーザー ID を入力し、**Enter** キーを押します。

このユーザー ID は、エージェントがモニター対象データベース・インスタンスにアクセスするときに使用する ID です。このユーザー ID には、エージェントが必要とする動的パフォーマンス・ビューおよび表に対する選択特権が必要です。

b) 「Enter Default Password」というプロンプトが出されたら、指定したデフォルト・データベース・ユーザー ID に関連付けられたパスワードを入力し、**Enter** キーを押します。 次に、プロンプトが出されたら、パスワードを確認します。

c) Oracle エージェントのバージョンが 8.0 の場合は、以下のステップを実行します。

1) **「Oracle JDBC JAR ファイル (Oracle JDBC Jar File)」**を入力します。これは Oracle データベースとの通信に使用される Oracle JDBC ドライバーの jar ファイルの絶対パスです。Oracle エージェントのモニター対象である Oracle データベースのバージョンをサポートする Oracle Java Database Connectivity (JDBC) ドライバーが、エージェント・コンピューター上で使用可能になっていなければなりません。

d) Oracle エージェントのバージョンが 6.3.1.10 の場合は、以下のステップを実行します。

1) 「Oracle Home Directory」を求めるプロンプトが出されたら、Oracle Database エージェントがモニター対象の Oracle データベース・サーバーにインストールされている場合は、Oracle ホーム・ディレクトリーを入力し、**Enter** キーを押します。Oracle Database エージェントがモニター対象の Oracle データベース・サーバーにインストールされていない場合は、この設定をブランクのままにして **Enter** キーを押し、次のステップを実行します。「Oracle Home Directory」のディレクトリーの値をクリアする場合は、スペース・バーを押してから、**Enter** キーを押します。

注 : オプションとして、ローカル・モニターでは、「Oracle Home Directory」と「Oracle Instant Client Installation Directory」はブランクのままにすることができます。そうすると、*ORACLE_HOME* システム環境変数が使用されます。

2) Oracle Database エージェントがモニター対象の Oracle データベース・サーバーのリモートに配置されている場合、「Oracle Instant Client Installation Directory」のディレクトリーを入力し、**Enter** キーを押します。ステップ 624 ページの『5.d.i』 で「Oracle Home Directory」を設定した場合、この値は無視されます。

e) 「Net Configuration Files Directories」はブランクのままにすることができます。そうした場合、デフォルト・ディレクトリーが使用されます。Oracle エージェントのバージョンが 6.3.1.10 の場合は、Windows 「;」または Linux AIX 「:」を使用してディレクトリーを区切ることで、複数の Net 構成ファイル・ディレクトリーを入力できます。Oracle エージェントのバージョンが 8.0 の場合は、1 つのディレクトリーのみサポートされます。**Enter** キーを押します。

この設定には、Oracle Database Net 構成ファイルが含まれます。ディレクトリーは、各 Oracle Database インスタンスの *TNS_ADMIN* 環境変数によって定義されます。デフォルトのディレクトリーは、Linux AIX $ORACLE_HOME/network/admin または Windows %ORACLE_HOME%¥NETWORK¥ADMIN です。この項目が構成されていない場合は、デフォルトのディレクトリーが使用されます。デフォルト・ディレクトリーの使用を無効にするには、エージェント環境変数 KRZ_LOAD_ORACLE_NET を false に設定します (KRZ_LOAD_ORACLE_NET=false)。

f) このワークステーションでデフォルトの動的リスナーが構成されているかどうかを選択して、**Enter** キーを押します。

デフォルトの動的リスナーは、(PROTOCOL=TCP)(HOST=localhost)(PORT=1521) です。このワークステーションでデフォルトの動的リスナーが構成されている場合は、 この値を True に設定します。

g) 「カスタマイズ SQL 定義ファイル名」はブランクのままにします。これは使用されません。

6. 以下の出力が画面に表示された後に、「Edit 'Database Connection' settings」というプロンプトが出されます。

```
Instance Configuration :
Summary :
Database Connection :
```

注 : このステップで、実際のデータベース接続インスタンスを定義します。少なくとも 1 つのタイプを追加しなければなりません。ここは、データベース接続インスタンスを編集および削除する場所でも

あります。複数のデータベース接続インスタンス構成が存在する場合は、「Next」オプションを使用して、編集または削除するインスタンスに到達するまで、編集や削除の必要がないインスタンスをスキップします。

7. 新規データベース接続を追加するには、1 を入力して、**Enter** キーを押します。
8. データベース接続情報を入力するには、以下のステップを実行します。
   a) 「Database Connection Name」を求めるプロンプトが出されたら、データベースへの接続の別名を入力し、**Enter** キーを押します。

      この別名は、データベース接続を表す任意のものにすることができますが、以下の制限があります。接続名に使用できるのは、文字、アラビア数字、下線文字、およびマイナス文字のみです。 接続名の最大長は 25 文字です。

   b) 「Connection Type」を求めるプロンプトが出されたら、以下のいずれかの接続タイプを選択します。

      1) (オプション) 基本

         デフォルトで、最も一般的な接続タイプは、「**基本**」です。 必要な接続タイプが分からない場合は、この接続タイプを選択することをお勧めします。

         a) 標準ファイル・システム・インスタンスや ASM 単一インスタンスなど、ターゲットのモニター対象データベースが単一インスタンスの場合は、「**基本**」接続タイプを選択します。

         b) ホスト名を求めるプロンプトが出されたら、Oracle データベースのホスト名または IP アドレスを入力し、**Enter** キーを押します。

         c) ポートを求めるプロンプトが出されたら、ポート番号を入力し、**Enter** キーを押します。

         d) 次の 2 つの設定のいずれかを入力します。「Service Name」または「SID」。

            i. (オプション) 「Service Name」を求めるプロンプトが出されたら、データベースの論理表記であるサービス名 (グローバル・データベース・サービス名のストリング) を入力し、**Enter** キーを押し、ステップ 8c に進みます。

               サービス名は、データベースの論理表記であり、データベースがクライアントに 表示される表記です。1 つのデータベースは複数のサービスとして表示でき、また 1 つのサービスは複数のデータベース・インスタンスとして実装できます。サービス名はグローバル・データベース名であるストリングです。この名前は、データベース名とドメイン名で構成され、インストール時またはデータベース作成時に入力されます。グローバルなデータベース名が分からない場合は、初期化パラメーター・ファイルの SERVICE_NAMES パラメーターの値から取得できます。ステップ 625 ページの『8.b.i.4.b』 で「SID」を設定する場合、このパラメーターをブランクのままにすることができます。

            ii. (オプション) 「SID」を求めるプロンプトが出されたら、実行中のデータベースの特定のインスタンスを識別する Oracle システム ID を入力し、**Enter** キーを押し、ステップ 8c に進みます。

               このパラメーターは、データベースの特定のインスタンスを識別する Oracle システム ID です。ステップ 625 ページの『8.b.i.4.a』 で Service Name を定義した場合は、この項目をブランクのままにすることができます。

      2) (オプション) TNS

         a) 「**TNS**」接続タイプを選択するのは、*ORACLE_HOME* システム環境変数が設定されていて、ターゲットのモニター対象データベースの TNS 別名が $ORACLE_HOME/network/admin/tnsnames.ora ファイルに定義されている場合です。

         b) TNS 別名を入力し、**Enter** キーを押し、ステップ 8c に進みます。

      3) (オプション) Advanced

         a) ターゲットのモニター対象データベースの複数の物理ノードにまたがって複数の Oracle インスタンスが存在する場合は、「**Advanced**」接続タイプを選択します。例えば、ASM with Real Applications Cluster (RAC) データベース。

         b) Oracle 接続文字列を入力し、**Enter** キーを押し、ステップ 8c に進みます。

この属性は、以下のすべての Oracle Net ネーミング・メソッドをサポートしています。

- //host:port/service name という形式の SQL 接続 URL ストリング。例えば、//dlsun242:1521/bjava21 です。
- Oracle Net のキーワードと値の対。例えば、次の場合があります。

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))
```

- **inst1** などの **TNSNAMES** エントリー (*TNS_ADMIN* または *ORACLE_HOME* 環境変数を設定し、構成ファイルを構成します)。

注 : コマンド・ラインでの構成時に示される説明では、コロンおよび等号の前に円記号 (¥) が付いている場合があります (¥:、¥=)。接続ストリングには円記号 (¥) を入力しないでください。説明でそうした記載になっているのは、コマンドの一部として等号を解釈するという通常の動作をエスケープし、単にテキストとして解釈するためです。

c) ステップ 8c に進んでください。

c) 「Database Username」を求めるプロンプトが出されたら、接続用のデータベース・ユーザー ID を入力し、**Enter** キーを押します。

標準ファイル・システム・インスタンスの場合は、このユーザー ID には、エージェントが必要とする動的パフォーマンス・ビューおよび表に対する選択特権が必要です。

ASM インスタンスの場合は、**SYSDBA** 役割または **SYSASM** 役割を持つアカウントを使用します。例えば、sys アカウントなどを使用してください。

d) 「Enter Database Password」というプロンプトが出されたら、指定したデータベース・ユーザー ID に関連付けられたパスワードを入力します。

e) 「Role」を求めるプロンプトが出されたら、指定したユーザー ID に付与されている許可に一致している役割を選択し、**Enter** キーを押します。

役割は、接続に関連付ける特権セットです。SYSDBA システム特権が付与されたユーザーの場合、その特権が含まれている役割を指定します。

ASM インスタンスの場合は、**SYSDBA** 役割または **SYSASM** 役割を使用します。

f) 「Oracle Alert Log File Paths (including alert log file name)」を求めるプロンプトが出されたら、アラート・ログ・パスを入力し、**Enter** キーを押します。

このパラメーターは、このデータベース接続におけるリモート・データベース・インスタンスのマップされたアラート・ログ・ファイルの絶対ファイル・パス用です。エージェントは、これらのファイルを読み取ることでアラート・ログをモニターします。 通常、その場所は、$ORACLE_BASE/diag/rdbms/*DB_NAME*/*SID*/trace/alert_*SID*.log です。 例えば、*DB_NAME* と *SID* がともに db11g で、*ORACLE_BASE* が /home/dbowner/app/oracle の場合、アラート・ログは、/home/dbowner/app/oracle/diag/rdbms/db11g/db11g/trace/alert_db11g.log です。

Windows Oracle Database エージェントが実行されていて、ネットワークを介してアラート・ログ・ファイルを読み取る場合、リモート・ファイルのパスは Windows システムの汎用命名規則に従っている必要があります。例えば、¥¥tivx015¥path¥alert_orcl.log のようになります。

重要 : パスとアラート・ログ・ファイル名を組み合わせて入力してください。マップされたネットワーク・ドライバーは、アラート・ログのパスに対してサポートされません。

Linux AIX Oracle Database エージェントが実行されている場合は、リモート・アラート・ログに対して、ローカルにマウントされたファイル・システムが必要です。

Windows 各ファイルをセミコロン (;) で区切ります。

Linux AIX 各ファイルをコロン (:) で区切ります。

各ファイルは、alert_*instance*.log ファイル名パターンを使用してデータベース・インスタンスと照合されます。一致しなかった場合、無視されます。

ローカル・データベース・インスタンスのアラート・ログ・ファイルは、自動的にディスカバーすることができます。

g) 「**Oracle Alert Log File Charset**」を求めるプロンプトが出されたら、マップされたアラート・ログ・ファイルのコード・ページを入力し、**Enter** キーを押します。

このパラメーターが空の場合、システムの現行ロケール設定が使用されます。例えば、以下のようになります。

- ISO8859_1、ISO 8859-1 西ヨーロッパ・エンコード
- UTF-8、Unicode の UTF-8 エンコード
- GB18030、中国語 (簡体字) GB18030 エンコード
- CP950、中国語 (繁体字) エンコード
- EUC_JP、日本語エンコード
- EUC_KR、韓国語エンコード

サポートされるすべてのコード・ページの完全なリストについては、ICU でサポートされるコード・ページ (ICU supported code pages)を参照してください。

9. 「Edit 'Database Connection' settings」というプロンプトが再び出されたら、ステップ 8a で設定したデータベース接続の名前が表示されます。再度編集したり、削除したりすることができます。複数のデータベース接続インスタンスが既に構成されている場合は、「**Next**」を使用して各インスタンスを順に対象にします。
10. (オプション) このエージェント・インスタンスで複数のデータベース・インスタンスをモニターするために別のデータベース接続を追加する場合は、1 と入力し、**Enter** キーを押してステップ 8 に戻ります。
11. データベース接続の変更が終了したら、5 と入力し、**Enter** キーを押して構成プロセスを終了します。
12. エージェントを開始するには、以下を入力します。
install_dir/bin/oracle_database-agent.sh start *instance_name*。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. テキスト・エディターで oracle_silent_config.txt ファイルを開きます。
   - Linux AIX *install_dir*/samples/oracle_database_silent_config.txt.
   - Windows *install_dir*¥samples¥oracle_database_silent_config.txt
2. 「**Default Username**」に、このエージェント・インスタンス用に作成するデータベース接続のデフォルトのデータベース・ユーザーの名前を入力します。 例えば、**KRZ_CONN_USERID=**user1 です。

注: このユーザーは、データベースへの接続中にこのエージェントで実行するタスク (表の照会など) を実行するための十分な特権を備えている必要があります。

3. **Default Password** には、指定したデフォルト・データベース・ユーザーに関連付けられたパスワードを入力する必要があります。例えば、**KRZ_CONN_PASSWORD=**Password です。
4. Oracle エージェントのバージョンが 8.0 の場合は、以下のステップを実行します。
   a) **「Oracle JDBC JAR ファイル (Oracle JDBC Jar File)」**を入力します。これは Oracle データベースとの通信に使用される Oracle JDBC ドライバーの jar ファイルの絶対パスです。

   Oracle エージェントのモニター対象である Oracle データベースのバージョンをサポートする Oracle Java Database Connectivity (JDBC) ドライバーが、エージェント・コンピューター上で使用可能になっていなければなりません。
5. Oracle エージェントのバージョンが 6.3.1.10 の場合は、以下のステップを実行します。
   a) Oracle Database エージェントがモニター対象の Oracle データベース・サーバーにインストールされている場合、Oracle ホーム・ディレクトリーを入力します。例えば、**KRZ_ORACLE_HOME=**home_path です。

   注: このようなオプション・パラメーターを使用するには、先頭のハッシュ記号 (#) を削除します。

   Oracle Database エージェントがモニター対象の Oracle データベース・サーバーにインストールされていない場合は、この設定をブランクのままにし、次のステップを実行します。

   注: オプションとして、ローカル・モニターの場合、「Oracle Home Directory」および「Oracle Instant Client Installation Directory」をブランクのままにすること (サイレント構成テキスト・ファイルで、パラメーター行の先頭位置にハッシュ記号 (#) を使用してコメント化すること) ができます。そうすると、*ORACLE_HOME* システム環境変数が使用されます。

   b) Oracle Database エージェントがモニター対象の Oracle データベース・サーバーのリモートに配置されている場合、「Oracle Instant Client Installation Directory」のディレクトリーを入力します。前のステップで「Oracle Home Directory」のディレクトリーを入力した場合、この値は無視されます。

   - Windows Oracle Call Interface (OCI) ライブラリー・ファイルが含まれている **Oracle Home** ディレクトリーの絶対フォルダー・パスを定義します。oci.dll ファイルの絶対パスが C:¥instantclient_10_2¥oci.dll の場合、その C:¥instantclient_10_2 パスを定義する必要があります。例えば、**KRZ_INSTANT_CLIENT_LIBPATH=**C:¥instantclient_10_2 です。
   - AIX Oracle Call Interface (OCI) ライブラリー・ファイルが含まれている **Oracle Home** ディレクトリーの絶対フォルダー・パスを定義します。libocci.so.10.1 ファイルの絶対パスが /home/tivoli/oci/libocci.so.10.1 の場合、その /home/tivoli/oci パスを定義する必要があります。例えば、**KRZ_INSTANT_CLIENT_LIBPATH=**/home/tivoli/oci です。
6. 「Net Configuration Files Directories」はブランクのままにすることができます。そうした場合、デフォルト・ディレクトリーが使用されます。Oracle Database エージェントはこのファイル・パスを使用して、tnsnames.ora ファイルを取得します。このディレクトリーは、各 Oracle Database インスタンスの *TNS_ADMIN* 環境変数によって定義されます。デフォルトのディレクトリーは、Linux AIX $ORACLE_HOME/network/admin または Windows %ORACLE_HOME%¥NETWORK¥ADMIN です。この設定で複数の Net 構成ファイル・ディレクトリーを入力する場合は、Windows 「;」または Linux AIX 「:」を使用してそれらのディレクトリーを区切ります。

   Oracle データベースをリモート側でモニターしている場合、リモート・システムの Net 構成ファイルを、エージェントがインストールされているシステムにコピーできます。また、リモート・システムの Net 構成ファイルの内容を、エージェントがインストールされているシステムの Net 構成ファイルにマージすることもできます。
7. **Dynamic listener** の場合、デフォルトの動的リスナーを構成されているかどうかを確認します。デフォルトの動的リスナーは、(PROTOCOL=TCP)(HOST=localhost)(PORT=1521) です。 デフォルトの動的リスナーが構成されている場合は、この値を TRUE に設定します (**KRZ_DYNAMIC_LISTENER=**TRUE)。

有効な値は TRUE と FALSE です。

8. 「カスタマイズ SQL 定義ファイル名」はブランクのままにします。これは使用されません。
9. ここから、実際のデータベース接続インスタンスを定義します。少なくとも 1 つのタイプを追加しなければなりません。`oracle_silent_config.txt` には、インスタンス名 *config1* の 1 つのインスタンスの項目が示されています。インスタンス名を変更した場合は、必ず、すべての参照を変更してください。

   この別名は、データベース接続を表す任意のものにすることができますが、以下の制限があります。接続名に使用できるのは、文字、アラビア数字、下線文字、およびマイナス文字のみです。 接続名の最大長は 25 文字です。
10. **Connection Type** には、**Basic**、**TNS**、または **Advanced** のいずれかの接続タイプを指定します。例えば、**KRZ_CONN_TYPE.config1=**`Basic` です。
11. 前のステップで選択した接続タイプについて、以下の必須パラメーターを指定します。

    基本

    - **Hostname** には、Oracle データベースのホスト名または IP アドレスを指定します (例えば、**KRZ_CONN_HOST.config1=** `hostname`)。
    - **Port** には、Oracle データベースのリスナー・ポートを指定します (例えば、**#KRZ_CONN_PORT.config1=** 1521)。
    - **Service Name** には、グローバル・データベース名のストリングを使用して、データベースの論理表記を指定します (例えば、**KRZ_CONN_SERVICE.config1=**`orcl`)。

      重要 : Service Name を定義しない場合、Oracle システム ID (SID) を指定する必要があります。

      Oracle System Identifier (SID) には、実行中のデータベースの特定のインスタンスを識別する SID を指定します (例えば、**KRZ_CONN_SID.config1=** `sid`)。

    TNS

    TNS alias には、`tnsnames.ora` ファイルからのネットワーク別名を指定します。例えば、**KRZ_CONN_TNS.config1=** `tnsalias` です。

    拡張

    Oracle Connection String には、OCI のデータベース接続ストリングを指定します。例えば、**KRZ_CONN_STR.config1=** `//host:port/service` です。

    このストリングでは、以下に示すような、すべての Oracle Net ネーミング・メソッドがサポートされます。

    - 以下の SQL 接続 URL ストリング:

      ```
      //host:[port][/service name]
      ```

    - 以下のような Oracle Net のキーワードと値の対:

      ```
      "(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
      (CONNECT_DATA=(SERVICE_NAME=bjava21)))"
      ```

    このストリングでは、**TNSNAMES** エントリー (例えば、**inst1**) もサポートされます (*TNS_ADMIN* または *ORACLE_HOME* 環境変数が設定され、構成ファイルが構成されている場合)。

    重要 : この属性は、拡張タイプの接続にのみ適用されます。
12. **Database Username** には、接続用のデータベース・ユーザーの名前を指定できます (例えば、**KRZ_CONN_USERID=**`UserID`)。

    このユーザーは、データベースへの接続中にエージェントが必要とするタスク (表の作成、編集、削除など) を実行するための十分な特権を備えている必要があります。

    このフィールドを空にすると、エージェントは、デフォルトのデータベース構成セクション内にあるデフォルトのユーザー名を使用します。**Database Username** を構成しなかった場合、この接続にはデフォルトのユーザー名が使用されます。

13. **Database Password** には、指定したデータベース・ユーザーに関連付けられたパスワードを指定できます (例えば、**KRZ_CONN_PASSWORD=**Passsword)。

    このフィールドを空にすると、エージェントは、デフォルトのデータベース構成セクション内にあるデフォルトのパスワードを使用します。**Database Password** を構成しなかった場合、この接続にはデフォルトのパスワードが使用されます。
14. **Role** には、接続に関連付けられた特権セットを指定できます (例えば、**KRZ_CONN_MODE.config1=**DEFAULT)。

    有効な値には、*SYSDBA*、*SYSOPER*、*SYSASM*、および *DEFAULT* があります。

    SYSDBA システム特権を付与されているユーザーに対しては、この特権が含まれた接続を指定できます。この項目が定義されていない場合は、*DEFAULT* 役割をユーザーに割り当てることができます。
15. **「Oracle アラート・ログ・ファイルのパス」**では、アラート・ログ・ファイル名を含める場合に、このデータベース接続でリモート・データベース・インスタンス用にマップされたアラート・ログ・ファイルの絶対ファイル・パスを指定できます。例えば、**KRZ_LOG_PATHS.config1=**AlertLogPath です。

    Windows 複数のファイルを区切るには、セミコロン (;) を使用します。

    Linux AIX 複数のファイルを区切るには、コロン (:) を使用します。

    各ファイルは、alert_*instance*.log ファイル名パターンによってデータベース・インスタンスと照合されます。また、一致しなかった場合、無視されます。

    ローカル・データベース・インスタンスのアラート・ログ・ファイルは、自動的にディスカバーされます。

    「Oracle アラート・ログ・ファイルのパス」が構成されていない場合、アラート・ログは使用できません。
16. **Oracle Alert Log File Charset** には、マップされたアラート・ログ・ファイルのコード・ページを指定できます。例えば、**KRZ_LOG_CHARSET.config1=** CharSet です。

    このフィールドが空の場合、以下に示すシステムの現行ロケール設定が使用されます。

    ```
    ISO8859_1: ISO 8859-1 Western European encoding
    UTF-8: UTF-8 encoding of Unicode
    GB18030: Simplified Chinese GB18030 encoding
    CP950: Traditional Chinese encoding
    EUC_JP: Japanese encoding
    ```
17. oracle_database_silent_config.txt ファイルを保存して閉じます。次に、以下のように入力します。
    install_dir/bin/oracle_database-agent.sh config *instance_name install_dir*/samples/oracle_database_silent_config.txt
    ここで、*instance_name* は、インスタンスに指定する名前です。
18. エージェントを開始するには、以下を入力します。
    install_dir/bin/oracle_database-agent.sh start *instance_name*。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Oracle Database エージェントのユーザーへの特権の付与

エージェントをインストールしたら、Oracle Database エージェントによって使用される Oracle ユーザー・アカウントに特権を付与する必要があります。

以下のユーザーに特権を付与できます。

- 標準ファイル・システム (非 ASM) インスタンス・ユーザー

- ASM with RAC インスタンスの非 SYS ユーザー

標準ファイル・システム・インスタンスのユーザーに対する特権の付与
標準ファイル・システム・インスタンスの場合、Oracle Database エージェントが使用する Oracle ユーザー ID には、エージェントが必要とする動的パフォーマンス・ビュー、表、およびデータ・ディクショナリー・ビューに対する選択特権が必要です。また、一部のデータベース・コマンドの実行に必要な他の Oracle オブジェクト特権やシステム特権も必要です。

手順

1. (オプション) Oracle データベース・ユーザー ID が存在しない場合は、Oracle の機能を使用し、次のコマンドを実行してこの ID を作成します。`create user UserName identified by Password`
2. 動的パフォーマンス・ビュー、表、およびデータ・ディクショナリー・ビューに対する選択特権を、作成した Oracle ユーザー ID に付与します。このためには、Oracle Database エージェントに用意されている **krzgrant.sql** スクリプトを実行します。このステップは、エージェントの構成前に行う必要があります。**krzgrant.sql** スクリプトのカスタマイズと実行の手順については、631 ページの『krzgrant.sql スクリプトの実行』および 632 ページの『krzgrant.sql スクリプトの実行』を参照してください。

   注 : 動的パフォーマンス・ビュー、表、およびデータ・ディクショナリー・ビューに対する選択特権は、特定のアプリケーション環境での Oracle データベースの機能に依存します。Oracle データベース・ユーザー ID には、Oracle Database エージェントにより使用される動的パフォーマンス・ビュー、表、およびデータ・ディクショナリー・ビューのみに対する承認済みの Oracle 特権を付与できます。
3. Oracle の機能を使用して、Oracle Database エージェントにより使用される Oracle ユーザー ID に他の Oracle オブジェクト特権およびシステム特権を付与します。

***krzgrant.sql* スクリプトの実行**
krzgrant.sql スクリプトで一部の動的パフォーマンス・ビュー、表、およびデータ・ディクショナリー・ビューに対する Oracle 承認済み選択特権を許可しないようにする場合は、**krzgrant.sql** スクリプトを実行前にカスタマイズできます。

注 : エージェント・インスタンスにより、**krzgrant.sql** スクリプトのすべてのデフォルト特権が検査され、エージェント開始時に特権が不足しているエージェント・イベントが報告されます。特権の検査を無効にするには、変数設定 KRZ_CHECK_ORACLE_PRIVILEGE=FALSE を使用します。 GUI 構成の接続テスト・ステップでは、krzgrant.sql ファイルに定義されているすべての Oracle 特権が検査されます。Oracle ユーザーに正しい特権が付与されていることを確認している場合は、テスト接続ステップでの特権検査の失敗は無視してください。

プレーン・テキスト・エディターで krzgrant.sql ファイルを編集し、grant ステートメントの先頭で「--」プレフィックスを削除または追加します。このプレフィックスを追加すると、未検査の Oracle 表またはビューの特権付与の実行がスキップされます。

例えば次の行を変更します。

```
    execute immediate 'grant select on DBA_HIST_SNAPSHOT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQLSTAT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQLTEXT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQL_PLAN to '||userName;
    execute immediate 'grant select on DBA_HIST_SYSMETRIC_SUMMARY to '||userName;
```

変更後は次のようになります。

```
--    execute immediate 'grant select on DBA_HIST_SNAPSHOT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQLSTAT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQLTEXT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQL_PLAN to '||userName;
--    execute immediate 'grant select on DBA_HIST_SYSMETRIC_SUMMARY to '||userName;
```

ASM インスタンスの非 SYS ユーザーに対する特権の付与
ユーザーの SYSDBA の役割および SYSASM の役割を使用する、ASM インスタンスに接続する必要があります。ASM インスタンスに接続するために SYS アカウントを使用しない場合は、ユーザー・アカウントを作成して、そのアカウントに SYSDBA の役割および SYSASM の役割を付与します。

手順

1. 以下のコマンドを実行して、ユーザー・アカウントを作成し、役割を付与してください。
   - SYSASM の役割を使用して ASM データベースにログインし、エージェントの新規ユーザーを作成し、SYSDBA の役割または SYSASM の役割を付与します。
     a. `create user UserName identified by Password`
     b. `grant sysdba to UserName`

     または

     `grant sysasm to UserName`
2. 構成ウィンドウで ASM 接続を作成する際、*UserName* ユーザーと SYSDBA または SYSASM 役割を指定します。

   注: ASM データベースにアクセスするために SYSASM の役割を選択した場合は、Oracle ホームまたは Oracle インスタント・クライアントを使用して、Oracle データベースに接続するようにエージェント・インスタンスを構成する必要があります。

krzgrant.sql スクリプトの実行

始める前に

- **krzgrant.sql** スクリプトを実行しない場合、エージェントのイベント・ワークスペースでイベントが生成されます。
- インストール手順の実行については、123 ページの『第 6 章 エージェントのインストール』を参照してください。

インストール後、**krzgrant.sql** スクリプトは以下のディレクトリーにあります。

- Windows *install_dir*¥TMAITM6_X64
- Linux AIX *install_dir*/*architecture*/rz/bin

ここで、

install_dir
Oracle Database エージェントのインストール・ディレクトリー。

アーキテクチャー
IBM Application Performance Management または Cloud APM システム・アーキテクチャー ID。 例えば、lx8266 は、Linux Intel v2.6 (64 ビット) を表します。アーキテクチャー・コードの完全なリストについては、*install_dir*/registry/archdsc.tbl ファイルを参照してください。

krzgrant.sql スクリプトの使用法は `krzgrant.sql user_ID temporary_directory` です。

ここで、

user_ID
Oracle ユーザーの ID。 このユーザー ID は、この SQL ファイルの実行前に作成する必要があります。値の例: *tivoli*

temporary_directory
krzgrant.sql スクリプトの出力ファイル krzagent.log を含む一時ディレクトリーの名前。 このディレクトリーは、この SQL スクリプトを実行する前に存在していなければなりません。値の例: install_dir/tmp

以下の手順を実行するには、Oracle Database 管理者 (DBA) の許可の役割と、一時ディレクトリーへの書き込み権限が必要です。

手順

1. コマンド行からコマンドを実行して環境変数を設定します。

- Windows

```
SET ORACLE_SID= sid
SET ORACLE_HOME= home
```

- Linux AIX

```
ORACLE_SID = sid
export ORACLE_SID
ORACLE_HOME = home
export ORACLE_HOME
```

ここで、

sid
Oracle システム ID (大/小文字が区別されます)。

home
モニター対象 Oracle インスタンスのホーム・ディレクトリー。

2. 環境変数を設定した同じコマンド行ウィンドウから、Oracle SQL Plus、または SQL ステートメントの実行に使用する代替ツールを開始します。
3. Oracle DBA 特権を持つユーザーとして、Oracle Database にログオンします。
4. **krzgrant.sql** スクリプトを含むディレクトリーに移動し、以下のコマンドを実行して選択特権を付与します。

```
@krzgrant.sql user_ID temporary_directory
```

出力は、一時ディレクトリー内の krzagent.log ファイルに記録されます。このログには、Oracle Database エージェントに選択特権が付与されるビューおよび表が記録されます。

特権の付与が正常に終了したら、Oracle Database エージェントを構成して開始することができます。

OS のモニターの構成

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS、および Monitoring Agent for Windows OS のエージェントは自動的に構成されます。これらの OS エージェント用にログ・ファイル・モニターを構成すれば、アプリケーションのログ・ファイルをモニターできます。OS エージェントは非 root ユーザーとして実行できます。また、Linux OS エージェントには追加の構成オプションがいくつかあります。

root 以外のユーザーとしての OS エージェントの実行

Monitoring Agent for Windows OS、Monitoring Agent for UNIX OS、および Monitoring Agent for Linux OS は root 以外のユーザーとして実行できます。

root 以外のユーザーとして Windows OS エージェントを実行するには、634 ページの『root 以外のユーザーとしての Monitoring Agent for Windows OS の実行』を参照してください。

root 以外のユーザーとして Monitoring Agent for UNIX OS および Monitoring Agent for Linux OS のエージェントを実行するには、1013 ページの『非 root ユーザーとしてのエージェントの開始』を参照してください。

制約事項 :

root 以外のユーザーとして実行している場合、エージェントは /proc/pid/status にアクセスできないため、以下の属性を報告できません。

- -ユーザー CPU 時間 (UNIXPS.USERTIME)
- -システム CPU 時間 (UNIXPS.SYSTEMTIM)
- -合計 CPU 時間 (UNIXPS.TOTALTIME)
- -スレッド数 (UNIXPS.THREADCNT)
- -子ユーザー CPU 時間 (UNIXPS.CHILDUTIME)

- -子システム CPU 時間 (UNIXPS.CHILDSTIME)
- -合計子 CPU 時間 (UNIXPS.CHILDTIME)
- -待機 CPU 時間 (UNIXPS.WAITCPUTIM)
- -端末 (UNIXPS.USERTTY)

これらの属性は Cloud APM コンソールには表示されませんが、しきい値の作成に使用できます。

root 以外のユーザーとしての Monitoring Agent for Windows OS の実行

Windows OS エージェントは、root 以外のユーザーとして実行できます。ただし、使用できない機能があります。

root 以外のユーザーとして Windows OS エージェントを実行するとき、以下の属性グループを管理者アカウントのみが所有している場合は、これらの属性グループで一部の機能を使用できません。

- レジストリー
- ファイル・トレンド
- ファイル変更

他のエージェントをリモート・デプロイすることはできません。新規エージェントをインストールするには管理者権限が必要であるためです。

エージェント管理サービスに関して、Watchdog は、停止または開始の特権を持っていないエージェントを停止または開始することはできません。

root でないユーザーを作成するには、新しい制限付き (非 root) ユーザーを作成し、その新規ユーザーのレジストリー権限を下記の例のようにセットアップします。

- HKEY_LOCAL_MACHINE¥SOFTWARE¥Candle に対する全アクセス権限
- HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows NT¥CurrentVersion¥Perflib に対する読み取り権限

Monitoring Agent for Windows OS – Primary サービスを開始するユーザーは、Monitoring Agent for Windows OS - Watchdog サービスを管理するための権限を持っている必要があります。Monitoring Agent for Windows OS - Watchdog サービスを開始するユーザーは、エージェント管理サービスによって管理されるすべてのサービス (Monitoring Agent for Windows OS – Primary サービスなど) を管理する権限も持っている必要があります。Windows のシステム・サービスを管理する権限をユーザーに付与するには、セキュリティー・テンプレートまたはグループ・ポリシーを使用するか、Subinacl.exe ファイルを編集します。詳しくは、Microsoft の資料 http://support.microsoft.com/kb/325349(http://support.microsoft.com/kb/325349) を参照してください。

セキュリティー・テンプレートを使用することでシステム・サービスを管理するための権限をユーザーに付与する方法を以下の例に示します。

1. 「**スタート**」 > 「**ファイル名を指定して実行**」をクリックし、「名前」ボックスに mmc と入力して「**OK**」をクリックします。
2. 「**ファイル**」メニューで「**スナップインの追加と削除**」をクリックします。
3. 「**追加**」 > 「**セキュリティの構成と分析**」とクリックし、再び「**追加**」をクリックします。
4. 「**閉じる**」をクリックしてから「**OK**」をクリックします。
5. コンソール・ツリーで「**セキュリティの構成と分析**」を右クリックして、「**データベースを開く**」をクリックします。
6. データベースの名前と場所を指定して、「**開く**」をクリックします。
7. 「**テンプレートのインポート**」ダイアログ・ボックスが表示されたら、インポートするセキュリティー・テンプレートをクリックして、「**開く**」をクリックします。
8. コンソール・ツリーで「**セキュリティの構成と分析**」を右クリックして、「**コンピュータの分析**」をクリックします。

9. 「**分析の実行**」ダイアログ・ボックスが表示されるので、「エラー ログ ファイルのパス」ボックスに表示されたログ・ファイルのデフォルトのパスを受け入れます。あるいは、任意の場所を指定します。「**OK**」をクリックします。
10. 分析が完了したら、次の手順を実行してサービスの権限を構成します。
    a. コンソール・ツリーで、「**システム サービス**」をクリックします。
    b. 右側のペインで Monitoring Agent for Windows OS - Primary サービスをダブルクリックします。
    c. 「**このポリシーをデータベースに定義する**」チェック・ボックスを選択して、「**セキュリティの編集**」をクリックします。
    d. 新しいユーザーまたはグループの権限を構成するには、「**追加**」をクリックします。
    e. 「**ユーザー、コンピュータまたはグループの選択**」ダイアログ・ボックスに、権限を設定するユーザーまたはグループの名前を入力して、「**OK**」をクリックします。 「**ユーザーまたはグループの権限 (Permissions for User or Group)**」リストで「**許可**」チェック・ボックス (「**開始**」の横) を選択します。停止および一時停止の権限はデフォルトで選択されているため、ユーザーやグループはサービスの開始、停止、または一時停止が可能です。
    f. 「**OK**」を 2 回クリックします。
11. ステップ 10 を繰り返して、Monitoring Agent for Windows OS - Watchdog サービスに対するサービス権限を構成します。
12. 新しいセキュリティー設定をローカル・コンピューターに適用するために、「**セキュリティの構成と分析**」を右クリックして、「**コンピュータの構成**」をクリックします。

注 : Secedit コマンド行ツールを使用してシステム・セキュリティーを構成および分析することもできます。Secedit について詳しくは、「**スタート**」 > 「**ファイル名を指定して実行**」をクリックし、cmd と入力して「**OK**」をクリックします。コマンド・プロンプトで `secedit /?` と入力して、**ENTER** キーを押します。この方法を使用して設定を適用すると、テンプレートにあるすべての設定が再適用されます。この方法を使用すると、以前に構成した他のファイル、レジストリー、またはサービス権限がオーバーライドされる場合があります。

Windows のサービス・コンソールを使用することで root 以外のユーザーとしてログオンするための Monitoring Agent for Windows OS および Watchdog サービスを設定する方法を以下の例に示します。

1. 「**スタート**」 > 「**ファイル名を指定して実行**」をクリックし、`services.msc` と入力して「**OK**」をクリックします。
2. 「**Monitoring Agent for Windows OS - Primary**」を選択します。
3. 「**プロパティ**」を右クリックします。
4. 開始タイプが自動になっていることを確認します。
5. 「**ログオン**」タブを選択し、「**このアカウントでログオン (Log on as "This account")**」を選択して、ID とパスワードを入力します。「**OK**」をクリックします。
6. 「**Monitoring Agent for Windows OS - Watchdog**」を選択します。
7. 「**プロパティ**」を右クリックします。
8. 開始タイプが手動になっていることを確認します。
9. 「**ログオン**」タブを選択し、「**このアカウントでログオン (Log on as "This account")**」を選択して、ID とパスワードを入力します。「**OK**」をクリックします。

OS エージェント・ログ・ファイル・モニターの構成

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS、および Monitoring Agent for Windows OS のエージェントは自動的に構成されます。ただし、アプリケーション・ログ・ファイルをモニターできるように、OS エージェントに対してログ・ファイル・モニターを構成できます。

ログ・データはエージェントでフィルタリングされた後、ログ・イベントの形式で Cloud APM コンソールに送信されます。

OS エージェント用のログ・ファイル・モニター構成の追加または削除

OS エージェント用のログ・ファイル・モニター構成を追加して、OS エージェントがログ・ファイル・データをフィルタリングできるようにします。その後、必要に応じて、OS エージェント用のログ・ファイル・モニター構成を削除することもできます。

始める前に

OS エージェントにはサンプルの regex1.conf ファイルと regex1.fmt ファイルが含まれるようになり、.conf ファイルおよび .fmt ファイルを構成する前にこれらを参照できます。これらのファイルは以下の場所にあります。

- UNIX/LINUX の場合: `<install_dir>/samples/logfile-monitoring`
- Windows の場合: `<install_dir¥samples¥logfile-monitoring`

テキスト・エディターを使用して、構成ファイル `.conf` とフォーマット・ファイル `.fmt` を作成します。これらのファイルの内容について詳しくは、640 ページの『構成ファイル』および 649 ページの『フォーマット・ファイル』を参照してください。これらのファイルは、必ず Performance Management コンソールにアクセスするシステムに保存して、Cloud APM サーバーにアップロードできるようにしてください。

このタスクについて

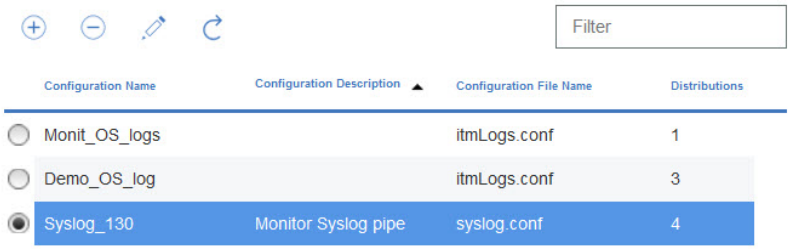
OS エージェントがログ・ファイルをモニターできるようにするには、構成ファイルおよびフォーマット・ファイルをアップロードして、構成の適用先 OS エージェントを指定する必要があります。この OS エージェントが `.conf` ファイルと `.fmt` ファイルをダウンロードして、構成内に指定したログ・ファイルをモニターします。

手順

OS エージェント用のログ・ファイル・モニター構成の追加

1. 「**システム構成**」 > 「**エージェント構成**」をクリックします。
2. ログ・ファイルをモニターするシステムに応じて、「**Unix OS**」タブ、「**Linux OS**」タブ、または「**Windows OS**」タブのいずれかをクリックします。
3. 新規構成を作成するには、(+) アイコンをクリックして 「**新規ログ・ファイルの構成**」ウィンドウを開きます。構成の名前と構成の説明を入力します。
4. `.conf` ファイルと `.fmt` ファイルの内容を表示するには、「**表示**」をクリックします。
5. Cloud APM サーバーを使用して構成をアップロードするには、Performance Management コンソールを開いたのと同じシステムから `.conf` ファイルと `.fmt` ファイルを選択し、「**完了**」をクリックします。
6. OS エージェント・タブで、アップロードした構成を選択します。

   重要 : エージェントに配布された `.conf` ファイルおよび `.fmt` ファイルの名前は、定義した構成名に変更されます。



7. 構成をデプロイするには、「**ログ構成の配布リスト**」表で、構成のデプロイ先エージェントを選択し、「**変更の適用**」をクリックします。

OS エージェント用のログ・ファイル・モニター構成の削除

8. 構成名を選択します。
9. 管理対象システムをクリアし、「**変更の適用**」をクリックします。

   重要 :

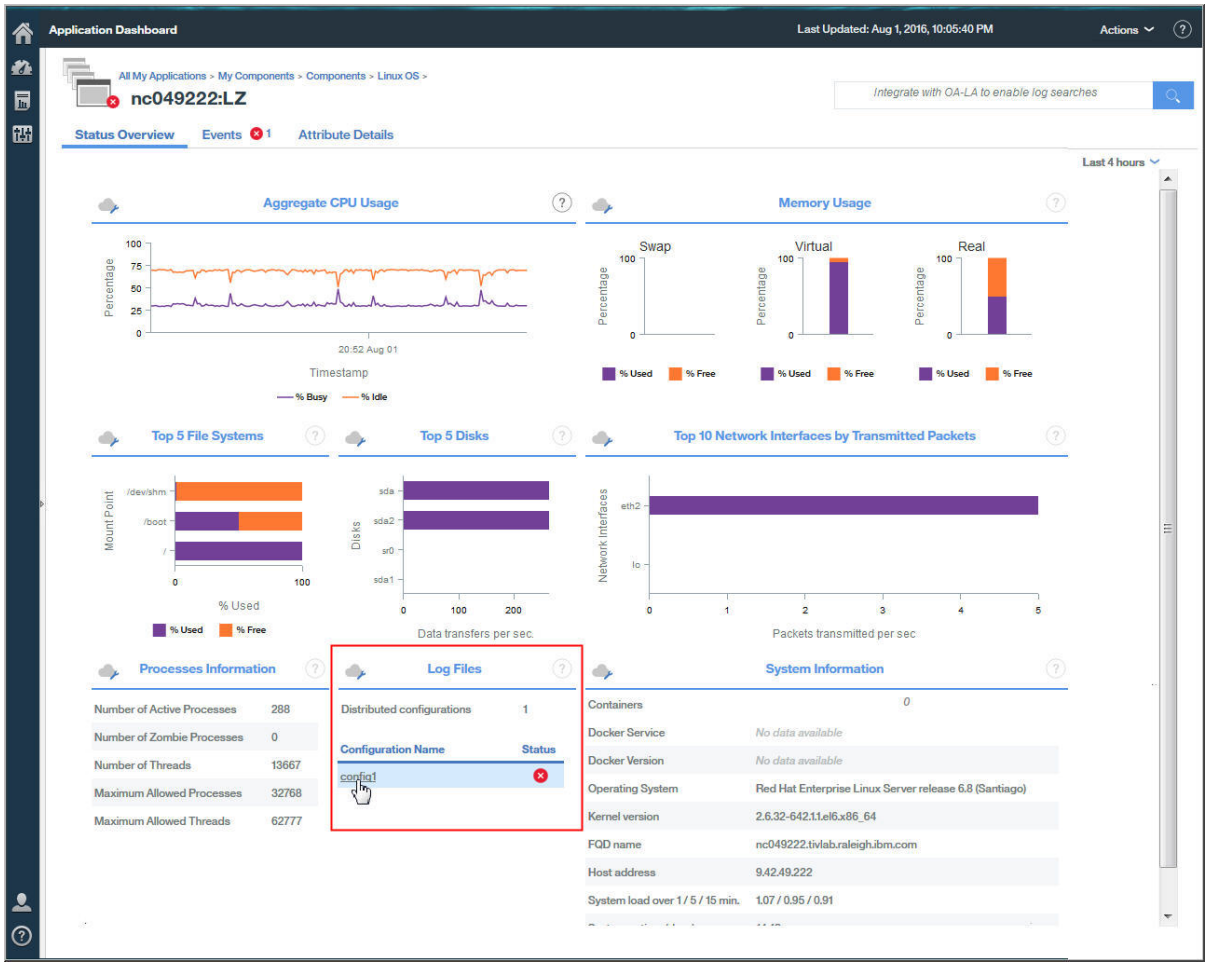
ログ・モニター構成を削除した後も、ログ・ファイル・モニター・リソースは維持され、OS エージェントを再始動するまでオンラインのままになります。オフラインのログ・ファイル・モニター・リソースは、「**オフライン・システム除去の遅延**」オプションで指定された時刻が経過するとクリアされます。

ログ・ファイル・モニターの内容の表示

ログ・ファイルをモニターするためにデプロイした、OS エージェント用のログ・ファイル・モニター構成を表示できます。

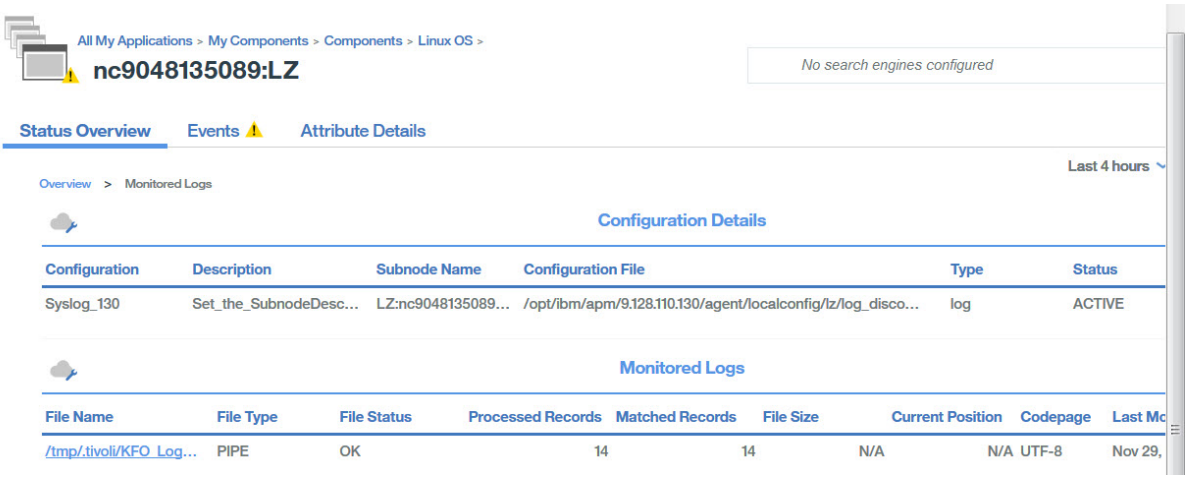
手順

1. 「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックし、ログ・ファイル・モニター構成をデプロイした OS エージェントを含むアプリケーションを選択します。
2. OS エージェント・ダッシュボードにドリルダウンし、「ログ・ファイル」ウィジェットでプロファイルをクリックして、配布されているログ・モニター構成とモニター対象ログを表示します。

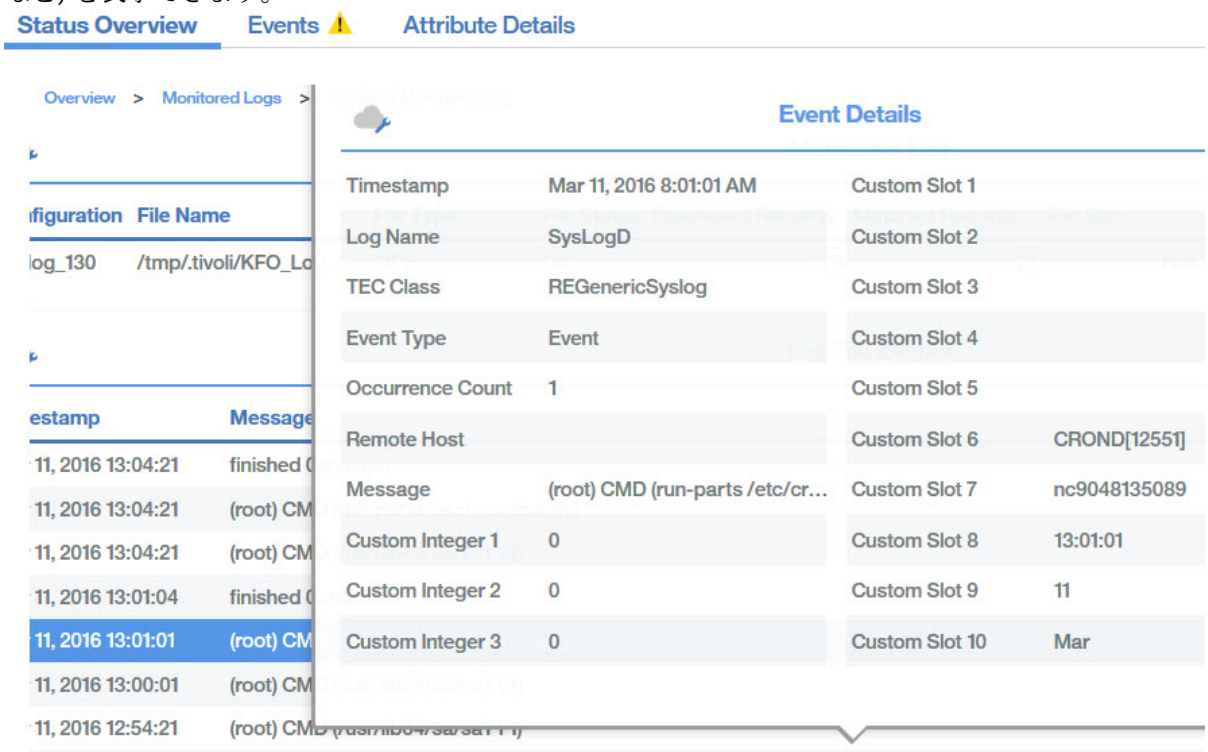


構成の詳細には、構成名、説明、サブノード、構成ファイル、状況、およびエラー・コードが含まれています。

3. ログ・ファイル名をクリックして、ログ・ファイルに関連付けられたログ・ファイル・イベントをすべて表示します。



4. イベントをクリックして、イベントの詳細 (フォーマット・ファイル内に定義したすべてのフィールドなど) を表示できます。



ログ・ファイルのモニター・イベントの表示

アプリケーションのログ・ファイルをモニターするよう OS エージェントを構成したら、しきい値を作成して、警報を受けたいログ・ファイル条件の発生時にアラームを生成できます。

手順

1. ナビゲーション・バーで、「**システム構成**」 > 「**しきい値マネージャー**」をクリックします。
2. 「**データ・ソース・タイプ**」でターゲット OS を選択します。
3. 「追加」をクリックして、新しいしきい値を作成します。
4. このしきい値を超えるイベントの重大度を設定します。
5. しきい値を作成するデータ・セットを選択します。ログ・ファイルのモニターが可能なデータ・セットは以下のとおりです。
   - *Kpp* ログ・ファイル正規表現統計
   - *Kpp* ログ・ファイル状況

- K*pp* ログ・ファイル・プロファイル・イベント

6. ⊕「**追加**」をクリックして、新しい条件を追加します。「**条件の追加**」ボックスで、属性と演算子を選択し、しきい値の値を入力します。

   必要に応じて、このステップを繰り返して、しきい値にさらに条件を追加します。
7. 「グループ割り当て」セクションで、しきい値を割り当てるリソース・グループを選択します。
8. 「**保存**」をクリックします。
9. ナビゲーション・バーで、「**システム構成**」 > 「**拡張構成**」をクリックします。
10. 「**UI 統合**」カテゴリーで、「**サブノード・イベントの有効化**」の値を「True」に設定します。
11. 「**保存**」をクリックします。

タスクの結果
指定した条件が true になると、アラートをトリガーしたログ・ファイル・イベントが「イベント」タブに表示されます。

ログ・ファイル・モニター環境変数
ログ・ファイル・モニターの環境変数は、OS エージェントの環境ファイルで設定できます。

以下の環境変数を設定して、K*PC* を OS エージェント・コードに置き換えます。ここで *PC* は、2 文字のエージェント・コードです。例えば、klz は Linux OS エージェントのコードです。

K*PC*_FCP_LOG
この変数は、*install_dir*/config/.*pc*.environment ファイルにあります。デフォルト値は True です。この変数を使用してログ・モニター機能の有効/無効を切り替えます。

K*PC*_FCP_LOG_PROCESS_MAX_CPU_PCT
この設定値は、すべてのシステム CPU のうち、エージェントが 1 分の間隔内で使用する最大許容率を表します。有効な値は 5 から 100 です。デフォルト値は 100 です。この設定は、CPU スロットル機能と関連付けられています。5 より小さい値を指定すると、最小値である 5 が使用されます。

K*PC*_FCP_LOG_PROCESS_PRIORITY_CLASS
この設定は、プロセスに対するオペレーティング・システムのスケジューラー優先順位です。A が最低の優先順位で、C がオペレーティング・システムのデフォルト、F が最高の優先順位です。設定値は、A、B、C、D、E、F のいずれかです。これらの値は、.conf ファイルに指定した値に置き換えられます。

K*PC*_FCP_LOG_SEND_EVENTS
デフォルト設定は True です。OS エージェントが Cloud APM サーバーにイベントを送信する場合に、この設定が使用されます。

K*PC*_FCP_LOG_SEND_EIF_EVENTS
デフォルトの設定は True です。このオプションを Yes に設定した場合、エージェントはイベント・データを Cloud APM サーバーまたは任意の EIF 受信側 (OMNIbus EIF プローブなど) に送信します。このオプションを No に設定した場合、エージェントはイベント・データを送信しません。このオプションの設定はグローバルであり、すべてのモニタリング・プロファイルに適用されます。

注: イベントは EIF 受信側で使用します。そうしないと、エージェント・キャッシュがいっぱいになった場合に問題が発生する可能性があります。

K*PC*_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN
ログ・ファイル・イベント・モニターを設定すると、OS エージェントではサブノードの 制限が発生します。ログ・ファイル・イベントを管理するために、サブノード MSN の構造は UX:*CITRAHOSTNAME_PROFILENAME* になっています。サブノード名の最大サイズ制限は、32 文字です。生成されるサブノード MSN 名が長すぎて 32 文字を超える場合、32 文字で切り捨てられます。この名前は、プロファイル名から取得するサブストリングに対応しています。

長すぎるプロファイル名を管理するには、OS エージェント構成ファイルで以下の変数を使用します。

- UNIX OS エージェント: KUX_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true
- Linux OS エージェント: KLZ_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true
- Windows OS エージェント: KNT_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true

例えば、長さが 20 文字の aixhost_nc123456789A というエージェントの場合、CTIRAHOSTNAME=aixhost_nc123456789A は 20 文字です。

さらに、以下の名前の 2 つのプロファイルがあるとします。

```
ProfileLong12A (14 characters)
ProfileLong12B (14 characters)
```

関連付けられるサブノード MSN として、以下が予期されます。

```
UX:aixhost_nc123456789A_ProfileLong12A (38 characters)
UX:aixhost_nc123456789A_ProfileLong12B (38 characters)
```

しかし、サブノード MSN が 32 文字の制限に切り捨てられるため、生成される名前は以下のように両方とも同じになります。

```
UX:aixhost_nc123456789A_ProfileL
UX:aixhost_nc123456789A_ProfileL
```

プロファイル名ではなく CTIRAHOSTNAME を切り捨てるには、*Kpc_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true* 変数を設定します。

例えば、*n* をプロファイル名の長さとし、n が 14 であるとすると、*CTIRAHOSTNAME* に関連付けられる MSN 名のサブストリングは 32-n-3 文字に切り捨てられます。そのため、*CTIRAHOSTNAME* 変数は aixhost_nc1234 になります。よって、識別されたサブノード MSN は以下のようになります。

```
UX:aixhost_nc1234_ProfileLong12A
UX:aixhost_nc1234_ProfileLong12B
```

構成ファイル
OS エージェントは、開始時にエージェントによって読み取られる構成ファイルを使用します。このファイルには、構成オプションおよびフィルターが 含まれます。この構成ファイルを作成して、それを使用するためのエージェント・インスタンスを構成してください。

構成ファイルは、その後 60 秒ごとに、そのタイム・スタンプの変更がモニターされます。ファイルのタイム・スタンプが変更された場合、エージェントは、その構成を動的に再初期化します。エージェントの再始動は不要です。詳しくは、653 ページの『エージェントの構成ファイルとフォーマット・ファイルの変更』を参照してください。

OS エージェントの .conf ファイルは、以下のオプションを受け入れます。

codepage
このパラメーターは、モニター対象ファイルのコード・ページです。モニター対象ファイルのコード・ページがシステムのコード・ページと異なる場合は、構成ファイルでこのパラメーターを使用してください。モニター対象ファイルのコード・ページを指定します (ibm-5348_P100-1997、UTF-16、UTF-8 など)。

ConfigFilesAreUTF8=Y
このパラメーターは、構成ファイルおよびフォーマット・ファイルが UTF-8 であると指定します。構成ファイルのエンコードは UTF-8 であるがシステムのコード・ページのエンコードは異なる場合は、このパラメーターを使用してください。デフォルトでは、エージェントはシステムのエンコードを想定します。

DupDetectionKeyAttributes
重複するイベントを判別するために使用する Cloud APM 属性のコンマ区切りリスト。指定されたすべての属性が 2 つのイベントで同じである場合、それらの 2 つのイベントは重複イベントであると見なされます。 このオプションは、イベントだけに適用されます。 詳しくは、1018 ページの『イベント・フィルタリングおよび要約』を参照してください。

注 :

1. 属性名は大/小文字が区別されるため、示すとおりの名前を入力する必要があります。
2. 属性のリストを指定しなかった場合、値はデフォルトで Class および Logname になります。

ENFORCE_STRICT_TEC_COMPATIBILITY
このパラメーターは、ログ・データ内のすべての空白文字を参照して、空白文字が考慮されるようにします。たとえば、ログ・メッセージから情報を抽出するために "%s %s" などのフォーマットを使用すると、OS エージェントは、リテラル・スペースだけでなく、それ以外のタブや復帰として存在している空白文字も突き合わせます。

このパラメーターを設定しない場合、OS エージェントが Tivoli Enterprise Console® スタイル書式制御ストリングを突き合わせるときのデフォルト動作では、フォーマットを左から右に処理しながら、できるだけ多くの入力テキストを突き合わせます。

例えば、書式制御ストリングが %s:%s であり、入力ストリングが one:two:three である場合、OS エージェントはデフォルトでは、one.two を最初のパラメーター (最初の %s に対応) に割り当て、three を 2 番目のパラメーターに割り当てます。

注:

1. このパラメーターは、正規表現構文を使用するフォーマット指定ステートメントには適用されません。
2. このパラメーターを設定すると、パフォーマンスに影響を与えます。マッチングの動作およびパフォーマンスをより効果的に制御するためには、このパラメーターの設定を回避し、代わりに正規表現を使用してください。

EventSummaryInterval
エージェントが抑止対象の重複イベントを検索する秒数を指定します。 このパラメーターを正の整数に設定してください。このオプションは、イベントだけに適用されます。 詳しくは、1018 ページの『イベント・フィルタリングおよび要約』を参照してください。

EventFloodThreshold
重複イベントが検出されたときに送信するイベントを指定します。 このパラメーターは send_none、send_all、send_first、または正の整数に設定してください。このオプションは、イベントだけに適用されます。 詳しくは、1018 ページの『イベント・フィルタリングおよび要約』を参照してください。

EventMaxSize
生成されるイベントの最大サイズをバイト単位で指定します。このパラメーターを指定した場合、パラメーターは 2 カ所で使用されます。

1. このパラメーターは、イベントの処理に使用されるバッファーのサイズを設定するために、エージェントが使用することができます。このパラメーターを指定しない場合、デフォルトのバッファー・サイズとして 16384 バイトが使用されます。バッファーの設定値が小さすぎる場合、イベントは切り捨てられ、場合によっては廃棄されます。
2. このパラメーターは、イベントを OMNIbus EIF プローブなどの EIF 受信側に送信する目的で使用されるバッファーのサイズを設定するために、EIF 送信側が使用することができます。このパラメーターを指定しない場合、デフォルトのバッファー・サイズとして 4096 バイトが使用されます。バッファーの設定値が小さすぎる場合、イベントは廃棄されます。

FileComparisonMode
複数のファイルがワイルドカード・パターンに一致した場合にモニターするログ・ファイルを指定します。使用可能な値は以下のとおりです。

CompareByAllMatches
この値はデフォルトの動作です。LogSources で指定されたワイルドカード・パターンに一致するすべてのファイルがモニターされます。

CompareByLastUpdate
LogSources で指定されたワイルドカード・パターンに一致するファイルのうち、更新のタイム・スタンプが最も新しいファイルがモニターされます。

CompareBySize
ファイル名パターン基準に一致する 2 つ以上のファイルのうち、より大きなファイルがモニター対象として選択されます。同時に更新されてファイル・サイズが増大する一致ファイルが複数ある場合には、CompareBySize を使用しないでください。 最大のファイルが頻繁に変更される場合、モニタリングは常に、新規に選択されたファイルの先頭から再開される可能性があります。 複数のフ

ファイルが一致し、そのうち 1 つのみがアクティブであり特定の時点に更新される場合には、代わりに CompareBySize を使用してください。

CompareByCreationTime
LogSources で指定されたワイルドカード・パターンに一致するファイルのうち、作成のタイム・スタンプが最も新しいファイルがモニターされます。この値には、以下の制限があります。

- UNIX および Linux オペレーティング・システムではファイルの実際の作成時刻は保管されないため、値は Windows オペレーティング・システムにのみ適用されます。
- セキュア・シェル (SSH) ファイル転送プロトコルを使用してモニターするリモート・ファイルでは、この値はサポートされません。

ヒント: CompareByLastUpdate、CompareBySize、および CompareByCreationTime の各値は、ローリング・ログ・ファイルに使用できます。CompareByLastUpdate は通常、これらのファイルに使用されます。

FQDomain
エージェントがドメイン名を設定するかどうか、およびその設定方法を以下のように指定します。

- yes に設定すると、エージェントがシステム・ドメイン名を決定します。
- no に設定すると、エージェントはドメイン名を設定しません。 fqhostname 属性には空白ストリングが割り当てられます。
- yes 値も no 値も含まれないように設定すると、ドメイン名が値として受け入れられ、ホスト名の後に追加されます。

詳しくは、649 ページの『フォーマット・ファイル』を参照してください。

IncludeEIFEventAttr
このエージェントには、*EIFEvent* という大きな属性が含まれています。この属性は、Event Integration Facility 機能が使用可能になっている場合に、この機能によって送信されるイベントを表しています。*EIFEvent* 属性に含まれている情報は、他の属性にも含まれていることがあります。 サイズが大きいことが原因で問題が発生することがあったため、デフォルトでは無効にされました。この値を y に設定すると、EIFEvent 属性が再有効化されます。

注: この属性を使用すると、大きなイベントがある場合にしきい値が失敗することがあります。このコンテキストにおける大きなイベントとは、すべての属性およびその名前の値をすべて含めるために必要なバイト数を合計すると 3600 バイトを超えるストリングになるイベントです。

LognameIsBasename
y に設定すると、Logname 属性の値が、イベントが検出されたログ・ファイルのベース名になります。このオプションは、Performance Management イベントにのみ適用されます。パスは削除されます。たとえば、/data/logs/mylog.log は mylog.log になります。この値を n に設定すると、絶対パスが得られます。 ただし、属性の長さは 64 文字までに制限されているため、n に設定すると、それより長い名前の部分は切り捨てられます。 したがって、デフォルト値は y になっています。長い属性で絶対パス名を表示するには、.fmt ファイル内のフォーマットのマッピング・セクションで、例えば filename FILENAME CustomSlot1 のように指定することができます。このマッピングにより、filename という名前のスロットに、イベントが検出されたファイルの絶対パスが入り、そのスロットが 256 文字の長さの CustomSlot1 にマップされます。

LogSources
メッセージをポーリングするテキスト・ログ・ファイルを指定します。各ファイルへの完全パスを指定する必要があり、ファイル名をコンマで区切る必要があります。 それぞれのファイル名内では、アスタリスク (*) を使用して任意の一連の文字を表したり、疑問符 (?) を使用して任意の 1 文字を表したりすることもできます。 例えば、mylog* と指定すると、mylog で始まる名前を持つすべてのログ・ファイルがポーリングされ、mylog??? と指定すると、mylog とその後に続く厳密に 3 文字で構成される名前を持つすべてのログ・ファイルがポーリングされます。これらのワイルドカード文字は、ファイル名の中でのみサポートされます。パスは明示的に指定する必要があります。

パスで正規表現またはパターン・マッチングを使用する場合は、RegexLogSources の説明を参照してください。

ログ・ファイル・ソースは、エージェントの開始時点で存在している必要はありません。ログ・ファイルは作成されたときにポーリングされます。

NewFilePollInterval
エージェントがモニター対象の新規ファイルを検査する頻度を秒単位で指定します。 例えば、エージェントの開始時に *LogSources* または *RegexLogSources* 構成ファイルの設定で指定された名前のファイルがまだ存在していない場合、この時間間隔が経過した後でそのファイルの存在の有無を再検査します。

NumEventsToCatchUp
ログ内の、エージェントの開始点になるイベントを指定します。このオプションを使用すると、モニター対象のソースが新規である場合またはエージェントが長時間停止していた場合の柔軟性がある程度高くなります。 有効な値は以下のとおりです。

注: テキスト・ファイルの場合、値 `0` および `-1` が適用されます。Windows イベント・ログの場合、値 `0`、`-1`、および n が適用されます。

0
ログ内の次のイベントから開始します。この値はデフォルトです。

-1
`-1` に設定すると、エージェントはモニター対象のファイルにおけるエージェントの位置を保存します。 位置が保存されるため、エージェントが停止して後で再開したときに、停止中にログに書き込まれたイベントを処理することができます。 それ以外の場合、エージェントは停止中に到着したイベントを無視し、ファイルの末尾から再開します。 この設定は、パイプ、または UNIX および Linux システムでの syslog モニタリングには適用されません。

n
正の整数に設定してください。ログ内の最新イベントから *n* 番目のイベントから開始します。つまり、ログ内の最新イベントより前の *n* 件のイベントが開始されます。 *n* が選択可能なイベントの数より大きい場合、選択可能なすべてのイベントが処理されます。

注: 値 n は、Windows イベント・ログの場合にのみ使用することができます。値 n は、`UseNewEventLogAPI` が *y* に設定されている場合には無視されます。

PollInterval
`LogSources` オプションでリストされた各ログ・ファイルをポーリングして新規メッセージの有無を調べる頻度を、秒単位で指定します。 デフォルト値は 5 秒です。

前のリリースから Windows イベント・ログ・アダプターをアップグレードして 、Windows レジストリーで `PollingInterval` の値を設定している場合、Windows レジストリーで使用した値と同じ値をエージェント構成ファイルの `PollInterval` オプションに指定する必要があります。このルールは、レジストリーに値が保管されている Tivoli Enterprise Console OS エージェントを置き換える場合にのみ適用されます。

ProcessPriorityClass
エージェントの処理優先順位を指定します。エージェントが大ボリュームのイベントを処理していて、使用するプロセッサー・リソースが多すぎる場合、この値を調整してシステム・パフォーマンスを向上させることができます。 指定できる値は以下のとおりです。

- `A` - 非常に低い優先順位
- `B` - 低い優先順位
- `C` - 標準的な優先順位
- `D` - 標準より高い優先順位
- `E` - 高い優先順位
- `F` - 非常に高い優先順位
- `USE_CONF_FILE_VALUE` - 構成ファイルに指定された値を使用します。この値はデフォルトです。

RegexLogSources
メッセージをポーリングするテキスト・ログ・ファイルを指定します。このオプションは、ファイル名のベース名部分およびファイル名のうち 1 つのサブディレクトリーで正規表現のメタ文字が使用でき

る点で、LogSources オプションと異なっています。この違いにより、複数のディレクトリーにあるモニター対象の複数ファイルを、LogSources オプションよりはるかに柔軟に記述することができます。

例えば、LogSources ステートメントで /var/log/mylog* を指定することは、RegexLogSources ステートメントで、ドット (.) メタ文字を使用し、その後にアスタリスク (*) メタ文字を使用して、/var/log/mylog.* にするのと同じ意味を持ちます。 このタイプの修飾子を使用すると、/var/log ディレクトリー内の、ベース名が mylog で始まってその後にゼロ個以上の文字が続く、すべてのログ・ファイルがポーリングされます。 /var/log/mylog.+ という修飾子を使用すると、/var/log ディレクトリー内の、名前が mylog で始まってその後に 1 つ以上の文字が続く、すべてのログ・ファイルがポーリングされます。

LogSources と同様に、各ファイルへの完全パスを指定する必要があり、ファイル名をコンマで区切る必要があります。 ただし、コンマも正規表現内の有効な文字です。正規表現の一部として使用されるコンマと、ファイル名を区切るのに使用されるコンマを区別するため、正規表現の一部として使用されるコンマは、円記号 (¥) 文字を付けてエスケープする必要があります。

例えば、正規表現 /logs/.*¥.log および /other/logs/[a-z]{0,3}¥.log のいずれかに一致するログを検索する場合は、2 番目の正規表現の {0,3} 節のコンマを以下のようにエスケープして、エージェントがこのコンマを新しい正規表現の始まりと間違えることがないようにする必要があります。
RegexLogSources=/logs/.*¥.log,/other/logs/[a-z]{0¥,3}¥.log

パス名でメタ文字を使用した場合、そのメタ文字はパスの 1 つのサブディレクトリーでのみ使用できます。 例えば、/var/log/[0-9¥.]*/mylog.* と指定すると、1 つのサブディレクトリーでメタ文字を使用できます。[0-9¥.]* を指定すると、/var/log 内の、番号とドット (.) のみで構成されるサブディレクトリーが突き合わされます。mylog.* を指定すると、これらの /var/log サブディレクトリー内の、mylog で始まり、ゼロ個以上の文字が続くファイル名が突き合わされます。

一部のオペレーティング・システムでは円記号 (¥) がディレクトリー分離文字として使用されるため、正規表現のエスケープ・メタ文字と混同される可能性があります。この混同を避けるため、ディレクトリーを表すには常にスラッシュを使用する必要があります。例えば、Windows ファイルを C:¥temp¥mylog.* と指定した場合、¥t がタブ文字の短縮表記と解釈される可能性があります。そのため、すべてのオペレーティング・システムのディレクトリー 分離文字として常にスラッシュ (/) を使用してください。例えば C:/temp/mylog.* は、C:/temp ディレクトリー内の、mylog で始まるすべてのファイルを表します。

複数のサブディレクトリーにメタ文字が含まれている場合、トレース・メッセージの発行も行われます。 例えば、c:/[0-9¥.]*/temp.files/mylog.* には、メタ文字が使用された 2 つのサブディレクトリーがあります。[0-9¥.]* はメタ文字を使用した最初のサブディレクトリーであり、temp.files はドット (.)・メタ文字を使用した 2 番目のサブディレクトリーです。 この場合エージェントは、メタ文字を含む最初のサブディレクトリーを使用し、メタ文字を含む後続のディレクトリーは無視するものと見なします。

SubnodeName
モニタリング・プロファイル・サブノードに割り当てられたデフォルト名をオーバーライドするために使用できるストリング値。 デフォルトでは、モニタリング・プロファイルに割り当てられたサブノード名は、そのプロファイルに使用される構成ファイルのベース名に対応します。この設定を使用することにより、別のサブノード名を割り当てることができます。

SubnodeDescription
LFAProfiles の *Subnode Description* 属性に値を割り当てるために使用できるストリング値。

UnmatchLog
エージェントによってイベント・クラスに解析できない廃棄済みイベントを記録するファイルを指定します。 その後、廃棄済みイベントを分析して、エージェントのフォーマット・ファイルを修正する必要があるかどうかを判別できます。*DISCARD* を使用したパターンに一致するイベントは、パターンに一致したため、不一致ログには表示されません。

このオプションは、フォーマット・ファイル内のフィルターを検証するために、テスト環境で使用されます。このオプションを長時間にわたってオンのままにすると、ファイル・システムがいっぱいになってしまいます。

SSH を使用するリモート・ログ・ファイル・モニターのオプション

リストである **SshHostList** を除く、すべてのオプションで指定できる値はそれぞれ 1 つのみです。この値は **SshHostList** に指定されるすべてのリモート・ホストに適用されます。

サポートされるのはテキスト・ログ・ファイルのみです。AIX エラー・レポート、syslog、および Windows イベント・ログはサポートされません。

ヒント : syslog を設定して、syslog の出力をテキストのログ・ファイルに書き込み、OS エージェントを使用して、そのテキスト・ファイルをリモートでモニターすることができます。

SshAuthType
PASSWORD または *PUBLICKEY* のいずれかに設定する必要があります。*PASSWORD* に設定した場合、**SshPassword** の値は、すべてのリモート・システムでの SSH 認証で使用されるパスワードとして処理されます。 *PUBLICKEY* に設定した場合、**SshPassword** の値は、秘密鍵ファイルへのアクセスを制御するパスフレーズとして処理されます。*PUBLICKEY* に設定する場合は、**SshPrivKeyfile** と **SshPubKeyfile** も指定する必要があります。

SshHostList
モニター対象のリモート・ホストのコンマ区切りリスト。**LogSources** または **RegexLogSources** ステートメントに指定されているすべてのログ・ファイルは、ここにリストされる各ホストでモニターされます。指定されているホスト名のいずれかが *localhost* である場合、エージェントはローカル・システム上の同じファイル・セットを直接モニターします。*localhost* を指定すると、ローカル・システム上のファイルにアクセスする際に SSH は使用されず、ログ・ファイルが直接読み取られます。

SshPassword
SshAuthType の値が *PASSWORD* である場合、この値は、**SshUserid** に指定されているユーザーのアカウント・パスワードになります。 アカウント・パスワードは平文で指定することができます。あるいは、IBM Tivoli Monitoring CLI **itmpwdsnmp** コマンドで暗号化されたパスワードを指定することもできます。**itmpwdsnmp** コマンドによるパスワードの暗号化方法について詳しくは、658 ページの『リモート・ログ・ファイル・モニター: パスワードまたはパスフレーズの暗号化』を参照してください。

SshAuthType の値が *PUBLICKEY* である場合、この値は、**SshPrivKeyfile** パラメーターで指定されている秘密鍵の暗号化を解除するパスフレーズとなります。パスフレーズは平文で指定することができます。あるいは、IBM Tivoli Monitoring CLI **itmpwdsnmp** コマンドで暗号化されたパスフレーズを指定することもできます。**itmpwdsnmp** コマンドによるパスワードの暗号化方法について詳しくは、658 ページの『リモート・ログ・ファイル・モニター: パスワードまたはパスフレーズの暗号化』を参照してください。

注 : SshAuthType の値が *PUBLICKEY* であり、パスフレーズを必要としないように SSH を構成した場合は、**SshPassword** を NULL に設定する必要があります。 **SshPassword** を NULL に設定する場合、構成ファイル内の項目は以下のようになります。

```
SshPassword=
```

SshPort
SSH 用の接続先 TCP ポート。これを設定しない場合、デフォルトの *22* に設定されます。

SshPrivKeyfile
SshAuthType が *PUBLICKEY* に設定されている場合、この値は、**SshUserid** に指定されているユーザーの秘密鍵を含むファイルへの絶対パスでなければなりません。また、**SshPubKeyfile** も設定する必要があります。**SshAuthType** が *PUBLICKEY* に設定されていない場合、この値は必要ないため、無視されます。

SshPubKeyfile
SshAuthType が *PUBLICKEY* に設定されている場合、この値は、**SshUserid** に指定されているユーザーの公開鍵を含むファイルへの絶対パスでなければなりません。また、**SshPrivKeyfile** も設定する必要があります。**SshAuthType** が *PUBLICKEY* に設定されていない場合、この値は必要ないため、無視されます。

SshUserid
エージェントが SSH 認証で使用するリモート・システムでのユーザー名。

UNIX システムおよび Linux システムのみでサポートされるオプション

Linux AIX

AutoInitSyslog

このオプションを Yes に設定すると、エージェントは、イベントの標準的なセットをエージェントのモニター対象のパイプに書き込むように、syslog 機能を自動的に構成します。 この設定を有効にすると、ログ・ファイルの保守およびロールオーバーを行わずに syslog イベントをモニターすることができます。構成ファイルでこのオプションを設定しない場合、No に設定した場合と同じになります。

制約事項 : このオプションは、リモート・ログ・ファイル・モニターではサポートされていません。

Windows システムのみでサポートされるオプション

Windows

NTEventLogMaxReadBytes

Windows システムでイベント・ログ・データを読み取るために古い NT イベント・ログ・インターフェースを使用している場合 (UseNewEventLogAPI が y に設定されていない場合)、エージェントは、イベント・ログで新規データの有無を検査するたびにこのバイト数まで読み取ります。この値を 0 に設定すると、エージェントは、前のリリースの場合と同様にすべての新規データを読み取ろうとします。 このアクティビティーによって、多くのイベントが発生するシステムではエージェントがかなりの時間占有されることがあります。 デフォルト値は 655360 です。この値を設定した場合、エージェントは、指定どおりの正確な値で停止するのではなく、この値に最も近い、内部バッファー・サイズの倍数で停止します。

PreFilter

エージェントで処理を行う前に Windows イベント・ログ内のイベントをフィルタリングする 方法を指定します。PreFilter ステートメントは、フィルターがイベント・ログからエージェントに送信するイベントを決定するときに PreFilterMode によって使用されます。PreFilter ステートメント内のそれぞれの *attribute=value* 指定がイベント・ログ内のイベントと一致したときに、イベントは PreFilter ステートメントと一致したと見なされます。 PreFilter ステートメントには少なくともログ指定が含まれている必要があり、オプションで 3 つまでの追加指定 (イベント ID、イベント・タイプ、およびイベント・ソース) を含めることができます。ステートメント内での属性の順序は任意です。

PreFilter ステートメントの基本的な形式は次のとおりです。

```
PreFilter:Log=log_name;EventId=value; EventType=value;Source=value;
```

属性ごとに、それぞれの値をコンマで区切って複数の値を指定することができます。

各 PreFilter ステートメントは 1 行で指定する必要があります。

PreFilter は必須ではありません。 PreFilter が指定されていなくて、 PreFilterMode=OUT になっている場合、すべての Windows ログ・イベントがエージェントに送信されます。

PreFilterMode

このオプションは Windows イベント・ログにのみ適用されます。このオプションは、PreFilter ステートメントに一致する Windows システム・ログ・イベントを送信するか (PreFilterMode=IN) 無視するか (PreFilterMode=OUT) を指定します。有効な値は IN、in、OUT、または out です。デフォルト値は OUT です。

PreFilterMode はオプションです。PreFilterMode が指定されていない場合、どの PreFilter ステートメントにも一致しないイベントのみがエージェントに送信されます。

注 : PreFilterMode=IN を設定した場合、PreFilter ステートメントも定義する必要があります。

SpaceReplacement

Windows イベント・ログ (Windows Server 2008 のみ) ではデフォルトで TRUE に設定します。ただし、前のバージョンのイベント・ログではそのように設定しないでください。SpaceReplacement が TRUE になっている場合、イベント・ログ・メッセージのセキュリティー ID、サブソース、レベル、およびキーワード・フィールドに含まれるスペースは、下線 (_) で置き換えられます。
SpaceReplacement が FALSE になっている場合、イベント・ログ・メッセージのセキュリティー ID、

サブソース、レベル、およびキーワード・フィールド内のスペースは変わりません。このオプションについて詳しくは、1021 ページの『Windows イベント・ログ』を参照してください。

UseNewEventLogAPI
Windows システムで y に設定すると、イベント・ログに新規の Windows イベント・ログ・インターフェースが使用されます。このオプションは Windows 2008 以降でのみサポートされます。 Windows 2008 およびそこで実行されるアプリケーションで初めて使用されるようになった新規イベント・ログの多くにアクセスするには、このオプションが必要です。 このオプションは、前のバージョンの Windows、および UNIX と Linux では無視されます。このオプションについて詳しくは、1021 ページの『Windows イベント・ログ』を参照してください。

WINEVENTLOGS
どの Windows イベント・ログをモニターするのかを制御します。

WINEVENTLOGS ステートメントは、スペースを含まないコンマ区切りリストです。 詳しくは、1021 ページの『Windows イベント・ログ』を参照してください。

注 : Windows イベント内の復帰、タブ、または改行は、スペースで置き換えられます。

AIX システムのみでサポートされるオプション

AIX

AIXErrptCmd
エージェントが実行する **errpt** (エラー・レポート) コマンド・ストリングをここに指定することができます。コマンド出力は、モニター対象のログ・データのストリームにフィードされます。

例えば、次のコマンドを使用すると、エージェントは始動時に *mmddhhmmyy* ストリングを検索し、このストリングを実際の日時に置き換えます。 ストリングの 1 番目のオカレンスのみが置換されます。

```
AIXErrptCmd=errpt -c -smmddhhmmyy
```

独自の `errpt` コマンドを指定できますが、コマンドを連続的に実行するために `-c` (並行モード) オプションを使用する必要があります。`-t` オプション、または詳細出力を生成する `-a`、`-A`、または `-g` オプションを使用することはできません。

データ・ストリームは `errpt` コマンドからの標準出力であるため、突き合わせるためには `.fmt` ファイル内で正規表現を作成する必要があります。 例えば、データ出力は以下のようになることがあります。

```
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
F7FA22C9   0723182911 I O SYSJ2      UNABLE TO ALLOCATE SPACE IN FILE SYSTEM
2B4F5CAB   1006152710 U U ffdc       UNDETERMINED ERROR
2B4F5CAB   1006152610 U U ffdc       UNDETERMINED ERROR
```

ヘッダー以外のデータ行を選出するフォーマットの例を以下に示します。

```
REGEX GenericErrpt
^([A-F0-9]{8}) +([0-9]{10}) ([A-Z]) ([A-Z]) (¥S+) +(.*)$
Identifier $1 CustomSlot1
Timestamp  $2 CustomSlot2
T          $3 CustomSlot3
C          $4 CustomSlot4
Resource   $5 CustomSlot5
msg        $6
END
```

詳しくは、「IBM Agent Builder ユーザーズ・ガイド」の『*AIX* バイナリー・ログのモニター』を参照してください。

イベントを EIF に転送している場合のみに適用されるオプション

重要 : これらのオプションは、Operations Analytics - Log Analysis や OMNIbus などの汎用 EIF 受信側に直接送信される EIF イベントに適用されます。これらのオプションは、Cloud APM サーバーで使用することは意図されていません。

BufferEvents
イベント・バッファリングを使用可能にする方法を指定します。指定できる値は以下のとおりです。

- **YES** - BufEvtPath オプションで指定されたファイルにイベントを保管します (この値がデフォルトです)。
- **MEMORY_ONLY** - イベントをメモリー内のバッファーに入れます。
- **NO** - イベントを保管したりまたはバッファーに入れたりしません。

BufEvtPath
エージェント・キャッシュ・ファイルの絶対パス名を指定します。このパスを修正しなかった場合のデフォルトは、以下のとおりです。

- AIX `/etc/Tivoli/tec/cache`
- Windows `¥etc¥Tivoli¥tec¥cache`

注 : イベントが複数のサーバーに転送される場合、転送チャネルごとに *BufEvtPath* 値を指定する必要があります。追加項目ごとに索引番号が *BufEvtPath* 名に付加されます。例えば、最初の追加サーバーに転送するエージェント・キャッシュ・ファイルのパス名を指定するには、*BufEvtPath1* を使用します。それぞれの *BufEvtPath* で設定する値は固有でなければなりません。

BufEvtMaxSize
エージェント・キャッシュ・ファイルの最大サイズを KB 単位で指定します。デフォルト値は 64 です。*BufferEvents* オプションが Yes に設定されている場合、キャッシュ・ファイルはイベントをディスクに保管します。このファイルの最小サイズは 8 KB です。このレベルより低く指定されたファイル・サイズは無視され、8 KB が使用されます。最大ファイル・サイズに指定する値に上限はありません。

注 : キャッシュ・ファイルが存在する場合、オプション変更を有効にするには、そのファイルを削除する必要があります。

NO_UTF8_CONVERSION
Event Integration Facility がイベント・データを UTF-8 でエンコードするかどうかを指定します。 このオプションが YES に設定されている場合、EIF はイベント・データを UTF-8 でエンコードしません。このデータは EIF に渡されたときに、既に UTF-8 でエンコードされていると想定されます。ただし、データが UTF-8 でエンコードされていることを示すためのフラグに接頭部が追加されます (このフラグがイベント・データの先頭に存在しない場合)。デフォルト値は NO です。

MaxEventQueueDepth
この値は、転送のためのキューに入れることができるイベントの最大数を示します。 この限界に達すると、新しいイベントがキューに入るごとに最も古いイベントがキューから削除されます。指定しない場合、デフォルト値は 1000 です。この設定は、*NumAdditionalServers* が使用されている場合にはすべての転送チャネルに適用されます。

NumAdditionalServers
複数の Netcool/OMNIbus ObjectServer にイベントを転送する場合、この項目は必須です。 この値は、イベントの転送先サーバーの数を示すために使用されます。 有効な値は 1 から 8 です。

ServerLocation
イベント・サーバーのインストール先ホストの名前を指定します。 ホスト名または IP アドレスを指定します。IP アドレスの場合、ドット付きフォーマットを使用します。必要に応じて、サーバー・ロケーションのフェイルオーバー値 (`ServerLocation1=2.3.4.5,2.3.4.6` など) を指定できます。*ServerLocation* のフェイルオーバー値を指定する場合は、*ServerLocation* ごとに追加の *ServerPort* 値も指定する必要があります。

注 : イベントが複数のサーバーに転送される場合、サーバーごとに *ServerLocation* 値を指定する必要があります。 追加項目ごとに索引番号が *ServerLocation* 名に付加されます。例えば、最初の追加サーバーのインストール先ホストの名前を指定するには、*ServerLocation1* を使用します。

ServerPort
EIF 受信側がイベントを listen するポート番号を指定します。 *ServerPort* オプションには、コンマで区切られた 8 つまでの値を含めることができます。*ServerLocation* のフェイルオーバー値が指定されている場合、それに相当する *ServerPort* 値を設定する必要があります。ServerPort は、*TransportList* オプションが指定されている場合には使用されません。

注 : イベントが複数のサーバーに転送される場合、サーバーごとに *ServerPort* 値を指定する必要があります。 追加項目ごとに索引番号が *ServerPort* 名に付加されます。例えば、EIF 受信側が最初の追加サーバーに関するイベントを listen するポート番号を指定するには、*ServerPort1* を使用します。

TransportList
コンマで区切られた、トランスポート・メカニズムのユーザー定義名を指定します。あるトランスポート・メカニズムが送信側アプリケーションで失敗した場合、API は、リストで指定された順に以下のトランスポート・メカニズムを使用します。 受信アプリケーションの場合、API がすべてのトランスポート・メカニズムを作成して使用します。 各 *type_name* のトランスポート・タイプとチャネルは、Type キーワードと Channels キーワードを使用して指定する必要があります。

***type_name*Type**

TransportList オプションで指定されたトランスポート・メカニズムのトランスポート・タイプを指定します。サポートされるトランスポート・タイプは SOCKET のみです。

各 channel_name のサーバーとポートは、*ServerLocation* オプションと *ServerPort* オプションで指定する必要があります。

***type_name*Channels**

***channel_name*Port**
トランスポート・メカニズム・サーバーが (*Channel* オプションで設定された) 指定のチャネルを listen するポート番号を指定します。このキーワードをゼロに設定した場合、ポートマッパーが使用されます。 このキーワードは必須です。

***channel_name*PortMapper**
指定されたチャネルについてポートマッパーを使用可能にします。

***channel_name*PortMapperName**
ポートマッパーが使用可能になっている場合、そのポートマッパーの名前を指定します。

***channel_name*PortMapperNumber**
リモート・プロシージャー・コールによって登録された ID を指定します。

***channel_name*PortMapperVersion**
ポートマッパーが使用可能になっている場合、そのポートマッパーのバージョンを指定します。

***channel_name*ServerLocation**
指定されたチャネルに関するトランスポート・メカニズム用のサーバーが置かれている領域およびイベント・サーバーの名前を指定します。このチャネルは *Channel* オプションで設定する必要があります。このキーワードは必須です。

構成ファイルは、OMNIbus で直接使用される一般 EIF オプションを受け入れます。 これらのオプションは、OMNIbus への EIF 接続を介してのみ作動します。 Cloud APM サーバー に送信されるイベントには影響を与えません。これらの EIF オプションについて詳しくは、EIF キーワードを参照してください。

フォーマット・ファイル

OS エージェントは、システム・ログ・メッセージから情報を抽出し、さまざまなログ・メッセージをイベント・クラスと突き合わせます。フォーマット・ファイルは、ログ・メッセージをイベント・クラスと突き合わせるためのルックアップ・ファイルとして機能します。これにより、読み取る内容、突き合わせる内容、およびデータをフォーマット設定する方法をイベント・クラスに指示します。

フォーマット・ファイルがルックアップ・ファイルとして使用されると、ファイル内のすべてのフォーマット仕様は、ファイルの先頭から末尾への順序で比較されます。2 つのクラスが一致する場合、または 1 つのメッセージに複数の一致クラスがある場合は、末尾から開始して最初に一致する式が使用されます。一致が見つからない場合、イベントは廃棄されます。不一致ログが `.conf` ファイル内で定義されている場合、廃棄されたイベントは不一致ログに書き込まれます。

ログ・メッセージとイベントを突き合わせるパターンを作成するために使用する正規表現構文について説明します。検査対象の属性値が指定のパターンに一致するかどうかを調べるために、International Components for Unicode (ICU) ライブラリーを使用した正規表現フィルタリングのサポートが提供されます。

正規表現の使用について詳しくは、「*ICU User Guide*」の『Regular Expressions』を参照してください。

フォーマット・ファイル仕様

フォーマット・ファイルは、モニター対象ログ内のイベントを突き合わせるためにエージェントが検索するパターンを記述します。フォーマット・ファイルは、1 つ以上のフォーマット仕様から成り立っています。

エージェント・インスタンスの実行中にフォーマット・ファイルを変更できます。ファイルは、エージェントの開始時にエージェントによって読み取られ、その後 60 秒ごとに、そのタイム・スタンプの変更がモニターされます。ファイルのタイム・スタンプが変更された場合、エージェントは、その構成を動的に再初期化します。エージェントの再始動は不要です。詳しくは、653 ページの『エージェントの構成ファイルとフォーマット・ファイルの変更』を参照してください。

イベントを突き合わせる新規パターンを作成するには、以下の部分から成り立っている新規の正規表現構文を使用します。

- フォーマット・ヘッダー
- 正規表現
- スロット・マッピング
- 終了ステートメント

フォーマット・ヘッダーには、**REGEX** キーワードが含まれます。これは、エージェントに対して、モニター対象ログ内のパターンを突き合わせるために正規表現を使用していることを通知します。

以下の例に示すように、この正規表現をイベント・クラスに割り当てます。

```
REGEX REExample
```

特別な事前定義イベント・クラス *DISCARD* をイベント・クラスとして使用する場合、関連付けられたパターンと一致するログ・レコードはすべて廃棄され、それらに対するイベントは生成されません。例えば、次のようにします。

```
REGEX *DISCARD*
```

パターンと一致した場合、不一致ログには何も書き込まれません。これらの廃棄されたイベントは、一致するログ・ファイル状況レコードに含まれます。

注 : 複数のイベント定義を同じイベント・クラスまたは複数の異なるイベント・クラスに割り当てることができます。クラス名は任意であり、これを使用して、イベントのタイプを示したり、さまざまな方法でイベントをグループ化したりできます。

フォーマット・ヘッダーの後のフォーマット・コンテンツは、最初の行の正規表現と、それに続くマッピングから成り立っています。各マッピングはそれぞれ別個の行に表示されます。これらのマッピングについては、以下の例で説明します。

正規表現と一致するすべての行が選択され、モニター・サーバーにイベントとして送信されます。正規表現には副次式が含まれます。副次式を使用すると、同じであるそれらの行の特定の部分を、Event Integration Facility 内の *slot* という変数と突き合わせることができます。

以下に示すモニター・ログには、モニター対象にできる 3 つの行が含まれます。

```
Error:  disk failure
Error: out of memory
WARNING: incorrect login
```

例えば、`Error` で開始される行などの特定のエラーに対してイベントを生成し、`Warning` で開始される行を無視します。正規表現は、`Error` で開始される行と一致し、副次式を含む必要があります。副次式は括弧で示され、*msg* スロットに割り当てる入力テキストのみと一致する必要があります。1 つの副次式のみを含む単純な正規表現を以下のフォーマット定義に示します。

```
REGEX REExample
Error: (.*)
msg $1
END
```

このフォーマット仕様およびログ・データの以前のセットに基づいて、エージェントは 2 つのイベントを生成します。どちらのイベントにも REEXample イベント・クラスが割り当てられます。最初のイベントでは、disk failure の値が *msg* スロットに割り当てられます。また、2 番目のイベントでは、out of memory の値が *msg* スロットに割り当てられます。Warning 行は正規表現と一致しなかったため、この行は無視され、イベントは生成されません。

$1 の値を *msg* スロットに割り当てると、最初の副次式の値が割り当てられます。

ログ・テキストに以下のエラーが存在する場合は、即座にディスク障害の通知を受け取れるように、これらのエラー・メッセージを専用のイベント・クラスに割り当てることができます。

```
Error: disk failure on device /dev/sd0: bad sector
Error: disk failure on device /dev/sd1: temperature out of range
```

エラーが発生したディスクの説明およびディスク・エラーのより具体的な説明をイベント内に組み込むことができます。

以下の正規表現には、この情報を識別する 2 つの副次式が含まれます。

```
REGEX DiskFailure
Error: disk failure on device (/dev/sd[0-9]):(.*)
device $1 CustomSlot1
msg $2
END
```

これらの 2 つの副次式をイベント・スロットに割り当てます。生成される 2 つのイベントには、以下の値が含まれます。

```
"device=/dev/sd0" and "msg=bad sector"
"device=/dev/sd1" and "msg=temperature out of range"
```

EIF を使用して最初のイベントを生成する場合は、以下の例に示すように表示されます。

```
DiskError;device='/dev/sd0';msg='bad sector';END
```

イベントが Cloud APM サーバーに送信される場合は、*msg* という名前のスロットが、同じ名前の Performance Management エージェント属性に割り当てられます。ただし、*device* スロットに事前定義属性はありません。

device に割り当てられている値を Cloud APM コンソールで直接表示する必要がある場合、またはそれに対するしきい値を書き込む必要がある場合は、それを Performance Management 属性に割り当てる必要があります。

OS エージェントには、以下の 13 個の事前定義属性が含まれています。

- *CustomSlot1* から *CustomSlot10* までの範囲の 10 個のストリング・タイプ属性
- *CustomInteger1* から *CustomInteger3* までの範囲の 3 個の整数タイプ属性

フォーマット・ファイル内でこれらの属性名を使用して、同じ名前の Performance Management 属性のデータ設定を行います。これらの属性を使用しても、OMNIbus に直接送信される EIF イベントのコンテンツには影響しません。

注 : CustomSlot 属性名および CustomInteger 属性名では大/小文字が区別されるため、示された名前を正確に入力する必要があります。

フォーマット・ファイル内でスロットをイベント定義からこれらのカスタム Performance Management 属性の 1 つに割り当てます。

以下の例に示すように、*device* スロットを、*CustomSlot1* という名前の Performance Management ストリング・タイプ属性に割り当てます。

```
REGEX DiskFailure
Error: disk failure on device (/dev/sd[0-9]):(.*)
device $1 CustomSlot1
msg $2
END
```

イベントが Application Performance ダッシュボードに表示されると、*device* スロットに割り当てられた値が、Performance Management の CustomSlot1 属性に割り当てられます。この値を Cloud APM コンソールで確認するか、この値を使用してしきい値を定義します。イベント定義内の任意のスロットを、10 個のカスタム・エージェント属性のいずれかに同じ方法で割り当てることができます。割り当てるには CustomSlot*n* を使用します。ここで *n* は、スロット定義に続く 1 から 10 までの番号です。

この例では、最初の副次式は (/dev/sd[0-9]) として具体的に定義されますが、2 番目の副次式は (.*) として一般的に定義されます。正規表現をできるだけ具体的に定義しておくと、パフォーマンスが向上します。したがって、ここで定義された特定のエラー・メッセージと一致しないデバイス上のエラーに対する検索が入力された場合、検索手順は、エラーが見つからないと即時に停止します。一致を検索するために時間を浪費することはありません。

END キーワードは、フォーマット仕様を完了します。以下の例に示すように、フォーマット・ヘッダー、正規表現、および *END* キーワードは、それぞれ改行後に開始される必要があります。

```
REGEX REExample
Error:
msg $1
END <EOL>
<EOF>
```

注 : ファイル内の最後のフォーマットでは、例に示すように、END キーワードの後に改行を挿入する必要があります。 これに違反すると、構文解析エラーが発生します。

CustomInteger1 から *CustomInteger3* までは 64 ビットのカスタム整数属性です。これらは、ストリング・タイプ CustomSlot 属性と同じ方法で使用できます。これらの属性を使用して、個別のスロットまたは副次式を、ログ・ファイルから個別の Cloud APM 属性にマップできます。これらの属性は数値であるため、属性に対して < および > などの算術比較を使用できます。ストリング属性に対してはこのような比較を使用できません。

注 : これらの値は、Cloud APM サーバーによって整数として評価されますが、EIF の用途およびフォーマット・ファイル内ではストリングとして扱われます。例えば、整数スロットを PRINTF ステートメント内で使用するには、「%d」ではなく「%s」を使用してスロットを識別します。

以下の例は、カスタム整数属性の使用を示しています。定期的な UNIX syslog メッセージを受信し、以下の架空のログ・レコードのように、ファイル・システムの空きスペースの割合が報告されるとします。

```
Oct 24 11:05:10 jimmy fschecker[2165]: Filesystem /usr is 97% full.
```

フォーマット・ファイル内で以下のステートメントを使用して、ファイル・システムの空きスペースの割合を検査できます。

```
REGEX FileSystemUsage
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2}) (.*?) (.*?):
Filesystem (.*?) is ([0-9]+)% full¥.$
Month   $1 CustomSlot1
Date    $2 CustomSlot2
Time    $3 CustomSlot3
Host    $4 CustomSlot4
Service $5 CustomSlot5
Filesystem     $6 CustomSlot6
PctFull         $7  CustomInteger1
msg          PRINTF("%s: %s% full", Filesystem, PctFull)
END
```

注 : 上記のステートメントでは、2 行目の ^ シンボルと 3 行目の $ シンボルの間のストリング全体を単一行で指定する必要があります。

他のイベントが *CustomInteger1* 内に値を挿入する可能性があるため、異なるイベント・タイプの混同を避けるために、*Class* 属性の値を使用して、影響の範囲を適切なイベント・タイプのみに制限できます。例えば、以下のしきい値式を使用すると、*FileSystemUsage* イベント・クラスのイベントの値が *CustomInteger1* 内で 95 以上である場合にのみ、しきい値が起動されます。

```
(  Class == 'FileSystemUsage' AND CustomInteger1 >= 95)
```

それ以外のイベントが *CustomInteger1* を別の目的で使用しても、このしきい値を誤ってトリガーすることはありません。

これで、Performance Management 内で CustomInteger 属性に対して算術演算子を使用するしきい値を記述できます。CustomSlots 属性に対してはこのような演算子を使用できません。

注 : 非整数データを CustomInteger 属性にマップした場合、結果の値は、ゼロまたは何らかの予期しない値になることがあります。

エージェントの構成ファイルとフォーマット・ファイルの変更

OS エージェントは、開始時にその構成ファイル (.conf) とフォーマット・ファイル (.fmt) を読み取り、その後 60 秒ごとにタイム・スタンプをモニターします。

構成ファイルまたはフォーマット・ファイルのタイム・スタンプが変更された場合、エージェントは、再始動することなくその構成を動的に再初期化します。 再初期化中に、モニタリングが一時的に中断します。モニタリングの再開時に、エージェントは、モニター・ログ内の再開位置を決定する必要があります。 これにより、エージェントは完全な停止後の再始動の場合と同様に動作します。

注 : 構成ファイルまたはフォーマット・ファイルが変更された後のエージェント再初期化によって、ログ・ファイル正規表現統計、ログ・ファイル状況、およびログ・ファイル・イベントの各属性グループ内の情報がリセットされます。

デフォルトでは、再初期化が完了すると、エージェントはファイルの末尾からモニタリングを開始します。この開始位置によって、モニタリングの中断中に発生したイベントが欠落する可能性があります。 モニタリングの再開時にこのようなイベントを確実に収集するには、NumEventsToCatchUp=-1 設定を使用してください。

NumEventsToCatchUp=-1 を設定すると、位置ファイルが維持されます。 位置ファイルは、エージェントがログ・ファイルを読み取るたびに更新されます。更新では、エージェントの再始動に備えてログ・ファイル内にエージェントの位置が保存されます。 位置ファイルの維持はパフォーマンスに若干の影響を与えますので、このファイルは必要な場合にのみ維持してください。NumEventsToCatchUp について詳しくは、640 ページの『構成ファイル』を参照してください。

注 : 一部の構成値は構成ファイル内に存在しないため、初期構成時に設定されません。 これらの値を変更した場合には、エージェントを再始動する必要があります。

継承

フォーマット・ファイルは、継承を使用して、以前に定義されたフォーマット仕様からスロット定義を派生させます。

FOLLOWS 関係を指定すると、継承を使用して、汎用フォーマット仕様から特定のフォーマット仕様を作成できます。

最初に、以下の例に示すように、基本クラスを定義し、それに DiskFailure という名前を付けます。

```
REGEX DiskFailure
Disk Failure on device (.*)
device $1 CustomSlot1
END
```

この正規表現はモニター・ログ内の Disk Failure on device/dev/sd0 エラーと一致するため、/dev/sd0 値が *device* スロットに割り当てられます。

ただし、モニター・ログ内で報告されたこのエラー・メッセージの拡張バージョンを表示することもできます。

例えば、Disk Failure on device /dev/sd0, error code: 13 エラー・メッセージがあるとします。

このエラー・メッセージは、以下の例に示すように、スロットと突き合わされます。

```
REGEX DiskFailureError FOLLOWS DiskFailure
Disk Failure on device (.*), error code: ([0-9]*)
errcode $2 CustomSlot2
END
```

この場合、イベントには、*device* スロットおよび *errcode* スロットが含まれます。DiskFailure イベント・クラスではデバイス名のスロットが既に定義されているため、サブクラスにそのスロットを継承させると、この継承により 2 回目の宣言を省略できます。スロットは $1 として定義されるため、正規表現内の最初の副次式がそのスロットに割り当てられます。

しかし、DiskFailureError クラスでは、2 番目の副次式も定義されます。この副次式を errcode という名前の新規スロットに割り当て、それを $2 として定義して、正規表現内の 2 番目の副次式を参照することができます。このタイプの割り当ては、ログ・テキストを表示する上記の例に示されています。

このイベントでは、device スロットに /dev/sd0 の値が割り当てられ、errcode スロットに 13 の値が割り当てられます。CustomSlot1 にデバイスが割り当てられ、CustomSlot2 にエラー・コードが割り当てられます。

Performance Management カスタム属性マッピングも継承されます。Performance Management カスタム属性マッピングについて詳しくは、650 ページの『フォーマット・ファイル仕様』を参照してください。

複数行
複数行構文を使用して、複数の行にまたがるレコードを、モニターしているログ内のパターンと突き合わせます。

正規表現の一部として ¥n 改行文字を指定して、モニター・ログ内で改行が発生する位置を示します。以下の例にあるこのタイプの構文を参照してください。

```
REGEX REMultiLine
Line1:(.*)¥nLine2(.*)
msg $1
second_msg $2
END
```

注 : Windows ¥r¥n 復帰および改行の組み合わせを指定してください。

以下のエラー・メッセージがログ・テキスト内で報告されている場合は、REMultiLine イベントが作成されます。

```
Line1: An error occurred
Line2: The error was "disk error"
```

msg スロットには An error occurred の値が割り当てられ、second_msg スロットには The error was "disk error" の値が割り当てられます。

マッピング
OS エージェントは、マッピングを使用して、システム・ログ・メッセージのイベント・クラスを判別します。エージェントは、メッセージをフォーマット・ファイル内のパターンと突き合わせて、イベント・クラスを判別します。

エージェントは、ログ・メッセージを、属性 name=value ペアを含むイベント・クラス・インスタンスに変換します。その後、イベントはイベント・サーバーに送信されます。

エージェントは、ソースでのシステム・ログ・メッセージのイベント・クラスを判別します。エージェントは、システム・ログ・メッセージをフォーマット・ファイル内のパターンと突き合わせて、イベント・クラスを判別します。この突き合わせ手順を使用してクラスを判別した後で、値を属性に割り当てる必要があります。

属性値は、以下のようなさまざまなソースから取得されます。

- エージェントによって提供されるデフォルト値
- 正規表現内の特定の副次式と一致するログ・テキスト

フォーマット・ファイル内に含まれているマップ・ステートメントの構文は以下のとおりです。

```
name    value CustomSlotn
```

ここでは、任意の ID を指定して、スロット (変数、属性、または値 ID とも呼ばれます) の name を定義します。次に、655 ページの『値指定子』で説明されている任意の値を適用して、このスロットに割り当てる値を指定します。

カスタム・スロットを使用して、データを Performance Management コンソール内に表示し、しきい値を定義します。しきい値を作成する場合、すべてのカスタム・スロット値はストリングです。重複の判別に使用するスロットを識別する必要があるため、重複検出のためのカスタム・スロットも必要です。イベントのフィルタリングについて詳しくは、1018 ページの『イベント・フィルタリングおよび要約』を参照してください。msg は、イベント・テーブル内で専用の属性を持つ特別なスロット名です。msg では、カスタム・スロットを使用する必要はありません。

スロットがフォーマット定義内にのみ存在するように、スロットの有効範囲を制限できます。スロットを定義するときに、スロット名の前にダッシュを付けます。以下に例を示します。

```
-name    value
```

この方法で定義したスロットは、最終イベント内に含まれません。しかし、フォーマット定義内の他の場所ではこのスロットを参照できます。具体的には、PRINTF ステートメント内で参照します。以下の REGenericSyslog の例では、service スロットを生成しても組み込まれませんが、このスロットは PRINTF ステートメントで参照できます。元のスロットがダッシュなしで定義された場合にスロットに適用されたのと同じ値が保持されます。この手順を使用して、最終イベント内に含まれない一時変数をフォーマット定義内で使用できます。例えば、以下の方法でイベント・クラス REGenericSyslog を定義して、汎用 UNIX syslog イベントを突き合わせることができます。

```
REGEX REGenericSyslog
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2}) (.*?) (.*?): (.*)$
month $1
date $2
time $3
host $4
-service $5
msg $6
syslog_msg PRINTF("service %s reports %s", service, msg)
END
```

値指定子
フォーマット仕様でのマッピングによって値を属性に割り当てます。

フォーマット仕様のマッピング部分は、以下のタイプの値指定子から成り立っています。

- $i
- ストリング定数
- PRINTF ステートメント

$i

i は、書式制御ストリング内の副次式の位置を示します。各副次式には、1 から書式制御ストリング内の副次式の最大数までの番号が付きます。

$i 値指定子 (変数、スロット、または属性とも呼ばれます) の値は、システム・ログ・メッセージにおいて、対応する副次式と一致する部分を示します。

以下の例では、ログ・エージェントは、UNIX syslog 機能からのすべてのログ・メッセージを、割り当てられた値を持つ syslog イベントに変換します。

```
REGEX REGenericSyslog
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2})
 (.*?) (.*?): (.*)$
month   $1
date    $2
time    $3
host    $4
service $5
msg     $6
END
```

$1 から $6 までの番号が付いた各副次式は、正規表現内の括弧で囲まれた 1 つの項目と一致します。

したがって、例えば、以下の syslog イベントがあるとします。

```
Apr  6 10:03:20 jimmy syslogd 1.4.1: restart.
```

このイベントには、以下の値が割り当てられます。

```
month=Apr
date=6
time=10:03:20
host=jimmy
service=syslogd 1.4.1
msg=restart.
```

例えば、syslog イベント内で、`10:03:20` 値は、正規表現内の括弧で囲まれた 3 番目の項目と一致します。したがって、値は、*$3* の時刻値に割り当てられます。同様に、`jimmy` 値は、正規表現内の括弧で囲まれた 4 番目の項目と一致します。したがって、値は、*$4* のホスト値に割り当てられます。

ストリング定数

ストリング定数は、指定されたストリングが属性の値であることを宣言します。属性値がスペースを含まない単一の定数である場合は、以下の例に示すように、二重引用符 (" ") で囲まないで指定します。

```
severity WARNING
```

反対に、属性値がスペースを含む場合は、以下の例に示すように、二重引用符を使用する必要があります。

```
component "Web Server"
```

PRINTF ステートメント

PRINTF ステートメントは、他の属性値を組み合わせた複合属性値を作成します。PRINTF ステートメントは、キーワード PRINTF およびそれに続く printf() C スタイル書式制御ストリングと 1 つ以上の属性名から成り立っています。

書式制御ストリングは、%s コンポーネント指定子のみをサポートします。PRINTF ステートメント内で使用される属性の値は、*$i* 値指定または定数ストリング値指定から派生させる必要があります (別の PRINTF ステートメントから派生させることはできません)。

引数属性の値を使用して、書式制御ストリングに従って新規定数ストリングを作成します。この新規定数ストリングが属性の値になります。

PRINTF キーワードを使用すると、REGenericSyslog 基本クラスおよび *service* と *msg* のスロットを定義した上記の例に基づいて、*syslog_msg* という名前の属性を定義できます。

```
syslog_msg PRINTF("service %s reports %s", service, msg)
```

例えば、以下のログ・メッセージが報告されたとします。

```
Apr  6 10:03:20 jimmy syslogd 1.4.1: restart.
```

書式制御ストリングからの属性値を含む新規定数ストリングが作成されます。

```
syslog_msg="service syslogd 1.4.1 reports restart."
```

キーワード
フォーマット・ファイルでは、キーワードを使用して、実行時に拡張する値を割り当てます。

以下のキーワードは、実行時に拡張されます。

- DEFAULT
- FILENAME

- LABEL
- REGEX

DEFAULT

DEFAULT キーワードを使用して、DEFAULT 値を特定のスロットまたは属性に割り当てます。OS エージェントは、内部デフォルト値を、以下の表で説明するスロットに割り当てます。

表 *191.* スロットおよび *DEFAULT* 値

スロット	**説明**
hostname	*hostname* は、エージェントが実行されているシステムの短いホスト名です。システムのドメイン名は含まれません。
origin	*origin* は、エージェントが実行されているシステムの IP アドレスです。
fqhostname	*fqhostname* は、エージェントが実行されているシステムの完全修飾ホスト名です。システムのドメイン名が含まれます。
RemoteHost	イベントがローカル・システムで発生した場合、この属性は空になります。イベントがリモート・システムで発生した場合、*RemoteHost* には *user@host:port* という形式のストリングが設定されます。このストリングは、イベントが発生したリモート・ホストの名前と、そのホスト上で接続に使用されるユーザーとポートを示しています。

fqhostname に割り当てられる値は、`.conf` ファイル内の以下の FQDomain (オプション) 設定によって影響されます。

- FQDomain を yes に設定すると、エージェントがシステム・ドメイン名自体を決定します。
- FQDomain の値を設定しなかった場合、または値を no に設定した場合、エージェントはドメイン名を設定せず、*fqhostname* 属性には空ストリングが割り当てられます。
- FQDomain を「yes」または「no」以外の値に設定した場合、ドメイン名は値として受け入れられ、ホスト名の後に付加されます。

以下の例では、フォーマット定義には 3 つの属性またはスロットが含まれます。

- *hostname* DEFAULT
- *origin* DEFAULT
- *fqhostname* DEFAULT

例えば、`.conf` ファイル内で FQDomain を「yes」に設定し、以下のプロパティーを持つコンピューター上で実行するとします。

- *hostname*: `myhost`
- *IP address*: `192.168.1.100`
- *domainname*: `mycompany.com`

イベントが作成され、3 つのスロットに以下の値が割り当てられます。

```
"hostname=myhost", "origin=192.168.1.100", "fqhostname=myhost.mycompany.com"
```

FILENAME

FILENAME キーワードは、メッセージを格納するログ・ファイルの完全修飾ファイル名 (パスを含む) を示します。単一のエージェントを使用して複数のログ・ファイルをモニターし、イベントのソースを識

別する必要がある場合は、このキーワードを使用して、イベント属性にファイル名を取り込みます。メッセージがシステム・ログから取得される場合、マッピングは、EventLog (Windows OS エージェントの場合) および SysLogD (UNIX OS エージェントの場合) に設定されます。

注 : パスには、このキーワードの属性が含まれます。

LABEL
LABEL キーワードは、エージェントが実行されているシステムのホスト名を示します。

REGEX
REGEX キーワードは、メッセージと一致し、イベントの原因となった正規表現に拡張されます。

最大メッセージ長
この値は、OS エージェントがメッセージを切り捨てないで受信できる最大メッセージ長です。

最大メッセージ長は、Performance Management の場合と Tivoli Netcool/OMNIbus の場合で異なります。

Performance Management
Performance Management に送信されるイベントの場合、msg 属性は 2048 バイトに制限されます。これより長いメッセージは切り捨てられます。

Tivoli Netcool/OMNIbus
Probe for Tivoli EIF を通じて Netcool/OMNIbus に送信されるイベントの場合、クラス名およびすべてのスロットとその値を含むイベントの合計サイズは、4096 バイトを超えることはできません。例えば、以下のサンプル EIF イベントでは、;END は、4096 バイト制限の一部としてカウントされません。しかし、セミコロン、引用符、等号などの構文要素を含むその他のすべてのバイトは、制限の一部としてカウントされます。

```
Class;attr1=value1;attr2=value2;msg='Hello, world';END
```

リモート・ログ・ファイル・モニター: パスワードまたはパスフレーズの暗号化

セキュリティーを高めるために、リモート・ログ・ファイル・モニターの使用時に、リモート・システムに送信されるパスワードとパスフレーズを暗号化することができます。

このタスクについて

暗号化されたパスワードとパスフレーズは、構成ファイル (.conf) に保管されます。構成ファイルの詳細については、640 ページの『構成ファイル』を参照してください。

手順

- 以下のように、**itmpwdsnmp** コマンドを実行し、暗号化するパスワードまたはパスフレーズを指定します。
  - Linux AIX Cloud APM のインストール・ディレクトリーからコマンドを実行します。デフォルトのインストール・パスは opt/ibm/apm/agent であり、*install_dir* はエージェントをインストールした場所です。
  - Windows デフォルトのインストール・パスは、C:¥IBM¥APM です。

  Linux Linux システムで実行されるコマンドの例を以下に示します。

```
$ export install_dir=/opt/ibm/apm/agent/bin
$ /opt/ibm/apm/agent/bin

Enter string to be encrypted:
mypassword

Confirm string:
mypassword

{AES256:keyfile:a}Z7BS23aupYqwlXb1Gh+weg==
$
```

この例では、{AES256:keyfile:a}Z7BS23aupYqwlXb1Gh+weg== コマンドからの出力全体を使用してエージェント構成ファイルの **SshPassword** を設定しています。接頭部 {AES256:keyfile:a} は、エージェントに対して、パスワードが暗号化されていることを示します。

秘密鍵ファイルのパスフレーズを暗号化する場合も同じ手順に従ってください。

OS エージェントのカスタム・スクリプトの構成

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS、および Monitoring Agent for Windows OS のエージェントは自動的に構成されます。この機能により、ユーザーは、指定した頻度で OS エージェントを実行するスクリプトを定義できます。

カスタム・スクリプト機能はデフォルトで有効になっています。管理者は、エージェントの構成ファイルに新しい環境変数 *KXX_FCP_SCRIPT*=true/false (デフォルトは true) を設定して、この機能を有効または無効にできます。ここで、XX は以下の値にすることできます。

- Monitoring Agent for Linux OS の場合は LZ

- Monitoring Agent for UNIX OS の場合は UX

- Monitoring Agent for Windows OS の場合は NT

詳細については、以下のセクションで説明しています。

カスタム・スクリプトのクイック・スタート

スクリプトを指定した頻度で OS エージェントで実行するよう定義するには、OS エージェントにカスタム・スクリプトを追加します。

この機能は、OS エージェントが開始されるとすぐに、デフォルト値で有効になります。このスクリプト機能を開始するための唯一のアクションは以下のとおりです。

用意されているテンプレート `script_property.txt` を例に使用して、デフォルト・ディレクトリー (Linux™ または UNIX™ の場合は `install_dir/localconfig/`*`product code`*`/scripts_definitions`、Windows™ の場合は `install_dir¥localconfig¥nt¥scripts_definitions`) にプロパティー・ファイルを作成します。

必要なプロパティーは以下の 2 つのみです。

ATTRIBUTE_NAME
プロパティー・ファイル内のスクリプト定義を一意的に識別するのに使用される名前。

SCRIPT_PATH_WITH_PARMS
引数付きスクリプトの完全修飾パス。

シェル・スクリプトだけでなく、perl やその他のタイプのスクリプトも使用できます。SCRIPT_PATH_WITH_PARMS プロパティーには、実行する完全なコマンドを指定します。

例えば、`perl C:¥IBM¥scripts¥Custom_Scripts¥date.pl` などです。この例では、エージェントがその環境の PATH 変数で `perl` の場所を解決できるようにしてください。そうでなければ、`perl` がインストールされている場所の絶対パスを指定してください。

OS エージェント環境ファイル内のパラメーター

カスタム・スクリプトのパラメーターを OS エージェント環境ファイルに設定できます。

OS エージェント環境ファイルに以下のパラメーターを設定することによって、スクリプト機能をカスタマイズできます。

***install dir*/config/lz.environment**
Monitoring Agent for Linux OS の環境ファイル。

***install dir*/config/ux.environment**
Monitoring Agent for UNIX OS の環境ファイル。

***install dir*¥TMAITM6_x64¥KNTENV**
64 ビット Monitoring Agent for Windows OS の環境ファイル。

***install dir*¥TMAITM6¥KNTENV**
32 ビット Monitoring Agent for Windows OS の環境ファイル。

KXX_FCP_SCRIPT
スクリプト機能はデフォルトで有効になっています。無効にするには、KXX_FCP_SCRIPT=false に設定します。

その他のパラメーターについては、特定のニーズに応じて、エージェント環境ファイル内に定義できます。

KXX_FCP_SCRIPT_DEFINITIONS
プロパティー・ファイルが保管される場所。

Linux™ または UNIX™ でのデフォルトの場所は *install dir*/localconfig/*PC*/scripts_definitions、Windows™ では *install dir*¥localconfig¥nt¥scripts_definitions です。

KXX_FCP_SCRIPT_INTERVAL

OS エージェントは、実行中のスクリプトの実行状態を確認するためのループ間隔 (秒単位) としてこの変数の値を使用し、フィルター条件が満たされている場合にイベントを送信します。最小値は 30 秒で、最大値は 300 秒です。無効な値はデフォルト値にリセットされます。デフォルト値は 60 秒です。

注 : KXX_FCP_SCRIPT_SYNC_INTERVALS が USE_SCRIPT (KXX_FCP_SCRIPT_SYNC_INTERVALS の定義を参照) に設定されている場合、このパラメーターは無視されます。

KXX_FCP_SCRIPT_SYNC_INTERVALS

KXX_FCP_SCRIPT_INTERVAL に定義されているエージェントのループ間隔がスクリプトの実行頻度より大きい場合、一部のスクリプト実行ループで生成されたデータが失われることがあります。この動作を回避するには、KXX_FCP_SCRIPT_SYNC_INTERVALS を以下のように設定して、スクリプトの実行頻度をエージェントのループ間隔と同期させることができます。

- USE_AGENT - 各スクリプトの実行頻度の値を、プロパティー・ファイルで定義された KXX_FCP_SCRIPT_INTERVAL と EXECUTION_FREQUENCY のうちの最大値に強制的に設定します。
- USE_SCRIPT - エージェントのループ間隔を、定義されているすべてのスクリプトの頻度の値 (プロパティー・ファイルの EXECUTION_FREQUENCY) のうちの最小値に動的に設定します。KXX_FCP_SCRIPT_INTERVAL に設定された値は無視されます。スクリプトの頻度はプロパティー・ファイルで定義されたままとなります。USE_SCRIPT を使用すると、エージェントのループ間隔は、スクリプトの定義が追加、変更、削除されるたびに変わる可能性があります。いずれの場合でも、KXX_FCP_OVERRIDE_MIN_FREQUENCY_LIMIT に設定した値より低くなることも 300 秒より大きくなることもありません。
- NO - 同期は行われず、一部の実行結果は失われる場合があります。

KXX_FCP_SCRIPT_DEFINITIONS_CHECK_INTERVAL

OS エージェントは、開始時と、この変数に定義された間隔ごとに、スクリプトやプロパティー・ファイルに変更がないか確認します。KXX_FCP_SCRIPT_DEFINITIONS_CHECK_INTERVAL がエージェントのループ間隔より小さい場合は、エージェントのループ間隔にリセットされます。許可される最大値は、デフォルト値の 300 秒です。

KXX_FCP_USER

このパラメーターは、Linux™ または UNIX™ の OS エージェントでのみ有効です。このパラメーターは、fcp_deamon プロセスの作成に使用するユーザーが OS エージェント・プロセス・ユーザーと異なる場合に、その user を定義します。すべてのスクリプトは、このユーザーで実行されます。OS エージェントの所有者には、fcp_daemon プロセスを作成するための適切な権限が必要であることに留意してください。Windows™ では、サービス Monitoring Agent for Windows OS「FCProvider」のログインとして別のユーザーを定義する必要があります。このユーザーには、エージェントのインストール・ディレクトリーとスクリプトのリポジトリー・ディレクトリーに対する「フル コントロール」権限が必要です。詳しくは、以下を参照してください。

https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/install/install_linuxaix_agent_nonroot.html

KXX_FCP_MAX_CDP_DP_THREAD_POOL_SIZE
スクリプトの最大同時実行数を定義します。最大値は 32 です。

KXX_FCP_MAX_DAEMON_RESTARTS

OS エージェントは fcp_daemon を監視し、プロセスの異常終了が発生した場合にそれを再始動します。デフォルト値は 4 です。再始動は KXX_FCP_MAX_DAEMON_RESTARTS (1 日あたりの回数) の回数まで行われます。再始動を回避するには、値を 0 に設定する必要があります。-1 を設定すると、OS エージェントは fcp_daemon の再始動を永久的に再試行します。OS エージェントが再始動するたびに、再始動のカウンターは減少していきます。

KXX_FCP_SEND_SCRIPT_RUNTIME_EVENTS

デフォルト値は true です。false に設定した場合、OS エージェントは、スクリプトの標準出力の各行のイベントを送信することを停止します。この場合、スクリプトの出力はコンソールのワークスペースに表示されますが、シチュエーションは表示されず、履歴データも収集されません。

KXX_FCP_OVERRIDE_MIN_FREQUENCY_LIMIT

KXX_FCP_SCRIPT_SYNC_INTERVALS を USE_SCRIPT に設定する場合に使用されます。そのような状況で、OS エージェントのループ間隔の最小値を設定します。

OS エージェントのループ間隔に低い値 (5 秒以下) を使用すると、介入の頻度が高まり、OS エージェントのパフォーマンスに影響を及ぼすおそれがあります。データを頻繁 (毎秒など) に収集する必要がある場合は、スクリプトをカスタマイズすることを推奨します。スクリプトで必要な頻度でデータをキャッシュに入れ、収集したデータを長めの間隔 (60 秒ごとなど) で OS エージェントに返すようにします。

fcp_daemon の動作の制御には、以下の Agent Builder (CDP) の変数も使用できます。

CDP_DP_REFRESH_INTERVAL

(デフォルトは 60 秒) スクリプトのグローバルな開始予定時間。スクリプトのプロパティー・ファイルに頻度が渡されない場合に使用されます。

CDP_DP_SCRIPT_TIMEOUT

(デフォルトは 30 秒) スクリプトのグローバルな最大実行時間。スクリプトの実行時間がこの制限を超えると、その Status_Code が TIMEOUT に設定されます。

CDP_DP_KILL_ORPHAN_SCRIPTS

(Y または N - デフォルトは N) スクリプトがタイムアウトになった場合に fcp_daemon プロセスによって使用されるグローバルな動作。「Y」に設定するとスクリプトは終了され、それ以外の場合は中止されます。特定のスクリプトのプロパティー・ファイルで KILL_AFTER_TIMEOUT キーが設定されている場合、そのスクリプトではこの値が無視されます。

CDP_MAXIMUM_ROW_COUNT_FOR_CPCI_DATA_RESPONSES

(デフォルト値は 1000) スクリプトによって返される出力行の最大数を制限するために、パフォーマンス上の理由で追加するグローバル値。この制限を超えた後の余分な行は、無視されます。許可される値は正整数です。値が無効である場合は、制限なしに変更されます。

fcp_daemon では、Agent Builder のエージェントを制御するために使用する環境変数を他にもサポートしています。詳細なリストについては、こちらの Agent Builder の公式資料を参照してください。

http://www-01.ibm.com/support/knowledgecenter/api/redirect/tivihelp/v61r1/topic/com.ibm.itm.doc_6.3/agentbuilder63_user.pdf

プロパティー・ファイル内のパラメーター

カスタム・スクリプトのパラメーターをプロパティー・ファイルに設定できます。

KXX_FCP_SCRIPT_DEFINITIONS ディレクトリーには、*.properties ファイルのリストが含まれています。各プロパティー・ファイルには、実行するスクリプトのリストが、それぞれのプロパティー (key=value の形式) とともに含まれています。定義できるプロパティー (大/小文字を区別しない) は以下のとおりです。

ATTRIBUTE_NAME

必須 - 最大 256 字の文字列。特定のスクリプトとその属性を定義する任意の名前。ATTRIBUTE_NAME 名に使用できる文字は、英数字と、特殊文字ではアンダースコアーのみです。その他の特殊文字 (スペースやブランク) を使用すると、それらはアンダースコアーに 変換されます。同じプロパティー・ファイル内に複数のスクリプトがリストされている場合は、さらに、異なる ATTRIBUTE_NAME を定義する必要があります (スクリプトごとに 1 つ)。これは、定義済みの各スクリプトに指定される最初の値であ

る必要があり、次の ATTRIBUTE_NAME まで、特定のスクリプトのプロパティー・セットの開始を区切るものです。

SCRIPT_PATH_WITH_PARMS
必須 - 最大 512 字の文字列。このパラメーターは、スクリプトへの絶対パスとパラメーターを定義します。パラメーターはブランクで区切ります。スクリプトのパス名に特殊文字を使用することはできません。ブランクが含まれる値は、単一引用符 (') または二重引用符 (") で囲む必要があります。 環境変数を渡すことができますが、どのオペレーティング・システムでも ${...} で囲む必要があります。 環境変数は、OS エージェント・プロセスの状況に応じて使用できる必要があります。

EXECUTION_FREQUENCY
オプション - デフォルト値は 60 秒です。このパラメーターは、スクリプトの実行頻度を定義します。

CUSTOM_NAME
オプション - 最大 256 字の文字列。このパラメーターは、スクリプトの説明のために使用できます。

IS_ACTIVE
オプション - true または false。デフォルト値は true です。スクリプトをアクティブ化します。false の場合、スクリプトは実行されません。

DISABLE_USE_AGENT_SYNC
オプション - true または false。デフォルト値は false です。true の場合、グローバル変数である KXX_FCP_SCRIPT_SYNC_INTERVALS が USE_AGENT に設定されていても、スクリプトの EXECUTION_FREQUENCY が考慮されます。

KILL_AFTER_TIMEOUT
オプション - true または false。デフォルト値は CDP_DP_KILL_ORPHAN_SCRIPTS 変数によって定義されます。true の場合、スクリプトはタイムアウト後に終了します。 タイムアウトは、スクリプトの実行時間が OS エージェント構成ファイルの CDP_DP_SCRIPT_TIMEOUT パラメーターで指定された値を超えた場合に発生します。それ以外の場合は無視されます。どちらの場合も、データは収集されません。KILL_AFTER_TIMEOUT を設定すると、プロパティー・ファイルに定義されているスクリプトのみが停止され、スクリプトが作成した子プロセス (ある場合) は停止されません。この機能は、Solaris™ と Windows™ 32 ビットの OS エージェントではサポートされません。また、すべてのタイムアウト・スクリプトは放棄されます。

スクリプトによって返される出力行が解析されます。
スクリプトは標準出力 (最初のトークンと呼ばれるもの) を返します。スクリプトが出力行にさらに値を返すと、それらはさらにトークンとして追加されます。定義済みの構文に従って、最大 5 つの文字列、5 つの整数、および 5 つの浮動小数点数です。

OUTPUT_TYPE

STRING|INTEGER|FLOAT - オプション - デフォルト値は STRING です。スクリプトの各行で返される最初のトークンのタイプを定義します。OUTPUT_TYPE は以下の値にすることができます。

- STRING (デフォルト) - 最大 2048 字までの文字列。使用する場合、KXX_Custom_Scripts_Rtm_Smp の「Standard_Output_String」属性には、最初のトークンが入力されます。
- INTEGER - -9223372036854775806 から 9223372036854775806 までの間の数値を取得できます。 使用する場合、KXX_Custom_Scripts_Rtm_Smp の「Standard_Output_Integer」属性には、最初のトークンが入力されます。
- FLOAT - -92233720368547758.06 から 92233720368547758.06 の間の数値を、2 桁 の小数精度で取得できます。使用する場合、KXX_Custom_Scripts_Rtm_Smp の「Standard_Output_Float」属性には、最初のトークンが入力されます。

TOKEN_TYPES
STRING|INTEGER|FLOAT - オプション - 最初のトークンの後のさらなるトークンの出力タイプを定義します。 ユーザーは、最大 5 つの文字列、5 つの整数、および 5 つの浮動小数点を定義できます。これは、コンマで区切ったタイプのリストで、<token_type>,<token_type>,... のようにします。token_type は空か、以下のいずれか (大/小文字を区別しない) にできます。

- - STRING または S
- - INTEGER または I

- - FLOAT または F
- *TOKEN_TYPES* が空の場合、対応するトークンはスキップされます。

  効果が同じであるレイアウトの例を以下に示します。
- - TOKEN_TYPES=S,I,S,,,F,,F,F
- - TOKEN_TYPES=String,integer,S,,,Float,,f,FLOAT

TOKEN_LABELS
STRING - オプション - ラベルごとに最大 16 字。TOKEN_TYPES で定義されているトークンのラベルを定義します。この値は、コンマ区切りのトークン・ラベルのリストであり、TOKEN_TYPES で定義されたトークンに対応する必要があります。例:

- TOKEN_TYPES=S,I,S,,,F,,F,F
- TOKEN_LABELS=Cpu Name,Cpu number,Description,,,value 1,,value 2,value 3
- TOKEN_TYPES が設定されない場合、TOKEN_LABELS は無視されます。

TOKEN_SEPARATOR
オプション - デフォルトはセミコロン「;」。トークンの出力行を分割するための区切り文字として使用する文字列を設定します。TOKEN_TYPES が設定されない場合は無視されます。空の値 (ブランク) は区切り文字として受け入れられますが、出力行内の連続した複数のブランクは単一のブランクと見なされます。

次の 2 つのパラメーターを使用すると、スクリプトの行出力をフィルタリングできます。これらは OS エージェントによって最初のトークンにのみ適用され、2 つ一緒に使用する必要があります。

FILTER_VALUE
オプションです。 比較に使用される値。FILTER_OPERATOR が定義されている場合は必須です。 OUTPUT_TYPE が文字列の場合、フィルター値は、フィルターを掛ける対象のスクリプトによって返される文字列値を、追加の引用符なしで正確に反映する必要があります (ワイルドカードは使用できません)。

FILTER_OPERATOR
オプションです。 比較に使用される演算子。FILTER_VALUE が定義されている場合は必須です。 使用できる FILTER_OPERATOR 値は次のとおりです。

- = (等しい)
- != (等しくない)
- > (より大きい) 数値タイプのみ
- >= (より小さくない) 数値タイプのみ
- < (より小さい) 数値タイプのみ
- <= (より大きくない) 数値タイプのみ

プロパティー・ファイルの例
プロパティー・ファイルのパラメーター設定例。

最初のスクリプトの定義: スクリプト ex_script1.sh が 150 秒ごとに開始されます。浮動小数点値を返し、エージェントは 0.5 に等しい出力行のみを考慮します。

```
ATTRIBUTE_NAME=sample1
SCRIPT_PATH_WITH_PARMS=/opt/ibm/apm/agent/localconfig/lz/scripts_definitions/ex_script1.sh
EXECUTION_FREQUENCY=150
OUTPUT_TYPE=FLOAT
FILTER_VALUE=0.5
FILTER_OPERATOR==
```

2 つ目のスクリプトの定義: スクリプト ex_script2 が 60 秒ごとに開始されます。整数値を返し、エージェントは 0 でない行のみを考慮します。

```
ATTRIBUTE_NAME=ex_script2
SCRIPT_PATH_WITH_PARMS=${CANDLE_HOME}/tmp/check_out.sh
EXECUTION_FREQUENCY=60
OUTPUT_TYPE=INTEGER
```

```
FILTER_VALUE=0
FILTER_OPERATOR=!=
```

3 つ目のスクリプトの定義: スクリプト ex_script3.sh が、入力パラメーターを 3 つ指定された状態で (最初の入力パラメーターは整数、2 番目と 3 番目は文字列です)、120 秒ごとに開始されます。このスクリプトは、停止するか、実行時間がタイムアウト値を超えると終了します。

```
ATTRIBUTE_NAME=ex_script3
SCRIPT_PATH_WITH_PARMS=/opt/scripts/ex_script3.sh 1 "second input parameter" "third input
parameter"
EXECUTION_FREQUENCY=120
OUTPUT_TYPE=STRING
KILL_AFTER_TIMEOUT=TRUE
```

4 つ目のスクリプトの定義: スクリプト cpu_mem_percentage.sh が 50 秒ごとに開始され、標準出力の文字列として cpuid と、アイドル状態の CPU の割合と使用中の CPU の割合を示す浮動小数点値が 2 つ、およびメモリー使用量と仮想メモリー使用量を示す整数が 2 つ返されます。出力を解析するために区切り文字としてパイプが使用されます。このスクリプトによって返される行の例を以下に示します。

cpu2|35,5|65,5|3443|123800

```
ATTRIBUTE_NAME=cpu and mem Usage
SCRIPT_PATH_WITH_PARMS=${SCRIPT_HOME}/cpu_mem_percentage.sh
OUTPUT_TYPE=STRING
TOKEN_TYPES=F,F,I,I
TOKEN_LABELS= Idle CPU %, Used CPU %, Virt MEM used MB, MEM used MB
TOKEN_SEPARATOR=|
EXECUTION_FREQUENCY=50
```

既知の問題と制限事項
既知の問題と制限事項

- Windows™ 2003 64 ビット・システムではスクリプト機能がサポートされていません。
- Solaris™ と Windows™ 32 ビットの OS エージェントでタイムアウト後の強制終了が機能しません。
- タイムアウト期間内に一部のスクリプトが完了せず、ユーザーが負荷の高いトレースを有効にした場合、Windows™ 32 ビット版では、fcp_daemon がスクリプトの実行を停止する場合があります。実行中のスクリプトが fcp_daemon によって停止される場合、コンソールに表示されるデータには、スクリプトが最後に実行された時間が反映されます。実行中のデータは OS エージェントによって停止される可能性もあります。fcp_daemon プロセスを停止すると、エージェントが適切な操作を再開できます。
- Solaris™ で、SCRIPT_NOT_FOUND または SCRIPT_LAUNCH_ERROR の代わりに SCRIPT_NONZERO_RETURN が返されます。
- スクリプト機能は、完全なグローバリゼーションを提供していません。プロパティー・ファイルやスクリプトの出力に各国語文字を使用すると、いくつかの問題が見つかる場合があります。
- Windows™ OS エージェントで、マップされたネットワーク・ドライブにあるスクリプトが実行される可能性がありません。
- Windows™ OS エージェントをアップグレードすると、スクリプト機能がデフォルトで有効になりません。KNTENV を編集し、「KNT_FCP_SCRIPT=FALSE」を「KNT_FCP_SCRIPT=TRUE」に変更します。

カスタム・スクリプトのトラブルシューティング
カスタム・スクリプトのトラブルシューティング

標準の *KBB_RAS1* 変数は、OS エージェントと fcp_daemon プロセスに適用されます。特定のトレース設定を fcp_daemon のみに適用するには、*KXX_FCP_KBB_RAS1* 変数を使用します。*KXX_FCP_KBB_RAS1* を設定すると、*KBB_RAS1* に指定されている値は fcp_daemon によって無視されます。

機能の OS エージェントのコア・スレッドによってログに記録された操作をトレースするには、以下を設定します。

```
KBB_RAS1=ERROR (UNIT:factory ALL)
```

APM サーバーからのスクリプトの照会と、サーバーに送信されたイベントをトレースするには、以下のエントリーを追加します。

Monitoring Agent for Linux OS の場合

```
(UNIT:klz34 ALL) (UNIT:klz36 ALL)
```

Monitoring Agent for UNIX OS の場合

```
(UNIT:kux48 ALL) (UNIT:kux50 ALL)
```

Monitoring Agent for Windows OS の場合

```
(UNIT:knt84 ALL) (UNIT:knt86 ALL)
```

TEMA トレースを表示して専用シチュエーションの実行を検証するには、以下のエントリーを追加します。

```
(UNIT:kraavp all) (UNIT:kraapv all)
```

スクリプトの実行と、スクリプトから得たデータの解析方法を確認するには、以下を設定します。

```
KXX_FCP_KBB_RAS1=Error (UNIT:command ALL)
```

OS エージェントと fcp_daemon の間の通信の問題をトラブルシューティングするには、以下のトレース・レベルを *KBB_RAS1* と *KXX_FCP_KBB_RAS1* の両方に追加します。

```
(UNIT:cps_socket FLOW) (UNIT:cpci FLOW)
```

OS エージェントのプロセスと fcp_daemon の間の対話を詳細に確認するには、*KBB_RAS1* と *KXX_FCP_KBB_RAS1* の両方に以下を追加します。

```
(UNIT:cps_socket ALL) (UNIT:cpci ALL)
```

クイック・スタート・シナリオ
このシナリオでは、サンプル・シナリオのカスタム・スクリプトを構成するのに必要な最小ステップを紹介します。

以下のセクションでは、Monitoring Agent for Linux OS が 2 つのカスタム・スクリプトを実行するよう構成するのに必要な最小ステップについて説明します。

カスタム・スクリプトの説明

この例では、ユーザーは /scripts_repo ディレクトリーの下に次の 2 つのスクリプトを持っています。

checkDIRsize.sh – このスクリプトは、入力パラメーターとして渡される指定されたディレクトリーのサイズを確認します。出力は 4594740 のような整数です。

cpu_mem_usage.sh – このスクリプトは、CPU 使用率とスワップ・メモリー使用量 (メガバイト) を確認します。 出力は cpu1|96,5|23800 の形式で返されます。

最初のトークンは CPU ID、2 つ目のトークンは CPU 使用率、3 つ目のトークンはスワップ・メモリー使用量 (メガバイト単位) です。

Monitoring Agent for Linux OS でこれらのスクリプトを実行するには、カスタマイズが必要です。

この機能は、OS エージェントが開始されるとすぐに、デフォルト値で有効になります。

プロパティー・ファイル AnyName.properties をデフォルト・ディレクトリー install dir/localconfig/lz/scripts_definitions の下に作成します。この例では、各スクリプトに 1 つずつ、checkDIRsize.properties と cpu_mem_usage.properties という名前で、計 2 つのプロパティー・ファイルを作成します。

```
#CheckDIRsize.properties
ATTRIBUTE_NAME=OPT_DIR_SIZE
SCRIPT_PATH_WITH_PARMS=/scripts_repo/checkDIRsize.sh /opt
EXECUTION_FREQUENCY=20
OUTPUT_TYPE=INTEGER
```

```
#cpu_mem_usage.properties
ATTRIBUTE_NAME=cpu_mem_usage
SCRIPT_PATH_WITH_PARMS=/scripts_repo/cpu_mem_percentage.sh
OUTPUT_TYPE=string
TOKEN_TYPES=F,I
TOKEN_LABELS= Used CPU %, Swap MEM used MB
TOKEN_SEPARATOR=|
EXECUTION_FREQUENCY=10
```

これら 2 つのプロパティー・ファイルを追加 (または変更) した後に、OS エージェントを再始動する必要はありません。OS エージェントは、指定した時間間隔 (デフォルト値は 300 秒) で、スクリプト定義のディレクトリーを確認します。 スクリプトの詳細と結果は、コンソールを開き、「カスタム・スクリプト」ワークスペースの下で確認できます。

Linux OS エージェント・ファイル・システム・データ収集の構成

Monitoring Agent for Linux OS は自動的に構成されます。ただし、ファイル・システム・データ収集の動作を構成できます。

Monitoring Agent for Linux OS には、ファイル・システム・データ収集のデフォルト動作があります。

デフォルト動作では、/etc/fstab のみのファイル・システムがモニターされます。ファイル・システム・データ収集動作を制御するために、環境変数 *KBB_SHOW_MTAB_FS* が lz.environment ファイルに定義されています。(/etc/fstab と /etc/mtab にリストされている) すべてのファイル・システムをモニターする場合は、KBB_SHOW_MTAB_FS=true に設定します。

KBB_SHOW_MTAB_FS
この変数は、*install_dir*/config/*.lz*.environment ファイルにあります。 デフォルト値は false であり、エージェントは /etc/fstab のみのファイル・システムをモニターするように定義されています。(/etc/fstab と /etc/mtab にリストされている) すべてのファイル・システムをモニターする場合は、この値を true に変更します。例えば、*KBB_SHOW_MTAB_FS=true* のようにします。

PHP モニターの構成

エージェントがモニター対象の PHP アプリケーションからデータを収集できるように、Monitoring Agent for PHP を構成する必要があります。

始める前に

1. php-process パッケージを必ずインストールしてください。yum install コマンドを使用して PHP をインストールする場合は、yum install php-process コマンドを実行して php-process パッケージをインストールします。
2. エージェントを構成する前に、必ず Apache HTTPD サーバーを始動してください。

   Apache HTTP サーバーの httpd.conf 構成ファイルを開き、mod_status と ExtendedStatus On の両方のオプションが有効になっていることを確認してください。例:

   ```
   ExtendedStatus On
   <Location /server-status>
           SetHandler server-status
           Order deny,allow
           Allow from all
           Allow from 127.0.0.1
   </Location>
   ```

   この例では、エージェントが正しく動作するために http://127.0.0.1/server-status が正常に動作する必要があります。

   注: エージェントがモニター・データを取得するためには、Lynx または Links が Linux にインストールされていなければなりません。

apachectl コマンドにコード変更を加えていないモニター対象の Apache サーバーで、コマンド apachectl status が正しく動作することを確認します。コマンド apachectl status が正しく動作するためには、Lynx がインストールされていなければなりません。

このタスクについて

エージェントの構成時に権限の問題が発生しないようにするために、必ず、エージェントのインストールに使用したのと同じ root ユーザー ID または非 root ユーザー ID を使用してください。選択されたユーザーとしてエージェントをインストールした後で、別のユーザーとしてエージェントを構成する場合は、187 ページの『非 root ユーザーとしてのエージェントの構成』を参照してください。選択されたユーザーとしてエージェントをインストールおよび構成した後で、別のユーザーとしてエージェントを開始する場合は、1013 ページの『非 root ユーザーとしてのエージェントの開始』を参照してください。

PHP エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は 32 文字に制限されています。指定するインスタンス名は 28 文字 (ホスト名の字数は除く) までに制限されています。例えば、インスタンス名に PHP2 を指定すると、管理対象システム名は PHP2:hostname:PJ になります。

重要: 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが正しく表示されません。

手順

- ご使用の環境がデフォルト設定と同じ場合、デフォルトの実行バイナリー・パス、デフォルトの php.ini ファイル・パス、およびデフォルト・ポートを使用してエージェントを構成できます。
  a) 以下を入力します。
  install_dir/bin/php-agent.sh config *instance_name* *install_dir*/samples/php_silent_config.txt
  ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は PHP エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
  b) エージェントを開始するには、以下を入力します。
  install_dir/bin/php-agent.sh start *instance_name*
- サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで *install_dir*/samples/php_silent_config.txt を開きます。
  b) **Location of PHP execution binary** には、PHP 実行ファイルが格納されているディレクトリーを指定できます。デフォルトの場所は、/usr/local/bin です。
  c) **Location of PHP INI file** には、php.ini ファイルが格納されているディレクトリーを指定できます。 デフォルトの場所は、/etc です。
  d) **Web server port** には、WordPress を実行している Web サーバーのポート番号を指定できます。デフォルトは 80 です。
  e) **Application DocumentRoot** には、PHP WordPress アプリケーションのドキュメント・ルートを指定できます。複数のレコードを区切るにはコロンを使用します。エージェントですべてのレコードを自動的に検索するには、デフォルト値の ALL を使用します。
  f) php_silent_config.txt フィルを保存して閉じてから、以下を入力します。
  install_dir/bin/php-agent.sh config *instance_name* *install_dir*/samples/php_silent_config.txt
  ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は PHP エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
  g) エージェントを開始するには、以下を入力します。

install_dir/bin/php-agent.sh start *instance_name*

- スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

  a) 以下を入力します。
  install_dir/bin/php-agent.sh config *instance_name*
  ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は PHP エージェントのインストール・ディレクトリーです。

  b) `Edit Monitoring Agent for PHP settings` というプロンプトが出されたら、1 を入力して続行します。

  c) `Location of PHP execution binary` というプロンプトが出されたら、Enter キーを押してデフォルトの場所を受け入れるか、独自の場所を指定します。

  d) `Location of PHP INI file` というプロンプトが出されたら、Enter キーを押してデフォルトの場所を受け入れるか、独自の場所を指定します。

  e) `Web server port` というプロンプトが出されたら、Enter キーを押してデフォルトのポートを受け入れるか、別のポート番号を指定します。

  f) `Application DocumentRoot` というプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、PHP WordPress アプリケーションのドキュメント・ルートを指定します。コロンを使用して複数のレコードを区切ることができます。

  g) エージェントを開始するには、以下を入力します。
  install_dir/bin/php-agent.sh start *instance_name*

タスクの結果

エージェントは、WordPress アプリケーションでの PHP 要求のパフォーマンスのみを評価します。CSS および JS のロードは評価されません。エージェントは、URL を識別するために URL 引数を使用しません。

次のタスク

PHP エージェントのデータが Cloud APM コンソール に表示されることを確認できます。

エージェント用の WordPress プラグインがアクティブ化されていることを確認する必要があります。 確実にアクティブ化するには、以下のステップを実行します。

1. Web ブラウザーで、URL `http://`*hostname*`:`*port*`/wp-admin/` を入力します。
2. 「**プラグイン**」>「**インストール済みプラグイン (Installed Plugins)**」にナビゲートして、管理ページにアクセスします。
3. PHP エージェント プラグインがアクティブ化されていることを確認します。PHP エージェントのプラグインが「**WordPress エージェント**」としてリストされています。通常、プラグインは既にアクティブ化されています。まだアクティブ化されていない場合は、「**アクティブ化 (Activate)**」をクリックします。

PostgreSQL のモニターの構成

エージェントがモニター対象の PostgreSQL データベースからデータを収集できるように、Monitoring Agent for PostgreSQL を構成する必要があります。

始める前に

このエージェントをインストールする前に、PostgreSQL JDBC ドライバーをインストールする必要があります。エージェントの構成時には、このドライバーのパスが必要です。

JDBC タイプ 4 ドライバーが新しいバージョンであるため、このバージョンを推奨します。エージェントが使用する JDK のバージョンに従って、ユーザーは JDBC 4 バージョンのサブタイプをインストールできます。JDBC のバージョンと JDK のバージョンのマッピングについて詳しくは、https://jdbc.postgresql.org/download.html を参照してください。

エージェントによって収集されるいくつかの属性は、pg_stat_statements 拡張に依存します。pg_stat_statements を追加するために、最初にパッケージ postgresql-contrib をインストールします。PostgreSQL サーバーが pg_stat_statements 拡張をロードするように postgresql.conf 構成ファイルを変更する必要があります。

1. postgresql.conf ファイルをテキスト・エディターで開き、以下のように shared_preload_libraries の行を更新します。

   ```
   shared_preload_libraries = 'pg_stat_statements'
   pg_stat_statements.track_utility = false
   ```

   これらの変更は、ユーティリティー・コマンド以外の SQL ステートメントをモニターするために必要です。

   注: pg_stat_statements.track_utility の状況の設定または変更は、スーパーユーザーのみが行います。

2. postgresql.conf を更新して保存した後、PostgreSQL サーバーを再始動します。
3. psql を使用して以下の SQL コマンドを実行します (JDBC 接続のために後でエージェント構成で指定するデータベースと同じデータベースに接続する必要があります)。

   ```
   create extension pg_stat_statements;
   select pg_stat_statements_reset();
   ```

   注: コマンド create extension および関数 pg_stat_statements_reset() は、スーパーユーザーによってのみ実行されます。

   特定のデータベースではビュー pg_stat_statements を有効にする必要があります。詳しくは、https://www.postgresql.org/docs/9.6/static/pgstatstatements.html を参照してください。

pg_hba.conf ファイルは PostgreSQL データベース・ファイルであり、このファイルには認証設定が格納されています。pg_hba.conf ファイルの auth-method パラメーター値を ident に設定すると、PostgreSQL エージェントは PostgreSQL データベースに接続できません。auth-method パラメーターの認証設定が正しいことを確認してください。例えば、auth-method パラメーターには md5、trust、password の値を設定できます。

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the PostgreSQL エージェント を参照してください。

このタスクについて

PostgreSQL エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は 32 文字に制限されています。指定するインスタンス名は 28 文字 (ホスト名の字数は除く) までに制限されています。例えば、インスタンス名に PostgreSQL2 を指定すると、管理対象システム名は PostgreSQL2:hostname:PN になります。

重要: 長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが完全には表示されません。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for PostgreSQL**」を右クリックしてから「**エージェントを構成します**」をクリックします。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。
4. 「**Monitoring Agent for PostgreSQL**」ウィンドウで、次の手順を実行します。
   a. 「**IP アドレス**」フィールドに、リモート側でモニターする PostgreSQL サーバーの IP アドレスを入力します。モニター対象のサーバーにエージェントがインストールされている場合は、デフォルト値を保持します。

      注 :

      リモート・モニターの場合、「**現在の CPU 使用率 (%)**」および「**使用物理メモリー (MB)**」のデータはダッシュボードに表示されません。これらのウィジェットには「**N/A**」と表示されます。
   b. 「**JDBC データベース名**」フィールドに、データベース名を入力して、デフォルトのデータベース名の `postgres` を変更します。
   c. 「**JDBC ユーザー名**」フィールドに、ユーザー名を入力して、デフォルト名の `postgres` を変更します。
   d. 「**JDBC パスワード**」フィールドに、JDBC ユーザー・パスワードを入力します。
   e. 「**JDBC パスワードの確認 (Confirm JDBC password)**」フィールドに、パスワードを再入力します。
   f. 「**JDBC ポート番号**」フィールドに、ポート番号を入力して、デフォルトのポート番号の 5432 を変更します。
   g. 「**JDBC JAR ファイル (JDBC JAR file)**」フィールドに、Java JAR ファイルの PostgreSQL コネクターへのパスを入力し、「**次へ**」をクリックします。
   h. 「**Java トレース・レベル**」フィールドに、IBM サポートの説明に従って、トレース・レベルを入力します。デフォルトのトレース・レベルは `Error` です。
   i. 「**OK**」をクリックします。 「IBM Performance Management」ウィンドウに、エージェント・インスタンスが表示されます。
5. 「**Monitoring Agent for PostgreSQL**」インスタンスを右クリックして「**開始**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド行に以下のコマンドを入力します。
   install_dir/bin/postgresql-agent.sh config *instance_name*
2. PostgreSQL 設定用にエージェントを編集するようプロンプトが出されたら、1 を入力して続行します。

3. 以下のパラメーターの値を入力するようにプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、別の値を指定して Enter キーを押します。
   - IP アドレス

     注 :

     リモート側からモニターする PostgreSQL サーバーの IP アドレスを入力します。モニター対象のサーバーにエージェントがインストールされている場合は、デフォルト値を保持します。

     リモート・モニターの場合、「**現在の CPU 使用率 (%)**」および「**使用物理メモリー (MB)**」のデータはダッシュボードに表示されません。これらのウィジェットには「**N/A**」と表示されます。
   - JDBC データベース名
   - JDBC ユーザー名
   - JDBC パスワード
   - JDBC ポート番号
   - JDBC JAR ファイル

   重要 : JDBC JAR ファイルのバージョンは、モニター対象の PostgreSQL データベースのバージョンと同じでなければなりません。
4. 「Java トレース・レベル (Java trace level)」パラメーターの値を入力するようプロンプトが出されたら、2 を入力してデフォルト値を受け入れるか、IBM サポートの指示に従って、トレース・レベルを指定します。
5. 以下のコマンドを実行して、エージェントを開始します。

```
install_dir/bin/postgresql-agent.sh start instance_name
```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システムおよび Windows システム上で PostgreSQL エージェントを構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント応答ファイルを編集し、プロンプトに応答せずにスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。
  1. テキスト・エディターで、パス *install_dir*/samples/postgresql_silent_config.txt にあるサイレント応答ファイルを開きます。ここで、*install_dir* は PostgreSQL エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
  2. サイレント構成ファイルを編集するために、以下の手順を実行します。

a. **「IP アドレス」**パラメーターには、リモート側からモニターする PostgreSQL サーバーの IP アドレスを指定します。モニター対象のサーバーにエージェントがインストールされている場合は、デフォルト値を保持します。

   注 :

   リモート・モニターの場合、「**現在の CPU 使用率 (%)**」および「**使用物理メモリー (MB)**」のデータはダッシュボードに表示されません。これらのウィジェットには「**N/A**」と表示されます。

b. **JDBC database name** パラメーターに、データベース名を指定して、postgres のデフォルトのデータベース名を変更します。

c. **JDBC user name** パラメーターに、ユーザー名を指定して、postgres のデフォルト名を変更します。

d. **JDBC password** パラメーターに、JDBC ユーザー・パスワードを入力します。

e. **JDBC port number** パラメーターに、ポート番号を指定して、デフォルトのポート番号 5432 を変更します。

f. **JDBC JAR file** パラメーターでは、デフォルト・パスが正しくない場合に、Java JAR ファイルの PostgreSQL コネクターのパスを指定します。Java JAR ファイルのデフォルト・パスは次のとおりです。

   /opt/PostgreSQL/lib/postgresql-9.3-1100.jdbc4.jar

   重要 : JDBC JAR ファイルのバージョンは、モニター対象の PostgreSQL データベースのバージョンとの互換性があるバージョンでなければなりません。

g. **Java trace level** パラメーターでは、IBM サポートの指示に従って、トレース・レベルを指定します。デフォルトのトレース・レベルは Error です。

3. サイレント応答ファイルを保存して閉じ、次のコマンドを実行します。

```
install_dir/bin/postgresql-agent.sh config
instance_name
install_dir/samples/postgresql_silent_config.txt
```

   ここで、*instance_name* は、インスタンスに指定する名前です。

4. エージェントを始動するために、以下のコマンドを入力します。

```
install_dir/bin/postgresql-agent.sh start instance_name
```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

Python モニターの構成

オンプレミスと IBM Cloud の両方の Python アプリケーションをモニターできます。ご使用のアプリケーション・タイプに応じて、対応する構成手順を実行してください。

このタスクについて

オンプレミスと IBM Cloud の Python アプリケーションをモニターするように Python データ・コレクターを構成します。

手順

- IBM Cloud アプリケーションをモニターするようにデータ・コレクターを構成します。

a) IBM Cloud アプリケーションに関するデータを収集して送信するように Python データ・コレクターを構成します。説明については、673 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成』を参照してください。

b) オプション: Python データ・コレクターのモニター機能をカスタマイズします。詳しくは、674 ページの『IBM Cloud アプリケーション用の Python データ・コレクター のカスタマイズ』を参照してください。

- オンプレミス・アプリケーションをモニターするようにデータ・コレクターを構成します。

a) データを収集して Cloud APM サーバーに送信するようにデータ・コレクターを構成します。説明については、678 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成』を参照してください。

b) オプション: Python データ・コレクターのモニター機能をカスタマイズします。詳しくは、679 ページの『オンプレミス・アプリケーション用の Python データ・コレクターのカスタマイズ』を参照してください。

IBM Cloud アプリケーション用の Python データ・コレクターの構成

IBM Cloud 上の Python アプリケーションに関する情報を収集するには、Python データ・コレクターを構成する必要があります。

始める前に

1. モニターする Python アプリケーションが固有の名前を持っていることを確認してください。Python データ・コレクターは 2 つの異なるアプリケーションが同じ名前を持っていると 1 つのアプリケーションとして処理するため、Cloud APM コンソールでデータ表示の問題が発生することがあります。
2. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

このタスクについて

データ・コレクターを構成するには、まず pypi パッケージ・サーバーをデプロイしてから、データ・コレクターを Python Django アプリケーションにインストールします。

手順

1. データ・コレクター・パッケージからファイルを抽出します。
`python_datacollector_8.1.4.0.tgz` パッケージは、解凍されたディレクトリーに含まれています。
2. 以下のようなコマンドを実行して、`python_datacollector_8.1.4.0.tgz` パッケージを解凍します。

```
tar -zxf python_datacollector_8.1.4.0.tgz
```

3. 解凍されたディレクトリー内でパッケージ・サーバーの `manifest.yml` ファイルを見つけ、以下の例に従って、このファイルで domain、host、および name を定義します。

```
domain: mybluemix.net
name: pythondc
host: pythondc
```

要確認 : *host* 値と *name* 値は同一かつ固有である必要があります。

4. python_dc ディレクトリーで、以下のコマンドを実行して pythondc アプリケーションを IBM Cloud にプッシュします。

```
cf push
```

5. Python アプリケーションの `requirements.txt` ファイルで、以下の行を追加します。

```
cryptography==1.9.0
--extra-index-url https://<your_host_name_and_domain>/python-dc-repos/simple/
ibm_python_dc
```

6. Python アプリケーションの settings.py ファイルで、ibm_python_dc.kpg_plugin.ResourceMiddleware を MIDDLEWARE_CLASSES セクションの先頭に追加します。以下に例を示します。

```
MIDDLEWARE_CLASSES = (
    "ibm_python_dc.kpg_plugin.ResourceMiddleware",
    "mezzanine.core.middleware.UpdateCacheMiddleware",
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
```

7. Python アプリケーションの manifest.yml ファイルを含むディレクトリーから、以下のコマンドを実行します。

```
cf push
```

ヒント: サンプルの manifest.yml ファイルについては、192 ページの『サンプル manifest.yml ファイル』を参照してください。

タスクの結果

データ・コレクターが構成され、Cloud APM サーバーに接続されました。

次のタスク

ご使用の IBM Cloud アプリケーションのモニター・データが Cloud APM コンソールに表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

IBM Cloud アプリケーション用の Python データ・コレクター のカスタマイズ

IBM Cloud ユーザー・インターフェース (UI) で環境変数を追加して、ご使用の IBM Cloud アプリケーションのモニターをカスタマイズできます。以下の情報を使用して、ニーズに応じて変数を追加します。

Python データ・コレクター用のユーザー定義の環境変数

以下の表の情報を使用して、IBM Cloud での Python モニターをカスタマイズできます。

表 *192. IBM Cloud* での *Python* モニター用にサポートされるユーザー定義の環境変数

変数名	重要度	値	説明
APM_BM_GATEWAY_URL	オプション	• https://*<server ip または hostname>*:443 • http://*<server ip または hostname>*:80	ターゲットのオンプレミス・サーバー・ゲートウェイ URL。
APM_KEYFILE_PSWD	オプション	鍵ファイルの暗号化されたパスワード	鍵ファイルと対になっている暗号化された鍵ファイル・パスワード。Linux ユーザーの場合は、echo -n *<keyfile password>* | base64 コマンドを使用してパスワードを暗号化できます。 **注:** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。

表 *192. IBM Cloud* での *Python* モニター用にサポートされるユーザー定義の環境変数 *(*続き*)*

変数名	重要度	値	説明
APM_KEYFILE_URL	オプション	http://*<hosted http server>*:*<port>*/keyfile.p12	鍵ファイルをダウンロードするための URL。 **注 :** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。
KPG_ENABLE_DEEPDIVE	オプション	• False • True	診断データの収集を有効または無効にします。 • True: デフォルト値。この変数を `True` に設定すると、診断データが収集されます。 • False: この変数を `False` に設定すると、診断データは収集されません。 この変数を設定しない場合、診断データは収集されます。
KPG_DD_CONFIG_FILE	オプション	診断モニターの構成ファイルのファイル名。	診断モニターの構成ファイルのファイル名。デフォルトのファイル名は、`kpg_dd_config.xml` です。 **注 :** このファイル内で設定値をカスタマイズした後、ファイルをアプリケーションのルート・ディレクトリーに配置する必要があります。 この変数を設定しない場合、データ・コレクター・パッケージ内のデフォルト構成ファイル `kpg_dd_config.xml` が使用されます。
KPG_DD_APP_PATH	オプション	Python アプリケーションへのパス。	データ・コレクターが診断データを収集する対象の Python アプリケーションまたはモジュールへのパス。モニターする Python アプリケーションおよびモジュールが複数ある場合は、それぞれのパスをセミコロン ; で区切ります。 この変数を設定しない場合、データ・コレクターは、アプリケーションによって使用される要求およびモジュールのデータを収集します。Python lib 内の要求のデータは収集されません。

表 *192. IBM Cloud* での *Python* モニター用にサポートされるユーザー定義の環境変数 *(*続き*)*

変数名	重要度	値	説明
KPG_DD_SECURITY_FILTER	オプション	• True • False	• True: デフォルト値。この変数を True に設定すると、値 (パスワードなど) が SQL ステートメント内でマスクされ、グループ・ウィジェット「**要求コンテキスト**」にパラメーターが表示されません。 • False: この変数を False に設定すると、SQL ステートメント内の値はマスクされず、グループ・ウィジェット「**要求コンテキスト**」にパラメーターが表示されます。 この変数を設定しない場合、値 (パスワードなど) が SQL ステートメント内でマスクされ、グループ・ウィジェット「**要求コンテキスト**」にパラメーターが表示されません。
KPG_GC_STATS	オプション	True	python ガーベッジ・コレクションのすべての統計関数が有効になります。この値を True に設定すると、以下のコマンドを実行した場合と同じ結果が得られます。 `gc.set_debug(gc.DEBUG_STATS | gc.DEBUG_COLLECTABLE | gc.DEBUG_UNCOLLECTABLE | gc.DEBUG_INSTANCES | gc.DEBUG_OBJECTS )` KPG_GC_STATS を無効にするには、この環境変数を削除します。False に設定しないでください。 **注 :** 正式な実稼働環境では、決して KPG_SAVE_ALL=True を設定しないでください。これは、デバッグ・モード専用です。アプリケーションに十分なメモリーが割り当てられていることを確認してください。
KPG_LOG_LEVEL	オプション	• DEBUG • ERROR • INFO	• DEBUG: 有用なデバッグ情報のみがログに出力されます (収集されたデータ、サーバーに送信されたデータ、サーバーの応答など)。 • ERROR: 例外および予期しないシチュエーションに関する情報のみがログに出力されます。 • INFO: ユーザーがデータ・コレクターの処理内容を把握するための、データ・コレクターに関する要約情報がログに出力されます。

表 192. *IBM Cloud* での *Python* モニター用にサポートされるユーザー定義の環境変数 (続き)			
変数名	**重要度**	**値**	**説明**
KPG_LOG_TOCONSOLE	オプション	• Y • True • False 以外のすべての値	ログがコンソールに出力され、コマンド **cf logs <appname>** を実行してログを表示できます。
KPG_SAVE_ALL	オプション	True	参照されていないオブジェクトはすべて gc.garbage に保存されるため、1 分ごとに gc.garbage をクリアする必要があります (データ・コレクターによってクリアされます)。この値を True に設定すると、以下のコマンドを実行した場合と同じ結果が得られます。 `gc.set_debug(gc.SAVE_ALL)` KPG_SAVE_ALL を無効にするには、この環境変数を削除します。False に設定しないでください。 **注:** 正式な実稼働環境では、決して KPG_SAVE_ALL=True を設定しないでください。これはデバッグ・モード専用です。アプリケーションに十分なメモリーが割り当てられていることを確認してください。

IBM Cloud アプリケーション用の Python データ・コレクターの構成解除
Python 環境をモニターする必要がない場合、または Python データ・コレクターをアップグレードする場合は、最初に Python データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. Python アプリケーションのホーム・ディレクトリーに移動します。
2. アプリケーションの requirements.txt から以下の行を削除します。

```
--extra-index-url https://<your_host_name_and_domain>/python_dc/static/python-dc-repos/
simple/
ibm-python-dc
```

3. settings.py ファイルで、MIDDLEWARE_CLASSES セクションから以下の行を削除します。

```
ibm_python_dc.kpg_plugin.ResourceMiddleware
```

4. 以下のコマンドを実行して、アプリケーションを再プッシュし、変更を有効にします。

```
cf push
```

タスクの結果
Python データ・コレクターが正常に構成解除されました。

次のタスク
データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示され

ません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

オンプレミス・アプリケーション用の Python データ・コレクターの構成

ローカル環境で実行される Python アプリケーションに関する情報を収集するには、Python データ・コレクターを構成する必要があります。

始める前に

1. モニターする Python アプリケーションが固有の名前を持っていることを確認してください。Python データ・コレクターは 2 つの異なるアプリケーションが同じ名前を持っていると 1 つのアプリケーションとして処理するため、Cloud APM コンソールでデータ表示の問題が発生することがあります。
2. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

このタスクについて

データ・コレクター・パッケージは、etc フォルダーにコピーされた事前構成済み global.environment ファイルおよび keyfile.p12 ファイルを含む事前構成済みパッケージです。結果として、データ・コレクターは、Cloud APM サーバーに自動的に接続します。

以下の手順では、デフォルト設定を使用して Python アプリケーション内のデータ・コレクターを構成します。データ・コレクターの構成をカスタマイズするには、データ・コレクター構成ファイル内の環境変数を使用します。詳しくは、679 ページの『オンプレミス・アプリケーション用の Python データ・コレクターのカスタマイズ』を参照してください。

手順

1. データ・コレクター・パッケージからファイルを抽出します。
   python_datacollector_8.1.4.0.tgz パッケージは、解凍されたディレクトリーに含まれています。
2. データ・コレクター・パッケージからファイルを解凍します。例えば、以下のコマンドを実行します。

   ```
   tar -zxf python_datacollector_8.1.4.0.tgz
   ```

3. python_dc ディレクトリーから、以下のコマンドを実行します。

   ```
   python server.py
   ```

4. 次のコマンドを実行します。

   ```
   pip install ibm_python_dc --extra-index-url http://host name or ip:8000/
   python-dc-repos/simple/ --trusted-host host name or ip
   ```

   ここで、*host name or ip* は、Python データ・コレクター・リポジトリーを実行するホストの名前または IP アドレスです。

   重要 : このコマンドで URL とトラステッド・ホストの両方に対するホストを指定するには、名前または IP アドレスのいずれかを使用します。例えば、IP アドレスを使用してホストを指定し、IP アドレスが 9.42.36.180 である場合、コマンドは次のようになります。

   ```
   pip install ibm_python_dc --extra-index-url http://9.42.36.180:8000/
   python-dc-repos/simple/ --trusted-host 9.42.36.180
   ```

5. Python アプリケーションの settings.py ファイルで、以下の例のフォーマットのように、MIDDLEWARE_CLASSES セクションに ibm_python_dc.kpg_plugin.ResourceMiddleware を追加します。

   ```
   MIDDLEWARE_CLASSES = (
       "ibm_python_dc.kpg_plugin.ResourceMiddleware",
       "mezzanine.core.middleware.UpdateCacheMiddleware",
   ```

```
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
```

タスクの結果
データ・コレクターは、デフォルト設定を使用して構成され、Cloud APM サーバーに接続されます。

次のタスク
これで、Cloud APM サーバーにログインして、モニター・データを表示できます。

要確認 : Python アプリケーションを Cloud APM コンソールに追加した後で、Python ランタイム・アプリケーションという名前のコンポーネントでモニター・データを表示できます。

Cloud APM サーバーを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

オンプレミス・アプリケーション用の Python データ・コレクターのカスタマイズ
データ・コレクター・パッケージ内のファイルを変更して、Python アプリケーションのモニターをカスタマイズするための環境変数を設定できます。

データ・コレクター設定をカスタマイズするための 2 つのファイル `global.environment` および `config.properties` が提供されています。これらのファイル内の設定を変更した後で、Python アプリケーションを再始動して、変更を有効にします。

`global.environment` ファイルを変更すると、データ・コレクターと Cloud APM サーバーの間の接続をカスタマイズできます。デフォルト・サーバーの代わりに別の Cloud APM サーバーを使用する場合や、鍵ファイルまたはそのパスワードが変更された場合は、Cloud APM サーバーを変更して、データ・コレクターを Cloud APM サーバーに再接続します。

`config.properties` ファイルを変更すると、ニーズに従って、メソッド・トレースを有効化または無効化するなど、データ・コレクターの動作をカスタマイズできます。

`global.environment` 構成ファイル

679 ページの表 193 は、`global.environment` 構成ファイル内で設定できる環境変数および関連する説明を示しています。`global.environment` ファイルは、Python データ・コレクターがインストールされている `etc` フォルダー (例えば、`/root/.pyenv/versions/3.5.2/lib/python3.5/site-packages/ibm_python_dc/etc` ディレクトリー) にあります。

表 *193. `global.environment`* ファイル内でサポートされる環境変数

変数名	重要度	値	説明
APM_BM_GATEWAY_URL	オプション	• https://*<server ip または hostname>*:443 • http://*<server ip または hostname>*:80	ターゲットのオンプレミス・サーバー・ゲートウェイ URL。
APM_KEYFILE_PSWD	オプション	鍵ファイルのパスワード	鍵ファイルとペアになっている鍵ファイル・パスワード。 **注 :** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。

表 193. *global.environment* ファイル内でサポートされる環境変数 (続き)			
変数名	**重要度**	**値**	**説明**
APM_KEYFILE_URL	オプション	http://<*hosted http server*>:<*port*>/keyfile.p12	鍵ファイルをダウンロードするための URL。 **注 :** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。

config.properties ファイル

680 ページの表 194 は、config.properties 構成ファイル内で設定できる環境変数および関連する説明を示しています。config.properties ファイルは、Python データ・コレクターのインストール・ディレクトリー (例えば、/root/.pyenv/versions/3.5.2/lib/python3.5/site-packages/ibm_python_dc ディレクトリー) にあります。

表 194. *config.properties* ファイル内でサポートされる環境変数			
変数名	**重要度**	**値**	**説明**
KPG_ENABLE_DEEPDIVE	オプション	• False • True	• False: デフォルト値。この変数を False に設定すると、診断データは収集されません。 • True: この変数を True に設定すると、診断データが収集されます。 デフォルト・レベルは True です。 この変数を設定しないと、診断データは収集されません。
KPG_DD_CONFIG_FILE	オプション	診断モニターの構成ファイルのファイル名。	診断モニターの構成ファイルのファイル名。デフォルトのファイル名は、kpg_dd_config.xml です。 **注 :** このファイル内で設定値をカスタマイズした後、ファイルをアプリケーションのルート・ディレクトリーに配置する必要があります。 この変数を設定しないと、データ・コレクター・パッケージ内のデフォルト構成ファイル kpg_dd_config.xml が使用されます。
KPG_DD_APP_PATH	オプション	Python アプリケーションへのパス。	データ・コレクターが診断データを収集する対象の Python アプリケーションまたはモジュールへのパス。モニターする Python アプリケーションおよびモジュールが複数ある場合は、それぞれのパスをセミコロン ; で区切ります。 この変数を設定しない場合、データ・コレクターは、アプリケーションによって使用される要求およびモジュールの診断データを収集します。Python lib 内の要求のデータは収集されません。

表 194. *config.properties* ファイル内でサポートされる環境変数 (続き)

変数名	重要度	値	説明
KPG_DD_SECURITY_FILTER	オプション	• True • False	• True: デフォルト値。この変数を True に設定すると、値 (パスワードなど) は SQL ステートメント内でマスクされ、パラメーターは「**要求コンテキスト**」グループ・ウィジェットに表示されません。 • False: この変数を False に設定すると、値は SQL ステートメント内でマスクされず、パラメーターは「**要求コンテキスト**」グループ・ウィジェットに表示されます。 この変数を設定しないと、値 (パスワードなど) は SQL ステートメント内でマスクされ、パラメーターは「**要求コンテキスト**」グループ・ウィジェットに表示されません。
KPG_GC_STATS	オプション	True	python ガーベッジ・コレクションのすべての統計関数が有効になります。この値を True に設定すると、以下のコマンドを実行した場合と同じ結果が得られます。 `gc.set_debug(gc.DEBUG_STATS | gc.DEBUG_COLLECTABLE | gc.DEBUG_UNCOLLECTABLE | gc.DEBUG_INSTANCES | gc.DEBUG_OBJECTS )` KPG_GC_STATS を無効にするには、この環境変数を削除します。False に設定しないでください。 デフォルト値は True です。 **注 :** 正式な製品環境では KPG_GC_STATS=True を設定しないでください。これはデバッグ・モード専用です。また、アプリケーションに十分なメモリーが割り当てられていることを確認してください。
KPG_LOG_LEVEL	オプション	• DEBUG • ERROR • INFO	• DEBUG: 収集されたデータ、サーバーに送信されたデータ、サーバー応答など、デバッグに役立つ情報のみがログに出力されます。 • ERROR: 例外および予期しないシチュエーションに関する情報のみがログに出力されます。 • INFO: ユーザーがデータ・コレクターの動作を確認するための要約情報がログに出力されます。 デフォルト値は ERROR です。

表 194. *config.properties* ファイル内でサポートされる環境変数 (続き)

変数名	重要度	値	説明
KPG_LOG_TOCONSOLE	オプション	• Y • True • False 以外のすべての値	ログはコンソールに出力され、コマンド **cf logs <appname>** を実行するとログを表示できます。 デフォルト値は True です。
KPG_SAVE_ALL	オプション	True	参照されていないオブジェクトはすべて gc.garbage に保存されるため、1 分ごとに gc.garbage をクリアする必要があります (これはデータ・コレクターが自動的に行います)。この値を True に設定すると、以下のコマンドを実行した場合と同じ結果が得られます。 `gc.set_debug(gc.SAVE_ALL)` KPG_SAVE_ALL を無効にするには、この環境変数を削除します。False に設定しないでください。 デフォルト値は True です。 **注:** 正式な製品環境では KPG_SAVE_ALL=True を設定しないでください。これはデバッグ・モード専用です。また、アプリケーションに十分なメモリーが割り当てられていることを確認してください。
APM_GW_PROXY_CONNECTION	オプション	http://*<server ip* または *hostname>*:port	Python データ・コレクターがモニター・データを送信するために使用する HTTP または HTTPS プロキシー。

オンプレミス・アプリケーション用の Python データ・コレクターの構成解除

Python 環境をモニターする必要がない場合、または Python データ・コレクターをアップグレードする場合は、最初に Python データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. Python アプリケーションのホーム・ディレクトリーに移動します。
2. アプリケーションの requirements.txt から以下の行を削除します。

```
--extra-index-url https://<your_host_name_and_domain>/python_dc/static/python-dc-repos/
simple/
ibm-python-dc
```

3. settings.py ファイルで、MIDDLEWARE_CLASSES セクションから以下の行を削除します。

```
ibm_python_dc.kpg_plugin.ResourceMiddleware
```

4. pip uninstall ibm_python_dc コマンドを実行して、Python ランタイムから Python データ・コレクターをアンインストールします。

タスクの結果

Python データ・コレクターが正常に構成解除されました。

次のタスク

データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

RabbitMQ モニターの構成

Monitoring Agent for RabbitMQ は、RabbitMQ クラスター・リソース (クラスターのノード、キュー、チャネルなど) の正常性およびパフォーマンスをモニターします。RabbitMQ データをエージェントが収集できるように、RabbitMQ エージェントを構成する必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。
- ノードに接続する RabbitMQ ユーザーが読み取り権限を持っていること、およびこのユーザーに対してモニター、管理者、または管理のいずれかのタグが有効になっていることを確認します。
- クラスターのすべてのノードで RabbitMQ 管理プラグインが有効になっていることを確認します。これは、クラスターの 1 つのノードで障害が発生した場合、RabbitMQ エージェントはクラスター内で使用可能なピア・ノードに接続するためです。

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the RabbitMQ エージェント を参照してください。

このタスクについて

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

RabbitMQ エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で始動する必要があります。

- Windows システム 上でエージェントを構成するには、「IBM Cloud Application Performance Management」ウィンドウまたはサイレント 応答ファイルを使用できます。
- Linux システム 上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント 応答ファイルを使用することができます。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for RabbitMQ**」を右クリックしてから「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェントを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェントを再度構成するには、「**再構成**」をクリックします。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。
4. 「**Monitoring Agent for RabbitMQ**」ウィンドウで、構成パラメーターの値を指定し、「**次へ**」をクリックします。

構成パラメーターについては、以下のトピックを参照してください。685 ページの『エージェントの構成パラメーター』

5. 「**Monitoring Agent for RabbitMQ**」インスタンスを右クリックし、「**開始**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド行で、*install_dir*/bin/rabbitmq.sh config *instance_name* コマンドを入力します。*instance_name* は、インスタンスに付ける名前です。
2. 以下のパラメーターの値の指定を求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れるか、値を指定してから Enter キーを押します。
   - IP アドレス
   - ユーザー名
   - パスワード
   - ポート番号
   - Java ホーム
   - Java トレース・レベル

   構成パラメーターについては、以下のトピックを参照してください。685 ページの『エージェントの構成パラメーター』
3. 以下のコマンドを実行して、エージェントを開始します。

```
install_dir/bin/rabbitmq.sh start instance_name
```

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システムおよび Windows システム上で RabbitMQ エージェントを構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. 次のパスにあるサイレント応答ファイルを開きます。
   install_dir¥samples¥rabbitmq_silent_config.txt
2. rabbitmq_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

   構成パラメーターについては、以下のトピックを参照してください。685 ページの『エージェントの構成パラメーター』
3. 応答ファイルを保存し、以下のコマンドを実行します。

   Linux AIX *install_dir*/bin/rabbitmq-agent.sh config *install_dir*/samples/rabbitmq_silent_config.txt

   Windows *install_dir*/bin/rabbitmq-agent.bat config *install_dir*/samples/rabbitmq_silent_config.txt
4. 以下のようにして、エージェントを開始します。

   Linux AIX コマンド *install_dir*¥bin¥rabbitmq-agent.sh start を実行します。

   Windows 「**Monitoring Agent for RabbitMQ**」を右クリックし、「**開始**」をクリックします。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

エージェントの構成パラメーター

RabbitMQ エージェントを構成する際に、インスタンス名や SSL 検証証明書などのパラメーターのデフォルト値を変更できます。

以下の表に、RabbitMQ エージェント の構成パラメーターの詳細な説明を示します。

表 *195. RabbitMQ* エージェントの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
IP アドレス	RabbitMQ アプリケーションがインストールされているノードの IP アドレス。	はい
ユーザー名	RabbitMQ ユーザーのユーザー名。	はい
パスワード	RabbitMQ 管理のユーザー・インターフェースに 接続するためのパスワード。	はい
パスワードの確認入力	「**パスワード**」フィールドに入力したものと同じパスワード。	はい
ポート番号	RabbitMQ 管理プラグインが有効になっているポート番号。デフォルトのポート番号 15672 を使用するか、別のポート番号を指定します。	いいえ
Java ホーム	Java プラグインがインストールされているパス。デフォルトのパス C:¥Program Files¥IBM¥Java50、または Java プラグインがインストールされているディレクトリー・パスを使用します。	いいえ

表 195. RabbitMQ エージェントの構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
Java トレース・レベル	Java プロバイダーのトレース・レベル。有効なトレース・レベル値は、以下のとおりです。 • オフ • ERROR • WARN • INFO • DEBUG_MAX • ALL	いいえ

Response Time Monitoring の構成

Response Time Monitoring エージェントは、HTTP サーバー上の HTTP および HTTPS トランザクションをモニターします。実際のブラウザー・ベースのユーザー・トランザクション (ブラウザーのタイミング) もモニターされます。

Response Time Monitoring エージェントを使用して、以下のレベルのモニター情報を表示できます。

HTTP および HTTPS トランザクション・モニター

Response Time Monitoring エージェントをインストールすると、HTTP トランザクション・モニターは自動的に使用可能になります。

モニターする HTTP サーバーのタイプに応じて、HTTPS トランザクション・モニターは、自動的に使用可能になるか、手動で構成する必要があります。詳しくは、687 ページの『Response Time Monitoring コンポーネント』を参照してください。

Response Time Monitoring は、ユーザー数、セッション数、およびデバイスに関連するデータもモニターします。

データは、「エンド・ユーザー・トランザクション」ダッシュボードにユーザーの現地時間で表示され、「要求数および応答時間」ウィジェットでも使用されます。

実際のエンド・ユーザーのトランザクション・モニター (ブラウザーのタイミング)

モニターする HTTP サーバーのタイプに応じて、ブラウザー・ベースのタイミングは、自動的に使用可能になるか、手動で構成する必要があります。詳しくは、687 ページの『Response Time Monitoring コンポーネント』を参照してください。

ブラウザー・ベースのタイミングを使用可能にするには、JavaScript インジェクションを使用します。

JavaScript インジェクションを使用すると、「エンド・ユーザー・トランザクション」ダッシュボード内に追加のウィジェットと詳細が表示されます。JavaScript インジェクションは、実際のエンド・ユーザーの応答時間をブラウザー内で収集できるようにします。HTTP サーバーによって提供される HTTP ページおよび Web ページの組み込みオブジェクトのパフォーマンスをモニターします。以下の実際のエンド・ユーザーのトランザクション詳細が追加され、使用可能になります。

- 「トランザクション要求数および応答時間」ウィジェット内のクライアント合計時間
- 「トランザクション - 上位 10 件」ウィジェット内のクライアント時間トランザクションの応答時間
- レンダリング時間明細

JavaScript インジェクションを構成する方法については、690 ページの『JavaScript インジェクション』を参照してください。

トランザクション・ダッシュボードの表示

Application Performance ダッシュボードにトランザクション・データを表示します。

トランザクションに関するコンテキスト詳細を報告する多くのウィジェットが Application Performance ダッシュボードに用意されています。

「正常に実行された要求数」は応答時間が 10 秒未満です。「低速要求数」は応答時間が 10 秒を超えています。応答時間が正常か低速かの判別に使用する 10 秒の値は構成できません。以下のウィジェットが使用できます。

- 「下位のパフォーマンス (ユーザー別) - 上位 5 件」グループ・ウィジェット
- 「下位のパフォーマンス (デバイス別) - 上位 5 件」グループ・ウィジェット
- 「要求数および応答時間」グループ・ウィジェット
- 「トランザクション - 上位 10 件」グループ・ウィジェット
- 「トランザクション要求数および応答時間」グループ・ウィジェット
- 「実行先」グループ・ウィジェット
- 「サブトランザクション」グループ・ウィジェット
- 「トランザクション・インスタンス」グループ・ウィジェット
- 「ロケーション別のユーザー」グループ・ウィジェット
- 「選択されたロケーションのユーザー」グループ・ウィジェット
- 「選択されたロケーションのユーザー・セッション - 上位 10 件」グループ・ウィジェット
- 「ユーザー要求および応答時間」グループ・ウィジェット
- 「ユーザー・セッション - 上位 10 件」グループ・ウィジェット
- 「デバイス要求および応答時間」グループ・ウィジェット
- 「セッション」グループ・ウィジェット
- 「セッション要求数」グループ・ウィジェット
- 「セッション・インスタンス」グループ・ウィジェット
- 「トランザクション・インスタンス」グループ・ウィジェット
- ミドルウェア・トランザクション要約ダッシュボード
- ミドルウェア・トランザクション詳細ダッシュボード
- トランザクション・モニターのイベントしきい値
- 対話集約データ
- トランザクション集約データ
- WRT: トランザクションの状況

Response Time Monitoring コンポーネント

Response Time Monitoring エージェントの基本機能は以下のとおりです。

- HTTP トランザクション・モニター
- HTTPS トランザクション・モニター
- ブラウザー・ベースのタイミング (JavaScript インジェクションを使用) のモニター

この機能の詳細な説明を確認するには、686 ページの『Response Time Monitoring の構成』を参照してください。

モニターする HTTP サーバーのタイプに応じて、Response Time Monitoring エージェントの基本機能は、以下のいずれかのコンポーネントを使用して提供されます。

IBM HTTP Server 応答時間モジュール

IBM HTTP Server 応答時間モジュールがモニターできるのは、text/html、application/xml、または application/json の HTTP コンテンツ・タイプのみです。(Javascript インジェクションなし)

IBM HTTP Server 応答時間モジュールは現在、JavaScript インスツルメンテーション圧縮要求をモニターできません。

IBM HTTP Server 応答時間モジュール Javascript インジェクションが現在モニターできるのは、text/html の HTTP コンテンツ・タイプのみです。

Packet Analyzer

Packet Analyzer がモニターできるのは、text/html のコンテンツ・タイプのみです。Packet Analyzer は、gzip 圧縮要求をモニターできます。

AIX または Linux 上で IBM HTTP Server または Apache HTTP Server をモニターする場合は、IBM HTTP Server 応答時間モジュールを使用してください。Packet Analyzer を使用することは可能ですが、お勧めしません。IBM HTTP Server 応答時間モジュールは、Windows ではサポートされていません。Windows 環境では、Packet Analyzer を使用してください。

その他の HTTP サーバーをモニターする場合は、Packet Analyzer を使用してください。Packet Analyzer は、Window、Linux、および AIX でサポートされています。

インストールの計画

ご使用のオペレーティング・システムおよび HTTP サーバーのタイプに基づいて、Response Time Monitoring エージェントのインストールを計画します。

Response Time Monitoring の基本機能は、以下のいずれかのコンポーネントを使用して提供できます。

- Packet Analyzer
- IBM HTTP Server 応答時間モジュール

以下に基づいて、使用するコンポーネントを決定します。

- Response Time Monitoring エージェントがインストールされる HTTP サーバーのタイプ。
- HTTP サーバーがインストールされているオペレーティング・システムの種類。

Response Time Monitoring エージェントを Packet Analyzer とともにインストールするための考慮事項は、以下のとおりです。

- Packet Analyzer は、すべてのオペレーティング・システム (Windows、Linux、および AIX) でサポートされています。
- すべてのオペレーティング・システムで、Packet Analyzer は、ポート 80 で HTTP トランザクションをモニターします。
- HTTPS トランザクション・モニターは、自動的には行われず、手動で構成する必要があります。HTTP サーバーからの SSL トラフィックを暗号化解除できるように、Response Time Monitoring エージェントは SSL 証明書にアクセスする必要があります。詳しくは、707 ページの『HTTPS トランザクションのモニター』を参照してください。
- Packet Analyzer は、すべての HTTP サーバーでサポートされていますが、Sun Java System Web Server および Microsoft Internet Information Services でのみ使用することをお勧めします。
- AIX Linux Windows IBM HTTP Server または Apache HTTP Server 上で Packet Analyzer と連携するように Response Time Monitoring エージェントをインストールするには、HTTP サーバーが停止している必要があります。Response Time Monitoring エージェントをインストールし、HTTP サーバーが停止されていると、Packet Analyzer は自動的に有効になります。
- AIX Linux Windows IBM HTTP Server または Apache HTTP Server 用に Packet Analyzer を構成できますが、お勧めしません。IBM HTTP Server 応答時間モジュールをお勧めします。
- Windows Response Time Monitoring エージェントをインストールする前に、WinPcap 4.1.3 が必要です。
- Windows AIX Linux Sun Java System Web Server または Microsoft Internet Information Services 上に Response Time Monitoring エージェントをインストールする場合、Packet Analyzer は自動的に構成されます。

Response Time Monitoring エージェントを IBM HTTP Server 応答時間モジュールとともにインストールするための考慮事項は、以下のとおりです。

- IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントのコンポーネントの 1 つです。Response Time Monitoring エージェントより前に HTTP Server エージェントをインストールするか、両方を同時にインストールする必要があります。詳しくは、696 ページの『IBM HTTP Server 応答時間モジュール』を参照してください。
- IBM HTTP Server 応答時間モジュール は、すべてのオペレーティング・システム (Windows、Linux、および AIX) でサポートされています。IBM HTTP Server 応答時間モジュールでは、IBM HTTP Server バージョン 7、8、および 9 がサポートされます。
- Response Time Monitoring エージェントと HTTP Server エージェントを同じマシンにインストールしてください。
- AIX、Linux、および Windows で、IBM HTTP Server 応答時間モジュールは、すべてのポートで HTTP および HTTPS 要求をモニターします。
- IBM HTTP Server 応答時間モジュールは、IBM HTTP Server または Apache HTTP Server でのみサポートされています。
- エージェントは両方とも自動的に開始されますが、HTTP サーバーを再始動する必要があります。

以下の表では、Response Time Monitoring エージェントを自動的に構成するためのさまざまな組み合わせについて説明します。

表 *196. Response Time Monitoring* エージェントを自動的に構成するためのシナリオ

HTTP サーバーと OS の組み合わせ	**Packet Analyzer**	**IBM HTTP Server 応答時間モジュール**
AIX または Linux 上の Sun Java System Web Server または Microsoft Internet Information Services	自動	サポートされない
Windows 上の Sun Java System Web Server または Microsoft Internet Information Services	自動	サポートされない
AIX、Linux、または Windows 上の IBM HTTP Server または Apache HTTP Server	HTTP サーバーが停止されている場合は自動。	HTTP サーバーが存在し、構成されている場合は自動。

構成の計画

Response Time Monitoring エージェントをインストールすると、HTTP モニターは自動的に有効になります。ご使用の環境によっては、HTTPS および JavaScript インジェクションを手動で構成する必要があります。

HTTP モニター
インストールのガイドラインに従った場合、Packet Analyzer および IBM HTTP Server 応答時間モジュール用の HTTP トランザクション・モニターは、自動的に構成されます。688 ページの『インストールの計画』を参照してください。

HTTPS モニター
インストールのガイドラインに従った場合、IBM HTTP Server 応答時間モジュール用の HTTPS トランザクション・モニターは、自動的に構成されます。688 ページの『インストールの計画』を参照してください。

Packet Analyzer 用の HTTPS トランザクション・モニターは、手動で構成する必要があります。詳しくは、705 ページの『 Packet Analyzer のロードマップ』を参照してください。

ブラウザー・ベースのタイミング (JavaScript インジェクションを使用)
IBM HTTP Server 応答時間モジュール用のブラウザー・ベースのタイミング (JavaScript インジェクションを使用) は、自動的に構成されます。

Packet Analyzer 用のブラウザー・ベースのタイミング (JavaScript インジェクションを使用) は、手動で構成する必要があります。詳しくは、705 ページの『 Packet Analyzer のロードマップ』を参照してください。

以下の表では、各コンポーネントの基本機能がどのように構成されるかについて説明します。

表 *197.* 基本機能の構成

	Sun Java System Web Server または Microsoft Internet Information Services (Windows、Linux、または AIX) 上の Packet Analyzer	**IBM HTTP Server または Apache HTTP Server (Windows、Linux、または AIX) 上の IBM HTTP Server 応答時間モジュール**
HTTP トランザクション・モニター	自動的に有効になります。	自動的に有効になります。
HTTPS トランザクション・モニター	手動で構成する必要があります。	自動的に有効になります。
JavaScript インスツルメンテーションを使用した実際のエンド・ユーザーのトランザクション・モニター (ブラウザーのタイミング)	手動で構成する必要があります。	自動的に有効になります。

JavaScript インジェクション

Response Time Monitoring エージェントで収集してエンド・ユーザー・トランザクション・ダッシュボードに表示するデータをカスタマイズできます。

Web ベース・アプリケーションの良好なユーザー・エクスペリエンスを確実に提供するには、実際のユーザーによって認識されるパフォーマンスをモニターする必要があります。これは、ブラウザー・レベルでモニターすることを意味します。

ブラウザー・レベルでモニターできるようにするには、モニターするページに JavaScript モニター・コードを注入する必要があります。このコードは、特定のブラウザーのタイミングに関するデータを収集します。

これは、モニターする Web ページおよびオブジェクト内で JavaScript インジェクションを使用して行われます。Response Time Monitoring エージェントをインストールした HTTP サーバーのタイプに応じて、以下の 2 つの方法を使用して、実際のエンド・ユーザーのトランザクション応答時間情報を収集できます。

- IBM HTTP Server または Apache HTTP Server を使用する場合は、IBM HTTP Server 応答時間モジュールを使用します。IBM HTTP Server 応答時間モジュールは、JavaScript インジェクションを自動的に実行します。IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントのコンポーネントの 1 つです。これは、HTTP Server エージェントの一部としてインストールおよび構成されます。詳しくは、696 ページの『IBM HTTP Server 応答時間モジュール』を参照してください。
- サポートされる他の HTTP サーバーを使用する場合は、Packet Analyzer を使用します。Packet Analyzer の場合は、ブラウザーのタイミングを収集するように Web ページを手動で設定する必要があります。詳しくは、705 ページの『アプリケーションへの JavaScript モニター・コンポーネントの追加』を参照してください。

以下の表は、Packet Analyzer または IBM HTTP Server 応答時間モジュール用に環境を構成する場合に Application Performance ダッシュボードで使用可能な機能を示しています。

	Packet Analyzer	**IBM HTTP Server 応答時間モジュール**
トランザクション - 上位 10 件	✔	✔

	Packet Analyzer	IBM HTTP Server 応答時間モジュール
サーバー時間	✓	✓
レンダリング時間明細	—	✓
AJAX サブトランザクション	✓	✓
「サブトランザクション」表のリソース時間データ	—	✓
トランザクション・インスタンス (上位 10 件)	✓	✓
トランザクション・インスタンス・トポロジー	✓	✓
アプリケーション・トポロジー	✓	✓
JavaScript インジェクションの自動インスツルメンテーション	N/A	✓

Windows での Response Time Monitoring の再構成

rt-agent 対話式構成コマンドまたは「IBM Cloud Application Performance Management」ユーティリティーを使用して、エージェントを構成または再構成します。

始める前に

HTTPS トランザクション・モニターを有効にする場合は、同じマシン上に Monitoring Agent for HTTP Server がインストールされていないことを確認してください。 インストールされている場合、Response Time Monitoring 構成が、Packet Analyzer 用の HTTPS 設定の変更を行いません。

このタスクについて

Response Time Monitoring エージェントは、インストール後に自動的に構成されます。インストールのガイドラインについては、688 ページの『インストールの計画』に従ってください。異なるポートをモニターする場合や HTTPS トランザクションをモニターする場合などに、再構成することが必要になることがあります。

インストール・ディレクトリーは、*install_dir* で参照されます。デフォルトのインストール・ディレクトリーは、C:¥IBM¥APM¥ です。

rt-agent 対話式構成コマンドを使用する代わりに、IBM Cloud Application Performance Management ユーティリティーでエージェントを 構成することができます。詳しくは、186 ページの『Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用』を参照してください。

手順

データ設定をカスタマイズするには、以下の手順を実行します。

1. Response Time Monitoring エージェントがインストールされているコンピューターで、次のようにこのエージェントを停止します。

   install_dir¥BIN¥rt-agent.bat stop

2. サイレント構成を使用して、エージェントを構成します。

   ```
   install_dir¥BIN¥rt-agent.bat config install_dir¥samples¥rt_silent_config.txt
   ```

HTTPS トランザクション・モニターを有効にする場合は、サイレント構成ファイル内の次の行のコメントを外します。Windows 上で HTTPS をモニターするように Response Time Monitoring エージェントを構成する `rt_silent_config.txt` のサンプルは次のようになります。

```
# Monitor HTTPS transactions
KT5MONITORHTTPSAPP=YES

# HTTPS keystore (e.g. - /tmp/keys.kdb)
KT5KEYSTORE=C:¥keys¥key.kdb

# HTTPS server certificate map (eg - certAlias,9.48.152.1,443;...)
KT5SERVERMAP=certalias,9.48.152.1,443

# Monitor network traffic for the NIC hosts this IP address
#KT5MONITORIP=9.48.152.1
```

3. Response Time Monitoring エージェントを再始動し、変更を有効化します。
   install_dir¥BIN¥rt-agent.bat start

タスクの結果
Response Time Monitoring に関連付けられたダッシュボードに、新規ソースのデータが表示されます。

AIX および Linux での Response Time Monitoring の再構成

`rt-agent` 構成コマンドを使用して、Response Time Monitoring エージェントを構成または再構成します。

このタスクについて

Response Time Monitoring エージェントは、インストール後に自動的に構成されます。インストールのガイドラインについては、688 ページの『インストールの計画』に従ってください。別のポートをモニターする場合などに、再構成する必要が生じることがあります。

インストール・ディレクトリーは、*install_dir* として表されます。デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。

エージェントのインストールに使用したのと同じ root ユーザーを使用して、エージェントを始動、停止、および構成します。

手順

再構成するには、以下のステップを実行します。

1. Response Time Monitoring エージェントのインストール先のコンピューター上で、次のようにしてエージェントを停止します。

   ```
   install_dir/bin/rt-agent.sh stop
   ```

2. 対話式の構成またはサイレント構成のいずれかを使用します。

   a) 対話式の構成:

   ```
   install_dir/bin/rt-agent.sh config
   ```

   b) サイレント構成:

   ```
   install_dir/bin/rt-agent.sh config  install_dir/samples/rt_silent_config.txt
   ```

   HTTPS トランザクション・モニターを有効にする場合は、サイレント構成ファイル内の次の行のコメントを外します。AIX および Linux 上で HTTPS をモニターするように Response Time Monitoring エージェントを構成する `rt_silent_config.txt` のサンプルは次のようになります。

   ```
   # Monitor HTTPS transactions
   KT5MONITORHTTPSAPP=YES

   # HTTPS keystore (e.g. - /tmp/keys.kdb)
   KT5KEYSTORE=/tmp/keys.kdb
   ```

```
# HTTPS server certificate map (eg - certAlias,9.48.152.1,443;...)
KT5SERVERMAP=certalias,9.48.152.1,443

# Monitor network traffic for the NIC hosts this IP address
#KT5MONITORIP=9.48.152.1
```

3. Response Time Monitoring エージェントを再始動し、変更を有効化します。
   install_dir/bin/rt-agent.sh start

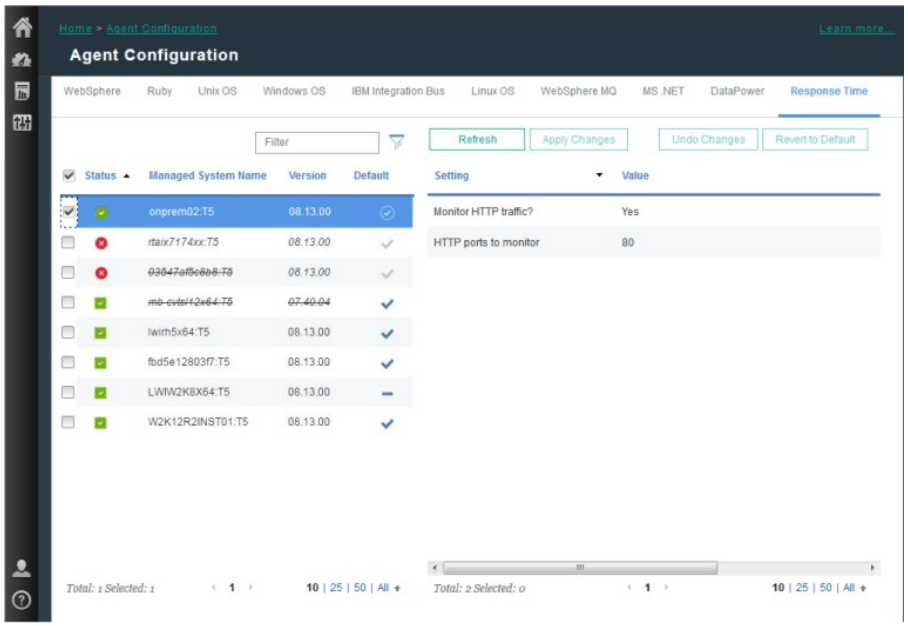
タスクの結果
Response Time Monitoring に関連付けられたダッシュボードに、新規ソースのデータが表示されます。

「エージェント構成」ページを使用した構成

Cloud APM コンソールの「**エージェント構成**」ページを使用して、インストールされているエージェントを確認できます。必要に応じて、HTTP トランザクションのモニターを無効または有効にしたり、Response Time Monitoring エージェントでモニターされるポートを設定したりすることができます。

エージェント構成

「Response Time Monitoring 「**エージェント構成**」ページにアクセスするには、Cloud APM コンソール で、「**システム構成**」 > 「**エージェント構成**」を選択してから、「**応答時間**」タブを選択します。



「**エージェント構成**」ページには、Response Time Monitoring がインストールされている環境内のシステムがリストされています。

Response Time Monitoring エージェントがインストールされている各システムについて、「**エージェント構成**」ページに以下の情報が表示されます。

- システムがオンライン (背景が緑のチェック・マーク) またはオフライン (背景が赤のバツ印) のいずれであるか。
- インストールされている Response Time Monitoring エージェントのバージョン。
- 中央構成でエージェントのタイプ (つまり、HTTP トランザクションをモニターするために Packet Analyzer または IBM HTTP Server 応答時間モジュールのどちらが使用されているかどうか) を判別できない場合、エージェントだけに取り消し線が付けられます。一般に、エージェントが ASF アクティビティーによってエージェントの詳細を送信していなければ、エージェントのタイプを判別できません。
- システムでデフォルトの 構成値が使用されるか、あるいはカスタム値が設定されているか。
- Response Time Monitoring エージェントが Packet Analyzer を使用して HTTP トランザクションをモニターしている場合のモニター対象ポート。

Setting	Value
Monitor HTTP traffic?	Yes
HTTP ports to monitor	80

- HTTP トランザクションのモニターに IBM HTTP Server 応答時間モジュールが HTTP Server エージェントとともに使用されているかどうか。

Setting	Value
Is IBM HTTP Server Response Time module enabled?	*Yes*

ヒント：IBM HTTP Server 応答時間モジュールは、HTTP トランザクションと HTTPS トランザクションを自動的にモニターします。Response Time Monitoring エージェントをさらに構成する必要はありません。

構成設定を表示するエージェントを選択します。特定のエージェントを検索するには、そのエージェントがインストールされているシステムの名前の一部または名前全体を「**フィルター**」フィールドに入力します。

「**エージェント構成**」ページで行われたカスタマイズは、他のカスタマイズ内容やデフォルト値よりも優先されます。

設定を変更した後に変更を取り消す場合は、「**変更の取り消し**」をクリックして最後に保存した設定に戻すか、「**デフォルトに戻す**」をクリックしてデフォルト値に戻します。

新規構成値が Central Configuration Services に送信され、オンラインのエージェントは自動的に再構成されます。再始動する必要はありません。 エージェントがオフラインである場合は、オンラインになったときに新規構成設定がダウンロードされます。 データが最新表示されたときに、Response Time Monitoring に関連付けられたダッシュボードに新規ポートのデータが表示されます。

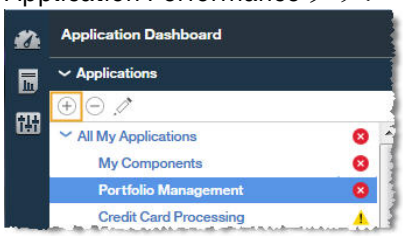
アプリケーションの追加

Response Time Monitoring エージェントをインストールしたら、モニター対象のアプリケーションを Application Performance ダッシュボードに追加しなければならない場合があります。

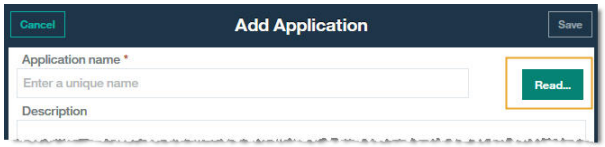
手順

Application Performance ダッシュボードにアプリケーションを追加するには、以下のようにします。

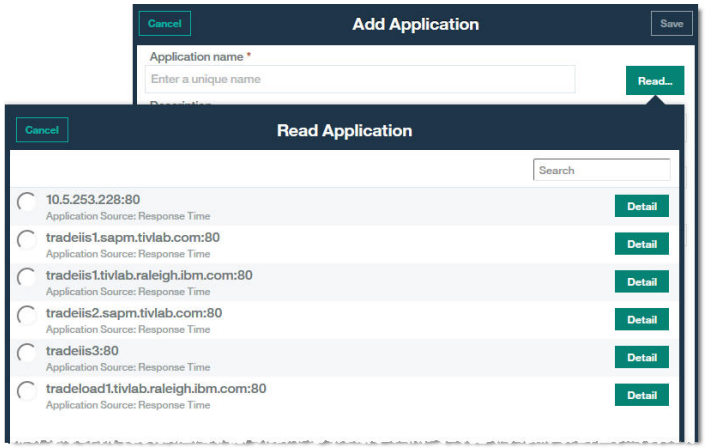
1. Application Performance ダッシュボードで、「**アプリケーションの追加**」をクリックします。



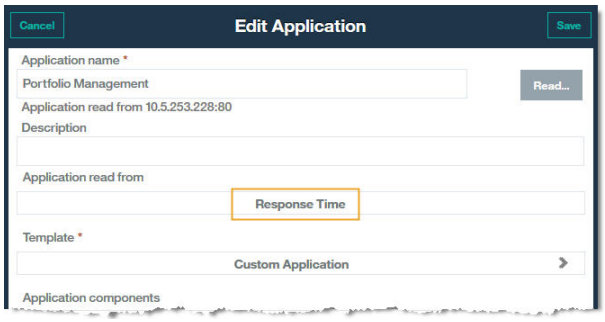
2. 「**読み取り**」を選択して、検出されたアプリケーションのリストを開きます。



3. モニターしたいアプリケーションを選択します。



「**アプリケーションの読み取り元**」フィールドに「**応答時間**」がソース・リポジトリーとして表示され、「**アプリケーション・コンポーネント**」にすべてのコンポーネントがリストされます。



4. Application Performance ダッシュボードの Response Time Monitoring エージェントでモニターされるアプリケーションを表示する場合、追加の構成は必要ありません。「**アプリケーションの追加**」ウィンドウで、「**保存**」をクリックします。

タスクの結果

Response Time Monitoring エージェントで検出されたアプリケーションが、Application Performance ダッシュボードの「**すべてのマイ・アプリケーション**」にリストされます。

IBM HTTP Server 応答時間モジュールの構成

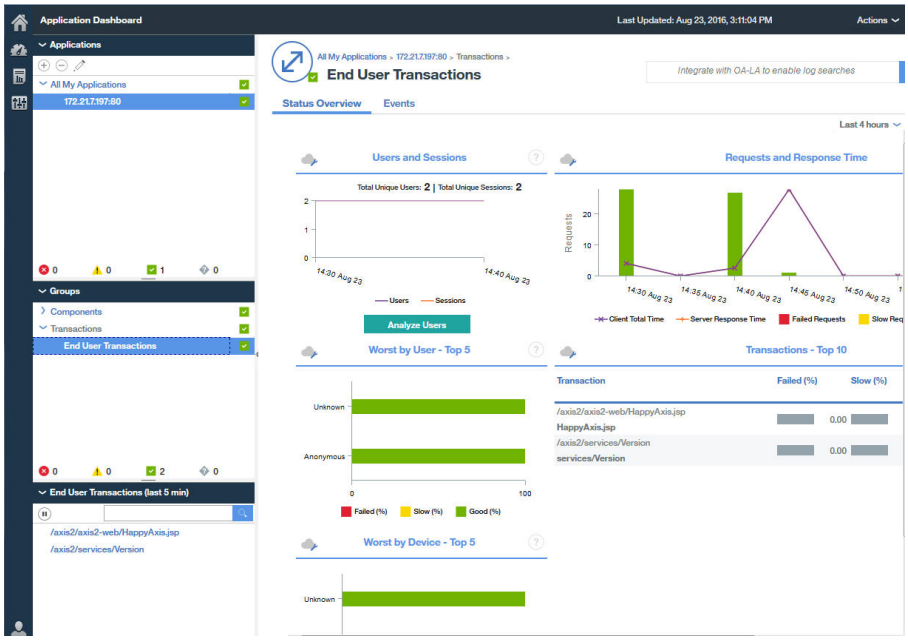
IBM HTTP Server および Apache HTTP Server の場合は、IBM HTTP Server 応答時間モジュールを使用して、HTTP ページの実際のエンド・ユーザーの Response Time Monitoring メトリックを表示します。

IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントの一部としてインストールおよび構成されます。IBM HTTP Server 応答時間モジュールは、AIX、Linux および Windows での IBM HTTP Server および Apache HTTP Server とのみ連携します。IBM HTTP Server 応答時間モジュールは、すべてのポートで HTTP および HTTPS 要求をモニターします。

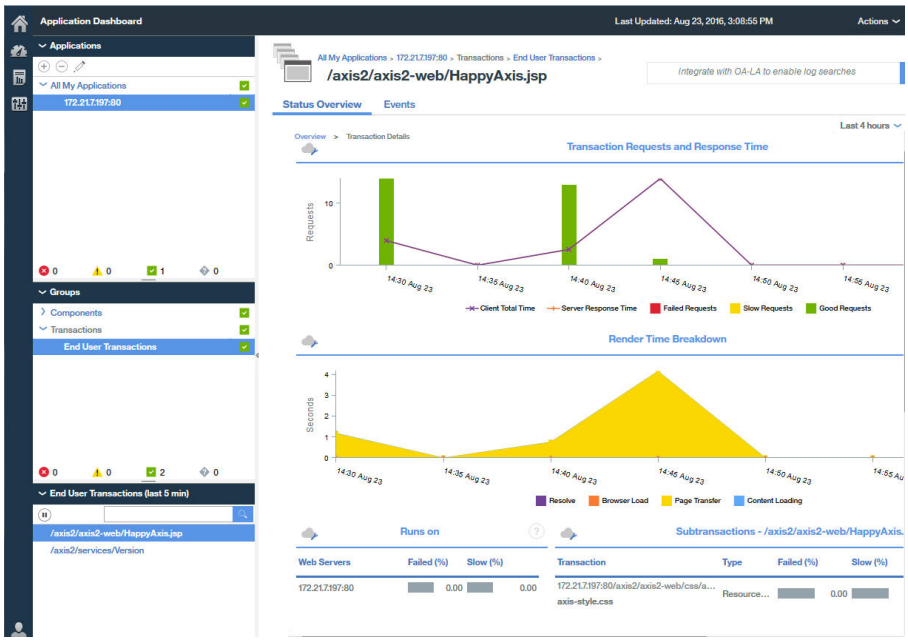
IBM HTTP Server 応答時間モジュールは、JavaScript を使用することで、IBM HTTP Server によって提供される Web ページにヘッダーを挿入し、Response Time Monitoring エージェントがそれらのページをモニターできるようにします。ページによってロードされる組み込みオブジェクトは、Cookie を使用して追跡されます。その後、IBM HTTP Server または Apache によって提供される Web ページから取得したトランザクション情報がエンド・ユーザー・トランザクションのダッシュボードに表示されます。

例えば、次のようにします。

「エンド・ユーザー・トランザクション」ワークスペースの「トランザクション - 上位 10 件」に、IBM HTTP Server 応答時間モジュールから収集したデータが表示されます。



「エンド・ユーザー・トランザクション」ワークスペースの「**サブトランザクション**」表に、IBM HTTP Server 応答時間モジュールから収集したデータが表示されます。



IBM HTTP Server 応答時間モジュール

IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントの一部です。ただし、サポートされる HTTP サーバー上のアプリケーション・トランザクションをモニターするために、Response Time Monitoring エージェントと連携します。

IBM HTTP Server 応答時間モジュールと連携するように Response Time Monitoring エージェントをインストールすると、このモジュールは、すべてのポートで HTTP および HTTPS 要求をモニターします。

IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントの一部です。Response Time Monitoring エージェントより前に HTTP Server エージェントをインストールするか、両方を同時にインストールする必要があります。

HTTP Server エージェントは、以下の 2 つのプラグインで構成されています。

1. `khu_module` - これは HTTP Server エージェントです。このプラグインは、HTTP Server エージェントに関連付けられているすべてのダッシュボードの処理を行います。詳しくは、「HTTP Server エージェント Reference」を参照してください。

2. `wrt_module` - これは IBM HTTP Server 応答時間モジュールです。

これらの 2 つのプラグインは、HTTP Server エージェント構成ファイルの名前によって示されます。Apache HTTP Server 用の HTTP Server エージェント構成ファイルは、以下のとおりです。

```
khu.usr.local.apache24.conf.httpd.conf
```

IBM HTTP Server 用のファイルは、以下のとおりです。

```
khu.opt.IBM.HTTPServer.conf.httpd.conf
```

このファイルの命名規則は、以下のとおりです。khu.(HTTP サーバー構成ファイルの絶対パス、/ を . に変更).conf

```
LoadModule khu_module

LoadModule wrt_module
```

IBM HTTP Server 応答時間モジュールが機能するためには、HTTP サーバー構成ファイルには、HTTP Server エージェント構成ファイルを参照する include ステートメントが必要です。例:

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
```

この include ステートメントは、両方のプラグインを同時に有効にします。詳しくは、275 ページの『HTTP Server モニターの構成』を参照してください。

IBM HTTP Server 応答時間モジュールのインストールおよび構成

構成は、Response Time Monitoring エージェント側では自動的に行われます。IBM HTTP Server 応答時間モジュールは、HTTP Server エージェントの一部としてインストールおよび構成する必要があります。Response Time Monitoring エージェントは、IBM HTTP Server 応答時間モジュールを自動的に検出し、有効にします。

このタスクについて

手順

1. HTTP Server エージェント をインストールします。これにより、IBM HTTP Server 応答時間モジュールが自動的にインストールされます。
2. HTTP Server エージェントを構成します。これにより、IBM HTTP Server 応答時間モジュールが有効になります。詳しくは、275 ページの『HTTP Server モニターの構成』を参照してください。
3. ご使用のオペレーティング・システムに応じて、Response Time Monitoring エージェント エージェントを `root` または管理者としてインストールします。説明については、123 ページの『第 6 章 エージェントのインストール』を参照してください。
4. IBM HTTP Server を再始動します。Response Time Monitoring インストーラーが HTTP Server エージェントを検出すると、Response Time Monitoring エージェントは IBM HTTP Server 応答時間モジュールを自動的に有効にします。

IBM HTTP Server 応答時間モジュールの手動での有効化

IBM HTTP Server によって提供される HTTP ページおよび Web ページの組み込みオブジェクトのパフォーマンスをモニターするように IBM HTTP Server 応答時間モジュールを手動で有効化できます。

このタスクについて

HTTP Server エージェントをインストールして構成すると、IBM HTTP Server 応答時間モジュールが自動的に有効になります。ただし、IBM HTTP Server 応答時間モジュールを手動で有効にすることもできます。

手順

Linux、AIX、または Windows で IBM HTTP Server 応答時間モジュールを手動で使用可能にするには、以下の手順を実行します。

1. Linux AIX
   Linux または AIX で IBM HTTP Server 応答時間モジュールを手動で使用可能にするには、以下の手順を実行します。
   a) Response Time Monitoring エージェントを停止します。
      以下を実行します。

      ```
      $AGENT_HOME/bin/rt-agent.sh stop
      ```

      ここで、*$AGENT_HOME* は、例えば /opt/ibm/apm/agent (Linux システムの場合) や /opt/ibm/ccm/agent (AIX システムの場合) です。
   b) 構成コマンドを実行し、*$AGENT_HOME*/samples/rt_silent_config_ihs.txt を使用してロード・モジュールを Web サーバー構成ファイルに追加し、IBM HTTP Server 応答時間モジュールの構成パラメーターを設定します。
   c) Response Time Monitoring エージェントを再始動します。
2. Windows
   Windows で IBM HTTP Server 応答時間モジュールを手動で使用可能にするには、以下の手順を実行します。
   a) Response Time Monitoring エージェントを停止します。
      以下を実行します。

      ```
      AGENT_HOME/bin/rt-agent.bat stop
      ```

      ここで、*AGENT_HOME* は、例えば Windows システムの C:¥IBM¥APM です。
   b) 構成コマンドを実行し、*AGENT_HOME*¥samples¥rt_silent_config_ihs.txt を使用してロード・モジュールを Web サーバー構成ファイルに追加し、IBM HTTP Server 応答時間モジュールの構成パラメーターを設定します。
      例えば、次の場合があります。

      ```
      AGENT_HOME¥bin¥rt_agent.bat config AGENT_HOME¥samples¥rt_silent_config_ihs.txt
      ```

   c) Response Time Monitoring エージェントを再始動します。

IBM HTTP Server 応答時間モジュールの手動での無効化

IBM HTTP Server 応答時間モジュールを無効にして再び Packet Analyzer を使用するには、エージェントを再構成し、IBM HTTP Server 応答時間モジュール でモニターをオフにします。

このタスクについて

以下の手順を使用して IBM HTTP Server 応答時間モジュールを無効にします。

手順

Linux、AIX、または Windows でエージェントを対話式に再構成するには、以下の手順を実行します。

1. Linux AIX
   Linux および AIX でエージェントを対話式に再構成するには、以下の手順を実行します。
   a) *install_dir*/bin/rt-agent.sh config を実行します。
      ここで、*install_dir* は、Linux および AIX 上の /opt/ibm/apm/agent です。
   b) Response Time Monitoring エージェントを再始動します。

   あるいは、パラメーターを手動で設定する手順:

   a) テキスト・エディターで *install_dir*/config/*hostname*_t5.cfg を開きます。

ここで、*install_dir* は、Linux および AIX では `/opt/ibm/apm/agent` です。

b) 以下のパラメーターを設定します。

```
KT5DISABLEANALYZER=NO
KT5ENABLEWEBPLUGIN=NO
```

c) Response Time Monitoring エージェントを再始動します。

2. Windows

Windows でエージェントを対話式に再構成するには、以下の手順を実行します。

a) 関連パラメーターを手動で設定するために、テキスト・エディターで *install_dir* ¥TMAITM6_x64¥*hostname*_t5.cfg を開きます。

ここで、*install_dir* は Windows 上の `C:¥IBM¥APM` です。

b) 以下のパラメーターを設定します。

```
KT5DISABLEANALYZER=NO
KT5ENABLEWEBPLUGIN=NO
```

c) Response Time Monitoring エージェントを再始動します。

IBM HTTP Server 応答時間モジュールの拡張構成

IBM HTTP Server 応答時間モジュール用の多数の拡張構成オプションがあります。

IBM HTTP Server 応答時間モジュールは自動的に構成されますが、多数の拡張構成タスクを実行して、パフォーマンスと機能を微調整できます。

リソース時間モニターの無効化

HTTP Server エージェントによってインストールされるすべての IBM HTTP Server 応答時間モジュール・インスタンスに対して、リソース時間モニターが有効になります。

このタスクについて

IBM HTTP Server 応答時間モジュールでモニターされるリソースの数を減らす場合、あるいはリソース時間モニターを無効にして特定の IBM HTTP Server のモニターに要する処理負荷を削減する場合は、以下の手順を実行します。

手順

生成された HTTP Server エージェント構成ファイルを編集するには、以下の手順を実行します。

1. HTTP サーバー構成ファイル (httpd.conf) の末尾に以下を追加します。

```
WrtMaxPostResourcesSize
```

2. 以下のいずれかの値を設定します。
   - `WrtMaxPostResourcesSize -1` (すべてのリソースをモニターする場合)
   - `WrtMaxPostResourcesSize 0` (リソースのモニターをオフにする場合)
   - `WrtMaxPostResourcesSize` *n* (特定の数のリソースをモニターする場合。デフォルトは 10)。例えば、`WrtMaxPostResourcesSize 2` に設定すると、最大で 2 つのリソースがサーバーに送信されるように設定されます。
3. HTTP サーバーを再始動します。

ARM 相関関係子の生成の無効化

ARM 相関関係子の生成はデフォルトで有効になっています。これにより、IBM HTTP Server 応答時間モジュールがトポロジー内の任意のバックエンド・サーバーにリンクすることができます。 必要に応じて、ARM 相関関係子の生成を無効にすることができます。

このタスクについて

制約事項 : ARM 相関関係子の生成を無効にすると、IBM HTTP Server 応答時間モジュールがバックエンド・サーバー (WebSphere Application Server など) にリンクできません。ARM 相関関係子の生成を無効にするのは、IBM ソフトウェア・サポートから指示があった場合のみにしてください。

手順

ARM 相関関係子の生成を無効にするには、以下の手順を実行します。

1. HTTP サーバー構成ファイル (httpd.conf) の末尾に以下を追加します。

```
WrtDisableArmCorr
```

2. HTTP サーバーを再始動します。

Response Time Monitoring with Client Time (JavaScript インスツルメンテーション*)* の無効化

IBM Application Performance Management では、すべてのコンピューターで稼働するすべての IBM HTTP Server インストール済み環境に対して Response Time Monitoring with Client Time (JavaScript インスツルメンテーション) が有効になります。

このタスクについて

手順

JavaScript インジェクションを手動で無効にするには、以下の手順を実行します。

1. *HTTP_Server_root*/conf/httpd.conf にある HTTP Server 構成ファイルを開きます。
2. HTTP Server エージェント用に追加された行に移動し、その後に次の行を追加します。

```
WrtDisableJSI
```

   例えば、次の場合があります。

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
WrtDisableJSI
```

3. httpd.conf ファイルを保存し、HTTP Server をリサイクルします。

WRTCorrelator Cookie のバイパス

WebSphere Portal の Response Time Monitoring エージェントで JavaScript インジェクションが有効になっていると、WebSphere Portal の使用中に WRTCorrelator Cookie などの Cookie で問題が発生する可能性があります。この問題を防ぐために、WRTCorrelator Cookie を無視するように設定できます。

手順

1. 必要な場合は WebSphere_Portal サーバーを始動します。
2. WebSphere® Integrated Solutions Console にログインします。
3. 「**リソース**」 > 「**リソース環境**」 > 「**リソース環境プロバイダー**」に移動します。
4. 「**WP ConfigService**」を選択します。
5. 「追加プロパティー」の下の「**カスタム・プロパティー**」を選択します。
6. 「**新規**」をクリックします。
7. プロパティーの名前 (この場合は WRTCorrelator Cookie) を指定し、無視するプロパティーの値を設定します。

   WRTCorrelator Cookie を無視するように設定するには、以下のように入力します。

```
cookie.ignore.regex =
digest¥.ignore.*|LTPAToken|LTPAToken2|JSESSIONID|WASReqURL|WRTCorrelator
|PD_STATEFUL.*
```

8. 「**適用**」をクリックして、変更内容を保存します。
9. WebSphere® Integrated Solutions Console からログアウトします。

タスクの結果
WRTCorrelator Cookie が無視されるように設定され、ループなどの問題が防止されます。

クライアント時間レポートからのページの除外
ブラウザーで、クライアント時間レポートから特定のページを除外することが必要な場合があります。

このタスクについて
パラメーターを構成ファイルに追加することで、指定するパターンと一致するファイルへの、IBM HTTP Server 応答時間モジュール による JavaScript インジェクションを停止できます。これにより、それらのページのブラウザーによるクライアント時間レポートが停止されます。

手順

ブラウザーで、クライアント時間レポートから特定のページを除外するには、次の手順を実行します。

1. テキスト・エディターで以下のファイルを開きます。

   Linux AIX *install_dir*/config/*hostname*_t5.cfg where *install_dir* is /opt/ibm/apm/agent

   Windows *install_dir*¥TMAITM6_x64¥*hostname*_t5.cfg (*install_dir* は C:¥IBM¥APM)

2. **advconfig** セクションに以下を追加します。

   ```
   {KT5WEBPLUGIN_JSI_EXCLUDE_URI_WITH_PATTERNS=URL_path_pattern to_exclude}
   ```

   例えば、次の場合があります。

   ```
   {KT5WEBPLUGIN_JSI_EXCLUDE_URI_WITH_PATTERNS=*/DoNotJSIMe.jsp,/absolutePath/index.jsp,/
   skipThisDir/*}
   ```

   URL パスは 256 文字に制限されています。

   ヒント: 突き合せパターンのプレフィックスまたはサフィックスとしてアスタリスク (*) を使用します。必要に応じて、パラメーターにコンマで区切った複数の値を追加します。

3. *hostname*_t5.cfg を保存して閉じます。
4. 以下のようにして Response Time Monitoring エージェントを再始動します。

   ```
   rt-agent.sh stop
   rt-agent.sh start
   ```

***root* 以外のユーザーとしての *IBM HTTP Server* 応答時間モジュール の使用**
慎重なセットアップでは、root 以外のユーザー ID を使用して IBM HTTP Server 応答時間モジュールを使用できます。ただし、Network Packet Analyzer を使用している場合は、root ユーザーを使用する必要があります。

root 以外のユーザー ID を使用する場合は、以下のガイドラインに従ってください。

Response Time Monitoring エージェントの場合

実行する際に使用するユーザー ID を使用して、Response Time Monitoring エージェントを、書き込み権限があるディレクトリーにインストールします。

- この手順では、Response Time Monitoring エージェントのユーザー ID は *agentuser* です。エージェントがインストールされるディレクトリーは *$AGENT_HOME* です。

- root として Response Time Monitoring エージェントをインストールしてから別のユーザーとしてそのエージェントを実行した場合、ユーザーはファイルを作成できません。

IBM HTTP Server 応答時間モジュールの場合

ServerRoot は、apache start|stop の実行で使用するユーザー ID と同じユーザー ID によって所有される必要があります。

- apache start の処理中に、wrt トラッキング・ディレクトリーが ServerRoot の下に作成されます。そのため、このユーザーには、ServerRoot の下にファイルとディレクトリーを作成するための十分な権限が必要です。
- IBM HTTP Server 応答時間モジュールのユーザーである *ihsuser* が *agentuser* と異なる場合、そのユーザーには $AGENT_HOME/tmp への書き込み権限が必要です。

  エージェントのインストール中に $AGENT_HOME/tmp が作成されます。*ihsuser* には、$AGENT_HOME/tmp に kt5 ディレクトリーを作成するための権限が必要です。
- ServerRoot の複数のバージョンがそれぞれ別のユーザーによって管理されている場合があります。

Response Time Monitoring エージェントおよび IBM HTTP Server 応答時間モジュールの両方について

agentuser と *ihsuser* のいずれにも、以下のディレクトリーに対する読み取り/書き込み権限が必要です。

- $AGENT_HOME/tmp/kt5
- ServerRoot/wrt

通常、まず IBM HTTP Server 応答時間モジュールが開始され、Response Time Monitoring が最初に開始されるときに、すべてに対する読み取り/書き込みアクセス権限を使用して、Response Time Monitoring によって wrt ディレクトリーが自動的に作成されます。

また、構成をプッシュするために ServerRoot/wrt も camconfig によって使用されます。 *ihsuser* は、*agentuser* によって取得される共有キュー ID ファイルを作成します。*agentuser* はディレクトリーからキュー ID を読み取り、構成をプッシュします。

最初に root を使用して apache start を実行したが、*ihsuser* が root でない場合、以下のステップを実行します。

1. Response Time Monitoring エージェントを停止します。
2. root を使用して、apachectl stop を実行します。
3. 以下のディレクトリーを削除します。
   - $AGENT_HOME/tmp/kt5
   - ServerRoot/wrt
4. *ihsuser* を使用して、apachectl start を実行し、正しい権限でディレクトリーを再作成します。

wrt ディレクトリーに対する IBM HTTP Server 応答時間モジュール権限のリセット

セキュリティーを強化するために、IBM HTTP Server 応答時間モジュールの wrt ディレクトリーの権限をリセットできます。この更新では、構成ファイルにパラメーターを追加して、IBM HTTP Server のインストール中に非 root ユーザーによって作成された wrt ディレクトリーに対する権限を制限します。

IBM HTTP Server 応答時間モジュールのインストール先システムで以下の手順を実行し、非 root ユーザーに対する wrt ディレクトリーのアクセス権を 777 から 700 に変更します。

1. テキスト・エディターで IHS_HOME/conf/httpd.conf ファイルを開きます。
2. 以下のように、プロパティー **WrtDisableDirPermNonRoot** をファイルの末尾に追加します。
   - このプロパティーが有効な場合は、wrt ディレクトリー (権限 700) の作成には、httpd を開始し、かつディレクトリーを作成したユーザー ID に一致するユーザー ID のみ使用されます。他のユーザーの場合は、このディレクトリーを操作するためのアクセスが拒否されます。

- このプロパティーが有効でない場合は、デフォルトの権限 777 で wrt ディレクトリーが作成されます。

3. 以下のようにして Response Time Monitoring エージェントを再始動します。

```
rt-agent.sh stop
rt-agent.sh.start
```

Response Time Monitoring の構成データおよび一部の永続ファイルは wrt ディレクトリーに格納されます。このディレクトリーは通信プロセス中に使用され、各接続プロセスではこのディレクトリーに wrt 構成ファイルが作成されます。**WrtDisableDirPermNonRoot** を httpd.conf ファイルに追加した後は、限られた特定のユーザーのみが正常に Response Time Monitoring モニター・エージェントと通信できます。

ロード・バランサーの使用

ご使用の環境でロード・バランサーを使用している場合は、追加でカスタマイズを行う必要があります。

手順

ロード・バランサーを使用している場合は、以下のガイドラインに従ってください。

1. ロード・バランサーで URL の再書き込みをオフにします。
2. モニターする各 Web サーバーに Response Time Monitoring エージェントをインストールします。
   ロード・バランサーに Response Time Monitoring をインストールしないでください。

次のタスク

Response Time Monitoring エージェントをロード・バランサーの背後で実行する場合は、モニター・パフォーマンスを最適化するために、クライアントの IP アドレスを転送するようにロード・バランサーを構成できます。例として、次の手順を参照してください。

1. HTTP ヘッダーの **X-Forwarded-For** フィールドに、クライアントの IP アドレスを設定します。
2. $AGENT_HOME/config/*hostname*_t5.cfg ファイルで、SECTION=advconfig セクションに {KT5WEBPLUGIN_OVERRIDE_SOURCE_ADDR_HEADERS=X-Forwarded-For} を追加します。

   ヒント: 必要に応じて、パラメーターに複数の値を追加します。例: {KT5WEBPLUGIN_OVERRIDE_SOURCE_ADDR_HEADERS=x-forwarded-for, iv-remote-address}
3. Response Time Monitoring エージェントを再始動します。以下のコマンドを実行してください。

```
rt-agent.sh stop
rt-agent.sh start
```

IBM HTTP Server のモニターに使用する *CPU* の制限

飽和状態の環境では、IBM HTTP Server のインスツルメンテーションによる CPU 使用率を制限することをお勧めします。

このタスクについて

IBM HTTP Server 応答時間モジュールが使用できる CPU 使用率を指定します。デフォルトでは CPU 使用率は制限されていません。エージェントがインストールされているサーバーで CPU 使用率を構成します。

手順

CPU 使用率を構成するには、以下の手順を実行します。

1. エージェントを対話式または手動で再構成します。
   - 対話式に構成するために次のエージェント・スクリプトを実行します。

Linux AIX

```
$AGENT_HOME/bin/rt-agent.sh config
```

Windows

```
install_dir¥BIN¥rt-agent.bat config
```

- テキスト・エディターで以下のファイルを開きます。

Linux AIX

```
/opt/ibm/apm/agent/config/hostname_t5.cfg
```

Windows

```
C:¥IBM¥APM¥TMAITM6_x64¥hostname_T5.cfg
```

2. **advconfig** セクションに、次のパラメーターを追加し、値を 0 から 100 の間に設定します。

```
KT5WEBPLUGIN_TARGET_CPU_PERCENTAGE=10
```

ここで指定する値が、CPU 使用率の限度です。デフォルト値 *0* は、CPU 使用率に制限がないことを意味します。

3. また、以下のパラメーターを設定することもできます。

オプション	説明
KT5WEBPLUGINCONFIGPOSTURL	IBM HTTP Server インストール済み環境に対応する URL のリスト。 デフォルト: `http://localhost/WrtUpdateConfig.dat`
KT5WEBPLUGIN_MAX_REQUESTS_PER_SECOND	各 IBM HTTP Server インストール済み環境でモニターされる 1 秒当たりの要求数。要求数がこの数値を超えると、それ以降の要求はモニターされません。この制限に達すると、JavaScript の挿入が停止し、Response Time Monitoring エージェントにデータが返されなくなります。 デフォルト: `0` (最大値なし)
KT5WEBPLUGIN_CPUMAN_PERIOD_IN_SEC	目標値を超えたかどうかを判別するために CPU 使用率がチェックされる期間 (秒数)。 デフォルト: `60` 秒
KT5WEBPLUGIN_CATCHUP_PERIOD_COUNT	CPU がスケールバックされるまでに同じ状態に留まることを許容する期間の数。例えば、デフォルトを使用した場合、CPU 使用率が高いまま 4 サイクルが経過すると、CPU 使用量がスケールバックされます。 デフォルト: 3

タスクの結果

IBM HTTP Server 応答時間モジュールで使用できる CPU 使用率が固定の割合で設定されます。

Packet Analyzer のロードマップ

Packet Analyzer を使用して、HTTP トランザクションをモニターします。HTTPS モニターは手動で構成する必要があります。ブラウザーのタイミングを収集するように Web ページを手動で設定する必要があります。

Packet Analyzer を使用できる環境を判別するには、687 ページの『Response Time Monitoring コンポーネント』および 688 ページの『インストールの計画』を参照してください。

Response Time Monitoring エージェントをインストールすると、Packet Analyzer は自動的に有効になりますが、多数の追加ステップおよびカスタマイズを実行する必要が生じることがあります。

1. 「エージェント構成」ウィンドウでポート番号などの Packet Analyzer 設定をカスタマイズできます。詳しくは、705 ページの『「エージェント構成」ウィンドウを使用した Packet Analyzer の構成』を参照してください。
2. HTTPS トランザクションをモニターするには、ブラウザーのタイミングを収集するように Web ページを手動で設定します。詳しくは、707 ページの『HTTPS トランザクションのモニター』を参照してください。
3. ブラウザーのタイミングを有効にするには、JavaScript インジェクション・コンポーネントをアプリケーションに追加し、JavaScript モニター・コンポーネントをアプリケーションと 関連付けます。詳しくは、705 ページの『アプリケーションへの JavaScript モニター・コンポーネントの追加』を参照してください。
4. トランザクション負荷が高い環境を運用している場合は、いくつかの拡張チューニング・ステップを実行する必要が生じることがあります。詳しくは、712 ページの『Packet Analyzer の拡張構成』を参照してください。

「エージェント構成」ウィンドウを使用した Packet Analyzer の構成

「エージェント構成」ウィンドウを使用して、Packet Analyzer を構成できます。

Packet Analyzer を使用して特定のシステムの HTTP トラフィックをモニターするには、以下の手順を実行します。

1. 「エージェント構成」ページにアクセスするには、APM UI で、「**システム構成**」 > 「**エージェント構成**」を選択してから、「**応答時間**」タブを選択します。
2. 更新したいシステム (複数可) を選択します。複数のシステムのそれぞれに対して同じ HTTP 設定を使用する場合は、対象となる複数のシステムを選択します。

   選択するシステムに異なる HTTP 値が設定されている場合は、個々の値の代わりに、「複数の値」 (複数のリスト) が表示されます。 異なる値が含まれたシステムを同時に更新することはできません。
3. 「**HTTP トラフィックをモニターする**」フィールドで、値をダブルクリックして、リストから「**はい**」を選択します。
4. 「**モニター対象の HTTP ポート**」フィールドで、値をダブルクリックして、既にリストされているデフォルト・ポート 80 など以外の、モニター対象の追加のポートを入力します。

   ポートのモニターを停止するには、モニター対象から外すポートを選択して、「**削除**」をクリックします。
5. 「**変更の適用**」をクリックします。

アプリケーションへの JavaScript モニター・コンポーネントの追加

ユーザーがブラウザーの Web ページのパフォーマンスおよびエラーを把握できるように、Response Time Monitoring エージェントがブラウザーから時間データを収集できる必要があります。この機能を有効にするには、モニターするアプリケーションを 構成する必要があります。

このタスクについて

Web ページ内の対話をモニターできるようにするには、事前に JavaScript モニター・コンポーネントをアプリケーションの各 Web ページに追加しておく必要があります。JavaScript モニター・コンポーネントは、各 Web ページの状態およびそのページに関連する JavaScript の対話を収集します。モニターするアプリケーションに JavaScript モニター・コンポーネントを追加します。関連するコンテンツとアクションが自動的に収集され、分析と相関のために Cloud APM サーバーに送信されます。

手順

ブラウザーから実ユーザー・モニター・データを収集できるようにするためには、以下のステップに従ってください。アプリケーションの構成が変更されない限り、この手順を実行する必要があるのは 1 回のみです。

1. アプリケーションに JavaScript モニター・コンポーネントを追加します。使用する手順は、アプリケーションのタイプによって異なります。

   a) Java EE アプリケーションの場合、インストール・パッケージから、HTTP サーバーがアクセスできるディレクトリーに *install_dir*/clienttime/ClientTime.war を抽出します。

   b) Ruby、.NET、Python、Node.js などの Java EE 以外のアプリケーションの場合、インストール・パッケージから、HTTP サーバーがアクセスできるディレクトリーに *install_dir*/clienttime/wrtInstrumentation.js を保存します。

   一時的なパスに *install_dir*/clienttime/ClientTime.war ファイルを解凍します。解凍された wrtTimingTarget.dat ファイルを文書ルートにコピーする必要があります。文書ルートとは、HTTP サーバー (Apache、IIS など) 上の設定です。これは、文書を格納するディレクトリーです。デフォルトでは、すべての要求がこのディレクトリーから取り出されますが、他の位置を指すためにシンボリック・リンクと別名が使用される場合もあります。 例えば、Apache の文書ルートは /opt/IBM/HTTPServer/htdocs です。

   wrtInstrumentation.js ファイルは、任意のディレクトリーに配置できます。HTML ヘッダーで、wrtInstrumentation.js ファイルへのパスの場所を更新したことを確認してください。

2. JavaScript モニター・コンポーネントをアプリケーションに関連付けます。

   この関連付けは通常、アプリケーション・ヘッダー・スクリプトを変更することによって行うことができます。一般的に、モニターするコンポーネントごとまたはアプリケーションごとに、1 つのヘッダー・スクリプトのみを変更する必要があります。

   Java EE アプリケーションおよび非 Java EE アプリケーションの両方で、他のすべての JavaScript よりも前に、以下の JavaScript をアプリケーション・ヘッダーに追加します。

   ```
   <script language="JavaScript" src="path/wrtInstrumentation.js"
   type="text/JavaScript"></script>
   ```

   ここで、*path* は JavaScript モニター・コンポーネントの相対パスです。

   例:

   ```
   <script language="JavaScript" src="/ClientTime/wrtInstrumentation.js"
   type="text/JavaScript"></script>
   ```

タスクの結果

JavaScript モニタリング・コンポーネントが追加されたページがモニターされ、それらのページのデータが分析されて、エンド・ユーザー・トランザクション・ダッシュボードに表示されます。

ブラウザーのタイミングの有効化

リソース時間モニターを有効にすることで、Response Time Monitoring エージェントは Packet Analyzer を使用して W3C のリソース時間データを処理します。この機能を有効にすると、フロントエンド・エレメントの詳細なパフォーマンス情報を表示できます。

このタスクについて

リソース時間データをモニターできるようにするには、リソース時間モニター・コンポーネントをアプリケーションに追加し、リソース時間モニター・コンポーネントをアプリケーションと 関連付ける必要があります。リソース時間モニター・コンポーネントは、フロントエンド・エレメントの状態および対話を自動的に収集し、そのデータを分析のために Cloud APM サーバーに送信します。この分析の結果は、**サブトランザクション**・ダッシュボードに表示されます。

手順

リソース時間モニター機能を有効にするには、次の手順を実行します。アプリケーションの構成が変更されない限り、この手順を実行するのは 1 回のみです。

1. アプリケーションにリソース時間モニター・コンポーネントを追加します。
   a) インストール・パッケージから *install_dir*/clienttime/wrtInstrumentation.js ファイルを抽出します。
   b) wrtInstrumentation.js ファイルをアプリケーションの JavaScript ディレクトリーに追加します。
2. 以下の行をアプリケーション・ヘッダーに追加します。

```
<script> var wrt_enableResourceTiming=true; </script>
```

   例えば、次の場合があります。

```
<script language="JavaScript" src="path/wrtInstrumentation.js"
type="text/JavaScript"></script>
<script> var wrt_enableResourceTiming=true; </script>
```

タスクの結果

ページに、リソース時間モニター・コンポーネントの機能が備わりました。このコンポーネントは、デフォルトで有効になります。リソース時間モニター・コンポーネントの機能が備わったページのリソース時間データは、分析されて、**サブトランザクション**・ダッシュボードに表示されます。

次のタスク

リソース時間モニター・コンポーネントを無効にする場合は、**wrt_enableResourceTiming** パラメーターを false に設定します。

HTTPS トランザクションのモニター

Response Time Monitoring は、デフォルトでは HTTP トランザクションをモニターします。HTTPS トランザクションをモニターするには、リモート Web サーバーからの SSL トラフィックを暗号化解除できるように、Response Time Monitoring は SSL 証明書にアクセスする必要があります。

始める前に

モニター対象の HTTPS Web サーバーと、そのサーバーの IP アドレスおよび構成済みポートを識別します。例えば、192.168.1.23、ポート 443 などです。HTTPS Web サーバーごとに、Response Time Monitoring がその暗号を読み取ることができることを確認してください。Response Time Monitoring は、IBM Java でサポートされる暗号をサポートします。これには以下の暗号が含まれます。

- RSA_WITH_RC4_40_MD5
- RSA_WITH_RC4_128_MD5
- RSA_WITH_RC4_128_SHA
- RSA_WITH_RC4_40_SHA
- RSA_WITH_DES40_CBC_SHA
- RSA_WITH_DESC_CBC_SHA
- RSA_WITH_3DES_EDE_CBC_SHA
- RSA_WITH_AES_128_CBC_SHA
- RSA_WITH_AES_256_CBC_SHA
- RSA_EXPORT1024_WITH_RC4_56_MD5
- RSA_EXPORT1024_WITH_RC2_CBC_56_MD5
- RSA_EXPORT1024_WITH_DES_CBC_SHA
- RSA_EXPORT1024_WITH_RC4_56_SHA

- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256

制約事項 : Response Time Monitoring は、Diffie-Hellman 鍵交換を使用するトラフィックを暗号化解除できません。

手順

HTTPS トランザクション・モニターを有効にするには、以下の手順を実行します。

1. 鍵ストアをセットアップします。詳しくは、709 ページの『鍵ストアのセットアップ』を参照してください。
2. 次のいずれかのコマンドを実行し、プロンプトが出されたら値を指定して、Response Time Monitoring エージェントを構成します。

   例:

```
Response Time Monitoring エージェントの構成
「Response Time Monitoring エージェント」設定を編集しますか? [1=はい、2=いいえ] (デフォルト: 1): 1

基本構成: 基本モニター構成を指定します。
注: HTTP は、「エージェント構成」の下の「応答時間」タブを使用して一元的に構成されるようになっています。

HTTPS トランザクションをモニターする必要があるかどうかを指定します。
HTTPS トランザクションをモニターしますか? [1=はい、2=いいえ] (デフォルト: 2): 1

この鍵ストアには、モニター対象の HTTPS Web サイトの証明書が含まれます。
HTTPS 鍵ストア (例: /tmp/keys.kdb) (デフォルト: ): /tmp/keys.kdb

このテーブルでは、HTTPS サーバーを適切な証明書にマップします。(例:  cert1,
server ip,server port; cert2,server2 ip,server2 port);...
HTTPS サーバー証明書マップ (例: certAlias,9.48.152.1,443;...)(デフォルト: ):
 label1,10.0.0.1,9443;label1,9.185.150.71,443

拡張構成:
拡張モニター構成を指定します。

選択された IP アドレスを持つ NIC カードがモニターされます。
モニター対象の NIC の IP アドレス (デフォルト: ): 10.0.0.1

データ収集および分析の構成:
データの分析方法についての構成情報を指定します。

構成が正常に完了しました。
構成変更を適用するには、エージェントの再始動が必要です。
```

   ここで、

   - HTTPS 鍵ストアは、ステップ 1 で構成された鍵ストアです。
   - HTTPS サーバー証明書マップでは、以下を指定します。
     - `label 1`: ステップ 1 で構成された鍵ラベル
     - `server ip`: サーバーの IP アドレス。パケットの IPV4 ヘッダー内のソース/宛先属性と一致する必要があります
     - `server port`: サーバー・ポート番号。パケットの TCP ヘッダー内のソース/宛先ポート属性と一致する必要があります

     同じ鍵ラベルのサーバー IP が複数ある可能性がある場合は、複数の項目を追加します。
   - モニター対象の NIC の IP アドレス: パケットを認識でき、eth0 や en0 などにマップされるインターフェース。名前がパケットの IPV4 ヘッダーまたは TCP ヘッダーの属性と一致する必要はありません。`10.0.0.1` が eth0 に対応する場合は、`tcpdump -s0 -i eth0 ...` を使用して、Packet Analyzer で分析する必要があるすべてのパケットを表示します。

3. Response Time Monitoring エージェントを再始動します。

鍵ストアのセットアップ

HTTPS トランザクションをモニターするには、モニターするすべての Web サーバー用に、KT5Keystore に鍵をインポートします。

このタスクについて

IBM 鍵管理 (iKeyman) を使用して、モニター対象の Web サーバーから SSL 証明書をエクスポートし、それらを HTTPS 鍵ストアにインポートするか、あるいは HTTPS 鍵ストアで Web サーバーの鍵ストア stash ファイル (.kdb) を指定することができます。Response Time Monitoring のインストール時または構成時には、keys.kdb ファイルの場所の入力を求めるプロンプトが出されます。

鍵ストア stash ファイル (.kdb および .sth) がない場合は、iKeyman を使用して鍵データベースをセットアップできるように、ご使用の Java バージョンで CMS プロバイダーが有効になっていることを確認してください。

1. *install_dir*/ibm-jre/jre/lib/security ディレクトリーに移動します。例:
   - Linux /opt/ibm/apm/agent/JRE/lx8266/lib/security
   - Windows C:¥Program Files¥IBM¥APM¥ibm-jre¥jre¥lib¥security
2. java.security ファイルで、以下に示すようにセキュリティー・プロバイダーのリストにステートメントを追加します。ここで *number* は、リスト内の最後のシーケンス番号です。

   ```
   security.provider.number=com.ibm.security.cmskeystore.CMSProvider
   ```

   プロバイダーのリストは、以下の例のようになります。

   ```
   ## List of providers and their preference orders #
   security.provider.1=com.ibm.jsse.IBMJSSEProvider
   security.provider.2=com.ibm.crypto.provider.IBMJCE
   security.provider.3=com.ibm.security.jgss.IBMJGSSProvider
   security.provider.4=com.ibm.security.cert.IBMCertPath
   security.provider.5=com.ibm.security.cmskeystore.CMSProvider
   ...
   #
   ```

3. ファイルを保存して閉じます。

制約事項 : Response Time Monitoring は、Diffie-Hellman 鍵交換を使用してトラフィックを暗号化解除することはできません。

手順

HTTPS トランザクション・モニターを有効にするには、モニターする Web サーバーから SSL 証明書を収集し、iKeyman を使用して証明書および鍵ストア stash ファイルを HTTPS 鍵ストアにインポートします。以下の例では、iKeyman を使用して、IBM HTTP Server から証明書をエクスポートし、それらを HTTPS 鍵ストアにインポートします。

1. モニターする各 HTTPS Web サーバーに Response Time Monitoring エージェントをインストールします。
2. オペレーティング・システムに応じて以下のコマンドのいずれかを実行し、**IBM 鍵管理** (iKeyman) を IBM Java の bin ディレクトリー内から実行します。
   - AIX Linux /opt/ibm/apm/agent/JRE/lx8266/bin/ikeyman

     注 : iKeyman が正しく動作するためには、環境に X Window が必要です。
   - Windows c:¥IBM¥APM¥java¥java80_x64¥jre¥bin¥ikeyman
3. 新規鍵ストア・データベースを作成します。「**新規**」ダイアログ・ボックスで、以下の手順を実行します。

   a)「**鍵データベース・タイプ**」リストから、「**CMS**」を選択します。

   CMS がリストで選択不可である場合、CMS プロバイダーが有効になっていない可能性があります。Java セキュリティー・ファイルで CMS プロバイダーを有効にしてください。

b)「**ファイル名**」フィールドで、HTTPS 鍵ストア・ファイルの名前を入力し、「**OK**」をクリックします。
例えば、keys.kdb です。

4.「**パスワード・プロンプト**」ダイアログ・ボックスで、以下の手順を実行します。
a)「**パスワード**」および「**パスワードの確認**」フィールドで、パスワードの入力および確認を行い、keys.kdb にアクセスします。

定期的に鍵ストア・データベースを再作成して Response Time Monitoring エージェントを再始動したい場合を除き、有効期限を設定しないでください。

b)「**パスワードをファイルに入れて隠しますか?**」を選択して、keys.kdb のパスワードを暗号化された形式で stash ファイル keys.sth 内に格納します。

注: 応答時間エージェントがサポートする隠されたパスワード・バージョンは 1 つのみです。APM 8.1.4 以降は、次のコマンドを実行して、keys.kdb のパスワードを暗号化された stash ファイル keys.sth に格納します。

Linux の場合:

```
cp keyfile.sth keyfile.sth.new-format
```

```
cd /opt/IBM/ccm/agent/lx8266/gs/bin
```

```
#export LD_LIBRARY_PATH=/opt/ibm/apm/agent/lx8266/gs/lib64:$LD_LIBRARY_PATH
```

```
./gsk8capicmd_64 -keydb -stashpw
-db /opt/IBM/ccm/agent/keyfiles/keyfile.kdb -v1stash
```

Windows:

```
copy server.sth server.sth.backup
```

```
set PATH=c:¥IBM¥APM¥GSK8_x64¥lib64;%PATH%
```

```
C:¥IBM¥APM¥GSK8_x64¥bin¥gsk8capicmd_64 -keydb
-stashpw -db .¥server.kdb -pw passw0rd -v1stash
```

5. iKeyMan ウィンドウの「**鍵データベースの内容**」セクションで、以下の手順を実行します。
a)「**個人証明書**」を選択します。
b)「**インポート**」をクリックします。
c)「**鍵のインポート**」ダイアログ・ボックスで、「**鍵ファイル・タイプ**」リストから「**CMS**」を選択します。
d) 鍵ストア・ファイルを参照し、「**開く**」をクリックし、次に「**OK**」をクリックします。
e)「**パスワード・プロンプト**」ダイアログ・ボックスで、鍵ストアのパスワードを入力します。
f) リストから鍵を選択し、「**OK**」をクリックします。
g)「**ラベルの変更**」ダイアログ・ボックスで、鍵のラベル名を選択します。「**新規ラベルの入力 (Enter a new label)**」フィールドにサーバーのホスト名を指定し、「**適用**」をクリックします。

注 : この値は、Response Time Monitoring を構成するときに必要なため、メモしておいてください。
h)「**OK**」をクリックします。
6. HTTPS 鍵ストアを保存します。

インターネット インフォメーション サービスからの鍵のインポート

Internet Information Services から鍵を抽出して KT5Keystore にインポートするには、次の手順を実行します。

1. モニターする各 HTTPS Web サーバーに Response Time Monitoring エージェントをインストールします。

2. 次のようにして、Internet Information Services から .pfx ファイルをエクスポートします。
   a. Windows の「**スタート**」メニューで、「**管理ツール**」 > 「**インターネット インフォメーション サービス (IIS) マネージャ**」を選択します。
   b. エクスポートする秘密鍵を持つ Web サーバーとサイトを選択し、右クリックして、コンテキスト・メニューから「**プロパティ**」を選択します。
   c.「**ディレクトリ セキュリティ**」タブを選択し、「**セキュリティで保護された通信**」セクションで「**サーバー証明書**」を選択します。
   d.「**IIS 証明書ウィザード**」で、「**次へ**」をクリックします。
   e.「**.pfx ファイルに現在の証明書をエクスポートする**」を選択し、「**次へ**」をクリックします。
   f. パスとファイル名を入力し、「**次へ**」をクリックします。
   g. 鍵のエクスポート・パスワードを入力し、「**次へ**」をクリックします。
   h. 以降のすべてのページで「**次へ**」をクリックし、「**完了**」をクリックします。
3. 次のようにして、.pfx ファイルから個人証明書と署名者証明書を抽出します。
   a. IBM Java の bin ディレクトリー内からコマンド c:¥IBM¥APM¥java¥java80_x64¥jre¥bin¥ikeyman を使用して、**IBM Key Management** (iKeyman) を実行します。環境変数 JAVA_HOME が設定されていることを確認します。
   b. 鍵ストア・データベースで、「**ファイル**」 > 「**開く**」を選択します。
   c.「**鍵データベース・タイプ**」リストから、「**PKCS12**」を選択します。
   d. 上の手順で作成した .pfx ファイルの名前とパスを入力し、「**OK**」をクリックします。プロンプトが表示されたら、パスワードを入力し、「**OK**」をクリックします。
   e.「**鍵データベースの内容**」 > 「**個人証明書**」を選択し、「**エクスポート/インポート**」をクリックします。
   f. アクション・タイプとして「**鍵のエクスポート**」、鍵ファイル・タイプとして「**PKCS12**」を選択します。エクスポートした鍵のファイル名とロケーションを入力し、「**OK**」をクリックします。プロンプトが表示されたら、エクスポート・パスワードを入力し、「**OK**」を再度クリックします。
   g. 個人証明書が認証局によって署名されている場合は、「**鍵データベースの内容**」 > 「**署名者の証明書**」を選択し、「**抽出**」をクリックします。デフォルトのファイル・タイプを選択し、エクスポートした証明書のファイル名とロケーションを入力して、「**OK**」をクリックします。
4. 次のようにして、署名者の .cer ファイルを抽出します (必要な場合)。
   a. 署名者の証明書ファイルが .pfx ファイルから抽出された場合、そのファイルが保存されたディレクトリーに移動し、拡張子 .cer を持つ新しいコピーを作成します。その新しいコピーをダブルクリックして、Windows の証明書ビューアーで開きます。
   b.「**証明のパス**」タブで、署名者の証明書チェーンを表示できます。チェーンの一番下の項目が個人証明書です。それより上のすべての証明書について、次の手順を実行します。
      1) 証明書を選択し、「**証明書の表示**」をクリックします。
      2)「**詳細**」を選択し、「**ファイルへコピー**」をクリックします。
      3) 証明書のエクスポート・ウィザードのデフォルト値をすべて受け入れ、.cer 拡張子を持つファイル名を入力します。
5. 新規鍵ストア・データベースを作成します。「**新規**」ダイアログ・ボックスで、以下の手順を実行します。
   a.「**鍵データベース・タイプ**」リストで、「**CMS**」を選択し、ファイル名とロケーションを入力します。プロンプトが表示されたら、新しい鍵ストアのパスワードを入力します。

      注 : 「**パスワードをファイルに入れて隠しますか?**」を選択するようにしてください。
   b. .pfx から署名者証明書を抽出したら、次の手順を実行します。
      1)「**鍵データベースの内容**」 > 「**署名者の証明書**」を選択します。
      2) 署名者証明書ごとに、「**追加**」をクリックして、.cer ファイルを追加します。

c.「**鍵データベースの内容**」 > 「**個人証明書**」を選択し、「**インポート**」をクリックします。

d. 鍵ファイル・タイプとして「**PKCS12**」を選択し、**.p12** ファイルの名前とロケーションを選択します。プロンプトで求められたら、パスワードを入力します。

e. 鍵ストアを保存し、鍵管理ユーティリティーを 終了します。

f. `.kdb` ファイルと `.sth` ファイルを、Response Time Monitoring アプライアンス・マシンの KT5Keystore にコピーします。

g. IBM 鍵管理データベース・ファイル (`.kdb`) とスタッシュ (`.sth`) を安全なディレクトリーに配置し、それらを読み取ることができるのは、管理者または `root` (あるいは Response Time Monitoring エージェントのインストールに使用したユーザー ID) のみになるようにします。

Packet Analyzer の拡張構成
Packet Analyzer 用の多数の拡張構成オプションがあります。

Packet Analyzer を構成した後で、多数の拡張構成タスクを実行して、パフォーマンスと機能を微調整できます。

ロード・バランサーの使用
ご使用の環境でロード・バランサーを使用している場合は、追加でカスタマイズを行う必要があります。

手順

ロード・バランサーを使用している場合は、以下のガイドラインに従ってください。

1. ロード・バランサーで URL の再書き込みをオフにします。
2. モニターする各 Web サーバーに Response Time Monitoring エージェントをインストールします。
   ロード・バランサーに Response Time Monitoring をインストールしないでください。
3. ご使用のアプリケーションに JavaScript モニター・コンポーネントを追加します。詳しくは、705 ページの『アプリケーションへの JavaScript モニター・コンポーネントの追加』を参照してください。

次のタスク

Response Time Monitoring エージェントをロード・バランサーの背後で実行する場合は、モニター・パフォーマンスを最適化するために、クライアントの IP アドレスを転送するようにロード・バランサーを構成できます。例として、次の手順を参照してください。

1. HTTP ヘッダーの **X-Forwarded-For** フィールドに、クライアントの IP アドレスを設定します。
2. クライアントの IP アドレス・ヘッダーを使用するように Response Time Monitoring エージェントを構成します。ご使用のオペレーティング・システムに応じて、次のいずれか 1 つのファイルの **KFC_OVERRIDE_SOURCE_ADDR_HEADER** フィールドにクライアントの IP アドレス・ヘッダーを設定します。

   - AIX Linux `/opt/ibm/apm/agent/tmaitm6/wrm/kfcmenv`
   - Windows `C:¥IBM¥ITM¥TMAITM6_x64¥wrm¥Analyzer¥kfcmenv`

   例:

   ```
   KFC_OVERRIDE_SOURCE_ADDR_HEADER=x-forwarded-for
   ```

   WebSEAL を使用している場合は、次のようにします。

   ```
   KFC_OVERRIDE_SOURCE_ADDR_HEADER=iv-remote-address
   ```

CPU オーバーヘッド制限の構成

トランザクション負荷が高い環境を運用している場合、Response Time Monitoring エージェントによって使用されるモニター・リソースを制限できます。

このタスクについて

この機能は、サンプリングによって一部の Web トラフィックのみをモニターして報告することで、Response Time Monitoring エージェントの CPU 使用率を制限します。 CPU オーバーヘッド制限の構成は、デフォルトでは構成されていません。CPU オーバーヘッド制限は、エージェントのインストール先サーバーで構成する必要があります。

手順

CPU オーバーヘッド制限を構成するには、以下の手順を実行します。

1. テキスト・エディターで以下のファイルを開きます。

   Linux AIX `/opt/ibm/apm/agent/tmaitm6/wrm/kfcmenv`

   Windows `C:¥IBM¥ITM¥TMAITM6_x64¥wrm¥Analyzer¥kfcmenv`

2. 以下のパラメーターの値を構成します。

   KFC_MAX_PROTOCOL_PACKETRATE
   初期の最大パケット率。例えば、このパラメーターを **KFC_MAX_PROTOCOL_PACKETRATE=2000** と設定すると、 最大パケット率が 1 秒間に 2000 パケットになります。この比率は、**KFC_CPUTHROTTLE_TARGET** および現在の CPU 使用量の値に応じて動的に変化します。

   KFC_CPUTHROTTLE_TARGET
   kfcmserver プロセスが使用できる全 CPU リソースの割合。例えば、 このパラメーターを **KFC_CPUTHROTTLE_TARGET=10.0** と設定すると、kfcmserver プロセスが全 CPU リソースの 10% までを使用できるようになります。

   注 : パラメーター **KFC_CPUTHROTTLE_TARGET** の値は、kfcmserver プロセスに使用可能な全 CPU リソースの割合です。例えば、CPU コアが 4 つ存在する場合に **KFC_CPUTHROTTLE_TARGET** を 10 に設定すると、Windows のリソース・モニターでは CPU リソースが 400% と測定されます。結果として、kfcmserver プロセスは、全部で 400% の使用可能 CPU リソースのうち 40% までを使用できます。

タスクの結果

CPU オーバーヘッド制限は Response Time Monitoring エージェントを対象として構成されます。

IBM HTTP Server 応答時間モジュールから Packet Analyzer への再構成

IBM HTTP Server 応答時間モジュールから Packet Analyzer にモニター環境を変更することが必要な場合が考えられます。

手順

1. HTTP Server エージェントをアンインストールします。
   a) /etc/httpd/conf/httpd.conf を編集して、応答時間プラグイン行をコメント化します。例:

   ```
   #include /opt/ibm/apm/agent/tmp/khu/khu.etc.httpd.conf.httpd.conf
   ```

   b) HTTP Server エージェントをアンインストールします。例:

   ```
   /opt/ibm/apm/agent/bin/http_server-agent.sh uninstall
   ```

   c) 次の手順で応答時間エージェントを再構成する前に、新しいコマンド・プロンプト・ウィンドウを開いて、システム変数をクリーンアップします。
2. 次のファイルを開きます。

   Linux AIX *install_dir*/config/*hostname*_t5.cfg

ここで、*install_dir* は /opt/ibm/apm/agent です

Windows *install_dir*¥TMAITM6_x64¥*hostname*_t5.cfg

ここで、*install_dir* は C:¥IBM¥APM です

3. 次のようにパラメーターを設定します。

```
{ KT5DISABLEANALYZER=NO } { KT5ENABLEWEBPLUGIN=NO }
```

4. 次のようにエージェントを再構成します。

```
re-agent.bat config  install_dir¥samples¥rt_silent_config.txt
```

エンド・ユーザー・トランザクションのロケーション値のカスタマイズ

エンド・ユーザー・トランザクション・ダッシュボードで指定の IP アドレスまたはアドレス範囲に適用されるロケーションは、特定の環境に合わせてカスタマイズできます。

始める前に

ロケーション値をカスタマイズするには、「**エージェント構成**」の「**地理位置情報**」タブを使用します。

この機能を使用して、ダッシュボードに「**不明**」と表示される IP アドレスのロケーションを設定します。これらのアドレスは、192.168.x.x や 10.x.x.x などの内部アドレスの場合もあれば、解決されない外部 IP アドレスの場合もあります。この機能を使用して、IP アドレスの正しくないロケーションをオーバーライドすることもできます。例えば、IP アドレス 9.1.1.1 は Los Angeles に存在することが分かっていますが、San Francisco と表示される場合は、このロケーションをオーバーライドして、9.1.1.1 を Los Angeles に設定できます。

このタスクについて

必要な値を格納する CSV ファイルをアップロードすることで、「エンド・ユーザー・トランザクション」ダッシュボードのロケーション値をカスタマイズします。サンプルの CSV ファイルが「**地理位置情報**」タブにあります。

CSV ファイルには、ヘッダーとして以下の値が含まれている必要があります。これらの値の順序は任意ですが、エントリーは以下の順序と一致していなければなりません。

```
IP_ADDRESS, COUNTRY, REGION, CITY
```

例えば、次の場合があります。

```
IP_ADDRESS, COUNTRY, REGION, CITY
10.0.5.0/24, Australia, WA, Perth
10.1.0.6, Australia, VIC, Melbourne
```

単一の IPv4 アドレスまたはアドレスの範囲を指定できます。範囲を指定する場合、1 から 32 の範囲の有効な値を使用するようにしてください。

手順

「エンド・ユーザー・トランザクション」ダッシュボードに表示されるロケーション値をカスタマイズするには、Application Performance ダッシュボードで次の手順を実行します。

1. ロケーションに対応する IP アドレスを使用して、CSV ファイルをセットアップします。
2. その CSV ファイルをアップロードします。
   a)「**エージェント構成**」 > 「**地理位置情報**」に移動します。
   b)「**CSV のアップロード**」をクリックし、アップロードするファイルを選択して、「**オープン**」をクリックします。
   - CSV ファイルに、特定の IP アドレスより前に、まず汎用 IP アドレスの範囲がリストされていることを確認します。
   - 必要に応じて複数のファイルをアップロードします。

- あるファイルの値が別のファイルの値と重なる場合、新しいファイルの値で前のファイルの値がオーバーライドされます。

3. 「**アップロードの結果**」を展開してエラーを確認します。次の問題を確認します。
   - オーバーライド
   - 無効な IP アドレス
   - 無効な行
   - 250 文字より長い値

タスクの結果

数分間待つと、「エンド・ユーザー・トランザクション」ダッシュボードにカスタマイズした値が表示されます。

次のタスク

必要に応じてカスタマイズした値を削除できます。以下のいずれかのステップを実行します。

- カスタマイズした値の一部を削除するには、削除する IP アドレスを選択し、「**選択したエントリーのクリア**」をクリックします。次に、「**OK**」をクリックして、削除を確認します。
- カスタマイズした値をすべて削除するには、「**すべてのエントリーのクリア**」をクリックし、「**OK**」をクリックして削除を確認します。

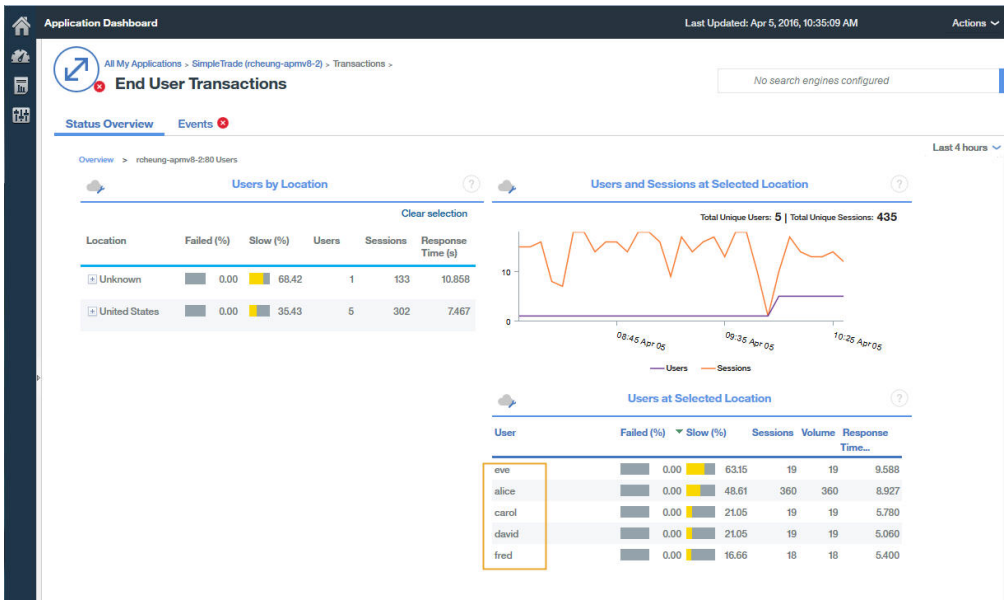
追加の Web アプリケーションのトラッキング

デフォルトでトラッキングされるアプリケーション 以外の Web アプリケーションもトラッキングするには、ユーザーおよびセッションをトラッキングする方式を識別して構成します。

始める前に

デフォルトでサポートされていないアプリケーションについては、ダッシュボードにユーザーとセッションの詳細は含まれず、ユーザー名は anonymous または unknown として表示され、セッション情報は入手できません。

ユーザー・トラッキング方式が正しく構成されている場合、ユーザー ID が抽出されて、「**エンド・ユーザー・トランザクション**」 > 「**ユーザー要約**」 > 「**選択されたロケーションのユーザー**」ダッシュボードにリストされます。



注: ユーザー・トラッキングは、セッション・トラッキングに基づいています。まず応答時間エージェントの構成設定で、ユーザー・トラッキングとセッション・トラッキングの方式に対して、正しいセッション・トラッキング方式の変数を設定する必要があります。

このタスクについて

IBM Application Performance Management V8.1.4 以降では、「**エージェント構成**」 > 「**応答時間**」ページを使用して、Packet Analyzer または IBM HTTP Server 応答時間モジュールによるトラッキング対象のアプリケーションを追加できます。このページで定義されている値が、`WRT_Defaults.xml` ファイル内の値より優先されます。

追加のアプリケーションをトラッキングするには、まず、モニター対象のアプリケーションのユーザー ID およびセッション ID 方式と値を識別する必要があります。例:

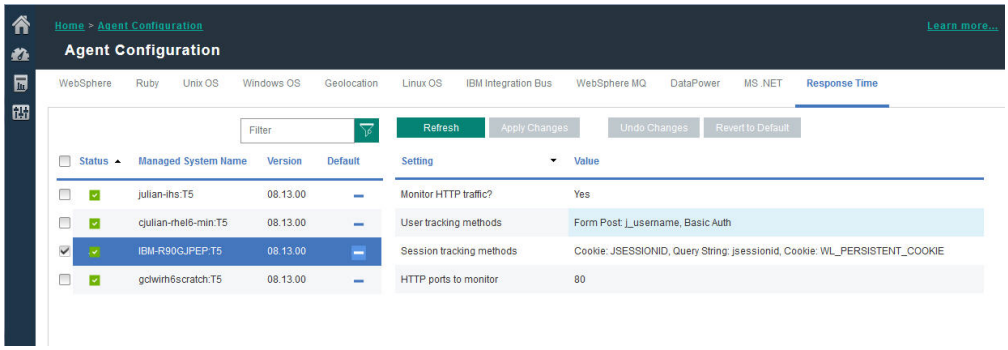
1. ブラウザーの開発者向けツールを開き、モニター対象のアプリケーションの要求を確認できるようにします。
2. ブラウザーのネットワーク・ログで最後の要求を選択します。これにより、テスト要求を簡単に識別できるようになります。
3. 認識できるパラメーターを使用してテスト要求を作成します。例えば、`testuser` を使用して Web サイトにログインします。
4. テスト要求を選択し、`Headers` を調べます。
5. 要求ログからセッション ID を識別します。通常、セッション ID は `cookie`、`POST`、`request/response header`、または `query string` に指定されています。デフォルト・プロファイルで `cookie` が定義済みの場合、ステップ 2 でこれを追加する必要はありません。
6. 要求ログからユーザー ID を識別します。ユーザー ID は、`cookie`、`request header`、`POST`、または `query string` の内容として指定されている場合があります。 例えば、`testuser` を検索すると、ユーザー ID の値がわかります。
7. 「**ユーザー・トラッキング方式 (User tracking methods)**」と「**セッション・トラッキング方式 (Session tracking methods)**」の両方を、お客様のアプリケーション・コードで使用中のセッションとユーザーの正しい値の名前で更新する必要があります。セッションとユーザー名の値の名前の識別方法は、アプリケーションのコードに応じて異なります。8.1.4 でのユーザー/セッション設定のデフォルト値を次に示します。

```
Session tracking methods=cookie¥:JSESSIONID,querystring¥:jsessionid,cookie¥
       :WL_PERSISTENT_COOKIE
User tracking methods=formpost¥:j_username,formpost¥:uid,formpost¥
       :ctl00%24MainContent%24uid,basicauth¥:Authorization¥: Basic
```

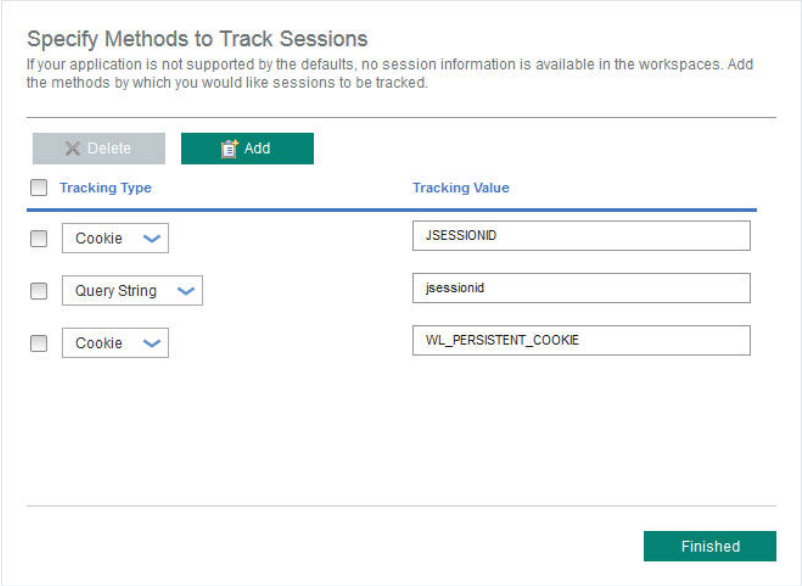
手順

アプリケーション内で使用するユーザーおよびセッションをトラッキングする方式と値を識別したら、以下の手順を実行します。

1. 「**エージェント構成**」 > 「**応答時間**」構成ページに移動します。



2. 更新する管理対象システムを選択します。
3. 必要に応じて、セッション・トラッキング方式を更新します。
   a) 「**セッション・トラッキング方式**」フィールドの値をクリックします。



b)「**セッションをトラッキングする方式の指定**」ウィンドウで、「**追加**」をクリックします。

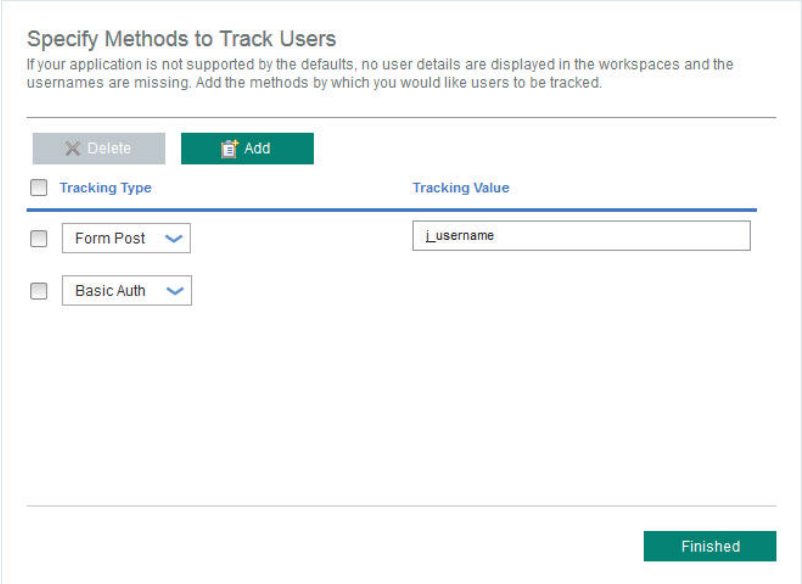
c)「**トラッキング・タイプ**」リストから、トラッキング・タイプを選択します。例えば、「Cookie」です。

d)「**トラッキング値**」フィールドに値を指定します。例えば、WL_PERSISTENT_COOKIE です。

e)「**終了**」をクリックします。

4. 必要に応じて、ユーザー・トラッキング方式を更新します。

a)「**ユーザー・トラッキング方式**」フィールドの値をクリックします。



b)「**ユーザーをトラッキングする方式の指定**」ウィンドウで、「**追加**」をクリックします。

c)「**トラッキング・タイプ**」リストから、トラッキング・タイプを選択します。例えば、「ヘッダー」です。

d)「**トラッキング値**」フィールドに値を指定します。例えば、username です。

e)「**終了**」をクリックします。

5.「エージェント構成」ページで、「**変更の適用**」をクリックします。

タスクの結果

アプリケーション・ダッシュボードに、新しく指定したトラッキング方式を使用するアプリケーションが表示されます。

次のタスク

アプリケーション・ダッシュボードに、アプリケーションのユーザー ID およびセッション情報が表示されるかどうかをテストします。

Response Time Monitoring エージェントの固有の管理対象システム名の指定

Cloud APM コンソールに表示される Response Time Monitoring エージェントのインスタンス名は、管理対象システム名 (MSN) とも呼ばれます。エージェント構成パラメーターを使用して、各エージェント・インスタンスに固有の MSN を指定できます。

このタスクについて

Response Time Monitoring エージェントの管理対象システム名の形式は以下のとおりです。

```
instancename:hostname:T5
```

T5 は Response Time Monitoring エージェントの製品コードです。

手順

1. すべての既存のエージェント・インスタンスを停止します。既存のエージェント・インスタンスがない場合は、次のステップに進みます。エージェント・インスタンスの停止について詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。
2. Linux AIX runagent ファイル内の **CTIRA_SUBSYSTEM_ID** を変更します。通常は、マシン上のすべてのエージェント・インスタンスが同じホスト名の値を使用します。
   a) 以下のファイルのバックアップ・コピーを作成します。

   Linux AIX
   ```
   install_dir/platform/t5/bin/runagent
   ```

   b) ファイルを編集します。Linux システムまたは AIX システムでは *newinstancename* を追加します。

   Linux AIX
   ```
   CTIRA_SUBSYSTEM_ID=newsubsystemid
   ```
3. エージェントの既存のインスタンスを開始します。
4. Cloud APM コンソールを始動します。以前の MSN のエージェント・インスタンスを削除し、新しいエージェント・インスタンスを追加して、アプリケーションを変更します。

Ruby モニターの構成

オンプレミスと IBM Cloud の両方の Ruby アプリケーションをモニターできます。オンプレミスの Ruby アプリケーションをモニターする場合は、Ruby エージェントを構成します。IBM Cloud の Ruby アプリケーションをモニターする場合は、Ruby データ・コレクターを構成します。

このタスクについて

ここに示す手順は、明記されている場合を除き、最新リリースのエージェント用です。

以下の手順は、Ruby エージェントおよび Ruby データ・コレクターを構成するためのロードマップであり、必須のステップとオプションのステップの両方が含まれています。ニーズに応じて、構成手順を実行してください。

手順

- オンプレミスの Ruby アプリケーションをモニターするには、以下の手順を実行して Ruby エージェントを構成します。
  a) Ruby アプリケーションをモニターするようにエージェント・インスタンスを構成します。『Ruby アプリケーションをモニターするための Ruby エージェントの構成』を参照してください。
  b) モニター・データを Cloud APM コンソールで表示するために、データ・コレクターをインストールします。『データ・コレクターのインストール』を参照してください。
  c) オプション: Cloud APM, Advanced ユーザーの場合、ニーズに従って、以下のタスクを実行できます。
    – 診断データを収集するようにデータ・コレクターを構成するには、『診断データ・コレクターの構成』を参照してください。
    – 要求のメソッド・トレースを有効にし、「要求スタック・トレース」ウィジェットに表示されるファイル・パス・パラメーターの長さを調整するには、『メソッド・トレースの有効化およびパス表示の調整』を参照してください。
    – JVM ヒープ・サイズを増やして「`out of memory`」エラーを回避するには、『JVM ヒープ・サイズの増加』を参照してください。
    – 診断を無効にするには、Ruby アプリケーション用の診断データの無効化または有効化を参照してください。
- IBM Cloud の Ruby アプリケーションをモニターする 場合は、以下のタスクを実行して Ruby データ・コレクターを構成します。
  a) IBM Cloud アプリケーション用に Ruby データ・コレクターを構成します。説明については、727 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクターの構成』を参照してください。
  b) オプション: Ruby データ・コレクターの動作を変更するには、728 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクター のカスタマイズ』を参照してください。

Ruby エージェントの構成

Ruby エージェントがアプリケーションをモニターするようにするには、Ruby ランタイムを指定します。結果として、ランタイムを使用して、Ruby アプリケーションからデータを収集し、エージェントを構成します。

始める前に

Ruby アプリケーションの開始に使用するサーバー、およびエージェントで使用している Ruby または Rake 実行可能ファイルの修飾 bin ディレクトリーを判別します。

1. 使用しているアプリケーション・サーバーを判別するには、以下のコマンドを実行します。

   ```
   ps -ef | grep ruby
   ```

   アプリケーションの開始に使用されるサーバーの名前が表示されます。考えられるサーバー名のリストは、以下のとおりです。

   - `Passenger`
   - `Unicorn`
   - `Puma`
   - `Thin`

   上記リストに示されているサーバー名がコマンド出力で表示されない場合、アプリケーションの開始に使用するサーバーは、WEBrick の可能性があります。

   重要 : 複数の Web サーバーを使用して Ruby アプリケーションを開始している場合、アプリケーション Web サーバーごとに 1 つのエージェント・インスタンスを作成する必要があります (例えば、PUMA に 1 つ、Unicorn に 1 つのインスタンスを作成します)。

2. Ruby エージェントが使用している Ruby または Rake 実行可能ファイルの修飾 bin ディレクトリーを判別するには、以下のコマンドを実行します。

```
which ruby
```

このタスクについて

ニーズに応じて、複数のエージェント・インスタンスを構成するために、このタスクを繰り返すことができます。

手順

1. エージェントを構成するには、以下のコマンドを実行します。
   install_dir/bin/ruby-agent.sh config *instance_name*
   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は Ruby エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。

   重要 : 長いインスタンス名は指定しないでください。ホスト名とエージェント・インスタンス名の長さの合計は、28 文字を超えてはなりません。長さが制限を超過している場合、管理対象システム名が切り捨てられ、Ruby エージェントの製品コードが正しく表示されません。

   管理対象システム名には、指定したインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、エージェントの 2 文字の製品コードです。例えば、インスタンス名に Ruby2 を指定すると、管理対象システム名は Ruby2:hostname:KM になります。ここで、*KM* は、Ruby エージェントの 2 文字の製品コードです。
2. 「Edit 'Monitoring Agent for Ruby' settings」というプロンプトが出されたら、1 と入力して続行します。
3. 「Fully Qualified Rubies Bin Directory」というプロンプトが出されたら、バイナリーのディレクトリーを指定します。例えば、Ruby Version Manager (RVM) を使用している場合、/usr/local/rvm/rubies/ruby-2.0.0-p247/bin と入力します。
4. 「Auto Detect Ruby Applications Flag」というプロンプトが出されたら、Y と入力して続行します。エージェントが、エージェント・データ・コレクターによって送信されたデータを受信します。
5. 「Application Server Process name」というプロンプトが出されたら、Enter キーを押してデフォルトの ruby を受け入れるか、以下のリストに従って、使用しているサーバーの値を指定します。
   - WEBrick サーバーの場合、デフォルトを受け入れるか、ruby と指定します。Rails が Ruby スタックによってインストールされている場合は、.ruby.bin と指定します。
   - Passenger サーバーの場合は passenger と指定します。
   - Unicorn サーバーの場合は unicorn と指定します。
   - Puma サーバーの場合は puma と指定します。
   - Thin サーバーの場合、コマンド thin start を実行してアプリケーションを始動する場合はデフォルトを受け入れて ruby を使用します。コマンド thin start -d を実行してアプリケーションを始動する場合は thin と指定します。Rails が Ruby スタックによってインストールされており、コマンド thin start を実行してアプリケーションを始動する場合は .ruby.bin と指定します。
6. 「Socket Data Source」というプロンプトが出されたら、Enter を押してデフォルトの 0 を受け入れ、一時ポートを使用します。
7. 「Edit 'Application' settings」というプロンプトが出されたら、5 と入力して設定を終了します。
8. エージェントを始動するには、次のコマンドを実行します。
   install_dir/bin/ruby-agent.sh start *instance_name*

次のタスク
Ruby エージェントが正しく機能し、データが Cloud APM UI で表示されるようにするために、データ・コレクターをインストールします。説明については、『データ・コレクターのインストール』を参照してください。

データ・コレクターのインストール
エージェントを正常に機能させるために、データ・コレクターをインストールする必要があります。データ・コレクターをインストールした後には、Application Performance ダッシュボードにモニター・データが表示されます。

始める前に
非 root アカウントを使用して Linux システムに Ruby on Rails アプリケーションをインストールした場合に診断データを収集するには、その非 root ユーザーには、診断データ・コレクターのホーム・ディレクトリーへのアクセス権限が付与されている必要があります。その非 root ユーザーが *install_dir*/install-images/kkm ディレクトリーに対する読み取りおよび書き込み権限を持っていることを確認します。ここで、*install_dir* は Ruby エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。 必要であれば、chmod 777 コマンドを使用して読み取りおよび書き込み権限を与えます。

手順

1. Ruby on Rails アプリケーションを停止します。
2. オプション: Ruby データ・コレクターを新しいバージョンにアップグレードする 場合は、事前に以下のコマンドを実行して古いバージョンのデータ・コレクターをアンインストールする 必要があります。

   ```
   gem uninstall stacktracer
   ```

3. 診断データ・コレクターをインストールします。gem install --local *install_dir*/lx8266/km/bin/stacktracer-*version*.gem と入力します。ここで、*version* はバージョン番号であり、*install_dir* は Ruby エージェントのインストール・ディレクトリーです。エージェント・インストール・ディレクトリー内の stacktracer-*version*.gem ファイルの名前にあるバージョン番号は、ここに入力する必要があるバージョン番号を示しています。デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。

   重要 : Ruby on Rails アプリケーションをインストールして実行する際と同じユーザーとして、データ・コレクターをインストールします。
4. アプリケーションのホーム・ディレクトリーにナビゲートし、その Gemfile を開いて、ファイルの末尾に gem 'stacktracer', '*version*' という行を追加します。
   ここで、*version* は データ・コレクターのバージョン番号です。このバージョン番号は、Ruby エージェントの *install_dir* 内にある stacktracer-*version*.gem ファイルの名前に示されています。
   例えば、Ruby データ・コレクターのバージョン 1.0 フィックスパック 8 をインストールする場合は、エージェントのインストール・ディレクトリーに stacktracer-01.00.08.00.gem ファイルがあります。次に、gem 'stacktracer', '01.00.08.00' 行をアプリケーションに追加して、データ・コレクターをインストールします。

   注 : 環境に stacktracer のバージョンが 1 つしか存在しない場合は、gem 'stacktracer' という行をファイルの末尾に追加します。この行でバージョン番号を指定しないでください。
5. アプリケーションのホーム・ディレクトリーで bundle install と入力します。
6. Ruby on Rails アプリケーションを再始動します。

タスクの結果
データ・コレクターがインストールおよび構成され、Ruby on Rails アプリケーションが開始されます。

次のタスク

- ログインしていない場合は、977 ページの『Cloud APM コンソールの始動』の手順に従ってください。
  「**パフォーマンス**」> **Application Performance ダッシュボード**を選択して、「**すべてのマイ・アプリ**

ケーション」ダッシュボードを開きます。Ruby アプリケーション・リソース・モニター・ダッシュボードおよび診断ダッシュボードにドリルダウンして、状況要約から個々の要求インスタンスに至るまで Ruby on Rails アプリケーションを監視します。

- 診断データ・コレクターの設定を表示および変更するには、次のトピック 722 ページの『診断データ・コレクターの構成』に進んでください。
- Cloud APM UI で要求のメソッド・トレース・データを表示するには、『メソッド・トレースの有効化およびパス表示の調整』を参照してください。
- メソッド・トレースが有効になっているか、データ要求が大きい場合、「`out of memory`」エラーを受け取ることがあります。このようなエラーを回避するために、JVM ヒープ・サイズを増やすことができます。『JVM ヒープ・サイズの増加』を参照してください。
- Cloud APM コンソール を使用して、1 つ以上の管理対象 Ruby on Rails アプリケーションで診断データの収集をいつでも無効または有効にすることができます。726 ページの『Ruby アプリケーション用の診断データの無効化または有効化』を参照してください。この機能はリソース・モニターでは使用できません。

診断データ・コレクターの構成

Cloud APM, Advanced のユーザーの場合、診断データ用にデータ・コレクターを引き続き構成できます。デフォルトでは、診断データ収集はデータ・コレクター構成ファイルで無効になっています。

始める前に

診断データ・コレクターをインストールして、診断データを収集するためのサポートを構成しておく必要があります。説明については、721 ページの『データ・コレクターのインストール』を参照してください。

このタスクについて

エージェントが `Gemfile` を適切に構成して Ruby on Rails アプリケーションの存在を登録した後に、`instrumenter_settings.rb` 構成ファイルが表示されます。 この構成ファイルは Ruby エージェントの実行中に変更でき、変更内容は自動的に反映されます。あるいは、モニター対象のすべての Ruby on Rails アプリケーションに変更内容を適用できます。そのためには、構成ファイルの編集中にアプリケーションが停止している必要があります。

手順

- 実行中の特定アプリケーションのデータ・コレクター設定を変更するには、以下のようにします。
  1. *`install_dir`*`/install-images/kkm/dchome/`*`appClassName`*`/config` ディレクトリーに移動します。 ここで、*`appClassName`* は Ruby アプリケーションのクラス名であり、*install_dir* は Ruby エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。
  2. テキスト・エディターで `instrumenter_settings.rb` を開きます。
  3. データ・コレクター設定を以下のように変更します。

     :instrumentation_enabled
     診断データを収集するためのサポートを有効にするには、`:instrumentation_enabled => true` を設定します。

     診断データを収集するためのサポートを無効にするには、`:instrumentation_enabled => false` を設定します。

     :sample_frequency
     要求のサンプリング頻度を変更するには、あるサンプリングから次のサンプリングまでの要求の数を入力します。

     データ・コレクターは、サンプリングされた要求についてのみ診断データを収集します。 例えば、`:sample_frequency => 10` を設定すると、10 回の要求ごとに 1 回データが収集されます。

:max_methods_to_instrument
メソッド・トレースの収集を無効にするには、値を 0 に設定します。メソッド・トレースの収集を有効にして、トレースされるメソッドの数を制限するには、トレースするメソッドの最大数を入力します。

メソッド・トレースの収集を無効にするには、`:max_methods_to_instrument => 0` を設定します。

メソッド・トレースの収集を有効にするには、`:max_methods_to_instrument => 10000` を設定します。この値をさらに大きくすることができますが、非常に大きな値はパフォーマンスの問題の原因になる可能性があります。メソッド・データが収集されると、メソッドへの呼び出しは要求トレース・ダッシュボードのメソッド・トレース・ウィジェットに表示されます。このウィジェットは、すべての要求インスタンスとそれらのインスタンスのネストされた要求を表示します。

:min_wallclock_to_include_in_trace
要求またはメソッドをトレースするかどうかを決定するしきい値を変更するには、最小応答時間を設定します。例えば、`:min_wallclock_to_include_in_trace => 0.001` を設定した場合は、応答時間が 1 ミリ秒より長い要求およびメソッドのみがトレースされます。

要確認 : 要求トレース診断ダッシュボードでは、要求スタック・トレース・グループ・ウィジェットから特定の要求インスタンスにドリルダウンできます。`:min_wallclock_to_include_in_trace` および `:min_wallclock_to_include_stacks` で設定されたフィルターによって、一部のデータが除外される場合があるため、インスタンスの応答時間の合計は不正確である可能性があります。

:min_wallclock_to_include_stacks
要求またはメソッドのスタック・トレース情報を収集するかどうかを決定するしきい値を変更するには、最小応答時間を設定します。

例えば、`:min_wallclock_to_include_stacks => 0.1` を設定した場合は、応答時間が 100 ミリ秒より長いすべての要求およびメソッドのスタック・トレース情報が収集されます。

:include_subclasses_of_these_modules
要求トレース診断ダッシュボードは、要求インスタンスのネストされた要求とメソッドに対する呼び出し順序を識別するために役立ちます。データ・コレクターは、フィルター・リストに含まれていないクラスからのメソッドを、プリエンプティブにフィルタリングして 除外します。トレースしたい操作がメソッド・スタック・トレースに含まれていない場合は、ここで追加できます。

トレースするメソッドを指定するには、それらのメソッドのクラス名を追加します。

例えば、以下のような Ruby コードで Moped API をトレースするとします。

```
session = Moped::Session.new(['ip:27017'])
session.use(:HR)
session[:profiles].insert({......})
session[:profiles].find({...}).remove
```

これらの Moped API のモジュール名をプロパティーに追加します。

```
:include_subclasses_of_these_modules => {"
ActionController" => true,
                     "ERB" => true,
                     "erb" => true,
                     "Arel" => true,
                     "Mongoid" => true,
                     "Moped" => true
                    },
```

制約事項 : メソッド・トレースには、クラス・メソッドとプライベート・メソッド (暗黙的または明示的な "private" アクセス指定子があるクラス内で定義されたメソッド) は含まれません。

:include_sql_text
メソッドのコンテキスト・データを収集するには、このプロパティーを `true` に設定します。

:num_samples_per_file
各ファイルに保管するトレース済み要求の最大数を変更するには、`:num_samples_per_file => 1000` などの値を入力します。ここで設定された制限に達すると、新規ファイルが作成されます。

構成を調整してより多くのデータを収集する場合は、`:num_samples_per_file` の値を減らすことをお勧めします。例えば、トレースするクラスとメソッドの 数を増やすように `:include_subclasses_of_these_modules` の値を設定すると、収集されるデータの量を増やすことができます。また、`:sample_frequency`、`:min_wallclock_to_include_in_trace`、`:min_wallclock_to_include_stacks` のいずれかのプロパティー値を減らした場合も、収集されるデータの量が増えます。

:verbose_request_instrumentation
:verbose_class_instrumentation
:verbose_method_instrumentation
診断データ・コレクターのロギング・レベルを高くするには、これらのプロパティーを `true` に設定します。

ヒント: トレースしたい特定の操作がメソッド・スタック・トレースに含まれていない場合は、`:verbose_class_instrumentation => true` を設定し、トレースしたいクラスが装備されているかどうかをログで確認してください。装備されていない場合は、そのクラスのモジュール名のクラス名を `:include_subclasses_of_these_modules` プロパティーに追加します。

4. 以下のいずれかのプロパティーを編集した場合は、対応する Ruby on Rails アプリケーションを再始動して、変更内容を有効にします。

```
:include_subclasses_of_these_modules
:max_methods_to_instrument
```

これらのプロパティーは、アプリケーションが起動されるときにのみ使用されて、Ruby データ・コレクターがどのクラスまたはメソッドを 装備するかを決定するため、再始動が必要です。

- すべての Ruby on Rails アプリケーションのデータ・コレクター設定を変更するには、以下のステップを実行します。

1. 現在実行中の Ruby on Rails アプリケーションをすべて停止します。
2. *install_dir*/install-images/kkm/dchome/*application_name*/config ディレクトリーから instrumer_settings.rb を削除します。
3. *Gem_dir*/gems/stacktracer-*version*/config/instrumenter_settings_template.rb 内のデータ・コレクターの設定を変更します。ここで、*version* はバージョン番号 (01.00.05.00 など) であり、*Gem_dir* は stacktracer-version.gem のインストール・ディレクトリー (/usr/local/rvm/gems/ruby-2.1.4/ など) です。詳しくは、特定のアプリケーションのデータ・コレクター設定の変更に関する手順のステップ 722 ページの『3』 を参照してください。
4. 現在実行中の Ruby on Rails アプリケーションをすべて再始動します。

タスクの結果
指定された実行中のアプリケーションまたはすべてのアプリケーションに 関して、診断データ・コレクターの構成が変更されました。

メソッド・トレースの有効化およびパス表示の調整
IBM Cloud Application Performance Management, Advanced と診断データにより、ユーザーは「**要求トレース**」ダッシュボードを利用できます。メソッド・データが収集されると、メソッドへの呼び出しが表示されます。 **メソッド・トレース**・ウィジェットには、要求インスタンスとそのインスタンスのネストされた要求が表示されます。メソッド・トレースを有効にして、メソッドに対する呼び出しをネストされた要求に含めることができます。各ファイル・パスのデフォルトである 50 文字よりも多くの文字数を表示するように、**要求スタック・トレース**・ウィジェットの構成を調整することもできます。

このタスクについて

デフォルトでは、メソッド・トレースが無効になっています。**要求トレース**・ダッシュボードに表示するためにメソッド・トレースを有効にするには、最初の手順を実行します。メソッド・トレースは、構成ファイル内の設定を変更することにより有効にすることができます。

要求スタック・トレース・ウィジェット内のファイル・パスで表示される文字数を調整するには、2 番目の手順を実行します。

手順

- メソッド・トレースを有効にするには、`instrumenter_settings.rb` 設定を編集します。
  a) Ruby エージェント インストール済み環境で `instrumenter_settings.rb` ファイルを探します。 例えば、このファイルは *install_dir*`/install-images/kkm/dchome/`*appClassName*`/config` にあります。 ここで、*appClassName* は Ruby アプリケーションのクラス名であり、*install_dir* は Ruby エージェントのインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。
  b) テキスト・エディターで `instrumenter_settings.rb` を開きます。
  c) 以下のプロパティーを 10000 に設定します。

  ```
  max_method_to_instrument
  ```

  この値をさらに大きくすることができますが、非常に大きな値はパフォーマンスの問題の原因になる可能性があります。(725 ページの『JVM ヒープ・サイズの増加』も参照してください。)
  d) Ruby on Rails アプリケーションを再始動して、メソッド・データ収集を開始します。
  すべての `instrumenter_settings.rb` プロパティーについて詳しくは、722 ページの『診断データ・コレクターの構成』を参照してください。
- **要求スタック・トレース**・ウィジェット内のファイル・パス表示サイズを調整するには、`dfe.properties` ファイルを編集します。
  a) Ruby エージェント インストール済み環境内で `dfe.properties` ファイルを見つけます。例えば、*install_dir*`/lx8266/km/bin/dfe.properties` です。
  ここで、*install_dir* は Ruby エージェント のインストール・ディレクトリーです。デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。
  b) テキスト・エディターで `dfe.properties` を開きます。
  c) 以下のプロパティーの値を調整して、各スタック・トレース・エレメントに表示されるファイル・パスの最大サイズを変更します。

  ```
  dfe.stacktrace.filepath.maxsize
  ```

  d) Ruby エージェントを再始動します。

JVM ヒープ・サイズの増加

Ruby 診断の**要求トレース**・ダッシュボードのメソッド・トレースが有効な場合、またはデータ要求が非常に大きい場合、メモリー不足エラーを回避するために、JVM ヒープ・サイズを増やすことができます。

このタスクについて

Ruby エージェント は Java ベースのエージェントであり、デフォルトの JVM ヒープ・サイズは 384 MB です。これらのステップを実行してヒープ・サイズを増やすことにより、メモリー不足状態になる可能性を減らします。メモリー不足状態が発生する可能性があるのは、頻繁な大容量データ要求があった場合や、メソッド・トレースがオンになっている場合です。

手順

1. Ruby エージェントのインストール・ディレクトリー内で、JVM ヒープ・サイズ設定を見つけます。例えば、*install_dir*`/lx8266/km/bin/runDeepDiveClient.sh` などです。

ここで、*install_dir* は Ruby エージェント のインストール・ディレクトリーです。 デフォルトのインストール・ディレクトリーは、`/opt/ibm/apm/agent` です。

デフォルト値は `-Xmx384m` です。

2. この値を例えば 1024 MB に増やします (`-Xmx1024m` と表記)。
```
export JAVA_OPT="-Djlog.common.dir=$CANDLEHOME/logs -DCONFIG_DIR=
$DC_RUNTIME_DIR -Dkqe.cache.interval=60 -Xmx1024m -Dkqe.timespan=900 -
Djlog.propertyFileDir.CYN=$CANDLEHOME/$ITM_BINARCH/$PRODUCT_CODE/bin"
```
3. Ruby エージェントを再始動します。

Ruby アプリケーション用の診断データの無効化または有効化

IBM Cloud Application Performance Management, Advanced がある場合、いつでも、Cloud APM コンソールの「**エージェント構成**」ページを使用して、1 つ以上の管理対象システムに対して診断データ収集を無効または有効にすることができます。

始める前に

- Cloud APM, Advanced が現在の環境内にある必要があります。
- 仮想マシン上に Monitoring Agent for Ruby をインストールし、構成する必要があります (128 ページの『エージェントのインストール』 (AIX システム 上) または 137 ページの『エージェントのインストール』 (Linux システム 上)、および 718 ページの『Ruby モニターの構成』の説明に従います)。
- 721 ページの『データ・コレクターのインストール』の説明に従って、診断データ・コレクターをインストールし、診断データを収集するためのサポートを構成する必要があります。

このタスクについて

データ・コレクター構成で診断データをサポートするように構成すると、デフォルトで管理対象システムごとに診断データの収集が無効になります。診断ダッシュボードにデータを表示するには、モニターされる管理対象システムごとに診断データの収集を有効にする必要があります。

管理対象のシステムごとに診断データの収集を有効または無効にするには、以下のステップに従ってください。

手順

1. ナビゲーション・バーで、 「**システム構成**」**->**「**エージェント構成**」を選択します。
   「**エージェント構成**」ページが表示されます。
2. 「**Ruby**」タブをクリックします。
3. 診断データの収集を有効または無効にする管理対象システムのチェック・ボックスを選択します。
4. 「**アクション**」リストから、以下のいずれかのオプションを選択して、選択した管理対象システムの診断データ収集を無効または有効にします。
   - **「データ収集の無効化」**を選択します。選択された管理対象システムごとに、「データ・コレクターの有効化 (`Data Collector Enabled`)」列の状況が「いいえ」に更新されます。
   - **「データ収集の有効化」**を選択します。選択された管理対象システムごとに、「データ・コレクターの有効化 (`Data Collector Enabled`)」列の状況が「はい」に更新されます。

タスクの結果

選択した管理対象システムごとに診断データの収集が構成されました。

IBM Cloud アプリケーション用の Ruby データ・コレクターの構成

IBM Cloud 上の Ruby アプリケーションに関する情報を収集するには、Ruby データ・コレクターを構成する必要があります。

始める前に

1. IBM Marketplace からデータ・コレクター・パッケージをダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。

手順

1. データ・コレクター・パッケージからファイルを抽出します。ruby_datacollector_8.1.4.0.tgz パッケージは、解凍されたディレクトリーに含まれています。
2. 以下のコマンドを実行して、ruby_datacollector_8.1.4.0.tgz 内のファイルを解凍します。

   ```
   tar -zxf ruby_datacollector_8.1.4.0.tgz
   ```

   ibm_ruby_dc フォルダーが得られます。
3. 以下のコマンドを実行して、ibm_ruby_dc 内の etc フォルダー全体を、ご使用の Ruby アプリケーションのルート・フォルダーにコピーします。

   ```
   cp –r directory to the etc folder  home directory of your Ruby application
   ```

   以下のコマンドは、データ・コレクターを /opt/ibm/ccm/ibm_ruby_dc/etc ディレクトリーに解凍します。Ruby アプリケーションのホーム・ディレクトリーは /root/ruby_app/ です。

   ```
   cp –r /opt/ibm/ccm/ibm_ruby_dc/etc /root/ruby_app/
   ```
4. Ruby アプリケーションのホーム・フォルダーにある Gemfile に以下のセクションを追加します。

   ```
   gem 'logger',  '>= 1.2.8'
   source 'https://maagemserver.ng.bluemix.net/' do
     gem 'ibm_resource_monitor'
     gem 'stacktracer'
   end
   ```
5. bundle lock コマンドを実行して、Gemfile.lock ファイルを再生成します。
6. Ruby アプリケーションの manifest.yml ファイルを含むディレクトリーから、以下のコマンドを実行します。

   ```
   cf push
   ```

   ヒント: サンプルの manifest.yml ファイルについては、192 ページの『サンプル manifest.yml ファイル』を参照してください。

タスクの結果

データ・コレクターが構成され、Cloud APM サーバーに接続されました。

次のタスク

ご使用の IBM Cloud アプリケーションのモニター・データが Cloud APM コンソールに表示されることを確認できます。Cloud APM コンソールを開始する方法の説明については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

IBM Cloud アプリケーション用の Ruby データ・コレクター のカスタマイズ

IBM Cloud ユーザー・インターフェース (UI) で環境変数を追加して、ご使用の IBM Cloud アプリケーションのモニターをカスタマイズできます。以下の情報を使用して、ニーズに応じて変数を追加します。

Ruby データ・コレクター用のユーザー定義の環境変数

以下の表の情報を使用して、IBM Cloud での Ruby モニターをカスタマイズできます。

表 *198. IBM Cloud* での *Ruby* モニター用にサポートされるユーザー定義の環境変数

変数名	重要度	値	説明
APM_BM_GATEWAY_URL	オプション	• https://*<server ip* または *hostname>*:443 • http://*<server ip* または *hostname>*:80	ターゲットのオンプレミス・サーバー・ゲートウェイ URL。
APM_KEYFILE_PSWD	オプション	鍵ファイルの暗号化されたパスワード	鍵ファイルと対になっている暗号化された鍵ファイル・パスワード。Linux ユーザーの場合は、`echo -n <keyfile password> | base64` コマンドを使用してパスワードを暗号化できます。 **注 :** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。
APM_KEYFILE_URL	オプション	http://*<hosted http server>*:*<port>*/keyfile.p12	鍵ファイルをダウンロードするための URL。 **注 :** この変数を設定するのは、HTTPS を使用するようにゲートウェイを構成した場合に限ります。
kkm_instrumentation_enabled	オプション	• true • false	診断データの収集を有効または無効にします。 true: 値を `true` に設定すると、診断データが収集されます。 false: 値を `false` に設定すると、診断データは収集されません。 デフォルト値は `true` です。

表 198. *IBM Cloud* での *Ruby* モニター用にサポートされるユーザー定義の環境変数 (続き)

変数名	重要度	値	説明
kkm_max_methods_to_instrument	オプション	トレースされるメソッドの最大数	トレースされるメソッドの最大数。 値を 0 に設定するとメソッド・トレースを無効にできます。 デフォルトでは、値は 10000 に設定され、メソッド・トレースが有効になっています。 **注:** 10000 より大きい値を設定しないことをお勧めします。10000 よりもはるかに大きい値を設定すると、アプリケーションの実行効率が低下する可能性があります。
kkm_sample_frequency	オプション	要求のサンプリング頻度	1 件のサンプル要求の取得元になる要求数。例えば、値を 10 に設定した場合、10 件の要求ごとに 1 件のモニター・データが収集されます。 デフォルト値は 10 です。
kkm_min_wallclock_to_include_in_trace	オプション	メソッド・トレースを収集するための応答時間しきい値 (秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのメソッド・トレースを収集します。例えば、0.001 に設定した場合、応答時間が 1 ミリ秒より長い要求およびメソッドがトレースされます。 デフォルト値は 0 です。これは、すべての要求およびメソッドのメソッド・トレースが有効であることを意味します。
kkm_min_wallclock_to_include_stacks	オプション	スタック・トレースを収集するための応答時間しきい値 (秒単位)	要求インスタンスの応答時間がこの変数の値を超過した場合、データ・コレクターはそのスタック・トレースを収集します。例えば、0.001 に設定した場合、応答時間が 1 ミリ秒より長い要求およびメソッドがトレースされます。 デフォルト値は 0 です。これは、すべての要求およびメソッドのスタック・トレースが有効であることを意味します。

IBM Cloud アプリケーション用の Ruby データ・コレクターの構成解除

Ruby 環境をモニターする必要がない場合、または Ruby データ・コレクターをアップグレードする 場合は、最初に Ruby データ・コレクターの以前の設定を構成解除する必要があります。

手順

1. アプリケーションのルート・フォルダーに移動します。
2. Ruby アプリケーションのホーム・フォルダーにある `Gemfile` から以下の行を削除します。

```
gem 'logger',  '>= 1.2.8'
source 'https://maagemserver.ng.bluemix.net/' do
  gem 'ibm_resource_monitor'
  gem 'stacktracer'
end
```

3. `bundle lock` コマンドを実行します。
4. アプリケーションのホーム・ディレクトリーから、以下のコマンドを実行して、アプリケーションを IBM Cloud に再プッシュし、変更を有効にします。

```
cf push
```

タスクの結果

Ruby データ・コレクターが正常に構成解除されました。

次のタスク

データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

SAP モニターの構成

SAP システムをモニターするには、モニター対象システム内のアプリケーション・サーバーに Monitoring Agent for SAP Applications を接続して、エージェントが製品付属の Advanced Business Application Programming (ABAP) コードにアクセスできるようにする必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for SAP agent を参照してください。
- SAP エージェントは、非 Unicode SAP システムをサポートしません。

このタスクについて

SAP エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

- Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
  - 731 ページの『Windows システムでのエージェントの 構成』
  - 733 ページの『サイレント応答ファイルを使用したエージェントの構成』
- Linux システムまたは AIX システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。
  - 732 ページの『Linux または AIX システム 上でのエージェントの構成』
  - 733 ページの『サイレント応答ファイルを使用したエージェントの構成』

SAP エージェントをインストールした後、Advanced Business Application Programming (ABAP) トランスポートを SAP システムにインポートして、SAP システムでのデータ収集をサポートすることができます。詳しくは、737 ページの『SAP システムでの ABAP トランスポートのインポート』を参照してください。

SAP エージェントを構成したら、エージェント構成を検証する必要があります。詳しくは、744 ページの『エージェント構成の検証』を参照してください。

SAP エージェントを構成した後、OSLC (Open Source Lifecycle Collaboration) 準拠に必要なデータベース通信ポート番号を追加できます。詳しくは、748 ページの『データベース通信ポート番号の追加』を参照してください。

SAP システムから ABAP トランスポートを削除するには、SAP システムに DELETE トランスポートをインポートする必要があります。 詳しくは、743 ページの『SAP システムからの ABAP トランスポートの削除』を参照してください。

新しい CCMS 設計はデフォルトで有効になっています。値が YES に設定された isnewccmsdesign パラメーターに対する項目がデータベース表 /IBMMON/ITM_CNGF に存在します。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

Windows システムでのエージェントの構成

エージェントがモニター対象の SAP アプリケーション・サーバーのデータを収集できるように、「**IBM Performance Management**」ウィンドウを使用して Windows システム上に SAP エージェントを構成することができます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**デフォルトを使用して構成**」をクリックします。

   「**Monitoring Agent for SAP Applications**」ウィンドウが開きます。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。

   重要 : エージェント・インスタンス名は、管理対象の SAP アプリケーション・サーバーの 3 桁のシステム ID (SID) と一致しなければなりません。例えば、管理対象の SAP アプリケーション・サーバーの SID が PS1 の場合、インスタンス名として PS1 と入力します。
4. アプリケーション・サーバー・モードまたはログオン・グループ・モードのいずれかで、SAP エージェントを構成します。
   - アプリケーション・サーバー・モードで SAP エージェントを構成するには、以下の手順を実行します。
     a. 「**接続モード**」フィールドで、「**アプリケーション・サーバー・モード**」を選択し、「**次へ**」をクリックします。
     b. 「**アプリケーション・サーバーの情報を指定します**」領域で、構成パラメーターの値を指定し、「**次へ**」をクリックします。
     c. 「**SAP システムに対するログオン情報を指定します**」領域で、構成パラメーターの値を指定し、「**OK**」をクリックします。

     構成パラメーターについては、734 ページの『エージェントの構成パラメーター』を参照してください。
   - ログオン・グループ・モードで SAP エージェントを構成するには、以下の手順を実行します。
     a. 「**接続モード**」フィールドで、「**ログオン・グループ・モード**」を選択し、「**次へ**」をクリックします。

b.「**ログオン・グループの情報を指定します**」領域で、構成パラメーターの値を指定し、「**次へ**」をクリックします。

c.「**SAP システムに対するログオン情報を指定します**」領域で、構成パラメーターの値を指定し、「**OK**」をクリックします。

構成パラメーターについては、734 ページの『エージェントの構成パラメーター』を参照してください。

重要 : アプリケーション・サーバー・モードの場合、メッセージ・サーバーまたは ASCS が構成されている SAP システム上でディスパッチャーがあるダイアログ・インスタンスを構成することが必須です。ログオン・グループ・モードの場合、メッセージ・サーバーまたは ASCS が構成されている SAP システム上でディスパッチャーがあるダイアログ・インスタンスを構成することは必須ではありません。

5.「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

重要 : SAP エージェントの別のインスタンスを作成する場合は、ステップ 1 から 6 を繰り返します。作成する SAP エージェント・インスタンスそれぞれに固有のシステム ID を使用してください。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。
- 定義済みの R3_Alert_Crit シチュエーションおよび R3_Alert_Warn シチュエーションを編集して、「アラート状況」属性の条件を `Alert Status!= DONE` に設定し、クローズされた CCMS アラートに対してこれらのシチュエーションがトリガーされないようにする必要があります。

Linux または AIX システム上でのエージェントの構成

エージェントがモニター対象の SAP アプリケーション・サーバーのデータを収集できるように、Linux システムまたは AIX システム上に SAP エージェントを構成することができます。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
   例: `/opt/ibm/apm/agent/bin`
2. 以下のコマンドを実行します。ここで、instance_name は、インスタンスに付ける名前です。
   `./sap-agent.sh config` *`instance_name`*

   重要 : エージェント・インスタンス名は、管理対象の SAP アプリケーション・サーバーの 3 桁のシステム ID (SID) と一致しなければなりません。例えば、管理対象の SAP アプリケーション・サーバーの SID が PS1 の場合、インスタンス名として PS1 と入力します。
3. コマンド行に次のメッセージが表示されたら、「1」を入力して Enter キーを押します。 `Edit 'Monitoring Agent for SAP Applications' setting? [1=Yes, 2=No]`
4. アプリケーション・サーバー・モードまたはログオン・グループ・モードを使用して SAP エージェントを構成します。
   - アプリケーション・サーバー・モードで SAP エージェントを構成するには、以下の手順を実行します。
     a. コマンド・ラインに「`Connection Mode [ 1=Application Server Mode, 2=Logon Group Mode ]`」というメッセージが表示されたら、「1」を入力して Enter キーを押します。
     b. 構成パラメーターの値を指定します。

     構成パラメーターについては、734 ページの『エージェントの構成パラメーター』を参照してください。
   - ログオン・グループ・モードで SAP エージェントを構成するには、以下の手順を実行します。

a. コマンド・ラインに「Connection Mode [1=Application Server Mode, 2=Logon Group Mode]」というメッセージが表示されたら、「2」を入力して Enter キーを押します。

b. 構成パラメーターの値を指定します。

構成パラメーターについては、734 ページの『エージェントの構成パラメーター』を参照してください。

重要 : アプリケーション・サーバー・モードの場合、メッセージ・サーバーまたは ASCS が構成されている SAP システム上でディスパッチャーがあるダイアログ・インスタンスを構成することが必須です。ログオン・グループ・モードの場合、メッセージ・サーバーまたは ASCS が構成されている SAP システム上でディスパッチャーがあるダイアログ・インスタンスを構成することは必須ではありません。

5. 次のコマンドを実行して、SAP エージェントを始動します。

```
./sap-agent.sh start instance_name
```

重要 : SAP エージェントの別のインスタンスを作成する場合は、手順 1 から 5 を繰り返します。作成する SAP エージェント・インスタンスごとに固有のシステム ID を使用してください。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。
- 定義済みの R3_Alert_Crit シチュエーションおよび R3_Alert_Warn シチュエーションを編集して、「アラート状況」属性の条件を Alert Status!= DONE に設定し、クローズされた CCMS アラートに対してこれらのシチュエーションがトリガーされないようにする必要があります。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルを使用して、SAP エージェント を Windows システム、Linux システムまたは、AIX システムで構成することができます。

手順

1. テキスト・エディターで、*install_dir*¥samples のパスにある sap_silent_config.txt ファイルを開き、すべての構成パラメーターの値を指定します。

   Windows C:¥IBM¥APM¥samples

   Linux AIX /opt/ibm/apm/agent/samples

   構成パラメーターについては、734 ページの『エージェントの構成パラメーター』を参照してください。

2. コマンド行で、パスを次の bin ディレクトリーに変更します。

   Windows *install_dir*¥BIN

   Linux AIX *install_dir*¥bin

3. 次のコマンドを実行します。

   Windows
   ```
   sap-agent.bat config instance_name install_dir¥samples
   ¥sap_silent_config.txt
   ```
   Linux AIX
   ```
   sap-agent.sh config instance_name install_dir¥samples
   ¥sap_silent_config.txt
   ```

   重要 : エージェント・インスタンス名は、管理対象の SAP アプリケーション・サーバーの 3 桁のシステム ID (SID) と一致しなければなりません。例えば、管理対象の SAP アプリケーション・サーバーの SID が PS1 の場合、インスタンス名として PS1 と入力します。

4. エージェントを開始します。

   Windows 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

Linux AIX 次のコマンドを実行します: `./sap-agent.sh start instance_name`

重要 : SAP エージェントの別のインスタンスを作成する場合は、手順 1 から 4 を繰り返します。作成する SAP エージェント・インスタンスごとに固有のシステム ID を使用してください。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。
- 定義済みの R3_Alert_Crit シチュエーションおよび R3_Alert_Warn シチュエーションを編集して、「アラート状況」属性の条件を `Alert Status!= DONE` に設定し、クローズされた CCMS アラートに対してこれらのシチュエーションがトリガーされないようにする必要があります。

エージェントの構成パラメーター

SAP エージェント を構成する際、「SAP ホスト名」や「SAP システム番号」などのパラメーターのデフォルト値を変更できます。

以下の表で、SAP エージェントの構成パラメーターについて詳しく説明します。

表 *199. SAP* エージェントの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド	例
SAP ホスト名 (1 次)	エージェントの接続先の SAP アプリケーション・サーバーのホスト名。SAP サーバーが専用 LAN 経由で通信する場合は、 これらのサーバーをホストするコンピューターに 複数のネットワーク・カードがあります。ホスト名には、SAPGUI ログオンなどで、外部システムからアプリケーション ・サーバーにアクセスするための名前を入力します。専用 LAN のホスト名は 使用しないでください。デフォルト値は、エージェントのインストール先ホスト名です。	はい	saphost.domain.com
SAP システム番号 (1 次)	SAP ホスト・サーバーに接続するために使用する 2 桁の SAP システムまたはインスタンスの 番号。デフォルト値は 00 です。	はい	
SAP ホスト名 (代替 1)	1 次ホストが使用不可の場合の、第 2 候補のホスト名。	いいえ	
SAP システム番号 (代替 1)	最初の代替のホスト名のシステム番号。	いいえ	
SAP ホスト名 (代替 2)	SAP ホスト名 (1 次) および SAP ホスト名 (代替 1) が両方とも使用不可の場合の、第 3 候補のホスト名。	いいえ	
SAP システム番号 (代替 2)	2 番目の代替のホスト名のシステム番号。	いいえ	
SAP クライアント番号	SAP への RFC ログオンのための SAP クライアント番号。デフォルト値は 000 です。ABAP によって生成された IBMMON_AGENT ユーザーを使用する場合は、トランスポート・インポートで指定されたクライアント番号を入力します。この番号は、プロファイルにおける nnn のクライアント番号と同じです。	はい	

表 199. SAP エージェントの構成パラメーターの名前と説明 (続き)

パラメーター名	説明	必須フィールド	例
SAP ユーザー ID	SAP への RFC ログオンのための SAP ユーザー ID。デフォルト値は IBMMON_AGENT です。これは、インポート中に作成される事前定義のユーザー ID です。	はい	
SAP ユーザー・パスワード	デフォルトのパスワードを使用するか、別のパスワードを指定します。	はい	
SAP ユーザーのパスワードの確認	「**SAP ユーザー・パスワード**」フィールドに指定したパスワード。	はい	
SAP 言語コード	SAP システムに接続するときにエージェントが使用する言語を示す言語コード。指定された言語によって、アラート・メッセージ、syslog メッセージ、ジョブ・ログ・メッセージなどの SAP 情報が表示されるときの言語が決まります。 SAP システムは、すべて英語およびドイツ語で納入されます。別の言語が必要な場合は、その言語が SAP システムにインストールされているかどうか SAP 管理者に確認してください。サポートされない言語を指定すると、エージェントが SAP システムに接続できません。 以下の言語およびコードがサポートされます。 • CS - チェコ語 • EN - 英語 • FR - フランス語 • DE - ドイツ語 • HU - ハンガリー語 • IT - イタリア語 • ES - スペイン語 • JA - 日本語 • KO - 韓国語 • PL - ポーランド語 • PT - ポルトガル語 • RU - ロシア語 • ZH - 中国語 • ZF - 中国語 (繁体字)	はい	
RFC トレース	*SAPTRACE* 変数のリモート関数呼び出し (RFC) トレース設定。このチェック・ボックスを選択すると、RFC トレースがアクティブになります。デフォルト値は、RFC トレースを実行しないことです。コマンド・ラインの場合は、2 がトレースなし、1 がトレースありです。RFC トレースを実行すると広範囲にわたる診断情報が生成されるため、このトレース機能は慎重に使用してください。RFC トレースについて詳しくは、IBM サポートにお問い合わせください。	いいえ	

表 199. *SAP* エージェントの構成パラメーターの名前と説明 (続き)

パラメーター名	説明	必須フィールド	例
SAP ログオン・グループ	SAP サーバーのログオン・グループの名前。	はい	
SAP メッセージ・サーバー名	SAP メッセージ・サーバーのホスト名。	はい	
SAP メッセージ・サービス	SAP メッセージ・サーバーが配置されているサービスの名前。 以下のオペレーティング・システム・サービス・ファイルにサービス名を含める必要があります。 • /etc/services • ¥windows¥system32¥drivers¥etc ¥services	はい	メッセージ・サービス名 sapmsTV1 を使用することも、完全なメッセージ・サービス・ポート番号 3601 を使用することもできます。
SAP 経路ストリング	SAP ルーターを持つ SAP サーバーにアクセスする場合は、SAP ルーター・ストリングを指定します。	いいえ	ルーター・ストリング /H/host/H/ は、以下の形式でなければなりません。 /H/beagle/H/brittany/H/ または /H/amsaix11.tivlab.raleigh.ibm.com/W/tivoli/H/amsaix25
SNC	セキュア・ネットワーク・コミュニケーション (SNC) を有効化するか無効化するかを指定します。デフォルト値は無効化です。	はい	**sap_conn.sap_snc_mode** =true または false
SNC セキュリティー・レベル	SNC のセキュリティー・レベル。	はい	**sap_snc_mode1.sap_snc _qop**=*QOP* 値 デフォルト値は 8 です。
クライアントまたはエージェントの SNC 名	クライアントまたはエージェントの SNC 名。	はい	**sap_snc_mode1.sap_snc _client=** クライアントの *SNC* 名
パートナーまたは SAP サーバーの SNC 名	パートナーまたは SAP サーバーの SNC 名。	はい	**sap_snc_mode1.sap_snc _server=** サーバーの *SNC* 名
SAP の暗号ライブラリー・パス	SAP の暗号ライブラリーのパス。	はい	**sap_snc_mode1.sap_snc _library=** 暗号ライブラリーのパス

SAP ホスト名が管理対象システム名の長さ制限に従ってトリムされる

APM コンソールに公開されるリソースの管理対象システム名は、32 文字に制限されます。 SAP エージェントは、管理対象システム名を制限内で形成するために、ドメイン名のトリミングをサポートしています。

シナリオ 1

サブノードが **Sys** タイプである場合、管理対象システム名の形式は次のとおりです。

SID-DBHOST:**Sys**
各部の意味は次のとおりです。

- *SID* は、SAP システム ID です。
- *DBHOST* は、SAP システム・ホスト名です。

例:
SID が「**P27**」で *DBHOST* が「**VPT02F90.mycorporation.co.in**」である場合、形成される管理対象システム名の完全修飾ドメイン名 (FQDN) は「**P27-VPT02F90.mycorporation.co.in:Sys**」になります。

管理対象システム名が 32 文字を超える場合、SAP エージェントはドメイン名をトリムして管理対象システム名「**P27-VPT02F90:Sys**」を形成します。サブノードのトリム済み管理対象システム名は、APM コンソールに公開されます。

注 : 管理対象システム名の長さ (ドメイン名を含む) が 32 文字以下の場合は、管理対象システム名の FQDN はトリムされません。 管理対象システム名の FQDN は、そのまま APM コンソールに公開されます。
管理対象システム名の長さ制限を満たすために必要な場合は、SAP エージェントによって公開されているすべてのサブノード・タイプに、ドメイン名のトリミングを適用できます。

シナリオ 2

エージェント・インスタンスのサブノードが **mySAP** の場合、管理対象システム名の形式は次のとおりです。

$SAPSYSTEMNAME-$dbhost:*$CTIRA_HOSTNAME*:**mySAP**
各部の意味は次のとおりです。

- *$SAPSYSTEMNAME* は、構成中に提供されるエージェント・インスタンス名です。
- *$dbhost* は、SAP システム・ホスト名です。
- *$CTIRA_HOSTNAME* は、エージェント・マシン・ホスト名です。

例:
$SAPSYSTEMNAME が「**SA2**」で、*$dbhost* が「**VPT02F90. mycorporation.co**」で、*$CTIRA_HOSTNAME* が「**mysap1-v27.mycorp.co**」である場合、形成される「管理対象システム名」の完全修飾ドメイン名 (FQDN) は「**SA2-VPT02F90.mycorporation.co.in: mysap1-v27:mySAP**」になります。

注 : エージェント・マシン・ホスト名のデフォルトのドメイン名はトリムされ、「**mysap1-v27**」になります。

管理対象システム名が 32 文字を超えています。最初に、SAP エージェントは SAP システム・ホスト名のドメイン名をトリムして、「**SA2-VPT02F90:mysap1-v27:mySAP**」に形成します。
結果の管理対象システム名が依然として 32 文字を超えている場合、SAP エージェントは、エージェント・マシン・ホスト名の末尾の文字をトリムして、32 文字制限以内の管理対象システム名を形成します。 その後、サブノードの管理対象システム名が APM コンソールに公開されます。

SAP システムでの ABAP トランスポートのインポート

Advanced Business Application Programming (ABAP) トランスポート要求をインポートした SAP システムごとに 1 つの SAP エージェントをインストールして、SAP システムでのデータ収集をサポートすることができます。

始める前に

SAP システムで ABAP トランスポートをインポートする前に、次の前提条件が満たされていることを確認してください。

- 製品のトランスポート要求をインポートするには、R3trans バージョン 01.07.04 以降が必要です。これは、Dynpro、Export、および Import の各テーブルが非互換であるためです。エージェントの基本操作は、Dynpro と、Export と Import の非互換性による影響を受けず、SAP 構成ウィンドウのみが影響を受けます。
- Solution Manager システムをモニターするには、MAI 構成が使用可能なクライアント上に確実に SAP エージェント V7.1.1 トランスポートをインポートする必要があります。PI システムの機能を表示するには、PI 構成が使用可能なクライアント上の PI システムに SAP エージェント V7.1.1 トランスポートをインポートします。
- SLM サブノードの下のグループ・ウィジェットでデータを表示するには、PI および Solution Manager 用の MAI 構成を完了する必要があります。 また、「BPM アラート」グループ・ウィジェットにデータを表示できるようにするために、ビジネス・プロセス・モニタリングも構成する必要があります。「最新のクリティカルおよび優先度の高いアラート」グループ・ウィジェットのデータを表示するには、以下の構成を行います。
  - Solution Manager 7.1 で、SOLMAN_SETUP トランザクションを実行して「**システム・モニタリング**」を選択し、Third-Party Component をアクティブ化または有効化し、「**実装: アラート対応用の BADI 定義 (Implementation: BADI Definition for Alert Reactions)**」とサード・パーティー・コネクターを追加します。
  - スコープ・フィルターを「**すべてのアラートおよびメトリック (All Alerts and Metrics)**」に設定します。
  - 実装状態が「**アクティブ**」であることを確認します。

  必要な SAP サービス・パック・レベルのリストなどについて詳しくは、次の Online Service System (OSS) ノートを参照してください。
  - OSS ノート 454321
  - OSS ノート 330267
  - OSS ノート 743155
- SAP システムをモニターするために、SAP エージェントは SAP 統計データを必要とします。SAP 7.0 システムで、SAP システム時間帯を設定して、オペレーティング・システムの時間帯を突き合わせる必要があります。 これにより SAP 統計が正しいタイム・スタンプで収集されます。 同様に、SAP エージェントがデータを収集できるように、SAP エージェントの SAP システム時間帯を更新します。この問題について詳しくは、SAP ノート 926290 を参照してください。

このタスクについて
SAP トランスポートのインポートについては、739 ページの『SAP トランスポートのインポート』を参照してください。

ABAP トランスポートをインポートするための MAI アラート関連の前提条件
ABAP トランスポートをインポートする前に、MAI アラート関連の前提条件を確認する必要があります。

`transport.prop` ファイル内の構成設定

E メール通知設定を構成せず、BAdi 実装なしで MAI アラートをフェッチするなど、新しい MAI アラート・フェッチ・メカニズムを使用した場合、`transport.prop` ファイルで以下の構成設定を変更する必要があります。

SPLEVEL=X 行を追加します (ここで、X は、Solution Manager システムのサポート・パック (SP) レベルです)。

例えば、システム ID が S10 で、サポート・パック・レベルが 13 の場合は、SPLEVEL=13 を追加します。

重要 : SP レベル 10 以降の SAP システムでは、SAP Solution Manager で E メール通知を構成せず、BAdi 実装なしで MAI アラートがフェッチされている場合、SAP Solution Manager ダッシュボードの「最新の格付け「赤」の MAI アラート」グループ・ウィジェットでテクニカル名 (MEA) 属性の値は取り込まれません。SAP Solution Manager で E メール通知を構成し、BAdi 実装で MAI アラートがフェッチされている場合は、SAP Solution Manager ダッシュボードの「最新の格付け「赤」の MAI アラート」グループ・ウィジェットでテクニカル名 (MEA) 属性の値が取り込まれます。

Solution Manager サポート・パック (SP) レベルに基づいた、新旧の MAI アラート・フェッチ・メカニズムの判別

旧 MAI アラート・フェッチ・メカニズム
このメカニズムは、E メール通知設定の構成、および IF_ALERT_DYN_COFIGURATION インターフェースを備えた /IBMMON/ITM_IMPL_ALRTINBX BAdi 実装に基づいて、MAI アラートを収集して SAP エージェントに送信します。

新 MAI アラート・フェッチ・メカニズム
このメカニズムは、E メール通知設定を構成せず、また IF_ALERT_DYN_COFIGURATION インターフェースを備えた /IBMMON/ITM_IMPL_ALRTINBX BAdi 実装なしで MAI アラートをフェッチする処理に基づいています。

以下の表を使用して、`transport.prop` ファイルの使用法、および E メール通知設定の構成に対するその依存関係を把握できます。

表 *200. transport.prop* ファイルの使用法およびその依存関係

SAP システム SP レベル	transport.prop 設定		E メール通知設定の構成	使用する MAI アラート・メカニズム
	MAI_CONFIGURED	Solution Manager SP レベル		
任意	No、またはファイルが存在しない	適用できません	構成、または構成しない	SLM サブノードは表示されず、代わりに SOL サブノードが表示されます。
SP 6 から 9	はい	記載	構成	旧メカニズム
SP 6 から 9	はい	記載なし	構成	旧メカニズム
SP 6 から 9	はい	記載なし	未構成	旧メカニズムは、E メール通知設定の構成が必須であるため、機能しません。
SP 6 から 9	はい	記載	未構成	旧メカニズムは、E メール通知設定の構成が必須であるため、機能しません。
SP 10 以降	はい	記載	構成	新メカニズム
SP 10 以降	はい	記載	未構成	新メカニズム
SP 10 以降	はい	記載なし	構成	旧メカニズム
SP 10 以降	はい	記載なし	未構成	旧メカニズムは、E メール通知設定の構成が必須であるため、機能しません。

SAP トランスポートのインポート

SAP エージェントは、SAP システムでのデータ収集をサポートする一連の Advanced Business Application Programming (ABAP) ルーチンを提供します。この ABAP コードは SAP トランスポートとして配信され、モニター対象の各 SAP システムにインストールする必要があります。SAP 管理者がトランスポートをインストールします。

このタスクについて

ZITM_610AUTH 許可プロファイルと **ZITM_610AUT** 許可の役割は、6.1 リリースまでに限り有効です。リリース 6.2 以降は、**/IBMMON/AUTH** 許可プロファイルが使用されます。未許可の使用を防ぐために、SAP システムに インストールする ABAP コードは、SAP システム内から 不可視になっています。さらに、このコードは変更も 生成もできません。このコードのサポートを IBM ソフトウェア・サポート Web サイトから入手する必要があります。

ABAP コードのインストールに加え、トランスポートは翻訳されたテキスト要素をインストールし、SAP トランスポートのテキスト要素に多文化サポートを提供します。

重要 : このトランスポートを SAP システムにインポートする 前に、この SAP システムをモニターするように構成された SAP エージェント・インスタンスを開始してはなりません。

SAP トランスポートをインポートすると、ユーザーが SAP システムに暗黙的に定義されます。

次の手順を使用して、SAP トランスポートを SAP システムにインポートします。

手順

1. エージェントがインストールされているコンピューター上で、以下のパスから IBM Tivoli Monitoring トランスポート・ファイルをコピーします。
   - Windows の場合: *install_dir*¥TMAITM6_x64¥ABAP
   - Windows 以外の場合: *install_dir*/*intrp*/sa/ABAP。ここで、*intrp* は、**lx8266** または **aix526** でなければなりません。
2. 以下のトランスポート・ファイルを、ステップ 1 に示されているパスから SAP 環境にコピーします。
   - K711_00xxxU.ITM および R711_00xxxU.ITM

     これらのファイルは Unicode 版の トランスポートです。これらには、SAP エージェント ABAP コードおよび、ローマ字コード・ページと 2 バイト・コード・ページのテキスト・ストリング用の Unicode サポートが含まれています。
   - K711_00xxx_DELETE.ITM および R711_00xxx_DELETE.ITM

     これらのファイルは、ABAP コードを削除します。製品の使用を完全にやめて SAP システムからトランスポートを削除する場合でない限り、DELETE トランスポートをインポートする必要はありません。743 ページの『SAP システムからの ABAP トランスポートの削除』を参照してください。
3. 以下の手順で、トランスポート・ファイルを SAP トランスポート・システムのデータ・ディレクトリーにコピーします。 この際、トランスポート・ファイル名は変更しないでください。

   Unicode トランスポート

   a. K711_00xxxU.ITM ファイルを cofiles ディレクトリーにコピーします

   b. R711_00xxxU.ITM ファイルを data ディレクトリーにコピーします。
4. 単一の IBM Tivoli Monitoring トランスポート・ファイルを SAP システム 上にインストールするために、以下のいずれかのファイル・インポート・オプションを選択します。
   - Solution Manager 7.1 サービス・パック 6 レベル以降の、MAI が構成されている SAP システムでは、transport.prop ファイルを SAP システムの usr/sap/SID/DVEBMGS*instancenumber*/work 作業ディレクトリーに作成する必要があります。SAP システムが ABAP SAP Central Services (ASCS) を使用した分散システムである場合は、transport.prop ファイルをセントラル・インスタンス (CI) の usr/sap/SID ディレクトリーに作成します。 次に、そのファイルに MAI_CONFIGURED = YES 項目を追加します。この項目により、/IBMMON/ITM_CNFG 表に MAI_CONFIGURED = YES 項目が作成されます。 これで、SAP システムで単一の IBM Tivoli Monitoring トランスポート・ファイルをインポートできるようになります。

     注 : 単一の IBM Tivoli Monitoring トランスポート・ファイルをインポートする 前に、transport.prop ファイルを SAP システムの usr/sap/SID/DVEBMGS*instancenumber*/work 作業ディレクトリーに作成し、そのファイルに MAI_CONFIGURED = YES 項目を追加する必要があります。/IBMMON/ITM_CNFG テーブル内の項目は編集しないでください。

- 7.0 以降の基本バージョンと、MAI 構成がない Solution Manager V7.1 を持つその他すべての SAP システムでは、単一の IBM Tivoli Monitoring トランスポート・ファイルを直接インポートする必要があります。

5. 次のコマンドを実行して、SAP トランスポートをインポートします。

```
tp addtobuffer ITMK711_00xxxU SID
pf=¥usr¥sap¥trans¥bin¥PROFILE_NAME
```

各項目の意味は次のとおりです。

SID
ターゲット SAP システム ID。

PROFILE_NAME
tp プロファイル・ファイルの名前。コマンド行からエージェントトランスポート・ファイルをインポートするときは、必ず現行の tp パラメーター・ファイルを指定してください。通常、tp パラメーター・ファイルの名前は TP_DOMAIN_*SID*.PFL になります。UNIX システムでは、このファイル名の大/小文字を区別します。

nnn
エージェントが実行され、ユーザー ID、IBMMON_AGENT、許可プロファイル、および /IBMMON/AUTH が定義されているターゲット・クライアントの番号。

別の方法として、SAP STMS トランザクションを使用して ITMK711_00xxxU.ITM トランスポート要求をインポートすることもできます。「**トランスポート要求のインポート (Import Transport Request)**」ウィンドウの「**インポート・オプション (Import Options)**」タブで、以下のオプションが選択されていることを確認します。

- **後でインポートするためにキューにトランスポート要求を残す (Leave Transport Request in Queue for Later Import)**
- **トランスポート要求の再インポート (Import Transport Request Again)**
- **オリジナルの上書き (Overwrite Originals)**
- **未確認修復のオブジェクトの上書き (Overwrite Objects in Unconfirmed Repairs)**

SAP 基本バージョンで、「**無効なコンポーネント・バージョンを無視 (Ignore Invalid Component Version)**」オプションが有効化されている場合、当該オプションが選択されていることを確認します。

タスクの結果

ご使用の SAP のリリース・レベルによっては、**tp import** コマンドを実行した際に戻りコード 4 を受け取ることがあります。これは、問題が発生したことを示すものではありません。戻りコード 4 を受け取ることは、**import** コマンドで予想される結果です。

SAP エージェントで必要なユーザーおよび権限

SAP システムに対する無許可アクセスから保護するために、SAP システムにログインするユーザーに権限を割り当てることができます。それらの権限により、SAP システムでのユーザーのアクセス・レベルを定義します。

ABAP トランスポートをインポートすると、SAP エージェントは、SAP システムでデフォルトのユーザー ID「IBMMON_AGENT」をデフォルトのパスワード「ITMMYSAP」で作成します。このユーザーはシステム・ユーザーであり、/IBMMON/AUTH 権限プロファイルがユーザーに関連付けられます。ABAP トランスポートのインポート後に /IBMMON/AUTH プロファイルおよび IBMMON_AGENT ユーザーが作成されます。/IBMMON/AUTH プロファイルを使用することで、SAP システムからパフォーマンス・データを読み取るために必要なトランザクションに IBMMON_AGENT ユーザーがアクセスできます。以下に、使用されるトランザクションの例をいくつか示します。

- CCMS アラートおよび管理
- PI/XI メッセージ・モニターのための権限
- Solution Manager の権限

エージェントに対して任意の他のシステム・タイプ・ユーザーを作成できます。そのユーザーには /IBMMON/AUTH プロファイルを割り当てる必要があります。

SAP コンポーネントのデータの表示およびアクセスのために、エージェントに対して作成するユーザーが、以下の表に指定されたすべての権限を持っている必要があります。

表 *201.* 権限のリスト

コンポーネント	権限オブジェクト	権限の説明
以下のコンポーネントを含む一般的なシステム権限: • SAP インスタンス • SAP システム	S_ADMI_FCD	SAP システムへのアクセス
	S_BDS_DS -BC-SRV-KPR-BDS	文書セットへのアクセス
	S_BTCH_JOB	バックグラウンド・ジョブの操作の実行
	S_CCM_RECV	セントラル・システム・リポジトリー・データの転送
	S_C_FUNCT	ABAP プログラムでの C カーネル関数呼び出しの実行
	S_DATASET	ファイルへのアクセス
	S_RFC	RFC アクセス権限の検査。S_RFC 権限オブジェクトには、以下の 2 つのサブ権限が含まれます。 • RFC1: RFC1 機能グループへの権限の付与。 • SDIFRUNTIME: SDIFRUNTIME 機能グループへの権限の付与。
	S_RFCACL	RFC ユーザーに対する権限の検査
	S_RZL_ADM	R/3 システム管理のための Computing Center Management System (CCMS) へのアクセス
	S_TCODE	アプリケーションに対して定義されたトランザクションを開始するための権限の検査
	S_TOOLS_EX	モニター・ツールでの外部統計レコードの表示
SAP Process Integration を含む PI に対する権限	S_XMB_MONI	XI メッセージ・モニターへのアクセス

表 *201.* 権限のリスト *(*続き*)*		
コンポーネント	**権限オブジェクト**	**権限の説明**
SAP Solution Manager を含む MAI に対する権限	AI_DIAGE2E	E2E 診断機能の制限
	AI_LMDB_OB	Landscape Management Database (LMDB) オブジェクトへのアクセス
	SM_MOAL_TC	SAP Solution Manager のアラート生成およびモニター機能へのアクセスの制御
	SM_WC_VIEW	Solution Manager のワーク・センターの特定の UI 要素に対するアクセスの制限
	S_RFC_ADM	RFC 宛先を管理する権限の制御
	S_RS_AUTH	役割内の分析権限の指定
	SM_APPTYPE	Solution Manager のアプリケーション・タイプへのアクセス
	SM_APP_ID	ワーク・センターで提供されるアプリケーションへのアクセス

SAP システムからの ABAP トランスポートの削除

ご使用のシステムから SAP エージェントを削除する場合は、DELETE トランスポートを SAP システムにインポートする必要があります。DELETE トランスポートにより、SAP エージェントのディクショナリー・オブジェクトと汎用モジュールが除去されます。

始める前に

このトランスポートを SAP システムから削除する前に、この SAP システムをモニターするように 構成された SAP エージェント・インスタンスを停止する必要があります。

SAP システムのバージョンが 7.20 以降の場合は、 DELETE トランスポートをインポートする前に、ご使用のトランスポート・プロファイルにトランスポート・プロファイル・パラメーター **tadirdeletions=true** を追加する必要があります。このトランスポート・プロファイル・パラメーターは tp バージョン 375.57.68 と、R3trans バージョン 6.14 リリース 700 以降でも使用できます。SAP システムからのトランスポート要求の除去について詳しくは、Deleting transport requests を参照してください。

手順

1. 以下のパスに移動します。
   - Windows の場合: *install_dir*¥TMAITM6_x64¥ABAP
   - Windows 以外の場合: *install_dir*/*intrp*/sa/ABAP。ここで、*intrp* は、**lx8266** または **aix526** でなければなりません。
2. トランスポート・ファイルを SAP 環境にコピーします。
3. 以下の手順で、K711_00xxx_DELETE ファイルと R711_00xxx_DELETE ファイルを、SAP トランスポート・システムのデータ・ディレクトリーにコピーします。
   a) K711_00xxx_DELETE ファイルを cofiles ディレクトリーにコピーします。
   b) R711_00xxx_DELETE ファイルを data ディレクトリーにコピーします。
4. 以下のコマンドを実行して、DELETE トランスポートをインポートします。
   a) **tp addtobuffer ITMK711_00xxx_DELETE SID pf=¥usr¥sap¥trans¥bin¥***PROFILE_NAME*

b) **tp import ITMK711_00xxx_DELETE SID client=nnn U16 pf=¥usr¥sap¥trans¥bin¥***PROFILE_NAME* 各部について以下に説明します。

SID
ターゲット SAP システム ID。

PROFILE_NAME
tp プロファイル・ファイルの名前。

nnn
エージェントが実行されるターゲット・クライアントの番号。

エージェント構成の検証

SAP エージェントをインストールしたら、NetWeaver RFC SDK V7.20 ライブラリーをダウンロードし、コピーして、検証することによって、エージェント構成を検証する必要があります。MAI モニターを持つ Solution Manager V7.1 の構成の検証、MAI アラートの検証、Third-Party Component に固有の構成設定の検証も行う必要があります。

以下の手順を実行して、エージェント構成を検証します。

- 744 ページの『NetWeaver RFC SDK V7.20 ライブラリーのダウンロード』
- 745 ページの『NetWeaver RFC SDK V7.20 ライブラリーの SAP エージェント・セットアップへのコピー』
- 745 ページの『NetWeaver RFC SDK V7.20 ライブラリーの検証』
- 746 ページの『MAI モニターを持つ Solution Manager V7.1 の構成の検証』
- 747 ページの『MAI アラートの検証』
- 747 ページの『Third-Party Component に固有の構成設定の検証』

NetWeaver RFC SDK V7.20 ライブラリーのダウンロード

SAP エージェントのインストールが終了したら、NetWeaver RFC SDK V7.20 ライブラリーをダウンロードします。NetWeaver RFC SDK V7.20 ライブラリーに関連するすべてのファイルは、SAP の Web サイトからダウンロードできます。

手順

1. 次の URL を使用して、SAP Marketplace にログインします。
   http://service.sap.com
2. 「**SAP Support Portal**」をクリックします。
3. ご使用の Service Marketplace のユーザー名とパスワードを入力します。
4. 「**Software Downloads**」をクリックし、「**Support Packages and Patches**」リンクを展開します。
5. 「**Browse our Download Catalog**」をクリックし、「**Additional Components**」をクリックします。
6. 「**SAP NetWeaver RFC SDK**」をクリックし、「**SAP NetWeaver RFC SDK 7.20**」をクリックします。
7. SAP エージェントが配置されているマシンのオペレーティング・システムを選択します。
8. `*.SAR` ファイルをコンピューターにダウンロードします。
9. SAP から提供されている SAPCAR ユーティリティーを 使用して `SAP Netweaver RFC SDK *.SAR` ファイルを解凍するには、次のコマンドを実行します。
   `sapcar -xvf SAP NetWeaver RFC SDK File Name.SAR`

   注 : SAPCAR ユーティリティーは、SAP の Web サイトからダウンロードできます。
10. 解凍されたフォルダー内の `lib` フォルダーにナビゲートします。

次のタスク

NetWeaver RFC SDK V7.20 ライブラリーを SAP エージェント・セットアップにコピーします。

NetWeaver RFC SDK V7.20 ライブラリーの SAP エージェント・セットアップへのコピー

NetWeaver RFC SDK V7.20 ライブラリーには、SAP エージェント・セットアップ・ロケーションに手動でコピーする必要があるファイルが含まれています。

手順

1. NetWeaver RFC SDK V7.20 ライブラリーをダウンロードしたディレクトリーにナビゲートします。
2. ファイルを SAP エージェント・セットアップ・ロケーションにコピーします。
   - Windows 64 ビット・オペレーティング・システムでは、以下のファイルをコピーする必要があります。
     - `icuin34.dll`
     - `libicudecnumber.dll`
     - `libsapucum.dll`
     - `icudt34.dll`
     - `icuuc34.dll`
     - `sapnwrfc.dll`

     これらのファイルを `install_dir¥TMAITM6_x64` のロケーションにコピーする必要があります。
   - Windows 以外のオペレーティング・システムでは、`install_dir/`*`intrp`*`/sa/lib` ロケーションにファイルをコピーする必要があります。ここで、*intrp* はオペレーティング・システムのコード (aix526、li6263、sol606) です。次のファイルをコピーする必要があります。
     - `libsapnwrfc.so`
     - `ibicudecnumber.so`
     - `ibicuuc34.a`
     - `libicui18n34.a`
     - `libicudata34.a`
     - `libsapucum.so`

次のタスク

ダウンロードされた NetWeaver RFC SDK V7.20 ライブラリーのバージョンを確認します。

NetWeaver RFC SDK V7.20 ライブラリーの検証

解凍したファイルをコピーしたら、ファイルのバージョンを検証する必要があります。

手順

- Windows ファイルのバージョンを検証するには、以下のステップを実行します。

  a) `sapnwrfc.dll` を右クリックし、「**プロパティ**」をクリックします。

  b)「**詳細**」タブをクリックします。

  c)「**製品バージョン**」セクションで、ファイルがバージョン 720、パッチ 514、変更リスト 1448293 以降であることを確認します。
- Linux AIX ファイルのバージョンを検証するには、以下のステップを実行します。

  a) 解凍した `*.SAR` ファイルの `lib` フォルダーに移動します。

  b) コマンド **strings libsapnwrfc.so | grep SAPFileVersion** を実行します。

  c) `[root@IBMSAP2V6 lib]# strings libsapnwrfc.so | grep SAPFileVersion GetSAPFileVersion #[%]SAPFileVersion: 7200, 514, 22, 6206 .GetSAPFileVersion` というメッセージが表示されます。

注 : このメッセージは、このライブラリーがバージョン 720、パッチ 514 以降であることを示しています。

MAI モニターを持つ Solution Manager V7.1 の構成の検証

MAI アラートのデータを受信するには、Solution Manager V7.1 が正しく構成されていることを検証する必要があります。

このタスクについて

MAI (Monitoring and Alert Infrastructure) を持つ Solution Manager V7.1 を使用して、管理対象システムをモニターできます。Solution Manager V7.1 は、それ自体、およびサテライト・システムをモニターします。各サテライト・システムには、プラグインと診断エージェントがあります。診断エージェントは、ホスト・レベルまたはオペレーティング・システム・レベルで、データをフェッチします。各ホストは、Solution Manager の複数のモニター対象ホストについて、複数の診断エージェントを持つことができます。Solution Manager の MAI モニターで使用されるキーワードは次のとおりです。

- メトリック: サテライト・システムからのデータ。
- アラート: 構成可能なしきい値をある程度超えることに基づく通知。
- インシデント: チケットに変換され、任意のユーザーに割り当てられるアラート。

MAI モニターを持つ Solution Manager V7.1 の構成を検証するには、基本設定、グローバル・レベルの設定、およびテンプレート・レベルの設定を検証する必要があります。

手順

1. 基本設定を検証するには、トランザクション・コード SOLMAN_SETUP を入力し、**Enter** を押します。
   次のタブのすべての LED が緑であることを確認します。
   - 概要
   - Basic Configuration
   - Managed System Configuration

   注 : 管理対象システムには、技術システム、技術シナリオ、ホスト、データベース、インスタンス、PI ドメイン、技術コンポーネント、接続などのさまざまなカテゴリーがあります。これらの管理対象システムを、ビジネス要件に合わせて構成する必要があります。MAI アラートは、構成した管理対象システムに基づいて発行されます。
2. トランザクション・コード SE38 を入力し、**Enter** を押します。
3. プログラム名を RTCCTOOL と指定して、レポートを実行します。
   出力で、すべての LED が緑であることを確認します。
4. グローバル・レベルの設定を検証するには、トランザクション・コード SOLMAN_WORKCENTER を入力し、**Enter** を押します。
   次のタブのすべての LED が緑であることを確認します。
   - 概要
   - Configure Infrastructure
   - Pre-requisites
   - Configure
5. 「**Notification**」の「**Global Settings**」状況が「**Active**」であることを確認します。
6. テンプレート・レベルの設定を検証するには、トランザクション・コード SOLMAN_SETUP を入力し、**Enter** を押します。
   「**Technical Settings**」の「**Auto-Notifications**」リストで、「**Active**」が選択されていることを確認します。

   注 : 初期のトラブルシューティングのために、E メール通知をアクティブにするようにしてください。
7. MAI システム・モニタリングについて、次の手順を使用して、End-User Experience Monitoring (EEM) の構成を検証します。

a) トランザクション・コード SE37 を入力し、**Enter** を押します。

b)「**Function Module name**」フィールドに **AI_EEM_LIST_ALL_SCENARIOS** と入力し、F8 を押します。

End-User Experience Monitoring (EEM) の項目が存在する必要があります。

MAI アラートの検証

Solution Manager MAI が、Technical Monitoring 内の MAI アラート受信箱をモニターするように正しく構成されていることを確認するには、MAI アラートを出力として受信することを検証する必要があります。

手順

1. トランザクション・コード SOLMAN_WORKCENTER を入力し、**Enter** を押します。Technical Monitoring の下の Solution Manager の MAI アラート受信箱に MAI アラートを表示できるかどうかを確認します。
2. 次の手順を使用して、BAdi 実装を確認します。
   a) トランザクション・コード SE19 を入力し、**Enter** を押します。
   b)「**Enhancement Implementation**」フィールドに /IBMMON/ITM_IMPL_ALRTINBX と入力します。
   c)「**Display**」をクリックして、「**Runtime Behavior**」セクションで BAdi 実装がアクティブになっているかどうかを確認します。
3. 以下の手順を実行して、データベース /IBMMON/ITM_ALIX に MAI アラートが含まれているかどうかを確認します。
   a) トランザクション・コード SE16 を入力し、**Enter** を押します。
   b)「**Table Name**」フィールドに /IBMMON/ITM_ALIX と入力して実行します。表内に MAI アラートが受信されていることを確認します。
4. トランザクション・コード SE37 を入力し、**Enter** を押します。
5.「**Function Module Name**」フィールドに /IBMMON/ITM_MAIALRT_INX と入力し、F8 を押します。

   出力として MAI アラートが表示されます。

次のタスク

/IBMMON/ITM_ALIX データベースで MAI アラートを表示できない場合は、Third-Party Component の設定を検証する必要があります。

Third-Party Component に固有の構成設定の検証

MAI アラートを表示できない場合は、Third-Party Component の設定を検証する必要があります。

手順

1. Third-Party Component がアクティブであることを検証します。
2.「**OS Adapter**」の「**BAdi Implementation**」の下で、「**Alert Reaction**」が選択可能であることを確認します。「**Alert Reaction**」が選択不可の場合は、デフォルトの設定を削除し、「**BAdi implementation - Alert Reaction**」を選択します。
3. 次の手順を使用して、テンプレート設定を確認します。
   a) 特定のアラートを SAP ABAP 7.0.0 などの Third-Party System に転送するために使用される設定を検証します。
   b)「**Expert Mode**」を選択し、「**Alerts**」を選択して、「**Third Party Component**」をクリックします。

   Alert Reaction BAdi 名を表示できることを確認します。

   注 : 最新の SAP ノートが実装されていることを確認します。Solution Manager V7.1 サービス・パック 8 の場合、次のノートが実装されていることを確認します。

   - https://service.sap.com/sap/support/notes/1959978
   - https://service.sap.com/sap/support/notes/1820727
4. /IBMMON/ITM_MAIALRT_INX データベースで MAI アラートを表示できない場合は、Third-Party Component に対して Solution Manager の MAI の以下の構成手順を実行する必要があります。

a) トランザクション・コード SOLMAN_SETUP を入力し、**Enter** を押します。

b)「**Technical Monitoring**」で、「**System Monitoring**」を選択します。

c)「**Configure Infrastructure**」タブをクリックし、「**Default Settings**」タブをクリックします。

d)「**Third Party Components**」タブをクリックし、「**Edit**」をクリックします。

e) リストから「**Active**」を選択します。

f) 選択したコネクターについて、スコープ・フィルターが「**All alerts, Events and Metrics (with Internal Events)**」に設定されていることを確認します。

注 : OS コマンド・アダプターも、サード・パーティーのコネクターにデータをプッシュする 方法の 1 つです。OS コマンド・アダプターを 構成するには、「How-to guide for OS Command Adapter」で構成の詳細設定を参照してください。

データベース通信ポート番号の追加

データベース通信ポート番号は、統合されたシナリオ内のデータベース・エンティティーを 一意的に識別するために不可欠です。コンポーネント間のコラボレーションを実現するために、SCM AI のコンポーネントには OSLC (Open Source Lifecycle Collaboration) が含まれています。OSLC 準拠では、コラボレーション・コンポーネントを一意的に識別することが不可欠です。したがって、データベース通信ポート番号が重要となります。

このタスクについて

適切な IBM Tivoli Monitoring トランスポートを SAP システムにインポートすると、/IBMMON/ITM_PORT データベース表が自動的に作成されます。この表には、以下のデータベース・フィールドがあります。

- システム ID
- システム・ホスト名
- DB (データベース) 通信ポート番号 (DB (Database) Communication Port #)

手順

OSLC 準拠に必要な SAP エージェントの SAP データベース通信ポート番号を追加するには、以下のステップを実行します。

1. SE16 トランザクション・コードに移動し、Enter キーを押します。
2.「**データベース表名 (Database table name)**」フィールドに、/IBMMON/ITM_PORT と入力し、F7 キーを押します。
3. **/IBMMON/ITM_PORT** データベース表の選択画面が表示されたら、F8 キーを押します。
   /IBMMON/ITM_PORT データベース表には、以下の 3 つのデータベース・フィールドがあります。
   - システム ID
   - システム・ホスト名
   - DB 通信ポート番号 (DB Communication Port #)

   注 : /IBMMON/ITM_PORT データベース表に表示される SAP システムは、Java および ABAP の両方のアーキテクチャー用のものです。
4.「**DB 通信ポート番号 (DB Communication Port #)**」フィールドに、各 SAP システム ID の該当する SAP データベース通信ポート番号と SAP システムのホスト 名を入力し、変更を保存します。

   注 : /IBMMON/ITM_PORT データベース表の「**DB 通信ポート番号 (DB Communication Port #)**」フィールドに値を入力しない場合、デフォルトで DB 通信ポート番号は 0 になります。

SAP エージェントの拡張インストールおよび構成

下記は、SAP エージェントに固有の拡張インストールと構成です。

以下のインストールと構成のトピックについて説明します。

- 749 ページの『SAP 汎用モジュール』

- 750 ページの『SAP ユーザー ID』
- SAP エージェント のユーティリティー
- 750 ページの『SAP RFC 接続』
- 761 ページの『「接続のテスト」機能』
- 752 ページの『SAP でのオプションの拡張構成』
- 759 ページの『CEN CCMS レポート』
- 760 ページの『SAP システムからの Advanced Business Application Programming (ABAP) トランスポートのアンインストール』

注 : SAP エージェントの拡張インストールおよび構成では、資料に ABAP トランスポートのカスタム・トランザクション・コード UI との互換性を持たせるために、IBM Tivoli Monitoring に言及しています。

SAP 汎用モジュール
SAP サーバー上のデータ・ボリュームが大きい場合、特定のウィジェットで問題が発生して、サーバーからの応答時間が長くなることがあります。それらのウィジェットが不可欠でない場合には、関連する SAP 汎用モジュールを使用不可にすることができます。

デフォルトでは、SAP エージェントの汎用モジュールは使用可能になっています。ただし、以下の汎用モジュールはデフォルトで使用不可になっています。

- SYS サブノードにある HTTP サービス (`/IBMMON/ITM_HTTP_SRVS`)
- PI/XI サブノードにある XML メッセージ (`/IBMMON/ITM_SXMB_MONI_NEW`)
- PI/XI サブノードにある同期/非同期通信 (`/IBMMON/ITM_SYN_ASYN_COMM`)
- Sys サブノードにある qRFC インバウンド・キューの詳細 (`/IBMMON/ITM_QIN_QDETAILS`)

SAP 汎用モジュールを使用不可にした後は、ウィジェットを選択しても、IBM Application Performance Management ダッシュボードにはデータは表示されません。したがって、パフォーマンス関連の問題を避けることができます。

SAP エージェント汎用モジュールの使用可能化
以前にパフォーマンス上の問題を解決するために SAP エージェント汎用モジュールを使用不可にした場合は、汎用モジュールを使用可能にすることもできます。

手順

1. SAP システムにログオンします。
2. SE16 トランザクション・コードを実行します。
3. 表名として `/IBMMON/ITM_CNFG` と入力します。
4. 削除する行を選択し、**Shift+F2** キーを押してその項目を削除します。
5. 「**保存**」をクリックします。

SAP 汎用モジュールの使用不可化
一部のウィジェットにより、SAP サーバーからの応答が遅くなる場合があるため、SAP 汎用モジュールを使用不可にすることにより、サーバーのパフォーマンスを改善することができます。

手順

1. SAP システムにログオンします。
2. SE16 トランザクション・コードを実行します。
3. 表名として `/IBMMON/ITM_CNFG` と入力します。
4. **F5** キーを押して、新規項目を作成します。
5. 「**PARM NAME**」フィールドに SAP 汎用モジュールの名前を入力します。
6. 「**VALUE CHAR**」フィールドに「No」と入力します。
7. 「**保存**」をクリックします。

SAP ユーザー ID

このセクションでは、SAP エージェントで必要となる SAP ユーザー ID および許可について説明します。

ユーザー ID は以下の目的をサポートします。

- 750 ページの『SAP RFC 接続』
- 750 ページの『基本エージェント・モニター』

SAP RFC 接続

SAP エージェントは、内部の集中 Computing Center Management (CCMS) のポーリングと CCMS アラート・データの収集にリモート関数呼び出し (RFC) 接続を使用します。この動作は SAP RFC アーキテクチャーに固有のものです。

SAP エージェントでは、このエージェントによってモニターされる SAP システムへの専用 RFC 接続を 1 つ開きます。その後、SAP システムは、汎用モジュールおよびプログラムを介してデータを収集するために、アプリケーション・サーバーごとに内部接続を 1 つ開きます。エージェントが CCMS アラートを収集する場合、SAP システムは、この収集スレッドの各アプリケーション・サーバーに対して、追加で (システム内部) RFC 接続を 1 つずつ開きます。データ収集が開始されると、このエージェント用の RFC 接続が 1 つ開きます。その後、追加の内部システム RFC 接続用に、最大で SAP アプリケーション・サーバー数の 2 倍の数の接続が開きます。

モニター中のインスタンスで、追加の RFC セッションに対応できる必要があります (特に、10 個以上のインスタンスがある大規模なシステムでの場合)。モニターに予測される RFC 負荷がシステム・パフォーマンスおよび許容度に悪影響を及ぼす可能性がある場合は、SAP プロファイル・パラメーターを調整してください。SAP 管理者に問い合わせ、また次の SAP ノートも参照してください。

- 端末セッション (デフォルト設定: 200) 22099
- 通信/ゲートウェイ/会話設定 887909 316877 384971

基本エージェント・モニター

SAP エージェントは、エージェント・トランスポートのインポート時に、SAP システム上に IBMMON_AGENT を作成します。

このユーザー ID は IBMMON_AGENT で、デフォルトのパスワードは ITMMYSAP です。これは、通信タイプ・ユーザー専用となり、/IBMMON/AUTH 許可プロファイルを使用するように事前構成されます。このプロファイル (トランスポートのインポート時に作成されるプロファイル) には、エージェントの Advanced Business Application Programming (ABAP) コードを実行するための最小限の許可セットが含まれています。また、このプロファイルは、SAP システムで制限されているアクションのセットを受け入れます。

このユーザー ID を受け入れられない場合 (インストール命名規則に違反する場合など) は、別のユーザー ID を作成できます。ユーザー ID は、許容される任意の SAP ユーザー ID にすることができますが、/IBMMON/AUTH プロファイルに完全な許可のセットが必要です。ユーザー ID は、通信タイプ・ユーザー専用アクセスを必要とします。

デフォルトのユーザー ID は、以下の目的のために十分な権限しか提供しません。

- モニターおよびデータ収集
- Computing Center Management System (CCMS) アラートのクローズ
- ゲートウェイ統計の使用可能化、使用不可化、およびリセット
- Oracle データベース統計のリセット

エージェントのアクション機能を制限する場合は、一部のアクション権限 (CCMS アラートのクローズなど) を除去できます。

IBM Application Performance Management UI ポータルで特定のコンポーネントについてのデータにアクセスする場合、適切な権限を所持していることを確認してください。次の表に、さまざまなサブノードのデータにアクセスするために必要な権限をリストします。

表 *202.* 権限のリスト

サブノード	権限オブジェクト	権限の説明
以下のサブノードを含む一般的なシステム権限: • Ins • Sys	S_ADMI_FCD	システムへのアクセス
	S_BDS_DS -BC-SRV-KPR-BDS	文書セットへのアクセス
	S_BTCH_JOB	バックグラウンド・ジョブの操作の実行
	S_CCM_RECV	中央システム・リポジトリーのデータの転送
	S_C_FUNCT	ABAP プログラムでの C 呼び出しの実行
	S_DATASET	ファイルへのアクセス
	S_RFC	RFC アクセス権限の検査。S_RFC 権限オブジェクトには、以下の 2 つのサブ権限が含まれます。 • RFC1: RFC1 機能グループへの権限の付与。 • SDIFRUNTIME: SDIFRUNTIME 機能グループへの 権限の付与。
	S_RFCACL	RFC ユーザー用
	S_RZL_ADM	Computing Center Management System (CCMS): システム管理へのアクセス
	S_TCODE	トランザクション開始時のトランザクション・コードのチェック
	S_TOOLS_EX	ツール・パフォーマンス・モニターへのアクセス
以下のサブノードを含む Solution Manager の許可: • Lds • Sol	D_MD_DATA -DMD	マスター・データのデータ内容の表示
	D_SOLMANBU	Solution Manager のセッション・タイプへのアクセス
	D_SOLM_ACT	Solution Manager 内のソリューションへのアクセス
	D_SOL_VSBL	Solution Manager 内のソリューションの表示
	S_CTS_SADM	システム固有の管理 (トランスポート) の表示
	S_TABU_RFC	クライアントの比較とコピー、RFC でのデータ・エクスポートの表示
PI サブノードを含む PI の許可	S_XMB_MONI	XI メッセージ・モニターへのアクセス

表 202. 権限のリスト (続き)		
サブノード	**権限オブジェクト**	**権限の説明**
Slm サブノードを含む MAI の許可	AI_DIAGE2E	Solution Diagnostics エンドツーエンド分析へのアクセス
	AI_LMDB_OB	Landscape Management Database (LMDB) オブジェクトへのアクセス
	SM_MOAL_TC	Monitoring and Alerting へのアクセス
	SM_WC_VIEW	ワーク・センターのユーザー・インターフェース 要素へのアクセス
	S_RFC_ADM	RFC 宛先の管理オプションへのアクセス
	S_RS_AUTH	役割内の BI 分析へのアクセス
	SM_APPTYPE	Solution Manager のアプリケーション・タイプへのアクセス
	SM_APP_ID	ワーク・センターで提供されるアプリケーションへのアクセス

集中ユーザー管理 (CUA) の使用
集中ユーザー管理 (CUA) は、SAP システムをモニターするために 使用されます。

手順

事前定義のユーザー ID と 許可の役割を使用して、集中ユーザー管理を使用してセットアップされた SAP システムをモニターするには、以下のいずれかの手順を実行します。

- 集中ユーザー管理の親論理システム・クライアントにトランスポートをインストールします。
- トランスポートをインストールするクライアントで、ユーザー ID または役割を手動で作成します。ユーザー ID または役割が、トランスポートをインストール (インポート) するクライアントに置かれます。
- 集中ユーザー管理の親論理システム・クライアントでユーザー ID または役割を手動で作成します。その後、エージェントを実行するクライアントにそのユーザー ID または役割を配布します。
- 集中ユーザー管理の親論理システム・クライアントでユーザー ID または役割を手動で作成し、そのクライアントでエージェントを実行します。

SAP でのオプションの拡張構成
標準の SAP 機能またはエージェント提供の SAP 機能を使用して、SAP エージェントを構成できます。

SAP のエージェント提供トランザクションを使用して、さまざまなエージェントの動作をカスタマイズします。/n/IBMMON/ITM_CONFIG トランザクションを実行して SAP のメイン構成メニューにアクセスした後、次の構成オプションのいずれかを選択します。

- 753 ページの『コピー、バックアップ、リストアの機能とトランザクション』
- 753 ページの『トランザクションを使用したデータのコピー、バックアップ、リストア』
- 754 ページの『コマンド・ラインユーティリティー・ツール』
- 755 ページの『Windows 環境でのコマンド・ラインユーティリティーの 実行』
- 755 ページの『非 Windows 環境でのコマンド・ラインユーティリティーの 実行』
- 756 ページの『アラートの保守』
- 757 ページの『モニター・セットおよびモニター・トランザクションの選択』
- 757 ページの『SAP システムでのダイアログ・ステップ応答しきい値の構成』

注 : /IBMMON/ITM* トランザクションの前には、必ず /n を付けてください。

これらのトランザクションで行われた構成変更は、管理対象グループを保守するための変更を除き、即時に SAP エージェントによって使用されます。管理対象グループの構成を変更すると、その変更内容は、次のハートビートで SAP エージェントによってディスカバーされます。

SAP の標準機能を使用して、757 ページの『SAP システムでのダイアログ・ステップ応答しきい値の構成』の構成を完了させてください。

コピー、バックアップ、リストアの機能とトランザクション

SAP サーバーにログオンし、トランザクション /n/IBMMON/ITM_CONFIG を実行すると、コピー、バックアップ、およびリストアの機能を使用できます。

コピー、バックアップ、およびリストアの操作により、IBM Tivoli Monitoring 構成データのコピー、バックアップ、およびリストアを実行できます。

この機能を使用すると、以下の機能の中から選択したり、IBM Tivoli Monitoring 構成データを保存したりできます。

- **コピー**

  この機能を使用して、ある SAP サーバーから別の SAP サーバーに IBM Tivoli Monitoring の構成設定をコピーします。例えば、IBM Tivoli Monitoring の構成設定をエージェント **a1** から SAP サーバー・インスタンス SAP2 にコピーしたい場合があります。このエージェントはシステム **m1** 上で稼働し、SAP サーバー・インスタンス SAP 1 用に構成されています。SAP サーバー・インスタンス・モニター設定を除くすべての IBM Tivoli Monitoring の構成設定は、ターゲット SAP システムにコピーされます。コピー機能は、コマンド・ラインユーティリティーまたは SAP GUI を使用して実装します。

- **バックアップ**

  システムのバックアップを 行うことにより、SAP サーバー上で作成したエージェント固有の構成を保管できます。この機能を使用して、IBM Tivoli Monitoring に固有の構成設定を SAP システムに保存します。/IBMMON/ITM_CONFIG トランザクションを使用して、設定を入力します。バックアップ・ファイルは、SAP サーバー上の作業ディレクトリー (パス: /usr/sap//DVEBMGS/work) に保管されます。

- **復元**

  この機能を使用して、IBM Tivoli Monitoring の構成データを作業ディレクトリーから SAP サーバーにリストアします。IBM Tivoli Monitoring の構成データのリストアは、この構成データのバックアップ手順を実行したのと同じ SAP サーバーで行うことも、別の SAP サーバーで行うこともできます。IBM Tivoli Monitoring の構成データは、特定の SAP の表と IBM Tivoli Monitoring の表にリストアすることができます。構成ファイルには日付とタイム・スタンプが付けられて保管されるため、どの時点でのファイルをリストアするかを選択することができます。

エージェント固有の構成には、SAP の /IBMMON/ITM_CONFIG トランザクション内の構成設定が含まれています。以下の構成手順を実行できます。

- アラートの頻度をサンプリングする。
- 特定のアラートを有効にする。
- ログ・ファイル名を保管する。
- グループ定義を管理する。
- モニター・セットとモニターを選択する。
- モニターの目的で SAP インスタンスを選択する。

トランザクションを使用したデータのコピー、バックアップ、リストア

SAP ユーザー・インターフェースで /n/IBMMON/ITM_CONFIG トランザクションを使用して、データのコピー、バックアップ、リストアを行います。

始める前に

コピー、バックアップ、およびリストアの手順を使用して、ある SAP サーバーから別の SAP サーバーに IBM Tivoli Monitoring 構成設定をコピーします。SAP サーバー・インスタンス・モニター設定以外のすべての IBM Tivoli Monitoring 構成設定は、ターゲット SAP システムにコピーされます。

手順

SAP でデータをコピー、バックアップ、およびリストアするには、以下の手順に従います。

- **コピー**
  a. ターゲット SAP のシステム ID と既存ファイルの名前を `source system id__<filenam>date_time` 形式で入力します。

     /IBMMON/ITM_COPY トランザクションにより、IBM Tivoli Monitoring 構成ファイルが `SAP target SAP system id__<filename>_date_time` というファイル名で作業ディレクトリー内に作成されます。
  b. IBM Tivoli Monitoring 構成データをこのファイルにコピーするには、「**実行**」をクリックします。
  c. 前の IBM Tivoli Monitoring 構成画面に戻るには、「**戻る**」または「**キャンセル**」をクリックします。

     必要な入力パラメーターは、コピーされる **Target System id** と **filename** です。
- **バックアップ**
  a. SAP サーバーにログオンして、/IBMMON/ITM_CONFIG トランザクションを開始します。
  b.「**バックアップ**」を選択します。
  c. バックアップ・ファイルの名前を入力します。

     ファイル名は `sys_id_<filename>_date_time` として保管されます。
  d. バックアップを実行してファイルをアプリケーション・サーバーに保管するには、「**実行**」をクリックします。

     注 : バックアップ・ファイルは、アプリケーション・サーバーの作業ディレクトリーに保管されます。
  e. 前の IBM Tivoli Monitoring 構成画面に戻るには、「**戻る**」または「**キャンセル**」をクリックします。
- 「**復元**」
  a. SAP サーバーにログオンして、/IBMMON/ITM_CONFIG トランザクションを開始します。
  b.「**復元**」を選択します。
  c. リストアするファイル名は、`sys_id_<filename>_date_time` の形式で入力します。
  d. IBM Tivoli Monitoring 構成データをリストアするには、「**実行 (Execute)**」をクリックします。
  e. 前の IBM Tivoli Monitoring 構成画面に戻るには、「**戻る**」または「**キャンセル**」をクリックします。

コマンド・ラインユーティリティー・ツール

コマンド行ユーティリティー・ツールを使用して、SAP サーバー上で IBM Tivoli Monitoring 構成データのコピー、バックアップ、およびリストアを行うことができます。

コマンド・ラインユーティリティー・ツールは、Windows 環境と非 Windows 環境で実行できます。755 ページの『Windows 環境でのコマンド・ラインユーティリティーの 実行』および 755 ページの『非 Windows 環境でのコマンド・ラインユーティリティーの 実行』を参照してください。

- **コピー**

  backup コマンドを実行して、エージェント・ディレクトリー SAP サーバー・インスタンス sap1 から sap2 に IBM Tivoli Monitoring 構成ファイルをコピーします。sap1 エージェント・ディレクトリーから、ファイル名と、ソース・システムとして sap1 を入力します。これにより ABAP 機能が呼び出され、このファイルから sap2 の IBM Tivoli Monitoring 構成ファイルに IBM Tivoli Monitoring の設定がコピーされます。次に、sap1 エージェント・ディレクトリー・ユーティリティー・ツールから **Copy** を選択し、ファイル名と、ターゲット SAP システムとして sap2 を入力します。
- **バックアップ**

  コマンド・ラインユーティリティー・ツールを実行した後、「**バックアップ**」 オプションを選択します。次に、ファイル名と SAP システム ID を入力する必要があります。ツールにより、/IBMMON/ITM_BACKUP SAP 汎用モジュールが呼び出されます。この汎用モジュールは、表に保管されている特定の IBM Tivoli Monitoring 構成設定を読み取り、その設定を行分離文字と列分離文字を使用して保管しま

す。次に、コマンド・ラインユーティリティー・ツールは、ストリングを読み取ってデータをファイルに書き込みます。生成されるファイル名の形式は、ID>_<filename>-<date&time> です。このファイルは、ユーティリティー・プログラムが保管されているディレクトリーに格納されます。

- **復元**

  コマンド・ラインユーティリティー・ツールの実行後に、リストアするファイル名と、そのファイルのリストア先となるターゲット SAP システムを入力します。コマンド・ラインユーティリティー・ツールは、エージェント・ディレクトリーからファイルを読み取り、/IBMMON/ITM_RESTORE SAP 汎用モジュールを呼び出します。次に、ツールは IBM Tivoli Monitoring 構成をストリングとして渡します。SAP 汎用モジュールは、IBM Tivoli Monitoring の特定の表を更新し、IBM Tivoli Monitoring の特定の構成をリストアします。

Windows 環境でのコマンド・ラインユーティリティーの実行

Windows 環境でコマンド・ラインユーティリティーを 実行して、コピー、バックアップ、およびリストアの各手順を実行できます。

手順

1. 使用しているオペレーティング・システムに応じて、以下のいずれかの手順を実行します。
   - 64 ビット・オペレーティング・システムの場合は、コマンド **set CANDLE_HOME = C:¥IBM¥APM** を使用して CANDLEHOME パスを設定し、パス %candle_home%¥ TMAITM6x64 から **ksacopybackuprestore.bat** コマンドを実行します。
2. バックアップ・ファイルを作成するには、以下の手順を実行します。
   a)「**バックアップ**」を選択して、ファイル名とソース SAP システム名を入力します。
   b) バックアップ・ファイルが SYS ID>_<filename>_<date&time> の形式で作成されます。
3. ファイルを復元するには、以下の手順を実行します。
   a)「**復元**」を選択して、ターゲット SAP システム名を入力します。
   b) ファイル名を入力してください。
4. ファイルをコピーするには、以下の手順を実行します。
   a) ソース・エージェントで、「**バックアップ**」を選択してバックアップ・ファイルを作成します。
   b) ソース・エージェント・ディレクトリーからターゲット・エージェント・ディレクトリーにバックアップ・ファイルをコピーします。
   c) ソース・ディレクトリーでコマンド・ラインユーティリティー・ツールを実行して「**コピー**」を選択します。
   d) ファイル名とターゲット SAP システムを入力します。

非 *Windows* 環境でのコマンド・ラインユーティリティーの実行

非 Windows 環境でコマンド・ラインユーティリティーを 実行して、コピー、バックアップ、およびリストアの各手順を実行できます。

手順

1. パス /candle_home/<arch>/sa/shell から **ksacopybackuprestore.sh** コマンドを実行します。
2. バックアップ・ファイルを作成するには、以下の手順を実行します。
   a)「**バックアップ**」を選択して、ファイル名とソース SAP システム名を入力します。
   b) バックアップ・ファイルが SYS ID>_<filename>_<date&time> の形式で作成されます。
   バックアップ・ファイルは %candlehome% / arch /sa/bin に保存されます。
3. ファイルを復元するには、以下の手順を実行します。
   a)「**復元**」を選択して、ターゲット SAP システム名を入力します。
   b) ファイル名を入力してください。
4. ファイルをコピーするには、以下の手順を実行します。

a) ソース・エージェントで、「**バックアップ**」を選択してバックアップ・ファイルを作成します。
b) ソース・エージェント・ディレクトリーからターゲット・エージェント・ディレクトリーにバックアップ・ファイルをコピーします。
c) ソース・ディレクトリーでコマンド・ラインユーティリティー ・ツールを実行して「**コピー**」を選択します。
d) ファイル名とターゲット SAP システムを入力します。

アラートの保守
Tivoli Monitoring によって生成されたアラートは、それらの状況としきい値を変えることによって変更することができます。

このトランザクションは、Tivoli Monitoring によって生成されるアラートを使用可能または使用不可にしたり、警告およびクリティカルしきい値を設定したりするために使用します。Tivoli Monitoring によって生成されるすべてのアラートは、現在の状況およびしきい値とともに表示されます。

アラート状況およびしきい値を変更した場合、変更後の値は次のサンプリング時に使用されます。

デフォルト・サンプル期間の保守

デフォルト・サンプル期間では、特定の属性グループのリアルタイム ・レポートに関する情報を提供します。

いくつかの属性グループには、グループ内の各レコードに対して暗黙的な日時があります。例えば、R/3_Abap_Dumps 属性グループはダンプに対して作成時間を報告し、R/3_System_Log 属性グループはログ・エントリーに対して作成時間を報告します。これらのレコードには、日時フィールドがあります。 最新の情報だけでなく、表の簡単なヒストリーのレポートを取得できます。この時間間隔はデータ収集のためのタイム・スパンであり、データを収集する際のリアルタイム・インターバルとして使用されます。/IBMMON/ITM_PERIOD トランザクションは、これらの属性グループそれぞれに対して、デフォルトのサンプル期間 (リアルタイム・レポートのタイム・スパン) を定義します。サンプル期間は、現在時刻から過去にさかのぼって、データのサンプル期間の長さを識別します。

ログ・ファイル名の保守

インスタンスにのみ一致した特定のログ・ファイルが、ログ・ファイル情報とともに IBM Tivoli Monitoring レポート内に組み込まれます。

このトランザクションは、ログ・ファイル情報を含む IBM Tivoli Monitoring レポートに組み込むログ・ファイルを識別するために使用します。指定されたインスタンスで指定された名前パターンに一致する名前を持つすべてのログ・ファイルが、次のデータ収集間隔のレポートに組み込まれます。

管理対象グループの保守

管理対象グループ名トランザクションによって、SAP システム内の特定のトランザクションがモニターおよび処理されます。

このトランザクションは、IBM Tivoli Monitoring 管理対象グループ定義の保守に使用します。管理対象グループ名はすべて IBM Application Performance Management UI ポータルに渡され、「管理対象システム 選択リスト (Managed System Selection Lists)」に表示されます。データ収集時には、属性選択条件に一致するデータのみが SAP エージェントに送信されます。このデータは、レポートに表示されるか、シチュエーションとポリシーでの評価に使用されます。

管理対象グループを使用して、SAP システム内の情報のサブセットをモニターします。SAP システムで関心のある部分のみに焦点を当て、関心のない部分を無視します。例えば、金融アプリケーションに含まれるトランザクションの応答時間のみに関心がある場合は、Financials という名前の管理対象グループを作成します。その後、金融関連のトランザクション・コードのみを組み込みます。 Financials 管理対象グループが Tivoli Enterprise Portal によって処理されるときには、レポートの表示、シチュエーションの評価、またはポリシー評価の対象として、指定のトランザクション・コードを含む情報のみが考慮されます。

モニター・セットおよびモニター・トランザクションの選択
モニター・セットおよびモニター・トランザクションの選択を使用して、集中 Computing Central Management (CCMS) アラート構成を編集します。例えば、CCMS アラートの収集を完全にオフにすることができます。

このトランザクションは、IBM Tivoli Monitoring がアラートを取得する CCMS モニターを選択するために使用します。デフォルトでは、このウィンドウが最初に表示されるときにシステム全体のモニターが選択されます。モニター・セット、モニター、またはモニター・セットとモニターの両方を変更して構成を保存できます。CCMS アラートの収集には、最大 3 つのモニターを選択できます。

CCMS アラートの収集を完全にオフにするには、すべてのモニターのチェック・ボックスをクリアしてこの構成を保存します。

既に実行されているエージェントは、この構成を読み取り、選択されたモニターの CCMS アラートを収集します。ただし、CCMS アラート構成の変更前にエージェントによって既に収集された CCMS アラートは、エージェントおよび IBM Tivoli Monitoring に残ります。

このトランザクションは、モニターおよびモニター・セットを選択するほかに、取得するアラート・タイプの発生回数を指定します。また、取得されていない古いアラートのオカレンスを自動的にクローズするかどうかを決定するためにも使用できます。

***SAP* システムでのダイアログ・ステップ応答しきい値の構成**
SE16 トランザクションを実行することで、トランザクションのダイアログ・ステップ応答しきい値を構成できます。

手順

1. 「**表の名前**」フィールドに /IBMMON/ITM_TRSH と入力してから、「**表の内容 (Table Contents) (F7)**」を選択して、表にアクセスします。
2. 現行のしきい値設定を表示するには、「**実行 (Execute) (F8)**」を選択します。トランザクション名が「**ワークロード**」列に表示され、しきい値が「**しきい値**」列に表示されます。
3. 新しいしきい値設定を追加するには、「**作成 (Create) (F5)**」を選択します。「**ワークロード**」フィールドにトランザクション名を入力します。「**ワークロード**」の値として以下のワイルドカードを使用できます。
   - * は、複数の文字と一致します
   - + は、任意の 1 文字と一致します
4. 「**しきい値**」フィールドに、しきい値をミリ秒単位で入力します。 この設定を保存するには、「**保存**」を選択します。新規のしきい値および変更されたしきい値は、すぐに有効になるわけではなく、以下のいずれかの条件が満たされたときに有効になります。
   - エージェントが再始動される。
   - エージェントが SAP システムへの RFC 接続を再オープンする。この手順は、12 ハートビートごとに起こります。これは、デフォルトで 約 2 時間 10 分ごとです。

タスクの結果
「**しきい値**」列に入力した値は、R/3_Transacation_Performance 属性グループのダイアログ・ステップ応答しきい値属性で返されます。

バッチ・ジョブ操作
指定された時間間隔内のすべてのバッチ・ジョブをフェッチすることができます。

手順

737 ページの『SAP システムでの ABAP トランスポートのインポート』の後のステップを実行します。

要確認 : すべてのバッチ・ジョブにクリティカル定数が設定されます。

1. 指定された時間間隔内のアクティブなバッチ・ジョブとキャンセルされたバッチ・ジョブをすべてフェッチするには、次のようにします。
   次の項目を /IBMMON/ITM_CNFG 表に追加します。

表 *203. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES

2. 指定された時間間隔内のキャンセルされたすべてのジョブ、および時間間隔に関係なくすべてのアクティブなジョブをフェッチするには、次のようにします。
   次の項目を /IBMMON/ITM_CNFG 表に追加します。

表 *204. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES_LONG_RUN

3. 指定された時間間隔内のすべてのバッチ・ジョブ、および時間間隔に関係なくすべてのアクティブなバッチ・ジョブをフェッチするには、次のようにします。
   次の項目を /IBMMON/ITM_CNFG 表に追加します。

表 *205. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES_ALL

注:

- 構成パラメーターが追加されない場合、クリティカル定数が設定されていない、指定された時間間隔内のすべてのバッチ・ジョブをフェッチします。
- フェッチされる行数は、トランザクション・コード /n/IBMMON/ITM_CONFIG で設定されるクリティカル定数の値と常に等しくなります。

IBMMON/ITM_MAIALRT_INX 汎用モジュールのパフォーマンスの向上

SAP エージェントに対する /IBMMON/ITM_MAIALRT_INX 汎用モジュールのパフォーマンスを向上させることができます。

手順

/IBMMON/ITM_MAIALRT_INX 汎用モジュールのパフォーマンスを向上させるには、以下の手順を実行します。

1. SAP エージェント GUI にログオンします。
2. SE16 トランザクション・コードを実行し、表の名前を /IBMMON/ITM_CNFG と入力し、F7 キーを押します。
3. F5 キーを押すか、「**エントリーの作成 (Create Entries)**」をクリックし、IBMMON/ITM_CNFG 表に次のエントリーを追加します。

表 *206. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
MAI_ALERTS_PERF	YES

注:

- トランザクション・コード /N/IBMMON/ITM_CONFIG でクリティカル定数が設定されない場合、デフォルト値は 2500 です。
- このプロセスは、PERIOD_START および PERIOD_END が明示的に指定されていない状況で MAI アラートを SAP システムからフェッチする 場合にのみ適用されます。

要確認 : 汎用モジュール /IBMMON/ITM_MAIALRT_INX は、トランザクション・コード /N/IBMMON/ITM_CONFIG で設定されるクリティカル定数と等しい数の MAI アラートをフェッチするようになりました。

- /IBMMON/ITM_CNFG 内のこのエントリーがデフォルトで作成されない場合、最新の 2500 個の MAI アラートがフェッチされます。
- フェッチされる行数は、トランザクション・コード /n/IBMMON/ITM_CONFIG で設定されるクリティカル定数の値と常に等しくなります。

CEN CCMS レポート

集中 (CEN) Computing Center Management System (CCMS) は、SAP のモニター機能の 1 つです。

この機能を使用して、複数の SAP システムの CCMS アラートを中央モニター・ハブに報告します。 SAP 環境を単一の CCMS コンソールからモニターします。集中 CCMS レポートは、以下の環境での使用に最適です。

- 主に CCMS 操作であり、必要なモニター・データが CCMS アラートのみである。
- 集中 CCMS は SAP 環境の部分である。
- ISV や ISP などの多くの SAP システムが存在する大規模な SAP 環境。
- IBM Tivoli Monitoring V5.x の SAP エージェント CCMS アダプターとの統合。
- 非 ABAP の SAP コンポーネントおよびアプリケーション・サーバーから、アラートを収集する。

SAP エージェントでは、アラートのみのレポートのための集中 CCMS をサポートします。このため、中央 SAP システムに SAP エージェントを 1 つ配置して、SAP 環境全体の CCMS アラートを表示します。このサポートは、以下の方法で提供されます。

- エージェントでは、CCMS アラートをレポートする際に、そのアラートがエージェントによって直接モニターされる SAP システムに関連付けられているかどうかを検査します。エージェントは、アラートが別の SAP システムに属すると判断すると、集中 CCMS の役割を果たして、追加の R3_Group 管理対象システムを自動的に作成します。
- <local_SID>-All_CCMS_alerts:Grp 管理対象システムを使用して、すべてのリモート SAP システムからアラートの完全セットをレポートします。<local_SID> の値は、直接モニターされている SAP システムのシステム ID です。例えば、ローカル SAP システムが QA1 の場合、このグループ名は QA1-All_CCMS_alerts:Grp になります。
- <local_SID>-<remote_SID>_CCMS_alerts:Grp 管理対象システムを使用して、1 つのリモート SAP システムのすべてのアラートをレポートします。<local_SID> の値は、直接モニターされている SAP システムのシステム ID です。<remote_SID> の値は、リモート SAP システムのシステム ID です。例えば、ローカル SAP システムが QA1 であり、リモート SAP システムが QA2 の場合、このグループ名は QA1-QA2_CCMS_alerts:Grp になります。
- これらの各管理対象システムはナビゲーター・ツリー内でウィジェットの完全セットを持ちますが、意味があるデータを持つのはアラート・ウィジェットのみです。

SAP エージェントは、直接管理する SAP システムの Advanced Business Application Programming (ABAP) コードで集中 CCMS グループの定義を管理します。集中アラートを受け取る SAP システムも SAP エージェントの別のインスタンスによって直接モニターされている場合、これらの定義の変更が必要になる場合があります。アラートを両方のシステムでレポートすることはお勧めしません。集中アラート・レポートは、以下のようにして制限できます。

- /IBMMON/ITM_CONFIG トランザクションを使用して、管理対象グループの保守を実行します。全 CCMS アラート・グループを変更します。グループ定義を編集してリモート・システム ID を除外することにより、このリストからリモート・システムを除去します。
- /IBMMON/ITM_CONFIG トランザクションを使用して、管理対象グループの保守を実行します。<remote_SID> CCMS アラート・グループを削除します。例えば、リモート SAP システムが QA2 の場合、このグループ名は QA2 CCMS alerts になります。

代わりに、集中 CCMS を使用して、すべての SAP システムからのアラートをレポートできますが、ローカルにインストールされた各エージェントからのアラートのレポートが行えなくなります。この構成をセットアップするには、以下の手順を使用します。

- 集中 CCMS システムをモニターするように SAP エージェント のインスタンスを構成します。エージェントが、すべてのリモート SAP システムからのすべてのアラートを検出およびレポートできるようにします。
- 各リモート SAP システムをモニターするように SAP エージェントのインスタンスを構成します。/IBMMON/ITM_CONFIG トランザクションを使用してモニター・セットおよびモニターを選択することにより、これらのエージェント・インスタンスのアラート収集およびレポートを使用不可にします。この機能内で、すべてのモニターのチェック・ボックスをクリアしてこの構成を保管します。

SAP エージェント による集中 CCMS のサポートは、純粋な CCMS モニター環境で使用して、すべてのアラートを共通コンソールに表示します。 また、機能の完全セットとともに使用してリモート SAP システムにシチュエーション、ポリシー、およびアクション実行コマンドを提供したりすることもできます。

SAP システムからの Advanced Business Application Programming (ABAP) トランスポートのアンインストール

ご使用のシステムから SAP エージェントを削除する場合は、DELETE トランスポートを SAP システムにインポートする必要があります。DELETE トランスポートにより、SAP エージェントのディクショナリー・オブジェクトと汎用モジュールが除去されます。

始める前に

SAP システムのバージョンが 7.20 以降の場合は、 DELETE トランスポートをインポートする前に、ご使用のトランスポート・プロファイルにトランスポート・プロファイル・パラメーター **tadirdeletions=true** を追加する必要があります。このトランスポート・プロファイル・パラメーターは tp バージョン 375.57.68 と、R3trans バージョン 6.14 リリース 700 以降でも使用できます。SAP システムからのトランスポート要求の除去について詳しくは、Deleting transport requests を参照してください。

手順

1. 製品 CD の /ABAP ディレクトリーに切り替えます。
2. トランスポート・ファイルを SAP 環境にコピーします。
3. 以下の手順で、K711_00xxx_DELETE ファイルと R711_00xxx_DELETE ファイルを、SAP トランスポート・システムのデータ・ディレクトリーにコピーします。
   a) K711_00xxx_DELETE ファイルを cofiles ディレクトリーにコピーします。
   b) R711_00xxx_DELETE ファイルを data ディレクトリーにコピーします。
4. 以下のコマンドを実行してください。
   a) **tp addtobuffer ITMK711_00xxx_DELETE SID pf=¥usr¥sap¥trans¥bin¥***PROFILE_NAME*
   b) **tp import ITMK711_00xxx_DELETE SID client=nnn U16 pf=¥usr¥sap¥trans¥bin¥** *PROFILE_NAME*
   各項目の意味は次のとおりです。

   SID
   ターゲット SAP システム ID

   PROFILE_NAME
   tp プロファイル・ファイルの名前

   nnn
   エージェントが実行されるターゲット・クライアントの番号

SAP インスタンスのカスタマイズ

デフォルトでは、SAP システムのすべてのインスタンスがモニターされて 、IBM Application Performance Management UI ポータル上に表示されます。

管理者として、モニターする SAP インスタンスを選択します。また、管理者は、モニターしない SAP インスタンスをオフにすることもできます。

/IBMMON/ITM_INSTANCE カスタム・トランザクションは、/IBMMON/ITM_CONFIG トランザクションにリンクされます。

SAP サーバーの選択可能なインスタンスを表示するには、「**SAP インスタンス (SAP Instances)**」オプションを選択します。次に、モニターするインスタンスを選択します。これらのインスタンスは、IBM Application Performance Management UI ポータル上に表示されます。非アクティブなインスタンスやクリアされたインスタンスは、IBM Application Performance Management UI ポータル上に表示されません。

「接続のテスト」機能
「接続のテスト」機能を使用すると、モニター対象の SAP システムへのエージェントの 接続が可能であることを検証できます。

「接続のテスト」手順を実行するには、GUI でパラメーターを入力してください。SAP システムへの接続が正常に行われた場合は、成功メッセージが表示されます。また、接続が失敗した場合には、失敗メッセージが表示されます。

「**接続のテスト**」ボタンは、「IBM Application Performance Management」ウィンドウでのみ使用できます。

CCMS 設計の使用可能化
Computing Center Management System (CCMS) モニターは、最後のサンプル期間からオープンまたはクローズの状態の CCMS レコードを収集するよう拡張されています。サンプル期間はユーザーにより構成することができ、デフォルト値は 3 分です。ただし、SAP エージェントによって参照されるトランスポート・ファイルと Advanced Business Application Programming (ABAP) トランスポートによって参照されるトランスポート・ファイルのバージョンが必ず同じであるようにしてください。

手順

1. SAP システムにログオンします。
2. SE16 トランザクションを開き、/IBMMON/ITM_CNFG 表の名前をそのトランザクションに追加します。
3. /IBMMON/ITM_CNFG ABAP 汎用モジュールを実行して、ABAP プログラムの構成を指定するには、**Enter** キーを押してから **F8** キーを押します。
4. 新規の構成パラメーターを追加する新規項目を作成するには、**F5** キーを押します。
5. **ISNEWCCMSDESIGN** という名前の新規構成パラメーターを、値 *YES* を指定して SAP サーバー上に作成するには、「**PARM NAME**」フィールドに ISNEWCCMSDESIGN と入力し、「**VALUE CHAR**」フィールドに YES と入力します。
6. 「**保存**」をクリックします。
   「VALUE INT」フィールドは無視してかまいません。

アラートのしきい値の変更
アラートに関連付けられた**最大 CCMS アラート**のしきい値を変更することができます。デフォルト値は 1000 で、これは IBM Application Performance Management 内で表示されるアラート数が 1000 であることを示します。これより古いアラートはキャッシュから削除されます。

手順

1. 以下のいずれかのステップを実行します。
   - Windows オペレーティング・システムの場合は <cancle home>¥tmaitm6¥KSAENV ファイルを開きます。
   - Windows 以外のオペレーティング・システムの場合は <candle home>/config/sa.ini ファイルを開きます。
2. ファイルの末尾に *MAX_CCMS_ALERT_THRESHOLD=< Value>* を追加します。

   注 : この値は 100 より大きくなければなりません。

CCMS 設計の使用不可化
Computing Center Management System (CCMS) 設計を使用不可にすることができます。

手順

1. SAP システムにログオンします。

2. SE16 トランザクションを開き、/IBMMON/ITM_CNFG 表の名前をそのトランザクションに追加します。
3. /IBMMON/ITM_CNFG ABAP 汎用モジュールを実行して、ABAP プログラムの構成を指定するには、**Enter** キーを押してから **F8** キーを押します。
4. 既存の項目を削除するには、**ISNEWCCMSDESIGN** を選択して右クリックしてから「**削除**」をクリックします。

SAP HANA データベースのモニターの構成

エージェントがモニター対象の SAP HANA データベース・サーバーのデータを収集できるように、SAP HANA データベース・エージェントを構成する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for SAP HANA Database agent を参照してください。

SAP HANA データベース・エージェントを構成するための前提条件は以下のとおりです。

1. SAP HANA システムのすべてのデータベース (システムおよびテナント) に、以下の特権を持つユーザーを作成します。
   - 役割: モニター
   - システム特権: モニター管理者

   システム・データベースおよびテナント・データベースのユーザー名とパスワードは同じでなければなりません。
2. SAP HANA データベース・エージェント・システムでマスター接続からスタンバイ接続に切り替える場合、エージェントは、スタンバイ・サーバーのホスト名を使用し、そのホスト名はエージェント・システムで解決される必要があります。ホスト名を IP アドレスに解決するためには、エージェントがインストールされているマシンのホスト・ファイルにマッピング・エントリーを追加する必要があります。

注 : マスター・ホストを使用してエージェントを構成する場合は、マスター・ホストの完全修飾ホスト名または IP アドレスを入力します。ユーザーがスタンバイ・ホストを使用してエージェントを構成する場合は、スタンバイ・ホストの完全修飾ホスト名または IP アドレスを入力します。スタンバイ・ノードを使用してエージェントを構成するときは、マスター・ノードがホスト・マシンと共にダウンしている必要があります。

このタスクについて

SAP HANA データベース・エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

手順

- Windows Windows システム上でエージェントを構成するには、以下の手順を実行します。
  a)「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
  b)「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**デフォルトを使用して構成**」をクリックします。
  「**Monitoring Agent for SAP HANA データベース**」ウィンドウが開きます。
  c)「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。

  重要 : エージェント・インスタンス名は、HANA データベースの 3 桁のシステム ID (SID) と同じでなければなりません。例えば、管理対象の SAP HANA データベースの SID が H01 の場合、インスタンス名として H01 を入力します。
  d)「**Monitoring Agent for SAP HANA データベース**」ウィンドウで、以下のフィールドに値を指定します。

インスタンス名
このフィールドのデフォルト値は、「**固有のインスタンス名を入力してください**」フィールドに指定した値と同じです。

サーバー名
システム・データベースがインストールされている SAP HANA サーバーの完全修飾ホスト名または IP アドレス。

データベース名
SAP HANA データベースの名前。

ポート番号
SAP HANA データベース・サーバーのシステム・データベースのインデックス・サーバー・サービスの SQL ポート番号。

HANA DB 管理者 (HANA DB Administrator)
SAP HANA データベース・サーバーにアクセスするためのユーザー名。

HANA DB 管理者パスワード (HANA DB Administrator Password)
SAP HANA データベース・サーバーにアクセスするためのパスワード。

HANA DB 管理者パスワードの確認 (Confirm HANA DB Administrator Password)
「**HANA DB 管理者パスワード (HANA DB Administrator Password)**」フィールドに指定したパスワード。

e)「**OK**」をクリックします。

f)「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

- Linux AIX Linux または AIX システム上でエージェントを構成するには、以下の手順を実行します。

a) コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
例: `/opt/ibm/apm/agent/bin`

b) 以下のコマンドを実行します。ここで、instance_name は、インスタンスに付ける名前です。
`./sap_hana_database-agent.sh config instance_name`

重要 : インスタンス名は、HANA データベースの 3 桁のシステム ID (SID) と同じでなければなりません。管理対象の SAP HANA データベースの SID が H01 の場合、インスタンス名として H01 を入力します。

c) コマンド行に次のメッセージが表示されたら、「1」を入力して Enter キーを押します。 `Edit 'Monitoring Agent for SAP HANA Database' setting? [1=Yes, 2=No]`

d) エージェントの以下のパラメーターの値を指定します。

サーバー名
システム・データベースがインストールされている SAP HANA サーバーの完全修飾ホスト名または IP アドレス。

データベース名
SAP HANA データベースの名前。

ポート番号
SAP HANA データベース・サーバーのシステム・データベースのインデックス・サーバー・サービスの SQL ポート番号。

HANA DB 管理者 (HANA DB Administrator)
SAP HANA データベース・サーバーにアクセスするためのユーザー名。

HANA DB 管理者パスワード (HANA DB Administrator Password)
SAP HANA データベース・サーバーにアクセスするためのパスワード。

HANA DB 管理者パスワードの確認 (Confirm HANA DB Administrator Password)
「**HANA DB 管理者パスワード (HANA DB Administrator Password)**」フィールドに指定したパスワード。

e) 次のコマンドを実行して、SAP HANA データベース・エージェントを始動します。

./sap_hana_database-agent.sh start *instance_name*

- サイレント応答ファイルを使用してエージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで、*install_dir*¥samples のパスにある sap_hana_silent_config.txt ファイルを開き、すべてのパラメーターの値を指定します。

  Windows C:¥IBM¥APM¥samples

  Linux AIX /opt/ibm/apm/agent/samples

  b) コマンド行で、パスを *install_dir*¥bin に変更します。

  c) 次のコマンドを実行します。

  Windows sap_hana_database-agent.bat config *instance_name install_dir* ¥samples¥sap_hana_silent_config.txt

  Linux AIX sap_hana_database-agent.sh config *instance_name install_dir*¥samples¥sap_hana_silent_config.txt

  d) エージェントを開始します。

  Windows 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

  Linux AIX 次のコマンドを実行します: ./sap_hana_database-agent.sh start *instance_name*

  製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

SAP NetWeaver Java スタックのモニターの構成

SAP NetWeaver Java Stack エージェントが、モニター対象の SAP NetWeaver アプリケーション・サーバーのリソース・モニター・データを収集できるように、エージェントを構成する必要があります。トランザクション・トラッキング・データおよび診断データをモニターするには、いくつかの構成タスクを実行する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for SAP NetWeaver Java Stack agent を参照してください。

エージェントを構成する前に、以下の前提条件タスクを実行してください。

- 以下の JAR ファイルを bin ディレクトリーにコピーします。
  - sapj2eeclient.jar (JMX アダプターを 含む SAP J2EE エンジン・クライアント API)
  - logging.jar (ロギング・ライブラリー)
  - com_sap_pj_jmx.jar (SAP-JMX ライブラリー)
  - exception.jar (SAP 例外フレームワーク)

  bin ディレクトリーは、以下のパスにあります。

  Windows candle_home¥TMAITM6_x64

Linux candle_home/interp/sv/bin

重要 : JAR ファイルは、サポートされるすべてのオペレーティング・システムで同じです。これらのファイルは、診断エージェント・パッチまたは Software Update Manager (SUM) から使用可能です。

- 「環境変数」で、パス変数に *<candleHome>¥svdchome¥<build number>¥toolkit¥lib¥win64¥ttapi* を追加します。
- トランザクション・トラッキング・データおよび診断データを収集するゲスト・ユーザーに NWA_READONLY 役割を割り当てます。

このタスクについて

SAP NetWeaver Java Stack エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

- Windows システム上でエージェントを構成するには、GUI またはサイレント応答ファイルを使用できます。
- Linux システムまたは AIX システム上でエージェントを構成するには、コマンド行またはサイレント応答ファイルを使用できます。

トランザクション・トラッキング・データおよび診断データの収集を構成するには、以下のタスクを実行します。

1. データ・コレクターを構成します。詳しくは、768 ページの『データ・コレクターの構成』を参照してください。
2. トランザクション・トラッキング・データおよび診断データの収集を有効にします。詳しくは、769 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』を参照してください。

このトピックに記載されている指示は、明記されている場合を除き、最新リリースのエージェント用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン』を参照してください。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。

始める前に

764 ページの『SAP NetWeaver Java スタックのモニターの構成』トピックの『始める前に』セクションにリストされているファイルが bin ディレクトリー内で使用可能であることを確認してください。

このタスクについて

SAP NetWeaver Java Stack エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**エージェントを構成します**」をクリックします。

   「**Monitoring Agent for SAP NetWeaver Java Stack**」ウィンドウが開きます。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。

   重要 : エージェント・インスタンス名は、SAP NetWeaver Java Stack の 3 桁のシステム ID (SID) と同じでなければなりません。例えば、管理対象の SAP NetWeaver Java Stack の SID が P14 の場合、インスタンス名として P14 を入力します。

4. 「**Monitoring Agent for SAP NetWeaver Java Stack**」ウィンドウで、構成パラメーターの値を指定し、「**OK**」をクリックします。

   構成パラメーターについては、771 ページの『エージェントの構成パラメーター』を参照してください。
5. 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるリソース・モニター・データをダッシュボードに表示します。Cloud APM コンソールの使用法については、『Performance Management コンソールの始動』を参照してください。
- トランザクション・トラッキング・データおよび診断データを収集するには、データ・コレクターを構成し、トランザクション・トラッキングおよび診断のためのデータ収集を有効にします。

Linux または AIX システム上でのエージェントの構成

Linux または AIX システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

始める前に

764 ページの『SAP NetWeaver Java スタックのモニターの構成』トピックの『始める前に』セクションにリストされているファイルが bin ディレクトリー内で使用可能であることを確認してください。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。

   Linux `/opt/ibm/apm/agent/bin`

   Linux AIX `/opt/ibm/apm/agent/bin`
2. 次のコマンドを実行します。

   ```
   ./sap_netweaver_java_stack-agent.sh config instance_name
   ```

   ここで、*instance_name* は、インスタンスに指定する名前です。

   重要 : エージェント・インスタンス名は、SAP NetWeaver Java Stack の 3 桁のシステム ID (SID) と同じでなければなりません。例えば、管理対象の SAP NetWeaver Java Stack の SID が P14 の場合、インスタンス名として P14 を入力します。
3. コマンド行に以下のメッセージが表示されたら、1 と入力して Enter キーを押します。

   ```
   Edit 'Monitoring Agent for SAP NetWeaver Java Stack' setting? [1=Yes, 2=No]
   ```
4. プロンプトが表示されたら、構成パラメーターの値を指定します。

   構成パラメーターについては、771 ページの『エージェントの構成パラメーター』を参照してください。
5. 以下のコマンドを実行して、エージェントを開始します。

   ```
   ./sap_netweaver_java_stack-agent.sh start instance_name
   ```

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるリソース・モニター・データをダッシュボードに表示します。Cloud APM コンソールの使用法については、『Performance Management コンソールの始動』を参照してください。
- トランザクション・トラッキング・データおよび診断データを収集するには、データ・コレクターを構成し、トランザクション・トラッキングおよび診断のためのデータ収集を有効にします。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

始める前に

764 ページの『SAP NetWeaver Java スタックのモニターの構成』トピックの『始める前に』セクションにリストされているファイルが bin ディレクトリー内で使用可能であることを確認してください。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. テキスト・エディターで、以下のパスにある sap_netweaver_java_stack_silent_config.txt ファイルを開き、構成パラメーターの値を指定します。

   Windows C:¥IBM¥APM¥samples

   Linux AIX /opt/ibm/apm/agent/samples

   構成パラメーターについては、771 ページの『エージェントの構成パラメーター』を参照してください。
2. コマンド行で、パスを install_dir¥bin に変更します。
3. 次のコマンドを実行します。

   Windows sap_netweaver_java_stack-agent.bat config *instance_name* install_dir¥samples¥sap_netweaver_java_stack_silent_config.txt

   Linux AIX ./sap_netweaver_java_stack-agent.sh config *instance_name* install_dir¥samples¥sap_netweaver_java_stack_silent_config.txt
4. エージェントを開始します。

   Windows 「IBM Cloud Application Performance Management」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックします。あるいは、次のコマンドを実行することもできます。sap_netweaver_java_stack-agent.bat start *instance_name*

   Linux AIX 次のコマンドを実行します。./sap_netweaver_java_stack-agent.sh start *instance_name*

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるリソース・モニター・データをダッシュボードに表示します。Cloud APM コンソールの使用法については、『Performance Management コンソールの始動』を参照してください。
- トランザクション・トラッキング・データおよび診断データを収集するには、データ・コレクターを構成し、トランザクション・トラッキングおよび診断のためのデータ収集を有効にします。

データ・コレクターの構成

モニターするアプリケーション・サーバー・インスタンスごとにデータ・コレクターを構成できます。

始める前に

764 ページの『SAP NetWeaver Java スタックのモニターの構成』トピックの『始める前に』セクションにリストされているファイルが bin ディレクトリー内で使用可能であることを確認してください。

手順

プロンプトに応答を入力してデータ・コレクターを構成する場合は、以下のステップを実行します。

1. コマンド行で、パスを Windows `install_dir¥svdchome¥build no¥bin¥configNW` または Linux AIX `install_dir/svdchome/build no/bin/configNW` ディレクトリーに変更し、次のスクリプトを実行します。

   Windows config.bat

   Linux AIX config.sh
2. データ・コレクターを構成する製品に対応する番号を入力して NetWeaver Server バージョンを選択し、Enter キーを押します。
3. ユーザー名を求めるプロンプトが出されたら、SAP NetWeaver Application Server with Java Stack で構成されているユーザー名を入力し、Enter キーを押します。
4. パスワードを求めるプロンプトが出されたら、パスワードを入力し、Enter キーを押します。
5. パスワードの再入力を求めるプロンプトが出されたら、パスワードを再入力し、Enter キーを押します。
6. P4 ポート番号を求めるプロンプトが出されたら、ローカル・コンピューターで使用可能な SAP NetWeaver Application Server インスタンスの P4 ポート番号を入力し、Enter キーを押します。

   重要 : P4 ポート番号を計算するには、50000 + (インスタンス番号*100) + 4 の式を使用します。
7. NetWeaver Server インスタンス番号の選択を求めるプロンプトが出されたら、構成するインスタンスに対応する番号を入力し、Enter キーを押します。

   要確認 : インスタンスごとにデータ・コレクターを別個に構成する必要があります。
8. Java ホームへのパスの入力を求めるプロンプトが出されたら、SAP インスタンスから JAVA_HOME を使用します。例: `E:¥usr¥sap¥J01¥J04¥exe¥sapjvm_6`
9. プロンプトが出されたら、トランザクション・トラッキング・データの収集を有効にする場合は、1 を入力します。そうしない場合は、2 を入力し、Enter キーを押します。
10. プロンプトが出されたら、診断データの収集を有効にする場合は、1 を入力します。そうしない場合は、2 を入力し、Enter キーを押します。

タスクの結果

クラス・ファイルをロードするためのパスが生成されます。

次のタスク

1. 生成されたパスを該当する環境変数に追加します。

   Windows *PATH*

   Linux *LD_LIBRARY_PATH* および *LIBPATH*

   AIX *LD LIBRARY PATH* および *LIB PATH*

   要確認 :

   Windows 生成されたパスを *PATH* 環境変数に追加します。

   Linux 生成されたパスを `/home/sidadm/.cshrc` ファイル内の *LD_LIBRARY_PATH* および *LIBPATH* に以下の形式で追加します。

```
setenv LD_LIBRARY_PATH /path
setenv LIBPATH /path
```

AIX 生成されたパスを /etc/environment ファイル内の *LD LIBRARY PATH* および *LIB PATH* に以下の形式で追加します。

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path
LIBPATH=$LIBPATH:/path
```

2. アプリケーション・サーバー・インスタンスを再始動します。
3. トランザクション・トラッキングおよび診断のためのデータ収集を有効にします。詳しくは、769 ページの『トランザクション・トラッキング・データおよび診断データの収集の有効化』を参照してください。

トランザクション・トラッキング・データおよび診断データの収集の有効化

「**エージェント構成**」ページで、トランザクション・トラッキングまたは診断用のデータ収集を有効または無効にすることができます。

始める前に

データ・コレクターが構成されていることを確認します。詳しくは、768 ページの『データ・コレクターの構成』を参照してください。

このタスクについて

トランザクション・トラッキング・データの収集を有効にすると、エージェントは、以下のコンポーネントのデータを収集します。

- サーブレット JSP
- リモート EJB
- JMS

手順

SAP NetWeaver アプリケーション・サーバーの各インスタンスのデータ収集を構成するには、以下のステップを実行します。

1. Cloud APM コンソール にログインします。
2. ナビゲーション・バーで、 「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
3. 「**NetWeaver**」タブをクリックします。
4. データ収集を構成する SAP NetWeaver アプリケーション・サーバー・インスタンスのチェック・ボックスを選択し、「**アクション**」リストから以下のいずれかのアクションを実行します。
   - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。選択した SAP NetWeaver アプリケーション・サーバーの各インスタンスの「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
   - 診断データの収集を有効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**診断モードのみの有効化 (Enabled Diagnostic Mode Only)**」を選択します。選択した SAP NetWeaver アプリケーション・サーバーの各インスタンスの「診断モード」列の状況が「有効」に更新されます。
   - 診断データの収集とメソッド・トレースを有効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**診断モードおよびメソッド・トレースの有効化**」を選択します。選択した SAP NetWeaver アプリケーション・サーバーの各インスタンスの「診断モード」列と「メソッド・トレース」列の状況が「有効」に更新されます。
   - トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。選択した SAP NetWeaver アプリケーショ

ン・サーバーの各インスタンスの「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。

- 診断データの収集を無効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」>「**診断モードおよびメソッド・トレースの無効化**」をクリックします。選択した SAP NetWeaver アプリケーション・サーバーの各インスタンスの「**診断モード**」**列と**「**メソッド・トレース**」**列**の状況が「無効」に更新されます。

タスクの結果
SAP NetWeaver アプリケーション・サーバーの各インスタンスのデータ収集が構成されました。

次のタスク
Cloud APM コンソールにログインして、エージェントにより収集されるトランザクション・トラッキングおよび診断データをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

データ・コレクター構成の削除

SAP Netweaver Application Server with Java Stack インスタンスに対するデータ・コレクターの構成時に行われた変更をロールバックできます。

手順

プロンプトに応答してデータ・コレクター構成を削除するには、以下のステップを実行します。

1. コマンド行で、パスを Windows `install_dir¥svdchome¥build no¥bin¥configNW` または Linux AIX `install_dir/svdchome/build no/bin/configNW` ディレクトリーに変更し、次のスクリプトを実行します。

   Windows unconfig.bat

   Linux AIX unconfig.sh

   データ・コレクターが構成されているすべてのインスタンスがリストされます。
2. データ・コレクター構成を削除するインスタンスに対応する番号を入力し、Enter キーを押します。

   ヒント : 複数のインスタンスのデータ・コレクター構成を削除するために、インスタンスに対応する番号をコンマで区切って入力します。すべてのインスタンスのデータ・コレクター構成を削除する場合は、以下のスクリプトを実行できます。

   Windows config.bat -a

   Linux AIX config.sh -a

次のタスク
SAP NetWeaver AS with Java Stack インスタンスを再始動します。

SAP NetWeaver アプリケーション・サーバー・インスタンスの復元

復元ユーティリティーを 使用すると、SAP NetWeaver データ・コレクターの構成後に SAP NetWeaver アプリケーション・サーバー・インスタンスが始動しない場合に JVM パラメーターを復元することや、SAP NetWeaver アプリケーション・サーバー・インスタンスを復元することができます。

手順

プロンプトに応答を入力して SAP NetWeaver アプリケーション・サーバー・インスタンスを復元する場合は、以下のステップを実行します。

1. コマンド行で、パスを Windows `install_dir¥svdchome¥build no¥bin¥configNW` または Linux AIX `install_dir/svdchome/build no/bin/configNW` ディレクトリーに変更し、次のスクリプトを実行します。

   Windows `restoreNW.bat`

Linux AIX `restoreNW.sh`

2. JVM パラメーターを復元する製品に対応する番号を入力して、NetWeaver サーバーのバージョンを選択し、Enter キーを押します。
3. ユーザー名を求めるプロンプトが出されたら、SAP NetWeaver アプリケーション・サーバー・インスタンスのユーザー名を入力し、Enter キーを押します。
4. ユーザー・パスワードを求めるプロンプトが出されたら、SAP NetWeaver アプリケーション・サーバー・インスタンスのユーザー・パスワードを入力し、Enter キーを押します。
5. P4 ポート番号を求めるプロンプトが出されたら、ローカル・コンピューターで使用可能な SAP NetWeaver Application Server インスタンスの P4 ポート番号を入力し、Enter キーを押します。

   入力した P4 ポートを使用してもインスタンス情報が見つからない場合は、「SAP NetWeaver サーバーに接続できませんでした (Could not connect to SAP NetWeaver Server)」というメッセージが表示され、NetWeaver サーバー・インスタンスのホームへのパスを指定するようプロンプトが出されます。

   この場合は、usr¥sap¥*System_Name*¥*Instance_Number* のように指定します。
6. NetWeaver サーバーのインスタンス番号の選択を求めるプロンプトが出されたら、復元するインスタンスに対応する番号を入力し、Enter キーを押します。

タスクの結果

以下のメッセージが表示されます。

復元が正常に行われました。(Restore successful.)インスタンスを再始動してください。(Please restart the instance.)

エージェントの構成パラメーター

SAP NetWeaver Java Stack エージェントを構成するときに、SAP_NETWEAVER_P4_HOSTNAME や SAP_NETWEAVER_P4_PORT などのパラメーターのデフォルト値を変更できます。

以下の表で、SAP NetWeaver Java Stack エージェントの構成パラメーターについて詳しく説明します。これらのフィールドは必須であるため、すべてのフィールドの値を指定する必要があります。

表 *207.* 構成パラメーターの名前と説明

パラメーター名	説明
インスタンス名	The name of the instance. このフィールドのデフォルト値は、「**固有のインスタンス名を入力してください**」フィールドに指定した値と同じです。
SAP_NETWEAVER_P4_HOSTNAME	SAP NetWeaver アプリケーション・サーバーのホスト名または IP アドレス。
SAP_NETWEAVER_P4_PORT	SAP NetWeaver アプリケーション・サーバーの P4 ポート番号。
SAP_NETWEAVER_P4_USERNAME	SAP NetWeaver アプリケーション・サーバーにアクセスするための管理者のユーザー名。
SAP_NETWEAVER_P4_PASSWORD	SAP NetWeaver アプリケーション・サーバーにアクセスするための管理者のパスワード。
Confirm SAP_NETWEAVER_P4_PASSWORD	**SAP_NETWEAVER_P4_PASSWORD** パラメーターに指定したパスワード。

Siebel モニターの構成

Siebel エージェントは、Siebel 統計、ユーザー・セッション、コンポーネント、タスク、アプリケーション・サーバー、Siebel Gateway Name Server、プロセス CPU およびメモリー使用量、ログ・イベントのモニターなど、Siebel リソースに対するモニターの中心点となります。

始める前に

- 771 ページの『Siebel モニターの構成』のトピック全体を読んで、構成を完了するために何が必要かを判断します。
- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、Siebel エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Siebel エージェントを参照してください。
- Siebel エージェントを構成する前に、Siebel エージェントで使用される Siebel ユーザー・アカウントの確認を行う必要があります。

デフォルトでは、コンポーネントごとの統計モニターは無効になっています。コンポーネントごとの統計モニターの有効化を行うことができます。

このタスクについて

Siebel エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で始動する必要があります。

手順

1. Windows システム上でエージェントを構成するには、「IBM Performance Management」ウィンドウまたはサイレント応答ファイルを使用できます。
   - 774 ページの『Windows システムでのエージェントの 構成』.
   - 779 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. Linux システムおよび UNIX システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。
   - 778 ページの『プロンプトへの応答によるエージェントの構成』.
   - 779 ページの『サイレント応答ファイルを使用したエージェントの構成』.

次のタスク

Cloud APM コンソールで Application Performance ダッシュボード・ページに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Siebel ユーザー・アカウントの確認

Siebel エージェントを構成する前にそのエージェントの実行に使用するユーザー・アカウントを確認する必要があります。

このタスクについて

Siebel エージェントの実行に使用するユーザー・アカウントは、Siebel **srvrmgr** コマンド行ユーティリティーを実行するための許可を備えている必要があります。ユーザー・アカウントが必要な許可を備えていることを確認するには、以下のステップを実行します。

手順

1. Siebel エージェントの実行に使用するユーザー・アカウントを使用して、コンピューターにログオンします。
2. Siebel サーバーがインストールされているディレクトリーに 移動します。
3. Siebel 環境ファイルをソースとして読み込んで実行します。
   source siebenv.sh
4. 次のコマンドを実行します。

```
srvrmgr /s Siebel_server /g Siebel_gateway /e Siebel_enterprise
/u useraccount /p password
/c "list servers"
```

ここで、

Siebel_server
Siebel アプリケーション・サーバーの名前。

Siebel_gateway
現在アクティブなゲートウェイ ・ネーム・サーバーの名前。

Siebel_enterprise
Siebel エンタープライズの 名前。

useraccount
コンピューターへのログオンに使用するユーザー・アカウント。

パスワード
ユーザー・アカウントに関連付けられたパスワード。

ユーザー・アカウントが必要な許可を備えている場合、以下の例のような出力が表示されます (返されたフィールドは、3 つに限定)。

```
Connected to 1 server(s) out of a total of 1 server(s) in the enterprise

srvrmgr:s82win8> list servers show SBLSRVR_NAME, HOST_NAME, SBLSRVR_STATUS

SBLSRVR_NAME  HOST_NAME  SBLSRVR_STATUS
------------  ---------  --------------------------------
s82win8       s82win8    16.0.0.0 [23057] LANG_INDEPENDENT

1 row returned.
```

srvrmgr コマンドが正常に実行されなかった場合、サーバーの Siebel 管理者に問い合わせください。ユーザー・アカウント用に必要な Siebel 環境変数を設定し、ユーザー・アカウントが **srvrmgr** コマンドを実行するための適切な許可を備えるようにしてください。

コンポーネントごとの統計モニターの有効化

デフォルトでは、コンポーネントごとの統計モニターは無効になっています。KUY_ENABLE_COMP_STATS 環境変数を使用して、コンポーネントごとの統計モニターを有効にすることができます。

始める前に

Siebel V8.1 以降のサーバーにおける既知の問題のため、Siebel コンポーネント統計の収集が、Siebel Gateway Server のメモリー使用に悪影響を及ぼすことがあります。この問題は、『Gateway Service on Siebel 8.1 or 8.2 Might Consume High Memory Consumption: Recovery (Doc ID 1269177.1)』という Oracle 公開の技術情報で解決されます。その記事で問題の修正が用意されています。修正は、Siebel サーバーで実装されます。

コンポーネントごとの統計モニターが環境で必要な場合、Oracle の修正を Siebel V8.1 以降の Gateway Server に適用してから、コンポーネントごとの統計モニターを有効にします。

このタスクについて

Oracle の修正を適用した後に、以下のステップを実行して、Siebel エージェントでコンポーネントごとの統計モニターを有効にします。

手順

1. 以下の Siebel エージェントのエージェント・インストール・ディレクトリーに移動します。
   - Windows *install_dir*¥TMAITM6_x64
   - Linux AIX *install_dir*/config
2. Siebel エージェント構成ファイルを編集して、KUY_ENABLE_COMP_STATS を true に設定します。
   - Windows KUYENV_*instance_name*
   - Linux AIX uy.environment

   ここで、*instance_name* は、Siebel エージェントのインスタンス名です。
3. エージェントを再始動します。

   重要 : この設定をすべての新規エージェント・インスタンスのデフォルトにする場合は、以下の構成テンプレート・ファイルで KUY_ENABLE_COMP_STATS を true に設定します。
   - Windows KUYENV
   - Linux AIX ステップ 2 で uy.environment を編集したので、この設定は既に新規エージェント・インスタンスのデフォルトになっています。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Siebel エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for Siebel**」テンプレートを右クリックし、「**エージェントの構成**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、下線文字、およびマイナス文字のみを使用してください。例えば、siebel01 です。

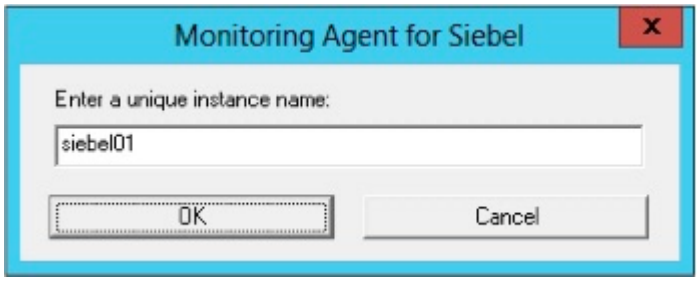


図 22. 固有のインスタンス名を入力するウィンドウ

4. サーバー・タイプを選択し、そのサーバー・タイプの必須フィールドに値を入力してから、「**次へ**」をクリックします。

   各構成パラメーターの説明については、780 ページの表 208 を参照してください。

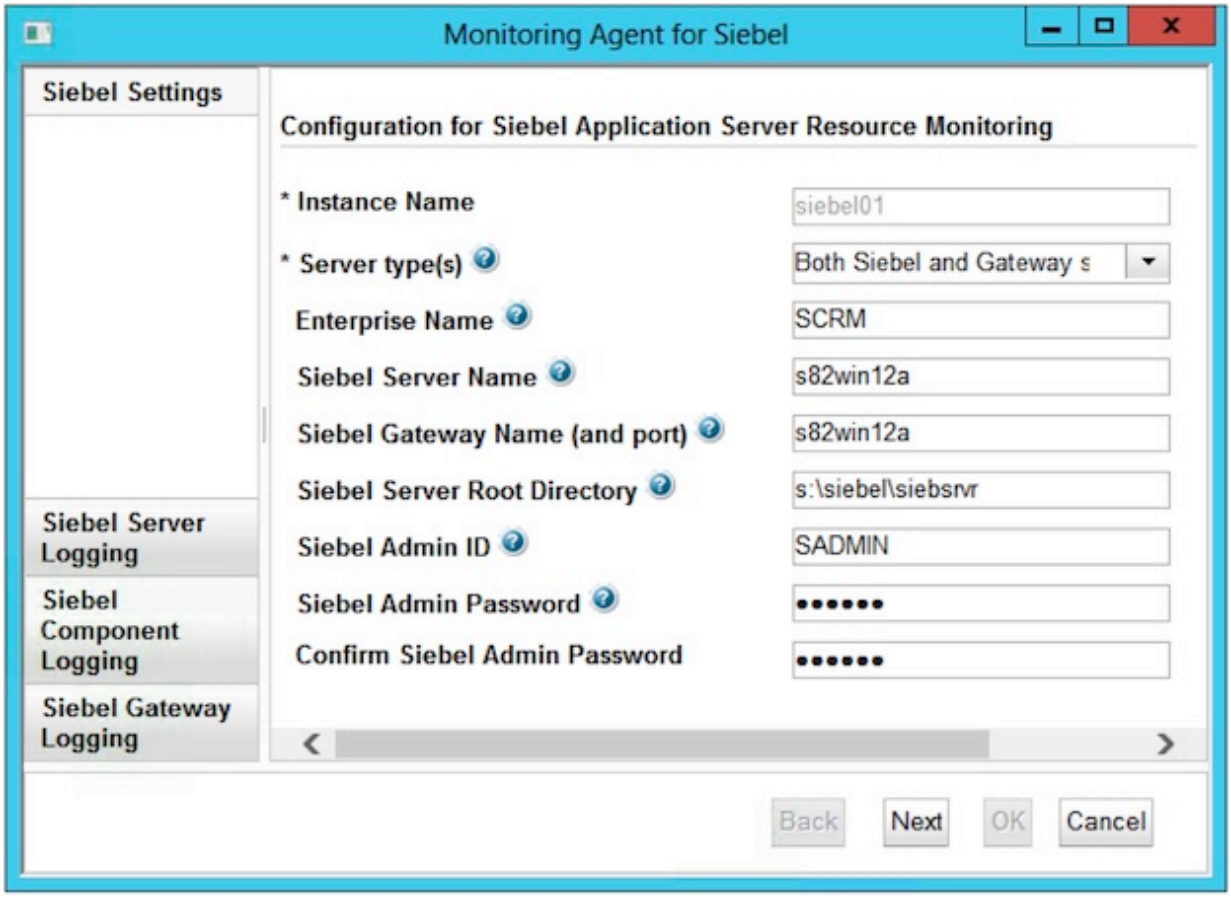


図 *23. Siebel* ホストにインストールされている *Siebel* サーバー・タイプの構成パラメーターのウィンドウ

重要 : Siebel Gateway Name Server があるが Siebel Server はないコンピューター上に Siebel エージェントがインストールされている場合、アプリケーション・ダッシュボードに表示されるデータは、当該インスタンスの Siebel Gateway Name Server にのみ適用できます。他のすべての Siebel エージェント・ビューは空です。

5. オプション: Siebel サーバー・ロギングの値を編集してから、「**次へ**」をクリックします。
   各構成パラメーターの説明については、781 ページの表 209 を参照してください。

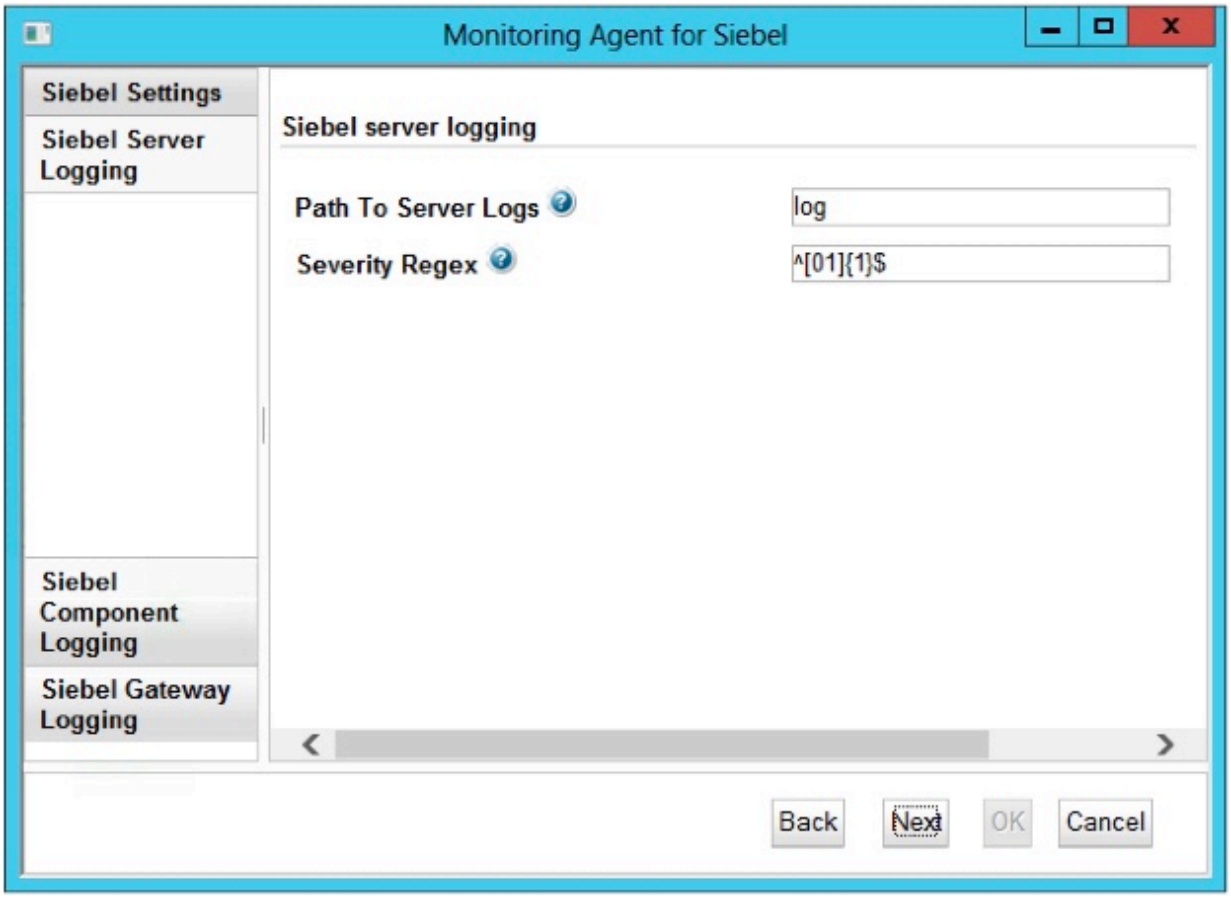


図 *24. Siebel* サーバー・ロギング設定を指定するウィンドウ

6. オプション: Siebel コンポーネント・ロギングの値を編集してから、「**次へ**」をクリックします。デフォルトでは、783 ページの表 212 に示されているコンポーネントのログが Siebel エージェントによってモニターされます。モニター対象にするコンポーネント・ログ (最大で 10) を追加するには、対応するコンポーネントの別名を指定します (例えば、`SCBroker`)。

   各構成パラメーターの説明については、782 ページの表 210 を参照してください。

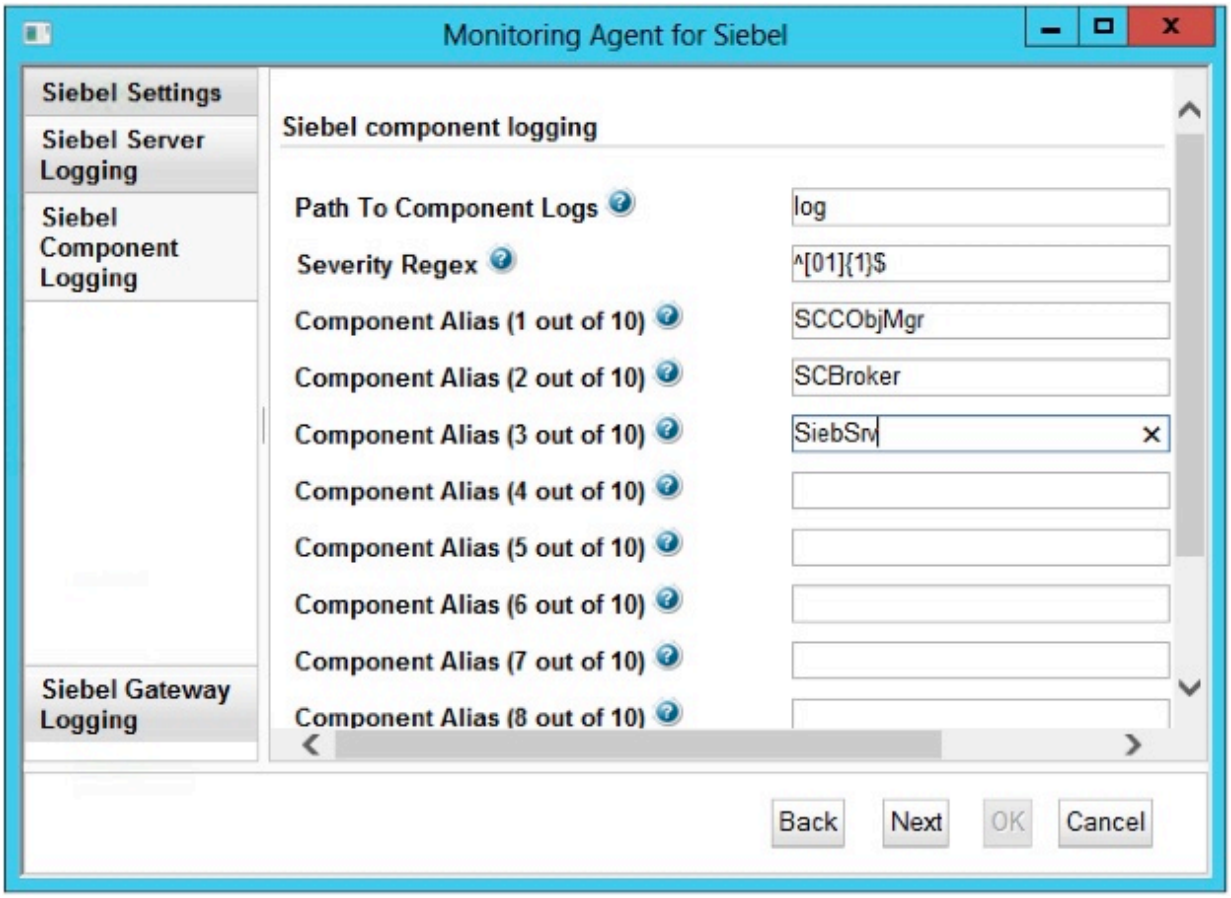


図 *25.* モニターする追加コンポーネント・ログを指定するウィンドウ

7. オプション: Siebel ゲートウェイ・ロギングの値を編集します。
   各構成パラメーターの説明については、782 ページの表 211 を参照してください。

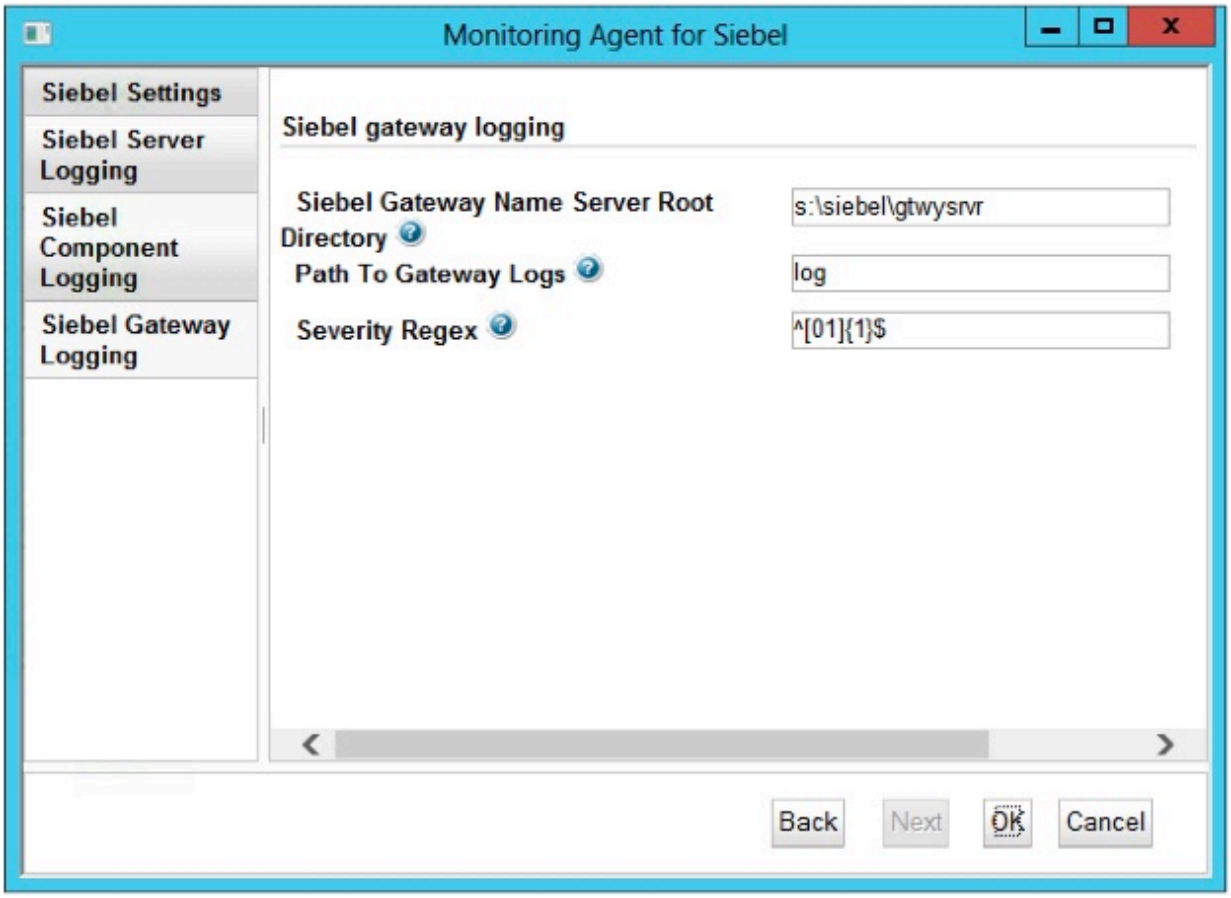


図 *26. Siebel* ゲートウェイ・ロギング設定を指定するウィンドウ

8. 「**OK**」をクリックして、構成を完了します。
9. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

Siebel エージェントのインストール後、エージェントを開始する前に構成する必要があります。Siebel エージェントがローカル Linux または UNIX コンピューターにインストールされている場合、以下の説明に従って、コマンド行プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

- 以下のステップに従って、スクリプトを実行してプロンプトに応答することで Siebel エージェントを構成します。
  a) コマンド・ラインで、以下のコマンドを実行します。

  install_dir/bin/siebel-agent.sh config **instance_name**

  ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスに指定する名前です。

  例

  ```
  /opt/ibm/apm/agent/bin/siebel-agent.sh config example-inst01
  ```

b) プロンプトに応答してエージェントの構成値を設定します。

各構成パラメーターの説明については、780 ページの『Siebel エージェントの構成パラメーター』を参照してください。

c) 以下のコマンドを実行して、エージェントを開始します。

install_dir/bin/siebel-agent.sh start **instance_name**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

例

```
/opt/ibm/apm/agent/bin/siebel-agent.sh start example-inst01
```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで Siebel エージェントを構成するには、以下の手順を実行します。

  a) テキスト・エディターで、siebel_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。

  – Linux AIX *install_dir*/samples/siebel_silent_config.txt

  – Windows *install_dir*¥samples¥siebel_silent_config.txt

  ここで、*install_dir* はエージェントのインストール先パスです。

  例

  – Linux AIX /opt/ibm/apm/agent/samples/siebel_silent_config.txt

  – Windows C:¥IBM¥APM¥samples¥siebel_silent_config.txt

  b) siebel_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

  各構成パラメーターの説明については、780 ページの『Siebel エージェントの構成パラメーター』を参照してください。

  c) siebel_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。

  – Linux AIX *install_dir*/bin/siebel-agent.sh config **instance_name** *install_dir*/samples/siebel_silent_config.txt

  – Windows *install_dir*¥bin¥siebel-agent.bat config **instance_name** *install_dir*¥samples¥siebel_silent_config.txt

  ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

例

- Linux AIX `/opt/ibm/apm/agent/bin/siebel-agent.sh config example-inst01 /opt/ibm/apm/agent/samples/siebel_silent_config.txt`
- Windows `C:¥IBM¥APM¥bin¥ siebel-agent.bat config example-inst01 C:¥IBM¥APM¥samples¥siebel_silent_config.txt`

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux AIX *install_dir*`/bin/siebel-agent.sh start` **instance_name**
- Windows *install_dir*`¥bin¥siebel-agent.bat start` **instance_name**

ここで、*install_dir* はエージェントのインストール先パスであり、*instance_name* はエージェント・インスタンスの名前です。

例

- Linux AIX `/opt/ibm/apm/agent/bin/siebel-agent.sh start example-inst01`
- Windows `C:¥IBM¥APM¥bin¥siebel-agent.bat start example-inst01`

Siebel エージェントの構成パラメーター

Siebel エージェントの構成パラメーターを、カテゴリーに従ってグループ化した表に表示します。

1. Siebel 設定 - 一般的な Siebel 環境設定。
2. Siebel サーバー・ロギング - Siebel サーバー・ログのモニターに固有の設定。
3. Siebel コンポーネント・ロギング - Siebel コンポーネント・ログのカスタム・リストのモニターに固有の設定。
4. Siebel ゲートウェイ・ロギング - Siebel ゲートウェイ・ログのモニターに固有の設定。

表 *208. Siebel* 設定

パラメーター名	説明	このサーバー・タイプの選択で必須	サイレント構成ファイルのパラメーター名
サーバー・タイプ	ローカル・コンピューターにインストールされているサーバー・タイプを指示します。	• **ゲートウェイ・サーバーのみ** • **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_SERVER_TYPE**
エンタープライズ名	Siebel エンタープライズの名前。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_ENTERPRISE**

表 *208. Siebel* 設定 *(*続き*)*

パラメーター名	説明	このサーバー・タイプの選択で必須	サイレント構成ファイルのパラメーター名
Siebel サーバー名	モニターする Siebel サーバーの名前。 **注 :** この名前は、サーバーのホスト名ではありません。Siebel **srvrmgr** コマンドを実行する際に使用するサーバー名です。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_SERVER**
Siebel ゲートウェイ名 (Siebel Gateway Name)	モニター対象の Siebel ゲートウェイ・ネーム・サーバーとオプション・ポート (例: `gtwysrvr` または `gtwysrvr:1234`)。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_GATEWAY**
Siebel サーバーのルート・ディレクトリー (Siebel Server Root Directory)	Siebel アプリケーション・サーバーのベース・インストール・ディレクトリー。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_INSTALL_ROOT**
Siebel 管理者 ID (Siebel Admin ID)	**srvrmgr** コマンドを実行する際に Siebel エンタープライズでの認証にエージェントが使用する Siebel 固有のユーザー ID。例えば、`SADMIN` です。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_ADMIN_ID**
Siebel 管理者パスワード (Siebel Admin password)	Siebel サーバー管理者のパスワード。	• **Siebel サーバーのみ** • **Siebel サーバーおよびゲートウェイ・サーバー**	**KUY_ADMIN_PASSWORD**

表 *209. Siebel* サーバー・ロギング設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
サーバー・ログのパス (Path To Server Logs)	「Siebel サーバーのルート・ディレクトリー (Siebel Server Root Directory)」を基準としたサーバー・ログの相対パス。Siebel サーバー・ロギングのキャプチャーを無効にするには、任意の無効なパスを入力します。例えば、`xyz` です。	**KUY_SERVER_LOGGING_PATH**

表 *209. Siebel* サーバー・ロギング設定 *(続き)*

パラメーター名	説明	サイレント構成ファイルのパラメーター名
重大度の正規表現 (Severity Regex)	重大度レベルに合わせて Siebel サーバー・ログをキャプチャーするために使用する正規表現。デフォルトの `^[01]{1}$` を使用すると、レベル 0 とレベル 1 のエラーのキャプチャーを行うことができます。	**KUY_SERVER_LOGGING_SEVERITY_REGEX**

表 *210. Siebel* コンポーネント・ロギング設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
コンポーネント・ログのパス (Path To Component Logs)	「Siebel サーバーのルート・ディレクトリー (Siebel Server Root Directory)」を基準としたサーバー・ログの相対パス。Siebel サーバー・ロギングのキャプチャーを無効にするには、任意の無効なパスを入力します。例えば、`xyz` です。	**KUY_COMPONENT_LOGGING_PATH**
重大度の正規表現 (Severity Regex)	重大度レベルに合わせて Siebel サーバー・ログをキャプチャーするために使用する正規表現。デフォルトの `^[01]{1}$` を使用すると、レベル 0 とレベル 1 のエラーのキャプチャーを行うことができます。	**KUY_COMPONENT_LOGGING_SEVERITY_ REGEX**
コンポーネント別名 (*N*/10) (Component Alias (*N* out of 10))	追加のコンポーネントのログをモニターするコンポーネント別名。例: `SCBroker`。ここで、*N* は、1 から 10 のオプション・コンポーネントです。	**KUY_CUSTCOMPLOG_00** から **KUY_CUSTCOMPLOG_09**

表 *211. Siebel* ゲートウェイ・ロギング設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
Siebel ゲートウェイ・ネーム・サーバーのルート・ディレクトリー (Siebel Gateway Name Server Root Directory)	Siebel ゲートウェイ・ネーム・サーバーのベース・インストール・ディレクトリー。	**KUY_GATEWAY_ROOT**

表 211. Siebel ゲートウェイ・ロギング設定 (続き)		
パラメーター名	**説明**	**サイレント構成ファイルのパラメーター名**
ゲートウェイ・ログのパス (Path To Gateway Logs)	「Siebel ゲートウェイ・ネーム・サーバーのルート・ディレクトリー (Siebel Gateway Name Server Root Directory)」を基準にしたゲートウェイ・ログの相対パス。ゲートウェイ・ネーム・サーバー・ロギングのキャプチャーを無効にするには、任意の無効なパスを入力します。例えば、xyz です。	**KUY_GW_LOGGING_PATH**
重大度の正規表現 (Severity Regex)	重大度レベルに合わせて Siebel サーバー・ログをキャプチャーするために使用する正規表現。デフォルトの ^[01]{1}$ を使用すると、レベル 0 とレベル 1 のエラーのキャプチャーを行うことができます。	**KUY_GW_LOGGING_SEVERITY_REGEX**

常にモニターされる Siebel コンポーネント・ログ

10 個の Siebel コンポーネントのコンポーネント・ログが常にモニターされます。

表 212. コンポーネント・ログが常にモニターされる Siebel コンポーネントの別名と名前	
コンポーネント別名	**コンポーネント名**
SCCObjMgr	Call Center Object Manager
SMObjMgr	Marketing Object Manager
SSEObjMgr	Sales Object Manager
CommInboundRcvr	Communications Inbound Receiver
CommOutboundMg	Communications Outbound Manager
CommSessionMgr	Communications Session Manager
WorkMon	Workflow Monitor Agent
WfProcBatchMgr	Workflow Process Batch Manager
WfProcMgr	Workflow Process Manager
SiebSrvr	Siebel サーバー

Sterling Connect Direct モニターの構成

Sterling Connect Direct エージェントが Connect Direct サーバーからデータを収集し、Connect Direct サーバーのファイル転送および正常性の統計をモニターできるように、Sterling Connect Direct エージェントを構成する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for Sterling Connect Direct agent を参照してください。

このタスクについて

- Windows システム上でエージェントを構成するには、「IBM Cloud Application Performance Management」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、Windows システムでエージェントを構成できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**タスク/サブシステム**」列にある「**テンプレート**」を右クリックし、「**エージェントを構成します**」をクリックします。
3. 「**固有のインスタンス名を入力してください**」フィールドにエージェント・インスタンス名を入力し、「**OK**」をクリックします。

   注 : エージェント・インスタンス名の長さを制限します。7 から 10 文字の範囲内の長さをお勧めします。
4. 「**Monitoring Agent for Sterling Connect Direct**」ウィンドウの「**Connect Direct Server の詳細 (Connect Direct Server Details)**」タブで、構成パラメーターの値を指定し、「**OK**」をクリックします。

   構成パラメーターについては、785 ページの『エージェントの構成パラメーター』を参照してください。
5. 「**次へ**」をクリックします。
6. 「Java パラメーター」タブで、デフォルト値のままにして、「**次へ**」をクリックします。
7. 「Java API クライアント構成」タブで、「**OK**」をクリックします。
8. 「**IBM Performance Management**」ウィンドウで、作成したエージェント・インスタンスを右クリックし、「**開始**」をクリックしてエージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

Linux システムでのエージェントの構成

Linux オペレーティング・システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

1. コマンド・ラインで、パスをエージェント・インストール・ディレクトリーに変更します。
   例: `/opt/ibm/apm/agent/bin`
2. コマンド **/sterling_connect_direct-agent.sh config instance_name** を実行します。

   注 : *instance_name* は、エージェント・インスタンスに指定する名前です。
3. コマンド行に、メッセージ「`「Monitoring Agent for Sterling Connect Direct」の設定を編集しますか? [1=はい, 2=いいえ] (Edit 'Monitoring Agent for Sterling Connect Direct' setting? [1=Yes, 2=No])`」が表示されます。
4. 設定を編集するには、1 を入力します。

5. プロンプトが出されたら、構成パラメーターの値を指定します。構成パラメーターについては、785 ページの『エージェントの構成パラメーター』を参照してください。
6. コマンド **./sterling_connect_direct-agent.sh start instance_name** を実行して、エージェントを開始します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルを使用すると、Linux システムおよび Windows システム上で Monitoring Agent for Sterling Connect Direct を構成できます。サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

1. テキスト・エディターで、*install_dir*/samples/sterling_connect_direct_silent_config.txt にあるサイレント応答ファイルを開きます。
2. ファイルにサーバー名、ユーザー名、パスワード、インストール・ディレクトリーを入力し、ファイルを保存します。
3. コマンド・プロンプトで、*install_dir*/bin に移動し、以下のコマンドを実行します。

   Linux AIX `./sterling_connect_direct-agent.sh config <Instance_name> install_dir/samples/sterling_connect_direct_silent_config.txt`

   Windows `./sterling_connect_direct-agent.bat config <Instance_name> install_dir/samples/sterling_connect_direct_silent_config.txt`

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トラブルシューティングのヘルプについては、developerWorks の IBM Cloud APM Forum を参照してください。

エージェントの構成パラメーター

Monitoring Agent for Sterling Connect Direct の構成中に、構成パラメーターの値を設定できます。

以下の表で、Monitoring Agent for Sterling Connect Direct の構成パラメーターについて詳しく説明します。

表 *213.* 構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
インスタンス名	このフィールドのデフォルト 値は、「**固有のインスタンス名を入力してください**」フィールドに指定する値と同じです。	はい

表 213. 構成パラメーターの名前と説明 (続き)		
パラメーター名	**説明**	**必須フィールド**
サーバー名	Sterling Connect Direct サーバーのホスト名または IP。	はい
サーバー・ポート (Server port)	Sterling Connect Direct サーバーのポート。Sterling Connect Direct のデフォルト値は 1363 です。	はい
ユーザー名	Sterling Connect Direct サーバーに接続するためのユーザー名。	はい
パスワード	Sterling Connect Direct サーバーに接続するためのパスワード。	はい
Java ホーム	Java がインストールされているフォルダーのパス。	いいえ
Java トレース・レベル	Java プロバイダーで使用されるトレース・レベル。Sterling Connect Direct のデフォルト値は Error です。	はい
JVM 引数	このパラメーターでは、Java 仮想マシンの引数のリストを指定できます (任意指定)。	いいえ
外部 JAR のクラスパス	Java API データ・プロバイダーで必要となる、エージェントに含まれていない JAR のパス。	いいえ

Sterling File Gateway モニターの構成

Monitoring Agent for Sterling File Gateway は、企業間 (B2B) REST API と File Gateway データベースを使用して、IBM Sterling File Gateway アプリケーションをモニターします。Sterling File Gateway エージェントがデータ・ソースからデータを収集し、Sterling File Gateway アプリケーションの統計および正常性をモニターできるように、Sterling File Gateway エージェントを構成する必要があります。 エージェントは、Windows システムと Linux システムで構成できます。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。Software Product Compatibility Reports for Sterling File Gateway agent を参照してください。
- B2B REST API が File Gateway ノードにインストールされていることを確認してください。B2B REST API インストールについて詳しくは、786 ページの『B2B REST API のインストール』を参照してください。

このタスクについて

Sterling File Gateway エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

- Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

B2B REST API のインストール

Sterling File Gateway ノードに企業間 (B2B) REST API をインストールして構成できます。B2B REST API は、B2B Integrator インストーラー (V5.2.6.2) で使用可能です。

手順

1. `<install_dir>/bin` ディレクトリーにナビゲートします。

   ここで、*install_dir* は、B2B Integrator のエージェント・インストーラーのディレクトリーです。

2. 次のコマンドを実行します。

   - Linux ./InstallService.sh/*install_dir*/bin/b2bAPIs_10000602.jar

     ここで、<*install_dir*> は、メディア・ファイルの内容を解凍した場所です。
   - Windows ./InstallService.cmd/*install_dir*/bin/b2bAPIs_10000602.jar

     ここで、<*install_dir*> は、B2B インストーラーのフォルダーです。

Windows システムでの Sterling File Gateway エージェントの構成

「**IBM Cloud Application Performance Management**」ウィンドウを使用して、Sterling File Gateway エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

Sterling File Gateway エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Sterling File Gateway**」を右クリックしてから「**エージェントの構成**」をクリックします。

   要確認 : 初めてエージェントを構成した後には、「**エージェントの構成**」オプションは使用できません。エージェントを再度構成するには、「**再構成**」をクリックします。
3. 「Sterling File Gateway エージェント」ウィンドウで、以下のステップを実行します。
   a) Sterling File Gateway エージェント・インスタンスの固有の名前を入力し、「**OK**」をクリックします.
   b) 「**B2B API 詳細**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。
   c) 「**データベース詳細**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。
   d) 「**Java API**」タブで、構成パラメーターの値を指定してから、「**OK**」をクリックします。

   「Sterling File Gateway エージェント」ウィンドウの各タブにある構成パラメーターについて詳しくは、以下のトピックを参照してください。

   - 791 ページの『B2B API 詳細の構成パラメーター』
   - 792 ページの『データベース詳細の構成パラメーター』
   - 792 ページの『Java API の構成パラメーター』
4. 「**IBM Performance Management**」ウィンドウで、「**Sterling File Gateway エージェント**」を右クリックし、「**始動**」をクリックします。

Linux システムでの Sterling File Gateway エージェントの構成

Linux オペレーティング・システムで、構成スクリプトを実行し、プロンプトに応答することで、Sterling File Gateway エージェントを構成することができます。

手順

1. コマンド行に移動し、**<install_dir>/bin/sterling_file_gateway-agent.sh config instance_name** コマンドを実行します。

   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* はエージェントのインストール・ディレクトリー・パスです。
2. すべての必須の構成パラメーターに値を指定するように求めるプロンプトが出されます。構成パラメーターのデフォルト値を変更することができます。

構成パラメーターについて詳しくは、以下のトピックを参照してください。

- 791 ページの『B2B API 詳細の構成パラメーター』
- 792 ページの『データベース詳細の構成パラメーター』
- 792 ページの『Java API の構成パラメーター』

3. エージェントを始動するには、**<install_dir>/bin/sterling_file_gateway-agent.sh start instance_name** コマンドを実行します。

サイレント応答ファイルを使用した Sterling File Gateway エージェントの構成

サイレント応答ファイルを使用することで、構成スクリプトを実行するときに、プロンプトに応答することなく、Sterling File Gateway エージェントを構成できます。Windows システムと Linux システムのいずれでも、サイレント応答ファイルを使用するエージェントを構成できます。サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで、Sterling File Gateway エージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで、sterling_file_gatway_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
     - Linux *install_dir*/samples/sterling_file_gatway_silent_config.txt

       例: **/opt/ibm/apm/agent/samples/sterling_file_gateway_silent_config.txt**
     - Windows *install_dir*¥samples¥sterling_file_gateway_silent_config.txt

       例: **C:¥IBM¥APM¥samples¥sterling_file_gateway_silent_config.txt**

  b) sterling_file_gateway_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

     構成パラメーターについて詳しくは、以下のトピックを参照してください。
     - 791 ページの『B2B API 詳細の構成パラメーター』
     - 792 ページの『データベース詳細の構成パラメーター』
     - 792 ページの『Java API の構成パラメーター』

  c) sterling_file_gateway_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
     - Linux *install_dir*/bin/sterling_file_gateway-agent.sh config *instance_name* *install_dir*/samples/sterling_file_gateway_silent_config.txt

       例: **/opt/ibm/apm/agent/bin/sterling_file_gateway-agent.sh config instance_name /opt/ibm/apm/agent/samples/ sterling_file_gateway_silent_config.txt**
     - Windows *install_dir*/bin/sterling_file_gateway-agent.bat config *instance_name* *install_dir*/samples/sterling_file_gateway_silent_config.txt

例: **C:¥IBM¥APM¥bin¥sterling_file_gateway-agent.bat config instance_name C:¥IBM¥APM¥samples¥sterling_file_gateway_silent_config.txt**

ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* はエージェントがインストールされるパスです。

重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

d) 以下のコマンドを実行して、エージェントを開始します。

- Linux *install_dir*/bin/sterling_file_gateway-agent.sh start *instance_name*

  例: **/opt/ibm/apm/agent/bin/sterling_file_gateway-agent.sh start instance_name**

- Windows *install_dir*¥bin¥sterling_file_gateway-agent.bat start *instance_name*

  例: **C:¥IBM¥APM¥bin¥sterling_file_gateway-agent.bat start instance_name**

Linux でのデータ・プロバイダーのエージェント環境変数の構成

Linux オペレーティング・システムで、データ・プロバイダーの Sterling File Gateway エージェント環境変数を構成できます。

このタスクについて

Sterling File Gateway エージェントには、データ・プロバイダー用に構成可能な環境変数が用意されています。

手順

1. <install_dir>/agent/config ディレクトリーに移動します。
2. エディターで .fg.environment ファイルを開き、環境変数を編集します。

   構成可能なエージェント環境変数について詳しくは、790 ページの『データ・プロバイダーの環境変数』を参照してください。

Windows でのデータ・プロバイダーのエージェント環境変数の構成

「**IBM Performance Management**」ウィンドウを使用して、Windows オペレーティング・システムでデータ・プロバイダーの Sterling File Gateway エージェント環境変数を構成できます。

このタスクについて

Sterling File Gateway エージェントには、データ・プロバイダー用の構成可能な環境変数が用意されています。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、エージェント・インスタンスを右クリックし、「**拡張**」 > 「**ENV ファイルの編集**」をクリックし、環境変数のデフォルト値を編集します。

   構成可能なエージェント環境変数について詳しくは、790 ページの『データ・プロバイダーの環境変数』を参照してください。

データ・プロバイダーの環境変数

Sterling File Gateway エージェントを構成した後、エージェントのデータ収集に関連したいくつかのしきい値期間の値を変更できます。これらの値は、エージェント環境ファイルで指定できます。

以下の表に、データ・プロバイダーの環境変数の詳細な説明を示します。

表 *214.* データ・プロバイダーの環境変数の名前と説明

パラメーター名	**説明**
ファイル転送の収集期間 (時間単位) (**KFG_FILE_ARRIVED_INTERVAL**)	エージェントがファイル転送のデータを収集する期間 (時間単位)。デフォルト値は 24 時間です。
折れ線グラフとして表示されるファイル転送アクティビティーの 収集間隔 (時間単位) (**KFG_FILE_ACTIVITY_INTERVAL**)	エージェントがファイル転送アクティビティーのデータを 収集する期間 (時間単位)。デフォルト値は 1 時間です。 例えば、エージェントは過去 1 時間に発生したファイル転送アクティビティーを収集します。このデータは、インスタンス・ページに折れ線グラフとして表示可能です。デフォルト値は 1 時間です。
非アクティブ・パートナーのしきい値間隔 (日単位) (**KFG_INACTIVE_PARTNERS_INTERVAL**)	パートナーが非アクティブである、つまり、ファイルの受信もアップロードも行っていないしきい値期間。デフォルト値は 10 日です。 例えば、過去 10 日間にファイルの受信も転送も行っていないパートナーは、エージェントで「非アクティブ」として表示されます。
データ・プロバイダー・ログ・ファイルの最大数 (**KFG_LOG_FILE_MAX_COUNT**)	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。
各データ・プロバイダー・ログの最大サイズ (KB) (**KFG_LOG_FILE_MAX_SIZE**)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。

表 214. データ・プロバイダーの環境変数の名前と説明 (続き)	
パラメーター名	**説明**
データ・プロバイダー・ログの詳細レベル (**KFG_LOG_LEVEL**)	データ・プロバイダーによって作成されるログ・ファイルに含まれる詳細のレベル。デフォルト値は 4 (Info) です。有効な値は以下のとおりです。 • 1 (オフ): メッセージは記録されません。 • 2 (重大): エラーのみが記録されます。 • 3 (Warning (警告)): 重大レベルで記録されるすべてのエラーおよびメッセージと、望ましくない動作の原因となる可能性がある潜在的なエラー。 • 4 (Info (情報)): 警告レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した概要的な情報メッセージ。 • 5 (Fine (詳細レベル (低))): 情報レベルで記録されるすべてのエラーおよびメッセージと、データ・プロバイダーの実行時の状態を説明した詳しい情報メッセージ。 • 6 (Finer (詳細レベル (中))): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、パフォーマンスのプロファイル情報およびデバッグ・データなどの詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 7 (Finest (詳細レベル (高))): Fine (詳細レベル (低)) で記録されるすべてのエラーおよびメッセージと、詳細なプログラミング・メッセージおよびデータを含む、非常に詳細な情報メッセージ。このオプションを選択すると、モニター・エージェントのパフォーマンスに悪影響が及ぼされる可能性があります。この設定は、IBM サポート・スタッフと連携した問題判別用ツールとしての使用のみを意図しています。 • 8 (すべて): すべてのエラーおよびメッセージが記録されます。
すべてのファイル転送のイベントのフェッチ (**KFG_ALL_FGEVENTS**)	すべてのファイル転送のイベントをフェッチするフラグ。有効な値は Yes または No です。デフォルト値は No です。この値を No に設定すると、エージェントは、ユーザーが構成可能な期間について、失敗したファイル転送のイベントをフェッチします。この値を Yes に設定すると、エージェントは、ユーザーが構成可能な期間について、すべてのファイル転送のイベントをフェッチします。

B2B API 詳細の構成パラメーター

Sterling File Gateway エージェントを構成する際は、企業間 (B2B) API 詳細の構成パラメーターの値を指定する必要があります。

以下の表に、B2B API 詳細の構成パラメーターの詳細な説明を示します。

表 215. *B2B API* 詳細の構成パラメーターの名前と説明	
パラメーター名	**説明**
インスタンス名 (**KFG_Instance_Name**)	The name of the instance. **制約事項 :** 「インスタンス名」フィールドでは、初めてエージェントを構成したときに指定したインスタンスの名前が表示されます。エージェントを再構成する場合に、エージェントのインスタンス名を変更することはできません。

表 215. B2B API 詳細の構成パラメーターの名前と説明 (続き)	
パラメーター名	**説明**
サーバー名 (**KFG_API_SERVICES_Node_ADDRESS**)	B2B API サービスのホスト名または IP アドレス。
サーバー・ポート (**KFG_API_SERVICES_PORT**)	B2B API のポート。
ユーザー名 (**KFG_API_SERVICES_USERNAME**)	B2B API サービスに接続するためのユーザー名。
パスワード (**KFG_API_SERVICES_PASSWORD**)	B2B API サービスへの接続に使用するユーザー名のパスワード。

データベース詳細の構成パラメーター

Sterling File Gateway エージェントを構成する際は、データベース詳細の構成パラメーターの値を指定する必要があります。

以下の表に、データベース詳細の構成パラメーターの詳細な説明を示します。

表 216. データベース詳細の構成パラメーターの名前と説明	
パラメーター名	**説明**
データベース・サーバー名 (**KFG_DB_Node_ADDRESS**)	Sterling File Gateway データベース・サーバーのホスト名または IP アドレス。
データベース・ユーザー (**KFG_DB_USERNAME**)	データベース・ユーザーの名前。
データベース・パスワード (**KFG_DB_PASSWORD**)	データベースのパスワード。
データベース・ポート (**KFG_DB_PORT**)	データベースのポート。
データベース・タイプ (**KFG_DB_TYPE**)	データベースのタイプ。

Java API の構成パラメーター

Sterling File Gateway エージェントを構成する際は、Java API の構成パラメーターの値を指定する必要があります。

以下の表に、Java API の構成パラメーターの詳細な説明を示します。

表 217. Java API の構成パラメーターの名前と説明	
パラメーター名	**説明**
外部 JAR のクラスパス (**KFG_CLASSPATH**)	対応するデータベースに対して指定するデータベース・ドライバーの JAR パス。

Sybase サーバーのモニターの構成

Sybase エージェント は、分散データベースの一元的な管理を可能にします。データベース管理者およびシステム管理者が Sybase サーバー・システムのパフォーマンスを 調べるために必要な情報を収集し、問題を早期に検出し、問題を予防します。データベース管理者およびシステム管理者は、必要に応じてしきい値のレベルとフラグを設定し、システムがこれらのしきい値に到達したときにアラートを起動できます。Sybase サーバーをモニターするには、Monitoring Agent for Sybase Server を構成する必要があります。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Sybase エージェント を参照してください。

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

Sybase エージェントは複数インスタンス・エージェントであり、エージェントの各インスタンスを手動で構成し開始する必要があります。

手順

1. モニター・エージェントを構成します。
   - 795 ページの『コマンド行インターフェースを 使用したエージェントの構成』
   - 796 ページの『サイレント応答ファイルを使用したエージェントの構成』
2. モニター・エージェントの開始および停止にはエージェント・コマンド **sybase-agent** を使用します。

   sybase-agent について詳しくは、https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm の『エージェント・コマンドの使用』を参照してください。
3. コマンド **agent2server** を使用してモニター・エージェントを Performance Management サーバーに接続します。

   agent2server について詳しくは、https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm の『エージェント・コマンドの使用』を参照してください。

権限の付与

Sybase サーバーのモニターに使用するユーザー ID に権限を付与する必要があります。

始める前に

Sybase エージェント をインストールします。

権限を付与するには、データベース管理者の役割が必要です。

このタスクについて

モニター・エージェントが使用するユーザー ID には、Sybase 表およびインストールされているモニター表に対するアクセス権限が必要です。

以下のタスクを実行できます。

- モニター・エージェント用のユーザー ID を作成する。
- 新しいユーザー ID およびインストールされているモニター表に対する権限を付与する。

Sybase エージェントを `root` ユーザーとして実行しない場合は、ユーザー ID が Sybase グループに属し、Sybase ログ・ファイルに対する読み取り権限を持っていることを確認してください。

手順

1. 使用しているオペレーティング・システムに応じたコマンドを入力します。
   - Windows

     ```
     cd install_dir¥tmaitm6¥SQLLIB
     ```

- UNIX

```
cd install_dir/misc
```

ここで、*install_dir* は Sybase サーバーのインストール先ホーム・ディレクトリーです。

2. **isql** コマンドを使用して、ユーザー sa として Sybase サーバーにログインします。
3. 以下のコマンドを実行して、Sybase エージェントが Sybase サーバーとの通信に使用する ID を構成します。

```
1>sp_addlogin user_name, password 2>g
```

各項目の意味は次のとおりです。

- *user_name* はユーザー ID です。デフォルトでは tivoli です。

  ユーザー ID が tivoli でない場合は、koygrant.sql ファイルを編集して、tivoli を正しいユーザー ID に変更してください。
- *password* はユーザーのパスワードです。

注:

koygrant.sql ファイルの場所:

- Windows ¥opt¥ibm¥apm¥agent¥misc¥
- UNIX /opt/ibm/apm/agent/misc/

4. 以下のコマンドを実行して、データベースの表に対する権限を付与します。

```
isql -U sa -P password -S servername -i koygrant_filepathkoygrant.sql
```

各項目の意味は次のとおりです。

- *password* はユーザー sa のパスワードです。
- *servername* はデータベース・サーバー名です。
- *koygrant_filepath* は以下の場所です。

  注:

  – Windows ¥opt¥ibm¥apm¥agent¥misc¥

  – UNIX /opt/ibm/apm/agent/misc/

5. 以下のコマンドを実行して、インストールされているモニター表に使用するプロキシー・テーブルを作成します。

```
isql -U sa -P password -S servername
     -i $SYBASE/ASE-12_5/scripts/installmontables
```

各項目の意味は次のとおりです。

- *password* はユーザー sa のパスワードです。
- *servername* はデータベース・サーバー名です。

次のタスク

権限が正常に付与されたら、モニター・エージェントを構成できます。

コマンド行インターフェースを使用したエージェントの構成

コマンド行インターフェースを 使用して Monitoring Agent for Sybase Server を構成できます。

始める前に

Sybase エージェントはリモート構成をサポートしていません。そのため、Sybase サーバーは、Sybase エージェントのインストール先と同じホストにインストールする必要があります。

Sybase エージェントは、Sybase サーバー バージョン 15.7 および 16.0 のみをサポートしています。

データベース・サーバーへの接続に使用するユーザー ID を作成します。

このタスクについて

Sybase エージェントは複数インスタンス・エージェントであり、エージェントの各インスタンスを手動で構成し開始する必要があります。

手順

1. 以下のコマンドを実行して、エージェントを構成します。
   - Windows

     ```
     install_dir¥bin¥sybase-agent.bat instance_name
     ```

   - UNIX

     ```
     install_dir/bin/sybase-agent.sh instance_name
     ```

   各部の意味は次のとおりです。
   - *install_dir* は、エージェントのインストール・ディレクトリーです。
   - *instance_name* は Sybase サーバーのインスタンス名です。
2. 以下のパラメーターの値を指定するプロンプトが出た場合は、Enter キーを押してデフォルト値を受け入れるか、値を指定してから Enter キーを押します。
   a) `Home Directory` パラメーターには、Sybase サーバーのホーム・ディレクトリーのパスを入力します。
      - Windows

        例えば `Home Directory` を `¥opt¥sybase` とします。
      - UNIX

        例えば `Home Directory` を `/opt/sybase` とします。

   b) `ASE Directory` パラメーターには、データベース・サーバー ASE のパスを入力します。
      - Windows

        例えば `ASE Directory` を `¥opt¥sybase¥ASE-12_5` とします。
      - UNIX

        例えば `ASE Directory` を `/opt/sybase/ASE-12_5` とします。

   c) `Open Client Directory` パラメーターには、Sybase オープン・クライアントのインストール場所を入力します。
      - Windows

        例えば `Open Client Directory` を `¥opt¥sap¥ocs-16_0` とします。
      - UNIX

        例えば `Open Client Directory` を `/opt/sap/ocs-16_0` とします。

d) USER ID パラメーターには、モニター・エージェントが Sybase サーバーへの接続に使用するユーザー ID を入力します。

デフォルトの USER ID は tivoli です。

e) PASSWORD パラメーターには、モニター・エージェントが Sybase サーバーへの接続に使用するユーザー ID のパスワードを入力します。

f) VERSION パラメーターには、Sybase サーバーのバージョンを入力します。

Sybase エージェントは、Sybase サーバー バージョン 15.7 および 16.0 のみをサポートしています。

g) ERROR LOG FILE パラメーターには、Sybase サーバーのエラー・ログ・ファイルの完全修飾ファイル名を入力します。

- Windows

  例えば ERROR LOG FILE を ¥opt¥sap¥ASE-16_0¥install¥*servername*.log とします。

- UNIX

  例えば ERROR LOG FILE を /opt/sap/ASE-16_0/install/*servername*.log とします。

ここで、*servername* は Sybase サーバー名です。

h) EXTENDED パラメーターには、特定のカーソル実行を除外するためにサポートが使用する拡張パラメーターを入力します。値を指定せずに Enter キーを押すと、すべてのカーソルが実行されます。

EXTENDED パラメーターのオプションは DBD2、DBD15、KOYSEGD です。

- DBD2 は、データ・セット Sybase_Database_Detail および Sybase_Database_Summary に対するカーソル実行を除外します。
- DBD15 は、データ・セット Sybase_Database_Detail に対するカーソル実行を除外します。
- KOYSEGD は、データ・セット Sybase_Segment_Detail に対するカーソル実行を除外します。

次のタスク

構成が完了したら、モニター・エージェントを開始して、モニター・エージェントを Performance Management サーバーに接続できます。

Sybase エージェントを開始するには、エージェント・コマンド sybase-agent を使用します。

Sybase エージェントを Performance Management サーバーに接続するには、agent2server コマンドを使用します。

sybase-agent および agent2server について詳しくは、https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm の『エージェント・コマンドの使用』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルを使用して Monitoring Agent for Sybase Server を構成できます。

始める前に

Sybase エージェントはリモート構成をサポートしていません。そのため、Sybase サーバーは、Sybase エージェントのインストール先と同じホストにインストールする必要があります。

Sybase エージェントは、Sybase サーバー バージョン 15.7 および 16.0 のみをサポートしています。

データベース・サーバーへの接続に使用するユーザー ID を作成します。

このタスクについて

Sybase エージェントは複数インスタンス・エージェントであり、エージェントの各インスタンスを手動で構成し開始する必要があります。

サイレント応答ファイルを編集し、エージェント・コマンドを実行してモニター・エージェントを構成する必要があります。

手順

1. サイレント応答ファイルを編集します。
   - Windows

     サイレント応答ファイルは *install_dir*¥samples¥sybase_silent_config.txt です。
   - UNIX

     サイレント応答ファイルは *install_dir*/samples/sybase_silent_config.txt です。

   ここで、*install_dir* はエージェントのインストール・ディレクトリーです。

   a) Home Directory パラメーターには、Sybase サーバーのホーム・ディレクトリーのパスを指定します。
      - Windows

        例えば Home Directory を ¥opt¥sybase とします。
      - UNIX

        例えば Home Directory を /opt/sybase とします。

   b) ASE Directory パラメーターには、データベース・サーバー ASE のパスを指定します。
      - Windows

        例えば ASE Directory を ¥opt¥sybase¥ASE-12_5 とします。
      - UNIX

        例えば ASE Directory を /opt/sybase/ASE-12_5 とします。

   c) Open Client Directory パラメーターには、Sybase オープン・クライアントのインストール場所を指定します。
      - Windows

        例えば Open Client Directory を ¥opt¥sap¥ocs-16_0 とします。
      - UNIX

        例えば Open Client Directory を /opt/sap/ocs-16_0 とします。

   d) USER ID パラメーターには、モニター・エージェントが Sybase サーバーへの接続に使用するユーザー ID を指定します。

      デフォルトの USER ID は tivoli です。

   e) PASSWORD パラメーターには、モニター・エージェントが Sybase サーバーへの接続に使用するユーザー ID のパスワードを指定します。

   f) VERSION パラメーターには、Sybase サーバーのバージョンを指定します。

      Sybase エージェントは、Sybase サーバー バージョン 15.7 および 16.0 のみをサポートしています。

   g) ERROR LOG FILE パラメーターには、Sybase サーバーのエラー・ログ・ファイルの完全修飾ファイル名を指定します。
      - Windows

        例えば ERROR LOG FILE を ¥opt¥sap¥ASE-16_0¥install¥*servername*.log とします。
      - UNIX

        例えば ERROR LOG FILE を /opt/sap/ASE-16_0/install/*servername*.log とします。

      ここで、*servername* は Sybase サーバー名です。

h) EXTENDED パラメーターには、特定のカーソル実行を除外するためにサポートが使用する拡張パラメーターを指定します。ブランクにすると、すべてのカーソルが実行されます。

EXTENDED のオプションは DBD2、DBD15、KOYSEGD です。

- DBD2 は、データ・セット Sybase_Database_Detail および Sybase_Database_Summary に対するカーソル実行を除外します。
- DBD15 は、データ・セット Sybase_Database_Detail に対するカーソル実行を除外します。
- KOYSEGD は、データ・セット Sybase_Segment_Detail に対するカーソル実行を除外します。

2. サイレント応答ファイルを保存します。
3. 以下のエージェント・コマンドを実行してモニター・エージェントを構成します。
   - Windows

     ```
     install_dir¥bin¥sybase-agent.bat config instance_name
      install_dir¥samples¥sybase_silent_config.txt
     ```

   - UNIX

     ```
     install_dir/bin/sybase-agent.sh config instance_name
      install_dir/samples/sybase_silent_config.txt
     ```

   各部の意味は次のとおりです。

   - *install_dir* は、エージェントのインストール・ディレクトリーです。
   - *instance_name* は Sybase サーバーの名前です。

次のタスク

構成が完了したら、モニター・エージェントを開始して、モニター・エージェントを Performance Management サーバーに接続できます。

Sybase エージェントを開始するには、**sybase-agent** コマンドを使用します。

Sybase エージェントを Performance Management サーバーに接続するには、**agent2server** コマンドを使用します。

sybase-agent および **agent2server** について詳しくは、https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm の『エージェント・コマンドの使用』を参照してください。

照会のダーティー読み取り (dirty reads) の無効化

Sybase エージェントは、ロックを防止するために、照会実行に対する dirty reads をデフォルトで有効にします。
照会の dirty reads を有効または無効にするには、COLL_USE_NOLOCK 変数を使用します。
dirty reads を有効にすると、ロックを回避するために、照会が分離レベル・ゼロで実行されます。
エージェント照会の dirty reads を無効にする場合には、COLL_USE_NOLOCK 変数をゼロに設定できます。

始める前に

エージェント照会の dirty reads を無効にするには、エージェントがインストール済みであることを確認します。

このタスクについて

Sybase エージェントは、デフォルトで dirty reads を有効にします。エージェント照会の dirty reads を無効にするには、以下のステップを実行してください。

手順

1. エージェントを停止します。
2. COLL_USE_NOLOCK 変数をゼロに設定します。
   - UNIX
     a. *CANDLEHOME*/config/.oy.environment ファイルに **COLL_USE_NOLOCK=0** を追加します。
     b. ファイルを保存して閉じます。
   - Windows
     a. エージェント・インスタンス・ファイル *CANDLEHOME*¥TMAITM6_x64¥KOYENV_*INSTANCENAME* を見つけます。
     b. ファイルに次の行を追加します。

        COLL_USE_NOLOCK=0

     c. ファイルを保存して閉じます。

   CANDLEHOME はエージェントのインストール・ディレクトリーです。
   INSTANCENAME はエージェント・インスタンス名です。
3. エージェントを開始します。

Synthetic Playback モニターの構成

エージェントが内部 Web アプリケーションの可用性およびパフォーマンスに関するデータを収集できるように、Synthetic Playback エージェントを構成する必要があります。このデータは Application Performance ダッシュボード に表示されます。

このタスクについて

Synthetic Playback エージェントを構成するには、スクリプトを実行し、プロンプトに応答します。次に、スクリプトを開始し、スクリプトが実行されていることを確認します。

重要 : IBM Website Monitoring on Cloud アドオンの既存のユーザーのみが Synthetic Playback エージェントをインストール、構成、および実行できます。Website Monitoring は、IBM Cloud Availability Monitoring によって置き換えられました。詳しくは、1047 ページの『Availability Monitoring について』を参照してください。

手順

- スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。
  a) *install_dir*/bin/synthetic_playback-agent.sh config と入力します。ここで、*install_dir* は、Synthetic Playback エージェントのインストール・ディレクトリーです。
  b)「Monitoring Agent for Synthetic Playback 設定の編集 (Edit Monitoring Agent for Synthetic Playback settings)」というプロンプトが出されたら、1 を入力して続行します。
  c) 再生 POP (point of presence) のデータ・センター名の入力を求めるプロンプトが出されたら、エージェントのロケーションを識別する名前を入力します。

  重要 : 再生 POP の記述名を選択します。エージェントのインストールが完了したら、シンセティック・トランザクションの再生ロケーションとして、そのロケーションを名前で選択でき、Application Performance ダッシュボードにそのロケーションのトランザクション・データを表示できます。
  d)「Java パラメーター」というプロンプトが出されたら、「Java トレース・レベル」を選択します。Enter を押してデフォルト・パラメーターを選択するか、1 から 8 の範囲内の数値を入力してトレース・レベルを指定します。

e)「外部 JAR のクラスパス」というプロンプトが出されたら、ブランクのままにするか、あるいは外部 jar のロケーションを指定します。

- サイレント応答ファイルを使用してエージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで、*install_dir*/samples のパスにある synthetic_playback_silent_config.txt ファイルを開きます。
  例:

  Linux /opt/ibm/apm/agent/samples

  b) synthetic_playback_silent_config.txt ファイルで、以下のプロパティーのコメントを外し、各プロパティーに値を割り当てます。
  - LOCATION について、このパラメーターの値に、データ・センターの名前、またはエージェントのインストール先を表す名前を設定します。
  - JAVA_TRACE_LEVEL について、このパラメーターの値に、リストされたトレース・レベルのうちの 1 つを設定します (例えば、JAVA_TRACE_LEVEL=ERROR など)。

  このファイルを保存します。
  c) コマンド行で、パスを *install_dir*/bin に変更します。
  d) 次のコマンドを実行して、エージェントをサイレント ・モードで構成します。

```
synthetic_playback-agent.sh config install_dir/samples/
synthetic_playback_silent_config.txt
```

- Synthetic Playback エージェント を開始するには、 *install_dir*/bin/synthetic_playback-agent.sh start と入力します。
- Synthetic Playback エージェントが実行されていることを確認するには、*install_dir*/bin/synthetic_playback-agent.sh status と入力します。詳しくは、183 ページの表 12 を参照してください。

次のタスク
内部 Web アプリケーションのパフォーマンスを 表示するには、Synthetic Script Manager でシンセティック・トランザクションを作成する必要があります。詳しくは、1028 ページの『Website Monitoring でのシンセティック・トランザクションおよびシンセティック・イベントの管理』 を参照してください。

Synthetic Playback エージェントに対するアップストリーム・プロキシー・サポートの有効化

Synthetic Playback エージェントに対するアップストリーム・プロキシー・サポートを有効にして、内部 Web アプリケーションから外部 Web アプリケーションに対する HTTP 要求をモニターします。

始める前に
Synthetic Playback エージェントのバージョン 01.00.05.08 以降が実行されていることを確認します。 実行中のエージェントのバージョンを確認するには、コマンド行に *install_dir*/bin/cinfo -t と入力します。ここで、*install_dir* はエージェントのインストール・ロケーションです。他のバージョンの Synthetic Playback エージェントを実行している場合、IBM Fix Central から IBM Cloud Application Performance Management, Private 8.1.4.0 の Synthetic Playback エージェント暫定修正 08 をダウンロードしてインストールする必要があります (「**検索**」フィールドに Synthetic と入力すると、Synthetic Playback エージェント暫定修正のリストが表示されます)。 インストール手順については、8.1.4.0-IBM-IPM-SYNTHETIC-PLAYBACK-AGENT-IF0008 Readme を参照してください。

このタスクについて

企業のファイアウォール背後の内部 Web アプリケーションは、外部の Web リソースにアクセスするためにアップストリーム・プロキシーを必要とします。内部 Web アプリケーションから外部 Web アプリケーションに対する HTTP 要求をモニターできるようにするため、エージェントによるアップストリーム・プロキシーのサポートを許可するように Synthetic Playback エージェントのプロキシー設定を構成します。

手順

- エージェントに対するアップストリーム・プロキシー・サポートを構成して有効にするには、次のステップを実行します。

  a) root ユーザーとして、コマンド行で次のコマンドを実行してプロキシー設定を構成します。

  ```
  cd install_dir/agent/lx8266/sn/bin
  #./set_proxy.sh
  ```

  プロンプトが出されたら、エージェントのインストール・パス (デフォルト・パスは /opt/ibm/apm/agent) を入力します。Synthetic Playback エージェント用に構成するプロキシー・タイプの番号を入力します。

  例:

  ```
  # cd /install_dir/agent/lx8266/sn/bin/
  #./set_proxy.sh
  please input the agent install path, default is (/opt/ibm/apm/agent)
  agent install path is:/opt/ibm/apm/agent
  please input the number of proxy type:
  1 system proxy
  2 manual proxy
  3 pac proxy
  4 no proxy
  ```

  b) *install_dir*/bin/synthetic_playback-agent.sh start と入力して、エージェントを再開します。

- Linux エージェントに対するアップストリーム・プロキシー・サポートを無効にするには、再度 ./set_proxy.sh コマンドを実行して、4 no proxy を選択します。次に、エージェントを再始動します。

Tomcat モニターの構成

Tomcat アプリケーション・サーバーのリソースをモニターするために、デフォルトまたはカスタム設定を使用して Monitoring Agent for Tomcat を構成できます。エージェントは、Windows および Linux システムで構成できます。

始める前に

ハードウェアおよびソフトウェアの前提条件を確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the Tomcat エージェント を参照してください。

このタスクについて

Tomcat エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。管理対象システム名には、指定するインスタンス名が含まれます (例: *instance_name*:*host_name*:*pc*)。ここで *pc* は、ご使用の製品の 2 文字の製品コードです。管理対象システム名は 32 文字に制限されています。指定するインスタンス名は 28 文字からホスト名の字数を除いた長さに制限されています。例えば、インスタンス名に TOMCAT2 を指定すると、管理対象システム名は TOMCAT2:hostname:OT になります。長いインスタンス名を指定した場合、管理対象システム名は切り捨てられ、エージェント・コードが完全には表示されません。

エージェントの構成時に権限の問題が発生しないようにするために、必ず、エージェントのインストールに使用したのと同じ root ユーザー ID または非 root ユーザー ID を使用してください。選択されたユーザーとしてエージェントをインストールした後で、別のユーザーとしてエージェントを構成する場合は、187 ページの『非 root ユーザーとしてのエージェントの構成』を参照してください。選択されたユーザーとしてエージェントをインストールおよび構成した後で、別のユーザーとしてエージェントを開始する場合は、1013 ページの『非 root ユーザーとしてのエージェントの開始』を参照してください。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェ

ント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

デフォルト設定を使用した Tomcat エージェントの構成

Tomcat エージェントのデフォルト設定を使用して、Tomcat サーバーをモニターできます。新規インスタンス名以外の構成情報を指定する必要はありません。

始める前に

デフォルト設定を使用してエージェントを構成する前に、以下の前提条件が満たされていることを確認します。

- エージェントがデフォルトのディレクトリーにインストールされている 。
- JMX サービス URL がポート 8686 を使用している。
- Tomcat サーバーが JMX 許可なしで構成されている。

このタスクについて

要確認 : デフォルト設定を使用してエージェントを構成すると、トランザクション・トラッキングと詳細診断のデータの収集は有効になりません。

手順

1. 次のコマンドを実行します。
   Linux *install_dir*/bin/tomcat-agent.sh config *instance_name install_dir*/samples/tomcat_silent_config.txt
   Windows *install_dir*/bin/tomcat-agent.bat config *instance_name install_dir*/samples/tomcat_silent_config.txt
   表の見方
   install_dir
   Tomcat エージェント のインストール・ディレクトリー。
   instance_name
   インスタンスに指定する名前です。
2. 以下のコマンドを実行して、エージェントを開始します。
   Linux *install_dir*/bin/tomcat-agent.sh start *instance_name*
   Windows *install_dir*/bin/tomcat-agent.bat start *instance_name*

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。

始める前に

以下の前提条件が満たされていることを確認してください。

- エージェントがインストールされている Tomcat サーバーに Java がインストールされている。
- エージェント・インストーラーのインストール元のプロンプトで JDK 1.6 以降のバージョンが設定されている。
- Tomcat サーバーに対して JMX リモートが有効になっている。詳しくは、「Enabling JMX Remote」を参照してください。
- Tomcat サーバーが稼働中である。

このタスクについて

コマンド・プロンプトからエージェントを構成できます。詳しくは、806 ページの『Linux システムでの Tomcat エージェントの構成』トピックで説明されているステップに従って、.sh 拡張子の代わりに .bat 拡張子を指定してコマンドを実行してください。以下の手順では、エージェント構成パネルを使用したエージェントの構成について説明します。

手順

1. 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for Tomcat**」を右クリックします。
3. 「**エージェントを構成します**」をクリックします。

   重要 : 「**エージェントを構成します**」が使用不可になっている場合は、「**再構成**」をクリックします。
4. 「**インスタンス名**」ウィンドウで、Tomcat エージェント・インスタンスの固有の名前を指定し、「**OK**」をクリックします。

   制約事項 : MSN は 32 文字を超えないようにしてください。
5. 「**サーバー名**」フィールドに、モニターされている Tomcat サーバーを識別する固有の名前を入力します。
6. 「**Java パラメーター設定 (Java Parameter Settings)**」ウィンドウで、以下のステップのいずれかを実行します。
   - 「**次へ**」をクリックして、Java がインストールされているデフォルトの場所を受け入れます。デフォルトのインストール・パスは、`C:¥IBM¥APM¥java¥java80_x64¥jre` です。
   - IBM Java が別のパスにインストールされている場合は、「**Java ホーム**」フィールドにパスを指定します。
7. 「**JSR-160 準拠のサーバー**」ウィンドウに、以下のパラメーターの詳細を指定します。
   a) 「**JMX ユーザー ID**」フィールドに、JMX 許可が Tomcat で有効になっている場合に Tomcat MBean サーバーへの接続に使用するユーザーの ID を指定します。
   b) 「**JMX パスワード**」フィールドに、JMX 許可が Tomcat で有効になっている場合に使用する JMX ユーザーのパスワードを指定します。
   c) 「**JMX サービス URL**」フィールドに、Tomcat MBean サーバーへの接続に使用する URL を入力します。

   URL の形式は `service:jmx:rmi:///jndi/rmi://`*`host_name`*`:`*`port_number`*`/jmxrmi` です。サーバーがローカル・ホスト上で実行され、JMX ポートとしてポート 8686 を使用している場合は、デフォルトの URL が有効です。同じ形式を維持しながら URL 内のホスト名とポート番号を変更することができます。
   d) トランザクション・トラッキングと詳細診断のデータの収集を有効にする場合、「**データ・コレクターの構成**」リストから、「はい」を選択します。
8. 「**Monitoring Agent for Tomcat**」ウィンドウで Tomcat エージェント・インスタンスを右クリックし、「**開始**」をクリックします。
9. トランザクション・トラッキングと詳細のデータの収集を有効にして、Tomcat サーバーを再始動します。

次のタスク

Tomcat エージェントがサービスとして実行されている場合は、Windows でエージェントを構成した後で Tomcat データ・コレクターを構成します。詳しくは、805 ページの『Tomcat データ・コレクターの構成』を参照してください。

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Windows 上の Monitoring Agent for Tomcat を非管理者ユーザーによって実行するためのローカル・セキュリティー・ポリシーの指定

ローカル・セキュリティー・ポリシーを使用して、Windows 上の Monitoring Agent for Tomcat を非管理者ユーザーによって実行できるようにすることができます。

このタスクについて

以下の 2 つのローカル・セキュリティー・ポリシーを組み合わせることで、Windows 上の Tomcat エージェントを非管理者ユーザーによって実行できるようにすることができます。Tomcat エージェントの開始と停止、構成、およびデータ検証を行うには、以下の 2 つのポリシーが必要です。

1. プログラムのデバッグ
2. サービスとしてログオン (Log on as Service)

これらのローカル・セキュリティー許可を非管理者ユーザーに付与するには、以下の手順を実行します。

手順

1. TEMA にアクセスして、Tomcat エージェントの始動を非管理者ユーザーを指定して変更します。
2. Tomcat エージェントのインストール・フォルダーに非管理者ユーザーを追加して、すべての権限を付与します。
3. レジストリー・キー HKEY_LOCAL_MACHINE に非管理者ユーザーを追加して、「**全アクセス許可 (Full Permissions)**」をクリックします。
4. 「**スタート**」**メニュー**で **secpol.msc** コマンドを実行して、「ローカル セキュリティ ポリシー」を開きます。
5. 次に、804 ページの『ローカル・セキュリティー・ポリシーの許可』を参照して非管理者ユーザーをポリシーに追加します。
6. Tomcat エージェントを再始動します。
7. APM ポータルで Tomcat エージェントの状況を確認し、データを検証します。

ローカル・セキュリティー・ポリシーの許可

「プログラムのデバッグ」許可の付与

このタスクについて

「プログラムのデバッグ」許可を付与するには、Tomcat エージェントで以下の手順を実行します。

手順

1. 「**スタート**」**>**「**管理ツール**」**>**「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル・ポリシー**」を展開し、「**ユーザー権限の割り当て**」をクリックします。ユーザー権限のリストが開きます。
3. 「**プログラムのデバッグ**」ポリシーをダブルクリックします。「**プログラムのデバッグのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「選択するオブジェクト名を入力してください」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**OK**」をクリックします。

「サービスとしてログオン *(Log on as Service)*」許可の付与

このタスクについて

「サービスとしてログオン (Log-on as Service)」許可を付与するには、Tomcat エージェントで以下の手順を実行します。

手順

1. 「**スタート**」>「**管理ツール**」>「**ローカル セキュリティ ポリシー**」をクリックします。「**ローカル セキュリティ設定**」ウィンドウが開きます。
2. 「**ローカル・ポリシー**」を展開し、「**ユーザー権限の割り当て**」をクリックします。ユーザー権限のリストが開きます。
3. 「**サービスとしてログオン (Log-on as service)**」ポリシーをダブルクリックします。「**サービスとしてログオンのプロパティ**」ウィンドウが開きます。
4. 「**ユーザーまたはグループの追加**」をクリックします。「**ユーザーの選択**」または「**グループの選択**」ウィンドウが開きます。
5. 「選択するオブジェクト名を入力してください」フィールドに、権限を割り当てるユーザー・アカウント名を入力して、「**OK**」をクリックします。
6. 「**OK**」をクリックします。

Tomcat データ・コレクターの構成

Tomcat エージェントがサービスとして実行されている場合は、Windows でエージェントを構成した後で、以下に示されている手順で Tomcat データ・コレクターを構成します。

このタスクについて

Tomcat エージェント・インスタンスを構成して開始すると、このインスタンスによって /CANDLEHOME/setenv_<instance_name>.bat で setenv.bat ファイルが生成または更新されます。このファイルには、Tomcat データ・コレクターの構成に必要なデータ・コレクター構成パラメーターが含まれています。

手順

1. 「**Apache Tomcat Properties**」ウィンドウを開き、「**Java**」をクリックします。
2. ロケーション /CANDLEHOME/setenv_<instance_name>.bat から setenv_<instanceName>.bat を開きます。
3. 以下のブロックに示す setenv_<instance_name>.bat から **JAVA_OPTS** パラメーターの値をコピーします。

```
agentlib:am_ibm_16=C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥runtime¥TOMTKWIN1
-Xbootclasspath/p:C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥toolkit¥lib¥bcm-bootstrap.jar
-Djava.security.policy=C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥itcamdc¥etc¥datacollector.policy
-Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=
C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥runtime¥TOMTKWIN1¥TOM_TK_1_DCManual.txt
-Dcom.ibm.tivoli.itcam.serverHome=C:¥TOMCAT_9¥apache-tomcat-9.0.5¥apache-tomcat-9.0.5
-Dam.home=C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥itcamdc
-Dcom.ibm.tivoli.itcam.toolkit.runtime.dir=C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥runtime
```

4. 「**Apache Tomcat Properties**」の「**Java**」タブにある「**Java Options**」というラベルの付いたテキスト・ボックスに値を貼り付けます。
5. 「**適用**」をクリックします。
6. 「**コントロール パネル**」に進み、「**システム**」 > 「**拡張**」 > 「**環境変数**」をクリックします。
7. 「**システム変数**」で、ファイル・パス <OTDC_home>¥toolkit¥lib¥win64;<OTDC_HOME>/ toolkit¥lib¥win64¥ttapi を追加して *PATH* 変数を編集し、「**OK**」をクリックします。

   注 : <OTDC_home> をツールキットのインストール・ディレクトリーの実パスに置き換えます。例えば、 C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥toolkit¥lib¥win64;C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥toolkit¥lib¥win64¥ttapi のようになります。
8. 「**新規**」をクリックして変数 *RUNTIME_DIR* を追加します。
9. 「**変数名**」に *RUNTIME_DIR*、「**変数パス**」に C:¥IBM¥APM¥otdchome¥7.3.0.13.0¥runtime を指定して追加します。このパスは、setenv_<instancename>.bat で確認できます。
10. Windows を再始動します。Tomcat サービスの起動が「自動」に設定されていることを確認してください。

Linux システムでの Tomcat エージェントの構成

Linux システムで、構成スクリプトを実行し、プロンプトに応答することで、Tomcat エージェントを構成します。

始める前に

- Tomcat サーバーに対して JMX リモートが有効になっている。詳しくは、「Enabling JMX Remote」を参照してください。
- Tomcat サーバーが稼働中である。

手順

1. 次のコマンドを実行します。
   install_dir/bin/tomcat-agent.sh config *instance_name*
   ここで、*instance_name* は、インスタンスに指定する名前です。
2. SERVER の値を指定するように求めるプロンプトが出されたら、モニターされている Tomcat サーバーを識別する固有の名前を指定し、Enter キーを押します。
3. Java home の値を指定するように求めるプロンプトが出されたら、Enter キーを押して Java 仮想マシンのインストール先のデフォルトの場所を受け入れます。デフォルトの場所は、/opt/ibm/apm/agent/JRE/lx8266/jre です。エージェントがデフォルトのディレクトリーにインストールされていない場合は、*install_dir*/JRE/lx8266/jre と指定します。
4. JMX user ID の値を指定するように求めるプロンプトが出されたら、Tomcat MBean サーバーに接続するユーザーの ID を指定します。 JMX 許可が有効になっていない場合は、Enter キーを押します。
5. JMX password の値を指定するように求めるプロンプトが出されたら、JMX ユーザーのパスワードを指定してから確認します。 JMX 許可が有効になっていない場合は、Enter キーを押します。
6. JMX service URL の値を指定するように求めるプロンプトが出されたら、Enter キーを押してデフォルト URL を受け入れるか、または Tomcat MBean サーバーに接続するための別のサービス URL を指定します。
   URL の形式は service:jmx:rmi:///jndi/rmi://*host_name*:*port_number*/jmxrmi です。 サーバーがローカル・ホスト上で実行され、JMX ポートとしてポート 8686 を使用している場合は、デフォルトの URL が有効です。同じ形式を維持しながら URL 内のホスト名とポートを変更することができます。
7. Data Collector Configuration の値を指定するように求めるプロンプトが出されたら、1 を指定し、Enter キーを押して、トランザクション・トラッキングと詳細診断のデータの収集を有効にします。
8. 以下のコマンドを実行して、エージェントを開始します。
   install_dir/bin/tomcat-agent.sh start *instance_name*
9. トランザクション・トラッキングと詳細診断のデータの収集を有効にして、Tomcat サーバーを再始動します。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

サイレント応答ファイルを使用した Tomcat エージェントの構成

サイレント応答ファイルを使用することで、プロンプトに応答することなく、Tomcat エージェントを構成できます。

手順

1. テキスト・エディターで、tomcat_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
   install_dir/samples

2. **KOT_SERVER** パラメーターには、モニターされている Tomcat サーバーを識別する固有の名前を指定します。
3. **Java home** パラメーターには、Java 仮想マシンのインストール先のパスを指定します。デフォルトの場所は、/opt/ibm/apm/agent/JRE/lx8266/jre です。エージェントがデフォルトのディレクトリーにインストールされていない場合は、*install_dir*/JRE/lx8266/jre と指定します。
4. **「JMX user ID」** パラメーターに、Tomcat MBean サーバーへの接続に使用するユーザーの ID を指定します。JMX 許可が Tomcat で有効になっている場合、このパラメーターに値を指定する必要があります。
5. **JMX password** パラメーターには、JMX ユーザーのパスワードを指定します。JMX 許可が Tomcat で有効になっている場合、このパラメーターに値を指定する必要があります。
6. **JMX service URL** パラメーターには、Tomcat MBean サーバーに接続するためのサービス URL を指定します。URL の形式は service:jmx:rmi:///jndi/rmi://*host_name*:*port_number*/jmxrmi です。 サーバーがローカル・ホスト上で実行され、JMX ポートとしてポート 8686 を使用している場合は、デフォルトの URL が有効です。同じ形式を維持しながら URL 内のホスト名とポート番号を変更することができます。
7. トランザクション・トラッキングと詳細診断のデータの収集を有効にする場合、**KOT_DCCONFIGURATION** パラメーターには、Yes を指定します。
8. tomcat_silent_config.txt ファイルを保存して閉じ、以下のコマンドを実行してエージェント構成設定を更新します。
   Linux *install_dir*/bin/tomcat-agent.sh config *instance_name install_dir*/samples/tomcat_silent_config.txt
   Windows *install_dir*\bin\tomcat-agent.bat config *instance_name install_dir*\samples\tomcat_silent_config.txt
   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は Tomcat エージェントのインストール・ディレクトリーです。
9. 以下のコマンドを実行して、エージェントを開始します。
   Linux *install_dir*/bin/tomcat-agent.sh start *instance_name*
   Windows *install_dir*\bin\tomcat-agent.bat start *instance_name*
10. トランザクション・トラッキングと詳細診断のデータの収集を有効にするには、Tomcat サーバーを再始動する必要があります。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

トランザクション・トラッキング・データおよび診断データの収集の有効化

「**エージェント構成**」ページで、トランザクション・トラッキング・データまたは診断データの収集を有効化または無効化できます。

このタスクについて

トランザクション・トラッキング・データの収集を有効にすると、エージェントは、以下のコンポーネントのデータを収集します。

- サーブレット JSP
- EJB アプリケーション
- JMS

手順

各管理対象システムでデータ収集を構成するには、以下のステップを実行します。

1. Cloud APM コンソール にログインします。

2. ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
3. 「**Tomcat**」タブをクリックします。
4. データ収集を構成する管理対象システムのチェック・ボックスを選択し、「**アクション**」リストから以下のいずれかのアクションを実行します。
   - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。選択した各管理対象システムの「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
   - 診断データの収集を有効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**診断モードのみの有効化 (Enabled Diagnostic Mode Only)**」を選択します。選択した各管理対象システムの「診断モード」列の状況が「有効」に更新されます。
   - 診断データの収集とメソッド・トレースを有効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**診断モードおよびメソッド・トレースの有効化**」を選択します。選択した各管理対象システムの「診断モード」列と「メソッド・トレース」列の状況が「有効」に更新されます。
   - トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。選択した各管理対象システムの「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。
   - 診断データの収集を無効にするには、「**診断モードの設定 (Set Diagnostic Mode)**」 > 「**診断モードおよびメソッド・トレースの無効化**」をクリックします。選択した各管理対象システムの「**診断モード**」および「**メソッド・トレース**」列の状況が「無効」に更新されます。

次のタスク

Cloud APM コンソールにログインして、エージェントにより収集されるトランザクション・トラッキングおよび診断データをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Tomcat アプリケーション・サーバーの更新または変更

Tomcat エージェントの構成後に Tomcat アプリケーション・サーバーを更新または変更するには、このトピックに記載されているステップに従います。これらのステップは、Windows と Linux で構成される両方の Tomcat に共通です。

手順

1. Tomcat エージェント・インスタンスおよび Tomcat サーバーを停止します。
2. <TOMCAT_SERVER>/bin に移動して、setenv.sh ファイルをエディターで開きます。
3. setenv.sh からデータ・コレクターのすべての開始パラメーターを削除します。 ファイルから以下の行を削除します。

```
export LD_LIBRARY_PATH="<CANDLE_HOME>/otdchome/7.3.0.13.0/toolkit/lib/lx8266"
export RUNTIME_DIR="<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime"
export JAVA_OPTS="-agentlib:am_ibm_16=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/
<Tomcat_Application_
Server> -
Xbootclasspath/p:<CANDLE_HOME>/otdchome/7.3.0.13.0/toolkit/lib/bcm-bootstrap.jar -
Djava.security.policy=<CANDLE_HOME>/otdchome/7.3.0.13.0/itcamdc/etc/datacollector.policy -
Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/
<Tomcat_
Application_Server>/<Agent_Instance>_DCManual.txt -
Dcom.ibm.tivoli.itcam.serverHome=<TOMCAT_HOME> -
Dam.home=<CANDLE_HOME>/otdchome/7.3.0.13.0/itcamdc -
Dcom.ibm.tivoli.itcam.toolkit.runtime.dir=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime"
```

4. 変更を保存して、Tomcat サーバーを始動します。
5. Tomcat エージェントを再構成して、Tomcat アプリケーション・サーバーを更新または変更します。
6. Tomcat アプリケーション・サーバーのみの更新または変更を行い、その他の構成設定は変更しないでください。
7. Tomcat エージェント・インスタンスを開始します。

8. データ・コレクターの開始パラメーターの新しい Tomcat アプリケーション・サーバーで setenv.sh ファイルが更新されていることを確認します。
9. Tomcat サーバーを再始動します。
10. Tomcat アプリケーション・サーバーに対して行われた変更がエージェント・マシンおよび IBM Cloud Application Performance Management ダッシュボードに反映されていることを確認します。
    - エージェント・マシン上の <CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/<Tomcat_Application_Server> で Tomcat アプリケーション・サーバーの変更を確認します
    - 「**集約トランザクション・トポロジー**」ページ、および IBM Cloud Application Performance Management ダッシュボードの *KOT_Server* 属性グループの *appserver* 属性で、Tomcat アプリケーション・サーバーの変更を確認します

VMware VI のモニターの構成

Monitoring Agent for VMware VI のインストール後、エージェントがモニター対象の VMware 仮想インフラストラクチャーのデータを収集できるように、最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

始める前に

- ハードウェアおよびソフトウェアの前提条件を確認します。
- ご使用の VMware Virtual Infrastructure でユーザー ID を作成します。エージェントは、VMware 仮想インフラストラクチャーをモニターするために、このユーザー ID を使用して VMware vCenter に接続します。モニター対象のすべての vCenter と ESX サーバーに対して「System.View」特権および「System.Read」特権を持っている必要があります。ユーザー ID を作成する方法については、ユーザー、グループ、許可、および役割の管理についての VMware 資料を参照してください。
- vCenter が SSL 通信用に構成されているかどうかを判別します。構成されている場合は、vCenter との通信に SSL を使用するように VMware VI エージェントを構成する必要があります。
  - vCenter が通信に SSL を使用するかどうかを判別するには、https://*vCenterIPaddress* URL を使用して vCenter にアクセスします。VCenter にアクセスできる場合、vCenter はネットワークを介して通信するために SSL を使用しています。
  - vCenter との通信に SSL を使用するように VMware VI エージェントを構成するには、810 ページの『VMware VI データ・ソースとの SSL 通信の使用可能化』で説明されているステップを実行します。
- VMware 仮想インフラストラクチャーのモニターに必要なエージェント・インスタンスの数を決定します。ご使用のモニター環境に従ったエージェント・インスタンスのサイズ設定については、810 ページの『VMware VI エージェントのデプロイメントのサイズ設定と計画』を参照してください。

このタスクについて

VMware VI エージェントは複数インスタンス・エージェントです。1 つのみのモニター対象アプリケーションのデータをモニターして収集するようにエージェントを構成できる単一インスタンス・エージェントとは異なり、VMware VI エージェントでは、複数の vCenter サーバーに接続して VMware 仮想インフラストラクチャーをリモートでモニターする複数のインスタンスを構成できます。

構成パラメーターにより、モニターする VMware VI データ・ソースを定義し、VMware vCenter、vCenter サーバー・アプライアンス、または個別の VMware ESX サーバーへの接続を定義します。該当するアプリケーションのサポートされるバージョンを確認するには、VMware VI エージェントの Software Product Compatibility Reports を参照してください。

製品バージョンとエージェント・バージョンは、通常、異なります。以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

すべてのエージェント属性のデータを表示するには、エージェントを手動で構成する必要があります。

- Windows オペレーティング・システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウまたはサイレント応答ファイルを使用できます。
- Linux オペレーティング・システム上でエージェントを構成するには、スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用することができます。

VMware VI エージェントのデプロイメントのサイズ設定と計画

単一のシステムで構成できるエージェント・インスタンスの数は、システムにおけるリソースの可用性と使用状況によって異なります。

以下の表では、VMware 環境をさまざまなサイズに分類し、必要な Java ヒープ・サイズを示しています。

表 *218. VMware* 環境と *Java* ヒープ・サイズ

VMware 環境のサイズ	**ESX サーバーの数**	**Java ヒープ・サイズ**
小規模環境	最大 125 台の ESX(i) サーバーおよび 300 から 1500 人のゲストを管理する vCenter サーバー。	**-Xmx2048m** (2 GB)
中規模環境	125 から 250 台までの ESX(i) サーバーおよび 1500 から 4000 人のゲストを管理する vCenter サーバー。	**-Xmx4096m** (4 GB)
大規模環境	250 から 500 台までの ESX(i) サーバーおよび 4000 から 7500 人のゲストを管理する vCenter サーバー。	**-Xmx8192m** (8 GB)
非常に大規模な環境	500 台を超える ESX(i) サーバーおよび 7500 人を超えるゲストを管理する vCenter サーバー。	**-Xmx16384m** (16 GB)

Java データ・プロバイダーのヒープ・サイズを増やすには、816 ページの『Java ヒープ・サイズの増大』で説明されているステップを実行します。

エージェント・インスタンスで環境を正常にモニターするには、エージェントをインストールするサーバーに、該当するエージェント・インスタンスによって収集されるデータに対応するための十分なメモリー・リソースがなければなりません。VMware VI エージェントの単一インスタンスでは、小規模環境をモニターするために、約 300 から 400 MB が必要です。構成するエージェント・インスタンスの数に関する以下のガイドラインを参照してください。

- 単一の vCenter をモニターするために単一インスタンスを使用します。同じインスタンスを使用して複数の vCenter をモニターしないでください。
- 非クラスター環境では、単一インスタンスを使用して、最大で 8 個の小規模 ESX サーバー (1 つの ESX サーバー内に 100 から 200 個の仮想マシン) をモニターします。単一のエージェント・インスタンスの下に複数の個別 ESX サーバーを構成しないでください。
- 複数の vCenter が含まれている環境をモニターするには、VMware VI エージェントの複数のエージェント・インスタンスを使用します。複数インスタンスを構成する前に、エージェントをインストールするシステム上に十分なメモリー・リソースがあることを確認してください。

VMware VI データ・ソースとの SSL 通信の使用可能化

エージェントが SSL を使用してその VMware VI データ・ソースとセキュアに通信するように構成する前に、データ・ソース SSL 証明書をエージェントの証明書トラストストアに追加する必要があります。

このタスクについて

重要 : 以下の情報は、SSL 証明書を検証するようにエージェントを構成する場合にのみ当てはまります。

SSL 証明書の検証がオフになっていると、VMware VI エージェントは、VMware データ・ソースの SSL 証明書が期限切れである、信頼されていない、または無効である場合でも VMware データ・ソースに接続できます。ただし、SSL 証明書の検証をオフにするとセキュリティーで保護されなくなる可能性があるため、実行には注意が必要です。

VMware データ・ソースで、共通認証局 (Verisign、Entrust、Thawte など) によって署名された SSL 証明書を使用する場合は、VMware VI エージェントの証明書トラストストアに証明書を追加する必要はありません。しかし、データ・ソースで、共通認証局によって署名されていない証明書を使用する場合 (これがデフォルトです) は、エージェントが正常に接続し、データを収集するためには、トラストストアに証明書を追加する必要があります。

注 :

1. デフォルトの VMware 証明書ファイルの名前は `rui.crt` です。
2. Virtual Center の場合の SSL 証明書ファイルは、デフォルトで以下のパスにあります。

   ```
   C:¥Documents and Settings¥All Users¥Application Data¥VMware¥VMware
   VirtualCenter¥SSL
   ```

3. ESX Server の場合の SSL 証明書ファイルは、デフォルトで `/etc/vmware/ssl` ディレクトリーにあります。

手順

1. データ・ソースからエージェント・コンピューターに証明書ファイルをコピーします。
2. エージェント・コンピューターで、選択したディレクトリーに証明書ファイルを配置します。証明書ファイルを上書きしないでください。追加する証明書ごとに固有のファイル名およびラベルを使用してください。
3. *keytool* コマンドを使用して、データ・ソース証明書をエージェントの証明書トラストストアに追加します。

   ```
   keytool -import -noprompt -trustcacerts -alias CertificateAlias -file
   CertificateFile -keystore Truststore -storepass TruststorePassword
   ```

   表の見方

   CertificateAlias

   エージェントの証明書トラストストアに追加される各証明書の固有の参照です。例えば、*datasource.example.com* からの証明書に対する適切な別名は *datasource* です。

   CertificateFile
   トラストストアに追加する VMware データ・ソース証明書への完全パスおよびファイル名です。

   トラストストア

   Complete path and file name to the VMware VI エージェント certificate database. 次のパスおよびファイル名を使用します。

   - Windows (64 ビット): *install_dir*¥tmaitm6_x64¥kvm.truststore
   - Linux (64 ビット): *install_dir*/lx8266/vm/etc/kvm.truststore

   TruststorePassword

   VMware VI エージェント・トラストストアのデフォルトのパスワードは ITMVMWAREVI です。このパスワードを変更するには、使用するツールについての Java ランタイムの資料を参照してください。

   重要 : *keytool* コマンドを使用するには、Java ランタイムの bin ディレクトリーが PATH になければなりません。以下のコマンドを使用します。

   - Windows (64 ビット): `set PATH=%PATH%;`*install_dir*`¥java¥java70_x64¥jre¥bin`
   - Linux (64 ビット): `PATH="$PATH":/opt/ibm/apm/agent/JRE/lx8266/bin`
4. すべてのデータ・ソース証明書を追加した後で、モニター・エージェントを開始します。

次のタスク

エージェント構成を完了します。

Windows システムでのエージェントの構成

「**IBM Performance Management**」ウィンドウを使用して、エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

このタスクについて

VMware VI エージェントでは、いくつかのパラメーターについてはデフォルト値が指定されています。これらのパラメーターに異なる値を指定できます。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで、「**Monitoring Agent for VMware VI**」を右クリックしてから「**エージェントの構成**」をクリックします。

   要確認 : 初めてエージェントを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェントを再度構成するには、「**再構成**」をクリックします。
3. 「Monitoring Agent for VMware VI」ウィンドウで、以下のステップを実行します。
   a) VMware VI エージェント・インスタンスの固有の名前を入力し、「**OK**」をクリックします.
   b) 「**データ・プロバイダー**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。
   c) 「**データ・ソース**」タブで、構成パラメーターの値を指定してから、「**次へ**」をクリックします。

   VMware VI エージェントは複数データ・ソースのエージェントです。同じエージェントからの複数のデータ・ソースをモニターできます。

   - 新規データ・ソースを構成する場合には、「**新規**」をクリックしてください。
   - 既存のデータ・ソースを削除する場合は、「**削除**」をクリックします。

   「Monitoring Agent for VMware VI」ウィンドウの各タブにある構成パラメーターについて詳しくは、以下のトピックを参照してください。

   - データ・プロバイダーの構成パラメーター
   - データ・ソースの構成パラメーター
4. 「**IBM Performance Management**」ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 500 を超える ESX ホストを持つ大規模な VMware 環境をモニターする場合は、Java データ・プロバイダーのヒープ・サイズの増加が必要になる可能性があります。詳しくは、816 ページの『Java ヒープ・サイズの増大』を参照してください。

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

- サイレント・モードで、VMware VI エージェントを構成するには、以下の手順を実行します。
  a) テキスト・エディターで、vmware_vi_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
  - Linux *install_dir*/samples/vmware_vi_silent_config.txt

    例: /opt/ibm/apm/agent/samples/vmware_vi_silent_config.txt
  - Windows *install_dir*¥samples¥vmware_vi_silent_config.txt

    例: C:¥IBM¥APM¥samples¥vmware_vi_silent_config.txt

  b) vmware_vi_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

  構成パラメーターについては、以下のトピックを参照してください。
  - 815 ページの『データ・プロバイダーの構成パラメーター』
  - 815 ページの『データ・ソースの構成パラメーター』

  c) vmware_vi_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
  - Linux *install_dir*/bin/vmware_vi-agent.sh config *instance_name* *install_dir*/samples/vmware_vi_silent_config.txt

    例: **/opt/ibm/apm/agent/bin/vmware_vi-agent.sh config instance_name /opt/ibm/apm/agent/samples/vmware_vi_silent_config.txt**
  - Windows *install_dir*¥bin¥vmware_vi-agent.bat config *instance_name* *install_dir*¥samples¥vmware_vi_silent_config.txt

    例: **C:¥IBM¥APM¥bin¥ vmware_vi-agent.bat config instance_name C:¥IBM¥APM ¥samples¥vmware_vi_silent_config.txt**

    表の見方

    instance_name
    インスタンスに指定する名前。

    install_dir
    エージェントのインストール先パス。

  重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

  d) 以下のコマンドを実行して、エージェントを開始します。
  - Linux *install_dir*/bin/vmware_vi-agent.sh start *instance_name*

    例: **/opt/ibm/apm/agent/bin/vmware_vi-agent.sh start instance_name**

– Windows *install_dir*¥bin¥vmware_vi-agent.bat start *instance_name*

例: **C:¥IBM¥APM¥bin¥vmware_vi-agent.bat start instance_name**

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 500 を超える ESX ホストを持つ大規模な VMware 環境をモニターする場合は、Java™ データ・プロバイダーのヒープ・サイズの増加が必要になる可能性があります。詳しくは、816 ページの『Java ヒープ・サイズの増大』を参照してください。

プロンプトへの応答によるエージェントの構成

Linux オペレーティング・システムでエージェントを 構成するには、スクリプトを実行してプロンプトに応答する必要があります。

手順

- スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

  a) コマンド・ラインで、以下のコマンドを実行します。

  install_dir/bin/vmware_vi-agent.sh config *instance_name*

  例: **/opt/ibm/apm/agent/bin/vmware_vi-agent.sh config instance_name**

  表の見方

  instance_name
  インスタンスに指定する名前。

  install_dir
  エージェントのインストール先パス。

  b) 以下のトピックを参照して、プロンプトに応答します。

  – 815 ページの『データ・プロバイダーの構成パラメーター』
  – 815 ページの『データ・ソースの構成パラメーター』

  c) 以下のコマンドを実行して、エージェントを開始します。

  install_dir/bin/vmware_vi-agent.sh start *instance_name*

  例: **/opt/ibm/apm/agent/bin/vmware_vi-agent.sh start instance_name**

次のタスク

- Cloud APM コンソールにログインして、エージェントにより収集されるデータをダッシュボードに表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

  トラブルシューティングにあたって 支援が必要な場合は、developerWorks の IBM Cloud APM Forum を参照してください。
- 500 を超える ESX ホストを持つ大規模な VMware 環境をモニターする場合は、Java™ データ・プロバイダーのヒープ・サイズの増加が必要になる可能性があります。詳しくは、816 ページの『Java ヒープ・サイズの増大』を参照してください。

データ・ソースの構成パラメーター

VMware VI エージェントを構成する際に、データ・ソースのアドレス、ユーザー ID、およびパスワードなど、データ・ソースのパラメーターのデフォルト値を変更できます。

以下の表に、データ・ソースーの構成パラメーターの詳細な説明を示します。

表 *219.* データ・ソースの構成パラメーターの名前と説明

パラメーター名	説明	必須フィールド
データ・ソース ID (Data Source ID)	データ・ソースの ID。	はい
データ・ソース・アドレス (Data Source Address)	データ・ソースのアドレス。 エージェントで SSL 証明書の検証を行わない場合は、この値を、モニター対象の VMware Virtual Center または ESX Server のホスト名または IP アドレスに設定します。 ネットワークを介した通信に SSL を使用する場合に、エージェントが SSL 証明書を検証する必要がある場合は、証明書に指定されている SAN (Subject Alternative Name) を使用してエージェントを構成します。 データ・センターのサブジェクトの別名を確認するには、以下の手順に従います。 1. 証明書を開きます。 2.「**証明書**」ウィンドウで、「**詳細**」タブをクリックします。 3.「**サブジェクトの別名 (Subject Alternative Name)**」を選択し、DNS 名の値を使用します。例えば、DNS 名の値が「`ibmesx3v3vc.ITMfVS.com`」である場合は、値「`ibmesx3v3vc.ITMfVS.com`」をホスト名に使用します。	はい
データ・ソースとの接続に SSL を使用 (Use SSL Connection to Data Source)	エージェントが VMware 仮想インフラストラクチャーのデータ・ソースへの接続に SSL 接続を使用するかどうかを指示します。 エージェントがデータ・ソースへの接続に SSL 接続を使用する場合は、「はい」を指定します。そうではない場合は、「いいえ」を指定します。デフォルト値は「はい」です。	はい
データ・ソース・ユーザー ID (Data Source User ID)	モニター・データの収集に十分な特権を備えていて、データ・ソースに認識されているユーザー ID。	はい
データ・ソース・パスワード	データ・ソースへのアクセス用に構成されているユーザー ID のパスワード。	はい
データ・ソース・パスワードの確認 (Confirm Data Source Password)	「**データ・ソース・パスワード**」フィールドで指定したものと同じパスワード。	

データ・プロバイダーの構成パラメーター

VMware VI エージェントを構成する際に、データ・プロバイダーのログ・ファイルの最大数、ログ・ファイルの最大サイズ、ログ・ファイルに含まれる詳細レベルなど、データ・プロバイダーのパラメーターのデフォルト値を変更できます。

以下の表に、データ・プロバイダーの構成パラメーターの詳細な説明を示します。

表 220. データ・プロバイダーの構成パラメーターの名前と説明		
パラメーター名	**説明**	**必須フィールド**
インスタンス名	The name of the instance. **制約事項 :** 「**インスタンス名**」フィールドでは、初めてエージェントを構成したときに指定したインスタンスの名前が表示されます。エージェントを再構成する場合に、エージェントのインスタンス名を変更することはできません。	はい
有効な SSL 証明書 (Valid SSL Certificates)	エージェントがネットワークを介した通信で SSL を使用する場合に、そのエージェントが SSL 証明書を検証するかどうかを示します。 ネットワーク上で SSL を使用して通信するときにエージェントが SSL 証明書を検証する必要がある場合は、値を Yes に設定します。エージェントが SSL 証明書を検証しないようにする場合は、値を No に設定します。デフォルト値は「はい」です。 エージェントの証明書トラストストアへのデータ・ソース SSL 証明書の追加については、810 ページの『VMware VI データ・ソースとの SSL 通信の使用可能化』を参照してください。	はい
データ・プロバイダー・ログ・ファイルの最大数	データ・プロバイダーが作成するログ・ファイルの最大数で、これを超えると以前のログ・ファイルが上書きされます。デフォルト値は 10 です。	はい
各データ・プロバイダー・ログの最大サイズ (KB)	データ・プロバイダーのログ・ファイルの最大サイズ (KB)。これに達すると、データ・プロバイダーによって新規ログ・ファイルが作成されます。デフォルト値は 5190 KB です。	はい
データ・プロバイダー・ログの詳細レベル	データ・プロバイダーによって作成されるログ・ファイルに含めることができる詳細のレベル。デフォルト値は INFO です。有効な値は、OFF、SEVERE、WARNING、INFO、FINE、FINER、FINEST、および ALL です。	はい
KEY_STORE_PASSWORD	**KEY_STORE_PASSWORD** を使用すると、ユーザーはエージェント JRE に設定された新しい鍵ストア・パスワードでエージェントを構成できます。この Java 鍵ストアは、vCenter の鍵ストアとは関係がないことに注意してください。 各構成でのパスワードの入力は必須ではありません。このフィールドをブランクのままにすると、エージェントは、エージェント JRE の使用中にデフォルトの Java 鍵ストア・パスワードを使用するものと想定します。	いいえ

Java ヒープ・サイズの増大

大規模な VMware 仮想インフラストラクチャー環境をモニターする場合は、VMware VI エージェントを構成した後で、Java™ データ・プロバイダーのヒープ・サイズを増やす必要がある場合があります。

このタスクについて

Java データ・プロバイダーのデフォルトの最大ヒープ・サイズは 256 メガバイトです。最大ヒープ・サイズは、VMware 環境のサイズに応じた適切な値に設定する必要があります。さまざまな VMware 環境で必要とされるヒープ・サイズについては、810 ページの表 218 を参照してください。

重要 : VMware VI エージェントをインストールして構成するシステムには、必要なヒープ・サイズに対応するために十分なメモリー・スペースがなければなりません。

以下のいずれかの問題が発生した場合に、ヒープ・サイズを増やす必要がある場合があります。

- javacore の問題が原因で Java データ・プロバイダーが停止し、javacore.*date.time.number*.txt という名前のファイルが CANDLEHOME¥tmaitm6_x64 ディレクトリー内に作成される。
- javacore.*date.time.number*.txt ファイルに java/lang/OutOfMemoryError というストリングが含まれている。

手順

- Windows
  ヒープ・サイズとして 1 GB の値を設定するには、以下の手順を実行します。
  1. %CANDLE_HOME%¥TMAITM6_x64¥kvm_data_provider.bat ファイルを開きます。
  2. KVM_JVM_ARGS="%KVM_CUSTOM_JVM_ARGS... で始まる行の前に次の行を追加します。
     ```
     SET KVM_CUSTOM_JVM_ARGS=-Xmx1024m
     ```
  3. エージェントを再始動します。
- Linux
  ヒープ・サイズとして 1 GB の値を設定するには、以下の手順を実行します。
  1. $CANDLEHOME/lx8266/vm/bin/kvm_data_provider.sh ファイルを開きます。
  2. KVM_JVM_ARGS="$KVM_CUSTOM_JVM_ARGS... で始まる行の前に次の行を追加します。
     ```
     KVM_CUSTOM_JVM_ARGS=-Xmx1024m
     ```
  3. エージェントを再始動します。

WebLogic モニターの構成

Monitoring Agent for WebLogic は、WebLogic サーバー環境の正常性、可用性、およびパフォーマンスについてモニターする中央ポイントを提供します。このエージェントでは、Java 仮想マシン (JVM)、Java メッセージング・サービス (JMS)、Java Database Connectivity (JDBC) などの WebLogic リソースに関して十分な情報に基づく意思決定をするのに役立つ、包括的な一連のメトリックが表示されます。

始める前に

- 以下の手順は、明記されている場合を除き、最新リリースのエージェント用です。
- ご使用の環境で、WebLogic エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Software Product Compatibility Reports (SPCR) for the WebLogic エージェント を参照してください。
- WebLogic エージェント を構成する前に、まず、次のタスクを実行することで、Oracle WebLogic サーバーを構成する必要があります。

  注 : Oracle WebLogic サーバー構成のほとんどは、管理コンソール (通常 http://*weblogic-server*:7001/console にある) を使用して実行できます。

  1. 「モニター」グループのモニター・ユーザーをセットアップします。
     a. モニター/編集対象のドメインを選択します。
     b. 「**セキュリティー・レルム**」を選択します。
     c. 使用するセキュリティー・レルムを選択 (存在しない場合は作成) します。
     d. JMX 経由で WebLogic と通信するために使用するユーザーを作成します。
     e. 「モニター」グループにこのユーザーを追加します。
     f. ドメインへの変更内容を保存します。
  2. listen ポートを有効にします。
     a. モニター/編集対象のドメインを選択します。

b. モニター対象のサーバーごとに、「**環境**」 > 「**サーバー**」 > 「***Select a server***」をクリックします。
c.「**Listen Port**」が有効になっていることを確認し、そのポート番号をメモします。
d. SSL を有効にする場合、「**SSL のリスナー・ポート**」が有効であることを確認し、SSL に対するポートも設定します。

3. JMX MBean サーバー接続を有効にします。
a. モニター/編集対象のドメインを選択します。
b.「**構成**」 > 「**拡張**」を選択します。
c.「**Platform Mbean Server Enabled**」にチェック・マークを付けます。
d. 変更内容を保存します。

4. IIOP プロトコル・オプションを有効にします。
a. モニター/編集対象のドメインを選択します。
b. モニター対象のサーバーごとに、「**環境**」 > 「**サーバー**」をクリックし、サーバーを選択します。
c.「**プロトコル**」**タブ** > 「***IIOP* の選択 *(Select IIOP)***」を選択します。
d.「**拡張**」セクションの下で、デフォルトの IIOP ユーザ名とパスワードを入力します。
e. 変更内容を保存します。

5. SSL を有効にします。
a. HTTP トンネリングを有効にします。
1)「**環境**」 > 「**サーバー**」 > 「***Select a server***」 > 「**プロトコル**」 > 「**一般**」に移動します。
2)「**HTTP トンネリングを有効にする**」にチェック・マークを付けます。
b. SSL listen ポートを有効にします。
1)「**環境**」 > 「**サーバー**」 > 「***Select a server***」 > 「**構成**」 > 「**一般**」に移動します。
2) ポート番号を構成します。

このタスクについて

WebLogic エージェントは複数インスタンス・エージェントであり、複数サブノード・エージェントでもあります。複数のサブノードを含む 1 つのエージェント・インスタンスを作成できます (各 WebLogic サーバーが 1 つのサブノードに対応します)。または、各 WebLogic サーバーに対して 1 つのエージェント・インスタンスを作成できます (そのサーバーに対応する 1 つのサブノードが含まれます)。または、各タイプの構成の組み合わせを作成できます。エージェント・インスタンスを構成した後で、各エージェント・インスタンスを手動で開始する必要があります。

手順

1. Windows システム上でエージェントを構成するには、「**IBM Performance Management**」ウィンドウを使用するか、またはサイレント応答ファイルとエージェント構成バッチ・ファイルを使用します。
   - 819 ページの『Windows システムでのエージェントの 構成』.
   - 824 ページの『サイレント応答ファイルを使用したエージェントの構成』.
2. Linux システムおよび UNIX システム上でエージェントを構成するには、エージェント構成スクリプトを実行してプロンプトに応答するか、またはサイレント応答ファイルを使用します。
   - 823 ページの『プロンプトへの応答によるエージェントの構成』.
   - 824 ページの『サイレント応答ファイルを使用したエージェントの構成』.
3. オプション: トランザクション・トラッキングを構成するには、トランザクション・トラッキング・データを提供するように個別のエージェント・インスタンスを構成し、トランザクション・トラッキング・データを表示するように Application Performance ダッシュボードを構成します。
   a) 826 ページの『WebLogic エージェントのトランザクション・トラッキングの構成』の手順に従ってください。

b) 832 ページの『WebLogic エージェントのトランザクション・トラッキング・データを表示するための Application Performance ダッシュボードの構成』の手順に従ってください。

注 : トランザクション・トラッキング機能は、Cloud APM, Advanced オファリングの WebLogic エージェント で使用できます。Cloud APM, Base オファリングに含まれる基本的なリソース・モニター機能を持つ WebLogic エージェントについては、この手順をスキップします。

次のタスク

Cloud APM コンソールで Application Performance ダッシュボードに移動し、収集されたデータを表示します。Cloud APM コンソールの使用について詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

エージェント・ダッシュボードでデータを表示できない場合は、まずサーバー接続ログを確認し、次にデータ・プロバイダー・ログを確認します。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:¥IBM¥APM¥TMAITM6_x64¥logs`

トラブルシューティングのヘルプについては、Cloud Application Performance Management フォーラム を参照してください。

Windows システムでのエージェントの構成

「IBM Cloud Application Performance Management」ウィンドウを使用して、WebLogic エージェントを Windows オペレーティング・システムで構成することができます。構成値を更新したら、エージェントを始動し、更新した値を保存する必要があります。

手順

1. 「**スタート**」>「**すべてのプログラム**」>「**IBM Monitoring エージェント (IBM Monitoring agents)**」>「**IBM Cloud Application Performance Management**」をクリックします。
2. 「**IBM Performance Management**」ウィンドウで「**Monitoring Agent for WebLogic**」テンプレートを右クリックし、「**エージェントを構成します**」をクリックします。

   要確認 : 初めてエージェント・インスタンスを構成した後には、「**エージェントの構成**」オプションは無効になっています。エージェント・インスタンスを再度構成するには、対象を右クリックしてから、「**再構成**」をクリックします。
3. 固有のインスタンス名を入力してから、「**OK**」をクリックします。インスタンス名では、英字、アラビア数字、下線文字、およびマイナス文字のみを使用してください。例えば、`weblogic01` です。

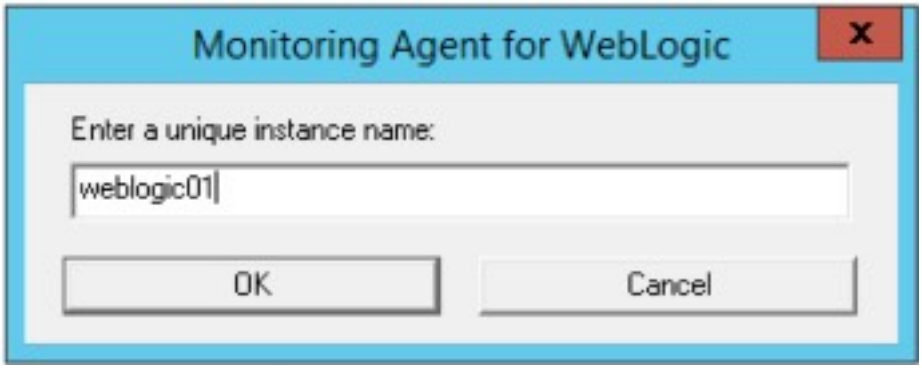


図 27. 固有のインスタンス名を入力するウィンドウ

4. 「**インスタンス名**」エージェント構成パネルで「**次へ**」をクリックします。

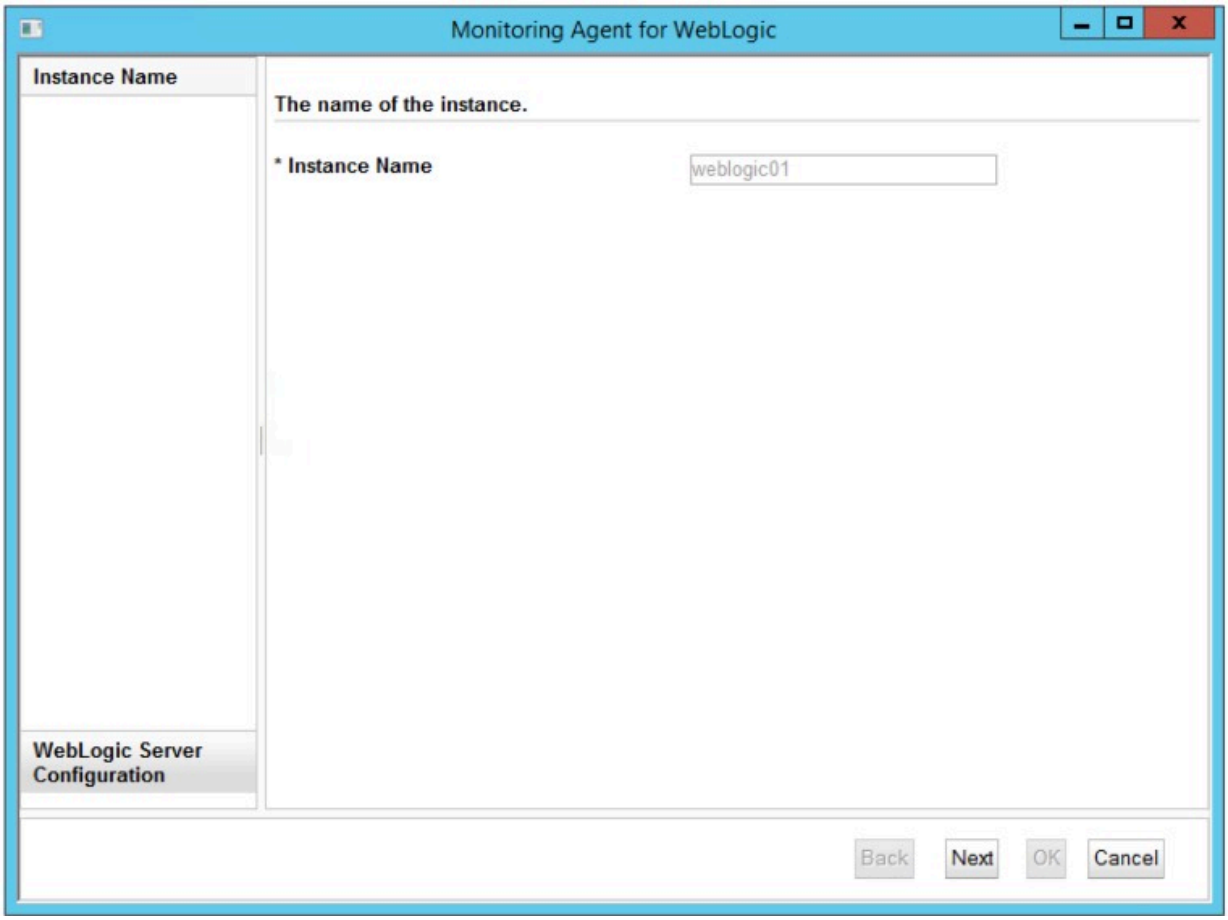


図 *28.* エージェント・インスタンス名を表示するウィンドウ

5. 「**WebLogic サーバー構成 (WebLogic Server Configuration)**」インスタンス・テンプレート設定を入力します。

   注 : このセクションは、WebLogic サーバー接続インスタンスの構成ではありません。ステップ 6 から実際の WebLogic サーバー接続インスタンスの構成を追加する際にデフォルト値として使用される内容を設定するためのテンプレート・セクションです。

   各構成パラメーターの説明については、825 ページの表 221 を参照してください。

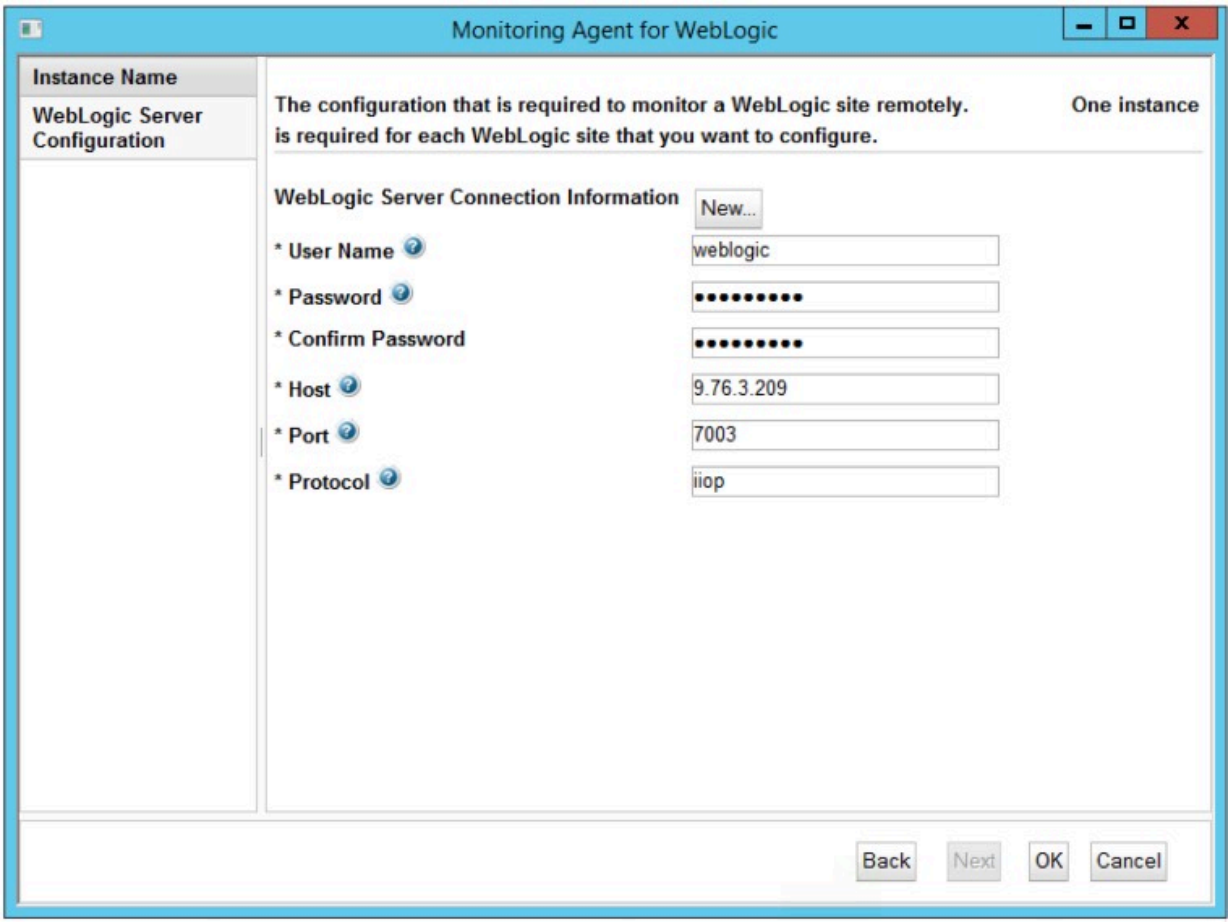


図 *29. WebLogic* サーバー接続インスタンス・テンプレート設定を指定するウィンドウ

6.「**新規**」をクリックし、WebLogic サーバー接続インスタンス設定を入力し、「**次へ**」をクリックします。各構成パラメーターの説明については、825 ページの表 221 を参照してください。

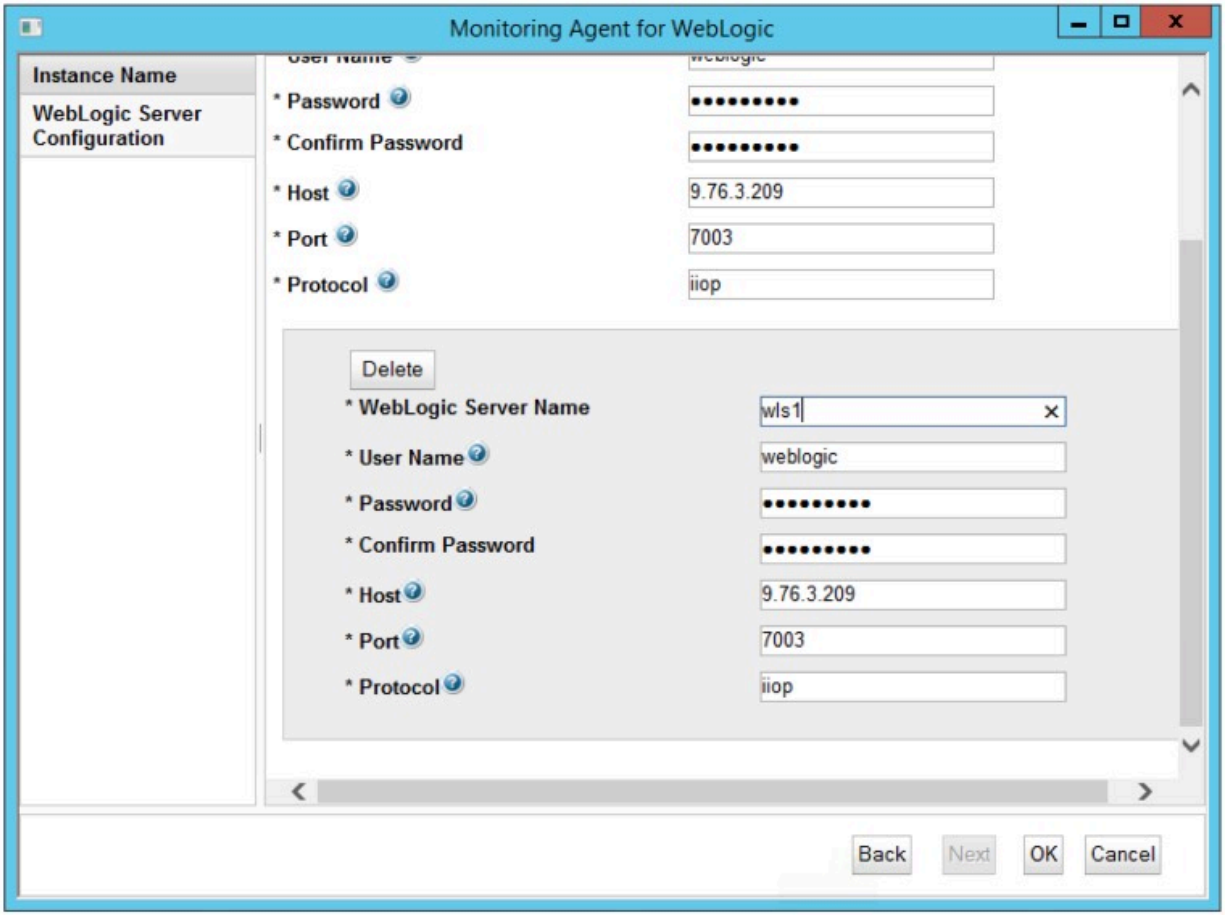


図 *30. WebLogic* サーバー接続インスタンス設定を指定するウィンドウ

7. 「**OK**」をクリックして、構成を完了します。
8. WebLogic セキュリティー・ファイルを WebLogic エージェントのバイナリー・ディレクトリーにコピーします。
   a. ORACLE_HOME の下で、`wlclient.jar` ファイルと `wljmxclient.jar` ファイルを見つけます。例えば、`C:¥Oracle¥Middleware¥Oracle_Home¥wlserver¥server¥lib` です。
   b. ステップ 822 ページの『8.a』 のファイルを WebLogic エージェントのバイナリー・ディレクトリーにコピーします。
      - Linux AIX *install_dir*/bin.
      - Windows *install_dir*¥TMAITM6_x64

      ここで、*install_dir* はエージェントのインストール先パスです。*install_dir* のデフォルト・パスは、以下のとおりです。
      - Linux AIX `/opt/ibm/apm/agent`
      - Windows `C:¥IBM¥APM¥TMAITM6_x64`
9. IBM Cloud Application Performance Management ウィンドウで、構成したインスタンスを右クリックしてから、「**開始**」をクリックします。

プロンプトへの応答によるエージェントの構成

WebLogic エージェントのインストール後、エージェントを開始する前に構成する必要があります。WebLogic エージェントがローカル Linux または UNIX コンピューターにインストールされている場合、以下の説明に従って、コマンド行プロンプトで対話式に構成できます。

このタスクについて

要確認 : 構成済みのエージェント・インスタンスを再構成する場合は、前回の構成で設定された値が設定ごとに表示されます。既存の値をクリアする場合は、設定が表示されているときにスペース・キーを押します。

手順

以下のステップに従って、スクリプトを実行してプロンプトに応答することで WebLogic エージェントを構成します。

1. 以下のコマンドを実行します。

   install_dir/bin/weblogic-agent.sh config **instance_name**

   ここで、*install_dir* はエージェントのインストール先パス、*instance_name* はエージェント・インスタンスに指定する名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/weblogic-agent.sh config example-inst01
   ```

2. プロンプトに応答してエージェントの構成値を設定します。

   各構成パラメーターの説明については、825 ページの『WebLogic エージェントの構成パラメーター』を参照してください。

3. WebLogic クライアント・ライブラリー・ファイルを WebLogic エージェント・バイナリー・ディレクトリーにコピーします。

   a) ORACLE_HOME の下で、wlclient.jar ファイルと wljmxclient.jar ファイルを見つけます。

   b) ステップ 823 ページの『3.a』 のファイルを WebLogic エージェントのバイナリー・ディレクトリーにコピーします。

   install_dir/bin

   ここで、*install_dir* はエージェントのインストール先パスです。

   例

   ```
   /opt/ibm/apm/agent/bin
   ```

4. 以下のコマンドを実行して、エージェントを開始します。

   install_dir/bin/weblogic-agent.sh start **instance_name**

   ここで、*install_dir* はエージェントのインストール先パス、*instance_name* はエージェント・インスタンスの名前です。

   例

   ```
   /opt/ibm/apm/agent/bin/weblogic-agent.sh start example-inst01
   ```

サイレント応答ファイルを使用したエージェントの構成

サイレント応答ファイルには、エージェント構成パラメーターがあります。サイレント応答ファイルを編集して各パラメーターの値を変更し、構成スクリプトを実行してインスタンスを作成し、エージェント構成値を更新できます。この構成モードは、サイレント・モードと呼ばれます。

このタスクについて

サイレント応答ファイルには、エージェントの構成パラメーターが格納されていて、いくつかのパラメーターについてはデフォルト値が定義されています。サイレント応答ファイルを編集して、構成パラメーターに異なる値を指定できます。

サイレント応答ファイルの構成値を更新したら、構成スクリプトを実行し、更新したそれらの値でエージェントを構成する必要があります。

手順

以下の手順を実行して、サイレント・モードで WebLogic エージェントを構成します。

1. テキスト・エディターで、weblogic_silent_config.txt ファイルを開きます。このファイルは、以下のパスにあります。
   - Linux AIX *install_dir*/samples/weblogic_silent_config.txt
   - Windows *install_dir*¥samples¥weblogic_silent_config.txt

   ここで、*install_dir* はエージェントのインストール先パスです。

   例
   - Linux AIX /opt/ibm/apm/agent/samples/weblogic_silent_config.txt
   - Windows C:¥IBM¥APM¥samples¥weblogic_silent_config.txt
2. weblogic_silent_config.txt ファイルで、すべての必須パラメーターに値を指定します。他のパラメーターのデフォルト値を変更することもできます。

   各構成パラメーターの説明については、825 ページの『WebLogic エージェントの構成パラメーター』を参照してください。
3. weblogic_silent_config.txt ファイルを保存して閉じ、次のコマンドを実行します。
   - Linux AIX *install_dir*/bin/weblogic-agent.sh config **instance_name** *install_dir*/samples/weblogic_silent_config.txt
   - Windows *install_dir*¥bin¥weblogic-agent.bat config **instance_name** *install_dir*¥samples¥weblogic_silent_config.txt

   ここで、*install_dir* はエージェントのインストール先パス、*instance_name* はエージェント・インスタンスに指定する名前です。

   install_dir のデフォルト・パスは、以下のとおりです。
   - Linux AIX /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64

   重要 : サイレント応答ファイルの絶対パスを必ず含めてください。このように指定しないと、ダッシュボードにエージェント・データが表示されません。

   例
   - Linux AIX /opt/ibm/apm/agent/bin/weblogic-agent.sh config example-inst01 /opt/ibm/apm/agent/samples/weblogic_silent_config.txt
   - Windows C:¥IBM¥APM¥bin¥weblogic-agent.bat config example-inst01 C:¥IBM¥APM¥samples¥weblogic_silent_config.txt

4. WebLogic クライアント・ライブラリーを WebLogic エージェント・バイナリー・ディレクトリーにコピーします。
   a. ORACLE_HOME の下で、wlclient.jar ファイルと wljmxclient.jar ファイルを見つけます。
   b. ステップ 825 ページの『5.a』 のファイルを WebLogic エージェントのバイナリー・ディレクトリーにコピーします。
      - Linux AIX *install_dir*/bin.
      - Windows *install_dir*¥TMAITM6_x64

   ここで、*install_dir* はエージェントのインストール先パスです。*install_dir* のデフォルト・パスは、以下のとおりです。
   - Linux AIX /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64
5. 以下のコマンドを実行して、エージェントを開始します。
   - Linux AIX *install_dir*/bin/weblogic-agent.sh start **instance_name**
   - Windows *install_dir*¥bin¥weblogic-agent.bat start **instance_name**

   ここで、*install_dir* はエージェントのインストール先パス、*instance_name* はエージェント・インスタンスの名前です。

   install_dir のデフォルト・パスは、以下のとおりです。
   - Linux AIX /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64

   例
   - Linux AIX /opt/ibm/apm/agent/bin/weblogic-agent.sh start example-inst01
   - Windows C:¥IBM¥APM¥bin¥weblogic-agent.bat start example-inst01

WebLogic エージェントの構成パラメーター

WebLogic エージェントの構成パラメーターを表に表示します。

1. WebLogic エージェント設定 - WebLogic エージェント環境設定。

表 *221. WebLogic* エージェント設定

パラメーター名	説明	サイレント構成ファイルのパラメーター名
WebLogic サーバー名 (WebLogic Server Name)	WebLogic サーバー・エージェント・インスタンスを識別する名前を指定します。例: *wls1* **注 :** この別名は、WebLogic サーバー・エージェント・インスタンスを表す任意のものにすることができますが、以下の制限があります。接続名に使用できるのは、文字、アラビア数字、下線文字、およびマイナス文字のみです。 接続名の最大長は 25 文字です。	以下の各パラメーターには、インスタンス名サフィックスを付ける必要があります。このサフィックスは、1 つのエージェント・インスタンスのどのパラメーターでも同じです。新規エージェント・インスタンスのパラメーターのセットでは、固有のインスタンス名を使用する必要があります。例えば、以下のパラメーター名の *.instance_name* の代わりに、あるエージェント・インスタンスでは *.wls1* を使用でき、別のエージェント・インスタンスでは *.wls2* を使用できます。
ユーザー名	WebLogic サーバーでの認証に使用するユーザー名。	**KWB_WLS_USERNAME.instance_name**

表 221. WebLogic エージェント設定 (続き)		
パラメーター名	**説明**	**サイレント構成ファイルのパラメーター名**
パスワード	WebLogic サーバーでの認証に使用するパスワード。	**KWB_WLS_PASSWORD.instance_name**
ホスト	WebLogic サーバーでの認証に使用するホスト。WebLogic サーバーの完全修飾ホスト名または IP アドレスを入力します。	**KWB_WLS_HOST.instance_name**
ポート	WebLogic サーバーでの認証に使用するポート。	**KWB_WLS_PORT.instance_name**
プロトコル	WebLogic サーバーでの認証に使用するプロトコル。サポートされるプロトコルは、*iiop* と *https* です。	**KWB_WLS_PROTOCOL.instance_name**

WebLogic エージェントのトランザクション・トラッキングの構成

WebLogic エージェントのトランザクション・トラッキング機能を使用するには、エージェント・インスタンス環境設定ファイルおよび WebLogic サーバー・スタートアップ・ファイルを変更する必要があります。これらの変更を行うために役立つスクリプトが用意されています。

始める前に

Linux AIX トランザクション・トラッキング・ツールキットが適切に機能するように、オープン・ファイルのリソース制限が 5,000 より大きいことを確認します。

- 現在のオープン・ファイル制限設定の表示: **ulimit -n**
- オープン・ファイル制限を 5,056 に設定する例: **ulimit -n 5056**

この手順を実行する前に、817 ページの『WebLogic モニターの構成』の Windows ステップ 1 または Linux AIX ステップ 2 を行います。

注 : トランザクション・トラッキング機能は、Cloud APM, Advanced オファリングの WebLogic エージェント で使用できます。Cloud APM, Base オファリングに含まれる基本的なリソース・モニター機能を持つ WebLogic エージェントについては、この手順をスキップします。

WebLogic エージェントは、トランザクション・トラッキング機能によってモニターする WebLogic サーバーにローカルでインストールする必要があります。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *WEBLOGIC_HOME* ディレクトリー
2. *WEBLOGIC_HOME*/bin ディレクトリーおよびファイル
3. *install_dir*/config ディレクトリー。
4. *install_dir*/config/*hostname*_wb_*instance_name*.cfg ファイル。

ここで、

WEBLOGIC_HOME
WebLogic Server のインストール・ディレクトリーです。

install_dir
エージェントのインストール先パスです。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

hostname
エージェントがインストールされるホスト・コンピューターの名前です。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。

- Windows システムでのエージェントの 構成の場合は、手順 819 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 823 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 824 ページの『2』

手順

simpleConfig スクリプトを実行します。

1. WebLogic エージェントがインストールされている WebLogic サーバーにログインします。
2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux AIX *install_dir*
   - Windows *install_dir*¥TMAITM6_x64

   ここで、*install_dir* はエージェントのインストール先パスです。

   install_dir のデフォルト・パスは、以下のとおりです。
   - Linux AIX /opt/ibm/apm/agent
   - Windows C:¥IBM¥APM¥TMAITM6_x64
3. ディレクトリーを wbdchome/8.1.4.0.0/bin に切り替えます。
4. セットアップ・スクリプトを実行します。
   - Linux AIX ./simpleConfig.sh
   - Windows simpleConfig.bat
5. プロンプトに従って、次のようにご使用の環境に合わせてパラメーターを入力します。
   a) 検出されたエージェント・インスタンスおよびサブノードの 組み合わせのリストから、構成する WebLogic エージェントの *instance_name* およびサブノードを選択します。ここで、*instance_name* はエージェント・インスタンスの名前です。

   b) WebLogic サーバー開始方法の番号を入力します。

   c) WebLogic ドメインのルート検索パスを入力します。
   このパスは、WebLogic ドメインの検索を開始するためのベースとして使用されます。
   WEBLOGIC_HOME 環境変数が設定されている場合、その値がデフォルト値として提供されます。

   d) 構成する WebLogic サーバーの WebLogic ドメインの番号を入力します。

   e) 構成する WebLogic サーバー名の番号を入力します。

Linux WebLogic 開始方法として `WebLogic startup script` を選択した場合の構成の例。

```
./simpleconfig.sh

The following agents and subnodes are not yet configured for transaction tracking:

1) wlinst1 Server1
2) wlinst1 Server2

Type the number that corresponds to the agent instance and subnode that you want to
configure.

Type your selection here (For example: 1): 1

The following WebLogic startup methods are supported:

1) WebLogic startup script
2) WebLogic Node Manager

Type your selection here (default is 1): 1
```

```
The path to begin looking for WebLogic domains.
WebLogic domain search root (default is: ): /home/wlsadmin

The found WebLogic domain paths are:

1) /home/wlsadmin/oracle/user_projects/domains/ttdd

Type the number that corresponds to the WebLogic domain containing the WebLogic server that
you want to configure.

Type your selection here (For examble: 1): 1

The following WebLogic servers are available for configuration:

1) AdminServer
2) Server1

Select a WebLogic server name (default is: 2): 2
INFO: [2000] Automatic configuration of agent environment file succeeded.
INFO: [3000] Automatic configuration of WebLogic start script succeeded.
INFO: [9000] Restart the WebLogic agent and WebLogic server for configuration to take effect.
```

6. ステップ 827 ページの『5.b』 で **WebLogic server startup method** として WebLogic Node Manager を選択した場合は、以下のステップを実行します。あるいは、829 ページの『7』 に進んでください。

   a) ステップ 827 ページの『5』 の出力テキストの通知メッセージ番号 3011 にリストされている weblogic_nodemanager_dc_opts ファイルを開きます。

   Windows WebLogic 開始方法として WebLogic Node Manager を選択した場合の構成出力の例。

   ```
   INFO: [2000] Automatic configuration of agent environment file succeeded.
   INFO: [3010] Automatic configuration of WebLogic start script skipped.
   INFO: [3011] Please review C:¥IBM¥APM¥TMAITM6_x64¥wbdchome¥8.1.4.0.0¥runtime¥ttdd_win
   ¥win_Server1¥
   staging¥weblogic_nodemanager_dc_opts.win for required WebLogic JVM start options.
   INFO: [9000] Restart the WebLogic agent and WebLogic server for configuration to take
   effect.
   ```

   b) WebLogic コンソールにログインし、「**環境**」 > 「**サーバー**」を選択します。

   c) 構成するサーバーを選択します。

   d)「**構成**」 > 「**サーバー始動 (Server Start)**」タブを選択します。

   e) WebLogic コンソールで、weblogic_nodemanager_dc_opts ファイルからサーバーの「**サーバーの始動 (Start Server)**」引数にサーバー始動引数をコピーし、変更を保存します。

   サーバー始動引数は、weblogic_nodemanager_dc_opts ファイル内のコメント行 # Add the following lines to the server start arguments に続くすべての行です。

   f) トランザクション・トラッキング・ツールキットが実行時に共有ライブラリー・パス内にあることを確認します。

   方法を選択します。

   - Node Manager の開始スクリプトを更新します。

     注 : Node Manager によって始動されたすべての WebLogic サーバーでは、このライブラリー・パスが設定され、トランザクション・トラッキング・ツールキットのオブジェクト・ファイル・ライブラリーが組み込まれています。

     1) ステップ 827 ページの『5』 の出力テキストの通知メッセージ番号 3011 にリストされている weblogic_nodemanager_dc_opts ファイルを開きます。

     2) Node Manager の開始スクリプト内のトランザクション・トラッキング・ツールキットのパスを設定します。パスを設定するコマンドは、weblogic_nodemanager_dc_opts ファイル内のコメント行 # Make sure that the executable path available to the WebLogic server includes the toolkit lib directory に続く行です。

        – Linux 生成された weblogic_nodemanager_dc_opts.linux ファイルから LD_LIBRARY_PATH 行をコピーし、Node Manager の開始スクリプト内の export

JAVA_OPTIONS 行の下に LD_LIBRARY_PATH 行を貼り付けます。例えば、*WEBLOGIC_HOME*/user_projects/domains/*domain_name*/bin/startNodeManager.sh です。

– Windows 生成された weblogic_nodemanager_dc_opts.win ファイルから PATH をコピーし、Node Manager の開始スクリプト内の export JAVA_OPTIONS 行の下に PATH を貼り付けます。例えば、*WEBLOGIC_HOME*¥user_projects¥domains¥*domain_name*¥bin¥startNodeManager.bat です。

ここで、*WEBLOGIC_HOME* は、WebLogic サーバーのインストール・ディレクトリーであり、*domain_name* は、WebLogic ドメインの名前です。

- Node Manager を開始するユーザー・アカウントの環境を更新します。

注 : このユーザー・アカウントによって開始されたすべてのアプリケーションでは、このライブラリー・パスが設定され、ツールキットのオブジェクト・ファイル・ライブラリーが組み込まれています。

1) Node Manager を開始するユーザーの環境設定を編集します。

– Linux AIX シェル・リソース・ファイルまたはシェル・プロファイル・ファイルを編集します。例えば、bash シェルでは、.bashrc または .bash_profile です。

– Windows 「**コントロール パネル**」 > 「**システムとセキュリティ**」 > 「**システム**」 > 「**システムの詳細設定**」 > 「**環境変数...**」 > 「***user_name* のユーザー環境変数**」 > 「**Path**」を編集します。ここで、*user_name* は、WebLogic サーバーの始動に使用されるユーザー・アカウントの名前です。

2) ユーザー・アカウントの環境内のトランザクション・トラッキング・ツールキットのパスを設定します。パスを設定するコマンドは、weblogic_nodemanager_dc_opts ファイル内のコメント行 # Make sure that the executable path available to the WebLogic server includes the toolkit lib directory に続く行です。

– Linux AIX 生成された weblogic_nodemanager_dc_opts.linux ファイルから export LD_LIBRARY_PATH 行をコピーします。export LD_LIBRARY_PATH 行が存在しない場合は、追加します。変数が存在しない場合は、等号の右側のパスのみが既存のパスに追加され、正しいパス区切り文字が指定されるように、値を編集します。

– Windows 生成された weblogic_nodemanager_dc_opts.win ファイルから set PATH 行をコピーします。「**Path**」変数が「***user_name* のユーザー環境変数**」セクションに存在しない場合 (*user_name* は、WebLogic サーバーの始動に使用されるユーザー・アカウントの名前)、Path を変数名として入力し、等号の右側のパスを値として入力して、変数を追加します。変数が存在しない場合は、等号の右側のパスのみが既存のパスに追加され、正しいパス区切り文字が指定されるように、値を編集します。

3) 環境を再ロードします。

警告 : startNodeManager スクリプトは、WebLogic 構成ユーティリティーによって生成されます。したがって、WebLogic 構成が再度実行されると、変更が失われることがあります。

7. WebLogic サーバーおよびエージェントが実行中である場合は、再始動します。

タスクの結果

トランザクション・トラッキングの構成中に変更される WebLogic サーバー・ファイルは以下のとおりです。

- startManagedWebLogic スクリプト
  – Linux AIX *WEBLOGIC_HOME*/bin/startManagedWebLogic.sh
  – Windows *WEBLOGIC_HOME*¥bin¥startManagedWebLogic.cmd

ここで、*WEBLOGIC_HOME* は、WebLogic サーバーのインストール・ディレクトリーです。

このファイルは、トランザクション・トラッキング機能の構成設定で更新されます。このファイルには、トランザクション・トラッキング機能を使用不可にする場合に使用する構成マーカーが挿入されます。スクリプトがトランザクション・トラッキング機能の変更を追加または削除する前に、*WEBLOGIC_HOME*/bin/bak/ ディレクトリーにバックアップ・ファイルが保存されます。

トランザクション・トラッキングの構成中に変更されるエージェント・ファイル:

- エージェント・インスタンス構成ファイル
  - Linux AIX *install_dir*/config/*hostname*_wb_*instance_name*.cfg
  - Windows *install_dir*¥TMAITM6_x64¥*hostname*_WB_*instance_name*.cfg
- エージェント環境設定ファイル
  - Linux AIX *install_dir*/config/wb_*instance_name*.environment
  - Windows *install_dir*¥TMAITM6_x64¥KWBENV_*instance_name*

ここで、

install_dir
エージェントのインストール先パスです。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

hostname
エージェントがインストールされるホスト・コンピューターの名前です。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。

- Windows システムでのエージェントの 構成の場合は、手順 819 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 823 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 824 ページの『2』

WebLogic エージェント・インスタンスのトランザクション・トラッキングの無効化

WebLogic エージェントのトランザクション・トラッキング機能は削除が可能です。エージェント・インスタンスのトランザクション・トラッキング機能を削除するために、スクリプトが提供されています。

始める前に

WebLogic サーバーおよび WebLogic エージェントがシャットダウンしていることを確認します。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *WEBLOGIC_HOME* ディレクトリー
2. *WEBLOGIC_HOME*/bin ディレクトリーおよびファイル
3. *install_dir*/config ディレクトリー。
4. *install_dir*/config/*hostname*_wb_*instance_name*.cfg ファイル。

手順

remove オプションを指定して **unconfig** スクリプトを実行します。

1. WebLogic エージェントがインストールされている WebLogic サーバーにログインします。
2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux AIX *install_dir*
   - Windows *install_dir*¥TMAITM6_x64

3. ディレクトリーを wbdchome/8.1.4.0.0/bin に切り替えます。
4. **remove** オプションおよびエージェント・インスタンス名とサブノード名を指定して **unconfig** を実行します。
   - エージェント・インスタンスの 1 つのサブノードを無効にするには、*subnode_name* パラメーターを使用します。
     - Linux AIX ./unconfig.sh **remove** *instance_name subnode_name*
     - Windows unconfig.bat **remove** *instance_name subnode_name*
   - エージェント・インスタンスのすべてのサブノードを 無効にするには、*subnode_name* パラメーターを省略します。
     - Linux AIX ./unconfig.sh **remove** *instance_name*
     - Windows unconfig.bat **remove** *instance_name*
5. WebLogic サーバーおよびエージェントを始動します。

ここで、

WEBLOGIC_HOME
WebLogic Server のインストール・ディレクトリーです。

hostname
エージェントがインストールされるホスト・コンピューターの名前です。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。

- Windows システムでのエージェントの 構成の場合は、手順 819 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 823 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 824 ページの『2』

subnode_name
エージェントの構成方法のトピックで「**WebLogic サーバー名**」パラメーターに割り当てられたエージェント・インスタンス・サブノードの名前です。

- Windows システムでのエージェントの 構成の場合は、手順 821 ページの『6』
- プロンプトへの応答によるエージェントの構成の場合は、手順 823 ページの『2』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 824 ページの『2』

install_dir
エージェントのインストール先パスです。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

WebLogic エージェントのトランザクション・トラッキングのアンインストール

WebLogic エージェントのトランザクション・トラッキング機能はアンインストールが可能です。すべてのエージェント・インスタンスからトランザクション・トラッキング機能を削除し、さらにトランザクション・トラッキング・ツールキットを削除するためのスクリプトが用意されています。

始める前に

WebLogic サーバーとすべての WebLogic エージェント・インスタンスがシャットダウンされていることを確認します。

このスクリプトを実行するユーザー・アカウントには、以下のディレクトリーおよびファイルに対する書き込み権限が必要です。

1. *WEBLOGIC_HOME* ディレクトリー
2. *WEBLOGIC_HOME*/bin ディレクトリーおよびファイル

3. *install_dir*/config ディレクトリー。
4. *install_dir*/config/*hostname*_wb_*instance_name*.cfg ファイル。

手順

uninstall オプションを指定して **unconfig** スクリプトを実行します。

1. WebLogic エージェントがインストールされている WebLogic サーバーにログインします。
2. ディレクトリーをエージェント・インストール・ディレクトリーに切り替えます。
   - Linux AIX *install_dir*
   - Windows *install_dir*¥TMAITM6_x64
3. ディレクトリーを wbdchome/8.1.4.0.0/bin に切り替えます。
4. **uninstall** オプションを指定して **unconfig** を実行します。
   - Linux AIX ./unconfig.sh **uninstall**
   - Windows unconfig.bat **uninstall**
5. WebLogic サーバーおよびすべてのエージェント・インスタンスを開始します。

ここで、

WEBLOGIC_HOME
WebLogic Server のインストール・ディレクトリーです。

hostname
エージェントがインストールされるホスト・コンピューターの名前です。

instance_name
エージェントの構成方法のトピックで割り当てられたエージェント・インスタンスの名前です。

- Windows システムでのエージェントの 構成の場合は、手順 819 ページの『3』
- プロンプトへの応答によるエージェントの構成の場合は、手順 823 ページの『1』
- サイレント応答ファイルを使用したエージェントの構成の場合は、手順 824 ページの『2』

install_dir
エージェントのインストール先パスです。これらのログのデフォルト・パスは、以下のとおりです。

- Linux AIX /opt/ibm/apm/agent
- Windows C:¥IBM¥APM¥TMAITM6_x64

WebLogic エージェントのトランザクション・トラッキング・データを表示するための Application Performance ダッシュボードの構成

WebLogic エージェントのトランザクション・トラッキング機能によって収集されたデータを表示するには、Application Performance ダッシュボードを構成変更する必要があります。

始める前に

この手順を実行する前に、826 ページの『WebLogic エージェントのトランザクション・トラッキングの構成』を行います。

手順

1. Cloud APM, Advanced オファリングに含まれるトランザクション・トラッキング機能を持つ WebLogic エージェントがあり、トランザクション・トラッキング機能を有効にする場合は、Application Performance ダッシュボードでトランザクション・トラッキング・データを有効にします。
   a) ナビゲーション・バーで、「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
   b)「**WebLogic**」タブを選択します。

c) モニターする WebLogic サーバー・エージェントのチェック・ボックスを選択し、「**アクション**」リストから次のいずれかのアクションを実行します。

- トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。
- トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。

2. WebLogic エージェントのトランザクション・トラッキング・データのダッシュボードを表示するには、WebLogic エージェント・インスタンスを Application Performance ダッシュボード 内のアプリケーションに追加します。

   アプリケーション・エディターについて詳しくは、『アプリケーションの管理』を参照してください。

3. ユーザー・アカウントが、WebLogic エージェントのトランザクション・トラッキング・アプリケーション・ダッシュボードの、次のボタンにアクセスするための診断ダッシュボード許可が含まれるロールに割り当てられていることを確認します。

   そのように割り当てられていない場合、アプリケーション・ダッシュボード内で、そのユーザーに対してこれらのボタンが無効になります。

   a.「**最も遅い 5 応答時間**」ウィジェットの「**診断**」ドリルダウン・ボタン。

   b.「**アプリケーション**」ウィジェットの「**処理中要求**」ボタン。

WebSphere Applications モニターの構成

WebSphere Applications モニターを構成するには、アプリケーション・サーバー用にデータ・コレクターを構成します。スタンドアロンのデータ・コレクターでも、WebSphere Applications エージェントに組み込まれたデータ・コレクターでも使用できます。

組み込みデータ・コレクター

IBM Cloud 上の Liberty プロファイルを除き、ほとんどの WebSphere アプリケーション・サーバーは、組み込みデータ・コレクターによってモニターできます。組み込みデータ・コレクターは、使用可能なモニター機能のすべてを提供できます。

組み込みデータ・コレクターを構成するには、まず、アプリケーション・サーバーが実行されているシステムに WebSphere Applications エージェントをインストールする必要があります。その後、提供されている構成ユーティリティーを 使用して、データ・コレクターを対話式またはサイレントに構成します。

スタンドアロンのデータ・コレクター

スタンドアロン・データ・コレクターは、Linux for System x 上の WebSphere Application Server Liberty および IBM Cloud 上の WebSphere Liberty プロファイルにのみ適用できます。

スタンドアロンのデータ・コレクターを構成する場合は、エージェントのインストール手順をスキップして、Liberty で直接、データ・コレクターを構成することができます。

ただし、スタンドアロンのデータ・コレクターでは、現在時刻のヒープ・ダンプやインフライト要求情報など、一部のオンデマンド診断データが収集されません。つまり、データ・コレクターを有効にして、指定した間隔でのみ、自動的にヒープ・ダンプ情報を収集することはできますが、Cloud APM コンソールの「**スナップショットの作成**」ボタンを使用して、随時、ヒープ・スナップショットを作成することはできません。 組み込みデータ・コレクターが提供できる、インフライト要求に関連するダッシュボードはいずれも、スタンドアロンのデータ・コレクターでは利用できません。

ご使用のアプリケーション・サーバーに適切なデータ・コレクターを判別するには、834 ページの表 222 を使用してください。

表 222. WebSphere アプリケーションおよび適用可能なデータ・コレクター		
モニター対象のアプリケーション	**適用可能なデータ・コレクター**	**資料**
WebSphere Application Server traditional	組み込みデータ・コレクター	834 ページの『WebSphere Applications エージェント』
WebSphere Application Server Liberty (オンプレミス)	• 組み込みデータ・コレクター • スタンドアロンのデータ・コレクター (Linux for System x のみ)	• 834 ページの『WebSphere Applications エージェント』 • 881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』
IBM Cloud 上の WebSphere Liberty プロファイル	スタンドアロンのデータ・コレクター	886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』
Docker コンテナー内の WebSphere Liberty プロファイル	組み込みデータ・コレクター	868 ページの『Docker コンテナー内の WebSphere Application Server Liberty のモニター』

WebSphere Applications エージェント

WebSphere Applications エージェントでは、デフォルト・ポートを変更しない限り、エージェントのインストール後に構成は不要です。ただし、WebSphere 環境用のモニターをセットアップするために、エージェントのコンポーネントであるデータ・コレクターを構成する必要があります。

このタスクについて

以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

手順

- (ファスト・トラック) 環境内にレガシー製品 ITCAM Agent for WebSphere Applications がない状態で WebSphere Applications エージェントのみを使用する場合、モニター用に環境を迅速にセットアップするための簡素化された構成フローについては、835 ページの『ファスト・トラック: WebSphere Applications エージェント用のデータ・コレクターの構成』を参照してください。
- (簡易構成) 純粋な IBM Cloud Application Performance Management 環境の完全な構成フローについては、838 ページの『簡易構成ユーティリティーを 使用したデータ・コレクターの構成』を参照してください。
- (フル構成) 追加のカスタマイズ・オプションを設定してデータ・コレクターを構成するには、フル構成ユーティリティーを 使用します。説明については、840 ページの『フル構成ユーティリティーを 使用したデータ・コレクターの構成または再構成』を参照してください。
- (サイレント構成) 多数のアプリケーション・サーバー・インスタンスに対して同じモニターをデプロイするには、データ・コレクターをサイレント・モードで構成します。説明については、849 ページの『サイレント・モードでのデータ・コレクターの構成』を参照してください。
- (WebSphere Portal Server) WebSphere Portal Server インスタンスをモニターするには、拡張構成手順を使用します。説明については、840 ページの『フル構成ユーティリティーを 使用したデータ・コレクターの構成または再構成』を参照してください。
- (手動構成) 提供されている構成ユーティリティーを 使用して WebSphere Applications エージェント用のデータ・コレクターを構成できない場合は、WebSphere 管理コンソールでデータ・コレクターを手動で構成します。説明については、856 ページの『構成ユーティリティーが 失敗した場合のデータ・コレクターの手動構成』を参照してください。

- (エージェント共存) WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications の両方がインストールされているエージェント共存環境で機能するようにデータ・コレクターを構成する場合は、860 ページの『(エージェント共存) WebSphere Applications エージェントおよびデータ・コレクターの構成』を参照してください。
- (Docker のモニター) Docker コンテナー内で実行される WebSphere Application Server Liberty をモニターするには、868 ページの『Docker コンテナー内の WebSphere Application Server Liberty のモニター』を参照してください。

ファスト・トラック: WebSphere Applications エージェント用のデータ・コレクターの構成
WebSphere Applications エージェントでは、エージェントのインストール後に構成は不要です。ただし、WebSphere 環境用のモニターをセットアップするために、エージェントのコンポーネントであるデータ・コレクターを構成する必要があります。

始める前に

1. モニターするアプリケーション・サーバーがインストールされていて実行されているシステムに WebSphere Applications エージェントをインストールします。
2. ユーザー・アクセス権限の要件を確認します。
   - Windows アプリケーション・サーバーのインストールに使用する管理者 ID を使用して、データ・コレクターを構成します。このユーザー ID が、データ・コレクター・ホーム・ディレクトリー *install_dir*¥dchome¥7.3.0.14.08 に対するフル書き込み権限があることを確認します。
   - Linux AIX アプリケーション・サーバーのインストールに使用するユーザー ID を使用して、データ・コレクターを構成します。このユーザー ID が、*install_dir*/yndchome/7.3.0.14.08 内の以下のサブディレクトリーに対する読み取りおよび書き込み許可を備えていることを確認します。
     - bin
     - data
     - runtime

このタスクについて

この手順では、簡易構成ユーティリティー simpleconfig を使用して、データ・コレクターの基本構成を指定します。

simpleconfig ユーティリティーは、デフォルト設定を使用してデータ・コレクターを構成します。追加のカスタマイズ・オプションを設定してデータ・コレクターを構成するには、同じディレクトリー内のフル構成ユーティリティー config を使用します。説明については、840 ページの『フル構成ユーティリティーを使用したデータ・コレクターの構成または再構成』を参照してください。

ほとんどの場合、simpleconfig ユーティリティーで十分です。複雑な環境の場合は、config 構成ユーティリティーを使用して、データ・コレクターを構成できます。simpleconfig ユーティリティーが失敗した場合は、代わりに config ユーティリティーを使用します。

手順

1. アプリケーション・サーバーのインストールに使用するユーザー ID でシステムにログインします。
2. データ・コレクター・ホーム・ディレクトリー内の bin ディレクトリーに変更します。
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
3. 以下の簡易構成ユーティリティーを実行します。
   - Windows **simpleconfig.bat**
   - Linux AIX **./simpleconfig.sh**

4. プロンプトに従って、データ・コレクターの構成を続行します。

   アプリケーション・サーバーの設定に応じて、以下の操作の一部またはすべてを実行する必要があります。

   - 従来の WebSphere Application Server の場合:
     - 自動的にディスカバーされた WebSphere インストール・ディレクトリーを選択するか、インストール・ディレクトリーを手動で指定します。
     - モニターする WebSphere Application Server プロファイルを選択します。
     - 使用するセキュリティー・プロパティー・プロファイルを選択するか、WebSphere 管理コンソールのユーザー名とパスワードを指定します (アプリケーション・サーバーについてセキュリティーが有効である場合)。
   - WebSphere Application Server Liberty の場合:
     - `bin` ディレクトリーおよび `servers` ディレクトリーを含む Liberty ホーム・ディレクトリーの絶対パスを指定します。例えば、`/opt/ibm/wlp` です。
     - Liberty によって使用される JRE のホーム・ディレクトリーを指定します。
5. データ・コレクターの構成が完了した後に、アプリケーション・サーバーを再始動します。
   a) アプリケーション・サーバー・プロファイルのホーム・ディレクトリー内の `bin` ディレクトリーに移動します。例えば、`opt/IBM/WebSphere/AppServer/profiles/profile_name/bin` です。
   b) コマンド・コンソールで **stopServer** コマンドを入力して、アプリケーション・サーバーを停止します。
      - Linux AIX `./stopServer.sh server_name`
      - Windows `stopServer.bat server_name`
   c) プロンプトが出されたら、WebSphere 管理コンソールの管理者のユーザー ID とパスワードを入力します。
   d) コマンド・コンソールで **startServer** コマンドを入力して、アプリケーション・サーバーを再び始動します。
      - Linux AIX `./startServer.sh server_name`
      - Windows `startServer.bat server_name`
6. Cloud APM コンソールにログインして、ダッシュボードにデータを表示します。
   a) サービスが使用可能になったことを知らせる E メールに記載されているリンクを使用して、コンソールにアクセスします。あるいは、IBM Marketplace Web サイトからコンソールにアクセスします。説明については、977 ページの『Cloud APM コンソールの始動』を参照してください。
   b) アプリケーション・エディターを使用して、モニター対象アプリケーション・サーバーを Application Performance ダッシュボードに追加します。新規コンポーネントとして既存のアプリケーションに追加するか、アプリケーションを作成してこのコンポーネントを含めることができます。

      アプリケーション・エディターについて詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

タスクの結果

アプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターが構成されます。データ収集によってアプリケーション・サーバーのオーバーヘッドが増加する可能性があることに注意してください。チューニングのための高度な構成オプションを使用して、データ収集を制御できます。

ユーザー・アクセス権限の要件の確認

WebSphere Applications エージェントには、データ・コレクターを構成するユーザー ID のユーザー・アクセス権限の要件があります。

このタスクについて

アプリケーション・サーバー・インストール ID に対して適切な許可を付与した後に、アプリケーション・サーバーのインストールに使用する ID を使用してデータ・コレクターを構成します。

手順

- Windows アプリケーション・サーバーのインストールに使用する管理者 ID を使用して、データ・コレクターを構成します。このユーザー ID が、データ・コレクター・ホーム・ディレクトリー *install_dir*¥dchome¥7.3.0.14.08 に対するフル書き込み権限があることを確認します。
- Linux AIX アプリケーション・サーバーのインストールに使用するユーザー ID を使用して、データ・コレクターを構成します。このユーザー ID が、*install_dir*/yndchome/7.3.0.14.08 内の以下のサブディレクトリーに対する読み取りおよび書き込み許可を備えていることを確認します。
  - bin
  - data
  - logs
  - runtime

  要確認 : 別のユーザー ID を使用してアプリケーション・サーバーをインストールした場合、別のユーザー ID を使用してデータ・コレクターを構成する必要が生じることがあります。初めてデータ・コレクターを構成した後には、別のユーザー ID を使用してデータ・コレクターを構成するたびに、以下のファイルに対する書き込み許可を付与します (ここで、*profile_name* は、アプリケーション・サーバーのプロファイル名です)。
  - *install_dir*/yndchome/7.3.0.14.08/data/findservers.inputlist
  - *install_dir*/yndchome/7.3.0.14.08/data/*profile_name*.findservers.progress
  - *install_dir*/yndchome/7.3.0.14.08/data/config_inputlist
  - *install_dir*/yndchome/7.3.0.14.08/runtime/custom/connections.properties

アプリケーション・サーバー内の他の既存のデータ・コレクターの処理

アプリケーション・サーバー内にデータ・コレクターが既に存在する場合は、WebSphere Applications エージェントのデータ・コレクターと競合しないように、既存のデータ・コレクターの処理を決定する必要があります。

このタスクについて

モニター対象のアプリケーション・サーバー内に以下のタイプのデータ・コレクターが既に存在する可能性があります。

- 以前のバージョンの IBM Cloud Application Performance Management でインストールされた WebSphere Applications エージェントのデータ・コレクター
- 古い IBM® Tivoli® Monitoring インフラストラクチャーでインストールされた ITCAM Agent for WebSphere Applications のデータ・コレクター
- IBM が提供していない他のデータ・コレクター

手順

データ・コレクターの競合を回避するために、適切なアクションを取ります。

- 以前のバージョンの IBM Cloud Application Performance Management でインストールされた以前のバージョンの WebSphere Applications エージェントのデータ・コレクターについては、以下のオプションがあります。

  - 最新のデータ・コレクターのホーム・ディレクトリーからマイグレーション・ユーティリティーを使用して、データ・コレクターをマイグレーションします。説明については、1147 ページの『WebSphere Applications エージェント: データ・コレクターのマイグレーション』を参照してください。
  - 以前のバージョンのデータ・コレクターを構成解除し、最新のデータ・コレクターのホーム・ディレクトリーから構成ユーティリティーを使用して、データ・コレクターを再構成します。データ・コレクターの構成解除については、152 ページの『WebSphere Applications エージェント: データ・コレクターの構成解除』を参照してください。
- ITCAM Agent for WebSphere Applications のデータ・コレクターについては、エージェント共存環境内にモニターをデプロイする場合は、以下のステップを実行します。

  a) ITCAM Agent for WebSphere Applications のデータ・コレクターをアンインストールします。

  b) WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications の両方にデータを送信するためにデータ・コレクターを 1 つのみ構成します。説明については、860 ページの『(エージェント共存) WebSphere Applications エージェントおよびデータ・コレクターの構成』を参照してください。
- IBM が提供していない他のデータ・コレクターについては、それらのデータ・コレクターを削除する必要があるかどうか調べてください。WebSphere Applications エージェントのデータ・コレクターは Java バイトコード操作を使用してデータを収集します。同じ方法を使用してデータを収集する他のデータ・コレクターは、WebSphere Applications エージェントのデータ・コレクターと競合する可能性があります。

簡易構成ユーティリティーを使用したデータ・コレクターの構成

WebSphere Applications エージェントはインストール後に自動的に開始されますが、アプリケーション・サーバー・インスタンスをモニターするためには、エージェントのコンポーネントであるデータ・コレクターを手動で構成する必要があります。

始める前に

- ご使用の環境でユーザー・アクセス要件が満たされていることを確認します。説明については、837 ページの『ユーザー・アクセス権限の要件の確認』を参照してください。
- モニター対象のアプリケーション・サーバー内に他のデータ・コレクターが存在する場合は、データ・コレクターの競合を回避するために、適切なアクションを取ります。説明については、837 ページの『アプリケーション・サーバー内の他の既存のデータ・コレクターの処理』を参照してください。

このタスクについて

重要 :

- データ・コレクターをリソースのモニター用にのみ構成する場合、または追加オプションを設定する場合は、フル構成手順を使用します。説明については、840 ページの『フル構成ユーティリティーを使用したデータ・コレクターの構成または再構成』を参照してください。
- モニター・ユーザー・インターフェース 内でのサーバーの名前を変更する場合は、データ・コレクターを再構成し、サーバー別名を指定します。説明については、840 ページの『フル構成ユーティリティーを使用したデータ・コレクターの構成または再構成』を参照してください。

WebSphere Applications エージェントの場合、*dc_home* 変数は、データ・コレクターのホーム・ディレクトリーを参照します。各オペレーティング・システムでの *dc_home* 変数の場所は以下のとおりです。

- Windows *install_dir*¥dchome¥7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

手順

1. アプリケーション・サーバーのインストールに使用するユーザー ID でシステムにログインします。
2. データ・コレクター・ホーム・ディレクトリー内の bin ディレクトリーに変更します。
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥bin

- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin

3. 以下の簡易構成ユーティリティーを実行します。
   - Windows simpleconfig.bat
   - Linux AIX ./simpleconfig.sh

   simpleconfig ユーティリティーでは、アプリケーション・サーバーのホーム・ディレクトリーが自動的にディスカバーされます。
4. プロンプトに従って、データ・コレクターの構成を続行します。

   アプリケーション・サーバーの設定に応じて、以下の操作を実行する必要があります。
   - 従来の WebSphere Application Server の場合:
     - 自動的にディスカバーされた WebSphere インストール・ディレクトリーを選択するか、インストール・ディレクトリーを手動で指定します。
     - モニターする WebSphere Application Server プロファイルを選択します。
     - 使用するセキュリティー・プロパティー・プロファイルを選択するか、WebSphere 管理コンソールのユーザー名とパスワードを指定します (アプリケーション・サーバーについてセキュリティーが有効である場合)。
   - WebSphere Application Server Liberty の場合:
     - bin ディレクトリーおよび servers ディレクトリーを含む Liberty ホーム・ディレクトリーの絶対パス (/opt/ibm/wlp など) を指定します。
     - Liberty によって使用される JRE のホーム・ディレクトリーを指定します。
5. 可能であれば、データ・コレクターの構成が完了した後に、アプリケーション・サーバー・インスタンスを再始動します。
   a) アプリケーション・サーバー・プロファイルのホーム・ディレクトリー内の bin ディレクトリーに移動します。例えば、opt/IBM/WebSphere/AppServer/profiles/*profile_name*/bin です。
   b) コマンド・コンソールで **stopServer** コマンドを入力して、アプリケーション・サーバーを停止します。
      - Linux AIX ./stopServer.sh *server_name*
      - Windows stopServer.bat *server_name*

   c) プロンプトが出されたら、WebSphere 管理コンソールの管理者のユーザー ID とパスワードを入力します。
   d) コマンド・コンソールで **startServer** コマンドを入力して、アプリケーション・サーバーを再び始動します。
      - Linux AIX ./startServer.sh *server_name*
      - Windows startServer.bat *server_name*

タスクの結果

- データ・コレクターがプロファイルのすべてのインスタンス (WebSphere Application Server Liberty の場合は単一のインスタンスまたは同じディレクトリーにある複数のインスタンス) をモニターするように構成されます。追加のプロファイルまたはインスタンスをモニターするには、この構成を繰り返します。
- データ・コレクターはサーバー・インスタンス内で構成され、最大限のモニターを行います。
- Cloud APM, Base の場合は、リソース・モニターが有効になります。
- Cloud APM, Advanced の場合は、リソース・モニター、トランザクション・トラッキング、および診断のデータが有効になります。

既知の制限 : WebSphere Application Server Liberty をモニターしている場合、データ・コレクターは Java Naming and Directory Interface (JNDI) イベントを生成できません。

次のタスク

- Cloud APM コンソールにログインし、アプリケーション・エディターを使用してモニター対象アプリケーション・サーバーを Application Performance ダッシュボードに追加します。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

  要確認 : WebSphere Portal Server をモニターするように WebSphere Applications エージェントが構成されている場合、このエージェントは、WebSphere Application Server ではなく、Application Performance ダッシュボード上の WebSphere Portal Server アプリケーション・コンポーネントに関連付けられています。
- Application Performance ダッシュボード上のモニター・ユーザー・インターフェースにアプリケーション・サーバー・インスタンスの情報が何も表示されない場合は、以下の手順に従って、WebSphere Applications エージェントのモニター・エージェント・コンポーネントを再始動します。
  - Windows

    ```
    cd install_dir¥bin
    was-agent.bat stop
    was-agent.bat start
    ```
  - Linux AIX

    ```
    cd install_dir/bin
    ./was-agent.sh stop
    ./was-agent.sh start
    ```

フル構成ユーティリティーを使用したデータ・コレクターの構成または再構成

追加の構成オプションを設定するには、簡易構成ユーティリティーの代わりにフル構成ユーティリティー (対話式またはサイレント) を使用して、データ・コレクターを構成できます。データ・コレクターが既に構成されていて、その再構成を行う場合にも、フル構成ユーティリティーを使用できます。WebSphere Portal Server インスタンスのモニターを構成する場合にも、フル構成ユーティリティーを使用する必要があります。

始める前に

- ご使用の環境でユーザー・アクセス要件が満たされていることを確認します。説明については、837 ページの『ユーザー・アクセス権限の要件の確認』を参照してください。
- モニター対象のアプリケーション・サーバー内に他のデータ・コレクターが存在する場合は、データ・コレクターの競合を回避するために、適切なアクションを取ります。説明については、837 ページの『アプリケーション・サーバー内の他の既存のデータ・コレクターの処理』を参照してください。

このタスクについて

構成ユーティリティーおよび再構成ユーティリティーは、以下のディレクトリーにあります。

- Windows *install_dir*¥dchome¥7.3.0.14.08¥bin
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin

手順

- 構成ユーティリティーの名前は **config** です。以下の場合には、フル構成ユーティリティーを使用してデータ・コレクターを構成する必要があります。
  - **simpleconfig** 構成ユーティリティーが失敗する。
  - WebSphere Portal Server インスタンスのモニターを構成する必要がある。
  - データ・コレクターの構成中にモニター・ユーザー・インターフェースに表示されるサーバー別名を指定する必要がある。

- 収集するデータの種類を詳細に制御する必要がある。例えば、リソース・モニターのみを使用し、診断データおよびトランザクション・トラッキングを無効にする必要がある。
- 一度に同じプロファイル内のすべてのアプリケーション・サーバーを構成しないようにする必要がある。
- データ・コレクターがアプリケーション・サーバー内に構成されていないため、データ・コレクターを再構成する必要がある。

対話式フル構成ユーティリティーについては、841 ページの『対話式でのデータ・コレクターの構成』を参照してください。

- 再構成ユーティリティーの 名前は **reconfig** です。以下の場合には、データ・コレクターを再構成する必要があります。
  - データ・コレクターが対話式またはサイレントに構成された後に、再構成する必要がある。

  対話式再構成ユーティリティーについては、846 ページの『対話式でのデータ・コレクターの再構成』を参照してください。
- サイレント構成については、849 ページの『サイレント・モードでのデータ・コレクターの構成』を参照してください。

対話式でのデータ・コレクターの構成

simpleconfig ユーティリティーが失敗した場合、データ・コレクターを構成するために、対話式構成ユーティリティー (config.sh または config.bat) を使用します。config.sh または config.bat ユーティリティーを使用して、モニターするアプリケーション・サーバー・インスタンスごとにデータ・コレクターを構成できます。

始める前に

WebSphere Application Server Liberty をモニターするようにデータ・コレクターを構成する場合は、**JAVA_HOME** システム環境変数を、アプリケーション・サーバー用に使用されているのと同じ JVM に設定します。例えば、Windows システムでは、**JAVA_HOME** の値を C:¥Program Files¥IBM¥java に設定します。Linux システムでは、export JAVA_HOME=/opt/IBM/java を実行します。

このタスクについて

以下のフル構成ユーティリティーを 使用して、データ・コレクターを構成します。

- Windows *install_dir*¥dchome¥7.3.0.14.08¥bin¥config.bat
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin/config.sh

手順

プロンプトに応答を入力してデータ・コレクターを構成する場合は、以下のステップを実行します。

1. アプリケーション・サーバーのインストールに使用するユーザー ID でシステムにログインします。
2. *dc_home* データ・コレクター・ホーム・ディレクトリー内の bin ディレクトリーに移動します。
3. 以下のコマンドを発行して、構成ユーティリティーを 開始します。
   - Windows config.bat
   - Linux AIX ./config.sh

   構成ユーティリティーに、ローカル・コンピューター・システム上で検出されたすべてのネットワーク・カードの IP アドレスおよびホスト名が表示されます。
4. IP アドレスおよびホスト名に対応する数値を入力します。使用する IP アドレスおよびホスト名がリスト上にない場合は、IP アドレスまたはホスト名を入力します。
5. モニター対象のアプリケーション・サーバーのホーム・ディレクトリーを指定します。

- 従来の WebSphere Application Server の場合は、自動的にディスカバーされたアプリケーション・サーバーのホーム・ディレクトリーに対応する番号を入力するか、アプリケーション・サーバーのホーム・ディレクトリーへの絶対パスを指定します。
- WebSphere Application Server Liberty の場合は、`bin` ディレクトリーおよび `servers` ディレクトリーを含む WebSphere Application Server Liberty ホーム・ディレクトリーへの絶対パス (`/opt/ibm/wlp` など) を入力します。

6. WebSphere Application Server Liberty 用のデータ・コレクターを構成している場合は、Java ホーム・ディレクトリーを求めるプロンプトが出されます。アプリケーション・サーバー用に使用されている Java ホーム・ディレクトリーを指定します。例えば、`/opt/IBM/java` です。
7. 指定されたアプリケーション・サーバーのホーム・ディレクトリーの下にあるすべてのプロファイルが構成ユーティリティーによってリストされたら、構成するアプリケーション・サーバー・プロファイルに対応する番号を入力します。
   - 従来の WebSphere Application Server の場合は、構成ユーティリティーは、指定した WebSphere Application Server プロファイルについて WebSphere グローバル・セキュリティーが有効かどうかを、以下のように示します。グローバル・セキュリティーが有効でない場合は、ステップ 842 ページの『9』 に進みます。
   - WebSphere Application Server Liberty の場合は、ステップ 842 ページの『10』 に進みます。
8. WebSphere Application Server プロファイルについてグローバル・セキュリティーが有効である場合は、クライアント・プロパティー・ファイルからセキュリティー設定を取得するかどうかを指定します。構成ユーティリティーが、該当するクライアント・プロパティー・ファイルからユーザー名とパスワードを取得できるようにするには 1 を入力します。そうしない場合は、2 を入力して、ユーザー名とパスワードを入力します。

   データ・コレクターはリモート・メソッド呼び出し (RMI) または SOAP プロトコルを使用して WebSphere Administrative Services と通信します。プロファイルでグローバル・セキュリティーが有効になっている場合は、そのアプリケーション・サーバー・プロファイルに対して WebSphere Application Server 管理コンソールにログインする権限があるユーザーのユーザー ID とパスワードを指定する必要があります。あるいは、ユーザー名とパスワードを暗号化して、それらをクライアント・プロパティー・ファイルに保存してから、データ・コレクターを構成することもできます。`sas.client.props` ファイル (RMI 接続の場合) または `soap.client.props` ファイル (SOAP 接続の場合) を使用する必要があります。
9. WebSphere 管理コンソールのホスト名を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、WebSphere 管理コンソールのホスト名または IP アドレスを指定します。デフォルト値は `localhost` です。

   要確認 : ネットワーク・デプロイメント環境の場合は、デプロイメント・マネージャーのホスト名または IP アドレスを入力します。
10. まだデータ収集用に構成されていないすべてのサーバー・インスタンスが構成ユーティリティーによってリストされたら、1 つ以上のアプリケーション・サーバー・インスタンスをリストから選択します。アプリケーション・サーバー・インスタンスに対応する数値を入力し、データ収集の対象としてそれらのインスタンスを構成するか、アスタリスク (*) を入力し、データ収集の対象としてすべてのアプリケーション・サーバー・インスタンスを構成します。 サーバーのサブセットを指定する場合は、サーバーを表すそれぞれの番号をコンマで区切って入力します。
    例えば、`1,2,3` です。

    要確認 :
    - スタンドアロン環境の場合、構成中はアプリケーション・サーバー・インスタンスが実行中である必要があります。(WebSphere Application Server Liberty のインスタンスは稼働している必要はありません)。
    - ネットワーク・デプロイメント環境では、デプロイメント・マネージャーが稼働している必要があります。
    - 選択するアプリケーション・サーバー・インスタンスが、モニター対象のアプリケーションまたはサービスをホストする実際のサーバーであるようにします。
11. 「**Agent for WebSphere Applications との統合 (Integration with Agent for WebSphere Applications)**」セクションで、データ・コレクターを WebSphere Applications エージェントと統合す

ることを指定します。1 と入力してこの統合オプションを選択してから、Enter キーを押す必要があります。

選択したサーバーが、PMI リソース・モニター用に登録されます。

12. 従来の WebSphere Application Server 用のデータ・コレクターを構成している場合は、データ・コレクターをアプリケーション・サーバー・インスタンス内に構成するかどうかを指定します。
    - データ・コレクターをアプリケーション・サーバー内に構成するには、1 を入力します。すべての運用モニターおよび診断データ収集を利用するには、このオプションを使用して、データ・コレクターをアプリケーション・サーバーと統合する必要があります。ただし、データ・コレクターをアプリケーション・サーバー内に構成すると、アプリケーション・サーバーの再始動が必要になります。 また、データ・コレクターはサーバーのパフォーマンスに影響を与える可能性があります。
    - データ・コレクターをアプリケーション・サーバー内に構成しないようにするには、2 を入力し、ステップ 843 ページの『14』 に進みます。このオプションを使用すると、データ・コレクターはスタンドアロン・プロセスとして実行され、リソース・モニターのみを有効にすることができます。
13. プロンプトが出されたら、診断データ用にデータ・コレクターを有効にするかどうかを指定します。診断データ収集を有効にするには、1 を入力します。デフォルトは 2 です。
14. V8 モニター・エージェントのホスト名を求めるプロンプトが出されたら、WebSphere Applications エージェントのホスト名または IP アドレスを入力するか、Enter キーを押してデフォルトを受け入れます。デフォルト値は、ステップ 3 での選択内容に応じて異なります。

    V8 モニター・エージェントとは、IBM Cloud Application Performance Management でインストールされた WebSphere Applications エージェントのことです。
15. V8 モニター・エージェントのポート番号を求めるプロンプトが出されたら、WebSphere Applications エージェントのポート番号を入力するか、Enter キーを押してデフォルトを受け入れます。デフォルトは 63335 です。
16. WebSphere Applications 用の V6 モニター・エージェントを構成するかどうかを確認するメッセージが表示されたら、Enter キーを押してデフォルトの「いいえ」を受け入れます。

    V6 モニター・エージェントとは、古い IBM® Tivoli® Monitoring インフラストラクチャーでインストールされた ITCAM Agent for WebSphere Applications のことです。V6 モニター・エージェントの構成は、エージェント共存環境でのみ必要です。
17. サーバー別名を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、デフォルトとは異なる別名を入力します。複数のアプリケーション・サーバー・インスタンスを構成する場合、構成ユーティリティーは、インスタンスごとに別名を求めるプロンプトを出します。

    重要 : 別名に使用できる文字は、A から Z、a から z、下線 (_)、ダッシュ (-)、およびピリオド (.) のみです。これ以外の文字を別名に使用しないでください。

    サーバー別名は、Cloud APM コンソールに表示されるエージェント・インスタンス名 (MSN とも呼ばれます) の最初の修飾子です。デフォルトは、サーバー名にそのノード名を結合したものになります。例えば、別名 **node1server1** は、**node1** というノード内の **server1** というサーバーを示します。
18. PMI リソース・モニター用のポート番号を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、新しい番号を入力します。デフォルトのポートは 63355 です。

    このポートは、同じホストで稼働しているコンポーネント間の内部通信に使用されます。デフォルトが使用中の場合は、別の番号を設定できます。
19. 「**トランザクション・トラッキングのサポート**」セクションで、トランザクション・トラッキングを有効にするかどうかを指定します。トランザクション・トラッキングのサポートを有効にするには、1 を入力します。そうしない場合は、2 を入力して、ステップ 844 ページの『22』 へスキップします。
20. Transaction Framework Extension のホスト名または IP アドレスを求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、別のホスト名または IP アドレスを入力します。

    Transaction Framework Extension は、データ・コレクターからメトリックを収集する、WebSphere Applications エージェントの内部コンポーネントです。

21. データ・コレクターが Transaction Framework Extension への接続に使用するポート番号を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、別のポート番号を入力します。デフォルトは 5457 です。
22. データ・コレクターを Application Performance Diagnostics Lite と統合するかどうかを指定します。Enter キーを押してデフォルトの「いいえ」を受け入れます。
23. 「**拡張設定**」セクションで、ガーベッジ・コレクション・ログのパスを変更するかどうかを指定します。ガーベッジ・コレクション・ログのパスを選択する場合は、1 を入力します。 そうしない場合は、2 を入力して、ステップ 844 ページの『25』 へスキップします。アプリケーション・サーバーの JVM 引数で既に指定されているログ・パスを使用するには、2 を入力します。
24. ガーベッジ・コレクション・ログのパスを指定します。絶対パスを使用して、ファイル名を入力します。 WebSphere Application Server Liberty の場合、パスに変数は使用しないでください。データ・コレクターは、サーバー・インスタンス情報を追加することで、ログ・ファイル名を自動的に変更します。

    例えば、ファイル名に `gc.log` を指定すると、構成されるアプリケーション・サーバー・インスタンスごとに、実際の名前は *`profile_name`*`.`*`cell_name`*`.`*`node_name`*`.`*`server_name`*`.gc.log` に設定されます。

    重要 : ガーベッジ・コレクション・ログのパスには、`${SERVER_LOG_ROOT}` などの WebSphere 変数を使用できます。 ただし、`%pid` などのテンプレートは使用しないでください。
25. 指定されたアプリケーション・サーバー・インスタンスに適用されるデータ・コレクター構成の要約を確認します。必要であれば、変更を適用する前にデータ・コレクター構成を部分的に再構成します。
26. 変更を受け入れるには、a を入力します。
27. プロンプトが出されたら、現在の構成のバックアップを作成するかどうかを指定します。現在の構成のバックアップを作成するには、1 を入力します。 そうしない場合は、2 を入力します。

    構成ユーティリティーによって 変更が適用され、プロファイルに対するデータ・コレクターの構成が完了したことを示す状況メッセージが表示されます。
28. 従来の WebSphere Application Server 用のデータ・コレクターを構成している場合は、ステップ 843 ページの『12』 での選択に応じて、アプリケーション・サーバー・インスタンスを再始動するか、エージェントを再始動します。

    - アプリケーション・サーバー内のデータ・コレクターを有効にした場合は、構成ユーティリティーの指示に従って、アプリケーション・サーバー・インスタンスを再始動します。
    - アプリケーション・サーバー内のデータ・コレクターを有効にせずに PMI リソース・モニターを有効にした場合は、以下のコマンドを実行して、WebSphere Applications エージェントを再始動します。
        - Windows

          ```
          cd install_dir¥bin
          was-agent.bat stop
          was-agent.bat start
          ```
        - Linux AIX

          ```
          cd install_dir/bin
          ./was-agent.sh stop
          ./was-agent.sh start
          ```

    データ・コレクターの構成は、アプリケーション・サーバーまたはエージェントが再始動されると有効になります。
29. Cloud APM コンソールにログインして、ダッシュボードにデータを表示します。
    a) サービスが使用可能になったことを知らせる E メールに記載されているリンクを使用して、コンソールにアクセスします。あるいは、IBM Marketplace Web サイトからコンソールにアクセスします。説明については、977 ページの『Cloud APM コンソールの始動』を参照してください。

b) アプリケーション・エディターを使用して、モニター対象アプリケーション・サーバーを Application Performance ダッシュボードに追加します。新規コンポーネントとして既存のアプリケーションに追加するか、アプリケーションを作成してこのコンポーネントを含めることができます。

アプリケーション・エディターについて詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

次のタスク

- データ・コレクターの構成に使用されている現行ユーザー ID が、アプリケーション・サーバーを実行しているユーザーの ID と同じでない場合は、データ・コレクターの構成用のユーザー ID が、データ・コレクター・ホーム・ディレクトリー内の `runtime` および `logs` ディレクトリーに対する読み取り許可と書き込み許可を備えていることを確認してください。これらの 2 つのサブディレクトリーは、サーバーの再始動時にアプリケーション・サーバーを実行しているユーザーの ID によって作成されます。
- Cloud APM コンソールにログインして、ダッシュボードにモニター・データを表示します。モニター・データがすぐに使用できない場合は、以下のコマンドを実行して、WebSphere Applications エージェントを再始動します。
  - Windows

    ```
    cd install_dir¥bin
    was-agent.bat stop
    was-agent.bat start
    ```

  - Linux AIX

    ```
    cd install_dir/bin
    ./was-agent.sh stop
    ./was-agent.sh start
    ```

- サーバー別名を変更すると、Cloud APM コンソールに登録されているエージェント・インスタンス名が変更されます。今回がデータ・コレクターの初めての構成ではなく、サーバー別名を変更した場合、以下のステップを実行して、一部のキャッシュ・ファイルをクリアする必要があります。
  1. モニター・エージェントが実行されている場合は停止します。
  2. テキスト・エディターで、以下のディレクトリー内の *hostname*_yn.xml ファイルを開きます。ここで、*hostname* は、WebSphere Applications エージェントがインストールされているホストの名前です。
     - Windows *install_dir*¥TMAITM6_x64 (デフォルトは、 C:¥IBM¥APM¥TMAITM6_x64 です)
     - Linux AIX *install_dir*/config (デフォルトは、 /opt/ibm/apm/agent/config です)
  3. 以下のストリングで始まり、以前のサーバー名が含まれている行を見つけます。例えば、was85.win4net01Cell02.win4net01Node02.AppSrv01.server1 (server1 はアプリケーション・サーバーの以前の名前)。

     ```
     <!ENTITY was_product_code.cellname.nodename.profilename.servername
     ```

     ここで、*was_product_code* は WebSphere Application Server の製品コードであり、*cellname* はセルの名前であり、*nodename* はノード名であり、*profilename* はアプリケーション・サーバー・プロファイルの名前であり、*servername* はアプリケーション・サーバーの以前の名前です。
  4. 行で示されている .XML ファイルを現行ディレクトリー内で見つけ、そのファイルを削除します。
  5. ステップ 3 で見つけた行を *hostname*_yn.xml ファイルから削除します。
  6. *hostname*_yn.xml ファイルの末尾にある、以前のサーバー名が含まれている行を削除します。
  7. ファイルを保存して閉じます。
  8. モニター・エージェントを再始動します。

対話式でのデータ・コレクターの再構成

1 つ以上のアプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターを構成した場合は、再構成ユーティリティー (reconfig.sh または reconfig.bat) を使用して、データ・コレクターを再構成できます。

始める前に

WebSphere Application Server Liberty をモニターするようにデータ・コレクターを構成する場合は、**JAVA_HOME** システム環境変数を、アプリケーション・サーバー用に使用されているのと同じ JVM に設定します。例えば、Windows システムでは、**JAVA_HOME** の値を C:¥Program Files¥IBM¥java に設定します。Linux システムでは、export JAVA_HOME=/opt/IBM/java を実行します。

このタスクについて

以下のフル再構成ユーティリティーを 使用して、データ・コレクターを構成します。

- Windows *install_dir*¥dchome¥7.3.0.14.08¥bin¥reconfig.bat
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin/reconfig.sh

要確認 : reconfig ユーティリティーは、以下の場合には適用されません。代わりに **config** 構成ユーティリティーを使用します。**config** ユーティリティーは、サーバーが既に構成されていることを警告しますが、すべての必要な変更を行うことができます。

- データ・コレクターがリソースのモニター用にのみ既に構成されているため、データ・コレクターを再構成する必要がある。
- WebSphere Portal Server 用のデータ・コレクターを再構成する必要がある。

ヒント : この再構成ユーティリティーは、エージェント構成設定を尋ねるプロンプトで、現在構成されている値をデフォルトとして提供します。

手順

プロンプトに応答を入力してデータ・コレクターを再構成する場合は、以下のステップを実行します。

1. アプリケーション・サーバーのインストールに使用するユーザー ID でシステムにログインします。
2. *dc_home* データ・コレクター・ホーム・ディレクトリー内の bin ディレクトリーに移動します。
3. 以下のコマンドを発行して、再構成ユーティリティーを 開始します。
   - Windows reconfig.bat
   - Linux AIX ./reconfig.sh

   ヒント : この再構成ユーティリティーを 実行した場合の効果は、-reconfig 引数を指定して config.bat スクリプトを実行した場合 (Windows システムでの方法。Linux または AIX システムでは、-reconfig 引数を指定して config.sh スクリプトを実行した場合) と同じです。

   再構成ユーティリティーに、ローカル・コンピューター・システム 上で検出されたすべてのネットワーク・カードの IP アドレスが表示されます。
4. 使用する IP アドレスに対応する数値を入力します。

   再構成ユーティリティーは、このホスト上でデータ・コレクターが構成されているすべてのアプリケーション・サーバー・インスタンスを表示し、1 つ以上のアプリケーション・サーバー・インスタンスをリストから選択するようプロンプトを出します。
5. リストから 1 つ以上のアプリケーション・サーバー・インスタンスを選択します。データ収集に向けて再構成するアプリケーション・サーバー・インスタンスに対応する数値を入力するか、すべてのアプリケーション・サーバー・インスタンスをデータ収集に向けて再構成するためのアスタリスク (*) を入力します。 サーバーのサブセットを指定するには、当該のサーバーを表す番号をコンマで区切って入力します。 例えば、1,2,3 です。

   要確認 :

- スタンドアロン環境の場合、構成中はアプリケーション・サーバー・インスタンスが実行中である必要があります。(WebSphere Application Server Liberty のインスタンスは稼働している必要はありません)。
- ネットワーク・デプロイメント環境では、デプロイメント・マネージャーが稼働している必要があります。
- 選択するアプリケーション・サーバー・インスタンスが、モニター対象のアプリケーションまたはサービスをホストする実際のサーバーであるようにします。

6. 「**Agent for WebSphere Applications との統合 (Integration with Agent for WebSphere Applications)**」セクションで、データ・コレクターを WebSphere Applications エージェントと統合することを指定します。1 と入力してこの統合オプションを選択してから、Enter キーを押す必要があります。
7. 従来の WebSphere Application Server 用のデータ・コレクターを構成している場合は、データ・コレクターをアプリケーション・サーバー・インスタンス内に構成するかどうかを指定します。
   - データ・コレクターをアプリケーション・サーバー内に構成するには、1 を入力します。すべての運用モニターおよび診断データ収集を利用するには、このオプションを使用して、データ・コレクターをアプリケーション・サーバーと統合する必要があります。ただし、データ・コレクターをアプリケーション・サーバー内に構成すると、アプリケーション・サーバーの再始動が必要になります。 また、データ・コレクターはサーバーのパフォーマンスに影響を与える可能性があります。
   - データ・コレクターをアプリケーション・サーバー内に構成しないようにするには、2 を入力し、ステップ 847 ページの『9』 に進みます。このオプションを使用すると、データ・コレクターはスタンドアロン・プロセスとして実行され、PMI リソース・モニターのみを有効にすることができます。
8. プロンプトが出されたら、データ・コレクターの診断データ収集を有効にするかどうかを指定します。「はい」の場合には 1、「いいえ」の場合には 2 を入力します。
9. ホスト名を求めるプロンプトが出されたら、WebSphere Applications エージェントのホスト名または IP アドレスを入力するか、Enter キーを押してデフォルトを受け入れます。デフォルト値は、ステップ 846 ページの『4』 での選択内容に応じて異なります。
10. ポート番号を求めるプロンプトが出されたら、モニター・エージェントのポート番号を入力するか、Enter キーを押してデフォルトを受け入れます。デフォルトは 63335 です。
11. WebSphere Applications 用の V6 モニター・エージェントを構成するかどうかを確認するメッセージが表示されたら、Enter キーを押してデフォルトの「いいえ」を受け入れます。

    V6 モニター・エージェントとは、古い IBM® Tivoli® Monitoring インフラストラクチャーでインストールされた ITCAM Agent for WebSphere Applications のことです。V6 モニター・エージェントの構成は、エージェント共存環境でのみ必要です。
12. サーバー別名を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、デフォルトとは異なる別名を入力します。複数のアプリケーション・サーバー・インスタンスを構成する場合、構成ユーティリティーは、インスタンスごとに別名を求めるプロンプトを出します。

    重要 : 別名に使用できる文字は、A から Z、a から z、下線 (_)、ダッシュ (-)、およびピリオド (.) のみです。これ以外の文字を別名に使用しないでください。

    サーバー別名は、Cloud APM コンソールに表示されるエージェント・インスタンス名 (MSN とも呼ばれます) の最初の修飾子です。デフォルトは、サーバー名にそのノード名を結合したものになります。例えば、別名 **node1server1** は、**node1** というノード内の **server1** というサーバーを示します。
13. PMI リソース・モニター用のポート番号を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、新しい番号を入力します。デフォルトのポートは 63355 です。

    このポートは、同じホストで稼働しているコンポーネント間の内部通信に使用されます。デフォルトが使用中の場合は、別の番号を設定できます。
14. 「**トランザクション・トラッキングのサポート**」セクションで、トランザクション・トラッキングを有効にするかどうかを指定します。トランザクション・トラッキングのサポートを有効にするには、1 を入力します。そうしない場合は、2 を入力して、ステップ 848 ページの『17』 へスキップします。
15. Transaction Framework Extension のホスト名または IP アドレスを求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、別のホスト名または IP アドレスを入力します。

Transaction Framework Extension は、データ・コレクターからメトリックを収集する、WebSphere Applications エージェントの内部コンポーネントです。

16. データ・コレクターが Transaction Framework Extension への接続に使用するポート番号を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、別のポート番号を入力します。デフォルトは 5457 です。
17. データ・コレクターを Application Performance Diagnostics Lite と統合するかどうかを指定します。Enter キーを押してデフォルトの「いいえ」を受け入れます。
18. 「**拡張設定**」セクションで、ガーベッジ・コレクション・ログのパスを変更するかどうかを指定します。

    ガーベッジ・コレクション・ログのパスを選択する場合は、1 を入力します。 そうしない場合は、2 を入力して、ステップ 848 ページの『20』 へスキップします。アプリケーション・サーバーの JVM 引数で既に指定されているログ・パスを使用するには、2 を入力します。
19. ガーベッジ・コレクション・ログのパスを指定します。絶対パスを使用して、ファイル名を入力します。 WebSphere Application Server Liberty の場合、パスに変数は使用しないでください。データ・コレクターは、サーバー・インスタンス情報を追加することで、ログ・ファイル名を自動的に変更します。

    例えば、ファイル名に gc.log を指定すると、構成されるアプリケーション・サーバー・インスタンスごとに、実際の名前は *profile_name*.*cell_name*.*node_name*.*server_name*.gc.log に設定されます。

    重要 : ガーベッジ・コレクション・ログのパスには、${SERVER_LOG_ROOT} などの WebSphere 変数を使用できます。 ただし、%pid などのテンプレートは使用しないでください。
20. 指定されたアプリケーション・サーバー・インスタンスに適用されるデータ・コレクター構成の要約を確認します。必要であれば、変更を適用する前にデータ・コレクター構成を部分的に再構成します。
21. 変更を受け入れるには、a を入力します。
22. プロンプトが出されたら、現在の構成のバックアップを作成するかどうかを指定します。現在の構成のバックアップを作成するには、1 を入力します。 そうしない場合は、2 を入力します。

    構成ユーティリティーによって 変更が適用され、プロファイルに対するデータ・コレクターの構成が完了したことを示す状況メッセージが表示されます。
23. 従来の WebSphere Application Server 用のデータ・コレクターを構成している場合は、ステップ 847 ページの『7』 での選択に応じて、アプリケーション・サーバー・インスタンスを再始動するか、エージェントを再始動します。

    - アプリケーション・サーバー内のデータ・コレクターを有効にした場合は、構成ユーティリティーの指示に従って、アプリケーション・サーバー・インスタンスを再始動します。
    - アプリケーション・サーバー内のデータ・コレクターを有効にせずに PMI リソース・モニターを有効にした場合は、以下のコマンドを実行して、WebSphere Applications エージェントを再始動します。

      – Windows

      ```
      cd install_dir¥bin
      was-agent.bat stop
      was-agent.bat start
      ```

      – Linux AIX

      ```
      cd install_dir/bin
      ./was-agent.sh stop
      ./was-agent.sh start
      ```

    データ・コレクターの構成は、アプリケーション・サーバーまたはエージェントが再始動されると有効になります。

次のタスク

- サーバー別名を変更すると、Cloud APM コンソールに登録されているエージェント・インスタンス名が変更されます。再構成手順でサーバー別名を変更した場合、以下のステップを実行して、一部のキャッシュ・ファイルをクリアする必要があります。

  1. モニター・エージェントが実行されている場合は停止します。
  2. テキスト・エディターで、以下のディレクトリー内の *hostname*_yn.xml ファイルを開きます。ここで、*hostname* は、WebSphere Applications エージェントがインストールされているホストの名前です。
     - Windows *install_dir*¥TMAITM6_x64 (デフォルトは、 C:¥IBM¥APM¥TMAITM6_x64 です)
     - Linux AIX *install_dir*/config (デフォルトは、 /opt/ibm/apm/agent/config です)
  3. 以下のストリングで始まり、以前のサーバー名が含まれている行を見つけます。例えば、was85.win4net01Cell02.win4net01Node02.AppSrv01.server1 (server1 はアプリケーション・サーバーの以前の名前)。

     ```
     <!ENTITY was_product_code.cellname.nodename.profilename.servername
     ```

     ここで、*was_product_code* は WebSphere Application Server の製品コードであり、*cellname* はセルの名前であり、*nodename* はノード名であり、*profilename* はアプリケーション・サーバー・プロファイルの名前であり、*servername* はアプリケーション・サーバーの以前の名前です。
  4. 行で示されている .XML ファイルを現行ディレクトリー内で見つけ、そのファイルを削除します。
  5. ステップ 3 で見つけた行を *hostname*_yn.xml ファイルから削除します。
  6. *hostname*_yn.xml ファイルの末尾にある、以前のサーバー名が含まれている行を削除します。
  7. ファイルを保存して閉じます。
  8. モニター・エージェントを再始動します。

サイレント・モードでのデータ・コレクターの構成

多数のアプリケーション・サーバー・インスタンスを構成する場合は、データ・コレクターをサイレント・モードで構成した方が便利な場合があります。

このタスクについて

データ・コレクターをサイレント・モードで構成する場合、まず構成オプションをプロパティー・ファイルに指定します。 サンプル・プロパティー・ファイル sample_silent_config.txt が、構成ユーティリティーのパッケージに付属しています。 このファイルは以下のディレクトリーにあります。ここで *dc_home* はデータ・コレクターがインストールされているディレクトリーです。*dc_home* ディレクトリーの絶対パスについては、WebSphere Applications エージェント用のデータ・コレクターの構成の導入部を参照してください。

- Windows *dc_home*¥bin
- Linux AIX *dc_home*/bin

このファイル内の使用可能な各構成プロパティーについて詳しくは、851 ページの『データ・コレクターのサイレント構成用のプロパティー・ファイル』を参照してください。

手順

サイレント構成を実行するには、以下のステップを実行します。

1. 構成オプションをプロパティー・ファイルに指定します。 サンプル・プロパティー・ファイルをコピーして、必須指定のオプションを変更することができます。
2. ユーティリティーを 実行する前に、Java ホーム・ディレクトリーのロケーションを 設定します。
   例:
   - Windows

```
set JAVA_HOME=C:¥Program Files¥IBM¥WebSphere¥AppServer80¥java
```

- Linux AIX

```
export JAVA_HOME=/opt/IBM/AppServer80/java
```

重要 : WebSphere Application Server Liberty のモニターを構成する際には、アプリケーション・サーバーに使用されているものと同じ JVM バージョンを使用する必要があります。このようにしなければ、モニターは失敗する可能性があります。

3. 以下のディレクトリーに移動します。
   - Windows *dc_home*¥bin
   - Linux AIX *dc_home*/bin
4. コマンドを実行して、サイレント・モードでのデータ・コレクターを構成します。
   - Windows WebSphere Application Server をインストールした管理者として、以下のコマンドを実行します。

     ```
     config.bat -silent [dir_path]¥silent file
     ```

   - Linux AIX root ユーザー特権で、以下のコマンドを実行します。

     ```
     config.sh -silent [dir_path]/silent file
     ```

   ヒント : アプリケーション・サーバーのインストールに wsadmin ユーザーを使用した場合は、wsadmin ユーザーまたは root ユーザー特権で config ユーティリティーを 実行します。
5. アプリケーション・サーバー・インスタンスをモニターするためにデータ・コレクターを構成したら、アプリケーション・サーバー内でデータ・コレクターを有効にしていた場合、インスタンスを再始動する必要があります。 データ・コレクターの構成は、アプリケーション・サーバー・インスタンスが再始動されると有効になります。
6. アプリケーション・サーバー内のデータ・コレクターを有効にせずに PMI リソース・モニターを有効にした場合は、モニターを開始するために、WebSphere Applications エージェントを再始動しなければならない可能性があります。モニター・データがすぐに使用できない場合は、以下のコマンドを実行して、モニター・エージェントを再始動します。
   - Windows

     ```
     cd install_dir¥bin
     was-agent.bat stop
     was-agent.bat start
     ```

   - Linux AIX

     ```
     cd install_dir/bin
     ./was-agent.sh stop
     ./was-agent.sh start
     ```

次のタスク

サイレント構成後に、データ・コレクターを再構成するには、以下の 2 つのオプションがあります。

- **reconfig** 再構成ユーティリティーを 使用して、対話式に再構成します。説明については、846 ページの『対話式でのデータ・コレクターの再構成』を参照してください。
- サイレントに構成解除してから、同じ手順を使用してサイレントに再構成します。説明については、154 ページの『サイレント・モードでのデータ・コレクターの構成解除』を参照してください。

関連資料

851 ページの『データ・コレクターのサイレント構成用のプロパティー・ファイル』

データ・コレクターをサイレントで構成するには、まずプロパティー・ファイルで構成オプションを指定した後、構成ユーティリティーを 実行します。

データ・コレクターのサイレント構成用のプロパティー・ファイル

データ・コレクターをサイレントで構成するには、まずプロパティー・ファイルで構成オプションを指定した後、構成ユーティリティーを 実行します。

プロパティー・ファイルの作成時には、次のような考慮事項に留意してください。

- ファイル中、番号記号 (#) で始まる行は、コメントとして扱われ、処理されません。 行内の他の場所で番号記号を使用しても、それはコメントの開始とはみなされません。
- 各プロパティーは、*property* = *value* のフォーマットで、別の行に記述されます。

  property
  プロパティーの名前。構成できる有効なプロパティーのリストは、851 ページの表 223 に示されています。

  value
  プロパティーの値。一部のプロパティーのデフォルト 値は、既に用意されています。 デフォルト値を削除して、プロパティー値をブランクまたは空のままにすることができます。 空の値は、デフォルト値を使用するのとは対照的に、プロパティーが指定されていないものとして扱われます。 デフォルト値を使用する場合、ファイル内のプロパティーをコメント化することができます。
- パスワードはプレーン・テキストになります。
- プロパティーとその値においては、大/小文字が区別されます。

851 ページの表 223 で、データ・コレクターをサイレント ・モードで構成する場合に使用可能なプロパティーについて説明します。

重要 : WebSphere Application Server Liberty インスタンス用のデータ・コレクターを構成する場合、一部のプロパティーは使用されません。

表 *223.* サイレント・モードで構成ユーティリティーを 実行する際に使用可能なプロパティー

プロパティー	**コメント**
default.hostip	コンピューター・システムが複数の IP アドレスを使用している場合は、データ・コレクターが使用する IP アドレスを指定します。
ITCAM for Application Diagnostics 管理サーバーとのデータ・コレクターの統合 **重要 :** 管理サーバーが使用可能なのは、ITCAM for Application Diagnostics がある場合のみです。 WebSphere Application Server Liberty インスタンスの場合、および Cloud APM 環境内では、これらのプロパティーは使用されません。	
ms.connect	ITCAM for Application Diagnostics 環境内の管理サーバーに接続するようデータ・コレクターを構成するかどうかを指定します。 有効な値は、`True` と `False` です。
ms.kernel.host	管理サーバーの完全修飾ホスト名を指定します。
ms.kernel.codebase.port	管理サーバーが listen するコードベース・ポートを指定します。
ms.am.home	管理サーバーのホーム・ディレクトリーを指定します。
ms.am.socket.bindip	データ・コレクターが管理サーバーと通信するのに使用する IP アドレスまたはホスト名を指定します。 複数のネットワーク・インターフェースまたは IP アドレスがデータ・コレクター・コンピューター・システムで構成されている場合、そのうちの 1 つを選択します。

表 223. サイレント・モードで構成ユーティリティーを実行する際に使用可能なプロパティー (続き)	
プロパティー	**コメント**
ms.probe.controller.rmi.port	データ・コレクターがファイアウォールの背後にある場合や、データ・コレクターのコントローラー RMI ポートを変更する特別な要件がある場合、このポート番号の範囲を設定します。 このポート番号は、データ・コレクターのホストに対してファイアウォールで許可されるとおりに構成します。 例えば、`ms.probe.controller.rmi.port=8300-8399`、ms.probe.controller.rmi.port=8300 などです。
ms.probe.rmi.port	データ・コレクターがファイアウォールの背後にある場合や、データ・コレクターの RMI ポートを変更する特別な要件がある場合、このポート番号の範囲を設定します。 このポート番号は、データ・コレクターのホストに対してファイアウォールで許可されるとおりに構成します。 例えば、`ms.probe.rmi.port=8200-8299`、`ms.probe.rmi.port=8200` などです。
トランザクション・トラッキングのサポート トランザクション・トラッキング情報を表示するには、Cloud APM コンソールでトポロジー・ビューを使用可能にし、「エージェント構成」ウィンドウでトランザクション・トラッキングを有効にする必要があります。	
ttapi.enable	データ・コレクターでトランザクション・トラッキングをサポートするかどうかを指定します。有効な値は、`True` と `False` です。
ttapi.host	Transaction Framework Extension のホストを指定します。Transaction Framework Extension は、データ・コレクターからメトリックを収集する、Monitoring Agent for WebSphere Applications のコンポーネントです。ローカル・ホスト値 `127.0.0.1` を使用します。
ttapi.port	Transaction Framework Extension のポートを指定します。5457 を使用します。
データ・コレクターと ITCAM for SOA の統合 **重要 :** WebSphere Application Server Liberty インスタンスの場合、および Cloud APM 環境内では、このプロパティーは使用されません。	
soa.enable	データ・コレクターを ITCAM for SOA に統合するかどうかを指定します。この構成を完了するには、ITCAM for SOA agent がインストールされている必要があります。
Tivoli Performance Monitoring とのデータ・コレクターの統合 **重要 :** WebSphere Application Server Liberty インスタンスの場合、および Cloud APM 環境内では、このプロパティーは使用されません。	
tpv.enable	データ・コレクターが ITCAM for WebSphere Application Server バージョン 8.5 の一部として組み込まれている場合に、データ・コレクターを Tivoli Performance Monitoring に統合するかどうかを指定します。 Tivoli Performance Monitoring にアクセスするには、WebSphere Application Server 管理コンソールを使用します。 有効な値は、*True* と *False* です。
Application Performance Diagnostics Lite とのデータ・コレクターの統合 **重要 :** WebSphere Application Server Liberty インスタンスの場合、このプロパティーは使用されません。	

表 223. サイレント・モードで構成ユーティリティーを実行する際に使用可能なプロパティー (続き)	
プロパティー	**コメント**
de.enable	Application Performance Diagnostics および Application Performance Diagnostics Lite に必要な診断データを収集するかどうかを指定します。 有効な値は、True と False です。 Application Diagnostics がある場合、または将来入手する可能性がある場合は、この統合を有効にします。これによりサーバーの始動時の診断データの収集が有効になります。そのようにしないと、始動時には収集が無効になります。ユーザー・インターフェースの「エージェント構成」ページを使用して診断データの収集を有効にすることができますが、サーバーを再始動すると再び無効になります。 この設定では、Application Performance Diagnostics Lite (WebSphere Application Server および WebSphere Portal Server で実行されているアプリケーションの診断用の調査を行うためのツール) との統合も有効になります。このツールを使用すると、データをリアルタイムで分析することも、診断情報をファイルに保存しておいて後で分析することもできます。
PMI リソースおよびデータ・コレクターのモニター 選択したサーバーは常にリソース (PMI) モニター用に構成され、アプリケーション・サーバーには何らかの変更も加えられません。 このモニター・オプションで提供されるメトリックは限定的であり、WebSphere Applications エージェント でのみ機能しますが、アプリケーション・サーバーの再始動を必要とせず、パフォーマンスに影響を与えません。	
tema.appserver	データ・コレクターをアプリケーション・サーバー・インスタンス内で構成するかどうかを指定します。 アプリケーション・サーバー・インスタンス内のデータ・コレクターは、WebSphere Applications エージェントのすべてのメトリックで、また他の製品との統合で必要です。ただし、データ・コレクターを構成するには、アプリケーション・サーバーを再始動する必要があります。 また、データ・コレクターは、サーバーのパフォーマンスに影響を与える可能性があります。 有効な値は、True と False です。 このパラメーターを False に設定した場合、WebSphere Applications エージェント以外の製品との統合用のデータ・コレクター構成パラメーターは無視されます。このパラメーターが False に設定されていると、診断およびトランザクション・トラッキング機能が使用不可になり、リソース・モニター・データのみが収集されます。
tema.jmxport	リソース・モニター用の TCP/IP ポート番号。 このポートは、同じホストで稼働しているコンポーネント間の内部通信に使用されます。 デフォルト・ポートは 63355 です。このポートが使用中の場合は、別の番号を設定します。
WebSphere Applications エージェントのモニター・エージェント・コンポーネントおよび Application Performance Diagnostics Lite とデータ・コレクターとの統合	
temaconnect	WebSphere Applications エージェントのモニター・エージェント・コンポーネントにデータ・コレクターを接続するかどうかを指定します。 有効な値は、True と False です。 **重要 :** WebSphere Applications エージェントを使用するには、True の値を使用する必要があります。

表 223. サイレント・モードで構成ユーティリティーを実行する際に使用可能なプロパティー (続き)

プロパティー	コメント
tema.appserver	データ・コレクターをアプリケーション・サーバー・インスタンス内で構成するかどうかを指定します。 アプリケーション・サーバー・インスタンス内のデータ・コレクターは、WebSphere Applications エージェントのすべてのメトリックで、また他の製品との統合で必要です。ただし、これにより、アプリケーション・サーバーの再始動が必要になります。また、データ・コレクターは、サーバーのパフォーマンスに影響を与える可能性があります。 有効な値は、True と False です。 このパラメーターを False に設定した場合、データ・コレクターを WebSphere Applications エージェント以外の製品と統合するための構成パラメーター、ならびに tema.host および tema.port パラメーターが無視されます。このパラメーターが False に設定されていると、診断およびトランザクション・トラッキング機能が使用不可になり、リソース・モニター・データのみが収集されます。
tema.host	WebSphere Applications エージェントのモニター・エージェント・コンポーネントの完全修飾ホスト名または IP アドレスを指定します。ローカル・ホスト・アドレス (127.0.0.1) を使用します。
tema.port	WebSphere Applications エージェントのモニター・エージェント・コンポーネントのポート番号を指定します。デフォルト値 63335 を変更しないでください。
tema.jmxport	リソース・モニター用の TCP/IP ポート番号。 このポートは、同じホストで稼働しているコンポーネント間の内部通信に使用されます。 デフォルト・ポートは 63355 です。このポートが使用中の場合は、別の番号を設定します。
ITCAM Agent for WebSphere Applications バージョン 6 とのデータ・コレクターの統合 以下のプロパティーを使用して、WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications バージョン 6 の両方のデータを収集するように 1 つのデータ・コレクターを構成します。	
config.tema.v6	データ・コレクターを ITCAM Agent for WebSphere Applications バージョン 6 のモニター・エージェント・コンポーネントと統合するかどうかを指定します。有効な値は、True および False です。デフォルト値は、False です。
tema.host.v6	データ・コレクターを ITCAM Agent for WebSphere Applications バージョン 6 のモニター・エージェント・コンポーネントと統合するかどうかを指定します。有効な値は、True および False です。デフォルト値は、False です。
tema.port.v6	ITCAM Agent for WebSphere Applications バージョン 6 のモニター・エージェント・コンポーネントのポート番号を指定します。デフォルト値の 63336 を変更しないでください。
WebSphere Application Server のバックアップ	
was.backup.configuration	新しい構成を適用する前に、WebSphere Application Server 構成の、現行の構成をバックアップするかどうかを指定します。 有効な値は、True と False です。
was.backup.configuration.dir	バックアップ・ディレクトリーのロケーションを指定します。
拡張構成設定	
was.gc.custom.path	ガーベッジ・コレクション・ログ用のカスタム・パスを設定するかどうかを指定します。

表 223. サイレント・モードで構成ユーティリティーを実行する際に使用可能なプロパティー (続き)	
プロパティー	**コメント**
was.gc.file	カスタムのガーベッジ・コレクション・ログへのパスを指定します。 この値には、絶対パス付きでファイル名を設定します。 データ・コレクターは、サーバー・インスタンス情報を追加することで、ログ・ファイル名を自動的に変更します。 例えば、ファイル名に gc.log を指定すると、構成されるアプリケーション・サーバー・インスタンスごとに、実際の名前は *profile_name*.*cell_name*.*node_name*.*server_name*.gc.log に設定されます。 **重要 :** ガーベッジ・コレクション・ログのパスでは、${SERVER_LOG_ROOT} などの WebSphere 変数を使用できます。 ただし、%pid などのテンプレートは使用しないでください。
WebSphere 管理サービスの接続設定	
was.wsadmin.connection.host	wsadmin ツールの接続先のホストの名前を指定します。 ネットワーク・デプロイメント環境では、デプロイメント・マネージャーへの wsadmin 接続を指定します。 スタンドアロン環境では、サーバーへの wsadmin 接続を指定します。 **要確認 :** WebSphere 管理コンソールが同じシステム上にある場合、接続には localhost の値が使用されます。ただし、場合によっては、システム・ネットワークまたはセキュリティー設定により、通信には localhost を使用できないことがあります。その場合は、このパラメーターをサイレント応答ファイルで指定する必要があります。
was.wsadmin.connection.type	wsadmin ツールが WebSphere Application Server へ接続するのに使用する必要のあるポートを指定します。
was.wsadmin.connection.port	wsadmin ツールが WebSphere Application Server へ接続するのに使用する必要のあるポートを指定します。
WebSphere Application Server のグローバル・セキュリティー設定	
was.wsadmin.username	IBM WebSphere Application Server 管理コンソールにログオンする権限があるユーザーのユーザー ID を指定します。このユーザーは、アプリケーション・サーバー上でエージェントの役割を持っている必要があります。
was.wsadmin.password	was.wsadmin.username プロパティーに指定されたユーザーに対応するパスワードを指定します。
was.client.props	クライアント・プロパティー・ファイルからセキュリティー設定を取り出すかどうかを指定します。 指定できる値は、True と False です。
WebSphere Application Server の設定	
was.appserver.profile.name	構成するアプリケーション・サーバー・プロファイルの名前を指定します。WebSphere Application Server Liberty の場合は使用されません。
was.appserver.home	WebSphere Application Server ホーム・ディレクトリーを指定します。
was.appserver.cell.name	WebSphere Application Server のセル名を指定します。WebSphere Application Server Liberty の場合は使用されません。
was.appserver.node.name	WebSphere Application Server のノード名を指定します。WebSphere Application Server Liberty の場合は使用されません。
WebSphere Application Server のランタイム・インスタンス設定	

表 223. サイレント・モードで構成ユーティリティーを実行する際に使用可能なプロパティー (続き)	
プロパティー	**コメント**
was.appserver.server.name	構成するアプリケーション・サーバー・プロファイル内のアプリケーション・サーバー・インスタンスを指定します。 **ヒント:** • サイレント応答ファイルには、このプロパティーの複数のインスタンスを含めることができます。 • 別のサーバーを追加する場合は、その別サーバー (#[SERVER]) のコメントを外してサーバー名を追加します。
tema.serveralias	このアプリケーション・サーバー・インスタンスのモニター情報を格納するノードの、モニター・ユーザー・インターフェース 内での名前を指定します。デフォルトは、サーバー名にそのノード名を結合したものになります。 **重要:** 別名に使用できる文字は、A から Z、a から z、下線 (_)、ダッシュ (-)、およびピリオド (.) のみです。これ以外の文字を別名に使用しないでください。 **ヒント:** サイレント応答ファイルには、このプロパティーの複数のインスタンスを含めることができます。 **要確認:** サーバー別名を変更すると、Cloud APM コンソールに登録されているエージェント・インスタンス名が変更されます。今回がデータ・コレクターの初めての構成ではなく、サーバー別名を変更した場合、一部のキャッシュ・ファイルをクリアする必要があります。詳細な説明については、旧サーバー名が含まれているキャッシュ・ファイルのクリア方法を参照してください。

構成ユーティリティーが失敗した場合のデータ・コレクターの手動構成
提供されている構成ユーティリティーを 使用して WebSphere Applications エージェントのデータ・コレクターを構成できない場合、WebSphere 管理コンソールでデータ・コレクターを手動で構成できます。

始める前に

- WebSphere Applications エージェント をインストールします。
- データ・コレクター・ホーム・ディレクトリーを把握します。このディレクトリーは、データ・コレクターの構成で必要になります。デフォルトは、/opt/ibm/apm/agent/yndchome/7.3.0.14.08 (Linux および UNIX システムの場合) または C:¥IBM¥APM¥dchome¥7.3.0.14.08 (Windows システムの場合) です。
- Liberty サーバー用のデータ・コレクターを構成する場合は、Liberty サーバーのホーム・ディレクトリーを確認してください。例えば、/opt/ibm/was/liberty/usr/servers/defaultServer です。
- itcam_wsBundleMetaData.xml という名前のファイルが *dc_home*/runtime/wsBundleMetaData フォルダーに存在し、そのファイルに以下の内容が含まれていることを確認してください。フォルダーまたはファイルが存在しない場合は、手動で作成します。

  要確認: *plugins_dir_within_dc_home* 値は、データ・コレクターのホーム・ディレクトリー内の plugins フォルダーの絶対パスに設定する必要があります。デフォルトは、/opt/ibm/apm/agent/yndchome/7.3.0.14.08/plugins (Linux および UNIX システムの場合) または C:¥IBM¥APM¥dchome¥7.3.0.14.08¥plugins (Windows システムの場合) です。

```
<bundles>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.bundlemanager_7.2.0.jar</bundle>
  </directory>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.classicsca_7.2.0.jar</bundle>
```

```
  </directory>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.toolkitsca.classicsca_7.2.0.jar</bundle>
  </directory>
</bundles>
```

このタスクについて

重要 :

- WebSphere 管理ユーザーとして、データ・コレクターに関する WebSphere Application Server の構成を手動で変更する必要があります。
- データ収集のために WebSphere Application Server に手動で変更を加える作業は、経験豊かな WebSphere 管理者が行う必要があります。手動による構成変更で何らかのエラーが発生すると、アプリケーション・サーバーは開始されないことがあります。
- アプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターを手動で構成したら、構成解除ユーティリティーを 使用してデータ・コレクターを構成解除することはできません。そうではなく、データ・コレクターを手動で構成解除する必要があります。

手順

- WebSphere Application Server 用のデータ・コレクターを手動で構成するには、857 ページの『従来の WebSphere Application Server 用のデータ・コレクターの手動構成』を参照してください。
- Liberty サーバー用のデータ・コレクターを手動で構成するには、859 ページの『WebSphere Application Server Liberty 用のデータ・コレクターの手動構成』を参照してください。

従来の *WebSphere Application Server* 用のデータ・コレクターの手動構成

手順

1. WebSphere の管理コンソールにアドミニストレーターとしてログインします。
2. ナビゲーション・ペインで、「**サーバー**」をクリックし、「**サーバー・タイプ**」を展開して、「**WebSphere Application Server**」をクリックします。
3. 「構成」タブの「**サーバー・インフラストラクチャー**」セクションで、「**Java およびプロセス管理**」を展開し、「**プロセス定義**」をクリックします。
4. 「**追加プロパティー**」のセクションで、「**Java 仮想マシン**」をクリックします。
5. 「**汎用 JVM 引数**」フィールドに、以下の項目を追加します。

   ```
   -agentlib:am_ibm_16=${WAS_SERVER_NAME} -Xbootclasspath/p:${ITCAMDCHOME}/
   toolkit/lib/bcm-bootstrap.jar -Djava.security.policy=${ITCAMDCHOME}/itcamdc/
   etc/datacollector.policy -verbosegc
   ```

   項目を追加する場合、以下のことに注意してください。

   - すべての項目を 1 行で指定する必要があります。
   - 異なる引数は、負符号 (-) の前にスペースを入れて区切ります。それ以外の場所にはスペースを使用しないでください。
6. 「**適用**」をクリックし、マスター構成への変更を保存します。
   - Network Deployment 環境でない場合は、「**保存**」をクリックします。
   - Network Deployment 環境の場合は、必ず「**コンソール設定**」オプションで「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
7. ナビゲーション・ペインで「**サーバー**」をクリックし、「**サーバー・タイプ**」を展開し、「**WebSphere Application Server**」をクリックしてから、サーバー名をクリックします。
8. 「構成」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**環境エントリー**」に移動します。
9. オペレーティング・システム、ハードウェア・プラットフォーム、およびアプリケーション・サーバー JVM に応じて、以下の環境エントリーを設定します。

表 *224.* 環境エントリー		
プラットフォーム	**環境エントリー名**	**環境エントリーの値**
AIX R6.1 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536
AIX R7.1 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536
Solaris 10 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/sol296
Solaris 11 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/sol296
Linux Intel R2.6 (32 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/li6263
Linux x86_64 R2.6 (64 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx8266
Linux on Power リトル・エンディアン (64 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpl266
Linux on System z (32 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/ls3263
Linux on System z (64 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/ls3266
Windows (32 ビット JVM)	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win32
Windows (64 ビット JVM)	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win64

10. 「**適用**」をクリックし、マスター構成への変更を保存します。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
    - Network Deployment 環境の場合は、必ず「**コンソール設定**」オプションで「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
11. ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックします。
12. スコープを適切なサーバー・レベルに指定し、*ITCAMDCHOME* 変数を追加します。*ITCAMDCHOME* 変数の値をデータ・コレクターのホーム・ディレクトリーに設定します。例えば、`/opt/ibm/apm/agent/yndchome/7.3.0.14.08` です。
13. 「**適用**」をクリックし、マスター構成への変更を保存します。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
    - Network Deployment 環境の場合は、必ず「**コンソール設定**」オプションで「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
14. アプリケーション・サーバーを再始動します。

タスクの結果

これで、このアプリケーション・コンポーネントをアプリケーションに追加した後に、Cloud APM コンソールで WebSphere Applications エージェントのデータを確認できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーションを追加または編集する方法に関する説明については、1098 ページの『アプリケーションの管理』を参照してください。

次のタスク

データ・コレクターを手動で構成した後に、提供されている unconfig ユーティリティーを 使用してデータ・コレクターを構成解除することはできません。代わりに、データ・コレクターを手動で構成解除してください。説明については、158 ページの『データ・コレクターの手動による構成解除』を参照してください。

WebSphere Application Server Liberty 用のデータ・コレクターの手動構成

手順

1. Liberty サーバーのホーム・ディレクトリーにナビゲートします。例えば、/opt/ibm/wlp/usr/servers/defaultServer です。
2. 以下のパラメーターを追加して jvm.options ファイルを編集します。ここで、*dc_home* はデータ・コレクターのホーム・ディレクトリー、*server_name* は Liberty サーバー名です。 jvm.options ファイルが存在しない場合は、テキスト・エディターで作成します。

   ```
   -agentlib:am_ibm_16=server_name
   –Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
   -Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
   -verbosegc
   ```

   項目を追加する場合、以下のことに注意してください。

   - 項目は、それぞれ 1 行に収めなければなりません。
   - *server_name* は実際の Liberty サーバー名に置き換えます。例えば、defaultServer です。
   - *dc_home* は、実際のデータ・コレクター・ホーム・ディレクトリーに置き換えてください。例えば、/opt/ibm/apm/agent/yndchome/7.3.0.14.08 です。
3. 同じディレクトリーにある server.env ファイルを開き、オペレーティング・システムに応じて環境エントリーに以下のパスを追加します。ここで、*dc_home* はデータ・コレクターのホーム・ディレクトリーです。 server.env ファイルが存在しない場合は、テキスト・エディターで作成します。

   表 *225.* 環境エントリー

プラットフォーム	環境エントリー名	環境エントリーの値
AIX R6.1 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
AIX R7.1 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
Solaris 10 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Solaris 11 (64 ビット JVM)	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Linux x86_64 R2.6 (64 ビット JVM)	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/lx8266
Linux Intel R2.6 (32 ビット JVM)	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/li6263
Windows (32 ビット JVM)	PATH	/lib;*dc_home*/toolkit/lib/win32
Windows (64 ビット JVM)	PATH	/lib;*dc_home*/toolkit/lib/win64

4. 同じディレクトリーにある server.xml ファイルを開き、以下の行を追加してモニター機能を有効にします。

```
<featureManager>
              <feature>webProfile-7.0</feature>
              <feature>monitor-1.0</feature>
              <feature>usr:itcam-730.140</feature>
   </featureManager>
```

5. Liberty サーバーを再始動します。

タスクの結果

これで、このアプリケーション・コンポーネントをアプリケーションに追加した後に、Cloud APM コンソールで WebSphere Applications エージェントのデータを確認できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーションを追加または編集する方法に関する説明については、1098 ページの『アプリケーションの管理』を参照してください。

次のタスク

データ・コレクターを手動で構成した後に、提供されている unconfig ユーティリティーを 使用してデータ・コレクターを構成解除することはできません。代わりに、データ・コレクターを手動で構成解除してください。説明については、158 ページの『データ・コレクターの手動による構成解除』を参照してください。

(エージェント共存) WebSphere Applications エージェントおよびデータ・コレクターの構成

WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications の両方がインストールされているエージェント共存環境では、エージェントの追加構成を実行し、別の手順に従ってデータ・コレクターを構成する必要があります。

このタスクについて

エージェント共存環境では、WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications の両方にデータを送信するためにデータ・コレクターを 1 つのみ構成します。両エージェントでは、別々のポートを使用してデータ・コレクターからの要求を listen する必要があります。

手順

1. 古い IBM® Tivoli® Monitoring インフラストラクチャーにインストールされている ITCAM Agent for WebSphere Applications のデータ・コレクターがご使用の環境に存在する場合は、アンインストールします。
2. IBM Cloud Application Performance Management 8.1.3 以降で提供されている WebSphere Applications エージェントをインストールします。これにより、エージェント共存でサポートされるデータ・コレクターのバージョン 7.3.0.11.0 以降がインストールされていることが保証されます。
3. データ・コレクターを構成するユーザー ID、およびアプリケーション・サーバーをインストールしたユーザー ID が、エージェントで必要な適切なユーザー特権を持っていることを確認してください。説明については、837 ページの『ユーザー・アクセス権限の要件の確認』を参照してください。
4. WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications が別々のポート番号を使用してデータ・コレクターからの要求を listen していることを確認します。ポート番号は、ご使用の環境内の他のどのコンポーネントでも使用されていない固有の番号である必要があります。必要に応じて、エージェントを再度構成して、ポートを変更します。
   - WebSphere Applications エージェントを構成する方法については、861 ページの『WebSphere Applications エージェントの構成』を参照してください。
   - ITCAM Agent for WebSphere Applications を構成する方法については、ITCAM for Application Diagnostics の資料 または ITCAM for Applications の資料を参照してください。
5. 提供されている構成ユーティリティーを 使用して、データ・コレクターを構成します。説明については、862 ページの『エージェントの共存環境用のデータ・コレクターの構成』を参照してください。

   ヒント: データ・コレクターの構成を熟知している場合は、サイレント・モードでデータ・コレクターを構成することもできます。説明については、849 ページの『サイレント・モードでのデータ・コレクターの構成』を参照してください。

WebSphere Applications エージェントの構成

エージェントの共存環境では、データ・コレクターは、WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications で共有されます。両エージェントでは、別々のポートを使用してデータ・コレクターからの要求を listen する必要があります。必要に応じて、エージェントを構成してポートを変更する必要があります。

このタスクについて

- Linux または AIX システムでは、エージェントを構成するには、構成スクリプトを実行してプロンプトに応答することで対話式に行うか、サイレント応答ファイルを作成して対話なしで構成スクリプトを実行することでサイレントに行います。
- Windows システムでは、エージェントを構成するには、サイレント応答ファイルを作成して構成スクリプトを実行するか、提供されている「Monitoring Services の管理 (Manage Monitoring Services)」ユーティリティーを使用します。Windows システム上で Monitoring Services の管理を開始する方法については、186 ページの『Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用』を参照してください。

手順

- サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。

  a) サイレント応答ファイルとして `.txt` ファイルを作成します。

  b) サイレント応答ファイル内に以下のパラメーターを指定します。構文は、*parameter_name=parameter_value* です。

  configure_type
  構成のタイプを指定します。このパラメーターは必須です。

  有効な値は、エージェント構成の場合、`tema_configure` です。

  KYN_ALT_NODEID
  エージェントを識別するための代替ノード ID を指定します。このパラメーターは必須です。

  有効な値は、24 文字までの英数字ストリングです。

  KYN_PORT
  エージェントで使用される listen ポートを指定します。これは、エージェントがデータ・コレクターからの接続要求を listen するために使用する TCP ソケットです。このパラメーターは必須です。

  デフォルト値は 63335 です。

  要確認 : エージェントの共存環境では、ここで指定したポート番号が、ITCAM Agent for WebSphere Applications によって使用されていないことを確認してください。

  例えば、作成した `.txt` ファイルで以下の行を追加します。

  ```
  configure_type=tema_configure
  KYN_ALT_NODEID=WASAgent
  KYN_PORT=63335
  ```

  c) ファイルを保存して閉じてから、以下のコマンドを入力して構成スクリプトを実行します。

  – Linux AIX *install_dir*/bin/was-agent.sh config *path_to_responsefile*

  – Windows *install_dir*¥bin¥was-agent.bat config *path_to_responsefile*

  ここで、*install_dir* はエージェント・インストール・ディレクトリーです。デフォルトは、Windows システムの場合は `C:¥IBM¥APM`、Linux システムおよび AIX システムの場合は `/opt/ibm/apm/agent` です。

  d) 構成が完了したら、以下のコマンドを使用して WebSphere Applications エージェントを再始動します (実行中でない場合)。

  - Linux AIX *install_dir*/bin/was-agent.sh start
  - Windows *install_dir*¥bin¥was-agent.bat start
- スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。
  a) コマンド行で、*install_dir*/bin ディレクトリーに移動します。ここで、*install_dir* は、エージェント・インストール・ディレクトリーです。

  デフォルトは、Linux システムおよび AIX システムの場合は /opt/ibm/apm/agent です。
  b) このディレクトリーで以下のように構成スクリプトを実行します。

```
./was-agent.sh config
```

  c) プロンプトが出されたら、1 と入力して Enter キーを押し、WebSphere Applications エージェントのモニター・エージェントの設定を編集します。
  d) モニター・エージェントを識別するための代替ノード ID を求めるプロンプトが出されるまで Enter キーを押します。
  e) ノード ID を指定し、Enter キーを押します。ノード ID の有効なフォーマットは、24 文字までの英数字ストリングです。
  f) ポート番号を求めるプロンプトが出されたら、エージェントがデータ・コレクターからの接続要求を listen するために使用するポートを指定し、Enter キーを押します。

  要確認 : エージェントの共存環境では、指定するポートが、ITCAM Agent for WebSphere Applications で使用されていないことを確認してください。
  g) 構成が完了したら、WebSphere Applications エージェントを再始動します。

タスクの結果
WebSphere Applications エージェントが構成されました。

次のタスク
次に、データ・コレクターを構成する必要があります。データ・コレクターを構成する際に、WebSphere Applications エージェントおよび ITCAM Agent for WebSphere Applications 用に構成したポート番号を指定するように求められます。説明については、862 ページの『エージェントの共存環境用のデータ・コレクターの構成』を参照してください。

エージェントの共存環境用のデータ・コレクターの構成
WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications の両方が環境に含まれている場合、両エージェントに対してデータ・コレクターを 1 つだけ構成できます。

始める前に
860 ページの『(エージェント共存) WebSphere Applications エージェントおよびデータ・コレクターの構成』に記載されている他のステップが完了していることを確認します。

このタスクについて

提供されている対話式構成ユーティリティーを 使用して、WebSphere Applications エージェントと ITCAM Agent for WebSphere Applications がともに存在していてデータ・コレクターを共有する環境用にデータ・コレクターを構成します。

制約事項 : ITCAM Agent for WebSphere Applications では、以下のコンポーネントおよび製品とのデータ・コレクターの統合はサポートされません。

- ITCAM for Application Diagnostics 管理サーバー
- ITCAM for Transactions
- Tivoli Performance Viewer

要確認 : ITCAM Agent for WebSphere Applications では、WebSphere Application Server Liberty のモニターはサポートされません。WebSphere Application Server Liberty をモニターするには、WebSphere Applications エージェントのみを使用します。Liberty のモニター用のデータ・コレクター構成については、841 ページの『対話式でのデータ・コレクターの構成』または 849 ページの『サイレント・モードでのデータ・コレクターの構成』を参照してください。

手順

1. アプリケーション・サーバーのインストールに使用するユーザー ID でシステムにログインします。
2. コマンド行で、*dc_home* ディレクトリー内の bin ディレクトリーに移動します。*dc_home* ディレクトリーは次のとおりです。
   - Windows *install_dir*¥dchome¥7.3.0.14.08
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08
3. 以下のコマンドを実行して、構成ユーティリティーを始動します。
   - Windows `config.bat`
   - Linux AIX `./config.sh`

   構成ユーティリティーが始動して、ローカル・コンピューター・システム上で検出されたすべてのネットワーク・カードの IP アドレスが表示されます。
4. 使用する IP アドレスに対応する数値を入力し、Enter キーを押します。

   構成ユーティリティーに、システム上で検出された WebSphere Application Server のホーム・ディレクトリーが表示されます。
5. アプリケーション・サーバーのホーム・ディレクトリーを求めるプロンプトが出されたら、WebSphere Application Server のホーム・ディレクトリーに対応する番号を入力するか、アプリケーション・サーバーのホーム・ディレクトリーの絶対パスを入力し、Enter キーを押します。

   構成ユーティリティーに、指定されたホーム・ディレクトリーの下で検出されたすべてのアプリケーション・サーバー・プロファイルが表示されます。
6. 構成するアプリケーション・サーバー・プロファイルを求めるプロンプトが出されたら、WebSphere Application Server プロファイルに対応する番号を入力し、Enter キーを押します。

   構成ユーティリティーは、指定した WebSphere Application Server プロファイルについて WebSphere グローバル・セキュリティーが有効かどうかを、以下のように示します。グローバル・セキュリティーが有効でない場合は、ステップ 863 ページの『8』にスキップします。
7. クライアント・プロパティー・ファイルからセキュリティー設定を取り出すかどうかを指定します。構成ユーティリティーが、該当するクライアント・プロパティー・ファイルからユーザー名とパスワードを取得できるようにするには 1 を入力します。そうしない場合は、2 を入力して、WebSphere 管理者のユーザー名とパスワードを入力します。

   データ・コレクターはリモート・メソッド呼び出し (RMI) または SOAP プロトコルを使用して WebSphere Administrative Services と通信します。プロファイルでグローバル・セキュリティーが有効になっている場合は、そのアプリケーション・サーバー・プロファイルに対して WebSphere Application Server 管理コンソールにログインする権限があるユーザーのユーザー ID とパスワードを指定する必要があります。あるいは、ユーザー名とパスワードを暗号化して、それらをクライアント・プロパティー・ファイルに保存してから、データ・コレクターを構成することもできます。`sas.client.props` ファイル (RMI 接続の場合) または `soap.client.props` ファイル (SOAP 接続の場合) を使用する必要があります。
8. WebSphere 管理コンソールのホスト名を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、WebSphere 管理コンソールのホスト名または IP アドレスを指定します。デフォルト値は `localhost` です。
9. まだデータ収集用に構成されていないすべてのサーバー・インスタンスが構成ユーティリティーによってリストされたら、1 つ以上のアプリケーション・サーバー・インスタンスをリストから選択します。データ収集用に構成するアプリケーション・サーバー・インスタンスに対応する番号を入力するか、アスタリスク (*) を入力してすべてのアプリケーション・サーバー・インスタンスをデータ収集用

に構成してから、Enter キーを押してください。サーバーのサブセットを指定する場合は、サーバーを表すそれぞれの番号をコンマで区切って入力します。
例えば、1,2,3 です。

要確認 :

- スタンドアロン環境の場合、構成中はアプリケーション・サーバー・インスタンスが実行中である必要があります。
- ネットワーク・デプロイメント環境では、デプロイメント・マネージャーが稼働している必要があります。
- 選択するアプリケーション・サーバー・インスタンスが、モニター対象のアプリケーションまたはサービスをホストする実際のサーバーであるようにします。

構成ユーティリティーには、WebSphere Applications エージェント用にデータ・コレクターを統合するためのオプションが用意されています。

10. 「**Agent for WebSphere Applications との統合 (Integration with Agent for WebSphere Applications)**」セクションで、データ・コレクターをモニター・エージェントと統合することを指定します。1 と入力してこの統合オプションを選択してから、Enter キーを押す必要があります。

    選択したサーバーが、PMI リソース・モニター用に登録されます。
11. データ・コレクターをアプリケーション・サーバー・インスタンス内で構成するかどうかを指定します。「はい」を表す 1 を入力してから、Enter キーを押す必要があります。
12. 診断データ用にデータ・コレクターを有効にするかどうかを指定します。「はい」の場合には 1、「いいえ」の場合には 2 を入力します。
13. V8 モニター・エージェント・コンポーネントのホスト名を求めるプロンプトが出されたら、WebSphere Applications エージェントのホスト名または IP アドレスを入力するか、デフォルトを受け入れます。
14. V8 モニター・エージェントのポート番号を求めるプロンプトが出されたら、WebSphere Applications エージェントによって使用されるポート番号を入力します。

    要確認 : デフォルト値が別のコンポーネントによって使用されている場合、デフォルト値を使用するのが適切でないことがあります。指定したポートが、ご使用の環境内の他のどのコンポーネントでも使用されていないことを確認する必要があります。
15. V6 モニター・エージェントを構成することを指定します。ITCAM Agent for WebSphere Applications を構成するために 1 と入力し、Enter キーを押します。
16. V6 モニター・エージェントのホスト名または IP アドレスを求めるプロンプトが出されたら、ITCAM Agent for WebSphere Applications のホスト名または IP アドレスを指定します。
17. V6 モニター・エージェントのポート番号を求めるプロンプトが出されたら、ITCAM Agent for WebSphere Applications のモニター・エージェント・コンポーネントによって使用されるポート番号を入力します。

    要確認 : デフォルト値が別のコンポーネントによって使用されている場合、デフォルト値を使用するのが適切でないことがあります。指定したポートが、ご使用の環境内の他のどのコンポーネントでも使用されていないことを確認する必要があります。
18. サーバー別名を求めるプロンプトが出されたら、デフォルト値は使用せず、使用する固有のサーバー別名を指定します。複数のアプリケーション・サーバー・インスタンスを構成する場合、構成ユーティリティーは、インスタンスごとに別名を求めるプロンプトを出します。

    重要 : 別名に使用できる文字は、A から Z、a から z、下線 (_)、ダッシュ (-)、およびピリオド (.) のみです。これ以外の文字を別名に使用しないでください。

    サーバー別名は、Cloud APM コンソールに表示されるエージェント・インスタンス名 (MSN とも呼ばれます) の最初の修飾子です。デフォルトは、サーバー名にそのノード名を結合したものになります。例えば、別名 **node1server1** は、**node1** というノード内の **server1** というサーバーを示します。
19. PMI リソース・モニター用の TCP/IP ポート番号を求めるプロンプトが出されたら、Enter キーを押してデフォルトを受け入れるか、新しい番号を入力します。デフォルトのポートは 63355 です。

    このポートは、同じホストで稼働しているコンポーネント間の内部通信に使用されます。 デフォルト・ポートが使用中の場合は、別の番号を設定します。

20. 「**トランザクション・トラッキングのサポート**」セクションで、トランザクション・トラッキングを有効にするかどうかを指定します。「はい」の場合は、1 と入力します。「いいえ」の場合は、2 と入力し、22 までスキップします。

    要確認 : トランザクション・トラッキング情報を表示するには、Cloud APM コンソールの「エージェント構成」ページでトランザクション・トラッキングを有効にする必要があります。
21. Transaction Framework Extension のデフォルトのホスト名または IP アドレスを受け入れます。Transaction Framework Extension は、データ・コレクターからメトリックを収集する、WebSphere Applications エージェントの内部コンポーネントです。
22. データ・コレクターが Transaction Framework Extension に接続するために使用するデフォルトのポート番号を受け入れます。デフォルトは 5457 です。
23. データ・コレクターを Application Performance Diagnostics Lite と統合するかどうかを指定します。Enter キーを押してデフォルトの「いいえ」を受け入れます。
24. 「**拡張設定**」セクションで、ガーベッジ・コレクション・ログのパスを変更するかどうかを指定します。ガーベッジ・コレクション・ログのパスを選択する場合は、1 を入力します。 そうしない場合は、2 を入力して、 ステップ 865 ページの『26』 へスキップします。
25. ガーベッジ・コレクション・ログのパスを指定します。絶対パスを使用して、ファイル名を入力します。

    例えば、ファイル名に gc.log を指定すると、構成されるアプリケーション・サーバー・インスタンスごとに、実際の名前は *profile_name*.*cell_name*.*node_name*.*server_name*.gc.log に設定されます。

    重要 : ガーベッジ・コレクション・ログのパスには、${SERVER_LOG_ROOT} などの WebSphere 変数を使用できます。 ただし、%pid などのテンプレートは使用しないでください。
26. 「**データ・コレクター構成の要約**」セクションで、指定したアプリケーション・サーバー・インスタンスに適用されるデータ・コレクター構成の要約を確認します。必要に応じて、構成の設定を変更してください。
27. a と入力して変更を受け入れます。
28. プロンプトが出されたら、現在の構成のバックアップを作成するかどうかを指定します。現在の構成のバックアップを作成するには、1 を入力します。 そうしない場合は、2 を入力します。
29. 構成ユーティリティーの 指示に従って、アプリケーション・サーバー・インスタンスを再始動します。
    a) アプリケーション・サーバー・プロファイルのホーム・ディレクトリー内の bin ディレクトリーに移動します。例えば、opt/IBM/WebSphere/AppServer/profiles/*profile_name*/bin です。
    b) コマンド・コンソールで **stopServer** コマンドを入力して、アプリケーション・サーバーを停止します。
    - Linux AIX ./stopServer.sh *server_name*
    - Windows stopServer.bat *server_name*

    c) プロンプトが出されたら、WebSphere 管理コンソールの管理者のユーザー ID とパスワードを入力します。
    d) コマンド・コンソールで **startServer** コマンドを入力して、アプリケーション・サーバーを再び始動します。
    - Linux AIX ./startServer.sh *server_name*
    - Windows startServer.bat *server_name*

    データ・コレクターの構成は、アプリケーション・サーバーが再始動されると有効になります。
30. Cloud APM コンソールにログインして、ダッシュボードにデータを表示します。
    a) サービスが使用可能になったことを知らせる E メールに記載されているリンクを使用して、コンソールにアクセスします。あるいは、IBM Marketplace Web サイトからコンソールにアクセスします。説明については、977 ページの『Cloud APM コンソールの始動』を参照してください。

b) アプリケーション・エディターを使用して、モニター対象アプリケーション・サーバーを Application Performance ダッシュボードに追加します。新規コンポーネントとして既存のアプリケーションに追加するか、アプリケーションを作成してこのコンポーネントを含めることができます。

アプリケーション・エディターについて詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

次のタスク

- データ・コレクターの構成に使用されている現行ユーザー ID が、アプリケーション・サーバーを実行しているユーザーの ID と同じでない場合は、データ・コレクターの構成用のユーザー ID が、データ・コレクター・ホーム・ディレクトリー内の `runtime` および `logs` ディレクトリーに対する読み取り許可と書き込み許可を備えていることを確認してください。これらの 2 つのサブディレクトリーは、サーバーの再始動時にアプリケーション・サーバーを実行しているユーザーの ID によって作成されます。
- WebSphere Applications エージェントの場合、Cloud APM コンソールにログインして、ダッシュボードでモニター・データを表示します。ITCAM Agent for WebSphere Applications の場合、Tivoli Enterprise Portal にログインしてデータを表示します。モニター・データがすぐに使用できない場合は、以下のコマンドを実行して、モニター・エージェントを再始動します。
  - Windows

    ```
    cd install_dir¥bin
    was-agent.bat stop
    was-agent.bat start
    ```

  - Linux AIX

    ```
    cd install_dir/bin
    ./was-agent.sh stop
    ./was-agent.sh start
    ```

- サーバー別名を変更すると、Cloud APM コンソールに登録されているエージェント・インスタンス名が変更されます。今回がデータ・コレクターの初めての構成ではなく、サーバー別名を変更した場合、以下のステップを実行して、一部のキャッシュ・ファイルをクリアする必要があります。

  1. モニター・エージェントが実行されている場合は停止します。
  2. テキスト・エディターで、以下のディレクトリー内の *hostname*_yn.xml ファイルを開きます。ここで、*hostname* は、WebSphere Applications エージェントまたは ITCAM Agent for WebSphere Applications がインストールされているホストの名前です。
     - Windows *install_dir*¥TMAITM6_x64 (デフォルトは、C:¥IBM¥APM¥TMAITM6_x64 (WebSphere Applications エージェントの場合) または C:¥IBM¥ITM¥TMAITM6_x64 (ITCAM Agent for WebSphere Applications の場合))
     - Linux AIX *install_dir*/config (デフォルトは、/opt/ibm/apm/agent/config (WebSphere Applications エージェントの場合) または /opt/ibm/itm/agent/config (ITCAM Agent for WebSphere Applications の場合))
  3. 以下のストリングで始まり、以前のサーバー名が含まれている行を見つけます。例えば、was85.win4net01Cell02.win4net01Node02.AppSrv01.server1 (server1 はアプリケーション・サーバーの以前の名前)。

     ```
     <!ENTITY was_product_code.cellname.nodename.profilename.servername
     ```

     ここで、*was_product_code* は WebSphere Application Server の製品コードであり、*cellname* はセルの名前であり、*nodename* はノード名であり、*profilename* はアプリケーション・サーバー・プロファイルの名前であり、*servername* はアプリケーション・サーバーの以前の名前です。
  4. 行で示されている .XML ファイルを現行ディレクトリー内で見つけ、そのファイルを削除します。
  5. ステップ 3 で見つけた行を *hostname*_yn.xml ファイルから削除します。
  6. *hostname*_yn.xml ファイルの末尾にある、以前のサーバー名が含まれている行を削除します。
  7. ファイルを保存して閉じます。

8. モニター・エージェントを再始動します。

Cloud APM サーバーでオファリング・タイプを変更する場合のデータ・コレクターの再構成

Cloud APM サーバーにインストールしたオファリング・タイプを Cloud APM, Base から Cloud APM, Advanced に変更し、WebSphere Applications エージェントが Cloud APM, Base オファリングでインストールおよび構成されている場合、Cloud APM, Advanced オファリングで提供されているエージェント拡張機能を使用するには、前の WebSphere Applications エージェントをアンインストールし、Cloud APM, Advanced オファリングでエージェントを再びインストールする必要があります。あるいは、新しいオファリングで使用可能な機能に対してデータ・コレクターを再構成することもできます。

このタスクについて

WebSphere Applications エージェントの構成は、エージェントのインストールに使用するエージェント・パッケージによって異なります。Cloud APM サーバーでオファリング・タイプを変更した後は、以下の 2 つの方法で新しいオファリングのエージェント機能を使用可能にすることができます。

- 以前のオファリングとともにインストールしたエージェントを削除した後、新しいオファリングのエージェントをインストールする。
- 新しいオファリングの機能を使用するようにデータ・コレクターを再構成する。

手順

- 以前のオファリングとともにインストールしたエージェントを削除した後、新しいオファリングのエージェントをインストールする。
  a) データ・コレクターを構成解除します。説明については、152 ページの『WebSphere Applications エージェント: データ・コレクターの構成解除』を参照してください。
  b) 以前のオファリングのエージェント・パッケージとともにインストールした WebSphere Applications エージェントをアンインストールします。説明については、150 ページの『エージェントのアンインストール』を参照してください。
  c) 新しいオファリングのエージェント・パッケージを使用して WebSphere Applications エージェントをインストールし、再度データ・コレクターを構成します。説明については、838 ページの『簡易構成ユーティリティーを 使用したデータ・コレクターの構成』を参照してください。
- 新しいオファリングの機能を使用するようにデータ・コレクターを再構成する。
  a) データ・コレクターのホーム・ディレクトリーにある `offering.id` ファイルを編集し、新しいオファリング・タイプに応じて **IOFFERING** の値を以下のいずれかの値に変更します。

  BASE
  新しいオファリング・タイプが Cloud APM, Base Private である場合。

  ADVANCED
  新しいオファリング・タイプが Cloud APM, Advanced Private である場合。

  データ・コレクターのホーム・ディレクトリーは以下のとおりです。

  – Windows *install_dir*¥dchome¥7.3.0.14.08
  – Linux AIX *install_dir*/yndchome/7.3.0.14.08

  b) データ・コレクターを再構成し、新しいオファリング・タイプでサポートされる機能に応じて、データ・コレクターで診断、トランザクション・トラッキング、またはそれら両方を有効にします。データ・コレクターの構成方法について詳しくは、838 ページの『簡易構成ユーティリティーを 使用したデータ・コレクターの構成』を参照してください。
  c) WebSphere Application Server を再始動します。
  d) Cloud APM コンソールの任意のページで、「**システム構成 (System Configuration)**」 > 「**エージェント構成**」をクリックして、「**エージェント構成**」ページを開きます。トランザクション・トラッキングの設定が、新しいオファリング・タイプで使用可能な機能と整合していることを確認します。整合していない場合は、設定を更新します。

トランザクション・トラッキングの設定は、Cloud APM, Advanced では有効にする必要がありますが、Cloud APM, Base では無効にする必要があります。

Docker コンテナー内の WebSphere Application Server Liberty のモニター
Docker コンテナー内の Liberty プロファイルをモニターするには、事前にいくつかのオプションを指定して **docker run** コマンドを実行してデータ・コレクターを構成し、WebSphere Application Server Liberty を開始できるようにする必要があります。

始める前に
Docker ホストに WebSphere Applications エージェントをインストールする必要があります。

このタスクについて

Docker コンテナー内で実行されているそれぞれの Liberty プロファイルについて、リソース・メトリック、トランザクション・メトリック、および診断データを収集し、Docker ホストで実行されているモニター・エージェントにデータを送信するために、データ・コレクターが必要です。同じ Docker ホスト上で構成されているすべてのデータ・コレクターは、そのホスト上の同じモニター・エージェントを共有します。

手順

Liberty プロファイル・コンテナー用のデータ・コレクターを構成するには、以下のステップを実行します。

1. `.txt` 形式のサイレント応答ファイルを作成し、そのファイルに以下の構成オプションを指定して保存します。

   ```
   tema.host=agent_host
   was.appserver.server.name=liberty_profile_name
   ```

   tema.host は、モニター・エージェント・ホストの IP アドレスを指定するために使用します。
   was.appserver.server.name は、Liberty プロファイルの名前を指定するために使用します。

   ヒント: サンプルのサイレント応答ファイル (`sample_silent_liberty_config.txt`) が、*<agent_install_dir>*`/agent/yndchome/7.3.0.14.08/bin` ディレクトリーに用意されています。このサンプル・ファイルを基に、独自の応答ファイルを作成できます。
2. 次のコマンドを実行して、新規 Docker コンテナーを起動します。**LICENSE** パラメーターを `accept` に設定することで、ライセンスに同意して構成を完了する必要があることに注意してください。

   ```
   $docker run -d -e LICENSE=accept ¥
   -e JAVA_HOME=<java_home_dir> ¥
   -p <port_number>:<port_number> ¥
   -v <web_app_dir>:<liberty_install_dir>/usr/servers/<liberty_profile_name> ¥
   -v <agent_install_dir>/agent/yndchome:<agent_install_dir>/agent/yndchome
   websphere-liberty /bin/bash ¥
   -c "<agent_install_dir>/agent/yndchome/<dcversion>/bin/config.sh -silent
   <absolute_path_to_silent_response_file> && <liberty_install_dir>/bin/server
   run <liberty_profile_name>"
   ```

   ここで、

   - *<java_home_dir>* は、Liberty プロファイルによって使用される JRE のディレクトリーです。例えば、`/opt/ibm/java/jre` です。
   - *<port_number>* は、コンテナーとホストの間の通信に使用されるポート番号です。
   - *<web_app_dir>* は、Web アプリケーションが配置されているディレクトリーです。
   - *<liberty_install_dir>* は、WebSphere Application Server Liberty のインストール・ディレクトリーです。デフォルトは `/opt/ibm/wlp` です。
   - *<liberty_profile_name>* は、Liberty プロファイルの名前です。
   - ここで、*<agent_install_dir>* は WebSphere Applications エージェントのインストール・ディレクトリーです。デフォルトは `/opt/ibm/apm` です。
   - *<dcversion>* は、WebSphere Applications エージェント用のデータ・コレクターのバージョン番号です。例えば、7.3.0.14.08 などです。

- *<absolute_path_to_silent_response_file>* は、作成したサイレント応答ファイルの絶対パスです。

例えば、次のコマンドでは、newitcam という名前の Liberty プロファイル用のデータ・コレクターを構成します。WebSphere Applications エージェントと Liberty プロファイルの両方が、デフォルト・ディレクトリーにインストールされます。モニター・エージェントおよびデータ・コレクターのバージョンは 7.3.0.14.08 です。

```
$docker run -d -e LICENSE=accept ¥
-e JAVA_HOME=/opt/ibm/java/jre ¥
-p 9082:9082 ¥
-v /home/kub/liberty-docker/newitcam:/opt/ibm/wlp/usr/servers/newitcam ¥
-v /opt/ibm/apm/agent/yndchome:/opt/ibm/apm/agent/yndchome websphere-liberty
/bin/bash ¥
-c "/opt/ibm/apm/agent/yndchome/7.3.0.14.08/bin/config.sh -silent
/opt/ibm/wlp/usr/servers/newitcam/silent_config.txt && /opt/ibm/wlp/bin/server
run newitcam"
```

タスクの結果

これで、WebSphere Applications エージェントのデータが Cloud APM コンソールに表示されることを確認できます。「**WAS 情報**」ウィジェットの「**セル名**」列には、Liberty プロファイルが実行されている Docker コンテナーの ID が表示されます。

次のタスク

対話式でデータ・コレクターを構成解除するために、以下のコマンドを使用して、構成解除ユーティリティーを開始します。

```
docker exec -i container_id "<agent_install_dir>/yndchome/7.3.0.14.08/bin
/unconfig.sh"
```

動的クラスター・サーバーをモニターするためのデータ・コレクターの手動構成

動的クラスター・サーバー・インスタンスの作成時に使用したサーバー・テンプレートにデータ・コレクターの構成パラメーターを追加することにより、動的クラスター内のアプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターを構成することができます。これは、動的クラスター・サーバー・インスタンスを構成する方法として、WebSphere Applications エージェントに固有のサーバー・テンプレートを作成する方法に代わるものです。

このタスクについて

動的クラスターのモニターのためにデータ・コレクターを構成するには、2 つの設定ファイルを作成し、次に、WebSphere 管理コンソールで設定を手動で追加して、動的サーバー・テンプレートを変更する必要があります。アプリケーション・サーバー・インスタンス用のデータ・コレクターが開始されると、`runtime` ディレクトリーが自動的に作成されます。サーバー・テンプレートに対して何らかのアップグレードを行うと、この方法で行った変更が消去されることに注意してください。

重要 :

- クラスター名にはスペースを使用できません。
- WebSphere 管理ユーザーとして、データ・コレクターに関する WebSphere Application Server の構成を手動で変更する必要があります。
- データ収集のために WebSphere Application Server に手動で変更を加える作業は、経験豊かな WebSphere 管理者が行う必要があります。手動による構成変更で何らかのエラーが発生すると、アプリケーション・サーバーは開始されないことがあります。
- アプリケーション・サーバー・インスタンスをモニターするようにデータ・コレクターを手動で構成する場合は、構成解除ユーティリティーを 使用してデータ・コレクターを構成解除することはできません。データ・コレクターを構成解除するには、手動で設定の変更を元に戻す必要があります。

手順

1. データ・コレクターの `runtime` ディレクトリーに `dcManualInput.txt` を作成します。
   870 ページの『dcManualInput.txt ファイルの作成』の説明に従ってください。

2. データ・コレクターの wsBundleMetaData ディレクトリーに itcam_wsBundleMetaData.xml ファイルを作成します。872 ページの『itcam_wsBundleMetaData.xml ファイルの作成』の説明に従ってください。
3. WebSphere 管理コンソールを使用して、データ・コレクターの構成パラメーターを追加することにより、動的サーバー・テンプレートを変更します。873 ページの『WebSphere 管理コンソールでの設定の追加』の説明に従ってください。

   ヒント: 動的クラスター・メンバー名は、Cloud APM コンソールで表示される WebSphere Applications エージェント・インスタンス名の中間修飾子として使用されます。エージェント・インスタンス名の長さ制限のため、クラスター・メンバー名が切り捨てられることがあります。この場合、*${MEP_NAME}* という変数を追加し、その値を各サーバー・インスタンスの JVM 名に設定することで、動的サーバー・テンプレートを変更できます。そうすれば、Cloud APM コンソールで実際の JVM 名によって各クラスター・メンバーを区別できます。説明については、876 ページの『オプション: クラスター・メンバーを区別するための実際の JVM 名の表示』を参照してください。

dcManualInput.txt ファイルの作成

このタスクについて

dcManualInput.txt ファイルには、データ・コレクターの初期構成に必要となるいくつかの値が含まれています。

手順

dcManualInput.txt ファイルを作成するには、以下の手順を実行します。

1. 以下のディレクトリーに、*platform*_Template.DCManualInput.txt という名前のファイルが存在するかどうかを確認します。存在しない場合は作成してください。
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime
   - Windows *install_dir*¥dchome¥7.3.0.14.08¥runtime

   ファイル名の *platform* 変数は、オペレーティング・システムのアーキテクチャー (aix32、xLinux64 など) を示します。

   このファイルには、任意の名前を付けることができます。ただし、*platform*_Template.DCManualInput.txt は、configtemplate.sh スクリプトを実行してこのファイルが作成されたときの標準命名規則に従っています。 後続のステップで WebSphere 管理コンソールを使用してサーバー・テンプレート用にこのファイルを指定する必要があります。
2. 前のステップで見つけた、または作成した .txt ファイルに、以下のファイルの内容をコピーします。
   - Linux AIX *dc_home*/itcamdc/etc/was/dcInput_manual.properties
   - Windows *dc_home*¥itcamdc¥etc¥was¥dcInput_manual.properties
3. .txt ファイルの内容を編集します。870 ページの表 226 の説明に従って、このファイルのセクション 1 でパラメーターを設定する必要があります。

   要確認:

   - セクション 2 のパラメーターは変更しないでください。
   - runtime ディレクトリーを作成するためにデータ・コレクターで使用される構成パラメーターの一部は、常に none に設定されます。その理由は、動的クラスターのモニターにおいて、データ・コレクターは、WebSphere サーバー・インスタンスの構成を使用して、JVM 始動時にこのディレクトリーを作成するためです。

表 *226.* セクション *1* の構成パラメーター

パラメーター	値
local.hostname	ローカル・システムの IP アドレスまたは完全修飾ドメイン名。

表 226. セクション 1 の構成パラメーター (続き)	
パラメーター	**値**
was.profile.home	プロファイルのホーム・ディレクトリー。 動的クラスター・サーバーの場合、常に none に設定されます。
was.version	短いバージョン番号。 動的クラスター・サーバーの場合、常に none に設定されます。
itcam.home	データ・コレクターのホーム・ディレクトリー。 例: `/opt/ibm/apm/agent/yndchome/7.3.0.14.08`
was.nodename	ノード名。 動的クラスター・サーバーの場合、常に none に設定されます。
was.servername	サーバー名。 動的クラスター・サーバーの場合、常に none に設定されます。
was.profilename	WebSphere プロファイル名。 動的クラスター・サーバーの場合、常に none に設定されます。
am.camtoolkit.gpe.dc.operation.mode	データ・コレクターの動作モード。有効な値は、`WR`、`TT`、および `DE` の任意の組み合わせです。ここで、 **WR** データ・コレクターを WebSphere Applications エージェントと統合します。 **TT** データ・コレクターを ITCAM for Transactions と統合します。 **DE** データ・コレクターを ITCAM 診断ツールと統合します。このツールは、ITCAM for Application Diagnostics のベータ版でプレビューが提供されています。 必要な動作モードのみを指定してください。例えば、データ・コレクターを WebSphere Applications エージェントのみに接続する場合は、`WR` を指定します。 動作モードが複数ある場合はコンマで区切ります。 例: `am.camtoolkit.gpe.dc.operation.mode=WR, DE`
interp	プラットフォーム・コード。 例: `interp=win64` または `interp=lx6266`

表 226. セクション 1 の構成パラメーター (続き)	
パラメーター	**値**
kwj.serveralias	WebSphere Application Server の別名。 動的クラスター・サーバーの場合、常に none に設定されます。
temagclog.path	(オプション) ガーベッジ・コレクション・ログ・ファイルのパス名。絶対パスを使用して固有のファイル名を入力します。パス名にスペースを入れることはできません。
tema.host	WebSphere Applications エージェントのホスト名または IP アドレス。動作モードに WebSphere Applications エージェント (WR) を含める場合、このパラメーターは必須です。 通常、モニター・エージェントは、データ・コレクターが稼働している各システムにインストールされており、ループバック・アドレスを指定できます。 例: `tema.host=127.0.0.1`
tema.port	WebSphere Applications エージェントとの通信に使用するポート。動作モードに WebSphere Applications エージェント (WR) を含める場合、このパラメーターは必須です。デフォルト値は 63335 です。 例: `tema.port=63335`
tt.connection.string	ITCAM for Transactions のトランザクション・コレクター・コンポーネントのホスト名または IP アドレスおよびポート番号。形式は、`tcp:`*`host_name`*`(IP):`*`port`* です。動作モードに ITCAM for Transactions (TT) を含める場合、このパラメーターは必須です。 例: `tt.connection.string=192.38.234.77:5455`

4. `.txt` ファイルのセクション 1 に以下の行を追加します。

```
bcm.helper=com.ibm.tivoli.itcam.was.bcm.websphere.DefaultWASBCMHelper
BCM_HELPER=@{bcm.helper}
```

5. ファイルを保存して閉じます。

itcam_wsBundleMetaData.xml ファイルの作成

このタスクについて

`itcam_wsBundleMetaData.xml` ファイルには、データ・コレクターの初期構成に必要ないくつかの値が含まれています。

手順

このファイルを作成するには、以下の手順を実行します。

1. 以下のディレクトリーの下に、`wsBundleMetaData` という名前のディレクトリーを作成します。
   - Linux AIX *install_dir*`/yndchome/7.3.0.14.08/runtime`
   - Windows *install_dir*`¥dchome¥7.3.0.14.08¥runtime`
2. `itcam_wsBundleMetaData.xml` という名前のファイルを作成し、以下のファイルの内容をこのファイルにコピーします。

- Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc/etc/was/itcam_wsBundleMetaData_template.xml
- Windows *install_dir*¥dchome¥7.3.0.14.08¥itcamdc¥etc¥was¥itcam_wsBundleMetaData_template.xml

3. itcam_wsBundleMetaData.xml ファイルで、*@{CONFIGHOME}* 変数を、データ・コレクターのホーム・ディレクトリーへの絶対パスに置き換えます。

   各オペレーティング・システムでのデータ・コレクターのホーム・ディレクトリーは、以下のとおりです。

   - Linux AIX *install_dir*/yndchome/7.3.0.14.08
   - Windows *install_dir*¥dchome¥7.3.0.14.08

4. ステップ 1 で作成した wsBundleMetaData ディレクトリーに itcam_wsBundleMetaData.xml ファイルを格納します。

WebSphere 管理コンソールでの設定の追加

手順

以下のステップを実行して、WebSphere 管理コンソールを使用して動的サーバー・テンプレートを変更します。

1. WebSphere 管理コンソールにログインします。
2. 「**サーバー**」をクリックします。
3. 「**クラスター**」を展開し、「**動的クラスター**」を選択します。
4. データ・コレクターを構成する動的サーバー・クラスターの名前をクリックします。
5. 「**追加プロパティー**」セクションで、「**サーバー・テンプレート**」をクリックします。
6. 「**サーバー・インフラストラクチャー**」セクションで、「**Java およびプロセス管理**」を展開し、「**プロセス定義**」をクリックします。
7. 「**追加プロパティー**」のセクションで、「**Java 仮想マシン**」をクリックします。
8. 「**汎用 JVM 引数**」 フィールドに、以下の項目を追加します。

```
-agentlib:am_$jvm-vendor_$jvm-version=${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
$platform_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
-Ditcamdc.dyncluster=true
```

   項目を追加する場合、以下のことに注意してください。

   - すべての項目を 1 行で指定する必要があります。
   - 異なる引数は、- 符号の前にスペースを入れて区切ります。それ以外の場所にはスペースを使用しないでください。
   - 以下の変数は、実際の名前に置き換えてください。
     - *$jvm-vendor*: 使用する JVM ベンダー。
     - *$jvm-version*: JVM バージョン情報。15 (Java 5 ベース)、16 (Java 6 ベース)、17 (7 ベース) などです。
     - *$platform_Template_DCManualInput.txt*: 前のステップで作成した .txt ファイル。

     例:

```
-agentlib:am_ibm_16=${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
```

```
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
aix64_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
–Ditcamdc.dyncluster=true
```

9. 「**適用**」をクリックします。
10. 「メッセージ」ダイアログ・ボックスで、「**保存**」をクリックします。
11. 「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
12. 前のステップに戻って、「**クラスター**」を展開し、「**動的クラスター**」をクリックし、同じサーバー名をクリックします。
13. 「**構成**」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**環境エントリー**」に移動します。
14. オペレーティング・システム、ハードウェア・プラットフォーム、およびアプリケーション・サーバー JVM に応じて、以下の環境エントリーを設定します。

表 *227.* 環境エントリー

プラットフォーム	**環境エントリー名**	**環境エントリーの値**
AIX R6.1 (32 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix533:${ITCAMDCHOME}/toolkit/lib/aix533/ttapi
AIX R6.1 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536:${ITCAMDCHOME}/toolkit/lib/aix536/ttapi
AIX R7.1 (32 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix533:${ITCAMDCHOME}/toolkit/lib/aix533/ttapi
AIX R7.1 (64 ビット JVM)	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536:${ITCAMDCHOME}/toolkit/lib/aix536/ttapi
Linux x86_64 R2.6 (64 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx8266:${ITCAMDCHOME}/toolkit/lib/lx8266/ttapi
Linux Intel R2.6 (32 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx6263:${ITCAMDCHOME}/toolkit/lib/lx6263/ttapi
Linux ppc R2.6 (32 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpp263:${ITCAMDCHOME}/toolkit/lib/lpp263/ttapi

表 227. 環境エントリー (続き)		
プラットフォーム	**環境エントリー名**	**環境エントリーの値**
Linux ppc R2.6 (64 ビット JVM)	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpp266:${ITCAMDCHOME}/toolkit/lib/lpp266/ttapi
Windows (32 ビット JVM)	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win32;${ITCAMDCHOME}/toolkit/lib/win32/ttapi
Windows (64 ビット JVM)	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win64;${ITCAMDCHOME}/toolkit/lib/win64/ttapi

15. NLSPATH という名前の環境エントリーを以下の値に設定します。

```
${ITCAMDCHOME}/toolkit/msg/%L/%N.cat
```

16.「**適用**」をクリックし、「**保存**」をクリックします。
17.「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。
18. 前のステップに戻って、「**クラスター**」を展開し、「**動的クラスター**」をクリックし、同じサーバー名をクリックします。
19.「**構成**」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**Java 仮想マシン**」 > 「**追加プロパティー: カスタム・プロパティー**」に移動します。
20.「**新規**」をクリックし、以下の名前と値のペアを追加し、次に、「**適用**」をクリックします。
    - am.home プロパティーを作成し、その値を以下のディレクトリーに設定します。
        - Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc
        - Windows *install_dir*¥dchome¥7.3.0.14.08¥itcamdc
    - am.orig.wascell プロパティーを作成し、その値をセル・ディレクトリーに設定します。例えば、am.orig.wascell =cellname1 です。
    - com.ibm.tivoli.itcam.toolkit.ai.runtimebuilder.enable.rebuild プロパティーを作成し、その値を true も設定します。
    - ITCAM_DC_ENABLED プロパティーを作成し、その値を true も設定します。
    - TEMAGCCollector.gclog.path プロパティーを作成します。JVM の verlogsegclog 汎用引数が設定されている場合、TEMAGCCollector.gclog.path プロパティーの値を同じ値に設定します。設定されていない場合は、TEMAGCCollector.gclog.path プロパティーを None に設定します。

      ヒント: verlogsegclog プロパティーの値を特定するために、「**構成**」タブで、「**サーバー・インフラストラクチャー**」 > 「**Java およびプロセス管理**」 > 「**プロセス定義**」 > 「**Java 仮想マシン**」をクリックします。値 verlogsegclog は、「**汎用 JVM 引数**」フィールドにあります。
21.「メッセージ」ダイアログ・ボックスで、「**保存**」をクリックします。
22.「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」 を選択します。「**保存**」をクリックします。

- Network Deployment 環境でない場合は、「**保存**」をクリックします。

23. ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックします。
24. 以下の変数を設定します。それぞれの変数について、各種システムのデータ・コレクターのインストール・ディレクトリーに応じて、適切な有効範囲レベルを選択する必要があります。データ・コレクターのインストール・ディレクトリーがシステムによって 異なる場合は、これらの変数は、各ノード・レベルの有効範囲に応じて正しく設定する必要があります。これらすべてが同じインストール・ディレクトリーである場合は、有効範囲を高くする (例えば、クラスター・レベルにする) ことができます。
    - *ITCAMDCHOME* を以下のディレクトリーに設定します。
        - Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc
        - Windows *install_dir*¥dchome¥7.3.0.14.08¥itcamdc
    - *ITCAMDCVERSION* をデータ・コレクターのバージョン (例えば、7.3.0.14.08) に設定します。
25. 「**適用**」をクリックし、「**保存**」をクリックします。
26. 「マスター構成に保存」ダイアログ・ボックスで、以下の手順を実行します。
    - Network Deployment 環境の場合は、必ず「**ノードと変更を同期化**」を選択してから、「**保存**」をクリックします。
    - Network Deployment 環境でない場合は、「**保存**」をクリックします。

    テンプレートの変更後、動的クラスター内のすべてのサーバー・インスタンスと値が同期されます。動的に作成されたすべての新規サーバーにも、同じデータ・コレクター構成パラメーターが設定されます。
27. アプリケーション・サーバー・インスタンスを再始動して、データ・コレクターをアクティブ化します。データ・コレクターによって設定ファイルが読み取られ、runtime ディレクトリーが作成されます。

オプション: クラスター・メンバーを区別するための実際の *JVM* 名の表示

このタスクについて

Cloud APM コンソールで、WebSphere Applications エージェント・インスタンス名の形式は *host_name*::*was_server_name*:KYNS であり、その最大長は 32 文字です。動的クラスター環境では、中間修飾子 *was_server_name* に動的クラスター・メンバー名が使用されます。

文字長制限のため、クラスター・メンバー名が切り捨てられることがあります。この場合、エージェント・インスタンス名の中間修飾子に実際の JVM 名を使用するように指定できます。

手順

以下の手順に従って、エージェント・インスタンス名に実際の JVM 名を表示します。

1. WebSphere 管理コンソールにログインし、以下のように新規環境変数 *${MEP_NAME}* を追加することで、汎用 JVM 引数を更新します。

```
-agentlib:am_$jvm-vendor_$jvm-version=${MEP_NAME}${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
$platform_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
-Ditcamdc.dyncluster=true
```

例:

```
-agentlib:am_ibm_16=${MEP_NAME}${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc
-Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
aix64_Template_DCManualInput.txt
```

```
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
–Ditcamdc.dyncluster=true
```

2. 変更を保存して適用します。
3. ナビゲーション・ペインで、「**環境**」 > 「**WebSphere 変数**」をクリックし、各動的クラスター・メンバーの *${MEP_NAME}* 変数を定義します。値をクラスター・メンバーの実際の JVM 名に設定します。
4. 変更を保存して適用します。
5. アプリケーション・サーバー・インスタンスを再始動します。

   Cloud APM コンソールで、上記で指定した *${MEP_NAME}* 値が含まれている名前を持つ新しい WebSphere Applications エージェント・インスタンスが表示されます。

「エージェント構成」ページでのデータ収集の動的な構成

データ・コレクターでのトランザクション・トラッキングまたは診断データ収集のサポートを有効にした後、 データ収集を動的に有効または無効にするには、「**エージェント構成**」ページを使用します。

始める前に

- Monitoring Agent for WebSphere Applications をインストールして構成する必要があります。
- モニター対象アプリケーション・サーバーのトランザクション・トラッキングを有効または無効にするには、 トランザクション・トラッキングをインストールする必要があります。エージェントでトランザクション・トラッキングのサポートを有効にする必要もあります (841 ページの『対話式でのデータ・コレクターの構成』を参照してください)。 以下の簡易構成手順に従うと、データ・コレクターはトランザクション・トラッキングをサポートするように自動的に構成されます。
- メソッド・トレースなどの診断データの収集を有効または無効にするには、Cloud APM, Advanced が必要です。 841 ページの『対話式でのデータ・コレクターの構成』の説明に従って、データ・コレクターで診断情報およびメソッド・トレース情報を収集するためのサポート機能を有効にする必要もあります。(Cloud APM, Base の場合は使用できません)。

ヒント :「**エージェント構成**」ページには、エージェントによってモニターされるすべてのサーバーが表示されます。欠落しているサーバーがある場合は、そのサーバーが正しくモニターされていない可能性があります。接続エラーなどのエラー・メッセージが出ていないか、モニター対象システムのエージェント・ログ・ファイルを確認してください。

要確認 : WebSphere Applications エージェントでは、Db2 と Oracle のみがデータ・ソースとしてサポートされます。それ以外のタイプのデータ・ソースでは、トランザクション・トラッキング・ダッシュボードおよびグループ・ウィジェットで一部の KPI 値が NULL と表示される場合があります。

手順

各サーバーでデータ収集を構成するには、以下のステップを実行します。

1. ナビゲーション・バーで、 「**システム構成**」 > 「**エージェント構成**」をクリックします。

   「**エージェント構成**」ページが表示されます。
2. 「**WebSphere アプリケーション**」タブをクリックします。
3. データ収集を構成するサーバーのチェック・ボックスを選択し、「**アクション**」リストから、以下のアクションのいずれかを実行します。
   - トランザクション・トラッキングを有効にするには、 「**トランザクション・トラッキングの有効化**」をクリックします。選択した各サーバーの「**現在のトランザクション・トラッキング**」列内の状況が「はい」に更新されます。
   - 診断データの収集のみを有効にするには、 「**診断モードの有効化**」をクリックします。選択した各サーバーの「**現在の診断モード**」列内の状況が「はい」に変わります。
   - 診断データとメソッド・トレース情報の両方を収集するには、 「**診断モードおよびメソッド・トレースの有効化**」をクリックします。選択した各サーバーの「**現在の診断モード**」列および「**現在のメソッド・トレース**」列の状況が「はい」に更新されます。

- 選択したサーバーのトランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの無効化**」をクリックします。選択した各サーバーの「**現在のトランザクション・トラッキング**」列内の状況が「いいえ」に更新されます。
- 選択したサーバーで診断データの収集のみが有効である場合、データ収集を無効にするには、「**診断モードの無効化 (Disable Diagnostic Mode)**」をクリックします。選択した各サーバーの「**現在の診断モード**」列の状況が「いいえ」に更新されます。
- 選択したサーバーで診断データとメソッド・トレース・データの両方が有効になっている場合、データ収集を無効にするには、「**診断モードおよびメソッド・トレースの無効化**」をクリックします。選択した各サーバーの「**現在の診断モード**」列および「**現在のメソッド・トレース**」列の状況が「いいえ」に更新されます。

要確認 :

- データ・コレクターでトランザクション・トラッキングまたは診断データの収集のサポートが構成されていない限り、「**エージェント構成**」ページで操作してもデータ収集は有効にならず、 列の値は「いいえ」に設定されます。
- ホスト名として 127.0.0.1 を使用するようにアプリケーション・サーバー・プロファイルが構成されている場合、「**エージェント構成**」ページ上の「**IP アドレス**」列には、WebSphere Applications エージェントがインストールされて実行されているホストの IP アドレスが表示されます。

タスクの結果

選択した各サーバーのデータ収集を構成しました。トランザクション・トラッキング・データおよび診断データは、データ収集を有効にした後にトポロジー・ダッシュボードに表示できます。

重要 : アプリケーション・サーバーを再始動する場合は、 そのサーバーのトランザクション・トラッキングまたは診断データの収集を再び有効化することが必要な場合もあります。

メモリー・リーク・モニターの有効化

「メモリー分析」ダッシュボードにデータを取り込むには、データ・コレクターのメモリー・リーク・モニターを有効にする必要があります。アプリケーション・サーバーによって使用される JRE がサポートされている場合、メモリー・リーク・モニター機能は、診断データの収集を有効にした後にデフォルトで有効になります。

始める前に

- アプリケーション・サーバーの JVM 引数に -Xtrace:none が定義されていないことを確認します。
- メモリー・リーク・モニターを有効にすると、アプリケーション・サーバーの JVM 引数に以下の設定が定義されます。現在の JVM 引数にこれらの設定を定義済みの場合、データ・コレクター構成によりこれらが変更されても問題がないことを確認してください。

```
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
```

- アプリケーション・サーバーによって使用される JRE が以下のいずれかのバージョンであることを確認します。
  - IBM JRE 1.6.0 SR16 FP3 以降
  - IBM JRE 1.6.1 SR8 FP3 以降
  - IBM JRE 1.7.0 SR8 FP10 以降
  - IBM JRE 1.7.1 SR2 FP10 以降
  - IBM JRE 1.8 以降
  - IBM JRE 1.6.0 SR7 より後のその他のバージョン (APAR IV67574 の iFix が適用済み)

このタスクについて

メモリー・リーク・モニター機能は、IBM JRE の IBM Health Center コンポーネントを必要とします。アプリケーション・サーバーによって使用される JRE がこの機能によってサポートされていることを確認する必要があります。

- AIX または Linux システムでは、現行の JRE がサポートされている場合、診断データの収集を有効にするようにデータ・コレクターを構成すると、構成ユーティリティーは、Health Center コンポーネントが適格であるかどうかを自動的に検査し、適格でなければ Health Center をアップグレードします。
- Windows システムでは、現行の JRE バージョンがサポートされていない場合、構成ユーティリティーは、実行中の JRE のファイルを置換できません。そのため、Health Center コンポーネントを手動でアップグレードする必要があります。

要確認 : 以下の手順は、Windows システムでのみ必要です。AIX または Linux システムの場合、メモリー・リーク・モニターを有効にするには、JRE のバージョンがサポートされていることおよび診断データの収集が有効になっていることを確認するだけで十分です。Solaris システムの場合、IBM JRE の Health Center はサポートされないため、Solaris システムではメモリー・リークを有効にできません。

手順

1. アプリケーション・サーバーによって使用される JRE に組み込まれている IBM Health Center のバージョンを確認します。

   a) コマンド・プロンプトで、JRE ホーム・ディレクトリー内の `bin` ディレクトリーに移動します。

   b) `java -Xhealthcenter -version` と入力し、Enter キーを押します。

   コマンドは、JRE のバージョンおよび IBM Health Center のバージョンを返します。メモリー・リーク・モニター機能は、IBM Health Center 3.0.11 以降を必要とします。
2. IBM Health Center のバージョンが適格でない場合は、IBM Health Center 3.0.11 以降を含むバージョンに JRE をアップグレードします。
3. データ・コレクターの構成ユーティリティーまたは 再構成ユーティリティーを 実行して、診断データの収集を有効にします。
   - データ・コレクターが未構成である場合は、`simpleconfig` または `config` を使用します。
   - データ・コレクターが構成済みである場合は、`reconfig` ユーティリティーを 使用します。

   要確認 : JRE をアップグレードする 前に診断データの収集が有効になっている場合でも、データ収集の構成ユーティリティーを 再実行する必要があります。

PMI の構成

運用モニター・ダッシュボードにパフォーマンス・データを表示するには、パフォーマンス・データを収集するように WebSphere Application Server の Performance Monitoring Infrastructure (PMI) を構成する必要があります。

このタスクについて

WebSphere 管理コンソールを使用して、WebSphere Application Server で PMI を有効にし、PMI レベルを設定します。

PMI には、以下の 4 つのレベルが事前定義されています。

1. なし
2. 基本
3. 拡張
4. すべて

カスタム・オプションを使用すると、個々の統計を選択して有効化または無効化することができます。各レベルには、その下のレベルの統計が含まれます。

運用モニター・ダッシュボードにデータを表示するには、ダッシュボードの計算に使用される属性が、選択されたレベルに含まれている必要があります。

デフォルトでは、WebSphere Applications エージェント により、必要な属性を収集するために十分な PMI レベルが設定されます。

制約事項 : プロセス・サーバーおよびトランザクション・マネージャーのグループ・ウィジェットの中には、それにデータを表示するために、PMI レベルを手動で設定する必要があるものがあります。詳しくは、グループ・ウィジェットの吹き出しヘルプを参照してください。

WebSphere 管理コンソールを使用して PMI レベルを修正する場合は、そのレベルが、必要なデータを収集するのに十分高いレベルであるかどうかを確認する必要があります。

手順

- アプリケーション・サーバー上の PMI を使用可能にするには、以下のステップを実行します。
  a) WebSphere 管理コンソールで、「**モニターおよびチューニング**」を展開してから、「**Performance Monitoring Infrastructure (PMI)**」を選択します。
  b) サーバーのリストから、ご使用のサーバーの名前をクリックします。
  c)「構成」タブをクリックしてから、「**Performance Monitoring Infrastructure (PMI) を使用可能にする (Enable Performance Monitoring Infrastructure (PMI))**」チェック・ボックスを選択します。
  d)「**適用**」または「**OK**」をクリックします。
  e)「**保存**」をクリックして、PMI を使用可能にします。
- アプリケーション・サーバー上の PMI レベルを設定するには、以下のステップを実行します。
  a) WebSphere 管理コンソールで、「**モニターおよびチューニング**」を展開してから、「**Performance Monitoring Infrastructure (PMI)**」を選択します。
  b) サーバーのリストから、ご使用のサーバーの名前をクリックします。
  c)「構成」タブをクリックしてから、使用する統計セットを「基本」、「拡張」、「すべて」、「カスタム」の中から選択します。
  d)「**適用**」または「**OK**」をクリックします。
  e)「**保存**」をクリックして、PMI レベルを設定します。

  各属性に必要な PMI レベルについては、「WebSphere Applications エージェント Reference」の『Dashboard attributes』セクションを参照してください。各属性の収集をオンにしたときに発生するモニター・オーバーヘッドが示されています。

バックアップからのアプリケーション・サーバー構成のリストア

手動で、または構成ユーティリティーかマイグレーション ・ユーティリティーを 使用して、データ収集用にスタンドアロン・アプリケーション・サーバー・インスタンスを構成した場合で、アプリケーション・サーバーが始動しないときは、バックアップからアプリケーション ・サーバー構成をリストアする必要があります。バックアップを作成しなかった場合は、IBM サポートに連絡してください。

このタスクについて

Network Deployment 環境において、手動で、あるいは構成ユーティリティーまたはマイグレーション ・ユーティリティーを 使用して、データ収集用にアプリケーション・サーバー・インスタンスを構成した場合に、アプリケーション・サーバーが始動しないときは、以下のオプションがあります。

- アプリケーション･サーバー構成をバックアップ構成からリストアできます。バックアップを作成しなかった場合は、IBM サポートに連絡してください。
- データ・コレクターを手動で構成解除できます。アプリケーション・サーバー上でデプロイメント・マネージャーとノード・エージェントが稼働している必要があります。詳しくは、156 ページの『手動によるアプリケーション・サーバー・インスタンスからのデータ・コレクター構成の削除』を参照してください。

このセクションは、Windows、UNIX、および Linux の各オペレーティング・システムのみに適用されます。

手順

restoreConfig コマンドを使用してバックアップ構成を適用するには、以下のいずれかの手順を使用します。

- Network Deployment 環境でない場合は、以下の手順を実行します。

a) バックアップ構成ファイルを見つけます。

デフォルトのディレクトリーは *dc_home*/data です。複数のバックアップ・ファイルが存在する場合は、ファイルの変更日時を確認します。変更日時は、失敗した構成の日時でなければなりません。失敗した構成の後、同じホストで他のデータ・コレクター構成を行わなかった場合は、ディレクトリー内の最新のファイルを使用してください。

b) アプリケーション・サーバーのすべてのインスタンスを停止します。

c) *appserver_home*/profiles/*profile_name*/bin ディレクトリーから **restoreConfig** コマンドを実行します。

コマンド構文は以下のとおりです。

– Windows restoreConfig.bat *full_path_to_backup_file*

– Linux AIX ./restoreConfig.sh *full_path_to_backup_file*

restoreConfig コマンドの引数について詳しくは、WebSphere Application Server の Knowledge Center を参照してください。

d) アプリケーション・サーバーのインスタンスを再始動します。

- Network Deployment 環境の場合は以下の手順を実行します。

a) バックアップ構成ファイルを見つけます。

デフォルトのディレクトリーは *dc_home*/data です。複数のバックアップ・ファイルが存在する場合は、ファイルの変更日時を確認します。変更日時は、失敗した構成の日時でなければなりません。失敗した構成の後、同じホストで他のデータ・コレクター構成を行わなかった場合は、ディレクトリー内の最新のファイルを使用してください。

b) アプリケーション・サーバーのすべてのインスタンスを停止します。

c) 適当なパスに一時ディレクトリー (*temp_directory*) を作成します。UNIX システムまたは Linux システムでは、/tmp ディレクトリーの下に作成します。

d) *appserver_home*/profiles/*profile_name*/bin ディレクトリーから restoreConfig コマンドを実行します。

コマンド構文は以下のとおりです。

– Windows restoreConfig.bat *full_path_to_backup_file*

– Linux AIX ./restoreConfig.sh *full_path_to_backup_file*

restoreConfig コマンドを実行すると、元のアプリケーション・サーバー構成が一時ディレクトリーにリストアされます。

e) server.xml、variables.xml、および pmi-config.xml の各ファイルを一時ディレクトリーから Deployment Manager システムにコピーします。

– ソース・ディレクトリー: *temp_directory*/*restored_configuration_home*/cells/*cell_name*/nodes/*node_name*/servers/*server_name*

– ターゲット・ディレクトリー: *appserver_home*/profiles/*profile_name*/config/cells/*cell_name*/nodes/*node_name*/servers/*server_name*

f) 該当のノードに対して、デプロイメント・マネージャーの管理コンソールからノード同期を実行します。

g) デプロイメント・マネージャーの管理コンソールで、変更内容をマスター構成に保存します。

h) アプリケーション・サーバーのインスタンスを始動します。

オンプレミス・アプリケーション用の Liberty データ・コレクターの構成

Linux for System x 上の liberty プロファイルをモニターするために、WebSphere Applications エージェントをインストールすることなく、スタンドアロンのデータ・コレクターをローカルの Liberty ディレクトリーに直接デプロイできます。

始める前に

1. IBM パスポート・アドバンテージ Web サイトから IBM_Data_Collectors_Install.tgz データ・コレクター・パッケージをダウンロードします。 説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。
2. ご使用のファイアウォール・ルールで外部ホストへの透過的アウトバウンド HTTPS 接続が許可されていない場合は、フォワード・プロキシーにトラフィックを送信するようにデータ・コレクターを構成できます。説明については、167 ページの『フォワード・プロキシーを使用して通信するためのデータ・コレクターの構成』を参照してください。
3. データ・コレクターには monitor-1.0 フィーチャーが必要です。**installUtility** コマンドを使用して、Liberty フィーチャー・リポジトリーからこのフィーチャーをダウンロードできます。手順については、WebSphere Application Server Network Deployment Knowledge Center のアセットのダウンロードに関するセクションを参照してください。
4. 「メモリー分析」ダッシュボードにデータを取り込むには、構成中にデータ・コレクターのメモリー割り振り収集を有効にする必要があります。この診断フィーチャーには IBM Health Center 3.0.8 以降が必要です。IBM Health Center のバージョンが適格でない場合は、IBM Health Center 3.0.8 以降を含むバージョンに、アプリケーション・サーバーによって使用される JRE をアップグレードします。

   ヒント: アプリケーション・サーバーによって使用される JRE に組み込まれている IBM Health Center のバージョンを確認するには、JRE ホーム・ディレクトリー内の bin ディレクトリーに移動し、`java -Xhealthcenter -version` を発行します。

このタスクについて

データ・コレクターを手動で構成することも、提供されている構成スクリプトを使用してデータ・コレクターを構成することもできます。

手順

- データ・コレクターを手動で構成するには、データ・コレクター・ファイルをデータ・コレクター・パッケージから取得し、Liberty サーバー用の一部のローカル・ファイルを変更します。

  a) 以下のコマンドを実行して、データ・コレクター・パッケージからファイルを解凍します。

  ```
  tar -xzf IBM_Data_Collectors_Install.tgz
  ```

  liberty_datacollector_8.1.4.0.tgz パッケージは、解凍されたディレクトリーに含まれています。

  b) 次のコマンドを使用して liberty_datacollector_8.1.4.0.tgz パッケージからローカル・ディレクトリーにファイルを解凍します。解凍されたディレクトリーは、データ・コレクターのホーム・ディレクトリーになります。

  ```
  tar -xzf liberty_datacollector_8.1.4.0.tgz
  ```

  例えば、ファイルを /opt/ibm/apm/ ディレクトリーに解凍するには、次のコマンドを発行します。

  ```
  cd /opt/ibm/apm
  tar -xzf liberty_datacollector_8.1.4.0.tgz
  ```

  解凍したファイルは、/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08 ディレクトリーで見つかります。以下のステップでは、このディレクトリーは、データ・コレクターのホーム・ディレクトリー (*dc_home*) として表されます。

  c) Liberty サーバーのホーム・ディレクトリーにナビゲートします。例えば、/opt/ibm/wlp/usr/servers/defaultServer です。

  d) 以下のパラメーターを追加して、jvm.options ファイルを編集します。jvm.options ファイルが存在しない場合は、テキスト・エディターで作成します。

  ```
  -agentlib:am_ibm_16=server_name
  -Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
  -Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
  ```

```
-Dliberty.home=liberty_home
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
-verbosegc
-Xverbosegclog:absolute_path_to_log_file,1,10000
```

項目を追加する場合、以下のことに注意してください。

- 項目は、それぞれ 1 行に収めなければなりません。
- *server_name* を Liberty サーバーの名前に置き換えます。
- *dc_home* をデータ・コレクターのホーム・ディレクトリーに置き換えます。例えば、/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08 です。
- *liberty_home* を Liberty インストール・ディレクトリーのルートに置き換えます。例えば、/opt/ibm/wlp です。
- Liberty サーバーが処理しているワークロードが大きい場合は、-Xmx パラメーターを追加して、データ・コレクターに 512M の追加のヒープ・サイズを割り振ります。例えば、-Xmx1024M です。
- -Xhealthcenter:level=inprocess 行および -Xgc:allocationSamplingGranularity=10000 行はオプションです。この 2 つの行を追加するのは、デフォルトでは無効になっているメモリー割り振り収集を有効にする場合に限ります。「メモリー分析」ダッシュボードにデータを取り込むには、メモリー割り振り収集を有効にする必要があります。
- -Xverbosegclog:*absolute_path_to_log_file*,1,10000 行はオプションです。これは、ガーベッジ・コレクション・ログ・ファイルのリダイレクト先のパスを指定します。指定しない場合、ログは 1 つのファイルに書き込まれ、10000 回の割り振り失敗ごとに循環します。元の stdout または stderr ファイル (console.log) は、サーバーが実行されるに従って非常に大きくなることがあります。ガーベッジ・コレクション出力ログ・ファイルを別のディレクトリーに保存し、ログ・ファイルの数とサイズを制限する場合は、この行を追加します。指定されたパスが無効である場合、この行は無効になり、ガーベッジ・コレクション・ログ・ファイルは stdout または stderr ファイルのままです。

e) 同じディレクトリー内の server.env ファイルを開き、環境エントリーに以下のパスを追加します。server.env ファイルが存在しない場合は、テキスト・エディターで作成します。

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/lib:dc_home/toolkit/lib/lx8266:dc_home/
toolkit/lib/lx8266/ttapi
```

項目を追加する場合、以下のことに注意してください。

- 項目は、それぞれ 1 行に収めなければなりません。
- *dc_home* をデータ・コレクターのホーム・ディレクトリーに置き換えます。例えば、/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08 です。

f) 同じディレクトリー内の server.xml を変更して、以下の行を <featureManager> セクションに追加し、モニター機能を有効にします。

```
<feature>monitor-1.0</feature>
```

g) Liberty サーバーを再始動します。

- プロンプトに応答してデータ・コレクターを構成する場合は、データ・コレクター・パッケージで提供される構成スクリプトを使用します。

a) 以下のコマンドを実行して、データ・コレクター・パッケージからファイルを解凍します。

```
tar -xzf IBM_Data_Collectors_Install.tgz
```

liberty_datacollector_8.1.4.0.tgz パッケージは、解凍されたディレクトリーに含まれています。

b) 次のコマンドを使用して liberty_datacollector_8.1.4.0.tgz パッケージからファイルを解凍します。

```
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

例えば、次の場合があります。

```
cd /opt/ibm
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

解凍されたデータ・コレクター・ファイルは liberty_dc ディレクトリーにあります。

c) liberty_dc/.gdc/7.3.0.14.08/bin ディレクトリーに移動し、次のコマンドを実行して構成スクリプトを開始します。

```
./config_liberty_dc.sh
```

d) プロンプトが出されたら、Liberty インストール・ディレクトリーのルートを入力するかデフォルトを受け入れます。例えば、/opt/ibm/wlp です。

e) プロンプトが出されたら、アプリケーション・サーバーによって使用される JVM のホームを入力するかデフォルトを受け入れます。例えば、/opt/ibm/java です。

f) 構成プログラムは、指定されたディレクトリー内の構成されていないアプリケーション・サーバーを自動的に検出し、リストできます。 構成する Liberty サーバーに対応する番号を入力します。複数のサーバーを選択する場合はそれらの番号をスペースで区切るか、* を入力してすべてを選択します。

g) 構成プログラムがすべての Liberty サーバーでファイルの更新を終了したら、JVM ヒープ・サイズを手動で更新して、データ・コレクターに 512M の追加のヒープを割り振ります。

h) サーバーを再始動して、構成を有効にします。

タスクの結果

データ・コレクターが構成され、Cloud APM サーバーに接続されました。リソース・モニター、トランザクション・トラッキング、および診断のデータが有効になります。 しかし、ヒープ収集およびメモリー割り振り収集は無効になります。「ヒープ・ダンプ」ダッシュボードおよび「メモリー分析」ダッシュボードのデータが必要な場合は、データ・コレクターのプロパティー・ファイルを使用して、それらを有効にすることができます。

次のタスク

- ご使用の Liberty サーバーのモニター・データを表示するには、Cloud APM コンソールを開始します。手順については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

  要確認 : アプリケーション・ダッシュボードに Liberty データ・コレクター・インスタンスを追加する場合は、コンポーネント・リストから「**WebSphere Application Server**」ではなく「**Liberty ランタイム**」を選択します。
- 「ヒープ・ダンプ」ダッシュボードまたは「メモリー分析」ダッシュボード (あるいはその両方) にデータを取り込むには、ヒープ・スナップショット収集またはメモリー割り振り収集 (あるいはその両方) に関してデータ・コレクターを有効にする必要もあります。これは、データ・コレクターの .properties ファイルで行うことができます。894 ページの『トランザクション・トラッキングおよび診断データ収集の有効化または無効化』を参照してください。
- 鍵ファイルまたは Cloud APM サーバーが変更された場合は、データ・コレクターを Cloud APM サーバーに再接続します。説明については、190 ページの『Cloud APM サーバーへのデータ・コレクターの再接続』を参照してください。

オンプレミス・アプリケーション用のデータ・コレクターの構成解除

Liberty サーバーをモニターする必要がない場合、またはデータ・コレクターを新規バージョンにアップグレードする場合は、Liberty サーバーにデプロイされているデータ・コレクターを構成解除する必要があります。

このタスクについて

Liberty サーバーにデプロイされているデータ・コレクターを構成解除するには、データ・コレクターの構成時に加えた変更をロールバックします。データ・コレクターを構成するには、手動で行うか、提供されている unconfig_liberty_dc スクリプトを使用できます。

手順

- データ・コレクターを手動で構成解除するには、以下のステップを実行します。
  a) Liberty サーバーのホーム・ディレクトリーにナビゲートします。例えば、/opt/ibm/wlp/usr/servers/defaultServer です。
  b) jvm.options ファイルを編集して、以下のパラメーターが存在する場合は削除します。

```
-agentlib:am_ibm_16=server_name
-Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
-Dliberty.home=liberty_home
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
-verbosegc
-Xverbosegclog:absolute_path_to_log_file,1,10000
```

  c) 同じディレクトリー内の server.env ファイルを編集して、**LD_LIBRARY_PATH** の以下の値を削除します。

```
/lib:dc_home/toolkit/lib/lx8266:dc_home/toolkit/lib/lx8266/ttapi
```

  ここで、*dc_home* は、データ・コレクターのホーム・ディレクトリーです。例えば、/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08 です。
  d) 同じディレクトリー内の server.xml ファイルを編集して、<featureManager> セクションの <feature>monitor-1.0</feature> を削除します。
  e) Liberty サーバーを再始動します。
- unconfig_liberty_dc.sh スクリプトを使用してデータ・コレクターを構成解除するには、以下のステップを実行します。
  a) *dc_home*/bin ディレクトリーに移動します。例: /opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08/bin
  b) 以下のコマンドを実行して、構成解除スクリプトを開始します。

```
./unconfig_liberty_dc.sh
```

  c) プロンプトが出されたら、Liberty インストール・ディレクトリーのルートを入力するかデフォルトを受け入れます。例えば、/opt/ibm/wlp です。
  d) 構成解除プログラムは、指定されたディレクトリー内の構成されているアプリケーション・サーバーを自動的に検出し、リストできます。構成解除する Liberty サーバーに対応する番号を入力します。複数のサーバーを選択する場合はそれらの番号をスペースで区切るか、* を入力してすべてを選択します。
  e) 構成解除プログラムがすべての Liberty サーバーでファイルの更新を終了したら、変更を有効にするためにそれらのサーバーを再始動します。

次のタスク

データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケ

ーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

データ・コレクターのホーム・ディレクトリーが不要になった場合は、削除することもできます。

IBM Cloud アプリケーション用の Liberty データ・コレクターの構成

IBM Cloud 環境で実行されている Liberty プロファイルをモニターするには、データ・コレクター・パッケージを IBM Marketplace からダウンロードし、データ・コレクターをローカル・アプリケーション・ファイルにデプロイしてから、更新を IBM Cloud にプッシュする必要があります。

始める前に

Cloud Foundry コマンドを使用して Liberty アプリケーションが IBM Cloud 環境にプッシュされていることが想定されます。`manifest.yml` ファイルおよび Liberty サーバーのホーム・ディレクトリー (`server.xml` ファイルを含む) は既に存在しています。

Liberty アプリケーションが WAR ファイルとしてデプロイされている場合、一部のローカル・ファイルを変更して、WAR ファイルとデータ・コレクター・ファイルの両方が格納されているローカル・ディレクトリーをプッシュすることでアプリケーションを 更新する必要があります。以下の例で、WAR ファイルだけを使用している場合に、ローカル Liberty サーバーのホーム・ディレクトリーを取得する方法を説明します。

1. 以下のコマンドを発行して、Liberty アプリケーションをローカルで実行します。

   ```
   mvn install liberty:run-server
   ```

   Liberty WAR ファイルが格納されているディレクトリーに、`/liberty/wlp/usr/servers/defaultServer` というサブディレクトリーが作成されます。このディレクトリーは、以下の手順で Liberty サーバーのホーム・ディレクトリーの役割を果たします。
2. Liberty WAR ファイルが格納されているルート・ディレクトリー内にある *application_name*-SNAPSHOT フォルダー全体を、`/liberty/wlp/usr/servers/defaultServer` ディレクトリーにコピーします。
3. `/liberty/wlp/usr/servers/defaultServer` ディレクトリー内にある `bootstrap.properties` ファイルを編集して、**appLocation** パスを変更します。**appLocation** パスは、IBM Cloud 内のアプリケーション・ディレクトリーを基準とした相対パスに設定する必要があります。
4. `logs` フォルダーおよび `workarea` フォルダーを削除します。これらを IBM Cloud にプッシュする必要はありません。
5. `manifest.yml` ファイル内の **path** 値を、`defaultServer` ディレクトリーを指すように変更します。

   例えば、`path: target/liberty/wlp/usr/servers/defaultServer` です。

手順

Liberty データ・コレクターを構成するには、以下のステップを実行します。

1. `IBM_Data_Collectors_Install.tgz` という名前のデータ・コレクター・パッケージを IBM Marketplace からダウンロードします。詳しい説明については、105 ページの『エージェントのダウンロードおよびデータ・コレクター』を参照してください。
2. 以下のコマンドを実行して、データ・コレクター・パッケージからファイルを解凍します。

   ```
   tar -xzf IBM_Data_Collectors_Install.tgz
   ```

   `liberty_datacollector_8.1.4.0.tgz` パッケージは、解凍されたディレクトリーに含まれています。
3. `liberty_datacollector_8.1.4.0.tgz` パッケージから一時ディレクトリーにファイルを解凍します。

```
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

例えば、次の場合があります。

```
cd /root/tmp
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

解凍したファイルは、一時ディレクトリー内の liberty_dc ディレクトリーで見つかります。

4. liberty_dc ディレクトリーから、server.xml ファイルが保管されている Liberty サーバーのホーム・ディレクトリーに .gdc ディレクトリーをコピーします。以下の手順で、Liberty サーバーのホーム・ディレクトリーは *liberty_server_home* として表されています。

```
cp -rf temp_dir/liberty_dc/.gdc liberty_server_home
```

例えば、次の場合があります。

```
cp -rf /root/tmp/liberty_dc/.gdc /opt/liberty855/wlp/usr/servers/defaultServer/
```

5. liberty_dc/etc ディレクトリーから *liberty_server_home* ディレクトリーに jvm.options ファイルおよび server.env ファイルの内容をコピーするかマージします。

   - jvm.options ファイルおよび server.env ファイルが *liberty_server_home* ディレクトリーにない場合は、これらの 2 つのファイルを *temp_dir*/liberty_dc/etc から *liberty_server_home* にコピーします。

     ```
     cp temp_dir/liberty_dc/etc/jvm.option liberty_server_home
     cp temp_dir/liberty_dc/etc/server.env liberty_server_home
     ```

   - jvm.options ファイルまたは server.env ファイルが *liberty_server_home* ディレクトリー内にある場合は、その内容を *temp_dir*/liberty_dc/etc ディレクトリー内の内容とマージします。

6. ネットワークまたはファイアウォールの設定が原因で、IBM Cloud アプリケーションが Cloud APM サーバーに直接接続できない場合は、データ・コレクターがフォワード・プロキシーを介してトラフィックを送信するように構成します。これを行うには、以下のいずれかの方法で jvm.options ファイルを編集します。

   - 認証が不要な場合は、以下の行をファイルに追加します。

     ```
     -Dhttp.proxyHost=http_proxy_host
     -Dhttp.proxyPort=http_proxy_port
     -Dhttps.proxyHost=http_proxy_host
     -Dhttps.proxyPort=http_proxy_port
     -Djava.net.useSystemProxies=true
     ```

   - フォワード・プロキシー・サーバーにアクセスするためにユーザー名とパスワードが必要な場合は、以下の行をファイルに追加します。

     ```
     -Dhttp.proxyHost=http_proxy_host
     -Dhttp.proxyPort=http_proxy_port
     -Dhttp.proxyUser=http_proxy_user
     -Dhttp.proxyPassword=http_proxy_password
     -Dhttps.proxyHost=http_proxy_host
     -Dhttps.proxyPort=http_proxy_port
     -Dhttps.proxyUser=http_proxy_user
     -Dhttps.proxyPassword=http_proxy_password
     -Djava.net.useSystemProxies=true
     ```

7. Liberty サーバーのホーム・ディレクトリー内の server.xml ファイルを変更して、以下の行を <featureManager> セクションに追加し、モニター機能を有効にします。

```
<featureManager>
        <feature>monitor-1.0</feature>
</featureManager>
```

8. Liberty アプリケーションの `manifest.yml` ファイルを変更し、512M の追加メモリーを割り振ります。
9. コマンド・プロンプトを開き、Liberty サーバーの `manifest.yml` ファイルが格納されているローカル・ディレクトリーに切り替えます。例えば、`/opt/liberty855/` です。
10. IBM Cloud にログインし、**`cf push`** コマンドを使用して Liberty プロファイルを更新します。

タスクの結果
データ・コレクターが構成され、Cloud APM サーバーに接続されました。リソース・モニター、トランザクション・トラッキング、および診断のデータが有効になります。 しかし、ヒープ収集およびメモリー割り振り収集は無効になります。「ヒープ・ダンプ」ダッシュボードおよび「メモリー分析」ダッシュボードのデータが必要な場合は、データ・コレクターのプロパティー・ファイルを使用して、それらを有効にすることができます。

次のタスク

- ご使用の IBM Cloud アプリケーションのモニター・データを表示するには、Cloud APM コンソールを開始します。手順については、『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、『アプリケーションの管理』を参照してください。

  要確認 : アプリケーション・ダッシュボードに Liberty データ・コレクター・インスタンスを追加する場合は、コンポーネント・リストから 「**WebSphere Application Server**」 ではなく 「**Liberty ランタイム**」を選択します。
- 「ヒープ・ダンプ」ダッシュボードまたは「メモリー分析」ダッシュボード (あるいはその両方) にデータを取り込むには、ヒープ・スナップショット収集またはメモリー割り振り収集 (あるいはその両方) に関してデータ・コレクターを有効にする必要もあります。これは、データ・コレクターの `.properties` ファイルで行うことができます。890 ページの『プロパティー・ファイルを使用したデータ・コレクターのカスタマイズ』を参照してください。
- 鍵ファイルまたは Cloud APM サーバーが変更された場合は、データ・コレクターを Cloud APM サーバーに再接続します。説明については、190 ページの『Cloud APM サーバーへのデータ・コレクターの再接続』を参照してください。

Liberty データ・コレクターをカスタマイズするための環境変数
IBM Cloud アプリケーション用に Liberty データ・コレクターをカスタマイズするには、IBM Cloud UI を使用して、そのデータ・コレクターでサポートされている環境変数を追加します。

ヒント : IBM Cloud UI 上で環境変数を追加するには、まず IBM Cloud UI にログインし、ご使用のアプリケーションをクリックしてから、「**ランタイム**」 > 「**環境変数**」をクリックします。「**ユーザー定義**」セクションで環境変数を追加します。

- 888 ページの表 228 にリストされている変数を使用して、Liberty データ・コレクターと Cloud APM サーバーの間の接続を構成します。
- 889 ページの表 229 にリストされている変数を使用して、IBM Cloud アプリケーションのメソッド・トレースを有効または無効にします。
- メソッド・トレースが有効になった後で、889 ページの表 230 にリストされている変数を使用して、さまざまなタイプの要求に対するしきい値を指定します。これにより、さまざまなレベルのモニター・データを収集できます。

要確認 : 環境変数を追加または変更した後で、アプリケーションを再始動して、変更を有効にします。

表 *228.* サーバー接続の環境変数

変数名	可能な値	説明
APM_BM_GATEWAY_URL	• `https://` *`server_ip_or_hostname`*`:443` • `http://` *`server_ip_or_hostname`*`:80`	ターゲットの Cloud APM サーバー・ゲートウェイの URL。

表 228. サーバー接続の環境変数 (続き)		
変数名	**可能な値**	**説明**
APM_KEYFILE_PSWD	鍵ファイルの暗号化されたパスワード	鍵ファイルと対になっている暗号化された鍵ファイル・パスワード。Linux ユーザーの場合は、`echo -n <keyfile password> | base64` コマンドを使用してパスワードを暗号化できます。 **要確認 :** この変数を設定するのは、HTTPS を使用するように *APM_BM_GATEWAY_URL* 変数を構成した場合に限ります。
APM_KEYFILE_URL	`http://hosted_http_server:port/keyfile.jks`	鍵ファイルをダウンロードするための URL。 **要確認 :** この変数を設定するのは、HTTPS を使用するように *APM_BM_GATEWAY_URL* 変数を構成した場合に限ります。

表 229. メソッド・トレースの環境変数		
変数名	**可能な値**	**説明**
METHOD_TRACE_ENABLE	• `true` • `false`	この変数を使用して、メソッド・トレースを有効または無効にします。`true` の値は、メソッド・トレースを有効にします。 デフォルト値は `false` です。

メソッド・トレースを有効にした後、各種要求タイプのしきい値を構成してメソッド・トレースをカスタマイズできます。各要求タイプには以下のしきい値を構成できます。これらのしきい値は、異なるレベルのモニター・データの収集をトリガーします。

1 次しきい値
ある要求タイプに対して 1 次しきい値を構成すると、そのタイプの要求の時間情報が収集されます (その要求タイプの CPU 時間や応答時間など)。これにより、要求が 完了するために 1 次しきい値で指定された時間を超える時間を要した場合は、その要求の時間が収集されます。

2 次しきい値
ある要求タイプに対して 2 次しきい値を構成すると、詳細なコンテキスト・データが収集されます (データベース要求のスタック・トレースや SQL など)。収集されるコンテキスト・データは要求タイプによって異なります。要求が完了するために 2 次しきい値で指定された時間を超える時間を要した場合は、そのコンテキスト・データが収集されます。

それぞれの要求しきい値の環境変数には、*<request_type>_<threshold level>* の形式で名前を付けます。例えば、JMS 要求に対する 1 次しきい値を構成するには、JMS_PRIMARY 変数を追加し、その値を設定します。

889 ページの表 230 に、各種要求タイプに対して追加可能な対応する環境変数をリストします。値の単位はミリ秒です。

表 230. 各種要求しきい値の環境変数	
変数名	**デフォルト値 (ミリ秒)**
SERVLET_PRIMARY	20
SERVLET_SECONDARY	50

表 230. 各種要求しきい値の環境変数 (続き)	
変数名	**デフォルト値 (ミリ秒)**
JDBC_PRIMARY	20
JDBC_SECONDARY	50
JNDI_PRIMARY	20
JNDI_SECONDARY	50
EJB_PRIMARY	20
EJB_SECONDARY	50
WEBSERVICES_PRIMARY	20
WEBSERVICES_SECONDARY	50
APP_METHODS_PRIMARY (アプリケーション・メソッド – 非 J2EE)	50
APP_METHODS _SECONDARY	1000
JCA_PRIMARY	50
JCA_SECONDARY	80
JMS_PRIMARY	40
JMS_SECONDARY	70

プロパティー・ファイルを使用したデータ・コレクターのカスタマイズ

デフォルトでは、データ・コレクターのトランザクション・トラッキングおよびメソッド・トレースが有効になります。ヒープ・スナップショット収集およびメモリー割り振り収集は無効になります。データ・コレクターの .properties ファイルを編集して、データ収集、または診断データが収集される間隔をカスタマイズできます。

このタスクについて

データ・コレクターのプロパティー・ファイルは、*dc_home* ディレクトリー (例えば、/opt/liberty855/wlp/usr/servers/defaultServer/.gdc/7.3.0.14.08) にあります。さまざまなプロパティーを使用して、以下の目的でデータ・コレクターをカスタマイズします。

- トランザクション・トラッキングを有効または無効にします。
- ヒープ・スナップショット収集を有効または無効にします。
- データ・コレクターがヒープ・ダンプのスナップショットを取得する間隔を指定します。
- メモリー割り振りモニターを有効または無効にします。
- データ・コレクターがメモリー割り振り情報を収集する間隔を指定します。
- メソッド・トレースを有効または無効にします。

要確認 : .properties ファイルを変更した後で、**cf push** コマンドを使用して、更新を IBM Cloud 環境にプッシュします。

手順

- トランザクション・トラッキングを有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.dc.bluemix.transaction.enabled** プロパティーを true または false に設定します。

  dc_home/ldc/etc/ldc.properties

トランザクション・トラッキングが有効な場合は、トポロジー内で IBM Java アプリケーション・スタックをモニターできます。

- ヒープ・スナップショット収集を有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.hc.send.heap.enable** プロパティーおよび **com.ibm.tivoli.itcam.hc.snapshot.automatic.enable** プロパティーを true または false に設定します。

  dc_home/healthcenter/etc/hc.properties

  ヒープ・スナップショット収集が有効な場合、データ・コレクターは、指定された間隔でヒープ・スナップショットを取得できます。ヒープ・ダンプ情報を「ヒープ・ダンプ」ダッシュボードで表示できます。
- データ・コレクターによってヒープ・スナップショットが取得される間隔を変更するには、同じファイルで **com.ibm.tivoli.itcam.hc.snapshot.automatic.interval** プロパティーを正の整数に設定します。間隔の単位は分であり、デフォルトは 360 です。

  dc_home/healthcenter/etc/hc.properties
- メモリー割り振り収集を有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.hc.events.collection.automatic.enable** プロパティーを true または false に設定します。

  dc_home/healthcenter/etc/hc.properties

  要確認 : メモリー割り振り収集を有効にする場合は、以下の 2 行を Liberty サーバーの jvm.options ファイルに確実に追加する必要もあります。

  ```
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  ```

  メモリー割り振り収集が有効になった後で、「メモリー分析」ダッシュボードでデータが使用可能になります。
- メモリー割り振り情報が収集される間隔を指定するには、同じファイルで **com.ibm.tivoli.itcam.hc.events.collection.automatic.interval** プロパティーを正の整数に設定します。間隔の単位は分であり、デフォルトは 15 です。

  dc_home/healthcenter/etc/hc.properties
- メソッド・トレースを有効または無効にするには、以下のファイルで **dfe.enable.methoddata** プロパティーを true または false に設定します。

  dc_home/gdc/etc/gdc_dfe.properties

次のタスク

- メソッド・トレースが有効になった後で、環境変数を使用して、さまざまなタイプの要求に対するしきい値を設定できます。これにより、さまざまな要求に対してさまざまなレベルのモニター・データを収集できます。該当する環境変数については、889 ページの表 230 を参照してください。
- メモリー割り振り収集を無効にした場合は、以下の行を Liberty サーバーの jvm.options ファイルから確実に削除してください。

  ```
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  ```

IBM Cloud アプリケーション用のデータ・コレクターの構成解除

IBM Cloud 環境内の Liberty プロファイルをモニターする必要がない場合、またはデータ・コレクターを新規バージョンにアップグレードする 場合は、以前にデプロイしたデータ・コレクターを構成解除する必要があります。

このタスクについて

IBM Cloud 環境内の Liberty プロファイル用のデータ・コレクターを構成解除するには、`jvm.options`、`server.env`、および `server.xml` ファイルに加えられている変更をロールバックしてから、**cf push** コマンドを使用して IBM Cloud 内の Liberty プロファイルを更新します。

手順

1. Liberty プロファイルのローカル・ディレクトリーで、`jvm.options` ファイルを変更して以下のパラメーターを削除します。変更後にファイルが空であれば、ファイルを削除できます。

```
-agentlib:am_ibm_16=defaultServer
-Xbootclasspath/p:../../../../.gdc/7.3.0.14.08/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=../../../../.gdc/7.3.0.14.08/itcamdc/etc/datacollector
.policy
-Dliberty.home=/home/vcap/app/.liberty
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
-verbosegc
-Xverbosegclog:/home/vcap/app/wlp/usr/servers/defaultServer/logs/gc.log,1,10000
```

2. `server.env` ファイルで、**LD_LIBRARY_PATH** 環境変数の以下の値を削除します。変更後にファイルが空であれば、ファイルを削除できます。

```
LD_LIBRARY_PATH=:/lib:../../../../.gdc/7.3.0.14.08/toolkit/lib/lx8266
:../../../../.gdc/7.3.0.14.08/toolkit/lib/lx8266/ttapi
```

3. `server.xml` ファイルを変更し、`<featureManager>` セクションの以下の行を削除することで、monitor-1.0 フィーチャーを削除します。

```
<feature>monitor-1.0</feature>
```

4. Liberty ホーム・ディレクトリー内の `.gdc` ディレクトリーを削除します。
5. コマンド・プロンプトを開き、Liberty サーバーの `manifest.yml` ファイルが含まれているディレクトリーに移動します。
6. IBM Cloud にログインし、**cf push** コマンドを使用して Liberty プロファイルを更新します。

次のタスク

データ・コレクターを構成解除した後も、Cloud APM コンソールには、データ・コレクターの追加先アプリケーションのデータ・コレクターが引き続き表示されます。Cloud APM コンソールには、このアプリケーションには使用可能なデータがないと表示され、データ・コレクターがオフラインであるとは表示されません。アプリケーションおよびリソース・グループからデータ・コレクターを削除する方法について詳しくは、193 ページの『Cloud APM コンソール からのデータ・コレクターの削除』を参照してください。

拡張データ・コレクター構成

データ・コレクターの構成ファイルを変更することで、追加のモニター設定を変更できます。

Liberty データ・コレクターのプロパティー・ファイル

データ・コレクターの構成と動作をさらに制御できるように、各種の構成ファイルが用意されています。

データ・コレクター・パッケージをローカル・ディレクトリーに解凍した後、データ・コレクター・ファイルは、*local_dir*/liberty_dc/.gdc/7.3.0.14.08 ディレクトリーにあります。例えば、/opt/ibm/apm/.gdc/7.3.0.14.08 です。このフォルダーがデータ・コレクターのホーム・ディレクトリーになります。以下の説明では、分かりやすいように *dc_home* として表されます。

データ・コレクターのプロパティー・ファイル

データ・コレクターによってモニターされる個々のアプリケーション・サーバー・インスタンスには、それぞれ独自のプロパティー・ファイルがあります。このプロパティー・ファイルは、データ・コレクターによって自動的に作成されます。ファイルの名前は *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/datacollector.properties です。

今後のアップグレードを容易にするために、このファイルを変更しないでください。

代わりに、データ・コレクター・カスタム・プロパティー・ファイルに、変更する設定を追加してください。このファイルの名前は、*dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/datacollector_custom.properties です。 データ・コレクター・カスタム・プロパティー・ファイルでの設定内容によって、データ・コレクター・プロパティー・ファイルの値がオーバーライドされます。

重要: *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/datacollector_custom.properties ファイルが存在しない場合は、変更を行うときに作成してください。custom ディレクトリーも作成する必要がある場合があります。

ツールキットのプロパティー・ファイル

ツールキットのプロパティー・ファイルは、さまざまな入力ファイルを使用して、始動時にデータ・コレクターによって作成されます。これは、データ・コレクターのモニター対象となるアプリケーション・サーバー・インスタンスごとに固有です。ファイルの名前は、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/toolkit.properties です。

このファイルはデータ・コレクターが始動されるたびに再作成されるため、このファイルには**何も変更を加えないでください**。変更すると、ファイルは上書きされてしまいます。

代わりに、ツールキット・カスタム・プロパティー・ファイルに、変更する設定を追加してください。 このファイルの名前は、*dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties です。 ツールキット・カスタム・プロパティー・ファイルでの設定内容によって、ツールキット・プロパティー・ファイルの値が上書きされます。

このデータ・コレクターのインストール環境でモニターされるすべてのアプリケーション・サーバー・インスタンスを対象にして、ツールキット・プロパティーを設定することもできます。これを行うには、グローバル・ツールキット・カスタム・プロパティー・ファイル *dc_home*/runtime/custom/toolkit_global_custom.properties に設定を追加します。 ただし、プロパティーがインスタンス固有の toolkit_custom.properties ファイルに設定されている場合、その設定がこのインスタンスのグローバル・ファイル内の値をオーバーライドします。

重要: *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties または *dc_home*/runtime/custom/toolkit_custom.properties ファイルが存在しない場合は、変更を行うときに作成してください。custom ディレクトリーも作成する必要がある場合があります。

その他のプロパティー・ファイル

データ・コレクターのプロパティー・ファイルとツールキットのプロパティー・ファイルに加えて、データ・コレクターのモニター対象のアプリケーション・サーバー・インスタンスごとに固有のその他のプロパティー・ファイルがあります。

***dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties**

診断データおよびメソッド・トレース・データの収集に関する詳細を定義します。このファイルの変更について詳しくは、896 ページの『詳細な診断情報収集の構成』を参照してください。

***dc_home*/runtime/*appserver_version.node_name.server_name*/hc.properties**
ヒープ・スナップショット収集およびメモリー割り振り収集に関する詳細を定義します。このファイルの変更について詳しくは、896 ページの『詳細な診断情報収集の構成』を参照してください。

***dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/cynlogging.properties**
データ・コレクターの Java 部分のログ・ファイル名およびロギングの詳細を定義します。

***dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/cyncclog.properties**
データ・コレクターの C++ 部分のログ・ファイル名およびロギングの詳細を定義します。

データ・コレクターのトレース・ファイル

データ・コレクターのトレース・ファイルは、デフォルトで以下の場所に保存されます。

- Windows *dc_home*¥logs¥CYN¥logs
- Linux AIX *dc_home*/logs/CYN/logs

トランザクション・トラッキングおよび診断データ収集の有効化または無効化

デフォルトでは、データ・コレクターのトランザクション・トラッキングおよびメソッド・トレースが有効になります。ヒープ・スナップショット収集およびメモリー割り振り収集は無効になります。データ・コレクターの .properties ファイルを編集して、データ収集、または診断データが収集される間隔をカスタマイズできます。

このタスクについて

データ・コレクターのプロパティー・ファイルは、*dc_home* ディレクトリー (例えば、/opt/ibm/apm/.gdc/7.3.0.14.08) にあります。さまざまなプロパティーを使用して、以下の目的でデータ・コレクターをカスタマイズします。

- トランザクション・トラッキングを有効または無効にします。
- ヒープ・スナップショット収集を有効または無効にします。
- データ・コレクターがヒープ・ダンプのスナップショットを取得する間隔を指定します。
- メモリー割り振りモニターを有効または無効にします。
- データ・コレクターがメモリー割り振り情報を収集する間隔を指定します。
- メソッド・トレースを有効または無効にします。

要確認 : データ・コレクターの構成後に Liberty サーバーを再始動したかどうかに応じて、異なる .properties ファイルが適用されます。データ・コレクターの構成後に Liberty サーバーを再始動した場合は、runtime ディレクトリーが *dc_home* ディレクトリー内に作成されます。その後に、各アプリケーション・サーバーのデータ・コレクターをカスタマイズするには、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name* ディレクトリー内の .properties ファイルのみを使用できます。

手順

- トランザクション・トラッキングを有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.dc.bluemix.transaction.enabled** プロパティーを true または false に設定します。

  dc_home/runtime/*appserver_version.node_name.server_name*/ldc.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/ldc/etc/ldc.properties を使用します)

  トランザクション・トラッキングが有効になった後で、トポロジー内で IBM Java アプリケーション・スタックをモニターできます。
- ヒープ・スナップショット収集を有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.hc.send.heap.enable** プロパティーおよび

com.ibm.tivoli.itcam.hc.snapshot.automatic.enable プロパティーを true または false に設定します。

dc_home/runtime/*appserver_version*.*node_name*.*server_name*/hc.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/healthcenter/etc/hc.properties を使用します)

ヒープ・スナップショット収集が有効になった後で、データ・コレクターは、指定された間隔でヒープ・スナップショットを取得できます。ヒープ・ダンプ情報を「ヒープ・ダンプ」ダッシュボードで表示できます。

- データ・コレクターによってヒープ・スナップショットが取得される間隔を変更するには、同じファイルで **com.ibm.tivoli.itcam.hc.snapshot.automatic.interval** プロパティーを正の整数に設定します。間隔の単位は分であり、デフォルトは 360 です。

  dc_home/runtime/*appserver_version*.*node_name*.*server_name*/hc.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/healthcenter/etc/hc.properties を使用します)
- メモリー割り振り収集を有効または無効にするには、以下のファイルで **com.ibm.tivoli.itcam.hc.events.collection.automatic.enable** プロパティーを true または false に設定します。

  dc_home/runtime/*appserver_version*.*node_name*.*server_name*/hc.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/healthcenter/etc/hc.properties を使用します)

  要確認 : メモリー割り振り収集を有効にする場合は、以下の 2 行を Liberty サーバーの jvm.options ファイルに確実に追加する必要もあります。

  ```
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  ```

  メモリー割り振り収集が有効になった後で、「メモリー分析」ダッシュボードでデータが使用可能になります。
- メモリー割り振り情報が収集される間隔を指定するには、同じファイルで **com.ibm.tivoli.itcam.hc.events.collection.automatic.interval** プロパティーを正の整数に設定します。間隔の単位は分であり、デフォルトは 15 です。

  dc_home/runtime/*appserver_version*.*node_name*.*server_name*/hc.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/healthcenter/etc/hc.properties を使用します)
- メソッド・トレースを有効または無効にするには、以下のファイルで **dfe.enable.methoddata** プロパティーを true または false に設定します。

  dc_home/runtime/*appserver_version*.*node_name*.*profile_name*.*server_name*/custom/gdc/gdc_custom.properties (runtime ディレクトリーが存在しない場合は、*dc_home*/gdc/etc/gdc_dfe.properties を使用します)

タスクの結果

.properties ファイルを変更した後で、Liberty サーバーを再始動して、変更を有効にします。

各アプリケーション・サーバーのデータ・コレクターの .properties ファイルについて詳しくは、892 ページの『Liberty データ・コレクターのプロパティー・ファイル』を参照してください。

次のタスク

- メソッド・トレースが有効になった後で、さまざまなタイプの要求に対するしきい値を設定できます。これにより、さまざまな要求に対してさまざまなレベルのモニター・データを収集できます。説明については、898 ページの『要求のしきい値のカスタマイズ』を参照してください。

- メモリー割り振り収集を無効にした場合は、以下の行を Liberty サーバーの jvm.options ファイルから確実に削除してください。

```
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
```

詳細な診断情報収集の構成

IBM Cloud Application Performance Management, Advanced がある場合、データ・コレクターを使用して、モニター対象のアプリケーション・サーバー・インスタンスに関する詳細な診断情報を収集できます。データ・コレクターが保管する診断情報の量など、診断データ収集の動作を構成するには、gdc_custom.properties 構成ファイルをカスタマイズします。

このタスクについて

gdc_custom.properties ファイルは *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc ディレクトリーにあります。

以下の例では、gdc_custom.properties 構成ファイル内のプロパティーを使用して以下の操作を実行する方法について説明します。

- 詳細情報ファイルのサイズおよび数の制限の設定
- 要求データおよびメソッド診断データの完全または部分的収集の設定

また、gdc_custom.properties ファイルにその他のプロパティーを設定して、診断データの収集をカスタマイズできます。ファイル内にある、プロパティーについて説明しているコメントを参照してください。

要確認 : gdc_custom.properties ファイルの編集が完了した後に、変更を有効にするために、モニター対象アプリケーション・サーバー・インスタンスを再始動する必要があります。

詳細情報ファイルのサイズおよび数の制限の設定

このタスクについて

データ・コレクターは多数のファイルに診断情報を保存します。 デフォルトでは 100 個のファイルを保存します。既に 100 個のファイルが保存されている場合、新しいファイルが作成され、最も古いファイルが削除されます。 データ・コレクターは 15 分ごと、または現在のファイルのサイズが 200 メガバイトを超えると、新しいファイルを作成します。ファイルが格納されているディレクトリーの合計サイズが 2 ギガバイトを超えると、データ・コレクターは最も古いファイルを削除します。

手順

dc_home/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties ファイルで以下の設定を変更できます。

- 診断情報を含むファイルの最大数を設定するには、com.ibm.itcam.gdc.dfe.filelimit プロパティーを設定します。
  例:

  ```
  com.ibm.itcam.gdc.dfe.filelimit=100
  ```

- データ・コレクターが新しい診断データ・ファイルの作成を開始するまでの時間を分単位で設定するには、com.ibm.itcam.gdc.dfe.frequency プロパティーを設定します。
  例えば、次のようにします。

  ```
  com.ibm.itcam.gdc.dfe.frequency=15
  ```

- 診断データ・ファイルの最大サイズ (メガバイト単位) を設定するには、dfe.file.maxlimit プロパティーを設定します。
  例:

```
dfe.file.maxlimit=200
```

現在の診断データ・ファイルがこのサイズに達すると、データ・コレクターは新しい診断データ・ファイルを作成します。

- すべてのデータ・ファイルの最大合計サイズ (バイト単位) を設定するには、trace.dir.size.limit プロパティーを設定します。
  例:

```
trace.dir.size.limit=2147483648
```

すべての診断データ・ファイルの合計サイズがこの値を超えると、データ・コレクターは最も古いデータ・ファイルを削除します。最小合計サイズは 25 メガバイトです。

要求データおよびメソッド診断データの完全または部分的収集の設定

このタスクについて

データ・コレクターには、以下のデフォルト設定がされています。

- データ・コレクターは、選択した要求についてのみ診断データを収集します。要求を選択 (サンプリング) するのは、すべてのエラーと一部の正常な要求を含めるためです。
- メソッド・データ収集は、サーバーの始動時には無効です。
- メソッド・データ収集が有効な場合、データ・コレクターは一部の要求 (診断データの収集対象となる要求) についてのみ メソッド・データを収集します。 ここでの選択 (サンプリング) もすべてのエラーと一部の正常な要求を含めるためです。

重要 : これらの設定は、アプリケーション・サーバーのパフォーマンスに影響します。 実動サーバーで、大きな性能低下が発生することがあります。

手順

dc_home/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties ファイルのプロパティーを使用して、これらの設定を変更できます。

- メソッド・データの収集を有効にするには、プロパティーを以下のように設定します。

```
dfe.enable.methoddata=true
```

ヒント : 「**エージェント構成**」ページを使用して、メソッド・トレース・データの収集を動的に有効または無効にすることもできます。

- すべての要求の診断データを収集するには、プロパティーを以下のように設定してサンプリングを次のように無効化します。

```
dc.sampling.enable=false
```

- 診断データの収集対象であるすべての要求に対してメソッド・データ収集を有効にするには、 プロパティーを以下のように設定します。

```
dc.sampling.enable=false
dc.sampling.methsampler.enabled=false
```

要確認 : dc.sampling.methsampler.enabled プロパティーが有効になるのは、「エージェント構成」ページまたは dfe.enable.methoddata プロパティーでメソッド・データの収集が有効になっている場合のみです。

要求のしきい値のカスタマイズ

デフォルトのしきい値が高いと、一部の要求に十分な情報が含まれない場合があります。データ・コレクターでより多くの要求またはより多くの要求コンテキスト・データを収集できるように、要求のしきい値をカスタマイズすることができます。

このタスクについて

要求タイプにはそれぞれ、**perfThreshold** と **secondaryPerfThreshold** という 2 つのしきい値のタイプがあります。**perfThreshold** しきい値に指定された時間を超える時間を要した要求のみが、データ・コレクターによって収集されます。 その要求が **secondaryPerfThreshold** しきい値に指定された時間を超える時間を要した場合にのみ、スタック・トレースや SQL ステートメントなどのコンテキスト・データが収集されます。これらのしきい値を、要件に合わせて調整することができます。

手順

1. *dc_home*¥gdc¥etc ディレクトリーに移動します。ここで、*dc_home* はデータ・コレクターのホーム・ディレクトリーです。
2. テキスト・エディターで、カスタマイズする要求タイプ用の XML ファイルを開きます。どの XML ファイルがどの要求タイプ用であるかは、XML ファイルの名前からわかります。
   例えば、ejb.xml ファイルは EJB 要求用のファイルであり、 custom.xml ファイルはカスタム要求用、 appMethods.xml ファイルはメソッド・トレースが有効な場合のクラスとメソッド用のファイルです。
3. <collectContextData>、<collectStackTrace>、および <createDataRow> の各タグを、ifThresholdExceeded に設定します。

   ```
   <collectContextData>ifThresholdExceeded</collectContextData>
   <collectStackTrace>ifThresholdExceeded</collectStackTrace>
   <createDataRow>ifThresholdExceeded</createDataRow>
   ```

4. <perfThreshold> タグと <secondaryPerfThreshold> タグを、それぞれの必要なしきい値に設定します。しきい値の単位はミリ秒です。
   例えば、ejb.xml ファイルには、EJB 要求用の以下の設定値が指定されます。これにより、要した時間が 1 秒 (1000 ミリ秒) を超える EJB 要求のみがデータ・コレクターによって収集されます。さらに、EJB 要求が 1.5 秒 (1500 ミリ秒) を超える時間を要した場合にのみ、スタック・トレースや EJB ホームなど、その EJB 要求に関連したコンテキスト・データが収集されます。

   ```
   <requestProbePoint id="EJB">
   <interface>com.ibm.tivoli.itcam.toolkit.ai.boot.aspectmanager.ITurboEJBEventListener</
   interface>
   <family>EJB</family>
   <collectContextData>ifThresholdExceeded</collectContextData>
   <collectStackTrace>ifThresholdExceeded</collectStackTrace>
   <perfThreshold>1000</perfThreshold>
   <secondaryPerfThreshold>1500</secondaryPerfThreshold>
   <dataToCollect>instanceAndSummary</dataToCollect>
   <createDataRow>ifThresholdExceeded</createDataRow>
   <requestType>EJB Method</requestType>
   </requestProbePoint>
   ```

5. 変更を保存してアプリケーション・サーバーを再始動します。

Java EE API のさまざまなタイプのバイトコード・インスツルメンテーションの使用不可化

バイトコード・インスツルメンテーション (BCI) では、データ・コレクターはアプリケーションの 各要求の実行フローを作成するために、さまざまなタイプの Java Platform Enterprise Edition (Java EE) API のメソッドの入り口および出口呼び出しをインターセプトします。モニターするために多少のリソースが使用されます。 データ・コレクターを調整して、一部の API をモニターしないようにすることで、リソースの使用量を削減できます。

Java EE API の BCI モニターを使用不可にするには、ツールキットのカスタム・プロパティー・ファイルに、以下のプロパティーを追加します。このファイルの詳細については、893 ページの『ツールキットのプロパティー・ファイル』 を参照してください。

表 231. ツールキット・カスタム・プロパティー・ファイルへの行の追加

Java EE API のタイプ	toolkit_custom.properties ファイルに追加する行
Enterprise JavaBeans (EJB)	`com.ibm.tivoli.itcam.toolkit.ai.enableejb=false`
Java コネクター・アーキテクチャー (JCA)	`com.ibm.tivoli.itcam.toolkit.ai.enablejca=false`
Java データベース・コネクティビティー (JDBC)	`com.ibm.tivoli.itcam.toolkit.ai.enablejdbc=false`
Java Naming and Directory Interface (JNDI)	`com.ibm.tivoli.itcam.toolkit.ai.enablejndi=false`
Java Message Service (JMS)	`com.ibm.tivoli.itcam.toolkit.ai.enablejms=false`
Web containers for Servlets/JavaServer Pages (JSP)	`com.ibm.tivoli.itcam.dc.was.webcontainer=false`
HTTP セッション・カウント・トラッキング	`com.ibm.tivoli.itcam.toolkit.ai.enablesessioncount=false`
CICS® Transaction Gateway (CTG)	`com.ibm.tivoli.itcam.dc.ctg.enablectg=false`
IMS	`com.ibm.tivoli.itcam.dc.mqi.enableims=false`
Java データ・オブジェクト (JDO)	`com.ibm.tivoli.itcam.dc.mqi.enablejdo=false`
Message Queue Interface (MQI)	`com.ibm.tivoli.itcam.dc.mqi.enablemqi=false`
Axis Web サービス	`com.ibm.tivoli.itcam.toolkit.ai.axis.enablewebservice=false`
リモート・メソッド呼び出し (RMI)	`am.ejb.rmilistener.enable=false`
WebSphere Application Server EJB コンテナー	`com.ibm.tivoli.itcam.dc.was.enableEJBContainer=false`

特定のタイプのトランザクションのトランザクション・トラッキングの無効化

データ・コレクターに対してトランザクション・トラッキングが有効になっている場合、デフォルトではすべてのタイプのトランザクションがモニターされます。ツールキット・プロパティー・ファイルを使用して、特定のタイプのトランザクションのトランザクション・トラッキングを無効にすることができます。

このタスクについて

`toolkit_custom.properties` ファイルを編集して、各アプリケーション・サーバーのトランザクション・トラッキングをカスタマイズするか、`toolkit_global_custom.properties` ファイルを編集して、すべてのアプリケーション・サーバー・インスタンスをカスタマイズします。

`toolkit_custom.properties` ファイルは、単一のアプリケーション・サーバー用に以下の手順で使用されます。プロパティーは、`toolkit_global_custom.properties` ファイルでもサポートされます。ツールキット・プロパティー・ファイルについて詳しくは、『Liberty データ・コレクターのプロパティー・ファイル』を参照してください。

手順

1. アプリケーション・サーバーの toolkit_custom.properties ファイルをテキスト・エディターで開きます。このファイルは、以下のディレクトリーにあります。

   dc_home/runtime/*app_server_version*.*node_name*.*profile_name*.*server_name*/custom
2. 必要に応じて、以下のプロパティーを 1 つ以上指定し、プロパティー値を false に設定して、特定のタイプのトランザクションのトランザクション・トラッキングを無効にします。

   CICS 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.cics.enabled=false

   カスタム要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.ims.enabled=false

   EJB 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.ejb.enabled=false

   HTTP クライアント呼び出しの場合
   com.ibm.tivoli.itcam.dc.ttapi.httpclient.enabled=false

   例外 : Apache HTTP クライアント呼び出しのトランザクション・トラッキングを無効にするには、com.ibm.tivoli.itcam.toolkit.dc.enable.apache.httpclient=false を指定します。

   IMS 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.ims.enabled=false

   JDBC 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.jdbc.enabled=false

   JMS 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.jms.enabled=false

   JNDI 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.jndi.enabled=false

   MQI 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.mqi.enabled=false

   ポータル要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.portal=false

   RMI-IIOP 要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.rmiiiop.enabled=false

   サーブレット要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.arm.servlet.enabled=false

   Web サービス要求の場合
   com.ibm.tivoli.itcam.dc.ttapi.webservice.enabled=false

   ヒント : これらのプロパティーについて詳しくは、*dc_home*/ttdc/etc/ttdc.properties ファイルを参照してください。
3. toolkit_custom.properties ファイルを保存して閉じます。
4. アプリケーション・サーバーを再始動して、変更を有効にします。

モニター対象からのクラスの除外

モニター対象から特定のクラスを除外して、データ収集をカスタマイズできます。このカスタマイズには、ツールキット・プロパティー・ファイルを使用します

このタスクについて

toolkit_custom.properties ファイルを編集して、各アプリケーション・サーバーのトランザクション・トラッキングをカスタマイズするか、toolkit_global_custom.properties ファイルを編集して、すべてのアプリケーション・サーバー・インスタンスをカスタマイズします。

toolkit_custom.properties ファイルは、単一のアプリケーション・サーバー用に以下の手順で使用されます。プロパティーは、toolkit_global_custom.properties ファイルでもサポートされます。ツールキット・プロパティー・ファイルについて詳しくは、『Liberty データ・コレクターのプロパティー・ファイル』を参照してください。

手順

1. アプリケーション・サーバーの toolkit_custom.properties ファイルをテキスト・エディターで開きます。このファイルは、以下のディレクトリーにあります。

   dc_home/runtime/*app_server_version*.*node_name*.*profile_name*.*server_name*/custom
2. ファイルを編集して以下のプロパティーを追加し、変更を保存します。

   ```
   am.camtoolkit.gpe.customxml.exclude=excludes.xml
   ```
3. 同じ custom ディレクトリーに、以下の内容を含む excludes.xml ファイルを作成し、除外するクラス名を <exclude> タグで囲んで指定します。必要に応じて任意の数のクラスを追加でき、アスタリスク・ワイルドカード (*) がサポートされます。

   ```
   <gpe>
     <bci>
       <classExcludes>
         <exclude>class_name_to_be_exclued</exclude>
         <exclude>class_name_to_be_exclued</exclude>
       </classExcludes>
     </bci>
   </gpe>
   ```

   例:

   ```
   <gpe>
     <bci>
       <classExcludes>
         <exclude>org.apache.struts.action.ActionServlet</exclude>
         <exclude>com.company.package.*</exclude>
       </classExcludes>
     </bci>
   </gpe>
   ```
4. アプリケーション・サーバーを再始動します。

次のタスク

クラスが除外されていることを確認するには、toolkit.xml ファイルで、そのクラス名が <classExcludes> セクションにリストされているかどうかを調べます。

要確認 : toolkit.xml ファイルにはランタイム設定が含まれており、アプリケーション・サーバーが再始動されるたびに最新表示されます。

要求情報マッピングのカスタマイズ

場合によっては、エージェントのモニター対象となる要求を識別する情報を変更することが必要になります。この情報には、要求名と、その要求に対して表示できるデータ (例えば、SQL 要求に対する照会テキスト) が含まれます。この情報を変更するには、カスタム要求マッパー構成をセットアップします。

このタスクについて

要求情報のマッピングをカスタマイズするには、カスタム要求マッパー構成を XML ファイルに定義する必要があります。

このファイルでは、いくつかの組み込みシンボル によって、要求のランタイム・コンテキストからの値を表します。新しい値を計算するシンボルを追加で作成することができます。計算には、元の要求値、式、Java メソッドの呼び出し (モニター対象のアプリケーション内のメソッドを含む)、条件、および一連の値の反復を含めることができます。

その後、シンボルの内容を、Cloud APM サーバーに提供される新しい要求データにマップ させることができます。要求データ内の特定の変数がマップされない場合は、元の値が保持されます。

要求タイプに応じて異なるデータが収集されるため、カスタム要求マッパー構成は、要求タイプに固有でなければなりません。 同じデータ・コレクター上の異なる要求タイプに対して異なる要求マッパーを構成することができます。

手順

ある要求タイプに対してカスタム要求マッパー構成を設定するには、以下の手順を実行します。

1. カスタム要求マッパー構成を XML ファイルに定義します。
   XML の構文については、902 ページの『XML ファイルの構文』を参照してください。
2. この XML ファイルを、すべてのアプリケーション・サーバー・インスタンスで使用する場合は、*dc_home*/runtime/custom ディレクトリー内に配置し、1 つのアプリケーション・サーバー・インスタンスで使用する場合は、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom ディレクトリー内に配置します。
3. toolkit_custom.properties または toolkit_global_custom.properties のいずれかのツールキット・カスタム構成ファイル内で、このタイプに対してカスタム要求マッピングを有効にします。
   説明については、912 ページの『要求マッパーの使用可能化』を参照してください。
4. 定義した XML ファイルを、同じツールキット・カスタム構成ファイルから参照します。
   説明については、912 ページの『要求マッパーの使用可能化』を参照してください。

XML ファイルの構文

要求マッパー構成用に作成する XML ファイルは有効な XML でなければならず、構成が使用されているときに使用可能な状態である必要があります。この XML ファイルを、すべてのアプリケーション・サーバー・インスタンスで使用する場合は、*dc_home*/runtime/custom ディレクトリー内に配置し、1 つのアプリケーション・サーバー・インスタンスで使用する場合は、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom ディレクトリー内に配置します。

最上位

最上位タグは <gpe> です。 このタグ内で、<runtimeConfiguration> タグを使用します。これらのタグには属性はありません。

<runtimeConfiguration> タグ内に、<requestMapperDefinition> タグを作成します。このタグは、type 属性を持っていなければなりません。 その属性を、必要な要求タイプの要求マッパー・タイプ名に設定します。詳しくは、914 ページの表 233 を参照してください。

<requestMapperDefinition> タグ内に、次の 2 つのタグが存在していなければなりません。

<symbolDefinitions>
シンボルのすべての定義が含まれます。 シンボルは、このタイプの要求が検出されるたびにエージェントが計算する値を表します。

<selection>
コンテキスト・キーと値とのマッピングが含まれます。キーは、エージェントに渡されるカスタム・データを表します。キーは、要求タイプごとに事前定義されています。マッピングは条件付きにすることができます。

要求マッパーの使用可能化プロパティーおよびタイプ名について詳しくは、914 ページの表 233 を参照してください。

また、<runtimeConfiguration> タグ内に <requestMapperClassPath> タグを作成することもできます。このタグ内で、JAR ファイルを定義できます。要求マッパー定義内のこれらの JAR ファイルで、Java クラスを参照することができます。

式の定義
シンボルを定義するには、式を使用する必要があります。エージェントは式を評価して、シンボルに値を割り当てます。

式の中でのデータの使用

式では、以下のデータを使用できます。

- その要求タイプの入力データ・シンボル
- 同じ要求マッパー定義に記述された他のシンボル
- 数値定数
- ストリング定数 (「"」で区切られます。例えば、"string")
- ブール定数 (true、TRUE、false、FALSE)
- null 定数

入力データ・シンボルについて詳しくは、915 ページの表 234 を参照してください。

シンボルの値が Java クラスのインスタンス化である場合、式には、そのクラス内に定義されているフィールドおよびメソッドへの参照を含めることができます。フィールドを参照するには、*symbol.fieldname* を使用します。 メソッドを参照するには、*symbol.methodname*(*parameters*) を使用します。 メソッド呼び出しは、値を返す必要があります。例えば、ストリング値を持つシンボルで Java ストリング・メソッドを使用できます。

クラスの静的フィールドまたはメソッドを 参照するには、*classname.fieldname* および *classname.methodname*(*parameters*) も使用できます。

シンボルが配列オブジェクトを参照する場合、式でエレメントを選択して (*symbol*[*selector*])、配列の長さ (*symbol*.length) を決めることができます。

演算子

式の中では以下の演算子を使用できます。

- ブール演算子: AND、&、OR、|、NOT、!
- 比較: ==、!=、GT、>、LT、<、GE、>=、LE、<=
- 数値演算子: +、 -、 *、 /
- 評価の順序を強制するための括弧: (、)

重要 : XML 内では、シンボル <、>、および & をエスケープする必要があります。代わりに、GT (より大)、GE (より大か等しい)、LT (より小)、LE (より小か等しい)、および AND の演算子を使用できます。

式では、次のように instanceof 演算子を使用して、値がクラスのインスタンスであるかどうかを評価することができます。

```
expression instanceof java.class.name
```

この演算子は、Java instanceof 演算子に似ており、ブール値を生成します。この例では、*expression* 値が属するクラスが以下のいずれかの条件に適合する場合、値は TRUE です。

- 指定された *java.class.name* である
- *java.class.name* で識別されたクラスの直接的または間接的なサブクラスである。
- *java.class.name* で識別されたインターフェースを 直接的または間接的に実装する。

式では、new 演算子を使用して、Java クラスの新規オブジェクトをインスタンス 化することもできます。この演算子は、Java new 演算子に似ており、次のようになります。

```
new java.class.name(expression1, expression2, ... expressionN)
```

演算子優先順位
演算子は優先順に評価されます。同じ優先順位の演算子は、左から右に評価されます。括弧 (と) を使用して、評価の順序を変更できます。

優先順位は以下のとおりです。

1. . 演算子 (メソッド呼び出しまたはフィールド参照)
2. [ ] (配列エレメント・セレクター)
3. new
4. !、NOT
5. *, /
6. +, -
7. GT、>、LT、<、GE、>=、LE、<=、instanceof
8. ==, !=
9. AND、&
10. OR、|

例

```
$s1 >= ( 2 * ($s2.sampMethod($s3, true) + 1))
```

エージェントは、以下の方法でこの式を評価します。

1. $s1 シンボルが評価されます。これは数値を生成する必要があります。
2. $s2 シンボルが評価されます。これは Java オブジェクトを生成する必要があります。
3. $s3 シンボルが評価されます。
4. $s2 の評価から得られたオブジェクトに対して、sampMethod メソッドが呼び出されます。$s3 の評価の結果が最初のパラメーターとして渡され、ブール値 true が 2 番目のパラメーターとして渡されます。sampMethod への呼び出しは、数値を返す必要があります。
5. ステップ 904 ページの『4』 の結果に 1 が加算されます。
6. ステップ 904 ページの『5』 の結果に 2 が乗算されます。
7. ステップ 904 ページの『1』 の結果が、ステップ 904 ページの『6』 の結果と比較されます。ステップ 904 ページの『1』 の結果が、ステップ 904 ページの『6』 の結果より大か等しい場合、true が返されます。それ以外の場合は、false が返されます。

基本シンボルの定義
シンボルを定義するには、式を使用する必要があります。エージェントは式を評価して、シンボルに値を割り当てます。

<symbol> タグ内で、以下のタグを使用します。

<name>
シンボルの名前。これはストリングで、$ 文字で始める必要があります。

<eval>
エージェントがこのシンボルの値を生成するために評価する必要がある式。式の定義について詳しくは、903 ページの『式の定義』を参照してください。

<type >
シンボルが返す値のタイプ。この値は、完全修飾 Java クラス名、つまり Java プリミティブとして指定します。シンボルのタイプの指定はオプションです。これが定義されていない場合、要求マッパーは、式に基づいてフィールド・タイプを設定しようとします。 要求マッパーが式を評価する前にシンボルのタイプを判別できないと、パフォーマンスに影響が出ます。そのため、最良のパフォーマンスを得るために、タイプを指定することをお勧めします。

<args>
シンボルの引数。このタグはオプションです。これを指定した場合、そのシンボルを評価するために引数を提供する必要があります。詳しくは、905 ページの『シンボルの引数の定義』を参照してください。

例

```
<symbol>
   <name>$doubles1</name>
   <eval>$s1*2</eval>
   <type>int</type>
</symbol>
```

このシンボルは、別のシンボル $s1 の値の 2 倍を返します。

シンボルの引数の定義

シンボル定義の <args> タグ内で、そのシンボルの引数のタイプを定義できます。

このタグでは、<type> タグを使用して引数のタイプを指定します。この値は、完全修飾 Java クラス名、つまり Java プリミティブとして指定します。<type> タグはいくつでも指定できます。それぞれのタグで 1 つの引数を定義します。

この場合、次のように引数を括弧で囲んで、シンボルを参照する必要があります。

```
$symbol(argument1,argument2...)
```

引数の個数は、引数タイプの定義の数と同じでなければなりません。

シンボル定義内で、最初の引数は $p0、2 番目の引数は $p1 というように参照します。

引数を持つシンボルは、Java メソッドと同様に機能します。 シンボルは、入力引数を取り、引数の値に応じた値を返します。

例

```
<symbol>
   <name>$double</name>
   <eval>$p0*2</eval>
   <type>int</type>
   <args>
       <type>int</type>
   </args>
</symbol>
```

このシンボルは、引数の値の 2 倍を返します。これを評価するには、数値引数 ($double(2)、$double($s1)) を提供します。

反復シンボルの定義

<symbolDefinitions> タグ内で、<iterationSymbol> タグを使用して反復シンボルを定義することができます。反復シンボルは、Java 配列、列挙、または集合内の一連のオブジェクトにわたって評価を繰り返すことによって得られる値を表します。各メンバーについて、要求マッパーは 1 つ以上の条件式を評価します。式が true を返した場合、要求マッパーはそのメンバーを使用して戻り値を計算します。1 つのメンバーが条件式を満たすと、要求マッパーは残りのメンバーを評価しません。

<iterationSymbol> タグ内で、以下のタグを使用します。

<name>
シンボルの名前。これはストリングで、$ 文字で始める必要があります。

<type >
シンボルが返す値のタイプ。この値は、完全修飾 Java クラス名、つまり Java プリミティブとして指定します。シンボルのタイプの指定はオプションです。これが定義されていない場合、要求マッパーは、式に基づいてフィールド・タイプを設定しようとします。 要求マッパーが式を評価する前にシンボルのタイプを判別できないと、パフォーマンスに影響が出ます。そのため、最良のパフォーマンスを得るために、タイプを指定することをお勧めします。

<args>
シンボルの引数。このタグはオプションです。これを指定した場合、そのシンボルを評価するために引数を提供する必要があります。詳しくは、905 ページの『シンボルの引数の定義』を参照してください。

<iterate over="*expression*">
繰り返し評価されるメンバーを含んでいるオブジェクト (配列、列挙、または集合) を定義します。式は、そのようなオブジェクトを返す必要があります。要求マッパーは、メンバーの 1 つで条件式が `true` を返すまで、あるいは残りのメンバーがなくなるまで、メンバー上で評価を繰り返します。このタグ内の以下のタグで、一連の反復式を定義します。

<test>
このタグ内で、条件式と戻り式を定義します。1 つの <iterate> タグに複数の <test> タグを含めることができます。その場合、要求マッパーはすべてのタグを評価します。いずれかの条件式が TRUE の場合、シンボルは同じ <test> タグ内の結果式を使用して値を返し、それ以上は評価を実行しません。

<castTo>
オプション。このタグが存在する場合、このタグ内で Java タイプの名前を、完全修飾 Java クラス名、つまり Java プリミティブとして指定します。 要求マッパーは、繰り返し評価されるエレメントをこのタイプにキャストしてから、条件式および戻り式を評価します。このタグが存在しない場合、要求マッパーは、配列のメンバーを配列の基本タイプにキャストし、列挙または集合のメンバーを `java.lang.Object` にキャストします。配列のメンバーの場合は通常、配列の基本タイプが正しい選択です。したがって、このタグは要求マッパーが列挙または集合で評価を繰り返す場合に使用してください。

<condition>
ブール値を生成する必要がある式。繰り返されるエレメントを参照するには、`$iterElement` を使用します。

<return>
<condition> タグ内の式が `true` を返した場合、要求マッパーは <return> タグ内の式を評価します。 反復シンボルは、この式が生成する値を返します。 繰り返されるエレメントを参照するには、`$iterElement` を使用します。

<defaultValue>
オプションです。 要求マッパーがオブジェクトのすべてのメンバー 上で繰り返し実行されたが、どの条件式も `true` を返さなかった場合、要求マッパーは <defaultValue> タグ内の式を評価します。 反復シンボルは、その式が生成する値を返します。 このタグが存在しない場合、デフォルト値は `null` です。

例

```
<iterationSymbol>
  <name>$userNameCookieValue</name>
  <iterate over="$httpServletRequest.getCookies()">
     <test>
        <condition>$iterElement.getName().equals("userName")</condition>
        <return>$iterElement.getValue()</return>
     </test>
   </iterate>
</iterationSymbol>
```

このシンボルは、「username」という名前の Cookie を検索し、その値を返します。
`$httpServletRequest.getCookies()` は配列を返すため、<castTo> エレメントは不要です。

```
<iterationSymbol>
  <name>$headerNameStartingWithA</name>
  <iterate over="$httpServletRequest.getHeaderNames()">
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.startsWith("A")</condition>
       <return>$iterElement</return>
    </test>
```

```
  </iterate>
</iterationSymbol>
```

このシンボルは、名前が「A」で始まるヘッダーを検索し、その名前を返します。
$httpServletRequest.getHeaderNames() は列挙を返すため、<castTo> エレメントが必要です。

```
<iterationSymbol>
  <name>$determined_gender</name>
  <iterate over="$children">
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.equals("male")</condition>
       <return>"It's a boy"</return>
    </test>
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.equals("female")</condition>
       <return>"It's a girl"</return>
    </test>
  </iterate>
  <defaultValue>"unknown"</defaultValue>
</iterationSymbol>
```

このシンボルは $children 上で繰り返し実行され、これはストリングの配列、列挙、または集合でなければなりません。いずれかのストリングが「male」と等しい場合、シンボルは「it's a boy」を返します。いずれかのストリングが「female」と等しい場合、シンボルは「it's a girl」を返します。最後に、$children オブジェクト内に「male」にも「female」にも等しいストリングがない場合、シンボルは「unknown」を返します。

条件シンボルの定義
<symbolDefinitions> タグ内で、<conditionalSymbol> タグを使用して条件シンボルを定義することができます。条件シンボルは、一連の条件式の評価によって得られる値を表します。いずれかの式が true を返した場合、要求マッパーはそのメンバーを使用して戻り値を計算します。1 つのメンバーが条件式を満たすと、要求マッパーは対応する戻り式を評価し、結果を返します。返す結果が見つかると、要求マッパーはそれ以上式を評価しません。

<conditionalSymbol> タグ内で、以下のタグを使用します。

<name>
シンボルの名前。これはストリングで、$ 文字で始める必要があります。

<type >
シンボルが返す値のタイプ。この値は、完全修飾 Java クラス名、つまり Java プリミティブとして指定します。シンボルのタイプの指定はオプションです。これが定義されていない場合、要求マッパーは、式に基づいてフィールド・タイプを設定しようとします。 要求マッパーが式を評価する前にシンボルのタイプを判別できないと、パフォーマンスに影響が出ます。そのため、最良のパフォーマンスを得るために、タイプを指定することをお勧めします。

<args>
シンボルの引数。このタグはオプションです。これを指定した場合、そのシンボルを評価するために引数を提供する必要があります。詳しくは、905 ページの『シンボルの引数の定義』を参照してください。

<if condition="*expression*">
condition 属性は、評価する条件式を定義します。この式は、ブール値を生成する必要があります。値が true の場合、要求マッパーは <if> タグの内容を使用して、戻り値を判別しようとします。<if> タグには、以下の内容のいずれか (ただし、両方ではない) が入っている必要があります。

- <return> タグ。このタグには式が入ります。条件式が TRUE の場合、要求マッパーは式を評価して、その結果を返します。
- この <if> タグ内にネストされた任意の数の <if> タグ。条件式が TRUE の場合、要求マッパーは、最上位の <if> タグと同じ方法で、ネストされた <if> タグを処理します。 つまり、condition 属性内の式を評価して、式が true の場合は、そのタグの内容を使用して戻り値を決定しようとします。

重要 : 戻り値が判別された場合、要求マッパーはそれ以上の式を評価しません。ただし、<if> タグ内の条件式は TRUE であるが、そのタグにはネストされた <if> タグが含まれており、それらの条件式のいずれも TRUE でない場合には、値は判別されません。この場合、要求マッパーは後続の式の評価を続けます。

<defaultValue>
オプションです。 要求マッパーがすべての条件式を評価したが、どの条件式も `true` を返さなかった場合、要求マッパーは <defaultValue> タグ内の式を評価します。条件シンボルは、その式が生成した値を返します。このタグが存在しない場合、デフォルト値は `null` です。

例

```
<symbol>
  <name>$GET</name>
  <eval>"GET"</eval>
</symbol>
<symbol>
  <name>$PUT</name>
  <eval>"PUT"</eval>
</symbol>
<conditionalSymbol>
  <name>$sessionAttribute</name>
  <if condition="$httpServletRequest.getSession(false) != null>
     <if condition="$httpServletRequest.getSession(false).getAttribute($GET)
!= null">
       <return>$httpServletRequest.getSession(false).getAttribute($GET)</return>
     </if>
     <if condition="true">
       <return>$httpServletRequest.getSession(false).getAttribute($PUT)</return>
      </if>
  </if>
</conditionalSymbol>
```

このシンボルは、サーブレットの要求マッパーの一部であると想定されています。最初に、シンボルはサーブレットの HTTP セッションが存在するかどうかをチェックします。存在しない場合、シンボルは `null` を返します。セッションが存在する場合、シンボルはサーブレットが GET 属性を持っているかどうかをチェックし、その属性の値を返します。GET 属性がない場合、シンボルは PUT 属性の値を返します。2 番目の条件式は true です。この値は else 節として使用されます。最初の条件が TRUE の場合、要求マッパーはそれ以上の式を評価しません。それ以外の場合は、2 番目の式の評価を続けます。

外部クラス・シンボルの定義
<symbolDefinitions> タグ内で、<externalClassSymbol> タグを使用して外部クラスを定義することができます。外部クラス・シンボルは、外部 Java クラスを表します。 外部クラス・シンボル定義はオプションです。外部 Java クラスを式の中で直接使用することができます。ただし、外部クラス・シンボルを使用すると、要求マッパー構成が読みやすくなる場合があります。

<externalClassSymbol> タグ内で、以下のタグを使用します。

<name>
シンボルの名前。これはストリングで、$ 文字で始める必要があります。

<className >
ユーザー定義クラスの名前。

重要 : 要求マッパー構成内で Java クラスを参照するには、それが外部クラス・シンボル定義の中であるか、式の中であるかに関係なく、そのクラスが入っている JAR ファイルの絶対パスと名前を、<runtimeConfiguration> タグ内の <requestMapperClassPath> タグに追加する必要があります。

外部シンボルを定義した後は、クラスをシンボルの名前で参照することができます。また、クラスの静的メソッドおよびフィールドも 、シンボルを使用して参照できます。

例

```
<externalClassSymbol>
  <name>$rand</name>
  <className>user.class.Random</className>
</externalClassSymbol>
```

このシンボルは、ユーザーが作成したクラスを参照し、乱数を生成します。このクラスが入っている JAR ファイルの絶対パスと名前が、<runtimeConfiguration> タグ内の <requestMapperClassPath> タグに存在しなければなりません。

式の中で静的メソッド `user.class.Random.generate()` を参照するには、次の外部シンボルを使用できます。

```
$rand.generate()
```

コンテキスト・キーへの値のマッピング
<requestMapperDefinition> タグ内で、<selection> タグを使用して値をコンテキスト・キーにマップします。このマッピングにより、モニター情報の変更内容が提供されます。

要求タイプに定義された出力キーに値をマップさせることができます。詳しくは、914 ページの表 233 を参照してください。

要求マッパー構成の評価後に、キーに値がマップされていない場合、ITCAM は要求から抽出された元の値を使用します。

<selection> タグ内で、以下のタグを使用します。

<matchCriteria>
ブール値を返す必要がある式。このタグ内で定義されたマッピングは、この式が `true` を返した場合にのみ使用されます。

<mapTo>
キーとそのキーにマップする値を定義します。このタグ内で、<key> タグにはキーが入り、<value> タグには値が入ります。

<selection>
1 つのタグの中に別のタグを入れて、<selection> タグをネストすることができます。

<selection> タグがネストされている場合は、外部 <matchCriteria> 式とネストされた <matchCriteria> 式の両方が `true` を返したときにのみ、ネストされたマッピングが使用されます。

1 つの <requestMapperDefinition> タグ内または別の <selection> タグ内で、複数の <selection> タグを使用できます。同じネスティング・レベルの複数の <selection> タグ内 (つまり、同じ親タグ内) に同じキーが複数回マップされている場合、<matchCriteria> 式が `true` を返した最初のマッピングが使用されます。

外部 <selection> タグとネストされた <selection> タグの両方に同じキーをマップしないでください。

通常は、`true` の <matchCriteria> 値を、ネスティング・レベルの最後の selection タグの「else」値として使用します。異なるケースに異なる値をマップしたい場合は、この外部タグ内で複数の <selection> タグを使用します。それぞれのタグに、特定のケース用の基準と値を含めることができます。`true` の値を持つ最後のタグは、使用可能なデータがどの基準も満たさない場合に適用されます。

例

```
<selection>
   <matchCriteria>true</matchCriteria>
   <mapTo>
      <key>Result</key>
      <value>$s1</value>
   </mapTo>
</selection>
```

このマッピング構成では、`Result` はシンボル `$s1` の値に設定されます。

```
<matchCriteria>true</matchCriteria>
   <selection>
      <matchCriteria>$b1</matchCriteria>
      <mapTo>
         <key>Result</key>
         <value>1</value>
      </mapTo>
   </selection>
   <selection>
      <matchCriteria>true</matchCriteria>
```

```
      <mapTo>
         <key>Result</key>
         <value>2</value>
      </mapTo>
</selection>
```

このマッピング構成では、シンボル $b1 はブール値を返さなければなりません。Result は、$b1 が true を返した場合は 1 に設定され、$b1 が false を返した場合は 2 に設定されます。$b1 が true を返した場合、要求マッパーは最初の <selection> タグ内の Result のマッピングを使用します。2 番目のタグ内の同じキーのマッピングは使用されません。

カスタム要求の定義

デフォルトでは、特定のタイプの Java クラスおよびメソッドのみがデータ・コレクターによって要求としてモニターされます。サーブレット、JSP、EJB ビジネス・メソッド、および特定の標準 Java EE API が、要求として認識されます。追加のクラスおよびメソッドをカスタム要求 として指定することができます。

このタスクについて

データ・コレクターによるカスタム要求のモニターを有効にするには、カスタム要求を XML ファイルに定義し、ツールキット・カスタム・プロパティー・ファイルに am.camtoolkit.gpe.customxml.custom プロパティーを設定します。

例えば、デフォルトでは、データ・コレクターは、Struts Action クラスを要求として認識しません。ただし、カスタム要求定義をセットアップすることにより、そのアクションをネストされた要求として認識させることができます。

手順

カスタム要求のモニターを使用可能にし、1 つまたは複数のメソッドをカスタム要求として指定するには、以下の手順を実行します。

1. 一時的なロケーションに *dc_home*/itcamdc/etc/custom_requests.xml ファイルのコピーを作成します。次に、コピーをテキスト・エディターで開きます。
2. ファイル内のパラメーターを変更します。

   以下の表は、変更できるパラメーターについて説明しています。

表 *232.* カスタム要求構成ファイルのパラメーター

タグ名	説明
edgeRequest	カスタム要求の処理にバイトコード・インスツルメンテーションを適用する 1 つ以上のアプリケーション・メソッドを識別します。 edgeRequest タグ内の requestName、Matches、type、methodName の各タグを変更することにより、選択をカスタマイズできます。 それぞれの edgeRequest タグは、正確に 1 つの methodName タグと、1 つ以上の Matches タグが含まれている必要があります。複数の edgeRequest タグを指定できます。
requestName	この要求の固有の名前を定義します。メソッドの入り口および出口がトレースされると、要求名がユーザーに表示されます。
Matches	カスタム要求の処理にバイトコード・インスツルメンテーションを適用するメソッドを含んでいる 1 つ以上のクラスを識別します。 単一の edgeRequest タグ内に複数の Matches タグを使用できます。
type	edgeRequest タグに一致するために、クラスがシステム・クラスとアプリケーション・クラスのどちらであることが必要かを示します。
methodName	Matches タグによって識別されたいずれかのクラス内にあり、カスタム要求の処理にバイトコード・インスツルメンテーションを適用するメソッドの名前を識別します。それぞれの edgeRequest タグの中で、正確に 1 つの methodName タグを指定できます。

表 232. カスタム要求構成ファイルのパラメーター (続き)	
タグ名	**説明**
requestMapper	オプションです。 このタグを指定した場合、データ・コレクターは、要求マッパーを使用して、要求を識別する情報を判別します。この情報を抽出する非標準の方法を定義することが可能です。 要求マッパーの使用可能化および定義について詳しくは、901 ページの『要求情報マッピングのカスタマイズ』を参照してください。
要確認 : Matches タグおよび methodName タグは、ワイルドカード文字を含めることができます。ワイルドカード文字の機能を以下に示します。 • アスタリスク (*) は、単独で使用された場合、任意の文字のゼロ個以上のオカレンスを表します。文字のシーケンスの中に組み込まれている場合は (例えば、java.*.String)、パッケージ分離文字 (.) を除く任意の文字のゼロ個以上のオカレンスに一致します。 • 2 つのピリオド (..) を使用して、すべてのサブパッケージを指定できます。これは、パッケージ分離文字 (.) で始まって終わる任意の文字シーケンスに一致します。例えば、java..String は java.lang.String に一致し、com.ibm..* は com.ibm で始まる任意の宣言に一致します。	

例えば、com.mycompany.myapp というパッケージ名が付いたアプリケーションには、以下の要件があります。

- Customer クラス内の creditCheck() メソッドは、CreditCheck と呼ばれるカスタム要求として扱う必要があります。
- Supplier クラス内の inventoryCheck() メソッドは、SupplyCheck と呼ばれるカスタム要求として扱う必要があります。

これらの要件を実現するようにカスタマイズされた custom_requests.xml ファイルの内容を以下に示します。

```
<customEdgeRequests>
      <edgeRequest>
              <requestName>CreditCheck</requestName>
              <Matches>com.mycompany.myapp.Customer</Matches>
              <type>application</type>
              <methodName>creditCheck</methodName>
      </edgeRequest>
      <edgeRequest>
              <requestName>SupplyCheck</requestName>
              <Matches>com.mycompany.myapp.Supplier</Matches>
              <type>application</type>
              <methodName>inventoryCheck</methodName>
      </edgeRequest>
</customEdgeRequests>
```

3. 以下のいずれかのステップを実行します。
   - ファイルを *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom ディレクトリーに保存します。次に、ツールキットのカスタム・プロパティー・ファイルで、プロパティー am.camtoolkit.gpe.customxml.custom を、ステップ 910 ページの『2』 で変更したファイルの名前 (パスなし) に設定します。
   - ご使用のコンピューター上の任意のディレクトリーにファイルを保存します。次に、ツールキットのカスタム・プロパティー・ファイルで、プロパティー am.camtoolkit.gpe.customxml.custom を、ステップ 910 ページの『2』 で変更したファイルのパスと名前に設定します。

   ツールキット・カスタム・プロパティー・ファイルに関して詳しくは、893 ページの『ツールキットのプロパティー・ファイル』を参照してください。

要求マッパーの使用可能化

ある要求タイプに対する要求マッパーを使用可能にするには、ツールキットのカスタム構成ファイルまたはツールキットのグローバル・カスタム構成ファイルを編集します。一般的な要求タイプの場合とカスタム要求の場合で手順が異なります。

始める前に

要求マッパー構成を XML ファイルに定義します。その後、要求マッパー構成を定義した XML ファイルを、ツールキット・プロパティー・ファイルと同じディレクトリー内に配置します。

- すべてのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、この XML ファイルを *dc_home*/runtime/custom ディレクトリー内に配置します。
- 1 つのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、この XML ファイルを *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/ ディレクトリー内に配置します。

XML ファイルの構文については、902 ページの『XML ファイルの構文』を参照してください。

このタスクについて

toolkit_custom.properties ファイルまたは toolkit_global_custom.properties ファイルを編集して、1 つのアプリケーション・サーバー・インスタンス、またはすべてのアプリケーション・サーバー・インスタンスに対して、要求マッパーを使用可能にします。

手順

- 一般的な要求に対して要求マッパーを使用可能にするには、以下の手順を実行します。
  a) テキスト・エディターで、次のいずれかのツールキット・カスタム構成ファイルを開きます。
     - すべてのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、*dc_home*/runtime/custom/toolkit_global_custom.properties ファイルを開きます。
     - 1 つのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties/ ファイルを開きます。
  b) ツールキット・プロパティー・ファイルを次のように編集します。
     - この要求タイプの使用可能化プロパティーを true に設定する行を追加します。詳しくは、914 ページの表 233 を参照してください。
     - am.camtoolkit.gpe.customxml.* プロパティーをマッパーの XML ファイルの名前に設定する行を追加します。* シンボルの代わりに任意の固有値を使用してください。詳しくは、902 ページの『XML ファイルの構文』を参照してください。
  c) プロパティー・ファイルを保存して閉じます。

  例:

  SQL 要求タイプに対して renameDataSource.xml に定義されている要求マッパーを使用可能にするには、ツールキット・カスタム構成ファイルまたはツールキット・グローバル・カスタム構成ファイルに、以下の行を追加します。

  ```
  com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
  am.camtoolkit.gpe.customxml.renameDataSource=renameDataSource.xml
  ```

- カスタム要求に対して要求マッパーを使用可能にするには、以下の手順を実行します。
  a) カスタム要求定義の XML ファイルで、<edgerequest> タグの下に <requestMapper> タグを作成します。このタグに、固有の要求マップ・タイプ名を入れます。カスタム要求の定義については、910 ページの『カスタム要求の定義』を参照してください。

b) 要求マッパーの XML ファイルで、<requestMapperDefinition> タグの type 属性に、この固有の要求マップ・タイプ名を使用します。

c) テキスト・エディターで、次のいずれかのツールキット・カスタム構成ファイルを開きます。

- すべてのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、*dc_home*/runtime/custom/toolkit_global_custom.properties ファイルを開きます。
- 1 つのアプリケーション・サーバー・インスタンスに対して要求マッパーを使用可能にするには、*dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties/ ファイルを開きます。

d) ツールキット・プロパティー・ファイルを編集して、am.camtoolkit.gpe.customxml.* プロパティーをマッパーの XML ファイルの名前に設定する行を追加します。* シンボルの代わりに任意の固有値を使用してください。詳しくは、902 ページの『XML ファイルの構文』を参照してください。

e) プロパティー・ファイルを保存して閉じます。

例:

custom_requests.xml ファイルで定義した SupplyCheck カスタム要求タイプに対して、customMapper.xml で定義した要求マッパーを使用可能にする場合は、以下の手順を実行します。

1. custom_requests.xml ファイルで、以下の定義を使用します。

```
<customEdgeRequests>
  <edgeRequest>
    <requestName>SupplyCheck</requestName>
    <Matches>com.mycompany.myapp.Supplier</Matches>
    <type>application</type
    <methodName>inventoryCheck</methodName>
    <requestMapper>customMapper</requestMapper>
  </edgeRequest>
</customEdgeRequests>
```

2. customMapper.xml ファイルで、タイプ名が次のように設定されていることを確認します。

```
<requestMapperDefinition type="customMapper">
```

3. ツールキット・カスタム構成ファイルまたはツールキット・グローバル・カスタム構成ファイルに、次の行を追加します。

```
am.camtoolkit.gpe.customxml.customMapper=customMapper.xml
```

要求マッパーのタイプ名、入力データ、および出力データ

以下の表に、各種の要求タイプの要求マッパーを構成して使用可能にするために必要な情報がリストされています。

各表の見出しの意味を以下に示します。

要求タイプ
要求タイプ。

使用可能化プロパティー
要求マッパーを使用可能にするには、toolkit_custom.properties ファイルまたは toolkit_global_custom.properties ファイルで、このプロパティーを true に設定します。

重要 : この値を表からコピーする場合は、スペースと改行を含めないようにしてください。

要求マッパー・タイプ名
要求マッパー定義 XML ファイルで、この値を <requestMapperDefinition> タグの type 属性に割り当てます。

入力データ・シンボル名
要求情報を表すシンボル。これらのシンボルを、要求マッパー定義内の式の中で使用できます。詳しくは、903 ページの『式の定義』を参照してください。

出力データ・コンテキスト・キー

モニター情報の変更内容を提供するには、要求マッパー定義内でこれらのキーに値を割り当てます。詳しくは、909 ページの『コンテキスト・キーへの値のマッピング』を参照してください。

表 *233.* 要求マッパーの使用可能化プロパティーおよびタイプ名

要求タイプ	**使用可能化プロパティー**	**要求マッパー・タイプ名**
サーブレット	com.ibm.tivoli.itcam.toolkit.ai.enable.servletrequestmapper	servlet
JNDI	com.ibm.tivoli.itcam.toolkit.ai.enable.jndirequestmapper	jndiLookup
カスタム要求		edgeRequest 定義内でユーザーが定義
EJB	com.ibm.tivoli.itcam.toolkit.ai.enable.ejbrequestmapper	ejb
JCA	com.ibm.tivoli.itcam.toolkit.ai.enable.jcarequestmapper	jca
JDBC データ・ソース	com.ibm.tivoli.itcam.toolkit.ai.enable.datasourcerequestmapper	dataSource
JDBC SQL ステートメント	com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper	sqlStatement
JMS	com.ibm.tivoli.itcam.toolkit.ai.enable.jmsrequestmapper	jms
JAX-RPC Web サービス	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
Axis Web サービス	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
MQI	com.ibm.tivoli.itcam.toolkit.ai.enable.mqrequestmapper	mqi
EJB	com.ibm.tivoli.itcam.toolkit.ai.enable.ejbrequestmapper	ejb
JDBC 接続ファクトリー	com.ibm.tivoli.itcam.toolkit.ai.enable.sqlconnectfactoryrequestmapper	connectionFactory
SCA	com.ibm.tivoli.itcam.toolkit.ai.enable.scarequestmapper	sca
JAX-WS Web サービス	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
WebSphere Portal Server Portal (extending the org.apache.jetspeed.portlet.Portlet class)	com.ibm.tivoli.itcam.toolkit.ai.enable.portalrequestmapper	portalPortal

表 233. 要求マッパーの使用可能化プロパティーおよびタイプ名 (続き)		
要求タイプ	**使用可能化プロパティー**	**要求マッパー・タイプ名**
WebSphere Portal Server バージョン 6.1、7、および 8 ポータル (`javax.portlet.Portlet` インターフェースを実装)	`com.ibm.tivoli.itcam.toolkit.ai.enable.portal6requestmapper`	`Portal6Portal`

重要 : 914 ページの表 233 に示されていない要求タイプに対してカスタム要求マッパーを構成するための有効な方法はありません。

表 234. 要求マッパーの入出力データ		
要求タイプ	**入力データ・シンボル名**	**出力データ・コンテキスト・キー**
サーブレット	詳しくは、919 ページの表 235 を参照してください。	**remappedURI** 名前変更された URI を定義します。 **remappedURL** 名前変更された URI を定義します。 **appName** 名前変更されたアプリケーション名を定義します。 **userid** 要求のユーザー ID を定義します。
JNDI	• **$jndiContext** Context オブジェクト • **$lookup** ルックアップ・ストリング • **$context** "JNDIlookup"	**renamedLookup** 名前変更されたルックアップ・ストリングを定義します。
カスタム要求	• **$this** カスタム要求メソッドの「this」オブジェクト • **$0** カスタム要求メソッドに渡される引数 (Object の配列として指定) • **$className** カスタム要求のクラス名 • **$methodName** カスタム要求のメソッド名 • **$context** edgeRequest 定義からの元の要求名	**customRequestName** 名前変更されたカスタム要求名を定義します。
EJB	• **$ejb** EJB 実装オブジェクト • **$appName** アプリケーションの名前 • **$ejbType** EJB のタイプ • **$className** EJB 実装オブジェクトのクラス名 • **$methodName** EJB ビジネス・メソッド名 • **$context** "EJBBusinessMethod"	**appName** 名前変更されたアプリケーション名を定義します。 **ejbType** 名前変更された EJB タイプを定義します。 **className** 名前変更されたクラス名を定義します。 **methodName** 名前変更されたメソッド名を定義します。

表 *234.* 要求マッパーの入出力データ *(続き)*

要求タイプ	入力データ・シンボル名	出力データ・コンテキスト・キー
JCA	• **$interaction** Interaction オブジェクト • **$interactionSpec** InteractionSpec オブジェクト • **$record** Record オブジェクト • **$context** "J2Cexecute"	**lookupName** 名前変更されたルックアップ名 **productName** 名前変更された製品名 **productVersion** 名前変更された製品のバージョン
JDBC データ・ソース	• **$this** データ・ソース・オブジェクトまたはドライバー・オブジェクト • **$dataSource** データ・ソースとしてキャストされた $this オブジェクト • **$driver** Driver としてキャストされた $this オブジェクト • **$dataSourceName** java.lang.String としてのデータ・ソース名 • **$context** 要求のタイプ (「JDBCgetConnection」 または 「JDBCgetConnection FromDriver」) を示します	**dataSourceName** 名前変更されたデータ・ソース名 ($this オブジェクトがデータ・ソースである場合) **url** 名前変更された Driver URL ($this オブジェクトが Driver の場合)
JDBC SQL ステートメント	• **$this** SQL ステートメントまたは SQL Connection • **$sqlText** java.lang.String としての SQL テキストを格納します ($this オブジェクトが SQL ステートメントの場合) • **$sqlStatement** SQL Statement としてキャストされた $this オブジェクト) • **$sqlConnection** SQL Connection としてキャストされた $this オブジェクト • **$dataSourceName** データ・ソース名 • **$context** 要求のタイプ (「JDBCexecute」、「JDBCexecuteQuery」、「JDBCexecuteUpdate」、「JDBCcreateStatement」、「JDBCprepareStatement」) を示します	**dataSourceName** 名前変更されたデータ・ソース名 **sqlText** 名前変更された SQL テキスト

表 234. 要求マッパーの入出力データ (続き)

要求タイプ	入力データ・シンボル名	出力データ・コンテキスト・キー
JMS	• **$this** 装備されたメソッドの'this' オブジェクト。QueueBrowser, MessageConsumer, MessageProducer, または MessageListener です。 • **$0** Queue オブジェクト (Send 要求の場合)、またはトピック・オブジェクト (Publish 要求の場合) • **$queueBrowser** QueueBrowser としてキャストされた $this オブジェクト • **$messageConsumer** MessageConsumer としてキャストされた $this オブジェクト • **$messageProducer** MessageProducer としてキャストされた $this オブジェクト • **$messageListener** MessageListener としてキャストされた $this オブジェクト • **$queue** キューとしてキャストされた $0 オブジェクト • **$topic** トピックとしてキャストされた $0 オブジェクト • **$context** 要求のタイプ (「JMSreceive」、「JMSsend」、「JMSbrowse」、「JMSpublish」、「JMSonmessage」) を示します	**queueName** 名前変更されたキュー名 **providerURL** 名前変更されたプロバイダー URL **topicName** 名前変更されたトピック名
JAX-RPC Web サービス	• **$messageContext** IMessageContextWrapper • **$appName** アプリケーション名 • **$requestName** デフォルトの要求名 • **$url** URL • **$context** 要求のタイプ (「WebServicesJaxRpc ProviderRequest」、「WebServicesJaxRpc ClientRequest」) を示します	**appName** 名前変更されたアプリケーション名 **requestName** 名前変更された要求名 **url** 名前変更された URL
Axis Web サービス	• **$messageContext** IMessageContextWrapper • **$appName** アプリケーション名 • **$requestName** デフォルトの要求名 • **$url** URL • **$context** 要求のタイプ (「WebServicesAxisClient Request」、「WebServicesAxis ProviderRequest」) を示します	**appName** 名前変更されたアプリケーション名 **requestName** 名前変更された要求名 **url** 名前変更された URL

表 234. 要求マッパーの入出力データ (続き)

要求タイプ	入力データ・シンボル名	出力データ・コンテキスト・キー
MQI	• **$queue** MQQueue オブジェクト (既知の場合) • **$qmgr** MQQueueManager オブジェクト (既知の場合) • **$message** MQMessage オブジェクトまたは MQMsg2 オブジェクト (既知の場合) • **$session** MQSESSION オブジェクト (既知の場合) • **$getMsgOptions** MQGetMessageOptions オブジェクト (既知の場合) • **$qmgrName** キュー・マネージャーの名前 • **$queueName** キューの名前 • **$context** MQ 要求のタイプ (「MQCONN」、「MQCONNX」、「MQDISC」、「MQBACK」、「MQBEGIN」、「MQCLOSE」、「MQCMIT」、「MQINQ」、「MQOPEN」、「MQSET」、「MQGET」、「MQPUT」、「MQPUT1」、「MQGETBROWSE」)	**qmgrName** 名前変更されたキュー・マネージャーの名前 **qname** 名前変更されたキュー名
EJB	• **$appName** アプリケーションの名前 • **$ejbType** EJB のタイプ • **$className** EJB 実装オブジェクトのクラス名 • **$methodName** EJB ビジネス・メソッド名 • **$context** "EJBBusinessMethod"	**appName** 名前変更されたアプリケーション名を定義します。 **ejbType** 名前変更された EJB タイプを定義します。 **className** 名前変更されたクラス名を定義します。 **methodName** 名前変更されたメソッド名を定義します。
JDBC 接続ファクトリー	• **$connectionFactory** ConnectionFactory • **$dataSourceName** データ・ソース名 • **$context** "JDBCgetConnection"	**dataSourceName** 名前変更されたデータ・ソース名
SCA	• **$uri** the URI • **$operationName** 操作名 • **$context** 要求のタイプ (「SCA_Generic」、「SCA_Ref」、「SCA_Target」) を示します	**uri** 名前変更された URI **operationName** 名前変更された操作名

表 *234.* 要求マッパーの入出力データ *(続き)*

要求タイプ	入力データ・シンボル名	出力データ・コンテキスト・キー
JAX-WS Web サービス	• **$messageContext** IMessageContextWrapper • **$appName** アプリケーション名 • **$requestName** デフォルトの要求名 • **$url** URL • **$context** 要求のタイプ (「WebServicesJAXWS ClientRequest」、「WebServicesJAXWS ProviderRequest」、「WebServicesJAXWS AsyncRequest」) を示します	**appName** 名前変更されたアプリケーション名 **requestName** 名前変更された要求名 **url** 名前変更された URL
WebSphere Portal Server Portal (extending the `org.apache.jetspeed.portlet.Portlet` class)	• **$portletAdapter** PortletAdapter • **$portletRequest** PortletRequest • **$portletResponse** PortletResponse • **$portletName** ポートレット名 • **$pageTitle** ページ・タイトル • **$url** 要求の URL • **$userid** 要求のユーザー ID • **$context** "Portal.Portlet"	**portletName** 名前変更されたポートレット名 **title** 名前変更されたページ・タイトル **url** 名前変更された URL **userid** 名前変更されたユーザー ID
WebSphere Portal Server バージョン 6.1、7、および 8 ポータル (`javax.portlet.Portlet` インターフェースを 実装)	• **$portlet** Portlet • **$renderRequest** RenderRequest • **$renderResponse** RenderResponse • **$portletName** ポートレット名 • **$pageTitle** ページ・タイトル • **$url** 要求の URL • **$userid** 要求のユーザー ID • **$context** "Portal.Portlet"	**portletName** 名前変更されたポートレット名 **title** 名前変更されたページ・タイトル **url** 名前変更された URL **userid** 名前変更されたユーザー ID

サーブレット要求の場合、より多数の入力データ・シンボルが提供されます。

表 *235.* サーブレット要求の入力データ・シンボル名

シンボル名	値のタイプ	シンボルの内容
$context	ストリング	"ServletMethod"
$servlet	javax.servlet.http.HttpServlet	サーブレット要求に関連する HttpServlet オブジェクト
$httpServletRequest	javax.servlet.http.HttpServletRequest	サーブレット要求に関連する HttpServletRequest オブジェクト
$httpServletResponse	javax.servlet.http.HttpServletResponse	サーブレット要求に関連する HttpServletResponse オブジェクト
$appName	java.lang.String	サーブレットに関連付けられたアプリケーション名

表 235. サーブレット要求の入力データ・シンボル名 (続き)

シンボル名	値のタイプ	シンボルの内容
$URL	java.lang.StringBuffer	クライアントが要求を行うために使用した URL
$RemoteUser	java.lang.String	この要求を行っているユーザーのログイン名 (認証された場合)
$URI	java.lang.String	要求 URL のプロトコル名から照会ストリングまでの部分
$ServletPath	java.lang.String	要求 URL のサーブレットを呼び出す部分。
$SessionID	javax.servlet.http.HttpSession	この要求に関連する現行セッション
$QueryString	java.lang.String	要求 URL 内のパスの後に含まれている照会ストリング。
$SessionAttribute	java.lang.String	このパラメーター化されたシンボルは、セッションの属性値を返します。これは 1 つのパラメーター (属性名) を持ちます。属性名はストリングでなければなりません。 例えば、`$SessionAttribute("attr1")` は、attr1 という名前の属性の値を返します。
$cookie	javax.servlet.http.Cookie	このパラメーター化されたシンボルは、名前付き Cookie を返します。これは 1 つのパラメーター (Cookie 名) を持ちます。Cookie 名はストリングでなければなりません。 例えば、`$cookie("cookie1")` は、cookie1 という名前の属性の値を返します。

要求マッパー定義の例
以下の例は、要求マッパー機能の使用法を示しています。

サーブレットのアプリケーション名の変更
この例では、サーブレット要求内のアプリケーション名が、URI と照会ストリングで置き換えられます。

dc_home/runtime/changeAppname.xml ファイルに、以下の要求マッパー定義が入っています。

```
<gpe>
   <runtimeConfiguration>
         <requestMapperDefinition type="servlet">
               <selection>
                     <matchCriteria>true</matchCriteria>
                     <mapTo>
                           <key>appName</key>
                           <value>$URI + “.” + $QueryString</value>
                     </mapTo>
               </selection>
         </requestMapperDefinition>
   </runtimeConfiguration>
</gpe>
```

データ・ソースの名前変更
この例では、SQL 要求内のデータ・ソース名が、ユーザーに分かりやすいバージョンに変更されます。

dc_home/runtime/renameDataSource.xml ファイルに、以下の要求マッパー定義が入っています。

```
<gpe>
 <runtimeConfiguration>
   <requestMapperDefinition type=”sqlStatement”>
      <selection>
         <matchCriteria>$dataSourceName != null</matchCriteria>
         <selection>
           <matchCriteria>$dataSourceName.equals("jdbc/TradeDataSource")
</matchCriteria>
           <mapTo>
             <key>dataSourceName</key>
             <value>"Daytrader Data Source"</value>
           </mapTo>
         </selection>
         <selection>
           <matchCriteria>$dataSourceName.equals("jdbc/LongDataSource")
</matchCriteria>
           <mapTo>
             <key>dataSourceName</key>
             <value>"Long term trader Data Source"</value>
           </mapTo>
         </selection>
      </selection>
   </requestMapperDefinition>
 </runtimeConfiguration>
<gpe>
```

最初の <selection> タグにより、$dataSourceName が NULL でないことが確実になります。次に、2 番目の <selection> タグで、$dataSourceName.equals() を安全に評価することができます。

最初の <selection> タグが存在せず、NULL の $dataSourceName が渡された場合、要求マッパーは例外を生成します。このような例外により、モニター情報が欠落することがあります。

この要求マッパーを使用可能にするには、ファイル *dc_home*/runtime/toolkit_global_custom.properties に、以下の行を含めます。

```
com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
am.camtoolkit.gpe.customxml.renameDataSource=renameDataSource.xml
```

SQL 要求からの機密情報の削除
この例では、アプリケーションの SQL 要求に社会保障番号が含まれています。要求マッパーは、ユーザーに見える可能性がある要求のバージョンから番号を削除します。

SQL 要求では、社会保障番号が SS 列名で SS = *number* としてリストされています。要求マッパーは、ストリング「SS =」を探し、その後にある 9 個のシンボルを削除します。

dc_home/runtime/removeSSN.xml ファイルに、以下の要求マッパー定義が入っています。

```
<gpe>
    <runtimeConfiguration>
         <requestMapperDefinition type="sqlStatement">
             <symbolDefinitions>
                 <symbol>
                     <name>$offsetOfSS</name>
                     <eval>$sqlText.indexOf("SS = ")</eval>
                 </symbol>
                 <symbol>
                     <name>$sqlTextContainsSS</name>
<eval>$sqlText != null AND $offsetOfSS > 0 AND $sqlText.length() GE
$offsetOfSS+16</eval>
                 </symbol>
                 <conditionalSymbol>
                     <name>$sqlTextPriorToSSKeyword</name>
                     <type>java.lang.String</type>
                     <defaultValue>""</defaultValue>
                     <if condition="$sqlTextContainsSS">
                       <return>$sqlText.substring(0, $offsetOfSS+5)</return>
                     </if>
                 </conditionalSymbol>
                 <conditionalSymbol>
```

```
                    <name>$sqlTextAfterSS</name>
                    <type>java.lang.String</type>
                    <defaultValue>""</defaultValue>
                    <if condition="$sqlTextContainsSS">
                      <return>$sqlText.substring($offsetOfSS+16)</return>
                    </if>
                  </conditionalSymbol>
              </symbolDefinitions>
              <selection>
                   <matchCriteria>$sqlText != null AND $sqlText.length() >
0</matchCriteria>
                   <selection>
                        <matchCriteria>$sqlTextContainsSS</matchCriteria>
                        <mapTo>
                             <key>sqlText</key>
                             <value>$sqlTextPriorToSSKeyword + "?" +
$sqlTextAfterSS</value>
                        </mapTo>
                   </selection>
              </selection>
         </requestMapperDefinition>
    </runtimeConfiguration>
</gpe>
```

この要求マッパーを使用可能にするには、ファイル *dc_home*/runtime/toolkit_global_custom.properties に、以下の行を含めます。

```
com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
am.camtoolkit.gpe.customxml.renameDataSource=removeSSN.xml
```

WebSphere Extreme Scale をモニターするように WebSphere Applications エージェントを構成

WebSphere Applications エージェントのインストール後、スタンドアロン環境または WebSphere Application Server 環境で WebSphere Extreme Scale (WXS) をモニターするように追加の構成を実行できます。

このタスクについて

構成手順は、WebSphere Extreme Scale のインストール・モードと、セキュリティーが有効かどうかに応じて異なります。構成プロセスの実行前に、次の手順を実行してください。

手順

1. WebSphere Extreme Scale のインストール・モードを確認します。

   スタンドアロン・モード
   WebSphere Extreme Scale は、WebSphere Application Server がない環境にインストールされています。

   組み込み WAS モード
   WebSphere Extreme Scale は、WebSphere Application Server 環境にインストールされています。

2. WebSphere Extreme Scale に対してセキュリティーが有効になっているかどうかを確認します。セキュア Java™ クライアントが組み込み WebSphere Application Server モードで使用されている場合、セキュリティー接続手順を実行する必要があります。
3. 手順については、次のリンクをクリックしてください。

   - スタンドアロン環境で WebSphere Extreme Scale を構成するには、923 ページの『スタンドアロン環境での WebSphere Extreme Scale モニターの構成』をクリックします。
   - セキュリティーのない組み込み環境で WebSphere Extreme Scale を構成するには、924 ページの『セキュリティーが有効になっていない WebSphere 環境での WebSphere Extreme Scale モニターの構成』をクリックします。
   - セキュリティーが有効な組み込み環境で WebSphere Extreme Scale を構成するには、925 ページの『セキュリティーが有効な WebSphere 環境での WebSphere Extreme Scale モニターの構成』をクリックします。

スタンドアロン環境での WebSphere Extreme Scale モニターの構成

WebSphere Application Server がない環境に WebSphere Extreme Scale がインストールされているときの WebSphere Applications エージェントの構成方法について説明します。

手順

1. WebSphere Applications エージェントを停止します。
   a) WebSphere Applications エージェントをインストールしたディレクトリー *install_dir* に移動します。
   b) コマンド `bin/was-agent.sh stop` を実行します。
2. 構成スクリプトを実行します。

   ```
   install_dir/platform_code/yn/bin/wxs-agent-config.sh config
   ```

   構文の内容は次のとおりです。

   - *install_dir* は WebSphere Applications エージェント のインストール・ディレクトリーです。
   - *platform_code* はエージェントをインストールしたプラットフォーム ・コードです。例えば、lx8266 は Linux x86_64 R2.6 (64 ビット) を、aix536 は AIX R5.3 (64 ビット) を表します。

   コマンド例:

   ```
   /opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh config
   ```

   ```
   /opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh config
   ```

3. エージェントのインストール・パスを求めるプロンプトが表示されたら、WebSphere Applications エージェントのホーム・ディレクトリーを指定します。

   注: スクリプトは、指定するインストール・パスに基づいて構成ファイル名を検索します。 デフォルトは *install_dir*`/config/${hostname}_yn.xml` です。ファイルが存在しないというプロンプトが表示された場合は、この構成を開始する前に WebSphere Applications エージェントを開始しなかったことが原因である可能性があります。WebSphere Applications エージェントを少なくとも 1 回開始してから、停止してください。
4. `WebSphere Extreme Scale Catalog Server connector type` というプロンプトが表示されたら、1 を入力して続行します。
5. `Input a node name to identify this agent node on UI` というプロンプトが表示されたら、ノード名を入力します。

   このノード名は、モニター対象の WebSphere Extreme Scale ゾーンを識別するために使用され、Application Performance ダッシュボード UI で確認できるインスタンス名に表示されます。
6. `WebSphere Extreme Scale Catalog Server security enabled?` というプロンプトが表示されたら、有効なセキュリティーがある場合は、1 を入力します。次に、ユーザー名とパスワードを入力します。有効なセキュリティーがない場合は、2 を入力します。
7. カタログ・サーバーのホスト名とポート番号を指定します。複数のカタログ・サーバーがある場合は、1 つずつ追加できます。複数のゾーンも、1 つずつ追加できます。

   - ホスト名は、カタログ・サーバーが配置されているシステムの名前です。ホスト名にアクセスできることを確認してください。アクセスできない場合は、ホスト名として IP アドレスを使用してください。
   - ポート番号は、WebSphere Extreme Scale カタログ・サーバーの **JMXServicePort** 番号です。デフォルト値は 1099 です。ポート番号について詳しくは、WebSphere Extreme Scale Knowledge Center を参照してください。
8. 以下のコマンドを入力して、エージェントを始動します。

   ```
   install_dir/bin/was-agent.sh start
   ```

   注:

- エージェント構成は、*install_dir*/config/${hostname}_yn.xml に格納されます。構成を変更する場合は、このスクリプトを再度実行するか、.xml ファイルを直接変更します。
- 前の構成は、*install_dir*/config/${hostname}_yn.xml.bak としてバックアップされます。必要に応じて、前の構成を復元できます。
- *install_dir*/*platform_code*/yn/bin/wxs-agent-config.sh config の実行時に Ctrl を押しながら C を押すと、スクリプトを終了できます。既存の構成は変更されません。

組み込み WebSphere 環境での WebSphere Extreme Scale モニターの構成
WebSphere Application Server 環境に WebSphere Extreme Scale がインストールされているときの WebSphere Applications エージェントの構成方法について説明します。

このタスクについて

WebSphere Extreme Scale サーバーに対してセキュリティーが有効になっていない場合は、構成プロセスを直接実行できます。それ以外の場合は、まず 925 ページの『セキュリティーが有効な WebSphere 環境での WebSphere Extreme Scale モニターの構成』を実行する必要があります。

セキュリティーが有効になっていない *WebSphere* 環境での *WebSphere Extreme Scale* モニターの構成
セキュリティーが有効になっていない WebSphere Application Server 環境に WebSphere Extreme Scale をインストールする場合、WebSphere Applications エージェントを直接構成できます。

手順

1. WebSphere Applications エージェントを停止します。
   a) WebSphere Applications エージェントをインストールしたディレクトリー *install_dir* に移動します。
   b) コマンド bin/was-agent.sh stop を実行します。
2. 構成スクリプトを実行します。

   ```
   install_dir/platform_code/yn/bin/wxs-agent-config.sh config
   ```

   構文の内容は次のとおりです。
   - *install_dir* は WebSphere Applications エージェント のインストール・ディレクトリーです。
   - *platform_code* はエージェントをインストールしたプラットフォーム ・コードです。例えば、lx8266 は Linux x86_64 R2.6 (64 ビット) を、aix536 は AIX R5.3 (64 ビット) を表します。

   コマンド例:

   ```
   /opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh
   ```

   ```
   /opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh
   ```

3. エージェントのインストール・パスを求めるプロンプトが表示されたら、WebSphere Applications エージェントのホーム・ディレクトリーを指定します。

   注: スクリプトは、指定するインストール・パスに基づいて構成ファイル名を検索します。 デフォルトは *install_dir*/config/${hostname}_yn.xml です。ファイルが存在しないというプロンプトが表示された場合は、この構成を開始する前に WebSphere Applications エージェントを開始しなかったことが原因である可能性があります。WebSphere Applications エージェントを少なくとも 1 回開始してから、停止してください。
4. WebSphere Extreme Scale Catalog Server connector type というプロンプトが表示されたら、2 を入力して続行します。
5. Input a node name to identify this agent node on UI というプロンプトが表示されたら、ノード名を入力します。

   このノード名は、モニター対象の WebSphere Extreme Scale ゾーンを識別するために使用され、Application Performance ダッシュボード UI で確認できるインスタンス名に表示されます。

6. WebSphere Extreme Scale Catalog Server security enabled? というプロンプトが表示されたら、2 を入力して続行します。
7. カタログ・サーバーのホスト名とポート番号を指定します。複数のカタログ・サーバーがある場合は、1 つずつ追加できます。複数のゾーンも、1 つずつ追加できます。
   - ホスト名は、カタログ・サーバーが配置されているシステムの名前です。ホスト名にアクセスできることを確認してください。アクセスできない場合は、ホスト名として IP アドレスを使用してください。
   - ポート番号は、WebSphere Extreme Scale カタログ・サーバーの **JMXServicePort** 番号を示します。これは、各 WebSphere Application Server の **BOOTSTRAP_ADDRESS** 値から継承されます。ポート番号について詳しくは、WebSphere Extreme Scale Knowledge Center を参照してください。
8. 以下のコマンドを入力して、エージェントを始動します。

   ```
   install_dir/bin/was-agent.sh start
   ```

   注:
   - エージェント構成は、*install_dir*/config/${hostname}_yn.xml に格納されます。構成を変更する場合は、このスクリプトを再度実行するか、.xml ファイルを直接変更します。
   - 前の構成は、*install_dir*/config/${hostname}_yn.xml.bak としてバックアップされます。必要に応じて、前の構成を復元できます。
   - *install_dir*/*platform_code*/yn/bin/wxs-agent-config.sh config の実行時に Ctrl を押しながら C を押すと、スクリプトを終了できます。既存の構成は変更されません。

セキュリティーが有効な *WebSphere* 環境での *WebSphere Extreme Scale* モニターの構成

セキュリティーが有効になっている WebSphere Application Server 環境に WebSphere Extreme Scale をインストールする場合、WebSphere Applications エージェントを構成する前に、初期セットアップ手順を実行する必要があります。

このタスクについて

WebSphere Application Server セキュリティー有効環境で WebSphere Extreme Scale サーバーをモニターする場合、セキュリティー設定を手動で構成することが必要です。

この手順は、次の場合にあてはまります。

- WebSphere Application Server アプリケーション・サーバー (またはノード・エージェントあるいは DMGR プロセス) 内に WebSphere Extreme Scale サーバーをデプロイする必要がある場合。
- WebSphere Extreme Scale ゾーンのカタログ・サービスが実行されているノードに WebSphere Applications エージェントをデプロイする必要がある場合。このノードに WebSphere Extreme Scale モニターのエージェントを構成し、このカタログ・サービス・インスタンスに接続するように設定します。
- 1 つのみの WebSphere Extreme Scale ゾーンをモニターするために 1 つのエージェント・インスタンスを使用する必要がある場合。

手順

1. WebSphere Application Server の JDK バージョンが 1.7 より前の場合、WebSphere Application Server と同じ JRE を使用するように WebSphere Applications エージェントを再構成する必要があります。

   a) *install_dir*/config/.yn.environment ファイルを開きます。

   b) 最初の行に次の値を追加します。

   ```
   #JAVAHOME=/opt/IBM/WebSphere/AppServer/java/8.0/jre
   ```

2. WebSphere Applications エージェントのセキュリティー・プロパティー・ファイルを構成します。

   説明については、926 ページの『WebSphere Application Server の JAR ファイルおよびセキュリティー・プロパティーと連携して動作するようにエージェントを構成』を参照してください。

3. オプション: セキュア Java™ クライアントが使用されている場合、認証が適切に構成されていることを確認する必要があります。クライアント・プロパティー・ファイルと SSL プロパティー・ファイルを編集する必要があります。説明については、927 ページの『接続資格情報のセットアップ』を参照してください。

   注 : 鍵が SSL 設定で保護されていない場合、入力する必要があるのはパスワードとユーザー名のみで、その後、このステップをスキップできます。
4. 構成スクリプトを実行して、構成コンソールを起動します。929 ページの『構成の実行』を参照してください。

WebSphere Application Server の *JAR* ファイルおよびセキュリティー・プロパティーと連携して動作するようにエージェントを構成

WebSphere Application Server の JAR ファイルおよびセキュリティー・プロパティーと連携して動作するように、WebSphere Applications エージェントを構成します。

このタスクについて

この構成を実行するには、kynwb.properties ファイルを編集します。

手順

1. ファイル *install_dir*/*platform_code*/yn/config/kynwb.properties を開きます。このファイルが存在しない場合は、作成します。
   - *install_dir* は WebSphere Applications エージェント のインストール・ディレクトリーです。
   - *platform_code* はエージェントをインストールしたプラットフォーム ・コードです。例えば、lx8266 は Linux x86_64 R2.6 (64 ビット) を、aix536 は AIX R5.3 (64 ビット) を表します。
2. ファイルの先頭に、クラスパスがリストされています。 既存の行の前に、次の行を追加します。
   - WebSphere Application Server 9.0 の場合:

     ```
     appserver_home/plugins/com.ibm.ws.runtime.jar:¥
     appserver_home/lib/bootstrap.jar:¥
     appserver_home/runtimes/com.ibm.ws.admin.client_9.0.jar:¥
     appserver_home/lib/wsogclient.jar:¥
     ```

   - WebSphere Application Server 8.5 の場合:

     ```
     appserver_home/plugins/com.ibm.ws.runtime.jar:¥
     appserver_home/lib/bootstrap.jar:¥
     appserver_home/runtimes/com.ibm.ws.admin.client_8.5.0.jar:¥
     appserver_home/lib/wsogclient.jar:¥
     ```

   変更したクラスパスの例:

   ```
   /opt/IBM/WebSphere/plugins/com.ibm.ws.runtime.jar:¥
   /opt/IBM/WebSphere/lib/bootstrap.jar:¥
   /opt/IBM/WebSphere/runtimes/com.ibm.ws.admin.client_8.5.0.jar:¥
   /opt/IBM/WebSphere/lib/wsogclient.jar:¥
   lib/kynwb.jar:¥
   lib/kynwxssec_api.jar:¥
   lib/itcam.cg.mbean.jar:¥
   wasdc/7.3/installer/lib/itcamfwas.jar:¥
   ```

3. *install_dir*/*platform_code*/yn/config/kynwb.properties ファイルの末尾に、エージェントが使用するセキュリティー・プロパティー・ファイルを示す行を追加します。通常、これらのファイルは、wsadmin ユーティリティーが 使用するファイルです。

   ```
   -Dcom.ibm.CORBA.ConfigURL=file:/appserver_profile/properties/sas.client.props
   -Dcom.ibm.SSL.ConfigURL=file:/appserver_profile/properties/ssl.client.props
   ```

   エージェントに必要なセキュリティー設定が wsadmin ユーティリティーが 使用する設定とは異なる場合、ファイルの別個のコピーを作成し、それらのファイルへのパスを指定します。例:

```
-Dcom.ibm.CORBA.ConfigURL=file:/opt/IBM/ITM/config/sas.client.props
-Dcom.ibm.SSL.ConfigURL=file:/opt/IBM/ITM/config/ssl.client.props
```

注 : WebSphere Applications エージェントのフィックスパックまたは暫定修正をインストールすると、yn.ini ファイルと kynwb.properties ファイルに加えられた変更が上書きされます。 このため、フィックスパックまたは暫定修正のインストール後、これら 2 つのファイルに再度変更を加える必要があります。

接続資格情報のセットアップ
セキュア Java クライアントを使用する場合、CSIv2 設定のリストを含むプロパティー・ファイルを読み取る必要があります。これらの設定は、クライアントがサーバーでの 認証を行う方法を決定します。認証が適切に構成されていることを確認する必要があります。

このタスクについて

通常、これらの設定があるファイルは、com.ibm.CORBA.ConfigURL JVM プロパティーで指定されます。追加の SSL 設定は、com.ibm.SSL.ConfigURL JVM プロパティーで指定されたファイルにあります。

WebSphere Application Server に組み込まれている eXtreme Scale サーバーをモニターするように WebSphere Applications エージェントが構成されているとき、これはセキュア Java クライアントとして動作します。そのため、-Dcom.ibm.CORBA.ConfigURL と -Dcom.ibm.SSL.ConfigURL を、kynwb.properties ファイルで指定する必要があります。

多くの場合、これらのプロパティーは、*appserver_profile*/properties ディレクトリーの sas.client.props および ssl.client.props ファイルを指しています。 これらのファイルは、wsadmin、xscmd などのツールによって使用されます。 このため、これらのいずれかのツールを使用して、資格情報を入力することなく eXtreme Scale カタログ・サーバーに接続できる場合、設定をカスタマイズする必要はありません。

接続が失敗するか、ユーザー名とパスワードの入力が必要な場合は、追加の構成を行う必要があります。

クライアント・プロパティー・ファイルの変更
WebSphere Applications エージェントが使用する sas.client.props ファイルを編集します。

このタスクについて
sas.client.props ファイルへの絶対パスは、-Dcom.ibm.CORBA.ConfigURL プロパティーの kynwb.properties で指定されています。 エージェントを構成する、カタログ・サービス・インスタンスを実行している WebSphere Application Server インスタンスに対する接続情報およびセキュリティー情報を指定します。

手順

1. *appserver_profile*/properties/sas.client.props ファイルを開きます。
2. com.ibm.CORBA.loginSource プロパティーの値を properties に変更します。

```
com.ibm.CORBA.loginSource=properties
```

3. プロパティー com.ibm.CORBA.securityServerHost を、WebSphere Extreme Scale ゾーン内のアプリケーション・サーバーのホスト名に設定します。このサーバーは、ローカル・サーバーでも、それ以外のサーバーでも問題ありません。 このサーバーは、エージェントの始動時に常に使用可能である必要があります。 例:

```
com.ibm.CORBA.securityServerHost=server.company.com
```

4. com.ibm.CORBA.securityServerPort プロパティーを、アプリケーション・サーバー・プロファイルの RMI ポートに設定します。例:

```
com.ibm.CORBA.securityServerPort=2819
```

5. プロパティー com.ibm.CORBA.loginUserid をアプリケーション・サーバーと通信するためのログイン名に、プロパティー com.ibm.CORBA.loginPassword をそのパスワードに設定します。例:

```
com.ibm.CORBA.loginUserid=admin
com.ibm.CORBA.loginPassword=password
```

6. WebSphere 管理コンソールの **CSIv2 inbound communications** 設定に応じて、次のプロパティーを true または false に設定します。

```
com.ibm.CSI.performTLClientAuthenticationRequired
com.ibm.CSI.performTLClientAuthenticationSupported
com.ibm.CSI.performTransportAssocSSLTLSRequired
com.ibm.CSI.performTransportAssocSSLTLSSupported
```

   com.ibm.CSI.performTLClientAuthentication* の各プロパティーは、**Client certificate authentication** 認定に関連しています。com.ibm.CSI.performTransportAssocSSLTLS* は、**Transport** 設定に関連しています。

7. オプション: デフォルトの SSL 別名 (DefaultSSLSettings) を使用しない場合、com.ibm.ssl.alias プロパティーで SSL 構成別名を設定します。
8. ファイルを保存し、sas.client.props ファイル内のパスワードを暗号化します。 パスワードを暗号化するには、次のコマンドを実行します。
   - Linux および UNIX システムの場合、*appserver_profile*/bin/PropFilePasswordEncoder.sh sas.client.props com.ibm.CORBA.loginPassword を実行します

   重要 : クライアント証明書認証が必要な場合で、基本認証が有効な場合、com.ibm.CORBA.validateBasicAuth=false プロパティーの設定も必要になる場合があります。

クライアント *SSL* プロパティー・ファイルの変更
サーバー証明書にアクセスするために WebSphere Applications エージェントが使用する SSL プロパティー・ファイルを変更します。

このタスクについて

エージェントが使用する ssl.client.props ファイルを編集します。このファイルへの絶対パスは、kynwb.properties ファイルの -Dcom.ibm.SSL.ConfigURL プロパティーで指定されています。エージェントが構成されているカタログ・サービス・インスタンスを実行している WebSphere Application Server インスタンスの SSL トラストストアと鍵ストアの情報を指定します。

WebSphere 管理コンソール (「**セキュリティー**」 > 「**SSL 証明書および鍵管理**」 > 「**鍵ストアと証明書**」) を使用するか、iKeyman ツールを使用して、証明書を作成および管理できます。

手順

1. *appserver_profile*/properties/ssl.client.props ファイルを開きます。
2. com.ibm.ssl.alias プロパティーの値を、sas.client.props ファイルの同じプロパティーの値と一致するように変更します。

   ヒント : ssl.client.props ファイルには、複数の SSL 構成が含まれている場合があります。 それぞれの構成は、com.ibm.ssl.alias プロパティーで始まります。

3. com.ibm.ssl.enableSignerExchangePrompt プロパティーを false に設定します。
4. クライアント・アプリケーションが暗号鍵にアクセスできるように、次の鍵ストア・プロパティーを設定します。

   com.ibm.ssl.keyStoreName
   この鍵ストアを識別する名前

   com.ibm.ssl.keyStore
   鍵ストア・ファイルの絶対パスと名前

   com.ibm.ssl.keyStorePassword
   鍵ストアのパスワード

com.ibm.ssl.keyStoreType
鍵ストア・タイプ。他のアプリケーションとのインターオペラビリティーという 観点から、デフォルトの PKCS12 タイプを使用してください。

重要 : クライアント 証明書認証が不要な場合、鍵ストアには任意の自己署名鍵を含めることができます。それ以外の場合、鍵ストアにはサーバー・トラストストアにある証明書で署名された鍵を含める必要があります。

5. クライアント・アプリケーションが署名者証明書にアクセスできるように、次のトラストストア・プロパティーを設定します。

com.ibm.ssl.trustStoreName
このトラストストアを識別する名前

com.ibm.ssl.trustStore
トラストストア・ファイルの絶対パスと名前

com.ibm.ssl.trustStorePassword
トラストストアのパスワード

com.ibm.ssl.trustStoreType
トラストストア・タイプ。他のアプリケーションとのインターオペラビリティーという 観点から、デフォルトの PKCS12 タイプを使用してください。

重要 : クライアントが SSL 接続を使用する場合、サーバーの署名者証明書は、そのトラストストア内に存在している必要があります。

構成の実行
環境とセキュリティーを確認後、構成プロセスを実行できます。

手順

1. WebSphere Applications エージェントを停止します。
   a) WebSphere Applications エージェントをインストールしたディレクトリー *install_dir* に移動します。
   b) コマンド `bin/was-agent.sh stop` を実行します。
2. 構成スクリプトを実行します。

```
install_dir/platform_code/yn/bin/wxs-agent-config.sh config
```

   構文の内容は次のとおりです。

   - *install_dir* は WebSphere Applications エージェント のインストール・ディレクトリーです。
   - *platform_code* はエージェントをインストールしたプラットフォーム ・コードです。例えば、lx8266 は Linux x86_64 R2.6 (64 ビット) を、aix536 は AIX R5.3 (64 ビット) を表します。

   コマンド例:

```
/opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh config
```

```
/opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh config
```

3. エージェントのインストール・パスを求めるプロンプトが表示されたら、WebSphere Applications エージェントのホーム・ディレクトリーを指定します。

   注 : スクリプトは、指定するインストール・パスに基づいて構成ファイル名を検索します。 デフォルトは *install_dir*`/config/${hostname}_yn.xml` です。ファイルが存在しないというプロンプトが表示された場合は、この構成を開始する前に WebSphere Applications エージェントを開始しなかったことが原因である可能性があります。WebSphere Applications エージェントを少なくとも 1 回開始してから、停止してください。
4. `WebSphere Extreme Scale Catalog Server connector type` というプロンプトが表示されたら、2 を入力して続行します。

5. `Input a node name to identify this agent node on UI` というプロンプトが表示されたら、ノード名を入力します。

   このノード名は、モニター対象の WebSphere Extreme Scale ゾーンを識別するために使用され、Application Performance ダッシュボード UI で確認できるインスタンス名に表示されます。
6. `WebSphere Extreme Scale Catalog Server security enabled?` というプロンプトが表示されたら、1 を入力して続行します。次に、ユーザー名とパスワードを入力します。
7. カタログ・サーバーのホスト名とポート番号を指定します。複数のカタログ・サーバーがある場合は、1 つずつ追加できます。複数のゾーンも、1 つずつ追加できます。
   - ホスト名は、カタログ・サーバーが配置されているシステムの名前です。ホスト名にアクセスできることを確認してください。アクセスできない場合は、ホスト名として IP アドレスを使用してください。
   - ポート番号は、WebSphere Extreme Scale カタログ・サーバーの **JMXServicePort** 番号を示します。これは、各 WebSphere Application Server の **BOOTSTRAP_ADDRESS** 値から継承されます。ポート番号について詳しくは、WebSphere Extreme Scale Knowledge Center を参照してください。
8. 以下のコマンドを入力して、エージェントを始動します。

   ```
   install_dir/bin/was-agent.sh start
   ```

   注:
   - エージェント構成は、*install_dir*`/config/${hostname}_yn.xml` に格納されます。構成を変更する場合は、このスクリプトを再度実行するか、`.xml` ファイルを直接変更します。
   - 前の構成は、*install_dir*`/config/${hostname}_yn.xml.bak` としてバックアップされます。必要に応じて、前の構成を復元できます。
   - *install_dir*`/bin/wxs-agent-config.sh config` の実行時に Ctrl を押しながら C を押すと、スクリプトを終了できます。既存の構成は変更されません。

WebSphere Extreme Scale モニターの構成解除

WebSphere Extreme Scale をモニターしないときは、WebSphere Applications エージェントの構成を解除できます。

手順

1. WebSphere Applications エージェントを停止します。
   a) WebSphere Applications エージェントをインストールしたディレクトリー *install_dir* に移動します。
   b) コマンド `bin/was-agent.sh stop` を実行します。
2. 構成解除スクリプトを実行します。

   ```
   install_dir/{pc}/yn/bin/wxs-agent-config.sh unconfig
   ```

   構文の内容は次のとおりです。
   - *install_dir* は WebSphere Applications エージェント のインストール・ディレクトリーです。
   - *platform_code* はエージェントをインストールしたプラットフォーム ・コードです。例えば、lx8266 は Linux x86_64 R2.6 (64 ビット) を、aix536 は AIX R5.3 (64 ビット) を表します。

   コマンド例:

   ```
   /opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh unconfig
   ```

   ```
   /opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh unconfig
   ```

WebSphere Infrastructure Manager モニターの構成

WebSphere デプロイメント・マネージャーおよびノード・エージェントのパフォーマンスをモニターするように WebSphere Infrastructure Manager エージェントを構成します。

このタスクについて

WebSphere Infrastructure Manager エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

手順

1. エージェントを構成するには、以下のコマンドを実行します。

   ```
   install_dir/bin/wim-agent.sh config instance_name
   ```

   ここで、*instance_name* はインスタンスに指定する名前であり、*install_dir* は WebSphere Infrastructure Manager エージェントのインストール・ディレクトリーです。 デフォルトのインストール・ディレクトリーは、/opt/ibm/apm/agent です。
2. 「Monitoring Agent for WebSphere Infrastructure Manager」の設定を編集する（Edit 'Monitoring Agent for WebSphere Infrastructure Manager' settings）というプロンプトが出されたら、1 を入力して続行します。
3. Java ホームの入力を求めるプロンプトが出されたら、Java のインストール先ディレクトリーを指定します。

   デフォルト値は、/opt/ibm/apm/agent/JRE/lx8266/jre です。
4. DMGR プロファイル・ホームの入力を求めるプロンプトが出されたら、 デプロイメント・マネージャー・プロファイルのホーム・ディレクトリーを指定します。

   デフォルトのディレクトリーは、/opt/IBM/WebSphere/AppServer/profiles/Dmgr01 です。
5. JMX ユーザー ID の入力を求めるプロンプトが出されたら、MBean サーバーへの接続に使用するユーザー ID を指定します。
6. 「JMX パスワードの入力（Enter JMX password)」というプロンプトが出されたら、 そのユーザーのパスワードを指定します。
7. 「JMX パスワードの再入力（Re-type JMX password)」というプロンプトが出されたら、 再度パスワードを入力します。
8. 以下のコマンドを入力して、エージェントを始動します。

   ```
   install_dir/bin/wim-agent.sh start instance_name
   ```

タスクの結果

WebSphere Infrastructure Manager エージェント インスタンスを作成し、 モニター・エージェントを開始して、リソース・モニター用のデータ・サンプルの収集を開始しました。

WebSphere MQ のモニターの構成

エージェントを開始するには、その前にインスタンス名をエージェントに割り当て、ユーザー ID および管理対象システム名のためのいくつかの構成タスクを実行する必要があります。オプションとして、エージェントのトランザクション・トラッキングを有効にすることもできます。

始める前に

- 以下の手順は、このエージェントの最新リリース用です。ご使用の環境内のエージェントのバージョンを確認する方法については、『エージェント・バージョン・コマンド』を参照してください。エージェントのバージョン・リストと各バージョンの新機能について詳しくは、52 ページの『変更履歴』を参照してください。

- ご使用の環境で、WebSphere MQ エージェント のシステム要件が満たされているか確認します。システム要件に関する最新情報については、Detailed System Requirements Report for the WebSphere MQ エージェント を参照してください。

このタスクについて

ここに示す手順は、明記されている場合を除き、最新リリースのエージェント用です。

WebSphere MQ エージェント用に環境をセットアップするには、まず、エージェント・ユーザー ID で IBM MQ (WebSphere MQ) オブジェクトにアクセスできることを確認し、データ使用可能化のために IBM MQ (WebSphere MQ) を構成してから、WebSphere MQ エージェントを構成する必要があります。

以下の手順は、WebSphere MQ エージェントを構成するためのロードマップであり、必須のステップとオプションのステップの両方が含まれています。ニーズに従って、必要なステップを実行します。

手順

1. IBM MQ (WebSphere MQ) オブジェクトにアクセスするためにエージェントの 構成、開始、および停止に使用するユーザー ID に権限を付与します。932 ページの『エージェントを実行するためのユーザー ID の許可』を参照してください。
2. IBM MQ(WebSphere MQ) を構成して、モニターするデータを使用可能にします。934 ページの『データ使用可能化のための IBM MQ (WebSphere MQ) の構成』を参照してください。
3. エージェント・インスタンス名、キュー・マネージャー名、およびオプションでエージェント 名を指定して、エージェントを構成します。936 ページの『WebSphere MQ エージェントの構成』を参照してください。
4. オプション: モニター要件によっては、複数のモニター・エージェントを区別するために固有の管理対象システム 名が必要になることがあります。`mq-agent.sh config` コマンドの「`Agent Name`」オプションを使用して、管理対象システム 名の中間修飾子を指定します。939 ページの『複数のキュー・マネージャーに対する固有の管理対象システム 名の指定』を参照してください。
5. オプション: モニター対象キュー・マネージャーのトランザクション・トラッキング・データを収集するようにエージェントを構成するには、「**エージェント構成**」ページを参照してください。説明については、941 ページの『WebSphere MQ エージェントのトランザクション・トラッキングの構成』を参照してください。
6. オプション: エージェントによるキューおよびチャネルの長期間ヒストリー・データの収集を有効にします。説明については、941 ページの『キューおよびチャネルの長期間ヒストリーのためのデータ収集の有効化』を参照してください。
7. オプション: MQ アプライアンス 上のキュー・マネージャーをリモートでモニターするには、エージェントと IBM MQ (WebSphere MQ) の両方の追加構成が必要になります。説明については、943 ページの『MQ アプライアンス 上のキュー・マネージャーのリモートでのモニター』または 944 ページの『MQ アプライアンス 上の HA キュー・マネージャーのリモートでのモニター』を参照してください。

エージェントを実行するためのユーザー ID の許可

ユーザー ID が WebSphere MQ エージェントを構成、開始、および停止するには、そのユーザー ID は、**mqm** グループに属している必要があります。このグループは、IBM MQ (WebSphere MQ) における完全な管理特権を備えています。また、非 root ユーザーまたは非管理者ユーザーについては、IBM MQ (WebSphere MQ) 制御コマンドを使用して、IBM MQ (WebSphere MQ) オブジェクトに対するアクセス権限をユーザーに付与する必要があります。

このタスクについて

AIX システムおよび Linux システムでは、ユーザー ID を **mqm** グループに追加した後、**setmqaut** コマンドを使用して IBM MQ (WebSphere MQ) オブジェクトに対する適切なアクセス権限をそのユーザー ID に付与する必要があります。

Windows システムでは、ユーザー ID を **mqm** グループに追加する必要があります。ユーザー ID が管理者ユーザー・グループに属していない場合は、レジストリー・エディターを使用して、エージェントの開始または停止を行う許可をそのユーザー ID に付与する必要もあります。

手順

- Linux AIX

  AIX システムまたは Linux システムでは、以下の手順を完了します。

  a) AIX または Linux システムに、ID として root を使用してログオンします。

  b) エージェントの実行に使用するユーザー ID を **mqm** グループに追加します。

  c) (WebSphere MQ V7.5 以降): ユーザー ID が、AIX システムまたは Linux システム上の非 root ユーザーである場合は、以下のコマンドを実行して、IBM MQ (WebSphere MQ) オブジェクトにアクセスするための適切なレベルの権限をそのユーザー ID に設定します。

  ```
  setmqaut -m queue_manager -t qmgr -p user_ID +inq +connect +dsp +setid
  ```

  ここで、*queue_manager* は WebSphere MQ V7.5 以降のキュー・マネージャーの名前であり、*user_ID* は、エージェントを実行する非 root ユーザーまたは非管理者ユーザーの ID です。

- Windows

  Windows システムでは、以下の手順を完了します。

  a) システム管理者として Windows システムにログオンします。

  b) エージェントの実行に使用するユーザー ID を **mqm** グループに追加します。

  c) エージェントを開始、実行、および停止するために使用するユーザー ID が管理者グループのメンバーでない場合は、レジストリー・エディターを使用して、エージェントを正常に開始および停止できる許可をユーザー ID に設定します。

  a.「**スタート**」>「**ファイル名を指定して実行**」をクリックし、`regedit.exe` と入力して、「レジストリ エディタ」を開きます。

  b.「レジストリ エディタ」で、キー `HKEY_LOCAL_MACHINE¥SOFTWARE¥Candle` を見つけます。

  c. キーを右クリックし、「**アクセス許可**」をクリックします。

  d. WebSphere MQ エージェント用のユーザー ID が「グループ名またはユーザー名」リストにない場合は、「**追加**」をクリックしてそのユーザー ID をリストに追加します。

  e. リストにあるユーザー ID をクリックします。

  f.「*user-ID* のアクセス許可」リスト (ここで、*user-ID* は WebSphere MQ エージェントのユーザー ID) で、「許可」列の「**フル コントロール**」を選択し、「**OK**」をクリックします。

  g.「レジストリ エディタ」で、キー `HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows NT¥CurrentVersion¥Perflib` を見つけます。

  h. キーを右クリックし、「**アクセス許可**」をクリックします。

  i. WebSphere MQ エージェント用のユーザー ID が「グループ名またはユーザー名」リストにない場合は、「**追加**」をクリックしてそのユーザー ID をリストに追加します。

  j.「グループ名またはユーザー名」リストにあるユーザー ID をクリックします。

  k.「*user-ID* のアクセス許可」リスト (ここで、*user-ID* は WebSphere MQ エージェントのユーザー ID) で、「許可」列の「**読み取り**」を選択し、「**OK**」をクリックします。

  l.「レジストリ エディタ」を閉じます。

  m. *install_dir* ディレクトリーを見つけます。ここで、*install_dir* はエージェントのインストール・ディレクトリーです。

  n. ディレクトリーを右クリックし、「**プロパティ**」をクリックします。

  o.「セキュリティー」タブで、WebSphere MQ エージェント用のユーザー ID が「グループ名またはユーザー名」リストにない場合は、「**編集**」と「**追加**」を順にクリックし、そのユーザー ID をリストに追加します。

  p.「グループ名またはユーザー名」リストにあるユーザー ID をクリックします。

  q.「*user-ID* のアクセス許可」リスト (ここで、**user-ID** は WebSphere MQ エージェントのユーザー ID) で、「許可」列の「**フル コントロール**」を選択します。

  r.「**OK**」をクリックします。

次のタスク

次のステップでは、データの使用可能化のために IBM MQ (WebSphere MQ) を構成します。934 ページの『データ使用可能化のための IBM MQ (WebSphere MQ) の構成』を参照してください。

データ使用可能化のための IBM MQ (WebSphere MQ) の構成

WebSphere MQ エージェントを構成する前に、まず、IBM MQ (WebSphere MQ) を構成して、モニターするデータを使用可能にすることをお勧めします。

このタスクについて

WebSphere MQ エージェントでモニターするデータのタイプを決定します。データがデフォルトでキュー・マネージャーによって生成されない場合は、MQSC コマンドを使用して、キュー・マネージャーでデータを使用可能にします。

要確認 : MQSC コマンドを発行する前に、ターゲット・キュー・マネージャーに対して MQSC を開始する必要があります。キュー・マネージャーのリストを取得するには、IBM MQ (WebSphere MQ) インストール・ディレクトリー内の bin ディレクトリーから **dspmq** コマンドを発行します。キュー・マネージャーに対して MQSC を開始するには、bin ディレクトリーから以下のコマンドを発行します (ここで、*<qmgr_name>* は、構成するキュー・マネージャーの名前です)。

```
runmqsc <qmgr_name>
```

手順

- キューに存在する最も古いメッセージの経過時間を調べるには、934 ページの『キューのリアルタイム・モニターの有効化』に記載されている手順を実行します。
- キュー・マネージャーによってデフォルトで生成されない特定のキュー・マネージャー・イベントをモニターするには、935 ページの『キュー・マネージャーのイベント・モニターの有効化』に記載されているステップを実行します。
- トランザクション・トラッキング・データを取得するには、935 ページの『MQI アプリケーション・アクティビティー・トレースの有効化』に記載されているステップを実行します。
- リモート・キュー・マネージャーをモニターするには、WebSphere MQ エージェントがリモート・システムのチャネルを通じてモニター・データを収集できる必要があります。詳しくは、935 ページの『リモート・モニターのセキュリティー設定』を参照してください。

キューのリアルタイム・モニターの有効化

このタスクについて

キューに存在する最も古いメッセージの経過時間 (秒数) を調べるには、キューのリアルタイム・モニターを有効にする必要があります。

手順

ご使用の環境でキューのリアルタイム・モニターを有効にするには、以下のコマンドを使用します。

- MONQ 属性が QMGR に設定されているすべてのキューのリアルタイム・モニターを有効にするには、以下のコマンドを実行します。

  ```
  ALTER QMGR MONQ(collection_level)
  ```

  ここで *collection_level* は、キューのモニター・データの収集レベルを指定します。使用環境の要件に応じて、LOW、MEDIUM、または HIGH に設定できます。
- 個々のキューのリアルタイム・モニターを有効にするには、以下のコマンドを実行します。

  ```
  ALTER QLOCAL(queue_name) MONQ(collection_level)
  ```

  ここで、*queue_name* はキューの名前であり、*collection_level* はキューのモニター・データの収集レベルを指定します。使用環境の要件に応じて、LOW、MEDIUM、または HIGH に設定できます。

タスクの結果
データは、WebSphere MQ エージェントの開始後に、「キューに対する最も古いメッセージの経過時間」グループ・ウィジェットに表示できます。

キュー・マネージャーのイベント・モニターの有効化

このタスクについて

イベント・モニターは、IBM MQ ネットワークをモニターするために使用可能なモニター手法の 1 つです。キュー・マネージャーによる特定タイプのイベントの発行を有効にすると、該当するイベントが発生した場合にイベント・メッセージがイベント・キューに配置されます。これにより、WebSphere MQ エージェントでそのようなイベント・メッセージをモニターして表示できます。

デフォルトのキュー・マネージャー構成では、以下のタイプのイベントはモニターされず、表示されません。**ALTER QMGR** コマンドを使用して、キュー・マネージャーによる以下のイベントの生成を有効にして、Application Performance ダッシュボードで表示できるようにします。

- チャネル・イベント
- パフォーマンス・イベント

手順

以下のコマンドを使用して、キュー・マネージャーによる対象イベントの生成を有効にします。

- チャネル・イベントを生成するには `ALTER QMGR CHLEV(ENABLED)` を実行します。
- パフォーマンス・イベントを生成するには `ALTER QMGR PERFMEV(ENABLED)` を実行します。

タスクの結果
モニター対象イベントは、WebSphere MQ エージェントの開始後に、「キュー・マネージャー・イベント」グループ・ウィジェットに表示できます。

MQI アプリケーション・アクティビティー・トレースの有効化

このタスクについて
トランザクション・トラッキング・データがミドルウェアおよびトポロジー・ダッシュボードで表示されるようにするには、MQI アプリケーション・アクティビティー・トレースをキュー・マネージャーで有効にする必要があります。

手順

- MQI アプリケーション・アクティビティー・トレース情報の収集を有効にするには、以下の MQSC コマンドを発行します。

```
ALTER QMGR ACTVTRC(ON)
```

リモート・モニターのセキュリティー設定

このタスクについて
WebSphere MQ エージェントを使用してリモート・キュー・マネージャーをモニターするには、IBM MQ (WebSphere MQ) のセキュリティー設定によってリモート・システムのチャネルを通じたモニター・データの収集が阻害されないようにする必要があります。

リモート・モニターの簡単なセキュリティー設定の例を以下の手順に示します。チャネル認証レコードを使用すれば、接続システムに与えるアクセス権限をチャネル・レベルで細かく制御できるようになります。詳しくは、IBM MQ セキュリティー・メカニズムの資料を参照してください。

手順

1. チャネル認証を無効にするには、以下の MQSC コマンドを実行します。

```
ALTER QMGR CHLAUTH(DISABLED) CONNAUTH(' ')
```

2. チャネル設定を以下のように変更します。ここで *channel_for_remote_monitor* は、リモート・モニターに使用するチャネルの名前です。
   - Linux AIX

     ```
     ALTER CHANNEL(channel_for_remote_monitor) CHLTYPE(SVRCONN) MCAUSER('mqm')
     ```

   - Windows

     ```
     ALTER CHANNEL(channel_for_remote_monitor) CHLTYPE(SVRCONN) MCAUSER(MUSR_MQADMIN)
     ```

3. セキュリティー設定を更新します。

```
REFRESH SECURITY
```

WebSphere MQ エージェントの構成

WebSphere MQ エージェントで IBM MQ (WebSphere MQ) 環境のモニターを開始するには、事前にこのエージェントにインスタンス名を割り当て、エージェントを構成する必要があります。

始める前に

- エージェント・ユーザー ID が IBM MQ (WebSphere MQ) オブジェクトにアクセスするための適切な許可を備えていることを確認します。まだそうなっていない場合は、932 ページの『エージェントを実行するためのユーザー ID の許可』の説明に従います。
- IBM MQ (WebSphere MQ) を構成して、必要なデータ収集を有効にします。まだ行っていない場合は、934 ページの『データ使用可能化のための IBM MQ (WebSphere MQ) の構成』を参照してください。
- WebSphere MQ エージェントでモニターするキュー・マネージャーの名前を指定する必要があります。該当するキュー・マネージャー名が分からない場合は、IBM MQ (WebSphere MQ) 管理者に問い合わせてください。あるいは、IBM MQ (WebSphere MQ) インストール・ディレクトリー内の `bin` ディレクトリーから **dspmq** コマンドを発行して、キュー・マネージャーのリストを取得します。返された QMNAME 値が、WebSphere MQ エージェントの構成時に指定する必要があるものです。

このタスクについて

WebSphere MQ エージェントは複数インスタンス・エージェントです。最初のインスタンスを作成し、エージェントを手動で開始する必要があります。

UNIX または Linux システムでは、対話式または非対話式でエージェントを構成することを選択できます。Windows システムでは、非対話式でのみ、エージェントを構成できます。

- 対話式でエージェントを構成するには、構成スクリプトを実行してプロンプトに応答します。936 ページの『対話式の構成』を参照してください。
- 対話操作なしでエージェントを構成するには、サイレント応答ファイルを編集してから構成スクリプトを実行します。937 ページの『サイレント構成』を参照してください。

重要 : Cloud APM で提供される WebSphere MQ エージェントと同じシステムに、ITCAM for Applications 製品の 1 つのコンポーネントとして提供される Monitoring Agent for WebSphere MQ もインストールした場合には、これらをシステム上の同じキュー・マネージャーをモニターするために使用しないでください。

対話式の構成

手順

スクリプトを実行し、プロンプトに応答を入力してエージェントを構成する場合は、以下の手順を実行します。

1. 次のコマンドを入力して、エージェント・インスタンスを作成します。

```
install_dir/bin/mq-agent.sh config instance_name
```

ここで、*instance_name* は、インスタンスに指定する名前です。

2. Queue Manager Name の入力を求めるプロンプトが出されたら、モニターするキュー・マネージャーの名前を指定します。
3. Agent Name の入力を求めるプロンプトが出されたら、管理対象システム名の中間修飾子として使用するエージェント名を指定します。Enter キーを押してこのパラメーターの指定をスキップしないでください。

   要確認 : このエージェント名は、エージェント・インスタンス名とは異なります。エージェント・インスタンス名は、各エージェントの構成ファイルを区別するためにエージェント構成ファイル名で使用されます (例えば *hostname*_mq_*instancename*.cfg など)。エージェント名は、固有の管理対象システム名を作成するための短い ID として使用されます。固有の管理対象システム名が必要になる場合については、939 ページの『複数のキュー・マネージャーに対する固有の管理対象システム名の指定』を参照してください。
4. リモート・キュー・マネージャーをモニターする場合は、以下の構成パラメーターを指定します。ローカル・キュー・マネージャーをモニターする場合は、Enter キーを押して続行します。
   - Connection Name: リモート・モニターの接続名。形式は *IP_address* (*port_number*) です (例: 127.0.0.1(1414))。初めてエージェント・インスタンスを構成する場合は、Enter キーを押して、デフォルト (ヌル) を受け入れることができます。適切な接続名を自動的に検出できます。
   - Channel: リモート・データ収集に使用するチャネルの名前。初めてエージェント・インスタンスを構成する場合は、Enter キーを押して、デフォルト (ヌル) を受け入れることができます。SYSTEM.DEF.SVRCONN チャネルが使用されます。

   制約事項 : リモートのキュー・マネージャーのエラー・ログはモニターできません。エージェントがリモート・キュー・マネージャーをモニターしているときは、「MQ エラーの詳細」ダッシュボードにデータが表示されません。
5. WebSphere MQ ライブラリー・パスを求めるプロンプトが出されたら、Enter キーを押してデフォルト値を受け入れます。デフォルト値は、WebSphere MQ エージェントによって自動的にディスカバーされた IBM MQ (WebSphere MQ) の 64 ビットのライブラリー・パスです。デフォルト値が表示されない場合は、必ず IBM MQ (WebSphere MQ) の 64 ビットのライブラリーのパスを指定してから先に進んでください。
   64-bit ビット・ライブラリー・パスの例は、Linux システムの場合、/opt/mqm8/lib64 です。
6. エージェントを始動するために、以下のコマンドを入力します。

```
install_dir/bin/mq-agent.sh start instance_name
```

サイレント構成

手順

サイレント応答ファイルを編集し、対話操作なしでスクリプトを実行することでエージェントを構成するには、以下の手順を実行します。

1. テキスト・エディターで mq_silent_config.txt ファイルを開きます。
   - Linux AIX *install_dir*/samples/mq_silent_config.txt
   - Windows *install_dir*¥tmaitm6_x64¥samples¥mq_silent_config.txt

   ここで、*install_dir* は WebSphere MQ エージェント のインストール・ディレクトリーです。
2. 必須: **QMNAME** には、モニターするキュー・マネージャーの名前を指定します。
3. 必須: **AGTNAME** には、管理対象システム名の中間修飾子として使用するエージェント名を指定します。

   要確認 : このエージェント名は、エージェント・インスタンス名とは異なります。エージェント・インスタンス名は、各エージェントの構成ファイルを区別するためにエージェント構成ファイル名で使用されます (例えば *hostname*_mq_*instancename*.cfg など)。エージェント名は、固有の管理対象システム名を作成するための短い ID として使用されます。固有の管理対象システム名が必要になる場合につ

いては、939 ページの『複数のキュー・マネージャーに対する固有の管理対象システム名の指定』を参照してください。

4. リモート・キュー・マネージャーをモニターする場合は、以下の構成パラメーターを指定します。
   - **CONNAME**: リモート・モニターの接続名。形式は *IP_address*（*port_number*）です (例: 127.0.0.1(1414))。
   - **CHANNEL**: リモート・データ収集に使用するチャネルの名前。指定しない場合は、SYSTEM.DEF.SVRCONN チャネルが使用されます。

   制約事項 : リモートのキュー・マネージャーのエラー・ログはモニターできません。エージェントがリモート・キュー・マネージャーをモニターしているときは、「MQ エラーの詳細」ダッシュボードにデータが表示されません。
5. オプション: **WMQLIBPATH** には、WebSphere MQ (WebSphere MQ) の 64 ビット・ライブラリー・パスを指定します。例えば、/opt/mqm8/lib64 です。値が指定されていない場合、エージェント構成時にパスを自動的に検出できます。
6. mq_silent_config.txt ファイルを保存して閉じてから、コマンド・ラインで以下のコマンドを実行します。
   - Linux AIX *install_dir*/bin/mq-agent.sh config *instance_name path_to_responsefile*
   - Windows *install_dir*¥BIN¥mq-agent.bat config *instance_name* "*path_to_responsefile*"

   ここで、*instance_name* は構成するインスタンスの名前であり、*path_to_responsefile* はサイレント応答ファイルの絶対パスです。

   要確認 : Windows システムでは、特にパスに特殊文字が含まれているときは、サイレント応答ファイルへのパスを囲む二重引用符 ("") を省略しないでください。

   例えば、応答ファイルがデフォルト・ディレクトリーにある場合は、以下のコマンドを実行します。
   - Linux AIX

     ```
     /opt/ibm/apm/agent/bin/mq-agent.sh config instance_name
     /opt/ibm/apm/agent/samples/mq_silent_config.txt
     ```
   - Windows

     ```
     C:¥IBM¥APM¥BIN¥mq-agent.bat config instance_name
     "C:¥IBM¥APM¥tmaitm6_x64¥samples¥mq_silent_config.txt"
     ```
7. エージェントを始動するために、以下のコマンドを入力します。
   - Linux AIX

     ```
     install_dir/bin/mq-agent.sh start instance_name
     ```
   - Windows

     ```
     install_dir¥bin¥mq-agent.bat start instance_name
     ```

タスクの結果

これで、Cloud APM コンソールにログインし、アプリケーション・エディターを使用して WebSphere MQ エージェント・インスタンスを Application Performance ダッシュボードに追加できます。Cloud APM コンソールを開始する方法については、977 ページの『Cloud APM コンソールの始動』を参照してください。アプリケーション・エディターの使用について詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

次のタスク

- キュー・マネージャーで MQI アプリケーション・アクティビティーのトレース 情報収集を有効にした場合、「**エージェント構成**」ページを使用して、モニター対象のキュー・マネージャーのトランザクション・トラッキング・データを収集するように WebSphere MQ エージェントを構成します。941 ページの『WebSphere MQ エージェントのトランザクション・トラッキングの構成』を参照してください。エージェントが「**エージェント構成**」ページに表示されない場合は、Cloud APM サーバーを再始動してください。
- モニター要件によっては、複数のモニター・エージェントを区別するために固有の管理対象システム名が必要になることがあります。`mq-agent.sh config` コマンドの「`Agent Name`」オプションを使用して、管理対象システム名の中間修飾子を指定します。939 ページの『複数のキュー・マネージャーに対する固有の管理対象システム名の指定』を参照してください。
- リモート・モニター用に WebSphere MQ エージェントを構成する場合、エージェント・インスタンスの作成後にいくつかの手動構成を実行する必要があります。手順については、次のトピックを参照してください。
  - 943 ページの『MQ アプライアンス 上のキュー・マネージャーのリモートでのモニター』
  - 944 ページの『MQ アプライアンス 上の HA キュー・マネージャーのリモートでのモニター』

複数のキュー・マネージャーに対する固有の管理対象システム名の指定

同じ Cloud APM サーバーに接続するさまざまなモニター・エージェントを区別するために、固有の管理対象システム名が必要になる場合があります。サイレント応答ファイルで **AGTNAME** パラメーターを使用するか、`mq-agent.sh config` コマンドで **`Agent Name`** オプションを使用して、管理対象システム名で使用する中間修飾子を指定します。

このタスクについて

WebSphere MQ エージェント では、始動時に、以下の管理対象システムを登録します。

```
monitoredqueuemanagername:agentname:MQ
```

ここで、

- *monitoredqueuemanagername* は、エージェントによってモニターするキュー・マネージャーの名前です。
- *agentname* は、管理対象システム名の中間修飾子です。*agentname* 値が指定されない場合、値は使用されません。

エージェント名 (agentname) の値を指定すると、以下の場合に役立ちます。

- 同じ名前を持つ複数のキュー・マネージャーをそれぞれ別のノードで実行しているサイトの場合は、各キュー・マネージャーにエージェント名を指定して、WebSphere MQ エージェントが固有の管理対象システム名を作成できるようにします。
- 管理対象システム名が 32 文字を超えると、切り捨てが原因で、2 つの異なるキュー・マネージャー名が同じ名前に解決されることがあります。キュー・マネージャーの管理対象システム名を区別するために、キュー・マネージャーごとにエージェント名を指定してください。
- ホスト名およびキュー・マネージャー名以外の値、例えば高可用性クラスター名などでキュー・マネージャー名をグループ化して識別する場合。
- 同じ Cloud APM サーバーに接続されている複数のエージェントが、異なる複数のホスト上の同じ名前のキュー・マネージャーをモニターできるようにする場合。

対話式の構成

手順

`mq-agent.sh config` コマンドで **`Agent Name`** オプションを使用するには、以下の手順を実行します。

1. コマンド行で、次のコマンドを実行して、WebSphere MQ エージェントの構成を開始します。

```
./mq-agent.sh config instance_name
```

ここで、*instance_name* は開始したインスタンスの名前です。

2. オプションに従って、エージェント・インスタンスを構成します。

   キュー・マネージャーの名前は必須です。その他のオプションについては、変更する必要がない場合は、デフォルト値を使用します。

3. Agent Name オプションが表示されたら、管理対象システム名の中間修飾子を指定します。

   要確認 : 完全な管理対象システム名は *monitoredqueuemanagername*:*agentname*:MQ です。 完全な管理対象システム名の最大長は 32 文字であるため、中間修飾子 *agentname* の最大長は、キュー・マネージャー名の長さによって変わります。Agent Name オプションに指定された値が最大長を超えた場合、*agentname* の値は 8 文字以上を残して切り捨てられます。

   例えば、AIX1 ノードで PERSONNEL という名前のキュー・マネージャーをモニターし、LINUX2 という名前のノードで PERSONNEL という名前の別のキュー・マネージャーをモニターする場合、まず次のコマンドを AIX1 ノードに対して実行します。

   ```
   ./mq-agent.sh config PERSONNEL
   ```

   次に、*Agent Name* オプションが表示されたら、次のようにエージェント名を指定します。

   ```
   Agent Name (default is: ): AIX1
   ```

   LINUX2 ノードで PERSONNEL キュー・マネージャーを同時にモニターするには、まず次のコマンドを実行します。

   ```
   ./mq-agent.sh config PERSONNEL
   ```

   次に、次のようにエージェント名を指定します。

   ```
   Agent Name (default is: ): LINUX2
   ```

   要確認 : コード・サンプルで Agent Name オプションに使用したエージェント・ノードの名前は、単なる説明用です。Agent Name オプションには他のストリングも指定できます。

サイレント構成

手順

サイレント応答ファイルで **AGTNAME** パラメーターを使用するには、以下の手順を実行します。

1. テキスト・エディターでサイレント 応答ファイル mq_silent_config.txt を開きます。
2. **AGTNAME** パラメーターにエージェント名を指定します。

   要確認 : 完全な管理対象システム名は *monitoredqueuemanagername*:*agentname*:MQ です。 完全な管理対象システム名の最大長は 32 文字であるため、中間修飾子 *agentname* の最大長は、キュー・マネージャー名の長さによって変わります。**AGTNAME** パラメーターに指定された値が最大長を超えた場合、*agentname* の値は 8 文字以上を残して切り捨てられます。

3. mq_silent_config.txt ファイルを保存して閉じてから、コマンド・ラインで以下のコマンドを実行します。

   ```
   install_dir/BIN/mq-agent.sh config instance_name path_to_responsefile
   ```

   ここで、*instance_name* は構成するインスタンスの名前であり、*path_to_responsefile* はサイレント応答ファイルの絶対パスです。

次のタスク

Cloud APM コンソール にログインします。前の MSN を持つエージェント・インスタンスが引き続きオフラインとして表示される場合、アプリケーションを編集して、そのエージェント・インスタンスを削除し、割り当て済みのエージェント名を使用して新しいエージェント・インスタンスを追加してください。

WebSphere MQ エージェントのトランザクション・トラッキングの構成

WebSphere MQ エージェントの「**エージェント構成**」ページでデータ収集を有効にすると、IBM MQ (WebSphere MQ) のトランザクション・トラッキング・データをミドルウェア・ダッシュボードおよびトポロジー・ダッシュボードに表示できます。

始める前に

- MQI アプリケーション・アクティビティー・トレース情報収集がキュー・マネージャーで有効になっていることを確認します。WebSphere MQ エージェントを構成して開始する前にこれを行っていない場合は、935 ページの『MQI アプリケーション・アクティビティー・トレースの有効化』の説明に従ってから、エージェントを再始動します。
- 使用する IBM MQ (WebSphere MQ) のバージョンが、トランザクション・トラッキング機能でサポートされていることを確認します。サポートされている IBM MQ (WebSphere MQ) に関する最新情報については、Detailed System Requirements Report for the WebSphere MQ エージェント で、前提条件ステートメントを参照してください。
- WebSphere MQ エージェントがキュー・マネージャーをモニターするように構成されていることを確認します。説明については、936 ページの『WebSphere MQ エージェントの構成』を参照してください。

要確認 : 必ず WebSphere MQ エージェントを最新バージョンにアップグレードしておいてください。 一部のウィジェット (メッセージ・ボリューム・ウィジェットなど) でデータを表示するには、エージェントをアップグレードして、トランザクション・トラッキングを構成して有効化する必要があります。

手順

WebSphere MQ エージェント 用のトランザクション・トラッキングを構成するには、以下の手順を実行します。

1. ナビゲーション・バーで、 「**システム構成**」 > 「**エージェント構成**」をクリックします。
   「**エージェント構成**」ページが表示されます。
2. 「**WebSphere MQ**」タブをクリックします。
3. モニターするキュー・マネージャーのチェック・ボックスを選択し、「**アクション**」リストから以下のいずれかのアクションを実行します。
   - トランザクション・トラッキングを有効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**有効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「有効」に更新されます。

     ヒント : 別名キューとリモート・キューのトラッキングは、デフォルトで有効になっています。トラッキングされるデータの量を減らすために、「**アクション**」リストから「**別名キュー・トラッキングの設定 (Set Alias Queue Tracking)**」 > 「**無効**」をクリックして、別名キューとリモート・キューのトラッキングを無効にすることができます。別名キューとリモート・キューのトラッキングが無効にされると、別名キューとリモート・キューは「トランザクション・トポロジー」ビューから除去されます。
   - トランザクション・トラッキングを無効にするには、「**トランザクション・トラッキングの設定 (Set Transaction Tracking)**」 > 「**無効**」をクリックします。「**トランザクション・トラッキング**」列の状況が「無効」に更新されます。

タスクの結果

選択したキュー・マネージャーをトラッキングするように WebSphere MQ エージェントを構成しました。トランザクション・トラッキング・データは、ミドルウェア・ダッシュボードおよびトポロジー・ダッシュボードに表示できます。詳しくは、100 ページの『Application Performance ダッシュボードへの ミドルウェア・アプリケーションの追加』を参照してください。

キューおよびチャネルの長期間ヒストリーのためのデータ収集の有効化

デフォルトでは、キューの長期間ヒストリーおよびチャネルの長期間ヒストリーは収集されず、どの事前定義のダッシュボードおよびグループ・ウィジェットにも表示されません。ただし、エージェントによる

長期間ヒストリー・データの収集を有効にし、その後、「**属性の詳細**」タブを使用して、収集されたデータを照会できます。

始める前に

WebSphere MQ エージェントがインストールおよび構成されていることを確認します。詳しくは、936 ページの『WebSphere MQ エージェントの構成』を参照してください。

このタスクについて

チャネルの長期間ヒストリー・データまたはキューの長期間ヒストリー・データは、個別のチャネルまたはキューの問題を検出するのに役立ちます。

Tivoli Data Warehouse のユーザーの場合、エージェントは、長期間ヒストリー・データを詳細処理用に Tivoli Data Warehouse にも送信できます。

手順

以下のステップを実行して、WebSphere MQ エージェントによるキューの長期間ヒストリー・データおよびチャネルの長期間ヒストリー・データの収集を有効にします。

1. テキスト・エディターで以下のエージェント環境ファイルを開きます。`mq.environment` ファイルが存在しない場合、自分で作成します。
   - Linux AIX *install_dir*/config/mq.environment
   - Windows *install_dir*¥Config¥KMQENV_*instance*

   ここで、
   - *install_dir* は、エージェントのインストール・ディレクトリーです。デフォルトは、`/opt/ibm/apm` (Linux および AIX システムの場合) および `C:¥IBM¥APM` (Windows システムの場合) です。
   - *instance* は、エージェント・インスタンス名です。
2. **LH_COLLECTION** の値を ENABLED に設定して、データ収集を有効にします。

   ```
   LH_COLLECTION=ENABLED
   ```
3. オプション: Tivoli Data Warehouse のユーザーで、エージェントで収集したデータを Tivoli Data Warehouse に送信するようにする場合は、**LH_PVTHISTORY** の値を ENABLED に設定します。

   ```
   LH_PVTHISTORY=ENABLED
   ```

   要確認 : このオプションを有効にするには、収集されたデータを Tivoli Data Warehouse に送信する必要がある場合のみにしてください。
4. 変更を保存し、エージェントを再始動します。

タスクの結果

WebSphere MQ エージェントが、キューの長期間ヒストリー・データおよびチャネルの長期間ヒストリー・データの収集を開始します。LH_PVTHISTORY=ENABLED を指定した場合、収集された長期間ヒストリー・データは、Tivoli Data Warehouse にも送信できます。

次のタスク

「**属性の詳細**」タブを使用して、構成されているエージェント・インスタンスのダッシュボードで、収集されたデータを表示します。「**データ・セット**」リストから「**Channel_Long-Term_History**」または「**Queue_Long-Term_History**」を選択します。「**属性の詳細**」タブについて詳しくは、1093 ページの『カスタムのグラフまたは表のページの作成』を参照してください。

IBM MQ のキュー・マネージャーのキュー統計モニターの有効化

キューの統計はデフォルトでは収集されないため、どの事前定義ダッシュボードやグループ・ウィジェットにも表示されません。しかし、エージェントがキュー・マネージャーの統計を収集できるようにすると、収集されたデータを表示できます。

始める前に

WebSphere MQ エージェントがインストールおよび構成されていることを確認します。詳しくは、936 ページの『WebSphere MQ エージェントの構成』を参照してください。

手順

以下のステップを実行して、WebSphere MQ エージェント が統計データを収集できるようにします。

1. キュー・マネージャーが統計情報を収集するように構成します。以下の MQSC コマンドを実行します。

   ```
   ALTER QMGR STATQ(ON)
   ```

2. アカウンティング・データを収集する期間を設定します。次のコマンドを実行します。

   ```
   ALTER QMGR STATINT(n)
   ```

   ここで、**n** はアカウンティング・データを収集する秒数です。

3. 特定のキューに対する統計情報の収集を有効にします。以下の MQSC コマンドを実行します。

   ```
   ALTER QLOCAL(queue_name) STATQ(QMGR)
   ```

   ここで、**queue_name** は統計情報を収集するキューの名前です。

次のタスク

以下のいずれかのメソッドを使用して、MQ キューの統計モニター・データを表示します。

- 「**MQ_Queue_Statistics**」データ・セットの「**属性の詳細**」タブからモニター・データを確認します。「**属性の詳細**」タブについて詳しくは、1093 ページの『カスタムのグラフまたは表のページの作成』を参照してください。
- 「**期限切れメッセージ数**」と「**MQ_Queue_Statistics**」のその他のメトリックに基づいてしきい値を定義します。しきい値について詳しくは、978 ページの『しきい値とリソース・グループ』を参照してください。

MQ アプライアンス上のキュー・マネージャーのリモートでのモニター

WebSphere MQ エージェントを使用して、MQ アプライアンス環境のリモート・キュー・マネージャーをモニターできます。

始める前に

- サポート対象のプラットフォームに WebSphere MQ エージェントをインストールします。
- IBM MQ クライアントをインストールします。MQ クライアントのバージョンは、リモート MQ キュー・マネージャーと同じである必要があります。

手順

1. リモート・キュー・マネージャーへの接続を設定します。リモート・キュー・マネージャーで、サーバー接続チャネルと、モニター・エージェントとの通信に使用されるリスナーを定義します。次のコマンドを実行します。

   ```
   M2000# mqcli
   M2000(mqcli)#runmqsc qmgr_remote
   > DEFINE LISTENER(listener) TRPTYPE(TCP) PORT(port_NO)
   > DEFINE CHANNEL(chl_name)CHLTYPE(SVRCONN) TRPTYPE(TCP)
   CONNAME('host_IP(port_NO)') QMNAME(Qmgr_remote)
   > END
   ```

ここで、

- *qmgr_remote* は、リモート・キュー・マネージャーの名前です。
- *listener* は、リモート・キュー・マネージャー上のリスナーの名前です。
- *port_NO* は、リスナーに使用されるポート番号です。
- *chl_name* は、サーバー・チャネルとクライアント・チャネルの両方に割り当てる名前です。
- *host_IP* は、リモート・システムの IP アドレスです。

2. リモート・システムで次のコマンドを実行することで、自動的に開始するようにリスナーを構成し、リモート・キュー・マネージャーでリスナーを開始します。

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_remote
> ALTER LISTENER(listener) TRPTYPE(tcp) CONTROL(qmgr_remote)
> START LISTENER(listener)
> END
```

3. MQ エージェント・インスタンスの開始に使用するユーザー ID に対して、チャネル認証の設定が適切に構成されていることを確認します。詳しくは、IBM MQ Appliance Knowledge Center の Setting up a queue manager to accept client connections を参照してください。
4. 936 ページの『WebSphere MQ エージェントの構成』 の手順に従ってリモート・モニター用に WebSphere MQ エージェントのインスタンスを作成し、「**Remote Monitoring Settings**」の後に出されるプロンプトにリモート・キュー・マネージャーの接続情報を指定します。

```
Remote Monitoring Settings (For a local queue manager, just press Enter in
this section) :
Connection name for remote monitoring, for example: 192.168.1.1(1415)
Connection Name (default is: null):
Channel name for remote monitoring, SYSTEM.DEF.SVRCONN is as default.
Channels (default is: null):
```

5. WebSphere MQ エージェントのインスタンスを開始します。

MQ アプライアンス上の HA キュー・マネージャーのリモートでのモニター

MQ アプライアンス上の HA キュー・マネージャーをリモートでモニターするには、2 つの選択肢があります。1 つは、単一のエージェント・インスタンスを使用してアクティブなキュー・マネージャーがあるいずれかのシステムに接続することです。もう 1 つの選択肢は、キュー・マネージャーが実行されている可能性がある各アプライアンスに対して別個のエージェント・インスタンスを使用することです。

このタスクについて

ここでは、2 つめの選択肢についてのみ説明します。異なるエージェント・インスタンスを使用するには、Linux システムまたは UNIX システムに、2 つの WebSphere MQ エージェントのインストール済み環境が必要です。Windows システムでは、必要なエージェントのインストール済み環境は 1 つのみで、別個のエージェント・インスタンスを作成できます。

手順

- Linux AIX

  Linux システムまたは UNIX システムにインストールされている WebSphere MQ エージェントをリモート・モニターのために使用するには、次の手順を実行します。

  a) WebSphere MQ エージェントを、システム上の異なるディレクトリーにインストールします。

  b) インストールされた各 WebSphere MQ エージェントのインスタンスを作成します。説明については、936 ページの『WebSphere MQ エージェントの構成』 を参照してください。

  c) 各エージェント・インスタンスの構成ファイルを変更して、内容を次の行で置換することで、リモート・モニターを有効にします。

```
SET GROUP NAME (GROUP1) -
DEFAULT(YES) -
RETAINHIST(120) -
COMMAND (YES) -
MSGACCESS(DESC) -
```

```
EVENTS(REMOVE) -
ACCOUNTINGINFO(REMOVE) -
STATISTICSINFO(REMOVE)

SET MANAGER NAME(qmgr_name) REMOTE(YES)
SET AGENT NAME(agentID)
SET QUEUE NAME(*) MGRNAME(qmgr_name) QDEFTYPE(PREDEFINED)
SET CHANNEL NAME(*) MGRNAME(qmgr_name)
PERFORM STARTMON SAMPINT(300) HISTORY(NO)
```

ここで、

- *qmgr_name* は、HA キュー・マネージャーの名前です。
- *agentID* は、キュー・マネージャー・システムを識別するための ID です。通常、HA キュー・マネージャーが実行されているリモート・システムのホスト名または IP アドレスです。

構成ファイル名とパスは、*install_dir*/config/*hostname*_mq_*qmgr_name*.cfg です。

d) 1 次キュー・マネージャーと WebSphere MQ エージェントとの間、および 2 次キュー・マネージャーと 1 次キュー・マネージャーがインストールされているリモート・システム上の WebSphere MQ エージェントとの間に、クライアント・チャネルとサーバー・チャネルのペアを作成します。

要確認 : 次のステップに進む前に、次のコマンドをすべて実行する必要があります。

a. 1 次キュー・マネージャーに対して次のコマンドを実行します。

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_primary
>DEFINE LISTENER(listener_primary) TRPTYPE(TCP) PORT(port_no_primary)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_primary)') QMNAME(qmgr_primary)
```

ここで、

- *qmgr_primary* は、1 次キュー・マネージャーの名前です。
- *listener_primary* は、1 次キュー・マネージャーのリスナーの名前です。
- *port_no_primary* は、リスナーによって使用されるポート番号です。
- *chl_name_primary* は、サーバー・チャネルとクライアント ・チャネルの両方に割り当てる名前です。
- *host_IP* は、1 次キュー・マネージャーがインストールされているシステムの IP アドレスです。

b. 1 次キュー・マネージャー上で 2 次キュー・マネージャーに対して次のコマンドを実行します。これは、2 次キュー・マネージャーの接続情報を 1 次キュー・マネージャーのクライアント ・チャネル定義テーブル・ファイルに追加するためです。これにより、1 次キュー・マネージャーがフェイルオーバーしたときに同じエージェントが 2 次キュー・マネージャーに自動的に接続できます。

```
>DEFINE LISTENER(listener_secondary) TRPTYPE(TCP) PORT(port_no_secondary)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_secondary)') QMNAME(qmgr_secondary)
```

ここで、

- *qmgr_secondary* は、リモート・システム上の 2 次キュー・マネージャーの名前です。これは、1 次キュー・マネージャー名と同じです。
- *listener_secondary* は、2 次キュー・マネージャーのリスナーの名前です。
- *port_no_secondary* は、リスナーによって使用されるポート番号です。
- *chl_name_secondary* は、サーバー・チャネルとクライアント ・チャネルの両方に割り当てる名前です。
- *host_IP* は、2 次キュー・マネージャーがインストールされているシステムの IP アドレスです。

c. 最後に、次のコマンドを実行します。

```
>END
>EXIT
```

e) 1 番目の MQ アプライアンスで WebSphere MQ エージェント・インスタンスのクライアント・チャネル定義テーブル・ファイル (AMQCLCHL.TAB) を作成します。

a. 以下の **runmqsc** コマンドまたは **runmqsc -n** コマンドを使用して、1 番目の MQ アプライアンスでキュー・マネージャーの AMQCLCHL.TAB ファイルを作成します。

```
runmqsc -n
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance1(port_no_primary)') QMNAME(qmgr_name)
```

ここで、*host_IP_appliance1* は、1 番目の MQ アプライアンスの IP アドレスで、*chl_name_primary* および *port_no_primary* はステップ 4 で定義したものと同じです。

ヒント: デフォルトでは、AMQCLCHL.TAB ファイルは var/mqm/qmgrs/*qmgr_name*/@ipcc ディレクトリーに作成されます。

b. 1 次キュー・マネージャーの WebSphere MQ エージェントがインストールされているシステム上の *agent_install_dir*/*arch*/mq/bin ディレクトリーに 1 次 AMQCLCHL.TAB ファイルを移動します。

f) 2 番目の MQ アプライアンスで WebSphere MQ エージェント・インスタンスのクライアント・チャネル定義テーブル・ファイル (AMQCLCHL.TAB) を作成します。

a. 以下の **runmqsc** コマンドまたは **runmqsc -n** コマンドを使用して、2 番目の MQ アプライアンスでキュー・マネージャーの AMQCLCHL.TAB ファイルを作成します。

```
runmqsc -n
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance2(port_no_secondary)') QMNAME(qmgr_name)
```

ここで、*host_IP_appliance2* は、2 番目の MQ アプライアンスの IP アドレスで、*chl_name_secondary* および *port_no_secondary* はステップ 4 で定義したものと同じです。

b. 2 次キュー・マネージャーの WebSphere MQ エージェントがインストールされているシステム上の *agent_install_dir*/*arch*/mq/bin ディレクトリーに 2 次 AMQCLCHL.TAB ファイルを移動します。

g) エージェント・インスタンスとキュー・マネージャーとの間の接続のセットアップに使用されるユーザー ID に対して、チャネル認証設定が適切に構成されていることを確認します。

h) リモートでモニターされる両方のキュー・マネージャーに対してリスナーを開始し、すべての WebSphere MQ エージェント・インスタンスを開始します。

- Windows

Windows システムにインストールされている WebSphere MQ エージェントをリモート・モニターのために使用するには、次の手順を実行します。

a) WebSphere MQ エージェントを、Windows システムにインストールします。

b) 各 HA キュー・マネージャーに対して 2 つの WebSphere MQ エージェント・インスタンスを作成します。

c) 各エージェント・インスタンスの構成ファイルを変更して、内容を次の行で置換することで、リモート・モニターを有効にします。

```
SET GROUP NAME (GROUP1) -
DEFAULT(YES) -
RETAINHIST(120) -
COMMAND (YES) -
MSGACCESS(DESC) -
EVENTS(REMOVE) -
ACCOUNTINGINFO(REMOVE) -
STATISTICSINFO(REMOVE)

SET MANAGER NAME(qmgr_name) REMOTE(YES)
SET AGENT NAME(agentID)
SET QUEUE NAME(*) MGRNAME(qmgr_name) QDEFTYPE(PREDEFINED)
```

```
SET CHANNEL NAME(*) MGRNAME(qmgr_name)
PERFORM STARTMON SAMPINT(300) HISTORY(NO)
```

ここで、

- *qmgr_name* は、HA キュー・マネージャーの名前です。
- *agentID* は、キュー・マネージャー・システムを識別するための ID です。通常、HA キュー・マネージャーが実行されているリモート・システムのホスト名または IP アドレスです。

ヒント : 構成ファイル名とパスは、*install_dir*¥TMAITM6_x64¥mq_<*instance_name*>.cfg です。

d) 1 次キュー・マネージャーと WebSphere MQ エージェントとの間、および 2 次キュー・マネージャーと 1 次キュー・マネージャーがインストールされているリモート・システム上の WebSphere MQ エージェントとの間に、クライアント・チャネルとサーバー・チャネルのペアを作成します。

要確認 : 次のステップに進む前に、次のコマンドをすべて実行する必要があります。

a. 1 次キュー・マネージャーに対して次のコマンドを実行します。

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_primary
>DEFINE LISTENER(listener_primary) TRPTYPE(TCP) PORT(port_no_primary)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_primary)') QMNAME(qmgr_primary)
```

ここで、

- *qmgr_primary* は、1 次キュー・マネージャーの名前です。
- *listener_primary* は、1 次キュー・マネージャーのリスナーの名前です。
- *port_no_primary* は、リスナーによって使用されるポート番号です。
- *chl_name_primary* は、サーバー・チャネルとクライアント ・チャネルの両方に割り当てる名前です。
- *host_IP* は、1 次キュー・マネージャーがインストールされているシステムの IP アドレスです。

b. 1 次キュー・マネージャー上で 2 次キュー・マネージャーに対して次のコマンドを実行します。これは、2 次キュー・マネージャーの接続情報を 1 次キュー・マネージャーのクライアント ・チャネル定義テーブル・ファイルに追加するためです。これにより、1 次キュー・マネージャーがフェイルオーバーしたときに同じエージェントが 2 次キュー・マネージャーに自動的に接続できます。

```
>DEFINE LISTENER(listener_secondary) TRPTYPE(TCP) PORT(port_no_secondary)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_secondary)') QMNAME(qmgr_secondary)
```

ここで、

- *qmgr_secondary* は、リモート・システム上の 2 次キュー・マネージャーの名前です。これは、1 次キュー・マネージャー名と同じです。
- *listener_secondary* は、2 次キュー・マネージャーのリスナーの名前です。
- *port_no_secondary* は、リスナーによって使用されるポート番号です。
- *chl_name_secondary* は、サーバー・チャネルとクライアント ・チャネルの両方に割り当てる名前です。
- *host_IP* は、2 次キュー・マネージャーがインストールされているシステムの IP アドレスです。

c. 最後に、次のコマンドを実行します。

```
>END
>EXIT
```

e) 各 WebSphere MQ エージェント・インスタンスのクライアント ・チャネル定義テーブル・ファイル (AMQCLCHL.TAB) を作成します。

a. 以下の **runmqsc** コマンドまたは **runmqsc -n** コマンドを使用して、1 番目の MQ アプライアンスでキュー・マネージャーの AMQCLCHL.TAB ファイルを作成します。

```
runmqsc -n
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance1(port_no_primary)') QMNAME(qmgr_name)
```

ここで、*host_IP_appliance1* は、1 番目の MQ アプライアンスの IP アドレスで、*chl_name_primary* および *port_no_primary* はステップ 4 で定義したものと同じです。

b. 以下のように、2 番目の MQ アプライアンスでキュー・マネージャーの AMQCLCHL.TAB ファイルを作成します。

```
runmqsc -n
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance2(port_no_secondary)') QMNAME(qmgr_name)
```

ここで、*host_IP_appliance2* は、2 番目の MQ アプライアンスの IP アドレスで、*chl_name_secondary* および *port_no_secondary* はステップ 4 で定義したものと同じです。

f) 2 つの AMQCLCHL.TAB ファイルを異なる名前 (NODE1.TAB と NODE2.TAB など) に変更します。それらをディレクトリー *install_dir*¥TMAITM6_x64 に転送します。ここで、*install_dir* は WebSphere MQ エージェントのインストール・ディレクトリーです。

g) kmqcma_*instance_name*.ini ファイルを変更して、**MQCHLTAB** の値を各エージェント・インスタンスのクライアント・チャネル定義テーブル・ファイルに設定します。
例えば、kmqcma_instance1.ini ファイルに MQCHLTAB=NODE1.TAB を、kmqcma_instance2.ini ファイルに MQCHLTAB=NODE2.TAB を設定します。

h) Windows レジストリ エディターを開き、**MQCHLTAB** の次のキーを見つけて、AMQCLCHL.TAB から、各エージェント・インスタンスに該当するクライアント・チャネル定義テーブル・ファイル名に変更します。

- **HKEY_LOCAL_MACHINE¥SOFTWARE¥Candle¥KMQ¥Ver730¥*instance1*¥Environment**

  MQCHLTAB=*NODE1.TAB*

- **HKEY_LOCAL_MACHINE¥SOFTWARE¥Candle¥KMQ¥Ver730¥*instance2*¥Environment**

  MQCHLTAB=*NODE2.TAB*

i) エージェント・インスタンスとキュー・マネージャーとの間の接続のセットアップに使用されるユーザー ID に対して、チャネル認証設定が適切に構成されていることを確認します。

j) リモートでモニターされる両方のキュー・マネージャーに対してリスナーを開始し、すべての WebSphere MQ エージェント・インスタンスを開始します。

第 8 章 他の製品およびコンポーネントとの統合

他の製品およびコンポーネントを IBM Cloud Application Performance Management と統合して、堅固なソリューションを実現することができます。

Cloud Event Management との統合

Cloud Event Management は、サービス、アプリケーション、およびインフラストラクチャーのリアルタイム・インシデント管理を提供します。Cloud Event Management と IBM Cloud Application Performance Management の間での統合をセットアップすると、Cloud APM で生成されるイベントはすべて Cloud Event Management に送信されます。

このタスクについて

Webhook URL を Cloud Event Management で構成します。その後で、この Webhook URL を使用してイベントを Cloud Event Management に送信するよう、Cloud APM を構成します。Cloud Event Management について詳しくは、IBM Cloud Event Management Knowledge Center を参照してください。

手順

1. Cloud Event Management の「**管理**」ページで「**統合**」をクリックします。
2. 「**統合の構成**」をクリックします。
3. 「**IBM Cloud Application Performance Management**」タイルに移動し、「**構成**」をクリックします。
4. 統合の名前を入力して、「**コピー**」をクリックし、生成された Webhook URL をクリップボードに追加します。生成された Webhook を保存して、後に構成プロセスで使用できるようにします。例えば、ファイルに保存できます。
5. Cloud APM からのアラート情報の受信を開始するには、Cloud Event Management で「**このソースからのイベント管理を有効にする (Enable event management from this source)**」が「オン（On）」に設定されていることを確認します。
6. 「**保存**」をクリックします。
7. Cloud APM サブスクリプションにログインします。
8. 「**システム構成 (System Configuration)**」 > 「**拡張構成**」 > 「**イベント・マネージャー**」にナビゲートします。 詳しくは、拡張構成を参照してください。
9. Webhook URL を「**Cloud Event Management Webhook**」フィールドに貼り付けます。
10. 「**保存**」をクリックします。

IBM Tivoli Monitoring V6.3 との統合

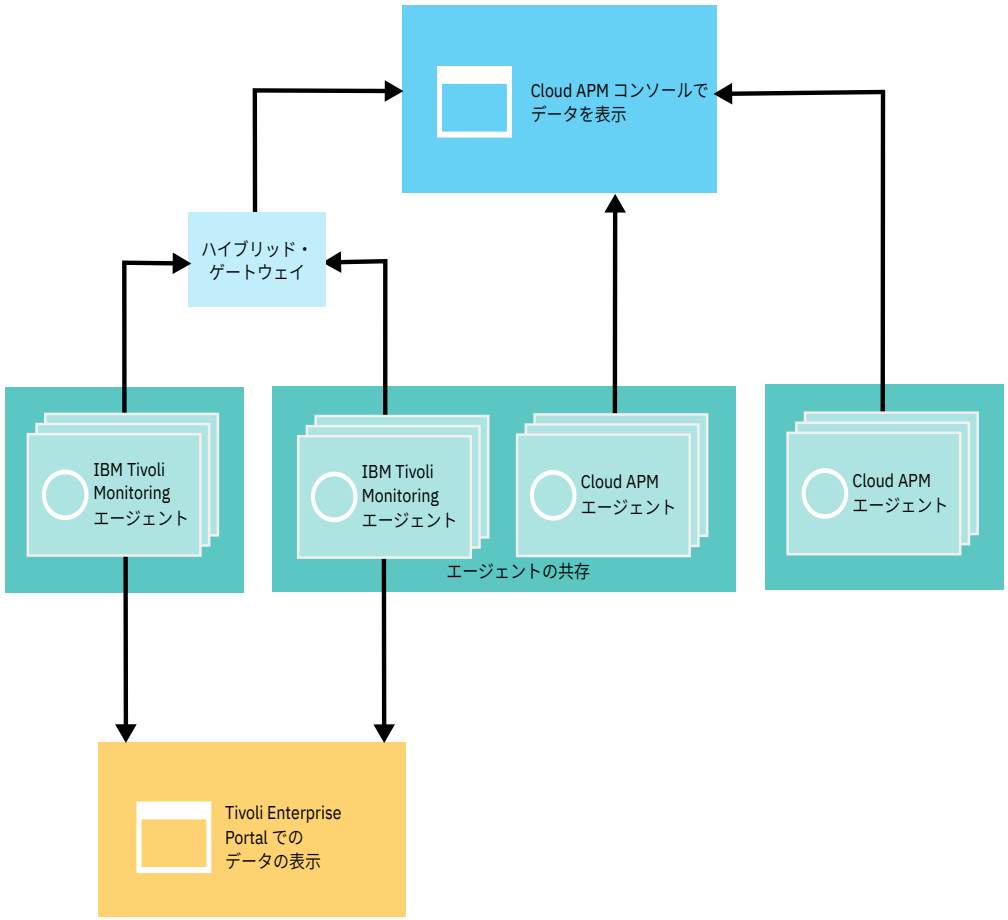
IBM Tivoli Monitoring 製品と IBM Cloud Application Performance Management 製品の両方が含まれている環境では、両製品をいくつかの方法で一緒に使用できます。

IBM Tivoli Monitoring との統合オプションは以下のとおりです。

- IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールして、Application Performance ダッシュボード・ページで 1 つ以上の Tivoli Monitoring ドメインおよび Cloud APM ドメインの管理対象システムを示す統合ビューを提供できます。 エージェントの統合について詳しくは、953 ページの『ハイブリッド・ゲートウェイ』を参照してください。
- Tivoli Monitoring エージェントと Cloud APM エージェントは同じシステムにインストールできます。これらのエージェントが同じコンピューターに共存しているが同じディレクトリーに存在しない場合は、Cloud APM エージェントからのデータを Cloud APM コンソールで使用でき、Tivoli Monitoring エージェントからのデータを Tivoli Enterprise Portal で使用できます。共存するエージェントが同じリソースを

© Copyright IBM Corp. 2014, 2019

モニターしている場合は、一定の制限が課されます。エージェントの共存について詳しくは、950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』を参照してください。



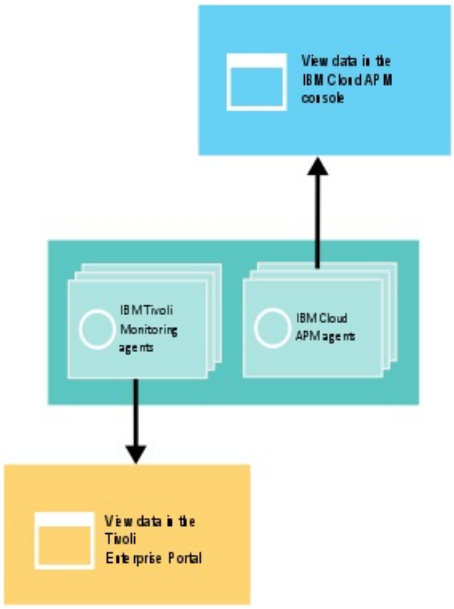
Cloud APM エージェントと Tivoli Monitoring エージェントの共存

エージェントの共存はサポートされています。IBM Tivoli Monitoring エージェントがインストールされているコンピューターに IBM Cloud Application Performance Management エージェントをインストールできます。ただし、同じディレクトリーに両方のエージェント・タイプをインストールすることはできません。

Cloud APM エージェントをバージョン 8 エージェントと呼びます。Tivoli Monitoring エージェントをバージョン 6 または 7 エージェントと呼びます。

エージェントが同じコンピューターに共存している場合、バージョン 8 エージェントからのデータは Cloud APM コンソールで使用可能であり、バージョン 6 または 7 エージェントからのデータは Tivoli Enterprise Portal で使用可能です。

バージョン 6 または 7 エージェントがバージョン 8 エージェントと同じコンピューターに共存していて、別々のリソースをモニターしている場合は、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイと統合すると、どちらのエージェントからのデータも Cloud APM コンソールで使用できます。詳しくは、953 ページの『ハイブリッド・ゲートウェイ』を参照してください。



Tivoli Monitoring エージェントと資料へのリンクを以下の表に示します。

表 *236. Tivoli Monitoring* エージェントの資料へのリンク

Tivoli Monitoring エージェント	**資料へのリンク**
IBM Monitoring Agent for Citrix Virtual Desktop Infrastructure	IBM Tivoli Monitoring for Virtual Environments Knowledge Center
IBM Tivoli Monitoring for Virtual Environments Agent for Cisco UCS	Tivoli Monitoring for Virtual Environments Knowledge Center
IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines	Tivoli Monitoring for Virtual Environments Knowledge Center
IBM Tivoli Monitoring for Virtual Environments Agent for VMware VI	Tivoli Monitoring for Virtual Environments Knowledge Center
IBM Tivoli Monitoring: HMC Base エージェント	IBM Tivoli Monitoring Knowledge Center
IBM Tivoli Monitoring: Linux OS エージェント	Tivoli Monitoring Knowledge Center
IBM Tivoli Monitoring: UNIX OS エージェント	Tivoli Monitoring Knowledge Center
IBM Tivoli Monitoring: Windows OS エージェント	Tivoli Monitoring Knowledge Center
ITCAM Agent for DB2	ITCAM for Applications Knowledge Center
ITCAM Agent for HTTP Servers	ITCAM for Applications Knowledge Center
ITCAM Agent for J2EE	ITCAM for Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft Active Directory エージェント	IBM Tivoli Composite Application Manager for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft Cluster Server エージェント	ITCAM for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft Exchange Server エージェント	ITCAM for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft Hyper-V Server エージェント	ITCAM for Microsoft Applications Knowledge Center

表 236. *Tivoli Monitoring* エージェントの資料へのリンク (続き)	
Tivoli Monitoring エージェント	**資料へのリンク**
ITCAM for Microsoft Applications: Microsoft Internet Information Services エージェント	ITCAM for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Skype for Business Server エージェント	ITCAM for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft .NET Framework エージェント	ITCAM for Microsoft Applications Knowledge Center
ITCAM for Microsoft Applications: Microsoft SharePoint Server エージェント	ITCAM for Microsoft Applications Knowledge Center
Monitoring Agent for Microsoft SQL Server	ITCAM for Microsoft Applications Knowledge Center
ITCAM Agent for SAP Applications	ITCAM for Applications Knowledge Center
ITCAM Agent for WebSphere Applications	IBM Tivoli Composite Application Manager for Application Diagnostics Knowledge Center (バージョン 7.1 までの場合) および ITCAM for Applications Knowledge Center (バージョン 7.2 以降の場合)。
ITCAM Agent for WebSphere DataPower Appliance	IBM Tivoli Composite Application Manager for Applications Knowledge Center
Monitoring Agent for WebSphere Message Broker	ITCAM for Applications Knowledge Center
Monitoring Agent for WebSphere MQ	ITCAM for Applications Knowledge Center
ITCAM Extended Agent for Oracle Database	ITCAM for Applications Knowledge Center
ITCAM Monitoring Agent for SAP HANA Database	ITCAM Monitoring Agent for SAP HANA Database 参照資料
ITCAM Web Response Time Agent	IBM Tivoli Composite Application Manager for Transactions Knowledge Center

共存しているエージェントが同じリソースをモニターしている場合、以下のシナリオはサポートされません。

- バージョン 6 または 7 のエージェントをハイブリッド・ゲートウェイと統合して、Cloud APM コンソールで両方のエージェントからのデータを表示する。例えば、バージョン 6 または 7 のエージェントがハイブリッド・ゲートウェイを介して同じ Cloud APM サーバー に接続されている場合、システム 上で同じブローカーをモニターするためにバージョン 8 の IBM Integration Bus エージェントとバージョン 6 または 7 の Monitoring Agent for WebSphere Message Broker を使用しないでください。

Cloud APM コンソールでデータを表示するためにハイブリッド・ゲートウェイと統合されている Tivoli Monitoring エージェントでリソースをモニターしていて、Cloud APM エージェントでそのリソースをモニターする場合、以下のステップを実行します。

1. Tivoli Monitoring エージェントが含まれているアプリケーションからそのエージェントを削除します。
2. Cloud APM が使用するように構成されている Tivoli Monitoring 管理対象システム・グループから Tivoli Monitoring エージェントを削除します。
3. 少なくとも 24 時間待機してから、Cloud APM エージェントをインストールし、それをアプリケーションに追加します。

同じコンピューターで共存している複数インスタンス・エージェントがハイブリッド・ゲートウェイと統合されていて、同じリソースをモニターする場合、各インスタンスで別の名前を使用して、Cloud APM コンソールで両方のエージェントからのデータを表示します。

データ・コレクターが含まれているエージェントの場合、同じタイプの 2 つのエージェントがサポートされます。Cloud APM コンソールで、詳細診断、リソース、およびトランザクション・トラッキングのデータが表示されます。Tivoli Enterprise Portal にリソース・データが表示されます。以下のエージェントはデータ・コレクターを共有します。

Monitoring Agent for HTTP Server
HTTP Server エージェントは Cloud APM エージェントであり、ITCAM Agent for HTTP Servers は IBM Tivoli Monitoring エージェントです。ご使用の環境に両方のエージェントがある場合、両方のエージェントに対して同じ HTTP サーバー上の両方のデータ・コレクターを構成できます。HTTP Server エージェントの詳細については、275 ページの『HTTP Server モニターの構成』を参照してください。

Microsoft .NET エージェント
Microsoft .NET エージェントの共存について詳しくは、531 ページの『エージェントの共存環境におけるトランザクション・トラッキングの有効化』を参照してください。

WebSphere Applications エージェント
WebSphere Applications エージェントの共存について詳しくは、861 ページの『WebSphere Applications エージェントの構成』 and 862 ページの『エージェントの共存環境用のデータ・コレクターの構成』を参照してください。

ハイブリッド・ゲートウェイ

Cloud APM コンソールで IBM Tivoli Monitoring および OMEGAMON エージェントのモニター・データおよびイベントを表示するには、管理対象システム・グループを作成し、Tivoli Monitoring ドメインに IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールし、Cloud APM コンソールの「**ハイブリッド・ゲートウェイ・マネージャー**」で通信を構成する必要があります。 Tivoli Monitoring および Cloud APM 環境で 1 つ以上のハイブリッド・ゲートウェイをインストールして構成するための計画に役立つ背景情報を確認してください。

ハイブリッド・ゲートウェイをインストールする場所
1 つ以上の Tivoli Monitoring ドメインにハイブリッド・ゲートウェイをインストールできます (ドメインごとに 1 つのハブ Tivoli Enterprise Monitoring Server)。ハイブリッド・ゲートウェイをインストールする場所について詳しくは、『ハイブリッド・ゲートウェイをインストールするための準備』情報を参照してください。ハイブリッド・ゲートウェイのシステム要件 (Tivoli Enterprise Portal Server を含む) については、ハイブリッド・ゲートウェイに関する『Software Product Compatibility Reports』の「**Prerequisites**」タブを参照してください。

サポートされる Tivoli Monitoring エージェントおよび OMEGAMON エージェント
Tivoli Monitoring エージェントをハイブリッド・ゲートウェイに対応させるには、このエージェントが Cloud APM でもサポートされている必要があります (ただし、iOS および OMEGAMON エージェントを除く)。Tivoli Monitoring の使用可能なエージェントおよびバージョンのリストについては、Hybrid Gateway supported agents (APM Developer Center)を参照してください。

Cloud APM コンソールでモニターできる OMEGAMON エージェントのリストについては、IBM Knowledge Center の IBM OMEGAMON for Application Performance Management トピック・コレクション内のご使用のリリース用のトピック『はじめに』を参照してください。

Cloud APM コンソールの Tivoli Monitoring エージェントと OMEGAMON エージェント
Tivoli Monitoring または OMEGAMON 管理対象システム (あるいは両方) を含む Application Performance ダッシュボードで「マイ・コンポーネント」事前定義アプリケーションまたは定義済みアプリケーションを選択すると、すべての管理対象システムの要約状況ダッシュボードを表示できるほか、単一の管理対象システム・インスタンスの詳細ダッシュボードも表示できます。「**カスタム・ビュー**」タブでダッシュボード・ページを作成することもできます。

これらのエージェントのシチュエーション・イベントを「**イベント**」タブで表示できます。ただし、しきい値マネージャーで Tivoli Monitoring および OMEGAMON エージェント用の新しいしきい値を作成することはできません。代わりに、新規シチュエーションを Tivoli Monitoring で作成します。

可能性のあるすべての Tivoli Monitoring および OMEGAMON イベントを「イベント」ダッシュボードで使用できるわけではありません。アプリケーションに追加できるエージェント・ノードからのイベントのみが表示されます。例えば、WebSphere Application Server 用の Tivoli Monitoring エージェントの場合、特定のサーバー・インスタンスに関連付けられているイベントは表示されますが、エージェント全体としてイベントは表示されません。

各 Tivoli Monitoring ドメインの管理対象システムを最大 1500 件表示
Tivoli Monitoring ドメインから表示できる管理対象システムの最大数は、サブノードを含めて 1500 件です。デフォルトでは、この制限は 200 件のシステムになっています。より多くのシステムのサポートを計画できます。説明については、962 ページの『管理対象システムが多い場合の計画』を参照してください。

すべての Tivoli Monitoring ドメインの制限値は、Cloud APM でサポートされる最大値以下にする必要があります。詳しくは、45 ページの『アーキテクチャーの概要』を参照してください。

リソース・モニターのみ
リソース・モニターを Tivoli Monitoring エージェントで使用できます。リソース・モニターについて詳しくは、47 ページの『オファリングとアドオン』および 54 ページの『機能』を参照してください。IBM Cloud Application Performance Management, Advanced サブスクリプションがある場合、トランザクション・トラッキングおよび診断ダッシュボードを Tivoli Monitoring 環境から管理対象システムで使用することはできません。

Tivoli Authorization Policy Server は、Tivoli Monitoring 管理対象システムの可用性に影響します
Tivoli Authorization Policy Server を含む Tivoli Monitoring 環境の場合、ハイブリッド・ゲートウェイから使用可能な管理対象システムは許可ポリシーの影響を受けます。詳しくは、 Tivoli Monitoring Knowledge Center の 役割ベースの許可ポリシーの使用を参照してください。

ビデオ・デモンストレーションについては、Integrating with Tivoli Monitoring - ハイブリッド・ゲートウェイ を参照してください。

ハイブリッド・ゲートウェイをインストールするための準備
IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールするには、まずご使用の環境が正しくセットアップされていることを確認する必要があります。この情報を参照して、ハイブリッド・ゲートウェイのインストール計画に役立ててください。

ハイブリッド・ゲートウェイをインストールする場所
ハイブリッド・ゲートウェイは、IBM Tivoli Monitoring および IBM Cloud Application Performance Management とネットワークで接続された Red Hat Enterprise Linux v6.2 以降の x86-64 システムにインストールする必要があります。

ハイブリッド・ゲートウェイは、システムが Red Hat Enterprise Linux で実行されている場合、Tivoli Enterprise Portal Server と同じシステムにインストールすることも、Tivoli Enterprise Portal Server とは別個のシステムにインストールすることもできます。ただし、ハイブリッド・ゲートウェイは、Cloud APM サーバーと同じシステムにインストールすることができません。

Tivoli Monitoring ドメインには、1 つのハブ Tivoli Enterprise Monitoring Server があります。Tivoli Monitoring 環境が複数のドメインで構成されている場合、複数のドメインにハイブリッド・ゲートウェイをインストールできます。

ハイブリッド・ゲートウェイに関連するシステム要件については、ハイブリッド・ゲートウェイの Software Product Compatibility Report で、「**Hardware**」タブをクリックしてください。

Tivoli Enterprise Portal Server のハイブリッド・ゲートウェイ用のセットアップ
ポータル・サーバーの負荷が高い Tivoli Monitoring 環境では、ハイブリッド・ゲートウェイからの要求を処理するために別個の専用ポータル・サーバーをインストールする必要があります。別個のポータル・サーバーをセットアップする 場合、次のとおりです。

- ポータル・サーバーが Red Hat Enterprise Linux を実行している場合、ポータル・サーバーとハイブリッド・ゲートウェイに同じホストを使用できます。
- 別個のポータル・サーバーに、Cloud APM コンソールにデータが表示される元のエージェントに対するアプリケーション・サポートがあることを確認してください。
- Tivoli Enterprise Portal クライアントが、カスタム・ワークスペースの作成、シチュエーションの作成、管理対象システム・グループの作成などの管理用タスクを実行するために、別個のポータル・サーバーに接続しないことを確認してください。

Tivoli Enterprise Portal Server は V6.3 フィックスパック 6 以上である必要があります。使用しているポータル・サーバーのバージョンがこれより古い場合は、統合した Tivoli Monitoring エージェントを Cloud APM コンソールでアプリケーションに追加できない可能性があります。

IBM Tivoli Monitoring ダッシュボード・データ・プロバイダーが Tivoli Enterprise Portal Server で有効である必要があります。詳しくは、IBM Knowledge Center にある IBM Tivoli Monitoring トピック・コレクションの『ダッシュボード・データ・プロバイダーが使用可能であることの確認』を参照してください。

ハイブリッド・ゲートウェイで開いている必要がある TCP ポート
次の TCP ポートがハイブリッド・ゲートウェイで開かれている必要があります。各ポートについて、一方の側が要求を送信し、もう一方の側が応答を返します。接続の開始側が示されています。

- ハイブリッド・ゲートウェイ は、ポート 443 で Cloud APM サーバーとの単一方向接続を開始して HTTPS 要求を送信します。

  ハイブリッド・ゲートウェイがパススルー転送プロキシーを使用して Cloud APM サーバーに接続する場合、Cloud APM サーバーとの間で開始する単一方向接続に対してポート 443 の代わりにプロキシー・ポートを使用するようにハイブリッド・ゲートウェイを構成します。説明については、957 ページの『フォワード・プロキシーを使用した Cloud APM サーバーとの通信』を参照してください。
- ポータル・サーバーとの通信に HTTP を使用する場合は、ポート 15200 を開きます。HTTPS を使用する場合は、ポート 15201 を開きます。ハイブリッド・ゲートウェイは、ポート 15200 または 15201 で、ポータル・サーバーとの単一方向接続を開始します。カスタム・ポートを使用するには、「**Portal Server ポート**」設定の値を更新します。 詳しくは、964 ページの『ハイブリッド・ゲートウェイ Manager』を参照してください。

  別の方法として、ハイブリッド・ゲートウェイがポータル・サーバーに接続するためにパススルー転送プロキシーを使用する場合は、ポータル・サーバーとの間で開始する単一方向接続のために、プロキシー・ポートを代わりに使用するようにハイブリッド・ゲートウェイを構成します。「**パススルー・プロキシーのポート**」設定の値を設定します。 詳しくは、964 ページの『ハイブリッド・ゲートウェイ Manager』を参照してください。
- ハイブリッド・ゲートウェイが Tivoli Enterprise Monitoring Server からのインバウンド EIF イベントを listen するようにするには、ポート 9998 を開きます。モニター・サーバーは、ポート 9998 で、ハイブリッド・ゲートウェイとの単一方向接続を開始します。このポートが開いていない場合は、インストール・ユーティリティーで 警告が表示されます。

ハイブリッド・ゲートウェイのインストール・スクリプトを実行するために必要な root 特権
ハイブリッド・ゲートウェイのインストール・スクリプトは、root 権限を使用して実行する必要があります。サポートされるオペレーティング・システムの完全なリストについては、System requirements (APM Developer Center) を参照してください。

ハイブリッド・ゲートウェイのインストール
Cloud APM コンソールに IBM Tivoli Monitoring ドメインの管理対象システムを表示するには、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをダウンロードしてインストールします。

始める前に
『ハイブリッド・ゲートウェイをインストールするための準備』で必須の準備タスクを参照し、完了してください。

手順

Tivoli Monitoring ドメインにハイブリッド・ゲートウェイをインストールするには、次の手順を実行します。

1. ハイブリッド・ゲートウェイ・パッケージをダウンロードします。

   `APM_Hybrid_Gateway_Install.tar` ファイルにはハイブリッド・ゲートウェイとインストール・スクリプトが含まれています。

   a) アカウントにサインインして、IBM Marketplace の製品およびサービスにアクセスします。

b) **IBM Cloud APM** で、「**他の操作**」をクリックします。

c)「**追加パッケージの表示**」をクリックします。

d)「**ハイブリッド・ゲートウェイ**」を選択します。必要に応じて、この項目が見つかるまでスクロールダウンしてください。

e)「**ダウンロード**」をクリックします。

2. 必要に応じて、ハイブリッド・ゲートウェイを実行するシステムにファイルを転送します。
3. 次のコマンドを入力して、ファイルを解凍します。

```
tar -xf APM_Hybrid_Gateway_Install.tar
```

このアーカイブ・ファイルには、ハイブリッド・ゲートウェイのデプロイに使用するスクリプトが入っています。 インストール・スクリプトはディレクトリーに解凍され、ハイブリッド・ゲートウェイのファイルはサブディレクトリーに解凍されます。

4. ハイブリッド・ゲートウェイのディレクトリーに移動し、以下のように root ユーザー特権を使用してインストール・スクリプトを実行します。

```
cd APM_Hybrid_Gateway_Install_version
./install.sh
```

ここで、*version* は現行バージョン (8.1.4.0 など) です。

ご使用の環境の前提条件スキャンが開始されます。このスキャンが完了するまでにはしばらく時間がかかります。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ディスク・スペース不足など、前提条件が原因でインストールが停止します。失敗に対処してから、インストールを再度開始する必要があります。前提条件チェックをオフにすることもできます。『前提条件スキャナーのバイパス』を参照してください。

5. システムが前提条件スキャンを通過した後、ご使用条件への同意を促すプロンプトが表示されます。このプロンプトに対して 1 (はい) を選択して応答します。

次に、Cloud APM コンソールにログインしてハイブリッド・ゲートウェイを構成してから続行するようにメッセージで指示されます。ホスト名から派生されるデフォルト・プロファイル名も表示されます。

6. Enter キーを押してデフォルト名を受け入れるか、プロファイル名を入力します。

この Tivoli Monitoring ドメイン用のプロファイルが既に作成済みの場合は、ハイブリッド・ゲートウェイ・マネージャーで指定したのと同じ名前を使用します。まだプロファイルが作成されていない場合は、デフォルト名を受け入れるか、新規名を指定できます。ただし、この名前は、後でプロファイルを作成する際に使用する必要があるため、必ず記録しておいてください。 (959 ページの『Cloud APM コンソールを使用したハイブリッド・ゲートウェイの構成』 を参照。)

Enter キーを押すと、ハイブリッド・ゲートウェイのインストールが続行します。

タスクの結果

ハイブリッド・ゲートウェイが /opt/ibm/hybridgateway ディレクトリーにインストールされ、自動的に始動します。インストール・ログ・ファイルは、/opt/ibm/hybridgateway/logs/install-hybridgateway-*timestamp*.log です。 ハイブリッド・ゲートウェイのログ・ファイルは、/opt/ibm/wlp/usr/servers/hybridgateway/logs ディレクトリー内にあります。 Tivoli Enterprise Portal Server への接続が構成されるまでは接続の失敗がログに記録されますが、問題はありません。

次のタスク

- Cloud APM サーバーとの通信にフォワード・プロキシーを使用するようにハイブリッド・ゲートウェイを構成できます。説明については、957 ページの『フォワード・プロキシーを使用した Cloud APM サーバーとの通信』を参照してください。
- ハイブリッド・ゲートウェイの状況を確認するには、*install_dir*/hybridgateway/bin/hybridgateway.sh status コマンドを使用します。オプションについて詳しくは、963 ページの『ハイブリッド・ゲートウェイの管理』 を参照してください。

- ハイブリッド・ゲートウェイ用の管理対象システム・グループをまだ作成していない場合は、957 ページの『管理対象システム・グループの作成』の説明に従ってください。
- まだ Tivoli Monitoring ドメイン用のハイブリッド・ゲートウェイ・プロファイルが作成済みでない場合は、959 ページの『Cloud APM コンソールを使用したハイブリッド・ゲートウェイの構成』の説明に従ってください。
- ご使用の Tivoli Monitoring 環境に複数のハブ・ドメインがある場合は、他のドメインにハイブリッド・ゲートウェイをインストールできます。この手順のステップを繰り返して、ハイブリッド・ゲートウェイを別の Tivoli Monitoring ドメインにインストールしてください。

フォワード・プロキシーを使用した Cloud APM サーバーとの通信
Cloud APM サーバーとの通信にフォワード・プロキシーを使用するように IBM Cloud Application Performance Management ハイブリッド・ゲートウェイを構成できます。

手順

1. ハイブリッド・ゲートウェイをインストールしたホストで、以下のように `/opt/ibm/wlp/usr/servers/hybridgateway/bootstrap.properties` ファイルを編集します。
   - ハイブリッド・ゲートウェイが HTTP を使用して Cloud APM サーバーと通信している場合、以下の行を追加します。

     ```
     http.proxyHost=proxy_host
     http.proxyPort=proxy_port
     ```

   - ハイブリッド・ゲートウェイが HTTPS を使用して Cloud APM サーバーと通信している場合、以下の行を追加します。

     ```
     https.proxyHost=proxy_host
     https.proxyPort=proxy_port
     ```

   ここで、*proxy_host* はハイブリッド・ゲートウェイのホストからアクセス可能なプロキシーのホスト名または IP アドレスであり、*proxy_port* はプロキシーのポートです。
2. ハイブリッド・ゲートウェイを再始動します。

管理対象システム・グループの作成
Cloud APM コンソールに表示する管理対象システムの管理対象システム・グループを作成するには、Tivoli Enterprise Portal クライアントのオブジェクト・グループ・エディターを使用します。

始める前に

- 管理対象システム・グループに含めることができる IBM Tivoli Monitoring エージェントおよび OMEGAMON エージェントのタイプは、サポートされるエージェントのいずれかでなければなりません。例えば、Tivoli Monitoring では、サポートされるエージェントは、Monitoring Agent for Oracle Database や Monitoring Agent for Linux OS です。

  サポートされる Tivoli Monitoring エージェントのリストについては、Hybrid Gateway supported agents (APM Developer Center)を参照してください。Cloud APM コンソールでモニターできる OMEGAMON エージェントのリストについては、IBM Knowledge Center の IBM OMEGAMON for Application Performance Management トピック・コレクション内のトピック『はじめに』を参照してください。
- Tivoli Monitoring および OMEGAMON エージェントは、同じ IBM Tivoli Monitoring インフラストラクチャーに接続する必要があります。ご使用の環境に複数の Tivoli Monitoring ドメインがある場合は、ハイブリッド・ゲートウェイがインストールされているハブ Tivoli Enterprise Monitoring Server ごとに管理対象システム・グループを作成します。
- デフォルトでは、Application Performance ダッシュボードに Tivoli Monitoring ドメインから表示する管理対象システム・グループには、管理対象システムを 200 件まで追加することができます。いくつかの計画ステップを実行して、この制限を最大で 1500 システムまで増加させることができます。詳しくは、962 ページの『管理対象システムが多い場合の計画』を参照してください。複数のハブがある環境で Tivoli Monitoring 用の 複数のハイブリッド・ゲートウェイがある場合、各ドメインの管理対象システム・

グループは、Cloud APM でサポートされる最大以内でなければなりません。詳しくは、『アーキテクチャーの概要』を参照してください。

- Cloud APM の 2017 年 3 月の リリースで、サブノードを処理するデフォルトが変更されました。以前のリリースでは、WebSphere Applications エージェントなど、サブノードを持つエージェントがある場合管理ノードを管理対象システム・グループに割り当てる必要がありましたが、それにより、すべてのサブノードが自動的に組み込まれていました。管理対象システム・グループに割り当てた管理ノードが 1 つであっても、すべてのサブノードが管理対象システム数に算入され、最大数の制限を受けていました。

  Cloud APM の 2017 年 3 月の 以降のリリースでは、Cloud APM コンソールでメトリック・データを表示するサブノードを明示的に管理対象システム・グループに割り当てなければなりません。サブノードのいずれかが管理対象システム・グループに明示的に割り当てられている場合、管理エージェントは自動的にディスカバーされます。サブノードに基づいているモニター・アプリケーションでは、Cloud APM は、Cloud APM ダッシュボード・ナビゲーターで表示されるモニター・リソースを明確に識別するために必要な情報を管理エージェントに照会する必要が生じることがあります。そのため、現行バージョンのディスカバリー・モードでは、管理エージェントが自動的に組み込まれ、ハイブリッド・ゲートウェイが使用するように構成された管理対象システム・グループに割り当てられた関連サブノードが少なくとも 1 つ存在しています。現在のディスカバリー・モードは、Cloud APM コンソールで表示できるサブノード・リソースに対する詳細な制御をサポートしており、Cloud APM アプリケーションの構成とうまく整合します (特に、大量のサブノード・リソース・インスタンス・セットが関わるアプリケーションの場合)。OMEGAMON エージェントを Cloud APM と統合する場合は、常に現在のデフォルト・ディスカバリー・モードを使用してください。

  MSN_DISCOVERY_MODE という外部プロパティーに該当値を割り当てることで、ハイブリッド・ゲートウェイで使用するディスカバリー・モード・バージョンを指定できます。このプロパティーは、ハイブリッド・ゲートウェイによって初期化時に処理されます。ハイブリッド・ゲートウェイで使用されるディスカバリー・モードを制御するには、ハイブリッド・ゲートウェイがインストールされているシステム 上の以下のプロパティー・ファイルで MSN_DISCOVERY_MODE プロパティーを追加 (または現行値を変更) してから、ハイブリッド・ゲートウェイを再始動します。

  ```
  HG_install_dir/wlp/usr/servers/hybridgateway/bootstrap.properties
  ```

  MSN_DISCOVERY_MODE プロパティーに指定可能な値は、以下のとおりです。

  - MSN_DISCOVERY_MODE=1 の場合、強制的にハイブリッド・ゲートウェイは元のエージェント・ディスカバリー・モードを使用するようになります。このモードでは、Tivoli Monitoring 管理対象システム・グループに割り当てられているすべての管理エージェントのすべてのサブノードが 自動的にディスカバーされます。
  - MSN_DISCOVERY_MODE=2 の場合、強制的にハイブリッド・ゲートウェイは新しいデフォルト・エージェント・ディスカバリー・モードを使用するようになります。このモードでは、管理対象システム・グループに明示的に割り当てられているサブノードのみが、ハイブリッド・ゲートウェイから照会されます。関連付けられている単一または複数の管理エージェントは、自動的にディスカバーされます。

- **tacmd createsystemlist** コマンドおよび **tacmd editsystemlist** IBM Tivoli Monitoring コマンドで管理対象システム・グループを作成する場合は、それらのコマンドの実行方法について、*IBM Tivoli Monitoring* コマンド・リファレンス (https://www.ibm.com/support/knowledgecenter/SSTFXA_6.3.0/com.ibm.itm.doc_6.3/cmdref/itm_cmdref.htm) を参照してください。

手順

Tivoli Enterprise Portal クライアントに管理対象システム・グループを作成するには、以下のステップを実行します。

1. すべての管理対象システム・タイプに対する全アクセス権限を持つ (「**許可されたアプリケーション**」が「**すべてのアプリケーション**」に設定されている) ユーザー ID およびパスワードを使用して Tivoli Enterprise Portal クライアントを始動します。
2. 「**オブジェクト・グループ・エディター**」をクリックします。
3. 「**管理対象システム**」オブジェクトを展開し、「**すべての管理対象システム**」を選択して、 管理対象システム・グループに複数のエージェント・タイプ (Windows OS や Oracle など) を結合します。管

理対象システム・グループに含めるモニター・エージェントのタイプを 1 つだけにする場合 (Linux OS や WebSphere Applications など) は、そのエージェントのタイプを選択します。

4. 「**新規グループの作成**」をクリックし、 管理対象システム・グループの名前を入力します。
   名前に使用できるのは英字、数字、および下線 (_) のみです。スペース、句読点、その他の特殊文字は使用できません。
   「**OK**」をクリックすると、 管理対象システムのフォルダーに 新しい管理対象システム・グループが表示されます。
5. 「**使用可能な管理対象システム**」リストから管理対象システムを選択し、 を選択して「**割り当て済み**」リストに移動します。
   Ctrl キーを押しながら個々の管理対象システムをクリックすることにより、複数の管理対象システムを選択できます。管理対象システムを選択した後、Shift キーを押しながら管理対象システムをクリックして選択すると、その管理対象システムと最初の管理対象システムの間にあるすべての管理対象システムを選択できます。
6. 管理対象システムをグループに 追加した後、「**OK**」をクリックして変更内容を保存し、オブジェクト・グループ・エディターを閉じます。

次のタスク

- 管理対象システム・グループを作成してハイブリッド・ゲートウェイをインストールした後は、Cloud APM コンソールでハイブリッド・ゲートウェイを構成する必要があります。
- 構成には、作成した管理対象システム・グループ名、すべてのエージェント・タイプにアクセス可能な Tivoli Enterprise Portal のユーザー ID、および Tivoli Enterprise Portal Server のホスト名とポートを指定します。
- インストール手順については、955 ページの『ハイブリッド・ゲートウェイのインストール』を参照してください。
- 構成の手順については、959 ページの『Cloud APM コンソールを使用したハイブリッド・ゲートウェイの構成』を参照してください。

Cloud APM コンソールを使用したハイブリッド・ゲートウェイの構成

Cloud APM コンソールの「**ハイブリッド・ゲートウェイ・マネージャー**」ページを使用すると、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイが Tivoli Enterprise Portal Server に接続するように構成したり、および管理対象システム・グループを指定したりすることができます。環境内のハブ Tivoli Enterprise Monitoring Server ごとにハイブリッド・ゲートウェイ・プロファイルを作成できます。

手順

Cloud APM コンソールでハイブリッド・ゲートウェイを構成するには、次の手順を実行します。

1. まだ Cloud APM コンソールにログインしていない場合は、ここでログインします。
   (977 ページの『Cloud APM コンソールの始動』 を参照。)
2. 「**システム構成**」 > 「**ハイブリッド・ゲートウェイ・マネージャー**」をクリックします。
   Tivoli Monitoring ドメイン用に構成されたハイブリッド・ゲートウェイの表を含むページが表示されます。名前がブランクのプロファイルが表示された場合、そのプロファイルは、2017 年 8 月リリースよりも前にインストールされたハイブリッド・ゲートウェイ用です。詳しくは、964 ページの『プロファイル名』を参照してください。
3. 「**追加**」をクリックして「**ハイブリッド・ゲートウェイの追加**」ウィンドウを開き、「**プロファイル名**」フィールドに新しい名前を入力し、「**追加**」をクリックします。
   ハイブリッド・ゲートウェイを Tivoli Monitoring ドメインに既にインストールしている場合、ハイブリッド・ゲートウェイのインストール時に指定した、またはそのまま使用した同じ名前を使用する必要があります。インストール時に入力したプロファイル名と、ここで入力する名前は、完全に一致する必要があります。
   「**ハイブリッド・ゲートウェイの編集**」ウィンドウが開きます。

4. 「**管理対象システム・グループ名**」フィールドに、ハイブリッド・ゲートウェイの管理対象システム・グループの名前を入力します。

   これは、957 ページの『管理対象システム・グループの作成』で使用した名前です。
5. Tivoli Enterprise Portal Server のアドレス、ポート、および Web 通信プロトコルを以下のように指定します。

オプション	説明
Portal Server ホスト名	Portal Server のホスト IP アドレスまたは完全修飾ホスト名またはドメイン名を入力します。
Portal Server ポート	Portal Server が Web 通信に使用するポート番号を入力します。デフォルトのポートは、HTTP の場合は 15200、HTTPS の場合は 15201 です。値 0 を使用すると、ポートが、HTTP の場合は 15200、HTTPS の場合は 15201 のデフォルトに設定されます。
Portal Server プロトコル	Portal Server に接続するための **HTTP** インターネット・プロトコルまたはセキュア **HTTPS** インターネット・プロトコルを選択します。

6. 「**Portal Server ユーザー名**」および「**Portal Server ユーザー・パスワード**」の各フィールドに、Tivoli Enterprise Portal クライアントを始動するためのログオン・ユーザー名および対応するパスワードを入力します。

   このユーザー ID には、`sysadmin` ID などのすべてのモニター・エージェント・タイプに対するアクセス権限が必要です (「**許可されたアプリケーション**」を「**すべてのアプリケーション**」に設定します)。詳しくは、Tivoli Monitoring IBM Knowledge Center のユーザー管理を参照してください。
7. プロキシー・サーバー経由でポータル・サーバーにアクセスする場合は、アドレス、ポート、および Web プロトコルを以下のように指定します。

オプション	説明
パススルー・プロキシーのホスト名	プロキシー・ホスト・システムの IP アドレスまたは完全修飾名を入力します。
パススルー・プロキシーのポート	プロキシー・ホスト・システムのポート番号を入力します。
パススルー・プロキシーのプロトコル	プロキシー経由の通信に使用するプロトコル (HTTP または HTTPS) を選択します。

タスクの結果

「**保存**」をクリックすると、ハイブリッド・ゲートウェイ・サービスとの接続が確立され、Tivoli Monitoring ドメインにある管理対象システムがディスカバーされます。管理対象システム・グループが 5 分ごとにポーリングされ、 リソース・モニター・データが収集されます。

次のタスク

- Tivoli Monitoring が Cloud APM と対話するように構成する必要があります。 説明については、961 ページの『Cloud APM と統合するための Tivoli Monitoring の構成』を参照してください。
- これらのステップを繰り返すことで、Cloud APM から管理対象システムをモニターする Tivoli Monitoring ドメインごとにプロファイルを追加できます。
- 「**ハイブリッド・ゲートウェイ・マネージャー**」ツールを使用して、既存のプロファイルを管理できます。

  ハイブリッド・ゲートウェイを選択し、「**編集**」をクリックすると、「**ハイブリッド・ゲートウェイの編集**」ウィンドウが開きます。

  不要になったハイブリッド・ゲートウェイを選択し、「**削除**」をクリックします。ハイブリッド・ゲートウェイを削除することを確定すると、そのプロファイルは完全に削除されます。

  列見出しをクリックすると、その列を基準に表がソートされます。Ctrl キーを押しながら別の列をクリックして、2 次ソートを追加します。

  フィルター・テキスト・ボックス の内側をクリックして、フィルタリングする 値の先頭部分を入力します。入力するにしたがい、条件に一致しない行は除外されていきます。フィルターをク

リアするには、フィルター・ボックス の × をクリックするか、Backspace キーを押します。

Cloud APM と統合するための Tivoli Monitoring の構成

IBM Tivoli Monitoring ドメインを Cloud APM と統合するには、Tivoli Enterprise Portal Server の構成やハブ Tivoli Enterprise Monitoring Server の構成などのタスクを実行する必要があります。

手順

ハイブリッド・ゲートウェイがインストールされている Tivoli Monitoring ドメインごとに、以下のステップを実行します。

1. ダッシュボード・データ・プロバイダーを有効にするように Tivoli Enterprise Portal Server を構成します。

   このデータ・プロバイダーは、Cloud APM をハイブリッド・ゲートウェイと統合するために必要です。手順については、次のトピックを参照してください。

   - Windows Windows: Installing the portal server (ステップ 16c)。
   - Linux AIX Configuring the portal server on Linux or AIX: command-line procedure (ステップ 14)。

   ホット・スタンバイ機能を使用している場合、ドメイン・オーバーライドを指定する必要があります。ハイブリッド・ゲートウェイは、ポータル・サーバーが接続されているハブに関係なく、ドメイン・ネームを使用して両方のハブ・モニター・サーバーからデータを収集します。

2. Cloud APM コンソールで Tivoli Monitoring エージェントからのシチュエーション・イベントを表示する場合、以下のいずれかのシナリオのためにハブ・モニター・サーバーを構成します。

   - イベントをハイブリッド・ゲートウェイにのみ送信する。
   - イベントをハイブリッド・ゲートウェイおよび追加の EIF 受信側 (Netcool/OMNIbus サーバーなど) に送信する。

   該当するシナリオに応じて、以下のステップのいずれかを実行します。

   a) イベントをハイブリッド・ゲートウェイのみに送信するようにハブ Tivoli Enterprise Monitoring Server を構成するには、『イベントを転送するためのハブ・モニター・サーバーの構成』トピックのステップを実行します。

   ServerPort パラメーターにポート番号 9998 を指定します。
   Cloud APM は、Tivoli Monitoring イベントを Netcool/OMNIbus に転送しません。Tivoli Monitoring イベントを Cloud APM および Netcool/OMNIbus で表示する場合、イベントを両方のシステムに送信するように Tivoli Monitoring を構成する必要があります。

   b) イベントをハイブリッド・ゲートウェイおよび別の EIF 受信側 (Netcool/OMNIbus サーバーなど) に送信するようにハブ Tivoli Enterprise Monitoring Server を構成するには、『イベントを転送するためのハブ・モニター・サーバーの構成』トピックのステップを使用して、デフォルト EIF 受信側を構成します。

   このトピックでは、**tacmd createEventDest** コマンドを使用した追加の EIF 宛先の作成に関する情報も示されています。ハイブリッド・ゲートウェイの宛先の EIF ポート番号としてポート 9998 を指定してください。

   c) ハイブリッド・ゲートウェイ管理対象システム・グループ内のエージェントに対して存在するすべてのシチュエーションを構成して、シチュエーション・イベントがハイブリッド・ゲートウェイの EIF 宛先に送信されるようにします。説明については、『Netcool/OMNIbus にイベントを転送するシチュエーションの指定』トピックを参照してください。

次のタスク

以下のように、Application Performance ダッシュボードを参照して、Tivoli Monitoring ドメインにある管理対象システムがハイブリッド・ゲートウェイをパススルーすることを確認します。

1. 「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックして、「**すべてのマイ・アプリケーション**」ダッシュボードを開きます。

2. 「マイ・コンポーネント」の要約ボックスで、「**コンポーネント**」をクリックし、(WebSphere Applications エージェント以外の) すべてのコンポーネントの管理対象システムの状況要約ダッシュボードを開きます。「マイ・コンポーネント」アプリケーションがない場合は、1098 ページの『アプリケーションの管理』で説明する手順に従ってアプリケーションを追加します。
3. Tivoli Monitoring ドメインの管理対象システムを探します (状況要約グループ・ウィジェットのタイトルにある **ITM** (IBM Tivoli Monitoring) ドメイン・アイコンによって示されています)。いずれかの管理対象システムがない場合、Cloud Application Performance Management フォーラム に移動し、「ハイブリッド・ゲートウェイ」を検索します。

Tivoli Monitoring ドメインの管理対象システムでアプリケーションを 作成したり、Cloud APM ドメインの管理対象システムを含めたりすることができます。詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

管理対象システムが多い場合の計画

IBM Tivoli Monitoring ドメインから表示できる管理対象システムの最大数は 1500 です。ハイブリッド・ゲートウェイ・プロファイルに対して作成した管理対象システム・グループに、サブノードのあるエージェントを含める場合、エージェントとすべてのサブノードが総計されて制限が適用されます。デフォルトではこの制限値は管理対象システム 200 件ですが、計画手順を実行することでその制限を拡大できます。すべての Tivoli Monitoring ドメインの制限値は、Cloud APM でサポートされる最大値以下にする必要があります。詳しくは、45 ページの『アーキテクチャーの概要』を参照してください。

- Tivoli Enterprise Portal Server **KFW_REPORT_NODE_LIMIT** 環境変数の値を、ハイブリッド・ゲートウェイの管理対象システムの数以上の数値に設定します。デフォルト値は 200 です。説明については、Tivoli Enterprise Portal Server 構成設定を参照してください。
  管理対象システムの数がこの設定値を超えると、Tivoli Enterprise Portal Server の `KfwServices` メッセージ・ログに以下の例のようなメッセージが出力されます:
  `56C6246F.0000-10:ctreportmanager.cpp,2864,"CTReport::Manager::executeDefinitionDual") Query is targeting 1497 nodes which exceeds the current limit of 200 nodes.`
- 多数のシステムを表示する場合、エージェントの種類、ハイブリッド・ゲートウェイと管理対象システムの間のネットワーク待ち時間、各エージェントがモニターする環境の規模 (収集して送信するデータの量) によってはパフォーマンスが低下する場合があります。この影響を避けるには、必要なデータを提供するエージェントのみを選択し、モニター対象システムとハイブリッド・ゲートウェイ・ホストの間で高速なネットワーク接続を確保してください。
- ネットワーク待ち時間が長くなると、一定数のエージェントからデータを収集するための時間も長くなります。ハイブリッド・ゲートウェイは、5 分ごとに各エージェントからデータを収集しようとします。すべてのエージェントからデータを収集するための時間が 5 分を超える場合、 ハイブリッド・ゲートウェイはデータ・サンプルを収集しきれないため、一部の管理対象システムのメトリックが Application Performance ダッシュボード ページで利用できなくなります。
- ネットワークが特に低速な場合は、対策としてハイブリッド・ゲートウェイがデータ・サンプルの収集に使用するスレッドの数を増加させてみてください。スレッド数は、ハイブリッド・ゲートウェイの `bootstrap.properties` ファイルの **MAX_COLLECTOR_THREADS** パラメーターで制御します。デフォルト値は 50 です。

ハイブリッド・ゲートウェイのアンインストール

Cloud APM コンソールに IBM Tivoli Monitoring 管理対象システムを表示する必要がなくなった場合は、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをアンインストールします。

手順

1. ハイブリッド・ゲートウェイの *install_dir*/hybridgateway/bin ディレクトリー (/opt/ibm/hybridgateway/bin など) で、次のコマンドを実行します。

   ```
   ./hybridgateway.sh uninstall
   ```

   ハイブリッド・ゲートウェイが削除され、メッセージにより正常にアンインストールされたことが確認できます。ハイブリッド・エージェントを含むアプリケーションが Cloud APM コンソールに存在する場

合は、モニター・インフラストラクチャーによって削除が処理されるまで、引き続きハイブリッド・エージェントが表示されます。

2. Cloud APM コンソールで、「**システム構成**」 > 「**ハイブリッド・ゲートウェイ・マネージャー**」をクリックします。
3. 不要になったハイブリッド・ゲートウェイ・プロファイルを選択し、「**削除**」をクリックします。
   ハイブリッド・ゲートウェイを削除することを確定すると、そのプロファイルは完全に削除されます。

次のタスク

- Cloud APM コンソール内のアプリケーションからハイブリッド ・エージェントの管理対象システムをすべて削除するには、1098 ページの『アプリケーションの管理』の説明に従ってアプリケーションを編集してください。
- ソフトウェアが正常に削除されずに、以下の例に示すようなエラー・メッセージが出された場合は、 ログ・ファイルを参照して考えられる原因を調べてください。

```
error: Failed dependencies:
ibm-java-x86_64-jre is needed by (installed) smai-kafka-00.08.00.00-1.el6.x86_64
Uninstallation failed. The uninstaller was unable to remove some of the components,
please inspect the
 log file ("/tmp/hybridgateway/logs/uninstall-hybridgateway-20150228080551.log")
for more information.
```

  例に示したエラーが発生した原因は、 システムの外部でインストールされたパッケージが ibm-java-x86_64-jre を必要としているためです。他のパッケージが機能しなくなる可能性があるため、 インストーラーは JRE を削除しません。この事態を回避するには、ハイブリッド・ゲートウェイをアンインストールする前に、ibm-java-x86-64-jre に依存する製品をアンインストールしてください。

ハイブリッド・ゲートウェイの管理

IBM Cloud Application Performance Management ハイブリッド・ゲートウェイ・サービスの開始または停止、状況の確認、ハイブリッド・ゲートウェイのアンインストール、IBM サポートから依頼されたログ・ファイルの収集には使用可能なコマンドを使用します。

このタスクについて

ここに示すステップでは、ハイブリッド・ゲートウェイのインストール・ディレクトリーが /opt/ibm/ であると想定しています。ハイブリッド・ゲートウェイをインストールしたシステムで、 コマンド・プロンプトから以下のいずれかのステップを実行します。

手順

- ハイブリッド・ゲートウェイ・サービスを開始するには、**/opt/ibm/hybridgateway/bin/hybridgateway.sh start** と入力します。
- ハイブリッド・ゲートウェイ・サービスを停止するには、**/opt/ibm/hybridgateway/bin/hybridgateway.sh stop** と入力します。
- ハイブリッド・ゲートウェイ・サービスの状況を確認するには、**/opt/ibm/hybridgateway/bin/hybridgateway.sh status** と入力します。
- ハイブリッド・ゲートウェイをアンインストールするには、**/opt/ibm/hybridgateway/bin/hybridgateway.sh uninstall** と入力します。

  962 ページの『ハイブリッド・ゲートウェイのアンインストール』も参照してください。
- ハイブリッド・ゲートウェイのログ・ファイルを確認するには、/opt/ibm/wlp/usr/servers/hybridgateway/logs に移動します。
- IBM サポートに提供するハイブリッド・ゲートウェイのログ・ファイルを収集するには、**/opt/ibm/hybridgateway/collectLogs.sh** と入力します。

  ログ・ファイルが収集され、圧縮されたログ・ファイルの場所がメッセージに表示されます。このログ・ファイルを IBM サポートにご提供ください。

ハイブリッド・ゲートウェイ Manager
IBM Tivoli Monitoring ドメインからのモニター・データを Cloud APM コンソール に表示するために、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイを構成します。環境内のハブ Tivoli Enterprise Monitoring Server ごとにハイブリッド・ゲートウェイ・プロファイルを作成できます。

「**システム構成**」>「**ハイブリッド・ゲートウェイ・マネージャー**」をクリックすると、定義済みの ハイブリッド・ゲートウェイのリストがページに表示されます。

このページには、Tivoli Monitoring ドメイン用に構成されたすべてのハイブリッド・ゲートウェイの表があり、ハイブリッド・ゲートウェイ・プロファイルを管理するためのツールがあります。

- 「**追加**」をクリックすると、新規プロファイルに名前を付けるための「**ハイブリッド・ゲートウェイの追加**」ウィンドウが開きます。名前を入力して「**追加**」をクリックすると、「**ハイブリッド・ゲートウェイの編集**」ウィンドウが開きます。
- ハイブリッド・ゲートウェイを選択し、「**編集**」をクリックすると、「**ハイブリッド・ゲートウェイの編集**」ウィンドウが開きます。
- 不要になったハイブリッド・ゲートウェイを選択し、「**削除**」をクリックします。ハイブリッド・ゲートウェイを削除することを確定すると、そのプロファイルは完全に削除されます。
- 列見出しをクリックすると、その列を基準に表がソートされます。Ctrl キーを押しながら別の列をクリックして、2 次ソートを追加します。
- フィルター・テキスト・ボックス の内側をクリックして、フィルタリングする 値の先頭部分を入力します。入力するにしたがい、条件に一致しない行は除外されていきます。フィルターをクリアするには、フィルター・ボックス の × をクリックするか、Backspace キーを押します。

ハイブリッド・ゲートウェイを構成するためにデータを設定する必要がある必須フィールドには、「**ハイブリッド・ゲートウェイの編集**」ウィンドウでアスタリスク (*) が付けられています。

プロファイル名
ハイブリッド・ゲートウェイ・プロファイルに付与する名前。最大 128 個の文字、数字、およびアンダースコアー (_) を使用できます。

このプロファイル名は、ハイブリッド・ゲートウェイのインストール時に要求されます。ハイブリッド・ゲートウェイを Tivoli Monitoring ドメインに既にインストールしている場合、ハイブリッド・ゲートウェイのインストール時に指定した名前、またはそのまま使用した名前を使用します。

古いバージョンのハイブリッド・ゲートウェイでは、構成データにアクセスするために名前付きプロファイルは使用されません。Cloud APM 2017 年 8 月リリースよりも前のハイブリッド・ゲートウェイをインストールした場合、特殊な、名前なし (ブランク) のプロファイル名が使用されます。1 つ古いバージョンのハイブリッド・ゲートウェイのみが、Cloud APM サーバーに接続できます。 前のバージョンのハイブリッド・ゲートウェイを構成した場合、名前なしのプロファイルで構成済みの値が示されます。前のバージョンのハイブリッド・ゲートウェイを構成していない場合、名前なしプロファイルでデフォルト値が示されます。名前なしプロファイルは、必要に応じて、保持することも、削除して後で再追加することもできます。この名前なしプロファイルは、2017 年 3 月 (またはそれ以前の) バージョンのハイブリッド・ゲートウェイに対してのみ使用できます。

管理対象システム・グループ名
Cloud APM コンソールでサポートされるモニター・エージェントを表示するために作成された Tivoli Enterprise Portal Server 管理対象システム・グループ。ご使用の Cloud APM オファリングでサポートされていないモニター・エージェント・タイプは、管理対象システム・グループに含まれていても、コンソールに表示されません。
ハイブリッド構成を使用可能にするために管理対象システム・グループを作成するときの指針と制限については、 を参照してください。

Portal Server ホスト名
Tivoli Enterprise Portal Server のホスト IP アドレス、または完全修飾ドメイン名。

Portal Server ポート
通信用に Tivoli Enterprise Portal Server が使用するポート番号。デフォルトのポートは、HTTP の場合は 15200、HTTPS の場合は 15201 です。値 0 を使用すると、ポートが、HTTP の場合は 15200、HTTPS の場合は 15201 のデフォルトに設定されます。

Portal Server プロトコル
Tivoli Enterprise Portal Server に接続するために、HTTP インターネット・プロトコルとセキュア HTTPS プロトコルのどちらを使用するかを決定します。デフォルト: `http`。

Portal Server ユーザー名
Tivoli Enterprise Portal クライアントを始動するためのユーザー名。このユーザー ID には、すべてのモニター・エージェント・タイプに対するアクセス権限が必要です (「**許可されたアプリケーション**」を「**すべてのアプリケーション**」に設定します)。 詳しくは、 Tivoli Monitoring Knowledge Center の ユーザー管理を参照してください。

Portal Server ユーザー・パスワード
Tivoli Enterprise Portal のログオン・ユーザー名に関連付けられたパスワード。

パススルー・プロキシーのホスト名
Tivoli Enterprise Portal Server がパススルー・プロキシー・サーバー経由で通信する場合に使用します。プロキシー・ホスト・システムの IP アドレスまたは完全修飾名を入力します。

パススルー・プロキシーのポート
Tivoli Enterprise Portal Server がパススルー・プロキシー・サーバー経由で通信する場合に使用します。プロキシーと通信するためのポート番号を入力します。

パススルー・プロキシーのプロトコル
Tivoli Enterprise Portal Server がパススルー・プロキシー・サーバー経由で通信する場合に使用します。プロキシー経由の通信に使用するプロトコルを入力します。デフォルト: `http`。

Cloud APM コンソールに表示されている Tivoli Monitoring エージェントは、IBM Tivoli Monitoring 環境内にあります。これらは Application Performance ダッシュボード・ページで表示できますが、これらのエージェントのしきい値を「**しきい値マネージャー**」で作成することはできません。

OMEGAMON との統合

z Systems Extension Pack を購入し、ハイブリッド・ゲートウェイを使用して、デプロイされている 1 つ以上の OMEGAMON エージェントを Cloud APM に接続することで、OMEGAMON アプリケーション・コンポーネントのデータおよびイベントを Cloud APM コンソールで表示できます。

始める前に

- z Systems Extension Pack を使用するには、IBM Cloud Application Performance Management, Advanced オファリング、または IBM Cloud Application Performance Management, Base オファリングを持っている必要があります。
- モニターする z Systems LPAR 上で、ライセンス交付を受けた 1 つ以上の OMEGAMON エージェントが実行されている必要があります。
- OMEGAMON エージェントは、IBM Tivoli Monitoring インフラストラクチャーに接続されます。

Cloud APM コンソールでモニターできる OMEGAMON エージェントのリストについては、IBM Knowledge Center の IBM OMEGAMON for Application Performance Management トピック・コレクション内のご使用のリリース用のトピック『はじめに』を参照してください。

手順

OMEGAMON を Cloud APM に統合するには、以下の手順を実行します。

1. z Systems Extension Pack がご使用の Cloud APM 製品に追加された後、以下の ハイブリッド・ゲートウェイのタスクを実行します。
   a) ハイブリッド・ゲートウェイ をインストールします。
   b) Cloud APM コンソールで表示する管理対象システム・グループを作成します。
   c) Cloud APM コンソールでハイブリッド・ゲートウェイを構成してハイブリッド・ゲートウェイを Tivoli Enterprise Portal Server に接続できるようにし、管理対象システム・グループを指定します。
   詳しくは、953 ページの『ハイブリッド・ゲートウェイ』のセクションにある必要なトピックを参照してください。

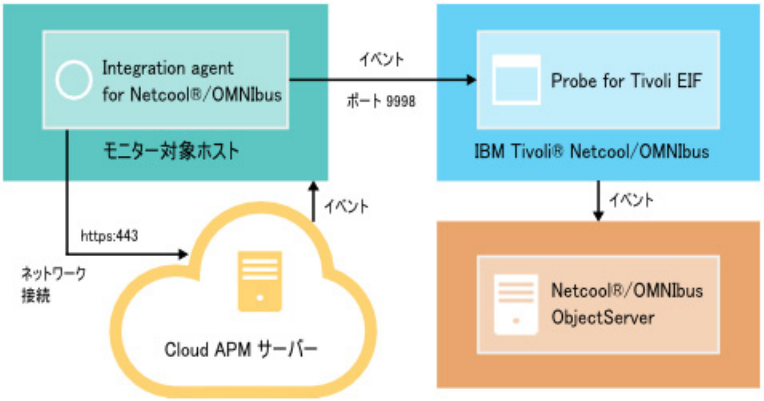
2. ダッシュボードでアプリケーションの状況を表示するには、ブラウザーから Cloud APM コンソールにログインします。詳しくは、977 ページの『Cloud APM コンソールの始動』を参照してください。

Netcool/OMNIbus との統合

IBM Cloud Application Performance Management からオンプレミスの IBM Tivoli Netcool/OMNIbus イベント・マネージャーにイベントを転送することができます。

手順

1. Integration Agent for Netcool/OMNIbus と、同製品が Cloud APM 内で Probe for Tivoli EIF とどのように統合されて Netcool/OMNIbus にイベントを転送するかを確認するには、以下の構成を参照してくださ



い。

Integration Agent for Netcool/OMNIbus は、Cloud APM サーバーに自動的に接続します。この接続により、インバウンド・ネットワーク接続がなくても、イベントがサーバーからネットワークに流れるようになります。

2. Netcool/OMNIbus の統合を構成します。

Integration Agent for Netcool/OMNIbus のインストールおよび構成

Integration Agent for Netcool/OMNIbus をインストールするには、IBM Marketplace Web サイトからアーカイブ・ファイルをダウンロードし、エージェント・インストール・ファイルを解凍してから、インストール・スクリプトを開始します。インストール後に、エージェントは自動的に開始されますが、構成する必要があります。

このタスクについて

一度に 1 つの Integration Agent for Netcool/OMNIbus のインスタンスのみが、Cloud APM サービス・サブスクリプションの単一インスタンスから Netcool/OMNIbus イベント・マネージャーにイベントを転送できます。

手順

1. 以下のようにして、Integration Agent for Netcool/OMNIbus を含む Cloud APM 統合アーカイブ・ファイルをダウンロードします。
   a) アカウントにサインインして、IBM Marketplace の製品およびサービスにアクセスします。
   b)「**IBM Performance Management**」の下で「**他の操作**」をクリックします。
   c)「**追加パッケージの表示**」をクリックします。
   d)「**IBM Performance Management OMNIbus Integration on Cloud**」を選択します。必要に応じて、この項目が見つかるまでスクロールダウンしてください。
   e)「**ダウンロード**」をクリックします。

2. ファイルは任意のステージング・ディレクトリーに保存します。Tivoli Netcool/OMNIbus Probe for Tivoli Event Integration Facility (EIF) とネットワーク接続されている任意のシステムにエージェントをインストールします。必要に応じて、モニター対象のシステムにインストール・アーカイブ・ファイルを転送します。アーカイブ・ファイルにはエージェントとインストール・スクリプトが含まれています。
3. インストール・ファイルを以下の手順で抽出します。

   Linux

   a. Red Hat Enterprise Linux システムで端末シェル・セッションを開きます。
   b. アーカイブ・ファイルが格納されているディレクトリーに移動します。
   c. 次のコマンドを使用して、インストール・ファイルを抽出します。

   ```
   tar -xf ./apm_integration_agents_xlinux_8.1.4.0.tar
   ```

   Windows

   apm_integration_agents_win_8.1.4.0.zip ファイルを抽出します。

   インストール・スクリプトは、このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーに解凍されます。例えば、IPM_Agent_Install_8.1.3.2 です。エージェント・バイナリーおよび構成関連ファイルは、そのディレクトリー内のサブディレクトリーに解凍されます。
4. このアーカイブ・ファイルとバージョンを表す名前が付けられたディレクトリーから、管理者特権を使用してインストール・スクリプトを実行します。

   Probe for Tivoli EIF が存在するのと同じシステムに Integration Agent for Netcool/OMNIbus をインストールし、Probe for Tivoli EIF がデフォルト・ポート 9998 を使用している場合、Integration Agent for Netcool/OMNIbus は Probe for Tivoli EIF と接続するように自動的に構成されます。

   重要 : Probe for Tivoli EIF が存在するシステムとは異なるシステムに Integration Agent for Netcool/OMNIbus をインストールする場合、または Probe for Tivoli EIF のデフォルトとは異なるポート番号を使用している場合、インストールが完了した後で Integration Agent for Netcool/OMNIbus を構成する必要があります。

   以下のステップに従ってエージェントをインストールします。

   Linux `installAPMAgents.sh`
   Windows `installAPMAgents.bat`

   Integration Agent for Netcool/OMNIbus のインストールを求めるプロンプトが出されます。

   ご使用の環境の前提条件スキャンが開始されます。このスキャンが完了するまでにはしばらく時間がかかります。要件が欠落している場合、失敗の原因が記録されたログ・ファイルを示すメッセージが表示されます。ライブラリーの欠落やディスク・スペースの不足といった前提条件の欠落が原因で、インストールが停止します。失敗に対処してから、インストールを再度開始する必要があります。

   エージェントは、以下のデフォルト設定を使用して構成されます。

   ```
   Tivoli EIF Probe Host: localhost
   Tivoli EIF Probe Port: 9998
   ```

   インストールが終わると、Integration Agent for Netcool/OMNIbus が自動的に始動します。

   モニター・エージェントは指定したディレクトリー (*install_dir*) にインストールされます。以下に示すデフォルト・ディレクトリーが使用されます。

   Linux `/opt/ibm/apm/agent`
   Windows `C:¥IBM¥APM¥`
5. Probe for Tivoli EIF が存在するシステムとは異なるシステムに Integration Agent for Netcool/OMNIbus をインストールしている場合、または Probe for Tivoli EIF がデフォルト・ポート 9998 とは異なるポート番号を使用している場合、Integration Agent for Netcool/OMNIbus を Probe for Tivoli EIF に接続するように構成する必要があります。

注 : Probe for Tivoli EIF が存在するシステムと同じシステムに Integration Agent for Netcool/OMNIbus がインストール済みで、Probe for Tivoli EIF がデフォルト・ポート 9998 を使用している場合、このステップを実行する必要はありません。

Linux エージェントを構成するには、以下のステップを実行します。

a. 次のコマンドを実行します。

```
install_dir/bin/omnibus-agent.sh config
```

b. プロンプトが出されたら、Probe for Tivoli EIF のホスト名とポート番号を指定してください。

構成が完了すると、Integration Agent for Netcool/OMNIbus が自動的に始動します。

Linux あるいは、以下のステップを使用して、構成設定を確認および変更することができます。

a. テキスト・エディターで応答ファイル *install_dir*/samples/omnibus_silent_config.txt を開きます。
b. ファイルを編集して構成設定を設定または変更します。構成行のコメントは必ず外してください。
c. 応答ファイルを保存して閉じます。
d. エージェントを再構成します。編集したサイレント構成ファイルの完全修飾パスを指定して以下のコマンドを実行します。

```
install_dir/bin/omnibus-agent.sh config install_dir/samples/omnibus_silent_config.txt
```

e. エージェントを再始動して変更内容を実装します。

```
install_dir/bin/omnibus-agent.sh stop
install_dir/bin/omnibus-agent.sh start
```

Windows エージェントを構成するには、以下のステップを実行します。

a. テキスト・エディターで *install_dir*¥samples¥omnibus_silent_config.txt 応答ファイルを開きます。
b. ファイルを編集して、Probe for Tivoli EIF のホスト名とポート番号を指定します。構成行のコメントは必ず外してください。
c. 応答ファイルを保存して閉じます。
d. 編集したサイレント構成ファイルへの完全修飾パスを指定して、エージェントを再構成します。

```
install_dir¥BIN¥omnibus-agent.bat config install_dir¥samples¥omnibus_silent_config.txt
```

e. エージェントを再始動して変更内容を実装します。

```
install_dir¥BIN¥omnibus-agent.bat stop
install_dir¥BIN¥omnibus-agent.bat start
```

次のタスク

『Netcool/OMNIbus の統合の構成』に記載された手順を実行します。

Integration Agent for Netcool/OMNIbus の使用を停止するか、エージェントを別のシステムに移動する場合、以下のコマンドを使用してエージェントをアンインストールします。

Linux *install_dir*/bin/omnibus-agent.sh uninstall
Windows *install_dir*¥BIN¥omnibus-agent.bat uninstall

Netcool/OMNIbus の統合の構成

Integration Agent for Netcool/OMNIbus のインストール後に、イベント・ルールを Probe for Tivoli EIF にコピーして変更する必要があります。Netcool/OMNIbus ObjectServer およびデータベース・スキーマを更新することも必要です。

始める前に

統合ステップを実行する前に、以下のコマンドを使用して Integration Agent for Netcool/OMNIbus を停止します。

Linux *install_dir*/bin/omnibus-agent.sh stop

Windows *install_dir*¥BIN¥omnibus-agent.bat stop

install_dir は、デフォルトの /opt/IBM/apm/agent ディレクトリーまたは C:¥IBM¥APM ディレクトリー、もしくは Integration Agent for Netcool/OMNIbus のインストール時に指定したディレクトリーです。

このタスクについて

Integration Agent for Netcool/OMNIbus をインストールすると、以下の構成ファイルが *install_dir*/localconfig/i0/omnibus ディレクトリーと *install_dir*¥localconfig¥i0¥omnibus ディレクトリーに作成されます。

- itm_apm_db_update.sql
- itm_event.rules
- itm_apm_event.rules

ここで *install_dir* は、デフォルトの /opt/IBM/apm/agent または C:¥IBM¥APM、もしくは Integration Agent for Netcool/OMNIbus のインストール時に指定したディレクトリーです。

重要 : この手順は、ご使用の Probe for Tivoli EIF および Netcool/OMNIbus ObjectServer が IBM Tivoli Monitoring、 Probe for Tivoli EIF、IBM SmartCloud Monitoring - Application Insight®、IBM SmartCloud Application Performance Management、または Cloud APM の以前のバージョンと既に統合されている場合でも実行する必要があります。

手順

この手順で、IBM Tivoli Monitoring 資料へのリンクをたどるときには、該当のリンク先ページに示されたステップのみを実行してください。

1. Integration Agent for Netcool/OMNIbus itm_event.rules ファイルと itm_apm_event.rules ファイルを Probe for Tivoli EIF インストール・ディレクトリーにコピーします。
   以下は、デフォルトのディレクトリーです。

   Linux *install_dir*/tivoli/netcool/omnibus/probes/linux2x86

   Windows *install_dir*¥Tivoli¥Netcool¥omnibus¥probes¥win32

   ここで *install_dir* はデフォルトです。
2. Probe for Tivoli EIF の tivoli_eif.rules ファイルをテキスト・エディターで開き、以下のいずれかのステップを実行します。
   - ユーザーが IBM Tivoli Monitoring の既存のお客様であり、OMNIbus の統合を既に完了している場合は、itm_event.rules ファイルに「include "itm_apm_event.rules"」という行を追加します。
   - OMNIbus の統合のセットアップがまだ完了していない場合は、itm_event.rules ファイルを参照する行からコメントを削除します。

   詳しいステップについては、IBM Tivoli Monitoring 資料の EIF プローブのルール・ファイルの更新を参照してください。
3. 多層 OMNIbus ソリューションを使用している場合は、IBM Tivoli Monitoring の資料の『IBM Tivoli Monitoring の属性、テーブル、トリガーによる Netcool/OMNIbus ObjectServer の更新』セクションに記載されたすべてのタスクを完了させます。
4. 以下のコマンドで itm_apm_db_update.sql ファイルをデータベースにロードして、Netcool/OMNIbus ObjectServer データベース・スキーマを更新します。

   Linux

```
$OMNIHOME/bin/nco_sql -user user_name -password password
-server server_name < itm_apm_db_update.sql
```

例:

```
$OMNIHOME/bin/nco_sql -user smadmin -password passw0rd -server NCOMS <
/tmp/apm/itm_apm_db_update.sql
```

Windows

```
itm_apm_db_update.sql | %OMNIHOME%¥..¥bin¥isql -U user_name
-P password -S server_name
```

例:

```
¥temp¥apm¥itm_apm_db_update.sql | %OMNIHOME%¥..¥bin¥isql -U smadmin
-P passw0rd -S NCOMS
```

スクリプトの実行時に以下のエラー・メッセージが表示される場合がありますが、これらのメッセージは無視してもかまいません。

- `Object exists` および `Attempt to insert duplicate row`。スクリプトが以前実行された場合に表示される (古いバージョンの Cloud APM との統合や Tivoli Monitoring との統合など)。
- `ERROR=Object not found on line 4 of statement "-- A workspace table for the ITM event clear automation..."`。`itm_event_clear` またはその近くで表示される。
- `ERROR=Object not found on line 1 of statement "delete from alerts.itm_problem_events;..."`。`itm_problem_events` またはその付近で表示される。
- `ERROR=Object not found on line 1 of statement "drop table alerts.itm_problem_events;..."`。`itm_problem_events` またはその付近で表示される。

5. ステップ 5 を繰り返します。これによりファイルが Object Server に 2 回ロードされ、すべての依存関係が確実に正しくロードされます。
6. Probe for Tivoli EIF を始動 (または再始動) します。
7. 以下のコマンドを使用して、Integration Agent for Netcool/OMNIbus を再始動します。

   Linux *install_dir*/bin/omnibus-agent.sh start

   Windows *install_dir*¥BIN¥omnibus-agent.bat start

Operations Analytics - Log Analysis との統合

ご使用の環境に IBM Operations Analytics - Log Analysis が含まれている場合、これを統合して、Cloud APM コンソールでアプリケーション・ログを検索できるようにすることができます。

このタスクについて

インストール済みの Log Analysis アプリケーションと統合するには、URL を使用して Cloud APM サーバーを構成する必要があります。Log Analysis について詳しくは、IBM Operations Analytics - Developers Community を参照してください。

Log Analysis インストール済み環境のトップレベル URL を指定する必要があります。以下に例を示します。

```
https://loganalysis.example.com:9987/Unity
```

Log Analysis の URL は、ユーザーが Cloud APM コンソールで作業するホストからアクセス可能にする必要があります。公開インターネットからアクセス可能にする必要はありません。

手順

1. Cloud APM コンソールで、 「**システム構成**」 > 「**拡張構成**」をクリックします。

2.「**UI 統合**」カテゴリーを選択します。
3.「**Log Analysis の URL**」フィールドに、Log Analysis アプリケーションを起動するために使用する URL を入力します。

タスクの結果
Log Analysis アプリケーションが統合され、Application Performance ダッシュボードからアプリケーション・ログを検索する機能が有効になります。

注 :

Cloud APM コンソールから Log Analysis アプリケーションへのシングル ・サインオンはサポートされていません。

次のタスク
「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」を選択します。オプションで、アプリケーションを選択し、 検索ボックスを使用してログ・ファイルを検索します。デフォルトでは直近 1 時間分のエントリーが検索されますが、この期間は変更できます。アプリケーションを選択すると、そのアプリケーションに関連したサーバーのログのみが検索されます。説明については、1081 ページの『ログ・ファイルの検索』を参照してください。

Operations Analytics - Predictive Insights との統合

IBM Cloud Application Performance Management を Operations Analytics - Predictive Insights と統合すると、Operations Analytics - Predictive Insights は Cloud APM によって収集されたメトリック・データを分析し、データの異常を検出するとアラームを生成します。

異常は Cloud APM ダッシュボードにイベントとして表示されます (1111 ページの『Operations Analytics - Predictive Insights を使用した異常の調査』を参照してください)。その後、Operations Analytics - Predictive Insights ユーザー・インターフェースをドリルダウンして 、異常の詳細を表示できます。

Operations Analytics - Predictive Insights を Cloud APM サブスクリプションに追加すると、パフォーマンス・メトリックを収集して分析するように自動的に構成されます。これ以上の構成は不要です。Operations Analytics - Predictive Insights を Cloud APM サブスクリプションに追加するには、IBM Support にアクセスし、サービス要求をオープンします。

Operations Analytics - Predictive Insights と統合できるのは以下の Cloud APM エージェントです。

- Monitoring Agent for Db2
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for JBoss
- Monitoring Agent for Linux OS
- Monitoring Agent for Oracle Database
- Response Time Monitoring エージェント
- Monitoring Agent for UNIX OS
- Monitoring Agent for VMware VI
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ
- Monitoring Agent for Windows OS
- Monitoring Agent for Tomcat

Alert Notification との統合

Cloud APM の基本的な E メール転送を超えた柔軟性を実現するために、Alert Notification 製品と統合できます。これにより、問題が発生した場合に、ユーザーに通知が送信されるように Cloud APM 機能を拡張できます。

Alert Notification との統合により、通知を誰がどのように受信するか詳細に制御できます。例えば、各ユーザーは、E メール、SMS、またはボイス・メールのうちのどれを受信するか決定できます。通知を Slack に送付することもできます。各ユーザーは、時刻、曜日などに基づいて、各タイプの通知を受信できます。各ユーザーは、どのタイプのアラートを受信するか決定できます。例えば、データベース管理者は、警告以上の重大度のデータベース・アラートのみを受信できます。

始める前に

アラート通知は、IBM Cloud Application Performance Management サブスクリプションおよび試用サブスクリプションに自動的に統合されます。

このタスクについて

イベントは、Alert Notification に送信されるように自動的に構成されます。次に、サブスクリプションが通知を受信するアラートを判別するための通知ポリシーを作成できます。

モニター対象アプリケーションをリソース・グループに追加できます。保存したリソース・グループごとに、1 つ以上の E メール・アドレスを構成できます。グループ内のいずれかの管理対象システムのパフォーマンスがしきい値を超えると、そのグループに対して構成されているアドレスに E メール通知が送信されます。

Alert Notification は、グループに割り当てられた管理対象システムで発生したすべてのイベントについて E メール通知を送信します。

手順

リソース・グループに基づいてアラートを送付する場合、その対象のリソース・グループに対してステップ 972 ページの『1』 と 972 ページの『2』 を実行します。

1. リソース・グループに基づいて通知を送付する場合、以下のサブステップを実行します。
   a) Cloud APM コンソールで、「**システム構成**」 > 「**リソース・グループ・マネージャー**」をクリックします。
   b) リソース・グループを選択して、「**編集**」をクリックし、「**リソース・グループ・エディター**」を開きます。

      重要 : Alert Notification を開く前に、新しいリソース・グループまたは編集済みのリソース・グループを保存する必要があります。保存後、リソース・グループ・エディターが閉じるため、引き続き使用する場合は再び開く必要があります。

      保存しない場合、ステップ 972 ページの『2』 で E メール通知を構成し、リソース・グループ・エディターに戻ると、同期更新が許可されないことを通知するエラー・メッセージを受け取ることがあります。
2. 以下のステップを実行して E メール通知を構成します。
   a) APM コンソールのリソース・グループ・エディターで、「**E メール通知の構成**」 の URL をクリックして、新しいブラウザー・タブまたはウィンドウで IBM Alert Notification アプリケーションを開きます。

      このアクションにより、ステップ 972 ページの『1』 で選択したリソース・グループに基づいて新しいポリシーが自動的に作成されます。

      重要 : ご使用のブラウザーがポップアップ・ウィンドウを許可しない場合、Alert Notification ウィンドウは開きません。リソース・グループの E メール通知を構成するために Alert Notification ウィンドウを開くことができるように、ブラウザーを設定する必要があります。

b) イベント通知を E メールで受信するには、Alert Notification でユーザーとグループを構成し、それらの E メール・アドレスをリソース・グループに関連付けます。

受信するアラートのタイプについて、独自の Alert Notification ポリシーおよび通知をセットアップできます。例えば、Linux 管理者は、すべての Linux システムに関する E メール、SMS メッセージ、またはボイス・メールを受信することが考えられます。また、重大な重大度アラートをオフにすることが考えられます。ポリシーにより、Linux OS エージェントから生成されたすべてのイベントをフィルターに掛けることができます。Alert Notification アプリケーションの Notifications Editor の使用について詳しくは、Creating notification policies を参照してください。

次のタスク
Alert Notification について詳しくは、IBM Alert Notification Documentation を参照してください。

Control Desk との統合

Cloud APM からイベントが、IBM Control Desk で自動的にチケットをオープンするように 構成することができます。

このタスクについて
IBM Control Desk のオンプレミス版またはクラウド版を使用して、IBM Cloud Application Performance Management を統合できます。

手順

以下のいずれかの手順を使用します。

- オンプレミス IBM Control Desk V 7.6 でチケットをオープンするには、以下の手順を実行します。

1. Control Desk 用の IBM SmartCloud Notes E メール・アカウントを、IMAP メール・クライアントを使用するように構成します。構成時に、「**今すぐ IMAP アクセスを有効にする**」を必ず選択する必要があります。詳しくは、IBM Connections Social Cloud Knowledge Center の『IMAP アクセスの有効化』を参照してください。
2. Cloud APM コンソールで、「**システム構成**」>「**拡張構成**」をクリックしてから、以下のパラメーターを設定します。

   ターゲット E メール・アドレス
   サービス要求チケットの作成に使用される IBM SmartCloud Notes E メール・アドレスを指定します。

   E メールの件名行
   E メールの件名行 (例えば、`PMaaS Event`) を指定します。
3. Marketplace サポートにアクセスし、「**Service Request**」を選択してサポート・チケットを送信し、使用可能化を完了します。

   チケットには以下の情報を指定します。

   – IBM SmartCloud Notes E メール・アドレス

     例えば、`user@ibmserviceengage.com` です。
   – IBM SmartCloud Notes E メールのパスワード
   – IBM SmartCloud Notes E メールの完全修飾サーバー名

     例えば、`imap.notes.na.collabserv.com` です。
   – IBM SmartCloud Notes E メールのポート番号

     例えば、993 です。
   – 顧客用 IBM Control Desk URL

リンクの書式は次のようになります。`https://<subscriber-id>.sccd.ibmserviceengage.com/maximo_t4hj/webclient/login/login.jsp?welcome=-true`

4. E メールを解析し、IBM Control Desk on Cloud 内の他のグループにチケットを割り当てるときにそれを適切に処理するように電子メール・リスナーを構成するには、『電子メール・リスナーの構成』を参照してください。

- IBM Control Desk Cloud 環境でチケットをオープンするには、以下のステップを実行します。

1. Marketplace サポートにアクセスし、「**Service Request**」を選択してサポート・チケットを送信し、使用可能化を完了します。

   チケットには以下の情報を指定します。

   – IBM SmartCloud Notes E メール・アドレス

     例えば、`user@ibmserviceengage.com` です。

   – IBM SmartCloud Notes E メールのパスワード

   – IBM SmartCloud Notes E メールの完全修飾サーバー名

     例えば、`imap.notes.na.collabserv.com` です。

   – IBM SmartCloud Notes E メールのポート番号

     例えば、993 です。

   – 顧客用 IBM Control Desk URL

     リンクの書式は次のようになります。`https://<subscriber-id>.sccd.ibmserviceengage.com/maximo_t4hj/webclient/login/login.jsp?welcome=-true`

2. E メールを解析し、IBM Control Desk on Cloud 内の他のグループにチケットを割り当てるときにそれを適切に処理するように電子メール・リスナーを構成するには、『電子メール・リスナーの構成』を参照してください。

IBM Cloud との統合

特定のデータ・コレクターを使用して、IBM Cloud 環境内のアプリケーションのモニター情報を表示できます。

IBM Cloud アプリケーションからデータを収集するようにデータ・コレクターが構成されている場合は、そのデータ・コレクターによって IBM Cloud にモニター機能を統合できるようになります。データ・コレクターは、IBM Cloud アプリケーションに関するリソース・モニター・データおよび診断データを Cloud APM サーバーに転送します。Cloud APM サーバーは、データ・コレクターによって収集されたモニター情報を受信し、処理します。モニターできる IBM Cloud アプリケーションのタイプは、以下のとおりです。

- Liberty アプリケーション
- Node.js アプリケーション
- Python アプリケーション
- Ruby アプリケーション

データ・コレクターを適切に構成した後、Cloud APM コンソールでモニター・データを表示できます。構成の手順については、189 ページの『データ・コレクターの一般的な構成手順』を参照してください。

IBM Agent Builder との統合

Agent Builder を使用して、IBM Tivoli Monitoring または IBM Cloud Application Performance Management 環境のモニター機能を拡張するエージェントを作成、変更、デバッグ、およびパッケージ化できます。カスタム・エージェントは、これらの環境のいずれかを使用して、すべてのタイプの社内ソフトウェアおよびカスタマイズ済みのソフトウェアをモニターできます。

詳しくは、「IBM Agent Builder ユーザーズ・ガイド」を参照してください。

第 9 章 管理

Cloud APM コンソールの始動

ブラウザーから Cloud APM コンソール にログインし、ダッシュボードでアプリケーションの 正常性状況を確認します。

始める前に

- このサービスへの初回のサインアップの 後に受け取った確認用の E メールに記載されているリンクを使用して、アカウントをアクティブにします。
- ユーザー・インターフェースが 切り捨てられないようにするには、最小解像度の 1280 x 1024 を使用します。
- 最適なパフォーマンスを得るには、サポートされるいずれかのブラウザーを使用してください。サポートされるブラウザーのリストについては、System requirements (APM Developer Center) を参照してください。いずれかの IBM Cloud Application Performance Management 製品リンクを選択し、「Server」リンクをクリックします。表示されたレポートで、「Web Browsers」をクリックするか「Web Browsers」までスクロールダウンします。

手順

1. Cloud APM コンソール にアクセスするには、サービスが使用可能になったことを知らせる E メールに記載されているリンクを使用します。
2. このコンソールには IBM Marketplace Web サイトからもアクセスできます。
   a. IBM Marketplace Web サイトの 製品およびサービス にアクセスします。
   b. このサービスへの登録に使用したユーザー名とパスワードでログインします。
   c.「Cloud APM サーバー」行で、「**起動**」をクリックします。

タスクの結果

ログインすると、「**はじめに**」ページに 「**ユーザー・タスク**」および「**管理者タスク**」の学習オプションが表示されるほか、 「**コミュニティー・リソース**」へのリンクも表示されます。

次のタスク

- Cloud APM ダッシュボードのツアーを行うためのハイパーテキスト・リンクをクリックして、ユーザー・インターフェースの 要素を理解します。 ユーザー・タスクおよび管理者タスクのビデオを視聴すると、Cloud APM 環境の使用とカスタマイズを開始するのが容易になります。
- Online Ordering などの論理グループに 含まれるリソースのダッシュボードを表示するためのアプリケーションを追加します。説明については、1098 ページの『アプリケーションの管理』を参照してください。
- 条件のテストに使用するしきい値を作成します。このしきい値に到達すると、イベントがオープンされます。例えば、ストレージ容量が 90% に到達するとイベントがオープンされるようにしきい値を定義することができます。説明については、987 ページの『しきい値マネージャー』を参照してください。
- Cloud APM コンソール の機能および管理対象リソースに対するアクセスを制御するには、 ユーザーを追加し、ユーザー・グループおよび役割に割り当てます。詳しくは、1003 ページの『ユーザー・アクセスの管理』を参照してください。
- IBM Java アプリケーション・スタックと IBM 統合スタックのモニターについて詳しくは、90 ページの『シナリオ』を参照してください。

© Copyright IBM Corp. 2014, 2019

- 「はじめに」ページでも Application Performance ダッシュボードでもなく IBM の Web サイトがブラウザーで開く場合は、ご使用のユーザー ID が Cloud APM コンソール に対する権限を持っていません。管理者にアクセス権を要求する必要があります。
- データ・ソースのメトリックが表示されない場合は、developerWorks の Cloud Application Performance Management フォーラム を参照してください。フォーラムで「dashboard」を検索し、エントリーに返答して関連質問をするか、新規エントリーを作成して症状を記述してください。
- Cloud APM コンソールを Internet Explorer 8、9、または 10 から開始したときに、「このページは表示できません」というエラーが表示される場合は、セキュリティー・オプション TLS 1.2 の有効化が必要であることがあります。詳しくは、Cloud Application Performance Management フォーラムにアクセスし、「tls」を検索してください。

しきい値とリソース・グループ

しきい値は、100 未満のトランザクションの数 (1 分あたり) などの特定の条件をテストし、条件が満たされたときにイベントをオープンします。しきい値は、ご使用のモニター対象のリソースにある実際の問題や潜在的な問題をモニターするために使用します。リソース・グループに属する同じタイプのすべての管理対象システムをモニターするために、しきい値をそのグループに割り当てます。

背景情報

しきい値、エージェントの事前定義しきい値、事前定義しきい値の割り当て先のリソース・グループ、およびしきい値のカスタマイズについて確認するには、背景情報を参照してください。

事前定義しきい値

モニター・エージェントには、エージェントとともに使用可能になり開始される事前定義しきい値が付属しています。エージェントのインストール後に最初に「**しきい値マネージャー**」を開いたときに、選択したデータ・ソース・タイプについてリストされるしきい値が事前定義しきい値です。これらの事前定義しきい値は、エージェントのデフォルトのシステム・リソース・グループに割り当てられ、「**割り当て済みグループ**」欄に表示されます。

事前定義しきい値を編集すると (名前や条件の変更など)、そのしきい値は事前定義しきい値としては処理されなくなり、カスタムしきい値とみなされるようになります。ただし、事前定義しきい値に割り当てられたリソース・グループを、デフォルトのシステム・グループからユーザー定義のグループに変更できますが、この場合は、このしきい値は事前定義しきい値のままです。

事前定義しきい値を使用しない場合は、「**拡張構成**」ページで事前定義しきい値をオフにできます (「1076 ページの『しきい値の使用可能化』」を参照してください)。事前定義しきい値を使用不可にしても、「**しきい値マネージャー**」からはしきい値は削除されません。しきい値のグループ割り当てが削除され、非アクティブと表示されるのみです。事前定義しきい値を使用不可にした後、「**しきい値マネージャー**」を開いて、すべての事前定義しきい値の「**割り当て済みグループ**」欄が空であることを確認できます (980 ページの『使用不可のしきい値の例』を参照してください)。

しきい値を使用可能なリソース・グループに割り当てると、カスタムしきい値として使用可能にできます。

カスタムしきい値

作成する新しいしきい値は、「**しきい値マネージャー**」の「**起点**」欄に示されるように、カスタムしきい値です。事前定義しきい値を編集した場合も、カスタムしきい値になり、その起点が「事前定義」から「カスタム」に変化します。

実行コマンド

true に評価されるしきい値に対してイベントがオープンされた後で、そのイベントがオープンされたモニター対象システムに対してコマンドまたはコマンド・スクリプトを自動で実行することができます。例えば、情報のログ記録、ビープ音のトリガー、リソースを過剰に使用しているジョブの停止などを実行できます。

このコマンドでは、以下の構文を使用します。

```
&{data_set.attribute}
```

ここで、*data_set* はデータ・セット名であり、*attribute* はしきい値エディターに表示される属性名です。データ・セット名または属性名にスペースが含まれている場合は、下線に置き換えてください。

次の例は、ディスク名パラメーターを管理対象リソースに渡す方法を示しています。

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

データ・セットの 1 つ以上の属性を渡すことができます。複数の属性が指定されている場合、それらの属性は ($1、$2、...) の順でコマンドに渡されます。

コマンドは、エージェントの始動に使用したものと同じユーザー・アカウントを使用してコマンド行から実行します。例えば、エージェントが root として実行されている場合、root ユーザーが管理対象システムでコマンドを実行します。

以下のオプションは、コマンドの実行頻度を制御します。

データ・セットから複数の行が返された場合、データ・サンプルの最初のイベント発生時にのみコマンドを実行するには、☐「**最初のイベント時のみ**」を選択します。イベントを発生させた各行に対してコマンドを実行するには、このチェック・ボックスをクリアします。

しきい値が true に評価されるたびにコマンドを実行するには、☐「**true が連続する間は毎回**」を選択します。しきい値が true である場合はコマンドを実行するが、しきい値が false と評価された後、時間間隔経過後に再び true と評価されるまではコマンドを再実行しない場合は、このチェック・ボックスをクリアします。

リソース・グループ

リソース・グループは、管理対象システムのコレクションを表し、しきい値の配布状況を制御します。しきい値は、実行する管理対象システムが含まれているリソース・グループに割り当てます。

すべての事前定義しきい値には、エージェント・タイプ (Db2、Microsoft IIS など) に対するシステム定義グループであるデフォルトのリソース・グループ割り当てがあります。

カスタム・リソース・グループを作成し、各グループを含める管理対象システムを選択できます。カスタム・リソース・グループには複数のエージェント・タイプを設定できます。グループに割り当てられるしきい値は、同じエージェント・タイプの管理対象システムにのみ配布されます。例えば、Linux OS 属性で作成され、Linux OS、MongoDB、および Python の各管理対象システムのリソース・グループに割り当てられたしきい値は、Linux OS 管理対象システムにのみ配布されます。

詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。

Application Performance ダッシュボードのイベント状況

Application Performance ダッシュボードに表示される状況の重大度は、選択したアプリケーション、グループ、サブグループ、および管理対象システム・インスタンスの最高のイベント重大度を示します。

ナビゲーターまたは「**すべてのマイ・アプリケーション**」ダッシュボードの要約ボックスからアプリケーションを選択すると、タブ付きのダッシュボードに、アプリケーションのさまざまな側面が表示されます。「**イベント**」タブは、1109 ページの『イベント状況』に説明されているように、選択したナビゲーター項目に関する情報を示します。

しきい値の変更が、同じモニター・エージェントに割り当てられている他のしきい値に及ぼす影響

しきい値の定義の作成、変更、または削除を行った後や、モニター・エージェントに配布するしきい値のリストを変更した後は、そのしきい値の配布先エージェントのすべてのサンプル・イベントがクローズされます。イベントがクローズした後、true と評価されたすべてのしきい値条件のイベントをモニター・エージェントが再オープンします。Cloud APM コンソールでは、クローズされたイベントは、新しい「**タイム・スタンプ**」の値で再オープンされるまで、コンソールに表示されません。イベントの E メール通知を受信している場合は、イベントのクローズとイベントのオープンに関する E メール通知が届きます。

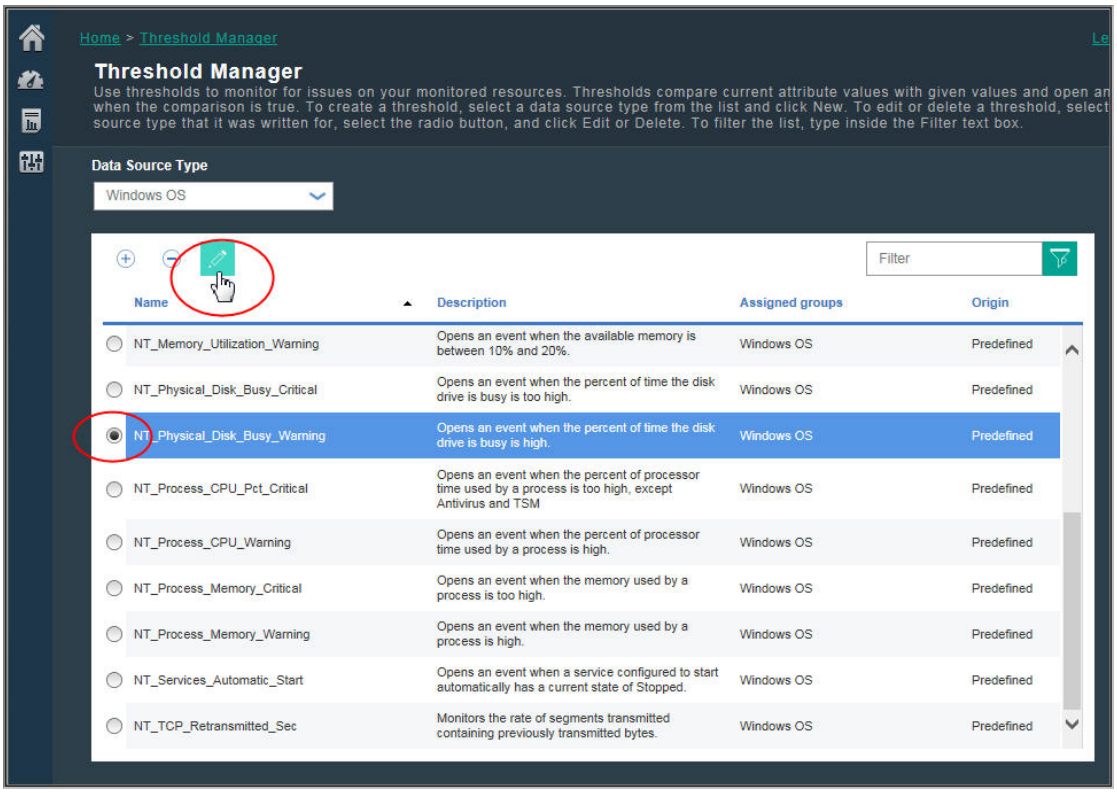
例えば、「Site Systems」という名前のカスタム・リソース・グループがあり、Linux OS と WebSphere アプリケーションのしきい値およびエージェントが割り当てられているとします。新しい Linux OS しきい値を作成し、「Site Systems」に割り当てるとします。すると、「Site Systems」に割り当てられている Linux OS エージェントに関するオープン状態のサンプル・イベントはすべてクローズされます。その後、まだしきい値条件が true であれば、そのサンプル・イベントが再オープンされます。

使用不可のしきい値の例

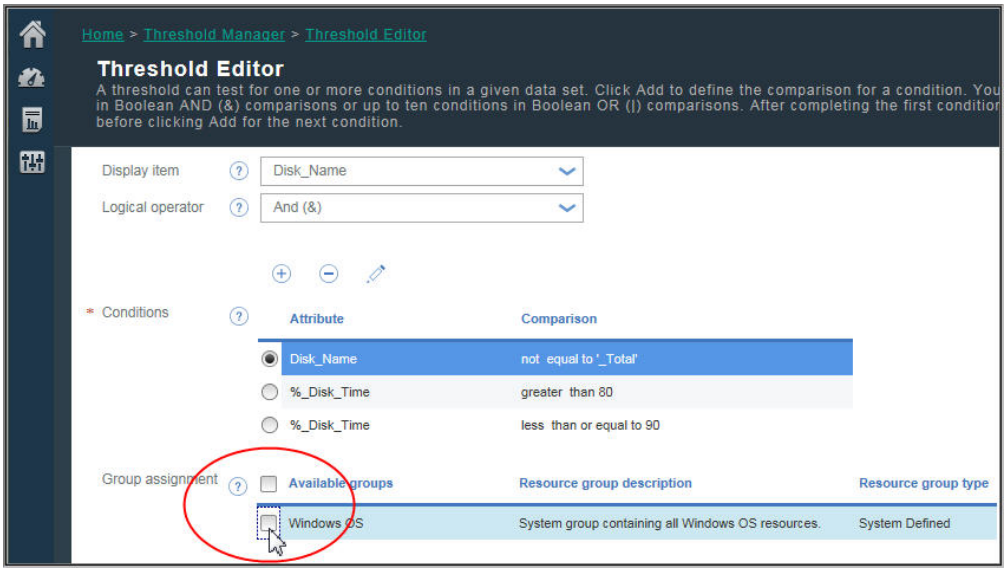
環境内のすべてのエージェントの事前定義しきい値を使用不可にできます。事前定義かカスタムかに関係なく、しきい値を個別に使用不可にすることもできます。しきい値を使用不可にすると、そのしきい値は管理対象システムで実行されず、イベントはオープンされません。しきい値を使用不可にするには、そのリソース・グループ割り当てを削除します。「**拡張構成**」設定を使用して、すべてのエージェントの事前定義しきい値を使用不可にすることもできます。

単一のしきい値の使用不可化

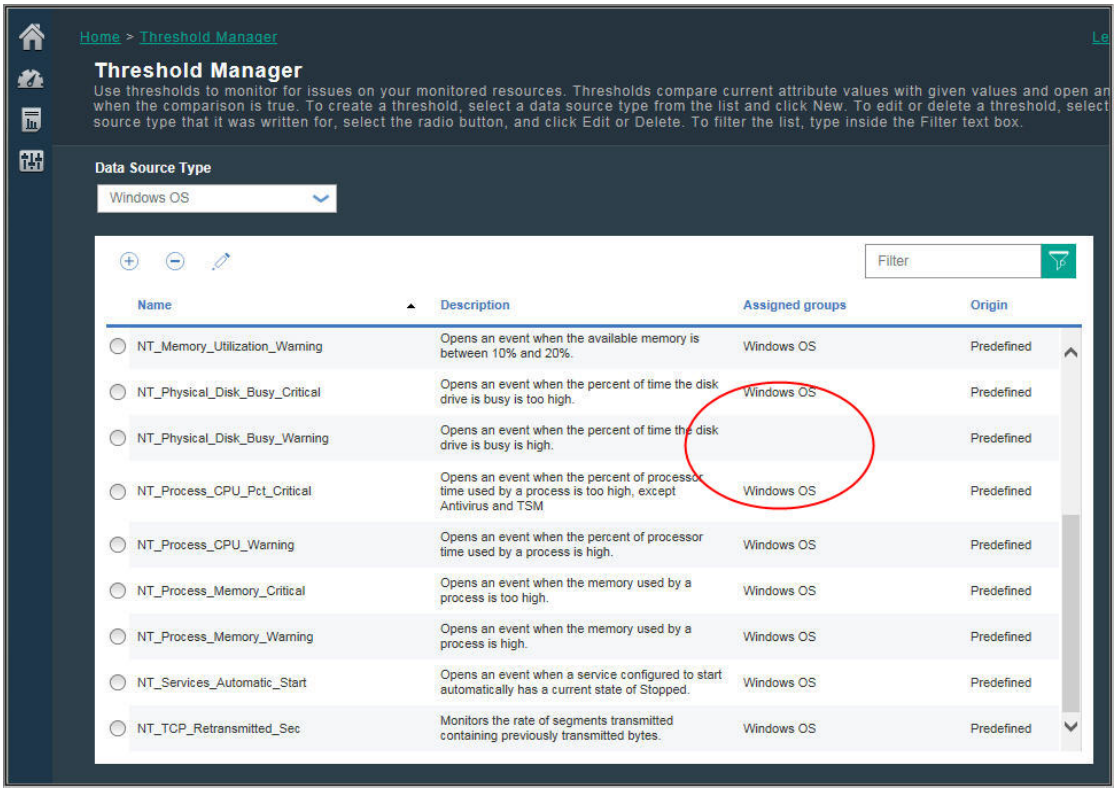
このイメージでは、「**しきい値マネージャー**」で使用不可化対象のしきい値が選択されていて、ユーザーが「**編集**」をクリックします。



しきい値エディターでしきい値が開かれます。ユーザーは、「**グループ割り当て**」フィールドにある割り当てられたリソース・グループのチェック・ボックスをクリアします。

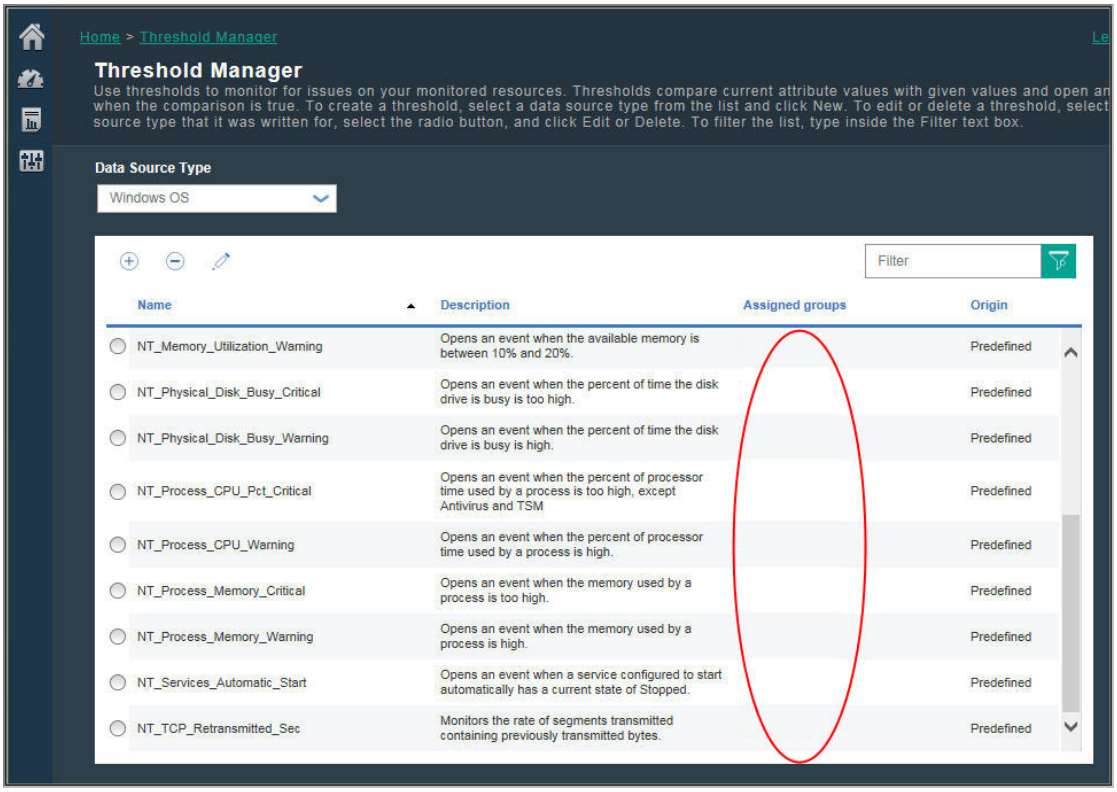


ユーザーが「**保存**」をクリックすると、「**しきい値マネージャー**」が表示されます。しきい値が使用不可になり、「**割り当て済みグループ**」 欄が空になります。



すべての事前定義しきい値の使用不可化

1076 ページの『しきい値の使用可能化』の説明に従って、「**拡張構成**」ページですべてのモニター・エージェントのすべての事前定義しきい値をオフにします。次回「**しきい値マネージャー**」を開いたときに、「**割り当て済みグループ**」欄はすべての事前定義しきい値について空であり、しきい値が非アクティブであることが示されます。



関連概念

978 ページの『背景情報』

しきい値、エージェントの事前定義しきい値、事前定義しきい値の割り当て先のリソース・グループ、およびしきい値のカスタマイズについて確認するには、背景情報を参照してください。

関連資料

987 ページの『しきい値マネージャー』

リソース・グループ・マネージャー

モニター対象環境には、その目的により分類できる複数の管理対象システムが存在する場合があります。そのようなシステムには通常、同じしきい値要件が存在します。「**リソース・グループ・マネージャー**」を使用して管理対象システムをグループに編成し、しきい値を割り当てることができます。役割ベースのアクセス制御 (RBAC) ポリシーに関連するリソース・グループを作成することもできます。

「**システム構成**」 > 「**リソース・グループ・マネージャー**」をクリックすると、定義済みのリソース・グループの表のページが表示されます。最初は、インストールされているモニター・エージェント・タイプ (Windows OS など) ごとに 1 つの事前定義システム・グループが表示されます。. 各システム・グループには、エージェントのすべての事前定義されたしきい値が含まれています。

「**リソース・グループ・マネージャー**」およびリソース・グループへのアクセスは、ユーザー権限によって制御されます。リソース・グループを表示するには、そのグループの表示権限が必要です。リソース・グループを作成、編集、または削除するには、変更権限が必要です。

この表にはリソース・グループを管理するための次のようなツールがあります。

- 「**新規**」は、管理対象システムおよびしきい値を割り当てるための「**リソース・グループ・エディター**」を開きます。
- リソース・グループを選択して、隣接するペインで、グループに割り当てられているリソースとしきい値を表示します。
- 管理対象システムおよびしきい値の割り当てを変更するには、リソース・グループを選択して「**編集**」をクリックし、「**リソース・グループ・エディター**」を開きます。
- 不要なリソース・グループを削除するには、リソース・グループを選択して「**削除**」をクリックします。 削除の確認後、そのグループに割り当てられていたしきい値を管理対象システムで引き続き実行したい場合は、別のグループに割り当てる必要があります。
- フィルター・テキスト・ボックス 内をクリックして、フィルタリングの条件とする値を入力できます。入力するにしたがい、条件に一致しない行は除外されていきます。フィルターをクリアするには、フィルター・ボックス の をクリックするか、Backspace キーを押します。

この表には、次のように使用可能なリソース・グループが表示されます。

リソース・グループ名
事前定義グループにはエージェント・タイプの名前が付けられ、カスタム・グループには作成者によって名前が付けられます。

リソース・グループの説明
事前定義グループの説明は、モニター対象リソースのシステム・グループ です。カスタム・グループの説明は、作成者が記入します。

Linux OS などのシステム・グループには、エージェントのすべての事前定義しきい値、およびエージェントがインストールされているすべての管理対象システムが含まれます。システム・グループを編集して、しきい値の割り当てまたは削除を行うことはできますが、管理対象システムの割り当てまたは削除を行うことはできません。管理対象システムは、同じタイプのシステム・グループに自動的に割り当てられます。ハイブリッド・ゲートウェイを構成した場合は、Tivoli Monitoring ドメインからのすべての管理対象システムも含まれます。

一部のシステム・リソース・グループは、サブノードをサポートするエージェントに関連しています。エージェントのタイプに応じて、サブノード、エージェント・ノード、あるいは両方をアプリケーションに追加できます。サブノードのみを定義済みアプリケーションに追加できる場合は、エージェント・ノードに定義されたどのしきい値のイベントも表示できません。ただし、イベントを Netcool/OMNIbus などのイベント・マネージャーに転送することはできます。また、IBM Cloud Application Performance Management サブスクライバーは、Alert Notification をセットアップできます。

リソース・グループ・タイプ
事前定義グループのタイプは、システム定義 です。環境にインストールしたすべてのタイプのエージェントに事前定義グループがあります。

環境内でユーザーが作成するカスタム・グループのタイプは、ユーザー定義 です。

リソース・グループ・エディター

「**新規**」をクリックしてグループを追加するか、グループを選択してから「**編集**」をクリックしてグループを編集すると、以下のフィールドを持つ「**リソース・グループ・エディター**」が表示されます。

グループ名
グループの名前は必須です。既存のカスタム・グループ名を変更することができます。変更すると、変更の保存後にそのグループへの参照はすべて自動的に更新されます。

グループの説明
カスタム・グループの場合はオプションです。グループ組織の説明を追加します。この説明は「**リソース・グループ・マネージャー**」に表示されます。

リソース割り当て
グループに追加するために使用できるすべての管理対象システムは、その管理対象システム名、ホスト名、エージェント・タイプ、およびそのドメイン別にエージェント・リストに表示されます。列見出しをクリックすることで、エージェント名、ホスト名、タイプ、またはドメインでリストをソートできます。

グループに追加するには、1 つ以上の管理対象システムのチェック・ボックスを選択します。

「**選択したリソースのみを表示**」を選択すると、割り当てられていない管理対象システムを非表示にすることができます。

IBM Cloud Application Performance Management ハイブリッド・ゲートウェイを構成した場合は、管理対象システムを IBM Tivoli Monitoring ドメインからユーザー定義リソース・グループに追加できます。Tivoli Monitoring 管理対象システムをシステム 定義グループに追加したり、システム 定義グループのしきい値を作成したりすることはできません。

しきい値割り当て
事前定義されたしきい値または「**しきい値マネージャー**」で追加されたしきい値はすべて、名前別およびエージェント・タイプ別にしきい値リストに表示されます。列見出しをクリックすることで、リストをソートできます。

しきい値をグループに追加するには、名前の横のチェック・ボックスを選択します。グループからしきい値を削除するには、そのチェック・ボックスをクリアします。 しきい値を追加または削除するには、「**しきい値マネージャー**」の表示権限が必要です。システム・グループにしきい値を追加する場合、使用可能なしきい値は、データ・セットがそのシステム・グループに適しているしきい値のみに制限されます。

グループに割り当てたしきい値は、同じエージェント・タイプのグループ内のすべての管理対象システムに配布されます。どのようなタイプのモニター・エージェントのしきい値でもグループに割り当てることができますが、割り当てられたしきい値は、そのグループのメンバーである、同じタイプの管理対象システムのみに配布されます。例えば、MySQL_Process_Down しきい値をグループに割り当てた場合、それがグループに含まれますが、配布されるのは、そのグループに属する Monitoring Agent for MySQL 管理対象システムに対してのみです。

「**選択したしきい値のみを表示**」を選択すると、割り当てられていないしきい値を非表示にすることができます。リストをフィルタリングしている 場合に、フィルターをクリアしてこのチェック・ボックスを使用できるようにするには、フィルター・ボックス の × をクリックします。

「**しきい値マネージャー**」からリソース・グループをしきい値に割り当てることもできます。

E メール通知の構成
IBM Cloud Application Performance Management で使用可能: 「**E メール通知の構成**」をクリックし、新しいブラウザー・タブまたはウィンドウで IBM Alert Notification アプリケーションを開きます。イベント通知を E メールで受信するには、Alert Notification を使用してユーザーを作成し、その E メール・アドレスをリソース・グループに関連付けます。

重要: ご使用のブラウザーがポップアップ・ウィンドウを許可しない場合、Alert Notification ウィンドウは開きません。リソース・グループの E メール通知を構成するために Alert Notification ウィンドウを開くことができるように、ブラウザーを設定する必要があります。

「**保存**」をクリックすると、リソース・グループが、リソース・グループのリストと共に保存され、「**リソース・グループ・マネージャー**」表に表示されます。

関連タスク

972 ページの『Alert Notification との統合』
1073 ページの『API の解説』

関連資料

987 ページの『しきい値マネージャー』

チュートリアル: しきい値の定義

しきい値は、管理対象リソースに関する潜在的および実際の問題について警告するメカニズムです。 このチュートリアルでは、条件が満たされたときにアラームを出すためのしきい値を定義する基本的なステップについて説明します。

このタスクについて

このチュートリアルでは、Linux OS エージェントを使用して、「**しきい値マネージャー**」でしきい値を定義する方法と、出されたアラームを Application Performance ダッシュボードに表示する方法を示します。このステップを実行するには、使用するユーザー ID に「**しきい値マネージャー**」の表示権限が必要です。

手順

1. ナビゲーション・バーで、「**システム構成**」 > 「**しきい値マネージャー**」をクリックします。
2. 「**データ・ソース・タイプ**」リスト・ボックスをクリックして **「Linux OS」** のデータ・タイプを選択します。

   Linux OS エージェントに対して定義されたしきい値が表に表示されます。
3. 「**新規**」をクリックし、しきい値を定義するために「**しきい値エディター**」を開きます。
4. 平均 CPU 使用率が 75% 未満のときに重大度が「不明」のアラームを出すようにしきい値を定義します。
   a) 「**名前**」 フィールドに CPU_average_below_75_percent と入力します。
   b) 「**説明**」フィールドに「しきい値のチュートリアル」と入力します。
   c) 「**重大度**」、「**間隔**」、および「**必要な連続サンプル数**」の各フィールドはデフォルト 値のままにしておきます。
   d) 「**データ・セット**」フィールドで **「KLZ CPU 平均 (KLZ CPU Averages)」** を選択します。
   e) 「**条件**」フィールドで「**新規**」をクリックして、ポップアップ表示されるダイアログ・ボックスに以下の比較を追加します。
      1) 「**属性**」フィールドで **CPU_Usage_Current_Average** を選択します。
      2) 「**演算子**」フィールドで **「より小」** を選択します。
      3) 「**値**」フィールドに 75 と入力します。

      「**OK**」をクリックすると、「**条件**」フィールドに属性と比較が表示されます。
   f) 「**グループ割り当て**」フィールドで **Linux OS** システム・グループを選択します。
   g) 「**保存**」をクリックして定義を完了し、「**しきい値マネージャー**」ページに戻ります。

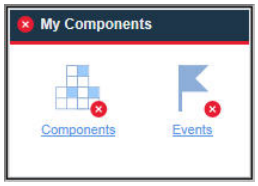
   CPU_average_below_75_percent が、Linux OS データ・ソースに対して定義されているしきい値のリストに表示されます。

タスクの結果

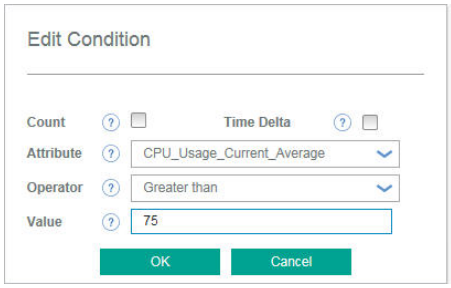
いずれかの Linux OS 管理対象システムで平均 CPU 使用率が 75% 未満であるときにアラームを出すしきい値を定義しました。

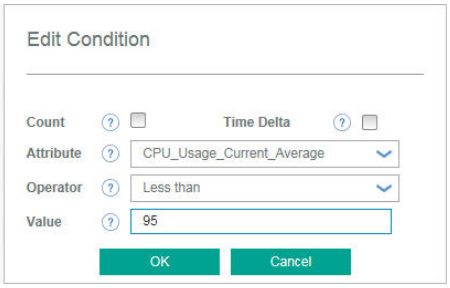
次のタスク

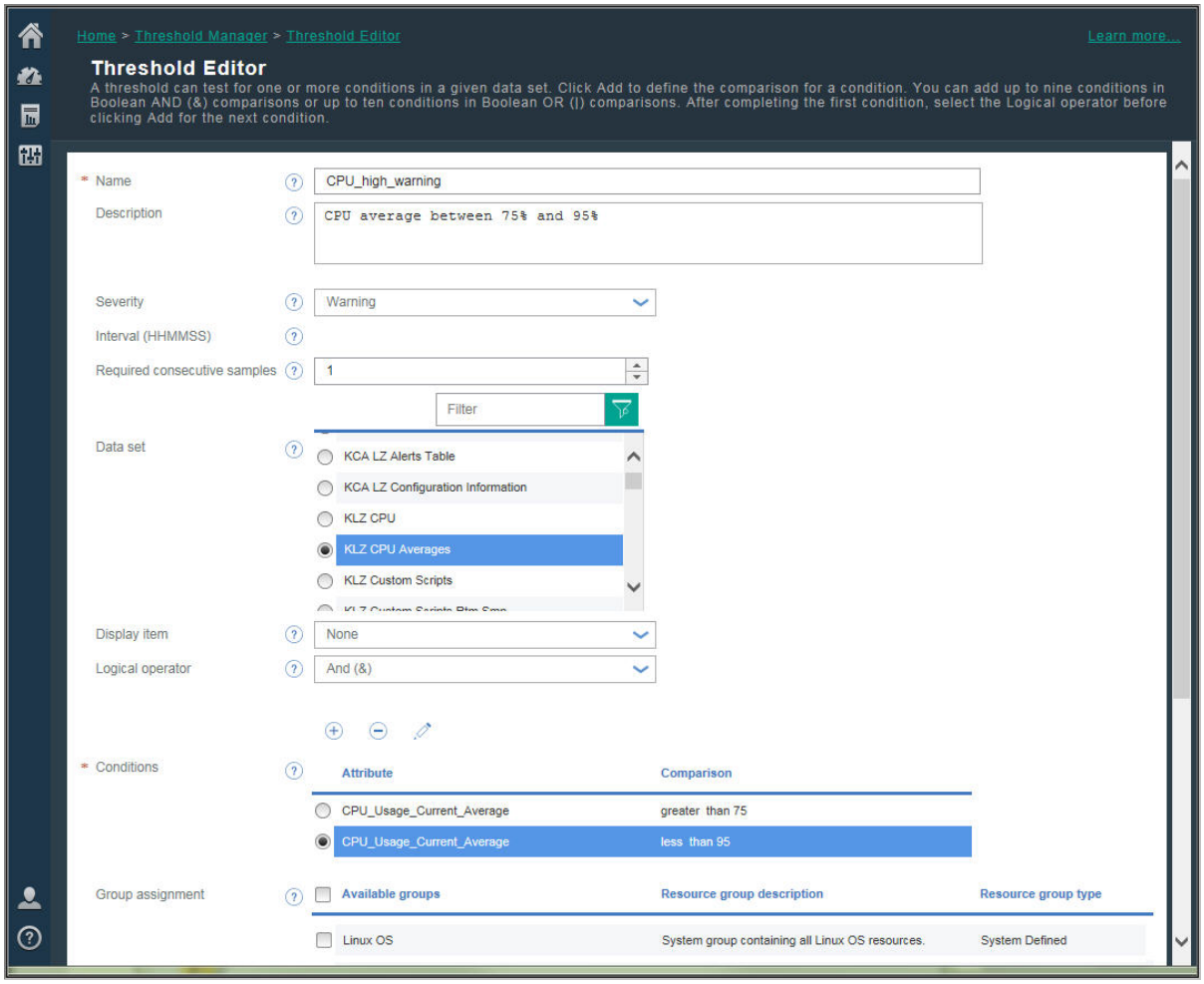
- 以下のようにして、イベントを表示します。
  1. ナビゲーション・バーで、「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックします。
  2. 「**マイ・コンポーネント**」要約ボックスで「**イベント**」リンクをクリックします。



  3. 開いた「**イベント**」タブにあるリストで CPU_average_below_75_percent というしきい値を探します。アラームが出されるまでに 1 分から 2 分かかる場合があります。ただし、CPU 平均値が 75 パーセントを超えている場合はアラームが出されません。
- 以下のようにして、しきい値を編集します。
  1. ナビゲーション・バーで、「**システム構成**」 > 「**しきい値マネージャー**」をクリックします。
  2. 「**データ・ソース・タイプ**」リスト・ボックスをクリックして「**Linux OS**」のデータ・タイプを選択します。
  3. リストから CPU_average_below_75_percent しきい値を選択し、「**編集**」をクリックします。
  4. アラートを受けとりたい条件になるように任意のフィールドを変更します。例えば、「**名前**」を CPU_high_warning に変更し、「**重大度**」を「**警告**」に変更し、条件を編集して「**75 より大**」に編集し、「**95 より小**」という条件を追加します。比較を編集するには、「**編集**」をクリックし、「**演算子**」を「**より大**」に変更して「**OK**」をクリックします。新しい条件を追加するには、「**論理演算子**」フィールドをデフォルトの「**And (&)**」のままにして、新しい条件のために「**新規**」をクリックし、「**演算子**」として「**より小**」を選択し、「**値**」として 95 と入力します。







- エージェントの事前定義しきい値を確認し、ご使用の環境での必要に応じて比較値を調整します。
- アラートを受けとりたいその他の条件でアラームを出すように新しいしきい値を作成します。

関連資料

987 ページの『しきい値マネージャー』

チュートリアル: 管理対象リソースでコマンドを実行するためのしきい値の定義

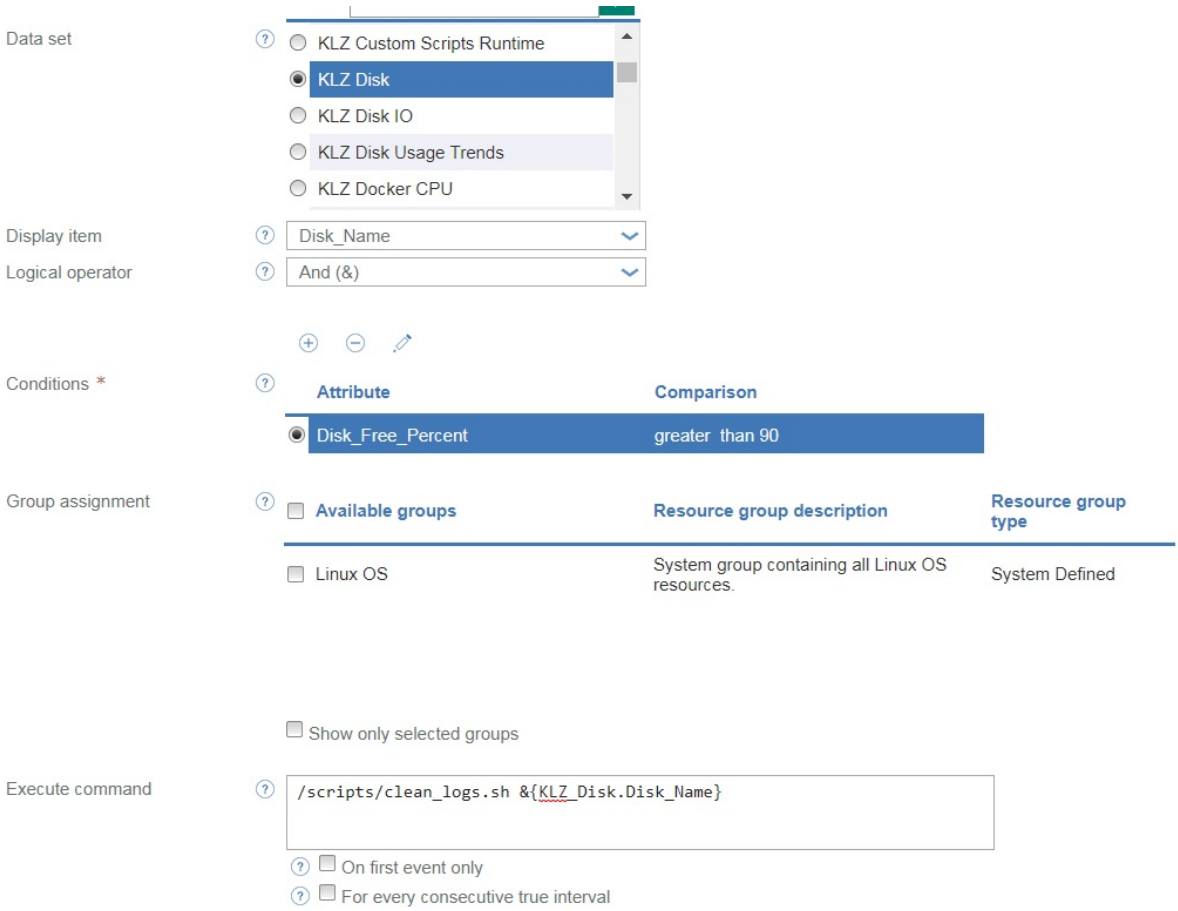
しきい値エディターを使用して、エージェントに特定のパラメーターを渡すことができます。イベントがトリガーされたときに自動的に実行するコマンドまたはコマンドのスクリプトを指定できます。

このタスクについて

このチュートリアルでは、「**実行コマンド**」フィールドを使用して、IBM Cloud Application Performance Management エージェントにパラメーターを渡す方法を示します。

手順

1. 「**システム構成**」 > 「**しきい値マネージャー**」を選択して、「**しきい値マネージャー**」を開きます。
2. 「**データ・ソース・タイプ**」フィールドで 「*Linux OS*」を選択します。
   Linux OS エージェントに対して定義されたしきい値が表に表示されます。
3. 「**新規**」をクリックし、しきい値を定義するために「**しきい値エディター**」を開きます。
4. 各種パラメーター (「**名前**」、「**重大度**」、「**条件**」など) の値を指定して、しきい値および条件を定義します。
5. 「**データ・セット**」フィールドで 「*KLZ Disk*」を選択します。



6. 「**実行コマンド**」フィールドに、以下のコマンドを入力します。

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

KLZ Disk データ・セット名のスペースを下線に置き換える必要があります。
KLZ_Disk.Disk_Name がコマンド・スクリプトに渡されます。

タスクの結果
定義されたしきい値に対して自動的に実行されるように、コマンドのスクリプトがセットアップされます。イベントがトリガーされたモニター対象システムでコマンドが実行されます。
関連資料
987 ページの『しきい値マネージャー』

しきい値マネージャー

モニター・エージェントに対する事前定義しきい値の確認や、しきい値の作成および編集を行うには、**しきい値マネージャー**を使用します。しきい値は、サンプリングした属性の値をしきい値での設定値と比較するために使用します。 サンプリングした値が比較条件を満たすと、イベントがオープンされます。しきい値の比較結果が true でなくなると、イベントは自動的にクローズされます。

「**システム構成**」 > 「**しきい値マネージャー**」をクリックすると、選択したデータ・ソース・タイプに対して定義されたしきい値の表がページに表示されます。

「**データ・ソース・タイプ**」リスト・ボックスをクリックすると表示されるデータ・タイプは、管理対象環境にインストールされているモニター・エージェントおよびデータ・コレクターのタイプです。しきい値を作成または表示するデータ・タイプを選択します。

この表には、選択したデータ・タイプに対して作成されたすべてのしきい値がリストされ、しきい値を管理するための以下のツールがあります。

- 「**新規**」をクリックすると、選択したデータ・タイプのしきい値を定義するための「**しきい値エディター**」が開きます。
- しきい値を選択し、「**編集**」をクリックして、定義を編集するために「**しきい値エディター**」を開きます。
- 不要になったしきい値を選択し、「**削除**」をクリックします。しきい値を削除することを確認すると、そのしきい値はリストおよび割り当て先のすべてのリソース・グループから削除されます。 そのしきい値に関するすべてのオープン・イベントがクローズされます。
- 長いリストの場合、フィルター・テキスト・ボックス 内をクリックして、フィルタリングの条件とする値の先頭部分を入力します。入力するにしたがい、条件に一致しない行は除外されていきます。フィルターをクリアするには、フィルター・ボックス の × をクリックするか、Backspace キーを押します。

テーブルに表示される事前定義しきい値とカスタムしきい値およびリソース・グループの割り当て (または割り当てないこと) の重要性について詳しくは、978 ページの『背景情報』を参照してください。クイック・ハンズオン・レッスンについては、984 ページの『チュートリアル: しきい値の定義』を参照してください。

しきい値エディター

「**新規**」をクリックするか、しきい値を選択して「**編集**」をクリックすると、以下のフィールドを持つしきい値エディターが表示されます。

Name
しきい値に固有の名前を付けてください。名前は、英字で開始する必要があります。名前の最大長は 31 文字 (英字、数字、および下線) です (「Average_Processor_Speed_Warning」など)。しきい値名は、Application Performance ダッシュボード 「**イベント**」タブおよび特定のダッシュボード・テーブルに表示されます。

説明
オプションです。 説明は、しきい値の目的を記録しておくのに便利です。これは、**しきい値マネージャー**で表示できます。

重大度
リストから該当するイベント重大度を選択します。「致命的」、「クリティカル」、「軽微」、「警告」、または「不明」があります。

重大度は、Application Performance ダッシュボードでの表示用に集約されます。「致命的」イベントと「重大」イベントは で示されます。「軽微」イベントと「警告」イベントは で示されます。「不明」イベントは で示されます (1109 ページの『イベント状況』を参照してください)。

EIF イベントを転送しますか?
「**システム構成**」 > 「**拡張構成**」ページ (1074 ページの『イベント・マネージャー』) でイベント転送を構成した場合、デフォルトでオープン・イベントは、構成されているイベント宛先 (例えば、EIF イベント・ターゲット、Cloud Event Management、または Alert Notification) に転送されます。このしきい値に関してイベントをイベント宛先に転送しない場合は、設定を「**いいえ**」に変更します。

「**システム構成**」 > 「**拡張構成**」ページ (イベント・マネージャー) でイベント転送を構成した場合、デフォルトでオープン・イベントは EIF 受信側に転送されます。このしきい値に関してイベントを EIF 受信側に転送しない場合は、設定を「**いいえ**」に変更します。

転送されるイベントにしきい値がどのようにマップされるかをカスタマイズし、しきい値と、イベント・サーバーに転送されるイベントの間のデフォルト・マッピングをオーバーライドするには、「**EIF スロット・カスタマイズ**」をクリックします。詳しくは、992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』を参照してください。

間隔
データ・サンプルを取得するための待機時間間隔を *HHMMSS* 形式で入力するか、選択します。例えば、00 15 00 は 15 分を意味します。サンプル・イベントしきい値の場合、最小間隔は 000030 (30 秒) で、最大間隔は 235959 (23 時間 59 分 59 秒) です。

値 000000 (ゼロ 6 個) は、ピュア・イベント しきい値を示します。ピュア・イベントは非送信請求の通知です。ピュア・イベントのしきい値にはサンプリング間隔がないため、現行値についてモニターできる定数メトリックがありません。ピュア・イベントは、24 時間後、または「**拡張構成**」ページ内の

カテゴリー「1074 ページの『イベント・マネージャー』」の「**ピュア・イベント・クローズ時間**」フィールドの設定に従ってクローズされます。

必要な連続サンプル数
イベントを生成するためには、何個のしきい値サンプルが連続して true と評価されなければならないかを指定します。しきい値に 1 を設定した場合、1 つのサンプルが true と評価されると、イベントが即時に生成されます。2 を設定した場合、イベントがオープンされるためには、2 つのしきい値サンプルが連続して true と評価されなければなりません。

データ・セット
サンプリングするデータ・タイプのデータ・セット (属性グループ) を選択します。条件に含めることができる属性は、選択されたデータ・セットの属性です。しきい値に複数の条件が設定されている場合は、 すべて同じデータ・セットに基づくものでなければなりません。

データ・セットの簡略説明を見るには、名前にマウス・ポインターを合わせます。吹き出しヘルプの「詳細」リンクをクリックすると、データ・セットおよび属性の詳細な説明が表示されます。また、ナビゲーション・バーの 「**ヘルプ**」 > 「**ヘルプ・コンテンツ**」をクリックしてモニター・エージェントのヘルプを開いたり、 「**ヘルプ**」 > 「**資料**」をクリックして参照資料をダウンロードできます。

一部のエージェントは、複数のエージェント・リソースをモニターするためのサブノードを 持つマルチノード・エージェントとして分類されます。マルチノード・エージェントには、しきい値で使用できるデータ・セットが含まれる場合がありますが、しきい値のために開かれたイベントは、Application Performance ダッシュボードには表示されません。メッセージにより、この制限が通知されます。 このようなイベントは、IBM Netcool/OMNIbus イベント・マネージャーに転送できます。

表示項目
オプションです。 複数行データ・セット専用です。ある行が評価された結果、イベントがオープンされると、そのイベントがクローズされるまでは、そのモニター対象システムでは、このしきい値について追加のイベントはオープンされません。表示項目を選択することにより、しきい値に基づいてデータ・サンプリング内の他の行を引き続き評価して、他の行が適格と判断された場合に追加のイベントをオープンできるようになります。また、表示項目は Application Performance ダッシュボードの「**イベント**」タブに表示されるため、イベントがオープンされた行を簡単に識別できます。リストには、表示項目として指定可能な属性のみが含まれます。

論理演算子
しきい値に条件が 1 つしかない場合は、このフィールドを無視してください。複数の条件を測定する場合は、以下の演算子のいずれかを選択してから、 「**新規**」をクリックして、2 つ目 (3 つ目以降も) の条件を追加してください。

And (&) 前の条件と次の条件の両方を満たした場合にしきい値に違反するとき
Or (|) いずれかの条件を満たした場合にしきい値に違反するとき

論理演算子の混用はサポートされていません。すべて And 演算子を使用するかすべて Or 演算子を使用してください。しきい値には、Or 演算子を使用する場合は最大 9 個の条件、And 演算子を使用する場合は最大 10 個の条件を指定できます。

「存在しない」関数 (後の『**演算子**』セクションで説明) を使用する場合は、式に And 演算子のみを使用できます。

条件
しきい値の定義には、同時に複数のしきい値または条件を論理的に含めることができます。

条件を追加するには、 「**新規**」をクリックします。式を変更する場合は、条件を選択して 「**編集**」をクリックし、式を削除する場合は、条件を選択して 「**削除**」をクリックします。

「**新規**」または 「**編集**」をクリックすると開く「**条件の追加**」または「**条件の編集**」ダイアログ・ボックスで、以下のフィールドに入力します。

カウント
各データ・サンプルに対して複数の行を返すデータ・セットの場合、条件の基準に一致する各行をカウント対象にできます。 イベントが開かれるのは、カウントが「**値**」に到達して、式内のその他の条件が満たされた後です。例えば、「ゾンビ」プロセスの数が 10 を超えると、アラートが出されます。

次の例では、カウントされた行数が 10 を超えると、条件が true になります。「**属性**」 タイム・スタンプ、「**演算子**」 より大、「**値**」 10

「**カウント**」チェック・ボックスを選択し、カウントする「**属性**」、関係「**演算子**」、カウントの「**値**」を選択します。

式に複数の条件がある場合は、各条件で **And** ブール演算子を使用する必要があります。「**カウント**」と「**時刻の差分**」は相互に排他的です。一方の関数のチェック・ボックスを選択すると、他方の関数は使用不可になります。この属性は、サーバー名や ORIGINNODE などのシステム ID とすることはできません。「**表示項目**」として指定することも、しきい値がピュア・イベントを開くデータ・セットから取得することもできません。

時刻の差分

条件で「**時刻の差分**」関数を使用して、サンプリングされたタイム・スタンプ (記録時刻など) を指定の時間差と比較します。

「**時刻の差分**」チェック・ボックスを選択すると、「時刻の差分」フィールドが表示され、+ (プラス) または - (マイナス) と日数、時間数、分数、秒数を組み合わせることができます。「**サンプル時刻**」または「**特定の時刻**」を比較で使用する「値」として選択します。

以下の「イベント・ログ」の例では、イベントがログに記録された時刻と、データ・サンプリングから取得したタイム・スタンプとが式によって比較されます。イベントが 7 日前に発生した場合、比較は true です。関係演算子を次のように「より小」または「等しい」に変更すると、比較は 8 日後、9 日後 (以降同様) に true となります。

「**属性**」 タイム・スタンプ
「**時刻の差分**」 - 7 日
「**演算子**」 等しい
「**値**」 入力時間

属性

この条件で比較する属性を選択します。 属性の簡略説明を見るには、リスト内の名前にマウス・ポインターを合わせます。

オペレーター

比較のタイプの関係演算子を選択します。

等しい
等しくない
より大
以上
より小
以下
正規表現が次を含む
正規表現が次を含まない

「正規表現が次を含む」および「正規表現が次を含まない」は、式に対するパターン一致を検索します。式でのストリングのマッチングが容易になればなるほど、エージェントでのワークロードがより効率的になります。式は行全体に一致する必要はありません。式に指定されたサブストリングのみが一致すれば十分です。例えば、「`See him run`」というストリングに「`him `」が含まれているかどうかを確認する場合、「`him`」を使用して正規表現を構成できますが、「`.*him.*`」を使用することもできます。あるいは、「`See`」を検索する場合、「`See`」と入力するか、または行の先頭にあることを確認するために「`^See`」と入力することができます。「`.*`」というワイルドカードを入力すると、検索の効率が向上せず、ワークロードが増えます。正規表現について詳しくは、developerWorks® 技術ライブラリーのトピックを参照するか、ブラウザーで「`regex`」を検索してください。

「存在しない」関数を選択することもできます。これは、指定したメトリックの値を、指定した値のリストと比較します。値がリスト内のどの値とも一致しない場合に、条件が TRUE になります。この機能は、システムに存在しないものがあることを通知する場合に有効です。要件および制限は以下のとおりです。

1. 選択するメトリックはテキスト属性である必要があります。時刻属性および数値属性は使用できません。
2. それぞれの値をコンマ (,) で区切ってください (例: `fred,mary,jean`)。
3. 「存在しない」条件は、しきい値で 1 つのみ使用できます。
4. 「存在しない」は、式の最後の条件にする必要あります。 他の条件が必要な場合は、「存在しない」関数を追加する前に入力し、式内では **And (&)** 演算子のみ使用してください。そうしないと、以降の行はすべて無効になります。

値

メトリックで許可される形式を使用して比較する値を入力します (例えば、20% の場合は 20、2 分の場合は 120)。

グループ割り当て

リソース・グループ内の同じタイプの管理対象システムにしきい値を配布するためにリソース・グループを割り当てます。使用可能なリソース・グループは、「変更」権限を持っているユーザー定義グループおよび「表示」権限を持っているシステム・グループ (エージェント・タイプ用) です。使用可能なシステム・グループはさらに、選択したデータ・セットに適したグループに制限されます。

グループが割り当てられていないしきい値はどのモニター対象システムにも配布されず、リソース・グループに配布されるまで停止したままになります。

`Linux OS` または `HTTP Server` などのシステム・グループは、エージェントがインストールされているすべての管理対象システムにしきい値を配布します。デフォルトでは、すべての事前定義しきい値は、そのエージェントのシステム・グループに割り当てられます。(1076 ページの『しきい値の使用可能化』の説明に従って、「**拡張構成**」ページですべての事前定義しきい値を使用不可にできます。)

例外は、IBM Tivoli Monitoring ドメインの管理対象システムです。Tivoli Monitoring ドメインの管理対象システムは、Tivoli Monitoring 環境で配布されたシチュエーションを使用してモニターする必要があります。

しきい値にグループを割り当てるには、1 つ以上のリソース・グループのチェック・ボックスを選択します。割り当てられたグループのリストが長すぎる場合は、「**選択したグループのみを表示**」を選択できます。

しきい値の割り当て先にするリソース・グループが表示されない場合でも、しきい値定義を保存できます。グループへの割り当てを行わずにしきい値を保存するかを確認するプロンプトが出されたら、「**OK**」をクリックします。その後、「**リソース・グループ・マネージャー**」で新規グループを作成し、「**リソース・グループ・エディター**」で新規グループにしきい値を割り当てることができます。詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。

実行コマンド

true に評価されるしきい値に対してイベントがオープンされた後で、そのイベントがオープンされたモニター対象システムに対してコマンドまたはコマンド・スクリプトを自動で実行することができます。例えば、情報のログ記録、ビープ音のトリガー、リソースを過剰に使用しているジョブの停止などを実行できます。

このコマンドでは、以下の構文を使用します。

```
&{data_set.attribute}
```

ここで、*data_set* はデータ・セット名であり、*attribute* はしきい値エディターに表示される属性名です。データ・セット名または属性名にスペースが含まれている場合は、下線に置き換えてください。

次の例は、ディスク名パラメーターを管理対象リソースに渡す方法を示しています。

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

データ・セットの 1 つ以上の属性を渡すことができます。複数の属性が指定されている場合、それらの属性は ($1、$2、...) の順でコマンドに渡されます。

コマンドは、エージェントの始動に使用したものと同じユーザー・アカウントを使用してコマンド行から実行します。例えば、エージェントが root として実行されている場合、root ユーザーが管理対象システムでコマンドを実行します。

以下のオプションは、コマンドの実行頻度を制御します。

データ・セットから複数の行が返された場合、データ・サンプルの最初のイベント発生時にのみコマンドを実行するには、☐「**最初のイベント時のみ**」を選択します。イベントを発生させた各行に対してコマンドを実行するには、このチェック・ボックスをクリアします。

しきい値が true に評価されるたびにコマンドを実行するには、☐「**true が連続する間は毎回**」を選択します。しきい値が true である場合はコマンドを実行するが、しきい値が false と評価された後、時間間隔経過後に再び true と評価されるまではコマンドを再実行しない場合は、このチェック・ボックスをクリアします。

「**保存**」をクリックすると、割り当てられたリソース・グループ内の同じデータ・タイプのすべてのモニター対象システムに、しきい値が適用されます。

ヒント: 「**拡張構成**」ページの「**イベント・マネージャー**」オプションを使用して、イベントの動作およびイベント転送を制御できます。1074 ページの『拡張構成』を参照してください。

注: しきい値定義に含めるのに適した属性のリストを表示するには、使用する予定のデータ・セットが設定された表を作成します。を参照してください。

関連概念

978 ページの『背景情報』
しきい値、エージェントの事前定義しきい値、事前定義しきい値の割り当て先のリソース・グループ、およびしきい値のカスタマイズについて確認するには、背景情報を参照してください。

関連タスク

984 ページの『チュートリアル: しきい値の定義』

986 ページの『チュートリアル: 管理対象リソースでコマンドを実行するためのしきい値の定義』
しきい値エディターを使用して、エージェントに特定のパラメーターを渡すことができます。イベントがトリガーされたときに自動的に実行するコマンドまたはコマンドのスクリプトを指定できます。

966 ページの『Netcool/OMNIbus との統合』
IBM Cloud Application Performance Management からオンプレミスの IBM Tivoli Netcool/OMNIbus イベント・マネージャーにイベントを転送することができます。

EIF 受信側に転送するためのイベントのカスタマイズ

Event Integration Facility (EIF) 受信側 (Netcool/OMNIbus ObjectServer、Cloud Event Management、または Alert Notification など) に送信されるしきい値イベントをカスタマイズできます。「**EIF スロット・カスタマイズ**」ウィンドウを使用して、イベント宛先に転送されるイベントの内容をカスタマイズし、デフォルトのマッピングをオーバーライドします。 Event Integration Facility 受信側に送信されるしきい値イベントのマップ定義を作成できます。「**EIF スロット・カスタマイズ**」ウィンドウを使用して、転送される EIF イベントとイベントのマッピングをカスタマイズし、デフォルトのマッピングをオーバーライドします。メッセージ・テンプレートをカスタマイズすることで、イベントによって特定された問題に関する情報と、そのイベントの特定のデータを追加できます。メッセージ・テンプレートをカスタマイズすることで、イベントによって特定された問題に関する情報を追加して、そのイベントの特定のデータを組み込むことができます。

このタスクについて

EIF 基本スロットをカスタマイズできます。これは、イベント宛先にしきい値式を送信する事前定義の **msg** スロットです。1 つ以上の EIF カスタム・スロットをイベントに追加することもできます。Netcool/OMNIbus ObjectServer を使用している場合、Netcool/OMNIbus UI にカスタム・スロットを表示するには、EIF プローブ・ルール・ファイルおよび ObjectServer トリガーを更新する必要があります。

EIF 基本スロットをカスタマイズできます。これは、EIF 受信側にしきい値式を送信する事前定義された **msg** スロットです。1 つ以上の EIF カスタム・スロットを追加することもできます。これには、EIF 受信側およびプローブ・ルール・ファイルに対する更新が必要です。

手順

現在のしきい値のイベントを転送されるイベントにマップする方法をカスタマイズするには、以下の手順を実行します。

1. 「**しきい値マネージャー**」が開いていない場合、「**システム構成**」 > 「**しきい値マネージャー**」をクリックします。
2. 「**データ・ソース・タイプ**」リスト・ボックスをクリックし、作業を行うデータ・タイプを選択します。
3. これが新規しきい値の場合は、「**新規**」をクリックし、そうでない場合はしきい値を選択して「**編集**」をクリックします。
4. このしきい値に対するイベントが転送されるイベントにどのようにマップされるかをカスタマイズするには、「**EIF フォワーダー (EIF Forwarder)**」が「はい」に設定されていることを確認し、「**EIF スロット・カスタマイズ**」をクリックして、次の手順のいずれかを実行します。
   - **EIF 基本スロット**: 基本スロットをカスタマイズするには、「**msg**」のラジオ・ボタンを選択し、「**編集**」をクリックします。
   - **EIF カスタム・スロット**: カスタム・スロットを追加するには、「**追加**」をクリックします。カスタム・スロットを編集するには、そのスロットのラジオ・ボタンを選択し、「**編集**」をクリックします。

   「**スロットの編集**」または「**スロットの追加**」ウィンドウが開きます。
5. 以下のフィールドに入力してスロットの値をカスタマイズします。

フィールド	説明	制限
スロット名	EIF カスタム・スロットの名前。先頭は文字である必要があります。	EIF 基本スロットは **msg** であり、変更できません。
スロット・タイプ	EIF カスタム・スロットのタイプ: **ストリング・タイプ**または**数値タイプ**。	EIF 基本スロットは**ストリング・タイプ**であり、変更できません。
サブタイプ	スロットに割り当てられる値。これは次のスロット・タイプに対応しています。 • **マップされた属性**を指定すると、イベントが発生した時点に選択した属性の値を追加するための「**マップされた属性**」フィールドが有効になります。 • **リテラル値**を指定すると、メッセージ・テンプレートにテキストを追加するための「**リテラル値**」フィールドが有効になります。 • **リテラル値 + マップされた属性**を指定すると、メッセージ・テンプレートにテキストと属性値を追加するための「**リテラル値**」フィールドと「**マップされた属性**」フィールドが有効になり、複数のテキストまたは属性値 (あるいはその両方) を追加するための「**追加**」ボタンが有効になります。各リテラル値または属性の後にスペースが追加されます。 EIF 基本スロット **msg** の標準的な使用法は、メッセージ・テンプレートの**リテラル値 + マップされた属性**を指定することです。	**数値タイプ**のスロットが使用できるのは、**マップされた属性**のみです。
追加	転送メッセージで複数のリテラル値または属性値を送信する場合、「**追加**」をクリックして、「**リテラル値**」フィールドと「**マップされた属性**」フィールドのセットをもう 1 つ追加します。「**追加**」を選択するたびに、これらのフィールドがパネルに追加されます。 「**リテラル値**」フィールドと「**マップされた属性**」フィールドのセットを削除するには、両方のフィールドをクリアしてから、「**OK**」をクリックします。 『例』を参照してください。	「**サブタイプ**」が**リテラル値 + マップされた属性**のときにのみ有効になります。 最大で 6 個の「**リテラル値**」フィールドと「**マップされた属性**」フィールドのセット。 追加したフィールドが表示されない場合は、ブラウザーのズームアウト機

フィールド	説明	制限
		能 (Ctrl -) を使用して、ダイアログ・ボックスに合わせてレイアウトを縮小します。
リテラル値	メッセージ・テンプレートに含めるテキスト。例えば、リテラル値 Memory utilization is high at をマップされた属性 **%Memory Utilization** とともに指定すると、**イベント・マネージャー**のユーザー・インターフェースに Memory Utilization is high at 97.3% と表示されます。 メッセージ・テンプレートは、固定されたメッセージ・テキストと、変数置換参照、または記号で構成されています。記号は、共通スロット・データまたはイベント・スロット・データ、あるいはしきい値式への特殊参照を参照します。共通スロットは、*threshold_name* など、転送されるすべてのイベントに含まれているスロットで、イベント・スロットはしきい値の msg に特有のスロットです。	「**サブタイプ**」が**マップされた属性**のときは無効です。
マップされた属性	メッセージ・テンプレートに追加する値を持つ属性。使用可能な属性は、しきい値に対して選択したデータ・セットの属性です。例えば、高いプロセッサー時間をモニターするしきい値の場合、「ユーザー時間 (%)」属性をマップできます。	最大 6 個の「**マップされた属性**」フィールド。 「**サブタイプ**」が**リテラル値**のときは無効です。 「**スロット・タイプ**」が**数値タイプ**のときに使用できるのは、数値属性のみです。
乗数	この乗数は、元のマップされた属性数値を倍数によってカスタマイズした後に定義される値です (スロット値 = *attribute1* * *n*)。例えば、EIF イベントで分を秒に変換する場合、乗数として 60 を指定します。乗数値は、0.5、5.4 などの 10 進数であらわした小数でもかまいません。	数値属性 (「**スロット・タイプ**」が**数値タイプ**) に対してのみ有効です。

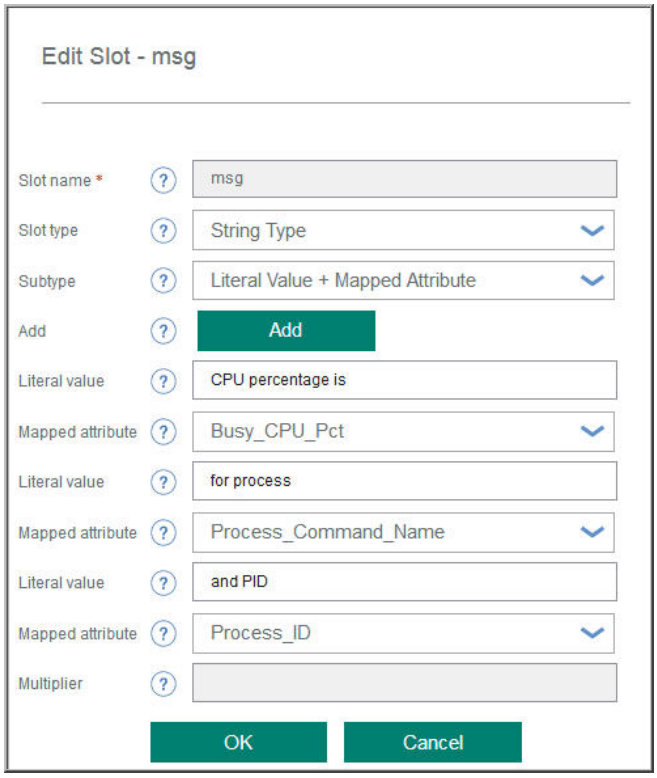
「**OK**」をクリックしてウィンドウを閉じた後、「**EIF スロット・カスタマイズ**」ウィンドウに、スロット名およびカスタマイズ済みかどうかがリストされます。

6. しきい値について EIF 基本スロットの編集または EIF カスタム・スロットの追加、削除、または編集が終了したら、「**OK**」をクリックします。
7. しきい値の編集を完了したら、「**保存**」をクリックします。

詳しくは、987 ページの『しきい値マネージャー』を参照してください。

例

Linux_BP_ProcHighCpu_Critical しきい値は、CPU 消費量が 95% 以上になっているかどうかをテストします。 要約メッセージ (msg スロットに含まれる) に、CPU 使用率、プロセス・コマンドの名前、プロセス ID を追加するために、msg スロットがカスタマイズされて、「**リテラル値**」フィールドと「**マップされた属性**」フィールドの 3 つのセットが追加されました。



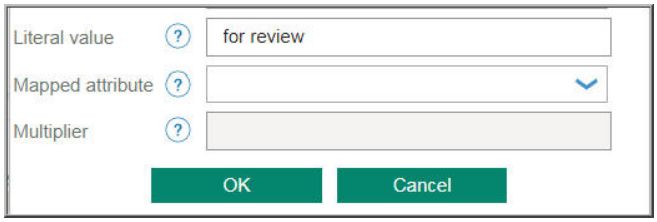
メッセージ・テンプレートは次のようになります。

```
CPU percentage is Busy_CPU_Pct for process Process_Command_Name and PID Process_ID
```

結果として、イベント・マネージャーに表示されるメッセージは、次のようになります。

```
CPU percentage is 97 for process large.exe and PID 9876
```

「**リテラル値**」フィールドと「**マップされた属性**」フィールドを追加し、1 つのフィールドを空のままにすることもできます。例えば、メッセージ・テンプレートに「for review」を追加するには、「**追加**」をクリックし、「リテラル値」に **for review** と入力します。



これで、メッセージ・テンプレートは次のようになります。

```
CPU percentage is Busy_CPU_Pct for process Process_Command_Name and PID Process_ID for review
```

結果として、**イベント・マネージャー**に表示されるメッセージは、次のようになります。

```
CPU percentage is 96 for process big.exe and PID 5432 for review
```

次のタスク

新規 EIF カスタム・スロットを作成した場合、Netcool/OMNIbus ObjectServer の `alerts.status` 表でその新規スロットを指定し、その後、Netcool/OMNIbus と Cloud APM の統合時にインストールされた `itm_apm_event.rules` 構成ファイルを更新する必要があります。

Netcool/OMNIbus ObjectServer データベースへの EIF カスタム・スロットの追加

しきい値の新しい EIF カスタム・スロットを追加する場合、EIF 受信側でそれらを指定する必要があります。その後、カスタム・スロットを使用する転送済みイベントを表示できます。Netcool/OMNIbus と Cloud APM を統合する場合、`alerts.status` 表を更新して新しいスロットを定義します。

このタスクについて

Netcool/OMNIbus と Cloud APM の統合を構成した場合、ステップ 969 ページの『4』 で、itm_apm_db_update.sql をロードしました。以下の手順では、SQL 対話式インターフェースを 使用して、itm_apm_db_update.sql データベース内の表 alerts.status を更新します。

手順

Netcool/OMNIbus ObjectServer で以下のステップを実行して、「**しきい値エディター**」で作成した新しい EIF カスタム・スロットを定義します。

1. データベースを編集するために SQL 対話式インターフェースを 始動します。

   Linux

   ```
   $OMNIHOME/bin/nco_sql -user user_name -password password
   -server server_name > itm_apm_db_update.sql
   ```

   例:

   ```
   $OMNIHOME/bin/nco_sql -user smadmin -password passw0rd -server NCOMS >
   /tmp/apm/itm_apm_db_update.sql
   ```

   Windows

   ```
   itm_apm_db_update.sql | %OMNIHOME%¥..¥bin¥isql -U user_name
   -P password -S server_name
   ```

   例:

   ```
   ¥temp¥apm¥itm_apm_db_update.sql | %OMNIHOME%¥..¥bin¥isql -U smadmin
   -P passw0rd -S NCOMS
   ```

2. EIF カスタム・スロットごとに、カスタム・スロット名とスロット・タイプを以下の形式で指定して、**ALTER TABLE** SQL コマンドを入力します。次に Enter を押し、go と入力して、Enter を押します。

   - ストリング・スロット・タイプの場合:

     ```
     alter table alerts.status add CustomSlotName varchar(512);
     ```

   - 数値スロット・タイプの場合:

     ```
     alter table alerts.status add CustomSlotName integer;
     ```

   ここで、*CustomSlotName* は EIF カスタム・スロットの名前です。この名前は、「**しきい値エディター**」の「**スロットの追加**」ウィンドウの「**slotName**」フィールドに入力した名前と正確に一致する必要があります。

例

以下の例では、**alter table** コマンドを使用して **BusinessApplication** カスタム・スロットと **GenericMetric** カスタム・スロットを追加しています。

```
alter table alerts.status add BusinessApplication varchar(512);
```

```
alter table alerts.status add GenericMetric integer;
```

次のタスク

Netcool/OMNIbus と Cloud APM の統合の一部としてインストールされた itm_apm_event.rules 構成ファイルを更新します。詳しくは、997 ページの『EIF 受信側イベント・ルールへの EIF カスタム・スロットの追加』 を参照してください。

EIF 受信側イベント・ルールへの EIF カスタム・スロットの追加

しきい値の新しい EIF カスタム・スロットを定義した場合、ルール・ファイルを更新して、EIF 受信側に対して新しいスロットを指定する必要があります。

このタスクについて

以下のステップでは、Netcool/OMNIbus Probe for Tivoli EIF の `itm_apm_event.rules` ファイルを更新して、新しい各 EIF カスタム・スロットを指定します。別の EIF 受信側を使用する場合、その受信側の要件に応じてルール・ファイルを更新します。

手順

1. Probe for Tivoli EIF がインストールされているシステムで、インストール・ディレクトリーに移動します。

   Linux cd *install_dir*/tivoli/netcool/omnibus/probes/linux2x86

   Windows cd *install_dir*¥Tivoli¥Netcool¥omnibus¥probes¥win32

   ここで、*install_dir* は、デフォルトの `/opt/IBM/` または `C:¥IBM¥`、あるいはプローブのインストール時に指定したディレクトリーです。
2. `itm_apm_event.rules` ファイルのバックアップ・コピーを作成します。
3. テキスト・エディターで Probe for Tivoli EIF `itm_apm_event.rules` ファイルを開きます。

   このファイルで、作成した 1 つまたは複数の EIF カスタム・スロットを追加するために 3 つの部分を編集します。
4. 以下の形式を使用して、各 EIF カスタム・スロットの新しいステートメントを含む **if** ステートメントを追加します。

   ```
   if(exists($CustomSlotName))
   {
       if(regmatch($CustomSlotName, "^'.*'$"))
       {
           $SourceType = extract($CustomSlotName, "^'(.*)'$")
       }
   }
   ```

   ここで、*CustomSlotName* は EIF カスタム・スロットの名前です。この名前は、「**スロットの追加**」ウィンドウの「**slotName**」フィールドに入力した名前と正確に一致する必要があります。
5. 以下の形式を使用して、各 EIF カスタム・スロットの新しい行を含む `@` 項目のリストを追加します。

   ```
   @CustomSlotName=$CustomSlotName
   ```

   ここで、*CustomSlotName* は EIF カスタム・スロットの名前です。
6. 以下の形式を使用して、各 EIF カスタム・スロットの新しい行を含む `$tmpEventData` 項目のリストを追加します。

   ```
   $tmpEventData = nvp_remove( $tmpEventData, "CustomSlotName")
   ```

   ここで、*CustomSlotName* は EIF カスタム・スロットの名前です。
7. `itm_apm_event.rules` ファイルを保存して閉じます。
8. 更新を適用するために、 Probe for Tivoli EIF を再始動します。

タスクの結果

ルール・ファイルが更新され、 Probe for Tivoli EIF は、新しい EIF カスタム・スロットを使用するしきい値イベントを処理して、イベント詳細を Netcool/OMNIbus ObjectServer に転送できるようになります。

例

以下は、EIF カスタム・スロットの **BusinessApplication** と **GenericMetric** (イタリックで示されています) を追加するために編集した itm_apm_event.rules の抜粋です。

```
# ------------------------------------------------------------
    if(exists($SourceID))
    {
        if(regmatch($SourceID, "^'.*'$"))
        {
            $SourceID = extract($SourceID, "^'(.*)'$")
        }
    }
    ...
    ...
    ...
    if(exists($ManagedSystemGroups))
    {
        if(regmatch($ManagedSystemGroups, "^'.*'$"))
        {
            $SourceType = extract($ManagedSystemGroups, "^'(.*)'$")
        }
    }
    if(exists($BusinessApplication))
    {
        if(regmatch($BusinessApplication, "^'.*'$"))
        {
            $SourceType = extract($BusinessApplication, "^'(.*)'$")
        }
    }
     if(exists($GenericMetric))
    {
        if(regmatch($GenericMetric, "^'.*'$"))
        {
            $SourceType = extract($GenericMetric, "^'(.*)'$")
        }
    }

 @SourceID=$SourceID
 @URL=$ManagementURL
 @Service=$Service
 @SourceType=$SourceType
 @SubscriberID=$TenantID
 @APMHostname=$apm_hostname
 @ManagedSystemGroups=$ManagedSystemGroups
 @BusinessApplication=$BusinessApplication
 @GenericMetric=$GenericMetric

 # ------------------------------------------------------------------
 # -  RTC 66157
 # ------------------------------------------------------------------
 if ( exists ( $appl_label ) )
 {
    if ( match($appl_label, "PI:A:S"))
    {
        @Class = 87723
    }
 }
 # ------------------------------------------------------------------
 # -  RTC 48775 - APM FP5 agents do not populate data in email of EMaaS Basic
 # ------------------------------------------------------------------
 if (match( $situation_eventdata, "~" ) )
 {
    # Dump all fields into the ITMEventData field
    $tmpEventData = nvp_add($*)
    # Remove the duplicated fields
    $tmpEventData = nvp_remove( $tmpEventData, "appl_label")
    $tmpEventData = nvp_remove( $tmpEventData, "control")
    ...
    ...
    ...
    $tmpEventData = nvp_remove( $tmpEventData, "ManagedSystemGroups")
    $tmpEventData = nvp_remove( $tmpEventData, "EventSeqNo")
    $tmpEventData = nvp_remove( $tmpEventData, "BusinessApplication")
    $tmpEventData = nvp_remove( $tmpEventData, "GenericMetric")
       @ITMEventData = $tmpEventData
```

イベントに対する応答としての E メールの送信

ご使用の管理対象環境に IBM Alert Notification が含まれている場合、アプリケーションのパフォーマンスがしきい値を超えたときに E メール通知を送信することができます。

このタスクについて

E メール通知を構成するには、IBM Knowledge Center の Alert Notification トピック・コレクションの説明に従って、IBM Alert Notification を有効にする必要があります。次に、モニター対象アプリケーションをリソース・グループに追加します。リソース・グループごとに 1 つまたは複数の E メール・アドレスを構成できます。グループのいずれかのアプリケーションのパフォーマンスがしきい 値を超えると、そのグループに対して構成されたアドレスに E メール通知が送信されます。

手順

1. 「**システム構成**」 > 「**リソース・グループ・マネージャー**」をクリックします。
2. 「**新規**」をクリックして E メール通知を構成するリソース・グループを作成するか、既存のグループを選択してから、 「**編集**」をクリックします。
   「**リソース・グループ・エディター**」が開きます。
3. 「**E メール通知の構成**」をクリックし、新しいブラウザー・タブまたはウィンドウで IBM Alert Notification アプリケーションを開きます。イベント通知を E メールで受信するには、Alert Notification を使用してユーザーを作成し、その E メール・アドレスをリソース・グループに関連付けます。

リソース・グループ管理サービス API の使用

リソース・グループ管理サービス API を使用して、コマンド行から管理対象システムのグループのライフサイクルを管理します。

このタスクについて

リソース・グループのタスク (管理対象システムのグループの 作成、表示、更新、削除など) を実行します。カスタム・グループに属する個々のシステムを追加および削除します。特定のカスタム・リソース・グループに追加したシステムのリストを表示し、組み込みグループ (システム・リソース・グループなど) に自動的に追加されたシステムのリストを表示します。

リソース・グループの定義、およびこれらのリソース・グループへのエージェントの割り当てなどのタスクを自動化するためのスクリプトを作成できます。リソース・グループは、しきい値分布やアクセス制御ポリシーのターゲットにすることができます。

以下の操作については、API Explorer およびこのトピックの最後にある『例』に説明があります。

- すべてのリソース・グループ、エージェント、または特定のリソース・グループやエージェントの取得
- カスタム・リソース・グループの作成、または既存のグループの定義の更新
- 指定したカスタム・リソース・グループの削除
- カスタム・リソース・グループへのエージェントの追加
- カスタム・リソース・グループからのエージェントの削除

手順

リソース・グループ管理サービス API でカスタム・リソース・グループを定義および変更するには、次の手順を実行します。システム・リソース・グループとシステム ・エージェントは変更できません。

1. API の解説トピックのステップ 1 からステップ 9 を実行します。
   ステップ 10 およびステップ 11 は、追加の詳細です。
2. 「**USE (使用)**」をクリックし、キーを選択します (例えば、「**Key1**」)。

   注 : 「**非表示 (Hide)**」をクリックして、client-Id と client secret を表示します。これらは API Explorer 以外の外部ツールで API 呼び出しを行う場合に必要となるため、ここでメモしておきます。次に「**表示 (Show)**」をクリックして、これらの項目を非表示にします。

3. アスタリスクで示されたすべての必須ヘッダーを入力します。

   X-IBM-Service-Location
   * ヘッダーはサブスクリプションの地理的位置 (例えば、北アメリカの場合は na) です。

   許可
   * ヘッダーは IBM ID とパスワードの base64 エンコード・ストリングです。IBM ID とパスワードを based64-encoder ツールでエンコードするときは、*IBMid:password* という形式にする必要があります (例えば、`Basic YXBtYWRtaW46YXBtcGFzcw==!`)。

4. すべての POST 要求、PUT 要求、および DELETE 要求に Referer ヘッダーを組み込む必要があります。Referer ヘッダーの値は、常に以下のようになります。

   ```
   -H 'Referer: https://api.ibm.com'
   ```

5. 「**テスト (Test)**」の位置までスクロールしてクリックします。

タスクの結果

API でカスタム・リソース・グループに対して加えた変更はすぐに有効になり、「**リソース・グループ・マネージャー**」に表示されます (982 ページの『リソース・グループ・マネージャー』を参照)。

例

次のコマンドでは、すべてのエージェントの名前、固有 ID、状況、ホスト名、バージョン、およびエージェント・タイプが返されます。

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes=Agent&_field=keyIndexName&
_field=online&_field=hostname&_field=version&_field=productCode&_field=description
```

次のコマンドでは、すべての Linux OS エージェントのリストが返されます。

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes=Agent&_filter=description=
"Linux OS"&_field=keyIndexName
```

次のコマンドでは、システム・グループとカスタム・グループのリストが返されます。

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes:AgentGroup,
AgentSystemGroup&_field=keyIndexName&_field=displayLabel
```

次のコマンドでは、固有 ID {id} を持つグループに割り当てられているエージェントのリストが返されます。

```
GET /1.0/topology/mgmt_artifacts/{id}/references/to/contains
```

以下の例では curl コマンドを使用して、カスタム・グループを作成します。

```
POST /1.0/topology/mgmt_artifacts
```

注 : POST 要求の本文には、**-d** パラメーターによって示されるグループを定義する JSON オブジェクトを含める必要があります。

```
curl -X POST ¥
  https://api.ibm.com/perfmgmt/run/1.0/topology/mgmt_artifacts ¥
  -H 'Referer: https://api.ibm.com' ¥
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' ¥
  -H 'content-type: application/json' ¥
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' ¥
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' ¥
  -d '{
  "keyIndexName": "customGroup",
  "description": "Custom group description",
  "displayLabel": "customGroupLabel",
  "entityTypes": [
    "AgentGroup"
  ],
  "arbitraryStringProperty": "Your custom property value"
}'
```

次のコマンドでは、固有 ID {otherid} のエージェントが、固有 ID {id} を持つカスタム・グループに追加されます。

```
POST /1.0/topology/mgmt_artifacts/{id}/references/to/contains/{otherid}
```

次のコマンドでは、固有 ID {otherid} のエージェントが、固有 ID {id} を持つカスタム・グループから削除されます。

```
DELETE /1.0/topology/mgmt_artifacts/{id}/references/to/contains/{otherid}
```

しきい値管理サービス API の使用

しきい値管理サービス API を使用して、コマンド行からモニターのしきい値のライフサイクルを 管理します。

このタスクについて

しきい値マネージャーのタスク (しきい値の作成、表示、更新、削除など) を実行します。リソース・グループをこれらのしきい値に割り当てます。すべてのしきい値およびリソース割り当てのリストを表示します。特定のリソース・グループに割り当てられたすべてのしきい値のリストを表示します。

しきい値の定義、およびリソース・グループへのこれらのしきい値の割り当てなどのタスクを自動化するためのスクリプトを作成できます。

以下の操作については、API Explorer およびこのトピックの最後にある『例』に説明があります。

- すべてのしきい値の取得、または特定のしきい値の取得。次の属性を使用して、要求をフィルターに掛けることができます。**label** (しきい値名に対応)、**_appliesToAgentType** (2 文字の製品コードに対応)、および **_uiThresholdType** (Cloud APM コンソールの「しきい値マネージャー」ページと「リソース・グループ・エディター」ページに対応)。しきい値の取得時には **_offset** または **_limit** を使用できます。
- しきい値の作成、または既存のしきい値の定義の更新。更新要求の場合は X-HTTP-Method-Override ヘッダーを組み込んで、PATCH に設定する必要があります。
- 指定したしきい値の削除
- すべてのリソース割り当てまたは特定のリソース割り当ての取得。これにより、各リソース・グループに割り当てられているしきい値が表示されます。属性 **resource._id** と **threshold._id** を使用し、サポートされる演算子 **=** (等しい) と **!=** (等しくない) を使用して、要求をフィルターに掛けることができます。
- リソース割り当ての作成。これにより、単一しきい値が単一リソース・グループに割り当てられます。
- リソース割り当ての削除。これにより、単一しきい値が単一リソース・グループから削除されます。

手順

1. API の解説トピックのステップ 1 からステップ 9 を実行します。
   ステップ 10 およびステップ 11 は、追加の詳細です。
2. 「**USE (使用)**」をクリックし、キーを選択します (例えば、「**Key1**」)。

   注 : 「**非表示 (Hide)**」をクリックして、client-Id と client secret を表示します。これらは API Explorer 以外の外部ツールで API 呼び出しを行う場合に必要となるため、ここでメモしておきます。次に「**表示 (Show)**」をクリックして、これらの項目を非表示にします。
3. アスタリスクで示されたすべての必須ヘッダーを入力します。

   X-IBM-Service-Location
   * ヘッダーはサーバーの地理的位置 (例えば、北アメリカの場合は na) です。

   許可
   * ヘッダーは IBMid とパスワードの base64 エンコード・ストリングです。IBMid とパスワードを based64-encoder ツールでエンコードするときは、*IBMid:password* という形式にする必要があります。例えば、`Basic YXBtYWRtaW46YXBtcGFzcw==!` です。

4.「**テスト (Test)**」の位置までスクロールしてクリックします。

例

次のコマンドでは、サーバーに登録されているすべてのしきい値が返されます。

```
GET /threshold_types/itm_private_situation/thresholds
```

次のコマンドでは、ラベル (名前) My_threshold を持つしきい値の情報が返されます。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=label%3DMy_threshold
```

次のコマンドでは、Linux OS エージェントのコンポーネント・コードであるエージェント・タイプ LZ のしきい値がすべて返されます。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=_appliesToAgentType%3DLZ
```

次のコマンドの出力は、前のコマンドと同じですが、Cloud APM コンソールに表示されるエージェント名が指定されています。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=_uiThresholdType%3DLinux OS
```

次のコマンドでは、しきい値 123 が割り当てられているすべてのリソース・グループが返されます。

```
GET /resource_assignments?_filter=threshold._id=123
```

以下の例では curl コマンドを使用して、新しいしきい値を作成します。

```
POST /1.0/thresholdmgmt/threshold_types/itm_private_situation/thresholds
```

要確認 : POST 要求の本文には、**-d** パラメーターによって示されるしきい値を定義する JSON オブジェクトを含める必要があります。 例:

```
curl -X POST¥
 https://api.ibm.com/perfmgmt/run/1.0/thresholdmgmt/threshold_types/itm_private_situation/
thresholds  ¥
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' ¥
  -H 'content-type: application/json' ¥
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' ¥
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' ¥
  -d '{
  "label": "Your_Linux_Threshold_Name",
  "description": "Your Linux Threshold Definition",
  "configuration": {
    "type": "json",
    "payload": {
      "formulaElements": [
        {
          "function": "*MKTIME",
          "metricName": "KLZ_CPU.Timestamp",
          "operator": "*EQ",
          "threshold": "1455767100000",
          "timeDelta": {
            "operator": "+",
            "delta": "3",
            "unit": "Hours"
          }
        }
      ],
      "period": "011500",
      "periods": "3",
      "severity": "Fatal",
      "matchBy": "KLZCPU.CPUID",
      "operator": "*OR",
      "actions": [
        {
          "name": "command",
          "command": "ps -ef",
          "commandFrequency": "Y",
          "commandWhen": "Y",
          "commandWhere": "N"
```

```
        }
      ]
    }
  }
}'
```

ユーザー・アクセスの管理

Cloud APM の役割ベースのアクセス制御機能を使用して、ユーザーの役割に必要なアクセス権をユーザーに付与します。

Cloud APM のセキュリティーは、役割に基づいています。役割とは、ユーザーが Cloud APM で実行できるアクションを制御する複数の権限から成るグループです。Cloud APM では、カスタマイズされた役割を作成できます。カスタマイズした役割に権限を割り当てたり、 既存のデフォルトの役割に追加の権限を割り当てたりすることができます。既存のデフォルトの役割またはカスタマイズされた役割にユーザーとユーザー・グループを割り当てることができます。複数の役割にユーザーとユーザー・グループを割り当てることができます。許可は累積されます。ユーザーまたはユーザー・グループには、割り当てられているすべての役割のすべての許可が割り当てられます。

Cloud APM のユーザー認証は、IBM Marketplace で管理されます。あるいは、Collaborative Operations のサブスクリプションがある場合は、Collaborative Operations で管理できます。

Performance Management でのユーザー認証には IBM ID が必要です。IBM ID は、「**Sign in to IBM**」ページの「**Create an IBMid**」リンクを選択することで作成します。「**Sign in to IBM**」ページにアクセスするには、IBM Marketplace の「**Products and Services**」 (http://ibm.biz/my-prodsvcs) ページに移動すると、「**Sign in to IBM**」ページにリダイレクトされます (ログアウトしてから「**Products and Services**」 (http://ibm.biz/my-prodsvcs) ページに戻ることが必要な場合があります)。

IBMid を取得したら、直接または「Products and Services」経由で Cloud APM にログインできます。Cloud APM のインスタンスへのアクセス許可は、「**Products and Services**」または Collaborative Operations (サブスクリプションがある場合) 経由で管理されます。デフォルトでは、試用版を要求したか、サブスクリプションを購入したユーザーに管理者特権が付与されます。

Cloud APM サブスクリプションの所有者が、Cloud APM のデフォルト・ユーザーになります。 このユーザーが、試用版を要求したか、サブスクリプションを購入したユーザーです。このデフォルトのユーザーは、役割管理者の役割のメンバーであり、このユーザーによる新しいユーザーの追加を許可する管理者特権を持っています。後続ユーザーは、デフォルトでは、モニター・ユーザー役割に追加されます。

役割のメンバー以外のユーザーが Cloud APM へのログインを試行すると、「**権限なし**」というメッセージが表示されます。

Cloud APM ユーザーを追加するには、以下のようにします。

1. IBM Marketplace の「**Products and Services**」に移動し、「**IBM Performance Management**」ウィジェットを展開します。
2. 「**権限の管理 (Manage Authorizations)**」をクリックし、IBM ID または E メール・アドレスを「**新規ユーザーの追加 (Add new user)**」または「**既存ユーザーの検索 (Search existing users)**」フィールドに入力します。「**ユーザーの追加**」をクリックしてユーザーを追加します。

Cloud APM サブスクリプションに追加できるユーザーの数に制限はありません。

注 : Cloud APM のユーザー・グループに対するサポートは提供されていません。

役割について詳しくは、1003 ページの『役割と権限』を参照してください。

役割と権限

役割とは、ユーザーが Cloud APM で実行できるアクションを制御する複数の権限から成るグループです。「役割ベースのアクセス制御」ページを使用してユーザーと役割を管理するか、許可 API を使用してコマンド行から役割ベースのアクセス制御タスクを実行します。詳しくは、1073 ページの『API の解説』を参照してください。

Cloud APM には、以下の 4 つのデフォルトの役割があります。

役割管理者
この役割は、Cloud APM のアクセス制御ポリシーを作成することを主な職務とするユーザーを対象にしています。 この役割には、すべての権限が備わっています。デフォルト・ユーザーを変更すると、新しいデフォルト・ユーザーが自動的に役割管理者の役割のメンバーになります。この役割は編集できません。役割管理者が、自分自身を役割管理者の役割から削除することはできません。この制限によって、役割管理者の役割から誤ってすべてのユーザーを削除してしまうリスクが排除されます。

モニター管理者
この役割は、Cloud APM を使用してシステムをモニターすることを 主な職務とするユーザーを対象にしています。モニター管理者は、 モニター・アプリケーションの追加、しきい値の作成、リソース・グループの追加、 それらのリソース・グループへのしきい値の配布などのタスクを実行します。この役割は編集可能です。

システム管理者
この役割は、Cloud APM システムの管理タスクを実行することを主な職務とするユーザーを対象にしています。システム 管理者は、イベント・マネージャーの構成やハイブリッド・ゲートウェイの構成などのタスクを実行します。この役割は編集可能です。

モニター・ユーザー
この役割は、Cloud APM によってモニターされるシステムの正常性と状態を構成および保守することを主な職務とするユーザーを対象にしています。この役割は、編集可能です。

以下の表では、役割に割り当てることができる権限、および利用できる 4 種類のデフォルトの役割とそれに関連付けられた権限を示します。

表 *237.* 役割と権限

	役割管理者		モニター管理者		システム 管理者		モニター・ユーザー	
	表示	変更	表示	変更	表示	変更	表示	変更
システム構成権限								
拡張構成	✓	N/A	–	N/A	✓	N/A	–	N/A
エージェント構成	✓	N/A	✓	N/A	–	N/A	–	N/A
情報ページ	✓	N/A	✓	N/A	✓	N/A	✓	N/A
検索プロバイダー	✓	N/A	✓	N/A	–	N/A	–	N/A
使用統計	✓	N/A	✓	N/A	–	N/A	–	N/A
リソース権限								
Application Performance ダッシュボード	✓	✓	✓	✓	✓	–	✓	–
アプリケーション	✓	✓	✓	✓	–	–	✓	–
個別のアプリケーション	1008 ページの『アプリケーションとリソース・グループの権限』							
診断ダッシュボード	✓	N/A	–	N/A	–	N/A	–	N/A
リソース・グループ・マネージャー	✓	N/A	✓	N/A	–	N/A	–	N/A
個別のリソース・グループ	1008 ページの『アプリケーションとリソース・グループの権限』							
リソース・グループ	✓	✓	✓	✓	–	–	–	–
Synthetic Script Manager	✓	N/A	–	N/A	–	N/A	–	N/A
しきい値マネージャー	✓	N/A	✓	N/A	–	N/A	–	N/A

表の見方
✓ は、この役割のメンバーがこの権限を持つことを示します

は、この役割のメンバーがこの権限を持たないことを示します

N/A はこの権限が存在しないことを示します

注 : 使用統計は**システム構成権限**のリストに表示されますが、Cloud APM には適用されなくなりました。

以下の表では、各権限に関連付けられたアクションについて説明します。

表 *238.* 許可

権限	説明
拡張構成	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、「**システム構成**」 > 「**拡張構成**」を表示する。 •「**拡張構成**」ウィンドウで変更を行い、変更内容を保存する。 • メニュー・バーで「**システム構成**」 > 「**ハイブリッド・ゲートウェイ・マネージャー**」を表示する。 •「**ハイブリッド・ゲートウェイ・マネージャー**」ウィンドウで変更を行い、変更内容を保存する。
エージェント構成	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、「**システム構成**」 > 「**エージェント構成**」を表示する。 •「**エージェント構成**」ウィンドウで変更を行い、変更内容を保存する。
情報ページ	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで「**はじめに**」および「**ヘルプ**」を表示する。 **注 :**「**はじめに**」ページを開いたときに、「**始動時にこの「はじめに」ページを表示する**」のチェック・マークを外すと、それ以降のログインでは権限が拒否されたことを通知するメッセージが表示されます。ただし、その場合でも、「**はじめに**」ページ、および権限を持っている他のすべての領域にナビゲートできます。
検索プロバイダー	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、「**システム構成**」 > 「**検索プロバイダーの構成**」を表示する。 •「**検索プロバイダーの構成**」ページで変更を行い、その変更内容を保存する。

表 *238.* 許可 *(*続き*)*

権限	説明
Application Performance ダッシュボード	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」を表示する。 • Application Performance ダッシュボードと、「**マイ・コンポーネント**」および「**マイ・トランザクション**」の事前定義アプリケーションを表示する。 **注 :**「マイ・コンポーネント」アプリケーションでシステムを 表示するために必要な権限を判別するには、1008 ページの『アプリケーションとリソース・グループの権限』を参照してください。 **注 :**「マイ・トランザクション」アプリケーションが表示されるのは、Web Site Monitoring を使用している場合のみです。Web Site Monitoring のシンセティック・トランザクションはすべて「マイ・トランザクション」アプリケーションで表示されます。 •「カスタム・ビュー」タブでカスタム・ダッシュボード・ページを開く。 •「属性の詳細」タブでビューを作成し、自分で使用するために保存する。 変更権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」を表示する。 • Application Performance ダッシュボード と、「**マイ・コンポーネント**」および「**マイ・トランザクション**」の事前定義アプリケーションを表示する。 **注 :**「マイ・コンポーネント」アプリケーションでシステムを 表示するために必要な権限を判別するには、1008 ページの『アプリケーションとリソース・グループの権限』を参照してください。 **注 :**「マイ・トランザクション」アプリケーションが表示されるのは、Web Site Monitoring を使用している場合のみです。Web Site Monitoring のシンセティック・トランザクションはすべて「マイ・トランザクション」アプリケーションで表示されます。 •「カスタム・ビュー」タブでカスタム・ダッシュボード・ページを作成および保存する。 •「属性の詳細」タブでビューを作成し、他のユーザーと共有する。 • コンポーネント・ページで「**アクション**」>「**編集**」オプションを表示する。このオプションを使用すると、コンポーネント・ダッシュボードに表示されるグループ・ウィジェットのしきい値その他の設定を編集できます。
アプリケーション	表示権限がある場合は、以下のタスクを実行できます。 • アプリケーション・ダッシュボードでアプリケーションを表示する。 変更権限がある場合は、以下のタスクを実行できます。 • アプリケーション・ダッシュボードでアプリケーションを表示する。 • アプリケーション・ダッシュボードでツール を使用してアプリケーションの作成、変更、削除を行う。
個別のアプリケーション	1008 ページの『アプリケーションとリソース・グループの権限』を参照してください。
リソース・グループ・マネージャー	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで 「**システム構成**」 > 「**リソース・グループ・マネージャー**」を表示する。

表 *238.* 許可 *(*続き*)*	
権限	**説明**
リソース・グループ	表示権限がある場合は、以下のタスクを実行できます。 • リソース・グループ・マネージャーの表示権限もある場合は、リソース・グループ・マネージャーでリソース・グループおよびそのシステムを 表示する。 • Application Performance Management の表示権限または変更権限もある場合は、「マイ・コンポーネント」事前定義アプリケーションでシステムを 表示する。 • アプリケーションを変更する権限もある場合は、「**アプリケーションの追加**」ウィンドウでシステムを 表示する。 変更権限がある場合は、以下のタスクを実行できます。 • リソース・グループ・マネージャーの表示権限もある場合は、リソース・グループ・マネージャーでリソース・グループおよびその内容を表示する。 • Application Performance Management の表示権限または変更権限もある場合は、「マイ・コンポーネント」事前定義アプリケーションでシステムを 表示する。 • アプリケーションを変更する権限もある場合は、「**アプリケーションの追加**」ウィンドウでシステムを 表示する。 • リソース・グループ・マネージャーの表示権限もある場合は、リソース・グループ・マネージャーでリソース・グループの作成、変更、削除を行う。リソース・グループにしきい値を割り当てるには、しきい値マネージャーに対する表示権限を持つ役割のメンバーであることも必要です。 **注 :** リソース・グループ・マネージャーを使用すると、モニター対象システムをグループに編成できるため、それらのグループにしきい値を割り当てることができるようになります。しきい値マネージャーに対する表示権限がない場合、リソース・グループに割り当てられたしきい値を表示できません。ある役割にリソース・グループの変更権限を割り当てた場合は、その役割に対してしきい値マネージャーの表示権限も割り当てる必要があります。
個別のリソース・グループ	1008 ページの『アプリケーションとリソース・グループの権限』を参照してください。
しきい値マネージャー	表示権限がある場合は、以下のタスクを実行できます。 • メニュー・バーで、 「**システム構成**」 > 「**しきい値マネージャー**」を表示する。 • しきい値マネージャーでしきい値を作成、変更、削除する。 • リソース・グループに対する適切な権限を保持している場合は、しきい値マネージャーでしきい値に対するリソース・グループの割り当てを表示、編集する。 • 別の方法として、リソース・グループ・マネージャーとリソース・グループに対する適切な権限を保持している場合は、リソース・グループ・マネージャーでリソース・グループに対するしきい値の割り当てを表示、編集し、しきい値マネージャーに対する権限を表示する。

表 238. 許可 (続き)	
権限	**説明**
Synthetic Script Manager	表示権限がある場合は、以下のタスクを実行できます。 • Synthetic Transaction Manager でシンセティック・トランザクションを作成、変更、削除する。 **注 :** Synthetic Transaction Manager でシンセティック・トランザクションを処理するには、「**エージェント構成**」の表示権限がある役割のメンバーでもある必要があります。
診断ダッシュボード	表示権限がある場合は、WebSphere Applications エージェント、Node.js エージェント、Ruby エージェント、および Microsoft .NET エージェントの診断ダッシュボードで「**診断**」ボタンが使用可能になります。「**診断**」ボタンをクリックすると、診断ダッシュボードにドリルダウンできます。

アプリケーションとリソース・グループの権限
個々のアプリケーションおよびリソース・グループに権限を割り当てることができます。

アプリケーション権限

Cloud APM では、アプリケーションは、コンポーネントと、それらのコンポーネント内のインスタンスのグループです。「**アプリケーションの追加**」ウィンドウを使用して、アプリケーションを定義します。アプリケーションの定義方法について詳しくは、アプリケーションの管理を参照してください。

Cloud APM コンソールで「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」を選択するには、Application Performance ダッシュボードの表示権限または変更権限が割り当てられている必要があります。この権限により、「**マイ・コンポーネント**」および「**マイ・トランザクション**」の事前定義アプリケーションを表示することもできます。「**マイ・トランザクション**」アプリケーションが表示されるのは、Web Site Monitoring を使用している場合のみです。その他のカスタム・アプリケーションを表示するには、すべてのアプリケーションまたは個々のアプリケーションの表示権限または変更権限が必要です。

注 : アプリケーションが名前変更される場合、権限は保持されません。表示権限および変更権限を再割り当てする必要があります。

表示

アプリケーションの表示権限は、他の権限よりも優先されます。アプリケーションを表示するには、そのアプリケーション内の各コンポーネントおよび各コンポーネント・インスタンスに対する表示権限を持つ役割のメンバーである必要はありません。以下の表で、アプリケーションの表示権限を持っている場合に実行できるアクションについて説明します。

表 239. アプリケーションの表示権限	
アクション	**権限で実行可能**
そのアプリケーション内のすべてのサポート・コンポーネントを表示する。	✓
ナビゲーション・ツリーにアプリケーションおよびそのコンポーネントを表示する。	✓
「マイ・コンポーネント」にアプリケーション・コンポーネントを表示する。	✓
アプリケーションに関連付けられている、カスタマイズされたダッシュボード・ページを表示する。	✓
アプリケーションのコンポーネントを追加または削除する。	–

表 239. アプリケーションの表示権限 (続き)	
アクション	**権限で実行可能**
アプリケーションのコンポーネントにしきい値を割り当てる。	—
リソース・グループ・マネージャーでアプリケーションのサポート・コンポーネントを表示する。	—

変更

個別のアプリケーションの変更権限を持つ役割のメンバーは、次の操作を実行できます。

- アプリケーションを削除する。
- 「カスタム・ビュー」タブで、カスタマイズされたダッシュボード・ページを作成する。1112 ページの『カスタム・ビュー』を参照してください。
- 「**アプリケーションの編集**」ウィンドウを使用して、コンポーネントおよびコンポーネント・インスタンスを追加または削除する。「**アプリケーションの編集**」ウィンドウに表示されるコンポーネントおよびコンポーネント・インスタンスは、ユーザーの役割の権限に基づいてフィルタリングされます。以下のコンポーネントが使用可能になります。
  - システム・リソース・グループおよびカスタム・リソース・グループ内の、ユーザーが直接アクセスする権限を持っているコンポーネント
  - 変更権限を持つ他のアプリケーションに基づいて権限を間接的に継承したコンポーネント

リソース・グループ権限

リソース・グループを使用して、コンポーネントをそのタイプまたは目的に基づいてグループ化します。リソース・グループの作成方法の詳細については、982 ページの『リソース・グループ・マネージャー』を参照してください。

「**システム構成 (System Configuration)**」 > 「**リソース・グループ・マネージャー**」を選択するには、リソース・グループ・マネージャーの表示権限が割り当てられている必要があります。「**リソース・グループ・マネージャー**」でリソース・グループを表示したり、「**マイ・コンポーネント**」アプリケーションでリソース・グループ・メンバーを表示したりするには、すべてのリソース・グループまたは個々のリソース・グループの表示権限または変更権限も割り当てられている必要があります。

リソース・グループには、カスタム・リソース・グループとシステム ・リソース・グループの 2 種類があります。

カスタム定義のリソース・グループ

カスタム・リソース・グループは、リソース・グループ・マネージャーで作成します。カスタム・リソース・グループを使用して、リソースをその目的に基づいてグループ化します。

以下の表で、カスタム・リソース・グループの表示権限を持っている場合に実行できるアクションについて説明します。

表 240. カスタム・リソース・グループの表示権限	
アクション	**権限で実行可能**
「リソース・グループ・マネージャー」でカスタム・リソース・グループとその中のリソースを表示する。	✔
アプリケーションの変更権限もある場合は、「**アプリケーションの追加**」ウィンドウでカスタム・リソース・グループに属しているリソースを表示する。	✔

表 240. カスタム・リソース・グループの表示権限 (続き)	
アクション	**権限で実行可能**
Application Performance ダッシュボードのいずれかの権限もある場合は、「**マイ・コンポーネント**」事前定義アプリケーションでカスタム・リソース・グループに属しているリソースを表示する。	✔
カスタム・リソース・グループにリソースを追加する。	–
カスタム・リソース・グループからリソースを削除する。	–

以下の表で、カスタム・リソース・グループの変更権限を持っている場合に実行できるアクションについて説明します。

表 241. カスタム・リソース・グループの変更権限	
アクション	**権限で実行可能**
「しきい値エディター」でカスタム・リソース・グループにしきい値を割り当てる。 **注 :** しきい値を割り当てるには、しきい値エディターに対する表示権限を持つ役割のメンバーであることも必要です。	✔
カスタム・リソース・グループにリソースを追加する。	✔
カスタム・リソース・グループからリソースを削除する。	✔

システム・リソース・グループ

システム・リソース・グループは、Cloud APM 環境のセットアップの一環として自動的に定義されます。手動でのシステム・リソース・グループの作成、削除、カスタマイズはできません。システム・リソース・グループには表示権限のみを使用でき、変更権限は使用できません。

システム・リソース・グループは、そのリソースが Cloud APM サーバーに認識された時点で、リソース・タイプごとに定義されます。システム・リソース・グループは、Cloud APM サーバーに接続されているリソース・タイプごとに存在します。

リソースの例として、Cloud APM エージェントがあります。例えば、Db2 エージェントを初めてダウンロードし、インストールして開始したときに、Db2 という名前のシステム・リソース・グループが作成されます。このグループに、その後 Performance Management 環境に追加されるすべての Db2 エージェントが含まれます。

各リソース・タイプのシステム・リソース・グループには、IBM Tivoli Monitoring のリソースを含む、そのタイプのすべてのリソースが含まれます。 IBM Tivoli Monitoring と IBM Cloud Application Performance Management の両方を含む環境では、IBM Cloud Application Performance Management ハイブリッド・ゲートウェイをインストールすることで、両方のドメインからのエージェントのビューを利用することができます。システム定義のリソース・グループには、両方のドメインのエージェントが含まれます。詳しくは、949 ページの『IBM Tivoli Monitoring V6.3 との統合』を参照してください。

一部のシステム・リソース・グループは、サブノード・エージェントに基づいています。サブノード・エージェントに基づいたシステム・リソース・グループにしきい値を割り当てることはできますが、Application Performance ダッシュボードにはイベントは表示されません。しきい値は、イベント転送のためにサブノード・エージェントに基づくシステム・リソース・グループに割り当てられます。サブノード・エージェントに基づくシステム・リソース・グループについては、リソース・グループ・マネージャーに「このグループのメンバーはアプリケーションに 追加できず、Performance Management コ

ンソールにイベントは表示されません」という説明があります。詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。

以下の表で、システム・リソース・グループの表示権限を持っている場合に実行できるアクションについて説明します。

表 242. システム・リソース・グループの表示権限

アクション	権限で実行可能
「リソース・グループ・マネージャー」でシステム・リソース・グループを表示する。	✓
アプリケーションの変更権限もある場合は、「**アプリケーションの追加**」ウィンドウでシステム・リソース・グループに属しているリソースを表示する。	✓
Application Performance ダッシュボードのいずれかの権限もある場合は、「**マイ・コンポーネント**」事前定義アプリケーションでシステム・リソース・グループに属しているリソースを表示する。	✓
「しきい値エディター」でシステム・リソース・グループにしきい値を割り当てる。	✓
システム・リソース・グループにリソースを追加する。	—
システム・リソース・グループからリソースを削除する。	—

役割、ユーザー、および権限の操作
役割、ユーザー、および権限を操作するには、「役割ベースのアクセス制御」ページを使用します。

始める前に
別の方法として、許可 API を使用してコマンド行から「役割ベースのアクセス制御」タスクを実行します。詳しくは、以下を参照してください。 1073 ページの『API の解説』.

注 : Cloud APM ではユーザー・グループはサポートされていません。

手順

- 「役割ベースのアクセス制御」ページに表示されている役割、ユーザー、またはユーザー・グループのリストをフィルターに掛けるには、以下の手順を実行します。
  a) 「**システム構成**」>「**役割ベースのアクセス制御**」を選択します。
  b) 「**フィルター**」テキスト・ボックスの中をクリックし、 フィルタリングの条件とするテキストの一部または全体を入力します。
  入力すると、フィルター・ボックスに入力したテキストを含まない行が表から削除されます。
  c) クイック・フィルターを解除するには、値を削除するか「x」をクリックします。
  d) フィルターを適用するには をクリックします。
- 新規のカスタマイズされた役割を作成するには、以下の手順を実行します。
  a) 「**システム構成**」>「**役割ベースのアクセス制御**」を選択します。
  b)「**役割**」タブで、 をクリックします。「**役割エディター**」ページが表示されます。
  c)「**ユーザーの役割への割り当て**」タブで、 「**ユーザー・グループ**」タブまたは「**個別ユーザー**」タブを選択し、 役割に追加するユーザーおよびユーザー・グループを選択します。
  d)「**権限の役割への割り当て**」タブで、 「**システム構成権限**」タブまたは「**リソース権限**」タブを選択し、 役割に割り当てる権限を選択します。

e)「**保存**」をクリックします。

- 既存のデフォルト役割またはカスタマイズした役割を編集するには、以下の手順を実行します。
  a) 「**システム構成**」>「**役割ベースのアクセス制御**」を選択します。
  b)「**役割**」タブで、 をクリックします。「**役割エディター**」ページが表示されます。
  c)「**ユーザーの役割への割り当て**」タブで、 「**ユーザー・グループ**」タブまたは「**個別ユーザー**」タブをクリックし、 役割に追加するユーザーまたはユーザー・グループを選択します。
  d)「**権限の役割への割り当て**」タブで、 「**システム構成権限**」タブまたは「**リソース権限**」タブを選択し、 役割に割り当てる権限を選択します。
  e)「**保存**」をクリックします。
- 役割を削除するには、以下の手順を実行します。
  a) 「**システム構成**」>「**役割ベースのアクセス制御**」を選択します。
  b)「**役割**」タブで、削除する役割を選択し、 をクリックします。確認メッセージが表示されたら、「**OK**」をクリックします。

  注 : 役割を削除しても、 その役割のメンバーになっているユーザーは削除されません。引き続き「**個別ユーザー**」タブで使用可能であり、 役割管理者が別の役割に割り当てることができます。
- 個々のユーザーまたはユーザー・グループの権限を編集するには、以下の手順を実行します。
  a) 「**システム構成**」>「**役割ベースのアクセス制御**」を選択します。
  b)「**個別ユーザー**」タブまたは「**ユーザー・グループ**」タブで、 編集するユーザーまたはユーザー・グループを選択し、 をクリックします。「**個別ユーザー・エディター**」ページが開きます。
  c) ユーザーに割り当てる役割 (複数可) を選択します。
  d)「**保存**」をクリックします。
- ユーザーまたはユーザー・グループの権限を要約した csv ファイルを作成するには、以下の手順を実行します。
  a)「**個別ユーザー**」タブまたは「**ユーザー・グループ**」タブで、 必要なユーザーまたはユーザー・グループを選択し、 をクリックします。「**個別ユーザー・エディター**」ページまたは 「**ユーザー・グループ・エディター**」ページが開きます。
  b)「**要約のエクスポート**」をクリックします。
  c)「**ファイルの保存**」を選択し、「**OK**」をクリックします。
  ユーザーまたはユーザー・グループの権限を要約した csv ファイルが、指定の場所に保存されます。

タスクの結果

「**保存**」をクリックすると、 ただちに役割および権限の割り当てが有効になります。

役割ベースのアクセス制御サービス API へのアクセスとその使用

役割ベースのアクセス制御サービス API を使用して、コマンド行から役割ベースのアクセス制御ポリシーのライフサイクルを 管理します。

このタスクについて

役割ベースのアクセスのタスク (役割の作成、表示、更新、削除など) を実行します。特定の役割のユーザーまたはユーザー・グループのセットを追加および削除します。特定の役割に権限を付与します。システムに定義されている役割、ユーザー、ユーザー・グループ、および権限のリストを表示します。

新規役割の定義、この役割へのユーザー、ユーザー・グループ、および権限の割り当てなどのタスクを自動化するためのスクリプトを作成できます。

手順

1. API の解説トピックのステップ 1 からステップ 9 を実行します。
   ステップ 10 およびステップ 11 は、追加の詳細です。

2. 「**USE (使用)**」をクリックし、キーを選択します (例えば、「**Key1**」)。

   注 : 「**非表示 (Hide)**」をクリックして、client-Id と client secret を表示します。これらは API Explorer 以外の外部ツールで API 呼び出しを行う場合に必要となるため、ここでメモしておきます。次に「**表示 (Show)**」をクリックして、これらの項目を非表示にします。
3. アスタリスクで示されたすべての必須ヘッダーを入力します。

   X-IBM-Service-Location
   * ヘッダーはサブスクリプションの地理的位置 (例えば、北アメリカの場合は na) です。

   許可
   * ヘッダーは IBM ID とパスワードの base64 エンコード・ストリングです。IBM ID とパスワードを based64-encoder ツールでエンコードするときは、*IBMid:password* という形式にする必要があります (例えば、`Basic YXBtYWRtaW46YXBtcGFzcw==!`)。
4. すべての POST 要求、PUT 要求、および DELETE 要求に Referer ヘッダーを組み込む必要があります。Referer ヘッダーの値は、常に以下のようになります。

   ```
   -H 'Referer: https://api.ibm.com'
   ```
5. 「**テスト (Test)**」の位置までスクロールしてクリックします。

例

以下の例では curl コマンドを使用して、新しい役割を作成します。

```
POST /1.0/authzn/roles
```

注 : POST 要求の本文には、**-d** パラメーターによって示される役割を定義する JSON オブジェクトを含める必要があります。

```
curl -X POST ¥
  https://api.ibm.com/perfmgmt/run/1.0/authzn/roles  ¥
  -H 'Referer: https://api.ibm.com' ¥
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' ¥
  -H 'content-type: application/json' ¥
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' ¥
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' ¥
  -d '{
    "description": "Your Role Description",
    "id": "/authzn/roles/Your_Role_Id",
    "label": "Your Role Name"
}'
```

エージェントの管理

IBM Cloud Application Performance Management インストール済み環境には、モニター・エージェントを管理するためのツールが用意されています。

これらのツールの一部は、管理対象システム (181 ページの『エージェント・コマンドの使用』、186 ページの『「エージェント構成」ページ』、186 ページの『Windows システムでの「IBM Cloud Application Performance Management」ウィンドウの使用』) の初期構成時にも使用します。

非 root ユーザーとしてのエージェントの開始

エージェントを別のユーザーとして開始する場合、システム上に共通グループを作成し、各ユーザーをこのグループのメンバーにします。

始める前に

エージェントのインストールと構成を同じ非 root ユーザーとして行ったときに、その同じユーザーとしてエージェントを開始する場合は、特にアクションは必要ありません。エージェントのインストールと構成を特定のユーザーとして行ったときに、そのユーザーとは別のユーザーとしてエージェントを開始する場合は、システム上に共通グループを作成します。すべてのエージェント管理ユーザーを、この共通グルー

プのメンバーにします。すべてのエージェント・ファイルおよびディレクトリーの所有権をこのグループに移転します。

このタスクについて

自動開始スクリプトは、インストール、アップグレード、または構成によって生成されます。このスクリプト (名前は UNIX オペレーティング・システムに応じて ITMAgentsN または rc.itmN) には、特定のインストールの各アプリケーションに対する項目が含まれています。デフォルトでは、すべてのエージェントが root ユーザー権限で開始されます。システム開始スクリプトを更新して非 root ユーザーとしてエージェントを開始するには、install_dir/config/kcirunas.cfg ファイルを編集する必要があります。このファイルには XML 構文のスーパーセットが入っています。kcirunas.cfg ファイルの各 **productCode** セクションは、デフォルトでは無効になっています。ご使用のエージェントの **productCode** セクションをアクティブ化するには、**!productCode** からコメント標識を削除します。コメント化または非アクティブ化されているセクションは無視されます。インストールされていないアプリケーションの場合は、コメント化されていないセクションまたはアクティブ化されているセクションが無視されます。

手順

1. モニター・エージェントを Linux または UNIX にインストールします (128 ページの『エージェントのインストール』 (AIX システムの場合) または 137 ページの『エージェントのインストール』 (Linux システムの場合) の説明に従ってください)。
2. オプション: 必要に応じて、モニター・エージェントを Linux または UNIX で構成します。163 ページの『第 7 章 環境の構成』 を参照してください。
3. 非 root ユーザーのグループ名を指定して ./secure.sh スクリプトを実行することにより、ファイルを保護し、ファイルに対するファイル・グループ所有権を設定します。
   例: **./secure.sh -g db2iadm1**
4. システム開始スクリプトを更新するには、以下のステップを実行します。
   a) install_dir/config/kcirunas.cfg ファイルを更新します。エージェントの **productCode** セクションをアクティブ化します。インスタンス値が不要なエージェントの場合は、product_code、instance、および user を指定します。ここで、*product_code* の値は、181 ページの表 11 で指定されている 2 文字のコードです。インスタンス値が必要なエージェント (Db2 モニター・エージェント (製品コード: ud) など) の場合は、product_code、instance、user、および name を指定します。
   例:

```
<productCode>ud</productCode>
<instance>
<name>db2inst1</name>
<user>db2inst1</user>
</instance>
<instance>
<name>db2inst2</name>
<user>root</user>
</instance>
```

   b) root ユーザーまたは sudo ユーザー権限を使用してスクリプト *install_dir*/bin/UpdateAutoRun.sh を実行します。

次のタスク

./secure.sh スクリプトについて詳しくは、エージェント・インストール・ファイルの保護を参照してください。

エージェントのインストールとアップグレードには同じユーザー ID を使用してください。

トランザクション・モニターのイベントしきい値

イベントしきい値を使用して、環境を即時モニターできます。特定の条件をテストし、重要パフォーマンス指標がしきい値を超えたときにイベントを発生させるカスタム・イベントしきい値を作成することもできます。

Response Time Monitoring イベント

応答時間イベントは、Web トランザクションが「**応答時間**」しきい値を超えると作成されます。

「**システム構成 (System Configuration)**」 > 「**しきい値マネージャー**」をクリックし、**データ・ソース・タイプ**として「**応答時間**」を選択します。Response Time Monitoring 環境のすべてのイベントしきい値は、同じタイプのすべての管理対象システムに対して適用されます。

Response Time Monitoring エージェントに使用可能な事前定義のしきい値は、以下のとおりです。

表 *243. Response Time Monitoring* しきい値

しきい値	説明	式
応答時間と可用性 （クリティカル）	高い割合の Web トランザクションが失敗しました。	WRT Transaction Status.Percent_Failed が 10 より大きく、WRT Transaction Status.Transaction_Definition_Name が「Ignore Resources」ではない場合、Response_Time_Availability_Crit は true です
応答時間と可用性 （警告）	中程度の割合の Web トランザクションが失敗しました。	WRT Transaction Status.Percent_Failed が 0 より大きく、WRT Transaction Status.Percent_Failed が 10 より小さく、WRT Transaction Status.Transaction_Definition_Name が「Ignore Resources」ではない場合、Response_Time_Availability_Warn は true です
応答時間 （クリティカル）	応答時間が遅い Web トランザクションの割合が高くなっています。	WRT Transaction Status.Percent_Slow が 5 より大きく、WRT Transaction Status.Percent_Available が 100 に等しく、WRT Transaction Status.Transaction_Definition_Name が「Ignore Resources」ではない場合、Response_Time_Critical は true です
応答時間 （警告）	応答時間が遅い Web トランザクションの割合が中程度です。	WRT Transaction Status.Percent_Slow が 1 より大きく、WRT Transaction Status.Percent_Slow が 5 より小さく、WRT Transaction Status.Percent_Available が 100 に等しく、WRT Transaction Status.Transaction_Definition_Name が「Ignore Resources」ではない場合、Response_Time_Warning は true です

正常に実行された要求数」は応答時間が 10 秒未満です。「低速要求数」は応答時間が 10 秒を超えています。応答時間が正常か低速かの判別に使用する 10 秒の値は構成できません。

トランザクション・トラッキング・イベント

トランザクション・トラッキング・イベントは、ミドルウェア・トランザクションがトランザクション・トラッキングのしきい値を超えると作成されます。

デフォルトのトランザクション・トラッキングしきい値を表示するには、 「**システム構成 (System Configuration)**」 > 「**しきい値マネージャー**」をクリックし、**データ・ソース・タイプ**として「**トランザクション・トラッキング**」を選択します。

ヒント : 必要に応じて独自のトランザクション・トラッキングしきい値を作成することができます。

ミドルウェア・トランザクションに使用可能な事前定義のしきい値は、以下のとおりです。

表 *244.* トランザクション・トラッキングしきい値

しきい値	説明	式
Interaction_Avail_Critical	ミドルウェアの対話が失敗した割合が高くなっています。	KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED が 10 より大きい場合、Interaction_Avail_Critical は true です。
Interaction_Avail_Warning	ミドルウェアの対話が失敗した割合が中程度です。	KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 より大きく、KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED が 10 以下である場合、Interaction_Avail_Warning は true です
Interaction_Time_Critical	合計時間が長いミドルウェアの対話の割合が高くなっています。	KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW が 5 以上であり KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 に等しい場合、Interaction_Time_Critical は true です
Interaction_Time_Warning	合計時間が長いミドルウェアの対話の割合が中程度です。	KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW が 1 より大きく、KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW が 5 より小さく、KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 の場合、Interaction_Time_Warning は true です
Transaction_Avail_Critical	ミドルウェア・トランザクションが失敗した割合が高くなっています。	KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_FAILED が 10 より大きい場合、Transaction_Avail_Critical は true です。

表 *244.* トランザクション・トラッキングしきい値 *(*続き*)*

しきい値	説明	式
Transaction_Avail_Warning	ミドルウェア・トランザクションが失敗した割合が中程度です。	KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 より大きく、KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_FAILED が 10 以下である場合、Transaction_Avail_Warning は true です
Transaction_Time_Critical	合計時間が長いミドルウェア・トランザクションの割合が高くなっています。	KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_SLOW が 5 以上であり KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 に等しい場合、Transaction_Time_Critical は true です
Transaction_Time_Warning	合計時間が長いミドルウェア・トランザクションの割合が中程度です。	KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_SLOW が 1 より大きく、KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_SLOW が 5 より小さく、KTE TRANSACTION AGGREGATE DATA.PERCENTAGE_FAILED が 0 の場合、Transaction_Time_Warning は true です

正常に実行された要求数」は応答時間が 10 秒未満です。「低速要求数」は応答時間が 10 秒を超えています。応答時間が正常か低速かの判別に使用する 10 秒の値は構成できません。

トランザクション・モニターのイベントを生成するためのしきい値の作成

しきい値マネージャーを使用して、トランザクション用のしきい値を作成します。しきい値は、サンプリングした属性の値をしきい値での設定値と比較するために使用します。サンプリングした値が比較条件を満たすと、トランザクション・イベントが生成されます。

このタスクについて

しきい値により、アプリケーションが特定の状態を報告した場合にモニターできます。トランザクション・モニターのデフォルトのしきい値について詳しくは、1015 ページの『トランザクション・モニターのイベントしきい値』を参照してください。

トランザクションの他の側面をモニターするために追加のしきい値を作成することができます。例えば、ミドルウェア・トランザクションのイベント・レートが低くなった場合にモニターするためのしきい値を作成できます。これにより、トランザクションのイベント・レートがしきい値で指定した数を下回る場合は、イベントが生成されます。

手順

しきい値を作成して、1 つ以上のトランザクションにそれを関連付けるには、以下のタスクを実行します。

1. ナビゲーション・バーで、「**システム構成**」 > 「**しきい値マネージャー**」をクリックします。**データ・ソース・タイプ**を「**トランザクション・トラッキング**」に設定します。
2. 「追加」をクリックして、新しいしきい値を作成します。
3. このしきい値を超えるイベントの重大度を設定します。

4. このしきい値をトランザクションに関連付けるために、以下の値を設定します。
   - データ・セット - KTE TRANSACTION AGGREGATE DATA
   - 表示項目 - Resource_Value
   - 論理演算子 - And （&）
5. 別の方法として、このしきい値を対話と関連付けるために、次の値を設定します。
   - データ・セット - KTE INTERACTION AGGREGATE DATA
   - 表示項目 - Source_Resource_Value
   - 論理演算子 - And （&）
6. 「**追加**」をクリックして、新しい条件を追加します。「**条件の追加**」ボックスで、属性と演算子を選択し、しきい値の値を入力します。

   例えば、1 分当たりのトランザクションの数が 100 を下回った場合にトランザクション・イベントを生成するしきい値条件を追加するには、以下の値を設定し、「**OK**」をクリックします。
   - 属性 - Transaction_Rate
   - 演算子 - より小
   - 値 - 100

   必要に応じて、このステップを繰り返して、しきい値にさらに条件を追加します。
7. 「グループ割り当て」セクションで、「トランザクション・トラッキング」を選択して、そのリソース・グループにしきい値を追加します。
8. 「**保存**」をクリックします。

タスクの結果

これで、しきい値を作成して、それをトランザクションまたは対話に関連付けました。しきい値条件が満たされると、イベントが生成されます。Application Performance ダッシュボードの「イベント」タブのイベントをモニターすることができます。

例

Response Time Monitoring エージェントで、デフォルトのしきい値に加えて Web トランザクションの他の側面をモニターするためにしきい値を作成するには、以下のようにします。

1. しきい値マネージャーで、**データ・ソース・タイプ**を「**応答時間**」に設定します。
2. しきい値を追加する際に、以下の設定を使用します。
   - データ・セット - WRT トランザクションの状況
   - 表示項目 - アプリケーション
   - 論理演算子 - And （&）
   - グループ割り当て - Web 応答時間

OS エージェント・イベントの管理

イベントを管理するように OS エージェントを構成できます。

イベント・フィルタリングおよび要約

構成 (.conf) ファイル内に設定するイベント・フィルタリングおよび 要約オプションを使用して、OS エージェントが重複イベントを処理する方法を制御します。

ログがモニターされるときに、1 つのイベントが短期間で複数回表示されることがあります。例えば、この繰り返しロギングは、ログを生成するアプリケーションがエラーを検出し、そのしきい値が解決されるまでこのエラーを継続的にログに記録した場合に発生することがあります。このようなロギングが発生すると、非常に多くのイベントが Performance Management インフラストラクチャーに送信されます。大量のイベントはパフォーマンスに悪影響を及ぼします。

注 : イベント検出および要約手順は、Performance Management に送信されるイベントでのみサポートされます。EIF によって OMNIbus に送信されるイベントでこれらの手順を実行することはできません。

重複イベントの検出およびフィルタリング
重複イベントを処理するように、OS エージェントを構成できます。

複数の重複イベントの問題を軽減するために、.conf ファイル内で DupDetectionKeyAttributes タグを使用して、重複イベントの構成要件を定義します。コンマ区切りリストに、あるイベントが重複しているかどうかを判別するために使用する 1 つ以上の定義済み Performance Management 属性を組み込みます。以下の例では、同じメッセージおよび同じ CustomSlot1 を持つイベントは、重複していると見なされます。

```
DupDetectionKeyAttributes=msg,CustomSlot1
```

重複イベントは、Performance Management 属性から検出されます。したがって、定義した特定のスロットに基づいて重複を検出する場合は、以下のステップを実行します。

1. スロット値を Performance Management 属性にマップします。
2. Performance Management 属性を .conf ファイル内の DupDetectionKeyAttributes タグにマップします。

以下の例では、重要なスロット eventclass および eventid が *CustomSlot1* および *CustomSlot2* にマップされます。

```
REGEX BaseAuditEvent
^([A-Z][a-z]{2} [0-9]{1,2} [0-9]{1,2}:[0-9]{2}:[0-9]{2}
 [0-9] {4}) [0-9] (¥S+) (¥S+) ¥
Microsoft-Windows-Security-Auditing (¥S+) ([0-9]+) (.*)
timestamp $1
severity $2
eventclass $3 CustomSlot1
eventkeywords $4
eventid $5 CustomSlot2
msg $6
END
```

.conf ファイル内で特定のイベントを重複イベントとして指定する場合は、以下に示すように、Performance Management 属性を DupDetectionKeyAttributes タグにマップします。

```
DupDetectionKeyAttributes=CustomSlot1,CustomSlot2
```

注 :

1. CustomSlot 属性名では大/小文字が区別されるため、上記の例に示された名前を正確に入力する必要があります。
2. 属性のリストを指定しなかった場合、値はデフォルトで Class および Logname になります。

これらの属性が一致するイベントは、エージェントによって重複イベントであると見なされます。

重複検出はグローバルであるため、キーとして使用するカスタム・スロットのセットを選択し、それらのスロットをすべてのフォーマット指定ステートメント内でキーとして使用する手法をお勧めします。例えば、キーとしてスロット 1 から 3 を使用します。あるフォーマットでキーを 1 つのみ指定し、さらに追加のスロットを指定する必要がある場合は、名前値を格納するためにスロット 1 を使用し、その他のデータを格納するためにスロット 4 から n を使用します。

要約間隔
重複検出手順は、要約間隔と呼ばれる 1 つの期間の中で実行されます。

重複イベントは、この間隔内でカウントされ、間隔が終了するとリセットされます。カウンターは、新規の要約間隔が開始されるたびに、カウントを 0 から再び開始します。

エージェントは、間隔内でモニターした各イベント・セットの要約イベントを送信します。要約イベントには、一致した最初のイベントの属性値が含まれます。要約イベントには、要約間隔内で発生したそのイベントの重複回数を示す数も含まれます。

要約間隔は、構成 (.conf) ファイル内で、以下の例に示すように設定されます。

```
EventSummaryInterval=300
```

要約間隔に割り当てられる値は秒単位であるため、この例の要約間隔は 5 分です。

イベントのフィルタリング
イベントのフィルタリングが実行されている場合、(.conf) ファイル内の EventFloodThreshold 設定は、エージェントに対して、イベントをいつ送信するかを通知します。

以下の表は、EventFloodThreshold 値を示しています。

表 *245. EventFloodThreshold* 値

EventFloodThreshold 値	**説明**
send_all	*send_all* 値はデフォルト値です。イベントが重複イベントである場合でも、すべてのイベントが送信されます。
send_none	*send_none* 値は、個々のイベントが送信されないことを意味します。要約イベントのみが送信されます。
send_first	*send_first* 値を使用すると、最初のイベントが検出されたら直ちに送信されます。指定された時間内にその最初のイベントと重複するイベントが検出された場合、この最初のイベントの後続の重複イベントは送信されません。詳しくは、1019 ページの『要約間隔』を参照してください。
n integer	*n* integer 値を使用すると、特定の時間内に検出された *n* 番目ごとの重複イベント (例えば、5 番目ごとの重複イベント) のみが送信されます。詳しくは、1019 ページの『要約間隔』を参照してください。

要約属性
イベント・タイプ属性と発生回数属性は、イベントの要約を支援するために使用されます。

イベント要約が使用可能である場合、イベント・タイプ属性と発生回数属性は意味を持ちます。イベント・タイプ属性は、イベントのタイプを示し、*Event* または *Summary Event* のいずれかです。ログ内にあるレコードと 1 対 1 で対応する一般的なイベントには、*Event* タグが付けられます。要約間隔の最後に送信される要約イベントには、*Summary Event* タグが付けられます。

発生回数属性は、ログ内にあるイベントの重複レコードの合計数を示します。この属性は、直前の要約間隔内で受信されたイベントのうち、要約イベントと一致したイベントの数を表示します。したがって、要約イベントにはこの合計数が含まれます。

しきい値と要約イベント

1020 ページの『イベントのフィルタリング』で説明したフィルター値に関係なく、各要約間隔の最後には必ず、その間隔中に少なくとも 1 回発生したあらゆるイベントについての要約イベントが得られます。要約イベントを想定していないと、しきい値が誤ってトリガーされることがあります。このようにしきい値が誤ってトリガーされることを避けるには、しきい値に *Event Type== Event* または *Event Type!= Summary Event* の節を組み込みます。

Windows イベント・ログ

OS エージェントは、.conf ファイルを使用して Windows イベント・ログからのイベントをモニターします。

OS エージェントは従来どおり、WINEVENTLOGS 構成 (.conf) ファイル・オプションを使用して、Windows イベント・ログからのイベントをモニターします。エージェントは、以下の例に示すように、イベント・ログのコンマ区切りリストをモニターします。

```
WINEVENTLOGS=System,Security,Application
```

また、OS エージェントは従来どおり、WINEVENTLOGS=All 設定を使用します。All 設定は、2008 より前の Windows バージョンで提供される標準イベント・ログ (セキュリティー、アプリケーション、システム、ディレクトリー、ドメイン・ネーム・システム (DNS)、およびファイル複製サービス (FRS)) を意味します。しかし、システム上のすべてのイベント・ログが検査されるわけではありません。

UseNewEventLogAPI 構成ファイル・タグを指定すると、イベント・ログ (Windows イベント・ログ 2008 以降) は、Microsoft によって追加されたすべての新規ログ、および他のアプリケーションまたはユーザーによって作成されたすべての Windows イベント・ログにアクセスできます。新規ログは、WINEVENTLOGS キーワードによってリストされます。

以下の例では、UseNewEventLogAPI タグは y に設定されます。

```
UseNewEventLogAPI=y
WINEVENTLOGS=Microsoft-Windows-Hyper-V-Worker-Admin
```

この例では、Hyper-V 役割を持つ Windows システム上で Microsoft-Windows-Hyper-V/Admin がモニターされます。

Windows イベント・ログ内では、各イベントに以下のフィールドがあり、この順序で配列されています。

- 日付 (フォーマット: 月、日、時刻、および年)
- イベント・カテゴリー (整数)
- イベント・レベル
- Windows セキュリティー ID。構成 (.conf) ファイル内で SpaceReplacement=TRUE である場合、Windows セキュリティー ID 内のすべてのスペースは下線に置き換えられます。

  注: SpaceReplacement=TRUE は、(.conf) ファイル内で UseNewEventLogAPI を y に設定した (イベント・ログを使用していることを指定した) 場合のデフォルトです。
- Windows ソース。構成 (.conf) ファイル内で SpaceReplacement=TRUE である場合、Windows ソース内のすべてのスペースは下線に置き換えられます。
- Windows イベント・ログ・キーワード。構成 (.conf) ファイル内で SpaceReplacement=TRUE である場合、Windows イベント・ログ・キーワード内のすべてのスペースは下線に置き換えられます。

  注: ここで説明するキーワード・フィールドは、Windows 2008 バージョンのイベント・ログでは新規です。このキーワード・フィールドは前のイベント・ログには存在しなかったため、これが含まれている場合、古いイベント・ログ・フォーマット指定ステートメントをそのまま再使用することはできません。この追加のフィールドに対応するようにステートメントを変更する必要があります。
- Windows イベント ID
- メッセージ・テキスト

例えば、管理ユーザーが Windows 2008 システムにログオンすると、新規ユーザー・セッションに割り当てられた特権を示すイベントがセキュリティー・ログ内で生成されます。

```
Mar 22 13:58:35 2011 1 Information N/A Microsoft-Windows-
Security-Auditing Audit_Success 4672 Special privileges assigned to new logon.
S-1-5-21-586564200-1406810015-1408784414-500    Account Name:
Administrator   Account Domain:     MOLDOVA     Logon ID:
0xc39cb8e    Privileges:        SeSecurityPrivilege
SeBackupPrivilege          SeRestorePrivilege
SeTakeOwnershipPrivilege            SeDebugPrivilege
```

```
SeSystemEnvironmentPrivilege            SeLoadDriverPrivilege
SeImpersonatePrivilege
```

Microsoft-Windows-Security-Auditing イベント・ソースによって作成されたすべてのイベントを取り込むには、以下に示すように、フォーマット指定ステートメントを記述します。

```
REGEX BaseAuditEvent
^([A-Z][a-z]{2} [0-9]{1,2} [0-9]{1,2}:[0-9]{2}:[0-9]{2} [0-9]
{4}) [0-9] (¥S+) (¥S+) Microsoft-Windows-Security-Auditing (¥S+)
([0-9]+) (.*)
timestamp $1
severity $2
login $3
eventsource "Microsoft-Windows-Security-Auditing"
eventkeywords $4
eventid $5
msg $6
END
```

以下の例は、上記のイベント例のスロットに割り当てられる値を示します。

```
timestamp=Mar 22 13:58:35 2011
severity=Information
login=N/A
eventsource=Microsoft-Windows-Security-Auditing
eventid=4672
msg="Special privileges assigned to new logon.
S-1-5-21-586564200-1406810015-1408784414-500    Account Name:
Administrator   Account Domain:     MOLDOVA     Logon ID:
0xc39cb8e    Privileges:        SeSecurityPrivilege
SeBackupPrivilege          SeRestorePrivilege
SeTakeOwnershipPrivilege            SeDebugPrivilege
SeSystemEnvironmentPrivilege            SeLoadDriverPrivilege
SeImpersonatePrivilege
```

これらのイベントがどのように表示されるかを正確に予測するのは難しいため、正規表現を記述するときには、実際のイベントをファイル内に取り込む手法を使用することをお勧めします。その後、このファイルを確認し、エージェントに取り込ませるイベントを選択し、これらのイベントと一致する正規表現を記述することができます。Windows イベント・ログからすべてのイベントを取り込むには、以下のステップを実行します。

1. 以下の例に示すように、どのイベントとも一致しない 1 つのパターンのみを含むフォーマット・ファイルを作成します。

   ```
   REGEX NoMatch
   This doesn't match anything
   END
   ```

2. 以下の設定を構成 (.conf) ファイルに追加します。

   ```
   UnmatchLog=C:/temp/evlog.unmatch
   ```

3. エージェントを実行し、いくつかのサンプル・イベントを取り込みます。

イベント・マッピング

Tivoli Event Integration Facility (EIF) インターフェースは、シチュエーション・イベントを Tivoli Netcool/OMNIbus、Tivoli Enterprise Console、または Operations Analytics - Log Analysis に転送するために使用します。

EIF イベントはイベント・クラスを指定し、イベント・データは、イベント・スロットの名前とそのスロットの値を識別する名前と値の組として指定されます。イベント・クラスはサブクラスを持つことができます。Performance Management では、すべてのモニター・イベントに組み込まれる、基本イベント・クラス定義と一連の基本スロットが提供されます。エージェントは基本イベント・クラスを拡張して、エージェント固有のスロットが組み込まれたサブクラスを定義します。OS エージェント・ログ・ファイル・イベントでは、イベント・クラスはエージェント属性グループに対応し、エージェント固有スロットは属性グループ内の属性に対応します。

LFAProfiles 属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_LFAPROFILES イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- subnode_msn: STRING
- subnode_affinity: STRING
- subnode_type: STRING
- subnode_resource_name: STRING
- subnode_version: STRING
- subnode_config_file: STRING
- subnode_description: STRING
- subnode_description_enum: STRING

ログ・ファイル正規表現統計属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_LOG_FILE_REGEX_STATISTICS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- table_name: STRING
- attrib_name: STRING
- filter_number: INTEGER
- average_processor_time: REAL
- average_processor_time_enum: STRING
- total_processor_time: REAL
- total_processor_time_enum: STRING
- max_processor_time: REAL
- max_processor_time_enum: STRING
- min_processor_time: REAL
- min_processor_time_enum: STRING
- filter_count: REAL
- filter_count_matched: REAL
- filter_count_unmatched: REAL
- regex_pattern: STRING
- last_matched_time: STRING
- last_matched_time_enum: STRING
- last_unmatched_time: STRING
- last_unmatched_time_enum: STRING
- result_type: INTEGER
- result_type_enum: STRING

ログ・ファイル状況属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_LOG_FILE_STATUS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- table_name: STRING
- file_name: STRING
- regex_pattern: STRING
- file_type: INTEGER
- file_type_enum: STRING
- file_status: INTEGER
- file_status_enum: STRING
- num_records_matched: INTEGER
- num_records_not_matched: INTEGER
- num_records_not_matched_enum: STRING
- num_records_processed: INTEGER
- current_file_position: REAL
- current_file_position_enum: STRING
- current_file_size: REAL
- current_file_size_enum: STRING
- last_modification_time: STRING
- last_modification_time_enum: STRING
- codepage: STRING

LogfileEvents 属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_LOGFILEEVENTS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- klo_class: STRING
- logname: STRING
- eifevent: STRING
- klo_msg: STRING
- customslot1: STRING
- customslot2: STRING
- customslot3: STRING
- customslot4: STRING
- customslot5: STRING
- customslot6: STRING
- customslot7: STRING
- customslot8: STRING

- customslot9: STRING
- customslot10: STRING
- occurrence_count: INTEGER
- occurrence_count_enum: STRING
- event_type: INTEGER
- event_type_enum: STRING
- custominteger1: REAL
- custominteger1_enum: STRING
- custominteger2: REAL
- custominteger2_enum: STRING
- custominteger3: REAL
- custominteger3_enum: STRING
- remotehost: STRING

LogfileProfileEvents 属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_LOGFILEPROFILEEVENTS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- klo_class: STRING
- logname: STRING
- eifevent: STRING
- klo_msg: STRING
- customslot1: STRING
- customslot2: STRING
- customslot3: STRING
- customslot4: STRING
- customslot5: STRING
- customslot6: STRING
- customslot7: STRING
- customslot8: STRING
- customslot9: STRING
- customslot10: STRING
- occurrence_count: INTEGER
- occurrence_count_enum: STRING
- event_type: INTEGER
- event_type_enum: STRING
- custominteger1: REAL
- custominteger1_enum: STRING
- custominteger2: REAL

- custominteger2_enum: STRING
- custominteger3: REAL
- custominteger3_enum: STRING
- remotehost: STRING

パフォーマンス・オブジェクト状況属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_PERFORMANCE_OBJECT_STATUS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- query_name: STRING
- object_name: STRING
- object_type: INTEGER
- object_type_enum: STRING
- object_status: INTEGER
- object_status_enum: STRING
- error_code: INTEGER
- error_code_enum: STRING
- last_collection_start: STRING
- last_collection_start_enum: STRING
- last_collection_finished: STRING
- last_collection_finished_enum: STRING
- last_collection_duration: REAL
- average_collection_duration: REAL
- average_collection_duration_enum: STRING
- refresh_interval: INTEGER
- number_of_collections: INTEGER
- cache_hits: INTEGER
- cache_misses: INTEGER
- cache_hit_percent: REAL
- intervals_skipped: INTEGER

pro パフォーマンス・オブジェクト状況属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_PRO_PERFORMANCE_OBJECT_STATUS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- query_name: STRING
- object_name: STRING
- object_type: INTEGER
- object_type_enum: STRING
- object_status: INTEGER

- object_status_enum: STRING
- error_code: INTEGER
- error_code_enum: STRING
- last_collection_start: STRING
- last_collection_start_enum: STRING
- last_collection_finished: STRING
- last_collection_finished_enum: STRING
- last_collection_duration: REAL
- average_collection_duration: REAL
- average_collection_duration_enum: STRING
- refresh_interval: INTEGER
- number_of_collections: INTEGER
- cache_hits: INTEGER
- cache_misses: INTEGER
- cache_hit_percent: REAL
- intervals_skipped: INTEGER

スレッド・プール状況属性グループのしきい値によってイベントが生成される場合は、ITM_KLO_THREAD_POOL_STATUS イベント・クラスを使用してイベントが送信されます。このイベント・クラスには、以下のスロットが含まれます。

- node: STRING
- timestamp: STRING
- thread_pool_size: INTEGER
- thread_pool_size_enum: STRING
- thread_pool_max_size: INTEGER
- thread_pool_max_size_enum: STRING
- thread_pool_active_threads: INTEGER
- thread_pool_active_threads_enum: STRING
- thread_pool_avg_active_threads: REAL
- thread_pool_avg_active_threads_enum: STRING
- thread_pool_min_active_threads: INTEGER
- thread_pool_min_active_threads_enum: STRING
- thread_pool_max_active_threads: INTEGER
- thread_pool_max_active_threads_enum: STRING
- thread_pool_queue_length: INTEGER
- thread_pool_queue_length_enum: STRING
- thread_pool_avg_queue_length: REAL
- thread_pool_avg_queue_length_enum: STRING
- thread_pool_min_queue_length: INTEGER
- thread_pool_min_queue_length_enum: STRING

- thread_pool_max_queue_length: INTEGER
- thread_pool_max_queue_length_enum: STRING
- thread_pool_avg_job_wait: REAL
- thread_pool_avg_job_wait_enum: STRING
- thread_pool_total_jobs: INTEGER
- thread_pool_total_jobs_enum: STRING

Website Monitoring でのシンセティック・トランザクションおよびシンセティック・イベントの管理

異なるロケーションにある内部アプリケーション、外部アプリケーション、およびパブリック向け Web アプリケーションのパフォーマンスおよび可用性をモニターするシンセティック・トランザクションを作成します。

Synthetic Script Manager でシンセティック・トランザクション を作成します。 アプリケーションの可用性をテストするために Synthetic Script Manager で単純なスクリプトを生成するか、Selenium IDE を使用して、アプリケーションでのさまざまなユーザー・アクションを再現するシンセティック・スクリプトを記録します。その後、再生ロケーションで特定の間隔でスクリプトが再生されるようにシンセティック・トランザクションを構成します。

重要 : IBM Website Monitoring on Cloud アドオンの既存のユーザーのみが、Synthetic Playback エージェント および Synthetic Script Manager を使用できます。Website Monitoring は、2017 年 8 月リリースの IBM Cloud Availability Monitoring に置き換えられました。詳しくは、1047 ページの『Availability Monitoring について』を参照してください。

使用可能な再生ロケーションは、Monitoring Agent for Synthetic Playback をインストールしたロケーション、およびパブリック向け Web アプリケーションのモニター用に提供されている 15 個所のポイント・オブ・プレゼンス (PoP) です。PoP は次のロケーションについて使用できます。

- アムステルダム
- チェンナイ
- ダラス
- フランクフルト
- 香港
- ロンドン
- メルボルン
- メキシコ
- パリ
- サンノゼ
- サンパウロ
- シンガポール
- 東京
- トロント
- ワシントン

イベントを発生させるためのしきい値とリソース・グループを作成し、アプリケーションが低速または使用不可の場合に利用者に通知を送ります。Application Performance ダッシュボードでパフォーマンス・データを表示し、ヒストリカル・レポートを生成します。

応答時間エージェントを使用してアプリケーションのエンド・ユーザー応答時間をモニターする場合は、エンド・ユーザー・トランザクションとシンセティック・トランザクションの両方の KPI を Application Performance ダッシュボードで表示できます。シンセティック・トランザクションをコンポーネントとして、応答時間エージェントを使用してモニターしているアプリケーションに追加します。

注 : Synthetic Script Manager で作業するには、「Synthetic Script Manager」および「エージェント構成」の表示権限を持つ役割のメンバーである必要があります。詳しくは、1003 ページの『役割と権限』を参照してください。

シンセティック・スクリプトの記録

シンセティック・スクリプトを記録するには、Firefox Web ブラウザーと Selenium IDE アドオンを使用します。Selenium IDE を使用して、ページのロード、リンクのクリック、オブジェクトの選択などの、Web ページでのユーザー・アクションを記録できます。Selenium IDE が記録しているときには、スクリプト内のユーザー・アクションごとに 1 つのコマンドが生成されます。その後、Synthetic Script Manager を使用して、Web サイトでのユーザーの振る舞いを、設定した間隔でさまざまなロケーションでシミュレートするためのスクリプトを構成できます。

始める前に

スクリプトの記録時は Firefox Web ブラウザーを使用する必要があります。

Selenium IDE は、Firefox のアドオンとしてのみ使用可能です。Selenium IDE がインストールまたは実行されていない場合は、次のステップを実行します。

1. Selenium IDE 3.2.X または 3.3.X をサポートするバージョンの Firefox 60 以降を実行していることを確認します。新しいバージョンの Selenium IDE がある場合、それはサポートされません。アンインストールしてバージョン 3.2.X または 3.3.X をインストールする必要があります。

   注 : デフォルトでは、バージョン 3.2.X または 3.3.X をインストールした後に Selenium IDE が自動的に更新されます。バージョンアップを防止するために、Selenium IDE の自動更新をオフにしてください。
2. **Selenium** のホーム・ページ (https://addons.mozilla.org/firefox/addon/selenium-ide/versions/) から Selenium IDE 3.2.X または 3.3.X をダウンロードしてインストールします。Selenium IDE がすべてのプラグインをインストールするよう許可します。
3. Selenium IDE がインストールされたら、Firefox を再始動します。
4. テスト対象の Web ページにナビゲートして、その他のタブをすべて閉じます。Selenium IDE を開くには、「**ツール**」 > 「**Selenium IDE**」をクリックします。「**Selenium IDE**」ウィンドウで、「**ベース URL**」フィールドに、表示されている Web ページの URL が含まれていることを確認します。Selenium IDE が、表示されている Web ページでのすべてのユーザー・アクションの記録を開始します。

Selenium .side スクリプト・フォーマット

新しいバージョンの Selenium で作成されたスクリプトでは、`.side` フォーマットが使用されます。Selenium IDE 3.2.X または 3.3.X では、`.html` フォーマットで作成された古いスクリプトをインポートして、`.side` フォーマットで保存できます。詳しくは、1032 ページの『旧バージョンの Selenium IDE からのスクリプトの更新』を参照してください。

Selenium の `.side` スクリプトを使用する場合は、最初に以下の更新をインストールする必要があります。

- IBM Cloud Application Performance Management V8.1.4.0 Synthetic Playback エージェント 暫定修正 5 以降 (Synthetic Playback エージェント のインストール先システムにインストールします)。
- ご使用の Cloud APM サブスクリプションが IBM IBM Cloud Application Performance Management, Private Cloud APM V8.1.4.0 サーバー暫定修正 8 以降に更新されているかどうかを確認するには、IBM にお問い合わせください。
- Availability Monitoring のプライベート PoP (point of presence) を使用する場合は、PoP のインストール・ディレクトリーから **`cat build.info`** コマンドを入力することにより、シンセティック PoP のビルド番号が `APM_201903090832` 以降であることを確認してください。以前のビルド・バージョンでは、`.side` フォーマットがサポートされていません。

Cloud APM V8.1.4.0 用の暫定修正は IBM Support > Fix Central > IBM APM 8.1.4.0 からダウンロードできます。

このタスクについて

このタスクでは、Web ページ上でユーザー・アクションを実行し、Selenium IDE を使用してこれらのアクションをコマンドとして単純なスクリプトに記録します。スクリプトを使用して、Application Performance ダッシュボードで Web アプリケーションのパフォーマンスと 可用性をモニターできます。

手順

Web ページ上でのユーザー・アクションのスクリプトを記録するには、以下の手順を実行します。

1. 「**記録**」をクリックして、スクリプトの記録を開始します。Web ページで、リンクのクリックなどのユーザー・アクションを実行します。

   Web ページ上のユーザー・アクションごとに、Selenium IDE は 1 つのコマンドを記録し、それをスクリプトに追加します。

   例えば、ユーザーがいつ IBM Marketplace Web ページをロードして Cloud APM の無料トライアルにナビゲートしたかをスクリプトに記録するには、以下のアクションを実行します。

表 *246.* 記録されるユーザー・アクションおよび *Selenium IDE* コマンド	
ユーザー・アクション	**スクリプトに追加されるコマンド**
IBM Marketplace Web サイトの Cloud APM Web ページをいつ開いたかを記録するために、IBM Marketplace Web ページを開きます。表示された Web ページの任意の箇所を右クリックして、「**open**」を選択します。	`open`
スクリプトが Web ページがロードされたことを確実にチェックするように、Web ページのタイトル・テキスト (IBM Cloud Application Performance Management) を右クリックして、「**利用可能な全てのコマンド**」 > 「**verifyTitle IBM Cloud Application Performance Management**」をクリックします。	`verifyTitle`
Cloud APM の詳細を表示するためのリンクをユーザーがいつクリックしたかを記録するために、「**詳細**」リンクをクリックします。「**詳細**」ページがロードされます。	`clickAndWait`
「**詳細**」ページがロードされたことをスクリプトが確実にチェックするように、「Feature spotlights」という見出しを右クリックして、「**利用可能な全てのコマンド**」 > 「**verifyText css=h2.heading--TERTIARY**」を選択します。	`verifyText`
Cloud APM の購入方法の詳細を表示するためのリンクをユーザーがいつクリックしたかを記録するために、「**購入**」リンクをクリックします。「**購入**」ページがロードされます。	`clickandWait`
Cloud APM の無料トライアルに登録するためのボタンをユーザーがいつクリックしたかを記録するために、「**無料で試す**」ボタンをクリックします。	`click`

2. Selenium IDE ウィンドウで「**記録**」をクリックして記録を停止します。「**プロジェクトの保存 (Save Project)**」ツールをクリックし、スクリプトに分かりやすい名前を付けて、`.side` ファイル (`open_webpage.side` など) として保存します。
3. Selenium IDE ウィンドウで、記録されたスクリプトをレビューします。「**テーブル**」タブをクリックして、スクリプトをテーブル形式で表示します。Selenium IDE ウィンドウで、「**現在のテスト・ケースを再生 (Play Current Test Case)**」をクリックして、記録したスクリプトの再生をテストします。

   この例では、ステップ 1 に記載したような IBM Marketplace Web サイト上のユーザー・アクションのスクリプトが Selenium IDE に表示されます。

表 *247. IBM Marketplace Web* サイト上でのユーザーのアクションの *Selenium IDE* スクリプト記録の例		
コマンド	**ターゲット**	**値**
open	/	

表 247. *IBM Marketplace Web* サイト上でのユーザーのアクションの *Selenium IDE* スクリプト記録の例 (続き)

コマンド	ターゲット	値
verifyTitle	IBM Cloud Application Performance Management	
clickAndWait	css=ul > #details > a	
verifyText	css=h2.heading--TERTIARY	Feature spotlights
clickAndWait	css=ul > #purchase > a	
click	link=Try Free	

タスクの結果
Web アプリケーションのパフォーマンスおよび 可用性をモニターするために使用できるスクリプトを記録しました。

次のタスク

1 つの複雑なスクリプトを記録した場合は、スクリプトをより単純な複数のスクリプトに整理して、各スクリプトが Web アプリケーションの特定のビジネス・プロセスまたはユーザー・アクションを表すようにします。

Synthetic Script Manager を使用して、スクリプト・ファイルを新規シンセティック・トランザクションまたは既存のシンセティック・トランザクションにアップロードします。

複雑なスクリプトの構造化
1 つの複雑なスクリプトを複数のスクリプトに整理します。その後、それらのスクリプトをまとめて、テスト・スイート と呼ばれるスクリプトのコレクションとして保存します。

このタスクについて

複雑なスクリプトを作成した場合は、そのスクリプトを複数の単純なスクリプトとして整理し、その単純なスクリプトが Web アプリケーションの各種ビジネス・プロセスまたはユーザー・プロセスを表すようにすることができます。複数のスクリプトをテスト・スイートとしてまとめて保存します。その後、これらのスクリプトを使用して、特定のユーザー・アクションに応じて Application Performance ダッシュボードで Web アプリケーションのパフォーマンスと 可用性をモニターできます。

テスト・スイートは 1 つのみでなければならず、すべてのテストをそのテスト・スイートに追加する必要があります。

重要 : 複雑なスクリプトを、各スクリプトがモニター対象の標準的なユーザー・プロセスまたはビジネス・プロセスを表す個別のスクリプトに整理することが推奨されます。例えば、ユーザーがいつ Web サイトにログインしたか、またはいつ項目を検索したかを記録する、別々のスクリプトを作成します。これらのスクリプトをユーザー・プロセスまたはビジネス・プロセスに従って整理すると、これらの特定のプロセスに対する Web アプリケーションの応答を Application Performance ダッシュボード でモニターすることができます。

手順

複雑なスクリプトを個別のスクリプトに整理し、スクリプトをテスト・スイートとして保存するには、以下の手順を実行します。

1. スクリプトに記録されるユーザー・プロセスごとに別々のスクリプトを作成するために、Selenium IDE で「**テスト**」 > 「**+**」をクリックします。スクリプトにユーザー・プロセスを説明する分かりやすい名前を付け、各スクリプトを `.side` ファイル (例: `load_homepage.side`) として保存します。

   詳しくは、1029 ページの『シンセティック・スクリプトの記録』を参照してください。

重要 : Selenium IDE でスクリプトに指定する名前は、Application Performance ダッシュボードでモニターする、記録対象のビジネス・プロセスまたはユーザー・プロセスを識別する名前です。

2. Selenium IDE で、前に記録した複雑なスクリプトを開きます。スクリプト・コマンドを、各種ユーザー・アクションに従って個別のスクリプトに整理します。「**テストケース**」ウィンドウで、元の複雑なスクリプトからコマンドを「**切り取り**」、別の「**テストケース**」に「**貼り付け**」ます。.

   例えば、シンセティック・スクリプトの記録の複雑なスクリプト例には、3 つの異なるユーザー・プロセスの Selenium IDE コマンドが含まれています。

   - IBM Marketplace Web サイトの Cloud APM ホーム・ページを開く。
   - IBM Marketplace の「**詳細**」ページを開く。
   - 「**価格設定**」ページを開き、無料トライアルの登録ページをユーザーがいつ開いたかを記録する。

   これらのユーザー・アクションが、3 つの別々のスクリプトに整理されます。

表 *248. IBM Marketplace* ページを開くサンプル・スクリプト *(load_homepage.side)*		
コマンド	**ターゲット**	**値**
open	/	
verifyTitle	IBM Cloud Application Performance Management	

表 *249. IBM Marketplace* の **「詳細」** ページを開くサンプル・スクリプト *(load_products.side)*		
コマンド	**ターゲット**	**値**
clickAndWait	css=ul > #details > a	
verifyText	css=h2.heading--TERTIARY	Feature spotlights

表 *250. IBM Marketplace* の「**購入**」ページおよびトライアルの登録ページを開くためのサンプル・スクリプト *(load_APM.side)*		
コマンド	**ターゲット**	**値**
clickAndWait	css=ul > #purchase > a	
click	link=Try Free	

3. 個々のテスト・ケースをテスト・スイートに入れるには、「**テスト・スイート**」 ウィンドウに移動し、ビジネス・ロジック・シーケンスに従ってテスト・スイートにテストを追加します。最後に、「**プロジェクトの保存**」ツールをクリックして、テスト・スイートおよびテスト・スイート内のすべてのテストを .side ファイルに保存します。
   例として、URL のロード、「インベントリーの管理」を選択、IBM マシン・タイプを選択という論理シーケンスについて考えてみます。これらのテスト・ケースをテスト・スイートに追加する場合、まず URL のロードを確認し、続いて「インベントリーの管理」を選択、IBM マシン・タイプを選択の順に確認します。

タスクの結果

Web アプリケーションのパフォーマンスおよび可用性をモニターするために使用できる一連のスクリプトを記録しました。Synthetic Script Manager を使用して、スクリプトの .side テスト・スイートを新規シンセティック・トランザクションまたは既存のシンセティック・トランザクションにアップロードします。

旧バージョンの Selenium IDE からのスクリプトの更新

サポートされている Selenium IDE バージョン 2.2.X および 3.2.X は、以前のバージョンの Selenium IDE で使用していた .html フォーマットではなく、.side フォーマットを使用してシンセティック・スクリプトを記録します。既存の .html スクリプトがある場合は、引き続きそれを使用できます。以前のバージョンの Selenium IDE で作成したスクリプトは、IBM Cloud Availability Monitoring が使用する最新の Firefox および Selenium ドライバーで一部の機能が機能しない場合があります。場合によっては、.html スクリプ

トを編集したり、新しい .side フォーマットで記録し直したり、.html スクリプトをインポートしてから .side フォーマットで保存し直す必要があります。

手順

- 例外: Select2 エレメントと対話する場合、**select** コマンドを使用しないでください (https://github.com/SeleniumHQ/selenium-ide を参照)。
  古いスクリプトは以下のとおりです。

```
<td>select</td>
<td>id=country</td>
<td>label=United States</td>
```

  これを以下のように変更する必要があります。

```
<tr>
   <td>runScript</td>
   <td>window.scrollTo(0,810)</td>
   <td></td>
</tr>
<tr>
   <td>click</td>
   <td>id=select2-country-container</td>
   <td></td>
</tr>
<tr>
   <td>click</td>
   <td>xpath=(//ul[@id='select2-country-results']/li[text() = 'United States'])</td>
   <td></td>
</tr>
```

- 制限事項: Selenium IDE 3.2.X または 3.3.X で記録した .side スクリプトがサポートされます。**linkText** ロケーターはサポートされません。

シンセティック・トランザクションの管理
Synthetic Script Manager を使用して、シンセティック・トランザクションの作成、構成、および削除を行います。

Synthetic Script Manager を表示するには、「**システム構成**」アイコン をクリックし、「**Synthetic Script Manager**」を選択します。Synthetic Script Manager で作業するには、「**Synthetic Script Manager**」および「**エージェント構成**」の表示権限を持つ役割のメンバーである必要があります。詳しくは、1003 ページの『役割と権限』を参照してください。

外部向けパブリック Web アプリケーションの当月のシンセティック・トランザクションの使用状況に関するデータを表示できます。Synthetic Script Manager の「月次再生使用量」表で、実行された再生インスタンスと、現在の構成に基づいて予測された再生インスタンスの、当月の数を表示します。

重要 : 内部および外部のプライベート Web アプリケーション用にシンセティック・トランザクションを作成するには、モニター対象の Web アプリケーションが含まれているすべての場所に Synthetic Playback エージェントをインストールする必要があります。

Synthetic Script Manager では以下のタスクを実行できます。

- シンセティック・トランザクションの作成と編集。
- シンセティック・トランザクション変数の構成。
- シンセティック・トランザクションの削除。

シンセティック・トランザクションの作成と編集
Web アプリケーションのパフォーマンスおよび可用性についてのデータを表示するには、最初に Synthetic Script Manager でシンセティック・トランザクションを作成する必要があります。

このタスクについて
Synthetic Script Manager を使用して、シンセティック・トランザクションの作成、編集、および構成を行います。Synthetic Script Editor で Web アプリケーションの URL を入力して、シンセティック・トランザクション用の単純なスクリプトを生成します。複雑なユーザー・プロセスをシミュレートするために、

Synthetic Script Editor でシンセティック・トランザクションにシンセティック・スクリプトをアップロードします。その後、各ロケーションで一定の間隔でシンセティック・トランザクションが実行されるように構成します。

手順

トランザクションの作成、または既存のトランザクションの編集を行うには、以下のステップを実行します。

1. オプション: Synthetic Script Manager が表示されていない場合は、「**システム構成**」アイコン をクリックし、「**Synthetic Script Manager**」を選択します。
2. 新規トランザクションを作成するには、「**新規**」アイコン をクリックします。既存のトランザクションを編集するには、「**編集**」アイコン をクリックします。
3. 「**Synthetic Script Editor**」で「**スクリプトのアップロード**」タブをクリックして、「**トランザクション名**」テキスト・ボックスにトランザクション名を入力します。「**説明**」テキスト・ボックスに、トランザクションの説明を入力します。

   重要 : トランザクションには、24 時間以内に削除したトランザクションと同じ名前は付けないでください。トランザクションに最近削除したトランザクションと同じ名前を付けると、Application Performance ダッシュボードで、両方のトランザクションのデータがそのトランザクション名に誤って帰属します。
4. Web アプリケーションをテストする 単純なスクリプトを生成するには、「**テストする Web ページの URL の入力**」を選択し、URL を入力します。Synthetic Script Manager により、その URL に基づいて単純なシンセティック・スクリプトが生成されます。
5. 以前作成したスクリプト・ファイルをトランザクションに割り当てるには、「**スクリプト・ファイルのアップロード**」を選択します。「**スクリプトのアップロード**」をクリックして、ご使用のシステム上のスクリプトを参照します。スクリプトを選択して、「**開く**」をクリックします。

   重要 : シンセティック・スクリプト・ファイルは以下のいずれかのファイル・タイプである必要があります。

   - `.html`
   - `.zip`

   個々の単純なスクリプト (テスト・ケース) を `.html` ファイルとして保存します。テスト・ケースとテスト・スイートをまとめて `.zip` ファイルに圧縮します。
6. シンセティック・トランザクションの同時再生またはずらし再生を構成するには、「**スクリプトをスケジュールに入れる**」タブをクリックします。トランザクションをすべてのロケーションから同時に実行するには「**同時**」を選択し、異なるロケーションからのトランザクションを間隔ごとに実行するには「**ずらし**」を選択します。
7. スクリプトの実行頻度を選択するには、「**スクリプトをスケジュールに入れる**」タブをクリックします。「**間隔**」テキスト・ボックスをクリックし、Web アプリケーションをモニターする 頻度に基づいて、数値を入力します。`1` 分から `60` 分の間隔の長さを選択します。

   注 : 大きいスクリプトまたは複雑なスクリプトの実行には長い時間がかかることがあります。大きいスクリプトまたは複雑なスクリプトに対応した間隔の長さを選択してください。
8. スクリプトの再生ロケーションを選択するには、「**スクリプトをスケジュールに入れる**」タブをクリックしてから、スクリプトを実行するデータセンターのロケーションおよびエージェントのインストール・ロケーションを選択します。
9. シンセティック・トランザクションおよびサブトランザクションの応答時間しきい値を設定するには、「**拡張設定**」タブをクリックしてから、シンセティック・トランザクションをクリックして、すべてのサブトランザクションが表示されるように展開します。応答時間しきい値をダブルクリックして、値を入力します。`0` 秒から `3600` 秒の値を選択します。しきい値を設定しない場合は、`0` を入力します。デフォルトの応答時間しきい値は 10 秒です。

   注 : コマンドの実行にかかる時間の長さは、コマンドによって異なることがあります。テストするコマンドに適した応答時間しきい値を選択してください。Web ページが開くまでにかかる時間をテストするトランザクションの場合は、長い応答時間を選択してください。

10. トランザクションの作成または編集を完了するには、「**保存**」をクリックします。

タスクの結果
シンセティック・トランザクションを構成しました。シンセティック・トランザクションは、Synthetic Script Manager にリストされます。

次のタスク
シンセティック・トランザクションによって記録されたメトリックおよび KPI を Application Performance ダッシュボードで表示できます。また、トランザクションをコンポーネントとしてアプリケーションに 追加すること、およびそのアプリケーションに関連付けられているすべてのシンセティック・トランザクションを表示することもできます。

重要 : シンセティック・トランザクションを最初に追加したときに、Application Performance ダッシュボードの「**時間の経過にともなう可用性**」グループ・ウィジェットにブランク・スペースが表示される場合があります。このブランク・スペースは、サーバーがトランザクションの 最初の再生結果を受信するとすぐに消えます。

シンセティック・トランザクション変数の構成
シンセティック・スクリプトに保管されるユーザー名、パスワードなどの変数値を更新する際に、Synthetic Script Manager を使用すると、スクリプト・ファイルを編集する必要がありません。変数の値は、それぞれの再生ロケーションで固有なものにすることができます。Web アプリケーションがロケーションごとに異なる変数値を要求する場合は、シンセティック・スクリプトの変数を構成します。例えば、複数のロケーションで同じログイン詳細を使用することを Web アプリケーションが許可しない場合は、Synthetic Script Manager を使用して、ロケーションごとに異なるログイン詳細を指定します。Selenium-IDE プラグイン内の `store` コマンドを使用して、シンセティック・スクリプト内で変数を作成できます。

このタスクについて

このタスクでは、Synthetic Script Manager を使用して、シンセティック・スクリプト内に保管される変数を構成します。

手順

シンセティック・トランザクションの変数を構成するには、以下の手順を実行します。

1. Synthetic Script Manager が表示されていない場合は、「**システム 構成**」アイコン をクリックし、「**Synthetic Script Manager**」を選択します。リストからシンセティック・トランザクションを選択し、「**編集**」アイコン をクリックします。
2. シンセティック・トランザクションの再生ロケーションを選択します。
3. 「**拡張設定**」タブをクリックします。シンセティック・スクリプトに変数が含まれている場合は、「**別のロケーションの変数置換の構成**」ウィンドウでこれらの変数を編集できます。変数を編集するには、値をダブルクリックします。完了するには、「**保存**」をクリックします。

   例えば、以下のスクリプトには変数 *username* および *password* が含まれています。これらの変数の値 `user1` および `pass` は、Selenium-IDE 内の `store` コマンドを使用して保存されます。これらの変数は、2 つのロケーション (Dallas と San Jose) で同じ値を持ちます。

表 *251.* 変数があるスクリプトの例

コマンド	ターゲット	値
`store`	`user1`	*username*
`store`	`pass`	*password*
`type`	`id=j_username`	*${username}*
`type`	`id=j_password`	*${password}*

スクリプト変数の値が「**別のロケーションの変数置換の構成**」ウィンドウに表示されます。ロケーション `Dallas` での *username* の値を `user1` から `admin1` に変更して、シンセティック・トランザクションがロケーションごとに異なるログイン詳細を使用するようにします。

表 *252.* さまざまなロケーションでのスクリプト変数の値

ロケーション	ユーザー名	パスワード
San Jose	user1	pass
Dallas	admin1	pass

タスクの結果
シンセティック・トランザクションの変数を構成しました。これで、このシンセティック・トランザクションを使用して、各ロケーションで Web アプリケーションのパフォーマンスと 可用性をテストできます。

次のタスク
シンセティック・トランザクションによって記録されたメトリックおよび KPI を Application Performance ダッシュボードで表示できます。また、トランザクションをコンポーネントとしてアプリケーションに 追加すること、およびそのアプリケーションに関連付けられているすべてのシンセティック・トランザクションを表示することもできます。

シンセティック・トランザクション用の *URL* およびドメイン名のフィルタリング

Synthetic Script Manager を使用してテストのホワイトリストおよびブラックリストにルールを 追加し、特定の URL および ドメインへのアクセスを許可またはブロックします。

このタスクについて

テスト対象の Web アプリケーションの応答時間に寄与する依存関係およびリソースを制御できます。ブラックリストを使用して、指定したドメインからの要求をフィルターで除外し、測定される応答時間からこれらの要求を削除します。ホワイトリストを使用して、指定したドメインからの要求を包含し、測定される応答時間にこれらの要求を追加します。ブラックリストおよびホワイトリストを使用して、Web アプリケーションに関連付けられた依存関係 (サード・パーティー・メトリックなど) をフィルターで除外または包含します。

「**ブラックリスト**」フィールドには、指定された URL およびドメインへのアクセスをブロックするルールのリストが含まれます。

「**ホワイトリスト**」フィールドには、指定された URL およびドメインへのアクセスを許可するルールのリストが含まれます。

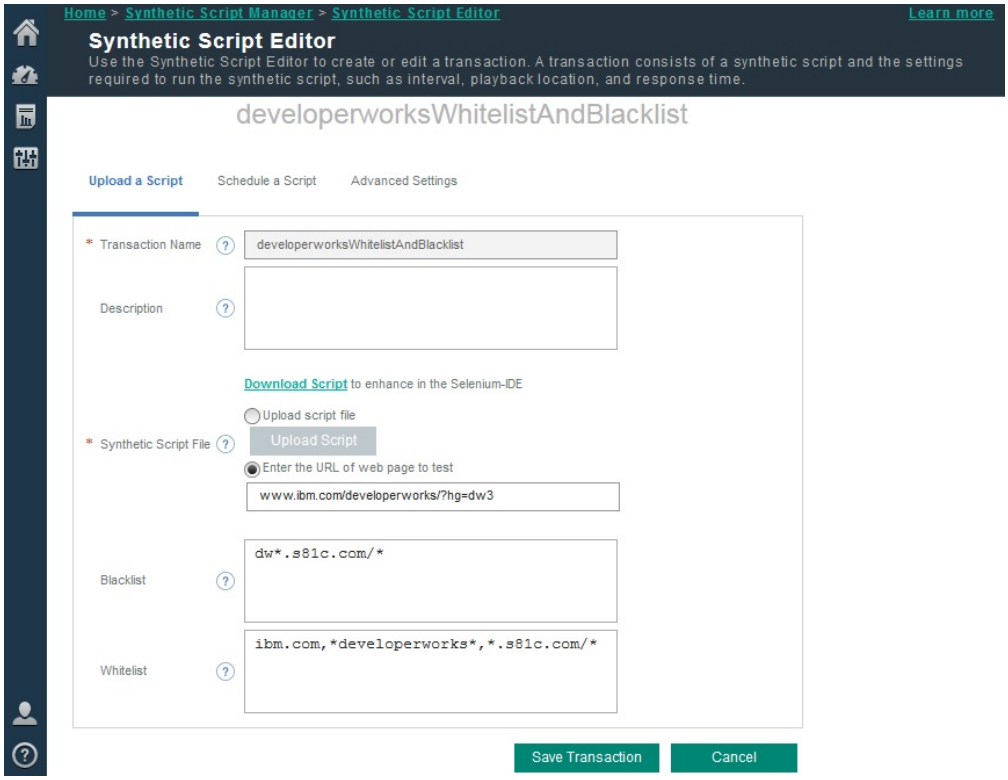
ブラックリストおよびホワイトリスト内のルールを区切るには、コンマ (,) を使用します。ドメイン名および URL をフィルタリングするには、ワイルドカード記号 (*) を使用します。例えば、`ibm.com, *.bluemix.net, *developerworks*, *.profile.*.cloundfront.net/*.png` のようにします。

注 : シンセティック・トランザクション用にブラックリストとホワイトリストの両方を構成する場合は、ブラックリストが優先されます。

手順

シンセティック・トランザクション用のブラックリストまたはホワイトリストにルールを追加するには、以下の手順を実行します。

1. Synthetic Script Manager が表示されていない場合は、「**システム 構成**」アイコン をクリックし、「**Synthetic Script Manager**」を選択します。
2. 既存のトランザクション用にブラックリストまたはホワイトリストを構成するには、「**編集**」アイコン をクリックします。トランザクションがリストされていない場合は、「**新規**」アイコン をクリックして新規トランザクションを作成します。
3. 「**Synthetic Script Editor**」で「**スクリプトのアップロード**」タブをクリックして、「**トランザクション名**」テキスト・ボックスにトランザクション名を入力します。「**ブラックリスト**」テキスト・ボックスおよび「**ホワイトリスト**」テキスト・ボックスにルールを追加します。



タスクの結果
シンセティック・トランザクションで Web アプリケーションに対する不要な要求がフィルターで除外され、他の特定の要求へのアクセスが許可されるようになりました。

Synthetic Script Manager でのパスワードの非表示
Synthetic Script Manager でパスワード値を非表示にするには、シンセティック・スクリプトに変数としてパスワードを格納します。

始める前に
この手順では、シンセティック・スクリプトを編集する必要があります。Selenium IDE を使用してシンセティック・スクリプトを記録してください。詳しくは、1029 ページの『シンセティック・スクリプトの記録』を参照してください。

このタスクについて

Selenium IDE で、変数としてパスワードを格納するように、シンセティック・スクリプトを手動で変更します。これで、Synthetic Script Manager で、非表示のパスワードのシンセティック・トランザクションを作成できます。 Synthetic Script Manager では、非表示のパスワードはアスタリスクとして表示されます。

重要 : パスワード値が Synthetic Script Manager に表示されないように、シンセティック・スクリプトにパスワードを格納することをお勧めします。非表示のパスワードにより、他のユーザーによるパスワードの表示を防止することで、Web アプリケーションが安全になります。

手順

1. Selenium IDE で、変更するスクリプトを開きます。`store` コマンドを使用して、この手順で説明している例に従って変数 *password* にパスワードを割り当て、次にスクリプトを保存します。

   重要 : パスワードは、Synthetic Script Manager に表示されないように、変数名 *password* として格納する必要があります。

例えば、次のシンセティック・スクリプトには、ユーザー名 *test@example.com* とパスワード値 *ibm4value* が含まれています。

```
<tr>
   <td>type</td>
   <td>id=username</td>
   <td>test@example.com</td>
</tr>
<tr>
   <td>type</td>
   <td>id=password</td>
   <td>ibm4value</td>
</tr>
```

次のスクリプトでは、`store` コマンドを使用して、パスワード値 *ibm4value* を変数 *password* に割り当てる方法を示しています。

```
<tr>
   <td>store</td>
   <td>ibm4value</td>
   <td>password</td>
</tr>
<tr>
   <td>type</td>
   <td>id=username</td>
   <td>test@example.com</td>
</tr>
<tr>
   <td>type</td>
   <td>id=password</td>
   <td>$(password)</td>
</tr>
```

2. オプション: スクリプト・レベルでパスワードを非表示にするには、`store` コマンドを使用して、変数 *password* にブランク値を割り当て、スクリプトを保存します。

   後で、Synthetic Script Manager でパスワードを設定できます。

   例えば、次のスクリプトでは、`store` コマンドを使用して、ブランク値を変数 *password* に割り当てる方法を示しています。

   ```
   <tr>
      <td>store</td>
      <td></td>
      <td>password</td>
   </tr>
   <tr>
      <td>type</td>
      <td>id=username</td>
      <td>test@example.com</td>
   </tr>
   <tr>
      <td>type</td>
      <td>id=password</td>
      <td>$(password)</td>
   </tr>
   ```

3. Cloud APM コンソールにログインし、**Synthetic Script Manager** を開きます。トランザクションを作成し、そのトランザクションに対してスクリプトをアップロードします。「**拡張設定**」タブをクリックします。

   それぞれの場所のパスワードが非表示になります。それぞれの場所のパスワードは、変更できます。 詳しくは、1033 ページの『シンセティック・トランザクションの管理』を参照してください。

シンセティック・トランザクションの削除

Synthetic Script Manager を使用してシンセティック・トランザクションを削除します。

手順

シンセティック・トランザクションを削除するには、以下のステップを実行します。

1. シンセティック・トランザクションがアプリケーションに割り当てられている場合、最初にアプリケーションからトランザクションを削除する必要があります。Application Performance ダッシュボードで、

「**すべてのマイ・アプリケーション**」をクリックして展開してから、削除するシンセティック・トランザクションに関連付けられているアプリケーションをクリックします。「**編集**」アイコン をクリックします。「**アプリケーションの編集**」ウィンドウで、アプリケーションからシンセティック・トランザクション・コンポーネントを削除します。詳しくは、『アプリケーションの管理』を参照してください。

これでシンセティック・トランザクションを削除できるようになりました。

2. ナビゲーション・バーで、「**システム構成**」アイコン をクリックし、「**Synthetic Script Manager**」を選択します。シンセティック・トランザクションを選択し、「**削除**」アイコン をクリックします。そのシンセティック・トランザクションを削除することを確定するには、「**OK**」をクリックします。

タスクの結果

シンセティック・トランザクションが削除されます。

Application Performance ダッシュボードでのシンセティック・トランザクション・データの表示

Application Performance ダッシュボードでシンセティック・トランザクション・データを表示します。シンセティック・トランザクションを新しいアプリケーションまたは既存のアプリケーションに関連付け、Application Performance ダッシュボードで、関連付けられたすべてのシンセティック・トランザクションをまとめて表示します。

このタスクについて

Application Performance ダッシュボードの「**マイ・トランザクション**」ウィンドウでシンセティック・トランザクション・データを表示できます。

シンセティック・トランザクションをアプリケーションに関連付けることで、シンセティック・トランザクションのグループを作成することもできます。Application Performance ダッシュボードで「**アプリケーションの追加**」ツールまたは「**アプリケーションの編集**」ツールを使用して、新しい Web アプリケーションまたは既存の Web アプリケーションにシンセティック・トランザクションをコンポーネントとして追加します。その後、Application Performance ダッシュボードで、そのアプリケーションに関連付けられているすべてのシンセティック・トランザクションのデータをまとめて表示できます。

応答時間エージェントを使用してアプリケーションのユーザー応答時間を既にモニターしている場合は、そのアプリケーションにシンセティック・トランザクションを追加できます。そうすることで、そのアプリケーションの追加メトリックおよび KPI を Application Performance ダッシュボードで表示できます。

手順

- シンセティック・トランザクションを表示するには、以下のステップを実行します。
  a)「**パフォーマンス**」アイコンをクリックし、「**Application Performance ダッシュボード**」を選択します。「**アプリケーション**」ウィンドウで、「**すべてのマイ・アプリケーション**」を展開し、「**マイ・トランザクション**」を選択します。「**グループ**」ウィンドウで、「**トランザクション**」を展開し、「**シンセティック・トランザクション**」を選択します。
  b) シンセティック・トランザクションをクリックして、そのトランザクションの可用性とパフォーマンスのデータを、定義した期間中のトランザクション・インスタンスの応答時間のグラフとともに表示します。
- シンセティック・トランザクションをアプリケーションに関連付けるには、以下のステップを実行します。
  a)「**パフォーマンス**」アイコンをクリックし、「**Application Performance ダッシュボード**」を選択します。既存のアプリケーションを選択するか、新規アプリケーションを作成および編集します。詳しくは、『アプリケーションの管理』を参照してください。
  b)「**アプリケーションの追加**」ウィンドウで、「**コンポーネントの追加**」アイコン をクリックし、コンポーネントのリストから「**シンセティック・トランザクション**」を選択します。「**コンポーネント・エディター**」ウィンドウで、シンセティック・トランザクションを選択し、「**追加**」をクリックして、シンセティック・トランザクションをアプリケーションに関連付けます。
  c)「**戻る**」をクリックします。「**閉じる**」をクリックして、「**コンポーネント・エディター**」ウィンドウを閉じます。「**保存**」をクリックします。別のシンセティック・トランザクションをコンポーネントとして追加するには、ステップ 1 から 3 を繰り返します。

タスクの結果
シンセティック・トランザクションをアプリケーションに関連付けました。これで、Application Performance ダッシュボードで、アプリケーションとそれに関連するシンセティック・トランザクションを表示できます。詳しくは、『アプリケーションの管理』を参照してください。

注 : シンセティック・トランザクションをアプリケーションに関連付けた時点では、そのアプリケーションの初期可用性は不明です。状況が更新されるまで数分かかることがあります。

シンセティック・イベントの管理
しきい値マネージャーとリソース・グループ・マネージャーを使用して、しきい値を構成し、それをシンセティック・トランザクションに割り当てます。シンセティック・イベントが生成されるのは、トランザクション属性の値が、しきい値に定義されている条件に一致した場合です。シンセティック・イベントは、Application Performance ダッシュボードでモニターできます。

シンセティック・トランザクションのしきい値の作成
しきい値マネージャーを使用して、シンセティック・トランザクション用のしきい値を作成します。しきい値は、属性値をしきい値での設定値と比較するために使用します。サンプリングした値が比較条件を満たすと、イベントが生成されます。

このタスクについて

しきい値により、ユーザーは、アプリケーションが特定の状態を報告した場合にモニターできます。例えば、Web サイトが特定のユーザー・コマンドに応答するのにかかる時間をモニターするためのしきい値を作成できます。Web サイトでかかった時間がしきい値に指定した時間よりも長くなると、シンセティック・イベントが生成されます。

手順

しきい値を作成して、1 つ以上のシンセティック・トランザクションにそれを関連付けるには、以下のステップを実行します。

1. ナビゲーション・バーで、「**システム構成**」アイコン をクリックし、「**しきい値マネージャー**」を選択します。**データ・ソース**・タイプを「**シンセティック・トランザクション**」に設定します。
2. しきい値を作成します。詳しくは、987 ページの『しきい値マネージャー』を参照してください。
3. しきい値をトランザクションに関連付けるには、「**データ・セット**」として「**KSO TRANSACTION**」を選択してから、「**表示項目**」として「**TRANSNAME**」を選択します。「**論理演算子**」には「**And (&)**」を選択します。

   注 : 「**表示項目**」として「**TRANSNAME**」を選択する必要があります。「**TRANSNAME**」を選択しないと、Application Performance ダッシュボードでシンセティック・イベントを表示できません。
4. 条件を追加するには、「**新規条件**」アイコン をクリックします。「新規条件」ボックスで、「**属性**」と「**演算子**」を選択します。次に、「**値**」にしきい値を入力します。この条件をしきい値に追加するには、「**OK**」をクリックします。

   例えば、低速なトランザクションの割合が 50% を超えた場合にシンセティック・イベントを生成するしきい値条件を追加するには、「**属性**」として「**PSLOW**」を選択してから、「**演算子**」として「**より大**」を選択します。イベントを生成するための低速トランザクションの割合を設定するために、「**値**」として 50 を入力します。
5. 追加のしきい値属性を定義するには、しきい値にさらに条件を追加します。
6. 終了したら、「**保存**」をクリックします。リソース・グループにしきい値を割り当てない場合は、「**OK**」をクリックします。

タスクの結果
これで、しきい値を作成して、それをシンセティック・トランザクションに関連付けました。しきい値条件が満たされると、イベントが生成されます。イベントは、Application Performance ダッシュボードの「**イベント**」タブでモニターできます。

次のタスク
シンセティック・トランザクションをリソース・グループにグループ化できます。

シンセティック・トランザクションのリソース・グループの作成

シンセティック・トランザクションをリソース・グループに編成し、そのリソース・グループ内のすべてのトランザクションにしきい値を適用します。

始める前に

リソース・グループ内のすべてのシンセティック・トランザクションに適用するしきい値を作成します。

このタスクについて

シンセティック・トランザクションをリソース・グループに編成し、そのリソース・グループ内のすべてのシンセティック・トランザクションにしきい値を適用できます。リソース・グループ・マネージャーを使用してリソース・グループを作成し、そのリソース・グループにしきい値を割り当てます。その後、そのリソース・グループに 1 つ以上のシンセティック・トランザクション・サブノードを割り当てます。これで、リソース・グループに関連付けられたしきい値が、関連するすべてのシンセティック・トランザクションに適用されます。

手順

シンセティック・トランザクションのリソース・グループを作成するには、以下のステップを実行します。

1. 「**システム構成**」アイコン をクリックし、「**リソース・グループ・マネージャー**」を選択します。リソース・グループを作成するか、既存のリソース・グループを編集します。詳しくは、982 ページの『リソース・グループ・マネージャー』を参照してください。

シンセティック・トランザクションのリソース・グループを作成するには、以下のステップを実行します。

2. リソース・グループの名前と説明を指定します。「**しきい値割り当て**」テーブルでリソース・グループにしきい値を割り当て、「**保存**」をクリックします。リソース・グループ・マネージャーで、リソース・グループを再度選択し、「**編集**」アイコン をクリックします。
3. 「**リソース割り当て**」テーブルで、シンセティック・トランザクション・サブノードにリソース・グループを関連付けて、「**保存**」をクリックします。

   シンセティック・トランザクション・サブノードのフォーマットは、S0:*TransactionName* です。例えば、トランザクション open_webpage の場合、使用可能なサブノードの名前は S0:open_webpage となります。

タスクの結果

シンセティック・トランザクションをリソース・グループに編成し、そのリソース・グループ内のすべてのトランザクションにしきい値を適用しました。

同時シンセティック・トランザクションとずらしシンセティック・トランザクションのクリティカルしきい値の作成

しきい値マネージャーを使用して、同時シンセティック・トランザクションとずらしシンセティック・トランザクションのクリティカルしきい値を作成します。

このタスクについて

ずらしトランザクションが連続して失敗するか、すべての再生ロケーションで同時トランザクションが失敗したときに利用者に通知するしきい値を作成します。詳しくは、1033 ページの『シンセティック・トランザクションの作成と編集』を参照してください。

手順

ずらしトランザクションの再生インスタンスが失敗したときにイベントを作成するクリティカルしきい値を作成するには、次の手順を実行します。

1. しきい値マネージャーで、シンセティック・トランザクションのしきい値を作成します。詳しくは、1040 ページの『シンセティック・トランザクションのしきい値の作成』を参照してください。
2. しきい値マネージャーで、「**重大度**」として「**重大**」を選択し、しきい値の「**間隔 (HHMMSS)**」として 1 分を入力します。次の式を使用して、「**必要な連続サンプル数**」を決定します。

必要な連続サンプル数 = (再生間隔 * 予測される連続失敗数) - 1

例えば、モニター対象のシンセティック・トランザクションの再生間隔が 5 分で、再生の失敗が 8 回連続するのを検出する場合、「**必要な連続サンプル数**」には「(5 * 8) - 1 =」の計算結果 39 を設定する必要があります。

3. 条件を追加します。「新規条件」ボックスで、「**属性**」として「**ロケーション**」を選択し、「**演算子**」として「**等しい**」を選択して、「**値**」に None と入力します。2 つめの条件を追加し、「**PFAILED**」を 100 に設定します。しきい値を保存します。
4. ナビゲーション・バーで、「**リソース・グループ・マネージャー**」を開きます。リソース・グループを作成します。リソース・グループに 1 つ以上のずらしシンセティック・トランザクションを割り当て、次に、そのリソース・グループに手順 1 から 3 で作成したしきい値を割り当てます。リソース・グループを保存します。 詳しくは、1041 ページの『シンセティック・トランザクションのリソース・グループの作成』を参照してください。

複数のロケーションで同時トランザクションの再生インスタンスが失敗したときにイベントを作成するクリティカルしきい値を作成するには、次の手順を実行します。

5. しきい値マネージャーで、シンセティック・トランザクションのしきい値を作成します。詳しくは、1040 ページの『シンセティック・トランザクションのしきい値の作成』を参照してください。
6. しきい値マネージャーで、「**重大度**」として「**重大**」を選択し、「**間隔 (HHMMSS)**」として 1 分を入力します。「**必要な連続サンプル数**」を、モニター対象のトランザクションの再生間隔と同じ値に設定します。

   例えば、モニター対象のシンセティック・トランザクションの再生間隔が 5 分の場合、「**必要な連続サンプル数**」を 5 に設定します。
7. 条件を追加します。「新規条件」ボックスで、「**属性**」として「**ロケーション**」を選択し、「**演算子**」として「**等しい**」を選択して、「**値**」に None と入力します。2 つめの条件を追加し、「**PFAILED**」を 100 に設定します。しきい値を保存します。
8. リソース・グループを作成します。リソース・グループに、1 つ以上のシンセティック・トランザクションと新しいクリティカルしきい値を割り当てます。詳しくは、1041 ページの『シンセティック・トランザクションのリソース・グループの作成』を参照してください。

タスクの結果

ずらしシンセティック・トランザクションまたは同時シンセティック・トランザクションのクリティカルしきい値を作成しました。しきい値条件が満たされると、イベントが生成されます。イベントは、Application Performance ダッシュボードの「**イベント**」タブでモニターできます。

シンセティック・イベントの *E* メール通知の管理

リソース・グループ・マネージャーおよび IBM Alert Notification を使用して、アプリケーションのパフォーマンスがしきい値を超えた場合に E メール通知を生成します。

始める前に

シンセティック・イベントの E メール通知を構成するには、最初にサブスクリプション用に Alert Notification を有効にする必要があります。詳しくは、IBM Knowledge Center の『Alert Notification』を参照してください。

このタスクについて

リソース・グループ・マネージャーで、Alert Notification を使用して E メール通知を構成します。E メール通知は、アプリケーションのパフォーマンスが、リソース・グループに関連付けられているしきい値によって設定された条件を満たす場合に生成されます。

手順

E メール通知を管理するには、次の手順を実行します。

1. 「**システム構成**」アイコン をクリックし、「**リソース・グループ・マネージャー**」を選択します。リソース・グループを作成または編集します。

2. 「**しきい値割り当て**」テーブルでしきい値を選択し、「**保存**」をクリックします。次に、「**リソース割り当て**」テーブルで、シンセティック再生リソースまたはシンセティック・イベント・エージェント・リソースにリソース・グループを関連付けます。「**保存**」をクリックします。リソース・グループ・マネージャーで、リソース・グループを再度選択し、「**編集**」アイコン をクリックします。
3. Alert Notification を新しいタブで開始するには、リソース・グループ・マネージャーで「**E メール通知の構成**」をクリックします。Alert Notification で、ナビゲーション・バーの「**ユーザー**」をクリックして、E メール通知の受信者を作成または編集します。詳しくは、IBM Knowledge Center の『Alert Notification』を参照してください。
4. Alert Notification で、ナビゲーション・バーの「**通知ポリシー (Notification Policies)**」をクリックします。「**通知エディター (Notifications Editor)**」は、新規ポリシーを使用して自動的にプロビジョンされます。ポリシーの名前およびフィルターは、リソース・グループから派生されます。「**ルールの追加 (Add Rule)**」をクリックして、E メール通知がいつ送信されるかを決定する条件を定義します。詳しくは、IBM Knowledge Center の『Alert Notification』を参照してください。
5. E メール通知の構成を完了するには、「**保存**」をクリックします。ポリシーは、「**通知ポリシー (Notification Policies)**」タブ内のテーブルにリストされます。

ログ・ファイル・モニターについてエージェントとサーバーのパフォーマンスを最大化するためのガイドライン

OS エージェントと Performance Management サーバーのパフォーマンスを最大にするには、フォーマット (.fmt) ファイルに正規表現を定義する必要があり、さらに Cloud APM コンソールに報告されるログ・ファイル・モニター・イベントの数を制限する必要もあります。

.fmt ファイルに正規表現を定義するためのガイドライン

.fmt ファイルは、多くの CPU 処理を必要とする正規表現を使用します。エージェントとサーバーのパフォーマンスを向上させるには、次のガイドラインを使用することで、.fmt ファイル内の正規表現に照らし合わせてモニター・ソース・ログ内のレコードを検査するのにかかる時間を最小化します。

複数行パターンの使用を最小限にとどめます。

複数行パターンは、エージェントが、どれがレコードであるかを判別してから、一致するかどうかを判別するため、負荷が高くなります。複数行パターン、つまり「¥n」文字が含まれる正規表現や、「%n」トークンが含まれる TEC スタイル形式を使用するとき、エージェントは、まずモニター対象ファイルをさまざまなサイズのレコードに分割する必要があります。次に、エージェントは、フォーマット・ファイル内の表現に照らし合わせてレコードを検査する必要があります。この手順では、正規表現を 2 回検査する必要があるため、処理が遅くなります。単一行パターンを使用する場合、ファイルの各行がレコードであると想定されるため、処理は大幅に速くなります。

場合によっては、一部の行を無視して、パフォーマンスを向上させることも可能なことがあります。例えば、RAS1 トレース・ログの単一レコードを次に示します。

```
(4D66DACB.0001-1:RAS1,400,"CTBLD")
```

```
+4D66DACB.0001 Component: ira
```

```
+4D66DACB.0001 Driver: agent_fac:15/4114877.7
```

```
+4D66DACB.0001 Timestamp: Feb 24 2011 13:18:54
```

```
+4D66DACB.0001 Target: sos510amdx6-d
```

この例で、以下の行のみを処理する場合、

```
+4D66DACB.0001 Driver: agent_fac:15/4114877.7
```

以下のように単一行パターンを記述できます。

```
^¥+.*Driver: agent_fac:([0-9¥.¥/]+)$
```

この単一行パターンを使用すると、複数行形式の必要なしに重要な値が処理されます。論理レコードのその他の 4 行は、何にも一致しない単一行レコードとして扱われ、破棄されます。

フォーマット・ファイル内の表現を、モニター・ログへの出現の頻度でソートします。
エージェントは、ログから読み取るレコードごとに、フォーマット・ファイル内の表現と照らし合わせて、一致が見つかるまで検査します。ファイル内の最後の表現から開始し、上方向に検索します。一致が見つかると、検索を停止します。最も多くログに記録された表現が最後にリストされていると、その表現がログに記録されているときに検査されるのは、その表現のみになります。

フォーマット・ファイルに 100 個の表現がある場合、フォーマット・ファイルの最初にリストされている表現にログ・レコードが一致するときは必ず、エージェントが他の 99 表現も検査する必要があるため、処理が遅くなります。ログから読み取るレコードがフォーマット・ファイルのいずれのパターンにも一致しないときは、エージェントが一致しないと判断するまでにすべてのパターンに照らし合わせて検査する必要があります。この処理は遅く、負荷が高くなります。

正規表現にできるだけ多くの定数データを含めます。
例えば、次のエラーがログに返されるとします。

```
Disk error on device: /dev/sda1 Disk error on device: /dev/sdb2 yyy
```

次のように表現を記述できます。

```
^Disk.*: .*$
```

この表現でも一致を検出できますが、正規表現エンジンが、類似しているがコロンがないなど最終的には一致しない他の行について、より多くの可能性を考慮することが必要になります。

以下の表現の方が厳密であり、一致しないエラーについては正規表現エンジンが処理を停止します。

```

^Disk error on device: /dev/sd[a-b][0-9]$
```

不要な副次式を使用しないでください。
次に示す例の括弧内に表示されている副次式は、一致データとして返される値を使用することを、正規表現エンジンに通知するために使用されています。これらの副次式を使用すると追加の処理が発生します。返される値を使用しない場合は不要です。例えば、次のエラーがログに返されるとします。

```
write failure in writing to client 9.27.135.191. Error Broken pipe
```

次の正規表現をフォーマット・ファイルに含めた場合、最後にエラー・メッセージがキャプチャーされますが、返される値を使用しない場合、パフォーマンスに悪影響が及びます。

```
REGEX
WriteFailure

^write failure in writing to client (.*)¥. エラー
(.*)$

ClientAddr $1
CustomSlot1

END
```

グループ化の目的で、表現に括弧を使用します。
? 演算子を使用して、返される値をキャプチャーしないように正規表現エンジンに通知できます。つまり、**?** 演算子を使用して、返される値のみをグループ化できます。このグループ化により、パフォーマンスに良い影響があります。例えば、次のログ・データが返されるとします。

```
Login succeeded on the first attempt for user Bob.
Login succeeded on the third attempt for user Joe.
```

返される値の両方と突き合わせる場合、1 回目 (first) や 3 回目 (third) のログイン試行の扱いを考慮する必要があります。成功したログイン試行がいずれであったかや、成功したユーザーがいずれであったかを気にしない場合は、返される値をグループ化するために次の表現を含めることができます。

```
REGEX
LoginSuceeded

^login succeeded on the (?:[a-z]+) attempt for user ([A-Z][a-z]*)¥.$

UserName $1
CustomSlot1
END
```

可能な場合、表現で OR (|) 演算子を使用しないでください。
| 演算子を使用すると処理の負荷が大きくなります。| 演算子を使用すると、正規表現エンジンは、バックアップを実行し、最初に一致しなかった値を突き合わせようとします。この手順では、2 つの別個の表現よりも効率が悪くなります。例えば、次の表現があるとします。

```
REGEX DiskError
^.*disk error.*4|^.*disk failure.*4
END
```

次の 2 つの表現を使用する方がはるかに効率が良くなります。

```
REGEX DiskError
^.*disk error.*4
END

REGEX DiskError
^.*disk failure.*4
END
```

これらの表現は、同じ結果を返します。

重要 : これらの表現は、できるだけ多くの定数データを使用するというガイドラインには違反しており、| 演算子の問題を示す目的のみで記載しています。

あいまいな表現を使用しないでください。
あいまいな表現を使用すると、バックアップおよび異なる方法での表現との突き合わせを正規表現エンジンに強制することになります。詳しくは、Performance Tips を参照してください。

あいまいな表現は、長いレコードを多くの副次式に分割するために含められた表現の結果として発生する場合があります。このような問題の悪い例では、以下のように、式の中で 2 つの (.*) の間にスペースが入ります。

```
(.*) (.*)
```

この悪い例では、正規表現エンジンは、1 つのスペースで区切られた任意の表現である 2 つのストリングを検索します。ただし、* はスペースとも一致するため、正規表現エンジンは最初に出現したスペースを 1 つめの (.*) に割り当てる可能性があります。新たなスペースを検出しないまま入力レコードの末尾に到達した場合、正規表現エンジンはバックアップを実行し、このスペースを、表現で要求されているリテラルのスペースとして使用して再試行しなければなりません。

パフォーマンスを向上させるには、具体的な表現のみを使用してください。Regex Pal ツールを使用して、定義するフォーマット・ファイルがモニター・ログに適合しているかどうかを確認できます。詳しくは、Regex Pal を参照してください。

報告されるログ・ファイル・イベントを制限するためのガイドライン

次のガイドラインにより、OS エージェントや Cloud APM サーバーのパフォーマンスを低下させる可能性があるログ・ファイル・イベントが制限されます。

.fmt ファイルには、具体的な形式で記述します。
.fmt ファイルには、関連するレコードを返せる具体的な形式で記述します。例えば、Error: で始まる行などの特定のエラーに対してイベントを生成し、Warning: で始まる行は無視するようにできます。

```
Error: disk failure
Error: out of memory
WARNING: incorrect login
```

.conf ファイル内の Unmatchlog 設定をオンにしないでください。
.conf ファイル内の **Unmatchlog** 設定をオンにすると、すべての不一致ファイルをログに記録し、ファイル・システムが過負荷になるため、オンにしないでください。

.fmt ファイルで *DISCARD* イベント・クラスを指定します。
.fmt ファイルで、データを意図的に破棄する事前定義 *DISCARD* イベント・クラスを指定して、エージェントの CPU 使用率を制限してみてください。*DISCARD* イベント・クラスを使用すると、.fmt ファイル内のパターンに一致するログ・レコードに対してはイベントが作成されません。例を以下に示します。

```
REGEX *DISCARD*
```

長期間が対象になる場合は重複イベント検出をオンにします。
.conf ファイルで次のキーを使用することで、重複イベント検出をオンにできます。

- DupDetectionKeyAttributes
- EventSummaryInterval
- EventFloodThreshold

次の例では、*msg* 値と *CustomSlot1* 値によって重複行が認識されます。

```
DupDetectionKeyAttributes=msg,CustomSlot1
EventSummaryInterval=300
EventFloodThreshold=send_first
```

重複イベントが多数ある場合は、イベントに *send_first* しきい値または *send_none* しきい値を適用します。詳しくは、1019 ページの『重複イベントの検出およびフィルタリング』を参照してください。

特定のしきい値条件を記述します。
しきい値と突き合わせる行のセットを制限する特定のしきい値条件を記述します。例えば、次のしきい値式を使用すると、FileSystemUsage イベント・クラスのイベントの値が CustomInteger1 内で 95 以上である場合にのみ、しきい値が発効するようになります。

```
( Class == 'FileSystemUsage' AND CustomInteger1 >= 95)
```

エージェントに対して .conf ファイルと .fmt ファイルの正しいセットを指定します。
エージェントに対して .conf ファイルと .fmt ファイルの正しいセットを指定するようにします。例えば、Windows OS エージェントのログ・ファイル・モニターを構成する場合、Windows OS エージェント専用に作成した .conf ファイルと .fmt ファイルを構成するようにします。

MongoDB アラーム・データベースを照会して、オープン・イベントの数またはイベント・レートを判別します。

- MongoDB アラーム・データベースを照会して、オープン・イベントの数またはイベント・レートを判別するには、次の手順を実行します。
  1. アラーム・データベースに対する MongoDB 照会で、event-query.js ファイルを作成します。例えば、次のようにします。
     - 次の照会は、指定したしきい値名のすべてのオープン・イベントとクローズしたイベントをカウントします。

       ```
       UDB_DB_Pool_Hit_Rat_Pct_Crit_2 db.alarms.count
       ```

       ```
       ({"threshold_name" : "UDB_DB_Pool_Hit_Rat_Pct_Crit_2"})
       ```

     - 次の照会は、MongoDB 内のオープン・イベントとクローズしたイベントをカウントします。

       ```
       db.alarms.count()
       ```

  2. コマンド **/opt/ibm/mongodb/bin/mongo 127.0.0.1:27000/alarm -u user -p mongoUsrpasswd@08 <event-count.js** を実行して、この照会に対する結果を event-query.js ファイル内に取得します。

- ログ・モニターに指定する CPU の量を制限します。詳しくは、639 ページの『ログ・ファイル・モニター環境変数』を参照してください。

Availability Monitoring

IBM Cloud Availability Monitoring を使用して、Web アプリケーションでのエンド・ユーザーの行動を模倣するシンセティック・テストを作成、編集、表示、および削除できます。

Availability Monitoring ダッシュボードには、モニター対象アプリケーション、URL、および REST API の可用性および応答時間に関する情報が表示されます。このダッシュボードを使用して、さまざまな場所にあるアプリケーション、URL、または REST API に関連付けられたアラートやアクティビティーをグラフ、明細表、およびマップ・ビューを使用してモニターします。

Availability Monitoring は、Availability Monitoring アドオンを使用する IBM Cloud Application Performance Management, Advanced on Cloud オファリングのユーザーが使用できます。

Availability Monitoring について

Availability Monitoring を使用して、さまざまなパブリック・ロケーションおよびプライベート・ロケーションから、Web アプリケーションの可用性とパフォーマンスを常にモニターするシンセティック・テストを作成します。

Availability Monitoring を使用してテストを作成します。定義した間隔と選択した場所で実行されるように、テストを構成します。ローカル・サーバーまたはプライベート・サーバーに独自のカスタム Point of Presence (PoP) をダウンロードしてデプロイします。以下のロケーションの 15 箇所のパブリック PoP からテストを実行します。

アジア
チェンナイ、香港、シンガポール、東京

オーストラリア
メルボルン

ヨーロッパ
アムステルダム、フランクフルト、ロンドン、パリ

中央アメリカ
メキシコ

北アメリカ
ダラス、サンノゼ、トロント、ワシントン D.C.

南アメリカ
サンパウロ

テストを作成および構成したら、Availability Monitoring ダッシュボードでアプリケーションの可用性とパフォーマンスのデータを表示できます。

Availability Monitoring の主な機能は以下のとおりです。

5 分未満で開始
Web アプリケーションのパフォーマンスと 可用性をモニターする単一アクション・テストを簡単に数分で作成できます。

アップタイムとユーザーの 満足度を最大化
複数の地理的場所からアプリケーションのアップタイムと 応答時間を頻繁にモニターします。シンセティック・テストを実行して、Web サイトのロードと API 呼び出しのパフォーマンスを測定します。さまざまな場所からユーザー・フローを模倣するために使用する Selenium スクリプトをモニターします。

事前予防
ユーザーに影響が及ぶ前に、問題を警告する通知を受け取ります。統合されたアラート通知サービスを使用して、アラート・ノイズを削減するアラート・ポリシーを作成できます。

障害の理由を正確かつ迅速に特定

ウォーターフォール式分析を使用して、障害が発生した場合の正確な手順や、障害の理由を特定できます。例えば、リンク切れ、サイズが大きすぎるイメージ、低速なルックアップ、または外部要求などを特定できます。画面キャプチャーが自動的に作成されるため、ブラウザーの障害や過去のパフォーマンスの問題を診断するのに役立ちます。テストに関する月次、週次、および日次の可用性および応答時間平均のレポートをダウンロードします。

Availability Monitoring で作業するには、モニター対象アプリケーションの表示権限を持つ役割のメンバーである必要があります。詳しくは、1003 ページの『役割と権限』を参照してください。

Availability Monitoring へのアクセス

アプリケーションの Availability Monitoring の「**状況の概要**」タブには、テストの可用性と状況に関する要約情報が表示されます。Availability Monitoring ダッシュボードには、Availability Monitoring の「**状況の概要**」タブからアクセスできます。

このタスクについて

Application Performance ダッシュボードの「**すべてのマイ・アプリケーション**」ペインで適格なアプリケーションをクリックして、Availability Monitoring 要約ページにアクセスします。この要約ページから、アプリケーションのテストを追加したり、アプリケーションの既存のテストを表示したり、Availability Monitoring ダッシュボードを表示したりできます。

手順

Availability Monitoring にアクセスするには、以下の手順を実行します。

1. 「**パフォーマンス**」アイコン をクリックし、「**Application Performance ダッシュボード**」をクリックします。
2. 「**すべてのマイ・アプリケーション**」ペインで、モニターするアプリケーションをクリックします。次に、「**グループ**」ペインで「**可用性モニター**」をクリックします。

   アプリケーションがリストされていない場合、アプリケーションを作成する必要があります。「**テンプレート**」として「**カスタム・アプリケーション**」を選択してください。詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

   Availability Monitoring の「**状況の概要**」タブには、過去 24 時間の平均テスト可用性、すべてのテストの現在のテスト状況、および現在の計画に対する割り振りのサービス使用状況を示す 3 つのゲージが表示されます。

   「**可用性の計算**」に含めるテストを選択し、Availability Monitoring で平均テスト可用性が計算される方法を構成できます。「平均テスト可用性」ゲージで矢印アイコン をクリックし、テスト・カードを表示します。次に、「**可用性の計算**」をクリックします。カードをクリックして、計算に追加したり、計算から削除したりします。除外されたテスト・カードは色が薄くなります。完了したら、「**完了**」をクリックします。「平均テスト可用性」ゲージが最新表示されます。

   「現在のテストの状況」ゲージで矢印アイコン をクリックして、すべてのテストの状況を表示できます。自分のテスト用のテスト・カードが表示されます。カードをクリックすると、その特定のテストの「**明細**」ダッシュボードにアクセスできます。
3. 「**モニター詳細の参照**」をクリックして、Availability Monitoring ダッシュボードにアクセスし、アプリケーションのすべてのテストに関するデータを表示します。「**すべてのテストを表示**」をクリックして、「**シンセティック・テスト**」ペインのテストを表示および編集します。「**新規テストの追加**」をクリックして、テストを作成します。

   注 : Availability Monitoring を初めて実行する場合、Availability Monitoring ダッシュボードでデータを表示するには、事前にテストを追加しておく必要があります。

テストの作成および構成

Web アプリケーションの可用性およびパフォーマンスを報告するテストを作成および構成します。

複数の地理的場所からアプリケーションの可用性および応答時間を頻繁にモニターするためのテストを作成および構成します。シンセティック・テストを実行し、Web サイトのロードと REST API 呼び出しのパフォーマンスを測定します。さまざまな場所からユーザー・フローを模倣する Selenium スクリプトを実行およびモニターするためのスクリプト動作テストを作成します。

REST API テストの作成

REST API テストを作成し、GET、POST、PUT、および DELETE などの HTTP メソッドを使用して、Web アプリケーションの応答時間と可用性をテストします。

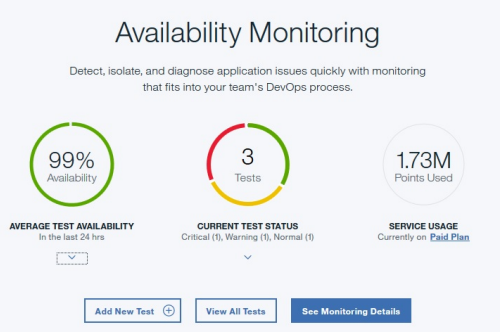
このタスクについて

REST API テストを使用して、REST 呼び出しへの応答で、Web アプリケーションとその他の URL の可用性とパフォーマンスをモニターします。

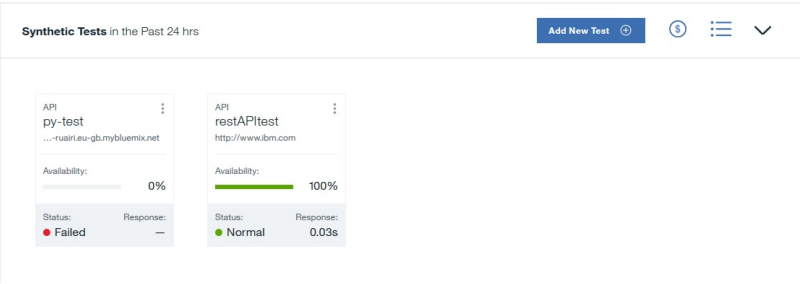
手順

REST API テストを作成するには、以下のようにします。

1. アプリケーションの Availability Monitoring 要約ページを表示している場合は、「**新規テストの追加 (Add New Test)**」をクリックします。



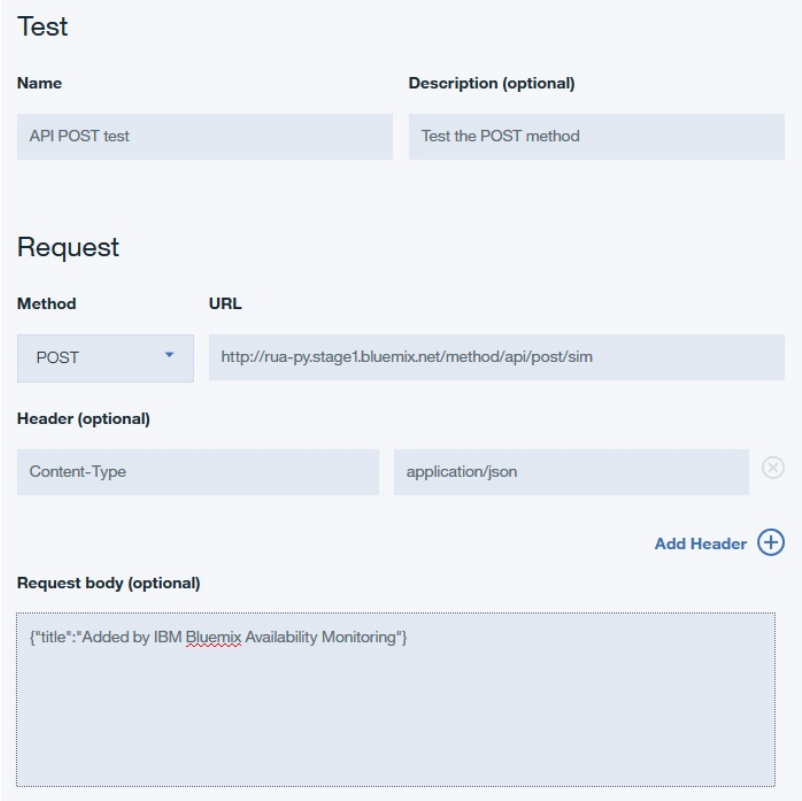
   Availability Monitoring ダッシュボードを表示している場合は、「**シンセティック・テスト**」ペインで「**新規テストの追加**」をクリックします。



2. 「**モニターのセットアップ**」ページの「**単一アクション**」をクリックし、「**単一アクション**」ページの「**REST API**」をクリックします。
3. 「**名前**」フィールドに、テストのわかりやすい名前を入力します。「**説明**」フィールドに、テストの目的の説明を追加します。
4. 「**要求**」セクションで、「**メソッド**」リストからメソッドのタイプを選択し、そのメソッドでテストする **URL** を入力します。

   「**GET**」、「**PUT**」、「**POST**」、または「**DELETE**」を選択できます。「**PUT**」または「**POST**」メソッドを選択する場合、「**要求本文 (オプション)**」フィールドにテストする 本文コンテンツを入力できます。

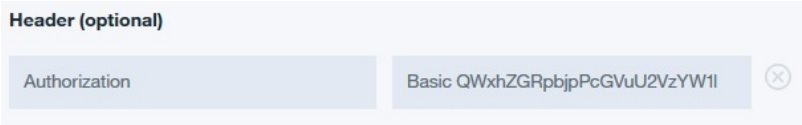
   例えば、以下の REST API テストでは POST メソッドを使用して、Web アプリケーションでデータを受け入れ、さらにその Web アプリケーションの可用性とパフォーマンスをテストするように要求します。



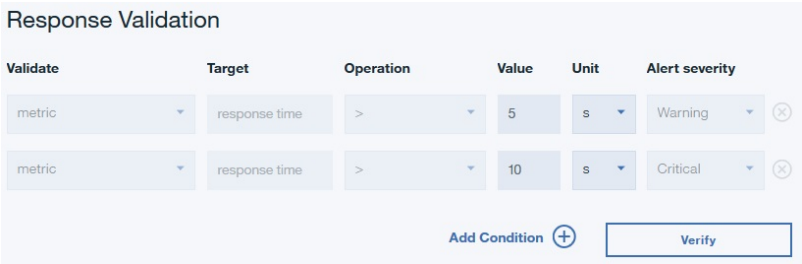
5. オプション: 特定のヘッダーと値が含まれるように、テストを構成します。「**ヘッダー**」フィールドに、ヘッダーの名前とヘッダーの値を入力します。

   テストする Web アプリケーションでユーザーのログインとパスワードが必要な場合、「**ヘッダー名**」フィールドに "Authorization" と入力します。「Basic」という単語、スペース文字、および *username:password* の base64 エンコード値を「**ヘッダー値**」フィールドに入力します。

   例えば、ユーザー名が *Aladdin* であり、パスワードが *OpenSesame* の場合、「Basic」という単語、スペース文字、および *Aladdin:OpenSesame* の base64 エンコード値を「**ヘッダー値**」フィールドに入力します。



6. 「**応答検証**」セクションでテストの警告アラートおよびクリティカル・アラートのしきい値を構成します。各行の「**値**」と「**単位**」を編集します。
   警告しきい値およびクリティカルしきい値を応答時間が超過すると、アラートがトリガーされます。



7. オプション: 「**条件の追加**」をクリックして、カスタマイズされた応答検証条件を定義および追加します。
   カスタマイズされた応答検証条件は集約して評価され、アラートが生成されます。テストに対して、最大で 6 つのカスタマイズされた条件を定義して追加できます。

   重要:

Availability Monitoring では、各テストによって合計で最大 3 つのアラートを生成できます。アラートを出す条件がすべて解決されるまで、最高の重大度を持つアラートがテストで報告されます。詳しくは、1060 ページの『Availability Monitoring でのアラート生成』を参照してください。

次のデータを検証できます。

ヘッダーの応答コード
「**ヘッダーの応答コード**」を選択し、1 つまたは特定の範囲の HTTP 応答コードをテストします。

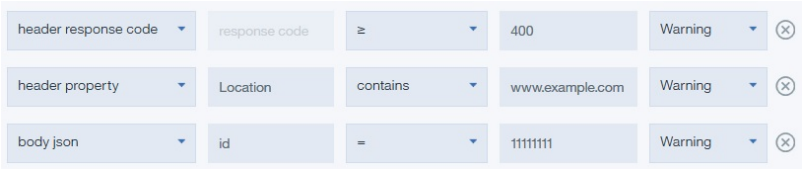
ヘッダーのプロパティー
「**ヘッダーのプロパティー**」を選択し、特定の HTTP ヘッダー・フィールドのプロパティーと 値をテストします。

本文 JSON
「**本文 JSON**」を選択して、JSON 本文の特定のプロパティーをテストします。

条件ごとに、「**ターゲット**」フィールドにテストするプロパティーを 入力し、「**値**」フィールドにテストする値を入力します。「**操作**」ドロップダウン・メニューから、演算子を選択します。最後に、条件に対する警告またはクリティカルの「**アラートの重大度**」を選択します。

重要 :

「**値**」フィールドに入力した数値は、デフォルトでストリングではなく数値として扱われます。応答検証条件の「**値**」を入力する場合、引用符 ("") を使用してストリングと数字を区別してください。例えば、ストリング 123 に対してテストする場合、「**値**」フィールドに "123" と入力します。数字 400 に対してチェックするには、引用符を使用せずに 400 と入力します。



8. 「**検証**」をクリックして REST API テストを作成し、テスト要求が有効かどうかを判別します。

   Availability Monitoring は、選択した HTTP メソッドと、テスト用に定義したすべての要求ヘッダーを使用してテストの有効性を判別します。テスト検証中は、応答検証は行われません。

   検証済みのテストは、「**検証済みの項目**」表に表示されます。ステップ 3 から 8 を繰り返して、さらに URL を追加できます。
9. テスト設定を構成するには、「**次へ**」をクリックします。

   テスト構成の要約が表示されます。デフォルト設定では、次のメッセージが表示されます。

   次のテストが実行されます: 15 分ごとに、同時に 3 箇所のパブリック・ロケーションおよび 0 箇所のプライベート・ロケーションから、テストが指定したしきい値を超過するかどうかが判別されます。
10. 「**設定**」ペインで、「**編集**」をクリックしてテストの現在の設定を表示します。
    以下の設定を更新できます。
    - 「**間隔**」は、テストの実行頻度を定義します。
    - 「**テスト頻度 (Testing frequency)**」は、テストをすべての場所から同時に実行するか、または間隔ごとに異なる場所から実行するかを決定します。「**同時**」を選択すると、すべての場所から同時にテストが実行されます。「**ずらし**」を選択すると、間隔ごとに、選択した異なる場所からテストが実行されます。
    - 「**ロケーション**」は、テストを実行するロケーションを決定します。
11. 「**パブリック・ロケーション**」のリストからロケーションを選択します。テストの実行元のプライベート・ロケーションを選択するには、テストの実行元のマシンに予めプライベート PoP をインストールして構成しておく必要があります。詳しくは、1056 ページの『プライベート PoP ロケーションのインストールおよび構成』を参照してください。
12. 「**保存**」をクリックしてテストの構成を完了します。次に、「**完了**」をクリックします。

Availability Monitoring ダッシュボードが表示されます。しばらくすると、ダッシュボードに新しいテストの情報とデータが表示されます。

Web ページ・テストの作成

Web アプリケーションの可用性をテストし、そのページを開くために要する時間をモニターするための Web ページ・テストを作成します。

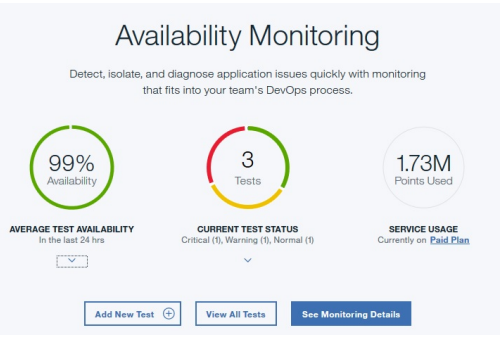
このタスクについて

Web ページ・テストは、Web アプリケーションの URL をロードするための応答時間をレポートします。Web アプリケーションの可用性と応答時間をモニターするための Web ページ・テストを作成します。

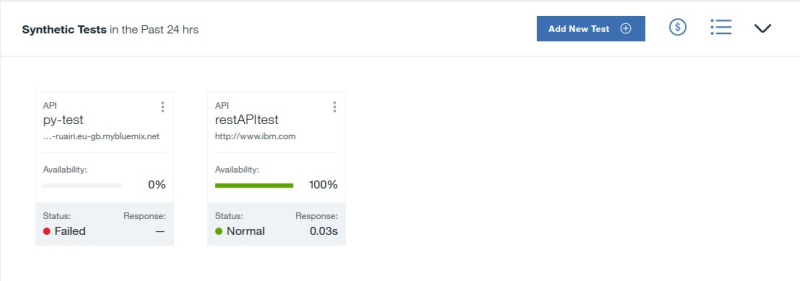
手順

Web ページ・テストを作成するには、以下の手順を実行します。

1. Availability Monitoring 要約ページを表示している場合は、「**新規テストの追加 (Add New Test)**」をクリックします。

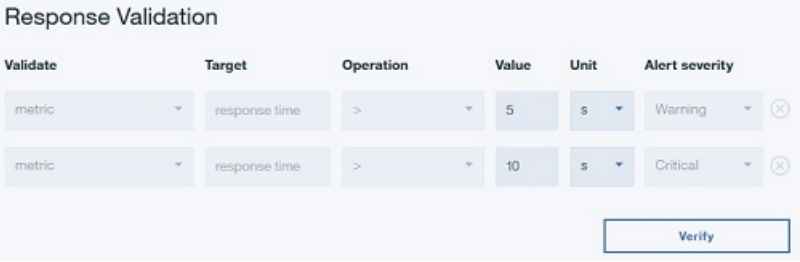


   Availability Monitoring ダッシュボードを表示している場合は、「**シンセティック・テスト**」ペインの「**新規テストの追加**」をクリックします。



2. 「**モニターのセットアップ**」ページで「**単一アクション**」をクリックし、「**単一アクション**」ページで「**Web ページ**」をクリックします。
3. 「**名前**」フィールドに、テストのわかりやすい名前を入力します。「**説明**」フィールドに、テストの目的の説明を追加します。
4. テストする Web アプリケーションの「**URL**」を入力します。
5. 「**応答検証**」セクションでテストの警告アラートおよびクリティカル・アラートのしきい値を構成します。各行の「**値**」と「**単位**」を編集します。

   警告しきい値およびクリティカルしきい値を応答時間が超過すると、アラートがトリガーされます。



6. 「**ブラックリスト**」と「**ホワイトリスト**」を使用して、要求の送信先の URL とドメイン、およびアプリケーション・テストのメトリックと状況に寄与する URL とドメインを指定します。包含する URL とドメインを「**ホワイトリスト**」に追加し、ブロックする URL とドメインを「**ブラックリスト**」に追加します。

   詳しくは、1055 ページの『ホワイトリストとブラックリストを使用したブロッキングとフィルタリング』を参照してください。
7. 「**検証**」をクリックして Web ページ・テストを作成し、テスト要求が有効かどうかを判別します。

   Availability Monitoring により、GET 要求がテスト URL に送信され、テストの有効性が判別されます。テスト検証中は、応答検証は行われません。

   検証済みのテストは、「**検証済みの項目**」表に表示されます。ステップ 3 から 7 を繰り返して、さらに URL を追加できます。
8. テスト設定を構成するには、「**次へ**」をクリックします。

   テスト構成の要約が表示されます。デフォルト設定では、次のメッセージが表示されます。

   次のテストが実行されます: 15 分ごとに、同時に 3 箇所のパブリック・ロケーションおよび 0 箇所のプライベート・ロケーションから、テストが指定したしきい値を超過するかどうかが判別されます。

   現在のテスト構成に基づいて、月ごとのテストの使用量と数の推定値が表示されます。
9. 「**設定**」ペインで、「**編集**」をクリックしてテストの現在の設定を表示します。

   以下の設定を更新できます。

   - 「**間隔**」は、テストの実行頻度を定義します。
   - 「**テスト頻度 (Testing frequency)**」は、テストをすべての場所から同時に実行するか、または間隔ごとに異なる場所から実行するかを決定します。「**同時**」を選択すると、すべての場所から同時にテストが実行されます。「**ずらし**」を選択すると、間隔ごとに、選択した異なる場所からテストが実行されます。
   - 「**ロケーション**」は、テストを実行するロケーションを決定します。

   「**パブリック・ロケーション**」のリストからロケーションを選択します。テストの実行元のプライベート・ロケーションを選択するには、テストの実行元のマシンに予めプライベート PoP をインストールして構成しておく必要があります。詳しくは、1056 ページの『プライベート PoP ロケーションのインストールおよび構成』を参照してください。

   「**保存**」をクリックして、テストの構成を完了します。
10. 「**完了**」をクリックします。

    Availability Monitoring ダッシュボードが表示されます。しばらくすると、ダッシュボードに新しいテストの情報とデータが表示されます。

アップロードされたスクリプトからのスクリプト・テストの作成

Selenium スクリプトをアップロードして、シミュレートされたユーザー動作に応答する Web アプリケーションの可用性とパフォーマンスをテストするスクリプト・テストを作成します。

始める前に

スクリプト・テストを作成するには、最初に Selenium スクリプトを作成する必要があります。Selenium スクリプトの作成について詳しくは、『シンセティック・スクリプトの記録』を参照してください。

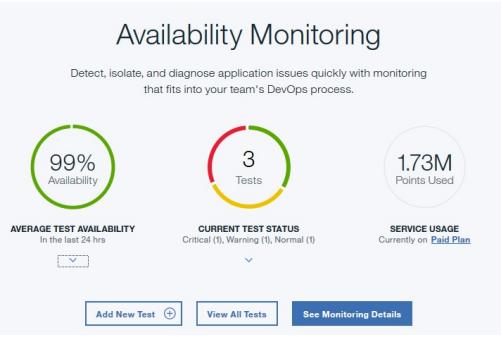
このタスクについて

Web アプリケーションとのユーザーの対話をシミュレートする Selenium スクリプトをモニターするスクリプト・テストを作成します。アプリケーションにログインするユーザーを 模倣する Selenium スクリプトを作成すると、スクリプト・テストを定期的に実行して、シミュレートされたユーザー動作に応答するアプリケーションのパフォーマンスをテストできます。

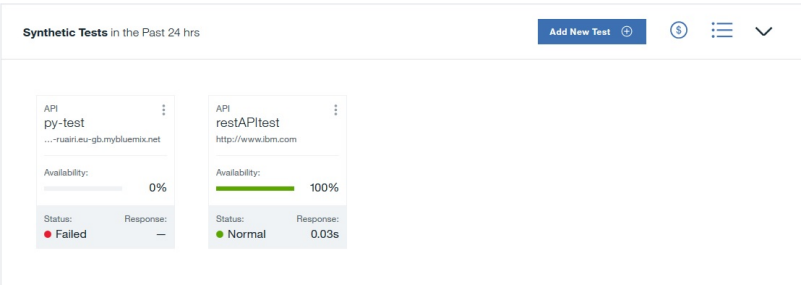
手順

スクリプト・テストを作成するには、以下の手順を実行します。

1. Availability Monitoring 要約ページを表示している場合は、「**新規テストの追加 (Add New Test)**」をクリックします。



   Availability Monitoring ダッシュボードを表示している場合は、「**シンセティック・テスト**」ペインの「**新規テストの追加**」をクリックします。



2. 「**モニターのセットアップ**」ページの「**スクリプト動作**」をクリックします。「**スクリプト動作のセットアップ**」ページが表示されます。「**ファイルのアップロード**」をクリックします。

   後でこのテストを編集するために戻る場合、アップロードされたスクリプト・ファイルをダウンロードできます。「**ダウンロード**」アイコン をクリックして、スクリプトをダウンロードします。
3. 「**名前**」フィールドに、テストのわかりやすい名前を入力します。「**説明**」フィールドに、テストの目的の説明を追加します。
4. 「**参照**」をクリックして、スクリプト・ファイルを見つけて、アップロードします。
5. 「**ブラックリスト**」と「**ホワイトリスト**」を使用して、要求の送信先の URL とドメイン、およびアプリケーション・テストのメトリックと状況に寄与する URL とドメインを指定します。包含する URL とドメインを「**ホワイトリスト**」に追加し、ブロックする URL とドメインを「**ブラックリスト**」に追加します。

   詳しくは、1055 ページの『ホワイトリストとブラックリストを使用したブロッキングとフィルタリング』を参照してください。
6. テスト設定を構成するには、「**次へ**」をクリックします。

   テスト構成の要約が表示されます。例えば、デフォルト設定では次のメッセージが表示されます。

   次のテストが実行されます: 15 分ごとに、同時に 3 箇所のパブリック・ロケーションおよび 0 箇所のプライベート・ロケーションから、テストが指定したしきい値を超過するかどうかが判別されます。

   現在のテスト構成に基づいて、月ごとのテストの使用量と数の推定値が表示されます。
7. 「**設定**」ペインで、「**編集**」をクリックしてテストの現在の設定を表示します。

   以下の設定を更新できます。

   - 「**間隔**」は、テストの実行頻度を定義します。
   - 「**テスト頻度 (Testing frequency)**」は、テストをすべての場所から同時に実行するか、または間隔ごとに異なる場所から実行するかを決定します。「**同時**」を選択すると、すべての場所から同時にテスト

が実行されます。「**ずらし**」を選択すると、間隔ごとに、選択した異なる場所からテストが実行されます。

- 「**クリティカルしきい値**」は、テストのクリティカル・アラートの応答時間を定義します。
- 「**警告しきい値 (Warning Threshold)**」は、テストの警告アラートの応答時間を定義します。
- 「**ロケーション**」は、テストを実行するロケーションを決定します。

デフォルトで表示される「**パブリック・ロケーション**」のリストからロケーションを選択します。テストの実行元のプライベート・ロケーションを選択するには、テストの実行元のマシンに予めプライベート PoP をインストールして構成しておく必要があります。詳しくは、1056 ページの『プライベート PoP ロケーションのインストールおよび構成』を参照してください。

必要に応じて、テスト・スクリプトに定義された変数の値を入力できます。例えば、スクリプトが Web サイトに接続するためにユーザー名とパスワードが必要な場合、それらの変数の値を入力できます。「**スクリプト変数**」表で、異なる場所の変数に異なる値を設定できます。

「**保存**」をクリックして、テストの構成を完了します。

8. 「**完了**」をクリックします。

Availability Monitoring ダッシュボードが表示されます。しばらくすると、ダッシュボードに新しいテストの情報とデータが表示されます。

ホワイトリストとブラックリストを使用したブロッキングとフィルタリング

ホワイトリストとブラックリストを使用して、要求の送信先のリソース、およびアプリケーション・テストのメトリックと状況に寄与するリソースを指定します。ホワイトリストとブラックリストは、Web ページおよびスクリプト動作のテストでのみ使用可能です。

「**ホワイトリスト**」フィールドと「**ブラックリスト**」フィールドでは、テストがアクセスできるリソースまたはアクセスできないリソース、およびテストのメトリックと状況に寄与するリソースを定義します。ホワイトリストとブラックリストは、サード・パーティーのメトリックなど、テスト対象の Web アプリケーションの応答時間に寄与する依存関係およびリソースを制御します。ホワイトリストとブラックリストは、Web ページまたはスクリプト動作のテストを作成する際に構成できます。

ホワイトリストを使用して、許可されるドメインと URL を定義できます。次に、**ブラックリスト**を使用して、許可されるロケーションの特定の要素をブロックします。

構文

ブラックリストとホワイトリスト内の項目を分離するには、コンマ (,) を使用します。各 URL またはドメインの要素をフィルターに掛けるには、ワイルドカード記号 (*) を使用します。

ホワイトリスト

要求およびメトリックの計算に含める URL、スキーマ、またはドメインを「ホワイトリスト」フィールドに追加します。ホワイトリストには、最大 10 個の項目をリストできます。各項目の長さは、200 文字を超えることはできません。ホワイトリストの項目に一致しないドメイン、スキーマ、および URL はすべてブロックされます。

例: `ibm.com, *developerworks*, *.s81c.com/*, https://www.ibm.com*, https://*`

注 : ホワイトリストの URL フィルターに `http://` または `https://` が含まれる場合、URL の直後にワイルドカード記号 (*) を含める必要があります (例: `https://www.ibm.com*`)。

ブラックリスト

要求およびメトリックの計算からブロックする URL、スキーマ、またはドメインを「ブラックリスト」フィールドに追加します。ブラックリストには、最大 20 個の項目をリストできます。各項目の長さは、200 文字を超えることはできません。

例: `*.profile.*.cloudfront.net/*.png, http://*`

注 : ブラックリストの URL フィルターに `http://` または `https://` が含まれる場合、URL の直後にワイルドカード記号 (*) を含める必要があります (例: `https://www.ibm.com*`)。

フィルタリングおよびブロッキングの動作

テストでは、ホワイトリストとブラックリストの両方を使用できます。どのロケーションが許可され、どのロケーションがブロックされるかを決定する際には、ブラックリストが常にホワイトリストより優先されます。以下の表は、ホワイトリストとブラックリストが関与するすべてのシナリオのフィルタリングとブロッキングの動作を示しています。

表 *253.* ホワイトリストとブラックリストのフィルタリングおよびブロッキングの動作

ブラックリスト	ホワイトリスト	動作	理由
空	空	アクセスを許可する	フィルター・ルールが入力されていない。
空	URL がリスト項目に一致しない	アクセスをブロックする	URL がホワイトリストに含まれていない。
空	URL がリスト項目に一致する	アクセスを許可する	URL がホワイトリストに含まれている。アクセスをブロックするためのブラックリスト項目がない。
URL がリスト項目に一致しない	空	アクセスを許可する	URL がブラックリストに含まれていない。ホワイトリストに含まれていない URL へのアクセスを防止するためのホワイトリスト項目がない。
URL がリスト項目に一致する	空	アクセスをブロックする	URL がブラックリストに含まれている。
URL がリスト項目に一致しない	URL がリスト項目に一致しない	アクセスをブロックする	URL がホワイトリストに含まれていない。
URL がリスト項目に一致しない	URL がリスト項目に一致する	アクセスを許可する	URL がホワイトリストに含まれている。URL がブラックリストに含まれていない。
URL がリスト項目に一致する	URL がリスト項目に一致しない	アクセスをブロックする	URL がホワイトリストに含まれていない。URL がブラックリストに含まれている。
URL がリスト項目に一致する	URL がリスト項目に一致する	アクセスをブロックする	URL がブラックリストに含まれている。ブラックリスト項目がホワイトリスト項目より優先される。

プライベート PoP ロケーションのインストールおよび構成

プライベート PoP をローカル・マシンにダウンロードしてインストールしてから、Availability Monitoring でのテストのロケーションとして使用するようにプライベート PoP を構成します。

始める前に

プライベート PoP をインストールするには、プライベート PoP のインストール・ロケーションが以下の要件を満たしている必要があります。

- Linux カーネル・バージョン 3.1.0 以上がインストールされている。
- Docker サービス・バージョン 1.7.1 以上がインストールされ、開始されている。

- 使用可能なディスク・スペースが 4 GB 以上ある。
- 使用可能メモリーが 2 GB 以上ある。
- CPU コア:
  - REST API の再生が必要になるのがプライベート PoP のみである場合は、少なくとも 2 つの CPU コアが使用可能である必要があります。
  - プライベート PoP で Web ページの再生およびスクリプトの再生を実行する場合は、1 つまたは 2 つのテストを 1 分ごとに実行するために 1 つの CPU コアが必要です。
- 新しいテストの追加前後には、プライベート PoP の CPU とメモリーの使用量を確認してください。また、更新版の Firefox または Selenium IDE を含むプライベート PoP のソフトウェア更新を適用した後も、新しいバージョンでシステム要件が引き上げられている可能性があるため、同様に確認してください。

どの時点で CPU コアを追加するか判別するためのベスト・プラクティスは、最も負荷の高いプロセスをプライベート PoP ホストで実行して CPU 使用量とメモリー使用量を調べることです。総 CPU 使用率が 70% を超え、かつ最も CPU 使用量の多いプロセスが Firefox である場合は、総 CPU 使用率が 50% 以下になるまで CPU コアを追加してください。プライベート PoP ホストの空きメモリーが 500 MB 以下の場合は、メモリーを増加させてください。

それ以上ハードウェア・リソースを用意できないが、例外を発生させずにプライベート PoP を実行したい場合は、以下の手順を実行して、Firefox の並列実行インスタンス数を減らしてください (現状のハードウェア・リソースでは多くのテストを実行できないため、UI で構成した間隔より長い間隔でテストが実行されるようになります)。

1. `start-pop.sh` スクリプトを編集して **`MAX_TASKPOOL_SIZE`** 環境変数を追加し、プライベート PoP ホストで使用可能な CPU コアの数を値として入力した後、**`stop-pop.sh`**、**`start-pop.sh`** の順に実行します。
2. UI で、テスト間隔の設定を長くします。

プライベート PoP のインストール先となるマシンのコマンド行インターフェース (CLI) へのユーザー権限が必要です。Docker にパッケージを追加するためのユーザー権限も必要です。

重要 :

- プライベート PoP を実行するマシンのシステム時刻が標準時刻と同期されており、その同期が維持されていることを確認してください。そうでない場合は、テスト・インスタンスが Availability Monitoring ダッシュボードに誤ったタイム・スタンプを表示します。
- Availability Monitoring のプライベート PoP が完全にサポートされるプラットフォームは Red Hat Enterprise Linux 7.4 と CentOS Linux 7.4 です。

このタスクについて

Availability Monitoring でテストを作成または編集するときに、パブリック・ロケーションに加えて、プライベート Points of Presence (PoP) をデプロイできます。プライバシーまたはセキュリティーの要件が高いアプリケーションなど、企業のファイアウォール内に配置されたアプリケーションをテストするためにプライベート PoP を使用します。最大で 50 個のプライベート・ロケーションを Availability Monitoring に登録できます。プリチェック・スクリプトとプライベート PoP パッケージをダウンロードし、プライベート PoP を実行するマシンにそのスクリプトとパッケージを保存します。

手順

1. テストを作成するか、既存のテストを編集します。

   テストを作成するには、「**シンセティック・テスト**」ペインの「**新規テストの追加**」をクリックします。テストを編集するには、「**アクション**」 ⋮ をクリックしてから、「**編集**」をクリックします。テストを作成する場合、テストを構成して検証します。「**設定**」セクションで、「**編集**」をクリックします。

   詳しくは、1049 ページの『REST API テストの作成』、1052 ページの『Web ページ・テストの作成』、および 1053 ページの『アップロードされたスクリプトからのスクリプト・テストの作成』を参照してください。

2. 「**設定**」セクションの「**編集**」をクリックして「**ロケーション**」セクションを表示してから、「**プライベート・ロケーション**」をクリックします。以前のテストを編集する場合、「**ロケーション**」セクションの「**プライベート・ロケーション**」をクリックします。

   以前に 1 つ以上のプライベート PoP をインストール済みの場合、インストールされたすべてのプライベート PoP のリストが表示されます。インストールおよび構成されたプライベート PoP がない場合、Availability Monitoring によりプライベート PoP のセットアップのガイドが示されます。
3. 「**プリチェックのダウンロード**」をクリックして、プリチェック・スクリプトをテストの実行元のマシンに保存します。

   重要 : プライベート PoP をインストールするには、スクリプトをコマンド・ライン・インターフェース (CLI) から解凍して実行する必要があります。プライベート PoP のスクリプトとパッケージを別のマシンにインストールし、そのマシンの CLI を介してアクセスすることができます。プライベート PoP のスクリプトの処理中にブラウザーで Availability Monitoring を閉じたり、更新したりしないでください。さもないと、保存されていないテスト設定がすべて失われます。

   プライベート PoP の配置先のマシンの CLI を開きます。CLI で、プリチェック・スクリプトを保存した場所にナビゲートします。次に、以下のようにしてプリチェック・スクリプトを実行します。

   ```
   ./precheck.sh
   ```

   マシンでシェル・スクリプトを実行するために必要な権限があることを確認してください。

   プリチェック・スクリプトによりチェック結果が表示されます。環境がチェックに不合格だった場合、表示された要件を満たすようにマシンをアップデートしてください。
4. Availability Monitoring に戻り、「**パッケージのダウンロード**」をクリックしてパッケージを保存します。このパッケージをテストの実行元のマシンに移動します。CLI で、ダウンロード・パッケージを保存した場所にナビゲートしてから、次のコマンドを実行してパッケージを解凍します。

   ```
   tar -xvf Availability_Monitoring_PoP.tar
   ```

   ここで、*Availability_Monitoring_PoP.tar* は、ダウンロードしたプライベート PoP パッケージが含まれる `.tar` ファイルの名前です。
5. プライベート PoP を構成します。CLI から次のスクリプトを実行します。

   ```
   ./config-pop.sh
   ```

   プロンプトが表示されたら、プライベート PoP の以下の情報を入力します。

   - PoP 名
   - 国のロケーション
   - 市区町村のロケーション
   - PoP の緯度
   - PoP の経度
   - PoP の説明
6. 自己署名証明書を使用するサーバー、または既知の CA 証明書プロバイダーによって署名されていないサーバーに接続する REST API テストがある場合は、`.pem` ファイル形式のトラステッド CA 証明書をすべて `keyfiles` ディレクトリーに格納します。

   注 :

   - `.pem` 証明書ファイルが変更された場合は、プライベート PoP を再始動する必要があります。
   - テスト対象のサーバーは、TLS ハンドシェーク中にルート CA 証明書以外をすべて送信する必要があります。送信が行われない場合は、(可能であれば) サーバーの構成を訂正してください。それ以外の場合は、このステップに示すように、不足している証明書をすべて `keyfiles` ディレクトリーに追加できます。ただし、他のクライアントの実行内容が PoP テストに反映されない場合があります。
7. Web ページ・テストまたはスクリプト動作テストを実行するときに、プライベート PoP がプロキシー・サーバーを使用するように構成するには、以下のいずれかのオプションを入力します。

   重要 : 手動または自動のプロキシー構成を使用してプライベート PoP ロケーションから実行される REST API テストでは、このプロキシーは使用されません。プライベート PoP ロケーションから実行す

るときにプロキシー・サーバーを使用できるのは、Web ページ・テストまたはスクリプト動作のテストだけです。

no
テストの実行時にプロキシーを使用しないようにプライベート PoP を構成するには、no と入力します。

manual
テストの実行時に使用するプライベート PoP プロキシーのプロキシー IP アドレスとポート番号を手動で構成するには、manual と入力します。スクリプトは、「*IP* アドレス:ポート番号」の形式で表されたプロキシー・サーバーの IP アドレスとポート番号を要求します。「プロキシーなし」リストを作成して、ドメイン・エレメント、ホスト名、または IPv4 アドレス・エレメントをブロックすることもできます。プロンプトが表示されたら、1 つ以上のドメイン・エレメントまたは IPv4 アドレス・エレメントを入力します。各リスト項目は空白スペースまたはコンマ (",") で区切ります。ワイルドカード演算子 (*) はサポートされていません。

- ドメインまたはサブドメインをブロックするには、ドットで始まるドメイン・サフィックス (例: .example.org、example.org) を入力します。
- ネットワークをブロックするには、ブロックする IP アドレスの範囲を識別するための CIDR サフィックスを付けて IP アドレスを入力します (例: 10.0.0.0/8)。

pac
自動プロキシー構成 URL を使用するようにプライベート PoP を構成するには、pac と入力します。スクリプトからプロンプトが出されたら、自動プロキシー構成 URL を入力します。

プライベート PoP 設定が pop.properties ファイルに保存されます。

8. プライベート PoP を始動します。CLI から次のスクリプトを実行します。

   ```
   ./start-pop.sh
   ```

   プライベート PoP が実行されると、Availability Monitoring で検出できます。

9. Availability Monitoring に戻り、 「**ロケーションのリフレッシュ (Refresh Locations)**」をクリックして、新規プライベート PoP を探して表示します。
   当該プライベート PoP が表にリストされます。
10. プライベート PoP をテストのロケーションとして選択するには、そのプライベート PoP を含む表の行のチェック・ボックスを選択します。プライベート PoP を削除するには、以下のステップを実行します。
    a) CLI から、プライベート PoP が配置されているマシンで **./stop-pop.sh** スクリプトを実行します。
    b) Availability Monitoring に戻り、削除するプライベート PoP を含む表の行の 「**削除**」をクリックします。
11. ステップ 3 から 10 を繰り返して、さまざまなマシンにプライベート PoP を追加し、Availability Monitoring でロケーションとして選択できるようにします。「**終了**」をクリックして保存し、テストを開始します。
    Availability Monitoring ダッシュボードが表示されます。約 1 分後に、新しいテストの情報とデータがダッシュボードに表示されます。
12. オプション: 既存のプライベート PoP をアップグレードするには、以下のステップを実行します。
    a) 新しいプライベート PoP のパッケージを新規フォルダーにダウンロードします。次に、**tar -xvf** コマンドを使用して、新しい PoP をそのフォルダーにアンパックします。
    b) CLI から、以前のプライベート PoP が配置されているフォルダーにディレクトリーを 移動します。以下のスクリプトを実行して以前のプライベート PoP を停止します。

       ./stop-pop.sh

    c) 以前のプライベート PoP が配置されているディレクトリーで、既存の system.properties ファイルおよび pop.properties ファイルをバックアップします。

重要：system.properties ファイルには、プライベート PoP が Cloud APM サーバーに接続するための重要な情報が含まれます。pop.properties ファイルには、プライベート PoP の構成データが含まれます。この構成を維持したい場合は、プライベート PoP をアップグレードする前に、以前のプライベート PoP の pop.properties ファイルおよび system.properties ファイルを保存しておいてください。

d) 新しいプライベート PoP のフォルダーから、pop.properties ファイルおよび system.properties ファイルを除くすべてのファイルをコピーして、以前のプライベート PoP が配置されている場所のファイルと置き換えます。

e) アップグレード後のプライベート PoP を再構成する必要がある場合は、CLI から以下のスクリプトを実行します。

./config-pop.sh

f) CLI から **./start-pop.sh** を実行して、アップグレード後のプライベート PoP を開始します。

g) Availability Monitoring に戻り、 「**ロケーションのリフレッシュ (Refresh Locations)**」をクリックして、アップグレードしたプライベート PoP を探して表示します。

Availability Monitoring でのアラート生成

Availability Monitoring では、各テストで最大で合計 3 つのアラートを生成できます。アラートを出す条件が解決されるまで、最高の重大度を持つアラートがテストで報告されます。

3 つの異なるシチュエーションに応じて、別個のアラートが生成されます。

- Web アプリケーションまたは URL の応答時間が、テストに対して設定された警告しきい値または重大しきい値を超えた場合。各テストはデフォルトで応答時間を測定し、そのテスト用の警告しきい値またはクリティカルしきい値に基づいて、アラートを出します。
- クライアント・エラーまたはサーバー・エラーのために Web アプリケーションまたは URL が使用できないことを示す HTTP 応答コードがテストで返された場合。各テストはデフォルトで応答コードをチェックし、テストが成功するか失敗するかを判別します。
- 1 つ以上のカスタマイズ条件に合致することがテストで検出された場合。この場合は、その 1 つ以上のカスタマイズ条件で定義されているアラームのうち、最高の重大度を持つアラームが出されます。Availability Monitoring では、アラームを出すかどうかを決定する際に、すべてのカスタマイズ条件を集合体として考慮します。このアラームは、カスタマイズされたすべての条件によって、警告またはクリティカル・アラートがそれ以上生成されないことがテストで確定されるまで続きます。

複数のアラートが出されると、アラートが存在する間、Availability Monitoring は最高の重大度を持つアラートを報告します。

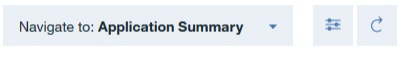
例えば、クリティカル・アラートを出す、カスタマイズされた条件を追加し、さらに、警告アラートを出す、カスタマイズされた別の条件を追加すると、テストによってクリティカル・アラートが生成されます。このアラートは、Availability Monitoring ダッシュボードに表示されます。クリティカル・アラートを出す条件が満たされなくなると、テスト・アラートの重大度は「警告」に変わります。アラートは、すべての条件でアラートがそれ以上発生しなくなるまで続きます。

モニター・ダッシュボードでのアプリケーションの可用性とパフォーマンスの表示

Availability Monitoring ダッシュボードで、アプリケーションの可用性とパフォーマンスの詳細と、アラートおよび関連付けられたテストを表示できます。

Availability Monitoring ダッシュボードは、以下のペインに分かれています。

- アプリケーション要約
- アラート頻度
- シンセティック・テスト
- 応答時間と可用性

「**ナビゲート先**」ドロップダウン・メニュー 

を使用して、任意のペインに素早く移動します。

Availability Monitoring の機能について学習するのに役立つガイドを利用してください。ガイドを開くには、「**ヘルプ**」アイコン をクリックし、表示したいガイドをクリックします。

ビデオ・チュートリアル・ライブラリー
ビデオ・チュートリアル・ライブラリーには、Availability Monitoring テストの作成方法、Selenium IDE を使用したテスト・スクリプトの作成方法、およびアラートの送信方法に関するビデオが用意されています。

Monitoring の概要
「Monitoring の概要」ガイドではダッシュボードの各領域に焦点を当て、Availability Monitoring の各機能が説明されています。

「アプリケーション要約」ペイン、「アラート頻度」ペイン、「シンセティック・テスト」ペイン、または「応答時間と可用性」ペインから「**明細**」ダッシュボードにアクセスできます。「**明細**」ダッシュボードには、テスト・インスタンスの主な統計情報が表示されます。

ニーズに合わせて、ペインの順序を変更できます。ペインを移動するには、見出しをクリックし、別の位置までペインをドラッグします。ログアウトした後もそれらの変更が維持されるように保存するには、「**レイアウトの保存**」をクリックします。

毎分自動的に最新表示されるように、ダッシュボードを設定できます。「**構成**」アイコン をクリックし、「**最新表示**」スライディング・バーをクリックして、「**1 分**」を選択します。すぐにページを最新表示するには、「**最新表示**」をクリックします。

アプリケーション要約

「アプリケーション要約」ペインには、過去 24 時間のアラート状況の概要および現在のテスト状況の情報が表示されます。

「アプリケーション要約」ペインには、次の情報が表示されます。

- 「**現在の状況**」には、すべてのテストの最も重大度が高い状況が表示されます。重大度は、「正常」、「警告」、「クリティカル」のいずれかです。
- 「**アラート**」には、オープン・アラートの数が表示され、警告アラートとクリティカル・アラートに分割されます。
- 「**可用性レポート (Availability Report)**」では、そのアプリケーションの月次、週次、または日次の可用性と応答時間平均の `.csv` ファイル・レポートをダウンロードできます。レポートをダウンロードするには、「**レポート**」アイコン をクリックします。

アラート頻度

「**アラート頻度**」パネルには、最近のアラートが表示されているマップが含まれています。アラートは場所ごとにグループ化され、「**アラート**」表にリストされます。

「アラート頻度」マップ

「**アラート頻度**」マップには、テスト用のすべてのパブリックおよびプライベートの Point of Presence (PoP) に関する情報が一目でわかるように表示されています。

ズーム機能を使用して、マップの特定の領域を拡大したり、その場所を元のサイズに戻したりします。各場所の上にカーソルを合わせると、その場所の名前、その場所の警告アラートおよびクリティカル・アラートの数が表示されます。「**アラート**」ドロップダウン・リストから「**すべて**」、「**オープン**」、または「**クローズ**」を選択して、マップに表示されるアラートをフィルタリングできます。

PoP の場所

「**PoP の場所**」アイコン

は、テスト用の PoP の場所を示します。各「**PoP の場所**」アイコンの色は、各場所の最近のアラートの重大度を示しています。正常

、警告

、クリティカル

で示されます。アニメーション表示されている「**PoP の場所**」アイコンは、テスト・インスタンスのすべての場所の中で重大度が最も高いアラートが最も多く発生している場所を示しています。

「**非アクティブな PoP の場所**」アイコン の上にカーソルを合わせて、「**ここをテスト**」をクリックし、選択したテストに PoP の場所を追加します。選択したテストに応じて、「**テスト編集モード**」ページが表示されます。「**応答時間**」および「**可用性**」ペインの「**テスト**」ドロップダウン・メニューから、テストを選択できます。

プライベート PoP の場所は、「**プライベート PoP の場所 (Private PoP location)**」アイコン によって表されます。

アラートの数

「**PoP の場所**」アイコンによって、場所ごとに生成されたすべてのアラート、オープン数、クローズ数が表示されます。「**クリティカル**」、「**警告**」、および「**正常**」アイコン

によって、場所の重大度ごとにアラート数が表示されます。

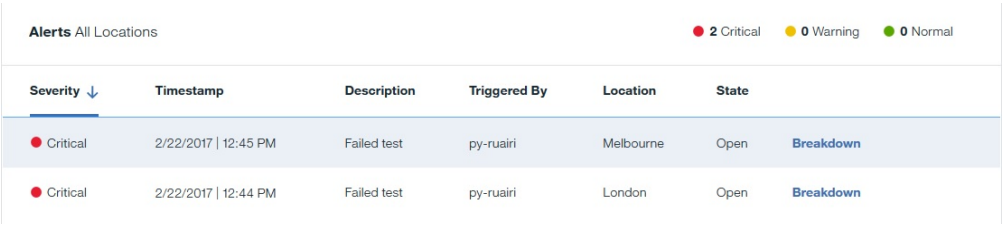
失敗したテスト

失敗したテストが発生した場所は、「**PoP の場所**」アイコンのボーダーに破線が表示されて示されます 。

ズーム機能を使用して、マップの特定の領域を拡大したり、その場所を元のサイズに戻したりします。各場所の上にカーソルを合わせると、その場所の名前、その場所の警告アラートおよびクリティカル・アラートの数が表示されます。「**アラート**」ドロップダウン・リストから「**すべて**」、「**オープン**」、または「**クローズ**」を選択して、マップに表示されるアラートをフィルタリングできます。

アラート表

すべての場所に関するアラートが 1 つの表に表示されます。



この表には、アラートに関する次の情報が表示されます。

- 「**重大度**」には、アラートが「クリティカル」または「警告」であるかが示されます。
- 「**タイム・スタンプ**」には、アラートが作成された時刻が表示されます。
- 「**説明**」には、テスト・インスタンスのパフォーマンスの 概要が示されます。
- 「**トリガー元**」には、アラートをトリガーしたテストの名前が表示されます。
- 「**ロケーション**」には、問題が発生した場所が示されます。
- 「**状態**」には、アラートがオープンとクローズのどちらであるかが 示されます。

アラート詳細の表示

表内の各アラートには、「**明細**」ダッシュボードへのリンクが含まれています。明細ダッシュボードを使用して、アラートを発生させた問題をトラブルシューティングするのに 役立てます。

アラートのフィルタリング

特定の場所に対してアラートをフィルタリングするには、マップ上の「**PoP の場所**」アイコンをクリックします。すべての場所に対してアラートを表示するには、「**PoP の場所**」アイコン 以外の、マップ上の任意の場所をクリックします。

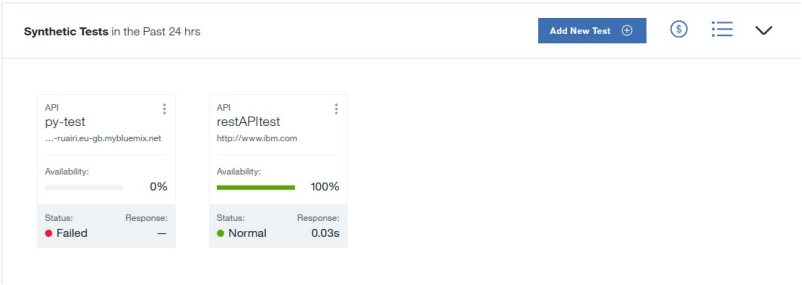
重大度によって表内のアラートをフィルタリングするには、「**クリティカル**」、「**警告**」、または「**正常**」アイコン 12 Critical 11 Warning 0 Normal をクリックします。フィルターを除去して、表に各重大度のアラートを含めるには、選択したアイコンをもう一度クリックします。

アラートのしきい値の変更

アラートは、テストを作成するときに指定するしきい値によってトリガーされます。大部分の場合、これらは可用性の問題や、応答時間が遅いことによって発生します。しきい値の設定を変更するには、「**シンセティック・テスト**」ペインでアラートを生成したテスト上の「**アクション**」アイコン をクリックし、「**編集**」をクリックします。

シンセティック・テスト

「**シンセティック・テスト**」ペインで、アプリケーションのパフォーマンスと可用性をモニターするシンセティック・テスト を作成、編集、削除、および表示できます。テストは、「**シンセティック・テスト**」ペインのリストまたはカード・ビューに表示されます。



各テスト・カードに、テストに関する以下の情報が表示されます。

可用性
過去 24 時間にわたるテストの可用性の割合がパーセンテージで表示されます。

状況
テストの現在の状況が表示されます。状況は、「重大」、「警告」、「正常」、「失敗」、または「不明」のいずれかになります。

平均 応答
過去 24 時間にわたるテストの平均応答時間が表示されます。

3 つの異なるタイプのテストをモニターできます。

REST API
REST 呼び出しの応答時間をレポートします。GET、POST、PUT、および DELETE など、すべての HTTP 要求の形式がサポートされています。

Web ページ
入力した URL の Web サイトをロードするための応答時間をレポートします。

スクリプト動作 (Scripted behavior)
Web サイトとのユーザーの対話を模倣するために作成する Selenium スクリプトをモニターします。例えば、アプリケーションにログインするユーザーを 模倣する Selenium スクリプトを作成できます。このスクリプトを定期的に実行して、スクリプトによって自動化されるユーザー・アクションに応答するアプリケーションのパフォーマンスをテストします 。Selenium スクリプトの作成について詳しくは、『1029 ページの『シンセティック・スクリプトの記録』』を参照してください。

別のテストを追加するには、「**新規テストの追加**」をクリックします。

シンセティック・テストを停止、開始、削除、または編集するには、「**アクション**」アイコン をクリックして、必要なアクションをクリックします。テストの「**明細**」を詳細に表示するには、テストをクリックします。

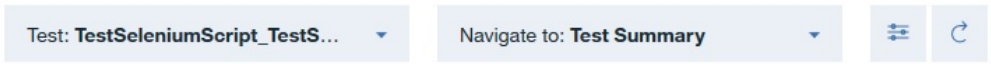
各シンセティック・テストの具体的な使用状況を表示するには、「**コスト (Cost)**」アイコン をクリックします。有料の計画をサブスクライブしている場合は、使用状況がデータ・ポイントに表示されます。

明細

「**明細**」ダッシュボードには、テストの主要な統計情報が表示されます。このダッシュボードには、過去 24 時間にわたる可用性および応答時間の情報、ヒストリカル・トレンド、およびテスト・パフォーマンス・データの概要も表示されます。

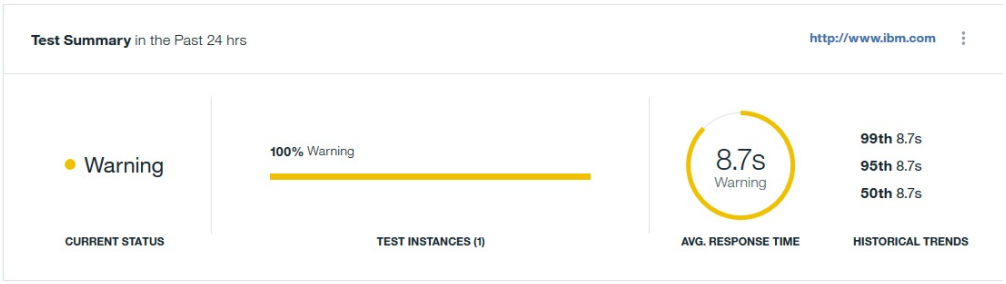
テストの詳細な明細を表示するには、「**シンセティック・テスト**」ペインでテストをクリックします。また、「アラート頻度」ペインの「**アラート**」表で「**明細**」をクリックして、「**明細**」ダッシュボードを開くこともできます。

「**テスト**」ドロップダウン・メニューを使用して各種テストの明細を表示します。「**ナビゲート先**」ドロップダウン・メニューを使用して、任意のペインに素早く移動します。



明細ダッシュボードには 4 つのペインが表示されます。

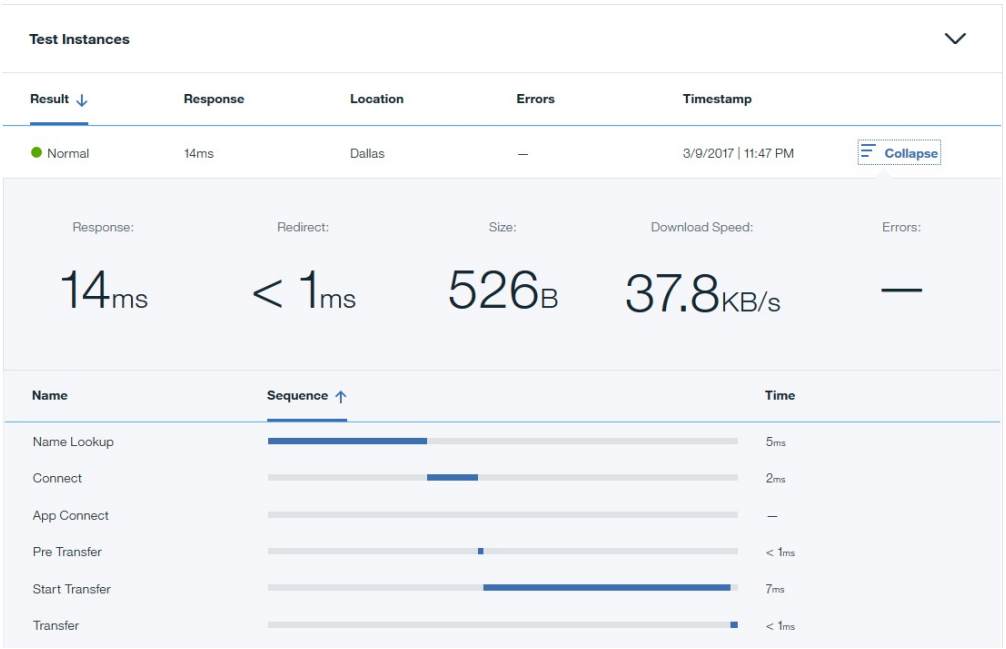
「**テストの要約**」



「**テストの要約**」ペインには、過去 24 時間分の以下のテスト情報が表示されます。

- 「**現在の状況**」には、テストの状況が表示されます。
- 「**テスト・インスタンス**」には、「正常」、「警告」、「クリティカル」のテスト・インスタンスのパーセント明細が表示されます。
- 「**平均応答時間**」には、テストの平均応答時間が表示されます。
- 「**ヒストリカル・トレンド**」には、秒単位またはミリ秒単位で、50、95、および 99 のパーセンタイルのテスト・パフォーマンスのヒストリカル・トレンドが表示されます。

テスト・インスタンス



「**テスト・インスタンス**」表には、状況、応答時間、テストが実行された場所、エラーの数、およびテスト実行時のタイム・スタンプなど、各テスト・インスタンスに関する詳細情報が表示されます。テスト・インスタンスにドリルダウンするには、「**展開**」をクリックします。テスト・インスタンスのステップごとに、対応する詳細情報が表示されます。列をソートして、スローダウンや障害が発生したステップを正確に、素早く特定できます。エラー、テスト・シーケンス、および応答時間を表示すると、問題を簡単に特定できます。

表示される情報は、モニターされているシンセティック・テストのタイプによって異なります。

API

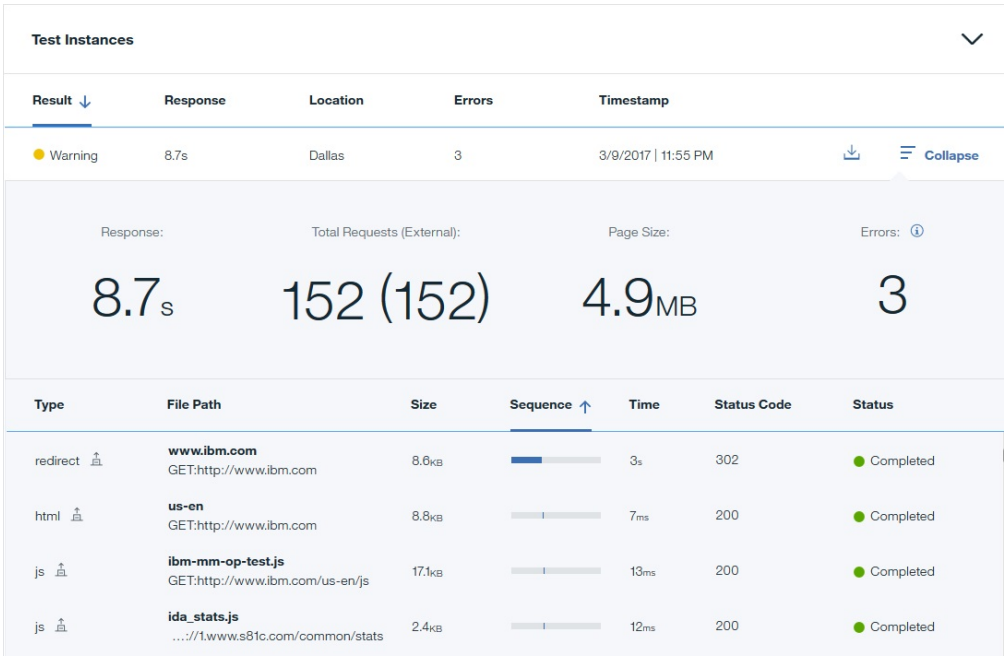
API テスト・インスタンスの「**展開**」をクリックすると、以下の詳細情報の全般的な概要が表示されます。

- 「**応答**」には、リダイレクト時間を含む、テスト・インスタンスの合計応答時間が表示されます。
- 「**リダイレクト**」には、テスト・インスタンスの合計リダイレクト時間が表示されます。
- 「**サイズ**」には、オブジェクトのサイズが 表示されます。
- 「**ダウンロード速度**」には、各オブジェクトのダウンロード速度が表示されます。
- 「**エラー**」には、テスト・インスタンス中に発生したエラー数が表示されます。エラーの詳細を確認するには、「**情報**」アイコンをクリックします。

表には、API 呼び出しの各ステップとともに、ステップの名前、ステップのシーケンス、および各ステップの応答時間が表示されます。以下のステップ名が表示されます。

- 「**名前のルックアップ**」には、オブジェクトの名前を解決するためにテスト・インスタンスが要した時間が表示されます。
- 「**接続**」には、ステップの開始からリモート・ホストまたはプロキシーへの接続が完了するまでにテスト・インスタンスが要した時間が表示されます。
- 「**アプリケーションの接続**」には、ステップの開始からリモート・ホストとの SSL 接続が完了するまでにテスト・インスタンスが要した時間が表示されます。
- 「**事前転送**」には、ステップの開始からファイル転送コマンドが開始する直前までにテスト・インスタンスが要した時間が表示されます。
- 「**転送開始**」には、ステップの開始から最初のバイトを受信するまでにテスト・インスタンスが要した時間が表示されます。
- 「**転送**」には、ファイルを転送するためにテスト・インスタンスが要した時間が表示されます。

Web ページ



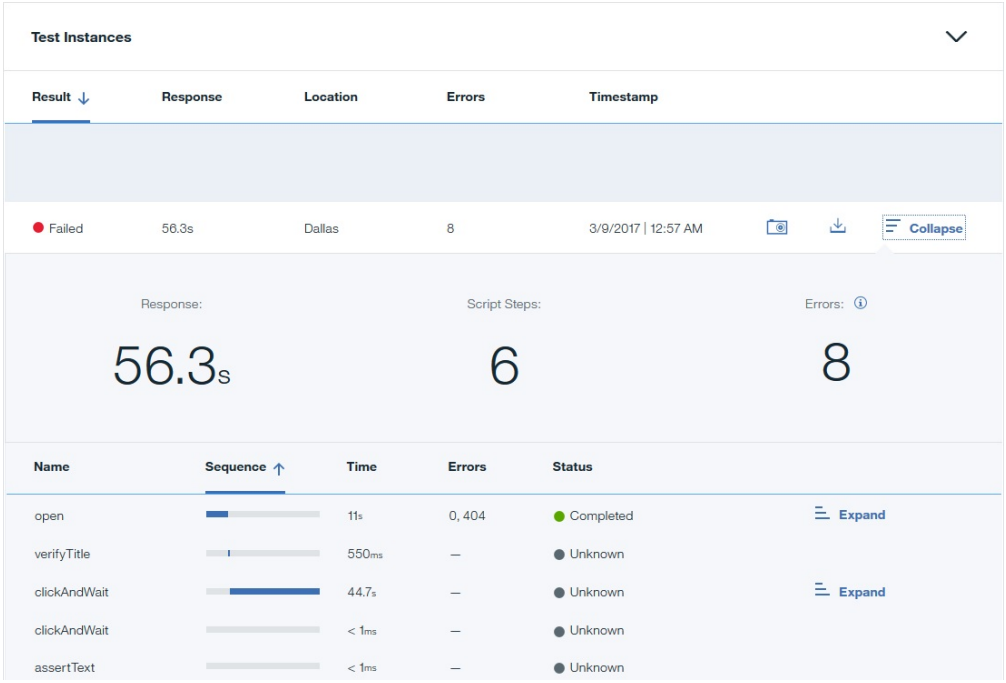
Web ページ・テスト・インスタンスの「**展開**」をクリックすると、以下の詳細情報の全般的な概要が表示されます。

- 「**応答**」には、テスト・インスタンスの応答時間が表示されます。
- 「**合計要求数 (外部)**」には、テスト・インスタンスの要求の合計数が表示されます。外部要求の数は、括弧の中に表示されます。
- 「**ページ・サイズ**」には、Web ページのサイズが表示されます。
- 「**エラー**」には、テスト・インスタンス中に発生したエラー数が表示されます。エラーの詳細を確認するには、「**情報**」アイコンをクリックします。

テストにより実行された各要求の以下の詳細がリストされた表も表示されます。

- 「**タイプ**」には、HTML、CSS、JavaScript、またはイメージなど、要求のタイプが表示されます。外部要求と内部要求がアイコンによって示されます。
- 「**ファイル・パス**」には、要求されたオブジェクトの場所が示されます。
- 「**サイズ**」には、要求されたオブジェクトのサイズが表示されます。
- 「**シーケンス**」には、テストによって実行された要求のシーケンスが表示されます。
- 「**時間**」には、各要求に要する時間が表示されます。
- 「**状況コード**」には、HTTP 要求の状況コードが表示されます。
- 「**状況**」には、「完了」、「不明」、または「失敗」など、要求の結果が示されます。

スクリプト



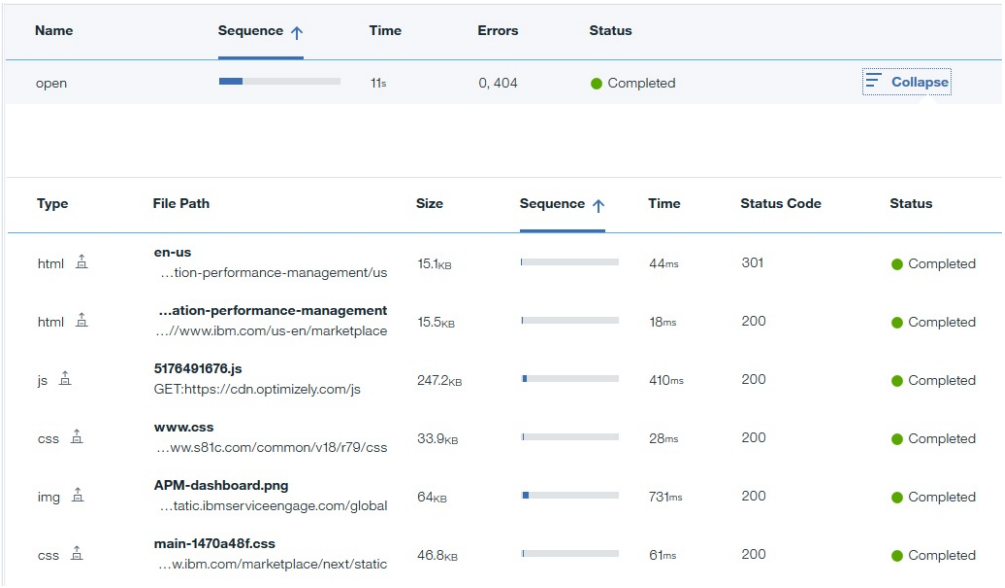
スクリプト・テスト・インスタンスの「**展開**」をクリックすると、応答時間、スクリプト・ステップの数、およびエラーの数が表示されます。エラーの詳細を確認するには、「**情報**」アイコンをクリックします。

各スクリプト・ステップに関する以下の詳細情報が表に表示されます。

- 「**名前**」には、`Open`、`ClickAt`、または `VerifyBodyText` など、テスト・インスタンスによって呼び出される各 Selenium コマンドが表示されます。
- 「**シーケンス**」には、テスト・インスタンスの最初から最後までのスクリプト・ステップのシーケンスが表示されます。
- 「**時間**」には、各スクリプト・ステップに要する時間が表示されます。
- 「**エラー**」には、各スクリプト・ステップ中に発生したエラー数が表示されます。

- 「**状況**」には、「完了」、「不明」、または「失敗」など、スクリプト・ステップの結果が示されます。

各スクリプト・ステップによって生成された要求にドリルダウンし、詳細を確認できます。



「**展開**」をクリックすると、以下の情報が含まれている表が表示されます。

- 「**タイプ**」には、HTML、CSS、JavaScript、またはイメージなど、要求のタイプが表示されます。外部要求と内部要求がアイコンによって示されます。
- 「**ファイル・パス**」には、要求されたオブジェクトの場所が示されます。
- 「**サイズ**」には、要求されたオブジェクトのサイズが表示されます。
- 「**シーケンス**」には、テストによって実行された要求のシーケンスが表示されます。
- 「**時間**」には、各要求に要する時間が表示されます。
- 「**状況コード**」には、HTTP 要求の状況コードが表示されます。
- 「**状況**」には、「完了」、「不明」、または「失敗」など、要求の結果が示されます。

Availability Monitoring により、Web ページでロードできなかったり、スクリプトでステップが失敗したりした場合に、画面キャプチャーを自動的に作成できます。例えば、スクリプトのステップの 1 つで Web ページが開かれたが、ロードされなかった場合、Availability Monitoring によって画面キャプチャーが自動的に作成されます。Web ページまたはスクリプトの画面キャプチャーを表示するには、「**スクリーン・ショット・エラー**」アイコン をクリックします。この機能は、Web ページとスクリプト・テストに対してのみ使用できます。REST API テストでは機能しません。

「**ダウンロード**」アイコン をクリックして、特定のテスト・インスタンスのネットワーク・トラフィックの記録を `.har` ファイルとしてダウンロードすることもできます。この機能は、Web ページとスクリプト動作テストに対してのみ使用できます。

応答時間と可用性

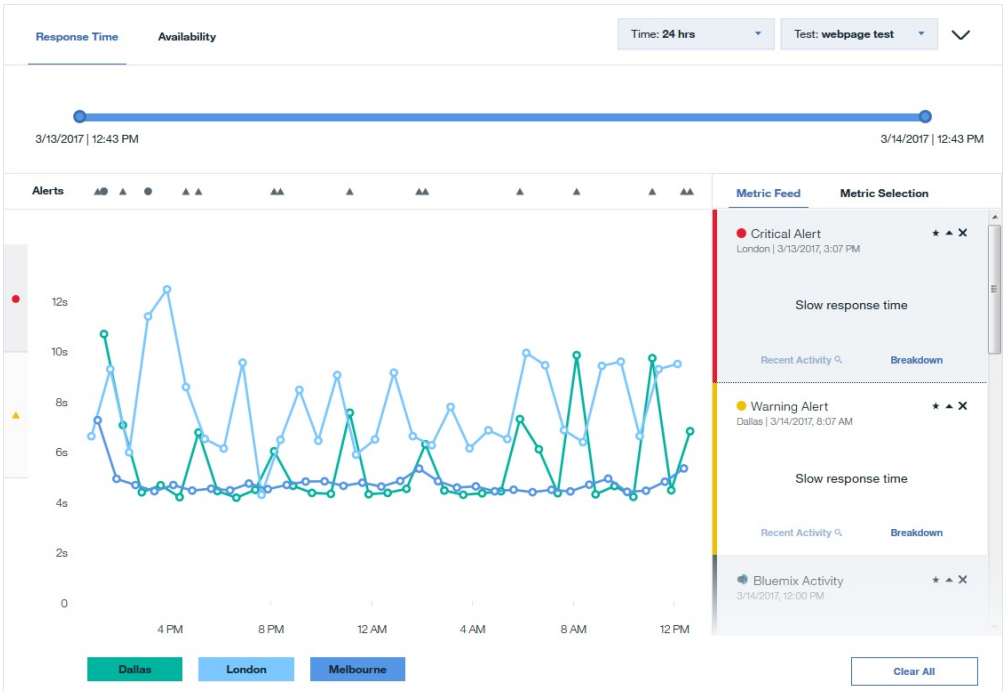
「**応答時間と可用性**」ペインには、定義された期間にわたるテスト・インスタンスの可用性と、測定された応答時間のグラフが表示されます。詳しくは、1068 ページの『応答時間と可用性』を参照してください。

応答時間と可用性

「**応答時間**」と「**可用性**」ペインを使用して、応答時間、可用性の傾向、およびアラートを時系列に視覚化できます。

応答時間グラフ

応答時間の情報は、折れ線グラフで表示されます。グラフを表示するには、「**応答時間**」タブをクリックします。



重要 : Availability Monitoring で測定される応答時間は、ユーザーが経験する応答時間よりも若干長くなります。Availability Monitoring は、実際のユーザーの行動をシミュレートするため、応答時間の測定にそれが追加されます。以下の要因で、応答時間が長くなります。

- 以前のテスト・インスタンスによって現在のテストに影響が及ぼされることを回避するために、Availability Monitoring によってテストごとに新しい Firefox インスタンスが作成されます。実際のユーザーは、ブラウザーのキャッシュにより、応答時間が短くなる可能性があります。
- Availability Monitoring によって、各テストの前に Firefox Web ドライバー・プラグインがインストールされます。

折れ線グラフ上に、テストの個々の応答時間が「**応答ポイント (Response point)**」アイコン で示されます。アプリケーションが実行されている地理的場所ごとに、異なる色で示されます。グラフの Y 軸では、アラート・アイコンを使用して警告しきい値およびクリティカルしきい値の範囲が示されます。黄色の警告アイコン は、警告しきい値の範囲を示し、赤色のクリティカル・アイコン は、クリティカルしきい値の範囲を示します。黄色の警告アイコン または赤色のクリティカル・アイコン をクリックすると、警告またはクリティカルしきい値範囲に表示されるテスト・インスタンスを容易に特定できます。特定のテスト・インスタンスの詳細を表示するには、グラフ上で「**応答ポイント (Response point)**」アイコン をクリックします。

フィルター

「**テスト**」ドロップダウン・メニューからテストを選択します。3 時間、24 時間、7 日間、30 日間、および 12 カ月分のデータをフィルタリングできます。24 時間を超える時刻範囲でフィルタリングする 場合、グラフに表示される値は平均値です。より詳細な情報を表示するには、グラフをクリックして、個々のアラートおよび警告にドリルダウンします。また、スライダーを使用して、時刻範囲を狭めたり、広げたりすることもできます。

「応答時間」グラフでは、特定の PoP ロケーションからのデータを強調表示したり、非表示にしたりできます。特定のロケーションの応答時間データを強調表示するには、PoP ロケーション名の上にポインターを移動して、「**ロケーションの強調表示 (Highlight location)**」アイコン をクリックします。特定のロケーションの応答時間データを非表示にするには、PoP ロケーション名の上にポインターを移動して、「**ロケーションを非表示にする (Hide location)**」アイコン をクリックします。PoP ロケーションのデータをグラフに復元するには、「**別のロケーションを追加 (Add more locations)**」 または「**メトリック選択**」タブをクリックして、以前に削除した PoP ロケーションをクリックします。

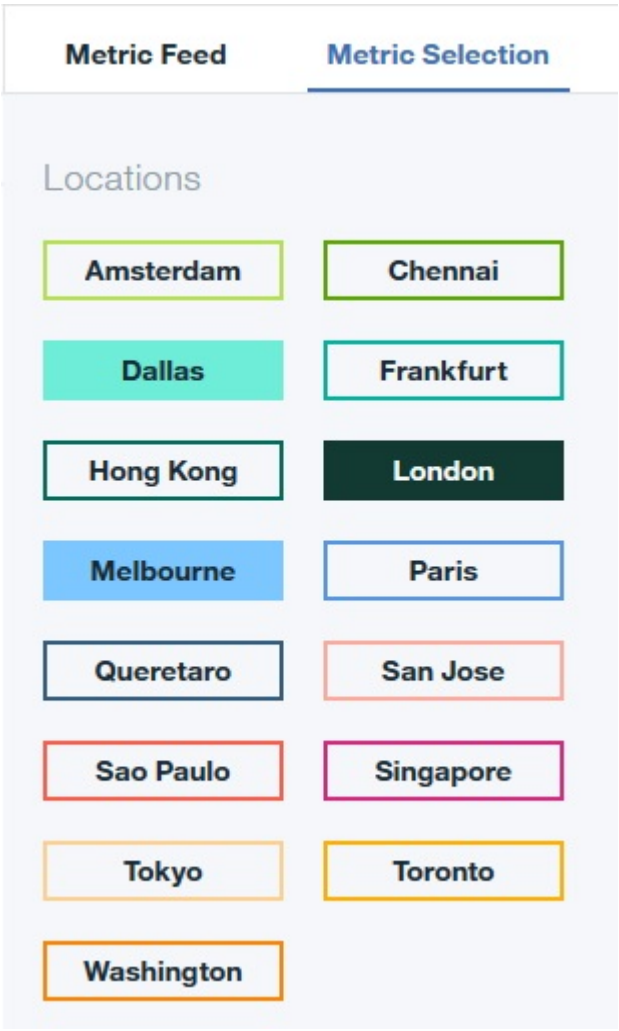
アラート

「アラート」行で警告アラートおよびクリティカル・アラートを容易に識別できます。**アラート・アイコン** ●▲ にマウスを重ねると、そのアラートの重大度とタイム・スタンプが表示されます。「**アラート」アイコン**をクリックすると、「**メトリック・フィード**」にそのアラートの詳細が表示されます。

「アラート」行に複数のアイコンが一緒に表示される場合は、「**数値**」アイコンにその時点のアラート数が表示されます。「**数値**」アイコンの 上にカーソルを合わせると、個々のアラートが表示されます。アラートをクリックすると、「**メトリック・フィード**」に情報が表示されます。

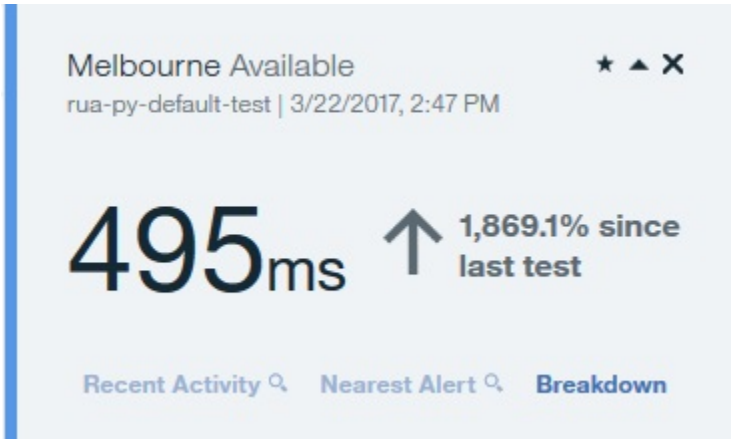
メトリック選択とメトリック・フィード

地理的領域でメトリックをフィルターに掛けるには、「**メトリック選択**」をクリックします。任意の場所をクリックし、その場所で測定されるメトリックをグラフに追加したり、グラフから削除したりします。「**別のロケーションを追加 (Add more locations)**」をクリックすると、「テスト編集モード (Test Edit Mode)」ページが開き、選択したテストに PoP ロケーションを追加できるようになります。

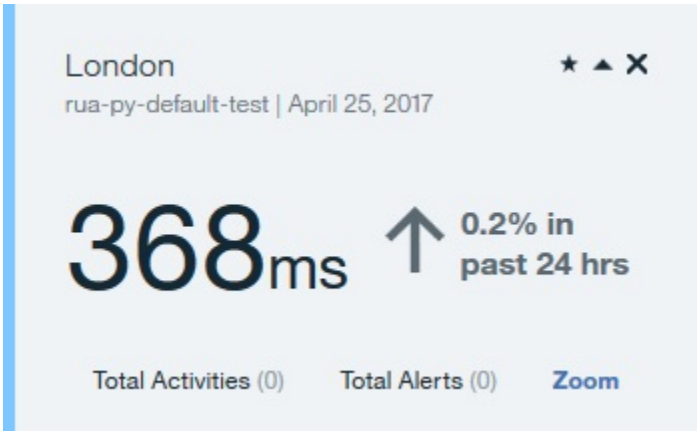


メトリック詳細のリストを表示するには、「**メトリック・フィード**」をクリックします。「**メトリック・フィード**」には、メトリックが満たされるインスタンスのリストが表示されます。

グラフの「**アラート**」アイコンまたは「**応答ポイント (Response point)**」 をクリックして、そのメトリックの詳細を「**メトリック・フィード**」に追加します。



24 時間を超える時刻範囲で「応答時間」グラフをフィルタリングする場合、「**応答ドット**」をクリックすると、「**メトリック・フィード**」にその日に関する集約された詳細情報を表示できます。

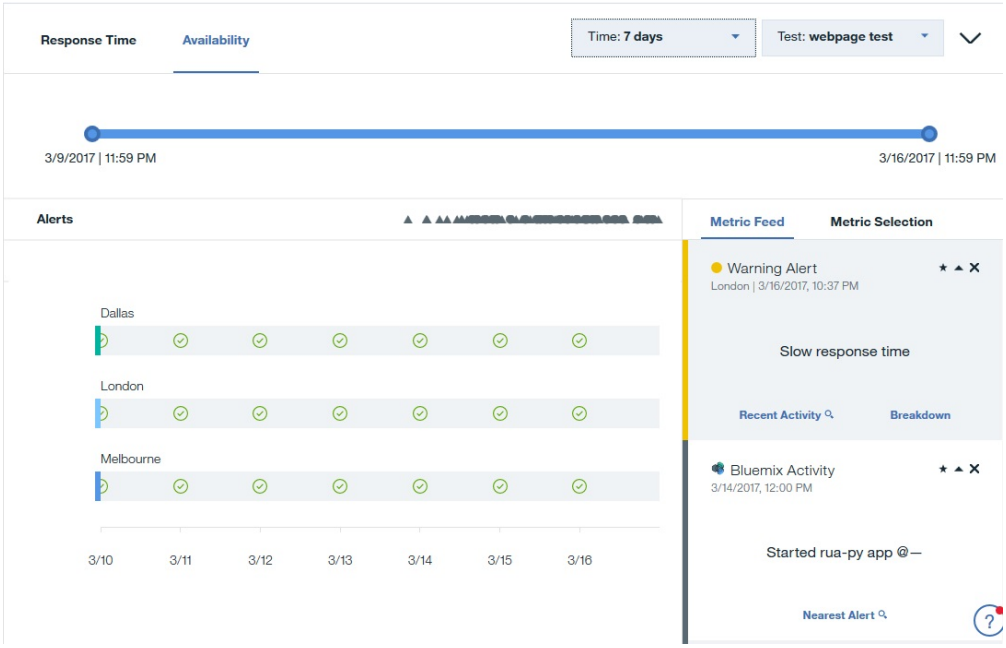


「**ズーム**」をクリックすると、「応答時間」グラフに、その日のテストによって生成されたすべての応答時間とアラートが表示されます。

アラートまたはテスト応答時間に関する詳細情報を表示するには、「**メトリック・フィード**」で「**明細**」をクリックします。「**最も近いアラート**」をクリックすると、アラートが発生していた場合、そのテスト・インスタンスに最も近いアラートが「**メトリック・フィード**」に表示されます。

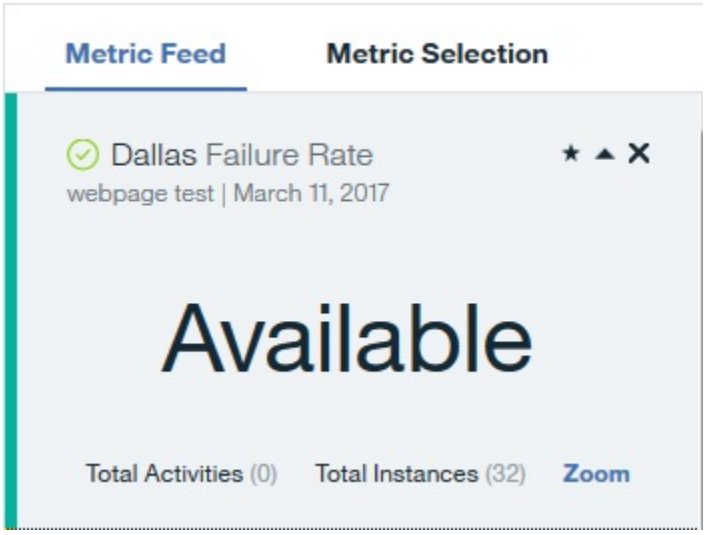
可用性

アプリケーションの可用性に関する情報を表示するには、「**可用性**」をクリックします。「可用性」グラフに、選択したテストの Point of Presence (PoP) ごとの日次可用性が表示されます。



「可用性」グラフでは、特定の PoP ロケーションからのデータを強調表示したり、非表示にしたりできます。特定のロケーションの可用性データを強調表示するには、PoP ロケーション名の上にポインターを移動して、「**ロケーションの強調表示 (Highlight location)**」アイコン をクリックします。特定のロケーションの可用性データを非表示にするには、PoP ロケーション名の上にポインターを移動して、「**ロケーションを非表示にする (Hide location)**」アイコン をクリックします。PoP ロケーションのデータをグラフに復元するには、「**メトリック選択**」タブをクリックして、以前に削除した PoP ロケーションをクリックします。

グラフの任意のポイントにカーソルを合わせると、その特定の日と場所に関するテスト・インスタンスの数と失敗率が表示されます。グラフの任意のポイントをクリックすると、「**メトリック・フィード**」にこの情報が表示されます。



「**ズーム**」をクリックして「**可用性**」グラフと「**応答時間**」グラフをフィルタリングし、選択した日付の情報を表示します。

Availability Monitoring の使用量

Availability Monitoring の使用量に関する詳細は、アプリケーション・ペインの「**モニター**」タブと、Availability Monitoring メイン・ダッシュボードで確認できます。

Availability Monitoring メイン・ダッシュボードから使用量の概要を表示するには、「**構成**」アイコン をクリックします。Availability Monitoring 試用版のユーザーの場合は、使用量は棒グラフにパーセンテージで表示され、使用中のテストの数がともに表示されます。有料の計画のユーザーの場合は、使用量がデータ・ポイントとして表示されます。「**シンセティック・テスト**」ペインで、各テストの使用量の詳細を確認できます

使用量がデータ・ポイントとして測定されます。データ・ポイントの推定数は以下の数式で計算されます。

1 カ月あたりのデータ・ポイントの推定数 = T * L * (60/M * 24 * 30)

T は実行したシンセティック・テストの数、L は場所の数、M はテストの間隔 (分単位) です。

単純なテスト (Web ページや REST API のテスト) では、テストごとに 1 個のデータ・ポイントを使用します。高度なテスト (Selenium スクリプトや REST API スクリプトのテスト) では、テストごとに 100 個のデータ・ポイントを使用します。

API の解説

Cloud APM 環境の準備を自動化するためのスクリプトを作成するには、IBM Cloud Application Performance Management API を使用します。IBM developerWorks の API Explorer で提供されている Cloud APM の API 管理サービス・オファリングから、使用可能なリソース・グループ管理サービス API、しきい値管理サービス API、および役割ベースのアクセス制御サービス API にアクセスし、これらの API を探索できます。

始める前に

クライアント ID を取得して API 操作を実行するには、Cloud APM のアクティブなサブスクリプションが必要です。

手順

1. ブラウザーで API Explorer (https://developer.ibm.com/api) を開きます。
2. IBM ID でサインインします。
3. 「**すべての API の検索 (Search All APIs)**」フィールドに、`performance management` と入力し、 をクリックします。
4. 「IBM Cloud Application Performance Management API」タイルを選択します。
5. API Explorer ウィンドウの左側にある「**資料 (Documentation)**」をクリックします。
6. 特定の API を選択します。
7. サブセクションを選択して API 操作のリストを展開します。
8. 次の手順に進むには、Cloud APM のアクティブなサブスクリプションが必要です。以下のいずれかのステップを実行します。
   a) Cloud APM のサブスクリプションをまだ取得していない場合は、30 日間の無料評価版サブスクリプションに登録します。
   b) サブスクリプションを既に取得している場合は、「**マイ API (My APIs)**」をクリックしてサインインし、API Explorer 内でいくつかの API 操作をテストします。

   サブスクリプションがアクティブな 状態で API Explorer ページにサインインすると、API サブスクリプションのリストが表示されます。
9. 詳細を確認するには、API 操作を選択します。
10. 要求の例を表示するには、API Explorer ページの上部でいずれかの言語 (shell、curl など) を選択します。
11. クライアント ID キーとクライアント秘密鍵を入手して、外部で使用できるように安全な場所に保管します。

クライアント ID キーとクライアント秘密鍵は、API 要求を実行するたびに送信します。このアクションを実行するには、Cloud APM のサブスクリプションが必要です。

次のタスク

API 操作を実行する方法について詳しくは、以下のトピックを参照してください。

1012 ページの『役割ベースのアクセス制御サービス API へのアクセスとその使用』
999 ページの『リソース・グループ管理サービス API の使用』
1001 ページの『しきい値管理サービス API の使用』

拡張構成

通信設定および拡張機能 (イベントの転送など) を制御するには、「**拡張構成**」ページを使用します。

「**システム構成**」 > 「**拡張構成**」をクリックすると、「拡張構成」ページに以下の構成カテゴリーが表示されます。

UI 統合

Cloud APM コンソール と統合される製品の場合は、統合されたアプリケーションを起動する URL を追加または編集できます。 フィールドには、統合構成手順で設定された URL が入力されます。

- **ログ分析 URL** は、Application Performance ダッシュボード からアプリケーション・ログをくまなく検索するための IBM Operations Analytics - Log Analysis を起動するために使用します。詳しくは、1081 ページの『ログ・ファイルの検索』を参照してください。
- 「**サブノード・イベントの有効化**」は、サブノードを持つエージェントについて、サブノードを「イベント」タブに表示するかどうかを制御します。サブノード・イベントが有効になっている場合、イベントのオープン対象であるノードおよびサブノードが表示されます。特に、「イベント」タブにログ・ファイルのモニター・シチュエーションを表示する場合は、サブノード・イベントが有効になっていることを確認する必要があります。 デフォルト: `False`。
- 「**ダッシュボードの最新表示率**」は、Application Performance ダッシュボードの自動最新表示の頻度を制御します。設定は、1 から 60 分の間の任意の値に調整できます。この設定は、ナビゲーターおよび「状況の概要」タブに表示されるリソース状況に影響します。「イベント」タブの項目には影響しません。 デフォルト: 1 分。

イベント・マネージャー

イベント・マネージャーは、Simple Mail Transfer Protocol を使用したイベントの転送と、E メール通知を制御します。「ターゲット E メール・アドレス」に値を入力すると、オープン、クローズ、および停止の各イベントが発生するたびに E メールが送信されます。「**イベント・マネージャー**」フィールドを使用して、Cloud APM からイベントが、IBM Control Desk で自動的にチケットをオープンするように構成することができます。追加の構成タスクについては、Control Desk との統合 973 ページの『Control Desk との統合』を参照してください。

SSL を使用するように SMTP 転送機能を構成する場合、SMTP サーバーが署名する CA 証明書を Cloud APM サーバーの鍵ストアに追加する必要があります。CA 証明書は、次のように JVM keytool コマンドを使用して、デフォルトの鍵ストアに追加します。

```
install_dir/java/jre/bin/keytool -importcert¥
-noprompt ¥
-alias your_CA cert_alias ¥
-file path_to_your_CA_cert_file (*.cer)
 -keystore /install_dir/wlp/usr/servers/min/resources/security/key.jks ¥
-storepass ccmR0cKs! \
-storetype jks ¥
-trustcacerts
```

E メールの例は 1076 ページの『イベント E メール』で参照できます。

- 「**ターゲット E メール・アドレス**」は、イベントの転送先の E メール・アドレスを指定します。各アドレスはコンマ (,) で区切ります。例えば、`annette@ibm.com,jim@ibm.com,owen@ibm.com`。

- 「**ITMv6 のアラート通知**」は、値 True を設定することにより、ITMv6 イベントのアラート通知を有効にするオプションです。デフォルト: False。
- 「**Cloud Event Management Webhook**」は、IBM Cloud Application Performance Management と Cloud Event Management の間での統合を構成するときに Cloud Event Management で生成される Webhook URL です。生成された Webhook URL をここに貼り付けて、イベントが Cloud APM から転送されるようにする必要があります。

「イベント・マネージャー」E メール機能の代わりに、アラート通知を使用できます。これにより、通知を受けるユーザーをイベントごとに制御したり、通知の方法を制御したりできます。詳しくは、972 ページの『Alert Notification との統合』を参照してください。

イベントを EIF (Event Integration Facility) 受信側に転送する場合、EIF イベントに属性を追加するなど、EIF スロットをカスタマイズできます。詳しくは、992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』を参照してください。IBM Netcool/OMNIbus イベント・マネージャーへのイベントの転送については、966 ページの『Netcool/OMNIbus との統合』を参照してください。

トラッキング分析サービス

トラッキング分析サービスに使用する設定。この設定は、ご使用の環境でトランザクション・トラッキングを構成する場合に、 Cloud APM, Advanced オファリングのみに適用されます。

- 「**接続プール・サイズ**」は、トラッキング分析サービスが「上位 *N* 件の照会」のために接続プール内に保持する同時 Db2 接続の数です。Cloud APM コンソールの同時ユーザーが多いことが原因で照会時間が遅くなっている場合は、この値を大きくしてください。デフォルト: 10。
- 「**疑似ノード**」は、インスツルメントされていないサービスの可視化を有効にします。デフォルト: True。
- 「**照会タイムアウト (秒)**」は、各「上位 *N* 件」(*N* は、「上位 5 件」、「上位 10 件」のような数値を表す) の照会がタイムアウトするまでの時間 (秒数) です。タイムアウト値は、ワークロードが高くて照会に時間がかかると予想されるアプリケーションでは、必要に応じて大きくできます。デフォルト: 120 秒。
- 「**DB2 照会の再最適化が有効**」は、通常は変更する必要はありません。 このパラメーターは、Db2 照会オプティマイザーに影響を与えます。環境によっては、オプティマイザーをオフにすると、一部のトランザクションではパフォーマンスが向上する場合があります。デフォルト: False。

エージェント・サブスクリプション・ファシリティー

エージェント・サブスクリプション・ファシリティーには、エージェントの REST (Representative State Transfer) インターフェースおよび Central Configuration Service の HTTP サーバーが含まれます。REST インターフェースは、Db2 サーバー およびしきい値イベントで保持されるモニター・データを送信するためにエージェントおよびデータ・コレクターが使用します。Central Configuration Services の HTTP サーバーは、構成ファイルに対するエージェントからの要求 (しきい値の定義など) を処理します。これらのパラメーターは、 エージェント・サブスクリプション・ファシリティーと Cloud APM サーバー の間の通信を構成するために使用します。

- 「**ポーリング無応答の制限 (高速ハートビート)**」は、ハートビート間隔が 60 秒以下のモニター・エージェントが接続に失敗するときに、オフラインのマークを付けるまでの無応答の最大数です。デフォルト: 30 間隔。
- 「**ポーリング無応答の制限 (低速ハートビート)**」は、ハートビート間隔が 60 秒より長いモニター・エージェントが接続に失敗するときに、オフラインのマークを付けるまでの無応答の最大数です。デフォルト: 3 間隔。
- 「**トランザクション・タイムアウト**」は、サーバーが要求に対する応答を待機する時間 (秒) です。デフォルト: 120 秒。
- 「**オフライン・システム除去の遅延**」は、オフラインの管理対象システムを表示しないようにするまでの待機時間 (分単位) を決定します。Application Performance ダッシュボード では、オフラインの管理対象システムは、 不明の状況標識によって示されます。管理対象システムは、エージェントをアンインストールしても 、遅延時間が経過するまでは引き続き表示されます。 詳しくは、1104 ページの『オフライン・エージェントの表示および削除』を参照してください。 デフォルト: 5760 分 (4 日間)。

しきい値の使用可能化

モニター・エージェントにはそれぞれ、エージェントとともに使用可能になり開始される事前定義しきい値のセットが付属しています。これらの事前定義しきい値は、エージェントのデフォルトのシステム・リソース・グループに割り当てられています。

- 「**事前定義ベスト・プラクティスしきい値のポリシーを定義するアクションの選択**」は、管理対象リソースの事前定義しきい値をデフォルトで有効にするか無効にするかを制御します。事前定義しきい値を実行したくない場合は、フィールドを「**すべて使用不可にする**」に設定します。「**すべて使用不可にする**」設定により、すべての事前定義しきい値からシステム・グループの割り当てが削除されます。グループが割り当てられていないしきい値はどのモニター対象システムにも配布されず、リソース・グループに配布されるまで停止したままになります。後で事前定義しきい値をオンにする場合は、当該フィールドを「**すべて使用可能にする**」に設定します。

事前定義しきい値、およびカスタムしきい値について詳しくは、978 ページの『背景情報』および 980 ページの『使用不可のしきい値の例』を参照してください。

イベント E メール

「拡張構成」ページのイベント・マネージャーのフィールドを 使用して、受信人のリストへの E メールによるイベント通知をセットアップします。

オープン・イベントの E メール

しきい値条件が true になると、イベントが開かれます。Cloud APM サーバーが送信する E メール・メッセージには、すべてのエージェント・イベントに適用される基本属性と、しきい値条件を満たしたデータ・セットの最初の行に記載された属性が含まれます。オープン・イベントの situation_status 属性の値は Y です。

クローズ・イベントの E メール

しきい値条件が true でなくなると、クローズ・イベントが生成されます。クローズ・イベントの E メール・メッセージには、すべてのエージェント・イベントに適用される基本属性だけが含まれ、situation_status の値は N です。しきい値条件が満たされていないため、これらの E メール・メッセージにはエージェントの属性が含まれません。

停止イベントの E メール

しきい値が停止されると、停止イベントが生成されます。停止イベントの E メール・メッセージには、すべてのエージェント・イベントに適用される基本属性だけが含まれ、situation_status の値は P です。しきい値条件が満たされていないため、これらの E メール・メッセージにはエージェントの属性が含まれません。

エージェントのしきい値が停止されるのは、しきい値定義を削除した場合や、エージェントに配布されたいずれかのしきい値定義に変更を加えた場合です。

以下のサンプルはオープン ・イベントの E メールを示しています。

```
From:    noreply@apm.ibmserviceengage.com
To:         tester@us.ibm.com
Date:    10/25/2017 01:56 PM
Subject:    Linux_Disk_Space_Low on nc049048:LZ (Notification)
ここに示すテキストは、このイベントをトリガーしたエージェントから受信した情報をリストしたものです。
IP の値と「エージェント」の値はこのイベントを検出したエージェントを識別します。
「説明」と「重大度」の値は、しきい値の定義の名前と、その重大度を示します。
「説明」の下には、イベントに存在するすべての属性/値ペアが、未加工形式で示されています。
    サーバー IP  : 10.107.76.230 (SIDR26APAP1BLUE.test.ibm.com)
    エージェント IP   : 9.42.49.48
    エージェント      : nc049048:LZ
    重大度   : warning
    説明: Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10
                AND FS_Type != nfs  AND FS_Type != iso9660 ]
      ITM_KLZ_Disk
      ManagedSystemGroups='*LINUX_SYSTEM'
      TenantID=F43E-D704-DADC-6270-1ED8-543E-A388-6513
      adapter_host=nc049048.tivlab.raleigh.ibm.com
      apm_hostname=SIDR26APAP1BLUE.test.ibm.com
      appl_label=A:P:S
      date=01/25/2017
      disk_free=5843
      disk_free_percent=20
      disk_name=/dev/sda2
      disk_used=22676
```

```
      disk_used_percent=80
      file_system_status=2
      file_system_status_enum=Up
      fqhostname=nc049048.test.ibm.com
      fs_type=ext4
      hostname=nc049048.test.ibm.com
      identifier=Linux_Disk_Space_Lownc049048:LZ/ITM_KLZ_Disk
      inodes_free=1721587
      inodes_free_percent=88
      inodes_used=232477
      inodes_used_percent=12
      integration_type=U
      mount_options=rw
      mount_point=/
      msg='Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10  AND FS_Type !
= nfs
     AND FS_Type != iso9660 ]'
      origin=9.42.49.48
      severity=WARNING
      situation_displayitem=/
      situation_eventdata='disk_name=/dev/
sda2;inodes_used_percent=12;mount_options=rw;fs_type=ext4;

system_name=nc049048:LZ;mount_point=/;disk_used_percent=80;disk_free=5843;file_system_status_enum=
Up;
     size=30040;disk_used=22676;inodes_used=232477;disk_free_percent=20;file_system_status=2;
     total_inodes=1954064;inodes_free=1721587;timestamp=1170125135553000;inodes_free_percent=88;~'
      situation_name=Linux_Disk_Space_Low
      situation_origin=nc049048:LZ
      situation_origin_uuid=09fb36afd6b3.22.02.09.2a.31.30.56.9d
      situation_status=Y
      situation_thrunode=nc049048:LZ
      situation_time='01/25/2017 13:55:55.000'
      situation_type=S
      size=30040
      source='ITM Agent: Private Situation'
      sub_origin=/
      sub_source=nc049048:LZ
      system_name=nc049048:LZ
      timestamp=1170125135553000
      tmz_diff=18000
      total_inodes=1954064
```

これらの E メールの購読を中止するには、Cloud APM コンソールにログインし、
「拡張構成」ページの「イベント・マネージャー」カテゴリーにある
「ターゲット E メール・アドレス」のリストから自分の E メール・アドレスを
削除します。

以下のサンプルはクローズ・イベントの E メールを示しています。

```
From:    noreply@apm.ibmserviceengage.com
To:      tester@us.ibm.com
Date:    01/25/2017 02:01 PM
Subject:    Linux_Disk_Space_Low on nc049048:LZ (Closed)
```

ここに示すテキストは、このイベントをトリガーしたエージェントから受信した情報をリストしたものです。
IP の値と「エージェント」の値はこのイベントを検出したエージェントを識別します。
「説明」と「重大度」の値は、しきい値の定義の名前と、その重大度を示します。
「説明」の下には、イベントに存在するすべての属性/値ペアが、未加工形式で示されています。

```
    サーバー IP  : 10.107.76.230 (SIDR26APAP1BLUE-12f.test.ibm.com)
    エージェント IP   : 9.42.49.48
    エージェント      : nc049048:LZ
    重大度   : warning
    説明: Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10
                AND FS_Type != nfs  AND FS_Type != iso9660 ]
      ITM_KLZ_Disk
      ManagedSystemGroups='*LINUX_SYSTEM'
      TenantID=F43E-D704-DADC-6270-1ED8-543E-A388-6513
      adapter_host=nc049048.tivlab.raleigh.ibm.com
      apm_hostname=SIDR26APAP1BLUE-12f.test.ibm.com
      appl_label=A:P:S
      date=01/25/2017
      fqhostname=nc049048.test.ibm.com
      hostname=nc049048.test.ibm.com
      identifier=Linux_Disk_Space_Lownc049048:LZ/ITM_KLZ_Disk
      integration_type=U
      msg='Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10  AND FS_Type !
= nfs
     AND FS_Type != iso9660 ]'
      origin=9.42.49.48
      severity=WARNING
      situation_displayitem=/
      situation_eventdata=~
```

```
situation_name=Linux_Disk_Space_Low
situation_origin=nc049048:LZ
situation_origin_uuid=09fb36afd6b3.22.02.09.2a.31.30.56.9d
situation_status=N
situation_thrunode=nc049048:LZ
situation_time='01/25/2017 14:00:55.000'
situation_type=S
source='ITM Agent: Private Situation'
sub_origin=/
sub_source=nc049048:LZ
tmz_diff=18000
```

これらの E メールの購読を中止するには、Cloud APM コンソールにログインし、「拡張構成」ページの「イベント・マネージャー」カテゴリーにある「ターゲット E メール・アドレス」のリストから自分の E メール・アドレスを削除します。

第 10 章 ダッシュボードの使用

アプリケーションの総合的な状況の概要を知るには、「**パフォーマンス**」>「**Application Performance ダッシュボード**」を選択します。同じ画面で、大まかな概要から詳細なメトリックにドリルダウンすることができます。

ご使用の環境のクリティカル状態および警告状態の調査、追加のメトリック・ビューの作成、およびトレース・ログの検索やメトリックの時間変化の比較などのアクションの実行を行うには、ダッシュボードで使用可能なツールを使用します。

すべてのマイ・アプリケーション - Application Performance ダッシュボード

Application Performance ダッシュボードは、「**すべてのマイ・アプリケーション**」にモニター対象ドメインの要約状況を示します。ユーザー定義アプリケーション (「インベントリー管理」など) と、事前定義アプリケーション (「マイ・コンポーネント」または「マイ・トランザクション」がご使用の環境に存在する場合はそれらのアプリケーション) ごとに要約ボックス が表示されます。要約ボックスまたはナビゲーターから、各アプリケーションおよびその構成要素にドリルダウンして、詳細メトリックを表示します。

項目を選択するとパスが表示されます。いずれかのパス・リンクをクリックすると、そのビューに戻ります。任意の Cloud APM コンソール・ページから、「**パフォーマンス**」>「**Application Performance ダッシュボード**」をクリックして「**すべてのマイ・アプリケーション**」ダッシュボードを開くことができます。興味のある分野を表示するには、ナビゲーターからそれらを選択するか、要約ボックスをクリックして次のレベルにドリルダウンします。

要約ボックス

「**すべてのマイ・アプリケーション**」には、定義済みアプリケーションごとの要約ボックスがあります。インディケーターにより、アプリケーションの最高の状況重大度がタイトル・バーに表示され、各グループの最高の状況重大度が要約ボックスに表示されます。定義されたアプリケーションに含まれているモニター製品に応じて、以下の事前定義グループが使用可能です。

「**Availability Monitoring**」にサブグループがありません。

「**コンポーネント**」には、ご使用のアプリケーションをサポートする 各モニター・エージェント・タイプごとのサブグループがあります

「**トランザクション**」には、「**エンド・ユーザー・トランザクション**」サブグループおよび「**シンセティック・トランザクション**」(2017 年 8 月リリースよりも前の IBM Website Monitoring on Cloud) サブグループが含まれる場合があります。

さらに、各要約ボックスには「**イベント**」が含まれています。ここには、そのアプリケーションに対してオープンされている最高の重大度のイベントの重大度が表示されます。「イベント」リンクをクリックして、オープン・イベントを調べることができます (1109 ページの『イベント状況』を参照)。

要約ボックスのタイトル・バーをクリックして、アプリケーションの「状況の概要」タブを開きます。 あるいは、要約ボックスのアイコンのいずれかをクリックして、「コンポーネント」グループまたは「ユーザー」または「トランザクション」のサブグループの「状況の概要」タブを開くか、アプリケーションのグループまたはサブグループの 「イベント」タブを開きます。

次のようにチェック・ボックスを選択またはクリアすることで、要約ボックスの縮小表示やフィルタリングを実行できます。

- 定義したアプリケーション内をスクロールしやすくするために要約ボックスのタイトル・バーのみを表示するには、「**詳細の表示**」チェック・ボックスをクリアします。
- 非表示にする重大度状況で要約ボックスをフィルタリングするには、カウンター (

など) のチェック・ボックスをクリアします。イベントがない重大度のチェック・ボックスは使用不可になります。例えば、次のグラフィックで、「クリティカル」と「正常」のフィルターは使用可能であり、「警告」と「不明」のフィルターはカウントが 0 のため、使用不可です。

.

© Copyright IBM Corp. 2014, 2019

検索

「検索」フィールドにテキストを入力すると、そのテキストが含まれているログ・エントリーを、過去 1 時間の範囲で検索することができます。 をクリックすると、過去 2 時間以外の範囲で結果を表示することができます。「検索」フィールドに入力されたテキストは、ナビゲーターでの選択内容に関連するログ・エントリーと比較され、一致するすべてのエントリーが、新しいブラウザー・タブまたはウィンドウに表示されます。 説明については、1081 ページの『ログ・ファイルの検索』を参照してください。検索機能は、IBM Operations Analytics - Log Analysis により実現しています。

アクション

「**アクション**」メニューには、URL のコピー、ダッシュボード・ログのオープン、およびトラブルシューティングのためにトレースの 設定を行うためのオプションがあります。詳しくは、1122 ページの『ダッシュボード URL のコピー』および 1123 ページの『トレースの設定』を参照してください。

直近のサーバー再始動以降に更新されたエージェント・ダッシュボードのリストを確認するには、「**ダッシュボードのログ**」オプションを使用します。

Cognos® ベースのレポートが使用可能であり、かつご使用の環境に Tivoli Common Reporting が含まれている場合、「**すべてのマイ・アプリケーション**」ホーム・ダッシュボードまたはアプリケーションのいずれかを選択すると、「アクション」メニューに「**レポートの起動**」が含まれます。これを使用して、使用率やパフォーマンスのトレンドの分析に役立てることができます。詳しくは、1124 ページの『レポート』を参照してください。

ナビゲーターの「**グループ**」セクションまたは要約ボックスから「**コンポーネント**」グループを選択すると、コンポーネント状況の概要ダッシュボードを編集するための「**編集**」オプションが「**アクション**」メニューに表示されます。「**編集**」オプションを使用できるのは、ログインしたユーザーが Performance Management ダッシュボードに対する変更権限とアプリケーションに対する作成権限を持っている場合に限られます。詳しくは、1090 ページの『コンポーネント・ダッシュボードのグループ・ウィジェットの編集』を参照してください。

ヘルプ

現在のダッシュボードの簡単な説明を見るには、ポップアップ・ヘルプを開きます。以下のリンクがあります。「**説明**」をクリックすると、Cloud APM のローカル・ヘルプ・システム内の、ダッシュボードに関する全トピックが開きます。「**ダッシュボード・ツアーの実行**」をクリックすると、IBM Cloud APM ダッシュボード・ツアーが始まります。このツアーでは、各機能をガイドしながら、ダッシュボード・エレメントを簡単に説明します。

ナビゲーター

ナビゲーターには、定義済みアプリケーションとそのユーザー、トランザクション、およびコンポーネントがどのように編成されているかを反映した階層が表示されます。ナビゲーターにはアプリケーション階層のレベルごとにセクションが設けられています。ナビゲーターのすべてのレベルで、ダッシュボードのメトリックが変化し、 コンポーネントからのデータが表示されます。項目を選択すると、 ダッシュボードのコンテキストがその選択内容に切り替わります。表示される範囲は ユーザーの権限によって決定されます。

各ナビゲーター項目には、 クリティカル、 警告、 正常、または 不明の状況標識 (これはエージェントが使用不可であることを示します) があります。各ナビゲーター・セクションには、選択されたナビゲーター項目に関連した重大度ごとにイベント数が表示されます。しきい値イベントへの状況のマッピングについては、1109 ページの『イベント状況』を参照してください。

他のセクションの表示領域を広げるには、タイトル・バーをクリックしてセクションを省略した後、再度クリックしてセクションを元に戻します。ナビゲーターの枠にある をクリックすることで、ナビゲーターを完全に非表示にすることもできます。元に戻すには をクリックします。幅を調整するには 枠をドラッグします。

ナビゲーターには以下の 3 つのセクションがあります。

- 「**アプリケーション**」セクションには、ドメイン内にあるすべての定義済みアプリケーションまたはユーザー役割で許可されているすべての定義済みアプリケーションがリストされます。
  - アプリケーションを選択すると、ダッシュボードはハイレベルの状況要約が示された「**状況の概要**」タブに変わり、最高の重大度状況の標識が「**イベント**」タブに表示されます。詳しくは、1082 ページの『アプリケーション - Application Performance ダッシュボード』を参照してください。

  - モニター・インフラストラクチャーによって検出されたドメイン内のコンポーネントはすべて、「マイ・コンポーネント」という名前の事前定義のアプリケーションで表示されます。これは、編集することも削除することもできません。
  - 管理者役割の特権を持っている場合は、「**アプリケーション**」ツールバーに ツールが表示され、アプリケーションを管理することができます。説明については、1098 ページの『アプリケーションの管理』を参照してください。
- アプリケーションを選択すると、そのアプリケーションをサポートするグループが「**グループ**」セクションに表示されます。詳しくは、1087 ページの『グループおよびインスタンス - Application Performance ダッシュボード』を参照してください。
- サブグループを選択すると、「**インスタンス**」セクションの名前がサブグループのタイトルに変更され、個々の管理対象システムの名前が取り込まれます。「**状況の概要**」が、選択したサブグループの KPI を表示するように変わります。管理対象システムを選択すると、管理対象システムからの KPI を示す詳細なグループ・ウィジェットが表示されます。コンポーネント・インスタンスの場合は、「**属性の詳細**」 タブでも、選択したデータ・セット属性からの KPI を示す表を表示できます。詳しくは、1092 ページの『カスタムのグラフおよび表の表示と管理』を参照してください。

ナビゲーターに「**ネットワーク・エラー**」のポップアップ・メッセージが出た場合に、接続が復旧されるまで状況標識が「正常」に変化する場合があります。そのときは、オープン・イベントがすべて再処理され、処理が完了するまで状況が「正常」と表示されたままになる可能性があります。

ログ・ファイルの検索

ユーザーが経験している問題 (処理速度が遅い、障害が発生するなど) の根本原因を調べるために、使用しているアプリケーションに関連するログ・データを検索できます。IBM Operations Analytics - Log Analysis は検索機能を提供します。ご使用のアプリケーションで経験する問題の根本原因を見つけ、迅速に問題を解決するのに役立つように、アプリケーションのログ・データとパフォーマンス・データがまとめられます。

手順

CPU の使用率が高いなど、調査中の問題に関係している可能性があるログ・エントリーを探すには、以下の手順を実行します。

1. Application Performance ダッシュボードが表示されない場合は、「**パフォーマンス**」メニューから選択します。
2. アプリケーション内を検索する場合は、「すべてのマイ・アプリケーション」ダッシュボードでいずれかのアプリケーションを選択します。
   例えば、すべてのコンポーネント・リソースのログを検索する場合は、「マイ・コンポーネント」をクリックします。
3. 検索するログ・ファイル・テキストを 検索ボックスに入力します。
   例えば、以前のレベルにロールバックされた WebSphere Applications エージェントのリソースを検索するには、「ロールバック」と入力します。
4. 「**過去 1 時間**」以外の時刻範囲のデータを検索する場合は、をクリックし、異なる期間を選択します。
5. をクリックします。

タスクの結果

現在のナビゲーター・レベルのコンテキストにおける検索テキストを持つすべてのログ・エントリーが、新しいブラウザー・タブまたはウィンドウに表示されます。ブラウザー・ウィンドウには、コンテキストに由来する名前が付けられます (「クレジット・カード処理」アプリケーションなど)。

次のタスク

検索結果を確認します。別のアプリケーションを選択すると、コンテキストに合わせて検索結果が変更されます。検索結果をさらに絞り込むには、検索フィールドを使用します。例えば、検索フィールドに db2 AND (datasourceHostName:Pear* OR datasourceHostname:Persimmon* OR

datasourceHostname:Pomegranate*) と表示されている場合は、次のようにデータ・ソースを削除して結果を絞り込むことができます: db2 AND (datasourceHostname:Persimmon*)。

詳しくは、IBM Knowledge Center の IBMOperations Analytics Log Analysis トピック・コレクションを参照するか、IBM Operations Analytics - Developers Community にアクセスしてください。

アプリケーション - Application Performance ダッシュボード

ナビゲーターまたは「**すべてのマイ・アプリケーション**」ダッシュボードの要約ボックスからアプリケーションを選択すると、タブ付きのダッシュボードに、アプリケーションのさまざまな側面が表示されます。「**状況の概要**」タブは、アプリケーションのハイレベルな状況要約を示します。グラフのしきい値と状況標識により、全体的な正常性とパフォーマンスのフィードバックが得られます。アプリケーションの正常性に寄与しているイベントしきい値を確認するには「**イベント**」タブを選択します。

ナビゲーターおよびバナーのエレメントの説明については、1080 ページの『ナビゲーター』、1080 ページの『検索』、1080 ページの『アクション』、および 1080 ページの『 ヘルプ』 を参照してください。

状況の概要

- 選択したアプリケーションの構成に応じて、「**状況の概要**」タブにアプリケーションの状況を全体的に評価するためのパースペクティブが 1 つ以上表示されます。

  時間の経過にともなう可用性

  IBM Website Monitoring on Cloud (2017 年 8 月リリースよりも前): 「**時間の経過にともなう可用性**」棒グラフは、アプリケーションに Synthetic Playback エージェント (「**トランザクション**」ナビゲーター・グループ内、および事前定義アプリケーションで「**マイ・トランザクション**」) が含まれる場合に表示されます。

  各プロット点はトランザクションのサンプルであり、「**正常**」、「**低速**」または「**使用不可**」の状況を表すカラー・インディケーターが示されます。

  棒のタイムラインで任意の場所をクリックして、「**トランザクション・リスト**」表および「**ロケーション・リスト**」表を示すポップアップ・ウィンドウを開きます。

  要求数および応答時間

  アプリケーションに Response Time Monitoring エージェント (「**トランザクション**」ナビゲーター・グループ内の「**エンド・ユーザー・トランザクション**」) が含まれている場合、「**要求数および応答時間**」積み重ね棒グラフが表示されます。

  このグラフを使用してパフォーマンスのトレンド・パターンを見つけます。それぞれの積み重ね棒は、良好な応答時間で完了した要求、低速な応答時間で完了した要求、および完了に失敗した要求のパーセンテージを示しています。線グラフのオーバーレイにより、5 分間の平均応答時間が示されます。1091 ページの『メトリックの調整および時間変化の比較』の説明に従って、時間セレクターを使用して、表示される時刻範囲を変更します。

  集約トランザクション・トポロジー

  「**集約トランザクション・トポロジー**」は、トランザクション・トラッキングが有効に設定されていて、アプリケーションに以下のいずれかのエージェントまたはデータ・コレクターが含まれている場合に表示されます。

  - DataPower エージェント
  - HTTP Server エージェント
  - IBM Integration Bus エージェント
  - J2SE データ・コレクター
  - JBoss エージェント
  - Liberty データ・コレクター
  - Microsoft .NET エージェント
  - Microsoft SQL Server エージェント
  - Node.js データ・コレクター

- Response Time Monitoring エージェント
- SAP NetWeaver Java Stack エージェント
- Tomcat エージェント
- WebLogic エージェント (Linux と Windows のみ)
- WebSphere Applications エージェント
- WebSphere MQ エージェント

トランザクション・トラッキングは、Response Time Monitoring エージェントを除くすべてのエージェントに対して手動で有効にする必要があります。 トランザクション・トラッキングは、データ・コレクターに対して自動的に有効になります。詳しくは、186 ページの『「エージェント構成」ページ』を参照してください。

「**集約トランザクション・トポロジー**」には、アプリケーションに関連付けられたリソースとその関係が表示されます。フッターには、選択したノードの数、リソース、関係、トポロジー内のフィルター (ある場合) に加えて、データが最後に更新された時刻が表示されます。

アプリケーション・コンポーネントをビジネス・アプリケーションに追加し、そのコンポーネントが複数のアプリケーションのトラフィックを 伝送する場合、それらのビジネス・アプリケーションについて表示されるアプリケーション・トポロジーには、すべてのアプリケーションのノードへのパスが含まれます。

JavaScript が自動的に挿入される IBM Java アプリケーション・スタックの場合、最上位ノードはブラウザーを表し、最も細分化されたノードはデータベースを表します。その他のアプリケーションの場合、最上位ノードはアプリケーションを 表し、最も細分化されたノードは管理対象のシステム・インスタンスを表します。

各ノードには状況標識と背景色があり、その集約レベルにおいて重大度が最も高い状況が示されます。スペースを広げるためにナビゲーターの表示を省略しても、「**集約トランザクション・トポロジー**」で引き続き同じ状況を確認できます。ノードのソース環境は、以下のように表示されます。「**クラウド (Cloud)**」(IBM Cloud Application Performance Management)、「**ITM**」(IBM Tivoli Monitoring)、「**オンプレミス**」(IBM Cloud Application Performance Management, Private) 、「**専用クラウド (Private Cloud)**」 (IBM Cloud Private)、または 「**公開クラウド (Public Cloud)**」(IBM Cloud)。「**その他**」(別の環境の管理対象リソース) についてはアイコンは 表示されません。

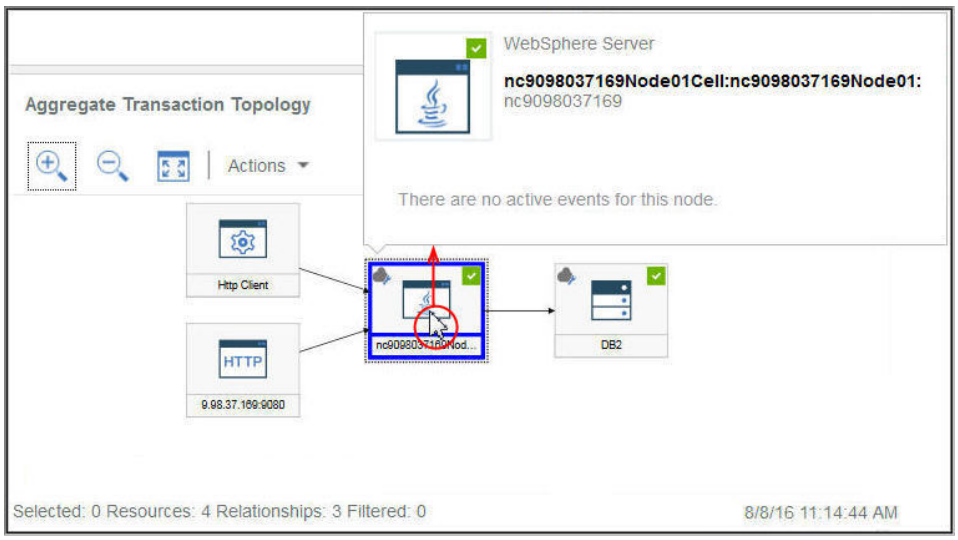
マウスをノードの上に移動して、ショートカット・メニューを開き、ノードを選択して、問題の根本原因の特定に役立つ状況の詳細情報を表示します。

- マウスをノードの上に移動すると、クリティカルおよび 警告のイベントのリストがポップアップ・メッセージに示されます。
- ノード内のリンク URL をダブルクリックして、コンポーネントまたはトランザクションの詳細が表示された、対応するダッシュボードを開きます。
- ノードを右クリックして、ダッシュボードのドリルダウン・オプションである、選択したサブグループ・ノードの「「**トランザクション要約**」**ページに移動**」、インスタンス・ノードの「「**コンポーネント・インスタンス**」**ページに移動**」、またはリソース名、状況、管理対象システム名、プロバイダー・ドメイン (「クラウド」など) を表示する「プロパティー」のいずれかを選択します。

表示を調整したり、アクションを実行したりするには、1086 ページの『集約トランザクション・トポロジー・ウィジェットの操作』の説明に従ってツールバー・アイコンを 使用します。

このビューと次に説明する「**現在のコンポーネント状況**」を切り替えるには、ツールをクリックします。

トポロジー情報のないモニター・エージェントは、「集約トランザクション・トポロジー」ウィジェットに表示されません。



現在のコンポーネント状況

「**現在のコンポーネント状況**」積み重ね棒グラフには、アプリケーション内の各コンポーネント・タイプについて、クリティカル、警告、正常、および不明の各状況のパーセンテージおよびカウントが表示されます。例えば、選択したアプリケーションを 5 つの Linux システムでサポートしているとします。積み重ね棒グラフにクリティカルが 40%、正常が 60% と表示されている場合、2 つのシステムがクリティカル状況であり、3 つのシステムが正常状況であることを示しています。

その状況のコンポーネント・インスタンスのパーセンテージおよびカウントをポップアップ・ウィンドウに表示するには、棒の部分にマウス・ポインターを合わせます。インスタンスが存在するドメインも、ドメイン (IBM Cloud、クラウド、オンプレミス、ITM、およびその他) ごとの状況カウントとともに表示されます。例えば、5 つの Linux システムのうち、2 つが ITM ドメインに、3 つがクラウド・ドメインにあるとします。クリティカル・システムのうちの一方が ITM ドメインに、他方がクラウド・ドメインにある場合、クリティカル 40% の棒の部分にマウス・ポインターを合わせると、状況のポップアップ・ウィンドウは、1 つのシステムを ITM ドメインに、1 つのシステムをクラウド・ドメインに表示します。

棒をクリックすると、そのコンポーネント・タイプの状況要約ダッシュボードと、モニター対象システムごとのグループ・ウィジェットを開くことができます。

このビューと上で説明した「**集約トランザクション・トポロジー**」を切り替えるには、⊕ ツールをクリックします。

Availability Monitoring

定義済みのアプリケーションが Availability Monitoring のみで構成される場合、1048 ページの『Availability Monitoring へのアクセス』で説明されているように要約ダッシュボードが表示されます。

- 「**グループ**」セクションからサブグループを選択すると、「**インスタンス**」セクションの名前がサブグループのタイトルに変更され、個々のインスタンスの名前が取り込まれます。ナビゲーターのグループ、サブグループ、およびインスタンス・レベルでのダッシュボードについて、および管理対象システムを選択すると開く「**属性の詳細**」タブについては、1087 ページの『グループおよびインスタンス - Application Performance ダッシュボード』および 1092 ページの『カスタムのグラフおよび表の表示と管理』を参照してください。
- ダッシュボード・ウィジェットには、時刻範囲に基づくメトリックを表示するものと、最新のメトリックを表示するものがあります。時間セレクター・バーが表示されている場合は、履歴データのサンプルから値が取得されるグラフまたは表に影響するダッシュボードの時刻範囲を調整できます。詳しくは、1091 ページの『メトリックの調整および時間変化の比較』を参照してください。
  グラフを表示しているときに、プロット点をクリックして、プロット点の値とその他の関連情報を示すツールチップを開くことができます。Internet Explorer バージョン 11 ブラウザーで折れ線グラフを表示した後、ウィンドウの周囲でカーソルを移動すると、ツールチップが引き続き表示される場合があります。この動作が起こった場合は、グラフで数回クリックすることにより、ツールチップを閉じることができます。

- コンソールでは、データは 1 分ごとに自動的に最新表示されます。このアクティビティーは 必要不可欠であるため、 一時停止したり、停止したり、非表示にしたりすることはできません。
- グラフまたは状況要約ボックスに使用可能なデータがない場合は、通知メッセージが表示されます。

イベント

- 選択したナビゲーター項目 (アプリケーション、グループ、サブグループ、またはインスタンス) の状況標識。これは、「**イベント**」タブ・タイトルの横に **14** **3** のように表示され、最高のイベント重大度の数を示します。しきい値の重大度は、以下の表に示すように統合されています。例えば、「**イベント**」 は、最高の重大度のイベントは軽微または警告であることを意味しています。

「イベント」タブ	限界値重大度
クリティカル	致命的およびクリティカル
警告	軽微および警告
正常	不明

  ご使用の管理対象環境に IBM Operations Analytics - Predictive Insights が含まれているときに異常が検出されると、イベントがオープンされます。状況標識上にひし形アイコン (など) が重なって表示され、Operations Analytics - Predictive Insights が少なくとも 1 つの異常を検出したことが通知されます。例えば、「**イベント**」 は、最高の重大度のイベントが 警告であり、少なくとも 1 つの異常イベントがオープンであることを示します。
- 「**イベント**」タブをクリックすると、合計イベント数、重大度タイプごとの数、重大度のパーセント・ゲージについて、要約を表示することができます。詳しくは、1109 ページの『イベント状況』を参照してください。

カスタム・ビュー

作成および保存したページは、選択したアプリケーションに関連付けられます。例えば、Cloud APM ガイド付きデモのインベントリー管理アプリケーションには、Linux OS、MySQL、Node.js、Hadoop、および Ruby の各モニター・エージェントがあります。カスタム・ページは、ナビゲーターの任意のレベル (アプリケーションからインスタンスまで) で作成および保存し、作成したのと同じレベルで開くことができます。特定のレベルで作成されたページは、同じレベルでのみ開くことができます。ウィジェットで使用可能なメトリックは、アプリケーション内のリソースのいずれかから取得できます。デモのインベントリー管理の例を使用すると、Ruby エージェントからの表や Linux OS エージェントからのグラフなどを含むページを作成できます。

- 「**カスタム・ビュー**」タブは、「**すべてのマイ・アプリケーション**」からアプリケーションを選択すると、ナビゲーターのすべてのレベルで使用できます。
- 「**カスタム・ビュー**」タブを開くと、「**カスタム・ページのテンプレートの選択**」ウィンドウが表示されるか、デフォルト・ページが既に設定されている場合はそのページが表示されます。
  - 「**カスタム・ページのテンプレートの選択**」ウィンドウで、ページを作成するためのテンプレートを選択できます。
  - デフォルト・ページで、 をクリックして新規ページを作成できます。
- デフォルト・ページで、ページ・リスト内の をクリックして、リストからいずれかの保存済みページを選択します。
- 「カスタム・ビュー」タブに表示されるオプションは、ページを編集中か表示中かによって異なります。ページの編集については、1113 ページの『カスタム・ページの作成および管理』を参照してください。ページの表示については、1120 ページの『カスタム・ページの表示』を参照してください。

集約トランザクション・トポロジー・ウィジェットの操作

「**集約トランザクション・トポロジー**」ウィジェットは、選択したアプリケーションのリソース階層を表示するために使用します。表示を調整および移動して、各コンポーネントの状況や他のコンポーネントとの関係を表示したり、ノードの対応するダッシュボードを開いたりできます。

始める前に

Application Performance ダッシュボードでアプリケーションを選択すると、そのアプリケーションに含まれているモニター対象リソースに応じて、1 つ以上のグラフとともに「**状況の概要**」タブが表示されます。

トランザクション・トラッキングをサポートする以下のエージェントについて、「**集約トランザクション・トポロジー**」が表示されます。

- DataPower エージェント
- HTTP Server エージェント
- IBM Integration Bus エージェント
- J2SE データ・コレクター
- JBoss エージェント
- Liberty データ・コレクター
- Microsoft .NET エージェント
- Microsoft SQL Server エージェント
- Node.js データ・コレクター
- Response Time Monitoring エージェント
- SAP NetWeaver Java Stack エージェント
- Tomcat エージェント
- WebLogic エージェント (Linux と Windows のみ)
- WebSphere Applications エージェント
- WebSphere MQ エージェント

「**集約トランザクション・トポロジー**」には、トポロジー機能をサポートするすべてのモニター対象リソースのノード・オブジェクトが表示されます。

手順

以下のいずれかの手順を使用して「**集約トランザクション・トポロジー**」ウィジェットを操作し、ノードに関連付けられたダッシュボードを開きます。

- リンクされたダッシュボードを開くには、トポロジー・ノードをダブルクリックします。ノードを右クリックし、ダッシュボードのドリルダウン・オプションである「「**トランザクション要約**」**ページに移動**」 または「「**コンポーネント・インスタンス**」**ページに移動**」のいずれかを選択するか、「**プロパティー**」を選択して管理対象システムについての 情報を表示することもできます。
- トポロジーの表示サイズを大きくするには、「**ズームイン**」をクリックします。ノードが選択されていない場合は、「**アクション**」 > 「**ズームイン**」をクリックすることもできます。
- トポロジーの表示サイズを小さくするには、「**ズームアウト**」をクリックします。ノードが選択されていない場合は、「**アクション**」 > 「**ズームアウト**」をクリックすることもできます。
- 現在のウィジェット・スペース内に収まるようにトポロジーの表示サイズを調整するには、「**コンテンツを適合させる**」をクリックします。ノードが選択されていない場合は、「**アクション**」 > 「**コンテンツを適合させる**」をクリックすることもできます。
- トポロジー・ノードをフィルタリングするには、フィルター・バーでいずれかのインディケーターを選択します。フィルターのオン/オフを切り替えたり、複数のフィルターを選択したりできます。フィルターと一致しないプロパティーを持つノードは表示が暗くなり、フィルターと一致するノードは表示されたままになります。
  - 「正常」、「警告」、「クリティカル」、または 「不明」(ノード状況別にフィルタリングするため)。

- 環境によってフィルタリングするには、以下を選択します。「**クラウド (Cloud)**」(IBM Cloud Application Performance Management)、「**ITM**」(IBM Tivoli Monitoring)、「**オンプレミス**」(IBM Cloud Application Performance Management, Private) 、「**専用クラウド (Private Cloud)**」(IBM Cloud Private)、または「**公開クラウド (Public Cloud)**」(IBM Cloud)。
- 「フィルター」(カスタム・フィルターを追加するため)。

- トポロジー・ウィジェットの表示領域を広げるには、ナビゲーターまたは周囲のグラフ・ウィジェットの「**セクションの省略 (Collaspe Section)**」をクリックするか、ウィジェットの枠をドラッグします。

グループおよびインスタンス - Application Performance ダッシュボード

選択されたアプリケーションのグループ、サブグループ、またはインスタンスについてこのダッシュボードを使用して、管理対象システムの全体的な状況を取得します。選択したインスタンスのメトリックを示す詳細なダッシュボードにドリルダウンして、カスタムのグラフと表を作成できます。

Application Performance ダッシュボードの「**すべてのマイ・アプリケーション**」の下でアプリケーションを選択すると、「**状況の概要**」タブと「**イベント**」タブが表示されます。

ナビゲーターの「**グループ**」セクションに、定義済みのアプリケーションの構成要素に応じて、可能性のあるグループが 1 つ以上リストされます。

ナビゲーターおよびバナーのエレメントの説明については、1080 ページの『ナビゲーター』、1080 ページの『検索』、1080 ページの『アクション』、および 1080 ページの『 ヘルプ』を参照してください。

状況の概要

グループおよびサブグループ

- グループを選択するか、 グループを展開してサブグループを選択し、アプリケーション内の各管理対象システムの、要約グループのウィジェットを 表示します。サブグループを選択すると、「**状況の概要**」タブの要約グループ・ウィジェットは、そのサブグループに固有のものになります。
- インストールされているモニター製品に応じて、以下の事前定義グループが使用可能です。

  Availability Monitoring
  このグループは、カスタム・アプリケーション用に表示されます。 Availability Monitoring のナビゲーション動作は、「**コンポーネント**」および「**トランザクション**」グループとは異なります。

  Availability Monitoring アドオンおよびダッシュボードについては、1047 ページの『Availability Monitoring』で説明されています。

  コンポーネント
  このグループは、Response Time Monitoring エージェント、Synthetic Playback エージェント、および Availability Monitoring を除くすべてのアプリケーションに対して表示されます。

  「**コンポーネント**」には、モニター対象のソフトウェア・コンポーネントごとに、選択したアプリケーションをサポートするサブグループがあります。

  トランザクション
  このグループには、「**エンド・ユーザー・トランザクション**」および「**シンセティック・トランザクション**」(2017 年 8 月リリースよりも前の IBM Website Monitoring on Cloud) サブグループが含まれます。詳しくは、 トランザクション・モニターおよび Synthetic Playback エージェントのヘルプまたは APM Developer Center の Agent metrics/Reference PDFs にあるリファレンスの PDF を参照してください。
- 「**グループ**」セクションで「**コンポーネント**」またはサブグループを選択すると、「**状況の概要**」タブの表示内容が変わり、各管理対象リソースのグループ・ウィジェットを示す要約ダッシュボードが表示されます。ソース環境は、以下のように表示されます。「**クラウド (Cloud)**」(IBM Cloud Application Performance Management)、「**ITM**」(IBM Tivoli Monitoring)、「**オンプレミス**」(IBM Cloud Application Performance Management, Private) 、「**専用クラウド (Private Cloud)**」

(IBM Cloud Private)、または「**公開クラウド (Public Cloud)**」(IBM Cloud)。「**インスタンス**」セクションの名前がサブグループのタイトルに変更され、個々のインスタンスの名前が取り込まれます。

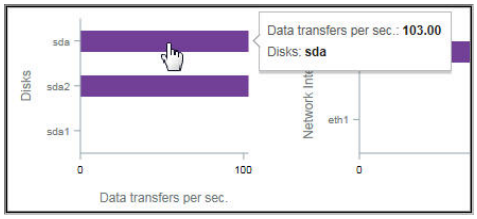
アプリケーションに多くの管理対象システム・インスタンスがある場合、多くグループ・ウィジェットが表示されます。リストをスクロールして、それらすべてを表示できます。コンポーネント・サブグループ (Windows OS など) のリストから管理対象システムを選択して、表示内容を同じ管理対象システム・タイプに絞り込むことができます。管理対象システム・インスタンスをフィルタリングすることもできます。

Firefox ブラウザーのみ: エージェントと帯域幅の数に応じて、「コンポーネント」ページをスクロールダウンするときに、リソース・ページをロードするスクリプトの実行に長い時間がかかることを示すポップアップ・メッセージが表示されることがあります。オプション「今後この表示をしないようにする」を選択して、メッセージを使用不可にし、ウィジェットを開き続けます。別の方法として、アドレス・ボックスに `about:config` と入力し、**dom.max_script_run_time** を検索して、タイムアウト値 (秒) を増やすこともできます。値を 0 (ゼロ) にすると、タイムアウトが使用不可になります。

インスタンス

- グループ・ウィジェットの内部をクリックするか、ナビゲーターでインスタンス名を選択すると、管理対象リソースの詳細ダッシュボードが開きます。
- ナビゲーターに多くのインスタンスが表示されている場合、インスタンス・ツールバーの検索フィールド を使用してください。入力していくと、一致しないすべてのインスタンスが表示から除外されます。
- Application Performance ダッシュボードの自動最新表示を一時停止するには、「インスタンス」ツールバーで 「**一時停止**」をクリックします。自動最新表示を再開するには、 「**再開**」をクリックします。
- 管理対象システム用に表示されるウィジェットおよび KPI は、エージェントのバージョンによって異なる場合があります。管理対象システムにインストールされているエージェントが 以前のバージョンである場合は、そのエージェントの現行バージョンほど多くの情報を表示できない可能性があります。データがないときは、グラフや表に 1 つ以上の KPI ではなくメッセージが表示されます。原因は単純であり、 そのタイム・スパンでデータが報告されていないなどです。また、原因が、データ・セットやグラフまたは表に含まれる属性をサポートしないバックレベルのエージェントに関するものである場合もあります。
  直近の Cloud APM サーバーの再始動以降に更新されたエージェント・ダッシュボードのリストを表示するには、「**アクション**」 > 「**ダッシュボードのログ**」を選択します。
- ダッシュボード・ウィジェットには、時刻範囲に基づくメトリックを表示するものと、最新のメトリックを表示するものがあります。時間セレクター・バーが表示されている場合は、履歴データのサンプルから値が取得されるグラフまたは表に影響するダッシュボードの時刻範囲を調整できます。詳しくは、1091 ページの『メトリックの調整および時間変化の比較』を参照してください。
  グラフを表示しているときに、プロット点をクリックして、プロット点の値とその他の関連情報を示すツールチップを開くことができます。Internet Explorer バージョン 11 ブラウザーで折れ線グラフを表示した後、ウィンドウの周囲でカーソルを移動すると、ツールチップが引き続き表示される場合があります。この動作が起こった場合は、グラフで数回クリックすることにより、ツールチップを閉じることができます。

- 棒がないグラフが表示される場合は、そのデータ・ポイントの値が 0 (ゼロ) であることを示しています。



- IBM Cloud Application Performance Management, Advanced のユーザーは、その他の診断ダッシュボードを使用できます。これらのダッシュボードにアクセスするには、詳細ダッシュボードのグループ・ウィジェットから「**診断**」リンクをクリックします。

  制約事項 : 診断ダッシュボードを開く対象の管理対象システムは、IBM Cloud APM ドメインに存在している必要があります。管理対象システムが IBM Cloud または IBM Tivoli Monitoring のソース・ドメインにある場合、診断ダッシュボードは使用できません。『950 ページの『Cloud APM エージェントと Tivoli Monitoring エージェントの共存』』も参照してください。

- ご使用の環境に Synthetic Playback エージェントが含まれる場合、「**アクション**」メニューから、エージェント・インスタンスに関する Cloud APM レポートを起動することができます。

イベント

- 選択したナビゲーター項目 (アプリケーション、グループ、サブグループ、またはインスタンス) の状況標識。これは、「**イベント**」タブ・タイトルの横に **14** **3** のように表示され、最高のイベント重大度の数を示します。しきい値の重大度は、以下の表に示すように統合されています。例えば、「**イベント**」 は、最高の重大度のイベントは軽微または警告であることを意味しています。

「イベント」タブ	限界値重大度
クリティカル	致命的およびクリティカル
警告	軽微および警告
正常	不明

  ご使用の管理対象環境に IBM Operations Analytics - Predictive Insights が含まれているときに異常が検出されると、イベントがオープンされます。状況標識上にひし形アイコン (など) が重なって表示され、Operations Analytics - Predictive Insights が少なくとも 1 つの異常を検出したことが通知されます。例えば、「**イベント**」 は、最高の重大度のイベントが 警告であり、少なくとも 1 つの異常イベントがオープンであることを示します。

- 「**イベント**」タブをクリックすると、合計イベント数、重大度タイプごとの数、重大度のパーセント・ゲージについて、要約を表示することができます。詳しくは、1109 ページの『イベント状況』を参照してください。

カスタム・ビュー

作成および保存したページは、選択したアプリケーションに関連付けられます。例えば、Cloud APM ガイド付きデモのインベントリー管理アプリケーションには、Linux OS、MySQL、Node.js、Hadoop、および Ruby の各モニター・エージェントがあります。カスタム・ページは、ナビゲーターの任意のレベル (アプリケーションからインスタンスまで) で作成および保存し、作成したのと同じレベルで開くことができます。特定のレベルで作成されたページは、同じレベルでのみ開くことができます。ウィジェットで使用可能なメトリックは、アプリケーション内のリソースのいずれかから取得できます。デモのインベントリー管理の例を使用すると、Ruby エージェントからの表や Linux OS エージェントからのグラフなどを含むページを作成できます。

- 「**カスタム・ビュー**」タブは、「**すべてのマイ・アプリケーション**」からアプリケーションを選択すると、ナビゲーターのすべてのレベルで使用できます。
- 「**カスタム・ビュー**」タブを開くと、「**カスタム・ページのテンプレートの選択**」ウィンドウが表示されるか、デフォルト・ページが既に設定されている場合はそのページが表示されます。

– 「**カスタム・ページのテンプレートの選択**」ウィンドウで、ページを作成するためのテンプレートを選択できます。
– デフォルト・ページで、 + をクリックして新規ページを作成できます。

- デフォルト・ページで、ページ・リスト内の をクリックして、リストからいずれかの保存済みページを選択します。
- 「カスタム・ビュー」タブに表示されるオプションは、ページを編集中か表示中かによって異なります。ページの編集については、1113 ページの『カスタム・ページの作成および管理』を参照してください。ページの表示については、1120 ページの『カスタム・ページの表示』を参照してください。

属性の詳細

- ナビゲーターの「**インスタンス**」セクション (名前は選択したサブグループの名前に変更されます) からコンポーネント・インスタンスを選択するか、要約グループのウィジェット内をクリックすることで、「**属性の詳細**」タブが表示されます。
- エージェントに対して「グラフ」ページまたは「表」ページが保存されている場合、直近に開いたページが、選択したコンポーネント・インスタンスのメトリック付きで表示されます。異なる保存されたページを「**マイ・ページ**」 › または 「**共有ページ**」 › から選択するには、タイトル ⌄ をクリックします。
- グラフまたは表を編集し、「**結果のプレビュー**」をクリックして、選択した属性を持つグラフや表をレンダリングできます。オプションについて詳しくは、1093 ページの『カスタムのグラフまたは表のページの作成』 を参照してください。

コンポーネント・ダッシュボードのグループ・ウィジェットの編集

「**コンポーネント**」ダッシュボード (ナビゲーターの「**グループ**」セクションで選択されています) で表示するグループ・ウィジェットのしきい値を編集することができます。表示するグループ・ウィジェットとその位置を制御し、コンポーネント状況の判別にウィジェットのしきい値を含める必要があるかどうかを決定することもできます。

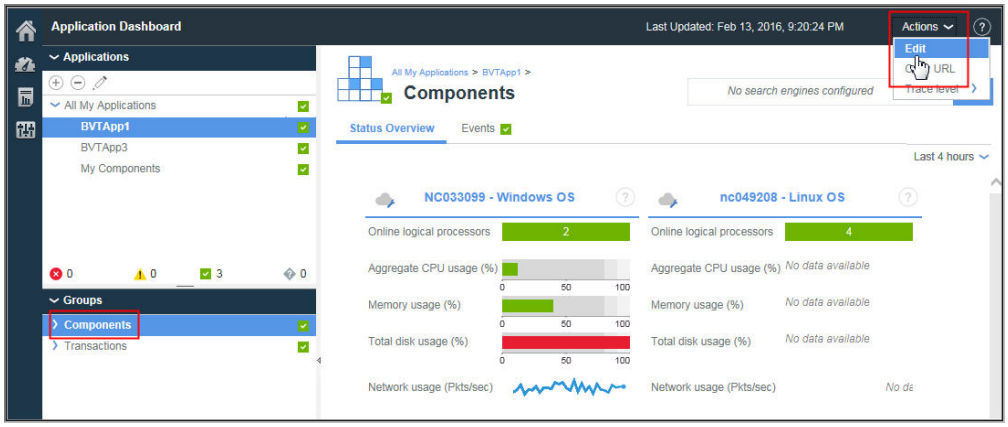
このタスクについて

このタスクでは、定義済みアプリケーションの「コンポーネント」グループ・ダッシュボードおよびその構成要素の要約グループ・ウィジェットを編集します。コンポーネント・グループ・エディターは、事前定義アプリケーション「**マイ・コンポーネント**」には使用できません。定義済みアプリケーションについて詳しくは、1098 ページの『アプリケーションの管理』を参照してください。

ユーザー ID には、Application Performance ダッシュボードに対する変更権限とアプリケーションに対する作成権限も必要です。詳しくは、1003 ページの『役割と権限』を参照してください。

手順

1. 「**パフォーマンス**」 メニューから「Application Performance ダッシュボード」を開いた後、「**すべてのマイ・アプリケーション**」ダッシュボードから、要約グループ・ウィジェットを編集するアプリケーションを選択します。
2. ナビゲーターの「**グループ**」セクションで、「**コンポーネント**」をクリックしてダッシュボードを開き、アプリケーション内のすべてのコンポーネントに関するグループ・ウィジェットを表示します。
3. 「**アクション**」 > 「**編集**」をクリックして、「**コンポーネント**」グループ内のグループ・ウィジェット用のエディターを開きます。
   「**編集**」オプションは、「**コンポーネント**」ダッシュボードが開いている場合にのみ表示されます。



4. グループ・ウィジェットに、以下の変更を行います。
   - ビューからグループ・ウィジェットを削除するには、⊖をクリックします。
   - ウィジェットの要約しきい値を変更するには、⚙「**設定**」をクリックし、「**しきい値**」タブを選択して、重大度 (クリティカル、警告、または正常) に対するしきい値を変更します。グループ・ウィジェットのしきい値を編集したら、「**完了**」をクリックします。
   - ウィジェットを追加するには、➕をクリックし、対象とするものが表示されるまでアプリケーション・アイコンをクリックします。 次に、グループ・ウィジェット内をクリックして選択してから「**追加**」をクリックします。
   - ウィジェットのサイズを変更するには、ハンドル・アイコン ◢ をドラッグします。ウィジェットのサイズを変更しても、テキストのサイズもウィジェットの 高さも変更されません。
   - ウィジェットを移動するには、ウィジェットを新しい場所にドラッグします。
5. 変更内容を保存してエディターを閉じるには、「**保存**」をクリックします。変更内容を破棄するには「**キャンセル**」をクリックします。

タスクの結果
選択したアプリケーションの「**コンポーネント**」ダッシュボードが、新しい設定で表示されます。

次のタスク
ナビゲーターでグループまたはサブグループが 選択されたときのダッシュボードについて詳しくは、1087 ページの『グループおよびインスタンス - Application Performance ダッシュボード』を参照してください。
モニター対象コンポーネントのダッシュボードについて詳しくは、「**アプリケーション・ダッシュボード**」バナーの ? ボタンをクリックしてください。

メトリックの調整および時間変化の比較
ダッシュボード・グラフには、時刻範囲に基づくメトリックを表示するものと、最新のメトリックのみを表示するものがあります。 管理対象システム・インスタンスの「**状況の概要**」タブに時間セレクターが表示されている場合、履歴データのサンプルから値が取得されるグラフの時刻範囲を調整できます。複数の日数分のデータが収集されて、折れ線グラフで表される属性の場合は、今日の値を前日と比較できます。

始める前に
以前の日の時刻範囲と比較する場合、どの程度まで遡ることができるかは、Cloud APM サーバーによって保存された日数とページに表示されるデータのタイプによって異なります。。Cloud APM の有料サブスクライバーの場合、 データ・サンプルは大部分のリソース・モニター・データセットについて 8 日間保存されます。 正確な数値は、エージェントまたはデータ・コレクターの属性ヘルプおよび参照 PDF で公開されています (43 ページの『第 2 章 PDF 資料』を参照)。Cloud APM の無料サブスクライバーの場合、リソース・モニター・データ・サンプルは 2 日間保存されます。 Response Time Monitoring エージェント または ミドルウェア・エージェントからのトランザクション・トラッキング・データは過去 24 時間分 (場合によっては過去 4 時間分) しか表示できず、その保存期間は変更できません。

手順

以下の手順に従って、管理対象リソース・インスタンスの折れ線グラフに表示される時刻範囲を調整したり、以前の日の同じ時刻範囲の値を比較したりします。

1. Application Performance ダッシュボードが表示されない場合は、「**パフォーマンス**」メニューから選択します。
2. 履歴折れ線グラフを示しているインスタンスのダッシュボード・ページに移動し、「**過去 4 時間**」の時間セレクターをクリックします。
3. 以下のオプション (複数可) から選択します。
   - 表示する時刻範囲を変更するには、 「**過去 4 時間**」、「**過去 12 時間**」、または「**過去 1 日**」を選択します。
   - 折れ線グラフに表示されている時刻範囲と、別の日のメトリックを比較するには、「**比較対象**」を選択し、最大でポップアップ・カレンダーで選択可能と示されている日数まで遡って以前の日を選択します (選択できない日には線が引かれています)。
   - モニター対象環境内のすべての定義済みアプリケーションのダッシュボードにこの時刻範囲を適用するには、 「**すべてのアプリケーション**」を選択します。それ以外の場合は、 設定を「**このアプリケーションのみ**」のままにしておけば、現在のアプリケーション (「マイ・コンポーネント」など) のみに時刻範囲が適用されます。「**比較対象**」の選択は、現行ページのみに有効です。

タスクの結果

- 比較せずに履歴データを表示している場合は、現在のアプリケーション (またはすべてのアプリケーション) のすべてのダッシュボードに変更の影響が及びます。
- 比較を表示している場合は、現行ページの折れ線グラフのみに影響が及びます。各 KPI の線が引かれ、選択した日のメトリックが表示されます。一部の折れ線グラフは、グラフ上の水準点で示されているように (「使用可能な比較がありません」)、比較に使用できません。これは、指定された日のデータがまだ収集されていない、新しい管理対象リソースで発生することがあります。より最近の日付を選択して、比較してください。
- 履歴データが収集されていないウィジェットには、引き続き最新の値が表示されます。
- データ・ポイントは、選択された時刻範囲のグラフの全長に沿って分布しています。一番初めのタイム・スタンプから始まり、最近のタイム・スタンプで終わるタイム・スタンプが軸ラベルに表示されます。最も古いデータ・サンプルを調べて、履歴データの部分的な範囲または全範囲のどちらが表示されているのかを確認してください。
- グラフと表に送信されたデータは、GMT (グリニッジ標準時) に正規化されます。「**タイム・スタンプ**」軸には、ブラウザーのタイム・ゾーンに基づくタイム・スタンプが表示されます。タイム・ゾーンで標準時間と夏時間が使用されている場合は、移行時間中に表示されるタイム・スタンプが 1 時間ずらされます。例えば、スペインで折れ線グラフを表示しているときに、時間が標準時間の午前 2 時から夏時間の午前 3 時に変わったとします。GMT のデータと現地時間のタイム・スタンプの間に差異があるため、タイム・スタンプに午前 2 時から午前 3 時までの 1 時間のギャップが生じます。同じグラフを、夏時間の午前 3 時から標準時間の午前 2 時に切り替わる間にニュージーランドで表示すると、午前 2 時から午前 3 時までのタイム・スタンプが繰り返して表示されます。

カスタムのグラフおよび表の表示と管理

Application Performance ダッシュボードは、管理対象システムの重要パフォーマンス指標の事前定義ダッシュボードを表示します。コンポーネント・インスタンスのダッシュボードを表示しているとき、「**属性の詳細**」タブを使用して、保存されているグラフや表のページを表示したり、その他のページを作成および管理したりすることができます。

例えば、要約ダッシュボードでクリティカルな指標が観察されたときに、その状態が発生しているインスタンスにドリルダウンします。そこから、使用中の CPU 率をプロットするグラフを追加して、時間の経過とともに何が発生しているかを確認します。選択したコンポーネント・インスタンスの使用可能な属性について詳細を表示したり、管理対象システム・インスタンスを開いたときに毎回表示するカスタムのグラフまたは表をエージェントとともに保存したりすることができます。

エージェントのデータ・セットのサブセットと属性は、カスタムのグラフおよび表で使用できます。これらの属性は、ダッシュボードに表示すると最も便利です。属性一式は、カスタムしきい値で使用できます (987 ページの『しきい値マネージャー』を参照してください)。

パフォーマンスを改善し、冗長性を削減するために、エージェントは、特定のデータ・セットについて「属性の詳細」に表示される行数を制限します。エージェント・ヘルプおよび参照 PDF のデータ・セットの説明で、デフォルトのデータ・サンプルにより、Cloud APM サーバーに送信される行数が制限されるかどうかが示されています。

視覚障害があるユーザーに対して、ヒストリカル表を作成する機能で、折れ線グラフの代わりとなるものを提供します (スクリーン・リーダー・ソフトウェアなどの支援技術では折れ線グラフを認識できません)。このため、Response Time Monitoring エージェントおよび Synthetic Playback エージェントのトランザクション・インスタンスでは、ヒストリカル表を作成するための「**属性の詳細**」タブが使用可能です。詳しくは、1095 ページの『キーボード・コントロールを使用したカスタム表の作成の例』を参照してください。

カスタムのグラフまたは表のページの作成

コンポーネント・インスタンスの Application Performance ダッシュボードを表示しているときに、「**属性の詳細**」タブを選択して、保存したグラフや表のページを表示したり、その他のページを作成して管理したりすることができます。

このタスクについて

Application Performance ダッシュボードのホーム・ページから管理対象リソースのインスタンスにドリルダウンすると、ダッシュボード・ページの「**状況の概要**」タブおよび「**イベント**」タブに「**属性の詳細**」タブが追加されます。

ここでは、コンポーネント・インスタンスのカスタムのグラフおよび表の作成について説明します。Response Time Monitoring エージェントおよび Synthetic Playback エージェントのトランザクション・インスタンスに対してこの手順を実行できますが、次の制限が適用されます。ヒストリカル表のみ (グラフはなし)。「**データ・セット**」や「**属性**」のリストをフィルターに掛けることはできない。すべての属性が選択される (個々の属性を選択したり選択解除したりすることはできない)。ページを保存することはできない。Synthetic Playback エージェントの場合は時間セレクターの「**前へ**」オプションを使用できない。

手順

選択したコンポーネント・インスタンスで使用可能な任意のデータ・セットからグラフまたは表を作成するには、次の手順を実行します。

1. 「**パフォーマンス**」メニューから、「Application Performance ダッシュボード」を開いた後、管理対象リソースのインスタンスにドリルダウンします。

   選択したシステムが、ナビゲーターの「**インスタンス**」セクションで強調表示されます。名前は、「**Ruby アプリケーション**」などのコンポーネント・タイプに応じた名前になっています。
2. 「**属性の詳細**」タブをクリックします。

   このモニター・エージェント・タイプについて保存されたグラフや表がない場合、「**リアルタイム表**」が選択されます。「リアルタイム」は、複数の行を返すデータ・セットについて使用でき、表にのみ使用可能です。
3. 保存したグラフまたは表が表示された場合は、「**新規**」をクリックします。
4. 「表題」フィールドに、グラフまたは表のページの名前を入力します。

   次の文字をタイトルに使用しないでください: ! " % & ' * ? < > } { ¥。
5. 時間の経過にともなえデータ・サンプルを表示する場合は、タイプを「**ヒストリカル**」に変更します。

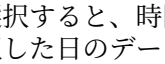
   「**グラフ**」オプションが使用可能になります。
6. 「**ヒストリカル**」を選択し、表ではなくグラフをレンダリングする場合は、「**グラフ**」をクリックします。
7. 「**データ・セット**」リストで、表示する属性タイプのラジオ・ボタンを選択します。リストが長い場合は、データ・セット名に含まれるテキストを フィルター・ボックスに入力して、リストを絞り込みます。

例えば Linux OS エージェントの場合に「config」と指定すると、データ・セットがフィルターに掛けられ、**Linux_CPU_Config** データ・セットおよび **Linux_OS_Config** データ・セットのみが表示されます。

8. 属性をグラフまたは表に含めるには、「**属性**」リストで、名前の横にあるチェック・ボックスを選択します。すべての属性を含めるには、リストの先頭にあるチェック・ボックスを選択します。特定の属性を探すには、 フィルター・ボックスにテキスト (「パーセント」など) を入力します。
例えば KLZ_VM_Stats データ・セットで「%」を適用すると、リストがフィルターに掛けられ、「**フリー仮想ストレージ (%)**」とさらに他に 5 つの「%」属性が表示されます。
グラフでは数値のみをプロットできます。テキストまたは時間の属性は使用できません。
9. 「**結果のプレビュー**」をクリックすると、選択したデータ・セットのページが生成されます。ページでは、各属性ごとに表の 1 列、またはグラフの 1 つの折れ線があり、各データ・サンプルに対して 1 つの行またはプロット点があります。
すべての値の総計を表す 1 つの行またはプロットされた線も表示されます。
10. グラフの線または表の行を非表示にするには、次のいずれかの手順を実行します。

オプション	説明
グラフ	凡例の名前の横にあるチェック・ボックスをクリアすることにより、メトリック (グラフ内の折れ線) を非表示します。メトリックを表示するには、チェック・ボックスを選択します。
表	フィルター・ボックスにフィルターに掛ける条件の値を入力することで、表示される行の数を削減します。 1096 ページの『表フィルターの定義』の説明に従って、拡張フィルターを作成することもできます。

11. 時刻範囲を調整するには、次のように時間セレクターを使用します。

オプション	説明
リアルタイム	表の場合にのみ、最後のデータ・サンプリングのみを最新表示します。
2 時間	過去 2 時間に所定の間隔で取られたデータ・サンプルごとにポイントをプロットするか、行を追加します。
4 時間	過去 4 時間に所定の間隔で取られたデータ・サンプルごとにポイントをプロットするか、行を追加します。
12 時間	過去 12 時間に所定の間隔で取られたデータ・サンプルごとにポイントをプロットするか、行を追加します。
24 時間	過去 24 時間に所定の間隔で取られたデータ・サンプルごとにポイントをプロットするか、行を追加します。
前へ	同じ期間の「**前日**」、「**2 日前**」、または「**1 週間前**」までの任意の日からのデータを含めるためのオプションのサブメニュー。例えば、現在 8 月 22 日午後 2 時 10 分であり、過去 4 時間を表示するようにグラフまたは表を設定してあるとします。「**前へ**」 > 「**1 週間前**」を選択すると、その日の午前 10 時 10 分から午後 2 時 10 分までと、8 月 15 日の午前 10 時 10 分から午後 2 時 10 分までのデータ・ポイントが表示されます。 前日を選択すると、時間セレクターに  のようにアスタリスク (*) が表示され、選択した日のデータでビューが再生成されます。

- ヒストリカル・データの最大保存期間を 8 日間を超える日数に増加させた場合でも、今日から最大 1 週間前までのデータしか表示できません。
- ヒストリカル・グラフは、選択した時刻範囲 (例: 過去 4 時間) について、最も古いデータ・サンプルから最新のデータ・サンプルまでを使用してプロットされます。前日を選択すると、前日の選択した時刻範囲および今日の時刻範囲のデータが表示されます。過去の日付からその日までのすべての日に、タイム・スタンプとともにプロット・ポイントがデータ・サンプルなしで表示されます。
- ヒストリカル表は、日時の降順にプロットされます。Response Time Monitoring エージェントの場合に前日を選択すると、前日の各列が複製され、列見出しに「前」と表示されます。

- 選択した時刻範囲に関係なく、最大 11,000 行を表示することができます。例えば、2 時間で 7,000 行が送信されるデータ・セットから 12 時間を表示することを選択した場合、返されるヒストリカル・データは 3 時間分より少なくなり、最も古いデータ・サンプルが表示されます。

12. グラフまたは表について、保存またはその他の変更を実行するには、次のいずれかのオプションを選択します。

オプション	説明
編集	次の任意の変更を行うためにエディターに戻ります。 • 表題の編集 •「**リアルタイム**」または「**ヒストリカル**」のデータ・サンプルへの変更 •「**グラフ**」または「**表**」への変更 • 異なる「**データ・セット**」または「**属性**」、あるいはその両方の選択
新規	現在のビューに対して行われたがまだ保存されていない変更を破棄し、新しいグラフまたは表を作成するための選択ページに戻ります。
キャンセル	現在のグラフまたは表のページの編集セッションをキャンセルします。
削除	ページを削除します。「**削除**」は、ページが保存され、現在の編集がキャンセルされた後にのみ使用できます。
自分用に保存	自分のユーザー ID のみが表示するためにグラフまたは表を保存します。他のユーザーは保存されたページを表示できません。
共有のために保存	Cloud APM コンソールにログインしている任意のユーザー ID で表示するためにグラフまたは表を保存します。

自分が保存したビューの場合は、タイトルの横に開いた鍵 が表示されます。別のユーザーが保存した、自分が編集権限を持っていないビューには、閉じた鍵のアイコンが表示されます。

タスクの結果
カスタムのグラフまたは表を保存すると、それが保存されたページのリストに追加されます。次に同じデータ・ソース・タイプ (WebSphere Applications など) のインスタンスを選択して「**属性の詳細**」タブを選択したときに、保存されたページのうち、直近に開いたページが表示されます。異なる保存されたページを「**マイ・ページ**」 › または 「**共有ページ**」 › から選択するには、タイトル ⌄ をクリックします。

次のタスク
その他のグラフや表のページを作成して管理するために、この手順を繰り返します。

マウスをクリックする代わりにキーボード・コントロールを使用して表を作成するためのガイドについては、1095 ページの『キーボード・コントロールを使用したカスタム表の作成の例』を参照してください。

キーボード・コントロールを使用したカスタム表の作成の例
視覚障害があるユーザーは、「**属性の詳細**」ダッシュボード・タブを使用して、履歴折れ線グラフに対するアクセス可能な代替手段としてヒストリカル表を作成することができます。履歴折れ線グラフは、スクリーン・リーダー・ソフトウェアなどの支援技術では認識できません。

このタスクについて
以下の例は、Synthetic Playback エージェントにより報告されるトランザクションのヒストリカル表を作成するためのキーボード・コントロールの使用方法を示しています。このエージェントについて詳しくは、1028 ページの『Website Monitoring でのシンセティック・トランザクションおよびシンセティック・イベントの管理』を参照してください。

Tab キーを押すと、フォーカスがアプリケーション・ウィンドウのフィールドの隣または次のセクションに、左から右および上から下の順序で移動します。これらのステップを使用して、エージェントを含むアプリケーションで「**マイ・トランザクション**」を置き換えることにより Response Time Monitoring エージ

ェントのトランザクション表を生成するか、他のアプリケーションで「**マイ・トランザクション**」を置き換えてから「**コンポーネント**」グループを選択することによりコンポーネント・インスタンスの表を生成することができます。

手順

以下のステップに従って、キーボード・ショートカットを使用して「**属性の詳細**」タブで Synthetic Playback エージェントからのトランザクションの表を作成します。

1. IBM Cloud Application Performance Management にログインします。
   フォーカスがナビゲーション・バーに置かれます。
2. Application Performance ダッシュボードを開くために、下矢印を押してフォーカスを「**パフォーマンス**」メニューに移動し、Enter キーを押してそれを選択し、下矢印を押してフォーカスを「**Application Performance ダッシュボード**」オプションに移動し、再度 Enter キーを押します。
3. 「**シンセティック・トランザクション**」ダッシュボード・ページを開くために、ナビゲーターにフォーカスが移動するまで Tab キーを繰り返し (約 7 回) 押し、下矢印を押してフォーカスを「**マイ・トランザクション**」事前定義アプリケーションに移動し、Enter キーを押します。
4. 「**トランザクション詳細**」ダッシュボード・ページを開くために、「**シンセティック・トランザクション**」トランザクション・インスタンスのナビゲーターの「**インスタンス**」セクションにフォーカスが移動するまで Tab キーを (約 10 回) 押し、Enter キーを押します。
   「**属性の詳細**」タブがダッシュボードに表示されます。
5. 「**属性の詳細**」タブを開くために、「**状況の概要**」タブにフォーカスが置かれるまで Tab キーを (約 6 回) 押し、「**属性の詳細**」タブにフォーカスが置かれるまで右矢印を押します。
   「**ヒストリカル**」 「**表**」および「**トランザクションの可用性 (時系列)**」データ・セットのすべての属性が選択されます。
6. 表を生成するために、「**結果のプレビュー**」ボタンにフォーカスが置かれるまで Tab キーを (約 18 回) 押し、Enter キーを押します。

タスクの結果

「**トランザクションの可用性 (時系列)**」属性が表に表示されます。表には、各属性の列と、過去 4 時間の各データ・サンプルに対して 1 行があります。

次のタスク

- 表示される行の数を減らすには、Tab キーを押して 「**フィルター**」テキスト・ボックスにフォーカスを置き、フィルタリングの条件として部分的なテキストまたは完全なテキストあるいはタイム・スタンプ値を入力します。
- 時刻範囲を変更するには、フォーカスを時間セレクターの「**過去 4 時間**」 ドロップダウン・メニューに移動し、他のオプションを選択します。詳しくは、1093 ページの『カスタムのグラフまたは表のページの作成』のステップ 1094 ページの『11』 を参照してください。
- 「**トランザクション応答時間**」データ・セットで表を生成するには、Tab キーを (約 2 回) 押してフォーカスを 「**新規**」ツールに移動し、Enter キーを押します。選択パネルが表示されます。「**トランザクション応答時間**」データ・セットを選択し、「**結果のプレビュー**」ボタンを選択します。

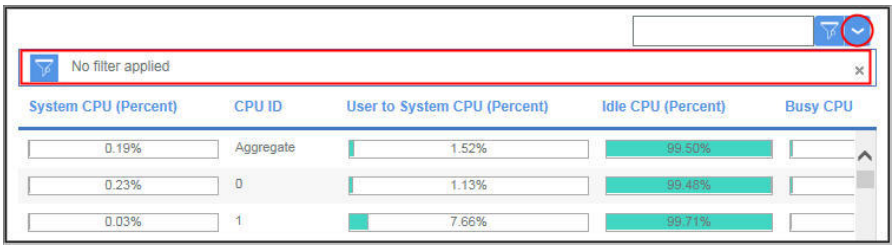
表フィルターの定義

ダッシュボードの「**属性の詳細**」タブに表示される表内の行を制限して、特定のタイプの行、または特定のテキストまたはタイム・スタンプの値を持つ行のみを表示させることができます。フィルター処理にはパーセンテージなどの数値を使用できませんが、一部の数値属性の値は表の表示値に変換され、テキストとして処理されます。クイック・フィルターを適用するか、エディターを開いて拡張フィルターを構成することができます。

手順

テキストまたはタイム・スタンプの属性値によってカスタム表をフィルターに掛けるには、以下の手順を実行します。フィルター処理にはパーセンテージなどの数値を使用できませんが、一部の数値属性の値は表の表示値に変換され、テキストとして処理されます。

1. 「**パフォーマンス**」メニューから、「Application Performance ダッシュボード」を開いた後、管理対象リソースのインスタンスにドリルダウンします。
2. 「**属性の詳細**」タブをクリックします。
   最後に保存されたページが表示されます。ページが保存されていない場合は、「**データ・セット**」および「**属性**」の選択リストが表示されます。
3. 保存されている表のページが表示された場合は、続いてステップ 1097 ページの『5』 に進むか、ドロップダウン・メニュー から別の保存済み表ページを選択するか、 「**追加**」をクリックして新しい表を作成します。
4. 新しい表を作成する場合、または保存されている表を編集する場合は、使用する「**データ・セット**」および「**属性**」を選択して「**結果のプレビュー**」をクリックします。
5. クイック・フィルターの場合、 「**フィルター**」テキスト・ボックスの中をクリックし、フィルタリングの条件とするテキストの一部または全体を入力します。
   入力していくと、入力した文字が含まれていないすべての行が表から削除されます。クイック・フィルターを解除するには、値を削除するか「x」をクリックします。
6. 拡張フィルターの場合は、 ドロップダウン・メニューをクリックしてから「**フィルターの作成**」を選択するか、フィルター・バー内の任意の場所をクリックします。
   「**フィルターの作成**」ウィンドウが開き、 定義されたルールが表示されます。



7. ルールを定義するには、以下のフィールドに入力します。
   a) 列設定を「任意の列」のままにするか、フィルタリングの条件とする属性をリストから選択します。
   b) 条件を「含む」のままにするか、リストから別の演算子 (下記) を選択し、フィルタリングの条件とするテキストまたはタイム・スタンプの値をテキスト・ボックスに入力します。

条件	表に行が含まれる条件...
含む	セル内のいずれかの部分でフィルター値が検出される。
等しい	文字の大/小文字を含め、セルの値に正確に一致する。
接頭部で開始	セル値がフィルター値と同じ文字で始まる。
接尾部で終了	セル値がフィルター値と同じ文字で終わる。
等しくない	セル値がフィルター値と完全には一致しない。
含まない	セル値にフィルター値と同じテキストまたは数値が含まれない。
接頭部で開始しない	セル値がフィルター値と同じ文字で始まらない。
接尾部で終了しない	セル値がフィルター値と同じ文字で終わらない。
空である	セルにデータが存在しない。

   c) ルールの作成が完了したら、「**フィルター**」をクリックして結果を表示するか、 「**フィルター・ルールの追加**」をクリックして別のルールを追加するか、次のステップに進みます。
8. フィルターに複数のルールを使用する場合は、以下のいずれかのステップを実行します。

- 「**一致**」は、初期設定では「**すべてのルール**」に設定されています。これは、行のデータがフィルターのすべてのルールに適合している場合のみ、その行が表示されることを意味します。テキストまたはタイム・スタンプの値が適合しないルールが 1 つでもある場合、その行は除外されます。 ルールが複数あり、行がいずれかのルールに適合していればその行を含めるようにする場合は、設定を「**任意ルール**」に変更します。

- ルールを編集するには、任意のフィールド値を変更します。
- ルールを削除するには、そのルールを選択して 「**ルールの削除**」をクリックします。

9. ルールの定義が完了したら、「**フィルター**」をクリックしてダイアログ・ボックスを閉じ、フィルターを適用します。

タスクの結果

フィルターの基準を満たさないグループがモニターから削除され、フィルター・バーに、項目の数 (「480/1200 項目を表示」など) が報告されます。

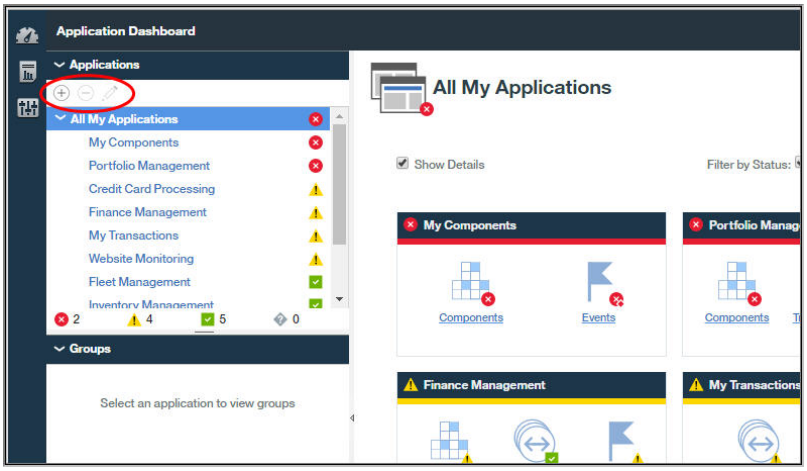
次のタスク

- フィルター・バーにマウス・ポインターを合わせると、 ポップアップ・ウィンドウが開き、フィルタリング条件が表示されます。ルールの削除 (をクリック) や、ウィンドウ内をクリックすることによるフィルターの基準の編集を実行できます。
- フィルター・バーの「**フィルターのクリア**」、または「**フィルターの作成**」ウィンドウの 「**クリア**」をクリックして、フィルターを削除し、すべての行を表示します。

アプリケーションの管理

管理対象リソースをアプリケーションとして編成するには、Application Performance ダッシュボードで使用可能なツールを使用します。

ナビゲーターの「**アプリケーション**」ツールで、アプリケーションを作成または編集し、使用可能な管理対象リソースを適用するためのアプリケーション・エディターが開きます。



「**マイ・コンポーネント**」アプリケーションは、Cloud APM サーバーによって検出された管理対象システムを含む事前定義アプリケーションです。「**マイ・コンポーネント**」は編集することも削除することもできません。

アプリケーションの追加に関するビデオ・デモンストレーションを見るには、Application Performance Management - アプリケーションの定義を参照してください。

IBM Java アプリケーション・スタックをモニターするためのアプリケーションの作成に関するシナリオについては、92 ページの『Application Performance ダッシュボードへの Web アプリケーションの追加』および 93 ページの『IBM Java アプリケーション・スタックと Web アプリケーションとの関連付け』を参照してください。

制約事項 : 「アプリケーションの追加」ツールを使用するには、アプリケーションに対する変更権限が必要です。「削除」ツールおよび「編集」ツールを使用するには、アプリケーションまたは特定のアプリケーションに対する変更権限が必要です。詳しくは、1011 ページの『役割、ユーザー、および権限の操作』を参照してください。

アプリケーションの追加

アプリケーション・エディターを使用して、新しいアプリケーションを作成し、使用可能な管理対象リソースを適用するか、検出されたアプリケーションからいずれかのアプリケーションを 選択します。

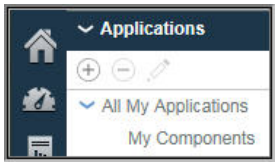
始める前に

「アプリケーションの追加」ツールを使用するには、アプリケーションに対する変更権限が必要です。詳しくは、1011 ページの『役割、ユーザー、および権限の操作』を参照してください。

手順

Application Performance ダッシュボードにアプリケーションを追加するには、Cloud APM コンソールで以下のステップを実行します。

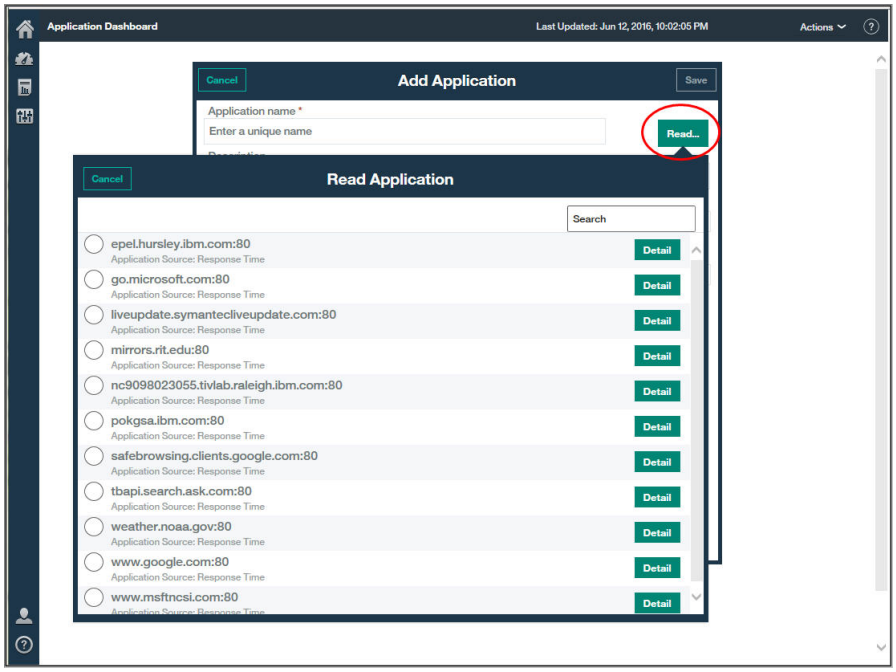
1. Application Performance ダッシュボードが表示されていない場合は、「**パフォーマンス**」メニューからこのダッシュボードを選択します。あるいは、別のコンソール・ページで作業している場合は、「**ホーム**」リンクをクリックします。
2. ナビゲーターの「**アプリケーション**」セクションで、⊕ をクリックします。「**アプリケーションの追加**」ウィンドウが表示されます。



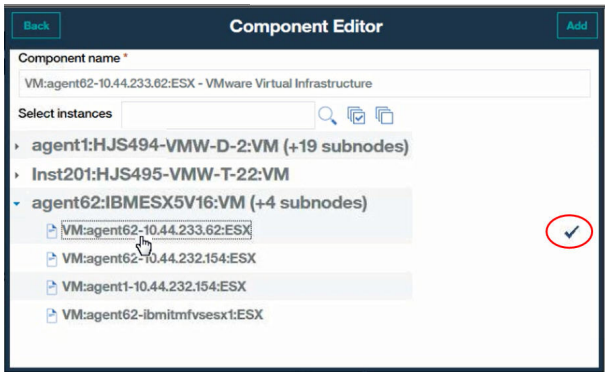
3. アプリケーションの名前を「**アプリケーション名**」フィールドに入力し、必要に応じて、説明を「**説明**」フィールドに入力します。

   名前と説明には、記号 ! " % & ' * ? < > } { ¥ を使用しないでください。

   ガイド付きデモで、アプリケーション名の例 (「Finance Management」や「Credit Card Processing」など) をいくつか確認できます。
4. 「**読み取り**」をクリックすると、「**アプリケーションの読み取り**」ウィンドウが開きます。このウィンドウには、検出されたアプリケーションのリストが表示されます。ここで、以下に示す 1 つ以上のステップを実行します。
   - アプリケーションのコンポーネントを表示するには、「**詳細**」をクリックします。
   - 使用するアプリケーションを選択して、「**保存**」をクリックします。「**アプリケーションの読み取り**」ウィンドウが終了し、「**アプリケーションの読み取り元**」フィールドにソース・リポジトリーが表示され、「**アプリケーション・コンポーネント**」にコンポーネントがリストされます。
   - 何も選択せずにウィンドウを閉じるには、「**キャンセル**」をクリックします。



5. 「**テンプレート**」フィールドで、**カスタム・アプリケーション**・テンプレートをそのまま使用するか、または ボタンを使用して別のテンプレートを選択してから「**保存**」をクリックします。
   関連するすべてのコンポーネントのタイプとインスタンスが、「**アプリケーション・コンポーネント**」リストに表示されます。
6. 「**コンポーネントの追加**」をクリックして「**コンポーネントの選択**」ウィンドウを開き、リストからコンポーネントを選択します。
   「**コンポーネント・エディター**」が表示されます。
7. アプリケーションのエージェント・インスタンスまたはサブノード・インスタンス (あるいはその両方) を検索および選択するには、以下に示す 1 つ以上のステップを実行します。
   - インスタンスをクリックして選択します。
   - サブノードを持つエージェント・ノードについては、ツリーが省略されているときに名前をクリックしてそのノードのみを選択するか、ノード・ツリーを展開 (をクリック) してノードをクリックすることによりノードおよびすべてのサブノードを選択するか、ノード・ツリーを展開してインスタンスをクリックすることにより個々のサブノードを選択します。
   - ツールバーを使用して、検索テキスト・ボックスのテキストが含まれているインスタンスを検索し、すべてのインスタンスを選択またはクリアします。
   - ナビゲーター内の表示名を変更するには、コンポーネント名を編集します。



V6-V7 Tivoli Monitoring エージェント・インスタンスを追加しても、使用可能なインスタンスのリスト内にそれが表示されない場合、ハイブリッド・ゲートウェイに関連付けられている Tivoli Enterprise Portal Server がサポートされるバージョンであることを確認してください (Hybrid Gateway supported agents (APM Developer Center) を参照)。

8. 「**追加**」をクリックして、選択したエージェント・ノードおよびサブノードをアプリケーションに 追加し、「**戻る**」をクリックします。

   「アプリケーション・コンポーネント」リストが、新しいコンポーネント名で更新されます。

9. インスタンスの追加先となる別のコンポーネントを選択し、1100 ページの『6』、1100 ページの『7』、1101 ページの『8』 を繰り返すか、「**閉じる**」をクリックします。
10. 「**アプリケーション・コンポーネント**」リスト内のコンポーネントに他のインスタンスが関連付けられている場合は、関連インスタンスの数を示すボタンが表示されます。この場合、以下の操作を実行することができます。

    a) 関連するインスタンスを「**更新された詳細**」ウィンドウに表示するには、1 ボタンをクリックします。
    更新のタイプごとにバーが表示されます。これにはインスタンスの名前が含まれます。例えば、いずれかのコンポーネントが削除された場合、「**削除済みコンポーネント**」バーの下にそのコンポーネントが表示されます。

    b) アプリケーションのリソース・リストを更新するには、1 つ以上のインスタンスを選択して「**保存**」をクリックします。

11. アプリケーションの定義が終了したら、「**保存**」をクリックしてアプリケーション・エディターを閉じます。この操作により、変更内容が保存されます。「**キャンセル**」をクリックすると、変更内容が取り消されます。

タスクの結果

アプリケーションの更新は、変更内容を保存した後に Cloud APM サーバーによって処理が完了します。変更内容がダッシュボードに表示されるまで数分かかる場合があります。(変更内容の表示に長い時間がかかる場合は、ブラウザー・キャッシュをクリアしてみてください。) 新しいアプリケーションが、Application Performance ダッシュボードとナビゲーターの「**アプリケーション**」セクションに表示されます。このアプリケーションを選択すると、コンポーネントが「**グループ**」セクションに表示されます。

アプリケーションの編集

アプリケーション・エディターを使用して、定義済みアプリケーションを変更し、アプリケーションのコンポーネントとして管理対象リソースを追加または削除します。

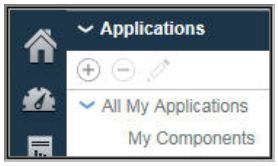
始める前に

「編集」ツールを使用するには、アプリケーションまたは特定のアプリケーションに対する変更権限が必要です。詳しくは、を参照してください。1003 ページの『役割と権限』.

手順

アプリケーションを編集するには、Cloud APM コンソールで以下のステップを実行します。

1. Application Performance ダッシュボードが表示されていない場合は、「**パフォーマンス**」メニューからこのダッシュボードを選択します。あるいは、別のコンソール・ページで作業している場合は、「**ホーム**」リンクをクリックします。
2. ナビゲーターの「**すべてのマイ・アプリケーション**」リストから編集対象のアプリケーションを選択し、をクリックします。



「**アプリケーションの編集**」ウィンドウが表示されます。

3. オプション:「**アプリケーション名**」または「**説明**」を編集します。
   名前と説明には、記号 ! " % & ' * ? < > } { ¥ を使用しないでください。
   アプリケーションの権限の表示や変更の対象が、すべてのアプリケーションではなく、個別のアプリケーションの場合、アプリケーションの名前を変更した後に、アプリケーションをダッシュボードに表示できなくなったり、アプリケーションを変更できなくなったりする場合があります。この制限は、名前変更されたアプリケーションが新規アプリケーションとして処理されるためです。役割管理者またはモニター管理者が、名前変更されたアプリケーションに対する表示または変更の権限をユーザーに付与する必要があります。
4. コンポーネントとインスタンスをアプリケーションに 追加するには、以下の手順を実行します。
   a) をクリックし、開いたウィンドウ内のリストからコンポーネントを選択します。
   「**コンポーネント・エディター**」が表示されます。
   b) 以下のようにして、アプリケーションのエージェント・ノード・インスタンスまたはサブノード・インスタンス (あるいはその両方) を選択します。
   - インスタンスをクリックして選択します。
   - サブノードを持つノードについては、ツリーが省略されているときに名前をクリックしてそのノードを選択するか、ノード・ツリーを展開 (をクリック) してノードをクリックすることによりノードおよびすべてのサブノードを選択するか、ノード・ツリーを展開してインスタンスをクリックすることにより個々のサブノードを選択します。
   - ツールバーを使用して、検索テキスト・ボックスのテキストが含まれているインスタンスを検索し、すべてのインスタンスを選択またはクリアします。
   - ナビゲーター内の表示名を変更するには、コンポーネント名を編集します。

   c)「**追加**」をクリックして 1 つ以上のインスタンスを追加し、「**戻る**」をクリックします。
   「アプリケーション・コンポーネント」リストが、新しいコンポーネント名で更新されます。
   d) インスタンスを追加する別のコンポーネントを選択することも、「**閉じる**」をクリックすることもできます。

「アプリケーション・コンポーネント」リストが、新しいコンポーネント名で更新されます。名前の後に表示される括弧内の数字は、そのコンポーネントに関連付けられているインスタンスの数を示しています。

5. コンポーネント名の編集、またはコンポーネントに関連付けられているインスタンスの編集を行うには、対象のコンポーネントを「**アプリケーション・コンポーネント**」リストから選択して をクリックします。
   a) 別のインスタンスをコンポーネントに関連付けるには、関連付けたいインスタンスを検索して選択します。
   b) このアプリケーションの表示名としてナビゲーター内で使用されているコンポーネント名を変更するには、「**コンポーネント名**」フィールドを編集します。
   c)「**保存**」をクリックします。

   「アプリケーション・コンポーネント」リストが、画面上で行った変更内容で更新されます。
6. アプリケーションからコンポーネントまたはインスタンスを 削除するには、選択してから をクリックします。「**OK**」をクリックすると、削除操作が確定します。
7. 「**アプリケーション・コンポーネント**」リスト内のコンポーネントに他のインスタンスが関連付けられている場合は、関連インスタンスの数を示すボタンが表示されます。この場合、以下の操作を実行することができます。
   a) 関連するインスタンスを「**更新された詳細**」ウィンドウに表示するには、 ボタンをクリックします。
   更新のタイプごとにバーが表示されます。これにはインスタンスの名前が含まれます。例えば、いずれかのコンポーネントが削除された場合、「**削除済みコンポーネント**」バーの下にそのコンポーネントが表示されます。
   b) アプリケーションのリソース・リストを更新するには、1 つ以上のインスタンスを選択して「**保存**」をクリックします。
8. アプリケーションの編集が終了したら、「**保存**」をクリックしてアプリケーション・エディターを閉じます。この操作により、変更内容が保存されます。「**キャンセル**」をクリックすると、変更内容が取り消されます。

タスクの結果

アプリケーションの更新は、変更内容を保存した後に Cloud APM サーバーによって処理が完了します。変更内容がダッシュボードに表示されるまで数分かかる場合があります。

関連資料

1003 ページの『役割と権限』

アプリケーションの削除

Application Performance ダッシュボードに表示するように定義したアプリケーションが不要になった場合は、削除できます。アプリケーションを削除しても、サポート・コンポーネントがアンインストールされることはありません。それらが含まれているアプリケーションのみが削除されます。同じコンポーネントを、他のアプリケーションに追加するために使用できます。また、同じコンポーネントは、そのコンポーネントが属する他のアプリケーションから削除されません。

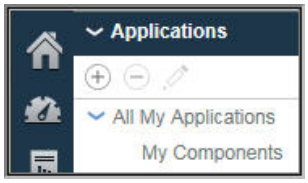
始める前に

「削除」ツールを使用するには、アプリケーションまたは特定のアプリケーションに対する変更権限が必要です。詳しくは、1011 ページの『役割、ユーザー、および権限の操作』を参照してください。

手順

Application Performance ダッシュボードでアプリケーションを削除するには、以下の手順を実行します。

1. Application Performance ダッシュボードが表示されていない場合は、「**パフォーマンス**」メニューからこのダッシュボードを選択します。あるいは、別のコンソール・ページで作業している場合は、「**ホーム**」リンクをクリックします。



2. ナビゲーターの「**アプリケーション**」セクションで、「**すべてのマイ・アプリケーション**」リストから削除対象のアプリケーションを選択し、⊖をクリックします。
   処理を確認するメッセージが表示されます。
3. アプリケーションの削除を確定する場合は「**はい**」をクリックし、確信が持てない場合は「**いいえ**」をクリックします。

タスクの結果
「**はい**」をクリックすると、アプリケーションが Application Performance ダッシュボードから削除されます。

次のタスク
削除するその他のアプリケーションすべてに対してこの手順を繰り返します。

オフライン・エージェントの表示および削除

エージェントがオフラインで 4 日間経過すると、そのエージェントは Cloud APM コンソールから削除されます。オフライン・エージェントがどのように表示されるかを確認すると共に、リソース・グループ、トポロジー・ビュー、およびその他の機能に対する影響を確認してください。4 日間経過する前にダッシュボードから管理対象システムを削除するには、アプリケーション・エディターを使用します。

このタスクについて
管理対象のシステムにエージェントがインストールされると、そのエージェントは Cloud APM サーバーに接続し、表示としきい値の評価のためにデータ・サンプルを Application Performance ダッシュボードに送信します。エージェントがオフラインの場合、ナビゲーターおよびダッシュボードに 状況標識が表示されます。サーバーは、エージェントから応答がない特定の間隔の数が経過するのを待機してから、エージェントが使用不可であることを示します。1105 ページの『オフライン・エージェントの例』を参照してください。

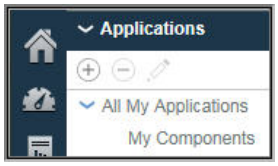
オフライン・エージェントは、次の例外を除き、4 日間経過するとユーザー・インターフェースから 削除されます。エージェントがトランザクション・トラッキングをサポートするエージェントである 場合、そのオフライン・エージェントは引き続き「集約トランザクション・トポロジー」ビューと「トランザクション・インスタンス・トポロジー」ビューに表示されます。

任意の定義済みアプリケーションからオフライン・エージェントを削除できます。それにより、4 日間の待機期間が経過する前に Cloud APM コンソールからオフライン・エージェントが削除されます。

手順

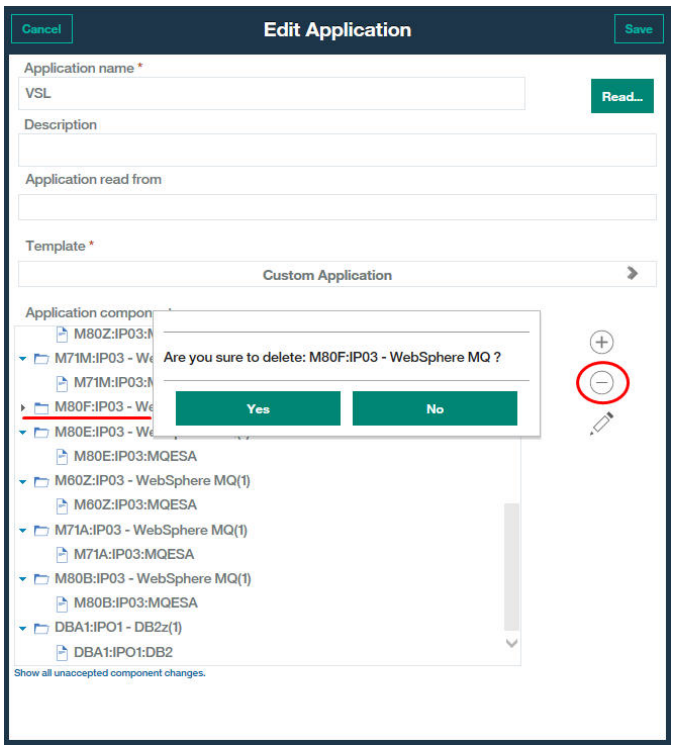
定義済みアプリケーションからオフライン・エージェントを削除するには、以下のステップに従ってください。

1. Application Performance ダッシュボードが表示されていない場合は、「**パフォーマンス**」メニューからこのダッシュボードを選択します。あるいは、別のコンソール・ページで作業している場合は、「**ホーム**」リンクをクリックします。
2. ナビゲーターの「**アプリケーション**」セクションで、オフライン・エージェントがコンポーネントであるアプリケーションを選択し、「**アプリケーションの編集**」をクリックします。



3. 「アプリケーション・コンポーネント」リストのエージェントまたはエージェント・サブノードを選択し、⊖ をクリックします。

   サブノードを持つエージェントについては、ツリーが省略されているときに名前をクリックしてそのエージェントのみを選択するか、ノード・ツリーを展開 (▸ をクリック) してノードをクリックすることによりノードおよびすべてのサブノードを選択するか、ノード・ツリーを展開してインスタンスをクリックすることにより個々のサブノードを選択します。



4. オフライン・エージェントまたはサブノードを削除するためのアプリケーションの編集が完了したら、「**保存**」をクリックします。

タスクの結果

アプリケーションの更新は、変更内容を保存した後に Cloud APM サーバーによって処理が完了します。オフライン・エージェントが Application Performance ダッシュボードから削除されるまで数分かかる場合があります。

オフライン・エージェントの例

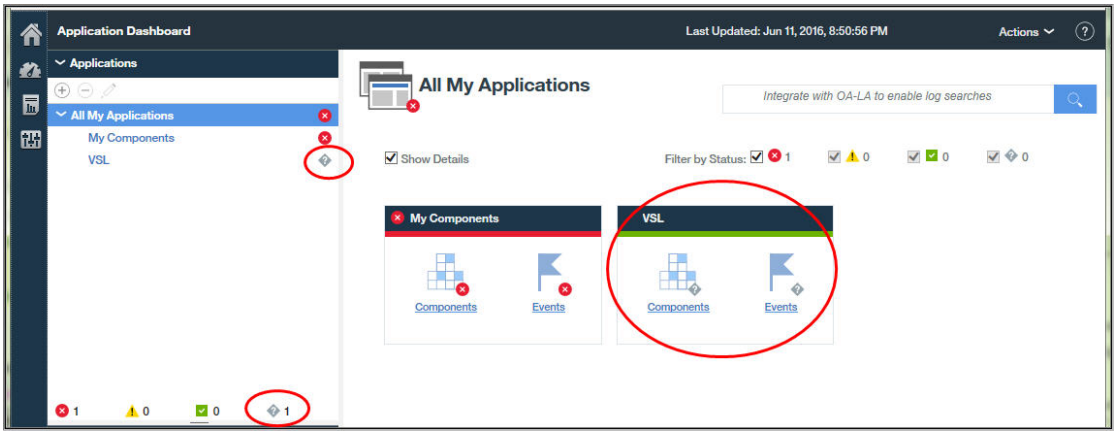
オフライン・エージェントが Cloud APM コンソールにどのように表示されるのかを示す例を確認してください。モニターする必要がなくなったオフライン・エージェントを表示しないようにすることができます。エージェントが後でオンラインに戻ると、モニタリングが再開します。

エージェントがオフラインのときは、Cloud APM コンソールに何のデータも送信されません。また、Application Performance ダッシュボードには、エージェントおよびエージェントが属するアプリケーションに対して ◆ 状況標識が表示されます。そのエージェントは、アプリケーション・エディターで定義済みアプリケーションに追加するために使用することも、リソース・グループ・マネージャーでカスタム・グループに追加するために使用することも、「属性の詳細」タブで表と履歴折れ線グラフを作成するために使用することもできません。

Application Performance ダッシュボード - すべてのマイ・アプリケーション

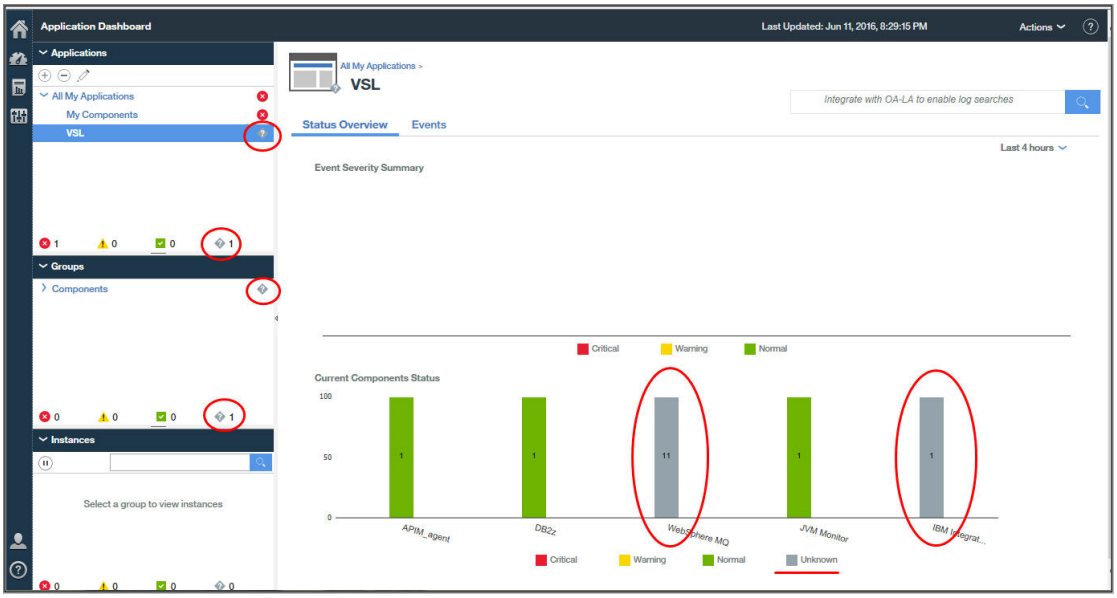
ホーム・ダッシュボード・ページ「**すべてのマイ・アプリケーション**」には、オフライン状況の最初の標識が表示されます。ナビゲーターの「アプリケーション」セクションのカウンターには、使用不可リソースがあるアプリケーションの 数が表示されます。

要約ボックスには、アプリケーションのいずれの管理対象リソースに対してもイベントがオープンされていないため「正常」のイベント状況が表示されます。



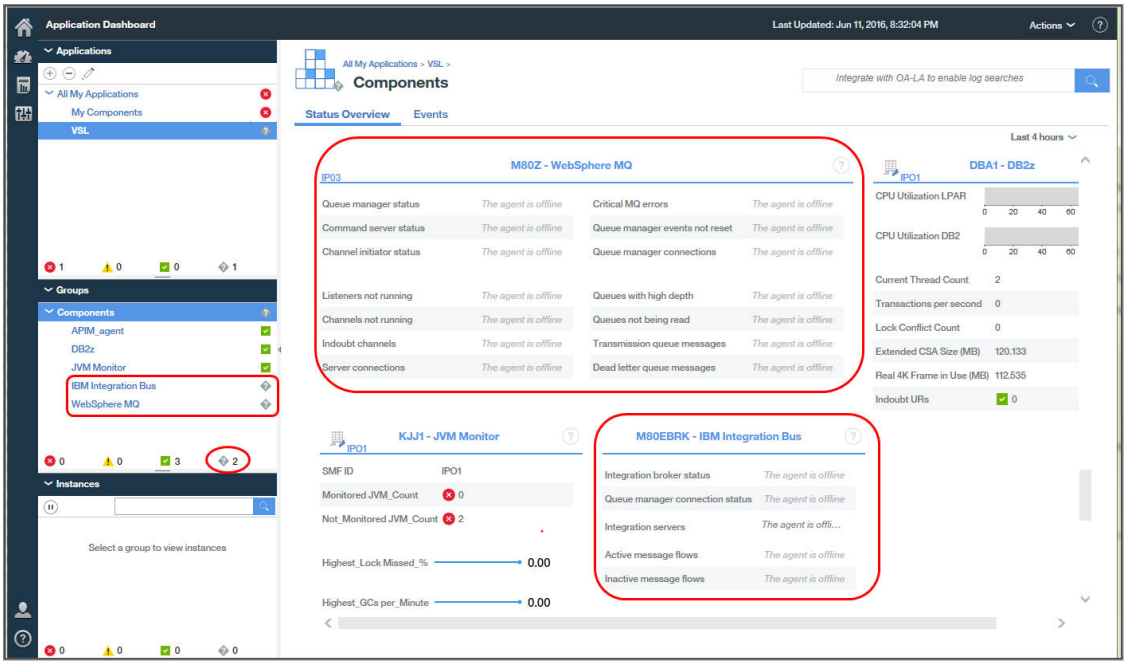
Application Performance ダッシュボード - アプリケーション

ユーザーが要約ボックスのタイトル・バーをクリックするかナビゲーターからアプリケーションを選択すると、「**状況の概要**」タブが表示され、空の「**イベント重大度要約**」グラフが示されます。「**現在のコンポーネント状況**」棒グラフには、オフライン・エージェントの状況が「不明」と表示されます。



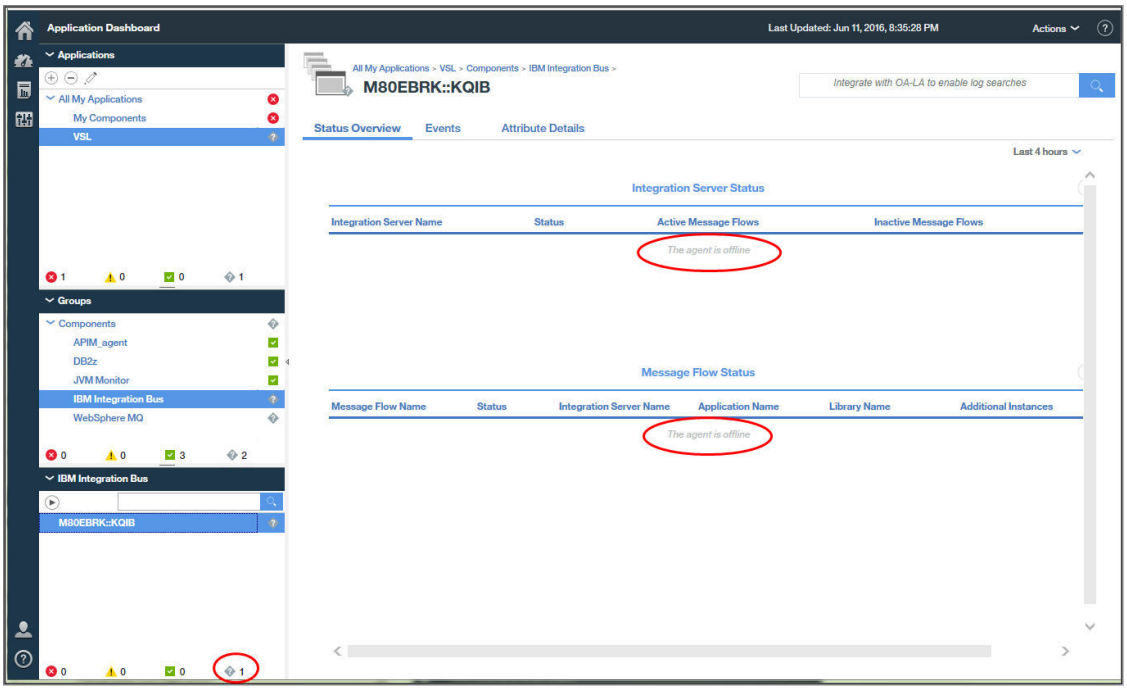
Application Performance ダッシュボード - グループ

ユーザーが「**現在のコンポーネント状況**」グラフまたはナビゲーターの「**コンポーネント**」グループをクリックすると、「**状況の概要**」タブの表示内容が変わり、各管理対象リソースの要約グループ・ウィジェットが表示されます。使用不可のエージェントに関するグループ・ウィジェットには、KPI ではなく、エージェントがオフラインであることを示すメッセージが表示されます。

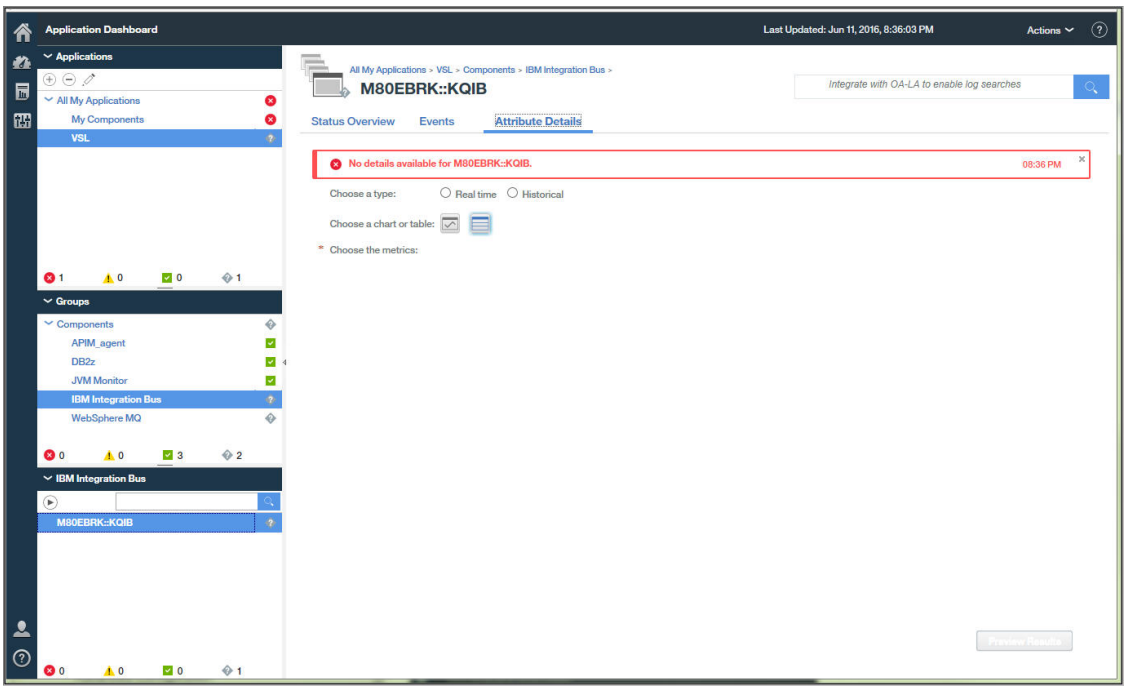


Application Performance ダッシュボード - インスタンス

オフラインのエージェントの要約グループ・ウィジェットのいずれかをユーザーがクリックすると、「**状況の概要**」タブに、選択したエージェントのグラフと表のウィジェットが表示されます。一方で、要約グループ・ウィジェットについては、エージェント・インスタンスの KPI ではなく、エージェントがオフラインであることを示すメッセージのみが表示されます。



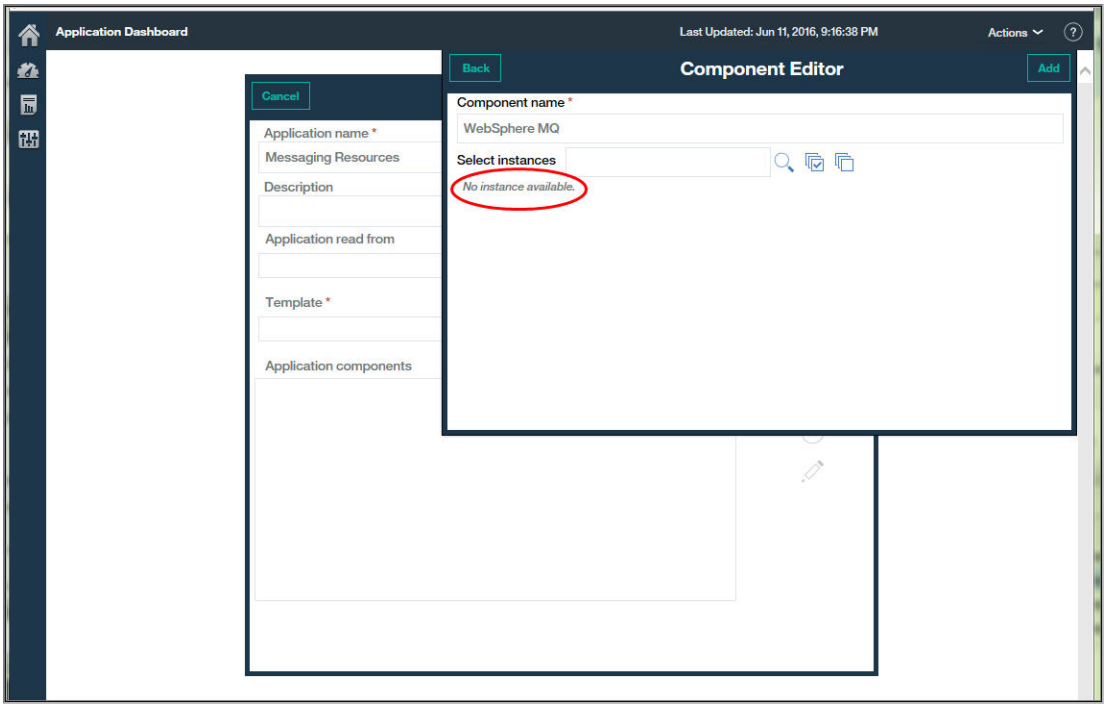
ユーザーが「**属性の詳細**」タブをクリックすると、エージェント・インスタンスに関する詳細が使用不可であることを示すメッセージが表示されます。オフラインのエージェント・インスタンスに対してカスタムのグラフや表を作成することはできません。



アプリケーション・エディター

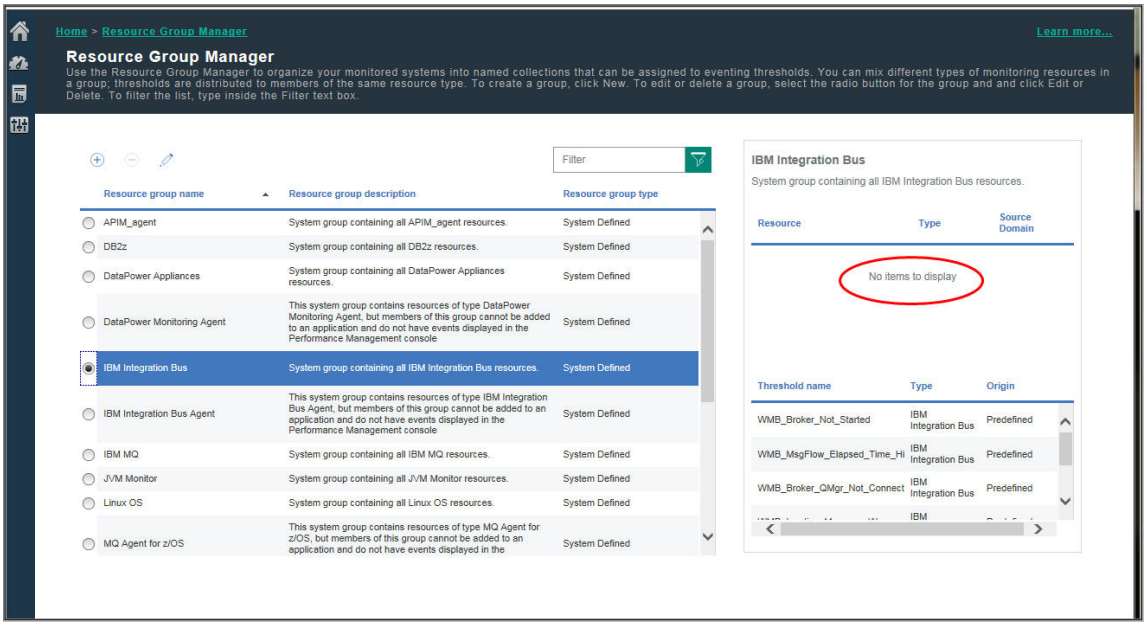
Application Performance ダッシュボードでユーザーがナビゲーターの「アプリケーション」の ⊕ 「**アプリケーションの追加**」ツールまたは ✎ 「**アプリケーションの編集**」ツールをクリックすると、アプリケーション・エディター・ウィンドウがポップアップ表示されます。

ユーザーが ⊕ 「**コンポーネントの追加**」をクリックしてエージェント・タイプを選択したが、そのタイプのエージェントがインストールされていなかったりオフラインであったりした 場合は、エージェント・インスタンスが使用不可であることを示すメッセージが表示されます。他のエージェント・インスタンスが使用可能である場合は、それらがリストに表示されます。



リソース・グループ・マネージャー

ユーザーが 🎚 「**システム構成**」 > 「**リソース・グループ・マネージャー**」を選択すると、リソース・グループの表が含まれたページが開きます。グループを選択すると、その構成要素エージェント・インスタンスが、割り当てられたしきい値と共にリストされます。すべてのエージェント・インスタンスがオフラインである場合は、インスタンスが使用不可であることを示すメッセージが表示されます。



関連概念

1098 ページの『アプリケーションの管理』
管理対象リソースをアプリケーションとして編成するには、Application Performance ダッシュボードで使用可能なツールを使用します。

1079 ページの『ダッシュボードの使用』

関連資料

982 ページの『リソース・グループ・マネージャー』
モニター対象環境には、その目的により分類できる複数の管理対象システムが存在する場合があります。そのようなシステムには通常、同じしきい値要件が存在します。「**リソース・グループ・マネージャー**」を使用して管理対象システムをグループに編成し、しきい値を割り当てることができます。役割ベースのアクセス制御 (RBAC) ポリシーに関連するリソース・グループを作成することもできます。

イベント状況

選択したナビゲーター項目のオープン・イベントの概要を要約して表示し、詳細ダッシュボードにドリルダウンして、状況がクリティカルまたは警告であるイベントに対応するには、「**イベント状況**」を使用します。

状況標識は、管理対象システムで実行されているしきい値からのイベントに関するものです。ハイブリッド・ゲートウェイが構成されている場合、IBM Tivoli Monitoring 環境内の管理対象システムで実行されているシチュエーションからもイベントが発生する可能性があります。ご使用の構成に IBM Operations Analytics - Predictive Insights が含まれている場合、検出された異常も表示されます。

一部のしきい値に関するイベントは、Application Performance ダッシュボードに表示されません。しきい値は、パブリッシュされていないリソースの属性を使用します。この状況は、サブノードをサポートするエージェントで起こることがあります。(サブノードの説明については、Agent Builder のトピック を参照してください。)

クリティカル、警告、正常

- 状況標識は、以下のようにしきい値を整理してイベントの重大度をまとめたものです。

  クリティカル状況は、各イベントの重大度が致命的またはクリティカルであることを示します。
  警告状況は、各イベントの重大度が軽微または警告であることを示します。
  正常状況は、各イベントの重大度が不明であることを示します。

不明状況は、管理対象システムがオフラインであることを示します。4 日間オフラインになると、管理対象システムはすべてのアプリケーションから削除され、ダッシュボードに表示されなくなります。
状況の確認方法、エージェントの停止または開始方法については、181 ページの『エージェント・コマンドの使用』を参照してください。
の**「オフライン・システム除去の遅延」**オプションを参照してください。

- V6-V7 1 つ以上のハイブリッド・ゲートウェイが構成されている場合、Tivoli Monitoring シチュエーションからのイベントの状況標識は、しきい値と同じになりますが、「正常」状況が、「安全」、「通知」、または「不明」の重大度のイベントを示すという点を除きます。
- ご使用の管理対象環境に IBM Operations Analytics - Predictive Insights が含まれている場合、検出された異常は、状況標識に重なるひし形アイコン (など) によって示されます。詳しくは、1111 ページの『Operations Analytics - Predictive Insights を使用した異常の調査』を参照してください。

「イベント重大度要約」パーセント・ゲージ

- 「イベント重大度要約」ゲージには、クリティカル、警告、および正常の各イベント状況の比率が表示されます。例えば、

は、イベントのうち、50% が重大度「軽微」または「警告」のしきい値にあり、50% が重大度「致命的」または「クリティカル」のしきい値にあることを示します。
- また、イベントの総数と、状況レベルごとの数も報告されます。
- イベント数には、Operations Analytics - Predictive Insights からの異常が含まれます。例えば、「8 (1 個の異常を含む)」という合計は、7 個のしきい値イベントと 1 個の異常イベントがあることを示します。

「イベント」表

- オープン・イベントおよび状況の表は、選択したナビゲーター項目 (アプリケーション、グループ、サブグループ、またはインスタンス) によって定義されます。
- イベントは「**重大度**」列でソートされ、重大度が最高のものが先頭に表示されます。ソート順を変更するには、列見出しをクリックします。
- それぞれの行には、イベントに関する以下の情報が表示されます。

  しきい値名
  しきい値に指定した名前。

  V6-V7 シチュエーションに指定した名前。

  状況
  イベントの状況 (**オープン**など)。

  重大度
  イベントの重大度値: 「クリティカル」(「致命的」および「クリティカル」しきい値の重大度にあてはまります)、「警告」(「軽微」および「警告」しきい値の重大度にあてはまります)、または「正常」。(「不明」しきい値重大度に適用されます。Tivoli Monitoring イベントの場合、「安全」、「情報」、および「不明」の各重大度に適用されます)。

  不明状況は、管理対象システムがオフラインであることを示します。4 日間オフラインになると、管理対象システムはすべてのアプリケーションから削除され、ダッシュボードに表示されなくなります。(状況の確認、エージェントの停止、およびエージェントの開始を行うには、181 ページの『エージェント・コマンドの使用』を参照してください)。

  ご使用の管理対象環境に IBM Operations Analytics - Predictive Insights が含まれている場合、ヒストリカル・データに適用される分析によって異常が検出され、イベントがオープンされることがあります。検出された異常に対してオープンされたイベントは、状況標識に重なるアイコン (など) によって示されます。「**異常分析の表示**」リンクをクリックすると、Predictive Insights の「**サービス診断**」ビューがブラウザーの新しいタブまたはウィンドウに開かれます。「**サービス診断**」ビューを使用して、アプリケーションをサポートするコンポーネントでの 異常動作を確認します。

表示項目
複数行データ・セットにのみ適用されます。表示項目は、同一の管理対象システムについてオープンされた複数のイベントを識別するためのしきい値として選択されたキー属性です。

ソース
イベントのソースを識別する、モニター・エージェントから派生したシステム・ホスト名またはその他の名前。

タイム・スタンプ
イベントが発生した日時、または、イベント取得元のエージェントによって状態が観察された日時。Cloud APM コンソールのユーザーのタイム・ゾーンで表します。

エージェントが再始動された場合またはエージェントに対するしきい値定義が変更された場合、しきい値条件が引き続き true であると、エージェントのサンプル・イベントがクローズされ、再オープンされます。このようなシナリオでは、**タイム・スタンプ**値が、イベント取得元のエージェントがイベントを再オープンした時刻に更新されます。

ピュア・イベントの場合、イベント取得元のエージェントがしきい値条件を true であると判別するたびに、新しいイベントがオープンされ、以前のイベント・インスタンスに取って代わります。ピュア・イベントは、しきい値条件が true であると評価された最終時刻から 24 時間 (または構成可能な時間数) オープンのままになります。Cloud APM コンソールに表示されるピュア・イベントは、最新のインスタンスのみです。

説明
しきい値のために書かれた説明 (存在する場合)。

- 行をクリックして、イベントに関する詳細を展開します。

ノード
ノード・インスタンスの管理対象システム名。

サブノードを持つエージェントの場合は、サブノードを表示するかどうかを**「サブノード・イベントの有効化」**オプションで制御します。詳しくは、1074 ページの『UI 統合』を参照してください。

しきい値 ID
しきい値の ID。

グローバル・タイム・スタンプ
Cloud APM サーバーがイベント取得元のエージェントからイベントを受信した日時。Cloud APM コンソールのユーザーのタイム・ゾーンで表します。

タイプ
イベントがピュアまたはサンプルのいずれであるか。ピュア・イベントは非送信請求の通知です。ピュア・イベントのしきい値には、現行値をモニターできるサンプリング間隔や定数メトリックがありません。

説明
しきい値のために書かれた説明 (存在する場合)。

式
しきい値エディターに書き込まれる式。例: `Percent Failed > 10.000 AND Transaction Definition Name != 'Ignore_Resources'`

msg 基本スロットの値をカスタマイズするために EIF スロット・カスタマイズ機能を使用した場合は、しきい値式の代わりに、カスタマイズした **msg** スロット値が表示されます。詳しくは、『しきい値マネージャー』トピックおよび 992 ページの『EIF 受信側に転送するためのイベントのカスタマイズ』の 988 ページの『EIF イベントを転送しますか?』を参照してください。

他の行を選択して展開したり、再度クリックして行を省略したりすることができます。行が展開されている間は、イベントの原因を判別するのに役立つ管理対象システムのダッシュボードにドリルダウンできます。

Operations Analytics - Predictive Insights を使用した異常の調査

IBM Cloud Application Performance Management のみ: 管理対象環境に IBM Operations Analytics - Predictive Insights が含まれている場合、ヒストリカル・データに適用した分析によって異常を検出し、イ

ベントを開くことができます。 Application Performance ダッシュボード ダッシュボードを使用して、Operations Analytics - Predictive Insights が検出した異常を見つけて表示します。

始める前に

Application Performance ダッシュボードで異常についてのアラートを受け取るには、Operations Analytics - Predictive Insights を Cloud APM 環境と統合する必要があります。詳しくは、971 ページの『Operations Analytics - Predictive Insights との統合』を参照してください。

このタスクについて

Application Performance ダッシュボードは、ドメイン内のアプリケーション、およびそのコンポーネントである管理対象システムの状況要約を表示します。「**すべてのマイ・アプリケーション**」ダッシュボードの要約ボックス内のイベント状況標識は、「クリティカル」、「警告」、および「不明」の重大度をそれぞれ示します。Operations Analytics - Predictive Insights によって検出された異常がイベントに含まれている場合、状況標識に異常アイコン (、、または) が含まれます。 クリティカルおよび警告の異常の場合は、アプリケーション、グループ、およびインスタンス・ダッシュボード・ページにドリルダウンすると同じ標識が「**イベント**」タブのタイトルの横に表示されます。Events 2 実際のデモンストレーションでは、IBM Cloud Application Performance Management のガイド付きデモを開始し、タスク・リストをスクロールダウンして「*Predictive Insights* の異常の識別および診断」を選択します。

手順

以下のステップを実行して、異常を識別し、Operations Analytics - Predictive Insights の「**サービス診断**」ビューに表示します。

1. 「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックして「**すべてのマイ・アプリケーション**」ダッシュボードを開きます。
2. 異常アイコンを示す「**イベント**」状況標識が要約ボックスにある場合は、「**イベント**」リンクをクリックします。

   アプリケーション・ダッシュボードの「**イベント**」タブが開きます。 異常の数を含むイベントの総数が、「**イベント重大度要約**」に表示されます。
3. 「**重大度**」列で、、または によって示されている異常イベントの表の行をクリックします。

   行が展開され、イベントの詳細が表示されます。
4. 「**異常分析の表示**」 をクリックして、新規ブラウザー・タブまたはウィンドウで Operations Analytics - Predictive Insights の「**サービス診断**」ビューを開きます。

次のタスク

- 「**サービス診断**」ビューを使用して、アプリケーションをサポートするコンポーネントでの 異常動作を確認します。 をクリックすると、 「**サービス診断**」ビューのオンライン・ヘルプが開きます。
- アプリケーション・ダッシュボードに戻り、関連する問題を示している可能性がある管理対象システム上の他のイベントを探します。さらに詳しく調べるには、「**状況の概要**」タブをクリックし、イベントが発生した管理対象システム・インスタンスにドリルダウンします。その情報を使用して、Predictive Insights によって特定された問題を回避するために実行する必要があるアクションを判別します。
- 異常が予期されるのに何も表示されない場合は、異常を生成するための Operations Analytics - Predictive Insights の学習時間が不十分な場合があります。標準的な学習時間は 2 週間です。追加の構成が必要な場合もあります。

カスタム・ビュー

IBM Cloud Application Business Insights Universal View を使用して、独自のページをカスタマイズすることで、Application Performance ダッシュボードの事前定義ページが既に提供している値を拡張できます。

Universal View は、リソース・モニター・データを表示するために使用できます。シンセティック・トランザクション・データ、トランザクション・トラッキング・データ、応答時間エージェント・データ、および詳細診断データを表示するためには使用できません。Universal View を使用することで、アプリケーショ

ン用のモニター・ページを素早く作成し、表示用に保存できます。保存済みのカスタム・ダッシュボード・ページの表示中に、ダッシュボードを自動最新表示モードで表示したり、ダッシュボードを生データ・ファイルにエクスポートしたりできるほか、ダッシュボードの編集や削除を行うことができます。

Cloud APM のデフォルトの 4 つの役割である役割管理者、モニター管理者、システム管理者、およびモニター・ユーザーには、ダッシュボード・ページを表示および変更するための異なる権限があります。詳しくは、以下を参照してください。表 1. 役割および権限 。

「**カスタム・ビュー**」タブで使用可能なオプションは、ページを編集するか、または表示するかによって異なります。

カスタム・ページの作成および管理

「カスタム・ビュー」タブを使用して、選択したリソース・メトリックが取り込まれるウィジェットを追加または更新することにより、選択したアプリケーション、グループ、またはインスタンス用のダッシュボード・ページを作成または編集します。

このタスクについて

作成および保存したページは、選択したアプリケーションに関連付けられます。例えば、Cloud APM ガイド付きデモのインベントリー管理アプリケーションには、Linux OS、MySQL、Node.js、Hadoop、および Ruby の各モニター・エージェントがあります。カスタム・ページは、ナビゲーターの任意のレベル (アプリケーションからインスタンスまで) で作成および保存し、作成したのと同じレベルで開くことができます。特定のレベルで作成されたページは、同じレベルでのみ開くことができます。ウィジェットで使用可能なメトリックは、アプリケーション内のリソースのいずれかから取得できます。デモのインベントリー管理の例を使用すると、Ruby エージェントからの表や Linux OS エージェントからのグラフなどを含むページを作成できます。

手順

作成および保存したページは、選択したアプリケーションに関連付けられます。ダッシュボード・ページを作成し、カスタマイズするには、以下の手順を実行します。

1. 「**パフォーマンス**」メニューから、「Application Performance ダッシュボード」を開いた後、アプリケーションを選択します。

   「**カスタム・ビュー**」タブが「**状況の概要**」および「**イベント**」タブの後に表示されます。ナビゲーターのグループ、サブグループ、またはインスタンスの各レベルにドリルダウンすることもできます。
2. 「**カスタム・ビュー**」タブをクリックします。

   このタブには、「**カスタム・ページのテンプレートの選択**」ウィンドウが表示されるか、デフォルト・ページが既に設定されている場合は、そのページが表示されます。

   - 「**カスタム・ページのテンプレートの選択**」ウィンドウが開いたら、ステップ 4 に進みます。
   - デフォルト・ページが表示されたら、ステップ 3 に進みます。
3. 「**追加**」をクリックして新規ページを作成します。
4. 以下のデフォルトのテンプレート ・オプションから、テンプレートをクリックします。
   - 1x1 テンプレート
   - 1x2 テンプレート
   - 2x2 テンプレート
   - 2x3 テンプレート
   - 3x3 テンプレート
   - 3x2 テンプレート
   - 2x1 テンプレート
   - 1x3 テンプレート
   - 3x1 テンプレート

「**戻る**」をクリックすると、お気に入りのマークが付けられたページ、またはリストの最初のページが開きます。ページが存在しない場合は、「**カスタム・ページのテンプレートの選択 (Select a Template for your Custom Page)**」ウィンドウが開きます。

5. テンプレートをカスタマイズします。詳しくは、テンプレートのカスタマイズを参照してください。
6. ウィジェットを作成します。詳しくは、1117 ページの『ウィジェット・プロパティーの定義』を参照してください。
7. 「**ページのデフォルト時間フレームの設定 (Set Default Timeframe for the page)**」をクリックして、ページのデフォルトのデータ保存期間を設定します。選択できる期間は、1、2、4、12、または 24 時間です。
8. ページを保存する準備ができたら、以下の手順を実行します。
   a) 「**ページ名**」フィールドに、ページの名前を入力します。

      重要 : 「**ページ名**」フィールドでは、スペース、下線 (_)、およびダッシュ (-) を使用できます。ただし、ダッシュの後に下線を続ける (-_) ことはできません。例えば、System-_Overview は使用できません。

   b) 「**保存**」をクリックします。
      ダッシュボードで以下の変更が起こるか、メッセージが表示される場合があります。
      - 「ダッシュボードが保存されました」というメッセージが表示されます。
      - 「**条件の設定 (Set Conditions)**」で ***** が選択されていると、次のメッセージが表示されます。

        「リソース・インスタンス」または「条件の設定 (Set Conditions)」で * を選択しました。これにより、多数のデータ系列 (グラフ上の折れ線など) が表示されます。(You selected * in Resource Instance or in Set Conditions, which will result in a large number of data series (such as lines on a graph).) 多数のデータ系列により、ページの読みやすさやパフォーマンスに悪影響を及ぼす場合があります。(The large number of data series can make the page readability or performance unusable.) このグラフに推奨される制限は、50 個のデータ系列です。(The advisable limit for this chart is 50 data series.) 特定の値を追加することにより、推奨される制限内にデータを絞り込み、ユーザー・エクスペリエンスを向上させることができます。(Adding specific values helps to narrow the data within recommended limits and results in a better user experience.)

      - 赤色のインディケーターが に表示され、グラフ・タイプが非選択のため、選択する必要があることを示します。
      - 赤色のインディケーターが に表示され、「メトリックの選択」が非選択のため、選択する必要があることを示します。また、次のメッセージが表示されます。

        グラフを保存するには、メトリックが保存されている必要があります。
9. ページ・タイトル・バーの以下のオプションのいずれかを選択します。

オプション	説明
最新のリソース・メトリック・タイプを取得します	クリックすると、メトリック・タイプが更新されます。エージェント・パッチの適用時にメトリック・タイプまたはメトリックの変更があった場合、メトリック・タイプを更新する必要があります。 更新から次の更新まで間隔は、15 分に制限されています。 「**最新のリソース・メトリック・タイプの取得**」を、前回の更新から 15 分以内にクリックすると、次のメッセージが表示されます。 メタデータ・キャッシュは最近更新されています (Metadata cache was refreshed recently)。 time_remaining 分待ってから再ロードしてください (Please wait for *time_remaining* minute(s) to reload it)。 メタデータのロード中は、ロード中の画像が表示されます。

オプション	説明
	メタデータがロードされると、次のメッセージが表示されます。 メタデータ・キャッシュが正常に再ロードされました (Metadata Cache reloaded successfully)。 メタデータの更新が 30 秒より長くなる場合、次のメッセージが表示されます。 メタデータ・キャッシュの再ロードに時間がかかる可能性があります (Reloading of Metadata cache might take some more time)。 完了するまで待機しますか? (Do you want to wait until it is done?) 「**OK**」または「**キャンセル**」をクリックできます。
ダッシュボードの表示	クリックしてダッシュボードにデータを表示します。 **重要 :** データ定義ごとに返される行数の制限は、11,000 行です。デフォルトでは、制限を超えると最新のデータが表示されます。データ量が多い場合は、選択した時間間隔のデータ全体が表示されるわけではありません。例えば、大量のデータ・ソースに対して、過去 24 時間のデータを表示することを選択した場合、11,000 行の制限に達すると、過去 6 時間分のデータのみが表示される可能性があります。 グラフのデータ系列が 50 を超えると、ウィジェットに次のメッセージが表示されます。 データ系列 (グラフ上の折れ線など) の数が 50 を超えているため、このグラフはロードできません。(This chart cannot be loaded because the number of data series (such as lines on a graph) exceeds 50.) データ系列の数は、現在 *current_data_series* です。(The number of data series is currently *current_data_series*.) より少ないメトリックまたはリソース・インスタンスを選択するか、メトリックごとに条件を調整することで、数を削減できます。(You can reduce the number by selecting fewer metrics or resources instances, or by refining the Conditions per Metric.) 詳しくは、『ウィジェット・プロパティーの定義』 (Cloud APM: http://ibm.biz/widgetprops、 Cloud APM Private: http://ibm.biz/widgetprops-private) を参照してください。 グラフをロードするのに 30 秒を超える時間がかかると、ウィジェットに次のメッセージが表示されます。 大量のデータ、長いネットワーク待ち時間、または接続の問題が原因で時間がかかりすぎたため、このグラフをロードできませんでした。(This chart took too long to load because of a large amount of data, long network latency, or connectivity issue.) リソース・インスタンスの数を減らすか、メトリックごとの条件を調整して、データを絞り込んでください。(Reduce the number of Resource Instances or refine the Conditions per Metric to narrow down the data.) 詳しくは、『ウィジェット・プロパティーの定義』 (Cloud APM: http://ibm.biz/widgetprops、 Cloud APM Private: http://ibm.biz/widgetprops-private) を参照してください。
別名保存	別の名前でページを保存するには、「**保存**」の横の矢印をクリックし、「**別名保存**」をクリックし、「**ページ名**」フィールドに別の名前を指定します。

オプション	説明
	重要 : 既存のページと同じページ名を指定すると、既存のページが上書きされます。
削除	クリックして現在のページを削除します。
Back **戻る**	クリックすると前のページまたはお気に入りのページに戻ります。

次のタスク

1120 ページの『カスタム・ページの表示』の説明に従って、カスタム・ページを表示します。

テンプレートのカスタマイズ

要件に応じてウィジェット・プレースホルダーをサイズ 変更、移動、追加することにより、テンプレートをカスタマイズできます。

このタスクについて

要確認 : 既存のテンプレートをカスタマイズして使用できます。ただし、カスタマイズしたテンプレートを今後の使用のために保存して、新しいダッシュボードを作成することはできません。

手順

1. 「カスタム・ビュー」タブで、 「**テンプレートの編集**」をクリックします。
2. ウィジェット・プレースホルダーを選択します。
   すべての側面からウィジェット・プレースホルダーのサイズを 変更し、別の場所にドラッグすることができます。ウィジェットのサイズ 変更やドラッグを行っているときに、ウィジェットが重なり合ってしまうと、「無効なサイズ変更操作です」や「無効な移動操作です」というメッセージが表示されます。
3. 既存のテンプレートにウィジェットのプレースホルダーを 追加するには、以下の手順を実行します。
   a) メニュー・オプションで「**ページの高さの設定**」をクリックして行数により大きい値を指定し、メニューの外の任意の場所をクリックしてページの高さを増加させます。
   b) ページ上の空白領域にポインターを置いてドラッグすることによりボックスを作成して、要件にかなったウィジェットのプレースホルダーを 指定します。
4. 以下のメニュー・オプションを使用して、テンプレートの各種操作を実行します。

オプション	説明
元に戻す	最後のアクションを元に戻します。
やり直し	最後のアクションをやり直します。
選択したボックスの削除	ウィジェットを削除するには、ウィジェットを選択し、「**選択したボックスの削除**」アイコンをクリックします。
リセット	空白のテンプレートを作成します。 空白のテンプレートのウィジェット・プレースホルダーを指定するには、「ここにテンプレートを描画」領域にポインターを置き、それをドラッグしてボックスを作成します。「ここにテンプレートを描画」領域で、さまざまなサイズのウィジェットのプレースホルダーを 作成できます。このプレースホルダーは移動したり、サイズ変更したりすることができますが、重ね合わせることはできません。
ページの高さの設定	ページの高さを設定します。20 行から 120 行までの行数を指定できます。

5. 「**テンプレートの編集**」をクリックして、作成したテンプレートを使用します。

次のタスク
ウィジェットを作成します。『カスタム・ページの作成および管理』トピックのステップ 6 に進みます。

ウィジェット・プロパティーの定義
メトリックやグラフなどのウィジェットの各種プロパティーを定義して、ウィジェットにリアルタイム・データを表示します。

手順

ウィジェットのプロパティーを定義するには、以下の手順を実行します。

1. ウィジェットで をクリックし、データを表示するグラフ・タイプを選択します。
   - 折れ線
   - 面
   - 棒
   - グリッド

   重要 : 折れ線グラフ、面グラフ、および棒グラフでは、凡例が 9 つを超える場合には、9 つの凡例の後でグラフの色が繰り返されます。グラフの色は 1 番目と 10 番目の凡例で同じになり、2 番目と 11 番目の凡例で同じになります。この後の凡例についても同様です。

   緑色のインディケーターが に表示され、グラフ・タイプが選択されていることを示します。
2. 折れ線グラフ、面グラフ、および棒グラフの以下のグラフ・プロパティーを指定します。
   - X 軸ラベル
   - Y 軸ラベル
   - 凡例の表示
   - 補間の表示 (Show Interpolation): グラフのプロット用に収集されるデータには、Null 値が含まれることがあります。このため、グラフのプロット時に Null 値が検出されると、グラフの線が途切れ、途切れた線がチャート内に複数表示されます。補間を選択すると、Null 値が検出された場合でも、グラフ上の線が途切れて表示されずに、次の使用可能な有効値までつながって表示されます。したがって、補間を選択すると、途切れのない単一のグラフ線が表示されます。

     注 : APM V8.1.4.0 IF0005 以降のバージョンでは、折れ線グラフと面グラフは、NULL 値に対して途切れた線を表示しなくなりました。これにより、「補間の表示」機能が必要なくなったため、サポートされなくなりました。

   重要 : グリッドにはプロパティーはありません。
3. をクリックして、メトリックのコンテンツを選択します。

オプション	説明
リソース・タイプ	「**リソース・タイプ**」リストから、リソースを選択します。使用可能なリソースは、アプリケーションに関連付けられています。 アプリケーションの一部であるリソースがリストで選択不可の場合、そのリソース定義は見つかりませんでした。リソース定義が公開されていないか、管理対象システムが Cloud APM サーバーに接続されていません。
メトリック・タイプ	「**メトリック・タイプ**」リストで、ウィジェットに含めるデータ・セットを選択します。
メトリック	「**メトリック**」リストから、ビューに含める属性を選択します。使用可能な属性は、選択したデータ・セットから取得されます。 メトリックを選択するには、以下のステップを実行します。 a.「**メトリック**」リストをクリックします。

オプション	説明
	メトリックがアルファベット順にリストされ、昇順にソートされたポップアップ・ウィンドウが開きます。 b.「メトリック」 (1 つ以上を選択) の下にリストされている属性をクリックするか、「**すべて選択**」をクリックします。 **注 :** 「**すべて選択**」をクリックすると、リスト内のすべてのメトリックが選択されます。 c. をクリックし、「**選択されたメトリック**」リストの下にメトリックを追加します。 d.「**選択されたメトリック**」の下のメトリックを削除する場合は、 をクリックします。 e.「**リソース・インスタンス**」リストをクリックして、ポップアップ・ウィンドウを閉じます。 **重要 :** 折れ線グラフ、棒グラフ、および面グラフの場合、数値を含むメトリックを選択する必要があります。ストリング値を含むメトリックは、これらのグラフには表示できません。 **ヒント :** グリッドの場合、UI の表示に適した出力に応じて、メトリックの選択を制限します。
リソース・インスタンス	初期状態では、選択項目は * で、リスト内のすべてのインスタンスからメトリックが取得されます。デフォルトの選択を維持するか、リストからインスタンスを選択します。 **重要 :** インスタンスを選択する場合、このウィジェットを使用して他のエージェントやインスタンスのデータを表示することはできません。ただし、インスタンスを指定することには、大量のデータを処理することを回避するという利点があります。
メトリック・グループに対する条件の設定	選択した**メトリック・タイプ** に複数のエレメント (CPU やディスクなど) が含まれている場合、「**メトリック・グループに対する条件の設定**」に他のエレメントが表示され、「**対象**」フィールドでそのエレメントを選択します。 **重要 :** 「**対象**」フィールドにエレメントの値を指定してください。大量データの処理を減らすために、***** の指定は避けてください。 デフォルトでは、WHERE 条件は過去 4 時間のデータを表示します。システム管理者は、この時間間隔を 1 から 24 時間の間で変更できます。 1119 ページの『WHERE 条件データの時間間隔の変更』を参照してください。
アクション	メトリックを保存するには、「**保存**」をクリックします。 メトリックを編集するには、 「**編集**」をクリックします。 メトリックを削除するには、 「**削除**」をクリックします。

4. 別のメトリックを追加するには、「**+ 別のメトリックを追加**」をクリックします。

   重要 : グリッドには適用されません。
5. すべてのメトリックを追加したら、「**メトリックの選択**」ウィンドウを閉じます。
   「**メトリックの選択**」ウィンドウを閉じると、すべてのメトリックが自動的に保存されます。

   ダッシュボードで以下の変更が起こるか、メッセージが表示される場合があります。

- いずれかのメトリックの「**リソース・インスタンス**」リストで ***** が選択されていると、次のメッセージが表示されます。

  「リソース・インスタンス」または「条件の設定 (Set Conditions)」で * を選択しました。これにより、多数のデータ系列 (グラフ上の折れ線など) が表示されます。(You selected * in Resource Instance or in Set Conditions, which will result in a large number of data series (such as lines on a graph).) 多数のデータ系列により、ページの読みやすさやパフォーマンスに悪影響を及ぼす場合があります。(The large number of data series can make the page readability or performance unusable.) このグラフに推奨される制限は、50 個のデータ系列です。(The advisable limit for this chart is 50 data series.) 特定の値を追加することにより、推奨される制限内にデータを絞り込み、ユーザー・エクスペリエンスを向上させることができます。(Adding specific values helps to narrow the data within recommended limits and results in a better user experience.)

- 緑色のインディケーターが に表示され、メトリックが正しく選択されているため、推奨される制限内にデータが絞り込まれていることを示します。

- オレンジ色のインディケーターが に表示され、メトリックが正しく選択されていないため、推奨される制限内にデータが絞り込まれていないことを示します。「**リソース・インスタンス**」または「**条件の設定 (Set Conditions)**」のいずれかで、***** が選択されています。

6. をクリックして、ウィジェットのタイトルを入力します。

   ウィジェットのタイトルを追加しないと、最初のメトリック名がウィジェットのタイトルとして自動的に割り当てられます。

次のタスク

同じようにして、すべてのウィジェットにグラフ、メトリック、およびタイトルを追加してから、『カスタム・ページの作成および管理』トピックのステップ 7 に進みます。

WHERE 条件データの時間間隔の変更

システム管理者は、時間間隔を 1 から 24 時間の間で変更できます。

手順

システム管理者は、以下のステップを実行して時間間隔を変更できます。

1. ビルドがデプロイされている APM サーバーにログインします。
2. コマンド行で、以下のコマンドを実行します。

   ```
   export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derbytools.jar:
   export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derbyclient.jar:
   export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derby.jar:
   java org.apache.derby.tools.ij
   connect 'jdbc:derby://localhost:port/
   gaiandb;user=gaiandb;password=gaian_db_password;';
   ```

   これらのコマンドで、*install_dir* は APM がデプロイされたディレクトリーを指します。デフォルトでは /opt/ibm です。**connect** コマンドにおいて、*port* はデータベースが構成されているポートの値であり、*gaian_db_password* は Gaian データベースのパスワードです。このパスワードについては IBM サポートにお問い合わせください。
3. データベースが接続されたら、次の照会を実行して時刻範囲の値を変更します。

   ```
   UPDATE "OED_TOOL"."PREFERENCETABLE" SET PREFERENCES='-24H' WHERE
   FIELD='TIMEINTERVAL';

   Commit;
   ```

```
Exit;
```

ここで、時刻範囲の値は SET PREFERENCES='-24H' で指定されています。この値は 1 H から 24 H の間に設定できます。

カスタム・ページの表示

「**カスタム・ビュー**」タブでアプリケーション、グループ、サブグループ、またはインスタンス用のダッシュボード・ページを作成および保存した後、いつでもそれらのページを表示できます。 選択できるオプションには、ページの最新表示、異なる時間間隔の選択、異なるリソースからデータを取得するためのページの編集、ダッシュボードの生データ・ファイルへのエクスポートなどがあります。

手順

保存されたページをダッシュボードの「**カスタム・ビュー**」タブで表示するには、以下の手順を実行します。

1. 「**パフォーマンス**」メニューから、「Application Performance ダッシュボード」を開いた後、アプリケーションを選択します。

   「**カスタム・ビュー**」タブが「**状況の概要**」および「**イベント**」タブの後に表示されます。ナビゲーターのグループ、サブグループ、またはインスタンスの各レベルにドリルダウンすることもできます。
2. 「**カスタム・ビュー**」タブを選択します。

   このタブには、「**カスタム・ページのテンプレートの選択**」ウィンドウが表示されるか、デフォルト・ページが既に設定されている場合は、そのページが表示されます。
3. ページ・リストの をクリックし、リストからいずれかの保存済みページを選択します。

   使用できるページは、ユーザーによって保存されたか、別のユーザーによって共有されています。

   保存されたページを選択すると、現在のデータ・サンプルおよびヒストリカル・データ・サンプルがページで報告されます。
4. ページ・タイトル・バーの表示オプションのいずれかを選択します。

オプション	説明
最新表示	自動最新表示がオフになっていることを示します。クリックすると、自動最新表示がオンになります。
最新表示	自動最新表示がオンになっていることを示します。クリックすると、自動最新表示がオフになります。 **重要 :** デフォルトの最新表示時間は 1 分です。
「**エクスポート**」 > 「**生データ (Raw Data)**」	クリックすると、ページが DAT 形式でエクスポートされます。複数の DAT ファイルがエクスポートされているため、ZIP 形式でダウンロードされます。 **要確認 :** ダウンロードしたファイルに拡張子がない場合は、そのファイルに拡張子として zip を追加してください。 **重要 :** このファイルがコンピューターにダウンロードされない場合は、ポップアップ広告ウィンドウをブロックするソフトウェアが有効になっていないか確認してください。このサイトを例外リストに追加できます。 ダウンロードした ZIP ファイルを解凍します。解凍したファイルは、プレーン・テキストのファイルです。このファイルには、ページ名、期間、フィルター、日付、時間、間隔、グラフのタイトル、およびデータが含まれています。データ区切り文字はパイプです。

オプション	説明
	適切なエディターを使用して DAT ファイルを開くか、適切な値の分離文字を指定してこのファイルを Excel にインポートすることができます。
「**エクスポート**」 > 「**PDF**」	クリックすると、ページが PDF 形式でエクスポートされます。 ファイルには、ページ名、時間間隔、ウィジェット、作成元、レポート作成日が含まれます。
編集	クリックすると、現在のページを編集できます。 ウィジェット内のグラフおよびメトリックを変更すること、新しいウィジェットを追加すること、デフォルトの時間フレームを変更することや、ページの名前を編集することができます。
削除	クリックして、現在のページを削除します。
追加	クリックすると、新規ページが作成されます。ページをカスタマイズして保存するには、1113 ページの『カスタム・ページの作成および管理』を参照してください。

5. ウィジェットの以下の表示オプションのいずれかを選択します。

オプション	説明
「**グラフのタイプ**」	既存のグラフ・タイプを変更するには、「**グラフ・タイプ**」アイコンをクリックし、リストから適切なオプションを選択します。 • 折れ線グラフおよび面グラフの場合、「**折れ線**」オプションおよび「**面**」オプションを使用できます。 • 棒グラフの場合、「**集合横棒 (Clustered Bar)**」オプションおよび「**集合縦棒 (Clustered Columns)**」オプションを使用できます。 **重要 :** グリッドの場合、グラフ・タイプ・オプションはありません。 グリッドに表示されるデータの場合、以下のようにデータをフィルター操作できます。 a. 「**フィルターの定義**」をクリックします。「**フィルター**」ウィンドウが開きます。 b. 「**列**」、「**条件**」、および「**値**」の値を指定し、フィルター・ルールを追加します。 **注 :** 適切な条件を選択して、数値およびテキスト値をフィルタリングできます。 c. フィルター・ルールをさらに追加するには、「**フィルター・ルールの追加**」をクリックします。複数のフィルター・ルールを追加できます。 d. 「**一致 (Match)**」フィールドで、データのフィルタリングの基準として「**すべてのルール (All rules)**」または「**いずれかのルール (Any rule)**」をクリックします。 「**値**」フィールドに指定したテキストの大/小文字の区別に従って検索を行う場合は、「**大/小文字の区別の一致 (Match case)**」を選択します。

オプション	説明
	e.「**フィルター**」をクリックして、グリッドに表示されるデータをフィルタリングします。 f.「**フィルターのクリア**」をクリックして、フィルター結果をクリアします。 g.「**キャンセル**」をクリックして「**フィルター**」ウィンドウを閉じます。
「**省略**」	クリックしてウィジェットを省略します。
「**展開**」	クリックして、ウィジェットを展開します。
「**最大化**」	クリックして、ページのサイズまでウィジェットを最大化します。
「**復元**」	クリックして、ウィジェットを元のサイズに復元します。
凡例	ウィジェットには、各メトリックのチェック・ボックスが含まれます。各メトリックのチェック・ボックスを選択またはクリアすることで、特定のメトリックまたは複数のメトリックのデータを表示できます。

6. 「**日付**」、「**時刻**」、および「**間隔**」の各リストを使用して、ページ上のデータをフィルタリングできます。ページに対してカスタム・フィルターを定義して、選択した日付と時間間隔のデータを表示することもできます。カスタム・フィルターを使用するには、「**間隔**」リストで「**カスタム**」を選択し、「**期間の選択**」ウィンドウで必要な日付と時間間隔を選択します。

   注:

   - 「カスタム」フィルター・オプションは、APM V8.1.4.0 IF0005 以降で使用できます。それより前の Cloud APM のバージョンで作成されたページには、「カスタム」フィルター・オプションは表示されません。
   - カスタム・フィルターは、最小の時間間隔である 1 分から最大の時間間隔である 24 時間のデータをフィルタリングするために使用します。
   - カスタム・フィルターを適用しているときに、「**期間の選択**」ウィンドウで「**キャンセル**」をクリックすると、前に適用した間隔はダッシュボード・ページの「**間隔**」リストに表示されません。
   - ページにカスタム・フィルターを適用すると、そのページのデータは自動最新表示されません。

7. デフォルト・ページを設定するには、ページ・リストの をクリックし、デフォルト・ページとして設定するページ名の横の 「**お気に入り (Favorite)**」をクリックします。

ダッシュボード・ユーティリティー

使用可能なオプションを使用して、**Application Performance ダッシュボード**・ページの外観と動作を管理します。

ダッシュボード URL のコピー

アプリケーション階層内のいずれかの場所にナビゲートした後も、 ブラウザーのアドレス・ボックス内の URL は新しいビュー用に変更されません。 表示している Application Performance ダッシュボード ページの URL をコピーできます。その URL を新しいブラウザー・ウィンドウに貼り付けてダッシュボード・ページを開いたり、その URL を使用して後でダッシュボードにアクセスしたり、他のユーザーと共有したりします。

手順

1. 自分が覚えておきたい Application Performance ダッシュボード ページにナビゲートします。

2.「**アクション**」 > 「**URL のコピー**」をクリックします。
3.「**現行ページへのリンク**」ハイパーテキスト・リンクを右クリックし、URL をコピーするためのオプションを選択します。

次のタスク
URL のコピーを保持するか、管理対象環境内の他のユーザーと共有します。ブラウザーのアドレス・ボックスに URL を貼り付けると、ターゲットのダッシュボード・ページが Cloud APM コンソール で開きます。

Cloud APM サーバー にログオンしていない場合、ターゲットのダッシュボード・ページが表示される前に、ユーザー ID とパスワードの入力を求めるプロンプトが出されます。ダッシュボード・ページではなく「**はじめに**」ページが開いた場合は、F5 を押すか、ブラウザーの最新表示ツールバー・ボタンをクリックします。「始動時にこの「**はじめに**」ページを表示する」チェック・ボックスをクリアすることにより、今後のワーク・セッションで「**はじめに**」ページをオフにすることができます。

トレースの設定

管理者または IBM サポートが Application Performance ダッシュボードでの問題の原因を診断する上で役立つように、トレース設定を調整してください。ナビゲーターおよび「**状況の概要**」タブで作業するときのトレース・レベルはいくつか用意されています。ユーザー・インターフェースでまさに 問題が生じているポイントで詳細なレベルのトレースを開始し、必要なログ・データを収集した後、トレースを低いレベルに戻すことができます。例えば、特定のダッシュボードの動作が予期しないものである場合、トレース・レベルを上げてから、ダッシュボードを開いてアクティビティーをログに 記録し、その後、トレース・ロギングを通常レベルに戻すことができます。

このタスクについて
トレース・ロギングの量を増減する場合は、以下のステップを実行してトレース・レベルを設定します。

手順

1. Application Performance ダッシュボードが開いていない場合は、ナビゲーション・バーの「**パフォーマンス**」オプションからそれを選択します。
2. ナビゲーターまたは「状況の概要」タブから「すべてのマイ・アプリケーション」またはアプリケーションを選択します。
3.「**アクション**」 > 「**トレース・レベル**」をクリックし、以下のレベルのいずれかを選択します。
   - 「**詳細**」。すべてのアクティビティーがログに 記録されます。 「詳細」トレース・レベルには、「中」、「軽」、および「最小限」の各トレース・ロギングが含まれます。
   - 「**中**」。渡されたパラメーターや行われた計算など、変数の変更がログに記録されます。「中」トレース・レベルには、「軽」および「最小限」の各トレース・ロギングが含まれます。
   - 「**軽**」。エラーおよび変数のアクティビティーがログに 記録されます。 データが返されないがダッシュボードが引き続き機能しているといった問題が発生している場合には、トレースをこのレベルに設定することをお勧めします。 「軽」トレース・レベルには、「最小限」トレース・ロギングが含まれます。
   - 「**最小限**」。これは、デフォルト設定であり、復旧不能エラーのみを記録します。特定のアクティビティー・シーケンスを収集した後に、トレース・レベルを「最小限」に戻すことができます。ログアウト前に別のトレース・レベルが設定されていた場合でも、次回ログイン時には常に、トレースは最低レベルにリセットされます。
4. パフォーマンス・レコードを共通のログ・ファイルに送信する場合、「**パフォーマンス統計のログ記録を使用可能にする**」を選択します。

   コンソールからのパフォーマンス情報がサーバーに書き込まれます。サーバー上で、パフォーマンス情報はサーバーからのパフォーマンス統計と結合されて、エンドツーエンドのトランザクション応答時間を提供することができます。必要なパフォーマンス情報には、機能が開始された時間と終了した時間が含まれます。

タスクの結果
選択したレベルにトレースが調整されます。次にログインしたときには、再度変更するまでトレースは「**最小限**」になります。

通信トラフィックを最小限に抑えるために、ログ・メッセージはバッチで転送されます。最終転送は、手動で実行する場合でもタイムアウト期間の経過後に実行する場合でも、ログアウト後に行われます。(ブラウザーで障害が発生する場合は、最終ロギングは送信されません。) ログはサーバー・コンピューター上に保存され、`itp.log` という名前が付けられます。サーバーが再始動するたびに、新しい `itp.log` が作成されます。

「**パフォーマンス統計のログ記録を使用可能にする**」を設定した場合、以下の例に類似したレコードが *install_dir*`/usr/servers/apmui/logs/itp.log` に保存されます。

```
<record>
    <date>2013-10-02T10:52:46</date>
    <millis>1380736366788</millis>
    <sequence>28008</sequence>
    <level>INFO</level>
    <class>StatusItemList</class>
    <method>tracing</method>
    <thread>96</thread>
    <message>BeginTrace:onSelectApp:272wt877d05</message>
</record>
<record>
    <date>2013-10-02T10:52:46</date>
    <millis>1380736366809</millis>
    <sequence>28009</sequence>
    <level>INFO</level>
    <class>StatusItemList</class>
    <method>tracing</method>
    <thread>96</thread>
    <message>EndTrace:onSelectApp:272wt877d05</message>
</record>
```

Cloud APM コンソールのロック

Cloud APM コンソールからログアウトする必要なしに、ワーク・セッションを一時的にロックすることができます。 セッションのロック機能は、Apple iPad では使用できません。

手順

1. Cloud APM コンソールにログインしているときに、 「***apmadmin***」 > 「**セッションのロック**」をクリックします。ここで、*apmadmin* はログインに使用した名前です。
   ログイン画面が表示され、セッションはロックされます。
2. セッションをアンロックするには、ユーザー ID のパスワードを入力します。
   ワーク・セッションが再開されます。

レポート

Cloud APM コンソールで、Response Time Monitoring エージェント、WebSphere Applications エージェント、および Synthetic Playback エージェントによって収集されたデータに対するヒストリカル・レポートが使用可能です。

「**すべてのマイ・アプリケーション**」ダッシュボードからレポートを実行できます。任意の Cloud APM コンソール・ページから、 「**パフォーマンス**」 > 「**Application Performance ダッシュボード**」をクリックして「**すべてのマイ・アプリケーション**」ダッシュボードを開きます。

注 : レポートを初めて実行するときは、Cloud APM レポートを実行する権限を持つユーザーとして Tivoli Common Reporting にログインする必要があります。IBM Cognos Viewer は、デフォルトのレポート出力ビューアーです。

Response Time Monitoring エージェント レポート

「**アプリケーションのパフォーマンスおよび 使用率**」または「**2 つの期間でのアプリケーション・パフォーマンスの比較**」レポートを表示するには、Response Time Monitoring エージェントの管理対象システムを含むアプリケーションを選択し、「**アクション**」 > 「**レポートの起動**」を選択します。

「**すべてのマイ・アプリケーション**」レポートまたは「**複数のアプリケーションのパフォーマンスの 比較**」レポートを表示するには、「**すべてのマイ・アプリケーション**」を選択し、「**アクション**」 > 「**レポートの起動**」を選択します。

WebSphere Applications エージェント レポート

WebSphere Applications エージェント の各レポートを表示するには、WebSphere Applications エージェント の管理対象システムを含むアプリケーションを選択し、「**アクション**」 > 「**レポートの起動**」を選択します。

Synthetic Playback エージェント

Synthetic Playback エージェント の各レポートを表示するには、シンセティック・トランザクションを含むアプリケーションを選択し、「**アクション**」 > 「**レポートの起動**」を選択します。

注 : オプションの「**レポートの起動**」が「**アクション**」メニューに表示されない場合は、Cloud APM レポートが正しくインストールされていることを確認してください。

Synthetic Playback エージェント、Response Time Monitoring エージェント、および WebSphere Applications エージェントのレポートがサポートされるブラウザーについては、Cognos 10.2.1.7 の Software Product Compatibility Reports を参照してください。

Response Time Monitoring エージェント レポート

Response Time Monitoring エージェントによって収集されるデータについて、履歴レポートが使用可能です。Response Time Monitoring エージェント・レポートは Cloud APM, Base では使用できません。これらは Cloud APM, Advanced でのみ使用できます。

Response Time Monitoring エージェントによって収集されるデータについて、2 つのタイプのレポート (アクティブと簡易) が使用可能です。

アクティブ・レポート
アクティブ・レポートは、MHTML 形式でブラウザーに表示されます。Internet Explorer は、デフォルトで MHTML をサポートしています。他のブラウザーについては、MHTML サポート・プラグインをインストールできます。アクティブ・レポートは、オフライン対話式レポートとも呼ばれます。

簡易レポート
簡易レポートは、IBM Cognos Viewer で表示します。IBM Cognos Viewer は、デフォルトのレポート出力ビューアーです。

Response Time Monitoring エージェントによって収集されるデータについて、以下の事前定義の履歴レポートが使用可能です。

表 *254.* 事前定義済みの履歴レポート

レポート	タイプ
すべてのマイ・アプリケーション	アクティブ
アプリケーションのパフォーマンスおよび 使用状況	アクティブ
2 つの期間でのアプリケーション・パフォーマンスの比較	簡易
複数のアプリケーションのパフォーマンスの 比較	簡易

レポートのためのデータは DATAMART という Db2 データベースに格納されます。レポートには、1 時間ごと、1 週間ごと、および 1 カ月ごとに要約したデータが表示されます。これらのデータは、それぞれ 26 週間、12 カ月間、および 3 年間保存されます。Cloud APM に、これらの保存期間を変更するためのスクリプトや手順は用意されていません。

Response Time Monitoring エージェントと Performance Management のレポートとの間のマッピングについて詳しくは、『Response Time Monitoring エージェントの属性マッピング』を参照してください。

「すべてのマイ・アプリケーション」レポート
「すべてのマイ・アプリケーション」レポートを使用して、ユーザー・デバイス、データ・ボリューム、応答時間、およびエラー件数に関する情報を表示します。

このレポートには、すべてのアプリケーションの 情報が表示されます。 レポート期間には、「**前日**」 (デフォルト)、「**先週**」、または「**先月**」を指定します。選択した期間別、およびアプリケーション別に、次の情報を閲覧します。

- トランザクション数の積み重ね縦棒グラフ
- トランザクション・データ・ボリュームの縦棒グラフ
- トランザクションの平均応答時間の縦棒グラフ
- エラー件数の積み重ね縦棒グラフ

「アプリケーションのパフォーマンスおよび使用率」レポート
このレポートを使用して、単一のアプリケーションのパフォーマンス、可用性、およびユーザー・デバイス情報を表示します。

「**アプリケーションの選択**」ウィンドウで、アプリケーションを選択します。「**次へ**」をクリックします。「**アプリケーションの重要トランザクションの選択**」で、レポートをフィルターに掛けるトランザクションを選択します。「**OK**」をクリックします。このレポートには、「パフォーマンス」、「可用性」、および「デバイス」の 3 つのタブがあります。 デフォルトの時間間隔は週です。

「パフォーマンス」タブには、Application Performance ダッシュボードに現在表示されているアプリケーションの次の情報が、選択した時間間隔別に表示されます。

- (主要な) トランザクション別の平均応答時間の折れ線グラフ
- 成功、サーバー・エラー、およびクライアント・エラー別のトランザクションの平均応答時間の折れ線グラフ
- トランザクション・データ・ボリュームの棒と折れ線のグラフ。棒はトランザクション・データ・ボリュームを示し、折れ線はトランザクション・データ・ボリュームの平均を示します
- トランザクション数の棒と折れ線のグラフ。棒はトランザクション数を示し、折れ線は多項式値と移動平均を示します

「可用性」タブには、Application Performance ダッシュボードに現在表示されているアプリケーションの次の情報が、選択した時間間隔別に表示されます。

- 成功したトランザクションと失敗したトランザクションのパーセンテージを示す積み重ね棒グラフ。失敗はサーバー・エラーとクライアント・エラーに分けられます
- 成功したトランザクション数と失敗したトランザクション数をデバイス・タイプ別に示す積み重ね棒グラフ。失敗はサーバー・エラーとクライアント・エラーに分けられます
- 最も頻繁に発生したエラー・コードを示す円グラフ

「デバイス」タブには、Application Performance ダッシュボードに現在表示されているアプリケーションの次の情報が、選択した時間間隔別に表示されます。

- デバイス・タイプ別のトランザクション数の棒グラフ
- デバイスのオペレーティング・システム別のトランザクション数の棒グラフ
- デバイスのブラウザー別のトランザクション数の棒グラフ
- 各種ディメンション別のトランザクション・パフォーマンスを示す表。この表は、デバイス・タイプ、デバイスのオペレーティング・システム、デバイスのブランド、およびデバイスのブラウザーに基づいてフィルターに掛けることができます

「2 つの期間でのアプリケーション・パフォーマンスの比較」レポート
このレポートを使用して、選択したアプリケーションのアプリケーション ・パフォーマンスを調べます。

「**アプリケーションおよび時間頻度の選択**」ウィンドウで、アプリケーションおよび時間頻度 (毎週、毎日、毎月) を指定します。 「**期間の選択**」ウィンドウで、時間間隔に対して該当する期間を選択し、「**OK**」をクリックします。

このレポートには、選択したアプリケーションの選択した期間について、選択した時間間隔で以下のグラフが表示されます。

- デバイス・タイプのトランザクション数の折れ線グラフ
- トランザクション・ボリュームの折れ線グラフ
- 平均応答時間の折れ線グラフ
- エラー件数の折れ線グラフ

「複数のアプリケーションのパフォーマンスの比較」レポート
このレポートを使用して、同じ期間での複数のアプリケーションのパフォーマンスを 比較します。

「**アプリケーションおよび時間頻度の選択**」で、アプリケーションおよび時間頻度 (毎週、毎日、毎月) を指定します。 「**次へ**」をクリックします。時間間隔に対して該当する期間を選択します。

このレポートには、選択したアプリケーションの選択した期間の情報を、選択した時間頻度別に示すグラフが表示されます。

- トランザクション数の折れ線グラフ
- トランザクション・データ・ボリュームの折れ線グラフ
- 平均応答時間の折れ線グラフ
- エラー件数の折れ線グラフ

Response Time Monitoring エージェントの属性マッピング
一部の Cloud APM レポートは、Response Time Monitoring エージェントによって収集されたデータに基づいています。これらのレポートのデータは、Response Time Monitoring エージェントの属性にマップされます。

以下の表は、Response Time Monitoring エージェントのレポートのデータ項目とエージェントの属性との間のマッピングを示しています。

表 *255. Response Time Monitoring* エージェントの属性マッピング

レポートのデータ項目	説明	ODI 属性名	ODI ファイルの列
アプリケーション名	Cloud APM コンソールに報告されるモニター対象アプリケーションの名前	アプリケーション名	T5TXCS.APPLICATIN
トランザクション数	現在の集約間隔中にモニター・エージェントによって監視された要求と応答のシーケンスの合計数。	要求総数	T5TXINS.TOTREQ
クライアント・エラー数	状況コードが 400 から 499 の HTTP 要求の数。	クライアント・エラー数	T5TXCS.NUM4XX
サーバー・エラー数	状況コードが 500 から 599 の HTTP 要求の数。	サーバー・エラー数	T5TXCS.NUM5XX

表 *255. Response Time Monitoring* エージェントの属性マッピング *(*続き*)*

レポートのデータ項目	説明	ODI 属性名	ODI ファイルの列
トランザクション名	Application Management Console に報告されるトランザクション名。	トランザクション名	T5TXCS.TRANSACTN
トランザクション状況	トランザクションに関連付けられている応答コード	状況コード	T5TXCS.STATUSCODE
コード応答キロバイト数	所定のデータ間隔での要求の各応答のキロバイトの合計数。	応答バイト数	T5TXCS.REPLYBYT
要求キロバイト数	所定のデータ間隔での要求のキロバイトの合計数。	要求 KB	T5TXCS.REQBYTES
キロバイトの合計数	所定の期間のすべての要求で転送されたキロバイトの合計数。	合計バイト数	T5TXCS.TOTBYTES
オブジェクト数の合計	所定の期間に Web ページに埋め込まれたオブジェクトの合計数。	オブジェクト数の合計	T5TXCS.OBJCNT
オブジェクト・サイズの合計	所定の期間に Web ページに埋め込まれたすべてのオブジェクトのサイズの合計。	オブジェクト・サイズの合計	T5TXCS.OBJSIZE
応答時間 (秒)	サーバーのトランザクション全体が完了するまでに必要な時間の合計数 (秒単位)。	応答時間	T5TXCS.RESPTIME
レンダリング時間	Web ブラウザーで Web ページを埋め込み JavaScript タグを使用して完全にレンダリングする経過時間 (秒単位)。	レンダリング時間	T5TXCS.RENDERTIME
クライアント時間	現在のモニター間隔中にトランザクションがクライアントで実行に費やした平均経過時間 (秒単位)。	平均クライアント時間	T5TXCS.CLIENTTIME
ロード時間	ユーザーがダウンロードを要求した時点から Web オブジェクトのダウンロード完了までの平均経過時間 (秒単位)。	平均ロード時間	T5TXCS.LOADTIME

表 255. *Response Time Monitoring* エージェントの属性マッピング (続き)			
レポートのデータ項目	**説明**	**ODI 属性名**	**ODI ファイルの列**
ブラウザー	Web ページが表示される Web ブラウザーの説明。	ブラウザーの説明	T5TXCS.BROWSEDESC
サーバー	TCP トランザクション用のサーバーの名前または IP アドレス。	サーバーの説明	T5TXCS.SERVERDESC
URL ホスト名	URL の TCP/IP ホスト名。	URL ホスト名	T5TXCS.URLHOST
URL メソッド	HTTP 要求の実行に使用されるメソッド (GET, POST, HEAD, PUT, OPTIONS, DELETE, TRACE, または CONNECT)。	メソッド	T5TXCS.METHOD
URL 詳細	Web ページをホストするサーバー上のファイルの URL パス。	URL パス	T5TXCS.URLPATH

Response Time Monitoring エージェントについて詳しくは、Transaction Monitoring Reference を参照してください。

Synthetic Playback エージェントのレポートの生成

シンセティック・トランザクションに関連付けられているアプリケーションについてレポートを実行します。

このタスクについて

Application Performance ダッシュボードで、アプリケーションおよび関連付けられているシンセティック・トランザクションを選択し、選択内容に基づいてレポートを生成します。レポートのためのデータは DATAMART という Db2 データベースに格納されます。レポートには、1 時間ごと、1 週間ごと、および 1 カ月ごとに要約したデータが表示されます。これらのデータは、それぞれ 371 日間、53 週間、および 12 カ月間保存されます。Cloud APM に、これらの保存期間を変更するためのスクリプトや手順は用意されていません。

次の 5 つの複数ページのレポートを使用できます。

トランザクション全体

この 2 ページから成るレポートには、設定した日付範囲中の、選択したシンセティック・トランザクションの応答時間および可用率が表示されます。

ページ 1 には、次のデータが表示されます。

- 選択したシンセティック・トランザクションの、設定した日付範囲中の設定した間隔での応答時間の折れ線グラフ
- 設定した日付範囲中の、各シンセティック・トランザクションの平均応答時間 (秒) の表

ページ 2 には、次のデータが表示されます。

- 設定した日付範囲中の設定した間隔での、選択したシンセティック・トランザクションの可用率の折れ線グラフ
- 設定した日付範囲中の、各シンセティック・トランザクションの平均可用率の表

「**トランザクション全体**」レポートから 2 つの追加レポート「**トランザクション別のタイムリーな分析**」および「**トランザクション別の HTTP メトリック**」にアクセスできます。

「**トランザクション別のタイムリーな分析**」では、設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションの HTTP メトリックを示す縦棒グラフ
- 設定した日付範囲における、選択したシンセティック・トランザクションの HTTP メトリック (ミリ秒) を示す表

「**トランザクション別の HTTP メトリック**」では、設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションの HTTP メトリックを示す縦棒グラフ
- 設定した日付範囲における、選択したシンセティック・トランザクションの HTTP メトリック (ミリ秒) を示す表

ロケーション別のトランザクション詳細

この 2 ページから成るレポートには、設定した日付範囲中の、選択したシンセティック・トランザクションおよびサブトランザクションの、ロケーション別の応答時間および可用率が表示されます。

ページ 1 には、次のデータが表示されます。

- 設定した日付範囲中の設定した間隔での、選択したシンセティック・トランザクションおよびサブトランザクションの、ロケーション別の応答時間の折れ線グラフ
- 設定した日付範囲中の各ロケーションでの、すべてのシンセティック・サブトランザクションの平均応答時間 (秒) の表

ページ 2 には、次のデータが表示されます。

- 設定した日付範囲中の設定した間隔での、選択したシンセティック・トランザクションおよびサブトランザクションの、ロケーション別の可用率の折れ線グラフ
- 設定した日付範囲中の各ロケーションでの、すべてのシンセティック・サブトランザクションの平均可用率の表

「**ロケーション別のトランザクションの詳細**」レポートから 4 つの追加レポート「**トランザクションのロケーション別のタイムリーな分析**」、「**トランザクションのロケーション別の HTTP メトリック**」、「**サブトランザクションのロケーション別のタイムリーな分析**」、および「**サブトランザクションのロケーション別の HTTP メトリック**」にアクセスできます。

「**トランザクションのロケーション別のタイムリーな分析**」では、設定した日付範囲における、設定した間隔でのロケーション別のシンセティック・トランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションのロケーション別の HTTP メトリックを示す縦棒グラフ
- 設定した日付範囲における、選択したシンセティック・トランザクションのロケーション別の HTTP メトリック (ミリ秒) を示す表

「**トランザクションのロケーション別の HTTP メトリック**」では、設定した日付範囲における、設定した間隔でのロケーション別のシンセティック・トランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲における、設定した間隔での選択したシンセティック・トランザクションのロケーション別の HTTP メトリックを示す縦棒グラフ
- 設定した日付範囲における、選択したシンセティック・トランザクションのロケーション別の HTTP メトリック (ミリ秒) を示す表

「**サブトランザクションのロケーション別のタイムリーな分析**」では、設定した日付範囲における、設定した間隔でのロケーション別のサブトランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲中の設定した間隔での、選択したサブトランザクションのロケーション別の HTTP メトリックの縦棒グラフ
- 設定した日付範囲における、選択したサブトランザクションのロケーション別の HTTP メトリック (ミリ秒) を示す表

「**サブトランザクションのロケーション別の HTTP メトリック**」では、設定した日付範囲における、設定した間隔でのロケーション別のシンセティック・サブトランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲中の設定した間隔での、選択したサブトランザクションのロケーション別の HTTP メトリックの縦棒グラフ
- 設定した日付範囲における、選択したサブトランザクションのロケーション別の HTTP メトリック (ミリ秒) を示す表

サブトランザクション別のトランザクション詳細

この 2 ページから成るレポートには、設定した日付範囲中の設定した間隔での、シンセティック・サブトランザクションの応答時間および可用率が表示されます。

ページ 1 には、次のデータが表示されます。

- 設定した日付範囲中の設定した間隔での、選択したシンセティック・サブトランザクションの応答時間の折れ線グラフ
- 設定した日付範囲中の、各シンセティック・サブトランザクションの平均応答時間 (秒) の表

ページ 2 には、次のデータが表示されます。

- 設定した日付範囲中の設定した間隔での、選択したシンセティック・サブトランザクションの可用率の折れ線グラフ
- 設定した日付範囲中の、各シンセティック・サブトランザクションの可用率の表

「**サブトランザクション別のトランザクションの詳細**」レポートから 2 つの追加レポート「**サブトランザクション別のタイムリーな分析**」および「**サブトランザクション別の HTTP メトリック**」にアクセスできます。

「**サブトランザクションのロケーション別のタイムリーな分析**」では、設定した日付範囲における、設定した間隔でのロケーション別のサブトランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲中の設定した間隔での、選択したサブトランザクションの HTTP メトリックの縦棒グラフ
- 設定した日付範囲中の、選択したサブトランザクションの HTTP メトリック (ミリ秒) の表

「**サブトランザクションのロケーション別の HTTP メトリック**」では、設定した日付範囲における、設定した間隔でのロケーション別のシンセティック・サブトランザクションの HTTP メトリックが表示されます。レポートには、以下の項目が含まれます。

- 設定した日付範囲中の設定した間隔での、選択したサブトランザクションの HTTP メトリックの縦棒グラフ
- 設定した日付範囲中の、選択したサブトランザクションの HTTP メトリック (ミリ秒) の表

トランザクションのトレンド

この 4 ページから成るレポートには、先週および直近の 5 週間の応答時間、可用率、および HTTP メトリックのトレンド分析が表示されます。

ページ 1 には、以下のようなシンセティック・トランザクションの応答時間および可用率に関するトレンド・データが表示されます。

- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均応答時間の結合折れ線グラフ

- 先週の可用率を直近の 5 週間のベースライン可用率と比較する、選択したシンセティック・トランザクションの可用率の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の日付範囲の平均応答時間および可用率の表
- 選択したシンセティック・トランザクションの、ロケーション別の先週および直近の 5 週間の平均応答時間の結合折れ線グラフ
- 先週の可用率を直近の 5 週間のベースライン可用率と比較する、ロケーション別の選択したシンセティック・トランザクションの可用率の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、ロケーション別の先週および直近の 5 週間の平均応答時間および可用率の表
- 選択したシンセティック・トランザクションのサブトランザクションの、先週および直近の 5 週間の平均応答時間の結合折れ線グラフ
- 先週の可用率を直近の 5 週間のベースライン可用率と比較する、選択したシンセティック・トランザクションのサブトランザクションの平均可用率の結合折れ線グラフ
- 選択したシンセティック・トランザクションのサブトランザクションの、先週および直近の 5 週間の平均応答時間および可用率の表

ページ 2 には、以下のようなシンセティック・トランザクションの HTTP メトリックに関するトレンド・データが表示されます。

- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均ブロッキング時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均 DNS 時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均 SSL 時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均接続時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均送信時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均受信時間の結合折れ線グラフ
- 選択したシンセティック・トランザクションの、先週および直近の 5 週間の平均レンダリング時間の結合折れ線グラフ
- シンセティック・トランザクションの、先週および直近の 5 週間の平均 HTTP メトリックの表

ページ 3 には、ロケーション別のシンセティック・トランザクションの HTTP メトリック (ミリ秒単位) に関するトレンド・データが表示されます。グラフおよび表では、以下のように、先週の HTTP メトリックの平均を、直近の 5 週間のベースライン・メトリック平均と比較します。

- 選択されたシンセティック・トランザクションの、先週のさまざまな平均 HTTP メトリックを直近の 5 週間のベースライン・メトリックとロケーション別で比較する 7 つの結合折れ線グラフ。
- 選択されたシンセティック・トランザクションの、ロケーション別の先週および直近の 5 週間の平均 HTTP メトリックの表

ページ 4 には、サブトランザクションの HTTP メトリック (ミリ秒単位) に関するトレンド・データが表示されます。グラフおよび表では、以下のように、先週の HTTP メトリックの平均を、直近の 5 週間のベースライン・メトリック平均と比較します。

- 選択されたシンセティック・トランザクションの、先週のさまざまな HTTP メトリックの平均値を直近の 5 週間のベースライン・メトリックとサブトランザクション別で比較する 7 つの結合折れ線グラフ
- 選択されたシンセティック・トランザクションの、サブトランザクション別の先週および直近の 5 週間の HTTP メトリックの平均値の表

サブトランザクションのトレンド

この 2 ページから成るレポートには、サブトランザクションの先週および直近の 5 週間の応答時間、可用率、および HTTP メトリックのトレンド分析が表示されます。

ページ 1 には、以下のような、サブトランザクションの直近の日曜日、先週、および直前の 5 週間の応答時間および可用率に関するトレンド・データが表示されます。

- 直近の日曜日、先週、および直前の 5 週間のサブトランザクションの応答時間および可用率を比較する表
- サブトランザクションの先週の平均応答時間を直近の 5 週間のベースライン応答時間と比較する結合折れ線グラフ
- サブトランザクションの先週の平均可用率を直近の 5 週間のベースライン可用率と比較する結合折れ線グラフ
- サブトランザクションの先週の平均応答時間および可用率の表
- サブトランザクションの直近の 5 週間の平均応答時間および可用率の表

ページ 2 には、サブトランザクションの HTTP メトリック (ミリ秒単位) に関するトレンド・データが表示されます。グラフおよび表では、以下のように、サブトランザクションの先週の HTTP メトリック平均を、直近の 5 週間のベースライン・メトリック平均と比較します。

- 選択されたトランザクションの、直近の日曜日、先週、および直近の 5 週間のサブトランザクション別の HTTP メトリック平均値の 7 つの表
- 選択されたトランザクションの、先週のさまざまな平均 HTTP メトリック (ミリ秒単位) を直近の 5 週間のベースライン可用率とサブトランザクション別で比較する 7 つの結合折れ線グラフ
- サブトランザクションの先週の HTTP メトリックの平均値の表
- サブトランザクションの直近の 5 週間の HTTP メトリックの平均値の表

手順

レポートを生成するには、次の手順を実行します。

1. 「**パフォーマンス**」アイコンをクリックし、「**Application Performance ダッシュボード**」を選択します。アプリケーションを選択するには、「**すべてのマイ・アプリケーション**」を展開し、アプリケーションを選択します。選択されたアプリケーションに関連付けられているすべてのシンセティック・トランザクションを表示するには、「**グループ**」 > 「**トランザクション**」 > 「**シンセティック・トランザクション**」をクリックします。
2. 「トランザクション・リスト」テーブルからシンセティック・トランザクションを選択します。レポートを実行するには、「**アクション**」 > 「**レポートの起動**」をクリックし、以下のいずれかのレポートを選択します。
   - **トランザクション全体 (Transactions Overall)**
   - **ロケーション別のトランザクション詳細 (Transaction Detail By Locations)**
   - **サブトランザクション別のトランザクション詳細 (Transaction Detail By Subtransactions)**
   - **トランザクションのトレンド**
   - **サブトランザクションのトレンド**

   Web ブラウザーで、構成ページが新しいタブで開きます。
3. レポートの日付範囲を設定するには、事前定義された日付範囲を選択するか、カスタム日付範囲を入力します。
4. レポートの時間間隔を設定するには、「**時間タイプ (Time Type)**」から間隔を選択します。設定された日付範囲中の「**毎時**」、「**日次**」、または「**週次**」の間隔でシンセティック・トランザクションおよびサブトランザクションのデータを表示するようにレポートを設定します。レポートを生成するには、「**完了**」をクリックします。
5. トランザクション、サブトランザクション、またはロケーションの HTTP メトリックに関するレポートを表示するには、「**トランザクション全体**」レポート、「**ロケーション別のトランザクション詳細**」レポート、または「**サブトランザクション別のトランザクションの詳細**」レポートでトランザクション、サブトランザクション、またはロケーションを選択する必要があります。

- 「**トランザクション別のタイムリーな分析**」を表示するには、「**トランザクション全体**」レポートでトランザクション名を右クリックし、「**移動**」 > 「**時刻別の HTTP メトリック分析**」を選択します。
- 「**トランザクション別の HTTP メトリック**」を表示するには、「**トランザクション全体**」レポートでトランザクション名を右クリックし、「**移動**」 > 「**HTTP メトリックの集約**」を選択します。
- 「**トランザクションのロケーション別のタイムリーな分析**」を表示するには、「**ロケーション別のトランザクションの詳細**」レポートでトランザクション名を右クリックし、「**移動**」 > 「**時刻別の HTTP メトリック分析**」を選択します。
- 「**トランザクションのロケーション別の HTTP メトリック**」を表示するには、「**ロケーション別のトランザクションの詳細**」レポートでトランザクション名を右クリックし、「**移動**」 > 「**HTTP メトリックの集約**」を選択します。
- 「**サブトランザクションのロケーション別のタイムリーな分析**」を表示するには、「**ロケーション別のトランザクションの詳細**」レポートでサブトランザクション名を右クリックし、「**移動**」 > 「**時刻別の HTTP メトリック分析**」を選択します。
- 「**サブトランザクションのロケーション別の HTTP メトリック**」を表示するには、「**ロケーション別のトランザクションの詳細**」レポートでサブトランザクション名を右クリックし、「**移動**」 > 「**HTTP メトリックの集約**」を選択します。
- 「**サブトランザクション別のタイムリーな分析**」を表示するには、「**サブトランザクションのトレンド**」レポートでサブトランザクション名を右クリックし、「**移動**」 > 「**時刻別の HTTP メトリック分析**」を選択します。
- 「**サブトランザクション別の HTTP メトリック**」を表示するには、「**サブトランザクションのトレンド**」レポートでサブトランザクション名を右クリックし、「**移動**」 > 「**HTTP メトリックの集約**」を選択します。

WebSphere Applications エージェント レポート

WebSphere Applications エージェントによって収集されるデータについて、事前定義のレポートが使用可能です。

レポートのためのデータは WAREHOUS という Db2 データベースに格納されます。レポートには、1 時間ごと、1 日ごと、1 週間ごと、および 1 カ月ごとのデータが表示されます。これらのデータは、それぞれ 1 カ月間、3 カ月間、1 年間、および 1 年間保存されます。Cloud APM に、これらの保存期間を変更するためのスクリプトや手順は用意されていません。WebSphere Applications エージェントによって収集されるデータについて、以下のレポートが使用可能です。

アプリケーション要求パフォーマンス

説明

このレポートでは、アプリケーション・サーバー全体の集約レベルでアプリケーションがどのように実行されているかを分析します。円グラフは、アプリケーションに対する集約レベルの要求を示しています。棒グラフは、集約レベルでのアプリケーションの平均応答時間を示しています。2 つの時系列折れ線グラフは、すべてのアプリケーションの平均応答時間と合計要求カウントの傾向を示しています。アプリケーションに対する個々の要求にドリルダウンするには、円のスライスまたは棒をクリックします。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー・タイプ

使用されるテーブル

Request_Analysis_*V

DB 接続プール

説明

このレポートでは、アプリケーション・サーバー内のデータベース接続プールを分析します。表には、集約レベルでのすべての接続プールの主要な統計が示されています。特定のデータ・ソースを選択すると、2 つの傾向グラフに主要な統計の傾向が示されます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー名

使用されるテーブル

DB_Connection_Pools_*V

EJB パフォーマンス

説明

このレポートでは、アプリケーション・サーバーにデプロイされた EJB がどのように実行されているかを分析します。 円グラフには、EJB の集約レベルでのメソッド・カウントが示されます。 棒グラフは、集約レベルでの EJB 全体の平均メソッド応答時間を示しています。2 つの時系列グラフには、すべての EJB についてのメソッド呼び出しカウントの傾向と平均メソッド応答時間の傾向が示されます。リストで行をクリックすると、傾向線を EJB でフィルタリングできます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー名

使用されるテーブル

Enterprise_Java_Beans_*V

アプリケーション・サーバーの GC 使用量

説明

このレポートでは、ガーベッジ・コレクションを分析します。このレポートを使用して、ガーベッジ・コレクションにより問題が発生しているかどうか、またはヒープ・サイズが不適切であるかどうかを判別します。最初のグラフは、平均ヒープ使用率と、ガーベッジ・コレクションの平均リアルタイム・パーセントを経時的に示します。2 つ目のグラフは、実行されたガーベッジ・コレクションの平均リアルタイム・パーセントおよび平均ガーベッジ・コレクション率を示します。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー・タイプ

使用されるテーブル

Garbage_Collection_Analysis_*V

アプリケーション・サーバーの JVM 使用量

説明

このレポートでは、アプリケーション・サーバーの JVM がどのように実行されているかを分析します。積み重ね棒グラフに、JVM メモリーの使用状況および空き状況が示されます。二重折れ線グラフには、JVM メモリー使用量と対比した JVM CPU 消費量が示されます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー・タイプ

使用されるテーブル

Application_Server_*V

スレッド・プール

説明

このレポートでは、アプリケーション・サーバー内のスレッド・プールを分析します。表には、集約レベルでのすべてのスレッド・プールの主要な統計が示されています。リストからスレッド・プールを選

択すると、傾向グラフに、選択したスレッド・プールの主要な統計の傾向が示されます。 スレッド・プールを選択しない場合、傾向には、すべてのスレッド・プールの要約が示されます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー・タイプ

使用されるテーブル

Thread_Pools_*V

Web アプリケーション・パフォーマンス

説明

このレポートでは、アプリケーション・サーバーの Web コンテナーでアプリケーションがどのように実行されているかを分析します (PMI データ)。円グラフは、アプリケーションに対する集約レベルの要求を示しています。棒グラフは、集約レベルでのアプリケーションの平均応答時間を示しています。2 つの時系列折れ線グラフは、すべてのアプリケーションの平均応答時間と合計要求カウントの傾向を示しています。円のスライス、棒、または線をクリックすると、そのアプリケーションの個々のサーブレット/JSP にドリルダウンできます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション・サーバー・タイプ

使用されるテーブル

Thread_Pools_*V

クラスターのアプリケーション要求パフォーマンス

説明

このレポートでは、クラスター内のサーバーがどのように実行されているかを分析します。 最初のグラフには、選択された時間間隔内に各クラスター・メンバーが完了した要求の数が示されます。2 つ目のグラフには、各クラスター・メンバーの平均応答時間の傾向に関する情報が示されます。このグラフでは、クラスター内のサーバーごとに個別の線が示されます。線をクリックすると、個々のサーバー・データにドリルダウンできます。当該サーバーの「アプリケーション要求パフォーマンス」レポートが開きます。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびクラスター名

使用されるテーブル

Request_Analysis_*V

クラスターの JVM および GC の使用量

説明

このレポートでは、各クラスター・メンバーの JVM およびガーベッジ・コレクションの使用状況の傾向を分析します。最初のグラフは、実行されたガーベッジ・コレクションの平均リアルタイム・パーセントを示します。2 つ目のグラフは、平均ヒープ使用率を示します。最後のグラフは、CPU および JVM メモリーの使用量を示します。これらのすべてのグラフは、各クラスター・メンバーのデータを個別の線で示します。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびクラスター名

使用されるテーブル

Garbage_Collection_Analysis_*V、Application_server_*V

サーバー間で応答時間が最も遅い上位アプリケーション

説明

このレポートでは、すべてのアプリケーション・サーバーにわたって集約レベルでアプリケーションがどのように実行されているかを分析します。棒グラフは、集約レベルでのアプリケーションの 平均応答時間を示しています。

パラメーター

日付範囲: いずれかの事前定義レポート期間を選択するか、カレンダーから正確な開始時間および終了時間を選択します。

必須パラメーター: 要約タイプおよびアプリケーション 数

第 11 章 アップグレード

ご使用の エージェントとデータ・コレクターをアップグレードして、現行リリースで使用できる最新のフィーチャーおよび機能を入手します。

エージェントのアップグレード

アップグレードされたモニター・エージェントを含む新規アーカイブ・ファイルが、定期的にダウンロードできるようになります。アーカイブ・ファイルは、IBM Marketplace Web サイトの 製品およびサービス から入手できます。

始める前に

以下のエージェントの場合、アップグレード手順を実行する前にエージェント固有のタスクを完了する必要があります。

- エージェントを AIX 上で非 root ユーザーとして実行している場合、エージェントをアップグレードするインストール手順を開始する前に、メモリーからライブラリーの 1 つを消去する必要があります。1142 ページの『AIX 上のエージェント: アップグレード前のエージェントの停止と slibclean の実行』の説明に従ってください。
- AIX 上の HMC Base エージェントで root 以外のユーザーとしてエージェントをアップグレードする 場合は、まず HMC Base エージェントを停止して、依存関係ライブラリーをキャッシュから消去する必要があります。1143 ページの『AIX 上の HMC Base エージェント: アップグレード前の root 以外のユーザーとしてのエージェントの停止および slibclean の実行』の説明に従ってください。
- Microsoft .NET エージェントの場合、エージェントをアップグレードする 前に、.NET アプリケーションからデータ・コレクターを削除する必要があります。 1146 ページの『Microsoft .NET エージェント: アップグレード前に .NET データ・コレクターを削除する』の説明に従ってください。
- Node.js エージェントの場合、エージェントをアップグレードする 前に、Node.js アプリケーションからデータ・コレクター・プラグインを削除する必要があります。1143 ページの『Node.js エージェント: アップグレード前にデータ・コレクター・プラグインを削除する』の説明に従ってください。
- Ruby エージェントの場合、エージェントをアップグレードする 前に、Ruby アプリケーションからデータ・コレクターを削除する必要があります。1147 ページの『Ruby エージェント: アップグレード前にデータ・コレクター・プラグインを削除する』の説明に従ってください。
- HTTP Server エージェントの場合、エージェントをアップグレードする 前に HTTP サーバーを停止する必要があります。
- WebSphere MQ エージェントの場合、以前のリリースでエージェントのトランザクション・トラッキングを有効化している場合は、エージェントをアップグレードする 前にエージェント・インスタンスを停止する必要があります。
- SAP NetWeaver Java Stack エージェント の場合、V8.1.3.2 から V8.1.4 にアップグレードするには、エージェントをアップグレードする 前にデータ・コレクターで構成されたすべての SAP NetWeaver Java Stack インスタンスを停止する必要があります。
- Skype for Business Server エージェントの場合、古いバージョンから V8.1.4.0.2 にアップグレードすると、エージェント側で、エージェント名が Skype for Business Server に変更されます。 また、SDA を介したサポートのアップグレード後に、APMUI サービスを再始動して、新しいエージェント名 (Skype for Business Server) を MIN サーバー・サイドで反映する必要があります。そうしないと、MIN サーバーのダッシュボードに古いエージェント名 (MS Lync Server) が表示されます。
- Tomcat エージェントの場合、Windows で TEMA コア・フレームワークをアップグレードするときは、エージェントとサーバーの両方を停止する必要があります。『Tomcat エージェント: Windows での TEMA コア・フレームワークのアップグレード』の説明に従ってください。

© Copyright IBM Corp. 2014, 2019

このタスクについて

新規バージョンのエージェントが使用可能な場合は、インストール・スクリプトを実行すると自動的にエージェントがアップグレードされます。エージェントの新規バージョンが存在しない場合は、エージェントが既にインストールされているという旨のメッセージが表示され、インストール済みのエージェントには何も影響がありません。

アップグレードされたエージェントをインストールするには、以下の手順を使用します。

手順

- 124 ページの『UNIX システムでのエージェントのインストール』
- 130 ページの『Linux システムでのエージェントのインストール』
- 139 ページの『Windows システムでのエージェントのインストール』

タスクの結果

エージェントが最新のバージョンにアップグレードされます。新しいバージョンのモニター・エージェントが存在しない場合は、 エージェントが既にインストールされているというメッセージが表示され、インストール済みのエージェントは影響を受けません。

次のタスク

Windows エージェントをアップグレードした後、Windows インストーラーによる自動構成も自動開始も行われないエージェントをすべて再始動する必要があります。次のコマンドを実行してエージェントの状況を検査します。

```
./name-agent.bat status
```

エージェントを開始するには、以下のいずれかの方法に従ってください。

- 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM Monitoring エージェント (IBM Monitoring agents)**」 > 「**IBM Performance Management**」をクリックします。エージェントを右クリックしてから、「**スタート**」をクリックします。
- 次のコマンドを実行します。

  ```
  ./name-agent.bat start
  ```

モニター・エージェント・コマンドの詳細 (使用する名前を含む) やエージェントの状況の検査方法などについて詳しくは、181 ページの『エージェント・コマンドの使用』を参照してください。自動で開始されるエージェントおよび手動で開始されるエージェントについて詳しくは、115 ページの『第 5 章 エージェント とデータ・コレクターのデプロイメント』を参照してください。

- Hadoop エージェントの場合、ソケット・ベースのエージェント (8.1.2、フィックスパック 2 以前) から REST API ベース・エージェント (8.1.3 以降) へのアップグレード後に、次の手順を実行します。
  1. 不要なログの生成を防ぐには、すべての Hadoop ノードの `hadoop-metrics2.properties` ファイルから 17 行のコードを削除します。
  2. Hadoop サービスを停止します。
  3. エージェント・インストーラーからコピーされた `Plugin.jar` ファイルを、Hadoop クラスター内のすべてのノードから削除します。
  4. Hadoop サービスを開始します。

  17 行のコードと `Plugin.jar` ファイルについては、「Hadoop ノードの構成」を参照してください。
- HMC Base エージェントの場合は、エージェントをバージョン 6.2.2.6 から 6.2.2.7 にアップグレードした後、再度エージェントを構成してエージェントを再始動する必要があります。説明については、269 ページの『HMC Base モニターの構成』を参照してください。
- HTTP Server エージェントでは、1.0.0.4 より前のバージョンから 1.0.0.4 以降にエージェントをアップグレードした場合、HTTP サーバーによって使用されている `.conf` ファイルを更新し、以前のデータ・コレクター構成ファイルを新たに生成されたファイルに置き換える必要もあります。新しいエージェン

ト・インスタンスをコンソールに追加する必要もあります。説明については、275 ページの『HTTP Server モニターの構成』を参照してください。

- Microsoft .NET エージェントの場合は、 エージェントをアップグレードしてからデータ・コレクターを構成します。説明については、527 ページの『データ・コレクターの登録』を参照してください。

  エージェントを新しいディレクトリーにインストールした場合、サービス・コントローラー (sc) コマンドを使用して、Profiler サービスの bin パスを変更する必要があります。例えば、次の場合があります。

  ```
  sc ¥¥localhost config DotNetProfilerService binPath=
  "${install_dir}¥qe¥bin¥DotNetProfilerService.exe
  ```

  ここで、*install_dir* は新しいインストール・ディレクトリーです。
- Node.js エージェントの場合は、エージェントをアップグレードしてからエージェント ・データ・コレクターを構成します。説明については、593 ページの『Node.js エージェントの構成』を参照してください。
- OpenStack エージェント の場合、エージェントが OpenStack identity API v3 を使用するように追加構成を行うには、すべてのエージェント・インスタンスを再構成し、エージェント・データ・コレクター構成ファイルを更新します。説明については、1146 ページの『OpenStack エージェント: OpenStack Identity API v3 を使用するためのエージェント・インスタンスの再構成』を参照してください。
- Ruby エージェントの場合は、 エージェントをアップグレードしてからデータ・コレクターを構成します。説明については、722 ページの『診断データ・コレクターの構成』を参照してください。
- WebSphere Applications エージェントの場合は、エージェントをアップグレードした後、 新しいバージョンのエージェントのインストール・ディレクトリーからコマンド *dc_home*/bin/migrate.sh/bat を実行してデータ・コレクターをマイグレーションし、 アプリケーション・サーバー・インスタンスを再始動します。説明については、1147 ページの『WebSphere Applications エージェント: データ・コレクターのマイグレーション』を参照してください。
- Linux /opt/ibm/ccm/agent ディレクトリーにインストールされている 古いバージョンのエージェントをアップグレードする 場合、Linux システムで以下のステップを実行する必要があります。

  1. 古いインストール・ディレクトリー /opt/ibm/ccm/agent から新しいインストール・ディレクトリー (例えば、/opt/ibm/apm/agent) にエージェント構成をマイグレーションすることを確認した場合、新しいインストール・ロケーションでエージェントを開始する必要があります。

     制約事項 : 前のバージョンのエージェントは、前のインストール・ロケーションで自動的に停止されますが、新規インストール・ロケーションで自動的に開始されることはありません。
  2. 新規インストール・ディレクトリーでエージェントが 機能することを確認したら、前のバージョンのエージェントを /opt/ibm/ccm/agent ディレクトリーからアンインストールする 必要があります。すべてのエージェントを削除する場合は、**/opt/ibm/ccm/agent/bin/smai-agent.sh uninstall_all** コマンドを実行します。
- Linux FP6 以前のエージェントをアップグレードする 場合は、エージェントを新しいディレクトリーにアップグレードし、エージェントの構成または再構成を行った後に、古いインストール・ディレクトリーを削除することをお勧めします。以下の手順を実行します。

  1. モニター・エージェントがインストールされている VM またはシステムで、コマンド行を開始し、 古いインストール・ディレクトリーのバイナリー・フォルダーに移動します (/opt/ibm/ccm/agent/bin)。
  2. インストールされているすべてのモニター・エージェントを古いインストール・ディレクトリーからアンインストールするために、./smai-agent.sh uninstall_all と入力します。
  3. 古いインストール・ディレクトリーを削除します。

エージェント構成変更の保持

上級者は、コンポーネントのカスタマイズにオーバーライド 値を適用できます。オーバーライド値を適用すると、アップグレード中、値が確実に保持されます。 変更をグローバルに適用する前に、ご使用の環境で変更をテストしてください。

このタスクについて

- 以下の説明は Linux および AIX エージェント向けのものです。エージェント製品コードのリストおよびエージェントの停止および開始のためのコマンドについては、181 ページの『エージェント・コマンドの使用』を参照してください。
- Windows エージェント・プロセスは、構成の変更を保持するように設計されています。k*pc*cma.ini ファイル (*pc* は製品コード) 内で更新された変数は、「ローカル設定のオーバーライド (Override Local Settings)」セクション内に維持されます。これらの変数は、それぞれの構成時に、エージェントが実行時に使用する Windows レジストリー項目を更新するために使用されます。
- .*pc*.environment ファイルおよび .global.environment ファイル内のカスタマイズされた設定は、エージェントのアップグレード後に失われます。設定した値を保持するには、*pc*.environment ファイルおよび global.environment ファイルでカスタマイズ変更を行います。これらのファイル内の設定値は、エージェントのアップグレードによって上書きされません。

手順

環境ファイルに対する構成変更を保存し、エージェントのアップグレード後もそれらを保持するには、以下の手順を実行します。

1. 必要に応じて、以下のファイルを作成または更新します (*install_dir* は、エージェントのインストール・ディレクトリー (Linux のデフォルト: /opt/ibm/apm/agent/、AIX のデフォルト: /opt/ibm/ccm/agent/ など) です)。

ファイル名	説明
***install_dir*/config/*pc*.environment**	ファイル名内の *pc* は、エージェント製品コード (mq や rz) です。
***install_dir*/config/global.environment**	すべてのエージェント・タイプに適用する変更についてグローバル環境ファイルを更新します。

   例えば、as.environment は永続的な WebSphere Applications エージェントの環境ファイルです。エージェントが新規バージョンにアップグレードされると as.environment は上書きされます。

   変数は *key*=*value* の形式で定義します。ここで、*key* は環境変数名、*value* は値または設定です (例: **KDC_FAMILIES=${KDC_FAMILIES}HTTP:10001**)。
2. 変数設定の更新が終了したら、環境ファイルを保存して閉じ、影響を受けるエージェントを再始動します。

タスクの結果

更新が、同じタイプのすべてのエージェントに適用されるか、またはグローバル環境ファイルを更新した場合は Cloud APM サーバーにレポートするすべてのエージェントに適用されます。変更内容はエージェントのバージョンアップにまたがって保持されます。

AIX 上のエージェント: アップグレード前のエージェントの停止と slibclean の実行

AIX システムで非 root ユーザーとしてエージェントをアップグレードする 場合は、このタスクを実行する必要があります。エージェント・インストーラーを実行する前に、エージェントを停止し、**slibclean** を実行して、libkududp.a ライブラリーをクリアする必要があります。

手順

1. エージェントが複数のインスタンスをサポートするかどうかに応じて、次のいずれかのコマンドを実行してエージェントを停止します。
   - ./*name*-agent.sh stop
   - ./*name*-agent.sh stop *instance_name*

   181 ページの『エージェント・コマンドの使用』を参照してください。

2. root ユーザー特権で、以下のコマンドを実行します。

```
slibclean
```

IBM Knowledge Center で「slibclean コマンド」を参照してください。

タスクの結果
エージェントが停止し、libkududp.a ライブラリーがクリアされます。

次のタスク
エージェントのインストーラーを実行して、そのエージェントをダウンロードしたエージェント・リリースにアップグレードします。123 ページの『第 6 章 エージェントのインストール』を参照してください。
アップグレードが失敗した場合は、サーバーをリブートし、手順を繰り返してください。

AIX 上の HMC Base エージェント: アップグレード前の root 以外のユーザーとしてのエージェントの停止および slibclean の実行

AIX で root 以外のユーザーとして HMC Base エージェントをアップグレードする 前に、HMC Base エージェントを停止して **slibclean** を実行し、依存関係ライブラリーをキャッシュから消去する必要があります。

このタスクについて

手順

1. root 以外のユーザーとして以下のコマンドを実行し、エージェントを停止します。

```
hmc_base-agent.sh stop
```

2. root ユーザー特権で、以下のコマンドを実行します。

```
slibclean
```

IBM Knowledge Center で「slibclean コマンド」を参照してください。

タスクの結果
HMC Base エージェントが停止し、依存関係にあるライブラリーが消去されます。

次のタスク
エージェント・インストーラーを実行して HMC Base エージェントをアップグレードします。

Node.js エージェント: アップグレード前にデータ・コレクター・プラグインを削除する

Node.js エージェント をアップグレードする 前に、Node.js アプリケーションからモニター・プラグインを削除する必要があります。

このタスクについて
Node.js エージェント・バージョンに応じて、Node.js アプリケーションからモニター・プラグインを削除するために異なる手順を実行する必要があります。エージェント・バージョンを確認するには、『エージェント・バージョン・コマンド』を参照してください。

手順

1. Node.js アプリケーション・ファイルの先頭からデータ・コレクター・プラグインを削除します。
   - Node.js エージェントを V01.00.12.00 から V01.00.13.00 にアップグレードする 場合は、以下の手順を実行します。

– リソース・データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_index.js');
```

ここで、*KNJ_NPM_LIB_LOCATION* は、npm パッケージのグローバル・インストール・ディレクトリーの lib フォルダーへのディレクトリーです。デフォルトのディレクトリーは、/usr/local/lib です。

– リソース・データの収集および詳細診断データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_deepdive.js');
```

– リソース・データの収集、詳細診断データの収集、およびメソッド・トレースの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_methodtrace.js');
```

- Node.js エージェントを V01.00.10.00 から V01.00.13.00 にアップグレードする 場合は、以下の手順を実行します。
  – リソース・データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('install_dir/lx8266/nj/bin/plugin/knj_index.js');
```

ここで、*install_dir* は Node.js エージェント のインストール・ディレクトリーです。

– リソース・データの収集および詳細診断データの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('install_dir/lx8266/nj/bin/plugin/knj_deepdive.js');
```

– リソース・データの収集、詳細診断データの収集、およびメソッド・トレースの収集を有効にした場合は、Node.js アプリケーション・ファイルの先頭から以下の行を削除します。

```
require('install_dir/lx8266/nj/bin/plugin/knj_methodtrace.js');
```

2. Node.js アプリケーションを再始動して、データ・コレクター・プラグインを無効にします。
   - 現在の Node.js エージェントのバージョンが V01.00.10.00 の場合、ここまでで、データ・コレクター・プラグインが正常に削除されました。
   - 現在の Node.js エージェントのバージョンが V01.00.12.00 の場合、次のステップに進みます。
3. *install_dir*/lx8266/nj/bin ディレクトリーから ./uninstall.sh コマンドを実行して、以前のエージェント設定を削除します。

次のタスク

Node.js エージェント をアップグレードします。 1139 ページの『エージェントのアップグレード』を参照してください。

Response Time Monitoring エージェント: IBM HTTP Server 応答時間モジュールのアップグレード

以前 IBM HTTP Server をモニターするために IBM HTTP Server 応答時間モジュールまたは HTTP Server エージェントを使用していた場合は、インストール済み環境をアップグレードしてください。

このタスクについて

以下の表に、ユーザーによる IBM HTTP Server のモニター方法に類似している可能性のあるインストール・シナリオをいくつか示します。

Response Time Monitoring エージェント	IBM HTTP Server 応答時間モジュールを使用するか?	Packet Analyzer を使用するか?	HTTP Server エージェントがインストールされるか?
AIX および xLinux: 08.11.00 以降 Windows: 08.14.02 以降	✓	–	✓
AIX および xLinux: 08.10.00	✓	–	–
AIX および xLinux: 08.10.00	✓	–	✓
7.40.07 以前	–	✓	–
7.40.07 以前	–	✓	✓

これらすべてのシナリオで、インストール・プロセスは類似しています。

手順

1. AIX または Linux ではリリース V8.1.1 以降、Windows では V8.1.4.02 以降の HTTP Server エージェントをインストールします。

   IBM HTTP Server 応答時間モジュールは、自動的にエージェントと一緒にインストールされます。
2. HTTP Server エージェントを構成します。

   注: 以前に IBM HTTP Server 応答時間モジュールを使用していた場合は、Web サーバー構成ファイル (`httpd.conf`) を新しい IBM HTTP Server 応答時間モジュールの場所で更新し、以前のロード・モジュール構成ファイル (`mod_wrt.so`) を削除します。

   注: Response Time Monitoring エージェント V8.1.1 以降では、以前のリリースのロード・モジュール・ファイル (`mod_wrt.so`) は機能しません。古いバージョンのこのファイルの使用を試みると、エラー・ログ・メッセージが作成されます。 この場合もトランザクションのトラッキングは可能ですが、トランザクション・インスタンス・データは表示されません。

   詳しくは、http://ibm.biz/agent-httpserver からダウンロードできる HTTP Server エージェントのリファレンス PDF を参照してください。
3. IBM HTTP Server および HTTP Server エージェントが実行中であることを確認します。 Response Time Monitoring インストーラーが HTTP Server エージェントを検出すると、Response Time Monitoring エージェントは Packet Analyzer ではなく IBM HTTP Server 応答時間モジュールを有効にします。
4. Response Time Monitoring エージェントを、HTTP Server エージェントと同じ場所 `AGENT_HOME` にインストールします。

   - Linux AIX **root** として V8.1.1 以降をインストールします。 `AGENT_HOME` (例: `/opt/ibm/apm/agent/`)
   - Windows 管理者権限を使用して V8.1.4.0.2 以降をインストールします。 `AGENT_HOME` (例: `C:¥IBM¥APM¥`)
5. 以前のリリースで Packet Analyzer を使用した場合、IBM HTTP Server 応答時間モジュールによる IBM HTTP Server のモニターを開始するために、Packet Analyzer を無効にする必要が生じる場合があります。
6. IBM HTTP Server を再始動します。

Microsoft .NET エージェント: アップグレード前に .NET データ・コレクターを削除する

Microsoft .NET エージェントをアップグレードする 前に、.NET アプリケーションから .NET データ・コレクターを削除する必要があります。

手順

1. データ・コレクターのすべてのモジュールを登録抹消します。
   管理者として、以下を入力します。

   ```
   cd install_dir¥qe¥bin configdc unregisterdc all
   ```

   ここで、*install_dir* は Microsoft .NET エージェント のインストール・ディレクトリーです。
2. .NET アプリケーションを再始動します。

次のタスク

Microsoft .NET エージェント をアップグレードします。 1139 ページの『エージェントのアップグレード』を参照してください。

OpenStack エージェント: OpenStack Identity API v3 を使用するためのエージェント・インスタンスの再構成

OpenStack Identity API v3 を使用するように OpenStack エージェントをアップグレードするには、最新バージョンのエージェントをインストールした後に、すべてのエージェント・インスタンスを再構成し、データ・コレクター構成ファイルを更新する必要があります。

このタスクについて

このタスクは、OpenStack Identity API v3 を使用するようにエージェントをアップグレードする 場合にのみ必須です。

手順

1. 既存のエージェント・インスタンスをすべて再構成します。説明については、613 ページの『OpenStack エージェントの構成』を参照してください。
2. エージェントのデータ・コレクター構成ファイルである ksg_dc_*instance_name*.cfg を特定します (*instance_name* はエージェント・インスタンスに指定した名前)。
   ファイルが存在しない場合は、*install_dir*/lx8266/sg/bin/ksg_dc.cfg を *install_dir*/config ディレクトリーにコピーして、ファイル名を ksg_dc_*instance_name*.cfg に変更します。
   例えば、インスタンス名が OS1 の場合は、名前を ksg_dc_OS1.cfg に変更します。
3. ksg_dc_*instance_name*.cfg ファイルに以下のセクションを追加します。

   ```
   #OpenStack authentication information
   [OS_authentication_info]
   OS_project_domain_name=Default
   OS_user_domain_name=Default
   OS_cert_path=
   ```

4. 以下のコマンドを実行して、エージェント・インスタンスを再始動します。

   ```
   install_dir/bin/openstack-agent.sh stop instance_name
   install_dir/bin/openstack-agent.sh start instance_name
   ```

   ここで、*instance_name* は、構成するエージェント・インスタンスの名前です。

Ruby エージェント: アップグレード前にデータ・コレクター・プラグインを削除する

Ruby エージェント をアップグレードする 前に、Ruby アプリケーションからモニター・プラグインを削除する必要があります。

手順

1. 以下のコマンドを実行することによって、古いバージョンのデータ・コレクターを削除します。

```
gem uninstall stacktracer
```

2. アプリケーションのホーム・ディレクトリーにナビゲートし、その Gemfile を開いて、gem 'stacktracer', '*version*' という行を削除します。
   ここで、*version* は Ruby エージェントのバージョン番号です。
3. アプリケーションのホーム・ディレクトリーで bundle install と入力します。

次のタスク

Ruby エージェント をアップグレードします。 1139 ページの『エージェントのアップグレード』を参照してください。

WebSphere Applications エージェント: データ・コレクターのマイグレーション

エージェントを更新した後で、データ・コレクターを対話式またはサイレント・モードでマイグレーションする必要があります。

対話式でのデータ・コレクターのマイグレーション

マイグレーション・ユーティリティーを 使用して、以前のメンテナンス・レベルのデータ・コレクターを対話式にマイグレーションすることができます。

始める前に

Linux AIX 非 root ユーザー・アカウントを使用して WebSphere Application Server または WebSphere Portal Server をインストールした場合は、構成ユーティリティーを 実行する前に、*install_dir*/yndchome/7.3.0.14.08 内の以下のエージェント・ディレクトリーに対する読み取り特権と書き込み特権がその非 root ユーザー・アカウントに付与されていることを確認してください。ここで、*install_dir* は、WebSphere Applications エージェントのインストール・ディレクトリーです。

- data
- bin
- runtime
- logs

必要であれば、chmod 777 コマンドを使用して読み取りおよび書き込み特権を与えます。また、アプリケーション・サーバーをインストールする際に使用したユーザーでログインします。

このタスクについて

マイグレーション・ユーティリティーを 使用して、以前のメンテナンス・レベルのデータ・コレクターを対話式にマイグレーションすることができます。多数のアプリケーション・サーバー・インスタンスをマイグレーションするときは、マイグレーション・ユーティリティーをサイレント ・モードで使用したほうが便利な場合があります。

重要 :

- 以前のメンテナンス・レベルでマイグレーションできるのは、7.3 レベルのデータ・コレクターのみです。データ・コレクターのバージョンは、データ・コレクターのホーム・ディレクトリーのパスに示されます。
- データ・コレクターをバージョン 7.3 から 7.3 フィックスパック 1 にマイグレーションすることはできません。マイグレーションする代わりに、データ・コレクターを構成解除してエージェント・バージョン

7.3 をアンインストールしてください。その後で、エージェント・バージョン 7.3 フィックスパック 1 をインストールしてデータ・コレクターを再び構成してください。

手順

1. Linux AIX アプリケーション・サーバーをインストールする際に使用したユーザーでログインします。
2. 最新バージョンのエージェントのインストール・ディレクトリーからマイグレーション・ユーティリティーを開始します。

   Linux AIX コマンド *dc_home*/bin/migrate.sh を実行します。

   Windows コマンド *dc_home*¥bin¥migrate.bat を実行します。
3. ユーティリティーに、ローカル・コンピューター・システム上で検出されたすべてのネットワーク・カードの IP アドレスが表示されます。

   使用する IP アドレスに対応する数値を入力します。
4. このユーティリティーは、古いメンテナンス・レベルのデータ・コレクターによって構成されたすべてのサーバーをディスカバーし、それらのリストを示します。データ・コレクターはメンテナンス・レベルごとにグループ化されます。

   リストから 1 つ以上のアプリケーション・サーバー・インスタンスを選択します。

   リストには従来の WebSphere Server インスタンスおよび Liberty サーバーの両方が含まれることがあります。従来の WebSphere Server インスタンスは、異なるプロファイル内に入っている可能性があります。

   ヒント:

   - 1 つのプロファイルの複数のインスタンスをモニターする場合は、これらをマイグレーション対象としてすべて同時に選択する必要があります。
   - Liberty プロファイル内のすべてのサーバーを同時にマイグレーションします。 構成されている複数のサーバーの一部のみをマイグレーションすると、安定性が損なわれる恐れがあります。

   要確認:

   - スタンドアロン環境の場合、アプリケーション・サーバー・インスタンスが実行中である必要があります。
   - ネットワーク・デプロイメント環境では、ノード・エージェントとデプロイメント・マネージャーは稼働している必要があります。
   - Liberty サーバーは、マイグレーション時に実行する必要はありません。
5. データ・コレクターをマイグレーションする 対象のアプリケーション・サーバー・インスタンスに対応する番号を入力するか、アスタリスク (*) を入力して、すべてのアプリケーション・サーバー・インスタンスのデータ・コレクターをマイグレーションします。

   サーバーのサブセットを指定するには、当該のサーバーを表す番号をコンマで区切って入力します。例えば、1,2,3 です。

   マイグレーション・ユーティリティーは、各データ・コレクターをモニター・エージェントと自動的に統合します。モニター・エージェントのホストの値およびポートの値は、既存の構成ファイルから検索されます。
6. 選択した各サーバーの別名を入力します。

   デフォルト値は既存のサーバー別名です。
7. Liberty サーバー・インスタンスの場合は、プロンプトが表示されたら JVM ホーム・ディレクトリーを入力します。例えば、/opt/IBM/java です。
8. このユーティリティーは、データ収集がマイグレーションされる各プロファイルについて、WebSphere グローバル・セキュリティーが有効かどうかを判別します。

   1 つ以上のプロファイルについて WebSphere グローバル・セキュリティーが有効である場合、クライアント・プロパティー・ファイルからセキュリティー設定を取得するかどうかを指定してください。

データ・コレクターは、 RMI または SOAP プロトコルを使用して WebSphere 管理サービスと通信します。プロファイルでグローバル・セキュリティーが有効な場合は、 そのプロファイルについて IBM WebSphere Application Server 管理コンソールにログインする権限があるユーザーのユーザー ID とパスワードを指定する必要があります。

あるいは、データ・コレクターを構成する前に、ユーザー名とパスワードを暗号化してアプリケーション・サーバーのクライアント・プロパティー・ファイルに保管することもできます。 `sas.client.props` ファイル (RMI 接続の場合) または `soap.client.props` ファイル (SOA 接続の場合) を使用する必要があります。

9. ユーティリティーが、該当するクライアント・プロパティー・ファイルからユーザー名とパスワードを取得できるようにするには 1 を入力し、ステップ 1149 ページの『11』 へスキップします。 そうしない場合は、2 を入力して、ユーザー名とパスワードを入力します。

   重要 : WebSphere Application Server 管理コンソールへのログインには、しばらく時間がかかることがあります。
10. WebSphere グローバル・セキュリティーが有効かどうかにかかわらず、プロファイルごとにユーザー名およびパスワードを入力します。
11. ユーティリティーは、選択されたアプリケーション・サーバー・インスタンスごとにデータ収集をマイグレーションします。そして、各サーバーのマイグレーションが正常に完了したかどうかを示す状況メッセージを表示します。
12. ユーティリティーの 指示に従って、インスタンスを再始動します。データ・コレクターの構成は、アプリケーション・サーバー・インスタンスが再始動されると有効になります。

タスクの結果
データ・コレクターが、インストールされている最新のメンテナンス・レベルにマイグレーションされます。

次のタスク
マイグレーション・ユーティリティーは、古いバージョンのデータ・コレクターで構成された設定を保持します。これらの設定を変更するには、新規データ・コレクターの *dc_home*¥bin ディレクトリーから構成ユーティリティーまたは 再構成ユーティリティーを 対話式またはサイレント・モードで実行することができます。詳しくは、840 ページの『フル構成ユーティリティーを 使用したデータ・コレクターの構成または再構成』を参照してください。

サイレント・モードでのデータ・コレクターのマイグレーション
マイグレーション・ユーティリティーをサイレント ・モードで使用して、以前のメンテナンス・レベルのデータ・コレクターをマイグレーションすることができます。

始める前に

Linux AIX 非 root ユーザー・アカウントを使用して WebSphere Application Server または WebSphere Portal Server をインストールした場合は、構成ユーティリティーを 実行する前に、*install_dir*/yndchome/7.3.0.14.08 内の以下のエージェント・ディレクトリーに対する読み取り特権と書き込み特権がその非 root ユーザー・アカウントに付与されていることを確認してください。ここで、*install_dir* は、WebSphere Applications エージェントのインストール・ディレクトリーです。

- `data`
- `bin`
- `runtime`
- `logs`

必要であれば、`chmod 777` コマンドを使用して読み取りおよび書き込み特権を与えます。また、アプリケーション・サーバーをインストールする際に使用したユーザーでログインします。

このタスクについて

サンプル・サイレント・プロパティー・ファイル sample_silent_migrate.txt が、マイグレーション・ユーティリティーのパッケージに 含まれています。このファイルは *install_dir*/yndchome/7.3.0.14.08/bin ディレクトリーに用意されています。

サイレント・プロパティー・ファイルの作成時には、次のような考慮事項に留意してください。

- ファイル中、番号記号 (#) で始まる行は、コメントとして扱われ、処理されません。行内の他の場所で番号記号を使用しても、それはコメントの開始とはみなされません。つまり、パスワードや他の用途に番号記号を使用することができます。
- 各プロパティーは、property = value のフォーマットで、別の行に記述されます。

  property
  これはプロパティーの名前です。構成できる有効なプロパティーのリストは、1150 ページの表 256 に示されています。サンプル・ファイルでは、この表にリストされていないプロパティーの変更や削除をしないでください。

  value
  これはプロパティーの値です。一部のプロパティーのデフォルト 値は、既に用意されています。 デフォルト値を削除して、プロパティー値をブランクまたは空のままにすることができます。 空の値は、デフォルト値を使用するのとは対照的に、プロパティーが指定されていないものとして扱われます。デフォルト値を使用する場合、ファイル内のプロパティーをコメント化することができます。
- パスワードはプレーン・テキストになります。
- プロパティーとその値においては、大/小文字が区別されます。

1150 ページの表 256 は、サイレント・モードでデータ・コレクターをマイグレーションする 場合に使用可能なプロパティーについて説明しています。

表 *256.* マイグレーション・ユーティリティーをサイレント・モードで実行する場合に使用可能なプロパティー

プロパティー	**コメント**
migrate.type	AD でなければなりません。
default.hostip	コンピューター・システムが複数の IP アドレスを使用している場合は、データ・コレクターが使用する IP アドレスを指定します。
itcam.migrate.home	古いメンテナンス・バージョンのデータ・コレクターのデータ・コレクター・ホーム・ディレクトリーを指定します。マイグレーション中にこのディレクトリーは削除されません。
was.wsadmin.connection.host	wsadmin ツールの接続先のホストの名前を指定します。 ネットワーク・デプロイメント環境では、デプロイメント・マネージャーへの wsadmin 接続を指定します。 スタンドアロン環境では、サーバーへの wsadmin 接続を指定します。
was.wsadmin.username	IBM WebSphere Application Server 管理コンソールにログオンする権限があるユーザーのユーザー ID を指定します。このユーザーは、アプリケーション・サーバー上でエージェントの役割を持っている必要があります。
was.wsadmin.password	was.wsadmin.username プロパティーに指定されたユーザーに対応するパスワードを指定します。

表 256. マイグレーション・ユーティリティーをサイレント・モードで実行する場合に使用可能なプロパティー (続き)

プロパティー	コメント
was.appserver.profile.name	構成するアプリケーション・サーバー・プロファイルの名前を指定します。 **要確認 :** Liberty プロファイルの場合には、このプロパティーは必要ありません。
was.appserver.home	WebSphere Application Server ホーム・ディレクトリーを指定します。
was.appserver.cell.name	WebSphere Application Server のセル名を指定します。 **要確認 :** Liberty プロファイルの場合には、このプロパティーは必要ありません。
was.appserver.node.name	WebSphere Application Server のノード名を指定します。 **要確認 :** Liberty プロファイルの場合には、このプロパティーは必要ありません。
was.appserver.server.name	新規バージョンのデータ・コレクターにマイグレーションする際に、アプリケーション・サーバー・プロファイル内のアプリケーション・サーバー・インスタンスを指定します。サイレント・プロパティー・ファイルには、このプロパティーの複数のインスタンスを含めることができます。

重要 :

- 以前のメンテナンス・レベルでマイグレーションできるのは、7.3 レベルのデータ・コレクターのみです。データ・コレクターのバージョンは、データ・コレクターのホーム・ディレクトリーのパスに示されます。
- データ・コレクターをバージョン 7.3 から 7.3 フィックスパック 1 にマイグレーションすることはできません。マイグレーションする代わりに、データ・コレクターを構成解除してエージェント・バージョン 7.3 をアンインストールしてください。その後で、エージェント・バージョン 7.3 フィックスパック 1 をインストールしてデータ・コレクターを再び構成してください。

手順

1. サイレント・マイグレーション・プロパティー・ファイルに構成オプションを指定します。
2. 最新バージョンのエージェントのインストール・ディレクトリーからコマンドを実行して、マイグレーション・ユーティリティーをサイレント ・モードで開始します。
   - Linux AIX *dc_home*/bin/migrate.sh -silent *sample_silent_migration_filename*
   - Windows *dc_home*¥bin¥migrate.bat -silent *sample_silent_migration_filename*

タスクの結果

データ・コレクターが、インストールされている最新のメンテナンス・レベルにマイグレーションされます。

次のタスク

マイグレーション・ユーティリティーは、古いバージョンのデータ・コレクターで構成された設定を保持します。これらの設定を変更するには、新規データ・コレクターの *dc_home*¥bin ディレクトリーから構

成ユーティリティーまたは再構成ユーティリティーを対話式またはサイレント・モードで実行することができます。詳しくは、840 ページの『フル構成ユーティリティーを使用したデータ・コレクターの構成または再構成』を参照してください。

Tomcat エージェント: Windows での TEMA コア・フレームワークのアップグレード

Tomcat エージェント用の Windows 上の TEMA コア・フレームワークをアップグレードするには、エージェントとサーバーの両方を停止し、TEMA フレームワークを正常にアップグレードする必要があります。

手順

1. Tomcat サーバーのセットアップを準備します。
2. Tomcat エージェントをインストールして構成します。
3. IBM Cloud Application Performance Management ダッシュボードにログインし、「**エージェント構成**」>「**Tomcat**」を選択し、Tomcat エージェントのインスタンスを選択して「**TT/DD の有効化 (Enable TT/DD)**」をクリックします。
4. Tomcat サーバーを再始動します。
5. IBM APM コア・フレームワークを適用するために、Tomcat エージェントとサーバーの両方を停止します。
6. TEMA/<IBM APM CORE FRAMEWORK_HOME> に移動します。コマンド **apmpatch.bat <Tomcat Agent Installationdir>** を実行します。
   フレームワークがアップグレードされます。
7. 以下の説明に従って、アップグレードされた IBM APM コア・フレームワークのバージョンを確認します。
   <TOMCAT_Agent_Install_Dir>¥InstallITM に移動します。
   KinCInfo.exe -i を実行します。
8. Tomcat サーバーとエージェントの両方を起動します。

データ・コレクターのアップグレード

アップグレードされたデータ・コレクターを含む新規アーカイブ・ファイルが、定期的にダウンロードできるようになります。アーカイブ・ファイルは、IBM Marketplace Web サイトの 製品およびサービスから利用できます。

始める前に

このタスクについて

データ・コレクターをアップグレードするには、以下のステップを実行します。

手順

- ローカル・アプリケーションまたは IBM Cloud アプリケーション、またはその両方からデータ・コレクターを構成解除します。
  - J2SE データ・コレクターの場合、構成解除ステップは不要です。
  - Liberty データ・コレクターの場合、892 ページの『IBM Cloud アプリケーション用のデータ・コレクターの構成解除』または 885 ページの『オンプレミス・アプリケーション用のデータ・コレクターの構成解除』、またはその両方の指示に従います。
  - Node.js データ・コレクターの場合、602 ページの『IBM Cloud アプリケーション用のスタンドアロン Node.js データ・コレクターの構成解除』または 607 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成解除』、またはその両方の指示に従います。
  - Python データ・コレクターの場合、677 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成解除』または 682 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成解除』、またはその両方の指示に従います。

  - Ruby データ・コレクターの場合、730 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクターの構成解除』の指示に従います。
- データ・コレクター・パッケージをダウンロードします。
- ローカル・アプリケーションまたは IBM Cloud アプリケーション、またはその両方をモニターするようにデータ・コレクターを再構成します。
  - Node.js データ・コレクターの場合は、データ・コレクターをアップグレードしてからデータ・コレクターを再構成します。詳しくは、598 ページの『IBM Cloud (旧 Bluemix) アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』または 603 ページの『オンプレミス・アプリケーション用のスタンドアロン Node.js データ・コレクターの構成』、またはその両方を参照してください。
  - Python データ・コレクターの場合は、データ・コレクターをアップグレードしてからデータ・コレクターを再構成します。詳しくは、673 ページの『IBM Cloud アプリケーション用の Python データ・コレクターの構成』または 678 ページの『オンプレミス・アプリケーション用の Python データ・コレクターの構成』、またはその両方を参照してください。
  - Liberty データ・コレクターの場合は、データ・コレクターをアップグレードしてからデータ・コレクターを再構成します。詳しくは、886 ページの『IBM Cloud アプリケーション用の Liberty データ・コレクターの構成』または 881 ページの『オンプレミス・アプリケーション用の Liberty データ・コレクターの構成』、またはその両方を参照してください。
  - J2SE データ・コレクターの場合は、データ・コレクターをアップグレードしてからデータ・コレクターを再構成します。説明については、455 ページの『J2SE のモニターの構成』を参照してください。
  - Ruby データ・コレクターの場合は、データ・コレクターをアップグレードしてからデータ・コレクターを再構成します。説明については、727 ページの『IBM Cloud アプリケーション用の Ruby データ・コレクターの構成』を参照してください。

タスクの結果
データ・コレクターが最新のバージョンにアップグレードされます。

第 12 章 トラブルシューティングとサポート

IBM Cloud Application Performance Management のインストール時、構成時、または使用時に発生する可能性がある問題については、トラブルシューティングの 項目を参照してください。

トラブルシューティング用のコンテンツはこの Knowledge Center で提供されています。以前は、トラブルシューティング用のコンテンツは developerWorks の Cloud Application Performance Management フォーラムで提供されていました。このフォーラムで、引き続き以前のエントリーを検索できます。「Troubleshooting」で始まるエントリーを検索してください。

IBM Cloud Application Performance Management ハイブリッド・ゲートウェイのトラブルシューティングについては、963 ページの『ハイブリッド・ゲートウェイの管理』を参照してください。

エージェントのトラブルシューティング

エージェントのインストールおよび構成に関する問題のトラブルシューティングを行います。

このトラブルシューティング用コンテンツは、developerWorks の Cloud Application Performance Management フォーラムからこの Knowledge Center に移行中です。以前は、トラブルシューティング用のコンテンツは developerWorks の Cloud Application Performance Management フォーラムで提供されていました。このフォーラムで、引き続き以前のエントリーを検索できます。「Troubleshooting」で始まるエントリーを検索してください。

Internet Service Monitoring

ここでは、Internet Service Monitoring の既知の問題について詳しく説明します。

プロファイルの作成ページを 10 分を超えて開いたままにするとプロファイルが作成されず、同じ名前の別のプロファイルも作成されない

問題

プロファイルの作成ページを 10 分を超えて開いたままにするとプロファイルが作成されず、同じ名前の別のプロファイルも作成されません。

症状

プロファイルの作成中、ユーザーが**プロファイルの作成ページ**を 10 分を超えてアイドル状態 (開いたまま何も操作しない状態) にすると、プロファイルを作成しようとしても作成されません。その後、ユーザーが同じ名前で再度プロファイルを作成しようとしても、プロファイルが作成されません。

原因

プロファイルの作成時に MIN 側でロック・ファイルが作成され、他のユーザーに対して、同じプロファイルを対象としたプロファイル作成操作がロックされます。プロファイルの作成が完了すると、このロック・ファイルは削除されます。しかし、作成ウィンドウのアイドル状態が 10 分を超えると、作成イベントがロックされ、ユーザーがプロファイルを作成できなくなります。

解決策

- ユーザーは、プロファイルの作成中にウィンドウのアイドル状態が 10 分を超えないようにしてください。
- 管理者は、作成されたプロファイル用ロック・ファイルを MIN 側の `/opt/ibm/wlp/usr/servers/min/dropins/CentralConfigurationServer.war/data_source/is` から削除してください。
  例えば、プロファイル名が ABC である場合は、ロック・ファイル `$$ABC$$1UjQ9wy1boIHTAQeoWSj1IU.lock` が作成されます。

© Copyright IBM Corp. 2014, 2019

Microsoft Active Directory のモニター

ここでは、Microsoft Active Directory のモニターに関する既知の問題について詳しく説明します。

更新されたオンライン・ヘルプのコンテンツが Microsoft Active Directory エージェントに表示されない

問題

オンライン・ヘルプ・ページが Microsoft Active Directory エージェントの最新コンテンツで更新されません。

症状

Microsoft Active Directory エージェント用の APM ダッシュボードの Eclipse ヘルプに、以下の新規に追加された属性グループのデータ収集間隔およびデータ保存期間に関するヘルプ・コンテンツが存在しません。

- ディレクトリー・サービス
- Kerberos 整合性チェッカー
- Kerberos 鍵配布センター
- ネーム・サービス・プロバイダー
- ディレクトリー交換サービス

原因

この問題は、ビルド・サーバーでの制約が原因で発生します。

解決策

APM ダッシュボードにある各属性グループのコンテキスト・ヘルプでヘルプ・コンテンツを参照できます。

注 : この問題は APM V8.1.4.10 リリースで発生しました。

Microsoft IIS のモニター

ここでは、Microsoft インターネット・インフォメーション・サービスに関する既知の問題について詳しく説明します。

オンライン・ヘルプ・ページが Microsoft IIS APM エージェントの最新コンテンツで更新されない

問題

オンライン・ヘルプ・ページが Microsoft IIS APM エージェントの最新コンテンツで更新されません。

症状

新規に追加された以下の属性グループがオンライン・ヘルプ・コンテンツに存在しません。

- WPROCESS
- MEMIISUS
- ASP ガーベッジ・コレクション
- IISSVRINFO

原因

この問題は、ビルド・サーバーの問題が原因で発生しています。

回避策

ありません。しかし、APM ダッシュボードで特定の属性グループのヘルプ・コンテンツを参照できます。

Microsoft .NET のモニター

ここでは、Microsoft .NET のモニターに関する既知の問題について詳しく説明します。

更新されたオンライン・ヘルプのコンテンツが Microsoft .NET エージェントに表示されない

問題

オンライン・ヘルプ・ページが Microsoft .NET エージェントの最新コンテンツで更新されません。

症状

Microsoft .NET エージェント用の APM ダッシュボードの Eclipse ヘルプにおいて、「データベース呼び出しの詳細」属性グループの下に「要求名」属性のヘルプ・コンテンツが存在しません。

原因

この問題は、ビルド・サーバーでの制約が原因で発生します。

解決策

APM ダッシュボードにある「データベース呼び出しの詳細」属性グループ・ウィジェットのコンテキスト・ヘルプでヘルプ・コンテンツを参照できます。

注 : この問題は APM V8.1.4.10 リリースで発生しました。

Microsoft SharePoint Server のモニター

ここでは、Microsoft SharePoint Server のモニターに関する既知の問題について詳しく説明します。

更新されたオンライン・ヘルプのコンテンツが Microsoft SharePoint Server エージェントに表示されない

問題

オンライン・ヘルプ・ページが Microsoft SharePoint Server エージェントの最新コンテンツで更新されません。

症状

Microsoft SharePoint Server エージェント用の APM ダッシュボードの Eclipse ヘルプにおいて、新規に追加されたグループ・ウィジェット「直近 1 時間のトレース・ ログ・カウント」および「トレース・ログの詳細」に関するヘルプ・コンテンツが存在しません。

原因

この問題は、ビルド・サーバーでの制約が原因で発生します。

解決策

APM ダッシュボードにある各グループ・ウィジェットのコンテキスト・ヘルプでヘルプ・コンテンツを参照できます。

注 : この問題は APM V8.1.4.10 リリースで発生しました。

PostgreSQL のモニター

ここでは、PostgreSQL のモニターに関する既知の問題について詳しく説明します。

アクティブな接続がないデータベースに対する「バッファー・ヒット率 (%) (Buffer Hit Percentage)」ウィジェットにヘルプ・コンテンツが表示されない

問題

アクティブな接続がないデータベースの場合に情報が表示されません。

症状

アクティブな接続がないデータベースが「バッファー・ヒット率 (%) (Buffer Hit Percentage)」ウィジェットに表示されません。ウィジェットのヘルプ・コンテンツには情報が表示されなければなりません。

原因

IBM Cloud App Management との互換性が原因の制約です。

解決策

ありません。ユーザーはこの制約に注意する必要があります。

SUSE15 プラットフォームにメモリーおよび IP アドレスの値が表示されない

問題

エージェントが SUSE15 プラットフォームのローカル環境で PostgreSQL サーバーをモニターしている場合に、メモリーおよび IP アドレスの値が表示されません。

症状

エージェントが SUSE15 プラットフォームで PostgreSQL サーバーをモニターしている場合に、メモリーおよび IP アドレスの値が表示されません。

原因

SUSE15 プラットフォームでエージェントに対するコマンド **netstat** が失敗します。

解決策

SUSE12 プラットフォームを使用すると、ローカル環境で PostgreSQL サーバーをモニターできます。

IBM サポートに提供するモニター・エージェント・ログの収集

問題判別コレクション・ツール *pdcollect* を使用して、IBM サポートから要求されたモニター・エージェントの必要なログおよびその他の問題判別情報を収集します。 PD コレクター・ツールは各モニター・エージェントと共にインストールされます。

始める前に
PD コレクター・ツールでモニター・エージェントからシステム 情報を収集するには、root 権限または管理者権限が必要です。エージェントのログは、以下のフォルダーで個別に確認することができます。

- Windows [64 ビット] *install_dir*¥TMAITM6_x64¥logs
- Windows [32 ビット] *install_dir*¥TMAITM6¥logs
- Linux AIX *install_dir*/logs

制約事項 : pdcollect スクリプトのインスタンスは 1 つしか実行できません。

このタスクについて

install_dir のデフォルトの場所は、以下のとおりです。

- Windows C:¥IBM¥APM
- Linux /opt/ibm/apm/agent
- AIX /opt/ibm/apm/agent

PD コレクター・ツールを実行するには、以下の手順を実行します。

手順

1. コマンド・ラインで、以下のエージェント・ディレクトリーに移動します。
   - Linux AIX *install_dir*/bin
   - Windows *install_dir*¥BIN
2. 次のコマンドを実行します。
   - Linux AIX **./pdcollect**
   - Windows **pdcollect**

   ファイル名にタイム・スタンプが含まれているファイルが /tmp ディレクトリーに生成されます (例: /tmp/pdcollect-nc049021.tar.Z)。
3. その出力ファイルを IBM サポート担当員に送信します。

次のタスク

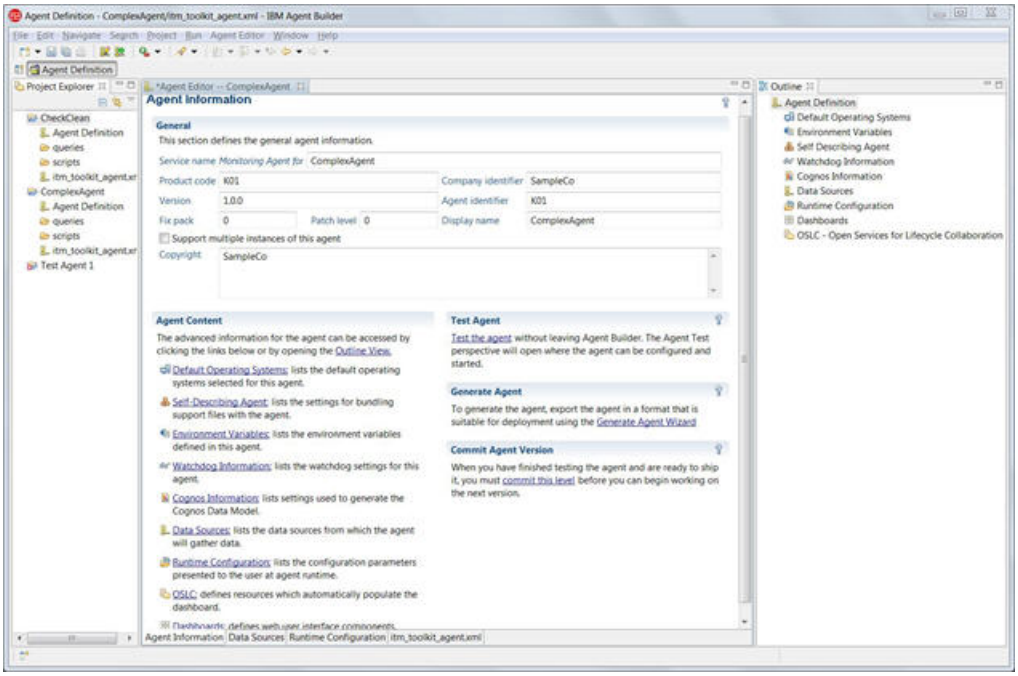
Ruby エージェントをインストールしており、かつそれを診断ダッシュボード用に構成した場合は、Linux システムで kkm コレクター・ツール *kkmCollector* を実行して、構成ファイル、出力ファイル (JSO ファイルなど)、およびログ・ファイルを収集します。

1. *install_dir*/lx8266/km/bin ディレクトリーに移動します。
2. コマンド **./kkmCollector** を実行します。

   ファイル名にタイム・スタンプが含まれているファイルが /tmp ディレクトリーに生成されます (例: /tmp/kkm_dchome.tar.gz)。
3. その出力ファイルを IBM サポート担当員に送信します。

第 13 章 Agent Builder

IBM Agent Builder ツールは、IBM Cloud Application Performance Management でデータ・ソースをモニターするためのエージェントの作成、 変更、デバッグ、パッケージ化を支援するグラフィカル・ユーザー・インターフェースを 提供します。



Agent Builder の概要

IBM Agent Builder を使用して、IBM Tivoli Monitoring 環境または IBM Cloud Application Performance Management 環境のモニター機能を拡張するカスタム・エージェントの作成と変更を行うことができます。カスタム・エージェントは、これらの環境のいずれかを使用して、すべてのタイプの社内ソフトウェアおよびカスタマイズ済みのソフトウェアをモニターできます。

Agent Builder は、Eclipse (オープン・ソースの統合開発環境) に基づいています。

Agent Builder には、Tivoli Monitoring 環境および Cloud APM 環境向けに以下の機能が含まれています。

エージェントの定義と変更
エージェントを作成および変更できます。エージェントは、ディスク、メモリー、プロセッサー、アプリケーションなど、さまざまなリソースの状態とパフォーマンスに関するデータを収集および分析し、そのデータをモニター環境に提供します。

エージェントのテストおよびデプロイメントのためのエージェントの準備
Agent Builder でエージェントをテストし、Agent Builder が稼働するホストに関するデータを収集できます (場合によっては、別のホストからも情報を収集することができます)。簡単に配布およびデプロイメントするためにエージェントをパッケージ化することができます。

Tivoli Monitoring の場合は、以下の追加機能が使用可能です。

カスタム・ワークスペース、シチュエーション、およびアクション実行コマンド
Agent Builder を使用して、Tivoli Monitoring 環境で稼働している新規または既存のエージェントとともに、追加のワークスペース、シチュエーション、およびアクション実行コマンドをアプリケーション・サポート拡張としてパッケージ化できます。

© Copyright IBM Corp. 2014, 2019

レポート・データ・モデル
Agent Builder を使用して、Cognos データ・モデルを生成できます。これを使用して、Tivoli Common Reporting レポートを作成できます。 これらのレポートは、エージェント・イメージの一部としてパッケージできます。

Agent Builder の一般的な手順

以下の表に、Agent Builder を使用して実行可能な主な手順をリストします。

Agent Builder を使用して、IBM Tivoli Monitoring 環境および IBM Cloud Application Performance Management 環境のためのエージェントを作成できます。 また、Agent Builder を使用して、Tivoli Monitoring 環境用のアプリケーション・サポート拡張を作成することもできます。 アプリケーション・サポート拡張を作成するには、ワークスペースとシチュエーションを 作成して 1 つ以上の既存のエージェントを拡張します。

Agent Builder を使用するには、事前に Agent Builder をインストールしておく必要があります。説明については、1167 ページの『Agent Builder のインストールおよび開始』を参照してください。

エージェントを作成、テスト、および使用するには、以下の表内の手順をリストされている順序で実行します。

表 *257.* エージェントの作成に関するクイック・リファレンス情報

目標	**参照**
「**エージェント**」ウィザードを使用して、エージェントを作成する。	• 1172 ページの『エージェントの作成』
エージェントのデータ・ソースおよび属性を作成する。 **重要 :** Cloud APM 環境の場合、要約ダッシュボードには約 5 つまでの属性を表示できます。その属性のいずれか 1 つは、エージェントまたはサブノードの 全体的な状況を示す必要があります。	• 1192 ページの『データ・ソースおよび属性のプロパティーの編集』
Tivoli Monitoring 環境の場合、エージェントのワークスペースおよびシチュエーションを 作成する。 • Tivoli Monitoring バージョン 6.1 フィックスパック 1 以上を実行する • Tivoli Universal Agent ソリューションのバージョンを「00」に戻す •「AppTag」の値を設定する	• 1367 ページの『ワークスペース、アクション実行コマンド、およびシチュエーションの作成』 • 1403 ページの『アプリケーション・サポート・ファイルのインポート』
Cloud APM 環境の場合、エージェントのリソース定義およびダッシュボードを作成する。	• 1373 ページの『Cloud APM 用のエージェントの準備』
Tivoli Monitoring 環境の場合、エージェントのレポート用の Cognos データ・モデルを作成する。	• 1473 ページの『Cognos データ・モデルの生成』
作成済みエージェントのテストおよびデバッグを 行い、モニター情報の可用性を確保する。	• 1376 ページの『Agent Builder でのエージェントのテスト』 • 1413 ページの『コマンド行オプション』 • 1175 ページの『Agent Editor を使用したエージェントの変更』.
インストール・パッケージを生成し、モニター対象ホストにエージェントをインストールする。	• 1385 ページの『エージェントのインストール』

表 257. エージェントの作成に関するクイック・リファレンス情報 (続き)	
目標	**参照**
Agent Builder で作成したエージェントを削除する。	• 1401 ページの『エージェントのアンインストール』

Agent Builder を使用して、カスタム・ワークスペース、シチュエーション、アクション実行コマンドを既存のエージェントのアプリケーション・サポート拡張としてパッケージ化することもできます。これらの機能は、Tivoli Monitoring 環境でのみ使用可能です。

表 258. 他の関数のクイック・リファレンス情報	
目標	**参照**
カスタム・ワークスペース、シチュエーション、およびアクション実行コマンドを作成する。	• 1367 ページの『ワークスペース、アクション実行コマンド、およびシチュエーションの作成』
アプリケーション・サポート拡張をパッケージ化する。	• 1471 ページの『既存エージェント用のアプリケーション・サポート拡張の作成』
カスタム・バンドルを作成する。	• 1494 ページの『非エージェント・ファイル・バンドルの作成』

データ・ソースおよびデータ・セット

エージェントは、1 つ以上のデータ・ソースの情報をモニターできます。エージェントは、その情報を、データ・セットに編成された属性としてモニター・インフラストラクチャーに提供します。

エージェントの作成時に、そのエージェントのデータ・ソース を定義する必要があります。データ・ソースは追加することができます。データ・ソースは、エージェントがモニター情報を収集する方法を定義します。

Agent Builder を使用して、以下のデータ・プロバイダー からのデータ・ソース・モニター情報を使用するエージェントを作成できます。

- プロセスとサービスの可用性
- ネットワーク・システムの可用性 (ICMP ping を使用)
- コマンドの戻りコード
- スクリプトのアウトプット
- Windows イベント・ログ
- Windows Management Instrumentation (WMI)
- Windows パフォーマンス・モニター (Perfmon)
- Simple Network Management Protocol (SNMP)
- SNMP イベント
- Hypertext Transfer Protocol (HTTP) の可用性と応答時間
- SOAP データ・ソースまたは他の HTTP データ・ソース
- Java Database Connectivity (JDBC)
- Java アプリケーション・プログラミング・インターフェース (API)
- Java Management Extensions (JMX)
- Common Information Model (CIM)
- ログ・ファイル
- AIX バイナリー・ログ
- ソケット

他の開発ツールを使用することで、ログ、スクリプトのアウトプット、および Java API データ・ソースによってエージェントに情報を渡すカスタム・モニター・アプリケーションを作成することもできます。

データ・ソースを追加すると、Agent Builder は、対応するデータ・セット をエージェントに追加します。データ・セットは、モニター環境に提供される情報を編成します。IBM Tivoli Monitoring では、データ・セットは属性グループ と呼ばれます。

データ・セットは、複数の属性 (データ・ソースが提供する値) で構成できます。モニター環境がエージェントを照会するたびに、エージェントはデータ・ソースから値を取り出し、データ・セットの属性として返します。

データ・ソースによっては、同じ照会で属性値の複数の行 を返すことがあります (データ・ソースが複数のサービスを同時にモニターする場合など)。

ほとんどのデータ・ソースは、1 つのデータ・セットとして情報を提供します。SNMP データ・ソースおよび JMX データ・ソースは、構成によっては、さまざまなセットの情報を提供する場合があります。SNMP データ・ソースまたは JMX データ・ソースを追加すると、Agent Builder は、この情報に対応するために複数のデータ・セットを作成します。

データ・セット編集して、データをフィルタリングしたり、追加の派生属性 (式を使用して既存の属性から計算される属性) を作成したりすることができます。また、データ・セットを結合し、2 つ以上のデータ・セットの情報を持つ新規データ・セットを作成することもできます。このようにすると、ユーザーは、複数の異なるデータ・ソースの結合情報を表示することができます。

IBM Tivoli Monitoring で、すべての属性内容を表示できます。また、すべてのエージェント・データ・セットからの情報をカスタマイズ・ビューに表示するワークスペースを作成することもできます。IBM Tivoli Monitoring を使用して、属性が特定の値に達するとトリガーされるシチュエーションを作成することができます。シチュエーションは、アラートを発行し、システム・コマンドを呼び出すことができます。

IBM Cloud Application Performance Management で、エージェントの要約ダッシュボードを定義する必要があります。その際に、ダッシュボードに表示される属性を最大で 5 個選択します。任意のデータ・セットからの情報を表として表示する詳細ダッシュボードを定義することもできます。属性が特定の値に達するとトリガーされるしきい値を作成することができます。この属性はダッシュボードに追加する必要はありません。しきい値は、アラートを発行できます。

複数のサーバーまたはサーバーの複数のインスタンスのモニター

エージェントは、同じサーバーの複数のインスタンスを含め、複数のサーバーをモニターすることができます。そのようなエージェントを作成するには、エージェントの複数インスタンス、およびエージェント内のサブノードという 2 つの方法があります。

複数インスタンスは、同じホスト上に多数の同様のインスタンスが存在する可能性があるアプリケーション・サーバーをモニターするための標準的な方法です。IBM Tivoli Monitoring および IBM Cloud Application Performance Management の標準エージェントの多くは、複数インスタンスをサポートします。

複数インスタンス を使用する場合は、エージェントをモニター対象ホストにインストールし、1 つ以上のインスタンスを構成します。その際に各インスタンスの名前を設定します。モニター対象のサーバーの各インスタンスに対してエージェントのインスタンスを構成します。各インスタンスは、エージェントの別々の同一コピーであり、別個に開始および停止することができます。

エージェント内に 1 つ以上のサブノード のタイプを定義することもできます。各タイプは、エージェントがモニターできる別のタイプのリソースに対応している必要があります。サブノード・タイプには、データ・ソースおよびデータ・セットが含まれます。サブノード外のエージェント・レベルのデータ・ソースおよびデータ・セットを定義することもできます。エージェントをホストにインストールする際に、各タイプのサブノードの必要な数を構成することができます。各サブノード・タイプについて、サブノードの数を個別に設定できます。IBM Cloud Application Performance Management の場合、エージェントのダッシュボードおよび各サブノードの個別のダッシュボードを作成することができます。

サブノードの場合は、モニター対象ホストで別の構成ステップが必要です。 また、サブノードを再構成、追加、または削除するには、エージェント全体を停止および再始動する必要があります。インスタンスの再構成、追加、削除は、他のインスタンスに影響を及ぼすことなく実行できます。しかしながら、サブノードには、以下のようにいくつもの利点があります。

- サブノードを使用すると、リソースの消費を抑えつつ、多数のサーバー・インスタンスをモニターできます。ガイドラインとしては、単一のシステムでサポートされる特定タイプのエージェント・インスタンスの数は 10 です。ただし、エージェントは、サブノードを使用して最大 100 個のローカル・サーバーまたはリモート・サーバーをモニターできます。
- 1 つのエージェントには、いくつかの異なる種類のサーバーのサブノード・タイプを含めることができます。モニター対象システム上で、各タイプのサブノードをいくつでも 構成することができます。この機能を使用して、リソースをさらに節約できます。
- サブノードを持つエージェントは、エージェント・レベルでシステム 全体のデータを提供することができます。

同じエージェントに対して複数インスタンスおよびサブノードの 両方を定義できます。この場合、各インスタンスに複数のサブノードを組み込むことができます。各インスタンスの停止および再始動は他のインスタンスとは独立して行うことができます。インスタンス内のすべてのサブノードの 停止および再始動はまとめて行われます。

エージェントのテスト、インストール、および構成

エージェントのインストール・パッケージを作成し、任意の数のモニター対象ホストにインストールすることができます。一部のデータ・ソースについては、データを収集するための構成値を設定する必要があります。

エージェントのデータ・ソースおよび属性を定義したら、Agent Builder でエージェントを実行することにより、テストできます。単一のデータ・セット (属性グループ) のテストを行うことも、エージェントの完全テストを行うこともできます。

エージェントをより広範囲にテストし、それを使用する場合は、インストール・イメージを作成することができます。このイメージは、任意のモニター対象ホストでエージェントをインストールおよび 構成するためのスクリプトを提供します。

ヒント : エージェントをインストールする 前に、ご使用のモニター環境 (IBM Tivoli Monitoring または IBM Cloud Application Performance Management) のオペレーティング・システム・エージェントがホストにインストールされていることを確認してください。

エージェントをインストールした後、そのエージェントを構成する必要が生じる場合があります。エージェントが複数インスタンスをサポートする場合、少なくとも 1 つのインスタンスを作成するようにエージェントを構成する必要があります。

一部のデータ・ソースには追加の構成値を指定する必要があります。例えば、SNMP データ・ソースの場合は、SNMP プロトコルを使用してモニターするホストの IP アドレスを構成する必要があります。インストール・パッケージによってデプロイされる構成スクリプトを使用して、これらの値を設定します。

あるいは、インストール・イメージを作成する前に、Agent Builder でこれらの値を設定することができます。この場合、モニター対象ホストで値を再設定する必要はありません。

ヒント : Cloud APM サーバーのアップグレード後、カスタム・エージェントのヘルプ・ファイルが「ヘルプ・コンテンツ」に表示されないことがあります。ヘルプ・ファイルを表示するには、以下の手順を実行します。

1. IBM Marketplace で、ご使用の Cloud APM サブスクリプションから最新バージョンの IBM Agent Builder をダウンロードします。
2. カスタム・エージェントを再作成します。必ずより高いバージョン番号、フィックスパック、またはパッチ・レベルを「エージェント情報」ページで割り当ててください。
3. モニター対象ホストにカスタム・エージェントをインストールします。
4. Cloud APM コンソールから、ナビゲーション・バーで、「**ヘルプ**」 > 「**ヘルプ・コンテンツ**」をクリックします。カスタム・エージェントのヘルプが表示されます。

オペレーティング・システム要件

Agent Builder によって作成されるエージェントは、モニター環境およびエージェントの作成時に選択した設定に応じて、さまざまなオペレーティング・システムでサポートされます。

Tivoli Monitoring 環境では、Agent Builder で作成されたエージェントは、以下のオペレーティング・システムをサポートできます。

- AIX
- HP-UX
- Linux
- Solaris
- Windows

エージェントは、OS エージェントと同じオペレーティング・システム・バージョンをサポートします。詳細を確認するは、Software Product Compatibility Reports Web サイトにアクセスします。`Tivioli Monitoring` という製品名を検索し、「`OS エージェントおよび TEMA（Tivoli Enterprise Monitoring Agent）（OS Agents & TEMA（Tivoli Enterprise Monitoring Agent））`」コンポーネント・チェック・ボックスを選択します。

IBM Cloud Application Performance Management 環境では、Agent Builder で作成されたエージェントは、以下のオペレーティング・システムをサポートできます。

- AIX
- Linux
- Windows

エージェントは、OS エージェントと同じバージョンをサポートします。詳しくは、System requirements (APM Developer Center) の『Component reports』セクションにあるリンクを使用してください。

Tivoli Monitoring 環境でモニター・エージェントを実行するには、エージェントを実行するすべてのモニター対象システムに適切なオペレーティング・システム・エージェントをインストールします。

IBM Cloud Application Performance Management 環境でモニター・エージェントを実行するには、エージェントを実行するすべてのモニター対象システムに、IBM Cloud Application Performance Management に付属する任意のエージェントをインストールします。

注 : Agent Builder のブラウザーでは、Agent Builder が実行されているシステムからアクセスできるデータ・ソースと情報を処理します。必ず次のいずれかのタイプのシステムで Agent Builder を実行してください。

- オペレーティング・システム、およびエージェントの開発対象となるモニター対象アプリケーションと同じレベルで稼働しているシステム
- エージェントの開発対象となるオペレーティング・システムおよびモニター対象アプリケーションと同じレベルで稼働する別のシステムに接続されているシステム

IBM Tivoli Monitoring に固有の機能

Agent Builder には、IBM Tivoli Monitoring にのみ適用されるいくつかの機能あります。

ナビゲーター・グループを使用して、エージェントが IBM Tivoli Monitoring ナビゲーター・ビューおよびワークスペースに表示するデータを編成することができます。ナビゲーター・グループは、複数の属性グループ (データ・セット) のデータを単一のビューに結合する一方で、元の別個のデータ・セットをユーザーに対して非表示にします。

Tivoli Enterprise Portal を使用して、エージェント用のワークスペース、シチュエーション、およびアクション実行コマンドを作成することができます。その後、Agent Builder を使用して、ワークスペース、シチュエーション、およびアクション実行コマンドをアプリケーション・サポート・ファイルとして保存し、それをエージェントとともにバンドルすることができます。また、Agent Builder は、他のエージェント用のワークスペース、シチュエーション、およびアクション実行コマンドをインポートし、それらに対するカスタム・アプリケーション・サポート・ファイルを作成することができます。

Agent Builder は、エージェント用に Cognos データ・モデルを生成できます。このデータ・モデルを使用して、レポート作成のためにエージェント情報を Cognos Framework Manager (IBM Tivoli Common Reporting の一部) にインポートします。

Agent Builder のインストールおよび開始

IBM Agent Builder をインストールする前に、システムが前提条件を満たしていることを確認してください。その後、インストール・ウィザードまたはサイレント・インストール手順を使用して Agent Builder をインストールします。

ヒント: エージェント のインストールおよび変更については、1385 ページの『エージェントのインストール』を参照してください。

Agent Builder をインストールおよび実行するための前提条件

Agent Builder をインストールして実行するには、現在のシステムが所定の要件を満たしている必要があります。

Agent Builder をインストールするには、以下が備わっていることを確認してください。

- 最小 1 GB の空きディスク・スペースを持つシステム。開発したエージェントには、追加のディスク・スペースが必要になります。
- サポートされているオペレーティング・システム。Agent Builder は以下のオペレーティング・システムで実行できます。
  - Windows Windows
  - Linux Linux (x86 64 ビットのみ)
- Linux Linux オペレーティング・システムを使用している場合は、`libstdc++.so.5` ライブラリーをインストールする必要があります。 このライブラリーを提供する以下のパッケージをインストールできます。
  - Red Hat Enterprise Linux の場合: `compat-libstdc++-33`
  - SUSE Enterprise Linux の場合: `libstdc++-33`

Windows Windows システムの場合、管理者権限を持つユーザーとして Agent Builder を実行できる必要があります。この権限により、Agent Builder の環境が、それによって開発されたエージェントと整合したものになります。

Linux Linux システムの場合、root ユーザーとしても通常のユーザーとしても Agent Builder を実行できます。ただし、通常のユーザーとして実行する場合は、エージェントのテストが制限され、場合によっては使用できないことがあります。

Agent Builder の詳細なシステム要件

Agent Builder の詳細なシステム要件を表示するには、Software Product Compatibility Report を使用します。

Software Product Compatibility Reports の Web サイトにアクセスします。「`IBM Agent Builder`」という製品名を検索します。

Agent Builder のインストール

インストール・ウィザードまたはサイレント・インストール手順を使用して Agent Builder をインストールできます。

ヒント: Agent Builder をインストールする前に、以前のバージョンをすべてアンインストールします。アンインストールについて詳しくは、1171 ページの『Agent Builder のアンインストール』を参照してください。アンインストールしても、既存のエージェント情報は一切失われません。

インストール・ウィザードを使用した Agent Builder のインストール

インストール・ウィザードを使用して、IBM Agent Builder をインストールできます。

始める前に

システムで前提条件が満たされていることを確認してください。前提条件について詳しくは、1167 ページの『Agent Builder をインストールおよび実行するための前提条件』を参照してください。

手順

1. IBM Marketplace にサインインしていない場合は、IBMid とパスワードでサインインして、「**製品およびサービス**」に移動します。
   「**製品およびサービス**」ページは、アクティブなサブスクライバーが 使用できます。問題がある場合は、Cloud Application Performance Management フォーラムまたは Marketplace サポート に移動してください。
2. Agent Builder のインストール・アーカイブ・ファイルを以下の手順でダウンロードします。
   a) Cloud APM のサブスクリプション・ボックスで、「**管理**」 > 「**ダウンロード**」をクリックします。
   b) オペレーティング・システムとして「**マルチプラットフォーム (Multi-Platform)**」を選択します。
   c) IBM Agent Builder パッケージを選択します。
   d)「**ダウンロード (Download)**」をクリックして、ご使用のシステムに `IBM_Agent_Builder_Install.tar` を保存します。
3. インストール・アーカイブ・ファイルを解凍します。
4. 解凍されたイメージ・ディレクトリーで以下のコマンドを使用してインストールを開始します。
   - Windows `setup.bat`
   - Linux AIX `./setup.sh`

   重要 : インストール・プログラムは、Agent Builder の実行に使用するユーザー ID を使用して実行してください。
5.「**IBM Agent Builder**」ウィンドウが表示されたら、言語を選択して「**OK**」をクリックします。
6.「**概要**」ページで、「**次へ**」をクリックします。
7.「**ソフトウェアのご使用条件**」ページで「**ご使用条件の条項に同意します**」をクリックし、「**次へ**」をクリックします。
8.「**インストール・フォルダーの選択**」ページで次のオプションのいずれか 1 つをクリックします。
   - 「**インストール場所を指定してください**」フィールドに指定したディレクトリーに Agent Builder をインストールするには、「**次へ**」をクリックします。
   - Agent Builder をデフォルトのディレクトリーにインストールするには、「**デフォルトのフォルダーに戻す**」をクリックします。
   - 異なるディレクトリーを選択するには、「**選択**」をクリックします。

   注 : 選択するディレクトリー名には、以下の文字が含まれていてはなりません。

   ```
   !
   #
   %
   ;
   ```

   これらの文字のいずれかが含まれている場合、Agent Builder が開始されない可能性があります。
9.「**プリインストールの要約**」ページで「**インストール**」をクリックします。
10.「**IBM Agent Builder のインストール**」ページで、「**インストールの完了**」ページが開くまで待ってから、「**完了**」をクリックします。

タスクの結果

Windows Agent Builder のインストール後に、「スタート」メニューにオプションが追加され、デスクトップに Agent Builder のアイコンが追加されます。インストール・ログ・ファイルは *install_dir*¥IBM_Agent_Builder_InstallLog.xml にあります。

Linux AIX Agent Builder のインストールが完了すると、Agent Builder の実行可能ファイルが *Install_Location*/agentbuilder という名前になります。インストール・ログ・ファイルは *install_dir*/IBM_Agent_Builder_InstallLog.xml にあります。

サイレント・インストール

サイレント・インストールの方法で Agent Builder をインストールすることができます。この方法は、グラフィック環境を必要とせず、複数のホストで容易に複製できます。

このタスクについて

インストール・ディレクトリーのルートにあるインストール・イメージには、サイレント・インストール・オプション・ファイル installer.properties が含まれています。必要に応じてこのファイルを変更してからサイレント・インストーラーを実行する必要があります。このファイルを別の複数のホストにコピーし、それらのすべてのホスト上で Agent Builder を迅速にインストールすることができます。

手順

1. IBM Marketplace にサインインしていない場合は、IBMid とパスワードでサインインして、「**製品およびサービス**」に移動します。

   「**製品およびサービス**」ページは、アクティブなサブスクライバーが 使用できます。問題がある場合は、Cloud Application Performance Management フォーラム または Marketplace サポート に移動してください。
2. Agent Builder のインストール・アーカイブ・ファイルを以下の手順でダウンロードします。
   a) Cloud APM のサブスクリプション・ボックスで、「**管理**」 > 「**ダウンロード**」をクリックします。
   b) オペレーティング・システムとして「**マルチプラットフォーム (Multi-Platform)**」を選択します。
   c) IBM Agent Builder パッケージを選択します。
   d)「**ダウンロード (Download)**」をクリックして、ご使用のシステムに IBM_Agent_Builder_Install.tar を保存します。
3. インストール・アーカイブ・ファイルを解凍します。
4. インストール・イメージ・ディレクトリーにある installer.properties ファイルのコピーを作成します。
5. 必要に応じてこの新規ファイルを編集します。このファイルの内容の例を以下に示します。

```
# ------------------------------------------------------------------------
# IBM Agent Builder
#
# (C) Copyright IBM Corporation 2009.  All rights reserved.
#
# Sample response file for silent install
#
# To use this file, use the following command:
#
# Windows:
#    setup.bat -i silent -f <path>¥installer.properties
#
# Linux or AIX:
#    setup.sh -i silent -f <path>/installer.properties
#
# Where
#    <path> is a fully-quailfied path to the installer.properties
#    file (including the drive letter or UNC path name on Windows).
#    <path> cannot contain spaces.
# ------------------------------------------------------------------------


# ------------------------------------------------------------------------
# This property indicates that the license has been accepted
```

```
# ------------------------------------------------------------------------
# LICENSE_ACCEPTED=FALSE

# ------------------------------------------------------------------------
# This property specifies the install directory
#
# On Windows, the default is:
#     C:¥¥Program Files (x86)¥¥IBM¥¥AgentBuilder
#
# On Linux, the default is:
#     /opt/ibm/AgentBuilder
# ------------------------------------------------------------------------
#USER_INSTALL_DIR=C:¥¥Program Files (x86)¥¥IBM¥¥AgentBuilder
#USER_INSTALL_DIR=/opt/ibm/AgentBuilder
```

6. 解凍したインストール・イメージ・ディレクトリー内の以下のコマンドを実行することにより、サイレント・インストールを開始します。

   Windows `setup.bat -i silent -f path/installer.properties`

   Linux AIX `./setup.sh -i silent -f path/installer.properties`

   ここで、*path* は `installer.properties` ファイルの完全修飾パスです (Windows の場合はドライブ名または UNC パス名を含みます)。パスにスペースを含めることはできません。

Agent Builder の開始

Agent Builder をインストールした後に、Agent Builder を開始することができます。

手順

- 以下のいずれかの手順を実行して、Agent Builder を開始します。
  - Windows Windows システムの場合:
    - コマンド行で、以下のコマンドを入力します。*Install_Location*¥agentbuilder.exe
    - 「**スタート**」 > 「**すべてのプログラム**」 > 「**IBM**」 > 「**Agent Builder**」を選択します。
    - **Agent Builder のデスクトップ・アイコン**をクリックします。
  - Linux Linux システムの場合は、実行可能ファイル *INSTALL_DIR*/agentbuilder を開始します。

  注 : Agent Builder の実行時に、ワークスペース・ディレクトリーの場所の入力を求めるプロンプトが出されます。エージェントを作成するためのファイルは、このディレクトリーに保存されます。 任意のディレクトリーをワークスペースとして指定できます。

Agent Builder でのデフォルトのブラウザーの設定

Linux Linux システムでは、ヘルプ・パネルが表示されるように Agent Builder のデフォルトのブラウザーを設定する必要がある場合があります。

手順

1. 「**ウィンドウ**」 > 「**設定**」を選択して、「**設定**」ウィンドウを開きます。
2. 「**一般**」ノードを選択して展開します。
3. 「**Web ブラウザー**」を選択します。
4. 「**外部 Web ブラウザーの使用**」を選択します。
5. 使用するブラウザーを選択します。
6. オプション: Web ブラウザーを追加するには、以下のステップを実行します。
   a) 「**新規**」をクリックします。
   b) 「**名前**」フィールドに、ブラウザーの記述名を入力します。
   c) 「**ロケーション**」フィールドに、ブラウザー実行可能ファイルの絶対パスを入力します。
   d) 「**OK**」をクリックします。

7.「**OK**」をクリックします。

Agent Builder でのデフォルトの時刻認証局の設定

Agent Builder の「**設定**」ウィンドウで、JAR ファイルの時刻認証局を設定できます。デフォルトの時刻認証局の署名証明書が期限切れになった場合、新しい認証局を設定することで、JAR ファイルの検証を続行できます。

手順

1.「**ウィンドウ**」 > 「**設定**」を選択して、「**設定**」ウィンドウを開きます。
2.「**IBM Agent Builder**」ノードを選択して展開します。
3.「**JAR 署名**」を選択します。
4.「**署名済み JAR ファイルにタイム・スタンプを追加**」を選択します。
5. 時刻認証局の URL を入力します。
6.「**OK**」をクリックします。

Agent Builder のアンインストール

ご使用のオペレーティング・システムに応じて、異なる手順を使用して Agent Builder をアンインストールすることができます。

手順

- Linux
  Linux システムでは、以下のコマンドを実行します。
  a) *INSTALL_DIR*/uninstall/uninstaller
  ここで、*INSTALL_DIR* は、Agent Builder のインストール先となるディレクトリーの名前です。
- Windows
  Windows 7、Windows Server 2008 R2、および Windows のそれ以降のバージョンでは、以下のステップを実行します。
  a) Windows の「**スタート**」 > 「**コントロール パネル**」 > 「**プログラム**」 > 「**プログラムと機能**」を選択して、「プログラムと機能」を開きます。
  b) インストールされたプログラムのリストから「**IBM Agent Builder**」を選択します。
  c)「**アンインストールと変更**」をクリックします。
  d)「**IBM Agent Builder のアンインストール**」ページで「**アンインストール**」をクリックします。
  e)「**アンインストールの完了**」ページの「**完了**」をクリックします。
  ヒント : Windows 7 および Windows Server 2008 R2 では、「**スタート**」 > 「**コンピューター**」 > 「**プログラムのアンインストールと変更**」を選択して、「**プログラムと機能**」ウィンドウに移動することもできます。その後、ステップ 1171 ページの『2』 から続行します。
- Windows
  その他の Windows システムでは、以下の手順を完了します。
  a)「Windows コントロール パネル」から、「**プログラムの追加と削除**」を選択します。
  b)「**IBM Agent Builder**」をクリックします。
  c)「**変更と削除**」をクリックします。
- すべてのオペレーティング・システムでサイレント・アンインストールの方法を使用することもできます。サイレント・アンインストールを開始するには、次のコマンドを実行します。
  - Windows Windows システムの場合: *INSTALL_DIR*/uninstall/uninstaller.exe -i silent
  - Linux Linux システムの場合: *INSTALL_DIR*/uninstall/uninstaller -i silent

サイレント・アンインストール

サイレント・アンインストールの方法を使用してアンインストールを実行することができます。

手順

- サイレント・アンインストールを開始するには、次のコマンドを実行します。

```
INSTALL_DIR/uninstall/uninstaller[.exe] -i silent
```

エージェントの作成

Agent Builder でエージェントの作成を開始するには、新規エージェント・ウィザードを使用します。このウィザードを使用して、基本エージェント構成を設定し、1 つのデータ・ソースを作成できます。その後、Agent Builder でそのエージェントを処理し、他のデータ・ソースやオプション (サブノードおよびナビゲーター・グループを含む) を追加することができます。

エージェントの命名および構成

「**エージェント**」ウィザードを使用して、エージェントに名前を付け、そのバージョン、サポート対象のオペレーティング・システム、およびその他の構成を設定します。

手順

1. 以下のいずれかの方法を使用して新規エージェント・ウィザードを開始します。
   a) ツールバーの 「**新規エージェントの作成 (Create New Agent)**」アイコンをクリックします。
   b) メインメニューから、「**ファイル**」 > 「**新規**」 > 「**エージェント**」と選択します。
   c) メインメニューから、「**ファイル**」 > 「**新規**」 > 「**その他**」と選択します。「**ウィザードの選択**」ページで、「**Agent Builder**」フォルダーをダブルクリックしてから、「**エージェント**」をダブルクリックします。

   「**エージェント**」ウィザードが開きます。
2. 「**次へ**」をクリックします。
3. 「**新規エージェント・プロジェクト**」ページで、「**プロジェクト名**」フィールドにプロジェクトの名前を設定します。Agent Builder は、エージェント・ファイルを格納するフォルダーにこの名前を使用します。オプションで以下の設定を変更できます。
   - エージェント・ファイルを別の場所に格納する場合は、「**デフォルトのロケーションを使用**」をクリアし、「**参照**」をクリックして「**ロケーション**」フィールドで新しいディレクトリーを選択します。
   - さまざまな実効ページ・セットにリソースを追加することで、Eclipse ナビゲーター・ビューでのリソースの表示方法を変更できます。詳しくは、Eclipse ヘルプを参照してください。エージェントを Eclipse 実効ページ・セットに追加するには、「**ワーキング・セットにプロジェクトを追加**」を選択し、「**選択**」ボタンをクリックしてセットを「**ワーキング・セット**」フィールドに追加します。
4. 「**次へ**」をクリックします。
5. 「**一般情報**」ページで、以下の設定を構成します。
   - 「**著作権**」フィールドに、新規エージェントに関して使用する著作権文を入力します。この文は、著作権の法的要件を満たしている必要があります。この著作権文は、エージェントに対して生成されるすべてのファイルに挿入されます。この文は後で編集することができます。
   - エージェントを作成するオペレーティング・システムを 選択します。

     重要 : Agent Builder 内でエージェントの完全テストを実行する場合は (手順については、1380 ページの『エージェントの完全テスト』を参照)、以下のことを確認してください。

     – Windows で Agent Builder を実行する場合は、32 ビット版のオペレーティング・システムがインストールされていること。

– Linux で Agent Builder を実行する場合は、64 ビット版のオペレーティング・システムがインストールされていること。

重要 : まれに、32 ビットのオペレーティング・システム・エージェントのみがインストールされている 64 ビットのシステムにエージェントをインストールする 必要が生じる場合があります。この場合は、64 ビット・バージョンのオペレーティング・システムは選択せず、必ず 32 ビット・バージョンを選択してください。

重要 : 64 ビットの Windows Server 2003 R2 以前の Windows システムは、Agent Builder を使用して作成されたエージェントによってサポートされません。

6.「**次へ**」をクリックします。
7.「**エージェント情報**」ページで、以下の設定を構成します。

- 「**サービス名**」フィールドで、エージェントのサービス名を設定します。この名前は、IBM Tivoli Monitoring 環境の「**Tivoli Monitoring Services の管理**」ウィンドウ、ならびに IBM Cloud Application Performance Management の「**Monitoring Services の管理**」ユーティリティーおよびしきい 値エディターで表示されます。Windows システムでは、エージェントを実行する Windows サービスの名前でもあります。サービスのフルネームは、常に「`Monitoring Agent for`」で始まります。名前の残りの部分を入力します。この部分は通常、このエージェントがモニターするサービスを記述します。名前には、文字、数字、スペース、および下線を使用できます。
- 「**製品コード**」フィールドで、エージェントの 3 文字の製品コードを設定します。製品コードは、IBM Tivoli Monitoring および IBM Cloud Application Performance Management の両方に必要です。Agent Builder で使用できる製品コードの範囲が予約されています。許可される値は、`K00` から K99、`K{0-2}{A-Z}`、および `K{4-9}{A-Z}` です。

  重要 : これらの値は内部使用のみを目的としたもので、お客様の組織外で共有または販売されるエージェントに対して使用することを目的としたものではありません。 他のユーザーと共有するエージェントを作成する場合は、製品コードを予約するために、toolkit@us.ibm.com にメモを送信する必要があります。製品コードの要求には、作成するエージェントの説明を組み込まなければなりません。 製品コードが割り当てられ、登録された上で、送り返されてきます。 3 文字の製品コードを受け取る際に、Agent Builder が割り当てられている製品コードを使用できるようにする手順が通知されます。
- 「**会社 ID**」フィールドで、エージェントを開発する組織を一意的に識別するストリングを設定します (IBM は予約されています)。お客様の会社の URL からそのストリングを採用することができます。例えば、会社の Web サイトが `mycompany.com` である場合は、テキスト `mycompany` を使用します。
- 「**エージェント ID**」フィールドで、エージェントを一意的に識別するストリングを設定します。デフォルトでは、Agent Builder は、製品コードと同じになるようにエージェント ID を設定します。

  重要 :「**エージェント ID**」フィールドと「**会社 ID**」フィールドの長さの組み合わせが 11 文字を超えることはできません。
- 「**バージョン**」フィールドで、エージェント・バージョンを設定します。エージェント・バージョンには、*V.R.R* 形式の 3 桁の数字が含まれます。

  V = バージョン
  R = リリース
  R = リリース

  モニター環境で表示する場合、*V.R.R* の値は、`0V.RR.00.00` の形式に変換されます。

  ヒント : Agent Editor では、「**パッチ・レベル**」フィールドを使用できます。「**パッチ・レベル**」フィールドを使用できるのは、エージェントのバージョン更新がないフィックスをリリースする 場合です。
- エージェントに複数のインスタンスを サポートさせる場合は、「**このエージェントの複数インスタンスの サポート**」チェック・ボックスを選択してください。 エージェントの複数インスタンスを使用して、同じホスト上のアプリケーションの複数のインスタンスをモニターすることも、1 つのホストにインストールされたエージェントを使用して、異なるホスト上の複数のソフトウェア・サーバーを

モニターすることもできます。 複数インスタンスをサポートするエージェントをインストールすると、必要な任意の数のインスタンスを作成して構成することができます。

次のタスク

「**次へ**」をクリックして、エージェントの初期データ・ソースを定義します。詳しくは、1174 ページの『初期データ・ソースの定義』を参照してください。

初期データ・ソースの定義

エージェントの作成時に、エージェントがモニターする 初期データを定義します。Agent Editor で後からデータ・ソースを追加することができます。

このタスクについて

「**エージェントの初期データ・ソース**」ページを使用して、新規エージェントがモニターするデータ・ソースを定義します。さまざまなデータ・プロバイダーからデータ・ソースを作成するための詳細な手順については、1218 ページの『データ・ソースの定義およびテスト』を参照してください。

手順

1. 「**エージェントの初期データ・ソース**」ページで、「**データ・カテゴリーのモニター**」のいずれかおよび「**データ・ソース**」のいずれかを選択します。
2. 「**次へ**」をクリックすると、ウィザードが提供するガイドに従い、指定した任意のデータ収集タイプを定義および構成することができます。

   ヒント: また、このウィザードを使用して、データ・ソースを定義したり、エージェントを編成するためのサブノードやナビゲーター・グループを追加したりすることができます。サブノードについて詳しくは、1344 ページの『サブノードの使用』を参照してください。ナビゲーター・グループ (IBM Tivoli Monitoring でのみ使用) について詳しくは、1342 ページの『ナビゲーター・グループの作成』を参照してください。
3. 複数のデータ行を返す可能性がある新規データ・ソースを定義した場合は、キー属性の選択を求めるプロンプトが表示されます。詳しくは、1174 ページの『キー属性の選択』 を参照してください。
4. 最初のデータ・ソースを定義すると、「**データ・ソース定義**」ウィンドウが表示されます。別のデータ・ソースを追加するには、エージェントを選択するか、サブノードまたはナビゲーター・グループ (存在する場合) を選択し、「**選択対象に追加**」ボタンをクリックします。
5. データ・ソースの定義を完了するには、「**終了**」をクリックします。Agent Builder は、新規エージェントを作成し、それを Agent Editor で開きます。

キー属性の選択

属性グループから複数のデータ行が返される場合、キー属性を選択する必要があります。

このタスクについて

属性グループが複数のデータ行を返す可能性がある場合は、各行がモニター対象のエンティティーを 表します。モニター対象のデータがサンプリングされるごとに、モニター環境は、モニター対象のエンティティーと、そのエンティティーの 以前のサンプルに対して、1 つの行をマッチングします。このマッチングはキー属性を使用して実行されます。属性グループ内の 1 つ以上の属性をキー属性として識別できます。これらのキー属性を比較することにより、あるモニター対象エンティティーを 他のエンティティーと 区別することができます。同じモニター対象エンティティーの 場合、あるサンプルと 別のサンプルとの間でキー属性が変わることはありません。

率属性と差分属性は、現在のサンプルと 以前のサンプルを比較することによって計算されます。キー属性が同じであれば、エージェントは必ず同じモニター対象エンティティーについて 値を比較します。同様に、要約およびプルーニング・エージェントは、同一のキー属性を持つサンプルを 要約します。さらに、キー属性として設定される属性は、シチュエーションの「表示項目」として使用することもできます。

「**エージェントの初期データ・ソース**」ページで新規データ・ソースに関する詳細を指定します。選択したデータ・ソースから複数のデータ行が返される場合、Agent Builder がキー属性を検出できることもあります。検出できない場合は、キー属性の選択を求めるプロンプトが出されます。

手順

- 「**キー属性の選択**」ページで、以下のステップのいずれかを実行します。
  - このエンティティーのキー属性である 1 つ以上の属性をリスト上でクリックします。複数の属性を選択するには、Ctrl キーを押し下げます。
  - この属性グループが 1 行のみ返す場合は、「**単一データ行の生成**」を選択します。このオプションを選択した場合、この属性グループでは常に 1 つのモニター対象エンティティーのみが報告されるため、キー属性は不要です。

Agent Editor を使用したエージェントの変更

Agent Editor を使用して、エージェントのバージョンの変更、保存、およびコミットを行います。

Agent Builder で新規エージェントを作成できます。詳しくは、1172 ページの『エージェントの作成』を参照してください。 エージェントを作成したら、Agent Editor を使用してそのエージェントを変更できます。

Agent Builder で作成したエージェントを Agent Editor で開くには、「**プロジェクト・エクスプローラー**」ペインでそのエージェントの名前を見つけ、展開します。そのエージェントの名前の下にある「**エージェント定義**」をダブルクリックします。 あるいは、`itm_toolkit_agent.xml` ファイル名をダブルクリックします。

Agent Editor は、既存のエージェントのプロパティーを変更するために使用することができる、マルチページの Eclipse エディターです。エディター内の各ページが、エージェントの特定の機能に対応しています。

「**エージェント定義**」ノードの下のアウトライン・ビューに、使用可能なページのリストが表示されます。アウトライン・ビューのノードをクリックすると、別のページへ簡単に切り替えることができます。アウトライン・ビューがない場合、または別のビューの後ろに隠れている場合は、「エージェント定義 (Agent Definition)」パースペクティブをリセットします。パースペクティブをリセットするには、「**ウィンドウ**」 > 「**パースペクティブのリセット**」を選択します。あるいは、「**エージェント定義**」タブを右クリックし、メニューから「**リセット**」を選択します。

注 : エージェントの作成に関する詳細情報および手順については、1172 ページの『エージェントの作成』を参照してください。

Agent Editor には以下のページがあります。

- 1175 ページの『「エージェント情報」ページ』
- 「データ・ソース定義」ページ
- 「ランタイム構成情報」ページ
- 「エージェント XML エディター」ページ (itm_toolkit_agent.xml)

注 : 「エディター」ページから別のページへは、そのページのタブをクリックすることでも、表示を切り替えることができます。一部のページでは、アウトライン・ビューで選択されているタブのみが表示されます。選択されていないタブでも、ページに強制的に表示することができます。タブをページに強制的に表示するには、ピン・アイコンをクリックし 、アイコンのピンがページを 指し示すようにします。

「エージェント情報」ページ

「**エージェント情報**」ページは、Agent Editor のメインページです。

「**エージェント情報**」ページには以下の情報が表示されます。

- エージェント・サービス名および製品コードを含む一般的なエージェント情報。「**拡張**」をクリックして、異なる用途のためにさまざまな名前を設定できますが、この設定は、通常、不要です。
- エージェント・コンテンツ情報
  - 「**デフォルト・オペレーティング・システム**」リンク
  - 「**自己記述型エージェント**」リンク
  - 「**環境変数**」リンク

  - 「**Watchdog 情報**」リンク
  - 「**Cognos 情報**」リンク
  - 「**データ・ソース**」リンク
  - 「**ランタイム構成**」リンク
  - 「**リソース**」リンク
  - 「**ダッシュボード**」リンク
- 「**テスト・エージェント**」リンク
- 「**エージェントの生成ウィザード**」リンク
- 「**エージェント・バージョンのコミット**」リンク

一時的なエラー・メッセージが表示される時間の設定

Agent Editor ウィザードには、一時的なエラー・メッセージが表示される場合があります。この一時的なメッセージは、ウィザードのヘッダーに少しの間だけ表示されます (デフォルトは 3 秒)。このメッセージの表示時間を構成することができます。この設定を変更するには、以下の手順を実行します。

1. Agent Builder のメニュー・バーから「**ウィンドウ**」 > 「**設定**」を選択します。「**設定**」ウィンドウが開きます。
2. 「**Agent Builder**」を選択します。
3. 「**一時的なエラー・メッセージが表示される時間 (秒)**」の値を設定します。
4. 「**OK**」をクリックします。

デフォルト・オペレーティング・システム

「**デフォルト・オペレーティング・システム**」ページは、エージェントの作成対象とするオペレーティング・システムを変更するために使用します。

手順

- 「**デフォルト・オペレーティング・システム**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションの「**デフォルト・オペレーティング・システム**」をクリックするか、またはアウトライン・ビューの「**デフォルト・オペレーティング・システム**」ノードをクリックします。
- 「**デフォルトのオペレーティング・システム**」ページで、エージェントがサポートしなければならないオペレーティング・システムを選択します。

  ユーザーがエージェントのインストール・パッケージを生成するときに、Agent Builder は、選択されたオペレーティング・システムのファイルをパッケージに追加します。エージェントに追加したデータ・ソースのうち、Windows オペレーティング・システムに固有でないものは、選択したすべてのオペレーティング・システムで使用可能になります。特定のデータ・ソースが使用可能なオペレーティング・システムのリストは、このデフォルト選択から変更できます。特定のデータ・ソースに使用可能なオペレーティング・システムを変更するには、「**データ・ソース定義**」ページの「**オペレーティング・システム**」ペインを使用します。デフォルトのオペレーティング・システムを選択しなかった場合は、「**データ・ソース定義**」ページでデータ・ソースごとにオペレーティング・システムを選択する必要があります。

  重要 : Agent Builder 内でエージェントの完全テストを実行する場合は (手順については、1380 ページの『エージェントの完全テスト』を参照)、以下のことを確認してください。

  - Windows で Agent Builder を実行する場合は、32 ビット版のオペレーティング・システムがインストールされていること。
  - Linux で Agent Builder を実行する場合は、64 ビット版のオペレーティング・システムがインストールされていること。

  重要 : まれに、32 ビットのオペレーティング・システム・エージェントのみがインストールされている 64 ビットのシステムにエージェントをインストールする 必要が生じる場合があります。この場合は、

64 ビット・バージョンのオペレーティング・システムは選択せず、必ず 32 ビット・バージョンを選択してください。

自己記述型エージェント

IBM Tivoli Monitoring 環境の場合、「**自己記述型エージェント**」ページを使用して、エージェントとともにエージェントのサポート・ファイルをバンドルするかどうかを指定します。IBM Cloud Application Performance Management 環境の場合、「自己記述型エージェント」を有効にしておく必要があります。

手順

- 「**自己記述型エージェント**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションにある「**自己記述型エージェント**」をクリックするか、アウトライン・ビューの「**自己記述型エージェント**」ノードをクリックします。

  Agent Builder 6.2.3 以降で作成されるすべての新規エージェントでは、デフォルトで自己記述が有効になっています。エージェントが IBM Cloud Application Performance Management 環境用である場合は、自己記述を有効にする必要があります。

  エージェントについて自己記述を有効にすると、アプリケーション・サポート・パッケージがエージェント・イメージに組み込まれます。これにより、このエージェントは、Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server、Tivoli Enterprise Portal Browser 用のサポート・ファイルをシードできるようになります。自己記述型エージェントの詳細については、「*IBM Tivoli Monitoring* インストールおよび設定ガイド」および「*IBM Tivoli Monitoring* 管理者ガイド」を参照してください。IBM Cloud Application Performance Management 環境において、自己記述により、エージェントは Cloud APM サーバーにサポート・ファイルをシードできるようになります。シードは、この環境で必須のステップです。

  注 : IBM Tivoli Monitoring 環境において、自己記述型エージェント機能を有効にするには、Tivoli Monitoring バージョン 6.2.3 以降がインストールされている必要があり、かつ Tivoli Monitoring で自己記述が有効になっている必要があります。デフォルトでは、Tivoli Monitoring で自己記述はオフになっています。

  注 : 「**このエージェントの自己記述の有効化**」チェック・ボックスを選択しても、エージェントは以前のバージョンの Tivoli Monitoring で処理を行えます。

環境変数

「**環境変数**」ページを使用して、エージェントの実行中に使用可能な環境変数を表示または変更します。

始める前に

Agent Editor および「**エージェント情報**」ページについて詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

このタスクについて

スクリプト内でアクセスするためにユーザーが環境変数を定義するか、エージェントに特定の動作を行わせる定義済みの環境変数を使用することができます。定義済みの変数のリストについては、1178 ページの『環境変数のリスト』 を参照してください。

手順

1. 「**環境変数**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションで「**環境変数**」をクリックします。あるいは、**アウトライン**・ビューの「**環境変数**」ノードをクリックします。
2. 新規変数を追加するには、「**環境変数**」ページで「**追加**」をクリックします。あるいは、既存の変数を編集する場合は、その変数を選択して、「**編集**」をクリックします。
3. 「**環境変数の情報**」ウィンドウで、以下の値を設定します。
   - 「**名前**」フィールドに変数名を入力するか、 リストから定義済みの名前を選択します。

- エージェントの変数を設定する場合は、「**値**」フィールドに 変数の値を入力します。値を入力しなかった場合、 エージェントは既存の変数の値を伝搬します。
- 「**説明**」フィールドに 変数の説明を入力するか、 定義済み変数の既存の説明を保持します。

a)「**OK**」をクリックします。
新しい変数が「**エージェント情報**」ページの 表にリストされます。

環境変数のリスト
環境変数を使用して、実行時のエージェントの動作を制御します。

「**環境変数**」ページを使用して、環境変数をエージェントに組み込むことができます。Windows システムの場合、環境変数はエージェントの K*XX*ENV ファイルで定義されています。UNIX および Linux システムの場合、これらの変数はエージェントの $CANDLEHOME/config/*XX*.ini ファイルで定義できます。*XX* は 2 文字の製品コードです。新しい設定を有効にするには、エージェントを再始動する必要があります。

注 : C シェルを実行するリモート・システムで環境変数が正しく設定されていません。環境変数を使用する場合は、異なるシェルを使用してください。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表

環境変数	デフォルト値	有効な値	説明
CDP_*ATTRIBUTE_GROUP*_REFRESH_INTERVAL	適用できません	負ではない整数	指定された特定の属性グループがバックグラウンドで更新される間隔 (秒数)。 この変数は、CDP_DP_REFRESH_INTERVAL と 同様に機能します。ただし、指定された属性グループのみを 対象とします。実際の属性グループ名が大文字でなくても、変数名の中の属性グループ名は大文字である必要があります。
CDP_DP_CACHE_TTL	55	1 以上の整数。	属性グループについて収集されたデータは、この秒数の間、キャッシュの中に入れられます。 この時間間隔中に同じデータが繰り返し要求された場合は、このデータのキャッシュ・コピーが返されます。 この値は、エージェント内のすべての属性グループに適用されます。
CDP_*ATTRIBUTE_GROUP*_CACHE_TTL	CDP_DP_CACHE_TTL の値。	1 以上の整数。	特定の指定された属性グループについて収集されたデータは、この秒数だけ、キャッシュに格納されます。この時間間隔中に同じデータが繰り返し要求された場合は、このデータのキャッシュ・コピーが返されます。 この値は、指定されたグループの CDP_DP_CACHE_TTL の値を上書きします。実際の属性グループ名が大文字でなくても、変数名の中の属性グループ名は大文字である必要があります。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_DP_IMPATIENT_ COLLECTOR_TIMEOUT	サブノードが定義されている場合は 5。それ以外は設定されません	任意の正の整数	データ収集が行われるのを待機する秒数。この時間が経過するとタイムアウトになり、キャッシュ・データが失効している場合でも、そのキャッシュ・データが返されます。(キャッシュされているデータは、CDP_DP_CACHE_TTL 秒経過すると、失効となります。) この変数が設定されていない場合、エージェントはデータ収集が完了するまで待機します。この待機は場合により Tivoli Enterprise Portal のタイムアウトとなり、待機は中止されます。スレッド・プールが構成されていない場合、この変数は無視され、データ収集は同期で実行されます。
CDP_DP_REFRESH_INTERVAL	サブノードが定義されている場合は 60。それ以外は設定されません	負ではない整数	属性グループがバックグラウンドで更新される間隔 (秒数)。 この変数が設定されていないか、または 0 に設定される場合、バックグラウンド更新は使用不可にされます。 スレッド・プールが構成されている場合 (変数 CDP_DP_THREAD_POOL_SIZE を参照)、属性グループは並行して最新表示できます。スレッド・プールがない場合、 更新は順次行われるため、 長時間かかることがあります。論理的にスレッド・プール・サイズ 1 に相当します。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(*続き*)*

環境変数	デフォルト値	有効な値	説明
CDP_DP_THREAD_POOL_SIZE	サブノードが定義されている場合は 15。それ以外は設定されません	負ではない整数	CDP_DP_REFRESH_INTERVAL で定義された間隔で、バックグラウンドでデータ収集を実行するために作成されたスレッドの数。この変数が設定されていないか、または 0 に設定される場合、スレッド・プールはありません。 CDP_DP_THREAD_POOL_SIZE に 1 よりも大きな値を設定し、かつ CDP_DP_REFRESH_INTERVAL に 0 を設定した場合、CDP_DP_THREAD_POOL_SIZE の値が無視され、要求した場合にのみデータ収集が実行されます。 「スレッド・プールの状況」属性グループは、スレッド・プールがどのように実行されているかを示します。 最善の結果を得るために、「スレッド・プールの状況」を使用して、スレッド・プール・サイズおよび最新表示間隔を調整します。デフォルトでは、 この属性グループの照会はエージェントのナビゲーター・ツリーに表示されません。エージェントのカスタム・ワークスペースに照会を組み込む必要はありません。 ただし、「スレッド・プールの状況」照会を基本エージェント・レベルのワークスペース・ビューに割り当てることにより、容易に表示させることができます。
CDP_JDBC_MAX_ROWS	1000	任意の正の整数	JDBC データ・プロバイダーが返すデータ行の最大数。結果セットに含まれるデータ行がこの行数を超える場合は、この最大値までのデータのみが処理されます。IBM Tivoli Monitoring に返されるデータが多くなりすぎないように照会を作成できます。
CDP_NT_EVENT_LOG_GET_ALL_ENTRIES_FIRST_TIME	NO	YES、NO	YES に設定した場合、エージェントは Windows イベント・ログのすべてのイベントを送信します。NO に設定した場合、Windows イベント・ログの新しいイベントのみが送信されます。
CDP_NT_EVENT_LOG_CACHE_TIMEOUT	3600	300 以上の整数。	Windows イベント・ログのイベントがエージェントによってキャッシュされる秒数。 イベント・ログ属性グループが照会されると、キャッシュにあるすべてのイベントが返されます。 **注 :** この変数は、現在は使用されていません。CDP_PURE_EVENT_CACHE_SIZE 変数を使用してください。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_PURE_EVENT_CACHE_SIZE	100	1 以上の正の整数。	キャッシュに入れるイベントの最大数。新規レコードを処理するために構成されているログ・ファイル・データ・ソース、Windows イベント・ログ属性グループに対して適用されます。また、JMX モニターおよび通知に対しても適用されます。ログにあるそれぞれの新規レコードについて、 イベントが送信されます。この環境変数により、エージェントがキャッシュに入れるイベントの数が定義されます。属性グループが照会されると、キャッシュ内の値が返されます。
CDP_DP_ACTION_TIMEOUT	20 秒	1 以上の正の整数。	エージェントによるアクション実行の処理が完了するのを待つ秒数。
CDP_DP_SCRIPT_TIMEOUT	30 秒	10 以上の正の整数。	スクリプト・ベースの属性グループが開始したプログラムが完了するのを待つ秒数。
CDP_DP_PING_TIMEOUT	30 秒	10 以上の正の整数。	コマンドの戻りコードが開始したプログラムが完了するのを待つ秒数。 **注 :** この変数は、ICMP ping データ・プロバイダーとは 関連していません。
CDP_SNMP_MAX_RETRIES	2	任意の正の整数	SNMP 要求の送信を再試行する回数。応答を受信しない場合、SNMP エージェントに送信される要求の総数は、 この値に 1 を加えた数です。
CDP_SNMP_RESPONSE_TIMEOUT	2 秒	任意の正の整数	各 SNMP 要求がタイムアウトになるまで待機する秒数。 属性グループの行は、それぞれ別個の要求です。このタイムアウト値は、応答を待機する秒数です。この秒数が経過した後、再試行が実行されます。単一のデータ行の合計タイムアウトは、(CDP_SNMP_MAX_RETRIES + 1) * CDP_SNMP_RESPONSE_TIMEOUT になります。デフォルトの 合計タイムアウト値は (2+1) * 2 = 6 秒になります。
CDP_DP_HOSTNAME	最初にインストールしたネットワーク・インターフェースの 名前	IP アドレスまたはホスト名	複数インターフェース・システムの優先ホスト名 (ネットワーク・インターフェース) を設定します。この環境変数は、エージェントがその listen ポートをデフォルト以外のネットワーク・インターフェース・アドレスにバインドする場合に使用します。SNMP データ・プロバイダーで使用されます。 ソケットのデータ・ソースの場合、この変数は CDP_DP_ALLOW_REMOTE も設定されている場合に適用されます。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_SNMP_ALLOW_ DECREASING_OIDS	NO	YES、NO	YES に設定されている場合、SNMP データ・プロバイダーは戻された OID が増加しているかどうかを検査しません。モニター対象エージェントでは、一般にこの検査で捕捉される問題が発生することがあるため、YES に設定する際には十分に注意してください。
KUMP_DP_COPY_MODE_SAMPLE_I NTERVAL	60	待機時間 (秒)	ログ・ファイル・データ・プロバイダーの場合に、エージェントが「**ファイルがサンプリングされたときにすべてのレコードを処理**」と定義されているときにファイル内容を再読み取りするまでの待機時間を指定します。この時間は秒単位で指定します。
KUMP_MAXPROCESS	100%	5-100%	ログ・ファイル・データ・プロバイダーの場合に、ファイル・データの処理に使用する最大プロセッサー使用量を指定します。値の範囲は 5 から 100 パーセントです。デフォルトは 100% です。
KUMP_DP_SAMPLE_FACTOR	5	負ではない整数	ログ・ファイル・データ・プロバイダーの場合、Agent Builder で「**ファイルがサンプリングされたときにすべてのレコードを処理**」を選択する際に、サンプリング係数を設定します。この待機時間により、パターンのスキャンがログに記録される前に、複数のレコードにわたるパターンが確実に書き込まれます。
KUMP_DP_EVENT	5	負ではない整数	ログ・ファイル・データ・プロバイダーの場合に、 イベント・データのサンプリング頻度を 秒単位で設定します。
KUMP_DP_FILE_EXIST_WAIT	YES	YES、NO	ログ・ファイル・データ・プロバイダーの場合に、 モニター対象ファイルがないか空であることが検出されたときに ファイル・モニター・スレッドの実行を継続することを 指定します。スレッドは、ファイルが作成されるまで待機し、数秒ごとに再確認し、ファイルが使用可能になった時点でモニターを開始または再開します。
KUMP_DP_FILE_SWITCH_ CHECK_INTERVAL	600	負ではない整数	ログ・ファイル・データ・プロバイダーが、動的ファイル名サポートが有効の場合に切り替え先として使用する別のモニター・ファイルを検索する頻度 (秒単位)。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
KUMP_DP_FILE_ROW_ PAUSE_INCREMENT	なし	負ではない整数	ログ・ファイル・データ・プロバイダーの場合、ファイル・モニター・スレッドが一時停止する前に読み取るファイル・レコードの数を指定します。この一時停止の目的は、前の更新を処理できるようにすることです。この環境変数を使用するのは、モニター対象ファイルが大量の新規レコードを受け取り、一部のレコード更新が失われる可能性がある場合に限ります。
CDP_COLLECTION_TIMEOUT	60 秒	任意の正の整数	別のプロセスで開始されたデータ・コレクターからの応答をエージェントが待機する秒数。このようなデータ・コレクターの例としては、JMX、JDBC、HTTP、および SOAP データ・コレクターがあります。
CDP_SSH_TEMP_DIRECTORY	. (ピリオド)	リモート・システム上の任意の有効なパス・ストリング	SSH 対応スクリプト・データ・プロバイダーの場合に、リモート・システム上の場所を指定します。エージェントに付属するスクリプト・ファイルは、この場所にアップロードされます。相対的な場所は、ユーザーのホーム・ディレクトリーを基準としています。デフォルトの . (ピリオド) は、ユーザーのホーム・ディレクトリーを示します。
CDP_SSH_DEL_COMMAND	rm -Rf	リモート・システム上の任意の有効な削除コマンド・ストリング	SSH 対応スクリプト・データ・プロバイダーの場合に、エージェントとともに提供されたアップロード済みのスクリプト・ファイルを削除するために開始するコマンドを指定します。
CDP_SNMP_SEND_DELAY_ FACTOR	0 ミリ秒	任意の正の整数	最初の SNMP 送信を 0 から 指定の時間 (ミリ秒単位) まで遅延します。この変数が使用可能になるのは、 スレッド・プールも使用可能な場合のみです。この遅延はすべての送信に適用されるわけではなく、 属性グループによって行われる最初の送信のみに適用されます。 この変数は、複数の要求を同時に受信するときにモニター対象デバイスが正常に応答しなくなる場合に役立ちます。
CDP_ICMP_PING_REFRESH_ INTERVAL	60 秒	1 以上の整数	デバイス・リスト・ファイルにあるシステムが この間隔で ping されます。ping に時間がかかりすぎる場合は、再度 ping が開始される前に、必ず CDP_PING_MIN_INTERVAL_DELAY 秒以上の遅延が設けられます。データはこの設定よりも低い頻度でリフレッシュされます。デバイス・リスト・ファイルの項目数および応答を受信するまでの時間に基づいて、より低い頻度でデータをリフレッシュできます。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_ICMP_PING_MIN_INTERVAL_DELAY	30 秒	1 以上 CDP ping リフレッシュ間隔未満の 任意の整数	デバイス・リスト・ファイルにあるデバイスに ping した後、少なくともこの秒数が経過するまでは、次の ping リフレッシュ間隔は開始しません。
CDP_ICMP_PING_BURST	10	0 以上の整数	CDP_ICMP_PING_BURST_DELAY 変数で指定された時間だけエージェントが一時停止するまでに ping を送信する回数。0 の値を指定すると、この機能が無効になります。
CDP_ICMP_PING_BURST_DELAY	10	0 以上の整数	CDP_ICMP_PING_BURST 変数で 定義された設定数の ping を送信した後、待機する時間 (ミリ秒単位)。0 の値を指定すると、この機能が無効になります。
CDP_ICMP_PING_TIMEOUT	2000 ミリ秒	1 以上の整数	ping 応答を待機する 時間 (ミリ秒単位)。この設定は、試行される各 ping に適用されます。ホストごとに ping が 3 回試行されます。3 回の試行で応答を受け取らなかった場合は、応答待機時間合計は CDP_ICMP_PING_TIMEOUT に 3 を掛けた値になります。デフォルトでは、この値は 6000 ミリ秒です。CDP_ICMP_PING_TIMEOUT の値を変更すると、「現行の応答時間」属性のデフォルトの TIMEOUT 列挙が今後適用されなくなります。TIMEOUT 列挙を、CDP_ICMP_PING_TIMEOUT の新しい値に 3 を掛けた値に変更してください。
CDP_JDBC_CONNECTIONLESS	false	true、false	true に設定すると、データ収集試行の後、毎回 JDBC 接続が閉じられます。つまり、データを収集するごとに、 毎回すべての属性グループが専用の接続を 作成します。この変数を有効にした場合、接続は再利用されません。false に設定すると、 データベースに対して 1 つの接続が作成され、複数の属性グループの間で その接続が共有されます。
CDP_SSH_EXCLUDED_ENVIRONMENT_VARIABLES	なし	環境変数名をコンマで区切った リスト	SSH 対応スクリプト・データ・プロバイダーの場合に、リモート・システムの環境変数に設定しないローカル環境変数のセットを指定します。

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME	0 秒	0、1、または 1 よりも大きな任意の整数。	これを 0 に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS を 1 以上の整数に設定しない場合、エージェントのシャットダウン中に生成されたイベントは処理されません。0 はデフォルトです。 これを 1 に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS を 1 よりも大きい整数に設定しない場合、エージェントのシャットダウン中に生成されたすべてのイベントが処理されます。 これを 1 よりも大きい値に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS を 1 よりも大きい値に設定しない場合、その値 (現在のコンピューター時間の秒単位の値) 内に生成されたイベントが処理されます。例えば、始動時にこの値が 300 に設定されている場合、エージェントは現在時刻で 300 秒以内に生成されるすべてのイベントを処理します。 CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME と CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS の両方の変数に 1 よりも大きい値を入力すると、指定されている期間内に生成されるイベント、または指定されている数のイベントのいずれかが処理されます。選択される変数は、最初に一致した変数です。
CDP_DP_EVENT_LOG_*Windows_Event_Log*_MAX_BACKLOG_TIME	0 秒 (エージェントのシャットダウン中に消失したイベントは処理されません)	0、1、または 1 よりも大きな任意の整数。	以下のように設定される場合

表 *259.* 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 *(続き)*

環境変数	デフォルト値	有効な値	説明
CDP_DP_EVENT_LOG_ MAX_BACKLOG_EVENTS	0 イベント	0、1、または 1 よりも大きな任意の整数。	これを 0 に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME 変数を 1 以上の整数に設定しない場合、エージェントのシャットダウン中に生成されたイベントは処理されません。0 はデフォルトです。 これを 1 に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME 変数を 1 よりも大きい整数に設定しない場合、エージェントのシャットダウン中に生成されたすべてのイベントが処理されます。 これを 1 よりも大きい値に設定し、CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME を 1 よりも大きい値に設定しない場合、エージェントのシャットダウン中に生成されるイベントのうち、指定されている数以下のイベントが処理されます。例えばこの値が 200 に設定されている場合、エージェントの始動時に、始動直前までに生成された 200 個のイベントが処理されます。 CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS と CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME の両方に 1 よりも大きい値を入力した場合、指定されている数のイベント、または指定されている期間内に生成されるイベントのいずれかが処理されます。選択される変数は、最初に一致した変数です。
CDP_DP_EVENT_LOG_ *Windows_Event_Log*_MAX_BACKLOG_ EVENTS	0 イベント (エージェントのシャットダウン中に消失したイベントは処理されません)	0 または 1 以上の整数	以下のように設定される場合
CDP_HTTP_READ_TIMEOUT	10	任意の正の整数	HTTP 要求に対する応答を待機する秒数。
CDP_JAT_THREAD_POOL_SIZE	15	任意の正の整数	Java プロバイダーによりデータ収集要求を処理するために使用されるスレッドの数。このスレッド・プールが役立つプロバイダーは、JMX、JDBC、HTTP、および SOAP データ・プロバイダーです。
CDP_HTML_OBJECTS_THREAD_ POOL_SIZE	10	任意の正の整数	HTTP データ・プロバイダーでモニターされる URL で検出されたページ・オブジェクトをダウンロードするために使用されるスレッドの数。

表 259. 環境変数. 環境変数、そのデフォルト値、有効な値の範囲、および各変数の説明がリストされている表 (続き)

環境変数	デフォルト値	有効な値	説明
CDP_HTTP_SOAP_MAX_ROWS	500	任意の正の整数	HTTP SOAP データ・プロバイダーにより返される行の最大数。
CDP_DP_ALLOW_REMOTE	NO	いいえ、はい	YES に設定すると、エージェントはリモート・ソケット接続を許可します。NO に設定すると、エージェントはローカル・ホストからのソケット接続のみを許可します。NO がデフォルトです。
CDP_DP_INITIAL_COLLECTION_DELAY	可変	任意の正の整数	エージェントが開始した後に、スレッド・プールがそのスケジュールされたデータ収集を開始するまでの秒数。

Watchdog 情報

「**Watchdog 情報**」ページを使用して、エージェント Watchdog の構成情報を指定します。

このタスクについて

「**Watchdog 情報**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションの「**Watchdog 情報**」ノードをクリックします。アウトライン・ビューで「**Watchdog 情報**」ノードを選択することもできます。

Watchdog エージェントの以下の構成情報を指定できます。

- **デフォルトでこのエージェントをモニターする**

  このチェック・ボックスを選択すると、 エージェントのインストール時にエージェントがエージェント管理サービスの管理下に 置かれます。Watchdog によって、エージェントが異常な動作や異常終了をしていないかがモニターされ、再始動されます。

- **確認頻度 (秒)**

  Watchdog が、エージェント・プロセスの正常ではない動作または異常終了を確認する頻度。デフォルトは 180 秒です。

- **最大再始動回数**

  24 時間の期間内に、異常な動作または異常終了のために Watchdog がエージェントを再始動する回数。この回数を超えると、問題がアラートで管理者に通知されます。この期間は、各日午前 0 時に開始されます。このため、エージェントが開始される時点からの最初の期間が短いことがあります。

  再始動が行われるのは、何らかの理由でエージェントがダウンした場合です。Watchdog は、エージェントが応答しなくなった場合や正常ではなくなった場合 (メモリーしきい値を超えた場合など) にもエージェントを停止して再始動します。デフォルトの再始動回数は、24 時間の期間 (午前 0 時から午後 11 時 59 分まで) 中 4 回です。日ごとのエージェント再始動カウントは、午前 0 時に自動的に 0 に戻ります。

- **メモリーしきい値情報**

  エージェントに許可されているエージェント・プロセスの最大サイズ (メガバイト単位)。このサイズを超えると、Watchdog により正常ではないと見なされます。Windows、Linux、および UNIX にはそれぞれ個別の値があります。エージェント・プロセスがこのしきい値を超えると、Watchdog がプロセスを停止してから再始動します。これらのプロパティーにはデフォルトはありません。 値が指定されない場合、Watchdog はプロセス・サイズをモニターしません。メトリックは、Windows では作業セット・サイズを使用し、UNIX と Linux ではユーザー・メモリーを使用します。

Watchdog がエージェントを停止した時点で、最大再始動回数に達している場合、Watchdog はエージェントが再始動回数を超えたという警告を送信し、自動再始動を停止します。エージェントが Tivoli Enterprise Portal など、Watchdog 以外のシステムによって開始されると想定し、Watchdog は引き続きエージェントが稼働しているかダウンしているかを報告します。

「AMS Start Agent」アクション実行コマンドを使用して、エージェントを手動で再始動する必要があります。これにより、再始動カウントがリセットされません。

カウントは次のいずれかの方法でリセットされます (Watchdog は引き続き動作し状況を報告しますが、自動再始動は実行しません)。

- 午前 0 時になった時点。
- ユーザーが「AMS Start Agent」アクション実行コマンドを使用する (入力パラメーター **resetRestartCount** を指定)。 値 1 (「true」つまり「yes」を意味する値) を入力すると、日次再始動カウントが 0 にリセットされます。

詳しくは、「*IBM Tivoli Monitoring* 管理者ガイド」の以下のセクションを参照してください。

- Tivoli System Monitor Agents の場合

  Tivoli システム・モニター・エージェントでのエージェント管理サービスの構成
- Tivoli Enterprise Monitoring Agents の場合

  Tivoli Agent Management Services のインストールおよび構成

Cognos 情報

「**Cognos 情報**」ページを使用して、当該エージェントに対して Cognos データ・モデルが生成されるときに使用する情報を指定します。この情報は、IBM Tivoli Monitoring 環境でのみ使用されます。

手順

1. 「**Cognos 情報**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションにある「**Cognos 情報**」をクリックするか、アウトライン・ビューの「**Cognos 情報**」ノードをクリックします。
2. **データ・ソース**フィールドで、Tivoli Common Reporting を IBM Tivoli Data Warehouse に接続するデータ・ソースの名前を入力します。

   デフォルト値は TDW です。
3. 「**スキーマ**」フィールドに、Tivoli Data Warehouse で使用するデータベース・スキーマの名前を入力します。これは、Cognos のレポートでテーブル名を完全修飾するために使用します。

   デフォルト値は ITMUSER です。この値は、生成された Cognos モデルを Framework Manager にロードするときに Framework Manager で変更できます。

   「**データ・ソース定義**」ページ の「**この属性グループをレポート・カテゴリーに追加します**」チェック・ボックスは、属性グループを Cognos モデルのどこに配置するかを決定します。チェック・ボックスが選択されていない場合、属性グループは Cognos モデルの拡張属性フォルダーに配置されます。選択した場合、属性グループは「主要メトリック」フォルダー内の選択されたサブフォルダー (可用性またはパフォーマンス) に配置されます。データ・ソース・フィールドの詳細については、1193 ページの表 260 を参照してください。

次のタスク

Cognos データ・モデルを使用して、エージェントの Tivoli Common Reporting レポートを作成できます。1473 ページの『Cognos データ・モデルの生成』を参照してください。

「エージェントの生成ウィザード」リンク

新しいエージェントの作成または編集が完了したら、「エージェントの生成」ウィザードを使用してインストールを準備します。

手順

- 新しいエージェントの作成または編集が完了したら、**Agent Editor** の「**エージェント情報**」ページで、「**エージェントの生成**」**ウィザード**のリンクをクリックします。

  エージェントの生成ウィザードでは、以下のことが可能です。

– ローカル・システムの Tivoli Monitoring インストール環境で エージェント・ファイルを生成する。説明については、1385 ページの『エージェントのローカル・インストール』を参照してください。
– パッケージを作成し、エージェントを他のシステムにインストールできるようにする。 説明については、1387 ページの『エージェント・パッケージの作成』を参照してください。

「データ・ソース定義」ページ

「**データ・ソース定義**」ページを使用して、データ・ソースを操作します。

このタスクについて

「**データ・ソース定義**」ページには、エージェント用に構成されているデータ・ソースがリストされています。ツリーでデータ・ソースまたは属性を選択すると、ページが更新され、選択したオブジェクトのプロパティーが表示されます。 それぞれのフィールドを使用して、選択したデータ・ソースまたは属性のプロパティーを変更します。

注 : さまざまなデータ・プロバイダーからデータ・ソースを作成するための詳細な手順については、1218 ページの『データ・ソースの定義およびテスト』を参照してください。

手順

- 「**データ・ソース定義**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションの「**データ・ソース**」をクリックするか、または「**アウトライン**」ビューの「**データ・ソース**」ノードをクリックします。
- 「**選択対象に追加**」をクリックするか、ナビゲーション・ツリー内で右クリックしてオプションのいずれか 1 つを選択することによって、データ・ソースを追加することができます。
- データ・ソースまたは属性を削除するには、それらを右クリックして、「**削除**」を選択します。
- 属性を追加、変更、および削除できます。説明については、1192 ページの『データ・ソースおよび属性のプロパティーの編集』を参照してください。

「データ・ソース定義」ページを使用したデータ・ソースのコピー

「**データ・ソース定義**」ページを使用して、データ・ソースをコピーします。

始める前に

「**データ・ソース定義**」ページに移動します。 詳しくは、1189 ページの『「データ・ソース定義」ページ』を参照してください。

このタスクについて

属性グループになるデータ・ソースはクリップボードにコピーし、このエージェントまたは別のエージェントに貼り付けることができます。属性グループにならないデータ・ソースは、「可用性」および「Windows イベント・ログ」のデータ・ソースです。

手順

1. コピーする属性グループを選択します。
2. 以下のいずれかの方法を使用して、属性グループを切り取るかコピーします。
   - メニュー・バーから「**編集**」 > 「**切り取り**」 > 「**編集**」 > 「**コピー**」をクリックする。
   - 選択する項目のいずれかを右クリックし、メニューから「**切り取り**」または「**コピー**」をクリックする。
   - オペレーティング・システムまたは Eclipse の、切り取りまたはコピーのアクションを呼び出すいずれかのキー・ストロークを使用する。例えば、Windows システムの場合は **Ctrl-C** を押すとコピー・アクションが呼び出されます。

   データ・ソースを既存の場所から削除してクリップボードに入れるには、「**切り取り**」を使用します。データ・ソースをそのまま残してクリップボードにコピーするには、「**コピー**」を使用します。

3. 属性グループの親 (エージェント、サブノード、またはナビゲーター・グループ) を選択するか、 既存の属性グループを選択します。
4. 以下のいずれかの手順を実行して、選択内容を貼り付けます。
   - メニュー・バーから「**編集**」 > 「**貼り付け**」を選択する。
   - ツリーで、選択内容を貼り付けるノードを右クリックし、メニューの「**貼り付け**」をクリックする。
   - オペレーティング・システムまたは Eclipse の、貼り付けのアクションを呼び出すいずれかのキー・ストロークを使用する。例えば、Windows システムの場合は **Ctrl-V** を押すと貼り付けアクションが呼び出されます。

タスクの結果

クリップボードの属性グループが、選択された親に入れられます。代わりに、属性グループが選択された場合、属性グループは、選択した属性グループの親に入れられます。

貼り付けるときに別の属性グループと名前が競合する場合は、 貼り付ける属性グループの名前が少し変更されて 競合が回避されます。

「ランタイム構成情報」ページ

「**ランタイム構成情報**」ページには、エージェントの構成可能変数が表示されます。モニター対象ホストにエージェントをインストールする際に、変数の値を設定できます。

これらの値は、環境を介してコマンドの戻りコードおよびスクリプトから使用することができます。「**ランタイム構成情報**」ページを開くには、「**エージェント情報**」ページの「**エージェント・コンテンツ**」セクションの「**ランタイム構成**」をクリックするか、またはアウトライン・ビューの「**ランタイム構成**」ノードをクリックします。Agent Builder は、製品コードおよびラベルから自動的に環境変数の名前を構成します。

「**ランタイム構成情報**」ページを使用して、構成プロパティーの追加と変更、およびデフォルト値の指定を行うことができます。

「エージェント XML エディター」ページ

「**エージェント XML エディター**」ページには、エージェント定義の XML が表示されます。

エージェント定義 XML に含まれる情報は、Agent Builder の他のすべての部分に表示されます。 XML を変更すると、その変更は Agent Builder に表示される情報に反映されます。

重要 : XML で変更を行わないでください。そのような変更で発生したエラーが原因となって、エージェントの生成ができなくなったり、エージェントの機能に悪影響が及ぶ可能性があります。

編集内容および変更内容の保存

エディターを使用して変更した内容は、保存操作を行うまで保存されません。

手順

- 以下のいずれかの方法で保存操作を実行します。
  - 「**ファイル**」 > 「**保存**」を選択するか、「保存」(ディスケット) アイコンをクリックする 。
  - **Ctrl+S** を押す。

  保存時に、情報がすべて入力されていることを確認するために検証が実行されます。問題が発生した場合は、Eclipse の「**問題**」ビューにエラーに関する情報が表示されます。このビューが表示されない場合は、「**ウィンドウ**」 > 「**ビューの表示**」 > 「**問題**」と選択します。エラーのあるエージェントを生成しようとすると、エラー・メッセージが表示されます。

  注 : エージェントを生成してインストールするには、すべてのエラーを訂正して変更内容を保存しておかなければなりません。

エージェントのバージョンのコミット

このバージョンのエージェントの開発が完了したことがはっきりしており、配信の準備が整っている場合は、エージェントをコミットします。

このタスクについて

IBM Tivoli Monitoring システムでは、新しいバージョンのエージェントが、モニター環境で使用されていたそのエージェントの以前のバージョンに含まれる情報をすべて含んでいる必要があります。新しいエージェントをあるモニター対象ホストにインストールしたが、他のモニター対象ホストに以前のエージェントが残っている場合、以前のバージョンのすべての情報を含めることは、ワークスペース、シチュエーション、および照会が引き続き機能するための要件になります。

エージェントの開発およびテストが完了したら、特定のバージョン番号に対する最終バージョンとしてエージェントをコミットする必要があります。Agent Builder により、エージェントのコミット後に情報が削除されないことが保証されます。エージェントの後続のビルドでは新しいバージョン番号が設定されます。

バージョンの数は 1024 に制限されています。

要確認 : IBM Cloud Application Performance Management 環境でテストが行われて実行されるエージェントに変更を加える場合、そのエージェントのバージョンを変更する必要があります。

手順

1. 「**Agent Editor**」ウィンドウの 「**エージェント情報**」ページを開きます。
2. 「**エージェント・バージョンのコミット**」領域で、「**このレベルをコミットする**」をクリックします。
3. コミットしたエージェントをバックアップするか、ご使用のバージョン管理システムにチェックインします。

次のタスク

エージェントをコミットした後でエージェントに対して追加変更を行うと、その変更は新しいバージョンの一部になります。新しいバージョン番号を入力しなければ、追加の変更を保存することはできません。新しいバージョンに対する変更は、以前のバージョンのエージェントとの互換性を維持しなければなりません。

エージェントをコミットした後で、エージェントのコミット前から存在したオブジェクトに対して以下のアクションを実行することはできません。

- 属性グループからの属性の削除。
- 属性グループの削除。
- 属性グループ内の既存属性の再配列。
- 既存の属性グループの 再編成 (ナビゲーター項目を使用)。
- サブノードへのまたはサブノードからの 属性グループまたはナビゲーター・グループの移動。
- 属性グループの名前変更。
- 属性の名前変更。
- 既存属性のデータ・タイプの変更。
- エージェントのコミット前から存在した属性グループを含むサブノード名またはタイプの変更。
- エージェントの会社 ID またはエージェント ID の変更。
- エージェントの製品コードの変更。詳しくは、1192 ページの『製品コードの変更』 を参照してください。

以下のアクションは、エージェントのコミット後でも実行できます。

- 既存の属性グループへの新規属性の追加。
- 新規属性グループの追加。
- 新規属性の再配列。

- ナビゲーター項目による新規属性グループの編成。
- 新規サブノード・タイプの作成。
- 新規照会の追加。
- 新規シチュエーションの追加。
- 新規ワークスペースの追加。

エージェントの新しいバージョン番号の設定

コミットしたエージェントへの変更を保存するには、新しいバージョン番号を入力する必要があります。

手順

1. 「**Agent Editor**」ウィンドウの 「**エージェント情報**」ページを開きます。
2. 「バージョン」のプロンプトの後に、現行レベルよりも高いバージョン、フィックスパック、またはパッチ・レベルを入力します。
3. エージェントを編集します。

   ヒント: エージェントをコミットしたがエージェント・バージョンの変更を忘れた場合は、変更を保存しようとすると、新しいバージョンの入力を求めるプロンプトが表示されます。

製品コードの変更

製品コードを変更した場合、エージェントはそのエージェントの以前のどのバージョンとも非互換になります。以前のコミット・アクションのレコードはすべて失われ、新しいエージェントを開発することになります。

IBM Tivoli Monitoring からエクスポートしてエージェントにインポートしたファイル、シチュエーション、アクション実行コマンド、またはワークスペースがすべてエージェントから削除されます。

コミットされたエージェントの製品コードを変更しようとすると、Agent Builder から警告が出され、続行するかどうかを尋ねられます。

「**エージェント製品コード**」ウィンドウで「**はい**」をクリックすると、エージェント・サポート・ファイルの内容が無効になっているという警告が出されます。また、次にエージェントが保存されるときにそのファイルが削除されるという警告も出されます。

データ・ソースおよび属性のプロパティーの編集

エージェントにデータ・ソースを追加すると、Agent Builder は対応するデータ・セットを作成します。そのデータ・セットおよびそのデータ・セット内の属性を編集し、必要なモニター情報を指定することができます。

手順

データ・セット (属性グループ) の情報を編集または削除するには、以下のようにします。

1. 「**エージェント情報**」ページの「**エージェント・コンテンツ**」領域で、「**データ・ソース**」をクリックします。

   「**データ・ソース定義**」ページが開きます。
2. データ・セット (属性グループ) を選択します。

   ページの属性グループ情報域が更新され、選択したデータ・セットのプロパティーが表示されます。

   注: **エージェント**・ウィザードの最後のページであれば、データ・ソースをダブルクリックして、「**属性グループ情報**」ウィンドウを開くこともできます。このウィンドウには、「**データ・ソース定義**」ページの属性グループ情報域と同じ情報が表示されます。

   1193 ページの表 260 では、フィールド情報のうち、すべてのデータ・ソースに当てはまるものについて説明しています。それぞれのフィールドを使用して、選択したデータ・ソースまたは属性のプロパティーを変更します。

表 260. データ・ソース編集用フィールド		
フィールド名	**説明**	**許容値および例**
属性グループ名	Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールで表示されるデータ・ソースの名前	許容値: 32 文字以下の長さの、説明のストリング。エージェントの範囲内で固有である必要があります。先頭文字は英字でなければなりません。残りの文字は英字、数字、または下線にすることができます。下線はスペースとして表示されます。スペースまたは特殊文字を使用しないでください。
ヘルプ・テキスト	データ・ソースのヘルプ・テキスト	許容値: 最大で 256 文字の長さのストリング。
単一データ行の生成	データ・ソースは 1 つのデータ行を返します。すべてのサンプル・データ・ソースで編集可能です。	例: 物理システム・メモリーをモニターする場合は、単一行を選択します。一般にシステムはすべてのメモリーを単一のプール内で管理します。これにより、1 つのデータ行のみが返されます。
複数のデータ行を生成可能	データ・ソースは任意の数のデータ行を返せます。すべてのサンプル・データ・ソースで編集可能です。	例: システム内に複数のディスクが存在する可能性があるため、ディスク・ドライブをモニターする場合は、複数行を選択します。キーについては、1 つのディスクを別のディスクと区別する属性を選択します。ディスクの場合、キー属性はディスク番号、ドライブ名、ボリューム・ラベル、または環境内で適切なその他の属性です。
イベントの生成	データ・ソースがイベントに基づくデータを返します (1 イベントにつき 1 データ行)。	例: SNMP イベントに基づくデータ・ソースは、パフォーマンスしきい値を超えると通知 (トラップ) を送信します。 **注 :** すべてのデータ・ソースがイベントを生成できるわけではありません。
この属性グループをレポート・カテゴリーに追加します	この属性グループ内の属性を割り当てる先の、生成される Cognos モデル内のカテゴリー。	チェック・ボックスを選択し、「主要メトリック」フォルダーの選択されたサブフォルダー (「可用性」または「パフォーマンス」) に属性グループを配置します。 チェック・ボックスが選択されていない場合、属性グループは、Cognos データ・モデルの「**拡張メトリック**」フォルダーに配置されます。
メトリック・カテゴリー	この属性グループ内の属性を割り当てるカテゴリー。	「**パフォーマンス**」または「**可用性**」を選択します。

注 :

a. **単一データ行の生成**および**複数のデータ行を生成可能**フィールドは、イベント・データ・ソースのデータには影響しません。
b. サンプル・データ・タイプおよびイベント・データ・タイプについて詳しくは、1213 ページの『データ・タイプ』を参照してください。
c. 特定のデータ・ソースのフィールドについては、1218 ページの『データ・ソースの定義およびテスト』で該当するデータ・プロバイダーの情報を参照してください。

属性の作成、変更、および削除

データ・セット (属性グループ) 内の属性を作成、変更、または削除することができます。

属性を処理するには、「**データ・ソース定義**」ページを開きます。詳しくは、1189 ページの『「データ・ソース定義」ページ』を参照してください。

属性の作成

データ・セットに新規属性を追加することができます。

手順

1. データ・ソースを右クリックして、メニューの「**属性の追加**」を選択します。
   「**属性情報**」ページが表示されます。

   注 : 表示されるページは、属性のデータ・ソースによって 異なります。
2. 「**属性情報**」ページで、選択する新規属性の情報を指定します。

   フィールドおよびオプションについては、1197 ページの『属性の定義に関するフィールドおよびオプション』を参照してください。
3. 属性を追加するには、「**追加属性の追加**」を選択して「**次へ**」をクリックします。
4. 属性の追加が完了したら「**完了**」をクリックします。

属性のコピー

「**データ・ソース定義**」ページから属性をコピーできます。

手順

1. Agent Editor の「**データ・ソース定義**」ページで、コピーする属性を右クリックし、「**属性のコピー**」をクリックします。
2. 「**属性のコピー**」ウィンドウの「**名前**」フィールドに新規属性の名前を入力し、「**OK**」をクリックします。

属性の編集

「**データ・ソース定義**」ページを使用して、属性情報を編集および変更できます。

手順

1. 編集する「属性」を選択します。
   ページの「**属性情報**」ペインが更新され、選択した属性のプロパティーが表示されます。
2. 新規属性の情報を指定します。

   注 : エージェント・ウィザードの最後のページ (「**データ・ソース定義**」ページ) で、属性をダブルクリックして、「**属性情報**」ウィンドウを開くことができます。 このウィンドウには、「**データ・ソース定義**」ページの「属性情報」ペインと同じ情報が表示されます。

派生属性の作成

属性を作成する際に、データ・ソースから直接値を派生させるのではなく、 他の属性から値を派生させることができます。

このタスクについて

派生属性では、ソース属性の値に対する操作を実行することができます。例えば、数値属性に対する基本的な算術演算、ストリング属性に対するストリング連結を実行できます。

派生式に使用される基本式構文には関数 が含まれています。 これらの関数は、短期集約、ストリングから整数への変換、構成プロパティーおよび環境変数へのアクセスなど、より複雑なデータ操作を提供します。また、エディターによって式の作成中に式が 視覚化されます。

手順

1. 「**データ・ソース定義**」ページで、データ・ソースを右クリックして、「**属性の追加**」をクリックします。
2. 「**属性情報**」ページで、属性名とヘルプ・テキストを入力します。
3. 「**他の属性値から派生**」を選択します。
4. 「**式**」フィールドに式のテキストを入力するか、 「**編集**」をクリックしてグラフィカル・エディターで式を入力します。

   式で使用できる演算子および関数については、1206 ページの『数式演算子および関数』を参照してください。

   注 : 「**編集**」をクリックすると、式エディターが開きます。派生属性の編集については、1196 ページの『派生属性の編集』を参照してください。
5. オプション: 「**間隔の特定の計算**」チェック・ボックスを選択またはクリアして、関数の計算時に使用する 2 つの属性サンプル値を決定します。

   式に `rate` または `delta` 関数を使用する場合は、このオプションを使用します。「**間隔の特定の計算**」について詳しくは、1195 ページの『間隔の特定の計算』を参照してください。`rate` および `delta` 関数について詳しくは、1206 ページの『数式演算子および関数』 を参照してください。
6. 「**属性タイプ**」領域で、属性のタイプをクリックします。
7. 「**OK**」をクリックします。

   データ・ソースが以前と同じようにリストされた、「**データ・ソース定義**」ページが再度表示されます。
8. 「**完了**」をクリックします。

   重要 : 他の派生属性を参照する派生属性を作成する場合は、参照される属性が新規属性よりも前にリストされているようにします。属性が、リストの後の方に位置する別の派生属性を参照していると、エージェントはこの属性の値を表示できません。そのような属性を作成する場合、Agent Builder が警告を表示します。

間隔の特定の計算

`rate` または `delta` 関数に基づく派生属性の定義時には、「**間隔の特定の計算**」を選択できます。

「**属性情報**」ページの「**派生属性の詳細**」タブで「**間隔の特定の計算**」を選択します。詳しくは、1195 ページの『派生属性の作成』を参照してください。

「**間隔の特定の計算**」を選択して使用する場合は、差分 (つまり属性値間の違い) の概念を理解しておくことが重要です。差分とは、属性の最新の値とその属性の直前の値との差です。差分は、`delta` 関数から直接返され、`rate` 関数で結果を計算するために使用されます。

`delta` 関数または `rate` 関数には、唯一の引数として常に `last` 関数を指定する必要があります。`last` 関数は、差分の決定に使用する属性の値を指定します。「**間隔の特定の計算**」を選択していない場合は、直前に使用された値が常に 2 番目に最新の値になります。「**間隔の特定の計算**」を選択した場合、直前に使用された値は、経過時間 (最新の値に対する) が要求側の収集間隔に等しい値となります。

例えば、`CDP_DP_REFRESH_INTERVAL` が 120 秒に設定され、`attribute A` に以下のサンプル値が指定されているとします。

時刻	サンプル値
現在	2800
2 分 (120 秒) 前	2600
4 分 (240 秒) 前	2499
6 分 (360 秒) 前	1500
8 分 (480 秒) 前	1200
10 分 (600 秒) 前	1000

「**間隔の特定の計算**」を選択していない場合、delta 関数は常に 200 を返します。この値は、2 つの最新の値の差分 (2800 - 2600) です。値が Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールで表示されるか、シチュエーションで使用されるか、またはヒストリカル収集で使用されるかに関係なく、同じ値が返されます。

「**間隔の特定の計算**」を選択した場合、delta 関数は、要求側の収集間隔に応じた値を返します。

delta 関数を指定した派生属性を収集間隔が 4 分のシチュエーションで使用する場合、delta 関数が返す値は 301 です。この値は、最新の値とその 4 分前に取得された値の差分 (2800 - 2499) です。

rate 関数を指定した派生属性を収集間隔 10 分 (600 秒) のシチュエーションで使用する場合、rate 関数が返す値は 3 です。この値は、最新の値とその 10 分前に取得された値の差分を間隔の秒数で除算したもの ((2800 - 1000) / 600) です。

注 : Tivoli Enterprise Portal には固有の収集間隔がないため、Tivoli Enterprise Portal 要求の差分および比率の計算には、常に最新の値および 2 番目に最新の値の属性値が使用されます。そのため、「**間隔の特定の計算**」を選択したかどうかに関係なく、結果は同じになります。

「**間隔の特定の計算**」を指定した場合に delta または rate が正しく機能するには、以下の条件を満たす必要があります。

- エージェントがバックグラウンドで定期的にデータを収集する必要があり、オンデマンドであってはなりません (CDP_DP_THREAD_POOL_SIZE が 0 より大きくなければなりません)。
- 属性が使用されるシチュエーション収集またはヒストリカル収集の各間隔が、バックグラウンドの最新表示間隔 (CDP_DP_REFRESH_INTERVAL) の倍数でなければなりません。
- count (last 関数の 2 番目の引数) には、シチュエーション収集またはヒストリカル収集の最大収集間隔が収まる十分な大きさが必要です。例えば、エージェントが 10 分 (600 秒) のヒストリカル収集をサポートしなければならず、CDP_DP_REFRESH_INTERVAL が 120 秒である場合、count は少なくとも 6, 1+(600 / 120) でなければなりません。count 値が 6 の場合、last 関数は最新のサンプルと 600 秒前までのサンプルを返すことになります。

注 : これらの条件が満たされない場合、入力値は恐らく無効になり、0 という結果が返されます。

派生属性の編集

式エディターを使用して、派生属性を編集します。

式エディターは、派生属性に対して「**属性情報**」ページで使用することができます (1195 ページの『派生属性の作成』を参照してください)。式エディターについて詳しくは、1202 ページの『式エディター』を参照してください。

属性の削除

「**データ・ソース定義**」ページを使用して、データ・セットから 1 つ以上の属性を削除することができます。

手順

- 属性を削除するには、1 つ以上の属性を右クリックして、表示されたメニューから「**削除**」を選択します。

  注 : 派生属性によって使用されている属性は削除できません。最初に、削除する属性への派生属性からの参照を削除する必要があります。

属性の定義に関するフィールドおよびオプション

「**属性情報**」ページのフィールド情報およびオプションのうち、すべてのデータ・ソースに当てはまるものについての説明。

各データ・ソースに固有のフィールド情報については、各データ・ソースの関連資料を参照してください。

表 *261.* 属性の定義に関するフィールドおよびオプション

フィールド名/オプション	説明	許容値
属性名	Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに表示される属性の名前	以下の文字によるストリング • A-Z • _ • a-z • 0-9 **注 :** 名前の先頭は、A から Z または a から z のいずれかにする必要があります。 属性名の最大長は 63 文字、属性グループ名の最大長は 63 文字です。
ヘルプ・テキスト	属性のヘルプ・テキスト。	ストリング
非表示 - 派生属性でのみ使用可能	選択した場合、属性は Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに表示されません。 最終行の注を参照してください。	適用外
他の属性値から派生	属性値は、他の属性の値から計算されます。	適用外
キー属性	属性は表内のキーです。 レポート対象のオブジェクトを一意的に定義するためにこの属性が役立つかどうかを確認してください。 データがウェアハウス化および要約される場合、キー属性は要約テーブルでデータをロールアップするために使用されます。	このオプションは、Perfmon 属性では使用できません。

表 261. 属性の定義に関するフィールドおよびオプション (続き)		
フィールド名/オプション	**説明**	**許容値**
「**属性情報**」ペイン	このタブの内容は、この属性が属するデータ・ソースのタイプによって異なります。詳しくは、モニター対象のデータ・ソースに関する章の情報を参照してください。 派生属性の場合、「**式**」フィールドに、他の属性または定数に基づいた属性の値を計算するための式を入力します。「**式**」フィールドに式を入力するか、「**編集**」をクリックしてグラフィカルな式エディターを 使用することができます。1202 ページの『式エディター』 を参照してください。	
属性タイプ	Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールでの属性の表示方法を示します。 次の 3 つのタイプがあります。 • ストリング • 数値 • タイム・スタンプ 属性タイプについて詳しくは、1198 ページの『属性タイプ』 を参照してください。	1199 ページの表 262 には、数値属性タイプの値の説明があります。
列挙	スケールがゼロの数値またはストリング値にすることができます。	1201 ページの『属性の列挙の指定』 の手順を実行して、列挙を表に追加します。 エージェントから属性に対応する値を受信すると、Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに列挙名が表示されます。 この属性は、指定の意味 (例えば、1=UP、2=DOWN など) を持つ一連の固有値に関して使用します。
注 : 計算の中でこの属性が他の属性と共に使用されている場合、さまざまな理由から基本値は表示されません。例えば、バイト・カウントを 表す数は非常に早く切り替わるため、あまり役に立ちません。		

属性タイプ
3 つの属性タイプがあります。

以下の 3 つの属性タイプがあります。

- ストリング
- 数値
- タイム・スタンプ

ストリング属性

「**ストリング**」を選択する場合は、「**最大サイズ**」フィールドを使用して、ストリングの最大長をバイト単位で指定します。デフォルトのサイズは 64 バイトです。

ストリング値には任意の UTF-8 文字を含めることができます。最大サイズは、そのストリングを入れるために割り振られたバッファーのバイト単位の全体長です。 一部の非 ASCII UTF-8 文字は 1 バイトより大きいので、最大サイズを選択するときにはこのことを考慮する必要があります。 ウェアハウス内のデータ集約では、その期間内に収集された最新の値が表示されます。

数値

「**数値**」を指定する場合は、複数のオプションを設定できます。これらのオプションについては、1199 ページの表 262 を参照してください。

タイム・スタンプ

タイム・スタンプ属性は、CYYMMDDHHMMSSmmm (C=1 は 21 世紀を表す) 形式に準拠する形式のストリング属性です。スクリプトまたはソケット・クライアントでは 16 文字すべてを使用する必要があります。タイム・スタンプ属性タイプは、Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールで表示される際に、そのロケールに対して適切な形式で表示されます。

WMI 用にブラウズ・フィーチャーを使用している場合、Agent Builder は、CIM_DATETIME という CIM タイプを持つ属性を、自動的にタイム・スタンプとしてマークします。 データ・プロバイダーは、WMI の属性を自動的にこの形式に変換します。

属性の数値の特性
属性のサイズ、目的、スケール、および範囲についての説明。

数値属性を指定する場合は、属性のサイズ、目的、スケール、および範囲を指定する必要があります。詳しくは、1199 ページの表 262 を参照してください。

表 *262.* 数値属性オプション

数値の特性	オプションおよびフィールド	説明
サイズ	**32 ビット** **64 ビット**	32 ビット数値の範囲は -2147483648 から 2147483647 まで (概数では -2,000,000,000 から 2,000,000,000 まで) です。 64 ビット数値の範囲は -9223372036854775808 から 9223372036854775807 まで (概数では $-9x10^{18}$ から $9x10^{18}$ まで) です。

表 262. 数値属性オプション (続き)		
数値の特性	**オプションおよびフィールド**	**説明**
目的	**ゲージ**	返された未加工の値が以前の値よりも大きいまたは小さい場合の整数値。負の値がサポートされています。 整数については、このタイプがデフォルトのタイプです。ウェアハウス内のデータ集約では、最小値、最大値、および平均値が示されます。
	カウンター	通常は時間の経過と共に増加していく未加工値を含む正の整数値。ウェアハウス内のデータ集約では、デルタの合計値、最高値、最低値、および最新値が表示されます。以下に示すデルタに基づく計算例では、1 時間の詳細データ値が 9、15、12、20、22 で、デルタ・ベース処理に次の規則が適用されます。 • 現行値が直前の値以上である場合は、出力は現行値から直前の値を引いた値になります。 • 現行値が直前の値未満である場合は、出力は現行値になります。 • 15 は 9 よりも大きいため、出力は 6 になります。 • 12 は 15 よりも小さいため、出力は 12 になります。 • 20 は 12 よりも大きいため、出力は 8 になります。 • 22 は 20 よりも大きいため、出力は 2 になります。 • TOT_ 値 (出力の合計) は 28 です。 • LOW_ 値 (最低出力) は 2 です。 • HI_ 値 (最高出力) は 12 です。
	プロパティー	頻繁には変更されないオブジェクトのプロパティー。ウェアハウス内のデータ集約では、その期間内に収集された最新の値が表示されます。
	差分	この属性の現在の値と過去の値の差を表す整数値。 この属性はウェアハウス内のゲージとして表されるため、ウェアハウス内のデータ集約では、最小値、最大値、および平均値が示されます。
	割合の変化	現行値と過去値の間での % の変化を表す整数値。 このタイプは、`((new -old)*100)/old` という式で計算されます。このタイプはウェアハウス内のゲージとして表されるため、ウェアハウス内のデータ集約では、最小値、最大値、および平均値が示されます。
	変化率	現在の値と過去の値との差を、サンプル間の秒数で除算した整数値。これは、値 (バイト数など) を秒当たりの値 (バイト/秒など) に変換するものです。 このタイプはウェアハウス内のゲージとして表されるため、 ウェアハウス内のデータ集約では、最小値、最大値、および平均値が示されます。
スケール	**小数部の調整**	スケールは、数値の小数点以下の桁数を決定します。小数点以下の桁数が 1 つ増えるごとに、上記で説明した範囲が 10 分の 1 に減少します。例えば、「小数部の調整」の値 2 は、小数点以下の桁数が 2 桁であることを示し、32 ビット数値の許容範囲は `-21474836.48` から `21474836.47` までになります。 「小数部の調整」にゼロ以外の値が指定されると、数値は内部的に浮動小数点数として操作されます。したがって、大きな 64 ビット数値の精度は低減することがあります。

表 262. 数値属性オプション (続き)		
数値の特性	**オプションおよびフィールド**	**説明**
範囲	**最小** **最大**	範囲は、予期される値の範囲を指定します。最小範囲と最大範囲を指定しなかった場合は、上記で説明した最大値が使用されます。範囲は、一部のグラフィカル Tivoli Monitoring ワークスペース・ビュー内で、より実用的な初期表示を生成するために使用されます。

属性の列挙の指定
「**属性情報**」ページを使用して値の列挙を指定します。

このタスクについて
属性に列挙を指定するには、若干の手順を要します。定義済みの列挙を持つ値が見つかった場合には、その値ではなく列挙名が Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに表示されます。

手順

1. 「**属性情報**」ページの「**属性タイプ**」領域で、「**数値**」をクリックします。
2. 「**列挙**」領域で、列挙をクリックし、「**追加**」をクリックします。
   「**列挙の定義**」ウィンドウが表示されます。
3. ウィンドウ内のフィールドに、列挙の名前と値を入力します。
4. 「**OK**」をクリックします。
   続いて列挙を追加することができます。

状況標識として使用される属性の重大度の指定
IBM Cloud Application Performance Management 環境では、要約ダッシュボードで状況を表示する必要があります。属性を使用して状況の値を提供する必要があります。この属性に対して、特定の状況の重大度を示す値を指定してください。

このタスクについて
状況標識に使用する属性は数値でなければなりません。「**ダッシュボード・セットアップ**」ウィザードでこの属性を選択します。このウィザードの使用については、1373 ページの『Cloud APM 用のエージェントの準備』を参照してください。

「正常」、「警告」、「重大」の各重大度に対応する属性の値を指定できます。他のすべての値は「不明」の重大度状況を示します。いくつかの値を明示的に「未定義」として定義し、これらの値に対してユーザー・インターフェースで 「不明」状況として表示できます。

手順

1. 編集する「属性」を選択します。
   ページの「属性情報」ペインが更新され、選択した属性のプロパティーが表示されます。
2. 「属性情報」ペインで、「**重大度**」タブをクリックします。
3. 必要な重大度 (正常、警告、重大、未定義) を選択し、「**編集**」をクリックします。
4. 「**範囲**」または「**単一の数値**」を選択し、値の範囲または 1 つの数値を入力して「**OK**」をクリックします。
5. オプション: 同一の重大度に対して別の値を追加する必要がある場合 (2 と 25 の両方が「警告」を示す場合など) は、「**追加**」をクリックし、重大度を選択し、値を入力してから、「**OK**」をクリックします。

属性グループのフィルタリング

サンプル・データを返す属性グループから返されるデータを制限するフィルターを作成できます。

始める前に

属性グループが既に存在する場合は、「**データ・ソース定義**」ページを開きます。詳しくは、1189 ページの『「データ・ソース定義」ページ』を参照してください。

属性グループを作成する場合は、1174 ページの『初期データ・ソースの定義』のステップに従って、初期データ・ソース情報ページで「**拡張**」をクリックします。

手順

1. 以下のいずれかのステップを実行して、フィルターの作成を開始します。
   - 属性グループを作成している場合は、初期データ・ソース情報ページで「**拡張**」をクリックします。
   - 属性グループが存在する場合は、「**データ・ソース定義**」ページでその属性グループを選択し、「**データ・ソース定義**」ページで「**拡張**」をクリックします。
2. 「**詳細なデータ・ソース・プロパティー**」ページで、選択式を入力します。入力する選択式は、ブール値の結果 (true または false) に評価される必要があります。

   「**詳細なデータ・ソース・プロパティー**」ページで 「**編集**」をクリックすると、 式エディターを使用して式の入力や変更を行えます。式エディターについて詳しくは、1202 ページの『式エディター』を参照してください。
3. フィルター選択式の入力が終わったら、「**データ・ソース定義**」ページに戻るまで「**OK**」をクリックします。

   フィルターが作成されると、エージェントがフィルターを 使用して各データ行を評価します。あるデータ行でフィルターが *true* に評価されると、そのデータが IBM Tivoli Monitoring または IBM Cloud Application Performance Management に送信されます。フィルターが *false* に評価された場合、そのデータ行は送信されずに破棄されます。

次のタスク

属性グループのテスト 機能を使用して、意図したとおりにフィルターが機能しているかどうかを検証できます。属性グループのテストについて詳しくは、1376 ページの『属性グループのテスト』を参照してください。

式エディター

式エディターを使用して、Agent Builder 内の式を作成および変更します。

式エディターは、グラフィック・ツールであり、以下のいずれかのタスクを実行するときに表示されます。

1. 派生属性の作成または編集 (1195 ページの『派生属性の作成』および 1196 ページの『派生属性の編集』を参照してください)
2. フィルタリング属性グループの作成 (1341 ページの『フィルタリング属性グループの作成』を参照してください)
3. 属性グループからのデータのフィルター 処理 (1202 ページの『属性グループのフィルタリング』を参照してください)

重要 :

- 派生属性を作成するときには、作成する式の結果のデータ・タイプが属性のタイプに一致しなければなりません。例えば、派生属性のタイプが数値である場合は、作成する式の結果も数値に評価されなければなりません。
- フィルタリング属性グループを作成するとき、または属性グループのデータをフィルターに 掛けるときは、作成する式の結果がブール値 (「true」 または 「false」) にならなければなりません。

注 : 以下のビューでは、式エディターが表示され、派生属性の式を作成しています。フィルタリング属性グループで式エディターを使用するとき、または式エディターを使用して属性グループのデータをフィルタ

ーに掛けるときのビューは同じです。ビューの見出しには、使用状況に応じて「**派生式エディター**」または「**式エディターのフィルター**」と表示されます。

式エディターが表示されるときに、このエディターに現行の式が読み込まれます。式が存在しない場合は、「**式エディター**」ウィンドウで式スペースに式を直接入力できます。代わりに、「**挿入**」をクリックし、エディターのメニュー・オプションを使用して式を入力することもできます。エディターのデフォルト・ウィンドウには式の 2 つのビューが表示されます。また、オプションでもう 1 つ (3 番目) のビューも表示できます。

「**コンポーネント**」ビュー **(デフォルト)**
編集する式のコンポーネントが「**オペランド**」領域および「**演算子**」フィールドに表示されます。選択メニューを使用して、この演算子と 2 つのオペランドを編集できます。

「**式**」ビュー **(デフォルト)**
ウィンドウにある式フィールドには、完全な式が表示されます。式を編集するには、このボックスに式を入力します。

「**式**」階層ツリー・ビュー **(オプション)**
「**式の階層の表示**」チェック・ボックスを選択すると、式の階層ツリーが表示されます。これ以降、式エディターの起動時にはこのチェック・ボックスの状態が記憶されます。

式エディターのコンポーネント・ビューの変更

式エディターのコンポーネント・ビューを変更します。

このタスクについて

コンポーネント・ビューに表示されるコンポーネントは、次の方法で変更できます。

手順

- 式テキストでカーソルを移動します。
- 式階層ツリーで異なるノードを選択します。
- 「**1 レベル上へ**」またはいずれかの編集ボタンを選択します。

コンポーネント・タイプ

式エディターを使用して、現行コンポーネントと、そのコンポーネントのあらゆるオペランドまたは関数引数を編集することができます。一部のコンポーネントは、選択した際に式エディターで異なって表示される場合があります。

式エディターの属性コンポーネント

式エディターの属性コンポーネントを使用して、式内の属性を選択および操作します。

このタスクについて

式エディターのコンポーネント・ビューで、属性グループの属性リストから属性を選択できます。

手順

1. 特定の属性を操作するには、リストからその属性を選択し、「**編集**」をクリックします。
   「**選択されている属性の編集**」ウィンドウが表示されます。
2. 選択した属性は、以下の方法で操作できます。
   - 「**ストリング**」または「**数値**」を選択して、属性をストリングまたは数値に置き換えることができます。属性リストが入力フィールドに置き換わります。またこれ以降、内容と有効な属性名のリストとの比較がされません。
   - 「**関数**」をクリックして、属性を関数に置き換えることができます。名前の後に括弧が追加され、選択対象となる有効な関数名がリストに入れられます。
   - 属性名を選択する代わりに属性名を入力することができます。 名前の入力は、この属性グループですべての属性を定義していない場合に役立ちます。
     – 入力した名前を持つ属性がない場合は、警告が 表示されます。

  - 属性名に含めることができない文字が入力された場合は、エラーが表示されます。
  - 警告およびエラーが修正されるまで「**OK**」ボタンは使用できません。
- 属性はタイプに基づいてフィルタリングされません。誤ったタイプの属性 (または値) が選択または入力されると、警告メッセージが表示されます。

式エディターのリテラル・コンポーネント
式エディターのストリング・コンポーネントおよび数値コンポーネントを使用して、式内のリテラルを操作します。

このタスクについて

リテラルは、属性値または関数から得られた値ではない、式の中に直接入力する値です。リテラル値には、ストリングまたは数値を使用できます。

手順

- 「**属性**」をクリックして、リテラル・ストリングまたは数値を属性に置き換えることができます。有効な属性名を選択するか、引用符なしで入力す る必要があります。
- 「**関数**」をクリックして、リテラル・ストリングまたは数値を関数に置き換えることができます。名前の後に括弧が追加され、選択対象となる有効な関数名が選択リストに入れられます。
  - ストリングが予期されるときに数値を入力した場合や、数値が予期されるときにストリングを入力した場合は、警告が表示されます。
  - フィールドの内容が数値でない場合に「**数値**」を選択すると、エラーが表示されます。エラーが訂正されるまで「**OK**」は使用不可です。

式エディターの演算子コンポーネント
式エディターの演算子コンポーネントを使用して、式内の演算子を操作します。

このタスクについて

演算子コンポーネントには、演算子とそのオペランドが表示されます。

手順

- 式エディターのコンポーネント・ビューで、2 つのオペランドの間の演算子を「**演算子**」リストから選択します。 (%) 演算子は第 1 オペランドに 100 を掛けた値を、第 2 オペランドで割ります。
- 演算子 (`+ - * / or %`) を選択します。
  - ページの「**左方オペランド**」セクションは演算子の前に表示されます。
  - 「**右方オペランド**」セクションは演算子の後に表示されます。
  - 単純なオペランド (属性およびリテラル) を編集する場合には、選択コンポーネントをオペランドに変更する必要はありません。これについては、1203 ページの『式エディターの属性コンポーネント』と 1204 ページの『式エディターのリテラル・コンポーネント』で説明されています。
  - 他の演算子または関数で構成される複雑なオペランドは、「**編集**」をクリックして編集できます。このアクションにより、 演算子全体ではなく、オペランド部分が強調表示されます。

式エディターの条件式コンポーネント

条件式コンポーネントは、 条件、条件が true の場合に返す値、 および条件が false の場合に返す値を示します。

- 「**条件**」セクションの式は true または false に 評価されなければなりません。true または false を返す式の作成には、演算子 (==)、(!=)、(<)、 (<=)、 (>)、(>=)、(&&)、(||)、(!) を使用できます。
- 単純なオペランド (属性およびリテラル) を編集する場合には、選択コンポーネントをオペランドに変更する必要はありません。これについては、1203 ページの『式エディターの属性コンポーネント』と 1204 ページの『式エディターのリテラル・コンポーネント』で説明されています。

- 他の演算子または関数で構成される複雑なオペランドは、「編集」をクリックして編集できます。このアクションにより、 条件式全体ではなく、オペランド部分が強調表示されます。
- オプション「**挿入**」、「**削除**」、「**1 レベル上へ**」、および「**編集**」の使用法については、1205 ページの『式エディターの共通オプション』を参照してください。

関連概念
1202 ページの『式エディター』
式エディターを使用して、Agent Builder 内の式を作成および変更します。

式エディターの関数コンポーネント
式エディターの関数コンポーネントを使用して、式内の関数コンポーネントを選択および操作します。

このタスクについて

関数コンポーネントには、関数とその引数が表示されます。

手順

- 関数を操作するには、式エディターのリストから「**関数名**」を選択します。
  - 選択された関数の説明が、関数の後に表示されます。
  - 「**関数引数**」セクションが関数名の後に表示されます。選択された関数の適切な数の引数が表示されます。選択された関数に特化した説明が表示されます。
  - 単純な引数 (属性およびリテラル) を編集する場合には、選択コンポーネントをオペランドに変更する必要はありません。これについては、1203 ページの『式エディターの属性コンポーネント』と 1204 ページの『式エディターのリテラル・コンポーネント』で説明されています。
  - 演算子または他の関数で構成される複雑な引数は、 「**編集**」をクリックして編集できます。 このアクションにより、 関数全体ではなく、引数部分が強調表示されます。
- 引数の個数が可変である関数の場合、引数を追加するには「**挿入**」をクリックし、引数を削除するには「**削除**」をクリックします。その他のアクションについては、1205 ページの『式エディターの共通オプション』で説明されています。
- getenv 関数の場合、構成プロパティーは、「**挿入**」をクリックして選択できます。 「構成プロパティー」の選択項目を選択した場合、「**構成プロパティー**」ウィンドウが表示されます。

式エディターの共通オプション
式エディターのすべてのビューで使用できるいくつかのオプションがあります。

式エディターの共通オプションは以下のとおりです。

- **挿入**
- **削除**
- **1 レベル上へ**
- **編集**

挿入

「**挿入**」により、コンポーネントの前に演算子または関数が挿入されます。コンポーネントのレベルが、演算子オペランドの 1 つ、または関数引数の 1 つに下げられます。例えば sqrt(attr2) 関数の前の「**挿入**」をクリックすると、何を挿入するかを選択するように求められ、次の選択項目が表示されます。

- **演算子のいずれかのオペランドが sqrt(attr2) の演算子**
- **1 番目の引数が sqrt(attr2) の関数**
- **sqrt(attr2) を true または false の値と解釈する 条件式**

getenv 関数の前の「**挿入**」をクリックすると、何を挿入するかを選択するように求められ、次の選択項目が表示されます。

- **構成プロパティー**: このオプションを使用して、エージェントに対してセットアップ済みの構成プロパティーの値を取得するか、エージェントが稼働するホスト上の他の任意の環境変数の値 (JAVA_HOME など) を取得します。
- **演算子のいずれかのオペランドが attr2 の演算子**
- **1 番目の引数が attr2 の関数**
- **true または false の値と解釈する条件式 attr2**

削除

「**削除**」は、演算子と関数に対してのみ使用可能です。 これは「**挿入**」とは逆の操作です。「**削除**」をクリックすると、削除する演算子または関数を何で置き換えるかを選択するように求められます。例えば、`sqrt(attr2)` 関数の前にある「**削除**」の場合、以下の選択項目が表示されます。

- **現行引数 1、attr2**
- **新規のストリング、数値、または属性参照**

「**新規のストリング、数値、または属性参照**」を選択すると、削除するポイントから後のツリー全体が破棄され、その部分が新たな属性またはリテラル値で置き換えられます。

「**現行引数**」をクリックすると、選択されたオペランドまたは引数のレベルが上がり、削除された演算子または関数を置き換えます。さらに引数またはオペランドがある場合は、後続の選択項目をクリックできます。その他のオペランドや引数はすべて廃棄されます。

1 レベル上へ

ツリー内で 1 つ上へ移動するには、「**1 レベル上へ**」をクリックします。

編集

複雑なオペランドまたは引数の前にある「**編集**」をクリックし、編集対象コンポーネントにします。

「**編集**」をクリックした後に「**1 レベル上へ**」をクリックすると、現行コンポーネントが「**編集**」をクリックする前のコンポーネントに復元されます。

式エディター - 数式エラー

式エディターでの数式エラーの修正

式がない場合、または入力した式を構文解析できない場合は、コンポーネント・ビューの内容が異なります。式ツリーは表示されません。 代わりに、エラー・メッセージが表示されます。

構文解析エラーのある式を訂正するには、式フィールドに直接入力するか、または「**挿入**」をクリックして新規の式に置き換えます。 この場合、「**挿入**」には次の選択項目が示されます。

- **属性**
- **ストリング**
- **数値**
- **演算子**
- **条件式**
- **関数**

関連概念

1202 ページの『式エディター』
式エディターを使用して、Agent Builder 内の式を作成および変更します。

数式演算子および関数

式エディターで使用される数式演算子および関数のリファレンス (例を含む)。

派生属性の値は、同じデータ・ソースにある定数および他の属性値に基づいて式を評価した結果の値です。式の文法は通常の数式 (オペランド 演算子 オペランド) で、グループ化には括弧を使用します。数値属性

は、通常の数学演算子 (+ - * / %) を使用して、他の数値属性や定数と組み合わせることができます。% は、「**左方オペランド**」に 100 を乗算し、「**右方オペランド**」で除算します。 ストリング属性は、+ を使用して、他のストリング属性や定数と組み合わせることができます。また、以下で説明する関数を使用できます。関数は function_name(argument_1, argument_2, argument_3) の形式で入力します。

属性は、その名前 (「**データ・ソース情報**」ツリーに表示される名前) によって表されます。整数定数は、数値として指定されます。 ストリング定数は、引用符で囲みます。

次の関数を式で使用できます。

abs
数値の絶対値を戻します。

atof
ストリングを浮動小数点値に変換します。

atoi
ストリングを整数値に変換します。 通常の **C atoi** と同じ動作をします。10 進数でない最初の文字で停止します。

average
値のセットの平均である単 一の値を戻します。 値のセットは、関数の引数から取得されます。 複数の個別の値 (属性名や定数など) をそれぞれ別個の引数として指定できます。または、last 関数のみをこの関数の唯一の引数として指定することもできます (属性の最新の値の平均を計算する場合)。

この関数の使用例を以下に示します。

```
average (Attr_A, AttrB, Attr_C)
```

```
average (last (Attr_A, 10))
```

ceiling
引数以上の最小の整数を返します。

例えば、attribute_a = 12.4 の場合、ceiling(attribute_a) は値 13 を返します。また、attribute_a = -12.4 の場合、ceiling(attribute_a) は値 -12 を返します。

delta
ある属性の最新の値と、その属性の以前に収集された値の差分。delta の単一引数は last 関数である必要があります。この関数は属性の現行値と以前の値を取得します。通常は以下のように使用します。

```
delta (last(OtherAttribute, 2))
```

delta を計算するために last 関数のどの属性値が使用されるかについて詳しくは、1195 ページの『間隔の特定の計算』を参照してください。この関数は派生属性にのみ適用でき、属性グループ・フィルターには適用できません。

floor
引数以下の最大の整数を返します。

例えば、attribute_a = 12.4 の場合、floor(attribute_a) は値 12 を返します。また、attribute_a = -12.4 の場合、floor(attribute_a) は値 -13 を返します。

getenv
指定された環境変数または「構成変数」の値を戻します。

ipAddressToName
IP アドレスをホスト名に変換します。この関数は 1 つの 引数 (ドット 10 進表記の IP アドレス・ストリング) を 必要とします。アドレスを解決できない場合は、IP アドレスが返されます。

itoa
整数をストリングに変換します。 この関数は、数値をストリングに連結する場合に、最も役立ちます。派生ストリング + 関数は、ストリング引数を 2 つだけ取ります。

last
min、max、average、stddev、rate、および delta 関数によって使用される値のリストを戻します。これは、収集する属性および計算で使用する値 の数という 2 つの引数を使用します。 必要とする属性が、整数値をストリングとして格納する属性である場合は、最初の引数に atoi 関数を含めることができます。 例えば、atoi(numericalStringAttribute) のようにします。 2 番目の引数は数値でなければなりません。 定数としてハードコーディングするか、または atoi(getenv("ENV_VAR")) 式の結果を使用します。属性値を参照することはできません。

この関数の使用例を以下に示します。

```
average (last (Attr_A, 10))
```

```
last (Attribute_A, ${K01_NUM_COLLECTIONS}))
```

制約事項 : last 関数は、特定の式の中で 1 回しか使えません。

matches
正規表現が値と一致したかどうかを示す、ブール値 (true または false) を返します。ストリング・ソースおよび結果をストリングと比較する正規表現の 2 つの引数を取ります。この関数は属性グループのフィルタリングに役立ちます。

max
値のセットの最大である単 一の値を戻します。 値のセットは、関数の引数から取得されます。 複数の個別の値 (属性名や定数など) をそれぞれ別個の引数として指定できます。または、last 関数のみをこの関数の唯一の引数として指定することもできます (属性の最新の値の最大値を計算する場合)。

min
値のセットの最小である単 一の値を戻します。 値のセットは、関数の引数から取得されます。 複数の個別の値 (属性名や定数など) をそれぞれ別個の引数として指定できます。または、last 関数のみをこの関数の唯一の引数として指定することもできます (属性の最新の値の最小値を計算する場合)。

nameToIpAddress
ホスト名を IP アドレスに変換します。この関数は 1 つの引数 (ホスト名ストリング) を必要とします。アドレスを解決できない場合は、ホスト名が返されます。

NetWareTimeToTivoliTimestamp
Novell NetWare の 16 進時刻値を Tivoli Monitoring のタイム・スタンプに 変換します。この関数は 1 つの引数 (特別な NetWare 16 進時刻値) を 必要とします。 属性タイプはタイム・スタンプです。

rate
ある属性の最新の値と、その属性の以前に収集された値の間の変化率 (1 秒当たり)。 rate の単一引数は last 関数である必要があります。この関数は属性の現行値と以前の値を取得します。通常は以下のように使用します。

```
rate (last(OtherAttribute, 2))
```

rate を計算するために last 関数のどの属性値が使用されるかについて詳しくは、1195 ページの『間隔の特定の計算』を参照してください。この関数は派生属性にのみ適用でき、属性グループ・フィルターには適用できません。

replaceFirst
正規表現に一致する最初のサブストリングを、置き換えストリングで置き換えます。 この関数では、3 つの引数を使用します。1 つ目は、入力ストリングです。2 つ目は、入力ストリング内のサブストリングの突き合わせに使用する正規表現です。 3 つ目は、置き換えストリングです。 正規表現、および置き換えストリングで使用できる置換値の詳細については、1489 ページの『ICU 正規表現』を参照してください。

replaceAll
正規表現に一致するすべてのサブストリングを、置き換えストリングで置き換えます。 この関数では、3 つの引数を使用します。1 つ目は、入力ストリングです。2 つ目は、入力ストリング内のサブストリングの突き合わせに使用する正規表現です。 3 つ目は、置き換えストリングです。 正規表現、および置き換えストリングで使用できる置換値の詳細については、1489 ページの『ICU 正規表現』を参照してください。

round
数値を最も近い整数に数学的に丸めます。

sqrt
数値の平方根を返します。

stddev
値のセットの標準偏差である単 一の値を戻します。 値のセットは、関数の引数から取得されます。 複数の個別の値 (属性名や定数など) をそれぞれ別個の引数として指定できます。または、last 関数のみをこの関数の唯一の引数として指定することもできます (属性の最新の値の標準偏差を計算する場合)。

StringToTivoliTimestamp
日時ストリングを Tivoli Monitoring のタイム・スタンプに 変換します。この関数は 2 つの引数を必要とします。最初の引数は、タイム・スタンプをフリー・フォームのストリングで表したものです。2 番目の引数は、 タイム・スタンプのフリー・フォーム・ストリング表記の解析方法を示す 書式制御ストリングです。有効な書式パラメーターを 1209 ページの表 263 に示します。属性タイプはタイム・スタンプです。

表 *263. StringToTivoliTimestamp* の有効な書式パラメーター

シンボル	意味	フォーマット	例
y	年	yy yyyy	*96* *1996*
M	月 **注 :** サポートされているのは 英語の月ストリングだけです。	M または MM MMM MMMM	*09* *Sept* *September*
d	日	d dd	*2* *02*
E	曜日 **注 :** サポートされているのは 英語の曜日ストリングだけです。	EE EEE EEEE	*Sa* *Sat* *Saturday*
h	午前または午後の時間 (1 から 12)	hh	*07*
H	時間 (0 から 23)	HH	*00*
m	分	mm	*04*
s	秒	ss	*05*
S	ミリ秒	S SS SSS	*2* *24* *245*
a	AM または PM マーカー	a または aa	*am*

表 263. StringToTivoliTimestamp の有効な書式パラメーター (続き)			
シンボル	**意味**	**フォーマット**	**例**
その他の ASCII 文字	この文字をスキップします	- (ハイフン) (スペース) / (スラッシュ) : (コロン) * (アスタリスク) , (コンマ)	

1210 ページの表 264 は、タイム・スタンプのストリング表記およびその解析に使用する書式制御ストリングの例を示します。

表 264. StringToTivoliTimestamp の例 . タイム・スタンプのストリング表記のいくつかの例がリストおよび説明されている表	
タイム・スタンプのストリング表記	**書式制御ストリング**
96.07.10 at 15:08:56	`yy.MM.dd ** HH:mm:ss`
Wed, August 10, 2010 12:08 pm	`EEE, MMMM dd, yyyy hh:mm a`
Thu 21/01/2010 14:10:33.17	`EEE dd/MM/yyyy HH:mm:ss.SS`

sum
値のセットの合計である単 一の値を戻します。 値のセットは、関数の引数から取得されます。 複数の個別の値 (属性名や定数など) をそれぞれ別個の引数として指定できます。または、last 関数のみをこの関数の唯一の引数として指定することもできます (属性の最新の値の合計を計算する場合)。

TivoliLogTimeToTivoliTimestamp
Tivoli のログ・ファイルのタイム・スタンプを、Tivoli Monitoring のタイム・スタンプに変換します。この関数は 1 つの引数 (Tivoli ログ・ファイルのストリング・タイム・スタンプ) を必要とします。 属性タイプはタイム・スタンプです。

tokenize
トークン化されたストリングの中の 1 つのトークン。この関数は 3 つの引数を 必要とします。最初の引数は、 トークンに分割するストリングです。2 番目の引数は、トークンを相互に区切る ストリング内の文字を 1 つ以上指定します。この引数で指定された 文字のいずれかが出現すると、最初の引数の中にあるトークンが識別され、 分離されます。3 番目の引数は、この関数の結果として返す トークンのインデックスです。最初のトークンは インデックス 0 であり、2 番目のトークンはインデックス 1 です (以下同様)。この引数をストリング LAST にして、最後のトークンを返させることもできます。

UTCtoGMT
協定世界時を GMT の Tivoli Monitoring タイム・スタンプに変換します。この関数は 1 つの引数 (整数の time_t 値) を 必要とします。 属性タイプはタイム・スタンプです。

UTCtoLocalTime
協定世界時をローカルの Tivoli Monitoring タイム・スタンプに 変換します。この関数は 1 つの引数 (整数の time_t 値) を 必要とします。 属性タイプはタイム・スタンプです。

引数を取らずに数値を返す関数を以下に示します。

count
最初に呼び出されたときに 1 から開始するカウンターを保持し、 続けて呼び出されるごとにカウンターは 1 増加します。 `last` を使用している式でこの関数を同時に使用すると、`last()` によって格納された要素の数と一致しますが、それは `last()` が最大個数に達するまでの間に限ります。この時点で、`last()` は新しい値のために 最も古い値の削除を開始します。そのため、`count()` は永久に増加し続けますが、last() の値の総数は同じ ままになります。

cumulativeSum
フロー制御要約イベントによって表される重複イベントの引数値の合計を返します。または、データ・ソースからの単一イベントである場合、その引数を返します。この関数では、1 つの数値引数を使用します。 この関数は、イベント・フィルタリングおよび要約がオンに設定されたイベントの属性グループにのみ適用されます。

eventThreshold
イベントを生成した属性グループについて構成されているしきい値を返します。 以下の 3 つの列挙を使用した数値です。

- SEND_ALL (-3)
- SEND_FIRST (-2)
- SEND_NONE (-1)

括弧内の数値は未加工値です。ただし、Agent Builder は列挙を定義するため、デフォルトではテキスト・バージョンを Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールで確認できます。 3 つの事前定義された選択項目のいずれかではなく、実際の数値のしきい値を指定した場合、この関数はその数を返します。 値は 0 より大きい整数です。この関数は、イベント・フィルタリングおよび要約がオンに設定されたイベントの属性グループにのみ適用されます。

isSummaryEvent
データ・ソースからの単一イベントである場合、0 を返します。イベントがフロー制御要約イベントである場合、1 を返します。 関数にデフォルトの属性を使用する場合、表示される値はイベントおよび要約イベントです。 属性を手動で作成する場合、名前を列挙として定義しない限り、表示される値は 0 および 1 です。 この関数は、イベント・フィルタリングおよび要約がオンに設定されたイベントの属性グループにのみ適用されます。

occurrenceCount
フロー制御要約イベントによって表される一致イベントの数を返します。データ・ソースからの単一イベントである場合、1 を返します (フロー制御要約イベントには、最初のイベントが含まれます)。この関数は、イベント・フィルタリングおよび要約がオンに設定されたイベントの属性グループにのみ適用されます。

summaryInterval
イベントを生成した属性グループについて構成されている要約間隔を秒単位で返します。この関数は、イベント・フィルタリングおよび要約がオンに設定されたイベントの属性グループにのみ適用されます。

例

派生属性およびフィルタリング属性を作成する数式演算子および関数の使用例

例 1 - 派生属性

以下の属性タイプを定義するデータ・ソースがあるとします。

Name	ストリング
xBytes	数値
yBytes	数値
Virtual_Size	数値

以下を定義することができます。

- 属性 totalBytes は、xBytes と yBytes の合計として定義できます。式 xBytes + yBytes を入力します。
- 属性 yPercent が合計バイト数 (yBytes) の百分率である場合、yBytes % (xBytes + yBytes) または yBytes % totalBytes のように定義できます。

例 2 - 派生属性

この式は、Virtual_Size 属性について最近収集された値の最大値を返します。収集するサンプル数には、getenv によって取得した構成変数 *K4P_COLLECTIONS_PER_HISTORY_INTERVAL* を atoi によって数値に変換したものを使用しています。

```
max(last(Virtual_Size,atoi(getenv("K4P_COLLECTIONS_PER_HISTORY_INTERVAL"))))
```

例 3 - 派生属性

この式は、xBytes および yBytes 属性値の平方合計の平方根を返します。

```
sqrt(xBytes * xBytes + yBtyes * yBytes)
```

例 4 - 派生属性

この式は、属性グループの最新の 20 個のサンプルをもとに、xBytes 属性の平均を返します。エージェントが開始されてから収集されたサンプルが 20 個よりも少ない場合は、すべてのサンプルをもとに xBytes 属性の平均を返します。

```
average(last(xBytes,20))
```

例 5 - フィルタリング属性

以下を返すデータ・ソースがあるとします。

```
Name    Type  Size  Used  Free
Memory  MEM      8     4     4
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

ディスク使用量のみを調べたいとします。解決策は、フィルターを作成して、返されるデータを制限することです。返されるデータを制限するには、 以下のようにブール値 (true または false の値) を返す単純なフィルターを作成します。

ディスク・フィルター:

```
Type=="DISK"
```

このようにすると、フィルター Type=="DISK" が true の場合に、属性グループは以下の例のようにディスク使用量のデータのみを返します。

```
Name    Type   Size  Used  Free
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

例 6 - フィルタリング属性

以下を返すデータ・ソースがあるとします。

```
Name   Size  Used  Free
Memory    8     4     4
Disk1   300   200   100
Disk2   500   100   400
```

返されるデータは前の例に似ていますが、今回は Type 属性が存在しません。ここで matches 関数を使用して、「Name」属性の値と "Disk" の後ろに数値が続くものとが一致する、すべてのデータ行を検索できます。

ディスク・フィルター:

```
matches(Name, "Disk[0-9]*")
```

フィルターにより、属性 Name のストリングと "Disk" の後ろに数値が続くものとが一致した場合、以下のようにディスクの使用量のデータ行のみが返されます。

```
Name   Size  Used  Free
Disk1   300   200   100
Disk2   500   100   400
```

オペレーティング・システムの指定

エージェントがサポートするオペレーティング・システムの一部では使用できないデータ・ソースを定義するときには、そのデータ・ソースを実行するオペレーティング・システムを指定する必要があります。

このタスクについて

デフォルトでは、データ・ソースは、エージェント・レベルで定義されたすべてのオペレーティング・システムでデータを提供します (1176 ページの『デフォルト・オペレーティング・システム』を参照)。 データ・ソースごとにオペレーティング・システムを変更できます。

手順

1. 「オペレーティング・システム」セクションを開くには、データ・ソースを追加するときに、「**データ・ソース情報**」ページで「**オペレーティング・システム**」をクリックします。
2. データ・ソースを操作するオペレーティング・システムを選択します。

   個々のオペレーティング・システム、すべてのオペレーティング・システム、特定の種類のすべてのオペレーティング・システム、またはエージェントのデフォルト・オペレーティング・システムを選択します。

データ収集の構成およびチューニング

Agent Builder エージェントを作成するときに、最善の結果を得るためにエージェントのデータ収集を構成およびチューニングできます。

エージェントを構成およびチューニングする方法は、Agent Builder エージェントごとに異なることがあります。単一のエージェント内でも、属性グループごとに異なることがあります。Agent Builder エージェントは、2 つのタイプのデータを組み込むことができ、最も一般的なタイプのデータに対しては 2 つの基本的なデータ収集方法をサポートします。

データ・タイプ

エージェントは 2 つのタイプのデータを収集します。

1. ほとんどの Tivoli Monitoring 属性グループは、データのスナップショットを表します。データが要求されると、そのデータが返されます。エージェントは、このタイプのデータを使用して、構成、パフォーマンス、状況、および一連のデータの 1 回の収集が意味を持つその他の情報を表します。このデータを「サンプル・データ」と呼びます。
2. 一部の Tivoli Monitoring データは、イベントを表します。この場合には、イベントが発生すると、エージェントはデータを Tivoli Monitoring に転送する必要があります。イベントの例は、SNMP トラップ、Windows イベント・ログ項目、ログ・ファイルに書き込まれる新規レコードなどです。簡略化のために、これらのタイプのデータをグループ化して、イベント・データ と呼びます。

サンプル・データ

サンプル・データが必要な場合には、特定の属性グループのエージェントに要求が送信されます。要求は、Tivoli Enterprise Portal のワークスペースをクリックすることによって開始される場合があります。要求を開始する場合があるその他のものとして、実行中のシチュエーション、ウェアハウスのデータ収集、および SOAP 要求があります。エージェントは、要求を受信すると、その属性グループの現在のデータを返します。Tivoli Enterprise Portal 要求は、特定の管理対象システム名 (MSN) の特定の属性グループを対象とします。シチュエーションおよびヒストリカルな要求は、特にサブノードを含むエージェントの場合に顕著な特徴を示します。シチュエーションがサブノード内の属性グループのデータを必要とするとき、エージェントは、対象となるサブノードのリストを持つ要求を 1 つ受信します。エージェントが、すべてのサ

ブノードの要求された属性グループのデータをすべて返さなければ、Tivoli Monitoring は次の要求を処理できません。

エージェントが要求を満たすための最も簡単な方法は、Tivoli Monitoring から要求を受信するたびにデータを収集することです。Agent Builder エージェントは、データを毎回収集するわけではありません。データの収集が毎回行われないのは、データの収集には時間がかかり、また、リソースを使用するためです。さらに、多くの場合、同じデータを短期間に何度も要求されることが多いためです。例えば、同じ間隔で実行される複数のシチュエーションを属性グループで定義しており、シチュエーションが複数の条件をシグナル通知できるとします。これらの各シチュエーションによってエージェントに対する要求が生成されますが、各シチュエーションに同じデータを提供したいとします。各シチュエーションに同じデータを提供することで結果の整合性を確保し、モニター・エージェントによるシステム・リソースの要求を最小限に抑えるのは、現実的な方策です。

エージェント開発者は、以下の 2 つのモードのいずれかでの収集の実行を選択することにより、データ収集が最適化されるようにエージェントを構成できます。

1. **要求時収集**: エージェントは要求の受信時にデータを収集し、そのデータを返します。
2. **スケジュール収集**: エージェントは、スケジュールされた間隔でバックグラウンドでデータを収集し、要求を受信すると、収集してある最新のデータを返します。

エージェントは、これらのいずれのモードでも短期キャッシュを使用します。キャッシュが有効な間に別のデータ要求を受信すると、エージェントは、要求ごとに新しいデータを収集するのではなく、キャッシュにあるデータを返します。 キャッシュにあるデータを使用することで、複数の並行シチュエーション (および他の種類の) 要求に起因する問題が解消されます。データが有効である期間、スケジュール収集の間隔、収集に使用するスレッドの数、およびエージェントを要求時モードとスケジュール・モードのいずれで実行するかは、すべて環境変数で定義します。 環境変数を使用すると、各環境で最適な動作になるように各エージェントを調整できます。

各モードでのエージェントの動作については、以下の例を参照してください。

- エージェント 1 (要求時 収集): 通常はシチュエーションによってアクセスされるか、Tivoli Enterprise Portal で低い頻度でアクセスされるだけの、少量のデータを収集する単純なエージェント。データ収集は十分に高速ですが、コンピューティングおよびネットワーキングのリソースを 使い果たしてしまう可能性があります。通常、 このようなエージェントは要求時で実行するように定義します。シチュエーションが実行されていない場合、またはユーザーが Tivoli Enterprise Portal でクリックを行わない場合は、エージェントは何も実行しません。データは、必要になると収集されて返されます。 データは短期キャッシュに格納されるため、ほぼ同時に別の要求が出された場合は、同じデータが返されます。このタイプの収集は、実際にデータが必要になった場合にのみデータが収集されるため、このエージェントを最も効率よく実行できると考えられます。
- エージェント 2 (スケジュール 収集): サブノードを 含み、モニター対象リソースの複数のコピーからデータを収集する複雑なエージェント。1 つのエージェントが、リソースの多くのコピーを管理できます。通常は、比較的高い頻度でデータに対するシチュエーションを実行し、モニター対象リソースの状況およびパフォーマンスをモニターします。このようなエージェントは、「スケジュール」収集を実行するように定義します。「スケジュール収集」を実行する理由の 1 つに、Tivoli Monitoring エージェントによってシチュエーションが評価される仕組みが挙げられます。シチュエーションはサブノードの 属性グループを対象として実行されるため、エージェントは 1 つのデータ要求をすべてのサブノードから 同時に受信します。 シチュエーションのすべてのデータが返されるまで、エージェントは他の要求に応答できません。エージェントが要求の受信時にすべてのデータを収集するとすると、Tivoli Enterprise Portal でワークスペースをクリックしたときにエージェントがフリーズしてしまいます 。エージェントのフリーズを避けるために、Agent Builder は、すべてのサブノード ・エージェントがスケジュール収集を実行するように自動的に定義します。エージェント開発者は、データ・タイプに応じた合理的な間隔でデータを収集するように、スレッド数および最新表示間隔を調整します。例えば、最新表示間隔を 1 分に 1 回や 5 分に 1 回にすることができます。

環境変数

エージェントは、一連の環境変数の値に基づいて、使用するモードと、スケジュールされたデータ収集を実行する方法を判別します。これらの環境変数は、「**環境変数**」パネルのエージェントの定義で設定できます。各環境変数は、デフォルト値とともにメニューにリストされます。Windows の場合はエージェントの環境 (env) ファイル、UNIX の場合は初期化 (`ini`) ファイルを編集することで、インストール済みエージェ

ントの環境変数を設定したり変更したりすることもできます。サンプル属性グループのデータ収集を制御する環境変数は以下のとおりです。

- CDP_DP_CACHE_TTL=<キャッシュに入っているデータの有効期間 - デフォルト値は 55 秒>
- CDP_DP_THREAD_POOL_SIZE=<並行収集に使用するスレッドの数 - サブノード・エージェントの場合のデフォルト値は 15>
- CDP_DP_REFRESH_INTERVAL=<秒単位で表した収集間隔 - サブノード・エージェントの場合のデフォルト値は 60 秒>
- CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT=<有効期間が経過した後に新規データを待機する時間 - デフォルト値は 5 秒>

これらの変数のうち、特に重要なものは CDP_DP_CACHE_TTL、CDP_DP_REFRESH_INTERVAL、および CDP_DP_THREAD_POOL_SIZE です。

CDP_DP_THREAD_POOL_SIZE の値が 1 以上である場合、 またはエージェントがサブノードを 含む場合、エージェントは「スケジュール」収集モードで動作します。CDP_DP_THREAD_POOL_SIZE が未設定または 0 である場合、エージェントは「要求時」収集モードで実行されます。

エージェントが「スケジュール」モードで実行されている場合、エージェントは CDP_DP_REFRESH_INTERVAL 秒ごとに 自動的にすべての属性グループを収集します。収集の実行には、一連のバックグラウンド・スレッドを使用します。スレッドの数は CDP_DP_THREAD_POOL_SIZE を使用して設定します。CDP_DP_THREAD_POOL_SIZE の正しい値は、エージェントの処理内容によって異なります。例えば、次のようにします。

- エージェントが SNMP を使用してリモート・システムからデータを収集する場合は、CDP_DP_THREAD_POOL_SIZE を、モニター対象リモート・システムの数と同様の値にするのが最適です。モニター対象リモート・システムの数と同様の値にプール・サイズを設定すると、エージェントが並行してデータを収集しますが、リモート・システムでの並行ロードは制限されます。SNMP デーモンは、処理が集中すると要求を破棄してしまいます。 要求が破棄されると、エージェントが強制的に再試行モードに移行され、データの収集にさらに長い時間とより多くのリソースを消費してしまいます。
- 収集に長い時間がかかる属性グループがエージェントに多く含まれている場合は、長時間のデータ収集を並行して実行できるのに十分なスレッドを使用してください。残りの属性グループのために、さらにスレッドを多少増やすこともできます。ターゲット・リソースによる処理が可能な場合、この方法でスレッドを使用します。属性グループの収集に長い時間がかかる場合の例として、スクリプトが長時間実行される場合や、JDBC 照会に長い時間がかかる場合などがあります。

大きなスレッド・プールでエージェントを実行すると、エージェントは、より多くのメモリーを (主に各スレッドに割り振るスタックのために) 使用します。しかし、プロセスのプロセッサー使用量は増加せず、プロセスの実際の実効ページ・セット・サイズが顕著に増加することもありません。 ワークロードに応じた適切なスレッド・プール・サイズを使用するとエージェントがより効率的になります。特定の環境の特定のエージェントが目的の動作をするようにスレッド・プール・サイズを調整できます。

データは、収集されると内部キャッシュに格納されます。このキャッシュは、新しいデータが収集されるまで、後続の要求を満たすために使用されます。キャッシュの有効期間は CDP_DP_CACHE_TTL によって制御します。 デフォルトでは、有効期間は 55 秒に設定されています。エージェントがスケジュール・モードで実行されているときは、有効期間を CDP_DP_REFRESH_INTERVAL と同じ値に設定するのが最適です。データ収集に長い時間がかかる場合は少し大きい値に設定します。この方法で有効期間を設定すると、データは、次のスケジュール収集まで有効と見なされます。

最後の変数は CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT です。この変数は、新しいデータが収集される前に CDP_DP_CACHE_TTL が満了した場合にのみ使用されます。新しいデータが収集される前にキャッシュの有効期限が切れた場合、エージェントは、次のデータ収集を直ちにスケジュールに入れます。その後、その収集が完了するのを最大 CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT 秒間待機します。新規の収集が完了すると、キャッシュが更新され、最新のデータが返されます。新規の収集が完了しない場合は、既存のデータが返されます。Universal Agent で見られる問題を避けるために、エージェントは、CDP_DP_CACHE_TTL が完了してもキャッシュを消去しません。Universal Agent は、有効期間が終了すると、必ずデータ・キャッシュを消去します。次の収集が完了する前に Universal Agent がデータ・キャッシュを消去すると、その属性グループのキャッシュは空になり、収集が完了するまでデータは返されません。

シチュエーションが実行中である場合、データが返されないと問題が発生します。キャッシュが消去された後、次の収集が完了する前に実行されたシチュエーションはデータを取得できず、起動されるシチュエーションはすべてクリアされます。データ収集は少し遅いため、イベントの起動とクリアが大量に実行されてしまいます。Agent Builder エージェントではこの問題は発生しません。「古い」データでシチュエーションが起動されても、通常は、データが同じであるため、シチュエーションも同じ状態にとどまります。次の収集が完了すると、シチュエーションが新しいデータを取得し、有効なデータに基づいて起動またはクリアのいずれかを実行します。

属性グループ

Agent Builder エージェントには、データ収集の動作を検査し、ご使用の環境に応じてエージェントをチューニングするために使用できる 2 つの属性グループが含まれています。それらの属性グループは「パフォーマンス・オブジェクト状況」と「スレッド・プールの状況」です。 これらの属性グループがデータ収集のパフォーマンスをチューニングするために使用された場合における、最も有用なデータを以下に示します。

- 「パフォーマンス・オブジェクト状況」、「平均収集期間」属性。この属性は、各属性グループがデータ収集に費やしている時間を示します。多くの場合、エージェントの属性グループのごく一部が、そのエージェントの大部分のプロセッサー使用量または時間を占めています。該当する 1 つ以上の属性グループの収集を最適化できる場合があります。また、他のデータほど情報が新しくなくても構わない場合は、該当する 1 つ以上のグループの収集間隔を変更することができます。詳しくは、1217 ページの『例および拡張チューニング』 を参照してください。
- 「パフォーマンス・オブジェクト状況」、「スキップされた間隔」属性。この属性は、エージェントが属性グループに対する新規の収集をスケジュールに入れようとしたが、以前の収集がまだキューで実行待機中であったか既に実行中であった回数を示します。正常に動作しているエージェントの場合、すべての属性グループでこの属性の値がゼロになります。この数値が増加を始めた場合は、スレッドの追加、収集間隔の延長、または収集の最適化によってデータ収集をチューニングしてください。
- 「スレッド・プールの状況」、「スレッド・プールの平均アクティブ・スレッド」属性。この値を「スレッド・プール・サイズ」属性グループと比較すると、スレッド・プールが適切に使用されているかどうかが分かります。100 スレッドのスレッド・プール・サイズを割り振ったが、アクティブ・スレッドの平均数が 5 である場合、メモリーを浪費しているだけです。
- 「スレッド・プールの状況」、「スレッド・プールの平均ジョブ待機」属性および「スレッド・プール・キューの平均長さ」属性。これらの属性は、データ収集にあたってスレッドによる処理をキューで待機していた時間の平均と、キューに置かれていた収集の数の平均を表します。このデータ収集の方法により、アイドル状態のシステムであっても、平均して少なくとも 1 つのジョブがキューで待機していると示されます。待機中ジョブの数が多い場合、または平均待機時間が長い場合は、収集が輻輳しています。1 つ以上の属性グループについてスレッドの追加、収集間隔の延長、または収集の最適化を検討してください。

イベント・データ

Agent Builder エージェントは、いくつかの種類のイベント・データを公開できます。一部の動作は、すべてのイベント・データに共通です。エージェントは、新規のイベントをそれぞれ別個のデータ行として受信します。イベント・データの行を受信すると、直ちに Tivoli Monitoring に送信して処理させるとともに、エージェントの内部キャッシュに追加します。それぞれの行が Tivoli Monitoring に送信されると、Tivoli Monitoring によってシチュエーションおよびヒストリカル収集が実行されます。キャッシュは、データを取得する Tivoli Enterprise Portal または SOAP 要求を満たすために使用されます。エージェントはこのキャッシュを使用して、重複の検出、フィルタリング、および要約を実行できます (属性グループに対して定義されている場合)。各属性グループのイベント・キャッシュのサイズは `CDP_PURE_EVENT_CACHE_SIZE` により設定されます。このキャッシュには、最新の `CDP_PURE_EVENT_CACHE_SIZE` 個のイベントが格納され、最新のイベントが最初に返されます。キャッシュは、イベント属性グループごとに個別に用意されています。属性グループのキャッシュがいっぱいになると、最も古いイベントがリストから除去されます。

Agent Builder エージェントは、以下のイベントを公開できます。

- Windows イベント・ログの項目
- SNMP トラップまたは情報
- ログ・ファイルに追加されたレコード
- JMX MBean 通知

- JMX モニター
- Java API プロバイダーまたはソケット・プロバイダーからのイベント
- 属性グループの結合 (いずれかのデータ・ソースがイベント・データ・ソースになる)

これらのイベントは、各ソースに最適な方法で処理されます。SNMP トラップおよび情報、JMX 通知、Java API およびソケット・プロバイダーからのイベントは、非同期的に受信され、直ちに Tivoli Monitoring に転送されます。これらのコレクターをチューニングする必要はありません。エージェントは、Windows イベント・ログ API を使用してオペレーティング・システムから Windows イベント・ログ項目の受信をサブスクライブします。エージェントが古いイベント・ロギング API を使用している場合は、スレッド・プールの設定を使用してシステムに対して新規イベントをポーリングします。いずれかのデータ・ソースがイベント・データ・ソースである属性グループの結合の場合、結合された属性グループに適用されるチューニングはありません。ただし、結合された属性グループには、イベント・ソース・グループに適用されるチューニングの効果が及びます。

ファイルのモニターはより複雑です。エージェントはファイルが存在するかどうか、およびいつ新しいレコードがファイルに追加されたかをモニターする必要があります。 ファイル名または静的な名前のパターンを使用してファイルをモニターするようにエージェントを構成できます。その後、パターンに一致するファイルのセットが変化する可能性があるため、エージェントは、KUMP_DP_FILE_SWITCH_CHECK_INTERVAL 秒ごとに新規ファイルまたは変更されたファイルがあるかどうかを検査します。このグローバル環境変数は、エージェント・インスタンスのすべてのファイル・モニターに影響します。エージェントは、モニター対象の該当ファイルを判別するときに、いつファイルが変更されたかを判別する必要があります。Windows システムの場合、エージェントは、オペレーティング・システムの API を使用してこれらの変更を listen します。エージェントは、ファイルが更新されると通知を受け取り、直ちに処理します。UNIX システムの場合、エージェントは KUMP_DP_EVENT 秒ごとにファイルが変更されたかどうかを検査します。このグローバル環境変数は、エージェント・インスタンスのすべてのファイル・モニターに影響します。ファイルが変更されたことをエージェントが認識すると、 そのファイルの新しいデータをすべて処理し、次の変更を待機します。

例および拡張チューニング

例

高度なチューニングに使用する環境変数が、エージェント・レベルで定義されています。以下の変数を一度設定すると、エージェント内のすべての属性グループに適用されます。

- CDP_DP_CACHE_TTL
- CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT
- KUMP_DP_FILE_SWITCH_CHECK_INTERVAL
- KUMP_DP_EVENT

以下の変数は個々の属性グループに適用できます。ただし、エージェント内の他のすべての属性グループに適用されるグローバル設定も持っています。

- CDP_DP_REFRESH_INTERVAL
- CDP_PURE_EVENT_CACHE_SIZE

以下の 6 つの属性グループを含めるエージェントを定義する場合、

- EventDataOne
- EventDataTwo
- EventDataThree
- SampledDataOne
- SampledDataTwo
- SampledDataThree

以下のデフォルト変数を設定するとします。

- CDP_DP_CACHE_TTL=55

- CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT=2
- CDP_DP_REFRESH_INTERVAL=60
- CDP_PURE_EVENT_CACHE_SIZE=100

この場合は、サンプル・データ (SampledDataOne、SampledDataTwo、および SampledDataThree) を含むすべての属性グループが 60 秒ごとに収集されます。イベント属性グループ (EventDataOne、EventDataTwo、および EventDataThree) は、それぞれのキャッシュに直近のイベント 100 件を保管します。

これらの設定で十分機能する場合もあれば、設定を詳細なレベルで制御することが必要な場合もあります。例えば、通常の状態で EventDataOne が EventDataTwo および EventDataThree の 10 倍の数のイベントを受信する場合について考えてみます。実際には EventDataOne と EventDataTwo の間に関連があるとすると、事態はさらに複雑になります。EventDataTwo のイベントを 1 つ受信するときには、必ず EventDataOne のイベントが複数発生しており、これらのイベントを相関させる必要があるとします。キャッシュ・サイズの適切な設定値は、一概には決まりません。EventDataOne が多数のイベントを格納できるようにして、EventDataTwo はそれより少ないイベントを格納できるようにするのが適切です。このようなストレージを実現するには、CDP_PURE_EVENT_CACHE_SIZE を、大多数のイベント属性グループに対して妥当なサイズ (100 程度) に設定します。その上で、CDP_EVENTDATAONE_PURE_EVENT_CACHE_SIZE を 1000 に設定できます。このようにすると、対応するすべてのイベントを Tivoli Enterprise Portal で表示できます。

同じことを、CDP_DP_REFRESH_INTERVAL でも実現できます。 エージェントで最も多くの属性グループに対して効果的なデフォルト値を設定します。それと異なる方法で収集する必要がある属性グループには CDP_*attribute group name*_REFRESH_INTERVAL を設定します。収集を最適化するために、デフォルトの CDP_DP_REFRESH_INTERVAL は CDP_DP_CACHE_TTL の値に一致させてください。CDP_DP_CACHE_TTL はグローバル値であるため、最新表示間隔より小さい値に設定すると、予期しない収集が行われる可能性があります。

データ・ソースの定義およびテスト

Agent Builder は、多数のデータ・プロバイダーをサポートします。各データ・プロバイダーからデータ・ソースを作成することができます。データ・ソースを作成およびテストするための手順は、データ・プロバイダーごとに異なります。

ほとんどのデータ・プロバイダーでは、データ・ソースを作成すると、データ・セット (属性グループ) がエージェントに追加されます。データ・セットには、このデータ・ソースによって収集された情報が含まれます。

プロセス、Windows サービス、またはプログラム戻りコードのデータ・プロバイダーを含むデータ・ソースは、特別な可用性データ・セットを使用します。1 つのエージェント内で作成できる可用性データ・セットは 1 つのみです。このエージェント内でプロセス、Windows サービス、またはプログラム戻りコードのデータ・プロバイダーを含むすべてのデータ・ソースによって収集された情報が格納されます。

エージェントまたはサブノードのすべての Windows ログ・データ・ソースは、イベント情報を 1 つの「イベント・ログ」データ・セット内に配置します。

Cloud APM のデータ・ソースのセットアップ

Cloud APM では、すべてのデータ・セットからのデータを詳細ダッシュボードで使用できます。また、しきい値マネージャーを使用してしきい値をセットアップすることができます。エージェントまたはサブノードの要約ダッシュボード内で、状況標識やリソース情報 (サービス名、アドレス、ポート) などのデータ・セットからの情報を使用する場合、そのデータ・セットで 1 行だけ生成されるようにする必要があります。

ほとんどのデータ・プロバイダーでは、データ・セット構成で「**単一データ行の生成**」を選択できます。収集された情報に複数の行が含まれる場合、「**拡張**」をクリックして、正しい行を確実に生成するフィルターをセットアップできます (説明については、1202 ページの『属性グループのフィルタリング』を参照してください)。収集された情報で、必要な行が生成されることを確認するために、データ・ソースをテストできます。

一部のデータ・プロバイダーでは、データ・セットは、複数行を生成する必要があります。また、プロセス、Windows サービス、コマンド戻りコードのデータ・ソースの場合、1 つの可用性データ・セット内にデータが格納され、複数の行が生成されます。この場合、1 行だけを生成するフィルタリングされたデータ・セットを作成する必要があります。フィルタリングされたデータ・セット (属性グループ) の作成方法については、1341 ページの『フィルタリング属性グループの作成』を参照してください。

イベント・データを生成するデータ・プロバイダーもあります。この場合は、新規イベントごとに 1 つの行が組み込まれます。Cloud APM の要約情報やリソース情報で、これらのデータ・プロバイダーを使用しないでください。

以下のデータ・プロバイダーは、複数行を含むデータ・セットを生成する必要があります。

- プロセス (可用性データ・セットを使用)
- Windows サービス (可用性データ・セットを使用)
- プログラム戻りコード (可用性データ・セットを使用)
- 一部のデータ・タイプでは、SNMP および JMX
- アプリケーションによっては、ソケットおよび Java API

以下のデータ・プロバイダーは、イベント・データを生成します。

- SNMP イベント
- ログ・ファイル
- AIX バイナリー・ログ
- Windows イベント・ログ
- アプリケーションによっては、ソケットおよび Java API

データ・セットのいずれかの属性で、状況値を指定する必要があります。Cloud APM は、状況標識全体でその値を使用します。状況標識として使用できる属性が行に含まれていない場合は、派生属性を作成して、状況を計算できます。状況重大度値を構成する必要があります。手順については、1201 ページの『状況標識として使用される属性の重大度の指定』を参照してください。

プロセスのモニター

サーバー上で実行される 1 つまたは複数のプロセスをモニターするデータ・ソースを定義できます。プロセスは、エージェントと同じホスト上で実行されている必要があります。プロセスごとに、データ・ソースは、可用性データ・セットに 1 つの行を追加します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**プロセス**」をクリックします。
2. 「**データ・ソース**」領域で、「**プロセス**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**プロセス・モニター**」ページの「**プロセス情報**」領域に、表示名とプロセス名を指定します。プロセス名は、手動で入力すること、または「**参照**」をクリックして取得することができます。「**参照**」をクリックすると、ローカル・システムまたはリモート・システムで現在実行されているプロセスのリストが表示されます。

   「**引数マッチングの使用**」オプションおよび「**コマンド行全体の突き合わせ**」オプションを選択することで、プロセスをさらに絞り込むことができます。例えば、同じプロセスの複数のインスタンスがシステムで実行されている場合、これらのオプションを使用することで、個々のインスタンスを見分けることができます。

表 265. **「プロセス・モニター」**ページのフィールド. **「プロセス・モニター」**ページのフィールドおよびその説明がリストされている表

フィールド名	説明	許容値
表示名	Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに表示される、プロセスによって実装されたアプリケーションのコンポーネントの記述名	説明のストリング
プロセス名	モニター対象となるプロセスの名前	有効な実行可能ファイル名
引数マッチングの使用	プロセス引数のマッチングを行う場合に選択します	オンまたはオフ
引数	マッチング対象となる引数ストリング。引数のマッチングでは、引数のサブストリングとして指定されたストリングを探します。 引数の任意の部分を入力ストリングとして指定した場合、マッチングは成功します。	ストリング
コマンド行全体の突き合わせ	パスを含んでいる可能性のある実行可能ファイルの完全な名前を指定します	オンまたはオフ
コマンド行	指定されたストリングを、プロセスを開始するために使用される完全修飾コマンド名と突き合わせます。コマンド引数は含まれません。完全修飾とは、コマンドのパスを含める必要があることを意味します。	ストリング
オペレーティング・システム	このプロセスを実行するオペレーティング・システムを選択します	任意選択

5. 「**参照**」をクリックすると、「**プロセス・ブラウザー**」ウィンドウが開きます。最初の状態では、このウィンドウには、Agent Builder システム上の各プロセスの詳細情報が入っています。情報には、ID、プロセス名、そのプロセスに対する完全なコマンド行が含まれます。 1 つ以上のプロセスを選択するか、以下のアクションのうちの 1 つ以上を実行して「**プロセス・ブラウザー**」ウィンドウのリストを操作します。

   a) プロセスのリストをソートするために、列見出しをクリックする。

   b) ウィンドウ内の情報を最新表示するために、「**最新表示**」(稲妻) アイコンをクリックする 。

   c) 特定のプロセスを検索するために、「**検索**」(双眼鏡) アイコンをクリックする 。

   検索句を入力し、オプション・セクションを選択してプロセス ID、名前、およびコマンド行で検索できます。

   d) 別のシステムのプロセスを表示させるには、 以前に定義したシステムを「**接続名**」リストから選択します。 または、「**追加**」をクリックして新しいシステムのシステム 情報を入力します。

   詳しくは、1222 ページの『プロセス参照のための接続の定義』を参照してください。一度に複数のシステムからプロセスをロードすること、および 1 つ以上の接続について プロセスのロード中に接続を切り替えることができます。

   注 : リモート・システムを参照する場合、Tivoli Enterprise Portal Server を通じて参照する場合にのみ、コマンド行詳細を使用できます。

次の例では、svchost.exe を選択すると、それが「**プロセス・モニター**」ページの「**プロセス名**」フィールドに表示されます (1221 ページの図 31)。

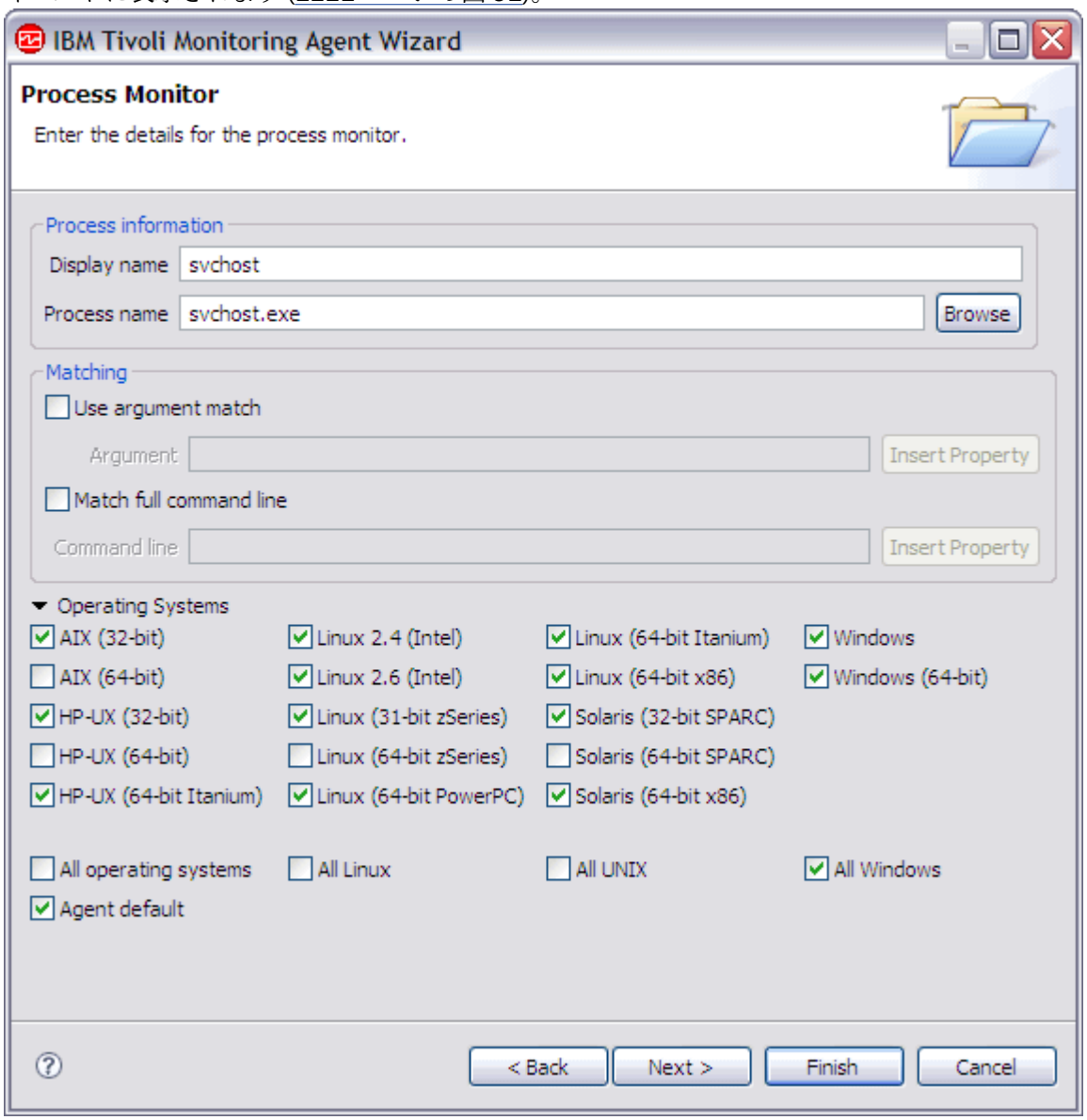


図 *31.* 「プロセス・モニター」ページの例

6. 1220 ページの表 265 の情報を使用して、「**プロセス・モニター**」ページに入力します。

   注 : このモニターで記述したプロセスが、 アプリケーションを実行するオペレーティング・システムの一部にしか適用できない場合は、 同じ表示名を持つプロセス・モニターを 1 つ以上作成して、 それ以外のオペレーティング・システムを カバーすることができます。一度に 1 つのプロセス・モニターを追加します。各モニターの表示名が同じになっている一方で、プロセス名は、選択したオペレーティング・システムに表示されるようにしてください。

7. 以下の手順のいずれかを行います。

   - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
   - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

次のタスク

このデータ・ソースのデータを IBM Cloud Application Performance Management の要約ダッシュボードで使用するには、フィルタリングされたデータ・セット (属性グループ) を可用性データ・セットに基づいて作成し、そのデータ・セットを単一行を生成するデータ・セットとして構成する必要があります。 プロセス用の行を選択するには、「名前」フィールドを使用します。

「状況」フィールドを使用して、プロセスの状況を確認することができます。このフィールドが「ダウン」になっている場合はプロセスが停止していることを表し、「稼働中」になっている場合はプロセスが稼働していることを表します。新しいフィルタリング属性グループで「状況」フィールドを選択し、重大度の値を指定してください。

プロセスの複数のコピーが実行されている場合は、そのプロセス名の複数の行が可用性データ・セット内に存在し、それらのすべての行に「稼働中」状況が含まれています。フィルタリングされたデータ・セットは、1 行だけを返すように構成する必要があります。これにより、上記のすべての行が返されるようになり、すべての場合において「状況」フィールドの値が正しく設定されるようになります。

詳しくは、以下を参照してください。

- 1341 ページの『フィルタリング属性グループの作成』
- 1201 ページの『状況標識として使用される属性の重大度の指定』
- 1373 ページの『Cloud APM 用のエージェントの準備』

プロセス参照のための接続の定義
プロセス・データ・ソースを定義すると、別のシステムからプロセスを表示および選択できます。ただし、エージェントが実行されるときには、エージェントと同じシステム上で実行されるプロセスがモニターされます。

このタスクについて
他のシステムの資格情報が必要です。または、それらのシステムを Tivoli Monitoring オペレーティング・システム・エージェントでモニターする必要があります。

手順

1. 接続を定義するには、「**プロセス・ブラウザー**」ウィンドウで 「**追加**」をクリックします。
   接続タイプ (セキュア・シェル (SSH)、Windows、または Tivoli Enterprise Portal Server 管理対象システム) を選択するか、既存の接続を選択してテンプレートとして使用することができます。

   管理対象システム接続を 追加するには、Tivoli Enterprise Server のホスト名、Tivoli Monitoring のユーザー名とパスワードが必要です。また、リモート接続の管理対象システム 名も必要です。管理対象システムを選択すると、リモート・システムにあるプロセスが表にリストされます。

   注 : OS エージェントが、参照するシステムで実行されている 必要があります。また、稼働中の Tivoli Enterprise Monitoring Server および Tivoli Enterprise Portal Server にエージェントが 接続されている必要があります。

   セキュア・シェル (SSH) 接続または Windows 接続を追加するには、ホスト名、 ユーザー名、およびパスワードが必要です。
2. 接続を追加すると、「**プロセス・ブラウザー**」ウィンドウの 「**接続名**」リストから接続を 選択できるようになります。

   接続の作成に必要なすべてのフィールド (パスワードなど) が保存されていない場合は、その接続の「**接続プロパティー**」ウィンドウが開きます。欠落している情報を入力してください。Tivoli Enterprise Portal Server 管理対象システム 接続の場合は、Tivoli Enterprise Portal Server に接続しなければ 管理対象システムに 入れません。
3. ユーザー名およびパスワードを入力してから「**最新表示**」(稲妻) アイコンをクリックして 接続した後、管理対象システムを選択してください。

次のタスク

接続を削除するには、接続を選択して「**編集**」をクリックし、「**接続プロパティー**」ウィンドウを開きます。「**この接続の削除**」チェック・ボックスを選択して 「**OK**」をクリックします。

Windows サービスのモニター

Windows システム上で実行される 1 つまたは複数のサービスをモニターするデータ・ソースを定義できます。サービスは、エージェントと同じホスト上で実行されている必要があります。サービスごとに、データ・ソースは、可用性データ・セットに 1 つの行を追加します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**プロセス**」をクリックします。
2. 「**データ・ソース**」領域で「**Windows サービス**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**サービス・モニター**」ページで、「**表示名**」フィールドに、説明を入力します。 「**サービス名**」フィールドに、サービス・アプリケーションの名前を入力します。これらは、手動で入力することも、「**参照**」をクリックして 現在ローカル・システムまたはリモート・システムで実行されているサービスのリストを 表示させることもできます。

   「**参照**」をクリックすると、「**サービス・ブラウザー**」ウィンドウが開きます。最初の状態では、このウィンドウには、Agent Builder システム上の各サービスの詳細情報が入っています。この情報には、サービスのサービス名、表示名、状態、および説明が含まれます。

   注 : Agent Builder が Windows システムで実行されていない場合、ローカル・サービスは表示されません。リモートの Windows システムを定義または選択する必要があります。1224 ページの『サービス参照のための接続の定義』を参照してください。

   注 : Tivoli Enterprise Portal Server、UNIX システム、または Linux システムから参照していない場合、サービス記述は無効になります。
5. 1 つ以上のサービスを選択するか、以下のステップのうちの 1 つ以上を実行して「**サービス・ブラウザー**」ウィンドウのリストを操作します。
   - サービスのリストをソートするために、列見出しをクリックする。
   - ウィンドウ内の情報を最新表示するために、「**最新表示**」(稲妻) アイコンをクリックする 。
   - サービスを検索するには、「**検索**」(双眼鏡) アイコンをクリックして 「**サービス検索**」ウィンドウを開きます。 サービス名、表示名、および説明で検索できます。
   - 別のシステムのサービスを表示させるには、 以前に定義したシステムを「**接続名**」リストから選択するか、 「**追加**」をクリックしてシステム 情報を入力します。 詳しくは、1224 ページの『サービス参照のための接続の定義』 を参照してください。 一度に 複数のシステムからサービスをロードすること、および 1 つ以上の接続について サービスのロード中に接続を切り替えることができます。
6. サービスの名前を選択するか入力したら、以下のいずれかのステップを実行します。
   - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
   - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

次のタスク

このデータ・ソースのデータを IBM Cloud Application Performance Management の要約ダッシュボードで使用するには、フィルタリングされたデータ・セット (属性グループ) を可用性データ・セットに基づいて作成し、そのデータ・セットを単一行を生成するデータ・セットとして構成する必要があります。 プロセス用の行を選択するには、「名前」フィールドを使用します。

新しいフィルタリング属性グループで「機能性テスト状況」フィールドを選択し、重大度の値を指定してください。

詳しくは、以下を参照してください。

- 1341 ページの『フィルタリング属性グループの作成』
- 1201 ページの『状況標識として使用される属性の重大度の指定』
- 1373 ページの『Cloud APM 用のエージェントの準備』

サービス参照のための接続の定義

Agent Builder が実行されているシステムのサービスを選択するほかに、他の Windows システムのサービスを選択することができます。

このタスクについて

他の Windows システムのサービスを選択するには、リモート・システムに対する接続を定義します。システムの資格情報をユーザーが持っているか、それらのシステムを Tivoli Monitoring オペレーティング・システム・エージェントでモニターする必要があります。

手順

1. 接続を定義するには、「**サービス・ブラウザー**」ウィンドウで「**追加**」をクリックします。

   「**接続タイプの選択**」ウィンドウが開きます。管理対象システム接続を 追加するには、Tivoli Enterprise Server のホスト名、Tivoli Monitoring のユーザー名とパスワード、 および管理対象システム 名が必要です。管理対象システムを選択すると、リモート・システムにあるサービスが表にリストされます。

   注 : OS エージェントは、参照しようとしているシステムの上で稼働している必要があります。また、稼働中の Tivoli Enterprise Monitoring Server および Tivoli Enterprise Portal Server に接続している必要があります。

   Windows 接続を追加するには、ホスト名、ユーザー名、およびパスワードが必要です。
2. 接続タイプ (Windows、または Tivoli Enterprise Portal Server 管理対象システム) を選択するか、既存の接続を選択してそれをテンプレートとして使用します。

   「**接続プロパティー**」ウィンドウが開きます。
3. 接続プロパティーを作成します。
4. 「**完了**」をクリックします。
5. 接続を追加すると、「**サービス・ブラウザー**」ウィンドウの 「**接続名**」リストから接続を 選択できるようになります。

   接続の作成に必要なフィールド (パスワードなど) が保存されていない場合は、「**接続プロパティー**」ウィンドウが開き、不足している情報を入力できます。

   a) Tivoli Enterprise Portal Server 管理対象システム 接続の場合は、Tivoli Enterprise Portal Server に接続しなければ管理対象システムに入れません。ユーザー名およびパスワードを入力してから「**最新表示**」(稲妻) アイコンをクリックして 接続した後、管理対象システムを選択してください。
6. 接続を削除するには、以下のステップを実行します。

   a)「**サービス・ブラウザー**」ウィンドウで接続を選択します。

   b)「**編集**」をクリックして、「**接続プロパティー**」ウィンドウを開きます。

   c)「**この接続の削除**」チェック・ボックスを選択します。

   d)「**OK**」をクリックします。

Windows Management Instrumentation (WMI) からのデータのモニター

データ・ソースを定義して、エージェントが実行されているシステム 上またはリモート・システム 上の Windows Management Instrumentation (WMI) からデータを収集することができます。データ・ソースは、単一の WMI クラスをモニターし、このクラスからのすべての値を、データ・ソースが生成するデータ・セット内に配置します。クラスが複数のインスタンスを提供する場合、データ・セットには複数行が含まれます。データ・セットの行が 1 つのみになるように、インスタンス名によってフィルタリングできます。

始める前に

Windows Management Instrumentation (WMI) を使用してリモート・システムからデータを収集するエージェントには、リモート・システム 上の WMI データにアクセスする許可が必要です。エージェントがリモート・システム 上の WMI データにアクセスできるのは、そのシステム 上の WMI データにアクセスする許可を持つアカウントの資格情報を指定した場合です。管理者アカウントは必要な許可を持っています。以下の手順では、管理者の資格情報か、必要な許可を持つ別のユーザーの資格情報のいずれかを指定できま

す。WMI データを参照する許可を持つユーザー・アカウントの作成について詳しくは、1364 ページの『Windows Management Instrumentation (WMI) 許可を持つユーザーの作成』を参照してください。

Windows API によってメトリックを収集するには、Windows オペレーティング・システム上でエージェントがホスティングされている必要があります。 リモート・システム上で、リモート・レジストリー管理が有効である必要があります。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**WMI**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**Windows Management Instrumentation (WMI) 情報**」ページで、以下のいずれかのステップを実行します。
   - フィールドに、WMI ネーム・スペースの名前と WMI クラス名を 入力します。 続いて、ステップ1225 ページの『9』に進んでください。
   - 「**参照**」をクリックして、システム上のすべての WMI クラスを表示します。

     リモート・システムを参照するには、リストからシステムを選択します (システムが定義済みである場合)。あるいは、「**追加**」をクリックして、Windows システムのホスト名を追加します。リモート・システム上の WMI データにアクセスする許可を持つユーザー・アカウントの資格情報を指定するか、リモート・システムの管理者の資格情報を指定します。ページが更新され、リモート・システムに関する情報が表示されます。参照を使用できるのは Agent Builder が Windows システムで実行される場合に限られ、参照できるのは、Windows システムのみです。

5. クラスの横の正符号 (+) をクリックしてクラスを展開し、属性を表示します。
6. リストから、関連した属性を持つクラスで指定したいものを選択し、 「**OK**」をクリックします。

   注 : 「**検索**」(双眼鏡) アイコンを クリックすると、選択したい項目をリストから見つけることが できます。「**検索語句**」フィールドに語句を入力し、「**名前で検索**」、「**クラス記述で検索**」、「**クラス・プロパティーで検索**」のいずれかのフィールドをクリックして希望の検索方法を指定して、「**OK**」をクリックします。検索対象の項目が見つかったら、それを選択して「**OK**」をクリックしてください。

   ウィザードの「**WMI 情報**」ページが再び開き、選択した WMI クラス情報が表示されます。
7. オプション: 「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1226 ページの『WMI 属性グループのテスト』を参照してください
8. オプション: 「**拡張**」をクリックして、 この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
9. 「**次へ**」をクリックします。

   注 : WMI ネーム・スペースおよび WMI クラス名を手動で入力した場合は、「**属性情報**」ページに切り替わります。このページで、属性情報を入力できます。属性を追加したい場合は、「**属性情報**」ページで「**追加属性の追加**」を選択できます。「**完了**」をクリックして完了します。
10. 「**キー属性の選択**」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』を参照してください。
11. 以下の手順のいずれかを行います。
    - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。
12. 属性を追加して、それらの属性に情報を指定することができます。詳しくは、1194 ページの『属性の作成』を参照してください。

    WMI データ・ソースの「**属性情報**」ページには、すべてのデータ・ソースに適用できるフィールド (1197 ページの表 261) に加えて、以下のフィールドがあります。

メトリック名
収集対象クラスから得られるプロパティー名。

13. データ・ソースのグローバル・オプションを設定する場合は、「**グローバル・オプション**」をクリックします

    このオプションを含める場合は、「**リモート Windows の構成プロパティーを含める** 」チェック・ボックスを選択して 「**OK**」をクリックします。

    Windows データ・ソースの Windows リモート接続構成について詳しくは、1363 ページの『Windows リモート接続の構成』を参照してください。

WMI 属性グループのテスト
Agent Builder を Windows システムで実行している場合は、Agent Builder 内で WMI 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**WMI 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**WMI テスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。
3. 「**エージェントの開始**」をクリックします。

   ウィンドウに、エージェントが開始されていることが表示されます。
4. エージェント・データに対するモニター環境の要求をシミュレートするには、「**データの収集**」をクリックします。

   エージェントが WMI に対してデータを照会します。「**WMI テスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
5. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
6. 「**エージェントの停止**」をクリックしてエージェントを停止します。
7. 「**OK**」または「**キャンセル**」をクリックして「**WMI テスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念
1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Windows パフォーマンス・モニター (Perfmon) のモニター

データ・ソースを定義して、Windows パフォーマンス・モニター (Perfmon) からデータを収集することができます。1 つのデータ・ソースは、1 つの Perfmon オブジェクトをモニターします。オブジェクト内のカウンターは、結果のデータ・セット内の属性に配置されます。クラスが複数のインスタンスを提供する場合、データ・セットには複数行が含まれます。データ・セットの行が 1 つのみになるように、インスタンス名によってフィルタリングできます。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**Perfmon**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**パフォーマンス・モニター情報**」 ページで、以下のステップのいずれかを行います。
   - 「**オブジェクト名**」フィールドにオブジェクトの 名前を入力し、 「**次へ**」をクリックして属性グループの最初の属性を 定義します。

     注 : Windows パフォーマンス・モニター オブジェクトの名前は英語で入力する必要があります。
   - 「**参照**」をクリックして Perfmon オブジェクトのリストを表示します。

     「パフォーマンス・モニター (Perfmon) オブジェクト・ブラウザー」ウィンドウが最初に開く際、ウィンドウにローカル・システムからの情報が取り込まれます。リモート・システムを参照するには、リストからシステムを選択するか (リストが定義済みの場合)、「**追加**」をクリックして Windows システムのホスト名を追加します。管理者 ID とパスワードを入力します。ウィンドウが更新され、リモート・システムに関する情報が表示されます。 参照を使用できるのは Agent Builder が Windows システムで実行される場合に限られ、参照できるのは、Windows システムのみです。例えば、リモート参照を行うために Linux または Solaris システムのホスト名を追加することはできません。

     – オブジェクト名をクリックすると、そのオブジェクトで使用可能なカウンターがウィンドウに表示されます。

       - Windows パフォーマンス・モニター のオブジェクトまたはカウンターをソートするためには、列見出しをクリックします。
       - ウィンドウ内の情報を最新表示するためには、「**最新表示**」をクリックします。
       - 特定のオブジェクトまたはカウンターを 検索するには、「**検索**」(双眼鏡) アイコンをクリックして「**パフォーマンス・モニター検索**」ウィンドウを 開きます。オブジェクト名を検索することも、カウンター名を検索することも、両方を検索することもできます。検索操作はサブストリングのマッチングを行い、大/小文字の区別を行いません。
       - オブジェクトを選択して「**OK**」をクリックします。
       - 「**パフォーマンス・モニター情報**」ページが、「**オブジェクト名**」フィールドで選択したオブジェクトの名前で開きます。
   - データ・ソースのグローバル・オプションを設定する場合は、「**グローバル・オプション**」をクリックします。

     このオプションを含める場合は、「**リモート Windows の構成プロパティーを含める** 」チェック・ボックスを選択して 「**OK**」をクリックします。

     Windows データ・ソースの Windows リモート接続構成について詳しくは、 1363 ページの『Windows リモート接続の構成』 を参照してください。

5. 選択した Windows パフォーマンス・モニター・オブジェクトが複数のインスタンスを返す場合、インスタンス名を基に結果をフィルタリングするには、以下のようにします。
   a) 「**パフォーマンス・モニター情報**」ページの「**Perfmon インスタンス名によるフィルタリング**」チェック・ボックスを選択します。
   b) 「**Perfmon インスタンス名**」フィールドで、フィルタリングするインスタンスの 名前を入力するか、「**参照**」をクリックして使用できるインスタンスのリストを表示します。
   c) リモート・システムを参照するには、リストからリモート・システムを選択するか、「**追加**」をクリックして、Windows システムのホスト名を追加します。ホストを選択したら、管理者 ID とパスワードを入力してください。 表が更新され、リモート・システムにあるインスタンスのリストが 表示されます。

   注 : 属性グループでフィルターに 掛けることもできます。ステップ 1228 ページの『9』 を参照してください。

6. 選択した Windows パフォーマンス・モニター・オブジェクトが複数のインスタンスを返す場合、インスタンス名が返されるようにするには、「**パフォーマンス・モニター情報**」ページの「**インスタンス名を戻す**」を選択します。

   このオプションにチェック・マークを付けると、属性リストに表示されない属性がデータ・ソースに追加されます。 この属性にはインスタンス名が含まれています。

   注 : 選択したオブジェクトを参照し、そのオブジェクトが 複数インスタンスを持つものとして定義されている場合、このチェック・ボックスが自動的に選択されます。
7. インスタンス名を戻すオプションにチェック・マークを付けなかった場合は、「キー属性の選択」ページが開きます。「キー属性の選択」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』 を参照してください。
8. オプション: 「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1228 ページの『Perfmon 属性グループのテスト』 を参照してください
9. オプション: 「**拡張**」をクリックして、 この属性グループから返されるデータを制限するフィルターを作成できます。

   属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』 のステップを参照してください。

   注 : インスタンス名でフィルターに掛けることもできます。ステップ 1227 ページの『5』 を参照してください。
10. 以下の手順のいずれかを行います。
    - 新規エージェント・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

      Agent Editor の「**データ・ソース定義**」ページに、オブジェクトとそのオブジェクトに 関する情報を含むリストが表示されます。
11. 属性を追加して、それらの属性に情報を指定することができます。詳しくは、1194 ページの『属性の作成』 を参照してください。

    すべてのデータ・ソースに適用可能なフィールドのほかに、データ・ソースの「**Perfmon 属性情報**」ページには以下のフィールドがあります。

    メトリック名
    特定オブジェクトのカウンターの 名前。

次のタスク

Perfmon データ・ソースの Windows リモート接続構成については、1363 ページの『Windows リモート接続の構成』 を参照してください。

Perfmon 属性グループのテスト

Agent Builder を Windows システムで実行している場合は、作成した Perfmon 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**パフォーマンス・モニター情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』 を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**パフォーマンス・モニターのテスト**」ウィンドウが表示されます。

2. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。
3. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
4. モニター環境からのエージェント・データの要求をシミュレートするには、「**データの収集**」をクリックします。

   エージェントがパフォーマンス・モニターに対してデータを照会します。「**パフォーマンス・モニターのテスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。

   注 : 「**データの収集**」ボタンを再びクリックするまで、すべての属性に対して有用なデータが表示されないことがあります。これは、一部のパフォーマンス・モニター属性はデルタ値を返しますが、デルタ値を計算するには前の値が必要であるためです。
5. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
6. 「**エージェントの停止**」をクリックしてエージェントを停止します。
7. 「**OK**」または「**キャンセル**」をクリックして「**パフォーマンス・モニターのテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Simple Network Management Protocol (SNMP) サーバーのデータのモニター

データ・ソースを定義して、SNMP サーバーをモニターすることができます。データ・ソースは、単一の SNMP オブジェクト ID (OID) および単一のホストからのすべてのデータをモニターします。OID 登録ツリーで、登録されている下位オブジェクトを持つ要素を選択すると、スカラー値または表の値の別個のセットごとにデータ・セットが作成されます。 オブジェクトがスカラー・データを返す場合、データ・セットには単一行が含まれます。オブジェクトが表データを返す場合、データ・セットには複数行が含まれます。

このタスクについて

エージェントでは、Simple Network Management Protocol V1、V2C (このバージョンが単なる V2 ではなく V2C であることに注意)、および V3 がサポートされています。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**SNMP**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「Simple Network Management Protocol (SNMP) 情報」ページで、表示名を入力するか、「**参照**」をクリックしてシステム上のすべてのオブジェクトを表示します。

   データ・ソースの定義後は、属性を追加できます。 これらの属性の OID は長く、正確に入力しにくい場合があります。 参照を使用すると、正しい OID を容易に入力できます。

   注 : ブラウザーは、実際のシステムを参照するのではなく、SNMP 定義である管理情報ベース (MIB) を読み取ります。

   注 : 「**最新表示**」アイコンをクリックすると、解析済み MIB ファイルのメモリー内バージョンが消去され、ワークスペース・キャッシュ内のこのファイルが再解析されます。キャッシュは次のロケーションにあります: *workspace_directory*¥.metadata¥.plugins¥com.ibm.tivoli.monitoring.agentkit¥mibs

各部の意味は次のとおりです。

workspace_directory
Agent Builder を初めて実行したときに指定したワークスペース・ディレクトリーを示します (1170 ページの『Agent Builder の開始』 を参照)。

a) 目的のオブジェクトを定義した MIB がロードされない場合、「**カスタム MIB の管理**」をクリックして「カスタム MIB の管理」ダイアログを開きます。

b)「**追加**」をクリックして、追加する MIB ファイルを参照します。キャッシュから MIB を削除するには、削除する MIB を選択して「**削除**」をクリックします。

c)「**OK**」をクリックしてキャッシュを更新します。

MIB の構文解析時にエラーが発生した場合、「カスタム MIB の管理」ダイアログは開いたままとなります。このダイアログで、MIB を追加または削除してエラーを解消できます。

「**キャンセル**」をクリックすると、MIB キャッシュはダイアログが開いた時の状態に戻ります。

Agent Builder には以下の MIB のセットが含まれています。

- hostmib.mib
- rfc1213.mib
- rfc1243.mib
- rfc1253.mib
- rfc1271.mib
- rfc1286.mib
- rfc1289.mib
- rfc1315.mib
- rfc1316.mib
- rfc1381.mib
- rfc1382.mib
- rfc1443.mib
- rfc1461.mib
- rfc1471.mib
- rfc1493.mib
- rfc1512.mib
- rfc1513.mib
- rfc1516.mib
- rfc1525.mib
- rfc1573a.mib
- rfc1595.mib
- rfc1650.mib
- rfc1657.mib
- rfc1659.mib
- rfc1666.mib
- rfc1695.mib
- rfc1747.mib
- rfc1748.mib
- rfc1757.mib

- rfc1903.mib
- rfc1907.mib
- rfc2011.mib
- rfc2021.mib
- rfc2024.mib
- rfc2051.mib
- rfc2127.mib
- rfc2128.mib
- rfc2155.mib
- rfc2206.mib
- rfc2213.mib
- rfc2232.mib
- rfc2233.mib
- rfc2238.mib
- rfc2239.mib
- rfc2320.mib
- rfc3411.mib

これらの MIB はすべて、標準の IETF 定義 MIB です。MIB が含まれているのは、これらがモニターに役立つ可能性のある共通定義を表しているためです。また、カスタム MIB がインポートするシンボルを解決できるようにするには MIB の多くが必要なためです。

d) リストから SNMP オブジェクトを選択します。

オブジェクトの横の正符号 (+) をクリックしてオブジェクトを展開し、レベルを表示します。

e) このリストから指定したいオブジェクトを選択し、「**OK**」をクリックしてください。

新規データ・ソースが「**データ・ソース定義**」ページにリストされます。

注 : 他のオブジェクト (最初のオブジェクトの下にネストされたオブジェクト) を定義するオブジェクトを選択すると、それらのすべてのオブジェクトがデータ・ソースになります。上位のオブジェクトを選択すると、多数のデータ・ソースが追加されます。

5. 「**Simple Network Management Protocol 情報**」ページで、オペレーティング・システムを選択します。
6. オプション: 「**Simple Network Management Protocol 情報**」ページで、「**テスト**」をクリックして、データ・ソースをテストできます。

   テストについて詳しくは、1233 ページの『SNMP 属性グループのテスト』を参照してください
7. オプション: 「**拡張**」をクリックして、 この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
8. 「**次へ**」をクリックします。
9. 「**属性情報**」ページで、属性の情報を指定します。
10. 以下の手順のいずれかを行います。
    - 新規エージェント・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。
11. 属性の追加および属性の情報の指定について詳しくは、1194 ページの『属性の作成』を参照してください。

    SNMP データ・ソースの「**属性情報**」ページには、すべてのデータ・ソースに適用できるフィールドに加えて、以下のフィールドがあります。

メトリック名
任意のストリング

オブジェクト ID
オブジェクトに登録済みのフル OID です (索引値は含みません)

次のタスク

エージェントのランタイム構成を使用して、モニター対象ホストを設定できます。

Agent Builder が、64 ビット・データ・タイプを生成して、32 ビット符号なし MIB プロパティーの最大値を処理できるようにする方法については、1232 ページの『SNMP MIB 構文解析オプション』を参照してください。

SNMP MIB エラー
SNMP MIB のエラーの処理。

SNMP MIB の追加時にエラーが見つかるのはよくあることです。MIB エラーの内容を確認するには、**Agent Builder の「エラー」**ウィンドウで「**詳細>>**」をクリックします。

最も一般的なエラーの 1 つは、他の MIB 内で定義されている定義が欠落していることです。複数の MIB を同時にインポートすることで、この問題を解決できます。欠落している定義がすべて解決されるまで、MIB を増分的に追加することもできます。Agent Builder では、解決されたすべての定義を使用できます。したがって、使用する予定のない MIB の部分のみに影響するエラーを無視するという選択肢もあります。 MIB がすべてロードされた後で参照が解決されるため、MIB の順序は関係ありません。

SNMP MIB 構文解析オプション
SNMP MIB 構文解析の設定

手順

1. Agent Builder で、「**ウィンドウ**」 > 「**設定**」を選択して、「**設定**」ウィンドウを開きます。
2. ナビゲーション・ペインで、「**IBM Agent Builder**」を展開します。
3. 「**MIB 構文解析**」をクリックして、「**MIB 構文解析**」ウィンドウを開きます。

   Agent Builder が使用する MIB パーサーは、ASN.1 によって定義された文法を使用して MIB を構文解析します。一部の MIB が文法に正しく従っていません。パーサーは、起こりがちなエラーを許容するように特定の規則を緩和できます。これらの規則を緩和することで、非準拠の MIB を構文解析できます。

   小文字で始まるタイプを許可する
   MIB に記述するタイプを許容します (values など)。

   指定された数値を許可する
   大文字で始まる数値を許容します。

   値名に下線を許可する
   下線文字を許容します。

   大文字で始まる値を許可する
   大文字で始まる値を許容します。

   重複する MIB は無視する
   重複する MIB モジュールの警告をオフにします。
4. オプション: 「**32 ビット符号なし MIB プロパティー用の 64 ビット属性を作成する**」チェック・ボックスを選択すると、Agent Builder が、64 ビット・データ・タイプを生成して、32 ビット符号なし MIB プロパティーの最大値を処理できるようになります。 このオプションを選択しても、既存のエージェント・フィールド定義は変更されません。 MIB ファイルを参照して、これらのプロパティーの新規データ・ソースを作成する必要があります。
5. 設定の編集が完了したら、「**OK**」をクリックします。

SNMP 属性グループのテスト

Agent Builder 内で作成した SNMP 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**Simple Network Management Protocol 情報**」ページで、「**テスト**」をクリックします。

     注 :

     選択された SNMP オブジェクトに複数の属性グループが含まれる場合は、テスト対象の属性グループを選択するように要求するプロンプトが出されます。
   - エージェントの作成後に、**Agent Editor** の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「SNMP テスト設定」ウィンドウが開きます。
2. 「**接続名**」から既存の接続を選択するか、「**追加**」をクリックすると、接続タイプを選択するためのプロンプトが表示されます。あるいは、「**接続ウィザードの作成**」を使用して、テンプレートとして使用する既存の接続を選択します。
3. 接続タイプまたは既存の接続を選択した後で、「**次へ**」をクリックして、SNMP 接続プロパティーを指定します。 完了したら、「**完了**」をクリックして「SNMP テスト設定」ウィンドウに戻ります。
4. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。
5. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
6. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。エージェントは、構成済み SNMP 接続でデータを照会します。
7. 「**テスト設定**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
8. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
9. 「**エージェントの停止**」をクリックしてエージェントを停止します。
10. 「**OK**」または「**キャンセル**」をクリックして「**設定のテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Simple Network Management Protocol イベント送信側からのイベントのモニター

データ・ソースを定義して、SNMP のトラップ・イベントおよび通知イベントからデータを収集することができます。エージェントのランタイム構成でポートを設定し、このポートでエージェント・ホストにイベントを送信するようにサーバーを構成する必要があります。すべてのモニター対象イベントは、データ・セット内に行として配置されます。

このタスクについて

エージェントでは、Simple Network Management Protocol (SNMP) V1、V2C (このバージョン名は単なる V2 ではなく V2C)、および V3 がサポートされています。SNMP トラップおよび通知は、エージェントが受

け取って処理することができます。このプロバイダーによって受け取られたデータは、イベントとしてモニター環境に渡されます。

SNMP イベントの属性グループについて詳しくは、1448 ページの『SNMP イベントの属性グループ』を参照してください。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**SNMP イベント**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**Simple Network Management Protocol イベント情報**」ウィンドウで、以下の手順のいずれかを行います。
   - 「**すべてのイベント**」をクリックして、SNMP イベントを受け取った場合に イベントを送信する属性グループを作成します。
   - 「**総称イベント**」をクリックして、選択した総称イベント・タイプのいずれかに一致する総称 SNMP イベントを受け取った場合に イベントを送信する属性グループを作成します。
   - 「**カスタム・イベント**」をクリックして、エンタープライズ固有の SNMP イベントの場合にイベントを送信する属性グループを 1 つ以上作成します。「**参照**」をクリックしてモニターするイベントを選択します。

     「**Simple Network Management Protocol (SNMP) 管理情報ベース (MIB) ブラウザー**」ウィンドウでは、選択ペイン内のイベントは、それらのイベントが定義されている MIB モジュールごとに編成されます。SNMP オブジェクトを展開すると、 その MIB モジュール内のイベントが表示されます。リストで、 指定するオブジェクトをクリックして「**OK**」をクリックします。

     トラップの静的データを含むトラップ構成ファイルがある場合は、「**トラップ構成ファイルで定義されている情報を表示する属性を含める**」チェック・ボックスを選択します。SNMP トラップ構成ファイルについて詳しくは、1501 ページの『SNMP トラップ構成』を参照してください。

     トラップ・プロトコル・データ単位 (PDU) で受信したすべての変数バインディング (VarBind) データを含む属性を組み込む場合は、「**変数バインディング (VarBind) データ属性を含める**」チェック・ボックスを選択します。この属性について詳しくは、属性定義 (1448 ページの『SNMP イベントの属性グループ』) を参照してください。

     注 :

     a. ブラウザーは、実際のシステムを参照するのではなく、 定義および管理情報ベース (MIB) を読み取ります。Agent Builder に組み込まれている MIB のリストは、1229 ページの『Simple Network Management Protocol (SNMP) サーバーのデータのモニター』で定義しています。 いずれの SNMP データ・プロバイダーによってロードされた MIB であっても、両方から使用可能です。

     b. MIB モジュールまたは個々のイベントを選択すると、そのモジュール内のすべてのイベントがそれぞれ別個のデータ・ソースに変換されます。 イベントで定義された変数ごとに 1 つの属性が追加されます。 選択したモジュールまたはトラップのすべてのイベントが単一のイベント・ソースに送られるようにするには、「**単一の属性グループのイベントを収集する**」チェック・ボックスを選択します。個々のトラップを選択し、「**単一の属性グループのイベントを収集する**」フラグを選択した場合は、各イベントで定義された変数ごとに 1 つの属性が追加されます (重複する変数は無視されます)。モジュールを選択した場合は、変数属性は追加されません。

     c. 独自のフィルターを 入力する場合は、 以下の構文を使用してください。

        OID (オブジェクト ID) 要素の値を使用して、 この属性グループの場合に処理する トラップを判別します。

– **トラップの突き合わせ:** global_snmp_event_settings_for_group 要素の OID 属性は、トークンをコンマで区切った リストにすることができます。単一のトークンの構文は以下のとおりです。

```
[enterpriseOID][-specificType]
```

– **例:** "1.2.3.5.1.4,1.2.3.4.5.6.7.8.9-0" 最初のトークンは、エンタープライズ OID が 1.2.3.5.1.4 であるすべてのトラップに一致します。 2 番目のトークンは、エンタープライズ OID が 1.2.3.4.5.6.7.8.9、固有タイプが 0 であるすべてのトラップに一致します。 トークンは 1 つの属性グループで一括してリストするため、受け取ったイベントがいずれかに一致すると、その属性グループによって処理されます。

d. 受け取ったイベントはいずれも、受け取ったイベントに一致する 最初の属性グループのみによって処理されます。最初に サブノード属性グループが処理され、次に基本属性グループが 処理されます。エージェントの開発者は、 イベントが確実に所定の属性グループで受け取られるようにグループを定義する 必要があります。

5. イベントをサブノードと突き合わせるには、「**SNMP イベント情報**」ウィンドウ で「**サブノード・ホストの突き合わせ**」チェック・ボックスを選択します。SNMP イベントの属性グループがサブノードの一部である場合は、「**サブノード・ホストの突き合わせ**」チェック・ボックスを選択して、モニターする SNMP エージェントから来るイベントである必要があるかどうかを制御できます。

   例えば、ルーターをモニターするエージェントがあり、各サブノード・インスタンスが特定のルーターを表すとします。SNMP データ・コレクターでルーターからデータを収集するエージェントを開発します。 そのルーターによって送信された SNMP イベントを受け取る属性グループも定義します。各ルーター・インスタンスには、イベント・フィルターに対して定義されたものと同じデータが含まれます。そのため、ルーターからのイベントがそのルーターの属性グループに表示されるようにするための別の手段が必要です。

   サブノード・ホストの突き合わせを選択すると、 ルーターによって送信されたイベントが、SNMP データ・コレクターに対して 定義されているホストと比較されます。SNMP データ・コレクターによって使用されているホストが、受け取ったイベントの送信元ホストと同じである場合は、サブノード・インスタンスが SNMP イベントを処理します。それ以外の場合、イベントは次のサブノード・インスタンスに 渡されます。アドレスの突き合わせはサブノードのみに適用されます。基本エージェントの SNMP イベントの属性グループは アドレスの突き合わせを行いません。アドレスの突き合わせを機能させるには、サブノード定義に少なくとも 1 つの SNMP 属性グループを含める 必要があります。そのサブノード・インスタンスに対して SNMP が使用する SNMP ホストが、 突き合わせに使用されるホストになります。

   「**サブノード・ホストの突き合わせ**」チェック・ボックスをクリアした場合、サブノード・インスタンスはこの追加の比較を行いません。この場合は、 ユーザーがサブノードごとに別個の OID フィルターを構成できるようにする 必要があります。それ以外の場合は、サブノード定義に SNMP イベントの属性グループを含める必要はありません。

6. 「**SNMP イベント情報**」ウィンドウで オペレーティング・システムを選択します。
7. オプション: 「**SNMP イベント情報**」ウィンドウで「**テスト**」をクリックして、エージェントを開始およびテストできます。

   詳しくは、1237 ページの『SNMP イベントの属性グループのテスト』を参照してください。
8. オプション:

   「**SNMP イベント情報**」ウィンドウで、「**拡張**」をクリックし、「**イベント・フィルタリングおよび要約オプション**」を選択します。詳しくは、1405 ページの『イベント・フィルタリングおよび要約』を参照してください。

   a) 「**イベント・フィルタリングおよび要約オプション**」での選択が完了したら、「**SNMP イベント情報**」ウィンドウに戻ります。「**SNMP イベント情報**」ウィンドウで以前に「**カスタム・イベント**」を選択した場合は、「**次へ**」をクリックして、キー属性を選択します。選択しなかった場合は、次のステップをスキップします。

   b) 「キー属性の選択」ページで、属性グループの 1 つ以上のキー属性をクリックするか、「**単一データ行の生成**」をクリックします。
9. 「**次へ**」をクリックします。 新規エージェント・ウィザードを使用してエージェントを保存し、Agent Editor で開く場合は、 「**完了**」をクリックします。

10.

次のタスク
その他の属性の追加について詳しくは、1194 ページの『属性の作成』を参照してください。

SNMP イベントの構成プロパティー
SNMP イベント属性グループがエージェントに追加されると、特定の構成プロパティーが自動的に作成されます。

データ・ソースが追加されると、Agent Editor の「**ランタイム構成情報**」ページに構成が表示されます。例えば、1236 ページの図 32 には、構成セクションと一部の構成プロパティー (SNMP イベントの属性グループがエージェントに追加されるときに自動的に作成されます) が表示されています。

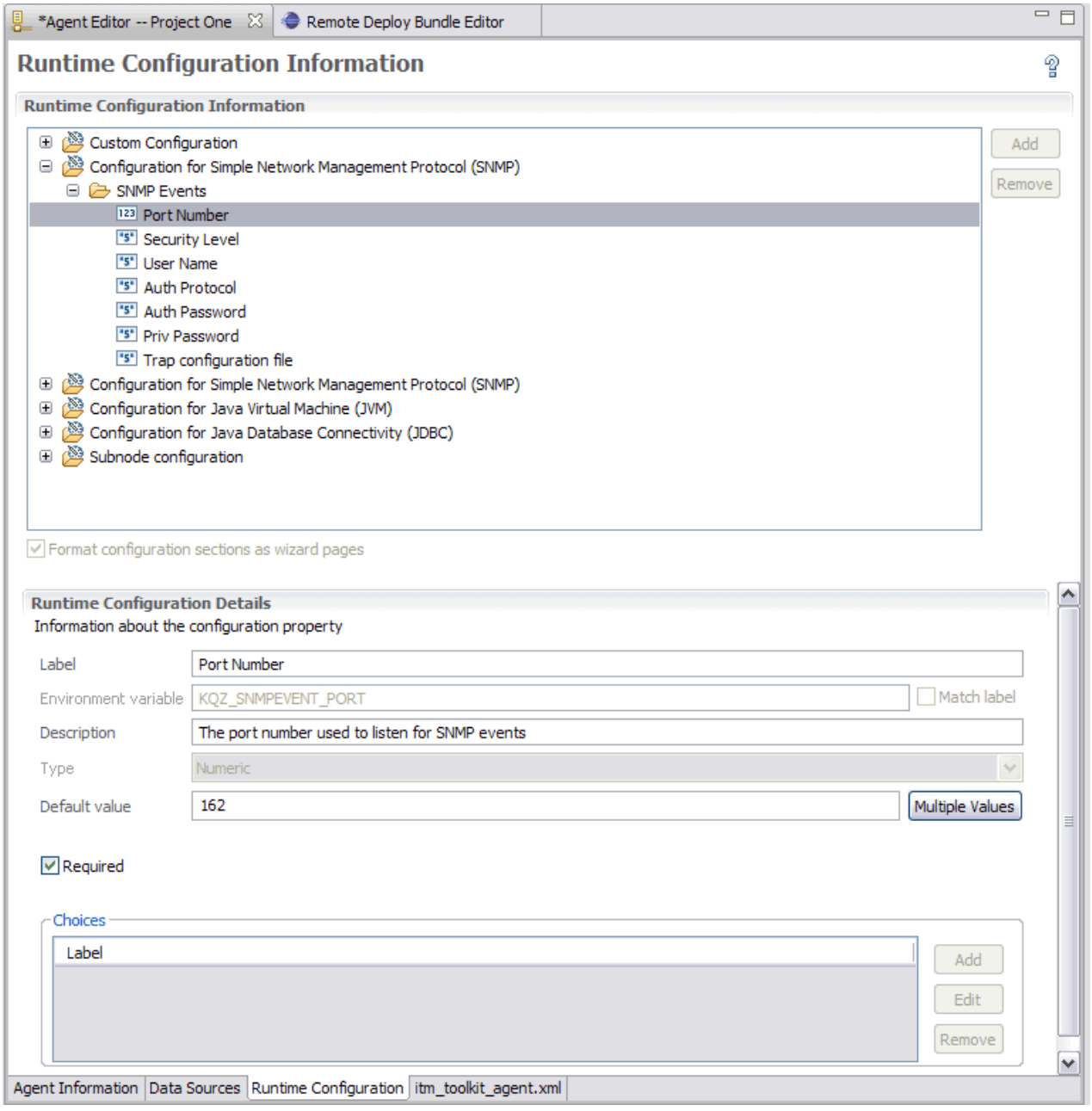


図 32. 「ランタイム構成」ページ

事前定義構成プロパティーのラベル、説明、およびデフォルト値は 変更できますが、変数名と変数のタイプは変更できません。 「SNMP イベント構成」セクションには 以下のプロパティーがあります。

表 266. SNMP イベント構成プロパティー			
Name	**有効な値**	**必須**	**説明**
ポート番号	正整数	はい	イベントの listen に使用する必須のポート番号
セキュリティー・レベル	noAuthNoPriv、authNoPriv、 authPriv	いいえ	SNMP V3 セキュリティー・レベル
ユーザー名	ストリング	いいえ	SNMP V3 ユーザー名
認証プロトコル	MD5 または SHA	いいえ	SNMP V3 認証プロトコル
認証パスワード	ストリング	いいえ	SNMP V3 認証パスワード
プライバシー・パスワード	ストリング	いいえ	SNMP V3 プライバシー・パスワード
トラップ構成ファイル	パスを含むファイル名	いいえ	トラップ構成ファイルの場所。この構成プロパティーでファイルの場所が指定されていない場合は、エージェントの bin ディレクトリーで `trapcnfg` ファイルの検索が試行されます。

V1 イベントまたは V2C イベントの場合、構成は不要です。V1 イベントまたは V2C イベントは、ソースまたはコミュニティーの名前が指定されているかどうかにかかわらず、すべて処理されます。 サポートされるプライバシー・プロトコルは DES のみであるため、プライバシー・プロトコルを 指定するためのオプションはありません。SNMP V3 の構成オプションは 必須ではありません (それぞれオプションで指定することはできます)。指定する場合は、選択したセキュリティー・レベルに応じた適切な値を指定してください。

SNMP イベントの属性グループのテスト

Agent Builder 内で作成した SNMP イベント属性グループをテストできます。

始める前に

SNMP イベント属性グループをテストするには、テスト・プログラムまたはアプリケーションを使用して、SNMP イベントを生成します。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**SNMP イベント情報**」ウィンドウで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**イベントの設定のテスト**」ウィンドウが開きます。
2. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。SNMP イベント構成プロパティーについて詳しくは、1236 ページの『SNMP イベントの構成プロパティー』を参照してください。
3. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。

   エージェントが開始されると、エージェントはその構成に従って SNMP イベントを listen します。

注 : 開始されるエージェントは、テストする属性グループを 1 つ含む簡素化バージョンです。

4. エージェントのデータ収集をテストするには、エージェントの構成に一致する SNMP イベントを生成します。これは、アプリケーションまたはイベント・ジェネレーターを使用して実行できます。

   エージェントがその構成に一致する SNMP イベントを受信すると、エージェントはイベントを内部キャッシュに追加します。

5. モニター環境からのエージェント・データの要求をシミュレートするには、「**データの収集**」をクリックします。

   「**イベントの設定のテスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのイベントが収集および表示されます。データ収集の例を 1238 ページの図 33 に示します。

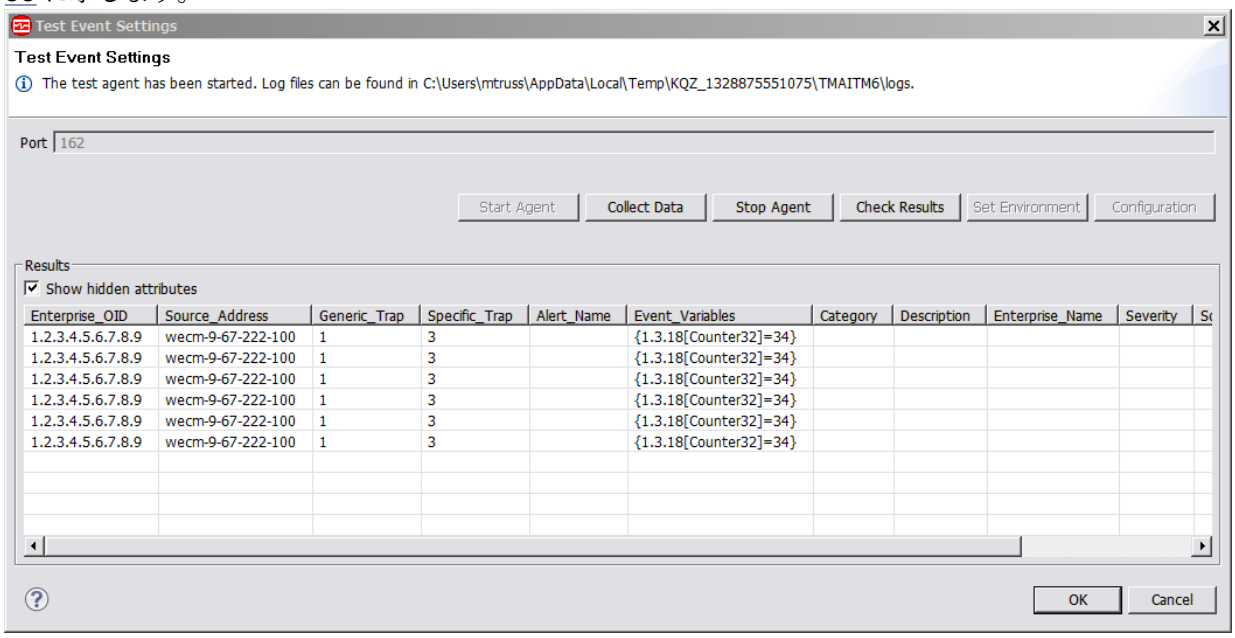


図 *33.* 収集された *SNMP* イベント・データが表示された**「イベントの設定のテスト」**ウィンドウ

6. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。その例を 1238 ページの図 34 に示します。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。

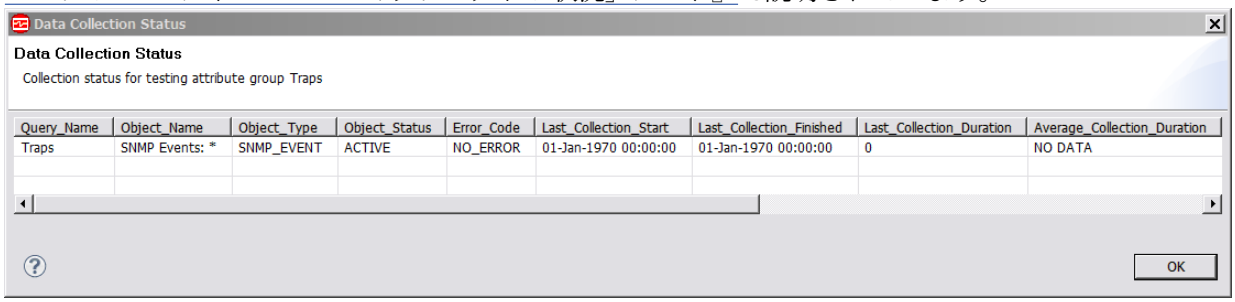


図 *34.* **「データ収集状況」**ウィンドウ

7. 「**エージェントの停止**」をクリックしてエージェントを停止します。
8. 「**OK**」または「**キャンセル**」をクリックして「**イベントの設定のテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』

Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Java Management Extensions (JMX) MBean のモニター

データ・ソースを定義して、JMX MBean からデータを収集することができます。モニター対象の各 MBean からのデータは、データ・セット内に配置されます。MBean に応じて、データ・セットは単一行を生成する場合と複数行を生成する場合があります。

このタスクについて

定義する JMX データ・ソースごとに、単一の MBean (単一インスタンス) または特定タイプの MBean (複数インスタンス) を指定する必要があります。収集したいデータが含まれている MBean のオブジェクト名、または MBean タイプのオブジェクト名パターンを知っておく必要があります。類似の MBean のセットを識別するだけであれば、オブジェクト名パターンを使用してください。そのパターンに合致する MBean セットのすべての MBean には、モニター表に表示するデータが組み込まれていなければなりません。典型的なオブジェクト名パターンは、`*:j2eeType=Servlet,*` のようになります。このオブジェクト名パターンは、j2eeType が servlet になっているすべての MBean に合致します。 通常、そのパターンに合致するすべての MBean には、データ・ソースに追加できる一連の公開済みの属性と操作が含まれています。そのパターンを使用するデータ・ソースは、そのパターンに合致するそれぞれの MBean からデータを収集します。そのデータ・ソースで定義する属性は、データ・ソースのオブジェクト名パターンに合致するすべての MBean で使用可能な属性でなければなりません。

Java バージョン 5 以降がサポート対象です。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」域で「**JMX**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**JMX 情報**」ページで、「**参照**」をクリックして MBean サーバー上のすべての JMX MBean を表示します。

   データ・ソースの定義後に参照機能を使用すると、属性リストにあらかじめデータを取り込むことができます。 その後、ブラウザーが挿入した属性の追加、削除、または変更を行うことができます。 これらの属性の名前は長く、正確に入力しにくい場合があります。 参照オプションを使用すると、正しい名前を容易に入力できます。

   注 : ブラウザーを使用せずにオブジェクト名を指定して「**次へ**」をクリックすると、手動で JMX データ・ソースを作成できます。手動で JMX データ・ソースを作成すると 2 つのデータ・ ソースが作成されます。 JMX 通知用の事前定義された属性を含むイベント・データ・ソースが 1 つ作成されます。また、ウィザードで指定する必要のある 1 つの属性を含むコレクション・データ・ソースが定義されます。

   MBean パターン
   MBean パターンを表示します。

   グローバル JMX オプション
   サポートのレベルを表示します。

   以下の JMX サーバーのサポートが用意されています。

   - Java 5 オペレーティング・システムの MBean サーバー。JSR-160 コネクターで接続します。通知とモニターがサポートされています。
   - WebSphere Application Server、バージョン 6 以降。SOAP プロトコルと RMI プロトコルの両方に対応したコネクターが用意されています。リモート・エージェントでは MBean を作成できないので、JMX モニターはサポートされていません。
   - WebSphere コミュニティー版と他の Apache Geronimo ベースのアプリケーション・サーバー。標準の JSR-160 コネクターで接続します。バージョン 1.1 以降では、JMX の通知とモニターがサポートされています。

- JBoss アプリケーション・サーバー、バージョン 4.0 以前。
- JBoss Application Server、JSR-160 接続。
- WebLogic Server、バージョン 9 以降。T3 プロトコルに対応したコネクターが用意されています。

5. 初めて JMX ブラウザーを実行したときには、「**MBean サーバー**」スクロールダウン・メニューには項目がありません。接続を追加するには、「**追加**」ボタンをクリックします。

   「**編集**」ボタンを使用して、既に定義済みの、スクロールダウン・メニューで選択した接続を変更または削除します。接続定義はワークスペースに格納されるので、作成した接続は記憶されています。接続を作成するには、以下の手順を実行します。接続が既に存在する場合は、次の手順にスキップします。

   a) MBean サーバーに対する接続を作成するには、「**追加**」をクリックして、接続を追加するか、既存の接続を編集します。

      接続が定義されていない場合は、「**Java Management Extensions (JMX) ブラウザー**」ウィンドウが表示されます。

   b)「**追加**」をクリックして接続を追加すると、「**接続タイプの選択**」ページが開きます。

   c) MBean サーバー接続ウィザードを使用して、MBean サーバーに接続します。接続を作成するときには、ページに表示される新しい接続のリストの中から選択できます。既存の接続のリストを使用すると、既存の接続をテンプレートとして使用して、新しい接続を作成できます。新しい接続タイプのいずれかを選択し、「**次へ**」をクリックして、接続の作成を開始します。

   d) 接続タイプを選択した後に、より具体的な接続タイプを選択するように求める画面が表示される場合もあります。「**標準 JMX 接続 (JSR-160)**」接続タイプに基づく 2 つのテンプレートが表示されます。ご使用の MBean サーバーに最適のテンプレートを選択して、「**次へ**」をクリックします。

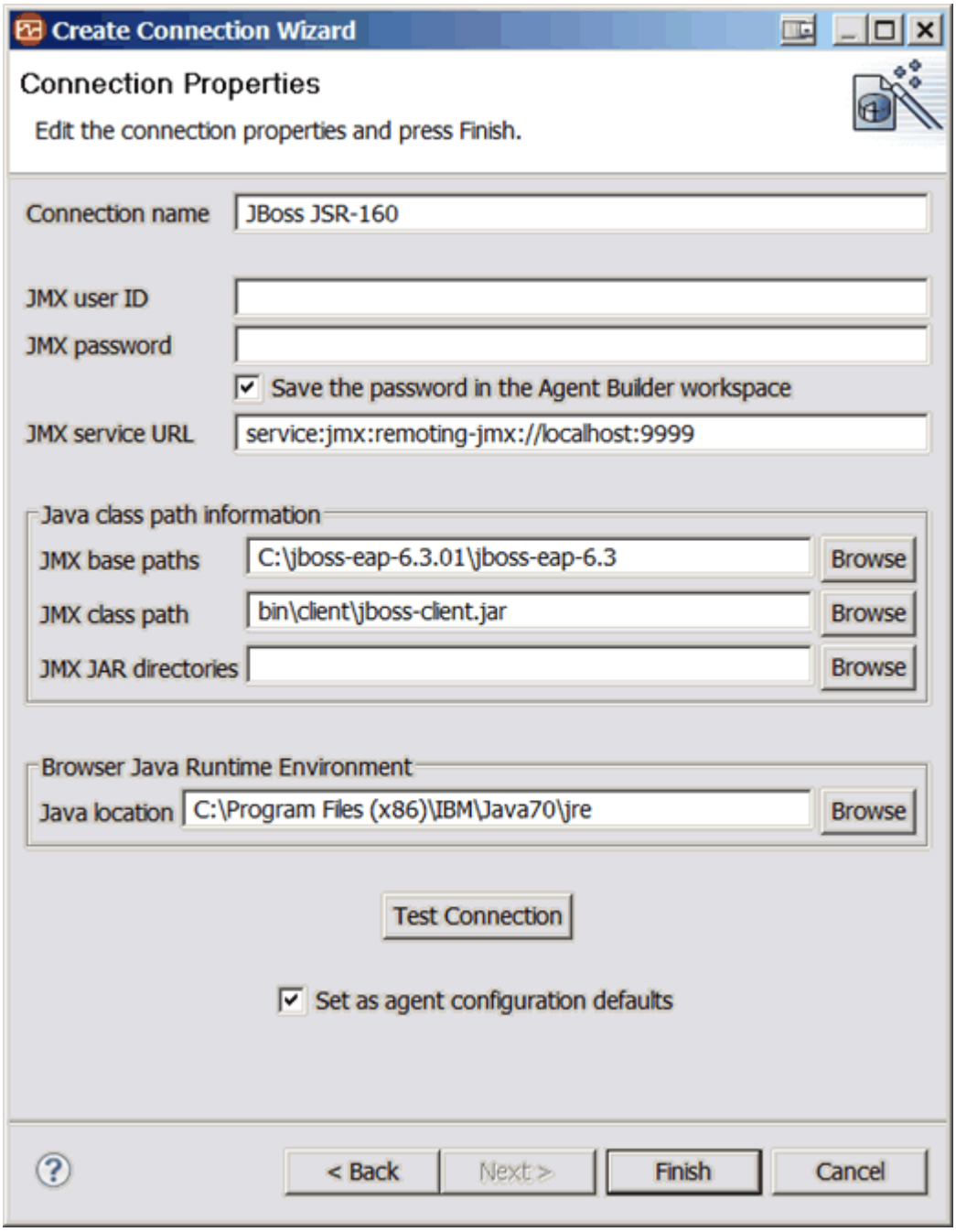


図 *35. JMX* 接続プロパティー

「**接続プロパティー**」ページ (1241 ページの図 35) には、MBean サーバーに接続するための詳しい情報が表示されます。MBean サーバーの詳細情報をこのページに入力する必要があります。

重要 : データ・ソースがリモートの WebSphere Application Server に接続されている場合は、Agent Builder が稼働しているホストにも WebSphere Application Server をインストールし、ローカルの WebSphere Application Server が使用する Java ランタイム環境として **Java location** 設定を指定してください。

e) この接続のパスワードを保存する場合は、「**パスワードを Agent Builder ワークスペースに保存**」チェック・ボックスを選択します。

f) オプション: JMX のデフォルトをこれらの接続プロパティーからコピーする 場合は、「**エージェント構成のデフォルトとして設定**」を選択してください。

例えば、1241 ページの図 35 では、デフォルトの 「**JMX 基本パス**」は `C:¥jboss-eap-6.3.01¥jboss-eap-6.3`、「**JMX サービス URL**」は `service:jmx:remoting-jmx://localhost:9999`、「**Java ロケーション**」は `C:¥Program Files¥IBM¥Java70¥jre` です。

1) 接続に必要なプロパティーを指定したら、「**接続のテスト**」をクリックして、接続を確立できるかどうかを確認します。接続が成功しない場合は、必要なプロパティーを訂正します。
2) 接続が成功したら、「**完了**」をクリックしてブラウザーに戻ります。構成した接続が、ブラウザーで使用されます。

「**接続プロパティー**」ページの Java クラスパス情報には 3 つのフィールドがあります。 これらのフィールドには、Java ランタイム環境に含まれていない Java クラスを必要とする MBean サーバーに接続するための情報を、必要に応じて入力してください。通常、接続先の MBean サーバーは、Agent Builder と同じシステムにインストールする必要があります。その場合は、MBean サーバーが含まれているアプリケーションのインストール先のディレクトリーを「**JMX 基本パス**」フィールドに指定します。これにより、「**JMX JAR ディレクトリー**」フィールドに、「基本パス」ディレクトリーに関連するディレクトリーがリストされます。これらのディレクトリーには、MBean サーバーへの接続に必要な JAR ファイルが格納されています。 「**JMX クラスパス**」フィールドを使用して、特定の JAR ファイルを組み込むことができます。「**JMX JAR ディレクトリー**」フィールドに表示されている JAR ファイルを、「**JMX クラスパス**」フィールドに別個にリストする必要はありません。

どのフィールドにも 複数の参照項目を組み込めます。複数の項目を指定する場合は、セミコロンで区切ってください。それらの値は、エージェントを構成するときに必要になるのと同じ値です。詳しくは、1246 ページの『JMX 構成』 を参照してください。

6. 接続を選択したら、JMX ブラウザーが JMX サーバーから MBean に関する情報をダウンロードします。この情報は、「**JMX ブラウザー**」ウィンドウ (1243 ページの図 36) の以下の 4 つの領域に表示されます。

「Java Management Extensions (JMX) ブラウザー」ウィンドウから Agent Editor の「**ランタイム構成**」タブまでの画面に関する指示: 「**JMX 情報**」ページで「**参照**」を選択します。ブラウザー (接続が選択されていない JMX ブラウザー) で「**追加**」を選択します。「**JMX 接続の選択**」ページで「**JBoss**」」を選択し、「**次へ**」を選択します。「**JMX 接続プロパティー**」ページで、2 つの接続プロパティーをカスタマイズします (JBoss プロバイダー URL: `jnp://wapwin3.tivlab.raleigh.ibm.com:1099/` および「**JBoss JAR ディレクトリー**」: `jbossall-client.jar`、`jboss-jmx.jar`、`jboss-jsr77-client.jar`、`jboss-management.jar` という JAR ファイルを含むディレクトリーへの絶対パス)。「**完了**」を選択します。この構成により、JBoss 接続がセットアップされ、ここに示されているのと同じような画面が表示されます。

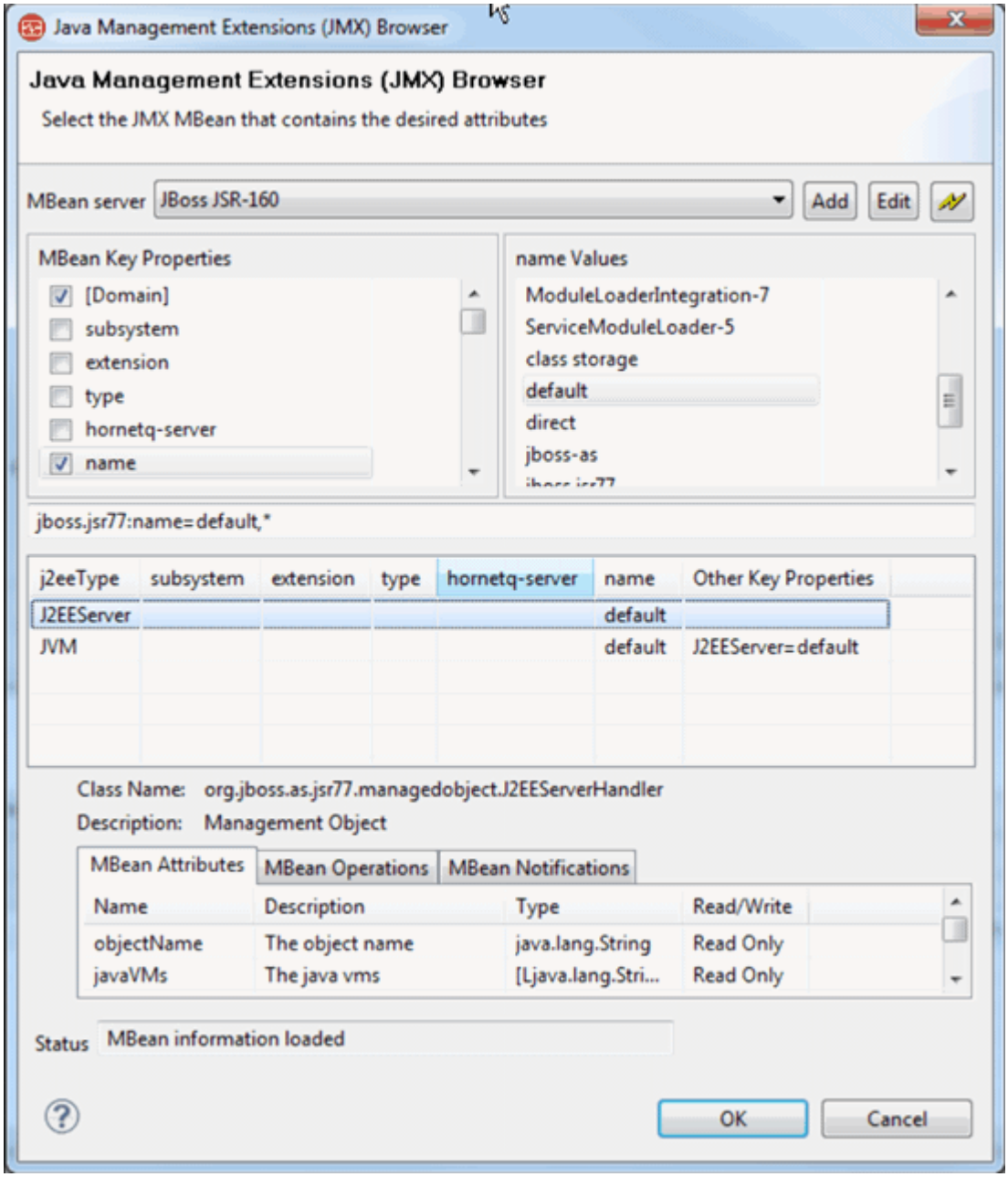


図 36. 「*Java Management Extensions (JMX)* ブラウザー」ウィンドウ

- 「**MBean キー・プロパティー**」領域: この領域は、サーバー上のすべての MBean から検出されたすべての固有のオブジェクト名キーのコレクションです。**[Domain]** は、実際にはキーではないため、特別な項目です。ただし、**[Domain]** 項目は、MBean ドメインの値に対する暗黙のキーとして処理されます。このリストから項目を選択すると、そのキー・プロパティーが含まれている MBean が検出されます。キー・プロパティーの値のリストは、「**選択したキー・プロパティー値**」リストに表示されます。キー・プロパティーにチェック・マークを付けると、そのキー・プロパティーがデータ・ソースのオブジェクト名パターンに組み込まれます。
- 「**選択したキー・プロパティー値**」領域: この領域には、すべての MBean から現在選択されている MBean キー・プロパティーの値が表示されます。これらの値のいずれかを選択すると、その MBean キー・プロパティーにチェック・マークが付きます。また、これらの値のいずれかを選択すると、メッセージ・フィールドに表示されているオブジェクト名パターンが、その MBean キー・プロパティーの名前と値で更新されます。
- オブジェクト名パターンに一致するすべての MBean の一覧表: 「MBean キー・プロパティー」リストと「選択したキー・プロパティー値」のリストからキー・プロパティーと値を選択すると、オブジ

ェクト名パターンが更新されます。また、選択したパターンに一致する MBean のリストに合わせて、この一覧表の MBean のリストも変更されます。どの MBean にも一致しないパターンの場合は、「MBean キー・プロパティー」リストの項目をクリアすることができます。項目をクリアするには、パターンで使用しているキーの横にあるチェック・ボックスをクリックしてチェック・マークを外します。対象の MBean を見つけるために、パターンを手動で編集することもできます。「*:*」というパターンを使用すると、すべての MBean が選択されます。

その表を使用すれば、サーバーから MBean を表示して、モニター対象にするデータが含まれている MBean を判別できます。多数の MBean が表示される可能性がある場合は、メニューを使用するか、いずれかの列見出しをクリックすることにより、任意のキー属性でソートすることができます。また、メニューから「**キー・プロパティーの表示**」を選択すると、任意の列の任意のキー属性を表示することができます。モニター対象にする MBean を識別するためのキー・プロパティー値を表の中で見つけたら、その値を右クリックして、メニューから「**キー・プロパティーを持つ MBean のみを選択**」を選択します。

- 選択した MBean の詳細を示した表: JMX ブラウザーに 1 つの MBean に関する情報が表示されます。MBean の詳細を表示するには、現在のフィルターに 合致する MBean の一覧表から対象の MBean を選択します。MBean のキー情報が、定義されている属性、操作、通知のリストとして表示されます。

JMX ブラウザーからデータ・ソースを作成するには、上記の 4 つのパネルを使用してオブジェクト名パターンを作成します。収集するモニター・データが含まれている MBean のセットに一致するようにオブジェクト名パターンを作成してください。例えば、すべての ThreadPool MBean のデータをモニターする場合は、以下の手順を実行します。

a)「**MBean キー・プロパティー**」パネルから「**type**」を選択します。「**type**」を選択すると、「**選択したキー・プロパティー値**」の値が更新され、あらゆる MBean の type キーのすべての固有値が表示されます。

b) type キーの値のリストから「**ThreadPool**」を選択します。「**ThreadPool**」を選択すると、「**MBean キー・プロパティー**」パネルの type キー・プロパティー名が選択され、オブジェクト名パターンが *:type=ThreadPool,* に更新されます。MBean のリストも更新され、そのパターンに合致する MBean だけが表示されるようになります。

c) MBean のリストからいずれかの MBean を選択して、その MBean で使用できる属性、操作、通知を表示します。モニターしたい数を超える MBean がリストに含まれている場合は、キー・プロパティーと値を選択する手順を続行する必要があります。 この手順は、モニターしたい MBean のセットを識別するオブジェクト名パターンが取得できるまで続行します。MBean のリストでメニューを開いて、表に表示されているキー・プロパティー値でオブジェクト・パターンを更新することもできます。

7. 正しいオブジェクト名パターンが得られたら、表から MBean を選択します。

選択した MBean のすべての属性が、新しい JMX データ・ソースの初期状態の属性になります。一部の属性にはデータが含まれていない可能性もあります。JMX データ・ソースを作成したら、属性を確認して、無関係の属性を削除します。選択した MBean に属性が含まれていない場合は、データ・ソースが属性なしで作成されるという趣旨の警告メッセージが表示されます。選択した MBean に通知が含まれている場合は、MBean から通知を受け取るためのイベント・データ・ソースも作成されます。

重要 : すべての MBean 属性について、Agent Builder は新規データ・セットに属性を作成します。 数値 MBean 属性については、Agent Builder は数値属性を作成します。すべてのオブジェクト・タイプ (ストリングを含む) について、Agent Builder は、値のストリング表記が含まれたストリング属性を作成します。MBean 属性からのオブジェクトが `javax.management.openmbean.CompositeData` タイプであり、Agent Builder ブラウザーでそのオブジェクト自体を読み取ることができる場合、複数の属性 (`CompositeData` オブジェクトに組み込まれたオブジェクトごとに 1 つ) が作成されます。`CompositeData` オブジェクト以外のオブジェクトの内部の値 (フィールドまたはメソッドの 戻り値) を含めるには、1254 ページの『Java Management Extensions (JMX) MBean に固有のフィールド』で説明されているように、より複雑なメトリック名を持つ属性を作成する必要があります。

8.「JMX 情報」ページで情報を入力したら、「**完了**」をクリックします。

前の手順で選択した MBean に基づいて、データ・ソースが作成されます。MBean を選択しなかった場合は、属性を持たない属性グループが作成されます。この場合は警告が表示され、MBean を選択する機会が与えられます。通知データ・ソースの場合は、属性を表示するデータ・ソースと区別するために、データ・ソース名の先頭に「**Event**」という語が付きます。

9. エージェントの他の JMX オプションを変更する場合は、「**グローバル JMX オプション**」をクリックします。これらのオプションでは、以下のことが可能です。

   a) そのエージェントで JMX モニターをサポートするかどうかを選択します。JMX モニターの属性グループとアクション実行コマンドを作成する場合は、「**JMX モニター属性グループおよびアクション実行を組み込みます**」を選択します。

   JMX モニターの説明については、次のセクションを参照してください。

   b) エージェントのデプロイ時に接続する MBean サーバーのタイプを選択します。

   いくつかのベンダー固有のサーバー・タイプと、標準仕様に基づくサーバーに対応した「JSR-160 準拠のサーバー」という汎用タイプが表示されます。必要な数だけ選択できますが、モニター対象の MBean をサポートするサーバー・タイプだけを選択してください。少なくとも 1 つのタイプを選択しなければなりません。複数のタイプを選択した場合は、エージェントの構成時に、接続先のサーバーのタイプを指定するためのプロンプトが表示されます。

10. 必要なオプションを選択したら、「**OK**」をクリックします。
11. オプション:「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1257 ページの『JMX 属性グループのテスト』を参照してください。
12. オプション:「**拡張**」をクリックして、この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
13. 「**次へ**」をクリックします。
14. 「**キー属性の選択**」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』を参照してください。
15. 「**次へ**」をクリックします。

    「**JMX エージェント全体のオプション**」ウィンドウに Agent Builder がサポートするアプリケーション・サーバーのタイプが表示されます。前に「**接続プロパティー**」ページで「**エージェント構成のデフォルトとして設定**」を選択した場合は、参照したアプリケーション・サーバーのタイプが自動的に選択されます。
16. 「**JMX エージェント全体のオプション**」ウィンドウ (1246 ページの図 37) で、エージェントからの接続を可能にする他のタイプのアプリケーション・サーバーを選択します。

    注 : 例で示している、「**JBoss Application Server JSR-160 接続**」の選択は、異なるデフォルト値が指定されていることを除き、「**JSR-160 準拠のサーバー**」を選択した場合と同じです。

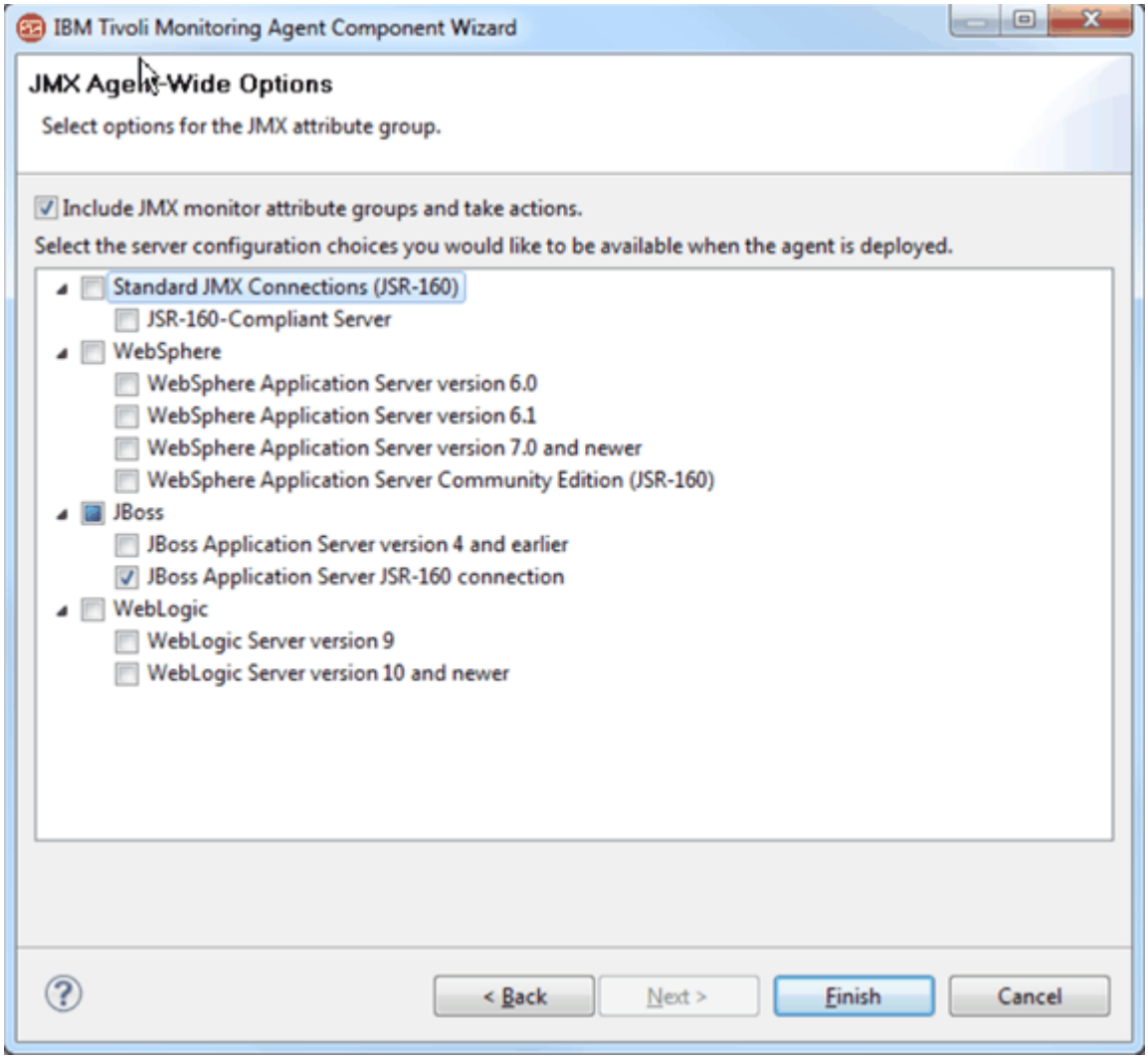


図 *37.* 「*JMX* エージェント全体のオプション」ウィンドウ

17. 以下の手順のいずれかを行います。
    - 新規エージェント・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。
18. エージェントの作成後に、接続可能なアプリケーション・サーバーのタイプを変更する場合は、「**JMX データ・ソース情報**」領域で「**グローバル JMX オプション**」をクリックします。
19. 「**JMX エージェント全体のオプション**」ウィンドウで、必要に応じて選択項目を変更します。
20. 「**OK**」をクリックします。
21. 自動的に生成された構成セクションとプロパティーを表示するには、Agent Editor の「**ランタイム構成**」タブをクリックします。

    「JBoss 基本パス」プロパティーのデフォルト値には、JMX ブラウザーに入力した値が設定されます。

次のタスク

JMX イベントの属性グループについて詳しくは、1449 ページの『JMX イベントの属性グループ』を参照してください。

JMX 構成

エージェント内に JMX データ・ソースを定義すると、一部の構成プロパティーが作成されます。

JMX ランタイム構成は、表示する構成の数を制御できる点に特徴があります。エージェントの JMX クライアントは、いくつかの異なるタイプのアプリケーション・サーバーに接続できます。ただし、どのエージ

ェントでも、それらのタイプのアプリケーション・サーバーをすべてサポートする必要はありません。サポートするアプリケーション・サーバーのタイプはユーザーが決定できるため、不要な構成セクションがエージェントに組み込まれることはありません。

ほとんどの場合、エージェントは 1 つの JMX アプリケーション・サーバー・タイプをモニターするように設計されています。JMX データ・ソースの作成時には、JMX ブラウザーを使用できます。JMX ブラウザーを使用する場合は、MBean サーバーの参照に使用される JMX サーバー構成オプションが、使用するエージェントに自動的に追加されます。エージェントの作成後に、接続可能なアプリケーション・サーバーのタイプを変更するには、「**JMX 情報**」領域で「**グローバル JMX オプション**」をクリックします。「**JMX エージェント全体のオプション**」ページで、必要に応じて選択項目を変更します。

複数のタイプの JMX アプリケーション・サーバーをモニターする汎用エージェントを 設計することができます。この場合は、「**JMX エージェント全体のオプション**」ページで、複数の JMX サーバー構成の選択項目を選択できます。複数のタイプの JMX 接続がサポートされている場合は、ランタイム構成により、そのエージェント・インスタンスで使用する接続のタイプを指定するためのプロンプトが表示されます。

注 : エージェントの 1 つのインスタンスは、1 つのタイプの JMX アプリケーション・サーバーにしか接続できません。サブノードを使用すると、1 つのエージェント・インスタンス内で同じタイプの別の JMX アプリケーション・サーバーに接続できます。複数のタイプの JMX アプリケーション・サーバーに接続するには、JMX アプリケーション・サーバー・タイプごとに少なくとも 1 つのエージェント・インスタンスを構成する必要があります。

Agent Editor を使用して、構成プロパティーの表示、追加、および変更を行うことができます。説明については、1363 ページの『Agent Editor を使用した構成プロパティーの変更』を参照してください。JMX データ・ソースがサブノードで 定義されている場合は、「サブノード 構成のオーバーライド」を指定することもできます。説明については、1350 ページの『サブノード 構成』を参照してください。

エージェント内に JMX データ・ソースを定義した場合、そのエージェントは Java を使用して JMX アプリケーション・サーバーに接続する必要があります。Java 構成プロパティーは自動的にエージェントに追加されます。

以下の Java 構成プロパティーは、エージェントのランタイム構成に固有のものです。

Java ホーム
Java のインストール・ディレクトリーを指す完全修飾パス。

特に WebLogic Server および WebSphere Application Server の場合は、モニター中のアプリケーションが使用する JVM と同じ JVM を使用するようにエージェントを構成します。

JVM 引数
Java 仮想マシンの引数のリストを指定します (任意指定)。

トレース・レベル
Java トレース・ファイルに書き込む情報の量を定義します。デフォルトでは、エラー・データだけがログ・ファイルに書き込まれます。

注 : Agent Builder は、JLog プラグインによって構成される独自の JVM とロギングを使用するため、これらのプロパティーを必要としません。

エージェント内に JMX データ・ソースを定義した場合は、以下に示す共通の必須構成フィールドがエージェントに自動的に追加されます。

接続
MBean サーバーとの接続のタイプ

ユーザー ID
MBean サーバーでの認証に使用するユーザー ID。

パスワード
ユーザー ID のパスワード。

基本パス
「**クラスパス**」フィールドで指定した JAR ファイルを検索するディレクトリー、または「**JAR ディレクトリー**」で指定したディレクトリー。これらのディレクトリーは、完全修飾ではありません。Windows では、ディレクトリー名を区切るためにセミコロン (;) を使用し、UNIX システムでは、セミコロン (;) またはコロン (:) を使用します。

クラスパス
エージェントによって検索する JAR ファイルを明示的に指定したパス。完全修飾でない場合は、JAR ファイルが見つかるまで、それぞれの基本パスに追加されます。

JAR ディレクトリー
JAR ファイルを検索するディレクトリー。Windows では、ディレクトリー名を区切るためにセミコロン (;) を使用し、UNIX システムでは、セミコロン (;) またはコロン (:) を使用します。これらのディレクトリーに格納されている JAR ファイルを明示的に指定する必要はありません。いずれかのディレクトリーに格納されていれば検出されます。それらのディレクトリーのサブディレクトリーは 検索対象になりません。完全修飾でないディレクトリー名は、ディレクトリーが見つかるまで、それぞれの基本パスに追加されます。

注 : リモート・モニターの場合、JAR ファイルとそのすべての従属 JAR ファイルは、エージェントを実行するコンピューター上にローカルにインストールする必要があります。これらの JAR ファイルは、モニター対象のアプリケーションに接続するために必要なファイルです。これらの JAR ファイルは、「**JAR ディレクトリー**」と、「**基本パス**」および「**クラスパス**」で構成する必要があります。さらに、モニターするアプリケーションに応じたサポートされる JVM をローカルにインストールし、「**Java ホーム**」構成フィールドでパスを指定してください。

例:

- WebLogic 10 の場合、クラスパスは `server/lib/wlclient.jar;server/lib/wljmxclient.jar` です。基本パスは、`server/lib` ディレクトリーがある WebLogic アプリケーション・サーバー・ディレクトリーを指します。
- WebSphere の場合、基本パスは WebSphere Application Server がインストールされているロケーションを指します。この例では、Windows 用と UNIX 用のデフォルトを提示するため、複数の基本パスがリストされています。クラスパスには、基本パスに対して相対的なパスの JAR ファイルがリストされています。「**Jar ディレクトリー**」フィールドの相対値 `lib` により、基本パスの下にあるこのディレクトリー内のすべての JAR ファイルがロードされます。
  - 「**基本パス**」: `C:¥Program Files¥IBM¥WebSphere¥AppServer;/opt/IBM/WebSphere/AppServer`
  - 「**クラスパス**」: `runtimes/com.ibm.ws.admin.client_6.1.0.jar;plugins/com.ibm.ws.security.crypto_6.1.0.jar`
  - **JAR ディレクトリー**: `lib`

「JMX エージェント全体のオプション」ページで選択される JMX サーバーのタイプに応じて、以下の構成プロパティーの一部またはすべてが追加されます。デフォルト値は Agent Builder によって提供され、変更可能です。

JSR-160 準拠のサーバー接続固有の構成プロパティー

JMX サービス URL
モニター操作のために接続する JMX サービス URL。

WebSphere Application Server (バージョン 6.0 以降) 接続に固有の以下の構成プロパティー。

ホスト名
モニター対象のアプリケーション・サーバーが存在するシステムのホスト名。ローカル・モニターの場合、この名前はローカル・システム名です。リモート・モニターの場合は、アプリケーション・サーバーが存在するシステムのホスト名です。

ポート
モニター対象のホスト名で使用するポート番号。

コネクター・プロトコル
モニター用の接続で使用するコネクター・プロトコル。RMI と SOAP がサポートされています。

プロファイル名
接続の構成に使用するプロファイルの名前。

JBoss Application Server (非 JSR-160) 接続に固有の以下の構成プロパティー。

JNDI 名
MBean サーバーの検索に使用する JNDI 名。

プロバイダー URL
モニター操作のために接続する JMX サービス・プロバイダー URL。

WebLogic Server 接続固有の構成プロパティー

サービス URL
モニター操作のために接続する JMX サービス・プロバイダー URL (JNDI 名が含まれます)。

注 : WebSphere 管理セキュリティーが有効になっている場合、該当するクライアントの接続プロパティー・ファイルで、クライアント・ログイン・プロンプトが無効になっていることを確認する必要があります。RMI 接続の場合、クライアントによるプロンプトがユーザーに表示されないようにするには、WebSphere Application Server のプロファイル・プロパティー・ディレクトリー内の `sas.client.props` ファイルの *com.ibm.CORBA.loginSource* プロパティーを変更する必要があります。SOAP 接続の場合、同じディレクトリーにある `soap.client.props` ファイルの *com.ibm.SOAP.loginSource property* を変更する必要があります。 いずれの場合も、*loginSource* プロパティーには、値が指定されないように設定する必要があります。

Agent Editor を使用して、構成プロパティーの表示、追加、および変更を行うことができます。1363 ページの『Agent Editor を使用した構成プロパティーの変更』 を参照してください。Windows データ・ソースがサブノードで定義されている場合は、「サブノード構成のオーバーライド」を指定することもできます。1350 ページの『サブノード構成』を参照してください。

JMX 通知
要求に応じてモニター・データを提供するほかに、通知を行う MBean もあります。

通知とは、イベント発生時に登録済みのリスナーに渡されるオブジェクトのことで、MBean によって生成されます。

Agent Builder で作成するエージェントでは、MBean ではなく通知から取得した値を含む属性グループを定義できます。

エージェントを開始すると、属性グループの MBean パターンと一致する各 MBean に、通知リスナーが登録されます。属性グループでは、受け取った通知ごとに 1 行の表示が行われます。列ごとに、通知に含まれるデータが 1 項目ずつ表示されます。通知で必要なデータは、MBean 用に列データを定義する場合と同様に、列の値によって定義されます。

イベントに基づかない属性グループについては、必要なときにデータが収集されます。 イベントに基づく属性グループについては、受け取ったイベントのうち、最新の 100 件のイベントのキャッシュがエージェントによって保守されます。Tivoli Enterprise Portal からの要求に応答するときには、それらのイベントが使用されます。それらのイベントは、シチュエーションによる分析とウェアハウス処理のためにすぐに転送されます。

JMX モニター
要求に応じてモニター・データを提供するほかに、モニターを行う MBean もあります。

JMX プロバイダーは、エージェントが持つ JMX モニター作成機能をサポートしています。 JMX モニターは、JMX エージェントが JMX サーバー上に作成する MBean です。 JMX モニターは、別の MBean の属性値をモニターし、その値が何らかの基準と一致した場合に通知を送信します。 モニターが特定の属性値について報告することを可能にするしきい値が定義されています。

JMX クライアントからのモニターの作成は、すべてのアプリケーション・サーバーでサポートされているわけではありませんが、現行リリースの WebSphere Application Server ではサポートされています。「**グローバル JMX オプション**」の下にある「**JMX モニター属性グループおよびアクション実行を組み込みます**」を選択すると、JMX のモニターとアクション実行コマンドをエージェントに組み込むことができます。

別の MBean の属性について報告する MBean は、どれもモニターと見なすことができます。 実際の JMX では、作成対象モニターのタイプである 3 つの具象モニター・クラスが定義されます。 作成される具象モニター・タイプは以下のとおりです。

- ストリング・モニター - ストリング属性を監視し、そのストリングが同じであるか異なっているかを報告します。
- ゲージ・モニター – 変数の数値属性を監視し、しきい値をまたいでの上下動について報告します。
- カウンター・モニター – 増加する数値属性を監視し、しきい値に達した場合や一定量の増加が見られた場合に報告します。

JMX モニターの通知を収集するため、または表示するために、以下の属性グループがエージェントに自動的に追加される場合があります。

- 登録済みモニター

  この属性グループは、ユーザーが追加したすべての JMX モニターを表示します。
- カウンター通知

  この属性グループは、カウンター・モニターから受け取ったすべての通知を報告します。
- ゲージ通知

  この属性グループは、ゲージ・モニターから受け取ったすべての通知を報告します。
- ストリング通知

  この属性グループは、ストリング・モニターから受け取ったすべての通知を報告します。

JMX モニターのアクション実行コマンド
モニターは、アクション実行コマンドを実行することによって作成されます。

各タイプのモニターを作成するためのアクション実行コマンドがタイプごとに 1 つずつ (計 3 つ) あるほか、既存のモニターを削除するための 4 つ目のアクション実行が定義されています。 256 文字の制限は、アクション実行コマンドに適用されます。

作成されるすべての JMX エージェント (Agent Builder が作成するすべてのエージェントを含む) の一部として、モニター属性グループがあります。すべてのエージェントに 4 つのアクション実行コマンドが用意されていますが、それらを利用できるのは JMX エージェントだけです。

JMX ストリング・メトリック追加モニター
ストリング属性を監視するためのモニターを作成するには、このアクション実行コマンドを使用します。

パラメーター

MBean パターン
このパターンに一致するすべての MBean が、このモニターによってモニターされます。

監視対象属性
監視対象となる MBean ストリング属性の名前。

一致の通知
モニター対象のストリングと基準値が一致した場合に通知を送信する場合は真、しない場合は偽です (デフォルトは偽)。

不一致の通知
モニター対象のストリングと基準値が一致しない場合に通知を送信する場合は真、しない場合は偽です (デフォルトは真)。

基準値
監視対象属性と比較するストリング。

デフォルトとは、引数が指定されていない場合のことです。

例えば、サービスが停止した場合に通知するように要求する場合は、以下のようになります

```
STRING_METRIC_WATCHER [*:type=Service,*] [StateString] [true] [false] [Stopped]
```

各項目の意味は次のとおりです。

:type=Service,
MBean パターン: Service という値を持つ type という名前のキー・プロパティーがあるすべての MBean をモニターします。

StateString
監視対象属性: type=Service のすべての MBean に共通のストリング属性。

true
一致の通知: StateString 属性が基準値である Stopped と一致した場合に、エージェントに通知が送信されます。

false
不一致の通知: Service 属性が Stopped と一致しない場合に、通知は送信されません。

Stopped
基準値: StateString 属性が値 Stopped に変わった場合に、通知が送信されます。

JMX ゲージ・メトリック追加モニター
ゲージ属性を監視するためのモニターを作成するには、このアクション実行コマンドを使用します。

パラメーター

MBean パターン
このパターンに一致するすべての MBean が、このモニターによってモニターされます。

監視対象属性
監視対象となる MBean ストリング属性の名前。

差分モード
モニターする値が、属性の実際の現行値と以前の値の差分である場合は true。モニターする値が、属性の実際の現行値である場合は false (デフォルトは false)。

上限の通知
モニターする値が上昇して上限しきい値を超えた場合に通知を送信する場合は真、しない場合は偽です (デフォルトは真)。

下限の通知
モニターする値が下降して下限しきい値を超えた場合に通知を送信する場合は真、しない場合は偽です (デフォルトは真)。

上限しきい値
監視対象属性がそれを下回っている必要がある値。

下限しきい値
監視対象属性がそれを上回っている必要がある値。

例: 空きメモリーが 10MB を下回ったときに通知を要求する

```
GAUGE_METRIC_WATCHER [ServerInfo] [FreeMemory] [false] [false] [true] [30000000] [10000000]
```

各項目の意味は次のとおりです。

***:type=ServerInfo**
MBean パターン: これは、ServerInfo という値を持つ type という名前の単一キー・プロパティーが名前に含まれているすべての MBean をモニターします。

FreeMemory
監視対象属性: アプリケーション・サーバーの空きメモリー量を示す、上下動する数値属性。

false
差分モード: ある監視操作と別の監視操作の間の差分ではなく、実際の属性値をモニターします。

false
上限の通知: 空きメモリーが上限値を上回った場合に通知を行いません。

true
下限の通知: 空きメモリーが少なくなりすぎた場合に通知を行いません。

30000000
上限しきい値: 上限しきい値の超過を通知しない場合でも、上限しきい値として妥当な値を設定する必要があります。一度下限しきい値を下回った属性値については、上限しきい値に達しない限り、2 回目の下限しきい値の通知は行われません。

10000000
下限しきい値: 通知の対象となる下限しきい値。

JMX カウンター・メトリック追加モニター
カウンター属性を監視するためのモニターを作成するには、このアクション実行コマンドを使用します。

パラメーター

MBean パターン
このパターンに一致するすべての MBean が、このモニターによってモニターされます。

監視対象属性
監視対象となる MBean ストリング属性の名前。

初期しきい値
監視対象属性の比較対象となる値。

オフセット
しきい値の超過が発生した際に、そのしきい値を変更するために加算する値。

モジュラス
カウンターの最大値。最大値に達した場合は 0 に戻ります。

差分モード
モニターする値が、属性の実際の現行値と以前の値の差分である場合は true。モニターする値が、属性の実際の現行値である場合は false (デフォルトは false)。このモードは、実質的に、変化率のモニターを有効にした場合と同じです。

期間細分度
測定の実行頻度 (デフォルトは 20 秒)。 差分モードを真にする場合の最も重要な値です

例: いずれかのサーバーで 3 件以上エラーが発生した場合に通知するよう要求する

```
COUNTER_METRIC_WATCHER [*:j2eeType=Servlet,*] [errorCount] [3] [4] [] [diff] [gran]
```

各項目の意味は次のとおりです。

`*:j2eeType=Servlet,*`
MBean パターン: これは、`ServerInfo` という値を持つ type という名前の単一キー・プロパティーが名前に含まれているすべての J2EE Servlet MBean をモニターします。

`errorCount`
監視対象属性: サーブレットのエラーの 件数を示す、増加する数値属性。

3
初期しきい値: `errorCount` が 3 以上になった場合に通知を行います。

4
オフセット: エラーが 3 件発生して通知を受け取った場合、以前のしきい値である 3 に 4 を加算して 7 という新しいしきい値が設定されます。2 回目の通知は、`errorCount` が 7 に達した時点で送信され、3 回目は 11、4 回目は 15 というように続きます。必ずゼロ以外の値を指定する必要があります。カウンターは常に増加していくものであり、オフセットによる増加がないカウンターは意味をなさないためです。

モジュラス:
errorCount に設計上の最大値はないため、実際には考えられない、余裕のある上限値を使用してください。

false
差分モード: モニターの対象となるのはエラーの絶対カウントです。errorCount の増加率を知る必要がある場合には、 差分モードを真にします。

期間細分度: 設定なし、つまりデフォルトの期間細分度である 20 秒が使用されます。期間細分度は、すべてのモニター・タイプで使用することができます。 ただし、表示されるのはカウンター・モニターの場合です。これは、(差分モードが true の場合に) 意味のある変化率を判断できるようにするためです。

JMX 削除メトリック・モニター
モニターを削除するには、このアクション実行を使用します。

パラメーター

番号
REGISTERED_MONITORS テーブルに示されているモニターの番号。

例えば、モニター番号 2 を削除する場合は、以下のようになります。

```
DELETE_WATCHER [2]
```

各項目の意味は次のとおりです。

2=
削除するモニターの番号。

JMX 操作
要求に応じてモニター・データを提供するほかに、操作を行う MBean もあります。

JMX データ・ソースを持つエージェントには、JMX_INVOKE アクション実行コマンドが組み込まれています。このコマンドを使用して、モニター対象のサーバーに対して JMX 操作を実行することができます。

アクション実行コマンド構文

アクションには、以下の構文があります。

```
JMX_INVOKE [MBean pattern] [Operation name] [Argument 1] [Argument 2]
[Argument 3] [Argument 4]
```

各項目の意味は次のとおりです。

MBean パターン
操作を実行する対象の MBean を選択する MBean 照会。 パターンが複数の MBean に一致する場合、一致した各 MBean に対して操作が実行されます。

Operation name
実行する MBean 操作の名前。

Argument 1、Argument 2、Argument 3、Argument 4
MBean 操作に指定できるオプションの引数。 引数は、ストリングや整数などの単純データ・タイプでなければなりません。

操作が正常に実行されると、JMX 呼び出しのアクション実行コマンドは success を返します。操作が値を返した場合は、その値が JMX データ・プロバイダーの ログ・ファイルに書き込まれます。

例: カウンターをリセットする操作を開始する

このアクションは、以下のように Threading MBean に対して resetPeakThreadCount 操作を実行します。

```
JMX_INVOKE [*:type=Threading,*] [resetPeakThreadCount][] [] [] []
```

各項目の意味は次のとおりです。

:type=Threading,
MBean パターン: このパターンは、タイプが Threading であるすべての MBean に一致します。

resetPeakThreadCount
操作名: パターンに一致する各 MBean に対して実行する操作。

[] [] [] []
引数 1、2、3、4: この操作に引数は不要です。 これらは、アクションの構文に準拠させる目的で指定するに過ぎません。

例: 引数を指定してアクションを開始する

このアクションは、Threading MBean に対して getThreadCpuTime 操作を 実行します。結果は JMX データ・プロバイダーのトレース・ファイルに記録されます。

```
JMX_INVOKE [*:type=Threading,*] [getThreadCpuTime] [1] [] [] []
```

各項目の意味は次のとおりです。

:type=Threading,
MBean パターン: このパターンは、タイプが Threading であるすべての MBean に一致します。

getThreadCpuTime
操作名: パターンに一致する各 MBean に対して実行する操作。

1
引数 1: 照会するスレッド ID。

[] [] []
引数 2、3、4: この操作にこれらの引数は不要です。 これらは、アクション実行コマンドの構文に準拠させるための空の引数として指定します。

JMX_INVOKE アクション実行コマンドの実行

ユーザーが JMX_INVOKE アクション実行コマンドを実行するとエージェント開発者が想定することはできません。 代わりに、JMX_INVOKE アクション実行を実行するその他のアクションを開発する必要があります。これらのアクションでは、可能であれば、操作名および MBean パターンなどの詳細をユーザーから隠蔽してください。

JMX モニターの開始および停止

JMX モニターは、エージェントおよび JMX サーバーが開始してから停止するまでの間、継続的に実行されます。

JMX サーバーのリサイクルを 検出すると、エージェントはモニターを再登録します。エージェントがリサイクルされると、モニターは再登録されます。モニター定義は、default_*instanceName*.monitors という名前のファイルに格納されます (*instanceName* は、エージェントのインスタンス名であり、単一インスタンス・エージェントの場合は default になります)。このファイルは、以下のディレクトリーに格納されています (*xx* は 2 文字の製品コードです)。

- Windows システム: TMAITM6¥k*xx*¥config
- UNIX および Linux システム: *architecture*/*xx*/config (architecture の値を確認するための情報については、1393 ページの『システム 上の新規ファイル』を参照してください)

エージェントを再始動すると、エージェントはモニター定義ファイルを使用してモニターを復元します。

Java Management Extensions (JMX) MBean に固有のフィールド

JMX 属性グループのメトリック名の構文は、「**属性情報**」ウィンドウで指定するときのルールに従う必要があります。

JMX 属性グループのメトリック名の構文では、各トークンをピリオドで区切ります。各トークンによって 1 次値を形成し、オプションとして 2 次値を形成することもできます。

- **1 次値**: 表の特定の行にある MBean または通知から直接取得する値。MBean から取得する 1 次値は、MBean のいずれかの属性または MBean 操作の呼び出し (メソッド呼び出し) によって取得します。通知から取得する 1 次値は、通知オブジェクトのいずれかのフィールドまたはメソッド 呼び出しによって取得します。1 次値として使用できるのは、プリミティブ型または Java オブジェクトです。
- **2 次値**: 1 次値の追加処理または他の 2 次値から取得する値。2 次値はエンジンの内部で処理され、JMX サーバーへの呼び出しを伴いません。1 次値 (または他の 2 次値) が Java オブジェクトの場合は、そのオブジェクトのパブリック・フィールドの取り出しの結果が 2 次値になります。そのオブジェクトに対す

るメソッド呼び出しの結果が 2 次値になる場合もあります。そのような 2 次値を取得するときには、1 次 (または他の 2 次) Java オブジェクトの Java イントロスペクションを使用します。1 次値 (または他の 2 次値) が MBean 名の形式の Java ストリングの場合は、2 次値がドメインになることがあります。その MBean 名を構成する他のいずれかのプロパティーが 2 次値になる場合もあります。

メトリック名フィールドの構文形式を以下に示します。

```
Metric Name    =    PrimaryValue [ .SecondaryValue ]
PrimaryValue    =    Attribute.attributeName |
        Method.methodName |
        Domain |
        Property.propertyName |
        Field.fieldName |
        Name
SecondaryValue    =    Field.fieldName |
        Method.methodName |
        Domain |
        Property.propertyName |
        Explode |
        ElementCount
```

```
propertyName    =    the name of a key property in an MBean ObjectName
attributeName    =    the name of an MBean attribute
methodName    =    a zero-argument operation of an MBean or a zero-argument method
of a Notification or other Java object.
methodName(argument) = A single-argument operation of an MBean or a
single-argument method of a Notification or other Java object. The
argument will be passed to the method as a string.
fieldName    =    the name of a public instance variable in a Notification or
other Java object
notificationMethod    =    the name of a public zero-argument method of a
Notification object
```

メトリック名定義に 1 次値だけを組み込んだ場合は、以下の項目のいずれかのデータが収集されます。

- MBean ドメイン
- MBean のストリング値
- MBean 名からのキー・プロパティー
- MBean 属性の数値またはストリングの属性値 (別の MBean の絶対パス名を含む)。MBean 操作からの、数値またはストリングの戻り値。
- 通知オブジェクトの数値またはストリングのパブリック・インスタンス変数の値
- 通知操作からの、数値またはストリングの戻り値。

2 次値をメトリック定義に追加することにより、Java オブジェクトの 1 次値にドリルダウンすることができます。public メソッドを開始することも、public インスタンス変数を取り出すこともできます。

2 次値をメトリック定義の別の 2 次値に追加すると、2 次値のオブジェクトにドリルダウンすることができます。MBean 内または通知内でオブジェクトがネストされている深さまでドリルダウンすることができます。

1 次値と 2 次値を構成するトークンは、キーワードまたは名前です。ほとんどの場合は、キーワード・トークンの後に名前トークンが続きます。いくつかの例を以下の表にまとめます。

メトリック名の例	属性グループ・タイプ	返されるデータの説明
Domain	MBean	MBean のドメイン部分 (コロンの前の部分)。
Name	MBean	MBean の完全ストリング表記。
Attribute.serverVendor	MBean	MBean 属性 serverVendor。
Method.getHeapSize	MBean	MBean の getHeapSize() から返される値。
Property.j2eeType	MBean	j2eeType の値は MBean 名から抽出されます。

メトリック名の例	属性グループ・タイプ	返されるデータの説明
Field.Message	イベント (通知)	通知の**メッセージ**・フィールド。

キーワードとして、Attribute、Method、Field のいずれかを指定すると、他のデータが含まれている Java オブジェクトを抽出できます。2 次値の定義を追加して、それらのオブジェクトに対して操作を実行することもできます。さらに例を挙げましょう。

メトリック名の例	属性グループ・タイプ	返されるデータの説明
Attribute.deployedObject.Method.getName	MBean	MBean から deployedObject 属性を取り込んで、getName() メソッドの結果を取得します。
Attribute.eventProvider.Method.getException.Method.getDescription	MBean	この場合は 3 段階になります。最初に、eventProvider という名前の属性が、getException() メソッドを持つオブジェクトであると想定されます。そのメソッドから、getDescription() メソッドを持つオブジェクトが返されます。さらにそのメソッドが呼び出され、戻り値が列に挿入されます。
Attribute.HeapMemoryUsage.Method.get(used)	MBean	MBean から HeapMemoryUsage 属性を取り込んで、get(String value) メソッドの結果を取得します。 使用されるストリングが引数としてメソッドに渡されます。指定可能な引数は 1 つだけであり、リテラル・ストリング値でなければなりません。 開いている MBean 複合データ構造からデータを収集する方法を示します。

前の値から MBean 名形式のストリングが返された場合は、2 次値のキーワードとして Domain と Property を使用することができます。例えば、次のようにします。

メトリック名の例	属性グループ・タイプ	返されるデータの説明
Attribute.jdbcDriver.Property.name	MBean	属性 jdbcDriver から MBean 名が返され、その MBean 名からキー・プロパティー name が抽出されます。
Attribute.jdbcDriver.Domain	MBean	属性 jdbcDriver から MBean 名が返され、その MBean 名からドメインが抽出されます。

ElementCount キーワードおよび Explode キーワードを使用して、データの配列またはコレクションに対して操作を実行します。

- ElementCount – 配列に含まれているエレメントの数を返します。
- Explode - 1 つの行をいくつかの行に分解し、配列に含まれているエレメントごとに 1 つの新しい行を作成します。

各キーワードの例を以下に示します。

メトリック名の例	属性グループ・タイプ	返されるデータの説明
Attribute.deployedObjects.ElementCount	MBean	MBean 属性の deployedObjects は配列であり、その配列内のエレメントの数がこの列に入ります。
Attribute.deployedObjects.Explode.MBean.Property.j2eeType	MBean	この表では、デプロイ済みのオブジェクト内のエレメントごとに 1 つの行が作成されます。この列には、デプロイ済みオブジェクトの j2eeType が含まれます。
Attribute.SystemProperties.Method.values.Explode.Method.get(key)	MBean	開いている MBean 表データ構造内の項目ごとに 1 つの新規行が取得されます。各表データ構造には、「key」という名前の項目がある複合データ構造が含まれており、この項目が返されます。

JMX 属性グループのテスト

Agent Builder 内で作成した JMX 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**JMX 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**JMX テスト**」ウィンドウが表示されます。
2. 「**接続名**」の下にあるリストから接続を選択します。あるいは、接続を追加するために「**追加**」をクリックし、1239 ページの『Java Management Extensions (JMX) MBean のモニター』で説明する手順を実行します。
3. オプション: テストを開始する前に、環境変数、構成プロパティー、および Java 情報を設定できます。

   詳しくは、1376 ページの『属性グループのテスト』を参照してください。JMX 構成について詳しくは、1246 ページの『JMX 構成』を参照してください。
4. 「**エージェントの開始**」をクリックします。

   ウィンドウに、エージェントが開始されていることが表示されます。
5. 「**データの収集**」をクリックして、Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートします。

   エージェントは、JMX サーバーでデータをモニターします。「**JMX テスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
6. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
7. 「**エージェントの停止**」をクリックしてエージェントを停止します。
8. 「**OK**」または「**キャンセル**」をクリックして「**JMX テスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』

Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Common Information Model (CIM) のデータのモニター

データ・ソースを定義して、Common Information Model (CIM) データ・ソースからデータを受信することができます。データ・ソースは、単一の CIM クラスをモニターし、このクラスからのすべての値を、データ・ソースが生成するデータ・セット内に配置します。クラスが複数のインスタンスを提供する場合、データ・セットには複数行が含まれます。データ・セットの行が 1 つのみになるように、インスタンス名によってフィルタリングできます。

このタスクについて

このタスクでは、Common Information Model (CIM) データ・ソースの構成手順について説明します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」域で「**CIM**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**Common Information Model (CIM) 情報**」ページの「**CIM 管理対象システム情報**」領域で、以下からいずれか 1 つを選択します。
   - 収集するデータに対応する「**ネーム・スペース**」および「**CIM 管理対象システム・クラス名**」フィールドに入力します。
   - 「**参照**」をクリックして、特定のシステム上の CIM リポジトリーを参照します。

     「**Common Information Model (CIM) クラス・ブラウザー**」ウィンドウが表示されます。このブラウザーは CIM サーバーに接続して、そのサーバー上に存在するクラスに関する情報を提供します。

     リモート・システムを参照するには、「**ホスト名**」リストからシステムを選択します (ドロップダウン・リストが定義済みである場合)。あるいは、「**追加**」をクリックして、CIM サーバーがあるシステムのホスト名を追加します。

     ホスト名を指定するための構文は `http[s]://hostname:port` です。ホスト名のみを指定した場合、Common Information Model (CIM) クラス・ブラウザーはデフォルト URL (`http://hostname:5988`) を使用して接続します。

     ポートを指定せずにプロトコルを指定すると、`http` の場合は デフォルトとして 5988 が使用され、`https` の場合は デフォルトとして 5989 が使用されます。

     プロトコルを指定せずにポートを指定した場合は、 指定したポートで `http` が使用されます。

     ネーム・スペース内の、参照するオブジェクトに対する読み取り権限を持つアカウントのユーザー ID とパスワードを入力します。ウィンドウが更新され、リモート・システムに関する情報が表示されます。

     Agent Builder は、CIM サーバーで使用可能なネーム・スペースのディスカバーを試行します。 ディスカバーされたネーム・スペースは 「**ネーム・スペース**」リストに表示されます。 しかし、サーバーで使用可能なネーム・スペースを Agent Builder が すべてディスカバーできない場合があります。「**ネーム・スペース**」リストにリストされていない ネーム・スペースを参照する場合は、「**ネーム・スペース**」リストの横にある 正符号 (+) アイコンをクリックします。フィールドにネーム・スペースの 名前を入力して「**OK**」をクリックしてください。 ネーム・スペースが CIM サーバーに存在する場合は、そのネーム・スペースで 定義されているクラスがリストされます。入力したネーム・スペースは保存され、 次にその CIM サーバーを参照したときに「**ネーム・スペース**」リストに 表示されます。

     「**ネーム・スペース**」リストからネーム・スペースを選択すると、Agent Builder がその特定のネーム・スペースに関するすべてのクラス情報を収集します。 次に Agent Builder は、ネーム・スペースを素早く切り替えられるよう、この情報をキャッシュに入れます。特定のネーム・スペースに関するクラス情報を Agent Builder に強制的に再収集させる場合は、そのネーム・スペースを選択し

て「**接続**」をクリックします。「**接続**」をクリックすると、キャッシュに入っている情報がすべて削除され、Agent Builder がクラス情報を再収集するようになります。

「**検索**」(双眼鏡) アイコンを クリックすると、選択したい項目をリストから見つけることが できます。「**検索語句**」フィールドに語句を入力し、「**名前で検索**」または「**クラス・プロパティーで検索**」のいずれかのフィールドをクリックして 希望の検索方法を指定し、「**OK**」をクリックします。検索対象の項目が見つかったら、それを選択して「**OK**」をクリックしてください。

5. 「Common Information Model (CIM) 情報」ページの「**オペレーティング・システム**」域で、収集が行われるオペレーティング・システムを選択します。
6. 「**CIM 管理対象システム情報**」領域にネーム・スペースと CIM クラス名を入力した場合は、以下のステップを実行してください。
   a) 「**次へ**」をクリックして「**属性情報**」ページを表示し、属性グループ内に最初の属性を定義します。
   b) 「**属性情報**」ページに関する情報を指定し、「**完了**」をクリックします。
7. CIM 情報を参照する場合は、「キー属性の選択」ページが表示されます。「キー属性の選択」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』 を参照してください。
8. CIM 情報を参照していた場合は、「**完了**」をクリックします。
9. オプション: 「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1260 ページの『CIM 属性グループのテスト』を参照してください
10. オプション: 「**拡張**」をクリックして、 この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
11. 以下の手順のいずれかを行います。
    a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    b) データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

CIM 構成
CIM 構成プロパティーに関する詳細。

エージェント内に CIM データ・ソースを定義すると、CIM 構成プロパティーが自動的にそのエージェントに追加されます。Agent Editor を使用して、構成プロパティーの表示、追加、および変更を行うことができます。手順については、1363 ページの『Agent Editor を使用した構成プロパティーの変更』を参照してください。 サブノードで CIM データ・ソースが定義されている場合は、サブノード構成のオーバーライドを指定してください。 説明については、1350 ページの『サブノード構成』を参照してください。

以下の接続固有の構成プロパティーは、「CIM 構成」ページにあります。

CIM ローカルまたはリモート
CIM サーバーに対するローカルまたはリモート認証。ローカル/リモートのデフォルト値はリモートです。

CIM ユーザー ID
CIM サーバーへのアクセスに使用されるユーザー ID

CIM パスワード
CIM サーバーにアクセスするためのパスワード

CIM ホスト名
CIM データ用にアクセスされるホスト名

SSL 経由の CIM
CIM サーバーとの通信に SSL を使用。オプションは「はい」および「いいえ」です。デフォルト値は「いいえ」です。

CIM ポート番号
保護されない通信に使用されるポート番号。

CIM SSL ポート番号
保護された通信に使用されるポート番号。デフォルトは 5989 です。(Solaris 8 のデフォルト値は通常はこれと異なります。)

CIM 属性グループのテスト

Agent Builder 内で作成した CIM 属性グループをテストできます。

手順

1. テスト・プロシージャーは以下の方法で始動します。
   - エージェントの作成中に、「**CIM 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**設定のテスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数と構成プロパティーを設定します。
   詳しくは、1376 ページの『属性グループのテスト』を参照してください。
3. **ホスト名**を選択または追加します。
   ホスト名の追加について詳しくは、1258 ページの『Common Information Model (CIM) のデータのモニター』を参照してください。
4. 「**エージェントの開始**」をクリックします。
   ウィンドウが開いて、エージェントが開始されていることが表示されます。
5. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。
   エージェントは、CIM サーバーでデータを照会します。「**テスト設定**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
6. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。
   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
7. 「**エージェントの停止**」をクリックしてエージェントを停止します。
8. 「**OK**」または「**キャンセル**」をクリックして「**設定のテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

ログ・ファイルのモニター

データ・ソースを定義して、テキスト・ログ・ファイルからデータを受信することができます。エージェントはログ・ファイルに追加された行を定期的に解析し、これらの行に基づいてイベント情報を生成します。エージェントがログをイベントとして解析する方法を構成できます。データをフィルタリングおよび要約するようにエージェントを構成することもできます。結果のイベントは、データ・セット内に配置されます。

始める前に

注 : エージェントがモニター対象とするのは、エージェントが実行されているのと同じロケールおよびコード・ページのログ・ファイルです。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**ログに記録されたデータ**」をクリックします。
2. 「**データ・ソース**」領域で「**ログ・ファイル**」をクリックします。

3.「**次へ**」をクリックします。
4.「**ログ・ファイル情報**」ページの「**ログ・ファイル情報**」域で、モニターするログ・ファイル名を入力します。

ファイル名は完全修飾名である必要があります。

a) オプション: ログ・ファイル名の一部を、ランタイム構成プロパティーから取得するようにすることができます。 ログ・ファイル名を作成するには、「**構成プロパティーの挿入**」をクリックして構成プロパティーを選択します。

b) オプション: また、ファイルに動的ファイル名を使用することもできます。詳しくは、1498 ページの『動的ファイル名のサポート』 を参照してください。

5.「**フィールド ID**」領域で、以下のオプションのいずれか 1 つをクリックします。

一定の文字数
これを選択すると、文字数を制限します。

このオプションでは、ログ・ファイルから保持可能な最大数の文字が各属性に割り当てられます。例えば、3 つの属性 A、B、C がこの順序で指定されていて、各属性が最大長 20 のストリングの場合、ログ・レコードの最初の 20 バイトが A に入り、次の 20 バイトが B に入り、その次の 20 バイトが C に入ります。

タブ分離文字
これを選択すると、タブ分離文字を使用できます。

スペース分離文字
これを選択すると、同時に出現する複数のスペースを単一の分離文字として使用できます。

分離文字テキスト
これを選択した場合は、分離文字テキストを入力します。

開始テキストと終了テキスト
これを選択した場合は、開始テキストと終了テキストの両方を入力します。

エレメントの XML
選択した場合は、レコードとして使用する XML エレメントの名前を入力するか、「**参照**」をクリックしてエレメントを定義します。

「**参照**」をクリックした場合は、「**XML ブラウザー**」ウィンドウが 表示されます。参照機能を使用すると、Agent Builder は、子タグとその属性を検出して、レコードの可能な属性をすべて識別します。

注 :「**拡張**」をクリックして表示されたウィンドウで情報を入力する場合を除き、入力する情報に関しては、以下の条件が想定されています。

- 一度に 1 つだけのログ・ファイルがモニターされる。
- 定義される属性を入力するのに必要なすべてのフィールドが、ログ・ファイルの各行に含まれている。

ログ・ファイルの構文解析および分離文字について詳しくは、1269 ページの『ログ・ファイル構文解析および分離文字』 を参照してください。

6. オプション:「**ログ・ファイル情報**」で「**拡張**」をクリックし、「**詳細なデータ・ソース・プロパティー**」ページを使用して以下の操作を実行します。

- 複数のファイルをモニターするか、複数の異なったオペレーティング・システム上にある複数の異なった名前のファイルをモニターする。あるいは正規表現に一致する名前のファイルをモニターする。
- ログ・ファイルの複数の行から、一連のフィールドを取得する。
-「**イベント・フィルタリングおよび要約オプション**」を選択する。
- 出力要約情報を生成する。この要約により、各インターバルで追加の属性グループが作成されます。この属性グループについて詳しくは、1433 ページの『ログ・ファイルの要約』 を参照してください。この機能は、「イベント情報」タブで使用可能なオプションにより、非推奨となっています。

a) 複数のログ・ファイルをモニターするには、「**追加**」をクリックして、名前を入力します。

複数のファイルをリストする場合は、それぞれのファイルに固有のラベルを入力しなければなりません。このラベルは、どのファイルがレコードを生成したかを示す属性として表示することができます。このラベルにスペースを入れることはできません。

b) オプション: 各ログ・ファイルをモニターするオペレーティング・システムを選択するには、以下の手順を実行します。

1) ログ・ファイルの「**オペレーティング・システム**」列をクリックします。

2)「**編集**」をクリックします。

3)「**オペレーティング・システム**」ウィンドウで、オペレーティング・システムを選択します。

4)「**OK**」をクリックして変更内容を保存し、「**詳細なデータ・ソース・プロパティー**」ページに戻ります。

c) オプション: 指定するファイル名が正規表現であり、この正規表現がファイル名ではなくファイルの検索に使用される場合は、「**ファイル名が正規表現に一致する**」を選択します。

詳しくは、1489 ページの『ICU 正規表現』を参照してください。このボックスを選択しない場合、実際のファイル名を指定する必要があります。あるいは、ファイル名パターンのルールに従ったパターンを指定する必要があります (これについては、1499 ページの『動的ファイル名の構文』を参照してください)。

d) オプション: ファイル名パスの 1 つのサブディレクトリーを正規表現と突き合わせるには、「**1 つのディレクトリー・エレメントが正規表現に一致する**」を選択します。

このオプションを選択できるのは、直前のステップで「**ファイル名が正規表現に一致する**」も選択してある場合のみです。

パス名に正規表現メタ文字を使用している場合、そのメタ文字はパスの 1 つのサブディレクトリーでのみ使用できます。例えば、`/var/log/[0-9¥.]*/mylog.*` と指定すると、1 つのサブディレクトリーでメタ文字を使用できます。`[0-9¥.]*` を指定すると、`/var/log` 内の、番号とドット (`.`) のみで構成されるサブディレクトリーが突き合わされます。`mylog.*` を指定すると、これらの `/var/log` サブディレクトリー内の、`mylog` で始まり、ゼロ個以上の文字が続くファイル名が突き合わされます。

一部のオペレーティング・システムでは円記号 (¥) がディレクトリー分離文字として使用されるため、正規表現のエスケープ・メタ文字と混同される可能性があります。この混同を避けるため、ディレクトリーを表すには常にスラッシュを使用する必要があります。例えば、Windows ファイルを `C:¥temp¥mylog.*` と指定した場合、`¥t` がタブ文字の短縮表記と解釈される可能性があります。したがって、すべてのオペレーティング・システムで常に、ディレクトリー分離文字としてスラッシュ (`/`) を使用してください。`C:/temp/mylog.*` という例は、`C:/temp` ディレクトリー内の、`mylog` で始まるすべてのファイルを表します。

e)「**複数のファイルが一致する場合は次のファイルを選択**」リストで、 以下のオプションのいずれかを選択します。

- **ファイル名の数値が最も大きいファイル**
- **サイズが最大のファイル**
- **更新日時が最新のファイル**
- **作成日時が最新のファイル**
- **一致するすべてのファイル**

注 :「**一致するすべてのファイル**」を選択すると、 エージェントは、動的ファイル名パターンに一致する ディレクトリー内のすべてのファイルを識別します。エージェントは、すべてのファイルに対する更新を並行してモニターします。データ収集処理中には、すべてのファイルのデータが混合されます。最善の方法は、「**フィールド情報の記録**」で「**ログ・ファイル名**」を選択して属性を追加し、ログ・メッセージを、そのログ・メッセージを含むログ・ファイルに相関させる方法です。必ず、動的ファイル名パターンに一致するすべてのファイルを一定の方法で 属性に分割できるようにしてください。選択されたログ・ファイルを一貫した方法で解析できない場合の最善の方法は、「**フィールド情報の記録**」で「**レコード全体**」を選択し、単一の属性を定義する方法です。 属性に対する「**フィールド情報の記録**」の指定について詳しくは、ステップ 1264 ページの『8』 を参照してください。

f) ファイルの処理方法を選択します。

「**ファイルがサンプリングされたときにすべてのレコードを処理**」を選択すると、 ログ・モニターに定義されたサンプリング間隔が終了するたびに、ファイル全体にあるすべてのレコードを処理できます。デフォルトの間隔は 60 秒です。この間隔は、*KUMP_DP_COPY_MODE_SAMPLE_INTERVAL* 環境変数 (値は秒単位で指定します) を使用して変更できます。ファイルから削除されない限り、毎回、同じレコードが報告されます。 これを選択した場合、ファイルへの新規レコードの書き込み時にイベント・データは生成されません。「**ファイルに付加された新規レコードの処理**」を選択すると、エージェント実行時に、ファイルに付加された新規レコードを処理できます。 ファイルに追加されたレコードごとに、イベント・レコードが生成されます。 ファイルが置き換えられると (なんらかの方法で最初のレコードが変更されると)、そのファイルが処理され、ファイル内のレコードごとにイベントが生成されます。

注 : XML ログ・ファイルにレコードを追加する場合、**フィールド ID** として選択した XML エレメント内で定義されているすべてのエレメント・セットが、その追加レコードに含まれている必要があります。

g) ファイルに付加された新規レコードを処理するようにした場合、新規レコードの検出方法も選択できます。

「**レコード・カウントが増加したときに新規レコードを検出**」を選択すると、ファイル・サイズが変更されたかどうかに関係なく、ファイル内のレコード数が増加したときに新規レコードを検出することができます。この機能は、ログ・ファイルにレコードが書き込まれる前にログ・ファイル全体を事前割り振りする場合に便利です。事前割り振りされないファイルについてこのオプションを選択することもできますが、ファイルのサイズをモニターするよりも 効率が下がります。「**ファイル・サイズが増加したときに新規レコードを検出**」を選択すると、ファイルにいつ新しい項目が追加されたかを、標準的な方法で判別することができます。ただし、モニター対象ファイルの置換を認識する際に、若干の遅延が発生することがあります。

h) 「**ファイル・サイズが増加したときに新規レコードを検出**」を選択した場合、モニター・エージェントの開始時に存在しているファイルの処理方法を選択することもできます。

「**既存のレコードを無視**」を選択すると、エージェントの開始時にファイル内のレコードに関するイベント生成が無効になります。「**ファイルの既存レコード ___ の処理**」を選択すると、エージェントの開始時に、ファイルの最後から一定数のレコードに関してイベントの生成を指定します。「**以前にエージェントによって処理されていないレコードを処理**」: 再始動データがモニター・エージェントによって保守され、前回の実行時にどのレコードを処理したかをエージェントが認識できるように指定されます。前回エージェントを実行して以来ファイルに付加されたすべてのレコードについてイベントが生成されます。 このオプションを選択すると、ファイルにレコードが追加されるたびに、若干の余分な処理が発生します。

i) 「**以前にエージェントによって処理されていないレコードを処理**」を選択した場合、エージェントの開始時に既存のファイルが置き換えられていることが検出された場合の動作を選択することができます。

「**ファイルが置き換えられている場合にすべてのレコードを処理**」: モニター対象ファイルに関する情報と再始動データ情報が一致しない場合、ファイル内のすべてのレコードについてイベントが処理されます。不一致の例としては、ファイル名が異なる場合、ファイル作成時刻が異なる場合、ファイル・サイズが減少した場合、ファイルの最終変更時刻が以前のものよりも早い場合などがあります。「**ファイルが置き換えられている場合にレコードを処理しない**」: モニター対象ファイルに関する情報と再始動データ情報が一致しない場合、ファイル内の既存のレコードの処理を無効にします。

j) ログ・ファイル内の複数の行を単一の論理レコードとして解釈するには、「**レコード ID**」タブをクリックします。

注 : 「**ログ・ファイル情報**」ページで、フィールド ID として「**エレメントの XML**」を選択した場合、「**レコード ID**」タブは表示されません。

- 「**単一行**」は、各行を単一の論理レコードと 解釈します。
- 「**分離行**」では、あるレコードを別のレコードから分離する行を識別する文字列を入力できます。

  注 : 分離行は、前のレコードおよび次のレコードの一部ではありません。
- 「**ルール**」は、レコードを構成する行の最大数を指定し、オプションで、レコードの始まりまたは終わりを示す文字列を指定します。「**ルール**」では、 以下のプロパティーを指定できます。

– 「**非ブランク行の最大数**」は、規則で処理できる非ブランク行の最大数を定義します。
– ***ルールのタイプ***: 以下のいずれかを指定できます。
  - **テキスト比較がありません** (レコードごとの最大行数は、1 つの論理レコードを示します)。
  - **レコードの先頭の指定** (1 つの論理レコードの先頭にマークを付けます)。
  - **レコードの終わりの指定** (1 つの論理レコードの終わりにマークを付けます)。
– 「**オフセット**」は、比較ストリングが出現すべき 行内の位置を指定します。
– 「**比較テスト**」は「**等しい**」(特定のオフセットで 文字列が一致する必要がある) または「**等しくない**」(特定のオフセットに 特定の文字列が出現しない) のいずれかにすることができます。
– 「**比較ストリング**」は、比較する文字列を 定義します。

• 「**正規表現**」は、レコードの先頭または終わりを示すのに使用するパターンを指定します。「**正規表現**」を使用することにより、 以下のプロパティーを指定できます。
  – 「**比較ストリング**」は、突き合わせる文字列を 定義します。

    または

  – レコードの先頭または終わり
    - 「**レコードの先頭の指定**」は、単一の論理レコードの先頭に マークを付けます。
    - 「**レコードの終わりの指定**」は、単一の論理レコードの終わりに マークを付けます。

k) 前のステップで「**ファイルがサンプリングされたときにすべてのレコードを処理**」を選択した場合は、「**フィルター式**」タブをクリックします。「**フィルター式**」をクリックすることにより、行として返されたデータを、1 つ以上の属性または構成変数 (あるいはその両方) の値に基づいてフィルタリングすることができます。

前のステップで「**ファイルに付加された新規レコードの処理**」を選択した場合は、フィルター式を作成することはできません。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。

l) 前のステップで「**ファイルに付加された新規レコードの処理**」を選択した場合は、「**イベント情報**」タブをクリックして、「**イベント・フィルタリングおよび 要約オプション**」を選択します。

詳しくは、1405 ページの『イベント・フィルタリングおよび 要約』 を参照してください。

注 : 以前のバージョンの Agent Builder を使用してエージェントを作成した場合は、「要約」タブが表示される場合があります。「要約」タブは現在、「イベント情報」タブによって非推奨となっています。

7. オプション: データ・ソースを開始してテストするには、「**ログ・ファイル情報**」ページで「**ログ・ファイルの設定のテスト**」をクリックします。. ログ・ソースについてのオプションを 選択した後で、「**ログ・ファイルの設定のテスト**」をクリックします。ログ・ファイル・データ・ソースをテストし、ログ・ファイルの内容を入力すると、Agent Builder はログの構文解析の結果に基づいて、グループ内の属性を自動的に作成します。テストについて詳しくは、1270 ページの『ログ・ファイル属性グループのテスト』を参照してください。
8. 前のステップでテスト機能を使用せず、「**ログ・ファイル情報**」ページの「**ログ・ファイル情報**」領域にログ・ファイル名を入力した場合は、以下のステップを実行します。

a) 「**次へ**」をクリックして「**属性情報**」ページを表示し、属性グループ内に最初の属性を定義します。

b) 「**属性情報**」ページで情報を指定し、「**完了**」をクリックします。

注 : ログ・ファイル属性グループが、デフォルトの最小バージョンである Tivoli Monitoring バージョン 6.2.1 以降のエージェントに追加された場合、「ログ・ファイル状況」属性グループが組み込まれます。「ログ・ファイル状況」属性グループの詳細については、 1463 ページの『「ログ・ファイル状況」属性グループ』を参照してください。

すべてのデータ・ソースに適用可能なフィールドに加えて、ログ・ファイル・データ・ソースの「**属性情報**」ページの「**フィールド 情報の記録**」領域には、いくつかの追加フィールドがあります。

「**フィールド 情報の記録**」フィールドは、以下のとおりです。

次のフィールド
属性グループの区切り文字 (または「拡張」ダイアログのこの属性の特殊な区切り文字) を使用して、解析後に次のフィールドを表示します。

レコードの残り
前の属性が解析された後にレコードの残りを表示します。 この属性は最後の属性です (ログ・ファイル名またはログ・ファイル・ラベルを除く)。

レコード全体
レコード全体を表示します。これは唯一の属性になります (ログ・ファイル名またはログ・ファイルのラベルを除く)。

ログ・ファイル名
ログ・ファイルの名前を表示します。

ログ・ファイル・ラベル
拡張パネルでファイルに割り当てられたラベルを表示します。

注 :「**派生属性の詳細**」タブは、ログ・ファイルにある直接の属性ではなく、派生属性が必要な場合にのみ使用します。

9. 「**フィールド情報の記録**」領域で「**拡張**」をクリックして、「**詳細なログ・ファイル属性情報**」ページを表示します。
   a) 「**属性フィルター**」セクションで、組み込むかまたは除外するデータの基準を指定します。
   属性をフィルタリングすることで、処理するデータの量が減るため、ソリューションのパフォーマンスが向上します。1 つ以上の属性フィルターをクリックします。
   - 「**包括的**」の場合、属性フィルター・セットは受け入れフィルターです。つまり、フィルターが正常に処理されると、レコードがフィルターを通過して、出力されます。
   - 「**排他的**」の場合、属性フィルター・セットは拒否フィルターです。つまり、属性フィルターが正常に処理されると、レコードは拒否され、出力されません。
   - 「**すべてのフィルターに一致**」の場合、フィルターが正常に処理されるためには、フィルターに定義されているすべてのフィルターに属性レコードが一致する必要があります。
   - 「**いずれかのフィルターと一致**」の場合、フィルターに定義されているいずれかのフィルターに属性レコードが一致すると、フィルターが正常に処理されます。

   b) 「**追加**」、「**編集**」および「**削除**」を使用して、属性フィルター・セットの個々のフィルターを定義します。
   c) フィルターを追加するには、以下の手順を実行します。
      1) 「**追加**」をクリックし、「**フィルターの追加**」ウィンドウのオプションを次のように設定します。
         a) 「**フィルターの基準**」セクションはフィルターの 基本的特性を定義し、 以下のプロパティーが用意されています。
         - 「**開始オフセット**」は、属性ストリング内の比較開始位置を定義します。
         - 「**比較ストリング**」には、パターン・ストリングを定義します。このパターンに一致するレコードのみを検索結果に含めるか、一致するレコードを検索結果から除外します。

           ファイルから読み取るデータをフィルタリングするために、エージェントによって使用されるストリング、パターン、または正規表現を入力します。フィルター・パターンに一致するレコードが、モニター環境に返されるレコードから除去されるか、それらのレコードのみがモニター環境に返されます。結果は、フィルターが包括的であることを選択したか排他的であることを選択したかによって異なります。
         - 「**値全体と一致**」は、属性ストリング内の比較ストリングと正確に一致するものを検索します。検査は、開始オフセット位置から開始されます。
         - 「**値のいずれかの部分と一致**」は、属性ストリング内の比較ストリングのいずれかの部分と一致するものを検索します。検査は、開始オフセット位置から開始されます。

         b) 「**比較ストリングは正規表現**」は、比較ストリングが、属性ストリングに対して適用可能な正規表現パターンであることを指定します。

検査対象の属性値が指定のパターンに一致するかどうかを調べるために、International Components for Unicode (ICU) ライブラリーを使用した正規表現フィルタリングのサポートが提供されます。

正規表現サポートを有効に活用するには、ICU による正規表現の実装に関する詳細を理解しておく必要があります。この実装は、`Perl`、`grep`、`sed`、Java 正規表現などでの正規表現サポートの実装方法とは異なります。 正規表現フィルターの作成について詳しくは、1489 ページの『ICU 正規表現』を参照してください。

c)「**オーバーライド・フィルターの定義**」は、 以前に定義した基本的特性をオーバーライドする詳細なフィルター比較を 指定することを示します。この追加の比較ストリングは、フィルター結果の反転に使用します。フィルターが「**包括的**」である場合、オーバーライドはフィルター式の排他修飾子として機能します。フィルターが「**排他的**」である場合、オーバーライドはフィルター式の包括修飾子として機能します。(「**包括的**」 と **排他的**」について詳しくは、ステップ 1265 ページの『9』 とそれに続く例を参照してください)。オーバーライド・フィルターには以下のプロパティーがあります。

- 「**開始オフセット**」は、属性ストリング内の比較開始位置を定義します。
- 「**比較ストリング**」には、属性が一致するパターン・ストリングを定義します。

  ファイルから読み取るデータをフィルタリングするために、エージェントによって使用される正規表現を入力します。フィルター・パターンに一致するレコードが、モニター環境に返されるレコードから除去されるか、それらのレコードのみがモニター環境に返されます。結果は、フィルターが包括的であることを選択したか排他的であることを選択したかによって異なります。

d)「**置き換え値**」は、raw 属性ストリングを新しい値で変更する場合に使用します。使用できる特殊文字について詳しくは、1489 ページの『ICU 正規表現』を参照してください。

e)「**最初の検索結果を置換**」では、比較ストリングに一致した最初の検索結果が、新規テキストに置き換えられます。

f)「**すべての検索結果を置換**」では、比較ストリングに一致したすべての検索結果が、新規テキストに置き換えられます。

2)「**OK**」をクリックします。

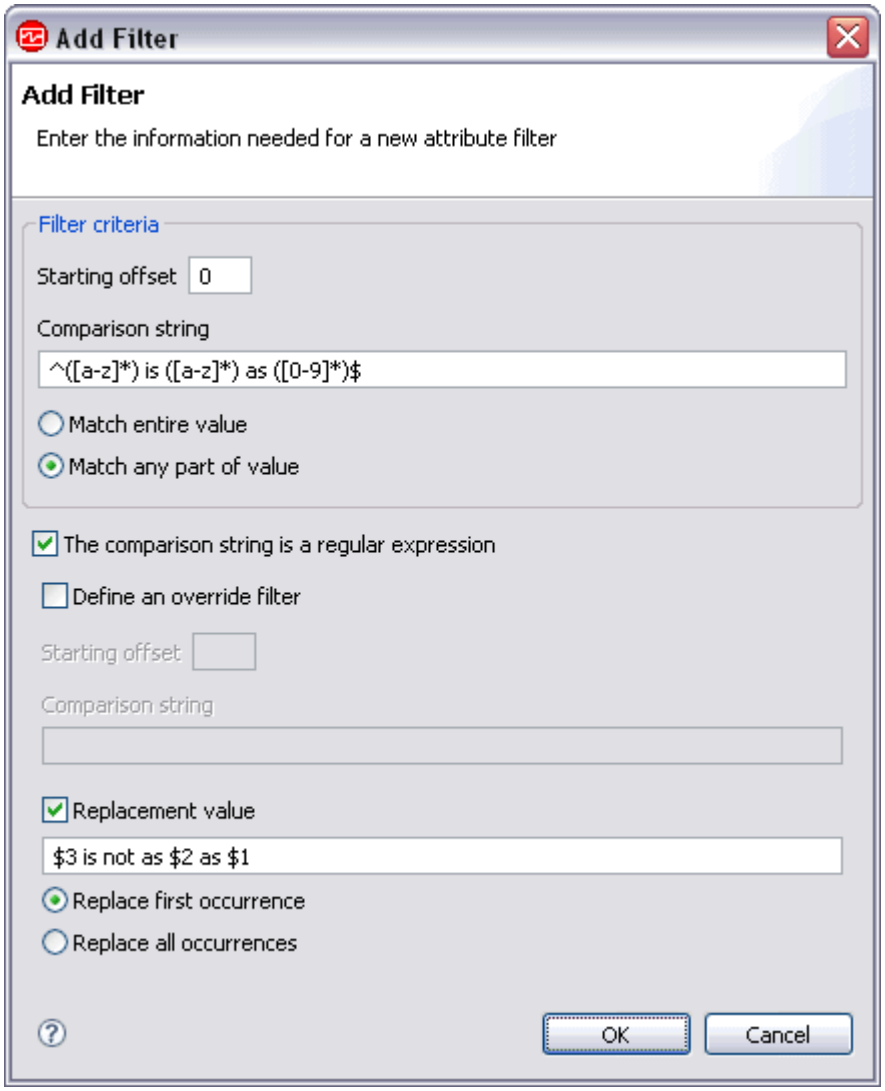


図 *38.* **フィルターの追加**の例 *1*

属性ストリングが `abc is easy as 123` の場合、置換後のテキストは Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに `123 is not as easy as abc` と表示されます。

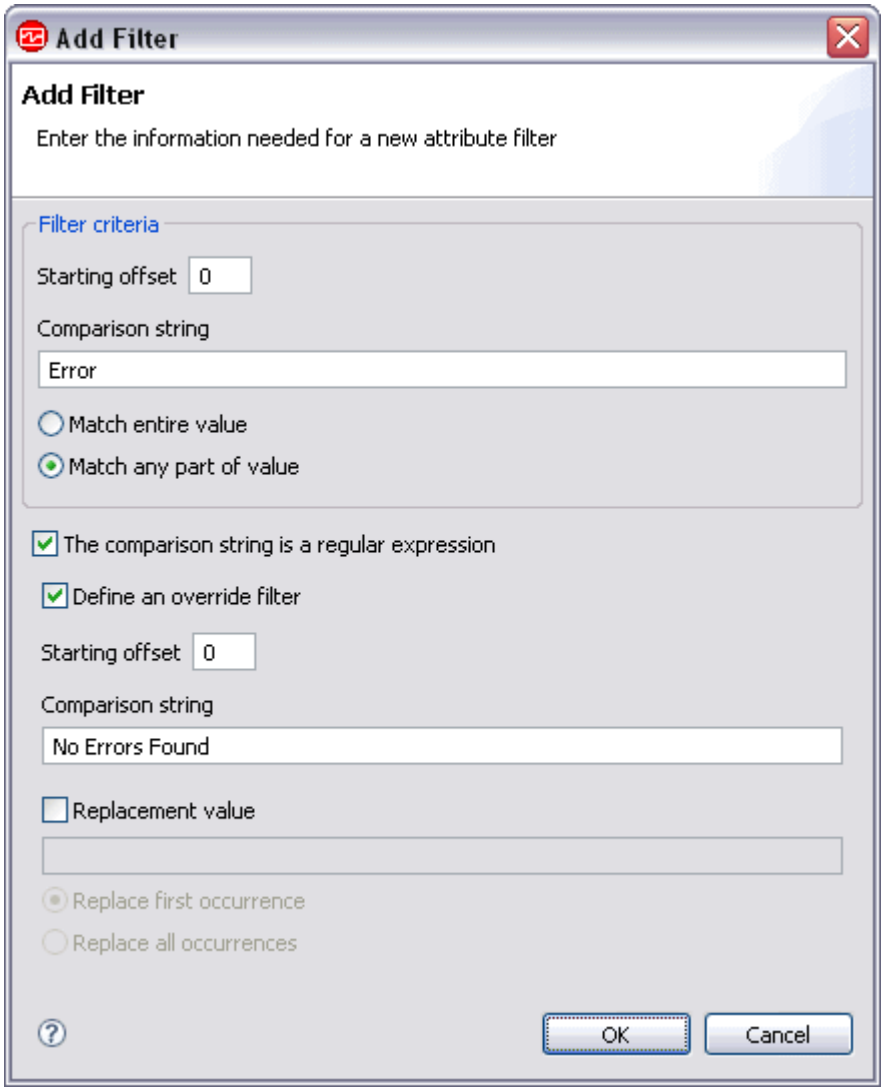


図 *39.* **フィルターの追加**の例 *2*

属性ストリングが「`Unrecoverable Error reading from disk`」であり、フィルターが「**包括的**」である場合は、その属性が Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールに表示されます。属性ストリングが「`No Errors Found during weekly backup`」であり、フィルターが「**包括的**」である場合、その属性は表示されません。

d) 「**詳細なログ・ファイル属性情報**」ページの「**フィールド ID**」セクションでは、この 1 つの属性のみについて、属性グループ・フィールド区切り文字の指定変更方法を指定します。いずれかの属性フィルターをクリックし、オプションの必須フィールドに情報を入力します。

- **文字数**: 文字数の制限を入力します。
- **タブ分離文字**: タブ分離文字の用途を指定します。
- **分離文字テキスト**: 使用する分離文字テキストを入力します。
- **開始テキストと終了テキスト**: **開始**テキストと**終了**テキストの両方を入力します。

e) 「**詳細なログ・ファイル属性情報**」ページの「**要約**」セクションで「**要約属性グループに属性を追加する**」チェック・ボックスをクリックして、要約属性グループに属性を追加します。

ユーザーがログ属性要約をオンにすると、属性グループが作成されます。

f) 「**OK**」をクリックします。

10. ステップ 1264 ページの『7』 でテスト機能を使用した場合は、「**キー属性の選択**」ページが表示されます。「**キー属性の選択**」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。

詳しくは、1174 ページの『キー属性の選択』 を参照してください。

11. 以下の手順のいずれかを行います。
    - 新規エージェント・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

    注 : ログ・ファイル属性グループが、デフォルトの最小バージョンである Tivoli Monitoring バージョン 6.2.1 以降のエージェントに追加された場合、「ログ・ファイル状況」属性グループが組み込まれます。「ログ・ファイル状況」属性グループの詳細については、1463 ページの『「ログ・ファイル状況」属性グループ』を参照してください。

ログ・ファイル構文解析および分離文字

ログ・ファイル・レコード内の 1 つ以上の属性を区切るために使用するデフォルト分離文字を変更できます。

ログ・ファイル属性グループを作成すると、分離文字がデフォルトで割り当てられます。デフォルトの区切り文字はタブです。分離文字は、エージェントがデータ行の各属性のデータを構文解析し、区切るために使用されます。デフォルトの属性分離文字を以下の文字に変更できます。

- 一定数の文字
- スペース
- 1 文字以上の異なる文字
- 特定の開始テキストおよび終了テキスト
- XML エレメント。

以下の方法で、グループ内のすべての属性に使用するデフォルトの分離文字を変更することができます。

1. 属性グループの作成中に、「**ログ・ファイル情報**」ページで変更する。
2. 属性グループの作成後に、「**エージェント・エディター**」 > 「**データ・ソース**」タブを開き、属性グループを選択して、「**フィールド ID**」領域で分離文字を選択する。

必要に応じて、特定の分離文字を 1 つ以上の個々の属性に割り当てることもできます。 特定の分離文字を個々の属性に割り当てる際、以下の文字を使用できます。

- 一定数の文字
- タブ分離文字
- スペース分離文字
- 1 文字以上の異なる文字
- 特定の開始テキストおよび終了テキスト

以下の方法で、個々の属性に使用する分離文字を変更することができます。

1. 属性の作成時に、「**属性情報**」ページで「**拡張**」を選択する。
2. 「**エージェント・エディター**」 > 「**データ・ソース**」タブを開き、属性を選択して、「**ログ・ファイル属性情報**」タブで「**拡張**」 を選択する。

例 1 - 単純なログ・ファイル出力

一部のログ・ファイル・レコードには、明確かつ通常の分離文字が含まれています。以下に例を示します。

```
one,two,three
```

この例で、"," 文字は、行のデータの 3 つの部分を分離する明確かつ通常の分離文字です。この場合、「**分離文字テキスト**」を選択し、"," を属性グループのデフォルトの 分離文字として指定します。変更したり、他の分離文字を定義したりする必要はありません。

上記の例で使用したデータ行を含むログ・ファイルに対してこの分離文字を定義すると、出力は以下のように表示されます。

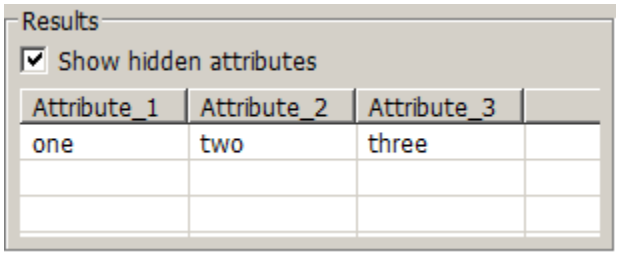


図 *40.* エージェントが単純なログ・ファイル・データ行を構文解析するときの属性値出力の例

例 2 - 複雑なログ・ファイル出力

一部のログ・ファイルには、不規則な分離文字または可変分離文字を含むデータ行が含まれていることがあります。以下に例を示します。

```
one,two,three,[four]12:42,five
```

この例では、属性定義に対して、以下の分離文字を割り当てることができます。

1. 前の例では、デフォルト分離文字を "," に設定しました。特定の分離文字でオーバーライドしない限り、すべての属性でこの分離文字が使用されます。この例でも、行の最初の 3 つの属性でデフォルト分離文字 "," の使用は適切です。
2. 4 番目の属性では、"[" と "]" の間のストリングの値を抽出すると仮定します。この場合には、4 番目の属性を定義するときに、開始テキスト値と終了テキスト値が "[" と "]" に設定された「**開始テキストと終了テキスト**」という分離文字タイプを割り当てます。
3. 5 番目の属性では、"]" 文字と ":" 文字の間の値を抽出すると仮定します。 この場合には、5 番目の属性を定義するときに、":" に設定された「**分離文字テキスト**」という分離文字タイプを割り当てます。
4. 6 番目の属性では、デフォルトの属性グループ分離文字 "," が再び適切になります。
5. 7 番目の属性は最後の属性なので、この属性に分離文字を指定する必要はありません。

上記の例で示されたデータ行を含むログ・ファイルでこれらの分離文字を定義した場合の出力は以下のとおりです。

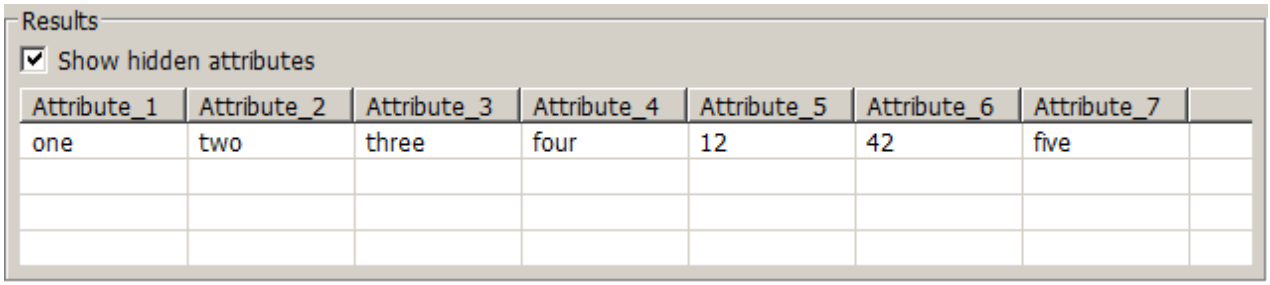


図 *41.* エージェントが複雑なログ・ファイル・データ行を構文解析するときの属性値出力の例

属性分離文字を定義する手順については、1260 ページの『ログ・ファイルのモニター』のステップ 1261 ページの『5』 で説明されています。

ログ・ファイル属性グループのテスト

Agent Builder を使用して、作成したログ・ファイル・データ・セット (属性グループ) をテストできます。グループに対して属性が定義されていない場合、テスト・プロセスによって属性が自動的に定義されます。

始める前に

いずれかの属性が既にこのデータ・セットに対して定義されている場合に、テスト中に自動的に属性を定義するには、Agent Editor を使用して、データ・セットからのすべての既存の属性を削除します。説明については、1197 ページの『属性の削除』を参照してください。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**ログ・ファイル情報**」ページで、「**ログ・ファイルの設定のテスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**ログ・ファイルの設定のテスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**ログ・ファイルの設定のテスト**」をクリックすると、「**ログの構文解析**」ウィンドウが開きます。
2. 次のように、テストするログ・データのソースを選択します。
   - **属性グループの設定を使用**: データ・ソースに指定されているファイル名とロケーションを使用します。デフォルトでは、データ・ソースは、テスト・プロセスの開始後にログ・ファイルに追加された情報のみを処理します。ログ・ファイルがリアルタイムで更新される場合は、このオプションを使用できます。
   - **サンプル・ファイルを指定**: サンプルのログ・ファイルを指定します。この設定では、テスト・プロシージャーは、ログ・ファイルの内容全体を構文解析します。このオプションを使用すると、データ・ソースをテストし、それに対する属性を、既存のサンプルに基づいて迅速に作成できます。「**ログ・ファイル名**」フィールドにファイルのパスと名前を指定するか、「**参照**」ボタンを使用してファイルを選択します。
3. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。

   詳しくは、1376 ページの『属性グループのテスト』 を参照してください。
4. 「**エージェントの開始**」をクリックします。

   ウィンドウが開いて、エージェントが開始されていることが表示されます。エージェントが開始されると、エージェントは構成済みログ・ファイルで新規レコードをモニターします。
5. エージェントのデータ収集をテストするには、モニター対象ログ・ファイル内に新規レコードを生成します。

   新規レコードがログ・ファイルに追加されると、エージェントはその構成に従ってレコードを構文解析し、キャッシュ内で対応する属性値を更新します。
6. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。

   「**ログの構文解析**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべての新規属性値が収集および表示されます。データ収集の例を 1272 ページの図 42 に示します。

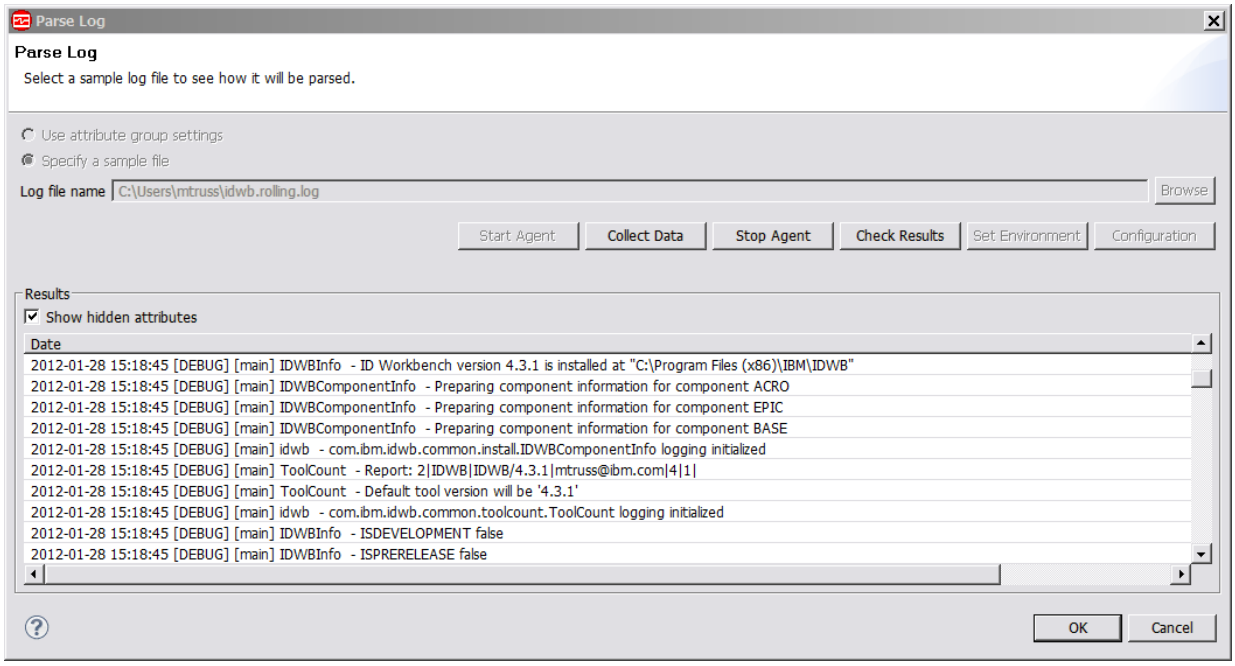


図 *42.* 「ログの構文解析」ウィンドウ。構文解析されるログ・ファイルの属性値が表示されています。

7. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。
   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
8. 「**エージェントの停止**」をクリックすると、エージェントは停止します。
9. 「**OK**」または「**キャンセル**」をクリックして「**ログの構文解析**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

AIX バイナリー・ログのモニター

データ・ソースを定義して、`errpt` コマンドを使用した AIX バイナリー・エラー・ログのモニターを実行できます。データをフィルタリングおよび 要約するように構成することもできます。結果のイベントは、データ・セット内に配置されます。

このタスクについて

ログ・モニターでは、`errpt` コマンドを使用した AIX バイナリー・エラー・ログのモニターがサポートされています。 `errpt` コマンドは、エラー・ログのエントリーからエラー・レポートを生成します。特定の基準に一致するエラーを選択するためのフラグがあります。`errpt` コマンドを使用した AIX バイナリー・エラー・ログのモニターのサポートは、Tivoli Monitoring UNIXLogs Agent (製品コード「`kul`」または「`ul`」) の同様の機能のサポートをモデルとしています。

`errpt` コマンド・ストリングを Agent Builder に対して指定すると、Agent Builder はこのコマンドの実行結果として生成されるイベントを処理します。Agent Builder では、Monitoring Agent for UNIX Logs が適用するのと同じ制約をこのコマンドに適用します。特に、コマンドを連続的に実行するために「**`-c`**」オプション (並行モード) を使用する必要があります。「**`-t`**」オプション、または詳細出力を生成する「**`-a`**」、「**`-A`**」、「**`-g`**」オプションを使用することはできません。

AIX **`errpt`** コマンドをモニターする Agent Builder エージェントは、Monitoring Agent for UNIX Logs の場合と同じ情報を自動的に組み込みます。AIX バイナリー・エラー・ログの属性グループについて詳しくは、1435 ページの『「AIX バイナリー・ログ」属性グループ』を参照してください。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**ログに記録されたデータ**」をクリックします。
2. 「**データ・ソース**」領域で「**AIX バイナリー・ログ**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**バイナリー・ログ情報**」ページに、errpt コマンドを入力します。

   デフォルト値は以下のとおりです。

   ```
   errpt -c -smmddhhmmyy
   ```

   エージェントは始動時に「mmddhhmmyy」ストリングを検索し、このストリングを実際の日時に置き換えます。ストリングの 1 番目のオカレンスのみが置換されます。

   ユーザーは各自の errpt コマンドを指定できますが、Agent Builder では、Monitoring Agent for UNIX Logs の場合と同じ制約がこのコマンドに適用されます。特に、コマンドを連続的に実行するために「**-c**」オプション (並行モード) を使用する必要があります。「**-t**」オプション、または詳細出力を生成する「**-a**」、「**-A**」、「**-g**」オプションを使用することはできません。
5. (オプション)「**拡張**」をクリックして、イベントのフィルタリングおよび要約オプションを選択します。詳しくは、1406 ページの『重複イベントの制御』 を参照してください。
6. 以下の手順のいずれかを行います。
   - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
   - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

関連資料

1435 ページの『「AIX バイナリー・ログ」属性グループ』
「AIX バイナリー・ログ」属性グループには、指定された errpt コマンド・ストリングにより選択された AIX バイナリー・ログのイベントが表示されます。

Windows イベント・ログのモニター

データ・ソースを定義して、Windows イベント・ログからデータを収集することができます。データをフィルタリングするように構成できます。結果のイベントは、「イベント・ログ」データ・セット内に配置されます。

このタスクについて

Windows イベント・ログからデータを収集する際、イベントのタイプ、ソース、または ID を使用します。これらのパラメーターを使用して、Windows システムが収集したログ・イベントをフィルタリングします。エージェントは、モニター対象イベント・ログ内のそれぞれの新規イベントを、指定されたフィルターと比較します。 フィルターで指定されたいずれかのイベント・タイプ、イベント・ソース、およびイベント ID と一致したイベントは、フィルター操作にパスします。

例: アプリケーション・ログをイベント・ログ・フィルターに掛ける場合は、イベント・タイプとして **Error** を指定します。これを選択すると、アプリケーション・ログに記録されているすべてのイベントと、イベント・タイプ値 error のマッチングが行われます。イベント・ソースとして **Diskeeper** および **Symantec AntiVirus** を追加すると、エージェントはこれらのソースのいずれかから得られたエラー・イベントもすべてマッチングします。特定のイベント ID を追加して、さらにフィルターを詳細化できます。 イベント・タイプ、イベント・ソース、およびイベント ID の間には直接の関連は存在しません。 それぞれの値のいずれかがあるイベントと一致すると、そのイベントは一致したことになります。

デフォルトでは、エージェント開始後に生成されるイベントのみが処理されます。ただし、エージェントの再始動時に、シャットダウン中に生成されたログ・イベントを処理するようにエージェントを設定できます。 エージェントがシャットダウン中に生成されたイベントをエージェントが処理できるように設定する方法について詳しくは、ステップ 1274 ページの『6』 を参照してください。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**ログに記録されたデータ**」をクリックします。
2. 「**データ・ソース**」領域で「**Windows イベント・ログ**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**Windows イベント・ログ**」ページで、「**Windows イベント・ログ名**」リストからログの名前を 1 つ選択するか、またはイベント・ログの名前を入力します。

   このリストは、現行システムにあるログのセットから構成されています。以下に例を示します。

   アプリケーション
   セキュリティー
   システム
5. 「**Windows イベント・ログ**」ページで、以下のメカニズムの内の 1 つ以上を使用して結果をフィルタリングするかどうかを指定します。

   - 1275 ページの『イベント・タイプによるフィルタリング』
   - 1275 ページの『イベント・ソースによるフィルタリング』
   - 1275 ページの『イベント ID によるフィルタリング』

   注 : これらのフィルター基準のうち、少なくとも 1 つを選択する必要があります。
6. エージェントのシャットダウン中に生成されたログ・イベントを、エージェントの再始動時に処理するには、「**Windows イベント・ログ**」ページで「**オフライン・イベント設定**」をクリックします。

   「**Windows イベント・ログのブックマーク設定**」ウィンドウが開きます。
7. 以下のブックマーク・オプションのいずれかを選択します。

   注 : これらのオプションは、モニター対象 Windows イベント・ログすべてに適用されます。

   - **いずれのオフライン・イベントも収集しない**: エージェントがシャットダウン中に生成されたイベントは処理されません。このオプションはデフォルト ・オプションです。
   - **すべてのオフライン・イベントを収集する**: エージェントがシャットダウン中に生成されたすべてのイベントが処理されます。
   - **カスタム収集設定を指定する**: 時間値またはイベント数、あるいはこの両方に基づいて古いイベントの処理を制御するための値を入力できます。このオプションを使用することで、エージェントの開始時にイベントによってモニター環境に過負荷がかかることがないようにすることができます。

     例えば、「**収集するイベントの最大数**」フィールドに 100 を入力し、「**限定収集期間 (秒)**」フィールドに 30 を入力したとします。この場合、処理されるイベント数は、エージェント開始前に生成された最終 100 イベント、またはエージェント始動前の 30 秒間に生成されたすべてのイベントのいずれかになります。どちらの結果になるかは、最初に一致した変数に基づきます。

     収集する最大イベント数の値を入力すると、 *CDP_DP_EVENT_LOG_MAX_ BACKLOG_EVENTS* 環境変数が追加されます。 限定収集期間の値を入力すると、 *CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME* 環境変数が追加されます。これらの変数のいずれかまたは両方が追加されると、*eventlogname_productcode_instancename_subnodename*.rst ファイルが作成されます。このファイルには、イベント・ログ用に処理される最後のイベント・レコードが含まれます。このファイルは *%CANDLE_HOME%*¥tmaitm6¥logs ディレクトリーに格納されており、エージェントが再始動され、エージェントのシャットダウン中に生成された古いイベントの処理を実行するときに使用されます。
8. データ・ソースのグローバル・オプションを設定する場合は、「**Windows イベント・ログ**」ページの「**グローバル・オプション**」をクリックします。

   「**Windows データ・ソースのグローバル・オプション**」ウィンドウが開きます。
9. このオプションを含める場合は、「**リモート Windows の構成プロパティーを含める** 」チェック・ボックスを選択して 「**OK**」をクリックします。

   Windows データ・ソースの Windows リモート接続構成については、1363 ページの『Windows リモート接続の構成』を参照してください。

10. フィルターを指定して「**OK**」をクリックした後、「**Windows イベント・ログ**」ページで、以下の手順のいずれかを行います。
    - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。新規 Windows イベント・ログの名前が **Agent Editor** の「**データ・ソース定義**」ページに表示されます。

次のタスク

Windows イベント・ログのデータ・ソースの Windows リモート接続構成については、1363 ページの『Windows リモート接続の構成』を参照してください。

イベント・タイプによるフィルタリング

イベント・タイプにより Windows イベント・ログ結果をフィルタリングします。

手順

1. 「**Windows イベント・ログ**」ページで、「**イベント・タイプによるフィルタリング**」を選択します。
2. 以下のイベント・タイプを 1 つ以上選択してください。
    - **情報**
    - **警告**
    - **エラー**
    - **成功の監査**
    - **障害の監査**
3. 「**完了**」をクリックして完了します。

イベント・ソースによるフィルタリング

イベント・ソースにより Windows イベント・ログ結果をフィルタリングします。

手順

1. 「**Windows イベント・ログ**」ページの「**イベント・ソース**」領域で「**イベント・ソースによるフィルタリング**」を選択し、「**追加**」をクリックします。
   「**イベント・ソース**」ウィンドウが開きます。
2. 以下のいずれかを選択します。
    - イベント・ソース名を入力して、「**OK**」をクリックします。
    - 「**参照**」をクリックして、リストからイベント・ソースを見つけて選択し、「**OK**」をクリックします。
      選択した名前が「**イベント・ソース**」ウィンドウに表示されます。
      注 :
      a. イベント・ソースのリストをソートするためには、列見出しをクリックします。
      b. ウィンドウ内の情報を最新表示するためには、「**最新表示**」アイコンをクリックします。
      c. 特定のイベント・ソースを検索するには、「**検索**」(双眼鏡) アイコンをクリックします。
3. 「**OK**」をクリックして、「**Windows イベント・ログ**」ウィンドウの「イベント・ソース」リスト内の新規イベント・ソース・フィルターを確認します。

イベント ID によるフィルタリング

Windows イベント・ログ・データ・ソースでは、イベント ID によりイベントをフィルタリングできます。

このタスクについて

イベント ID によってフィルタリングするためには、以下の手順に従ってください。

手順

1. 「**Windows イベント・ログ**」ウィンドウの「**イベント ID**」領域で「**イベント ID によるフィルタリング**」を選択し、「**追加**」をクリックします。
   「**イベント ID**」ウィンドウが表示されます。
2. あるアプリケーションの特定イベントをモニターする場合、アプリケーションで定義されたイベントの番号を指定します。イベント ID として整数を入力し、「**OK**」をクリックします。
   新規の数値イベント ID フィルターが、「**Windows イベント・ログ**」の「イベント ID」リストに表示されます。

   注 : 各イベント ID は個別に定義する必要があります。
3. Windows イベント・ログを変更したい場合には、それを選択して「**編集**」をクリックします。
4. Windows イベント・ログを削除したい場合には、それを選択して「**削除**」をクリックします。
5. リストにイベント・ログを追加することも、「**完了**」をクリックすることもできます。

コマンドの戻りコードのモニター

コマンドの戻りコード を使用してアプリケーションまたはシステムをモニターするためのデータ ・ソースを定義できます。エージェントは、コマンドを実行し、戻りコードを収集し、可用性データ・セットに結果を追加します。

このタスクについて

ユーザー作成のスクリプト、実行可能ファイル、照会、システム・コマンドは、コードを返すことができます。コマンドの戻りコードは、アプリケーション固有のメカニズムであり、アプリケーションまたはモニター対象のシステムが使用可能であるかどうかを判別します。エージェントが、指定されたコマンドを実行し、戻りコードを調べてアプリケーションまたはモニター対象のシステムの状態を判別します。

コマンドは、記述する状態ごとに固有の戻りコードを表示する必要があります。コマンドは、それらの各戻りコードについてエージェントが使用するメッセージを定義する必要もあります。ユーザー作成のスクリプト、実行可能ファイル、照会、またはシステム・コマンド内では、コマンドは環境変数および構成変数を使用できます。 コマンドは、コマンドのコマンド行呼び出しでは環境変数および構成変数を使用できません。ただし、*AGENT_BIN_DIR*、*AGENT_ETC_DIR*、*AGENT_LIB_DIR*、*CANDLE_HOME*、*CANDLEHOME* を除きます。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**コマンドまたはスクリプト**」を選択します。
2. 「**データ・ソース**」域で、「**コマンドの戻りコード**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**コマンドの戻りコード**」ページの「**コマンドの戻りコード情報**」域に表示名を入力します。
5. コマンド戻りコードで使用するコマンド行を定義し、記述するには、以下のサブステップを実行します。

   注 : エージェントによってサポートされるオペレーティング・システムごとにコマンドを定義してください。コマンドは共有可能ですが、すべてのコマンドに対応するオペレーティング・システムの集合全体が、エージェントがサポートしているオペレーティング・システムの集合と同じになっている必要があります。

   a) 「**コマンドの戻りコード**」ウィンドウの「**コマンド**」領域で「**追加**」をクリックし、「**コマンド情報**」ウィンドウを開きます。

   b) コマンド行を入力して、「**コマンド情報**」ウィンドウの「**オペレーティング・システム**」域に表示されたリストからオペレーティング・システムを選択します。

   注 :

   1) Windows コマンドの場合は、コマンドの完全な名前を入力する必要があります。例えば、`command_to_run` のみではなく、`command_to_run.bat` と入力する必要があります。

2) コマンド・インタープリターによる構文解析が行われないように、名前は引用符で囲んでください。例えば、"this is a test.bat"argument と入力します。this is a test.bat argument とは入力しないでください。

3) コマンドをクリックして「**編集**」をクリックすると、そのコマンドを変更することができ、「**削除**」をクリックするとそのコマンドを削除することができます。

c)「**コマンド情報**」ウィンドウの「**戻りコード**」域で「**追加**」をクリックします。

d)「**戻りコード定義**」ウィンドウに表示されたリストから、戻りコードのタイプを選択します。

テスト戻りコードに以下の状態を割り当てることができます。

- ALREADY_RUNNING
- DEPENDENT_NOT_RUNNING
- GENERAL_ERROR
- NOT_RUNNING
- OK
- PREREQ_NOT_RUNNING
- WARNING

e) 選択した戻りコード・タイプに対応する数値を入力してください。

戻りコード値は、コマンド戻りコードに関して定義された戻りコードを指定する整数です。 オペレーティング・システム間での移植性を 考慮して、0 から 255 の戻りコード値を使用してください。Windows のみで実行するコマンドの場合は、戻りコードの値を -2147483648 から 2147483647 までにすることができます。

f) それぞれの戻りコードごとにメッセージを定義して、メッセージとコードを一緒に表示できるようにします。「**参照**」をクリックして、メッセージ・テキストをセットアップします。

メッセージ・ウィンドウには、エージェントで定義されたメッセージがリストされます。「**メッセージ**」(リスト) ウィンドウが開きます。

注 :

1)「**参照**」をクリックする代わりに、以前に入力したテキストをメッセージ・テキストのリストで選択することにより、そのテキストを指定できます。その後は、 ステップ 5k に進んでください。

2) メッセージを定義するまで、リストはブランクになっています。 「**編集**」を使用すると定義済みメッセージを変更することができ、「**削除**」を使用すると定義済みの 1 つ以上のメッセージを削除することができます。

g)「**メッセージ**」(リスト) ウィンドウで、「**追加**」をクリックします。

「**メッセージ定義**」ウィンドウが開きます。

注 : メッセージ ID は自動的に生成されます。

h)「**メッセージ・テキスト**」フィールドに、新規メッセージの意味を示すテキストを入力します。

i)「**OK**」をクリックします。

「**メッセージ**」(リスト) ウィンドウが開き、新規メッセージが表示されます。

j) メッセージを確認して永続させるためには、リストからそれを選択して「**OK**」をクリックします。

「**戻りコードの定義**」ウィンドウに、新しい戻りコードのタイプ、値、およびテキストが表示されます。

k) このコマンドの戻りコードについて、他のオペレーティング・システムにおける他のコマンドでもこの戻りコードを使用できるようにする場合は、「**グローバル戻りコードをすべてのコマンドに適用する**」を 選択します。この戻りコードをこのコマンドのみで使用できるようにする場合は、 「**ローカル戻りコードをこのコマンドにのみに適用する**」が 選択されたままにします。

l)「**戻りコードの定義**」ウィンドウで「**OK**」をクリックします。

m) 少なくとも 2 つの戻りコードを定義してから、「**コマンド情報**」ウィンドウを終了します。戻りコードの 1 つは可用性の問題がないことを示すもの、もう 1 つは問題が発生したかどうかを示すものです。別の戻りコードを追加する場合は、 ステップ c に戻ります。

n) オプション: エージェントが実行する 1 つ以上のスクリプトまたは実行可能ファイルを選択する場合は、「**コマンド情報**」ウィンドウの「**コマンド・ファイル**」領域で「**追加**」をクリックします。

このファイルは、scripts/*operating system* の下にあるエージェントのプロジェクト・フォルダーにコピーされます。*operating system* は、「**コマンド情報**」ウィンドウの「**オペレーティング・システム**」 領域で選択した内容に応じた変数です。これらのファイルは、エージェントを使用してパッケージ化され、配布されます。既存のコマンド・ファイルの定義を編集したり、プロジェクトへのコピー後に元のコマンド・ファイルの定義を編集したりするには、ファイルを選択して「**編集**」をクリックします。1279 ページの『コマンド・ファイル定義の編集』 を参照してください。

o)「**コマンド情報**」ウィンドウで「**OK**」をクリックします。

注 : コマンド・ファイル・テーブルには、エージェント・パッケージに含めるすべての外部ファイルが定義されています。 これらのファイルはプロジェクト・ディレクトリーにコピーされ、エージェントと一緒にパッケージ化されて配布されます。

6. まだ定義していない戻りコードがある場合は、コマンド戻りコードで使用できるグローバル戻りコードを定義し、記述します。

a)「**コマンドの戻りコード**」ページの「**グローバル戻りコード**」領域で「**追加**」をクリックします。

注 : ここで定義した戻りコードはグローバルです。 つまり、コマンド戻りコード用に定義されたすべてのコマンドに適用される戻りコードです。(コマンドの戻りコードの間では 共有されません)。また、コマンド情報を入力するときに戻りコードを定義することもできます。ここで定義した戻りコードは、 グローバルにすることもローカルにすることもできます。ローカルな戻りコードは、 このコマンドだけに適用されます。この階層は、戻りコードが全オペレーティング・システムを通じて同じである場合に役立ちます。(例えば、戻りコード 0 は、すべてが正常に機能していることを意味します。グローバル・レベルでこのように定義しておくと、定義済みのすべてのコマンドで 0 がそのように解釈されます。) 他のどのオペレーティング・システムでも 5 が返されない場合は、戻りコード 5 を Windows コマンドにのみ定義することができます。既にグローバル・レベルで定義した戻りコードをローカル・コマンド・レベルで定義した場合は、コマンド・レベルのものが使用されます。この方法により、特定のオペレーティング・システムの戻りコードをオーバーライドすることができます。例えば、すべての UNIX オペレーティング・システムでは戻りコード 2 がある状態を表し、Windows では戻りコード 2 が違う状態を表すとします。この場合には、戻りコード 2 を、UNIX オペレーティング・システムで 予期されるものとして、まずグローバル・レベルで定義します。次に、Windows のコマンドで、戻りコード 2 を Windows での意味に再定義できます。

b)「**戻りコード定義**」ウィンドウに表示されたリストから、戻りコードのタイプを選択します。

テスト戻りコードに以下の状態を割り当てることができます。

- ALREADY_RUNNING
- DEPENDENT_NOT_RUNNING
- GENERAL_ERROR
- NOT_RUNNING
- OK
- PREREQ_NOT_RUNNING
- WARNING

c) 選択した戻りコード・タイプに対応する数値を入力してください。戻りコード値は、コマンド戻りコードに関して定義された戻りコードを指定する整数です。

d)「**参照**」をクリックして、メッセージ・テキストとそれに関連付ける意味をセットアップします。それぞれの戻りコードごとにメッセージを定義して、メッセージとコードを一緒に表示するようにする必要があります。

「**メッセージ**」ウィンドウには、エージェントで定義されたメッセージがリストされます。

注 :

1) メッセージを定義するまで、リストはブランクになっています。「**編集**」を使用して定義済みメッセージを変更することができ、また、「**削除**」を使用して定義済みの 1 つ以上のメッセージを削除することができます。

2)「**参照**」をクリックする代わりに、 以前に入力したテキストを「**メッセージ・テキスト**」リストで選択することができます。その後は、 ステップ 6h に進んでください。

e)「**メッセージ**」(リスト) ウィンドウで「**追加**」をクリックして、「**メッセージ定義**」ウィンドウを表示します。このウィンドウで、新規メッセージの意味を示すテキストを入力できます。

f)「**OK**」をクリックします。

g)「**メッセージ**」(リスト) ウィンドウが開き、新規メッセージが表示されます。メッセージを確認して永続させるためには、リストからそれを選択して「**OK**」をクリックします。

h) 新しいテキスト、タイプ、および値が「**戻りコードの定義**」ウィンドウに表示されたら 、「**OK**」をクリックします。

i)「コマンドの戻りコード」ページで、サポートされるすべてのオペレーティング・システムについて戻りコードおよびコマンドの定義が完了したら、以下のいずれかのステップを実行します。

- 新規エージェント・ウィザードを使用している場合は「**次へ**」または「**終了**」をクリックして、データ・ソースを保存し Agent Editor を開いてください。
- 新規エージェント・コンポーネント・ウィザードを使用している場合は、 「**完了**」をクリックして Agent Editor に戻ります。

次のタスク

このデータ・ソースのデータを IBM Cloud Application Performance Management の要約ダッシュボードで使用するには、フィルタリングされたデータ・セット (属性グループ) を可用性データ・セットに基づいて作成し、そのデータ・セットを単一行を生成するデータ・セットとして構成する必要があります。 プロセス用の行を選択するには、「名前」フィールドを使用します。

新しいフィルタリング属性グループで「状況」フィールドを選択し、重大度の値を指定してください。

詳しくは、以下を参照してください。

- 1341 ページの『フィルタリング属性グループの作成』
- 1201 ページの『状況標識として使用される属性の重大度の指定』
- 1373 ページの『Cloud APM 用のエージェントの準備』

コマンド・ファイル定義の編集

プロジェクトにインポートされたコマンド・ファイルを変更できます。また、既存のコマンド・ファイルに対する変更内容をプロジェクトにインポートできます。

手順

1.「**コマンド情報**」ウィンドウの「**コマンド・ファイル**」領域でファイルを選択します。
2.「**編集**」をクリックして「**コマンド・ファイルのインポート**」ウィンドウを開きます。
「**コマンド・ファイルのインポート**」ウィンドウから、コマンド・ファイルの状況を取得できます。また、元のソース・ファイルの場所の変更、およびソース・ファイルのエージェントへの再コピーを行うこともできます。
3. 次のいずれかのステップを選択します。
   - ファイルのコピーがエージェントの次回保存時に行われるようにスケジュールするには、「**OK**」をクリックします。
   - 最初にエージェントを保存せずにファイルをコピーするには、「**今すぐコピー**」をクリックします。

注 : 新規エージェント・ウィザードから「**コマンド・ファイルのインポート**」ウィンドウにアクセスしている場合、「**今すぐコピー**」オプションは使用できません。

ファイルの分離および統合

分離機能および統合機能を使用して、エージェント内のオペレーティング・システム固有のフォルダーとの間でファイルを移動できます。

ファイルがエージェントに最初に追加されるときには、単一のコピーが scripts/all_windows フォルダー、scripts/all_unix フォルダー、または scripts/common フォルダーに追加されます。ファイルが Windows および UNIX の両方で使用される場合、scripts/common フォルダーが使用されます。

オペレーティング・システムに応じて異なるファイル (バイナリー実行可能ファイルなど) のコピーを保管するには、「**編集**」をクリックしてから「**分離**」をクリックします。ファイルは、common フォルダーから削除され、オペレーティング・システム固有のフォルダーにコピーされます。その後で、ファイルの個々のコピーを、特定のオペレーティング・システムに適したコピーに置き換えることができます。

注 : Java リソース・ファイルは、scripts/common フォルダー内に残っている必要があります。「**分離**」をクリックして、個々のオペレーティング・システム用の Java リソース・ファイルの分離コピーを作成することはできません。

ファイルをオペレーティング・システム・フォルダー内に分離した場合は、「**統合**」を使用して common フォルダーに移動できます。common フォルダーをサポートしない Agent Builder のバージョンでエージェントを作成した場合は、「**統合**」を使用して common フォルダーに移動できます。ファイルのコピーのいずれかが他のコピーと異なる場合は、共通ファイルとして使用するファイルを選択するように要求するプロンプトが出されます。その他のコピーはすべて廃棄されます。

スクリプトからのアウトプットのモニター

データ・ソースを定義して、スクリプトまたは外部プログラムからデータを収集することができます。標準管理インターフェースを 介してアプリケーション・データを入手できない場合、または複数行データの要約を単一行で提供する必要がある場合は、これを使用します。エージェントは、スクリプトを実行し、そのアウトプットを収集します。スクリプトのアウトプット内の各行は、結果のデータ・セットの 1 つの行として解析されます。

データは、ローカル・システムまたはリモート ・システムから収集できます。 スクリプトまたはプログラムの出力には、属性グループ内の各属性の値のみを含めるようにする必要があります。複数行のデータが返されるようにするには、各行のデータを改行で区切る必要があります。 各データ行の属性は、ユーザーが定義する分離文字によって分離されます。分離文字について詳しくは、1281 ページの『スクリプト構文解析および分離文字』を参照してください。

ユーザー作成のスクリプト、実行可能ファイル、照会、またはシステム・コマンド内では、コマンドは環境変数および構成変数を使用できます。コマンドのコマンド行呼び出しでは、コマンドは環境変数および構成変数を使用できません。ただし、AGENT_BIN_DIR、AGENT_ETC_DIR、AGENT_LIB_DIR、CANDLE_HOME、CANDLEHOME を除きます。

エージェントがモニター対象とするのは、エージェントが実行されているのと同じロケールおよびコード・ページを使用して作成されるスクリプトのアウトプットです。

リモート・システムからのスクリプト・データの収集

Agent Builder は、リモート・システムからスクリプトまたはプログラム・データを収集するために、セキュア・シェル (SSH) を使用します。

Agent Builder は、リモート・システムからデータを収集するために、セキュア・シェル (SSH) セッションを作成し、リモート・システムでスクリプトまたは外部プログラムを開始します。エージェントは SSH セッションを確立し、ログオンします。次にエージェントは、リモート・システムにスクリプトをアップロードし、スクリプトまたは外部プログラムを開始して、出力を取得します。 エージェントを構成して、セッションを開いたままにすることも、起動するたびにセッションを再確立することもできます。 セッションを開いたままにすると、起動のたびにスクリプトを再使用またはアップロードできます。 デフォルトでは、単一の SSH セッションが使用され、起動のたびにスクリプトが再使用されます。

Agent Builder は、RSA (Rivest, Shamir, and Adleman) またはデジタル署名アルゴリズム (DSA) の鍵による SSH プロトコル・バージョン 2 のみの使用をサポートしています。エージェントは、 ユーザー名およびパスワードで認証されるか、 公開鍵認証で認証されます。公開鍵の生成および配布は、エージェントおよび Agent Builder の外部で実行しなければならない管理用タスクです。

セキュア・シェル (SSH) が有効なスクリプト・データ・プロバイダー向けに作成されたアクション実行コマンドを、リモート・システム上で実行する方法については、1505 ページの『SSHEXEC アクション』を参照してください。

制約事項 : バージョン 6.3 より前の Agent Builder でビルドされたエージェントに SSH を使用するスクリプト・データ・プロバイダーがある場合、IBM Tivoli Monitoring バージョン 6.3 以降とともに実行すると、そのプロバイダーでは障害が発生します。この問題を解決するには、現行バージョンの Agent Builder を使用してエージェントを再ビルドしてください。

この制約事項があるのは、IBM Tivoli Monitoring バージョン 6.3 では新規バージョンの Global Secure ToolKit (GSKit) API が使用されているためです。 IBM Tivoli Monitoring バージョン 6.3 以降で実行するには、Agent Builder 6.3 以降を使用してエージェントを再ビルドする必要があります。Agent Builder 6.3 でエージェントをビルドした場合、以前のバージョンの IBM Tivoli Monitoring でも実行できます。

スクリプト構文解析および分離文字

特定のスクリプト分離文字を変更すること、および 1 つ以上の属性に割り当てることができます。

スクリプト属性グループを作成すると、単一文字のテキスト分離文字がデフォルトで割り当てられます。デフォルトの分離文字は ";" です。分離文字は、エージェントがデータ行の各属性のデータを構文解析し、区切るために使用されます。デフォルトの分離文字を変更して、別の文字を使用できます。特定の分離文字を 1 つ以上の個々の属性に割り当てることもできます。

以下の機能を持つ特定の分離文字を個々の属性に割り当てることができます。

- 出力から固定のバイト数を取得します。
- カスタム分離文字を使用して、1 つの属性を次の属性から分離します。複数の文字から成るカスタム分離文字を使用できます。
- 値の先頭および末尾に挿入されるストリングを使用して、属性値を区切ります。
- テキストの残りの部分を (組み込み分離文字が含まれるかどうかとは無関係に) 属性値として返します。

これらの分離文字のうちの 1 つ以上を使用して、データ行から属性値を抽出できます。

例 1 - 単純なスクリプト出力

一部のスクリプトは、明確かつ通常の分離文字を含むデータ行を出力することがあります。以下に例を示します。

```
Row One;1;2
Row Two;3;4
Row Three;5;6
```

この例で各行のデータの 3 つの部分を分離する明確かつ通常の分離文字は、";" 文字です。この場合、デフォルトの分離文字で問題ないため、分離文字を変更したり、他の分離文字を定義したりする必要はありません。分離文字が別の文字である類似のスクリプト出力を考えることもできます。以下に例を示します。

```
Row One-1-2
Row Two-3-4
Row Three-5-6
```

この例では、分離文字が ";" 文字から "-" 文字に変更されています。 この場合には、属性を定義するときに、デフォルトの分離文字を変更して、"-" 文字を使用します。

例 2 - 複雑なスクリプト出力

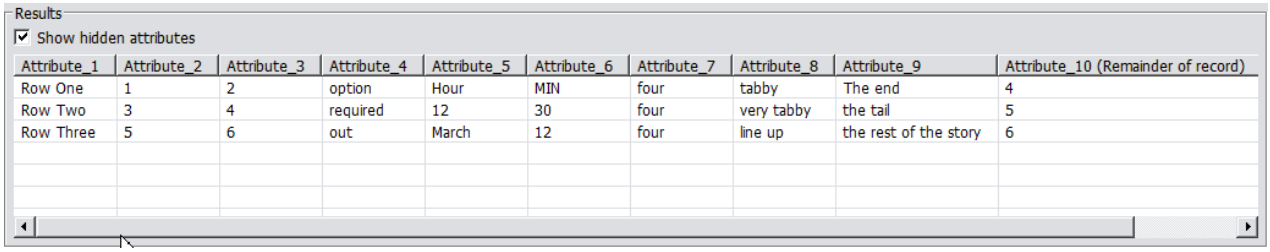
スクリプトは、不規則または可変的な分離文字を含むデータ行を出力することがあります。以下に例を示します。

```
Row One;1;2;[option]Hour:MIN;fourtabby     The end;4
Row Two;3;4;[required]12:30;fourvery tabby the tail;5
Row Three;5;6;[out]March:12;fourline up    the rest of the story;6
```

この例では、属性定義に対して、以下の分離文字を割り当てることができます。

1. まず、各データ行の最初の 3 つの属性では、デフォルトの分離文字 ";" を使用します。この場合には、各属性を定義するときに、";" に設定された「**分離文字テキスト**」という分離文字タイプを割り当てます。この設定はデフォルトです。
2. 4 番目の属性では、"[" と "]" の間のストリングの値を抽出すると仮定します。この場合には、4 番目の属性を定義するときに、開始テキスト値と終了テキスト値が "[" と "]" に設定された「**開始テキストと終了テキスト**」という分離文字タイプを割り当てます。
3. 5 番目の属性では、"]" 文字と ":" 文字の間の値を抽出すると仮定します。 この場合には、5 番目の属性を定義するときに、":" に設定された「**分離文字テキスト**」という分離文字タイプを割り当てます。
4. 6 番目の属性では、デフォルトの分離文字 ";" を再び使用するので、デフォルトを受け入れます。
5. 7 番目の属性では、次の 4 文字のストリング "four" を抽出します。このストリングの末尾には、明示的な分離文字がありません。文字数を割り当てて、次の属性からの分離を定義できます。「**文字数**」という分離文字タイプを割り当て、4 文字を長さとして指定します。
6. 8 番目の属性では、ストリング tabby、very tabby、および line up を抽出します。この場合には、これらのどのストリングの後にもタブ文字が続くと仮定できます。この場合には、「**タブ分離文字**」という分離文字タイプを割り当てます。
7. 9 番目の属性では、デフォルトの分離文字タイプに再び戻って、テキストの残りの部分をこの属性に抽出します。
8. 10 番目の属性では、「**レコードの残り**」を指定して、データ行の残りの部分をこの属性に割り当てます。

上記の例で示されたデータ行を出力するスクリプトでこれらの分離文字を定義した場合の出力は以下のとおりです。


Results

Show hidden attributes

Attribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5	Attribute_6	Attribute_7	Attribute_8	Attribute_9	Attribute_10 (Remainder of record)
Row One	1	2	option	Hour	MIN	four	tabby	The end	4
Row Two	3	4	required	12	30	four	very tabby	the tail	5
Row Three	5	6	out	March	12	four	line up	the rest of the story	6

図 *43.* エージェントが複雑なスクリプト出力を構文解析するときの属性値出力の例

属性分離文字を定義する手順については、1282 ページの『スクリプトの出力のモニターのステップ』のステップ 1284 ページの『10』 で説明されています。

スクリプトの出力のモニターのステップ

スクリプト・データ・ソースからデータを受け取るようにエージェントを構成します。

始める前に

1280 ページの『スクリプトからのアウトプットのモニター』を参照してください。

このタスクについて

スクリプトからのアウトプットをモニターするには、以下の手順を実行します。

手順

1. 「**エージェントの初期データ・ソース**」ページ または「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域のオプション「**コマンドまたはスクリプト**」を選択します。
2. 「**データ・ソース**」領域で「**スクリプトからのアウトプット**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**コマンド・リスト**」ページで、「**追加**」をクリックして「**コマンド情報**」ウィンドウを表示します。

   注 : 「**SSH を使用したデータ収集を使用可能にする**」チェック・ボックスを選択すると、この属性グループに対して SSH が有効になります。このチェック・ボックスを選択しないと、属性グループはローカルで実行されます。

注 : Agent Builder が稼働しているオペレーティング・システムで実行可能なコマンドが存在する場合は、「**テスト**」オプションが使用可能です。「**テスト**」を使用して、定義したコマンドをテストできます。

5. 「**コマンド情報**」ウィンドウの「**コマンド情報**」域で、「**コマンド**」フィールドにコマンド名と必要な引数を入力し、「**分離文字**」フィールドに分離文字を入力します。

   注 :

   a. Windows のスクリプトは、しばしば、コマンド行で `.bat` または `.cmd` 拡張子を指定せずに開始されることがあります。リモート実行の場合、シェル環境がインストールされており、スクリプトを実行するためにスクリプト・データ・ソース・コマンドに `.bat` または `.cmd` を指定する必要があります。Windows 、Linux、Red Hat、および AIX で使用可能なシェル環境の 1 つに `Cygwin` があります。シェル環境が存在することを確認するには、リモート・ホストに SSH 接続またはログオンし、次のコマンドを入力します。

   ```
   PATH=$PATH:. <command>
   ```

   このコマンドが実行される場合は、シェル環境が存在しています。

   b. コマンド・インタープリターによる 構文解析が行われないように、名前は引用符で囲んでください。例えば、`this is a test.bat argument` は次のようになります。

   ```
   "this is a test.bat" 引数
   ```

   c. 環境変数および構成変数は、ユーザー提供のスクリプトの中で使用できますが、そのスクリプトを開始するコマンド行の一部として使用することはできません。以下の変数は、この規則の例外です。

   AGENT_BIN_DIR
   エージェントがバイナリー・ファイルまたはスクリプトを入れるディレクトリー

   AGENT_ETC_DIR
   エージェントが構成ファイルを入れるディレクトリー

   AGENT_LIB_DIR
   エージェントが共有ライブラリーまたはダイナミック・リンク・ライブラリーを入れるディレクトリー

   CANDLEHOME
   Linux または UNIX Tivoli Monitoring のインストール・ディレクトリー

   CANDLE_HOME
   Windows Tivoli Monitoring のインストール・ディレクトリー

   d. SSH データ収集オプションを使用する場合、コマンド行は、リモート・システム上のユーザーのホーム・ディレクトリーに対する相対ディレクトリーで実行されます。スクリプトまたは実行可能プログラムをリモート・システムにアップロードする 場合は、エージェントの環境変数 *CDP_SSH_TEMP_DIRECTORY* に指定されているロケーションにこれらのスクリプトまたはプログラムがコピーされます。このロケーションは、デフォルトでは、リモート・システム上のユーザーのホーム・ディレクトリーです。一部のシステムでは、 コマンド行を相対パス (`./Script.sh` など) で定義する 必要があります。

6. 「**オペレーティング・システム**」域で、1 つ以上のオペレーティング・システムを選択します。SSH を使用してリモート・システムからデータを収集する場合、「オペレーティング・システム」は、エージェントのインストール先システムのプロパティーです。これは、リモート・システムのオペレーティング・システムではありません。SSH データ収集機能を使用する場合は、「**すべてのオペレーティング・システム**」チェック・ボックスを選択することをお勧めします。

7. オプション: コマンドを実行するために 1 つ以上のユーザー定義ファイルが必要な場合は、「コマンド・ファイル」領域の「**追加**」をクリックして、システムのファイルを指定します。

   このファイルは、`scripts/`*`operating system`* の下にあるエージェントのプロジェクト・フォルダーにコピーされます。ここで、*operating system* は、「**コマンド情報**」ウィンドウで選択した内容に応じた変数です。これらのファイルは、エージェントを使用してパッケージ化され、配布されます。既に追加したコマンド・ファイルの定義、または内容を変更したコマンド・ファイルの定義を編集する

には、ファイルを選択し、「**編集**」をクリックします。1279 ページの『コマンド・ファイル定義の編集』を参照してください。

8. 「**OK**」をクリックします。 「**コマンド・リスト**」ページが表示されます。
9. コマンドをテストするには、以下の手順を実行します。
   a) 「**テスト**」をクリックして、コマンド情報を開き「**コマンドのテスト**」ウィンドウ を表示します。リモート・システムでスクリプトをテストするには、「**接続名**」リストからシステムを 選択するか、「**追加**」をクリックしてシステムのホスト 名を追加します。
   b) 「**コマンドのテスト**」ウィンドウを使用してコマンド、デフォルト分離文字、および属性分離文字を変更し、返されるデータにこれらの変更がどのように影響するのかを調べます。
      1) コマンドと分離文字がまだフィールドに入力されていない場合は、それらを入力します。

         注 : 属性の作成時に「**属性情報**」ウィンドウ を使用するか、既存の属性を変更するために Agent Editor を使用して、他の分離文字を指定できます。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。データ・ソースおよび属性の操作について詳しくは、1192 ページの『データ・ソースおよび属性のプロパティーの編集』を参照してください。
      2) テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』 を参照してください。
      3) 「**OK**」をクリックして、「**テスト設定**」ウィンドウに戻ります。
      4) 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
      5) Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。Agent Builder はコマンドを実行します。リモート・システムを指定した場合は、 ユーザー ID およびパスワードを指定します。 戻りコードが 0 でない場合でも、Agent Builder はエージェントの場合と同じ方法でコマンドの結果を解析します。
      6) 「**テスト設定**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。属性の最初の名前は、**Attribute_1、Attribute_2** のように付けられますが、該当する列見出しをクリックすることで属性のプロパティーを変更できます。
      7) 「**結果の確認**」をクリックすると、コマンドの戻りコード、解析されていないデータ、およびエラー・メッセージが返された場合はそれらがすべて表示されます。
      8) 「**エージェントの停止**」をクリックすると、エージェントは停止します。
      9) 「**OK**」をクリックして、「**コマンド情報**」ウィンドウに戻ります。

         コマンドまたは分離文字を変更する場合は、該当するコマンドが更新されて、それらの変更が反映されます。

         スクリプト・データ・ソースの作成時にこのウィンドウが開いた場合は、新規のスクリプト・データ・ソースに属性が追加されます。

         既存のスクリプト・データ・ソースからこのウィンドウが開いた場合は、属性に対して行ったすべての変更はスクリプト・データ・ソースに対しても行われます。すべての追加属性が追加され、どの追加属性も削除されません。これらのオプションは、スクリプト出力から解析される属性に対してのみ影響します。 派生属性には影響しません。派生属性のいずれかが、その参照属性に基づいて無効になる場合、これらの派生属性を手動で更新または削除できます。表示されるのは派生属性の式であり、 実際の結果値ではありません。

   注 : 属性グループが存在する場合に、テストを開始するには、以下の手順を実行します。

   a. **Agent Editor** の「**データ・ソース定義**」ページで属性グループを選択します。
   b. テスト対象のスクリプトをコマンド・リストから選択します。
   c. 「**テスト**」をクリックし、ステップ 1284 ページの『9』 の手順を実行します。
10. ステップ 1284 ページの『9』 でコマンドのテストをスキップした場合は、以下の手順を実行します。
    a) 入力したコマンド情報を含む「**コマンド・リスト**」ページで、「**次へ**」をクリックします。

b) 1197 ページの表 261 を使用して、「**属性情報**」ページで属性名およびタイプ情報を指定します。「**追加属性の追加**」を選択して、その他の属性を追加します。

c) 「**属性情報**」ページで、「**スクリプト属性情報**」タブを使用して、この属性の特定のデータ分離文字を選択します。

デフォルトでは、標準分離文字 **;** が選択されます。ストリング、文字数、タブ、スペースなどの他のさまざまな分離文字を選択できます。また、データの先頭と末尾で異なるストリング分離文字を使用することも選択できます。最後に、「**レコードの残り**」を選択して、レコードの残りの部分を属性に割り当てることもできます。スクリプト構文解析および分離文字について詳しくは、1281 ページの『スクリプト構文解析および分離文字』を参照してください。

11. 以下の手順のいずれかを行います。
    - **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    - データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。
12. 属性を追加して、それらの属性に情報を指定することができます。詳しくは、1194 ページの『属性の作成』を参照してください。

スクリプト・データ・ソースの「**データ・ソース定義**」ページには、すべてのデータ・ソースに適用できるフィールド (1197 ページの『属性の定義に関するフィールドおよびオプション』を参照) に加えて、以下のオプションがあります。

コマンド・リスト
データ収集中に開始するコマンドおよびスクリプトにアクセスできるようにします。

追加
この属性グループが開始するコマンドをユーザーが追加できます。

編集
既存のコマンド項目をユーザーが編集できます。

削除
既存のコマンド項目をユーザーが削除できます。

テスト
ユーザーがこの属性グループのテスト環境に アクセスできます。

SSH を使用したデータ収集を使用可能にする
このチェック・ボックスを選択すると、この属性グループに対して SSH が有効になります。 このチェック・ボックスを選択しないと、属性グループはローカルで実行されます。

スクリプト・データ・ソースの SSH リモート接続構成については、1366 ページの『Secure Shell (SSH) リモート接続の構成』を参照してください。

Java Database Connectivity (JDBC) のデータのモニター

データ・ソースを定義して、JDBC データベースからデータを受信することができます。エージェントは、SQL 照会を実行して、データベースからデータを収集します。照会によって返される各列は、結果のデータ・セット内の 1 つの属性です。

このタスクについて

JDBC データ・プロバイダーは、 以下のデータベース・サーバーをサポートしています。

- IBM DB2 9.x および 8.x
- Microsoft SQL Server 2008、2005、および 2000
- Oracle Database 11g および 10g

Agent Builder には、これらのデータベースのための JDBC ドライバーは 組み込まれていません。JDBC ドライバーは、データベースに対する JDBC 接続を 確立するために必要な、ベンダーによって提供される一連の JAR ファイルです。便宜のために、 これらのドライバーをダウンロードできるサイトへのリンクを示します。

- IBM DB2: JDBC ドライバーは、データベース・サーバーのインストール環境に含まれており、メイン DB2 インストール・ディレクトリーの下の `java` という名前のサブディレクトリーにあります。

- Microsoft SQL Server の Web サイト (www.microsoft.com)
- Oracle データベース: Oracle データベース JDBC (http://www.oracle.com/technetwork/database/features/jdbc/index.html)

注 : 重要なことは、JDBC データ・プロバイダーがデータベース・サーバーをリモート側からモニターできることです。Java ランタイム環境と、接続先データベース・サーバー用の JDBC ドライバーの JAR ファイルを、エージェントを実行するシステム上に用意しておく必要があります。

以下の Java のバージョンをサポートします。

- Oracle Corporation Java バージョン 5 以降
- IBM Corporation Java バージョン 5 以降

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**JDBC**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**JDBC 情報**」ページの「**JDBC 情報**」領域で「**参照**」をクリックし、データベースに接続して SQL 照会を作成します。

   JDBC ブラウザーを使用してデータベースに接続し、そのデータベースの表を参照することにより、必要なデータを収集する SQL 照会を作成することができます。表と列を選択すると、照会が生成され、この照会によって返される各列に属性が追加されます。生成された照会を変更してテストし、 必要なデータが返されることを確認できます。

   注 : 「**参照**」をクリックせずに 手動で JDBC データ・ソースを作成することもできます。 手動でデータ・ソースを作成する場合は、照会を指定して 「**次へ**」をクリックします。照会によって返される各列の属性を、それらの列が返される順序で定義する必要があります。

   JDBC データ・プロバイダーにより、SQL 照会およびストアード・プロシージャーをデータベースに対して実行し、 モニター・データを収集できます。データを収集するための SQL 照会を指定するときには、SQL ステートメントに where 節を含めて、 返されるデータをフィルターに掛けることができます。SQL ステートメントでは、 複数の表からのデータを結合することもできます。JDBC データ・プロバイダーは、SQL の select ステートメントのほかにストアード・プロシージャーを 実行することもできます。ストアード・プロシージャーの実行について詳しくは、1291 ページの『ストアード・プロシージャー』を参照してください。
5. 最初にブラウザーを開くと、接続が選択されていないというメッセージが Java Database Connectivity (JDBC) ブラウザー・ウィンドウに表示されます。接続を追加する必要があります。「**追加**」をクリックして、接続を追加するための手順を実行します。

   既に接続を定義している場合は、その接続が使用されます。その場合は、ステップ 1287 ページの『6』 に進むことができます。

   注 : 「**状況**」フィールドに、 現在の接続の状況が表示されます。

   接続を追加するには、以下の手順を実行します。

   a) 「**JDBC 接続**」ページで、「**JDBC 接続**」をクリックして「**次へ**」をクリックします。

   b) 「**接続プロパティー**」ページで、以下のように、各フィールドに情報を入力します。

   接続名
   JDBC 接続の名前。この接続の固有の名前を入力します。 この名前は、ブラウザーで接続を参照するために使用します。

   データベース・タイプ
   データベースのタイプ。接続先のデータベース製品を選択します。例えば、IBM DB2 データベースに接続するには、「**DB2**」を選択します。

   ユーザー名
   データベースに対して少なくとも読み取り権限を持つように定義する必要がありますが、 データベース管理者である必要はありません。

パスワード
データベースに対して少なくとも読み取り権限を持つように定義する必要がありますが、 データベース管理者である必要はありません。

ホスト名
データベース・サーバーが稼働しているホスト名。JDBC では、 リモート・データベースをモニターできます。 ローカル・システムでのデータベースのモニターには制限されません。

ポート
データベース・サーバーが listen するホスト名のポート。

データベース
接続先データベースの名前。

Jar ディレクトリー
データベースへの接続に使用する JDBC JAR ファイルが存在するディレクトリー。パス名を入力するか、「**参照**」をクリックしてディレクトリーの 位置を指定します。

c) オプション: この接続のパスワードを保存する場合は、「**パスワードを Agent Builder ワークスペースに保存**」チェック・ボックスを選択します。

d) オプション: このアプリケーション・サーバー・タイプのデフォルトをこれらのプロパティーからコピーする場合は、「**エージェント構成のデフォルトとして設定**」チェック・ボックスを選択します。

モニター対象システムと類似したシステム 上にエージェントを作成する場合は、このボックスを選択することをお勧めします。このボックスを選択しなかった場合、エージェントを構成するユーザーに対して空のフィールドが表示されます。その場合、このユーザーは、デフォルト値を使用せずに、すべての情報の値を入力する必要があります。

e)「**接続のテスト**」をクリックして、指定した構成パラメーターを使用するデータベースへの接続を作成します。

「**接続プロパティー**」ページのメッセージに、 接続が成功したかどうかが示されます。

f) 接続が機能している場合は、 「**完了**」をクリックします。

6.「**Java Database Connectivity (JDBC) ブラウザー**」ウィンドウで、構成されているデータベースへの接続が行われます。データベースに含まれる表が 「**データベース表**」領域に表示されます。データベース表を選択すると、 その表に含まれる列が「**選択した表の列**」領域に表示されます。

注 :

a. 双眼鏡アイコンをクリックして、「**データベース表**」リストで表を検索します。

b. デフォルトでは、すべての表が表示されます。別のフィルター・オプションを選択することによって、 表示させる表をフィルターに掛けることができます。使用可能なフィルター・オプションを 1287 ページの表 267 に示します。

表 *267.* フィルター・オプション

フィルター・オプション	**説明**
すべて	すべての表を表示します。
ユーザー	ユーザー表のみを表示します。
システム	システム 表のみを表示します。
表示	データベース・ビューのみを表示します。

注 : 特定の列を取得する場合は、それらの列のみを選択してください。表を選択すると、Agent Builder が、その表からすべての列を収集する照会を自動的に作成し、現在その表にあるすべての列の属性を作成します。

列を選択するには、以下のようにします。

- すべての列のデフォルト照会を取得するには、表を選択します。
- 列のみを取得するには、その列を選択します。

7. オプション: 「**属性情報**」ページで、「エラー」、「欠落データ」、および「値なし」に設定されている列挙値を変更します。

   データベース表の列から返される可能性がある正当な値とのオーバーラップを回避するために、値を変更します。

8. オプション: SQL ステートメントのテストおよび変更を行うには、「**Java Database Connectivity (JDBC) ブラウザー**」ウィンドウで「**テスト**」をクリックします。

   「**SQL ステートメントの実行**」ウィンドウが開きます。

   a) 「**SQL ステートメント**」フィールドで、SQL ステートメントを入力または変更します。

   b) SQL ステートメントを実行するには、「**実行**」をクリックします。

   結果が「**結果**」領域に表示されます。必要なデータが返されるようになるまで、ステートメントの変更とテストを続けます。

   c) 「**OK**」をクリックすることで、ステートメントの保存および適切な属性の作成を行い、「**JDBC 情報**」ウィンドウに戻ります。

9. オプション: 「**JDBC 情報**」ウィンドウで「**テスト**」をクリックすると、より現実に即したエージェント環境でこの属性グループをテストすることができます。JDBC 属性グループのテストについて詳しくは、1292 ページの『JDBC 属性グループのテスト』を参照してください。このテスト時に JDBC ステートメントを変更する場合、JDBC ステートメントによって列ごとに 1 つの属性が正しい順序で返されるように、属性を調整する必要もあります。

10. オプション: 「**拡張**」をクリックして、この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。

11. 「**JDBC 情報**」ページの「**オペレーティング・システム**」セクションでオペレーティング・システムを選択し、「**次へ**」をクリックします。選択するオペレーティング・システムについて詳しくは、1213 ページの『オペレーティング・システムの指定』を参照してください。

    注 : 挿入するプロパティーを選択するには、 「**構成プロパティーの挿入**」をクリックします。詳しくは、1360 ページの『エージェント構成のカスタマイズ』 を参照してください。

12. 「**キー属性の選択**」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』を参照してください。

13. 以前に定義したデータ・ソースをテストする 場合は、Agent Editor ウィンドウで 「**データ・ソース**」タブを選択し、JDBC データ・ソースを選択します。「**JDBC 属性グループ情報**」領域で 「**テスト**」をクリックします。テストについて詳しくは、1292 ページの『JDBC 属性グループのテスト』を参照してください。

14. 自動的に生成された構成セクションを表示する場合は、Agent Editor の「**構成プロパティーの挿入**」タブをクリックします。

    エージェントの初期構成時にユーザーに対して表示されるデフォルトに合わせて、これらのプロパティーのラベルまたはデフォルト値を変更できます。

15. オプション: 「**属性情報**」ページに入力します。詳細については、1197 ページの『属性の定義に関するフィールドおよびオプション』を参照してください。このステップは、ステップ 1286 ページの『4』 で「参照」をクリックせずに JDBC データ・ソースを手動で作成することを選択した場合に実行します。

    Agent Builder の JDBC データ・ソースは、ほとんどの SQL タイプからのデータ収集をサポートしています。1288 ページの表 268 では、JDBC ブラウザーがこれらのいずれかのタイプの列を検出したときに作成する属性のタイプについて説明します。これらのデータ・タイプは、モニター・エージェントでの使用がサポートされています。

表 *268.* モニター・エージェントでの使用がサポートされる *SQL* データ・タイプ

SQL データ・タイプ	**作成される IBM Tivoli Monitoring 属性**
BIGINT	このデータ・タイプは、IBM Tivoli Monitoring の 64 ビットのゲージ値です。IBM Tivoli Monitoring V6.2 との互換性を選択した場合は、32 ビットのゲージになります。

表 *268.* モニター・エージェントでの使用がサポートされる *SQL* データ・タイプ *(*続き*)*	
SQL データ・タイプ	**作成される IBM Tivoli Monitoring 属性**
DECIMALDOUBLEFLOATNUMERICREAL	これらの SQL タイプは、IBM Tivoli Monitoring の 64 ビットのゲージ属性として作成されます。データベースのメタデータにスケール値が含まれている場合は、その値が使用されます。それ以外の場合、スケールは 1 に設定されます。IBM Tivoli Monitoring V6.2 との互換性を選択した場合、属性は 32 ビットのゲージになります。
BITINTEGERSMALLINTTINYINT	以下の SQL タイプは、IBM Tivoli Monitoring の 32 ビットのゲージ属性として作成されます。
BOOLEAN	この値は、TRUE と FALSE の列挙値を持つ、IBM Tivoli Monitoring の 32 ビットのゲージです。
TIMESTAMP	このタイプの列にあるデータは、16 バイトの IBM Tivoli Monitoring タイム・スタンプ属性に変換されます。
TIMEDATECHARLONGVARCHARVARCHAR	これらの SQL タイプはいずれも、ブラウザーによってストリング属性と見なされます。属性のサイズとして列サイズが使用されます。 最大値は 256 であり、これは JDBC ブラウザーのストリング属性のデフォルト・サイズです。

注 : リストされていないデータ・タイプからデータを収集する場合は、 デフォルトでストリング属性が使用されます。また、エージェントは、 データベースからストリングとしてデータを収集しようとします。

「**属性情報**」ページで、「エラー」、「欠落データ」、および「値なし」に設定されている列挙値を変更します (必要な場合)。データベース表の列から返される可能性がある正当な値とのオーバーラップを回避するために、値を変更します。

JDBC 構成

エージェント内に JDBC データ・ソースを定義すると、一部の構成プロパティーが作成されます。

エージェント内に JDBC データ・ソースを定義した場合、そのエージェントは Java を使用して JDBC データベース・サーバーに接続する必要があります。Java 構成プロパティーは自動的にエージェントに追加されます。以下の Java 構成プロパティーは、エージェントのランタイム構成に固有のものです。

- *Java* ホーム: Java のインストール・ディレクトリーを指す完全修飾パス
- *JVM* 引数: このパラメーターを使用して、Java 仮想マシンの引数のリストを指定します (任意指定)。
- トレース・レベル: このパラメーターは、Java トレース・ログ・ファイルに書き込む情報の量を定義します。デフォルトでは、エラー・データだけがログ・ファイルに書き込まれます。

注 : Agent Builder は、JLog プラグインによって構成される独自の JVM とロギングを使用するため、Java プロパティーを必要としません。

エージェント内に JDBC データ・ソースを定義した場合は、以下に示す共通の必須構成フィールドがエージェントに自動的に追加されます。

- *JDBC* データベース・タイプ: 接続先データベースのタイプ (IBM DB2、Microsoft SQL Server、または Oracle Database Server)。
- *JDBC* ユーザー名: データベース・サーバーでの認証に使用するユーザー名。
- *JDBC* パスワード: データベース・サーバーでの認証に使用するパスワード。
- 基本パス:「クラスパス」フィールドで指定した JAR ファイルを見つけるために検索するディレクトリー、または「*JAR* ディレクトリー」フィールドで指定したディレクトリーのリスト。これは完全修飾でな

いディレクトリーです。 Windows では、ディレクトリー名を区切るためにセミコロン (;) を使用し、UNIX システムでは、セミコロン (;) またはコロン (:) を使用します。

- クラスパス: エージェントによって検索する JAR ファイルを明示的に指定したパス。完全修飾でないファイルは、JAR ファイルが見つかるまで、それぞれの基本パスに追加されます。
- *JAR* ディレクトリー: JAR ファイルを見つけるために検索するディレクトリーのリスト。Windows では、ディレクトリー名を区切るためにセミコロン (;) を使用し、UNIX システムでは、セミコロン (;) またはコロン (:) を使用します。これらのディレクトリーに格納されている JAR ファイルを明示的に指定する必要はありません。いずれかのディレクトリーに格納されていれば検出されます。それらのディレクトリーのサブディレクトリーは検索対象になりません。完全修飾でないディレクトリーは、ディレクトリーが見つかるまで、それぞれの基本パスに追加されます。

ランタイム構成についても、 データベースに接続するための追加の詳細を指定する必要があります。残りの構成項目を、JDBC URL と基本的な構成プロパティー (デフォルト) のいずれとして指定するかを選択できます。

- URL 構成オプション
  - JDBC connection URL: データベースが存在するホストの詳細情報を提供するベンダー固有の接続 URL および接続先ポート番号。通常、URL の形式は以下のようになります。

    ```
    jdbc:identifier://server:port/database
    ```

    その他の URL の形式については、JDBC ドライバーのベンダーの資料を参照してください。
- JDBC 基本プロパティー・オプション (デフォルト)

  JDBC サーバー名: データベース・サーバーが稼働しているホストの名前。
  JDBC データベース名: 接続を実行するホストのデータベースの名前。
  JDBC ポート番号: データベース・サーバーが listen するポートの番号。

注 : JDBC データ・プロバイダーにより、サブノードを使用して同一のエージェントで複数のデータベース・タイプをモニターできます。 この方法でモニターするには、サブノード構成のオーバーライドを慎重に定義する必要があります。複数のデータベース・タイプをモニターする場合は、 以下の構成設定が変わると考えられます。

- JDBC データベース・タイプ
- JDBC ユーザー名
- JDBC パスワード

基本的な構成オプションを使用する場合は、「**サブノード構成のオーバーライド**」ページで以下のプロパティーのオーバーライドも 定義する必要があります。

- JDBC サーバー名
- JDBC ポート番号
- JDBC データベース名

サブノードに対する構成オーバーライドを定義するには、「**サブノード構成のオーバーライド**」ページにアクセスするための詳細について、1344 ページの『サブノードの使用』を参照してください。実行時にエージェントを構成する場合は、 新規に作成する各サブノード・インスタンスについて、これらすべてのプロパティーを構成する必要があります。

構成のオーバーライドのほかに、サブノードから接続する各データベース・タイプの JDBC ドライバーをエージェントが指すようにする必要もあります。*JAR* ディレクトリー・パラメーターは、JDBC ドライバーを指すための最も便利な方法です。その場合は、JDBC ドライバーが格納されているディレクトリーを、それぞれセミコロンで区切ってリストします。例えば、エージェントを使用して DB2 データベースと Oracle データベースに接続する場合は、`C:¥Program Files¥IBM¥SQLLIB¥java;C:¥oracle¥jdbc` などの *JAR* ディレクトリーの値を指定する必要があります。

ストアード・プロシージャー

JDBC データ・プロバイダーと併用可能なサンプルの SQL と DB2 ストアード・プロシージャー。

JDBC データ・プロバイダーは、ストアード・プロシージャーから戻される結果セットを処理できます。ストリングまたは整数入力パラメーターをストアード・プロシージャーに渡すことができます。以下の構文によりストアード・プロシージャーが実行されます。

```
call[:index] procedureName [argument] ...
```

各項目の意味は次のとおりです。

index
どの結果セットがデータ・プロバイダーによって使用されるかを指定するオプションの整数。このパラメーターは、ストアード・プロシージャーが複数の結果セットを返し、それらの結果セットのいずれか 1 つからの値だけを収集する必要がある場合に便利です。index が指定されていない場合は、それぞれの結果セットのデータが収集されて返されます。

procedureName
JDBC データ・プロバイダーによって実行されるストアード・プロシージャーの名前。

argument
ストアード・プロシージャーへの入力引数。複数の引数はスペースで分離する必要があります。引数にスペース文字が含まれている場合は、その引数全体を二重引用符で囲みます。引数を整数として構文解析できる場合、その引数は整数引数としてストアード・プロシージャーに渡されます。二重引用符で囲まれた引数は、すべてストリング引数として渡されます。

SQL Server サンプル

call sp_helpdb

引数が不要なプロシージャー call sp_helpdb を実行します。 データ・プロバイダーにより返されるデータには、返されたすべての結果セットのデータが含まれます。

call:2 sp_helpdb master

引数 master を指定したプロシージャー sp_helpdb を実行します。この引数は、ストリング入力引数です。ストアード・プロシージャーから返される 2 番目の結果セットのデータのみが、データ・プロバイダーが返すデータに組み込まれます。

索引が指定されていない場合は、返されたすべての結果セットのデータが収集されます。このような状況で返されるデータに、定義する属性との互換性があることを確認してください。Agent Builder では、返された 1 番目の結果セットから属性が作成されます。返されるその他の結果セットには、1 番目の結果セットとの互換性があることが期待されています。

DB2 ストアード・プロシージャー

以下は、SQL で書かれた DB2 関数のサンプルです。この関数では、Agent Builder JDBC データ・プロバイダーにより処理可能な結果を返す方法を示します。

```
-- Run this script as follows:
-- db2 -td# -vf db2sample.sql

-- Procedure to demonstrate how to return a query from
-- a DB2 stored procedure, which can then be used by
-- an Agent Builder JDBC provider. The stored procedure
-- returns the following columns:
-- Name               Description              Data Type
-- current_timestamp  The current system time  timestamp
-- lock_timeout       The lock timeout         numeric scale 0
-- user               The user for the session String 128 characters long
DROP procedure db2sample#

CREATE PROCEDURE db2sample()
  RESULT SETS 1
  LANGUAGE SQL
BEGIN ATOMIC

  -- Define the SQL for the query
  DECLARE c1 CURSOR WITH HOLD WITH RETURN FOR
  SELECT CURRENT TIMESTAMP as current_timestamp,
```

```
CURRENT LOCK TIMEOUT as lock_timeout, CURRENT USER as user
  FROM sysibm.sysdummy1;

  -- Issue the query and return the data
  OPEN c1;
END#
```

この関数は、その他のストアード・プロシージャーに対して定義されている同じ構文を使用して Agent Builder から呼び出すことができます。この場合、このストアード・プロシージャーを実行する JDBC ステートメントとして `call db2sample` を定義します。

Oracle ストアード・プロシージャー

Oracle ストアード・プロシージャーは結果セットを返しません。代わりに、Oracle 参照カーソルを返す関数を作成する必要があります。以下の PL/SQL で記述された Oracle 関数の例に、Agent Builder JDBC データ・プロバイダーによる処理が可能な結果を返す方法を示します。

```
CREATE OR REPLACE FUNCTION ITMTEST
RETURN SYS_REFCURSOR
IS
        v_rc SYS_REFCURSOR;
BEGIN
  OPEN v_rc FOR SELECT * FROM ALL_CLUSTERS;
  RETURN v_rc;
END;
```

この関数は、その他のストアード・プロシージャーに対して定義されている同じ構文を使用して Agent Builder から呼び出すことができます。この場合、このストアード・プロシージャーを実行する JDBC ステートメントとして `call ITMTEST` を定義します。Oracle 関数はカーソル参照を返す必要があるため、Oracle 関数により処理できる結果セットは 1 つのみです。つまり、複数の結果セットを返されることがないため、Oracle については索引オプションがサポートされていません。

JDBC 属性グループのテスト

Agent Builder 内で作成した JDBC 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**JDBC 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**JDBC ステートメントのテスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数、構成プロパティー、および Java 情報を設定できます。

   詳しくは、1376 ページの『属性グループのテスト』を参照してください。JDBC 構成プロパティーについて詳しくは、1289 ページの『JDBC 構成』を参照してください。
3. 「**エージェントの開始**」をクリックします。

   ウィンドウに、エージェントが開始されていることが表示されます。
4. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。

   エージェントは、指定された SQL 照会を使用してデータベースを照会します。「**JDBC ステートメントのテスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。

   注 : 返されたデータの順序は重要です。例えば、最初に返された列のデータ値は、常に最初の属性に割り当てられます。JDBC ステートメントを変更する場合は、その JDBC ステートメントによって返される列と一致するように、属性の追加、削除、または再配列を行う必要があります。
5. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。

6. 「**エージェントの停止**」をクリックしてエージェントを停止します。
7. 「**OK**」または「**キャンセル**」をクリックして「**JDBC ステートメントのテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

ping を使用したシステム可用性のモニター

データ・ソースを定義して、Internet Control Message Protocol (ICMP) エコーの ping を使用したネットワーク・デバイスのリストのテストを実行できます。テストするデバイスのホスト名または IP アドレスは、1 つ以上のデバイス・リスト・ファイルにリストされます。別個の ping 構成ファイルで、各デバイス・リスト・ファイルへのパスを指定します。次に、ping 構成ファイルの名前がエージェントのランタイム構成に設定されます。結果には、各ネットワーク・デバイスの状況が含まれます。

始める前に

デバイス・リスト・ファイルと ping 構成ファイルを作成します (1294 ページの『構成ファイル』を参照)。

このタスクについて

ネットワーク管理の一環として、システムが Internet Control Message Protocol (ICMP) ping に応答しているかどうかを判別できる 必要があります。このデータ・ソースを使用して、ご使用の環境内の一連のサーバーや他の重要なデバイスの基本的なオンラインまたはオフラインの状況をモニターします。 ping によるモニターは単純で、オーバーヘッドも少なく済みます。デバイスのリストをモニターするには、エージェントに ping データ・コレクターを追加します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**ネットワーク管理データ**」をクリックします。
2. 「**データ・ソース**」領域で、「**ping**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**ping 情報**」ウィンドウの 「**オペレーティング・システム**」域で、 オペレーティング・システムを選択します。
5. オプション: 「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1295 ページの『ping 属性グループのテスト』を参照してください
6. オプション: 「**拡張**」をクリックして、 この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
7. 以下の手順のいずれかを行います。

   a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。

   b) データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。
8. 属性の追加について詳しくは、1194 ページの『属性の作成』を参照してください。

タスクの結果

ping の属性グループについて詳しくは、1450 ページの『ping 属性グループ』を参照してください。

構成ファイル
構成ファイルを使用することで、エージェントに対して ping するデバイスのリストを指定します。

エージェントは、2 つのタイプの構成ファイルを必要とします。

デバイス・リスト・ファイル
ping するデバイスのリストが含まれます。多くのデバイスがある場合、複数のデバイス・リスト・ファイルに分割できます。エージェントは、デバイス・リスト・ファイルごとに別個のスレッドを開始し、ファイルを並行して循環処理します。60 秒と、30 秒にリストの ping にかかる時間を加えた時間のいずれか長い方ごとに各ファイルを循環処理します。

デバイス・リスト・ファイルの構文は、次のとおりです。

```
LISTNAME=list_name
device_name or host_name
device_name or host_name
device_name or host_name device_name or host_name
```

ここで、*list_name* は、そのファイル内のデバイスの説明です。リスト名が定義されていない場合は、デバイス・リスト・ファイルの名前が使用されます。リスト名が、ファイル内の最初の項目である必要はありません。ただし、ファイルに複数のリスト名定義がある場合、最後の定義が使用されます。

デバイス・リスト・ファイルに含めることができるデバイスの数に制限はありません。ただし、あまり多くの項目を記述すると、 重要なデバイスを対象としたリストを作成するという目的が損なわれるほか、 全体のワークロードが増加してしまいます。60 秒のモニター間隔以内に各デバイスの状況を取得するのがより困難になる可能性もあります。

各周期の開始時に、エージェントは、デバイス・リスト・ファイルの最終変更時刻を検査します。このファイルの最終変更時刻が、エージェントがこのファイルを読み取った最終時刻よりも後の場合、エージェントは再始動することなく、このファイルを再読み取りします。

ping 構成ファイル
各デバイス・リスト・ファイルの場所を指定します。完全修飾パスか、ping 構成ファイルの場所への相対パスを使用します。ping 構成ファイルは、ランタイム構成パラメーターとしてエージェントに渡されます。

例

次の例では、デバイスが 2 つのファイルに分割されます。
`/data/retailList.txt` ファイルには、次の項目が格納されます。

```
LISTNAME=Retail
frontend.mycompany.com
productdb.mycompany.com
```

`/data/manufacturingList.txt` ファイルには、次の項目が格納されます。

```
LISTNAME=Manufacturing systems
manufloor.mycompany.com
stats.supplier.com
```

ping ファイルである `/data/pinglists.txt` には、次の項目が格納されます。

```
/data/retailList.txt
/data/manufacturingList.txt
```

ネットワーク管理の構成プロパティー

ping データ・ソースが追加されると、Agent Editor の「**ランタイム構成情報**」ページに構成が表示されます。

「**ランタイム構成情報**」ページの「**ネットワーク管理**」構成セクションには、以下のプロパティーがあります。

表 269. ネットワーク管理の構成プロパティー			
Name	**有効な値**	**必須**	**説明**
ping 構成ファイル	ファイルのパス	いいえ。このファイルを指定しなかった場合は、エージェントの bin ディレクトリーにある KUMSLIST ファイルが使用されます。	ファイルのリストを含むファイルのパス。リストする各ファイルには、ICMP ping を使用してモニターするホストのリストを記述します。

ping 属性グループのテスト
Agent Builder 内で作成した ping 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**ping 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**設定のテスト**」ウィンドウが開きます。
2. オプション: テストを開始する前に、環境変数および構成プロパティーを設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。
3. 「**参照**」をクリックして、ping 構成ファイルを選択します。ping 構成ファイルについて詳しくは、1294 ページの『構成ファイル』を参照してください。
4. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
5. モニター環境からのエージェント・データの要求をシミュレートするには、「**データの収集**」をクリックします。エージェントは、ping 構成ファイルから参照されているデバイス・リスト・ファイルで指定されたデバイスを ping します。
6. 「**テスト設定**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
7. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。
   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
8. 「**エージェントの停止**」をクリックしてエージェントを停止します。
9. 「**OK**」または「**キャンセル**」をクリックして「**設定のテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念
1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

HTTP 可用性および応答時間のモニター

選択された URL の可用性と応答時間をモニターするようにデータ・ソースを構成できます。構成ファイルを使用して、URL のリストを定義します。エージェント・ランタイム構成内でそのファイルの名前を設定します。IBM Tivoli Monitoring では、アクション実行コマンドを使用して、モニター対象の URL を追加および削除することもできます。各 URL の状況は、結果のデータ・セット内に 1 つの行として追加されます。

このタスクについて

モニターする URL ごとに、結果には HTTP 要求に対する HTTP 応答に関する一般情報が示されます。結果には、取得の可否、取得にかかった時間、および応答のサイズが示されます。応答コンテンツが HTML の場合、URL 内のページ・オブジェクトに関する情報も示されます。

HTTP、HTTPS、FTP、およびファイル・プロトコルを使用する URL をモニターできます。 モニターする URL を HTTP URL ファイルに指定するか、またはアクション実行オプションを使用してモニターする URL を指定します。

重要 : 現行リリースでは、アクション実行コマンドは IBM Cloud Application Performance Management 環境で使用できません。Tivoli Monitoring 環境でのみ使用できます。

このデータ・ソースを使用するには、Java ランタイム環境が必要です。以下の Java のバージョンをサポートします。

- Oracle Corporation Java バージョン 5 以降
- IBM Corporation Java バージョン 5 以降

一連の URL をモニターするために属性グループを作成するには、以下の手順を実行します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。
2. 「**データ・ソース**」領域で、「**HTTP**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**HTTP 情報**」ページの「**オペレーティング・システム**」域で 1 つ以上のオペレーティング・システムを選択します。
5. オプション: 「**テスト**」をクリックして、この属性グループをテストします。テストについて詳しくは、1303 ページの『HTTP 属性グループのテスト』を参照してください
6. オプション: 「**拡張**」をクリックして、この属性グループから返されるデータを制限するフィルターを作成します。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。
7. 以下の手順のいずれかを行います。
   a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
   b) データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

タスクの結果

HTTP データ・ソースにより、2 つの属性グループ (「管理対象 URL」および「URL オブジェクト」) が作成されます。属性を追加、変更、または削除できます。

関連タスク

1194 ページの『属性の作成』
データ・セットに新規属性を追加することができます。

関連資料

1453 ページの『HTTP 属性グループ』
「管理対象 URL」および「URL オブジェクト」という 2 つの HTTP 属性グループは、URL からの情報と、これらの URL 内のオブジェクトを受け取るために使用されます。

HTTP テーブル

デフォルトの HTTP 属性グループに関する参照情報。

HTTP データ・ソースによって、以下の 2 つの属性グループが作成されます。

管理対象 URL

「管理対象 URL」テーブルには、各モニター対象 URL の可用性および応答時間のデータが示されます。

URL オブジェクト

「URL オブジェクト」テーブルには、組み込みオブジェクトごとに個別の URL エントリーが含まれています。例えば、「管理対象 URL」レポートにリストされる Web サイトで使用される可能性がある .gif ファイルおよび .jpg ファイルなどです。

「管理対象 URL」テーブルおよび「URL オブジェクト」テーブルで使用される構文について詳しくは、1297 ページの『HTTP 属性に固有のフィールド』を参照してください。

Web サイト内にある特定のオブジェクトの応答時間と可用性をモニターする場合は、「URL オブジェクト」テーブルの内容を確認してください。「URL オブジェクト」テーブルは、ダウンロードされた HTML ファイルで検出された特定のオブジェクトのリストをモニターします。モニター対象オブジェクトに対応して検索される HTML 要素と、このオブジェクトを参照するこれらの要素内の属性を以下の表にリストします。

表 *270.* モニター対象オブジェクトに対応して検索される *HTML* 要素

HTML 要素	**モニター対象オブジェクトが含まれている属性**
img	src
script	src
embed	src
object	codebase または data
body	background
input	src

以下の HTML 抽出例では、モニター対象オブジェクトは、img エレメントの src 属性により参照される画像です。

```
<img src="/v8/images/id-w3-sitemark-simple.gif" alt="" width="54" height="33" />
```

この画像の完全な URL は、ソース文書の URL に基づいて計算されます。

注 : Web ページで検出されたオブジェクトをモニターしない場合は、「URL モニター」構成セクションで「**ページ・オブジェクト収集**」プロパティーを「**いいえ**」に設定します。

HTTP 属性に固有のフィールド

「**属性情報**」ページには、URL からのデータ収集方法を定義する HTTP 属性の 2 つのフィールドがあります。「**属性タイプ**」フィールドでは、返される URL に関する情報を制御する任意の値をリストから選択できます。一部の属性タイプを使用する場合は、「**タイプ値**」フィールドに値を指定する必要があります。

「管理対象 URL」属性グループのすべての属性タイプとタイプ値 (必要な場合) の説明を以下の表に示します。

表 *271. HTTP* 属性情報 - 管理対象 *URL*

属性タイプ	説明	タイプ値	返されるデータ・タイプ	FTP およびファイル・プロトコルでの違い
XPath 照会	URL 接続から返されるコンテンツに対して XPath 照会を実行します。ノード・リストではなく、属性に有用なデータを返すようにこの照会を記述する必要があります。	URL 接続から取得するコンテンツに対して実行する XPath 照会。	返されるデータは、ストリング、数値、またはタイム・スタンプ値です。データが XML 日時フォーマットの場合、属性タイプにタイム・スタンプを指定することができます。エージェントがその値を Candle タイム・スタンプに変換します。	なし
応答時間	要求された URL からのコンテンツのダウンロードにかかった時間 (ミリ秒)。	なし	整数 (ミリ秒数)	なし
応答メッセージ	サーバーから返される HTTP 応答メッセージ。	なし	ストリング	応答メッセージは、URL で HTTP または HTTPS プロトコルが使用される場合にのみ適用されます。
応答コード	サーバーから返される応答コード。	なし	整数	応答コードは、URL で HTTP または HTTPS プロトコルが使用される場合にのみ適用されます。ファイルまたは FTP の URL の場合は常に 0 です。
応答の長さ	要求された URL からダウンロードされるコンテンツのサイズ (バイト単位)。	なし	整数 (バイト単位のサイズ)	なし
応答ヘッダー	応答ヘッダーを使用して、URL 応答ヘッダーのいずれかのフィールドの値を取得できます。引数には、要求するフィールドを指定します。	収集対象の応答ヘッダー。	ストリング	一般に FTP およびファイル・プロトコルでは、収集できるヘッダーはありません。

表 271. *HTTP* 属性情報 - 管理対象 *URL (続き)*				
属性タイプ	**説明**	**タイプ値**	**返されるデータ・タイプ**	**FTP およびファイル・プロトコルでの違い**
要求 URL	接続先 URL。すべての応答キーワードによって、この URL への接続に関する情報が提供されます。XPath 照会を使用して、この URL にアクセスした結果返されるコンテンツから取得された情報を取得することができます。	なし	ストリング	なし
ページ・オブジェクト	「URL オブジェクト」属性グループによりモニターされる HTML ページでディスカバーされたオブジェクトの数。	なし	整数	なし
オブジェクト・サイズの合計	この Web ページの「URL オブジェクト」属性グループでモニターされたオブジェクトの合計サイズ。	なし	整数 (バイト単位)	なし
別名	この URL に対してユーザーが指定した別名。	なし	ストリング	なし
ユーザー	この URL に対してユーザーが指定したデータ。	なし	ストリング	なし

「URL オブジェクト」属性グループの属性タイプを以下の表で説明します。

表 272. *HTTP* 属性情報 - *URL* オブジェクト				
属性タイプ	**説明**	**タイプ値**	**返されるデータ・タイプ**	**FTP およびファイル・プロトコルでの違い**
URL	「管理対象 URL」テーブルでモニターされる URL。	なし	ストリング	なし
オブジェクト名	**HTML** ページ内でモニターされるオブジェクトの URL。	なし	ストリング	なし

表 272. HTTP 属性情報 - URL オブジェクト (続き)				
属性タイプ	**説明**	**タイプ値**	**返されるデータ・タイプ**	**FTP およびファイル・プロトコルでの違い**
オブジェクト・サイズ	「オブジェクト名」の URL からダウンロードされたコンテンツのサイズ (バイト単位)。	なし	数値	なし
オブジェクト応答時間	ページ・オブジェクトのダウンロードにかかった時間 (ミリ秒)。	なし	数値	なし

URL のモニター

URL のモニターを開始するには、URL ファイルにその URL を追加するか、または「HTTP URL Add」アクション実行オプションを使用します。

URL ファイル

構成で指定する URL ファイルは、任意のディレクトリーに格納できます。このファイルが存在しないかまたは空の場合、アクション実行を使用して URL モニターを開始できます。詳しくは、1300 ページの『アクション実行オプション』を参照してください。Tivoli Universal Agent HTTP データ・プロバイダーを使用する Tivoli Universal Agent を既に使用している場合は、KUMPURLS ファイルを再利用できます。エージェントを構成する際に、KUMPURLS ファイルを指し示します。

URL の追加方法に応じた URL ファイルでの URL 入力例を以下の表に示します。

表 273. URL ファイルのエントリー	
URL	**追加方法**

`www.bbc.co.uk`
`http://weather.com`
`www.ibm.com`
	ファイルに手動でエントリーを追加。プロトコルが指定されていない場合 (例: `www.ibm.com`)、http が想定されます。

`ftp://userid:password@ftpserver/index.html`	ファイル転送プロトコル (FTP) を使用して手動で追加
`http://www.ibm.com USER=ibm ALIAS=ibm`	「HTTP URL Add」アクション実行を使用
`file:/tmp/samples.html USER=samples ¥` `ALIAS=samples`	FTP を使用する「HTTP URL Add」アクション実行を使用
`http://google.com INTERVAL=60 CACHE=50 ¥` `USER=google ALIAS=search`	Tivoli Universal Agent KUMPURLS ファイルの例

URL ファイルを直接編集する場合、エージェントが次回のデータ収集を実行する際に、変更内容が実装されます。

アクション実行オプション

「HTTP URL Add」アクション実行オプションを使用して、モニターする URL を指定することもできます。

制約事項 : このオプションは、現行リリースの IBM Cloud Application Performance Management で使用することはできません。これは、アクション実行コマンドを手動で開始できないためです。

このオプションを選択すると、以下のパラメーターを指定できるウィンドウが表示されます。

URL
URL 自体を表す必須パラメーター。このパラメーターを入力するときには、接頭部 `http://` または `https://` を指定してもしなくてもかまいません。

別名
URL により分かりやすい名前を関連付けるために指定できるオプション・パラメーター。このパラメーターには、スペースを使用することはできません。 このパラメーターが完全に入力されていない場合は、別名はデフォルトのブランクになります。

ユーザー・データ
URL に関するデータを入力するために指定できるオプション・パラメーター。このパラメーターが完全に入力されていない場合、**ユーザー・データ**はデフォルトの `INITCNFG` になります。

情報を入力してウィンドウを閉じた後に、エージェントに関連付けられている宛先管理対象システムに「`HTTP URL Add`」アクションを割り当てます。新規 URL のモニターが即時に開始されます。この URL は URL ファイルにも追加されるため、エージェントの再始動後も継続してモニターされます。

対応するアクション実行オプションの名前は「`HTTP URL Remove`」です。「`HTTP URL Remove`」アクションは、特定の URL のモニターをただちに停止するために使用します。削除された URL は、URL ファイルからも削除されます。「**HTTP URL Remove**」ウィンドウは、「URL」値と「**ユーザー・データ**」値のみを要求します。「URL」および「**ユーザー・データ**」の値は、Tivoli Enterprise Portal に表示される値と一致している必要があります。一致していない場合、「Remove」アクションは失敗します。例えば、「Add」アクションの「URL」フィールドで `http://` を省略した場合は、「Remove」アクションの「URL」フィールドにはこれを含める必要があります。「**ユーザー・データ**」を指定しなかった場合は、Tivoli Enterprise Portal で表示される `INITCNFG` を指定する必要があります。

URL を URL ファイルに手動で追加した場合は、アクション実行を使用してこの URL を削除できます。アクション実行を使用してこの URL を削除する場合は、Tivoli Enterprise Portal で表示されるとおりにこの値を指定する必要があります。例えば URL ファイルに `www.ibm.com` を追加した場合、Tivoli Enterprise Portal では、URL として `http://www.ibm.com` が表示され、**ユーザー・データ**として `INITCNFG` が表示されます。アクション実行を使用してこの URL を削除するには、Tivoli Enterprise Portal で表示される値を使用する必要があります。

情報の入力を完了してウィンドウを閉じた後に、エージェントに関連付けられている宛先管理対象システムに「`HTTP URL Remove`」アクションを割り当てます。

`https://` URL のモニター
HTTP データ・ソースは、スクリプトによるアクセスまたは対話式プロンプトが不要なセキュア `https://` のみをモニターできます。

標準の HTTP Get 呼び出しを使用して取得できる `https://` をモニターできます。

プロキシー・サーバー
エージェントが稼働しているシステムで、SOAP データ・プロバイダーにアクセスするためにプロキシーが必要な場合は、プロキシー・サーバー構成プロパティーを指定する必要があります。

詳しくは、1302 ページの『プロキシー・サーバーの構成』を参照してください。

HTTP 構成
HTTP 構成に関する参照情報。

HTTP データ・ソースが追加されると、Agent Editor の「**ランタイム構成**」ページに構成が表示されます。URL モニター、プロキシー・サーバー認証、および Java の構成セクションが追加されました。

URL モニターの構成

『URL モニターの構成』セクションには、以下のプロパティーがあります。

表 *274. URL* モニターの構成プロパティー			
Name	**有効な値**	**必須**	**説明**
HTTP URL ファイル	ファイルのパス	はい	URL のリストが記述されているファイルのパス。
ページ・オブジェクト収集	はい、いいえ デフォルト値は「はい」です。	いいえ	Web ページで検出されたオブジェクトをダウンロードし、これらのオブジェクトからデータを収集するかどうか。

プロキシー・サーバーの構成

『プロキシー・サーバーの構成』セクションには、以下のプロパティーがあります。

表 *275.* プロキシー・サーバーの構成プロパティー			
Name	**有効な値**	**必須**	**説明**
プロキシー・ホスト名	ストリング	いいえ	HTTP 接続に使用されるプロキシー・ホスト名。
プロキシー・ユーザー名	ストリング	いいえ	プロキシー・サーバーのユーザー名。
プロキシー・ポート	正整数 デフォルト値は 80 です。	いいえ	プロキシー・サーバーの HTTP ポート番号。
プロキシー・パスワード	パスワード	いいえ	プロキシー・サーバーのパスワード。

注 :「**プロキシー・ホスト名**」プロパティーがブランクの場合は、プロキシーは使用されません。

Java 構成

エージェント内に HTTP データ・ソースを定義した場合、そのエージェントは Java を使用して HTTP サーバーに接続する必要があります。Java 構成プロパティーは自動的にエージェントに追加されます。以下の Java 構成プロパティーは、エージェントのランタイム構成に固有のものです。 Agent Builder は、JLog プラグインによって構成される独自の JVM とロギングを使用するので、Java プロパティーを必要としません。

表 *276. Java* 構成プロパティー			
Name	**有効な値**	**必須**	**説明**
Java ホーム	ディレクトリーの完全修飾パス	いいえ	Java のインストール・ディレクトリーを指す完全修飾パス。
トレース・レベル	選択項目 (デフォルト値は「エラー」)	はい	このプロパティーを使用して、Java プロバイダーで使用されるトレース・レベルを指定します。
JVM 引数	ストリング	いいえ	このプロパティーを使用して、Java 仮想マシンの引数のリストを指定します (任意指定)。

HTTP 属性グループのテスト

Agent Builder 内で作成した HTTP 属性グループをテストできます。

手順

1. テスト・プロシージャーは以下の方法で始動します。
   - エージェントの作成中に、「**HTTP 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**HTTP テスト**」ウィンドウが表示されます。
2. 「**参照**」をクリックして、HTTP URL ファイルを選択します。URL ファイルについて詳しくは、1300 ページの『URL ファイル』を参照してください。
3. オプション: テストを開始する前に、環境変数、構成プロパティー、および Java 情報を設定します。

   詳しくは、1376 ページの『属性グループのテスト』を参照してください。HTTP の構成について詳しくは、1301 ページの『HTTP 構成』を参照してください。
4. 「**エージェントの開始**」をクリックします。

   ウィンドウに、エージェントが開始されていることが表示されます。
5. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。

   エージェントは、HTTP URL ファイルで定義された URL をモニターします。「**HTTP テスト**」ウィンドウは、返されたデータをすべて表示します。
6. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
7. 「**エージェントの停止**」をクリックしてエージェントを停止します。
8. 「**OK**」または「**キャンセル**」をクリックして「**HTTP テスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

SOAP データ・ソースまたは他の HTTP データ・ソースのデータのモニター

データ・ソースを定義して、(例えば SOAP プロトコルを使用して) HTTP サーバーからデータを受信することができます。データ・ソースは、URL に HTTP 要求を送信し、応答 (XML 形式、HTML 形式、または JSON 形式) を結果のデータ・セットの属性として解析します。要求から取得されるデータを選択できます。

このタスクについて

SOAP データ・ソースを使用すると、HTTP URL を指定し、GET、POST、または PUT 要求を送信できます。POST または PUT 要求の場合、関連する POST データを指定できます。XML、HTML、または JSON 応答が取得および解析され、データが属性に含まれるモニター環境に公開されます。特定の要素内のすべての値として属性を定義できます。または、カスタム XPath 値を定義して、個々の属性にデータを取り込む方法を指定できます。また、この 2 つのメカニズムを組み合わせて使用することもできます。

URL から XML 応答、HTML 応答、または JSON 応答を収集して解析するには、以下の手順を実行します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**サーバーからのデータ**」をクリックします。

2. 「**データ・ソース**」領域で、「**SOAP**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**SOAP 情報**」ページで URL を入力します。
   デフォルト値は以下のとおりです。

   ```
   http://${KQZ_HTTP_SERVER_NAME}:${KQZ_HTTP_PORT_NUMBER}
   ```

   注 : URL に解決される 1 つ以上の構成変数を使用できます。挿入するプロパティーを選択するには、「**構成プロパティーの挿入**」をクリックします。詳しくは、1360 ページの『エージェント構成のカスタマイズ』を参照してください。
5. 要求タイプを選択します。デフォルトの要求タイプは Get です。Post 要求および Put 要求の場合は、処理するデータを入力します。

   注 : Post 要求および Put 要求の場合は、「**構成プロパティーの挿入**」が使用可能になります。処理するデータに構成変数を含めるには、「**構成プロパティーの挿入**」をクリックします。詳しくは、1360 ページの『エージェント構成のカスタマイズ』を参照してください。
6. 「**参照**」をクリックします。

   注 : URL を入力し、要求タイプを選択した後、SOAP ブラウザーを使用して定義を作成しない場合、「**行選択 XPath**」を入力します。「**SOAP 情報**」ウィンドウで「**行選択 XPath**」を入力します。次に、属性グループのすべての属性を定義します。
7. 「**SOAP ブラウザー**」ウィンドウで、以下のステップを実行します。
   a) URL の入力と要求タイプの選択をまだ行っていない場合は、これらの操作を行います。
   b) 「**構成**」をクリックして、URL またはその他のフィールドで参照される構成プロパティーを設定します。
   c) 「**接続**」をクリックして SOAP プロバイダーからデータを取得します。

      URL に接続すると、Document Object Model (DOM) ツリーにこの URL の XML エレメントのリストが表示されます。HTML 応答および JSON 応答は XML に変換され、DOM ツリーとして表示されます。JSON 応答の XML への変換について詳しくは、1307 ページの『JSON データの XML 表現』を参照してください。WebSphere Application Server の例 (1305 ページの図 44) では、以下の URL が入力されています。

      ```
      http://nc053011.tivlab.raleigh.ibm.com:9080/wasPerfTool/servlet
      /perfservlet?module= ¥threadPoolModule
      ```

      `PerformanceMonitor` XML エレメントが表示されます。このエレメントは、SOAP プロバイダーから返される XML 文書の最上位 XML エレメントです。

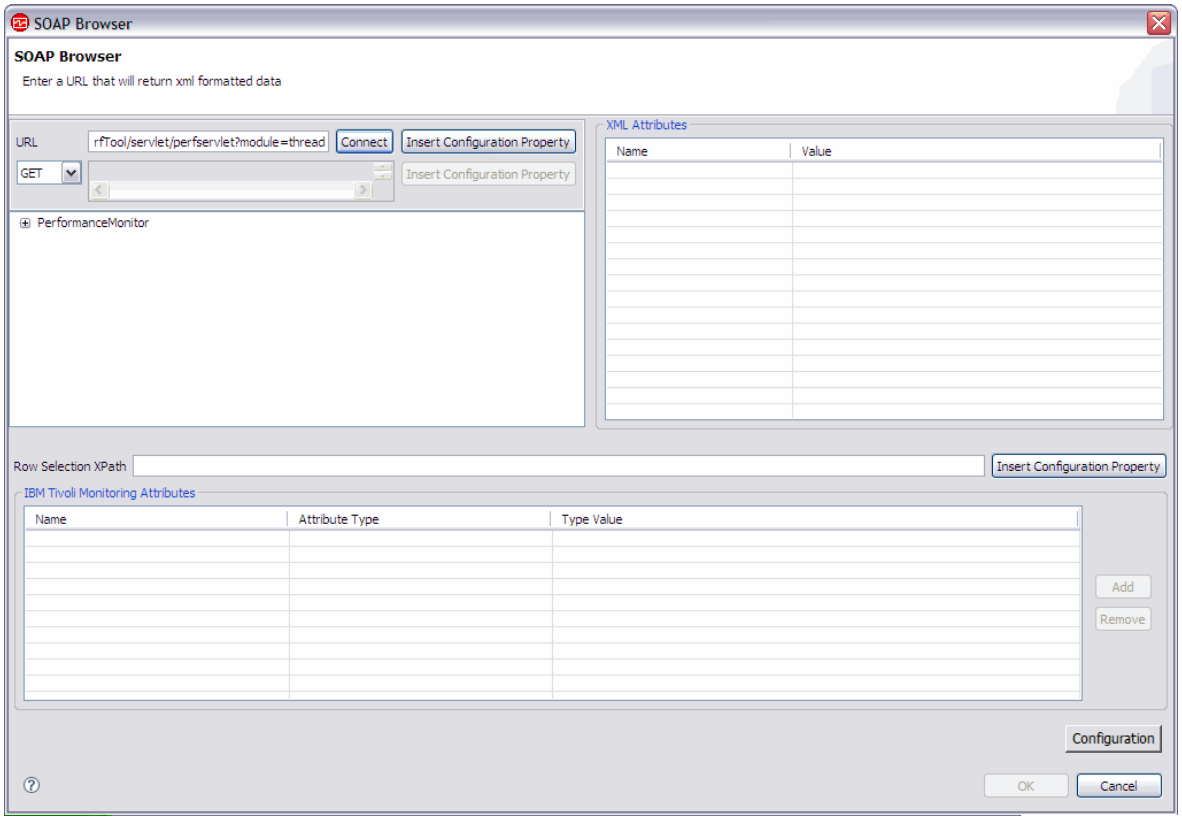


図 *44.* **「*SOAP* ブラウザー」**ウィンドウ

d) DOM ツリーで、**行選択 XPath** として設定する XML ノードを検索して選択します。

WebSphere Application Server の例 (1306 ページの図 45) では、`PerformanceMonitor/Node/Server/Stat/Stat/Stat` ノードが選択されています。 このノードは、属性グループのデータ行を表します。 DOM ツリーでノードを選択し、「**追加**」をクリックすると、ツリーのそのノードで定義されているすべての属性とエレメントが表示されます。(「**エージェント属性**」領域で「**追加**」をクリックします)。

ノードが選択されると、「**XML 属性**」域に、選択されたノードに対して定義されているすべての XML 属性が表示されます。「エージェント属性」のリストに XML 属性を追加するには、その属性を選択して「**追加**」をクリックします。

注 : 複数のデータ行が必要な場合は、XPath がノード・セットにマップされている必要があります。行選択 XPath から 1 つの項目のみが含まれるノード・セットが返される場合、属性グループに含まれる行は 1 行のみです。

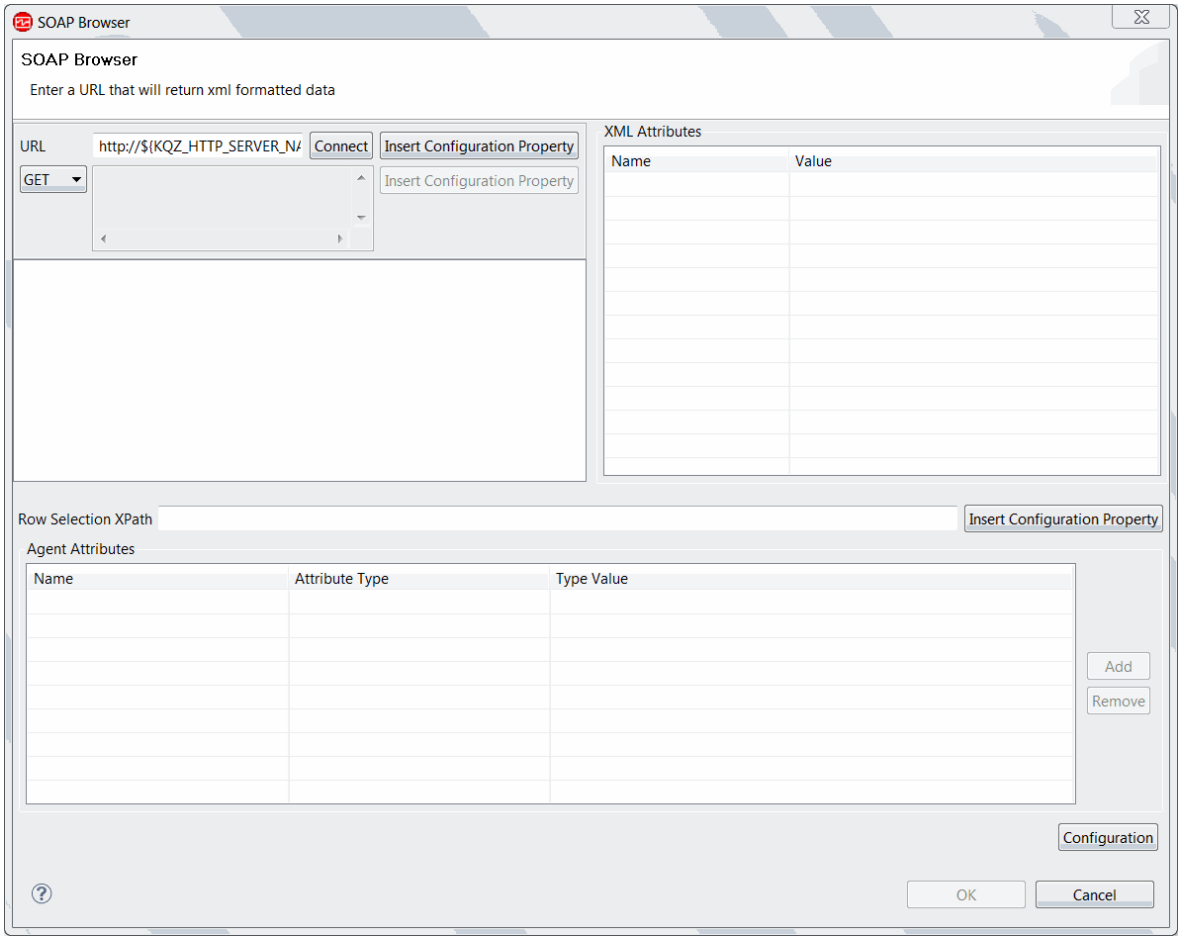


図 *45.* ***「SOAP ブラウザー」*** ウィンドウ

e) 「エージェント属性」領域で「**追加**」をクリックします。

エージェント属性のリストが表示され、「**行選択 XPath**」 フィールドにデータが入力されます。

各エージェント属性の XPath は、XML ノードまたはエレメントをエージェント属性にマップするために使用されます。WebSphere Application Server の例 (1307 ページの図 46) では、エージェントの属性リストの 1 番目の属性 `Stat` は使用されていないため、削除されます。

「**タイプ値**」フィールドで、エージェント属性の名前と XPath を編集できます。XPath の使用法について詳しくは、1309 ページの『XPath オプション』を参照してください。

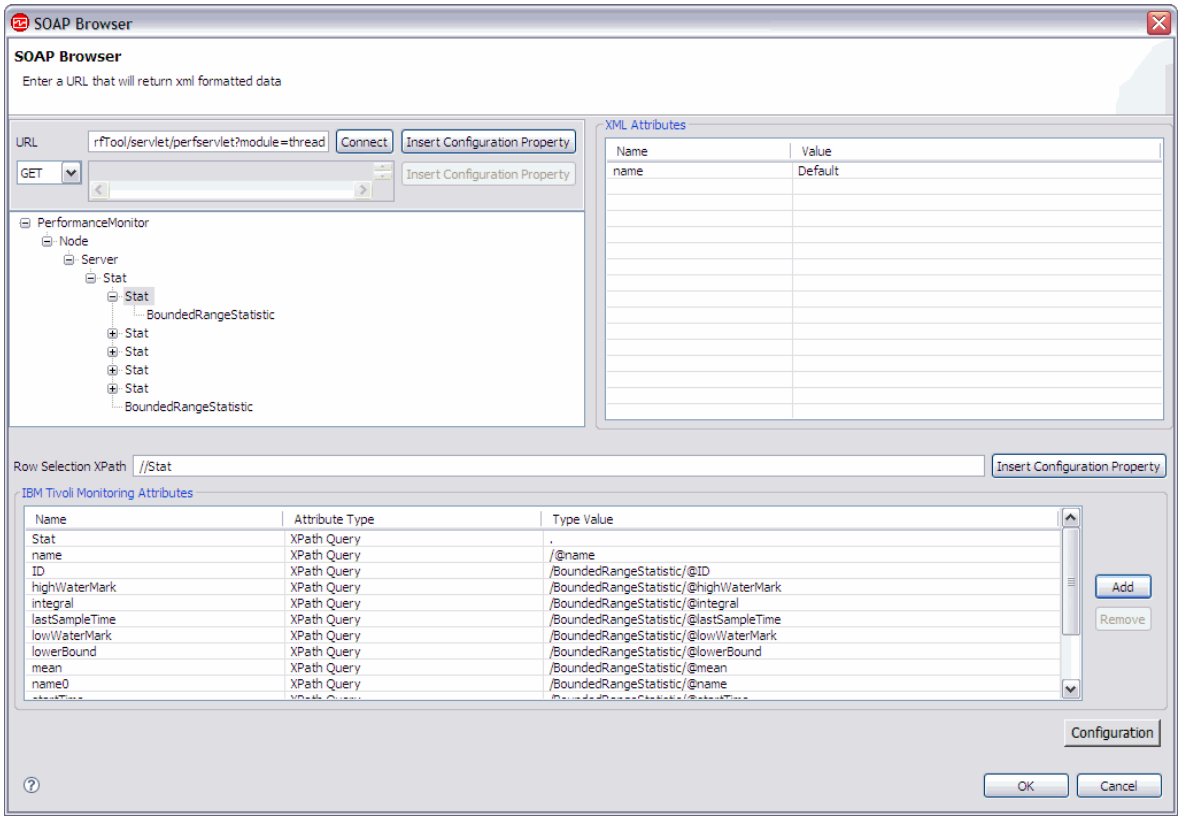


図 *46.* **「*SOAP* ブラウザー」**ウィンドウ

f)「**SOAP ブラウザー**」ウィンドウで「**OK**」をクリックし、変更を保存して「**SOAP 情報**」ウィンドウに戻ります。

8.「**SOAP 情報**」ウィンドウで「**次へ**」をクリックします。

9. 前に「**参照**」を使用せず、「**SOAP 情報**」ウィンドウで **URL** と**行選択 XPath** を入力した場合は、「**属性情報**」ページが表示されます。「**属性情報**」ページで 1 番目の属性に関する情報を指定し、「**完了**」をクリックします。この後、Agent Editor を使用して、その他の属性を指定できます。属性の作成について詳しくは、1194 ページの『属性の作成』を参照してください。

10. ステップ 1304 ページの『6』 で「**参照**」機能を使用した場合は、「**キー属性の選択**」ページが表示されます。「**キー属性の選択**」ページで、キー属性を選択するか、このデータ・ソースが 1 つのデータ行のみを生成することを指定します。詳しくは、1174 ページの『キー属性の選択』を参照してください。

11. オプション:「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1311 ページの『SOAP 属性グループのテスト』を参照してください

12. オプション:「**拡張**」をクリックして、この属性グループから返されるデータを制限するフィルターを作成できます。属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。

13. 以下の手順のいずれかを行います。

    a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。

    b) データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

JSON データの XML 表現

HTTP 要求で JSON データが返される場合、データ・プロバイダーはそのデータを XML に変換します。

データ・プロバイダーは、JSON 属性の名前をエレメント名に変換します。単純タイプの JSON 属性の場合は、値をエレメント内のテキスト・データに変換します。組み込み JSON オブジェクトは、組み込み XML エレメントに変換されます。すべての従属属性は、従属エレメントに変換されます。

ルート XML エレメントは、`JSON_document` です。

JSON 属性名にエレメント名で無効な文字が含まれている場合、データ・プロバイダーはそれを変更して有効なエレメント名を生成します。また、データ・プロバイダーはエレメントに JSON_name 属性も追加します。この属性の値は、元の JSON 属性名です。

データ・プロバイダーは、JSON 配列のエレメントごとに JSON_*xxx*_array_element XML エレメント (*xxx* は配列の名前) を作成します。配列エレメントの値は、XML エレメント内のテキストに変換されます。JSON_index 属性は、各 XML エレメントに追加されます。すなわち、この属性の値は、配列内の配列エレメントのインデックスです。

データ・プロバイダーは、エレメントごとに以下の属性を追加します。

- JSON_level: JSON ファイル内のノードのレベル。JSON_document タグで表されるツリーのルートはレベル 1 です。
- JSON_type: JSON ノードのタイプ (オブジェクト、配列、ストリング、数値)。

SOAP 属性に固有のフィールド

「**属性情報**」ウィンドウには、SOAP 応答からのデータの収集方法を定義する SOAP 属性の 2 つのフィールドがあります。

「**属性タイプ**」フィールドでは、返される応答に関する情報を制御する任意の値をリストから選択できます。一部の属性タイプを使用する場合は、「**タイプ値**」フィールドに値を指定する必要があります。デフォルトの属性タイプは「XPath 照会」です。この属性タイプを使用すると、SOAP サーバー応答コンテンツに対して XPath 照会が実行されます。タイプ値は、実行される XPath 照会です。すべての属性タイプとタイプ値 (必要な場合) の説明を以下の表に示します。

表 *277. SOAP* 属性情報

属性タイプ	説明	タイプ値	返されるデータ・タイプ	FTP およびファイル・プロトコルでの違い
XPath 照会	URL 接続から返されるコンテンツに対して XPath 照会を実行します。ノード・リストではなく、属性に有用なデータを返すようにこの照会を記述する必要があります。	URL 接続から取得するコンテンツに対して実行する XPath 照会。行選択照会が定義されている場合、この XPath 照会は行選択照会に関連している必要があります。	返されるデータは、ストリング、数値、またはタイム・スタンプ値です。一般に、SOAP 用の Agent Builder ブラウザーは、ブラウズされるデータから属性の正しいデータ・タイプを検出します。 データが XML 日時フォーマットの場合、属性タイプにタイム・スタンプを指定することができます。エージェントがその値を Candle タイム・スタンプに変換します。	なし
応答時間	要求された URL からのコンテンツのダウンロードにかかった時間 (ミリ秒)。	なし	整数 (ミリ秒数)	なし

表 277. SOAP 属性情報 (続き)

属性タイプ	説明	タイプ値	返されるデータ・タイプ	FTP およびファイル・プロトコルでの違い
応答メッセージ	サーバーから返される HTTP 応答メッセージ。	なし	ストリング	応答メッセージは、URL で HTTP または HTTPS プロトコルが使用される場合にのみ適用されます。
応答コード	サーバーから返される応答コード。	なし	整数	応答コードは、URL で HTTP または HTTPS プロトコルが使用される場合にのみ適用されます。ファイルまたは FTP の URL の場合は常に 0 です。
応答の長さ	要求された URL からダウンロードされたコンテンツのサイズ (バイト単位)。	なし	整数 (バイト単位のサイズ)	なし
応答ヘッダー	応答ヘッダーを使用して、URL 応答ヘッダーのいずれかのフィールドの値を取得できます。引数には、要求するフィールドを指定します。	収集対象の応答ヘッダー・フィールド。	ストリング	一般に FTP およびファイル・プロトコルでは、収集できるヘッダーはありません。
要求 URL	接続先 URL。すべての応答キーワードによって、この URL への接続に関する情報が提供されます。XPath 照会を使用して、この URL にアクセスした結果返されるコンテンツから取得された情報を取得することができます。	なし	ストリング	なし

XPath オプション

XML パス言語を使用して、XML 文書からノードを選択できます。 SOAP データ・ソースに対する XPath の使用例の一部を以下に示します。

- XPath の述部を使用して、IBM Tivoli Monitoring 属性グループのデータ行に対応する XML エレメントを特定する。 以下の例に示すように、XML エレメントまたは属性を Tivoli Monitoring 属性にマップする XPath の述部を使用できます。

```
Stat[@name="URLs"]/CountStatistic[@name="URIRequestCount"]/@count
```

  XPath に複数のロケーション・ステップがある場合、各ロケーション・ステップで 1 つ以上の述部を使用できます。述部は複雑になることがあります。述部にはブール値または式の演算子を組み込むことができます。以下に例を示します。

```
//PerformanceMonitor/Node/Server[@name="server1"]/Stat/Stat/Stat[@name=
"Servlets"]/Stat
```

- 行に同じタイプの XML エレメントが複数含まれている場合に、XPath にノード設定関数を含めます。ノード・リストでの XML エレメントの位置によって、エレメントがマップされる Tivoli Monitoring 属性が決定します。ノード設定関数は、例えば `position()`、`first()`、`last()`、`count()` です。
- 単純なデータ形式変更 (サブストリングなど) を実行する。以下のサブストリングを指定するとします。

```
substring(myXMLElement,1,3)
```

  この場合、XPath から、XML エレメント myXMLElement の先頭 3 文字が返されます。

行選択 XPath コンテキストの外部のエレメントを指定するには、2 つのピリオド (`..`) を以下の例に示すように使用します。

```
/../OrganizationDescription/OrganizationIdentifier
```

SOAP 構成
SOAP データ・ソースが追加されると、Agent Editor の「**ランタイム構成**」ページに構成が表示されます。

HTTP サーバー、プロキシー・サーバー、および Java の構成セクションが追加されました。プロキシー・サーバーの構成について詳しくは、1302 ページの『プロキシー・サーバーの構成』を参照してください。Java 構成について詳しくは、『1302 ページの『Java 構成』』を参照してください。

HTTP Server

「HTTP サーバー」構成セクションには、以下のプロパティーがあります。

表 *278. HTTP* サーバーの構成プロパティー

Name	**有効な値**	**必須**	**説明**
HTTP ユーザー名	ストリング	いいえ	HTTP ユーザー
HTTP パスワード	パスワード	いいえ	HTTP サーバー・パスワード
HTTP サーバー名	ストリング (デフォルト値は localhost)	いいえ	HTTP サーバーのホストまたは IP アドレス
HTTP ポート番号	数値 (デフォルト値は 80 です)	いいえ	HTTP サーバーのホストまたは IP アドレス
証明書検証が有効	True、False (デフォルト値は True)	はい	証明書検証を無効にする操作は、安全ではない可能性があります。
HTTP トラストストア・ファイル	ファイルのパス	いいえ	HTTP トラストストア・ファイル

表 278. HTTP サーバーの構成プロパティー (続き)			
Name	**有効な値**	**必須**	**説明**
HTTP トラストストア・パスワード	HTTP トラストストア・パスワード	いいえ	HTTP トラストストア・パスワード

プロキシー・サーバー

エージェントが稼働しているシステムで、SOAP データ・プロバイダーにアクセスするためにプロキシーが必要な場合は、プロキシー・サーバー構成プロパティーを指定する必要があります。詳しくは、1302 ページの『プロキシー・サーバーの構成』を参照してください。

SOAP 属性グループのテスト

Agent Builder 内で作成した SOAP 属性グループをテストできます。

手順

1. テスト手順は以下の方法で開始できます。
   - エージェントの作成中に、「**SOAP 情報**」ページで、「**テスト**」をクリックします。
   - エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**SOAP 収集のテスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数、構成プロパティー、および Java 情報を設定できます。詳しくは、1376 ページの『属性グループのテスト』を参照してください。SOAP の構成について詳しくは、1310 ページの『SOAP 構成』を参照してください。
3. URL、行選択 XPath、および要求タイプを変更します。
4. 「**エージェントの開始**」をクリックします。

   ウィンドウに、エージェントが開始されていることが表示されます。
5. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。この操作により「結果」テーブルにデータが取り込まれ、データがどのように解析され、Tivoli Enterprise Portal でどのように列に表示されるかプレビューできます。

   「結果」領域では、属性定義を変更してデータを再ロードし、行った変更が属性グループにどのように影響するか確認できます。「結果」領域の列を右クリックすることで、属性を編集するためのオプションを表示できます。属性の編集オプションは以下のとおりです。
   - **属性の編集**
   - **属性の非表示**
   - **属性を前に挿入**
   - **属性を後ろに挿入**
   - **削除**
   - **後続の属性の削除**
   - **すべて削除**
6. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「**データ収集状況**」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
7. 「**エージェントの停止**」をクリックしてエージェントを停止します。
8. 「**OK**」または「**キャンセル**」をクリックして「**テスト SOAP 収集**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

ソケットを使用したデータのモニター

データ・ソースを定義して、TCP ソケットを使用した外部アプリケーションからのデータの収集を実行できます。アプリケーションは、エージェントへの TCP 接続を開始し、構造化された XML 形式でデータを送信する必要があります。アプリケーションに応じて、データ・ソースは、単一行、複数行、またはイベント・データを含むデータ・セットを生成します。

このタスクについて

ソケット・データ・ソースを使用して、エージェントと同じシステムで稼働する外部アプリケーションからエージェントにデータを提供します。外部アプリケーションは、必要に応じていつでもエージェントにデータを送信できます。外部アプリケーションは、必要に応じていつでもエージェントにデータを送信できます。例えば、コマンド行インターフェースを 作成し、ユーザーがそれを実行して属性グループにデータを送れるようにすることが可能です。別のオプションとして、更新がエージェントに送信されるよう、モニター対象アプリケーションを変更することもできます。 エージェントは、データをソケットに送信するアプリケーションを開始または停止しません。この操作は、ユーザーが制御します。

ソケット・データ・ソースには制限がいくつかあります。

- デフォルトでは、ローカル・ホスト (127.0.0.1) への接続のみ可能です。リモート・ホストからの接続を受け入れるようにエージェントを構成する方法について詳しくは、1320 ページの『リモート・ソケット・ポート接続』を参照してください。
- クライアントのソケット API に、使用可能なサブノードを判別するためのメカニズムがありません。 クライアントは特定のサブノードに対してデータを送信できますが、そのサブノード名を事前に知っておく必要があります。

属性グループを作成し、伝送制御プロトコル (TCP) ソケットを使用してデータを収集するには、以下の手順を実行します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**カスタム・プログラム**」をクリックします。
2. 「**データ・ソース**」領域で、「**ソケット**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**ソケット情報**」ページで、属性グループ名を入力します。
5. 属性グループのヘルプ・テキストを入力します。
6. 属性グループに対して、「**単一データ行の生成**」、「**複数のデータ行を生成可能** 」、または「**イベントの生成**」のいずれかを選択します。詳しくは、1314 ページの『データの送信』を参照してください。
7. 「ソケット情報」セクションで、「**コード・ページ**」を選択します。詳しくは、1318 ページの『文字セット』を参照してください。
8. オプション: 「**拡張**」をクリックして、属性グループの拡張プロパティーを変更します。「**拡張**」オプションは、属性グループに対して「**複数のデータ行を生成可能**」または「**イベントの生成**」を選択した場合にアクティブとなっています。
9. 「**次へ**」をクリックします。
10. 「**属性情報**」ページで、属性グループの最初の属性を指定します。属性の作成について詳しくは、1194 ページの『属性の作成』を参照してください。
11. 「**次へ**」をクリックします。
12. オプション: 「**グローバル・ソケット・データ・ソース情報**」ページの「**エラー・コード**」セクションで、ソケット・クライアントがデータを収集できない場合に送信できるエラー・コードを定義することができます。詳しくは、1316 ページの『データに代わるエラーの送信』 を参照してください。 エラー・コードを定義するには、以下のステップを実行します。

a)「**エラー・コード**」セクションで、「**追加**」をクリックします。エラー・コードは 256 文字までに制限されています。使用できるのは、ASCII 文字、数字、および下線のみです。 スペースは許可されません。

b)「**ソケット・エラー・コードの定義**」ウィンドウで、「**パフォーマンス・オブジェクト状況**」属性グループに表示される表示値を入力します。

c) 内部値を入力してください。内部値は、1,000 から 2,147,483,647 の間の整数でなければなりません。

d) 各エラーのメッセージ・テキストを定義する必要があります。 以前に入力したメッセージ・テキストを、リストから選択して利用することもできます。「**OK**」をクリックして、「**グローバル・ソケット・データ・ソース情報**」ページに戻ります。メッセージ・テキストはエージェント・ログ・ファイルで使用されます。

適切なメッセージ・テキストがない場合、「**参照**」をクリックして、メッセージ・テキストをセットアップします。「メッセージ」(リスト) ウィンドウが開きます。メッセージ・ウィンドウには、エージェントで定義されたメッセージがリストされます。 メッセージを定義するまで、リストはブランクになっています。 「**編集**」を使用すると定義済みメッセージを変更することができ、「**削除**」を使用すると定義済みの 1 つ以上のメッセージを削除することができます。

e)「メッセージ」(リスト) ウィンドウで、「**追加**」をクリックして「**メッセージ定義**」 ウィンドウを表示します。「**メッセージ定義**」ウィンドウで、新規メッセージの意味を示すテキストを入力し、メッセージ・タイプを選択します。

注 : メッセージ ID は自動的に生成されます。

f)「**OK**」をクリックします。

g)「メッセージ」(リスト) ウィンドウが開き、新規メッセージが表示されます。 メッセージを検証して「**グローバル・ソケット・データ・ソース情報**」ページに戻るには、「**OK**」をクリックします。

13. オプション: 「**グローバル・ソケット・データ・ソース情報**」ページの「**補足ファイル**」セクションで、エージェントと一緒にパッケージ化され、エージェントのインストール時にエージェント・システムにコピーされるファイルを追加することができます。 これらのファイルは、エージェントのインストール時にエージェント・システムにコピーされます。

「**ファイル・タイプ**」列は、各ファイルの想定される使用方法を説明します。 次の表では、可能な 3 つの使用方法を説明します。

表 *279.* 補足ファイルのファイル・タイプ

ファイル・タイプ	**説明**
実行可能ファイル	エージェントと一緒に実行可能ファイルを含める場合は、このオプションを選択します。エージェントはこれらのファイルを使用しません。
ライブラリー	エージェントと一緒にライブラリーを含める場合は、このオプションを選択します。 エージェントはこれらのファイルを使用しません。
Java リソース	エージェントと一緒に Java リソースを含めるにはこのオプションを選択します。 エージェントはこれらのファイルを使用しません。

エージェントでの補足ファイルのインストール場所については、1393 ページの『システム 上の新規ファイル』を参照してください。

「**編集**」をクリックして、インポートしたファイルを編集します。詳しくは、1279 ページの『コマンド・ファイル定義の編集』 を参照してください。

14. オプション: 「**テスト**」をクリックすると、この属性グループをテストできます。テストについて詳しくは、1321 ページの『ソケット属性グループのテスト』を参照してください

15. オプション: データ・ソースがサンプル・データである場合は、「**拡張**」をクリックすると、この属性グループから返されるデータを制限するフィルターを作成できます。「**ソケット情報**」ページで「イベントの生成」を選択しなかった場合、データ・ソースはサンプル・データです。属性グループからの

データのフィルタリングについて詳しくは、1202 ページの『属性グループのフィルタリング』を参照してください。

16. 以下の手順のいずれかを行います。
    a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。
    b) データ・ソースを保存して Agent Editor を開く場合は、「**終了**」をクリックしてください。

「**ソケット・プロバイダー設定**」ページの「**オペレーティング・システム**」セクションで、エージェントがソケット・クライアントからのデータを listen するオペレーティング・システムを選択します。 ページを開くには、アウトライン・ビューで「**ソケット・プロバイダー設定**」をクリックするか、任意のソケット属性グループ・ページの Agent Editor で「**グローバル設定**」をクリックします。

注 : エラー・コードおよび補足ファイルの更新は、「**ソケット・プロバイダー設定**」ページの「**エラー・コード**」セクションおよび「**補足ファイル**」セクションで行えます。

エージェントへのソケット情報の送信

エージェントに 1 つ以上のソケット属性グループが含まれる場合、エージェントはソケットを開き、クライアントからのデータを listen します。

ソケット・データをエージェントに送信するアプリケーションは、エージェントで定義されているポートに接続します。このポートは、エージェントの構成プロパティーで設定されている値か、TCP/IP によって自動的に割り振られる一時ポートのいずれかです。ソケット・ポートとその構成について詳しくは、1319 ページの『ソケット構成』を参照してください。

受信するデータは、構造化された XML 形式に従っている必要があります。ソケット・データ・ソースを使用する場合、以下の XML 情報フローが可能です。

- サンプル属性グループに関する 1 行以上のデータをエージェントに送信する
- イベントを生成する 1 つの属性グループに関する 1 行のデータをエージェントに送信する
- データの代わりにエラー・コードをエージェントに送信する
- タスク接頭部の登録をエージェントに送信する
- タスク要求をエージェントから受信する
- タスク応答をエージェントに送信する

データの送信

属性グループは、サンプル・データまたはイベント・データを受信するように定義されます。属性グループを作成するときに、受信するデータにより行われる以下のいずれかの動作を示すオプションを指定します。

- 単一データ行の生成
- 複数のデータ行の生成
- イベントの生成

「**単一データ行の生成**」または「**複数のデータ行を生成可能**」を選択した場合、属性グループはサンプル属性グループです。「**イベントの生成**」を選択した場合、その属性グループは、行を受信するたびにモニター環境にイベントを送信します。

Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソールでサンプル・データを表示する場合、収集した最新の行セットが表示されます。イベント属性グループに関して表示されるデータは、エージェントが管理するローカル・キャッシュの内容です。イベント・データの場合、エージェントはキャッシュに新しい項目を追加し、最も古い項目が削除されるサイズに達するまで追加し続けます。 サンプル・データの場合、ユーザーがデータを送信するたびに、エージェントがキャッシュの内容を置き換えます。

「**イベントの生成**」または「**単一データ行の生成**」を選択した場合は、各メッセージにおいて、当該属性グループに関してエージェントに送信するデータ行は 1 つのみとする必要があります。必要な数のイベントを送信可能であり、それぞれのイベントを個別のメッセージで送信します。

通常、サンプル・データは要求に応じてエージェントが収集しますが、ソケット・クライアントは独自のスケジュールで更新サンプルを提供します。 必要な頻度で、サンプル属性グループ (単一行または複数行) を更新できます。 Tivoli Monitoring または IBM Cloud Application Performance Management によりデータが要求された場合、エージェントは最新のデータを提供します。

Tivoli Enterprise Portal または IBM Cloud Application Performance Management コンソール内でソケット属性グループに関して欠落しているデータ行がある場合は、ログ・ファイルでエラーを確認してください。また、属性グループのデータが想定どおりでない場合も、ログ・ファイルでエラーを確認してください。ソケット・データ・ソースは、入力から可能な限りの処理を行おうとします。 例えば、クライアントが適切な形式の行を 3 行と無効な (誤った形式の XML など) 行を 1 行送信すると、以下を確認できます。

- 属性グループ内の 3 つのデータ行
- エージェントのログ・ファイルに、誤った形式の行に関するエラーが記録されます。
- 有効な行が返されたため、パフォーマンス・オブジェクト状況には NO_ERROR の状況が表示されます。

イベント・データとサンプル・データのどちらの場合も、データはソケット・クライアントから単一の XML データ・フローとしてエージェントに送信されます。 ソケット・クライアントから送信されるデータは、必ず改行文字 (「¥n」) で終了します。エージェントは改行文字を検出するまでデータを読み取った後、受信した内容の処理を試みます。 処理不能なデータを受信すると、そのデータは破棄されます。 以下の例は、abc という名前の属性グループに関する 2 つのデータ行をエージェントに送信する方法を示しています。

```
<socketData><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/></in><in> ¥
<a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>¥n
```

この例では 2 つのデータ行がエージェントに送信され、その各行には 3 つの属性が含まれています。 属性の順序は重要であり、その属性グループで定義された順序に従っている必要があります。唯一の例外は派生属性です。この属性は、属性グループ内での位置に関係なくスキップしなければなりません。

サブノードで属性グループが定義された場合は、エージェントへのデータの送信時にサブノード・インスタンス ID を指定する必要があります。サブノード・インスタンス ID は、socketData エレメントのサブノード属性を使用して指定します。必ず、ソケット・クライアントが使用するサブノード・インスタンス ID の構成に関する規則を採用してください。ソケット・クライアントはインスタンス ID も構成プロパティーも照会できないためです。 構成されていないサブノードに送信されたデータは無視されます。

以下に例を示します。

```
<socketData subnode="app1"><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/>
</in><in> ¥<a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>¥n
```

この例では、インスタンス ID が "app1" のサブノードにデータが送信されます。"app1" は管理対象システム名ではなく、このサブノード・インスタンスの構成時に指定されたインスタンス ID です。

ソケット・データは以下の XML エレメントによって構成されます。

socketData
ルート要素。ここには、サブノード・インスタンス ID を指定する subnode というオプション属性が 1 つ含まれます。

attrGroup
このエレメントは、ソケット・データの対象である属性グループを指定します。name 属性は必須であり、属性グループ名を指定するために使用します。

in
このエレメントは、新しいデータ行を指定するために必要です。データ行に関する属性値はすべて、同じ in エレメントの子でなければなりません。

a
a エレメントは属性値を指定します。v 属性は必須であり、属性値を指定するために使用します。

データに代わるエラーの送信

ときにより、ソケット・データを送るアプリケーションが属性グループに必要なデータを収集できない場合があります。この場合、エージェントにデータを送信する代わりに、エラー・コードを返すことができます。 エラー・コードは、問題についてモニター環境に通知する手段となります。以下がエラーの例です。

```
<socketData><attrGroup name="abc"/><error rc="1000"/></attrGroup></socketData>¥n
```

エラー・コードは、エージェントで、すべてのソケット属性グループに共通するリストに定義されている必要があります。 エージェントがエラー・コードを受信すると、定義されたエラー・メッセージがエージェント・ログ・ファイルに記録されます。 さらに、パフォーマンス・オブジェクト状況という名前の属性グループにエラー・コード属性があり、この属性はエラー・コード・タイプによって更新されます。エラー・コード・タイプは、ユーザーが送信するエラー・コードに対し定義されます。

上記の例では、エージェントでエラー・コード値 1000 を定義する必要があります。以下のエラー・コード定義の例を参照してください。

表 *280.* エラー・コードの例

エラー・コード値	**エラー・コード・タイプ**	**メッセージ**
1000	APP_NOT_RUNNING	アプリケーションは実行されていません。

エラー・コードの送信時には、以下のようなメッセージがエージェント・ログ・ファイルに記録されます。

```
(4D7FA153.0000-5:customproviderserver.cpp,1799,"processRC") Received error code 1000
 from client. ¥Message: K1C0001E The application is not running
```

Tivoli Enterprise Portal で「パフォーマンス・オブジェクト状況」照会を選択した場合、その表内の **abc** 属性グループの行の「**エラー・コード**」列に APP_NOT_RUNNING の値が表示されます。

サンプル属性グループにエラーを送信すると、その属性グループに関して前に受信したデータはすべてクリアされます。 その属性グループにデータを送信すると、「パフォーマンス・オブジェクト状況」属性グループにエラー・コードが表示されなくなります。 さらに、エラー・コード 0 を送信して、その表からエラー・コードをクリアすることもできます。

イベントを生成する属性グループにエラーを送信しても、前に送信されたイベントのキャッシュはクリアされません。

アクション実行要求の処理

ソケット・クライアントは、アクション・コマンドが特定の接頭部と一致したときにエージェントからアクション実行要求を受信するように登録できます。 一致しないアクションはエージェントによって処理されます。この接頭部はエージェントが処理することが想定されているアクションと競合してはならないため、接頭部としてエージェント製品コードを使用してください。 Agent Builder によって提供されるアクション実行は、そのアクション実行が使用するデータ・ソース名にちなんで命名されます。例えば、JMX_INVOKE take action アクション実行は JMX データ・ソースで動作します。もう 1 つ、SSH スクリプト・データ・プロバイダーを使用する SSHEXEC アクション実行といった例もあります。 これらのアクションは製品コードを使用しないため、アクション実行の接頭部として使用するには、製品コードが安全な接頭部となります。

ソケット・クライアントは、長時間実行クライアントである必要があり、ソケットを開いたままにする必要があります。 ソケット・クライアントは、接頭部の登録要求を送信する必要があり、ソケットからの要求を listen していなければなりません。 エージェントは、データが流れていない場合でも、長時間実行クライアントのソケットをタイムアウトにしません。以下は、登録要求の例です。

```
<taskPrefix value="K42"/>¥n
```

この例で、エージェントが受信した "K42" で始まるアクション実行コマンドは、登録を開始したソケット・クライアントに転送されます。 以下は、ソケット・クライアントが受信する可能性のあるアクション実行要求の例を示しています。

```
<taskRequest id="1"><task command="K42 refresh" user="sysadmin"/></taskRequest>¥n
```

id は、クライアントに送信される要求をトラッキングするために、エージェントが使用する固有 ID です。ソケット・クライアントは、タスクに応答するときに、taskResponse エレメントの id 属性にこの ID を指定する必要があります。

ソケット・クライアントはアクションを処理し、応答を送信する必要があります。 応答の例を以下に示します。

```
 <taskResponse id="1" rc="1"/>¥n
```

アクションが正常に完了すると、0 の rc 属性値が返されます。rc の値は整数でなければなりません。0 以外の rc 値はすべて失敗と見なされます。 タスクの戻りコード値はエージェント・ログ・ファイルに記録され、エージェントに含まれる「アクション実行の状況」照会に表示されます。アクションの実行後に Tivoli Enterprise Portal に表示されるダイアログには、戻りコードは表示されません。このダイアログは、アクション実行コマンドで成功または失敗のどちらが返されたかを示します。障害が発生した場合の実際の戻りコードを確認するには、エージェント・ログまたは「アクション実行の状況」照会を参照する必要があります。

エージェントで使用されるソケット・クライアントがサポートするアクションを 文書化、作成、およびインポートする作業は、エージェント開発者の職責です。サポートされないアクションをユーザーがソケット・クライアントに送信した場合、こうしたシナリオに適切な方法で対応できるよう、このクライアントを発展させる必要があります。 登録済みの接頭部で始まるその他のアクションをユーザーが定義した場合、それらのアクションはクライアントに渡されます。こうしたシナリオに適切な方法で対応できるよう、このクライアントを開発する必要があります。

エージェントがソケット・クライアントからの応答を待つ時間を制御するタイムアウトが存在します。この設定はエージェントで定義される CDP_DP_ACTION_TIMEOUT という環境変数であり、デフォルト値は 20 秒です。

注 : ソケット・データ・ソース属性グループに定義されているエラー・コード・メッセージは、アクション実行には使用されません。同じ戻りコードの値を返すことができます。ただし、エージェントは定義されたメッセージを記録せず、「パフォーマンス・オブジェクト状況」属性グループの「エラー・コード」フィールドへの影響もありません。

ソケット・データのエンコード

ソケット・クライアントは、エージェントに送信されるデータをエンコードします。

エージェントに送信されるデータをソケット・クライアントがどのようにエンコードするかを知っておくことは重要です。

特殊文字

エージェントに送信されるデータには、各イベントまたはデータ・サンプルの末尾を除き、改行文字を含めてはなりません。 属性値の中に出現する改行文字は、別の文字に置き換えるか、1317 ページの表 281 に示すようにエンコードする必要があります。また、属性値によって XML 構文を切断しないように注意することも必要です。 次の表に、属性値の中に出現する、エンコードする文字を示します。

表 *281.* 属性値内のエンコードする文字

文字	ヘッダー
&	&
<	<
>	>
“	"

表 281. 属性値内のエンコードする文字 (続き)	
文字	**ヘッダー**
'	'
¥n

注 : エージェントは改行文字を使用して、クライアントから受信する応答を区切ります。 予期しない改行文字があると、データが正しく解析されません。

エージェントにはフル機能の XML パーサーは含まれていないため、1317 ページの表 281 にない文字に対して特殊なエンコードを行わないでください。例えば、セント記号 ¢ の代わりに ¢ または &cent をエンコードしないでください。

文字セット

特殊文字のエンコード方式に加え、エージェントはデータのエンコードに使用されたコード・ページも知る必要があります。各ソケット属性グループを定義して、エージェントにデータを **UTF-8** データとして送信するか、**ローカル・コード・ページ**として送信するか示します。クライアントがデータを送信する方法を確認してください。Java で作成されたクライアントを使用する場合は、エージェントへのデータ送信に使用するライターで、エンコード方式として **UTF-8** を指定します。属性グループの「**コード・ページ**」として **UTF-8** を指定します。**ローカル・コード・ページ**は、エージェントのローカル・コード・ページを意味します。 データがリモート・ソケットを使用して送信される場合、そのデータはエージェントのローカル・コード・ページに準拠しているか、UTF-8 を使用する必要があります。

数値データ

数値の属性値をフォーマットしている方法を確認してください。 エージェントに送信する数値には、特殊文字を含めてはなりません。 特殊文字の一例として、3 桁ごとの区切り文字があります。通貨記号や値の単位を示す文字なども特殊文字の例です。エージェントは、数値データの解析中に問題が発生すると、その問題を示すエラーを記録します。属性の解析に失敗すると、「パフォーマンス・オブジェクト状況」エラー・コードは設定されません。 以下は、エージェント・ログのエラー・メッセージの例です。

```
(4D3F1FD6.0021-9:utilities.cpp,205,"parseNumericString") Invalid characters :00:04 ¥
found getting numeric value from 00:00:04, returning 0.000000
```

注 : タイム・スタンプ属性に対して行う必要があるフォーマット設定の方法については、1199 ページの『タイム・スタンプ』 を参照してください。

ソケット・エラー

ソケット・クライアントから受信したデータで発生した問題について、エージェント・ログ・ファイルにエラーが書き込まれました。

記録されるその他のエラーとして、0 以外の値を返すアクション実行があります。ソケット・クライアントによって送信されるエラー値は、エラー・コードに関連するメッセージとともに記録されます。

ソケット・クライアントがエージェントにエラーの戻りコードを送信すると、属性グループの「パフォーマンス・オブジェクト状況」が設定されます。 エージェントによって定義された値以外の、他の値が表示される場合があります。 次の表では、ソケット属性グループで検出される可能性のある、その他の「エラー・コード」値を示します。

表 282. パフォーマンス・オブジェクト状況の値	
エラー・コード	**説明**
NO_ERROR	エラーは発生していません。属性グループに問題がないことを示します。サンプル・データの行に関する問題により、状態が NO_ERROR から変更されることはありません。エラー・コードとして NO_ERROR が表示されている場合でも、表示されている行数および属性値を検証する必要があります。

表 282. パフォーマンス・オブジェクト状況の値 (続き)	
エラー・コード	**説明**
NO_INSTANCES_RETURNED	サンプル属性グループに関して、ソケット・クライアントはデータ行を送信しませんでした。 エラーではありません。これは、この属性グループによってモニターされているリソースのインスタンスがないことを示しています。
XML_PARSE_ERROR	エージェントはクライアントから受信したデータを解析できませんでした。 詳しくは、エージェント・ログを参照してください。
OBJECT_CURRENTLY_UNAVAILABLE	クライアントは、エラー・コードのグローバル・リストに定義されていないエラー・コードをエージェントに送信しました。
GENERAL_ERROR	クライアントからのデータ収集中に問題が発生しました。クライアントがタイムアウト間隔内に要求に応答しなかったことが原因である可能性があります。詳しくは、エージェント・トレース・ログを参照してください。 クライアントはエラー・コードに GENERAL_ERROR を指定することもありますが、 さらに詳細なエラー・コードが定義されている場合は、そのコードを参照することをお勧めします。

ソケット構成
ソケット・データ・ソースをエージェントに追加した後に、指定したソケット・ポートからのデータを受信するようにエージェントを構成することができます。

このタスクについて

ソケット・データ・ソースを追加すると、Agent Editor の「**ランタイム構成**」ページに構成が表示されます。「ソケット構成」セクションには 以下のプロパティーがあります。

表 283. ソケット構成プロパティー			
Name	**有効な値**	**必須**	**説明**
ポート番号	0 または任意の正整数 デフォルト値は 0 です。	はい	ソケット・クライアントからのデータに関してエージェントが listen するために使用するポート。値 0 は、一時ポートが使用されることを示します。

エージェントは、使用されているポートの値をファイルに書き込みます。エージェント・コンピューターで稼働するソケット・クライアントは、後でこのファイルを読み取って、どのポートに接続するかを決定します。ポートが書き込まれるファイルの名前は *kxx_instanceName*_cps.properties です。この場合、*kxx* はエージェントの 3 文字の製品コードで、*instanceName* は複数インスタンス・エージェントのエージェント・インスタンス名です。 エージェントが複数インスタンス・エージェントでない場合、名前のこの部分は含まれないため、ファイル名は *kxx*_cp.properties となります。

Windows では、このファイルは %CANDLE_HOME%¥TMAITM6 ディレクトリー (32 ビット・インストールの場合) または %CANDLE_HOME%¥TMAITM6_x64 ディレクトリー (64 ビット・インストールの場合) に書き込まれます。UNIX では、このファイルは /tmp に書き込まれます。

手順

1. オプション: ご使用のシステムに複数のインターフェースがある 場合は、環境変数 CDP_DP_HOSTNAME を、ご使用のネットワーク・インターフェースのホスト 名または IP アドレスに設定します。
   a) Agent Editor の「**エージェント情報**」ビューに移動し、「**環境変数**」を選択します。
   b)「**追加**」をクリックして、「名前」フィールドを使用して環境変数のリストで CDP_DP_HOSTNAME を選択します。
   c)「値」フィールドにホスト名または IP アドレスを設定します。
2. エージェントを開始します。

   エージェントを開始すると、CDP_DP_HOSTNAME 環境変数で定義されたインターフェースにそのエージェントがバインドされます。CDP_DP_HOSTNAME が設定されていない場合、エージェントはデフォルトのホスト名にバインドされます。

   エージェントを一時ポートではなく、定義済みポートにバインドする場合は、構成プロパティー「**ポート番号**」 (CP_PORT) を設定できます。

   ポート番号の構成プロパティーを設定するには、以下の手順を実行します。

   a) Agent Editor の「**ランタイム構成**」ビューに移動します。
   b)「**ランタイム構成情報**」ペインで、「**ソケット構成**」 > 「**ソケット**」 > 「**ポート番号**」を選択します。
   c)「**デフォルト値**」にポート番号の値を入力します。
   値を入力しないと、値 0 が使用されます。値 0 は、一時ポートが使用されることを示します。

リモート・ソケット・ポート接続
リモート・ソケット・ポートからのデータを受信するようにエージェントを構成することができます。エージェントは、リモート・システムへのネットワーク・インターフェース 接続が存在するシステムで実行する必要があります。

手順

1. 以下のステップを実行して、環境変数 CDP_DP_ALLOW_REMOTE の値を YES に設定します。
   a) Agent Editor の「**エージェント情報**」ページに移動し、「**環境変数**」を選択します。
   b)「**追加**」をクリックして、「名前」フィールドを使用して環境変数のリストで **CDP_DP_ALLOW_REMOTE** を選択します。
   c)「**値**」フィールドを「はい」に設定します。
2. 1319 ページの『ソケット構成』で詳しく説明されている手順に従います。

   制約事項 :

   - ソケット・アプリケーションとエージェントの間で送信されるデータには以下が適用されます。
     - ソケット・データ・プロバイダー用に定義されている XML 構文に準拠している必要があります。詳しくは、1317 ページの『 ソケット・データのエンコード』を参照してください。
     - UTF-8 でエンコードする必要があります。
     - 平文で (暗号化されずに) 送信されます。データに機密情報が含まれる場合は、SSH トンネルまたはその他のエージェント外部のメカニズムによって通信を保護する必要があります。
   - エージェントはどのリモート・ホストから受信したデータでも処理するため、適切なファイアウォールやネットワーク・トラフィック・フィルターで環境を保護する必要があります。

タスクの結果

エージェントが稼働しているシステムに接続可能な任意のシステム上で、ソケット・データ・プロバイダーを実装するコードを実行することができます。

ソケットのサンプル・スクリプト

このサンプル・スクリプトは、ソケット・クライアントを作成する場合に例として使用できます。

Perl のサンプル

以下の Perl スクリプトのサンプルは、ソケットに接続し、データを送信します。 このサンプルは、製品コードが k00 であり、属性グループが SocketData である、UNIX で実行するエージェント用に作成されたものです。

```
#!/usr/bin/perl -w
# SocketTest.pl
# A simple Agent Builder Socket client using IO:Socket
#------------------------

use strict;
use IO::Socket;

# Initialize socket connection to the agent
#-----------------------
my $host = '127.0.0.1';
my $port = 0;
# This sample is for an agent with the k00 product code.  The product code is
# used in the following line to find the file containing the port number to use.
open PORTFILE, "/tmp/k00_cps.properties" || die "Port file not found $!¥n";
while (<PORTFILE>) {
    if (/^CP_PORT=([0-9]+)/) {
        $port = $1;
    }
}

if ($port == 0) {
    die "Could not find port to use to connect to agent.¥n";
}

my $sock = new IO::Socket::INET( PeerAddr => $host, PeerPort => $port,
Proto => 'tcp'); $sock or die "no socket :$!";

# The following call sends 2 rows of data to the agent.  Each row contains 1
# String attribute and 3 numeric attributes.
syswrite $sock, "<socketData><attrGroup name=¥"SocketData¥"><in><a v=¥"A message
from perl¥"/>  ¥<a v=¥"1¥"/><a v=¥"2¥"/><a v=¥"123¥"/></in><in><a v=¥"More from
perl¥"/><a v=¥"456¥"/>  ¥<a v=¥"123¥"/><a v=¥"789¥"/></in></attrGroup>
</socketData>¥n";

close $sock;
```

ソケット属性グループのテスト

Agent Builder 内で作成したソケット属性グループをテストできます。

始める前に

属性グループをテストするには、データを送信するソケット・クライアントが必要です。Perl スクリプトで作成されたソケット・クライアントの例は、1321 ページの『ソケットのサンプル・スクリプト』で確認できます。

制約事項 : 他のほとんどの属性グループとは異なり、ソケット属性グループの作成中にソケット属性グループをテストすることはできません。 この属性グループは、作成が完了したらテストできます。

手順

1. エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**ソケット・クライアントのテスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数と構成プロパティーを設定します。

   詳しくは、1376 ページの『属性グループのテスト』を参照してください。

3. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
4. エージェントが開始されると、エージェントはその構成に従ってソケット・データを listen します。
5. エージェントのデータ収集をテストするには、エージェントの構成に一致するソケット・データを生成します。

   ソケット・クライアントを使用して、ソケット・データを生成できます。

   エージェントがその構成に一致するソケット・データを受信すると、エージェントはデータを内部キャッシュに追加します。
6. Tivoli Enterprise Portal からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。

   「**ソケット・クライアントのテスト**」ウィンドウで、エージェントのキャッシュ内にある、エージェントが最後に始動された後のすべてのデータが収集および表示されます。
7. 何かが正しく機能していないと思われる場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
8. 「**エージェントの停止**」をクリックしてエージェントを停止します。
9. 「**OK**」または「**キャンセル**」をクリックして「**ソケット・クライアントのテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

Java API を使用したデータのモニター

データ・ソースを定義して、Java API を使用した Java プラットフォーム上の長時間実行アプリケーションとの対話を実行できます。始動時にエージェントがアプリケーションを始動し、定期的にアプリケーションと対話します。エージェントをビルドする際に、Agent Builder によってアプリケーションのソース・コードが作成されます。適切なデータを収集するために、このコードをカスタマイズする必要があります。コードに応じて、データ・ソースは、単一行、複数行、またはイベント・データを含む複数のデータ・セットを生成します。

このタスクについて

他の Agent Builder データ・ソースを使用して収集できないデータを収集するには、Java API データ・ソースと Java プログラミング言語を使用します。エージェントは Java アプリケーションを開始し、シャットダウンする時刻になるとシャットダウン要求を送信します。 Java アプリケーションが終了するのは、終了を要求されたときのみでなければなりません。

Java API 属性グループを含むエージェントは、Java アプリケーション・プロセスとのインターフェースをとります。Java アプリケーションは Java プロバイダー・クライアント API を使用して、エージェントとのインターフェースをとります。API については、Tivoli Monitoring Knowledge Center の Javadoc を参照してください。Java API を使用して、以下を行うことができます。

- エージェント・プロセスに接続し、Java アプリケーションがサポートする属性グループに登録する
- サンプル・データの要求を受信し、応答する
- イベントを生成する属性グループに関するデータを非同期的に送信する
- データ収集に失敗している属性グループに関するエラーを送信する
- サブノード・インスタンスが構成されているサブノード内の属性グループをサポートする
- 「アクション実行」要求を受信し、応答する

Java アプリケーション内でデータを収集し、そのデータを Java API を使用して送信する属性グループを作成するには、以下の手順を実行します。この手順では、Java アプリケーションを作成する際の開始点として使用するサンプル Java アプリケーションの作成方法を示します。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**カスタム・プログラム**」をクリックします。
2. 「**データ・ソース**」領域で「**Java API**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**Java API 情報**」ページで、属性グループ名を入力します。
5. 属性グループのヘルプ・テキストを入力します。
6. 属性グループに対して、「**単一データ行の生成**」、「**複数のデータ行を生成可能** 」、または「**イベントの生成**」のいずれかを選択します。この選択は、ウィザードの最後で作成されるサンプル Java アプリケーションに影響します。詳しくは、1314 ページの『データの送信』を参照してください。
7. オプション: 「**拡張**」をクリックして、属性グループの拡張プロパティーを変更します。「**拡張**」は、属性グループに対して「**複数のデータ行を生成可能**」または「**イベントの生成**」を選択した場合に選択可能です。
8. 「**次へ**」をクリックします。
9. 「**属性情報**」ページで、属性グループの最初の属性を指定します。属性の作成について詳しくは、1194 ページの『属性の作成』を参照してください。
10. 「**追加属性の追加**」を選択し、「**次へ**」をクリックして、その他の属性をエージェントに追加します。属性への参照は、ウィザードの最後で作成されるサンプル Java アプリケーションに取り込まれます。
11. 「**次へ**」をクリックします。
12. 「**グローバル Java API データ・ソース情報**」ページで、クラス名と JAR ファイル名を入力します。

    このクラス名は、Java の開始時にその main メソッドが呼び出される完全修飾クラス名です。 サンプル Java アプリケーションは、このクラスの main Java メソッドを使用して作成されます。

    JAR ファイルは、Java アプリケーションを構成する Java クラスを格納するアーカイブです。 JAR ファイルはエージェントと一緒にパッケージ化され、インストールされます。
13. オプション: 「**グローバル Java API データ・ソース情報**」ページの「**エラー・コード**」セクションで、Java アプリケーションが送信できるエラー・コードを定義します。これらのエラー・コードは、データを収集できない場合に Java アプリケーションによって送信されます。

    制約事項 : エラー・コードは 256 文字までに制限されています。使用できるのは、ASCII 文字、数字、および下線のみです。 スペースは許可されません。

    a) 「エラー・コード」セクションで「**追加**」をクリックします。

    b) 「**Java API エラー・コードの定義**」ウィンドウで、表示値を入力します。

    c) 内部値を入力してください。内部値は、1,000 から 2,147,483,647 の間の整数でなければなりません。

    d) 各エラーのメッセージ・テキストを定義します。以前に入力したメッセージ・テキストを、リストから選択して利用することもできます。「**OK**」をクリックして、「**グローバル Java API データ・ソース情報**」ページに戻ります。

    メッセージはエージェント・ログ・ファイルに記録されます。

    e) 適切なメッセージ・テキストがない場合、 「**参照**」をクリックして、メッセージ・テキストをセットアップします。

    「メッセージ」(リスト) ウィンドウが表示されます。メッセージ・ウィンドウには、エージェントで定義されたメッセージがリストされます。 メッセージを定義するまで、リストはブランクになっています。 「**編集**」を使用すると定義済みメッセージを変更することができ、「**削除**」を使用すると定義済みの 1 つ以上のメッセージを削除することができます。

    f) 「メッセージ」(リスト) ウィンドウで、「**追加**」をクリックして「**メッセージ定義**」 ウィンドウを表示します。「**メッセージ定義**」ウィンドウで、新規メッセージの意味を示すテキストを入力し、メッセージ・タイプを選択できます。

    注 : メッセージ ID は自動的に生成されます。

    g) 「**OK**」をクリックします。

h) 「メッセージ」(リスト) ウィンドウに新規メッセージが表示されます。メッセージを検証して「**グローバル Java API データ・ソース情報**」ページに戻るには、「**OK**」 をクリックします。

14. オプション: 「**グローバル Java API データ・ソース情報**」ページの「**補足ファイル**」セクションで、エージェントと一緒にパッケージ化され、エージェントのインストール時にエージェント・システムにコピーされるファイルを追加することができます。 Java プロバイダー・クライアント API JAR ファイルは、ここにはリストされません。このファイルは自動的にエージェント・システムにコピーされます。「**ファイル・タイプ**」列は、各ファイルの想定される使用方法を説明します。 次の表 (1324 ページの表 284) では、可能な 3 つの使用方法を説明します。「**編集**」をクリックして、インポートしたファイルを編集します。詳しくは、1279 ページの『コマンド・ファイル定義の編集』 を参照してください。

表 *284.* 補足ファイルのファイル・タイプ

ファイル・タイプ	**説明**
実行可能ファイル	エージェントと一緒に実行可能ファイルを含める場合は、このオプションを選択します。エージェントはこのファイルを使用しませんが、このファイルは、使用する Java アプリケーションのパス内に置かれます。
ライブラリー	エージェントと一緒にライブラリーを含める場合は、このオプションを選択します。 エージェントはこのファイルを使用しませんが、このファイルは、使用する Java アプリケーションのライブラリー・パス内に置かれます。
Java リソース	エージェントと一緒に Java リソースを含めるにはこのオプションを選択します。 エージェントはこのファイルを使用しませんが、このファイルは、使用する Java アプリケーションのクラスパス内に置かれます。

注 : Java リソース補足ファイルが Agent Builder に追加されると、補足ファイルはプロジェクト・クラスパスに自動的に追加されます。Java コンパイラーは補足ファイルを使用して、リソース内のクラスへの、コードに含まれるすべての参照を解決します。

エージェントでの補足ファイルのインストール場所については、1393 ページの『システム 上の新規ファイル』 を参照してください。

15. オプション: データがサンプリングされる 場合は、この属性グループから返されるデータを制限するフィルターを作成します。 「**拡張**」をクリックしてフィルターを 作成します。

注 : データは、「**Java API 情報**」ページで「**イベントの生成**」を選択しなかった場合にサンプリングされます。

属性グループからのデータのフィルタリングについて 詳しくは、1202 ページの『属性グループのフィルタリング』 を参照してください。

16. オプション: 構成プロパティーをサブノードに 追加します。

サブノードにこのデータ・ソースを追加する場合は、「**サブノード構成のオーバーライド**」ページが表示されるので、ここで構成プロパティーをサブノードに 追加できます。サンプル Java アプリケーションを作成するには、サブノードに少なくとも 1 つの構成プロパティーが必要です。サンプルは構成プロパティーを使用してサブノード・インスタンス同士を区別するため、少なくとも 1 つの構成プロパティーが必要です。

17. 以下の手順のいずれかを行います。
    a) **エージェント**・ウィザードを使用する場合は、「**次へ**」をクリックしてください。必要に応じて、ウィザードに入力します。
    b) それ以外の場合は、「**完了**」をクリックし、データ・ソースを保存して Agent Editor を開いてください。次に、メインメニューで「**ファイル**」 > 「**保存**」を選択します。

この時点で、Agent Builder によってモニター・アプリケーションのソース・コードが作成されます。このコードは、プロジェクト・ディレクトリーの src サブディレクトリーに配置されます。このコードを編集してモニター・アプリケーションを作成してください。

次のタスク

「**Java API 設定**」ページで正しいオペレーティング・システムを選択します。 この選択は、エージェントに定義されたオペレーティング・システムとは異なるオペレーティング・システムでこの属性グループおよび Java アプリケーションが動作する場合に行います。ページを開くには、アウトライン・ビューで「**Java API 設定**」をクリックするか、任意の Java API 属性グループ・ページの Agent Editor で「**グローバル設定**」をクリックします。

注 : エラー・コードおよび補足ファイルは、後で「**Java API 設定**」ページの「**エラー・コード**」セクションおよび「**補足ファイル**」セクションで更新できます。

Java アプリケーションの実行

Java アプリケーションおよびその依存関係の初期化に関する情報

Java アプリケーションの初期化

エージェントは、その開始および初期化中に Java アプリケーションを開始します。 プロセスの開始に使用される Java ランタイムを制御するには、構成設定を使用します。その構成には Java 仮想マシンの引数と Java ロギング・レベルも指定できます。 Java API 構成について詳しくは、1334 ページの『Java API 構成』を参照してください。Java プロセスは、そのエージェントで定義されている環境変数を継承します。ランタイム構成の設定もその環境に置かれ、API 呼び出しを使用して照会できます。

Java アプリケーションは長時間実行プロセスでなければなりません。 API からシャットダウン要求を受信するまで Java クライアントは終了してはなりません。 Java アプリケーションがエージェントへの 登録後に終了すると、エージェントは 3 回を限度として Java アプリケーションの再始動を試みます。 データ収集が正常に再開されると、この再始動カウントはリセットされます。 Java アプリケーションが終了し、再始動が開始されると、エージェントはエラーを記録します。

注 : 属性グループの登録の完了前に Java アプリケーションが終了した場合、再始動は試みられません。

依存関係

Java アプリケーションは Java ランタイム環境を使用する必要があります。 以下の Java のバージョンをサポートします。

- Oracle Corporation Java バージョン 5 以降
- IBM Corporation Java バージョン 5 以降

エージェントが構成および開始される時点で、Java がエージェント・システムにインストール済みになっている必要があります。 エージェントとの通信に使用される API を格納する JAR ファイルは、エージェントのランタイムに組み込まれ、さらに JVM のクラスパスにも組み込まれます。ご使用の Java アプリケーションに必要な追加の JAR ファイルはすべて、補足ファイルとして Java API 属性グループに定義する必要があります。 ファイル・タイプが *Java* リソース である補足ファイルは、Java API JAR ファイルとともに Java アプリケーションの基本クラスパスに自動的に追加されます。

エージェントには含まれていない、Java アプリケーションのランタイム 動作に必要な JAR ファイルは、すべて「外部 *JAR* のクラスパス」構成設定に含まれている必要があります。

生成されたサンプル Java アプリケーション

Agent Builder が生成するコード、およびモニターするリソース用に追加または置換する必要があるコードについて説明するリファレンス。

1 つ以上の Java API データ・ソースを持つエージェントを作成すると、Agent Builder は Java アプリケーション・ソース・コードを生成します。このコードはエージェント・プロジェクト内に生成され、エージェントの構造に従います。生成されたアプリケーションに独自の Java コードを追加する必要があります。このコードは、サンプル属性グループに関するデータを収集し、イベント・ベースの属性グループに送信されるイベントを処理し、問題が発生した場合はエラーを報告し、タスクを実行します。生成されたアプ

リケーションによってエージェントにデータが提供されますが、それはサンプル・データであるため、モニター対象のリソースから取得されたデータに置き換えられます。

サンプル・エージェントには、以下の特性があるものと想定されます。

- 製品コード: K91
- Java API Main クラス: agent.client.MainClass
- 1326 ページの図 47 に示すエージェント・データ・ソース構造:

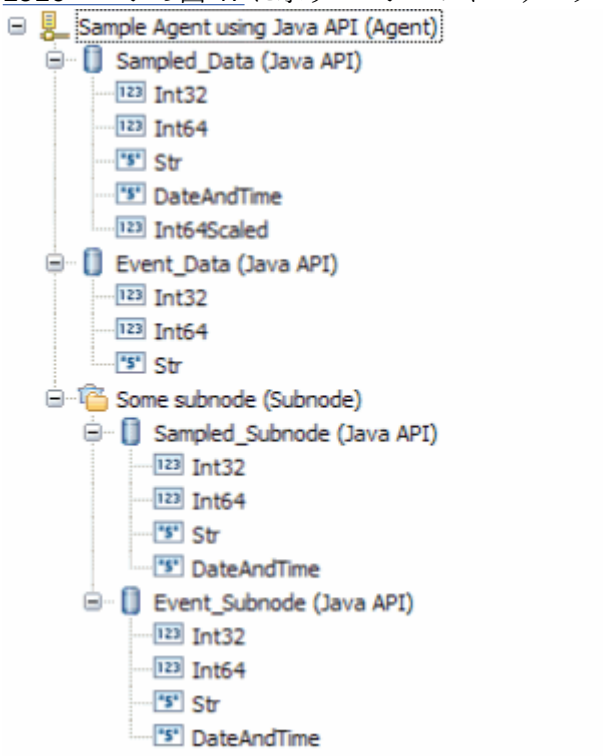


図 47. エージェント構造の例

- 何らかのサブノード 構成プロパティー: K91_INSTANCE_KEY

クラス構造

生成された Java アプリケーションは、多くの場合、エージェントとのインターフェースをとるコードと、モニターするリソースとのインターフェースをとるコードを 分離しています。 ここには、変更するファイルと変更しないファイルが含まれています。

Agent Builder によって以下の Java クラスが作成されます。

MainClass (agent.client パッケージ)
「**グローバル Java API データ・ソース情報**」ページで指定したクラス。このクラスには、main メソッドと、アクション実行要求を処理するメソッドが含まれています。このクラスは、次に説明するヘルパー・クラスから継承されます。 モニターするリソースと実行するアクションとのインターフェースをとるには、このクラスを変更する必要があります。

MainClassBase (agent.client パッケージ)
サーバーとの接続を初期化し、属性グループを登録し、サーバーからの要求を待つヘルパー・クラス。このクラスは変更しないでください。

Sampled_Data、Sampled_Subnode、Event_Data、および Event_Subnode クラス (agent.client.attributeGroups パッケージ)
Java API 属性グループごとに、その属性グループに関するデータ収集要求を処理する、またはその属性グループに関するイベントを生成するクラスが 1 つずつ存在します。 これらのクラスはそれぞれ、次に説明するヘルパー・クラスのいずれかから継承されます。 モニターするリソースからデータを収集するには、これらのクラスを変更する必要があります。

Sampled_DataBase、Sampled_SubnodeBase、Event_DataBase、 および Event_SubnodeBase クラス (agent.client.attributeGroups パッケージ)
Java API 属性グループごとに 1 つずつ存在するヘルパー・クラス。このクラスは、内部クラス内のグループの属性の構造を定義します。これらのクラスは変更しないでください。

ICustomAttributeGroup インターフェース (agent.client.attributeGroups パッケージ)
各属性グループ・クラス内の public メソッドを定義するインターフェース 。このインターフェースは変更しないでください。

変更可能なクラスは、Agent Builder によって上書きされることはありません。Agent Builder は、それらのクラスが存在しない場合のみ、クラスを新規に作成します。

ヘルパー・クラスおよびインターフェースは 、Agent Builder が保存されるたびに上書きされます。エージェントを変更して保存すると、Java API 属性グループに対する構造上の変更を反映するためにヘルパー・クラスが更新されます。このインターフェースとヘルパー ・クラスのヘッダーには、そのファイルを変更しないように通知する警告が記述されています。

初期化とクリーンアップ

エージェントの開始時に MainClass の main メソッドが呼び出されます。 このメソッドは MainClass インスタンスを作成した後、エージェントの要求を受信し、処理するための長時間実行メソッドを入力します。

初期化およびクリーンアップ・コードのほとんどは、MainClass に追加する必要があります。コンストラクターで、リソースの作成またはリソースへのアクセスに必要な初期化を追加します。リモート・リソースへの接続のオープン、ハンドルの作成、またはデータ構造の初期化を行うことができます。

エージェントが終了する前に、stopDataCollection メソッドが呼び出されます。Java アプリケーションの終了前に接続を閉じるかクリーンアップする場合は、stopDataCollection メソッドにそのコードを追加します。

特定の属性グループにのみ初期化が必要な場合、その属性グループのクラスのコンストラクターにその 初期化を追加できます。同様に、特定の属性グループについてのみクリーンアップが 必要な場合は、その属性グループの stopDataCollection メソッドにクリーンアップ・コードを追加することができます。

Java アプリケーションのどのコードも、ロガー・オブジェクトを使用してログ項目を書き込むことができます。 (メイン・ヘルパー・クラスは、保護されたロガー・オブジェクトをそのコンストラクター内に作成します。属性グループのヘルパー・オブジェクトは、このロガーに対する保護された参照を、自分のコンストラクター内に作成します)。ロガー・オブジェクトは、Java トレース・ログ・ユーティリティーを 使用します。このロガーによって作成されるトレース・ログから、エラーおよび詳細なトレース情報を取得できます。 このトレース情報は、プロバイダーに関する問題のトラブルシューティングを 行う場合の重要な情報です。

stopDataCollection の呼び出し時に、クリーンアップ作業を別のスレッドに渡す場合は、そのスレッドが終了するのを待ってから、stopDataCollection メソッドから戻る必要があります。そうしないと、main スレッドが完了していることが原因で、プロセスの終了時にクリーンアップ作業が突然終了してしまう可能性があります。

エージェント構成設定の 1 つは、Java トレース・レベルに関するものです。 次の表に、JAVA_TRACE_LEVEL 構成プロパティーで設定できる値を示します。API によってユーザー用のロガーが作成される場合は、そのロガーが使用するレベルも以下の表に示します。

表 *285. Java* トレース・レベル・オプション

構成するトレース・レベル	Java ロギング・トレース・レベル	説明
オフ	オフ	ロギングは行われません。
エラー	SEVERE	Java アプリケーションで発生した問題をトレースします。
警告	WARNING	エラーおよび潜在的なエラーをトレースします。

表 285. Java トレース・レベル・オプション (続き)		
構成するトレース・レベル	**Java ロギング・トレース・レベル**	**説明**
情報	INFORMATION	Java アプリケーションに関する重要情報をトレースします。
デバッグ詳細度 - 低	FINE	Java アプリケーションの動作分析に必要な詳細の概要をトレースします。
デバッグ詳細度 - 中	FINER	Java アプリケーションのプログラム・フローに関する詳細をトレースします。
デバッグ詳細度 - 高	FINEST	Java アプリケーションに関するすべての詳細をトレースします。
すべて	ALL	すべてのメッセージをトレースします。

この例で、Java アプリケーションによって作成されたログ・ファイルの名前は k91_trace0.log です。エージェントが複数インスタンス・エージェントの場合、ログ・ファイル名にそのインスタンス名が含まれます。

注 : 標準エラーまたは標準出力にメッセージを書き込まないでください。Windows システムでは、これらのメッセージは失われます。 UNIX および Linux システムでは、ラップされないファイルにこのデータが書き込まれます。

サンプル属性グループ・データの収集

サンプル属性グループ (1 つ以上のデータ行を収集する属性グループ) のクラスには、Sampled_Data.collectData などの collectData メソッドが含まれています。 このメソッドは、エージェントがデータを要求すると必ず呼び出されます。

属性グループのヘルパー・クラスは、Attributes という内部クラスを定義します。このクラスには、属性グループで定義されている属性ごとにフィールドが 1 つずつあります。派生属性はエージェントが計算するため、ここに含まれません。 1328 ページの表 286 に示すように、属性フィールドのデータ・タイプは、Java において Tivoli Monitoring 属性タイプに相当するものです。

表 286. 属性フィールドのデータ・タイプとそれに相当する *IBM Tivoli Monitoring* 属性タイプ	
Tivoli Monitoring タイプ	**属性フィールドのデータ・タイプ**
ストリング	ストリング
数値、32 ビット、小数部の調整なし	int
数値、64 ビット、小数部の調整なし	long
数値、ゼロ以外の小数部の調整	double
タイム・スタンプ	カレンダー

collectData メソッドは以下を行う必要があります。

1. モニター対象リソースから適切なデータを収集する。
2. 属性オブジェクトを作成する。
3. 属性オブジェクトのフィールドにデータを 追加する。
4. Attributes.setAttributeValues メソッドを呼び出して、内部バッファーにデータをコピーする。
5. 各データ行で必要に応じてステップ 1 から 4 を繰り返す。 (ステップ 1 から 4 をまとめてスキップし、行を返さないようにすることもできます。この場合、「パフォーマンス・オブジェクト状況」テーブル

の「エラー・コード」列に NO_INSTANCES_RETURNED の値が表示されます。エラー・コードについて詳しくは、1331 ページの『エラー・コード』を参照してください。)

6. AgentConnection.sendData を呼び出してデータをエージェントに送信するか、または sendError を呼び出して、setAttributeValues に対する呼び出しでコピーされたデータを破棄し、代わりにエラー・コードを送信する。

リソースからデータを収集し (ステップ 1)、生成されたアプリケーションで使用されているサンプル・データを置き換える必要があります。

属性オブジェクトにデータを取り込むには、(生成されたアプリケーションで行うのと同様に) 属性コンストラクターを使用する際にデータを渡すことができます。あるいは、ゼロ引数コンストラクターを使用して属性オブジェクトを作成してから、収集した属性値に属性オブジェクトのフィールドを 割り当てます。フィールドの 名前は属性と同じですが、英小文字で始まります。

サブノードに関するサンプル・データの収集

サンプル属性グループがサブノード 内にある場合、モニター対象のリソースは複数存在すると考えられます (サブノードごとにそれぞれ異なるリソース)。 どのリソースからデータを収集するかを判別する必要があります。モニターするリソースを識別する 1 つ以上の構成プロパティーが存在するはずです。

この例では、1 つの構成プロパティー K91_INSTANCE_KEY に、データを収集するリソースを識別する値が含まれていると想定します。

正しいリソースを見つけるには、以下のステップを使用します。

1. AgentConnection.getConfiguredSubnodeInstanceIDs を呼び出して、構成されているすべてのサブノードのインスタンス ID を取得します。構成されている各サブノードには固有のインスタンス ID があります。
2. インスタンス ID ごとに、AgentConnection.getSubnodeConfigurationProperty を呼び出して K91_INSTANCE_KEY 構成プロパティーを取得します。
3. K91_INSTANCE_KEY 内の値によって表されるリソースを見つけます。

これらのステップは、1328 ページの『サンプル属性グループ・データの収集』で説明している一連のステップの前に、collectData メソッドで実行することができます。

または、これらのステップを属性グループ・クラス・コンストラクターで実行して、インスタンス ID からリソースへの直接マッピングを作成することもできます。サンプルの属性グループのクラス・コンストラクターは Sampled_Subnode コンストラクターです。このプロシージャーにより、エージェントの存続期間を通して使用可能なハンドルを作成したり接続を開いたりすることもできます。ハンドルを作成したり接続を開いたりすることで、より効率的にリソースにアクセスできるようになります。

生成されたコードはコンストラクター内に MonitoredEntity タイプのサンプル・リソース・オブジェクトを作成し、それらを configurationLookup マップに追加します。MonitoredEntity 内部クラスを削除して、MonitoredEntity オブジェクトを、ユーザー自身のリソースにアクセスするオブジェクトに置き換える必要があります。 collectData メソッドですべてのルックアップ 手順を実行する場合、そのクラスから configurationLookup マップを削除することができます。

コンストラクターを使用する場合は、サブノード・インスタンス ID をリソースにマップするために collectData メソッドで行うステップは以下のようになります。

1. Request.getSubnodeInstanceID を呼び出して、要求パラメーターからサブノードのインスタンス ID を取得します。
2. コンストラクターで作成されたマップからリソース・オブジェクトを取得します。
3. 1328 ページの『サンプル属性グループ・データの収集』で詳しく説明されている一連のステップを実行して、エージェントにデータを送信します。

Agent Builder の例では、任意のサブノード・プロパティーが選択されます (ここでは K91_INSTANCE_KEY)。正しいプロパティーではない場合、または正しいリソースを識別するために複数のプロパティーが必要な場合は、リソースを識別するためのプロパティーを選択する必要があります。

イベントの送信

イベントを生成する属性グループでは、collectData メソッドは定期的に呼び出されません。イベントは、リソースがそのイベントを通知するとユーザーのアプリケーションによって送信されます。

イベント生成の 1 つの例として、イベント・ベースの属性グループ用に生成されたコードが、SampleEventClass という内部クラスから実行されるスレッドを作成および開始する場合があります。この例で使用されるイベント・ベースの属性グループは Event_Data クラスです。このスレッドは定期的にウェイクアップして、イベントを送信します。定期的にリソースをポーリングしてイベントがあるか確認する場合は、イベントが生成されたときの Event_Data クラスの構造を使用することができます。

1. Event_Data コンストラクターから、スレッドを作成および開始します。
2. そのスレッドの run メソッドで、エージェントが終了するまでループします。
3. イベントが発生したかどうかを確認するまで、一定期間スリープします。ポーリング間隔を、5000 ミリ秒からエージェントに適した数値に変更することもできます。
4. 1 つ以上のイベントが発生したかどうかを確認します。生成されたアプリケーションはチェックしませんが、常に 1 つのイベントを通知します。
5. 通知が必要なイベントごとに、通知するイベント・データを取得します。
6. 属性オブジェクトを作成し、データを取り込みます (collectData メソッドがサンプル属性グループに対して行った場合と同様にします)。
7. Attributes.sendEventData メソッドを呼び出します。イベントは 1 行で構成されるため、一度に送信できるイベントは 1 つのみです。

また、自身のスレッドからイベントを報告する Java API で作業している場合は、Event_Data コンストラクターでそのスレッドを初期化することができます。 また、独自のイベント処理オブジェクトをリソースのイベント処理メカニズムに登録することもできます。イベント・ハンドラーで、以下のステップを実行します。

1. 通知されるイベント・データを取得します。
2. 属性オブジェクトを作成し、データを取り込みます。
3. Attributes.sendEventData メソッドを呼び出します。

この場合、Event_Data クラスで独自のスレッドを作成する必要はなく、SampleEventClass クラスも必要ありません。

サブノードでのイベントの送信

サブノード属性グループに関するイベントが検出された場合、Java アプリケーションはそのイベントを正しいサブノードに通知する必要があります。

この例では、1 つの構成プロパティー K91_INSTANCE_KEY に、イベントを生成する可能性のあるリソースのインスタンスを識別する値が含まれていると想定します。 さらに、イベントで通知されるデータとともに、K91_INSTANCE_KEY プロパティーの値が取得されると想定します。 プロパティーとデータを取得するために、Java アプリケーションは以下のステップを実行します。

1. 通知されるイベント・データを「インスタンス・キー」とともに取得します。
2. 属性オブジェクトを作成し、データを取り込みます。
3. AgentConnection.getConfiguredSubnodeInstanceIDs を呼び出して、構成されているすべてのサブノード・インスタンス ID のリストを取得します。
4. サブノード・インスタンスごとに、AgentConnection.getSubnodeConfigurationProperty を呼び出して K91_INSTANCE_KEY の値を取り出します。
5. イベント・データとともに取得した値と一致する K91_INSTANCE_KEY の値を検出したら、対応するサブノード・インスタンス ID を覚えておきます。
6. Attributes.sendSubnodeEventData を呼び出して、覚えておいたサブノード・インスタンス ID を渡します。

生成されたアプリケーションは、ステップ 4 と 5 に示す検索処理は実行しませんが、代わりにすべてのサブノードの属性グループにイベントを通知します。この動作は、実動エージェントの正しい動作ではない可能性があります。

アクション実行コマンド

アクション実行コマンドは、Tivoli Enterprise Portal 内で、または tacmd createaction コマンドを使用して定義されます。アクションをエージェントの Agent Builder プロジェクトにインポートすることにより、エージェントのインストール時にそれらのアクションを作成することができます。アクション実行コマンドのインポートについて詳しくは、1403 ページの『アプリケーション・サポート・ファイルのインポート』を参照してください。

生成された Java アプリケーションは、エージェントの製品コードで始まるアクション (K91Refresh など) に登録します。 この登録処理は、メイン・ヘルパー・クラス (MainClassBase) 内の registerActionPrefix メソッドから実行します。他の接頭部を登録したい場合、またはどのアクションについても登録しない場合は、MainClassBase で registerActionPrefix をオーバーライドします。

ユーザーのエージェントによって登録された接頭部で始まるアクションをエージェントが実行する必要がある場合、MainClass.takeAction メソッドが呼び出されます。Request.getAction() を呼び出すコードを追加し、適切なアクションを実行してから AgentConnection.sendActionReturnCode を呼び出して、アクションからの戻りコードを送信します。戻りコード 0 はアクションが成功したことを意味し、それ以外の戻りコードはアクションが失敗したことを意味します。

例外処理

collectData メソッドと takeAction メソッドは任意の Java 例外をスローできるため、収集コードを使用して、例外をキャッチすることなくスローすることができます。ヘルパー・クラスが例外を受け取ると、handleException メソッド (collectData の場合) または handleActionException メソッド (takeAction の場合) が呼び出されます。

collectData 例外の場合、例外の発生時、またはデータ収集での問題発生時に、AgentConnection.sendError を呼び出す必要があります。生成されたアプリケーションは、エラー・コード GENERAL_ERROR を渡します。ただし、このエラー・コードは、検出された問題を最も的確に記述する、エージェントによって定義されたエラー・コードに置き換える必要があります。エラー・コードの追加について詳しくは、ステップ 1323 ページの『13』 を参照してください。

takeAction 例外では、ゼロ以外の戻りコードが送信されるように AgentConnection.sendActionReturnCode を呼び出す必要があります。

AgentConnection メソッドによっては、com.ibm.tivoli.monitoring.agentFactory.customProvider.CpciException から派生した例外をスローするものがあります。 データの収集中に CpciException がスローされた場合、ヘルパー・クラスがこの例外を処理するため、handleException メソッドは呼び出されません。

注 : handleException メソッドを使用するのではなく、collectData メソッド内で例外をキャッチする場合は、必ず CpciException を再スローしてください。CpciException を再スローすることにより、基本クラスで CpciException を処理することができます。

エラー・コード

例外またはその他のリソース・エラーに対する典型的な応答は、AgentConnection.sendError メソッドを呼び出してエージェントにエラー・コードを送信することです。イベント・ベースの属性グループに関するエラーは、いつでも送信できます。 サンプル属性グループに関するエラーは、データ収集要求への応答として、および sendData 呼び出しの代わりとしてのみ送信できます。

エージェントにエラーを送信する場合、以下のようになります。

1. エージェント・トレース・ログにエラー・メッセージが記録されます。このエラー・メッセージには、エラー・コードと、そのエラー・コードに対して定義されているメッセージが含まれます。

2. 属性グループに関する状況情報を取得するために、「パフォーマンス・オブジェクト状況」照会を表示できます。 「エラー・コード」列は、送信されたエラーに対して定義されているエラー・コード・タイプに設定されます。 属性グループに関するデータをエージェントが正常に受信すると、エラー状況はクリアされます。 sendData 呼び出しによるデータ収集要求に応答してもデータ行を取得できない場合は、「エラー・コード」列に NO_INSTANCES_RETURNED が表示されます。

次の表に、特定の状態で表示されると想定できる、エージェント内部のエラー・コードをいくつか示します。

表 *287.* エージェントの内部エラー・コード

エラー・コード	**説明**
NO_ERROR	現在のところ、属性グループに問題はありません。
NO_INSTANCES_RETURNED	Java アプリケーションはデータ収集要求に応答しましたが、データは提供しませんでした。 データが提供されなくても、エラーではありません。これは通常、その属性グループによってモニターされているリソースのインスタンスがないことを示しています。
OBJECT_NOT_FOUND	エージェントは、クライアント API によって登録されていない属性グループに関するデータを収集しようとしました。このエラーは、アプリケーションを開始できなかったか、またはエージェントがデータを収集しようとしたときに属性グループの登録が実行されなかったことを示している可能性があります。
OBJECT_CURRENTLY_UNAVAILABLE	アプリケーションは、エラー・コードのグローバル・リストに定義されていないエラー・コードをエージェントに送信しました。
GENERAL_ERROR	アプリケーションからのデータ収集中に問題が発生しました。アプリケーションがタイムアウト間隔内に要求に応答しなかったことが原因である可能性があります。詳しくは、エージェント・トレース・ログを参照してください。 アプリケーションはエラー・コードに GENERAL_ERROR を指定することもありますが、さらに詳細なエラー・コードが定義されている場合は、そのコードを参照することをお勧めします。

エージェントへの変更

エージェントに特定の変更を加えると、Java アプリケーションでも、それに対応する変更を行う必要が生じます。構造上の変更が複雑な場合は、エージェントを保存する前に、任意またはすべての Java ソース・ファイルを削除することができます。行ったカスタマイズを適用せずにやり直す場合も、これらのファイルを削除することができます。

次の表では、エージェントを保存する際に、Agent Builder で特定の変更を行うと必要になる、Java アプリケーション・ソース・ファイルへの変更について説明します。

表 *288. Java* ソースへの変更が必要となるエージェントへの変更

エージェントの変更	**Agent Builder の動作**	**Java ソースに必要な手動による変更**
メイン・クラス・パッケージ名の変更	• 新しいパッケージ構造内ですべてのクラスを生成する。 • 古いパッケージからすべてのヘルパー・クラスを削除する。	• メイン・クラスと属性グループ・クラスの内容を、古いパッケージのクラスから新しいパッケージのクラスに移植する。 • マイグレーションの完了後に古いパッケージからクラスを削除する。
メイン・クラス名の変更	• 新しいメイン・クラスを作成する。 • 古いメイン・ヘルパー・クラスを削除する。	• メイン・クラスの内容を新しいクラスに移植する。 • 属性グループ・クラスのクラス名への参照を更新する。
Java API 属性グループの追加	• 新しい属性グループのクラスを作成する。 • メイン・ヘルパー・クラスで新しい属性グループの登録を追加する。	属性グループ・クラスでサンプル・コードにカスタム・ロジックを上書きする。
Java API 属性グループの削除	メイン・ヘルパー・クラスから登録を削除する。	• 属性グループ・クラスを削除する、またはカスタマイズしたロジックを他のクラスに移植する。 • 属性グループ・ヘルパー・クラスを削除する。
Java API 属性グループの名前変更	• 属性グループの新規名に応じたクラスを作成する。 • メイン・ヘルパー・クラスで名前変更した属性グループの登録を更新する。	• 古い名前の属性グループ・クラスのカスタマイズされたロジックを、新しい名前の属性グループ・クラスに移植する。 • 古い名前の属性グループ・クラスを削除する。 • 古い名前の属性グループ・ヘルパー・クラスを削除する。
Java API 属性グループへの属性の追加	属性グルー・ヘルパー・クラス内の属性内部クラスを更新する。	属性グループ・クラス内の新しい属性に関するデータを収集する。
Java API 属性グループからの属性の削除	属性グループ・ヘルパー・クラス内の属性クラスを更新する。	属性グループ・クラス内の以前の属性に関するデータ収集を削除する。
Java API 属性グループ内の属性の名前変更	属性グループ・ヘルパー・クラス内の属性クラスの属性名を更新する。	属性クラス内の属性名への参照を更新する (多くの場合、定位置引数を持つ属性コンストラクターが使用されるため、参照はありません)。
Java API 属性グループ内の属性の再配列	属性グループ・ヘルパー・クラス内の属性クラスで属性の順序を更新する。	属性コンストラクターの呼び出しでの引数の順序を更新する。

Eclipse のリファクタリング (名前変更) アクションを使用する場合は、上の表に示した変更の一部を簡素化できます。変更したエージェントを保存する前に、影響を受けるすべての名前 (ヘルパー・クラス名を含む) でこのアクションを使用してください。

Java API の使用

Java API は、生成された Java アプリケーション全体でエージェントとの通信用に使用されます。 多くの場合、Java API との直接対話は、既存のメソッド呼び出しのパラメーターを変更する場合に限られます。例えば、GENERAL_ERROR から通知されたエラー・コードを、エージェントで定義されたエラー・コードに変更する場合などです。

Java API を使用してより大規模なコーディングを行う場合は、Eclipse テキスト・エディターから Javadoc を表示することができます。以下の手順を実行すると、Java コードの編集中に Javadoc を表示することができます。

1. API でパッケージ、クラス、またはメソッドの名前を強調表示します。
2. **F1** を押して、Eclipse のヘルプ・ビューを開きます。
3. Javadoc リンクを選択してください。

クラスまたはメソッドの名前の上にカーソルを移動することで、Javadoc からの要旨を参照することもできます。API の Javadoc は、Tivoli Monitoring Knowledge Center でも見つけることができます。そこで、Javadoc を参照してください。

Java API のクラスは cpci.jar の中にあります。cpci.jar ファイルは、Java API 属性グループを含むエージェントを作成すると、自動的にプロジェクトの Java ビルド・パスに追加されます。このファイルは、Java API 属性グループを含むエージェントをインポートした場合にも追加されます。また、既存のエージェントに Java API 属性グループを追加した場合にも、このファイルが追加されます。また、cpci.jar は、自動的に Java API 属性グループを含む各エージェントと一緒にパッケージ化され、Java アプリケーションの CLASSPATH に追加されます。

Java API 構成

エージェント内に Java API データ・ソースを定義すると、一部の構成プロパティーが作成されます。

エージェント内に Java API データ・ソースを定義した場合、そのエージェントは Java を使用して Java API サーバーに接続する必要があります。Java 構成プロパティーは自動的にエージェントに追加されます。以下の Java 構成プロパティーは、エージェントのランタイム構成に固有のものです。

表 *289. Java* 構成プロパティー

Name	有効な値	必須	説明
Java ホーム	ディレクトリーの完全修飾パス	いいえ	Java のインストール・ディレクトリーを指す完全修飾パス。
Java トレース・レベル	選択項目	はい	このプロパティーを使用して、Java プロバイダーで使用されるトレース・レベルを指定します。
JVM 引数	ストリング	いいえ	このプロパティーを使用して、Java 仮想マシンの引数のリストを指定します (任意指定)。
外部 JAR のクラスパス	ストリング	いいえ	エージェントには含まれていないが、ランタイム・クライアントの動作に必要な JAR ファイルを含むパス。

これらの構成変数は、Agent Editor の「**ランタイム構成情報**」ページにおいて、「**Java 仮想マシン (JVM) の場合の構成**」および「**Java API の場合の構成**」で使用できます。

Java アプリケーション属性グループのテスト

Agent Builder 内で作成した Java アプリケーション属性グループをテストできます。

始める前に

制約事項 : 他のほとんどの属性グループとは異なり、Java アプリケーション属性グループの作成中に Java アプリケーション属性グループをテストすることはできません。この属性グループのテストは、エージェントに追加してそのエージェントを保存すると実行できます。エージェントを保存すると、この属性グループ用の Java コードが生成されます。

手順

1. エージェントの作成後に、**Agent Editor** の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。

   Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください

   上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「**Java クライアントのテスト**」ウィンドウが表示されます。
2. オプション: テストを開始する前に、環境変数、構成プロパティー、および Java 情報を設定します。詳しくは、1376 ページの『属性グループのテスト』を参照してください。デフォルトの Java ランタイム構成プロパティーについて詳しくは、1334 ページの『Java API 構成』を参照してください。
3. 「**エージェントの開始**」をクリックします。ウィンドウに、エージェントが開始されていることが表示されます。
4. Tivoli Enterprise Portal または SOAP からのエージェント・データの要求をシミュレートするために、「**データの収集**」をクリックします。

   エージェントは、Java クライアントでデータをモニターします。「**Java クライアントのテスト**」ウィンドウは、返されたデータをすべて表示します。
5. オプション: 返されたデータが期待どおりではない場合は、「**結果の確認**」をクリックします。

   「**データ収集状況**」ウィンドウが開き、データに関する詳細情報が表示されます。「データ収集状況」ウィンドウで収集および表示されるデータについては、1421 ページの『「パフォーマンス・オブジェクトの状況」ノード』で説明されています。
6. 「**エージェントの停止**」をクリックしてエージェントを停止します。
7. 「**OK**」または「**キャンセル**」をクリックして「**Java クライアントのテスト**」ウィンドウを終了します。「**OK**」をクリックすると、行った変更がすべて保存されます。

関連概念

1376 ページの『Agent Builder でのエージェントのテスト』
Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

既存のソースからのデータ・セットの作成

データ・セットが少なくとも 1 つ存在する場合、既存のデータ・セットのデータを使用して新規データ・セットを作成することができます。

新規データ・セットを作成するためのオプションは、「**エージェントの初期データ・ソース**」ページと「**データ・ソース・ロケーション**」ページで使用できます。既存のデータ・ソースを使用してデータ・セットを作成するには、以下のようにします。

1. 2 つの既存のデータ・セット (属性グループ) からのデータの結合。詳しくは、1336 ページの『2 つの属性グループの結合』を参照してください。

2. 既存のデータ・セット (属性グループ) からのデータのフィルタリング。詳しくは、1341 ページの『フィルタリング属性グループの作成』を参照してください。

ヒント: 2 つのデータ・セットを結合するオプションは、2 つ以上のデータ・セットが作成された後でのみ使用できます。

2 つの属性グループの結合

2 つの属性グループから 1 つの属性グループを作成します。

このタスクについて

属性グループの結合は、エージェントが 2 種類のデータ・ソースからデータを収集する場合に特に有用です。例えば、エージェントがデータ WMI と Perfmon または SNMP とスクリプトの各データ・ソースを収集する場合などです。それぞれの属性セットを 1 つの Tivoli Enterprise Portal ビューにまとめて使用すると特に有用です。

例えば、属性グループが以下のように定義されているとします。

```
First_Attribute_Group
    index integer
    trafficRate    integer
    errorCount    integer
```

```
Second_Attribute_Group
    index2    integer
    name string
    traffic string
```

一方の定義ではカウンター (Perfmon など) を提供し、もう一方の定義では識別情報を提供します。いずれの属性グループも、単独では役に立ちません。しかし、索引を使用して各属性グループの該当する行を突き合わせることで両方の属性グループを結合すれば、便利な属性グループになります。 この結合した属性グループを使用すると、名前、タイプ、およびメトリックを一緒に表示できます。

これと同じメカニズムを使用して、通常の属性グループから収集した情報にタグを追加します。これにより、問題を検出したときに、イベント・システムで情報の相関関係を簡単に確認できます。例えば、社内のすべてのサーバーを管理するために、共通のデータを収集し、共通のシチュエーションを使用して各サーバーの正常性をモニターすることを計画している企業があるとします。また、この企業では、特定のサーバーでどのようなアプリケーションが稼働しているかを示す詳細情報によって、各サーバーを識別できるようにする必要もあります。各サーバーで使用する値を制御することは必要ですが、アプリケーションごとに別々のエージェントを作成することは望ましくありません。このような制御を実現するには、1 つのエージェントで以下のような追加の属性グループを作成します。

```
Application_Information
    application_type     integer
    application_name    string
    application_group    string
```

この属性グループは、エージェント構成から値を収集するスクリプト属性グループとして定義できます。エージェント・インスタンスごとに別々の値を指定することによって、1 つのエージェントですべてのシステムを管理できるようになります。このアプリケーション情報を必要とするすべてのソース属性グループにその属性グループを結合します。それにより、Tivoli Enterprise Portal でも、シチュエーションでも、イベントでも、ウェアハウスのデータでも 、その情報を利用できます。

2 つの属性グループを結合すると、第 3 の属性グループが 作成されます。この属性グループには、ソース属性グループに含まれる すべての属性が含まれます。

結合操作の結果は、それぞれのソース属性グループでサポートされている 行の数によって異なります。両方の属性グループが 1 行のデータだけを返すように定義されていれば、結果として得られる結合属性グループは 1 行のデータを持つことになります。その 1 行には、両方のソース属性グループのすべての属性が含められます。

表 *290.* ソース属性グループ *1* (単一行)		
Attribute1	**Attribute2**	**Attribute3**
16	何らかのテキスト	35

表 *291.* ソース属性グループ *2* (単一行)			
Attribute4	**Attribute5**	**Attribute6**	**Attribute7**
5001	さらにあるデータ	56	35

表 *292.* 結果の結合						
Attribute1	**Attribute2**	**Attribute3**	**Attribute4**	**Attribute5**	**Attribute6**	**Attribute7**
16	何らかのテキスト	35	5001	さらにあるデータ	56	35

一方のソース属性グループが 1 つの行だけを返すように定義されていて (単一行)、もう一方のソース属性グループが複数の行を返せる (複数行) とします。結果として得られる結合属性グループには、複数行のソース属性グループと同じ数の行が含まれることになります。 単一行の属性グループのデータは、複数行の属性グループの 各行に追加されます。

表 *293.* ソース属性グループ *1* (単一行)		
Attribute1	**Attribute2**	**Attribute3**
16	何らかのテキスト	35

表 *294.* ソース属性グループ *2* (複数行)			
Attribute4	**Attribute5**	**Attribute6**	**Attribute7**
user1	path1	56	35
user2	path2	27	54
user3	path3	44	32

表 *295.* 結果の結合						
Attribute1	**Attribute2**	**Attribute3**	**Attribute4**	**Attribute5**	**Attribute6**	**Attribute7**
16	何らかのテキスト	35	user1	path1	56	35
16	何らかのテキスト	35	user2	path2	27	54
16	何らかのテキスト	35	user3	path3	44	32

最後に、両方のソース属性グループが複数の行を返すように定義されているとします。結合を実行するソース属性グループのそれぞれからの属性を識別する必要があります。結果の属性グループには、最初の属性グループの属性値と 2 番目の属性グループの属性値が一致するデータ行が含まれます。

表 *296.* ソース属性グループ *1* (複数行)		
Attribute1	**Attribute2**	**Attribute3**
16	何らかのテキスト	35

表 *296.* ソース属性グループ *1* (複数行) (続き)		
Attribute1	**Attribute2**	**Attribute3**
27	さらにあるテキスト	54
39	別のストリング	66

表 *297.* ソース属性グループ *2* (複数行)			
Attribute4	**Attribute5**	**Attribute6**	**Attribute7**
user1	path1	56	35
user2	path2	27	54
user3	path3	44	32

表 *298.* 結果の結合 (*Attribute3* と *Attribute7* での結合)						
Attribute1	**Attribute2**	**Attribute3**	**Attribute4**	**Attribute5**	**Attribute6**	**Attribute7**
16	何らかのテキスト	35	user1	path1	56	35
27	さらにあるテキスト	54	user2	path2	27	54

Agent Builder では、ご使用のエージェントに可用性フィルターが 定義されていれば、ユーザー定義の属性グループを「可用性」属性グループに 結合することもできます。「可用性」の属性グループに含まれるデータについて詳しくは、1417 ページの『可用性ノード』を参照してください。

このタイプの属性グループを作成するには、右クリックによってデータ・ソース・ツリーのメニューにアクセスし、「**属性グループの結合**」を選択します。

手順

1. 「**データ・ソース定義**」ページで、結合したい属性グループのいずれかを右クリックして、「**属性グループの結合**」を選択します。

   このオプションが表示されるのは、少なくとも 2 つの属性グループが定義されている場合のみです。 可用性フィルターを定義しておくと、属性グループが定義されているものとみなされます。

   「**属性グループ情報**」ページが表示されます。

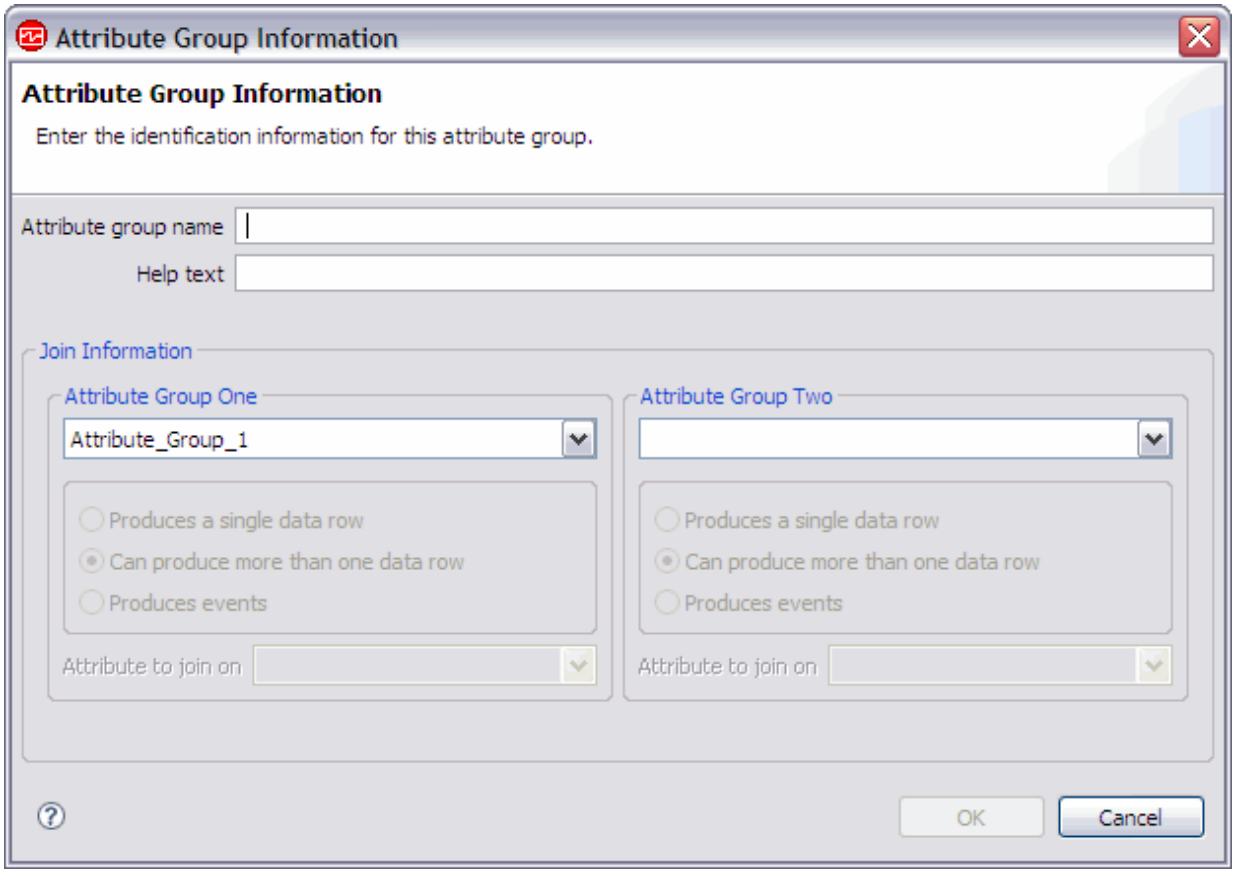


図 *48.* **「属性グループ情報」**ページ「属性グループ情報」ウィンドウ

2. 「**結合情報**」領域で、結合する 2 つの属性グループを選択します。属性グループを選択するには、「**属性グループ 1**」および「**属性グループ 2**」のリストから選択可能なグループを選択します。

   属性グループごとに、「**単一データ行の生成**」または「**複数のデータ行を生成可能**」のいずれかが選択されています。この選択はロックされていて、ソース属性グループがもともとどのように定義されていたかによって決められます。

   注 : 属性グループの結合には以下の制限があります。

   - あるサブノード・タイプの属性グループを 別のサブノード・タイプの属性グループに結合することはできません。
   - イベント属性グループの結合先は、単一行の非イベント属性グループに限ります。

   a) 「**結合する属性**」で両方の属性グループに「**複数のデータ行を生成可能**」が表示されている場合は、属性グループごとに結合を実行する属性を選択します。

   「**属性グループ名**」フィールドおよび「**ヘルプ**」フィールドは、選択した属性グループの情報を使用して自動的に設定されます。必要な場合には、これらの項目を変更できます。
3. 「**OK**」をクリックします。

タスクの結果

作成した結合属性グループが、「**データ・ソース定義**」ページの「**属性グループ情報**」領域に追加されます。

結合された属性グループの属性の操作

結合属性グループの属性を使用すると、それらの属性の操作方法にルールを適用できます。

属性グループの削除

属性グループが結合属性グループで参照されている場合、その結合属性グループも削除するのでない限り、その属性グループを削除することはできません。

属性の削除

属性の親属性グループが結合属性グループで参照されており、以下のいずれかに該当する場合は、 その属性を削除することはできません。

- 属性が結合属性グループ内で結合属性として 定義されている。
- 属性が結合属性グループ内のいずれかの派生属性で使用されている。

結合属性は削除できません。結合属性グループから削除できるのは派生属性だけです (派生属性が追加された場合)。

属性の再配列

結合属性の順序は、ソース属性の順序で 固定されています。結合属性リストを 並べ替えることはできません。並べ替えが可能なのは派生属性だけです (派生属性がある場合)。

エージェントのバージョンが既にコミットされた場合、ソース属性および派生属性の並べ替えや削除はできません。新しいバージョンの エージェントで追加された属性は、ソース属性か派生属性かに関係なく、コミット済みのすべての属性の後に置かれます。詳しくは、1191 ページの『エージェントのバージョンのコミット』を参照してください。

属性の追加

新しい結合属性を明示的に追加することはできません。 明示的に作成できるのは派生属性だけです。

可用性フィルターの削除

「可用性」属性グループが結合属性グループで参照されている場合は、最後の可用性フィルターを削除することはできません。

結合された属性

結合属性に関連した情報を操作します。

手順

- 結合された属性の属性名およびヘルプ・テキストを変更して、ソース属性とは別のものにします。
  a)「**データ・ソース定義**」ページの「**属性グループ情報**」ペインで結合属性グループの属性を選択します。
  b) 新しい名前とヘルプ・テキストを入力します。
- 結合された属性は、「**Tivoli Enterprise Portal の属性表示**」チェック・ボックスを 選択したりクリアしたりすることで、Tivoli Enterprise Portal で表示させたり 非表示にしたりすることができます。このチェック・ボックスは、「**データ・ソース定義**」ページの「**結合属性情報**」セクションにあります。この選択は、Tivoli(r) Enterprise Portal でソース属性を表示するかどうかとは無関係です。
- 属性または属性の組み合わせ (Tivoli Enterprise Portal に表示されるもの) はいずれも、 「**キー属性**」チェック・ボックスを選択することで キー属性のマークを付けることができます。これは、ソース属性グループでその属性がキー属性になっているかどうかとは無関係です。また、この選択は、ソース属性が Tivoli(r) Enterprise Portal で表示されるかどうかとも無関係です。
- 結合された属性の属性タイプ情報はソース属性から引き継がれ、結合された属性で変更することはできません。 Agent Editor の「**結合属性グループ情報**」セクション (1341 ページの図 49) で、「**ソース属性の検索**」をクリックしてソース属性に移動します。

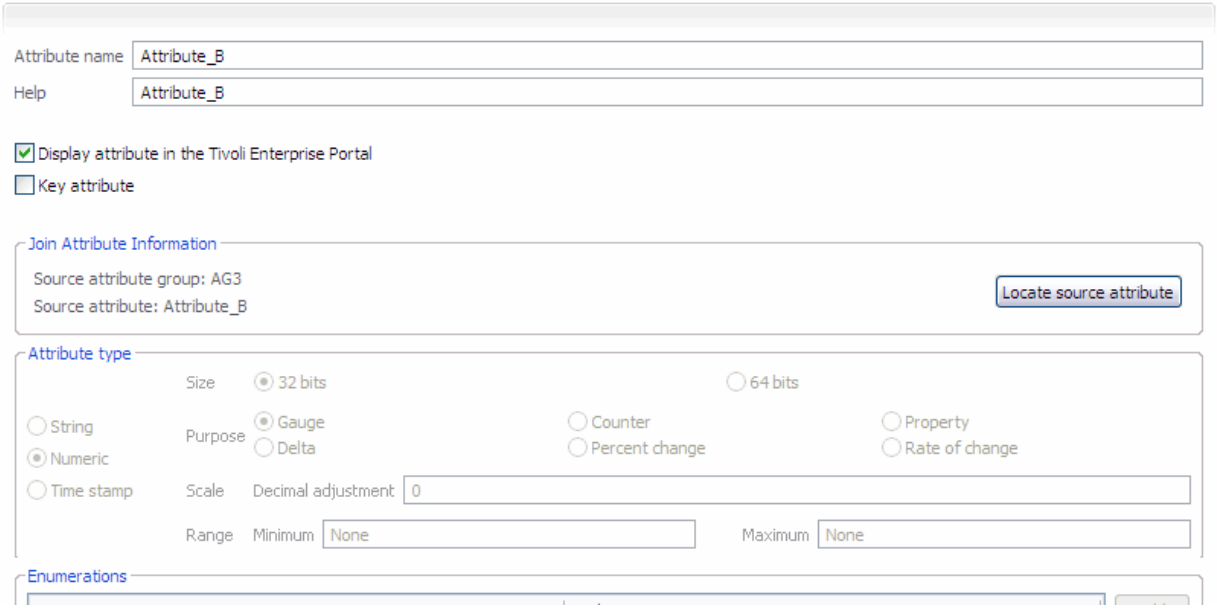


図 *49.* ソース属性情報の検索

ソース属性グループに対する変更は、結合属性にすべて反映されます。ソース属性グループが変更された場合、結合された属性グループでもそれらの属性が自動的に更新されます。この自動更新は、別の属性グループがソース属性グループとして設定された場合にも行われます。ソース属性タイプに対する変更は、結合された属性にコピーされます。ソース属性の名前またはヘルプ・テキストに対する変更は、結合された属性にコピーされます。ただし、結合された属性の名前またはヘルプ・テキストを変更した後は、ソース属性の変更がコピーされなくなります。

フィルタリング属性グループの作成

既存の属性グループ (データ・セット) からデータ行をフィルタリングして、フィルタリング属性グループを作成します。既存のデータ・セットが複数の行を返す場合は、IBM Cloud Application Performance Management で使用される単一行を返すフィルタリング・グループを作成することができます。

このタスクについて

フィルタリング属性グループは、ソース属性グループと同じ列を持ちますが、一部の行を除外することができます。フィルタリング属性グループは、選択式を使用して、含める行を決定します。

Cloud APM の状況情報と要約情報を指定するには、単一行を返すデータ・セットを使用する必要があります。詳しくは、1373 ページの『Cloud APM 用のエージェントの準備』を参照してください。複数の行を返すデータ・セット内にソース情報が存在する場合、単一行を返すフィルタリング属性グループを作成することができます。

例えば、プロセス、Windows サービス、コマンド戻りコードのデータ・ソースの場合、単一の可用性データ・セット内の行として情報が指定されます。選択式の「名前」フィールドを使用して、フィルタリング属性グループを作成することができます。このグループには、必要なアプリケーションの状況が含まれます。このフィルタリング属性グループは、単一行を返すグループとして定義してください。このフィルタリング属性グループは、Cloud APM の要約データ・セットとして使用することができます。

フィルタリング属性グループは、個別のグループに分割したいデータが基本データ・ソース照会から返された場合に使用しても便利です。データ・ソースの例としては Windows パフォーマンス・モニター、SNMP、WMI などが挙げられます。

例えば、データ・ソースが以下のデータを返すとします。

```
Name    Type  Size  Used  Free
Memory  MEM      8     4     4
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

このテーブルは、システム上に存在するストレージについてのレポートであり、メモリーとディスク・スペースの両方が含まれています。場合によっては、このテーブルをメモリーとディスクのそれぞれのテーブルに分割する必要があります。2 つの基本属性グループを作成することにより、テーブルを分割できます。これらの各基本属性グループは同じデータを収集し、不要な行を除外します。しかし、これは最も効率的な方法ではありません。代わりに、メモリーとディスクの両方の使用量データを返す 1 つの基本属性グループを定義します。次に、2 つのフィルタリング属性グループを定義します。どちらでも同じ基本テーブルをソースとして使用します。1 つには Type=="MEM" のフィルターを組み込み、もう 1 つには Type=="DISK" のフィルターを組み込みます。

この例では、Type=="MEM" のフィルタリング属性グループに対して、以下のデータが返されます。

```
Name    Type  Size   Used  Free
Memory  MEM      8      4     4
```

Type=="DISK" のフィルタリング属性グループに対して、以下のデータが返されます。

```
Name    Type  Size  Used  Free
Disk1   DISK  300   200   100
Disk2   DISK  500   100   400
```

注 : イベント・ベースのデータを収集する属性グループを使用してフィルタリング属性グループを作成することはできません。サンプル・データを収集する属性グループのみを使用できます。

手順

1. 「**エージェントの初期データ・ソース**」ページまたは「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**既存のデータ・ソース**」をクリックします。

   注 :

   - 「**エージェントの初期データ・ソース**」ページに移動するには、新規エージェント・ウィザードを使用します。詳しくは、1172 ページの『エージェントの作成』を参照してください。
   - **Agent Editor** の「**データ・ソース定義**」ページでエージェントを右クリックして「**データ・ソースの追加**」を選択することで、「**データ・ソース・ロケーション**」ページに移動できます。
2. 「**データ・ソース**」領域で「**属性グループのデータ行のフィルター処理**」を選択します。
3. 「**次へ**」をクリックします。

   「**フィルター情報**」ページが表示されます。
4. リストから「**ソース属性グループ**」を選択します。
5. 選択した属性グループからのデータをフィルタリングする 「**選択式**」を入力します。
   例えば、上で示した「**フィルター情報**」ページでは、選択式で、Type 属性が "DISK" に等しいデータ行をフィルタリングします。Type 属性が "DISK" と一致しないデータ行は廃棄されます。入力する選択式は、ブール値の結果 (true または false) に評価される必要があります。

   注 : 「**フィルター情報**」ページで、「**編集**」をクリックすると、式エディターを使用して式を入力または変更できます。式エディターについて詳しくは、1202 ページの『式エディター』を参照してください。
6. 「**次へ**」をクリックします。
7. 「**単一データ行の生成**」または「**複数のデータ行を生成可能**」を選択します。

   a) 「**複数のデータ行を生成可能**」を選択した場合は、リストからキー属性 (複数可) を選択します。
8. 「**完了**」をクリックします。

ナビゲーター・グループの作成

IBM Tivoli Monitoring 環境では、ナビゲーター・グループを使用して、複数の関連したデータ・ソース (属性グループ) をまとめてグループ化し、データ・ソースを結合するビューを表示するワークスペースを作成することができます。新規エージェント・ウィザードを使用してエージェントを基本エージェント・レベルで作成する際に、ナビゲーター・グループを作成できます。また、新規エージェント・コンポーネント・ウィザードを使用してサブノードを定義する際に、ナビゲーター・グループを作成することもできます。

このタスクについて

例えば、複数のデータ・ソースからファイル・システム・データを収集することができます。それらのさまざまなデータ・ソースから、すべてのファイル・システム・データのビューを表示するワークスペースを 1 つ作成すると役立ちます。

また、ナビゲーター・グループは、Tivoli Enterprise Portal でデータ・ソースを隠すのにも役立ちます。2 つのデータ・ソースを結合して新規の結合データ・ソースを作成する場合、その 2 つのデータ・ソースから収集されたメトリックが最も役立つ場合があります。その場合、結合データ・ソースで結合されたデータだけを表示することができます。3 つのすべてのデータ・ソースを含むナビゲーター・グループを作成し、結合データ・ソースのみを表示するビューを含むワークスペースを作成できます。 2 つのオリジナル・データ・ソースは、Tivoli Enterprise Portal のビューから事実上隠されます。データ・ソースの結合については、1335 ページの『既存のソースからのデータ・セットの作成』を参照してください。

注 : ナビゲーター・グループでデータ・ソースをグループ化する場合、Tivoli Monitoring によって照会がそのナビゲーター・グループに関連付けられることはありません。データ・ソースを有用な形式で表示するために、ナビゲーター・グループにデフォルトのワークスペースを 定義していると想定されます。

ナビゲーター・グループは基本エージェントまたはサブノードで 定義できます。ナビゲーター・グループに別のナビゲーター・グループを 含めることはできません。

ナビゲーター・グループは IBM Cloud Application Performance Management 環境では機能しません。

手順

1. 以下のいずれかのステップを実行します。
   - 「**エージェント**」ウィザードを使用して新規エージェントを作成する際に、「**エージェントの初期データ・ソース**」ページで、「**データ・カテゴリーのモニター**」領域の「**データ・ソース・グループ**」をクリックします。
   - 既存のエージェントを使用する場合は、Agent Editor で以下のステップを実行します。
     a. 「**データ・ソース**」タブをクリックして、「**データ・ソース定義**」ページを開きます。
     b. エージェントを選択して、「**選択対象に追加**」をクリックします。
     c. 「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**データ・ソース・グループ**」をクリックします。
2. 「**データ・ソース**」領域で「**ナビゲーター・グループ**」をクリックします。
3. 「**次へ**」をクリックします。
4. 「**ナビゲーター・グループ情報**」ページで、ナビゲーター・グループ名と、その名前に関連付けるヘルプ用テキストを入力し、「**次へ**」をクリックします。

   注 : 特定の状況では、Agent Builder によりナビゲーター・グループが自動的に作成されます。予約されているナビゲーター・グループ名は以下のとおりです。
   - 可用性

   .
5. 「**最初のナビゲーター・グループ・データ・ソース**」ページで、新規ナビゲーター・グループのモニター・データの最初のソースを選択します。「**データ・カテゴリーのモニター**」リスト内のカテゴリーと、「**データ・ソース**」リスト内のデータ・ソースをクリックします。次に、「**次へ**」をクリックします。

   ヒント : データ・ソースは通常どおりに作成できます。あるいは、「**既存のデータ・ソース**」をクリックして、作成済みの 1 つ以上のデータ・ソースをナビゲーター・グループに移動することもできます。
6. ナビゲーター・グループ内にデータ・ソースを作成する場合は、「**データ・ソース定義**」ページでナビゲーター・グループを選択し、「**選択対象に追加**」をクリックします。
7. 既存のデータ・ソースをナビゲーター・グループに移動する場合は、「**データ・ソース定義**」ページでナビゲーター・グループを選択し、「**選択対象に追加**」をクリックし、「**ナビゲーター・グループ・データ・ソース**」ページで「**既存のデータ・ソース**」を選択します。「**現在定義されているデータ・ソース**」ページで、データ・ソースを選択します。
8. ナビゲーター・グループからデータ・ソースを削除する場合は、「**データ・ソース定義**」ページで、以下の手順のいずれかを行います。

- データ・ソースを選択してデータ・ソース・ツリーのルートにドラッグする。
- データ・ソースを選択して、「**削除**」をクリックする。

9. ナビゲーター・グループを作成する場合は、「**データ・ソース定義**」ページで、以下の手順のいずれかを行います。
   - 「**エージェントに追加**」をクリックする。
   - サブノードを選択して、「**選択対象に追加**」をクリックする。

サブノードの使用

サブノードを使用して、単一のエージェント・インスタンスから複数のアプリケーション・コンポーネントをモニターできます。

サブノードを使用すれば、以下のタスクを実現する単一エージェントを作成できます。

- ソフトウェア・サーバー・インスタンスごとに 1 つずつ個別のエージェント・インスタンスを使用することなく、システムで実行中のソフトウェア・サーバーの各インスタンスをモニターする。
- リモート・システムごとに 1 つずつ 個別のエージェント・インスタンスを使用せず、複数のリモート・システムをモニターする。
- 複数のエージェントを作成してデプロイするのではなく、1 つのエージェントから複数の異なるタイプのリソースをモニターする。
- IBM Tivoli Monitoring で、Tivoli Enterprise Portal の物理ナビゲーション・ツリーに追加レベルを表示して、その他のグループ化とカスタマイズを可能にする。また、管理対象システム・グループを定義して、シチュエーションの細分度レベルを追加する。
- IBM Cloud Application Performance Management で、複数の異なるリソースを指定し、個別の要約ダッシュボードと詳細ダッシュボードを表示する。(サブノード・リソースは、エージェント・リソースのピアとして表示することも、エージェント・リソースのサブコンポーネントとして表示することもできます。これらのリソースは、個別にアプリケーションに含めることができます)。

Agent Builder でサブノード・タイプを作成することができます。各タイプは、エージェントがモニターできる別のタイプのリソースに対応している必要があります。データ・ソースおよびデータ・セットを、特定のモニター対象リソースのサブノード・タイプに追加します。

エージェントをモニター対象ホストにデプロイし、これを構成する際には、サブノード・タイプごとに 1 つ以上のインスタンスを作成することができます。サブノードの各インスタンスは、サーバー、リモート・システム、その他サブノード・タイプをモニターするように設計されているあらゆるリソースのインスタンスに対応している必要があります。単一のサブノード・タイプのすべてのサブノード・インスタンスは、同じ形式の属性グループとワークスペースを持っています。ただし、各サブノード・インスタンスには、モニター中の特定のリソースのデータが存在します。

モニター対象ホスト上のエージェントを構成する際に、サブノード・インスタンスの数を決定することができます。一部の構成データはエージェント全体に適用できますが、単一のサブノード・インスタンスに適用される構成データもあります。各サブノード・インスタンスが同じリソースをモニターしたり、まったく同じデータを表示したりすることがないよう、各サブノードについて、他のサブノードとは異なる構成にしてください。

IBM Tivoli Monitoring 環境では、サブノード・インスタンスは、Tivoli Enterprise Portal のナビゲーション物理ビューのエージェント内に表示されます。ワークスペースには、サブノード・インスタンスによって作成されたデータが表示され、サブノードの 1 つ以上のインスタンスにシチュエーションを配布することができます。サブノードのすべてのインスタンスを 含む管理対象システム・リストは、 エージェント用に作成される管理対象システム・リストと同様、自動的に作成されます。

IBM Cloud Application Performance Management 環境では、エージェント・インスタンスとサブノード・インスタンスの両方をモニター対象リソースとして表示することができます。 各サブノード・インスタンスは、個別のリソースになります。詳しくは、1349 ページの『IBM Cloud Application Performance Management のサブノード』を参照してください。

Agent Builder によって作成されるエージェントは、構成値を基にサブノード・インスタンスを作成するため、これらのサブノードの存続期間はエージェントと同じになります。エージェントで実行されるハート

ビートは 1 つだけで、サブノードごとに個別のハートビートがあるわけではありません。そのため、サブノードを使用することにより、モニター環境の潜在的なスケールを大幅に増やすことができます。あるいは、複数のエージェント・インスタンスを使用して、IBM Tivoli Monitoring 環境または IBM Cloud Application Performance Management 環境の潜在的なスケールを制限することもできます。

サブノードを追加または削除するには、エージェントを再構成する必要があります。エージェントを再構成するには、すべてのサブノードを含めてエージェントを停止し、再始動する必要があります。エージェントは、マルチインスタンス・エージェントとして定義することができます。マルチインスタンス・エージェントの場合、1 つのインスタンスだけを開始して停止し、残りのインスタンスはそのまま稼働させておくことができます。

エージェントは、サブノード内のデータ・セットとともに、サブノード外に存在するエージェント・レベルのデータ・セットも定義することができます。

Tivoli Enterprise Portal のナビゲーター・ツリーでは、エージェント名の下にサブノード・タイプが表示され、サブノード・タイプの下にサブノード・インスタンスが表示されます。サブノードは、 エージェントと同様に管理対象システム名 (MSN) (例えば `94:Hill.cmn`) で識別されます。

例えば、1345 ページの図 50 のナビゲーター・ツリーで、エージェント「**Watching Over Our Friends**」には、3 つのリソース (**Boarder**、**Common Area**、**Kennel Run**) と 2 つのサブノード・タイプ (**Common Area** と **Kennel Run**) があります。この リソースのうち 2 つ (**Common Area** と **Kennel Run**) には、サブノード・タイプが 定義されています。3 つ目のリソース (**Boarder**) は基本エージェント・レベルで 表内の単一行で表されるため、これにはサブノードは不要です。 Common Area サブノード・タイプには 3 つのサブノード・インスタンス (`94:Hill:cmn`、`94:Meadow:cmn`、 `94:Tree:cmn`) があり、Kennel 内の 3 つの Common Area を表します。 Kennel Run サブノード・タイプには 4 つのサブノード・インスタンス (`94:system1:run`、`94:system2:run`、`94:system4:run`、 `94:system5:run`) があり、4 つの Kennel Run を表します。

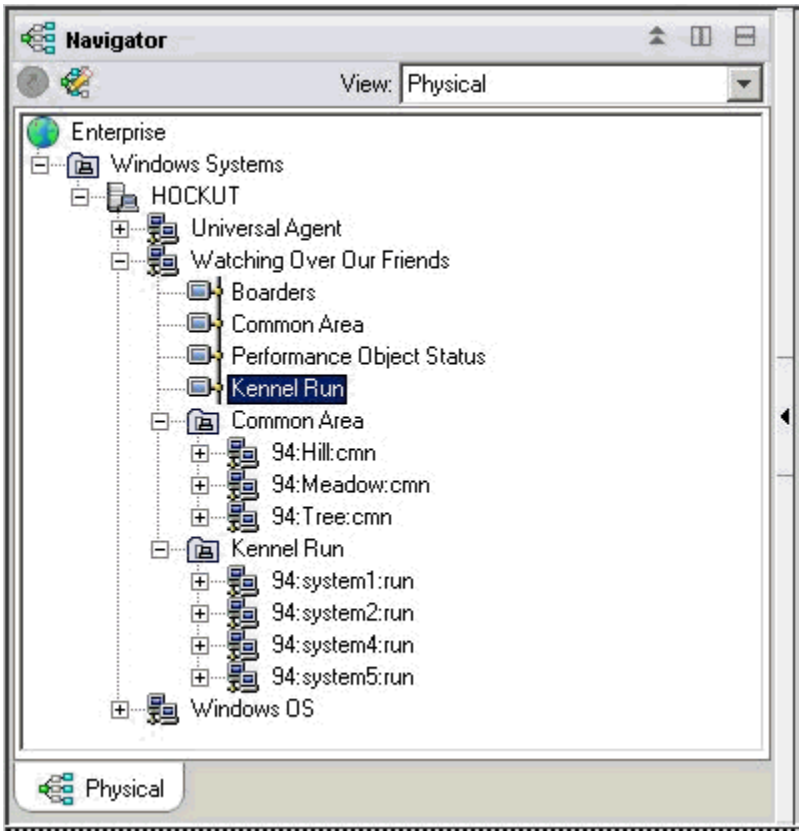


図 *50.* ナビゲーター・ツリー内のサブノード

単一のエージェントは、以下の 2 つの方法でサブノードを使用することができます。

- エージェントは同じタイプの異なるサブノードを持つことができる。
- エージェントは異なるタイプのサブノードを持つことができる。

ソースが異なる同一データのサブノード

同じタイプのサブノードを 使用して、モニター対象リソース・タイプの複数インスタンスを表すことができます。同じタイプの各サブノードには、同じ属性グループと、特定のモニター対象リソース・インスタンスの正しい値が含まれています。サブノードの数は、エージェント構成によって異なります。1346 ページの図 51 の例では、異なるシステムをモニターしています。

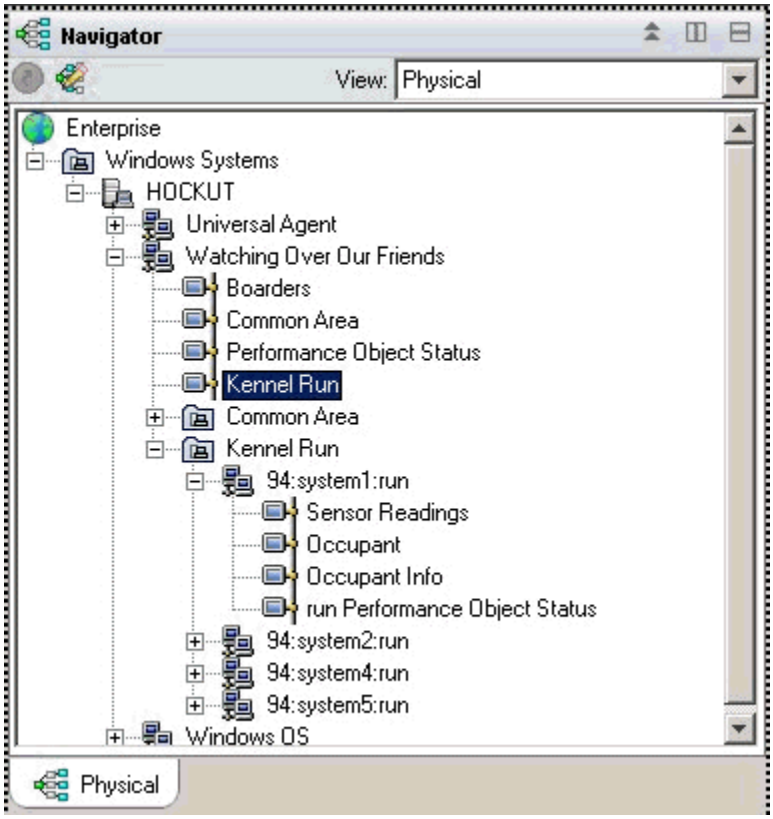


図 *51.* 異なるシステムをモニターするサブノード

複数のデータ・タイプのサブノード

1 つのエージェントが複数のタイプのモニター対象リソースをモニターする場合は、リソース・タイプごとにサブノード・タイプを作成することができます。各サブノードには、そのサブノード・タイプで定義されている情報が含まれます。以下の例は、2 つのサブノード・タイプを示しています。各タイプは、異なるタイプのリソースをモニターしており、リソースごとに異なるタイプのデータが提供されます。

- Common Area
- Kennel Run

1347 ページの図 52 のエージェントは、サブノード・タイプごとに 1 つのコピーを実行します。特定のエージェントが定義済みエージェントのサブセットを作成することがあります。 サブノードを使用することで、Tivoli Monitoring V5 のプロファイルと同様の機能を実現できます。

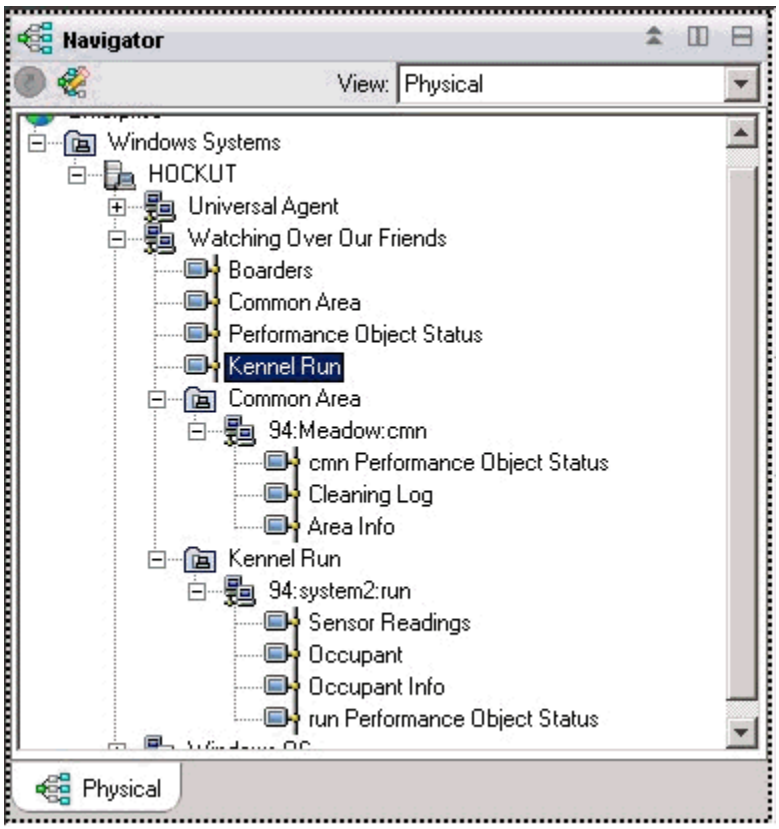


図 *52.* ナビゲーター・ツリーのサブノード・タイプ

サブノードの両方の使用方法を同じエージェントで使用できます。各タイプは複数のサブノード・インスタンスを持つことができます。

1347 ページの図 52 は、2 つのタイプのリソースである Common Area と Kennel Run をモニターする 2 つのタイプのサブノードを示しています。さらに、各タイプにはいくつかのサブノードが定義されています。Common Area タイプのサブノードは 3 つあります。 これらのサブノードは、Meadow、Hill、 および Tree という ID を持っています。また、Kennel タイプのサブノードが 4 つあり、各サブノードは Kennel Run 専用の別のシステムからデータを収集します。 これらのサブノードは system1、system2、system4、 および system5 という ID を持っています。

注 : サブノード ID の最初の 24 文字は、IBM Tivoli Monitoring インストール環境内の該当するサブノード・タイプのすべてのインスタンスで固有である必要があります。

サブノード内のデータ・プロバイダー

サブノードには、異なるデータ・プロバイダー・タイプからのデータを混合して含めることができます。サブノードでは、以下のデータ・プロバイダーを含め、現在のほとんどの Agent Builder データ・プロバイダーを使用することができます。

- WMI
- Perfmon
- Windows イベント・ログ
- SNMP
- SNMP イベント
- JMX
- ICMP ping
- スクリプト

- ログ
- CIM
- JDBC
- HTTP
- SOAP
- ソケット
- Java API

サブノードには、同じサブノードの 2 つの属性グループからのデータ、 またはエージェント・レベルの属性グループからのデータを結合する 結合データ・ソースを含めることもできます。

サブノードの状況

サブノード・エージェントの状況を判別するには、2 つの方法があります。最初の方法では、パフォーマンス・オブジェクト状況属性グループに表示されるデータを調べます。この属性グループは、エージェント内にある同じレベルの他の各属性グループの状況を表示します。エージェント・レベルの「パフォーマンス・オブジェクト状況」属性グループは、エージェント・レベルの他の属性グループの収集状況を表示します。各サブノード内の「パフォーマンス・オブジェクト状況」属性グループは、そのサブノード内にある属性グループの収集状況を表示します。

また、Agent Builder はサブノード・タイプごとに 1 つの属性グループを作成します。この属性グループは、そのタイプの構成済みサブノードごとに 1 つの行を表示します。1348 ページの図 53 の例では、4 つのサブノードが稼働してデータを収集しています。

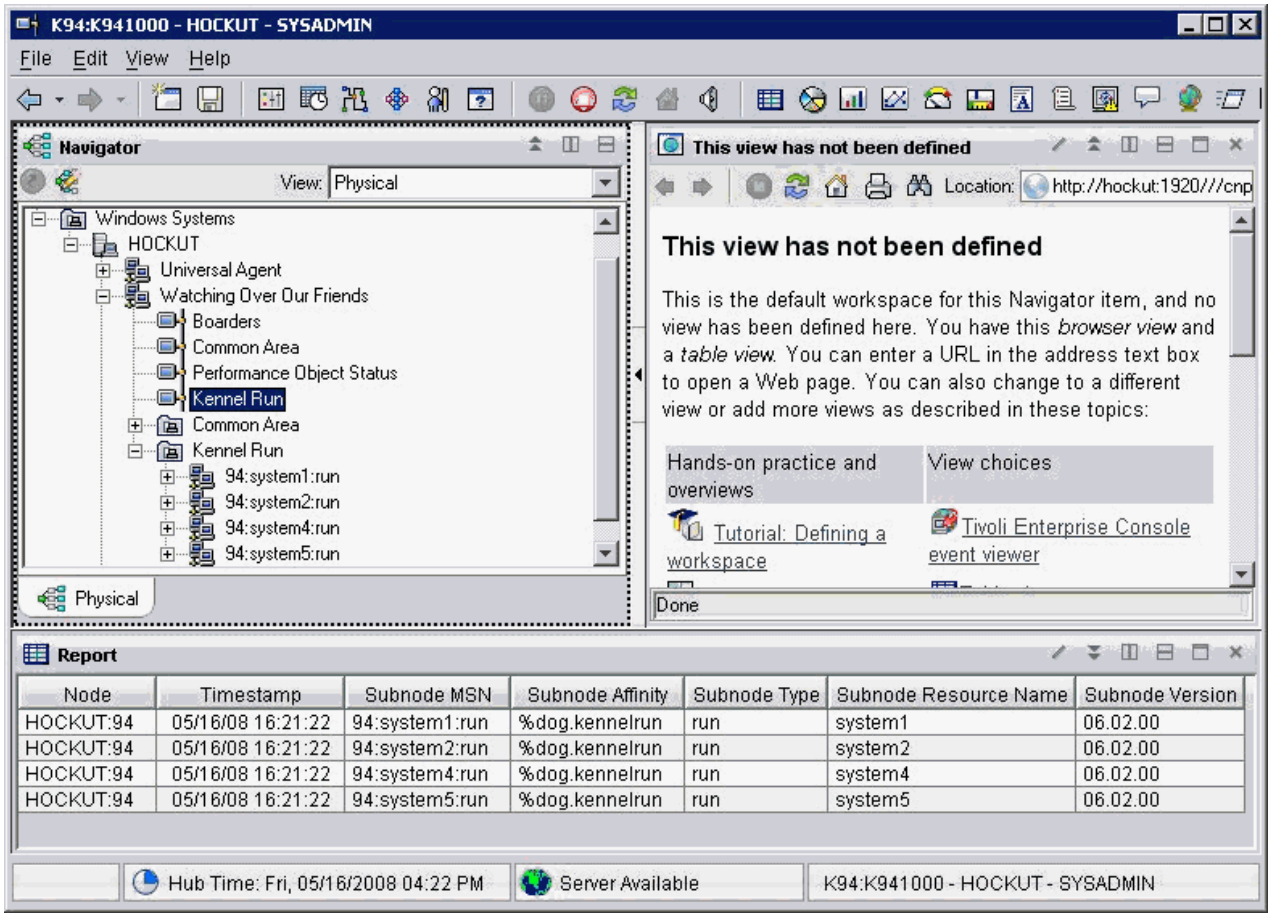


図 *53.* 同じサブノード・タイプの複数のサブノード・インスタンスのモニター

IBM Tivoli Monitoring 環境では、「**パフォーマンス・オブジェクト状況**」サブノードは、ナビゲーター・ツリーに表示されるデータを格納します。また、このサブノードは、他のデータ収集の状況をモニターするシチュエーションを持つことができます。

IBM Cloud Application Performance Management 環境では、「**パフォーマンス・オブジェクト状況**」データをモニターするためのしきい値を作成することができます。

1349 ページの図 54 の例は、データ収集が失敗した (スクリプト・シェル・コマンドが見つからなかった) 場合を示しています。一般に、NO_ERROR 以外の値は、問題があることを示しています。表には、サブノードで定義されているデータ・コレクターごとに 1 つの行があります。

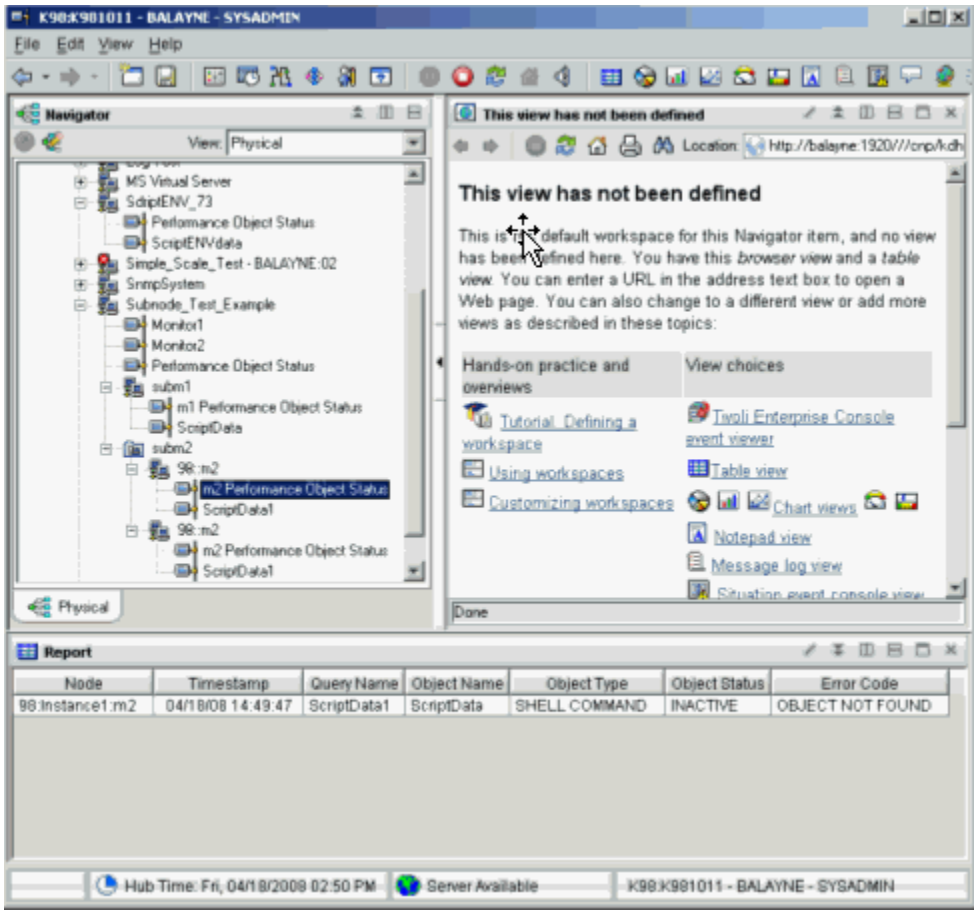


図 *54. 例:* サブノードでのデータ収集

IBM Cloud Application Performance Management のサブノード

IBM Cloud Application Performance Management では、エージェント・インスタンスとサブノード・インスタンスのどちらか 1 つだけをモニター対象リソースとして定義することも、両方をモニター対象リソースとして定義することもできます。各リソースが、要約ダッシュボードに対応します。

サブノード・ダッシュボードでエージェント・レベルのデータを表示することはできません。この環境でエージェント・レベルのデータを表示するには、エージェントの要約ダッシュボードを定義してください。

選択した設定に応じて、エージェントとサブノードのリソースを 階層的な差異のない同じレベルで表示したり、サブノードのリソースをエージェントのリソースの 子としてリスト表示したりすることができます。

エージェントとサブノードのリソースの 構成方法については、1373 ページの『Cloud APM 用のエージェントの準備』を参照してください。

サブノードの作成

エージェントを作成または編集する際に、サブノードを作成することができます。

手順

1. 以下のいずれかのステップを実行します。
   - 「**エージェント**」ウィザードを使用して新規エージェントを作成する際に、「**エージェントの初期データ・ソース**」ページで、「**データ・カテゴリーのモニター**」領域の「**データ・ソース・グループ**」をクリックします。
   - 既存のエージェントを使用する場合は、Agent Editor で以下のステップを実行します。
     a. 「**データ・ソース**」タブをクリックして、「**データ・ソース定義**」ページを開きます。
     b. エージェントを選択して、「**選択対象に追加**」をクリックします。
     c. 「**データ・ソース・ロケーション**」ページで、「**データ・カテゴリーのモニター**」領域の「**データ・ソース・グループ**」をクリックします。
2. 「**データ・ソース**」領域で「**サブノード定義**」をクリックします。
3. 「**次へ**」をクリックします。
4. 以下のように「**サブノード情報**」ページに入力して、新規のサブノードを定義します。
   a) 「**名前**」フィールドに、作成するサブノードの名前を入力します。
   b) 「**タイプ**」フィールドに、作成するサブノードのタイプを識別する 1 文字から 3 文字の数字、文字、またはその両方を入力します。
   c) 「**説明**」フィールドに、作成するサブノードの説明を入力します。
   d) 「**このタイプのサブノードに対応するノード属性グループの表示**」チェック・ボックスをクリックして、可用性属性グループの表示/非表示を選択します。この属性グループについて詳しくは、1417 ページの『可用性ノード』を参照してください。
   e) 「**次へ**」をクリックします。
5. 「**初期サブノード・データ・ソース**」ページに入力して、新規サブノードの最初の項目としてデータ・ソースを選択します。「**データ・カテゴリーのモニター**」リスト内のカテゴリーと、「**データ・ソース**」リスト内のデータ・ソースをクリックします。次に、「**次へ**」をクリックします。

   ヒント: データ・ソースは通常どおりに作成できます。あるいは、作成済みの 1 つ以上のデータ・ソースをナビゲーター・グループに移動することもできます。データ・ソースを移動するには、「**既存のデータ・ソース**」をクリックし、「**現在定義されているデータ・ソース**」ページでデータ・ソースを選択します。

   重要: プロセス、Windows サービス、コマンド戻りコードのデータ・ソースをサブノードに含めることはできません。回避策として、必要なプロセス情報またはサービス情報を判別するスクリプトを作成し、スクリプトの出力データ・ソースを使用してください。
6. エージェントにカスタム構成プロパティーが含まれている場合、または選択したデータ・ソースを構成する必要がある場合は、「**サブノード構成のオーバーライド**」ページを使用して、構成プロパティーを選択します。

   「**サブノード構成のオーバーライド**」ページで、エージェント・レベルのサブノードに必要な構成プロパティーを選択します。次に、サブノードごとに変更したい構成プロパティーを選択します。

   1351 ページの『サブノードの構成』の説明に従って、「**移動**」、「**コピー**」、および「**削除**」を使用して、構成プロパティーを指定します。
7. 「**次へ**」をクリックします。
   「**データ・ソース定義**」ページが表示されます。

サブノード構成

サブノード・タイプを定義すると、そのサブノードに対して単一の構成セクションが定義されます。

サブノード構成セクションは、以下に示すいくつかの点で、他の構成セクションと異なります。

- サブノード・セクション内のプロパティーのセットは複製できるので、複数のプロパティー・セットが存在します。各プロパティー・セットは、独自のセクションを形成します。 セクションのレイアウトはすべて同一ですが、各セクションには異なる値を入力できます。

  これとは対照的に、(エージェント・レベルのセクションと呼ばれる) 他のセクションのプロパティーは、ランタイム構成中に一度しか表示されません。それらのプロパティーはサブセクションを形成せず、複製または削除できません。

  GUI およびコマンド行からのサブノードの 構成の例については、1354 ページの『サブノード構成の例』を参照してください。
- ランタイム構成時に作成されるサブノード・セクションの各コピーに対して、エージェントは個別のサブノード・インスタンスを作成します。それらのサブノード・インスタンスはすべて同じタイプです。
- サブノード・セクション内のプロパティー名には、エージェント・レベルのセクション内のプロパティー名の複製を指定できます。名前の重複が発生した場合には、サブノードのプロパティー 値によってエージェント・レベルのプロパティー値がオーバーライドされます。
- IBM Tivoli Monitoring V6.2.1 以降のサブノード・セクションでは、同じタイプのサブノードのすべてのインスタンスに適用されるデフォルトのプロパティー 値を持つことができます。この機能により、以下のように、単一のプロパティー値を 3 つのレベルで検索することができます。

  1. エージェントはサブノード ・インスタンスのサブセクションからプロパティー 値を取得します。
  2. 値がサブノード・インスタンス・レベルで構成されていない場合、プロパティー値はサブノードのデフォルト・レベルから取得されます。
  3. それら 2 つのレベルのいずれでも値が構成されていない場合、プロパティー値はエージェント・レベルのセクションから取得されます。

GUI およびコマンド行からのサブノードの 構成の例については、1354 ページの『サブノード構成の例』を参照してください。

サブノードの構成
「**サブノード構成のオーバーライド**」ページを使用して、サブノード・データ・ソースを構成します。

始める前に

1350 ページの『サブノードの作成』の手順を使用して、サブノードを作成します。

このタスクについて

サブノードにデータ・ソースを追加する際に、データ・ソースで構成が必要な場合は、「**サブノード構成のオーバーライド**」ページが表示されます。このページには、カスタム構成プロパティーと、該当のサブノード・タイプに適用できる他の構成プロパティーが表示されます。

手順

- 「**サブノード構成のオーバーライド**」ウィンドウで、エージェント・レベルのサブノードに必要な構成プロパティーを選択します。また、サブノードごとに変更したい構成プロパティーも選択します。
- 「**コピー >>**」を使用して構成プロパティーをコピーし、その構成プロパティーがエージェント・レベルとサブノード・レベルの両方に存在するようにします。

  エージェントは最初にサブノード・レベルの値を探し、その値が見つからなかった場合は、エージェント・レベルで探します。両方のレベルに存在するプロパティーが必須プロパティーの場合、そのプロパティーはエージェント・レベルでのみ必須となり、サブノード・レベルではオプションとなります。
- 「**移動 >>**」を使用して、プロパティーをエージェント・レベルからサブノード・レベルに移動します。エージェント・レベルのデータ・ソースまたは別のタイプのサブノードで 必要となるプロパティーの場合、「**移動 >>**」は使用できません。
- 「**削除**」を使用して、2 つのリストのいずれか一方を削除します。プロパティーを削除できるのは、そのプロパティーがエージェント・レベルとサブノード・レベルの両方にリストされている場合のみです。この機能を使用してプロパティーを完全に削除することはできません。

- 「**<< コピー**」を使用して、プロパティーをサブノード ・レベルからエージェント・レベルにコピーします。
- 「**<< 移動**」を使用して、プロパティーをサブノード ・レベルからエージェント・レベルに移動します。

次のタスク

Agent Editor を使用して、既存のサブノードの 構成を変更できます。

サブノード構成のオーバーライド
「サブノード 構成のオーバーライド」を使用すると、エージェント構成プロパティーをサブノード 固有のプロパティーでオーバーライドできます。

1351 ページの『サブノードの 構成』の手順では、自動的に生成されたプロパティーのサブノード 構成を管理する方法が説明されています。 カスタム構成プロパティーの管理も同様の方法で行います。定義されているカスタム構成プロパティーはすべて「**サブノード 構成のオーバーライド**」ウィンドウに表示されます。

カスタム・プロパティーをサブノード ・レベルからエージェント・レベルにコピーまたは移動する場合は、そのプロパティーを配置するセクションを指定するようプロンプトで要求されます。既存のカスタム・セクションを選択するか、または新規のカスタム・セクションの名前を入力します。

サブノード構成プロパティーの選択

サブノードがない場合、データ・ソースのすべてのインスタンスは構成パラメーターを共有します。例えば、すべての SNMP 属性グループは同じコミュニティー名を使用して同じホストに接続します。サブノードが存在し、SNMP_HOST プロパティーがサブノード ・レベルに配置されている場合、サブノードの各インスタンスは異なるホストに接続することができます。

サブノード・レベルでオーバーライドするプロパティーを 選択することは、エージェントの開発時の重要な考慮事項です。選択するプロパティーが多すぎると、サブノード 構成セクションが複雑になり、管理が難しくなります。 選択するプロパティーが少なすぎると、サブノード間で異なるプロパティーを使用したい場合に、エージェント機能が制限されることがあります。

以下のプロパティーは、サブノード・レベルにコピーすることはできません。すべてのサブノード 内のすべての属性グループと、基本エージェント内のすべての属性グループは、同じ SNMP バージョンと JMX 接続タイプを使用する必要があります。

- SNMP バージョン
- JMX MBean サーバー接続タイプ
- Java ホーム
- Java トレース・レベル
- JVM 引数
- 外部 JAR ファイルのクラスパス
- ソケット・データ・ソース・ポート番号
- JMX または JDBC のクラスパス設定

拡張サブノード構成
拡張サブノード 構成を使用して、サブノードのエージェント 構成プロパティーをオーバーライドします。

このタスクについて

IBM Tivoli Monitoring V6.2.1 以降のエージェントには、サブノード ・インスタンス内の任意のエージェント・レベルの構成セクションからプロパティーをオーバーライドできるオプションがあります 。「**サブノード構成のオーバーライド**」ページには、「**任意のサブノードでの任意の構成プロパティーのオーバーライドを許可します**」というラベルのついたチェック・ボックスがあります。詳しくは、1352 ページの『サブノード構成のオーバーライド』 を参照してください。 このオプションを有効にするには、エージェントに命名するときに (1172 ページの『エージェントの命名および構成』) 「**最低 ITM バージョン**」として **6.2.1** を選択する必要があります。このオプションを選択した場合、各サブノード・インスタンスは、 エージェン

ト・レベルの任意の構成セクションにある任意のプロパティーを オーバーライドできます。ただし、このプロパティーをオーバーライドできるのは GUI からだけです。**itmcmd** コマンド行からはオーバーライドできません。

手順

「**任意のサブノードでの任意の構成プロパティーのオーバーライドを 許可します**」オプションを指定すると、リストを含む「**拡張**」フィールドが、各サブノード構成パネルに表示されます。「**拡張**」フィールドで最初に選択されている内容は、「**値をオーバーライドするセクションを 選択**」という 簡単な指示です。

- リストをクリックすると、構成プロパティーを含む非サブノード・セクションがすべてリストされます。
- セクションを選択します。

  そのセクションのプロパティーがサブノード・パネルに一時的に追加されます。プロパティーの値を変更すると、サブノードに対して定義されている一連のプロパティーにその値が 追加されます。サブノード内のデータ・ソースは、サブノード内の プロパティー値を検索してからエージェント・レベルのセクションを 参照します。.

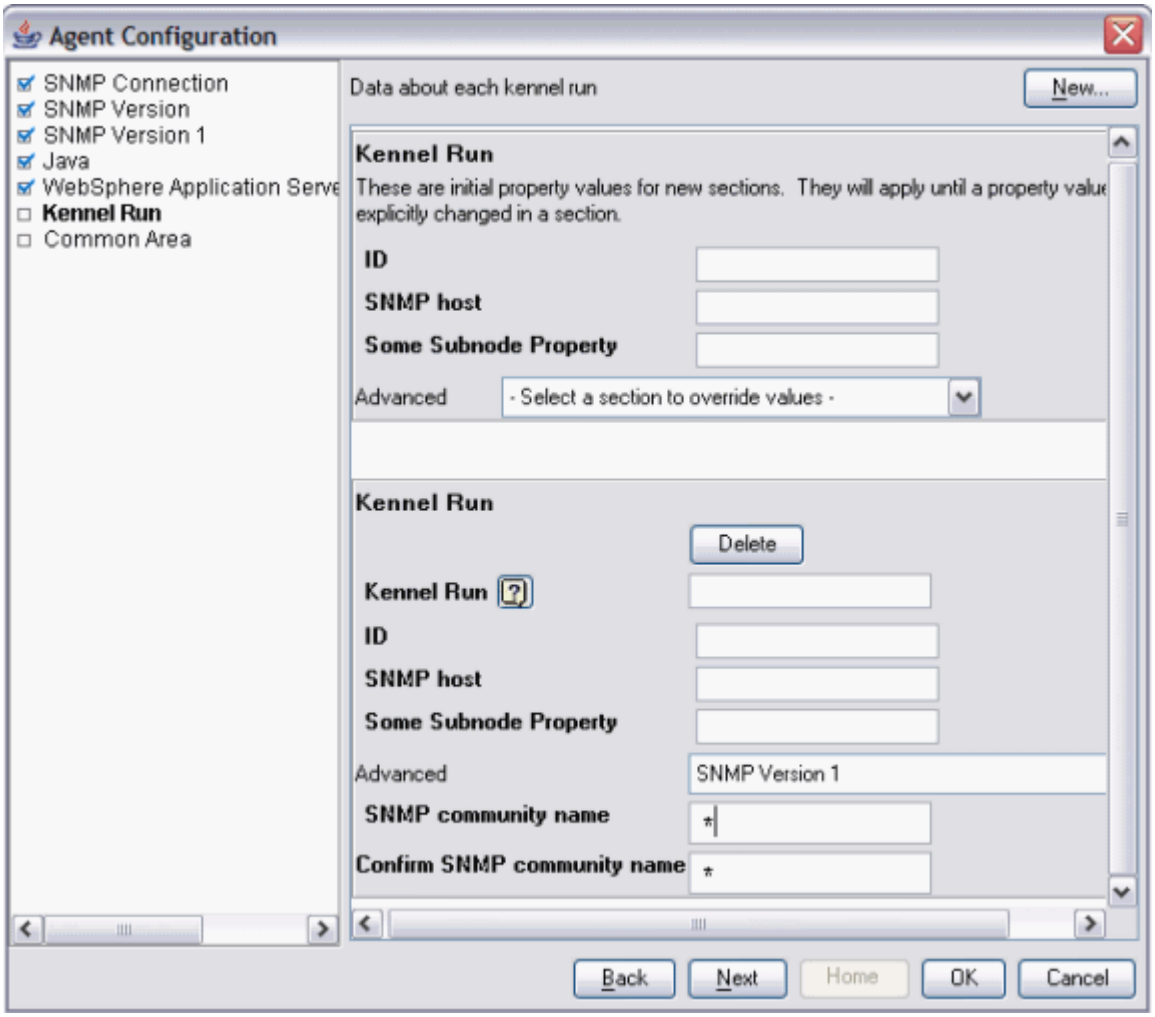


図 *55. SNMP* バージョン *1* のプロパティーを展開

以下の追加情報は、エージェント・レベル・セクションのプロパティーの指定変更に適用されます。

– 「**拡張**」リストでエージェント・レベルのセクションを選択した場合、 サブノード・セクションにコピーされたプロパティーは表示されません。例えば 1353 ページの図 55 では、「**SNMP ホスト**」は「**拡張**」リストの下には表示されません。これは、SNMP ホストがサブノード・プロパティーにコピーされ、既に表示されているためです。

– オーバーライドするプロパティーを 含まないセクションの場合、「**拡張**」リストに選択項目は表示されません。
– あるセクションで入力してオーバーライドした値は、別のセクションを選択して別のプロパティーを表示させても保持されます。
–「**任意のサブノードでの任意の構成プロパティーのオーバーライドを 許可します**」を選択すると、エージェントでこの機能が有効になります。

コマンド行からのサブノードの構成

IBM Tivoli Monitoring 環境では、コマンド行を使用してサブノードを 構成することもできます。

始める前に

サブノード 構成について詳しくは、1350 ページの『サブノード 構成』を参照してください。

このタスクについて

手順

- コマンド行からサブノード・インスタンスを構成するには、以下のコマンドを使用します。

```
tacmd configureSystem -m HOSTNAME:00 -p
section_name:subnode_instance_id.property_name=value
```

各項目の意味は次のとおりです。

section_name
サブノード・タイプと同様

subnode_instance_id
構成中に定義されたサブノードの ID

property_name
構成プロパティーの名前

value
プロパティーの値

サブノード構成の例

ここでは、定義済みの 1 つのサブノードを 使用してサンプル・エージェントを構成する方法について説明します。

例:

この例では、タイプが exs の「サブノード 例」という名前の 1 つのサブノードと、以下の 3 つの構成プロパティーを持つサンプル・エージェントの構成方法を示します。

- エージェント構成 (実際のプロパティー名は K00_AGENT_CFG) は、エージェント・レベルでのみ定義されます。
- サブノード構成 (実際のプロパティー名は K00_SUBNODE_CFG) は、サブノード 例の中でのみ定義されます。
- オーバーライド可能構成 (実際のプロパティー名は K00_OVERRIDABLE_CFG) は、エージェント・レベルで定義され、サブノード 例にコピーされました。

1355 ページの図 56 は、Agent Editor の「**ランタイム構成情報**」ページにあるこれらの構成プロパティーを示しています。

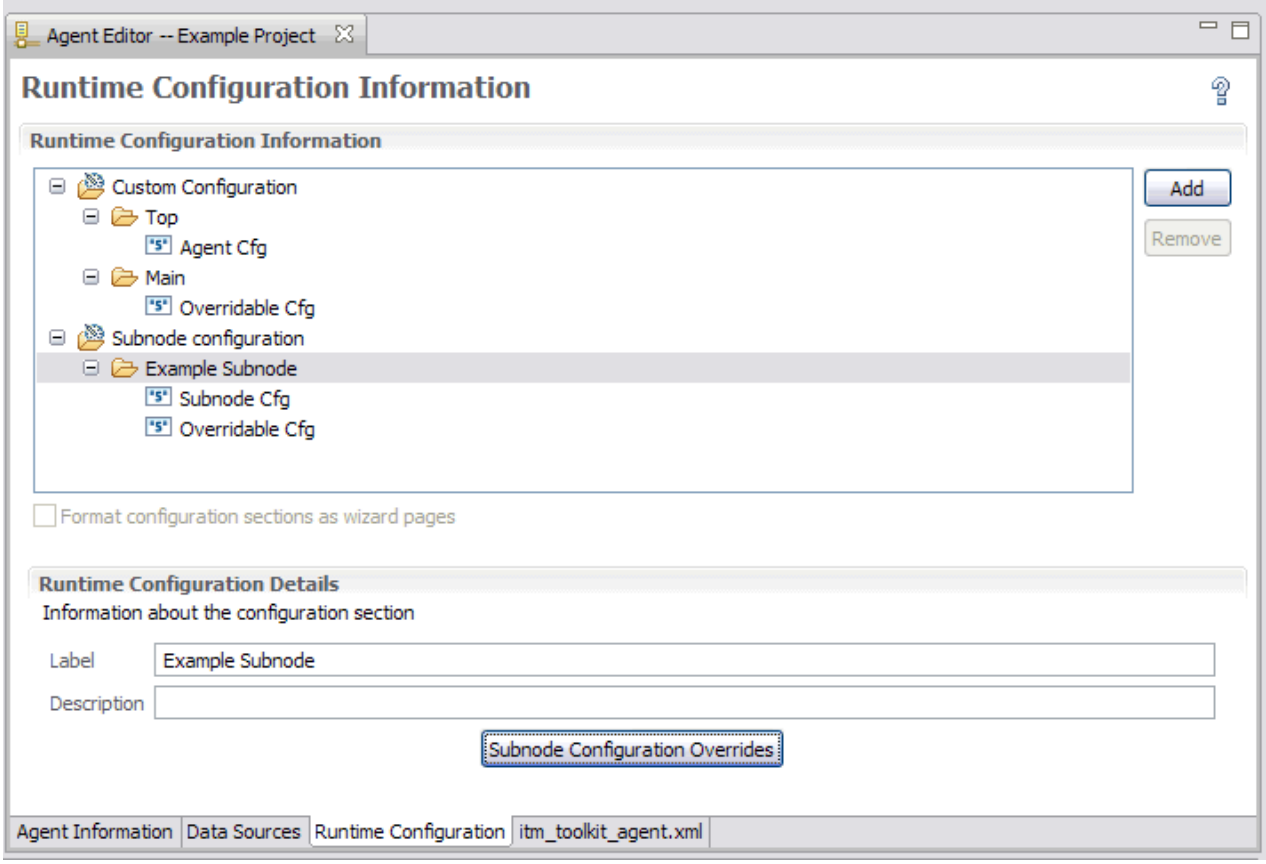


図 *56. Agent Builder* 内の構成プロパティー定義

このエージェント例を構成するときに最初に表示されるページは、「**上部**」セクションです。ここには、1356 ページの図 57 に示すように、「**エージェント構成**」プロパティーが表示されます。このプロパティーはエージェント・レベルのプロパティーであるため、エージェント構成中に 1 回だけ表示されます。サブノード例のどのインスタンスでもこのプロパティー値を参照できますが、すべてのインスタンスは同じ値を参照します。

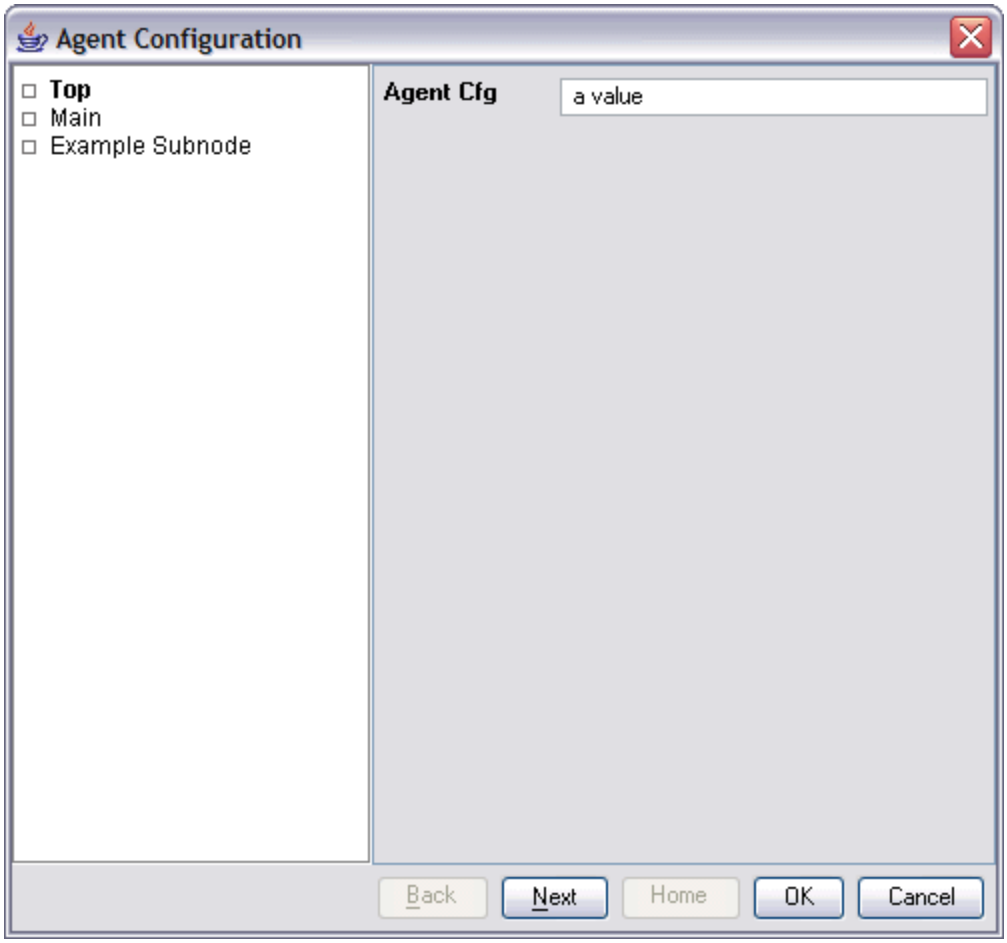


図 *57.* **「エージェント構成」**プロパティーのエージェント・レベル構成が指定された**「上部」**セクション

Tivoli Enterprise Monitoring Server コマンド行から構成する場合、「**エージェント構成**」プロパティーは、以下のコマンドを使用して設定できます。

```
tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00_AGENT_CFG=a value"
```

次に表示されるセクションは、1357 ページの図 58 に示す**メイン**・セクションです。これもエージェント・レベル・セクションであり、エージェント・レベルの**オーバーライド可能構成**プロパティーが含まれています。このプロパティーは、Agent Builder の「サブノード例」にコピーされたプロパティーであるため、「**エージェント構成**」プロパティーとは異なります。そのため、このプロパティーのデフォルト値を**メイン**ページ上に入力することができます。ただし、ここで入力された値は、それぞれの「サブノード例」インスタンスで別の値にオーバーライドすることができます。

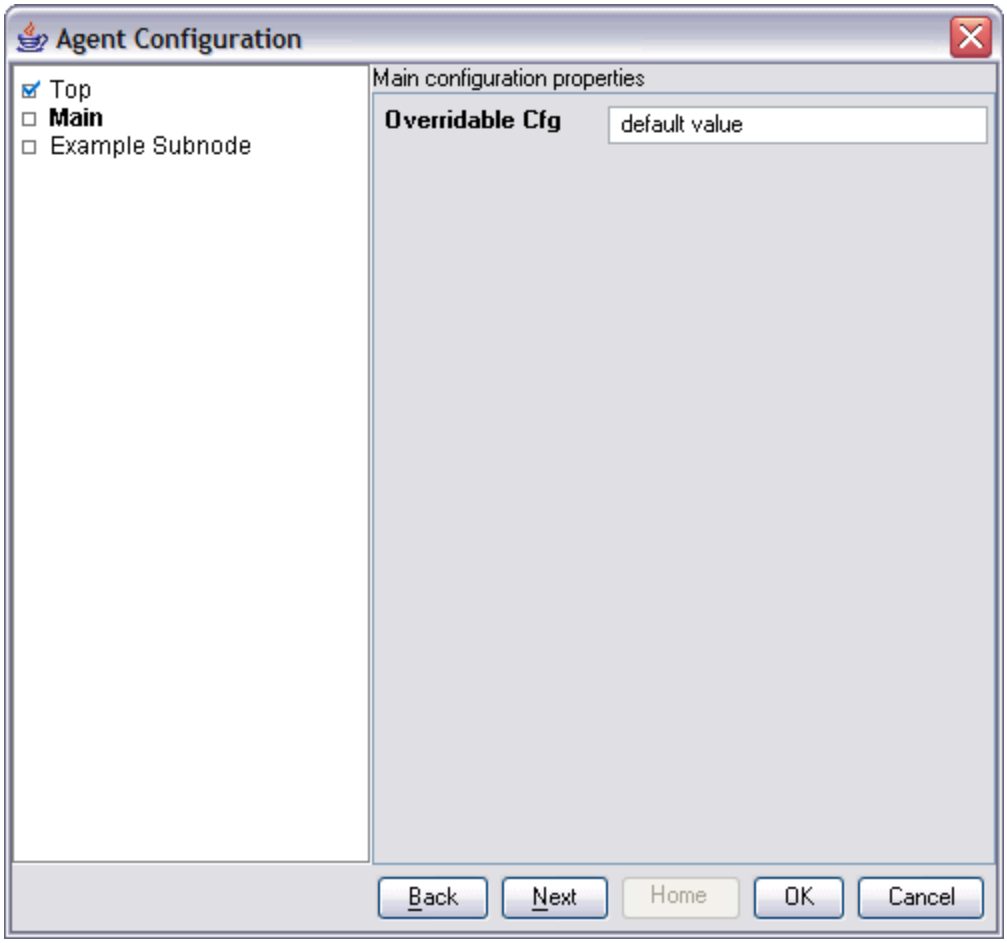


図 *58.* **オーバーライド可能構成**プロパティーのエージェント全体のデフォルト値が指定された**メイン**・セクション

Tivoli Enterprise Monitoring Server コマンド行から構成する場合、このプロパティーは、以下のコマンドを使用して設定できます。

```
tacmd configureSystem -m HOSTNAME:00 -p "MAIN.K00_OVERRIDABLE_CFG=default value"
```

これらのプロパティーは、両方とも同じエージェント・レベル・セクションに配置できます。カスタム・エージェント・レベル・セクションをいくつ作成するか、各セクションにカスタム・プロパティーをどのように配分するかを決定できます。

次に表示されるセクションは、1358 ページの図 59 に示す「**サブノード例**」セクションです。このエージェントは初めて構成されるため、定義されているサブノード・インスタンスは存在せず、サブノード・インスタンス・サブセクションも表示されません。初期プロパティー値サブセクションが表示されていますが、これはオプションであり、一部のサブノード・タイプではこれが表示されないことがあります。初期プロパティー値サブセクションが表示されているため、どの構成プロパティーに対してもデフォルト値を入力できます。「**オーバーライド可能構成**」プロパティーは、同じ名前のエージェント・レベル・プロパティーから取得されたデフォルト値を既に持っています。

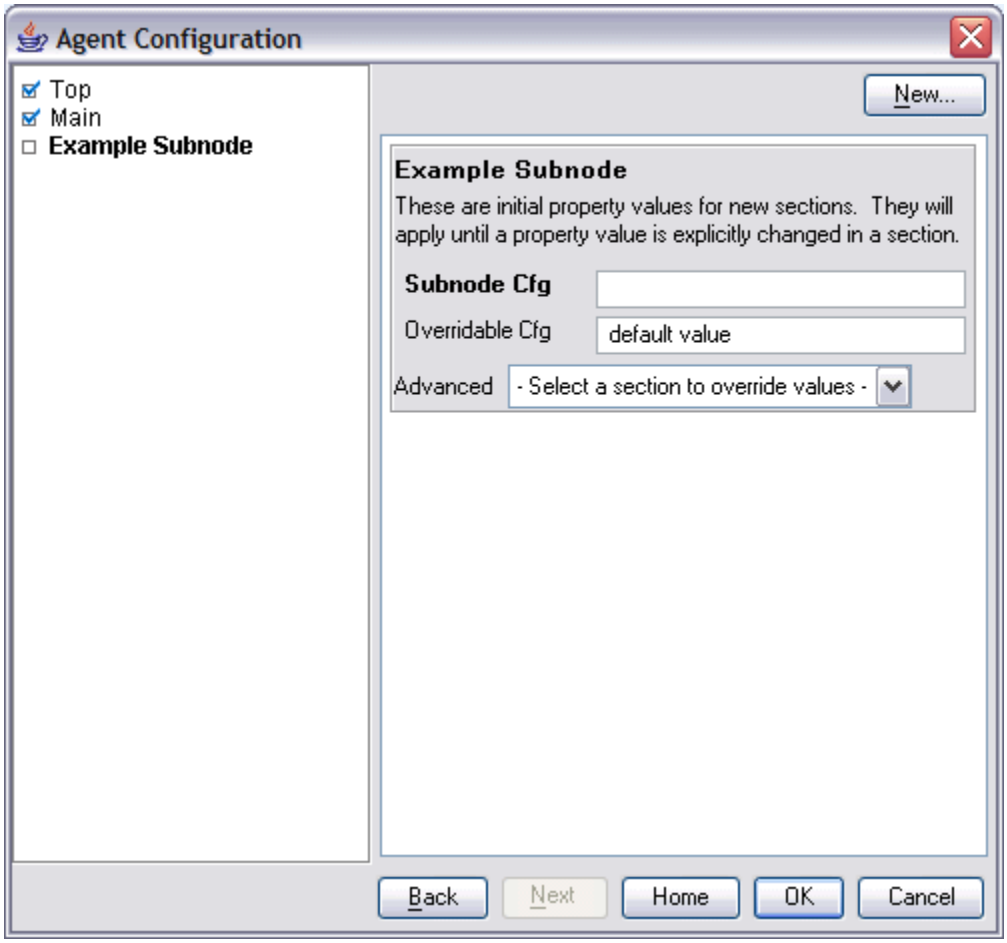


図 59. サブノードが指定されていない「**サブノード例**」セクション・ページ

サブノード・インスタンスは、空の「**サブノード例**」セクション・ページで以下のアクションを実行することによって定義されます (1359 ページの図 60)。

1. 初期「**サブノード例**」セクションの「**サブノード構成**」フィールドに、プロパティーのデフォルト・ストリング `sub-default value` を入力します。
2. 「**新規**」をクリックします。初期プロパティー・サブセクションの後に「**サブノード例**」サブセクションが表示されます。
3. 「**サブノード例**」フィールドに、サブノード・インスタンス ID `do` を入力します。
4. 「**新規**」をクリックします。最初のサブセクションの後に、2 番目の「**サブノード例**」サブセクションが表示されます。
5. 2 番目の「**サブノード例**」フィールドで、サブノード・インスタンス ID `re` を入力します。
6. 「**サブノード構成**」フィールドに、「**サブノード構成**」プロパティーの値 `sc override` を入力します。
7. 「**オーバーライド可能構成**」フィールドに、「**オーバーライド可能構成**」プロパティーの値 `oc override` を入力します。

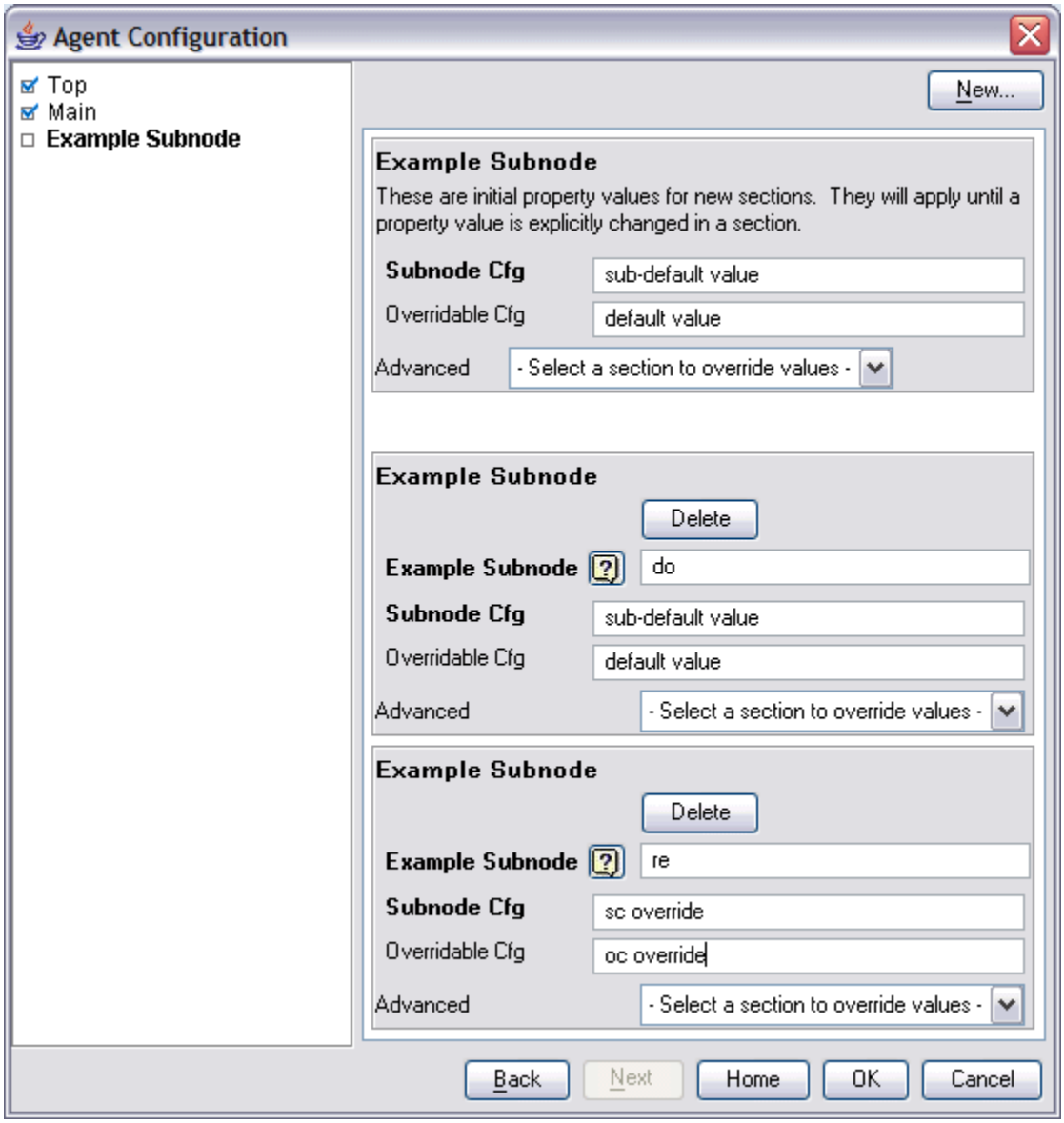


図 *60. 2* つのサブノード・インスタンスが定義された**「サブノード例」**セクション・ページ

エージェントが開始されると、この 2 つの新規サブセクションに基づいて、2 つのサブノード・インスタンスが作成されます。**do** サブノード・サブセクションのプロパティーは変更されなかったため、このサブノード・インスタンスでは、デフォルト・プロパティー値が使用されます。**re** サブセクションのプロパティーには異なる値が入力されたため、**re** サブノード・インスタンスでは、入力されたこれらの値が使用されます。

Tivoli Enterprise Monitoring Server コマンド行からデフォルト値を設定する場合、以下のコマンドを使用できます。

```
tacmd configureSystem -m HOSTNAME:00 -p "exs.K00_SUBNODE_CFG=sub-default value"
```

サブノード・デフォルト値を設定するフォーマットは、セクション名がサブノード・セクションを識別する点を除いては、エージェント・レベル・プロパティーを設定するフォーマットとまったく同じです。

Tivoli Enterprise Monitoring Server コマンド行からサブノード・インスタンスを作成する場合、以下のコマンドを使用できます。

```
tacmd configureSystem -m HOSTNAME:00 -p "exs:do.K00_OVERRIDABLE_CFG=default value" ¥
 "exs:re.K00_SUBNODE_CFG=sc override" "exs:re.K00_OVERRIDABLE_CFG=oc override"
```

サブノード・インスタンス ID は、セクション名とプロパティー名の間に挿入されます。コマンド行を使用してサブノード・インスタンスを作成するときには、すべてのプロパティーがデフォルト値を使用する場合でも、少なくとも 1 つのプロパティーを指定する必要があります。それ以外の場合、サブノード・インスタンスを定義するときに、コマンド行でデフォルト値を指定する必要はありません。

すべてのエージェント構成プロパティーを単一のコマンドで設定できます。 以下のコマンドは、上記の個別コマンドをすべて実行する操作のと同等です。

```
tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00_AGENT_CFG=a value" ¥
 "MAIN.K00_OVERRIDABLE_CFG=default value" ¥
 "exs.K00_SUBNODE_CFG=sub-default value" ¥
 "exs:do.K00_OVERRIDABLE_CFG=default value" ¥
 "exs:re.K00_SUBNODE_CFG=sc override" "exs:re.K00_OVERRIDABLE_CFG=oc override"
```

サブノードおよび Windows データ・ソース
Windows リモート接続プロパティーをエージェントに含めるかどうかを選択します。

このタスクについて

エージェントが Windows データ・ソースをエージェント・レベルで持っており、サブノードには持っていない場合、必要に応じて Windows リモート接続構成プロパティーをエージェントに含めることができます。Windows データ・ソースとは、Windows イベント・ログ、Windows Management Instrumentation、および Windows パフォーマンス・モニターのことです。構成プロパティーを含めない場合、これらのデータ・ソースはデフォルトでローカルの Windows システムをモニターするため、構成は不要です。デフォルトでは、Windows データ・ソースはどのサブノードにも 含まれません。

Windows リモート接続プロパティーをエージェントに含めるかどうかを選択するには、以下のステップを実行します。

手順

1. 「**Windows Management Instrumentation (WMI) 情報**」ページで、データ・ソース・プロパティーを表示しているときに「**グローバル・オプション**」をクリックします。データ・ソースの作成中、または Agent Editor の「**データ・ソース**」ページから、「**グローバル・オプション**」を選択します。
2. これらのプロパティーをエージェントに含める場合は、「**Windows データ・ソースのグローバル・オプション**」ウィンドウで、「**Windows リモート接続の構成を組み込む**」を選択します。

サブノードおよびスクリプト・データ・ソース
サブノード・インスタンス構成プロパティーは、エージェント・レベルのスクリプトの場合と同様に、サブノード・スクリプトでアクセスされます。

スクリプトは、エージェント・レベルのすべての構成プロパティー、およびサブノード・インスタンスのすべての構成プロパティーにアクセスできます。エージェント・レベルのプロパティーがサブノード・レベルでオーバーライドされている場合、スクリプトはそのサブノード・レベルのプロパティー値にしかアクセスできません。

エージェント構成のカスタマイズ

プロセス、ログ・ファイル、およびスクリプト・データ・ソースの構成をカスタマイズします。

始める前に

SNMP、JMX、CIM、JDBC、HTTP、および SOAP のデータ・ソースをエージェントに追加する場合は、以下のセクションで説明されているように、それらのデータ・ソースを構成します。

- 1229 ページの『Simple Network Management Protocol (SNMP) サーバーのデータのモニター』
- 1239 ページの『Java Management Extensions (JMX) MBean のモニター』
- 1258 ページの『Common Information Model (CIM) のデータのモニター』
- 1285 ページの『Java Database Connectivity (JDBC) のデータのモニター』

- 1295 ページの『HTTP 可用性および応答時間のモニター』
- 1303 ページの『SOAP データ・ソースまたは他の HTTP データ・ソースのデータのモニター』

このタスクについて
このタスクを使用して、プロセス、ログ・ファイル、およびスクリプト・データ・ソースの構成をカスタマイズして、エージェントがモニターしているアプリケーションにアクセスできるようにします。

エージェントを開始するには、それらをすべて構成しておく必要があります。すべてのエージェントは、Tivoli Enterprise Monitoring Server との接続方法などの基本構成情報を把握している必要があります。多くの場合、エージェントは自身が稼働しているシステムに固有の情報にアクセスするために、さらに多くの構成情報を必要とします。例えば、ソフトウェア製品のインストール・ロケーションを知る必要がある場合は、プロンプトを出してそれらの情報を要求するための構成プロパティーを追加します。プロンプトが出される情報には、ほかにもインターフェースにアクセスするためのユーザー ID やパスワードなどがあります。

カスタム構成は、エージェントの開発者によって定義されます。カスタム構成はすべてのエージェントに必要であるわけではありませんが、データ収集の以下の領域で使用できます。

- プロセス・モニターでの引数のマッチング
- プロセス・モニターでのコマンド行のマッチング
- ログ・ファイルのパスまたは名前の形成
- スクリプト内での環境変数の定義

注 : JMX や SNMP などの特定のデータ・ソースは、その構成を自動的に追加します。

注 : データ・ソース固有の構成が Agent Builder によって自動的に追加される場合は、英語のみで追加されます。

データ・ソース定義中にエージェントがデータ収集の領域に関するシステム固有の情報を必要とする場合は、「**プロパティーの挿入**」または「**構成プロパティーの挿入**」が表示されます。

例えば、ログ・ファイルをモニターする属性グループを作成している場合は、「**構成プロパティーの挿入**」が表示されます。

手順

1. 「**構成プロパティーの挿入**」をクリックして、「**構成プロパティー**」ウィンドウを表示します。
2. 「**構成プロパティー**」ウィンドウでプロパティーをクリックし、「**追加**」をクリックします。

   注 : 最初はエージェントには構成プロパティーが定義されていません。
3. 「**ランタイム構成プロパティー**」ウィンドウで、以下のフィールドに入力します。

   a) 「**セクション**」域で、以下のフィールドに入力します。

   ラベル
   プロパティーを説明するテキスト

   説明
   (オプション) プロパティーの説明

   b) 「**プロパティー**」域で、以下のフィールドに入力します。

   ラベル
   入力する必要のある情報を指定する、エージェント構成パネルに表示されるテキスト。

   環境変数
   環境変数は「**環境変数**」フィールドに表示され、ラベル・フィールドに入力したとおりに更新されます。Agent Builder は、製品コードおよびラベルから自動的に環境変数の名前を構成します。 ラベルとは無関係に環境変数を変更する場合は、「**ラベルに合わせる**」をクリアします。

   説明
   (オプション) 定義されているプロパティーの説明。

   タイプ
   収集される情報のタイプ (以下のいずれかのオプション)。

ストリング
収集する必要のある英字情報 (例えば、インストール・ロケーション、ユーザー名、ホスト名など) の場合。

パスワード
保管時に暗号化する必要のある情報の場合。データが暗号化されるだけでなく、テキスト・ボックスに入力したデータはアスタリスクで覆い隠されます。また、データを確認するため、この情報を 2 回入力する必要があります。

数値
数値情報の場合 (例えば、ポート番号)。

選択項目
特定の値のリストの場合。このオプションを使用すると、「選択項目」テーブルが有効になります。「**追加**」をクリックすると、特定の値を定義できます。入力した値は、エージェント構成パネルに選択項目のグループとして表示されます。グループから選択できるのは 1 つだけです。

読み取り専用テキスト
エージェントの構成時にテキストが表示されますが、 情報は収集されません。

分離文字
横方向の分離文字が表示されますが、 情報は収集されません。

ファイル・ブラウザー
ストリングを収集します。 このストリングはファイル名です。「**参照**」をクリックすると、対象とするファイルをファイル・システムで参照できます。

デフォルト値
(オプション) エージェントが初めて構成される実行時に構成パネルに表示される値を指定します。 Windows 用のデフォルト値と異なる UNIX/Linux 用のデフォルト値を指定する場合は、「**複数の値**」をクリックします。

「**構成プロパティーのデフォルト値**」ウィンドウで、Windows システム用、UNIX システム用、および Linux システム用のデフォルト値を指定します。

注 : 複数のデフォルト値のサポートは、IBM Tivoli Monitoring V6.2.1 以降でのみサポートされる機能です。ご使用のエージェントが IBM Tivoli Monitoring V6.2 と互換性がある場合、プロンプトによりこの要件についての警告が表示されます。キャンセルするか、V6.2.1 との互換性を有効にして続行することができます。

必須
エージェントの構成時にユーザーが値を入力する必要がある場合には、このフィールドにチェック・マークを付けます。 ユーザーによる値の入力がオプションの場合は、このフィールドをクリアします。

c) 選択項目を追加するには、「**追加**」をクリックします。

4. 「**構成プロパティー値**」ウィンドウで、「**ラベル**」フィールドと「**値**」フィールドに入力します。
「ラベル」は、選択項目の 1 つとして表示されます。この項目を選択すると、値がプロパティー値になります。
5. 「**OK**」をクリックします。
新規の構成セクションとプロパティーが「**構成プロパティー**」ウィンドウの「**カスタム構成**」の下に表示されます。
6. オプション: 別のプロパティーを既存のセクションに追加するには、そのセクションを選択するか、そのセクション内の既存のプロパティーを選択し、「**追加**」をクリックします。「**構成プロパティー**」ウィンドウのランタイム構成ツリーで選択を行います。
7. 新しいプロパティーのためのフィールドに 入力します。(ステップ 1361 ページの『3』 と同じフィールドに入力します。)
8. 「**OK**」をクリックします。 一番新しく追加されたプロパティーが選択されています。
9. 選択をそのままにしておくか、ログ・ファイル名に挿入するプロパティーを選択します。
10. 「**OK**」をクリックします。 プロパティーがログ・ファイル名に挿入されます。
ウィザードを進めていくと、ログ・ファイル属性グループの定義を完了できます。

注 : 構成プロパティーはログ・ファイル名のコンテキストで定義されていますが、この構成プロパティーは他のロケーションで使用することもできます。例えば、構成プロパティーを受け入れるロケーションにはスクリプト・データ・ソースもあります。このように柔軟性があるため、製品コードが K00 の場合は、スクリプト変数 *$K00_APPLICATION_LOG_FILE* を使用して、構成エレメント「**ファイル情報**」の値にアクセスできます。また、Windows バッチ・ファイル変数 *%K00_APPLICATION_LOG_FILE%* を使用することもできます。

Agent Editor を使用した構成プロパティーの変更

Agent Editor を使用して、エージェントの構成プロパティーを変更します。

このタスクについて

このタスクでは、Agent Editor を使用した構成プロパティーの表示、追加、および変更について説明します。

手順

1. 「**ランタイム構成**」タブをクリックします。
2. 構成セクションを選択して、「**追加**」をクリックします。

   「**追加**」は、1360 ページの『エージェント構成のカスタマイズ』で動作するのと同じように動作します。構成セクションまたはプロパティーは、それを選択したときに編集するので、「**編集**」の選択肢はありません。
3. 構成プロパティーを選択して、「**ランタイム構成の詳細**」域を表示します。
4. 「**ランタイム構成の詳細**」域で、フィールドを編集してプロパティーを構成します。

Windows リモート接続の構成

Windows リモート接続の構成に関する情報

このタスクについて

Windows Management Instrumentation (WMI)、Windows パフォーマンス・モニター (Perfmon)、および Windows Event Log の各データ・ソースは、エージェントがインストールされているシステム 上のデータをモニターできます。これらのデータ・ソースは、リモート Windows システム上のデータをモニターすることもできます。この 3 種類のデータ・ソース・タイプを Windows データ・ソースと呼びます。これら Windows データ・ソースが リモート側でデータをモニターしている場合は、定義済みのエージェント・レベルに対して、 すべて、Windows リモート接続構成プロパティーを共有します。

エージェントの基本レベルで Windows データ・ソースを定義する場合、Windows リモート接続構成プロパティーが自動的にエージェントに追加されることはありません。それらが追加されない理由は、リモート・モニターを使用可能にする前に Windows データ・プロバイダーを使用していた可能性がある旧バージョンのエージェントとの互換性を維持するためです。エージェント内の Windows データ・ソースは、そのエージェントがインストールされているローカル Windows システム 上のデータをモニターします。

エージェント内のサブノードで Windows データ・ソースを定義する場合は、Windows リモート接続構成プロパティーが自動的にエージェントに追加されます。Windows リモート接続がサブノード内にある場合は、Windows データ・ソースはこれをサポートする必要があります。 このオプションは、エージェント内のすべてのサブノードからすべての Windows データ・ソースを削除するまでは、クリアできません。サブノードの各インスタンスが、それぞれ別のリモート Windows システムをモニターするように 構成されている場合があります。サブノード内の Windows データ・ソースはすべて、Windows リモート接続構成プロパティーを共用します。

基本エージェントがリモート側から単一のリモート Windows システムをモニターするように 構成する場合は、以下の手順を使用します。

手順

1. Agent Editor の「**データ・ソース定義**」ウィンドウで、「**グローバル・オプション**」をクリックします。

   「**Windows データ・ソースのグローバル・オプション**」ウィンドウが開きます。

2.「**Windows リモート接続の構成を組み込む**」を選択します。
3.「**OK**」をクリックします。

タスクの結果

以下の接続固有の構成プロパティーには、Agent Editor の「**ランタイム構成情報**」ページから「**Windows リモート・アクセスの場合の構成**」 > 「**Windows リモート接続**」を選択することでアクセスできます。

リモート Windows ホスト
リモート Windows コンピューターのホスト名

リモート Windows パスワード
リモート Windows のパスワード

リモート Windows DOMAIN¥ユーザー名
リモート Windows ホストのユーザー名

次のタスク

Agent Editor を使用して、構成プロパティーの表示、追加、および変更を行うことができます。説明については、1363 ページの『Agent Editor を使用した構成プロパティーの変更』を参照してください。Windows データ・ソースがサブノードで 定義されている場合は、「サブノード 構成のオーバーライド」を指定することもできます。説明については、1350 ページの『サブノード 構成』を参照してください。

Windows Management Instrumentation (WMI) 許可を持つユーザーの作成

Windows システムで WMI を参照する許可を持つユーザーを追加および構成できます。

このタスクについて

Windows Management Instrumentation (WMI) を使用してリモート・システムからデータを収集するエージェントには、リモート・システム上の WMI データにアクセスする許可が必要です。 エージェントがリモート・システム上の WMI データにアクセスできるのは、そのシステム上の WMI データにアクセスする許可を持つアカウントの資格情報を指定した場合です。この手順は、Windows 7、Windows 2008 Server、および Windows Vista に適用されます。

注 : エージェントが Windows リモート・システムのデータにアクセスするには、Windows パフォーマンス・モニター (Perfmon) および Windows イベント・ログのデータ・ソースを使用することもできます。ただし、Windows パフォーマンス・モニター (Perfmon) および Windows イベント・ログのデータ・ソースの場合は、リモート・システムの管理者資格情報を指定する必要があります。

手順

1. ユーザー・アカウントの作成:
   a. Windows で「**スタート**」 > 「**管理ツール**」 > 「**コンピューターの管理**」を選択します。「**コンピューターの管理**」ウィンドウが開きます。
   b.「**ローカル ユーザーとグループ**」を展開します。
   c.「**ユーザー**」フォルダーを右クリックして「**新しいユーザー**」を選択します。
   d. ユーザーの詳細を入力して、「**作成**」および「**閉じる**」をクリックします。
2. 新規ユーザー・アカウントのグループ・メンバーシップを構成するには、以下の手順を実行します。
   a.「**コンピューターの管理**」ウィンドウで、「**ユーザー**」フォルダーを選択します。
   b. 新しいユーザー・アカウントを右クリックして、「**プロパティ**」を選択します。
   c.「**所属するグループ**」タブをクリックします。
   d.「**追加**」をクリックします。
   e.「**拡張**」をクリックします。
   f.「**検索**」をクリックします。
   g. 以下のグループを選択します。

- Distributed COM Users
- Performance Log Users
- Remote Desktop Users

ヒント: 複数のグループを選択するには、Ctrl キーを押しながらクリックします。

h.「**コンピューターの管理**」ウィンドウに戻るまで「**OK**」をクリックします。

i.「**ファイル**」 > 「**終了**」を選択して、「**コンピューターの管理**」ウィンドウを終了します。

3. 分散コンポーネント・オブジェクト・モデル (DCOM) 権限の割り当て:

a. Windows で「**スタート**」 > 「**管理ツール**」 > 「**コンポーネント サービス**」を選択します。「**コンポーネント サービス**」ウィンドウが開きます。

b.「**コンポーネント サービス**」 > 「**コンピューター**」 > 「**マイ コンピューター**」を展開します。

c.「**マイ コンピューター**」を右クリックして「**プロパティ**」を選択します。「**マイ コンピューターのプロパティ**」ウィンドウが開きます。

d.「**COM セキュリティ**」タブをクリックします。

e.「**アクセス許可**」領域で「**制限の編集**」をクリックします。

f.「**Distributed COM Users**」で、「**ローカル アクセス**」および「**リモート アクセス**」が選択されていることを確認します。

g.「**OK**」をクリックして設定を保存します。

h.「**マイ コンピューターのプロパティ**」ウィンドウの「**起動とアクティブ化のアクセス許可**」領域で、「**制限の編集**」をクリックします。

i.「**Distributed COM Users**」で、「**ローカルからの起動**」、「**リモートからの起動**」、「**ローカルからのアクティブ化**」、および「**リモートからのアクティブ化**」が選択されていることを確認します。

j.「**OK**」をクリックして設定を保存し、もう一度「**OK**」をクリックして「**マイ コンピューターのプロパティ**」ウィンドウを閉じます。

k.「**ファイル**」 > 「**終了**」を選択して、「**コンポーネント サービス**」ウィンドウを閉じます。

4. WMI ネーム・スペース・セキュリティーを構成します。

a. Windows で「**スタート**」 > 「**ファイル名を指定して実行...**」を選択します。

b. `wmimgmt.msc` と入力して、「**OK**」をクリックします。

c.「**WMI コントロール (ローカル)**」を右クリックして「**プロパティ**」を選択します。

d.「**セキュリティ**」タブをクリックします。

e.「**セキュリティ**」をクリックします。

f.「**追加**」をクリックします。

g.「**拡張**」をクリックします。

h.「**検索**」をクリックします。

i. 新しいユーザー・アカウントを選択して、「**セキュリティ Root**」ウィンドウに戻るまで「**OK**」をクリックします。

j.「**詳細設定**」をクリックして、新規追加したユーザー・アカウントを選択します。

k.「**編集**」をクリックします。

l.「**適用先:**」選択メニューから、「**この名前空間と副名前空間**」を選択します。

m.「**メソッドの実行**」で、「**アカウントの有効化**」、「**リモートの有効化**」、および「**セキュリティの読み取り**」が選択されていることを確認します。

n.「**wmimgmt**」ウィンドウに戻るまで「**OK**」をクリックします。

o.「**ファイル**」 > 「**終了**」を選択して、「**wmimgmt**」ウィンドウを終了します。

次のタスク

リモート・システムからの WMI データの収集について詳しくは、1224 ページの『Windows Management Instrumentation (WMI) からのデータのモニター』を参照してください。

Secure Shell (SSH) リモート接続の構成

SSH リモート接続の構成に関する情報

このタスクについて

スクリプト・データ・ソースは、エージェントがインストールされているシステムのほか、 リモート・システム上のデータもモニターできます。スクリプト・データ・ソースが リモート側でデータをモニターしている場合は、定義済みのエージェント・レベルに対して、 すべて SSH リモート接続構成プロパティーを共有します。 旧バージョンのエージェントは、リモート・モニターを使用可能にする前にスクリプト・データ・プロバイダーを使用する場合がありました。旧バージョンのエージェントとの互換性を維持するために、SSH リモート接続構成プロパティーは、自動的にはエージェントに追加されません。エージェント内のスクリプト・データ・ソースは、 そのエージェントがインストールされているローカル・システム上のデータをモニターします。

サブノードでスクリプト・データ・ソースを定義し、「**SSH を使用したデータ収集を使用可能にする**」を選択した場合は、サブノード・インスタンスごとに別のリモート・システムをモニターするように 構成できます。サブノード内のスクリプト・データ・ソースはすべて、 SSH リモート接続構成プロパティーを共有します。

エージェントにリモート側からリモート・システムをモニターさせる場合は、以下の手順を使用します。

手順

Agent Editor のスクリプト・データ・ソースの「**データ・ソース定義**」ウィンドウで、「**SSH を使用したデータ収集を使用可能にする**」を選択します。

タスクの結果

以下の接続固有の構成プロパティーには、**Agent Editor** の「**ランタイム構成情報**」ページから「**セキュア・シェル (SSH) の場合の構成**」 > 「**SSH リモート接続**」を選択することでアクセスできます。

ネットワーク・アドレス
リモート・コンピューターの IP アドレスまたはホスト名。

SSH ポート番号
SSH サーバーが稼働している IP ポート番号。デフォルト値は 22 です。

認証タイプ
リモート SSH サーバーにログオンするときに使用する認証の種類。「パスワード」または「公開鍵」を選択できます。

収集間隔が終わるごとにリモート・システムから 切断
スクリプト・データ・プロバイダーがデータの収集後にリモート・システムに対するログイン・セッションを打ち切るかどうかを決定するオプション。デフォルトでは、この値は「いいえ」です。

収集間隔が終わるごとにリモート・システムから スクリプトを削除
データ収集間隔が終わるごとにリモート・システムから スクリプトを削除するためのオプション。デフォルトでは、この値は「いいえ」です。

認証タイプを「パスワード」に設定する場合は、**Agent Editor** の「**ランタイム構成情報**」ページから「**セキュア・シェル (SSH) の場合の構成**」 > 「**パスワード**」を選択することで以下の構成プロパティーにアクセスできます。

ユーザー名
リモート・システムのユーザー名

パスワード
リモート・システムのパスワード

認証タイプを「公開鍵」に設定する場合は、**Agent Editor** の「**ランタイム構成情報**」ページから「**セキュア・シェル (SSH) の場合の構成**」 > 「**公開鍵**」を選択することで以下の構成プロパティーにアクセスできます。

ユーザー名
公開鍵ファイルに関連付けるユーザー名

公開鍵ファイル
ユーザーに関連付ける公開鍵ファイル

秘密鍵ファイル
ユーザーに関連付ける秘密鍵ファイル

パスワード
秘密鍵ファイルのロック解除に使用するパスワード

次のタスク

Agent Editor を使用して、構成プロパティーの表示、追加、および変更を行うことができます。説明については、1363 ページの『Agent Editor を使用した構成プロパティーの変更』を参照してください。SSH リモート接続構成プロパティーがサブノードに組み込まれている場合は、「サブノード構成のオーバーライド」を指定することもできます。 説明については、1350 ページの『サブノード構成』を参照してください。

ワークスペース、アクション実行コマンド、およびシチュエーションの作成

エージェントを IBM Tivoli Monitoring 環境にインストールした後は、モニター・ソリューション用のワークスペース、照会、アクション実行コマンド、およびシチュエーションを作成できます。

インストール・パッケージには、作成するシチュエーション、ワークスペース、アクション実行コマンド、および照会を組み込むことができます。 シチュエーション、ワークスペース、およびエージェント自体を単一のインストール・イメージに含めるには、シチュエーションとワークスペースのファイルが、エージェントと同じプロジェクト内になければなりません。Agent Builder には、エージェント・プロジェクト内に該当のファイルを作成するためのウィザードが用意されています。アプリケーション・サポート・ファイルのインポートについては、1403 ページの『アプリケーション・サポート・ファイルのインポート』を参照してください。

シチュエーション、アクション実行コマンド、および照会の作成

シチュエーション、アクション実行コマンド、および照会を作成するために役立つ情報を検索します。

シチュエーション、アクション実行コマンド、および照会を作成するには、Tivoli Enterprise Portal と組み込みシチュエーション・エディターを使用します。シチュエーションの作成方法について詳しくは、「Tivoli Enterprise Portal ユーザーズ・ガイド」を参照してください。また、Tivoli Enterprise Portal Server とともにインストールされたヘルプ資料も利用できます。Agent Builder モニター・エージェントは、一連の特定のアクション実行コマンドの特殊なプロセスを認識して実行できます。 これらの特殊なアクション実行コマンドについて詳しくは、1504 ページの『アクション実行コマンド参照』を参照してください。

システム・モニター・エージェント用のシチュエーションは、Tivoli Enterprise Portal のシチュエーション・エディターまたは **tacmd createSit** コマンドを使用して作成される Enterprise シチュエーションとは違った方法で作成されます。システム・モニター・エージェントの場合、専用シチュエーションは、エージェントのローカル専用シチュエーション構成 XML ファイルに作成されます。システム・モニター・エージェント用のシチュエーションの作成について詳しくは、「*IBM Tivoli Monitoring* 管理者ガイド」の『エージェントのオートノミー』という章の『専用シチュエーション』を参照してください。

ワークスペースの 作成

エクスポートしてご使用のソリューションに組み込むことができるワークスペースを作成するには、Tivoli Enterprise Portal を管理者モードにする必要があります。

このタスクについて

ワークスペースは、ワークスペースが使用されることになる環境でビルドしてください。ワークスペースをビルドする際に、ご使用の環境で通常使用される最小解像度でワークスペースをビルドするために、コンピューターの表示設定を変更します。これより高い解像度でワークスペースをビルドすると、より低い解像度に設定した場合に、表示が混み合って見にくくなることがあります。

エクスポートしてご使用のソリューションに組み込むことができるワークスペースを作成するには、Tivoli Enterprise Portal を「管理者」モードにする必要があります。Tivoli Enterprise Portal を「管理者」モードにするには、以下のステップを実行します。

手順

1. *ITM_INSTALL*/CNP ディレクトリーに進み、cnp.bat ファイルを開きます。

   デフォルト・インストールを使用した場合、ディレクトリーは C:¥IBM¥ITM¥CNP です。cnp.bat ファイルで、set _CMD= %_JAVA_CMD% 行を更新して、オプション -Dcnp.candle.mode="$_KCJ_$" を組み込む必要があります。

   Linux または AIX システムで拡張を作成する場合は、以下のパスを使用します。

   ```
   /opt/IBM/ITM/li263/cj/bin/cnp.sh
   ```

   ここで、*li263* は Tivoli Enterprise Portal が実行されているオペレーティング・システムです。

   更新された set _CMD= %_JAVA_CMD% は、以下のように表示されます。

   ```
   set _CMD= %_JAVA_CMD% -Dcnp.candle.mode="$_KCJ_$" -Xms64m -Xmx256m -showversion -noverify
    -classpath %CPATH% -Dkjr.trace.mode=LOCAL -Dkjr.trace.file=C:¥IBM¥ITM¥CNP¥LOGS¥kcjras1.log
   -Dkjr.trace.params=ERROR -DORBtcpNoDelay=true -Dibm.stream.nio=true
   -Dice.net.maxPersistentConnections=16 -Dice.net.persistentConnectionTimeout=1
   -Dcnp.http.url.host=SKINANE -Dvbroker.agent.enableLocator=false  -Dnv_inst_flag=%NV_INST_FLAG
   %
   -Dnvwc.cwd=%NVWC_WORKING_DIR% -Dnvwc.java=%NVWC_JAVA% candle.fw.pres.CMWApplet
   ```

   注 : ここではコマンドが複数の行にわたっていますが、あくまで書式上の理由によるものです。
2. 新規の Tivoli Enterprise Portal クライアントを開き、sysadmin ユーザー ID を使用してログインします。
3. 「管理者」モードで、「sysadmin」ユーザー ID を設定します。 Tivoli Enterprise Portal で、「**編集**」 > 「**ユーザー管理**」を選択します。「**SYSADMIN**」を選択し、「**権限**」タブで「**ワークスペース管理**」を選択します。「**ワークスペース管理モード**」チェック・ボックスを選択します。

   正しく選択すると、デスクトップのタイトル・バーに ***ADMIN MODE*** と表示されます。

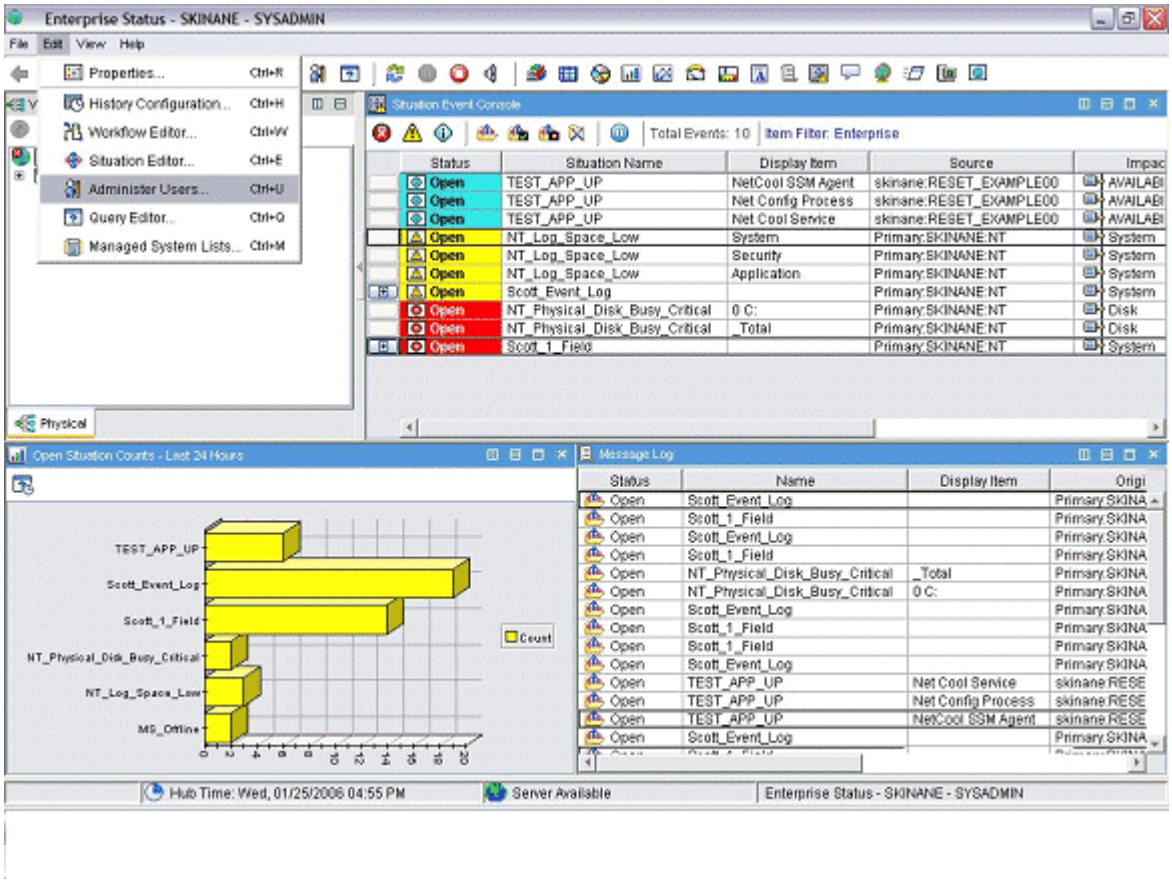


図 *61. sysadmin* ユーザー *ID* の設定

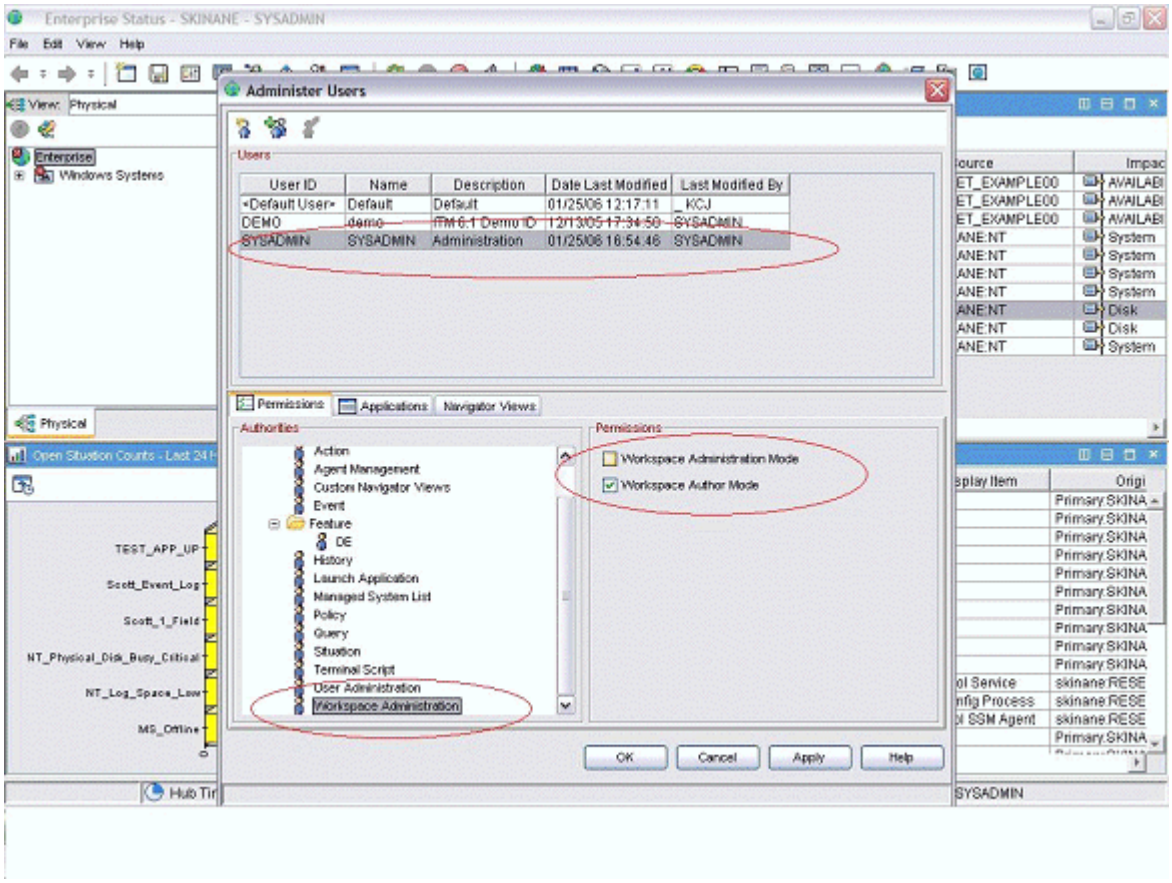


図 *62. sysadmin* ユーザー *ID* の設定 *(*続き*)*

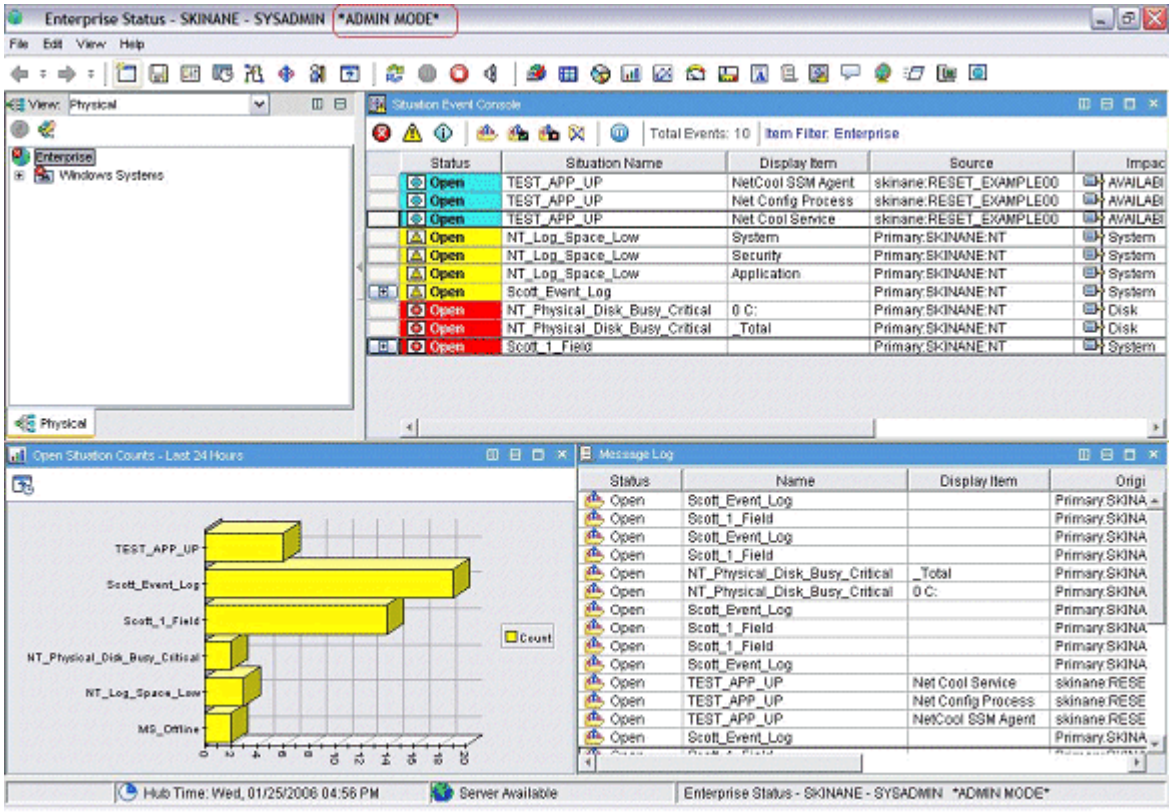


図 63. *sysadmin* ユーザー *ID* の設定 (続き)

次のタスク

1371 ページの図 63 に示したとおりに「管理者」モードになったら、アプリケーションのワークスペースを作成する準備ができています。ワークスペースのカスタマイズおよび作成方法について詳しくは、「Tivoli Enterprise Portal ユーザーズ・ガイド」を参照してください。 あるいは、ご使用の Tivoli Enterprise Portal コンポーネントと共にインストールされるヘルプ資料をご覧ください。

ワークスペースを「読み取り専用」にする場合、および顧客による削除を防止する場合には、「編集不可」および「削除不可」のプロパティーを各ワークスペースに設定します。ワークスペース・プロパティーで、以下のプロパティーを選択してください。

- **変更を許可しない**
- **IBM 提供の製品 (削除不可としてマーク)**

ワークスペースを参照するか、ワークスペースにあるコントロール付きアイコンをクリックして、これらのプロパティーに移動できます。あるいは、ビューのプロパティー・ページのいずれか 1 つに移動し、プロパティー・ツリーでワークスペース・レベルに移動することもできます。ナビゲーター項目ごとに複数のワークスペースがある場合は、必ずそれぞれのワークスペースごとに、これらのプロパティーを設定してください。 次の画面キャプチャーで例を示します。

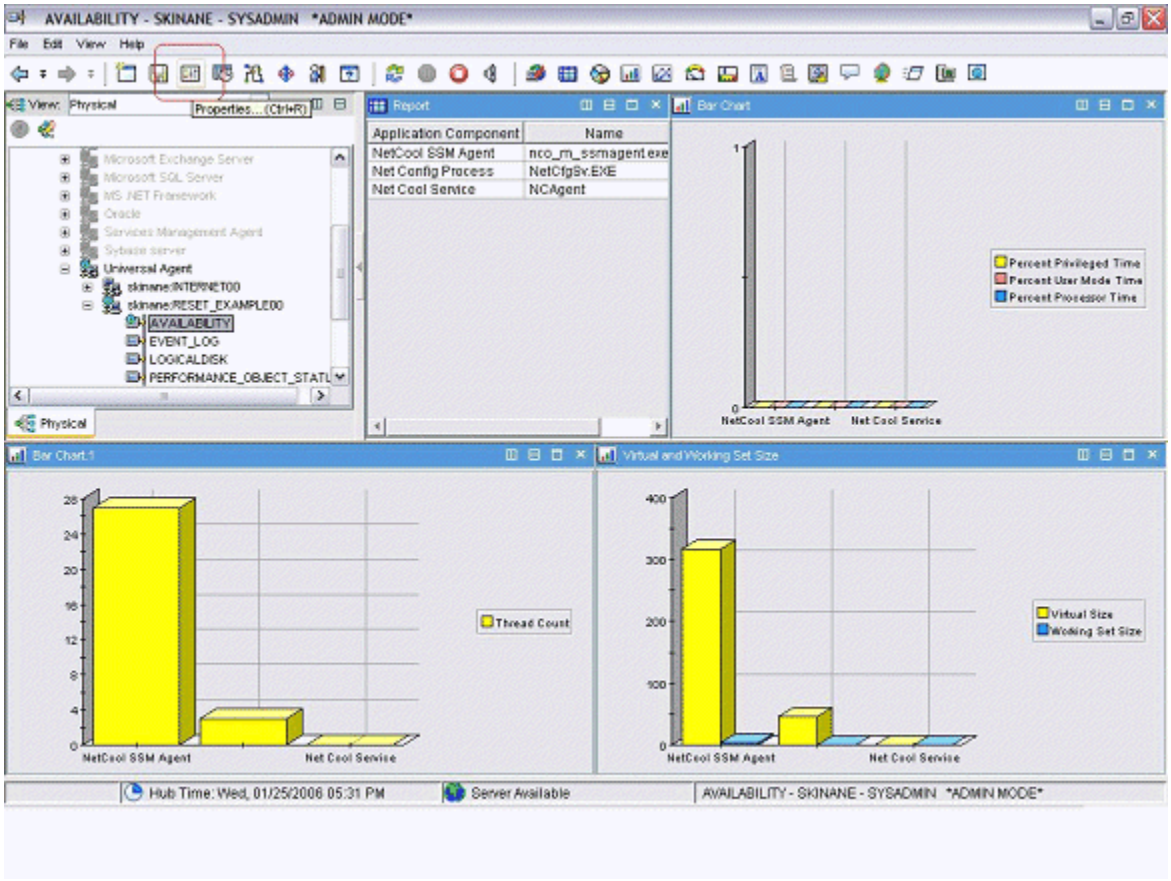


図 *64.* ワークスペース・プロパティーの設定

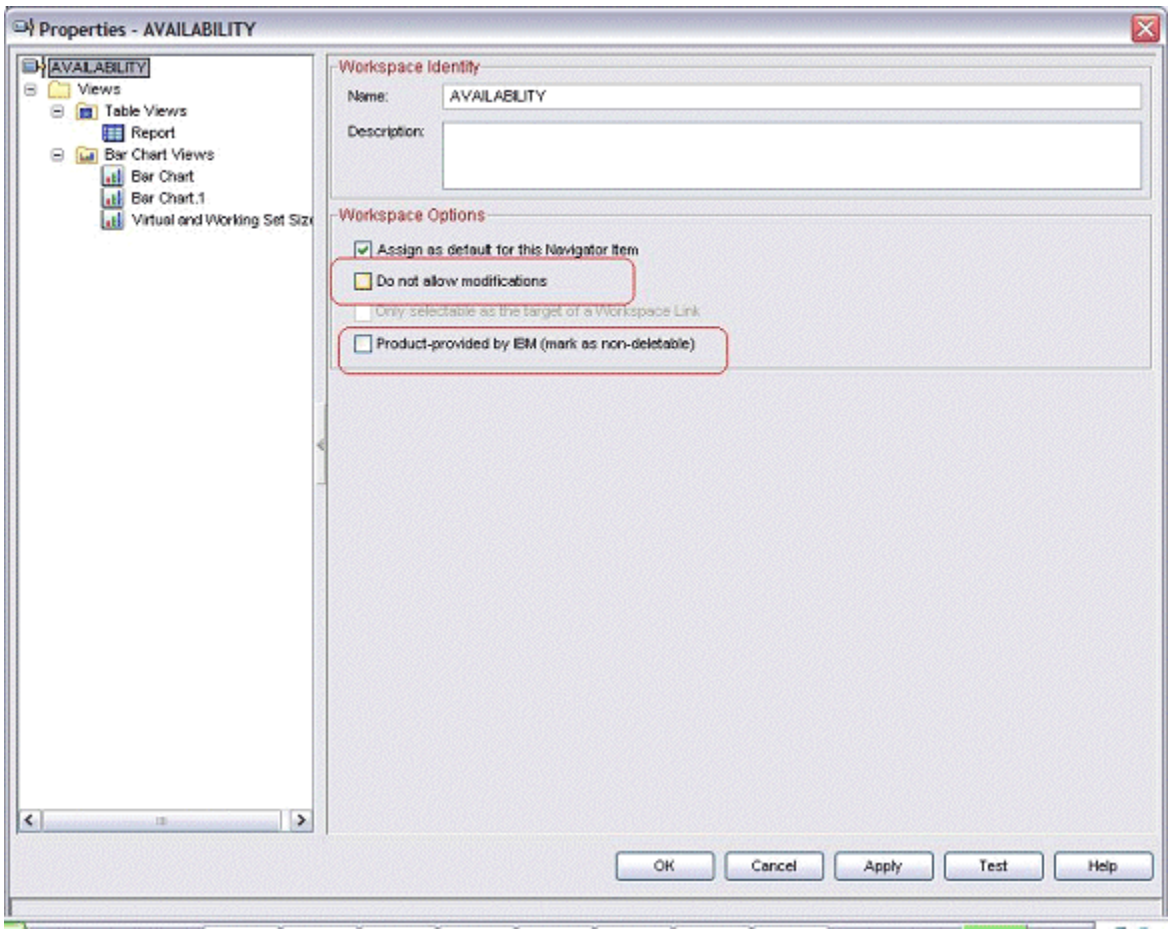


図 *65.* ワークスペース・プロパティーの設定 *(*続き*)*

Cloud APM 用のエージェントの準備

IBM Cloud Application Performance Management でエージェントを使用する必要がある場合は、「**ダッシュボード・セットアップ**」ウィザードを使用して、エージェントを準備する必要があります。このウィザードでは、Cloud APM の要約ダッシュボードおよび詳細ダッシュボードで確認できる情報を構成します。また、Cloud APM がエージェントで必要とする参照情報も設定されます。

始める前に

Cloud APM 用のエージェントを正常に準備するためには、エージェントが以下のデータを提供することを確認する必要があります。

- 1 行のデータを生成する 1 つ以上のデータ・セット (属性グループ)。これらのデータ・セットからの属性を使用して、要約ダッシュボードにデータを取り込むことができます。

  重要 : 要約ダッシュボードに情報を含めるには、単一行のデータを生成するデータ・セット内にその情報を指定する必要があります。一部のデータ・ソースでは、複数行のデータを生成するデータ・セットが作成されます。例えば、プロセス、Windows サービス、コマンド戻りコードのデータ・ソースの場合、複数の行を生成する可用性データ・セット内にデータが配置されます。この場合、要約ダッシュボードにデータを含めるには、単一行を生成するフィルタリングされたデータ・セットを作成する必要があります。説明については、1341 ページの『フィルタリング属性グループの作成』を参照してください。
- モニター対象サービスの状況 (正常、警告、重大などの状況値) を示す、これらのデータ・セットのいずれかに含まれている数値属性。この属性の状況重大度値を定義する必要があります。状況重大度値の定義については、1201 ページの『状況標識として使用される属性の重大度の指定』を参照してください。
- モニター対象アプリケーションがサービスを提供するポート番号が固定されている場合は、そのポートが分かっている必要があります。ポートがデプロイメントごとに変更される可能性がある場合は、1 行のデ

ータを生成するデータ・セットのいずれかに、ポートを示す数値フィールドが含まれている必要があります。

- 異なるホストで実行されているサーバーをモニターするためにホストにエージェントをインストールできる場合は、サーバー IP アドレスを示す、これらのデータ・セットのいずれかに含まれているストリング属性。エージェントが、その実行場所のホストを常にモニターする場合は、このような属性は不要です。

  ヒント: ホスト名を指定する属性が使用可能である場合は、nameToIpAddress 関数を使用して、IP アドレスの派生属性を作成できます。派生属性の作成については、1195 ページの『派生属性の作成』を参照してください。関数については、1207 ページの『ipAddressToName』を参照してください。

エージェントにサブノードがある 場合は、これらの要件はダッシュボードの作成対象となる各サブノードに適用されます。

このタスクについて

Cloud APM は、リソース をモニターします。 リソースは、エージェントのインスタンスに対応します。場合によっては、サブノードに対応することもあります。 リソースを定義するには、モニター対象のサービスに適用されるリソース・タイプ名、サーバー名、IP アドレス、ポート番号を指定する必要があります。

Cloud APM は、すべてのモニター対象リソースについて要約ダッシュボードを表示します。要約ダッシュボードには、状況標識が含まれています。この標識 (通常は、正常、警告、重大の各状況に対して緑、黄、赤) により、ユーザーはリソースの状況を一目で確認できます。同じダッシュボードに、他のいくつかの上位正常性メトリックが含まれていることがあります。

要約ダッシュボードでは、データは単一項目として表示されます。そのため、このデータが含まれているデータ・セットは 1 行のみを生成する必要があります。

オプションとして、エージェントに対して詳細ダッシュボードが使用可能な場合があります。ユーザーは要約ダッシュボードをクリックすることで、詳細ダッシュボードを表示できます。詳細ダッシュボードには、テーブルを表示することができます。そのため、任意のデータ・セットのデータを詳細ダッシュボード上で使用することができます。

要約ダッシュボード (状況標識を含む) および詳細ダッシュボードで表示する属性を選択する必要があります。

重要: 選択した属性内のデータは、毎分、エージェントから Cloud APM サーバーに自動的に渡されます。あまりにも多くのデータを指定すると、ネットワーク、サーバー、またはモニター対象ホストが過負荷状態になる可能性があります。必要な属性のみを選択するようにしてください。例えば、結合したデータ・セットまたは派生属性を表示する必要がある場合に、ソース属性も指定するようなことはしないでください。

重要: これらの属性以外のデータは Cloud APM に渡されません。エージェント・レベルでモニターされるしきい値を除いて、Cloud APM で他のデータを表示したり使用したりすることはできません。しきい値で他のデータを使用している場合、Cloud APM コンソールでしきい値状況を表示できないことがあります。

手順

1. 「**エージェント情報**」ビューで、「**ダッシュボード**」リンクをクリックします。
2. 「**ダッシュボード・コンポーネント**」で、「**ダッシュボードにエージェント・コンポーネントを表示**」を選択します。

   ヒント: または、IBM Tivoli Monitoring だけで使用するエージェントを作成する場合は、「**このエージェントにはダッシュボードが存在しません**」を選択することもできます。その場合、この操作の後続の手順は実行しないでください。 Cloud APM 環境では、上記のエージェントをインストールすることはできません。
3. 「**ダッシュボード・セットアップ・ウィザード**」リンクをクリックします。
4. エージェントにサブノードがある 場合は、Cloud APM でエージェントとサブノードのリソースの 配置を定義します。
   - 「**基本エージェント・インスタンス**」を選択して、基本エージェント (サブノードの外部のデータ) をリソースとして表示します。

- すべてのサブノードについて、「**サブノード "名前" インスタンス**」を選択して、このサブノードをリソースとして表示します。
- 必要に応じて、選択したサブノードで「**エージェントの子として表示**」を選択します。この場合、Cloud APM コンソールのエージェント・リソースの下に、サブノード・リソースがエージェントの子として一覧表示されます。

Cloud APM では、選択したコンポーネントごとに要約ダッシュボードおよび詳細ダッシュボードが表示されます。

重要 : ウィザードを再度実行し、エージェントまたはサブノードを選択解除しても、そのエージェントまたはサブノードのリソースは自動的に削除されません。こうしたリソースを削除するには、アウトライン・ビューで「**リソース**」を展開し、削除するリソースを選択して、キーボードの「Delete」キーを押します。

5. 「**属性選択 - 状況**」ページで、モニター対象サービスの状況を示す属性を選択します。単一データ行を返すグループからの数値属性が選択可能です。

   ヒント : あるいは、ダッシュボードに状況を表示したくない場合は、「**このエージェントの状況を指定**」の選択を解除します。
6. 同じページで、以下のように、要約ダッシュボードおよび詳細ダッシュボードに追加データを表示するかどうかを選択できます。
   - 要約ダッシュボードに追加の上位正常性メトリックを表示する場合は、「**このエージェントの要約情報に表示する追加属性を選択**」ボックスを必ず選択してください。そうでない場合は、ボックスをクリアします。
   - 詳細ダッシュボードに追加データを表示する場合は、「**このエージェントの詳細情報に表示する追加属性を選択**」ボックスを必ず選択してください。そうでない場合は、ボックスをクリアします。(通常、詳細ダッシュボードでは、モニター・エージェントが有意なものになるように十分なデータを表示する必要があるため、このボックスは選択することになります)。

   「**次へ**」をクリックします。
7. 「**このエージェントの要約情報に表示する追加属性を選択**」を選択した場合、「**属性選択 - 要約**」ページで、要約ダッシュボードに含める追加属性を最大で 4 つまで選択します。単一データ行を返すグループからの属性が選択可能です。「**次へ**」をクリックします。
8. 「**属性選択 - 詳細**」ページで「**このエージェントの詳細情報に表示する追加属性を選択**」を選択した場合は、詳細ダッシュボードに含める属性を選択します。エージェントのすべての属性を含めることができますが、パフォーマンスの低下を防ぐため、可能な限り少数の属性だけを含めるようにしてください。 「**次へ**」をクリックします。
9. 「**リソース・タイプ**」ページで、モニターするサーバー・タイプを入力します (例えば、「`Email サーバー`」や「`SampleCo データベース・サーバー`」)。「**次へ**」をクリックします。
10. 「**属性選択 - ソフトウェア・サーバー名**」ページで、「**固定名**」フィールドにソフトウェア・サーバーの固定名を入力するか、ソフトウェア・サーバー名を示す、エージェントからの属性を選択します。この名前は、この特定のモニター対象インスタンスのユーザーに対して表示されます (JBoss アプリケーション・サーバー・インスタンスの名前など)。「**次へ**」をクリックします。

    重要 : 同じモニター対象ホスト上で同じソフトウェア・サーバー名を持つモニター・エージェント、エージェント・インスタンス、またはサブノードを複数実行しないでください。エージェントにインスタンスまたはサブノードが含まれている場合、インスタンスまたはサブノードごとに固有のソフトウェア・サーバー名が生成されるようにしてください。2 つの異なるエージェントで同じソフトウェア・サーバー名が生成される場合は、同じモニター対象ホストにそれらをインストールしないでください。
11. 「**属性選択 - IP アドレス**」ページで、モニター対象サーバーまたはアプリケーションが使用する 1 次インターフェース接続の IP アドレス (ホスト名ではない) を示す、エージェントからの属性を選択します。例えば、HTTP サーバーの HTTP 接続や、データベース・サーバーのデータベース・クライアント接続です。あるいは、「**エージェントの IP アドレスを使用**」を選択して、エージェントが実行されているホストのアドレスを使用します。「**次へ**」をクリックします。
12. 「**属性選択 - ポート**」ページで、モニター対象アプリケーションがサービスを提供するポートを入力するか、このポートを示す、エージェントからの数値属性を選択します。「**完了**」をクリックします。

13. エージェントとサブノードの両方をリソースとして選択した場合や、複数のサブノードをリソースとして選択した場合は、「**次へ**」をクリックして、次のコンポーネント (エージェントまたはサブノード) のダッシュボードとリソースの情報を入力します。「**次へ**」ボタンが使用不可になっている場合は、必要なコンポーネントの情報がすべて入力済みであることを意味します。「**完了**」をクリックしてウィザードを終了してください。

タスクの結果

モニター対象ホストにエージェントをインストールした場合、「**状況の概要**」タブで、要約ダッシュボードと詳細ダッシュボードを表示することができます。

重要 : エージェントをインストールしてからダッシュボードが使用できるようになるまで、最大 30 分かかる場合があります。特に、このエージェント・タイプとバージョンを現在の環境に初めてインストールする場合は、時間がかかります。

エージェントの要約ダッシュボードをクリックすると、詳細ダッシュボードが表示されます。デフォルトでは、詳細ダッシュボード内のすべての情報が表として表示されます。

「**属性の詳細**」タブを使用して、表およびグラフとしてこの情報のカスタム表示を構成できます。

Agent Builder でのエージェントのテスト

Agent Builder を使用してエージェントを作成した後に、そのエージェントを Agent Builder でテストできます。

エージェントをテストして、予期したモニター・データが表示されるかどうかを確認する必要があります。エージェントをテストすることにより、エージェントの設定値を変更または調整して、表示されるデータを有益かつ正確なものにすることができます。

Agent Builder でのエージェントのテストは以下の方法で行えます。

1. Agent Builder の属性グループのテスト 機能を使用して、個々の属性グループを一度に 1 つずつテストします。詳しくは、1376 ページの『属性グループのテスト』を参照してください。
2. 属性グループのテストが完了したら、Agent Builder のエージェントのテスト 機能を使用して、エージェントのすべての属性グループをまとめてテストすることができます。詳しくは、1380 ページの『エージェントの完全テスト』を参照してください。

重要 : Agent Builder でエージェントをテストする際、数値属性の以下の特殊値を確認できます。

- -1: 一般エラー
- -2: 欠落データ
- -3: 値なし (例えば、データベースによってヌルが返された)

属性グループのテスト

属性グループのテストを使用して、Agent Builder で作成したエージェントの属性グループをテストすることができます。一度にテストできる属性グループは 1 つだけです。属性グループの定義を完了する前に、多くの属性グループをテストすることができます。例えば、新規エージェントの属性グループを定義している場合、「**IBM Tivoli Monitoring Agent ウィザード**」からテストを開始できます。また、既存のエージェントに属性グループを追加している場合、「**IBM Tivoli Monitoring Agent コンポーネント・ウィザード**」からテストを開始することもできます。

始める前に

属性グループのテストを始める前に、オプションで以下のことを実行できます。

- 属性グループのテスト 設定。詳しくは、1378 ページの『属性グループのテスト - 設定』を参照してください。
- 環境変数、構成プロパティー、および Java 情報 (該当する場合) の設定。詳しくは、1379 ページの『属性グループのテスト - 構成』を参照してください。

このタスクについて
Agent Builder は、ほとんどのデータ・ソースに対して属性グループ・テスト機能をサポートします

手順

- テスト・プロシージャーは以下の方法で始動します。

  1. エージェントまたは属性グループの作成中に、該当するデータ・ソースの「情報」ページで、「**テスト**」をクリックします。
  2. エージェントの作成後に、Agent Editor の「**データ・ソース定義**」ページで属性グループを選択して、「**テスト**」をクリックします。Agent Editor について詳しくは、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。

  上記の 2 つのステップのいずれかで「**テスト**」をクリックすると、「属性グループのテスト」ウィンドウが表示されます。このウィンドウは、データ・ソースに応じて異なります。

  Agent Builder は、ほとんどのデータ・ソースに対して属性グループ・テスト機能をサポートします。

  特定の属性グループのテスト手順について詳しくは、以下の『テスト』セクションを参照してください。

  - Windows Management Instrumentation (WMI)。WMI テスト手順について詳しくは、1226 ページの『WMI 属性グループのテスト』を参照してください。
  - Windows パフォーマンス・モニター (Perfmon)。Perfmon テスト手順について詳しくは、1228 ページの『Perfmon 属性グループのテスト』を参照してください。
  - Simple Network Management Protocol (SNMP)。SNMP テストについて詳しくは、1233 ページの『SNMP 属性グループのテスト』を参照してください。
  - Simple Network Management Protocol (SNMP) イベント送信側。SNMP イベント・テスト手順について詳しくは、1237 ページの『SNMP イベントの属性グループのテスト』を参照してください。
  - Java Management Extensions (JMX)。JMX テスト手順について詳しくは、1257 ページの『JMX 属性グループのテスト』を参照してください。
  - Common Information Model (CIM)。CIM テスト手順について詳しくは、1260 ページの『CIM 属性グループのテスト』を参照してください。
  - ログ・ファイル。ログ・ファイル・テスト手順について詳しくは、1270 ページの『ログ・ファイル属性グループのテスト』を参照してください。
  - スクリプトのテスト手順について詳しくは、1282 ページの『スクリプトの出力のモニターのステップ』を参照してください。
  - Java Database Connectivity (JDBC)。JDBC のテスト手順について詳しくは、1292 ページの『JDBC 属性グループのテスト』を参照してください。
  - Internet Control Message Protocol (ICMP) ping。ICMP テスト手順について詳しくは、1295 ページの『ping 属性グループのテスト』を参照してください。
  - Hypertext Transfer Protocol (HTTP) の可用性。HTTP テスト手順について詳しくは、1303 ページの『HTTP 属性グループのテスト』を参照してください。
  - SOAP のテスト手順について詳しくは、1311 ページの『SOAP 属性グループのテスト』を参照してください。
  - 伝送制御プロトコル (TCP) ソケット。ソケット・テスト手順について詳しくは、1321 ページの『ソケット属性グループのテスト』を参照してください。
  - Java アプリケーション・プログラミング・インターフェース (API)。Java API のテスト手順について詳しくは、1335 ページの『Java アプリケーション属性グループのテスト』を参照してください。

  属性グループ・テスト機能がないデータ・ソースもあります。以下に例を示します。

  - Agent Builder のブラウザーを使用して、システム上で稼働中のデータを表示できる場合。例えば、システム上で現在実行されているプロセスを表示することができます。別の例として、システムにインストールされているサービス (Windows サービス) と、システム上に存在する Windows イベント・ログを表示することができます。

– エージェント内でカスタマイズをほとんどまたはまったく実行できない場合 (AIX バイナリー・ログ、コマンドの戻りコード)。
– 結合およびフィルタリングされた属性グループは、属性グループのテスト機能を使用してテストすることはできません。これは、これらのグループが複数の属性グループに基づいているためです。

注 :

1. 属性グループのテスト機能ではテストできないデータ・ソースをテストするには、エージェントの完全テストを使用します。エージェントの完全テストについて詳しくは、1380 ページの『エージェントの完全テスト』を参照してください。
2. 「**データの収集**」をクリックした後にデータ・ソースをテストすると、最初のクリックの後に、データがまったく表示されないか、データが最新ではないことがあります。このような場合、もう一度「**データの収集**」をクリックすると、最新データが表示されます。

- デバッグ:

  テストされる各データ・ソースには、Agent Builder により作成されるテスト・ディレクトリーが存在します。このディレクトリーは、データ・ソースのテストのランタイム環境用に使用されます。データ・ソースに対して実行されるテストに関連するログ・ファイルが、このディレクトリーに保管されます。これらのログ・ファイルは、テスト中に検出された問題のデバッグに役立ちます。

  注 :

  1. テスト・ログ・ファイルのロケーションは、「**エージェントの開始**」をクリックした後および「**エージェントの停止**」をクリックした後に、「**テスト**」ウィンドウに状況メッセージとして表示されます。
  2. Agent Builder がシャットダウンされると、すべてのテスト・データ・ソース・ディレクトリーが削除されます。

属性グループのテスト - 設定

属性グループをテストする 前に設定を行います。

このタスクについて

属性グループのテストを始める前に、テスト中の属性の処理方法を決定するいくつかの設定をオプションで行うことができます。

手順

1. Agent Builder のメニュー・バーから「**ウィンドウ**」 > 「**設定**」を選択します。
   「**設定**」ウィンドウが開きます。
2. 「**Agent Builder**」を選択します。

   テスト属性グループと関連付けられた設定が表示されます。

   テスト中に変更されたデータ・タイプのダイアログを表示する
   このオプションを選択した場合は、属性のデータ・タイプに対する変更が Agent Builder から提案されます。Agent Builder が変更を提案するのは、属性のデータ・タイプが、その属性のテストによって返されたデータと一致しない場合です。例えば、属性に対して定義されたストリングの長さが短すぎて、テストによって返された値を保持できない場合などです。 この例の場合は、属性を再定義してストリングをもっと長くするように Agent Builder から提案されます。このオプションをクリアした場合、テスト中に Agent Builder によるデータ・タイプのチェックと提案は行われません。このオプションは、デフォルトで選択されています。

   作成されるスクリプトまたはログ属性の最大数 (Maximum script or log attributes created)
   このフィールドに入力した値によって、ログ・ファイルまたはスクリプト属性グループの初期テスト中に Agent Builder が解析する属性の最大数が決まります。デフォルトは 25 です。
3. 設定が完了したら、「**OK**」をクリックして設定を保存し、「**設定**」ウィンドウを閉じます。
   デフォルト設定に戻す場合は、「**デフォルトの復元**」をクリックしてから「**OK**」をクリックします。

属性グループのテスト - 構成

属性グループをテストする 前に、環境変数、構成プロパティー、および Java 情報を設定します。

このタスクについて

属性グループのテストを始める前に、オプションでデータ・ソースの「テスト」ウィンドウから環境変数、構成プロパティー、および Java 情報 (該当する場合) を設定することができます。Java 情報は構成データのサブセットです。一部の環境変数は、属性グループのテスト用にデフォルトで設定された特殊値を持っています。属性グループのテスト用の特殊値を持つ環境変数について詳しくは、1384 ページの『テスト環境変数』を参照してください。

手順

1. オプション: データ・ソースの「**テスト**」ウィンドウで、「**環境の設定**」をクリックします。
   「**環境変数**」ウィンドウが開きます。データが取り込まれると、「**環境変数**」ウィンドウにはテスト実行中に使用されるすべての環境変数がリストされます。「環境変数」ウィンドウの初期表示には、エージェントで定義されたすべての既存の環境変数が含まれます。また、このエージェントの以前のテストから追加された環境変数もすべて表示されます。
   a) 「**追加**」または「**編集**」をクリックして、個々の変数を追加または編集します。
   b) 「**削除**」をクリックして個々の変数を削除するか、「**デフォルトの復元**」をクリックしてデフォルト変数を復元し、残りの変数をすべて削除します。
   c) 「**OK**」をクリックして変更内容を保存し、「**テスト**」ウィンドウに戻ります。
2. オプション: データ・ソースの「**テスト**」ウィンドウで、「**構成**」をクリックします。「**ランタイム構成**」ウィンドウが開きます。
   a) 「**エージェント構成の編集**」をクリックし、「**構成プロパティー**」ウィンドウを使用して、構成プロパティーの追加や既存のエージェント構成プロパティーの編集を行います。
   b) 構成プロパティーを選択して「**編集**」をクリックし、テスト中の属性グループに関連する既存の構成プロパティーを編集します。
   c) 構成プロパティーを選択して「**デフォルトの復元**」をクリックし、構成プロパティーをデフォルト値に復元します。

   重要 : JMX データ・ソースをリモートの WebSphere Application Server に接続する場合、ローカルの WebSphere Application Server をインストールし、Java の場所をサーバーが使用する JRE に設定してください。接続の設定方法については、1239 ページの『Java Management Extensions (JMX) MBean のモニター』を参照してください。
3. 「**OK**」をクリックして変更内容を保存し、「**テスト**」ウィンドウに戻ります。
4. **注 :** 以下のタイプの属性グループの Java 情報を設定できます。
   - Java Management Extensions (JMX)
   - Java データベース・コネクティビティー (JDBC)
   - Hypertext Transfer Protocol (HTTP) の可用性
   - SOAP
   - Java アプリケーション・プログラミング・インターフェース (API)

   Java 情報は、ステップ 1379 ページの『2』 で説明されている構成データのサブセットです。

   オプション: データ・ソースの「**テスト**」ウィンドウで、「**Java 情報**」をクリックします。

   「**Java 情報**」ウィンドウが開きます。
   a) Java 情報を入力します。
   例えば、Java ランタイム環境 (JRE) のロケーションを参照または入力するか、「**Java トレース・レベル**」でレベルを選択するか、「**JVM 引数**」に引数を入力します。
   b) 「**OK**」をクリックして変更内容を保存し、「**テスト**」ウィンドウに戻ります。

エージェントの完全テスト

エージェントの完全テストを使用して、エージェントのすべての属性グループをまとめてテストします。また、属性グループのテスト 機能を使用したテストが不可能なデータ・ソースをテストするために、エージェントの完全テストを使用することもできます。

このタスクについて

エージェントの完全テストを使用して、IBM Tivoli Monitoring のインストールを必要とせずに、IBM Tivoli Monitoring で実行する場合と同じ方法でエージェントを実行できます。

重要 : Windows システムでは、Agent Builder 内でエージェントの完全テストを実行する場合 (1380 ページの『エージェントの完全テスト』を参照)、Agent Builder を実行する 32 ビット・バージョンのオペレーティング・システム (つまり、32 ビットの Windows) を「エージェント情報」ウィンドウで必ず選択してください。Linux システムでは、64 ビット・バージョンを選択する必要があります。

手順

1. 「**エージェントのテスト**」パースペクティブを開きます。
   a) Agent Editor で、「**エージェント情報**」タブを開きます。
   b) 「**エージェントのテスト**」をクリックします。

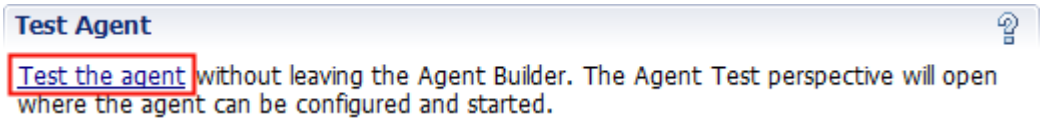


図 66. ***Agent Editor*** の**「エージェント情報」**ページの**「エージェントのテスト」**セクション

   または、Agent Builder メニューから「**ウィンドウ**」 > 「**パースペクティブを開く**」 > 「**その他**」を選択し、「**エージェントのテスト**」を選択して「**OK**」をクリックします。

「**エージェントのテスト**」パースペクティブが開きます (1382 ページの図 68)。「**エージェントのテスト**」ビューに、Agent Editor で開いたエージェントが表示されます。これらのエージェントをテストすることができます。また、「**属性グループのテスト**」ビューも表示されます。このビューは最初は空です。「**属性グループのテスト**」ビューには、エージェントの実行時に、選択された属性グループから収集したデータが表示されます。

ヒント : 編集しているエージェントがない場合は、「**エージェントのテスト**」パースペクティブは空です。ビューにデータを取り込むには、**IBM Tivoli Monitoring** パースペクティブに移動して、**Agent Editor** でエージェントを開きます。**Agent Editor** でエージェントを開いたら、「**エージェントのテスト**」パースペクティブに戻ってエージェントをテストします。

2. オプション: テストを開始する前に、環境変数と構成プロパティーを構成します。

「**エージェントのテスト**」ビューから、以下の 2 つの方法で「**環境変数**」ウィンドウおよび「**ランタイム構成**」ウィンドウにアクセスできます。

- 「**エージェントのテスト**」ビューでエージェントを右クリックして、選択メニューを開きます。メニューから「**環境の設定**」を選択して、「**環境変数**」ウィンドウを開くことができます。メニューから「**構成**」を選択して、「**ランタイム構成**」ウィンドウを開くことができます。
- 「**エージェントのテスト**」ビューのツールバーにあるビュー・メニュー・アイコン をクリックして、上記の選択と同様に、「**環境の設定**」および「**構成**」の各メニュー項目にアクセスします。

「**環境変数**」および「**ランタイム構成**」ウィンドウの使用について詳しくは、1376 ページの『属性グループのテスト』を参照してください。

重要 :

a. エージェントには、テストされた各属性グループに関連する構成の最終セットが自動的に取り込まれます。
b. 一部の環境変数には、属性グループのテスト用およびエージェントの完全テスト用の異なるデフォルト値を設定できます。属性グループのテスト用の特殊値を持つ環境変数について詳しくは、1384 ページの『テスト環境変数』を参照してください。

c. JMX データ・ソースをリモートの WebSphere Application Server に接続する場合、ローカルの WebSphere Application Server をインストールし、Java の場所をサーバーが使用する JRE に設定してください。接続の設定方法については、1239 ページの『Java Management Extensions (JMX) MBean のモニター』を参照してください。

d. Java API、JDBC、JMX、HTTP、または SOAP データ・ソースで「**Java**」 > 「**JVM 引数**」設定を使用して、エージェント・トレース・ログを制御することができます。以下の値を設定します。

```
-DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
```

files は、保持されるトレース・ログ・ファイルの最大数 (デフォルト値は 4) で、*size* は、ログ・ファイルの最大サイズ (キロバイト単位、デフォルト値は 5000) です。例えば、以下の値を設定することができます。

```
-DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
```

この場合、エージェントは最初のログ・ファイルに 100 キロバイト分のデータを書き込み、2 番目以降のログ・ファイルにも同じサイズのデータを書き込んでいくことになります。7 つのログ・ファイルすべてに 100 キロバイト分のデータが書き込まれると、次は最初のログ・ファイルに戻ってデータが上書きされます。

e. エージェントにサブノードが含まれている場合、インストールされているバージョンで、基本エージェントの属性グループに対して、異なるサブノードの異なる構成値を個別に設定することができます。ただし、「エージェントの完全テスト」構成の場合、各構成値を設定できるのは 1 回だけです。設定した値は、基本エージェントとすべてのサブノードに適用されます。各サブノードについてテストできるインスタンスは 1 つだけです。

3. 「**エージェントのテスト**」ビューで、テストするエージェントを選択し、 「**エージェントの開始**」アイコンをクリックします。

ウィンドウに、エージェントが開始されていることが表示されます。エージェントが開始されると、「**エージェントのテスト**」ビューに、エージェントの属性グループがエージェントの子として表示されます。属性グループは属性グループ・アイコン によって示されます。

エージェントに関する情報を提供する状況属性グループ (「**パフォーマンス・オブジェクト状況**」、「**スレッド・プールの状況**」、および「**アクション実行の状況**」) も、「**エージェントのテスト**」ビューにエージェントの子として表示されます。 状況属性グループは、 情報アイコンによって示されます。

同時に複数のエージェントを開始して実行できます。

エージェントが開始されると、 「**エージェントの停止**」アイコンが使用可能になります。

エージェントがサブノードまたはナビゲーター・グループを持っている場合、それらは、「**エージェントのテスト**」ビューにノードとして表示されます。エージェントの下にサブノード定義が表示されます。サブノード・インスタンス・ノードはサブノード定義ノードの下に表示されます。属性グループとナビゲーター・グループは、サブノード・インスタンス・ノードの下に表示されます。例:

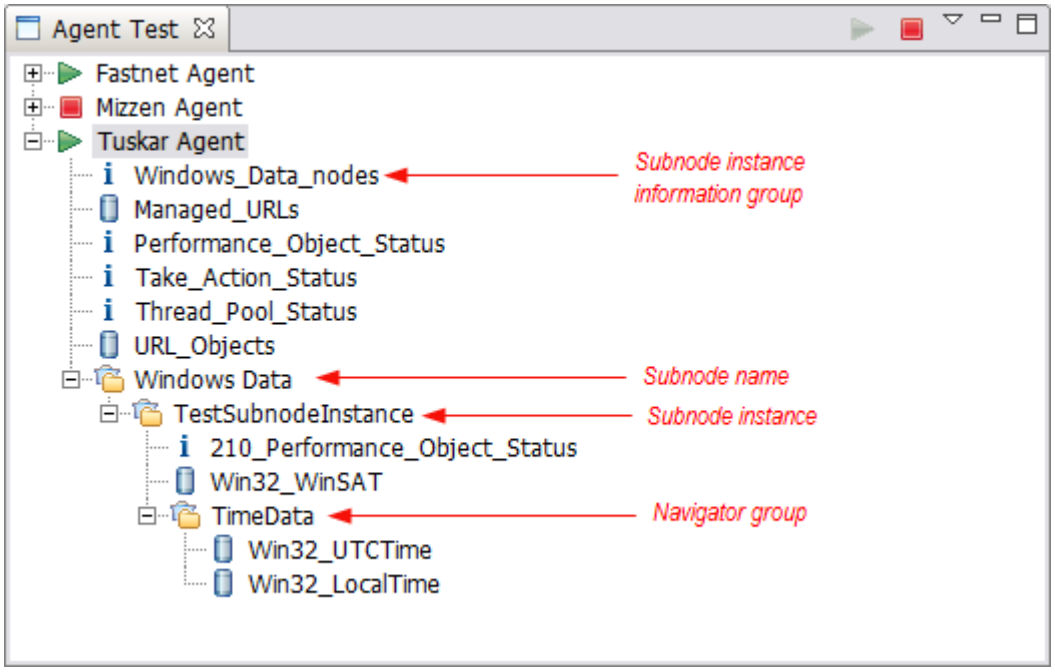


図 67. **「エージェントのテスト」**ビュー。サブノードおよびナビゲーター・グループの例が強調表示されています。

「**エージェントのテスト**」ビューで任意のノードを右クリックして、メニュー選択項目 (「**編集**」および「**エージェントの停止**」など) にアクセスできます。「**編集**」を選択すると、選択されたノードの「**データ・ソース定義**」が **Agent Editor** で開きます。

注 : Agent Editor で行った変更は、エージェントを停止して再始動するまで、実行中のエージェントには表示されません。

4. 「**エージェントのテスト**」ビューで、テストする最初の属性グループを選択します。

属性グループを選択すると、選択した属性グループについてデータ収集が開始されます。収集に時間がかかる場合は、ウィンドウにデータ収集が進行中であることが表示されます。データ収集が完了すると、収集されたデータが「**属性グループのテスト**」ビューに表示されます。以下に例を示します。

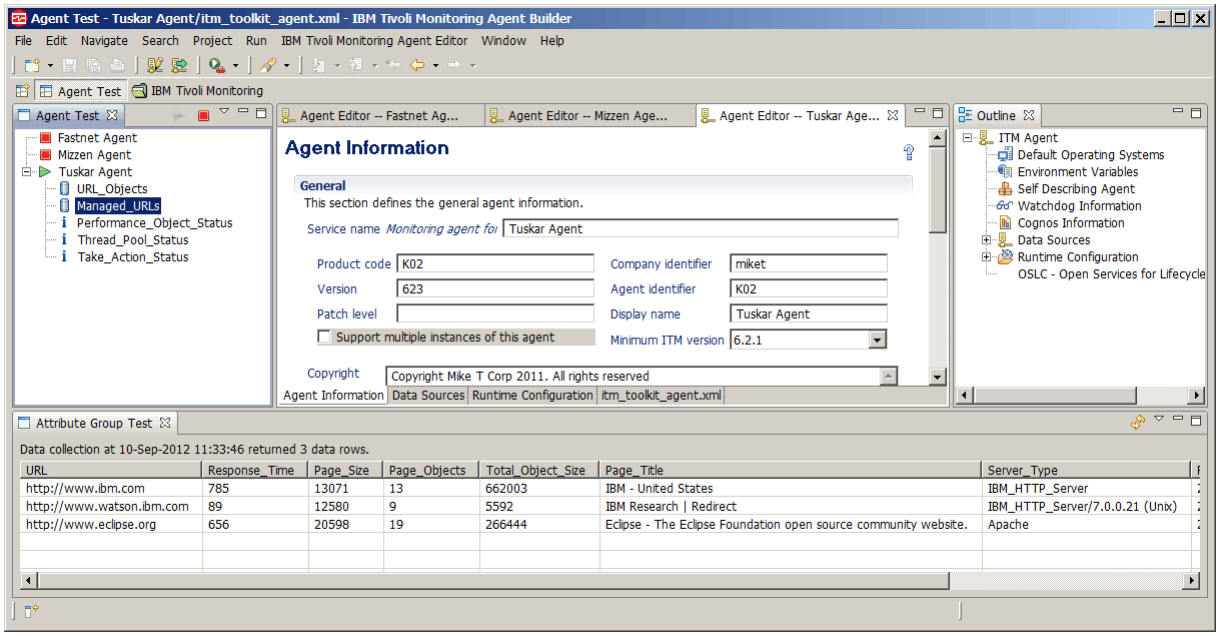


図 68. **「エージェントのテスト」**パースペクティブ

データが表示されない場合は、「データ行が返されませんでした」というメッセージが「**属性グループのテスト**」ビューに表示されます。エージェントからデータが返されない原因は複数あります。 以下のような原因があります。

- データがない
- 定義が正しくない
- 構成が正しくない

データが返されない原因を確認するには、「**パフォーマンス・オブジェクトの状況**」属性グループの「**エラー・コード**」の値を調べます。「**パフォーマンス・オブジェクトの状況**」属性グループの表示について詳しくは、ステップ 1383 ページの『9』 を参照してください。

実行中のエージェント内にある別の属性グループのデータを収集するには、対象の属性グループを選択します。

「**エージェントのテスト**」ビューで属性グループを選択すると、対応する属性グループが **Agent Editor** ビューに表示されます。

5. オプション: ある種の属性グループ・タイプでは、有用なデータ値を得るために、初回のデータ収集後に 2 回目のデータ収集を実行します。

   データ収集を実行するには、「**属性グループのテスト**」ビューでデータの収集アイコン をクリックします。

   収集に時間がかかる場合は、ウィンドウにデータ収集が進行中であることが表示されます。 データ収集が完了すると、新たに収集されたデータが「**属性グループのテスト**」ビューに表示されます。
6. オプション: 「**属性グループのテスト**」ビューで属性列見出しをクリックすると、「**属性情報**」が **Agent Editor** の「**データ・ソース定義**」タブで開きます。表内の任意のデータ・セルを右クリックし、メニューから「**編集**」を選択することにより、同じ「**属性情報**」にアクセスすることもできます。

   属性のプロパティーは、通常の方法で編集できます。行った変更は、エージェントを停止して再始動するまで、実行中のエージェントには表示されません。
7. オプション: 一度に複数の「**属性グループのテスト**」ビューを開きます。

   「**属性グループのテスト**」ビューをもう 1 つ開くには、「**属性グループのテスト**」ビューのツールバーにあるビュー・メニュー・アイコン をクリックしてから、「**属性グループのビューを開く**」を選択します。

   注 : 追加の「**属性グループのテスト**」ビューを開くと、元の「**属性グループのテスト**」ビューと同じ属性情報が表示されます。そのときに「**エージェントのテスト**」ビューで別の属性グループを選択すると、元の「**属性グループのテスト**」ビューに別の属性グループ情報が表示されます。別の「**属性グループのテスト**」ビューを最初に開くと、元のビューと同じ場所にそのビューのタブとともに表示されます。 2 つのビューを同時に表示するには、タブをワークスペース内の別の場所にドラッグします。
8. オプション: エージェントにサブノードがある 場合は、サブノード・インスタンス情報属性グループを選択して、エージェント内でサブノードがどのようにリストされるかを確認することができます (1382 ページの図 67)。サブノード・インスタンス情報属性グループを選択すると、「**属性グループのテスト**」ビューに (選択したタイプのすべてのオンライン・サブノードについて) サブノード・インスタンス情報が表示されます。
9. オプション: エージェントの動作に関する詳細情報を表示するには、「**エージェントのテスト**」ビューで「**パフォーマンス・オブジェクト状況**」および「**スレッド・プールの状況**」の各属性グループを選択します。これらの状況属性グループは、情報アイコン によって示されます。これらのグループを選択すると、自分の属性グループについて、以前のデータ収集に関する状況情報が表示されます。

   例:

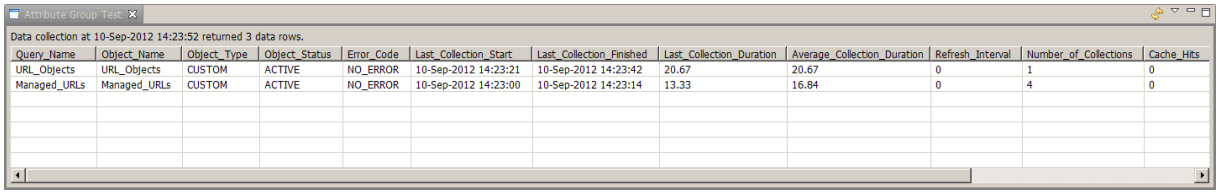


図 *69.* ***Managed_URLs*** および ***Managed_Nodes*** 属性グループのデータ収集に関する詳細情報 (パフォーマンス・オブジェクト状況) を示す**「属性グループのテスト」**ビュー

10. エージェントのテストが完了したら、エージェントの停止アイコン をクリックします。

テスト環境変数

これらの環境変数を使用して、テスト時のエージェントの動作を制御します。

環境変数は、エージェントの実行方法を決定する名前付きの動的な値です。属性グループのテストでは、一部のエージェント環境変数に特殊値が設定されています。エージェントが単一の属性グループのテストに適した方法で応答するように、これらの特殊値が使用されます。エージェントの完全テストでは、特殊値は使用されず、代わりにデフォルト値が使用されます。デフォルト値の使用によりエージェントは通常どおり動作します。エージェントの完全テストには通常の動作の方が適しています。

以下の表に、属性グループのテスト用の特殊値を持つ環境変数が要約されています。すべてのエージェントの環境変数について詳しくは、1178 ページの『環境変数のリスト』を参照してください。環境変数の設定について詳しくは、1177 ページの『環境変数』を参照してください。

表 *299.* 環境変数

環境変数	デフォルト値 (エージェントの完全テスト)	属性グループのテスト値	属性グループのテスト用に値が変更された理由
CDP_DP_INITIAL_COLLECTION_ DELAY	可変	1	この値は、スレッド・プールを持つエージェントに適用されます。この値は、初回のデータ収集要求がデータ・プロバイダーに送信されるまでスレッド・プールが待機する時間 (秒) です。 **注 :** CDP_DP_INITIAL_COLLECTION_DELAY を設定しなかった場合、スレッド・プールは、CDP_DP_REFRESH_INTERVAL または CDP_ATTRIBUTE_GROUP_REFRESH_INTERVAL で指定された時間が過ぎるまで待機します。この待機時間は、スレッド・プールがデータ収集間に待機する時間と同じです。この時間が長すぎると、最初のデータ収集まで待機できない場合があります。
CDP_DP_CACHE_TTL	55	1	1 に設定すると、データ収集要求により、データ・プロバイダーがデータを即時に収集する可能性が高くなります。 1 に設定しないと、最大 60 秒間キャッシュに入れられたデータが返されます。

テストおよび使用のためのモニター・インフラストラクチャーへのエージェントのインストール

Agent Builder でエージェントをテストした後、既存の IBM Tivoli Monitoring 環境または IBM Cloud Application Performance Management 環境にエージェントをインストールして、さらにテストを重ねたり使用したりすることができます。

モニター・インフラストラクチャーでエージェントをインストールしてテストすることには、以下の利点があります。

- エージェントの同時に実行される複数のインスタンスを構成およびテストできます。
- サブノードの同時に実行される複数のインスタンスを構成およびテストできます。
- Tivoli Monitoring 環境において、Tivoli Enterprise Portal でワークスペース、シチュエーション、アクション、照会をビルドできます。

重要 : 初期バージョンのエージェントをテスト・バージョンのモニター・インフラストラクチャーにデプロイしてください。Tivoli Monitoring では、別個のモニター・サーバーとポータル・サーバーを使用します。Cloud APM では、テスト・クラウド・アカウントを使用するか、オンプレミス・モニター・サーバーの個別のテスト・デプロイメントを使用します。最終バージョンのエージェントを実稼働インフラストラクチャーにデプロイします。

任意のバージョンのエージェントを実稼働モニター・インフラストラクチャーにデプロイした後にエージェントのデータ・セットを変更すると、新規バージョンがサーバー上の古いバージョンと競合する可能性があります。この場合、どのバージョンのエージェントも使用できなくなる可能性があります。

エージェントのインストール

Agent Builder で作成するエージェントをインストールするための方法は 2 つあります。

1. Agent Builder と同じシステム上で稼働しているモニター・インフラストラクチャーでエージェントをテストするために、エージェントをローカルの Tivoli Monitoring インストール済み環境または Cloud APM インストール済み環境にインストールできます。
2. Agent Builder と同じシステム上では稼働していない Tivoli Monitoring システム、または Cloud APM システムでエージェントをテストまたは使用するために、他のシステムに転送可能な圧縮ファイル (エージェント・パッケージ) を生成して、デプロイできます。

注 :

1. Tivoli Monitoring では、エージェントのインストール後、パフォーマンス・メトリックは Tivoli Enterprise Portal の表で確認することができます。 シチュエーションまたはワークスペースのサポートについては、1403 ページの『アプリケーション・サポート・ファイルのインポート』を参照してください。
2. Tivoli Monitoring では、エージェントのインストール後、Tivoli Enterprise Portal を使用して、エージェントからのデータを検査することができます。詳しくは、1397 ページの『Tivoli Enterprise Portal の変更点』を参照してください。Tivoli Enterprise Portal でデータを表示した後でエージェントを変更する場合は、1175 ページの『Agent Editor を使用したエージェントの変更』を参照してください。
3. Linux または UNIX をサポートするエージェントの場合、Linux または UNIX システムにインストーラー・イメージを生成します。Linux または UNIX システムによって、適切なアクセス権のあるファイルが作成されます。

エージェントのローカル・インストール

Agent Builder が実行されているローカル・システムのモニター環境にエージェントをインストールします。

このタスクについて

次の手順を実行して、ローカル・システムのモニター環境にエージェントをインストールします。

1. 以下のいずれかの方法を使用して、Agent Builder の「プロジェクト・エクスプローラー」のナビゲーション・ツリーから `itm_toolkit_agent.xml` ファイルをクリックします。

a. itm_toolkit_agent.xml ファイルを右クリックして、「**IBM**」 > 「**エージェントの生成**」と選択します。

b. itm_toolkit_agent.xml ファイルを選択して、ツールバーにある 「**エージェントの生成**」アイコンを選択します。

c. itm_toolkit_agent.xml ファイルをダブルクリックして、「**Agent Editor**」 > 「**エージェントの生成**」を選択します。

2. 「**エージェントの生成ウィザード**」ウィンドウの「**エージェントのローカル・インストール**」セクションで、モニター・インフラストラクチャーのインストール・ディレクトリーを入力します。Agent Builder によって CANDLE_HOME 環境変数から検出した値が入力されます。この変数が設定されていない場合、Windows のデフォルト値である C:¥IBM¥ITM が表示されます。

   チェック・ボックスが使用可能になる状況を以下に示します。

   エージェントのインストール
   Agent Builder が適切な Tivoli Enterprise Monitoring Agent または IBM Cloud APM エージェントを指定の場所で検出すると、有効になります。適切なエージェントとは、ローカル・オペレーティング・システムをサポートする、正しい最小バージョンであるエージェントです。

   TEMS サポートのインストール
   Agent Builder が Tivoli Enterprise Monitoring Server を指定の場所で検出すると、Tivoli Monitoring 環境で有効になります。

   TEPS サポートのインストール
   Agent Builder が Tivoli Enterprise Portal Server を指定の場所で検出すると、Tivoli Monitoring 環境で有効になります。

3. インストールするコンポーネント (エージェント、Tivoli Enterprise Monitoring Server サポート、Tivoli Enterprise Portal Server サポート) を選択します。

4. Tivoli Monitoring 環境で、Tivoli Enterprise Monitoring Server または Tivoli Enterprise Portal Server がローカル・コンピューターにインストールされていて、これらのサーバー用のサポート・ファイルをインストールする場合、これらのサーバーを再始動するかどうかを選択できます。

   この場合、「エージェントの生成」ウィザードの「**エージェントのローカル・インストール**」セクションで「**資格情報を使用せずに TEMS を再始動する**」チェック・ボックスと「**TEPS を再始動する**」チェック・ボックスがアクティブになります。サーバーをリサイクルせずにサポートをインストールする 場合は、これらのチェック・ボックスをクリアします。

   「**資格情報を使用せずに TEMS を再始動する**」チェック・ボックスをクリアすると、Tivoli Enterprise Monitoring Server のユーザー ID とパスワードの入力を求めるプロンプトが出されます。これらの詳細情報を入力して「**ログオン**」をクリックします。セキュリティーをオフにして TivoliMonitoring を実行している場合は、ユーザー ID として「sysadmin」を入力し、パスワードをブランクのままにして、「**ログオン**」をクリックします。

   あるいは、資格情報を入力せずに操作を続けるには、ユーザー ID とパスワードを指定せずに「**ログオン**」をクリックするか、「**キャンセル**」をクリックします。これらのステップを完了すると、Tivoli Enterprise Monitoring Server がリサイクルされます。

   重要 : Tivoli Enterprise Monitoring Server をリサイクルせずにサポート・ファイルをインストールするには、Tivoli Enterprise Monitoring Server が確実に稼動しているようにしてください。

5. 生成するエージェント・コンポーネントを選択します。「**基本エージェント**」または「**Cognos レポート**」、あるいはその両方を選択できます。

6. IBM Cloud APM 環境で、自己記述型エージェントのセキュリティー署名を指定できます。「**すべての JAR 署名設定を編集**」をクリックします。署名した JAR ファイルにタイム・スタンプを追加し、時刻認証局を指定できます。ご使用の Java 鍵ストア・ファイルに関する詳細を指定します。

   注 : Java ツールを使用して Java 鍵ストア・ファイルを作成する必要があります。例えば、秘密鍵と証明書を対応する公開鍵と共に Java 鍵ストア・ファイルで生成するには、以下のコマンドを実行します。

   - *ab_install_path*/jre/bin/keytool -genkeypair -keystore *keystore_file_path* -storepass *key_store_password* -alias *key_store_alias* -dname "CN=*common_name*,

```
OU=organizational_unit, L=city_or_locality, ST=state_or_province,
C=country" -keypass key_password
```

各部の意味は次のとおりです。

- *ab_install_path* は、Agent Builder がインストールされているロケーションです。
- *keystore_file_path* は、既存の JKS 鍵ストアが存在するロケーションのパス、または JKS 鍵ストアが作成されるロケーションのパスです。
- *key_store_password* は、この鍵ストア内のすべての項目にアクセスするために必要なパスワードです。
- *key_store_alias* は、鍵ストア内でこの鍵を識別する名前です (デフォルトは "mykey")。
- *key_password* は、この特定の鍵にアクセスするために必要なパスワードです (デフォルトは key_store_password)。

証明書は、サーバーの鍵ストアに含める必要があります。

7. 「**JAR 署名**」の詳細を入力したら、「**OK**」をクリックします。
8. 「**完了**」をクリックします。
9. エージェントを構成および開始します。詳しくは、1390 ページの『IBM Tivoli Monitoring 環境でのエージェントの構成と開始』を参照してください。IBM Cloud APM 環境については、1391 ページの『エージェントの構成』および 1393 ページの『エージェントの開始および停止』を参照してください。

   Tivoli Monitoring v6.2 FP1 以降の場合、サーバーを再始動せずに、Tivoli Enterprise Monitoring Server サポートおよび Tivoli Enterprise Portal Server サポートをインストールできます。この場合、「エージェントの生成」ウィザードの「**エージェントのローカル・インストール**」セクションで「**資格情報を使用せずに TEMS を再始動する**」チェック・ボックスと「**TEPS を再始動する**」チェック・ボックスがアクティブになります。サーバーをリサイクルせずにサポートをインストールする 場合は、これらのチェック・ボックスをクリアします。「**資格情報を使用せずに TEMS を再始動する**」チェック・ボックスをクリアすると、Tivoli Enterprise Monitoring Server のユーザー ID とパスワードの入力を求めるプロンプトが出されます。Tivoli Enterprise Monitoring Server のユーザー ID とパスワードを入力して、「**ログオン**」をクリックしてください。セキュリティーをオフにして TivoliMonitoring を実行している場合は、ユーザー ID として「sysadmin」を入力し、パスワードをブランクのままにして、「**ログオン**」をクリックします。資格情報を入力しないで操作を続ける (つまり、ユーザー ID とパスワードを指定しないで「**ログオン**」をクリックするか、「**キャンセル**」をクリックする) ことも可能です。その場合は、Tivoli Enterprise Monitoring Server がリサイクルされます。

   注 : Tivoli Enterprise Monitoring Server をリサイクルしないでサポート・ファイルをインストールするには、Tivoli Enterprise Monitoring Server を実行しておく必要があります。

エージェント・パッケージの作成

Agent Builder を使用して、エージェント・インストール用の圧縮パッケージを作成できます。

このタスクについて

エージェント・パッケージには、エージェントの実行に必要なすべてのファイルのほか、インストール・スクリプトと構成スクリプトも格納されます。パッケージには、モニター環境用のサポート・ファイルも組み込まれます。

エージェント・パッケージを使用すると、IBM Tivoli Monitoring 環境および IBM Cloud Application Performance Management 環境にエージェントをインストールできます。

手順

1. 以下のいずれかの方法を使用して、Agent Builder の「**プロジェクト・エクスプローラー**」のナビゲーション・ツリーから itm_toolkit_agent.xml ファイルをクリックします。
   - itm_toolkit_agent.xml ファイルを右クリックして、「**IBM**」 > 「**エージェントの生成**」と選択します。

- `itm_toolkit_agent.xml` ファイルを選択して、ツールバーにある 「**エージェントの生成**」アイコンを選択します。
- `itm_toolkit_agent.xml` ファイルをダブルクリックして、「**Agent Editor**」 > 「**エージェントの生成**」を選択します。

2. 「**エージェント・イメージの生成**」セクションで、出力 (圧縮したパッケージまたは展開したファイル) を格納するディレクトリーの名前を入力します。
3. 生成された展開ファイルを zip ファイルまたは tar ファイルとは別に保持する場合は、「**中間ファイルの保持**」チェック・ボックスを選択します。
4. 「**ZIP ファイルの作成**」チェック・ボックスを選択して、指定したディレクトリーに圧縮ファイルを作成します。Windows システムでの圧縮 zip ファイルのデフォルトの名前は `smai-`*`agent_name-version`*`.zip` です。
5. 「**TAR ファイルの作成**」チェック・ボックスを選択して、指定したディレクトリーに tar ファイルを作成します。UNIX システムおよび Linux システムでの圧縮 tar ファイルのデフォルトの名前は `smai-`*`agent_name-version`*`.tgz` です。
6. 生成するエージェント・コンポーネントを選択します。「**基本エージェント**」または「**Cognos レポート**」、あるいはその両方を選択できます。

   重要 : IBM Cloud Application Performance Management 環境では「**Cognos レポート**」を選択しないでください。現時点ではこのレポートはサポートされておらず、またこのレポートを含めるとパッケージのサイズが増大するためです。
7. オプションで、エージェント・アプリケーション・ファイルのセキュリティー署名を指定できます。セキュリティー署名を指定する場合は、「**自己記述型サポート JAR に署名**」を選択します。「**すべての JAR 署名設定を編集**」をクリックします。署名した JAR ファイルにタイム・スタンプを追加し、時刻認証局を指定できます。ご使用の Java 鍵ストア・ファイルに関する詳細を指定します。

   重要 : Java ツールを使用することにより Java 鍵ストア・ファイルを作成できます。例えば、秘密鍵と証明書を対応する公開鍵と共に Java 鍵ストア・ファイルで生成するには、以下のコマンドを実行します。

   - *ab_install_path*`/jre/bin/keytool -genkeypair -keystore `*keystore_file_path*` -storepass `*key_store_password*` -alias `*key_store_alias*` -dname "CN=`*common_name*`, OU=`*organizational_unit*`, L=`*city_or_locality*`, ST=`*state_or_province*`, C=`*country*`" -keypass `*key_password*

   各部の意味は次のとおりです。

   - *ab_install_path* は、Agent Builder がインストールされているロケーションです。
   - *keystore_file_path* は、既存の JKS 鍵ストアが存在するパス、または JKS 鍵ストアが作成されるパスです。
   - *key_store_password* は、この鍵ストア内のすべての項目にアクセスするために必要なパスワードです。
   - *key_store_alias* は、鍵ストア内でこの鍵を識別する名前です (デフォルトは `"mykey"`)。
   - *key_password* は、この特定の鍵にアクセスするために必要なパスワードです (デフォルトは `key_store_password`)。

   この証明書をサーバーの鍵ストアに含めます。
8. 「**完了**」をクリックします。

IBM Tivoli Monitoring 環境へのパッケージのインストール

IBM Tivoli Monitoring 環境でエージェントをテストまたは使用するには、生成されたパッケージを使用して、モニター対象のシステム、ハブ Monitoring Server システム、および Portal Server システムにエージェントをインストールしてください。

始める前に

モニター対象システムにエージェントをインストールする 前に、Tivoli Monitoring のオペレーティング・システム・エージェントが存在し、稼動していることを確認してください。Tivoli Monitoring エージェントの

インストールについては、Tivoli Monitoring Knowledge Center の『モニター・エージェントのインストール』を参照してください。

重要 : Tivoli Enterprise Portal にエージェント情報を表示するには、以下のコンポーネントをインストールする必要があります。

- エージェント (すべてのモニター対象システム上)
- Tivoli Enterprise Monitoring Server のサポート・ファイル (ハブ Tivoli Enterprise Monitoring Server 上)
- Tivoli Enterprise Portal Server のサポート・ファイル (Tivoli Enterprise Portal Server 上)
- Tivoli Enterprise Portal のサポート・ファイル (Tivoli Enterprise Portal Server と、該当する場合はすべての Tivoli Enterprise Portal デスクトップ・クライアント上)

手順

1. エージェントをインストールするシステムに圧縮ファイルをコピーします。圧縮ファイルのデフォルトの名前は、*product_code*.zip (Windows システムの場合) または *product_code*.tgz (UNIX システムおよび Linux システムの場合) です。
2. ファイルを一時的な場所に解凍します。

   注 : Linux AIX UNIX システムおよび Linux システムの場合、この一時的な場所を /tmp/*product_code* にしてはなりません (製品コードは小文字)。

   圧縮ファイルを使用して、リモート側でエージェントをインストールすることができます。

   - Linux Linux システムの場合は、以下のコマンドを使用して .tgz ファイルを解凍します。

     ```
     tar -xvzf filename
     ```

   - UNIX AIX システムの場合は、以下のコマンドを使用して .tgz ファイルを解凍します。

     ```
     gunzip filename
     tar -xvf filename
     ```

3. 該当のインストール・スクリプトを実行します。

   - エージェント、Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server、および Tivoli Enterprise Portal のサポートをすべて同時にインストールするには、以下を実行します。

     ```
     InstallIra.bat/.sh itm_install_location [[-h Hub_TEMS_hostname] -u
     HUB_TEMS_username -p Hub_TEMS_password]
     ```

   - サポート・ファイルをインストールせずにエージェントをインストールするには、以下のようにします。

     ```
     installIraAgent.bat/.sh itm_install_location
     ```

   - Tivoli Enterprise Monitoring Server サポートをインストールするには、以下の手順を実行します。

     ```
     installIraAgentTEMS.bat/.sh itm_install_location [[-h Hub_TEMS_hostname] -u
     HUB_TEMS_username -p Hub_TEMS_password]
     ```

   - Tivoli Enterprise Portal Server および Tivoli Enterprise Portal サポートをインストールするには、以下のようにします。

     ```
     installIraAgentTEPS.bat/.sh itm_install_location
     ```

   インストール・ロケーション *itm_install_location* は最初の引数として指定する必要があり、installIra.bat/.sh、installIraAgent.bat/.sh、installIraAgentTEMS.bat/.sh、および installIraAgentTEPS.bat/.sh のすべてのスクリプトで必須です。これが、このシステム上で Tivoli Monitoring コンポーネントがインストールされるロケーションです。

   他の引数はオプションです。

Monitoring Server のサポート・ファイルをインストールした場合、ユーザー ID を指定しないと、Tivoli Enterprise Monitoring Server がリサイクルされます。

4. エージェントを構成して開始します (1390 ページの『IBM Tivoli Monitoring 環境でのエージェントの構成と開始』を参照してください)。

次のタスク

エージェントのレイアウトを、結果としてナビゲーター項目が移動または削除されるように変更した場合は、Tivoli Enterprise Portal Server および Tivoli Enterprise Portal を再始動します。再始動により、変更内容が正しく認識されるようになります。

IBM Tivoli Monitoring 環境でのエージェントの構成と開始

IBM Tivoli Monitoring のモニター対象システムにエージェントをインストールした 後に、エージェントを構成して開始します。

手順

1. 「**Tivoli Monitoring Service の管理**」を開きます。
   新規項目 **Monitoring Agent for *agent_name*** が表示されます。
2. その項目を右クリックして、「**デフォルトを使用して構成**」を選択します。 プロンプトが出されたら、「**OK**」をクリックしてデフォルトを受け入れます。

   重要 :

   a. UNIX システムの場合、選択するオプションは「**構成**」です。

   b. マルチインスタンス・エージェントの場合、構成する際に、インスタンス名を要求するプロンプトが出されます。

   ヒント : エージェントが JMX データ・ソースを使用してリモートの WebSphere Application Server に接続している場合は、そのエージェントが稼働しているホストにも WebSphere Application Server をインストールし、Java home 設定をローカルの WebSphere Application Server が使用する Java ランタイム環境に指定してください。

   ヒント : Java API、JDBC、JMX、HTTP、または SOAP データ・ソースの場合、「**Java**」 > 「**JVM 引数**」設定を使用して、エージェント・トレース・ログを制御することができます。この設定で以下の値を設定します。

   ```
   -DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
   ```

   files は、保持するトレース・ログ・ファイルの最大数 (デフォルト値は 4) です。*size* は、ログ・ファイルの最大サイズ (キロバイト単位。デフォルト値は 5000) です。例えば、以下の値を設定することができます。

   ```
   -DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
   ```

   この場合、エージェントは最初のログ・ファイルに 100 キロバイト分のデータを書き込み、2 番目以降のログ・ファイルにも同じサイズのデータを書き込んでいくことになります。7 つのログ・ファイルすべてに 100 キロバイト分のデータが書き込まれると、次は最初のログ・ファイルに戻ってデータが上書きされます。

   エージェントにランタイム構成エレメントを追加した場合や、データ・ソースを選択した場合は、構成パネルが表示されます。このパネルは、エージェントに必要な情報を集めるために使用します。
3. エージェントの項目を右クリックして「**開始**」を選択します。
4. Tivoli Enterprise Portal を開いて、新規エージェントに移動します。

IBM Cloud Application Performance Management 環境でのエージェントのインストールと使用

IBM Cloud Application Performance Management 環境でエージェントをテストまたは使用するには、生成されたパッケージを使用して、モニター対象のすべてのシステムにエージェントをインストールしてください。場合によっては、エージェントを開始するために事前にエージェントを構成する必要があります。エージェントは、必要に応じて開始および停止できます。

エージェントのインストール
Agent Builder が準備したインストール・パッケージを使用して、すべてのモニター対象のシステムにエージェントをインストールします。

始める前に
IBM Cloud Application Performance Management のエージェント (通常は、オペレーティング・システム・エージェント) が既にモニター対象システムに存在し、稼働していることを確認します。

Windows Windows システムでは、管理者コマンド行シェルを使用してエージェントをインストールおよび構成します。管理者シェルを開始するには、Windows「プログラム」メニューから「**コマンドプロンプト**」を選択し、右クリックして「**管理者として実行**」をクリックします。

手順

1. パッケージを一時ディレクトリーに解凍し、そのディレクトリーに移動します。
2. エージェントのインストールには、ご使用のオペレーティング・システムに応じて、以下のコマンドを実行します。
   - Windows Windows システムの場合: `installIraAgent.bat` *`agent_install_location`*
   - Linux AIX Linux システムおよび UNIX システムの場合: `./installIraAgent.sh` *`agent_install_location`*

   ここで、*agent_install_location* は既存のエージェントのインストール・ロケーションです。デフォルトの場所は、以下のとおりです。
   - Windows Windows システムの場合は `C:¥IBM¥APM`
   - Linux Linux システムの場合は `/opt/ibm/apm/agent`
   - AIX AIX システムの場合は `/opt/ibm/apm/agent`

   重要 : Agent Editor の「**ランタイム構成**」ウィンドウで任意のカスタム構成プロパティーを追加している場合、またはエージェントが複数インスタンスをサポートする場合、またはユーザー ID やパスワードなどの構成が必要な事前定義されたデータ・ソースをエージェントが使用する場合は、エージェントを開始するには、事前にエージェントを構成しておく必要があります。エージェントの構成が不要である場合、エージェントはインストール後に自動的に開始します。

エージェントの構成
Agent Editor の「ランタイム構成」ウィンドウで任意のカスタム構成プロパティーを追加している場合、またはエージェントが複数インスタンスをサポートする場合、またはユーザー ID やパスワードなどの構成が必要な事前定義されたデータ・ソースをエージェントが使用する場合は、エージェントを開始するには、事前にエージェントを構成しておく必要があります。

始める前に

Windows Windows システムでは、管理者コマンド行シェルを使用してエージェントをインストールおよび構成します。管理者シェルを開始するには、Windows「プログラム」メニューから「**コマンドプロンプト**」を選択し、右クリックして「**管理者として実行**」をクリックします。

このタスクについて
この構成プロセスでは、以下を実行できます。

- インスタンス名を設定してインスタンスを作成するかインスタンスを変更します (エージェントが複数インスタンスをサポートする場合)。
- エージェントに対して使用できる任意の構成プロパティーを設定します。
- サブノードを作成して構成します (エージェントがサブノードをサポートする 場合)。

Windows Windows システムでは、構成プロパティーの設定またはサブノードの作成を行うには、サイレント構成手順を使用する必要があります。サイレント構成応答ファイルのサンプルは、*`install_dir`*

¥samples ディレクトリーに、*agentname*_silent_config.txt という名前で配置されています。このファイルのコピーを作成し、必要に応じて構成変数を設定します。

Linux AIX Linux システムおよび UNIX システムでは、オプションでサイレント構成手順を使用することができます。あるいは、対話式の手順を使用することもできます。応答ファイル名を指定せずに構成コマンドを開始した場合、構成ユーティリティーは、構成値の入力を求めるプロンプトを表示します。

手順

1. *install_dir*/bin ディレクトリーに移動します。
2. 以下のコマンドを実行して、エージェントを構成します。
   - エージェントが複数インスタンスをサポートしない場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat config [*response_file*]
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh config [*response_file*]
   - エージェントが複数インスタンスをサポートする場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat config *instance_name* [*response_file*]
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh config *instance_name* [*response_file*]

   各部の意味は次のとおりです。

   - *instance_name* は、インスタンスの名前です。 この名前のインスタンスが存在しない場合、そのインスタンスが作成されます。 そのインスタンスが既に存在する場合は、再構成されます。エージェントを使用するためにインスタンスを少なくとも 1 つは作成する必要があります。
   - *response_file* は、サイレント構成応答ファイルの名前です。

ヒント: エージェントが JMX データ・ソースを使用してリモートの WebSphere Application Server に接続している場合は、そのエージェントが稼働しているホストにも WebSphere Application Server をインストールし、Java home 設定をローカルの WebSphere Application Server が使用する Java ランタイム環境に指定してください。

ヒント: Java API、JDBC、JMX、HTTP、または SOAP データ・ソースの場合、「**Java**」 > 「**JVM 引数**」設定を使用して、エージェント・トレース・ログを制御することができます。この設定で以下の値を設定します。

```
-DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
```

files は、保持するトレース・ログ・ファイルの最大数 (デフォルト値は 4) です。*size* は、ログ・ファイルの最大サイズ (キロバイト単位。デフォルト値は 5000) です。例えば、以下の値を設定することができます。

```
-DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
```

この場合、エージェントは最初のログ・ファイルに 100 キロバイト分のデータを書き込み、2 番目以降のログ・ファイルにも同じサイズのデータを書き込んでいくことになります。7 つのログ・ファイルすべてに 100 キロバイト分のデータが書き込まれると、次は最初のログ・ファイルに戻ってデータが上書きされます。

エージェントの開始および停止

システムをモニターするには、システムでエージェントが確実に開始されているようにします。任意の時点でエージェントを開始および停止することができます。エージェントが複数インスタンスをサポートしている場合、すべてのインスタンスを個別に開始および停止することができます。

手順

1. *install_dir*/bin ディレクトリーに移動します。
2. 以下のコマンドを実行して、エージェントを開始します。
   - エージェントが複数インスタンスをサポートしない場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat start
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh start
   - エージェントが複数インスタンスをサポートする場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat start *instance_name*
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh start *instance_name*
3. 以下のコマンドを実行して、エージェントを停止します。
   - エージェントが複数インスタンスをサポートしない場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat stop
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh stop
   - エージェントが複数インスタンスをサポートする場合は、以下を実行します。
     - Windows Windows システムの場合: *name*-agent.bat stop *instance_name*
     - Linux AIX Linux システムおよび UNIX システムの場合: ./*name*-agent.sh stop *instance_name*

エージェントの生成およびインストール後の結果

Agent Builder エージェントをインストールすると、システムで特定のファイルが作成され、変更されます。IBM Tivoli Monitoring 環境では、Tivoli Enterprise Portal にも変更が表示されます。

システム上の新規ファイル

Agent Builder で作成したエージェントを生成およびインストールすると、エージェント・システム上に以下の新規ファイルが作成されます。

注 : *xx* は、2 文字の製品コードを示します。

Windows

Windows システム:

TMAITM6¥k*xx*agent.exe
エージェント・バイナリー

TMAITM6¥K*xx*ENV
環境変数の設定値

TMAITM6¥K*xx*.ref
エージェント・プロバイダーの構成

TMAITM6¥SQLLIB¥k*xx*.his
エージェントの属性情報の SQL 記述

TMAITM6¥SQLLIB¥k*xx*.atr
エージェントの属性情報

TMAITM6¥*xx*_dd_*version*.xmll
製品説明

TMAITM6¥*xx*_dd.properties
製品名

TMAITM6¥kxxcma.ini
エージェント・サービス定義ファイル

TMAITM6¥*your files*
Java API データ・ソースまたはソケット・データ・ソースからインクルードされた補足ファイル (ファイル・タイプは *executable* または *library*)。スクリプト・データ・ソースまたはコマンドの戻りコードのデータ・ソースからインクルードされたスクリプト。

Linux AIX

UNIX/Linux システム:
registry/*xx*architecture .ver
内部のバージョンおよび前提条件のファイル

***architecture*/*xx*/bin/*xx*_dd_*version*.xml**
製品説明

***architecture*/*xx*/bin/k*xx*agent**
エージェントのバイナリー

***architecture*/*xx* /bin/*xx*_dd.properties**
製品名

***architecture*/*xx*/work/k*xx*.ref**
エージェント・プロバイダーの構成

***architecture*/*xx* /tables/ATTRLIB/k*xx*.atr**
エージェントの属性情報

***architecture*/*xx* /hist/k*xx*.his**
エージェントの属性情報の SQL 記述

architecture*/*xx*/bin/*your files
Java API データ・ソースまたはソケット・データ・ソースからインクルードされた補足ファイル (ファイル・タイプは「実行可能ファイル」)。スクリプト・データ・ソースまたはコマンドの戻りコードのデータ・ソースからインクルードされたスクリプト。

architecture*/*xx*/lib/*your files
Java API データ・ソースまたはソケット・データ・ソースからインクルードされた補足ファイル (ファイル・タイプは「ライブラリー」)。

config/.*xx*.rc
内部セットアップ・ファイル

config/*xx*.environment
環境設定

config/*xx*_dd_*version*.xml
製品説明

config/*xx*_dd.properties
製品名

config/.ConfigData/k*xx*env
環境変数の設定値

注: システムのアーキテクチャーを 検出するには、以下のコマンドを実行してください。

```
cinfo -p xx
```

ここで、*xx* は、2 文字の製品コードを示します。

例えば、製品コードが 19 のエージェントを実行する Solaris 8 64 ビットのシステムの場合、出力は以下のようになります。

```
# /opt/ibm/apm/agent/bin/cinfo -p 19

*********** Fri Aug 17 11:23:58 EDT 2007 ******************
User : root Group: other
Host name : guadalajara  Installer Lvl:06.20.00.00
CandleHome: /opt/IBM/ITM
***********************************************************
Platform codes:
sol286 : Current machine
sol286 : Product (19)
tmaitm6/sol286 : CT Framework (ax)
```

太字の行が、関係する行です。コロンの前のストリング sol286 は、このエージェントに使用されているアーキテクチャーを示します。このストリングは、オペレーティング・システムとコンピューターのハードウェア・タイプのさまざまな組み合わせによって、異なるものになります。この機能を動作させるには、事前にエージェントをインストールしておく必要があります。

Java ベースのデータ・ソースに関するファイルを以下に示します。これらのファイルは、エージェントに JMX、JDBC、HTTP、または SOAP のデータ・ソースが含まれている場合にのみ作成されます。

- cpci.jar
- jlog.jar
- common/jatlib-1.0.jar

JMX ランタイム・サポートに関するファイルを以下に示します。 これらのファイルは、エージェントに JMX データ・ソースが含まれている場合にのみ作成されます。

- common/jmx-1.0.jar
- common/connectors/jboss/connJboss-1.0.jar
- common/connectors/jsr160/connJSR160-1.0.jar
- common/connectors/was/connWas-1.0.jar
- common/connectors/weblogic/connWeblogic-1.0.jar

JDBC ランタイム・サポートに関するファイルを以下に示します。これらのファイルは、エージェントに JDBC データ・ソースが含まれている場合にのみ作成されます。

- common/jdbc-1.0.jar

HTTP または SOAP のランタイム・サポートに関するファイルを以下に示します。これらのファイルは、エージェントに HTTP または SOAP のデータ・ソースが含まれている場合にのみ作成されます。

- http-1.0.jar

Java API ランタイム・サポートに関するファイルを以下に示します。これらのファイルは、エージェントに Java API データ・ソースが含まれている場合にのみ作成されます。

- cpci.jar
- custom/*your JAR file* この JAR ファイルの名前は、Java API データ・ソースの「**グローバル設定**」に指定されています。
- custom/*your JAR file* 補足ファイル (ファイル・タイプは「Java リソース」)。

Java ベースのデータ・ソースについては、Windows、UNIX、および Linux の各システム上に同じファイルが存在しますが、ディレクトリーは以下のように異なります。

- Windows Windows のパス: TMAITM6¥kxx¥jars
- Linux AIX UNIX/Linux のパス: *architecture*/*xx*/jars

ログ・ファイル・モニターのランタイム・サポートに関するファイルを以下に示します。これらのファイルは、エージェントにログ・ファイル・データ・ソースが含まれている場合にのみ作成されます。

- Windows システムの場合: TMAITM6¥kxxudp.dll
- Linux Solaris/Linux システムの場合: *architecture*/*xx*/lib/libkxxudp.so
- HP-UX システムの場合: *architecture*/*xx*/lib/libkxxudp.sl
- UNIX AIX システムの場合: *architecture*/*xx*/lib/libkxxudp.a

SSH スクリプト・モニターのランタイム・サポートに関するファイルを以下に示します。これらのファイルは、SSH 収集に使用可能なスクリプト・データ・ソースがエージェントに含まれている場合にのみ作成されます。

- Windows システムの場合: TMAITM6¥kxxssh.dll
- Linux Solaris/Linux システムの場合: *architecture*/*xx*/lib/libkxxssh.so
- HP-UX システムの場合: *architecture*/*xx*/lib/libkxxssh.sl
- UNIX AIX システムの場合: *architecture*/*xx*/lib/libkxxssh.a

「Tivoli Enterprise Monitoring Services の管理」ウィンドウの変更点

IBM Tivoli Monitoring 環境にエージェントをインストールすると、「**Tivoli Enterprise Monitoring Services の管理**」ウィンドウにエージェントの項目が表示されます。この項目の名前は「**Monitoring Agent for *agent_name***」です。

重要 : 「**Tivoli Enterprise Monitoring Services の管理**」は、IBM Cloud Application Performance Management 環境ではサポートされていません。

Windows Windows システムの場合、この項目には「**タスク/サブシステム**」列が 表示され、エージェントが複数のインスタンスをサポートするかどうかが 以下のように示されます。

- 単一インスタンス・エージェントの場合は、「**Tivoli Enterprise Monitoring Services の管理**」ウィンドウに新規アプリケーションが表示されます。 このアプリケーションの名前は「**Monitoring Agent for *agent_name***」です。このエージェントに対する サービスが作成されます (1397 ページの図 70)。「**タスク/サブシステム**」列には 値「**プライマリー**」が表示されます。
- 複数インスタンス・エージェントの場合は、「**Tivoli Enterprise Monitoring Services の管理**」ウィンドウに新規アプリケーション・テンプレートが表示されます。このテンプレートの名前は「**Monitoring Agent for *agent_name***」です。このテンプレートから エージェントのインスタンスを作成するまで、エージェントに対するサービスは 作成されません。「**タスク/サブシステム**」列には値「**テンプレート**」が表示され、この項目が、エージェントのインスタンスを 作成するために使用するテンプレートであることが示されます。

Linux AIX Linux および UNIX システムの場合、エージェントが複数のインスタンスをサポートするかどうかにかかわらず、エージェントの項目は同じです。

注 : 以下の画面は、Windows システムの場合の画面です。UNIX システムおよび Linux システムの場合も類似した画面になります。

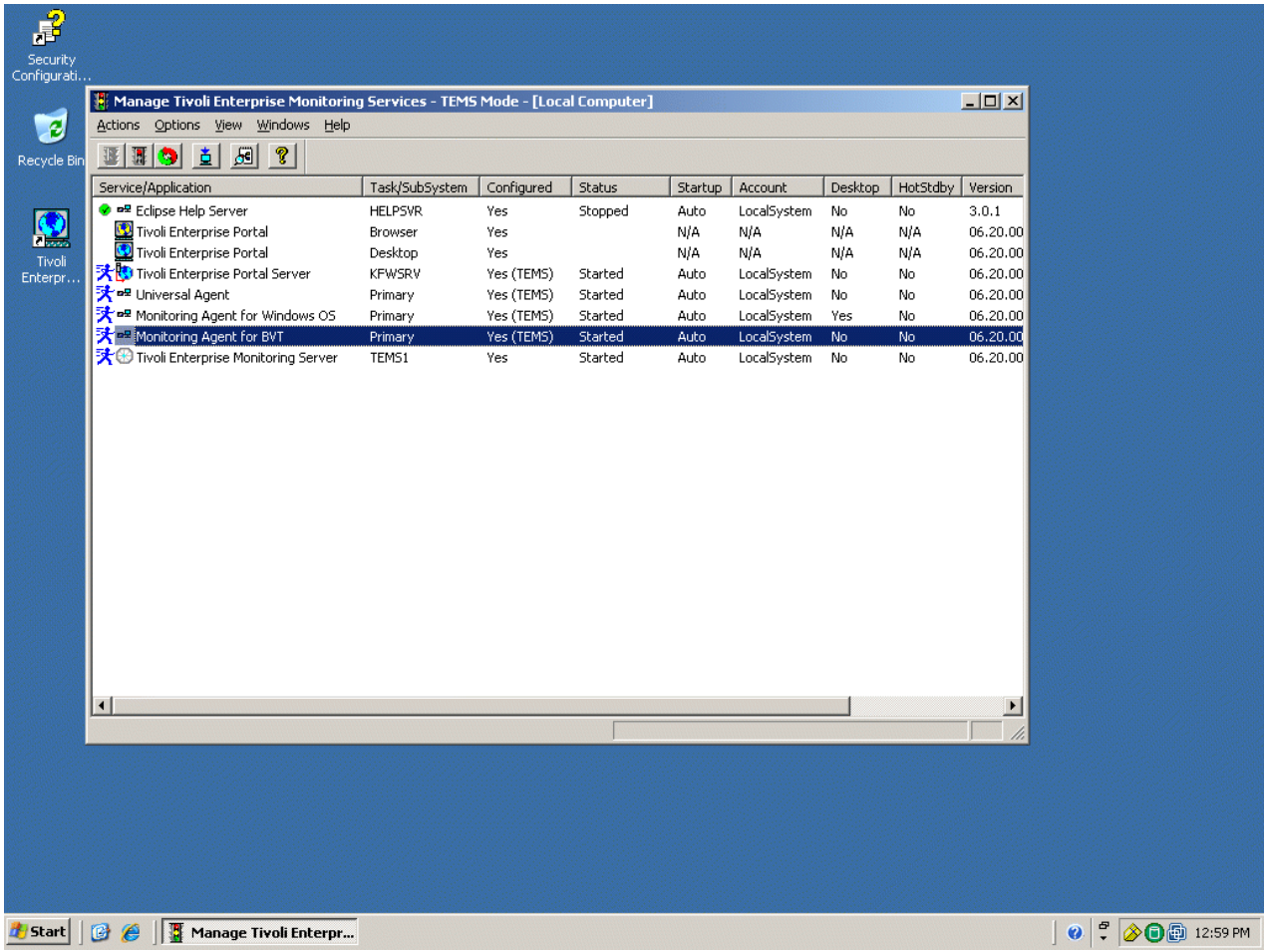


図 *70. Tivoli Enterprise Monitoring Services* ウィンドウの管理

Tivoli Enterprise Portal の変更点

IBM Tivoli Monitoring 環境では、エージェントをインストールして開始した後に、Tivoli Enterprise Portal で緑色の「**最新表示**」アイコンをクリックします。これにより、新規エージェントを表示できます。ポータルに以下の変更点が発生します。

- Tivoli Enterprise Portal の物理ビュー内のエージェント用新規サブノード。
- Agent Builder を使用して定義したすべてのナビゲーター・グループおよびデータ・ソース用のノード (1398 ページの図 71)。

  注 : 各ナビゲーター項目ごとに、デフォルトの照会を定義しなければなりません。

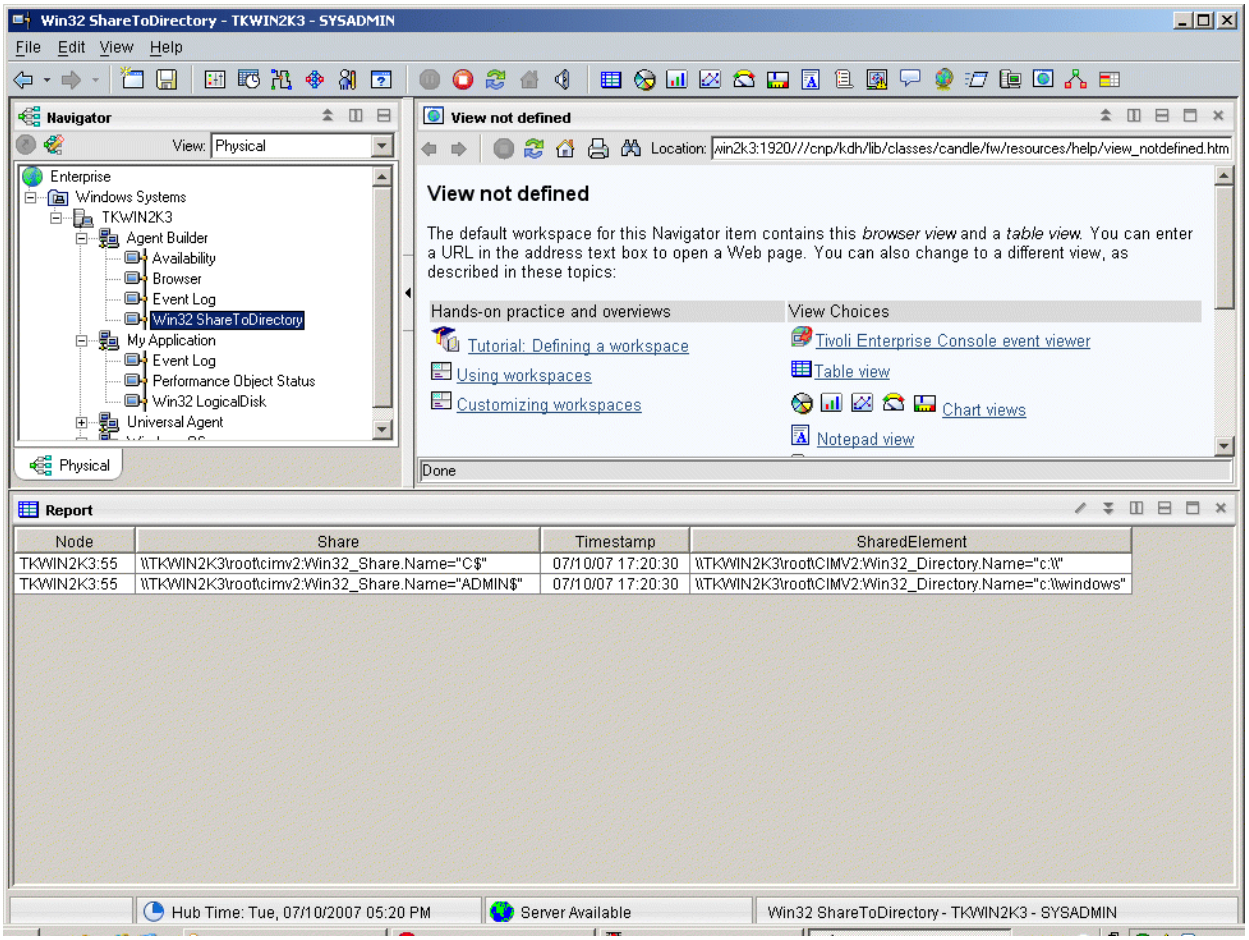


図 71. 新規エージェントの属性グループのノード。

- エージェントにサブノードが含まれる場合は、エージェント内で定義された各サブノードに対して 1 つの拡張可能ノードが存在します。拡張可能ノードの下には、以下のノードが表示されます。
  - xxx のパフォーマンス・オブジェクト状況 (xxx は 3 文字のサブノード・タイプ)
  - サブノード内で定義したすべてのナビゲーター・グループおよびデータ・ソース用のノード
  - xxx の「イベント・ログ」ノード (イベント・ログがある場合)
  - xxx の「JMX モニター」ノード (JMX があり、JMX モニターを含めた場合)
- 以下の自動ノード。
  - 「可用性」ノード (エージェントに可用性データ・ソースが含まれる場合) (1399 ページの図 72)

    注 : このノードの振る舞いは、エージェントの内容によって異なります。エージェントが可用性のみをモニターする場合、「可用性」ノードは可用性データ・ソースを表します。エージェントが可用性およびパフォーマンスをモニターする場合、「可用性」ノードは、可用性データ・ソースおよびパフォーマンス・オブジェクト状況のデータ・ソースを表すナビゲーター項目になります。

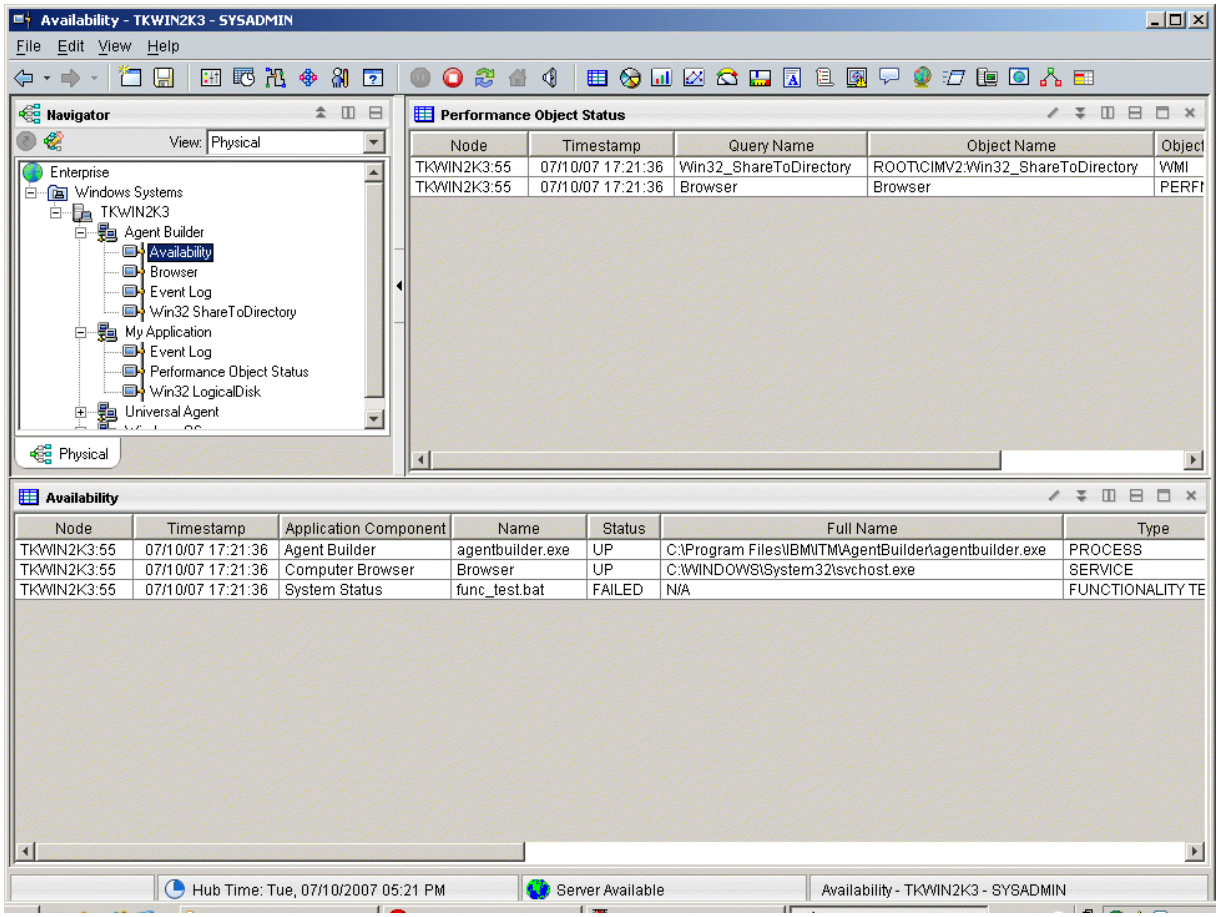


図 72. 可用性ノード

– パフォーマンス・オブジェクト状況 (エージェントに (可用性ではなく) パフォーマンス・モニターのデータ・ソースが含まれる場合) (1400 ページの図 73)

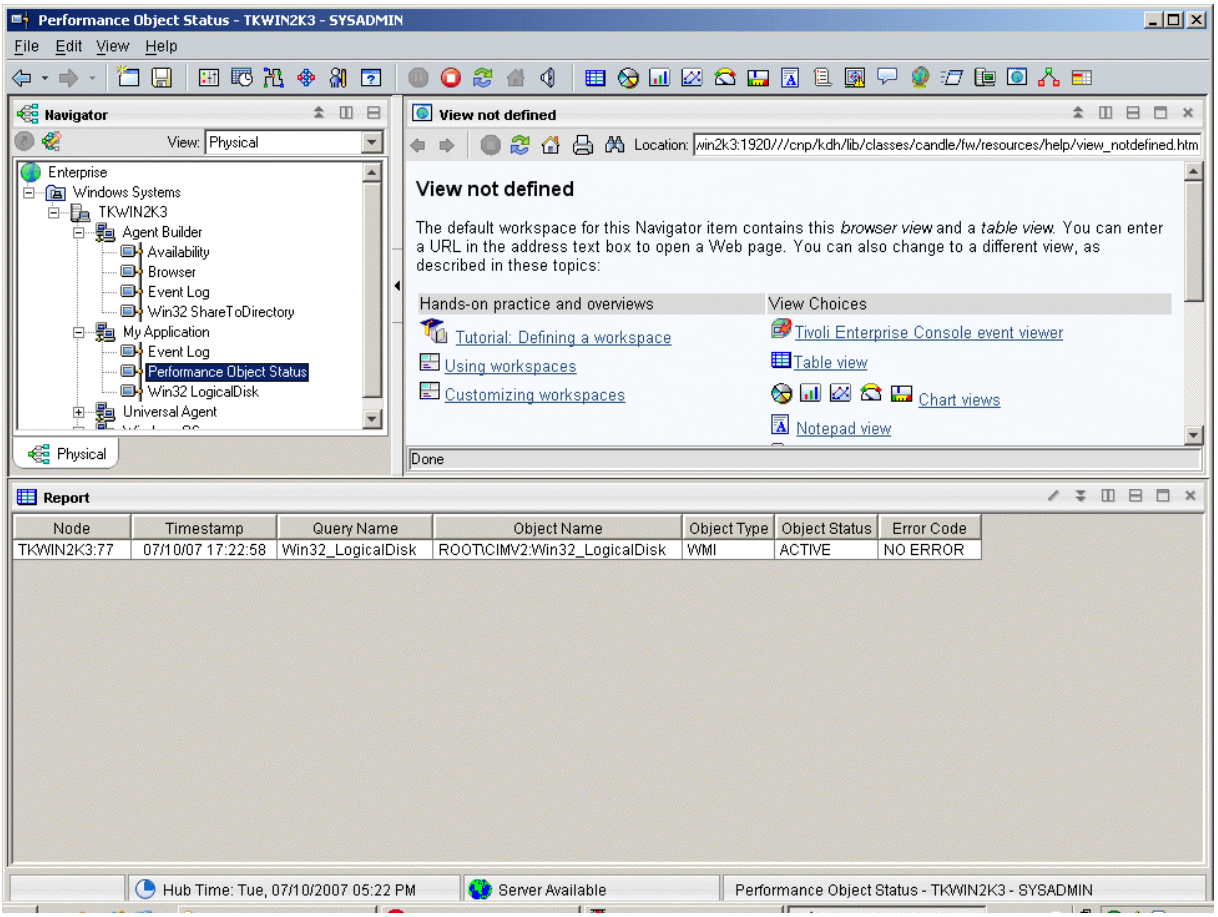


図 73. 「パフォーマンス・オブジェクトの状況」ノード

– イベント・ログ (エージェントにデータ・ソース生成のログ・データが含まれる場合) (1401 ページの図 74)

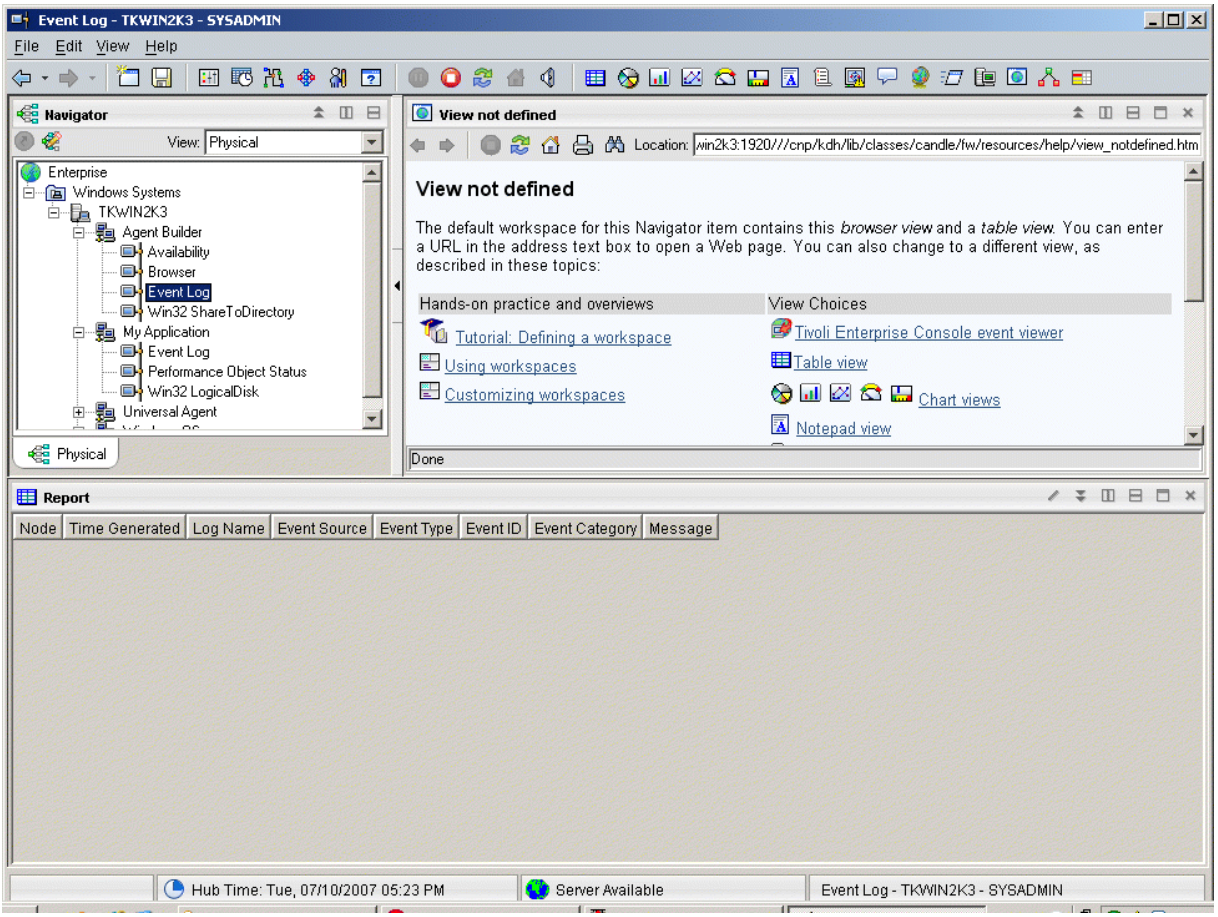


図 74. イベント・ログ・ノード

Agent Builder の属性グループおよび属性の説明については、1417 ページの『属性のリファレンス』を参照してください。

エージェントのアンインストール

Agent Builder が生成したエージェントをモニター対象ホストから削除できます。

このタスクについて

アンインストール・プロセスは、エージェント・システムからエージェントを アンインストールするだけです。このプロセスによって他のエージェントやモニター・インフラストラクチャーがアンインストールされることはありません。

IBM Tivoli Monitoring 環境で Agent Builder が生成したエージェントを削除するには、以下のいずれかの手順を使用します。

- 1402 ページの『Tivoli Enterprise Portal を使用した Tivoli Monitoring エージェントの削除』
- 1402 ページの『Tivoli Enterprise Portal を使用しない Tivoli Monitoring エージェントの削除』

これらのいずれかの手順でエージェントを削除した後に、1402 ページの『Tivoli Enterprise Portal からの Tivoli Monitoring エージェントの消去』のステップを使用して Tivoli Enterprise Portal からエージェントを消去します。

IBM Cloud Application Performance Management 環境では、1403 ページの『IBM Cloud Application Performance Management エージェントのアンインストール』のステップを使用します。

Tivoli Enterprise Portal を使用した Tivoli Monitoring エージェントの削除

IBM Tivoli Monitoring 環境では、Tivoli Enterprise Portal を使用してエージェントを削除できます。

始める前に

作成済みのエージェントを削除するには、オペレーティング・システム・エージェントが実行中であることが必要です。

手順

Tivoli Enterprise Portal を使用してエージェントを削除するには、以下の手順を実行します。

- Tivoli Enterprise Portal のナビゲーション・ツリーでエージェントを右クリックし、「**削除**」を選択します。

Tivoli Enterprise Portal を使用しない Tivoli Monitoring エージェントの削除

IBM Tivoli Monitoring 環境で Tivoli Enterprise Portal を使用できない場合は、オペレーティング・システムのスクリプトおよびコマンドを使用してエージェントを削除できます。

手順

Agent Builder によって生成されたエージェントを、Tivoli Enterprise Portal を使用せずにターゲット・システムから削除するには、以下のいずれかの手順を実行します。

- Windows

  Windows システムの場合は、以下のコマンドを使用します。

  ```
  cd ITM_INSTALL/TMAITM6
  kxx_uninstall.vbs ITM_INSTALL
  ```

  ここで、*xx* はエージェントの製品コードです
- Windows

  代わりに、Windows システムでは、`cscript.exe` コマンドを使用してアンインストール・スクリプトを実行することもできます。このコマンドは vbs スクリプト用のコマンド行インターフェース・パーサーであり、ウィンドウは表示されず、代わりにコンソールにメッセージが表示されます。

  ```
  cd ITM_INSTALL/TMAITM6
  cscript.exe kxx_uninstall.vbs ITM_INSTALL
  ```
- Linux AIX

  Linux システムまたは UNIX システムの場合は、*ITM_INSTALL*`/bin` にある `uninstall.sh` ファイルを使用します。

  ```
  uninstall.sh [-f] [-i] [-h ITM_INSTALL] [product platformCode]
  ```

Tivoli Enterprise Portal からの Tivoli Monitoring エージェントの消去

IBM Tivoli Monitoring 環境では、エージェントを削除した後も、エージェントからの情報に対する空のフィールドが Tivoli Enterprise Portal に残っている場合があります。これらのフィールドを削除するには、Tivoli Enterprise Portal からエージェントを消去します。

手順

1. Tivoli Enterprise Monitoring Server および Tivoli Enterprise Portal Server が稼働中であることを確認します。
2. Tivoli Enterprise Portal クライアントにログオンします。
3. Tivoli Enterprise Portal クライアントの「物理」ナビゲーター・ビューで「**エンタープライズ**」を右クリックし、「**ワークスペース**」 > 「**管理対象システム状況**」と選択します。

   「管理対象システム状況」ワークスペースが表示されます。
4. エージェントのすべての IBM Tivoli 管理対象システムを選択します。

5. 右クリックして「**オフライン項目のクリア**」を選択します。すべての項目がその表から消去されます。

IBM Cloud Application Performance Management エージェントのアンインストール
IBM Cloud Application Performance Management 環境の任意のモニター対象システムからエージェントをアンインストールすることができます。

手順

1. エージェントがインストールされているシステムで、コマンド行を開始し、*install_dir*/bin ディレクトリーに移動します。ここで、*install_dir* はモニター・エージェントのインストール・ディレクトリーです。
2. 特定のモニター・エージェントをアンインストールするには、エージェント・スクリプト名とアンインストール・オプションを入力します。*name* はエージェント・スクリプト名です。
   - Windows システムの場合: *name*`-agent.bat uninstall`
   - Linux システムまたは AIX システムの場合: `./`*name*`-agent.sh uninstall`

アプリケーション・サポート・ファイルのインポート

エージェントを IBM Tivoli Monitoring 環境で使用する場合、カスタム・シチュエーション、ワークスペース、アクション実行コマンド、および照会をインストール・パッケージに組み込むことができます。

このタスクについて

シチュエーション、ワークスペース、およびエージェントを単一のインストール・イメージに含めるには、シチュエーションとワークスペースのファイルが、エージェントと同じプロジェクト内にある必要があります。Agent Builder には、エージェント・プロジェクト内に該当のファイルを作成するためのウィザードが用意されています。

インストール・パッケージには、エージェントに関連する定義も組み込むことができます。これらの定義の内容は、エンタープライズ・モニター環境およびシステム・モニター環境で使用されているエージェントにより異なります。エンタープライズ・モニター・エージェント・イメージには、カスタムのシチュエーション、ワークスペース、アクション実行コマンドおよび照会を組み込むことができます。システム・モニター・エージェント・イメージには、専用のシチュエーション、トラップ定義、およびエージェント構成情報を組み込むことができます。

適切な定義およびエージェント自体を単一のインストール・パッケージに含めるには、ファイルが、エージェントと同じプロジェクト内になければなりません。Agent Builder には、エンタープライズ・モニターのインストール済み環境用の適切なファイルを作成するためのウィザードが用意されています。システム・モニター・エージェント環境用のファイルは、「*IBM Tivoli Monitoring* 管理者ガイド」の『エージェントのオートノミー』という章で説明しているプロセスを使用して作成します。その結果として作成されたファイルは、エージェントの Eclipse プロジェクトのルートにコピーされます。

Tivoli Enterprise Monitoring Agents のファイルのエクスポートおよびインポート

このタスクについて

Tivoli Enterprise Portal でシチュエーション、ワークスペース、照会、およびアクション実行コマンドを作成したら、それらをエクスポートして、別の Tivoli Monitoring バージョン 6.2 環境にインポートすることができます。シチュエーションおよびワークスペースの 作成方法について詳しくは、1367 ページの『ワークスペース、アクション実行コマンド、およびシチュエーションの作成』を参照してください。以下のステップを実行して、シチュエーション、ワークスペース、アクション実行コマンド、および照会を抽出します。

手順

1. 「**プロジェクト・エクスプローラー**」タブで、エージェントのプロジェクト・フォルダーを右クリックします。

2.「**IBM Corporation**」 >「**アプリケーション・サポート・ファイルのインポート**」を選択します。
3. Tivoli Enterprise Portal Server のホスト名を入力します。
4. 接続する Tivoli Monitoring 環境のユーザー名およびパスワードを 入力し、「**完了**」をクリックします。
5. エージェントのシチュエーションを定義した場合には、そのエージェント用に定義されているシチュエーションをリストしたダイアログ・ボックスが表示されます。
6. エクスポートするシチュエーションをリストから選択して「**<<**」をクリックすると、それらのシチュエーションが、選択したシチュエーションの表に追加されます。「**OK**」をクリックしてください。

   インポートには数分掛かる場合があります。 処理が完了したら、エージェント・プロジェクトの該当フォルダー内に SQL ファイルがあるのを確認できます。
7. エージェントのアクション実行コマンドを定義した場合には、定義されたアクション実行コマンドがダイアログに表示されます。 エクスポートするアクション実行コマンドをリストから選択して「**>>**」をクリックし、それらのアクション実行コマンドを選択済みアクション実行の表に追加して、「**OK**」をクリックしてください。

   インポートには数分掛かる場合があります。 処理が完了したら、エージェント・プロジェクトの該当フォルダー内に SQL ファイルがあるのを確認できます。
8. エージェントのカスタム照会を定義した場合には、定義された照会がダイアログに表示されます。エクスポートする照会をリストから選択して「**<<**」をクリックすると、それらの照会が、選択した照会の表に追加されます。「**OK**」をクリックしてください。

   インポートには数分掛かる場合があります。 処理が完了したら、エージェント・プロジェクトの該当フォルダー内に SQL ファイルがあるのを確認できます。ワークスペースは自動的にインポートされます。

次のタスク
カスタム・エージェントを再作成し、モニター対象ホストにエージェントをインストールして、Tivoli Enterprise Portal サポートをインストールします。

Tivoli System Monitor Agents のファイルのエクスポートおよびインポート

このタスクについて
システム・モニター・エージェント定義は、次の 3 つのタイプのファイルに含まれています。

- 専用シチュエーションは *xx*_situations.xml というファイル内で定義されています。ここで、*xx* は 2 文字の製品コードです。
- トラップ構成情報は *xx*_trapcnfg.xml というファイル内で定義されています。ここで、*xx* は 2 文字の製品コードです。
- 構成が必要なエージェントの場合、その構成はエージェントのインスタンスごとに 1 つのファイルに定義されます。エージェントが 単一インスタンス・エージェントである場合のファイル名は *xx*.cfg です。エージェントが複数インスタンスのエージェントである場合、インスタンスごとに 1 つのファイルがあります。ファイル名は *xx_instance name*.cfg です。 ここで、*xx* は 2 文字の製品コードであり、*instance name* はエージェント・インスタンスの 名前です。

手順

- 「*IBM Tivoli Monitoring* 管理者ガイド」の『エージェントのオートノミー』という章で説明しているプロセスを使用して、ファイルを作成します。このファイルを手動でプロジェクト・ディレクトリーのルートにコピーするか、Eclipse のインポート機能を使用してインポートするファイルを選択します (「**ファイル**」 >「**インポート**」 >「**一般**」 >「**ファイル・システム**」)。

  これらのファイルはエージェント・イメージに組み込まれ、インストーラーによってインストールされます。

  エージェントのインストール時には、以下の処理が行われます。

  – 組み込みファイルを適切なロケーションにコピーします。
  – pc_situations.xml ファイルで定義されている専用シチュエーションが、エージェントで実行されます。

– pc_trapcnfg.xml で定義されているトラップ定義が、シチュエーションに基づいたトラップの転送に使用されます。
– 以下の場合は、エージェントが自動的に構成されて開始されます。
  - エージェントが単一インスタンスのエージェントで、エージェントの一部として構成が定義されていない場合。
  - エージェントが単一インスタンスのエージェントで、エージェントの一部として構成が定義されていて、イメージに pc.cfg ファイルが含まれている場合。
  - エージェントが複数インスタンスのエージェントである (すべての複数インスタンス・エージェントで構成が必要)。インストーラーは、pc_*inst*.cfg ファイルごとにエージェントのインスタンスを 1 つ開始します。

イベント・フィルタリングおよび要約

属性グループは、ピュア・イベント またはサンプル として定義されます。ピュア・イベント属性グループには、非同期に発生するデータ行が含まれています。到着した各新規データ行は即時に Tivoli Monitoring により処理されます。サンプル属性グループは、データが要求されるたびに、現行のデータ行セットを収集します。以下の属性グループの例で、この相違を説明します。

- エージェントに送信された通知と SNMP トラップをすべて表す「SNMPEvent」属性グループが作成されます。モニター対象システムにより送信されたトラップまたは通知は、非同期に到着します。 到着した各イベントは、Tivoli Monitoring に渡されます。
- システムのすべてのディスクに 関する情報を表す「ディスク」属性グループが作成されます。ディスク情報は定期的に収集されます。 ディスク情報が収集されるたびに、エージェントはデータ行数 (ディスクごとに 1 件) を返します。

ピュア・イベント属性グループとサンプル 属性グループの違いは、Tivoli Monitoring のさまざまな面に影響します。このような面には、シチュエーション、ウェアハウス・データ、および Tivoli Enterprise Portal のビューなどがあります。

各シチュエーションは、一連の条件のうち特定の条件についてモニターする 1 つ以上の管理対象システムに割り当てます (配布 します)。 特定の間隔で実行する監視に基づいてイベントの決定を行う必要がある場合、そのイベントはサンプル・イベント と見なされます。イベントが自発的な発生に基づく場合、このイベントはピュア・イベントと見なされます。 したがって、サンプル・イベントのシチュエーションには間隔が関連付けられていますが、ピュア・イベントのシチュエーションには間隔は関連付けられていません。サンプル・イベントのもう 1 つの特性として、イベントの発生条件が変わり、条件が true にならなくなる場合があるということです。ピュア・イベントは変更できません。そのため、サンプル・イベントについて出されるアラートは、true から false に変更されることがありますが、ピュア・イベントは true のままになります。

サンプル・イベントの例として、number of processes > 100 を考えてみます。 プロセス数が 100 を超えるとイベントが True になり、その後、この数が 100 以下になると再び False になります。invalid logon attempt by user についてモニターするシチュエーションは、ピュア・イベントです。無効なログオン試行が検出されると、このイベントが発生しますが、false イベントにはなりません。特定の間隔で評価されるシチュエーションをサンプル属性グループに対して作成できますが、このような評価をピュア・イベント属性グループに対して行うことはできません。

同様に、ヒストリカル・データについて、サンプル・データを収集する頻度を構成することができます。ただし、ピュア・イベント・データの収集をオンにした場合、その発生ごとに各行を取得することになります。

サンプル・データとして Tivoli Enterprise Portal に表示されるデータは、収集された行の最新セットです。ピュア・イベント属性グループに関して表示されるデータは、エージェントが管理するローカル・キャッシュの内容です。これは、シチュエーション評価とヒストリカル収集のために Tivoli Monitoring に渡されたデータと必ずしも一致しているとは限りません。

重複イベントの制御

イベント・フィルタリングおよび要約オプションを使用して、重複イベントを Tivoli Monitoring に送信する方法を制御します。

始める前に

イベント・フィルタリングおよび要約について詳しくは、1405 ページの『イベント・フィルタリングおよび要約』を参照してください。

このタスクについて

Agent Builder は、イベント・データを Tivoli Monitoring 内でピュア・イベントとして表す属性グループを定義します。これらの属性グループには、ログ・ファイル、AIX バイナリー・ログ、SNMP イベント、および JMX 通知が含まれています。これらの属性グループでは、複数の重複イベントを生成できます。 これらの重複イベントを Tivoli Monitoring に送信する方法を制御できます。「**拡張**」ウィンドウの「**詳細なデータ・ソース・プロパティー**」の「**イベント情報**」タブで、ログ・ファイル、SNMP イベント、および JMX 通知の各属性グループについてこれらの制御をアクティブにできます。

イベントが、他のイベントの重複イベントとして処理されるかどうかは、属性グループで定義するキー属性によって決定されます。 イベント内のすべてのキー属性の値が、既存のイベント内の同じキー属性の値と一致する場合、重複イベントが発生します。イベント・フィルタリングおよび要約が有効な場合、`isSummary`、`occurrenceCount`、`summaryInterval`、および `eventThreshold` 関数の属性が自動的に追加されます。

手順

- 「**イベント・フィルタリングおよび要約オプション**」領域で、以下のいずれかのオプションを選択します。
  - **イベント・フィルタリングまたは要約なし**: イベント・フィルタリングまたは要約なしで、すべてのイベントを送信します。 このオプションはデフォルト・オプションです。
  - **イベントのフィルター処理および要約**: 重複する各イベント、およびキー属性に基づく固有の各イベントの要約レコードを作成します。イベント・フィルタリング・オプションも選択します。「**要約オプション**」領域で、要約間隔を入力します。 値を秒単位で入力するか、構成プロパティーを挿入します。

    イベント・フィルタリング・オプションは以下のとおりです。

    - **要約イベントのみの送信**: 指定された間隔の要約レコードのみ送信します。
    - **すべてのイベントの送信**: すべてのイベントおよび要約レコードを送信します。
    - **最初のイベントの送信**: イベントごとに、指定されている要約間隔で受信した最初のイベントのみが送信され、重複するイベントは送信されません。このオプションでは、要約レコードも送信されます。
    - **イベントしきい値**: 間隔内で受信した重複イベントの数が、しきい値で割り切れる数になると、イベントが Tivoli Monitoring に送信されます。例えば、イベントしきい値を 5 に設定し、特定の間隔で受信した重複イベントの数が (最初のイベントを含めて) 5 個未満であった場合、イベントは Tivoli Monitoring に送信されません。5、6、7、8、または 9 個の重複イベントを受信した場合、1 個のイベントが送信されます。10 個の重複イベントを受信した場合は、2 個のイベントが送信されます。「**イベントしきい値**」フィールドでは、数値を入力するか、構成プロパティーを挿入します。 このオプションでは、要約レコードも送信されます。

Tivoli Enterprise Portal でのイベント・フィルタリングおよび要約の表示

イベント・フィルタリングおよび要約のオプション選択により、データの処理方法がどのようになるかの例を示します。

エージェントは、最近受信されたイベントのキャッシュを保守します。デフォルトでは、このキャッシュのサイズは 100 です。エージェントのイベント・フィルタリングおよび要約を有効にした場合は、キャッシュ内のイベントの数と IBM Tivoli Monitoring に送信されるイベントの数が異なってくる場合があります。 キャッシュに追加されたイベントが、指定された送信しきい値に到達しない可能性があります。ある

いは、「**すべてのイベントの送信**」オプションを選択した場合、キャッシュ内のイベントの方が少なくなる可能性があります。「**すべてのイベントの送信**」オプションが設定されている場合、イベントは重複が発生するたびに送信されます。しかし、キャッシュ内に保持されるイベントのコピーは 1 つのみであり、発生回数はイベントが発生するたびに増加します。IBM Tivoli Monitoring に送信されるイベントを表示するには、ヒストリカル・ビューを作成します。ヒストリカル・ビューの作成については、「Tivoli Enterprise Portal ユーザーズ・ガイド」の『ヒストリカル・レポート』を参照してください。このビューを Tivoli Enterprise Portal 内のリアルタイム・キャッシュ・ビューと比較できます。シチュエーションを使用して同じ比較を実行することもできます。

以下の例は、イベント・フィルタリングおよび要約についてユーザーが選択した内容に応じて (該当する場合)、同じログ・データがどのように処理されるかを示しています。 この例のエージェントは、異なる動作を示すように作成されています。各属性グループは、同じログ・ファイルをモニターするように定義されています。 それぞれの例で、ヒストリカル・ビューとリアルタイム (キャッシュ) ビューが表示されます。Tivoli Enterprise Portal におけるノードの名前は、選択した設定内容を反映しています。デフォルトでは、ヒストリカル・ビューには最新のイベントが最後に表示されます。キャッシュにあるデフォルトのリアルタイム・ビューには、最新のイベントが最初に表示されます。これらの例では、ヒストリカル・ビューには直近の 1 時間が示されています。

新規イベントを受信すると、キャッシュ・ビューに表示されます。重複するイベントを受信した場合、既存の行のデータが更新されます。 要約間隔が経過すると、既存のイベントは要約イベントに変換され、送信されます。 その後、次の要約間隔で、新規の行が追加されます。

1408 ページの図 75 には、イベント・フィルタリングや要約を有効にしなかった場合のヒストリカル・ビューとキャッシュ・ビューが表示されています。いずれのビューも同じデータを表示しますが、順序が逆になっています。対応するイベントを表示するために、ヒストリカル・ビューはスクロールダウンされ、リアルタイム (キャッシュ) ビューはスクロールアップされます。

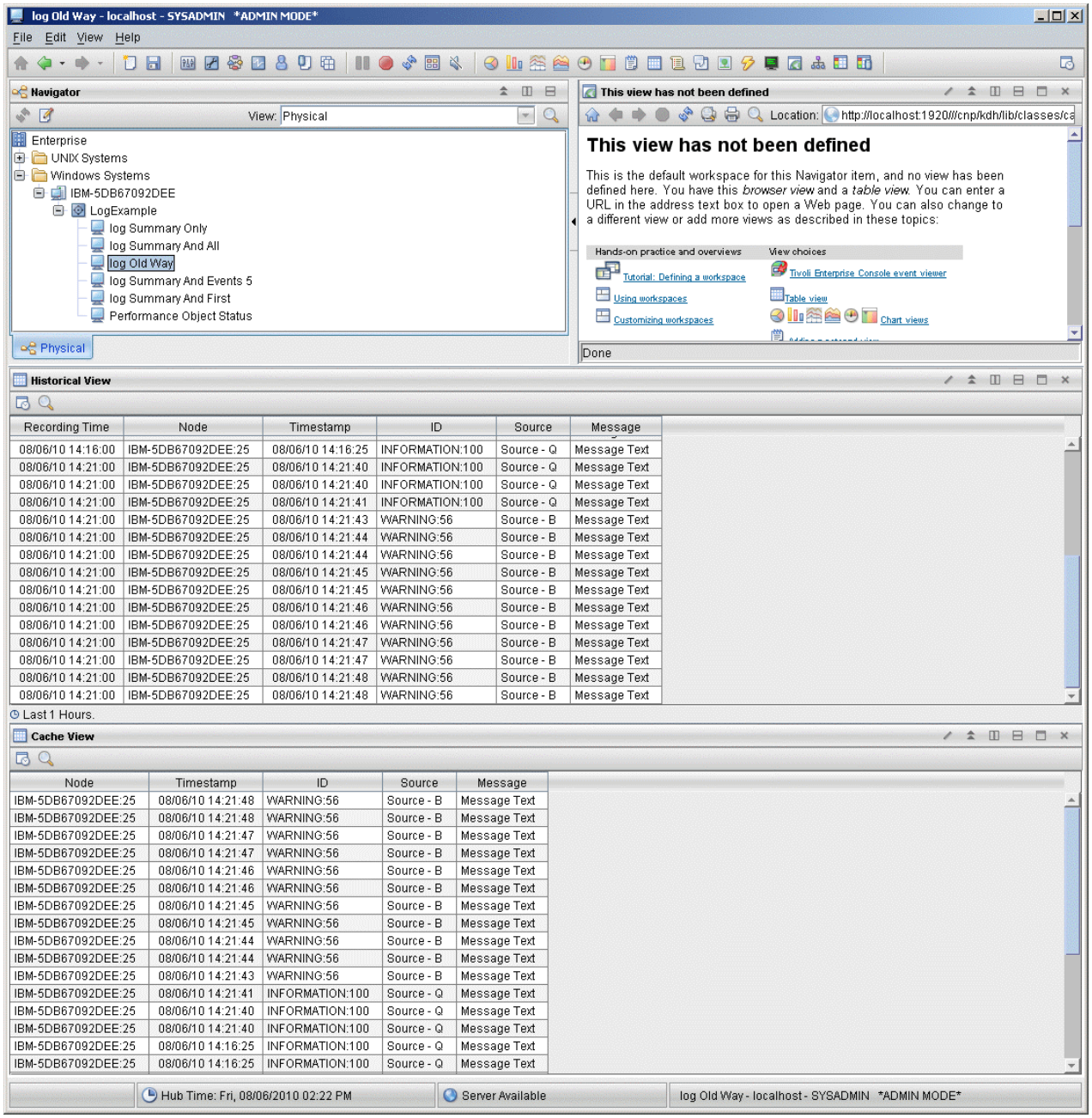


図 75. イベント・フィルタリングまたは要約が有効でない場合のヒストリカル・ビューおよびキャッシュ・ビュー

1409 ページの図 76 には、「**イベント情報**」タブで、「**要約イベントのみの送信**」オプションを選択した場合のヒストリカル・ビューとキャッシュ・ビューが表示されています。要約イベントは両方のビューに表示されますが、新規イベントはリアルタイム (キャッシュ) ビューにのみ表示されます。

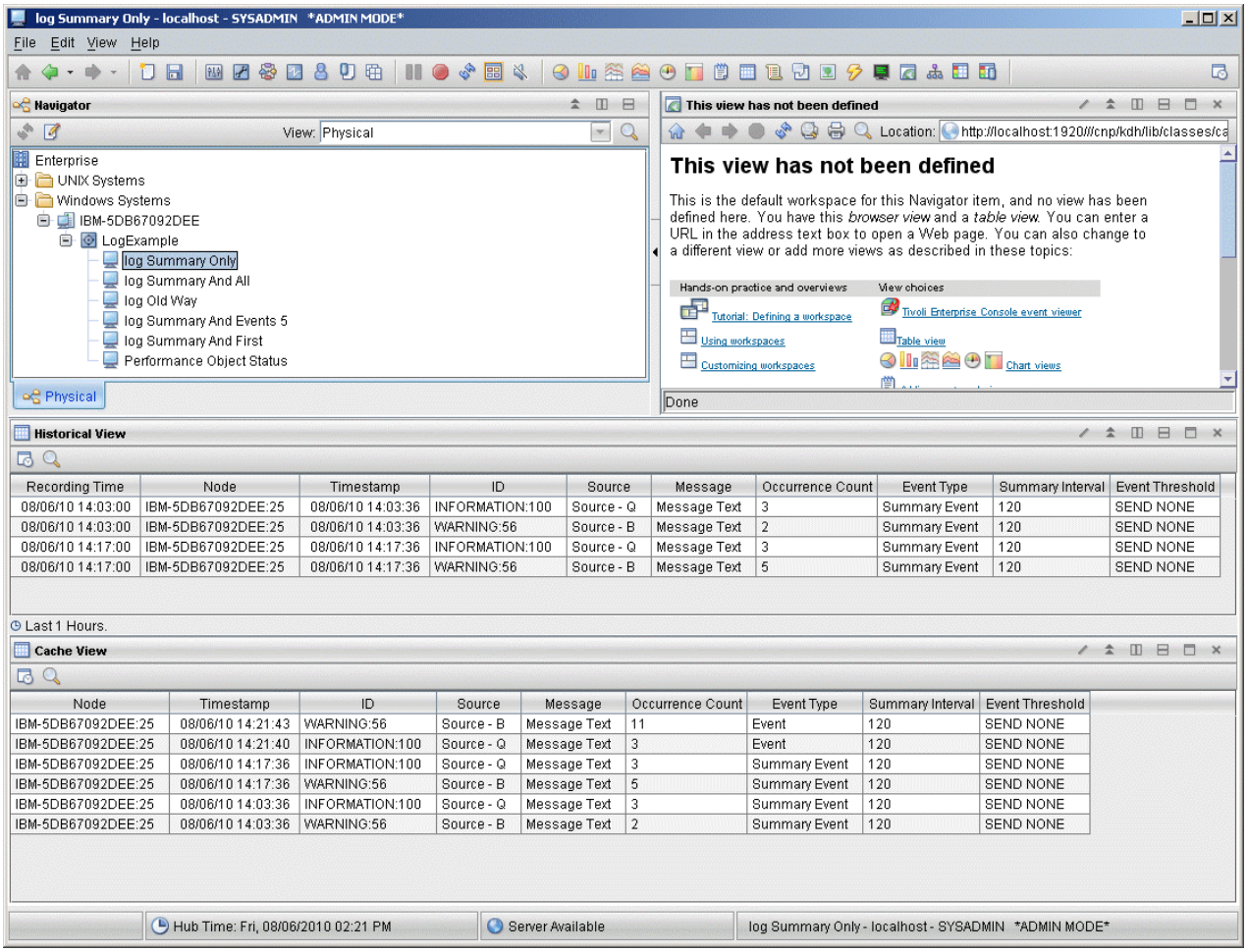


図 76. **「要約イベントのみの送信」**が選択されている場合のヒストリカル・ビューおよびキャッシュ・ビュー

1410 ページの図 77 には、「**イベント情報**」タブで、「**すべてのイベントの送信**」オプションを選択した場合のヒストリカル・ビューとキャッシュ・ビューが表示されています。すべてのイベントが両方のビューに表示されますが、各間隔の終了時に作成される要約イベントも表示されます。間隔が経過すると、リアルタイム・ビューが変更されます。 既存のイベントは要約レコードに変換され、新規イベントが追加されます。 要約間隔 (この例では 120 秒) および *SEND ALL* しきい値を表示するために使用される、他の 2 つの使用可能なイベント属性が追加されています。

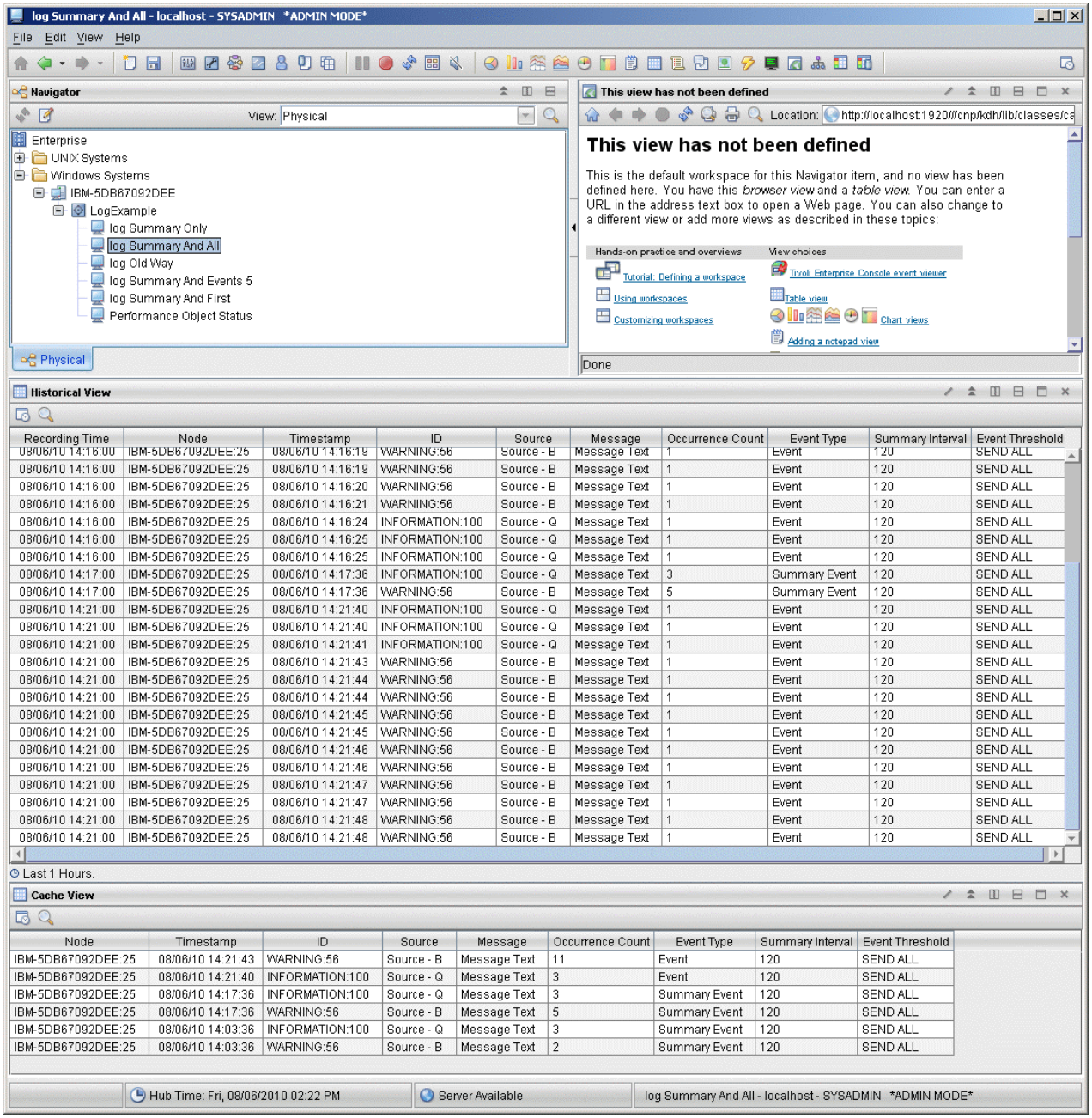


図 77. **「すべてのイベントの送信」**が選択されている場合のヒストリカル・ビューおよびキャッシュ・ビュー

1411 ページの図 78 には、「**イベント情報**」タブで、「**最初のイベントの送信**」オプションを選択した場合のヒストリカル・ビューとキャッシュ・ビューが表示されています。要約イベントは両方のビューに表示されますが、すべての新規イベントはリアルタイム (キャッシュ) ビューにのみ表示されます。各イベントのヒストリカル・ビューには、間隔で最初に受け取ったイベントのみが表示され、重複するイベントは表示されません。

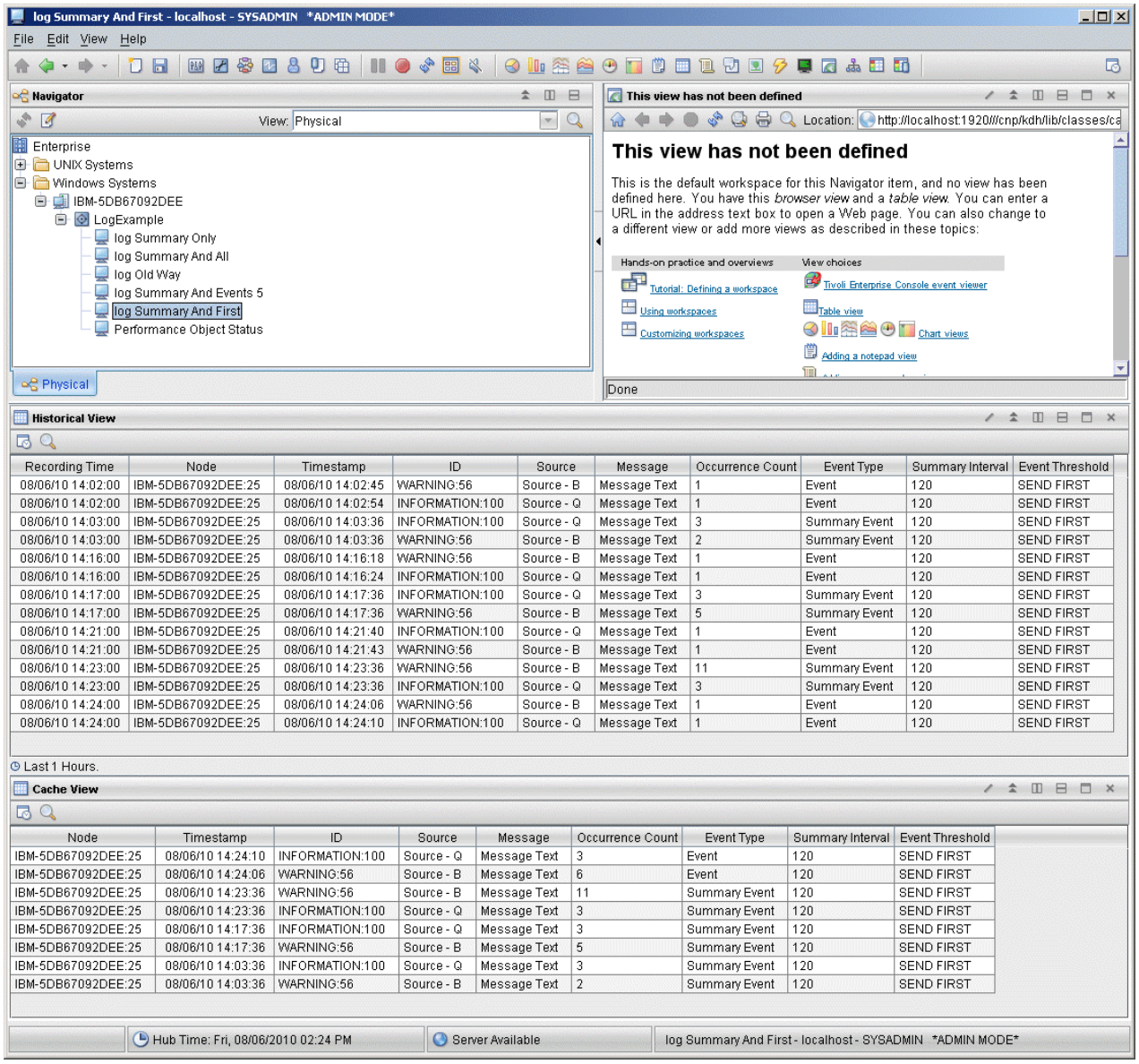


図 *78.* **「最初のイベントの送信」**が選択されている場合のヒストリカル・ビューおよびキャッシュ・ビュー

1412 ページの図 79 には、「**イベントしきい値**」オプションを選択して値 5 を入力した場合のヒストリカル・ビューとキャッシュ・ビューが表示されています。要約イベントは両方のビューに表示されますが、すべての新規イベントはリアルタイム (キャッシュ) ビューにのみ表示されます。 この例では、しきい値に 5 が指定されています。ヒストリカル・ビューには、間隔内に 5 個の重複イベント (1 番目のイベントを含む) を受信した場合にのみ、1 個のイベントが表示されます。受け取った重複イベント数が 5 未満の場合は、イベントは表示されません。間隔内で 6、7、8、または 9 個の重複イベントを受信した場合、1 個のイベントが表示されます。10 個の重複イベントを受信した場合、2 個のイベントが表示されます。

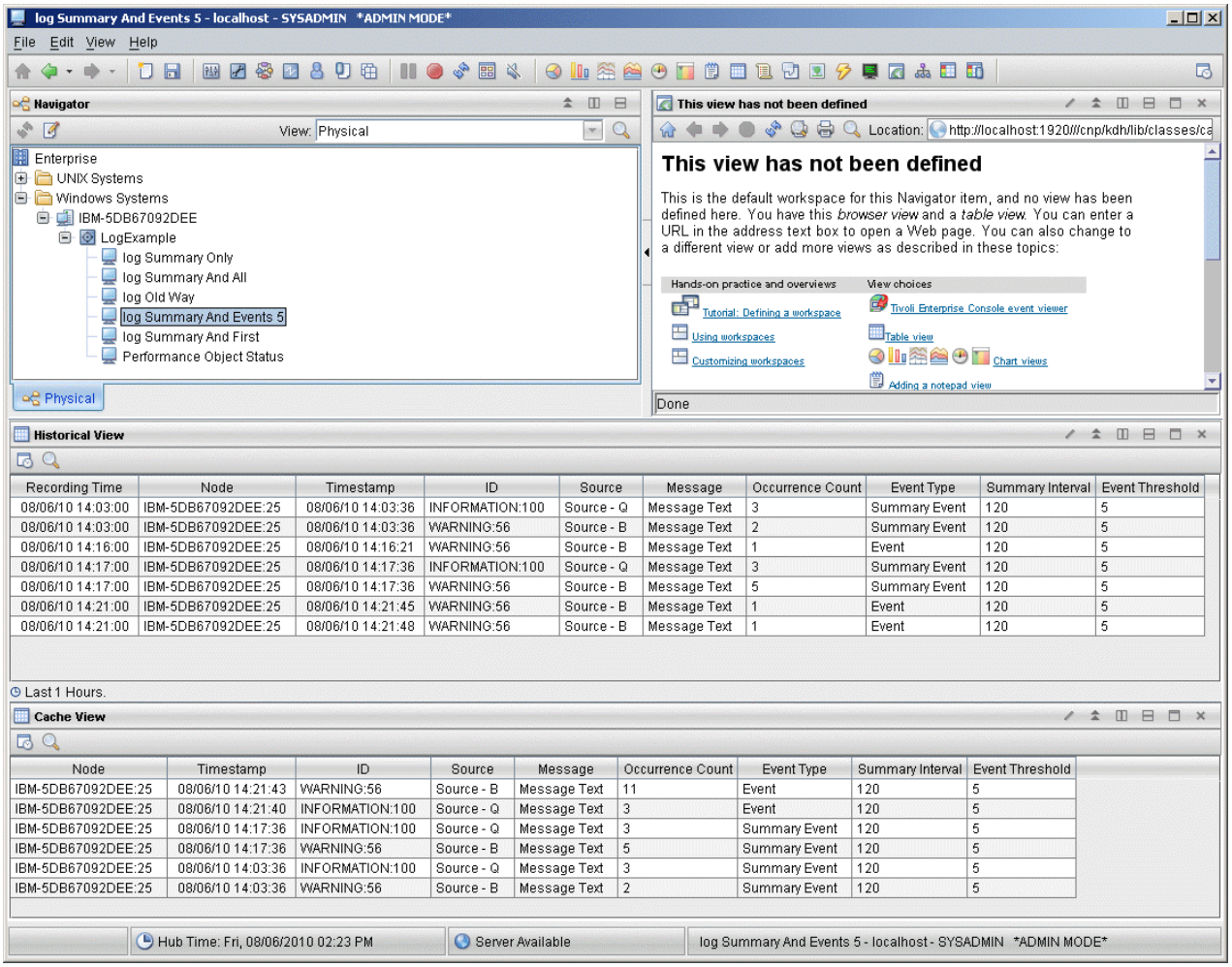


図 79. **「イベントしきい値」**が選択されている場合のヒストリカル・ビューおよびキャッシュ・ビュー

関連概念

1405 ページの『イベント・フィルタリングおよび要約』

トラブルシューティングとサポート

IBM Agent Builder のインストール時、構成時、または使用時に発生する可能性がある問題については、トラブルシューティング情報を参照してください。

IBM Cloud Application Performance Management 環境でカスタム・エージェントを開発、インストール、または使用しているときの問題のトラブルシューティングのヘルプについては、developerWorks の Cloud Application Performance Management フォーラムを参照してください。「agent_builder」タグを検索したり、項目に返信して関連する質問を行ったり、新しい項目を作成して質問したりできます。

ロギングとメッセージの参照情報について、および IBM Tivoli Monitoring 環境に関連する問題のトラブルシューティングのヘルプについては、「IBM Agent Builder Troubleshooting Reference」を参照してください。

プロジェクト・ファイルの共有

IBM Tivoli Monitoring エージェント・プロジェクトを他のユーザーと共有します。

手順

1. プロジェクトのファイルを用意します。ワークスペース・ディレクトリー内のプロジェクトと同じ名前のディレクトリーの内容全体が必要です。
   例えば、ワークスペース・ディレクトリーが `c:¥Documents and Settings¥User1¥workspace` である場合に、「`TestProject`」というプロジェクトを共有するとします。この場合、ディレクトリー `c:¥Documents and Settings¥User1¥workspace¥TestProject` とそのすべての内容を、システムからアクセスできるようにする必要があります。
2. 「**ファイル**」 > 「**インポート**」を選択します。
3. 「**IBM Tivoli Monitoring**」を開きます。
4. 「**IBM Tivoli Monitoring Agent**」を選択し、「**次へ**」をクリックします。
5. エージェントの xml ファイルの絶対パスを入力するか、「**参照**」をクリックしてそのファイルを参照します。
6. 「**完了**」をクリックします。

タスクの結果
ウィザードが完了すると、ワークスペースに新規の IBM Tivoli Monitoring エージェント・プロジェクトが表示されます。

ソリューション・インストーラー・プロジェクトの共有

ソリューション・インストーラー・プロジェクトを他のユーザーと共有します。

手順

1. プロジェクトのファイルを用意します。ワークスペース・ディレクトリー内のソリューション・インストーラー・プロジェクトと同じ名前のディレクトリーの内容全体が必要です。
   例えば、ワークスペース・ディレクトリーが `c:¥Documents and Settings¥User1¥workspace` である場合に、「`TestProject Installer`」というソリューション・インストーラー・プロジェクトを共有するとします。 この場合、ディレクトリー `c:¥Documents and Settings¥User1¥workspace¥TestProject Installer` およびそのすべての内容を、システムからアクセス可能にする必要があります。
2. 「**ファイル**」 > 「**インポート**」をクリックします。
3. 「**一般**」 を開きます。
4. 「**既存プロジェクトをワークスペースに**」を選択し、「**次へ**」をクリックします。
5. ソリューション・インストーラー・プロジェクトのルート・ディレクトリーの絶対パスを入力するか、「**参照**」をクリックして、ソリューション・インストーラー・プロジェクトのルート・ディレクトリーを参照します (この例では、`TestProject Installer` ディレクトリーです。) そのディレクトリーにあるプロジェクトがプロジェクト・リストに表示され、デフォルトで選択されています。
6. オプション: 「**プロジェクトをワークスペースにコピー**」をクリックします。
7. 「**完了**」をクリックします。

コマンド行オプション

Agent Builder コマンド行インターフェース (CLI) から使用可能なコマンド。

Tivoli Monitoring Agent Builder には、コマンド行インターフェース (CLI) が用意されており、これを使用することで、Eclipse のグラフィカル・ユーザー・インターフェース (GUI) を開始せずに Tivoli Monitoring Agent を生成することができます。例えば、ビルドの一部としてエージェントを生成することができます。

Windows システムでは、次のディレクトリーにあるバッチ・ファイルを使用して、CLI にアクセスできます。

```
install_location¥agenttoolkit.bat
```

UNIX および Linux システムの場合は、以下のディレクトリーにあるスクリプトを使用して、CLI にアクセスすることができます。

```
install_location/agenttoolkit.sh
```

この資料に記載したコマンドは Windows システム用のフォーマットであるため、ディレクトリー・パスで円記号 (¥) を使用します。

UNIX® または Linux® システムで Windows システムと同じコマンドを使用する場合は、以下のように変更します。

- ディレクトリー・パスで円記号 (¥) の代わりにスラッシュ (/) を使用します。
- `agenttoolkit.bat` スクリプトの代わりに `agenttoolkit.sh` スクリプトを使用します。

コマンド

1414 ページの表 300 には、text コマンドの各コマンド・オプションの名前と目的に関する記述がリストされています。

表 *300.* コマンド早見表

コマンド	目的
generatelocal	`itm_toolkit_agent.xml` ファイルをロードして検証し、Tivoli Monitoring Agent を実行するファイルを生成します。ローカルの Tivoli Monitoring 環境にインストールします。
generatemappingfile	カスタム IBM Tivoli Monitoring v5.x リソース・モデルを IBM Tivoli Monitoring v6 エージェントに移植するためのマッピング・ファイルを作成します。
generatezip	*productcode*`.zip` または *productcode*`.tgz` という名前の圧縮ファイルを生成します。

表から参照しているコマンドの説明では、以下の情報を示すことによってコマンドの実行方法を説明しています。

目的
コマンドの目的をリストしています。

フォーマット
コマンド行に入力する際の構文を示します。構文には、そのコマンドの名前と、パラメーターのリストが含まれています。 コマンド名の後に、各パラメーターの定義が続きます。

例
コマンドの例には、その例についての簡単な説明と、構文の例が含まれています。

使用率
コマンドおよびその目的についての説明です。

コメント
コマンドや、その他の情報を提供するテキストです。

コマンド - generatelocal

XML をロードして検証し、Tivoli Monitoring Agent を実行するためのファイルを生成するには、このコマンドを使用します。

目的

itm_toolkit_agent.xml ファイルをロードして検証し、Tivoli Monitoring Agent を実行するためのファイルを生成します。ローカルの Tivoli Monitoring 環境にインストールします。

フォーマット

Windows システムの場合:

```
install_location¥agenttoolkit.bat project_dir -generatelocal itm_install_dir
```

ここで、

install_location
Agent Builder がインストールされているディレクトリー

project_dir
itm_toolkit_agent.xml ファイルを含むディレクトリーの名前

itm_install_dir
Tivoli Monitoring がインストールされているロケーション (例えば、c:¥IBM¥ITM)。

例

Windows の場合の以下の例では、C:¥ABCAgent にあるエージェント定義が検証され、ABCAgent の実行に必要なファイルが C:¥IBM¥ITM に生成されます。

```
install_location¥agenttoolkit.bat C:¥ABCAgent -generatelocal C:¥IBM¥ITM
```

コマンド - generatemappingfile

このコマンドを使用して、IBM Tivoli Monitoring v5.x のカスタム・リソース・モデルを IBM Tivoli Monitoring v6 エージェントにマイグレーションします。

目的

このコマンドは、IBM Tivoli Monitoring v5.x のカスタム・リソース・モデルを IBM Tivoli Monitoring v6 のエージェントにマイグレーションするためのマッピング・ファイルを作成します。

フォーマット

Windows システムの場合:

```
install_location¥agenttoolkit.bat project_dir -generatemappingfile output_dir
   itm5_interp_list
```

各構成要素について説明します。

install_location
Agent Builder がインストールされているディレクトリー

project_dir
itm_toolkit_agent.xml を含むディレクトリーの名前

output_dir
マッピング・ファイルの作成先となるディレクトリーの名前

itm5_interp_list
カスタム・リソース・モデルが実行されていた ITM 5x オペレーティング・システムのコンマ区切りのリスト。以下の値をとります。

- aix4-r1
- hpux10
- linux-ix86
- linux-ppc
- linux-s390
- os2-ix86
- os400
- solaris2
- solaris2-ix86
- w32-ix86

例

Windows システムの場合:

```
install_location¥agenttoolkit.bat c:¥ABCAgent -generatemappingfile c:¥output
   linux-ix86,linux-ppc,linux-s390
```

コマンド - generatezip

XML をロードして検証し、エージェントを別のシステムにインストールするために使用できる圧縮ファイルを生成するには、このコマンドを使用します。

目的

itm_toolkit_agent.xml ファイルをロードして検証し、*productcode*.zip または *productcode*.tgz という名前の圧縮ファイルを生成します。生成された圧縮ファイルは、エージェントを別のシステムにインストールするために使用できます。ご使用の環境によっては、両方のファイル・タイプを生成できます。

フォーマット

Windows システムの場合:

```
install_location¥agenttoolkit.bat  project_dir -generatezip output_dir
```

各項目の意味は次のとおりです。

project_dir
itm_toolkit_agent.xml ファイルを含んでいるディレクトリーの名前

output_dir
圧縮ファイルの書き込み先となるディレクトリーの名前

例

Windows の場合の以下の例では、C:¥ABCAgent にあるエージェント定義が検証され、ABCAgent の実行に必要なファイルを含む圧縮ファイルが C:¥Output に生成されます。

```
install_location¥agenttoolkit.bat¥ C:¥ABCAgent -generatezip C:¥Output
```

属性のリファレンス

Agent Builder に組み込まれている各生成済み属性グループの属性について説明します。

可用性ノード

可用性属性グループには、アプリケーションの可用性データが含まれています。

この表は、プロセス、サービス (Windows のみ)、プロセス、およびコマンドの戻りコードという、アプリケーションの 3 つの側面に関する情報を含む、アプリケーションの可用性を表すための共通フォーマットとなります。

以下のリストには、可用性属性グループ内の各属性についての情報が含まれています。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

アプリケーション・コンポーネント属性 - この属性はキー属性です。

説明
アプリケーション・パーツの記述名

タイプ
ストリング

名前

属性名
アプリケーション・コンポーネント

列名
COMPONENT

名前属性

説明
プロセス、サービス、または機能テストの名前。この名前は、プロセスの実行可能ファイル名、サービスの短縮名、またはアプリケーションのテストに使用されるプロセスの名前に一致します。

タイプ
ストリング

名前

属性名
Name

列名
NAME

状況属性

説明
アプリケーション・コンポーネントの状況。

- プロセスの場合、値は「UP」、「DOWN」、「WARNING」、または「PROCESS_DATA_NOT_AVAILABLE」です。「PROCESS_DATA_NOT_AVAILABLE」は、対応するプロセスは実行されているものの、そのプロセスのリソース使用情報を収集できないプロセスについて表示されます。
- サービスの場合、値は「UP」、「DOWN」、または「UNKNOWN」です。「UNKNOWN」は、サービスがインストールされていない場合に表示されます。
- コマンドの戻りコードの場合、値は「PASSED」または「FAILED」です。

タイプ
ストリング

名前

属性名
状況

列名
STATUS

絶対パス名属性

説明
プロセスのフルネーム (プロセスに従属する情報を含む)。 名前には、プロセスが絶対パスで開始された場合は絶対パスが入ります。部分パス、あるいはプロセスによって変更されたパスを名前に入れることもできます。

タイプ
ストリング

名前

属性名
絶対パス名

列名
FULLNAME

タイプ属性

説明
アプリケーション・コンポーネントのタイプを示します。 コンポーネントはプロセス、サービス、またはコマンドの戻りコードです。

タイプ
整数 (ゲージ)

名前

属性名
タイプ

列名
タイプ

仮想サイズ属性

説明
プロセスの仮想サイズ (MB)

タイプ
整数 (ゲージ)

名前

属性名
仮想サイズ

列名
VIRTSIZE

ページ不在数 (秒当たり) 属性

説明
1 秒当たりのページ不在数により測定した、プロセスのページ不在率。この値には、プロセスに有効なデータのみが含まれます。

タイプ
整数 (ゲージ)

名前

属性名
ページ不在数 (秒当たり)

列名
PAGEFAULTS

実効ページ・セット・サイズ属性

説明
プロセスの実効ページ・セット・サイズ (MB 単位)。 この値には、プロセスに有効なデータのみが含まれます。

タイプ
整数 (ゲージ)

名前

属性名
実効ページ・セット・サイズ

列名
WORKSET

スレッド数属性

説明
このプロセスによって現在割り振られているスレッドの数。 この値には、プロセスに有効なデータのみが含まれます。

タイプ
整数 (ゲージ)

名前

属性名
スレッド数

列名
THREADS

PID 属性

説明
プロセスに関連したプロセス ID。この値には、プロセスに有効なデータのみが含まれます。

タイプ
整数 (ゲージ)

名前

属性名
PID

列名
PID

特権時間 (%) 属性

説明
使用可能なプロセッサー時間のうち、このプロセスが特権操作のために使用している時間の割合

タイプ
整数 (ゲージ)

名前

属性名
特権時間 (%)

列名
PERCPRIV

ユーザー・モード時間 (%) 属性

説明
使用可能なプロセッサー時間のうち、このプロセスがユーザー・モード操作のために使用している時間の割合

タイプ
整数 (ゲージ)

名前

属性名
ユーザー・モード時間 (%)

列名
PERCUSER

プロセッサー時間 (%) 属性

説明
このプロセスが命令を実行するためにプロセッサーを使用した際の経過時間の割合

タイプ
整数 (ゲージ)

名前

属性名
プロセッサー時間 (%)

列名
PERCPROC

コマンド行属性

説明
プロセスの開始時にコマンド行で指定したプログラム名および引数。 サービスまたは機能性テストを実行している場合、この属性の値は「*N/A*」になります。

タイプ
ストリング

名前

属性名
コマンド行

列名
CMDLINE

機能性テスト状況属性

説明
機能性テストの戻りコード。 モニター中のアプリケーションが正常に実行されている場合は「SUCCESS」が返されます。アプリケーションが正常に実行されていない場合は「NOT_RUNNING」を返します。行が機能テストを表していない場合は、「N/A」が返されます。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会では数値が返されます。定義されている値は、N/A(1)、SUCCESS (0)、GENERAL_ERROR (2)、WARNING (3)、NOT_RUNNING (4)、DEPENDENT_NOT_RUNNING (5)、ALREADY_RUNNING (6)、PREREQ_NOT_RUNNING (7)、TIMED_OUT (8)、DOESNT_EXIST (9)、UNKNOWN (10)、DEPENDENT_STILL_RUNNING (11)、または INSUFFICIENT_USER_AUTHORITY (12) です。その他の値では、Tivoli Enterprise Portal に数値が表示されます。

名前

属性名
機能性テスト状況

列名
FUNCSTATUS

機能性テスト・メッセージ属性

説明
「機能性テストの状況」に対応するテキスト・メッセージ。 この属性はコマンドの戻りコードに対してのみ有効です。

タイプ
ストリング

名前

属性名
機能性テスト・メッセージ

列名
FUNCMSG

「パフォーマンス・オブジェクトの状況」ノード

「パフォーマンス・オブジェクト状況」属性グループを使用して、エージェントを構成するすべての属性グループの状況を表示します。 属性グループはそれぞれ、この表または他のタイプのビューの単一行によって表されます。属性グループの状況には、その属性グループに関する最後のデータ収集試行の結果、またはデータ受信イベントの結果が反映されます。状況情報を調べると、エージェントが正しく動作しているかどうか確認することができます。エージェントがデータを収集せずに、データ (イベント・データ) を受信する場合は、サンプル・データに関連する属性には有用なデータは含まれません。リストされている属性のうち最初の 7 個のみが、イベント・データに関連しています。

ヒストリカル・グループ
この属性グループは、Tivoli Data Warehouse で使用するのに適格です。

属性の説明
パフォーマンス・オブジェクト状況属性グループの各属性についての情報を、 以下のリストに示します。

ノード属性: この属性はキー属性です。

説明
エージェントの管理対象システム名。

タイプ
ストリング

ウェアハウス名
NODE

タイム・スタンプ属性

説明
データ収集時のエージェントの現地時間。

タイプ
ストリング

ウェアハウス名
TIMESTAMP

照会名属性: この属性はキー属性です。

説明
属性グループの名前。

タイプ
ストリング

ウェアハウス名
QUERY_NAME または ATTRGRP

オブジェクト名属性

説明
パフォーマンス・オブジェクトの名前。

タイプ
ストリング

ウェアハウス名
OBJECT_NAME または OBJNAME

オブジェクト・タイプ属性

説明
パフォーマンス・オブジェクトのタイプ。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。ウェアハウスおよび照会は、括弧内に示された値を返します。 以下の値を定義します。

- WMI (0)
- パフォーマンス・モニター (1)
- WMI 関連グループ (2)
- JMX (3)

- SNMP (4)
- シェル・コマンド (5)
- 結合されたグループ (6)
- CIMOM (7)
- カスタム (8)
- ロールアップ・データ (9)
- WMI リモート・データ (10)
- ログ・ファイル (11)
- JDBC (12)
- 構成ディスカバリー (13)
- NT イベント・ログ (14)
- フィルター (15)
- SNMP イベント (16)
- PING (17)
- ディレクター・データ (18)
- ディレクター・イベント (19)
- SSH リモート・シェル・コマンド (20)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
OBJECT_TYPE または OBJTYPE

オブジェクト状況属性

説明
パフォーマンス・オブジェクトの状況。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。ウェアハウスおよび照会は、括弧内に示された値を返します。 以下の値を定義します。

- アクティブ (0)
- 非アクティブ (1)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
OBJECT_STATUS または OBJSTTS

エラー・コード属性

説明
照会に関連付けられているエラー・コード。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。ウェアハウスおよび照会は、括弧内に示された値を返します。 以下の値を定義します。

- エラーなし (0)
- 一般エラー (1)
- オブジェクトが見つかりません (2)
- カウンターが見つかりません (3)
- ネーム・スペース・エラー (4)
- オブジェクトは現在使用不可です (5)

- COM ライブラリー初期化障害 (6)
- セキュリティー初期化障害 (7)
- プロキシー・セキュリティー障害 (9)
- 返されるインスタンスはありません (10)
- アソシエーター照会は失敗しました (11)
- 参照照会は失敗しました (12)
- 応答が受け取られていません (13)
- 結合照会が見つかりません (14)
- 照会 1 の結果に結合属性が見つかりません (15)
- 照会 2 の結果に結合属性が見つかりません (16)
- 照会 1 は Singleton ではありません (17)
- 照会 2 は Singleton ではありません (18)
- 照会 1 で返されるインスタンスはありません (19)
- 照会 2 で返されるインスタンスはありません (20)
- ロールアップ照会が見つかりません (21)
- ロールアップ属性が見つかりません (22)
- ファイルがオフラインです (23)
- ホスト名がありません (24)
- ライブラリーが欠落しています (25)
- 属性カウントが不一致です (26)
- 属性名が不一致です (27)
- 共通データ・プロバイダーが開始していません (28)
- コールバック登録エラー (29)
- MDL ロード・エラー (30)
- 認証が失敗しました (31)
- ホスト名を解決できません (32)
- サブノードが使用不可です (33)
- サブノードが構成内に見つかりません (34)
- 属性エラー (35)
- CLASSPATH エラー (36)
- 接続障害 (37)
- フィルター構文エラー (38)
- ファイル名欠落 (39)
- SQL 照会エラー (40)
- SQL フィルター照会エラー (41)
- SQL データベース照会エラー (42)
- SQL データベース・フィルター照会エラー (43)
- ポートのオープンに失敗しました (44)
- アクセスが拒否されました (45)
- タイムアウト (46)
- インプリメントされていません (47)
- 誤った値を要求しました (48)
- 応答が大きすぎます (49)

- 一般的な応答エラー (50)
- スクリプトはゼロ以外を返しました (51)
- スクリプトが見つかりません (52)
- スクリプトの起動エラー (53)
- 構成ファイルが存在しません (54)
- 構成ファイルへのアクセスが拒否されました (55)
- 無効な構成ファイル (56)
- EIF の初期化に失敗しました (57)
- フォーマット・ファイルを開けません (58)
- フォーマット・ファイル構文エラー (59)
- リモート・ホスト利用不可 (60)
- イベント・ログが存在しません (61)
- ping ファイルが存在しません (62)
- ping デバイス・ファイルがありません (63)
- ping デバイス・リスト・ファイルがありません (64)
- SNMP のパスワードが欠落しています (65)
- 無効になっています (66)
- URL ファイルが見つかりません (67)
- XML 構文解析エラー (68)
- 初期化されていません (69)
- ICMP ソケット障害 (70)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
ERROR_CODE または ERRCODE

最終収集開始属性

説明
このグループの最後のデータ収集が開始された時刻。

タイプ
列挙値を持つタイム・スタンプ。Tivoli Enterprise Portal ではストリングが表示されます。 ウェアハウスおよび照会は、括弧内の値を戻します。 以下の値を定義します。

- 収集なし (0691231190000000)
- 収集なし (0000000000000001)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
LAST_COLLECTION_START または COLSTRT

最終収集完了属性

説明
このグループの最後のデータ収集が終了した時刻。

タイプ
列挙値を持つタイム・スタンプ。Tivoli Enterprise Portal ではストリングが表示されます。 ウェアハウスおよび照会は、括弧内の値を戻します。 以下の値を定義します。

- 収集なし (0691231190000000)
- 収集なし (0000000000000001)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
LAST_COLLECTION_FINISHED または COLFINI

最終収集期間属性

説明
このグループのデータ収集が完了した最新の期間 (秒)。

タイプ
小数点以下 2 桁の精度の実数 (32 ビット・カウンター)

ウェアハウス名
LAST_COLLECTION_DURATION または COLDURA

平均収集期間属性

説明
このグループのすべてのデータ収集の平均期間 (秒)。

タイプ
列挙型の値を持つ、小数点以下 2 桁の精度の実数 (32 ビット・カウンター)。 Tivoli Enterprise Portal ではストリングが表示されます。ウェアハウスおよび照会は、括弧内の値を戻します。以下の値を定義します。

- データなし (-100)

その他の値は、Tivoli Enterprise Portal ではエージェントから戻された値です。

ウェアハウス名
AVERAGE_COLLECTION_DURATION または COLAVGD

最新表示間隔属性

説明
このグループが最新表示される間隔 (秒)。

タイプ
整数 (32 ビット・カウンター)

ウェアハウス名
REFRESH_INTERVAL または REFRINT

収集の数属性

説明
エージェントの開始以降にこのグループが収集された回数。

タイプ
整数 (32 ビット・カウンター)

ウェアハウス名
NUMBER_OF_COLLECTIONS または NUMCOLL

キャッシュ・ヒット数属性

説明
このグループの外部データ要求がキャッシュで満たされる回数。

タイプ
整数 (32 ビット・カウンター)

ウェアハウス名
CACHE_HITS または CACHEHT

キャッシュ・ミス数属性

説明
このグループの外部データ要求がキャッシュで使用できなかった回数。

タイプ
整数 (32 ビット・カウンター)

ウェアハウス名
CACHE_MISSES または CACHEMS

キャッシュ・ヒット率属性

説明
このグループの外部データ要求がキャッシュから満たされる比率。

タイプ
小数点以下 2 桁の精度の実数 (32 ビット・カウンター)

ウェアハウス名
CACHE_HIT_PERCENT または CACHPCT

スキップされた間隔属性

説明
次の収集の開始予定時刻に前の収集がまだ実行中だったため、バックグラウンド・データ収集がスキップされた回数。

タイプ
整数 (32 ビット・カウンター)

ウェアハウス名
INTERVALS_SKIPPED または INTSKIP

「スレッド・プールの状況」属性グループ

「スレッド・プールの状況」属性グループには、データを非同期に収集するために使用する内部スレッド・プールの状況を反映した情報が含まれています。

以下はこの属性グループの属性のリストです。太字テキストの名前は Tivoli Enterprise Portal でその属性がどのように表示されるかを示しています。

以下のリストは、スレッド・プールの状況属性グループ内の各属性に関する情報を示したものです。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ行が作成され、エージェントから Tivoli Enterprise Monitoring Server に送信されたときに、エージェント・システムから収集した時間。または、履歴を残す目的でデータ行が保存されたときの時間です。エージェント・システムのローカル時間帯を示します。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

スレッド・プール・サイズ属性

説明
現在スレッド・プール内に存在するスレッドの数。

タイプ
整数

名前

属性名
スレッド・プール・サイズ

列名
THPSIZE

スレッド・プール最大サイズ属性

説明
スレッド・プール内に存在することが許可されているスレッドの最大数。

タイプ
整数

名前

属性名
スレッド・プール最大サイズ

列名
TPMAXSZ

スレッド・プールのアクティブ・スレッド属性

説明
現在作業を実行しているアクティブなスレッド・プール内のスレッドの数。

タイプ
整数

名前

属性名
スレッド・プールのアクティブ・スレッド

列名
TPACTTH

スレッド・プールの平均アクティブ・スレッド属性

説明
同時に作業を実行しているアクティブなスレッド・プール内のスレッドの平均数。

タイプ
整数

名前

属性名
スレッド・プールの平均アクティブ・スレッド

列名
TPAVGAT

スレッド・プールの最小アクティブ・スレッド属性

説明
同時に作業を実行しているアクティブなスレッド・プール内のスレッドの最小数。

タイプ
整数

名前

属性名
スレッド・プールの最小アクティブ・スレッド

列名
TPMINAT

スレッド・プールの最大アクティブ・スレッド属性

説明
同時に作業を実行しているアクティブなスレッド・プール内のスレッドのピーク数。

タイプ
整数

名前

属性名
スレッド・プールの最大アクティブ・スレッド

列名
TPMAXAT

スレッド・プール・キューの長さ属性

説明
現在スレッド・プール・キューで待機しているジョブの数。

タイプ
整数

名前

属性名
スレッド・プール・キューの長さ

列名
TPQLGTH

スレッド・プール・キューの平均長さ属性

説明
この実行中のスレッド・プール・キューの平均長さ。

タイプ
整数

名前

属性名
スレッド・プール・キューの平均長さ

列名
TPAVGQL

スレッド・プール・キューの最小長さ属性

説明
スレッド・プール・キューが到達した最小の長さ。

タイプ
整数

名前

属性名
スレッド・プール・キューの最小長さ

列名
TPMINQL

スレッド・プール・キューの最大長さ属性

説明
スレッド・プール・キューが到達したピークの長さ。

タイプ
整数

名前

属性名
スレッド・プール・キューの最大長さ

列名
TPMAXQL

スレッド・プールの平均ジョブ待機属性

説明
ジョブがスレッド・プール・キューで待機するのに費やす平均時間。

タイプ
整数

名前

属性名
スレッド・プールの平均ジョブ待機

列名
TPAVJBW

スレッド・プール合計ジョブ属性

説明
エージェントの開始以降、プール内のすべてのスレッドが実行したジョブの数。

タイプ
整数

名前

属性名
スレッド・プール合計ジョブ

列名
TPTJOBS

「イベント・ログ」属性ノード

イベント・ログ属性グループには、アプリケーションに関するすべての最新イベント・ログ項目が含まれています。

デフォルト状態のエージェントは、そのエージェントが開始された後に発生したイベントのみを表示します。 イベントが発生してから 1 時間後に、それらのイベントはイベント・ログ・ビューから削除されます。

以下のリストには、イベント・ログ属性グループ内の各属性についての情報が含まれています。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

ログ名属性

説明
イベント・ログ -「アプリケーション」、「システム」、「セキュリティー」、またはアプリケーション固有のログ

タイプ
ストリング

名前

属性名
ログ名

列名
LOGNAME

イベント・ソース属性

説明
アプリケーションで定義されるイベント・ソース

タイプ
ストリング

名前

属性名
イベント・ソース

列名
EVTSOURCE

イベント・タイプ属性

説明
イベント・タイプ - エラー (0)、警告 (1)、通知 (2)、監査成功 (3)、 監査失敗 (4)、不明 (5)

タイプ
整数

名前

属性名
イベント・タイプ

列名
EVTTYPE

イベント ID 属性

説明
イベントの ID

タイプ
整数

名前

属性名
イベント ID

列名
EVTID

イベント・カテゴリー属性

説明
イベントのカテゴリー

タイプ
ストリング

名前

属性名
イベント・カテゴリー

列名
EVTCATEG

メッセージ属性

説明
イベント・メッセージ

タイプ
ストリング

名前

属性名
メッセージ

列名
MESSAGE

生成時刻属性

説明
イベントが生成された時刻

タイプ
時刻

名前

属性名
生成時刻

列名
TIMESTAMP

ログ・ファイルの要約

データ・ソースの拡張プロパティーでオプションが 選択されている場合は、この属性グループの属性が要約属性グループに含められます。

データ・ソースの拡張プロパティーで「**要約属性グループに属性を組み込む**」が選択されている場合は、ログ・ファイル・データ・ソースごとに「要約」ノードが作成されます。要約ノードの名前は、データ・ソース名の末尾に「要約」を追加した名前になります。

ログ・ファイル要約属性グループの各デフォルト属性についての情報を、以下のリストに示します。これらの属性は、常に要約属性グループに含められます。「**要約属性グループに属性を組み込む**」を選択した場合 (1260 ページの『ログ・ファイルのモニター』のステップ 1265 ページの『9』 を参照)、そのログ属性グループの要約属性グループにも選択した各属性が含まれます。値は、ログ・ファイル属性グルー プ内の対応する属性のコピーです。

追加された属性のすべてがまとめられて 1 つのキーとなり、要約表にはキーの固有のセットごとに 1 つの行が含められます。行は、指定されたすべてのキーとそれらに対応する属性で報告された値が一致しているログ・レコードの、該当する間隔における受信数を示します。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

間隔単位属性

説明
要約属性の生成間隔の秒数

タイプ
整数 (ゲージ)

名前

属性名
間隔単位

列名
IU

間隔属性

説明
次に大きい時間単位 (例えば、時間内の分) 内での現行間隔のオフセット

タイプ
整数 (ゲージ)

名前

属性名
間隔

列名
INV

発生属性

説明
間隔中に記録される発生回数

タイプ
整数 (ゲージ)

名前

属性名
発生

列名
OCC

ローカル・タイム・スタンプ属性

説明
要約データが生成された時間

タイプ
タイム・スタンプ

名前

属性名
ローカル・タイム・スタンプ

列名
LTS

日時属性

説明
要約データが生成された時間

タイプ
ストリング

名前

属性名
日時

列名
DT

間隔単位名属性

説明
間隔単位の単語説明

タイプ
ストリング

名前

属性名
間隔単位名

列名
IUN

「AIX バイナリー・ログ」属性グループ

「AIX バイナリー・ログ」属性グループには、指定された errpt コマンド・ストリングにより選択された AIX バイナリー・ログのイベントが表示されます。

「AIX バイナリー・ログ」属性グループの各属性に関する情報を以下のリストにまとめます。

注 : Agent Builder では、ID、Errpt タイム・スタンプ、タイプ、クラス、リソース名、および説明の各属性の削除、再配列、およびサイズ変更はできません。エージェントは、errpt コマンドから返されるデータを、テキスト行内の列に基づいて解析します。これらの列は、ID、Errpt タイム・スタンプ、タイプ、クラス、リソース名、および説明の各属性の順序とサイズによって定義されます。これらの属性の削除、再配列、またはサイズ変更を行うと、各列が格納される属性が変更されます。このため、Tivoli Monitoring に正しくない結果行が表示されます。

ただし、これらの属性の名前を変更することはできます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「ID」属性 - この属性はキー属性です。

説明
errpt により報告されるイベント ID。

タイプ
ストリング

名前

属性名
ID

列名
IDENTIFIER

「Errpt タイム・スタンプ」属性

説明
errpt により報告済みであるとイベントが記録される時刻。

注 : この属性は実行時には表示されません。この属性には未加工値が格納されています。この属性から派生するその他の属性では、より処理しやすい形式で値が表示されます。この属性は、この目的で Agent Builder から使用可能ですが、デフォルトでは、Tivoli Monitoring 環境で実行時には表示されません。この属性を表示するには、Agent Editor の「**データ・ソース定義**」ページでこの属性を選択し、「**Tivoli Enterprise Portal の属性表示**」を選択します。

タイプ
ストリング

名前

属性名
Errpt タイム・スタンプ

列名
ERRPTTIMES

タイプ

説明
errpt から報告される 1 文字のイベント・タイプです。これは I(NFO)、P(END/ERF/ERM)、T(EMP)、および U(NKN) のいずれかです。

タイプ
ストリング

名前

属性名
タイプ

列名
タイプ

クラス属性 - この属性はキー属性です。

説明
errpt により報告されるイベント・クラス。「ハードウェア」、「ソフトウェア」、「オペレーター」、および「判別不能」のいずれかです。これらの値は列挙されます。 シチュエーションで使用する未加工値は、H、S、O および U です。

タイプ
ストリング

名前

属性名
クラス

列名
CLASS

リソース名

説明
errpt により報告されるリソース名。エラー・レコードの発生元を示します。

タイプ
ストリング

名前

属性名
リソース名

列名
RESOURCENA

「**説明**」属性

説明
errpt により報告される説明。これは通常、エラーの性質を説明する短いテキスト・メッセージです。

タイプ
ストリング

名前

属性名
説明

列名
DESCRIPTIO

「**ログ・ファイル**」属性

説明
バイナリー errpt ログの絶対パス名 (パスを含む)。

注 : この属性は実行時には表示されません。この属性には未加工値が格納されています。この属性から派生するその他の属性では、より処理しやすい形式で値が表示されます。この属性は、この目的で Agent Builder から使用可能ですが、デフォルトでは、Tivoli Monitoring 環境で実行時には表示されません。この属性を表示するには、Agent Editor の「**データ・ソース定義**」ページでこの属性を選択し、「**Tivoli Enterprise Portal の属性表示**」を選択します。

タイプ
ストリング

名前

属性名
ログ・ファイル

列名
LOGFILE

「**システム**」属性

説明
エラーが収集されたシステムのホスト名。

タイプ
ストリング

名前

属性名
システム

列名
SYSTEM

「**ログ名**」属性

説明
レコード収集元バイナリー errpt ログのベース名

タイプ
ストリング

名前

属性名
ログ名

列名
LOGNAME

「**ログ・パス**」**属性**

説明
レコード収集元バイナリー errpt ログが格納されているディレクトリーの名前

タイプ
ストリング

名前

属性名
ログ・パス

列名
LOGPATH

「**エントリー時刻**」**属性**

説明
errpt により報告済みであるとイベントが記録される時刻 (Tivoli タイム・スタンプ・フォーマット)。この時刻は、「**タイム・スタンプ**」フィールドに記録されている、エージェントがイベントを受け取った時刻と同一である必要はありません。

タイプ
タイム・スタンプ

名前

属性名
エントリー時刻

列名
ENTRYTIME

モニターと通知の属性グループ

モニターと通知の属性グループについての定義。

最初の 4 つはモニターに関連した属性グループであり、最後は通知に関連した属性グループです (すべて JMX に関連しています)。

それぞれの属性グループには、イベントに基づくかどうかを示す標識が表示されます。イベントに基づかない属性グループについては、必要なときにデータが収集されます。 イベントに基づく属性グループについては、受け取ったイベントのうち、最新の 100 件のイベントのキャッシュがエージェントによって保守されます。Tivoli Enterprise Portal からの要求に応答するときには、それらのイベントが使用されます。それらのイベントは、シチュエーションによる分析とウェアハウス処理のためにすぐに転送されます。

カウンター通知

「カウンター通知」属性グループは、イベントに基づかない属性グループであり、すべてのカウンター・モニターが受け取るイベントを送信します。

カウンター通知属性グループの各属性に関する情報を以下のリストにまとめます。

「**ノード**」**属性 - この属性はキー属性です**

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

通知タイプ属性

説明
受け取った通知のタイプ。MBean の監視対象属性がどのように通知を起動したかが記述されます。

タイプ
ストリング

名前

属性名
通知タイプ

列名
NOTIFICATI

モニター ID 属性

説明
この通知を生成したモニターのモニター ID

タイプ
整数

名前

属性名
モニター ID

列名
MONITOR_ID

監視対象 MBean 属性

説明
属性がモニター対象になっている MBean

タイプ
ストリング

名前

属性名
監視対象 MBean

列名
OBSERVED_M

「監視対象属性」属性

説明
監視対象 MBean でモニターされる属性の名前

タイプ
ストリング

名前

属性名
監視対象属性

列名
OBSERVED_A

しきい値属性

説明
モニターの現在のしきい値

タイプ
ストリング

名前

属性名
しきい値

列名
THRESHOLD

オフセット属性

説明
属性がしきい値を超えるたびにしきい値に追加される値。その値によって新しいしきい値が形成されます。

タイプ
ストリング

名前

属性名
オフセット

列名
OFFSET

モジュラス属性

説明
属性の最大値。この値に達するとロールオーバーして、再びゼロからのカウントが始まります。

タイプ
整数

名前

属性名
モジュラス

列名
MODULUS

カウンター値属性

説明
通知を起動したカウンターの値

タイプ
整数

名前

属性名
カウンター値

列名
COUNTER_VA

通知タイム・スタンプ属性

説明
通知が起動された時刻

タイプ
時刻

名前

属性名
通知タイム・スタンプ

列名
NOTIFICAT0

通知メッセージ属性

説明
通知のメッセージ

タイプ
ストリング

名前

属性名
通知メッセージ

列名
NOTIFICAT1

ゲージ通知
「ゲージ通知」属性グループは、イベントに基づかない属性グループであり、すべてのゲージ・モニターが受け取るイベントを送信します。

ゲージ通知属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

通知タイプ属性

説明
受け取った通知のタイプ。MBean の監視対象属性がどのように通知を起動したかが記述されます。

タイプ
ストリング

名前

属性名
通知タイプ

列名
NOTIFICATI

モニター ID 属性

説明
この通知を生成したモニターのモニター ID

タイプ
整数

名前

属性名
モニター ID

列名
MONITOR_ID

監視対象 MBean 属性

説明
属性がモニター対象になっている MBean

タイプ
ストリング

名前

属性名
監視対象 MBean

列名
OBSERVED_M

「監視対象属性」属性

説明
監視対象 MBean でモニターされる属性の名前

タイプ
ストリング

名前

属性名
監視対象属性

列名
OBSERVED_A

下限しきい値属性

説明
監視対象属性がクロスするかどうかをモニターが監視しているしきい値

タイプ
ストリング

名前

属性名
下限しきい値

列名
LOW_THRESH

上限しきい値属性

説明
監視対象属性がクロスするかどうかをモニターが監視しているしきい値

タイプ
ストリング

名前

属性名
上限しきい値

列名
HIGH_THRES

ゲージ値属性

説明
通知を起動したゲージの値

タイプ
ストリング

名前

属性名
ゲージ値

列名
MODULUSGAUGE_VALU

通知タイム・スタンプ属性

説明
通知が起動された時刻

タイプ
時刻

名前

属性名
通知タイム・スタンプ

列名
NOTIFICAT0

通知メッセージ属性

説明
通知のメッセージ

タイプ
ストリング

名前

属性名
通知メッセージ

列名
NOTIFICAT1

登録済みモニター
「登録済みモニター」属性グループは、イベントに基づく属性グループであり、エージェントによって作成されるすべての JMX モニターのリストを表示します。

登録済みモニター属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「モニター ID」属性 - この属性はキー属性です。

説明
モニターの固有の整数 ID

タイプ
整数

名前

属性名
モニター ID

列名
MONITOR_ID

モニター・パラメーター属性

説明
モニターの作成に使用されたパラメーター

タイプ
ストリング

名前

属性名
モニター・パラメーター

列名
MONITOR_PA

モニター名属性

説明
モニター MBean の JMX オブジェクト名

タイプ
ストリング

名前

属性名
モニター名

列名
MONITOR_NA

ストリング通知
「ストリング通知」属性グループは、イベントに基づかない属性グループであり、すべてのストリング・モニターが受け取ったイベントを送信します。

ストリング通知属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

通知タイプ属性

説明
受け取った通知のタイプ。MBean の監視対象属性がどのように通知を起動したかが記述されます。

タイプ
ストリング

名前

属性名
通知タイプ

列名
NOTIFICATI

「モニター ID」属性 - この属性はキー属性です。

説明
モニターの固有の整数 ID

タイプ
整数

名前

属性名
モニター ID

列名
MONITOR_ID

監視対象 MBean 属性

説明
属性がモニター対象になっている MBean

タイプ
ストリング

名前

属性名
監視対象 MBean

列名
OBSERVED_M

「監視対象属性」属性

説明
監視対象 MBean でモニターされる属性の名前

タイプ
ストリング

名前

属性名
監視対象属性

列名
OBSERVED_A

比較ストリング属性

説明
比較操作で使用するストリング

タイプ
ストリング

名前

属性名
比較ストリング

列名
COMPARE_ST

ストリング値属性

説明
通知を起動した属性の値

タイプ
ストリング

名前

属性名
ストリング値

列名
STRING_VAL

通知タイム・スタンプ属性

説明
通知が起動された時刻

タイプ
時刻

名前

属性名
通知タイム・スタンプ

列名
NOTIFICAT0

通知メッセージ属性

説明
通知のメッセージ

タイプ
ストリング

名前

属性名
通知メッセージ

列名
NOTIFICAT1

SNMP イベントの属性グループ

SNMP イベントの属性グループは、トラップおよび通知の受信に使用します。 これらの属性グループはイベント・ベースの属性グループです。

SNMP イベントの属性グループの各属性に関する情報を以下のリストにまとめます。

注 : これらの属性のデフォルトの表示名は 変更できます。これらの表示名は、各属性の内部 ID とは異なります。

エンタープライズ OID
トラップを生成したエンタープライズ OID。

ソース・アドレス
トラップを送信した SNMP エージェントのホスト名または IP アドレス。

汎用トラップ
受信したトラップから取り出された汎用トラップ番号。可能な値は 次のとおりです。

0 コールド・スタート
1 ウォーム・スタート
2 リンク停止
3 リンク稼働
4 認証障害
5 Egp ネイバー・ロス

特定トラップ
受信したトラップから取り出されたエンタープライズ特定トラップ番号。汎用トラップが 6 の場合にのみ適用されます。

アラート名
トラップ構成ファイルの定義に指定されたトラップ名。

カテゴリー
トラップ構成ファイルの定義に指定されたトラップ・カテゴリー。

説明
トラップ構成ファイルの定義に指定されたトラップ記述。この記述の最大長は 256 文字です。

エンタープライズ名
トラップ構成ファイルで指定され、トラップ・オブジェクト ID を通じて 判別されたトラップ・エンタープライズ名。

ソース状況
トラップ構成ファイルのトラップ定義で指定されている、トラップが発生したエージェントのトラップ送信後の状況。

ソース・タイプ
トラップ構成ファイルのトラップ定義で指定されている、トラップが発生したエージェントのタイプ。

イベント変数
トラップ・プロトコル・データ単位 (PDU) で受信した 変数バインディング (VarBind) データ。ストリングは以下のように構成されます。

```
{OID[type]=value}{OID[type]=value}{oid[type]=value}...
```

各項目の意味は次のとおりです。

oid
MIB 変数のオブジェクト ID

type
SMI データ・タイプ

value
変数値

{}
各トリプレットは中括弧 ({}) で囲みます。

注 :「アラート名」、「カテゴリー」、「説明」、「エンタープライズ名」、「ソース状況」、および「ソース・タイプ」の各属性は、追加情報を提供します。「**SNMP MIB ブラウザー**」ウィンドウで「**トラップ構成ファイルで定義されている情報を表示する属性を含める**」チェック・ボックスを選択すると、これらの属性が組み込まれます。

JMX イベントの属性グループ

JMX イベントの属性グループは、MBean サーバーから通知を受け取るために使用します。

これらの属性グループは、イベントに基づかない属性グループであり、エージェント開発者による編集が可能な以下の属性で生成されます。

JMX イベントの属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

タイプ属性

説明
通知のタイプ

タイプ
ストリング

名前

属性名
タイプ

列名
タイプ

ソース属性

説明
通知を送信する原因になった MBean

タイプ
ストリング

名前

属性名
ソース

列名
SOURCE

シーケンス番号属性

説明
通知オブジェクトのシーケンス番号

タイプ
ストリング

名前

属性名
シーケンス番号

列名
SEQUENCE_N

メッセージ属性

説明
通知メッセージ

タイプ
ストリング

名前

属性名
メッセージ

列名
MESSAGE

ユーザー・データ属性

説明
通知のユーザー・データ・オブジェクト

タイプ
ストリング

名前

属性名
ユーザー・データ

列名
USER_DATA

ping 属性グループ

ping 属性グループには、デバイスのリストに送信された ICMP ping の 結果が含まれています。

ping 属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名。

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ行が作成され、エージェントから Tivoli Enterprise Monitoring Server に送信されたときに、エージェント・システムから収集した時間。または、履歴を残す目的でデータ行が保存されたときの時間です。 エージェント・システムのローカル時間帯を示します。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「アドレス」属性 - この属性はキー属性です。

説明
モニターされるホストの IP アドレス。

タイプ
列挙値を持つストリング。IP アドレスが不明な場合は、値 UNKNOWN_ADDRESS が表示されます。この列挙値の場合、ウェアハウスおよび照会は 0.0.0.0 を返します。それ以外の IP アドレスの値はそのまま表示されます。

名前

属性名
アドレス

列名
PNGADDR

「デバイス・エントリー」属性 - この属性はキー属性です。

説明
このノードのデバイス・リスト・ファイルのエントリー。

タイプ
ストリング

名前

属性名
デバイス・エントリー

列名
PINGDEVC

「現行の応答時間」属性

説明
管理対象ノードに対する ICMP 要求の、 現行のネットワーク応答時間 (ミリ秒単位)。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。数値は、ウェアハウスと照会から返されます。定義されている値は TIMEOUT(-1) および SEND_FAILURE(-2) です。その他の値では、数値が表示されます。

名前

属性名
現行の応答時間

列名
PINGRSTM

名前属性

説明
管理対象ノードのホスト名。 ノード・アドレスが DNS で解決できない場合は、 小数点付き 10 進数の IP アドレスが表示されます。

タイプ
列挙値を持つストリング。ホスト名が不明な場合は 値 UNKNOWN_HOSTNAME が表示されます。この列挙値の場合、ウェアハウスおよび 照会は 0.0.0.0 を返します。それ以外のホスト名の値は そのまま表示されます。

名前

属性名
Name

列名
PNGNAME

「ノードの説明」属性

説明
管理対象ノードの説明。

タイプ
ストリング

名前

属性名
ノードの説明

列名
PNGDESC

「ノード状況」属性

説明
管理対象ノードの現在の操作状況。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。数値は、ウェアハウスと照会から返されます。定義されている値は INVALID(-2)、UNKNOWN(-1)、INACTIVE(0)、および ACTIVE(1) です。

名前

属性名
ノード状況

列名
PNGSTAT

「ノード・タイプ」属性

説明
管理対象ノードのタイプ。ノードが オンラインの場合は IP ノードです。オフラインの場合、タイプは「不明」です。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。数値は、ウェアハウスと照会から返されます。定義されている値は UNKNOWN(0) および IP NODE(1) です。

名前

属性名
ノード・タイプ

列名
PNGTYPE

状況のタイム・スタンプ

説明
そのノードが最後に検査された日時。

タイプ
時刻

名前

属性名
状況のタイム・スタンプ

列名
PNGTMSP

HTTP 属性グループ

「管理対象 URL」および「URL オブジェクト」という 2 つの HTTP 属性グループは、URL からの情報と、これらの URL 内のオブジェクトを受け取るために使用されます。

「管理対象 URL」テーブルおよび「URL オブジェクト」テーブルで使用される構文について詳しくは、1297 ページの『HTTP 属性に固有のフィールド』を参照してください。

管理対象 URL

「管理対象 URL」属性グループの各属性に関する情報を以下のリストにまとめます。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「**URL**」属性 **- この属性はキー属性です。**

説明
モニター対象の URL。

タイプ
ストリング

名前

属性名
URL

列名
HTTPURL

「**応答時間**」属性

説明
応答のダウンロードにかかった時間 (ミリ秒)。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会は数値を返します。定義されている値は TIMEOUT (-1) です。

名前

属性名
応答時間

列名
HTTPURL

「**ページ・サイズ**」属性

説明
HTTP 要求により返されるページのサイズ。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会は数値を返します。定義されている値は NO_RESPONSE_RECEIVED(-1) です。

名前

属性名
ページ・サイズ

列名
PAGESZ

「**ページ・オブジェクト**」属性

説明
モニター対象ページに関連付けられているオブジェクトの総数。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会は数値を返します。定義されている値は NOT_COLLECTED(-1) です。

名前

属性名
ページ・オブジェクト

列名
PGOBJS

「合計オブジェクト・サイズ」属性

説明
HTTP 要求により返されるページのサイズ。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会は数値を返します。定義されている値は NOT_COLLECTED(-1) です。

名前

属性名
合計オブジェクト・サイズ

列名
TOTOSZ

「ページ・タイトル」属性

説明
受信 URL ページのページ・タイトル。

タイプ
ストリング

名前

属性名
ページ・タイトル

列名
PAGETTL

「サーバー・タイプ」属性

説明
ターゲット URL Web サイトで使用されるサーバーのタイプ

タイプ
ストリング

名前

属性名
サーバー・タイプ

列名
SRVTYP

「応答コード」属性

説明
HTTP 要求の応答コード。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会は数値を返します。定義されている値は NO_RESPONSE_RECEIVED(-1) です。

名前

属性名
応答コード

列名
CODE

状況属性

説明
現在の管理対象 URL の状況 (「OK」 または状況の記述)。

タイプ
ストリング

名前

属性名
状況

列名
STATUS

「**URL 別名**」**属性**

説明
URL のユーザー指定別名。

タイプ
ストリング

名前

属性名
URL 別名

列名
ALIAS

ユーザー・データ属性

説明
URL で指定されているユーザー・データ。

タイプ
ストリング

名前

属性名
ユーザー・データ

列名
USER

URL オブジェクト

「URL オブジェクト」属性グループの各属性に関する情報を以下のリストにまとめます。

「**ノード**」**属性 - この属性はキー属性です**

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

「タイム・スタンプ」属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「URL」属性 - この属性はキー属性です。

説明
モニター対象の URL。

タイプ
ストリング

名前

属性名
URL

列名
HTTPURL

オブジェクト名属性

説明
ターゲット URL 内のページ・オブジェクトの名前。

タイプ
ストリング

名前

属性名
オブジェクト名

列名
ONAME

「オブジェクト・サイズ」属性

説明
ターゲット URL 内のページ・オブジェクトのサイズ (バイト単位)。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。数値は、ウェアハウスと照会から返されます。定義されている値は NOT_COLLECTED (-1)、OBJECT_NOT_FOUND (-2) です。その他の値では、数値が表示されます。

名前

属性名
オブジェクト・サイズ

列名
SIZE

「オブジェクト応答時間」属性

説明
オブジェクトのダウンロードにかかった時間 (ミリ秒)。

タイプ
列挙値を持つ整数。 Tivoli Enterprise Portal ではストリングが表示されます。数値は、ウェアハウスと照会から返されます。定義されている値は NOT_COLLECTED (-1)、NO_RESPONSE_RECEIVED (-2)、STATUS_CODE_ERROR (-3) です。その他の値では、数値が表示されます。

名前

属性名
オブジェクト応答時間

列名
ORTIME

ディスカバリー属性グループ

サブノード・タイプについて定義されたサブノード・インスタンスのセットを表す属性グループ

サブノード・タイプを作成すると、そのサブノード・タイプについて定義されたサブノード・インスタンスのセットを表す属性グループが作成されます。 これらの属性グループにはそれぞれ、同じ属性セットが含まれています。

次のリストには、「ディスカバリー」属性グループの各属性に関する情報を示します。太字テキストの名前は Tivoli Enterprise Portal でその属性がどのように表示されるかを示しています。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ行が作成され、Tivoli Enterprise Monitoring Server に送信されたとき (または履歴を残す目的で保管されたとき) に、エージェント・システムから収集した時間。エージェント・システムのローカル時間帯を示します。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

サブノード MSN 属性

説明
サブノード・エージェントの管理対象システム名。

タイプ
ストリング

名前

属性名
サブノード MSN

列名
SN_MSN

サブノード・アフィニティー属性

説明
サブノード・エージェントのアフィニティー。

タイプ
ストリング

名前

属性名
サブノード・アフィニティー

列名
SN_AFFIN

サブノード・タイプ属性

説明
このサブノードのノード・タイプ。

タイプ
ストリング

名前

属性名
サブノード・タイプ

列名
SN_TYPE

サブノード・リソース名属性

説明
サブノード・エージェントのリソース名。

タイプ
ストリング

名前

属性名
サブノード・リソース名

列名
SN_RES

サブノード・バージョン属性

説明
サブノード・エージェントのバージョン。

タイプ

名前

属性名
サブノード・バージョン

列名
SN_VER

「アクション実行の状況」属性グループ

「アクション実行の状況」属性グループには、エージェントが処理したアクションの状況が含まれています。

この属性グループはイベント・ベースであり、「アクション実行の状況」属性グループ内の、以下の各属性についての情報が含まれています。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名。

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ行が作成され、エージェントから Tivoli Enterprise Monitoring Server に送信されたときに、エージェント・システムから収集した時間。または、履歴を残す目的でデータ行が保存されたときの時間です。 エージェント・システムのローカル時間帯を示します。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「アクション名」属性

説明
実行されたアクションの名前

タイプ
ストリング

名前

属性名
アクション名

列名
TSKNAME

「**アクションの状況**」属性

説明
アクションの状況。

タイプ
列挙値を持つ整数。 値は OK (0)、NOT_APPLICABLE (1)、GENERAL_ERROR (2)、WARNING (3)、NOT_RUNNING (4)、DEPENDENT_NOT_RUNNING (5)、ALREADY_RUNNING (6)、PREREQ_NOT_RUNNING (7)、TIMED_OUT (8)、DOESNT_EXIST (9)、UNKNOWN (10)、DEPENDENT_STILL_RUNNING (11)、INSUFFICIENT_USER_AUTHORITY (12) です

名前

属性名
アクション状況

列名
TSKSTAT

「**アクション・アプリケーション戻りコード**」属性

説明
アクションによって開始されたアプリケーションの戻りコード。

タイプ
整数

名前

属性名
アクション・アプリケーション戻りコード

列名
TSKAPRC

「**アクション・メッセージ**」属性

説明
アクションの戻りコードに関連付けられたメッセージ。

タイプ
ストリング

名前

属性名
アクション・メッセージ

列名
TSKMSGE

「**アクション・インスタンス**」属性

説明
アクションの実行によって生成された出力に関連付けられた インスタンス。アクションがシステム・コマンドである場合、インスタンスは、 コマンドの出力の行番号です。

タイプ
ストリング

名前

属性名
アクション・インスタンス

列名
TSKINST

「**アクション結果**」属性

説明
アクションの実行によって生成された出力。

タイプ
ストリング

名前

属性名
アクション結果

列名
TSKOUTP

「**アクション・コマンド**」**属性**

説明
アクションによって実行されたコマンド。

タイプ
ストリング

名前

属性名
アクション・コマンド

列名
TSKCMND

「**アクション・ノード**」**属性**

説明
アクションが実行されたノード。

タイプ
ストリング

名前

属性名
アクション・ノード

列名
TSKORGN

「**アクション・サブノード**」**属性**

説明
アクションが実行されたサブノード。

タイプ
ストリング

名前

属性名
アクション・サブノード

列名
TSKSBND

「**アクション ID**」**属性**

説明
アクションの ID。

タイプ
整数

名前

属性名
アクション ID

列名
TSKID

「**アクション・タイプ**」属性

説明
アクションのタイプ。

タイプ
列挙値を持つ整数。Tivoli Enterprise Portal にはストリングが表示されます。ウェアハウスおよび照会では数値が返されます。定義されている値は UNKNOWN (0)、AUTOMATION (1) です。

名前

属性名
アクション・タイプ

列名
TSKTYPE

「**アクション所有者**」属性

説明
アクションを開始したシチュエーションまたはユーザーの名前。

タイプ
ストリング

名前

属性名
アクション所有者

列名
TSKOWNR

「ログ・ファイル状況」属性グループ

「ログ・ファイル状況」属性グループには、このエージェントがモニターしているログ・ファイルの状況を反映する情報が含まれています。

ログ属性グループが存在しており、エージェントが Tivoli Monitoring バージョン 6.2.1 (デフォルトの最小バージョン) 以降のエージェントである場合は、「ログ・ファイル状況」属性グループが組み込まれます。「ログ・ファイル状況」属性グループには、64 ビット数値として定義される 2 つの属性が含まれるため、大規模ファイルを処理できます。 64 ビット数値属性サポートは、Tivoli Monitoring バージョン 6.2.1 以降で提供されます。

「ログ・ファイル状況」属性グループの各属性についての情報を、以下のリストに示します。

「**ノード**」属性 **- この属性はキー属性です**

説明
エージェントの管理対象システム名。

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ行が作成され、エージェントから Tivoli Enterprise Monitoring Server に送信されたときに、エージェント・システムから収集した時間の値。または、履歴を残す目的でデータ行が保存されたときの時間です。エージェント・システムのローカル時間帯を示します。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「表の名前」属性 - この属性はキー属性です。

説明
このログがモニターされている表の名前

タイプ
ストリング

名前

属性名
表の名前

列名
TBLNAME

「ファイル名」属性 - この属性はキー属性です

説明
モニター対象のファイルの名前

タイプ
ストリング

名前

属性名
ファイル名

列名
FILNAME

「正規表現パターン」属性 - この属性はキー属性です

説明
このファイルをモニター対象として選択した正規表現パターン (存在する場合)

タイプ
ストリング

名前

属性名
正規表現パターン

列名
REPATRN

「ファイル・タイプ」属性

説明
このファイルのタイプ (標準ファイルまたはパイプ)

タイプ
列挙値を持つ整数。Tivoli Enterprise Portal ではストリングが表示されます。 定義されている値は UNKNOWN(0)、REGULAR FILE(1)、PIPE(2) です。

名前

属性名
ファイル・タイプ

列名
FILTYPE

「ファイル状況」属性

説明
モニター対象のファイルの状況

タイプ
列挙値を持つ整数。Tivoli Enterprise Portal ではストリングが表示されます。 定義されている値は OK(0)、PERMISSION DENIED(1)、FILE DOES NOT EXIST(2)、INTERRUPTED SYSTEM CALL(4)、I/O ERROR(5)、NO SUCH DEVICE(6)、 BAD FILE NUMBER(9)、OUT OF MEMORY(12)、ACCESS DENIED(13)、RESOURCE BUSY(16)、NOT A DIRECTORY(20)、IS A DIRECTORY(21)、INVALID ARGUMENT(22)、FILE TABLE OVERFLOW(23)、TOO MANY OPEN FILES(24)、TEXT FILE BUSY(26)、FILE TOO LARGE(27)、NO SPACE LEFT ON DEVICE(28)、ILLEGAL SEEK ON PIPE(29)、READ-ONLY FILE SYSTEM(30)、TOO MANY LINKS(31)、BROKEN PIPE(32) です。

名前

属性名
ファイル状況

列名
FILSTAT

「一致レコード数」属性

説明
このログで、指定されているパターンの 1 つに一致する処理レコードの数

タイプ
整数

名前

属性名
一致レコード数

列名
RECMTCH

「不一致レコード数」属性

説明
どのパターンにも一致せず、UnmatchLog に送られた処理レコードの数

タイプ
整数

名前

属性名
不一致レコード数

列名
RECUNMT

「処理レコード数」属性

説明
このログで、エージェント開始以降に処理されたレコードの数 (一致/イベント以外のレコードを含む)

タイプ
整数

名前

属性名
処理レコード数

列名
RECPROC

「現行ファイル位置」属性

説明
モニター対象ファイルの現行位置 (バイト)。この位置までのデータが処理されています。この位置より後のデータは処理されていません。パイプには適用されません。

タイプ
整数

名前

属性名
現行ファイル位置

列名
OFFSET

現行ファイル・サイズ属性

説明
モニター対象ファイルの現行サイズ。パイプには適用されません。

タイプ
整数

名前

属性名
現行ファイル・サイズ

列名
FILESIZE

「最終変更時刻」属性

説明
モニター対象ファイルの最終書き込み時刻。パイプには適用されません。

タイプ
タイム・スタンプ

名前

属性名
最終変更時刻

列名
LASTMOD

コード・ページ属性

説明
モニター対象ファイルの言語コード・ページ。

タイプ
ストリング

名前

属性名
コード・ページ

列名
CODEPG

「ログ・ファイル正規表現統計」属性グループ

「ログ・ファイル正規表現統計」属性グループには、ログ・ファイルの正規表現検索式の統計を示す情報が含まれます。

正規表現を使用することで、レコードのフィルタリングやレコードの定義を行うことができます。この属性グループは、両方のタイプに関する情報を示します。「結果タイプ」属性に INCLUDE または EXCLUDE が含まれる場合は、フィルターを使用してレコードがフィルタリングされます。「結果タイプ」属性に

BEGIN または END が含まれる場合は、フィルターを使用してレコードが定義されます。CPU 測定値は、オペレーティング・システムによって公開されるデータの細分度に基づく近似値です。 正規表現の評価にかかる時間が短い場合、これらの測定値は 0.00 になることがあります。 CPU 時間を使用して、正規表現の相対コストを判別し、特定の正規表現の動作を最適化します。

ログ属性グループが存在しており、エージェントが Tivoli Monitoring バージョン 6.2.1 以降のエージェントである場合は、「ログ・ファイル正規表現統計」属性グループが組み込まれます。Tivoli Monitoring の最小バージョンが「**エージェント情報**」ページで選択されています。詳しくは、1172 ページの『エージェントの命名および構成』 を参照してください。 「ログ・ファイル正規表現統計」属性グループには、64 ビット数値として定義される属性が含まれるため、長い期間を処理できます。 64 ビット数値属性のサポートは、Tivoli Monitoring バージョン 6.2.1 以降で提供されます。

「ログ・ファイル正規表現統計」属性グループの各属性についての情報を、以下のリストに示します。

「ノード」属性 - この属性はキー属性です

説明
エージェントの管理対象システム名。

タイプ
ストリング

名前

属性名
ノード

列名
ORIGINNODE

タイム・スタンプ属性

説明
データ収集時のエージェントの現地時間。

タイプ
時刻

名前

属性名
タイム・スタンプ

列名
TIMESTAMP

「表の名前」属性 - この属性はキー属性です。

説明
ログ・ファイル属性グループの名前。

タイプ
ストリング

名前

属性名
表の名前

列名
TBLNAME

「属性名」属性 - この属性はキー属性です

説明
このフィルターが適用される属性の名前。

タイプ
ストリング

名前

属性名
属性名

列名
ATRNAME

フィルター番号

説明
属性に使用されているフィルターのシーケンス 番号 (ゼロから開始)。

タイプ
整数 (数値プロパティー)

名前

属性名
Filter_Number

列名
FLTRNUM

「結果タイプ」属性

説明
フィルターが一致した場合に属性を受け入れる場合は結果タイプ INCLUDE、属性を拒否する場合は結果タイプ EXCLUDE を指定します。複数行レコードのレコードの先頭または末尾を指定するには、結果タイプ BEGIN または END を指定します。

タイプ
列挙値を持つ整数。Tivoli Enterprise Portal ではストリングが表示されます。 レコードのフィルタリングにフィルターを 使用する場合、定義されている値は INCLUDE(1) または EXCLUDE(2) です。レコードの定義にフィルターを 使用する場合、定義されている値は BEGIN(3) または END(4) です。

名前

属性名
結果タイプ

列名
RSTTYPE

「平均プロセッサー時間」属性

説明
この属性のフィルターを処理するために使用される平均プロセッサー時間 (秒)。 平均プロセッサー時間は、合計プロセッサー時間 (秒) をフィルター 回数で除算したものです。

タイプ
整数 (ゲージ)

名前

属性名
平均プロセッサー時間

列名
CPUTAVG

「プロセッサー時間」属性

説明
この属性のフィルターを処理するために使用される合計プロセッサー時間 (秒)。 プロセッサー時間は累積的であり、切り捨てられます。丸め処理は行われません。Linux `/proc/<pid>/task/thread/stat file` と同様です。

タイプ
整数 (カウンター)

名前

属性名
プロセッサー時間

列名
CPUTIME

「最大プロセッサー時間」属性

説明
単一のフィルター処理のために使用される最大プロセッサー時間 (秒)。フィルターが一度も使用されていない場合、または各フィルター処理の所要時間が 0.01 秒未満であった場合、最大値がゼロになることがあります。

タイプ
整数 (ゲージ)

名前

属性名
最大プロセッサー時間

列名
CPUTMAX

「最小プロセッサー時間」属性

説明
単一のフィルター処理のために使用される最小プロセッサー時間 (秒)。フィルター処理の所要時間が 0.01 秒未満であった場合、最小値がゼロになることがあります。

タイプ
整数 (ゲージ)

名前

属性名
最小プロセッサー時間

列名
CPUTMIN

「フィルター回数」属性

説明
フィルタリングの実行回数。合計プロセッサー時間とともに使用されて、平均プロセッサー時間が計算されます。

タイプ
整数 (カウンター)

名前

属性名
フィルター回数

列名
COUNT

「フィルター一致回数」属性

説明
フィルタリングが実行され、属性が一致した回数。

タイプ
整数 (カウンター)

名前

属性名
フィルター一致回数

列名
COUNTMA

「**フィルター不一致回数**」**属性**

説明
フィルタリングが実行され、属性が一致しなかった回数。

タイプ
整数 (カウンター)

名前

属性名
フィルター不一致回数

列名
COUNTUN

「**正規表現パターン**」**属性 - この属性はキー属性です**

説明
突き合わせに使用した正規表現。

タイプ
ストリング

名前

属性名
正規表現パターン

列名
REGXPAT

「**最終一致時刻**」**属性**

説明
フィルターが使用されて結果が一致した最後の時刻。

タイプ
時刻

名前

属性名
最終一致時刻

列名
LASTMAT

「**最終不一致時刻**」**属性**

説明
フィルターが使用されて結果が一致しなかった最後の時刻。

タイプ
時刻

名前

属性名
最終不一致時刻

列名
LASTUMA

既存エージェント用のアプリケーション・サポート拡張の作成

IBM Tivoli Monitoring 環境では、既存のエージェント用のアプリケーション・サポート拡張として、作成したカスタムのワークスペース、シチュエーション、照会、およびアクション実行コマンドを配布するためのインストール可能なパッケージを作成できます。

始める前に

カスタムのシチュエーション、ワークスペース、アクション実行コマンド、および照会の作成方法について詳しくは、1367 ページの『ワークスペース、アクション実行コマンド、およびシチュエーションの作成』 を参照してください。

このタスクについて

重要 : このタスクは、アプリケーション・サポートを、構築中のエージェントに追加する方法を示すものではありません。アプリケーション・サポートを、構築中のエージェントに追加するには、1403 ページの『アプリケーション・サポート・ファイルのインポート』を参照してください。

手順

1. Agent Builder から、「**ファイル**」 > 「**新規**」 > 「**その他**」と選択します。
2. 「**Agent Builder**」の「**Agent Builder アプリケーション・サポート拡張**」を選択します。
3. 「**次へ**」をクリックして、「**IBM Tivoli Monitoring アプリケーション・サポート拡張ウィザード**」のウェルカム・ページに進みます。
4. ウェルカム・ページの「**次へ**」をクリックします。
5. プロジェクトの名前を入力して、「**完了**」をクリックします。

アプリケーション・サポート拡張プロジェクトの作成

Agent Builder を使用して、アプリケーション・サポート拡張プロジェクトを作成します。

手順

1. Agent Builder から、「**ファイル**」 > 「**新規**」 > 「**その他**」と選択します。
2. 「**Agent Builder**」の「**Agent Builder アプリケーション・サポート拡張**」を選択します。
3. 「**次へ**」をクリックして、「**IBM Tivoli Monitoring アプリケーション・サポート拡張ウィザード**」のウェルカム・ページに進みます。
4. ウェルカム・ページの「**次へ**」をクリックします。
5. プロジェクトの名前を入力して、「**完了**」をクリックします。

プロジェクトへのサポート・ファイルの追加

アプリケーション・サポート拡張プロジェクトへのサポート・ファイルの追加

始める前に

アプリケーション・サポート拡張プロジェクトを作成します。詳しくは、1471 ページの『アプリケーション・サポート拡張プロジェクトの作成』を参照してください。

手順

1. アプリケーション・サポート拡張プロジェクトを右クリックして、「**IBM Tivoli**」 > 「**アプリケーション・サポート拡張のインポート**」を選択します。
2. 「**情報のインポート**」ウィンドウで、Tivoli Enterprise Portal Server があるホストの名前を選択するか、または「**追加**」をクリックしてホストを追加します。
3. 「**アプリケーション**」フィールドに、エージェント製品コードを入力します。
4. カスタム・アプリケーション・サポートを作成するエージェントのアフィニティーを 入力します。

エージェント・アフィニティーとは、ワークスペース、照会などの項目とエージェントを関連付ける Tivoli Monitoring 内部 ID です。 これは、Tivoli Monitoring インストール済み環境内で固有である必要があります。 「**参照**」をクリックして「**ノード・タイプ**」ウィンドウを開き、(この情報を入力するのではなく) リストからこの情報を選択します。

5. インポート情報を確認したら、「**完了**」をクリックします。
6. 「**シチュエーション**」ウィンドウの「使用可能なシチュエーション」リストで、インポートするシチュエーションを選択します。

   << をクリックして「選択したシチュエーション」リストにシチュエーションを追加し、「**OK**」をクリックします。プロジェクトの下に新規のフォルダーが作成され、ワークスペース、シチュエーション、および照会をインストールするために必要なファイルがそこに格納されます。
7. 「**照会**」ウィンドウで、「使用可能な照会」リストからインポートする照会を選択します。

   << をクリックして「選択した照会」リストに照会を追加し、「**OK**」をクリックします。
8. 「**アクション実行**」ウィンドウの「使用可能なアクション実行」リストで、インポートするアクション実行コマンドを選択します。

   << をクリックして「選択済みアクション実行」リストにアクション実行コマンドを追加し、「**OK**」をクリックします。 エージェントのサポート・ファイルが、プロジェクト内の適切なフォルダーに格納されます。

次のタスク

各種のエージェントに対して、この操作を必要なだけ繰り返し実行できます。Agent Builder は、アプリケーション・サポート拡張プロジェクト内のすべてのサポート・ファイルから、1 つのインストール・イメージを作成します。

アプリケーション・サポート拡張インストール・イメージの生成

アプリケーション・サポート拡張インストール・イメージを生成します。

手順

1. アプリケーション・サポート拡張プロジェクトを右クリックし、「**IBM Tivoli**」 > 「**アプリケーション・サポート拡張インストール・イメージの作成**」を選択します。
2. 「**アプリケーション・サポート拡張情報**」ウィンドウで、イメージが格納されるディレクトリーを入力します。
3. アプリケーション・サポート拡張には固有の製品コードが必要です。 新規エージェントの登録済み製品コードを入力します。Agent Builder で使用するために予約されている製品コードの 1 つを使用できます。許可される値は、`K00` から `K99`、`K{0-2}{A-Z}`、および `K{4-9}{A-Z}` です。

   注 : これらの値は内部使用のみを目的としたもので、共有または販売されるエージェントに対して使用することを目的としたものではありません。 他のユーザーと共有するエージェントを作成する場合は、製品コードを予約するために、toolkit@us.ibm.com にメモを送信する必要があります。製品コードの要求には、作成するエージェントの説明を組み込まなければなりません。 製品コードが割り当てられ、登録された上で、送り返されてきます。 3 文字の製品コードを受け取る際に、Agent Builder が割り当てられている製品コードを使用できるようにする手順が通知されます。
4. アプリケーション・サポート拡張の名前を入力します。
5. アプリケーション・サポート拡張の説明を入力します。
6. アプリケーション・サポート拡張のバージョンを VVRRMMFF 形式で入力します。ここで、`vv` はバージョン番号、`rr` はリリース番号、`mm` はモディフィケーション 番号 (フィックスパック番号)、`ff` は暫定修正番号です。
7. 「**完了**」をクリックします。

アプリケーション・サポート拡張のインストール

アプリケーション・サポート拡張のインストール

手順

1. イメージを Tivoli Enterprise Monitoring Server および Tivoli Enterprise Portal Server に転送します。
2. Tivoli Enterprise Monitoring Server サポートをインストールするには、以下のコマンドのいずれかを実行します。
   - Windows の場合: `install*KXX*TEMSSupport.bat`
   - UNIX の場合: `install*KXX*TEMSSupport.sh`

   コマンドの形式は次のとおりです。

   ```
   installKXXTEMSSupport[.bat | .sh] <ITM Install Directory> [-s tems_host]
    [-u tems_user]  ¥[-p tems_password]
   ```
3. Tivoli Enterprise Portal Server サポートをインストールするには、以下のコマンドのいずれかを実行します。
   - Windows の場合: `install*KXX*TEPSSupport.bat`
   - UNIX の場合: `install*KXX*TEPSSupport.sh`

   コマンドの形式は次のとおりです。

   ```
   installKXXTEPSSupport[.bat | .sh] <ITM Install Directory> [-r]
   ```

   `-r` は、インストール後に Tivoli Enterprise Portal Server を再始動することを指定します。

「ソリューション・インストール・プロジェクト」のアプリケーション・サポート拡張プロジェクトへの変換

既存の「**ソリューション・インストール・プロジェクト**」のアプリケーション・サポート拡張プロジェクトへの変換

このタスクについて

既存の**ソリューション・インストール・プロジェクト**をアプリケーション・サポート拡張プロジェクトに変換するには、以下の手順を実行します。

注 : ソリューション・インストール・プロジェクトでは、 サポート・ファイルのみがマイグレーションされます。

手順

1. **ソリューション・インストール・プロジェクト**を右クリックして、「**IBM Tivoli**」 > 「**ソリューション・インストール・プロジェクトの変換**」を選択します。
2. 新規アプリケーション・サポート拡張プロジェクトの名前を入力するか、リストから既存のプロジェクトを選択します。
3. 「**完了**」をクリックします。

Cognos データ・モデルの生成

Agent Builder は、エージェントごとに Cognos データ・モデルを生成できます。データ・モデルを使用して、エージェント情報を Cognos Framework Manager にインポートしてレポートを作成します。

この Cognos データ・モデルは、Framework Manager で開いて表示でき、Tivoli Common Reporting に公開するモデル・パッケージを作成できます。また、データ・モデルは、公開前に Framework Manager 内でカスタマイズまたは変更することもできます。

レポートの作成後、Agent Builder により、最終レポート・パッケージを Agent Builder プロジェクトにインポートすることも可能です。この機能より、エージェント・パッケージの一部としてレポートが既に含まれた状態で、以降のエージェント・プロジェクトを生成できます。エージェント・インストール・イメージの一部としてパッケージされたレポートは、実稼働環境の Tivoli Common Reporting にインポートできます。

注 : 本書では、以下の表記規則に注意してください。

- K*xx* または k*xx* は、エージェントに対して指定された製品コード (k99 など) を表します。
- *dbType* は、Tivoli Data Warehouse が使用するデータベース (DB2 など) を表します。

Cognos データ・モデルを生成するための前提条件

Cognos データ・モデルを生成する前に、ここに示すタスクを実行してください。

このタスクについて

注 :

- Agent Builder で今後生成されるすべてのデータ・モデルはこの環境を使用するため、これらのステップを実行する必要があるのは一度のみです。
- エージェントのテストおよびレポートの作成用に、別個の開発環境を構築することをお勧めします。

手順

1. 1474 ページの『Tivoli Data Warehouse』 をインストールおよび構成します。
2. Tivoli Data Warehouse に表およびプロシージャーを作成します。
   a) 1474 ページの『Tivoli Data Warehouse 内での表およびプロシージャーの作成』.
   b) 1477 ページの『Tivoli Data Warehouse への TivoliReporting and Analytics Model の取り込み』.
3. 1477 ページの『Tivoli Common Reporting』 をインストールおよび構成します。
4. 1478 ページの『Framework Manager』 をインストールおよび構成します。

Tivoli Data Warehouse

Tivoli Data Warehouse について。

レポートを作成するには、環境に Tivoli Data Warehouse、ウェアハウス・プロキシー・エージェント、要約およびプルーニング・エージェントがインストールされ、構成されている必要があります。詳しくは、「*IBM Tivoli Monitoring* インストールおよび設定ガイド」を参照してください。

Tivoli Data Warehouse 内での表およびプロシージャーの作成

Tivoli Data Warehouse での ManagedSystem 表およびストアード・プロシージャーの作成または変更

このタスクについて

生成された Cognos データ・モデルには、ManagedSystem ディメンションを定義するために使用される ManagedSystem 表が含まれています。 ManagedSystem ディメンションを使用することにより、管理対象システムを相互に関連付けることができるレポートを作成できます。 例えば、エージェントがサブノード・エージェントである場合、ディメンションを使用して、特定のエージェント・インスタンス用に存在するサブノードを判別することができます。

ManagedSystem 表は Tivoli Data Warehouse によって作成されません。そのため、エージェントが Agent Builder で生成されると、以下を行う SQL スクリプトがデータベース・プラットフォームごとに生成されます。

- ManagedSystem 表を作成する。Tivoli Data Warehouse に表が存在しない場合は、このスクリプトを使用します。
- ManagedSystem 表を編集する。Tivoli Data Warehouse に表が存在する場合は、このスクリプトを使用します。他のレポート作成製品でも ManagedSystem 表を作成できますが、必要な列がいくつか欠落した表が作成されます。

- Tivoli Data Warehouse の表から ManagedSystem 表にデータを取り込むストアード・プロシージャーを作成します。

以下に示すスクリプトは 1 回だけ実行します。

Tivoli Data Warehouse 内で表およびプロシージャーを作成するための *DB2* スクリプトの実行
DB2 データベースの場合、以下のスクリプトを使用して、Tivoli Data Warehouse 内に表を作成します。

始める前に

DB2 用のスクリプトは、以下のディレクトリーに格納されています。

```
reports/db2/Kxx/reports/cognos_reports/itmkxx/db_scripts
```

手順

1. 生成されたスクリプト (create_table.sql、alter_table.sql、および create_procedure.sql) はすべて、Tivoli Data Warehouse ユーザー ID として *itmuser* を使用します。*itmuser* が、ご使用の環境内での Tivoli Data Warehouse ユーザー ID ではない場合、出現するすべての *itmuser* を正しいユーザー ID に変更します。
2. Tivoli Data Warehouse ユーザーとして Tivoli Data Warehouse に接続します。

   ```
   db2 connect to <Tivoli Data Warehouse alias name> user
   <Tivoli Data Warehouse user id> using  <password>
   ```

3. ManagedSystem 表が存在するかどうかを判別します。

   ```
   db2 "select count(*) from sysibm.systables where name = 'MANAGEDSYSTEM'
   and creator=upper ('<Tivoli Data Warehouse user id>')"
   ```

4. 表を作成または変更します。
   - 照会が 1 を返す場合、表は存在します。以下のように、変更スクリプトを実行します。

     ```
     db2 -tvf alter_table.sql
     ```

   - 照会が 0 を返す場合、表は存在しません。 以下のように、作成スクリプトを実行します。

     ```
     db2 -tvf create_table.sql
     ```

5. 以下のスクリプトを実行してストアード・プロシージャーを作成します。

   ```
   db2 -td@ -f create_procedure.sql
   ```

Tivoli Data Warehouse 内で表およびプロシージャーを作成するための *Oracle* スクリプトの実行
Oracle データベースの場合、以下のスクリプトを使用して、Tivoli Data Warehouse 内に表を作成します。

始める前に

Oracle 用のスクリプトは、以下のディレクトリーに格納されています。

```
reports/oracle/Kxx/reports/cognos_reports/itmkxx/db_scripts
```

手順

1. 生成されたスクリプト (create_table.sql、alter_table.sql、および create_procedure.sql) はすべて、Tivoli Data Warehouse ユーザー ID として *itmuser* を使用します。*itmuser* が、ご使用の環境内での Tivoli Data Warehouse ユーザー ID ではない場合、出現するすべての *itmuser* を正しいユーザー ID に変更します。
2. sqlplus を開始します。

```
sqlplus <IBM Tivoli Monitoring user ID>/<password>@
<Tivoli Data Warehouse SID>
```

3. ManagedSystem 表が存在するかどうかを判別します。

```
select count(*) from user_tables where table_name = 'MANAGEDSYSTEM';
```

4. 表を作成または変更します。
   - 照会が 1 を返す場合、表は存在します。以下のように、変更スクリプトを実行します。

     ```
     @<path to alter_table.sql>;
     ```

   - 照会が 0 を返す場合、表は存在しません。 以下のように、作成スクリプトを実行します。

     ```
     @<path to create_table.sql>;
     ```

5. 以下のスクリプトを実行してストアード・プロシージャーを作成します。

```
@<path to create_procedure.sql>;
```

Tivoli Data Warehouse 内で表およびプロシージャーを作成するための *SQL Server 2005* スクリプトと *SQL Server 2008* スクリプトの実行

始める前に

SQL Server 用のスクリプトは、以下のディレクトリーに格納されています。

```
reports/mssql/Kxx/reports/cognos_reports/itmkxx/db_scripts
```

手順

1. 生成されたスクリプト (create_table.sql、alter_table.sql、および create_procedure.sql) はすべて、Tivoli Data Warehouse ユーザー ID として *itmuser* を使用します。*itmuser* が、ご使用の環境内での Tivoli Data Warehouse ユーザー ID ではない場合、出現するすべての *itmuser* を正しいユーザー ID に変更します。
2. ManagedSystem 表が存在するかどうかを判別します。

```
osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
<Tivoli Data Warehouse database name> -Q "Select count(*)
from INFORMATION_SCHEMA.TABLES where table_name = 'ManagedSystem'"
```

3. 表を作成または変更します。
   - 照会が 1 を返す場合、表は存在します。以下のように、変更スクリプトを実行します。

     ```
     osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
     <Tivoli Data Warehouse database name> -I -n -i <path to alter_table.sql>
     ```

   - 照会が 0 を返す場合、表は存在しません。 以下のように、作成スクリプトを実行します。

     ```
     osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
     <Tivoli Data Warehouse database name> -I -n -i <path to create_table.sql>
     ```

4. 以下のスクリプトを実行してストアード・プロシージャーを作成します。

```
osql -S <Server> -U <Tivoli Data Warehouse user ID> -P
<password> -d <Tivoli Data Warehouse database name>
-I -n -i <path to create_procedure.sql>
```

Tivoli Data Warehouse への *TivoliReporting and Analytics Model* の取り込み

提供されているデータベース・スクリプトを使用して、Tivoli Data Warehouse にデータを取り込みます。

このタスクについて

Tivoli Reporting and Analytics Model (TRAM) には、すべてのレポート・パッケージに共通する基本ナレッジ・セットが含まれています。TRAM は、各データベースに固有の一連のスクリプトを使用してインストールされます。サポートされる各データベースへのデータの取り込みに必要なスクリプトは、エージェント・インストール・イメージのレポート・ディレクトリー内に含まれています。 以下の手順を実行して、Tivoli Data Warehouse で Tivoli Reporting and Analytics Model 共通ディメンションを作成します。

手順

1. Tivoli Reporting and Analytics Model データベース・スクリプトを参照します。
2. エージェント・パッケージを解凍します。
   - Windows システムの場合、エージェント・パッケージは k*xx*.zip です。
   - Linux および UNIX システムの場合、エージェント・パッケージは k*xx*.tgz です。
3. 適切なデータベース・スクリプトに移動します。
   - DB2 スクリプトは、エージェント・パッケージの以下の場所に格納されています。

     ```
     reports/db2/K xx /reports/cognos_reports/itmk xx /db_scripts
     ```

   - Oracle スクリプトは、エージェント・パッケージの以下の場所に格納されています。

     ```
     reports/oracle/K xx /reports/cognos_reports/itmk xx /db_scripts
     ```

   - Microsoft SQL Server スクリプトは、エージェント・パッケージの以下の場所に格納されています。

     ```
     reports/mssql/K xx /reports/cognos_reports/itmk xx /db_scripts
     ```

4. データベース・スクリプトを実行して、Tivoli Data Warehouse 内で共通ディメンションを生成します。各スクリプト・セットには、使用方法が記載された README ファイルが用意されています。
5. スクリプトによって以下の表が Tivoli Data Warehouse に追加されたことを確認します。

   ```
   "Computer System", WEEKDAY_LOOKUP, MONTH_LOOKUP, TIMEZONE_DIMENSION, TIME_DIMENSION
   ```

Tivoli Common Reporting

Tivoli Common Reporting には、Cognos Business Intelligence エンジンが組み込まれており、エージェント・レポートの作成を支援する要素が含まれています。

Tivoli Common Reporting は、Tivoli Data Warehouse に接続しているデータ・ソースを使用して、インストールと構成を行う必要があります。

Tivoli Common Reporting のインストール

Tivoli Common Reporting をインストールする必要があります。サポートされているのは、バージョン 1.3、2.1、2.1.1 以降です。Tivoli Common Reporting のインストールについては、『Tivoli Common Reporting のインストール』を参照してください。

Tivoli Common Reporting の構成

Tivoli Common Reporting を構成する必要があります。Tivoli Common Reporting の構成については、『IBM Tivoli Common Reporting の構成』を参照してください。

Tivoli Data Warehouse と Tivoli Common Reporting の間のデータ・ソースを作成します。詳しくは、『データベース接続の構成』を参照してください。適切なデータベース・タイプをクリックします。データ・ソースに対して指定されている名前をメモします。デフォルトは **TDW** です。

注 : データ・ソース名は、「**Cognos 情報**」ページの「**データ・ソース**」フィールド内の名前と一致している必要があります。 「**Cognos 情報**」ページについて詳しくは、1188 ページの『Cognos 情報』を参照してください。

Framework Manager
Framework Manager は、Tivoli Common Reporting アプリケーションとともに提供されるアプリケーションですが、個別にインストールおよび構成する必要があります。

Framework Manager は、データ・モデルの表示および変更、および Tivoli Common Reporting へのデータ・モデルの公開に使用されます。

Framework Manager のインストール

Framework Manager をインストールする必要があります。サポートされているのは、バージョン 8.4、8.4.1 以降です。

Framework Manager は Tivoli Common Reporting とともに提供されますが、手動でインストールする必要があります。Tivoli Common Reporting 1.3 には、Framework Manager 8.4 が同梱されています。Tivoli Common Reporting 2.1 および 2.1.1 には、Framework Manager 8.4.1 が同梱されています。Framework Manager のインストールについては、「*Tivoli Common Reporting* ユーザーズ・ガイド」の『Framework Manager のインストール』を参照してください。

Framework Manager の構成

Framework Manager を構成する必要があります。Framework Manager の構成については、「*Tivoli Common Reporting* ユーザーズ・ガイド」の『Framework Manager の構成』を参照してください。

レポートの作成

エージェント・モデルを公開するには Framework Manager を使用します。レポートの作成を開始するには Report Studio を使用します。

始める前に

エージェントは、完了後に Tivoli Monitoring 環境にインストールする必要があります。 また、エージェントのヒストリカル収集が構成されている必要があり、さらにエージェントが少なくとも 1 つのウェアハウス・アップロード間隔で実行される必要があります。要約が構成されている必要があり、Tivoli Monitoring での要約設定の選択内容が、Agent Builder での要約の選択内容と同じであることが必要です。 エージェントのデータがウェアハウスにアップロードされた 後に、要約およびプルーニング・エージェントが少なくとも 1 回実行される必要があります。

1. エージェントをインストール、構成、および開始します。
2. レポートの作成対象となる各属性グループについて、ヒストリカル収集を作成し、エージェントに配布します。

   注 : ウェアハウス・アップロード間隔のデフォルトは「毎日」です。ただし、この間隔はより短くすることができます。

   ヒストリカル収集の構成については、「*IBM Tivoli Monitoring* 管理者ガイド」の『ヒストリカル・データの管理』を参照してください。
3. Tivoli Monitoring において、ステップ 2 でヒストリカル収集を作成したすべての属性グループについて要約を構成します。

   注 : ヒストリカル収集および要約を構成するときには、要約表でのデータ処理が完了するまで待機する必要があります。

   注 : デフォルトでは、要約およびプルーニング・エージェントは、1 日 1 回、午前 2 時に実行するよう構成されますが、この設定は変更できます。例えば、1 時間ごとに実行するよう構成できます。 Tivoli Data Warehouse の構成について詳しくは、「*IBM Tivoli Monitoring* インストールおよび設定ガイド」の『データウェアハウスのセットアップ』を参照してください。

このタスクについて

Agent Builder でエージェントを生成すると、Framework Manager プロジェクト全体が作成されます。これには、データ・モデルおよび Framework Manager プロジェクト・ファイルが含まれます。 Framework Manager は、プロジェクト・ファイルを直接開くことができます。これによって、変更、カスタマイズ、または公開を行うためにデータ・モデルが開かれます。

手順

注 : エージェントについて生成されるデータ・モデルには、各属性グループについてのすべての 要約時間ディメンション (毎時、毎日、毎週、毎月、毎四半期、および毎年) が含まれます。これらのディメンションがエージェントの Tivoli Data Warehouse に存在するのは、そのエージェントに対して要約およびプルーニングが構成されている場合のみです。 さらに、ディメンションが選択されている場合、および要約およびプルーニング・エージェントによって表が作成され、表にデータが取り込まれている場合という条件もあります。存在しないディメンションを使用するレポートを定義して、Tivoli Common Reporting に公開することはできます。それらのレポートは、要約およびプルーニング・エージェントによって要約表が作成されるまで機能しません。

1. Framework Manager でエージェント・データ・モデルを開きます。
   a) Framework Manager を開きます。
   b)「**ようこそ**」ページで、「**プロジェクトを開く**」をクリックします。

      ヒント : Framework Manager 内にいる場合は、「**ファイル**」メニューから「**開く**」をクリックします。
   c) エージェント・データ・モデルを参照します。
      - DB2 の場合:

        ```
        reports/db2/K<i>xx</i>/model/
        ```
      - Oracle の場合:

        ```
        reports/oracle/K<i>xx</i>/model/
        ```
      - Microsoft SQL Server の場合:

        ```
        reports/mssql/K<i>xx</i>/model/
        ```
   d) エージェント・プロジェクト・ファイル K*xx*.cpf を選択します。

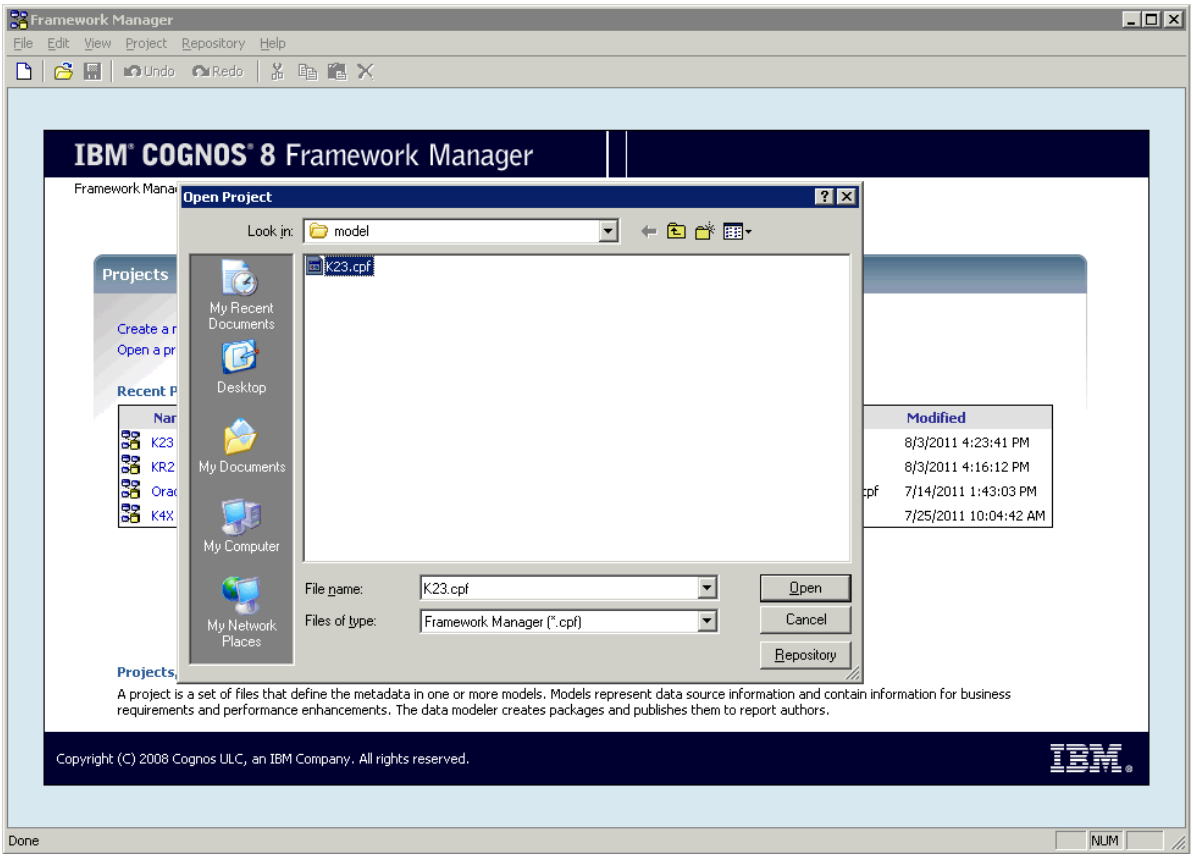


図 *80.* エージェント・プロジェクト・ファイルの選択

注 : Framework Manager でエージェント・プロジェクトを開くと、「最近のプロジェクト」の下にエージェント名がリストされます。

2. 管理対象システムの表にデータを取り込みます。詳しくは、1484 ページの『ManagedSystem 表のデータ設定』を参照してください。
3. Framework Manager を使用して、エージェント・モデルを Tivoli Common Reporting に公開します。
   a) Framework Manager を開きます。
   b) エージェント・プロジェクトを開きます。
   c) ナビゲーション・ツリーで「**パッケージ**」を展開します。
   d) エージェント・パッケージを右クリックし、「**パッケージの公開**」を選択します。

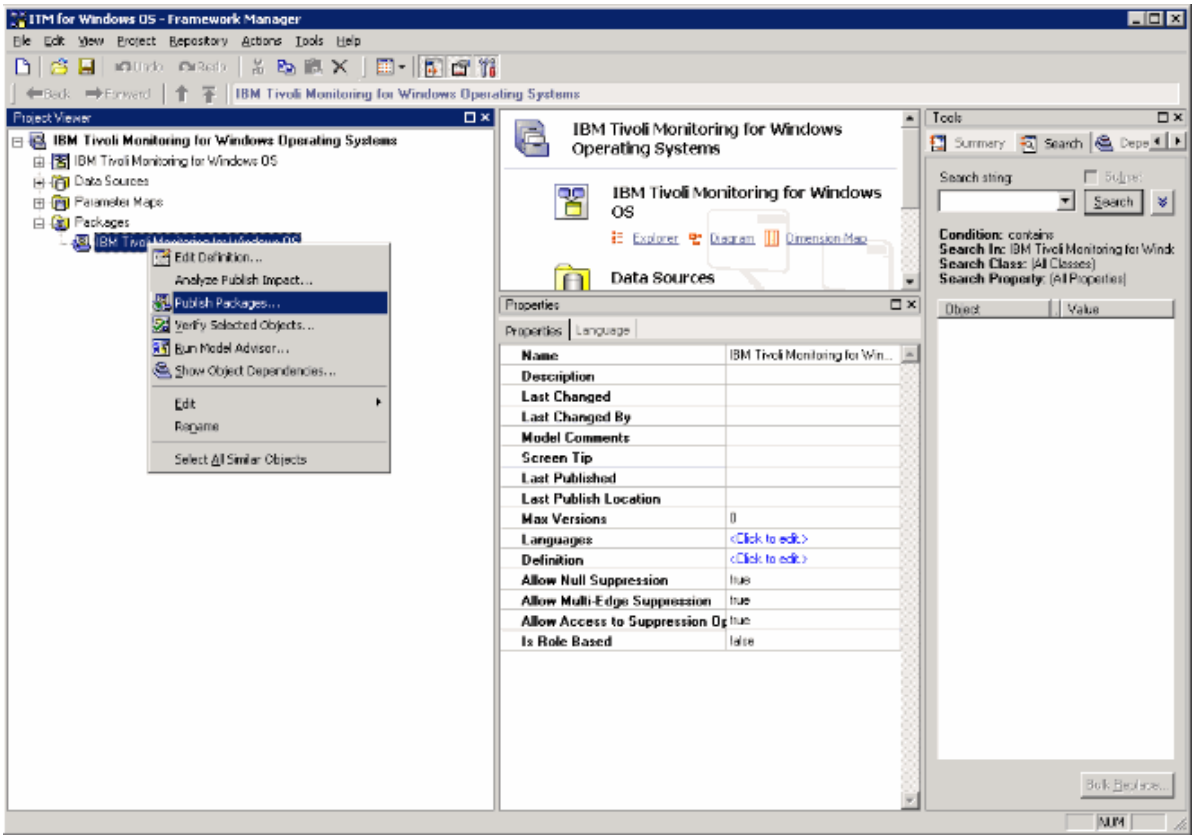


図 *81.* 「パッケージの公開」の選択

4. Report Studio を使用して、新規のレポートまたはテンプレートを作成します。
   a) Tivoli Common Reporting にログオンします。
   b)「パブリック・フォルダー」を参照し、ナビゲーション・パネルで「**レポート**」を展開して、「**Common Reporting**」を選択します。

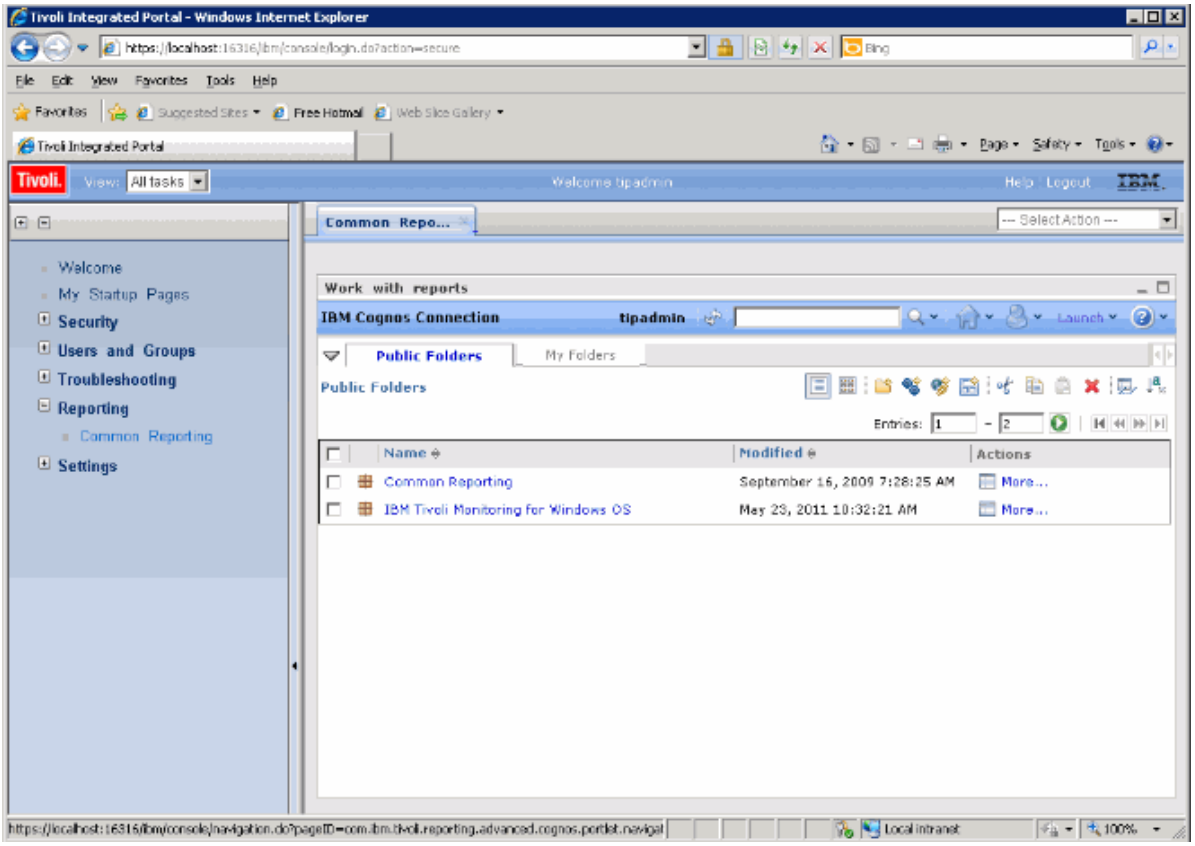


図 *82. Common Reporting* の選択

c) 示されているリストから、ご使用の Tivoli Monitoring エージェントを選択します。

d) 起動メニューをクリックして「**Report Studio**」または「**Query Studio**」を選択し、レポート作成ツールを開きます。

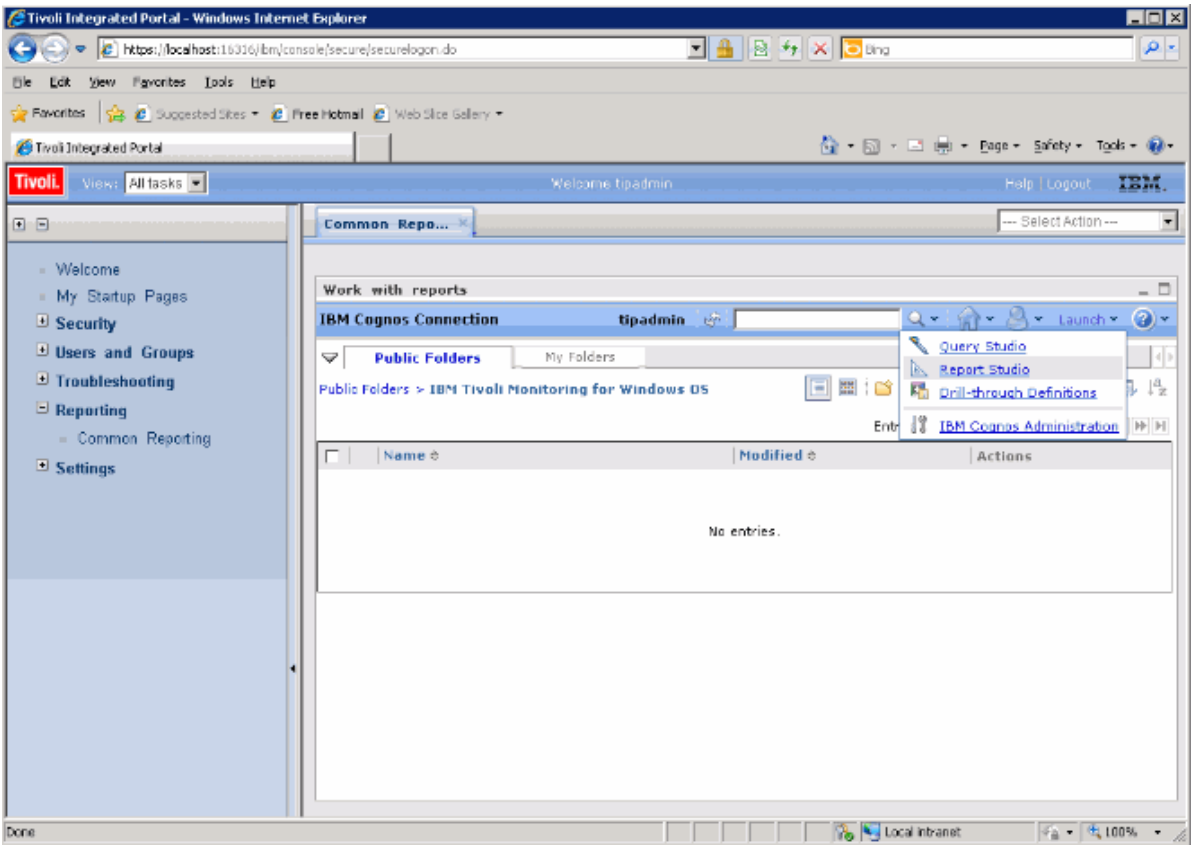


図 *83. Report Studio* の選択

次のタスク

Report Studio を使用することで、新規のレポートまたはテンプレートを作成したり、既存のレポートまたはテンプレートを変更したりすることができます。

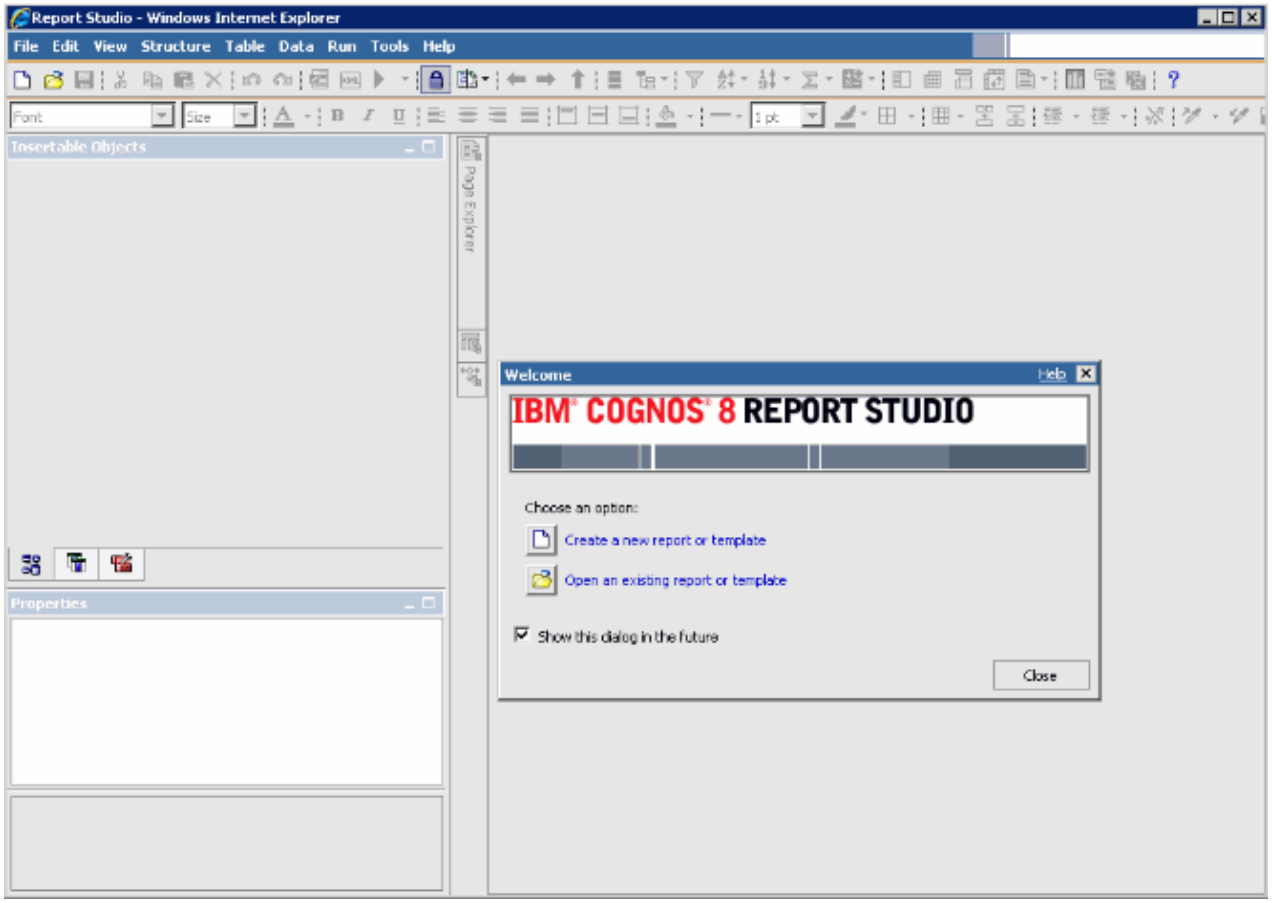


図 *84. Report Studio*

詳しくは、IBM Knowledge Center の Tivoli Common Reporting トピック・コレクションを参照してください。

ManagedSystem 表のデータ設定

ManagedSystem 表へのデータの取り込みは、kqz_populate_msn ストアード・プロシージャーを使用して行われます。

詳しくは、1486 ページの『DB2 ストアード・プロシージャーの実行』を参照してください。このプロシージャーは、ManagedSystem 表に管理対象システム名の最新のリストが含まれるように、定期的に実行する必要があります。

ストアード・プロシージャーは、Tivoli Data Warehouse 内の以下のヒストリカル表を読み取ります (存在する場合)。

- エージェントのパフォーマンス・オブジェクト状況表
- エージェントの可用性表。プロセスまたはサービスをモニターするエージェントには、可用性表があります。
- エージェントのディスカバリー表。サブノード・エージェントはディスカバリー表を作成します。

ヒストリカル収集は、特定のセットの属性グループで開始する必要があります。 これらの属性グループに対してヒストリカル収集を作成および開始する一連のスクリプトが生成されます。 スクリプトを使用しない場合は、属性グループのリストがスクリプトのコメント・ヘッダー・ブロックに表示されます。

ヒストリカル収集を使用可能にする必要のある表を示す、以下のサンプル・スクリプトが作成されます。

- reports/configuretdw.sh
- reports/configuretdw.bat

以下の表で、必須引数について説明します。

注: -n または -m のいずれかを指定する必要がありますが、両方を指定することはできません。

表 *301.* 必須引数

引数	説明
-h *candle_home*	Tivoli Monitoring のインストール・パス。
-u *teps_user*	ヒストリカル収集を作成する際のログイン・ユーザーである Tivoli Enterprise Portal Server ユーザー。
-n *tems_name*	収集を開始する Tivoli Enterprise Monitoring Server。スペース区切りのリストを使用して、複数の Tivoli Enterprise Monitoring Server を指定することができます。複数の Tivoli Enterprise Monitoring Server を指定する場合、リストを引用符で囲んでください。例えば、-n “tems1 tems2” のようになります。
-m *managed_system_group_or_managed_system*	収集の開始対象となる管理対象システム・グループまたは管理対象システム名。 スペース区切りのリストを使用して、複数の管理対象システム・グループまたは管理対象システムを指定できます。 複数の管理対象システム・グループまたは管理対象システムを指定する場合、リストを引用符で囲みます。例えば、-m “msg1 msg2” のようになります。

以下の表で、オプションの引数について説明します。

表 *302.* オプションの引数

引数	説明
-s *teps_host*	Tivoli Enterprise Portal Server のホスト名または IP アドレス。 指定しない場合のデフォルトは localhost です。
-p *teps_password*	-u オプションで指定した Tivoli Enterprise Portal Server ユーザーのパスワード。指定しない場合、スクリプトによってパスワードを求めるプロンプトが出されます。
-c *historical_collection_interval*	ヒストリカル収集を開始するときに使用するヒストリカル収集間隔。指定しない場合、デフォルト値は 1h (1 時間) です。有効な値は、15m、30m、1h、12h、および 1d です。ここで、m は分、h は時間、d は日を表します。
-r *pruning_interval*	ヒストリカル・データに使用するプルーニング間隔。表のサイズが大きくなり続けないように、ヒストリカル・データはプルーニングする必要があります。指定しない場合、デフォルトは 2d (2 日) です。 日には d、月には m、年には y を使用します。

ヒストリカル収集が開始された後、kqz_populate_msn ストアード・プロシージャーを定期的に実行する必要があります。このストアード・プロシージャーを定期的に実行することで、ManagedSystem 表に Tivoli Monitoring 環境にある管理対象システムの最新のリストが格納されます。

DB2 ストアード・プロシージャーの実行

DB2 でストアード・プロシージャーを実行します。

このタスクについて

以下のステップを実行して、DB2 でストアード・プロシージャーを実行します。

手順

1. ウェアハウス・ユーザーとして Tivoli Data Warehouse データベースに接続します。

```
connect to <Tivoli Data Warehouse database alias> user
<Tivoli Data Warehouse user id> using <password>
```

2. 以下のストアード・プロシージャーを実行します。

```
db2 "call <Tivoli Data Warehouse schema>.kqz_populate_msn
('<three letter product code for the agent>')"
```

Oracle ストアード・プロシージャーの実行

Oracle でストアード・プロシージャーを実行します。

このタスクについて

以下のステップを実行して、Oracle でストアード・プロシージャーを実行します。

手順

1. sqlplus を開始します。

```
sqlplus <Tivoli Data Warehouse user id>/<password>@
<Oracle SID>
```

2. 以下のストアード・プロシージャーを実行します。

```
execute kqz_populate_msn('<three letter product code for the agent>');
```

SQL Server 2005 および *2008* でのストアード・プロシージャーの実行

SQL Server でストアード・プロシージャーを実行します。

このタスクについて

以下のステップを実行して、SQL Server 2005 および 2008 でストアード・プロシージャーを実行します。

手順

以下のストアード・プロシージャーを実行します。

```
osql -S <server> -U <Tivoli Data Warehouse id> -P
<Tivoli Data Warehouse password> -d
<Tivoli Data Warehouse database name> -Q "EXEC
[<Tivoli Data Warehouse schema>].[kqz_populate_msn]
@pv_productcode = N'<three letter product code>'"
```

Tivoli Common Reporting からのレポートおよびデータ・モデルのエクスポート

Tivoli Common Reporting からのレポートおよびデータ・モデルのエクスポート。

手順

1. Tivoli Common Reporting にログインします。
2. 「パブリック・フォルダー」に移動し、ナビゲーション・パネルの「**レポート**」で「**Common Reporting**」を選択します。

3. 「作業」レポート・セクションで、**起動**メニューをクリックして「**IBM Cognos 管理**」を選択します。
4. 「**構成**」タブをクリックします。
5. 「**コンテンツ管理**」をクリックします。

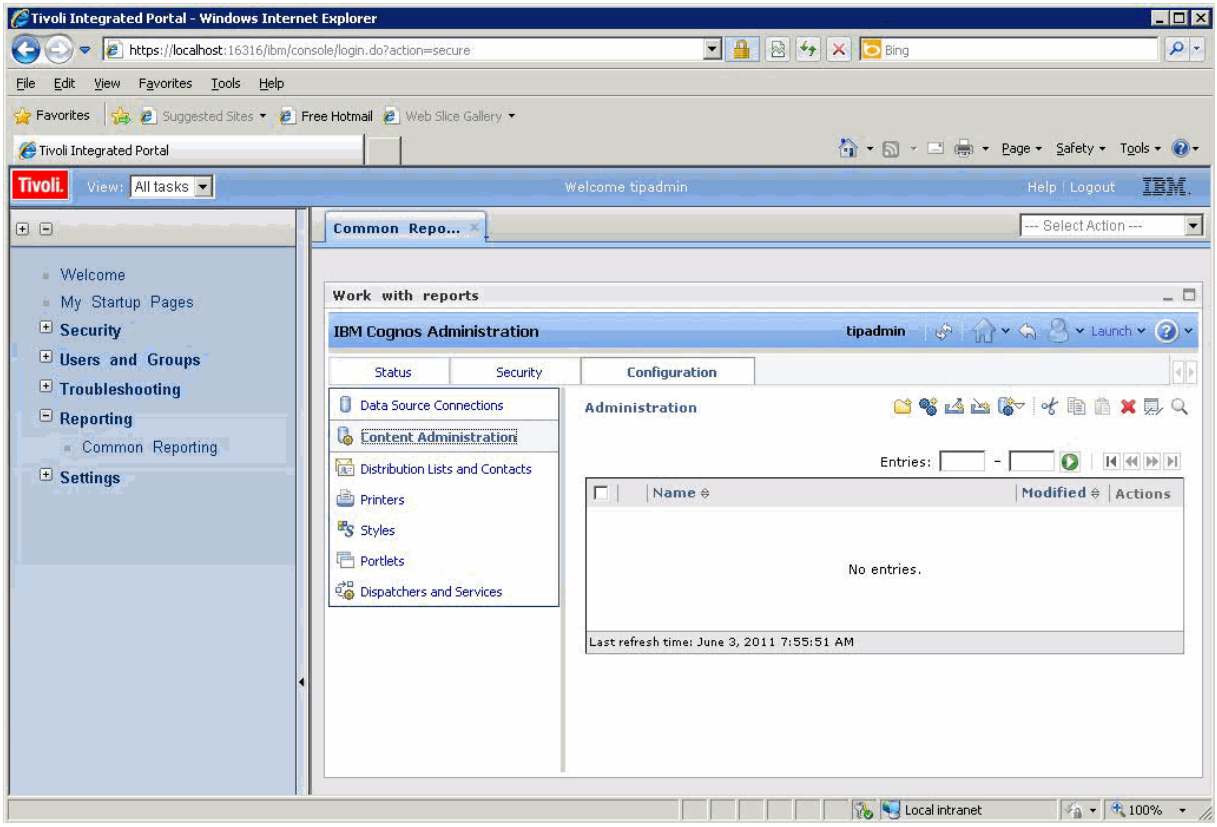


図 *85.* 「コンテンツ管理」タブ

6. 「**新規エクスポート**」アイコンをクリックして、新規パッケージをエクスポートします。
7. パッケージに名前を付けます。オプションとして、画面ヒントと説明を追加できます。
8. 「**共有フォルダーとディレクトリーの内容を選択**」を選択します。
9. 「パブリック・フォルダー」ダイアログで、「**追加**」リンクをクリックします。
10. エージェント・パッケージを「**選択されたエントリー**」に移動します。
11. ウィザードの最終ページで、「**保存のみ**」を選択します。ウィザードが完了すると、レポート・パッケージは「コンテンツ管理」タブにリストされます。
12. 「コンテンツ管理」タブで、緑色の矢印 (実行) をクリックして、`.zip` ファイルを作成します。

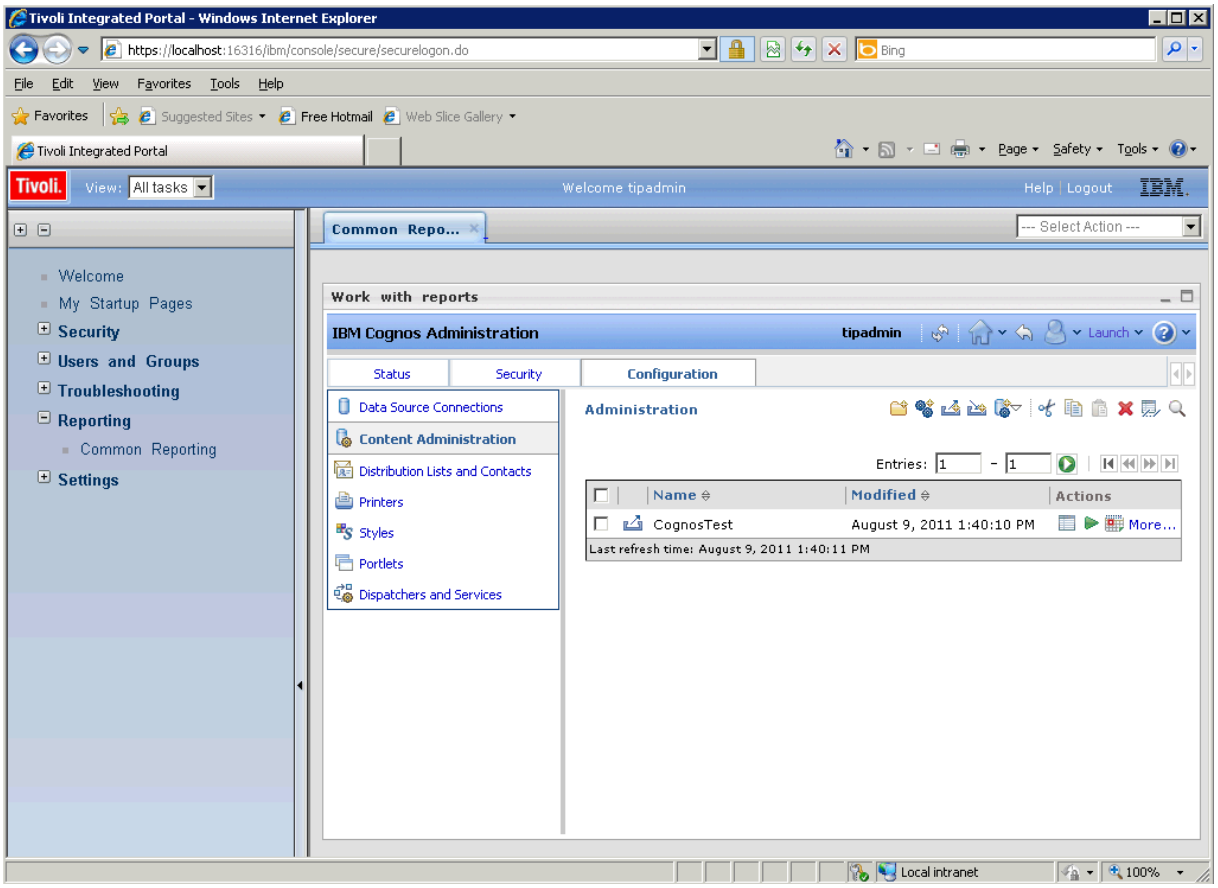


図 *86.* エージェント・パッケージがリストされた「コンテンツ管理」タブ

タスクの結果

エクスポート・プロセスで作成された `.zip` 圧縮ファイルがデプロイメント・ディレクトリーに配置されます。

- Tivoli Common Reporting バージョン 1.3 のディレクトリー・パスは、以下のとおりです。

```
C:¥IBM¥tivoli¥tip¥products¥tcr¥Cognos¥c8¥deployment
```

- Tivoli Common Reporting バージョン 2.1 以降のディレクトリー・パスは、以下のとおりです。

```
C:¥IBM¥tivoli¥tipv2Components¥TCRComponent¥cognos¥deployment
```

次のタスク

レポートのエクスポートについて詳しくは、「*Tivoli Common Reporting* ユーザーズ・ガイド」の『Exporting Cognos report packages』を参照してください。

Agent Builder へのレポートのインポート

Tivoli Common Reporting からエクスポートしたレポート・パッケージは、Agent Builder プロジェクトにインポートできます。その後、レポート・パッケージをエージェント・インストール・イメージに組み込むことができます。

手順

1. Agent Builder でエージェント・プロジェクトを右クリックします。
2. 「**IBM**」>「**インポート・レポート・パッケージ**」を選択します。
3. 「**インポート・レポート・パッケージ**」ウィンドウで、レポート・パッケージを作成した**データベース・タイプ**を選択します。

4. レポート・パッケージの完全修飾パスを入力するか、「**参照**」をクリックしてレポート・パッケージを選択します。
5. 「**OK**」をクリックします。
6. レポート・パッケージが、reports/*dbtype* ディレクトリーの下のエージェント・プロジェクトに表示されるようになります。

   注 : データベース固有のレポート・パッケージを作成する場合は、各パッケージを Agent Builder にインポートする必要があります。

エージェント・パッケージから Tivoli Common Reporting へのレポートのインストール
エージェントから Tivoli Common Reporting へのレポート・パッケージのインポート

手順

1. ウィザード内のステップに従って、エージェント・イメージから新規パッケージをインポートします。レポートは、エージェント・イメージの reports/*dbType*/K*xx*/reports/cognos_reports/itmk*xx*/packages にあります。
2. 圧縮 zip ファイルであるレポートを、Tivoli Common Reporting のデプロイメント・ディレクトリーにコピーします。
   - Tivoli Common Reporting バージョン 1.3 のディレクトリー・パスは、C:¥IBM¥tivoli¥tip¥products¥tcr¥Cognos¥c8¥deployment です。
   - Tivoli Common Reporting バージョン 2.1 以降のディレクトリー・パスは、C:¥IBM¥tivoli¥tipv2Components¥TCRComponent¥cognos¥deployment です。
3. Tivoli Common Reporting にログインします。
4. 「パブリック・フォルダー」に移動し、ナビゲーション・パネルの「**レポート**」で「**Common Reporting**」を選択します。
5. 「作業」レポート・セクションで、**起動**メニューをクリックして「**IBM Cognos 管理**」を選択します。
6. 「**構成**」タブに移動し、「**コンテンツ管理**」セクションを開きます。
7. 「**新規インポート**」をクリックし、パッケージのインポートを作成します。
8. エージェントのレポート・パッケージを選択します。
9. インポートするパブリック・フォルダーを選択します。
10. 「保存」を選択します。
11. 緑色 (実行) の矢印をクリックしてインポートします。

タスクの結果

詳しくは、「*Tivoli Common Reporting* ユーザーズ・ガイド」のレポート・インターフェースへのログイン を参照してください。

ICU 正規表現

ICU 正規表現の実装の詳細についての説明。

この参照コンテンツは、「*ICU User Guide*」からの抜粋です。このコンテンツでは、ICU 正規表現の実装の詳細について説明します。正規表現の実装方法はプログラミング言語によって多少異なるため、Agent Builder の正規表現機能を使用する場合にはこの情報は重要です。

表 *303.* 正規表現のメタキャラクター

文字	説明
¥a	ベル (¥u0007) に一致します。

表 303. 正規表現のメタキャラクター (続き)

文字	説明
¥A	入力の先頭に一致します。¥A は入力内の改行の後とは一致しないため、^ とは異なります。
¥b ([Set] 外部)	現行位置がワード境界の場合に一致します。 境界は、ワード (¥w) と非ワード (¥W) 文字の間の遷移で発生します。結合マークは無視されます。ワード境界について詳しくは、『ICU Boundary Analysis』を参照してください。
¥b ([Set] 内部)	バックスペース文字 (¥u0008) に一致します。
¥B	現行位置がワード境界以外の場合に一致します。
¥cX	Ctrl-X 文字に一致します。
¥d	Unicode 汎用カテゴリー Nd (Number, Decimal Digit) の任意の文字に一致します。
¥D	数字以外のすべての文字に一致します。
¥e	エスケープ (¥u001B) に一致します。
¥E	引用符で囲まれたシーケンスの終わりです (¥Q ... ¥E)。
¥f	改ページ文字 (¥u000C) に一致します。
¥G	現行位置が前回の一致の終わりである場合に一致します。
¥n	改行文字 (¥u000A) に一致します。
¥N{UNICODE CHARACTER NAME}	指定された文字に一致します。
¥p{UNICODE PROPERTY NAME}	指定された Unicode プロパティーを持つ任意の文字に一致します。
¥P{UNICODE PROPERTY NAME}	指定された Unicode プロパティーを持たない任意の文字に一致します。
¥Q	この文字から ¥E までの間のすべての文字を引用符で囲みます。
¥r	復帰文字 (¥u000D) に一致します。
¥s	空白文字に一致します。空白文字は [¥t¥n¥f¥r ¥p{Z}] として定義されます。
¥S	空白文字以外の文字に一致します。
¥t	水平タブ (¥u0009) に一致します。
¥uhhhh	16 進数値 hhhh で表される文字に一致します。
¥Uhhhhhhhh	16 進数値 hhhhhhhh で表される文字に一致します。最大 Unicode コード・ポイントは ¥U0010ffff ですが、8 桁の 16 進数値を正確に指定する必要があります。
¥w	単語構成文字に一致します。単語構成文字は [¥p{Ll}¥p{Lu}¥p{Lt}¥p{Lo}¥p{Nd}] です。

表 303. 正規表現のメタキャラクター (続き)

文字	説明
¥W	非単語構成文字に一致します。
¥x{hhhh}	16 進数値 hhhh で表される文字に一致します。1 から 6 桁の 16 進数値を指定できます。
¥xhh	2 桁の 16 進数値 hh で表される文字に一致します。
¥X	書記素クラスターに一致します。
¥Z¥	現行位置が入力の終わりである場合 (最終行末文字がある場合はさらに最終行末文字より前にある場合) に一致します。
¥z	現行位置が入力の終わりである場合に一致します。
¥n	後方参照。n 番目の捕捉グループに一致する内容に一致します。n は、1 より大きく、パターン内の捕捉グループの総数より小さい値でなければなりません。 **注 :** ICU の正規表現では、8 進数のエスケープ文字 (¥012 など) はサポートされていません。
[pattern]	セットの中の任意の 1 文字に一致します。パターンに使用できる文字についての詳細は、UnicodeSet を参照してください。
.	任意の文字に一致します。
^	行の先頭に一致します。
$	行の末尾に一致します。
\	この後に続く文字を引用符で囲みます。 リテラルとして扱われるために引用符で囲む必要がある文字は、* ? + [( ) { } ^ $ | ¥ . / です。

表 304. 正規表現の演算子

オペレーター	説明
|	選択演算子。A|B は A または B のいずれかに一致します。
*	0 回以上の繰り返しに一致します。最多一致。
+	1 回以上の繰り返しに一致します。最多一致。
?	0 回または 1 回の繰り返しに一致します。1 回が優先されます。
{n}	正確に n 回の繰り返しに一致します。
{n,}	n 回以上の繰り返しに一致します。最多一致。
{n,m}	n 回から m 回の繰り返しに一致します。最多一致 (最大 m 回)。
*?	0 回以上の繰り返しに一致します。最少一致。
+?	1 回以上の繰り返しに一致します。最少一致。

表 *304.* 正規表現の演算子 *(*続き*)*

オペレーター	**説明**
??	0 回または 1 回の繰り返しに一致します。0 回が優先されます。
{n}?	正確に n 回の繰り返しに一致します。
{n,}?	n 回から、パターン全体の一致に必要な回数までの繰り返しに一致します。
{n,m}?	n 回から m 回の繰り返しに一致します。最少一致 (最小 n 回)。
*+	0 回以上の繰り返しに一致します。1 番目の繰り返しが検出された時点での最多一致です。全体的な一致が失敗した場合でも、少ない数で再試行しないでください (所有的一致)。
++	1 回以上の繰り返しに一致します。所有的一致。
?+	0 回または 1 回の繰り返しに一致します。所有的一致。
{n}+	正確に n 回の繰り返しに一致します。
{n,}+	n 回以上の繰り返しに一致します。所有的一致。
{n,m}+	n 回から m 回の繰り返しに一致します。所有的一致。
(...)	捕捉括弧。この一致の後、括弧内の副次式に一致する範囲の入力が使用可能になります。
(?: ...)	非捕捉括弧。組み込まれているパターンをグループ化しますが、一致テキストの捕捉は行われません。捕捉括弧より効果的です。
(?> ...)	アトミック・グループを囲む括弧。括弧内の副次式との最初の一致のみが試行されます。全体的なパターン一致にならない場合は、「(?>」の前の位置との一致の検索がバックアップされます。
(?# ...)	自由形式のコメント (?# comment)。
(?= ...)	先読みアサーション。括弧内のパターンが現行の入力位置に一致するが、入力位置が先に進まない場合は true です。
(?! ...)	否定先読みアサーション。括弧内のパターンが現行の入力位置に一致しない場合は true です。入力位置が先に進みません。
(?<= ...)	後読みアサーション。括弧内のパターンが現在の入力位置より前のテキストに一致する場合は true となります。一致の最終文字は、現在位置の直前の入力文字です。入力位置は変更されません。後読みパターンに一致するストリングの長さは、アンバインドされていてはなりません (* または + 演算子なし)。

表 *304.* 正規表現の演算子 *(続き)*	
オペレーター	**説明**
(?<!...)	否定後読みアサーション。括弧内のパターンが現在の入力位置より前のテキストに一致しない場合に、true になります。一致の最終文字は、現在位置の直前の入力文字です。入力位置は変更されません。後読みパターンに一致するストリングの長さは、アンバインドされていてはなりません (* または + 演算子なし)。
(?ismx-ismx: ...)	フラグ設定。括弧で囲んだ式と指定されたフラグ (有効または無効) が評価されます。
(?ismx-ismx)	フラグ設定。フラグ設定を変更します。変更は、この設定の後に続くパターンの部分に適用されます。例えば (?i) は大/小文字を区別しない一致に変更されます。

置換テキスト

検索/置換操作の置換テキストには、検索の捕捉グループ・テキストへの参照を含めることができます。参照は $n という形式で指定します。この n は、捕捉グループの 番号です。

表 *305.* 置換テキストの文字	
文字	**説明**
$n	定位置捕捉グループ n のテキストが $n に置換されます。n は 0 以上で、捕捉グループの数以下である必要があります。$ の後に数字が指定されていない場合の特別な効果はありません。これは、代替テキストで文字 $ 自体として表示されます。
\	この文字はリテラルとして扱われ、特別な効果はありません。代替テキストで円記号によるエスケープが必要であるのは「$」と「¥」のみです。ただし、他の文字に対して円記号を使用しても、悪影響はありません。
$@n	捕捉グループ n のテキストが、捕捉グループ n に一致する正規表現に置換されます。n は、>0 以上で、捕捉グループの数以下である必要があります。$@ の後に数字が指定されていない場合の特別な効果はありません。これは、代替テキストで文字 $@ 自体として表示されます。
$#n	一致した捕捉グループ n のテキストが、$#n に代入されます。n は 0 以上で、一致した捕捉グループの数以下である必要があります。$# の後に数字が指定されていない場合の特別な効果はありません。これは、代替テキストで文字 $# 自体として表示されます。

フラグ・オプション

正規表現突き合わせのさまざまな特性を制御するフラグを以下に示します。式を RegexPattern オブジェクトにコンパイルする時点でフラグ値を指定することができます。あるいは、(?ismx-ismx) パターン・オプションを使用してパターン自体の中でフラグ値を指定できます。

表 306. フラグ・オプション

フラグ (パターン)	フラグ (API 定数)	説明
i	UREGEX_CASE_INSENSITIVE	設定されている場合、突き合わせでは大/小文字が区別されません。
x	UREGEX_COMMENTS	設定されている場合、パターン内で空白文字と # コメントを使用できます。
s	UREGEX_DOTALL	設定されている場合、パターン内の「.」は入力テキストの行終了文字と一致します。デフォルトでは一致しません。テキスト中では、復帰/改行のペアは単一行終了文字として動作し、RE パターン内の 1 つの「.」に一致します。
m	UREGEX_MULTILINE	パターン内の「^」および「$」の動作を制御します。デフォルトでは、これらのパターンはそれぞれ、入力テキストの開始と終了に一致します。 このフラグが設定されている場合、「^」と「$」は入力テキストの各行の開始と終了にも一致します。

非エージェント・ファイル・バンドルの作成

Tivoli Monitoring デポに格納可能なファイル・バンドルを作成できるようになっています。これらのファイル・バンドルは、 後から環境内のターゲット・システムにデプロイできます。

このタスクについて

この機能を使用して、リモート構成オプションがない製品をリモート側から構成することができます。この機能を使用するには、定義済みの構成ファイルをデポに格納し、目的とするシステムに送信します。

手順

1. Agent Builder から、「**ファイル**」 > 「**新規**」 > 「**その他**」と選択します。
2. 「**Agent Builder**」で、 「**非エージェント・リモート・デプロイ・バンドル**」を選択します。
3. 「**次へ**」をクリックします。
4. 「**プロジェクト名**」フィールドに プロジェクトの名前を入力します。
5. 「**次へ**」をクリックします。
6. 「**リモート・デプロイ・バンドル情報**」ウィンドウに、 以下のように情報を入力します。
   a) 「**バンドル ID**」フィールドに、ID とする 3 文字から 31 文字の固有の英数字ストリングを入力します。このストリングにはハイフンを入れることができます。ストリングの先頭は、K または ハイフン以外の英数字でなければなりません。
   b) 「**バンドル記述**」フィールドに、 バンドルの説明を入力します。

c)「**バージョン**」フィールドに、バンドルのバージョンを VVRRMMFFF の形式で入力します。ここで、vv はバージョン番号、rr はリリース番号、mm はモディフィケーション 番号 (フィックスパック番号)、fff は暫定修正番号です。

7.「**オペレーティング・システム**」域で、このバンドルをデプロイできる オペレーティング・システムを選択します。
8.「**完了**」をクリックすると、ワークスペースにプロジェクトが作成され、**Remote Deploy Bundle Editor** が開きます。

Remote Deploy Bundle Editor

Remote Deploy Bundle Editor を使用して、ファイル・バンドルのデプロイに役立つコマンドを生成します。

Remote Deploy Bundle Editor は、 プロジェクトのバンドルに関する情報を提供します。

「**バンドル識別情報**」セクションには 以下の情報が表示されます。

バンドル ID
バンドルの固有 ID

バンドル記述
バンドルの説明

バンドル・バージョン
バンドルのバージョン

ビルド
バンドルのビルド ID。ここにはビルド番号を入力します。 ビルド番号を指定しない場合は、バンドルを生成した日時から 番号が生成されます。

「**バンドル内のファイルに対するコピー・コマンドの 作成**」**チェック・ボックス**
このチェック・ボックスをクリックすると、バンドルのデプロイ時に実行される 一連のデフォルトのコピー・コマンドが生成されます。ファイルは、「**コピー場所**」テキスト・ボックスで指定された場所に コピーされます。デフォルトのロケーションは *INSTALLDIR* です。コマンド行デプロイメントから KDY.*INSTALLDIR*=... を設定することにより、このリモート・デプロイメント変数を指定します。

「**オペレーティング・システム**」セクションには、このバンドルをデプロイできる オペレーティング・システムが表示されます。

「**コマンド**」セクションには、このバンドルをデプロイするときに 実行するコマンドが表示されます。

「**前提条件バンドル**」セクションには、このバンドルが機能するために 存在する必要があるバンドルが表示されます。

Remote Deploy Bundle Editor を使用して、設定したロケーションにバンドル内のファイルをコピーする、一連のデフォルトのコピー・コマンドを選択します。このオプションを選択すると、バンドル・プロジェクト内の各ファイルに対するコピー・コマンドが生成されます。デフォルトの コピー場所は *INSTALLDIR* です。これは特殊なリモート・デプロイメント変数で、デプロイメント・コマンド行で設定しなかった場合、デフォルトで *CANDLEHOME* になります。*INSTALLDIR* によって 指定される場所を変更するには、**addSystem** コマンドを実行するときに **KDY.INSTALLDIR** プロパティーを 指定します。

バンドル・プロジェクトで指定されたものと 同じディレクトリー構造が *INSTALLDIR* に複製されます。例えば、バンドル・プロジェクトに config というフォルダーがあり、そのフォルダーに myprod.config というファイルが存在する場合は、生成されたコピー・コマンドにより、バンドルのデプロイ時にそのファイルが *INSTALLDIR*/config/myprod.config にコピーされます。

バンドルへのコマンドの追加

デプロイメント中に実行するその他のコマンドを指定できます。

このタスクについて

Remote Deploy Bundle Editor を使用して、デプロイメント中に実行するその他のコマンドを指定できます。

手順

1. デプロイメント中に実行するその他のコマンドを指定するには、**Remote Deploy Bundle Editor** の「**コマンド**」セクションにある「**追加**」をクリックします。
2. 「**コマンド**」ウィンドウで、コマンドの種類 (**プリインストール**、**インストール**、**ポストインストール**、または**アンインストール**) を選択した後、実行するコマンドを指定します。
   必ず、実行するコマンドの完全修飾パスを指定してください。便宜上、リモート・デプロイには 一連の定義済み変数が用意されています。コマンドで変数を参照するには、 変数を縦線で囲みます (例: |DEPLOYDIR|)。コマンドの定義済み変数について詳しくは、1496 ページの表 307 を参照してください。

表 *307.* コマンド用の定義済み変数

変数	説明
DEPLOYDIR	デプロイメント中にバンドルが格納されたエンドポイントの一時ディレクトリー。例えば、バンドルに組み込まれたスクリプト `myscript.sh` を実行する場合は、コマンド `|DEPLOYDIR|/myscript.sh` を指定します。
INSTALLDIR	*CANDLEHOME* または *KDY.INSTALLDIR* の値 (**addSystem** コマンドで指定された場合) のいずれか。
CANDLEHOME	Tivoli Monitoring の インストール・ディレクトリー。

3. 最後に、コマンドを実行する「**オペレーティング・システム**」を選択します。

バンドルへの前提条件の追加

Remote Deploy Bundle Editor を使用して、バンドルの前提条件を指定します。

手順

1. 前提条件を追加するには、**Remote Deploy Bundle Editor** の「**バンドル情報**」ページで、「**前提条件バンドル**」セクションにある「**追加**」をクリックします。
2. 「**新規前提条件**」ウィンドウに、このバンドルが依存するバンドルの ID および必要な最小バージョンを入力します。
3. この前提条件を必要とするオペレーティング・システムを選択します。
4. 「**OK**」をクリックして完了し、終了します。

バンドルへのファイルの追加

Remote Deploy Bundle Editor を使用して、ファイル・バンドルにファイルを追加します。

手順

1. リモート・デプロイ・バンドルにファイルを追加するには、 以下のいずれかの手順を行います。
   - Bundle Editor で「**バンドルへのファイルの追加**」を クリックします。

- ナビゲーター・ツリーでプロジェクトを右クリックした後、「**IBM Tivoli Monitoring リモート・デプロイ**」 > 「**バンドルへのファイルの追加**」をクリックします。

これらのいずれのアクションの場合も、「**バンドル・ファイルのインポート**」ウィンドウが表示されます。

2. 個々のファイルまたはファイルを含むディレクトリーを「**ファイル情報**」領域で指定します。
3. 「**完了**」をクリックします。

指定したファイルまたはディレクトリーがプロジェクト・ディレクトリーにコピーされます。リモート・デプロイ・バンドルの作成時には、 プロジェクトのディレクトリー構造が 保持されます。Agent Builder にデフォルトのコピー・コマンドを生成させる場合は、デプロイメント用の正しいディレクトリー構造にファイルが入っていることを確認してください。

バンドルの生成

Agent Builder を使用して、エージェントのリモート・デプロイメント用のバンドルを生成します。

手順

1. リモート・デプロイメント・バンドルを生成するには、以下のいずれかの手順を実行して「**最終リモート・デプロイ・バンドルの生成**」ウィンドウを表示させます。
   - **Remote Deploy Bundle Editor** で、「**最終リモート・デプロイ・バンドルを生成します**」をクリックします。
   - ナビゲーター・ツリーでプロジェクトを右クリックした後、「**IBM Tivoli Monitoring リモート・デプロイ**」 > 「**リモート・デプロイ・バンドルの生成**」をクリックします。
2. これで、バンドルを以下の 2 とおりの方法で生成できます。
   - Agent Builder を実行しているシステムに Tivoli Enterprise Monitoring Server がある場合は、「「**リモート・デプロイ・バンドル**」**をローカル TEMS デポに インストール**」をクリックします。

     Agent Builder は、Tivoli Monitoring のインストール先を判別して 「**ディレクトリー**」フィールドに入力しようとします。*CANDLE_HOME* が設定されていない場合は、 デフォルトの場所 `C:¥IBM¥ITM` または `/opt/IBM/ITM` が使用されます。インストール場所が正しいことを確認してから続行してください。

     バンドルをインストールするには、Tivoli Enterprise Monitoring Server の ログイン情報を指定する必要があります。
   - システム上のディレクトリーにバンドルを生成するには、「「**リモート・デプロイ・バンドル**」**をローカル・ディレクトリーに生成**」をクリックします

     処理が完了したら、必ずこのディレクトリーを Tivoli Enterprise Monitoring Server システムに転送し、`tacmd addbundles` コマンドを使用してバンドルをデポに追加してください。

次のタスク

バンドルをデプロイするときには、`tacmd addSystem` コマンドを使用してください。例えば、次のようにします。

```
tacmd addsystem -t MONITORINGCOLLECTION -n Primary:ITMAGT:NT
```

`-t` (type) は、`tacmd viewDepot` コマンドから返される製品コードです。

```
>tacmd viewDepot
Product Code : MONITORINGCOLLECTION
Version : 010000003
Description : MonitoringCollectionScripts
Host Type : WINNT
Host Version : WINNT
前提条件:
```

注 : Tivoli Enterprise Portal Desktop/Browser からリモートでデプロイすることはできません。Tivoli Enterprise Portal Desktop またはブラウザーからリモートでデプロイを実行すると、KFWITM219E メッセージが出力されます。

詳しくは、Tivoli Monitoring 資料を参照してください。

Tivoli Netcool/OMNIbus プローブのデプロイ可能バンドルの作成

Agent Builder を使用して、Tivoli Netcool/OMNIbus プローブを リモート・コンピューターにデプロイするために使用できるパッケージおよび 構成バンドルを作成できます。

このタスクについて

プローブのリモート・デプロイをサポートするために、 プローブのデプロイ前にリモート・コンピューターに デプロイできる Tivoli Netcool/OMNIbus バンドルを 作成することもできます。

手順

1. Agent Builder から、「**ファイル**」 > 「**新規**」 > 「**その他**」と選択します。
2. 「**IBM Tivoli OMNIbus のウィザード**」で 「**パッケージ・バンドル**」を選択します。
3. 「**次へ**」をクリックします。

次のタスク

次に、「**OMNIbus インストール・バンドル**」ウィザードを使用して、バンドルを作成します。このウィザードの使用について詳しくは、Tivoli Netcool/OMNIbus の資料を参照してください。

動的ファイル名のサポート

実際のファイル名の代わりにファイル名のパターンを指定するには、動的ファイル名のサポートを使用します。

一部のアプリケーション・プログラムで作成される出力ファイル名は、その後変更される可能性があります。特定の基準 (現在の年月日など) に基づいて名前が変更されたり、増分順序番号を含むファイル名であったりします。そのような場合は、実際のファイル名の代わりにファイル名のパターンを指定できます。ファイル名のパターンを指定するときに 認識されるパターンの形式は以下の 2 つです。

- 正規表現 (推奨)。
- IBM Tivoli Universal Agent 動的ファイル名の構文 (非推奨)。

正規表現のファイル名パターン

ファイル名のパターンを指定するために、International Components for Unicode (ICU) 構文に従った正規表現を使用できます (この構文については、1489 ページの『ICU 正規表現』 を参照してください)。この機能を使用するには、 「**詳細なログ・ファイル属性グループ情報**」ページで「**ファイル名が正規表現に一致する**」チェック・ボックスを 選択する必要があります。正規表現のパターンを指定する場合は、 「**詳細なログ・ファイル属性グループ情報**」ページの「**複数のファイルが一致する場合は次のファイルを選択**」リストから オプションを選択して、一致する最新のファイルを選択するための ガイドラインを指定する必要があります。

注 : ファイル名のパターンは、正規表現で指定することをお勧めします。

詳細なログ・ファイル属性グループ・プロパティーの構成方法について詳しくは、1260 ページの『ログ・ファイルのモニター』、ステップ 1261 ページの『6』 を参照してください。例えば、 以下のファイル名パターンを指定したとします。

```
d:¥program files¥logs¥tivoli.*
```

このパターンは、`d:¥program files¥logs` ディレクトリーにあり、 `tivoli` で始まるファイル名を検索します。正規表現を指定できるのはファイル名の部分だけです。 パス名の部分には指定できません。

動的ファイル名の構文

動的ファイル名の構文を使用すると、一度に 1 つだけのファイルを モニターできます。ファイル・データ・プロバイダーは、指定されたパス・ロケーション内のすべてのファイルを調べて、 定義されたパターンに一致するファイルを探します。ファイル・データ・プロバイダーは、一致するファイル名の中で最も大きい番号を持つものを判別するか、日時値に基づいて、常に最新の一致ファイルをモニターします。モニター対象とするファイルは、 ファイルの作成日時などの基準ではなく、ファイル名によって判別されます。

パターンは、任意の数の可変部分を持つファイル名に対して 指定できます。 例えば Log{###} は、Log010 や Log456 など、可変部分が 1 個所のファイル名の場合に 一致します。可変部分が複数のファイル名の場合、パターン文字は、 ファイル名の任意の部分 (つまり複数の可変部分) で指定できます。例えば、aaa.bbb{???}.ccc は 有効なパターンであり、aaa.bbb{???}.ccc{###} も有効です。

注 : ファイル名のパターンを指定するには、動的ファイル名の構文ではなく、正規表現をお勧めします。正規表現について詳しくは、1498 ページの『正規表現のファイル名パターン』を参照してください。

ファイル名パターンの仕様を以下の例で説明します。

{#########}.abc
長さが 8 でファイル拡張子が *.abc* の数字ファイル名に一致します (10252006.abc や 10262006.abc など)。10262006 は 10252006 より大きいため、ファイル 10262006.abc がモニターされます。

{#########}.*
長さが 8 の数字ファイル名に一致します。ファイル拡張子は 無視します。例えば 20061025.log、20061101.log、10252006.abc が該当します。 20061101 が最大であるため、ファイル 20061101.log がモニターされます。

{#######??}.abc
長さが 8 でファイル拡張子が .abc の数字ファイル名に一致します。 名前部分の最後の 2 桁は無視します。例えば 02110199.abc、02110200.abc、021101AZ.abc が 該当します。021102 が最大であるため、ファイル 02110200.abc がモニターされます。

Console.{######}
名前部分に *Console* を含み、拡張子部分に 6 桁の 数値を含むファイル名に一致します。例えば Console.000133、Console.000201、Console.000134 が 該当します。ファイル Console.000201 がモニターされます。

IN{######}.log
IN で始まり、その後に 6 桁の数字およびファイル拡張子 .log が続く ファイル名に一致します。例えば IN021001.log、IN021002.log、IN021004.log が 該当します。ファイル IN021004.log がモニターされます。

PS{###}FTP.txt
PS で始まり、その後に 3 桁の数字が続き、さらに FTP および 拡張子 .txt が続くファイル名に一致します。例えば PS001FTP.txt、PS005FTP.txt、PS010FTP.txt が 該当します。ファイル PS010FTP.txt がモニターされます。

ファイル名のパターンを設定するには、以下のガイドラインに従ってください。

- 中括弧 {} を使用してファイル名の中のパターン文字を囲みます。中括弧の中に パターン文字が存在することで、ファイル名のパターンが使用されていることが 示されます。
- アスタリスク (*) は、ファイル拡張子またはファイル名の中のすべての後続文字を 無視するためにワイルドカードとして使用します。例えば、Myapp{###}.log* は、Myapp で始まり、その後に 3 桁の数字が続き、さらに .log が続くファイル名が、その後に続く文字に関係なくすべて一致することを示します。

  アスタリスクは、中括弧 ({ }) の後に指定しなければならず、 ファイル名の先頭で使用することはできません。ファイル名拡張子に アスタリスクを使用する場合は、アスタリスクを単独で 使用しなければなりません。

  正しいワイルドカード (*) の使用例を以下に示します。

  err{??}.*

error{$}.*

誤ったワイルドカード (*) の使用例を以下に示します。

error.20*
アスタリスク (*) の前に中括弧がありません。

error*.{###}
アスタリスクがファイル名の末尾で使用されていません。

error.*
アスタリスク (*) の前に中括弧がありません。

- 特定のファイル拡張子を定義した場合は、それと同じ拡張子を持つ ファイルのみが考慮されます。
- ファイル名の各数字要素を示すには、番号記号を使用します。
- 疑問符は、命名規則の各要素のうち、 該当するファイル名を判別するときの検索基準として扱わない要素を 除外するために使用します。
- ドル記号 ($) は、任意の文字または文字なしのいずれかを 表すために使用します。例えば、Log および LogA の 2 つのファイルに 一致させるには、Log{$} と指定します。ドル記号の使用には いくつかの制約があります。{$$$$$$}_abc.log のように 1 つ以上のドル記号をファイル名の前に記述した場合、ドル記号数は、ファイル名のその位置にある文字の数に正確に一致しなければなりません。また、ファイル名パターンの中で複数の位置にドル記号を指定することはできません。例えば、{$$$}b{$$$}.log は abc.log に一致しません。ドル記号にはこのような制約があるため、ファイル名に含まれる文字の数が予測できない場合は、正規表現のファイル名パターンを使用してください。
- 中括弧で囲んだ番号記号および疑問符の総数は重要です。これは、ファイル名の一部に正確に一致しなければなりません。 例えば、パターン AA{####} は、AA0001 などのファイルを 検索するようにファイル・データ・プロバイダーに指示します。AA001 や AA00001 などの ファイル名は考慮に入れられません。
- 厳密なファイル名パターン (定数および数値部分) は、 ファイル名に正確に一致しなければなりません。例えば、パターン AA{###} は、ファイル AA101 を 検査するようにファイル・データ・プロバイダーに指示します。XAA101、AA222X、AA55555 などの ファイル名は考慮に入れられません。
- 予約済みパターン・ストリング {TIVOLILOGTIME} は、Tivoli Monitoring の エージェントまたはサーバー・ログ・ファイルの 16 進タイム・スタンプおよび ファイル・シーケンス番号の代わりとして使用します。このパターン・ストリングは、Tivoli Monitoring コンポーネントの自己モニターを実行しているときに役立ちます。例えば、/opt/IBM/ITM/logs ディレクトリーにある最新の モニター・サーバー ログをモニターする場合は、以下のファイル名パターンを指定できます。

```
/opt/IBM/ITM/logs/Host1_ms_{TIVOLILOGTIME}.log
```

Host1_ms_452053c0-01.log、Host1_ms_451f11f4-01.log、Host1_ms_45205946-01.log、および Host1_ms_451f11f4-02.log が /logs ディレクトリーに存在する場合は、Host1_ms_45205946-01.log ファイルがモニター対象として選択されます。

日付の構成要素 (年、月、および日) で構成されるファイル名を厳密に指定するには、 大文字の Y、M、および D を使用します。これらの文字は中括弧の中で 指定しなければなりません。さもなければ、ファイル名の中のリテラル文字として 扱われてしまいます。

以下に例を示します。

{YYYYMMDD}.log
20060930.log や 20061015.log などのファイル名を指定します。

{MMDDYY}.log
101106.log や 110106.log などのファイル名を指定します。

{DDMMYYYY}.log
01092006.log や 15082006.log などのファイル名を指定します。

{DDMMMYY}.log
24Jan07 や 13Sep06 などのファイル名を指定します。

{MM-DD-YY}.log
11-02-06 や 04-29-07 などのファイル名を指定します。日付フィールドの分離文字 (-) は無視されるため、疑問符 (?) のパターン文字を使用してスキップする必要はありません。

MY{YYDDD}.log
MY06202.log、MY06010.log、MY04350.log などのファイル名を指定します。

複雑なケースになると、長いファイル名の内部に日付フィールドが埋め込まれていて、前の例に示した日付パターンでは不十分な場合があります。複雑なケースの場合、番号記号と疑問符を混在させたパターンを作成して数値比較を実行し、最新のファイルをモニター対象として選択することができます。例えば、パターン ABC{?#####?##?##?##?##?##?}XYZ.TXT を ABC 2006-04-20 11_22_33 XYZ.TXT などのファイル名に対して使用できます。この例では、#- で示した桁のみを対象にしており、疑問符はプレースホルダーとして機能して、ファイル名の中の他の文字を無視します。

ファイル・データ・プロバイダーは、定義されたファイル・パターンに一致する新しいファイルがないか、ターゲットのパス・ロケーションを定期的に検査します。パターンに一致する 新しいファイルを検出すると、ファイル・データ・プロバイダーは、アプリケーションのモニターを その新しいファイルに自動的に切り替えます。ファイル・データ・プロバイダーは、 以下の場合に最良一致を検索します。

- ファイル・データ・プロバイダーが最初に開始した。
- 現在のモニター対象のファイルが存在しなくなった (多くの場合は名前変更や削除が原因)。
- 既存のファイルの内容が変更された (多くの場合は再書き込みが原因)。
- 検査間隔が経過した。デフォルトの間隔は 10 分です。 環境変数を指定することで、 間隔を長い値または短い値に変更できます。

```
KUMP_DP_FILE_SWITCH_CHECK_INTERVAL=number-of-seconds
```

SNMP トラップ構成

ここでは、トラップ情報をより読みやすい形式でレンダリングするために SNMP データ・プロバイダーによって使用される構成ファイルについて説明します。 このファイルは、トラップに対するカテゴリー、重大度、状況、およびソース ID の割り当てにも使用されます。

また、デフォルト・ファイルの変更や、ユーザー独自の構成ファイルへの置き換えについても説明されています。

SNMP トラップ構成ファイル、trapcnfg

SNMP データ・プロバイダー は、開始時に trapcnfg という構成ファイルを読み取ります。 このファイルの目的の 1 つは、SNMP トラップ情報を読みやすい形式に 変換することです。もう 1 つの目的は、特定のトラップに対して、カテゴリー、重大度、状況、およびソース ID を割り当てることです。これらのカテゴリーは、SNMP では定義されていないためです。

新しいトラップまたはエンタープライズ 定義を追加したり 既存のものを変更したりすることで、サイト固有のニーズに合わせて trapcnfg ファイルを 変更することができます。また、独自の構成ファイルを使用することもできます。

HP OpenView trapd.conf ファイルの使用

trapcnfg ファイルは、HP OpenView Network Node Manager のトラップ構成ファイル trapd.conf と同じ形式ではありませんが、類似した形式を持つファイルです。そのため、必要に応じて OpenView ファイルをコピーし、その多くの定義ステートメントを再利用することができます。

レコードのタイプ

trapcnfg には、以下の 3 種類のレコードまたはレコード・ブロックがあります。

コメント
コメント・レコードは番号記号 (#) で始まります。

エンタープライズ定義
エンタープライズ定義は、ブランクで区切られた 2 個のトークンから構成されます。 最初のトークンは名前であり、2 番目のトークンはオブジェクト ID (OID) を 中括弧 ({ }) で囲んだものです。

トラップ定義
トラップ定義は、ブランクで区切られた 8 個のトークンから構成されます。それぞれの定義は 複数のレコードから構成されるため、トラップ定義はブロック・レコードです。

最初のタイプは、改めて説明するまでもありません。1502 ページの図 87 に、2 番目と 3 番目のタイプの例を示します。

1502 ページの図 87 の最初の例には、エンタープライズ定義レコードを示しています。このレコードは、エンタープライズ OID 1.3.6.1.4.1.311.1.1.3.1.1 が Microsoft Windows NT であることを定義しています。

2 番目の例にはトラップ定義レコードを示しています。このレコードは、 エンタープライズ OID 1.3.6.1.4.1.311.1.1.3.1.1、汎用トラップ番号 0、 および固有トラップ番号 0 に trapName MSNTCOLD を関連付けると定義しています。 重大度は 10 進数の形式であるのに対し、カテゴリーはテキスト形式であることに注意してください。 重大度は、テキスト形式に変換されてから表示されます。 タイプ 3 レコード・ブロックの 次のレコードは簡略説明であり、Agent Builder は使用しません。Agent Builder が使用するのは、区切り文字 SDESC と EDESC で囲まれた詳細説明です。

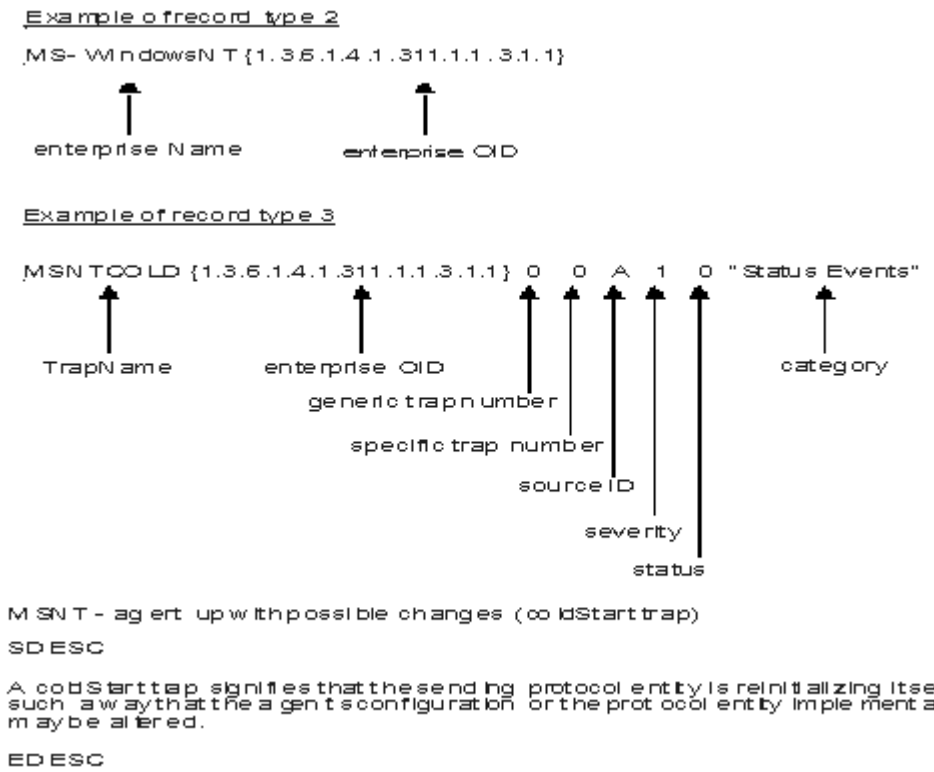


図 *87.* 構成レコード・タイプ *2* および *3* の例

trapcnfg ファイルのデフォルト

SNMP データ・プロバイダー によってサポートされるデフォルトのリストを示す表

サポートされるカテゴリー

1503 ページの表 308 に、Agent Builder によってサポートされるカテゴリーを示します。

表 *308. SNMP* データ・プロバイダーによってサポートされるカテゴリー	
カテゴリー	**テキスト表記**
0	しきい値イベント
1	ネットワーク・トポロジー・イベント
2	エラー・イベント
3	状況イベント
4	ノード構成イベント
5	アプリケーション・アラート・イベント
6	すべてのカテゴリー・イベント
7	ログ専用イベント
8	マップ・イベント
9	イベントを無視

1503 ページの表 309 に、Agent Builder によってサポートされる重大度を示します。

表 *309. SNMP* データ・プロバイダーによってサポートされる重大度	
重大度	**テキスト表記**
0	クリア
1	不定
2	警告
3	マイナー・エラー
4	クリティカル
5	メジャー・エラー

サポートされる状況

1503 ページの表 310 に、Agent Builder の構成ファイルで定義されている状況を示します。

表 *310. SNMP* データ・プロバイダーによってサポートされる状況	
状況	**テキスト表記**
0	未変更
1	不明
2	サービス中
3	最低限
4	ダウン
5	非管理対象
6	確認
7	User1
8	User2

サポートされるソース ID

1504 ページの表 311 に、trapcnfg によってサポートされるソース ID を示します。

表 *311. SNMP* データ・プロバイダーによってサポートされるソース *ID*

ソース ID	説明
a	アプリケーション
A	エージェント
C	Xnmcollect
d	デモ
D	データ・コレクター
E	Nvevents
I	Ipmap
L	LoadMIB
m	Shpmon
M	IP トポロジー
n	netmon 関連
N	netmon 生成のトラップ
O	OSI SA
P	非 IP トラップ
r	Tralertd
s	Spappld
S	セキュリティー・エージェント
t	Xnmtrap
T	Trapd
V	ベンダー関連
?	不明

アクション実行コマンド参照

アクション実行コマンドの概要、アクション実行コマンドに関する参照、および特殊なアクション実行コマンドの説明。

アクション実行コマンドについて

アクション実行コマンドは Agent Builder モニター・エージェントに組み込むことができます。アクション実行コマンドは、ポータル・クライアントから実行できます。また、シチュエーションやポリシーに組み込むことができます。 シチュエーションに組み込むと、シチュエーションが真になったときにコマンドが実行されます。シチュエーションのアクション実行コマンドは、リフレックス・オートメーションとも呼ばれます。シチュエーションでアクション実行コマンドを有効にすると、システム状態への応答を自動化することになります。 例えば、アクション実行コマンドを使用してコマンドを送信し、管理対象システムでプロセスを再開することができます。また、アクション実行コマンドを使用して、携帯電話にテキスト・メッセージを送信することもできます。

拡張オートメーションでは、ポリシー を使用して、アクションの実行、作業のスケジュール、 および手動タスクの自動化を行うことが可能です。 ポリシーは、アクティビティーと 呼ばれる一連の自動化されたステップで構成され、各ステップが接続されてワークフローを形成しています。アクティビティーの 完了後、Tivoli Enterprise Portal は、戻りコードのフィードバック、 およびそのフィードバックによって指示された次のアクティビティーが 含まれた拡張自動化論理応答を受信します。

基本的なアクション実行コマンドは、アクションの完了後に表示されるメッセージ・ボックスまたはログ・ファイル内に、操作の戻りコードを表示します。 このウィンドウを閉じると、このアクションの追加情報は使用できなくなります。

アクション実行コマンドの詳細

アクション実行コマンドの使用に関する詳細については「*Tivoli Enterprise Portal* ユーザーズ・ガイド」を参照してください。

このモニター・エージェントのアクション実行コマンドのリストと説明については、1505 ページの『特殊なアクション実行コマンド』を参照してください。個々のコマンドの該当セクションに記載されている情報も参照してください。

特殊なアクション実行コマンド

Agent Builder モニター・エージェントは、一連のアクション実行コマンドの特殊なプロセスを認識して実行できます。

- SSHEXEC

これらのコマンドを作成し、Agent Builder モニター・エージェント・プロジェクトに組み込む方法について詳しくは、1367 ページの『ワークスペース、アクション実行コマンド、およびシチュエーションの作成』を参照してください。

SSHEXEC アクション

始める前に

アクション実行コマンドについて詳しくは、1504 ページの『アクション実行コマンド参照』を参照してください。

このタスクについて

SSHEXEC アクションは、少なくとも 1 つの SSH スクリプト属性グループを持つモニター対象アプリケーション用のアクションと見なされます。これは、SSHEXEC キーワードに続くコマンドが、SSH ターゲット・システムにおいてリモートで 開始されることを示します。コマンドは、SSH ターゲット・システムをモニターするよう構成された、ユーザーの資格情報および特権を使用して開始します。このコマンドは、管理対象システム 名で表されるリモート・システムで実行します。

手順

シチュエーションまたはワークフロー・ポリシーにアクション実行コマンドを組み込むには、システム・コマンドの以下の構文を使用します。

```
SSHEXEC [Command]
```

以下に例を示します。

```
SSHEXEC [ls &path]
```

注 : *Command* とともにアクション実行引数オプションを使用して、アクション実行の呼び出し中にコマンド全体またはコマンドの一部をカスタマイズすることができます。

注 : *Command* に複数の引数を指定する場合、**tacmd** コマンド行インターフェースを 使用してアクション実行コマンドを呼び出すことができるように、大括弧を使用してください。

アクセシビリティー機能

アクセシビリティー機能は、運動障害または視覚障害など身体に障害を持つお客様が、情報技術コンテンツを快適に使用できるように支援します。

アクセシビリティー機能

IBM Cloud Application Performance Management の Web ベースのインターフェースは 、Cloud APM コンソールです。このコンソールの主なアクセシビリティー機能は、次のとおりです。

- ユーザーは、スクリーン・リーダー・ソフトウェアやデジタル音声シンセサイザーなどの支援技術を使用して、画面に表示されている内容を音声で聞き取ることができます。本製品の支援技術の使用について詳しくは、支援技術の製品資料を参照してください。
- キーボードのみを使用して、特定の機能または同等の機能をユーザーが操作できるようにします。
- すべての情報を、色に依存せずに伝えます。[1]

Cloud APM コンソールでは、US Section 508 (http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards) および Web Content Accessibility Guidelines (WCAG) 2.0 に準拠するための最新の W3C 標準 WAI-ARIA 1.0 (http://www.w3.org/TR/wai-aria/) が使用されています。アクセシビリティー機能を活用するには、最新リリースのスクリーン・リーダーを、この製品でサポートされる最新の Web ブラウザーと組み合わせて使用してください。

IBM Knowledge Center の Cloud APM コンソールのオンライン製品資料は、アクセシビリティーに 対応しています。 IBM Knowledge Center のアクセシビリティー機能については、IBM Knowledge Center のリリース情報 を参照してください。

キーボード・ナビゲーション

この製品は、標準ナビゲーション・キーを採用しています。

インターフェースに関する情報

Cloud APM コンソールの Web ユーザー・インターフェースは、コンテンツを適切にレンダリングし、使いやすさを実現するために、カスケーディング・スタイル・シートを使用していません。 ただし、製品資料はカスケーディング・スタイル・シートを使用しています。 IBM Knowledge Center には、低視力のユーザーがカスタムのシステム表示設定を使用するための、ハイコントラスト・モードなどの同等の方法が用意されています。 デバイスまたはブラウザーの設定を使用して、フォント・サイズを制御できます。

Cloud APM コンソールの Web ユーザー・インターフェースには WAI-ARIA ナビゲーション・ランドマークが含まれています。これを使用して、アプリケーションの機能領域に迅速に移動できます。

Cloud APM コンソール のユーザー・インターフェースには 、毎秒 2 回から 55 回フラッシュするコンテンツは含まれません。

アクセシビリティーの関連情報

標準の IBM ヘルプ・デスクおよびサポート Web サイトに加えて、IBM は、聴覚に障害をお持ちのお客様が営業担当者やサポート・サービスにアクセスするために使用できる TTY 電話サービスを用意しました。

TTY サービス 800-IBM-3383 (800-426-3383) (北アメリカ内)

IBM およびアクセシビリティー

IBM のアクセシビリティーへの 取り組みの詳細については、IBM Accessibility (www.ibm.com/able) を参照してください。

---

[1] 例外には、 Performance Management コンソールの一部の「**エージェント構成**」ページがあります。

© Copyright IBM Corp. 2014, 2019

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。この資料は、IBM から他の言語でも提供されている可能性があります。ただし、これを入手するには、本製品または当該言語版製品を所有している必要がある場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。 本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。 これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。 ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。 本書の提供は、お客様にこれらの特許権について 実施権を許諾することを意味するものではありません。 実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒 *103-8510*
東京都中央区日本橋箱崎町 *19* 番 *21* 号
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態で提供し、 商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示 もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。 本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。 IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、 改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの websites サイトを推奨するものでは ありません。それらの websites サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの websites サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、 自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムと その他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

© Copyright IBM Corp. 2014, 2019

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

本書に含まれるパフォーマンス・データは、特定の動作および環境条件下で得られたものです。 実際の結果は、異なる可能性があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。 IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向性および指針に関する記述は、予告なく変更または撤回される場合があります。これらは目標および目的を提示するものにすぎません。

本書はプランニング目的としてのみ記述されています。 記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。 より具体性を与えるために、それらの例には、個人、企業、ブランド、 あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、名前や住所が類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。 お客様は、サンプル・プログラムが書かれているオペレーティング・ プラットフォームのアプリケーション・プログラミング・インターフェースに 準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、 いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、 配布することができます。 このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。 従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性が あることをほのめかしたり、保証することはできません。 これらのサンプル・プログラムは特定物として現存するままの状態で提供されるものであり、いかなる保証も提供されません。 IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、
著作権表示を入れていただく必要があります。
© (お客様の会社名) (西暦年).
このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。
© Copyright IBM Corp. 2014, 2015.

商標

IBM、IBM ロゴおよび ibm.com は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、http://www.ibm.com/legal/copytrade.shtml をご覧ください。

Linux は、Linus Torvalds の米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

製品資料に関するご使用条件

これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。

適用範囲

IBM Web サイトの「ご利用条件」に加えて、以下のご使用条件が適用されます。

個人使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

商業的使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

権限

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入 関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態で提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。

IBM オンラインでのプライバシー・ステートメント

サービス・ソリューションとしてのソフトウェアも 含めた IBM ソフトウェア製品 (「ソフトウェア・オファリング」) では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。 IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。 ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項をご確認ください。

このソフトウェア・オファリングは、展開される構成に応じて、セッション管理、認証、およびシングル・サインオン構成の目的のために、それぞれのお客様のユーザー名を、セッションごとの Cookie を使用して収集する場合があります。これらの Cookie は無効にできますが、その場合、これらを有効にした場合の機能を活用できなくなる可能性があります。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、 お客様は、このような情報を収集するにあ

たって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれらには限られません。

このような目的での Cookie を含む様々なテクノロジーの使用の詳細については、『IBM オンラインでのプライバシー・ステートメント』(http://www.ibm.com/privacy-/details/jp/ja/) の『クッキー、ウェブ・ビーコン、その他のテクノロジー』および「IBM Software Products and Software-as-a-Service Privacy Statement」(http://www.ibm.com/software/info/product-privacy) を参照してください。