IBM Cloud Application Performance Management

2019 年 6 月

用户指南



注释

在使用本资料及其支持的产品之前，请阅读第 1277 页的『声明』中的信息。

此版本适用于 IBM® Cloud Application Performance Management 的 2019 年 6 月版本以及所有后续发行版和修订版，直至新版本中另有声明为止。

© Copyright International Business Machines Corporation 2014, 2019.

目录

第 1 章新增功能 1

第 2 章 PDF 文档 35

第 3 章产品概述 37
体系结构概述 37
用户界面 38
产品和附加组件 39
产品详细信息 41
代理程序和数据收集器 43
变更历史记录 44
功能 46
描述 49
功能部件 59
集成 65
文档 66
本文档中使用的约定 66

第 4 章规划部署 69
系统需求 69
代理程序和数据收集器使用的缺省端口。 69
方案 73
场景：监视 IBM API Connect 73
场景：监视 IBM Java 应用程序堆栈 74
场景：监视 IBM 集成堆栈 81
下载代理程序 以及数据收集器 86
教程：下载和安装代理程序 87
教程：下载和配置数据收集器 91

第 5 章代理程序 以及数据收集器部署 95

第 6 章安装代理程序 101
在 UNIX 系统上安装代理程序 102
AIX 系统上的安装前任务 103
Solaris 系统上的安装前任务 104
安装代理程序 105
在 Linux 系统上安装代理程序 107
Linux 系统上的安装前任务 109
安装代理程序 112
在 Windows 系统上安装代理程序 114
Windows 系统上的安装前任务 116
安装代理程序 117
以非 root 用户身份安装代理程序 119
保护代理程序安装文件 120
以静默方式安装代理程序 121
绕过先决条件扫描程序 123
卸载代理程序 124
WebSphere Applications 代理程序：取消配置数据收集器 126
Node.js 代理程序：移除监视插件 133
Microsoft .NET 代理程序：移除 .NET 数据收集器 133

第 7 章配置环境 135
通用主题 135
网络连接 135
受管系统名称 142
更改代理程序的受管系统名称 145
配置代理程序 150
用于配置数据收集器的常规过程 157
配置 Amazon EC2 监视 161
在 Windows 系统上配置代理程序 162
通过对提示做出响应来配置代理程序 165
使用静默响应文件来配置代理程序 166
Amazon EC2 代理程序的配置参数 167
配置 AWS 弹性负载均衡器监视 168
在 Windows 系统上配置代理程序 169
通过响应提示来配置代理程序 170
使用静默响应文件来配置代理程序 171
Amazon ELB 代理程序的配置参数 172
配置 Azure Compute 监视 173
Azure Compute 配置信息 173
在 Windows 系统上配置代理程序 174
通过响应提示来配置代理程序 177
使用静默响应文件来配置代理程序 179
Azure Compute 代理程序的配置参数 181
配置 Cassandra 监视 182
在 Windows 系统上配置代理程序 182
在 Linux 系统上配置代理程序 183
使用静默响应文件来配置代理程序 183
代理程序的配置参数 184
配置 Cisco UCS 监视 184
在 Windows 系统上配置代理程序 185
使用静默响应文件来配置代理程序 186
通过响应提示来配置代理程序 187
代理程序的配置参数 187
数据提供程序的配置参数 188
启用与 Cisco UCS 数据源的 SSL 通信 188
增加 Java 堆大小 189
配置 Citrix Virtual Desktop Infrastructure 监视 190
启用 Citrix 只读管理员权限 191
在 Windows 系统上配置代理程序 192
通过对提示做出响应来配置代理程序 195
使用静默响应文件来配置代理程序 196
Citrix VDI 代理程序的配置参数 197
启用 Windows 事件和 PowerShell 度量的监视 198
配置 DataPower 监视 199
配置 DataPower 设备 199
配置 DataPower 代理程序 207
配置 Db2 监视 210
在 Windows 系统上配置代理程序 211
在 Linux 或 UNIX 系统上配置代理程序 212
使用静默响应文件来配置代理程序 213
授予查看 Db2 度量的特权 214
配置本地环境变量 215
远程监视先决条件 217
配置 Hadoop 监视 219
在 Windows 系统上配置代理程序 220
在 Linux 和 AIX 系统上配置代理程序 222

使用静默响应文件来配置代理程序 224
配置仪表板以查看 Hadoop 事件 225
向非管理员用户授予许可权 226
配置 HMC Base 监视 226
设置 SSH 连接 228
为 HMC 准备 SDK 229
配置 HMC Console Server 以监视 Virtual I/O 230
启用对 CPU 和内存利用率的监视 230
配置 HTTP Server 监视 231
IBM HTTP Server 响应时间模块 232
HTTP Server 代理程序 代码样本 233
配置 IBM Cloud 监视 234
在 Windows 系统上配置代理程序 235
通过响应提示来配置代理程序 236
使用静默响应文件来配置代理程序 236
IBM Cloud 代理程序的配置参数 237
配置 IBM Integration Bus 监视 237
配置 IBM Integration Bus 代理程序 238
配置 IBM Integration Bus 以启用数据 242
禁用代理程序的快照数据收集 248
配置 IBM Integration Bus 代理程序的事务跟踪 248
为 IBM Integration Bus 代理程序指定唯一受管系统名称 249
移除 KQIUserExit 用户出口 250
配置 IBM MQ 设备监视 250
通过响应提示来配置代理程序 251
使用静默响应文件来配置代理程序 252
MQ Appliance 代理程序的配置参数 252
配置 InfoSphere DataStage 监视 255
在 Windows 系统上配置代理程序 255
在 Linux 系统上配置代理程序 255
配置环境变量 255
使用静默响应文件来配置代理程序 256
代理程序的配置参数 257
p 配置 Internet Service Monitor 258
通过用户界面配置 Internet Service Monitoring 258
在 Windows 系统上配置代理程序 385
启用 Netcool/OMNIbus 388
配置 J2SE 监视 388
检查事务跟踪和诊断数据收集的状态 393
更改事务跟踪和诊断数据收集的状态 393
配置 JBoss 监视 394
启用 JMX MBean 服务器连接 395
添加 JBoss 服务器管理用户 397
启用 Web/HTTP 统计信息收集 397
在 Windows 系统上配置代理程序 399
通过对提示做出响应来配置代理程序 401
使用静默响应文件来配置代理程序 402
JBoss 代理程序的配置参数 403
设置 JBoss 代理程序事务跟踪数据收集器 404
配置 Linux KVM 监视 409
创建用户和授予所需许可权 409
配置协议 410
配置 RHEVM 服务器连接 413
配置 RHEVH 服务器连接 415
用于连接至 RHEVM 服务器的配置参数 415
用于连接至 RHEVH 服务器的配置参数 416
配置 MariaDB 监视 418
在 Windows 系统上配置代理程序 418

在 Linux 系统上配置代理程序 419
使用静默响应文件来配置代理程序 420
配置 Microsoft Active Directory 监视 421
以管理员用户身份运行 Microsoft Active Directory 代理程序 421
配置本地环境变量 421
以非管理员用户身份运行 Microsoft Active Directory 代理程序 423
为属性组 AD_Services_Status 配置域服务 425
升级 Microsoft Active Directory 代理程序 426
配置 Microsoft Cluster Server 监视 426
在 Windows Server 2008、2012、2016 和 2019 系统上创建通用服务集群资源 427
使用静默响应文件来配置代理程序 427
更改用户帐户 428
配置 Microsoft Exchange 监视 428
创建用户 429
向 Exchange Server 用户分配管理员权限 431
使 Exchange Server 用户成为本地管理员 433
配置 Exchange Server 以获取可达性 434
配置代理程序以在域用户下运行 435
本地配置代理程序 435
使用静默响应文件来配置代理程序 438
配置代理程序的本地环境变量 439
配置 Microsoft Hyper-V 监视 439
提供本地安全策略以便在 Windows 上以非管理员用户身份运行 Monitoring Agent for Microsoft Hyper-V Server 440
授予本地安全策略许可权 441
修改 DCOM 许可权 442
在 Hyper-V Administrator Users 组中添加非管理员用户 442
在 Performance Business Monitor Users 组中添加非管理员用户 442
配置 Microsoft IIS 监视 443
在 Windows 系统上配置代理程序 443
使用静默响应文件来配置代理程序 444
更改用户帐户 444
配置 Skype for Business Server（旧称 Microsoft Lync Server）监视 445
非管理员用户的许可权和许可权 445
在 Windows 系统上配置代理程序 446
使用静默响应文件来配置代理程序 447
更改用户帐户 447
代理程序的配置参数 447
配置 Microsoft .NET 监视 449
使用本地帐户或域帐户运行代理程序所需的许可权 450
注册数据收集器 450
使用 .NET 代理程序的“IIS 响应时间”模块 451
启用事务跟踪和诊断数据收集 453
使用 configdc 命令来启用诊断数据的收集 453
在代理程序共存环境中启用事务跟踪 454
激活配置更新 455
数据收集器的性能调整 455
配置 Microsoft Office 365 监视 458
验证已配置的用户的可达性 458
在 Windows 系统上配置代理程序 459
使用静默响应文件来配置代理程序 460
更改用户帐户 461
监视 Skype QoS 461
配置本地环境变量 462
配置 Microsoft SharePoint Server 监视 463
更改用户帐户 464
通过非管理员用户来运行 Monitoring Agent for Microsoft SharePoint Server 464
本地安全策略许可权 465

配置 Microsoft SQL Server 监视 465
创建用户和授予许可权 466
本地环境变量 470
代理程序的配置参数 474
在 Windows 系统上配置代理程序 475
在 Linux 系统上配置代理程序 485
使用静默响应文件来配置代理程序 486
在集群环境中运行代理程序 487
使用集群实用程序来配置代理程序 489
为 ERRORLOG 文件配置多个校对 490
配置 MongoDB 监视 491
使用缺省设置配置代理程序 494
使用静默响应文件来配置代理程序 494
通过对提示做出响应来配置代理程序 495
配置 MySQL 监视 496
在 Windows 系统上配置代理程序 496
在 Linux 系统上配置代理程序 497
使用静默响应文件来配置代理程序 498
配置 NetApp Storage 监视 498
下载和安装 NetApp 可管理性 SDK JAR 文件 499
在 Windows 系统上配置代理程序 500
使用静默响应文件来配置代理程序 500
通过对提示做出响应来配置代理程序 501
数据提供程序的配置参数 502
OnCommand Unified Manager 的配置参数 503
OnCommand API Service 的配置参数 503
配置 Node.js 监视 504
配置 Node.js 代理程序 505
针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器 509
为本地应用程序配置独立 Node.js 数据收集器 513
为 Kubernetes 应用程序配置独立 Node.js 数据收集器 517
配置 OpenStack 监视 521
配置 OpenStack 代理程序 521
启用流程相关信息收集和 SSH 连接 523
添加配置值 524
配置 Oracle 数据库监视 526
在 Windows 系统上配置代理程序 527
通过对提示做出响应来配置代理程序 530
使用静默响应文件来配置代理程序 533
为 Oracle 数据库代理程序用户授予特权 536
配置操作系统监视 538
以非 root 用户身份运行操作系统代理程序 539
配置操作系统代理程序日志文件监视 540
配置操作系统代理程序定制脚本编制 560
配置 Linux OS Agent 文件系统数据收集 566
配置 PHP 监视 566
配置 PostgreSQL 监视 568
在 Windows 系统上配置代理程序 569
在 Linux 系统上配置代理程序 570
使用静默响应文件来配置代理程序 570
配置 Python 监视 571
针对 IBM Cloud 应用程序配置 Python 数据收集器 572
为本地应用程序配置 Python 数据收集器 576
配置 RabbitMQ 监视 580
在 Windows 系统上配置代理程序 581
在 Linux 系统上配置代理程序 581
使用静默响应文件来配置代理程序 582
代理程序的配置参数 582

配置响应时间监视 583
查看事务仪表板 583
Response Time Monitoring 组件 584
规划安装 584
规划配置 586
JavaScript 注入 586
在 Windows 上重新配置 Response Time Monitoring 587
在 AIX 和 Linux 上重新配置 Response Time Monitoring 588
使用“代理程序配置”页面进行配置 588
添加应用程序 590
配置 IBM HTTP Server 响应时间模块 591
Packet Analyzer 路线图 599
重新配置以从 IBM HTTP Server Response Time 模块切换为 Packet Analyzer 607
定制最终用户事务位置值 607
跟踪其他 Web 应用程序 608
为 Response Time Monitoring 代理程序指定唯一的受管系统名称 611
配置 Ruby 监视 612
配置 Ruby 代理程序 612
针对 IBM Cloud 应用程序配置 Ruby 数据收集器 618
配置 SAP 监视 621
在 Windows 系统上配置代理程序 622
在 Linux 或 AIX 系统上配置代理程序 623
使用静默响应文件来配置代理程序 624
代理程序的配置参数 624
根据“Managed System Name”长度限制对 SAP 主机名进行剪裁 627
在 SAP 系统上导入 ABAP 传输 627
从 SAP 系统删除 ABAP 传输 632
验证代理程序配置 633
添加数据库通信端口号 636
SAP 代理程序的高级安装和配置 637
配置 SAP HANA 数据库监视 648
配置 SAP NetWeaver Java Stack 监视 650
在 Windows 系统上配置代理程序 651
在 Linux 或 AIX 系统上配置代理程序 651
使用静默响应文件来配置代理程序 652
配置数据收集器 653
启用事务跟踪和诊断数据收集 654
移除数据收集器配置 655
复原 SAP NetWeaver Application Server 实例 655
代理程序的配置参数 656
配置 Siebel 监视 656
验证 Siebel 用户帐户 657
启用“按组件统计信息监视” 658
在 Windows 系统上配置代理程序 658
通过对提示做出响应来配置代理程序 662
使用静默响应文件来配置代理程序 663
Siebel 代理程序的配置参数 664
始终受监视的 Siebel 组件日志 666
配置 Sterling Connect Direct 监视 666
在 Windows 系统上配置代理程序 666
在 Linux 系统上配置代理程序 667
使用静默响应文件来配置代理程序 667
代理程序的配置参数 668
配置 Sterling File Gateway 监视 668
安装 B2B REST API 668
在 Windows 系统上配置 Sterling File Gateway 代理程序 669
在 Linux 系统上配置 Sterling File Gateway 代理程序 669
使用静默响应文件来配置 Sterling File Gateway 代理程序 670

在 Linux 上配置用于数据提供程序的代理程序环境变量 671
在 Windows 上配置用于数据提供程序的代理程序环境变量 671
数据提供程序的环境变量 671
B2B API 详细信息的配置参数 672
数据库详细信息的配置参数 673
Java API 的配置参数 673
配置 Sybase 服务器监视 673
授予许可权 674
使用命令行界面来配置代理程序 675
使用静默响应文件来配置代理程序 677
为查询禁用 dirty reads 678
配置 Synthetic Playback 监视 679
为 Synthetic Playback 代理程序 启用上游代理支持 680
配置 Tomcat 监视 681
使用缺省设置来配置 Tomcat 代理程序 681
在 Windows 系统上配置代理程序 682
在 Linux 系统上配置 Tomcat 代理程序 685
使用静默响应文件来配置 Tomcat 代理程序 685
启用事务跟踪和诊断数据收集 686
更新或更改 Tomcat 应用程序服务器 687
配置 VMware VI 监视 687
缩放和规划 VMware VI 代理程序 部署 688
启用与 VMware VI 数据源的 SSL 通信 689
在 Windows 系统上配置代理程序 690
使用静默响应文件来配置代理程序 690
通过对提示做出响应来配置代理程序 691
配置数据源的参数 692
配置数据提供程序的参数 693
增加 Java 堆大小 693
配置 WebLogic 监视 694
在 Windows 系统上配置代理程序 696
通过对提示做出响应来配置代理程序 699
使用静默响应文件来配置代理程序 700
WebLogic 代理程序的配置参数 702
配置 WebLogic 代理程序的事务跟踪 702
配置应用程序性能仪表板以显示 WebLogic 代理程序的事务跟踪数据 708
配置 WebSphere Applications 监视 709
为 WebSphere Applications 代理程序 配置数据收集器 709
为本地应用程序配置 Liberty 数据收集器 749
为 IBM Cloud 应用程序配置 Liberty 数据收集器 752
高级数据收集器配置 757
配置 WebSphere Applications 代理程序以监视 WebSphere Extreme Scale 783
配置 WebSphere Infrastructure Manager 监视 790
配置 WebSphere MQ 监视 791
授权用户标识运行代理程序 792
配置 IBM MQ (WebSphere MQ) 以启用数据 793
配置 WebSphere MQ 代理程序 795
为多个队列管理器指定唯一受管系统名称 798
配置 WebSphere MQ 代理程序的事务跟踪 799
为队列和通道长期历史记录启用数据收集 800
为 IBM MQ 的队列管理器启用队列统计信息监视 801
远程监视 MQ 设备上的队列管理器 801
远程监视 MQ 设备上的 HA 队列管理器 802

第 8 章与其他产品和组件集成 807
与 Cloud Event Management 集成 807
与 IBM Tivoli Monitoring V6.3 集成 807

代理程序共存 808
Hybrid Gateway 811
与 OMEGAMON 集成 820
与 Netcool/OMNIbus 集成 821
安装和配置 Integration Agent for Netcool/OMNIbus 821
为 Netcool/OMNIbus 配置集成 823
与 Operations Analytics - Log Analysis 集成 825
与 Operations Analytics - Predictive Insights 集成 825
与 Alert Notification 集成 826
与 Control Desk 集成 827
与 IBM Cloud 集成 828
与 IBM Agent Builder 集成 828

第 9 章管理 829
启动 Cloud APM 控制台 829
阈值和资源组 830
背景信息 830
资源组管理器 833
教程：定义阈值 835
教程：定义阈值以便在受管资源上运行命令 837
阈值管理器 838
定制事件以转发到 EIF 接收器 841
发送电子邮件以响应事件 847
使用“资源组管理服务”API 847
使用“阈值管理服务”API 848
管理用户访问权 850
角色和许可权 851
访问和使用基于角色的访问控制服务 API 858
管理代理程序 859
以非 root 用户身份启动代理程序 859
事务监视的事件阈值 860
管理操作系统代理程序事件 863
使用 Web 站点监视管理合成事务和事件 872
最大化代理程序和服务器性能以进行日志文件监视的相关准则 884
Availability Monitoring 887
关于 Availability Monitoring 887
访问 Availability Monitoring 888
创建和配置测试 888
在“监视仪表板”上查看应用程序可用性和性能 898
Availability Monitoring 用量 909
浏览 API 910
高级配置 910
事件电子邮件 912

第 10 章使用仪表板 915
我的所有应用程序 - 应用程序性能仪表板 915
搜索日志文件 916
应用程序 - 应用程序性能仪表板 917
使用“汇总事务拓扑”窗口小部件 920
组和实例 - 应用程序性能仪表板 921
编辑“组件”仪表板组窗口小部件 923
随着时间变化调整和比较度量值 924
查看和管理定制图表和表 925
管理应用程序 930
添加应用程序 930
编辑应用程序 932
删除应用程序 934

查看和移除脱机代理程序 934
事件状态 939
使用 Operations Analytics - Predictive Insights 调查异常 941
定制视图 942
创建和管理定制页面 942
查看定制页面 947
仪表板实用程序 949
复制仪表板 URL 949
设置跟踪 950
锁定 Cloud APM 控制台 951
报告 951
Response Time Monitoring Agent 报告 951
生成 Synthetic Playback 代理程序报告 955
WebSphere Applications 代理程序报告 958

第 11 章升级 961
升级代理程序 961
保留代理程序配置更改 963
AIX 上的代理程序：在升级前停止代理程序并运行 **slibclean** 964
AIX 上的 HMC Base 代理程序：以非 root 用户身份停止代理程序并运行 **slibclean**，然后再进行升级 964
Node.js 代理程序：升级之前移除数据收集器插件 965
Response Time Monitoring 代理程序：升级 IBM HTTP Server 响应时间模块 965
Microsoft .NET 代理程序：升级之前移除 .NET 数据收集器 966
OpenStack 代理程序：重新配置代理程序实例以使用 OpenStack 身份 API V3 967
Ruby 代理程序：升级之前移除数据收集器插件 967
WebSphere Applications 代理程序：迁移数据收集器 968
Tomcat 代理程序：升级 Windows 上的 TEMA 核心框架 971
升级数据收集器 971

第 12 章故障诊断和支持 973
对代理程序进行故障诊断 973
Internet Service Monitoring 973
Microsoft Active Directory 监视 974
Microsoft IIS 监视 974
Microsoft .NET 监视 975
Microsoft SharePoint Server 监视 975
PostgreSQL 监视 975
为 IBM 支持人员收集监视代理程序日志 976

第 13 章 Agent Builder 979
Agent Builder 概述 979
公共 Agent Builder 过程 980
数据源和数据集 981
监视多个服务器或一个服务器的多个实例 982
测试、安装和配置代理程序 982
操作系统需求 983
特定于 IBM Tivoli Monitoring 的功能部件 983
安装和启动 Agent Builder 983
安装和运行 Agent Builder 的先决条件 984
安装 Agent Builder 984
启动 Agent Builder 986
在 Agent Builder 中设置缺省浏览器 986
在 Agent Builder 中设置缺省时间戳记服务机构 987
卸载 Agent Builder 987
静默卸载 988
创建代理程序 988

对代理程序命名和配置 ... 988
定义初始数据源 ... 989
使用 Agent Editor 修改代理程序 ... 990
缺省操作系统 ... 991
自描述代理程序 ... 992
环境变量 ... 992
看守程序信息 ... 999
Cognos 信息 ... 999
“生成代理程序向导”链接 ... 1000
“**数据源定义**”页面 ... 1000
“**运行时配置信息**”页面 ... 1001
“代理程序 XML 编辑器”页面 ... 1001
保存编辑和更改 ... 1001
落实代理程序的版本 ... 1002
为代理程序设置新版本号 ... 1003
更改产品代码 ... 1003
编辑数据源和特性属性 ... 1003
创建、修改和删除属性 ... 1004
过滤属性组 ... 1011
Formula Editor ... 1011
公式运算符和函数 ... 1015
指定操作系统 ... 1021
配置和调整数据收集 ... 1021
定义和测试数据源 ... 1025
监视进程 ... 1026
监视 Windows 服务 ... 1028
监视来自 Windows Management Instrumentation (WMI) 的数据 ... 1030
监视 Windows 性能监视器 (Perfmon) ... 1032
监视来自简单网络管理协议 (SNMP) 服务器的数据 ... 1033
监视来自简单网络管理协议事件发送方的事件 ... 1037
监视 Java 管理扩展 (JMX) MBean ... 1042
监视来自公共信息模型 (CIM) 的数据 ... 1058
监视日志文件 ... 1060
监视 AIX 二进制日志 ... 1069
监视 Windows 事件日志 ... 1070
监视命令返回码 ... 1072
监视脚本的输出 ... 1076
监视来自 Java 数据库连接 (JDBC) 的数据 ... 1080
使用 Ping 来监视系统可用性 ... 1086
监视 HTTP 可用性和响应时间 ... 1088
监视来自 SOAP 或其他 HTTP 数据源的数据 ... 1094
使用套接字来监视数据 ... 1102
使用 Java API 来监视数据 ... 1110
根据现有来源创建数据集 ... 1121
连接两个属性组 ... 1121
处理连接式属性组中的属性 ... 1124
连接式属性 ... 1124
创建过滤属性组 ... 1125
创建导航器组 ... 1126
使用子节点 ... 1127
创建子节点 ... 1133
子节点配置 ... 1134
定制代理程序配置 ... 1143
使用 Agent Editor 更改配置属性 ... 1145
配置 Windows 远程连接 ... 1145
创建具有 Windows Management Instrumentation (WMI) 权限的用户 ... 1146
配置安全 Shell (SSH) 远程连接 ... 1148
创建工作空间、“执行操作”命令和情境 ... 1149

创建情境、“执行操作”命令和查询 1149
创建工作空间 1149
为 Cloud APM 准备代理程序 1154
在 Agent Builder 中测试代理程序 1156
属性组测试 1156
代理程序全面测试 1159
测试环境变量 1162
在监视基础结构中安装代理程序以供测试和使用 1163
安装代理程序 1163
生成及安装代理程序之后的结果 1170
卸载代理程序 1177
导入应用程序支持文件 1179
导出和导入 Tivoli Enterprise Monitoring Agent 的文件 1179
导出和导入 Tivoli System Monitor Agent 的文件 1179
事件过滤和摘要 1180
控制重复事件 1181
在 Tivoli Enterprise Portal 中查看事件过滤和摘要 1181
故障诊断和支持 1186
共享项目文件 1186
共享解决方案安装程序项目 1187
命令行选项 1187
命令 - **generatelocal** 1188
命令 - **generatemappingfile** 1189
命令 - **generatezip** 1189
属性参考 1190
可用性节点 1190
Performance Object Status 节点 1195
Thread Pool Status 属性组 1200
“事件日志”属性节点 1204
日志文件摘要 1206
“AIX 二进制日志”属性组 1208
“监视器和通知”属性组 1211
“SNMP 事件”属性组 1220
“JMX 事件”属性组 1221
Ping 属性组 1223
HTTP 属性组 1225
发现属性组 1230
Take Action Status 属性组 1232
“日志文件状态”属性组 1235
“日志文件正则表达式统计信息”属性组 1239
为现有代理程序创建应用程序支持扩展 1243
创建“应用程序支持扩展”项目 1243
将支持文件添加至项目 1243
生成 Application Support Extension 安装映像 1244
安装 Application Support Extension 1244
将**解决方案安装项目**转换为“应用程序支持扩展”项目 1245
生成 Cognos 数据模型 1245
生成 Cognos 数据模型的先决条件 1245
创建报告 1249
ICU 正则表达式 1259
创建非代理程序文件捆绑包 1263
远程部署捆绑包编辑器 1264
向捆绑包添加命令 1264
向捆绑包添加必备项 1265
向捆绑包添加文件 1265
生成捆绑包 1265
为 Tivoli Netcool/OMNIbus 探测器创建可部署捆绑包 1266

动态文件名支持....1266
SNMP 陷阱配置....1269
“执行操作”命令参考....1272
SSHEXEC 操作....1273

辅助功能....1275

声明....1277
商标....1278
产品文档的条款和条件....1278
IBM 网上隐私声明....1279

第 1 章 新增功能

最新发行版中提供了新增功能部件、功能和覆盖范围。

· 有关每个发行版或刷新中代理程序版本的信息，请参阅第 44 页的『变更历史记录』。

2019 年 12 月

新代理程序

MariaDB 代理程序

Monitoring Agent for MariaDB 提供针 MariaDB 环境或应用程序的集中管理点。该软件提供了全面的方法来收集所需信息，以便尽早检测到问题并加以防范。 整个系统的信息都是标准化的。您可以从单个控制台监视多个服务器。 通过使用 Monitoring Agent for MariaDB，您可以轻松地收集和分析特定于 MariaDB 的信息。
有关在安装之后配置代理程序的信息，请参阅第 418 页的『配置 MariaDB 监视』

扩展了代理程序的平台支持

现在支持以下代理程序和平台：

Solaris X86-64

· Oracle Database 代理程序

· WebLogic 代理程序

代理程序增强功能

Cassandra 代理程序

在集群详细信息、节点统计信息和键空间详细信息属性组中，添加了两个名为代理程序主机名和代理程序实例名称的新属性。

Db2® 代理程序

添加了支持以监视当前正在运行的 SQL。

IBM Integration Bus 代理程序

在集成服务器状态 - 详细信息页面中，添加了两个新的组窗口小部件 TCPIP 客户机连接和 TCPIP 服务器连接。

Internet Service Monitoring

· 添加了两个新的配置面板变量：

– *Active*：用于选择概要文件元素的状态：active 或 inactive。

– *sniServerName*：指示需要来自启用 SNI 的 Web 服务器的证书的主机/服务器的名称。

· HTTP、HTTPS 和 DNS 监视器的**数据验证**选项卡下的**缺省设置**现在可编辑

· 现在，代理程序在 HTTP 和 HTTPS 监视器的**页面**字段中支持 & 字符

· 现在，代理程序在 HTTP 和 HTTPS 监视器的 **regex** 字段中支持丹麦语字符

注: 在要使用此功能的代理程序安装之前，在 Linux 平台上将语言环境设置为 da_DK

Microsoft Active Directory 代理程序

· 在状态概述页面中，添加了新的名为 KCC 详细信息的窗口小部件。

· 在“属性详细信息”选项卡中添加了以下新属性组：

– 目录服务

– Kerberos 一致性检查程序

– Kerberos 密钥分发中心

– 名称服务提供程序

– 交换目录服务

© Copyright IBM Corp. 2014, 2019

Microsoft .NET 代理程序

在数据库调用详细信息属性组中添加了名为请求名称的新属性。此属性显示触发数据库查询的请求的名称。

Microsoft Exchange Server 代理程序

- 在状态概述页面中，添加了新的名为传输 SMTP 接收的窗口小部件。
- 在“属性详细信息”选项卡中添加了以下新属性组：
  - MS Exchange AB
  - MS Exchange ADAccess 进程
  - MS Exchange ADAccess 高速缓存
  - MS Exchange ADAccess 域控制器
  - MS Exchange ADAccess 森林发现

Microsoft Hyper-V Server 代理程序

已添加针对 Windows Server 2019 的支持。

Microsoft IIS 代理程序

- 添加了新的组窗口小部件：
  - 系统主内存统计信息
  - IIS 服务器 - 指定的内存使用情况
  - IIS 服务器 - 指定的 CPU 使用率
  - 工作程序进程详细信息
  - .Net 内存管理
- 在工作程序进程详细信息组窗口小部件中的每个应用程序池名称上，创建一个页面，显示“每秒处理的请求数”、“耗用时间”、“队列中的请求数”、“内存”和“CPU 使用率”的历史趋势。
- 在 .Net 内存管理组窗口小部件中的每个应用程序池名称上，添加一个弹出窗口，显示 GC 中百分比时间的历史趋势。

Microsoft SharePoint Server 代理程序

- 添加了名为 Trace_Log 的新属性组，其提供高严重性日志信息。
- 在“概述”页面中添加了两个新的名为跟踪日志详细信息和最近 1 小时跟踪日志计数的组窗口小部件，其中显示最近 100 个跟踪日志事件的详细信息以及最近 1 小时意外和高级别跟踪日志的计数。

MySQL 代理程序

代理程序在服务器重新启动后继续收集数据。

NetApp Storage 代理程序

代理程序现在显示映射到卷的 Qtress 的确切列表。

PostgreSQL 代理程序

代理程序现在支持 PostgreSQL Server V12。

Response Time Monitoring 代理程序

- 添加了新的配置参数 KT5AARIPTOUSERID。此参数允许在 AAR 原始数据的“用户名”属性中保存客户机 IP 地址。缺省情况下，其设置为 NO。要更改设置，需要重新启动 Response Time Monitoring 代理程序。
  - KT5AARIPTOUSERID=NO：如果值为 NO，那么 Response Time Monitoring 代理程序 代理程序会将事务的用户名保存到 AAR 的用户标识属性。
  - KT5AARIPTOUSERID=YES：如果值为 YES，那么 Response Time Monitoring 代理程序 会将事务的源 IP 地址保存到 AAR 的用户标识属性。
- Response Time Monitoring 代理程序 现在支持在静默配置中指定 KT5AARIPTOUSERID 值。

- 现有组窗口小部件按用户排序最差项 - 排名前 5 位的标题更改为按用户排序最差项 - 排名前 20 位。组窗口小部件更改为显示在选中时间段内事务故障百分比最高的前 20 个用户。

VMware

- 代理程序现在支持从 vSphere API 调用访存 vCenter 的 IP 地址或主机名，代替显示 `Configured Address`，因为其源自于配置面板。用户可以通过将代理程序环境中的标志设置为 Y 来激活此功能。例如，`KVM_RETRIEVE_HOSTNAME_FROM_API=Y`。
- 现在，可以为重试计数指定限制，以限制与数据源的连接尝试。例如，`KVM_DATA_PROVIDER_CONNECTION_RETRY_COUNT=1000`，如果与 vCenter 的连接失败，那么在代理程序环境文件中添加此变量会对连接重试计数添加锁定。1000 指示代理程序将尝试直至 1000 次连续不成功连接尝试，然后将关闭数据提供程序进程，并记录以下日志消息：`NO MORE ATTEMPTS OF CONNECTION;STOPPING THE DATA COLLECTION, TO RESUME MONITORING PLEASE RESTART THE AGENT. TO HAVE MORE ATTEMPT OF CONNECTIONS, RESET THE VALUE OF THE VARIABLE KVM_DATA_PROVIDER_CONNECTION_RETRY_COUNT`。重试连接尝试的缺省值为 6，用户可根据需求设置期望的阈值。
- 代理程序支持配置特定于实例的堆大小以有效地利用系统上分配的内存。例如，`KVM_CUSTOM_JVM_ARGS= -Xmx512m`，在实例的环境文件中设置此变量意味着，实例配置为使用 512 MB 堆内存。可根据实例监视的 vCenter 对象的总计数来更改大小。

2019 年 9 月

扩展了代理程序的平台支持

现在支持以下代理程序和平台：

Solaris X86-64

- Db2 代理程序
- SAP 代理程序
- Sybase 代理程序
- UNIX OS 代理程序
- WebSphere® Applications 代理程序
- WebSphere MQ 代理程序
- IBM Integration Bus 代理程序

RHEL on x86-64（64 位）

- Internet Service Monitoring 代理程序
- Microsoft SQL Server 代理程序
- Sybase 代理程序

RHEL on POWER Little Endian (ppc64le)

- RabbitMQ 代理程序

代理程序增强功能

Db2 代理程序

- 代理程序现在支持 Db2 服务器 V11.5。

Hadoop 代理程序

- 在配置面板中添加了一个新配置参数 **`Unique Cluster Name`**，这对于 Hadoop 集群是一个唯一名称，用于指示其版本和风格。
- Hadoop 代理程序现在可显示对应于 Ambari 服务上创建的阈值的项。
- Hadoop 代理程序现在支持监视 Hadoop 集群中的 Streaming Analytics Manager 服务。
- Hadoop 代理程序现在支持监视 Hadoop 集群中的 Schema Registry 服务。

HTTP Server 代理程序

· 代理程序现在支持 Solaris Sparc 上的 Oracle HTTP 服务器。

Internet Service Monitoring

· 代理程序现在支持 IBM Tivoli® Netcool/OMNIbus。

· 代理程序已增强，可从现有概要文件中删除概要文件并重命名现有概要文件。

MongoDB 代理程序

· 代理程序现在支持 MongoDB 数据库 V4.x。

MySQL 代理程序

· 在“帮助”部分中针对“应用程序可用性”添加了 FQDN 属性。

· 已修复针对“IP 地址代理程序配置”参数显示的工具提示。

· 已添加以下新属性用于在 IBM Cloud App Management 中进行监视。

– 数据库大小信息
– 错误信息
– Db 实例锁定计数
– 用户连接详细信息
– 进程列表详细信息
– 事件信息

PostgreSQL 代理程序

· 已添加两个名为 `Deadlocks_Count_Crit` 和 `Deadlocks_Count_Warn` 的新情境，用于监视数据库中的死锁数量，这将有助于您解决死锁问题。

· 已添加一个名为 `Deadlocks_Info` 的新属性组，用于检查死锁详细信息。

Sybase 代理程序

· 在“帮助”部分中针对“应用程序可用性”添加了 FQDN 属性。

Synthetic Playback 代理程序

· 现已支持 Firefox V68.0 ESR。

· 现已支持系统代理、PAC 代理和无代理配置。

Tomcat 代理程序

· 代理程序已增强，现已包含度量和 UI 视图，用于监视 JVM 的堆/非堆内存池使用情况。

· 代理程序已增强，现已包含度量和 UI 视图，用于监视 JVM 的线程和类装入信息。

· 代理程序 UI 现已在“服务器信息”视图上显示 FQDN。

UNIX OS 代理程序

· 代理程序现已更新，已包含定制脚本编制功能部件。Shell 脚本、PERL 脚本和其他类型的脚本均可供使用。

VMware VI 代理程序

· 已添加名为 **KEY_STORE_PASSWORD** 的新配置字段。它允许用户使用针对代理程序 JRE 设置的新密钥库密码来配置代理程序。

2019 年 6 月

扩展了代理程序的平台支持

现在支持以下代理程序和平台：

Red Hat Enterprise Linux (RHEL) 8

以下代理程序和数据收集器现在支持 RHEL 8。在 RHEL 8 上安装代理程序前，请务必阅读第 109 页的『Linux 系统上的安装前任务』的第 109 页的『特定操作系统』部分。

RHEL 8 on x86-64（64 位）

- Cassandra 代理程序
- Cisco UCS agent
- DataPower® 代理程序
- DataStage® 代理程序
- Db2 代理程序
- Hadoop 代理程序
- HTTP Server 代理程序
- Integration Agent for Netcool®/OMNIbus
- Internet Service Monitoring 代理程序
- J2SE 数据收集器
- Linux KVM 代理程序
- Linux OS 代理程序
- MongoDB 代理程序
- MQ Appliance 代理程序
- MySQL 代理程序
- NetApp Storage 代理程序
- Node.js 数据收集器
- PHP 代理程序
- Python 数据收集器
- PostgreSQL 代理程序
- RabbitMQ 代理程序
- Response Time Monitoring Agent
- Ruby 代理程序
- SAP 代理程序
- SAP HANA Database 代理程序
- SAP NetWeaver Java™ Stack 代理程序
- Sterling Connect Direct 代理程序
- Sterling File Gateway 代理程序
- Sybase 代理程序
- Tomcat 代理程序
- VMware VI 代理程序
- WebSphere Applications 代理程序
- WebSphere MQ 代理程序

RHEL 8 on System z

- Db2 代理程序
- Hadoop 代理程序
- Linux OS 代理程序
- MySQL 代理程序
- Node.js 数据收集器

- Python 数据收集器
- Response Time Monitoring Agent
- WebSphere Applications 代理程序
- WebSphere MQ 代理程序

RHEL 8 on POWER Little Endian (ppc64le)

- Db2 代理程序
- Hadoop 代理程序
- J2SE 数据收集器
- Linux OS 代理程序
- MySQL 代理程序
- Node.js 数据收集器
- SAP NetWeaver Java Stack 代理程序
- WebSphere Applications 代理程序
- WebSphere MQ 代理程序

Solaris Sparc 10 和 11

- JBoss 代理程序
- Oracle Database 代理程序
- WebLogic 代理程序

Windows Server 2019

- WebSphere Applications 代理程序

代理程序增强功能

Hadoop 代理程序

- Hadoop 代理程序 现已支持监视 HDF 3.3（含 HDP 3.1.0）Ambari 服务 Big SQL 6.0。
- Hadoop 代理程序 现已支持 x86-64 平台上的 SUSE Linux Enterprise Server (SLES) 15。

HMC Base 代理程序
HMC Base 代理程序 支持 HMC V9.1。

Integration Agent for Netcool/OMNIbus
代理程序已更新以支持 Red Hat Enterprise Linux (RHEL) 8 和 SUSE Linux Enterprise Server (SLES) 15

Internet Service Monitoring 代理程序
代理程序现已具有 Service Assurance 代理程序 监视器，用于监视 Cisco Service Assurance Agent 探测器。

Microsoft IIS 代理程序
代理程序已增强，具有针对 Windows 2019 服务器的容错功能。该增强功能会为 Windows 2019 服务器上安装的代理程序显示 FTP 站点数据。

MongoDB 代理程序
代理程序现已支持 Red Hat Enterprise Linux (RHEL) 8 on x86-64（64 位）平台。

Python 数据收集器 ifix02
数据收集器现已支持 Django 1.10 和更高版本。

SAP 代理程序

SAP 代理程序 现已支持以下平台：

- Red Hat Enterprise Linux (RHEL) 8 on x86-64（64 位）
- SAP NetWeaver Application Server 7.52 (SAP Basis 752)

SAP HANA Database 代理程序

SAP HANA Database 代理程序 现已增强，具有以下功能：

- 在 SAP HANA Database 代理程序 的 :HDB Subnode 节点中已添加主机名作为其唯一标识。
- 代理程序现已支持向外扩展体系结构。
- Red Hat Enterprise Linux (RHEL) 8 on x86-64（64 位）和 Linux ppc64le 平台。
- SUSE Linux Enterprise Server (SLES) 15 on x86-64（64 位）平台。
- 在“系统数据库”属性组下已添加新属性“已删除的属性”。

SAP NetWeaver Java Stack 代理程序

SAP NetWeaver Java Stack 代理程序 支持以下平台：

- Red Hat Enterprise Linux (RHEL) 8 on x86-64（64 位）
- SUSE Linux Enterprise Server (SLES) 15 on x86-64（64 位）
- Windows Server 2019 DE 和 SE
- Windows Server 2016 DE 和 SE

Synthetic Playback 代理程序

- 支持 Selenium IDE 3.2.X、3.3.X 或 3.5.X 记录的 .side 脚本
- 支持 Firefox ESR 60.5.1 的回放
- 支持 Selenium IDE 命令：**`wait`**、**`flow control`** 和 **`linkText locator type`**

Skype for Business Server 代理程序

Skype for Business Server 代理程序 现已增强，具有以下功能：

- 代理程序现已支持 Skype for Business Server 2019。
- 在“概述”页面上添加了两个新的组窗口小部件（例如，`Database-Throttled Requests(DBStore)` 和 `Database-Throttled Requests(SHAREDDBStore)`），用于显示由于 DBstore 和共享 DBstore 的数据库队列等待时间过长而导致 Skype for Business Server 发出的调速请求数。

先决条件扫描程序

`IGNORE_PRECHECK_WARNING` 命令现已作为 **`SKIP_PRECHECK`** 命令的替代方法以供使用。有关更多信息，请参阅第 123 页的『绕过先决条件扫描程序』。

文档增强功能

已创建一个页面，用于帮助您快速查找每个代理程序和数据收集器的版本信息和变更历史记录。请参阅第 44 页的『变更历史记录』。

2019 年 3 月

扩展了代理程序的平台支持

现在支持以下代理程序和平台：

Windows Server 2019

- Cassandra 代理程序
- DataStage 代理程序
- Db2 代理程序
- Hadoop 代理程序
- Internet Service Monitoring
- Microsoft Active Directory 代理程序
- Microsoft Cluster Server 代理程序

- Microsoft IIS 代理程序
- Microsoft Exchange Server 代理程序
- Microsoft SQL Server 代理程序
- MySQL 代理程序
- PostgreSQL 代理程序
- RabbitMQ 代理程序
- SAP 代理程序
- SAP HANA Database 代理程序
- Sybase 代理程序
- Tomcat 代理程序
- Windows OS 代理程序

Solaris SPARC 10 和 11

- Db2 代理程序
- HTTP Server
- MySQL 代理程序
- SAP 代理程序
- Sybase 代理程序
- UNIX OS 代理程序
- WebSphere Applications 代理程序

Monitoring Agent for Cassandra

Cassandra 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 操作系统的支持。
- 已添加针对故障诊断的详细日志记录。

Monitoring Agent for Db2

Db2 代理程序现已增强，具有以下功能：

- Db2 代理程序现已支持 Windows Server 2019。
- Db2 代理程序现已支持 Solaris SPARC 10/11 平台。

Monitoring Agent for Hadoop

Hadoop 代理程序 现已增强，具有以下功能：

- 已添加相应的支持，用于监视启用 SSL 的 Hadoop BigInsights®、Hortonworks 和 Cloudera 集群。
- 已添加支持用于测试到启用 SSL 的 Hadoop 集群的连接。
- 已添加针对 Windows Server 2019 操作系统（Datacenter 版本和 Standard 版本）的支持。
- 已添加监视 Hadoop 产品的支持：Cloudera 6.1.1 (CDH 6.1.1)。
- 已添加监视 Hadoop 产品的支持：Hortonworks 3.1.0 (HDP 3.1.0)。

Monitoring Agent for IBM Integration Bus

IBM Integration Bus 代理程序 现已增强，具有以下功能：

- 已添加容错支持，用于监视 IBM App Connect Enterprise V11。有关更多信息，请参阅第 238 页的『配置 IBM Integration Bus 代理程序』。

Monitoring Agent for Microsoft Internet Information Services

Microsoft IIS 代理程序 现已支持 Windows Server 2019 操作系统。

Monitoring Agent for InfoSphere® DataStage

DataStage 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 操作系统的支持。
- 已向数据收集查询添加查询超时，以改进代理程序性能。

Monitoring Agent for Microsoft Active Directory

Microsoft Active Directory 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 的支持。
- 已添加新属性组 AD_Services_Status，以提供与 Active Directory 服务器相关的服务状态。根据服务状态，它可确定 Active Directory 的服务器状态。
- 已添加新的情境 AD_Server_Status 用于监视 Active Directory 服务器状态。
- 已添加新属性组 Root_Directory_server 用于提供活动版本和受监视的操作系统名称。

Monitoring Agent for Microsoft Cluster Server

Microsoft Cluster Server 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 操作系统的支持。
- 已添加 CLUSTER_SERVICE_VERSION 属性。

Monitoring Agent for Microsoft Exchange Server

Microsoft Exchange Server 代理程序 现已增强，具有以下功能：

- 已添加针对 MS Exchange Server 2019 的支持。
- 已添加新属性组 MSExchange MAPIoverHTTP 以提供有关 MAPI over HTTP 协议统计信息的信息。

Monitoring Agent for Internet Services

Internet Service Monitoring 代理程序现已增强，具有以下功能：

- 已添加针对 LDAP、NTP、NNTP、SOAP、SNMP、SIP、RTSP、RPING、RADIUS 和 TFTP 监视器的支持。
- 已添加针对 Windows Server 2008 R2 操作系统和 Windows Server 2019 的支持。

Monitoring Agent for Microsoft SQL Server

Microsoft SQL Server 代理程序 现已支持 Windows Server 2019。

Monitoring Agent for MySQL

Monitoring Agent for MySQL 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 的支持。
- 已添加针对 Solaris SPARC 10/11 平台的支持。
- 已添加设置额外属性的功能，用于代理程序启动的与 MySQL 服务器之间的 JDBC 连接。

Monitoring Agent for PostgreSQL

PostgreSQL 代理程序 现已支持 Windows Server 2019 操作系统。

Monitoring Agent for RabbitMQ

RabbitMQ 代理程序 现已支持 Windows Server 2019 操作系统。

Monitoring Agent for Skype for Business Server

Skype for Business Server 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 操作系统的支持。
- 已添加名为 KQL_Server 的新属性组，以显示 Skype for Business Server 产品相关信息。
- 已添加名为 Skype_Server_Down 的新情境，以基于服务器前端和 IM 会议服务状态来监视 Skype for Business Server 状态。

Monitoring Agent for SAP Applications

SAP 代理程序 现已增强，具有以下功能：

- 已为 SAP 代理程序和 SAP 服务器之间使用的应用程序用户添加区分大小写密码功能。
- 已添加针对 Windows Server 2019 操作系统（Datacenter 版本和 Standard 版本）的支持。
- 已添加支持以查看 SAP 系统中存在的长时间运行的作业（超过 24 小时）。
- 已改进 /IBMMON/ITM_MAIALRT_INX 功能模块的性能。
- 已添加针对 Solaris v10 和 v11 SPARC 操作系统的支持。
- 已向 SAP 主机名添加删除功能，以与受管系统名称的最多 32 个字符的限制相匹配。

Monitoring Agent for SAP HANA Database

SAP HANA Database 代理程序 现已增强，具有以下功能：

- 已添加相应的支持，以便在租户数据库名称与 HANA 系统 SID 相同时可发现租户数据库。
- 已添加针对 Windows Server 2019 操作系统（Datacenter 版本和 Standard 版本）的支持。

Monitoring Agent for Sybase Server

Sybase 代理程序 现已增强，具有以下功能：

- 已添加针对 Windows Server 2019 的支持。
- 已添加针对 Solaris SPARC 10/11 平台的支持。
- 已增强 Sybase 查询，以提高并发性和减少锁定。

Monitoring Agent for Tomcat

Tomcat 代理程序 现已支持 Windows Server 2019 操作系统（Datacenter 版本和 Standard 版本）。

Monitoring Agent for UNIX OS

UNIX OS 代理程序 现已增强，具有以下功能：

- 已添加针对 Solaris SPARC 10 和 11 的支持。

Monitoring Agent for VMware VI

VMware VI 代理程序 已更新，在显示所有多行图表的图形上的均值趋势时将忽略 unavailable(-1) 值。

Monitoring Agent for WebSphere Applications

WebSphere Applications 代理程序 现已增强，具有以下功能：

- 已添加针对 Solaris SPARC 10 和 11 的支持。
- 已添加监视 WebSphere® Extreme Scale 的支持。您可在属于 Extreme Scale 区域或多个区域的任意服务器的节点下，配置针对该区域或其他多个区域配置的监视功能。您可向下钻取以查看该区域或其他多个区域内的不同服务器、映射集和分区的信息。有关更多信息，请参阅第 783 页的『配置 WebSphere Applications 代理程序以监视 WebSphere Extreme Scale』。

Monitoring Agent for WebSphere MQ

WebSphere MQ 代理程序 现已增强，具有以下功能：

- 已添加针对 SLES 15 xLinux 的支持。
- 已添加相应的支持，以收集队列管理器的统计信息并显示所收集的数据。有关更多信息，请参阅第 801 页的『为 IBM MQ 的队列管理器启用队列统计信息监视』。

数据收集器增强功能

J2SE 数据收集器

J2SE 数据收集器 现已增强，具有以下功能：

- 已添加针对 OpenJDK 9、10 和 11 版本的支持。
- 已添加针对 Windows Server 2019 操作系统（Datacenter 版本和 Standard 版本）的支持。
- 已添加相应的功能，以便自动发现特定于 J2SE 应用程序的类和方法，用于进行事务跟踪和诊断数据监视。

用于合成脚本的 Selenium IDE 3.2.X 和 3.3.X

如果您的预定包含 IBM Website Monitoring on Cloud 附加组件、Selenium IDE V3.2X 和 3.3.X 现已受支持；脚本和测试套件以 .side 格式保存，而不是以较低版本的 Selenium IDE 所使用的 .html 格式进行保存。现有 .html 脚本仍可供使用。在某些情况下，您可编辑 .html 脚本或者使用 .side 格式对其进行重新记录。

有关更多信息，请参阅第 872 页的『使用 Web 站点监视管理合成事务和事件』的下列子主题：第 872 页的『记录合成脚本』、第 874 页的『构造复杂的脚本』和第 875 页的『从先前的 Selenium IDE 版本更新脚本』。

2018 年 12 月

新代理程序

Monitoring Agent for IBM Cloud

Monitoring Agent for IBM Cloud 从 IBM Cloud (Softlayer) 帐户收集虚拟机库存和度量值。使用 IBM Cloud 代理程序 来跟踪已配置并正在 IBM Cloud 中运行的虚拟设备数。您可以在详细仪表板页面中查看分配给每个虚拟设备的资源，该页面还显示诸如设备所在的数据中心、操作系统以及相应月份的预计公共网络带宽之类的信息。

代理程序增强功能

Monitoring Agent for Cassandra

Cassandra 代理程序增强为具有以下功能：

- 添加了名为 Cassandra_Cluster_Down 的新阈值，用于监视受监视实例的状态。
- 添加了对 Ubuntu 18.04 操作系统的支持。
- 已添加针对 SUSE Linux Enterprise Server 15 平台的支持。

Monitoring Agent for Db2

Db2 代理程序现已增强，具有以下功能：

- Db2 代理程序现已支持针对多个备用数据库的 HADR 监视功能。
- Db2 代理程序现在支持“数据库状态”属性的新值 Stopped。
  已停止状态指示数据库不活动并具有零个活动连接，同时其运行状况正常并准备好接受新连接。
- 已添加新的窗口小部件 Db2 Server Information 以显示 Db2 服务器详细信息。
- 添加了新页面 HADR 状态 - 本地数据库以在下列新窗口小部件中显示有关合作伙伴数据库的信息：
  - HADR 数据库详细信息是表窗口小部件，其中显示合作伙伴数据库的重要属性值。
  - 日志间隔（历史记录）是图形窗口小部件，其中显示日志间隔趋势与时间。
  - 备用标志状态是表窗口小部件，其中显示备用标志状态值。
- 添加了新的预定义阈值 UDB_HADR_Aux_Standby_Disconnect 以监视 HADR 环境中的辅助备用数据库。
- “排名前 5 的数据库内存使用情况”窗口小部件更新为显示正确的值。
- Db2 代理程序现已支持以下平台：
  - Ubuntu zLinux 18.04
  - SUSE Linux Enterprise Server 15 on x86-64（64 位）
  - SUSE Linux Enterprise Server 15 for zLinux
  - SUSE Linux Enterprise Server 15 for Power® Linux Little Endian

Monitoring Agent for InfoSphere DataStage

InfoSphere DataStage 代理程序现已增强，具有以下功能：

- 添加了对所选属性组禁用数据收集的功能。
- 优化了“作业运行”属性组的数据收集。

- 已添加针对 SUSE Linux Enterprise Server 15 平台的支持。

Monitoring Agent for Internet Services
Internet Service Monitoring 代理程序现已支持 Windows 64 位平台和 Linux 64 位平台。

Monitoring Agent for Microsoft .NET
.NET 代理程序增强为具有以下功能：

- .NET 代理程序现在跟踪失败请求。这些请求的状态在“中间件事务详细信息”页面的“最新请求”组窗口小部件下显示为 `failed`。此外，“最新错误”组窗口小部件列出最近的失败请求以及状态码和错误描述。
- .NET 代理程序还监视通过 `ASP.NET` 身份和 `ASP.NET` 会话可用的用户数据。用户数据显示在“中间件事务详细信息”页面的“排名前 5 的用户”组窗口小部件中。

Monitoring Agent for MongoDB
MongoDB 代理程序 现已支持 SUSE Linux Enterprise Server 15 平台。

Monitoring Agent for NetApp Storage
使用以下功能增强“NetApp 存储”代理程序：

- 在“事件详细信息”页面上添加了新的搜索框，用于根据搜索条件来过滤事件数据。
- 针对 LUN 添加了新的“详细信息”页面。
- 用户现在能够在“**详细信息**”页面上检查映射到每个存储对象的相关设备的详细信息。

Monitoring Agent for OpenStack
添加了对监视虚拟机实例的支持，例如虚拟机实例 CPU、内存、磁盘和网络接口控制器的使用情况。

Monitoring Agent for PostgreSQL

- 已添加针对 SUSE Linux Enterprise Server 15 的支持
- 优化了对 CPU 和内存属性组的数据收集

Monitoring Agent for RabbitMQ
已添加针对 SUSE Linux Enterprise Server 15 的支持

Monitoring Agent for SAP Applications
SAP 代理程序现在支持以下平台：

- SUSE Linux Enterprise Server 15 平台
- SAP NW RFC SDK 7.50

Monitoring Agent for Skype for Business Server
Skype for Business 服务器增强为具有以下功能：

- 综合事务模块中的综合事务命令现在可由已配置的测试用户执行。要使此功能可用，请在代理程序配置面板中禁用**使用代理程序配置值**，并提供要为其执行综合命令的 池 FQDN 的值。针对代理程序配置面板的“池 FQDN”字段中提供的身份，确保通过 NewCsHealthMonitoringConfiguration 命令来配置测试用户。
- 用户现在可以禁用综合命令。要禁止执行任何特定命令，请针对位置`<CANDLE_HOME>\tmaitm6`（32 位版本）和 `<CANDLE_HOME>\TMAITM6_x64`（64 位版本）存在的 `LyncSyntheticTrans.exe.config` 文件中的命令名称提供 `false`。

Monitoring Agent for Tomcat

- 添加了新属性组“集群”。其中包含集群的属性信息。
- 添加了新窗口小部件“集群信息”。此窗口小部件显示属性组“集群”的信息。如果代理程序监视的是非集群 Tomcat 设置，那么该窗口小部件不会显示任何数据。
- 添加了配置面板变量 *Tomcat* 服务器端口。此变量表示 Tomcat 服务器运行所在的端口。此变量的缺省值为 8080。

Monitoring Agent for VMware VI

- 组件页面已增强为显示所配置的 vCenter 的 IP 地址或主机名及其与代理程序的连接。

WebSphere Infrastructure Manager 代理程序
WebSphere Infrastructure Manager 代理程序现已支持 AIX®。

WebSphere MQ 代理程序
WebSphere MQ 代理程序现已在 IBM WebSphere MQ 9.1 上受支持。

数据收集器增强功能

J2SE 数据收集器
J2SE 数据收集器 现已增强，具有以下功能：

- 已添加针对 SUSE Linux Enterprise Server 11 for Power Linux Big Endian（64 位）的支持。
- 已添加针对 Power Linux Big Endian (pLinux BE)（64 位）的支持。
- 已添加针对 Power Linux Little Endian (pLinux LE)（64 位）的支持。
- 已添加配置模块和 Jetty 服务器监视。

扩展了代理程序的平台支持

现在支持以下代理程序和平台：

SUSE Linux Enterprise Server 15 平台

- Cassandra 代理程序
- DataPower 代理程序
- DataStage 代理程序
- Db2 代理程序
- HTTP Server
- IBM Integration Bus 代理程序
- Linux OS 代理程序
- MongoDB 代理程序
- OpenStack 代理程序
- PostgreSQL 代理程序
- Monitoring Agent for RabbitMQ
- SAP 代理程序
- WebSphere MQ 代理程序

Power Linux

- J2SE 数据收集器

Ubuntu 18.04

- Cassandra 代理程序
- IBM Integration Bus 代理程序
- OpenStack 代理程序
- Linux OS 代理程序
- RabbitMQ 代理程序
- WebSphere MQ 代理程序

Power 9 支持
现在，所有代理程序都支持 Power 9。

V8.1.4 的 2018 年 10 月更新的新增内容

与 Cloud Event Management 集成

Cloud Event Management 提供跨设备、应用程序和体系结构的实时事件管理。现在，通过 Cloud Event Management 和 IBM Cloud Application Performance Management 之间的集成，Cloud APM 中生成的所有事件都发送到 Cloud Event Management。

2018 年 9 月

新代理程序可用

Monitoring Agent for MQ Appliance

MQ 设备代理程序提供特定于 MQ 设备上 MQ 设备级别的监视信息，例如，CPU、内存、存储器、传感器和队列管理器摘要信息。

代理程序增强功能

Monitoring Agent for Db2

Db2 代理程序现已支持 Power Linux Big Endian 操作系统。

Monitoring Agent for Hadoop

- Hadoop 代理程序现在监视另外两个服务的状态：SmartSense 和 Druid。
- Hadoop 代理程序现在支持 Hortonworks Data Platform (HDP) 3.0.0。

Internet Service Monitoring 代理程序代理程序

增强 Internet Service Monitoring 代理程序 代理程序的“编辑”功能。可编辑具有可配置参数的所有监视器。

Monitoring Agent for MySQL

MySQL 代理程序现在支持监视 MySQL v8.0.11。

Monitoring Agent for NetApp Storage

使用以下功能增强“NetApp 存储”代理程序：

- 将在“NetApp 存储实例”页面上添加名为“整体事件摘要”的新的窗口小部件。其显示事件的累计计数。您可以通过单击表示为“事件总数”的状态栏，查看在环境上发生的所有事件，而无论严重性或对象。

  另外，将在每个对象表中添加“事件状态”列，其中显示依次根据时间和严重性级别划分优先级的事件状态。

  “NetApp 存储实例”页面现在显示“事件摘要”表，而不是图表。
- 将更新“汇总详细信息”页面以显示与选中的汇总相关联的相关设备。

Monitoring Agent for SAP Applications

针对 SAP NW RFC SDK 750 添加支持。

Monitoring Agent for SAP HANA Database

添加了两个新功能：

- 可按支持方式监视 SAP HANA 数据库。
- SAP HANA Database 代理程序 支持 Power System 的 Big Endian 平台。

Monitoring Agent for VMware VI

- 现在支持监视 **HostVFlashManager**
- “ESX 服务器仪表板”现在显示根据 CPU 利用率处于“严重”、“警告”和“正常”状态的虚拟机的计数。

新平台：Linux on POWER Big Endian

有新的可用平台。现在，在 Linux on POWER Big Endian 上支持以下代理程序：

- Db2 代理程序
- IBM Integration Bus 代理程序
- Linux OS 代理程序
- SAP HANA Database 代理程序
- WebSphere MQ 代理程序
- WebSphere Applications 代理程序

2018 年 6 月

新代理程序可用

Monitoring Agent for Sybase Server

Sybase 代理程序 提供针对分布式数据库的集中管理点。它收集数据库和系统管理员检查 Sybase 服务器系统性能、提早发现问题并进行阻止所需的信息。

代理程序增强功能

Monitoring Agent for Hadoop

- 已添加监视 Hadoop 服务（例如，Mahout、Atlas 和 Falcon）的支持。
- 已添加监视 Hadoop 产品的支持：Cloudera CDH 5.13。
- 已添加监视 Hadoop 产品的支持：Hortonworks HDP 2.6.4。

Monitoring Agent for HMC Base

支持 HMC V8 R8.7.0。

Monitoring Agent for HTTP Server

已添加针对 64 位 Apache HTTP Server on Windows 的支持。

Monitoring Agent for Microsoft .NET

Monitoring Agent for Microsoft .NET 已增强，如下所示：

- “IIS 响应时间”模块通过对满足以下条件的 ASP.NET Web 表单（.aspx 页面）和 ASP.NET MVC Razor 视图插入 JavaScript，来监视子事务和呈现时间明细：
  - 该页面符合 W3C HTML 标准。
  - 响应头包含 `Content-Type: text/html, application/xml,application/json`。
  - 响应内容包含 *<head>* 元素。
- .NET 代理程序在 APM 服务器上将深入数据上载到诊断查询引擎 (DQE) 服务。DQE 服务深入仪表板快速上载并显示数据。
- 添加了新阈值 **`NET_Slow_IIS_Request_Crit`**，该阈值在“10 大慢速请求”窗口小部件具有响应时间超过 500 毫秒的请求时触发。
- 已更新选择性过滤工具，使用搜索框可从应用程序池列表中搜索应用程序池。
- 已添加 **`ProcListCaller`** 实用程序，来提供已装入 .NET Agent CLR 概要分析程序 (`CorProfLog.dll`) 的进程的列表。

Monitoring Agent for Microsoft SQL Server

- 根据 `koqErrConfig.ini` 文件中的整理设置，Microsoft SQL Server 代理程序 在 ERRORLOG 解析中支持多个整理。当 `koqErrConfig.ini` 文件不包含任何有效整理设置时，您将只能看到缺省英文错误消息且严重性级别高于缺省严重性级别（如果有）。缺省严重性级别为 17。在解析 ERRORLOG 文件时，将考虑 `koqErrConfig.ini` 文件中存在的所有整理。因此，只有正在使用的整理应该在 `koqErrConfig.ini` 文件中添加。ERRORLOG 解析区分大小写，您必须确保 `koqErrConfig.ini` 文件中的整理关键字值与 ERRORLOG 文件中或参考 `koqErrConfigSample.ini` 文件中找到的关键字值完全相同。请注意，`koqErrConfig.ini` 文件中所做的更改在代理程序升级期间不保存，您必须在代理程序升级前备份。
- 该代理程序还提供实用程序工具 **`koqVerifyPermissions.exe`** 来检查现有 SQL Server 用户是否有足够权限来监视 Microsoft SQL Server。如果现有 SQL Server 用户没有足够权限，那么您可以使用实用程序工具 **`permissions.cmd`** 作为备用工具向现有 SQL Server 用户授予数据收集的最低权限。

Monitoring Agent for NetApp Storage

Monitoring Agent for NetApp Storage 已增强，如下所示：

- 已添加新的“组件”页面，用于显示代理程序连接状态的详细信息（即数据提供程序是已启动还是已停止），以及受监视数据源的 IP 地址。各状态栏表示处于紧急、正常、警告或未知状态的节点、聚集、卷和磁盘的数量。
- 已添加新的“NetApp 存储实例”页面，用于突出显示集群、聚集、卷、磁盘和 vServer 的键属性。其还显示包含在环境上的每个可用存储实体或对象上发生的事件计数的“事件摘要”图表。例如，如果配置了 12 个卷，并且每个卷具有两个严重性为“严重”的事件，那么“事件摘要”图表描述在环境中可用的所有卷上发生的事件的总计数。在此情况下，图表作为 X 轴上绘制的点来显示针对卷具有 24 个严重事件的条形。
- 已更新“节点详细信息”页面，用于显示网络端口详细信息。
- 已更新“卷详细信息”页面，用于显示每个选定卷的相关快照镜像和 LUN 计数的详细信息。
- 已更新“vServer 详细信息”页面，用于显示有关网络逻辑接口的信息。

Monitoring Agent for Tomcat
支持 Tomcat V9.0.5 服务器。

Monitoring Agent for WebSphere MQ

支持远程监视。添加了两个配置参数以使该代理程序能够收集远程队列管理器的监视数据。但是，这些配置参数对本地队列管理器没有任何影响。如果要配置该代理程序以监视本地队列管理器，您可以按 Enter 键来跳过指定这些参数。

有关代理程序配置的更多信息，请参阅第 795 页的『配置 WebSphere MQ 代理程序』。

EIF 槽定制增强功能

您可以将多个属性值和字面值添加到 EIF 槽。例如，针对用于测定低可用磁盘空间的阈值，您没有收到磁盘可用百分比为 **Disk_Free_Percent** 消息，而是会收到磁盘可用百分比为 **Disk_Free_Percent**，索引节点可用百分比为 **Inodes_Free_Percent**。转发的消息可能如下所示：磁盘可用百分比为 13，索引可用百分比为 9。有关更多信息，请参阅第 841 页的『定制事件以转发到 EIF 接收器』。

2018 年 4 月

新代理程序可用

Monitoring Agent for AWS 弹性负载均衡器

Amazon ELB 代理程序为您提供集中监视点，用于监视 AWS 弹性负载均衡器的运行状况、可用性和性能。代理程序显示每个负载均衡器类型（应用程序、网络和传统）的一组综合度量值，用于帮助您做出有关 AWS 弹性负载均衡器环境的明智决定。

代理程序增强功能

Response Time Monitoring Agent
IBM HTTP Server 响应时间模块 现已在 Windows 上支持 IBM HTTP Server V7、8 和 9。

Monitoring Agent for Node.js
缺省情况下，Node.js 数据收集器不再收集用户敏感信息（例如，cookie、HTTP 请求上下文和数据请求上下文）。您可以指定新环境变量 *SECURITY_OFF* 来更改此缺省行为。

Monitoring Agent for Amazon EC2
组件名称反映代理程序名称。

添加了扩展数据保留时间支持。

Monitoring Agent for WebLogic
在 AIX 上已启用事务跟踪和深入诊断。原先，仅在 Linux 和 Windows 上启用这些功能。

增强了使用事务跟踪和深入诊断的注释执行的 servlet 请求摘要向下钻取。

Monitoring Agent for Skype for Business Server
针对 Windows Server 2016 的支持。

Monitoring Agent for Sterling File Gateway
作为缺省行为，代理程序对失败文件传输的事件进行访存。您可以通过为新环境变量 **KFG_ALL_FGEVENTS** 指定合适的值来更改此缺省行为。

Monitoring Agent for Sterling Connect Direct
代理程序日志记录功能已改进。有关更多信息，请参阅故障诊断部分。

数据收集器增强功能

Node.js 数据收集器
缺省情况下，Node.js 数据收集器不再收集用户敏感信息（例如，cookie、HTTP 请求上下文和数据请求上下文）。您可以指定新环境变量 *SECURITY_OFF* 来更改此缺省行为。

切记: 要获取这项增强功能，必须从 IBM Fix Central 下载并应用 IBM Cloud Application Performance Management Node.js 数据收集器临时修订 1。有关更多信息，请参阅临时修订 1 自述文件。

J2SE 数据收集器
添加了对自动发现入口点类（主类）和 J2SE 应用程序别名的支持。

通过使用配置脚本可以在本地启用和禁用事务跟踪和深入诊断。

文档增强功能

在 Application Performance Management Developer Center 中创建一个 Web 页面，来帮助您找到每个刷新或发行版中的代理程序级别。有关更多信息，请参阅 Cloud APM 发行版中的代理程序版本。

简化每个产品表中的代理程序和数据收集器功能以提高可读性。有关更多信息，请参阅第 46 页的『功能』。

2018 年二月

新代理程序可用

Monitoring Agent for Azure Compute

Azure Compute 代理程序为您提供集中监视点，用于监视 Azure Compute 实例的运行状况、可用性和性能。此代理程序会显示一组综合度量值，帮助您做出有关 Azure Compute 环境的明智决定。这些度量包括 CPU 使用率、网络使用率和磁盘性能。

Monitoring Agent for Sterling Connect Direct

您可以使用 Sterling Connect Direct 代理程序来监视 Sterling Connect Direct 服务器的运行状况和性能。它会监视 Sterling Connect Direct 服务器的功能，例如文件传输活动、已调度的进程、保持队列进程和等待队列进程。此代理程序支持远程监视，而且是多实例代理程序。

Monitoring Agent for Sterling File Gateway

Sterling File Gateway 代理程序监视 Sterling File Gateway 应用程序，后者使用各种协议、各种文件命名约定和各种文件格式，在内外部合作伙伴之间传输文件。另外，还支持远程监视功能。

代理程序增强功能

Monitoring Agent for DataPower
支持 WebSphere MQ 代理程序与 DataPower 代理程序之间的事务跟踪。

Monitoring Agent for Db2
已添加对远程监视的支持。

Monitoring Agent for Hadoop
已添加监视 Hadoop 服务状态的支持，例如 HBase、MapReduce2、Tez 和 Ranger。

已添加监视 Hadoop 产品的支持：Cloudera CDH 5.12。

Monitoring Agent for InfoSphere DataStage
已添加将 MS SQL 用作元数据存储库的支持。

已添加对 Windows 操作系统的支持。

Monitoring Agent for Tomcat
通过使用 8.1.4.0-IBM-APM-SERVER-IF0001 补丁升级代理程序框架，支持在 PLinux 上进行事务跟踪和深入诊断。

Monitoring Agent for SAP Applications
CCMS 功能增强：`idx` 文件删除自动化。此自动化仅在 SAP 系统重新启动时工作。

Monitoring Agent for Microsoft .NET
已通过“IIS 响应时间”模块添加对最终用户事务的支持。

Monitoring Agent for Skype for Business Server
代理程序名称已由 Monitoring Agent for Microsoft Lync Server 更改为 Monitoring Agent for Skype for Business Server。

Monitoring Agent for Linux KVM
已添加对 RHEV-M 4.x 的支持。

Monitoring Agent for Linux OS
内存上载时间间隔已更改为 1 分钟。

与网络接口相关联的 IP 地址会显示在“Linux 操作系统”仪表板和“系统信息”窗口小部件上。

Monitoring Agent for UNIX OS
内存上载时间间隔已更改为 1 分钟。

数据收集器增强功能

J2SE 数据收集器
已添加对弹性引导应用程序的支持。

Availability Monitoring 增强功能
利用 Availability Monitoring 附加组件，您现在可以创建白名单和黑名单，以指定您的测试可以以及无法访问的 URL。白名单和黑名单控制哪些依赖项和资源会影响所测试 Web 应用程序的响应时间，例如第三方度量。使用通配符，按模式、域或文件类型过滤 URL。

2017 年 12 月

新代理程序可用

Monitoring Agent for InfoSphere DataStage

您可使用 DataStage 代理程序来监视 DataStage 服务器资源的运行状况和性能，例如，引擎服务、引擎系统、作业活动、作业运行状态和作业运行详细信息。此代理程序支持远程监视。

代理程序增强功能

Monitoring Agent for Hadoop

已添加监视 Hadoop 集群的支持，该集群使用基于 SPNEGO 的 Kerberos 认证进行保护，且使用 Active Directory 密钥分发中心 (KDC)。

已添加测试 Hadoop 集群主机连接的支持，该集群使用基于 SPNEGO 的 Kerberos 认证进行保护，且使用 MIT 或 Active Directory 作为密钥分发中心 (KDC)。

已添加监视下列 Hadoop 产品的支持：Cloudera CDH 5.10 和 CDH 5.11。

已添加监视 Hadoop 服务状态的支持，这些服务包括 Flume、Kafka、Titan、Spark、Knox、Pig、Slider 和 Solr。

Monitoring Agent for HTTP Server
已添加对 Windows 32 位 IBM HTTP Server 和 Apache HTTP Server 的支持。

已添加对 Linux for System z® 的支持（不支持事务跟踪）。

已添加对 Linux for System x 上的 Oracle HTTP 服务器的支持。

Monitoring Agent for IBM Integration Bus
已添加对 Linux for Power Systems (Little Endian) 的支持。

Monitoring Agent for Microsoft .NET
已添加对 ODP.NET 的监视支持。

已添加 HttpWebRequest.GetResponse() 方法的方法跟踪详细信息。

Monitoring Agent for Microsoft SQL Server

已添加对 SQL Server 2017 的容错支持。

已添加对 SQL Server 开发者版的“始终开启”功能的支持。

Monitoring Agent for MySQL
已添加对正在迁移至性能模式的信息模式表的容错支持。

已添加通过性能模式对不推荐使用的信息模式表的支持。

Monitoring Agent for Microsoft Internet Information Services
已添加对 FTP Web 站点监视的支持。

Monitoring Agent for MongoDB
已添加对远程监视的支持。

已添加监视内存中存储引擎的支持。

Monitoring Agent for OpenStack
已添加对 OpenStack 认证 V3 API 的支持。

Monitoring Agent for Oracle Database
代理程序版本更改为 8.0。

已添加配置参数 **Oracle JDBC Jar 文件**，并且已移除配置参数 **Oracle 主目录**和 **Oracle 即时客户机安装目录**。

Monitoring Agent for PostgreSQL
已添加对远程监视的支持。

Monitoring Agent for SAP Applications
已添加对 SNC 通信的支持。

添加了新的 SAP 系统宕机阈值。

Monitoring Agent for SAP NetWeaver Java Stack
已添加复原 SAP NetWeaver Application Server 实例的功能。

Monitoring Agent for Tomcat
已添加对 Linux for Power Systems (Little Endian) 的支持（仅限资源监视）。

Monitoring Agent for VMware VI
已在 ESX 服务器概述页面上添加“网络摘要”和“磁盘计数”。

已在集群摘要页面上添加“事件”组窗口小部件。

Monitoring Agent for WebSphere Applications
已添加对 Linux for Power Systems (Little Endian) 和 Linux for System z 的事务跟踪支持。

切记: 要在 Linux for Power Systems (Little Endian) 和 Linux for System z 上获取事务跟踪支持，请完成下列步骤:

1. 下载代理程序安装映像。
2. 安装 WebSphere Applications 代理程序。
3. 从 Fix Central 下载 WebSphere Applications 代理程序 临时修订 2。
4. 按该临时修订的自述文件进行操作，以应用该修订。

Monitoring Agent for WebSphere MQ
已添加“MQ 服务状态”组窗口小部件，以提供 MQ 服务详细信息。

已添加对 Linux for Power Systems (Little Endian) 的支持。

数据收集器增强功能

Liberty 数据收集器
Liberty 数据收集器所注册的受管系统名称 (MSN) 更改为反映主机名和 Liberty 服务器名称。此数据收集器的新 MSN 是 `BI:servername_hostname_md5:BLP`，其中 *md5* 是基于 MD5 的本地应用程序 GUID。*servername_hostname_md5* 的长度是 25 个字符。

切记: 要获取这项增强功能，必须从 Fix Central 下载并应用 IBM Cloud Application Performance Management Liberty 数据收集器临时修订 1。

J2SE 数据收集器
已添加对 J2SE 应用程序的事务跟踪支持。

文档增强功能

为了方便您准备环境，提供了有关代理程序和数据收集器所用缺省端口的信息。请参阅第 69 页的『代理程序和数据收集器使用的缺省端口。』。

提供了有关 Cloud APM 代理程序的受管系统名称 (MSN) 的信息。另外，还提供了有关如何更改 MSN 中的主机名字符串的指示信息。请参阅第 142 页的『受管系统名称』。

在下列代理程序的配置主题中，提供了有关以非管理员用户身份运行代理程序或者以非管理员用户身份运行代理程序所需许可权的信息：

- Microsoft .NET 代理程序
- Microsoft Active Directory 代理程序
- Microsoft Exchange Server 代理程序
- Skype for Business Server 代理程序
- Microsoft SharePoint Server 代理程序
- Microsoft SQL Server 代理程序
- Tomcat 代理程序

2017 年 8 月

IBM Cloud Application Performance Management, Availability Monitoring

Availability Monitoring 附加组件从全球多个存在点提供了对 Web 应用程序的增强型合成监视。每隔一定时间创建模拟用户行为的合成测试。从公用存在点运行您的测试，或者下载您自己的定制存在点并在本地或专用服务器上对其进行部署。通过图形、明细表和映射视图，使用 Availability Monitoring 仪表板来监视应用程序可用性、性能和警报。使用瀑布分析来确定性能和可用性问题发生的时间，并查找这些问题的原因。

IBM API Connect® 监视

Cloud APM 代理程序和数据收集器现在支持 IBM API Connect 环境的监视。您可以部署相应的代理程序和数据收集器，以了解环境中组件的运行状况和性能。除资源监视和深入诊断数据外，还提供了事务跟踪数据，它允许您查看有关 IBM API Connect 环境的拓扑信息。有关更多信息，请参阅第 73 页的『场景：监视 IBM API Connect』。

操作系统支持

Linux for System z

已添加对下列监视代理程序的 Linux for System z 支持：Linux OS、WebSphere Application、Db2、WebSphere MQ、IBM Integration Bus、Tomcat 和 Response Time Monitoring。

Linux for Power Systems (Little Endian)

已添加对下列监视代理程序的 Linux for Power Systems (Little Endian) 支持：Linux OS、WebSphere Application 和 Db2。

Linux for System x

已添加 Linux on System x 用于支持 Liberty 数据收集器。

IBM i 操作系统代理程序支持

现在可以在 Cloud APM 控制台中显示 IBM i 操作系统代理程序的数据。此代理程序是 IBM Tivoli Monitoring V6 代理程序，并且保持为 V8.1.4 发行版的 V6 代理程序。您可以使用 Hybrid Gateway 检索代理程序数据并将其发送到 Cloud APM 服务器。因此，您可以在 Cloud APM 控制台中查看此代理程序的监视数据和事件。有关 IBM i 操作系统代理程序的更多信息，请参阅 Hybrid Gateway supported agents (APM Developer Center)。

新代理程序可用

Monitoring Agent for OpenStack

您可使用 OpenStack 代理程序来监视 OpenStack 应用程序的运行状况和性能，查看 API 端点、SSH 服务器连接、进程和系统管理程序等信息。

提供了新的增强型数据收集器

您可使用数据收集器来监视 IBM Cloud 或本地（或两者）上的下列应用程序的运行状况和性能：

J2SE 数据收集器

您可使用 J2SE 数据收集器来监视 Java 应用程序的运行状况和性能，查看响应时间、吞吐量、请求上下文和请求方法跟踪等诊断数据。

Liberty 数据收集器

Liberty 数据收集器监视 IBM Cloud 环境中的 Liberty 概要文件和 Linux for System x 上的本地 Liberty 概要文件。

Node.js 数据收集器

Node.js 数据收集器监视 IBM Cloud 和本地部署应用程序。您可查看资源和诊断监视数据，例如资源利用率、吞吐量以及请求和方法的相关详细信息。

Python 数据收集器

Python 数据收集器监视 IBM Cloud 应用程序。您可查看资源和诊断监视数据，例如资源利用率、吞吐量以及请求和方法的相关详细信息。

从 Cloud APM V8.1.4 中的代理程序安装包中移除 Python 代理程序。您只能使用 Python 数据收集器来监视 Python 应用程序。

Ruby 数据收集器

Ruby 数据收集器仅监视 IBM Cloud 应用程序。您可查看资源和诊断监视数据，例如资源利用率、吞吐量以及请求和方法的相关详细信息。

代理程序增强功能

Monitoring Agent for Amazon EC2

- 代理程序可以正确处理已调度事件的空的结束日期。
- 已添加对 Amazon EC2 代理程序和 Amazon Web Service 之间转发代理的支持。

Monitoring Agent for Citrix Virtual Desktop Infrastructure

- 已添加 Windows 事件和 PowerShell 度量的监视，即使代理程序安装在 Linux 系统上。
- 已添加“**VDA 会话**”页面，可通过“**VDA 机器详细信息**”页面访问。
- 已将“机器度量”窗口小部件添加到“**VDA 机器详细信息**”页面。
- 已增强桌面交付控制器 (DDC) 配置，以启用代理程序来处理分发环境中的 DDC 故障转移。

Monitoring Agent for Db2

- 已添加对 Linux for System z 的支持。

Monitoring Agent for Hadoop

- 已添加监视下列 Hadoop 产品的支持：Hortonworks HDP 2.6、Cloudera CDH 5.9、5.10 和 5.11
- 已添加监视 Hadoop 服务的状态的支持，例如，ZooKeeper、Sqoop、Hive、HDFS、YARN、Ambari 度量和 Oozie。
- 已添加监视 Hadoop 集群的支持，该集群使用基于 SPNEGO 的 Kerberos 认证进行保护，且仅使用 MIT Kerberos V5 密钥分发中心 (KDC)。

Monitoring Agent for IBM Integration Bus

已添加对 Linux for System z 的支持（不支持事务跟踪）。

Monitoring Agent for JBoss

- 高级代理程序产品中的 JBoss 代理程序简化了事务跟踪和深入诊断配置过程。
- 已将两个仪表板窗口小部件添加到“**垃圾回收详细信息**”页面。一个窗口小部件显示自上次垃圾回收以来释放的堆内存量，另一个窗口小部件显示历史 Eden/幸存者/永久性（旧代）堆内存池大小。

Monitoring Agent for Linux OS

已添加对 Linux for Power Systems (Little Endian) 的支持。

Monitoring Agent for Skype for Business Server

- 已将“Lync 使用情况摘要”组窗口小部件添加到“Lync Server 概述”仪表板，用于查看前端注册状态和质量较差的调用。
- 已添加一个仪表板，用于显示 Microsoft Lync Server 使用情况的详细信息。

Monitoring Agent for SAP NetWeaver Java Stack

在 SAP NetWeaver Java Stack 代理程序的仪表板中，已添加下列增强功能：

- 已添加数据集、组窗口小部件和页面，用于收集和查看事务跟踪和诊断数据。
- 已添加在 Windows 2016 系统上安装及配置代理程序的支持。
- 已将“按响应时间排名前 5 位的最慢请求”组窗口小部件添加到“SAP NW Java 实例”仪表板，用于提供有关用户对具有较高响应时间的应用程序发出的排名前 5 位请求的信息。
- 关于在“按响应时间排名前 5 位的最慢请求”组窗口小部件中显示的请求的诊断信息，可通过单击该请求在“请求实例”页面中查看。在“按响应时间排名前 5 位的最慢请求”组窗口小部件中，已添加显示有关请求的诊断信息的支持。
- 移除“按响应时间排名前 5 位的最慢请求”组窗口小部件。

Monitoring Agent for MongoDB

主节点失败时添加了监视 MongoDB 集群或复制设置的支持。

Monitoring Agent for MySQL

已添加数据集和配置参数，用于以远程方式监视 MySQL 资源。

Monitoring Agent for NetApp Storage

- 已更新“组件”页面，用于显示集群和 Vserver 的摘要信息。
- 已更新“NetApp 存储实例”页面，用于显示有关集群的信息。

Monitoring Agent for Node.js

在 Node.js 代理程序中，已添加下列增强功能，来利用节点应用程序度量 (Appmetrics)：

- 新的仪表板和组窗口小部件，用于查看垃圾回收详细信息
- 新的仪表板和组窗口小部件，用于查看事件循环详细信息

Monitoring Agent for PostgreSQL

- 支持在 Windows 系统上安装及配置代理程序。
- 支持监视 PostgreSQL V9.6。
- 已更新“**状态概述**”页面，使得缓冲区命中率为零时状态并非严重。

Monitoring Agent for SAP HANA Database

已添加“许可证到期天数”属性、HANA_License_Expiry_Crit_SYS 和 HANA_License_Expiry_Warn_SYS 阈值，用于监视许可证到期之前剩余的天数。

Monitoring Agent for Tomcat

- 已添加对 Linux for System z 的支持
- 已添加数据集、仪表板以及事务跟踪和深入诊断的组窗口小部件。

Monitoring Agent for VMware VI

已更新“服务器摘要”仪表板中的“ESX 服务器”组窗口小部件，用于显示 SSH 状态。

Monitoring Agent for WebLogic

已将事务跟踪和深入诊断添加到高级代理程序产品中的代理程序。

Monitoring Agent for WebSphere Applications

已添加对 Linux for System z 的支持（不支持事务跟踪）。

Monitoring Agent for WebSphere MQ

已添加对 Linux for System z 的支持（不支持事务跟踪）。

支持通道和队列长期历史数据。在队列管理器配置为收集通道或队列统计数据之后，可以配置代理程序以启用通道或队列长期历史数据收集。虽然没有预定义的仪表板或窗口小部件来显示收集的长期历史数据，但是可以使用**属性详细信息**选项卡来查询定制表中收集的数据。

Response Time Monitoring Agent

- 已添加对 Windows 32 位 IBM HTTP Server 和 Apache HTTP Server 的支持。
- 已添加在“**代理程序配置**”页面上为应用程序配置用户跟踪的支持。
- 已添加在“**代理程序配置**”页面上为应用程序配置会话跟踪的支持。

增强的可视化

定制视图

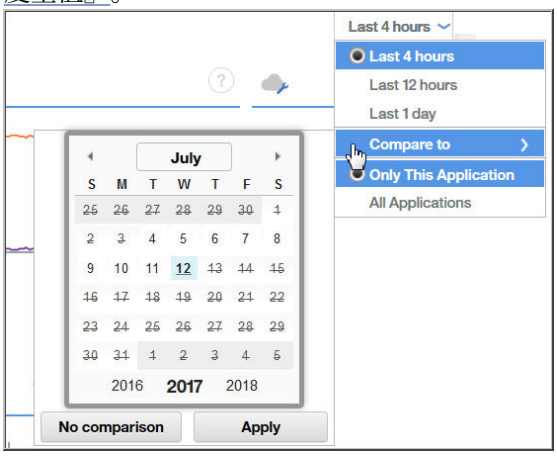
您可以使用 IBM Cloud Application Business Insights Universal View 为正在监视的应用程序创建定制的页面。在“定制视图”选项卡中，可以使用现有模板或针对页面创建定制的模板。您可以从不同的图表和度量选项中进行选择，以创建窗口小部件来根据需求监视数据。

通过使用 Universal View，可以创建仪表板来监视各种代理程序中的数据。您可以将定制页数据导出到原始数据文件。

有关更多信息，请参阅第 942 页的『定制视图』。

用于比较前一天数据的日历

当查看显示历史数据的折线图时，在选择时间选择器选项以比较前一天的时间范围之后，会打开一个日历。移除不可用于比较的天数。有关更多信息，请参阅第 924 页的『随着时间变化调整和比较度量值』。



Agent Builder 增强功能

当创建代理程序以监视 Java 数据库连接 (JDBC) 数据库中的数据时，可以修改为错误、缺失数据和 No 值设置的枚举值，以避免与数据库中的合法值重叠。

您可以为 Agent Builder **首选项**窗口中的 JAR 文件设置添加时间戳记权限。如果缺省添加时间戳记权限签名证书到期，通过设置新的权限，您可以继续验证 JAR 文件。

增强的集成

可定制的事件的 EIF 槽

配置事件转发后，可以定制基本 EIF 槽信息并为发送到接收方（例如 Netcool/OMNIbus）的事件创建定制 EIF 槽。阈值编辑器具有新的**转发 EIF 事件？**字段和 **EIF 槽定制**按钮，用于定制事件映射到转发事件的方式。有关更多信息，请参阅第 841 页的『定制事件以转发到 EIF 接收器』。

多个 Hybrid Gateway

在先前发行版中，可以将 Hybrid Gateway 安装到单个 IBM Tivoli Monitoring 域，该域有一个中心 Tivoli Enterprise Monitoring Server。现在，您可以将 Hybrid Gateway 安装到多个 Tivoli Monitoring 域。Cloud APM 控制台“**高级配置**”页面中的 Hybrid Gateway 类别已移动到其自己的“**Hybrid Gateway 管理器**”页面中。您可在此处创建和编辑用于监视来自多个 Tivoli Monitoring 域的受管系

统的 Hybrid Gateway 概要文件，每个域对应一个概要文件。有关更多信息，请参阅第 811 页的『Hybrid Gateway』。

增强型可扩展性

可以通过 Cloud APM 监视的最大受管系统数量从 4,000 个受管系统增加到 10,000 个受管系统。

先前发行版

有关先前发行版中新功能部件、功能和覆盖范围的信息，请参阅以下“新增功能”主题：

- 第 24 页的『新增功能：2017 年 4 月』
- 第 30 页的『新增功能：2016 年 9 月』
- 新增功能：2016 年 4 月

新增功能：2017 年 4 月

2017 年 4 月 Cloud APM 的发行版中提供了新增功能部件、功能和覆盖范围。

新的 Application Performance Management Developer Center

APM Developer Center 是一个中央位置，您可从中访问 APM 产品的资源：博客、视频、文档、支持、事件、IBM Marketplace 和其他资源。Cloud APM 控制台 **帮助**菜单包含指向 Application Performance Management Developer Center 的便利链接。

产品品牌再造及简化

IBM Performance Management on Cloud 已品牌再造为 IBM Cloud Application Performance Management。 组件名称也已更改。例如，Cloud APM 控制台和 Cloud APM 服务器在旧版本中分别称为 Performance Management 控制台和 Performance Management 服务器。

IBM Performance Management on Cloud 预订产品已合并，而且已重命名：

2016 年十月发行版及先前版本的产品名称	2017 年三月发行版及更新版本的产品名称
Monitoring on Cloud	Cloud APM, Base
Application Performance Management Advanced on Cloud	Cloud APM, Advanced

部分产品扩展已合并，而且已重命名：

2016 年十月发行版及先前版本的扩展名称	2017 年三月发行版及更新版本的扩展名称
Base Extension Pack (Hadoop 代理程序)	Base Extension Pack （添加新的 Cassandra 代理程序和 Microsoft Office 365 代理程序）
Advanced Extension Pack （SAP HANA Database 代理程序和 SAP NetWeaver Java Stack 代理程序）	Advanced Extension Pack （添加新的 RabbitMQ 代理程序）

操作系统支持

Windows 2016 操作系统

已添加对 Windows 2016 操作系统的支持。有关更多信息，请参阅所有代理程序的软件产品兼容性报告 (SPCR)：http://ibm.biz/agents-pm-systemreqs
请在此报告的 Windows 部分中查找您的操作系统，并单击组件图标以获取受支持代理程序的列表。

新扩展包可用

IBM Cloud Application Performance Management z Systems® Extension Pack

z Systems Extension Pack 在 Cloud APM 产品中启用对 IBM OMEGAMON® 代理程序的支持。OMEGAMON 代理程序数据由 Hybrid Gateway 发送到 Cloud APM 服务器。Hybrid Gateway 从

OMEGAMON 代理程序所连接的 IBM Tivoli Monitoring 基础结构检索 OMEGAMON 代理程序数据和事件。因此，您可在 Cloud APM 控制台中查看 OMEGAMON 代理程序的监视数据和事件。

如果您有任何 Cloud APM 产品，那么 Cloud APM z Systems Extension Pack 可用。

要将此扩展包与 Cloud APM 集成，请完成第 820 页的『与 OMEGAMON 集成』中的步骤。

提供了新的代理程序和数据收集器

Monitoring Agent for Cassandra

您可使用 Cassandra 代理程序来监视 Cassandra 集群资源（例如，节点、键空间和列系列）的运行状况和性能。

Monitoring Agent for Microsoft Office 365

您可使用 Microsoft Office 365 代理程序来监视 Office 365 资源（例如，Office 365 预订服务、Office 365 门户网站、邮箱用户、SharePoint 站点和 OneDrive 存储器）的运行状况和性能。

Monitoring Agent for NetApp Storage

您可使用 NetApp Storage 代理程序，利用 NetApp OnCommand Unified Manager (OCUM) 来监视 NetApp 存储系统的运行状况、可用性和性能。监视代理程序将执行下列任务：

- 识别性能低下的存储系统对象
- 使用焦点处的 OCUM 服务器执行发现和监视

Monitoring Agent for RabbitMQ

您可使用 RabbitMQ 代理程序来监视 RabbitMQ 集群资源（例如，集群的节点、队列和通道）的运行状况和性能。

用于 Bluemix® 应用程序的数据收集器

您可使用用于 Bluemix 应用程序的数据收集器来监视 Bluemix 上的下列各类应用程序的运行状况和性能：

- Liberty 应用程序
- Node.js 应用程序
- Python 应用程序
- Ruby 应用程序

您可查看资源和诊断监视数据，例如资源利用率、吞吐量以及请求和方法的相关详细信息。

Monitoring Agent for Siebel

您可使用 Siebel 代理程序来监视 Siebel 资源的运行状况和性能，其中包括 Siebel 统计信息、用户会话、组件、任务、应用程序服务器、Siebel Gateway Name Server、进程 CPU 和内存使用情况以及日志事件监视。

代理程序增强功能

Monitoring Agent for Amazon EC2

在 Amazon EC2 代理程序中，已添加下列增强功能：

- 当标记名称可用时，将实例标识替换为标记名称
- 允许根据标记名称对数据进行过滤和分组

Monitoring Agent for Db2

在 Hadoop 代理程序中，已添加下列增强功能：

- 支持 Linux on Power Little Endian (pLinux LE)
- 已添加一个脚本文件，用于授权 Db2 用户查看 Db2 代理程序中适用于受监视实例的所有属性的数据

Monitoring Agent for Hadoop

在 Hadoop 代理程序中，已添加下列增强功能：

- 已添加在 Windows 2016 和 AIX 7.2 系统上安装及配置代理程序的支持
- 已添加监视下列 Hadoop 产品的支持：Hortonworks HDP 2.5、Cloudera CDH 5.6、5.7 和 5.8 以及 IBM BigInsights 4.2
- 已添加“测试连接”按钮，用于在配置代理程序时验证与所指定 Hadoop 守护程序的连接
- 已改进代理程序配置过程，缩短了配置时间，并降低了复杂性。配置已简化，现在不需要下列先决条件及配置任务：
  - 在 Hadoop 集群的每个节点上安装插件
  - 配置和更新 `hadoop-metrics2.properties` 文件
  - 在配置 `hadoop-metrics2.properties` 文件之后重新启动 Hadoop 守护程序
  - 配置集群中的所有 DataNode 和 NodeManager
  - 在集群中添加其他节点时重新启动代理程序

Monitoring Agent for JBoss

在 JBoss 代理程序中，已添加下列增强功能：

- 在“高级代理程序”产品中已添加事务跟踪和深度监视功能
- 已添加用于监视数据源度量的仪表板页面
- 已添加监视下列 JBoss 产品的支持：WildFly 8.x/9.x/10.x、JBoss EAP 7.x 和 JBoss AS 7.x
- 已添加在 Windows 操作系统上运行代理程序的支持

Monitoring Agent for Linux KVM

在 Linux KVM 代理程序的仪表板中，已添加下列增强功能：

- 已更新“主机、集群和存储器”页面中的“主机”组窗口小部件，来显示“最大调度内存量 (GB)”和“目前使用的快照”KPI
- 已添加“存储器详细信息”页面，用于显示存储池中磁盘及磁盘快照的相关详细信息
- 在“主机详细信息”页面中添加“传输/接收的网络数据量 (GB)”组窗口小部件，用于显示通过网络传输及接收的总数据量 (GB) 历史信息

Monitoring Agent for Linux OS

在 Linux OS 代理程序中，已添加下列增强功能：

- 支持 Linux on Power Little Endian (pLinux LE)

Monitoring Agent for Microsoft Exchange Server

在 Microsoft Exchange Server 代理程序的仪表板中，已添加下列增强功能：

- 在“可达性”数据集中，已添加入站时间和出站时间属性
- 已添加用于显示可达性详细信息的页面及组窗口小部件
- 已添加可达性的事件触发阈值
- 已添加在 Exchange Server 2016 和 Windows Server 2016 系统上安装和配置代理程序的支持

Monitoring Agent for Microsoft Internet Information Services

在 Microsoft IIS 代理程序中，已添加下列增强功能：

- 已添加在 Microsoft Windows Server 2016 系统上安装和配置代理程序的支持

Monitoring Agent for Microsoft Active Directory

在 Microsoft Active Directory 代理程序中，已添加下列增强功能：

- 已添加组窗口小部件和页面，用于显示组策略对象、Netlogon、本地安全机构和 LDAP 的详细信息
- 已添加下列数据集（可在**属性详细信息**选项卡中查看）：
  - 服务数据集
  - 复制
  - 文件复制服务
  - 已移动或已删除的组织单元

- LDAP 属性
- 安全帐户管理器
- DFS
- 地址簿
- 事件日志
- 密码设置对象

· 已添加用于 ADFS、ADFS 代理和异步线程队列的数据集

· 已添加组窗口小部件和页面，用于显示 ADFS 和 ADFS 代理的详细信息

· 已添加在 Windows Server 2016 系统上安装和配置代理程序的支持

Monitoring Agent for Microsoft .NET

在 Microsoft .NET 代理程序的仪表板中，已添加下列增强功能：

· 已更新“组件”页面上的“MS .NET 状态”组窗口小部件，以显示数据库调用响应时间、具有较高线程计数的 .NET 进程的状态以及即时 (JIT) 编译故障数

· 已添加数据集、页面和组窗口小部件，用于显示所选 .NET 进程的 JIT 编译详细信息、数据库调用详细信息、GC 句柄及固定对象集合，以及显示线程争用比率和线程队列长度

· 已针对 .NET 进程中的 JIT 故障、.NET 请求故障、慢速命令、垃圾回收及活动线程数，添加事件触发阈值

Monitoring Agent for Microsoft SQL Server

在 Microsoft SQL Server 代理程序的仪表板中，已添加下列增强功能：

· 在“服务器性能 - 详细信息”页面中，已添加“花费时间较长的查询”组窗口小部件，用于根据 Microsoft SQL Server 的性能统计信息，查看排名前 10 位的已快取查询计划

· 已添加监视 Microsoft SQL Server 2016 的支持

· 已添加在 Microsoft Windows Server 2016 系统上安装和配置 Microsoft SQL Server 代理程序的支持

· 已添加新的 *COLL_ERRORLOG_RECYCLE_WAIT* 环境变量，用于设置代理程序在收集“MS SQL 错误事件详细信息”属性组的数据之前，所等待的时间间隔（以秒计）

Monitoring Agent for MongoDB

在 MongoDB 代理程序的仪表板中，已添加下列增强功能：

· 已更新“组件”页面，来显示 MongoDB 实例的数目及状态

· 已添加用于显示 MMAPv1 和 WiredTiger 存储引擎详细信息的页面

· 已添加“输入输出信息”页面，用于显示所选主机的已排队操作、活动连接、数据流及数据访问的游标详细信息及历史数据

· 已添加用于显示 V2.x、V3.x 或更高版本锁定详细信息的页面

· 已添加“复制详细信息”页面，用于显示复制成员、操作日志和复制延迟历史数据的详细信息，以及操作日志所用的空间量

Monitoring Agent for Node.js

在 Node.js 代理程序中，已添加下列增强功能，来利用节点应用程序度量 (Appmetrics)：

· 已添加新的仪表板和组窗口小部件，用于查看垃圾回收详细信息

· 已添加新的仪表板和组窗口小部件，用于查看事件循环详细信息

Monitoring Agent for PostgreSQL

在 PostgreSQL 代理程序中，已添加下列增强功能：

· 已添加在 Windows 系统上安装和配置代理程序的支持

· 已添加监视 PostgreSQL V9.6 的支持

· 已更新“状态概述”页面，使得缓冲区命中率为零时状态并非严重

Monitoring Agent for SAP NetWeaver Java Stack

在 SAP NetWeaver Java Stack 代理程序中，已添加下列增强功能：

- 已添加数据集、组窗口小部件和页面，用于收集和查看事务跟踪和诊断数据
- 已添加在 Windows 2016 系统上安装和配置代理程序的支持

Monitoring Agent for Synthetic Playback

在 Synthetic Playback 代理程序中，已添加下列增强功能：

- Synthetic Playback 代理程序提供了针对合成事务的新过滤功能。在 Synthetic Script Manager 中，可以为合成事务配置黑名单和白名单，用于排除或包括针对所指定 URL 和域的请求。使用黑名单和白名单可过滤掉或包括影响应用程序响应时间的依赖项，例如第三方度量。

Monitoring Agent for Tomcat

在 Tomcat 代理程序中，已添加下列增强功能：

- 已添加在 Windows 和 SUSE Linux Enterprise 12 系统上安装和配置 Tomcat 代理程序的支持

Monitoring Agent for WebSphere Applications

在 WebSphere Applications 代理程序 中，已添加下列增强功能：

- 支持 Linux on Power Little Endian (pLinux LE)。（在 pLinux LE 系统上不支持事务跟踪。）
- 已添加对传统 IBM WebSphere Application Server V9 的支持。
- 已添加“内存分析”仪表板，用于帮助您通过检查每个堆转储的堆使用率信息，诊断可能的内存泄漏。必须启用诊断方式，这个仪表板才会包含数据。
- 已添加支持，允许使用“应用程序运行状态”数据集来创建事件阈值，以用于应用程序状态监视。缺省情况下，将禁用有关此使用情况的数据收集。您必须先修改数据收集器属性文件以将其启用，然后才能创建事件阈值。
- 已简化数据收集器的手动配置。对于 WebSphere Applications Server，您只需在 WebSphere 管理控制台上添加一些用于应用程序服务器的 JVM 自变量和变量。对于 Liberty，您只需修改服务器的三个文件。

Response Time Monitoring Agent

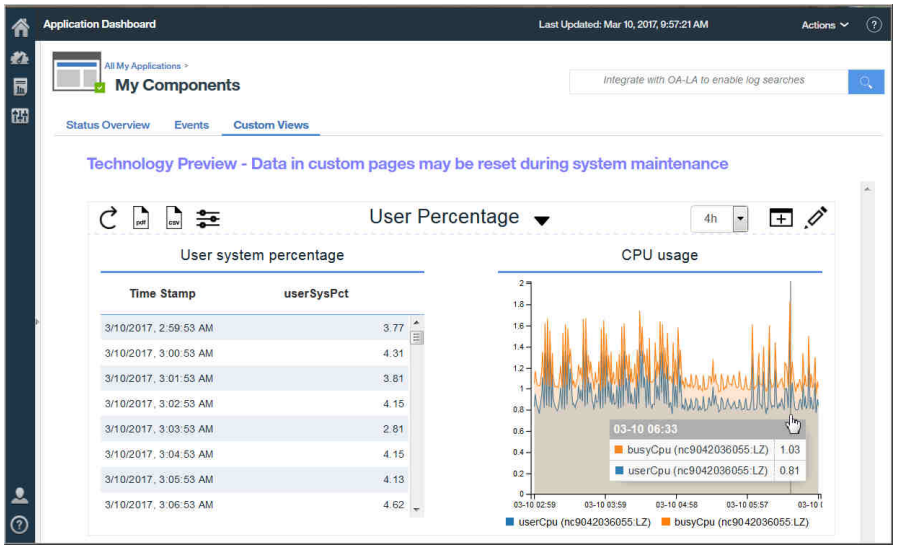
在 Response Time Monitoring 代理程序中，已添加下列增强功能：

- 已添加支持，允许使用“**代理程序配置**”页面为应用程序配置用户跟踪。
- 已添加支持，允许使用“**代理程序配置**”页面为应用程序配置会话跟踪。

Cloud APM 控制台增强功能

已对代理程序安装和配置界面进行各种改进，并已进行下列控制台增强：

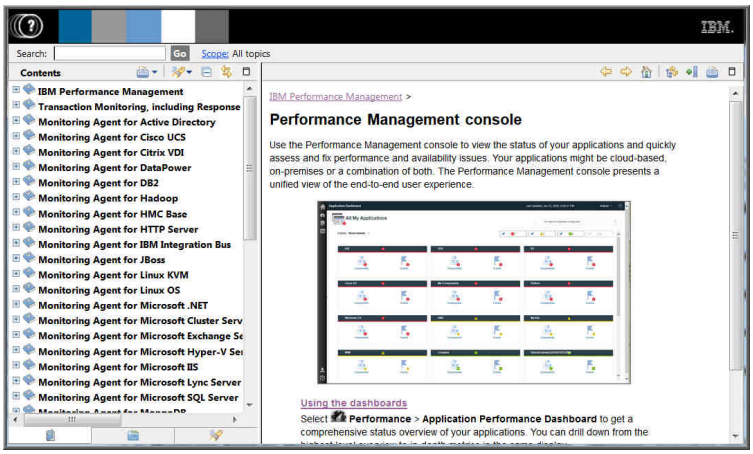
- 技术预览：在“应用程序仪表板”页面中，提供了新的“**定制视图**”选项卡。您可创建各种视图以报告受管资源的度量，以及应用平均值和计数等函数。打开已保存的页面之后，可以使用来自另一个资源的数据来刷新该页面，以及以 PDF 或 CSV 文件形式下载页面度量。有关更多信息，请参阅 第 942 页的『定制视图』.



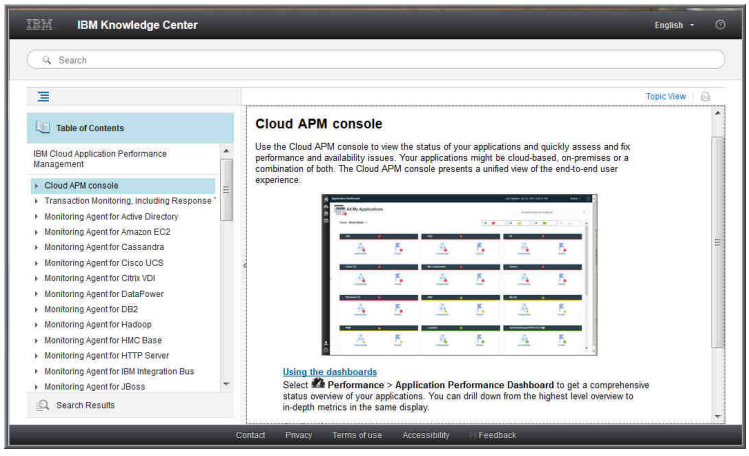
如果您在自己的 Cloud APM 预订中看不到“**定制视图**”，并且要试用这项新功能，请向 IBM 支持提交服务请求，以启用“**定制视图**”技术预览。请注意，在系统维护期间，不会保存定制仪表板页面以及其中填充的历史数据。

- 打开 Cloud APM 控制台帮助系统时，请注意，它是由 IBM Knowledge Center 托管。您可使用 **隐藏目录**工具、搜索和打印功能以及指向支持信息和反馈选项的链接。

在



之前 在



之后

Agent Builder 增强功能

已改进用于为 Agent Builder 代理程序构建 Cloud APM 摘要仪表板的支持。您必须使用单行数据集来为摘要仪表板提供数据。可以从完整的日志文件提供这样的数据集，也可以从任何可以过滤为单一行的数据集进行提供。

新增功能：2016 年 9 月

2016 年 9 月 Performance Management on Cloud 的发行版中提供了新增功能部件、功能和覆盖范围。

新代理程序可用

Monitoring Agent for Amazon EC2

您可以使用 Amazon EC2 代理程序 来监视 Amazon Elastic Compute Cloud (EC2) 实例资源的运行状况、可用性和性能。可以监视以下资源：

- CPU 利用率
- 弹性块存储器 (EBS) 利用率
- 网络利用率
- Amazon Web Service (AWS) 维护更新
- 磁盘性能

此代理程序位于 Infrastructure Extension Pack 中，并可用于下列产品：IBM Monitoring、IBM Application Performance Management 和 IBM Application Performance Management Advanced。

Monitoring Agent for SAP NetWeaver Java Stack

您可以使用 SAP NetWeaver Java Stack 代理程序 来监视 SAP NetWeaver Java Stack 集群资源和实例资源的运行状况、可用性和性能。您可以使用代理程序来监视集群资源，例如，堆转储、JVM 实例、用户会话的响应时间、事务详细信息、系统信息和许可证详细信息。您可以使用代理程序来监视实例资源，例如，CPU 利用率、磁盘利用率、内存利用率、数据库收集、垃圾回收、堆转储、失败的应用程序、Web 容器和会话信息。此代理程序位于 Advanced Extension Pack 中，如果您有下列其中一个产品，那么可以使用此代理程序：IBM Application Performance Management 和 IBM Application Performance Management Advanced。

代理程序增强功能

Monitoring Agent for Citrix Virtual Desktop Infrastructure

添加针对虚拟交付代理程序 (VDA) 和桌面交付控制器 (DDC) 机器检索 Windows 事件日志事件的能力。

Monitoring Agent for Linux KVM

仪表板可供代理程序监视基于 Linux 内核的虚拟机的部署。该仪表板提供了以下监控功能：

- 根据基于 Linux 内核的虚拟机环境或应用程序的 CPU 和内存利用率，摘要仪表板显示主机的总体状态。
- “主机详细信息”仪表板显示有关所选主机的详细信息。
- “主机”、“集群”和“存储器”仪表板显示有关受监视虚拟机的详细信息。
- “虚拟机详细信息”仪表板显示有关“主机详细信息”页面上选择的虚拟机的详细信息。

Monitoring Agent for Linux OS

支持 Docker V1.8.0 或更高版本。添加了新的属性组和窗口小部件，以使 Linux OS 代理程序 能够提供 Docker 监视功能。

Monitoring Agent for Oracle Database

“Oracle 数据库代理程序”仪表板在“实例详细信息”页面上包含以下新增功能：

- 显示所选实例上锁定争用相关信息的表。
- 显示关于 Oracle Real Application Cluster GCS 和 GES 的信息的表。
- 显示附加到所选实例的自动存储管理 (ASM) 磁盘组详细信息的表。
- 按表空间显示详细信息的视图，单击**后 5 个空闲表空间**时可见。
- 显示附加到所选实例的前台和后台进程历史详细信息的表。您可以在表中单击该实体并查看该实例的所有进程的详细表。
- 显示所选实例的前 5 个最差 SQL 查询（按运行时）的表。您可以单击该表并查看针对该实例的前 50 个最差 SQL 查询的详细表。

Monitoring Agent for Synthetic Playback
Synthetic Playback 代理程序包含新安全功能。您可以防止合成脚本中存储的密码显示在 Synthetic Script Manager 中。

Monitoring Agent for VMware VI
通过新增代理程序去耦功能，您可以在同一视图中查看和选择代理程序节点及其子节点。

在“选择组件”窗口中选择 VMware Virtual Infrastructure 组件时，组件编辑器显示包含其所有子节点的代理程序节点的树结构。

- 如果展开树并选择代理程序节点，那么自动选择所有子节点。您还可以展开树，然后单独选择要监视的子节点。
- 如果在折叠树时选择代理程序节点，那么自动排除所有子节点。

在“选择组件”窗口中选择 ESX 服务器组件，以及子节点时，独立的 ESX 服务器也显示在组件编辑器中。使用子节点，您可以选择用于监视的独立 ESX 服务器。

创建应用程序后，APM UI 仪表板将显示代理程序实例的树结构为父级，显示其子节点为子级。

Response Time Monitoring Agent

可以在“最终用户事务”仪表板中针对特定环境定制应用于特定 IP 地址或地址范围的位置。在**代理程序配置**中使用**地理位置**选项卡来定制位置值。

Cloud APM 控制台增强功能

- 对代理程序安装和配置界面进行了各种改进。
- 已将**仪表板日志**选项添加到**操作**菜单，来复审自上次重新启动服务器以来更新的代理程序仪表板的列表。有关更多信息，请参阅第 915 页的『我的所有应用程序 - 应用程序性能仪表板』。
- 所选应用程序的应用程序性能仪表板页面已简化，改善了查看效果。严重和警告严重性事件计数显示在“事件”选项卡标题中，取代了“**事件严重性摘要**”条形图。对于启用了拓扑视图的应用程序，“**汇总应用程序拓扑**”视图包含一个切换按钮，用于切换到“当前组件状态”条形图。有关更多信息，请参阅第 917 页的『状态概述』。
- 在先前发行版中，应用程序性能仪表板 **属性详细信息**选项卡仅适用于组件实例。**属性详细信息**选项卡可用于创建 Response Time Monitoring 代理程序 和 Synthetic Playback 代理程序事务实例的历史表。对于有视觉障碍的用户，创建历史表的能力可提供对于辅助技术（如屏幕朗读器软件）无法解释的折线图的替代方法。有关更多信息，请参阅第 925 页的『查看和管理定制图表和表』。

API
您可以使用 API 来创建脚本，以使 Performance Management 环境自动上线。有关更多信息，请参阅第 910 页的『浏览 API』。

新增功能：2016 年 4 月

2016 年 4 月 Performance Management on Cloud 的发行版中提供了新增功能部件、功能和覆盖范围。

IBM Marketplace
IBM Performance Management on Cloud 产品可从 IBM Marketplace 获得。注册以免费试用或预订帐户。有关更多信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

新代理程序可用

Monitoring Agent for Citrix Virtual Desktop Infrastructure
您可以使用 Citrix VDI 代理程序 来监视 Citrix XenDesktop 或 XenApp 资源的运行状况、可用性和性能，例如，站点、机器、应用程序、桌面、会话和用户。此代理程序位于 Infrastructure Extension Pack 中，并可用于下列产品：IBM Monitoring、IBM Application Performance Management 和 IBM Application Performance Management Advanced。

Monitoring Agent for Skype for Business Server
您可以使用 Skype for Business Server 代理程序 来监视 Microsoft Lync Server 资源的运行状况、可用性和性能，例如，数据库、调解服务器、合成事务、即时消息传递、CDR 服务写操作和 SIP 同级。

Monitoring Agent for WebLogic

您可以使用 WebLogic 代理程序 来监视 WebLogic Server 资源的运行状况、可用性和性能，例如 Java 虚拟机 (JVM)、Java 消息传递服务(JMS) 和 Java 数据库连接 (JDBC)。

集成增强功能

代理程序共存

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Performance Management 代理程序。但是，两种代理程序不能安装在相同目录中。请参阅第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

IBM Alert Notification

“警报通知”包含一个移动应用，后者可在 iOS 和 Android 设备上提供部分“警报通知”功能。

IBM 集成堆栈监视

您可以监视 IBM 集成堆栈，针对 IBM MQ、IBM Integration Bus 和 DataPower 设备中间件产品及其公开的服务查看事务跟踪信息，并在发生问题时进行故障诊断。请参阅第 81 页的『场景：监视 IBM 集成堆栈』。

代理程序增强功能

Monitoring Agent for Db2

添加了用于向缺省用户（针对 Windows 系统）和实例所有者用户（针对 Linux 和 AIX 系统）授予特权的命令，以查看某些 Db2 代理程序 属性的数据。

Monitoring Agent for Hadoop

Hadoop 代理程序在 Linux、Windows 和 AIX 操作系统上都受支持。

Monitoring Agent for HMC Base

为虚拟 I/O 和硬件事件提供了监视功能。

Monitoring Agent for IBM Integration Bus

在 Linux 和 AIX 系统上配置代理程序期间，可自动发现最新版本 IBM MQ (WebSphere MQ) 的库路径。

Monitoring Agent for Microsoft Cluster Server

Microsoft Cluster Server 代理程序在安装后将自动进行配置。

Monitoring Agent for Microsoft Exchange Server

在代理程序配置窗口的**交换服务**选项卡中添加了其他服务，以确定 Exchange Server 的状态。

Monitoring Agent for Microsoft Hyper-V Server

移除了代理程序配置面板。此代理程序无需进行配置。

Monitoring Agent for SAP HANA Database

在“**SAP HANA 数据库详细信息**”仪表板中添加了“高速缓存信息详细信息”组窗口小部件以提供有关已用内存百分比、可用内存百分比以及受监视数据库高速缓存命中率的信息。

Monitoring Agent for Synthetic Playback

Synthetic Playback 代理程序包含以下功能部件：

- 除公共、面向外部的应用程序外，您还可以安装和配置 Synthetic Playback 代理程序 以监视“应用程序性能仪表板”上专用、面向内部应用程序的性能和可用性。
- 使用 Synthetic Script Manager 来生成简单脚本以测试应用程序的可用性和性能。
- 在不同位置配置合成事务的同时或交错回放。
- 在 Synthetic Script Manager 中监视每月回放使用情况。
- 在 Synthetic Playback 代理程序 报告中查看 HTTP 度量和可用性比率。
- 查看两个新报告：事务的趋势和子事务的趋势。
- 将合成事务组织到资源组中，并将阈值应用于该资源组中的所有事务。
- 无需创建包含关联的合成事务的应用程序，便可在应用程序性能仪表板中的“**我的事务**”窗口中查看交易数据。
- 下载 Synthetic Script Manager 中的合成脚本。

Monitoring Agent for VMware VI

VMware VI 代理程序 仪表板增强为包含以下新功能：

- “**组件**”页面中还显示处于临界或警告状态的触发的警报数量。
- “**集群摘要**”页面上的新表提供有关前摄和故障警报的信息。您可以单击表中触发的实体并查看此触发的实体的详细信息页面。
- “**集群详细信息**”页面上的新表显示属于所选集群的 ESX 服务器的详细信息。您可以单击 ESX 服务器并查看此 ESX 服务器的详细信息页面。
- “**集群详细信息**”页面上的“数据存储”表显示数据存储的超额承担指标。
- “VM 详细信息”页面上的“虚拟机”表显示更多性能指标，例如，内存大小、NIC 和磁盘。您可以单击这些度量并查看其详细信息页面。
- 添加新窗口小部件和页面以显示所选虚拟机的内存、磁盘和网络的重要性能度量。
- “**ESX 服务器详细信息**”页面上的新表显示服务器网络的网络性能。
- “**VM 详细信息**”页面和“**ESX 服务器详细信息**”页面上的“数据存储”表显示数据存储的等待时间指标。
- “**数据存储详细信息**”页面上的新表显示有关与数据存储相关联的虚拟机的信息。您可以单击表中的虚拟机并查看“**虚拟机详细信息**”页面。
- “**VM 详细信息**”页面上的“% 内存（历史数据）”图表更改为“访客内存（历史记录）”。

Monitoring Agent for WebSphere Applications

In-flight Requests Summary 仪表板提供了识别当前缓慢或挂起的请求实例的功能。您可以在此仪表板上的“正在进行的请求”窗口小部件中选择该请求，然后单击**取消线程**，对正在进行的请求执行软取消操作。

重新定义所有预定义的事件触发阈值，以提供更好的用户体验。增强功能和更新包括触发警报的条件、采样时间间隔和阈值严重性。

残疾用户可以访问 Monitoring Agent for WebSphere Applications 的用户界面。

根据客户反馈和技术复核优化配置过程，以提供更好的用户体验。

Monitoring Agent for WebSphere MQ

对预定义事件触发阈值进行了某些更改：

- 所有预定义阈值都具有前缀 MQ_ 代替先前版本中的 MQSeries_。
- 添加了两个阈值 MQ_Channel_Initiator_Crit 和 MQ_Queue_Manager_Crit，以触发通道启动程序服务器状态和队列管理器状态的临界警报。
- MQ_Queue_Depth_High 事件的触发条件已从静态 80% 更改为队列的高深度值。

“**未读取的队列 - 排名前 5 位**”窗口小部件的名称已更改为“**未读取的使用中队列 - 排名前 5 位**”。此窗口小部件提供前 5 个队列的列表，这些队列包含消息且已连接一个或多个应用程序来将消息放入队列中，但尚未有任何应用程序进行读取。

在代理程序配置期间，可自动发现最新版本 IBM MQ (WebSphere MQ) 的库路径。您可以保持静默响应文件中的 **WMQLIBPATH** 参数为空或者在以交互方式配置代理程序时接受缺省值。

操作系统代理程序

操作系统代理程序包括监视应用程序日志文件的新功能。功能包括根据正则表达式配置日志文件监视的功能。

为确保兼容性，操作系统代理程序使用以下信息和格式：

- IBM Tivoli Monitoring 6.x Log File Agent 使用的配置信息和格式文件
- Tivoli Event Console Log File Adapter 使用的配置信息和格式字符串

这些格式字符串允许代理程序根据格式文件中的模式过滤日志数据，以及仅向事件使用者提交相关数据。OS 代理程序将数据发送到 Performance Management 服务器，或者通过事件集成工具 (EIF) 发送给任何 EIF 接收方，例如，OMNIbus EIF 探测器。

Response Time Monitoring Agent

“最终用户事务”仪表板包含用户和设备信息，这些信息以前显示在“用户”组内的“已认证的用户”和“移动设备用户”仪表板中。用户、会话和设备信息根据用户的 IP 地址按位置（国家或地区、省/直辖市/自治区和城市）进行排序。使用新的和更新的仪表板，来了解用户卷以及是否将问题隔离到特定用户组。

在“最终用户事务”仪表板中针对您的特定环境定制应用于特定 IP 地址或地址范围的位置。在**代理程序配置**中使用**地理位置**选项卡来定制位置值。

事务跟踪

“事务摘要”页面包含“服务依赖关系”拓扑，后者显示选中的资源节点，例如 IBM Integration Bus 及其依赖的服务。“事务详细信息”页面包含“事务依赖关系”拓扑，后者显示每个组件实例的事务节点，以及事务级别每个相依服务的未检测节点，例如 IBM Integration Bus 及其服务事务。“事务详细信息”页面还突出显示所选应用程序中遇到最慢响应时间的用户，以及事务量最高的主机。

一般代理程序增强功能

提供了以下一般代理程序增强功能的安装和配置：

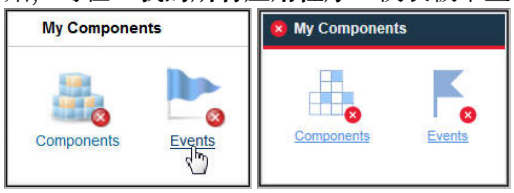
- 代理程序安装脚本会在安装启动前执行许可权检查。如果没有足够的许可权，那么将显示一条消息。
- 代理程序状态命令会检查代理程序与 Performance Management 控制台之间的状态。
- 在 Windows 系统上受支持的代理程序有一个 GUI 实用程序，可用于执行代理程序配置和检查连接状态。
- 可以使用新命令来移除代理程序实例而不必卸载代理程序。

Performance Management 服务器增强功能

Performance Management 用户认证是通过 IBMid OpenID Connect 提供程序进行管理。

Performance Management 控制台增强功能

- Performance Management 控制台外观已更新为与 IBM Bluemix 用户界面一致。例如，从 V8.1.2 开始，可在“**我的所有应用程序**”仪表板中查看摘要框之间的差别，并且现在：



- 向“高级配置”页面添加了新选项，因此高级用户可轻松地在所有系统组上启用或禁用所有预定义的阈值。请参阅第 830 页的『背景信息』。
- 向“高级配置”页面添加了新选项，以控制应用程序性能仪表板的自动刷新率。有关更多信息，请参阅第 910 页的『UI 集成』。
- 对代理程序安装和配置界面进行了各种改进。
- 对 Performance Management 控制台的可用性进行了改进。有关用户界面的辅助功能的信息，请参阅第 1275 页的『辅助功能』。

API

您可以使用 API 来创建脚本，以使 Performance Management 环境自动上线。有关更多信息，请参阅第 910 页的『浏览 API』。

Agent Builder 增强功能

Agent Builder 包含增强的数据集过滤功能。您可以使用过滤来创建数据集，根据包含可用性数据集的多行数据集返回一行。使用此功能提供摘要仪表板中的信息。

第 2 章 PDF 文档

针对该 IBM Knowledge Center 集合和代理程序参考中的主题提供了 PDF 文档。

IBM Knowledge Center（PDF 格式）

除了本用户指南，您还可以下载 IBM Agent Builder 用户指南。

代理程序参考 PDF

要下载代理程序的参考，请参阅 Application Performance Management Developer Center 上的代理程序度量/参考 PDF。参考提供有关仪表板、事件触发阈值和数据集的信息。数据集包含的特性是代理程序报告的度量值，用于构成关键性能指标 (KPI)。 您可在 PDF 文件的标题页面上找到代理程序版本。

© Copyright IBM Corp. 2014, 2019

第 3 章 产品概述

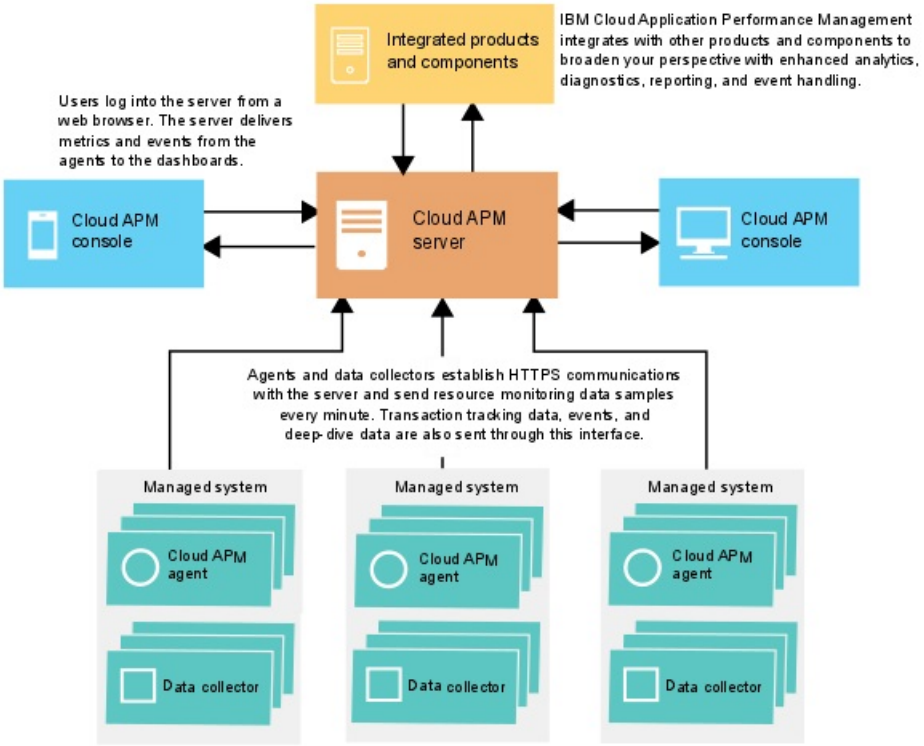
IBM Cloud Application Performance Management (Cloud APM) 是一项综合性的解决方案，可帮助管理部署在本地（专用）、公共云或在两者上都有部署的应用程序的性能和可用性。该解决方案提供应用程序的可视性、可控性和自动化，从而确保实现资源的最佳性能和高效利用。

通过使用此解决方案，借助认知智慧管理数据中心、云基础结构和工作负载。您可以整天减少并避免混合应用程序世界的中断和减速，因为 Cloud APM 帮助您从标识性能问题变为隔离发生问题的区域并在影响业务前诊断问题。

使用关键功能部件（根据产品的不同而不同），处理 Cloud APM 代理程序和数据收集器所收集的数据。通过与其他产品和组件集成可使用更多功能部件。

体系结构概述

IBM Cloud Application Performance Management 使用代理程序和数据收集器来收集受监视主机上的数据。代理程序和数据收集器将数据传递给 Cloud APM 服务器，后者将数据收集到 Cloud APM 控制台。Cloud APM 服务器托管于 IBM 云中。



数据收集

代理程序和数据收集器监视系统、子系统或应用程序并收集数据。代理程序或数据收集器与单个资源（例如系统或应用程序）进行交互，并且多数情况下是在正运行系统或应用程序的同一计算机或虚拟机上。例如，Linux OS 代理程序收集 Linux 主机上的操作系统的性能指标，WebSphere Applications 代理程序监视 WebSphere Application Server 的性能指标。另外，某些代理程序跟踪不同资源之间的事务。

您可以根据关键绩效指标设置阈值 (KPI)。如果某个指标更改为高于或低于阈值，那么代理程序或数据收集器将生成由服务器处理的警报。还可以将事件配置为转发到目标，例如 Netcool/OMNIbus Probe for Tivoli EIF 或 SMTP 服务器并使用 Alert Notification 来配置事件的电子邮件通知。

代理程序和数据收集器预先配置为与 Cloud APM 服务器通信。

© Copyright IBM Corp. 2014, 2019

服务器与代理程序或数据收集器之间的通信

每个受监视主机上的代理程序和数据收集器都建立与 Cloud APM 服务器（托管于 IBM 云中）的 HTTPS 通信。代理程序或数据收集器是连接的客户机端。

代理程序和数据收集器要求连接到因特网才能向服务器发送数据，如果无法直接通过因特网发送数据，可能需要转发代理。有关更多信息，请参阅第 135 页的『网络连接』。

服务器存储的数据

根据数据的类型，代理程序和数据收集器将数据推送到 Cloud APM 服务器，时间间隔范围为 1 分钟到 8 分钟。缺省情况下，服务器将存储所有由代理程序和数据收集器发送的值并保留 8 天。汇总的交易数据将存储更长时间。

保存的监视数据称为历史数据。服务器使用历史数据来显示表格和图形，您可以使用这些表格和图形来分析环境中的趋势。

对于某些代理程序，还提供了历史报告。有关更多信息，请参阅第 951 页的『报告』。

可伸缩性

通过 Cloud APM 最多可以监视 10,000 个受管系统。 受管系统是企业中受代理程序监视的单一操作系统、子系统或应用程序。

Cloud APM 支持每秒 150 到 400 个受监视用户事务。

集成

配置为与 Cloud APM 服务器进行通信时，IBM Cloud Application Performance Management 将与其他产品和组件集成。

可集成的产品包括 IBM Control Desk、Netcool/OMNIbus、Tivoli Monitoring、OMEGAMON、Operations Analytics - Log Analysis、Operations Analytics - Predictive Insights、IBM Alert Notification 和 IBM Cloud。

Agent Builder 是可用于创建定制代理程序的组件。

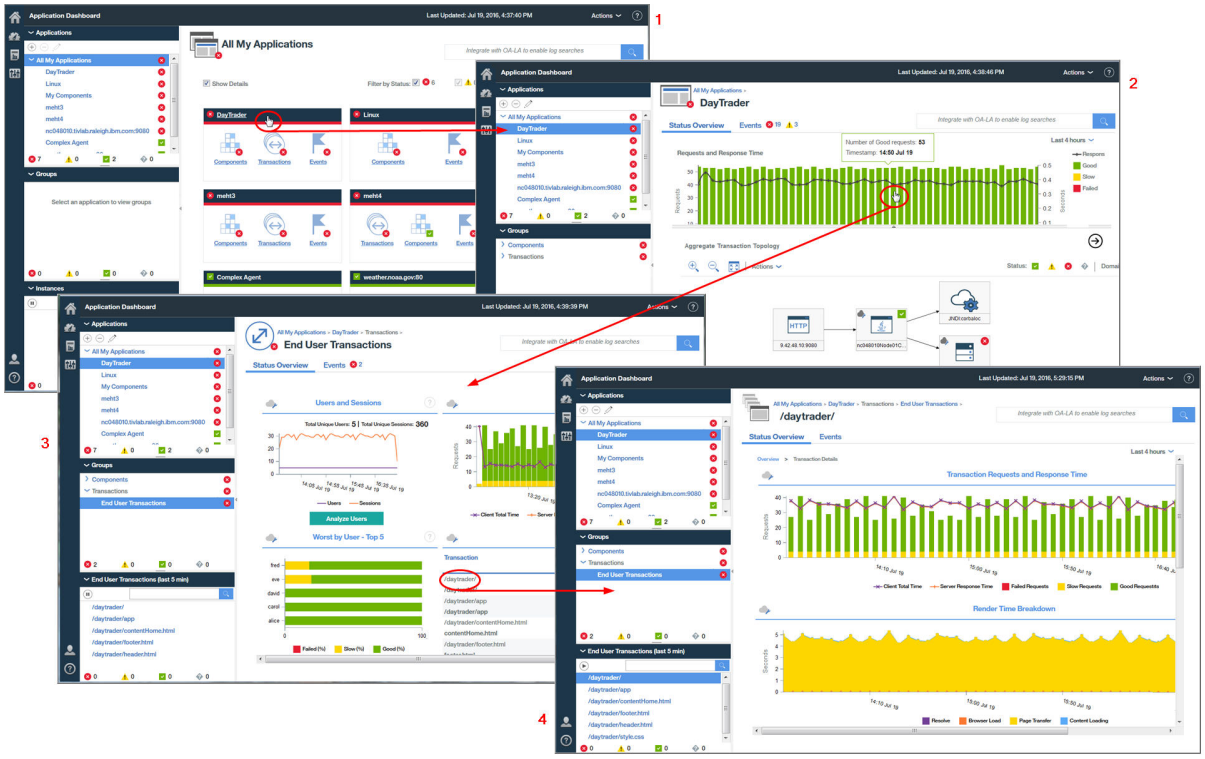
用户界面

Cloud APM 控制台是 Cloud APM 的用户界面。此统一的用户界面在混合应用程序上提供单个视图。使用控制台来查看应用程序的状态并快速评估和修订性能和可用性问题。

控制台中的仪表板简化问题确定，因此您可以在影响应用程序性能前隔离瓶颈。借助简单仪表板导航，您可以从应用程序状态视图进入代码级别详细信息。您可以在发生问题的确切时间查看源代码问题。您可以使用集成搜索分析来搜索和诊断问题。

控制台中的应用程序性能仪表板导航器为分级式结构，提供了应用程序的状态概述、其组件运行状况以及用户体验的质量。您可以单击导航器项或单击仪表板视图中的链接，以获取受监视资源的更多详细信息。例如，假设应用程序的响应时间较长。问题将显示在仪表板中。从仪表板开始，通过单击链接向下追溯到问题根源以发现原因：由于进程失控而导致系统上的 CPU 使用率过高。

有关在 Cloud APM 控制台中使用仪表板的更多信息，请参阅第 915 页的『第 10 章 使用仪表板』。



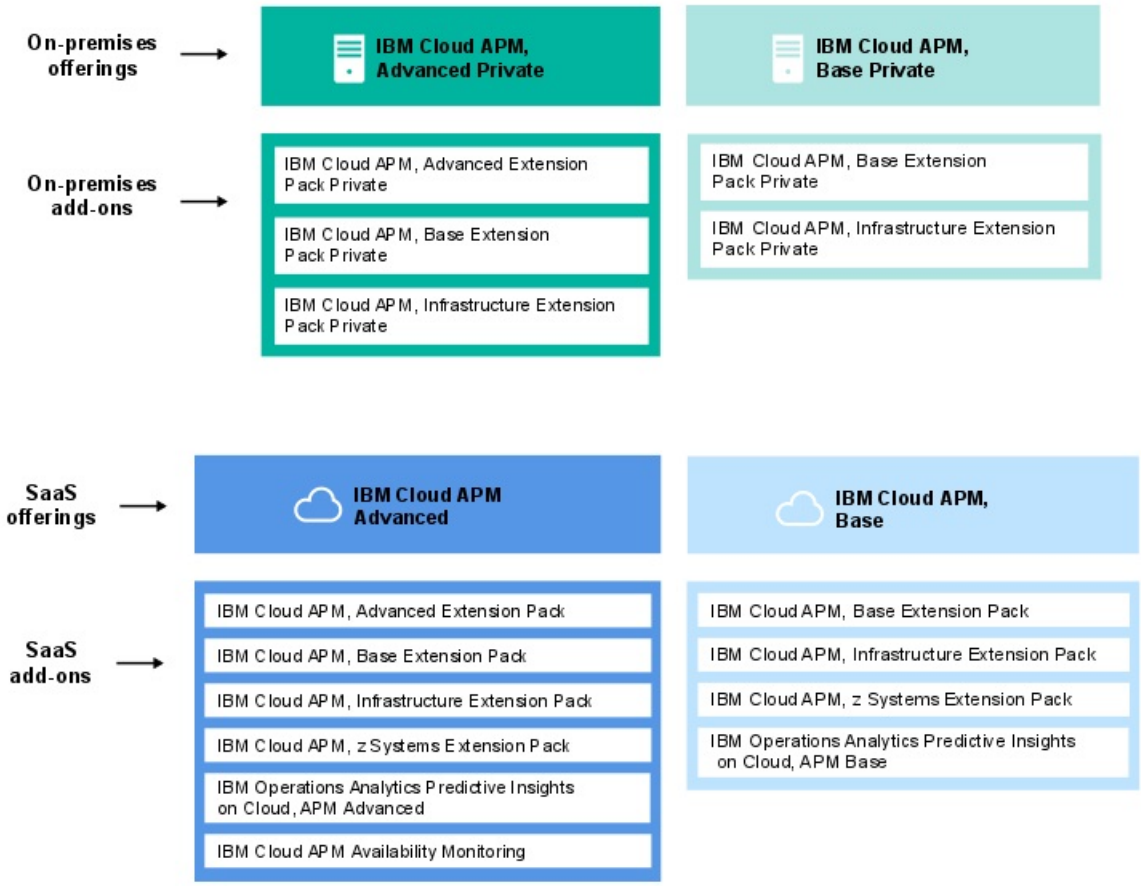
产品和附加组件

IBM Cloud Application Performance Management 包含两个产品和多个附加组件。 产品和附加组件包含代理程序和数据收集器。 特定的附加组件可以与每个产品配合使用。

要了解产品或附加组件中包含的代理程序以及代理程序和数据收集器功能，请参阅第 46 页的『功能』。

对于每个产品，附加组件在 IBM Marketplace 中提供。 IBM Cloud Application Performance Management, Advanced 是最全面的产品，其中包含所有代理程序、数据收集器和仪表板页面。IBM Cloud Application Performance Management, Base 是 Cloud APM, Advanced 的子集。您随时可以将 Cloud APM, Base 替换为 Cloud APM, Advanced。进行此替换之后，最终安装的产品是 Cloud APM, Advanced。下图显示可用于每个产品的附加组件。

对于所有产品，这些附加组件是相同的，但 Availability Monitoring 除外，这是仅适用于 Cloud APM, Advanced 产品的附加组件。



产品

IBM Cloud Application Performance Management, Advanced

此产品用于管理所有应用程序组件的最终用户体验、事务跟踪和资源监视。您可在代码级别查看应用程序，并可了解应用程序服务器的运行状况。使用诊断仪表板可以查找应用程序代码中的性能瓶颈，以及管理生产中的关键性应用程序。

此产品包含 IBM Cloud Application Performance Management, Base，并包含用来监视企业中安装的应用程序、事务及其他资源的代理程序和数据收集器。有关此产品中代理程序和数据收集器的列表，请参阅第 46 页的『功能』。

使用此产品，DevOps 具有完整的解决方案，可对应用程序和基础结构提供完整可视性和控制。业务线所有者可以管理产品中的关键应用程序和最终用户体验。应用程序开发者可以查看事务详细信息和诊断应用程序问题。

IBM Cloud Application Performance Management, Base

此产品用于对基础结构、应用程序组件和云工作负载进行资源监视。资源监视可以帮助您确定并解决缓慢事务、容量问题和停运。此产品包含用来监视安装在企业中的应用程序及其他资源的代理程序和数据收集器。有关此产品中代理程序和数据收集器的列表，请参阅第 46 页的『功能』。

使用此产品，IT 操作员可以处理缓慢事务、容量问题和停运。

附加组件

Advanced Extension Pack

此扩展包中包含 Monitoring Agent for SAP HANA Database、SAP NetWeaver Java Stack 代理程序及 Monitoring Agent for RabbitMQ。

使用 SAP HANA Database 代理程序来监视 SAP HANA 数据库。使用 SAP NetWeaver Java Stack 代理程序来监视 SAP NetWeaver Java Stack。 使用 RabbitMQ 代理程序可监视 RabbitMQ 消息传递。如果您有 IBM Cloud Application Performance Management, Advanced 产品，那么此扩展包可用。

Base Extension Pack
此扩展包中包含以下代理程序：

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

使用这些代理程序可监视 Cassandra 数据库、Hadoop 集群、DataStage 服务器资源、Microsoft Office 365 应用程序、Connect Direct 服务器和 Sterling File Gateway 应用程序。如果您有任何 Cloud APM 产品，那么此扩展包可用。

Infrastructure Extension Pack
此扩展包中包含以下代理程序：

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS 弹性负载均衡器
- Monitoring Agent for Azure Compute
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for IBM Cloud

使用 Amazon EC2 代理程序可监视 Amazon EC2 实例。使用 Amazon ELB 代理程序可监视 AWS 弹性负载均衡器。 使用 Azure Compute 代理程序可监视 Azure Compute 虚拟机。使用 Citrix VDI 代理程序可监视 Citrix Virtual Desktop Infrastructure。

如果您有任何 Cloud APM 产品，那么此扩展包可用。

z Systems Extension Pack
您可使用 z Systems Extension Pack 在 Cloud APM 控制台中查看 OMEGAMON 应用程序组件的监视数据及事件。如果您有任何 Cloud APM 产品，那么此扩展包可用。

Operations Analytics - Predictive Insights
此附加组件用于分析 Cloud APM 所收集的度量数据，并在检测到异常时生成警报。如果您有任何 Cloud APM 产品，那么此附加组件可用。

Availability Monitoring
此附加组件用于监视多个地理分布存在点中 Web 应用程序的可用性和性能。此附加组件并非作为独立产品而工作，而是在您拥有 IBM Cloud Application Performance Management, Advanced 产品时提供给您使用。

有关每个产品中的功能部件的概述，请参阅第 41 页的『产品详细信息』。

有关每个代理程序和数据收集器的描述以及各自特定的信息链接，请参阅第 49 页的『描述』。

产品详细信息

某些功能部件适用于所有产品，而其他功能部件仅适用于特定产品。

第 41 页的表 1 显示一览表中每个产品的主要功能部件。

表 *1.* 每个产品中的功能部件

功能部件	Cloud APM, Advanced（对于 DevOps、开发者及业务线）	Cloud APM, Base（对于操作）
应用程序资源监视： 语言、中间件（覆盖范围因产品而异）。	✓	✓
操作系统监视： Linux、UNIX 和 Windows 系统	✓	✓

表 1. 每个产品中的功能部件 (续)

功能部件	Cloud APM, Advanced（对于 DevOps、开发者及业务线）	Cloud APM, Base（对于操作）
日志文件监视: 使用 OS 代理程序来监视应用程序日志文件。	✓	✓
仪表板: · 在相同仪表板中查看 Tivoli Monitoring 和 Cloud APM KPI · 历史度量 · 可定制仪表板	✓	✓
API: 使用 API 管理环境。	✓	✓
基于角色的访问控制: 管理 IBM Cloud Application Performance Management 用户的访问权和特权。	✓	✓
历史报告: 针对按事务、设备、浏览器等细分的应用程序的性能和响应时间生成报告（覆盖范围根据产品而异）。	✓	✓
IBM Agent Builder: 构建定制代理程序以监视任何平台或技术。	✓	✓
数据库资源监视: （覆盖范围因产品而异）	✓	✓
基础结构资源监视: 系统管理程序、存储器和网络（覆盖范围因产品而异）。	✓	✓
商业应用程序资源监视: 业务应用程序和协作应用程序（覆盖范围因产品而异）。	✓	✓
响应时间监视: 查看应用程序性能如何影响用户。	✓	✓
与搜索分析集成: 查找快速隔离、诊断和解决问题所需的洞察力。	✓	✓
Operations Analytics - Predictive Insights（附件组件）: 在应用程序性能异常影响用户前进行确定。	✓	✓
实际最终用户体验监视: 请查看从基础结构到其设备的用户体验。	✓	–
事务跟踪: 通过应用程序环境跟踪端对端事务。 · 应用程序拓扑：查看所有组件在应用程序环境中是如何连接的。 · 事务实例拓扑：查看您的环境中针对每个事务实例所遵循的路径。	✓	–

表 1. 每个产品中的功能部件 (续)		
功能部件	**Cloud APM, Advanced（对于 DevOps、开发者及业务线）**	**Cloud APM, Base（对于操作）**
深入诊断： · 从摘要仪表板向下钻取以查看特定代理程序的代码级别、堆栈跟踪和 SQL 查询详细信息。 · 检测、诊断和终止仍在进行的已挂起或慢速的事务。	✓	–
阈值： 根据主动监视的定义检测特定应用程序行为和条件。	✓	✓
资源组： 按用途对受监视企业中的受管系统进行分类。	✓	✓

通过集成其他产品和组件，如下额外功能部件可供所有产品使用。请参阅第 65 页的『集成』，有关更多详细信息，请参阅第 807 页的『第 8 章 与其他产品和组件集成』。

- Tivoli Monitoring 和 OMEGAMON 代理程序：使用 Hybrid Gateway 可检索监视数据和事件，以使此信息显示在 Cloud APM 控制台中。
- 代理程序共存：在安装 Tivoli Monitoring 代理程序的同一计算机上安装 Cloud APM 代理程序。
- Netcool/OMNIbus 和其他 EIF 接收器：将事件转发到 IBM Tivoli Netcool/OMNIbus。
- Alert Notification：应用程序性能超过阈值时接收通知。
- IBM Control Desk：自动在 Control Desk 中提交凭单。
- IBM Cloud：监视 IBM Cloud 应用程序。

代理程序和数据收集器

IBM Cloud Application Performance Management 代理程序和数据收集器是在产品和附加组件中提供。

您的环境中的许多资源可以被代理程序监视。IBM Cloud 上的一些资源和本地一些资源可以被数据收集器监视。除 J2SE 和 Python 数据收集器以外，所有数据收集器都存在相应的代理程序。要获取代理程序和数据收集器及其描述的列表，请参阅第 49 页的『描述』。要确定每个产品中代理程序或数据收集器可以提供的功能，请参阅第 46 页的『功能』。要查找每个代理程序和数据收集器的变更历史记录，请参阅第 44 页的『变更历史记录』。

您可以安装这些代理程序或数据收集器，这取决于您的环境和需求。数据收集器直接将数据发送到 Cloud APM 服务器。配置代理程序后，数据收集器将数据发送到该代理程序，该代理程序又将数据转发到服务器。数据收集器在应用程序进程空间内运行，而代理程序作为单独的进程在应用程序进程空间外部运行。

在下列情况下安装数据收集器：

- 您需要更简单的安装过程。
- 您使用容器。

在下列情况下安装代理程序：

- 您需要更强的可伸缩性。
- 您想要将套接字限制为从端点到服务器。
- 在阈值编辑器中添加阈值时，您需要清除列表，其中只包含适用于要监视的环境的属性。如果使用数据收集器，您必须从多个数据收集器的属性中进行选择。
- 您想要打开或关闭 UI 上的一些数据收集功能，例如，诊断、事务跟踪或方法跟踪。
- 您想要查看按需诊断数据，例如，未完成的请求和当前时间的堆转储。

变更历史记录

查找有关每个代理程序和数据收集器的版本和变更历史记录的信息。

下表列出代理程序和数据收集器名称以及变更历史记录技术注释链接。单击链接以查看变更历史记录详细信息。

表 2. 代理程序和数据收集器变更历史记录

代理程序和数据收集器	链接
Amazon EC2 代理程序	变更历史记录
Amazon ELB 代理程序	变更历史记录
Azure Compute 代理程序	变更历史记录
Cassandra 代理程序	变更历史记录
Cisco UCS agent	变更历史记录
Citrix VDI 代理程序	变更历史记录
DataPower 代理程序	变更历史记录
DataStage 代理程序	变更历史记录
Db2 代理程序	变更历史记录
Hadoop 代理程序	变更历史记录
HMC Base 代理程序	变更历史记录
HTTP Server 代理程序	变更历史记录
IBM Cloud 代理程序	变更历史记录
IBM Integration Bus 代理程序	变更历史记录
Internet Service Monitoring	变更历史记录
J2SE 数据收集器	变更历史记录
JBoss 代理程序	变更历史记录
Liberty 数据收集器	变更历史记录
Linux KVM 代理程序	变更历史记录
Linux OS 代理程序	变更历史记录
MariaDB 代理程序	变更历史记录
Microsoft Active Directory 代理程序	变更历史记录
Microsoft Cluster Server 代理程序	变更历史记录
Microsoft Exchange Server 代理程序	变更历史记录
Microsoft Hyper-V Server 代理程序	变更历史记录
Microsoft IIS 代理程序	变更历史记录
Microsoft .NET 代理程序	变更历史记录
Microsoft Office 365 代理程序	变更历史记录
Microsoft SharePoint Server 代理程序	变更历史记录
Microsoft SQL Server 代理程序	变更历史记录
MongoDB 代理程序	变更历史记录

表 2. 代理程序和数据收集器变更历史记录 (续)	
代理程序和数据收集器	**链接**
MQ Appliance 代理程序	变更历史记录
MySQL 代理程序	变更历史记录
NetApp Storage 代理程序	变更历史记录
Node.js 代理程序	变更历史记录
Node.js 数据收集器	变更历史记录
OpenStack 代理程序	变更历史记录
Oracle Database 代理程序	变更历史记录
PHP 代理程序	变更历史记录
PostgreSQL 代理程序	变更历史记录
Python 数据收集器	变更历史记录
RabbitMQ 代理程序	变更历史记录
Response Time Monitoring Agent	变更历史记录
Ruby 代理程序	变更历史记录
Ruby 数据收集器	变更历史记录
SAP 代理程序	变更历史记录
SAP HANA Database 代理程序	变更历史记录
SAP NetWeaver Java Stack 代理程序	变更历史记录
Siebel 代理程序	变更历史记录
Skype for Business Server 代理程序	变更历史记录
Sterling Connect Direct 代理程序	变更历史记录
Sterling File Gateway 代理程序	变更历史记录
Sybase 代理程序	变更历史记录
Synthetic Playback 代理程序	变更历史记录
Tomcat 代理程序	变更历史记录
UNIX OS 代理程序	变更历史记录
VMware VI 代理程序	变更历史记录
WebLogic 代理程序	变更历史记录
WebSphere Applications 代理程序	变更历史记录
WebSphere Infrastructure Manager 代理程序	变更历史记录
WebSphere MQ 代理程序	变更历史记录
Windows OS 代理程序	变更历史记录

功能

代理程序和数据收集器的功能随产品的不同而有所变化。代理程序和数据收集器的主要功能是资源监视、事务跟踪和诊断。您可以在 IBM Cloud Application Performance Management 中预订任何产品和附加组件。附加组件需要特定产品。

每个代理程序和数据收集器都监视该代理程序或数据收集器的名称所指的资源，例如，Monitoring Agent for Cisco UCS 监视 Cisco UCS 资源。

根据您是开发人员、运营人员还是业务线所有者，您使用不同的 Cloud APM 功能。

- 资源监视功能包括响应时间监视、应用程序资源监视和基础结构资源监视。所有代理程序和数据收集器都可以提供资源监视功能。
- 事务跟踪功能提供事务实例和拓扑信息。
- 诊断功能包括跟踪和分析单个请求，并在必要时跟踪和分析方法调用。

切记: 资源监视功能对所有产品和附加组件通用。诊断功能和事务跟踪功能仅在 Cloud APM, Advanced 产品和附加组件中可用。

您可从“**产品和服务**”下载所要监视的应用程序的相应代理程序和数据收集器。代理程序需要几分钟时间来完成安装。数据收集器无需安装，您只需在下载完成后对其进行配置。 有关安装代理程序的指示信息，请参阅第 101 页的『第 6 章 安装代理程序』。

第 46 页的表 3 提供全面的代理程序和数据收集器列表，其中显示哪个产品或附加组件包含代理程序或数据收集器，并显示了代理程序或数据收集器的功能。针对代理程序或数据收集器标注了附加组件（例如 Infrastructure Extension Pack）时，表示这些附加组件为必需。 支持事务跟踪和/或诊断功能的代理程序和数据收集器也在 Cloud APM, Advanced 列中进行标注。

✔ 指示代理程序或数据收集器在产品中可用并且可提供资源监视功能。
— 指示数据或功能在此产品中不可用，或者代理程序或数据集收集器不需要附加组件。
TT 指示事务跟踪。
DD 指示诊断。

表 3. 每个产品中的代理程序和数据收集器功能

代理程序和数据收集器	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	附加组件（如果需要）
Amazon EC2 代理程序	✔	✔	Infrastructure Extension Pack
Amazon ELB 代理程序	✔	✔	Infrastructure Extension Pack
Azure Compute 代理程序	✔	✔	Infrastructure Extension Pack
Cassandra 代理程序	✔	✔	Base Extension Pack
Cisco UCS agent	✔	✔	—
Citrix VDI 代理程序	✔	✔	Infrastructure Extension Pack
Db2 代理程序	✔	✔	—

表 3. 每个产品中的代理程序和数据收集器功能 (续)

代理程序和数据收集器	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	附加组件（如果需要）
DataPower 代理程序	✓	✓ TT	—
DataStage 代理程序	✓	✓	Base Extension Pack
Hadoop 代理程序	✓	✓	Base Extension Pack
HMC Base 代理程序	✓	✓	—
HTTP Server 代理程序	✓	✓ TT	—
IBM Cloud 代理程序	✓	✓	Infrastructure Extension Pack
IBM Integration Bus 代理程序	—	✓ TT	—
Internet Service Monitoring	—	—	Base Extension Pack
适用于本地部署应用程序的 J2SE 数据收集器	—	✓ TT DD	—
JBoss 代理程序	✓	✓ TT DD	—
Liberty 数据收集器 适用于 IBM Cloud 和本地部署应用程序	—	✓ TT DD	—
Linux KVM 代理程序	✓	✓	—
Linux OS 代理程序	✓	✓	—
Microsoft Active Directory 代理程序	✓	✓	—
Microsoft Cluster Server 代理程序	✓	✓	—
Microsoft Exchange Server 代理程序	✓	✓	—
Microsoft Hyper-V Server 代理程序	✓	✓	—
Microsoft IIS 代理程序	✓	✓	—
Microsoft .NET 代理程序	✓	✓ TT DD	—

表 3. 每个产品中的代理程序和数据收集器功能 (续)

代理程序和数据收集器	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	附加组件（如果需要）
Microsoft Office 365 代理程序	✓	✓	Base Extension Pack
Microsoft SharePoint Server 代理程序	✓	✓	—
Microsoft SQL Server 代理程序	✓	✓	—
MongoDB 代理程序	✓	✓	—
MQ Appliance 代理程序	—	✓	—
MySQL 代理程序	✓	✓	—
NetApp Storage 代理程序	✓	✓	—
Node.js 代理程序	✓	✓ DD	—
Node.js 数据收集器 适用于 IBM Cloud 和本地部署应用程序	—	✓ TT DD	—
OpenStack 代理程序	✓	✓	—
Oracle Database 代理程序	✓	✓	—
PHP 代理程序	✓	✓	—
PostgreSQL 代理程序	✓	✓	—
Python 数据收集器 适用于 IBM Cloud 和本地部署应用程序	✓	✓ DD	—
RabbitMQ 代理程序	—	✓	Advanced Extension Pack
Response Time Monitoring Agent	✓	✓ TT	—
Ruby 代理程序	✓	✓ DD	—
Ruby 数据收集器 适用于 IBM Cloud 应用程序	—	✓ DD	—
SAP 代理程序	—	✓	—
SAP HANA Database 代理程序	—	✓	Advanced Extension Pack
SAP NetWeaver Java Stack 代理程序	—	✓ TT DD	Advanced Extension Pack
Siebel 代理程序	✓	✓	—

表 3. 每个产品中的代理程序和数据收集器功能 (续)

代理程序和数据收集器	Cloud APM, BaseCloud APM, Base Private	Cloud APM, Advanced	附加组件（如果需要）
Skype for Business Server 代理程序（旧称 Microsoft Lync Server 代理程序）	✓	✓	—
Sterling Connect Direct 代理程序	✓	✓	Base Extension Pack
Sterling File Gateway 代理程序	✓	✓	Base Extension Pack
Sybase 代理程序	✓	✓	—
Tomcat 代理程序	✓	✓ TT	—
UNIX OS 代理程序	✓	✓	—
VMware VI 代理程序	✓	✓	—
WebLogic 代理程序	✓	✓ TT DD	—
WebSphere Applications 代理程序	✓	✓ TT DD	—
WebSphere Infrastructure Manager 代理程序	✓	✓	—
WebSphere MQ 代理程序	—	✓ TT	—
Windows OS 代理程序	✓	✓	—

有关缺省情况下是否对代理程序或数据收集器启用事务跟踪或诊断功能的更多信息，请参阅代理程序和数据收集器表的事务跟踪功能启用。有关预定义诊断仪表板的信息，请参阅代理程序和数据收集器的诊断仪表板。

描述

代理程序和数据收集器的描述提供其中每个组件监视内容的信息以及指向每个组件的更多信息的链接。

每个代理程序和数据收集器都具有版本号，每次更新代理程序或数据收集器时，版本号也将更改。在任何发行版中，可能会添加新的代理程序和数据收集器，并可能更新现有的代理程序和数据收集器。如果您没有最新版本的代理程序或数据收集器，请考虑进行更新。有关如何检查环境中代理程序版本或数据收集器版本的信息，请参阅代理程序版本命令。

每个代理程序和数据收集器描述都包含了链接，它们指向有关这些组件的下列各类详细信息：

- 有关特定代理程序或数据收集器功能的代理程序或数据收集器配置和其他信息
- 参考 PDF，其中包含 Cloud APM 代理程序或数据收集器仪表板、组窗口小部件、阈值、数据集和属性（指标和 KPI）的描述

要获取可与 Cloud APM V8 代理程序和数据收集器共存的 IBM Tivoli Monitoring V6 和 V7 代理程序的文档链接，请参阅第 809 页的表 236。

Amazon EC2 监视

Monitoring Agent for Amazon EC2 为您提供集中监视点，用于监视 Amazon Elastic Compute Cloud (EC2) 实例的运行状况、可用性和性能。此代理程序提供了一组全面的度量，以帮助您就 EC2 环境（包括 CPU 利用率、弹性块存储器 (EBS) 利用率、网络利用率、Amazon Web Service (AWS) 维护更新和磁盘性能）做出明智决策。

- 有关在安装之后配置代理程序的信息，请参阅第 161 页的『配置 Amazon EC2 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Amazon EC2 代理程序 参考。

AWS 弹性负载均衡器监视

Amazon ELB 代理程序为您提供集中监视点，用于监视 AWS 弹性负载均衡器的运行状况、可用性和性能。代理程序显示每个负载均衡器类型（应用程序、网络和传统）的一组综合度量值，用于帮助您做出有关 AWS 弹性负载均衡器环境的明智决定。

- 有关在安装之后配置代理程序的信息，请参阅第 168 页的『配置 AWS 弹性负载均衡器监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Amazon ELB 代理程序 参考。

Azure Compute 监视

Azure Compute 代理程序为您提供集中监视点，用于监视 Azure Compute 实例的运行状况、可用性和性能。此代理程序会显示一组综合度量值，帮助您做出有关 Azure Compute 环境的明智决定。这些度量包括 CPU 使用率、网络使用率和磁盘性能。

- 有关在安装之后配置代理程序的信息，请参阅第 173 页的『配置 Azure Compute 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Azure Compute 代理程序 参考。

Cassandra 监视

Monitoring Agent for Cassandra 为您提供用于监视 Cassandra 集群的功能。您可收集并分析 Cassandra 集群中节点、键空间和列系列的相关信息。

- 有关在安装之后配置代理程序的信息，请参阅第 182 页的『配置 Cassandra 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Cassandra 代理程序 参考。

Cisco UCS 监视

Monitoring Agent for Cisco UCS 提供了用于监视 Cisco UCS 的运行状况、网络和性能的环境。Cisco UCS 代理程序提供一种综合方法，来收集和分析及早检测到问题并进行语法所需的 Cisco UCS 特有信息。

- 有关在安装之后配置代理程序的信息，请参阅第 184 页的『配置 Cisco UCS 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Cisco UCS agent 参考。

Citrix Virtual Desktop Infrastructure 监视

Monitoring Agent for Citrix Virtual Desktop Infrastructure 为您提供集中监视点，用于监视 Citrix Virtual Desktop Infrastructure 的运行状况、可用性和性能。此代理程序会显示一组综合度量值，帮助您做出有关 XenDesktop 或 XenApp 资源（包括站点、机器、应用程序、桌面、会话、用户等）的明智决定。

- 有关在安装之后配置代理程序的信息，请参阅第 190 页的『配置 Citrix Virtual Desktop Infrastructure 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Citrix VDI 代理程序 参考。

DataPower 监视

Monitoring Agent for DataPower 提供了针对企业环境中的 DataPower 设备的集中监视点。您可以识别设备常见问题并接收相应的通知。代理程序还提供有关设备的性能、资源和工作负载的信息。

- 有关在安装之后配置代理程序的信息，请参阅第 207 页的『配置 DataPower 代理程序』。
- 有关仪表板、阈值和属性的信息，请参阅 DataPower 代理程序参考。
- 有关作为 IBM 集成堆栈 的一部分监视 DataPower 应用程序的信息，请参阅第 81 页的『监视 IBM 集成堆栈』。

Db2 监视

Monitoring Agent for Db2 针对 Db2 环境提供集中监视点。您可以从单个 IBM Performance Management 控制台监视大量服务器，每个服务器由一个 Db2 代理程序监视。您可以收集和分析与应用程序、数据库和系统资源相关的信息。

- 有关升级到新版本代理程序之前的信息，请参阅第 964 页的『AIX 上的代理程序：在升级前停止代理程序并运行 slibclean』
- 有关在安装之后配置代理程序的信息，请参阅第 210 页的『配置 Db2 监视 』。
- 有关仪表板、阈值和属性的信息，请参阅 Db2 代理程序 参考。
- 有关作为 IBM Java 应用程序堆栈 的一部分监视数据库事务的信息，请参阅第 75 页的『监视 IBM Java 应用程序堆栈』。

Hadoop 监视

Monitoring Agent for Hadoop 提供用于监视组织中的 Hadoop 集群的功能。您可以使用该代理程序来收集和分析有关 Hadoop 集群的信息，例如数据节点和 Java 虚拟机的状态、内存堆和非堆信息以及有关 Hadoop 节点、文件系统和队列的信息。

- 有关在安装之后配置代理程序的信息，请参阅第 219 页的『配置 Hadoop 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Hadoop 代理程序 参考。

HMC Base 监视

Monitoring Agent for HMC Base 为您提供用于监视 Hardware Management Console (HMC) 的功能。代理程序监视以下 HMC 资源的可用性和运行状况：CPU、内存、存储器和网络。代理程序还报告 Power 服务器、CPU 池和 LPAR 的 HMC 库存与配置。通过使用 HMC 性能样本数据，可监视 Power 服务器、LPAR 和池的 CPU 使用率。

- 有关在安装之后配置代理程序的信息，请参阅第 226 页的『配置 HMC Base 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 HMC Base 代理程序 参考。

HTTP Server 监视

Monitoring Agent for HTTP Server 收集有关 IBM HTTP Server 的性能数据。例如，将显示服务器信息，如服务器的状态和类型、服务器错误数，以及登录服务器的成功和失败次数。数据收集器可收集发送到 HTTP Server 代理程序的数据。代理程序与其监视的 IBM HTTP Server 在同一系统上运行。每个受监视服务器都会注册为子节点。IBM HTTP Server Response Time 模块随着 HTTP Server 代理程序一起安装。当您将 HTTP Server 代理程序与 Response Time Monitoring 代理程序、WebSphere Application 代理程序和数据库代理程序一起使用时，您可以看到从浏览器到数据库的 IBM Java 应用程序堆栈事务监视信息。

- 在开始安装代理程序之前，请参阅 AIX 系统上的安装前任务 - HTTP Server 代理程序 和 Linux 系统上的安装前任务 - HTTP Server 代理程序。
- 有关在安装代理程序之后如何复查数据收集器设置并激活数据收集器的指示信息，请参阅第 231 页的『配置 HTTP Server 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 HTTP Server 代理程序 参考。
- 有关作为 IBM Java 应用程序堆栈 的一部分监视 HTTP Server 事务的信息，请参阅第 75 页的『监视 IBM Java 应用程序堆栈』。

IBM Cloud 监视

Monitoring Agent for IBM Cloud 可从 IBM Cloud (Softlayer) 帐户收集虚拟机库存和度量值。IBM Cloud 代理程序用于跟踪 IBM Cloud 中已配置并且正在运行的虚拟设备数量。您可以在详细仪表板页面中查看分配给每个虚拟设备的资源，该页面还显示诸如设备所在的数据中心、操作系统以及相应月份的预计公共网络带宽之类的信息。

- 有关在安装后配置代理程序的信息，请参阅配置 IBM Cloud 监视。
- 有关仪表板、阈值和属性的信息，请参阅 IBM Cloud 代理程序 参考。

IBM Integration Bus 监视

Monitoring Agent for IBM Integration Bus 是一种监视和管理工具，它提供多种方法来验证、分析和调整与 IBM WebSphere Message Broker 和 IBM Integration Bus 产品关联的消息代理拓扑。

- 有关在安装之后配置代理程序的信息，请参阅第 237 页的『配置 IBM Integration Bus 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 IBM Integration Bus 代理程序 参考。
- 有关作为 IBM 集成堆栈 的一部分监视 IBM Integration Bus 代理的信息，请参阅第 81 页的『监视 IBM 集成堆栈』。

InfoSphere DataStage 监视

InfoSphere DataStage 监视代理程序监视 DataStage Server 的可用性、资源使用情况和性能。此代理程序监视引擎节点和作业的运行状态。您可以分析该代理程序所收集到的信息，并采取相应措施以解决 DataStage Server 中的问题。

· 有关在安装后配置代理程序的信息，请参阅配置 InfoSphere DataStage 监视。

· 有关仪表板、阈值和属性的信息，请参阅 DataStage 代理程序 参考。

Internet Service Monitoring

Internet Service Monitoring 用于确定是否充分执行了特定服务，识别问题区域以及报告根据服务级别协议测量的服务性能。Internet Service Monitoring 代理程序 通过模仿实际用户的操作来工作。它定期轮询或测试因特网服务来检查其状态和性能。

· 有关在安装后配置代理程序的信息，请参阅第 385 页的『在 Windows 系统上配置代理程序』

· 有关仪表板、阈值和属性的信息，请参阅 Internet Service Monitoring 代理程序参考

J2SE 数据收集器监视

J2SE 数据收集器收集针对 Java 应用程序的资源监视和深入诊断数据。深入诊断数据基于请求和汇总消息显示在仪表板中以支持各种向下钻取视图。资源监视和深入诊断均受支持，这有助检测、隔离和诊断 Java 应用程序的问题。您可以配置数据收集器以诊断慢请求。

· 有关配置数据收集器的信息，请参阅配置 J2SE 数据收集器。

· 有关仪表板、阈值和属性的信息，请参阅 J2SE 数据收集器参考。

JBoss 监视

Monitoring Agent for JBoss 监视 JBoss 应用程序服务器和 JBoss Enterprise Application 平台的资源。使用随 JBoss 代理程序一起提供的仪表板可识别最慢的应用程序、最慢的请求、线程池瓶颈、JVM 堆内存和垃圾回收问题、最忙碌的会话以及 JBoss 应用程序服务器的其他瓶颈。

· 有关在安装之后配置代理程序的信息，请参阅第 394 页的『配置 JBoss 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 JBoss 代理程序 参考。

Linux KVM 监视

Monitoring Agent for Linux KVM 是多实例、多连接的代理程序，它支持连接到基于 Enterprise Linux 的 KVM 系统管理程序以及 Red Hat Enterprise Virtualization Manager (RHEV-M) 环境。您可以创建此代理程序的多个实例，以用于监视 RHEV-M 或 KVM 系统管理程序环境中的多个系统管理程序。您还可以在不同虚拟机之间监视虚拟化工作负载并分析资源容量。要将代理程序连接到 KVM 管理程序环境中的虚拟机，必须安装必备软件：`libvirt*.rpm` 和 Korn Shell Interpreter (pdksh)。此代理程序将与用于管理虚拟机的 libvirt 系统管理程序建立远程连接，从而收集度量。

· 有关在安装之后配置代理程序的信息，请参阅第 409 页的『配置 Linux KVM 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Linux KVM 代理程序 参考。

Linux 操作系统监视

Monitoring Agent for Linux OS 提供了针对 Linux 操作系统环境的可用性、性能和资源使用情况的监视功能。此代理程序支持 Docker 容器监视。例如，显示与 Docker 容器有关的详细信息，例如，CPU 使用率、内存、网络和 I/O 使用率信息。还将显示在服务器上运行的 Docker 容器的常规信息，例如，Docker 标识和实例名称。另外，可以配置日志文件监视以监视应用程序日志文件。您可以收集和分析特定于服务器的信息，例如操作系统和 CPU 性能、Linux 磁盘信息和性能分析、进程状态分析以及网络性能。

· 有关在安装之后配置日志文件监视的信息，请参阅第 540 页的『配置操作系统代理程序日志文件监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Linux OS 代理程序 参考。

MariaDB 监视

Monitoring Agent for MariaDB 提供针 MariaDB 环境或应用程序的集中管理点。该软件提供了全面的方法来收集所需信息，以便尽早检测到问题并加以防范。 整个系统的信息都是标准化的。您可以从单个控

制台监视多个服务器。 通过使用 Monitoring Agent for MariaDB，您可以轻松地收集和分析特定于 MariaDB 的信息。
有关在安装之后配置代理程序的信息，请参阅第 418 页的『配置 MariaDB 监视』

Microsoft Active Directory 监视
Monitoring Agent for Microsoft Active Directory 提供用于监视组织中的 Active Directory 的功能。您可以使用此代理程序来收集和分析特定于 Active Directory 的信息，例如网络状态、Sysvol 复制、地址簿性能和目录系统使用情况。

· 有关在安装之后配置代理程序的信息，请参阅第 421 页的『配置 Microsoft Active Directory 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft Active Directory 代理程序 参考。

Microsoft Cluster Server 监视
Monitoring Agent for Microsoft Cluster Server 提供用于监视组织中的 Microsoft Cluster Server 的功能。您可以使用 Microsoft Cluster Server 代理程序来收集与集群资源可用性相关的信息，例如集群级别、集群节点、集群资源组、集群资源和集群网络。此代理程序还提供了集群资源使用情况的统计信息，例如处理器使用情况、内存使用情况、磁盘使用情况和网络使用情况。

· 有关在安装之后配置代理程序的信息，请参阅第 426 页的『配置 Microsoft Cluster Server 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft Cluster Server 代理程序 参考。

Microsoft Exchange Server 监视
Monitoring Agent for Microsoft Exchange Server 提供了用于监视组织中 Exchange Server 的运行状况、可用性和性能的功能。您可以使用 Microsoft Exchange Server 代理程序来收集特定于服务器的信息，例如邮件流量、邮箱数据库的状态和客户机的活动。另外，此代理程序还提供高速缓存使用情况、邮件使用情况、数据库使用情况和客户机活动的统计信息，以帮助分析 Exchange Server 的性能。

· 有关在安装之后配置代理程序的信息，请参阅第 428 页的『配置 Microsoft Exchange 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft Exchange Server 代理程序 参考。

Microsoft Hyper-V Server 监视
Monitoring Agent for Microsoft Hyper-V Server 提供用于监视组织中所有 Hyper-V 系统的可用性和性能的功能。Microsoft Hyper-V Server 代理程序 提供了配置信息，例如虚拟机数、虚拟机的状态、分配的虚拟盘数、分配的虚拟内存和分配的虚拟处理器数。此外，此代理程序还提供了物理处理器使用情况、内存使用情况、网络使用情况、逻辑处理器使用情况和虚拟处理器使用情况的统计信息。

· 有关在安装之后配置代理程序的信息，请参阅第 439 页的『配置 Microsoft Hyper-V 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft Hyper-V Server 代理程序 参考。

Microsoft Internet Information Services 监视
Monitoring Agent for Microsoft Internet Information Services 为您提供了用于监视 Microsoft Internet Information Server 的可用性和性能的功能。您可以使用 Microsoft Internet Information Server 代理程序来监视 Web 站点详细信息，例如请求频率、数据传输率、错误统计信息和连接统计信息。

· 有关在安装之后配置代理程序的信息，请参阅第 443 页的『配置 Microsoft IIS 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft IIS 代理程序 参考。

Microsoft .NET 监视
Monitoring Agent for Microsoft .NET 监视基于 Internet Information Services (IIS) 和 Microsoft .NET Framework 资源的 Microsoft .NET 应用程序。数据收集器组件可从入局 HTTP 请求中收集数据。数据收集器会收集方法调用并构造调用树，还会收集请求上下文和堆栈跟踪数据。使用随 Microsoft .NET 代理程序一起提供的仪表板可识别与 Microsoft .NET Framework 相关的问题，以及识别最慢的 HTTP 请求，从中可向下钻取到堆栈跟踪信息以隔离问题。

· 有关在安装之后配置代理程序的信息，请参阅第 450 页的『注册数据收集器』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft .NET 代理程序 参考。

Microsoft Office 365 监视
Monitoring Agent for Microsoft Office 365 为您提供用于监视 Microsoft Office 365 的功能。您可收集并分析 Microsoft Exchange Online、SharePoint Online、Skype for Business 和 OneDrive for Business 的相关信息。

· 有关在安装之后配置代理程序的信息，请参阅第 458 页的『配置 Microsoft Office 365 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft Office 365 代理程序 参考。

Microsoft SharePoint Server 监视

Monitoring Agent for Microsoft SharePoint Server 为您提供用于监视 Microsoft SharePoint Server 的可用性、事件和性能的环境。使用此代理程序可收集来自 Microsoft SharePoint Server 的数据并管理操作。

· 有关在安装之后配置代理程序的信息，请参阅第 463 页的『配置 Microsoft SharePoint Server 监视 』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft SharePoint Server 代理程序 参考。

Microsoft SQL Server 监视

Monitoring Agent for Microsoft SQL Server 为您提供用于监视 Microsoft SQL Server 的功能。Microsoft SQL Server 代理程序提供针对分布式数据库的集中管理点。使用 Microsoft SQL Server 代理程序仪表板可监视所有受监视 SQL Server 实例的可用性、性能、资源使用情况和整体状态。

· 有关在安装之后配置代理程序的信息，请参阅第 465 页的『配置 Microsoft SQL Server 监视 』。

· 有关仪表板、阈值和属性的信息，请参阅 Microsoft SQL Server 代理程序 参考。

MongoDB 监视

Monitoring Agent for MongoDB 提供了针对 MongoDB 部署的使用情况、状态和性能的监视功能。您可以收集和分析各种信息，例如数据库容量使用情况、打开连接的百分比、内存使用情况、实例状态和可视化仪表板中的响应时间。

· 有关在安装之后配置代理程序的信息，请参阅第 491 页的『配置 MongoDB 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 MongoDB 代理程序 参考。

MQ 设备监视

Monitoring Agent for MQ Appliance 提供关注于 MQ 设备上 MQ 设备级别的监视信息，例如，CPU、内存、存储器、传感器和队列管理器摘要信息。

· 有关在安装之后配置代理程序的信息，请参阅第 250 页的『配置 IBM MQ 设备监视』。

· 有关仪表板、阈值和属性的信息，请参阅 MQ 设备代理程序参考。

MySQL 监视

Monitoring Agent for MySQL 提供针对 MySQL 部署的状态、使用情况和性能的监视功能。您可以收集和分析各种信息，例如收到的字节数与发送的字节数、InnoDB 缓冲池页面数和历史性能。

· 在开始安装代理程序之前，请参阅 Linux 系统上的安装前任务 - MySQL 代理程序 或 Windows 系统上的安装前任务 - MySQL 代理程序。

· 有关在安装之后配置代理程序的信息，请参阅第 496 页的『配置 MySQL 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 MySQL 代理程序 参考。

NetApp 存储监视

Monitoring Agent for NetApp Storage 为您提供了通过 NetApp OnCommand Unified Manager (OCUM) 监视 NetApp 存储系统的功能。您可收集并分析 NetApp 存储系统中聚集、节点、磁盘和卷的相关信息。

· 有关在安装之后配置代理程序的信息，请参阅第 498 页的『配置 NetApp Storage 监视』。

· 有关仪表板、阈值和属性的信息，请参阅 NetApp Storage 代理程序 参考。

Node.js 监视

Monitoring Agent for Node.js 或独立的 Node.js 数据收集器可用于度量和收集有关 Node.js 应用程序性能的数据。例如，会监视并存储 HTTP 请求的吞吐量和响应时间以及与资源使用情况相关的其他度量，以供显示和分析。要在 Node.js 代理程序和 Node.js 数据收集器之间进行选择，请参阅第 504 页的『配置 Node.js 监视』以获取指示信息。

Node.js 代理程序

· 在开始安装之前，请参阅 Linux 系统上的安装前任务 - Node.js 代理程序。

· 有关在安装之后配置代理程序的信息，请参阅第 505 页的『配置 Node.js 代理程序』。

· 有关仪表板、阈值和属性的信息，请参阅 Node.js 代理程序 参考。

Node.js 数据收集器（独立）

Node.js 数据收集器监视 IBM Cloud 和本地部署应用程序。资源监视和深入诊断均受支持，这有助检测、隔离和诊断您的应用程序的问题。您可以将数据收集器配置为跟踪各个请求和方法调用的性能，并使用该信息来诊断速度较慢的请求以及相应地采取措施。

IBM Cloud 应用程序

- 有关配置数据收集器的信息，请参阅第 509 页的『针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

本地应用程序

- 有关配置数据收集器的信息，请参阅第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

OpenStack 监视

Monitoring Agent for OpenStack 为您提供了用于监视 OpenStack 应用程序的功能。使用仪表板来查看 OpenStack 应用程序的性能，例如有关 API 端点、SSH 服务器连接、进程和管理程序的信息。

- 有关在安装之后配置代理程序的信息，请参阅第 521 页的『配置 OpenStack 代理程序』。
- 有关仪表板、阈值和属性的信息，请参阅 OpenStack 代理程序 参考。

Oracle 数据库监视

Monitoring Agent for Oracle Database 提供针对 Oracle 数据库的可用性、性能和资源使用情况的监视功能。您可以配置多个 Oracle Database 代理程序实例，以监视不同的 Oracle 数据库。此代理程序还提供远程监视功能。

- 在开始安装代理程序之前，请参阅 AIX 系统上的安装前任务 - Oracle Database 代理程序、Linux 系统上的安装前任务 - Oracle Database 代理程序 或 Windows 系统上的安装前任务 - Oracle Database 代理程序 (Windows)。
- 有关在安装后配置代理程序的指示信息，请参阅第 526 页的『配置 Oracle 数据库监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Oracle Database 代理程序 参考。
- 有关作为 IBM Java 应用程序堆栈 的一部分监视数据库事务的信息，请参阅第 75 页的『监视 IBM Java 应用程序堆栈』。

PHP 监视

Monitoring Agent for PHP 通过 Apache Web 服务器收集 Web 访问度量，并从 MySQL 收集性能统计信息，从而监视 PHP Web 应用程序。此代理程序可发现 Apache 服务器上的所有 WordPress 应用程序，并提供 WordPress 应用程序统计信息。使用 PHP 代理程序可监视 Web 服务器可用性、Apache 服务器状态以及 GET/POST 请求。代理程序仅评估 WordPress 应用程序中 PHP 请求的性能。不会对 CSS 和 JS 装入进行评估。此代理程序不会使用 URL 自变量来标识 URL。

- 有关在安装之后配置代理程序的信息，请参阅第 566 页的『配置 PHP 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 PHP 代理程序 参考。

PostgreSQL 监视

Monitoring Agent for PostgreSQL 通过 JDBC 驱动程序收集 PostgreSQL 度量来监视 PostgreSQL 数据库。此代理程序提供有关系统资源使用情况、数据库容量、使用的连接数、运行中实例的各自状态、操作统计信息、SQL 查询语句响应时间、数据库大小详细信息和锁定信息的数据。

- 有关在安装之后配置代理程序的信息，请参阅第 568 页的『配置 PostgreSQL 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 PostgreSQL 代理程序 参考。

Python 监视

Python 数据收集器监视本地应用程序和 IBM Cloud Python 应用程序。资源监视和深入诊断均受支持，将提供监视数据，例如，CPU 和内存使用情况、垃圾回收以及线程。您可以将数据收集器配置为跟踪各个请求和方法调用的性能，并使用该信息来诊断速度较慢的请求以及相应地采取措施。

IBM Cloud 应用程序

- 有关配置数据收集器的信息，请参阅第 572 页的『针对 IBM Cloud 应用程序配置 Python 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

本地应用程序

- 有关配置数据收集器的信息，请参阅第 576 页的『为本地应用程序配置 Python 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

RabbitMQ 监视

Monitoring Agent for RabbitMQ 为您提供用于监视 RabbitMQ 集群的功能。您可收集并分析 RabbitMQ 集群中节点、队列和通道的相关信息。

- 有关在安装之后配置代理程序的信息，请参阅第 580 页的『配置 RabbitMQ 监视 』。
- 有关仪表板、阈值和属性的信息，请参阅 RabbitMQ 代理程序 参考。

响应时间监视

Response Time Monitoring Agent 使用网络监视来捕获 HTTP 和 HTTPS 事务数据，例如响应时间和状态码。使用 Response Time Monitoring 代理程序可监视 Web 应用程序对于用户的性能和可用性，包括事务请求、应用程序和服务器信息。另外，此代理程序还用于监视设备和会话信息。

- 在开始安装 Response Time Monitoring 代理程序 之前，请参阅 Preinstallation on AIX 系统上的安装前任务 - Response Time Monitoring 代理程序、Linux 系统上的安装前任务 - Response Time Monitoring 代理程序 或 Windows 系统上的安装前任务 - Response Time Monitoring 代理程序。
- 有关在安装之后配置代理程序的信息，请参阅第 586 页的『JavaScript 注入』。
- 有关仪表板、阈值和属性的信息，请参阅事务监视参考。
- 有关作为 IBM Java 应用程序堆栈 的一部分监视 Response Time Monitoring 的信息，请参阅第 75 页的『监视 IBM Java 应用程序堆栈』。

Ruby 监视

Monitoring Agent for Ruby 或独立的 Ruby 数据收集器监视 Ruby on Rails 应用程序的性能，包括请求流量和配置统计信息。您还可以使用诊断功能来更深入地了解每个应用程序。

独立的 Ruby 数据收集器仅监视 IBM Cloud 应用程序。

Ruby 代理程序

- 有关在安装之后配置代理程序的信息，请参阅第 612 页的『配置 Ruby 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Ruby 代理程序 参考。

Ruby 数据收集器（独立）

您可以使用 Ruby 数据收集器来监视 IBM Cloud 应用程序。资源监视和深入诊断均受支持，这有助检测、隔离和诊断您的应用程序的问题。您可以将数据收集器配置为跟踪各个请求和方法调用的性能，并使用该信息来诊断速度较慢的请求以及相应地采取措施。

IBM Cloud 应用程序

- 有关配置数据收集器的信息，请参阅配置 Ruby 数据收集器。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

SAP 应用程序监视

Monitoring Agent for SAP Applications 使您能够监视在高级业务应用程序编程 (ABAP) 堆栈上运行的 SAP 应用程序。该代理程序还监视 SAP Solution Manager（这是 SAP 生命周期管理工具）和 SAP NetWeaver Process Integration (SAP PI)（这是用于 SAP 的企业集成软件）。它提供了一个集中管理点，用于收集及早检测到问题并执行相应步骤以防止问题重现所需的信息。它支持跨 SAP 发行版、应用程序和组件进行有效的系统管理，也支持对底层数据库、操作系统和外部接口进行有效的系统管理。

- 有关在安装之后配置代理程序的信息，请参阅第 621 页的『配置 SAP 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 SAP 代理程序 参考。

SAP HANA 数据库监视

Monitoring Agent for SAP HANA Database 监视 SAP HANA 数据库的可用性、资源使用情况和性能。此代理程序可以监视 HANA 部署方案，例如，“单个主机 - 单个数据库”、“单个主机 - 多个租户数据库”以及“多个主机 - 多个租户数据库”。您可以分析此代理程序所收集到的信息，并采取相应措施来解决 SAP HANA 数据库中的问题。

- 在开始安装代理程序之前，请参阅 AIX 系统上的安装前任务 - SAP HANA Database 代理程序、Linux 系统上的安装前任务 - SAP HANA Database 代理程序 或 Windows 系统上的安装前任务 - SAP HANA Database 代理程序。
- 有关在安装之后配置代理程序的信息，请参阅第 648 页的『配置 SAP HANA 数据库监视』。
- 有关仪表板、阈值和属性的信息，请参阅 SAP HANA Database 代理程序 参考。

SAP NetWeaver Java Stack 监视

Monitoring Agent for SAP NetWeaver Java Stack 监视 SAP NetWeaver Java Stack 的可用性、资源使用情况和性能。此代理程序可以监视 SAP NetWeaver Java Stack 部署方案，例如，“单个主机 - 单个实例”、“单个主机 - 多个实例”、“多个主机 - 单个实例”以及“多个主机 - 多个实例”。您可以分析此代理程序所收集到的信息，并采取相应措施以解决 SAP NetWeaver Java Stack 中的问题。

- 有关在安装之后配置代理程序的信息，请参阅第 650 页的『配置 SAP NetWeaver Java Stack 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 SAP NetWeaver Java Stack 代理程序 参考。

Siebel 监视

Monitoring Agent for Siebel 提供针对 Siebel 资源的集中监视点，其中包括 Siebel 统计信息、用户会话、组件、任务、应用程序服务器、Siebel Gateway Name Server、进程 CPU 和内存使用情况以及日志事件监视。

- 有关在安装之后配置代理程序的信息，请参阅第 656 页的『配置 Siebel 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Siebel 代理程序 参考。

Skype for Business Server（旧称 Microsoft Lync Server）监视

Monitoring Agent for Skype for Business Server 提供了用于监视 Skype for Business Server 的运行状况、可用性和性能的功能。您可以使用 Skype for Business Server 代理程序来收集特定于服务器的信息，例如等待时间、合成事务、调用详细信息记录 (CDR) 服务写操作、调速请求的状态和会话启动协议 (SIP) 对等方。此外，该代理程序还提供即时消息传递和调解服务器的历史使用统计信息，以帮助您分析 Lync Server 或 Skype for Business Server 的性能。

- 有关在安装之后配置代理程序的信息，请参阅第 445 页的『配置 Skype for Business Server（旧称 Microsoft Lync Server）监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Skype for Business Server 代理程序 参考。

Sterling Connect Direct 监视

Monitoring Agent for Sterling Connect Direct 可以监视 Connect Direct 服务器。它会提供服务器的运行状况和性能。另外，还会分析文件传输活动。

- 有关在安装之后配置代理程序的信息，请参阅第 666 页的『配置 Sterling Connect Direct 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Sterling Connect Direct 代理程序 参考。

Sterling File Gateway 监视

Monitoring Agent for Sterling File Gateway 监视 Sterling File Gateway 应用程序，后者使用各种协议、各种文件命名约定和各种文件格式，在内外部合作伙伴之间传输文件。另外，还支持远程监视功能。

- 有关在安装之后配置代理程序的信息，请参阅第 668 页的『配置 Sterling File Gateway 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Sterling File Gateway 代理程序 参考。

Sybase Server 监视

Monitoring Agent for Sybase Server 提供针对分布式数据库的集中管理点。它收集数据库和系统管理员检查 Sybase 服务器系统性能、提早发现问题并进行阻止所需的信息。

- 有关在安装之后配置代理程序的信息，请参阅第 673 页的『配置 Sybase 服务器监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Sybase 代理程序参考.

Tomcat 监视

Monitoring Agent for Tomcat 监视 Tomcat 应用程序服务器的资源。使用随 Tomcat 代理程序一起提供的仪表板可识别最慢的应用程序、最慢的请求、线程池瓶颈、JVM 堆内存和垃圾回收问题、最忙碌的会话以及 Tomcat 应用程序服务器的其他瓶颈。

- 有关在安装之后配置代理程序的信息，请参阅第 681 页的『配置 Tomcat 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Tomcat 代理程序 参考。

UNIX 操作系统监视

Monitoring Agent for UNIX OS 提供了针对 UNIX 操作系统环境的可用性、性能和资源使用情况的监视功能。（仅限 AIX 和 Solaris 操作系统。请参阅 APM Developer Center 中的系统需求。）另外，可以配置日志文件监视以监视应用程序日志文件。您可以收集和分析特定于服务器的信息，例如操作系统和 CPU 性能、UNIX 磁盘信息和性能分析、进程状态分析以及网络性能。

- 有关在安装之后配置日志文件监视的信息，请参阅第 540 页的『配置操作系统代理程序日志文件监视』。
- 有关仪表板、阈值和属性的信息，请参阅 UNIX OS 代理程序 参考。

VMware VI 监视

Monitoring Agent for VMware VI 通过连接到 VMware Virtual Center 来监视 VMware Virtual Infrastructure。您可以使用 VMware VI 代理程序来查看集群的状态摘要，并从单一控制台监视多个组件，例如集群、虚拟机、数据存储器和 ESX 服务器。

- 有关在安装之后配置代理程序的信息，请参阅第 687 页的『配置 VMware VI 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 VMware VI 代理程序 参考。

WebLogic 监视

Monitoring Agent for WebLogic 为您提供了集中监视点，用于监视 WebLogic Server 环境的运行状况、可用性和性能。此代理程序显示一组综合指标，用于帮助您做出有关 WebLogic 资源（包括 Java 虚拟机 (JVM)、Java 消息传递服务 (JMS) 和 Java 数据库连接 (JDBC)）的明智决定。

- 有关在安装之后配置代理程序的信息，请参阅第 694 页的『配置 WebLogic 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 WebLogic 代理程序 参考。

WebSphere Applications 监视

具有嵌入式数据收集器的 Monitoring Agent for WebSphere Applications，或独立的 Liberty 数据收集器监视 WebSphere Application Server 的资源。可以配置以下监视组件，以执行以下操作：

- 通过应用程序服务器上的 JMX 接口收集 PMI 度量以用于资源监视。
- 收集汇总的请求性能指标。
- 跟踪各个请求和方法调用的性能。

监视数据显示在仪表板中。您可以使用提供的仪表板来隔离应用程序服务器的特定问题区域。向下钻取以确定问题是否与某个底层资源有关，或者问题是否与应用程序代码相关。

有关使用代理程序还是某个数据收集器的信息，请参阅第 709 页的『配置 WebSphere Applications 监视』。

WebSphere Applications 代理程序和嵌入式数据收集器

- 有关在安装之后配置代理程序的信息，请参阅第 709 页的『为 WebSphere Applications 代理程序配置数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅 WebSphere Applications 代理程序 参考。
- 有关作为 IBM Java 应用程序堆栈 的一部分监视 WebSphere Application Server 事务的信息，请参阅第 75 页的『监视 IBM Java 应用程序堆栈』。

Liberty 数据收集器（独立）

您可以使用 Liberty 数据收集器来监视 WebSphere Liberty 概要文件（在 IBM Cloud 上），或监视 WebSphere Application Server Liberty（在 Linux for System x 上）。资源监视、诊断和事务跟踪均受支持，这有助于检测、隔离和诊断应用程序的问题。您可以将独立的数据收集器配置为跟踪各个请求和方法调用的性能，并使用该信息来诊断速度较慢的请求以及相应地采取措施。

IBM Cloud 应用程序

- 有关配置数据收集器的信息，请参阅第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

本地应用程序（仅限 Linux for System x）

- 有关配置数据收集器的信息，请参阅第 749 页的『为本地应用程序配置 Liberty 数据收集器』。
- 有关仪表板、阈值和属性的信息，请参阅数据收集器参考。

WebSphere Infrastructure Manager 监视

Monitoring Agent for WebSphere Infrastructure Manager 提供针对 WebSphere Application Server Deployment Manager 和 Node Agent 的监视功能，包括服务器状态、资源和事务。您可以使用 WebSphere Infrastructure Manager 代理程序所收集的数据来分析 Deployment Manager 和 Node Agent 的性能，以及是否发生问题。

- 有关在安装之后配置代理程序的信息，请参阅第 790 页的『配置 WebSphere Infrastructure Manager 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 WebSphere Infrastructure Manager 代理程序 参考。

WebSphere MQ 监视

通过 Monitoring Agent for WebSphere MQ，您可以从单个有利点轻松收集和分析队列管理器的特定于 WebSphere MQ 的数据。然后，您可以使用预定义的仪表板来跟踪所收集数据的趋势，并对系统问题进行故障诊断。

- 有关在安装之后配置代理程序的信息，请参阅第 791 页的『配置 WebSphere MQ 监视』。
- 有关仪表板、阈值和属性的信息，请参阅 WebSphere MQ 代理程序 参考。
- 有关作为 IBM 集成堆栈 的一部分监视消息队列的信息，请参阅第 81 页的『监视 IBM 集成堆栈』。

Windows 操作系统监视

Monitoring Agent for Windows OS 提供了针对 Windows 操作系统环境的可用性、性能和资源使用情况的监视功能。另外，可以配置日志文件监视以监视应用程序日志文件。您可以收集和分析特定于服务器的信息，例如操作系统和 CPU 性能、磁盘信息和性能分析、进程状态分析、因特网会话数据、受监视日志信息、因特网服务器统计信息、消息排队统计信息、打印机和作业状态数据、远程访问服务统计信息和服务信息。KNTCMA_FCProvider 服务随代理程序一起安装。

- 有关在安装之后配置日志文件监视的信息，请参阅第 540 页的『配置操作系统代理程序日志文件监视』。
- 有关仪表板、阈值和属性的信息，请参阅 Windows OS 代理程序 参考。

功能部件

关键功能部件根据产品的不同而不同。某些功能部件在其中一个产品中提供、在两个产品中都提供、在附加组件中提供，或者通过与其他产品和组件集成来提供。

应用程序资源监视

使用资源监视代理程序来监视语言和中间件。覆盖范围因产品而异。请参阅第 46 页的『功能』。

操作系统监视

使用资源监视代理程序可监视 Linux、UNIX 和 Windows 操作系统。请参阅第 46 页的『功能』。

日志文件监视

OS 代理程序包含监视应用程序日志文件的功能部件。此功能部件包括根据正则表达式配置日志文件监视的功能。

为确保兼容性，OS 代理程序使用以下信息和格式：

- IBM Tivoli Monitoring Log File Agent V6.x 使用的配置信息和格式文件
- Tivoli Event Console Log File Adapter 使用的配置信息和格式字符串

这些格式字符串允许代理程序根据格式文件中的模式过滤日志数据，以及仅向事件使用者提交相关数据。OS 代理程序将数据发送到 Cloud APM 服务器，或者通过事件集成工具 (EIF) 发送给任何 EIF 接收方，例如，Netcool/OMNIbus Probe for Tivoli EIF。

仪表板

应用程序性能仪表板为您提供了环境中应用程序的高级状态。通过从导航器中进行选择或者单击摘要框以向下钻取至下一级别，查看感兴趣的区域。

要了解有关每个仪表板级别可用的功能部件的信息，请参阅第 915 页的『我的所有应用程序 - 应用程序性能仪表板』、第 917 页的『应用程序 - 应用程序性能仪表板』 和第 921 页的『组和实例 - 应用程序性能仪表板』。

从 Tivoli Monitoring 和相同仪表板中的 Cloud APM 域查看 KPI。

在同时包含 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 产品的环境中，您可以安装 IBM Cloud Application Performance Management Hybrid Gateway 以提供来自两个域的受管系统的整合视图。要在 Cloud APM 控制台中查看混合环境，必须创建受管系统组，在 Tivoli Monitoring 环境中安装 Hybrid Gateway，并配置与 Hybrid Gateway 的通信。

有关更多信息，请参阅第 807 页的『与 IBM Tivoli Monitoring V6.3 集成 』。

历史度量

在应用程序性能仪表板上，最多可获取 24 小时的历史数据的直观图形。如果仪表板的“**状态概述**”选项卡中显示时间选择器，那么可以调整图表和表（其中的值派生自历史数据样本）的时间范围。对于折线图，您还可以将最多过去 24 小时的当前数据与最多 8 天的历史数据进行比较，从而发现异常。

有关更多信息，请参阅第 924 页的『随着时间变化调整和比较度量值』。

IBM Cloud Application Business Insights Universal View

您可以使用 Universal View 为正在监视的应用程序创建定制页面。从不同图表和度量选项中进行选择以根据您的需求创建用于监视数据的窗口小部件。通过使用 Universal View，您可以定制仪表板以查看来自多个代理程序的组合数据。

当您查看仪表板上的数据时，您可以动态更改图表类型。在网格窗口小部件上，您可以动态过滤数据。

您可以将定制页数据导出到原始数据文件。

有关更多信息，请参阅第 942 页的『定制视图』。

应用程序详细信息

从“**我的所有应用程序**”仪表板向下钻取到受管系统实例的详细仪表板后，将显示“属性详细信息”选项卡，供您创建和管理定制历史折线图和表，这些图表和表可以保存。您可以保存更多图表或表页面以仅供您查看或与同一环境中的所有用户共享。

有关更多信息，请参阅第 925 页的『创建定制图表或表页面』。

API

Cloud APM API 可用于管理环境，例如分配用户角色和创建阈值。有关更多信息，请参阅第 910 页的『浏览 API』。

基于角色的访问控制

在 Cloud APM 中，角色是一组许可权，用于控制可执行的操作。使用“基于角色的访问控制”功能部件可创建定制角色，这是基础安全性。还提供以下四个预定义角色：角色管理员、监视管理员、系统管理员和监视用户。您可以将用户分配到定制角色或预定义角色，并且用户可分配到多个角色。您可以将许可权分配给定制角色，或者可以将多个许可权分配到现有缺省角色。许可权是累积的。为用户分配角色后，该用户即分配有其所有角色的所有许可权。

您可以向单个应用程序、系统资源组和定制资源组分配查看许可权与修改许可权。例如，如果您是某个角色的成员，并且该角色具有应用程序的查看许可权，那么您可以查看该应用程序内的所有支持组件。

您可以向系统管理任务分配查看许可权与修改许可权。例如，如果您是某个角色的成员，并且该角色具有“高级配置”的查看许可权，那么您可以在“**高级配置**”窗口中执行和保存更改。

有关更多信息，请参阅第 851 页的『角色和许可权』。

历史报告
提供有关 WebSphere Applications 代理程序、Response Time Monitoring Agent 和 Synthetic Playback 代理程序所收集的数据的报告。Response Time Monitoring 代理程序报告需要事务跟踪（在 Cloud APM, Base 中不可用） 有关报告描述，请参阅第 951 页的『报告』。

Agent Builder
构建定制代理程序以监视任何平台或技术。请参阅 https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/agent_builder_guide.htm。

数据库资源监视
覆盖范围因产品而异。请参阅第 46 页的『功能』以获取可以监视的数据库的名称。

基础结构资源监视
使用资源监视代理程序来监视系统管理程序、存储器和网络。覆盖范围因产品而异。请参阅第 46 页的『功能』。

商业应用程序资源监视
使用资源监视代理程序来监视业务应用程序和协作应用程序。覆盖范围因产品而异。请参阅第 46 页的『功能』。

响应时间和最终用户体验监视
请查看从基础结构到其设备的用户体验。使用响应时间监视可监视从浏览器到数据库的 Web 站点和 Web 应用程序的性能和可用性，以及监视移动设备。在要监视的任何 Web 服务器上安装 Response Time Monitoring 代理程序 之后，只需稍作配置甚至不需进一步的配置，这些代理程序所收集的数据便会显示在应用程序性能仪表板中。来自 Response Time Monitoring 代理程序 的数据用于“**最终用户事务**”仪表板。在 Cloud APM, Advanced 中，您可以度量来自浏览器的响应时间，也可以将来自 Response Time Monitoring 代理程序 的数据用于“**汇总事务拓扑**”。有关更多信息，请参阅第 74 页的『场景：监视 IBM Java 应用程序堆栈 』。

事务跟踪
此功能通过 Cloud APM, Advanced 提供。事务跟踪功能启用拓扑视图和实例级别事务监视。事务跟踪作为 Cloud APM 服务器 的一部分安装。事务跟踪针对某些代理程序自动启用，但对其他代理程序必须手动启用。第 61 页的表 4 提供有关支持事务跟踪的代理程序的更多信息。

表 *4.* 对代理程序和数据收集器启用事务跟踪

代理程序 或数据收集器	缺省情况下启用	如何启用
DataPower 代理程序	✓	第 209 页的『配置 DataPower 代理程序的事务跟踪』
IBM Integration Bus 代理程序	—	第 248 页的『配置 IBM Integration Bus 代理程序的事务跟踪』 **注:** 如果在 Solaris X86 上部署此代理程序，那么不支持 TT。
J2SE 数据收集器	✓	第 388 页的『配置 J2SE 监视』
JBoss 代理程序	—	第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』
Liberty 数据收集器	✓	第 749 页的『为本地应用程序配置 Liberty 数据收集器』第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』
Microsoft .NET 代理程序	—	第 453 页的『启用事务跟踪和诊断数据收集』
Node.js 数据收集器	—	第 511 页的『针对 IBM Cloud 应用程序定制独立 Node.js 数据收集器』 第 515 页的『为本地应用程序定制 Node.js 数据收集器』
Response Time Monitoring 代理程序 + HTTP Server 代理程序	—	第 584 页的『规划安装 』

表 4. 对代理程序和数据收集器启用事务跟踪 (续)		
代理程序 或数据收集器	**缺省情况下启用**	**如何启用**
SAP NetWeaver Java Stack 代理程序	—	第 654 页的『启用事务跟踪和诊断数据收集』
Tomcat 代理程序	—	第 686 页的『启用事务跟踪和诊断数据收集』
WebLogic 代理程序	—	第 694 页的『配置 WebLogic 监视』
WebSphere Applications 代理程序	—	第 715 页的『以交互方式配置数据收集器』 **注:** 如果在 Solaris X86 上部署此代理程序，那么不支持 TT。
WebSphere MQ 代理程序	—	第 799 页的『配置 WebSphere MQ 代理程序的事务跟踪』 **注:** 如果在 Solaris X86 上部署此代理程序，那么不支持 TT。

对于支持事务跟踪的所有代理程序，“**汇总事务拓扑**”和“**事务实例拓扑**”视图中均显示数据。

应用程序拓扑
查看所有组件在应用程序环境中是如何连接的。有关更多信息，请参阅第 917 页的『应用程序 - 应用程序性能仪表板』。

事务实例拓扑
通过适用于事务的每个实例的环境，使以下路径可视化。有关更多信息，请参阅第 79 页的『事务实例拓扑 』

Availability Monitoring
IBM Cloud Availability Monitoring 从全球多个存在点提供对 Web 应用程序的增强型合成监视。每隔一定时间创建模拟用户行为的合成测试。从公用存在点运行您的测试，或者下载您自己的定制存在点并在本地或专用服务器上对其进行部署。通过图形、明细表和映射视图，使用 Availability Monitoring 仪表板来监视应用程序可用性、性能和警报。使用瀑布分析来确定性能和可用性问题的发生时间并查找这些问题的原因。

有关使用合成测试的更多信息，请参阅第 887 页的『Availability Monitoring』。

深入诊断
对于特定代理程序，您可以从摘要仪表板向下钻取到深入诊断仪表板，并查看各个请求的相关信息。从摘要仪表板向下钻取以查看代码级别、堆栈跟踪和 SQL 查询详细信息。使用诊断仪表板来识别哪些请求存在问题并调试有问题的事务。还可以检测、诊断和终止仍在进行的挂起/缓慢事务（请参阅 WebSphere Applications 代理程序参考）。第 62 页的表 5 提供了有关诊断代理程序的更多信息。

表 5. 代理程序和数据收集器的诊断仪表板				
代理程序或数据收集器	**缺省情况下，配置诊断数据**	**可用的诊断仪表板**	**如何访问诊断仪表板**	**如何配置代理程序或数据收集器以收集诊断数据**
J2SE 数据收集器	✓	详细信息、Web 模块、请求实例、请求摘要、请求跟踪	在“**概述**”仪表板或“**Web 模块**”仪表板中单击**诊断**。	第 388 页的『配置 J2SE 监视』
JBoss 代理程序	—	诊断仪表板、未完成的请求摘要、未完成的请求堆栈跟踪仪表板、JVM 垃圾回收、堆转储、堆转储比较	单击**诊断**、**未完成的请求**、**详细信息**或**堆转储**（在“**概述**”仪表板中）。	第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』

表 5. 代理程序和数据收集器的诊断仪表板 *(续)*

代理程序或数据收集器	缺省情况下，配置诊断数据	可用的诊断仪表板	如何访问诊断仪表板	如何配置代理程序或数据收集器以收集诊断数据
Liberty 数据收集器	✓	详细信息、堆转储、堆转储比较、内存分析	在“**概述**”仪表板中，单击**诊断**、**查看堆转储**或**查看内存分析**。	· 第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』 · 第 749 页的『为本地应用程序配置 Liberty 数据收集器』
Microsoft .NET 代理程序	–	请求实例、请求摘要和请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 453 页的『使用 configdc 命令来启用诊断数据的收集』
Node.js 代理程序	✓	GC 详细信息、请求实例、请求摘要、请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 505 页的『配置 Node.js 代理程序』
Node.js 数据收集器	✓	GC 详细信息、最慢请求实例、请求实例、请求跟踪	单击“概述”仪表板中的**诊断**或 **GC 详细信息**。	· 第 509 页的『针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器』 · 第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』
Python 数据收集器	✓	最慢请求详细信息、请求实例详细信息、请求跟踪详细信息、Python 线程详细信息、Python 垃圾回收、Python 堆详细信息	在“**概述**”仪表板中，单击**诊断**、**线程详细信息**或**内存详细信息**。	· 第 572 页的『针对 IBM Cloud 应用程序配置 Python 数据收集器』 · 第 576 页的『为本地应用程序配置 Python 数据收集器』
Ruby 代理程序	–	请求摘要详细信息、采样请求实例、请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 612 页的『配置 Ruby 监视』
Ruby 数据收集器	✓	请求实例、请求摘要和请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 618 页的『针对 IBM Cloud 应用程序配置 Ruby 数据收集器』

表 5. 代理程序和数据收集器的诊断仪表板 (续)

代理程序或数据收集器	缺省情况下，配置诊断数据	可用的诊断仪表板	如何访问诊断仪表板	如何配置代理程序或数据收集器以收集诊断数据
SAP NetWeaver Java Stack 代理程序	✓	请求实例、请求摘要和请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 654 页的『启用事务跟踪和诊断数据收集』
Tomcat 代理程序	—	请求实例、请求摘要和请求跟踪	在“**概述**”仪表板中单击**诊断**。	第 686 页的『启用事务跟踪和诊断数据收集』
WebLogic 代理程序	—	诊断仪表板、未完成的请求摘要、未完成的请求堆栈跟踪仪表板、JVM GC 详细信息、堆转储、堆转储比较	在“**概述**”仪表板中，单击**诊断**、**查看请求**、**详细信息**或**堆转储**。	第 694 页的『配置 WebLogic 监视』
WebSphere Applications 代理程序	—	诊断、请求实例、请求序列、未完成的请求摘要、未完成的请求堆栈跟踪、堆转储、堆转储比较、内存分析	在“**概述**”仪表板中，单击**诊断**、**查看请求**、**查看堆转储**或**查看内存分析**。 仅在启用内存泄漏监视后，**查看内存分析**按钮才可用。	· 第 712 页的『使用简单配置实用程序来配置数据收集器』 · 第 746 页的『启用内存泄漏监视』

仅当为代理程序配置了深入诊断，并且您是“角色管理员”角色、“监视管理员”角色或对诊断仪表板具有查看许可权的其他一些定制角色的成员时，才启用**诊断**按钮。

阈值

借助阈值，您可以根据主动监视的定义来检测特定应用程序行为和条件。预定义阈值可用于每个代理程序，而且您可以针对监视定义新阈值。有关更多信息，请参阅第 838 页的『阈值管理器』。

在配置事件转发后，事件将发送到 EIF 接收器。您可以使用转发到事件服务器的阈值和事件之间的缺省映射或定制阈值的映射方式。有关更多信息，请参阅第 841 页的『定制事件以转发到 EIF 接收器』。

在**应用程序性能仪表板**中，选择应用程序后，将会显示“**事件**”选项卡。“**事件**”选项卡会显示当前应用程序的打开事件。您可以向下钻取至包含性能指标的详细仪表板以帮助确定事件原因。有关更多信息，请参阅第 939 页的『事件状态』。

资源组

受监视企业中的受管系统可按目的进行分类。此类受管系统通常具有相同的阈值需求。使用“资源组管理器”可将受监视系统分组到可分配事件触发阈值的组。有关更多信息，请参阅第 833 页的『资源组管理器』。

“入门”页面

登录到 Cloud APM 控制台后，将显示一个“入门”页面。单击任何**用户任务**或**管理员任务**可链接到基于场景的教程或视频演示。点击“立即启动”链接将直接进入功能部件，例如，“阈值管理器”。点击**社区资源**链接转至**常见问题及解答**和 Cloud APM 论坛等。

通过与其他产品和组件集成可使用更多功能部件。有关更多信息，请参阅第 65 页的『集成 』和第 807 页的『第 8 章 与其他产品和组件集成』中的更多详细信息。

集成

通过与其他产品和组件集成，提供了额外的功能部件： Tivoli Monitoring、OMEGAMON、Netcool/OMNIbus、Operations Analytics - Log Analysis、Operations Analytics - Predictive Insights、Alert Notification、Control Desk、IBM Cloud 和 Agent Builder。

IBM Tivoli Monitoring

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Cloud Application Performance Management 代理程序。但是，这两种代理程序类型不能安装在同一目录中。有关更多信息，请参阅第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

如果环境中同时包含 IBM Tivoli Monitoring 和 Cloud APM 产品（云端和/或本地），那么您可以安装 IBM Cloud Application Performance Management Hybrid Gateway，以提供来自两个环境的受管系统的整合视图。有关更多信息，请参阅第 811 页的『Hybrid Gateway』。有关受支持的 Tivoli Monitoring 代理程序的最新列表，请参阅 Hybrid Gateway supported agents (APM Developer Center)。

IBM OMEGAMON

z Systems Extension Pack 可将一个或多个在 z Systems 大型机上运行的 OMEGAMON 代理程序连接至 Cloud APM。通过使用 z Systems Extension Pack 和 Hybrid Gateway 将已部署的 OMEGAMON 代理程序连接到 Cloud APM，可在 Cloud APM 控制台中查看 OMEGAMON 应用程序组件的监视数据和事件。

有关更多信息，请参阅第 820 页的『与 OMEGAMON 集成』。

IBM Netcool/OMNIbus

可以将事件从 Cloud APM 转发至内部部署 Netcool/OMNIbus 事件管理器。有关更多信息，请参阅第 821 页的『与 Netcool/OMNIbus 集成』。

IBM Operations Analytics - Log Analysis

环境包含 IBM Operations Analytics - Log Analysis 时，请将应用程序日志和性能数据集中在一起，以帮助查找应用程序所遇到问题的根本原因。您可以搜索与应用程序相关联的日志数据来查找问题的原因，例如缓慢或故障。有关更多信息，请参阅第 825 页的『与 Operations Analytics - Log Analysis 集成』。

IBM Operations Analytics - Predictive Insights

Operations Analytics - Predictive Insights 分析数据并了解系统的正常行为。它创建性能模型并将其用于检测或预测建模范围以外的行为，以及在发生异常行为时生成警报。您可以将 Operations Analytics - Predictive Insights 添加到 Cloud APM 预订。 然后，您可以在 应用程序性能仪表板 中查看异常，并向下钻取至 Operations Analytics - Predictive Insights 用户界面，以查看更多详细信息。有关更多信息，请参阅第 825 页的『与 Operations Analytics - Predictive Insights 集成』。

IBM Cloud

您可以使用独立数据收集器在 IBM Cloud 环境中查看应用程序的监视信息。数据收集器通过将 IBM Cloud 应用程序的资源和深入诊断监视数据传输到 Cloud APM 服务器，支持将监视能力与 IBM Cloud 集成。Cloud APM 服务器接收并处理数据收集器所收集的监视信息。可以监视下列类型的 IBM Cloud 应用程序：

- Liberty 应用程序
- Node.js 应用程序
- Python 应用程序
- Ruby 应用程序

配置数据收集器之后，您可在 Cloud APM 控制台上查看监视数据。有关更多信息，请参阅第 157 页的『用于配置数据收集器的常规过程』。

IBM Alert Notification

如果使用 IBM Cloud Application Performance Management，将自动为您集成 IBM Alert Notification。Alert Notification 是一种易于使用的简单通知系统，可以向 IT 人员提供有关 IT 操作环境中所发生问题的即时警告通知。从代理程序收到的数据提供警报源。在激活 Alert Notification 后，将其连接到 Cloud APM 实例。作为独立包，您可以将 Alert Notification 与可实现和调用 REST API 的任何内部部署监视工具集成。支持的工具包括 IBM Tivoli Netcool/OMNIbus。有关更多信息，请参阅与 Alert Notification 集成。

IBM Control Desk
通过将支持凭单提交给 IBM 支持，获取 IBM Control Desk 集成。您可以配置 Cloud APM 事件以自动在 IBM Control Desk 中打开凭单。转至 IBM Support 并选择**预订**。有关支持人员配置此集成所需详细信息的更多信息，请参阅第 827 页的『与 Control Desk 集成』。

IBM Agent Builder
您可以使用 Agent Builder 来针对任何技术构建定制代理程序。 有关更多信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/agent_builder_guide.htm。

文档

您可以在 IBM Knowledge Center、Cloud APM 控制台和 Application Performance Management Developer Center 中找到 IBM Cloud Application Performance Management 的资料。

IBM Knowledge Center
IBM Knowledge Center 中的 Cloud APM 是产品技术信息的官方来源。

用户界面帮助
在登录到 Cloud APM 控制台或浏览 Guided Demo 时，可以访问帮助系统：

- 单击导航栏中的**帮助内容**。“帮助”菜单。
- 单击应用程序性能仪表板条幅中的 。
- 单击“系统配置”页面中的**了解更多**链接。
- 单击仪表板窗口小部件中的 。

IBM Application Performance Management Developer Center
Application Performance Management Developer Center 是适用于各种 APM 用户的各种 APM 内容的中央位置。这些内容包括文档、博客、视频以及指向其他资源的链接。

IBM Cloud Application Performance Management 论坛和 dwAnswers
Cloud Application Performance Management 论坛和 dwAnswers 包含有关产品问题的技术讨论，包括问题诊断和解决方案。

信息也在下列 Web 站点上提供：

软件产品兼容性报告 (SPCR) 工具
您可以使用 SPCR 工具生成与产品需求相关的各种类型的报告。搜索其中一个 Cloud Application Performance Management 产品名或搜索 IBM Cloud Application Performance Management - 代理程序。

IBM Marketplace
视频演示和 FAQ 之类的资源在 IBM Marketplace 中提供。

IBM API Explorer
有关 Cloud APM API 的文档，请参阅第 910 页的『浏览 API』。

IBM Terminology
IBM 术语 Web 站点包含与 IBM 产品相关的术语，这些术语整合在一个方便的位置。

IBM Redbooks®
IBM Redbooks Web 站点包含 Redbooks 出版物、Redpapers 和 Redbooks 技术说明，用于从平台和解决方案角度提供有关产品的信息。

本文档中使用的约定

在本文档中，对取决于操作系统的特殊术语、操作、命令和路径以及特定于平台和特定于产品的信息使用了若干约定。

字型约定

本文档中使用了以下字型约定：

粗体

- 若不使用粗体则很难与周围文本区分开来的小写命令、大小写混合命令、参数和环境变量
- 界面控件（复选框、按钮、单选按钮、旋钮、字段、文件夹、图标、列表框，列表框中的项、多列列表、容器、菜单选项、菜单名称、选项卡、属性表）和标签（例如**提示：**）
- 文本中的关键字和参数

斜体

- 引用（示例：出版物书名、软盘和 CD）
- 文本中定义的词和短语（示例：非交换线路称为点到点线路）
- 强调词和字母（示例：LUN 地址必须以字母 *L* 开头。）
- 文本中的新术语，但定义列表中的除外（示例：视图是工作空间中包含数据的框架。）
- 必须提供的变量和值（例如：其中 *myname* 表示...）

等宽字体

- 示例和代码示例
- 文件名、目录名、路径名、编程关键字、属性和其他难以与周围文本区分的元素
- 消息文本和提示
- 必须输入的文本
- 参数或命令选项的值

粗体等宽字体

- 命令名以及可作为命令输入的宏和实用程序的名称
- 文本中的环境变量名称
- 关键字
- 文本中的参数名称：API 结构参数、命令参数和自变量以及配置参数
- 进程名称
- 文本中的注册表变量名称
- 脚本名称

与操作系统相关的变量和路径

目录路径的斜杠方向在文档中可能会变化。无论您在文档中看到的斜杠方向如何，都请遵循以下准则：

- Linux AIX 使用正斜杠 (**/**)。
- Windows 使用反斜杠 (****)。

环境变量的名称在 Windows 和 AIX 中未必总是相同。例如，Windows 中的 %TEMP% 相当于 AIX 或 Linux 中的 $TMPDIR。

对于环境变量，请遵循以下准则：

- Linux AIX 使用 **$***variable*。
- Windows 使用 **%***variable***%**。

Windows 如果在 Windows 系统上使用了 bash shell，那么可以使用 AIX 约定。

代理程序的安装目录变量和路径

install_dir 是代理程序的安装目录。缺省位置取决于操作系统：

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

第 4 章 规划部署

为确保 IBM Cloud Application Performance Management 部署成功，规划至关重要。

系统需求

对于 IBM Cloud Application Performance Management 代理程序和数据收集器，支持各种操作系统，且每个组件都有特定的需求。

时区

在受管系统上使用网络时间协议 (NTP) 以确保时间准确。设置时间，使其与服务器的物理位置相匹配，例如 UTC-03:00（对于巴西利亚）和 UTC +06:30（对于仰光），这有助于确保事件和事务的精确时间戳记。 代理程序以用户的本地时间报告 APM UI 中的数据。

Cloud APM 代理程序和数据收集器需求

获取有关您计划安装的每个监视代理程序和独立数据收集器需求的信息。

Cloud APM 代理程序和数据收集器通常对系统管理程序透明，这意味着它们可以安装并部署在任何受支持的操作系统上，而与托管操作系统的系统管理程序（例如，Hyper-V、IBM PowerVM、KVM、VMWare ESX，等等）无关。

有关代理程序和数据收集器的需求，请使用 System requirements (APM Developer Center) 的“组件报告”部分中的链接。

有关受支持浏览器的最新信息，请参阅 IBM Cloud Application Performance Management 详细系统需求报告。

您还可以在 软件产品兼容性报告工具中搜索 IBM Cloud Application Performance Management。

如果代理程序会将全球化数据发送至 Cloud APM 服务器，那么安装该代理程序的本地计算机系统必须支持 UTF-8 编码。

代理程序和数据收集器使用的缺省端口。

在 Cloud APM 组件和受监视的应用程序/系统（本地或远程）之间的通信使用各种端口。在大多数情况下，提供了缺省端口以促进配置。大多数缺省值都可以使用配置参数进行定制。

第 69 页的表 6 列出了 Cloud APM 代理程序和数据收集器用来与所监视的应用程序或系统通信的缺省端口。表中的“不适用”表示以下某种情况：

- 代理程序或数据收集器不使用任何端口与受监视的应用程序或系统进行通信。
- 用于通信的端口由受监视应用程序的配置确定。
- 代理程序或数据收集器使用的端口是动态分配的，不提供静态缺省值。
- 要使用的所有端口必须由用户指定，不提供缺省值。

表 6. 代理程序和数据收集器使用的缺省端口。

代理程序和数据收集器	缺省端口	可配置	本地	远程
Amazon EC2 代理程序	· TCP 端口 80（用于 HTTP） · TCP 端口 443（用于 HTTPS）	不适用	是	否
Amazon ELB 代理程序	· TCP 端口 80（用于 HTTP） · TCP 端口 443（用于 HTTPS）	不适用	否	否

© Copyright IBM Corp. 2014, 2019

表 6. 代理程序和数据收集器使用的缺省端口。 *(续)*

代理程序和数据收集器	缺省端口	可配置	本地	远程
Azure Compute 代理程序	· TCP 端口 80（用于 HTTP） · TCP 端口 443（用于 HTTPS）	否	否	否
Cassandra 代理程序	7199（用于本地和远程 JMX 服务器）	是	是	是
Cisco UCS agent	· TCP 端口 80（用于 HTTP） · TCP 端口 443（用于 HTTPS）	否	是	否
Citrix VDI 代理程序	对于 PowerShell 调用： · 5985（用于 HTTP） · 5986（用于 HTTPS）	是	是	是
Db2 代理程序	· 50000（Db2 服务器的缺省端口） · 支持远程监视：使用用户给予的端口号对远程服务器实例进行编目。	是	是	是
DataPower 代理程序	5550（用于连接到远程 DataPower 设备）	是	否	是
Hadoop 代理程序	· 本地监视：**CP_PORT** 环境变量值 · 远程监视： – 50070（备用 NameNode） – 50090（辅助 Namenode） – 8088（资源管理器） – 19888（作业历史记录服务器） – 8080 (Ambari)	是	是	是
HMC Base 代理程序	12443（用于从 HMC 下载 SDK）	否	是	否
HTTP Server 代理程序	HTTP Server 可配置到不同端口，但代理程序本身不含缺省端口。	不适用	是	否
IBM Cloud 代理程序	指向 api.softlayer.com 端口 443 的传出连接。	不适用	否	是
IBM Integration Bus 代理程序	不适用	不适用	是	否
Internet Service Monitoring	对于数据网桥： · 9510 · 9520	是	否	是
DataStage 代理程序	· 9443（WAS HTTPS 端口） · 50000（数据库 JDBC 端口） · 1433 (Microsoft SQL) · 1521 (Oracle)	是	是	是
J2SE 数据收集器	不适用	不适用	否	否

表 6. 代理程序和数据收集器使用的缺省端口。 *(续)*

代理程序和数据收集器	缺省端口	可配置	本地	远程
JBoss 代理程序	因 JBoss 服务器版本而异: · 9990 · 9994 · 9999	否	是	否
Liberty 数据收集器	不适用	不适用	否	否
Linux KVM 代理程序	· 8080（用于 HTTP） · 8443（用于 HTTPS）	是	是	否
Linux OS 代理程序	22（用于通过 SSH 进行远程日志监视）	是	是	否
MariaDB 代理程序	3306	是	是	是
Microsoft Active Directory 代理程序	端口号取决于监视使用的侦听器设置。	不适用	是	是
Microsoft Cluster Server 代理程序	不适用	不适用	否	否
Microsoft Exchange Server 代理程序	不适用	不适用	否	否
Microsoft Hyper-V Server 代理程序	不适用	不适用	否	否
Microsoft IIS 代理程序	不适用	不适用	否	否
Microsoft .NET 代理程序	要发送事务跟踪数据，缺省情况下使用端口 5456。	是	是	否
Microsoft Office 365 代理程序	7799（用于 Skype 综合事务）	是	是	否
Microsoft SharePoint Server 代理程序	1433（用于 SQL Server）	否	是	是
Microsoft SQL Server 代理程序	1433（SQL Server 的缺省值）	是（通过 **COLL_PORT**）	是	否
MQ Appliance 代理程序	· 162（用于接收 SNMP 事件） · 5554（用于连接到 MQ 设备）	是	是	是
MongoDB 代理程序	· 27017（用于单一实例） · 27019（用于集群）	是	是	否
MySQL 代理程序	3306（用于 JDBC 连接）	是	是	是
NetApp Storage 代理程序	对于远程监视: · 8088 · 8488 · 443 · 8443	否	否	是

表 6. 代理程序和数据收集器使用的缺省端口。 *(续)*

代理程序和数据收集器	缺省端口	可配置	本地	远程
Node.js 代理程序	63336	是	是	否
Node.js 数据收集器	不适用	不适用	否	否
OpenStack 代理程序	5000（用于连接 OpenStack 身份服务）	是	否	是
Oracle Database 代理程序	1521（用于 SQL 连接）	是	是	否
PHP 代理程序	· Apache 连接 · 端口号基于 Apache 配置	是	是	否
PostgreSQL 代理程序	5432（用于 JDBC 连接）	是	是	是
Python 数据收集器	不适用	不适用	否	否
RabbitMQ 代理程序	启用 RabbitMQ 管理插件的端口号（本地和远程）：15672	是	是	是
Response Time Monitoring Agent	· 程序包分析器模型监视端口 80 上的 HTTP 事务。 · HTTP Server 模型监视所有端口。	是	是	否
Ruby 代理程序	动态生成	不适用	是	否
Ruby 数据收集器	不适用	不适用	否	否
SAP 代理程序	33*nn*（其中，*nn* 是 SAP 实例编号）	否	是	否
SAP HANA Database 代理程序	缺省值：30013。范围：30013-39913。	是	是	否
SAP NetWeaver Java Stack 代理程序	缺省值：50004。范围：50004-59904。	是	是	否
Siebel 代理程序	不适用	不适用	是	否
Skype for Business Server 代理程序（旧称 Microsoft Lync Server 代理程序）	· 业务服务器缺省端口 5061 · SQL Server 端口 1433（本地或远程，取决于环境）。	否	是	否
Sterling Connect Direct 代理程序	1363	是	否	是
Sterling File Gateway 代理程序	50000 IBM B2B Integrator REST API 端口号和数据库端口号都是必需的并且都可配置。	是	是	是
Sybase 代理程序	5000	不适用	是	否
Synthetic Playback 代理程序	· 4444（用于连接内部 Selenium 服务器） · 在受监视的 Web 站点的 HTTP URL 中指定远程端口，通常为 HTTP 80 和 HTTPS 443	否	是	否

表 6. 代理程序和数据收集器使用的缺省端口。 *(续)*

代理程序和数据收集器	缺省端口	可配置	本地	远程
Tomcat 代理程序	8686（用于 Tomcat MBean Server）	是（通过 JMX 端口）	是	否
UNIX OS 代理程序	22（用于通过 SSH 进行远程日志监视）	是	是	否
VMware VI 代理程序	· 443（用于远程监视） · 80（用于本地监视）	否	是	是
WebLogic 代理程序	7003（JMX 管理 HTTP 流量）	是	是	否
WebSphere Applications 代理程序	· 63335（用于 V8 监视代理程序） · 63336（用于 V6 监视代理程序） · 63355（用于资源监视） · 5457（用于事务框架扩展）	是	是	否
WebSphere Infrastructure Manager 代理程序	不适用	不适用	是	否
WebSphere MQ 代理程序	端口号取决于监视使用的侦听器设置。	不适用	否	是
Windows OS 代理程序	22（用于通过 SSH 进行远程日志监视）	是	是	否

方案

根据您的环境的复杂程度，必须安装不同的代理程序来监视不同的组件。使用这些部署方案可以帮助您了解必须安装哪些代理程序，以及怎样从 IBM Cloud Application Performance Management 获取最佳结果。

场景：监视 IBM API Connect

您可以通过使用 APM 代理程序和数据收集器对 IBM API Connect 环境进行监视和故障诊断。

Cloud APM 产品可帮助您管理 API Connect 环境的性能和可用性。通过使用 Cloud APM 代理程序和数据收集器，为您提供了 API Connect 基础结构和应用程序 API 的可视性和控制，从而确保资源的最佳性能和高效使用。当您在 API Connect 环境中遇到性能问题时，Cloud APM 产品可帮助您对其进行检测、诊断和隔离。

例如，可以在所有适用的系统上安装操作系统代理程序。使用操作系统代理程序收集并分析特定于服务器的性能，包括 CPU 性能、磁盘 I/O 和利用率、进程可用性和性能以及网络性能。另外，可以配置操作系统代理程序来监视密钥 API Connect 日志和系统日志。

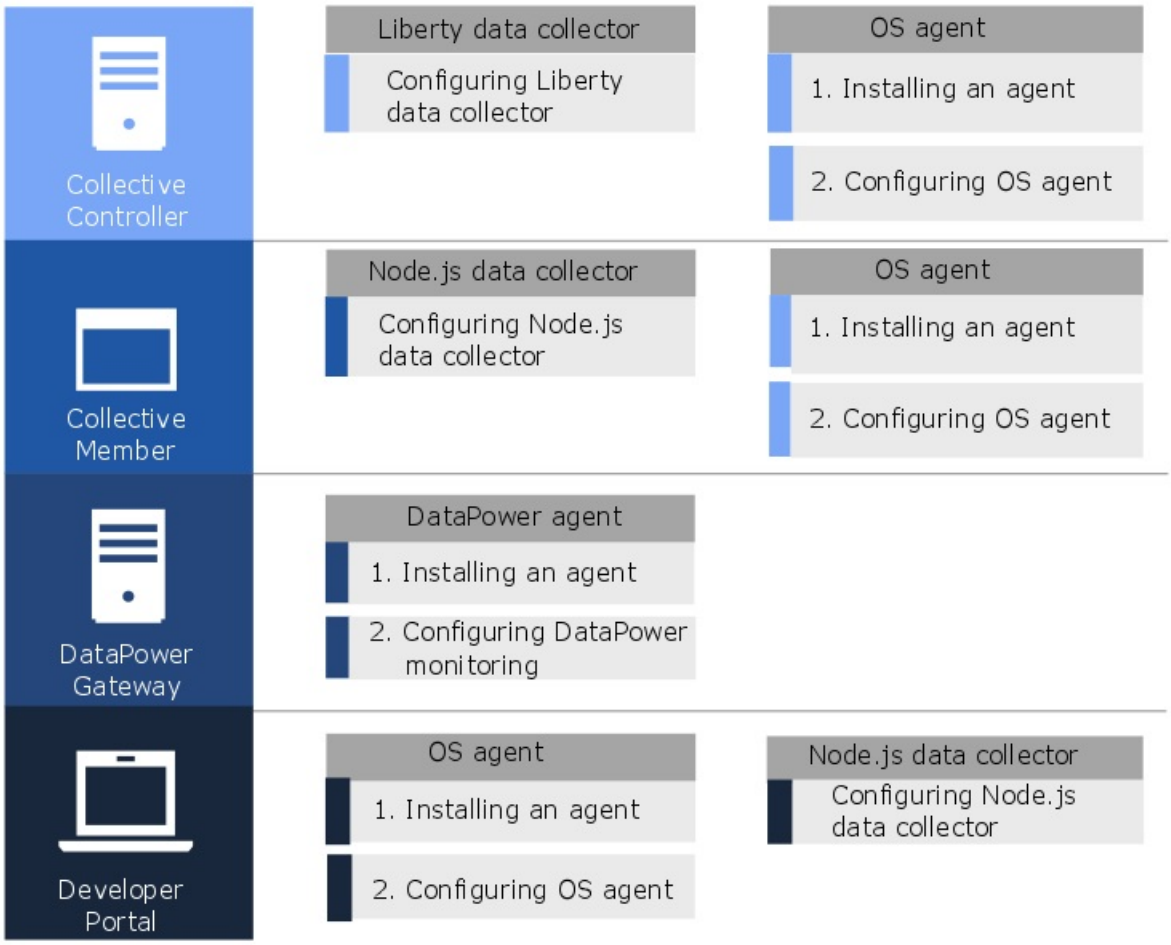
如果您有已部署的其他中间件产品，那么作为 Cloud APM 服务器的一部分安装的事物跟踪功能部件可以向您提供拓扑视图，以查看其显示的中间件产品和服务的事务跟踪信息，并在发生问题时进行故障诊断。

下图显示 API Connect 组件以及可以对其进行监视的相应的 Cloud APM 代理程序和数据收集器。要启用这些代理程序和数据收集器，完成在代理程序和数据收集器名称下列出的安装和配置任务。在包含任务名称的图中单击矩形框以转至安装或配置任务。

注:

· 将 Node.js 数据收集器安装到集合成员上每个发布的 IBM API Connect 应用程序。

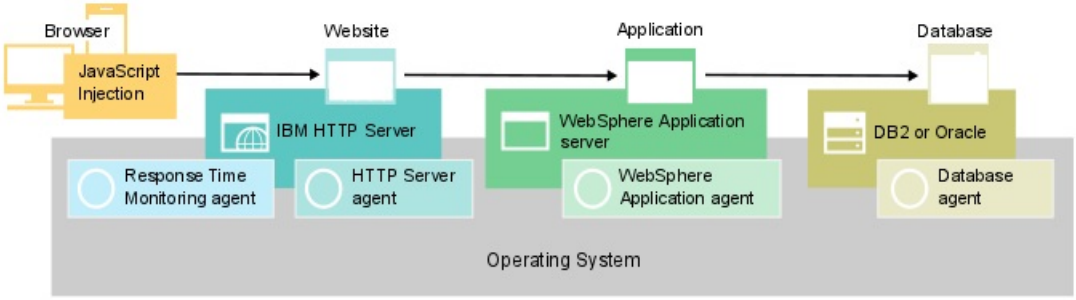
· 当监视 DataPower 网关时，DataPower 代理程序以远程方式从 DataPower 设备运行。



1. 第 749 页的『为本地应用程序配置 Liberty 数据收集器』
2. 第 101 页的『第 6 章 安装代理程序』
3. 第 538 页的『配置操作系统监视』
4. 第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』
5. 第 101 页的『第 6 章 安装代理程序』
6. 第 538 页的『配置操作系统监视』
7. 第 101 页的『第 6 章 安装代理程序』
8. 第 199 页的『配置 DataPower 监视』
9. 第 101 页的『第 6 章 安装代理程序』
10. 第 538 页的『配置操作系统监视』
11. 第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』

场景：监视 IBM Java 应用程序堆栈

您可以对 IBM Java 应用程序堆栈进行监视和故障诊断，以查看从浏览器到数据库的事务监视信息，包括来自各个组件的资源监视。IBM Java 应用程序堆栈包含 IBM HTTP Server、WebSphere Application Server 和 IBM Db2 或 Oracle 数据库。



监视 IBM Java 应用程序堆栈

要监视 IBM Java 应用程序堆栈，请按给定顺序为每个组件安装所列代理程序。

（可选）如果您还想监视系统，请在所有组件上安装操作系统代理程序。

对于 Web 服务器，请完成以下步骤：

1. 安装 HTTP Server 代理程序。

   捷径： 此过程还会安装 IBM HTTP Server 响应时间模块并自动配置 JavaScript 注入。
2. 配置 HTTP Server 代理程序安装。
3. 安装 Response Time Monitoring 代理程序。

对于应用程序服务器，安装 WebSphere Applications 代理程序。

对于数据库，根据您的数据库安装 Oracle 数据库代理程序或 Db2 代理程序。

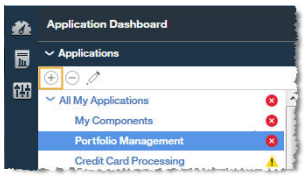
向应用程序性能仪表板添加 Web 应用程序

将您要监视的 Web 应用程序添加到应用程序性能仪表板。

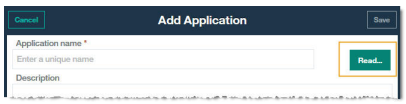
过程

要添加 Web 应用程序，请完成以下步骤：

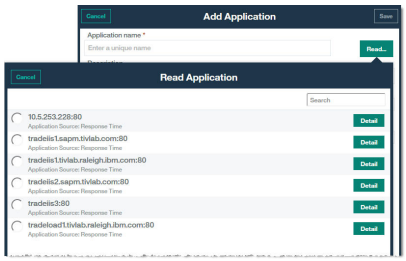
1. 在应用程序性能仪表板中，单击**添加应用程序**。



2. 单击**读取**以打开已发现的应用程序列表。



3. 选择要监视的 Web 应用程序。



4. 单击**保存**。

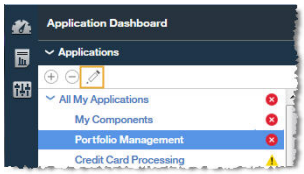
将 IBM Java 应用程序堆栈与 Web 应用程序关联

编辑 Web 应用程序以将其与要监视的 WebSphere Application Server 和数据库组件关联。

过程

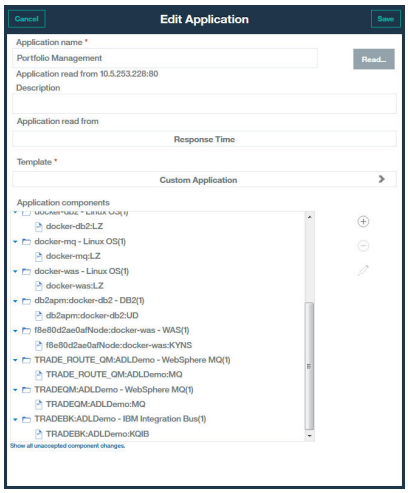
要在 Java 应用程序堆栈中显示组件，请在应用程序性能仪表板中完成以下步骤：

1. 选择 Web 服务器并单击**编辑应用程序**。



2. 在“**编辑应用程序**”窗口中，单击**添加组件** +.
3. 在“**选择组件**”窗口中，选择 WebSphere Application Server。
4. 在“**组件编辑器**”中，选择所需组件实例并单击**添加**。

   检测到的任何 WebSphere Application Server 实例都会自动添加到此列表中。
5. 单击**返回**并针对数据库重复步骤 第 76 页的『3』 - 第 76 页的『4』。继续添加 WebSphere Application Server 和数据库实例，直到完成 Java 应用程序堆栈为止。



6. 单击**关闭**，然后单击**保存**以返回到应用程序性能仪表板。

结果

提示: 如果“汇总事务拓扑”初始未显示您所期望的拓扑，请等待其刷新，几分钟后再查看该拓扑。如果该拓扑仍然不是您期望的拓扑，那么应用程序可能未与期望的组件进行通信。请检查您的环境。

查看 IBM Java 应用程序堆栈监视的结果

您可以查看拓扑中 IBM Java 应用程序堆栈监视的结果。

关于此任务

在拓扑中，您将看到从浏览器到数据库的事务监视信息，包括来自各个组件的资源监视。以下节点显示在“汇总事务拓扑”和“事务实例拓扑”中：

- 仅在启用 JavaScript 注入时显示的浏览器
- HTTP Server
- WebSphere Application Server
- 数据库

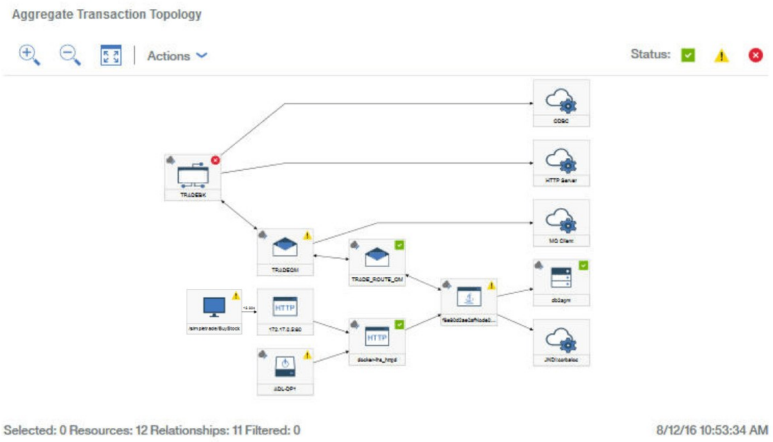
过程

您可以对拓扑中的节点添加链接以提供有关该节点的更多详细信息：

1. 将鼠标悬停于节点上会显示一个窗口，其中包含附加信息。
2. 要向下钻取到该节点的更详细仪表板，请右键单击该节点，然后选择链接。

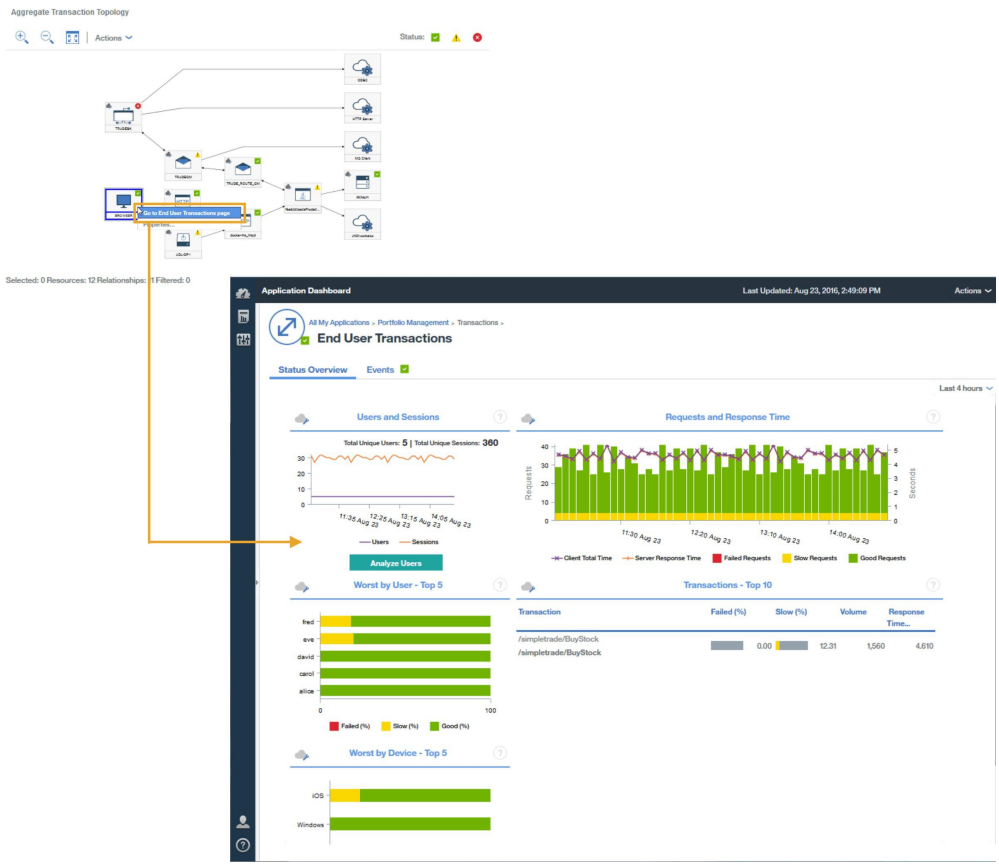
汇总事务拓扑

“汇总事务拓扑”显示在应用程序摘要仪表板中。



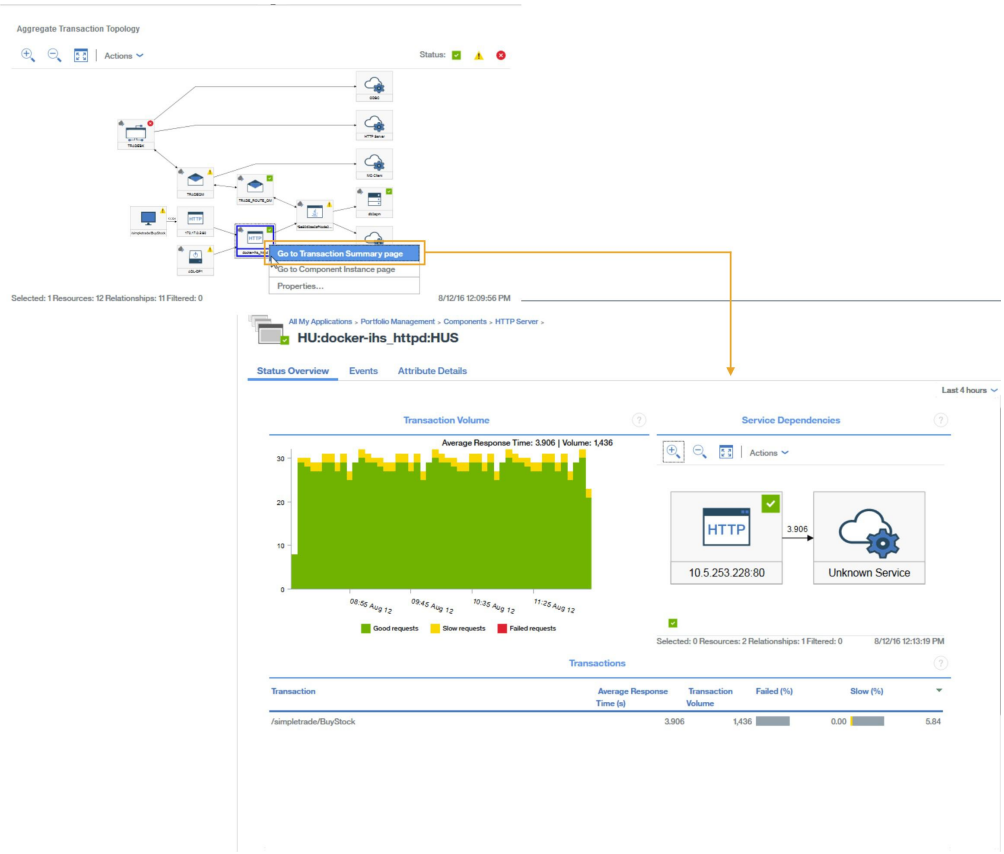
“汇总事务拓扑”显示以下信息：

· 基于浏览器的客户机的节点，向下钻取到最终用户体验



切记：此节点仅在自动 JavaScript 注入测量来自浏览器的数据时显示。

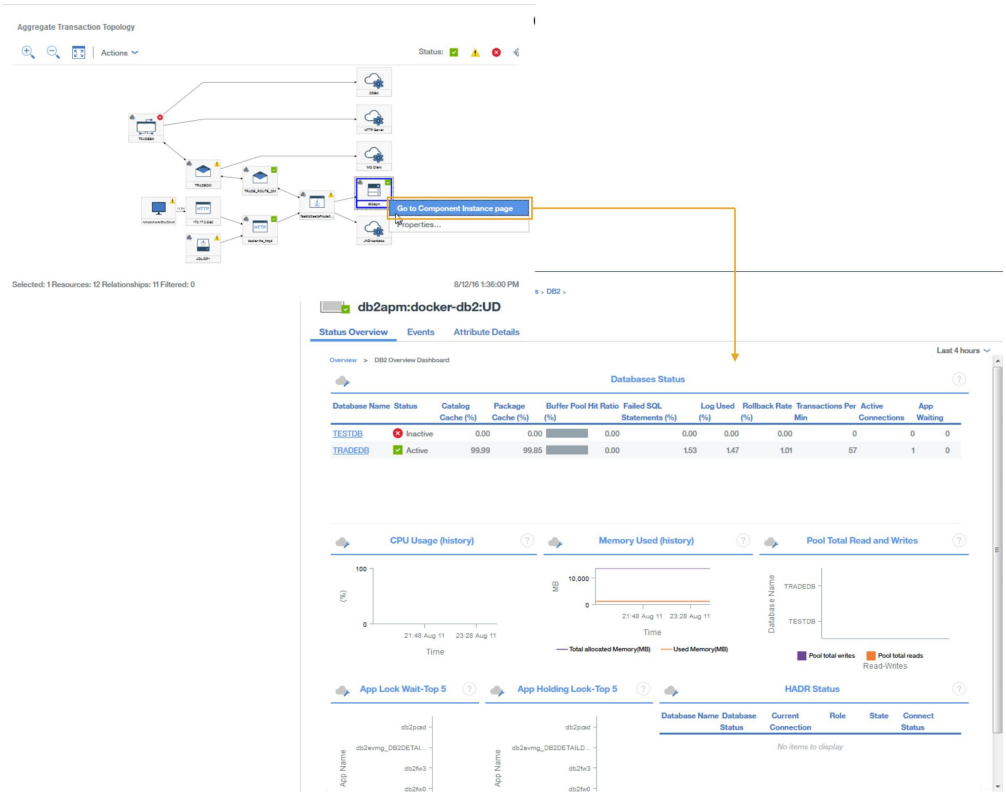
· 基于 HTTP 的应用程序的节点，向下钻取到 Web 服务器资源页面或事务摘要页面



· 基于 WebSphere Application Server 的应用程序的节点，向下钻取到应用程序资源页面或事务摘要页面



· 特定数据库服务器的节点，向下钻取到数据库资源页面（如果可用）

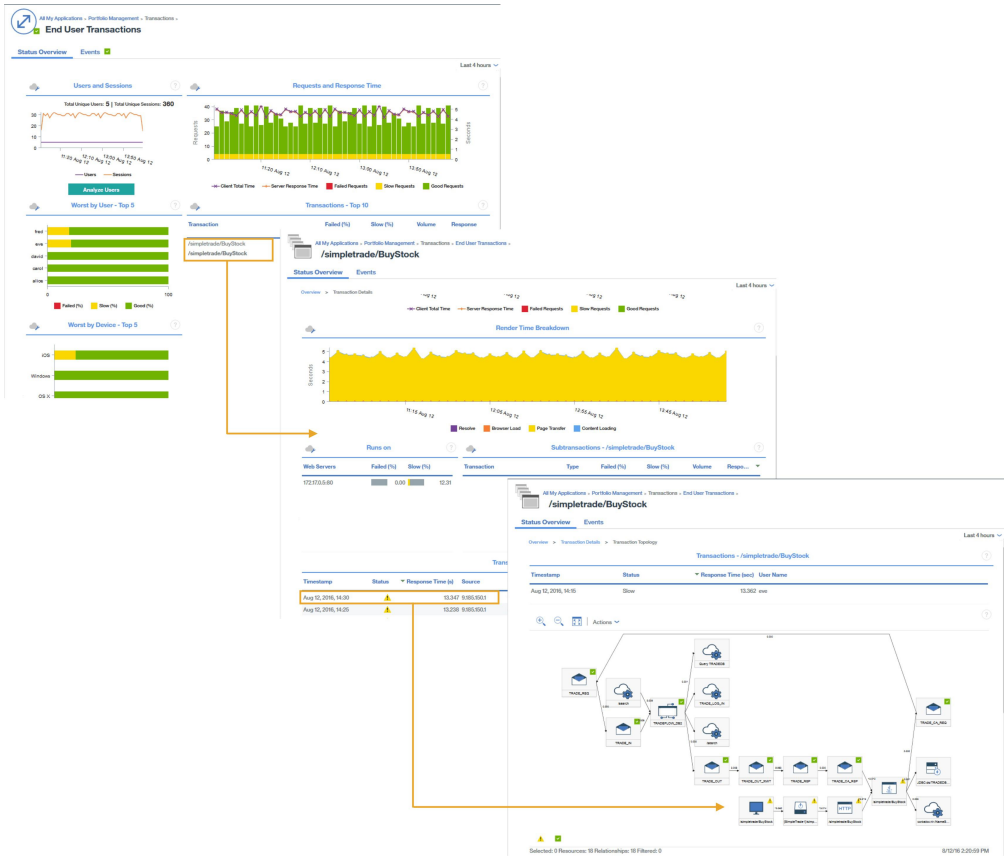


事务实例拓扑

事务实例拓扑针对实际的最终用户事务显示。

从“**最终用户事务**”摘要沿着以下窗口小部件向下钻取：

1. 在“事务 - 前 10 个”表中选择事务
2. 在“事务实例”表中选择实例



Java 应用程序堆栈的事务实例拓扑显示以下节点。单击节点可显示有关该节点的信息。

- 基于浏览器的客户机的节点

  切记: 此节点仅在自动 JavaScript 注入测量来自浏览器的数据时显示。
- HTTP 节点（包含来自浏览器的响应时间）
- DataPower 节点（如果受检测）
- WebSphere Application Server 节点，可以从该节点向下钻取到应用程序资源页面
- 数据库服务器节点，可以从该节点向下钻取到数据库资源状态以及有关 JDBC 请求的 SQL 语句诊断信息

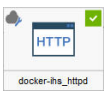
  提示: 如果该拓扑指示大部分响应时间都耗费在数据库中，那么在单击**诊断**时会直接打开 SQL 语句信息。

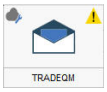
此外还显示甘特图，其中会概括实例计时。

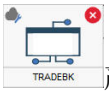
诊断环境中的问题

如果环境中一个组件的事务实例变慢或发生故障，那么会为受影响的组件分配相应状态。

节点可能具有下列其中一种状态：

- 

正常，节点的右上角显示内含钩号的绿色正方形
- 

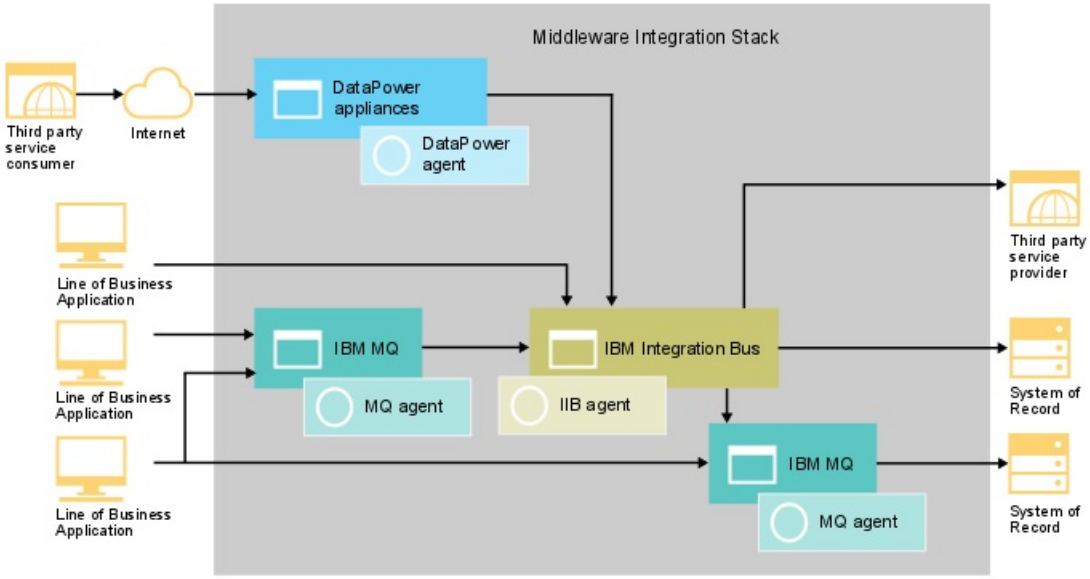
警告，节点的右上角显示内含惊叹号的黄色三角形
- 

严重：节点背景色为红色，右上角显示内含 X 符号的红色圆圈

要确定具有警告或严重状态的这些组件的问题的原因，请右键单击节点并向下钻取，以了解有关可能导致失败的原因的更多信息。

场景：监视 IBM 集成堆栈

您可以监视 IBM 集成堆栈以查看其显示的中间件产品和服务的事务跟踪信息，并在发生问题时进行故障诊断。IBM 集成堆栈 包含 IBM MQ、IBM Integration Bus 和 DataPower 设备。



监视 IBM 集成堆栈

要监视 IBM 集成堆栈，请按给定顺序为每个组件安装所列代理程序。

（可选）如果您还想监视系统，请在此系统上安装操作系统代理程序。

对于 IBM MQ，请完成以下步骤：

1. 安装 Monitoring Agent for WebSphere MQ。
2. 配置 WebSphere MQ 代理程序 以连接到队列管理器。
3. 在队列管理器中启用 **MQ 应用程序活动跟踪**。

对于 IBM Integration Bus，请完成以下步骤：

1. 安装 Monitoring Agent for IBM Integration Bus。
2. 启用 IBM Integration Bus 以便进行事务跟踪。
3. 为所需 IBM Integration Bus 代理程序 实例配置事务跟踪。

对于 DataPower 设备，请完成以下步骤：

1. 安装 Monitoring Agent for DataPower。
2. 配置 DataPower 代理程序 以连接到 DataPower 设备。
3. 确保针对必需的 DataPower 代理程序 实例启用事务跟踪。
4. 设置 DataPower 设备。

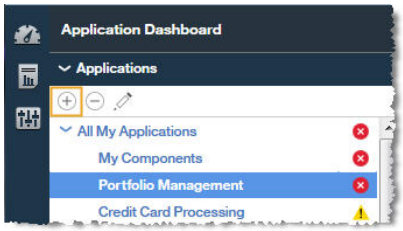
将中间件应用程序添加到应用程序性能仪表板

创建 IBM 集成堆栈 应用程序并添加想要监视的 IBM MQ、IBM Integration Bus 和 DataPower 设备 实例。

过程

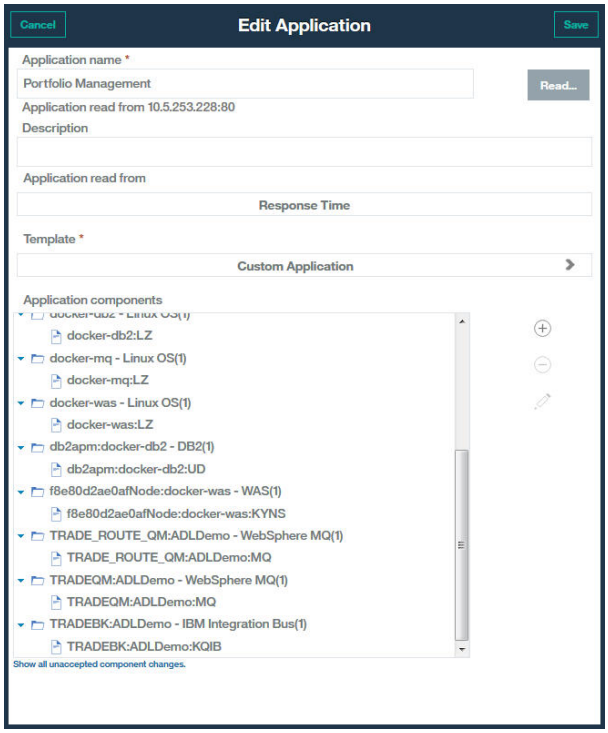
要在 IBM 集成堆栈中显示组件，请在应用程序性能仪表板中完成以下步骤：

1. 在应用程序性能仪表板中，单击**添加应用程序**。



2. 在“**编辑应用程序**”窗口中，添加应用程序名称，然后单击**添加组件** +.
3. 在“**选择组件**”窗口中，选择 **IBM Integration Bus**。
4. 在“**组件编辑器**”中，选择所需组件实例并单击**添加**。

   检测到的任何 IBM Integration Bus 实例都会自动添加到此列表中。
5. 单击**后退**，并针对 **WebSphere MQ** 和 **DataPower Appliance** 重复步骤 3 - 4。继续添加 IBM Integration Bus、IBM MQ 和 DataPower 设备 实例，直至 IBM 集成堆栈完成。



6. 单击**关闭**，然后单击**保存**以返回到应用程序性能仪表板。

结果

提示: 如果“汇总事务拓扑”初始未显示您所期望的拓扑，请等待其刷新，几分钟后再查看该拓扑。如果该拓扑仍不是您所期望的拓扑，那么表示可能存在问题，应用程序未与所期望组件通信。请检查您的环境。

查看 IBM 集成堆栈监视的结果

您可以在拓扑和中间件页面中查看 IBM 集成堆栈监视的结果。您还可以在事务违反所定义的阈值时查看生成的事件。

关于此任务

在拓扑中，您可以查看中间件组件之间的交互。以下中间件节点显示在“汇总事务拓扑”和“事务实例拓扑”中：

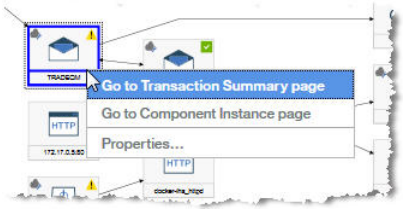
- IBM Integration Bus
- IBM MQ
- DataPower 设备

将鼠标指针悬停在节点上以显示“属性”窗口，其中显示了用于说明节点为何处于特定状态的信息。此状态由情境决定；将显示具有错误状态的情境。

过程

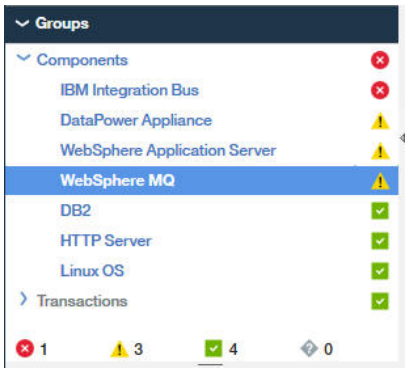
您可以对拓扑中的节点添加链接以提供有关该节点的更多详细信息：

1. 右键单击节点。



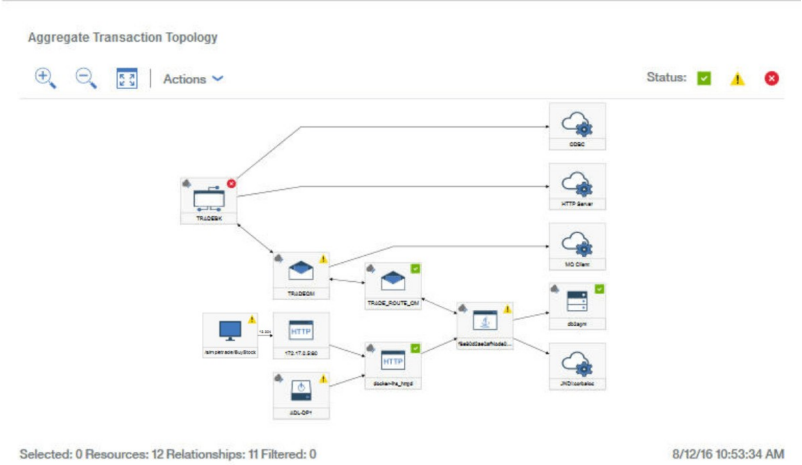
2. 选择**转至组件页面**以显示有关组件的信息。
3. 选择**转至事务摘要页面**以显示有关中间件事务的信息。

   提示: 选择导航器中的**组 > 组件 > 中间件组件**，然后选择卷窗口小部件中的请求时间段以转至相同的仪表板。



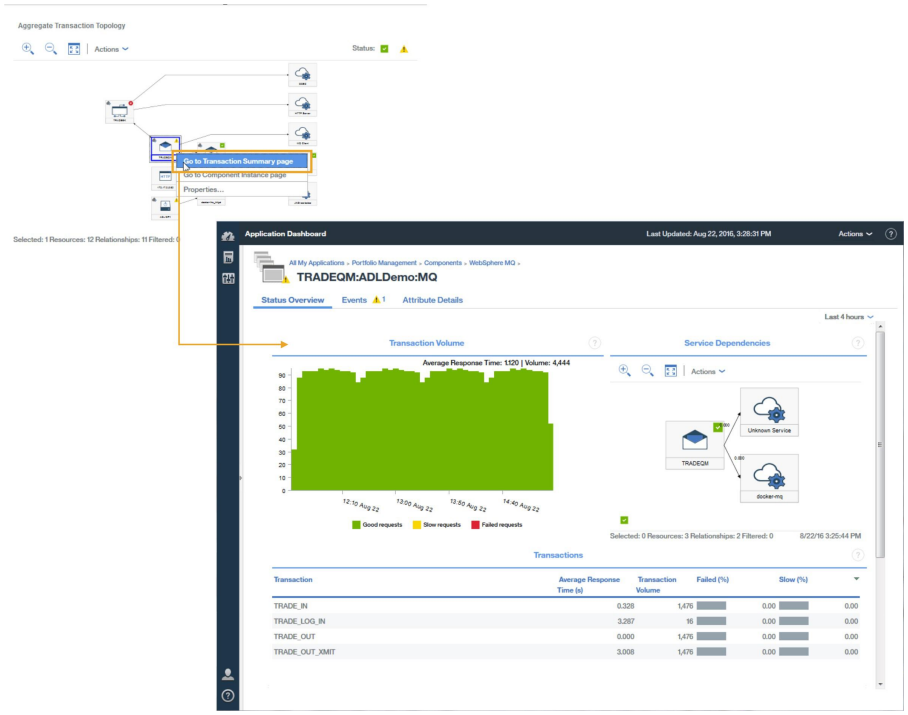
汇总事务拓扑

“汇总事务拓扑”显示在应用程序摘要仪表板中。



聚集事务拓扑可以显示 IBM MQ、IBM Integration Bus 和 DataPower 设备节点。从这些节点向下钻取到有关中间件集成堆栈的更多信息。

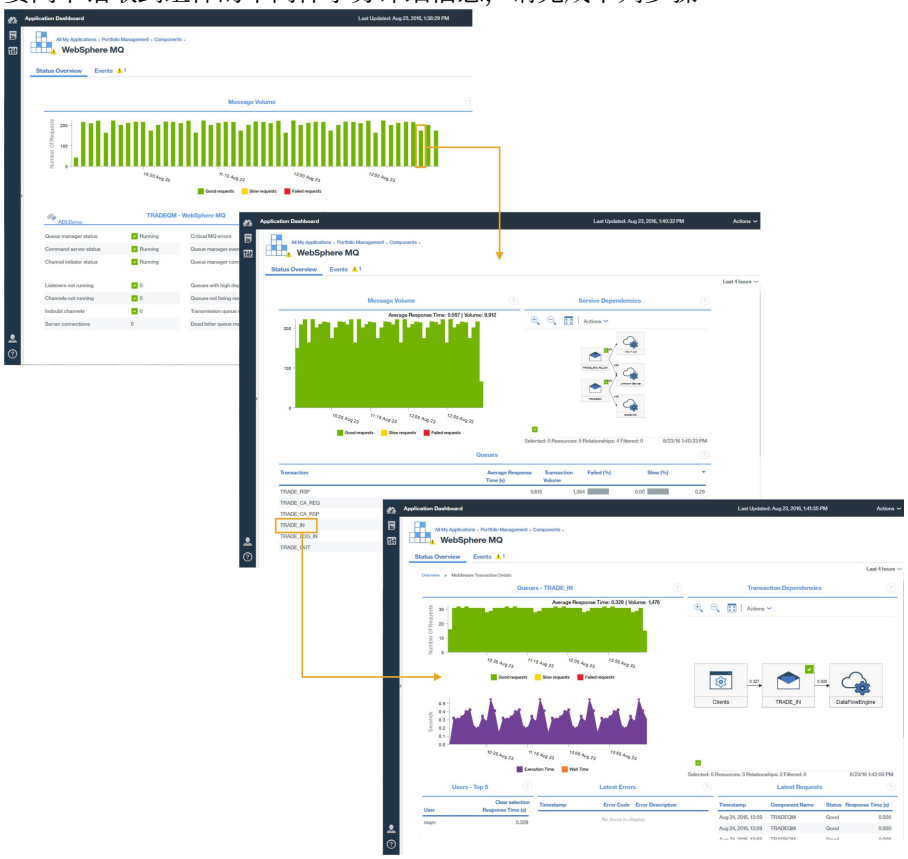
要向下钻取，请右键单击“汇总事务拓扑”中的中间件节点，然后选择**转至事务摘要页面**。或者，选择导航器中的**组 > 组件 > 中间件组件**，并选择卷窗口小部件中的请求时间段以访问相同信息。



中间件事务详细信息

从中间件“事务摘要”页面，您可以向下钻取至中间件事务详细信息。

要向下钻取到组件的中间件事务详细信息，请完成下列步骤：



1. 在组件的“中间件事务摘要”页面中，选择“**消息或卷**”图表中的监视时间间隔。
2. 在“**中间件事务摘要**”上的**队列**、**代理**或**设备**窗口小部件中，选择队列、代理或设备。

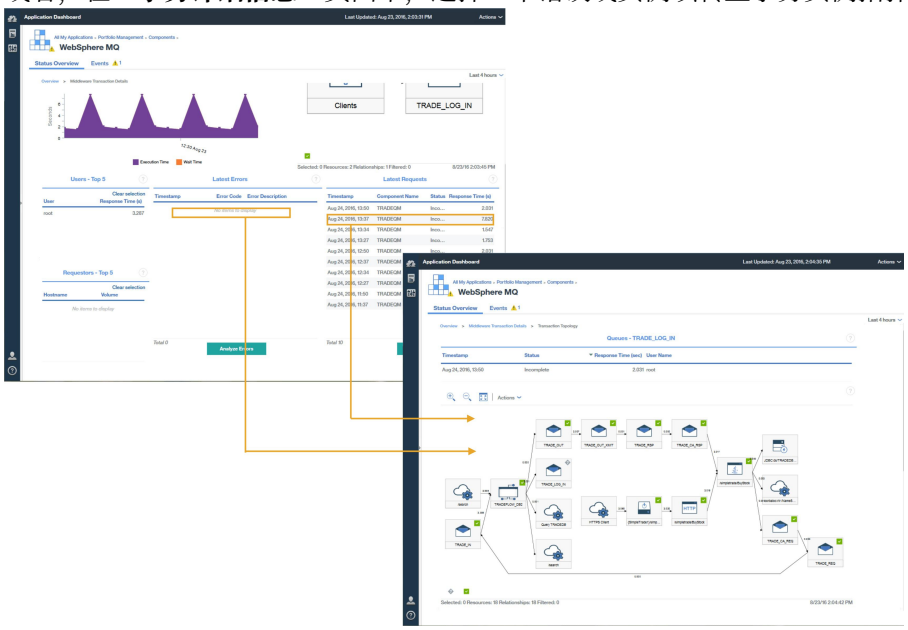
分析错误和实例

从“中间件事务详细信息”页面中，您可以进一步深入了解可帮助您分析错误和实例以及访问事务实例拓扑的信息。

要向下钻取至中间件组件的错误和实例，然后钻取至事务实例拓扑，请在“**事务详细信息**”页面上，完成以下某个步骤：

- 单击**分析错误**以显示“**错误分析**”页面，然后选择错误。
- 单击**分析请求**以显示“**实例分析**”页面，然后选择实例。

或者，在“**事务详细信息**”页面中，选择一个错误或实例以转至事务实例拓扑。

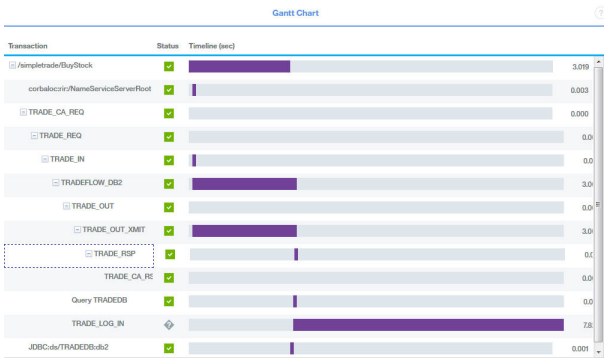


“事务实例拓扑”显示以下中间件节点：

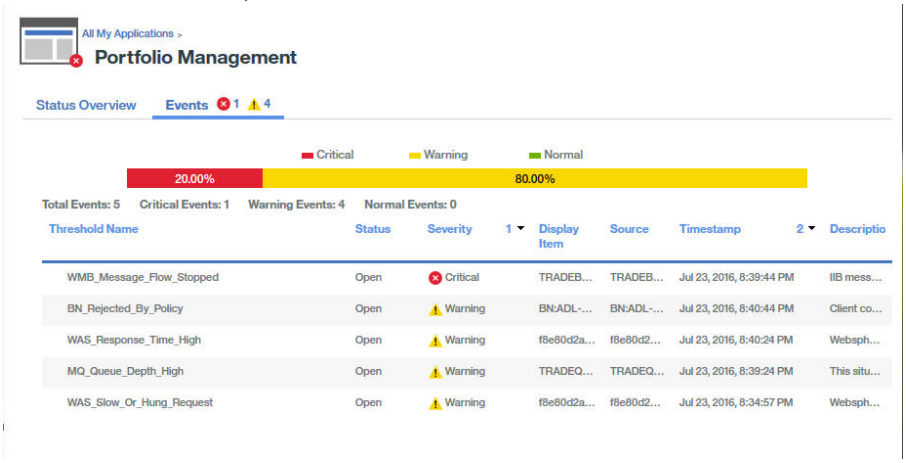
- 消息队列管理器
- IBM Integration Bus 代理
- DataPower Appliances

选择一个节点以显示其属性，这些属性用于说明为何节点处于特定状态。

将针对选中的队列或代理显示“事务甘特图”。甘特图帮助您找到对于事务的整体响应时间最重要的贡献因子。



事件
除其他代理程序以外，按照缺省事务跟踪阈值为 IBM 集成堆栈生成事件。



有关事务跟踪缺省事件的更多信息，请参阅第 860 页的『事务监视的事件阈值』。

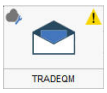
可以添加阈值来创建更多事件，例如，针对事务速率变慢或低于特定阈值的情况。

有关添加事件的更多信息，请参阅第 862 页的『创建阈值以针对事务监视生成事件』。

诊断环境中的问题
如果环境中一个组件的事务实例变慢或发生故障，那么会为受影响的组件分配相应状态。

节点可能具有下列其中一种状态：

- 

正常，节点的右上角显示内含钩号的绿色正方形
- 

警告，节点的右上角显示内含惊叹号的黄色三角形
- 

严重：节点背景色为红色，右上角显示内含 X 符号的红色圆圈

要确定具有警告或严重状态的这些组件的问题的原因，请右键单击节点并向下钻取，以了解有关可能导致失败的原因的更多信息。

下载代理程序 以及数据收集器

您可从 IBM Marketplace Web 站点访问您的 Cloud APM 预订。登录到您的帐户并下载安装归档文件。安装归档文件包含代理程序 和数据收集器的安装和配置文件。

您可以通过完成以下教程中的步骤，了解有关下载代理程序和数据收集器的更多信息：

- 第 87 页的『教程：下载和安装代理程序』
- 第 91 页的『教程：下载和配置数据收集器』

您可以从 IBM Marketplace 注册申请以下某个 IBM Cloud Application Performance Management 产品的有效试用版或已购买预订。

IBM Marketplace

通过 IBM Marketplace > Cloud APM > 免费试用可注册申请试用。通过 IBM Marketplace > Cloud APM > 购买 购买预订。登录 产品和服务 页面以下载代理程序和数据收集器的安装和配置文件。

“**产品和服务**”页面适用于活动订户。如果您有任何问题，请转至 Marketplace 支持。

测试连接

有关检查与 Cloud APM 服务器（用于下载软件包）的连接的信息，请参阅网络连接。

教程：下载和安装代理程序

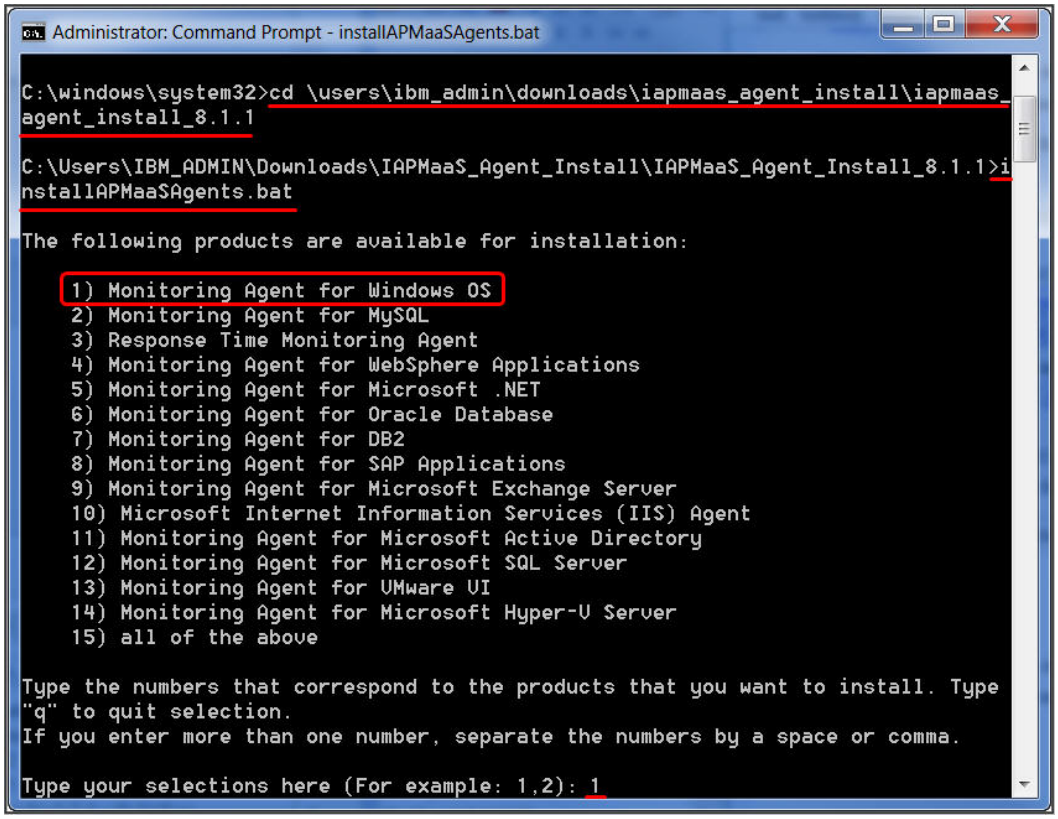
使用本教程可获取从 IBM Marketplace 下载和安装 Cloud APM Windows OS 代理程序的实际经验。然后，您可以启动 Cloud APM 控制台，通过在仪表板中查看关键绩效指标 (KPI) 来检查受监视资源的运行状况。

关于此任务

本教程说明如何从 IBM Marketplace 上的“**产品和服务**”页面下载 Windows 安装包，解压缩安装文件，以及安装 Windows OS 代理程序。然后，您返回到“**产品和服务**”以启动 Cloud APM 控制台，并打开应用程序性能仪表板以检查 Windows 系统的运行状况。

过程

1. 如果您未登录到 IBM Marketplace，请使用您的 IBMid 和密码登录并转至“**产品和服务**”。
   “**产品和服务**”页面适用于活动订户。如果您有任何问题，请转至 Cloud Application Performance Management 论坛或 Marketplace 支持。
2. 下载 Windows 安装归档文件：
   a) 在 Cloud APM 预订框中，单击**管理** > **下载**。
   b) 选择 Windows 操作系统。
      选择 IBM Cloud Application Performance Management Agents 64 位程序包。如果您使用的版本是 Windows 32 位，请选择 32 位代理程序包。
   c) 单击**下载**并将代理程序安装归档文件保存到系统。
   示例：
   ```
   C:\Users\MY_ADMIN\Downloads\IAPM_Agent_Install.zip
   ```
3. 在本地 Windows 系统上，浏览至已下载的归档文件所保存和解压缩的目录。
   例如，在 Windows 资源管理器中，打开“**下载**”目录，右键单击 `IAPM_Agent_Install.zip`，然后选择**全部解压缩**。
4. 以管理员身份打开命令提示符：
   a) 在 Windows **开始**菜单的搜索框中，输入 `command`。
   b) 右键单击所显示列表中的**命令提示符**并选择**以管理员身份运行**。
5. 切换到解压安装文件的目录。
   示例：
   ```
   cd C:\Users\MY_ADMIN\Downloads\IAPM_Agent_Install\IAPM_Agent_Install_8.1.4
   ```
6. 运行安装脚本以安装 Windows OS 代理程序：
   a) 输入以下命令：
   ```
   installAPMAgents.bat
   ```
   b) 从可用代理程序列表中，输入对应于 Windows OS 代理程序 的编号。
   c) 回答提示以确认想要安装 Windows OS 代理程序，并接受许可协议：
      然后将针对您的环境启动先决条件扫描，该扫描可能需要一些时间才能完成。对于任何未满足的需求，您将收到一条消息以将您定向至日志文件，其中包含故障原因。诸如磁盘空间不足等先决条件会停止安装。您必须解决故障，然后重新启动安装脚本。如果您有任何问题，请转至 Cloud Application Performance Management 论坛，或发送电子邮件至 `info@ibmserviceengage.com`。



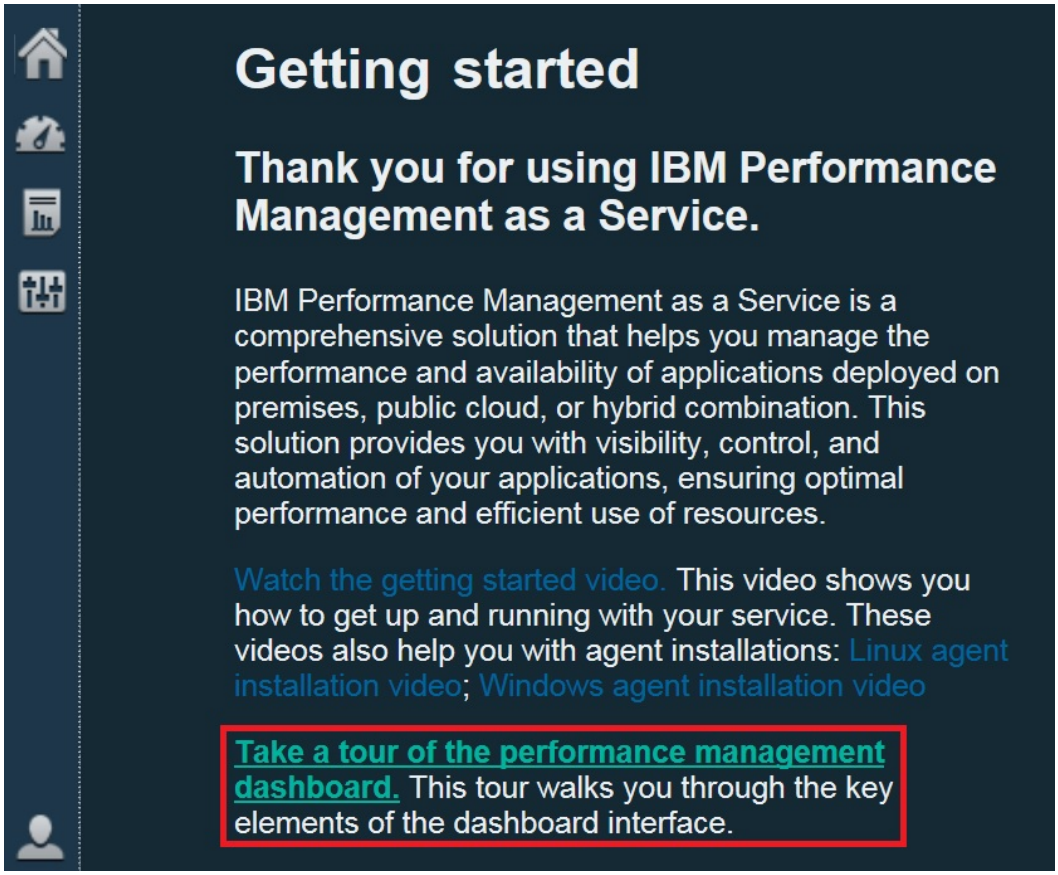
要查看 Windows 操作系统需求报告，请参阅 System requirements (APM Developer Center)。

在成功安装后，Windows OS 代理程序 将自动启动，并且您可以启动 Cloud APM 控制台以开始监视 Windows 系统。

注: 如果环境包含不允许到外部主机的透明出站 HTTPS 连接的防火墙，那么必须针对代理程序和 Cloud APM 服务器之间的通信设置转发代理。通过设置转发代理，您可以将所有流量转发到网络上的特定点，然后只允许单个连接通过防火墙。有关更多信息，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。

7. 返回到 IBM Marketplace 上的“**产品和服务**”，然后单击 Cloud APM 预订框上的**启动**。

   Cloud APM 控制台将打开至“**入门**”页面，您可以在其中了解功能、观看不同用户方案的视频以及打开关联的控制台页面。
8. 在“**入门**”页面中，单击“浏览性能管理仪表板”以快速浏览导航元素。



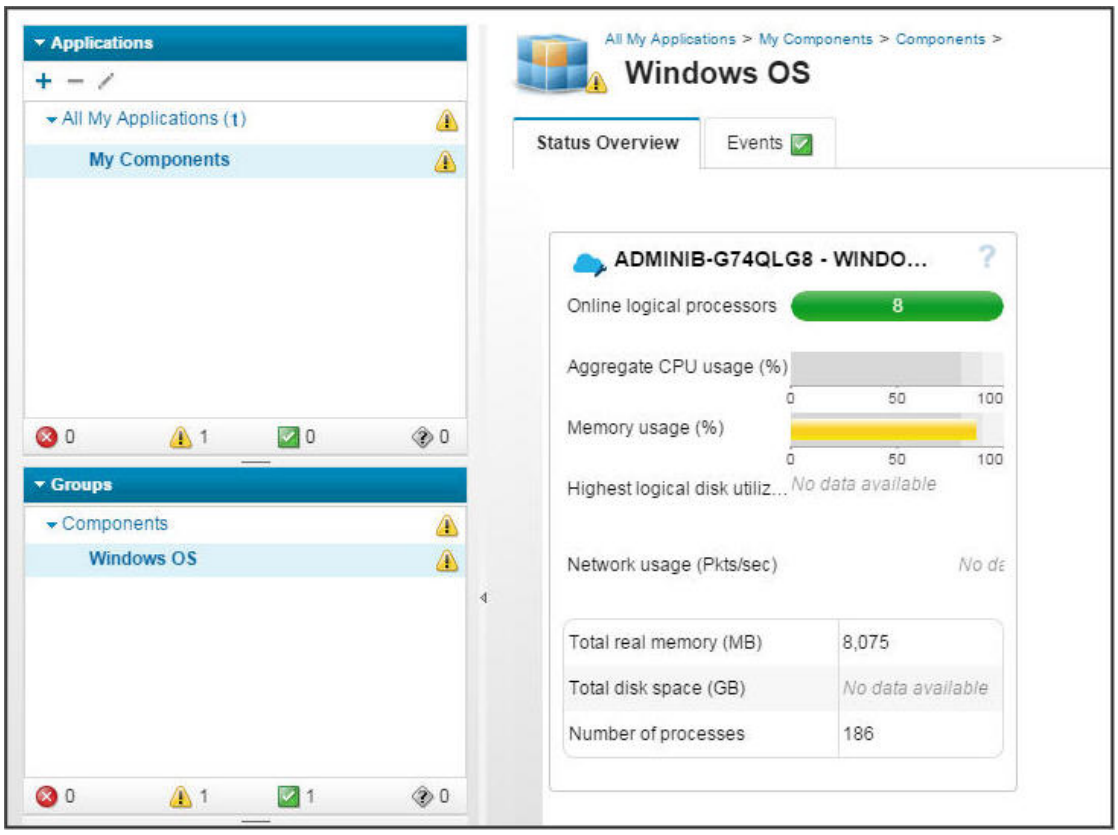
9. 打开 Windows 操作系统摘要仪表板：

a) 在导航栏中，单击 **性能** > **应用程序性能仪表板**。

- 将针对环境中每个定义的应用程序，显示带有摘要状态框的“**我的所有应用程序**”仪表板。最初，仅显示“**我的组件**”预定义应用程序。
- 如果看到“**添加应用程序**”窗口而不是“**我的组件**”，那么创建应用程序以查看您的受监视资源：
  1) 输入应用程序的名称，例如，“我的应用程序”。
  2) 单击 +。
  3) 滚动到“选择组件”列表的尾部，然后单击 **Windows 操作系统**。
  4) 在“组件编辑器”中，单击 `Primary:`*`Host_Name`*`:NT`，单击**添加**，然后单击**后退**以将代理程序添加到应用程序。
  5) 单击**保存**以关闭窗口并在仪表板中查看新应用程序的摘要状态框。

b) 在摘要框中，单击 **组件**。

- 此时将显示 Windows 操作系统受管系统的摘要仪表板。在此，您可以单击状态摘要组窗口小部件中的任意位置以向下钻取到详细仪表板，其中包含从 Windows OS 代理程序 报告的 KPI。
- 可能要等待几分钟时间，新启动的代理程序才能与监视基础结构进行通信并向控制台发送 KPI。



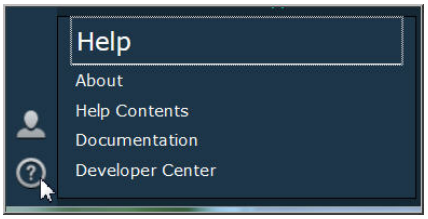
如果此代理程序没有与 Cloud APM 服务器通信或者未启动，那么摘要仪表板不会显示任何 KPI，并且状态将显示为 “未知”。您可以使用 **os-agent** 命令来检查状态并启动代理程序（如果需要）。以管理员身份 打开命令提示符，在 `C:\\IBM\APM\bin` 文件夹中输入 **os-agent status** 命令。如果未启动代理程序，请输入 **os-agent start** 命令。

结果

您已安装 Cloud APM 代理程序，观察到的监视数据将发送到 应用程序性能仪表板。

下一步做什么

- 浏览控制台：在使用 Cloud APM 控制台时，浏览功能。您可以通过单击窗口条幅中的 以了解当前仪表板。您可以在 IBM Knowledge Center 上通过导航栏中的 **帮助**菜单打开帮助系统或者 Cloud APM 主题集合。



- 安装其他代理程序：您有安装其他类型的 Windows 代理程序以监视环境 所需的所有安装文件。您还可以在环境中的其他系统上安装代理程序。如果您有 AIX 或 Linux 系统，请下载关联的安装归档文件。在安装前，某些代理程序类型具有必须完成的先决条件，并且大多数代理程序类型在安装后都需要一些配置。有关更多信息，请参阅第 103 页的『AIX 系统上的安装前任务』、第 109 页的『Linux 系统上的安装前任务』、第 116 页的『Windows 系统上的安装前任务』和第 135 页的『第 7 章 配置环境』。

教程：下载和配置数据收集器

使用本教程可获取从 IBM Marketplace 下载和配置 Cloud APM Bluemix Ruby 数据收集器的实际经验。然后，您可以启动 Cloud APM 控制台，通过在仪表板中查看关键绩效指标 (KPI) 来检查受监视资源的运行状况。

关于此任务

本教程说明如何从 IBM Marketplace 上的“**产品和服务**”页面下载用于 Bluemix 应用程序的数据收集器软件包，解压缩配置文件，以及在 Linux 系统上配置 Bluemix Ruby 数据收集器。然后，您返回到“**产品和服务**”以启动 Cloud APM 控制台，并打开应用程序性能仪表板以检查 Bluemix Ruby 应用程序的运行状况。

过程

1. 如果您未登录到 IBM Marketplace，请使用您的 IBMid 和密码登录并转至“**产品和服务**”。
   “**产品和服务**”页面适用于活动订户。如果您有任何问题，请转至 Cloud Application Performance Management 论坛或 Marketplace 支持。
2. 下载用于 Bluemix 应用程序的数据收集器软件包 `IBM_Bluemix_Data_Collectors_Install.tgz`。
3. 在本地系统上，浏览至保存所下载的归档文件的目录，并通过运行以下命令将其解压缩：

   ```
   tar -zxvf IBM_Bluemix_Data_Collectors_Install.tgz
   ```

   您将得到四个压缩文件，它们分别代表一种 Bluemix 应用程序类型的数据收集器。Bluemix Ruby 应用程序的数据收集器软件包是 `ruby_datacollector.tgz`。
4. 通过运行以下命令，将 `ruby_datacollector.tgz` 中的文件解压缩：

   ```
   tar -zxvf ruby_datacollector.tgz
   ```

   这将创建 `ibm_ruby_dc` 文件夹。
5. 通过运行以下命令，将 `ibm_ruby_dc` 中的整个 `etc` 文件夹复制到 Ruby 应用程序的根文件夹：

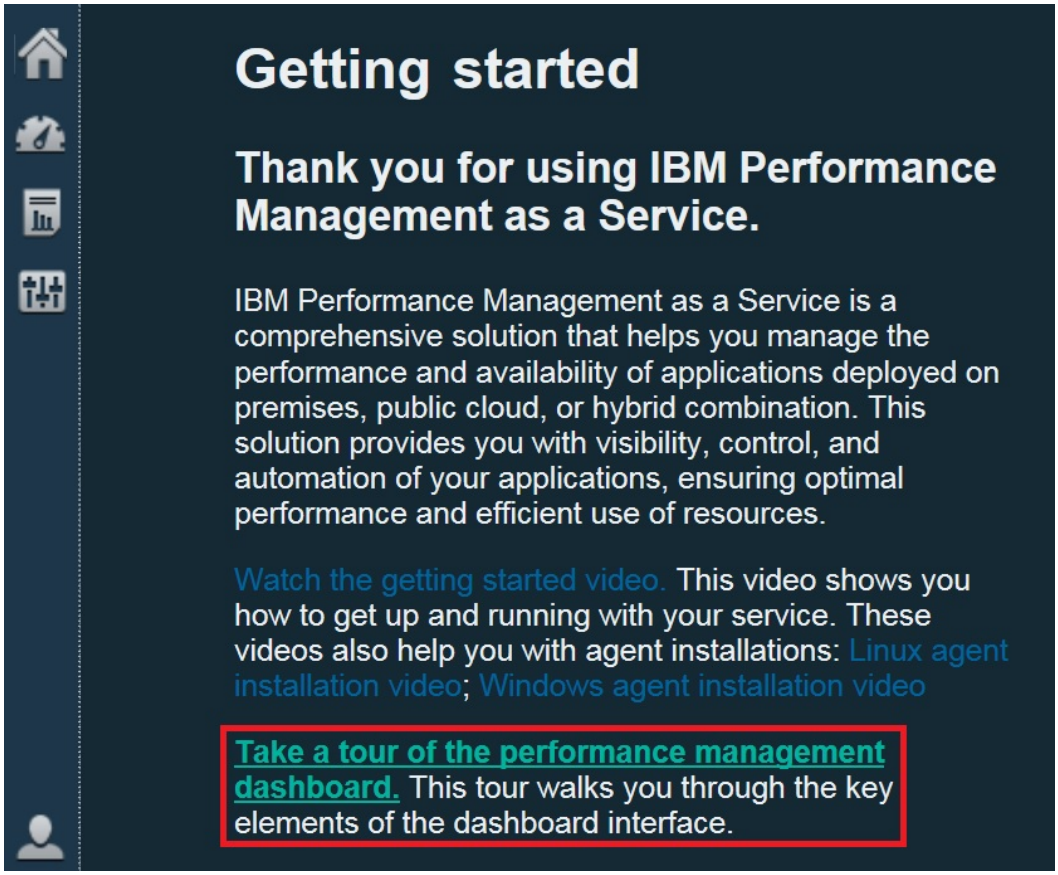
   ```
   cp -r directory to the etc folder  home directory of your Ruby application
   ```

   如果您将数据收集器解压缩到 `/opt/ibm/ccm/ibm_ruby_dc/etc` 目录，并且 Ruby 应用程序的主目录为 `/root/ruby_app/`，那么命令如下所示：

   ```
   cp -r /opt/ibm/ccm/ibm_ruby_dc/etc /root/ruby_app/
   ```
6. 在 Bluemix Ruby 应用程序的主文件夹中的 `Gemfile` 中，添加以下小节：

   ```
   gem 'logger',  '>= 1.2.8'
   source 'https://managemserver.ng.bluemix.net' do
     gem 'ibm_resource_monitor'
     gem 'stacktracer'
   end
   ```
7. 运行 `bundle lock` 命令，以重新生成 `Gemfile.lock` 文件。
8. 从 Ruby 应用程序的主目录中，运行以下命令：

   ```
   cf push
   ```
9. 返回到 IBM Marketplace 上的“**产品和服务**”，然后单击 Cloud APM 预订框上的**启动**。
   Cloud APM 控制台将打开至“**入门**”页面，您可以在其中了解功能、观看不同用户方案的视频以及打开关联的控制台页面。
10. 在“**入门**”页面中，单击“浏览性能管理仪表板”以快速浏览导航元素。



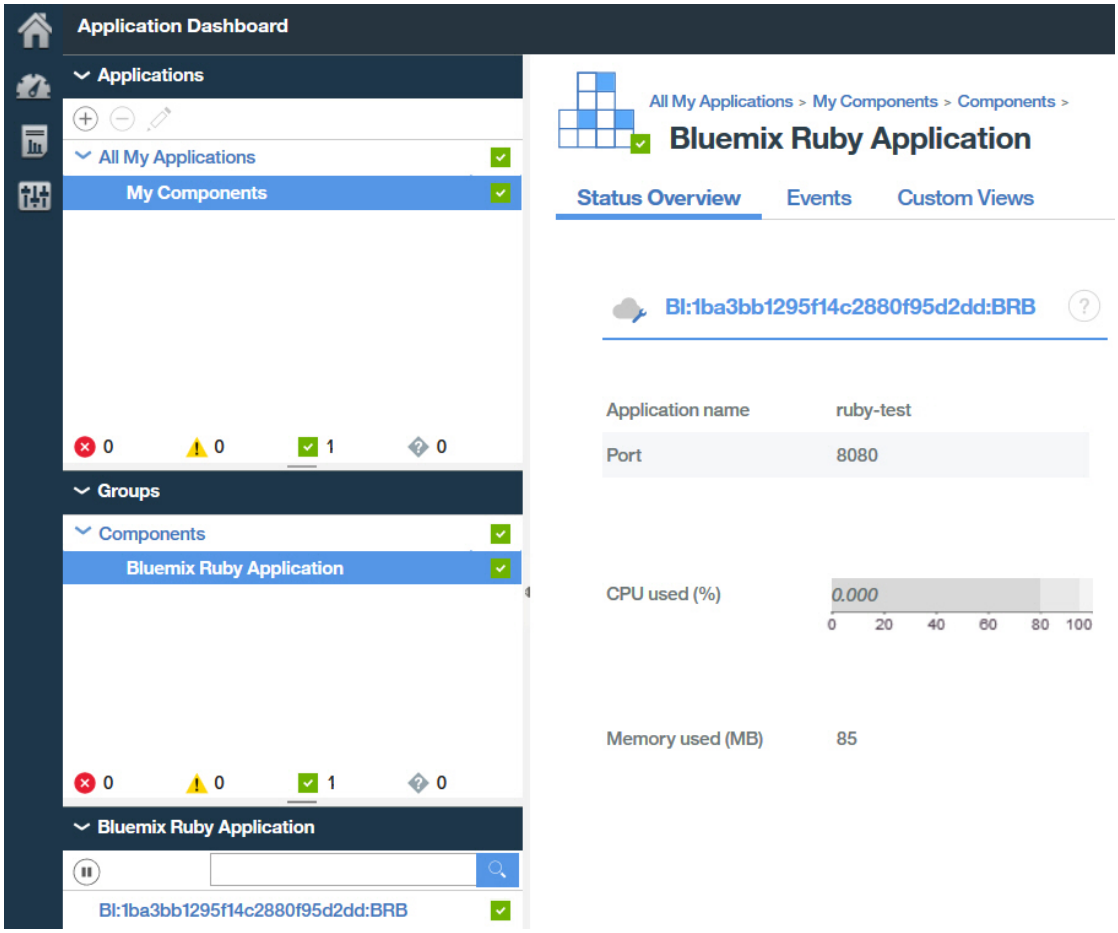
11. 打开 Bluemix Ruby 应用程序摘要仪表板：

a) 在导航栏中，单击 **性能** > **应用程序性能仪表板**。

- 将针对环境中每个定义的应用程序，显示带有摘要状态框的“**我的所有应用程序**”仪表板。最初，仅显示“**我的组件**”预定义应用程序。
- 如果看到“**添加应用程序**”窗口而不是“**我的组件**”，那么创建应用程序以查看您的受监视资源：
  1) 输入应用程序的名称，例如，“我的应用程序”。
  2) 单击 +。
  3) 单击 **Bluemix Ruby 应用程序**。
  4) 在“组件编辑器”中，选择实例，单击**添加**，然后单击**后退**以将数据收集器添加到应用程序。
  5) 单击**保存**以关闭窗口并在仪表板中查看新应用程序的摘要状态框。

b) 在摘要框中，单击 **组件**。

- 此时将显示 Bluemix Ruby 应用程序的摘要仪表板。在此，您可以单击状态摘要组窗口小部件中的任意位置以向下钻取到详细仪表板，其中包含从 Bluemix Ruby 数据收集器报告的 KPI。
- 可能要等待几分钟时间，新启动的数据收集器才能与监视基础结构进行通信并向控制台发送 KPI。

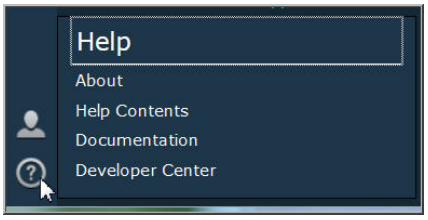


结果

您已安装 Cloud APM 数据收集器，观察到的监视数据将发送到 应用程序性能仪表板。

下一步做什么

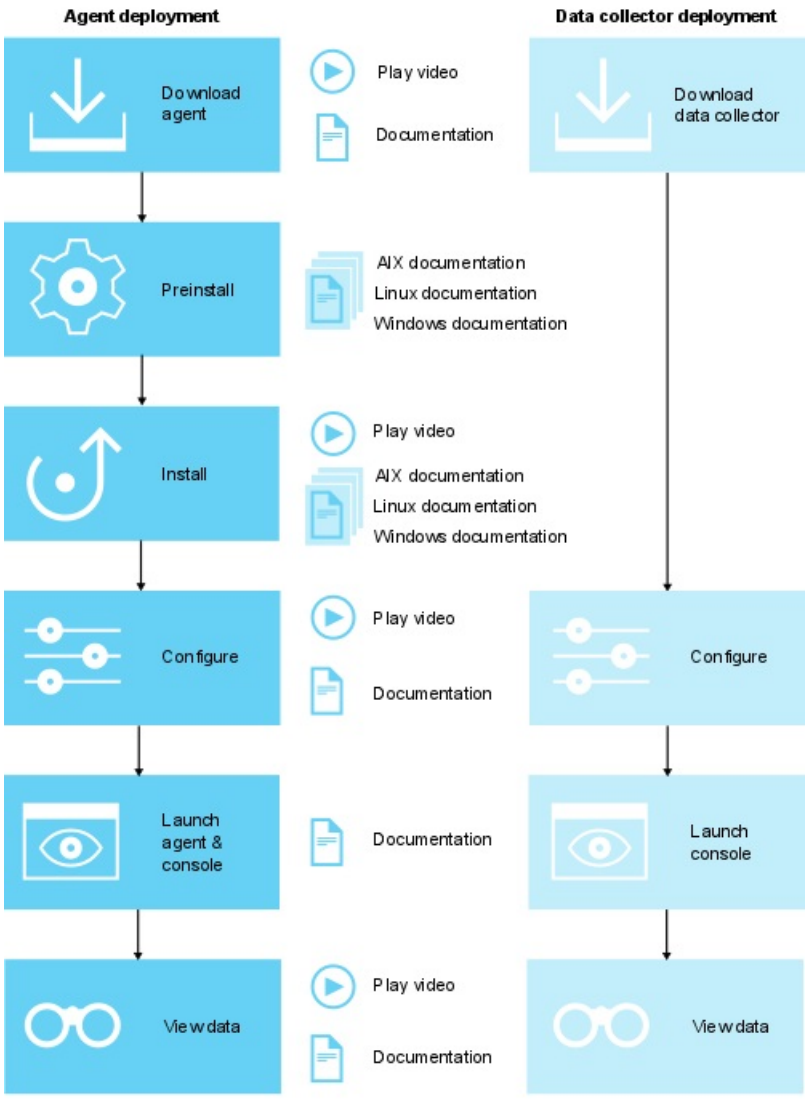
- 浏览控制台：在使用 Cloud APM 控制台时，浏览功能。您可以通过单击窗口条幅中的 以了解当前仪表板。您可以在 IBM Knowledge Center 上通过导航栏中的 **帮助**菜单打开帮助系统或者 Cloud APM 主题集合。



- 安装其他数据收集器：您有安装其他类型的数据收集器以监视环境所需的所有安装文件。您还可以在环境中的其他系统上安装数据收集器。有关更多信息，请参阅第 135 页的『第 7 章 配置环境』。

第 5 章 代理程序 以及数据收集器部署

代理程序在安装和查看收集的数据所需的任务之间是不同的。某些任务是自动的，而其他任务是手动的。下载数据收集器之后，您必须手动配置每个数据收集器。



1. 播放下载视频
2. 下载文档
3. AIX 预安装文档
4. Linux 预安装文档
5. Windows 预安装文档
6. 播放安装视频
7. AIX 安装文档
8. Linux 安装文档
9. Windows 安装文档
10. 播放配置视频

© Copyright IBM Corp. 2014, 2019

11. 配置文档
12. 启动文档
13. 播放视图数据视频
14. 功能部件文档

1. 播放下载视频
2. 下载文档
3. AIX 预安装文档
4. Linux 预安装文档
5. Windows 预安装文档
6. 播放安装视频
7. AIX 安装文档
8. Linux 安装文档
9. Windows 安装文档
10. 播放配置视频
11. 配置文档
12. 启动文档
13. 播放视图数据视频
14. 功能部件文档

安装后，某些代理程序会自动配置并启动
对于启动的任何代理程序，将使用缺省设置配置代理程序。要确定哪些代理程序是手动配置和启动的，请参阅第 96 页的表 7。

安装后，某些代理程序需要手动配置但是会自动启动
有关如何配置代理程序的信息，请参阅第 135 页的『第 7 章 配置环境』。要确定哪些代理程序是手动配置并自动启动的，请参阅第 96 页的表 7。

安装后，某些代理程序必须手动配置并启动
对于未自动启动的任何代理程序，必须先配置代理程序，然后才可以启动。要确定哪些代理程序是手动配置和启动的，请参阅第 96 页的表 7。

多实例代理程序需要手动创建和启动第一个实例
必须手动创建第一个实例并启动代理程序。多实例代理程序表示该代理程序的单次安装会为每个唯一应用程序实例实例化一个唯一的监视实例。之后，可从 Cloud APM 控制台看到这些实例。要确定哪些代理程序是多实例代理程序，请参阅第 96 页的表 7。

OS 代理程序和日志文件监视
将自动配置和启动 Linux OS 代理程序、UNIX OS 代理程序 和 Windows OS 代理程序。但是，您可以针对 OS 代理程序配置日志文件监视，从而可监视应用程序日志文件。有关更多信息，请参阅第 540 页的『配置操作系统代理程序日志文件监视』。

代理程序 以及数据收集器配置、启动和实例特征

表 7. 安装后核对表

代理程序 或数据收集器	自动配置和启动	手动配置并自动启动	手动配置和启动	多实例 （手动启动）
Amazon EC2 代理程序	—	—	—	✓
Amazon ELB 代理程序	—	—	—	✓
Azure Compute 代理程序	—	—	—	✓
Cassandra 代理程序	—	—	✓	✓
Cisco UCS agent	—	—	✓	✓

表 7. 安装后核对表 *(续)*

代理程序 或数据收集器	自动配置和启动	手动配置并自动启动	手动配置和启动	多实例 （手动启动）
Citrix VDI 代理程序	—	—	—	✓
DataPower 代理程序	—	—	—	✓
DataStage 代理程序	—	—	✓	✓
Db2 代理程序	—	—	—	✓
Hadoop 代理程序	—	—	✓	—
HMC Base 代理程序	—	—	—	✓
HTTP Server 代理程序	✓ 必须复审代理程序针对 HTTP 服务器创建的配置文件。然后，必须手动将数据收集器配置添加到服务器配置文件。	—	—	—
IBM Cloud 代理程序	—	—	—	✓
IBM Integration Bus 代理程序	—	—	—	✓
J2SE 数据收集器	—	✓	—	—
JBoss 代理程序	—	—	—	✓
Liberty 数据收集器	—	✓	—	—
Linux KVM 代理程序	—	—	✓	✓
Linux OS 代理程序	✓	—	—	—
Microsoft Active Directory 代理程序	—	✓ 自动启动此代理程序。但是，必须配置代理程序才能查看某些属性的数据。	—	—
Microsoft Cluster Server 代理程序	—	—	✓	—
Microsoft Exchange Server 代理程序	—	✓ 自动启动此代理程序。但是，必须配置代理程序才能查看所有属性的数据。	—	—
Microsoft Hyper-V Server 代理程序	✓	—	—	—
Microsoft IIS 代理程序	✓	—	—	—

表 7. 安装后核对表 *(续)*

代理程序 或数据收集器	自动配置和启动	手动配置并自动启动	手动配置和启动	多实例 （手动启动）
Skype for Business Server 代理程序（旧称 Microsoft Lync Server 代理程序）	✓	✓ 自动启动此代理程序。但是，必须配置代理程序才能查看某些属性的数据。	—	—
Microsoft Office 365 代理程序	—	—	✓	—
Microsoft .NET 代理程序	—	✓ 必须在报告数据前配置数据收集器。	—	—
Microsoft SharePoint Server 代理程序	✓	—	—	—
Microsoft SQL Server 代理程序	—	—	—	✓ 必须手动配置并启动每个代理程序实例。
MQ Appliance 代理程序	—	—	—	✓
MongoDB 代理程序	—	—	—	✓
MySQL 代理程序	—	—	—	✓
NetApp Storage 代理程序	—	—	✓	✓
Node.js 代理程序	—	✓ 必须在报告数据前配置代理程序。您必须将监视插件添加到 Node.js 应用程序。	—	—
Node.js 数据收集器	—	✓	—	—
OpenStack 代理程序	—	—	✓	✓
Oracle Database 代理程序	—	—	—	✓
PHP 代理程序	—	—	—	✓
PostgreSQL 代理程序	—	—	—	✓
Python 数据收集器	—	✓	—	—
RabbitMQ 代理程序	—	—	✓	✓
Response Time Monitoring Agent	✓	—	—	—

表 7. 安装后核对表 *(续)*

代理程序 或数据收集器	自动配置和启动	手动配置并自动启动	手动配置和启动	多实例 （手动启动）
Ruby 代理程序	—	—	—	✓ 要进行深入诊断，必须先配置代理程序，然后再报告数据。要启用诊断仪表板，必须安装和配置诊断数据收集器。
Ruby 数据收集器	—	✓	—	—
SAP 代理程序	—	—	✓	✓
SAP HANA Database 代理程序	—	—	✓	✓
SAP NetWeaver Java Stack 代理程序	—	—	✓	✓
Siebel 代理程序	—	—	—	✓
Sterling Connect Direct 代理程序	—	—	—	✓
Sterling File Gateway 代理程序	—	—	✓	✓
Sybase 代理程序	—	—	✓	✓
Synthetic Playback 代理程序	✓ 针对公共、面向外部的应用程序自动配置和启动代理程序，但是必须先在 Synthetic Script Manager 中创建事务，然后再报告数据。	✓ 代理程序将自动启动，但是必须针对专用、面向内部的应用程序配置代理程序。必须先在 Synthetic Script Manager 中创建事务，然后才能报告数据。	—	✓
Tomcat 代理程序	—	—	—	✓
UNIX OS 代理程序	✓	—	—	—
VMware VI 代理程序	—	—	✓	✓
WebLogic 代理程序	—	—	—	✓
WebSphere Applications 代理程序	—	✓ 该代理程序会自动启动，但时必须先配置数据收集器，然后再报告数据。	—	—

表 7. 安装后核对表 *(续)*

代理程序 或数据收集器	自动配置和启动	手动配置并自动启动	手动配置和启动	多实例 （手动启动）
WebSphere Infrastructure Manager 代理程序	–	–	–	✓
WebSphere MQ 代理程序	–	–	–	✓
Windows OS 代理程序	✓	–	–	–

第 6 章 安装代理程序

IBM Cloud Application Performance Management 基础结构由 IBM 安装并管理。要监视应用程序，请选择并安装所要监视的应用程序的相应监视代理程序。您可以在 Linux、AIX 或 Windows 操作系统上安装代理程序。 独立数据收集器不需要安装。

如果选择独立数据收集器来监视应用程序，可以跳过安装过程。继续第 135 页的『第 7 章 配置环境』，以了解有关如何部署数据收集器以监视应用程序的指示信息。

远程监视

某些代理程序可以与它们所监视的资源安装在不同位置。下列代理程序支持远程监视：

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS 弹性负载均衡器
- Monitoring Agent for Azure Compute
- Monitoring Agent for Cassandra
- Monitoring Agent for Cisco UCS
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for DataPower - 此代理程序只能安装在远程机器上。
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HMC Base
- Monitoring Agent for IBM Cloud
- Monitoring Agent for InfoSphere DataStage
- Monitoring Agent for JBoss - 如果要使用此代理程序进行资源监视，请将其安装在远程或本地位置。如果要使用此代理程序进行事务跟踪和深度诊断，请将其安装在本地位置。
- Monitoring Agent for Linux KVM
- Monitoring Agent for MariaDB
- Monitoring Agent for Microsoft Cluster Server
- Monitoring Agent for Microsoft Exchange Server
- Monitoring Agent for Microsoft Office 365
- Monitoring Agent for Microsoft SharePoint Server
- Monitoring Agent for MongoDB
- Monitoring Agent for MySQL
- Monitoring Agent for NetApp Storage
- Monitoring Agent for OpenStack
- Monitoring Agent for Oracle Database
- Monitoring Agent for PostgreSQL
- Monitoring Agent for RabbitMQ
- Monitoring Agent for SAP Applications
- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack - 如果要使用此代理程序进行资源监视，请将其安装在远程或本地位置。如果要使用此代理程序进行事务跟踪和深度诊断，请将其安装在本地位置。
- Monitoring Agent for Sterling Connect Direct
- Monitoring Agent for Sterling File Gateway

© Copyright IBM Corp. 2014, 2019

- Monitoring Agent for VMware VI
- Monitoring Agent for WebLogic - 如果要使用此代理程序进行资源监视，请将其安装在远程或本地位置。如果要使用此代理程序进行事务跟踪和深度诊断，请将其安装在本地位置。
- Response Time Monitoring Agent - 如果要使用 Packet Analyzer 组件，可以将此代理程序安装在远程或本地位置。如果要使用 IBM HTTP Server 响应时间模块，那么必须将此代理程序安装在 HTTP 服务器所在的机器上。

在 UNIX 系统上安装代理程序

在 AIX 或 Solaris 系统上针对要管理的资源安装监视代理程序。

可在 AIX 上安装的代理程序列表

- Monitoring Agent for DataPower
- Monitoring Agent for Cassandra
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HMC Base
- Monitoring Agent for HTTP Server
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for MQ Appliance
- Monitoring Agent for Oracle Database
- Monitoring Agent for SAP Applications
- Monitoring Agent for SAP HANA Database
- Monitoring Agent for SAP NetWeaver Java Stack
- Monitoring Agent for Siebel
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebLogic
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ
- Response Time Monitoring Agent

可在 Solaris Sparc 上安装的代理程序列表

- Monitoring Agent for Db2
- Monitoring Agent for HTTP Server
- Monitoring Agent for JBoss
- Monitoring Agent for MySQL
- Monitoring Agent for Oracle Database
- Monitoring Agent for SAP Applications
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebLogic

可在 Solaris X86 上安装的代理程序列表

- Monitoring Agent for Db2
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for SAP Applications
- Monitoring Agent for Sybase Server
- Monitoring Agent for UNIX OS
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ

AIX 系统上的安装前任务

您必须先完成所需的安装前任务，然后才能在 AIX 系统上安装代理程序。某些安装前任务特定于代理程序，而其他任务适用于多个代理程序。

注: 这些需求是对“系统产品兼容性报告”中所指示需求的补充。

要了解代理程序的当前版本需求及依赖关系，请参阅 System requirements (APM Developer Center)，以获得指向软件产品兼容性报告的链接。

所有代理程序

以下安装前任务适用于所有代理程序:

测试连接
在安装代理程序之前，请确保系统可以与 Cloud APM 服务器进行通信。有关检查与 Cloud APM 服务器的连接的信息，请参阅网络连接。

非 root 用户安装
您必须具有安装目录的读取、写入和执行许可权。否则，将取消安装。有关非 root 用户安装的更多信息，请参阅第 119 页的『以非 root 用户身份安装代理程序』。

安装路径限长 70 个字符
安装目录及其路径不得超过 70 个字符。

.tar 文件名限长 100 个字符
AIX 系统上的缺省 **tar** 命令无法处理长度大于 100 个字符的文件名。要避免安装问题，请完成下列步骤:

1. 从 AIX Toolbox for Linux Applications Web 站点下载并安装 **tar** 命令的 GNU 版本。
2. 使 GNU 版本成为缺省 **tar** 命令。完成下列其中一个步骤:
   - 将 /opt/freeware/bin 添加到当前 *PATH* 环境变量开头。例如:

     ```
     export PATH=/opt/freeware/bin:$PATH
     ```

     其中，/opt/freeware/bin 是 GUN bin 的目录。
   - 将 /bin/tar 替换为 /opt/freeware/bin/tar 的符号链接，如下所示:

     ```
     ln -s /opt/freeware/bin/tar /bin/tar
     ```

或者，升级到最新版本 AIX，以接收用于处理长度大于 100 个字符的文件名的代码修订。有关详细信息，请参阅 AIXV6.1 的 TAR 命令技术说明或 AIXV7.1 的 TAR 命令技术说明。

设置 *CANDLEHOME* 环境变量
如果使用 ITM Agent Converter 以在与之前相同的受管系统上安装和配置代理程序，那么将 *CANDLEHOME* 环境变量更改为随 Agent Converter 一起安装的代理程序所在的目录。在安装和配置本机 Cloud APM 代理程序之前，必须将 *CANDLEHOME* 环境变量设置为其他目录，否则，本机 Cloud APM 代理程序无法启动。

特定代理程序

以下安装前任务适用于指定的代理程序：

DataPower 代理程序
安装代理程序之前，先决条件检查程序将检查 *ulimit* 在 AIX 上是否设置为 **unlimited**。您必须运行 **ulimit -d unlimited** 命令，以确保 *max data segment size* 系统环境变量设置为 **unlimited**。 此代理程序不能与所要监视的 DataPower 设备安装在同一台机器上。

HMC Base 代理程序
如果您计划以 root 用户身份安装代理程序，则必须确保安装系统 TL07。如果您计划以非 root 用户身份安装代理程序，则必须确保仅针对 AIX V6 安装系统 TL08。

HTTP Server 代理程序
请以 root 用户身份安装并运行此代理程序。请使用同一个用户标识来安装并运行此代理程序。如果您以非 root 用户身份安装并运行代理程序，那么该非 root 用户必须与启动 IBM HTTP Server 的用户具有相同的用户标识。否则，此代理程序在发现 IBM HTTP Server 时会遇到问题。

在 AIX 上，安装将失败，这是因为在 AIX 系统上，缺省 **.tar** 命令会截断长路径。有关更多信息，请参阅本主题中的“.tar 文件名限长 100 个字符”部分。

仅限于 AIX：安装 lynx 实用程序或 curl 应用程序。

Oracle Database 代理程序
在 Red Hat Enterprise Linux V5 和 V6 以及 SUSE Linux Enterprise Server V11 和 V12 x64 上，如果 Oracle Database 代理程序远程监视 Oracle 数据库，那么必须先安装 Oracle 即时客户机。请安装 Oracle Technology Network - Instant Client Downloads 处提供的 Oracle 即时客户机。

Oracle Database 代理程序支持即时客户机 V10.x、V11.x 和 V12.x。

Response Time Monitoring Agent

在安装 Response Time Monitoring 代理程序之前，请查看安装规划部分：第 584 页的『规划安装 』。

SAP HANA Database 代理程序

1. 在 AIX 系统上安装 SAP HANA 数据库客户机 HDBSQL V1.00.102.06 或更高版本。
2. 运行以下命令以将安装目录的路径添加到 **LIBPATH** 环境变量：

   `export LIBPATH=$LIBPATH:`*`install_directory_path`*

   示例：`export LIBPATH=$LIBPATH:/usr/sap/hdbclient`，其中，`/usr/sap/hdbclient` 表示 SAP HANA 数据库客户机的安装路径。

要点:

如果 SAP HANA 数据库客户机的安装路径未添加到 **LIBPATH** 环境变量中，先决条件扫描程序将返回 FAIL 结果。

使用导出命令添加的环境变量仅存在于终端的特定会话中。因此，请确保从用于添加环境变量的同一终端运行代理程序安装脚本。

WebSphere Applications 代理程序
安装代理程序之前，先决条件检查程序将检查 *ulimit* 在 AIX 系统上是否设置为 **524000**。必须运行 **ulimit -d 524000** 命令来确保 *max data segment size* 系统环境变量设置为 **524000**。

Solaris 系统上的安装前任务

您必须先完成所需的安装前任务，然后才能在 Solaris 系统上安装代理程序。某些安装前任务特定于代理程序，而其他任务适用于多个代理程序。

注: 这些需求是对“系统产品兼容性报告”中所指示需求的补充。

要了解代理程序的当前版本需求及依赖关系，请参阅 System requirements (APM Developer Center)，以获得指向软件产品兼容性报告的链接。

所有代理程序

以下安装前任务适用于所有代理程序：

测试连接
在安装代理程序之前，请确保系统可以与 Cloud APM 服务器进行通信。有关检查与 Cloud APM 服务器的连接的信息，请参阅网络连接。

非 root 用户安装
您必须具有安装目录的读取、写入和执行许可权。否则，将取消安装。有关非 root 用户安装的更多信息，请参阅第 119 页的『以非 root 用户身份安装代理程序』。

安装路径限长 70 个字符
安装目录及其路径不得超过 70 个字符。

`.tar` 文件名限长 100 个字符
Solaris 系统上的缺省 **`tar`** 命令无法处理长度大于 100 个字符的文件名。要避免 @LongLink 错误问题，请完成以下步骤：

1. 从 http://www.gnu.org Web 站点下载并安装 **`tar`** 命令的 GNU 版本。
2. 使 GNU 版本成为缺省 **`tar`** 命令。完成下列其中一个步骤：
   - 在 *PATH* 环境变量中，首先放入以下变量：

     ```
     export PATH=/opt/freeware/bin:$PATH
     ```

   - 将 `/bin/tar` 替换为 `/opt/freeware/bin/tar` 的符号链接

设置 *CANDLEHOME* 环境变量
如果使用 ITM Agent Converter 以在与之前相同的受管系统上安装和配置代理程序，那么将 *CANDLEHOME* 环境变量更改为随 Agent Converter 一起安装的代理程序所在的目录。在安装和配置本机 Cloud APM 代理程序之前，必须将 *CANDLEHOME* 环境变量设置为其他目录，否则，本机 Cloud APM 代理程序无法启动。

特定代理程序

以下安装前任务适用于指定的代理程序：

HTTP Server 代理程序
请以 root 用户身份安装并运行此代理程序。请使用同一个用户标识来安装并运行此代理程序。如果您以非 root 用户身份安装并运行代理程序，那么该非 root 用户必须与启动 IBM HTTP Server 的用户具有相同的用户标识。否则，此代理程序在发现 IBM HTTP Server 时会遇到问题。

安装代理程序

您可以在受管系统上安装任意组合的监视代理程序。例如，如果安装 Ruby 代理程序以监视 Ruby On Rails 应用程序，那么可能还想要安装 Response Time Monitoring Agent 和/或 Linux OS 代理程序。使用 Response Time Monitoring 代理程序，您可以收集更多的 Ruby 应用程序的响应时间信息。使用 Linux OS 代理程序，您可以监视系统的其他方面，例如，总体 CPU、内存和磁盘。

产品确定哪些监视代理程序可用于安装。要获取每个产品中包含的代理程序列表，请参阅第 46 页的『功能』。

有关在 AIX 和 Solaris 系统上运行的代理程序的列表，请参阅第 102 页的『在 UNIX 系统上安装代理程序』。

开始之前

下载代理程序。请参阅第 86 页的『下载代理程序 以及数据收集器』。

查看第 69 页的『系统需求』中的信息，以确保满足您计划安装的代理程序的需求。

在安装代理程序之前查看代理程序安装前任务。

- 有关 AIX 系统，请参阅第 103 页的『AIX 系统上的安装前任务』。
- 有关 Solaris 系统，请参阅第 104 页的『Solaris 系统上的安装前任务』。

要点: 仅当代理程序需要 Java 运行时的情况下才会加以安装，并且它并非总是可用。另外，代理程序安装不再需要 ksh，并且支持强制模式的 SELinux。

关于此任务

您能以 root 用户或非 root 用户身份安装监视代理程序。如果您不具有 root 特权，并希望安装监视代理程序，那么可采用非 root 用户身份安装该代理程序，请参阅第 119 页的『以非 root 用户身份安装代理程序』。另外，如果您是主机管理员并且不想以 root 用户身份运行监视代理程序，那么可以使用非 root 用户身份安装代理程序。安装流程与 root 用户相同。

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Cloud Application Performance Management 代理程序。但是，这两种代理程序类型不能安装在同一目录中。有关代理程序共存的更多信息，请参阅 第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

过程

1. 打开 AIX 或 Solaris 系统上的终端 shell 会话。
2. 在系统上，浏览至下载 .tar 文件的目录。

   必须在安装想要监视的应用程序的系统上安装代理程序。如果需要，将安装归档文件传输至要监视的系统。归档文件包含代理程序和安装脚本。

   切记: 请确保目录不包含较旧版本的归档文件。
3. 通过使用以下命令来抽取 安装文件:

   ```
   tar -xf ./installation files
   ```

   其中，*installation files* 是产品的安装文件名。

   安装脚本解压缩到以归档文件和版本命名的目录。例如：*offering*_Agent_Install_8.1.4.0。代理程序二进制和配置相关文件解压缩至目录中的子目录。
4. 可选： 仅 Solaris 10 需要此步骤。在 Solaris 10 上运行安装脚本之前，必须创建指向 ksh 的软链接。

   a) 备份 /bin/sh 命令:

   ```
   mv /bin/sh /bin/sh.bkup_origin
   ```

   b) 创建指向 ksh 命令的软链接:

   ```
   ln -s /bin/ksh /bin/sh
   ```

   c) 确认结果指向 ksh:

   ```
   ls -l /bin/sh
   ```
5. 从以归档文件和版本命名的目录运行安装脚本:

   ```
   ./installAPMAgents.sh
   ```

   要以静默方式安装代理程序，请参阅第 121 页的『以静默方式安装代理程序』。
6. 指定是安装个别代理程序、多个代理程序的组合还是所有代理程序。
7. 根据是安装还是升级代理程序，执行以下某个步骤:
   - 如果正在安装代理程序，请指定不同的代理程序安装主目录或者使用适用的缺省目录:
     - /opt/ibm/apm/agent
   - 如果您正在升级代理程序，在提示输入代理程序安装主目录后，输入先前版本的代理程序的安装目录。
8. 在系统询问您是否接受许可协议时，输入 1 以接受协议并继续，或者输入 2 以拒绝。

在输入 1（表示接受）后，将针对您的环境启动先决条件扫描，该扫描需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。诸如缺少库或磁盘空间不足等先决条件会停止安装。您必须解决故障，然后重新启动安装脚本。

9. 如果使用非 root 用户标识安装代理程序，那么必须更新系统启动脚本（请参阅第 119 页的『以非 root 用户身份安装代理程序』）。
10. 在安装完成并且命令行可用后，您可以重复此过程中的步骤以在受管系统上安装更多监视代理程序。

下一步做什么

根据需要配置代理程序。如果监视代理程序需要按第 95 页的『第 5 章 代理程序 以及数据收集器部署』中的描述进行配置。或者您想要复查缺省设置，请参阅第 135 页的『第 7 章 配置环境』。

- 如果由于防火墙不允许与外部主机进行透明出站 HTTPS 连接而使用转发代理，那么必须编辑代理程序环境配置文件。有关指示信息，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。
- 如果您从先前版本升级代理程序，请标识在登录 Cloud APM 控制台之前必须完成的任何重新配置或迁移任务。有关这些任务的信息，请参阅第 961 页的『升级代理程序』。升级之后，您必须重新启动安装程序不会自动配置并启动的任何代理程序。

要启动代理程序，请运行以下命令：

```
./name-agent.sh start
```

有关监视代理程序命令的信息，包括要使用的 *name*，请参阅第 150 页的『使用代理程序命令』。有关自动启动和手动启动的代理程序的信息，请参阅第 95 页的『第 5 章 代理程序 以及数据收集器部署』。

升级之后，您必须重新启动安装程序不会自动配置并启动的任何代理程序。

在配置并启动代理程序之后，查看代理程序正在收集的数据。

- 如果您尚未登录，请按第 829 页的『启动 Cloud APM 控制台』中的指示信息执行操作。
- 如果想要在应用程序性能仪表板中查看来自 IBM Tivoli Monitoring 域的受管系统，请完成第 807 页的『与 IBM Tivoli Monitoring V6.3 集成 』中描述的任务。

在 Linux 系统上安装代理程序

在 Linux 系统上针对要管理的资源安装监视代理程序。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS 弹性负载均衡器
- Monitoring Agent for Azure Compute
- Monitoring Agent for Cassandra
- Monitoring Agent for Cisco UCS
- Monitoring Agent for Citrix Virtual Desktop Infrastructure
- Monitoring Agent for DataPower
- Monitoring Agent for Db2
- Monitoring Agent for Hadoop
- Monitoring Agent for HTTP Server
- Monitoring Agent for IBM Cloud
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for Internet Services
- Monitoring Agent for MQ Appliance
- Monitoring Agent for InfoSphere DataStage
- Monitoring Agent for JBoss
- Monitoring Agent for Linux OS

· Monitoring Agent for Linux KVM
· Monitoring Agent for MariaDB
· Monitoring Agent for Microsoft SQL Server
· Monitoring Agent for MongoDB
· Monitoring Agent for MySQL
· Monitoring Agent for NetApp Storage
· Monitoring Agent for Node.js
· Monitoring Agent for OpenStack
· Monitoring Agent for Oracle Database
· Monitoring Agent for PHP
· Monitoring Agent for PostgreSQL
· Monitoring Agent for RabbitMQ
· Monitoring Agent for Ruby
· Monitoring Agent for SAP Applications
· Monitoring Agent for SAP HANA Database
· Monitoring Agent for SAP NetWeaver Java Stack
· Monitoring Agent for Siebel
· Monitoring Agent for Sterling Connect Direct
· Monitoring Agent for Sterling File Gateway
· Monitoring Agent for Sybase Server
· Monitoring Agent for Synthetic Playback
· Monitoring Agent for Tomcat
· Monitoring Agent for VMware VI
· Monitoring Agent for WebLogic
· Monitoring Agent for WebSphere Applications
· Monitoring Agent for WebSphere Infrastructure Manager
· Monitoring Agent for WebSphere MQ
· Response Time Monitoring Agent

在 Linux on Power Little Endian (pLinux LE) 系统上，支持下列代理程序：

· Monitoring Agent for Db2
· Monitoring Agent for IBM Integration Bus
· Monitoring Agent for Linux OS
· Monitoring Agent for Tomcat - 支持资源监视。
· Monitoring Agent for WebSphere Applications
· Monitoring Agent for WebSphere MQ

在 Linux for System z 系统上支持以下代理程序：

· Monitoring Agent for Db2
· Monitoring Agent for HTTP Server - 不支持事务跟踪。
· Monitoring Agent for IBM Integration Bus
· Monitoring Agent for Linux OS
· Response Time Monitoring Agent
· Monitoring Agent for Tomcat
· Monitoring Agent for WebSphere Applications

· Monitoring Agent for WebSphere MQ

以下代理程序在 Linux for System x 系统上受支持：

· Monitoring Agent for HTTP Server - 不支持事务跟踪。

Linux 系统上的安装前任务

您必须先完成所需的安装前任务，然后才能在 Linux 系统上安装代理程序。某些安装前任务特定于代理程序，而其他任务适用于多个代理程序。

注: 这些需求是对“系统产品兼容性报告”中所指示需求的补充。

要了解代理程序的当前版本需求及依赖关系，请参阅 System requirements (APM Developer Center)，以获得指向软件产品兼容性报告的链接。

所有代理程序

以下安装前任务适用于所有代理程序：

测试连接
在安装代理程序之前，请确保系统可以与 Cloud APM 服务器进行通信。有关检查与 Cloud APM 服务器的连接的信息，请参阅网络连接。

非 root 用户安装
您必须具有安装目录的读取、写入和执行许可权。否则，将取消安装。有关非 root 用户安装的更多信息，请参阅第 119 页的『以非 root 用户身份安装代理程序』。

安装路径限长 70 个字符
安装目录及其路径不得超过 70 个字符。

设置 *CANDLEHOME* 环境变量
如果使用 ITM Agent Converter 以在与之前相同的受管系统上安装和配置代理程序，那么将 *CANDLEHOME* 环境变量更改为随 Agent Converter 一起安装的代理程序所在的目录。在安装和配置本机 Cloud APM 代理程序之前，必须将 *CANDLEHOME* 环境变量设置为其他目录，否则，本机 Cloud APM 代理程序无法启动。

特定操作系统

Red Hat Enterprise Linux (RHEL) 8

RHEL 8 上需要 libnsl.so.1 软件包

缺省情况下，`libnsl.so.1` 未安装在 Red Hat Enterprise Linux R8.0 中。如果没有此软件包，那么无法成功安装任何代理程序。请求管理员为您设置 yum 存储库，然后运行以下命令：

```
yum install libnsl
```

成功安装后，您可以查看 `/usr/lib64/libnsl.so.1`。

注: 仅代理程序需要 `libnsl.so.1` 软件包。您无需为数据收集器执行此步骤。

绕过某些代理程序的先决条件扫描程序

在将先决条件扫描程序更新为兼容最新需求之前，对于某些代理程序，您可以绕过先决条件扫描程序。有关适合的场景和指示信息，请参阅第 123 页的『绕过先决条件扫描程序』。

注: 您无需为数据收集器执行此步骤。

特定代理程序

以下安装前任务适用于指定的代理程序：

DataPower 代理程序
您必须运行 **ulimit -d unlimited** 命令，以确保 *max data segment size* 系统环境变量设置为 **unlimited**。 此代理程序不能与所要监视的 DataPower 设备安装在同一台机器上。

DataStage 代理程序

1. 在 DSODBConfig.cfg 文件中启用参数。请完成下列步骤：
   a. 在编辑器中打开以下位置的 DSODBConfig.cfg 文件：

      infosphere_information_server_install_dir/Server/DSODB

   b. 通过移除 # 符号取消注释下列参数：

      MonitorLinks=1
      JobRunUsage=1
      ResourceMonitor=1
      DSODBON=1

   c. 编辑这些参数的值，使其等于 1。
2. 复制代理程序计算机上用于元数据存储库配置的数据库 JDBC 驱动程序。
   a. 第 4 类 JDBC 4 或更高版本。例如，db2jcc4.jar
   b. 适用于 Oracle 的第 4 类 JDBC 驱动程序。例如，ojdbc6.jar
   c. 适用于 MS SQL 的 JDBC 驱动程序：
      · Sqljdbc41.jar 需要 JRE 7 并支持 JDBC 4.1 API。
      · Sqljdbc42.jar 需要 JRE 8 并支持 JDBC 4.2 API。

HTTP Server 代理程序

如果以 root 用户身份安装此代理程序，必须使用该同一用户标识运行和配置此代理程序。

如果您以非 root 用户身份安装并运行代理程序，那么该非 root 用户必须与启动 IBM HTTP Server 的用户具有相同的用户标识。否则，此代理程序在发现 IBM HTTP Server 时会遇到问题。您可以使用同一个用户标识来运行和配置代理程序。

Linux KVM 代理程序

Monitoring Agent for Linux KVM 是多实例、多连接的代理程序，它支持连接到基于 Enterprise Linux 的 KVM 系统管理程序以及 Red Hat Enterprise Virtualization Manager (RHEV-M) 环境。您可以创建此代理程序的多个实例，以用于监视 RHEV-M 或 KVM 系统管理程序环境中的多个系统管理程序。您还可以在不同虚拟机之间监视虚拟化工作负载并分析资源容量。要将代理程序连接到 KVM 管理程序环境中的虚拟机，必须安装必备软件：libvirt*.rpm 和 Korn Shell Interpreter (pdksh)。此代理程序将与用于管理虚拟机的 libvirt 系统管理程序建立远程连接，从而收集度量。

Microsoft SQL Server 代理程序

要监视 Microsoft SQL 环境，必须在安装 Monitoring Agent for Microsoft SQL Server 之前安装 Microsoft SQL Server 和 Microsoft SQL ODBC 驱动程序。例如，要在 Red Hat Enterprise Linux 上安装 ODBC 驱动程序，请使用以下命令：

```
sudo yum install unixODBC
sudo yum install msodbcsql17
```

要完成执行必备软件检查程序，需要在 Cloud Application Performance Management V8.1.4.0 服务器临时修订 15 (8.1.4.0-IBM-APM-SERVER-IF0015.tar) 或更高版本上配置代理程序。

MongoDB 代理程序

您必须在安装 MongoDB 数据库服务器的系统上安装和配置 MongoDB 代理程序。

MySQL 代理程序

要监视 MySQL 环境，必须先安装 MySQL 服务器和 MySQL JDBC 驱动程序，然后才能安装 Monitoring Agent for MySQL。例如，要在 Red Hat Enterprise Linux 上安装 JDBC 驱动程序，请使用以下命令：

```
yum install mysql-connector-java
```

启动代理程序安装后，在对 MySQL 程序包名称进行先决条件检查期间，如果使用的提供者不是 Red Hat（例如使用的是 Oracle），那么可能会看到一条警告。如果 MySQL 服务器和 JDBC 驱动程序可用，那么该警告不会导致安装失败，您可以忽略此消息。样本输出：

```
Scenario: Prerequisite Scan
KSE - Monitoring Agent for MySQL Prerequisite Checker [version 01000005]:
Property                                Result   Found
Expected
========                          ====== =====       ========
os.package.mysql-server         WARN  Unavailable  mysql-server-5.1.12+
os.package.mysql-connector-java WARN Unavailable mysql-connector-java-5.
1.12+
```

Node.js 代理程序

用于运行受监视应用程序的 Node.js 版本必须与缺省的安装版本相同。

当前不支持 Node.js v5。

OpenStack 代理程序

在可以使用 OpenStack 代理程序之前，必须具有安装代理程序的位置的服务器上的以下软件：

- Python 2.6.0 或更高版本，或 Python 2.7.0 或更高版本
- 最新的 OpenStack 客户机：
  - OpenStack
  - Keystone
  - Neutron
  - Swift

  要安装 OpenStack 命令行客户机，请参阅安装 OpenStack 命令行客户机。
- Python 中用于远程访问的 Paramiko 库

  注: 如果想要在干净的 RedHat Linux 服务器上安装 OpenStack 代理程序，请在安装 Paramiko 库之前，运行以下命令来安装所需软件：

  ```
  wget https://ftp.dlitz.net/pub/dlitz/crypto/pycrypto/pycrypto-2.6.1.tar.gz
  yum install gcc/openssl-devel/libffi-devel
  ```
- KornShell

Oracle Database 代理程序

在 Red Hat Enterprise Linux V5 和 V6 以及 SUSE Linux Enterprise Server V11 和 V12 x64 上，如果 Oracle Database 代理程序远程监视 Oracle 数据库，那么必须先安装 Oracle 即时客户机。请安装 Oracle Technology Network - Instant Client Downloads 处提供的 Oracle 即时客户机。

Oracle Database 代理程序支持即时客户机 V10.x、V11.x 和 V12.x。

PHP 代理程序

如果使用 root 用户部署 PHP 应用程序，那么必须使用 root 用户来安装、配置、启动或停止代理程序。如果使用非 root 用户部署 PHP 应用程序，那么可以使用 root 用户或相同的非 root 用户来安装、配置、启动或停止代理程序。

必须已安装现有的 WordPress 应用程序。PHP 代理程序监视 WordPress V3.7.1 或更高版本

代理程序仅评估 WordPress 应用程序中 PHP 请求的性能。不会对 CSS 和 JS 装入进行评估。

代理程序不会使用 URL 参数来标识 URL。

Python 数据收集器

Python 数据收集器用于监视 Django 应用程序。

Response Time Monitoring Agent

在安装 Response Time Monitoring 代理程序之前，请查看安装规划部分：第 584 页的『规划安装 』。

SAP HANA Database 代理程序

1. 在 Linux 系统上安装 SAP HANA 数据库客户机 HDBSQL V1.00.102.06 或更高版本。

   要点: 对于 RHEL 5.x 64 位操作系统，安装 Linux SUSE 9 on x86_64 64 位 SAP HANA 数据库客户机，而不是 Linux on x86_64 64 位。对于 RHEL 6.x 或更高版本的 64 位操作系统，安装 Linux on x86_64 64 位 SAP HANA 数据库客户机。

2. 运行以下命令以将安装目录的路径添加到 **LD_LIBRARY_PATH** 环境变量：

   `export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:`*`install_directory_path`*

   示例： `export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/sap/hdbclient`，其中，`/usr/sap/hdbclient` 表示 SAP HANA 数据库客户机的安装路径。

要点:

如果 SAP HANA 数据库客户机的安装路径未添加到 **LD_LIBRARY_PATH** 环境变量中，先决条件扫描程序将返回 FAIL 结果。

使用导出命令添加的环境变量仅存在于终端的特定会话中。因此，请确保从用于添加环境变量的同一终端运行代理程序安装脚本。

Synthetic Playback 代理程序

要安装 Synthetic Playback 代理程序，操作系统用户需要以下许可权：

· 为安装映像启用读取和执行许可权

· 为代理程序主页启用写许可权

要运行 Synthetic Playback 代理程序，操作系统用户需要以下许可权：

· 为代理程序安装位置及其子目录和文件启用读、写和执行许可权。

· 启用许可权以运行 Mozilla Firefox。

· 请确保 Mozilla Firefox 执行二进制位于用户概要文件的 PATH 环境变量中。

在安装 Synthetic Playback 代理程序之前，必须完成以下步骤：

1. 将代理程序安装位置与 Cloud APM 控制台同步。
2. 安装 Mozilla Firefox 和 Xvfb 显示服务器。
3. 验证 Xvfb 显示服务器是否正在工作。运行以下命令：

   ```
   # Xvfb -ac
   ```

   不应有错误输出。
4. 检查 Xvfb 进程是否正在运行。运行以下命令：

   ```
   # ps -ef|grep Xvfb
   ```

   样本输出：

   ```
   root 7192 1 0 Jan14 ? 00:00:14 Xvfb -ac
   root 20393 17900 0 02:05 pts/0 00:00:00 grep -i xvfb
   ```
5. 停止 Xvfb 进程。运行以下命令：

   ```
   # kill -9 7192
   ```
6. 导航到 *`install_dir`*`/etc/hosts` 并编辑 hosts 文件开头以包含下列参数：

   ```
   127.0.0.1 localhost
   ```

   然后，保存并关闭 hosts 文件。

WebSphere Applications 代理程序

安装代理程序之前，先决条件检查程序将检查 *ulimit* 在 Linux 系统上是否设置为 **524000**。必须运行 **ulimit -d 524000** 命令来确保 *max data segment size* 系统环境变量设置为 **524000**。

安装代理程序

您可以在受管系统上安装任意组合的监视代理程序。例如，如果安装 Ruby 代理程序以监视 Ruby On Rails 应用程序，那么可能还想要安装 Response Time Monitoring Agent 和/或 Linux OS 代理程序。使用

Response Time Monitoring 代理程序，您可以收集更多的 Ruby 应用程序的响应时间信息。使用 Linux OS 代理程序，您可以监视系统的其他方面，例如，总体 CPU、内存和磁盘。

产品确定哪些监视代理程序可用于安装。要获取每个产品中包含的代理程序列表，请参阅第 46 页的『功能』。

有关在 Linux 系统上运行的代理程序的列表，请参阅第 107 页的『在 Linux 系统上安装代理程序』。

开始之前

下载代理程序。请参阅第 86 页的『下载代理程序 以及数据收集器』。

查看第 69 页的『系统需求』中的信息，以确保满足您计划安装的代理程序的需求。

在安装代理程序之前查看代理程序安装前任务。有关详细信息，请参阅第 109 页的『Linux 系统上的安装前任务』。

注: 仅当代理程序需要 Java 运行时的情况下才会加以安装，并且它并非总是可用。此外，代理程序安装也不再需要 ksh，但 Summarization and Pruning Agent 安装除外，该组件在 Cloud APM 服务器安装期间进行安装。支持强制模式的 SELinux。

关于此任务

您能以 root 用户或非 root 用户身份安装监视代理程序。如果您不具有 root 特权，并希望安装监视代理程序，那么可采用非 root 用户身份安装该代理程序，请参阅第 119 页的『以非 root 用户身份安装代理程序』。另外，如果您是主机管理员并且不想以 root 用户身份运行监视代理程序，那么可以使用非 root 用户身份安装代理程序。安装流程与 root 用户相同。

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Cloud Application Performance Management 代理程序。但是，这两种代理程序类型不能安装在同一目录中。有关代理程序共存的更多信息，请参阅 第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

过程

1. 在 Red Hat Enterprise Linux 系统上打开终端 shell 会话。
2. 在系统上，浏览至下载 .tar 文件的目录

   必须在安装想要监视的应用程序的系统上安装代理程序。如果需要，将安装归档文件传输至要监视的系统。归档文件包含代理程序和安装脚本。

   切记: 请确保目录不包含较旧版本的归档文件。
3. 根据您的产品，使用以下命令解压缩 安装文件：

   ```
   tar -xf ./installation files.tar
   ```

   其中，*installation files* 是产品的安装文件名。

   安装脚本解压缩到以归档文件和版本命名的目录。例如：*offering*_Agent_Install_8.1.4.0。代理程序二进制和配置相关文件解压缩至目录中的子目录。
4. 从以归档文件和版本命名的目录运行安装脚本：

   ```
   ./installAPMAgents.sh
   ```

   要以静默方式安装代理程序，请参阅第 121 页的『以静默方式安装代理程序』。
5. 指定是安装个别代理程序、多个代理程序的组合还是所有代理程序。
6. 根据是安装还是升级代理程序，执行以下某个步骤：
   - 如果正在安装代理程序，请指定不同的代理程序安装主目录或者使用适用的缺省目录：
     - /opt/ibm/apm/agent
   - 如果您正在升级代理程序，在提示输入代理程序安装主目录后，输入先前版本的代理程序的安装目录。

a. 如果 /opt/ibm/apm/agent 目录中存在较旧版本的代理程序，那么必须指定新的安装目录。在下一步中，将询问您是否想要从 /opt/ibm/apm/agent 目录迁移代理程序配置。

b. 如果确认要将代理程序配置从旧安装目录 (/opt/ibm/ccm/agent) 迁移到新安装目录（例如，/opt/ibm/apm/agent），那么必须在新安装位置启动代理程序。

限制: 旧安装位置中的较旧版本代理程序将自动停止，但是新安装位置中的代理程序不会自动启动。

c. 在安装完成并且验证代理程序在新安装目录中正常工作后，必须从 /opt/ibm/ccm/agent 目录卸载较旧版本的代理程序。如果想要移除所有代理程序，请运行 **/opt/ibm/ccm/agent/bin/smai-agent.sh uninstall_all** 命令。

7. 在系统询问您是否接受许可协议时，输入 1 以接受协议并继续，或者输入 2 以拒绝。

   在输入 1（表示接受）后，将针对您的环境启动先决条件扫描，该扫描需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。缺少先决条件（诸如缺少库或磁盘空间不足）会停止安装。您必须解决故障，然后重新启动安装脚本。

   注: 如果安装退出并返回以下消息，请检查服务器服务是否已启动（启动 -> 管理工具 -> 服务）。如果未启动，那么再次启动服务器服务并运行 installAPMAgents.bat。

   ```
   此脚本 [installAPMAgents.bat] 必须以管理员身份运行。
   ```

8. 如果使用非 root 用户标识安装代理程序，那么必须更新系统启动脚本（请参阅第 119 页的『以非 root 用户身份安装代理程序』）。
9. 在安装完成并且命令行可用后，您可以重复此过程中的步骤以在受管系统上安装更多监视代理程序。

下一步做什么

根据需要配置代理程序。如果监视代理程序需要按第 95 页的『第 5 章 代理程序 以及数据收集器部署』中的描述进行配置。或者您想要复查缺省设置，请参阅第 135 页的『第 7 章 配置环境』。

- 如果由于防火墙不允许与外部主机进行透明出站 HTTPS 连接而使用转发代理，那么必须编辑代理程序环境配置文件。有关指示信息，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。
- 如果您从先前版本升级代理程序，请标识在登录 Cloud APM 控制台之前必须完成的任何重新配置或迁移任务。有关这些任务的信息，请参阅第 961 页的『升级代理程序』。升级之后，您必须重新启动安装程序不会自动配置并启动的任何代理程序。

要启动代理程序，请运行以下命令：

```
./name-agent.sh start
```

有关监视代理程序命令的信息，包括要使用的名称，请参阅第 150 页的『使用代理程序命令』。有关自动启动和手动启动的代理程序的信息，请参阅第 95 页的『第 5 章 代理程序 以及数据收集器部署』。

升级之后，您必须重新启动安装程序不会自动配置并启动的任何代理程序。

在配置并启动代理程序之后，查看代理程序正在收集的数据。

- 如果您尚未登录，请按第 829 页的『启动 Cloud APM 控制台』中的指示信息执行操作。
- 如果想要在应用程序性能仪表板中查看来自 IBM Tivoli Monitoring 域的受管系统，请完成第 807 页的『与 IBM Tivoli Monitoring V6.3 集成 』中描述的任务。

在 Windows 系统上安装代理程序

您可以在 Windows 系统上安装部分 Cloud APM 监视代理程序。

Windows 64 位系统上支持以下监视代理程序。这表示 Windows 32 位系统上也支持这些代理程序。

- Monitoring Agent for Amazon EC2
- Monitoring Agent for AWS 弹性负载均衡器
- Monitoring Agent for Azure Compute

· Monitoring Agent for Cassandra
· Monitoring Agent for Cisco UCS
· Monitoring Agent for Citrix Virtual Desktop Infrastructure
· Monitoring Agent for Db2
· Monitoring Agent for Hadoop
· Monitoring Agent for HTTP Server*
· Monitoring Agent for IBM Cloud
· Monitoring Agent for IBM Integration Bus
· Monitoring Agent for Internet Services*
· Monitoring Agent for MQ Appliance
· Monitoring Agent for JBoss
· Monitoring Agent for MariaDB
· Monitoring Agent for Microsoft Active Directory*
· Monitoring Agent for Microsoft Cluster Server*
· Monitoring Agent for Microsoft Exchange Server
· Monitoring Agent for Microsoft Hyper-V Server
· Monitoring Agent for Microsoft Internet Information Services
· Monitoring Agent for Skype for Business Server（旧称 Microsoft Lync Server）*
· Monitoring Agent for Microsoft .NET
· Monitoring Agent for Microsoft Office 365
· Monitoring Agent for Microsoft SharePoint Server
· Monitoring Agent for Microsoft SQL Server*
· Monitoring Agent for MySQL
· Monitoring Agent for NetApp Storage
· Monitoring Agent for Oracle Database
· Monitoring Agent for PostgreSQL
· Monitoring Agent for RabbitMQ
· Monitoring Agent for SAP Applications
· Monitoring Agent for SAP HANA Database
· Monitoring Agent for SAP NetWeaver Java Stack
· Monitoring Agent for Siebel
· Monitoring Agent for Sterling Connect Direct
· Monitoring Agent for Sterling File Gateway
· Monitoring Agent for Sybase Server
· Monitoring Agent for Tomcat
· Monitoring Agent for VMware VI
· Monitoring Agent for WebLogic
· Monitoring Agent for WebSphere Applications
· Monitoring Agent for WebSphere MQ
· Monitoring Agent for Windows OS*
· Response Time Monitoring Agent*

* 在 64 位和 32 位 Windows 系统上均受支持。

Windows 系统上的安装前任务

您必须先完成所需的安装前任务，然后才能在 Windows 系统上安装代理程序。某些安装前任务特定于代理程序，而其他任务适用于多个代理程序。

注: 这些需求是对“系统产品兼容性报告”中所指示需求的补充。

要了解代理程序的当前版本需求及依赖关系，请参阅 System requirements (APM Developer Center)，以获得指向软件产品兼容性报告的链接。

所有代理程序

以下安装前任务适用于所有代理程序:

测试连接

在安装代理程序之前，请确保系统可以与 Cloud APM 服务器进行通信。有关检查与 Cloud APM 服务器的连接的信息，请参阅网络连接。

从本地驱动器上的命令提示符安装

使用 Windows 命令提示符来启动安装脚本。请勿使用 Windows PowerShell 来启动安装脚本。

将安装文件复制到本地磁盘或映射的网络驱动器，然后启动安装脚本。请勿从网络位置启动安装脚本。

从新命令提示符启动安装脚本。请勿从现有命令提示符启动安装脚本，因为命令提示符导致环境变量过时。

设置 *CANDLEHOME* 环境变量

如果使用 ITM Agent Converter 以在与之前相同的受管系统上安装和配置代理程序，那么将 *CANDLEHOME* 环境变量更改为随 Agent Converter 一起安装的代理程序所在的目录。在安装和配置本机 Cloud APM 代理程序之前，必须将 *CANDLEHOME* 环境变量设置为其他目录，否则，本机 Cloud APM 代理程序无法启动。

特定代理程序

以下安装前任务适用于指定的代理程序:

DataStage 代理程序

1. 在 DSODBConfig.cfg 文件中启用参数。请完成下列步骤:
   a. 在编辑器中打开以下位置的 DSODBConfig.cfg 文件:

      ```
      infosphere_information_server_install_dir\Server\DSODB
      ```

   b. 通过移除 # 符号取消注释下列参数:

      ```
      MonitorLinks=1
      JobRunUsage=1
      ResourceMonitor=1
      DSODBON=1
      ```

   c. 编辑这些参数的值，使其等于 1。
2. 复制代理程序计算机上用于元数据存储库配置的数据库 JDBC 驱动程序。
   a. 第 4 类 JDBC 4 或更高版本。例如，db2jcc4.jar
   b. 适用于 Oracle 的第 4 类 JDBC 驱动程序。例如，ojdbc6.jar
   c. 适用于 MS SQL 的 JDBC 驱动程序:
      - Sqljdbc41.jar 需要 JRE 7 并支持 JDBC 4.1 API。
      - Sqljdbc42.jar 需要 JRE 8 并支持 JDBC 4.2 API。

IBM Integration Bus 代理程序

确保用于安装 IBM Integration Bus 代理程序 的用户标识位于 mqbrkrs 用户组中。

Internet Service Monitoring

对于 Internet Service Monitoring，您必须从此处将 IBM Cloud Application Performance Management 8.1.4.0 核心框架临时修订 3 应用于 APM 服务器，然后预配置代理程序。代理程序和网桥模块使用端口 9510 和 9520。如果这些端口已在使用，那么安装将不继续。

注:

- 对于现有用户，建议在 64 位平台（Windows 或 Linux）上安装 Internet Service Monitoring 代理程序，而不是将 Windows 32 位平台上的代理程序升级到更新版本。
- Internet Service Monitoring 代理程序不支持 Windows 64 位平台上的 Windows 2008 R2。

MySQL 代理程序

对于 Monitoring Agent for MySQL，必须先安装 MySQL 服务器和 MySQL JDBC 驱动程序，然后才能在该系统上安装 MySQL 代理程序。要安装 JDBC 驱动程序，请参阅 MySQL Connector/J JDBC 驱动程序。

Oracle Database 代理程序

如果 Oracle Database 代理程序远程监视 Oracle 数据库，那么首先必须从 Oracle Technology Network - Instant Client Downloads 在以下系统上安装 Oracle 即时客户机：

- Windows Server 2012 64 位
- Windows Server 2012 R2 64 位
- Windows Server 2008 R2 Datacenter 64 位
- Windows Server 2008 R2 Enterprise 64 位
- Windows Server 2008 R2 Standard 64 位

Oracle Database 代理程序支持即时客户机 V10.x、V11.x 和 V12.x。

Response Time Monitoring Agent

在安装 Response Time Monitoring 代理程序之前，请查看安装规划部分：第 584 页的『规划安装 』。

SAP HANA Database 代理程序

1. 在 Windows 系统上安装 SAP HANA 数据库客户机 HDBSQL V1.00.102.06 或更高版本。
2. 将 SAP HANA 客户机的安装路径添加到 **PATH** 环境变量中。

   示例：将 `C:\Program Files\sap\hdbclient` 添加到 **PATH** 环境变量中，其中 `C:\Program Files\sap\hdbclient` 表示 SAP HANA 数据库客户机的安装路径。

Tomcat 代理程序

1. Java SDK 安装在代理程序安装所在的 Tomcat 服务器上。
2. 请先将 SDK 路径直接添加到 *PATH* 变量或者使用 **set path** 命令进行添加，然后再安装代理程序。
3. **JAR** 命令可以工作。

安装代理程序

您可以在受管系统上安装任意组合的监视代理程序。例如，如果安装 Monitoring Agent for MySQL 以监视 MySQL 服务器，那么可能还希望安装 Response Time Monitoring Agent 以收集 Ruby 应用程序的更多响应时间信息。您可能还要安装 Monitoring Agent for Windows OS 以监视系统的其他方面（例如，总体 CPU、内存和磁盘）。

您的产品确定哪些监视代理程序可用于安装。要获取每个产品中包含的代理程序列表，请参阅第 46 页的『功能』。

有关在 Windows 系统上运行的代理程序的列表，请参阅第 116 页的『Windows 系统上的安装前任务』。

开始之前

下载代理程序。请参阅第 86 页的『下载代理程序 以及数据收集器』。

查看第 69 页的『系统需求』中的信息，以确保满足您计划安装的代理程序的需求。

在安装代理程序之前复查代理程序先决条件任务。有关详细信息，请参阅第 116 页的『Windows 系统上的安装前任务』。

关于此任务

确保您拥有足够的权限来运行代理程序安装脚本和代理程序命令。您必须使用以下用户帐户类型之一进行登录：

· 缺省 Windows 管理员用户帐户

· 管理员用户帐户

· 属于管理员组的用户帐户

· 在 Active Directory 服务中注册为管理员的用户帐户

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Cloud Application Performance Management 代理程序。但是，这两种代理程序类型不能安装在同一目录中。有关代理程序共存的更多信息，请参阅第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

过程

请完成以下步骤以在安装了 Windows 操作系统的 VM 和系统上安装监视代理程序：

1. 在您的系统中，浏览至下载压缩文件所在的目录。
2. 将产品的代理程序安装文件解压缩到想要安装监视代理程序软件的位置。
   .bat 安装脚本解压缩到以归档文件和版本命名的目录。例如：*offering*_Agent_Install_8.1.4.0。代理程序二进制和配置相关文件解压缩至目录中的子目录。
3. 以管理员身份打开命令提示符。
   a) 在**开始**菜单中，在搜索框中输入 command。
   b) 右键单击所显示列表中的**命令提示符**并选择**以管理员身份运行**。
4. 在命令提示符中，使用管理员特权从以归档文件和版本命名的目录运行安装脚本。

   ```
   cd offering_Agent_Install_version
   installAPMAgents.bat
   ```

   限制: 对于 WebSphere Applications 代理程序，管理员特权必须与用于安装 WebSphere Application Server 的特权相同。

   要以静默方式安装代理程序，请参阅第 121 页的『以静默方式安装代理程序』。
5. 如果您正在安装代理程序，请提供安装目录的名称。

   缺省安装路径为 C:\IBM\APM。安装目录的名称不能超过 80 个字符，且不能包含非 ASCII 字符、特殊字符或双字节字符。路径中的目录名称只能包含以下字符：
   abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ _\:0123456789()~-./。

   注: 禁用短文件名创建 (*8dot3Name*) 后，如果路径中的目录名称包含空格，那么不支持安装。

   如果您正在升级代理程序，请跳过此步骤，代理程序会安装到先前的安装目录。
6. 在系统询问您是否接受许可协议时，输入 1 以接受协议并继续，或者输入 2 以拒绝。

   在输入 1（表示接受）后，将针对您的环境启动先决条件扫描，该扫描需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。缺少先决条件（诸如缺少库或磁盘空间不足）会停止安装。您必须解决故障，然后重新启动安装脚本。

   注: 如果安装退出并返回以下消息，请检查服务器服务是否已启动（启动 -> 管理工具 -> 服务）。如果未启动，那么再次启动服务器服务并运行 installAPMAgents.bat。

   ```
   此脚本 [installAPMAgents.bat] 必须以管理员身份运行。
   ```
7. 在完成安装并且命令提示符可用后，重复这些步骤以安装更多监视代理程序。

下一步做什么

根据需要配置代理程序。要检查监视代理程序是否需要手动配置，请参阅第 95 页的『第 5 章 代理程序 以及数据收集器部署』。有关配置指示信息，或者如果您想要复查缺省配置设置，请参阅第 135 页的『第 7 章 配置环境』。

安装新的代理程序之前，Windows 安装程序将临时停止所安装的产品位置中当前正在运行的所有代理程序。安装完成之后，安装程序会重新启动任何已经停止的代理程序。您必须手动重新启动安装程序没有自动启动的任何监视代理程序。

- 如果由于防火墙不允许与外部主机进行透明出站 HTTPS 连接而使用转发代理，那么必须编辑代理程序环境配置文件。有关指示信息，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。
- 如果您从先前版本升级代理程序，请了解在登录 Cloud APM 控制台之前必须完成的任何重新配置或迁移任务。有关这些任务的信息，请参阅第 961 页的『升级代理程序』。

使用以下某个方法启动此代理程序：

- 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Cloud Application Performance Management**。右键单击一个代理程序，然后单击**开始**。
- 运行以下命令

```
name-agent.bat start
```

有关监视代理程序命令的信息，包括要使用的名称，请参阅第 150 页的『使用代理程序命令』。有关哪些代理程序将会自动启动以及哪些需要手动启动的信息，请参阅 第 95 页的『第 5 章 代理程序 以及数据收集器部署』

升级之后，您必须重新启动安装程序不会自动配置并启动的任何代理程序。

在配置并启动代理程序之后，查看代理程序正在收集的数据。

- 如果您尚未登录，请按第 829 页的『启动 Cloud APM 控制台』中的指示信息执行操作。
- 如果想要在应用程序性能仪表板中查看来自 IBM Tivoli Monitoring 域的受管系统，请完成第 807 页的『与 IBM Tivoli Monitoring V6.3 集成 』中描述的任务。

以非 root 用户身份安装代理程序

如果您没有 root 用户特权并且想要安装监视代理程序，那么可以使用非 root 用户身份安装代理程序。另外，如果您是主机管理员并且不想以 root 用户身份运行监视代理程序，那么可以使用非 root 用户身份安装代理程序。安装流程与 root 用户相同。 在非 root 安装之后，请运行 **UpdateAutoRun.sh** 脚本，运行时请使用 root 用户身份或 sudo 用户访问权。

开始之前

要唯一地识别计算机系统，Linux OS 代理程序必须识别计算机系统主板的通用唯一标识 (UUID)、制造商、型号和序列号。需要此信息，才能在 Cloud APM 控制台中将代理程序添加到应用程序。

通过验证计算机系统上存在的以下实体来获取计算机系统信息：

1. 检查计算机系统上是否安装了 **/usr/bin/hal-get-property** 命，并确保 hald 进程（HAL 守护程序）正在运行。如果未安装该命令，请继续步骤 2。如果安装了该命令，请跳过步骤 2 和步骤 3。注：如果操作系统版本为 Red Hat 7，那么 hald 进程不可用。
2. 如果计算机系统上未安装 **/usr/bin/hal-get-property** 命令，请确认 `/sys/class/dmi/id/product_uuid` 文件存在且包含计算机系统 UUID，并且安装 Linux 操作系统代理程序的用户具有此文件的读访问权。如果此文件不存在，请继续至步骤 3。如果文件存在，请跳过步骤 3。
3. 如果未安装 **/usr/bin/hal-get-property** 命令并且 `/sys/class/dmi/id/product_uuid` 文件不存在，必须确保 **hostname** 或 **hostnamectl** 命令返回标准主机名。如果这些命令返回不含域的短主机名，那么必须通过输入 “**hostname <fqhn>**” 或 “**hostnamectl set-hostname <fqhn>**” 命令来设置标准主机名，其中，*<fqhn>* 必须替换为标准主机名。

注: Linux OS 代理程序将定期检索此信息，因此上述步骤中的命令或文件必须保留在原位置，即使在安装之后也一样。

注: 在以非 root 用户身份运行时，Linux 操作系统代理程序不支持监视 Docker。

过程

1. 如第 107 页的『在 Linux 系统上安装代理程序』和第 102 页的『在 UNIX 系统上安装代理程序』中所述，在 Linux 或 UNIX 系统上安装监视代理程序。
2. 可选： 如果以一个选中的用户身份安装了代理程序，并且想要以其他用户身份配置代理程序，请运行 `./secure.sh` 脚本。

   有关 **./secure.sh** 脚本的更多信息，请参阅 第 156 页的『以非 root 用户身份配置代理程序』和保护代理程序安装文件。

   例如： **./secure.sh -g db2iadm1**
3. 可选： 根据需要在 Linux 或 UNIX 上配置监视代理程序，请参阅第 135 页的『第 7 章 配置环境』。
4. To update the system startup scripts, run the following script （使用 root 用户身份或 sudo 用户访问权）： *install_dir*`/bin/UpdateAutoRun.sh`

下一步做什么

如果以非 root 用户身份安装了代理程序，并且要以同一用户身份配置该代理程序，无需执行特殊操作。如果以一个选中的用户身份安装了代理程序，并且想要以其他用户身份配置代理程序，请参阅第 156 页的『以非 root 用户身份配置代理程序』。

如果以非 root 用户身份安装并配置了代理程序，并且要以同一用户身份启动该代理程序，无需执行特殊操作。如果以一个选中的用户身份安装和配置了代理程序，并且想要以其他用户身份启动代理程序，请参阅第 859 页的『以非 root 用户身份启动代理程序』。

将相同的用户标识用于代理程序安装和升级。

如果以 root 用户身份运行 **UpdateAutoRun.sh** 脚本，代理程序被配置为在操作系统重新启动后自动启动。如果您不需要此代理程序行为，您可以禁用代理程序自动启动。有关更多信息，请参阅第 156 页的『在 UNIX 和 Linux 系统上禁用自动代理程序启动』。

保护代理程序安装文件

在 Linux 或 UNIX 系统上以非 root 用户身份安装监视代理程序后，可以运行 `secure.sh` 脚本以通过移除全局写许可权并设置正确的文件所有权来保护代理程序安装的安全。

开始之前

- 您必须具有安装目录的读取、写入和执行许可权。
- 必须在系统上完成了监视代理程序安装和所有代理程序配置，并且这些代理程序必须成功启动。
- 如果是以不同用户帐户运行代理程序，那么这些帐户必须属于同一个组。（请参阅 `-g` 选项。）

关于此任务

完成此步骤可锁定安装中的文件许可权。提供的选项有：无需提供 root 用户密码、指定组名和查看命令的帮助。

过程

- 从 *install_dir*`/bin` 目录运行以下命令。用法：

  ```
  secure.sh [-g common_group] [-n] [-h]
  ```

  - 在最简单的方式中，运行 **./secure.sh** 脚本，用于移除全局写许可权并将当前用户和该用户的组设置为文件所有者。如果脚本由非 root 用户运行，那么会提示该用户提供 root 用户密码。
  - 如果非 root 用户使用 `-n` 选项运行 **./secure.sh** 脚本，那么不会提示此用户提供 root 用户密码。在此情况下，更改文件许可权和更改所有权的操作将使用此用户的特权来完成。如果安装目录包含其他

用户拥有的文件，那么在当前用户不具有相应特权来修改其他用户的文件的许可权和所有权时，此方式会失败。

- 如果希望将特定组设置为组所有者，那么该所有者必须为 -g 选项提供有效组名作为该选项的参数。（请参阅示例。）
  运行 `secure.sh -g common_group`。
  该命令会以递归方式更改文件和目录的所有权。

  如果 *common_group* 组不是用户的主组，那么您可以运行下列命令，将 *common_group* 组设置为目录中创建的新文件的组所有者：

  ```
  chmod g+s install_dir/bin/sub_dir
  ```

  其中，*sub_dir* 是任何子目录，例如，`/opt/ibm/ccm/agent`。
- 使用 –h 选项运行 **./secure.sh** 脚本以获取脚本的帮助信息。

结果

安装目录只允许运行该脚本的用户进行访问，或者只允许指定组中的用户进行访问。

示例

如果用户 Alice 是名为“apmgroup”的系统组的成员，那么此用户可通过以下命令使用此组来设置文件组所有权：

```
./secure.sh -g apmgroup
```

在运行脚本后，对于组的 *install_dir* 中的所有文件，将组置为 “apmgroup”。

下一步做什么

运行 **./secure.sh** 脚本应该会导致为代理程序设置以下许可权。

```
rwx rwx ---
```

在运行该脚本之后，检查代理程序文件的许可权。例如，对于 Response Time Monitoring，检查 *install_dir*/*arch*/hu/lib/mod_wrt.so 中的文件。如果未正确设置这些文件的许可权，请手动更新许可权。例如，对于 Response Time Monitoring 代理程序：

1. 要设置许可权，请运行：

   ```
   chmod g+rx $AGENT_HOME/bin/rt-agent.sh
   ```
2. 要设置用户和组，请运行：

   ```
   chown newuser:newgroup $AGENT_HOME/bin/rt-agent.sh
   ```

以静默方式安装代理程序

以静默方式安装代理程序将减少安装时间。要以静默方式安装监视代理程序，必须从 IBM Marketplace Web 站点下载代理程序安装映像归档文件，解压缩代理程序安装文件，准备静默响应文件，并以静默方式运行安装脚本。

开始之前

1. 复审安装监视代理程序的先决条件任务，下载并解压缩代理程序安装文件。有关详细信息，请参阅在 UNIX 系统上安装代理程序、在 Linux 系统上安装代理程序或在 Windows 系统上安装代理程序。
2. 完成以下步骤来准备用于安装代理程序的静默响应文件：
   a. 找到您的产品的静默安装文件 *offering*_silent_install.txt，创建此文件副本，然后在文本编辑器中将其打开。

b. 取消注释许可协议。

c. 完成以下某个步骤来指定想要安装的代理程序：

· 取消注释要安装的每个代理程序。例如：

```
INSTALL_AGENT=os
```

```
INSTALL_AGENT=ruby
```

· 取消注释 INSTALL_AGENT=all，以安装所有代理程序。

d. 取消注释 AGENT_HOME，然后指定要安装代理程序的目录。

切记： Linux 如果是在 Linux 系统上升级代理程序，那么不得指定 /opt/ibm/apm/agent 目录。

e. Linux 如果您正在 Linux 系统上升级代理程序，请取消注释 MIGRATE_CONF=yes。

f. 保存该文件。

过程

1. 在命令行上切换到解压安装脚本的目录，然后运行以下命令：

```
cd offering_Agent_Install_version
```

2. 可选： 仅 Solaris 10 需要此步骤。在 Solaris 10 上运行安装脚本之前，必须创建指向 ksh 的软链接。

a) 备份 /bin/sh 命令：

```
mv /bin/sh /bin/sh.bkup_origin
```

b) 创建指向 ksh 命令的软链接：

```
ln -s /bin/ksh /bin/sh
```

c) 确认结果指向 ksh：

```
ls -l /bin/sh
```

3. 运行安装命令：

· Linux AIX

```
./installAPMAgents.sh -p path_to_silent_response_file
```

· Windows

```
installAPMAgents.bat -p path_to_silent_response_file
```

如果先决条件扫描程序无法获取将安装代理程序的位置的磁盘类型，那么代理程序将在 Windows 上安装失败。如果发生这种情况，那么您将在安装日志文件中看到 validDestLocation 属性的失败结果。在此情况下，您可以通过将 SKIP_PRECHECK=1 添加到安装命令，跳过先决条件扫描。例如：

```
installAPMAgents.bat -p path_to_silent_response_file SKIP_PRECHECK=1
```

注： Windows 禁用短文件名创建 (*8dot3Name*) 后，如果路径中的目录名称包含空格，那么不支持安装。

结果

已安装代理程序。

下一步做什么

配置代理程序。请参阅针对 Linux 和 UNIX 系统以及针对 Windows 系统的过程和命令表。

绕过先决条件扫描程序

在安装监视代理程序时，将针对您的环境启动先决条件扫描，该扫描需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。在某些安装场景中，您可能想要忽略警告消息或者完全绕过先决条件检查。

关于此任务

有两个级别的故障消息，即 WARN 和 FAIL，并且有两个级别的绕过：

· 设置 **IGNORE_PRECHECK_WARNING** 变量将导致安装程序忽略警告 (WARN) 消息。

· 设置 **SKIP_PRECHECK** 变量将导致安装程序忽略所有故障消息。

如果代理程序安装失败并且您收到来自必备软件检查程序的警告 (WARN)，请查看警告。如果想要继续安装，请设置 **IGNORE_PRECHECK_WARNING** 并重新安装。

在虚拟机映像用作模板的环境中，在安装开始之前执行的先决条件扫描可以只对第一个模板映像执行。如果一个 VM 映像通过扫描，那么根据该映像创建的其他 VM 也都通过。通过绕过对从同一映像创建的其他 VM 的先决条件检查，可以节省时间。设置 **SKIP_PRECHECK** 变量并重新安装。

SKIP_PRECHECK 设置也适合以下场景：您有 IBM 支持或软件产品兼容性报告指示受支持但尚未更新必备软件检查程序的新操作系统。请确保首先尝试安装代理程序，检查日志，并确保此新操作系统是发生故障的唯一项 - 也是绕过的唯一项 - 因为 **SKIP_PRECHECK** 会导致安装程序绕过先决条件核对表中的每个项。

下载并解压缩安装文件后，请完成此过程忽略警告消息或者绕过先决条件扫描。

过程

在计划安装监视代理程序的系统上，输入以下某个命令：

· 在先决条件检查期间忽略警告 (WARN) 消息：
  · Linux AIX `export IGNORE_PRECHECK_WARNING=1`
  · Windows `set IGNORE_PRECHECK_WARNING=1`
· 绕过先决条件扫描：
  · Linux AIX `export SKIP_PRECHECK=1`
  · Windows `set SKIP_PRECHECK=1`

下一步做什么

要在下一次使用先决条件扫描程序安装代理程序时复原缺省设置，请关闭 **IGNORE_PRECHECK_WARNING** 或 **SKIP_PRECHECK** 变量：

· Linux AIX `unset IGNORE_PRECHECK_WARNING`
· Windows `set IGNORE_PRECHECK_WARNING=`

或者

· Linux AIX `unset SKIP_PRECHECK`
· Windows `set SKIP_PRECHECK=`

卸载代理程序

从受管系统卸载单个代理程序或所有代理程序。

开始之前

对于多实例代理程序，必须先移除所有代理程序实例，然后再卸载该代理程序。否则，不会从注册表中清除代理程序条目。要移除实例，请运行以下命令：

- Windows *name*-agent.bat remove *instance_name*
- Linux AIX ./*name*-agent.sh remove *instance_name*

其中，*name* 是代理程序名称，*instance_name* 是实例名称。有关更多信息，请参阅第 150 页的『使用代理程序命令』。要获取多实例代理程序的列表，请参阅第 96 页的表 7。

对于以下代理程序，必须先完成特定于代理程序的任务，然后再完成卸载过程：

- 对于 Monitoring Agent for HTTP Server，必须先删除 http.conf 文件中的 Include 语句，例如 "Include "/opt/ibm/apm/agent/tmp/khu/kvm65s2_8044.conf"，然后再重新启动 IBM HTTP Server。
- 对于 Monitoring Agent for Python，运行 *install_dir*/lx8266/pg/bin/uninstall.sh 以移除注入代码，然后再卸载代理程序。
- 对于 Monitoring Agent for PHP，运行 *install_dir*/bin/lx8266/pj/lib/uninstall.*instance_name*.sh 以移动注入代码，然后再卸载代理程序。
- 对于 Monitoring Agent for WebSphere Applications，在卸载代理程序之前，必须为所有受监视的服务器实例取消配置数据收集器。遵循第 126 页的『WebSphere Applications 代理程序：取消配置数据收集器』中的指示信息。

  对于 WebSphere Applications 代理程序，请确保用来卸载代理程序的用户标识对数据收集器主目录中的 logs 和 runtime 目录，及其中所含的所有子目录及文件，具有完全的读写许可权。数据收集器主目录如下：

  - Windows *install_dir*\dchome\7.3.0.14.08
  - Linux AIX *install_dir*/yndchome/7.3.0.14.08
- 针对 Node.js 代理程序，您必须在卸载代理程序前从 Node.js 应用程序移除监视插件。遵循第 133 页的『Node.js 代理程序：移除监视插件』中的指示信息。
- 针对 Microsoft .NET 代理程序，您必须在卸载代理程序前从 .NET 应用程序中移除数据收集器。遵循第 133 页的『Microsoft .NET 代理程序：移除 .NET 数据收集器』中的指示信息。
- 对于 IBM Integration Bus 代理程序，如果使用提供了用户出口的代理程序为代理配置了事务跟踪，那么首先必须移除用户出口，然后才能卸载代理程序。遵循 中的指示信息第 250 页的『移除 KQIUserExit 用户出口』.
- 有关 Internet Service Monitoring，请转至 <candle_home>\BIN 并使用 uninstall 作为自变量运行 ism-agent.bat 文件。以防您要使用 smai-agent.bat 文件卸载服务器上的所有监视代理程序，请先将 uninstall 作为自变量运行 ism-agent.bat，然后运行 smai-agent.bat
- 对于 Monitoring Agent for SAP NetWeaver Java Stack，在卸载代理程序前，请使用以下命令停止所有 SAPNetWeaver Java Stack 代理程序实例：

  - Windows sap_netweaver_java_stack-agent.bat stop instance_name

关于此任务

Windows 系统上的 Oracle 代理程序只能使用命令提示符进行卸载。

过程

1. 在安装了监视代理程序的 VM 或系统上，启动命令行并切换至二进制目录：

- Linux AIX *install_dir*/bin
- Windows *install_dir*\BIN

其中，*install_dir* 是这一个或多个监视代理程序的安装目录。

2. 要卸载特定监视代理程序，请输入代理程序脚本名称和卸载选项，其中 *name* 是代理程序脚本名称：
   - Linux AIX ./*name*-agent.sh uninstall
   - Windows *name*-agent.bat uninstall

   有关代理程序脚本名称的列表，请参阅第 150 页的『使用代理程序命令』。

   切记: 针对 Monitoring Agent for Microsoft .NET，您必须使用管理员特权运行该命令。

   将从受管系统卸载监视代理程序。

   如果已单独卸载所有监视代理程序，那么请继续移除框架文件。请参阅后续步骤。
3. 要在出现确认提示时从受管系统卸载所有监视代理程序，请输入脚本名称并卸载所有选项：
   - Linux AIX ./smai-agent.sh uninstall_all
   - Windows smai-agent.bat uninstall_all

   将显示确认提示。输入 1 以继续，或输入 2 以取消。

   将从系统或 VM 卸载所有监视代理程序。
4. Linux AIX

   在 Linux 和 UNIX 上，要强制卸载所有监视代理程序，而不提示确认，请输入脚本名称和强制卸载所有选项：

   ```
   ./smai-agent.sh uninstall_all force
   ```

下一步做什么

对于 Monitoring Agent for HTTP Server，在卸载代理程序之后，您必须手动移除以下文件：

- /tmp/khu_cps.properties
- /tmp/httpserver-disc.error

对于 Monitoring Agent for Python：

1. 删除 Django pyc 配置文件，以确保复原的 Django pyc 文件生成自己的二进制文件。
2. 重新启动 Apache 服务器以移除在 Apache 进程中装入的中间件。

针对 Monitoring Agent for Ruby，要卸载诊断数据收集器：

1. 浏览到应用程序的主目录，打开其 Gemfile，并从该文件移除以下行：gem 'stacktracer'
2. 重新启动 Ruby on Rails 应用程序。
3. 卸载诊断数据收集器。输入：gem uninstall *Gemfile*
4. 移除数据收集器的运行时目录。此目录的缺省位置为 *install_dir*/install-images/kkm/dchome

针对 Monitoring Agent for Microsoft .NET，完成以下步骤：

1. 使用以下一个选项移除数据收集器 dll 文件：
   - 重新引导操作系统。
   - 尝试删除文件 *install_dir*\qe\bin64\CorProfLog.dll。

     将显示 File in Use 对话框。它标识了当前正在运行的 .NET 进程。
   - 重新启动每个 .NET 进程。
2. 重新启动 .NET 应用程序。

WebSphere Applications 代理程序：取消配置数据收集器

如果您在取消配置数据收集器之前卸载 WebSphere Applications 代理程序，那么代理程序卸载失败。您可以手动或通过使用交互实用程序或静默取消配置过程，从应用程序服务器实例移除数据收集器。

针对使用 PMI 资源监视进行监视的实例，取消配置不可用。服务器可用时，会继续监视 PMI 数据。

以交互方式取消配置数据收集器

如果不再希望数据收集器监视一个或多个应用程序服务器实例，那么可以为这些实例取消对数据收集器的配置。

开始之前

请使用用于配置数据收集器的用户标识来取消配置数据收集器，这也是用于安装应用程序服务器的用户标识。请验证此用户标识具有数据收集器主目录及其所有子目录的读写许可权。 数据收集器主目录如下所示，其中，*install_dir* 是 WebSphere Applications 代理程序安装目录。

- Windows *install_dir*\dchome\7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

关于此任务

取消配置实用程序（unconfig.sh 或 unconfig.bat）是用于取消配置数据收集器的菜单驱动的命令行实用程序。

过程

1. 以用来配置数据收集器的用户标识身份登录系统。
2. 浏览至以下 bin 目录：
   - Windows *install_dir*\dchome\7.3.0.14.08\bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
3. 可选： 启动实用程序前，设置 Java 主目录的位置。
   例如：

   Linux AIX
   ```
   export JAVA_HOME=/opt/IBM/AppServer80/java
   ```

   Windows
   ```
   set JAVA_HOME=C:\Progra~1\IBM\WebSphere\AppServer80\java
   ```
4. 通过发出以下命令启动取消配置实用程序：

   Linux AIX
   ```
   ./unconfig.sh
   ```

   Windows
   ```
   unconfig.bat
   ```
5. 实用程序搜索由数据收集器监视的所有服务器实例。输入与应用程序服务器实例一致的数字以为数据收集取消配置或者输入星号 (*) 来为所有应用程序服务器实例取消配置数据收集。要指定服务器的子集，请输入代表这些服务器的编号（由逗号分隔）。例如：1,2,3。

   切记:

   - 对于单机环境，必须在配置期间运行应用程序服务器实例。（不需要运行 WebSphere Application Server Liberty 实例）。
   - 针对 Network Deployment 环境，Node Agent 和 Deployment Manager 必须正在运行。
6. 实用程序提示您指定是否希望创建当前 WebSphere Application Server 配置的备份。输入 1 以创建当前配置的备份。否则，请输入 2 并跳至步骤 8。
7. 实用程序提示您指定用于存储配置备份的目录。指定用于存储配置备份的目录或者接受缺省目录。
   实用程序显示 WebSphere 主目录的名称以及创建了备份的 WebSphere 概要文件。
8. 该实用程序指示是否为您指定的 WebSphere Application 概要文件启用了 WebSphere 全局安全性。如果未启用全局安全性，请跳至步骤 10。

9. 该实用程序提示您指定是否从客户机属性文件检索安全设置。输入 1 以使实用程序从相应的客户机属性文件检索用户名和密码并跳至步骤 第 127 页的『10』。否则，输入 2 以输入用户名和密码。

   数据收集器使用 RMI 或 SOAP 协议与 WebSphere Administrative Services 进行通信。如果为概要文件启用了全局安全性，那么必须指定有权登录到概要文件的 IBM WebSphere Application Server 管理控制台的用户的用户标识和密码。或者，您可以在配置数据收集器之前对用户名和密码进行加密，并将其存储在客户机属性文件中。必须将 `sas.client.props` 文件用于 RMI 连接，或将 `soap.client.props` 文件用于 SOAP 连接。

   如果选择了备份当前 WebSphere 配置的选项，那么实用程序会开始备份该配置。
10. 实用程序将针对指定的应用程序服务器实例取消配置数据收集器。会显示状态消息，指示数据收集器已成功取消配置。
11. 数据收集器取消配置完成之后，重新启动应用程序服务器实例。

    重新启动应用程序服务器实例之后，数据收集器配置会生效。服务器实例的 PMI 资源监视仍可用。
12. 可选： 如果在取消配置数据收集器后希望将资源监视用于某个服务器实例，请通过运行以下命令来重新启动监视代理程序：

    - Windows

      ```
      cd install_dir\bin
      was-agent.bat stop
      was-agent.bat start
      ```

    - Linux AIX

      ```
      cd install_dir/bin
      ./was-agent.sh
      stop
      ./was-agent.sh start
      ```

结果

已针对指定的应用程序服务器实例取消配置数据收集器。

以静默方式取消配置数据收集器

您可以使用取消配置实用程序以静默方式取消配置数据收集器。

开始之前

请使用用于配置数据收集器的用户标识来取消配置数据收集器，这也是用于安装应用程序服务器的用户标识。请验证此用户标识具有数据收集器主目录及其所有子目录的读写许可权。 数据收集器主目录如下所示，其中，*install_dir* 是 WebSphere Applications 代理程序安装目录。

- Windows *install_dir*\dchome\7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

关于此任务

如果您以静默方式取消配置数据收集器，那么首先在属性文件中指定配置选项。随取消配置实用程序一起打包了样本属性文件 `sample_silent_unconfig.txt`。此文件位于数据收集器主目录中的 `bin` 目录。

过程

1. 以用来配置数据收集器的用户标识身份登录系统。
2. 在 `properties.txt` 文件中指定配置选项。

   您可使用下列属性，以静默方式取消配置数据收集器：

   WebSphere Application Server 连接设置

   was.wsadmin.connection.host
   指定 wsadmin 工具正在连接到的主机名。

WebSphere Application Server 全局安全性设置

was.wsadmin.username
指定有权登录 IBM WebSphere Application Server 管理控制台的用户的用户标识。此用户在应用程序服务器上必须具有代理角色。

was.wsadmin.password
指定与在 `was.wsadmin.username` 属性中指定的用户对应的密码。

WebSphere Application Server 设置

was.appserver.profile.name
指定希望取消配置的应用程序服务器概要文件的名称。

was.appserver.home
指定 WebSphere Application Server 主目录。

was.appserver.cell.name
指定 WebSphere Application Server 单元名称。

was.appserver.node.name
指定 WebSphere Application Server 节点名。

WebSphere Application Server 配置的备份

was.backup.configuration
指定是否在取消数据收集器之前对 WebSphere Application Server 数据收集器的当前配置进行备份。有效值为 `True` 和 `False`。

was.backup.configuration.dir
指定备份目录的位置。

WebSphere Application Server 运行时实例设置

was.appserver.server.name
在希望为其取消配置的应用程序服务器概要文件内指定应用程序服务器实例。

提示: 静默响应文件可具有该属性的多个实例。

3. 浏览至以下目录:
   - Windows *install_dir*\dchome\7.3.0.14.08\bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
4. 运行以下命令:
   - Windows

     ```
     unconfig.bat -silent path_to_silent_file
     ```

   - Linux AIX

     ```
     unconfig.sh -silent path_to_silent_file
     ```

5. 数据收集器取消配置完成之后，重新启动应用程序服务器实例。
   重新启动应用程序服务器实例之后，数据收集器配置会生效。服务器实例的 PMI 资源监视仍可用。
6. 可选： 如果在取消配置数据收集器后希望将资源监视用于某个服务器实例，请通过运行以下命令来重新启动监视代理程序:
   - Windows

     ```
     cd install_dir\bin
     was-agent.bat stop
     was-agent.bat start
     ```

   - Linux AIX

     ```
     cd install_dir/bin
     ./was-agent.sh
     ```

```
stop
./was-agent.sh start
```

从应用程序服务器实例手动移除数据收集器配置

要从应用程序服务器实例手动移除数据收集器配置，必须可以通过使用 wsadmin 工具连接到该应用程序服务器。仅当正使用 WebSphere Application Server Network Deployment 且 Deployment Manager 正在运行时，才可执行此操作。如果 WebSphere Application Server 无法启动，那么必须从运行配置实用程序时进行的备份复原 WebSphere Application Server。

关于此任务

仅当以下任一条件适用时，才可从应用程序服务器实例手动移除数据收集器配置：

- 在非 Network Deployment 环境中，已将数据收集器配置手动添加到应用程序服务器实例，且希望取消配置数据收集。应用程序服务器实例必须正在运行。
- 在 Network Deployment 环境中，已将数据收集器配置手动添加到应用程序服务器实例，且希望取消配置数据收集。应用程序服务器上的 Node Agent 和 Deployment Manager 必须正在运行。
- 在 Network Deployment 环境中，已为数据收集手动配置应用程序服务器实例，且应用程序服务器启动失败。应用程序服务器上的 Node Agent 和 Deployment Manager 必须正在运行。

如果已手动或者通过配置或迁移实用程序为数据收集配置独立应用程序服务器实例，且应用程序服务器启动失败，那么必须使用备份配置复原 WebSphere Application Server 配置。有关更多信息，请参阅第 747 页的『从备份复原应用程序服务器配置』。

切记：

- 必须以 WebSphere 管理用户身份手动更改 WebSphere Application Server 的数据收集器配置。
- 对用于数据收集的 WebSphere Application Server 进行手动更改必须仅由经验丰富的 WebSphere 管理员来执行。手动配置更改中的错误可能导致应用程序服务器无法启动。
- 如果手动配置数据收集器以监视应用程序服务器实例，那么无法使用取消配置实用程序来取消配置数据收集器。

过程

要手动移除数据收集器配置，请完成下列过程：

1. 登录 WebSphere Administration Server Console。
2. 单击**服务器**。
3. 展开**服务器类型**并选择 **WebSphere 应用程序服务器**。
4. 单击服务器的名称。
5. 在“配置”选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **Java 虚拟机** > **其他属性**：**定制属性**。
6. 移除存在的以下任一 JVM 定制属性：
   - am.home
   - ITCAM.DC.ENABLED
   - TEMAGCCollector.gclog.path
   - com.ibm.tivoli.itcam.toolkit.ai.runtimebuilder.enable.rebuild
   - com.ibm.tivoli.jiti.injector.ProbeInjectorManagerChain primaryInjectorFile
7. 识别为数据收集器添加的 JVM 参数。
   a) 在导航窗格中，单击**环境** > **WebSphere 变量**。
   b) 如果为数据收集手动配置了应用程序服务器，请找到手动添加的 JVM 参数。
   如果使用配置实用程序为数据收集配置了应用程序服务器，请将 **AM_OLD_ARGS** 参数的值与 **AM_CONFIG_JVM_ARGS** 参数的值进行比较，以确定配置实用程序所添加的参数。
8. 单击**服务器** > **应用程序服务器**，并选择相应服务器名称。

9. 在“配置”选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **Java 虚拟机**。
10. 在**通用 JVM 参数**字段中，移除在步骤 7 针对数据收集器识别的 JVM 参数。
11. 单击**应用**或**确定**。
12. 在**消息**对话框中，单击**保存**。
13. 在**保存到主配置**对话框中，完成以下其中一个步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**，然后单击**保存**。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
14. 移除为数据收集器添加的环境条目。
    a) 在“配置”选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **环境条目**。
    b) 根据操作系统，删除以下环境条目：
       - UNIX **LIBPATH**
       - Linux **LD_LIBRARY_PATH**
       - Windows **PATH**
    c) 移除 **NLSPATH** 环境条目。
15. 单击**应用**或**确定**。
16. 在**消息**对话框中，单击**保存**。
17. 在**保存到主配置**对话框中，完成以下其中一个步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**复选框，然后单击**保存**。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
18. 在导航窗格中，单击**环境** > **WebSphere 变量**。
19. 删除下列变量：
    - **AM_CONFIG_JVM_ARGS**
    - **AM_OLD_JVM_ARGS**
    - **ITCAMDCHOME**
    - **ITCAMDCVERSION**
20. 在**消息**对话框中，单击**保存**。
21. 在**保存到主配置**对话框中，完成以下其中一个步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**复选框，然后单击**保存**。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
22. 如果使用数据收集器配置工具而不是通过手动方式为数据收集配置了服务器实例，请完成以下步骤：
    a) 浏览至 *dc_home*/runtime 目录。
    b) 将 $profile.$cell.$node.$server.input.properties 文件重命名为 $profile.$cell.$node.$server.input.properties.bak。
23. 如果要从概要文件中所有应用程序服务器实例手动移除数据收集器配置，请执行以下步骤：
    a) 浏览至 $appserverhome/bin 目录。
    b) 在 Windows 系统上运行 **osgiCfgInit.sh/bat -all** 命令，或在 UNIX 和 Linux 系统上运行 **osgiCfgInit.sh -all** 命令。
24. 重新启动数据收集器所监视的应用程序服务器实例。

手动取消配置数据收集器

在为 WebSphere Applications 代理程序手动配置数据收集器之后，如果要移除所配置应用程序服务器中的数据收集，您必须手动取消配置数据收集器。

关于此任务

仅当您按第 727 页的『在配置实用程序失败时手动配置数据收集器』中的指示信息手动配置数据收集器之后，以下过程才适用。如果已使用配置实用程序来配置数据收集器，那么也必须使用取消配置实用程序来取消配置数据收集器。有关指示信息，请参阅第 126 页的『以交互方式取消配置数据收集器』或第 127 页的『以静默方式取消配置数据收集器』。

过程

- 要为 WebSphere Application Server 手动取消配置数据收集器，请参阅第 131 页的『为传统 WebSphere Application Server 手动取消配置数据收集器』。
- 要为 Liberty 服务器手动取消配置数据收集器，请参阅第 132 页的『为 WebSphere Application Server Liberty 手动取消配置数据收集器』。

为传统 *WebSphere Application Server* 手动取消配置数据收集器

过程

1. 以管理员身份登录 WebSphere 管理控制台。
2. 在导航窗格中，单击**服务器**，展开**服务器类型**，然后选择 **WebSphere Application Server**。
3. 单击应用程序服务器的名称。
4. 在“配置”选项卡中的**服务器基础结构**部分下，展开 **Java 虚拟机**，并单击**进程定义**。
5. 在**其他属性**部分中，单击 **Java 虚拟机**。
6. 在**通用 JVM 参数**字段中，从内容中移除下列条目。

```
-agentlib:am_ibm_16=${WAS_SERVER_NAME} -Xbootclasspath/p:${ITCAMDCHOME}/
toolkit/lib/bcm-bootstrap.jar -Djava.security.policy=${ITCAMDCHOME}/itcamdc/
etc/datacollector.policy -verbosegc
```

7. 单击**应用**，然后单击**保存**。在“保存到主配置”对话框中，完成下列步骤：
   - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**，然后单击**保存**。
   - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
8. 在导航窗格中，单击**服务器**，展开**服务器类型**，单击 **WebSphere Application Server**，然后单击服务器名称。
9. 在“配置”选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **环境条目**。
10. 根据操作系统、硬件平台和应用程序服务器 JVM，移除以下环境条目。
    - UNIX LIBPATH
    - Linux LD_LIBRARY_PATH
    - Windows PATH
11. 在导航窗格中，单击**环境** > **WebSphere 变量**。
12. 移除 *ITCAMDCHOME* 变量（如果存在）。
13. 单击**应用**，然后单击**保存**。在“保存到主配置”对话框中，完成下列步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**，然后单击**保存**。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
14. 重新启动应用程序服务器实例。
15. 转到代理程序安装目录中的 `runtime` 目录，并移除 *`profile_name.cell_name.node_name.server_name`*`.manual.input.properties` 文件。

- Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime/*profile_name.cell_name.node_name.server_name*.manual.input.properties
- Windows *install_dir*\dchome\7.3.0.14.08\runtime*profile_name.cell_name.node_name.server_name*.manual.input.properties

为 *WebSphere Application Server Liberty* 手动取消配置数据收集器

过程

1. 导航至 Liberty 服务器目录，并打开 Liberty 服务器安装目录下 *server_name* 目录中的 jvm.options 文件。例如，/opt/ibm/wlp/usr/servers/defaultServer。
2. 从 jvm.options 文件中移除下列参数。

```
-agentlib:am_ibm_16=server_name
-Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
-verbosegc
```

   其中，*server_name* 是 Liberty 服务器的名称；*dc_home* 是数据收集器主目录。
3. 打开 server.xml 文件，并移除下列各行：

```
<feature>webProfile-7.0</feature>
<feature>monitor-1.0</feature>
<feature>usr:itcam-730.140</feature>
```

4. 打开 server.env 文件，并根据操作系统，从环境条目中移除下列条目值：

表 *8.* 环境条目

平台	环境条目名称	环境条目值
AIX R6.1（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
AIX R7.1（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
Solaris 10（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Solaris 11（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Linux x86_64 R2.6（64 位 JVM）	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/lx8266
Linux Intel R2.6（32 位 JVM）	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/li6263
Windows（32 位 JVM）	PATH	/lib;*dc_home*/toolkit/lib/win32
Windows（64 位 JVM）	PATH	/lib;*dc_home*/toolkit/lib/win64

5. 重新启动 Liberty 服务器。
6. 转到 WebSphere Applications 代理程序安装目录中的 runtime 目录，并移除 *cell_name.node_name.server_name*.manual.input.properties 文件。

- Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime/*cell_name.node_name.server_name*.manual.input.properties
- Windows *install_dir*\dchome\7.3.0.14.08\runtime*cell_name.node_name.server_name*.manual.input.properties

Node.js 代理程序：移除监视插件

卸载 Node.js 代理程序 之前，必须从 Node.js 应用程序中移除监视插件。

过程

1. 从 Node.js 应用程序文件开头移除数据收集器插件。
   - 如果将 Node.js 代理程序从 V01.00.12.00 升级到 V01.00.13.00，请完成下列过程：
     - 如果已启用资源数据收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_index.js');
       ```

       其中，*KNJ_NPM_LIB_LOCATION* 为 npm 包全局安装目录的 lib 文件夹目录。缺省目录为 /usr/local/lib。
     - 如果已启用资源数据收集和深入诊断数据收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_deepdive.js');
       ```

     - 如果已启用资源数据收集、深入诊断数据收集和方法跟踪收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_methodtrace.js');
       ```

   - 如果将 Node.js 代理程序从 V01.00.10.00 升级到 V01.00.13.00，请完成下列过程：
     - 如果已启用资源数据收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_index.js');
       ```

       ，其中，*install_dir* 是 Node.js 代理程序的安装目录。
     - 如果已启用资源数据收集和深入诊断数据收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_deepdive.js');
       ```

     - 如果已启用资源数据收集、深入诊断数据收集和方法跟踪收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_methodtrace.js');
       ```

2. 重新启动 Node.js 应用程序以禁用数据收集器插件。
   - 如果当前 Node.js 代理程序版本为 V01.00.10.00，那么到目前为止，已成功移除数据收集器插件。
   - 如果当前 Node.js 代理程序版本为 V01.00.12.00，那么继续以下步骤。
3. 从 *install_dir*/lx8266/nj/bin 目录运行 ./uninstall.sh 命令以移除先前的代理程序设置。

下一步做什么

有关卸载 Node.js 代理程序 的更多信息，请参阅第 124 页的『卸载代理程序』。

Microsoft .NET 代理程序：移除 .NET 数据收集器

在卸载 Microsoft .NET 代理程序 之前，您必须从 .NET 应用程序移除 .NET 数据收集器。

过程

1. 注销数据收集器。

作为管理员，输入：

```
cd install_dir\qe\bin configdc unregisterdc
```

其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。

2. 停止所有 .NET 应用程序来禁用数据收集器。
   输入 `net stop was /y`
3. 要确保在卸载后完全清除 .NET Data Collector，请执行以下步骤：
   a) 在命令提示符中，转至 `<APM_HOME>\qe\bin` 目录。
   b) 运行 `ProcListCaller.bat` 文件。
   c) 验证 `<APM_HOME>\qe\logs` 目录中的 `CorProfAttach.Log` 日志文件。该日志文件列出了必须将 .NET DC 概要分析程序组件附加到的过程。
   d) 在卸载代理程序前，请从 `CorProfAttach.Log` 文件停止这些过程。
   e) 如果未列出过程，那么继续执行代理程序卸载。

下一步做什么

卸载 Microsoft .NET 代理程序。请参阅第 124 页的『卸载代理程序』。

第 7 章 配置环境

如果监视代理程序需要配置或者您想要查看代理程序的缺省设置，请执行为代理程序提供的步骤。

通用主题

在配置代理程序和数据收集器时，某些主题是通用的。

网络连接

要确保建立代理程序服务器通信，请测试系统与 Cloud APM 服务器的连接。

要验证通信，请测试与 Cloud APM 数据中心的连接。要确保防火墙规则允许从三个特定 IP 地址和端口 443 返回流量，请找到验证连接所需的三个数据中心 IP 地址。有关更多信息，请参阅 Application Performance Management Developer Center 中的 Data Center IP addresses (SaaS only)。通过使用 **openssl** 命令，确认代理程序可以连接到这三个 IP 地址。有关使用 **openssl** 命令的更多信息，请参阅配置代理程序以通过转发代理进行通信。如果代理程序无法进行连接，请与您当地的 IT 团队联系。他们可以调整防火墙规则、启用端口 443、启用来自服务器的 TLS 1.2 流量，或者配置代理服务器以连接到 Cloud APM 服务器。

如果防火墙规则不允许与外部主机建立透明出站 HTTPS 连接，您可以配置代理程序，以将流量发送到转发代理。有关更多信息，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。

浏览器连接

要检查浏览器与 Cloud APM 控制台的连接，请找到在供应预订时 IBM 向您提供的**启动** URL。您还可以登录帐户，然后启动控制台。使用 IBM Marketplace 预定详细信息登录到 **“产品和服务”**(http://ibm.biz/my-prodsvcs) 页面。单击**启动**以启动控制台并查看 URL，例如：`8b68ba1b9.agents.na.apm.ibmserviceengage.com`。验证您是否可以使用此 URL 登录到控制台。

安全通信

代理程序与 Cloud APM 服务器之间的安全通信需要 TLS 1.2。

代理程序与 Cloud APM 服务器在 IBM Cloud 中使用 HTTPS 进行通信时，需要 TLS 1.2 和 FIPS Suite-B 密码套件。使用以下缺省密码：

```
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
```

浏览器与 Cloud APM 服务器之间的通信也需要 TLS 1.2。在某些浏览器中，缺省情况下未启用 TLS 1.2，必须手动启用。

配置代理程序以通过转发代理进行通信

如果防火墙规则不允许与外部主机的透明出站 HTTPS 连接，那么您可以配置 IBM 监视代理程序，以将流量发送到转发代理。编辑 KDH_FORWARDPROXY 环境变量以配置代理程序通过转发代理进行通信。

开始之前

要确定代理程序连接到的 Cloud APM 数据中心的 IP 地址，请参阅 Data Center IP addresses (APM Developer Center)。然后，调整防火墙规则，以允许要从转发代理发送到这些 IP 地址的请求。

可以使用 **openssl** 命令来检查安装了代理程序的计算机系统是否已连接到 Cloud APM 数据中心服务器。另外，还可以使用 **openssl** 命令来检查网络是否支持 Cloud APM 所使用的密码套件。如果 **openssl** 命令结果指出无法连接计算机系统，那么您可能需要设置转发代理。如果命令结果指出无法获取 Cloud APM 服务器证书，请与网络团队一起确定不支持所需密码套件的原因。要获取 Cloud APM 所使用的密码套件的列表，请参阅第 135 页的『安全通信』。

运行 **openssl** 命令，如以下示例所示：

```
echo quit | openssl s_client
```

© Copyright IBM Corp. 2014, 2019

```
-state -connect <domain-name>:443
-tls1_2 -cipher
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384
```
其中，*domain-name* 是 Cloud APM 预订的域名（例如：8b68ba1b9.agents.na.apm.ibmserviceengage.com）。

要确定预订的域名，请完成下列步骤：

1. 在文本编辑器中打开代理程序环境配置文件：

   Linux AIX /opt/ibm/apm/agent/config/global.environment

   Windows install_dir\TMAITM6_x64\K*pc*ENV（针对 Windows 64 位系统）和 install_dir\TMAITM6\K*pc*ENV（针对 Windows 32 位系统），其中，*pc* 是代理程序产品代码。

   有关产品代码的列表，请参阅第 150 页的『使用代理程序命令』。
2. 查找 *IRA_ASF_SERVER_URL* 变量。值的形式：https://domain-name/ccm/asf/request。通过 **openssl** 命令使用值的域名部分。

如果连接成功，那么会显示类似于以下示例的消息：
```
CONNECTED(00000003)
SSL_connect:before/connect initialization
SSL_connect:SSLv3 write client hello A
SSL_connect:SSLv3 read server hello A
depth=2 C = US, O = IBM Service Engage,
CN = ca_ec_384.ibmserviceengage.com
verify error:num=19:self signed certificate in certificate chain
verify return:0
SSL_connect:SSLv3 read server certificate A
SSL_connect:SSLv3 read server key exchange A
SSL_connect:SSLv3 read server certificate request A
SSL_connect:SSLv3 read server done A
SSL_connect:SSLv3 write client certificate A
SSL_connect:SSLv3 write client key exchange A
SSL_connect:SSLv3 write change cipher spec A
SSL_connect:SSLv3 write finished A
SSL_connect:SSLv3 flush data
SSL_connect:SSLv3 read finished A
---
Certificate chain
0 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN =ca_ec_384.apm.ibmserviceengage.com
1 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
2 s:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICkjCCAhegAwIBAgIIXlr284nLPaMwDAYIKoZIzj0EAwMFADCBhDELMAkGA1UE
BgwCVVMxGzAZBgNVBAoMEklCTSBTZXJ2aWNlIEVuZ2FnZTErMCkGA1UECwwiQXBw
bGljYXRpb24gUGVyZm9ybWFuY2UgTWFuYWdlbWVudDErMCkGA1UEAwwiY2FfZWNf
Mzg0LmFwbS5pYm1zZXJ2aWNlZW5nYWdlLmNvbTAeFw0xMzEyMDIxNjM2MDlaFw0y
MzEyMDExNjM2MDlaMIGGMQswCQYDVQQGDAJVUzEbMBkGA1UECgwSSUJNIFNlcnZp
Y2UgRW5nYWdlMSswKQYDVQQLDCJBcHBsaWNhdGlvbiBQZXJmb3JtYW5jZSBNYW5h
Z2VtZW50MS0wKwYDVQQDDCQqLmFnZW50cy5uYS5hcG0uaWJtc2VydmljZWVuZ2Fn
```

```
ZS5jb20wdjAQBgcqhkjOPQIBBgUrgQQAIgNiAAQmrGoCkAMoNAC3F6MIo1zR8fcO
mczYXtUux2bhlOibn3jQdxamhDR91nr2RBerGjMIITKNXd2MaOr3b6m8euk1BAL3
KsbN9lqvw94kXg0BTO1IHAcdsZQB+AuEVVhmDVGjUDBOMAwGA1UdEwEB/wQCMAAw
HwYDVR0jBBgwFoAU/zpE5TOnQ8LSuvbSWRfpbiGea08wHQYDVR0OBBYEFHL0At4O
GUdcOHVGg4Tfo4hl7LLGMAwGCCqGSM49BAMDBQADZwAwZAIwDWPHo5I04ZFVrkfk
St6gwH2UNF37jBscRN1lOE4SIwezZAqVs42BNMkWRjJBgiHzAjBm4m3zOjsXzNL8
+u8ALjQQCpBDT6dUHujzY5CRxG0xEHi5IXsXf4QwbctnjjvTeYA=
-----END CERTIFICATE-----
subject=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
issuer=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
---
Acceptable client certificate CA names
/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
/C=US/O=DigiCert Inc/OU=www.digicert.com/CN=DigiCert
Global Root CA/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
Server Temp Key: ECDH, prime256v1, 256 bits
---
SSL handshake has read 2659 bytes and written 261 bytes
---
New, TLSv1/SSLv3, Cipher is ECDHE-ECDSA-AES128-GCM-SHA256
Server public key is 384 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
Protocol : TLSv1.2
Cipher : ECDHE-ECDSA-AES128-GCM-SHA256
Session-ID:
A18C31D0B45A1166357C917E1CFCD86A9FBEDB4A0EB768EF5390AC28C95CB7EF
Session-ID-ctx:
Master-Key:
252B8FE2731E51AC0B79A27C7BED33CA8B15AF4CFD015C98DBACA46EA01DC40B
9E6B56E62E0F332FF6B56266B5ADD7B0
Key-Arg : None
Krb5 Principal: None
PSK identity: None
PSK identity hint: None
Start Time: 1510772474
Timeout : 7200 (sec)
Verify return code: 19 (self signed certificate in certificate chain)
---
DONE
SSL3 alert write:warning:close notify
```

如果计算机系统无法连接到 Cloud APM 服务器，那么会显示类似于以下示例的消息：

```
getaddrinfo: Name or service not known
connect:errno=2
```

如果因为密码套件在网络中的某个位置遭阻止，导致计算机系统无法获取服务器证书，那么会显示类似于以下示例的消息：

```
SSL_connect:failed
---
no peer certificate available
---
```

```
No client certificate CA names sent
```

关于此任务

在使用转发代理时，代理程序首先打开与代理的 TCP 连接。代理程序会将 HTTP CONNECT 请求和目标端点（Cloud APM 服务器）URL 发送到转发代理。然后，转发代理将与目标端点建立 TCP 连接，并在代理程序与 Cloud APM 服务器之间设置 HTTPS 隧道会话。

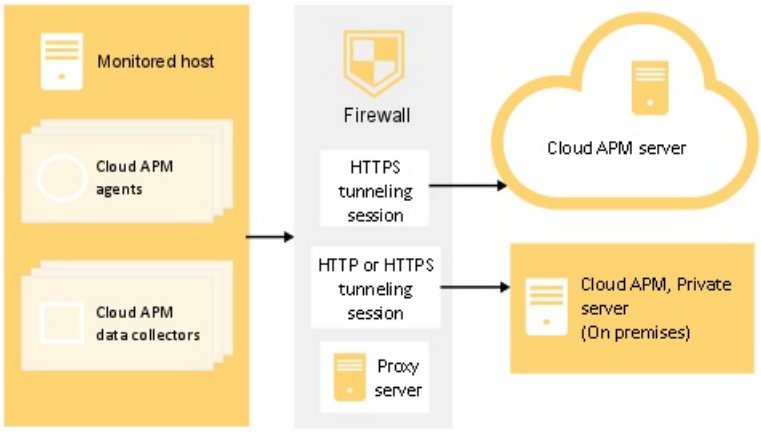


图 *1.* 使用转发代理的连接图

监视代理程序不支持认证代理，这表示代理程序不支持使用配置的代理用户标识和密码登录到转发代理。

过程

1. 在文本编辑器中打开代理程序环境配置文件：

   Linux AIX *install_dir*/config/global.environment 文件，其中，*install_dir* 是代理程序的安装主目录。global.environment 文件在安装目录中配置所有代理程序。

   .global.environment 文件中的定制设置在代理程序升级后丢失。要保留设置，请在 global.environment 文件中进行定制更改。代理程序升级不会覆盖此文件中的设置。

   Windows *install_dir*\TMAITM6_x64\K*pc*ENV（针对 64 位代理程序）和 *install_dir*\TMAITM6\K*pc*ENV（针对 32 位代理程序），其中，*pc* 是代理程序产品代码。配置每个代理程序的 K*pc*ENV 文件。

   有关产品代码的列表，请参阅第 150 页的『使用代理程序命令』。
2. 编辑 KDH_FORWARDPROXY 环境变量来指定代理地址和端口：

   ```
   KDH_FORWARDPROXY=http://proxy-address:proxy-port-number
   ```

   例如：

   ```
   KDH_FORWARDPROXY=http://HostA:8085
   ```
3. 重新启动代理程序以实施更改。请参阅第 150 页的『使用代理程序命令』。

配置数据收集器以通过转发代理进行通信

如果防火墙规则不允许与外部主机建立透明出站 HTTPS 连接，您可以配置外部主机，以将流量发送到转发代理。编辑 APM_GW_PROXY_CONNECTION 环境变量以配置数据收集器通过转发代理进行通信。

开始之前

要确定数据收集器连接到的 Cloud APM 数据中心的 IP 地址，请参阅 Data Center IP addresses (APM Developer Center)。然后，调整防火墙规则，以允许要从转发代理发送到这些 IP 地址的请求。

可以使用 **openssl** 命令来检查安装了数据收集器的计算机系统是否已连接到 Cloud APM 数据中心服务器。另外，还可以检查网络是否支持 Cloud APM 所使用的密码套件。如果 **openssl** 命令结果指出无法连接计算机系统，那么您可能需要设置转发代理。如果命令结果指出无法获取 Cloud APM 服务器 证书，请与网络团队一起确定不支持所需密码套件的原因。要获取 Cloud APM 所使用的密码套件的列表，请参阅第 135 页的『安全通信』。

运行 **openssl**，如以下示例所示：

```
echo quit | openssl s_client
-state -connect <domain-name>:443
-tls1_2 -cipher
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384
```

其中，*domain-name* 是 Cloud APM 订阅的域名.

要确定订阅的域名，请参阅第 135 页的『配置代理程序以通过转发代理进行通信』。

如果连接成功，那么会显示类似于以下示例的消息：

```
CONNECTED(00000003)
SSL_connect:before/connect initialization
SSL_connect:SSLv3 write client hello A
SSL_connect:SSLv3 read server hello A
depth=2 C = US, O = IBM Service Engage,
CN = ca_ec_384.ibmserviceengage.com
verify error:num=19:self signed certificate in certificate chain
verify return:0
SSL_connect:SSLv3 read server certificate A
SSL_connect:SSLv3 read server key exchange A
SSL_connect:SSLv3 read server certificate request A
SSL_connect:SSLv3 read server done A
SSL_connect:SSLv3 write client certificate A
SSL_connect:SSLv3 write client key exchange A
SSL_connect:SSLv3 write change cipher spec A
SSL_connect:SSLv3 write finished A
SSL_connect:SSLv3 flush data
SSL_connect:SSLv3 read finished A
---
Certificate chain
0 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN =ca_ec_384.apm.ibmserviceengage.com
1 s:/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
2 s:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
i:/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICkjCCAhegAwIBAgIIXlr284nLPaMwDAYIKoZIzj0EAwMFADCBhDELMAkGA1UE
BgwCVVMxGzAZBgNVBAoMEklCTSBTZXJ2aWNlIEVuZ2FnZTErMCkGA1UECwwiQXBw
bGljYXRpb24gUGVyZm9ybWFuY2UgTWFuYWdlbWVudDErMCkGA1UEAwwiY2FfZWNf
Mzg0LmFwbS5pYm1zZXJ2aWNlZW5nYWdlLmNvbTAeFw0xMzEyMDIxNjM2MDlaFw0y
MzEyMDExNjM2MDlaMIGGMQswCQYDVQQGDAJVUzEbMBkGA1UECgwSSUJNIFNlcnZp
Y2UgRW5nYWdlMSswKQYDVQQLDCJBcHBsaWNhdGlvbiBQZXJmb3JtYW5jZSBNYW5h
Z2VtZW50MS0wKwYDVQQDDCQqLmFnZW50cy5uYS5hcG0uaWJtc2VydmljZWVuZ2Fn
ZS5jb20wdjAQBgcqhkjOPQIBBgUrgQQAIgNiAAQmrGoCkAMoNAC3F6MIo1zR8fcO
mczYXtUux2bhlOibn3jQdxamhDR91nr2RBerGjMIITKNXd2MaOr3b6m8euk1BAL3
KsbN9lqvw94kXg0BTO1IHAcdsZQB+AuEVVhmDVGjUDBOMAwGA1UdEwEB/wQCMAAw
HwYDVR0jBBgwFoAU/zpE5TOnQ8LSuvbSWRfpbiGea08wHQYDVR0OBBYEFHL0At4O
```

```
GUdcOHVGg4Tfo4hl7LLGMAwGCCqGSM49BAMDBQADZwAwZAIwDWPHo5I04ZFVrkfk
St6gwH2UNF37jBscRN1lOE4SIwezZAqVs42BNMkWRjJBgiHzAjBm4m3zOjsXzNL8
+u8ALjQQCpBDT6dUHujzY5CRxG0xEHi5IXsXf4QwbctnjjvTeYA=
-----END CERTIFICATE-----
subject=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
issuer=/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
---
Acceptable client certificate CA names
/C=US/O=IBM Service Engage/CN=ca_ec_384.ibmserviceengage.com
/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=ca_ec_384.apm.ibmserviceengage.com
/C=US/O=DigiCert Inc/OU=www.digicert.com/CN=DigiCert
Global Root CA/C=US/O=IBM Service Engage/OU=Application Performance
Management/CN=*.agents.na.apm.ibmserviceengage.com
Server Temp Key: ECDH, prime256v1, 256 bits
---
SSL handshake has read 2659 bytes and written 261 bytes
---
New, TLSv1/SSLv3, Cipher is ECDHE-ECDSA-AES128-GCM-SHA256
Server public key is 384 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
Protocol : TLSv1.2
Cipher : ECDHE-ECDSA-AES128-GCM-SHA256
Session-ID:
A18C31D0B45A1166357C917E1CFCD86A9FBEDB4A0EB768EF5390AC28C95CB7EF
Session-ID-ctx:
Master-Key:
252B8FE2731E51AC0B79A27C7BED33CA8B15AF4CFD015C98DBACA46EA01DC40B
9E6B56E62E0F332FF6B56266B5ADD7B0
Key-Arg : None
Krb5 Principal: None
PSK identity: None
PSK identity hint: None
Start Time: 1510772474
Timeout : 7200 (sec)
Verify return code: 19 (self signed certificate in certificate chain)
---
DONE
SSL3 alert write:warning:close notify
```

如果计算机系统无法连接到 Cloud APM 服务器，那么会显示类似于以下示例的消息：

```
getaddrinfo: Name or service not known
connect:errno=2
```

如果因为密码套件在网络中的某个位置遭阻止，导致计算机系统无法获取服务器证书，那么会显示类似于以下示例的消息：

```
SSL_connect:failed
---
no peer certificate available
---
No client certificate CA names sent
```

关于此任务

在使用转发代理时，数据收集器首先打开与代理的 TCP 连接。数据收集器向转发代理发送连接请求和目标端点（Cloud APM 服务器）URL。然后，转发代理建立与目标端点的 TCP 连接，并在数据收集器和 Cloud APM 服务器之间设置 HTTPS 隧道会话。

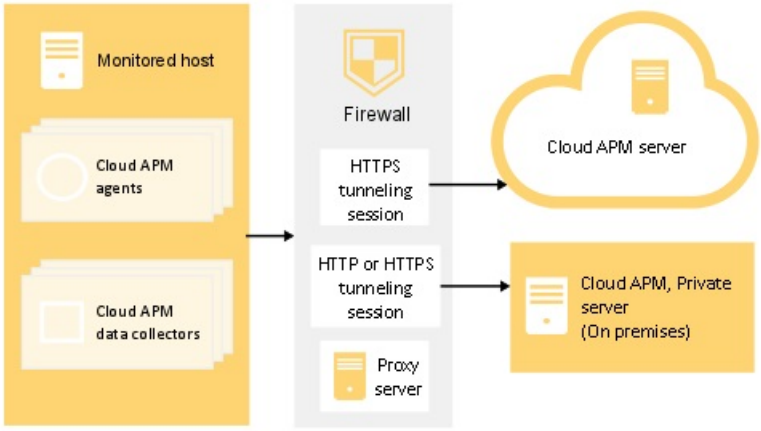


图 *2.* 使用转发代理的连接图

某些数据收集器（例如 Node.js 和 Liberty 数据收集器）支持认证代理。这些数据收集器支持使用配置的代理用户标识和密码登录转发代理。

过程

1. 要为 Python 数据收集器配置转发代理通信，请完成下列步骤之一：
   - 在文本编辑器中打开 *<dc home>*`/config.properties` 数据收集器属性文件，其中，*<dc home>* 是数据收集器的安装主目录，例如，`/usr/lib/python2.7/site-packages/ibm_python_dc`。使用代理主机和端口号更新 变量，例如，`APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085`。编辑此文件中的变量会影响启用了 Python 数据收集器的所有应用程序。

     注: 要为单个应用程序配置转发代理通信，将 *<dc home>*`/config.properties` 文件复制到该单个应用程序的目录。更新应用程序目录中的 变量。
   - 在 Linux 系统上运行以下命令：

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy host>:<http proxy port>
     ```

     例如，

     ```
     export APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085
     ```
2. 要为 Node.js 数据收集器配置转发代理通信，请完成下列步骤之一：
   - 在 Linux 系统上运行以下命令：

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy host>:<http proxy port>
     ```

     例如，

     ```
     export APM_GW_PROXY_CONNECTION =http://9.181.138.247:8085
     ```
   - 如果需要用户名和密码来访问 Node.js 数据收集器的转发代理服务器，请在 Linux 系统上运行以下命令：

     ```
     export APM_GW_PROXY_CONNECTION =http://<http proxy user>:
     <http proxy password>@<http proxy host>:<http proxy port>
     ```

     例如，

```
export APM_GW_PROXY_CONNECTION =http://Joe:passw0rd@9.181.138.247:8085
```

3. 要为 Liberty 数据收集器配置转发代理通信，编辑 *<Liberty server home>*/jvm.options 文件，其中，*<Liberty server home>* 是 Liberty 服务器主目录，例如：/opt/ibm/wlp/usr/servers/defaultServer/jvm.options。完成下列其中一个步骤：
   - 如果不需要认证，将以下代码添加到 jvm.options 文件：

```
-Dhttp.proxyHost=<http proxy host>
-Dhttp.proxyPort=<http proxy port>
-Dhttps.proxyHost=<https proxy host>
-Dhttps.proxyPort=<https proxy port>
-Djava.net.useSystemProxies=true
```

   - 如果需要用户名和密码访问转发代理服务器，将以下代码添加到 jvm.options 文件：

```
-Dhttp.proxyHost=<http proxy host>
-Dhttp.proxyPort=<http proxy port>
-Dhttp.proxyUser=<http proxy user>
-Dhttp.proxyPassword=<http proxy password>
-Dhttps.proxyHost=<https proxy host>
-Dhttps.proxyPort=<https proxy port>
-Dhttps.proxyUser=<https proxy user>
-Dhttps.proxyPassword=<https proxy password>
-Djava.net.useSystemProxies=true
```

4. 重新启动本地应用程序来实施更改。

结果
数据收集器已经配置为通过转发代理进行通信。

受管系统名称

受管系统名称 (MSN) 用来唯一地识别环境中的每个 Cloud APM 代理程序。这也是您在应用程序性能仪表板上，从导航器的**组**部分为每个受管系统选择组时，看到的实例名称。为了避免环境中发生冲突，请向代理程序分配唯一的 MSN。

根据代理程序类型不同，代理程序 MSN 格式也有所不同。它归入下列其中一个类别：

- 第 142 页的『单实例代理程序的一般 MSN 格式』
- 第 143 页的『多实例代理程序的一般 MSN 格式』
- 第 143 页的『特殊 MSN 格式』

单实例代理程序的一般 MSN 格式

对于大部分单实例代理程序，MSN 的一般格式如下所示：

```
hostname:pc
```

其中：

- *hostname* 是代理程序安装所在计算机的名称。如有需要，可以更改此部分。
- *pc* 是两个字符的大写代理程序代码，不可更改。有关代理程序代码的更多信息，请参阅第 150 页的『使用代理程序命令』。
- **:** 是分隔符，不可更改。

示例： linuxhost01:LZ 是 Linux OS 代理程序的 MSN。

某些单实例代理程序不采用第 143 页的表 9 所列的 MSN 格式。

MSN 限长 32 个字符。对于此 MSN 类别，共有 29 个字符可用于主机名，因为代理程序代码和分隔符不可更改。

要点： MSN 长度超过 32 个字符时将会截断，无法在 Cloud APM 控制台中正确显示。例如，如果主机名为 VeryLongSalesDivisionServerName03，那么受管系统名称应该为

VeryLongSalesDivisionServerName03:*PC*。但是，它会截断为 VeryLongSalesDivisionServerName0。

多实例代理程序的一般 MSN 格式

对于大部分多实例代理程序，MSN 的一般格式如下所示：

```
instancename:hostname:pc
```

其中：

- *instancename* 是您在代理程序配置期间指定的代理程序实例名称。使用此变量可确保在每台代理程序主计算机上，每种代理程序类型的每个实例都具有唯一的 MSN。

  切记：

  - 可以使用拉丁字母（a-z 和 A-Z）、阿拉伯数字 (0-9) 以及连字符/减号 (-) 来创建代理程序实例名称。
  - 在代理程序实例名称中，不允许使用下划线字符 (_)。
  - 您指定的实例名称有下列限制：
    - Linux AIX Linux 或 AIX 系统：28 个字符减去主机名长度。
    - Windows 在 Windows 系统上使用静默响应文件进行配置时：28 个字符减去主机名长度。例如，Server-Name 的长度是 11 个字符。因此，对于 Server-Name 主机上的代理程序实例，长度必须小于或等于 17 个字符。
    - Windows 在 Windows 系统上使用 Cloud APM 控制台进行配置时：20 个字符减去主机名超过 8 个字符的长度。例如，TestServer 的长度是 10 个字符，比 8 大 2。因此，对于 TestServer 主机上的代理程序实例，长度必须小于或等于 18 个字符。
- *hostname* 是代理程序安装所在计算机的名称。如有必要，可以更改 MSN 的主机名部分。
- *pc* 是两个字符的大写代理程序代码，不可更改。有关代理程序代码的更多信息，请参阅第 150 页的『使用代理程序命令』。
- **:** 是分隔符，不可更改。

示例： jboss1:win2016:JE 是 JBoss 代理程序的 MSN。

某些多实例代理程序不采用第 143 页的表 9 所列的 MSN 格式。

MSN 限长 32 个字符。对于此 MSN 类别，在实例名称与主机名之间可以使用 28 个字符，因为代理程序代码和分隔符不可更改。

要点： MSN 长度超过 32 个字符时将会截断，无法在 Cloud APM 控制台中正确显示。例如，如果指定 VeryLongInstanceName 作为实例名称，并且服务器名称为 Production09，那么受管系统名称应该是 VeryLongInstanceName:Production09:*PC*。但是，它会截断为 VeryLongInstanceName:Production0。

特殊 MSN 格式

特殊 MSN 格式适用于 MSN 不采用上述两种一般 MSN 格式的代理程序。这些代理程序列示在第 143 页的表 9 中。

这种特殊 MSN 限长 32 个字符。在第 143 页的表 9 中，只有 MSN 格式列中的斜体字符串可更改。

表 *9.* 特殊 *MSN* 格式

代理程序	MSN 格式	MSN 示例
Amazon EC2 代理程序	B5:*ec2subnodename*:INS	B5:sales:INS

表 9. 特殊 *MSN* 格式 *(续)*

代理程序	MSN 格式	MSN 示例
Amazon ELB 代理程序	· AL:*instancename*A:APP · AL:*instancename*C:CLA · AL:*instancename*N:NET	· AL:elb-inst3A:APP · AL:elb-inst3C:CLA · AL:elb-inst3N:NET
Azure Compute 代理程序	AK:*azure_compute_subnode_name*:AVM	AK:azc-inst3:AVM
Citrix VDI 代理程序	VD:*citrixsitename*:XDS	VD:xds1:XDS
DataPower 代理程序	BN:*datapowersystemname*:DPS	BN:datapower23:DPS
HTTP Server 代理程序	HU:*hostname_alias*:HUS	HU:docker-ihs_httpd:HUS
IBM Integration Bus 代理程序	*monitoredbrokername*:*agentID*:KQIB	TRADEBRK:AGT1:KQIB
MQ Appliance 代理程序	MK:*hostname_sectionname*:ARM	MK:bvtmin_linux150:ARM
Node.js 代理程序	NJ:*hostname_port*:NJA	NJ:KVM-014179_3000:NJA
Oracle Database 代理程序	· RZ:*dbconnection-instancename-hostname*:ASM · RZ:*dbconnection-instancename-hostname*:DG · RZ:*dbconnection-instancename-hostname*:RDB	RZ:11g-oracledbdemo-GVT-1BL:RDB
Ruby 代理程序	KM:*hostname_appname*:RAP	KM:nc9098036112_Blog:RAP
SAP 代理程序	· SAP 实例：*instancename-hostname_sid_instancenumber*:Ins · SAP 流程集成：*instancename-hostname*:PI · SAP Solution Manager: *instancename-hostname*:Slm · SAP 系统：*instancename-hostname*:Sys	· PS5-IBMSAP3V1_PS5_11:Ins · PS5-IBMSAP3V1:PI · PS8-IBMSAP3V3:Slm · PS5-IBMSAP3V1:Sys
SAP HANA Database 代理程序	· SAP Hana 数据库：S7:*dbname-systemsid*:HDB · SAP Hana 系统: *instancename*:*hostname*:S7	· S7:HNA-HNA:HDB · HNA:PS8760:S7
SAP NetWeaver Java Stack 代理程序	· SAP NW Java AS 集群: *instancename*:*hostname*:SV · SAP NW Java AS 实例: SV:*systemsid-jvmid*:NWJ	· J01:VPT02F17:SV · SV:J01-83309750:NWJ

表 9. 特殊 *MSN* 格式 *(续)*		
代理程序	**MSN 格式**	**MSN 示例**
UNIX OS 代理程序	*hostname*:KUX	worklight17:KUX
WebLogic 代理程序	WB:*instancename*:WLS	WB:Server1:WLS
WebSphere Applications 代理程序	· WebSphere Application Server: *serveralias*:*hostname*:KYNS · WebSphere Portal Server: *serveralias*:*hostname*:KYNR · WebSphere Process Server: *serveralias*:*hostname*:KYNP	simpletrade:worklight17:KYNS
WebSphere MQ 代理程序	*monitoredqmgrname*:*agentname*:MQ	TRADEQM:PoC:MQ
Windows OS 代理程序	Primary:*hostname*:NT	Primary:TRADEIIS1:NT

更改代理程序的受管系统名称

不同的过程适用于更改不同 Cloud APM 代理程序的受管系统名称。对于某些代理程序，更改受管系统名称意味着更改受管系统名称中的主机名和/或代理程序实例名称。对于其他代理程序，需要完成特定的过程才能更改受管系统名称。

开始之前

熟悉第 142 页的『受管系统名称』中描述的受管系统名称格式和命名限制。

关于此任务

对于大部分 Cloud APM 代理程序，您可以使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中使用的主机名。对于多实例代理程序，要更改受管系统名称中的代理程序实例名称，您可以使用代理程序配置参数。如果您已配置代理程序，那么必须进行重新配置才能指定另一个代理程序实例名称。重新配置代理程序之后，您将无法检索由先前代理程序实例收集的数据。

可能无法通过单一过程来更改受管系统名称，具体取决于您想要更改的受管系统名称部分。

要了解您所关注的代理程序的受管系统名称更改方法，请参阅第 145 页的表 10。

例外: HTTP Server 代理程序、Node.js 代理程序或 Synthetic Playback 代理程序不支持更改受管系统名称

表 *10. Cloud APM* 代理程序的受管系统名称更改方法	
代理程序	**受管系统名称更改方法**
Amazon EC2 代理程序	使用代理程序配置参数来更改受管系统名称中的 EC2 子节点名称，请参阅第 167 页的『Amazon EC2 代理程序的配置参数』。
Amazon ELB 代理程序	创建具有新实例名称的新代理程序实例以更改受管系统名称。
Azure Compute 代理程序	使用代理程序配置参数来更改受管系统名称中的子节点名称，请参阅第 181 页的『Azure Compute 代理程序的配置参数』。
Cassandra 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Cisco UCS agent	使用代理程序配置参数来更改代理程序实例名称，请参阅第 187 页的『代理程序的配置参数』。

表 *10. Cloud APM* 代理程序的受管系统名称更改方法 *(续)*	
代理程序	**受管系统名称更改方法**
Citrix VDI 代理程序	使用代理程序配置参数来更改 Citrix 站点名称，请参阅第 197 页的『Citrix VDI 代理程序的配置参数』。
Db2 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
DataPower 代理程序	使用代理程序配置参数来更改受管系统名称，请参阅第 207 页的『配置 DataPower 代理程序』。
DataStage 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Hadoop 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
HMC Base 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
IBM Cloud 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
IBM Integration Bus 代理程序	第 249 页的『为 IBM Integration Bus 代理程序指定唯一受管系统名称』
JBoss 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Linux KVM 代理程序	使用代理程序配置参数，请参阅第 409 页的『配置 Linux KVM 监视』。
Linux OS 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft .NET 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft Active Directory 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft Exchange Server 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft Hyper-V Server 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。

表 *10. Cloud APM* 代理程序的受管系统名称更改方法 *(续)*

代理程序	**受管系统名称更改方法**
Microsoft IIS 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft Office 365 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Microsoft SQL Server 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Microsoft SharePoint Server 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
MongoDB 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
MySQL 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
NetApp Storage 代理程序	使用代理程序配置参数，请参阅第 498 页的『配置 NetApp Storage 监视』。
OpenStack 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Oracle Database 代理程序	使用代理程序配置参数，请参阅第 526 页的『配置 Oracle 数据库监视』。
PHP 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
PostgreSQL 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
RabbitMQ 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Response Time Monitoring 代理程序	第 611 页的『为 Response Time Monitoring 代理程序指定唯一的受管系统名称』
Ruby 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。

表 *10. Cloud APM* 代理程序的受管系统名称更改方法 *(续)*	
代理程序	**受管系统名称更改方法**
SAP 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
SAP HANA Database 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
SAP NetWeaver Java Stack 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Siebel 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Skype for Business Server 代理程序（旧称 Microsoft Lync Server 代理程序）	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
Sterling File Gateway 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Sterling Connect Direct 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
Tomcat 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
UNIX OS 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。
WebSphere Applications 代理程序	要更改受管系统名称中的主机名，请参阅如何更改 WAS 代理程序实例的受管系统名称中使用的主机名？。 要更改受管系统名称中的服务器别名，必须重新配置代理程序，请参阅第 718 页的『以交互方式重新配置数据收集器』。
WebSphere Infrastructure Manager 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。 使用代理程序配置参数来更改受管系统名称中的实例名称。
WebSphere MQ 代理程序	第 798 页的『为多个队列管理器指定唯一受管系统名称』
Windows OS 代理程序	使用 **CTIRA_HOSTNAME** 参数来更改受管系统名称中的主机名。请参阅第 149 页的『更改受管系统名称中的主机名』。

更改受管系统名称中的主机名

关于此任务

更改受管系统名称中的主机名并不常见。在代理程序配置期间，将会自动检测并设置主机名。只有在必要时才应更改受管系统名称中的主机名，并且，请确保您指定的值不会因为受管系统名称命名限制而导致截断。

过程

1. 停止代理程序的所有现有实例并等待 Cloud APM 控制台显示代理程序或其子节点脱机。如果不存在任何现有的代理程序实例，请前进到下一步。

   有关停止代理程序实例的更多信息，请参阅第 150 页的『使用代理程序命令』。
2. 对于单实例代理程序，请完成下列步骤以更改 **CTIRA_HOSTNAME** 参数。对 **CTIRA_HOSTNAME** 参数指定的值就是应用于所有新代理程序实例的值。

   a) 建立下列文件的备份副本：

   - Linux AIX *install_dir*/config/*pc*.environment
   - Windows *install_dir*/TMAITM6_x64/k*pc*cma.ini

   其中：

   - *install_dir* 是代理程序安装目录。
   - *pc* 是两个字符的代理程序代码。请参阅代理程序名称和代理程序代码表。

   b) 编辑该文件，以如下方式更改 **CTIRA_HOSTNAME** 参数值，其中 *newhostname* 是定制字符串，用来替代代理程序安装所在计算机的实际主机名。

   - Linux AIX **CTIRA_HOSTNAME**=*newhostname*
   - Windows **CTIRA_HOSTNAME**=*newhostname* .TYPE=REG_EXPAND_SZ

   c) 保存更改。
3. 对于多实例代理程序，请完成下列步骤以更改 **CTIRA_HOSTNAME** 参数。通常，一台计算机上的所有代理程序实例使用同一个主机名值。如果要让各个代理程序实例使用不同的值，请更改执行此步骤时对 **CTIRA_HOSTNAME** 指定的值。

   a) 建立下列文件的备份副本：

   - Linux AIX *install_dir*/config/*pc_instance*.environment
   - Windows *install_dir*/TMAITM6_x64/k*pc*cma_*instance*.ini

   b) 编辑该文件，以如下方式更改 **CTIRA_HOSTNAME** 参数值：

   - Linux AIX **CTIRA_HOSTNAME**=*newhostname*
   - Windows **CTIRA_HOSTNAME**=*newhostname* .TYPE=REG_EXPAND_SZ

   c) 保存更改。
4. Windows
   重新配置现有的代理程序实例。
5. 启动所有代理程序实例。

下一步做什么

更改代理程序受管系统名称后，请启动 Cloud APM 控制台，并通过从应用程序中移除旧的受管系统名称，并在其位置添加新的受管系统名称修改应用程序。

配置代理程序

安装后，某些代理程序将自动配置和启动，而某些代理程序需要手动配置但是会自动启动。某些代理程序必须手动配置和启动。多实例代理程序需要手动创建和启动第一个实例。

开始之前

当安装代理程序时，样本静默配置文件放置在 /opt/ibm/apm/agent/samples 目录中，例如，ynv_silent_config_agent.txt 和 datapower_silent_config.txt。

注: 某些代理程序，例如 Monitoring Agent for WebSphere Applications，针对不同的任务（例如，配置数据收集器）具有多个静默配置文件。

关于此任务

有关代理程序的特定部署详细信息，请参阅第 95 页的『第 5 章 代理程序 以及数据收集器部署』。

要配置代理程序，您可以使用命令行或静默响应文件，如此过程中所述。

配置方法因代理程序而异；请使用针对您的代理程序提供的过程。

过程

- 运行 *agent-name*.sh config 命令。

  有关更多命令，请参阅第 153 页的表 12 和第 154 页的表 13。
- 编辑静默响应文件，然后运行以下命令之一：
  - 对于单个实例代理程序，请运行以下命令：

    ```
    agent-name.sh config response_file
    ```
  - 对于多个实例代理程序，请运行以下命令：

    ```
    agent-name.sh config instance_name response_file
    ```
  - 其中，*instance_name* 是实例名称，可以分配以指示您正在监视的内容。
- Windows

  对于 Windows 系统上支持的代理程序，可以使用 IBM Cloud Application Performance Management 窗口执行特定配置任务。单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Cloud Application Performance Management**。有关更多信息，请参阅第 155 页的『在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口』。
- 要为某些代理程序执行高级配置，例如，配置事务跟踪或数据收集以及启用诊断数据，请使用“**代理程序配置**”窗口。有关更多信息，请参阅第 155 页的『代理程序配置页面』。

使用代理程序命令

用来安装监视代理程序的脚本也可用于检查已安装代理程序的状态、停止或启动代理程序或者卸载代理程序。

关于此任务

提供了代理程序名称和代理程序代码供您参考。

请在以下命令中使用代理程序名称：

Linux AIX *name*-agent.sh

Windows *name*-agent.bat

其中，*name* 是在第 151 页的表 11 中指定的代理程序名称。

表 *11.* 代理程序名称和代理程序代码

监视代理程序	***name***	**两个字母的代理程序代码**
Monitoring Agent for Amazon EC2	amazon_ec2	b5
Monitoring Agent for Azure Compute	azure_compute	ak
Monitoring Agent for Cassandra	cassandra	zc
Monitoring Agent for Cisco UCS	cisco_ucs	v6
Monitoring Agent for Citrix Virtual Desktop Infrastructure	citrix_vdi	vd
Monitoring Agent for DataPower	datapower	bn
Monitoring Agent for Db2	db2	ud
Monitoring Agent for Hadoop	hadoop	h8
Monitoring Agent for HMC Base	hmc_base	ph
Monitoring Agent for HTTP Server	http_server	hu
Monitoring Agent for IBM Cloud	ibm_cloud	fs
Monitoring Agent for IBM Integration Bus	iib	qi
Monitoring Agent for MQ Appliance	ibm_mq_appliances	mk
Monitoring Agent for InfoSphere DataStage	datastage	dt
Monitoring Agent for JBoss	jboss	je
Monitoring Agent for Linux KVM	linux_kvm	v1
Monitoring Agent for Linux OS	os	lz
Monitoring Agent for MariaDB	mariadb	mj
Monitoring Agent for Microsoft Active Directory	msad	3z
Monitoring Agent for Microsoft Cluster Server	mscs	q5
Monitoring Agent for Microsoft Exchange Server	msexch	ex
Monitoring Agent for Microsoft Hyper-V Server	microsoft_hyper-v_server	hv
Monitoring Agent for Microsoft Internet Information Services	msiis	q7
Monitoring Agent for Skype for Business Server（旧称 Microsoft Lync Server）	skype_for_business_server	ql
Monitoring Agent for Microsoft .NET	dotnet	qe
Monitoring Agent for Microsoft Office 365	microsoft_office365	mo
Monitoring Agent for Microsoft SharePoint Server	ms_sharepoint_server	qp
Monitoring Agent for Microsoft SQL Server	mssql	oq
Monitoring Agent for MongoDB	mongodb	kj
Monitoring Agent for MySQL	mysql	se
Monitoring Agent for NetApp Storage	netapp_storage	nu

表 11. 代理程序名称和代理程序代码 (续)

监视代理程序	*name*	两个字母的代理程序代码
Monitoring Agent for Node.js	nodejs	nj
Monitoring Agent for OpenStack	openstack	sg
Monitoring Agent for Oracle Database	oracle_database	rz
Monitoring Agent for PHP	php	pj
Monitoring Agent for PostgreSQL	postgresql	pn
Monitoring Agent for Python	python	pg
Monitoring Agent for RabbitMQ	rabbitMQ	zr
Monitoring Agent for Ruby	ruby	km
Monitoring Agent for SAP Applications	sap	sa
Monitoring Agent for SAP HANA Database	sap_hana_database	s7
Monitoring Agent for SAP NetWeaver Java Stack	sap_netweaver_java_stack	sv
Monitoring Agent for Siebel	siebel	uy
Monitoring Agent for Sterling Connect Direct	sterling_connect_direct-agent	FC
Monitoring Agent for Sterling File Gateway	file_gateway	fg
Monitoring Agent for Sybase Server	sybase	oy
Monitoring Agent for Synthetic Playback	synthetic_transactions	sn
Monitoring Agent for Tomcat	tomcat	ot
Monitoring Agent for UNIX OS	os	ux
Monitoring Agent for VMware VI	vmware_vi	vm
Monitoring Agent for WebLogic	oracle_weblogic	wb
Monitoring Agent for WebSphere Applications	was	yn
Monitoring Agent for WebSphere Infrastructure Manager	wim	d0
Monitoring Agent for WebSphere MQ	mq	mq
Monitoring Agent for Windows OS	os	nt
Response Time Monitoring Agent	rt	t5

过程

- Linux AIX
  在要将命令发送到监视代理程序的系统上，切换到 *install_dir*/bin 目录。输入第 153 页的表 12 中的任意命令，其中 *name* 是在第 151 页的表 11 中指定的代理程序名称。

表 12. *UNIX* 和 *Linux* 系统的命令

命令	描述
./*name*-agent.sh status	检查代理程序状态。状态可以是正在运行或未在运行。当代理程序正在运行时，还会检查代理程序和 Cloud APM 服务器之间的连接状态。可能的负面连接状态包括：“连接失败”、“检测到错误”和“断开连接错误”。正面状态为“已连接”，这是预期状态。过渡状态是“正在连接”。状态“未知”表示无法识别代理程序状态，原因可能是文件系统或代理程序日志文件中存在错误。
./*name*-agent.sh start	启动监视代理程序。如果代理程序具有实例，请在命令后输入实例名称。
./*name*-agent.sh stop	停止代理程序。如果代理程序具有实例，请在命令后输入实例名称。
./*name*-agent.sh prereqcheck	运行先决条件扫描。该命令选项可供大多数代理程序使用。
./*name*-agent.sh install	安装监视代理程序。有关更多信息，请参阅第 102 页的『在 UNIX 系统上安装代理程序』和第 107 页的『在 Linux 系统上安装代理程序』。
./*name*-agent.sh config *instance_name path_to_silent_config_file*	配置监视代理程序。从 *install_dir*/bin 目录运行命令并添加响应文件路径（如果需要）。 如果代理程序具有实例，那么请输入一个实例名称。有关哪个代理程序是多实例代理程序的更多信息，请参阅第 96 页的表 7。 *silent_config_file* 是可选的。如果未指定用于静默配置的文件，那么可以通过遵循提示来以交互方式配置监视代理程序。
./*name*-agent.sh uninstall	卸载监视代理程序。有关更多信息，请参阅第 124 页的『卸载代理程序』。
./smai-agent.sh uninstall_all	卸载受管系统上的所有监视代理程序。
./*name*-agent.sh remove *instance_name*	移除多实例代理程序的某个实例。
./*name*-agent.sh	查看可用于脚本的函数的描述。

- Windows

在要将命令发送到监视代理程序的系统上，在命令提示符中切换到 *install_dir*\BIN 目录，例如：C:\IBM\APM\bin。输入第 154 页的表 13 中的任意命令，其中 *name* 是在第 151 页的表 11 中指定的代理程序名称。

表 13. 用于 Windows 系统的命令	
命令	**Description**
name-agent.bat status	检查代理程序状态。 检查代理程序和 Cloud APM 服务器之间的连接状态。可能的负面连接状态包括：“连接失败”、“检测到错误”和“断开连接错误”。正面状态为“已连接”，这是预期状态。过渡状态是“正在连接”。状态“未知”表示无法识别代理程序状态，原因可能是文件系统或代理程序日志文件中存在错误。
name-agent.bat start	启动监视代理程序。如果代理程序具有实例，请在命令后输入实例名称。
name-agent.bat stop	停止代理程序。如果代理程序具有实例，请在命令后输入实例名称。
name-agent.bat prereqcheck	运行先决条件扫描。该命令选项可供大多数代理程序使用。
name-agent.bat install	安装监视代理程序。有关更多信息，请参阅第 117 页的『安装代理程序』。
name-agent.bat config *instance_name path_to_silent_config_file*	配置监视代理程序。从 *install_dir*\bin 目录运行命令并添加响应文件路径（如果需要）。 如果代理程序具有实例，那么请输入一个实例名称。有关哪个代理程序是多实例代理程序的更多信息，请参阅第 96 页的表 7。 *silent_config_file* 是可选的。如果未指定用于静默配置的文件，那么可以通过遵循提示来以交互方式配置监视代理程序。
name-agent.bat uninstall	卸载监视代理程序。有关更多信息，请参阅第 124 页的『卸载代理程序』。
smai-agent.bat uninstall_all	卸载受管系统上的所有监视代理程序。
name-agent.bat remove *instance_name*	移除多实例代理程序的某个实例。
name-agent.bat	查看可用于脚本的函数的描述。

代理程序版本命令

- 要查看环境中的代理程序版本，请运行以下命令：
  - Linux AIX

    ```
    install_dir/bin/cinfo
    ```

    输入 1 可显示版本。
  - Windows

    ```
    install_dir/InstallITM/kincinfo
    ```

相关任务

第 155 页的『在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口』

在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口

Windows 支持的代理程序具有 GUI 实用程序，您可以用于执行代理程序配置和检查连接状态。

GUI 配置实用程序不可用于 Monitoring Agent for WebSphere MQ 或 Monitoring Agent for IBM Integration Bus。

过程

- 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。

结果

此时将显示 IBM Cloud Application Performance Management 窗口。将列出每个安装的代理程序组件及其配置状态、是启动还是停止、连接状态、版本号以及其他信息。

下一步做什么

可以通过右键单击代理程序并选择选项来启动或停止代理程序或者配置参数。

代理程序配置页面

使用“**代理程序配置**”页面可集中配置诸如 Response Time Monitoring Agent 和 WebSphere Applications 代理程序之类的代理程序的设置。

一般用法

单击导航栏中的 **系统配置** > **代理程序配置**后，系统会显示选项卡式仪表板，每个可配置的监视代理程序一个选项卡。表将显示配置信息（例如每个受管系统的名称和 IP 地址）的列，且每个受管系统各有一行。

操作

使用“操作”选项可启用或禁用诸如事务跟踪或数据收集之类的功能。

调整列大小

拖动列标题边框可调整列宽。

列排序

在列标题内单击，即可按该列进行排序。再次单击同一列标题可在升序和降序排序顺序之间切换。

表过滤器

在过滤器文本框  中单击鼠标，然后输入要作为表过滤依据的值的开头。您输入时，将会过滤掉不符合条件的表行，并且行**总计**将根据找到的行数进行更新。

单击过滤器框  中的“x” “”，或按退格键即可清除过滤器。

代理程序配置

有关特定代理程序的设置的更多信息，请参阅以下主题：

- DataPower 代理程序: 第 199 页的『配置 DataPower 监视』
- IBM Integration Bus 代理程序: 第 248 页的『配置 IBM Integration Bus 代理程序的事务跟踪』
- Internet Service Monitoring 第 385 页的『在 Windows 系统上配置代理程序』
- JBoss 代理程序: 第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』
- Microsoft .NET 代理程序: 第 453 页的『启用事务跟踪和诊断数据收集』
- OS 代理程序日志文件监视： 第 540 页的『为操作系统代理程序添加或移除日志文件监视配置』
- Response Time Monitoring Agent: 第 588 页的『使用“代理程序配置”页面进行配置』
- 地理位置： 第 607 页的『定制最终用户事务位置值』
- Ruby 代理程序: 第 618 页的『为 Ruby 应用程序禁用或启用诊断数据』
- SAP NetWeaver Java Stack 代理程序: 第 654 页的『启用事务跟踪和诊断数据收集』
- Tomcat 代理程序: 第 686 页的『启用事务跟踪和诊断数据收集』
- WebLogic 代理程序: 第 702 页的『配置 WebLogic 代理程序的事务跟踪』
- WebSphere Applications 代理程序: 第 745 页的『在代理程序配置页面上动态配置数据收集』

· WebSphere MQ 代理程序: 第 799 页的『配置 WebSphere MQ 代理程序的事务跟踪』

以非 root 用户身份配置代理程序
如果要以非 root 用户身份配置代理程序，请在系统上创建公共组并使每个用户成为此组的成员。

开始之前

如果以 root 用户或非 root 用户身份安装代理程序并且要以相同的用户身份配置代理程序，那么不需要特殊操作。

如果以所选用户身份安装了代理程序并要以另一个用户身份配置代理程序，请在系统上创建公共组。使所有代理程序管理用户成为此公共组的成员。将所有代理程序文件和目录的所有权传输给此组。

注:

· 对于 HTTP Server 代理程序，如果您以非 root 用户身份配置代理程序，那么该非 root 用户必须具有与启动 IBM HTTP Server 的用户相同的用户标识。否则，此代理程序在发现 IBM HTTP Server 时会遇到问题。
· 对于 IBM Integration Bus 代理程序，如果 IBM Integration Bus 安装是单用户部署，请使用安装 IBM Integration Bus 的用户标识来配置代理程序。在配置代理程序之前，请对此用户标识完成下列步骤。

过程

1. 如第 107 页的『在 Linux 系统上安装代理程序』和第 102 页的『在 UNIX 系统上安装代理程序』中所述在 Linux 或 UNIX 上安装监视代理程序。
2. 以非 root 用户的群组名来运行 `./secure.sh` 脚本，以保护文件的安全，并设置文件的文件组所有权。例如：**`./secure.sh -g db2iadm1`**
3. 根据需要在 Linux 或 AIX 上配置监视代理程序，请参阅第 135 页的『第 7 章 配置环境』。
4. 要更新系统启动脚本，请以 root 用户身份或 sudo 用户访问权运行以下脚本：*`install_dir`*`/bin/UpdateAutoRun.sh`

下一步做什么

有关 **`./secure.sh`** 脚本的更多信息，请参阅保护代理程序安装文件的安全。

使用相同的用户标识进行代理程序安装和升级。

在 UNIX 和 Linux 系统上禁用自动代理程序启动
在 UNIX 或 Linux 系统上，代理程序可在操作系统重新启动后自动启动。如果您不希望代理程序在系统重新启动后自动启动，您可以禁用代理程序自动启动。

关于此任务

如果以 root 用户身份在 UNIX 或 Linux 系统上安装代理程序，那么该代理程序可在系统重新启动后自动启动。或者，如果以非 root 用户身份安装代理程序但在安装后以 root 用户身份运行 **`UpdateAutoRun.sh`** 脚本，那么该代理程序可在系统重新启动后自动启动。

过程

1. 完成以下步骤以禁用某些代理程序上的自动启动:
   a. 对于 Linux OS 代理程序 和 WebSphere® Applications 代理程序，将以下代码添加到 `agent_install_dir/registry/kcirunas.cfg` 文件:

```
<productCode>lz</productCode>
 <default>
   <autoStart>no</autoStart>
 </default>
 <productCode>yn</productCode>
 <default>
   <autoStart>no</autoStart>
 </default>
```

b. 运行 **agent_install_dir/bin/UpdateAutoRun.sh** 命令。

2. 完成以下步骤以启用某些代理程序上的自动启动：

a. 对于 Linux OS 代理程序 和 WebSphere® Applications 代理程序，在 agent_install_dir/registry/kcirunas.cfg 文件中，将 *<autoStart>* 标记的值更改为 **yes**。

b. 打开 agent_install_dir/registry/AutoStart 文件并检查内容。

c. 删除 /etc/init.d/ITMAgents{$Num} 文件，其中，*{$Num}* 在 agent_install_dir/registry/AutoStart 文件中为正数。如果值为 1，那么必须删除 /etc/init.d/ITMAgents1 文件。

d. 运行 **agent_install_dir/bin/UpdateAutoRun.sh** 命令。

结果

在系统重新启动后，代理程序脚本将不会自动运行以启动代理程序。

用于配置数据收集器的常规过程

要在 Cloud APM 控制台中使用数据收集器查看应用程序的监视数据，必须完成多项配置任务。

关于此任务

以下过程是配置应用程序监视的路线图，其中包括必需步骤、有条件步骤和可选步骤。请根据您的需要，完成必要的步骤。

过程

1. 下载并解压缩数据收集器软件包。有关指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。
2. 配置数据收集器，以收集有关 IBM Cloud 应用程序和本地部署应用程序的监视数据并将其发送到 Cloud APM 服务器。请根据您的应用程序类型，完成下列一项或多项任务：

Liberty 应用程序

- 第 749 页的『为本地应用程序配置 Liberty 数据收集器』
- 第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』

Node.js 应用程序

- 第 509 页的『针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器』
- 第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』

Python 应用程序

- 第 572 页的『针对 IBM Cloud 应用程序配置 Python 数据收集器』
- 第 576 页的『为本地应用程序配置 Python 数据收集器』

Ruby 应用程序

- 第 618 页的『针对 IBM Cloud 应用程序配置 Ruby 数据收集器』

Java 应用程序

- 第 388 页的『配置 J2SE 监视』

3. 如果密钥文件或 Cloud APM 服务器更改，请将数据收集器重新连接到 Cloud APM 服务器。有关指示信息，请参阅第 158 页的『将数据收集器重新连接到 Cloud APM 服务器』。

下一步做什么

完成所有必要的配置任务之后，您可确认 IBM Cloud 应用程序的监视数据显示在 Cloud APM 控制台中。

将数据收集器重新连接到 Cloud APM 服务器

如果 Cloud APM 服务器、密钥文件或密钥文件密码已更改，您必须设置多个环境变量，以便将数据收集器重新连接到 Cloud APM 服务器。

开始之前

如果密钥文件已更改，请先使用 Base64 对密钥文件的明文密码进行加密。如果您是 Linux 用户，请运行以下命令：

```
echo -n keyfile_password | base64
```

命令输出就是加密后的密码。例如，如果纯文本密码是 password，那么命令输出 cGFzc3dvcmQ= 是加密密码。然后，使用加密密码在以下配置中设置 APM_KEYFILE_PSWD: *encrypted_keyfile_password* 和 APM_KEYFILE_PSWD=*encrypted_keyfile_password*。

过程

- 要将数据收集器重新连接到 IBM Cloud 应用程序的 Cloud APM 服务器，请参阅第 158 页的『重新连接 IBM Cloud 应用程序的数据收集器』。
- 要将数据收集器重新连接到本地部署的 Cloud APM 服务器，请参阅第 159 页的『重新连接本地部署应用程序的数据收集器』。

重新连接 *IBM Cloud* 应用程序的数据收集器

关于此任务

要将数据收集器重新连接到 Cloud APM 服务器，您有下列两个选项：

· 编辑应用程序的 manifest.yml 以设置变量。

· 在 IBM Cloud UI 上设置变量。

过程

- 要使用 IBM Cloud 应用程序的 manifest.yml 文件来重新连接数据收集器，请完成下列步骤：
  a) 根据更改，在 IBM Cloud 应用程序的 manifest.yml 文件中编辑变量。
  - 要将网关配置为使用 *HTTP*，请设置下列变量：

    ```
    APM_BM_GATEWAY_URL: http://server_ip_or_hostname:80
    ```

  - 要将网关配置为使用 *HTTPS*，请设置下列三个变量：

    ```
    APM_BM_GATEWAY_URL: https://server_ip_or_hostname:443
    APM_KEYFILE_PSWD: encrypted_keyfile_password
    APM_KEYFILE_URL: http://hosted_http_server:port/keyfile_name
    ```

    提示: Liberty 数据收集器的密钥文件是 .jks 文件。对于 Python、Node.js 和 Liberty 数据收集器，密钥文件是 .p12 文件。

  b) 切换到 IBM Cloud 应用程序的目录，然后运行以下命令：

  ```
  cf push
  ```

- 要使用 IBM Cloud UI 来重新连接数据收集器，请完成下列步骤：
  a) 登录 IBM Cloud UI。
  b) 单击 IBM Cloud 应用程序。
  c) 单击左侧面板上的**运行时**。
  d) 切换到**环境变量**选项卡。
  e) 在**用户定义**部分中，根据需要，使用下列其中一种方法来定义变量：

– 要将网关配置为使用 *HTTP*，请设置下列变量：

```
APM_BM_GATEWAY_URL: http://server_ip_or_hostname:80
```

– 要将网关配置为使用 *HTTPS*，请设置下列三个变量：

```
APM_BM_GATEWAY_URL: https://server_ip_or_hostname:443
APM_KEYFILE_PSWD: encrypted_keyfile_password
APM_KEYFILE_URL: http://hosted_http_server:port/keyfile_name
```

提示: Liberty 数据收集器的密钥文件是 `.jks` 文件。对于 Python、Node.js 和 Liberty 数据收集器，密钥文件是 `.p12` 文件。

f) 从您运行 **cf push** 命令以推送应用程序的目录中，运行以下命令以使更改生效：

```
cf restage <app_name>
```

结果

已正确设置变量的值，现在可将数据收集器连接到 Cloud APM 服务器。

重新连接本地部署应用程序的数据收集器

关于此任务

通过修改 `global.environment` 或 `dc.java.properties` 文件，可以定制数据收集器和 Cloud APM 服务器之间的连接。

过程

1. 查找包含连接变量的相应文件。
   a) 对于 Liberty 数据收集器、Node.js 数据收集器 和 Python 数据收集器，根据下表中的信息查找 `global.environment` 文件：

数据收集器名称	**global.environment 文件的目录**
Liberty 数据收集器	安装了 Liberty 数据收集器 的 `itcamdc/etc/global.environment` 文件夹。
Node.js 数据收集器	安装了 Node.js 数据收集器 的 `ibmapm/etc` 文件夹。
Python 数据收集器	安装了 Python 数据收集器 的 `etc` 文件夹。

   b) 对于 J2SE 数据收集器，在 *DC_HOME*`/itcamdc/etc` 文件夹中查找 `dc.java.properties` 文件。*DC_HOME* 是安装了 J2SE 数据收集器 的目录。
2. 根据更改编辑相应文件中的变量。
   a) 对于 Liberty 数据收集器、Node.js 数据收集器 和 Python 数据收集器，根据以下指示信息编辑 `global.environment` 文件：

   · 要将网关配置为使用 *HTTP*，请设置下列变量：

```
APM_BM_GATEWAY_URL=http://server_ip_or_hostname:80
```

   · 要将网关配置为使用 *HTTPS*，请设置下列变量：

```
APM_BM_GATEWAY_URL=https://server_ip_or_hostname:443
APM_KEYFILE_PSWD=encrypted_keyfile_password
APM_KEYFILE_URL=http://hosted_http_server_:port/keyfile_name
```

提示: Liberty 数据收集器 的密钥文件是 `.jks` 文件。对于 Python、Node.js 和 Liberty 数据收集器，密钥文件是 `.p12` 文件。

b) 对于 J2SE 数据收集器，根据以下指示信息编辑 `dc.java.properties` 文件：

- 要将网关配置为使用 *HTTP*，请设置下列变量：

```
apm.http.type=http
```

如果将该变量的值留空，那么 `http` 为缺省值

- 要将网关配置为使用 *HTTPS*，请设置下列变量：

```
apm.ssl.password=encrypted_keyfile_password
apm.http.type=https
```

要点: 如果密码发生更改，将 *DC_HOME*`/itcamdc/etc/keyfile.jks` 文件替换为 Cloud APM 服务器的 `/opt/ibm/ccm/keyfiles/default.agent/keyfiles/keyfile.jks` 文件，其中，*DC_HOME* 是 J2SE 数据收集器 的主目录。

3. 可选： 如果不使用 Node.js 数据收集器的缺省密钥文件，请设置以下变量：

```
APM_SNI=owner_host_in_the_key_file
```

提示: 要查明 *owner host* 变量的值，打开使用的密钥文件并搜索 `owner`。然后将 *APM_SNI* 变量设置为与 *owner* 相同的值。

4. 重新启动应用程序以使更改生效。

结果
已正确设置变量的值，现在可将数据收集器连接到 Cloud APM 服务器。

样本 `manifest.yml` 文件
请参阅下列各行，以了解 IBM Cloud 应用程序的 `manifest.yml` 文件内容：

```
applications:

- disk_quota: 1024M
  host: myBluemixApp
  name: myBluemixApp
  path: .
  domain: mybluemix.net

  instances: 1
  memory: 512M

  env:
    KNJ_ENABLE_TT: "true"
    KNJ_SAMPLING: 1
```

从 Cloud APM 控制台中移除数据收集器
取消配置数据收集器后，您还应该从应用程序中以及从其添加到的资源组中移除该数据收集器。否则，Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。

过程

1. 手动编辑应用程序以从其添加到的任何应用程序中移除数据收集器。
   类似于从应用程序中移除脱机代理程序，请参阅第 934 页的『查看和移除脱机代理程序』。
2. 从其添加到的任何定制资源组中移除数据收集器。
   有关更多信息，请参阅第 833 页的『资源组管理器』。
3. 为 Cloud APM 运营团队开具凭证以对 Cloud APM 服务器执行以下步骤：
   a) 编辑 *install_dir*`/serveragents/config/hostname_bi.cfg` 文件以移除已取消配置的数据收集器的行。
   b) 以 root 用户身份运行以下命令，为数据收集器重新启动服务器组件：

```
apm restart biagent
```

结果

几分钟后，当您为数据收集器选择系统资源组时，Cloud APM 控制台将在**我的组件**应用程序中和 **资源组管理器** UI 中指示数据收集器已脱机。

高级配置页面中的**移除脱机系统延迟**配置属性指定时间间隔后，将从**我的组件**及其系统资源组中自动移除数据收集器。

提示: 您可以在**高级配置**页面中调整**移除脱机系统延迟**设置以增加或减少从视图中移除脱机代理程序前的等待时间。有关更多信息，请参阅第 911 页的『代理程序预订设施』。

切记: 如果数据收集器为 Cloud APM 服务器提供事务跟踪数据，那么 Cloud APM 控制台可能会继续在**我的组件**应用程序中显示数据收集器，并在**移除脱机系统延迟**设置指定的时间段到期后，对数据收集器显示代理程序无效消息。如果已安装 Cloud APM 8.1.4.0 服务器临时修订 3 或更高版本，在停止从数据收集器接收事务跟踪数据 8 天后，无效数据收集器最终将从**我的组件**应用程序中移除。

配置 Amazon EC2 监视

The Amazon EC2 代理程序为您提供了中央监视点，用于监视 Amazon Elastic Compute Cloud (EC2) 实例的运行状况、可用性和性能。该代理程序显示一组综合度量值，用于帮助您做出有关 EC2 环境的明智决定。这些度量包括 CPU 使用率、弹性块存储器 (EBS) 使用率、网络使用率、Amazon Web Service (AWS) 维护更新和磁盘性能。

开始之前

- 请通读第 161 页的『配置 Amazon EC2 监视』主题，确定完成配置所需的内容。
- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 Amazon EC2 代理程序的系统需求。有关最新系统需求信息，请参阅 Amazon EC2 代理程序的软件产品兼容性报告 (SPCR)。
- 确保您掌握下列信息:
  - 所要监视的 EC2 实例所在 AWS 区域的名称列表。
  - 有权访问每个 AWS 区域的 AWS 安全凭证（访问密钥标识和访问密钥）。
- 请确保用于每个 AWS 区域的 AWS 安全凭证隶属于至少包含 *AmazonEC2ReadOnlyAccess* 策略的组。

关于此任务

Amazon EC2 代理程序既是多实例代理程序，也是子节点代理程序。您可以创建一个具有多个子节点（每个 Amazon EC2 区域对应于一个子节点）的代理程序实例，也可以为每个 Amazon EC2 区域创建一个代理程序实例（一个子节点对应于该类区域）。或者可以创建各种类型配置的组合。配置代理程序实例后，必须手动启动每个代理程序实例。如果每个 Amazon EC2 区域有 50 个以上资源，建议您为每个区域创建一个代理程序实例，或者在 EC2 实例上使用标记，并使用代理程序的过滤条件参数按您所创建的标记过滤代理程序实例。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   - 第 162 页的『在 Windows 系统上配置代理程序』.
   - 第 166 页的『使用静默响应文件来配置代理程序』.
2. 使用提示响应的脚本或静默响应文件在 Linux 系统上配置代理程序。
   - 第 165 页的『通过对提示做出响应来配置代理程序』.
   - 第 166 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Amazon EC2 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Amazon EC2** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项处于禁用状态。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。例如，`ec2-inst3`。



图 3. 用于输入唯一实例名称的窗口。

4. 在代理程序实例名称窗口上单击**下一步**。

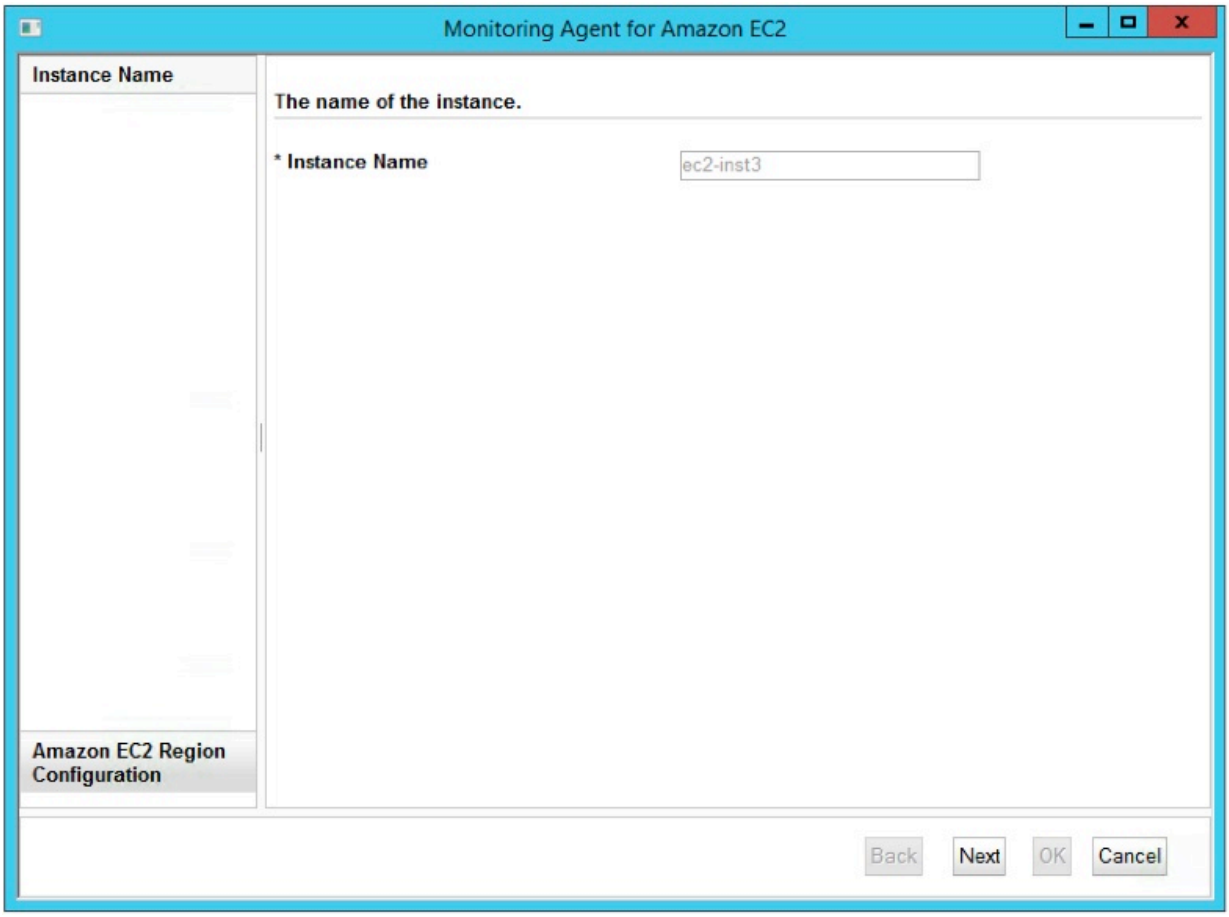


图 *4.* 代理程序实例名称窗口。

5. 输入 **Amazon EC2 区域配置**实例模板设置。

   注: 本节并非 Amazon EC2 区域实例配置。这是一个模板节，用于在步骤 6 中添加实际的 Amazon EC2 区域实例配置时要用作缺省值的设置。

   请参阅第 167 页的表 14，以获取每个配置参数的说明。

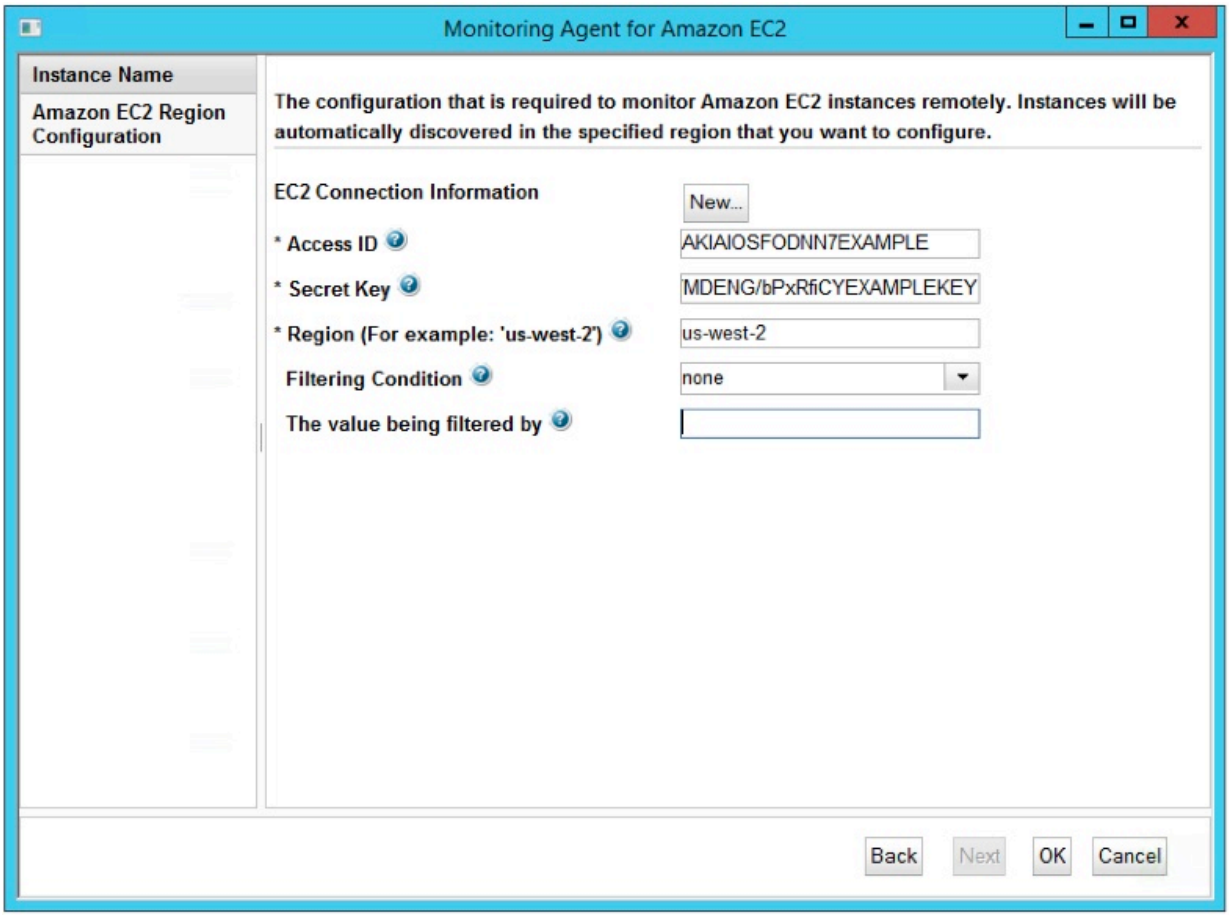


图 *5.* 用于指定 *Amazon EC2* 区域实例模板设置的窗口。

6. 按**新建**并输入 Amazon EC2 区域实例设置，然后单击**下一步**。
   请参阅：。第 167 页的表 14 以获取各个配置参数的说明。

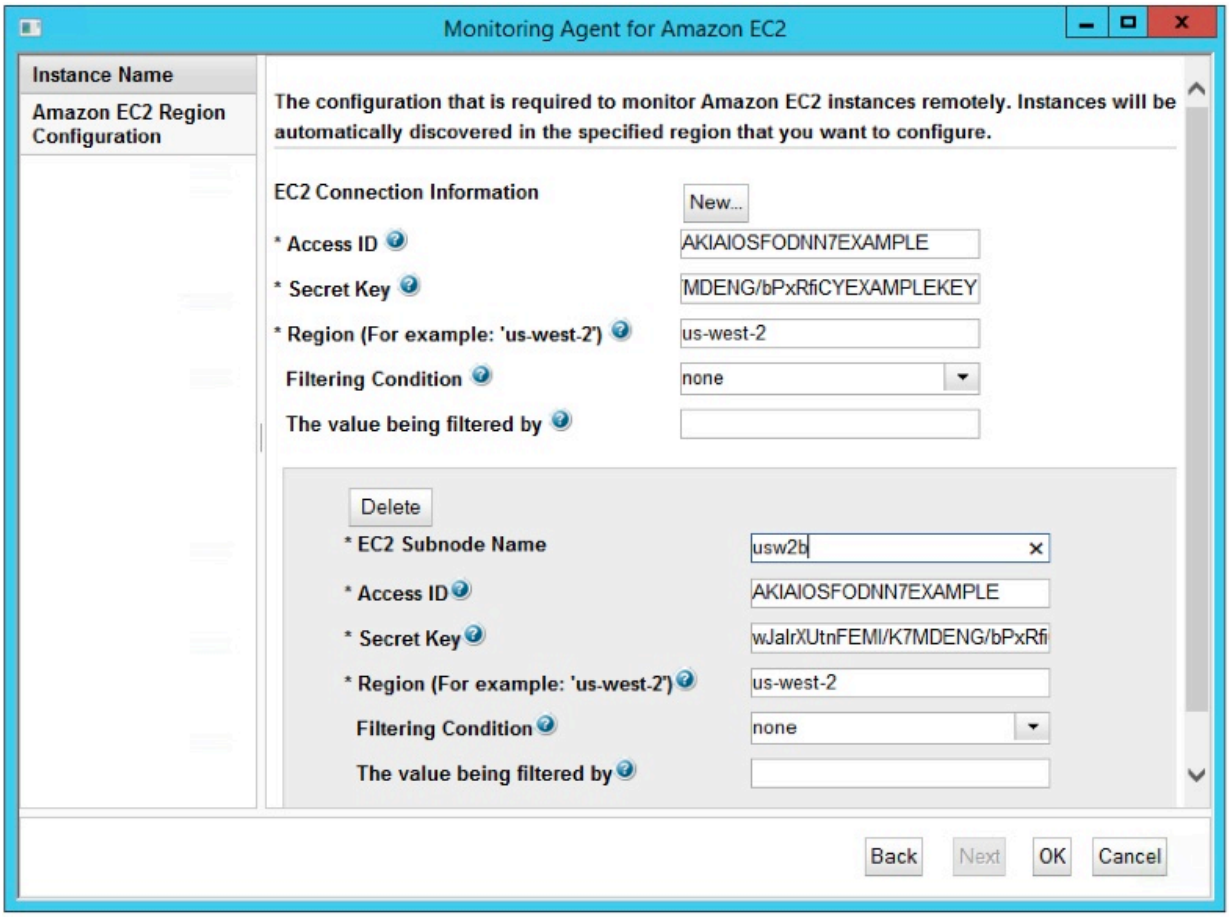


图 6. 用于指定 *Amazon EC2* 区域实例设置的窗口。

7. 单击**确定**以完成配置。
8. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过对提示做出响应来配置代理程序

安装 Amazon EC2 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 Amazon EC2 代理程序安装在本地 Linux 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 Amazon EC2 代理程序。

1. 运行以下命令:

```
install_dir/bin/amazonec2-agent.sh config instance_name
```

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/amazonec2-agent.sh config ec2-inst3
```

2. 对提示做出响应，以便为代理程序设置配置值。

请参阅第 167 页的『Amazon EC2 代理程序的配置参数』，以获取每个配置参数的说明。

3. 运行以下命令以启动代理程序：

```
install_dir/bin/amazonec2-agent.sh start instance_name
```

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/amazonec2-agent.sh start ec2-inst3
```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件来修改这些参数值，并运行配置脚本来创建实例和更新代理程序配置值。这种配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

· 以静默方式配置 Amazon EC2 代理程序：

a) 在文本编辑器中打开位于以下某个路径中的 amazonec2_silent_config.txt 文件。

– Linux *install_dir*/samples/amazonec2_silent_config.txt

例如，/opt/ibm/apm/agent/samples/amazonec2_silent_config.txt

– Windows *install_dir*\samples\amazonec2_silent_config.txt

例如，C:\IBM\APM\samples\amazonec2_silent_config.txt

其中，*install_dir* 是安装代理程序的路径。

b) 在 amazonec2_silent_config.txt 文件中，指定所有必需参数的值，并根据需要修改其他参数的缺省值。

请参阅：。第 167 页的『Amazon EC2 代理程序的配置参数』以获取各个配置参数的说明。

c) 保存并关闭 amazonec2_silent_config.txt 文件，然后运行下列命令：

– Linux *install_dir*/bin/amazonec2-agent.sh config **instance_name** *install_dir*/samples/amazonec2_silent_config.txt

例如，**/opt/ibm/apm/agent/bin/amazonec2-agent.sh config ec2-inst3 /opt/ibm/apm/agent/samples/amazonec2_silent_config.txt**

– Windows *install_dir*\bin\amazonec2-agent.bat config **instance_name** *install_dir*\samples\amazonec2_silent_config.txt

例如，**C:\IBM\APM\bin\amazonec2-agent.bat config ec2-inst3 C:\IBM\APM\samples\amazonec2_silent_config.txt**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

d) 运行以下命令以启动代理程序：

– Linux *install_dir*/bin/amazonec2-agent.sh start **instance_name**

例如，**/opt/ibm/apm/agent/bin/amazonec2-agent.sh start ec2-inst3**

– Windows *install_dir*\bin\amazonec2-agent.bat start **instance_name**

例如，**C:\IBM\APM\bin\amazonec2-agent.bat start ec2-inst3**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

Amazon EC2 代理程序的配置参数

下表显示 Amazon EC2 代理程序的配置参数。

1. Amazon EC2 区域配置 - 用于远程监视 Amazon EC2 实例的设置。系统会自动发现您所要配置的指定区域中的实例。

表 *14. Amazon EC2* 区域配置

参数名称	描述	静默配置文件参数名称
EC2 子节点名称	用于收集数据的 EC2 子节点名称。例如，*usw2a*。 此别名是受管系统名称 (MSN) 的一部分，用来在 Cloud APM 控制台中可视化识别受监视环境。 **注:** 此别名可以是您选择用来代表 Amazon EC2 子节点实例的任何内容，但有下列限制。可以使用拉丁字母表中的字母（a-z 和 A-Z）、阿拉伯数字 (0-9)、连字号/减号字符 (-) 及下划线字符 (_) 来创建代理程序子节点实例名称。EC2 子节点名称的最大长度为 25 个字符。	下列每个参数都必须具有与代理程序子节点实例的每个参数相同的代理程序子节点名称后缀。新代理程序子节点实例必须将唯一的名称用于其参数集。例如，在以下参数名称中，一个代理程序子节点实例可以使用 *.usw2a*，另一个代理程序子节点实例可以使用 *.usw2b* 来代替 *.subnode_name*。
访问标识	用来向指定的 Amazon 区域进行认证的 AWS 安全凭证访问密钥标识。例如，“AKIAxxxxxxxxxxxxxxx”。	**KB5_INS_ACCESS_ID.subnode_name**
密钥	用来向指定的 Amazon 区域进行认证的 AWS 安全凭证机密访问密钥。例如，“kK7txxxxxxxxxxxxxxxxxxxxxxxxx”。	**KB5_INS_SECRET_KEY.subnode_name**
区域	要监视的 AWS 区域。例如，“us-west-2”。	**KB5_INS_REGION.subnode_name**

表 14. Amazon EC2 区域配置 (续)		
参数名称	**描述**	**静默配置文件参数名称**
过滤条件	要执行的过滤类型。 您可在 EC2 实例上使用定制标记，来限制代理程序所监视的 EC2 实例。有关更多信息，请参阅标记您的 Amazon EC2 资源。 过滤选项： **none** 监视区域中所有的 EC2 实例。**过滤器值**将被忽略。 **tagName** 监视具有**过滤器值**中所指定标记键的 EC2 实例，而不论对应 EC2 实例标记值中的实际值为何。例如，要监视标记键为 *Stack* 的所有 EC2 实例，而不论其标记值中的值为何，请在**过滤器值**中指定 `Stack`。 **tagName|tagValue** 监视具有标记键值对的 EC2 实例，该标记键值对是以竖线 (|) 分隔，且在**过滤器值**中指定。例如，要监视具有标记键 *Stack* 和标记值 *Production* 的所有 EC2 实例，请在**过滤器值**中指定 `Stack|Production`。 **monitoring-tag** 监视至少有一个标记的 EC2 实例。**过滤器值**将被忽略。	**FILTER_CONDITION.subnode_name** 有效值 **none** none **tagName** tagName **tagName|tagValue** tagName|tagValue **monitoring-tag** monitoring-tag
过滤器值	针对**过滤条件**选择 `tagName` 或 `tagName|tagValue` 时，过滤 EC2 实例所依据的标记值。	**FILTER_VALUE.subnode_name**

配置 AWS 弹性负载均衡器监视

Amazon ELB 代理程序为您提供集中监视点，用于监视 AWS 弹性负载均衡器的运行状况、可用性和性能。代理程序显示每个负载均衡器类型（应用程序、网络和传统）的一组综合度量值，用于帮助您做出有关 AWS 弹性负载均衡器环境的明智决定。

开始之前

- 请通读第 168 页的『配置 AWS 弹性负载均衡器监视』主题，确定完成配置所需的内容。
- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 Amazon ELB 代理程序的系统需求。有关最新系统需求信息，请参阅 Amazon ELB 代理程序 的软件产品兼容性报告 (SPCR)。
- 确保您掌握下列信息：
  - 有权访问包含弹性负载均衡器的每个 AWS 区域的 AWS 安全凭证（访问密钥标识和访问密钥）。

关于此任务

Amazon ELB 代理程序既是多实例代理程序，也是子节点代理程序。自动为 AWS 环境中提供的每种弹性负载均衡器创建子节点。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   - 第 169 页的『在 Windows 系统上配置代理程序』.
   - 第 171 页的『使用静默响应文件来配置代理程序』.
2. 使用提示响应的脚本或静默响应文件在 Linux 系统上配置代理程序。
   - 第 170 页的『通过响应提示来配置代理程序』.
   - 第 171 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Amazon ELB 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for AWS 弹性负载均衡器** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项处于禁用状态。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。例如，`elb-inst3`。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

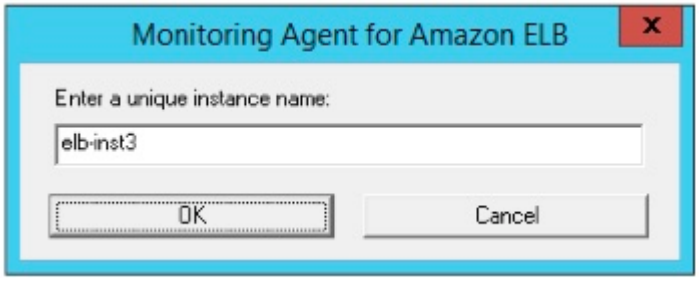


图 7. 用于输入唯一代理程序实例名称的窗口。

4. 输入 **Amazon ELB 预订凭证**，然后单击**下一步**。

   请参阅：。第 172 页的『Amazon ELB 代理程序的配置参数』以获取各个配置参数的说明。

   要点: Windows 如果您的**密钥/密码**包含等号 (=)，那么您每次重新配置代理程序时，都必须重新输入该密钥/密码。

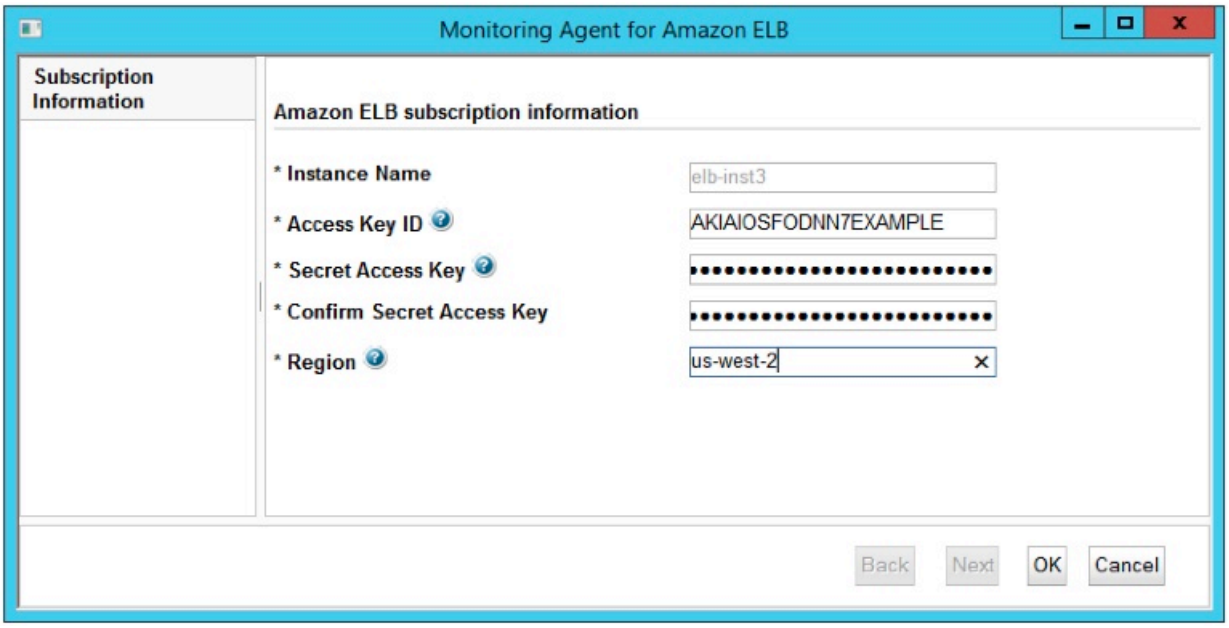


图 *8. Amazon ELB* 预订凭证窗口。

5. 单击**确定**以完成配置。
6. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过响应提示来配置代理程序

安装 Amazon ELB 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 Amazon ELB 代理程序安装在本地 Linux 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 Amazon ELB 代理程序。

1. 运行以下命令:

   ```
   install_dir/bin/amazon_elb-agent.sh config instance-name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance-name* 是要指定给代理程序实例的名称。在 *instance-name* 中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

   示例

   ```
   /opt/ibm/apm/agent/bin/amazon_elb-agent.sh config elb-inst3
   ```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 172 页的『Amazon ELB 代理程序的配置参数』，以获取每个配置参数的说明。

3. 运行以下命令以启动代理程序:

   ```
   install_dir/bin/amazon_elb-agent.sh start instance-name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance-name* 是代理程序实例的名称。

   示例

```
/opt/ibm/apm/agent/bin/amazon_elb-agent.sh start elb-inst3
```

示例

创建名为 elb-inst3 的代理程序实例。

```
# ./amazon_elb-agent.sh config elb-inst3
Configuring Monitoring Agent for Amazon ELB

Edit 'Monitoring Agent for Amazon ELB' settings? [ 1=Yes, 2=No ]  (default is: 1): 1

Subscription Information :
Amazon ELB subscription information

The access ID that is used to authenticate with the specified Amazon Region.
For example, 'AKIAxxxxxxxxxxxxxx'.
Access Key ID (default is: ): AKIAIOSFODNN7EXAMPLE

The secret access key that is used to authenticate with the specified Amazon
Region. For example, 'kK7txxxxxxxxxxxxxxxxxxxxxxxx'.
Enter Secret Access Key (default is: ): hidden

Re-type : Secret Access Key (default is: ): hidden

The Amazon region where the load balancers are located. For example, 'us-west-2'.
Region (default is: ): us-west-2

Configuration completed successfully.
Automatic start at system initialization has been configured.
Automatic stop at system shutdown has been configured.
```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

要以静默方式配置 Amazon ELB 代理程序，请完成下列步骤。

1. 在文本编辑器中打开位于以下某个路径中的 amazon_elb_silent_config.txt 文件。
   - Linux *install_dir*/samples/amazon_elb_silent_config.txt

     例如， /opt/ibm/apm/agent/samples/amazon_elb_silent_config.txt
   - Windows *install_dir*\samples\amazon_elb_silent_config.txt

     例如，C:\IBM\APM\samples\amazon_elb_silent_config.txt

   其中，*install_dir* 是代理程序的安装路径。
2. 在 amazon_elb_silent_config.txt 文件中，指定所有必需参数的值，并根据需要修改其他参数的缺省值。

   请参阅：。第 172 页的『Amazon ELB 代理程序的配置参数』以获取各个配置参数的说明。
3. 保存并关闭 amazon_elb_silent_config.txt 文件，然后运行下列命令：
   - Linux *install_dir*/bin/amazon_elb-agent.sh config **instance-name** *install_dir*/samples/amazon_elb_silent_config.txt

     例如，**/opt/ibm/apm/agent/bin/amazon_elb-agent.sh config elb-inst3 /opt/ibm/apm/agent/samples/amazon_elb_silent_config.txt**

- Windows *install_dir*\bin\amazon_elb-agent.bat config **instance-name** *install_dir*\samples\amazon_elb_silent_config.txt

  例如，**C:\IBM\APM\bin\amazon_elb-agent.bat config elb-inst3 C:\IBM\APM\samples\amazon_elb_silent_config.txt**

其中，*install_dir* 是代理程序的安装路径，*instance-name* 是要指定给代理程序实例的名称。在 *instance-name* 中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

4. 运行以下命令以启动代理程序：

   - Linux *install_dir*/bin/amazon_elb-agent.sh start **instance-name**

     例如，**/opt/ibm/apm/agent/bin/amazon_elb-agent.sh start elb-inst3**

   - Windows *install_dir*\bin\amazon_elb-agent.bat start **instance-name**

     例如，**C:\IBM\APM\bin\amazon_elb-agent.bat start elb-inst3**

   其中，*install_dir* 是代理程序的安装路径，*instance-name* 是代理程序实例的名称。

示例

已编辑 amazon_elb_silent_config.txt。

```
#
# This is a sample configuration response file for agent Amazon ELB.
#
# It contains an entry for every configuration property.
# Entries for optional properties that have no default value are included
# in comments.
# Ensure that all uncommented properties have a value before configuring
# the agent.
#

# Access Key ID: The access ID that is used to authenticate with the
# specified Amazon Region. For example, 'AKIAxxxxxxxxxxxxxx'.
KAL_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE

# Secret Access Key: The secret access key that is used to authenticate with
# the specified Amazon Region. For example, 'kK7txxxxxxxxxxxxxxxxxxxxxxxxx'.
KAL_SECRET_ACCESS_KEY_PASSWORD=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

# Region: The Amazon region where the load balancers are located. For
# example, 'us-west-2'.
KAL_REGION=us-west-2
```

Amazon ELB 代理程序的配置参数

下表显示 Amazon ELB 代理程序的配置参数。

1. 第 172 页的表 15 - 访问包含要监视的 AWS 弹性负载均衡器的 Amazon 预订所需的凭证。

表 *15.* 预订信息

参数名称	描述	静默配置文件参数名称
访问密钥标识	用于向指定的 Amazon 区域进行认证的访问标识。例如，“AKIAxxxxxxxxxxxxxx”。	**KAL_ACCESS_KEY_ID**
访问密钥	用于向指定的 Amazon 区域进行认证的访问密钥。例如，“kK7txxxxxxxxxxxxxxxxxxxxxxxxx”。	**KAL_SECRET_ACCESS_KEY_PASSWORD**
区域	负载均衡器所在的 Amazon 区域。例如，“us-west-2”。	**KAL_REGION**

配置 Azure Compute 监视

Azure Compute 代理程序为您提供集中监视点，用于监视 Azure Compute 实例的运行状况、可用性和性能。此代理程序会显示一组综合度量值，帮助您做出有关 Azure Compute 环境的明智决定。这些度量包括 CPU 使用率、网络使用率和磁盘性能。

开始之前

- 请通读第 173 页的『配置 Azure Compute 监视』主题，确定完成配置所需的内容。
- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 Azure Compute 代理程序的系统需求。有关最新系统需求信息，请参阅 Azure Compute 代理程序的软件产品兼容性报告 (SPCR)。
- 确保您掌握下列信息：
  - 有权访问所要监视的 Azure Compute 实例的 Azure 预订凭证。请参阅第 173 页的『Azure Compute 配置信息』以获取更多详细信息。

关于此任务

Azure Compute 代理程序既是多实例代理程序，也是子节点代理程序。每个 Azure Compute 代理程序 子节点都根据您定义的过滤器来监视一组 Azure Compute 虚拟机。您可以创建一个具有多个子节点（每个虚拟机分组对应于一个子节点）的代理程序实例，也可以为每个虚拟机分组创建一个代理程序实例（一个子节点对应于该分组）。或者可以创建各种类型配置的组合。配置代理程序实例后，必须手动启动每个代理程序实例。建议每个 Azure Compute 虚拟机分组的资源数不要超过 50。每个 Azure Compute 代理程序子节点名称在环境中都必须是唯一的。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   - 第 174 页的『在 Windows 系统上配置代理程序』.
   - 第 179 页的『使用静默响应文件来配置代理程序』.
2. 使用提示响应的脚本或静默响应文件在 Linux 系统上配置代理程序。
   - 第 177 页的『通过响应提示来配置代理程序』.
   - 第 179 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

Azure Compute 配置信息

Azure Compute 代理程序 在 Azure Compute 环境中需要一些额外的设置。

关于此任务

要运行这些步骤，您必须登录到 Microsoft Azure 控制台。

过程

1. 预订标识
   - 在左侧窗格上，选择“预订”，然后选择想要用于此代理程序的预订。
   - 选择“概述”，然后复制预订标识。这将用作代理程序的配置参数之一。
2. 租户标识
   - 浏览至“Azure Active 目录”
   - 选择“属性”，然后复制租户标识。
3. 注册应用程序
   - 转至“所有服务”并搜索“应用程序注册”。
   - 单击“新建应用程序注册”。
   - 填写名称，选择应用程序类型“Web 应用程序/API”和登录 URL（将不使用此 URL，从而选择您喜爱的 URL）。
   - 单击“创建”
   - 复制应用程序标识 - 这将用于代理程序的“客户机标识”字段。
4. 创建应用程序密钥
   - 单击刚刚创建的应用程序，然后转至“设置”，随后是“密钥”。
   - 输入描述（例如，“IBM 密钥”）和持续时间（例如，“从不到期”），然后单击“保存”。
   - 复制密钥并存储在安全位置 - 您只能看到此密钥一次，并且需要此密钥才能生成新密钥（如果丢失）。
5. 指定应用程序权限
   - 转至“预订”，选择要监视的预订。
   - 转至“访问控制”(IAM)，然后单击“添加”。
   - 选择“读者”角色或更高角色以用于监视。
   - 在“选择”下，查找刚刚注册的应用程序并选择，然后单击“保存”。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Azure Compute 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Azure Compute** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项处于禁用状态。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。例如，`azc-inst3`。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

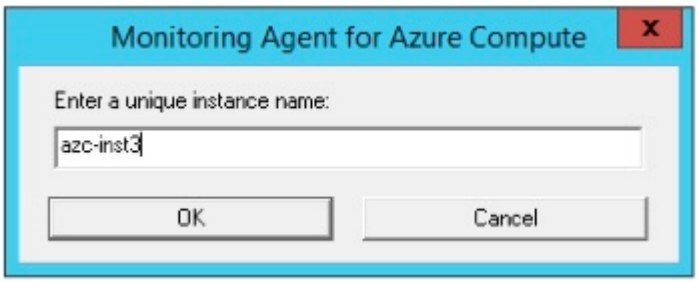


图 *9.* 用于输入唯一代理程序实例名称的窗口。

4. 输入 **Azure 预订凭证**，然后单击**下一步**。
   请参阅第 181 页的表 16，以获取每个配置参数的说明。

   要点: Windows 如果您的**密钥/密码**包含等号 (=)，那么您每次重新配置代理程序时，都必须重新输入该密钥/密码。

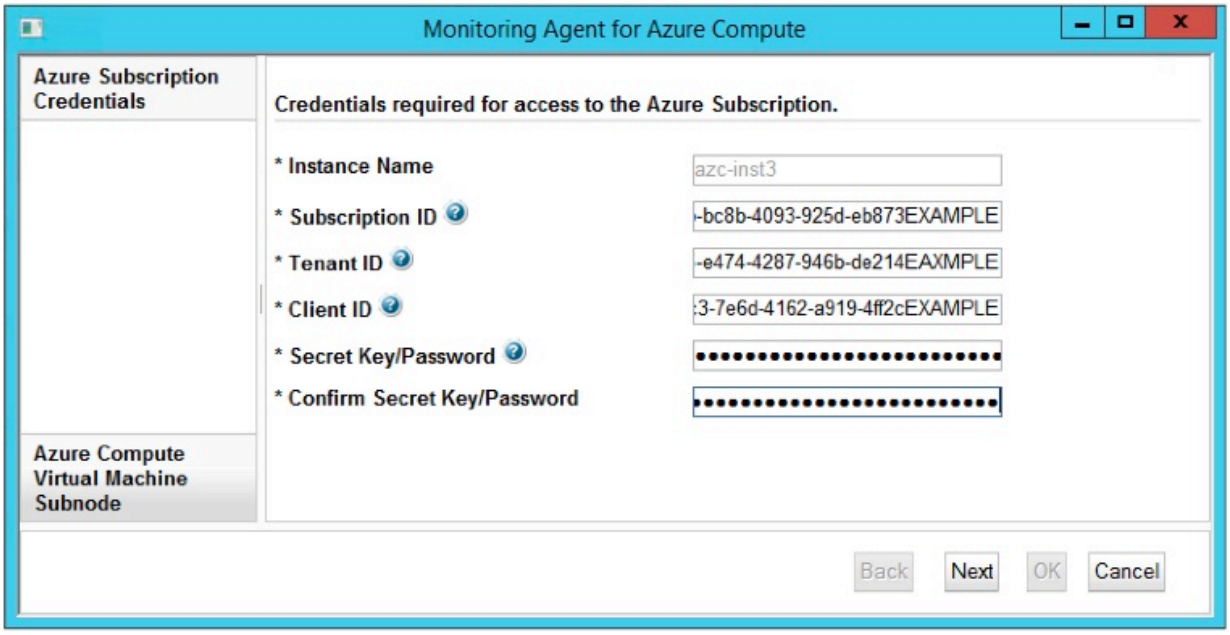


图 *10. Azure* 预订凭证窗口。

5. 输入 **Azure Compute 虚拟机子节点**模板设置。
   请参阅第 181 页的表 17，以获取每个配置参数的说明。

   注: 此节不是“Azure Compute 虚拟机子节点”实例配置。这是一个模板节，用于设置当您在步骤 6 中添加实际的 Azure Compute 虚拟机子节点实例配置时，要用作缺省值的内容。

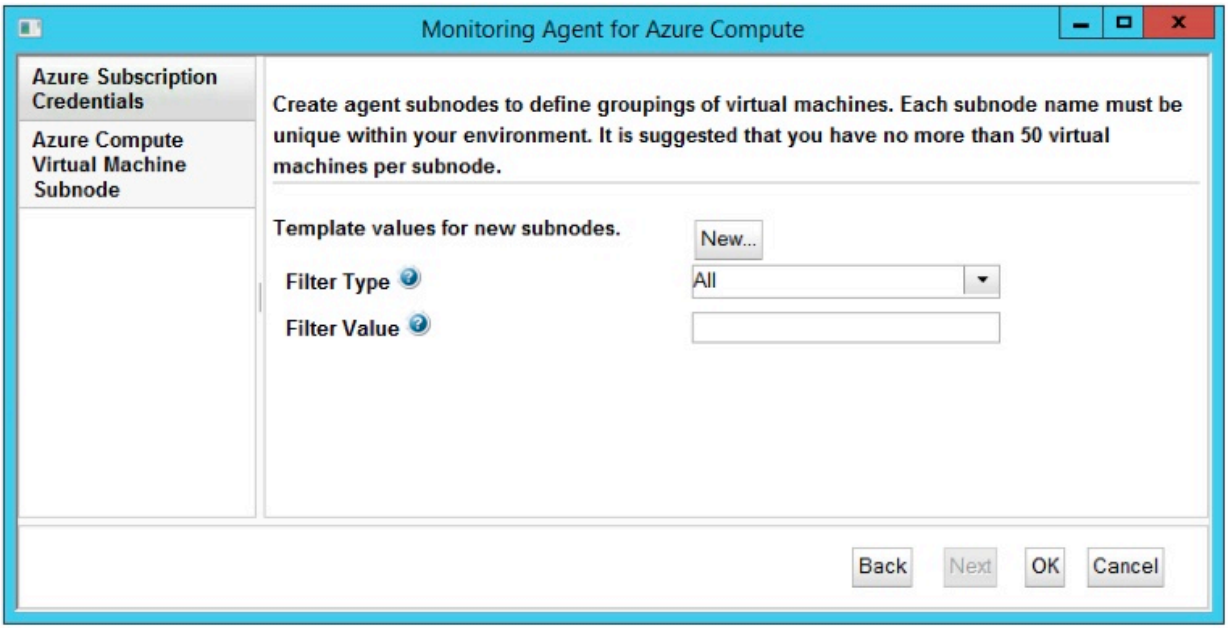


图 *11.* 用于指定 *Azure Compute* 虚拟机子节点模板设置的窗口。

6. 按**新建**并输入 **Azure Compute 虚拟机子节点**实例设置，然后单击**下一步**。
   请参阅第 181 页的表 17，以获取每个配置参数的说明。

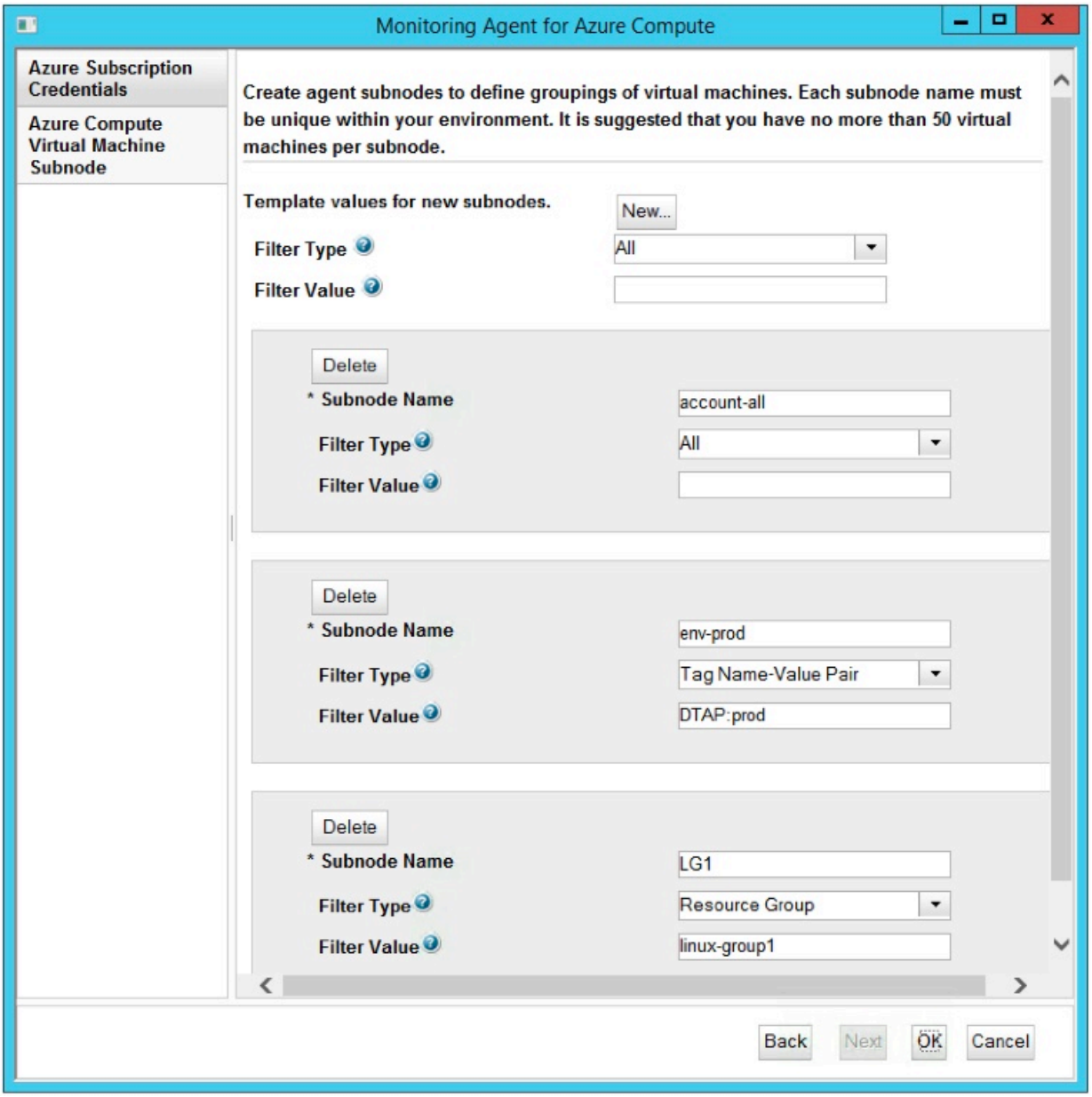


图 *12.* 用于指定 *Azure Compute* 虚拟机子节点实例设置的窗口。

7. 单击**确定**以完成配置。
8. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过响应提示来配置代理程序

安装 Azure Compute 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 Azure Compute 代理程序安装在本地 Linux 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 Azure Compute 代理程序。

1. 运行以下命令：

```
install_dir/bin/azure_compute-agent.sh config instance-name
```

其中，*install_dir* 是代理程序的安装路径，*instance-name* 是要指定给代理程序实例的名称。在 *instance-name* 中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

示例

```
/opt/ibm/apm/agent/bin/azure_compute-agent.sh config azc-inst3
```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 181 页的『Azure Compute 代理程序的配置参数』，以获取每个配置参数的说明。

   切记: 首次配置代理程序实例期间，出现 **Edit 'Azure Compute Virtual Machine Subnode' settings** 提示时，必须至少添加一个子节点。
3. 运行以下命令以启动代理程序：

```
install_dir/bin/azure_compute-agent.sh start instance-name
```

其中，*install_dir* 是代理程序的安装路径，*instance-name* 是代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/azure_compute-agent.sh start azc-inst3
```

示例

创建名为 azc-inst3 且具有一个子节点实例 azc1 的代理程序实例。

```
# ./azure_compute-agent.sh config azc-inst3
Configuring Monitoring Agent for Azure Compute

Edit 'Monitoring Agent for Azure Compute' settings? [ 1=Yes, 2=No ]  (default is: 1): 1

Azure Subscription Credentials :
Credentials required for access to the Azure Subscription.

The ID assigned by Azure for the Subscription that is monitored.
Subscription ID (default is: ): 09x73b6b-bcxb-40x3-92xd-ebx7-EXAMPLE

The tenant ID that is assigned by Azure. Used to log in to the Azure service API.
Tenant ID (default is: ): 75x2e745-e4x4-42x7-94xb-dex1-EXAMPLE

The client ID that is assigned by Azure to identify this agent as an external
application that monitors the Azure compute services.
Client ID (default is: ): 79x2e6c3-7exd-41x2-a9x9-4fx2-EXAMPLE

The secret access key or password that is created by Azure for the client application.
Enter Secret Key/Password (default is: ): hidden

Re-type : Secret Key/Password (default is: ): hidden

Azure Compute Virtual Machine Subnode :

Create agent subnodes to define groupings of virtual machines. Each subnode name
must be unique within your environment. It is suggested that you have no more
than 50 virtual machines per subnode.

No 'Azure Compute Virtual Machine Subnode' settings available.
Edit 'Azure Compute Virtual Machine Subnode' settings, [1=Add, 2=Edit, 3=Del,
4=Next, 5=Exit] (default is: 5): 1
Subnode Name (default is: ): azc1
The type of filter to be applied.
Filter Type [ 1=All, 2=Tag Name-Value Pair, 3=Resource Group ] (default is: 1): 2
The filter value corresponding to the selected Filter Type. This value can be a
Resource Group or Tag Name-Value Pair, for example Environment\:Production.
A backslash might appear in the example, do not enter a backslash in the value
you provide.
Filter Value (default is: ): Environment:Production
```

```
Azure Compute Virtual Machine Subnode settings: Subnode Name=azc1
Edit 'Azure Compute Virtual Machine Subnode' settings, [1=Add, 2=Edit, 3=Del,
4=Next, 5=Exit] (default is: 5): 5
Configuration completed successfully.
Automatic start at system initialization has been configured.
Automatic stop at system shutdown has been configured.
You have new mail in /var/spool/mail/root
```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

要以静默方式配置 Azure Compute 代理程序，请完成下列步骤。

1. 在文本编辑器中打开位于以下某个路径中的 azure_compute_silent_config.txt 文件。
   - Linux *install_dir*/samples/azure_compute_silent_config.txt

     例如，/opt/ibm/apm/agent/samples/azure_compute_silent_config.txt
   - Windows *install_dir*\samples\azure_compute_silent_config.txt

     例如，C:\IBM\APM\samples\azure_compute_silent_config.txt

   其中，*install_dir* 是代理程序的安装路径。
2. 在 azure_compute_silent_config.txt 文件中，指定所有必需参数的值，并根据需要修改其他参数的缺省值。

   请参阅第 181 页的『Azure Compute 代理程序的配置参数』，以获取每个配置参数的说明。

   要点: 必须至少对一个子节点名称启用并指定“过滤器类型”和“过滤器值”参数。
3. 保存并关闭 azure_compute_silent_config.txt 文件，然后运行下列命令:
   - Linux *install_dir*/bin/azure_compute-agent.sh config **instance-name** *install_dir*/samples/azure_compute_silent_config.txt

     例如，**/opt/ibm/apm/agent/bin/azure_compute-agent.sh config azc-inst3 /opt/ibm/apm/agent/samples/azure_compute_silent_config.txt**
   - Windows *install_dir*\bin\azure_compute-agent.bat config **instance-name** *install_dir*\samples\azure_compute_silent_config.txt

     例如，**C:\IBM\APM\bin\azure_compute-agent.bat config azc-inst3 C:\IBM\APM\samples\azure_compute_silent_config.txt**

   其中，*install_dir* 是代理程序的安装路径，*instance-name* 是要指定给代理程序实例的名称。在 *instance-name* 中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。有关更多信息，请参阅第 143 页的『多实例代理程序的一般 MSN 格式』下的 *instancename*。

   要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。
4. 运行以下命令以启动代理程序:
   - Linux *install_dir*/bin/azure_compute-agent.sh start **instance-name**

     例如，**/opt/ibm/apm/agent/bin/azure_compute-agent.sh start azc-inst3**
   - Windows *install_dir*\bin\azure_compute-agent.bat start **instance-name**

例如，**C:\IBM\APM\bin\azure_compute-agent.bat start azc-inst3**

其中，*install_dir* 是代理程序的安装路径，*instance-name* 是代理程序实例的名称。

示例

以下是编辑后的 azure_compute_silent_config.txt，其中有三个子节点，它们分别名为 account-all、env-prod 和 LG1。

```
#
# This is a sample configuration response file for agent Azure Compute.
#
# It contains an entry for every configuration property.
# Entries for optional properties that have no default value are included in
# comments.
# Entries for subnode AVM are given a sample subnode instance name of avm1.
# Ensure that all uncommented properties have a value before configuring the
# agent.
#

# Subscription ID: The ID assigned by Azure for the Subscription that is
# monitored.
KAK_SUBSCRIPTION_ID=09x73b6b-bcxb-40x3-92xd-ebx7-EXAMPLE
# Tenant ID: The tenant ID that is assigned by Azure. Used to log in to the
# Azure service API.
KAK_TENANT_ID=75x2e745-e4x4-42x7-94xb-dex1-EXAMPLE
# Client ID: The client ID that is assigned by Azure to identify this agent
# as an external
# application that monitors the Azure compute services.
KAK_CLIENT_ID=79x2e6c3-7exd-41x2-a9x9-4fx2-EXAMPLE
# Secret Key/Password: The secret access key or password that is created by
# Azure for the client application.
KAK_SECRET_PASSWORD=hZxWPq/IOxlnvg/wdxLwTf2Fs3x2sWQV/sCE-EXAMPLE

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
#KAK_FILTER_TYPE.avm1=ALL
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
#KAK_FILTER_VALUE.avm1=

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.account-all=ALL
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.account-all=

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.env-prod=TAG_NAME_VALUE
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.env-prod=DTAP:prod

# Filter Type: The type of filter to be applied.
# Valid values: ALL (All), TAG_NAME_VALUE (Tag Name-Value Pair),
# RESOURCE_GROUP (Resource Group)
KAK_FILTER_TYPE.LG1=RESOURCE_GROUP
# Filter Value: The filter value corresponding to the selected Filter Type.
# This value can be a Resource Group or Tag Name-Value Pair, for example
# Environment:Production. A backslash might appear in the example, do not
# enter a backslash in the value you provide.
KAK_FILTER_VALUE.LG1=linux-group1
```

Azure Compute 代理程序的配置参数

下列各表显示 Azure Compute 代理程序的配置参数，并根据小节进行分组。

1. 第 181 页的表 16 - 访问包含所要监视的 Azure Compute 资源的 Azure 预订所需的凭证。
2. 第 181 页的表 17 - 创建代理程序子节点，以定义虚拟机分组。每个子节点名称在环境中都必须是唯一的。建议您不要在单个子节点上运行 50 个以上虚拟机。

表 *16. Azure* 预订凭证

参数名称	描述	静默配置文件参数名称
预订标识	Azure 分配给所监视预订的标识。	**KAK_SUBSCRIPTION_ID**
租户标识	Azure 所分配的租户标识。用于登录 Azure 服务 API。	**KAK_TENANT_ID**
客户机标识	Azure 所分配的客户机标识，用于将此代理程序识别为监视 Azure Compute 服务的外部应用程序。	**KAK_CLIENT_ID**
密钥/密码	Azure 为客户机应用程序创建的访问密钥或密码。 **要点:** Windows 如果您的**密钥/密码**包含等号 (=)，那么您每次重新配置代理程序时，都必须重新输入该密钥/密码。	**KAK_SECRET_PASSWORD**

表 *17. Azure Compute* 虚拟机子节点

参数名称	描述	静默配置文件参数名称
子节点名称	用于收集数据的 Azure Compute 子节点名称。例如，*azc1*。子节点名称在环境中必须是唯一的。 此别名是受管系统名称 (MSN) 的一部分，用来在 Cloud APM 控制台中可视化识别受监视环境。 **注:** 此别名可以是您选择用来代表 Azure Compute 子节点实例的任何内容，但有下列限制。可以使用拉丁字母表中的字母（a-z 和 A-Z）、阿拉伯数字 (0-9)、连字号/减号字符 (-) 及下划线字符 (_) 来创建代理程序子节点实例名称。Azure Compute 子节点名称的最大长度为 25 个字符。	下列每个参数都必须使用句点 (.)，后跟代理程序**子节点名称**作为后缀。各个子节点参数的**子节点名称**必须相同。新代理程序子节点实例必须将唯一的**子节点名称**用于其参数集。例如，在以下参数名称中，一个代理程序子节点实例可以使用 *.azc1*，另一个代理程序子节点实例可以使用 *.azc2* 来代替 *.subnode_name*。
过滤器类型	要应用的过滤器类型。	**KAK_FILTER_TYPE.subnode_name** 有效值 **ALL** 全部 **TAG_NAME_VALUE** 标记名称/值对 **RESOURCE_GROUP** 资源组

表 17. *Azure Compute* 虚拟机子节点 *(续)*		
参数名称	**描述**	**静默配置文件参数名称**
过滤器值	对应于所选**过滤器类型**的过滤器值。此值可以是**资源组**或**标记名称/值对**。当**过滤器类型**为**全部**时，请将过滤器值保留为空。对于命令行配置，在显示的示例中可能会出现反斜杠。在您提供的值中，请勿输入反斜杠。 过滤器类型/过滤器值对的示例： · 要监视所有虚拟机的 Azure Compute 子节点：请保留过滤器值为空。过滤器类型**全部**不需要过滤器值，将会忽略该值。 – 过滤器类型：**全部** – 过滤器值： · 要监视所有符合以下条件的虚拟机的 Azure Compute 子节点：具有标记名称 DTAP，且标记值与字符串 prod 匹配。 – 过滤器类型：**标记名称/值对** – 过滤器值：`DTAP:prod` · 要监视所有符合以下条件的虚拟机的 Azure Compute 子节点：具有与字符串 `linux-group1` 匹配的资源组属性。 – 过滤器类型：**资源组** – 过滤器值：`linux-group1`	**KAK_FILTER_VALUE.subnode_name**

配置 Cassandra 监视

您必须配置 Cassandra 代理程序以便代理程序可以从集群中的节点收集数据，以监视 Cassandra 数据库的运行状况。

开始之前

查看硬件和软件先决条件，请参阅 Cassandra 代理程序的软件产品兼容性报告。

关于此任务

Cassandra 代理程序是多实例代理程序。您必须创建第一个实例，然后手动启动代理程序。

产品版本与代理程序版本通常不相同。以下指南针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

· 要在 Windows 系统上配置此代理程序，您可以使用 IBM Cloud Application Performance Management 窗口或静默响应文件。

· 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示进行响应，或使用静默响应文件。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。

2. 在 **IBM Performance Management** 窗口的**任务/子系统**列中，右键单击**模板**，然后单击**使用缺省值配置**。

   这将打开 **Monitoring Agent for Cassandra** 窗口。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。
4. 在 **Monitoring Agent for Cassandra** 窗口中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 184 页的『代理程序的配置参数』。
5. 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks® 上的 IBM Cloud APM 论坛。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示进行响应。

过程

1. 在命令行上，将路径切换到代理程序安装目录。
   示例：`/opt/ibm/apm/agent/bin`
2. 运行以下命令，其中 instance_name 是要对实例指定的名称：

   ```
   ./cassandra-agent.sh config instance_name
   ```
3. 当命令行显示以下消息时，请输入 1，并输入：

   ```
   Edit 'Monitoring Agent for Cassandra' setting? [1=Yes, 2=No]
   ```
4. 出现提示时，指定配置参数的值。

   有关配置参数的信息，请参阅第 184 页的『代理程序的配置参数』。
5. 运行以下命令以启动代理程序：

   ```
   ./cassandra-agent.sh start instance_name
   ```

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件来修改这些参数的值，并运行配置脚本来创建实例并更新代理程序配置值。这种配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可使用静默响应文件来配置 Cassandra 代理程序。在静默响应文件中更新配置值之后，必须运行配置脚本，以使用这些更新后的值来配置代理程序。

过程

1. 在文本编辑器中，打开位于以下位置的静默配置文件，并对所有参数指定值：

   Windows `install_dir\samples\cassandra_silent_config_windows.txt`

   Linux `install_dir\samples\cassandra_silent_config_UNIX.txt`

   Windows `C:\IBM\APM\samples`

   Linux `/opt/ibm/apm/agent/samples`

有关配置参数的信息，请参阅第 184 页的『代理程序的配置参数』。

2. 在命令行中，将路径更改为 *install_dir*\bin
3. 运行以下命令：

   Windows `cassandra-agent.bat config instance_name install_dir\samples\cassandra_silent_config_windows.txt`

   Linux `cassandra-agent.sh config instance_name install_dir\samples\cassandra_silent_config_UNIX.txt`
4. 启动代理程序。

   Windows 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

   Linux 运行以下命令：`./cassandra-agent.sh start instance_name`

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

代理程序的配置参数

配置 Cassandra 代理程序时，您可以更改参数的缺省值，例如 IP 地址和 JMX_PORT。

下表包含 Cassandra 代理程序 配置参数的详细描述。

表 *18.* 配置参数的名称和描述

参数名称	描述	必填字段
实例名称	此字段的缺省值与**输入唯一实例名称**字段中指定的值相同。	是
IP 地址	集群中任何节点的 IP 地址。	是
JMX_PORT	用于启用监视的 JMX 端口号。 **要点:** 请确保指定的“JMX 端口”、“JMX 用户名”和“JMX 密码”在整个集群中相同。供代理程序用来连接至集群的节点不工作时，代理程序可使用相同的参数，通过集群中的另一个节点来收集数据。	是
JMX_Username	用于访问 JMX 的用户名。	否
JMX_Password	用于访问 JMX 的密码。	否

配置 Cisco UCS 监视

Monitoring Agent for Cisco UCS 会连接到 Cisco UCSM，以监视 Cisco UCS Virtual Infrastructure。您必须配置 Cisco UCS agent，以便代理程序能够收集 Cisco UCS 数据。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 Cisco UCS agent 的软件产品兼容性报告 (SPCR)。
- 确保连接到 Cisco UCSM 基础结构的用户具有 aaa 或管理员权限。请使用具有 aaa 或管理员权限的现有用户标识，或者创建新用户标识。
- 如果 Cisco UCS agent 配置为与其使用 SSL 代理程序的 Cisco UCS 数据源进行通信，请将每个数据源的 SSL 证书添加到代理程序的证书信任库。有关启用与 Cisco UCS 数据源的 SSL 通信的更多信息，请参阅第 188 页的『启用与 Cisco UCS 数据源的 SSL 通信』。

关于此任务

Cisco UCS agent 是多实例代理程序。您必须创建第一个实例，然后手动启动代理程序。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅代理程序版本。要访问先前代理程序发行版的文档，请参阅下表：

表 *19.* 代理程序版本和文档

Cisco UCS agent 版本	**文档**
7.2.0.4 和 7.2.0.3	IBM Cloud Application Performance Management **注:** 该链接会打开本地 Knowledge Center 主题。
7.2.0.2	IBM Performance Management 8.1.3 **注:** 该链接会打开本地 Knowledge Center 主题。
7.2.0.1	IBM Performance Management 8.1.2 **注:** 该链接会打开本地 Knowledge Center 主题。

配置属性定义要监视哪个 Cisco UCS 基础结构。这些属性定义与 Cisco UCSM 1.4 或更高版本的连接。可以在远程监视主机系统上配置监视代理程序的多个实例。您还可以创建单独的实例以监视特定 Cisco UCS 基础结构。

安装 Cisco UCS agent 之后，即可启动代理程序。但是，必须手动配置代理程序才能查看所有代理程序属性的数据。

- 要在 Windows 系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。
- 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

在 Windows 系统上配置代理程序

在 Windows 操作系统上，您可以使用 **IBM Performance Management** 窗口来配置代理程序。更新配置值之后，必须启动代理程序以保存更新后的值。

关于此任务

Cisco UCS agent 会提供一些参数的缺省值。您可以为这些参数指定其他值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Cisco UCS**，然后单击**配置代理程序**。

   切记: 首次配置代理程序之后，**配置代理程序**选项处于禁用状态。要再次配置代理程序，请单击**重新配置**。
3. 在 Monitoring Agent for Cisco UCS 窗口中，完成下列步骤：
   a) 为 Cisco UCS agent 实例输入唯一名称，然后单击**确定**。
   b) 在 **CONFIG** 选项卡上，为配置参数指定值，然后单击**下一步**。
   c) 在 **LOG_CONFIG** 选项卡上，为配置参数指定值，然后单击**下一步**。

   有关“Monitoring Agent for Cisco UCS”窗口的每个选项卡中配置参数的信息，请参阅以下主题：
   - 第 187 页的『代理程序的配置参数』
   - 第 188 页的『数据提供程序的配置参数』
4. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Cisco UCS**，然后单击**启动**。

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

  如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。
- 如果要监视大型 Cisco UCS 环境，您可能需要增加 Java™ 数据提供程序的堆大小。有关更多信息，请参阅第 189 页的『增加 Java 堆大小』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件来修改这些参数的值，并运行配置脚本来创建实例并更新代理程序配置值。这种配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便为配置参数指定其他值。

在静默响应文件中更新配置值之后，必须运行配置脚本，以使用这些更新后的值来配置代理程序。

过程

- 要以静默方式配置 Cisco UCS agent，请完成下列步骤：

  a) 在文本编辑器中，打开位于以下路径的 `cisco_ucs_silent_config.txt` 文件：

  - Linux *install_dir*`/samples/cisco_ucs_silent_config.txt`

    示例：`/opt/ibm/apm/agent/samples/cisco_ucs_silent_config.txt`
  - Windows *install_dir*`\samples\cisco_ucs_silent_config.txt`

    示例：`C:\IBM\APM\samples\cisco_ucs_silent_config.txt`

  b) 在 `cisco_ucs_silent_config.txt` 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

  有关配置参数的信息，请参阅以下主题：

  - 第 187 页的『代理程序的配置参数』
  - 第 188 页的『数据提供程序的配置参数』

  c) 保存并关闭 `cisco_ucs_silent_config.txt` 文件，然后运行以下命令：

  - Linux *install_dir*`/bin/cisco_ucs-agent.sh config` *instance_name* *install_dir*`/samples/cisco_ucs_silent_config.txt`

    示例：**`/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh config instance_name /opt/ibm/apm/agent/samples/cisco_ucs_silent_config.txt`**
  - Windows *install_dir*`\bin\cisco_ucs-agent.bat config` *instance_name* *install_dir*`\samples\cisco_ucs_silent_config.txt`

    示例：**`C:\IBM\APM\bin\cisco_ucs-agent.bat config instance_name C:\IBM\APM\samples\cisco_ucs_silent_config.txt`**

    其中

    instance_name
    要对实例指定的名称。

    install_dir
    代理程序的安装路径。

  要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

  d) 运行以下命令以启动代理程序：

– Linux *install_dir*/bin/cisco_ucs-agent.sh start *instance_name*

  示例：**/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh start instance_name**

– Windows *install_dir*\bin\cisco_ucs-agent.bat start *instance_name*

  示例：**C:\IBM\APM\bin\cisco_ucs-agent.bat start instance_name**

下一步做什么

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

  如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

· 如果要监视大型 Cisco UCS 环境，您可能需要增加 Java™ 数据提供程序的堆大小。有关更多信息，请参阅第 189 页的『增加 Java 堆大小』。

通过响应提示来配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示进行响应。

过程

· 要通过运行脚本并响应提示来配置代理程序，请完成下列步骤：

a) 在命令行上，输入以下命令：

install_dir/bin/cisco_ucs-agent.sh config *instance_name*

示例：**/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh config instance_name**

其中

instance_name
要对实例指定的名称。

install_dir
代理程序的安装路径。

b) 请参阅下列主题以响应提示：

– 第 187 页的『代理程序的配置参数』
– 第 188 页的『数据提供程序的配置参数』

c) 运行以下命令以启动代理程序：

install_dir/bin/cisco_ucs-agent.sh start *instance_name*

示例：**/opt/ibm/apm/agent/bin/cisco_ucs-agent.sh start instance_name**

下一步做什么

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

  如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

· 如果要监视大型 Cisco UCS 环境，您可能需要增加 Java™ 数据提供程序的堆大小。有关更多信息，请参阅第 189 页的『增加 Java 堆大小』。

代理程序的配置参数

在配置 Cisco UCS agent 时，可以更改参数的缺省值，例如实例名称和 SSL 验证证书。

下表包含 Cisco UCS agent 的配置参数的详细描述。

表 *20. Cisco UCS agent* 的配置参数的名称和描述		
参数名称	**描述**	**必填字段**
实例名称	实例的名称。 **限制: 实例名称**字段显示首次配置代理程序时指定的实例名称。重新配置代理程序时，无法更改代理程序的实例名称。	是
URL	Cisco UCS 管理器的 URL。 要设置 Cisco UCS 管理器的 URL，请以 http://ip_address/nuova 格式输入 URL。	是
用户名	Cisco UCS 管理器的管理员用户名。	是
密码	Cisco UCS 管理器的管理员密码。	是
确认密码	在**密码**字段中输入的同一密码。	是
SSL 信任库文件路径	安全套接字层 (SSL) 信任库文件的路径。 如果希望代理程序在使用 SSL 以在网络上通信时验证 SSL 证书，那么指定安全套接字层 (SSL) 信任库文件所在的位置。	是
验证 SSL 证书	这是一个布尔值，指示当代理程序使用 SSL 在网络上通信时，代理程序是否验证 SSL 证书。 如果希望代理程序使用 SSL 在网络上通信时，代理程序验证 SSL 证书，请将此值设置为 Yes。将此值设置为 No 可阻止代理程序验证 SSL 证书。 **提示:** 有关启用与 Cisco UCS 数据源的 SSL 通信的更多信息，请参阅第 188 页的『启用与 Cisco UCS 数据源的 SSL 通信』。	是

数据提供程序的配置参数

在配置 Cisco UCS agent 时，可以更改数据提供程序的缺省参数值，例如最大数据提供程序日志文件数量、最大日志文件大小以及日志文件中包含的详细信息级别。

下表包含数据提供程序的配置参数的详细描述。

表 *21.* 数据提供程序的配置参数的名称和描述		
参数名称	**描述**	**必填字段**
最大数据提供程序日志文件数	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。	是
每个数据提供程序日志的最大大小 (KB)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。	是
数据提供程序日志中的详细信息级别	可以包含在数据提供程序创建的日志文件中的详细信息级别。缺省值为 INFO。有效值为：OFF、SEVERE、WARNING、INFO、FINE、FINER、FINEST 和 ALL。	是

启用与 Cisco UCS 数据源的 SSL 通信

Cisco UCS agent 可配置为使用 SSL 与其 Cisco UCS 数据源进行安全通信。在此配置中，必须将数据源 SSL 证书添加到代理程序的证书信任库。

关于此任务

要点: 以下信息仅当代理程序配置为验证 SSL 证书时适用。

如果关闭 SSL 证书验证，那么即使 Cisco UCS 数据源的 SSL 证书已到期、不可信或无效，Cisco UCS agent 会连接到这些数据源。但是，关闭 SSL 证书验证可能不安全，必须谨慎实施。

如果 Cisco UCS 数据源使用常见认证中心（例如，Verisign、Entrust 或 Thawte）签署的 SSL 证书，那么无需将证书添加到 Cisco UCS agent 证书信任库。但是，如果数据源使用未由常见认证中心签署的证书（缺省情况下即为这种情况），那么必须将该证书添加到信任库，以允许代理程序成功连接并收集数据。

过程

1. 将证书文件从数据源复制到代理程序计算机。
2. 在代理程序计算机上，将证书文件放入所选的目录中。不要覆盖证书文件。为您添加的每个证书使用唯一的文件名和标签。
3. 使用 `keytool` 命令将数据源证书添加到代理程序的证书信任库：

```
keytool -import -noprompt -trustcacerts -alias CertificateAlias -file
CertificateFile -keystore Truststore -storepass TruststorePassword
```

其中

CertificateAlias

这是对添加到代理程序证书信任库的每个证书的唯一引用，例如，来自 *datasource.example.com* 的证书的相应别名为 *datasource*。

CertificateFile
要添加到信任库的 Cisco UCS 数据源证书的完整路径和文件名。

Truststore

Cisco UCS agent 证书数据库的完整路径和文件名。请使用以下路径和文件名：

- Windows （64 位）*install_dir*\tmaitm6_x64\kv6.truststore
- Linux （64 位）install_dir/lx8266/vm/etc/kv6.truststore

TruststorePassword

ITMFORVE 是 Cisco UCS agent 信任库的缺省密码。要更改此密码，请参阅 Java 运行时文档，以获取有关要使用的工具的信息。

要点: 要使用 `keytool` 命令，Java 运行时 bin 目录必须位于路径中。请使用以下命令：

- Windows （64 位）`set PATH=%PATH%;install_dir\java\java70_x64\jre\bin`
- Linux （64 位）`PATH="$PATH":/opt/ibm/apm/agent/JRE/lx8266/bin`

4. 在添加完所有数据源证书之后，启动监视代理程序。

增加 Java 堆大小

配置 Cisco UCS agent 之后，如果要监视大型 Cisco UCS 环境，您可能需要增加 Java™ 数据提供程序的堆大小。

关于此任务

Java 数据提供程序的缺省堆大小为 256 兆字节。在大型 Cisco UCS 环境中，如果发生以下问题，可能需要增加堆大小：

- Java 数据提供程序由于 `javacore` 问题而停止，并在 `CANDLEHOME\tmaitm6_x64` 目录中创建名为 `javacore.date.time.number.txt` 的文件。
- `javacore.date.time.number.txt` 文件包含字符串 `java/lang/OutOfMemoryError`。

过程

- Windows
  请完成以下步骤，以将堆大小设置为值 1 GB：

1. 打开 %CANDLE_HOME%\TMAITM6_x64\kv6_data_provider.bat 文件。
2. 在以 KV6_JVM_ARGS="$KV6_CUSTOM_JVM_ARGS... 开头的行之前添加下面这一行：

```
SET KV6_CUSTOM_JVM_ARGS=-Xmx1024m
```

3. 重新启动代理程序。

- Linux

请完成以下步骤，以将堆大小设置为值 1 GB：

1. 打开 $CANDLEHOME/lx8266/vm/bin/kv6_data_provider.sh 文件。
2. 在以 KV6_JVM_ARGS="$KV6_CUSTOM_JVM_ARGS... 开头的行之前添加下面这一行：

```
KV6_CUSTOM_JVM_ARGS=-Xmx1024m
```

3. 重新启动代理程序。

配置 Citrix Virtual Desktop Infrastructure 监视

Citrix VDI 代理程序为 Citrix XenDesktop 或 XenApp 资源（包括交付组、目录、应用程序、桌面、用户和会话）提供了中央监视点。首先必须将代理程序配置为通过交付控制器收集数据，然后才能使用代理程序。

开始之前

- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 Citrix VDI 代理程序的系统需求。有关最新系统需求信息，请参阅 Citrix VDI 代理程序的软件产品兼容性报告 (SPCR)。
- 确保您掌握下列信息：
  - 您计划连接的交付控制器的主机名。
  - OData 用户名、密码和域。
  - PowerShell 用户名、密码、域、PowerShell 端口、SSL 验证类型和认证机制（如果启用 Windows 事件日志事件和 PowerShell 度量检索）。
- 请确保代理程序操作员用户帐户至少具有 Citrix 只读管理员权限。请参阅启用 Citrix 只读管理员权限。
- 从 Citrix VDI 代理程序 V8.1.3.1 开始，已提供检索 Windows 事件日志事件的功能。为了从所有桌面交付控制器 (DDC) 和虚拟交付代理程序 (VDA) 机器检索 Windows 事件日志事件，需要对代理程序实例配置期间指定的用户帐户启用远程 PowerShell 访问权。请执行下列步骤，以确保代理程序可执行此功能：
  1. 以代理程序实例配置中指定的用户身份登录 Windows 计算机。
  2. 运行下列 PowerShell 命令，其中 *vda_system* 是已开启的 VDA 机器的名称：
     ```
     Get-WinEvent -FilterHashtable
     @{ProviderName='Citrix*';LogName='Citrix*';StartTime=((Get-
     Date).AddDays(-10))} -ComputerName vda_system
     ```
- 请确保为受监视的环境启用以下负载平衡策略：
  - CPU 使用情况
  - 磁盘使用情况
  - 内存使用情况

  可通过 Citrix Studio 应用程序配置这些策略。

关于此任务

Citrix VDI 代理程序是多实例代理程序。您必须至少创建一个实例，然后手动启动该代理程序实例。

XenApp 服务器的配置与 XenDesktop 服务器相同。当配置参数名称或描述仅提及 XenDesktop 时，它也适用于 XenApp。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   - 第 192 页的『在 Windows 系统上配置代理程序』.
   - 第 196 页的『使用静默响应文件来配置代理程序』.
2. 使用提示响应的脚本或静默响应文件在 Linux 系统上配置代理程序。
   - 第 195 页的『通过对提示做出响应来配置代理程序』.
   - 第 196 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

启用 Citrix 只读管理员权限

Citrix VDI 代理程序 要求代理程序操作员用户帐户至少具有 Citrix 只读管理员权限。

关于此任务

要从已安装 Citrix Delegated Admin PowerShell Snap-in 的计算机远程运行这些步骤，请使用 `AdminAddress` 参数。例如，步骤 2 中的命令将变为 `New-AdminAdministrator -Name “YOURDOMAIN\NewAdmin” -AdminAddress “controller1.YOURDOMAIN.com”`。其中，*YOURDOMAIN* 是网络域的名称，*NewAdmin* 是被授予 Citrix 管理权限的用户帐户，*controller1.YOURDOMAIN.com* 是 Citrix 站点服务器的标准域名。

过程

1. 使用现有 Citrix 管理员帐户启动 PowerShell 会话。
2. 装入 Delegated Admin PowerShell Snap-in 以管理 Citrix XenApp 或 XenDesktop 站点。

   `(Add-PSSnapin Citrix.DelegatedAdmin.Admin.V1)`
3. 添加代理程序操作员用户帐户作为 Citrix 站点管理员。

   `New-AdminAdministrator -Name “YOURDOMAIN\NewAdmin”`

   其中，*YOURDOMAIN* 是网络域名，*NewAdmin* 是被授予 Citrix 管理权限的用户帐户。
4. 查询可以分配给 *NewAdmin* 的角色和作用域。

   ```
   Get-AdminRole
   Get-AdminScope
   ```
5. 将角色分配给代理程序操作员用户帐户（包括只读管理员权限）。

   `Add-AdminRight -Administrator “YOURDOMAIN\NewAdmin” -Role “Read Only Administrator” -Scope “All”`

   其中
   - *YOURDOMAIN* 是网络域的名称。
   - *NewAdmin* 是被授予 Citrix 管理权限的用户帐户。
   - *Read Only Administrator* 是您正在分配的 Citrix 站点管理员角色。

· *All* 是您正在分配的 Citrix 站点管理员作用域。

6. 确认添加和更改管理员。

   `Get-AdminAdministrator -Name “YOURDOMAIN\NewAdmin”`

   其中，*YOURDOMAIN* 是网络域名，*NewAdmin* 是被授予 Citrix 管理权限的用户帐户。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Citrix VDI 代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Citrix Virtual Desktop Infrastructure** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，将禁用**配置代理程序**选项。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用字母、阿拉伯数字、下划线字符和减号字符。例如，`vdi_inst2`。

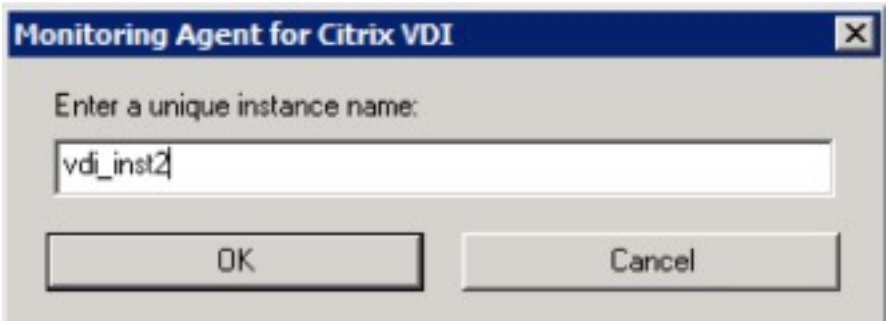


图 *13.* 用于输入唯一实例名称的窗口。

4. 在代理程序实例名称窗口上单击**下一步**。

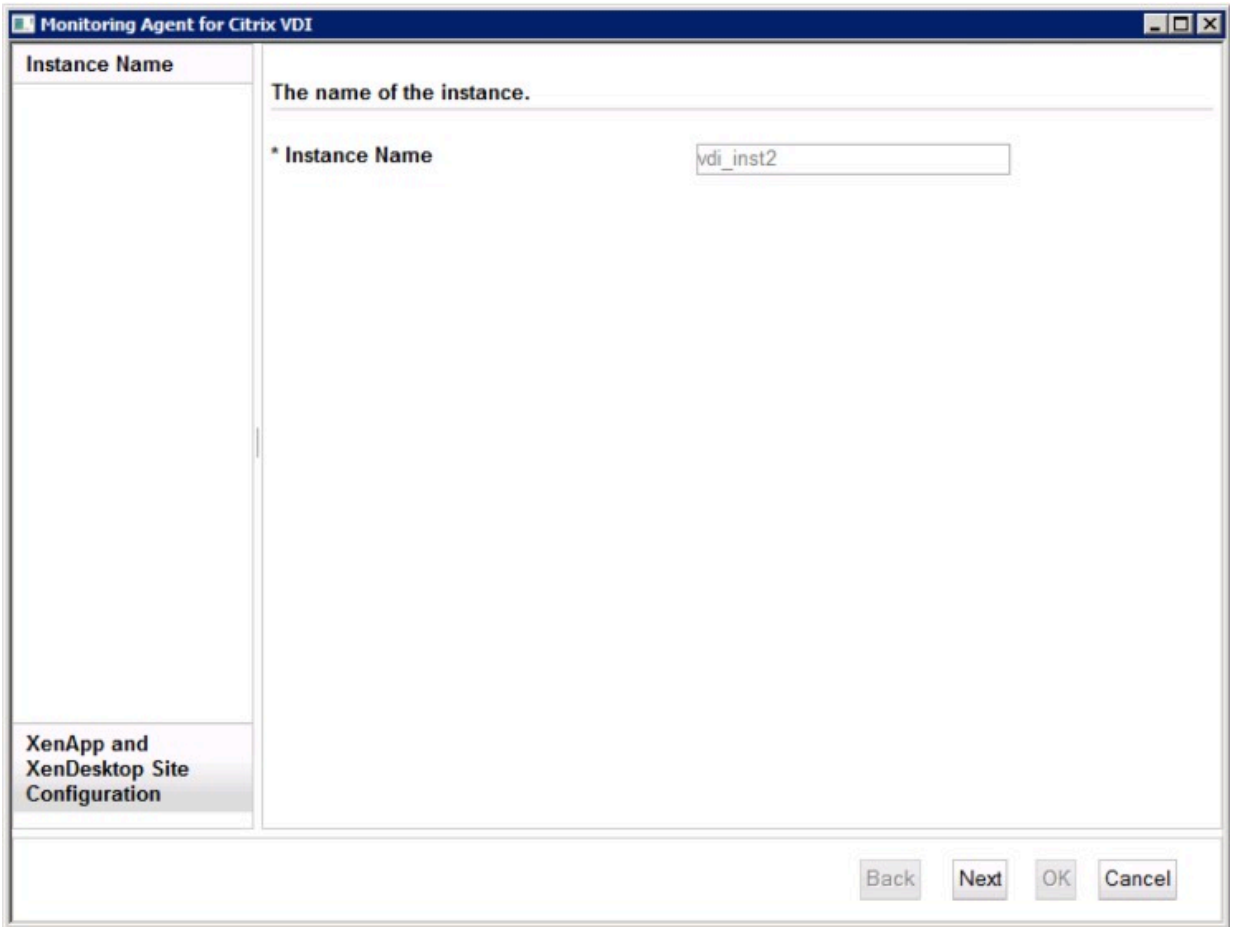


图 *14.* 代理程序实例名称窗口。

5. 输入 **XenApp 和 XenDesktop 站点配置**实例模板设置。

   注: 本节并非 XenApp 或 XenDesktop 站点实例配置。这是一个模板节，用于设置当您在步骤 6 中添加实际的 XenApp 或 XenDesktop 站点实例配置时，要用作缺省值的内容。

   请参阅第 197 页的表 22，以获取每个配置参数的说明。

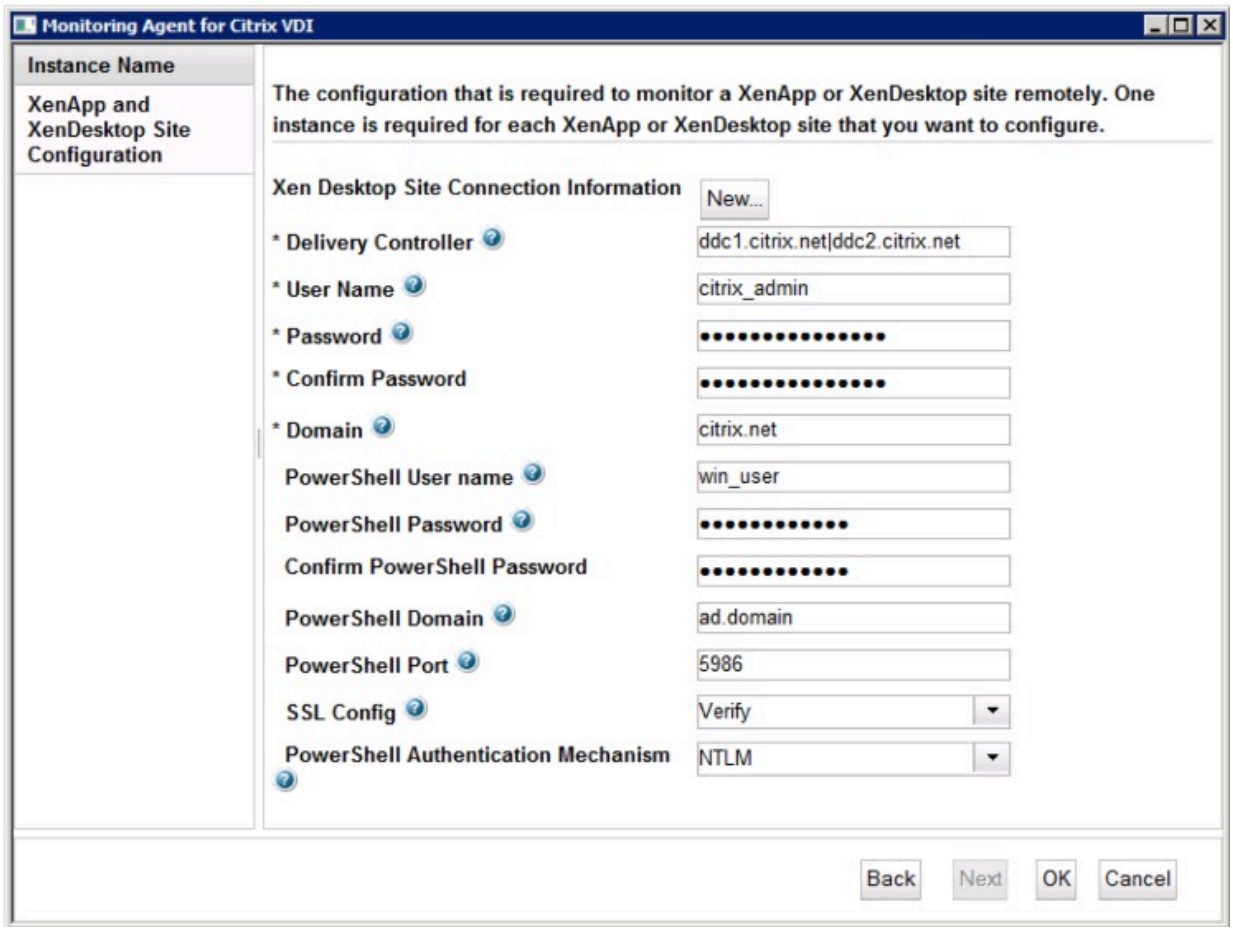


图 *15.* 用于指定 *XenApp* 或 *XenDesktop* 站点实例模板设置的窗口。

6. 按**新建**并输入 XenApp 或 XenDesktop 站点实例设置，然后单击**下一步**。
   请参阅第 197 页的表 22，以获取每个配置参数的说明。

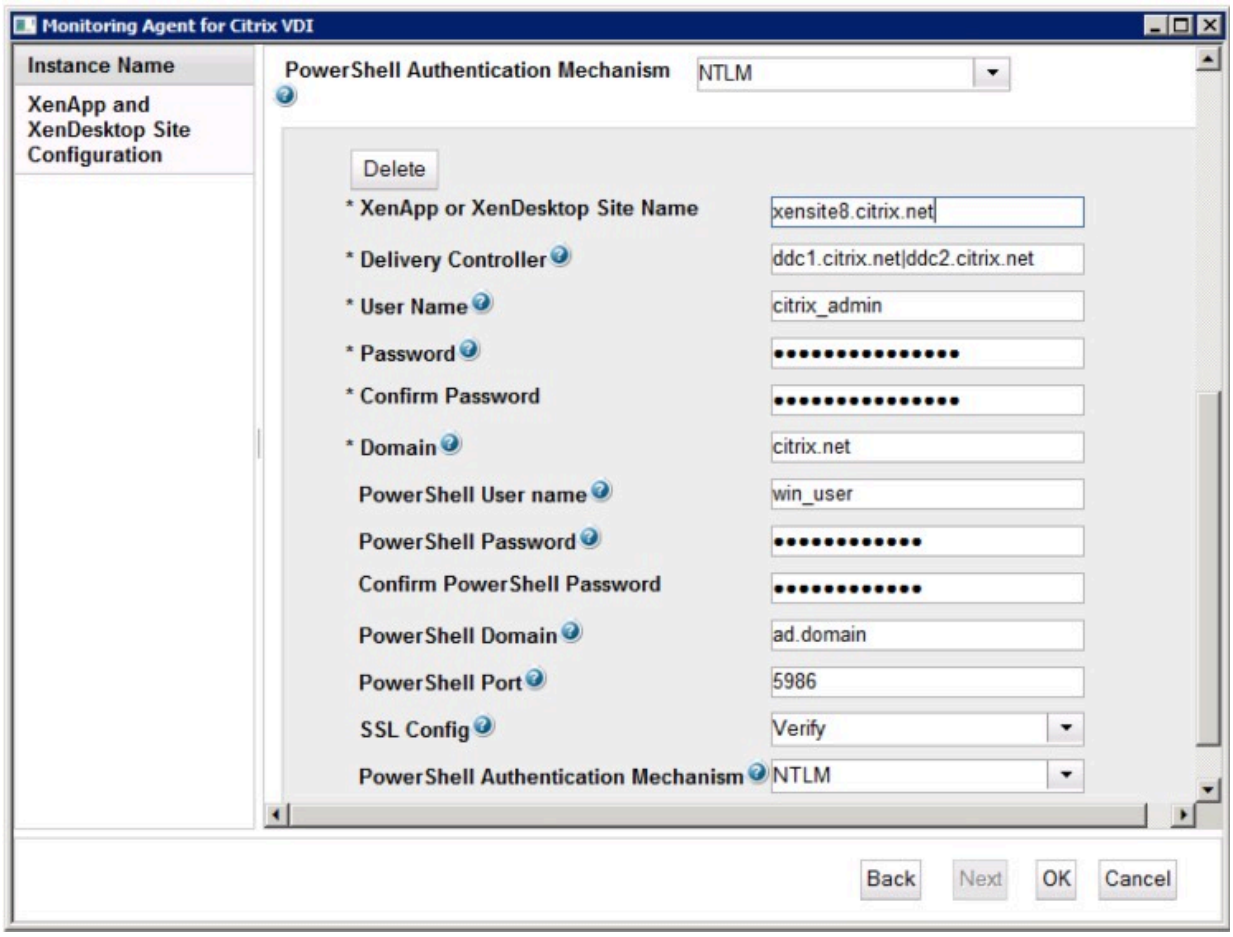


图 *16.* 用于指定 *XenApp* 或 *XenDesktop* 站点实例设置的窗口。

注: 仅当第 198 页的『启用 Windows 事件和 PowerShell 度量的监视』时，才需要 **PowerShell User name** 参数和所有以下 PowerShell 参数。缺省情况下，这些高级环境变量是关闭的，因为它们对受监视系统的负载非常大。

注: 确保为每个新的 XenApp 或 XenDesktop 站点实例正确设置了 **SSL Config** 和 **PowerShell Authentication Mechanism** 参数。缺陷导致设置缺省值而不是模板值。

7. 单击**确定**以完成配置。
8. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过对提示做出响应来配置代理程序

安装 Citrix VDI 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 Citrix VDI 代理程序安装在本地 Linux 机器上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 Citrix VDI 代理程序。

1. 运行以下命令:

```
install_dir/bin/citrixvdi-agent.sh
config instance_name
```

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/citrixvdi-agent.sh config vdi_inst01
```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 197 页的『Citrix VDI 代理程序的配置参数』，以获取每个配置参数的说明。

   注: 仅当第 198 页的『启用 Windows 事件和 PowerShell 度量的监视』时，才需要 **PowerShell User name** 参数和所有以下 PowerShell 参数。缺省情况下，关闭这些高级环境变量，因为它们放置在受监视系统上的负载非常大。
3. 运行以下命令以启动代理程序：

   ```
   install_dir/bin/citrixvdi-agent.sh
   start instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/citrixvdi-agent.sh start vdi_inst01
   ```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件来为配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

- 以静默方式配置 Citrix VDI 代理程序：

  a) 在文本编辑器中打开位于以下某个路径中的 `citrixvdi_silent_config.txt` 文件。

  – Linux *install_dir*/samples/citrixvdi_silent_config.txt

    例如， /opt/ibm/apm/agent/samples/citrixvdi_silent_config.txt

  – Windows *install_dir*\samples\citrixvdi_silent_config.txt

    例如，C:\IBM\APM\samples\citrixvdi_silent_config.txt

  其中，*install_dir* 是安装代理程序的路径。

  b) 在 `citrixvdi_silent_config.txt` 文件中，指定所有必需参数的值并根据需要修改其他参数的缺省值。

  请参阅第 197 页的『Citrix VDI 代理程序的配置参数』，以获取每个配置参数的说明。

  注: 仅当第 198 页的『启用 Windows 事件和 PowerShell 度量的监视』时，才需要 **PowerShell User name** 参数和所有以下 PowerShell 参数。缺省情况下，这些高级环境变量是关闭的，因为它们对受监视系统的负载非常大。

  c) 保存并关闭 `citrixvdi_silent_config.txt` 文件，然后运行以下命令：

  – Linux *install_dir*/bin/citrixvdi-agent.sh config **instance_name** *install_dir*/samples/citrixvdi_silent_config.txt

例如，**/opt/ibm/apm/agent/bin/citrixvdi-agent.sh config vdi_inst01 /opt/ibm/apm/agent/samples/citrixvdi_silent_config.txt**

– Windows *install_dir*\bin\citrixvdi-agent.bat config **instance_name** *install_dir*\samples\citrixvdi_silent_config.txt

例如，**C:\IBM\APM\bin\citrixvdi-agent.bat config vdi_inst01 C:\IBM\APM\samples\citrixvdi_silent_config.txt**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

d) 运行以下命令以启动代理程序：

– Linux *install_dir*/bin/citrixvdi-agent.sh start **instance_name**

例如，**/opt/ibm/apm/agent/bin/citrixvdi-agent.sh start vdi_inst01**

– Windows *install_dir*\bin\citrixvdi-agent.bat start **instance_name**

例如，**C:\IBM\APM\bin\citrixvdi-agent.bat start vdi_inst01**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

Citrix VDI 代理程序的配置参数

下表显示 Citrix VDI 代理程序的配置参数。

1. Citrix VDI 代理程序设置 - Citrix VDI 代理程序环境设置。

表 *22. Citrix VDI* 代理程序设置

参数名称	描述	静默配置文件参数名称
XenApp 或 XenDesktop 站点名称	提供一个名称，用于标识 XenApp 或 XenDesktop 站点代理程序实例。例如，*vdi_inst2* **注:** 此别名用来代表 WebLogic Server 代理程序实例，可以随您选择，但有下列限制。只有字母、阿拉伯数字、下划线字符和减号可用于连接名称中。连接名称的最大长度为 25 个字符。	下列每个参数都必须具有与代理程序实例的每个参数相同的实例名称后缀。新代理程序实例必须对其参数集使用唯一的实例名称。例如，一个代理程序实例可使用 *.vdi1* 来替代以下参数名称中的 *.instance_name*，另一个代理程序实例可使用 *.vdi2* 来替代。
交付控制器	交付控制器的主机名或 IP 地址。如果在集群中设置多个 DDC，那么可提供用“|”分隔的交付控制器列表。	**KVD_XDS_DELIVERY_CONTROLLER.instance_name**
用户名	在指定的 XenApp 或 XenDesktop 交付控制器上用于向 OData API 认证的用户名。	**KVD_XDS_ODATA_USERNAME.instance_name**
密码	在指定的 XenApp 或 XenDesktop 交付控制器上用于向 OData API 认证的密码。	**KVD_XDS_ODATA_PASSWORD.instance_name**
域	在指定的 XenApp 或 XenDesktop 交付控制器上用于向 OData API 认证的域。	**KVD_XDS_ODATA_DOMAIN.instance_name**
PowerShell 用户名	针对远程 VDA 和 DDC 机器的 PowerShell 调用，用于认证的用户名。 **注:** 仅当第 198 页的『启用 Windows 事件和 PowerShell 度量的监视』时，才需要这个及所有以下 PowerShell 参数。缺省情况下，这些高级环境变量是关闭的，因为它们对受监视系统的负载非常大。	**KVD_XDS_POWERSHELL_USERNAME.instance_name**

表 22. Citrix VDI 代理程序设置 (续)		
参数名称	**描述**	**静默配置文件参数名称**
PowerShell 密码	与所提供的 PowerShell 用户名关联的密码。	**KVD_XDS_POWERSHELL_PASSWORD.instance_name**
PowerShell 域	与所提供的 PowerShell 用户名关联的域。	**KVD_XDS_POWERSHELL_DOMAIN.instance_name**
PowerShell 端口	打开以供 WinRm 使用的 SSL 端口。 PowerShell 缺省远程连接端口为 5985（对于 HTTP）和 5986（对于 HTTPS）。	**KVD_XDS_POWERSHELL_PORT.instance_name**
SSL 需求	选择您的环境需要的 SSL 选项。	**KVD_XDS_SSL_CONFIG.instance_name** 有效值 **KVD_XDS_SSL_CONFIG_VERIFY** 验证 **KVD_XDS_SSL_CONFIG_NOVERIFY** 未验证 **KVD_XDS_SSL_CONFIG_NOSSL** 无 SSL
PowerShell 认证机制	定义认证类型，它用于创建凭证以使用 PowerShell 从远程系统检索信息。	**KVD_XDS_POWERSHELL_AUTH_MECH.instance_name** 有效值 **KVD_XDS_POWERSHELL_BASIC** 基本 **KVD_XDS_POWERSHELL_CREDSSP** CredSSP **KVD_XDS_POWERSHELL_NTLM** NTLM **KVD_XDS_POWERSHELL_DEFAULT** 缺省值 **KVD_XDS_POWERSHELL_DIGEST** 摘要 **KVD_XDS_POWERSHELL_KERBEROS** Kerberos **KVD_XDS_POWERSHELL_NEGOTIATE** 协商

启用 Windows 事件和 PowerShell 度量的监视

使用此过程启用 Windows 事件和 PowerShell 度量的监视。监视此数据可以对受监视的系统产生显著性能影响。

开始之前

确保设置了代理程序的 PowerShell 配置参数。

关于此任务

必须为代理程序启用一个或多个以下高级环境变量以监视 Windows 事件和 PowerShell 度量。

GET_SESSION_LATENCY
是否从来自 PowerShell 的已连接的 VDA 远程检索会话等待时间和往返时间。

GET_VDA_MACHINE_METRICS_REMOTELY
是否从 PowerShell 远程检索 VDA 机器。

RETRIEVE_WINDOWS_EVENTS
是否从来自 Windows VDA 和 DDC 的 PowerShell 检索 Windows 事件日志事件。

过程

1. 转至 Citrix VDI 代理程序的代理程序安装目录：
   - Linux *install_dir*/config
   - Windows *install_dir*\TMAITM6_x64

   其中，*install_dir* 是安装代理程序的路径。
2. 编辑 Citrix VDI 代理程序配置文件以将 *GET_SESSION_LATENCY*、*GET_VDA_MACHINE_METRICS_REMOTELY* 和 *RETRIEVE_WINDOWS_EVENTS* 变量中的一个或多个变量设置为 `true`。
   - Linux `vd.environment`
   - Windows KVDENV_*instance_name*

   其中，*instance_name* 是代理程序实例的名称。
3. 重新启动代理程序。

   要点： 要使这些设置成为所有新代理程序实例的缺省设置，请在配置模板文件中将其设置为 `true`：
   - Linux 通过在步骤 2 中编辑 `vd.environment`，此设置已成为新代理程序实例的缺省设置。
   - Windows KVDENV

示例

```
GET_SESSION_LATENCY=true
GET_VDA_MACHINE_METRICS_REMOTELY=true
RETRIEVE_WINDOWS_EVENTS=true
```

配置 DataPower 监视

要监视 DataPower 应用程序，您首先需要在您的设备上完成一些配置任务，然后再配置 Monitoring Agent for DataPower。

提示： 单击 APM v8：在 IBM APM 中配置 DataPower 监视以观看包含 DataPower 监视的基本配置过程的视频。

配置 DataPower 设备

在配置 Monitoring Agent for DataPower 之前，必须先在设备上完成一些配置任务。

提示： 有关受支持的 DataPower 设备的信息，请参阅 Software Product Compatibility Reports 中的 Prerequisites 选项卡。

您可以在三个不同级别监视 DataPower 设备。根据需要，在要监视的每个 DataPower 设备上配置三个级别，以在 Cloud APM 控制台中显示 DataPower 设备数据。

1. 资源监视

   要查看监视数据，例如资源利用率、吞吐量和连接统计信息，请启用资源监视。有关指示信息，请参阅第 200 页的『资源监视』。
2. 中间件事务跟踪

要查看事务的监视数据，例如事务详细信息、事务量和依赖关系，请启用中间件事务跟踪。有关指示信息，请参阅第 201 页的『中间件事务跟踪』。

3. DataPower 设备的实例级别事务跟踪

   要在实例拓扑中显示事务的监视数据，请配置 DataPower 设备的实例级别事务跟踪。有关指示信息，请参阅第 202 页的『DataPower 设备的实例级别事务跟踪』。

要点: 请确保用户标识具有正确的许可权来配置 DataPower 设备。您可以针对用来配置 DataPower 设备的用户标识，在**访问概要文件**字段中输入 `*/*/*?Access=r`。然后，使用此用户标识来配置 DataPower 设备。

导出公用证书
如果 DataPower 设备的“XML 管理接口”启用了“SSL 代理概要文件”，那么必须将 DataPower 设备的“XML 管理接口”使用的公用证书导出到运行 DataPower 代理程序的机器上。

过程

1. 要下载 DataPower 设备的“XML 管理界面”所使用的加密证书（例如，`pubcert:///mycert.pem`），请单击**管理** > **主要** > **文件管理**，然后将该证书保存到运行 DataPower 代理程序的机器。
2. 在配置 DataPower 代理程序时，有一个选项可用于指定 **SSL 代理概要文件**字段。输入公用证书的绝对路径。

   注: 添加其他多协议网关时，需要重复这些步骤。

资源监视
可用于 DataPower 设备的第一级监视是启用资源监视，例如 SOAP 管理、统计信息和事务速率。

在下列配置任务中，DataPower Gateway 用户界面 (UI) 上的操作适用于 DataPower Gateway V7.5.1 及先前版本。如果您使用的是比 V7.5.1 更新的 DataPower Gateway 版本，您可单击 UI 右上角的问号，并选择 **WebGUI** 以回到先前版本的 UI。然后，请按指示信息进行操作，以完成 DataPower 设备配置任务。

启用 *SOAP* 管理
如果希望 DataPower 代理程序从 DataPower 设备收集数据，必须配置 XML 管理接口并启用 SOAP 管理。

过程

要启用 SOAP:

1. 登录到想要监视的 DataPower 设备的 WebGUI。
2. 单击**对象** > **设备管理** > **XML 管理接口**。

   注: 确保启用“管理”状态。
3. 对于**端口号**，输入 DataPower 代理程序侦听通知报告的端口号。缺省情况下，端口号为 5550。
4. 对于**已启用的服务**，确保选中 **SOAP 管理**。

启用统计信息
如果希望 DataPower 代理程序从 DataPower 设备收集数据，必须启用统计信息。

过程

要启用统计信息，请完成以下步骤:

1. 登录到想要监视的 DataPower 设备的 WebGUI。
2. 单击**管理** > **设备** > **统计信息设置**。
3. 启用**统计信息设置**，然后单击**应用**。

启用事务速率
如果希望 DataPower 代理程序从 DataPower 设备收集数据，必须启用事务速率。

过程

要启用事务速率，请完成以下步骤：

1. 登录到想要监视的 DataPower 设备的 WebGUI。
2. 选择缺省域。
3. 单击**状态** > **连接** > **事务速率**。
4. 如果显示**统计信息当前处于禁用状态**，单击**已禁用**，然后在“统计信息设置”中，将**管理状态**设置为**已启用**。
5. 如果您有多个域，单击**显示所有域**，然后重复步骤 3-4 来为所有适用域启用事务速率。
6. 单击**应用**。

中间件事务跟踪
可用于 DataPower 设备的第二级监视是在工作空间中显示中间件事务跟踪。

在下列配置任务中，DataPower Gateway 用户界面 (UI) 上的操作适用于 DataPower Gateway V7.5.1 及先前版本。如果您使用的是比 V7.5.1 更新的 DataPower Gateway 版本，您可单击 UI 右上角的问号，并选择 **WebGUI** 以回到先前版本的 UI。然后，请按指示信息进行操作，以完成 DataPower 设备配置任务。

支持对通过 DataPower 设备传送的 SOAP 流量和 REST 流量进行事务跟踪。DataPower 事务跟踪支持使用文件 `store:///soapreq.xsl`、`store:///soaprsp.xsl` 和 `store:///soaperror.xsl` 的 SOAP。这些 XSL 文件检测 Web Service 代理以在 SOAP 包络中添加和报告 `kd4:KD4SoapHeaderV2`。

除了 `soap*.xsl` 文件，DataPower 事务跟踪还包含 `apm_req.xsl`、`apm_rsp.xsl` 和 `apm_error.xsl`，这三个文件支持包含 `ARM_CORRELATOR: HTTP Header` 的入局 HTTP 请求，或包含 `ITCAMCorrelator` 或 `kd4:KD4SoapHeaderV2` 的 SOAP 包络。Web Service 代理更新或设置出局请求以使其包含 `ARM_CORRELATOR: HTTP Header`，并移除 SOAP 相关因子。

注: 如果将 DataPower 设备添加到业务应用程序，且设备携带了多个应用程序的流量，那么在启用事务跟踪之后，针对这些业务应用程序显示的应用程序拓扑包含所有应用程序的节点路径。

配置 *Web Service* 管理
针对要显示其跟踪数据的每个 DataPower 设备完成以下步骤。

1. 登录到想要监视的 DataPower 设备的 WebGUI。
2. 选择缺省域。
3. 搜索 `XML 管理接口`，设置以下值并单击**应用**。
   - 在**主要**选项卡上的**已启用服务**部分中，启用 **WS 管理端点**
4. 搜索 `Web Service 管理代理程序`。设置以下值并单击**应用**。
   - 将**管理状态**设置为**已启用**
   - 将**捕获方式**设置为**无**
   - 将**缓存方式（不推荐）**设置为**废弃**
5. 按以下主题所述配置 Web Service 代理或多协议网关。

配置 *Web Service* 代理
针对要显示其跟踪数据的每个 Web Service 代理完成以下步骤。

过程

1. 选择 Web Service 代理所属的域。
2. 在**代理设置**选项卡上，设置以下值并单击**应用**：
   - 将**通过 Web Service 管理代理程序进行监视**设置为**打开**

3. 要报告 SOAP 故障，请在 Cloud APM 控制台中禁用错误处理并启用错误报告：在**高级代理设置**选项卡上，将**处理 HTTP 错误**设置为**关闭**，然后单击**应用**。

配置多协议网关
针对要显示其事务跟踪数据的每个多协议网关完成以下步骤。

过程

1. 选择多协议网关所属的域。
2. 在多协议网关的**高级**选项卡上，设置以下值并单击**应用**：
   - 将**通过 Web Service 管理代理程序进行监视**设置为**打开**
   - 如果 Web 服务器使用重定向，请将**遵循重定向**设置为**关闭**。然后将**重写位置 URL** 设置为**打开**。
3. 如果您要监视响应类型或请求类型为非 XML 的多协议网关，必须使用同时涵盖“客户机到服务器”和“服务器到客户机”方向的规则来定义多协议网关策略。如果非 XML 多协议网关在其策略中不包含规则，那么 Web Service 管理代理程序或 DataPower 调试探测器（如果启用）不会捕获到任何流量。
4. 要从后端服务器传播 HTTP 响应代码并报告 SOAP 故障，请在**高级设置**选项卡上，将**处理后端错误**设置为**关闭**，然后单击**应用**。

DataPower 设备的实例级别事务跟踪
可用于 DataPower 设备的第三级监视是在实例拓扑中显示其数据。

在下列配置任务中，DataPower Gateway 用户界面 (UI) 上的操作适用于 DataPower Gateway V7.5.1 及先前版本。如果您使用的是比 V7.5.1 更新的 DataPower Gateway 版本，您可单击 UI 右上角的问号，并选择 **WebGUI** 以回到先前版本的 UI。然后，请按指示信息进行操作，以完成 DataPower 设备配置任务。

配置变换
在希望在实例拓扑中显示的每个 DataPower 设备上完成以下步骤。

关于此任务

对于 IBM Performance Management V8.1.2 FP1，支持通过 DataPower 设备对 SOAP 流量进行事务跟踪。DataPower 事务跟踪支持使用文件 `store:///soapreq.xsl`、`store:///soaprsp.xsl` 和 `store:///soaperror.xsl` 的 SOAP。这些 XSL 文件检测 Web Service 代理以在 SOAP 包络中添加和报告 `kd4:KD4SoapHeaderV2`。

对于 IBM Performance Management V8.1.3 及更高版本，还支持通过 DataPower 设备对 REST 流量进行事务跟踪。除了 `soap*.xsl` 文件，DataPower 事务跟踪还包含 `apm_req.xsl`、`apm_rsp.xsl` 和 `apm_error.xsl`，这三个文件支持包含 `ARM_CORRELATOR: HTTP Header` 的入局 HTTP 请求，或包含 `ITCAMCorrelator` 或 `kd4:KD4SoapHeaderV2` 的 SOAP 包络。Web Service 代理更新或设置出局请求以使其包含 `ARM_CORRELATOR: HTTP Header`，并移除 SOAP 相关因子。

DataPower 代理程序支持对通过 DataPower 设备的 SOAP 流量、通过 DataPower 设备的 REST 流量以及 DataPower 与 WebSphere MQ 之间的流量执行事务跟踪。

- 如果要对通过 DataPower 设备的 SOAP 和 REST 流量启用事务跟踪，请应用 `apm_req.xsl`、`apm_rsp.xsl` 和 `apm_error.xsl`，它们支持包含 `ARM_CORRELATOR: HTTP Header` 的传入 HTTP 请求，或者包含 `ITCAMCorrelator` 或 `kd4:KD4SoapHeaderV2` 的 SOAP 包络。Web Service 代理更新或设置出局请求以使其包含 `ARM_CORRELATOR: HTTP Header`，并移除 SOAP 相关因子。
- 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想在 DataPower 与 WebSphere MQ 之间启用事务跟踪，请应用 `apm_req_MQ.xsl`、`apm_rsp_MQ.xsl` 和 `apm_error_MQ.xsl` 文件。应用这些文件后，针对 SOAP 和 REST 流量的事务跟踪也会自动启用。

过程

要在 DataPower 与 WebSphere MQ 之间跟踪 REST 流量并启用事务跟踪，请完成下列步骤：

1. 从以下位置下载文件：
   - 对于 Linux 系统：`/opt/ibm/apm/agent/lx8266/bn/bin`

· 对于 AIX 系统：`/opt/ibm/apm/agent/aix536/bn/bin`

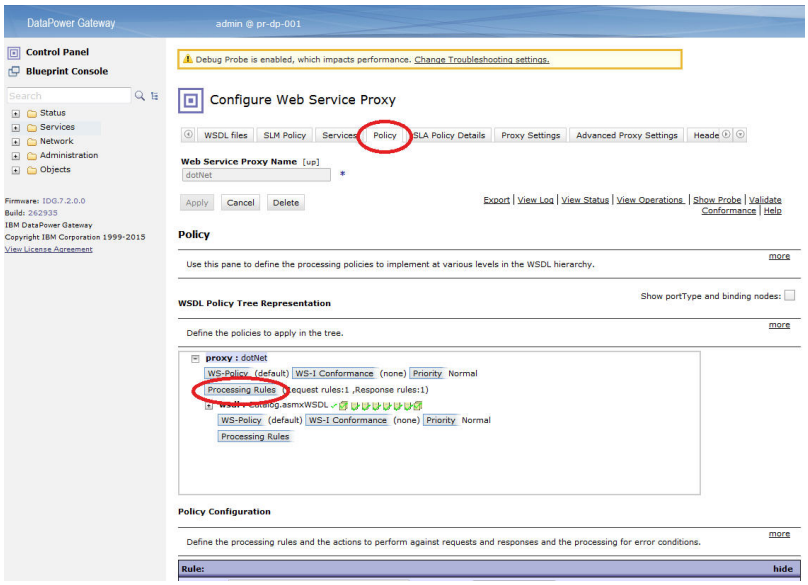
2. 将 XSL 文件上载到要作为 IBM 集成堆栈的一部分监视的每个 DataPower 设备。
3. 按以下主题所述配置 Web Service 代理或多协议网关。
4. 对于要监视的每个域，使用以下步骤进行配置：
   a) 从 DataPower Gateway 标题的下拉列表中选择域。
   b) 在“控制面板”导航器中，选择**对象 > 设备管理 > Web Service 管理代理程序**。
   c) 将**缓存方式（不推荐）**设置为废弃。
   d) 单击**应用**。

配置 *Web Service* 代理
在希望在实例拓扑中显示的每个 Web Service 代理上完成以下步骤。

过程

在 WebGUI 中，针对要监视的每个 Web Service 代理完成以下步骤：

1. 在**配置 Web Service 代理**页面中，选择要配置的 Web Service 代理的名称。
2. 在**策略**选项卡上，展开 **proxy : *domain***，然后单击**处理规则**。

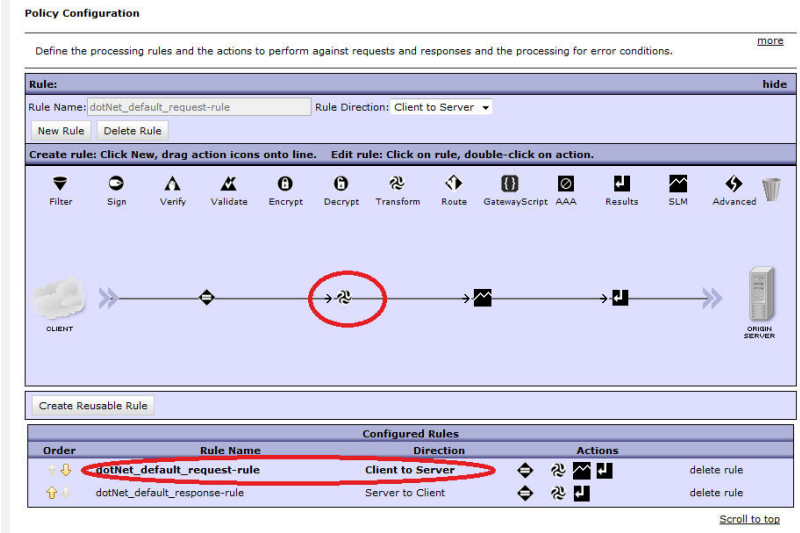


3. 在**策略配置**部分中，选择现有的客户机到服务器规则，或者单击**新建规则**以创建规则。
   a. 将“变换”拖到时间线上。

      注：

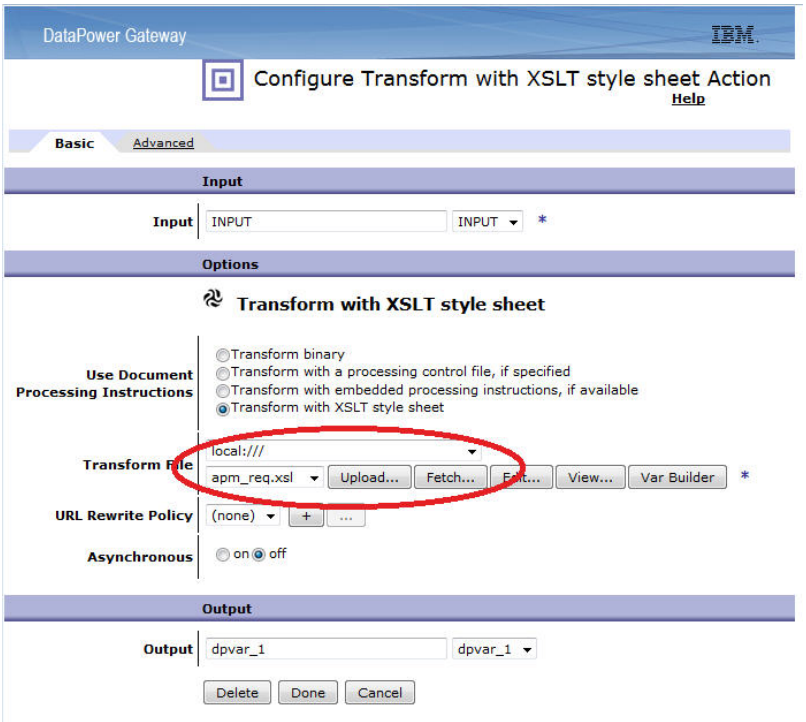
      1) 如果客户机到服务器规则已存在，请向其中添加变换节点。
      2) 如果客户机到服务器规则包含“认证、授权和审计”(AAA) 节点，请确保包含 DataPower 代理程序 xslt 文件的变换节点位于该 AAA 节点之前。

   b. 双击“变换”以进行编辑。



c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_req.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。



提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置客户机到服务器规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_req_MQ.xsl` 文件，而不要应用 `apm_req.xsl` 文件。

d. 单击**完成**。

4. 返回到**策略配置**部分中，重复步骤 3 以配置服务器到客户机规则，或者单击**新建规则**以创建规则。

a. 将“变换”拖到时间线上。

b. 双击该“变换”以进行编辑。

c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_rsp.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。

提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置服务器到客户机规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_rsp_MQ.xsl` 文件，而不要应用 `apm_rsp.xsl` 文件。

d. 单击**完成**。

5. 返回到**策略配置**部分中，重复步骤 3 以配置错误规则，或者单击**新建规则**以创建规则。

a. 将“变换”拖到时间线上。

b. 双击该“变换”规则以进行编辑。

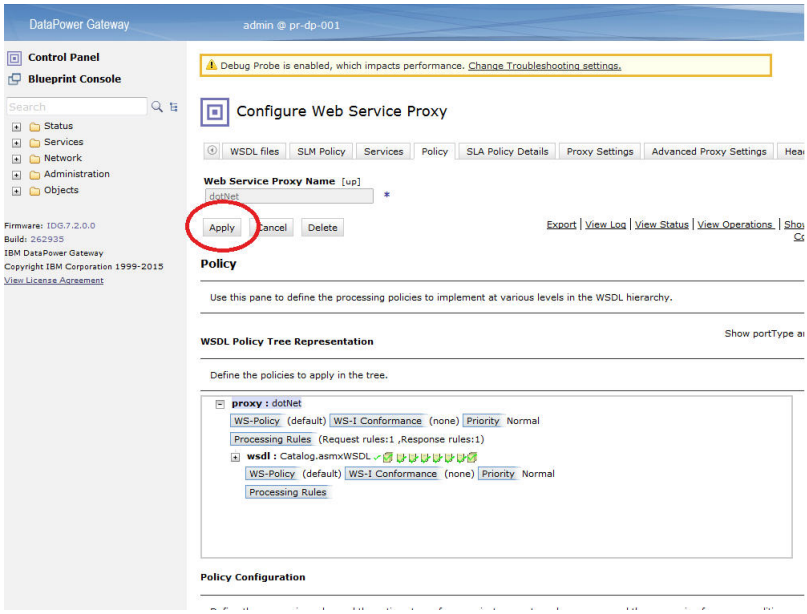
c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_error.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。

提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置错误规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_error_mq.xsl` 文件，而不要应用 `apm_error.xsl` 文件。

d. 单击**完成**。

6. 返回到**配置 Web Service 代理**页面，单击**应用**。



配置多协议网关
在希望在实例拓扑中显示的每个多协议网关上完成以下步骤。

过程

在 WebGUI 中，针对要监视的每个多协议网关完成以下步骤。

1. 在**配置多协议网关**页面中，单击要配置的多协议网关的名称。
2. 在**多协议网关策略**页面上，配置该策略。单击**...**。
3. 在**配置多协议网关样式策略**页面上，选择现有客户机到服务器规则，或者单击**新建规则**以创建规则。

a. 将“变换”拖到时间线上。

注:

1) 如果客户机到服务器规则已存在，请向其中添加变换节点。

2) 如果客户机到服务器规则包含“认证、授权和审计”(AAA) 节点，请确保包含 DataPower 代理程序 xslt 文件的变换节点位于该 AAA 节点之前。

b. 双击该“变换”规则以进行编辑。

c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_req.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。

提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置客户机到服务器规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_req_MQ.xsl` 文件，而不要应用 `apm_req.xsl` 文件。

d. 单击**完成**。

4. 返回到**配置多协议网关样式策略**页面上，选择现有服务器到客户机规则，或者单击**新建规则**以创建规则。

a. 将“变换”拖到时间线上。

b. 双击该“变换”规则以进行编辑。

c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_rsp.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。

提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置服务器到客户机规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_rsp_MQ.xsl` 文件，而不要应用 `apm_rsp.xsl` 文件。

d. 单击**完成**。

5. 返回到**配置多协议网关样式策略**页面上，选择现有错误规则，或者单击**新建规则**以创建规则。

a. 将“变换”拖到时间线上。

b. 双击该“变换”规则以进行编辑。

c. 在**使用 XSLT 样式表操作配置变换**窗口的“变换文件”旁边，从之前将 `apm_error.xsl` 上载到的数据存储器中选择该文件。例如，`local:///`

如果该文件不存在，单击**上载**以从安装位置获取该文件。

提示: 除了监视通过 DataPower 设备的 SOAP 和 REST 流量之外，如果您还想配置错误规则以监视 DataPower 与 WebSphere MQ 之间的流量，请在此步骤中应用 `apm_error_mq.xsl` 文件，而不要应用 `apm_error.xsl` 文件。

d. 单击**完成**。

6. 返回到**配置多协议网关样式策略**页面，在**高级**选项卡上，将**通过 Web Service 管理代理程序进行监视**设置为**打开**，然后单击**应用**。
7. 单击**应用**。

下一步做什么

在某些情况下，添加“事务跟踪”变换可能导致 DataPower 更改 HTTP Content-Type 头的值。您可能会看到包含未装入的图像的 Web 页面，或者呈现为乱码 HTML 文本的二进制文件。

将不包含 XSL 变换的规则与包含一个或多个 XSL 变换的规则进行比较，DataPower 的行为发生更改。如果服务处理 MIME、MTOM、XOP 或其他已编码的消息，那么可期望此行为，否则修改 DataPower 配置以阻止此行为。

要阻止 DataPower 修改 HTTP Content-Type 头，请在每个受影响的规则 (var://service/mpgw/proxy-content-type) 中设置变量：

1. 将“高级”对象拖动到规则。
2. 双击“高级”对象以进行编辑。
3. 选择**设置变量**，然后单击**下一步**。
4. 输入“变量名称” `service/mpgw/proxy-content-type` 和“变量值” `1`，然后单击**完成**。
5. 应用策略和服务配置更改。

6. 针对每个受影响的规则重复步骤 1-5。

配置 DataPower 代理程序

Monitoring Agent for DataPower 提供了针对企业环境中的 DataPower 设备的集中监视点。您可以识别设备常见问题并接收相应的通知。代理程序还提供有关设备的性能、资源和工作负载的信息。

关于此任务

DataPower 代理程序是多实例代理程序；必须创建第一个实例并手动启动该代理程序。“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。“受管系统名称”限制为 32 个字符。

指定的实例名称限制为 28 个字符（减去主机名长度后）。例如，如果指定 DataPower 作为实例名称，那么受管系统名称为 DataPower:hostname:BN。

要点: 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法正确显示。

注: DataPower 代理程序的 XSLT 不解析用于大型机应用程序的 BLOB 字符。

对于每个生产 DataPower 设备，请配置一个实例。如果 DataPower 设备是非生产或较小的设备，那么只能配置一个代理程序实例以监视所有设备。在同一台机器上，可以运行多个实例。您可以运行配置脚本以创建实例和更改任何配置设置。您可以先编辑代理程序静默响应文件，然后再运行脚本以绕过必需的提示和响应。

过程

- 要配置 DataPower 代理程序，请完成以下某个过程：
  - Linux AIX 要通过响应提示来配置代理程序，请完成下列步骤：
    1. 转至 *install_dir*/bin 目录，其中 *install_dir* 是 DataPower 代理程序的安装目录。
    2. 运行 ./datapower-agent.sh config *instance_name* 命令。

       选择对于服务器唯一的 *instance_name*。
    3. 当提示编辑 DataPower 代理程序设置时，请输入 1 以继续。
    4. 当提示编辑 **Managed System Details** 时，请输入下列其中一个选项：

       - 1=Add
       - 2=Edit
       - 3=Del
       - 4=Next
       - 5=Exit

       如果这是您第一次在系统上配置 DataPower 代理程序实例，那么将显示 No 'DataPower Appliances' settings available 消息。请输入 1 以添加 DataPower 设备设置。缺省值为选项 5=Exit。
    5. 输入 DataPower 设备属性：

       受管系统名称
       对于 **Managed System Name**，输入代理程序的受管系统名称。

       选择在代理程序的所有实例之间唯一且可用于轻松标识设备的 **Managed System Name**。名称应仅包含字母数字字符，例如，DataPower 设备的主机名。

       设备主机
       对于 **Device Host**，输入受监视的 DataPower 设备的 IP 地址。缺省 IP 地址为 *9.123.109.139*。

XML 管理接口端口
对于 **XML Management Interface Port**，输入 XML 管理接口的端口号。缺省端口号为 5550。

用户标识
对于 **User ID**，输入用于登录到受监视的 DataPower 设备的用户标识。缺省值为 admin。

密码
对于 **Password**，输入用于登录到受监视的 DataPower 设备的密码，然后确认密码。

SSL 代理概要文件
对于 **SSL Proxy Profile**，如果设备的 XML 管理接口配置为使用 SSL 代理概要文件，请输入此概要文件的公用证书的绝对路径。例如，

```
the location of the .pem file exported from datapower appliances/mycert.pem
```

其中，*the location of the .pem file exported from datapower appliances* 是公用证书的绝对路径。要导出公用证书，请参阅导出公用证书。

SSL 代理选项
对于 **SSL 代理选项**，如果将受监视设备的 XML 管理接口配置为使用定制 SSL 代理概要文件，那么设置为 Yes。否则，请将它设置为 No。

6. 要监视多个 DataPower 设备，请重复 第 207 页的『4』 和 第 207 页的『5』，以便为每个 DataPower 设备配置一个代理程序实例。否则，请输入 5 并按 **Enter** 键以完成配置。
7. 运行以下命令以启动代理程序：

```
./datapower-agent.sh start instance_name
```

- 静默配置

1. 要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请完成以下步骤：
   - Linux AIX 在文本编辑器中打开 *install_dir*/samples/datapower_silent_config.txt。
   - Windows 在文本编辑器中打开 *install_dir*/samples/datapower_silent_config.txt。
2. 要配置 DataPower 代理程序以监视设备，请输入以下属性：

设备主机
输入设备的主机名或 IP 地址。例如，**SOAP_HOST.ManageSystemName=** *datapower01*。

XML 管理接口端口
输入 XML 管理接口的端口号。缺省值是 5550。例如，**DP_PORT.ManageSystemName=** *5550*。

用户标识
输入用于连接到设备的用户标识。缺省值为 admin。例如，**DP_UID.ManageSystemName=** *admin*。

密码
输入用户标识的密码。例如，**DP_PASSWORD.ManageSystemName=** *password*。

SSL 代理概要文件
如果设备的 XML 管理接口配置为使用 SSL 代理概要文件，请输入此概要文件的公用证书的绝对路径。例如，

```
the location of the .pem file exported from datapower appliances/mycert.pem
```

其中，*the location of the .pem file exported from datapower appliances* 是公用证书的绝对路径。要导出公用证书，请参阅导出公用证书。

SSL 代理选项
对于 **SSL 代理选项**，如果将受监视设备的 XML 管理接口配置为使用定制 SSL 代理概要文件，那么设置为 Yes。否则，将其设置为 No。例如，**DP_SSL_OPTION.ManageSystemName1=** Yes。

要点: ManageSystemName 是唯一的。必须在所有条目中将其替换为您自己的系统名称。如果想要监视多个设备，请复制并重复所显示的用于监视设备的步骤。请务必设置适合的 ManageSystemName 和 DataPower 设备参数。

3. 转到该代理程序的安装目录，并运行以下命令以启动该代理程序：

```
./datapower-agent.sh start instance_name
```

下一步做什么

- 要检查已配置代理程序实例的名称和设置，请运行 **./cinfo -s bn** 命令。
- 您可验证 DataPower 代理程序 数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。
- 要在 Cloud APM 控制台中显示事务跟踪数据，请配置 DataPower 代理程序的事务跟踪。有关指示信息，请参阅配置 DataPower 代理程序的事务跟踪。
- 要在不同级别显示监视，请相应地配置 DataPower 设备。有关指示信息，请参阅资源监视、中间件事务跟踪和 DataPower 设备的实例级别事务跟踪。

配置 DataPower 代理程序的事务跟踪

要在中间件和拓扑仪表板中显示 DataPower 设备的事务跟踪数据，必须为 DataPower 代理程序启用事务跟踪。

开始之前

- 安装 DataPower 代理程序并将其配置为连接到 DataPower 设备。
- 在 DataPower 设备中针对 SOAP 或 ARM 启用监视。

过程

要启用 DataPower 代理程序 的事务跟踪，请完成下列步骤：

1. 在导航栏中，单击 **系统配置** > **代理程序配置**。
2. 在 **DataPower** 选项卡上，选择要为其启用事务跟踪的代理程序实例。
3. 选择**操作** > **设置事务跟踪** > **已启用**来启用事务跟踪。**事务跟踪**列中的代理程序状态将更新为**已启用**。

结果

您已为所选择的代理程序实例启用了事务跟踪。

下一步做什么

要在中间件和拓扑仪表板中查看 DataPower 设备的数据，现在必须将要监视的设备添加到应用程序性能仪表板。有关将 DataPower 设备添加到应用程序性能仪表板的更多信息，请参阅第 81 页的『将中间件应用程序添加到应用程序性能仪表板』。

注: 如果使用集成服务并想要监视在 IBM Integration Bus 与 DataPower 之间传输的数据，需要进行其他配置来显示准确的汇总事务拓扑。IBM Integration Bus 代理程序 不能包含对没有 SOAP 包络的 SOAP 消息的关联支持。SOAPRequest 节点、SOAPAsyncRequest 节点和 SOAPReply 节点可以接受没有 SOAP 包络的消息作为输入消息。对于这些节点，拓扑视图中不会显示从某个调解到下游调节或应用程序服务器的关系。为避免发生此问题，请将 SOAPEnvelope 节点正好插入到 IBM Integration Bus 消息流中的 SOAPRequest、SOAPAsyncRequest 或 SOAPReply 节点之前，并针对 SOAPEnvelope 节点选择**创建新包络**选项，然后为 SOAP 消息添加 SOAP 包络。

配置 Db2 监视

Monitoring Agent for Db2 监视 Db2 服务器 的可用性和性能。您可以从 Cloud APM 控制台监视多个服务器；一个 Db2 代理程序实例监视一个服务器。Db2 代理程序还支持远程监视。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Db2 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

Db2 代理程序是一个多实例代理程序，您必须先创建实例，然后手动启动该代理程序。

“受管系统名称”包含指定的代理程序实例名称，例如，*instance_name*:*host_name*:*pc*。

其中：

· *pc* 是两个字符的产品代码。

· *instance_name* 是代理程序实例名称，并且其必须与要监视的 Db2 实例名称相同。

受管系统名称至多可以包含 32 个字符。指定的代理程序实例名称最多可包含 8 个字符（不包括主机名的长度）。例如，如果指定 DB2inst1 作为代理程序实例名称，那么受管系统名称为 DB2inst1:hostname:ud。

要点: 如果指定较长的代理程序实例名称，那么会截断“受管系统名称”，并且代理程序代码无法正确显示。

要避免配置代理程序时发生许可权问题，请确保使用相同的 root 用户或用于安装代理程序的非 root 用户。如果作为所选用户身份安装代理程序并以另一用户身份配置代理程序，请参阅第 156 页的『以非 root 用户身份配置代理程序』。如果以一个选中的用户身份安装和配置了代理程序，并且想要以其他用户身份启动代理程序，请参阅第 859 页的『以非 root 用户身份启动代理程序』。

运行配置脚本以创建实例和更改配置设置。您可以先编辑 Db2 静默响应文件，然后再运行配置脚本以绕过其他情况下必需的提示和响应。

配置 Db2 代理程序 后，请确保使用具有受监视实例的 Db2 SYSADM 权限的用户标识来启动代理程序。代理程序需要 SYSADM 权限才能开启所有监视开关并收集监视数据。因此，代理程序必须由具有 SYSADM 权限的用户启动。请使用具有 SYSADM 权限的实例所有者用户来启动代理程序。

以下指南针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

过程

要使用缺省设置配置代理程序，请完成下列步骤：

1. 运行以下命令，其中 *instance_name* 是要对实例指定的名称：

   ```
   install_dir/bin/db2-agent.sh config instance_name
    install_dir/samples/db2_silent_config.txt
   ```

   代理程序实例名称 *instance_name* 总是与要监视的 Db2 实例名称相同。有关现有代理程序实例的更多详细信息，请参阅第 155 页的『代理程序配置页面』

2. 运行以下命令以启动 Db2 代理程序 代理程序：

   ```
   install_dir/bin/db2-agent.sh start instance_name
   ```

下一步做什么

· 授权该 Db2 用户查看 Db2 的某些属性数据。有关授予这些权限的信息，请参阅第 214 页的『授予查看 Db2 度量的特权』。

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

开始之前

在开始配置 Db2 代理程序以执行本地或远程监视之前，对于远程监视，请确保完成下列任务。

· 设置用于远程监视的客户机/服务器环境，请参阅第 217 页的『远程监视先决条件』。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 IBM Cloud Application Performance Management 窗口中，右键单击 **Monitoring Agent for DB2**，然后单击**配置代理程序**。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。

   要点: 对于本地监视，代理程序实例名称必须与受监视 Db2 实例的名称匹配。

   对于远程监视，代理程序实例名称必须是唯一的目录节点名。
4. 在 **Monitoring Agent for DB2** 窗口中，完成以下步骤：
   a) 在**用户名**中，输入 Db2 实例的用户名。

      对于本地 Db2，请输入 Db2 实例所有者的名称。

      对于远程 Db2，请输入来自远程 Db2 机器的实际 Db2 实例所有者名称。

      要点: 此参数是监视远程 Db2 实例所必需。

   b) 在**密码**中，输入 Db2 实例的密码。

      对于本地 Db2，请输入 Db2 实例所有者的密码。

      对于远程 Db2，请输入来自远程 Db2 机器的实际 Db2 实例所有者密码。

      要点: 此参数是监视远程 Db2 实例所必需。

   c) 在 **DB2Customized SQL 定义文件**字段中，输入 SQL 定义文件的完整文件路径。如果 SQL 定义文件在缺省目录中，请保留此字段为空。否则，请输入该文件的完整文件路径名。带有路径的缺省文件名如下所示：

      Linux AIX `CANDLEHOME/config/kudcussql.properties`

      Windows `CANDLEHOME\TMAITM6_x64\kudcussql.properties`

   d) 在 **db2diag 日志文件路径**字段中，输入 db2diag 日志文件的目录路径。如果 db2diag 日志文件在缺省目录中，请保留此字段为空。否则，请输入该目录的路径。缺省目录路径如下所示：

      Linux AIX `/home/DB2owner_home_dir/sqllib/db2dump`

      Windows `C:\ProgramData\IBM\DB2\DB2COPY\DB2INSTANCENAME`

      注: 此参数不适用于远程监视。

   e) 在**以正则表达式表示的 MSGID 过滤器**字段中，输入 *MSGID* 以过滤诊断日志。MSGID 由消息类型、消息号和严重性级别组成。使用正则表达式可根据消息类型、消息编号或严重性级别来过滤日志，例如，`ADM1\d*1E|ADM222\d2W`。

   f) 从**允许监视远程主机中的分区**列表中，选择是以指定 Db2 代理程序可以监视远程主机中的分区。

   g) 从**允许监视所有数据库**列表中，选择是以指定 Db2 代理程序可以监视所有数据库。

   h) 单击**确定**。

      代理程序实例将显示在 IBM Cloud Application Performance Management 窗口中。
5. 运行下列步骤，以配置远程监视。

a) 打开 *install_dir*\TMAITM6_x64\KUDENV_<instanceName>。

b) 将 *KUD_DB2_CLIENT_INST* 设置为编目远程 Db2 服务器实例的 Db2 客户机实例名称。

6. 右键单击 **Monitoring Agent for DB2** 实例，然后单击**启动**。

下一步做什么

- 授权该 Db2 用户查看 Db2 代理程序的某些属性数据。有关授予这些权限的信息，请参阅第 214 页的『授予查看 Db2 度量的特权』。
- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Linux 或 UNIX 系统上配置代理程序

在 Linux 系统上运行配置脚本以配置代理程序。

开始之前

在开始配置 Db2 代理程序以执行本地或远程监视之前，对于远程监视，请确保完成下列任务。

- 设置用于远程监视的客户机/服务器环境，请参阅第 217 页的『远程监视先决条件』。

过程

1. 运行命令 *install_dir*/bin/db2-agent.sh config *instance_name*

   其中，*instance_name* 是要对实例指定的名称：

   要点: 对于本地监视，代理程序实例名称必须与受监视 Db2 实例的名称匹配。

   对于远程监视，必须与受监视远程 Db2 服务器 实例的本地编目节点匹配。
2. 当提示您提供以下参数的值时，请按 Enter 键以接受缺省值，或指定一个值然后按 Enter 键：
   - 用户名
   - 密码
   - DB2® SQL 路径
   - 诊断日志路径
   - 诊断日志消息标识过滤器
   - 监视远程分区
   - 监视所有数据库
3. 运行以下命令以启动代理程序：

   对于本地监视，请以 Db2 实例所有者用户身份运行 *install_dir*/bin/db2-agent.sh start *instance_name*。

   对于远程监视，请以编目远程 Db2 服务器 实例的 Db2 客户机实例的实例所有者身份运行 *install_dir*/bin/db2-agent.sh start *node_name*。

下一步做什么

- 授权该 Db2 用户查看 Db2 代理程序的某些属性数据。有关授予这些权限的信息，请参阅第 214 页的『授予查看 Db2 度量的特权』。
- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置代理程序

使用静默响应文件配置代理程序，而无需对运行配置脚本时的提示进行响应。在 Windows 和 Linux 系统上，可以使用静默响应文件来配置代理程序。

开始之前

在开始配置 Db2 代理程序以执行本地或远程监视之前，对于远程监视，请确保完成下列任务。

· 设置用于远程监视的客户机/服务器环境，请参阅第 217 页的『远程监视先决条件』。

关于此任务

静默响应文件包含配置参数。您可以编辑响应文件中的参数值，并运行配置脚本来创建代理程序实例和更新配置值。

过程

1. 在文本编辑器中，打开位于以下路径的 db2_silent_config.txt 文件：

   Linux AIX *install_dir*/samples/db2_silent_config.txt

   Windows *install_dir*\tmaitm6_x64\samples\db2_silent_config.txt

2. 在响应文件中，对下列参数指定值：

   · 在**用户名**中，输入 Db2 实例的用户名。

   对于本地 Db2，请输入 Db2 实例所有者的名称。

   对于远程 Db2，请输入来自远程 Db2 机器的实际 Db2 实例所有者名称。

   要点: 此参数是监视远程 Db2 实例所必需。

   · 在**密码**中，输入 Db2 实例的密码。

   对于本地 Db2，请输入 Db2 实例所有者的密码。

   对于远程 Db2，请输入来自远程 Db2 机器的实际 Db2 实例所有者密码。

   要点: 此参数是监视远程 Db2 实例所必需。

   · 对于 **DB2 SQL 路径**参数，如果 SQL 定义文件位于缺省目录中，请将此字段保留为空。否则，请输入正确的目录路径。SQL 定义文件位于下列缺省路径中：

   Linux AIX CANDLEHOME/config/kudcussql.properties。例如，**KUD_DB2_SQL_PATH=** /opt/ibm/apm/agent/config/kudcussql.properties

   Windows CANDLEHOME\TMAITM6_x64\kudcussql.properties。例如，**KUD_DB2_SQL_PATH=** C:\IBM\ITM\TMAITM6_x64\kudcussql.properties

   · 对于**对话框路径**参数，如果 db2diag 日志文件位于缺省目录中，那么请保留此字段为空白。否则，请输入正确的目录路径。日志文件位于下列缺省路径中：

   Linux AIX /home/DB2owner_home_dir/sqllib/db2dump
   例如，**KUD_DIAGLOG_PATH=** /home/db2inst1/sqllib/db2dump.

   Windows Windows Install_Driver:\ProgramData\IBM\DB2\DB2COPY \DB2INSTANCENAME
   例如，**KUD_DIAGLOG_PATH=** C:\ProgramData\IBM\DB2\DB2COPY1\DB2

   注: 此参数不适用于远程监视。

   · 对于**对话框消息标识过滤器**参数，请指定 *MSGID* 以过滤诊断记录。MSGID 由消息类型、消息号和严重性级别组成。您还可使用正则表达式，例如，**KUD_DIAGLOG_MSGID_FILTER=** ADM1\d*1E|ADM222\d2W。

   · 对于**监视远程分区**参数，请输入 Yes 以指定远程主机中的 Db2 代理程序 受监视分区。例如，**KUD_MONITOR_REMOTE_PARTITIONS=** *Yes*。

· 对于**监视所有数据库**参数，请输入 Yes 以指定需要 Db2 代理程序 监视所有数据库。例如，**KUD_MONITOR_ALL_DATABASES=** *Yes*。

3. 保存并关闭 db2_silent_config.txt 文件，然后运行下列命令

   Linux AIX *install_dir*/bin/db2-agent.sh config *instance_name* *install_dir*/samples/db2_silent_config.txt

   Windows *install_dir*\bin\db2-agent.bat config*instance_name* \tmaitm6_x64\samples\db2_silent_config.txt

   <instance_name> 是：

   · 对于监视本地 Db2 服务器：要监视的 Db2 服务器实例名称。

   · 对于监视远程 Db2 服务器：远程 Db2 服务器实例的目录节点名称。

   要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

4. 对于 Windows，请打开 CANDLEHOME\TMAITM6_x64\KUDENV_<instance_name> 文件。编辑 KUD_DB2_CLIENT_INST 行，使其如下所示：KUD_DB2_CLIENT_INST=<编目远程 Db2 服务器实例的客户机实例名称>

5. 运行以下命令以启动代理程序：

   Linux AIX *install_dir*/bin/db2-agent.sh start *instance_name*

   Windows *install_dir*\bin\db2-agent.bat start instance_name

   切记: 从 UNIX 或 Linux 监视远程 Db2 服务器 实例时，必须以编目远程服务器实例的客户机实例所有者身份执行该命令。

下一步做什么

· 授权该 Db2 用户查看 Db2 代理程序的某些属性数据。有关授予这些权限的信息，请参阅第 214 页的『授予查看 Db2 度量的特权』。

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

授予查看 Db2 度量的特权

要监视 Db2 资源，Db2 用户必须具有受监视实例的 Db2 SYSADM、SYSCTRL、SYSMAINT 和 SYSMON 权限，才能查看 Db2 代理程序的某些属性的数据。

关于此任务

要在仪表板上查看代理程序为所有属性收集的监视数据，Db2 用户必须具有特定的特权。要将这些特权分配给 Db2 用户，请运行以下位置的脚本文件：

Linux AIX *install_dir*/config/KudGrantUserPermissions.sh

Windows *install_dir*\TMAITM6_x64\KudGrantUserPermissions.bat

具有 SYSADM 权限的 Db2 用户可以运行此脚本，以将特权授予自身或任何其他 Db2 用户。对于 Db2 实例，请使用已具有 SYSADM 权限的实例所有者来运行此脚本，以将其他许可权授予自身，或将所有许可权授予任何其他 Db2 用户。

过程

1. 对于本地监视，请参阅以下步骤。

   a) 在 Db2 代理程序安装所在的系统上，打开 Db2 命令行界面。

   b) 运行以下命令，其中 *instance_name* 是 Db2 实例的名称，*username* 是 Db2 用户的名称：

   Linux AIX *install_dir*/config/KudGrantUserPermissions.sh *instance_name username*

Windows *install_dir*\TMAITM6_x64\KudGrantUserPermissions.bat *instance_name username*

注: 对于 Windows 系统，命令中的 *username* 是可选项。如果命令中没有指定用户名，那么特权将分配给缺省用户 (system)。

2. 对于远程监视，请参阅以下步骤。
   a) 将 KudGrantUserPermissions.sh（对于 Unix 或 Linux）和 KudGrantUserPermissions.bat（对于 Windows）从代理程序工作站上的 *install_dir*/TMAITM6_x64/ 复制到 Db2 服务器机器。
   b) 以 Db2 实例所有者用户身份运行以下命令，其中 *instance_name* 是 Db2 实例的名称，而 *username* 是 Db2 用户的名称:

Linux AIX ./KudGrantUserPermissions.sh *instance_name username*

Windows KudGrantUserPermissions.bat *instance_name username*

切记: 对于 Windows 上的远程 Db2 监视，*username* 必须是在客户机工作站上配置 Db2 代理程序 期间提供的用户名。

配置本地环境变量

可以配置本地环境变量以更改 Db2 代理程序的行为。

过程

1. 在 Windows 系统上，单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，从**操作**菜单中，单击**高级 > 编辑 ENV 文件**。
3. 在 Linux 或 AIX 系统上，转至命令行并从 install_dir/config 目录编辑 ud.environment 文件。其中，install_dir 是代理程序安装目录。

   注: ud.environment 文件是隐藏文件。
4. 在环境变量文件中，输入环境变量的值。

   有关可配置的环境变量的信息，请参阅第 215 页的『本地环境变量』。

本地环境变量

您可以通过配置本地环境变量来更改 Db2 代理程序的行为。

用于为表空间数据集定义数据收集方法的变量

要设置表空间数据集的数据收集方法，请使用下列环境变量:

- **KUD_T1_BY_SQL**：利用此变量，可使用 SQL 查询来设置表空间数据集的数据收集方法。要使用 SQL 查询来启用数据收集，请将此变量的值设置为 Y。要使用快照方法来收集表空间数据集的数据，请将此变量的值设置为 N。此变量的缺省值为 N。

  要点: 要使用 SQL 查询来收集数据，Db2 版本必须是 9.7 或更高版本。另外，启动 Db2 代理程序的用户必须具有所有数据库的 SYSADM 权限。
- **KUD_T1_DISABLE**：使用此变量可对表空间数据集禁用数据收集。要对表空间数据集启用数据收集，请将此变量的值设置为 N。要对表空间数据集禁用数据收集，请将此变量的值设置为 Y。此变量的缺省值为 N。

用于从数据收集中排除高速缓存工具 (CF) 节点的变量

在 pureScale® 环境中，要从数据收集算法中排除高速缓存工具 (CF) 节点，请使用 **DB2_CF_PARTITION_NUMS** 变量。在代理程序环境文件中，将 **DB2_CF_PARTITION_NUMS** 变量设置为 DB2_CF_PARTITION_NUMS=<CF node number>。例如，DB2_CF_PARTITION_NUMS=1。对于多个 CF 节点，请将 **DB2_CF_PARTITION_NUMS** 变量值设置为使用 # . : , ; | @ 中任何特殊符号作为分隔符的列表。例如，DB2_CF_PARTITION_NUMS=12,13,23,34。没有为此变量设置缺省值。

用于对“Db2 表”数据集的数据收集进行限制的变量

要设置收集“Db2 表”数据集的数据时 Db2 代理程序 所必须返回的最大行数，请使用 **KUD_TABLE_NUMBER** 环境变量。缺省值为 10000。

用于设置定制 SQL 属性文件的重新装入时间间隔的变量

要设置定制 SQL 属性文件的重新装入时间间隔（以秒计），请使用 **KUD_CUS_SQL_INTERVAL** 变量。缺省值为 20 秒。

用于对“代理程序事件”数据集的数据收集限制行数的变量

要设置“代理程序事件”数据集的数据收集行数，请使用 **KUD_AGENT_EVENT_CACHE** 变量。“代理程序事件”数据集提供关于预定义事件和已触发事件的详细信息，并确定与受监视数据库的运行状况相关的问题。缺省值是 50。

用于对“Db2 日志记录”数据集的数据收集限制行数的变量

要设置“Db2 日志记录”数据集的数据收集行数，请使用 **KUD_DBHISTORY_MAXROW** 变量。“Db2 日志记录”数据集提供 Db2 归档日志的相关历史信息。缺省值是 500。

用于为“Db2 诊断日志”数据集定义数据收集的变量

要设置“Db2 诊断日志”数据集的数据收集方法，请使用下列环境变量：

- **KUD_DIAGLOG_BY_TABLE**：使用此变量可设置“Db2 诊断日志”数据集的数据收集方法。如果此变量的值设置为 Y，那么将使用 SQL 查询来收集“Db2 诊断日志”数据集的数据。如果此变量的值设置为 N，那么将通过解析 db2diag.log 来收集“Db2 诊断日志”数据集的数据。此变量的缺省值为 Y。

  要点: 要使用 SQL 查询来收集数据，Db2 版本必须是 10 或更高版本。
- **KUD_DIAGLOG_TAILCOUNT**：使用此变量可定义收集“Db2 诊断日志”数据集的数据时，Db2 代理程序所解析的 db2diag.log 文件行数。此变量用于限制 Db2 代理程序对 Db2 代理程序 日志文件的处理，以便仅监视最新的消息和事件。此变量的缺省值为 1000。
- **KUD_DIAGLOG_CACHE**：使用此变量可限制仪表板上针对“Db2 诊断日志”数据集显示的日志记录数。此变量的缺省值为 20。
- **KUD_DIAGLOG_INTERVAL**：使用此变量可定义 db2diag.log 文件的重新装入时间间隔（以秒计），以用于“Db2 诊断日志”数据集的数据收集。此变量的缺省值为 30 秒。
- **KUD_DISABLE_DIAGLOG**：使用此变量可对“Db2 诊断日志”数据集禁用数据收集。要对“Db2 诊断日志”数据集启用数据收集，请将此变量的值设置为 N。要对“Db2 诊断日志”数据集禁用数据收集，请将此变量的值设置为 Y。此变量的缺省值为 N。

用于设置查询超时时间间隔的变量

如果 SQL 查询花费很长时间才能完成，那么将影响 Db2 代理程序的性能。要为 Db2 代理程序设置查询超时时间间隔，请使用 **KUD_QUERY_TIMEOUT** 变量。使用此变量可定义对于发送到 Db2 服务器的查询，Db2 代理程序等待接收响应的最大时间量（以秒计）。此变量的值必须小于 300 秒。此变量的缺省值为 45 秒。

用于为“Db2 数据库 01（已取代）”数据集定义数据收集的变量

不存在 ASN 模式时，代理程序不得触发 ASN 查询以收集“Db2 数据库 01（已取代）”数据集的数据。要允许执行 ASN 查询，请使用 **KUD_REPLICATION_ON** 变量。如果此变量的值设置为 Y，那么 Db2 代理程序将运行 ASN 查询，即使不存在 ASN 模式也是这样。如果此变量的值设置为 N，那么 Db2 代理程序不会运行 ASN 查询。此变量的缺省值为 Y。

使用快照方法收集数据时用于配置监视开关的变量

如果要使用快照方法来收集 Db2 代理程序监视数据，请对数据集启用 Db2 监视开关。要启用 Db2 监视开关，请使用 **KUD_MON_SWITCH_OVERRIDE** 变量。Db2 监视开关列表如下所示：

LOCK
锁定信息

SORT
排序信息

STATEMENT
SQL 语句信息

TABLE
表活动信息

TIMESTAMP
时间戳记信息

UOW
工作单元信息

如果此变量的值设置为 Y，那么 Db2 代理程序将保留 Db2 监视开关的配置设置。如果此变量的值设置为 N，那么 Db2 代理程序将启用所有监视开关以收集数据。此变量的缺省值为 N。

用于跟踪数据集的 Db2 快照缓冲区数据的变量

要使用快照方法来查看针对某个数据集所收集的数据，请使用 **KUD_SNAPSHOT_DUMPOUT** 变量。如果此变量的值设置为 Y，那么 Db2 代理程序会将属性组的快照缓冲区数据转储到代理程序日志文件。如果此变量的值设置为 N，那么 Db2 代理程序不会将快照缓冲区数据转储到代理程序日志文件。此变量的缺省值为 N。

用于通过数据集的快照缓冲区数据来跟踪 Db2 代理程序的变量

要使用针对数据集所收集的快照缓冲区数据来跟踪 Db2 代理程序，请使用 **KUD_SNAPSHOT_READIN** 变量。要启用 Db2 代理程序跟踪，请将此变量的值设置为 Y。要禁用 Db2 代理程序跟踪，请将此变量的值设置为 N。

用于为“锁定冲突”数据集定义数据收集方法的变量

要设置“锁定冲突”数据集的数据收集方法，请使用 **KUD_LOCKCONFLICT_BY_SQL** 变量。要使用 SQL 查询来收集“锁定冲突”数据集的数据，请将此变量的值设置为 Y。要使用快照方法来收集“锁定冲突”数据集的数据，请将此变量的值设置为 N。此变量的缺省值为 Y。

要点: 要使用 SQL 查询来收集数据，Db2 版本必须是 9.7 FP1 或更高版本。另外，启动 Db2 代理程序的用户必须具有所有数据库的 SYSADM 权限。

用于在 Windows 上监视远程 Db2 服务器的变量

KUD_DB2_CLIENT_INST：将此变量设置为编目远程 Db2 服务器实例的 Db2 客户机实例名称。仅当要对代理程序在 Windows 上的位置使用远程监视时，才需要设置此变量。

远程监视先决条件

您可使用 Monitoring Agent for Db2 进行远程监视。请参阅 Db2 的远程监视先决条件主题。

关于此任务

要执行 Db2 远程监视，必须先执行基本的 Db2 客户机/服务器环境设置。对于 Windows 以及 UNIX 或 Linux，请执行此设置。

要执行此设置，用户必须具有 Db2 SYSADM 或 SYSCTRL 权限。

切记: 请在代理程序工作站上运行除步骤 2 以外的所有步骤。

过程

1. 在 Db2 代理程序 工作站上，安装 Db2 客户机。此客户机的版本必须高于或等于所要监视的 Db2 服务器实例版本。

2. 确认 Db2 实例的通信协议为 TCPIP。

   a) 要进行验证，请在 Db2 命令行上运行 **db2set** 命令。

   b) 如果未设置为 TCPIP，请在 Db2 命令行上运行 **db2set DB2COMM=tcpip**。

   要点: 此步骤是在服务器端执行。

3. 使用下列命令对 Db2 代理程序工作站上的远程服务器实例进行编目。

   要点: 服务器实例将在客户机实例下进行编目。因此，请在客户机实例上运行下列命令。

   db2=>CATALOG TCPIP NODE<node_name> REMOTE <hostname/ip_address> SERVER <service_name/port_number>

   在 Db2 上，其中:

   a. *<node_name>* 代表客户机组件上 Db2 实例的本地昵称。

      注: 对于 UNIX 或 Linux，*<node_name>* 不得与同一工作站上的任何 Db2 客户机或 Db2 服务器实例名称相同。

   b. <hostname/ip_address> 代表 Db2 服务器 工作站的名称或 IP 地址。

   c. <service_name/port_number> 是 Db2 TCPIP 所配置在的服务名称或端口号。

   要将在远程服务器 **myserver** 的端口号 50000 上运行的 Db2 服务器实例“”编目为节点“db2node”“”，请从 Db2 命令行输入以下命令:

   db2 => CATALOG TCPIP NODE db2node REMOTE myserver SERVER 50000

   有关目录节点的更多详细信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.qb.client.doc/doc/t0005621.html

4. 如果 Db2 代理程序 工作站是 UNIX/Linux:

   · 创建与编目命令中使用的节点同名的用户

     发出命令

     useradd -g <group> -m -d <home_dir> <user> -p <password>

     其中

     – **<group>** 表示 DB2 UDB 实例所有者的组。

     – **<user>** 表示客户机工作站上的本地 **username**。**Userame** 必须与在代理程序机器上编目服务器实例所使用的节点名相同。

   · 检查在其下编目远程 Db2 服务器 实例的 Db2 客户机实例名称，并将新创建的用户的主目录的读、写和执行许可权分配给此实例的所有者。必须执行此步骤，才能使客户机 Db2 环境可用于远程节点上的操作。

   · 发出命令

     chmod -R 775 /home/<nodename>

     其中

     – **<nodename>** 表示客户机组件上 Db2 实例的本地用户名

5. 在客户机实例上，编目 Db2 代理程序 工作站上存在的所有要监视的数据库。

   在 Db2 CLP 中发出命令以对数据库进行编目。

   CATALOG DATABASE <db_name> AS <db_alias> AT NODE <node_name>authentication server

   a. <db_name> 代表服务器数据库名称。

   b. <db_alias> 代表数据库在 Db2 客户机上的本地昵称。

   c. <node_name> 代表数据库编目所在客户机组件上 Db2 实例的本地昵称。

   要对目录节点“db2node”上名为“sample”的数据库进行编目，且将别名设置为“dbAlias1”，请从 Db2 提示符处输入以下命令“”“”“”。

```
db2 => CATALOG DATABASE sample AS dbAlias1 AT NODE db2node authentication
server
```

配置 Hadoop 监视

您必须配置 Monitoring Agent for Hadoop，以使此代理程序可以收集其所监视的 Hadoop 集群的数据。此代理程序可监视单节点 Hadoop 集群和多节点 Hadoop 集群。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Hadoop 代理程序 的软件产品兼容性报告 (SPCR)。

确保可以从安装有 Hadoop 代理程序的计算机解析下列主机：

- 所有要配置的 Hadoop 主机，例如 NameNode 和 ResourceManager，等等
- 仅具有 NodeManager 角色的 Hadoop 主机

例如，您可完成下列步骤以解析主机：

- 将所有 Hadoop 主机的 IP 地址、主机名和标准域名添加到位于以下路径的 hosts 文件中：
  - Windows `C:\Windows\System32\drivers\etc\hosts`
  - Linux AIX `/etc/hosts`
- 添加 Hadoop 主机所在域中安装有 Hadoop 代理程序的计算机。

切记： 要监视使用基于 Kerberos SPNEGO 的认证来保护的 Hadoop 集群，请确保可以从安装 Hadoop 代理程序的计算机解析所有主机。

关于此任务

Hadoop 代理程序是一个单实例代理程序。安装此代理程序之后，您必须对其进行手动配置。Hadoop 代理程序可以在 Windows、Linux 和 AIX 系统上配置。

切记：

- 在单节点 Hadoop 集群中，根据 Hadoop 集群的配置，同一个节点会扮演所有的角色，例如 NameNode、ResourceManager 和辅助 NameNode。但是，在多节点 Hadoop 集群中，这些角色由不同的 Hadoop 节点扮演。
- 配置此代理程序时，它会自动检测所监视 Hadoop 集群中的 DataNode 和 NodeManager。

从基于套接字的代理程序（8.1.2 FP2 或更早版本）升级到基于 REST API 的代理程序（8.1.3 或更高版本）时，请完成后续主题中指定的配置步骤。但是，在配置代理程序时，请确保根据下列准则来指定主机名。

- 您指定的各种守护进程（NameNode 和 ResourceManger，等等）的主机名必须与您为基于套接字的代理程序配置的主机名相同（大小写和格式均相同）。
- 指定主机名时，必须使用标准域名 (FQDN)。例如，hos1.ibm.com。如果 FQDN 的长度超过 25 个字符，请仅指定主机名的简称（不包含域名）。例如，如果主机的 FQDN 是 *myhadoopclustersetupnode.ibm.com*，那么主机名的简称为 myhadoopclustersetupnode。

配置所升级的代理程序并在 Cloud APM 控制台中查看数据之后，请撤销您在 Hadoop 代理程序的 hadoop-metrics2.properties 文件中所作的更改。有关详细信息，请参阅第 961 页的『升级代理程序』。

在 Windows 系统上，可以非管理员用户身份来运行 Hadoop 代理程序。但是，这样的用户需要特定的许可权，才能在仪表板中查看数据。有关如何授予此许可权的信息，请参阅第 226 页的『向非管理员用户授予许可权』。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

在 Windows 系统上配置代理程序

您可以使用 **IBM Performance Management** 窗口在 Windows 系统上配置代理程序。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Hadoop**。
3. 单击**配置代理程序**。

   注意: 如果禁用了**配置代理程序**，单击**重新配置**。

   这将打开**配置 Monitoring Agent for Hadoop** 窗口。
4. 要通过启用基于 Kerberos SPNEGO 的认证来监视 Hadoop 集群，请完成以下步骤：
   a) 在**是否启用 Hadoop 集群中基于 HTTP 的 Hadoop 服务的基于 SPNEGO 的 Kerberos 认证**下面，单击**是**。

      如果尚未执行基于 Kerberos SPNEGO 的认证以保护 Hadoop 集群中基于 HTTP 的 Hadoop 服务的 REST 端点，请单击**否**，然后**领域名称**、**KDC 主机名**、**SPNEGO 主体名称**和 **SPNEGO 密钥表文件**字段的值可以保留为空白。

   b) 在**领域名称**字段中，输入用于创建服务主体的 Kerberos 领域的名称。
      通常，领域名称与域名相同。例如，如果计算机位于 `tivoli.ibm.com` 域中，那么 Kerberos 领域名称为 `TIVOLI.IBM.COM`。此名称区分大小写。

   c) 在 **KDC 主机名**字段中，输入所指定领域的密钥分发中心 (KDC) 主机的标准域名 (FQDN)。
      您还可以指定 KDC 主机的 IP 地址来代替 FQDN。对于 Active Directory KDC，域控制器是 KDC 主机。

   d) 在 **SPNEGO 主体名称**字段中，输入 Kerberos 主体的名称，其用于访问基于 HTTP 服务的 SPNEGO 认证的 REST 端点。
      此名称区分大小写，名称格式为 HTTP/*fully_qualified_host_name*@*kerberos_realm*

   e) 在 **SPNEGO 密钥表文件**字段中，输入具有完整路径的 SPNEGO 服务的密钥表文件名称，或单击**浏览**并选择该文件。
      密钥表文件包含 Kerberos 服务主体和密钥的名称。此文件可直接访问 Hadoop 服务，而不需要每个服务的密码。此文件可以位于以下路径：`etc/security/keytabs/`
      确保 SPNEGO 主体名称和密钥表文件属于同一个主机。例如，如果主体名称为 *HTTP/abc.ibm.com@IBM.COM*，那么使用的密钥表文件必须属于 *abc.ibm.com* 主机。
      如果在远程计算机上安装代理程序，请将此主体的密钥表文件复制到任何路径中的远程计算机，然后在 **SPNEGO 密钥表文件**字段中指定此路径。

   f) 单击**下一步**。
5. 要监视启用 HTTPS/SSL 的 Hadoop 集群，请完成以下步骤：
   a) 在**是否启用 Hadoop 集群 SSL** 下，单击**是**
      如果不想启用 SSL 的 Hadoop 集群，那么选择**否**，然后输入**信任库文件路径**的值，**信任库密码**字段可以保留为空。

   b) 在**信任库文件路径**中，选择在本地机器上存储的信任库文件。
      可以将此文件从 Hadoop 集群复制到本地机器，然后用于配置。

   c) 在 **TrustStore Password** 中，输入配置信任库文件时创建的密码。
6. 要为 Hadoop 集群指定参数值，请完成下列步骤：
   a) 在**唯一 Hadoop 集群名称**字段中，输入指示 Hadoop 版本和风格的 Hadoop 集群的唯一名称，此字段的最大字符限制为 12。

   b) 在 **NameNode 主机名**字段中，输入运行 NameNode 的守护进程的节点主机名。

   c) 在 **NameNode 端口**字段中，输入与 NameNode 的守护进程相关联的端口号。缺省端口号为 50070。

   d) 在 **ResourceManager 主机名**字段中，输入运行 ResourceManager 的守护进程的节点主机名。

e) 在 **ResourceManager 端口**字段中，输入与 ResourceManager 的守护进程相关联的端口号。缺省端口号为 8088。

f) 可选： 在 **JobHistoryServer 主机名**字段中，输入运行 JobHistoryServer 的守护进程的节点主机名。

g) 可选： 在 **JobHistoryServer 端口**字段中，输入与 JobHistoryServer 的守护进程相关联的端口号。缺省端口号为 19888。

h) 可选： 在**附加 NameNode 主机名**字段中，输入运行备用 NameNode 或辅助 NameNode 的守护进程的主机名。

i) 可选： 在**附加 NameNode 端口**字段中，输入与备用 NameNode 或辅助 NameNode 的守护进程相关联的端口号。

切记: 如果附加 NameNode 是备用 NameNode，那么与备用 NameNode 守护进程相关联的缺省端口号是 50070。如果附加 NameNode 是辅助 NameNode，那么与辅助 NameNode 守护进程相关联的缺省端口号是 50090。

j) 单击**测试连接**，以验证与所指定主机名和端口的连接。

单击**测试连接**之后，在下列情况下，将显示相应的验证消息：

· 与指定的主机名和端口建立连接或者此连接失败。

· 主机名的值保留为空。

· 端口的值保留为空。

· 对端口号指定了非整数值。

请按验证消息的建议来更新配置值，并再次验证该连接。

k) 可选： 要在 Hadoop 集群中添加备用 ResourceManager，请在 **Hadoop 集群中的备用 ResourceManager** 下单击**是**。

稍后系统将提示您添加备用 ResourceManager 的详细信息。

l) 可选： 要在 Apache Ambari 管理的 Hadoop 集群中监视 Hadoop 服务，请在**为基于 Ambari 的 Hadoop 安装监视 Hadoop 服务**下单击**是**，然后单击**下一步**。

7. 可选： 要指定 Ambari 服务器的详细信息以监视 Hadoop 服务，请完成以下步骤：

a) 在 **Ambari 服务器主机名**字段中，输入运行 Ambari 服务器的主机名。

b) 在 **Ambari 服务器端口**字段中，输入与 Ambari 服务器相关联的端口号。缺省端口号为 8080。

c) 在 **Ambari 用户的用户名**字段中，输入 Ambari 用户的名称。

d) 在 **Ambari 用户的密码**字段中，输入 Ambari 用户的密码。

e) 单击**下一步**。

8. 要为 Java 参数指定值，请完成下列步骤：

a) 从 **Java 跟踪级别**列表中，选择 Java 提供程序所使用的跟踪级别值。

b) 可选： 在 **JVM 自变量**字段中，为 Java 虚拟机指定一列自变量。

自变量列表必须兼容随代理程序一起安装的 Java 版本。

c) 单击**下一步**。

9. 可选： 要添加备用 ResourceManager，请完成下列步骤：

a) 单击**新建**。

b) 在**备用 ResourceManager 主机名**字段中，输入运行备用 ResourceManager 的守护进程的节点主机名。

c) 在**备用 ResourceManager 端口**字段中，输入与备用 ResourceManager 的守护进程相关联的端口号。缺省端口号为 8088。

d) 单击**测试连接**，以验证与所指定主机名和端口号的连接。

单击**测试连接**之后，在下列情况下，将显示相应的验证消息：

· 与指定的主机名和端口建立连接或者此连接失败。

· 主机名的值保留为空。

· 端口的值保留为空。

· 对端口号指定了非整数值。

请按验证消息的建议来更新配置值，并再次验证该连接。

e) 重复步骤 a、b 和 c，以添加更多备用 ResourceManager。
如果要移除任何备用 ResourceManager，请单击对应于所要移除的备用 ResourceManager 的**删除**。

f) 单击**下一步**。

10. 在**外部 Jar 的类路径**字段中，指定 JAR 文件的类路径。
这个类路径将添加到代理程序所生成的类路径。您可保留此字段为空。
11. 单击**确定**。
指定的配置设置已保存完毕。
12. 右键单击 **Monitoring Agent for Hadoop**，然后单击**开始**。

下一步做什么

1. 启用子节点事件以查看 Hadoop 代理程序的事件触发阈值。有关启用子节点事件的信息，请参阅第 225 页的『配置仪表板以查看 Hadoop 事件』。
2. 登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Linux 和 AIX 系统上配置代理程序

运行配置脚本并对提示做出响应以在 Linux 和 AIX 系统上配置代理程序。

过程

1. 在命令行上，运行以下命令：`install_dir/bin/hadoop-agent.sh config`
其中，*install_dir* 是 Hadoop 代理程序的安装目录。
该代理程序安装在以下缺省安装目录中：
`/opt/ibm/apm/agent`
2. 当命令行显示以下消息时，请输入 1 以继续执行配置步骤并按 Enter 键。
`Edit “Monitoring Agent for Hadoop” setting? [1= yes, 2= No]`
3. 当命令行显示以下消息时，请输入 1 以指定监视 Hadoop 集群的值，并启用基于 SPNEGO 的 Kerberos 认证，然后按 Enter 键。否则，请输入 2 并按 Enter 键，您可以将**领域名称**、**KDC 主机名**、**SPNEGO 主体名称** 和 **SPNEGO 密钥表文件**字段保留为空值：
`Is Kerberos SPNEGO-based authentication for HTTP based Hadoop services in Hadoop cluster enabled\: [ 1=Yes, 2=No (default is: 2)`
   a) 对于**领域名称**参数，输入用于创建服务主体的 Kerberos 域的名称。
   通常，领域名称与域名相同。例如，如果计算机位于 `tivoli.ibm.com` 域，那么 Kerberos 领域名称为 `TIVOLI.IBM.COM`. 此名称区分大小写。
   b) 在 **KDC 主机名**字段中，输入所指定领域的密钥分发中心 (KDC) 主机的标准域名 (FQDN)。您还可以指定 KDC 主机的 IP 地址来代替 FQDN。对于 Active Directory KDC，域控制器是 KDC 主机。
   c) 对于 **SPNEGO 主体名称**参数，输入 Kerberos 主体的名称，其用于访问基于 HTTP 服务的 SPNEGO 认证的 REST 端点。
   该名称区分大小写，并且名称格式为 HTTP/*fully_qualified_host_name*@*kerberos_realm*
   d) 对于 **SPNEGO 密钥表文件**参数，输入具有完整路径的 SPNEGO 服务的密钥表文件名称。
   密钥表文件包含 Kerberos 服务主体和密钥的名称。此文件可直接访问 Hadoop 服务，而不需要每个服务的密码。该文件可位于以下路径：`etc/security/keytabs/`
   确保 SPNEGO 主体名称和密钥表文件属于同一主机。例如，如果主体名称为 *HTTP/abc.ibm.com@IBM.COM*，那么使用的密钥表文件必须属于 *abc.ibm.com* 主机。
   如果代理程序安装在远程计算机上，那么将主体的密钥表文件复制到远程计算机的任何路径中，然后为 **SPNEGO 密钥表文件**参数指定此路径。

4. 当命令行显示以下消息时，输入 1 以指定值来监视启用 SSL 的 Hadoop 集群，然后按 **Enter** 键。否则，请输入 2 并按 **Enter** 键，并且您可以将**信任库文件路径**和**信任库密码**字段保留为空值：
   `Is Hadoop Cluster SSL enabled [ 1=Yes, 2=No (default is: 2)`
   a) 在 **TrustStore file path** 中，指定本地机器上存储的信任库文件的路径。
      可以将此文件从 Hadoop 集群复制到本地机器，然后用于配置。
   b) 在 **TrustStore Password** 中，指定配置信任库文件时创建的密码。
5. 当提示您输入 Hadoop 集群的详细信息时，为下列各个参数指定适当的值，然后按 Enter 键。
   a) 在**唯一 Hadoop 集群名称**中，指定指示 Hadoop 版本和风格的 Hadoop 集群的唯一名称，此字段的最大字符限制为 12。
   b) 对于 **NameNode Hostname** 参数，请指定运行 NameNode 的守护进程的节点主机名，然后按 Enter 键。

      注意: 如果您未指定主机名就按 Enter 键，系统会提示您输入主机名。

   c) 对于 **NameNode Port** 参数，请指定与 NameNode 的守护进程相关联的端口号，然后按 Enter 键。缺省端口号为 50070。
   d) 对于 **ResourceManager Hostname** 参数，请指定运行 ResourceManager 的守护进程的节点主机名，然后按 Enter 键。

      注意: 如果您未指定主机名就按 Enter 键，系统会提示您输入主机名。

   e) 对于 **ResourceManager Port** 参数，输入与 ResourceManager 的守护进程相关联的端口号。缺省端口号为 8088。
6. 可选： 当提示您添加 Hadoop 集群的下列参数的详细信息时，请接受下列各个参数的缺省值或指定适当的值，然后按 Enter 键：
   a) 对于 **JobHistoryServer Hostname** 参数，输入运行 JobHistoryServer 的守护进程的节点主机名。
   b) 对于 **JobHistoryServer Port** 参数，输入与 JobHistoryServer 的守护进程相关联的端口号。缺省端口号为 19888。
   c) 对于 **Additional NameNode Hostname** 参数，输入运行辅助或备用 NameNode 的守护进程的节点主机名。
   d) 对于 **Additional NameNode Port** 参数，输入与辅助或备用 NameNode 的守护进程相关联的端口号。辅助 NameNode 的缺省端口号为 50090。对于备用 NameNode，缺省端口号为 50070。
7. 可选： 当命令行显示以下消息时，请输入 1 以添加高可用性集群的备用 ResourceManger 的详细信息，然后按 Enter 键。
   `Standby ResourceManager(s) in Hadoop Cluster [ 1=Yes, 2=No ] (default is: 2):`
8. 当命令行显示以下消息时，指定 1 并按 Enter 键以监视由 Ambari 管理的 Hadoop 集群中的 Hadoop 服务：
   `Monitoring of Hadoop services for Ambari based Hadoop installations [ 1=Yes, 2=No ] (default is: 2):`
   否则，保留缺省值 2 并按 Enter 键。如果启用 Hadoop 服务的监视，那么为 Ambari 服务器的下列各个参数分别指定一个值，然后按 Enter 键：
   a) 对于 **Ambari 服务器主机名**参数，输入运行 Ambari 服务器的主机名。
   b) 对于 **Ambari 服务器端口**参数，输入与 Ambari 服务器相关联的端口号。
      缺省端口号为 8080。
   c) 对于 **Ambari 用户的用户名**参数，输入 Ambari 用户的名称。
   d) 对于 **Ambari 用户的密码**参数，输入 Ambari 用户的密码。
9. 当命令行显示以下消息时，请选择相应的 Java 跟踪级别，然后按 Enter 键：

This parameter allows you to specify the trace level used by the Java providers Java trace level [1=Off, 2=Error, 3=Warning, 4=Information, 5=Minimum Debug, 6=Medium Debug, 7=Maximum Debug, 8=All] (default is: 2)

10. 可选：当命令行显示以下消息时，请为 Java 虚拟机指定自变量，然后按 Enter 键。自变量列表必须兼容随代理程序一起安装的 Java 版本。
    This parameter allows you to specify an optional list of arguments to the java virtual machine JVM arguments (default is:)
11. 可选：当命令行显示以下消息时，请输入 1 以添加备用 ResourceManger 的下列详细信息，然后按 Enter 键：
    Edit "Hadoop High Availability(HA) Cluster with Standby ResourceManagers" settings, [1=Add, 2=Edit, 3=Del, 4=Next, 5=Exit] (default is: 5): 1
    a) 对于 **Standby ResourceManager Hostname** 参数，输入运行备用 ResourceManger 的守护进程的节点主机名。
    b) 对于 **Standby ResourceManager Port**，输入与备用 ResourceManager 的守护进程相关联的端口号。缺省端口号为 8088。
    c) 出现提示时，输入 1 以添加更多备用 ResourceManager，并重复步骤 a 和 b，或输入 5 以转到下一步。
    - 要编辑特定备用 ResourceManager 的配置设置，请输入 4 并按 Enter 键，直至看到所需备用 ResourceManager 的主机名为止。
    - 要移除备用 ResourceManager，在看到所要移除的备用 ResourceManger 的主机名之后，请输入 3 并按 Enter 键。
12. 出现提示时，输入 Java API 数据提供程序所需的 JAR 文件的类路径，然后按 Enter 键。
    系统将保存所指定的配置值，并显示确认消息。
13. 运行以下命令可启动代理程序：*install_dir*/bin/hadoop-agent.sh start

下一步做什么

1. 启用子节点事件以查看 Hadoop 代理程序的事件触发阈值。有关启用子节点事件的信息，请参阅第 225 页的『配置仪表板以查看 Hadoop 事件』。
2. 登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。对于某些参数，会在注释中提供缺省值。您可以为这些参数指定不同的值，并移除参数开头的注释标记。

关于此任务

您可以使用静默响应文件在 Linux、AIX 和 Windows 系统上配置 Hadoop 代理程序。

过程

1. 打开位于以下路径的静默响应文件：*install_dir*\samples\hadoop_silent_config.txt
2. 在响应文件中，完成下列步骤：
   a) 要监视为基于 Kerberos SPNEGO 的认证启用的 Hadoop 集群，请指定 yes 并输入以下参数的值：

```
HADOOP_REALM_NAME
HADOOP_KDC_HOSTNAME
HADOOP_PRINCIPAL_NAME
HADOOP_SPNEGO_KEYTAB
```

   b) 要监视启用 SSL 的 Hadoop 集群，请输入 yes，然后输入以下参数的值：

```
HADOOP_TRUSTSTORE_PATH
HADOOP_TRUSTSTORE_PASSWORD
```

c) 针对 Cluster、NameNode (NN)、ResourceManager (RM) 和 Job History Server (JHS) 的下列参数输入值：

```
HADOOP_CLUSTER_NAME (optional)
HADOOP_NN_HOSTNAME
HADOOP_NN_PORT
HADOOP_RM_HOSTNAME
HADOOP_RM_PORT
HADOOP_JHS_HOSTNAME (optional)
HADOOP_JHS_PORT (optional)
```

d) 可选： 对于 **HADOOP_ADDITIONAL_NN_HOSTNAME** 参数，请指定备用或辅助 NameNode 的主机名。

e) 可选： 对于 **HADOOP_ADDITIONAL_NN_PORT** 参数，请指定备用或辅助 NameNode 的端口号。

切记: 如果附加 NameNode 是备用 NameNode，那么与备用 NameNode 守护进程相关联的缺省端口号是 50070。如果附加 NameNode 是辅助 NameNode，那么与辅助 NameNode 守护进程相关联的缺省端口号是 50090。

f) 可选： 对于 **Hadoop_SRM** 参数，请指定 Yes，以便为高可用性集群添加备用 ResourceManager，然后转到步骤 g。

g) 可选： 要监视 Ambari 管理的 Hadoop 集群中的 Hadoop 服务，请输入下列各个参数的值，然后按 Enter 键：

```
AMBARI_SERVER_HOSTNAME
AMBARI_SERVER_PORT
USERNAME_OF_AMBARI_USER
PASSWORD_OF_AMBARI_USER
```

h) 对于 **JAVA_TRACE_LEVEL** 参数，请指定相应的跟踪级别。

i) 可选： 对于 **JAVA_JVM_ARGS** 参数，请为 Java™ 虚拟机指定自变量。

j) 可选： 以如下格式添加备用 ResourceManager 的主机名和端口号：
HADOOP_SRM_PORT.*hadoop_srm_config_sec_1*=8088

其中，*hadoop_srm_config_sec_1* 是运行备用 ResourceManager 的守护进程的节点主机名，而 8088 是缺省端口号。要添加更多备用 ResourceManager，请以同一格式在新行上添加其他备用 ResourceManager 的主机名和端口号。

3. 保存响应文件，然后运行以下命令：

   Linux AIX *install_dir*/bin/hadoop-agent.sh config *install_dir*/samples/hadoop_silent_config.txt

   Windows *install_dir*\bin\hadoop-agent.bat config *install_dir*\samples\hadoop_silent_config.txt

4. 启动代理程序：

   Linux AIX 运行以下命令：*install_dir*\bin\hadoop-agent.sh start

   Windows 右键单击 **Monitoring Agent for Hadoop**，然后单击**开始**。

下一步做什么

1. 启用子节点事件以查看 Hadoop 代理程序的事件触发阈值。有关启用子节点事件的信息，请参阅第 225 页的『配置仪表板以查看 Hadoop 事件』。
2. 登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

配置仪表板以查看 Hadoop 事件

您必须配置仪表板以启用子节点事件，以便**事件**选项卡可以显示 Hadoop 事件。

关于此任务

启用子节点事件的缺省值为 false。将此值更改为 true 可以查看 Hadoop 事件。

过程

1. 打开 Cloud APM 控制台并转至**系统配置**。
2. 在“**高级配置**”页面上，单击**配置类别**下的 **UI 集成**。
3. 从**启用子节点事件**列表中，选择 **True**。
4. 单击**保存**。

向非管理员用户授予许可权

在 Windows 系统上，将调试程序许可权授予非管理员用户，以便运行 Hadoop 代理程序。要在 Hadoop 代理程序仪表板中查看数据，需要这项许可权。

过程

在安装有 Hadoop 代理程序的计算机上，完成下列步骤：

1. 单击**开始 > 控制面板 > 管理工具**。
2. 双击**本地安全策略**。
3. 在“安全设置”窗格中，展开**本地策略**，然后单击**用户权限分配**。
4. 右键单击**调试程序**，然后单击**属性**。
5. 单击**添加用户或组**，并添加要向其授予这项许可权的非管理员用户名。
6. 单击**确定**。

下一步做什么

使用该非管理员用户来配置和运行 Hadoop 代理程序。

配置 HMC Base 监视

Monitoring Agent for HMC Base 为您提供用于监视 Hardware Management Console (HMC) 的功能。代理程序监视以下 HMC 资源的可用性和运行状况：CPU、内存、存储器和网络。代理程序还报告 Power 服务器、CPU 池和 LPAR 的 HMC 库存与配置。通过使用 HMC 性能样本数据，可监视 Power 服务器、LPAR 和池的 CPU 使用率。

开始之前

在配置 HMC Base 代理程序之前，必须完成以下任务：

- 在运行代理程序的系统和 HMC 之间设置 SSH 连接。有关更多信息，请参阅第 228 页的『设置 SSH 连接』。
- 在启动第一个代理程序实例前，准备 HMC SDK。有关更多信息，请参阅第 229 页的『为 HMC 准备 SDK』。

过程

- 要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请完成以下步骤：
  1. 在文本编辑器中打开 `hmc_base_silent_config.txt` 文件：
     - AIX *install_dir*`/samples/hmc_base_silent_config.txt`。
  2. 针对 **`HMC Hostname`**，您可以指定 IP 地址或主机名。
  3. 针对 **`HMC Username`**，必须输入 HMC 的登录用户名，例如，**`HMC_USERNAME=`** *hscroot*。

     注: 分配给 HMC 的登录用户名至少需要 hscviewer 权限。
  4. 对于 **`HMC Password`**，必须输入用户密码。
  5. 针对 **`Maximum Number of Data Provider Log Files:`**，必须指定创建的数据提供程序日志文件的最大数量。例如，**`KPH_LOG_FILE_MAX_COUNT=`***10*。

6. 针对 **Maximum Size in KB of Each Data Provider Log**，必须输入在创建新的日志文件前数据提供程序日志文件可达到的最大大小 (KB)，例如，**KPH_LOG_FILE_MAX_SIZE=** *5190*。
7. 针对 **Level of Detail in Data Provider Log**，必须输入数据提供程序在数据提供程序日志文件中包含的详细信息量，例如，**KPH_LOG_LEVEL=***Fine*。必须指定以下值之一：
   - 1= Off
   - 2=Severe
   - 3=Warning
   - 4=Info
   - 5=Fine
   - 6=Finer
   - 7=Finest
   - 8=All

   要点: 缺省值为 4。
8. 保存并关闭 hmc_base_silent_config.txt 文件，然后输入：`./hmc_base-agent.sh config` *`instance_name install_dir`*`/samples/hmc_base_silent_config.txt`，其中，*instance_name* 是要为实例指定的名称，*install_dir* 是 HMC Base 代理程序 安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。

- AIX 要通过响应提示来配置代理程序，请完成以下步骤：

1. 打开 *`install_dir`*`/bin` 目录，其中，*install_dir* 是 HMC Base 代理程序的安装目录。
2. 要配置 HMC Base 代理程序，请运行以下命令：**./hmc_base-agent.sh config instance_name**。
3. 在出现 **edit the Monitoring Agent for HMC Base settings** 提示时，按 **Enter** 键。缺省值为 Yes。
4. 要输入 HMC 配置信息，请完成以下步骤。
   a. 在出现 **HMC Hostname** 提示时，输入主机名或 IP 地址，然后按 **Enter** 键。
   b. 在出现 **HMC Username** 提示时，输入与 HMC 相关联的登录用户名，然后按 **Enter** 键。
5. 在出现 **HMC Password** 提示时，输入用户密码。
6. 要输入数据提供程序信息，请完成以下步骤：
   a. 在出现 **maximum number of data provider log files** 提示时，输入日志文件量并按 **Enter** 键。

      数据提供程序日志文件的缺省最大数量为 10。
   b. 在出现 **maximum size in KB of each data provider log** 提示时，输入大小并按 **Enter** 键。

      缺省最大大小为 5190 (KB)。
   c. 在出现 **Level of detail in data provider log** 提示时，输入以下某个级别并按 **Enter** 键：
      - 1= Off
      - 2=Severe
      - 3=Warning
      - 4=Info
      - 5=Fine
      - 6=Finer
      - 7=Finest
      - 8=All

下一步做什么

- 要启动代理程序，请输入：`./hmc_base-agent.sh start InstanceName`。
- 根据第 230 页的『配置 HMC Console Server 以监视 Virtual I/O』 中的指示信息配置 HMC 控制台服务器以监视虚拟 I/O。
- 根据第 230 页的『启用对 CPU 和内存利用率的监视』中的指示信息启用 CPU 和内存利用率监视。

设置 SSH 连接

您必须在运行代理程序的系统和 HMC 之间设置 SSH 连接才能使代理程序收集数据。

关于此任务

代理程序数据提供程序通过在 SSH 上运行 CLI 命令从管理控制台收集数据。缺省情况下，数据提供程序最多等待 1 分钟以让 CLI 命令完成运行。在此时间之后，数据提供程序关闭用于运行 CLI 命令的 SSH 会话，并且在该命令成功运行之前，代理程序数据集中不会提供该命令的任何数据。SSH 命令的缺省路径是 `/usr/bin/ssh`。如果您将 SSH 安装在其他位置中，必须使用 **KPH_SSH_PATH** 环境变量来指示路径。

过程

使用以下某个方法设置 SSH 连接：

- 使用 `setup_hmc_key.pl` 脚本设置 SSH 连接。
  a) 登录到代理程序安装所在的服务器。
  b) 打开 `install_dir/aix526/ph/bin` 目录，其中，`install_dir` 是 HMC Base 代理程序的安装目录。
  c) 运行 `perl setup_hmc_key.pl` 命令。
  d) 响应提示并提供 HMC 主机名或 IP 地址；具有等效于 `hscviewer` 权限的权限的 HMC 用户名；以及用于创建密钥对的密码。
  e) 在创建密钥对之后，运行命令（例如 `ssh hscroot@hmchost lshmc -V`）来测试连接。
  如果 SSH 是首次连接到该 HMC，请对以下消息回应 `yes`，以将该 HMC 添加到 `ssh known_hosts` 文件：

```
The authenticity of host 'hmchost (3.3.333.333)' can't be established.
RSA key fingerprint is 4c:b4:26:27:38:f3:ec:58:01:92:26:f9:61:32:bb:4d.
Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'hmchost,3.3.333.333' (RSA) to the list of known hosts.
```

  代理程序现在可以使用 SSH 从 HMC 收集数据。
- 使用 `ssh-keygen` 实用程序生成密钥和设置 SSH 连接。
  a) 登录到代理程序安装所在的服务器。
  b) 使用 `ssh-keygen` 实用程序生成不带解释的公用密钥和专用密钥。
  例如，以下命令生成一组公用密钥和专用密钥：

```
ssh-keygen -t rsa -f /.ssh/id-rsa
```

  提示输入解释时按 Enter 键。生成的公用密钥存储在 `/.ssh/id-rsa.pub` 文件中。专用密钥存储在 `/.ssh/id-rsa` 文件中。
  c) 使用 `scp` 之类的实用程序将包含公用密钥的文件传输到 HMC。
  d) 在 HMC 计算机上，将公用密钥文件追加到 HMC 上存储的密钥集合。
  存储的密钥位于 `/.ssh/authorized_keys2` 文件中。
  e) 在 `known_hosts` 文件中添加 HMC 的主机名和密钥。
  此文件位于 `/.ssh` 目录中。
  a. 运行 `ssh "user"@"hmc_hostname" -i "private_keyfile" date` 命令。

b. 提示对密钥高速缓存时输入 yes。此命令会向 known_hosts 文件添加条目以用于将来连接。

f) 运行 ssh "user"@"hmc_hostname" date 命令。

如果返回日期且没有密码提示，表示成功设置了的 SSH 密钥。

为 HMC 准备 SDK

在首次启动代理程序实例之前，必须先为 HMC 准备 SDK。

关于此任务

在启动第一个代理程序实例之前，必须先为 HMC 准备对应版本的 SDK。在准备完成后，无需为针对同一版本的 HMC 创建的其他 HMC Base 代理程序 实例重复该任务。要监视其他版本的 HMC，重复该任务以再次准备 SDK。

过程

- 从 *agent_dir*/aix526/ph/bin 目录，运行 **prepareSDK.sh** 脚本工具以自动从 HMC 准备 SDK。
  - 如果看到用于 HMC 的 SDK 已经就绪消息，表示准备完成。
  - 如果没有看到用于 HMC 的 SDK 已经就绪消息，可以手动为 HMC 准备 SDK。

    对于 HMC V8.5.0，完成以下步骤：

    1. 使用浏览器打开以下 URL，直接从 HMC 下载 SDK：

       ```
       https://HMC_IP:12443/rest/api/web/sdk
       ```

       看到提示时，输入 hscroot 帐户的用户名和密码。SDK 文件名的格式为 pmc_sdk_*.zip。
    2. . 解压 SDK zip 文件，并转至 IBM HMC REST Web Services SDK Runtime/lib/ibm3 目录。
    3. 如果尚不存在，请创建 <agent_dir>/aix526/ph/lib/*my_hmc_version* 子目录，其中，*my_hmc_version* 是 HMC 环境的版本，例如，8502。要确定 HMC 环境的版本，请运行以下命令：

       ```
       ssh hscroot@<HMC_IP> 'lshmc -v' | grep RM |
       awk -FR '{print $3}' | tr -d '.'
       ```

    4. 将 HMC SDK 的 IBM HMC REST Web Services SDK Runtime/lib/ibm3 文件夹中的所有 .jar 文件复制到 *agent_dir*/aix526/ph/lib/*HMC_version* 目录。

    对于 HMC V8.6.0 或 V8.7.0，完成以下步骤：

    1. 使用浏览器打开以下 URL，直接从 HMC 下载 SDK：

       ```
       https://HMC_IP:12443/rest/api/web/sdk
       ```

       看到提示时，输入 hscroot 帐户的用户名和密码。SDK 文件名的格式为 pmc-rest-sdk*.zip。
    2. 解压 SDK zip 文件，并转至 lib 子目录。
    3. 如果尚不存在，请创建 <agent_dir>/aix526/ph/lib/*my_hmc_version* 子目录，其中，*my_hmc_version* 是 HMC 环境的版本，例如，8602 或 87012。要确定 HMC 环境的版本，请运行以下命令：

       ```
       ssh hscroot@<HMC_IP> 'lshmc -v' | grep RM |
       awk -FR '{print $3}' | tr -d '.'
       ```

    4. 将 HMC SDK 的 lib 文件夹中的所有 .jar 文件复制到 *agent_dir*/aix526/ph/lib/*my_hmc_version* 目录。

结果
您已成功为 HMC 准备好了 SDK。

下一步做什么
根据第 226 页的『配置 HMC Base 监视』中的指示信息配置 HMC Base 代理程序。

配置 HMC Console Server 以监视 Virtual I/O

在 HMC Base 代理程序能够监视 Virtual I/O 的状态之前，必须配置 HMC Console Server。

过程

执行以下步骤来将 HMC Console Server 配置为 HMC Base 代理程序监视 Virtual I/O 的先决条件。

- 启动 HMC Console Server 的 PMC 功能和 Virtual I/O 服务器。
  a) 以传统方式使用浏览器来登录 HMC Console Server。

  ```
  https://hmc_hostname
  ```

  b) 单击 **HMC Management** > **更改性能监视设置**。
  将显示“**更改性能监视设置**”窗口。
  c) 在**管理服务器的性能监视数据收集**部分，打开相应服务器的**收集**功能。
  d) 单击每个 Virtual I/O Server 以显示该服务器的“**分区属性**”窗口。
  e) 在“**常规**”选项卡中，确保已选中**允许性能信息收集**选项的复选框。
  单击**确定**以保存设置。
  几分钟后，您就可以在“**性能监视**”页面上看到相应服务器的网络和存储流量。
- 确保 HMC Base 代理程序的 HMC 用户拥有正确的权限。
  a) 添加或编辑用户时，请确保此用户拥有 **hmcviewer** 角色，并且已为此用户启用 **AllSystemResource** 选项。
  b) 在“**用户属性**”窗口中，启用**允许通过 Web 进行远程访问**选项。

启用对 CPU 和内存利用率的监视

如果已禁用 CPU 和内存利用率的数据收集，那么每个电源服务器的 CPU 和内存利用率数据将不会显示在 UI 中。

过程

使用以下某种方法来启用对 CPU 和内存利用率的监视。

- 运行以下 HMC 管理命令 `chlparutil`，以启用对 CPU 和内存利用率的监视。

  ```
  chlparutil-r config -m <CECname> -s <the sample rate in seconds, always 60>
  ```

- 在 HMC Console Server 上启用对 CPU 和内存利用率的监视。
  a) 以传统方式登录 HMC Console Server。
  b) 单击导航树上的**服务器**节点。
  c) 选择服务器，然后转至**操作** > **利用率数据** > **更改采样率**。
  d) 设置采样率。
  缺省情况下，采样率是禁用的。您可以为采样率设置合适的值，例如，`30` 分钟。

配置 HTTP Server 监视

在安装后，Monitoring Agent for HTTP Server 会自动启动。要启用数据收集，请确保 HTTP 服务器在运行中，并编辑 HTTP 服务器配置文件，使其包含对 HTTP Server 代理程序数据收集器配置文件的引用。

开始之前

HTTP Server 代理程序的配置涉及到两个文件。要查看这些文件的样本，请参阅样本。请找到并查看下列文件：

HTTP Server 代理程序数据收集器配置文件

安装 HTTP Server 代理程序之后，它会发现 HTTP 服务器，并在 *install_dir*/tmp/khu 目录中生成数据收集器配置文件，其中 *install_dir* 是 HTTP Server 代理程序的安装目录。

如果环境中有多个 HTTP 服务器，那么将针对每个 HTTP 服务器生成一个 HTTP Server 代理程序配置文件。

HTTP Server 代理程序配置文件名由两部分组成，其格式如下：

```
khu.full path of the HTTP Server configuration file name.conf
```

代理程序配置文件名的第一部分是 khu，其中 hu 是 HTTP Server 代理程序代码。
代理程序配置文件名的第二部分是使用 HTTP 服务器配置文件的完整路径和名称来创建，其中 / 将替换为 .。例如，可能的文件名如下所示：

Linux AIX
```
khu.usr.local.apache24.conf.httpd.conf
```

Windows
```
khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf
```

HTTP Server 代理程序数据收集器配置文件包含下列元素：

- 有关 HTTP 服务器使用的 httpd.conf 文件的路径的详细信息，例如，`KhuShmemPath "/IBM/HTTPServer/conf/httpd.conf"`。
- 要装入的库的位置
- 与共享内存相关联的许可权

HTTP 服务器配置文件

每个 HTTP 服务器都有一个配置文件，缺省情况下，其名称为 *http_server_install_dir*/conf/httpd.conf，其中 *http_server_install_dir* 是 HTTP 服务器的安装目录。在某些环境中，此文件名可定制。请向 HTTP 服务器管理员核实确切的文件名。

关于此任务

在下列情况下，请启用 HTTP Server 代理程序数据收集：

- 安装 HTTP Server 代理程序之后
- 升级到 HTTP Server 代理程序 V1.0.0.4 之后，新别名导致现有的 HTTP Server 代理程序节点在 Cloud APM 控制台中处于脱机状态。
- 在从 V1.0.0.4 进行升级之后，现有 HTTP Server 代理程序节点在 Cloud APM 控制台中变为脱机。发生这种情况的原因可能是，您有代理程序配置文件名相似（例如，httpd 与 httpd01）的多个 HTTP 服务器实例。

HTTP Server 代理程序 需要 netstat 命令行网络实用程序工具以成功发现正在运行的 HTTP Server。

要点: 要解决升级之后发生的代理程序节点脱机问题，完成此任务之后，必须在 Cloud APM 控制台中添加新的 HTTP 服务器实例。

过程

1. 要激活数据收集，必须在 HTTP 服务器配置文件中，使用 Include 语句来引用数据收集器配置文件。请在 HTTP 服务器配置文件末尾追加以下语句：

   ```
   Include “install_dir/tmp/khu/khu.full path of the HTTP Server configuration file name.conf”
   ```

   例如，

   Linux AIX 如果您的 IBM HTTP Server 是安装在 /opt/IBM/HTTPServer 目录中，而且数据收集器配置文件位于以下目录：

   ```
   /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
   ```

   请在 /opt/IBM/HTTPServer/conf/httpd.conf HTTP 服务器配置文件末尾追加以下语句：

   ```
   Include “/opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf”
   ```

   Windows 如果存在安装于 C:\ProgramFiles\IBM\HTTPServer 目录中的 IBM® HTTP Server，并且数据收集器配置文件位于以下目录：

   ```
   C:\IBM\APM\tmp\khu\khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf
   ```

   请在 HTTP 服务器配置文件 C:\Program Files\IBM\HTTPServer\conf\httpd.conf 中追加以下语句：

   ```
   Include “C:\IBM\APM\tmp\khu\khu.C.Program Files.IBM.HTTPServer.conf.httpd.conf”
   ```

2. 切换到以下目录：

   ```
   HTTP_server_installation_directory/bin
   ```

3. 重新启动 HTTP 服务器。例如：

   Linux AIX

   ```
   ./apachectl -k stop
   ./apachectl -k start
   ```

   Windows

   ```
   httpd.exe -k stop
   httpd.exe -k start
   ```

结果

您已成功配置代理程序。

下一步做什么

现在，您可验证 HTTP Server 代理程序 数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

注: 如果 HTTP 服务器中没有流量，在 Cloud APM 控制台中就不会显示任何数据。

IBM HTTP Server 响应时间模块

当您安装 Response Time Monitoring 代理程序以便与 IBM HTTP Server 响应时间模块配合工作时，它将在所有端口上监视 HTTP 和 HTTPS 请求。

IBM HTTP Server 响应时间模块

IBM HTTP Server 响应时间模块是 HTTP Server 代理程序的组成部分。在 AIX、Linux 或 Windows 上的 Apache HTTP Server 或 IBM HTTP Server 上，如果在 Response Time Monitoring 代理程序之前安装并配置 HTTP Server 代理程序，或同时安装并配置它们，那么将自动启用 IBM HTTP Server 响应时间模

块。有关 IBM HTTP Server 响应时间模块功能的描述，请参阅第 591 页的『配置 IBM HTTP Server 响应时间模块』。

数据收集器配置文件

安装 HTTP Server 代理程序之后，它会发现 HTTP 服务器，并在 *install_dir*/tmp/khu 目录中生成数据收集器配置文件，其中 *install_dir* 是 HTTP Server 代理程序的安装目录。

对于 Apache HTTP Server，数据收集器配置文件是：
`khu.usr.local.apache24.conf.httpd.conf`
对于 IBM HTTP Server，数据收集器配置文件是：
`khu.opt.IBM.HTTPServer.conf.httpd.conf`

插件

HTTP Server 代理程序由两个插件组成：

1. khu_module - 这是 HTTP Server 代理程序。此插件负责所有与 HTTP Server 代理程序相关联的仪表板。有关更多信息，请参阅 HTTP Server 代理程序参考。
2. wrt_module - 这是 IBM HTTP Server 响应时间模块

这两个插件在数据收集器配置文件中指示，如下所示：

```
LoadModule khu_module

LoadModule wrt_module
```

激活数据收集

要激活数据收集，必须在 HTTP 服务器配置文件中，使用 Include 语句来引用数据收集器配置文件。请在 HTTP 服务器配置文件末尾追加以下语句：

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
```

有关更多信息，请参阅第 231 页的『配置 HTTP Server 监视』。

激活数据收集后，将填充最终用户仪表板。

HTTP Server 代理程序 代码样本

HTTP Server 代理程序的配置涉及到两个文件。它们是 HTTP Server 代理程序 数据收集器配置文件和 HTTP 服务器配置文件。另外，还会提供实例别名映射文件的样本来帮助说明别名如何工作。

HTTP Server 代理程序 数据收集器文件样本

对于 IBM HTTP Server V8 和更高版本（64 位），HTTP Server 代理程序 数据收集器配置文件包含以下信息：

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache22dc_64.so"

<IfModule mod_khu.c>
   KhuShmemPerm 660
   KhuShmemPath "/opt/IBM/IHS/conf/httpd.conf"
   KhuCpsPath "/tmp/ihs/tmp/khu/khu_cps.properties"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  #Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap22_64.so
WrtOriginID HU:tivvm09_httpd:HUS
```

对于 IBM HTTP Server V7（32 位），此配置文件包含以下信息：

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache22dc_32.so"

<IfModule mod_khu.c>
 KhuShmemPerm 660
 KhuShmemPath "/opt/IBM/HTTPServer/conf/httpd.conf"
 KhuCpsPath "/tmp/ihs/tmp/khu/khu_cps.properties"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  #Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap22.so
WrtOriginID HU:linux_httpd:HUS
```

对于 Apache V2.4（64 位），HTTP Server 代理程序 配置文件包含以下信息：

```
#
# Settings for Monitoring Agent for HTTP Server module.
#

LoadModule khu_module "/tmp/ihs/lx8266/hu/lib/khuapache24dc_64.so"

<IfModule mod_khu.c>
 KhuShmemPerm 660
 KhuShmemPath "/usr/local/apache24/conf/httpd.conf"
</IfModule>

Alias /khu "/tmp/ihs/lx8266/hu/etc"
<Directory "/tmp/ihs/lx8266/hu/etc">
  Order deny,allow
  Allow from all
  Require all granted
</Directory>

LoadModule wrt_module /tmp/ihs/lx8266/hu/lib/mod_wrt_ap24_64.so
WrtOriginID HU:linux-tzsi_httpd:HUS
```

实例别名映射文件样本

```
# Monitoring Agent for HTTP Server instance alias mapping
# INSTANCE: auto discovered by agent. Please do NOT modify.
# ALIAS: alias name for the instance. The name will be displayed in APM UI dashboard. It
must be unique
# among all instances and it must be less than 10 characters and consist of only
alphanumeric characters.
#
INSTANCE.1=/usr/local/apache24/conf/httpd.conf
ALIAS.1=httpd

INSTANCE.1=/usr/local/apache24/conf/admin.conf
ALIAS.1=admin
```

配置 IBM Cloud 监视

Monitoring Agent for IBM Cloud 可从 IBM Cloud (SoftLayer) 帐户收集虚拟机库存和度量值。IBM Cloud 代理程序用于跟踪 IBM Cloud 中已配置并且正在运行的虚拟设备数量。您可以在详细仪表板页面中查看分配给每个虚拟设备的资源，该页面还显示诸如设备所在的数据中心、操作系统以及相应月份的预计公共网络带宽之类的信息。

开始之前

· 请通读第 234 页的『配置 IBM Cloud 监视』主题，确定完成配置所需的内容。

· 此处的方向针对的是代理程序的最新发行版，除非另有说明。

· 请确保在您的环境中符合 IBM Cloud 代理程序的系统需求。有关最新系统需求信息，请参阅 IBM Cloud 代理程序的软件产品兼容性报告 (SPCR)。

· 确保您掌握下列信息：

– 至少具有“审计员”许可权的用户的用户名。

– 此关联用户的 IBM Cloud 的 API 密钥。

关于此任务

IBM Cloud 代理程序既是多实例代理程序，也是子节点代理程序。配置代理程序实例后，必须手动启动每个代理程序实例。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   · 第 235 页的『在 Windows 系统上配置代理程序』.
   · 第 236 页的『使用静默响应文件来配置代理程序』.
2. 使用提示响应的脚本或静默响应文件在 Linux 系统上配置代理程序。
   · 第 236 页的『通过响应提示来配置代理程序』.
   · 第 236 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

· Linux `/opt/ibm/apm/agent/logs`

· Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 IBM Cloud 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for IBM Cloud** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项不可用。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置...**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用拉丁字母、阿拉伯数字和连字符/减号字符。例如，`icloud-inst`。
4. 在代理程序实例名称窗口上单击**下一步**。
5. 按**新建**并输入 IBM Cloud SoftLayer 用户名和 API 密钥设置，然后单击**下一步**。
6. 单击**确定**以完成配置。

7. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过响应提示来配置代理程序

安装 IBM Cloud 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 IBM Cloud 代理程序安装在本地 Linux 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 IBM Cloud 代理程序。

1. 运行以下命令:

   ```
   install_dir/bin/ibm_cloud-agent.sh config instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/ibm_cloud-agent.sh config icloud-inst
   ```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 237 页的『IBM Cloud 代理程序的配置参数』，以获取每个配置参数的说明。

3. 运行以下命令以启动代理程序:

   ```
   install_dir/bin/ibm_cloud-agent.sh start instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/ibm_cloud-agent.sh start icloud-inst
   ```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

- 以静默方式配置 IBM Cloud 代理程序:

  a) 在文本编辑器中，打开位于以下某个路径中的 ibm_cloud_silent_config.txt 文件。

  - Linux *install_dir*/samples/ibm_cloud_silent_config.txt

    例如， /opt/ibm/apm/agent/samples/ibm_cloud_silent_config.txt

  - Windows *install_dir*\samples\ibm_cloud_silent_config.txt

    例如，C:\IBM\APM\samples\ibm_cloud_silent_config.txt

其中，*install_dir* 是安装代理程序的路径。

b) 在 ibm_cloud_silent_config.txt 文件中，指定所有必需参数的值，并根据需要修改其他参数的缺省值。

请参阅：。第 237 页的『IBM Cloud 代理程序的配置参数』以获取各个配置参数的说明。

c) 保存并关闭 ibm_cloud_silent_config.txt 文件，然后运行以下命令：

- Linux *install_dir*/bin/ibm_cloud-agent.sh config **instance_name** *install_dir*/samples/ibm_cloud_silent_config.txt

  例如，**/opt/ibm/apm/agent/bin/ibm_cloud-agent.sh config icloud-inst /opt/ibm/apm/agent/samples/ibm_cloud_silent_config.txt**

- Windows *install_dir*\bin\ibm_cloud-agent.bat config **instance_name** *install_dir*\samples\ibm_cloud_silent_config.txt

  例如，**C:\IBM\APM\bin\ibm_cloud-agent.bat config icloud-inst C:\IBM\APM\samples\ibm_cloud_silent_config.txt**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

d) 运行以下命令以启动代理程序：

- Linux *install_dir*/bin/ibm_cloud-agent.sh start **instance_name**

  例如，**/opt/ibm/apm/agent/bin/ibm_cloud-agent.sh start icloud-inst**

- Windows *install_dir*\bin\ibm_cloud-agent.bat start **instance_name**

  例如，**C:\IBM\APM\bin\ibm_cloud-agent.bat start icloud-inst**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

IBM Cloud 代理程序的配置参数

下表显示 IBM Cloud 代理程序的配置参数。

1. IBM Cloud 配置 - 用于远程监视 IBM Cloud 实例的设置。针对要配置的 API 密钥自动发现实例。

表 *23. IBM Cloud* 配置

参数名称	描述	静默配置文件参数名称
用户名	用于从 IBM Cloud API 检索度量值的 IBM SoftLayer 帐户的用户名。	**KFS_USERNAME**
API 密钥	完成认证所需的特定于用户的 API 密钥。将生成 API 密钥并且可从 IBM SoftLayer 客户门户网站检索此密钥。	**KFS_API_KEY_PASSWORD**

配置 IBM Integration Bus 监视

IBM Integration Bus 代理程序是多实例代理程序。您必须创建第一个代理程序实例并手动启动。

开始之前

- 此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。
- 请确保在您的环境中符合 IBM Integration Bus 代理程序的系统需求。有关最新的系统需求信息，请参阅 IBM Integration Bus 代理程序 的详细系统需求报告。

关于此任务

以下过程是配置 IBM Integration Bus 代理程序的路线图，其中包括必需步骤和可选步骤。请根据您的需要，完成必要的步骤。

过程

1. 确保用于启动和停止 IBM Integration Bus 代理程序的用户标识属于 **mqm** 和 **mqbrkrs** 用户组。
2. Windows
   如果 IBM MQ (WebSphere MQ) 是安装在 Windows 系统上，请将 IBM MQ (WebSphere MQ) 库路径添加到 **PATH** 环境变量。这样，IBM Integration Bus 代理程序就可以装入所需的 IBM MQ (WebSphere MQ) 库以启动。
   a) 将 IBM MQ (WebSphere MQ) 库路径添加到 **PATH** 环境变量开头。

      例如，如果 IBM MQ (WebSphere MQ) 的安装路径为 `C:\IBM\WMQ75`，请在 Windows 系统的 **PATH** 环境变量开头添加 `C:\IBM\WMQ75\bin`。

   b) 重新启动 Windows 系统，以使更改生效。
3. 通过指定下列配置参数，配置 IBM Integration Bus 代理程序。您还可以针对代理程序指定一些可选配置参数。有关详细的指示信息，请参阅第 238 页的『配置 IBM Integration Bus 代理程序』。
   - 代理程序标识
   - 要监视的集成节点（代理）的安装目录
   - IBM MQ (WebSphere MQ) 的 64 位库路径
4. 配置 IBM Integration Bus，以启用所要监视的数据。请参阅第 242 页的『配置 IBM Integration Bus 以启用数据』。
5. 如果已对集成节点（代理）启用快照数据收集，请将 IBM Integration Bus 代理程序配置为不存储任何快照数据。有关指示信息，请参阅第 248 页的『禁用代理程序的快照数据收集』。
6. 可选： 要配置 IBM Integration Bus 代理程序以启用事务跟踪，请使用“**代理程序配置**”页面。有关指示信息，请参阅第 248 页的『配置 IBM Integration Bus 代理程序的事务跟踪』。
7. 可选： 如果您不再需要事务跟踪功能，或者要卸载 IBM Integration Bus 代理程序，请对 IBM Integration Bus 禁用事务跟踪，并移除代理程序所提供的用户出口。有关指示信息，请参阅第 247 页的『禁用事务跟踪』和第 250 页的『移除 KQIUserExit 用户出口』。

配置 IBM Integration Bus 代理程序

您必须为 IBM Integration Bus 代理程序分配一个实例名称并配置该代理程序，然后才能开始监视 IBM Integration Bus 环境。

开始之前

- 请确保用于启动和停止此代理程序的用户标识属于 **mqm** 和 **mqbrkrs** 用户组。
- Windows 如果 IBM MQ (WebSphere MQ) 是安装在 Windows 系统上，请将 IBM MQ (WebSphere MQ) 库路径添加到 **PATH** 环境变量。这样，IBM Integration Bus 代理程序就可以装入所需的 IBM MQ (WebSphere MQ) 库以启动。
  1. 将 IBM MQ (WebSphere MQ) 库路径添加到 **PATH** 环境变量开头。

     例如，如果 IBM MQ (WebSphere MQ) 的安装路径为 `C:\IBM\WMQ75`，请在 Windows 系统的 **PATH** 环境变量开头添加 `C:\IBM\WMQ75\bin`。

  2. 重新启动 Windows 系统，以使更改生效。
- 在代理程序配置期间，您可能需要根据您的环境提供下列信息。如果您不知道所需指定的适当配置值，请向 IBM MQ (WebSphere MQ) 及 IBM Integration Bus 的管理员收集信息。
  - 如果 IBM MQ (WebSphere MQ) 与 IBM Integration Bus 代理程序安装在同一个系统上，那么您必须提供 IBM MQ (WebSphere MQ) 的 64 位库路径。

– 如果 IBM Integration Bus 代理程序将配置为监视 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的集成节点，那么必须提供 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的安装目录。
– 如果您希望 IBM Integration Bus 代理程序 监视一些特定的集成节点（代理），而不是监视同一系统上的所有集成节点（代理），那么必须提供每个集成节点（代理）的名称及安装路径。

关于此任务

IBM Integration Bus 代理程序是一个多实例代理程序；您必须手动创建第一个实例并启动代理程序。

在 UNIX 或 Linux 系统上，您可以选择在进行或不进行交互的情况下配置此代理程序。在 Windows 系统上，只能在不进行交互的情况下配置此代理程序。

· 要通过交互配置代理程序，请运行配置脚本并对提示做出响应。请参阅第 239 页的『交互式配置』。
· 要不使用交互配置代理程序，请编辑静默响应文件，然后运行配置脚本。请参阅第 240 页的『静默配置』。

要点: 如果在安装 IBM Integration Bus 代理程序（在 Cloud APM 中交付）的系统上也安装有 ITCAM Agent for WebSphere Message Broker（作为其中一个 ITCAM for Applications 产品来交付），请勿使用它们来监视该系统上的同一个集成节点（代理）。

交互式配置

过程

要通过运行脚本并响应提示来配置代理程序，请完成下列步骤：

1. 输入以下命令：

   ```
   install_dir/bin/iib-agent.sh config instance_name
   ```

   其中，*instance_name* 是想要为代理程序实例指定的名称。

   要点: 在 Windows 系统上不支持交互式配置。
2. 确认要配置 IBM Integration Bus 代理程序之后，请指定一般代理程序设置的配置值。
   a) 提示输入 **Agent Id** 参数时，请指定最大长度为 8 个字符的唯一字母数字字符串。

      切记: 自 IBM Integration Bus 代理程序 V7.3.0.1 开始，最大代理程序标识长度更改为 8 个字符。对于先前版本，最大代理程序标识长度为 4 个字符。

      受管系统名称包含您指定的代理程序标识，例如，*monitoredbrokername*:*agentID*:KQIB，其中 *monitoredbrokername* 是受监视集成节点（代理）的名称。

   b) 在系统提示输入 **IIB version 10 or ACE version 11** 安装目录参数时，如果想要监视 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的集成节点，请指定 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的安装目录。例如，`/opt/ibm/mqsi/ace-11.0.0.3`。如果不想要监视 IBM Integration Bus V10 和 IBM App Connect Enterprise V11，请按 Enter 键以接受缺省值。

      切记: 对于 **IIB version 10 or ACE version 11** 安装目录参数，只能指定一个安装目录。如果 IBMIntegration Bus V10 或 IBM App Connect Enterprise V11 安装在不同的目录中，并且想要监视所有这些目录，请创建多个代理程序实例，并为每个代理程序实例指定一个 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的安装目录。
3. 可选： 使用 **Monitored Broker Settings** 区段，指定使用此代理程序是否仅监视某些特定集成节点（代理）。

   缺省情况下，将监视 IBM Integration Bus 代理程序所在主机系统上运行的所有集成节点（代理），这些集成节点（代理）是通过自发现过程来确定。如果要让此代理程序监视某些特定的集成节点（代理），请指定要监视的集成节点（代理）的名称，并在 **Monitored Broker Settings** 区段中，将 **Collect Node Data** 设置为缺省值 No。存在多个 **Monitored Broker Settings** 区段。每个区段控制一个集成节点（代理）的监视设置。

提示: 您可以指定多个 **Monitored Broker Settings** 区段。在编辑 **Monitored Broker Settings** 区段中，以下选项可用：

- Add：创建要针对另一个集成节点（代理）来配置的 **Monitored Broker Settings** 区段。
- Edit：修改当前 **Monitored Broker Settings** 区段的设置。
- Del：删除当前 **Monitored Broker Settings** 区段。
- Next：移至下一个 **Monitored Broker Settings** 区段。
- Exit：退出 **Monitored Broker Settings** 配置。

4. 如果您确认 IBM MQ (WebSphere MQ) 是安装在同一系统上，那么系统将提示输入 **WebSphere MQ 64-bit library path** 参数。请按 Enter 键以接受缺省值，即代理程序自动发现的 IBM MQ (WebSphere MQ) 64 位库路径。如果没有显示缺省值，那么您必须提供 IBM MQ (WebSphere MQ) 的 64 位库路径，然后再前进到下一步。例如，`/opt/mqm8/lib64`。

   切记: 如果各集成节点（代理）使用的是不同版本的队列管理器，请对此参数指定最新版本的 IBM MQ (WebSphere MQ) 64 位库路径。
5. 配置完成之后，请输入下列命令以启动此代理程序：

```
install_dir/bin/iib-agent.sh start
instance_name
```

静默配置

过程

要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请完成下列步骤：

1. 在文本编辑器中打开下列代理程序静默响应文件。
   - Linux AIX *install_dir*/samples/iib_silent_config.txt
   - Windows *install_dir*\tmaitm6_x64\samples\qi_silent_config.txt

   其中，*install_dir* 是 IBM Integration Bus 代理程序的安装目录。缺省安装目录如下所示：
   - Linux AIX /opt/ibm/apm/agent
   - Windows C:\IBM\APM
2. 对于 **agentId** 参数，请指定最大长度为 8 个字符的唯一字母数字字符串，用作此代理程序的简短标识。

   切记: 自 IBM Integration Bus 代理程序 V7.3.0.1 开始，最大代理程序标识长度更改为 8 个字符。对于先前版本，最大代理程序标识长度为 4 个字符。

   受管系统名称包含您指定的代理程序标识，例如，*monitoredbrokername*:*agentID*:KQIB，其中 *monitoredbrokername* 是受监视集成节点（代理）的名称。
3. 如果想要监视 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的集成节点，那么为 **defaultWMBInstallDirectory** 参数指定 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的安装目录。例如，`C:\Program Files\IBM\ACE\11.0.0.3\`（用于 Windows 系统）或 `/opt/ibm/mqsi/ace-11.0.0.3`（用于 Linux 系统）。如果不想监视 IBM Integration Bus V10 和 IBM App Connect Enterprise V11，那么不需要此参数，因为 IBM Integration Bus 代理程序 可自动从先前版本发现集成节点（代理）。

   切记: 对于 **defaultWMBInstallDirectory** 参数，只能指定一个安装目录。如果 IBMIntegration Bus V10 或 IBM App Connect Enterprise V11 安装在不同的目录中，并且想要监视所有这些目录，请创建多个代理程序实例，并为每个代理程序实例指定一个 IBM Integration Bus V10 或 IBM App Connect Enterprise V11 的安装目录。
4. 可选： 指定使用此代理程序是否仅监视某些特定集成节点（代理）。

   缺省情况下，将监视 IBM Integration Bus 代理程序所在主机系统上运行的所有集成节点（代理），这些集成节点（代理）是通过自发现过程来确定。要监视特定的集成节点（代理），请为所要监视的每个集成节点（代理）设置 **collectNodeData** 和 **WMBInstallDirectory** 参数。

collectNodeData
指定是否收集受监视集成节点（代理）的节点定义数据。语法为 `collectNodeData.brkr_name=NO|YES`，其中 *brkr_name* 是集成节点（代理）的名称。

缺省值为 NO。建议使用缺省值，因为在 Cloud APM 控制台中不支持节点定义数据。

WMBInstallDirectory
要监视的集成节点（代理）的安装目录。语法为 `WMBInstallDirectory.brkr_name=broker_install_dir`，其中 *broker_install_dir* 是要监视的集成节点（代理）的安装目录。

切记: 对于 V10 集成节点，**WMBInstallDirectory** 参数可能会覆盖上一个步骤中设置的 **defaultWMBInstallDirectory** 参数。

例如，仅监视两个分别名为 BK1 和 BK2 的集成节点（代理）时，请将参数设置如下：

```
collectNodeData.BK1=NO
collectNodeData.BK2=NO
WMBInstallDirectory.BK1=BK1_install_dir
WMBInstallDirectory.BK2=BK2_install_dir
```

5. 要监视版本低于 IBM Integration Bus V10 的代理，请对 **WMQLIBPATH** 参数指定 IBM MQ (WebSphere MQ) 的 64 位库路径。例如，`C:\Program Files\IBM\WebSphere MQ\bin64`（Windows 系统）或 `/opt/mqm8/lib64`（Linux 系统）。

   切记: 如果各集成节点（代理）使用的是不同版本的队列管理器，请对此参数指定最新版本的 IBM MQ (WebSphere MQ) 64 位库路径。
6. 保存并关闭代理程序静默响应文件，然后输入以下命令：
   - Linux AIX *install_dir*/bin/iib-agent.sh config *instance_name path_to_responsefile*
   - Windows *install_dir*\BIN\iib-agent.bat config "*instance_name path_to_responsefile*"

   其中，*instance_name* 是配置的实例的名称，*path_to_responsefile* 是静默响应文件的完整路径。

   警告: 在 Windows 系统上，请勿包含用于将默响应文件的完整路径括起的双引号 (""), 因为这将导致配置错误。
7. 配置完成之后，请输入下列命令以启动此代理程序：
   - Linux AIX

     ```
     install_dir/bin/iib-agent.sh start
     instance_name
     ```
   - Windows

     ```
     install_dir\bin\iib-agent.bat start instance_name
     ```

结果

现在，您可登录 Cloud APM 控制台，并使用“应用程序”编辑器将 IBM Integration Bus 代理程序实例添加到“应用程序性能仪表板”。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关使用“应用程序”编辑器的信息，请参阅第 930 页的『管理应用程序』。

切记: 每当更新或迁移受监视集成节点（代理）时，在集成节点（代理）升级或迁移后，必须重新启动 IBM Integration Bus 代理程序。

下一步做什么

下一步是配置 IBM Integration Bus 以启用数据。只有在 IBM Integration Bus 中启用下列数据之后，这些数据才会显示在“应用程序性能仪表板”上：

- 归档记帐和统计信息

· JVM 资源统计信息

· 事务跟踪

有关指示信息，请参阅第 242 页的『配置 IBM Integration Bus 以启用数据』。

配置 IBM Integration Bus 以启用数据

要让某些数据显示在 Cloud APM 控制台中，您必须配置 IBM Integration Bus，以启用所需的数据收集。

开始之前

确保配置 IBM Integration Bus 代理程序。

切记: 事务跟踪启用要求重新启动集成节点（代理）。

关于此任务

仅当对集成节点（代理）启用数据收集之后，IBM Integration Bus 代理程序才能监视归档统计信息及资源统计信息。同样，如果要在中间件及拓扑仪表板中查看事务跟踪，必须先在集成节点（代理）中启用事务跟踪，再对 IBM Integration Bus 代理程序启用事务跟踪。

确定要使用 IBM Integration Bus 代理程序来监视的数据类型，然后根据需要完成下列步骤。

集成节点拥有的集成服务器对于存储在集成节点目录的子目录中的每台集成服务器具有一个缺省 `server.conf.yaml` 配置文件。集成节点拥有的集成服务器将继承针对 `node.conf.yaml` 文件中的集成节点设置的任何属性。但是，您可以通过在其 `server.conf.yaml` 文件中进行修改，来更改集成服务器的属性。（有关更多信息，请参阅 IBM App Connect Enterprise 文档中的通过修改 node.conf.yaml 文件配置集成节点。）

过程

· 要对集成节点（代理）启用归档统计信息数据收集，请参阅第 242 页的『启用归档记帐和统计信息数据收集』。

· 要对集成节点（代理）启用资源统计信息数据，请参阅第 245 页的『启用 JVM 资源统计信息』。

· 要对集成节点（代理）内的消息流启用事务跟踪，请参阅第 246 页的『启用事务跟踪』。

· 不再需要事务跟踪数据时，切记对已启用事务跟踪的集成节点（代理）禁用事务跟踪。请参阅第 247 页的『禁用事务跟踪』。

启用归档记帐和统计信息数据收集

关于此任务

要对属于集成节点（代理）的消息流启用归档记帐及统计信息收集，请从集成节点（代理）安装目录的 `bin` 目录发出 **mqsichangeflowstats** 命令。

切记: 根据您的监视数据需求，向集成节点（代理）发出 **mqsichangeflowstats** 命令。建议仅启用所需统计信息，因为有许多消息流时可能会产生大量数据和处理。有关 **mqsichangeflowstats** 命令的更详细信息，请参阅 IBM Integration Bus 文档。

要点: 由于设置的 20 秒快照时间间隔所需的数据量和处理，IBM Cloud Application Performance Management 不支持快照记帐和统计信息数据。归档数据提供与快照数据相同的准确属性，更适合用于 IBM Cloud Application Performance Management 提供的一般产品监视。如果已对集成节点（代理）启用快照数据收集，请记得将 IBM Integration Bus 代理程序配置为不存储快照数据。有关指示信息，请参阅第 248 页的『禁用代理程序的快照数据收集』。

过程

· 要最大限度获取消息流的数据，请发出下列命令。建议您使用此命令，因为它不会启用最详细的终端统计信息（这些信息会提供每个节点的每个终端的调用计数）。终端级别会耗用大量存储器。

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n basic -o xml
```

- 在 ACE V11 中，要获取消息流的大部分数据，请按如下所示修改 node.conf.yaml 或 server.conf.yaml 文件。这些属性是推荐值，因为它们不会启用最详细的终端统计信息（这些信息会提供每个节点的每个终端的调用计数）。终端级别会耗用大量存储器。

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               #csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'     # choose 1 of : active|inactive,
                             # default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60         # Sets the interval in minutes at which
                              #archive statistics are published
    nodeDataLevel: 'basic'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml'  # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
```

注: 如果想要禁用此设置，请注释掉行 **archivalOn: 'active'**、**nodeDataLevel: 'basic'** 和 **outputFormat: 'xml'**。

- 要获取 IBM Integration Bus 代理程序所支持的所有数据，请发出以下命令：

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n advanced -o xml
```

- 在 ACE V11 中，要获取 IBM Integration Bus 代理程序 支持的所有数据，请按如下所示修改 node.conf.yaml 或 server.conf.yaml 文件：

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60       # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none' # choose 1 of : none|basic
```

注: 如果想要禁用此设置，请注释掉行 **archivalOn: 'active'**、**nodeDataLevel: 'advanced'** 和 **outputFormat: 'xml'**。

- 要减少数据量但仍合理监视所有消息流而不需要更多详细信息，请发出以下命令：

```
mqsichangeflowstats BrokerName -a -g -j -c active -t none -n none -o xml
```

- 在 ACE V11 中，要减少数量，但仍合理监视所有消息流而不包含进一步详细信息，请按如下所示修改 node.conf.yaml 或 server.conf.yaml 文件：

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  #set here
```

```
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #majorInterval: 60       # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'none'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none' # choose 1 of : none|basic
```

注: 如果想要禁用此设置，请注释掉行 **archivalOn: 'active'**、**nodeDataLevel: 'none'** 和 **outputFormat: 'xml'**。

- 如果有大量消息流，并要减少数据量，您可替换上述命令中的 `-g` 或 `-j` 选项，以指定要监视的消息流。
  - 要指定启用的特定集成服务器（执行组），将 `-g` 替换为 `-e IntegrationServerName`。
  - 要标识启用的特定消息流，将 `-j` 替换为 `-f MessageFlowName`。
  - 如果已将消息流分组到应用程序中，且要指定所要启用的特定应用程序，请将 `-k ApplicationName` 添加到 `-j` 选项。
- IBM Integration Bus 代理程序将以 5 分钟的时间间隔来收集归档记帐和统计信息数据。要将集成节点（代理）生成归档记帐及统计信息数据的时间间隔设置为相同，请在集成节点（代理）停止后发出下列命令，然后重新启动集成节点（代理）：

```
mqsichangebroker BrokerName -v 5
```

- 在 ACE V11 中，IBM Integration Bus 代理程序将以 5 分钟的时间间隔来收集归档记帐和统计信息数据。要将集成节点（代理）生成归档记帐及统计信息数据的时间间隔设置为相同，请按如下所示修改 `node.conf.yaml` 或 `server.conf.yaml` 文件：

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'     # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'none'         # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    #threadDataLevel: 'none'       # choose 1 of : none|basic
```

结果

配置并启动 IBM Integration Bus 代理程序之后，消息流记帐和统计信息数据将显示在以下组窗口小部件中：

- 消息流仪表板
  - 落实数和回退数
  - CPU 时间（微秒）

- 耗用时间（微秒）
- 输入字节速率
- 输入消息速率
- 输入消息大小
- 输入消息等待 CPU 时间（微秒）
- 输入消息等待耗用时间（微秒）
- 消息流错误数
- 消息处理节点统计信息

· 处理节点仪表板

- CPU 时间（微秒）
- 耗用时间（微秒）
- 调用次数
- 处理节点状态
- 终端统计信息

启用 JVM 资源统计信息

关于此任务

要对属于集成节点（代理）的集成服务器启用 JVM 资源统计信息，请从集成节点（代理）安装目录的 bin 目录发出 **mqsichangeresourcestats** 命令。

切记: JVM 资源统计信息是可选项，因为考虑到处理此数据的代理程序成本较高，每 20 秒仅显示少量数据属性。请务必考虑您是否确实需要 JVM 资源统计信息数据。

过程

· 要对集成节点（代理）中的所有集成服务器启用统计信息，请发出以下命令:

```
mqsichangeresourcestats BrokerName -c active
```

· 在 ACE V11 中，要对集成节点（代理）中的所有集成服务器启用统计信息，请按如下所示修改 node.conf.yaml 文件:

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    threadDataLevel: 'basic'      # choose 1 of : none|basic
   Resource:
     reportingOn: true       # choose 1 of : true|false, default false
......
```

注: 如果想要禁用此设置，请注释掉 **reportingOn: true**。

· 要对集成节点（代理）中的给定集成服务器启用统计信息，请发出以下命令:

```
mqsichangeresourcestats BrokerName -e IntegrationServerName -c active
```

- 在 ACE V11 中，要对集成节点（代理）中给定的集成服务器启用统计信息，请按如下所示修改 server.conf.yaml 文件：

```
Statistics:
  # Application message flows will by default inherit Snapshot and Archive values
  # set here
  Snapshot:
    #publicationOn: 'inactive' # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat json,xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    #nodeDataLevel: 'none'     # choose 1 of : none|basic|advanced
    #outputFormat: 'usertrace' # comma separated list of :
                               # csv,bluemix,json,xml,usertrace
    #threadDataLevel: 'none'   # choose 1 of : none|basic
  Archive:
    archivalOn: 'active'    # choose 1 of : active|inactive, default inactive
                               # Ensure Events.OperationalEvents.MQ|MQTT
                               # is set for outputFormat xml
    #accountingOrigin: 'none'  # choose 1 of : none|basic
    majorInterval: 5         # Sets the interval in minutes at which
                             # archive statistics are published
    nodeDataLevel: 'advanced'     # choose 1 of : none|basic|advanced
    outputFormat: 'xml' # comma separated list of : csv,xml,usertrace
    threadDataLevel: 'basic'   # choose 1 of : none|basic
  Resource:
    reportingOn: true        # choose 1 of : true|false, default false
```

注: 如果想要禁用此设置，请注释掉 **reportingOn: true**。

结果

JVM 资源统计信息数据将显示在以下组窗口小部件中：

- 垃圾回收计数
- 垃圾回收持续时间
- JVM 非堆内存
- JVM 堆内存

启用事务跟踪

开始之前

1. 确保安装 IBM Integration Bus 代理程序。提供了名为 KQIUserExit 的用户出口来为 IBM Integration Bus 启用事务跟踪。
2. 确保将启动集成节点（代理）的用户有权访问“KQI 用户出口”模块目录。即，确保将用于启动集成节点（代理）的用户标识添加到其中安装 IBM Integration Bus 代理程序 的组。

关于此任务

必须将 KQIUserExit 用户出口部署到集成节点（代理）。否则，即使您已启用 IBM Integration Bus 代理程序进行事务跟踪，中间件和拓扑仪表板中也没有任何可用的数据。

提示: 以下 IBM Integration Bus 节点由 KQIUserExit 用户出口作为未检测服务包含在中间件和拓扑仪表板中：

- 指定了 ODBC 数据源的数据库和计算节点
- TCP/IP 节点
- 远程 FTP 或 FTPS 服务器的文件节点
- MQ 节点（除非已经过检测）

过程

要为 IBM Integration Bus 启用事务跟踪，请完成下列步骤：

1. Linux AIX
   关闭任何已装入 MQSI 环境的代理 shell。
2. 使用下列其中一种方法打开 IBM Integration Bus 命令控制台。如果安装了多个版本的集成节点（代理），请确保启动正确版本的命令控制台。
   - Windows 单击**开始 > IBM Integration Bus > IBM Integration Console**
   - Linux AIX 从集成节点（代理）安装目录的 bin 目录中，发出 **mqsiprofile** 命令。
3. 使用 **mqsistop** 命令停止您想要配置的集成节点（代理）。
4. 通过使用 **mqsichangebroker** 命令添加 KQIUserExit 用户出口，对集成节点（代理）中的消息流启用事务跟踪。
   - 要对集成节点（代理）中的所有消息流启用事务跟踪，请运行以下命令：

     ```
     mqsichangebroker broker_name -e "KQIUserExit"
     ```

   - 要对集成节点（代理）中的特定消息流启用事务跟踪，请运行以下命令：

     ```
     mqsichangeflowuserexits broker_name -e execution_group_name -k application_name -f
     message_flow_name -a "KQIUserExit"
     ```

5. 或者，在 Ace V11 中，通过向 node.conf.yaml 或 server.conf.yaml 文件添加 KQIUserExit，对集成节点（代理）中的消息流启用事务跟踪。

   ```
   UserExits:
   activeUserExitList: 'KQIUserExit' # Specify the name
   #of an installed user exit to activate.
   ```

   注: 如果想要禁用事务跟踪，请注释掉 **activeUserExitList: 'KQIUserExit'**。
6. 使用 **mqsistart** 命令重新启动集成节点（代理）。

禁用事务跟踪

过程

要为 IBM Integration Bus 禁用事务跟踪，请完成下列步骤：

1. 使用下列其中一种方法打开 IBM Integration Bus 命令控制台。如果安装了多个版本的集成节点（代理），请确保启动正确版本的命令控制台。
   - Windows 单击**开始 > IBM Integration Bus > IBM Integration Console**
   - Linux AIX 从集成节点（代理）安装目录的 bin 目录中，发出 **mqsiprofile** 命令。
2. 使用以下其中一种方法，对集成节点（代理）中的消息流禁用事务跟踪：
   - 要对特定的消息流禁用事务跟踪，请使用 **mqsichangeflowuserexits** 命令：

     ```
     mqsichangeflowuserexits broker_name -e execution_group_name -f message_flow_name -a ""
     ```

   - 要对集成节点（代理）中的所有消息流都禁用事务跟踪，请先使用 **mqsistop** 命令停止该集成节点（代理），然后发出 **mqsichangebroker** 命令：

     ```
     mqsichangebroker broker_name -e ""
     ```

下一步做什么

- 要进行事务跟踪，在对 IBM Integration Bus 启用事务跟踪之后，还必须对代理程序启用事务跟踪。有关指示信息，请参阅第 248 页的『配置 IBM Integration Bus 代理程序的事务跟踪』。

· 如果已对集成节点（代理）启用快照数据收集，请将 IBM Integration Bus 代理程序配置为不存储任何快照数据。由于设置的 20 秒快照时间间隔所需的数据量和处理，Cloud APM 不支持快照记帐和统计信息数据。有关指示信息，请参阅第 248 页的『禁用代理程序的快照数据收集』。

禁用代理程序的快照数据收集

由于设置的 20 秒快照时间间隔所需的数据量和处理，Cloud APM 不支持快照记帐和统计信息数据。如果已为代理启用快照数据收集，请记得将 IBM Integration Bus 代理程序 配置为不存储快照数据。

过程

1. 在文本编辑器中打开代理程序配置文件。根据操作系统，代理程序配置文件为以下目录之一：
   · Linux AIX *install_dir*/config/<*hostname*>_qi_<*instance_name*>.cfg
   · Windows *install_dir*\TMAITM6_x64\<*hostname*>_qi_<*instance_name*>.cfg

   其中，*install_dir* 是代理程序安装目录；*hostname* 是操作系统名称；*instance_name* 是代理程序实例名称。
2. 通过向 KqiAgent 部分添加以下参数来编辑文件：

   ```
   defaultRetainRecentSnapshotSamples=0
   ```

   示例：

   ```
   INSTANCE=inst1 [
   SECTION=KqiAgent [ { agentId=inst1 } { instName=inst1 }
   {defaultRetainRecentSnapshotSamples=0}]
   SECTION=MonitorBroker:BRK1 [ { collectNodeData=NO } ]
   SECTION=MonitorBroker:BRK2 [ { collectNodeData=NO } ]
   ]
   ```
3. 保存并关闭该文件。
4. 重新启动 IBM Integration Bus 代理程序，以使更改生效。

配置 IBM Integration Bus 代理程序的事务跟踪

在 IBM Integration Bus 代理程序的“**代理程序配置**”页面上启用数据收集之后，IBM Integration Bus 事务跟踪数据会显示在中间件和拓扑仪表板中。

开始之前

· 确保使用代理程序所提供的用户出口 KQIUserExit，对 IBM Integration Bus 启用事务跟踪。如果您尚未完成此任务，请按第 246 页的『启用事务跟踪』中的指示信息进行操作。

· 确保适当地配置 IBM Integration Bus 代理程序。如果您尚未完成此任务，请按第 238 页的『配置 IBM Integration Bus 代理程序』中的指示信息进行操作。

过程

要为 IBM Integration Bus 代理程序配置事务跟踪，请完成下列步骤：

1. 从导航栏，单击 **系统配置** > **代理程序配置**。
   “**代理程序配置**”页面随即显示。
2. 单击 **IBM Integration Bus** 选项卡。
3. 选择代理程序实例的复选框，然后从**操作**列表中执行以下某个操作：
   · 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。**事务跟踪**列中的状态将更新为“已启用”。
   · 要禁用事务跟踪数据，请单击**设置事务跟踪** > **已禁用**。**事务跟踪**列中的状态将更新为“已禁用”。

结果

您已为所选择的代理程序实例配置了事务跟踪。在启用数据收集之后，事务跟踪数据可以显示在中间件和拓扑仪表板中。有关更多信息，请参阅第 81 页的『将中间件应用程序添加到应用程序性能仪表板』。

为 IBM Integration Bus 代理程序指定唯一受管系统名称

Cloud APM 控制台上显示的 IBM Integration Bus 代理程序实例名称也称为受管系统名称 (MSN)。您可以使用代理程序配置参数为每个代理程序实例指定唯一 MSN。

关于此任务

当 IBM Integration Bus 代理程序启动时，会以 *monitoredbrokername*:*agentID*:KQIB 格式为每个代理程序实例注册 MSN，其中，*monitoredbrokername* 是受监视代理的名称，*agentID* 是代理程序配置参数设置的代理程序标识。最大 MSN 长度为 32 个字符。如果 MSN 长度超过 32 个字符，将会被截断。

在以下情况下，可能需要唯一 MSN：

· 在同一系统上运行多个 IBM Integration Bus 代理程序。

· 在不同系统上运行多个同名的受监视代理。

要指定代理程序标识来获取唯一 MSN，请在交互式配置期间使用 **Agent Id** 选项或在静默响应文件中使用 **agentId** 参数。

切记: 如果在安装后未曾配置过 IBM Integration Bus 代理程序，请根据第 238 页的『配置 IBM Integration Bus 代理程序』所述步骤进行操作。

过程

· 要在交互式配置期间使用 **Agent Id** 选项，请完成以下步骤：

a) 输入以下命令：

```
install_dir/bin/iib-agent.sh config instance_name
```

其中，*instance_name* 是想要为其指定代理程序标识的代理程序实例名称。

b) 遵循各个选项来配置代理程序实例。

如果某选项无需更改，请使用缺省值。

c) Agent Id 选项显示时，请指定受管系统名称的中间限定符。

有效格式为最大长度为 8 个字符的字母数字字符串。

· 要在静默响应文件中使用 **agentId** 参数，请完成下列步骤：

a) 在文本编辑器中打开下列代理程序静默响应文件。

– Linux AIX *install_dir*/samples/iib_silent_config.txt

– Windows *install_dir*\tmaitm6_x64\samples\qi_silent_config.txt

b) 为 **agentId** 参数指定代理程序标识。

有效格式为最大长度为 8 个字符的字母数字字符串。

c) 保存并关闭静默响应文件，然后在命令行中运行以下命令：

– Linux AIX *install_dir*/bin/iib-agent.sh config *instance_name path_to_responsefile*

– Windows *install_dir*\BIN\iib-agent.bat config "*instance_name path_to_responsefile*"

其中，*instance_name* 是配置的实例的名称，*path_to_responsefile* 是静默响应文件的完整路径。

警告: 在 Windows 系统上，请勿包含用于将默响应文件的完整路径括起的双引号 (")，因为这将导致配置错误。

d) 配置完成之后，请输入下列命令以启动此代理程序：

– Linux AIX

```
install_dir/bin/iib-agent.sh start
instance_name
```

– Windows

```
install_dir\bin\iib-agent.bat start instance_name
```

下一步做什么

登录到 Cloud APM 控制台。如果具有先前 MSN 的代理程序实例仍显示为脱机，请编辑应用程序以将其移除，然后添加具有指定代理程序标识的新代理程序实例。

移除 KQIUserExit 用户出口

在卸载 IBM Integration Bus 代理程序之前，首先必须移除 KQIUserExit 用户出口。

过程

完成以下步骤，以移除为进行事务跟踪而部署到 IBM Integration Bus 的 KQIUserExit 用户出口：

1. 浏览至 IBM Integration Bus 代理程序的 bin 目录。
   - Windows *agent_install_dir*\arch\qi\bin
   - Linux AIX *agent_install_dir*/arch/qi/bin

   其中：
   - *agent_install_dir* 是代理程序安装目录。缺省值为 C:\IBM\APM（在 Windows 系统上）和 /opt/ibm/apm/agent（在 Linux 和 AIX 系统上）。
   - *arch* 是平台的体系结构代码。例如，lx8266 代表 Linux Intel V2.6（64 位）。要获取体系结构代码的完整列表，请参阅 *agent_install_dir*/archdsc.tbl 文件。
2. 运行 **configDC** 脚本以交互方式移除用户出口库：
   - Windows

     ```
     configDC.bat -disable iib_install_dir
     ```

   - Linux AIX

     ```
     ./configDC.sh -disable iib_install_dir
     ```

   其中，*iib_install_dir* 是 IBM Integration Bus 的安装目录。

示例

以下示例移除了代理程序为 AIX 系统上安装的 V9.0 代理提供的用户出口：

```
cd /opt/IBM/ITM/aix513/qi/bin
./configDC.sh -disable /opt/IBM/mqsi/9.0
```

配置 IBM MQ 设备监视

MQ Appliance 代理程序是一个多实例代理程序。安装后，必须通过创建代理程序实例来配置代理程序，然后才能开始使用代理程序进行监视。

开始之前

- 此处的方向针对的是代理程序的最新发行版，除非另有说明。

过程

- 在 Linux 和 UNIX 系统上，您可以使用提示响应的配置脚本或静默响应文件来配置代理程序。
  - 第 251 页的『通过响应提示来配置代理程序』
  - 第 252 页的『使用静默响应文件来配置代理程序』
- 在 Windows 系统上，只能使用静默响应文件配置代理程序。
  - 第 252 页的『使用静默响应文件来配置代理程序』

下一步做什么

在 Cloud APM 控制台中，转至“应用程序性能仪表板”页面以查看收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。在此处列出这些日志的缺省路径：

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

通过响应提示来配置代理程序

您必须为 MQ Appliance 代理程序分配一个实例名称并配置代理程序，然后才能开始监视 IBM® MQ 设备。

过程

要通过运行脚本并响应提示来配置代理程序，请完成下列步骤：

1. 运行以下命令：

   ```
   install_dir/bin/mq_appliance-agent.sh config instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

   示例：

   ```
   /opt/ibm/apm/agent/bin/mq_appliance-agent.sh config AQM904
   ```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 252 页的『MQ Appliance 代理程序的配置参数』，以获取每个配置参数的说明。

3. 运行以下命令以启动代理程序：

   ```
   install_dir/bin/mq_appliance-agent.sh start instance_name
   ```

   示例：

   ```
   /opt/ibm/apm/agent/bin/mq_appliance-agent.sh start AQM904
   ```

结果

现在，您可以登录 Cloud APM 控制台并使用“应用程序”编辑器将 MQ Appliance 代理程序实例添加到应用程序性能仪表板。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关使用“应用程序”编辑器的信息，请参阅第 930 页的『管理应用程序』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

要通过编辑静默响应文件并在无交互情况下运行脚本来配置代理程序，请完成下列步骤：

1. 在文本编辑器中打开以下一个目录中的 `mq_appliance_silent_config.txt` 文件。
   - Linux AIX *install_dir*`/samples/mq_appliance_silent_config.txt`
   - Windows *install_dir*`\samples\mq_appliance_silent_config.txt`

   其中，*install_dir* 是代理程序安装目录。例如，`/opt/ibm/apm/agent`。
2. 在 `mq_appliance_silent_config.txt` 文件中，指定所有必需参数的值，并根据需要修改其他参数的缺省值。

   请参阅第 252 页的『MQ Appliance 代理程序的配置参数』，以获取每个配置参数的说明。
3. 保存并关闭 `mq_appliance_silent_config.txt` 文件，然后运行下列命令：
   - Linux AIX

     ```
     install_dir/bin/mq_appliance-agent.sh config instance_name path_to_silent_file
     ```
   - Windows

     ```
     install_dir\bin\mq_appliance-agent.bat config instance_name path_to_silent_file
     ```

   其中：
   - *instance_name* 是要指定给代理程序实例的名称。例如，`AQM904`。
   - *path_to_silent_file* 是到 `mq_appliance_silent_config.txt` 文件的路径。例如，`/opt/ibm/apm/agent/samples/mq_appliance_silent_config.txt`。
4. 配置完成后，运行以下命令以启动代理程序：
   - Linux AIX

     ```
     install_dir/bin/mq_appliance-agent.sh start instance_name
     ```
   - Windows

     ```
     install_dir\bin\mq_appliance-agent.bat start instance_name
     ```

结果

现在，您可以登录 Cloud APM 控制台并使用“应用程序”编辑器将 MQ Appliance 代理程序实例添加到应用程序性能仪表板。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关使用“应用程序”编辑器的信息，请参阅第 930 页的『管理应用程序』。

MQ Appliance 代理程序的配置参数

下列各表显示 MQ Appliance 代理程序的配置参数，并根据小节进行分组。

- 第 253 页的表 24：用于接收 SNMP 事件和解码 V3 事件的属性。

· 第 253 页的表 25：Java 设置的属性。
· 第 254 页的表 26：HTTP 提供者使用的代理服务器的属性。
· 第 254 页的表 27：HTTP Server 的属性。
· 第 254 页的表 28：用于连接到 MQ 设备的属性。

表 *24. SNMP* 事件配置参数

参数名称	**描述**	**静默配置文件中的参数名称**
端口号	用于侦听 SNMP 事件的端口号。缺省值为 162。	KQZ_SNMPEVENT_PORT
安全级别	用于连接到 SNMP 事件的安全级别。可以是以下值之一： · 1=noAuthNoPriv · 2=authNoPriv · 3=authPriv 缺省值为 2。	KQZ_SNMPEVENT_ SECURITY_LEVEL
用户名	用于连接到 SNMP 代理程序的用户名。缺省值为 `snmpuser`。	KQZ_SNMPEVENT _USER_NAME
授权协议	用于连接到 SNMP 代理程序的授权协议。可以是以下值之一： · 1=MD5 · 2=SHA 缺省值为 2。	KQZ_SNMPEVENT_AUTH _PROTOCOL
授权密码	用于连接到 SNMP 代理程序的授权口令。	KQZ_SNMPEVENT_AUTH _PASSWORD
隐私密码	用于连接到 SNMP 代理程序的隐私口令。	KQZ_SNMPEVENT_PRIV _PASSWORD
陷阱配置文件	陷阱配置文件的位置。	KQZ_SNMPEVENT_ TRAPCNFG_FILE

表 *25. Java* 配置参数

参数名称	**描述**	**静默配置文件中的参数名称**
Java 跟踪级别	Java 提供者使用的跟踪级别。可以是以下值之一： · 1=Off · 2=Error · 3=Warning · 4=Information · 5=Minimum Debug · 6=Medium Debug · 7=Maximum Debug · 8=All 缺省值为 2。	JAVA_TRACE _LEVEL

表 *26.* 代理服务器配置参数

参数名称	**描述**	**静默配置文件中的参数名称**
代理主机名	代理服务器的主机名。	KQZ_HTTP _PROXY_HOSTNAME
代理端口	代理服务器的端口号。缺省值为 80。	KQZ_HTTP_PROXY_PORT
代理用户名	代理服务器的用户名。	KQZ_HTTP _PROXY_USER
代理密码	代理服务器的密码。	KQZ_HTTP _PROXY_PASSWORD

表 *27. HTTP Server* 配置参数

参数名称	**描述**	**静默配置文件中的参数名称**
HTTP 用户名	用于访问 MQ 设备 REST 管理界面的用户名。	KQZ_HTTP _USER
HTTP 密码	用于访问 MQ 设备 REST 管理界面的密码。	KQZ_HTTP _PASSWORD
已启用证书验证	是否启用证书验证。可以是以下值之一： · 1=true · 2=false 缺省值为 2。	KQZ_HTTP_CERTIFICATE _VALIDATION

表 *28. MQ* 设备连接配置参数

参数名称	**描述**	**静默配置文件中的参数名称**
设备主机或 IP 地址	MQ 设备的主机名或 IP 地址。缺省值为 `https://hostnameoripaddress:https://9.123.123.123`。	KMK_APPLIANCE _HOST_OR _IP_ADDRESS.arm1
设备端口号	到 MQ 设备的 HTTPS 连接的端口号。缺省值为 5554。	KMK_APPLIANCE _PORT_NUMBER.arm1
设备用户名	用于连接到 MQ 设备的用户名。	KMK_APPLIANCE _USER_NAME.arm1
设备用户密码	MQ 设备用户的密码。	KMK_APPLIANCE _USER_PASSWORD.arm1
代理主机标识	MQ Appliance 代理程序运行所在的系统的主机名。缺省值为 9.123.123.111。	KMK_APM_ AGENT_IDENTIFICATION .arm1
已启用证书验证	是否针对 HTTP 连接启用证书验证。 · 1=true · 2=false 缺省值为 2。	KMK_CERTIFICATE _VALIDATION_ ENABLED.arm1

配置 InfoSphere DataStage 监视

您必须配置 DataStage 代理程序，以便代理程序能够收集数据，以监视 DataStage 服务器资源的运行状况和性能。

开始之前

查看必备硬件和软件，请参阅 DataStage 代理程序的软件产品兼容性报告。

关于此任务

DataStage 代理程序是多实例代理程序。您必须手动创建第一个实例并启动代理程序。

产品版本和代理程序版本通常不同。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口的**任务/子系统**列中，右键单击**模板**，然后单击**使用缺省值配置**。

   这将打开 **Monitoring Agent for DataStage** 窗口。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。
4. 在 **Monitoring Agent for DataStage** 窗口中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 257 页的『代理程序的配置参数』。
5. 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**以启动代理程序。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

过程

1. 从命令行，将路径切换到代理程序安装目录。
   例如，`/opt/ibm/apm/agent/bin`
2. 运行以下命令，其中 instance_name 是要对实例指定的名称：
   `./datastage-agent.sh config instance_name`
3. 当命令行显示以下消息时，请输入 1 并输入：

   ```
   Edit 'Monitoring Agent for DataStage' setting? [1=Yes, 2=No]
   ```
4. 出现提示时，指定配置参数的值。

   有关配置参数的信息，请参阅第 257 页的『代理程序的配置参数』。
5. 运行以下命令以启动代理程序：

   `./datastage-agent.sh start instance_name`

配置环境变量

可以配置环境变量以更改 DataStage 代理程序的行为。

过程

1. 在文本编辑器中打开以下文件：

a) Windows *install_dir*\TMAITM6_x64\KDTENV_*instance_name*

b) Linux *install_dir*/config/.dt.environment

2. 编辑以下环境变量：

- **KDT_FIRST_COLLECTION_INTERVAL**：第一个数据收集的时间间隔（以秒为单位）。将此时间间隔设置为代理程序将收集指定的时间内（直至代理程序启动）先前作业运行数据的持续时间。缺省值为 300 秒（5 分钟）。因此，如果代理程序于下午 2:00 启动，那么其收集从下午 1:55 到下午 2:00 的作业运行数据。这是为了避免在代理程序开始收集数据时发生历史作业运行数据风暴。作业运行的所有后续代理程序数据收集仅访存自上次收集以来发生的新增作业运行。
- **KDT_SSL_CONTEXT**：在服务层 (WebSphere Application Server) 上启用的 SSL 协议。缺省值为 TLS。
- **KDT_META_SCHEMA_NAME**：针对元数据存储库创建的数据库模式的名称。缺省值为 DSODB（对于 Db2）和 xmeta（对于 MSSQL 和 Oracle 数据库）。
- **KDT_DATABASE_SERVICE_NAME**：代理程序用于连接到元数据存储库的数据库或服务名称。缺省值为 XMETA for Db2, xmeta（对于 MSSQL）和 ORCL（对于 Oracle 数据库）。
- **KDT_DISABLED_ATTRIBUTEGROUP**：其数据收集需要不可用的逗号分隔的属性组列表。对于各个属性组，可设置以下一个或多个值：JobRuns、JobProperties、JobRunLog、JobStages、JobParameters、EngineSystemConfiguration、EngineSystemResources、EngineServiceStatus、EngineStatusSummary、JobActivity、AgentConfiguration 和 JobConfiguration。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可使用静默响应文件来配置 DataStage 代理程序。在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

1. 在文本编辑器中，打开位于以下配置的静默配置文件并对所有参数指定值：

   Windows *install_dir*\samples\datastage_silent_config.txt

   Linux *install_dir*\samples\datastage_silent_config_UNIX.txt

   Windows C:\IBM\APM\samples

   Linux /opt/ibm/apm/agent/samples

   有关配置参数的信息，请参阅第 257 页的『代理程序的配置参数』。
2. 在命令行中，将路径更改为 *install_dir*\bin
3. 运行以下命令：

   Windows datastage-agent.bat config *instance_name install_dir*\samples\datastage_silent_config.txt

   Linux datastage-agent.sh config *instance_name install_dir*\samples\datastage_silent_config_UNIX.txt
4. 启动代理程序。

   Windows 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

   Linux 运行以下命令：./datastage-agent.sh start *instance_name*

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

代理程序的配置参数

配置 DataStage 代理程序时，您可更改服务层、元数据存储库以及高级配置参数。

服务层配置参数

代理程序连接到服务层所需的配置参数。

下表列出 DataStage 代理程序的服务层配置参数的详细描述。

表 *29.* 服务层配置参数的名称和描述

参数名称	描述	必填字段
主机名	服务层安装所在计算机的主机名。如果该计算机是某个域的组成部分，请提供标准域名 (FQDN)。缺省值为 localhost。	是
HTTPS 端口	服务层安装所在计算机上 REST 接口的 HTTPS 端口。缺省值为 9443。	是
WAS 用户名	用于连接到 WebSphere Application Server 的用户名。缺省值为 wasadmin。	是
WAS 密码	用于连接到 WebSphere Application Server 的密码。	是
确认 WAS 密码	**WAS 密码**字段中指定的密码。	是

元数据存储库配置参数

代理程序连接到元数据存储库所需的配置参数。

下表列出 DataStage 代理程序的元数据存储库配置参数的详细描述。

表 *30.* 元数据存储库配置参数的名称和描述

参数名称	描述	必填字段
数据库类型	元数据存储库的数据库类型。Db2 缺省值为 1。	是
主机名	元数据存储库安装所在计算机的主机名。如果该计算机是某个域的组成部分，请提供标准域名 (FQDN)。缺省值为 localhost。	是
数据库端口	元数据存储库上用于 JDBC 连接的数据库端口。缺省值为 50000。	是
数据库用户名	用于连接到操作数据库的用户名。缺省值为 dsodb。	是
数据库密码	用于连接到操作数据库的密码。	是
确认数据库密码	**数据库密码**字段中指定的密码。	是
JDBC 驱动程序路径	JDBC 驱动程序的路径（包括 Jar 文件）。例如，在 Linux 上，这是 /home/jars/db.jar。	是

高级配置参数

表 *31.* 高级配置参数的名称和描述

参数名称	描述	必填字段
Java 跟踪级别	Java 定制提供程序所使用的跟踪级别。缺省值为 2。	是

Java API 客户机配置参数

表 *32. Java API* 客户机配置参数的名称和描述

参数名称	描述	必填字段
外部 JAR 文件的类路径		否

p 配置 Internet Service Monitor

Internet Service Monitoring 代理程序能够确定是否充分执行特定服务，识别问题区域以及报告根据服务级别协议测量的服务性能。Internet Service Monitoring 代理程序 通过模仿实际用户的操作来工作。它定期轮询或测试因特网服务来检查其状态和性能。

概述

在监视因特网服务时，定义要监视的事项、人员和时间。您可以通过 Internet Service Monitoring 代理程序 配置用户界面来配置监视器。

监视器测试特定因特网服务并将测试结果转发给数据网桥。监视器模拟服务的实际用户的操作。

例如，HTTP 监视器通过模拟在用户进入页面时 Web 浏览器通常将发送的请求，定期尝试访问 Web 页面。监视器记录测试结果，然后将结果发送给数据网桥。

Internet Service Monitoring

每个监视器都设计为测试一种类型的协议或服务。例如，HTTP 监视器测试资源（例如基于超文本传输协议的 Web 页面）的可用性，FTP 监视器测试运行文件传输协议的主机之间的文件传输。

监视器可测试相同服务的大量不同实例，例如，一系列主机服务的一系列 Web 页面。

Web Service Monitoring

使用监视器的 Internet Service Monitoring 范围，您可以定制从测试 Web 页面可用性的基本 Internet Service Monitoring 提供的 Web Service Monitoring 的类型，以组合测试序列。

Internet Service Monitoring 使用大量、低复杂性轮询来测试 Web Service 的可用性。例如，如果想要监视 Web 站点的常规可用性，您可以使用 HTTP 监视器来定期轮询大量 URL。

使用监视器组合，您可以构建适用于需求的服务监视级别：

- HTTP 和 HTTPS 监视器

  通过大量运行、基本单请求测试，监视通过 HTTP 或 HTTPS 的资源的可用性。

- 事务监视器 (TRANSX)

  组合一组监视器所执行的测试序列，模拟实际用户的操作。例如，拨号服务，访问多个 Web 站点上的大量页面，然后访问电子邮件服务。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

通过用户界面配置 Internet Service Monitoring

要监视因特网服务，请创建用户概要文件、概要文件元素和监视调度。使用 Internet Service Monitoring 用户界面来创建用户概要文件、概要文件元素和监视调度。

关于此任务

用户概要文件是您会为其监视因特网或 Web 服务的客户、部门或服务组。对于每个用户概要文件，用户需要定义一个或多个概要文件元素。例如，用户可能会定义用于监视通过 HTTP 服务交付的 Web 页面的概要文件元素，或用于监视 FTP 服务的可用性的概要文件元素。用户概要文件通常包含多个概要文件元素，每个概要文件元素会测试向该用户提供的其中一个服务。每个用户概要文件还具有关联的监视调度，用于确定概要文件中定义的测试将要运行的日期和时间。

要通过 IBM Application Performance Management 仪表板来访问 Internet Service Monitoring 代理程序的配置窗口，请使用以下方法：

过程

1. 在 Application Performance Management 仪表板上，单击  图标。单击**代理程序配置**。
   此时将打开“代理程序配置”窗口。
2. 单击 **ISM** 选项卡以配置 Internet Service Monitoring 代理程序。
   您可以“创建”、“编辑”、“删除”、“刷新”、“调度”和“过滤”用户概要文件。在所选受管系统上部署创建的用户概要文件。显示的版本是受管系统的版本。概要文件名称指示针对所选受管系统部署的用户概要文件。请遵循以下步骤来配置任何概要文件并将其部署在受管系统上。
3. 要添加概要文件，请单击  图标。在对话框中输入**概要文件名称**和**描述**。
4. 单击**下一步**。
5. 从监视器下拉列表中选择监视器，然后单击**下一步**。
6. 提供字段的值，然后单击**添加**。
   可以为概要文件选择多个监视器。请参阅第 262 页的『可用的 Internet Service Monitoring 监视器』以获取可用监视器。
7. 单击**完成**。
8. 单击 图标。
9. 在**过滤器**字段中，按名称搜索用户概要文件。
10. 要在受管系统上部署所创建的概要文件，请选中要配置的所创建概要文件的复选框，然后选择受管系统。单击**部署**以在所选受管系统名称上部署概要文件。

编辑概要文件
所创建的所有用户概要文件都可编辑。

关于此任务

使用以下过程来编辑概要文件。

过程

1. 选择**概要文件名称**，然后单击  。
2. 选择要编辑的服务，然后单击**编辑**。
   a. 使用  图标添加监视器，使用 删除监视器
   b. 要重命名概要文件，请双击**概要文件名称**文本字段，修改概要文件名，然后单击**重命名概要文件**。
3. 编辑所选服务的值。
   注:
   · 要使 password 字段能够进行编辑，请双击 **username** 文本字段。用户可以编辑或添加用户名，并且更改密码。
   · 要使 **sslkeypassword** 字段能够进行编辑，请双击其文本以更改密钥。
4. 单击**保存**，然后单击  图标以刷新。

调度概要文件
可调度创建的概要文件以在特定日期和时间部署。

关于此任务

请使用以下过程来调度概要文件。

过程

1. 选择**概要文件名称**。
2. 单击**调度**按钮。
3. 通过选择日期与时间，调度概要文件。用户可拖动网格以选择任何想要的时间。
4. 单击**保存**。
5. 单击 图标以进行刷新。

删除概要文件
可以永久删除所创建的概要文件。

关于此任务

使用以下过程来删除概要文件。

过程

1. 选择**概要文件名称**。
2. 单击 图标以删除概要文件。

OID 组
对象标识 (OID) 组为特定于监视器的可选参数。这些参数可定义设备的管理信息库 (MIB) 对象的一个或多个 OID。SNMP 监视器使用 OID 组来从 MIB 对象检索数据，这些 MIB 对象的 OID 都显示在指定 OID 组中。

监视器从中抽取数据的 MIB 对象的详细信息如下所示：

· OID 值

MIB 对象实例的数字标识是使用 ASN.1 表示法（例如，`.1.3.6.1.2.1.1.2.0`）或对象名称（例如，`sysObjectID.0`）来表达的

注: 使用 ASN.1 表示法时，必须在 OID 中包含前置 `.` 字符。

注: 如果定义名称的 MIB 文档可供监视器访问，那么只能使用对象的实例名称来指定 OID 值。MIB 文档的缺省目录为 `$ISHOME/mibs`。

· OID 名称

MIB 对象的名称，例如，`sysObjectID`。在服务级别分类和 `$oidName`*n* 监视器元素中使用此名称。

· OID 单位

MIB 对象中包含的数据的单位。例如，秒、字节或比特/秒 (BPS)。设置为 BPS 可启用 OID 的比特/秒计算。比特/秒值计算方式如下：

```
current_poll_value - prev_poll_value) / poll_interval * 8
```

· 选择器

MIB 对象的索引值。下表显示了生成选择器的示例，该选择器会使用行索引 2 在所有 `ifDescr` 行中搜索 `FastEthernet0/1` 值。然后查询 `ifPhysAddress.2` 行，并返回值 `0:6:53:34:d2:a1`。这种方式并不会直接指定索引 2，因此如果 `FastEthernet0/1` 的索引发生更改，那么无需重新配置 OID 组。

表 *33.* 索引值的使用

MIB 对象	**MIB 对象值**
OID 值	`ifPhysAddress`
OID 名称	`FastEthernet0/1PhysicalAddress`
OID 单位	字符串
选择器	`ifDescr=FastEthernet0/1`

创建 *OID* 组和 *MIB* 对象

OID 组以全局方式创建，并且可供监视支持 SNMP 的设备的所有用户概要文件使用

过程

完成以下步骤来创建 OID 组和 MIB 对象。

1. 单击 **OID** 按钮以在 Internet Service Monitoring“代理程序”仪表板上创建 OID 组。
2. 单击 ⊕ 图标，并在 **OID 组名**字段中输入 OID 组名。
3. 单击 ⊕ 图标以添加 MIB 对象。
   a. 输入 MIB 对象的**值**、**名称**、**单位**和**选择器**。
   b. 单击**添加**。

   已成功创建 MIB 对象。
4. 单击 ↻ 图标以进行刷新。
   已成功创建 OID 组
5. 选择 **OID 组名**，单击**查看**以查看在选中的 **OID 组**下创建的所有 MIB 对象的列表。
6. 单击**关闭**。

编辑 *OID* 组和 *MIB* 对象

您可以编辑 OID 组。在创建 OID 组时或者在创建 OID 组之后，也可以编辑 MIB 对象。

过程

完成以下步骤来编辑 OID 组。

1. 单击 **OID** 按钮以在 Internet Service Monitoring“代理程序”仪表板上编辑 OID 组。
2. 从 **OID 组名**列表中选择 OID 组名，然后单击 ✎ 图标。
3. 从**编辑 OID 组**中选择值，然后单击 ✎ 图标。
4. 根据需求修改 MIB 对象字段，然后单击**保存**按钮。
5. 单击 **OID 组**弹出页面上的**保存**。
6. 单击**关闭**。

删除 *OID* 组

MIB 对象包含在 OID 组中并且供 SNMP 监视器用来获取数据。您可以从 OID 组删除单个 MIB 对象或者通过删除整个 OID 组来删除所有 MIB 对象。

过程

完成以下步骤来删除 OID 组。

1. 要删除 ODI 组，请单击 Internet Service Monitoring“代理程序”仪表板上的 **OID** 按钮。
2. 从 **OID 组名**列表中选择“OID 组名”，然后单击 ⊖ 图标。
   将删除 OID 组以及 MIB 对象。
3. 单击**关闭**。

删除 *MIB* 对象

过程

完成以下步骤以删除 MIB 组。

1. 要删除 MIB 对象，请单击 Internet Service Monitoring“代理程序”仪表板上的 **OID** 按钮。
2. 从 **OID 组名**列表中选择“OID 组名”，然后单击 ✎ 图标。
3. 选择 MIB 对象值，然后单击 ⊖ 图标。

将删除 MIB 对象。

4. 单击**编辑 OID 组**弹出页面上的**保存**。
5. 单击**关闭**。

可用的 Internet Service Monitoring 监视器

Internet Service Monitoring 代理程序是涵盖广泛因特网服务的监视器套件。

下表列出 Internet Service Monitoring 代理程序中可用的监视器及其监视的服务类型。

表 *34.* 可用的因特网服务监视器

监视器名称	所监视的服务的类型
DHCP	动态主机配置协议。 要配置 DHCP，请参阅第 272 页的『DHCP 监视器』。
DNS	域名服务。 要配置 DNS，请参阅第 274 页的『DNS 监视器』。
FTP	文件传输协议。 要配置 FTP，请参阅第 278 页的『FTP 监视器』。
HTTP	超文本传输协议。 要配置 HTTP，请参阅第 282 页的『HTTP 监视器』。
HTTPS	超文本传输协议（安全）。 要配置 HTTPS，请参阅第 290 页的『HTTPS 监视器』。
ICMP	因特网控制报文协议。 要配置 ICMP，请参阅第 295 页的『ICMP 监视器』。
LDAP	轻量级目录访问协议。 要配置 LDAP，请参阅 第 299 页的『LDAP 监视器』。
IMAP4	因特网消息访问协议。 要配置 IMAP4，请参阅第 304 页的『IMAP4 监视器 』。
NTP	网络时间协议。 要配置 NTP，请参阅 第 308 页的『NTP 监视器』。
NNTP	网络新闻传输协议。 要配置 NNTP，请参阅 第 311 页的『NNTP 监视器』。
POP3	邮局协议。 要配置 POP3，请参阅第 315 页的『POP3 监视器』。
RADIUS	远程认证拨入用户服务。 要配置 RADIUS，请参阅 第 319 页的『RADIUS 监视器』。
RPING	远程 Ping（Cisco、Juniper 和 RFC2925）。 要配置 RPING，请参阅 第 324 页的『RPING 监视器』。
RTSP	实时流协议。要 配置 RTSP，请参阅 第 329 页的『RTSP 监视器』。

表 *34.* 可用的因特网服务监视器 *(续)*	
监视器名称	**所监视的服务的类型**
SAA	Cisco Service Assurance Agent。要 配置 SAA，请参阅 第 334 页的『SAA 监视器』。
SIP	会话启动协议。要 配置 SIP，请参阅 第 349 页的『SIP 监视器』
SMTP	简单邮件传输协议。要配置 SMTP，请参阅第 354 页的『SMTP 监视器』。
SNMP	简单网络管理协议。要 配置 SNMP，请参阅 第 357 页的『SNMP 监视器』。
SOAP	基于 XML 的消息传递协议。要 配置 SOAP，请参阅 第 362 页的『SOAP 监视器』。
TCPPort	传输控制协议。 要配置 TCPPort，请参阅 第 366 页的『TCPPort 监视器』
TFTP	次要文件传输协议。要 配置 TFTP，请参阅 第 369 页的『TFTP 监视器』。
TRANSX	事务。 要配置 TRANSX，请参阅第 374 页的『TRANSX 监视器』。

Files

可执行文件

每个因特网服务监视器由可执行文件、属性文件、规则文件和日志文件组成。

监视器可执行文件位于 `$ISHOME/platform/arch/bin` 目录中。`arch` 的值是 Windows - `win 32` 操作系统的体系结构代码。

属性文件

属性文件是文本文件，包含前置散列符号的缺省设置。

要更改设置，请更改缺省设置并移除散列符号或者复制并粘贴包含缺省设置的行，进行更改并移除散列符号。这使您能够在以后复原缺省值。监视器属性文件位于 `$ISHOME/etc/props` 目录中。

规则文件

规则文件类似于 IBM Application Performance Management Netcool/OMNIbus 探测规则文件。有关其语法的信息，请参阅 *IBM Application Performance Management Netcool/OMNIbus* 探测器和网关指南。

监视器规则文件位于 `$ISHOME/etc/rules` 目录中。

日志文件

日志文件存储有关监视器操作的消息。

监视器日志文件位于 `$ISHOME/log` 目录中。`MessageLog` 属性确定日志文件的位置和名称。`MessageLevel` 属性选择写入到日志文件的信息级别，例如，详细调试消息或不可恢复错误消息。`MaxLogFileSize` 属性确定滚动前日志文件的大小。

日志文件的缺省名称为 `name.log`，其中，*name* 是监视器的名称。

常用功能部件

存在许多对于所有因特网服务监视器常用的功能部件。这些功能部件由属性、监视器生成的结果以及状态消息组成。

本节描述所有监视器的属性。个别监视器部分中描述了特定于监视器的属性。

在下表中，缺省属性参数在适用情况下带有下划线。

表 *35.* 常用属性

属性名称	属性参数	描述
AddRoute	0|1	创建从监视器使用的网络接口 IP 地址到受监视主机 IP 地址的路由。 0 - 已禁用 1 - 已启用（监视器使用概要文件元素中指定的路径，而不是在另一个网络接口上）。 **注:** 在 AIX 和 HP-UX 平台上不支持此属性。
BridgeIPAddress	不适用	指定 Databridge 的 IP 地址。此属性不可配置；Databridge 始终位于本地主机上。
BridgePort	整数	Databridge 使用的端口号。请将此属性与 Databridge SocketPort 属性设置为相同值。 缺省值：9510
BridgeSSLAuthenticatePeer	0|1	如果要在监视器与网桥之间或网桥与代理程序之间配置 SSL 认证，请将 BridgeSSLAuthenticatePeer 设置为 1 并重新启动网桥。此操作从服务器对证书进行认证。证书存储在 BridgeSSLTrustStore 中。 0 - 已禁用 1 - 已启用
BridgeSSLCertificateFile	字符串	指定 Bridge SSL 数字证书的路径和文件名。 缺省值：$ISHOME/certificates/monitorCert.pem
BridgeSSLCipherSet	字符串	指定 CipherSet。如果更新该值，请使用 OpenSSL 文档中定义的 Cipher 语法。 **注:** 请在 Internet Service Monitoring 代理程序、所有监视器和 Databridge 上设置相同值。 缺省值：RC4:3DES:DES:+EXP
BridgeSSLDisableSSLv2	0|1	确定接受哪些类型的套接字。 · 如果设置为 0，那么会接受 SSLv2 和 SSLv3。 · 如果设置为 1，那么仅在 SSLv3 中打开套接字。 **限制:** 请在 Internet Service Monitoring 代理程序、所有监视器和 Databridge 上设置相同值。

表 35. 常用属性 (续)

属性名称	属性参数	描述
BridgeSSLEncryption	0|1	启用网桥 SSL 加密。请将此属性与对应的 Databridge 属性设置为相同值。 0 - 已禁用 1 - 已启用 **注:** 请为所有监视器设置相同值。
BridgeSSLKeyFile	字符串	网桥 SSL 专用密钥文件的路径和文件名。 缺省值: $ISHOME/certificates/monitorKey.pem
BridgeSSLKeyPassword	字符串	用于加密网桥 SSL 专用密钥的密码。 缺省值: tivoli
BridgeSSLTruststore	字符串	用于认证的可信证书文件的路径和文件名。仅当使用 BridgeSSLAuthenticatePeer 设置时，才需要此属性。 如果要在监视器与网桥之间或网桥与代理程序之间配置 SSL 认证，请将 BridgeSSLAuthenticatePeer 设置为 1 并重新启动网桥。此操作从服务器对证书进行认证。可以将证书存储在 SSLTrustStoreFile 和 SSLTrustStorePath 中。 缺省值: · SSLTrustStoreFile，$ISHOME/certificates/trust.pem · SSLTrustStorePath，$ISHOME/certificates/ 要添加新证书，请完成以下步骤之一: · 将证书添加到 SSLTrustStoreFile 文本文件中的列表末尾。 · 将证书添加到 SSLTrustStorePath 目录，并运行 OpenSSL c_rehash *certificate_dir* 命令以散列证书。
BridgeTimeout	整数	监视器等待来自 Databridge 的响应的时间（以秒为单位）。
ConfigFile	字符串	用于指向监视器配置文件。 缺省值: 空白（空字符串）。
ConfigurationCheckInterval	整数	监视器检查对概要文件的更改的时间间隔（以秒为单位）。 缺省值: 1

表 35. 常用属性 (续)

属性名称	属性参数	描述
Datalog	0|1	强制监视器将性能数据记录在数据日志文件中。性能数据会记录在以下位置: $ISHOME/datalogs/*userprofile* 0 - 已禁用 1 - 已启用
DatalogFormat	字符串	定义数据日志文件的格式。参数是元素的空格分隔列表，其值应存储在数据日志文件中。对于写入到数据日志文件的每个轮询结果，系统会记录当前时间 ($time) 和所需时间 ($totalTime)，后跟此属性中定义的所有元素。
DatalogNameFormat	字符串	数据日志文件名的格式。
域	字符串	指定运行监视器的主机的域名。如果未设置此属性，那么监视器会尝试使用 NIS 和 DNS 配置来猜测域名。
DumpProps	不适用	显示监视器的所有属性的列表。
FullHostInfo	0|1	指定将 $host 元素映射到 IP 地址元素 $hostIP（如果 $host 是 DNS 名称）还是映射到 DNS 名称元素（如果 $host 是 IP 地址）。 0 - 已禁用 1 - 已启用 **注:** 此属性在 TRANSX 监视器中不可用。
GroupID	字符串	应当用于运行监视器的组标识。
帮助	0|1	显示命令行选项的帮助而不运行监视器。 0 - 已禁用 1 - 已启用
IdentifierChecksumFields	字符串	不推荐。
IgnoreUnmatchedDVC	0|1	如果特定服务级别分类不匹配，那么监视器不会创建元素，然后请在服务级别计算中忽略该元素。 **提示:** 在 Internet Service Monitoring 代理程序 的更低发行版中，服务级别分类称为离散值分类 (DVC)。 0 - 已禁用 1 - 已启用

表 35. 常用属性 (续)

属性名称	属性参数	描述
IpAddress	字符串	指定监视器在测试期间使用的网络接口的 IP 地址。 如果未设置此属性，那么监视器会尝试使用主机名查找来确定主机机器的 IP 地址。如果主机机器具有多个网络接口，那么此尝试可能会失败。
Manager	字符串	指定 ObjectServer 事件去重中使用的管理应用程序的名称。
MaxCCA	整数	设置监视器在任何时间可以具有的最大并发连接数。请注意，如果将该值设置过高，那么可能会严重影响监视器的性能。 此属性不可用于 ICMP 监视器。 缺省值：10
MaxLogFileSize	整数	日志文件的最大大小（以字节为单位）。 缺省值：1048576
MessageLevel	字符串	要发送到消息日志的消息的最低级别。值（按严重性的升序）为：debug、info、warn、error 和 fatal。 缺省值：warn
MessageLog	字符串	日志文件的位置。 缺省值：$ISHOME/log/*monitor*.log
MinPoll	整数	定义允许的最小轮询时间间隔。如果任何监视器配置文件将轮询时间间隔设置为小于该值，那么会覆盖配置文件中的值。 缺省值：60
MsgDailyLog	整数	启用每日日志文件生成。 缺省值：0 - 已禁用每日日志
MsgTimeLog	字符串	指定监视器在 MsgDailyLog 已启用的情况下生成每日日志之前所要经过的时间（采用 24 小时制 HHMM）。 缺省值：0000 - 午夜 12 点
Name	字符串	监视器的名称。设置此属性会将 PropsFile、RulesFile 和 MessageLog 属性重置为其缺省值。
NewProfileCheckMultiple	整数	用于指示监视器在检查概要文件更改时检查新配置文件的频率的倍数。 缺省值：10

表 35. 常用属性 (续)

属性名称	属性参数	描述
NoRecover	整数	指示监视器不要恢复存储转发文件。 缺省值：0 - 不禁止恢复
Pause	整数	设置监视器衍生线程的时间间隔（以秒为单位）。将此属性设置为较高值（例如 100 或更大值）会强制监视器以较慢速率衍生线程。通常仅在慢速系统上才必须增大该值。 在 ICMP 监视器上不支持此属性。 缺省值：50
PreviousFields	字符串	此属性指定的元素（使用“<element>,<element>,...”形式）存储用于一个轮询，并以字符串 previous 为前缀。
Profile	字符串	客户概要文件或要使用的概要文件的名称。字符串可以是单个概要文件名称、概要文件名称的空格分隔列表，或用于强制监视器使用所有可用概要文件的 *。 缺省值：*
ProfileUpdateTimeout	整数	概要文件在可由监视器读取并进行更新之前必须保持静态的毫秒数。允许的范围为 1-20000 毫秒。 缺省值：100
PropsFile	字符串	属性文件的名称。 缺省值：$ISHOME/etc/props/*monitor*.props
QFile	字符串	设置存储转发文件的名称。 缺省值：$ISHOME/var/*monitor*.saf。
QSize	整数	设置存储转发文件的保留大小（以字节为单位）。 缺省值：10240000
用户标识	字符串	应当用于运行监视器的用户标识。 **注:** 请勿将此属性用于 DHCP 监视器。
Version	不适用	列显监视器版本而不运行监视器。

常见监视器元素

本部分描述所有监视器生成的元素。个别监视器部分中描述了特定于监视器的元素。可以在 Internet Service Monitoring 代理程序仪表板中查看生成的元素。

如果使用 IBM Application Performance Management，那么可在代理程序仪表板上被视为属性的元素由 Internet Service Monitoring 代理程序生成的映射文件来确定。此映射文件不可配置。

第 269 页的表 36 列出所有监视器生成的元素。星号 (*) 指示的元素用作工作空间属性。属性名称显示在括号中。缺少星号指示没有等效的工作空间属性。显示在括号中但没有元素的属性指示这些属性仅可用作工作空间属性，而没有任何等效元素。

表 *36.* 常见监视器元素

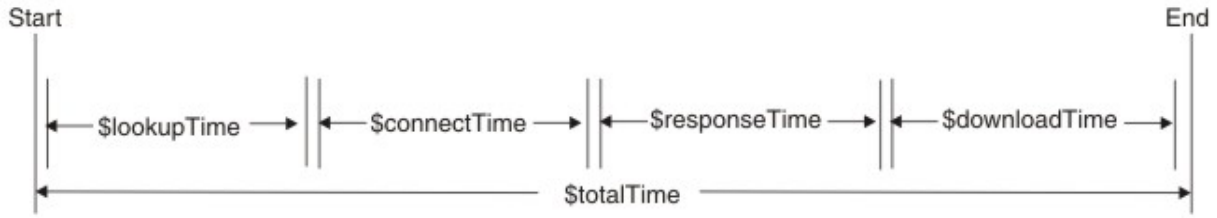
元素名称	**元素描述**
$consecutiveFailures	如果 $failureRetests 为非零值，并且根据服务级别分类，测试失败，那么将从值 1 开始创建此元素。值会增大直至测试不再失败为止（此时 $consecutiveFailures 设置为 0），或者直至发生以下轮询。 如果在此轮询中，服务级别已通过或再次开始提高，那么将不再创建此元素。如果此元素的值超过 $failureRetests 的值，那么 $consecutiveFailures 的值会重置为 1。 **注:** TRANSX 监视器不会生成此元素。
$datalogPath* (guid)	监视器使用的数据日志文件的路径。工作空间属性使用路径的最后 100 个字符。
$description* (Description)	包含监视器概要文件元素的 **Description** 字段中提供的文本描述。
$failureRetestInterval	失败重新测试期间使用的轮询时间间隔。仅当 $failureRetests 大于 0 时，此元素才有效。如果重新测试时间间隔大于一般轮询时间间隔，那么它会设置为等于一般轮询时间间隔。 **注:** TRANSX 监视器不会生成此元素。
$failureRetests	在记录失败事件并将其发送到 ObjectServer 之前必须超过的服务级别失败数。 **注:** TRANSX 监视器不会生成此元素。
$host* (Host)	主机或服务器的名称。存储在配置文件中。
$hostName	包含 $host 元素的主机名（如果 $host 是 IP 地址）。
$hostIP	包含 $host 的主机 IP（如果 $host 是 DNS 名称）。
$identchecksum* (Identchecksum)	概要文件元素的标识。
$lastServiceLevel* (LastServiceLevel)	上次轮询的服务级别号。如果概要文件更改，该属性将被清除。
$lastServiceLevelCounter	先前轮询中的 serviceLevelCounter。如果概要文件发生更改，那么会重置此元素。
$monitorDNSdomain	DNS 所使用的运行监视器的机器的域名。

表 36. 常见监视器元素 (续)	
元素名称	**元素描述**
$monitorHost* (MonitorLocation)	运行监视器的主机名称。
$monitorNISdomain	NIS（网络信息服务）所使用的运行监视器的主机的域名。
$monitorDomain	覆盖 $monitorDNSdomain 和 $monitorNISdomain 设置。
$message* (ResultMessage)	用于描述轮询结果的文本字符串。例如 Connection failed、OK 或 Success。
(Node)	运行 Internet Service Monitoring 的系统的名称。此属性由 Internet Service Monitoring 代理程序进行添加。
$pollInterval	每个监视器中指定的轮询时间间隔。
$resultString* (ResultString)	用于指示应用于轮询结果的服务级别分类的文本字符串。例如，TotalTime > 20。
$service* (Service)	受监视服务的名称。例如 FTP 或 HTTP。
$serviceLevel* (ServiceLevel)	轮询的服务级别编号，如服务级别分类中所定义： 0 - Unknown 1 - Good 2 - Marginal 3 - Failed
$serviceLevelCounter	服务级别编号已保持不变的次数。
(ServiceLevelString)	与返回的服务级别关联的字符串（Unknown、Good、Marginal 或 Failed）。
$startTimePoll	轮询启动时的时间。
$time	发生轮询时的 UNIX 时间（以秒为单位）。
$timeStamp* (Timestamp)	执行测试时的日期和时间。时间戳记格式使用本地设置。
$transxName	事务名称。如果在事务中使用监视器，那么该监视器会生成此元素。
概要文件详细信息	
$profile* (IsmProfile)	用户概要文件的名称。

表 36. 常见监视器元素 (续)	
元素名称	**元素描述**
计时 - 有关如何测量计时的信息，请参阅第 271 页的『时间计算』。	
$timeout	服务器必须响应的秒数。从配置文件中获取。
$totalTime* (TotalTime)	运行操作所需的总时间（以秒为单位）。这包括所有查找、连接和下载时间（适用时），以及临时处理时间。

时间计算

监视器尝试将完成轮询所需的时间划分为不同的限时阶段。例如，这可能包括获取主机 IP 地址所需的时间或成功连接到主机所需的时间。



$totalTime 始终比其他时间的总和略长，因为它包括监视器的活动（例如处理接收到的数据和执行系统调用）引起的开销。$totalTime 以秒为单位进行测量。

状态消息
监视器在各项服务测试后会返回生成的状态消息。状态消息指示测试的结果。

消息通常源自受监视服务或监视器外的网络环境。第 271 页的表 37 描述使用 IBM Application Performance Management 时监视器在 ResultMessage 属性中返回的常见状态消息。个别监视器部分中描述了特定于监视器的状态消息。

除个别监视器提供的消息以外，某些监视器（例如 HTTP 监视器）还会报告底层操作系统的消息。例如，如果 TCP 连接失败，那么 Internet Service Monitoring 代理程序 将使用操作系统定义的字符串（例如 **connection refused**、**timeout**、**network unreachable**）和其他字符串。

表 37. 常见状态消息	
Message	**描述**
OK	监视器请求已成功。 监视器可能具有指示进程成功的其他状态消息。请参阅每个监视器的状态消息部分。
Received response to request not originating from this monitor - ignored	从服务器接收到对未源自指定监视器的消息的回复。
Connection failed Connect to server failed	监视器无法连接到服务器。请参阅日志文件以获取更多信息。
Connection closed unexpectedly	与服务器的连接已断开。

表 37. 常见状态消息 (续)	
Message	**描述**
Connection timed out	连接成功，但是之后服务器停止响应。
Connection closed by foreign host	远程主机在监视器期望的时间前关闭连接。
Timed out waiting to read/ write	已建立与受监视服务器的数据连接，但是该服务器已停止响应。
No Response from server	请求已超时。
Format Error Server Failure No such host or domain Not Implemented Request refused 未知错误	受监视服务器已返回错误。
Network is down Network is unreachable Network dropped connection on reset Software caused connection abort Connection reset by peer Connection timed out Connection refused Host is down No route to host Remote peer released connection	网络发生问题。

DHCP 监视器

DHCP 监视器会检查 DHCP 服务器的可用性和响应时间。

通过使用总时间、查找时间或响应时间根据 DHCP 服务器对来自 DHCP 监视器的请求作出响应所需的时间来指定服务级别分类。

表 38. *DHCP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_dhcp
属性文件	$ISMHOME/etc/props/dhcp.props
规则文件	$ISMHOME/etc/rules/dhcp.rules
日志文件	$ISMHOME/log/dhcp.log

DHCP 监视器配置准则

DHCP 监视器通过充当有限 DHCP 客户机来测试 DHCP 服务。通过使用 UDP 作为所建立连接传输协议，向同一网络上的目标 DHCP 服务器发送 DHCP INFORM 请求，并等待来自服务器的相应 DHCP ACK。监视器不会请求 IP 地址，否则会影响现有 IP 地址的有效期。

注: 受监视 DHCP 服务器必须支持 DHCP INFORM 请求并符合 RFC2131。

DHCP 监视器必须作为 `root` 运行，因为它绑定到的端口小于 1024。

限制

DHCP 监视器不能使用任何通过使用 DHCP 客户机配置的网络接口。请改将监视器配置为使用未动态分配其 IP 地址的网络接口。

配置 DHCP 监视器服务测试

表 *39. DHCP* 监视器配置

字段	描述
服务器	DHCP 服务器的主机名。例如 dhcp1.mycompany.com
localip	监视器用来执行测试的网络接口的 IP 地址。例如 192.168.n.n
描述	提供关于该元素描述性信息的文本字段。例如 DHCP monitor
port	DHCP 服务器的端口号，并且缺省值为 67。
localport	监视器用于执行测试的端口号，并且缺省值为 68。
超时	等待服务器作出响应的时间（以秒为单位），并且缺省值为 30。
retries	监视器在退出之前必须重试与 DHCP 服务器的连接的次数，并且缺省值为 0。例如 2。
poll	通过使用当前概要文件元素对服务器进行的每次轮询之间间隔的时间（以秒为单位），并且缺省值为 300。
failureretests	在指示失败之前要重新测试的次数，并且缺省值为 0。
retestinterval	失败时前后两次进行重新测试之间等待的时间，并且缺省值为 30。

除所有元素常见的测试结果以外，DHCP 监视器还会生成包含特定于 DHCP 服务测试的数据的测试结果集。

表 *40. DHCP* 监视器元素

元素	描述
`$clientIP*ClientIp`	监视器运行所在的主机的 IP 地址。
`$lookupTime*(LookupTime)`	获取主机服务器 IP 地址所花的时间。
`$responseTime*ResponseTime`	建立连接到接收到数据首个字节之间的时间。
`$retries`	监视器配置期间所指定的最大重试次数。
`$router`	DCHP 服务器所返回的路由器的 IP 地址。

状态消息

当使用 IBM Application Performance Management 时，DHCP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

表 *41. DHCP* 监视器状态消息

Message	**描述**
Received DHCPACK Received DHCPNAK	DHCP 服务器对监视器发送的 DHCP 通知请求作出响应。
This monitor requires root privileges to run	以 root 用户身份登录。
Did not receive valid DHCP MESSAGE	无法识别来自 DHCP 服务器的响应。
Did not receive valid DHCP MESSAGE TYPE	无法识别来自 DHCP 服务器的响应（预期为 DHCPACK 或 DHCPNAK）。
Invalid transaction ID Received response to request not originating from this monitor - ignored	从 DHCP 服务器接收到对未源自监视器的消息的回复。
Unexpected op-code returned	在此端口上接收到意外消息。
Connection failed	指定的服务器名称无效。
Failed to send request to DHCP server	操作系统无法具体确定请求无法发送到服务器的原因，因此返回此状态消息，指示网络存在问题。
No Response from server	DHCP 服务器没有响应。

***DNS* 监视器**

DNS 监视器使用 DNS（域名系统）服务来查找有关一个或多个主机的信息。

DNS 监视器使用主机的 IP 地址来搜索主机名，或者使用主机名来搜索 IP 地址。监视器通过记录搜索结果和响应时间来度量服务的性能。监视器还会记录有关发送到服务器的每个查询的详细信息。

表 *42. DNS* 监视器文件摘要

监视器文件	**名称或位置**
监视器可执行文件	nco_m_dns
属性文件	$ISMHOME/etc/props/dns.props
规则文件	$ISMHOME/etc/rules/dns.rules
日志文件	$ISMHOME/log/dns.log

DNS 监视器配置准则

DNS 监视器可配置为查询目标主机的 IP 地址或主机名。根据查找的类型，该监视器以不同的方式与 DNS 服务器通信。

IP 地址查找

执行 IP 地址查找测试时，将为监视器提供一个主机名，用于确定 IP 地址。

监视器按以下方式测试 DNS：

1. 监视器使用 HostA（hosta.dev.net）的标准主机名称查询 DNS 服务器以请求其 IP 地址。

   如果 DNS 服务器可以找到该主机的 IP 地址，那么服务器会将该 IP 地址返回至监视器。如果 DNS 服务器无法找到主机的 IP 地址，那么它会将包含失败搜索的详细信息的消息返回到监视器。

如果请求超时，监视器将重试（如果配置了重试次数）。如果无重试次数，监视器将创建失败事件。

如果配置中指定的主机名是域名（例如 mycompany.com），而不是标准主机名称（例如 hostx.mycompany.com），那么监视器将检索有关整个域的信息。该信息存储在两个额外元素 $domainNameServer 和 $domainNameAddr 中。

2. 如果返回至监视器的消息包含规范名称，那么监视器将得出这样的结论：配置文件中给出的名称一定是别名。监视器会将规范名称发送至 DNS 服务器以请求主机的 IP 地址。

   如果 DNS 服务器使用规范名称找到了主机的 IP 地址，那么服务器会将该 IP 地址返回至监视器。如果 DNS 服务器无法找到主机的 IP 地址，那么它会将包含失败搜索的详细信息的消息返回到监视器。

3. 如果前两次查询 DNS 服务器的尝试失败，那么监视器会将 DNS 服务器的 IP 地址 (192.168.n.n) 发送到 DNS 服务器并请求其标准主机名。

   如果 DNS 服务器可以找到自己的标准主机名称，那么 DNS 服务器会将该名称返回至监视器。如果 DNS 服务器无法找到其自己的标准主机名，那么它会返回包含失败搜索的详细信息的消息。对服务器的标准主机名的请求（反向 DNS 查找请求）并非在所有类型的 DNS 服务器上都受支持。如果目标 DNS 服务器不支持反向查找，那么可以通过将 LookupServerName 属性设置为 0 来防止 DNS 监视器发送此请求。

递归查找

非递归查找可更加准确地描绘 DNS 服务器的执行情况，而递归查找则更好地指示因特网应用程序（乃至用户）所获得的 DNS 性能。DNS 监视器同时支持递归和非递归查找。

这通常是因特网应用程序执行 DNS 查询的方法。例如，Web 浏览器在其尝试解析 URL 的主机部分时始终指定递归查找。

如果 DNS 服务器由于在其数据库中不包含主机的条目而无法回复查询，那么它可以在层次结构中的更高级别以递归方式查询 DNS 服务器。

DNS 查询类型

DNS 监视器支持一组 DNS 查询类型。指定 DNS 查询类型时请使用查询代码。

表 *43. DNS* 查询类型

查询代码	**查询类型**
A	主机地址
NS	授权名称服务器
MD	邮件目标
MF	邮件转发器
CNAME	别名的规范名称
SOA	权限区域的起始点
MB	邮箱域名
MG	邮件组成员
MR	邮件重命名域名。
NULL	空 RR
WKS	众所周知的服务描述
PTR	域名指针
HINFO	主机信息
MINFO	邮箱或邮件列表信息
MX	邮件交换

表 *43. DNS* 查询类型 *(续)*	
查询代码	**查询类型**
TXT	文本字符串
AXFR	整个区域的传输
MAILB	邮箱的相关记录
MAILA	邮件代理程序 RR
ANY	所有记录

配置 DNS 监视器服务测试

使用 DNS 监视器配置参数可定义 DNS 服务测试。

表 *44.* 表 *3. DNS* 监视器配置	
字段	**描述**
服务器	主 DNS 服务器的 IP 地址。例如 192.168.n.n
host	目标主机的主机名。例如 www.myconpany.com
描述	提供关于该元素描述性信息的文本字段。例如 DNS monitor。
recursivelookups	启用或禁用递归查找。 · `recurse`（在 ismbatch 中使用 `true`）。 · `norecurse`（在 ismbatch 中使用 `false`）。 缺省值：`recurse`。
port	监视器侦听的 DNS 服务器上的端口，并且缺省值为 53。
localip	指定监视器在执行测试时绑定到的主机上的网络接口 IP 地址。如果设置了监视器的 `IpAddress` 属性，那么它会覆盖此字段的值。
querytype	测试中使用的 DNS 查询类型。有关受支持查询类型的列表，请参阅第 275 页的表 43。
超时	等待服务器响应的时间（秒）。 缺省值：`10`。
retries	监视器在退出前应该重新尝试连接 DNS 服务器的次数。
poll	轮询的间隔时间（秒）。 缺省值：`300`。
failureretests	指示失败之前重新测试的次数。 缺省值：`0`。
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：`10`。

监视元素

除所有元素常见的测试结果以外，DNS 监视器还会生成包含特定于 DNS 服务测试的数据的测试结果集。

表 45. 表 4. DNS 监视器元素	
元素	**描述**
$authoritative	仅当检索到的信息来自授权 DNS 服务器时，才会创建此元素。如果 DNS 服务器未经授权，那么不会创建此元素。
$domainEmailAddr	目标域的联系人地址。
$domainNameServer	目标域的 DNS 服务器的名称。
$fromAliasTime	对从先前查询中接收到的规范名称发出请求与接收 IP 地址之间间隔的时间。
$localIP	监视器配置为使用的本地 IP 地址。此元素在仅有一个接口的机器上可能为空白。
$lookup*(HostLookup)	监视器尝试找到的目标主机的主机名或 IP 地址。
$lookupCName	目标主机的正式主机名。仅当正式主机名与 $lookupName 中的主机名不同时，才会创建此元素。
$lookupIP*(HostIp)	目标主机的 IP 地址。
$lookupName*(Host)	目标主机的完整主机名。
$mxRecords	找到的 MX 记录数。
$port	服务受监视的端口。
$queryType	测试中使用的 DNS 查询的类型。有关受支持查询类型的列表，请参阅第 275 页的表 43。
$responseTime*(ResponseTime)	监视器向 DNS 服务器发出请求与从中收到回复之间间隔的时间。
$retries	概要文件元素中所指定的最大重试次数。
$serverIP	DNS 服务器的 IP 地址。
$serverName	DNS 服务器的主机名。
$serverTime	服务器解析其自己的名称所需的时间。

MX 记录处理

针对 DNS 监视器找到的每个 MX 记录创建了两个元素：$mxHostn 和 $mxPreferencen。

$mxHostn 存储 MX 记录的主机名。$mxPreferencen 包含主机的首选项加权。n 针对每个记录对进行递增，以对其进行区分。监视器将特定主机的 MX 记录总数存储在元素 $mxRecords 中。记录对按 MX 首选项的降序进行排序。

状态消息

当使用 IBM Application Performance Management 时，DNS 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

表 46. 表 5. DNS 监视器状态消息	
Message	**描述**
Domain information received	对域名的请求已成功。
成功	请求已成功。
Invalid Response	无法识别来自 DNS 服务器的响应。

表 46. 表 5. DNS 监视器状态消息 (续)	
Message	**描述**
Connection failed	指定的服务器名称无效。
No Response from server	请求已超时。
Failed to send DNS request	网络发生问题。
No such domain (no recursion)	域名不正确。

属性

下表中描述了特定于 DNS 监视器的属性。

表 47. DNS 监视器属性		
属性名称	**属性参数**	**描述**
AcceptCNAME	0|1	如果已启用，那么 DNS 监视器将接受 DNS 响应中的规范名称，并且不执行任何进一步查找。
DNSQueryType	字符串	测试中使用的 DNS 查询类型。请参阅第 275 页的表 43 以获取受支持查询类型的列表。 缺省值：ANY。
LookupServerName	0|1	对 DNS 服务器 IP 地址启用反向 DNS 查找。 0 - 已禁用 1 - 已启用

***FTP* 监视器**

FTP 监视器通过将文件上载到 FTP 服务器或从中下载文件来测试 FTP 服务。它通过记录响应时间和数据传输率来监视服务的性能，并会监视磁盘空间和文件完整性。

表 48. FTP 监视器摘要	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_ftp
属性文件	$ISHOME/etc/props/ftp.props
规则文件	$ISHOME/etc/rules/ftp.rules
日志文件	$ISHOME/log/ftp.log

FTP 监视器配置准则

FTP 监视器可测试 FTP 服务器的可用性，其方法是使用 FTP STOR 命令将文件上载至服务器，或使用 FTP RETR 命令从服务器下载文件。

配置 FTP 监视器服务测试

下表中描述了 FTP 监视器的配置参数。

表 *49. FTP* 监视器配置	
字段	**描述**
服务器	目标 FTP 服务器或要进行 FTP 传输的机器的 IP 地址。例如 `ftp.mycompany.com`
localfile	对于 FTP GET 操作，此字段指定文件下载到的名称和路径。 对于 FTP PUT 操作，此字段指定上载到 FTP 服务器的文件的名称和路径。 缺省值为 FULL PATHNAME。例如 `$ISMHOME/etc/ism/downloads/ftp-test.tar.Z`
remotefile	对于 FTP GET 操作，此字段指定从服务器下载的文件的名称和路径。 对于 FTP PUT 操作，此字段指定文件上载到 FTP 服务器上的名称和路径。 缺省值为 FULL PATHNAME。例如 `/sales/prodlist.tar.Z`
描述	提供关于该元素描述性信息的文本字段。
port	FTP 服务器使用的缺省端口。 缺省值：21
username	用于登录到目标 FTP 服务器的用户名。
密码	用于登录到目标 FTP 服务器的密码。如果 FTP 帐户无需密码，请将此字段保留为空白。
command	供监视器使用的 FTP 命令： · GET 或 RECV — 从目标 FTP 服务器下载文件 · SEND 或 PUT — 将文件上载到目标 FTP 服务器 缺省值：GET。
conntype	指定监视器尝试传输文件时要与服务器建立的连接的类型： · 动态 · Passive 缺省值：Active。
超时	等待服务器响应的时间（秒）。 缺省值：30。
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	指示失败之前重新测试的次数。 缺省值：0。
retestinterval	前后两次失败重新测试之间等待的时间段（以秒为单位）。 缺省值：10。

正则表达式匹配

您可以通过输入最多 50 个不同的正则表达式来对下载的信息执行正则表达式搜索。FTP 监视器尝试将检索到的内容与各个正则表达式进行匹配。

如果找到指定的正则表达式的匹配，那么将在对应的 $regexpMatchn 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。$regexpStatusn 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

在服务测试期间，正则表达式用于对下载的内容执行字符串匹配。这些表达式可能包含一个或多个正则表达式运算符，这些运算符决定了表达式与哪些内容匹配。

注: 正则表达式语法可用于仅匹配单行上的字符串。Internet Service Monitoring 无法匹配包含换行符或回车符的字符串。请使用多个正则表达式来匹配涵盖多行的字符串。您还可以使用 SLC 规则来根据多个正则表达式的结果提升警报。

表 *50.* 正则表达式运算符

字符	描述
.	与任意单个字符匹配。 例如，正则表达式 `r.t` 与字符串 `rat`、`rut`、`r t` 匹配，但与 `root` 不匹配。
$	与行的末尾匹配。 例如，正则表达式 `dog$` 与字符串 `it's a dog` 的末尾匹配，但是与字符串 `There are a lot of dogs` 的末尾不匹配。
^	与行的开头匹配。 例如，正则表达式 `^When in` 与字符串 `When in the course of human events` 的开头匹配，但与 `What and When in the` 不匹配。
*	通配符前面紧挨的字符不出现或出现多次时，均满足匹配条件。 例如，正则表达式 `.*` 与任意数目的任意字符匹配。
\	将后续字符作为普通字符对待。 例如，`\$` 与美元符号字符（$）匹配，并不代表行的结尾。同样，表达式 `\.` 与句点字符而不是任何单个字符匹配。
[]	与方括号之间的任意一个字符匹配。 例如，正则表达式 `r[aou]t` 与 `rat`、`rot` 和 `rut` 匹配，但与 `rit` 不匹配。 使用连字符可指定字符的范围。 例如，正则表达式 `[0-9]` 与任意数字匹配。 还可以指定多个范围。 例如，正则表达式 `[A-Za-z]` 与任意字母大小写匹配。
|	与包含任一指定条件的短语匹配。 例如，`him|her` 与行 `it belongs to him` 和行 `it belongs to her` 匹配，但与行 `it belongs to them` 不匹配。

注: 如果您首选使用包含 花括号 {} 或双引号 "" 的输出数据字符串，那么需要在正则表达式中的每个花括号和双引号之前添加一个转义字符反斜杠 \。

例如，如果数据字符串为 {"templates":true,"mongodb":true,"ldap":true,"ucd":true,"github":true}，那么正则表达式应显示为 \{\"templates\":true,\"mongodb\":true,\"ldap\":true,\"ucd\":true,\"github\":true\}

监视器元素

除所有元素常见的测试结果以外，FTP 监视器还会生成包含特定于 FTP 服务测试的数据的测试结果集。

表 *51. FTP* 监视器元素

元素	描述
$bytesPerSec*(BytesPerSec)	平均每秒传输的字节数。
$bytesTransfered*(BytesTransferred)	上载或下载的字节数。
$checksum	Checksum 元素通常不会为服务级别分类提供有意义的值，因为在创建概要文件元素时，校验和值未知（监视器在测试进行中时计算校验和值）。$checksum 和 $previousChecksum 监视器元素旨在使用监视器的规则文件来丰富警报。
$command*(FtpCommand)	监视器发出的 FTP 命令。
$connectionType*(FtpConnection)	使用的数据连接类型。此类型可以为 ACTIVE 或 PASSIVE。
$connectTime*(ConnectTime)	连接到 FTP 服务器所需的时间。
$downloadTime	下载文件所花的时间。
$localFile*(FtpLocalFile)	文件在本地主机上的完整路径名。从配置文件中获取此元素。
$lookupTime*(LookupTime)	查找 FTP 服务器 IP 地址所需的时间。
$previousChecksum	PreviousChecksum 元素通常不会为服务级别分类提供有意义的值，因为在创建概要文件元素时，校验和值未知（监视器在测试进行中时计算校验和值）。$previousChecksum 和 $checksum 监视器元素旨在使用监视器的规则文件来丰富警报。
$regexp*n*	正则表达式。
$regexpMatch*n*	与正则表达式匹配的行的内容。
$regexpStatus*n*	正则表达式的状态与以下内容匹配： NONE - 未配置正则表达式检查 MATCHED - 找到正则表达式的匹配项 FAILED - 找不到正则表达式的匹配项
$remoteFile*(FtpRemoteFile)	远程主机（FTP 服务器）上存储的文件的完整路径名。从配置文件中获取此元素。
$responseTime*(ResponseTime)	创建连接后直至接收到目标文件的第一个字节所需的时间。
$status	FTP 服务器返回的状态码。
$transferTime*(TransferTime)	将该值设置为 $uploadTime 或 $downloadTime。

表 *51. FTP* 监视器元素 *(续)*	
元素	**描述**
$uploadTime	上载文件所需的时间。
$username	监视器登录到目标主机所使用的用户名（帐户名称）。如果 $message 包含 OK，那么从配置文件中获取此元素。

状态消息

当使用 IBM Application Performance Management 时，FTP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

表 *52. FTP* 监视器状态消息	
Message	**描述**
OK	FTP 请求已成功。
Unable to open local file for reading/writing	请参阅 FTP 监视器日志文件以获取更多信息。
Unable to read from/write to local file	
Unable to read from data connection	已建立与 FTP 服务器的数据连接，但是发生问题。
Unable to upload to ftp server	
Timed out waiting to read/write	
Connection closed by foreign host	与 FTP 服务器的连接已中断。
Connection closed unexpectedly	
Connection failed	监视器无法连接到 FTP 服务器。请参阅 FTP 监视器日志文件以获取更多信息。

***HTTP* 监视器**

HTTP 监视器检查 Web 服务器的可用性和响应时间。

其可以监视单个 Web 页面，包括通常需要用户在字段中输入数据的使用 CGI 的页面。还可以监视元素的下载时间，例如，Web 页面上的图像。

表 *53. HTTP* 监视器文件摘要	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_http
属性文件	$ISHOME/etc/props/http.props
规则文件	$ISHOME/etc/rules/http.rules
日志文件	$ISHOME/log/http.log

用于配置 HTTP 监视器的准则

HTTP 和 HTTPS 监视器检查 Web 服务器的可用性和响应时间。请在以下情境中使用 HTTP 监视器：

- 目标 Web 站点是静态的。

  对于动态 Web 站点，请使用 TRANSX 监视器。
- 基于 HTTP 协议为目标 Web 站点提供服务。

  对于通过 HTTPS 协议交付内容的 Web 站点，选择 HTTPS 监视器。

· 要执行跨多个平台的监视。

· 其中速度是决定因子（HTTP 监视器提供高性能）。

HTTP 请求类型

HTTP 监视器模拟支持 HTTP/1.0 协议的 Web 浏览器。为测试 Web 服务器，监视器将使用以下任一 HTTP 请求类型向其发送 Web 页面请求：

· HEAD

HEAD 命令将尝试访问 Web 页面并返回 HTTP 头。发出 HEAD 命令是检查 Web 页面是否可访问的快捷方式。

· GET

GET 命令将尝试访问 Web 页面并返回整个页面，包括 HTTP 头。其不会尝试返回与页面相关联的文件，例如，图像。

· GETALL

GETALL 命令尝试访问 Web 页面并返回整个页面，包含 HTTP 头、背景、图像、applet、帧、级联样式表 (CSS) 文件和脚本。类似于 HEAD 和 GET 命令，该命令也将检查 Web 页面是否可访问，但是由于 GETALL 命令返回了整个页面及其所有关联的文件，所以它指示的访问页面所花的时间可能更加真实。监视器在执行 GETALL 命令期间还使用多个线程，以更加准确地模拟 Web 浏览器的行为。

· POST

POST 命令将尝试访问包含 HTTP 表单的 Web 页面，并填写该表单的字段。将 POST 请求的主体文本添加到 Internet Service Monitoring 代理程序 配置中的**主体**选项卡，或者使用 Internet Service Monitoring 代理程序 配置或 ismbatch 中的 @Body 组。或者，您可以使用 FORM 参数。不能在 POST 请求中同时使用“主体”文本和 FORM 参数。

使用代理服务器

可以通过代理服务器测试 Web 页面的可用性。在配置监视器以使用代理时，其通过代理发送 HTTP 请求。如有必要，可以绕过代理高速缓存。在**代理详细信息**选项卡上配置代理服务器的参数。HTTP 监视器支持对代理服务器进行认证访问。该认证独立于目标 Web 页面所需的任何认证。

代理服务器元素

在先前版本中，在配置概要文件元素以使用代理服务器时，缺省情况下，HTTP 监视器将代理服务器的名称和端口插入到 $server 和$port 元素，代替确定的目标服务器的名称和端口。要保留先前版本中确定的目标服务器名称和端口的值，请将 generateProxyTokens 属性设置为 1，或者使用 -generateproxytokens 命令行参数启动监视器。

除了在设置此属性或命令行参数时保留 $server 和 $port 元素的值，监视器还生成 $proxyServer、$proxyPort、$proxyAuthType、$proxyUsername 和 $proxyCache 元素。

认证

如果要监视的 Web 页面或者要测试的代理服务器需要认证，那么在“高级”或“代理详细信息”选项卡上的 authenticationtype、username 和 password 参数字段中指定用于访问页面的凭证。

要禁用认证，请将 authenticationtype 设置为 NONE。

要选择基本认证：

1. 请将 authenticationtype 设置为 BASIC。
2. 将 username 和 password 设置为 Web 页面或代理服务器所需要的用户名和密码。

要选择 NTLM：

1. 将 authenticationtype 设置为 NTLMv1 或 NTLMv2。
2. 将 username 和 password 设置为 Web 页面或代理服务器所需要的用户名和密码。

注:

监视器将 HTTP 请求的长度限制为 4096 个字符。如果其他表单数据的长度导致请求长度超过此限制，那么监视器不会在请求中包含其他表单数据。

配置 HTTP 监视器服务测试

使用 HTTP 监视器配置参数可定义 HTTP 服务测试。

表 *54. HTTP* 监视器配置

字段	描述
server	要监视的服务器的主机名。示例为 `www.mycompany.com`
page	要监视的页面的 URL。示例为 `index.html`
描述	提供关于该元素描述性信息的文本字段。示例为 `monitoring via a proxy server`
port	要使用的 HTTP 服务器上的端口。 缺省值：`80`
localip	指定监视器用于测试的网络接口的 IP 地址。如果此字段为空，那么监视器使用 `IpAddress` 属性指定的接口。
version	要使用的 HTTP 协议版本： · `1.0` · `1.1` 缺省值：`1.0`
command	HTTP 请求类型： · `HEAD` · `GET` · `GETALL` · `POST` 缺省值：`GET`
formname	在事务中使用时，HTTP 监视器将扫描指定表单以获取缺省值。事务中的下一个 HTTP 步骤，将自动完成任何值。
authenticationtype	指定用于认证网络用户的挑战/响应认证机制： · `NONE` — 无认证 · `BASIC` · `NTLMv1` - Windows NTLM V1 挑战/响应认证 · `NTLMv2` — Windows NTLM V2 缺省值：`NONE`
username	监视器用于登录到服务器的用户名（帐户名称）。
password	与监视器用于登录到服务器的用户名对应的密码。
超时	等待服务器响应的时间（秒）。 缺省值：`30`

表 *54. HTTP* 监视器配置 *(续)*	
字段	**描述**
poll	轮询的间隔时间（秒）。 缺省值：300 .
failureretests	指示失败之前重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
verifycertificate	缺省情况下禁用。
代理详细信息	
server	代理服务器的主机名。
port	要使用的代理服务器上的端口。 缺省值：8080
authenticationtype	代理服务器的服务器认证类型。有关更多信息，请参阅先前的 authenticationtype。 缺省值：NONE
username	与密码一起供监视器登录代理服务器时使用。
password	与密码一起供监视器登录代理服务器时使用，并且标签为密码。
useproxy	将监视器配置为使用代理服务器执行请求。 · proxy（在 ismbatch 中使用 true） · noproxy（在 ismbatch 中使用 false） 缺省值：noproxy
hostnamelookuppreference	确定应用于所提供主机名的 IP 版本（IPv6 或 IPv4）。选项为： · 缺省设置监视器使用监视器宽度属性设置。这是缺省值。 · 先 4 后 6 先选择 IPv4 再选择 IPv6。IPv4 地址可用时使用 IPv4 地址。如果未找到 IPv4 地址，则使用 IPv6 地址。 · 先 6 后 4 先选择 IPv6 在选择 IPv4。IPv6 地址可用时使用 IPv6 地址。如果未找到 IPv6 地址，则使用 IPv4 地址。 · 仅 4 仅选择 IPv4。只使用 IPv4 地址。如果 IPv4 地址不存在，则轮询返回错误。 · 仅 6 仅选择 IPv6。只使用 IPv6 地址。如果 IPv6 地址不存在，则轮询返回错误。 · 6 或 4 选择 IPv4 或 IPv6。使用主机名返回的首个地址。
nocache	缺省情况下设置为 cache。

正则表达式

您可以通过输入最多 50 个不同的正则表达式来对下载的信息执行正则表达式搜索。HTTP 监视器尝试将检索到的内容与每个正则表达式进行匹配。如果找到指定的正则表达式的匹配，那么将在对应的 $regexpMatchn 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。$regexpStatusn 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。请参阅第 280 页的表 50 以了解有关正则表达式语法的信息。

Head 和 Form 参数

HTTP 监视器可以发送 HTTP 请求的头字段和消息体中的额外数据。

在“参数”选项卡上针对此额外数据配置参数。参数为 Name、Value 和 Type，并且按以下方式运行：

- 类型为 HEAD 的名称/值对指定在监视器发送的所有 HTTP 请求中包含的其他头字段，例如，User-Agent 和 Referer。可以为任何类型的 HTTP 方法（GET、GETALL、HEAD 或 POST）指定头字段。

  对于 ITCAM for Transactions V7.4.0.1 及更高版本，将针对每个新的 HTTP 或 HTTPS 元素添加缺省用户代理头参数 Mozilla/5.0 (ISM-MONITOR)。缺省用户代理头可将 HTTP 和 HTTPS 监视器用于根据浏览器客户端切换内容的 Web 站点。
- 类型为 FORM 的名称/值对指定在监视器发送的 HTTP POST 请求的消息体中包含的额外数据。如果目标页面包含匹配在 formname 字段中指定的名称的表单，那么监视器将如同在概要文件元素中配置的那样处理表单中的任何名称/值对。

注:

监视器将 HTTP 请求的长度限制为 4096 个字符。如果其他表单数据的长度导致请求长度超过此限制，那么监视器不会在请求中包含其他表单数据。

监视器元素

除了对所有元素公共的测试结果，HTTP 监视器生成一组包含特定于 HTTP 服务测试的数据的测试结果。星号 (*) 指示的元素可用作属性。属性名称显示在括号中。缺少星号指示无等效属性。在括号中显示但是无元素的属性指示仅可用作属性，无等效元素。

表 *55. HTTP* 监视器元素

元素	描述
$bytesPerSec*(BytesPerSec)	平均每秒传输的字节数。
$bytesTransfered*(BytesTransferred)	上载或下载的字节数。
$checksum	Checksum 元素通常不提供 对于服务级别分类有意义的值，因为在创建概要文件元素时 校验和值未知（监视器在测试进行中时 计算校验和值）。$checksum 和 $previousChecksum 监视器 元素旨在使用监视器的规则文件来丰富警报。
$command	监视器发出的 HTTP 命令。例如，HEAD、GET、GETALL 或 POST。
$connectTime*(ConnectTime)	连接到服务器所用的时间。
$downloadTime*(DownloadTime)	下载文件所花的时间。
(Elements)	接收到的页面元素的数量。
$formname	在 POST 操作中使用的表单的名称。

表 55. *HTTP* 监视器元素 *(续)*

元素	描述
$lastStatus*(PageStatus)	如果概要文件元素检索到多个页面，那么此元素包含检索到的最后一个页面的结果字符串。该值与 $urlResult*n* 的值相同，其中 *n* 等于 $pageCount 的值。
$lastModified	检索的第一个页面的 Last-Modified HTTP 头字段的值。
$page*(Page)	在 HTTP Server 上访问的页面。
$pageCount	在 GETALL 测试期间下载的资源的总数，测试页面本身除外。如果测试的页面不引用任何其他资源，那么不会生成此元素。
$port*(Port)	用于访问 HTTP 服务器的端口。如果测试使用了代理服务器，那么这是请求提交到的代理服务器上的端口的值。要保留预期目标服务器的端口，请将 generateProxyTokens 属性设置为 1，或者使用 -generateproxytokens 命令行参数来启动监视器
$previousChecksum	PreviousChecksum 元素通常不提供对于服务级别分类有意义的值，因为在创建概要文件元素时校验和值未知（监视器在测试进行中时计算校验和值）。$previousChecksum 和 $checksum monitor 元素旨在使用监视器的规则文件来丰富警报。
$proxyAuthType	代理服务器的服务器认证类型。
$proxyCache	值 true 指示代理服务器已从服务器检索 Web 页面，而不是从其自己的高速缓存。
$proxyPort	向其提交请求的代理服务器的端口号。
$proxyServer	代理服务器的主机名。
$proxyUsername	与密码一起供监视器登录代理服务器时使用。
$regexpMatch*n*	匹配正则表达式的行的内容。
$regexp*n*	正则表达式。
$regexpMatch*n*	匹配正则表达式的行的内容。
$regexpStatus*n*	正则表达式匹配的状态: NONE - 未配置正则表达式检查 MATCHED - 已针对正则表达式找到匹配 FAILED - 未针对正则表达式找到匹配
$responsetime* (ResponseTime)	创建连接后直至收到页面的第一个字节所用的时间。
$timeSinceModification	自上次修改页面以来耗用的时间。这是测试时间与检索到的第一个页面的 Last-Modified HTTP 头字段的值之间的差值。
$urlDownloadTimes*n** (UrlDownloadTime)	GETALL 请求中每个元素的 URL 下载时间。每个元素都具有编号，从 000 开始（$urlDownloadTime000、$urlDownloadTime001、$urlDownloadTime002，如此类推）。
$url*n**(Url)	GETALL 测试中每个页面的 URL。每个页面都具有编号，从 000 开始（$url000、$url001、$url002，如此类推）。

表 55. *HTTP* 监视器元素 *(续)*	
元素	**描述**
$urlResult*n** (UrlResultString)	GETALL 请求中下载的每个页面的结果字符串。每个结果都具有编号，从 000 开始（$urlResult000、$urlResult001、$urlResult002，如此类推）。
$username	用于访问要求认证用户的页面的名称。

状态消息

在使用 IBM Application Performance Management 时，HTTP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

表 56. 表 4. *HTTP* 监视器状态消息	
Message	**描述**
OK	监视器请求已成功。
Form not found	无法找到请求的页面。
Initialise Fetch Page Failed	没有足够的内存为 HTTP 页面访存机制分配空间。前一行消息应包含更多信息。
Connection Failed	由于链接断开、链接重置、链接不可访问、链接超时、链接终止或主机停机之外的原因，监视器无法连接。请参阅 HTTP 监视器日志文件以获取更多信息。

表单参数和正则表达式匹配

发送表单参数为 search=ism 的 HTTP POST 请求，以监视表单 http://support.mycompany.com/cgi-bin/search.cgi 的操作，并使用正则表达式与响应中的字符串 Your search was successful 进行匹配。如果响应中返回了该字符串，请将服务级别分类为良好，否则请分类为失败。

创建新的 HTTP 概要文件元素并设置设置，如下表中所示。

表 57. *HTTP* 表单概要文件元素示例	
概要文件元素配置字段	**值**
server	support.mycompany.com
page	/cgi-bin/search.cgi
描述	例如 - 表单参数和正则表达式
正则表达式详细信息	
match 1	Your search was successful
服务级别分类详细信息	
statement	Regexp Status 1 = MATCHED then status GOOD
Head 和 Form 详细信息	
命名	search
value	ism
type	FORM

属性

下表中描述了特定于 HTTP 监视器的属性和命令行选项。

表 58. *HTTP* 监视器属性

属性名称	属性参数	描述
AllowDuplicateDownload	0|1	每次找到页面时强制下载页面。 0 - 已禁用（仅下载一次） 1 - 已启用
ForceHTMLParse	0|1	强制将没有内容类型 text/html 的页面解析为 HTML。 0 - 已禁用 1 - 已启用
GenerateProxyTokens	0|1	指定监视器是否要在测试中使用代理服务器时生成包含有关代理服务器信息的其他元素。 0 - 已禁用 1 - 已启用（其他元素 $server 和 $port 包含代理服务器的值）
GETALLThreadNum	1|2|3|4|5	指定在 GETALL 请求期间要使用的单独线程数。
GetLinkTags	0|1	为 GETALL 请求激活已链接的样式表的下载： 0 - 已禁用 1 - 已启用（如果目标页面包含带有属性值 rel=stylesheet 的 link 标记，那么监视器将尝试下载与 link 标记的 href 属性关联的资源）
HostnameLookupPreference	字符串	确定应用于所提供主机名的 IP 版本（IPv6 或 IPv4）。可能的值为： · 先 4 后 6 先选择 IPv4 再选择 IPv6。IPv4 地址可用时使用 IPv4 地址。如果未找到 IPv4 地址，则使用 IPv6 地址。 · 先 6 后 4 先选择 IPv6 在选择 IPv4。IPv6 地址可用时使用 IPv6 地址。如果未找到 IPv6 地址，则使用 IPv4 地址。 · 仅 4 仅选择 IPv4。只使用 IPv4 地址。如果 IPv4 地址不存在，则轮询返回错误。 · 仅 6 仅选择 IPv6。只使用 IPv6 地址。如果 IPv6 地址不存在，则轮询返回错误。 · 6 或 4 选择 IPv4 或 IPv6。使用主机名返回的首个地址。 缺省值：4Then6
Ipv6Address	整数	使用 HTTP IPv6 时要作为 HTTP 请求的源进行绑定的本地地址。 缺省值：无地址
NoParseExtensions	字符串	指示监视器不应解析而是仅下载的文件类型的文件扩展名的逗号分隔列表。

表 *58. HTTP* 监视器属性 *(续)*		
属性名称	**属性参数**	**描述**
OutputDirectory	字符串	如果 OutputResult 为 true（设置为 1），那么请指定要使用的输出目录。 缺省值：$ISHOME/var
OutputResult	0|1	指定监视器是否保存从服务接收到的数据。 0 - 已禁用 1 - 已启用
RelativeRedirects	0|1	允许 HTTP 301 和 HTTP 302 状态码中的 Location 字段包含相对 URL，代替绝对 URL。 0 - 绝对 URL 1 - 相对 URL
RFCPOST	0|1	指定监视器应遵循 RFC1945 和 RFC2616 的规定，并在重定向后再次发送 POST。大量 Web 服务器在重定向后不期望 POST，并且大多数浏览器不遵循 RFC。 0 - 已禁用 1 - 已启用

***HTTPS* 监视器**

HTTPS 监视器检查 Web 服务器的可用性和响应时间。其可以监视单个 Web 页面，包括通常需要用户在字段中输入数据的使用 HTML 表单的页面。

注: HTTPS 监视器的工作方式与 HTTP 监视器相同，但是其使用 SSL（安全套接字层）V2 或 V3 协议与 HTTP Server 进行通信，这将加密服务器与监视器之间的所有通信。

表 *59. HTTPS* 监视器文件摘要	
监视器文件。	**名称或位置**
监视器可执行文件	nco_m_https
属性文件	$ISHOME/etc/props/https.props
规则文件	$ISHOME/etc/rules/https.rules
日志文件	$ISHOME/log/https.log

用于配置 HTTPS 监视器的准则

HTTPS 监视器检查 Web 服务器的可用性和响应时间。请在以下情境中使用 HTTPS 监视器:

- 目标 Web 站点是静态的。

  对于动态 Web 站点，请使用 TRANSX 监视器。
- 基于 HTTPS 协议为目标 Web 站点提供服务。

  对于通过 HTTP 协议交付内容的 Web 站点，选择 HTTP 监视器。
- 要执行跨多个平台的监视。
- 其中速度是决定因子（HTTPS 监视器提供高性能）。

客户机端证书

使用该监视器可监视需要客户机端证书以进行相互认证的服务器。

在创建概要文件元素时，指定 SSL 证书文件、密钥文件和密钥密码。

证书必须为隐私增强邮件（PEM）格式。如果证书为其他格式，那么必须将其转换为 PEM 格式。可使用软件转换证书，例如，openSSL，可从 http://www.openssl.org 获取此软件。

注: 如果始终在所有概要文件元素中使用相同的证书、密钥和密码，那么使用监视器属性指定它们，代替在每个创建的概要文件元素中进行定义。

配置 HTTPS 监视器服务测试

使用 HTTPS 监视器配置参数定义 HTTPS 服务测试。

表 *60. HTTPS* 监视器配置

字段	描述
服务器	要监视的服务器的主机名。示例为 `www.myconpany.com`
page	要监视的页面的 URL。示例为 `/secure/`
描述	提供关于该元素描述性信息的文本字段。
端口	要使用的服务器上的端口。 缺省值：`443`
localip	指定监视器用于测试的网络接口的 IP 地址。如果此字段为空，那么监视器使用 `IpAddress` 属性指定的接口。
版本	要使用的 HTTPS 协议版本： · `1.0` · `1.1` 缺省值：`1.0`
command	请求类型： · `HEAD` · `GET` · `GETALL` · `POST` 缺省值：`GET`
formname	在事务中使用时，HTTPS 监视器将扫描指定表单以获取缺省值。事务中的下一个 HTTPS 步骤，将自动填写找到的任何值。
authenticationtype	指定用于认证网络用户的挑战/响应认证机制： · `NONE` - 无认证。 · `BASIC` · `NTLMv1` - Windows NTLM V1 挑战/响应认证。 · `NTLMv2` - Windows NTLM V2。 缺省值：`NONE`
username	监视器用于登录到 HTTPS 服务器的用户名（帐户名称）。
密码	与监视器用于登录到 HTTPS 服务器的用户名相对应的密码。

表 60. *HTTPS* 监视器配置 *(续)*	
字段	**描述**
sslcertificatefile	监视器元素中使用的数字证书文件的路径和文件名。如果路径不是绝对路径，那么监视器将其解释为相对于工作目录 (`$ISMHOME/platform/arch/bin`)。 如果未指定证书文件，那么监视器使用监视器属性 `SSLCertificateFile` 指定的证书。
sslkeyfile	包含 SSL 专用密钥的文件的路径和文件名，用于标识服务器和对 SSL 消息签名。
sslkeypassword	用于对 SSL 专用密钥进行加密的密码。
超时	等待服务器响应的时间（秒）。 缺省值：`30`
poll	轮询的间隔时间（秒）。 缺省值：`300`
failureretests	指示失败之前重新测试的次数。 缺省值：`0`
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：`10`
代理详细信息	
服务器	代理服务器的主机名。
端口	要使用的代理服务器上的端口。
authenticationtype	代理 HTTPS 服务器的服务器认证类型。请参阅 authenticationtype 以获取更多信息。
username	监视器用于登录到代理 HTTPS 服务器的用户名。
密码	监视器用于登录到代理 HTTPS 服务器的密码。
useproxy	将监视器配置为使用代理服务器执行请求。 · `proxy`（在 ismbatch 中使用 `true`） · `noproxy`（在 ismbatch 中使用 `false`） 缺省值为 `noproxy`

表 60. HTTPS 监视器配置 (续)	
字段	**描述**
hostnamelookuppreference	确定应用于所提供主机名的 IP 版本（IPv6 或 IPv4）。选项为: · 缺省设置监视器使用监视器宽度属性设置。这是缺省值。 · 先 4 后 6 先选择 IPv4 再选择 IPv6。IPv4 地址可用时使用 IPv4 地址。如果未找到 IPv4 地址，则使用 IPv6 地址。 · 先 6 后 4 先选择 IPv6 在选择 IPv4。IPv6 地址可用时使用 IPv6 地址。如果未找到 IPv6 地址，则使用 IPv4 地址。 · 仅 4 仅选择 IPv4。只使用 IPv4 地址。如果 IPv4 地址不存在，则轮询返回错误。 · 仅 6 仅选择 IPv6。只使用 IPv6 地址。如果 IPv6 地址不存在，则轮询返回错误。 · 6 或 4 选择 IPv4 或 IPv6。使用主机名返回的首个地址。

正则表达式匹配

您可以通过输入最多 50 个不同的正则表达式来对下载的信息执行正则表达式搜索。HTTPS 监视器尝试将检索到的内容与每个正则表达式进行匹配。

如果找到指定的正则表达式的匹配，那么将在对应的 `$regexpMatchn` 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。`$regexpStatusn` 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

有关更多信息，请参阅第 280 页的表 50。

Head 和 Form 参数

与 HTTP 监视器类似，HTTPS 监视器可在 HTTP 请求的头字段和消息体中发送额外数据。

有关 head 和 form 参数的详细信息，请参阅 HTTP Head 和 Form 参数。

监视器元素

表 61. HTTPS SSL 监视器元素	
元素	**描述**
$SSLcertificateSerialNumber	服务器提供的 X509 证书的序列号。
$SSLcipherSuiteCount	可用于连接的密码套件的数量。
$SSLcipherSuiteList	可用于连接的密码套件的列表。
$SSLcipherSuiteName	针对连接选择的密码套件。
$SSLeffectiveSessionKeyBits	会话密钥中的位数。针对导出版本，这通常是 128、168 或 40。
$SSLHandshakeTime* (SslHandshakeTime)	建立 SSL 连接所花的时间。
$SSLissuerName	服务器的 X509 格式证书的颁发者名称。
$SSLprotocolVersion	使用的 SSL 的版本，即 v2 或 v3。
$SSLpublicKeyLengthBits	服务器公用密钥的大小。通常为 1024 位，使用导出规范密码套件时除外。
$SSLserverCertificateValidFrom	服务器证书的生效日期。

表 61. *HTTPS SSL* 监视器元素 *(续)*	
元素	**描述**
$SSLserverCertificateValidTo	服务器证书的有效结束日期。
$SSLserverName	SSL 服务器名称。
$SSLsubjectName	X509 格式证书的主题名称。这通常是控制服务器的组织的名称。

星号 (*) 指示的元素可用作属性。属性名称显示在括号中。缺少星号指示无等效属性。在括号中显示而无元素的属性指示仅可用作属性，无等效元素。

HTTPS 监视器生成与 HTTP 监视器相同的额外元素，如第 286 页的表 55 中所述。此外，如果在测试中使用客户机端证书，那么其将生成与 SSL 相关的元素，如第 293 页的表 61 中所述。

除了对所有元素公共的测试结果，HTTPS 监视器生成一组包含特定于 HTTPS 服务测试的数据的测试结果。

状态消息

在使用 IBM Application Performance Management 时，HTTPS 监视器在 `ResultMessage` 属性中提供状态消息。这些消息指示测试的结果。

除了 HTTP 状态消息，HTTPS 监视器还生成第 294 页的表 62 中列出的消息。

表 62. *HTTPS* 监视器状态消息	
Message	**描述**
OK	监视器已成功连接到服务器。
SSL handshake failed	在建立到服务器的连接后，监视器无法初始化 SSL 连接。
Connection failed	由于链接断开、链接重置、链接不可访问、链接超时、链接终止或主机停机之外的原因，监视器无法连接。请参阅 HTTP 监视器日志文件以获取更多信息。

属性

HTTPS 监视器具有与 HTTP 监视器相同的属性。

有关与 HTTP 监视器相同的属性选项的详细信息，请参阅第 289 页的表 58。表 5 列出更多特定于 HTTPS 的属性。

表 63. 特定于 *HTTPS* 监视器的属性		
属性名称	**属性参数**	**描述**
SSLCertificate File	字符串	在创建期间未针对 HTTPS 元素明确指定证书时使用的数字证书的路径和文件名。 如果路径不是绝对路径，那么监视器将其解释为相对于工作目录 (`$ISHOME/platform/arch/bin`）。
SSLCipherSuite	字符串	用于 SSL 操作的密码套件。 缺省值：`RC4:3DES:DES:+EXP`
SSLDisableTLS	整数	针对遗留项支持禁用 TLSv1。 缺省值：`0` - 启用 TLSv1。 `1` - 禁用 TLSv1。
SSLKeyFile	字符串	包含 SSL 专用密钥的文件。

表 63. 特定于 *HTTPS* 监视器的属性 *(续)*		
属性名称	**属性参数**	**描述**
SSLKeyPassword	字符串	用于对 SSL 专用密钥进行加密的密码。

密码套件

SSLCipherSuite 属性指定 HTTPS 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

ICMP 监视器

ICMP 监视器测试在网络上运行的因特网控制报文协议服务的性能。要执行此操作，监视器使用 ICMP 回传命令。

下表列出 ICMP 监视器文件。

表 64. *ICMP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_icmp
属性文件	$ISHOME/etc/props/icmp.props
规则文件	$ISHOME/etc/rules/icmp.rules
日志文件	$ISHOME/log/icmp.log

用于配置 ICMP 监视器的准则

ICMP 监视器向目标主机发出 ICMP 回传请求（通常被称为 pings），并等待回传应答响应。其记录查找时间、往返时间以及成功率度量，这些提供网络执行情况的指示。监视器发出回传请求时，请求可能先通过一个或多个路由器，然后才到达目标主机。在目标主机接收到回传请求前，这些路由器可以先响应监视器。如果监视器发出的回传请求通过路由器，那么该路由器可以向监视器发送回复。此回复可能指示路由器找不到目标主机，或者路由器过于繁忙而无法处理请求。可能是监视器在收到来自目标主机的回传应答前收到来自多个路由器的应答。如果监视器成功接收到来自目标主机的回传应答，它将记录所用的时间。如果监视器未在指定的时间段内收到来自目标服务器的应答，那么将请求记录为失败。可以将监视器配置为在每个测试中将多个 ICMP 回传请求发送至同一目标。监视器将记录每个已发送请求的统计信息。

注: 以 root 身份运行 ICMP 监视器，因为其打开原始套接字以发送 ICMP 包。

配置 ICMP 监视器服务测试

使用 ICMP 监视器配置参数可定义服务测试。配置监视器时，对于超时和轮询时间间隔参数，将显示缺省值。缺省值分别为 30 和 300 秒。在配置期间不显示表中列出的其他缺省值，而是在保存配置详细信息时应用（如果未指定任何值）。

下表列出 ICMP 监视器配置。

表 65. *ICMP* 监视器配置	
字段	**描述**
服务器	要将回传请求发送到的服务器的主机名或 IP 地址。示例为 test.myconpany.com
描述	提供关于该元素描述性信息的文本字段。
超时	等待服务器响应每个回传请求的时间（秒）。 缺省值：10

表 65. *ICMP* 监视器配置 *(续)*

字段	描述
numberofpings	要发送的回传请求数。 缺省值：5
packetinterval	两次发送回传请求之间要等待的时间（秒）。 缺省值：1
packetsize	发送的每个回传请求的大小（字节）。 缺省值：64
typeofservice	在 IP 层中设置“服务类型”字段。IPv4 式的“服务类型”（TOS）值和“DSCP 差别服务”字段值均可输入其中。有效值范围从 0 到 255。
retries	监视器在退出前应该重新尝试发出各个回传请求的次数。 缺省值：0
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	指示失败之前重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
hostnamelookuppreference	确定应用于所提供主机名的 IP 版本（IPv6 或 IPv4）。选项为： · 缺省设置监视器使用监视器宽度属性设置。这是缺省值。 · 先 4 后 6 先选择 IPv4 再选择 IPv6。IPv4 地址可用时使用 IPv4 地址。如果未找到 IPv4 地址，则使用 IPv6 地址。 · 先 6 后 4 先选择 IPv6 在选择 IPv4。IPv6 地址可用时使用 IPv6 地址。如果未找到 IPv6 地址，则使用 IPv4 地址。 · 仅 4 仅选择 IPv4。只使用 IPv4 地址。如果 IPv4 地址不存在，则轮询返回错误。 · 仅 6 仅选择 IPv6。只使用 IPv6 地址。如果 IPv6 地址不存在，则轮询返回错误。 · 6 或 4 选择 IPv4 或 IPv6。使用主机名返回的首个地址。

注: 通过按 10 分钟时间间隔检查响应时间来监视主机 `test.mycompany.com` 的可用性。尝试在 30 秒内连接到服务器，如果超时，请再重试两次。如果仍然失败，那么按 5 秒重试间隔重复测试 3 次。

监视器元素

除了对所有元素公共的测试结果，ICMP 监视器生成一组包含特定于 ICMP 服务测试的数据的测试结果。

下表描述 ICMP 监视器的其他元素。

星号 (*) 指示的元素可用作属性。属性名称显示在元素下面的括号中。缺少星号指示无等效属性。在括号中显示而无元素的属性指示仅可用作属性，无等效元素。

表 66. *ICMP* 监视器元素	
元素	**描述**
$averageRTT*(Average RTT)	平均往返时间，以秒为单位。
$endTime	收到响应的 UNIX 时间。
$jitter	两个相邻 ICMP 回传请求的到达时间之差的绝对值减去其偏度值时间。根据 RFC2598 中指定的公式计算该值。仅在回传请求的数量大于 1 时才创建此元素。如果使用两个以上的回传请求，那么值是所有回传请求对之间的平均抖动。
$lookupTime*(LookupTime)	获取主机服务器 IP 地址所花的时间。
$maxRTT*(MaxRTT)	最长往返时间，以秒为单位。
$minRTT* (MinRTT)	最短往返时间，以秒为单位。
$numberPackets	发送的 ICMP 回传请求的数量，如在概要文件元素中指定。
$packetInterval	发送每个 ICMP 回传请求之间的时间，如概要文件元素中指定。
$packetRetries	监视器在退出前尝试重新发送 ICMP 回传请求的次数。
$packetSize	每个 ICMP 回传请求的大小（以字节为单位），如概要文件元素中指定。
$pingAttempts Failed	针对第一个不成功的 ICMP 回传请求尝试的次数。
$pingAttempts Responded	针对第一个成功的 ICMP 回传请求尝试的次数。
$pingMessageFailed	针对第一个不成功的 ICMP 回传请求返回的消息。
$pingMessage Responded	针对第一个成功的 ICMP 回传请求返回的消息。
$pingReceivedTime Failed	收到第一个不成功回传响应的 UNIX 时间。
$pingReceivedTime Responded	收到第一个成功回传响应的 UNIX 时间。
$pingRespondIP Failed	响应第一个不成功的 ICMP 回传请求的 IP 地址。
$pingRespondIP Responded	响应第一个成功的 ICMP 回传请求的 IP 地址。
$pingRTTFailed	第一个不成功的 ICMP 回传请求的往返时间，以秒为单位。
$pingRTTResponded	第一个成功的 ICMP 回传请求的往返时间，以秒为单位。
$pingSentTime Failed	发送第一个不成功的 ICMP 回传请求的 UNIX 时间。
$pingSentTime Responded	发送第一个成功的 ICMP 回传请求的 UNIX 时间。
$pingsFailed	无回传回送信号的已发送的 ICMP 回传请求的数量。
$pingsResponded	收到的有效回传响应的数量。
$pingTime	在发送 ICMP 回传请求后接收回传响应所用的时间。

表 66. *ICMP* 监视器元素 *(续)*	
元素	**描述**
$respondPercent* (RespondPercent)	有响应的已发送的 ICMP 回传请求的百分比。
$responseTime	目标主机响应 ICMP 回传请求所用的时间。
$sentTime	发送 ICMP 回传请求的 UNIX 时间。
$spreadRTT	$maxRTT 和 $minRTT 之差。
$startTime	测试开始的 UNIX 时间。
$totalHostTime	在开始测试后接收回传响应所用的时间。
$typeOfService	IP 层中的“服务类型”字段，在添加新的 ICMP 元素时指定。有关详细信息，请参阅第 295 页的『ICMP 监视器』。

ICMP 监视器创建单独一组 $pingname 元素以记录测试期间发送的每个 ICMP 回传请求的结果。已发送的请求数量由 $numberPackets 指示。例如，对于 $pingRTT 元素，如果 $numberPackets 是 3，那么监视器创建三个元素（$pingRTT1、$pingRTT2 和 $pingRTT3），其中包含已发送的三个 ICMP 回传请求的往返时间度量。

状态消息

在使用 IBM Application Performance Management 时，ICMP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 ICMP 状态消息。

表 67. *ICMP* 监视器状态消息	
Message	**描述**
Pings Complete	ICMP 回传请求成功。
ICMP echo failed	监视器无法发出 ICMP 回传请求，因为监视器主机或者其与网络的连接存在问题。
Timed out	ICMP 回传请求超时。
不可达	从路由器返回此消息，并且不一定准确。
Source quench	路由器过于繁忙而无法处理 ICMP 回传请求。
Time exceeded	从路由器返回此消息。指示已围绕着网络转发 ICMP 回传请求太多次。
Parameter problem	从路由器返回此消息。指示路由器无法处理 ICMP 回传请求。这可能是因为消息已损坏。

属性

以下描述特定于 ICMP 监视器的属性。

表 68. *ICMP* 属性		
属性名称	**属性参数**	**描述**
EventsPerSec	不适用	不支持此属性。

表 68. *ICMP* 属性 *(续)*		
属性名称	**属性参数**	**描述**
IntraPingWait	整数	ICMP 监视器发送的所有 ping 之间的最短时间间隔，以毫秒为单位。用于调整系统以使网络流量分布在较长时间段上。例如，在具有数千个目标 ICMP 主机的环境中，将 IntraPingWait 设置为 3。 缺省值：0
Ipv6Address	整数	在使用 ICMP IPv6 时，作为 ICMP 回传请求的源绑定到的本地地址。 缺省值：无地址
MaxDNSResolvingThreads	整数	DNS 解析器使用的最大线程数。 缺省值：20
MaxPacketSize	整数	最大 ICMP 包大小，以字节为单位。
PingsPerSec	整数	监视器每秒尝试发送的回传请求数。发送的实际请求数量取决于 CPU 和网络负载。 缺省值：100
SocketBufferSize	整数	接收缓冲区千字节的大小（以千字节为单位）。 缺省值：32

LDAP 监视器

LDAP 监视器测试轻量级目录访问协议 (LDAP) 服务器的操作。

下表列出 LDAP 监视器文件。

表 69. *LDAP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_ldap
属性文件	$ISHOME/etc/props/ldap.props
规则文件	$ISHOME/etc/rules/ldap.rules
日志文件	$ISHOME/log/ldap.log

用于配置 LDAP 监视器的准则

LDAP 监视器通过连接到 LDAP 服务器并尝试查找特定条目来测试 LDAP 服务。如果服务器查找条目成功，那么它将向监视器返回该条目的内容。LDAP 监视器可以使用 SSL 进行认证并连接到 LDAP 服务器。

要配置 LDAP 监视器，需要了解 LDAP 协议和受监视目录服务的工作方式。LDAP 是用于访问和管理目录服务的因特网协议。目录服务是分布式数据库应用程序。目录包含多个条目。例如，目录可能包含与组织的员工或资源相关的条目。每个条目包含属性集，例如，员工目录中的条目可能包含员工的姓名、电话号码和地址。

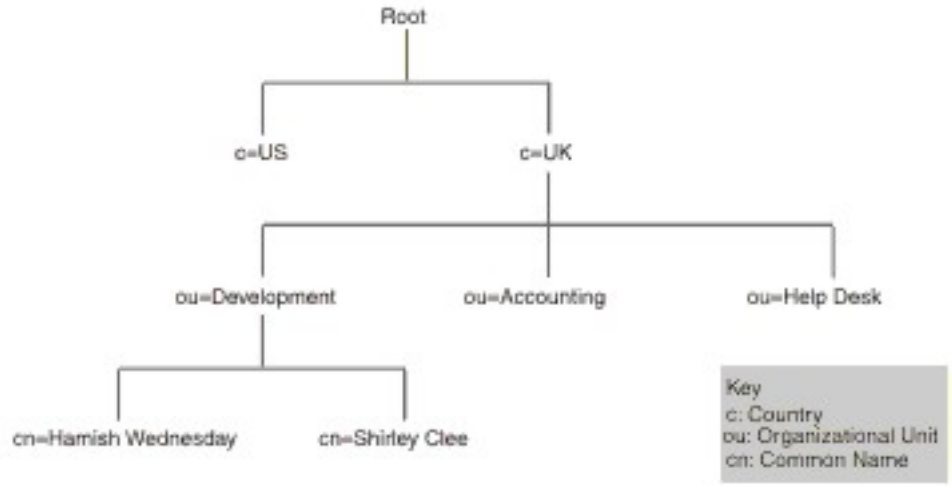
单独目录服务可能以不同方式进行构造，因此监视过程也可能不同。

LDAP 版本

LDAP 监视器同时支持 LDAP V2 和 V3。缺省情况下，监视器尝试连接到使用 V3 的目标 LDAP 服务器，然后在尝试失败的情况下自动回退到 V2。您可以通过设置 **NOLDAPV3** 属性来强制监视器始终使用 V2。

目录服务示例

此目录服务示例存储所有员工的个人详细信息。目录分为各个国家或地区，然后分为各个部门。员工及其属性存储在每个部门下。



目录层次结构示例图像显示从示例目录中抽取的内容。此图显示目录结构。顶点级别是根目录。两个子目录表示国家或地区，分别标记为 UK 和 US。UK 子目录进一步划分为表示组织单元的三个其他子目录。它们标记为 Development、Accounting 和 Help Desk。在 Development 组织单元中，有两个子目录，分别表示公共名称 Shirley Clee 和 Hamish Wednesday。

实体由专有名称引用。专有名称是实体的路由。例如，accounting 部门和 Hamish Wednesday 的专有名称将是：

```
dn="ou=accounting, c=UK"
dn="cn=Hamish Wednesday, ou=Development, c=UK"
```

每个员工的条目具有多个属性。例如，Hamish Wednesday 的条目包含以下详细信息。

```
cn: Hamish Wednesday
uid: ham
mail: HWednesday@development.mycompany.com
telephoneNumber: 88 88 55 44
```

目录层次结构中的每个实体可以通过用户名（在 LDAP 中是指专有名称）和密码来受保护。监视器使用此用户名和密码访问 LDAP 服务器。

当监视器访问服务器时，它会指示目录层次结构中开始进行目标实体搜索的位置。这在 searchBase 字段中指定为专有名称。例如，搜索可以在部门级别开始：

```
ou=Accounting, c=UK
```

注: 组成专有名称的实体顺序相反。即，它们从层次结构中的最低点开始，然后列出每个先前实体。

目标实体将传递到过滤器字段中的服务器。此字段包含目标实体的属性。例如，要搜索 Hamish Wednesday's 实体，过滤器字段可能包含：

```
(uid=ham)
```

LDAP 服务器使用监视器提供的字段来搜索目标实体。搜索结果将返回到监视器。

如果搜索成功，那么服务器还将返回目标实体的属性。监视器将其转换为动态创建名称的元素。例如，监视器会将 Hamish Wednesday 的条目转换为：

```
$dnMatched = "cn=Hamish Wednesday, ou=Development, c=UK"
$cn = "Hamish Wednesday"
$uid = "ham"
$mail = "HWednesday@development.mycompany.com"
$telephoneNumber = "88 88 55 44"
```

LDAP 认证

SSL LDAP 服务器认证依赖于认证中心签署的公用-专用密钥证书，例如 Verisign 和 Thawte。对于 SSL 认证，LDAP 监视器使用公用证书的 Netscape cert7db 数据库来验证认证中心签发的 LDAP 服务器证书签名。

如果您使用的是由 Netscape 所识别的认证中心（例如 Verisign 或 Thawte）签名的证书，那么 LDAP 监视器会自动识别这些证书。如果您使用的是由您的组织或 Netscape 数据库中未包含的组织签名的证书，那么必须将其添加到 cert7db 数据库。

使用可从 Netscape 获取的 certutil 实用程序将证书添加到数据库。LDAP 监视器的 cert7db 数据库位于文件 `$ISHOME/certificates/cert7.db` 中。

要监视通过 SSL 或 TLS 加密保护的 LDAP 服务器，请按照下表中所述设置环境变量：

表 *70.* 监视安全 *LDAP* 服务器所需的环境变量

变量	描述	设置
LDAPTLS_CACERT	指定包含 CA 证书的文件	包含服务器证书的文件。 例如，`cacert.pem`。
LDAPTLS_REQCERT	指定要对服务器证书执行的检查	从 `never|allow|try|demand` 中进行选择。

有关更多信息，请参阅 http://www.openldap.org。

属性

下表中描述了特定于 LDAP 监视器的属性：

表 *71. LDAP* 监视器属性选项

属性名称	属性参数	描述
NOLDAPV3	0|1	强制监视器使用 LDAP V2 而不是 LDAP V3。 0 - 使用 LDAP V3 1 - 使用 LDAP V2

密码套件

`SSLCipherSuite` 属性指定 LDAP 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

配置 LDAP 监视器服务测试

使用 LDAP 监视器配置参数来定义服务测试。

配置监视器时，对于超时参数显示的缺省值为 30 秒，对于轮询时间间隔参数显示的缺省值为 300 秒。在配置期间不会显示表中列出的其他缺省值，但如果未指定值，那么将在保存配置详细信息时应用这些缺省值。

下表描述 LDAP 监视器配置：

表 *72. LDAP* 监视器配置

字段	描述
server	要监视的 LDAP 服务器的名称或 IP 地址。例如，`ldap.mycompany.in`。
searchbase	搜索起始位置的专有名称。例如，`ou=Accounting, c=UK`。
过滤器	要搜索的目标实体的属性。例如，`(uid=ham)`。
描述	提供关于该元素描述性信息的文本字段。例如，`LDAP monitor`。

表 72. *LDAP* 监视器配置 *(续)*	
字段	**描述**
动态	选择在创建概要文件元素后是否要将其激活。例如，`Selected`。
port	要连接到的 LDAP 服务器上的端口。如果使用的是 SSL 认证，那么必须指定 SSL 端口。 缺省值：389
username	用于登录到目录服务的用户名。用户名的格式取决于认证类型的设置。 您可以指定 Windows 域，即 `DOMAIN\username`。例如，`jbloggs`。
密码	用于登录到目录服务的密码（如有必要）。例如，`secret9`。
authenticationtype	要使用的 LDAP 认证方法： · SIMPLE（匿名或明文密码） · SSL-SIMPLE · SASL-DIGEST-MD5 **注:** SASL-DIGEST-MD5 认证在 Linux 操作系统上不可用。 如果将 authenticationtype 设置为 SIMPLE 或 SSL-SIMPLE，请输入专有名称格式的用户名。如果将 authenticationtype 设置为 SASL-DIGEST-MD5，请将用户名输入为 SASL 绑定标识。要以匿名用户身份登录到 LDAP 服务器，请将 authenticationtype 设置为 SIMPLE，并将用户名和密码字段保留为空白。 缺省值：SIMPLE
saslrealm	LDAP 服务器的认证域；通常是服务器的标准域名。如果要在多个系统之间共享密码，那么可能会使用域名。例如，`mycompany.com`。
超时	等待服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

服务级别分类

LDAP 监视器的可用服务级别分类选项包括：

```
totalTime
connectTime
searchTime
initTime
dnMatched
message
```

在服务级别分类中：

· 通过手动输入监视器元素的名称来指定额外的服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。

- message 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。

监视器元素

除所有元素共同的测试结果以外，LDAP 监视器还会生成一组包含特定于 LDAP 服务测试的数据的测试结果。

下表列出 LDAP 监视器的其他元素。

表 *73. LDAP* 监视器元素

元素	**描述**
$authentication	LDAP 服务器需要的用户认证方法的类型（Standard 或 CRAM-MD5）。
$connectTime* (ConnectTime)	连接到 LDAP 服务器所需的时间。
$distinguishedName* (UserName)	用于登录到目录服务的专有名称。
$dnMatched	搜索中匹配的实体。
$filter* (SrchFilter)	用于查找目标实体的属性。
$initTime* (InitTime)	初始化 LDAP 客户机所需的时间。
$port* (Port)	监视器连接到的 LDAP 服务器上的端口。
$saslRealm	添加新 LDAP 元素后指定的 SASL 域。
$searchBase* (SearchBase)	从中启动搜索的实体的专有名称。
$searchTime* (SearchTime)	完成搜索所需的时间。

状态消息

使用 IBM Application Performance Management 时，LDAP 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

下表描述 LDAP 状态消息。

表 *74. LDAP* 监视器状态消息

消息	**描述**
`Search successful`	请求已成功。
`Search failed`	请求失败

表 74. LDAP 监视器状态消息 (续)	
消息	**描述**
No match	服务器可能在搜索条件中找不到匹配条目。
Connection timed out	连接成功，但是之后服务器停止响应。
Initialisation failed - an unrecognised authentication type was specified	如果使用 LDAP 监视器不支持的认证类型，那么会发生此情况。
Client initialisation failed	由于内存不足，LDAP 结构的初始化失败。
Bind (authentication) failed	正在等待绑定完成的服务器超时。
SASL bind is not possible because server does not support LDAPv3	服务器必须支持 LDAPv3 才能创建 SASL 绑定。
SASL bind is not possible because one of 'bind_id' (username), password or sasl_realm is blank	为了进行绑定，所有认证字段都必须具有值。因此，如果用户使用 SIMPLE 认证类型以匿名（明文）方式登录，那么无法进行 SASL 绑定。
SASL bind error	无法识别 SASL 绑定失败的原因。
SASL bind authorisation error	由于授权凭证不正确，SASL 绑定失败。

IMAP4 监视器

IMAP4 监视器与 SMTP 监视器一起测试 IMAP4 电子邮件服务的可用性和响应时间。

下表列出 IMAP4 监视器文件。

表 75. IMAP4 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_imap4
属性文件	$ISHOME/etc/props/imap4.props
规则文件	$ISHOME/etc/rules/imap4.rules
日志文件	$ISHOME/log/imap4.log

用于配置 IMAP4 监视器的准则

IMAP4 监视器与 SMTP 监视器配合使用，监视 SMTP 监视器向其发送测试消息的邮箱并度量交付这些消息所用的时间量。

注: 确保同步监视器主计算机和邮件服务器上的系统时钟以使交付时间计算正常工作。

IMAP4 监视器读取邮箱内容后，它将生成以下两种不同类型的事件：

- 特定于消息的事件

  IMAP4 监视器为从邮箱下载的每条电子邮件消息创建一个特定于消息的事件。在这类事件中，监视器将 $message 元素设置为 Message Successfully Downloaded。计算出的 $timeToDeliver 元素值是 SMTP 监视器发出消息后到邮箱接收到消息之间的消息传递时间。$hopCount 元素指示在消息到达邮箱前作为消息中继的主机数。

- 概要事件

处理完邮箱中的所有消息后，监视器将创建一个概要事件。在这类事件中，$message 元素指示从邮箱中成功下载的消息总数，$totaltime 元素指示完成这些请求所用的时间。$totaltime 以秒为单位。

安全邮件

IMAP4 监视器支持连接到安全邮件服务。可以使用 SSL/TLS 或 STARTTLS 命令进行连接。在定义概要文件元素时，使用 securitytype 字段以选择安全安全性。如果邮件服务器需要客户机端证书以用于 SSL 加密，请使用 SSL 属性以指定证书文件、密钥文件、密钥密码和密码套件。

客户机端证书

使用 IMAP4 监视器可监视需要客户机端证书以进行相互认证的服务器。

在创建概要文件元素时，指定 SSL 证书文件、密钥文件和密钥密码。

证书必须为隐私增强邮件（PEM）格式。如果证书为其他格式，那么必须将其转换为 PEM 格式。可使用软件转换证书，例如，openSSL，可从 http://www.openssl.org 获取此软件。

注: 如果始终在所有概要文件元素中使用相同的证书、密钥和密码，那么使用监视器属性指定它们，代替在每个创建的概要文件元素中进行定义。

邮箱

在 IMAP4 监视器处理 SMTP 监视器发送的电子邮件消息中所包含的信息后，其从邮箱中删除消息。您可以使用任何现有邮箱来存储两个监视器之间的电子邮件消息，即使邮箱属于实际用户。但是，建议针对服务测试创建专门的邮箱帐户。

配置 IMAP4 监视器服务测试

使用 IMAP4 监视器配置参数可定义服务测试。

配置监视器时，对于超时和轮询时间间隔参数，将显示缺省值。缺省值分别为 30 和 300 秒。在配置期间不显示表中列出的其他缺省值，而是在保存配置详细信息时应用（如果未指定任何值）。

表 *76. IMAP4* 监视器配置

字段	描述
服务器	邮件服务器的 IP 地址。示例为 test.mycompany.com
描述	提供关于该元素描述性信息的文本字段。
端口	IMAP4 服务器的 IP 端口。 缺省值：143
securitytype	使用邮件服务器打开的安全连接的类型： · NONE — 进行连接，但不具安全性。 · SSL — 发送 SSLv2 hello，然后协商 SSLv2、SSLv3 或 TLSv1。 · STARTTLS — 进行连接，但不具安全性，然后发出 STARTTLS 命令，之后通过 TLSv1 建立连接。 缺省值：NONE
username	邮箱的名称。
密码	用于登录到邮箱的密码（如果需要）。
authenticationtype	要使用的认证方法（STANDARD 或 CRAM_MD5） 缺省值：STANDARD
sharedsecret	用于 CRAM_MD5 认证的共享密码（如果适用）。

表 76. *IMAP4* 监视器配置 *(续)*	
字段	**描述**
超时	等待服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	指示失败之前重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

正则表达式匹配

您可以通过输入最多 50 个不同的正则表达式来对下载的信息执行正则表达式搜索。监视器尝试将检索的内容与每个正则表达式进行匹配。

如果找到指定的正则表达式的匹配，那么将在对应的 $regexpMatchn 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。$regexpStatusn 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

有关正则表达式语法的信息，请参阅第 280 页的表 50。

监视器元素

除了所有元素共有的测试结果外，IMAP4 监视器还将生成一组包含特定于 IMAP4 服务测试的数据的测试结果。

下表描述 IMAP4 监视器的其他元素。

星号 (*) 指示的元素可用作属性。属性名称显示在括号中。缺少星号指示无等效属性。在括号中显示而无元素的属性指示仅可用作属性，无等效元素。

表 77. *IMAP4* 监视器元素	
元素	**描述**
$authentication	IMAP4 服务器需要的用户认证方法的类型（Standard 或 CRAM-MD5）。
$bytesPerSec	平均每秒传输的字节数。
$bytesTransferred	上载或下载的字节数。
$connectTime	连接到 IMAP4 服务器所花的时间。
$downloadTime*(DownloadTime)	下载文件所花的时间。
$hopCount	消息到达邮箱前作为消息中继的主机数。
$inEvent	指示此事件为一系列事件中的一部分。 1 指示这不是最终事件。 0 指示这是最终事件。
$lookupTime*(LookupTime)	获取主机服务器 IP 地址所花的时间。

表 77. *IMAP4* 监视器元素 *(续)*	
元素	**描述**
$port*(Port)	服务受监视的端口。
$responseTime*(ResponseTime)	建立连接到接收到数据首个字节之间的时间。
$security	在添加 IMAP 元素时使用指定的邮件服务器打开的安全连接的类型（NONE、STARTTLS 或 SSL）。
$sentTo*(SentTo)	SMTP 监视器用于发送原始消息的电子邮件地址。
$smtpServer	从中发送电子邮件的 SMTP 服务器的名称。
$SSLHandshakeTime*(SslHandshakeTime)	建立 SSL 连接所花的时间。
$timeToDeliver	电子邮件消息在 SMTP 监视器及其目标邮箱之间传递所用的时间。
$user*(ImapUser)	监视器用于登录到 IMAP4 服务器的用户名（帐户名称）。

状态消息

在使用 IBM Application Performance Management 时，IMAP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 IMAP4 状态消息。

表 78. *IMAP4* 监视器状态消息	
Message	**描述**
Message successfully downloaded	已成功下载消息。
Downloaded x messages	指示已成功从邮箱下载的消息数量。
Server not IMAP4rev1 compliant	IMAP4 服务器不符合 IMAP4 规范 (RFC2060)。
Server does not support STARTTLS capability	未正确配置服务器。
Unable to log into server	监视器无法登录到 IMAP 服务器。
Unrecognised response to STATUS command	监视器未识别服务器返回的值。
Unrecognised response to FETCH INTERNALDATE command	
Failed to obtain Actual-Time-Sent header	监视器未从服务器获取期望的响应。
Failed to obtain Actually-To header	
Failed to obtain SMTP-Server header	

属性

下表中描述特定于 IMAP4 监视器的属性和命令行选项。

表 *79. IMAP4* 监视器属性		
属性名称	**属性参数**	**描述**
Originator	字符串	指定在检索 SMTP 监视器发送的测试电子邮件消息时要匹配的 From 字段。监视器仅检索 From 字段匹配 Originator 中的字符串的消息。IMAP4 Originator 必须匹配 SMTP 监视器中的 Originator。 缺省值：SMTP-Monitor
SSLCertificate File	字符串	在创建期间未针对 HTTPS 元素明确指定证书时使用的数字证书的路径和文件名。 如果路径不是绝对路径，那么监视器将其解释为相对于工作目录 ($ISHOME/platform/arch/bin）。
SSLCipherSuite	字符串	用于 SSL 操作的密码套件。有关可能的值的描述，请参阅密码套件。 缺省值：RC4:3DES:DES:+EXP
SSLDisableTLS	整数	针对遗留项支持禁用 TLSv1。 缺省值：0 - 启用 TLSv1。 1 - 禁用 TLSv1。
SSLKeyFile	字符串	包含 SSL 专用密钥的文件。
SSLKeyPassword	字符串	用于对 SSL 专用密钥进行加密的密码。

密码套件

SSLCipherSuite 属性指定 IMAP4 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

***NTP* 监视器**

网络时间协议 (NTP) 监视器使用 UDP（用户数据报协议）查询 NTP 服务器以确定服务器是否提供正确的时间。

NTP 使用全球标准时间将计算机时钟同步到毫秒。

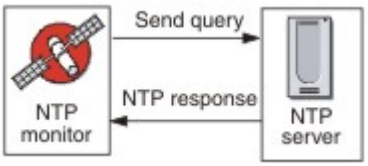
下表列出 NTP 监视器文件。

表 *80. NTP* 监视器文件	
监视器文件	**名称和位置**
监视器可执行文件	nco_m_ntp
属性文件	$ISHOME/etc/props/ntp.props
规则文件	$ISHOME/etc/ntp.rules
日志文件	$ISHOME/log/ntp.log

用于配置 NTP 监视器的准则

NTP 监视器通过向 NTP 服务器发送查询来获取数据，这将返回具有当前时间（如 NTP 服务器所见）的 UDP 响应包。

下图显示监视器与 NTP 服务器之间交换的消息示例。



配置 NTP 监视器服务测试

使用 NTP 监视器配置参数来定义服务测试。

下表描述 NTP 配置：

表 *81. NTP* 配置

字段	描述
server	NTP 服务器的主机名。例如 `ntp.mycompany.com`。
描述	提供关于该元素描述性信息的文本字段。例如 `NTP monitor`。
port	要使用的 NTP 服务器上的端口。 缺省值：`123`
localip	指定监视器在执行测试时绑定到的主机系统上的网络接口的 IP 地址。如果设置了监视器的 `IpAddress` 属性，那么它会覆盖此字段的值。例如 `102.168.n.n`。
version	要使用的 NTP 服务器版本（1、2、3 或 4）。 缺省值：`1`
超时	等待服务器响应的时间（秒）。 缺省值：`10`
retries	监视器重试访问 NTP 服务器的次数。 缺省值：`0`
poll	轮询的间隔时间（秒）。 缺省值：`300`
failureretests	在指示失败之前要重新测试的次数。 缺省值：`0`
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：`10`

服务级别分类

服务级别分类定义用于确定通过 NTP 提供的服务级别的规则。

NTP 监视器的可用服务级别分类选项包括：

```
totalTime
responseTime
lookupTime
offset
```

```
adjustedOffset
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 $message 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。

监视器元素

除所有元素共同的测试结果以外，NTP 监视器还会生成一组包含特定于 NTP 服务测试的数据的测试结果。

下表描述 NTP 监视器的其他元素。

表 *82. NTP* 监视器元素

元素	**描述**
$adjustedOffset	从服务器偏移的时间（以秒为单位）。
$localIP	监视器配置为使用的本地 IP 地址。此元素在仅有一个接口的系统上可能为空白。
$lookupTime* (LookupTime)	获取主机服务器 IP 地址所花的时间。
$ntpVersionIn	来自服务器的响应中使用的协议版本。
$ntpVersionOut	用于发送的协议版本。
$offset	NTP 服务器与运行监视器的系统之间的时间差（以秒为单位）。
$port* (Port)	要使用的 NTP 服务器上的端口。
$responseTime* (ResponseTime)	监视器连接到 NTP 服务器与接收响应之间间隔的时间。
$retries	在未接收到响应标识时要重新发送请求的次数。

状态消息

使用 IBM Application Performance Management 时，NTP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 NTP 状态消息。

表 *83. NTP* 监视器状态消息

消息	**描述**
Successful query	NTP 服务器给出预期响应。
Connection failed	无法初始化 UDP 套接字。

表 *83. NTP* 监视器状态消息 *(续)*	
消息	**描述**
`Failed to send request to NTP server`	无法对 UDP 套接字进行写入。
`No response from server`	NTP 服务器未响应。

NNTP 监视器

NNTP 监视器通过从新闻组进行读取和向其发布来测试 NNTP 服务的可用性。

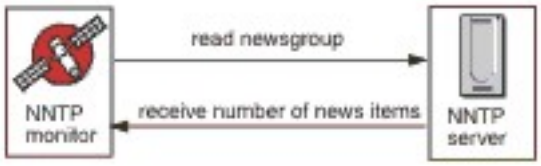
下表列出 NNTP 监视器文件。

表 *84. NNTP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_nntp`
属性文件	`$ISHOME/etc/props/nntp.props`
规则文件	`$ISHOME/etc/rules/nntp.rules`
日志文件	`$ISHOME/log/nntp.log`

用于配置 NNTP 监视器的准则

NNTP 监视器通过向 NNTP 服务器进行发布和从中读取来测试 NNTP 服务。您为监视器创建的每个概要文件元素执行读操作或发布操作。

在读操作中，监视器连接到 NNTP 服务以检查是否存在特定因特网新闻组。如果新闻组存在，那么监视器将记录其中的新闻项数。它还会尝试记录添加到新闻组的最后一个新闻项。下图显示读操作。



在发布操作中，监视器检查新闻组是否存在，然后尝试向其写入测试消息。测试消息的主题是 `NNTP Monitor Test Message`。下图显示发布操作。



每个概要文件元素指定监视器在访问 NNTP 服务器时提供的 `username` 和 `password`。监视器使用明文认证系统。

```
AUTHINFO USER username
AUTHINFO PASS password
```

其中，在监视器概要文件元素中指定 `username` 和 `password`。

属性

下表中描述了特定于 NNTP 监视器的属性选项。

表 *85. NNTP* 监视器属性选项		
属性名称	**属性参数**	**描述**
OutputDirectory	字符串	如果 **OutputResult** 为 true（设置为 1），那么请指定要使用的输出目录。 缺省值：$ISHOME/var
OutputResult	0|1	指定监视器可以保存它从服务接收的数据。 0 - 已禁用 1 - 已启用

配置 NNTP 监视器服务测试

使用 NNTP 监视器配置参数来定义服务测试。

下表列出 NNTP 监视器配置。

表 *86. NNTP* 监视器配置	
字段	**描述**
server	NNTP 服务器的 IP 地址。例如，news.mycompany.com。
newsgroup	监视器用于发布和读取测试消息的新闻组的名称。例如，mycompany.test。
描述	提供关于该元素描述性信息的文本字段。例如，READ。
port	NNTP 服务器的端口号。 缺省值：119
username	用于向 NNTP 服务器进行认证的用户名。
密码	用于向 NNTP 服务器进行认证的用户名的密码。
操作	指示要发布还是检索文章。它可以是 READ 或 POST。 缺省值：POST
超时	等待服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

匹配正则表达式

您可以对通过输入最多 50 个不同的正则表达式下载的信息执行正则表达式搜索。NNTP 监视器尝试将检索到的内容与每个正则表达式进行匹配。

如果找到指定的正则表达式的匹配，那么将在对应的 $regexpMatchn 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。$regexpStatusn 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

有关更多信息，请参阅第 280 页的表 50。

服务级别分类

服务级别分类定义用于确定通过 NNTP 提供的服务级别的规则。

NNTP 监视器的可用服务级别分类选项包括：

```
totalTime
lookupTime
connectTime
transferTime
responseTime
状态 bytesTransferred
bytesPerSec
newsItems
expected
lastLineReceived
checksum
previousChecksum
regexpMatch1 to 3
regexpStatus1 to 3
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 $message 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。
- status 代码 220 和 240 指示成功。请参阅 NNTP 协议以获取操作返回的其他状态码。
- egexpStatus*n* 可能具有以下值：
  - NONE：未配置正则表达式检查
  - MATCHED：找到正则表达式的匹配项
  - FAILED：找不到正则表达式的匹配项
- 评估使用以下格式的测试表达式的正则表达式匹配项：

  ```
  regexpMatchn [contains|!contains] expression
  ```

  使用 contains 和 !contains 运算符代替 = 和 !=，因为 regexpMatch*n* 通常包含与正则表达式匹配的整行而不只是匹配部分，因此 = 和 != 运算符往往与表达式不匹配。
- Checksum 和 PreviousChecksum 元素通常不会为服务级别分类提供有意义的值，因为在创建概要文件元素时，校验和值未知（监视器在测试进行中时计算校验和值）。$checksum 和 $previousChecksum 监视器元素旨在使用监视器的规则文件来丰富警报。

监视器元素

除所有元素共同的测试结果以外，NNTP 监视器还会生成一组包含特定于 NNTP 服务测试的数据的测试结果。

下表描述 NNTP 监视器的其他元素。

表 *87. NNTP* 监视器元素	
元素	**描述**
$action* (NntpAction)	监视器采取的操作。它可以是 READ 或 POST。
$bytesPerSec	平均每秒传输的字节数。
$bytesTransferred	上载或下载的字节数。
$checksum	Checksum 元素通常不会为服务级别分类提供有意义的值，因为在创建概要文件元素时，校验和值未知（监视器在测试进行中时计算校验和值）。$checksum 和 $previousChecksum 监视器元素旨在使用监视器的规则文件来丰富警报。
$connectTime* (ConnectTime)	建立与 NNTP 服务器的连接所需的时间。
$downloadTime	下载文件所花的时间。
$group* (NntpGroup)	受监视新闻组的名称。
$lastLineReceived	仅当 $message 元素包含消息 Expect Failed 时，才会设置此元素。如果设置了该元素，那么它会包含 NNTP 服务器的响应。
$lookupTime* (LookupTime)	查找服务器 IP 地址所需的时间。
$newsItems	新闻组中的新闻项数。
$password	用于认证监视器的密码。
$previousChecksum	PreviousChecksum 元素通常不会为服务级别分类提供有意义的值，因为在创建概要文件元素时，校验和值未知（监视器在测试进行中时计算校验和值）。$previousChecksum 和 $checksum 监视器元素旨在使用监视器的规则文件来丰富警报。
$responseTime* (ResponseTime)	创建连接后直至接收到目标文章的第一个字节所需的时间。
$status	NNTP 服务器返回的状态码。
$transferTime* (TransferTime)	将该值设置为 $uploadTime 或 $downloadTime。
$uploadTime	上载文件所用的时间。
$username	用于认证监视器的用户名。

表 *87. NNTP* 监视器元素 *(续)*	
元素	**描述**
如果 $message 包含 $ExpectFailed	
$expected	连接失败时监视器正在等待的连接中的文本。
$lastLineReceived	监视器从 NNTP 服务器接收到的连接中的最后一行文本。

状态消息

使用 IBM Application Performance Management 时，NNTP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 NNTP 状态消息。

表 *88. NNTP* 监视器状态消息	
消息	**描述**
Article Posted	NNTP POST 操作已成功。
Article Retrieved	NNTP READ 操作已成功。
Not Found	可能找不到文章。
Expect failed	NNTP 请求失败。
Timed out waiting to read	已建立到服务器的数据连接，但是其已停止响应。
Connection failed	监视器无法连接到服务器。有关更多信息，请参阅日志文件。
Connection closed by foreign host	远程主机在监视器期望之前已关闭连接。

***POP3* 监视器**

POP3 监视器与 SMTP 监视器一起测试 POP3 电子邮件服务的可用性和响应时间。

下表列出 POP3 监视器文件。

表 *89. POP3* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_pop3
属性文件	$ISHOME/etc/props/pop3.props
规则文件	$ISHOME/etc/rules/pop3.rules
日志文件	$ISHOME/log/pop3.log

用于配置 POP3 监视器的准则

POP3 监视器与 SMTP 监视器配合使用，监视 SMTP 监视器向其发送测试消息的邮箱并度量交付这些消息所用的时间量。

注: 确保同步监视器主计算机和邮件服务器上的系统时钟以使交付时间计算正常工作。

POP3 监视器读取邮箱内容后，它将生成以下两种不同类型的事件：

· 特定于消息的事件

  POP3 监视器为从邮箱下载的每条电子邮件消息创建一个特定于消息的事件。在这类事件中，监视器将 $message 元素设置为 Message Successfully Downloaded。计算出的 $timeToDeliver 元素值是 SMTP 监视器发出消息后到邮箱接收到消息之间的消息传递时间。$hopCount 元素指示在消息到达邮箱前作为消息中继的主机数。

· 概要事件

  处理完邮箱中的所有消息后，监视器将创建一个概要事件。在这类事件中，$message 元素指示从邮箱中成功下载的消息总数，$totaltime 元素指示完成这些请求所用的时间。$totaltime 以秒为单位。

安全邮件

POP3 监视器支持连接到安全邮件服务。可以使用 SSL/TLS 或 STARTTLS 命令进行连接。在定义 POP3 监视器元素时，使用“安全类型”字段以选择安全安全性。如果邮件服务器需要客户机端证书以用于 SSL 加密，请使用 SSLname 属性或命令行选项以指定证书文件、密钥文件、密钥密码和密码套件。

客户机端证书

使用 POP3 监视器可监视需要客户机端证书以进行相互认证的服务器。在创建概要文件元素时，指定 SSL 证书文件、密钥文件和密钥密码。证书必须为隐私增强邮件（PEM）格式。如果证书为其他格式，那么必须将其转换为 PEM 格式。可使用软件转换证书，例如，openSSL，可从 http://www.openssl.org 获取此软件。

注: 如果始终在所有概要文件元素中使用相同的证书、密钥和密码，那么使用监视器属性指定它们，代替在每个创建的概要文件元素中进行定义。

配置 POP3 监视器测试

注: 将 SMTP 监视器配置为将消息发送至测试邮箱，并将 POP3 监视器配置为检索这些消息，以监视邮件服务器 mail.mycompany.com 的操作。测试邮箱的地址为 ismtest@mycompany.com，凭证为 ismtest/secret1。请使用超时值为 20 秒的连接，如果失败，那么重新测试两次，且每次测试结束后间隔 5 秒再进行下一次测试，请按 10 分钟的时间间隔测试服务。请使用概要文件元素提供的缺省服务级别分类。

表 *90. POP3* 监视器配置

字段	**描述**
服务器	邮件服务器的 IP 地址。示例为 mail.mycompany.com
描述	提供关于该元素描述性信息的文本字段。
端口	邮件服务器的端口号。 缺省值：110
securitytype	使用邮件服务器打开的安全连接的类型： · NONE — 进行连接，但不具安全性 · SSL — 发送 SSLv2 hello，然后协商 SSLv2、SSLv3 或 TLSv1 · STARTTLS — 进行连接，但不具安全性，然后发出 STLS 命令，之后通过 TLSv1 建立连接。这是最安全的安全类型。 · NONE — 进行连接，但不具安全性 缺省值：NONE
username	邮箱的名称。
密码	用于登录到邮箱的密码（如果需要）。

表 *90. POP3* 监视器配置 *(续)*	
字段	**描述**
authenticationtype	要使用的认证方法，标签为认证类型： · STANDARD - 在部件密码时，使用用户/密码交换。这适合于间歇性地使用 POP3。 · APOP - 在 POP3 客户机定期连接到服务器时使用。相对于 STANDARD 而言，它提供了更高级别的安全性。如果选择 APOP，请确保指定 APOP 共享密码。请注意，不是所有服务器都支持 APOP。 缺省值：STANDARD。
sharedsecret	APOP 认证的共享密钥，仅在使用 APOP 认证类型时适用。字符串应该至少八个字符长，且在用户界面中是模糊的。
超时	等待服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	指示失败之前重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
verifycertificate	服务器的验证证书。 缺省值：Disabled

使用 POP3 监视器配置参数可定义服务测试。

正则表达式匹配

您可以通过输入最多 50 个不同的正则表达式来对下载的信息执行正则表达式搜索。监视器尝试将检索的内容与每个正则表达式进行匹配。

如果找到指定的正则表达式的匹配，那么将在对应的 $regexpMatchn 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。$regexpStatusn 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

有关正则表达式语法的信息，请参阅第 280 页的表 50。

监视器元素

除了对所有元素公共的测试结果，POP3 监视器生成一组包含特定于 POP3 服务测试的数据的测试结果。

表 1 描述 POP3 监视器的其他元素。

星号 (*) 指示的元素可用作属性。属性名称显示在方括号中。缺少星号指示无等效属性。在括号中显示而无元素的属性指示仅可用作属性，无等效元素。

表 *91. IMAP4* 监视器元素	
元素	**描述**
$authentication	IMAP4 服务器需要的用户认证方法的类型（Standard 或 CRAM-MD5）。
$bytesPerSec	平均每秒传输的字节数。
$bytesTransferred	上载或下载的字节数。
$connectTime	连接到 IMAP4 服务器所花的时间。
$downloadTime* (DownloadTime)	下载文件所花的时间。
$hopCount	消息到达邮箱前作为消息中继的主机数。
$inEvent	指示此事件为一系列事件中的一部分。1 指示这不是最终事件，0 指示这是最终事件。
$lookupTime*(LookupTime)	获取主机服务器 IP 地址所花的时间。
$port*(Port)	服务受监视的端口。
$responseTime* (ResponseTime)	建立连接到接收到数据首个字节之间的时间。
$security	在添加 IMAP 元素时使用指定的邮件服务器打开的安全连接的类型（NONE、STARTTLS 或 SSL）。
$sentTo*(SentTo)	SMTP 监视器用于发送原始消息的电子邮件地址。
$smtpServer	从中发送电子邮件的 SMTP 服务器的名称。
$SSLHandshakeTime* (SslHandshakeTime)	建立 SSL 连接所花的时间。
$timeToDeliver	电子邮件消息在 SMTP 监视器及其目标邮箱之间传递所用的时间。
$user*(ImapUser)	监视器用于登录到 IMAP4 服务器的用户名（帐户名称）。

状态消息

在使用 IBM Application Performance Management 时，POP3 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 POP3 状态消息。

表 *92. POP3* 监视器状态消息	
Message	**描述**
Message successfully downloaded	POP3 请求已成功。
Downloaded *x* messages	指示从邮箱下载的消息数量。
Timed out waiting to read/write	已建立到服务器的数据连接，但是其已停止响应。
Connection closed by foreign host	远程主机在监视器期望的时间前关闭连接。

表 92. POP3 监视器状态消息 (续)	
Message	**描述**
Connection failed	监视器无法连接到服务器。请参阅日志文件以获取更多信息。
APOP not supported by the server	服务器不支持 APOP 认证方法。改为使用 Standard 认证类型。
APOP service not available	监视器不支持 APOP 服务器实施。改为使用 Standard 认证类型。
Server does not support STLS capability	服务器不支持 STARTTLS。请使用其他安全类型。

属性

下表描述特定于 POP3 监视器的属性。

表 93. POP3 监视器属性和命令行选项		
属性名称	**属性参数**	**描述**
SSLCertificate File	字符串	在创建期间未针对 POP3 元素明确指定证书时使用的数字证书的路径和文件名。 如果路径不是绝对路径，那么监视器将其解释为相对于工作目录 ($ISHOME/platform/*arch*/bin)。
SSLCipherSuite	字符串	用于 SSL 操作的密码套件。缺省值: RC4:3DES:DES:+EXP。请参阅密码套件以获取可能的值的描述。
SSLDisableTLS	整数	针对旧支持禁用 TLSv1。 缺省值：0 - 启用 TLSv1。设置为 1 可禁用 TLSv1。
SSLKeyFile	字符串	包含 SSL 专用密钥的文件。
SSLKeyPassword	字符串	用于对 SSL 专用密钥进行加密的密码。

密码套件

SSLCipherSuite 属性指定 POP3 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

***RADIUS* 监视器**

远程认证拨入用户服务 (RADIUS) 提供对服务的远程访问的认证。RADIUS 监视器模拟访问 RADIUS 服务的客户机系统，并返回有关服务性能的数据。

下表列出 RADIUS 监视器文件。

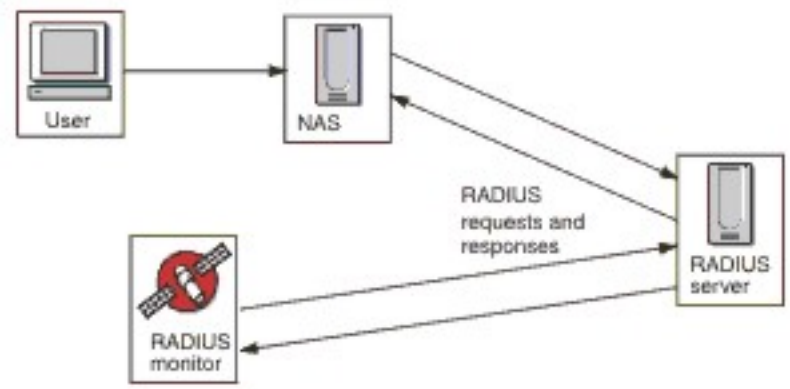
表 94. RADIUS 监视器文件	
监视器文件	**名称和位置**
监视器可执行文件	nco_m_radius
属性文件	$ISHOME/etc/props/radius.props
规则文件	$ISHOME/etc/rules/radius.rules

表 *94. RADIUS* 监视器文件 *(续)*	
监视器文件	**名称和位置**
日志文件	`$ISHOME/log/http.log`

用于配置 Radius 监视器的准则

RADIUS 监视器模拟网络访问服务器 (NAS) 向 RADIUS 服务器发送请求的操作。

RADIUS 监视器使用 UDP 向 RADIUS 服务器发送请求，然后生成包含这些请求的结果以及有关服务器性能的数据的事件。下图显示监视器的操作。



监视器可以测试 RADIUS 服务器的认证和记帐操作：

- 使用密码认证过程 (PAP) 测试 Access-Request
- 使用提问握手认证协议 (CHAP) 测试 Access-Request
- Accounting-Request：Start、Stop、Accounting On 和 Accounting Off

属性

下表中描述了特定于 RADIUS 监视器的属性选项。

表 *95. RADIUS* 监视器属性选项		
属性名称	**属性参数**	**描述**
`FramedServiceRequest`	`0`|`1`	当此属性设置为 `1` 时，监视器将选择 Access-Request 中设置的 Framed 服务类型。 `0` - 已禁用 `1` - 已启用

配置 Radius 监视器服务测试

使用 RADIUS 监视器配置参数来定义服务测试。

下表描述 Radius 监视器配置：

表 *96. RADIUS* 监视器配置	
字段	**描述**
server	RADIUS 服务器的 IP 地址。
sharedsecret	用于认证监视器的共享密钥。
username	监视器提供的用于认证 RADIUS 服务器的用户名。

表 *96. RADIUS* 监视器配置 *(续)*	
字段	**描述**
密码	监视器提供的用于认证 RADIUS 服务器的密码。
描述	提供关于该元素描述性信息的文本字段。
requesttype	指定发送到 RADIUS 服务器的请求的类型： ·Authenticate（CHAP） ·Authenticate（PAP） ·Accounting 缺省值：Authenticate(CHAP)
port	用于连接到 RADIUS 服务器的端口。 缺省值：1812
localip	指定监视器在执行测试时绑定到的主机系统上的网络接口的 IP 地址。如果设置了监视器的 IpAddress 属性，那么它会覆盖此字段的值。
loginhost	设置 Access-Request 中 Login-IP-Host 属性的值。
calledstation	设置 Access-Request 中 Called-Station-Id 属性的值。
callingstation	设置 Access-Request 中 Calling-Station-Id 属性的值。
accountsessionid	设置发送到记帐服务器的 Accounting-Request 包中 Acct-Session-Id 属性的值。 **注:** 此字段仅适用于 Accounting 请求类型。
accountstatustype	设置发送到记帐服务器的 Accounting-Request 包中 Acct-Status-Type 属性的值： ·开始 ·Stop ·Accounting On ·Accounting Off **注:** 此字段仅适用于 Accounting 请求类型。 缺省值：Start
accountsessiontime	设置发送到记帐服务器的 accounting-request 包中 Acct-Session-Time 属性的值（以秒为单位）。 **注:** 此字段仅适用于 Accounting 请求类型。
nasip	RADIUS 监视器发送的作为 Access-Request 包的一部分的 NAS-IP-Address 属性。
nasport	RADIUS 监视器发送的作为 Access-Request 包的一部分的 NAS-Port 属性。

表 96. *RADIUS* 监视器配置 *(续)*	
字段	**描述**
超时	等待服务器响应的时间（秒）。 缺省值：10
retries	在有问题的情况下重试连接到 RADIUS 服务器的次数。 缺省值：0
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

服务级别分类

服务级别分类定义用于确定 RADIUS 服务提供的服务级别的规则。

RADIUS 监视器的可用服务级别分类选项包括：

```
totalTime
lookupTime
responseTime
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 $message 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。

监视器元素

除所有元素共同的测试结果以外，RADIUS 监视器还会生成一组包含特定于 RADIUS 服务测试的数据的测试结果。

下表描述 RADIUS 监视器的其他元素。

表 97. *RADIUS* 监视器元素	
元素	**描述**
$accountSessionId	用于匹配启动和停止记录的唯一标识。
$accountSessionTime	当 accountStatusType 设置为 Stop 时，此字段显示用户接收服务的时间量（以秒为单位）。
$accountStatusType	指示它是用户服务的开始 (start) 还是结束 (stop)。

表 97. *RADIUS* 监视器元素 *(续)*	
元素	**描述**
$calledStationId	RADIUS 监视器将 calledStationId 作为 Access-Request 包的一部分发送。如果 RADIUS 服务器需要，那么会使用该元素；如果使用了 callingStationId，那么不会使用该元素。
$callingStationId	RADIUS 监视器将 callingStationId 作为 Access-Request 包的一部分发送。如果 RADIUS 服务器需要，那么会使用该元素；如果使用了 calledStationId，那么不会使用该元素。
$localIP	监视器配置为使用的本地 IP 地址。此元素在仅有一个接口的系统上可能为空白。
$loginIPHost* (LoginIpHost)	RADIUS 监视器将 loginIPHost 作为 Access-Request 包的一部分发送。受监视的服务器可能需要该元素。
$lookupTime* (LookupTime)	获取主机服务器 IP 地址所花的时间。
$nasPort* (NasPort)	RADIUS 监视器发送的作为 Access-Request 包的一部分的 NAS Port 参数。缺省值：0。
$password	用于认证监视器的密码。
$port* (Port)	服务受监视的端口。
$requestType	指示针对该元素选择的请求类型，即 PAP、CHAP 或 Accounting。
$responseTime	向 RADIUS 服务器发送请求与从它接收到回复之间间隔的时间。
$retries	最大重试次数。
$secret	从配置文件获取的共享密钥密码。
$username* (RadiusUser)	用于认证监视器的用户名。

状态消息

使用 IBM Application Performance Management 时，RADIUS 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 RADIUS 监视器状态消息

表 98. *RADIUS* 监视器状态消息	
消息	**描述**
CHAP authentication - Access granted	已认证监视器（使用 CHAP）。仅当使用了 CHAP 请求类型时，才会返回此消息。

表 98. *RADIUS* 监视器状态消息 *(续)*	
消息	**描述**
PAP authentication - Access granted	已认证监视器（使用 PAP）。仅当使用了 PAP 请求类型时，才会返回此消息。
Accounting response received	从服务器接收到记帐响应。事务将继续。
Connection failed	指定的服务器名称无效。
Failed to send request to RADIUS server	可能无法将 UDP 包写入到网络。没有进一步的错误信息可用。
No response from server	RADIUS 服务器未响应。
Incorrect identifier returned	服务器提供了对于未从监视器发送的请求的响应。
Invalid response authenticator	响应包含非预期授权。它可能是由于共享密钥或密码不正确所导致。
Unrecognized response	服务器未识别所发送的包。
PAP authentication - Access denied	未认证监视器（使用 PAP）。
CHAP authentication - Access denied	未认证监视器（使用 CHAP）。

RPING 监视器

RPING 监视器通过从路由器远程执行 ping 操作来测试网络设备的可用性。它提供最大、最小和平均往返时间性能数据。

监视器支持 Cisco、Juniper 路由器和 RFC2925 兼容路由器。

下表列出 RPING 监视器文件。

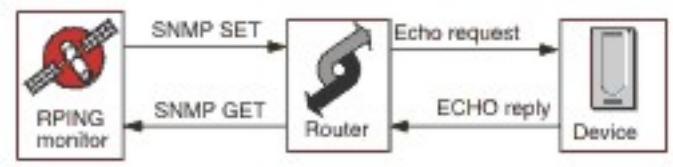
表 99. *RPING* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	nco_m_rping
属性文件	$ISHOME/etc/ims/props/rping.props
规则文件	$ISHOME/etc/ims/rules/rping.rules
日志文件	$ISHOME/log/rping.log
脚本文件	$ISHOME/scripts/rping/cisco.s（Cisco 路由器的 SNMP 脚本） $ISHOME/scripts/rping/juniper.s（Juniper 路由器的 SNMP 脚本） $ISHOME/scripts/rping/rfc2925.s（RFC2925 兼容路由器的 SNMP 脚本）

用于配置 RPING 监视器的准则

RPING 监视器通过配置路由器以对网络设备执行 ping 操作，然后定期轮询路由器以获取 ping 的结果来获取数据。

监视器通过使用 SNMP SET 命令在路由器的 ping MIB 中创建控制行来配置 ping 测试，然后使用 SNMP GET 命令从 MIB 检索 ping 数据。与路由器的所有通信都基于 SNMP。

下图显示监视器与网络设备之间交换的消息示例。



在 Cisco 路由器上启用远程 ping 请求

缺省情况下，Cisco 路由器上的远程 ping SNMP 请求处于禁用状态。但是，要使 RPING 监视器执行 SNMP SET 请求并开始 ping 操作，必须启用此请求。

要启用此请求，请登录到 Cisco 路由器并输入以下命令：

```
enable
config terminal
snmp-server community communitystring rw
write mem
logout
```

路由器中配置的 communitystring 必须与您在为该路由器创建的任何 RPING 概要文件元素的 R/W 社区字符串字段中输入的字符串匹配。write mem 行确保在重新引导路由器时保存设置。

在 Juniper 路由器上启用远程 ping 请求

缺省情况下，Juniper 路由器上的远程 ping SNMP 请求处于禁用状态。要使 RPING 监视器使用 Juniper 路由器运行，必须启用 SNMP 请求。

要在路由器上启用 SNMP 请求，请确保 JUNOS 配置的 SNMP 部分匹配：

```
[edit snmp]
view ping-mib-view {
  oid .1.3.6.1.2.1.80 include; # pingMIB
  oid jnxPingMIB include; # jnxPingMIB
}
community communitystring {
  authorization read-write;
  view ping-mib-view;
}
```

路由器中配置的 communitystring 必须与您在为该路由器配置的任何 RPING 概要文件元素的 communitystring 字段中输入的字符串匹配。

属性

下表中描述了特定于 RPING 监视器的属性选项。

表 *100. RPING* 属性选项

属性名称	属性参数	描述
MibDir	字符串	包含监视器使用的 MIB 文件的目录。 缺省值：`$ISHOME/mibs`。

配置 RPING 监视器服务测试

使用 RPING 监视器配置参数来定义服务测试。

表 *101. RPING* 监视器配置

字段	描述
server	路由器的名称或 IP 地址。例如，`rt1.mycompany.com`。

表 *101. RPING* 监视器配置 *(续)*	
字段	**描述**
routertype	路由器的类型： · CISCO · Juniper · RFC2925
host	您希望路由器 ping 的服务器的名称或 IP 地址。
communitystring	指定用于与路由器进行通信的 SNMP 社区字符串。例如，`server1.mycompany.com`。
描述	提供关于该元素描述性信息的文本字段。例如，`RPING monitor`。
vpn	要用于发送 ping 的 VPN 的可选名称。路由器使用指定的 VPN 而不是配置的缺省路由。
version	要使用的 SNMP 版本： 1 - SNMPv1 2 - SNMPv2c 3 - SNMPv3 缺省值：2
numberofpings	要发送的 ping 的数量。 缺省值：5
packetsize	要发送的包的大小（以字节为单位）。 缺省值：64
packettimeout	在 ping 之间等待的时间（以秒为单位）。 缺省值：500
securityname †	SNMP 会话的用户名。
authenticationphrase †	用户的认证密码。
privacyphrase †	用户的隐私密码。
authenticationprotocol †	用于认证用户的协议： · MD5 · SHA1 缺省值：MD5
privacyprotocol †	用于对会话进行加密的协议。 缺省值：DES
超时	轮询的间隔时间（秒）。 缺省值：10

表 *101. RPING* 监视器配置 *(续)*	
字段	**描述**
retries	监视器重试访问服务器的次数。 缺省值：3
poll	在 ping 之间等待的时间（以秒为单位）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
† 仅适用于 SNMPv3。	

服务级别分类

服务级别分类定义用于确定通过 RPING 提供的服务级别的规则。

RPING 监视器的可用服务级别分类选项包括：

```
totalTime
lookupTime
numPacketSent
numPacketsRecv
maxRTT
minRTT
averageRTT
respondPercent
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- `message` 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。

监视器元素

除所有元素共同的测试结果以外，RPING 监视器还会生成一组包含特定于 RPING 服务测试的数据的测试结果。

下表列出 RPING 监视器的其他元素。

表 *102. RPING* 监视器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
(AverageRTT)	平均往返时间，以秒为单位。
$community	路由器的 SNMP 社区字符串。
$communityString	用于与路由器进行通信的 SNMP 社区字符串。
(MaxRTT)	最长往返时间，以秒为单位。

表 *102. RPING* 监视器元素 *(续)*	
元素	**描述**
(MinRTT)	最短往返时间，以秒为单位。
$numPacketSent	监视器发送的包的数量。
$numPings	添加 RPING 元素时指定的已发送的 ping 的数量。
$packetSize	要发送的包的大小。
$packetTimeout	在发送包之间等待的时间。
$privProto	创建元素时指定的隐私协议。
$remoteHost* (RemoteHost)	您希望路由器 ping 的服务器的名称或 IP 地址。
(RespondPercent)	已发送的存在响应的 ping 的百分比。
$routerMan* (RouterName)	添加 RPING 元素时选择的路由器类型： · CISCO · Juniper · RFC2925
$securityName	创建元素时指定的安全用户名。
(SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	路由器的名称或 IP 地址。
$timeout	服务器必须响应的秒数。从配置文件中获取。
$vpn* (Vpn)	RPING 概要文件元素的 **vpn** 字段中指定的 VPN 的名称。

状态消息

使用 IBM Application Performance Management 时，RPING 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

下表描述 RPING 状态消息。

表 *103. RPING* 监视器状态消息	
消息	**描述**
`Got Response`	监视器接收到来自 Cisco 设备的响应。
`Error in packet - exiting thread`	其中一个包中存在错误。
`Timed out while trying initial sets`	尝试创建 `rowEntry` 字段时，路由器没有任何响应。
`Internal Error`	路由器中存在错误。

表 *103. RPING* 监视器状态消息 *(续)*	
消息	**描述**
`Host poll did not finish`	网络设备未完成 ping 操作。
`Response Failed` `Operation Failed`	路由器无法对网络设备执行 ping 操作。
`Timed out on Get requests`	尝试从路由器获取结果时，监视器超时。

RTSP 监视器

实时流协议 (RTSP) 监视器在流式服务器上测试音频和视频流回放。它收集有关媒体文件的信息，并启动流式会话的流回放、暂停和结束。

下表列出 RTSP 监视器文件。

表 *104. RTSP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_rtsp`
属性文件	`$ISHOME/etc/props/rtsp.props`
规则文件	`$ISHOME/etc/rules/rtsp.rules`
日志文件	`$ISHOME/log/rtsp.log`

用于配置 RTSP 监视器的准则

RTSP 监视器以 DESCRIBE 或 PLAY 方式连接到流式服务器。监视器下载由真正 RTSP 服务器（例如 Darwin）交付的信息或统计信息。



DESCRIBE 方式

在 DESCRIBE 方式下，RTSP 监视器连接到流式服务器，并请求有关音频和视频文件和流的信息。

服务器返回状态码，其中值为 200 指示可以下载的文件，而其他值指示无法播放请求的文件的原因。

但是，在此方式下不会报告与回放相关的统计信息，可以测试支持 RTSP 的服务器的基本功能。

PLAY 方式

在 PLAY 方式下，RTSP 监视器以与 DESCRIBE 方式下相同的方式连接到流式服务器，然后对该文件进行流式处理以提供有关所请求的下载的统计信息。

属性

下表中描述了特定于 RTSP 监视器的属性选项。

表 *105. RTSP* 监视器属性选项		
属性名称	**属性参数**	**描述**
StreamingSocket BufferSize	整数	流式套接字缓冲区的大小，范围为 8 到 64 KB。 缺省值：8

配置 RTSP 监视器服务测试

使用 RTSP 监视器配置参数来定义服务测试。

下表列出 RTSP 监视器配置：

表 *106. RTSP* 监视器配置	
字段	**描述**
server	运行流式服务器的目标系统。例如，`rtsp.mymusic.com`。
remotefile	下载的文件。例如，`singalong.mp3`。
描述	提供关于该元素描述性信息的文本字段。例如，`RTSP monitor`。
port	监视器在目标系统上连接到的端口。 缺省值：554
操作	服务器对流执行的操作： · DESCRIBE · PLAY 缺省值：DESCRIBE
duration	服务器回放的流的部分（以秒为单位）。 缺省值：5
maxbandwidth	用于流式传输的最大带宽（n 比特/秒）。 缺省值：1500000
超时	等待 RTSP 服务器响应的时间（以秒为单位）。 缺省值：10
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

服务级别分类

服务级别分类定义用于确定通过 RTSP 提供的服务级别的规则。

RTSP 监视器的可用服务级别分类选项包括：

```
totalTime
lookupTime
connectTime
responseTime
sdpDownloadTime
playbackTime
状态 percentPacketsLost
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。
- 状态码为 200 表示成功。请参阅 RTSP 协议以获取操作返回的其他状态码。

监视器元素

除所有元素共同的测试结果以外，RTSP 监视器还会生成一组包含特定于 RTSP 服务测试的数据的测试结果。

下表描述 RTSP 监视器的其他元素。

表 *107. RTSP* 监视器元素

元素	**描述**
$action	监视器采取的操作。
$averageBandwidth	平均总带宽（以比特为单位）。
$bytesReceived	接收到的字节总数。
$connectTime* (ConnectTime)	建立与目标服务器的连接所用的时间。
$describeStageStatus	RTSP 对话阶段的状态码。
$filename	媒体文件的名称。
$lookupTime* (LookupTime)	获取主机服务器 IP 地址所花的时间。
$maxBandwidth	使用配置接口时的最大带宽。
$mediaResponseTime	服务器对所请求的文件进行流式处理所需的时间。
$numberOfStreams	媒体中嵌入的流数。
$percentPacketsLost	丢包百分比。
$playbackTime* (PlaybackTime)	表示 setupResponseTime 和 mediaResponseTime 的总和的时间。

表 *107. RTSP* 监视器元素 *(续)*	
元素	**描述**
$playStageStatus	RTSP 对话阶段的状态码。
$port	用于访问监视器的服务器的端口。
$responseTime* (ResponseTime)	从建立连接到接收到数据的第一个字节的间隔时间。
$sdpDownloadTime* (SdpDownloadTimed)	下载有关媒体文件的数据所需的时间。
$setupResponseTime	表示 `playbackTime` 的一部分的时间。 **注:** 仅当 RTSP 监视器以 PLAY 方式运行时，才会生成此元素。
$setupStageStatus	RTSP 对话阶段的状态码。
$status	RTSP 服务器返回的状态码。
$streamingTime	服务器完成对所请求的文件的流式处理所需的时间。
$streamLength	媒体文件中最长的流的长度。
$teardownStageStatus	RTSP 对话阶段的状态码。
$totalBandwidthRequired	总带宽（以千比特/秒为单位）。
$totalPacketsLost	总丢包数。
$totalPacketsReceived	接收到的包的数量。

状态消息

使用 IBM Application Performance Management 时，RTSP 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

表 *108. RTSP* 监视器状态消息	
消息	**描述**
`OK`	请求已成功。
`Connection failed`	监视器无法连接到服务器。有关更多信息，请参阅日志文件。
`Connection closed by foreign host`	与 RTSP 服务器的连接已中断。
`Timed out waiting to read/write`	已建立与 RTSP 服务器的数据连接，但是发生问题。
`Play failed - no streams`	监视器接收到响应，但是没有音频或视频可供回放。

表 *108. RTSP* 监视器状态消息 *(续)*

消息	**描述**
select() failed on RTSP socket (PLAY stage)	套接字已从远程服务器关闭，或者在等待响应时超时。
RTSP Server response not in expected format	来自服务器的响应采用的是监视器不支持的格式。
Redirection requested by server not supported by client	来自服务器的响应不受客户机支持。
Server cannot fulfill client request	请求失败，并且没有进一步的信息可用。
Server Error	服务器发生问题，并且请求失败。 服务器返回了代码 500 或更大值。 有关更多信息，请参阅 RTSP 协议 (RFC 2326)。

表 *108. RTSP* 监视器状态消息 *(续)*	
消息	**描述**
RTSP response header CSeq doesn't match request CSeq	RTSP 服务器配置不正确，并且无法正常运作。
Corrupted RTSP server response	
Corrupted session description	
RTSP SETUP response CSeq doesn't match request CSeq	
RTSP SETUP response, incomplete Session string	
RTSP SETUP response, Session ID has changed within the same session	
RTSP SETUP response does not contain server ports to connect to	
RTSP SETUP response does not contain server port pair to connect to	
RTSP PLAY response CSeq doesn't match request CSeq	
RTSP PLAY response, incomplete Session string	
RTSP PLAY response, Session ID has changed within the same session	
RTSP PLAY response, incomplete RTP-Info string	
RTSP PLAY response does not valid RTP seqnum in RTP-Info response	
RTSP PLAY response does not valid RTP time in RTP-Info response	

SAA 监视器

Cisco Service Assurance 代理程序 (SAA) 是针对 IOS V12.2(2) 和更高版本的 Cisco 产品的性能监视代理程序。

SAA 监视器使用 Cisco Service Assurance 代理程序 工具来测试 Cisco 路由器之间的各种计时。

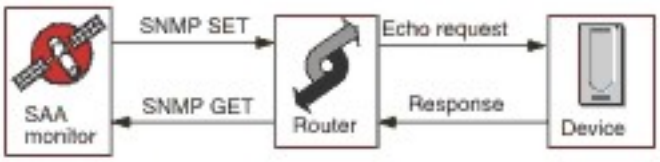
表 *109. SAA* 监视器摘要	
监视器文件	**名称或位置**
可执行文件名称	nco_m_saa
属性文件	$ISHOME/etc/props/saa.props

表 109. SAA 监视器摘要 (续)	
监视器文件	**名称或位置**
规则文件	$ISHOME/etc/rules/saa.rules
日志文件	$ISHOME/log/saa.log
脚本目录	$ISHOME/scripts/saa/

用于配置 SAA 监视器的准则

SAA 监视器将路由器的 SAA 配置为通过使用 Cisco Response Time Monitor 的管理信息库 (MIB) 中定义的定时回传请求或响应来测试其他网络设备或服务的可用性。监视器使用简单网络管理协议与 Service Assurance 代理程序 进行通信。

下图演示 SAA 监视器的操作。



操作

SAA 监视器将 Service Assurance 代理程序 配置为在其他网络设备上运行回传测试（称为探测器）。您可以配置一系列不同的探测器，其中每个探测器使用不同的协议。

所有探测器都可以针对任何已启用 IP 的目标进行操作（Jitter 除外，它需要另一个支持 SAA 的 Cisco 响应程序路由器）。

每个监视器概要文件元素会在启动时在路由器上启动 Service Assurance 代理程序 探测器，通过每个连续轮询，它将收集结果信息并重新调度探测器。如果探测器意外停止，那么监视器会立即将其重新启动。当探测器测试完成时，它会进入不活动状态，直至下一个监视器轮询为止。在下一个监视器轮询中，将会收集结果数据并开始另一个测试循环。在每个轮询期间，监视器会检查探测器的状态。如果探测器仍在运行，那么监视器会将其停止，然后轮询管理信息库 (MIB) 以获取上一个循环的结果数据和错误信息。之后它将重新调度探测器，重置统计数据，并重新激活探测器，该探测器在下一个监视器轮询之前将以无人照管方式运行。

为防止进程在路由器上处于不受控制状态的可能性，监视器将启动具有在每个监视器轮询时延长的预定义生命周期的探测器。如果终止监视器，它将继续运行，直至其生命周期到期为止。此后，它将转入不活动状态，直至到达老化时间为止，然后路由器将结束进程。

不必预配置 IOS 和 Service Assurance 代理程序，因为监视器会在运行时自动配置、控制和清除探测器。这包括配置某些探测器类型所需的响应程序路由器。

探测器持久性

ProbePersist 监视器属性控制监视器轮询期间的探测器持久性。如果未启用探测器持久性，那么探测器将在每次轮询时启动，并在生成测试结果后立即结束。

路由器负载

有时，探测器操作可能会受到路由器负载的影响。StatusWait 属性为探测器提供在将操作视为失败之前从一个状态更改为另一个状态所需的时间。

探测器类型

可用于 SAA 监视器的探测器类型为如下所列：

- DHCP
- DLSW
- DNS
- FTP
- HTTP Get Requests
- ICMP Echo
- ICMP Path Echo

· Jitter
· UDP Echo
· SNA-Echo
· VOIP

Echo 探测器基于时间范围执行测试，而 Jitter、VOIP 和 HTTP 探测器通过单个操作执行测试。

SAA 属性
您必须设置 SAA 监视器的属性。

下表描述 SAA 监视器的属性。

表 *110. SAA* 监视器属性

属性名称	**属性参数**	**描述**
AgeOut	整数	探测器在停止前保持不活动状态的最大秒数。 缺省值为 600。
MibDir	字符串	用于 MIB 文件的目录。 缺省路径为 `$ISHOME/mibs`
ProbeLife	整数	探测器在无人照管时保持活动状态的最大秒数。 缺省值为 600。
ProbePersist	0|1	探测器可以在两种方式下运行。它们在每个监视器轮询执行一个测试循环，或者它们启动一次并在每个轮询重新调度。 · 0 指示单个测试循环 · 1 指示在每个轮询重新调度
StatusWait	整数	监视器等待探测器在失败之前完成任何操作的秒数。

配置 *SAA* 监视器服务测试
您必须配置 SAA 监视器参数以定义服务测试。

下表描述 SAA 监视器配置字段。

表 *111. SAA* 监视器配置

字段	**描述**
server	Cisco 路由器的名称或 IP 地址。
communitystring	路由器的 SNMP 社区字符串。
probetype	适用于概要文件元素的 SAA 探测器类型。
描述	用于提供元素的描述性信息的文本字段。
动态	指示概要文件元素是否处于活动状态。
port	用于访问路由器的端口。 缺省端口为 161。

表 111. SAA 监视器配置 (续)	
字段	**描述**
version	要使用的 SNMP 版本： · 1 用于 SNMPv1 · 2 用于 SNMPv2c · 3 用于 SNMPv3 缺省值为 1。
probeid	指定用于生成探测器控制行索引的值。
securityname †	SNMP 会话的用户名。
authenticationphrase †	用户的认证密码。
privacyphrase †	用户的隐私密码。
authenticationprotocol †	用于认证用户的协议如下所示： · MD5 · SHA1 缺省值为 MD5。
privacyprotocol †	用于对会话进行加密的协议。这是 DES。
超时	等待路由器响应的时间（以秒为单位）。 缺省值为 5。
retries	监视器在退出之前重试访问路由器的次数。 缺省值为 0。
poll	轮询的间隔时间（秒）。 缺省值为 300
failureretests	在指示失败之前要重新测试的次数。 缺省值为 0。
retestinterval	失败时进行重新测试要等待的间隔时间（以秒为单位）。 缺省值为 10。

注: † 仅适用于 SNMPv3。

探测器类型配置

对于每种类型的探测器，探测器配置不同，并且 Internet Service Monitoring 代理程序提供一组特定于每种类型的配置字段。要创建概要文件元素，请选择探测器类型，然后提供适合于该类型的配置。有关个别配置项的信息，请参阅 Cisco Response Time Monitor MIB 文档。

服务级别分类

服务级别分类定义用于确定网络设备提供的服务级别的规则。

SAA 监视器的可用服务级别分类选项如下所示：

```
totalTime
errTotal
numRTT
minRTT
maxRTT
avgRTT
```

```
minPosJitterSD
maxPosJitterSD
minNegJitterSD
maxNegJitterSD
minPosJitterDS
maxPosJitterDS
minNegJitterDS
maxNegJitterDS
packetLossSD
packetLossDS
packetOutOfSequence
packetMIA
packetLateArrival
minDelaySD
maxDelaySD
minDelayDS
maxDelayDS
avgPosJitterSD
avgPosJitterDS
avgNegjitterSD
avgNegJitterDS
avgDelaySD
avgDelayDS
devPosJitterSD
devPosJitterDS
devNegJitterSD
devNegJitterDS
devDelaySD
devDelayDS
MOS
ICPIF
mMinRTT
httpRTT
dnsRTT
tcpConnectRTT
transactionRTT
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅第 349 页的『状态消息』。
- 操作数是字符串或正数。

监视器元素
除所有元素共同的测试结果以外，SAA 监视器还会生成一组包含特定于使用中的探测器类型的数据的测试结果。

DHCP 探测器
DHCP 探测器生成多个元素。

下表描述 DHCP 探测器元素。

表 *112. DHCP* 探测器元素

元素	**描述**
$authProto	创建元素时指定的认证协议。
$community	用于向 SAA 发送 SNMP 请求的社区。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	dhcp
$securityName	创建元素时指定的安全用户名。

表 *112. DHCP* 探测器元素 *(续)*	
元素	**描述**
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 DHCP 请求的路由器的名称。
$totalRTT † (TotalRTT)	从 DHCP 服务器获取 IP 所需的总往返时间（以秒为单位）。

注: † 指示该元素可用于服务级别分类。

DLSW 探测器
DLSW 探测器生成多个元素。

下表描述 DLSW 探测器元素。

表 *113. DLSW* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgRTT* † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
$errTotal* † (ErrorTotal)	出错包的总数。
$maxRTT* † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT* † (MinimumRTT)	最短往返时间（以秒为单位）。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType †	dlsw
$securityName	创建元素时指定的安全用户名。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于执行 SAA 测试的路由器。
$sumOfRTT (TotalRTT)	所有往返时间的总和（以秒为单位）。
(TargetHost)	目标 SAA 运行所在的主机的名称或 IP 地址。

注: † 指示该元素可用于服务级别分类。

DNS 探测器
DNS 探测器生成多个元素。

下表描述 DNS 探测器元素。

表 *114. DNS* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$community	用于向 SAA 发送 SNMP 请求的社区。
$dnsHost (Host)	要从服务器解析的主机。
$dnsServer	DNS 服务器的 IP。
(HostLookup)	主机的 IP 地址。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	dns
$securityName	创建元素时指定的安全用户名。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 DNS 请求的路由器的名称。
$totalRTT † (TotalRTT)	DNS 查找的总往返时间（以秒为单位）。

注: † 指示该元素可用于服务级别分类。

FTP 探测器
FTP 探测器生成多个元素。

下表描述 FTP 探测器元素。

表 *115. FTP* 探测器元素	
元素	**描述**
$activePassive	测试中使用的连接类型，即 `Active` 或 `Passive`。 缺省值：`Passive`
$authProto	创建元素时指定的认证协议。
$community	用于向 SAA 发送 SNMP 请求的社区。
$errorStatus	指示测试状态的结果字符串（来自 `rttMonLatestRttOperSense` MIB 对象）。
$ftpFile	测试期间检索的测试文件的名称。
$ftpUrl (FtpUrl)	FTP 测试中使用的 URL。

表 *115. FTP* 探测器元素 *(续)*	
元素	**描述**
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$securityName	创建元素时指定的安全用户名。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 FTP 请求的路由器的名称。
$totalRTT (TotalRTT)	测试的完成时间（来自 `rttMonLatestRttOperCompletionTime` MIB 对象）（以秒为单位）。

HTTP-Get 探测器
HTTP-Get 探测器生成多个元素。

下表描述 HTTP-Get 探测器元素。

表 *116. HTTP-Get* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$community	用于向 SAA 发送 SNMP 请求的社区。
$dnsRTT † (DnsRTT)	执行 DNS 查询的往返时间（以秒为单位）。
$httpRTT † (HttpRTT)	执行 HTTP 操作的往返时间（以秒为单位）。
(HttpUrl)	受监视的 URL。
$messageBodyBytes	接收到的消息体的大小。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	http-get
$securityName	创建元素时指定的安全用户名。
(SourceRouter)	用于发送 HTTP 请求的路由器的名称。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
$targetHost	受测试服务的主机的名称。
$tcpConnectRTT † (TcpConnectRTT)	连接到 HTTP 服务器的往返时间（以秒为单位）。

表 116. *HTTP-Get* 探测器元素 *(续)*	
元素	**描述**
$transactionRTT † (TransactionRTT)	下载 URL 指定的对象的往返时间（以秒为单位）。

注: † 指示该元素可用于服务级别分类。

ICMP-Echo 探测器
ICMP-Echo 探测器生成多个元素。

下表描述 ICMP-Echo 探测器元素。

表 117. *ICMP-Echo* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
$errBusies	由于先前 ping 未完成而失败的 ping 的数量。
$errDisconnects	通过断开连接失败的 ping 的数量。
$ErrDrops	由于内部资源不可用而失败的 ping 的数量。
$errNoConnects	由于无法建立与目标的连接而失败的 ping 的数量。
$errSequences	由于接收到意外序列标识而失败的 ping 的数量。
$errTimeouts	通过超时失败的 ping 的数量。
$errTotal † (ErrorTotal)	出错包的总数。
$errVerifies	由于接收到的数据与预期数据不同而失败的 ping 的数量。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	探测器类型必须如下所示: · icmp-echo · icmp-echo-path · udp-echo
$securityName	创建元素时指定的安全用户名。

表 *117. ICMP-Echo* 探测器元素 *(续)*	
元素	**描述**
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 ICMP 请求的路由器的名称。
$sumOfRTT	所有往返时间的总和（以秒为单位）。
$targetHost (Host)	受监视服务的主机名。
$tos (Tos)	服务类型值。
$vpn (Vpn)	VPN 的名称。

注: † 指示该元素可用于服务级别分类。

ICMP-Path-Echo 探测器
ICMP-Patch-Echo 探测器生成多个元素。

下表描述 ICMP-Patch-Echo 探测器元素。

表 *118. ICMP-Patch-Echo* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
(HopHost*One to Eight*)	使用 ICMP Echo Path 访问的第一个到第八个主机。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	探测器类型如下: ·icmp-echo ·icmp-echo-path
$securityName	创建元素时指定的安全用户名。

表 *118. ICMP-Patch-Echo* 探测器元素 *(续)*	
元素	**描述**
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 ICMP 请求的路由器的名称。
$sumOfRTT	所有往返时间的总和（以秒为单位）。
$targetHost	受测试服务的主机的名称。
$tos (Tos)	服务类型值。
$vpn (Vpn)	VPN 的名称。

注: † 指示该元素可用于服务级别分类。

Jitter 探测器
Jitter 探测器生成多个元素。

下表描述 Jitter 探测器元素。

表 *119. Jitter* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgDelayDS †	从目标到源的平均延迟（以秒为单位）。
$avgDelaySD †	从源到目标的平均延迟（以秒为单位）。
$avgNegJitterDS †	从目标到源的平均负 Jitter（以秒为单位）。
$avgNegJitterSD †	从源到目标的平均负 Jitter（以秒为单位）。
$avgPosJitterDS †	从目标到源的平均正 Jitter（以秒为单位）。
$avgPosJitterSD †	从源到目标的平均正 Jitter（以秒为单位）。
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
$devDelayDS †	从目标到源的延迟的标准偏差。
$devDelaySD †	从源到目标的延迟的标准偏差。
$devNegJitterDS †	从目标到源的负 Jitter 的标准偏差。
$devNegJitterSD †	从源到目标的负 Jitter 的标准偏差。
$devPosJitterDS †	从目标到源的正 Jitter 的标准偏差。
$devPosJitterSD †	从源到目标的正 Jitter 的标准偏差。
$errDescription	错误的描述。

表 *119. Jitter* 探测器元素 *(续)*	
元素	**描述**
$errTotal (ErrorTotal)	出错包的总数。
$maxDelayDS †	从目标到源的最大延迟（以秒为单位）。
$maxDelaySD †	从源到目标的最大延迟（以秒为单位）。
$maxNegJitterDS †	从目标到源的最大负 Jitter 值（以秒为单位）。
$maxNegJitterSD †	从源到目标的最大负 Jitter 值（以秒为单位）。
$maxPosJitterDS †	从目标到源的最大正 Jitter 值（以秒为单位）。
$maxPosJitterSD †	从源到目标的最大正 Jitter 值（以秒为单位）。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minDelayDS †	从目标到源的最小延迟（以秒为单位）。
$minDelaySD †	从源到目标的最小延迟（以秒为单位）。
$minNegJitterDS †	从目标到源的最小负 Jitter 值（以秒为单位）。
$minNegJitterSD †	从源到目标的最小负 Jitter 值（以秒为单位）。
$minPosJitterDS †	从目标到源的最小正 Jitter 值（以秒为单位）。
$minPosJitterSD †	从源到目标的最小正 Jitter 值（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$numNegJitterDS	从目标到源的负 Jitter 值的数量。
$numNegJitterSD	从源到目标的负 Jitter 值的数量。
$numOW	延迟的单向操作数。
$numPosJitterDS	从目标到源的正 Jitter 值的数量。
$numPosJitterSD	从源到目标的正 Jitter 值的数量。
$numRTT †	成功往返数。
$packetLateArrival †	超时后到达的包的数量。
$packetLossDS †	从目标到源丢失的包的数量。
$packetLossSD †	从源到目标丢失的包的数量。
$packetMIA †	方向未知时丢失的包的数量。
$packetOutOfSequence †	无序返回的包的数量。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType †	Jitter

表 *119. Jitter* 探测器元素 *(续)*	
元素	**描述**
(ResponderRouter)	用于响应 Jitter 请求的路由器的名称。
$securityName	创建元素时指定的安全用户名。
(SourceRouter)	用于发送 Jitter 请求的路由器的名称。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
$sum2DelayDS	从目标到源的延迟的平方和。
$sum2DelaySD	从源到目标的延迟的平方和。
$sum2NegJitterDS	所有负 Jitter 值的平方和。
$sum2NegJitterSD	所有负 Jitter 值的平方和。
$sum2PosJitterDS	所有正 Jitter 值的平方和。
$sum2PosJitterSD	所有正 Jitter 值的平方和。
$sum2Rtt †	往返值的平方和（以秒为单位）。
$sumDelayDS	从目标到源的延迟总和（以秒为单位）。
$sumDelaySD	从源到目标的延迟总和（以秒为单位）。
$sumNegJitterDS	所有负 Jitter 值的总和（以秒为单位）。
$sumNegJitterSD	负 Jitter 值的总和（以秒为单位）。
$sumPosJitterDS	所有正 Jitter 值的总和（以秒为单位）。
$sumPosJitterSD	所有正 Jitter 值的总和（以秒为单位）。
$sumRTT	所有往返的总和（以秒为单位）。。
$targetHost	受测试服务的主机的名称。
$tos (Tos)	服务类型值。
$vpn (\Vpn)	VPN 的名称。

注: † 指示该元素可用于服务级别分类。

SNA-Echo 探测器
SNA-Echo 探测器（SNA-RU-Echo、SNA-LU0-Echo、SNA-LU2-Echo、SNA-LU62-Echo 和 SNA-LU62Native-Echo）生成下表中所列的元素。

下表描述 JSNA-Echo 探测器元素。

表 *120. SNA-Echo* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。

表 *120. SNA-Echo* 探测器元素 *(续)*	
元素	**描述**
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
$errTotal †	出错包的总数。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType (ProbeType)	sna-*name*-echo
$securityName	创建元素时指定的安全用户名。
(SourceRouter)	用于发送 SNA 请求的路由器的名称。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
$sumOfRTT (TotalRTT)	所有往返时间的总和（以秒为单位）。
(TargetHost)	SNA 回传请求的主机目标。

注: † 指示该元素可用于服务级别分类。

UDP-Echo 探测器
UDP-Echo 探测器生成下表中所列的元素。

下表描述 UDP-Echo 探测器元素。

表 *121. UDP-Echo* 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。
$errBusies	由于先前 ping 未完成而失败的 ping 的数量。
$ErrDrops	由于内部资源不可用而失败的 ping 的数量。

表 121. UDP-Echo 探测器元素 (续)	
元素	**描述**
$errTimeouts	通过超时失败的 ping 的数量。
$errTotal † (ErrorTotal)	出错包的总数。
$errVerifies	由于接收到的数据与预期数据不同而失败的 ping 的数量。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$numRTT †	成功往返数。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType	udp-echo
$securityName	创建元素时指定的安全用户名。
$snmpVersion* (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
$sumOfRTT	所有往返时间的总和（以秒为单位）。
$targetHost (Host)	受监视服务的主机名。
$tos (Tos)	服务类型值。
$vpn (Vpn)	VPN 的名称。

注: † 指示该元素可用于服务级别分类。

VOIP 探测器
VOIP 探测器与 Jitter 探测器生成相同的元素。此外，它们还生成下表中所列的元素。

下表描述 VOIP 探测器元素。

表 122. VOIP 探测器元素	
元素	**描述**
$authProto	创建元素时指定的认证协议。
$avgRTT † (AverageRTT)	平均往返时间，以秒为单位。
$community	用于向 SAA 发送 SNMP 请求的社区。

表 122. VOIP 探测器元素 (续)	
元素	**描述**
$errTotal † (ErrorTotal)	出错包的总数。
$ICPIF †	ICPIF 值。
$maxRTT † (MaximumRTT)	最长往返时间（以秒为单位）。
$minRTT † (MinimumRTT)	最短往返时间（以秒为单位）。
$MOS †	来自测试的平均意见分数 (MOS) 的值。
$port	用于连接到 SAA 的端口。
$privProto	创建元素时指定的隐私协议。
$probeType †	voip
(ResponderRouter)	用于响应 VOIP 请求的路由器的名称。
$securityName	创建元素时指定的安全用户名。
$snmpVersion (SnmpVersion)	用于发送 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。
(SourceRouter)	用于发送 VOIP 请求的路由器的名称。
(Tos)	服务类型值。
(Vpn)	VPN 的名称。

注: † 指示该元素可用于服务级别分类。

状态消息
使用 IBM Application Performance Management 时，SAA 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

下表描述 SAA 监视器的状态消息。

表 123. SAA 监视器状态消息	
消息	**描述**
成功	探测器操作已成功。
`Operation failed`	探测器操作失败。
`Invalid status`	由于状态无效，探测器操作失败。

SIP 监视器
SIP 监视器检查会话启动协议 (SIP) 服务器的可用性，包括注册和认证端点所需的时间。监视器启动 SIP 会话，以便可以监视 SIP 请求和 SIP 响应。

下表列出 SIP 监视器文件。

表 *124. SIP* 监视器文件摘要	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_sip`
属性文件	`$ISHOME/etc/props/sip.props`
规则文件	`$ISHOME/etc/rules/sip.rules`
日志文件	`$ISHOME/log/sip.log`

用于配置 SIP 监视器的准则

SIP 监视器通过在 SIP 服务器上向已启用 SIP 的设备的 URI 发送请求，并在 SIP 服务器上接收来自 SIP 设备的响应，从而测试 SIP 服务器的可用性。

SIP 监视器充当用户代理程序客户机 (UAC)；它启动用于测试 SIP 服务的连接。用户代理服务器 (UAS)、呼叫的接收方或目标可以是任何已启用 SIP 的设备，例如运行软电话的计算机或消息库。

测试 SIP 服务器时，监视器会执行以下操作序列：

1. 使用概要文件元素中提供的凭证向 SIP 服务器注册。
2. 向 UAS 发送 OPTIONS 请求。
3. 向 UAS 发送 INVITE 请求。

   如果 UAS 接受该请求，那么注册成功的测试结果。
4. 向 UAS 发送 BYE 请求，并结束与 UAS 的连接。
5. 从 SIP 服务器注销，并立即到期。

监视器记录在测试中执行的每个操作的持续时间。

属性

下表中描述了特定于 SIP 监视器的属性选项。

表 *125. SIP* 监视器属性选项		
属性名称	**属性参数**	**描述**
ShowZeroes	0|1	指定值为 0 的 SIP 统计信息的显示。 0 - 已禁用 1 - 已启用
Transports	字符串	列出以空格分隔的本地协议端口传输，其中协议为 TCP 或 UDP。允许使用通配符端口号。 缺省值：UDP:*。

密码套件

SSLCipherSuite 属性指定 SIP 监视器使用的密码套件。有关更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

配置 SIP 监视器服务测试

使用 SIP 监视器配置参数来定义服务测试。

表 *126. SIP* 监视器配置	
字段	**描述**
server	指定要测试的服务器的名称。例如，`sip1.mycompany.com`。

表 126. SIP 监视器配置 (续)	
字段	**描述**
serverport	SIP 监视器访问要测试的服务器可以通过的端口。
username	指定呼叫的 SIP 监视器的分机号或帐户身份。例如，`jblogg`。
target	指定用于进行呼叫的已启用 SIP 的设备的分机号。例如，`5551234`。
密码	指定用户名的密码。
描述	提供关于该元素描述性信息的文本字段。例如，`SIP monitor`。
proxy	代理服务器的主机名。例如，`proxy.mycompany.com`。
proxyport	SIP 监视器访问代理服务器可以通过的端口。
超时	等待服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要等待的间隔时间（以秒为单位）。 缺省值：10

服务级别分类

服务级别分类定义用于确定通过 SIP 提供的服务级别的规则。

SIP 监视器的可用服务级别分类选项包括：

```
totalTime
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- `message` 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。

监视器元素

除所有元素共同的测试结果以外，SIP 监视器还会生成一组包含特定于 SIP 服务测试的数据的测试结果。

下表描述 SIP 监视器的其他元素。

表 127. SIP 监视器元素	
元素	**描述**
$AcceptReg	已接受的 SIP 注册请求数。

表 *127. SIP* 监视器元素 *(续)*	
元素	**描述**
$AuthTime* (AuthenticationTime)	授权 SIP 监视器和已启用 SIP 的设备所需的时间。
$authAttempts	监视器重新发送请求以包含其凭证所需的次数。
(CallSetupTime)	设置呼叫所需的时间。
$Invalid	无效的已发送和已接收请求数。
$InvalidReg	无效的 SIP 注册请求数。
$lastMethod	监视器看到的除 BYE 或 ACK 以外的最后一个方法。
$lastSequence [METHOD]	针对方法接收到的最后一个序列。
$lastStatus [METHOD]	针对方法或整体接收到的最后一个状态。
$method *METHOD*	针对方法看到的消息的总计。
$optionsTime* (OptionsTime)	协商选项更改所需的时间（OPTIONS 到 200 OK）。
$postDialTime* (PostDialTime)	在拨号后接收到响铃信号所需的时间（INVITE 到 180 Ring）。
$RegTime* (RegistrationTime)	注册 SIP 监视器和已启用 SIP 的设备所需的时间。
$registrationTime	向服务器注册所需的时间（REGISTER 到 200 OK）。
$Requests	已接收和已发送的 SIP 请求消息数。
($RequestsSent)	已发送的 SIP 请求消息数。
`${request|response}[Sent|Received|Transmitted|Total][METHOD][STATUS]`	针对各种类别看到的消息的总计，例如 `requestSentINVITE = 1`, `responseReceived = 10` 和 `responseReceivedBYE200 = 1`。
$Responses	已接收和已发送的 SIP 响应消息数。
($ResponseReceived)	已接收的 SIP 响应消息数。
$sessionAnswered	· 1 - 如果已应答呼叫 · 0 - 如果未应答呼叫

表 *127. SIP* 监视器元素 *(续)*	
元素	**描述**
$sessionCreated	· 1 - 如果已建立会话 · 0 - 如果未建立会话
$sessionTerminated	· 1 - 如果会话结束 · 0 - 如果会话未结束。
$shutdownTime* (ShutdownTime)	结束连接所需的时间（BYE 到 200 OK）。
$terminatedReason* (TerminatedReason)	连接关闭的原因。
(Username)	用于登录到 SIP 服务器的用户名。
(Target)	要将会话打开到的目标。

状态消息

使用 IBM Application Performance Management 时，SIP 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

下表描述 SIP 监视器状态消息。

表 *128. SIP* 监视器状态消息	
消息	**描述**
`Register timed out`	监视器无法注册到服务器。
`Invite timed out`	INVITE 消息超时。
`OK`	请求和响应已成功。
`n operation status description`	· *n* 是消息号序列。 · operation 是消息类型。 · status 是状态码。 · description 是状态的明文描述。 例如，`1 INVITE 200 OK`。

SIP 响应

SIP 监视器支持以下类型的响应。每个响应具有 3 位数代码：

· 参考响应 (100 - 199)

· 成功响应 (200 - 299)

· 重定向响应 (300 - 399)

· 客户机故障响应 (400 - 499)

· 服务器故障响应 (500 - 599)

· 全局故障响应 (600 - 699)

下表列出常见 SIP 响应。

表 *129.* 常见 *SIP* 响应	
响应	**描述**
100 Trying	已启用 SIP 的设备接收到消息但尚未处理。
180 Ringing	已启用 SIP 的设备接收到消息并进行了处理。设备正在响铃以提醒用户。
200 OK	成功完成方法时会返回此代码。例如，呼叫向服务器注册，或者用户已应答呼叫。
401 Unauthorized	用户未经授权。
407 Proxy Authentication Required	此代码类似于 401，但指示用户必须首先认证。
408 Request Timeout	用户未应答呼叫。

要获取 SIP 响应的完整列表，请参阅 RFC3261。

SMTP 监视器

SMTP 监视器与 IMAP4 或 POP3 监视器一起测试电子邮件服务器的性能。

下表列出 SMTP 监视器文件。

表 *130. SMTP* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_smtp`
属性文件	`$ISHOME/etc/props/smtp.props`
规则文件	`$ISHOME/etc/rules/smtp.rules`
日志文件	`$ISHOME/log/smtp.log`

用于监视 SMTP 监视器的准则

SMTP 监视器与 POP3 或 IMAP4 监视器配合使用。其定期向目标服务器上的邮箱发送电子邮件消息，并记录发出发送电子邮件请求所用的时间。然后，POP3 或 IMAP4 监视器从邮箱读取消息并使用它们来计算电子邮件服务的响应时间和可用性。

注: SMTP 监视器与 POP3 或 IMAP4 监视器配合使用。其定期向目标服务器上的邮箱发送电子邮件消息，并记录发出发送电子邮件请求所用的时间。然后，POP3 或 IMAP4 监视器从邮箱读取消息并使用它们来计算电子邮件服务的响应时间和可用性。

邮箱

您可以配置监视器以向现有邮箱发送电子邮件消息，即使邮箱属于实际用户。但是，建议针对服务测试创建专门的邮箱帐户。电子邮件参数指定收件人邮箱。缺省情况下，监视器使用主题行 `SMTP Monitor Test Message` 发送测试消息。如果需要，您可以配置不含邮箱名称的 SMTP 概要文件元素。在此配置中，监视器只是检查 SMTP 服务是否正在接受连接。

安全邮件

SMTP 监视器支持连接到安全邮件服务。可以使用 SSL/TLS 或 STARTTLS 命令进行连接。在定义 SMTP 监视器元素时，使用“安全类型”字段以选择安全安全性。如果邮件服务器需要客户机端证书以用于 SSL 加密，请使用 SSLname 属性或命令行选项以指定证书文件、密钥文件、密钥密码和密码套件。

客户机端证书

使用 SMTP 监视器可监视需要客户机端证书以进行相互认证的服务器。创建概要文件元素时您将指定 SSL 证书文件、密钥文件和密钥密码。证书必须为隐私增强邮件（PEM）格式。如果证书采用其他格式，那么必须将其转换为 PEM 格式。可使用软件转换证书，例如，openSSL，可从 http://www.openssl.org 获取此软件。

注: 如果始终在所有概要文件元素中使用相同的证书、密钥和密码，那么使用监视器属性指定它们，代替在每个创建的概要文件元素中进行定义。

配置 SMTP 监视器服务测试

使用 SMTP 监视器配置参数可定义服务测试。配置监视器时，对于超时和轮询时间间隔参数，将显示缺省值。缺省值分别为 30 和 300 秒。如果未指定任何值，那么在配置期间不显示表中列出的其他缺省值，而是在保存配置详细信息时应用。

表 *131. SMTP* 监视器配置

字段	描述
服务器	邮件服务器的 IP 地址。示例为 `mail.mycompany.com`
描述	提供关于该元素描述性信息的文本字段。
端口	邮件服务器的端口号。 缺省值：25 如果使用 SMTP 服务器之外的服务器，那么更新连接到服务器的端口。例如，如果将通过 SSL 的 IMAP4 服务器用于 Microsoft Exchange，请指定端口 465。
securitytype	使用邮件服务器打开的安全连接的类型: · NONE — 进行连接，但不具安全性 · SSL — 发送 SSLv2 hello，然后协商 SSLv2、SSLv3 或 TLSv1 · STARTTLS — 进行连接，但不具安全性，然后发出 STARTTLS 命令，之后通过 TLSv1 建立连接。 缺省值：NONE
username	用于登录到 SMTP 服务器的用户名。用于 PLAIN 或 CRAM-MD5 认证。
密码	用于登录到 SMTP 服务器的密码。用于 PLAIN 或 CRAM-MD5 认证。
authenticationtype	用于向 SMTP 服务器认证监视器的方法。可用选项为: · NONE — 不尝试认证 · PLAIN — 纯文本用户名和密码认证 · CRAM-MD5 — 使用 CRAM-MD5 认证 缺省值为 NONE。
sharedsecret	用于 CRAM-MD5 认证的共享密码密钥。
email	SMTP 和 POP3 监视器使用的邮箱的电子邮件地址。
超时	等待 SMTP 服务器响应的时间（秒）。 缺省值：30
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	指示失败之前重新测试的次数。 缺省值：0

表 *131. SMTP* 监视器配置 *(续)*	
字段	**描述**
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

注: 通过尝试以 10 分钟时间间隔连接到邮件服务器 mail.mycompany.com 来监视其可用性。使用 30 秒连接超时，并且如果连接失败，那么以 5 秒重试间隔重试 3 次。

监视器元素

除了对所有元素公共的测试结果，SMTP 监视器生成一组包含特定于 SMTP 服务测试的数据的测试结果。

下表描述 SMTP 监视器的其他元素。

表 *132. SMTP* 监视器元素	
元素	**描述**
$authentication	SMTP 服务器所需的用户认证方法的类型（Standard 或 APOP）。
$bytesPerSec	平均每秒传输的字节数。
$bytesTransferred	上载或下载的字节数。
$connectTime* (ConnectTime)	连接到 SMTP 服务器所用的时间。
$email* (EmailAddress)	监视器向其发送测试电子邮件的邮箱的电子邮件地址。
$lookupTime* (LookupTime)	获取主机服务器 IP 地址所花的时间。
$port* (Port)	服务受监视的端口。
$responseTime* (ResponseTime)	创建连接后直至可将测试电子邮件的第一个字节发送到 SMTP 服务器所用的时间。
$security	使用在概要文件元素 **securitytype** 字段中设置的邮件服务器打开的安全连接的类型（NONE、STARTTLS 或 SSL）。
$SSLHandshakeTime* (SslHandshakeTime)	建立 SSL 连接所花的时间。
$status* (ResultStatus)	SMTP 服务器返回的状态码。
$uploadTime* (UploadTime)	上载文件所用的时间。

表 132. SMTP 监视器元素 (续)	
元素	**描述**
$user* (SmtpUser)	监视器用于登录到 SMTP 服务器的用户名（帐户名称）。

状态消息

在使用 IBM Application Performance Management 时，SMTP 监视器在 $message 元素中提供状态消息。这些消息指示测试的结果。

属性

下表描述特定于 SMTP 监视器的属性。

表 133. SMTP 监视器属性和命令行选项		
属性名称	**属性参数**	**描述**
MailMessage Path	字符串	包含要在测试电子邮件中发送的文本的文件的路径。如果未设置，那么将发送缺省消息。
Originator	字符串	指定 From 字段以设置发送测试电子邮件的时间。确保这匹配 IMAP4 监视器中对应的字符串。 缺省值：SMTP-Monitor。
SSLCertificate File	字符串	在创建期间未针对 SMTP 元素明确指定证书时使用的数字证书的路径和文件名。 如果路径不是绝对路径，那么监视器将其解释为相对于工作目录 ($ISHOME/platform/arch/bin）。
SSLCipherSuite	字符串	用于 SSL 操作的密码套件。有关可能的值的描述，请参阅密码套件。 缺省值：RC4:3DES:DES:+EXP
SSLDisableTLS	整数	针对旧支持禁用 TLSv1。 缺省值：0 - 启用 TLSv1。 1 - 禁用 TLSv1。
SSLKeyFile	字符串	包含 SSL 专用密钥的文件。
SSLKeyPassword	字符串	用于对 SSL 专用密钥进行加密的密码。
UseBody	整数	指定监视器在电子邮件消息中写入跟踪信息的位置，即邮件标题或邮件正文。 缺省值：0 - 信息包含在邮件标题中。 1 - 将信息写入邮件正文。

密码套件

SSLCipherSuite 属性指定 SMTP 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

***SNMP* 监视器**

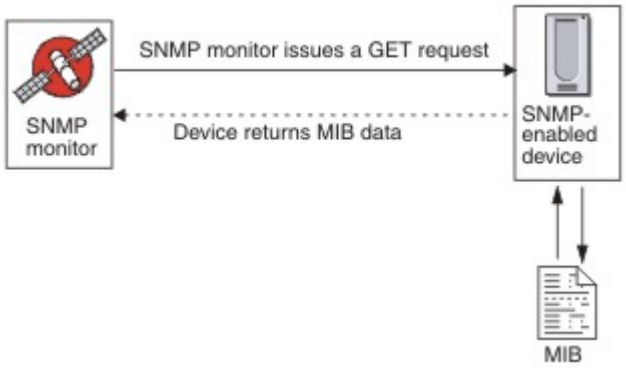
SNMP 监视器测试已启用 SNMP 的设备的性能和故障数据。

下表列出 SNMP 监视器文件。

表 *134. SNMP* 监视器摘要	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_snmp`
属性文件	`$ISHOME/etc/props/snmp.props`
规则文件	`$ISHOME/etc/rules/snmp.rules`
日志文件	`$ISHOME/log/snmp.log`

用于配置 SNMP 监视器的准则

SNMP 监视器通过针对设备 MIB 中包含的一个或多个对象发送 SNMP GET 请求来从已启用 SNMP 的设备获取数据。然后，设备将 MIB 数据返回到 SNMP 监视器。SNMP 监视器支持 SNMP V1、V2c 和 V3。



属性

下表中描述了特定于 SNMP 监视器的属性选项。

表 *135. SNMP* 监视器属性选项		
属性名称	**属性参数**	**描述**
InvalidBps Value	整数	指定一个整数值，当只有一个数据点可用时，该整数值替换为比特/秒 (Bps) 值计算。
MibDir	字符串	指定包含监视器使用的 MIB 文档的目录。 缺省值：`$ISHOME/mibs`。
StripQuotes	`0|1`	删除整数数据中的引号字符。 `0` - 已禁用 `1` - 已启用
Rollover Threshold	整数	在出现路由器重置之前发生回滚的情况下，增量必须达到或超过的值。 缺省值：`0`（从不回滚）

配置 SNMP 监视器服务测试

使用 SNMP 监视器配置参数来定义服务测试。

表 *136. SNMP* 监视器配置

字段	**描述**
server	要将 SNMP GET 请求发送到的服务器。
objectgroupname	要包含在 GET 请求中的 OID 组的文本名称。
communitystring	客户机上的 SNMP 服务器的 SNMP 读/写社区字符串。 **注:** 建议在共用名中使用插入标记字符 (^)，请参阅共用名以获取更多信息。
描述	提供关于该元素描述性信息的文本字段。
port	要使用的服务器上的端口。 缺省值：161
version	要使用的 SNMP 版本： 1 - SNMPv1 2 - SNMPv2c 3 - SNMPv3 缺省值：1
securityname †	SNMP 会话的用户名。
authenticationphrase †	用户的认证密码。
privacyphrase †	用户的隐私密码。
authenticationprotocol †	用于认证用户的协议： · MD5 · SHA1 缺省值：MD5
privacyprotocol †	用于对会话进行加密的协议。 缺省值：DES
超时	等待服务器响应的时间（秒）。 缺省值：20
poll	轮询的间隔时间（秒）。 缺省值：300
retries	监视器在退出之前重试访问服务器的次数。 缺省值：0
failureretests	在指示失败之前要重新测试的次数。 缺省值：0

表 136. SNMP 监视器配置 (续)	
字段	**描述**
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
hostnamelookuppreference	确定应用于所提供主机名的 IP 版本（IPv6 或 IPv4）。选项为： · 缺省设置监视器使用监视器宽度属性设置。这是缺省值。 · 先 4 后 6 先选择 IPv4 再选择 IPv6。IPv4 地址可用时使用 IPv4 地址。如果未找到 IPv4 地址，则使用 IPv6 地址。 · 先 6 后 4 先选择 IPv6 在选择 IPv4。IPv6 地址可用时使用 IPv6 地址。如果未找到 IPv6 地址，则使用 IPv4 地址。 · 仅 4 仅选择 IPv4。只使用 IPv4 地址。如果 IPv4 地址不存在，则轮询返回错误。 · 仅 6 仅选择 IPv6。只使用 IPv6 地址。如果 IPv6 地址不存在，则轮询返回错误。 · 6 或 4 选择 IPv4 或 IPv6。使用从主机名返回的第一个地址。
† 仅适用于 SNMPv3。	

共用名

Internet Service Monitoring 在将信息发送到目标设备时使用插入标记字符 (^) 作为转义字符。如果共用名包含插入标记，那么必须在一行中输入两个插入标记 (^^) 以使名称在路由器中正确。例如，要使共用名 a$^&b 在发送到设备时正确，请使用 a$^^&b。

服务级别分类

服务级别分类定义用于确定服务级别的规则。

SNMP 监视器的可用服务级别分类选项包括：

```
totalTime
message
```

在服务级别分类中：

· 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。

· message 可以是在 **$message** 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。

· 操作数是字符串或正数。

· oidName 是分配给 OID 组中定义的 OID 名称字段中的 MIB 对象的名称。

监视器元素

除所有元素共同的测试结果以外，SNMP 监视器还会生成一组包含特定于 SNMP 服务测试的数据的测试结果。

下表描述 SNMP 监视器的其他元素。

表 137. SNMP 监视器元素	
元素	**描述**
$community	客户机上的 SNMP 服务器的 SNMP 社区字符串。

表 *137. SNMP* 监视器元素 *(续)*	
元素	**描述**
$numOids	查询中使用的 OID 的数量。
$oidGroupName* (OidGroup)	OID 组的名称。OID 组包含监视器正在轮询的 OID。
$oidName*0 to n** (OIDName*Zero to Nine*)	OID 组中第一个到最后一个 MIB 对象的名称。在使用 Netcool/OMNIbus 时，它通过数字指示；在使用 IBM Application Performance Management 时，通过字母文本 (zero to nine) 指示。
$oidNames	以竖线 (|) 分隔的每个 OID 的名称。
$oidReturnValues*0 to n** (snmpResult*Zero to Nine*)	SNMP GET 命令针对 OID 组中第一个到最后一个 MIB 对象返回的数据。在使用 Netcool/OMNIbus 时，它通过数字指示；在使用 IBM Application Performance Management 时，通过字母文本 (zero to nine) 指示。
$oidUnit*0 to n*	OID 组中第一个到最后一个 MIB 对象的单元，通过数字指示。
$oidUnits	以竖线字符 (|) 分隔的每个 OID 的单元。
$port	服务受监视的端口。
$snmpVersion* (SnmpVersion)	用于发送概要文件中配置的 SNMP 包的 SNMP 版本（V1、V2c 或 V3）。

状态消息

使用 IBM Application Performance Management 时， SNMP 监视器在 `ResultMessage` 属性中提供状态消息。这些消息指示测试的结果。

下表描述 SNMP 监视器状态消息。

表 *138. SNMP* 监视器状态消息	
消息	**描述**
`Successful Get`	SNMP 代理程序的查询成功。
`Failed to open snmp sessions` `SNMP session - start failed`	无法初始化 SNMP 会话。
`Error in packet`	无法创建有效的 SNMP 包。
`Timed out while waiting for response`	未从 SNMP 代理程序接收到响应。
`Internal Error`	这是监视器中的内部错误。有关更多信息，请联系 IBM 技术支持。
`Error Processing OID`	处理其中一个 `OID` 时发生错误。
`ERROR: Too Many OIDs`	监视器设置为一次请求过多 `OID`。最大值为 `100`。

表 *138. SNMP* 监视器状态消息 *(续)*	
消息	**描述**
`ERROR: PDU received mismatch with PDU sent`	监视器接收到的协议数据单元 (PDU) 与发送到服务器的 PDU 不匹配。

SOAP 监视器

SOAP 监视器检查 SOAP 接口（SOAP 1.0 和 1.1）的可用性和响应时间。它还可以监视 SOAP 输入（请求）和 SOAP 输出（响应）的有效性。

SOAP 监视器支持以下消息编码样式:

· RPC - 已编码

· Document - Literal Unwrapped

· Document - Literal Wrapped

下表列出 SOAP 监视器文件。

表 *139. SOAP* 监视器文件摘要	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_soap`
属性文件	`$ISHOME/etc/props/soap.props`
规则文件	`$ISHOME/etc/rules/soap.rules`
日志文件	`$ISHOME/log/soap.log`

用于配置 SOAP 监视器的准则

SOAP 监视器通过向目标 SOAP 接口发送包含一组输入的请求，然后接收并分析从接口接收到的响应中包含的输出来测试 SOAP 服务的操作。向 SOAP 接口发送请求后，请求可能成功或失败。如果接收到响应并且响应消息中的值与指定的输出值匹配，那么请求成功。如果未接收到响应或者接收到响应但其消息中的值与输出值不匹配，那么请求失败。

请求和响应中包含的 SOAP 输入和输出取决于受测试 SOAP 服务的功能，在为 SOAP 服务设计测试时，必须指定适合该服务的输入和输出。输入由要发送的数据的名称及其分配的输入值组成。输出由要接收的数据的名称及其预期输出值组成。这些数据名称源自您在配置 SOAP 监视器时指定的本地 Web Service 描述语言 (WSDL) 文件。输入和输出数据名称必须与 WSDL 文件中的名称和数据类型相匹配。数据名称还必须与在 WSDL 文件中的顺序相同。如果名称不匹配，或者顺序不正确，那么在监视器尝试轮询 SOAP 接口时会生成错误消息。

输入的格式为:

```
dataname:datatype=assigned_value, dataname:datatype=assigned_value, ...
```

输出的格式为:

```
dataname:datatype=expected_value, dataname:datatype=expected_value, ...
```

SOAP 数据类型

SOAP 监视器支持简单、数组和用户定义的数据类型。简单数据类型包括整数、字符串和布尔值。数组可能包含简单数据类型以及其他数组和用户定义的数据类型。

表 *140.* 简单数据类型			
简单数据类型			
anyURI	float	language	Qname
boolean	gDay	long	short
byte	gMonth	姓名	字符串
date	gMonthDay	NCName	time
dateTime	gYear	negativeInteger	token
decimal	gYearMonth	NMTOKEN	unsignedByte
double	标识	NMTOKENS	unsignedInt
duration	IDREFS	nonNegativeInteger	unsignedLong
ENTITIES	int	nonPostiveInteger	unsignedShort
ENTITY	整数	normalizedString	

SOAP 认证

如果要监视的 SOAP 接口需要基本 HTTP 认证，请在使用 Internet Service Monitoring Configuration 工具时指定用于访问 SOAP 概要文件元素中的接口的凭证。

要设置必需的 SOAP 认证参数，请执行以下操作：

1. 在 Internet Service Monitoring Configuration 工具中，选择要为其添加认证信息的概要文件元素。
2. 在**高级**选项卡上的 `username` 参数的**值**字段中单击并输入必需值。
3. 在 `password` 参数的**值**字段中单击并输入必需值。密码已加密。
4. 单击**确定**。

如果不再需要认证，请删除 **username** 和 **password** 参数的值。

属性

下表中描述了特定于 SOAP 监视器的属性选项。

表 *141. SOAP* 监视器属性选项			
属性名称	**属性参数**	**描述**	**缺省值**
SoapParser	字符串	XML 解析库。	`$ISHOME/platform/$ARCH/bin/AxisXMLParserXerces.dll`
SoapTransport	字符串	SOAP 传输库。	`$ISHOME/platform/$ARCH/bin/HTTPTransport.dll`
SoapChannel	字符串	SOAP 通道库	`$ISHOME/platform/$ARCH/bin/HTTPChannel.dll`
SoapSecureChannel	字符串	SOAP 安全通道库。	`$ISHOME/platform/$ARCH/bin/HTTPSSLChannel.dll`
SoapClientLog	字符串	额外 SOAP 客户机日志文件的名称。	`$ISHOME/log/SoapClient.log`

密码套件

SSLCipherSuite 属性指定 SOAP 监视器使用的密码套件。

有关更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

配置 SOAP 监视器服务测试

使用 SOAP 监视器配置参数来定义服务测试。

表 *142. SOAP* 监视器配置

元素	**描述**
wsdl	WSDL 文件的本地副本的路径。
operation	SOAP 操作的名称。
operationnamespace	SOAP 操作的名称空间。
位置	要受监视的 SOAP 服务的 URL。
描述	提供关于该元素描述性信息的文本字段。
超时	等待 SOAP 服务响应的时间（以秒为单位）。 缺省值：10
poll	轮询的间隔时间（秒）。 缺省值：300
failureretests	在指示失败之前要重新测试的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10
Soap 参数	
inputs	为 SOAP 输入提供对名称、类型和值字段（包括属性）的访问权。使用简单、复杂或数组 SOAP 参数。例如： · 简单：symbol:string="IBM" · 复杂： outer:{item1:string,item2:string}**(aaa:string='bbb')** ={item1**(attr:string='ccc')**='', item2**(attr:string='ddd',attr2:string='eee')**='fff'} 在此示例中，括号中以粗体标记的属性是可选的。 · 数组：input:int[]=[1,2,3,4]
outputs	为 SOAP 输出提供对名称、类型和值字段（包括属性）的访问权。使用简单、复杂或数组 SOAP 参数。 有关语法的更多信息，请参阅 SOAP 参数输出示例。

服务级别分类

服务级别分类定义用于确定 SOAP 接口提供的服务级别的规则。

SOAP 监视器的可用服务级别分类选项包括：

```
totalTime
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是 **$message** 元素到 IBM Application Performance Management 服务器中的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。

监视器元素

除所有元素共同的测试结果以外，SOAP 监视器还会生成一组包含特定于 SOAP 服务测试的数据的测试结果。

下表列出 SOAP 监视器的其他元素。

表 *143. SOAP* 监视器元素

元素	**描述**
(Location)	受监视的 SOAP 服务的 URL。
(Operation)	受监视的 SOAP 服务的名称。
$outputMatch	如果返回的值与输出值匹配，那么显示 Success，否则显示 Failure。
$responseValueName	SOAP 响应中接收到的值名称。
$soapname	SOAP 响应中的容器名称。仅适用于数组和用户定义的复杂数据类型。
$soaptype	SOAP 响应中的容器类型。仅适用于数组和用户定义的复杂数据类型。
(WSDL)	WSDL 文件的本地副本的路径。

状态消息

使用 IBM Application Performance Management 时，SOAP 监视器在 **ResultMessage** 属性中提供状态消息。这些消息指示测试的结果。

如果返回的值与输出值或错误消息匹配，那么消息为 Success。错误消息包含错误的描述。

示例

按照 5 分钟的时间间隔监视 SOAP 接口的可用性。如果 SOAP 接口不可用，那么最多重复测试两次，每次重复测试间隔 5 秒。发送添加 1 + 2 的请求，并检查响应是否包含值 3。

创建 SOAP 概要文件元素并设置下表中显示的字段。

表 *144. SOAP* 概要文件元素示例

配置字段	**值**
wsdl	c:\%ISMHOME%\etc\SOAP.wsdl
operation	add
operationnamespace	http://localhost/SOAP/Calculator
位置	http://serverA/SOAP/Calculator
描述	basic Calculator SOAP monitor

表 *144. SOAP* 概要文件元素示例 *(续)*	
配置字段	**值**
动态	Selected
超时	30
poll	300
failureretests	2
retestinterval	5
inputs	[in0=1,in1=2]
outputs	[addReturn=3]

***TCPPort* 监视器**

TCPPort 监视器提供其他监视器未测试的服务的覆盖范围。其检测并响应 TCP 端口上的命令或字符串。此监视器专门用于监视定制服务。

下表列出 TCPPort 监视器文件。

表 *145. TCPPort* 监视器文件	
监视器文件	**名称或位置**
监视器可执行文件	`nco_m_tcpport`
属性文件	`$ISHOME/etc/props/tcpport.props`
规则文件	`$ISHOME/etc/rules/tcpport.rules`
日志文件	`$ISHOME/log/tcpport.log`

用于配置 TCPPort 监视器的准则

TCPPort 监视器通过连接到服务，监视从服务接收到的消息以及向其发送响应来测试基于 TCP 的服务。

要配置测试，需要定义包含该服务中常规交互的预期消息和响应序列。

例如，telnet 服务的标准交互包含以下序列：

- telnet 服务发送登录消息，提示输入用户名。
- 客户机发送包含用户名的响应。
- telnet 服务发送提示输入密码的消息。
- 客户机发送包含密码的响应。
- 如果尝试登录成功，那么 telnet 服务发送某种形式的问候消息。

所指定的监视器的 **WaitForn** 和 **Sendn** 属性定义期望的消息和对这些消息的响应。监视器属性文件中的这些属性定义监视器如何与 TCP 服务进行交互：

- **WaitForn** 属性是正则表达式。监视器使用这些属性来匹配在受监视端口上接收的消息。
- **Sendn** 属性是监视器写入到端口的文字串。

注: 如果需要，可以使用支持控制字符插入的文本编辑器将控制字符插入到这些属性中。

用于定义 `WaitForn` 和 `Sendn` 属性的格式为：

```
WaitFor1: 1st received message
Send1: 1st response
WaitFor2: 2nd received message
```

```
Send2: 2nd response
...
WaitFor5: 5th received message
Send5: 5th response
```

监视器到达第一个未设置的 **WaitFor** 属性时，会停止发送和接收。如果 **MonitorDisconnect** 属性设置为 0，那么受监视服务必须关闭监视器打开的连接，否则，监视器将在其 **$message** 元素中报告消息 `Timed out waiting to read`。使用大量服务时，可通过发送 `quit` 命令关闭连接。如果 **MonitorDisconnect** 设置为 1，那么在最后一个 `Send` 或 `WaitFor` 命令完成或者超时（无论哪个先发生）之后监视器会断开连接。

配置 TCPPort 监视器服务测试

使用 TCPPort 监视器配置参数可定义服务测试。

表 *146. TCPPort* 配置

字段	描述
服务器	目标服务运行所在的系统的 IP 地址。示例为 `server.mycompany.com`
端口	连接到目标服务的端口。
description	提供关于该元素描述性信息的文本字段。
超时	等待服务器响应的时间（秒）。 缺省值：`30`
poll	轮询的间隔时间（秒）。 缺省值：`300`
failureretests	在指示失败之前要重新测试的次数。 缺省值：`0`
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：`10`

注: 监视在主机 `server.mycompany.com` 的端口 23 上运行的 `telnet` 服务的可用性。使用凭证 `user` 或 `guest` 以登录到服务器并在登录后立即关闭连接。按 5 分钟时间间隔运行测试，并针对连接尝试设置 10 秒超时。

1. 将以下条目添加到到 TCPPort 属性文件：

```
WaitFor1: ".*[Ll]ogin:"
Send1: "user"
WaitFor2: ".*[Pp]assword:"
Send2: "guest"
WaitFor3: ".*%"
Send3: "exit"
```

2. 启动或重新启动 TCPPort 监视器。

正则表达式匹配

对通过输入最多 50 个不同的正则表达式下载的信息执行正则表达式搜索。TCPPort 监视器尝试将检索到的内容与每个正则表达式进行匹配。如果找到指定的正则表达式的匹配，那么将在对应的 `$regexpMatchn` 元素中返回匹配行（或可适用监视器的内部高速缓存数量的行）。如果在下载的信息中匹配多次正则表达式，那么仅返回第一个匹配。`$regexpStatusn` 元素指示每个正则表达式测试的状态。您可以使用正则表达式匹配及其状态信息作为服务级别分类的条件。

有关更多信息，请参阅第 280 页的表 50。

监视器元素

下表描述 TCPPort 监视器的其他元素。

星号 (*) 指示的元素可用作属性。属性名称显示在方括号中。缺少星号指示无等效属性。在方括号中显示但没有元素的属性指示它们仅可用作属性，无等效元素。

除所有元素共同的测试结果以外，TCPPort 监视器还会生成一组包含特定于 TCPPort 服务测试的数据的测试结果。

表 *147. TCPPort* 监视器元素

元素	描述
$bytesPerSec	平均每秒传输的字节数。
$bytesTransferred	上载或下载的字节数。
$connectTime*(ConnectTime)	建立与目标服务器的连接所用的时间。
$downloadTime*(DownloadTime)	下载数据所用的时间。
$lastlineThere’s	从目标服务器收到的最后一行的内容。
$lookupTime*(LookupTime)	获取主机服务器 IP 地址所花的时间。
$networkError	连接期间包含任何网络错误。
$port* (Port)	监视器尝试连接到的目标服务器上的端口。
$waitingFor	如果连接在监视器完成其等待和发送序列之前终止，那么此元素包含最后一个 WaitFor 属性的内容。

状态消息

在使用 IBM Application Performance Management 时，TCPPort 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述 TCPPort 状态消息。

表 *148. TCPPort* 监视器状态消息

Message	描述
OK	请求已成功。
Timed out waiting to read/write	已建立到服务器的数据连接，但其未响应。
Connection closed unexpectedly	与服务器的连接已断开。
Connection failed	监视器无法连接到服务器。有关更多信息，请参阅日志文件。
Network connect error	网络发生问题。
Network error whilst reading	

属性

下表中描述了特定于 TCPPort 监视器的属性选项。

表 *149. TCPPort* 属性		
属性名称	**属性参数**	**描述**
Monitor Disconnect	0|1	指定监视器必须在最后一个 Send 或 WaitFor 命令后断开连接。如果最后一个命令是 Send，那么监视器在发送字符串后立即断开连接。如果最后一个命令是 WaitFor，那么监视器在监视器收到匹配时或者在执行轮询超时时立即断开连接。 0 - 已禁用（监视器不连接） 1 - 已启用
OutputDirectory	字符串	指定在保存 OutputResult 时要使用的输出目录。 缺省值：$ISHOME/var。
OutputResult	0|1	指定监视器必须保存它从服务接收的数据。 0 - 已禁用 1 - 已启用
Send	*n*	监视器写入到端口的文字串。请参阅用于配置 TCPPort 监视器的准则。 n 是范围 1 - 30（包括边界）内的数字。
singleLineMatch	0|1	指定在匹配表达式时监视器应返回单行匹配。 0 - 已禁用（匹配多行） 1 - 已启用（匹配单行）
WaitFor	*n*	用于匹配受监视端口上的命令或字符串的正则表达式。有关更多信息，请参阅用于配置 TCPPort 监视器的准则。 n 是 1 - 30 之间的数字（包含）。

密码套件

SSLCipherSuite 属性指定 TCPPORT 监视器使用的密码套件。有关 SSL 设置的更多信息，请参阅第 377 页的『Internet Service Monitoring 中的 SSL 设置』。

TFTP 监视器

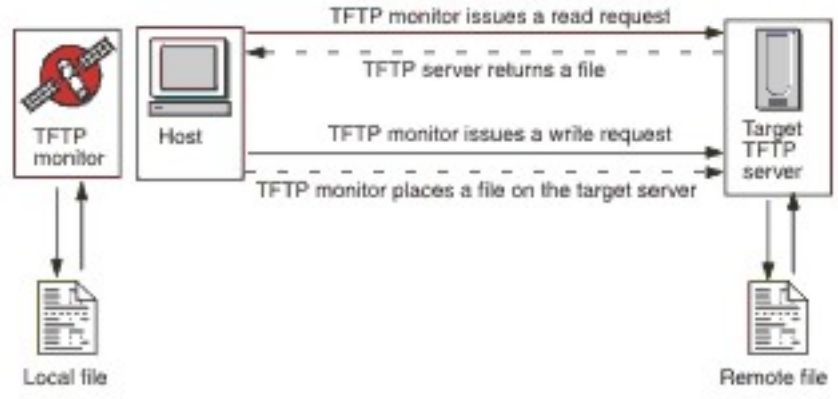
TFTP 监视器度量两个系统之间的次要文件传输协议 (TFTP) 服务的性能。

下表列出 TFTP 监视器文件。

表 *150. FTP* 监视器摘要	
监视器文件	**名称或位置**
可执行文件名称	nco_m_tftp
属性文件	$ISHOME/etc/props/tftp.props
规则文件	$ISHOME/etc/rules/tftp.rules
日志文件	$ISHOME/log/tftp.log

用于配置 TFTP 监视器的准则

TFTP 监视器使用 TFTP READ 或 WRITE 请求在主机系统与目标服务器之间传输文件，然后记录响应时间和数据传输率。使用该监视器可确保 TFTP 服务器已启动且正在运行，并以可接受的速率传输文件。



为上载文件，监视器将发送 TFTP WRITE 请求 (WRQ)，并下载它发送 TFTP READ 请求 (RRQ) 的文件。在 TFTP 客户机中，上载操作为 PUT，下载操作为 GET。

TFTP 监视器支持 octet（二进制）和 netascii 文件传输方式。

配置 TFTP 监视器服务测试

使用 TFTP 监视器配置参数来定义服务测试。

表 *151. TFTP* 监视器配置

字段	描述
server	目标 TFTP 服务器或要双向传输文件的系统的 IP 地址。
localfile	对于 GET 操作，此字段指定文件下载到的名称和路径。 对于 PUT 操作，此字段指定上载到服务器的文件的名称和路径。
remotefile	对于 GET 操作，此字段指定从服务器下载的文件的名称和路径。 对于 PUT 操作，此字段指定文件在服务器上上载到的名称和路径。
描述	用于提供有关 TFTP 监视器的描述性信息的文本字段。
port	TFTP 服务器使用的端口。 缺省值：69
localip	监视器打开 TFTP 连接时所在的主机网络接口的 IP 地址。如果此字段为空，那么监视器使用由 IpAddress 属性指定的接口
localport	监视器用于建立 TFTP 连接的端口。如果此字段的值为 0，那么监视器将选择合适的端口。
command	供监视器使用的 TFTP 命令： · GET - 将文件从目标服务器下载到监视器主机。 · PUT - 将文件从监视器主机上载到目标服务器。 缺省值：GET

表 *151. TFTP* 监视器配置 *(续)*	
字段	**描述**
transfermode	指定监视器传输文件所采用的格式： ·OCTET（8 位） ·NETASCII 缺省值：OCTET
超时	等待 TFTP 服务器响应的时间（以秒为单位）。 缺省值：10
retries	监视器在退出文件之前尝试传输文件的次数。 缺省值：3
poll	轮询的间隔时间（秒）。请勿将该值设置过低，因为不断轮询可能会使服务超负荷。 缺省值：300
failureretests	在指示失败之前，监视器在初始故障后重新测试 TFTP 服务器的次数。 缺省值：0
retestinterval	失败时进行重新测试要间隔的时间（秒）。 缺省值：10

服务级别分类

服务级别分类定义用于确定 TFTP 服务器提供的服务级别的规则。

TFTP 监视器的可用服务级别分类选项包括：

```
totalTime
lookupTime
responseTime
transferTime
bytesTransferred
bytesPerSec
checksum
message
```

在服务级别分类中：

- 通过手动输入监视器元素的名称来指定更多服务级别分类。名称必须与针对“监视器元素”部分中的元素显示的名称相匹配。
- message 可以是在 $message 元素中转发到 IBM Application Performance Management 服务器的任何消息（如果在任何窗口小部件中使用）。要获取可能的值的列表，请参阅状态消息。
- 操作数是字符串或正数。
- checksum 元素通常不提供有意义的服务级别分类结果。在创建概要文件元素时，其值未知。监视器在测试正在进行时计算校验和值。此元素旨在通过使用规则文件来丰富警报。

监视器元素

除所有元素共同的测试结果以外，TFTP 监视器还会生成一组包含特定于 TFTP 服务测试的数据的测试结果。

下表描述 TFTP 监视器的其他元素。

表 *152. TFTP* 监视器元素	
元素	**描述**
$bytesPerSec* (BytesPerSec)	平均每秒传输的字节数。
$bytesTransferred* (BytesTransferred)	上载或下载的字节数。
$checksum	下载的数据的校验和值。它由监视器生成，并提供用于通过使用规则文件进一步处理。
$command* (TftpCommand)	监视器发出的 TFTP 命令（GET 或 PUT）。
$localFile* (TftpLocalFile)	存储在本地主机上的文件的完整路径名。从配置文件中获取此元素。
$localIP	监视器配置为使用的本地 IP 地址。此元素在仅有一个接口的系统上可能为空白。
$lookupTime* (LookupTime)	获取主机服务器 IP 地址所花的时间。
$remoteFile* (TftpRemoteFile)	存储在远程主机（FTP 服务器）上的文件的完整路径名。从配置文件中获取此元素。
(TransferTime)	传输文件所需的时间。
(TftpConnection)	监视器传输文件所采用的格式。它是 OCTET（8 位）或 NETASCII。

状态消息

使用 IBM Application Performance Management 时，TFTP 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

表 *153. TFTP* 监视器状态消息	
消息	**描述**
OK	TFTP 请求已成功。
FAILED: connect failed	监视器无法连接到服务器。请检查服务器是否正在运行。
FAILED: internal tftp monitor error	监视器存在问题，可能是由于内存不足所导致。

表 *153. TFTP* 监视器状态消息 *(续)*	
消息	**描述**
FAILED: A send/wait timed out	TFTP 请求失败。网络可能存在问题。
FAILED: An unspecific error condition. The transfer should be aborted	
FAILED: Received a short or malformed packet	
FAILED: local file open/read/write failed	
FAILED: unrecognized status from transfer attempt	

示例

通过将文件 $ISHOME/etc/testfiles/upload.txt 上载到 /ism/test/upload_result.txt 来测试 TFTP 服务器 tftp.mycompany.com 的可用性。使用 netacii 方式按照 20 分钟时间间隔上载文件

使用以下条件对服务级别进行分类:

· 如果上载不成功，那么服务级别为 Failed

· 如果传输的总时间大于 10 秒，那么服务级别为 Marginal

· 否则，服务级别为 Good

创建 TFTP 监视器概要文件元素并设置配置，如下表中所示。

表 *154. TFTP* 概要文件元素示例	
配置字段	**值**
server	tftp.mycompany.com
localfile	$ISHOME/etc/ism/testfiles/upload.txt
remotefile	/ism/test/upload_result.txt
描述	TFTP test
动态	Selected
command	PUT
transfermode	NETASCII
poll	1200

statement	`If (Message != OK) then status Failed`
	`else if (TotalTime > 10) then status Marginal`
	`else status Good`

TRANSX 监视器

TRANSX 监视器通过运行一系列活动（使用其他因特网服务监视器执行）来模拟实际因特网用户的操作。

例如，配置 TRANSX 以使用 HTTP 监视器访问 Web 站点的页面，下载某些文件，使用 POP3 和 SMTP 监视器进行发送或接收。

下表列出 TRANSX 监视器文件。

表 *155. TRANSX* 监视器文件

监视器文件	名称或位置
监视器可执行文件	nco_m_transaction
属性文件	$ISHOME/etc/props/transx.props
规则文件	$ISHOME/etc/rules/transx.rules
日志文件	$ISHOME/log/transx.log

属性

表 *156. TRANSX* 监视器属性

属性名称	属性参数	描述
CompleteTransax	0|1	指定即使在步骤失败时事务也继续。 · 0 - 已禁用（不继续） · 1 - 已启用（继续）
DetailedTimings	0|1	指定 TRANSX 监视器生成包含每个步骤的细颗粒度计时的数据日志。数据日志中生成的细粒度计时为预设，并且无法修改。 · 0 - 已禁用 · 1 - 已启用
MultipleEvents	0|1	指定监视器是否针对事务结果生成多个事件： · 0 - 已禁用（监视器仅生成一个包含所有步骤的结果以及摘要结果的事件） · 1 - 已启用（监视器针对事务中的每个步骤生成一个事件以及最终摘要事件）
StepPause	整数	指定执行每个事务步骤之间的暂停长度，以秒为单位。 暂停长度不影响事务的 $totalTime 元素的值。$totalTime 表示事务的 $stepXTime 元素的总和。 缺省值：0

下表描述特定于 SMTP 监视器的属性。

用于配置 TRANSX 监视器的准则

TRANSX 监视器通过模拟构成典型用户体验的一组活动来测试服务。该组活动称为事务，事务中的每个活动称为事务步骤。

TRANSX 概要文件元素用于定义事务。每个事务步骤配置因特网服务监视器（例如，HTTP）以针对此步骤执行操作。通过 TRANSX 概要文件元素的“步骤”选项卡上的“编辑”按钮，配置事务步骤。

配置步骤的方式与配置任何其他概要文件元素的方式相同。例如，调用 HTTP 监视器的步骤的配置详细信息可能包含 Head/Form 参数、代理服务器参数、正则表达式和服务级别分类。

当 TRANSX 监视器测试事务中的步骤时，它会记录所需的时间以及该步骤的服务级别。

注: 如果任何事务步骤使用需要 root 用户特权的其他监视器，例如，ICMP，那么 TRANSX 监视器需要 root 用户特权。

使用 HTTP 和 HTTPS 监视器处理动态内容

大量 Web 站点使用动态内容来提供功能，例如，基于会话或基于区域的交互。当与 TRANSX 监视器配合使用时，HTTP 和 HTTPS 监视器可以测试包含动态内容（例如在链接中嵌入的会话标识、区域代码或日期和时间，每次测试事务时这些值都有可能不同）的 Web 页面。

在以事务方式运行时，HTTP 和 HTTPS 监视器提供的动态内容功能支持识别在 URL 中嵌入或者在 HTML `form` 元素中定义的名称/值对格式的动态内容（称为 `dynamic page elements`），然后监视器在每个测试期间从页面抽取这些内容，从而确保每次测试事务时使用相应的动态值。

例如，考虑 Web 站点的“登录”页面 `http://www.mycompany.com/login`，其中包含了登录 Web 站点的链接。登录操作的链接 URL `http://www.mycompany.com/doLogin?sessionID=id` 包含“登录”事务的会话标识。在此示例中，名称/值对 `sessionID=id` 是动态页面元素；在每次访问“登录”页面时，`id` 的值发生更改。要测试作为事务一部分的“登录”页面，请将事务配置为在每次测试事务时获取并使用 `sessionID` 的值，并将该值插入登录操作 URL。

TRANSX 监视器将动态页面元素从一个事务步骤传递至下一个事务步骤。在 Web 站点“登录”示例中，事务的第一个步骤是访问“登录”页面以获取会话标识，然后将该会话标识传递到第二个步骤，由该步骤提交包含此标识的登录请求。在这些步骤中执行的操作包括:

1. 访问包含动态页面元素的页面，例如 `http://www.mycompany.com/login`
2. 使用动态页面元素提交操作，例如 `http://www.mycompany.com/doLogin?sessionID=@@030671`

在识别事务步骤中的动态页面元素时，会将该动态页面元素传递到下一个事务步骤，从而将名称/值对插入到其请求中。此元素将传递到每个后续的 HTTP 或 HTTPS 事务，直到您显式移除该元素为止。

添加和移除动态页面

每个 HTTP 或 HTTPS 事务步骤都由返回 HTML 页面的请求构成。在运行包含动态页面元素的事务步骤时，监视器解析 HTML 页面以查找每个元素的名称/值对并将其传递到下一个事务步骤，这会将它们插入到其 HTTP 或 HTTPS 请求。

要指定步骤使用动态页面元素，请将元素的参数类型设置为 DYNAMIC。然后，指定要抽取的每个动态元素并传递到后续步骤。通过输入其名称（例如，`sessionID`）并选择“添加至”作为值来标识每个动态元素。“添加至”指示要将元素传递到后续步骤。

注: 要获取动态页面元素的名称，请查看其所在的页面的 HTML 源。

动态页面元素从一个事务步骤传递至下一个事务步骤。如果不再需要将页面元素传递到后续步骤，请将元素的值设置为“从中移除”。手动更新后续事务步骤可确保处理正确的页面元素。

添加和移除动态页面元素时请使用以下准则:

- 如果某个步骤不使用任何动态页面元素，请勿选择 DYNAMIC 作为参数类型。
- 如果某个步骤需要动态页面元素，那么检索显示动态元素的页面的步骤必须指定元素名称和值“添加至”。
- 如果某个步骤不需要从先前步骤传递的动态元素，请将先前步骤的值设置为 Remove From。

GET 和 POST

在 GET 方法中，所有动态页面元素都将自动插入到请求 URL。在 POST 方法中，您必须在“参数”选项卡上将每个动态元素指定为 FORM 参数。

运行事务步骤时，监视器将自动为每个表单插入动态值。

创建事务

通过使用 Internet Service Monitoring 用户界面创建 TRANSX 概要文件元素和事务步骤来定义事务。有关更多信息，请参阅第 377 页的『创建事务』。

配置 TANSX 监视器服务测试

表 *157. TRANSX* 监视器配置	
字段	**描述**
transxname	事务名称。
description	提供关于该元素描述性信息的文本字段。
poll	轮询的间隔时间（秒）。 缺省值：300

注: 使用一系列 Web 浏览、文件下载和已发送的电子邮件消息来监视 Web 站点的可用性。

1. 创建 TRANSX 概要文件元素。
2. 创建 HTTP 事务步骤以监视 Web 站点的可用性。
3. 创建 FTP 事务步骤监视文件下载。
4. 创建 POP3 或 SMTP 事务步骤监视电子邮件。

请参阅每个监视器文档以获取进一步信息。

监视器元素

TRANSX 监视器生成包含每个事务的结果的事件。这些事件包含整个事务的结果以及单个事务步骤的结果。

但是，缺省情况下，监视器将所有事务和步骤结果都放在单个事件中，方法是使用可以将监视器配置为创建个别事件（对于每个事务步骤）和摘要事件（对于整个事务）的 MultipleEvents 属性。表 1 列出 TRANSX 摘要元素。

星号 (*) 指示的元素可用作属性。属性名称显示在方括号中。缺少星号指示无等效属性。在方括号中显示但没有元素的属性指示它们仅可用作属性，无等效元素。

表 *158. TRANSX* 摘要监视器元素	
元素	**描述**
$numberOfSteps*(NumberOfSteps)	事务中的步骤数。
$stepDescriptions	以竖线字符 (|) 分隔的每个步骤的描述的列表。
$stepTimes*(Step*1 to 10*TotalTime)	每个步骤 (1 - 10) 返回的计时数据。
$stepUnits	每个步骤的单位的列表，通常为秒，用竖线字符 (|) 分隔每一个。
(TransName)	配置事务时指定的事务的名称。
(TransStepDescription)	配置步骤时指定的事务步骤的描述。

状态消息

在使用 IBM Application Performance Management 时，TRANSX 监视器在 ResultMessage 属性中提供状态消息。这些消息指示测试的结果。

下表描述状态消息。

表 *159. TRANSX* 监视器状态消息	
Message	**描述**
`Successfully completed transaction`	事务已成功完成。
`Error in transaction`	其中一个事务步骤失败。
`Service Level Failed, ending transactionService Level Failed`	其中一个步骤的服务级别返回失败响应，这导致事务停止。

创建事务
可以通过使用 Internet Service Monitoring 代理程序 用户界面创建 TRANSX 概要文件元素和事务步骤来定义事务。

过程

要使用界面创建事务：

1. 单击**系统配置**图标。在该图标下单击**代理程序配置**。
   此时将打开代理程序配置窗口。
2. 单击 **ISM** 选项卡以配置 Internet Service Monitoring 代理程序。
3. 单击 **(+)** 加号图标以创建新概要文件。输入**概要文件名称**和**描述**。
4. 单击**下一步**。
5. 单击监视器下拉列表中的 **TRANSX** 监视器以选择 TRANSX 监视器。
6. 单击**下一步**。
7. 输入强制参数。
8. 在**高级**选项卡中，指定**轮询时间间隔**。
9. 单击“步骤”选项卡上的 **(+)** 加号图标。
10. 单击需要从监视器下拉列表中选择的监视器。
11. 单击**选择**以配置事务步骤。
    a) 按照与先前输入相同的方式指定强制和可选参数，以配置概要文件元素。
    b) 如果为 HTTP 或 HTTPS 监视器创建动态步骤，请设置“参数”选项卡上的“名称”和“值”对，然后选择 DYNAMIC 作为参数类型。
12. 单击**添加**。
13. 单击步骤网格上的刷新图标。
14. 对每个其他事务帮助重复步骤 1 到 13。
15. 单击**添加**以完成。
16. 单击**完成**以保存。

结果

注: 要在每个事务步骤之间指定暂停，请使用 `StepPause` 属性。

Internet Service Monitoring 中的 SSL 设置
Internet Service Monitoring 通过各种监视器使用 OpenSSL 来安全地与通常远程的因特网服务进行通信，例如，HTTPS 监视器与受保护 HTTPD 进行通信。Internet Service Monitoring 代理程序 还在监视器与 Databridge 之间以及 Internet Service Monitoring 代理程序 (KIS) 与 Databridge 之间使用 OpenSSL。请在 `SSLCipherSuite` 属性中指定应用程序使用的密码套件。

Databridge 应配置为安全地与监视器和 Internet Service Monitoring 代理程序 进行通信，以便每个监视器共享与 Databridge 相关的常用属性集来管理与 Databridge 的安全通信。某些监视器还共享类似但不同的相关属性集来管理与其各自的受测试因特网服务的安全通信。

以下监视器支持监视受保护因特网服务：

· HTTPS

- IMAP4
- POP3
- SMTP

这些监视器会使用证书。所有证书都以 X509 格式通过隐私增强邮件 .Pem 文件形式存储在 `$ISMHOME/certificates` 中。Databridge 的证书也存储在同一位置。为此，以下属性由所有监视器、Databridge 和 Internet Service Monitoring 代理程序 共享：

- SSLTrustStore（缺省值：$ISMHOME/certificates/trust.pem）
- SSLTrustStorePath（缺省值：$ISMHOME/certificates/）

由于监视器与 Databridge 之间以及所选监视器与其受保护因特网服务之间的所有通信都构建在同一版本的 OpenSSL 上，因此它们具有相同特征。例如，Internet Service Monitoring 可以提供的最高安全性级别是底层 OpenSSL 提供的最高级别的函数。所提供的最低安全性级别同样取决于底层 OpenSSL。

如果已更新 Internet Service Monitoring 代理程序，并且该更新包括对底层 OpenSSL 的更新，那么受监视因特网服务可能会受影响。例如：

1. Internet Service Monitoring 代理程序 V7.x.1 中的 HTTPS 监视器监视的是受保护 HTTPD 服务器。
2. 应用新版本的 Internet Service Monitoring 代理程序，其中包含已更新的 OpenSSL 版本，这意味着 HTTPS 监视器现在是 V7.x.2。
3. 您注意到 HTTPS 监视器现在无法监视受保护 HTTPD。

HTTPD 服务器的安全性级别小于新近更新的 Internet Service Monitoring 代理程序 V7.x.2 提供的最低级别。即使 HTTPS 监视器的配置尚未更改，其行为也已更改，因为它取决于底层 OpenSSL 层。较新的 Internet Service Monitoring 代理程序/HTTPS 监视器/OpenSSL 组合比旧组合更安全，您现在需要提升远程 HTTPD 服务器的安全性级别。

监视受保护因特网服务使您进退两难。Internet Service Monitoring 代理程序 的安全性级别应低至可以监视弱保护的因特网服务，还是应与当前建议的最低配置同样高？如果选择前者，那么弱化的 Internet Service Monitoring 代理程序 可能会在两端均损害安全性。

所有监视器都使用同一版本的 OpenSSL。所有这些监视器都共享常用监视器属性集来配置底层 OpenSSL，下表中对这些属性进行了描述。

表 *160. 与 OpenSSL* 相关的监视器属性

属性名称	**属性参数**	**描述**
`SSLCipherSuite`	字符串	指定要用于监视器与受监视因特网服务之间的 SSL 操作的密码套件。此属性的值应采用 OpenSSL 建议的形式。 缺省值：`AES:3DES:DES:!EXP:!DHE:!EDH`
`SSLDisableSSLv2`	0|1	确定在监视受保护因特网服务时要建立哪种类型的安全连接。 `0` - 允许 SSLv2 `1` - 不允许 SSLv2 缺省值：`1`（不允许 SSLv2）。
`SSLDisableSSLv3`	0|1	确定在监视受保护因特网服务时要建立哪种类型的安全连接。 `0` - 允许 SSLv3 `1` - 不允许 SSLv3 缺省值：`1`（不允许 SSLv3）。

表 160. 与 OpenSSL 相关的监视器属性 (续)

属性名称	属性参数	描述
SSLDisableTLS	0|1	确定在监视受保护因特网服务时要建立哪种类型的安全连接。 0 - 允许 TLSv1.0 1 - 不允许 TLSv1.0 缺省值：0（允许 TLSv1.0）。
SSLDisableTLS11	0|1	确定在监视受保护因特网服务时要建立哪种类型的安全连接。 0 - 允许 TLSv1.1 1 - 不允许 TLSv1.1 缺省值：0（允许 TLSv1.1）。
SSLDisableTLS12	0|1	确定在监视受保护因特网服务时要建立哪种类型的安全连接。 0 - 允许 TLSv1.2 1 - 不允许 TLSv1.2 缺省值：0（允许 TLSv1.2）。
SSLCertificateFile	字符串	监视器使用的公用数字证书文件的路径和文件名。当监视器尝试设置与因特网服务的安全连接时，后者可能会选择性地请求监视器提供其客户机端证书，从而使因特网服务能够验证监视器或客户机（客户机端证书验证）。 证书必须为隐私增强邮件 (PEM) 格式。 对于 HTTPS 监视器，可以在创建时为每个 HTTPS 元素指定该值。但是，如果 HTTPS 监视器将对所有元素使用同一证书，那么会使用 HTTPS.props 文件中的值。 对于 IMAP、LDAP、POP3、SIP、SMTP 和 SOAP 监视器，将在监视器范围内设置该值。 如果路径不是绝对路径，那么监视器会相对于工作目录（$ISMHOME/certificates）对其进行解释。 缺省值：“”
SSLKeyFile	字符串	包含监视器使用的专用密钥的文件的路径和文件名。监视器使用此文件来解密其发送到其他监视器的消息。接收方使用监视器的公用数字证书来解密消息。 缺省值：monitoryKey.pem
SSLKeyPassword	字符串	用于对 SSL 专用密钥进行加密的密码。 缺省值：“”

表 160. 与 OpenSSL 相关的监视器属性 (续)

属性名称	属性参数	描述
SSLTrustStoreFile	字符串	用于将受监视因特网服务的所有 X509 公用证书存储为合并列表的文件的标准名称。 已撤销的证书 (CRL) 也在此处存储为合并列表。 Databridge 也可将其公用证书存储在此处。此属性出现在 bridge.props 文件中。 证书以隐私增强邮件 (PEM) 格式进行存储。请使用从 http://www.openssl.org 提供的 OpenSSL 软件将以其他格式获取的证书转换为 PEM 格式。 缺省值：**“$ISMHOME/certificates/trust.pem”**
SSLTrustStorePath	字符串	包含受监视安全因特网服务的 X509 证书的 .pem 文件的位置。 已撤销的证书 (CRL) 也存储在此处。 Databridge 也可将其公用证书存储在此处。此属性出现在 bridge.props 文件中。 如果将新证书添加到此目录，请运行 openssl rehash 命令以扫描目录并计算每个证书的散列。 如果同时使用了 SSLTrustStoreFile 和 SSLTrustStorePath 属性，那么 OpenSSL 会使用这两个属性来查找可信证书。 缺省值：“$ISMHOME/certificates/”
VerifyCertificate Preference	0|1	启用或禁用根据证书撤销列表 (CRL) 对受监视因特网服务提供的证书进行验证。 缺省值：0 - 已禁用

密码套件

可供 Internet Service Monitoring 使用的密码套件是 OpenSSL 允许的密码套件的子集。OpenSSL 允许的密码套件集随时间更改。随着新漏洞的发现和最佳做法的演进，OpenSSL 可能会完全限制或移除对特定或通用类型的密码套件的访问。由于这些更高版本的 OpenSSL 包含在更高版本的 ISM 中，因此存在流动效应，这可能会影响监视器的配置和操作。

请使用 SSLCipherSuite 监视器范围的属性来通过关键字从所有可用密码套件中指定监视器允许的密码套件。要指定多个套件，请使用关键字的冒号分隔列表。例如，缺省 SSLCipherSuite 属性为 AES:3DES:DES:!EXP:!DHE:!EDH。此选择意味着允许包含 AES、3DES 和 DES 的密码套件，但是排除使用 EXP（导出（短密钥长度））、DHE（Diffie Hellman 交换）或 EDH (Ephemeral Diffie Hellman) 密钥交换的任何密码套件。此外，在监视器与因特网服务之间进行安全连接后，首先会使用 AES，再使用 3DES，然后根据情况使用 DES。Internet Service Monitoring 代理程序 的密码套件列表的语法与 OpenSSL 的语法相同。

要为监视器选择正确的密码套件集，请考虑底层 OpenSSL 支持的密码套件、受监视因特网服务支持的密码范围以及组织的安全标准。您可能无法监视安全性级别低于 Internet Service Monitoring 或 OpenSSL 将容忍的安全性级别的安全外部站点。在某些情况下，曾经能够监视因特网服务的监视器在升级 Internet Service Monitoring 后可能会失败，因为安全性级别不兼容。

下表列出与 SSLCiperSuite 的缺省值 AES:3DES:DES:!EXP:!DHE:!EDH 等效的密码套件的子集及其属性。在该表中，您将看到以下术语：

· 密码套件名称：使用从关键字构造的名称来描述密码套件。

· 协议：描述支持的协议版本。

· 密钥交换：描述用于加密和解密的密钥交换系统。

· 加密与密钥长度：描述所使用的加密算法类型和所使用的密钥长度（以位为单位）。

· MAC：用于确保数据未被篡改的消息认证代码。

表 *161.* 密码套件名称和属性值 *AES:3DES:DES:!EXP:!DHE:!EDH*

密码套件名称	协议	密钥交换	认证	加密与密钥长度	消息认证代码
ECDHE-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH	RSA	AESGCM(256)	AEAD
ECDHE-ECDSA-AES256-GCM-SHA384	TLSv1.2	ECDH	ECDSA	AESGCM(256)	AEAD
ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
ECDHE-ECDSA-AES256-SHA384	TLSv1.2	ECDH	ECDSA	AES(256)	SHA384
ECDHE-RSA-AES256-SHA	SSLv3	ECDH	RSA	AES(256)	SHA1
ECDHE-ECDSA-AES256-SHA	SSLv3	ECDH	ECDSA	AES(256)	SHA1
SRP-DSS-AES-256-CBC-SHA	SSLv3	SRP	DSS	AES(256)	SHA1
SRP-RSA-AES-256-CBC-SHA	SSLv3	SRP	RSA	AES(256)	SHA1
SRP-AES-256-CBC-SHA	SSLv3	SRP	SRP	AES(256)	SHA1
DH-DSS-AES256-GCM-SHA384	TLSv1.2	DH/DSS	DH	AESGCM(256)	AEAD
...后面另跟 61 行					

下表列出与 SSLCiperSuite 的值 AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3 等效的密码套件的子集及其属性。某些协议现已消除，并且整体密码套件集已减小 71 - 31。

表 *162.* 密码套件名称和属性值 ***AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3***

密码套件名称	协议	密钥交换	认证	加密与密钥长度	消息认证代码
ECDHE-ECDSA-AES256-GCM-SHA384	TLSv1.2	ECDH	ECDSA	AESGCM(256)	AEAD
ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
ECDHE-ECDSA-AES256-SHA384	TLSv1.2	ECDH	ECDSA	AES(256)	SHA384
DH-DSS-AES256-GCM-SHA384	TLSv1.2	DH/DSS	DH	AESGCM(256)	AEAD
DH-RSA-AES256-GCM-SHA384	TLSv1.2	DH/RSA	DH	AESGCM(256)	AEAD
DH-RSA-AES256-SHA256	TLSv1.2	DH/RSA	DH	AES(256)	SHA256
DH-DSS-AES256-SHA256	TLSv1.2	DH/DSS	DH	AES(256)	SHA256

表 162. 密码套件名称和属性值 ***AES:3DES:DES:!EXP:!DHE:!EDH:!SSLv2:!SSLv3*** *(续)*					
密码套件名称	**协议**	**密钥交换**	**认证**	**加密与密钥长度**	**消息认证代码**
ADH-AES256-GCM-SHA384	TLSv1.2	DH	无	AESGCM(256)	AEAD
ADH-AES256-SHA256	TLSv1.2	DH	无	AES(256)	SHA256
ECDH-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH/RSA	ECDH	AESGCM(256)	AEAD
...后面另跟 21 行					

减少漏洞

在未来发行版中，由于漏洞，缺省情况下将禁用 DHE 和 EDH 密码。对于 Internet Service Monitoring 代理程序 的先前版本，您可能需要在所有监视器中禁用 DHE 和 EDH 密码。要禁用 DHE 和 EDH 密码，请更新 `SSLCipherSuite` 和 `BridgeSSLCipherSet` 监视器属性。

例如，要在 HTTPS 监视器中禁用 DHE 和 EDH 密码，请将 `https.props` 文件更新为包含以下属性：

```
SSLCipherSuite: AES:3DES:DES:!DES-CBC-SHA:!EXP:!DHE:!EDH
BridgeSSLCipherSet: AES:3DES:DES:!DES-CBC-SHA:!EXP:!DHE:!ED
```

确保验证此配置更改不会导致任何兼容性问题。如果在应用此修订后更改缺省设置，那么您可能会自身遭受安全漏洞攻击。您应复审整个环境来确定您已启用 TLS 中使用的 Diffie-Hellman 密钥交换协议的其他区域，并采取相应的缓解和补救措施。

协议选择

您可以从一系列历史和当前安全通信协议中进行选择。可以使用布尔监视器属性集分别选择这些协议：

- SSLDisableSSLv2
- SSLDisableSSLv3
- SSLDisableTLS
- SSLDisableTLS11
- SSLDisableTLS12
- BridgeSSLDisableSSLv2
- BridgeSSLDisableSSLv3

您应禁用 SSLv2 和 SSLv3。这些协议已受损并具有多个已知漏洞。它们在缺省情况下已禁用，并仅提供用于旧用途。

缺省情况下 Internet Service Monitoring 会启用 TLS。如果您知道您监视的因特网服务使用的不是 TLS 1.0 并已升级到 TLS 1.1 或 TLS 1.2，那么应在 Internet Service Monitoring 中禁用未使用的协议。

Databridge 组件与 Internet Service Monitoring 代理程序以及各个监视器进行通信。缺省情况下，此通信已加密，并且 TLS 为首选协议。

密钥信任库和证书

Internet Service Monitoring 将其证书存储在用户定义的位置中的用户定义的文件内。所有证书都必须以隐私增强邮件 (PEM) 格式进行存储。请确保将从其他组织获取的公用证书转换为 PEM 格式。转换软件可在 http://www.openssl.org 中获取。

使用 `SSLTrustStoreFile` 属性指定的可信证书在文件中存储为合并列表。

理想做法是将证书撤销列表 (CRL) 存储在可根据其验证证书的信任库中。认证中心具有适当的系统来生成已撤销证书的列表，并具有适当的分发系统以使其公开可用。然后，如果证书受损，那么会将其撤销。

Databridge 安全设置

所有监视器都与 Databridge 进行通信，因此所有监视器都具有应设置为管理监视器与 Databridge 之间的通信的常用属性集。缺省情况下，通信已加密。缺省加密协议为 TLS。与监视器属性不同，没有机制来控制启用还是禁用特定版本的 TLS。所有监视器都应具有相同的 Databridge 属性值，否则将会出现通信问题。同样，Databridge .props 文件中设置的属性应与监视器中的属性一致。Databridge 还与具有其自己的 .props 文件的 Internet Service Monitoring 代理程序进行通信。代理程序 .props 中的某些值与 Databridge 相关，并且与监视器一样，必须具有与 Databridge `.props` 文件中的值一致的值。

表 *163. 与 OpenSSL 相关的 Databridge 属性*

属性名称	属性参数	描述
BridgeSSLEncryption	0|1	确定与 Databridge 的通信是否已加密。这涵盖从 Databridge 到监视器和 Internet Service Monitoring 代理程序的所有通信。 0 - 未加密 1 - 已加密 **限制:** 请在 Internet Service Monitoring 代理程序、所有监视器和 Databridge 上设置相同值。
BridgeSSLCipherSet	字符串	指定要用于与 Databridge 之间的双向 SSL 操作的密码套件。此属性的值应采用 OpenSSL 建议的形式。 **限制:** 请在 Internet Service Monitoring 代理程序、代理程序、所有监视器和 Databridge 上设置相同值。 缺省值: `AES:3DES:DES:!EXP:!DHE:!EDH`
BridgeSSLDisableSSLv2	0|1	确定要与 Databridge 之间双向建立哪种类型的安全连接。 0 - 允许 SSLv2 和 SSLv3 1 - 不允许 SSLv2 **限制:** 请在 Internet Service Monitoring 代理程序、所有监视器和 Databridge 上设置相同值。 缺省值: 1（不允许 SSLv2）。
BridgeSSLDisableSSLv3	0|1	确定要与 Databridge 之间双向建立哪种类型的安全连接。 0 - 允许 SSLv3 1 - 不允许 SSLv3 **限制:** 请在 Internet Service Monitoring 代理程序、所有监视器和 Databridge 上设置相同值。 缺省值: 1（不允许 SSLv3）。
BridgeSSLCertificateFile	字符串	Databridge SSL 数字证书的路径和文件名。 缺省值: `$ISMHOME/certificates/bridgeCert.pem`
BridgeSSLKeyFile	字符串	Databridge SSL 专用密钥文件的路径和文件名。 缺省值: `$ISMHOME/certificates/bridgeKey.pem`

表 163. 与 OpenSSL 相关的 Databridge 属性 (续)

属性名称	属性参数	描述
BridgeSSLKeyPassword	字符串	用于加密 Databridge SSL 专用密钥的密码。 缺省值：**Tivoli**
BridgeSSLTrustStore	字符串	用于认证的可信证书文件的路径和文件名。仅当使用 **BridgeSSLAuthenticatePeer** 属性时，才需要此属性。 缺省值：`$ISMHOME/certificates/trust.pem` 如果要在监视器与 Databridge 之间或 Databridge 与代理程序之间配置 SSL 认证，请将 `BridgeSSLAuthenticatePeer` 设置为 `1` 并重新启动 Databridge。此操作从服务器对证书进行认证。可以将证书存储在 `SSLTrustStoreFile` 和 `SSLTrustStorePath` 中。 缺省值： · `SSLTrustStoreFile`，`$ISMHOME/certificates/trust.pem` · `SSLTrustStorePath`，`$ISMHOME/certificates/` 要添加新证书，请完成以下步骤之一： · 将证书添加到 `SSLTrustStoreFile` 文本文件中的列表末尾 · 将新证书添加到 `SSLTrustStorePath` 目录，并运行 OpenSSL **c_rehash certificate_dir** 命令以散列证书
SSLTrustStoreFile	字符串	此属性由安全监视器和 Databridge 使用。有关更多信息，请参阅第 378 页的表 160。
SSLTrustStorePath	字符串	此属性由安全监视器和 Databridge 使用。有关更多信息，请参阅第 378 页的表 160。
BridgeSSLAuthenticatePeer	0|1	指定 Databridge 是否应与其他 Internet Service Monitoring 组件进行交叉认证。 `0` - 已禁用 `1` - 已启用 如果监视器访问 Databridge，那么它必须向 Databridge 认证，并且 Databridge 必须向监视器认证。 如果 Internet Service Monitoring 代理程序访问 Databridge，那么它必须向 Databridge 认证，并且 Databridge 必须向该代理程序认证。 Databridge 的证书存储在 BridgeSSLTrustStore 中。 缺省值：`0` - 已禁用

Internet Service Monitoring 代理程序属性

Internet Service Monitoring 代理程序 具有其自己的属性文件，其中包含一组安全属性和设置。代理程序属性文件不与监视器进行通信，但它会与 Databridge 进行通信，因此代理程序的 .props 文件中的安全设置管理代理程序与 Databridge 之间的通信。

在 Windows 系统上配置代理程序

您可以使用 **IBM Performance Management** 窗口在 Windows 系统上配置代理程序。

过程

在用户系统上配置 Internet Service Monitoring 代理程序 代理程序，如下所示。

要在用户系统上手动配置 Internet Service Monitoring 代理程序，请执行以下操作：

1. 在 **IBM Performance Management** 仪表板上，单击**系统配置**下列出的**系统配置 > 代理程序配置**。
2. 单击 **ISM** 选项卡以打开 Internet Service Monitoring 代理程序仪表板。

配置 Databridge

配置 Databridge 涉及为 Databridge 设置用于控制其操作（例如连接组件模块和因特网服务监视器）的属性。

操作和配置

Databridge 及其组件模块通过属性文件进行配置。

属性确定 Databridge 及其组件模块的操作，该操作将测试结果发送到 IBM Cloud Application Performance Management 以在 Internet Service Monitoring 代理程序仪表板上进行报告。

配置 *Databridge*

必须将 Databridge 配置为从 Internet Service Monitors 接收数据并将该数据转发到其组件模块以进一步处理。

下表列出与 Databridge 关联的文件。在相应部分中对**属性文件**、**存储转发文件**和**日志文件**进行了更详细的描述。

表 *164. Databridge* 文件及其位置

Databridge 文件	**位置或名称**
可执行文件	$ISHOME/platform/arch/bin/nco_m_bridge
属性文件	$ISHOME/etc/props/bridge.props
存储转发文件	名称和位置由 bridge.props 文件中的属性指定。缺省名称和位置为 $ISHOME/var/sm_bridge.saf
日志文件	$ISHOME/log/bridge.log
错误日志文件	$ISHOME/log/bridge.err

存储转发文件

如果 Databridge 无法将数据转发到 Netcool/OMNIbus，那么它会将其通常发送的所有数据存储在存储转发 (SAF) 文件中。当 Netcool/OMNIbus 再次变为可用时，它会处理 SAF 文件中存储的所有事件。

Databridge 属性文件中的 QFile 和 QSize 属性确定存储转发处理的名称、位置和操作。

日志文件

Databridge 将有关其操作的每日消息发送到消息日志文件。缺省情况下，此文件的名称为 $ISHOME/log/bridge.log。它在午夜 12 点进行更新。Databridge 属性 MsgDailyLog 和 MsgTimeLog 控制消息日志记录的操作。

启动 *Databridge*
使用 Windows 服务控制台启动 Databridge。

过程

注: 如果 ObjectServer 模块连接到 Databridge，请确保其目标系统在启动 Databridge 之前正在运行。如果任何 Databridge 模块无法正确初始化，那么 Databridge 将不会启动。

1. 从 Windows 桌面中，单击**开始 > 管理工具 > 服务**。
2. 从服务列表中，选择名为 NCO BRIDGE Internet Service Monitor 的服务，然后单击菜单中的**启动**。

连接模块
Databridge 属性文件定义要连接到 Databridge 的模块。

关于此任务
每个 Module n `SharedLib` 和 Module n `PropFile` 属性对定义一个模块的连接。系统从 Module0 开始，按照定义顺序装入模块。

过程

1. 将个别模块连接到 Databridge:
   a) 在 Databridge 属性文件中，识别下一个可用的 Module n `SharedLib` 和 Module n `PropFile` 属性对。
   b) 将 Module n `SharedLib` 设置为模块的共享库（其二进制实施）的名称。
   c) 将 Module n `PropFile` 设置为模块的属性文件的完整路径。

   在此示例中，第 1 行和第 2 行连接 ObjectServer 模块，第 3 行和第 4 行连接 Datalog 模块，第 5 和第 6 行连接 IBM Application Performance Management (`pipe`) 模块。Datalog 模块不具有属性文件，因此属性文件条目的值为 `""`。
2. 禁用模块:
   a) 将对应的 Module n `SharedLib` 属性设置为 `"NONE"`，并将 Module n `PropFile` 属性设置为 `""`。此外，还会忽略值大于 n 的所有其他模块。

连接监视器
因特网服务监视器通过 TCP 连接到 Databridge。每个监视器具有用于配置与 Databridge 的连接的属性集。

关于此任务
要将监视器连接到 Databridge，请将监视器属性文件中定义的 `BridgePort` 属性值设置为 Databridge 属性文件中定义的 `SocketPort` 属性值。每个监视器的 `BridgePort` 属性和 Databridge 的 `SocketPort` 属性的缺省值为 `9510`。

Databridge 支持对其从监视器接收的测试结果进行 SSL 加密。要加密监视器的测试结果，请将监视器属性文件中定义的 `BridgeSSL` 属性值设置为 Databridge 属性文件中定义的 `BridgeSSL` 属性值。要加密所有监视器的测试结果，所有监视器都必须具有相同的 `BridgeSSL` 属性。

配置 *Databridge* 模块
Databridge 将测试结果定向到 Internet Service Monitoring 代理程序。监视代理程序将此数据转换为所需格式并将其分发到 IBM Application Performance Management 服务器。通过 Databridge 模块和 Internet Service Monitoring 代理程序各自的属性文件对其进行配置。

通过修改模块属性文件中定义的属性值来配置 Databridge 的操作。

模块属性文件名为 `pipe_module.props`。此文件位于 `$ISHOME/etc/props/` 目录中。

下表列出可用于模块的属性。如果对属性进行了更改，请重新启动 Databridge 以使这些更改生效。

表 *165. Databridge* 模块属性		
属性名称	**Type**	**描述**
TEMAHOST	字符串	运行监视代理程序的主机的名称。 缺省值：localhost
TEMAPORT	整数	主机使用的端口号。 缺省值：9520

通过修改监视代理程序属性文件中定义的属性值来配置 Internet Service Monitoring 代理程序的操作。

监视代理程序属性文件名为 kisagent.props。此文件位于 $ISMHOME/etc/props/ 目录中。

下表列出可用于监视代理程序的属性。

表 *166.* 监视代理程序属性		
属性名称	**Type**	**描述**
TEMAPORT	整数	主机使用的端口号。此端口号必须与模块属性文件中列出的 TEMAPORT 属性的端口号相同。 缺省值：9520
ObsoleteDuration	整数	从监视代理程序的内存中删除任何未更新的数据之前所要经过的时间（以秒为单位）。例如，在已停止概要文件元素或已发生网络故障时，数据可能未更新。 **注:** 请勿将 ObsoleteDuration 时间设置为小于轮询时间间隔的值，因为这会导致在轮询时间间隔期间丢失数据。 缺省值：900
AggDuration	整数	监视代理程序阻止数据汇总并在代理程序仪表板上进行报告之前所要经过的时间（以秒为单位）。系统会从监视代理程序的内存中删除任何晚于指定时间的数据。 系统通过将开始时间和当前时间之间的时间间隔与汇总持续时间进行比较来计算旧数据。如果该时间间隔大于汇总持续时间，那么会移除 10% 的旧数据，并将开始时间增大该时间间隔的 1/10。监视代理程序每 5 分钟计算一次此属性。 缺省值：3600
ManageServices	0|1	在启动或停止 Internet Service Monitoring 代理程序后启动和停止所有监视器及 Databridge。1 表示已启用，0 表示已禁用。 缺省值：1

安装 Internet Service Monitoring 时，将会创建 Internet Service Monitoring 代理程序与 Databridge 模块之间的连接。

启用 Netcool/OMNIbus

遵循这些步骤以支持 Tivoli Netcool/OMNIbus 将事件从 Internet Service Monitoring 代理程序 发送到 Netcool/OMNIbus。

开始之前

确保安装了 IBM Tivoli Netcool/OMNIbus。

过程

完成以下步骤以启用 Netcool/OMNIbus：

1. 通过使用以下命令，停止 Internet Service Monitoring 代理程序：

```
$CANDLEHOME/bin/ism-agent.sh stop
```

2. 打开位于 $ISMHOME/etc/props 路径中的 bridge.props 文件并使用以下代码片段更新文件：

```
Module0SharedLib : "libSMModulePipe"
Module0PropFile : "$ISMHOME/etc/props/pipe_module.props"
Module1SharedLib : "libSMModuleObjectServer"
Module1PropFile : "$ISMHOME/etc/props/objectserver.props"
```

3. 修改位于 $ISMHOME/objectserver 路径中的 8.1.0 目录的许可权，如下所示：

```
cd  $ISMHOME/objectserver/
chmod -R 777 8.1.0
```

注: 通过使用 chmod -R 777 <file-name> 命令，修改 8.1.0 目录中所有文件的许可权。其中，<file-name> 是 8.1.0 目录中文件的名称。

4. 修改位于 $ISMHOME/objectserver/8.1.0/etc 路径中的 omni.dat 文件以配置 Netcool/OMNIbus 服务器地址。
5. 从以下位置运行 nco_igen：

```
cd $ISMHOME/objectserver/8.1.0/bin
./nco_igen
```

6. 通过使用以下命令，启动 Internet Service Monitoring 代理程序：

```
$CANDLEHOME/bin/ism-agent.sh start
```

7. 验证 Internet Service Monitoring 代理程序、Databridge 和所有监视器都处于正在运行状态。
   要检查 Databridge 和监视器的状态，请运行以下命令：

```
ps -aef|grep -i nco_*
```

   要检查 Internet Service Monitoring 代理程序 的状态，请运行以下命令：

```
ps -aef|grep -i kis
```

8. 使用 IBM Tivoli Netcool/OMNIbus 用户界面以验证 Databridge 将数据发送到 Netcool/OMNIbus 服务器。
   必须针对 Internet Service Monitoring 代理程序 在 IBM Application Performance Management 用户界面上显示数据。

配置 J2SE 监视

要从受监视的本地 Java 应用程序收集资源监视和诊断数据，必须配置 J2SE 数据收集器。

开始之前

安装其中一个受支持的 Java 运行时：

- Oracle Java Platform Standard Edition 7 (Java SE Development Kit 7)

  切记: 此 Java 运行时不支持配置有 HTTPS 协议的 J2SE 数据收集器映像。
- Oracle Java Platform Standard Edition 7 (Java SE Runtime Environment 7)

  切记: 此 Java 运行时不支持配置有 HTTPS 协议的 J2SE 数据收集器映像。
- Oracle Java Platform Standard Edition 8 (Java SE Development Kit 8)
- Oracle Java Platform Standard Edition 8 (Java SE Runtime Environment 8)
- IBM SDK Java Technology Edition V7
- IBM SDK Java Technology Edition V8

要点: 对于 Windows Server 2016，安装 JDK 8 更新 131 (Java SE Development Kit 8u131) 或 Java SE Development Kit 7 更新 80 (JDK 7u80)。

有关系统需求的更多信息，请参阅 J2SE 数据收集器的软件产品兼容性报告。

关于此任务

您可以在 Windows、Linux 和 AIX 系统上配置 J2SE 数据收集器。

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序和数据收集器版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

过程

1. 将以下文件从 APM 安装程序复制到某个目录:

   要点: 目录路径不得包含任何空格。

   - Windows 将 `gdc.zip` 文件从 APM 安装程序复制到某个目录并解压。
   - Linux AIX 将 `gdc-apd.tar.gz` 文件从 APM 安装程序复制到某个目录并解压。
   - Linux AIX 为用户提供对 `j2se_dc` 文件夹的读/写和执行许可权。提供执行许可权旨在运行该文件夹中的脚本和 JAR 文件。提供读/写许可权是因为该文件夹中将会生成深入诊断文件。
2. 在命令行上，转至 *DCHOME*`\.gdc\<toolkit_version>\bin`

   其中，*toolkit_verion* 表示:

   - 对于 V8.1.4.0 和更低版本，*toolkit_verion* 为 7.3.0.5.0。
   - 对于 V8.1.4.0.1 和更高版本，*toolkit_verion* 为 7.3.0.14.0。
3. 运行以下命令:

   Windows `config.bat`

   Linux AIX `config.sh`
4. 出现提示时，指定 Java 主目录的路径，然后按 **Enter** 键。
   例如，

   Windows `C:\Program Files\jre7`

   Linux AIX `/opt/ibm/java`
5. 针对您使用的代理程序版本完成以下步骤:

   - 对于 V8.1.4.0.2 和更低版本，请完成以下步骤:

     a. 出现提示时，输入应用程序主类的全名（限定名），然后按 **Enter** 键。主类是需要监视的应用程序的入口点。例如，`testapp.TemperatureConveter`

     b. 出现提示时，输入不同的应用程序别名，然后按 **Enter** 键。您在此处输入的名称用于在 APM 仪表板上创建实例名称。

Windows 在以下位置生成 dcstartup.bat 文件：*DCHOME*\.gdc*toolkit_verion*\runtime\j2se*application_alias*.*hostname*.*application_alias*。此文件是用于运行应用程序以及数据收集器的脚本。

Linux AIX 在以下位置生成 dcstartup.sh 文件：*DCHOME*/.gdc/*toolkit_verion*/runtime/j2se*application_alias*.*hostname*.*application_alias*。此文件是用于运行应用程序以及数据收集器的脚本。

- 对于 V8.1.4.0.3 到 V8.1.4.0.5，请完成以下步骤

a. 出现提示时，输入 Java 应用程序的主目录。例如，/root/J2seApp/

b. 从提供给您的列表中选择 Java 应用程序，然后退出。

- com.ibm.SampleApplication
- com.ibm.DBApplication
- com.ibm.SpringBootApplication

选择列表中提供的任何应用程序，或者提供 0 以选择不在列表中的任何其他应用程序。

1) 如果提供 0，请输入任何其他应用程序的主类的全名。例如，com.ibm.testApp.Main

2) 如果从提供的列表中选择任何选项，那么将根据类名创建别名。如果别名超过字符限制，那么提供长度在字符限制范围内的别名。

要点: 别名的最长字符限制通过 alias_name + host_name 计算得到，不能超过 24 个字符。

c. 选择用于启用或禁用事务跟踪的选项。缺省值为 *Yes*。

d. 选择用于启用或禁用诊断数据收集的选项。缺省值为 *Yes*。

1) 如果选择 *Yes*，请选择“方法跟踪”方式选项。缺省值为 *No*。

e. 如果从步骤 b 中提供的列表选择一个选项，请将启动脚本 <DCHOME>/j2se_dc/.gdc/*toolkit_version*/runtime/j2se<application_alias>.<hostname>.<application_alias> 复制到所选位置。

- 对于 V8.1.4.0.6 和更高版本，请完成以下步骤：

a. 出现提示时，输入 Java 应用程序的主目录。例如，/root/J2seApp/

b. 选择要监视的应用程序类型。

- Java 应用程序
- Jetty 服务器

c. 如果选择 *Java* 应用程序作为应用程序类型，请遵循步骤 5 的 V8.1.4.0.3 到 V8.1.4.0.5 部分中提及的步骤。

d. 如果选择 *Jetty* 服务器作为应用程序类型，请遵循以下步骤：

1) 输入 Jetty 主目录。例如，/home/jetty/jetty-distribution-9.4.12.v20180830

2) 输入别名。如果别名超过字符限制，那么提供长度在字符限制范围内的别名。

要点: 如果选择 *Java* 应用程序作为应用程序类型，并且从 V8.1.4.0.3 到 V8.1.4.0.5 部分的步骤 b 中的列表中选择任何选项，请将启动脚本 <DCHOME>/j2se_dc/.gdc/toolkit_version/runtime/j2se<application_alias>.<hostname>.<application_alias> 复制到所选位置。

e. 如果所选类型为 *Jetty* 服务器，那么会将 dcstartup.bat/dcstartup.sh 复制到给定的 Jetty 主目录中。

- 对于 V8.1.4.0.7，请完成以下步骤：

– 如果您使用 Open JDK V9 或更高版本来配置 J2SE 数据收集器，那么在输入 Java 主目录的路径时，将会显示警告，其内容如下：

警告:

```
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by jnr.posix.JavaLibCHelper
(file:/root/testopen/preconf-13march/j2se_dc/.gdc/7.3.0.14.0/
bin/lib/jython.jar)to method sun.nio.ch.SelChImpl.getFD()
WARNING: Please consider reporting this to the maintainers of
jnr.posix.JavaLibCHelper
WARNING: Use --illegal-access=warn to enable warnings of next
illegal reflective access operations
WARNING: All illegal access operations will be denied in next
release Mar 15, 2019 11:35:06 AM
org.python.netty.util.internal.PlatformDependent <clinit>
INFO: Your platform does not provide complete lowlevel API for
accessing direct buffers reliably.
Unless explicitly requested, heap buffer is always preferred
to avoid potential system unstability.
```

但是，J2SE 数据收集器正常工作，您可以忽略警告。

对于 V8.1.4.0.2 和更低版本，请遵循步骤 6 来修改 Windows dcstartup.bat 或 Linux AIX dcstartup.sh 文件。

6. 要修改 Windows dcstartup.bat 或 Linux AIX dcstartup.sh 文件，请完成下列步骤：
   - 如果应用程序类和 JAR 文件捆绑到一个 JAR 文件中，请完成下列步骤：
     a. 打开以下文件：
        - Windows dcstartup.bat
        - Linux AIX dcstartup.sh
     b. 将 `-cp .:$classpath:$Classpath $ITCAM_JVM_OPTS` *full name of the main class* 替换为 `$ITCAM_JVM_OPTS -jar` *Application jar file* 并保存文件。
   - 如果应用程序使用多个 JAR 文件，请完成下列步骤：
     a. 打开以下文件：
        - Windows dcstartup.bat
        - Linux AIX dcstartup.sh
     b. 将 *CLASSPATH* 变量设置为 JAR 文件。
     c. 将 `-cp .:$classpath:$Classpath $ITCAM_JVM_OPTS` *full name of the main class* 替换为 `-cp .:$classpath:$Classpath $ITCAM_JVM_OPTS -jar` *Application jar file* 并保存文件。

        应用程序 JAR 文件必须包含主应用程序类。

   注: 要修改 V8.1.4.0.3 到 V8.1.4.0.5 和 V8.1.4.0.6 到更高版本的 Windows dcstartup.bat 或 Linux AIX dcstartup.sh 文件（如果选择 *Java* 应用程序作为应用程序类型），请遵循步骤 7。
7. 当应用程序使用多个 JAR 文件时，请完成以下步骤。

   a) 打开以下文件：
   - Windows dcstartup.bat
   - Linux AIX dcstartup.sh

   b) 将 *CLASSPATH* 变量设置为 JAR 文件。

   c) 对于 V8.1.4.0.7，如果使用 Java 9 或 10 来配置 J2SE 数据收集器，并且使用 SSL 连接进行 APM 连接，那么不会显示事务跟踪数据。要解决此问题，可以将标志 **--add-modules java.xml.bind** 添加到 dcstartup.bat 或 dcstartup.sh 文件的最后一行。

   例如，

· 如果应用程序是 jar 文件，那么按如下所示更新最后一行：

```
PathToJava --add-modules java.xml.bind --add-opens=
jdk.management/com.sun.management.internal=
ALL-UNNAMED -jar $Classpath $ITCAM_JVM_OPTS AppJarName.jar
```

· 如果应用程序未捆绑在 jar 文件中，那么按如下所示更新最后一行：

```
PathToJava --add-modules java.xml.bind --add-opens=
jdk.management/com.sun.management.internal=ALL-UNNAMED -cp
.:$classpath:$Classpath $ITCAM_JVM_OPTS FullyQualifiedClassName
```

8. 要启用深入诊断监视，请使用要监视的特定于 J2SE 的类和方法来编辑 custom_request.xml。您可以通过两种方式执行此操作：手动和自动过程。

   要使用特定于 J2SE 应用程序的类和方法自动填充 custom_request.xml，请执行以下操作：

   a) 转至 <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/ 目录并打开 dc.properties 文件。

   b) 通过将 is.auto.update.custom_requests.xml 属性的值设置为 *true* 并保存文件来启用该属性。

   c) 从步骤 9 运行命令。

   d) 在 10 到 15 分钟后停止数据收集器。

   e) 检查是否使用定制方法和类填充了 DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/custom/custom_requests.xml。

   f) 移除不需要的条目，然后再次打开 dc.properties 文件。

   g) 通过将 is.auto.update.custom_requests.xml 属性的值设置为 *false* 并保存文件来禁用该属性。

   h) 从步骤 9 运行命令。

   注: 如果未自动发现应用程序中的某些定制方法，那么您需要手动添加定制方法。

   要手动填充 custom_request.xml，请执行以下操作：

   a) 转至 <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/runtime/j2se<application_alias>.<hostname>.<application_alias>/custom/custom_requests.xml 并编辑 custom_request.xml。

   例如，

```
<edgeRequest>
<requestName>truncateDb</requestName>
<Matches>testApp.JDBC.DBManager</Matches>
<type>application</type>
<methodName>truncateDb</methodName>
</edgeRequest>
```

   b) 添加特定于应用程序的类和方法。

9. 运行以下命令：

   Windows dcstartup.bat

   Linux AIX dcstartup.sh

   注: 如果所选应用程序类型为 *Jetty* 服务器，请运行 Jetty 主目录中存在的 dcstartup.bat/dcstartup.sh。

J2SE 应用程序将随着配置的数据收集器一起启动。

下一步做什么

登录 Cloud APM 控制台可查看仪表板中数据收集器收集的数据。有关使用控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

检查事务跟踪和诊断数据收集的状态

对于 V8.1.0.3 和更高版本，在代理程序配置页面上，您可以检查事务跟踪和诊断数据的状态。

关于此任务

您可以借助于两个命令来检查事务跟踪和诊断数据收集的状态。请参阅此过程以了解这些命令。

过程

1. 使用 **config status** 命令。
   a) 打开 bin 目录。发出命令

   Linux AIX **cd <DCHOME>/j2se_dc/.gdc/toolkit_version/bin/**

   Windows **cd <DCHOME>\j2se_dc\.gdc\toolkit_version\bin**

   其中，*toolkit_version* 表示：

   · 对于 V8.1.4.0 和更低版本，*toolkit_verion* 为 7.3.0.5.0。

   · 对于 V8.1.4.0.1 和更高版本，*toolkit_verion* 为 7.3.0.14.0。

   b) 输入以下命令以检查状态

   Linux AIX **config.sh status**

   Windows **config.bat status**

   c) 从列表中选择具有已标识的别名的 Java 应用程序以检查其状态，或选择退出。

   1) ddperf
   2) 主要
   3) 退出
2. 使用 **config status <application_alias_name>** 命令。
   a) 输入以下命令以打开目录

   Linux AIX **cd <DCHOME>/j2se_dc/.gdc/7.3.0.14.0/bin/**

   Windows **cd <DCHOME>\j2se_dc\.gdc\7.3.0.14.0\bin**

   b) 输入以下命令以检查状态

   Linux AIX **config.sh status <application_alias_name>**

   Windows **config.bat status <application_alias_name>**

更改事务跟踪和诊断数据收集的状态

对于 V8.1.0.3 和更高版本，在代理程序配置页面上，您可以更改事务跟踪和诊断数据的状态。

关于此任务

您可以借助于命令提示符来更改事务跟踪和诊断数据收集的状态。请参阅以下过程以了解这些命令。

过程

1. 打开 bin 目录并运行以下命令：

Linux AIX **cd <DCHOME>/j2se_dc/.gdc/toolkit_version/bin/**

Windows **cd <DCHOME>\j2se_dc\.gdc\toolkit_version\bin**

其中，*toolkit_version* 表示：

- 对于 V8.1.4.0 和更低版本，*toolkit_verion* 为 7.3.0.5.0。
- 对于 V8.1.4.0.1 和更高版本，*toolkit_verion* 为 7.3.0.14.0。

2. 要检查状态，请输入以下命令：

   Linux AIX **config.sh <application_alias_name>**

   Windows **config.bat <application_alias_name>**

3. 当出现提示时，选择此选项以启用或禁用事务跟踪。缺省值为 Yes。
4. 当出现提示时，选择此选项以启用或禁用诊断数据收集。缺省值为 Yes。

   a) 如果值为 Yes，那么选择此选项以启用/禁用方法跟踪。缺省值为 NO。

配置 JBoss 监视

Monitoring Agent for JBoss 监视 JBoss 应用程序服务器和 JBoss Enterprise Application 平台的资源。使用随 JBoss 代理程序一起提供的仪表板可识别最慢的应用程序、最慢的请求、线程池瓶颈、JVM 堆内存和垃圾回收问题、最忙碌的会话以及 JBoss 应用程序服务器的其他瓶颈。

开始之前

- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 JBoss 代理程序的系统需求。有关最新系统需求信息，请参阅 JBoss 代理程序的软件产品兼容性报告 (SPCR)。
- 在配置 JBoss 代理程序之前，必须先通过完成以下步骤来配置 JBoss 服务器。

  1. 第 395 页的『启用 JMX MBean 服务器连接』.
  2. 第 397 页的『添加 JBoss 服务器管理用户』.
  3. 第 397 页的『启用 Web/HTTP 统计信息收集』. 该过程适用于 JBoss EAP V7.x 以及 WildFly V8.x、V9.x 和 V10.x。

关于此任务

“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。“受管系统名称”限制为 32 个字符。

指定的实例名称限制为 28 个字符减去主机名长度。例如，如果指定 JBoss 作为实例名称，那么受管系统名称为 JBoss:hostname:JE。

注: 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法正确显示。

JBoss 代理程序 是多实例代理程序。必须为监视的每个 JBoss 服务器创建一个代理程序实例，并手动启动每个代理程序实例。

事务跟踪功能可用于 Cloud APM, Advanced 产品中的 JBoss 代理程序。

- 要为新代理程序实例启用事务跟踪，请完成该过程的步骤 1 或步骤 2，然后执行该过程以第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』。
- 要启用已配置为基本监视的代理程序实例的事务跟踪，请遵循该过程以第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』。
- 要禁用代理程序实例的事务跟踪，请遵循该过程以第 407 页的『禁用 JBoss 代理程序事务跟踪数据收集器』。
- 要卸载所有代理程序实例的事务跟踪，并移除事务跟踪工具箱，请遵循该过程以第 408 页的『卸载所有 JBoss 代理程序事务跟踪』。

过程

1. 使用 **IBM Performance Management** 窗口或静默响应文件在 Windows 系统上配置代理程序。
   - 第 399 页的『在 Windows 系统上配置代理程序』.
   - 第 402 页的『使用静默响应文件来配置代理程序』.
2. 通过运行命令行脚本并响应提示，或者使用静默响应文件在 Linux 系统上配置代理程序。
   - 第 401 页的『通过对提示做出响应来配置代理程序』.
   - 第 402 页的『使用静默响应文件来配置代理程序』.
3. 可选： 通过配置各个代理程序实例提供事务跟踪数据，并配置应用程序性能仪表板以显示事务跟踪数据来配置事务跟踪。
   a) 遵循此过程以第 404 页的『设置 JBoss 代理程序事务跟踪数据收集器』。
   b) 在应用程序性能仪表板中为 JBoss 代理程序启用事务跟踪数据。
      1) 从导航栏中，单击 **系统配置** > **代理程序配置**。“**代理程序配置**”页面随即显示。
      2) 选择 **JBoss** 选项卡。
      3) 选中要监视的 JBoss 服务器代理程序实例的相应复选框，然后从**操作**列表执行下列其中一项操作：
         - 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。**事务跟踪**列中的状态将更新为已启用。
         - 要禁用事务跟踪，请单击**设置事务跟踪** > **已禁用**。**事务跟踪**列中的状态将更新为已禁用。
   c) 通过将 JBoss 代理程序实例添加到应用程序性能仪表板中的应用程序来查看 JBoss 代理程序事务跟踪数据仪表板。
      有关使用“应用程序”编辑器的更多信息，请参阅管理应用程序。
   d) 确保将用户帐户分配给包含“诊断仪表板”许可权的角色，以有权访问下列 JBoss 代理程序 事务跟踪 应用程序仪表板 按钮。
      否则，在“应用程序仪表板”中，对于该用户，禁用这些按钮。
      1) **最慢的 5 个响应时间**窗口小部件上的**诊断**向下钻取按钮。
      2) **应用程序**窗口小部件上的**正在进行的请求**按钮。

注: 事务跟踪功能可用于 Cloud APM, Advanced 产品中的 JBoss 代理程序。对于具备 Cloud APM, Base 产品中提供的基本资源监视功能的 JBoss 代理程序，请跳过此步骤。

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。这些日志的缺省路径如下所示。

- Linux `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

启用 JMX MBean 服务器连接

必须先启用 Java 管理扩展 (JMX) MBean 服务器连接，JBoss 代理程序才能从 JBoss 服务器收集数据。

过程

请根据您的 JBoss 服务器发行版和版本，完成相应的步骤。

- 配置 EAP 5.2。

备份 `run.conf` 文件，然后将下列行添加到该文件中：

```
JAVA_OPTS="$JAVA_OPTS -Djboss.platform.mbeanserver"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.ssl=false"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.authenticate=false"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.port=1090"
JAVA_OPTS="$JAVA_OPTS -Djavax.management.builder.initial=
                      org.jboss.system.server.jmx.MBeanServerBuilderImpl"
```

- 配置 AS 6.x。

  启动 JBoss 服务器时，请指定绑定地址作为参数。

  - Linux *jboss_server_home*/bin/run.sh **-b** Ip_address
  - Windows *jboss_server_home*\bin\run.bat **-b** <Ip_address>

  其中，*jboss_server_home* 是 JBoss 服务器的安装目录。

  例如，如果绑定地址为 10.77.9.250：

  `/apps/wildfly-9.0.2.Final/bin/run.sh` **-b** `10.77.9.250`

- 配置所有其他受支持的版本。

  缺省情况下，安装 JBoss 和 WildFly 服务器，并且其 JMX 端口已禁用远程管理。必须更改 JBoss 服务器的配置，以允许进行远程管理。必须编辑 *jboss_server_home*`/standalone/configuration/standalone.xml`，以允许进行远程管理。

  a) 备份 *jboss_server_home*`/standalone/configuration/standalone.xml` 文件。

  其中，*jboss_server_home* 是 JBoss 服务器的安装目录。

  b) 允许远程配置。

  搜索 `urn:jboss:domain:jmx` 并在其子系统部分中，确保 `remoting-connector` 条目具有 `use-management-endpoint="true"`。

  结果示例。

```
<subsystem xmlns="urn:jboss:domain:jmx:1.3">
            <expose-resolved-model/>
            <expose-expression-model/>
            <remoting-connector use-management-endpoint="true"/>
</subsystem>
```

  c) 允许远程连接。

  找到定义接口的位置，将 127.0.0.1（回送地址）替换为要与服务器绑定的外部 IP。请勿与 0.0.0.0 绑定。

  替换前的示例。

```
  <interfaces>
        <interface name="management">
            <inet-address value="${jboss.bind.address.management:127.0.0.1}"/>
        </interface>
        <interface name="public">
            <inet-address value="${jboss.bind.address:127.0.0.1}"/>
        </interface>
  ...
```

  替换后的示例（外部 IP 地址为 192.168.101.1）。

```
  <interfaces>
        <interface name="management">
            <inet-address value="${jboss.bind.address.management:192.168.101.1}"/>
        </interface>
        <interface name="public">
            <inet-address value="${jboss.bind.address:192.168.101.1}"/>
        </interface>
  ...
```

添加 JBoss 服务器管理用户

必须先添加管理用户（如果不存在），然后 JBoss 代理程序才能从 JBoss 服务器收集数据。

过程

使用 JBoss **add-user** 脚本来添加管理用户。

1. 转到 JBoss 服务器安装目录下的二进制文件目录或 bin 目录。
2. 运行 **add-user** 脚本。
   - Linux **./add-user.sh**
   - Windows **add-user.bat**
3. 遵循提示，以生成管理用户。

示例

```
root@jboss-wf10-rh7:/apps/wildfly-10.0.0.Final/bin
] ./add-user.sh

What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
(a): a

Enter the details of the new user to add.
Using realm 'ManagementRealm' as discovered from the existing property files.
Username : MyAdmin
Password recommendations are listed below. To modify these restrictions edit the add-
user.properties
configuration file.
 - The password should be different from the username
 - The password should not be one of the following restricted values {root, admin,
administrator}
 - The password should contain at least 8 characters, 1 alphabetic character(s), 1 digit(s),
1 non-alphanumeric symbol(s)
Password :
Re-enter Password :
What groups do you want this user to belong to? (Please enter a comma separated list, or leave
blank
for none)[  ]:
About to add user 'MyAdmin' for realm 'ManagementRealm'
Is this correct yes/no? yes
Added user 'MyAdmin' to file '/apps/wildfly-10.0.0.Final/standalone/configuration/mgmt-
users.properties'
Added user 'MyAdmin' to file '/apps/wildfly-10.0.0.Final/domain/configuration/mgmt-
users.properties'
Added user 'MyAdmin' with groups  to file
           '/apps/wildfly-10.0.0.Final/standalone/configuration/mgmt-groups.properties'
Added user 'MyAdmin' with groups  to file
           '/apps/wildfly-10.0.0.Final/domain/configuration/mgmt-groups.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for
server to
server EJB calls.
yes/no? no
```

启用 Web/HTTP 统计信息收集

必须先对每个子系统启用统计信息收集，JBoss 代理程序才能收集 JBoss 服务器 Web 度量和其他子系统度量。该过程适用于 JBoss EAP V7.x 以及 WildFly V8.x、V9.x 和 V10.x。

过程

各个 JBoss 子系统的 **statistics-enabled** 属性控制统计信息收集。可使用 JBoss 命令行界面来查看和更新此设置。

注: 该过程适用于 JBoss EAP V7.x 以及 WildFly V8.x、V9.x 和 V10.x。

1. 转至 JBoss 服务器安装目录下的二进制文件目录或 bin 目录。
2. 启动 JBoss 命令行界面。

- Linux **./jboss-cli.sh --connect** [**--controller**=*IP*:*port*]
- Windows **jboss-cli.bat --connect** [**--controller**=*IP*:*port*]

其中，*IP* 是 JBoss 服务器的 IP 地址，*port* 是 JBoss 服务器的端口。例如，192.168.10.20:9990。

提示: 如果连接尝试引起错误：“Failed to connect to the controller: The controller is not available at localhost:9990: java.net.ConnectException: WFLYPRT0053: Could not connect to http-remoting://localhost:9990. The connection failed: WFLYPRT0053: Could not connect to http-remoting://localhost:9990. The connection failed: Connection refused”，请使用 **--controller** 参数。

此错误表明管理服务器没有在本地主机 IP 地址 (127.0.0.1) 上进行侦听，而是配置为在计算机的 IP 地址上进行侦听。

3. 运行以下命令来查看每个子系统的 statistics-enabled 属性的当前状态：

   注: 如果 JBoss 是在域方式下运行，那么每个命令都必须以关联的概要文件为前缀，并且必须针对每个受监视概要文件运行这些命令。例如：/profile=full/subsystem=ejb3:read-attribute(name=statistics-enabled)

   /subsystem=**ejb3**:read-attribute(name=enable-statistics)

   /subsystem=**transactions**:read-attribute(name=statistics-enabled)

   /subsystem=**undertow**:read-attribute(name=statistics-enabled)

   /subsystem=**webservices**:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=pool**:read-attribute(name=statistics-enabled)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=jdbc**:read-attribute(name=statistics-enabled)

   其中，*Data_Source_Name* 是配置用于 JBoss 的数据源的名称。

   注: 使用命令 /subsystem=datasources:read-resource 可以列出数据源。

   未启用统计信息时的结果示例：

   ```
   {
       "outcome" => "success",
       "result" => false
   }
   ```

4. 运行以下命令将每个子系统的 statistics-enabled 属性值更改为 *true*：

   /subsystem=**ejb3**:write-attribute(name=enable-statistics, value=true)

   /subsystem=**transactions**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**undertow**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**webservices**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=pool**:write-attribute(name=statistics-enabled,value=true)

   /subsystem=**datasources/data-source**=*Data_Source_Name*/**statistics=jdbc**:write-attribute(name=statistics-enabled,value=true)

   为子系统启用统计信息时的结果示例：

```
{
    "outcome" => "success",
    "response-headers" => {
        "operation-requires-reload" => true,
        "process-state" => "reload-required"
    }
}
```

5. 退出 JBoss 命令行界面。
6. 重新启动 JBoss 服务器。

   注: 必须重新启动已启用事务跟踪的任何当前正在运行的 JBoss 代理程序。

在 Windows 系统上配置代理程序

在 Windows 操作系统上，您可使用 IBM Cloud Application Performance Management 窗口来配置 JBoss 代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for JBoss** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，将禁用**配置代理程序**选项。要再次配置代理程序实例，请右键单击该实例，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用字母、阿拉伯数字、下划线字符和减号字符。
   例如，`jboss01`。

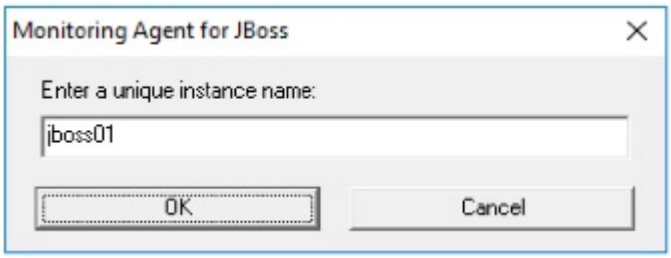


图 *17.* 用于输入唯一实例名称的窗口。

4. 输入 JBoss 服务器设置，然后单击**下一步**。
   请参阅第 403 页的表 167，以获取每个配置参数的说明。

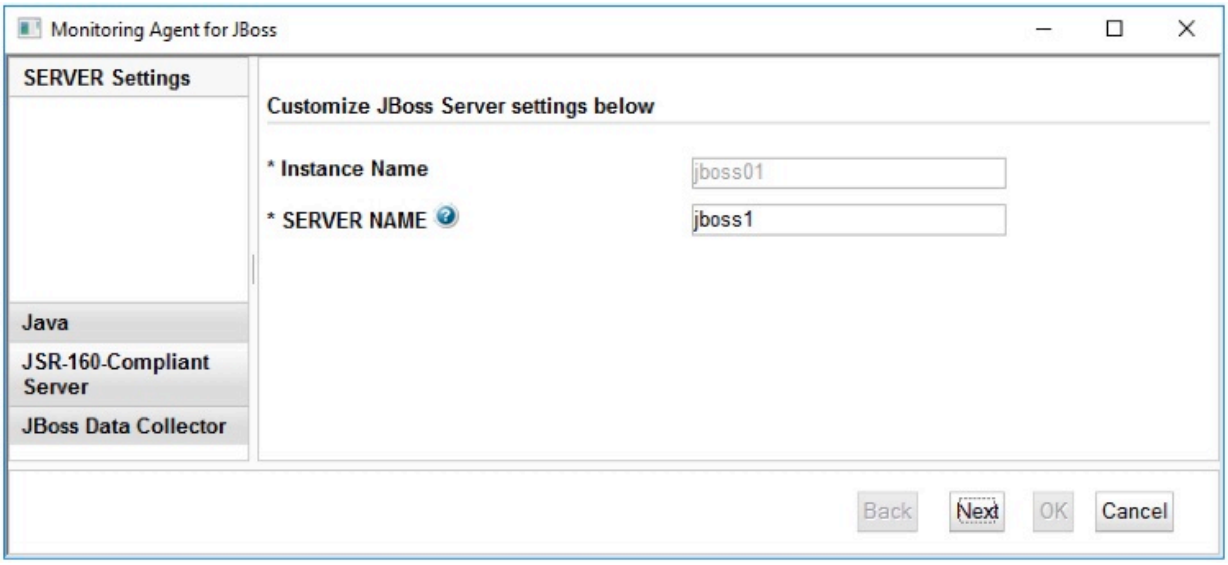


图 *18. JBoss* 服务器的配置参数窗口

5. 输入 Java 设置，然后单击**下一步**。
   请参阅第 403 页的表 167，以获取每个配置参数的说明。

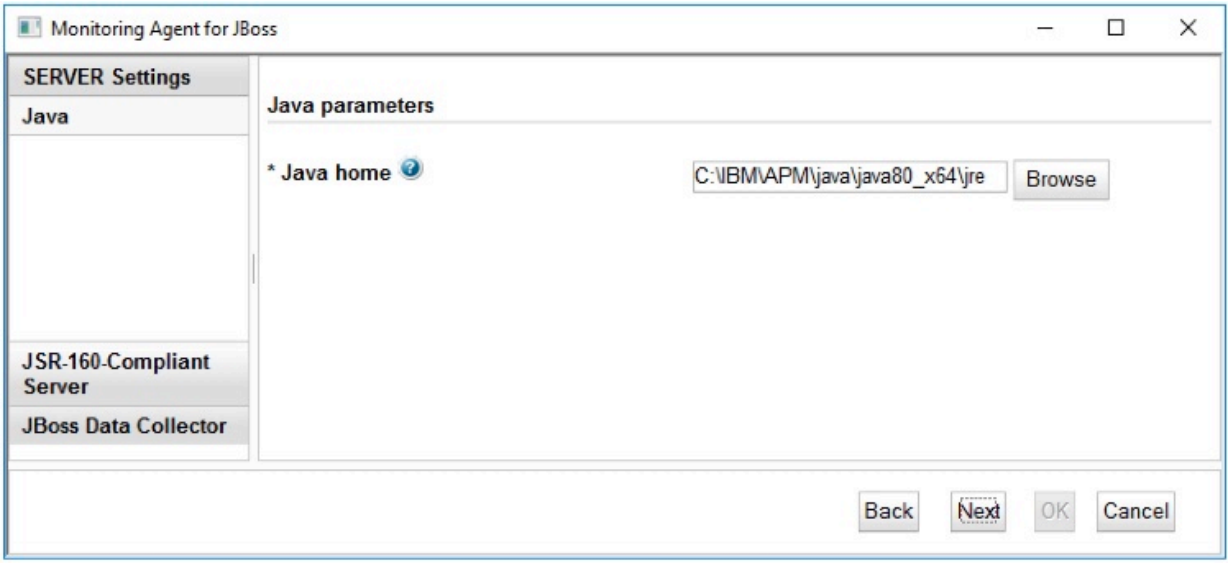


图 *19.* 用于指定 *Java* 设置的窗口。

6. 输入 JMX 设置，然后单击**下一步**。
   请参阅第 403 页的表 167，以获取每个配置参数的说明。

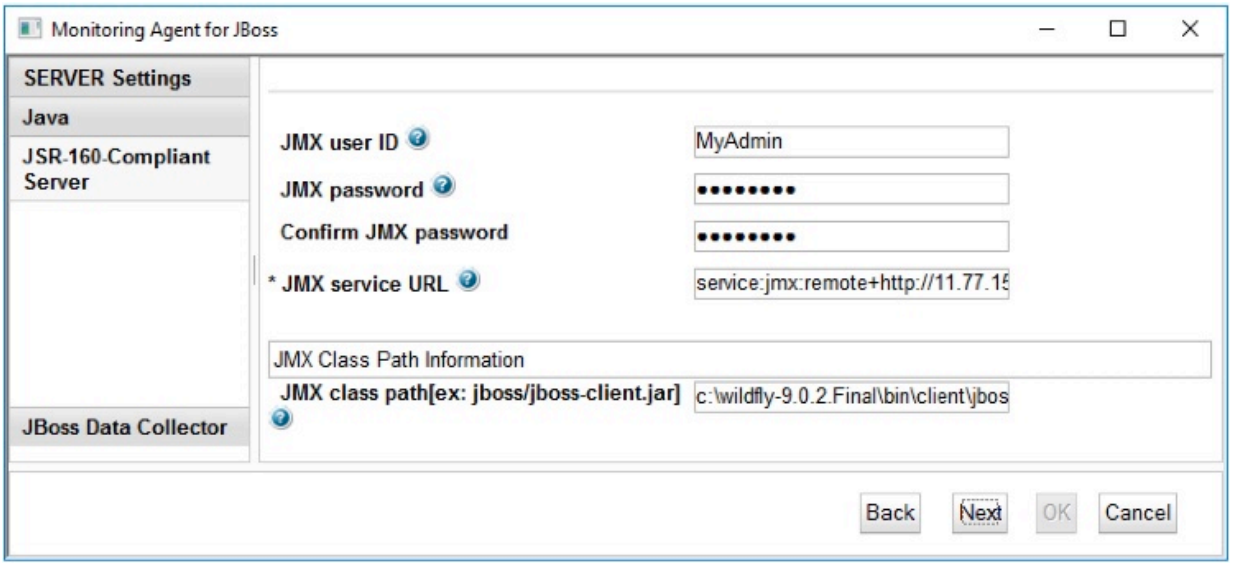


图 *20.* 用于指定 *JMX* 设置的窗口。

7. 查看 JBoss 代理程序数据收集器设置。

   在代理程序初始安装期间将 **DC Runtime Directory** 留空。请参阅第 403 页的表 167，以获取每个配置参数的说明。

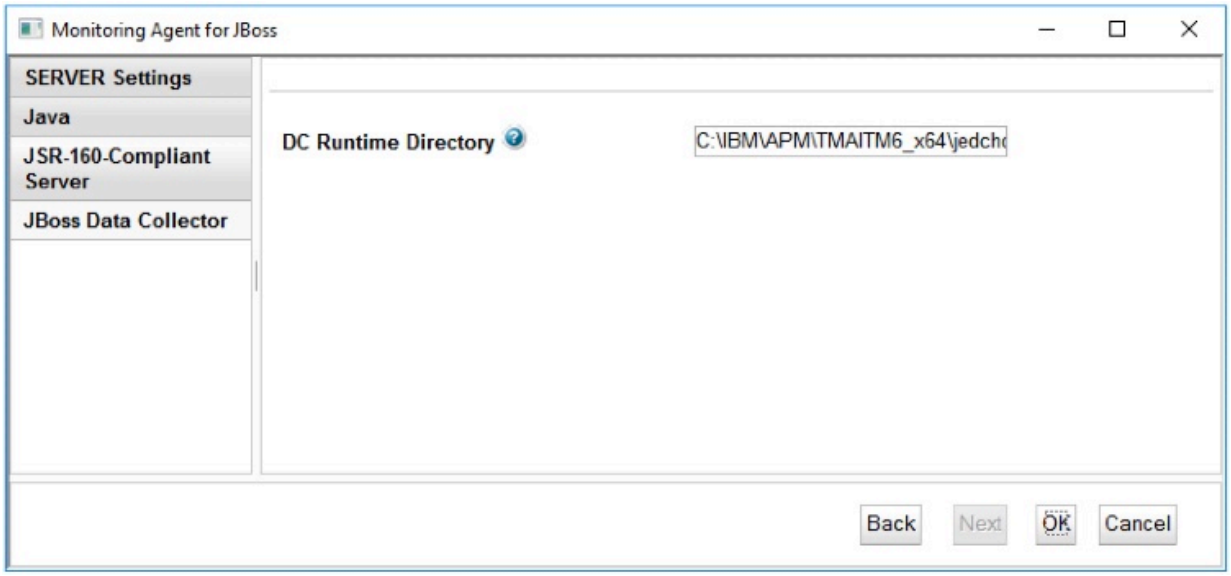


图 *21.* 用于指定 *JBoss* 代理程序数据收集器设置的窗口

8. 单击**确定**以完成代理程序配置。
9. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过对提示做出响应来配置代理程序

安装 JBoss 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 JBoss 代理程序是安装在本地 Linux 或 UNIX 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

1. 在命令行上，运行以下命令：

   ```
   install_dir/bin/jboss-agent.sh config
   instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/jboss-agent.sh config example-inst01
   ```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 403 页的『JBoss 代理程序的配置参数』，以获取每个配置参数的说明。

3. 运行以下命令以启动代理程序：

   ```
   install_dir/bin/jboss-agent.sh start
   instance_name
   ```

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/jboss-agent.sh start example-inst01
   ```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

- 要以静默方式配置 JBoss 代理程序，请完成下列步骤：

  a) 在文本编辑器中，打开位于以下路径的 `jboss_silent_config.txt` 文件：

  – Linux AIX *install_dir*/samples/jboss_silent_config.txt

    示例： /opt/ibm/apm/agent/samples/jboss_silent_config.txt

  – Windows *install_dir*\samples\jboss_silent_config.txt

  其中，*install_dir* 是安装代理程序的路径。

  以下列出了缺省 *install_dir* 路径：

  – Linux /opt/ibm/apm/agent

  – Windows C:\IBM\APM\TMAITM6_x64

  示例

  Linux AIX /opt/ibm/apm/agent/samples/jboss_silent_config.txt

  Windows C:\IBM\APM\samples\jboss_silent_config.txt

  b) 在 `jboss_silent_config.txt` 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

请参阅第 403 页的『JBoss 代理程序的配置参数』，以获取每个配置参数的说明。

c) 保存并关闭 `jboss_silent_config.txt` 文件，然后运行下列命令：

- Linux AIX *install_dir*/bin/jboss-agent.sh config **instance_name** *install_dir*/samples/jboss_silent_config.txt
- Windows *install_dir*\bin\jboss-agent.bat config **instance_name** *install_dir*\samples\jboss_silent_config.txt

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

以下列出了缺省 *install_dir* 路径：

- Linux `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

示例

```
Linux AIX /opt/ibm/apm/agent/bin/jboss-agent.sh config example-inst01
/opt/ibm/apm/agent/samples/jboss_silent_config.txt
```

```
Windows C:\IBM\APM\bin\jboss-agent.bat config example-inst01 C:\IBM\APM\samples
\jboss_silent_config.txt
```

d) 运行以下命令以启动代理程序：

- Linux AIX *install_dir*/bin/jboss-agent.sh start **instance_name**
- Windows *install_dir*\bin\jboss-agent.bat start **instance_name**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

以下列出了缺省 *install_dir* 路径：

- Linux `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

示例

```
Linux AIX /opt/ibm/apm/agent/bin/jboss-agent.sh start example-inst01
```

```
Windows C:\IBM\APM\bin\jboss-agent.bat start example-inst01
```

JBoss 代理程序的配置参数

下表显示 JBoss 代理程序的配置参数。

1. JBoss 代理程序设置 - JBoss 代理程序环境设置。
2. 第 404 页的表 168 - 示例 JMX 服务 URL。

表 *167. JBoss* 代理程序设置

参数名称	描述	静默配置文件参数名称
服务器名称	请提供一个名称，用于识别 JBoss/WildFly 服务器。	**KJE_SERVER**
Java 主目录	Java 的安装路径。	**JAVA_HOME**
JMX 用户标识	用于连接到 MBean 服务器的用户标识。	**KQZ_JMX_JSR160_JSR160_USER_ID**

表 167. JBoss 代理程序设置 (续)		
参数名称	**描述**	**静默配置文件参数名称**
JMX 密码	密码	**KQZ_JMX_JSR160_JSR160_PASSWORD**
JMX 服务 URL	用于连接到 MBean 服务器的服务 URL。 请参阅第 404 页的表 168 以获取示例。	**KQZ_JMX_JSR160_JSR160_SERVICE_URL**
JMX 类路径	查找类或资源时所要搜索的 JAR 文件。请查找 JBoss 服务器的 `jboss-client.jar` 文件，并输入其路径。JBoss EAP 6 服务器的示例：`/opt/EAP-6.3.0/jboss-eap-6.3/bin/client/jboss-client.jar`。	**KQZ_JMX_JSR160_JSR160_JAR_FILES**
DC 运行时目录	**注:** 此参数仅适用于具备事务跟踪功能的 JBoss 代理程序，该功能在 Cloud APM, Advanced 产品中提供。对于具备 Cloud APM, Base 产品中提供的基本资源监视功能的 JBoss 代理程序，请跳过此参数。 **simpleConfig** 脚本所设置的 JBoss 数据收集器运行时目录完整路径。在代理程序初始安装期间将该参数留空。	**KQZ_DC_RUNTIME_DIR**

表 168. JMX 服务 URL	
JBoss 服务器版本	**带有缺省端口的 JMX 服务 URL[1]**
WildFly 8、9 和 10 JBoss EAP 7	`service:jmx:remote+http://ip:9990` `service:jmx:remote+https://ip:9994`
JBoss EAP 6 JBoss AS 7	`service:jmx:remoting-jmx://ip:9999`
JBoss EAP 5.2 JBoss AS 6.1	`service:jmx:rmi:///jndi/rmi://ip:1090/jmxrmi`

[1] 该端口基于 JBoss 配置文件条目 `<socket-binding name="management-native" interface="management" port="${jboss.management.native.port:NNNN}"/>` 中的端口。如果将端口从缺省值更改为其他值，请根据配置文件中的端口号对其进行调整。

设置 JBoss 代理程序事务跟踪数据收集器

JBoss 代理程序的事务跟踪功能要求更改代理程序实例环境设置文件、JBoss 服务器启动文件和 DC 运行时目录代理程序配置参数。还提供了有助于执行更改的脚本。

开始之前

Linux 确保打开的文件的资源限制大于 5,000，以使事务跟踪工具箱正常工作。

· 显示当前打开的文件的限制设置。**ulimit -n**

· 示例：将打开的文件限制设置为 5,056。**ulimit -n 5056**

在遵循此过程前，执行配置 JBoss 代理程序步骤 第 395 页的『1』 或 第 395 页的『2』。

必须在本地将 JBoss 代理程序安装到使用事务跟踪功能监视的 JBoss 服务器。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权：

1. *JBOSS_HOME* 目录。

2. *JBOSS_HOME*/bin 目录和文件。
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml 文件。
4. *install_dir*/config 目录。
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg 文件。

其中

JBOSS_HOME
JBoss 服务器安装目录。

install_dir
代理程序的安装路径。缺省路径为：

- Linux /opt/ibm/apm/agent
- Windows C:\IBM\APM\TMAITM6_x64

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 399 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 402 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 403 页的『3』

过程

运行 **simpleConfig** 脚本。

1. 登录安装了 JBoss 代理程序的 JBoss 服务器。
2. 切换到代理程序安装目录。
   - Linux *install_dir*
   - Windows *install_dir*\TMAITM6_x64

   其中，*install_dir* 是安装代理程序的路径。

   以下列出了缺省 *install_dir* 路径：
   - Linux /opt/ibm/apm/agent
   - Windows C:\IBM\APM\TMAITM6_x64
3. 将目录切换到 jedchome/7.3.0.13.0/bin。
4. 运行设置脚本。
   - Linux ./simpleConfig.sh
   - Windows simpleConfig.bat
5. 遵循提示，以便为环境输入参数：

   a) 输入为代理程序实例选择的 JBoss 代理程序 *instance_name*。

   b) 输入 JBoss 服务器安装目录。
   如果设置了 *JBOSS_HOME* 环境变量，那么将提供它的值作为缺省值。

   其中

   JBOSS_HOME
   JBoss 服务器安装目录。

   instance_name
   在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 399 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 402 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 403 页的『3』

install_dir
安装代理程序的路径。缺省路径为：

- Linux `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

6. 如果 JBoss 服务器和代理程序正在运行，请将其重新启动。

结果

在事务跟踪配置期间更改的 JBoss 服务器文件：

- *JBOSS_HOME*`/bin/standalone.conf`

  使用事务跟踪功能需要的配置设置更新了此文件。配置标记插入到文件中以在禁用事务跟踪功能时使用。在添加或移除事务跟踪功能更改前，在 *JBOSS_HOME*`/bak` 目录中保存备份文件。

- *JBOSS_HOME*`/modules/system/layers/base/org/jboss/as/server/main/module.xml`

  使用 JAVA EE API 模块依赖关系更新了此文件。配置标记插入到文件中以在禁用事务跟踪功能时使用。在添加或移除事务跟踪功能更改前，在 *JBOSS_HOME*`/bak` 目录中保存备份文件。

在事务跟踪配置期间更改的代理程序文件：

- 代理程序实例配置文件
  - Linux *install_dir*`/config/`*hostname*`_je_`*instance_name*`.cfg`
  - Windows *install_dir*`\TMAITM6_x64\`*hostname*`_JE_`*instance_name*`.cfg`
- 代理程序环境设置文件
  - Linux *install_dir*`/config/je_`*instance_name*`.environment`
  - Windows *install_dir*`\TMAITM6_x64\KJEENV_`*instance_name*

其中

JBOSS_HOME
JBoss 服务器安装目录。

install_dir
代理程序的安装路径。缺省路径为：

- Linux `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 399 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 402 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 403 页的『3』

禁用 JBoss 代理程序事务跟踪数据收集器

JBoss 代理程序的事务跟踪功能要求更改代理程序实例环境设置文件、JBoss 服务器启动文件和 DC 运行时目录代理程序配置参数。提供了一个脚本，用来为禁用了事务跟踪的代理程序实例移除这些更改。

开始之前

确保 JBoss 服务器和 JBoss 代理程序已关闭。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权：

1. *JBOSS_HOME* 目录
2. *JBOSS_HOME*/bin 目录和文件
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml 文件
4. *install_dir*/config 目录
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg 文件

过程

使用 **remove** 选项运行 **simpleConfig** 脚本。

1. 登录安装了 JBoss 代理程序的 JBoss 服务器。
2. 切换到代理程序安装目录。
   - Linux *install_dir*
   - Windows *install_dir*\TMAITM6_x64
3. 将目录切换到 jedchome/7.3.0.13.0/bin。
4. 使用 **remove** 选项运行 **simpleConfig**。
   - Linux ./simpleConfig.sh **remove** *instance_name*
   - Windows simpleConfig.bat **remove** *instance_name*
5. 启动 JBoss 服务器和代理程序。

其中：

JBOSS_HOME
JBoss 服务器安装目录

hostname
安装代理程序的主计算机的名称

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 399 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 402 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 403 页的『3』

install_dir
安装代理程序的路径。缺省路径为：

- Linux /opt/ibm/apm/agent
- Windows C:\IBM\APM\TMAITM6_x64

architecture
IBM Application Performance Management 或 Cloud APM 系统体系结构标识。例如，lx8266 代表 Linux Intel V2.6（64 位）。要获取体系结构代码的完整列表，请参阅 *install_dir*/registry/archdsc.tbl 文件。

卸载所有 JBoss 代理程序事务跟踪

可卸载 JBoss 代理程序的事务跟踪功能。提供了一个脚本，用来移除禁用了事务跟踪的所有代理程序实例并移除事务跟踪工具箱。

开始之前

确保 JBoss 服务器和所有 JBoss 代理程序实例已关闭。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权：

1. *JBOSS_HOME* 目录。
2. *JBOSS_HOME*/bin 目录和文件。
3. *JBOSS_HOME*/modules/system/layers/base/org/jboss/as/server/main/module.xml 文件。
4. *install_dir*/config 目录。
5. *install_dir*/config/*hostname*_je_*instance_name*.cfg 文件。

过程

使用 **uninstall** 选项运行 **simpleConfig** 脚本。

1. 登录安装了 JBoss 代理程序的 JBoss 服务器。
2. 切换到代理程序安装目录。
   - Linux *install_dir*/*architecture*/je/bin。例如：/opt/ibm/apm/agent/lx8266/je/bin 或 /opt/ibm/apm/agent/lx8266/je/bin
   - Windows *install_dir*\TMAITM6_x64
3. 将目录切换到 jedchome/7.3.0.13.0/bin。
4. 使用 **uninstall** 选项运行 **simpleConfig**。
   - Linux `./simpleConfig.sh uninstall`
   - Windows `simpleConfig.bat uninstall`
5. 启动 JBoss 服务器和所有代理程序实例。

其中：

JBOSS_HOME
JBoss 服务器安装目录。

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 399 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 402 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 403 页的『3』

install_dir
安装代理程序的路径。缺省路径为：

- Linux /opt/ibm/apm/agent
- Windows C:\IBM\APM\TMAITM6_x64

architecture
IBM Application Performance Management 或 Cloud APM 系统体系结构标识。例如，lx8266 代表 Linux Intel V2.6（64 位）。要获取体系结构代码的完整列表，请参阅 *install_dir*/registry/archdsc.tbl 文件。

配置 Linux KVM 监视

您必须配置 Monitoring Agent for Linux KVM 才能收集 Red Hat Enterprise Virtualization Hypervisor (RHEVH) 和 Red Hat Enterprise Virtualization Manager (RHEVM) 服务器的数据。在服务器或虚拟机上安装此代理程序之后，您必须创建第一个实例，并手动启动此代理程序。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Linux KVM 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

Linux KVM 代理程序是多实例、多连接代理程序。多实例表示您可以创建多个实例，并且每个实例都可以与一个或多个 RHEVM 或 RHEVH 服务器建立多个连接。

切记: 请使用不同的实例来监视 RHEVM 或 RHEVH 服务器。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

您可以使用同一个配置脚本为 RHEVH 和 RHEVM 服务器配置实例。

- 要配置与 RHEVM 服务器的连接，请完成“配置 RHEVM 服务器连接”主题所述的步骤。
- 要配置与 RHEVH 服务器的连接，请完成“配置 RHEVH 服务器连接”主题所述的步骤。

创建用户和授予所需许可权

在您配置 Linux KVM 代理程序之前，必须创建一个用户并向该用户授予监视 RHEVM 和 RHEVH 服务器所需的许可权。

过程

1. 打开 **Red Hat Enterprise Virtualization Manager Web 管理**门户网站。
2. 单击**配置**。
3. 在“**配置**”窗口中，选择**角色**。
   a) 要创建角色，请单击**新建**。
   b) 在“**新建角色**”窗口中，添加角色名称，并选择**管理**作为帐户类型。
   c) 确保未选中“**允许操作的复选框**”窗格中的复选框，并单击**确定**。
4. 在“**配置**”窗口中，选择**系统许可权**。
   a) 要授予用户许可权，请单击**添加**。
   b) 在“**向用户添加系统许可权**”窗口中，选择要将许可权授予的用户。
   c) 从**向用户分配角色**列表中，选择您创建的角色，并单击**确定**。

下一步做什么

完成代理程序配置:

- 第 415 页的『配置 RHEVH 服务器连接』
- 第 413 页的『配置 RHEVM 服务器连接』

配置协议

代理程序使用不同的协议来连接到 RHEVH 服务器。您可配置下列任何协议：SSH、TLS 或 TCP。

关于此任务

Linux KVM 代理程序使用 **virsh** 工具以远程方式连接到每个管理 QEMU-KVM 虚拟机的系统管理程序，并收集度量。代理程序环境中的 `libvirt` API 使用多个不同的远程传输协议。要获取受支持协议的列表，请参阅远程支持页面。

配置 SSH 协议

您可配置 SSH 协议，以便以远程方式监视主机。

关于此任务

假设： Linux KVM 代理程序安装在主机 A 上。您想要远程监视主机 B 上的系统管理程序。

过程

1. 以运行 Linux KVM 代理程序进程的用户标识（例如，root 用户标识）登录主机 A。

   提示: 确保您知道主机 B 上接受 SSH 连接的标识以及主机 A 上的 root 用户标识。
2. 通过使用 *ssh-keygen* 实用程序，在主机 A 上生成 **id_rsa** 和 **id_rsa.pub** 密钥。

   这些密钥将保存在以下位置：`~/.ssh: $ ssh-keygen -t rsa`。
3. 从主机 B 复制授权密钥：

   $ scp Id on host B@name or IP address of host B:~/.ssh/authorized_keys ~/.ssh/authorized_keys_from_B
4. 将主机 A 的公用密钥追加到主机 B 的授权密钥末尾：

   cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys_from_B
5. 将授权密钥复制回到主机 B：

   $ scp ~/.ssh/authorized_keys_from_B Id on host B@name or IP address of host B:~/.ssh/authorizede_keys

   切记: 如果要监视多台主机，请对每台主机重复步骤 第 410 页的『3』、第 410 页的『4』 和 第 410 页的『5』。
6. 移除已复制到主机 B 的授权密钥：

   ~/.ssh/authorized_keys_from_B
7. 将以下命令添加到主机 A 上当前标识的 **~/.bash_** 概要文件中：

   $ eval `ssh-agent`

   切记: 请确保使用美式键盘上波浪符 (~) 下方的单反引号 (`)，而不是单引号 (')。
8. 将该标识添加到主机 A，并输入您创建该标识时使用的密码：

   $ ssh-add ~/.ssh/id_rsa
9. 如果您收到 `Could not open a connection to your authentication agent` 消息，请运行以下命令：

   exec ssh-agent bash

   提示: 可以将 bash 替换为您所使用的 Shell，然后再次运行以下命令：

   $ ssh-add ~/.ssh/id_rsa
10. 测试 SSH 协议，以确保无需输入 SSH 密码即可从主机 A 连接到主机 B：

    提示: 如果要监视多台主机，请使用以下命令来测试每台主机的连接：

    $ ssh Id on host B@name or IP address of host B
11. 要验证连接，请运行以下命令：

virsh -c qemu+ssh://Id on host B@name or IP address of host B:port/system

如果您未更改缺省 SSH 端口，请省略此命令的 **:port** 部分。

要点: 如果 **virsh** 命令成功，Linux KVM 代理程序 会连接到系统管理程序。

12. 在主机 A 上重新启动 Linux KVM 代理程序之前必须重新启动主机 A。重新启动时，每次都必须重新运行 **ssh-add** 命令并指定密码。

    提示: 您可以使用 SSH 密钥链以避免重新输入密码。

配置 TLS 协议

您可配置 TLS 协议，以便以远程方式监视主机。

关于此任务

假设： Linux KVM 代理程序安装在主机 A 上。您想要远程监视主机 B 上的系统管理程序。

过程

1. 要在系统管理程序中创建认证中心 (CA) 密钥和证书，请完成下列步骤：
   a) 登录主机 B。
   b) 创建临时目录，并将路径切换到这个临时目录：

   ```
   mkdir cert_files
   cd cert_files
   ```

   c) 创建 2048 位的 RSA 密钥：

   ```
   openssl genrsa -out cakey.pem 2048
   ```

   d) 创建本地 CA 的自签名证书：

   ```
   openssl req -new -x509 -days 1095 -key cakey.pem -out \
   cacert.pem -sha256 -subj "/C=US/L=Austin/O=IBM/CN=my CA"
   ```

   e) 检查 CA 证书：

   ```
   openssl x509 -noout -text -in cacert.pem
   ```

2. 要在系统管理程序中创建客户机和服务器密钥及证书，请完成下列步骤：
   a) 创建密钥：

   ```
   openssl genrsa -out serverkey.pem 2048
   openssl genrsa -out clientkey.pem 2048
   ```

   b) 创建用于服务器的证书签名请求：

   切记: 将服务器证书请求中使用的 kvmhost.company.org 地址更改为系统管理程序主机的标准域名。

   ```
   openssl req -new -key serverkey.pem -out serverkey.csr \
   -subj "/C=US/O=IBM/CN=kvmhost.company.org"
   ```

   c) 创建用于客户机的证书签名请求：

   ```
   openssl req -new -key clientkey.pem -out clientkey.csr \
   -subj "/C=US/O=IBM/OU=virtualization/CN=root"
   ```

   d) 创建客户机和服务器证书：

   ```
   openssl x509 -req -days 365 -in clientkey.csr -CA cacert.pem \
   -CAkey cakey.pem -set_serial 1 -out clientcert.pem
   openssl x509 -req -days 365 -in serverkey.csr -CA cacert.pem \
   -CAkey cakey.pem -set_serial 94345 -out servercert.pem
   ```

   e) 检查密钥：

   ```
   openssl rsa -noout -text -in clientkey.pem
   ```

openssl rsa -noout -text -in serverkey.pem

f) 检查证书：

openssl x509 -noout -text -in clientcert.pem

openssl x509 -noout -text -in servercert.pem

3. 要将密钥和证书分发到主机服务器，请完成下列步骤：

a) 将 CA 证书 cacert.pem 文件复制到以下目录：/etc/pki/CA

cp cacert.pem /etc/pki/CA/cacert.pem

b) 创建 /etc/pki/libvirt 目录，并将 servercert.pem 服务器证书文件复制到 /etc/pki/libvirt 目录。确保只有 root 用户能够访问专用密钥。

mkdir /etc/pki/libvirt

cp servercert.pem /etc/pki/libvirt/.

chmod -R o-rwx /etc/pki/libvirt

切记: 如果密钥或证书的命名不正确，或者复制到错误的目录中，那么授权将失败。

c) 创建 /etc/pki/libvirt/private 目录，并将 serverkey.pem 服务器密钥文件复制到 /etc/pki/libvirt/private 目录。确保只有 root 用户能够访问专用密钥。

mkdir /etc/pki/libvirt/private

cp serverkey.pem /etc/pki/libvirt/private/.

chmod -R o-rwx /etc/pki/libvirt/private

切记: 如果密钥或证书的命名不正确，或者复制到错误的目录中，那么授权将失败。

d) 确认已正确放置文件：

find /etc/pki/CA/*|xargs ls -l

ls -lR /etc/pki/libvirt

ls -lR /etc/pki/libvirt/private

切记: 如果密钥或证书的命名不正确，或者复制到错误的目录中，那么授权将失败。

4. 要将密钥和证书分发到客户机或管理站，请完成下列步骤：

a) 登录主机 A。

b) 在不更改文件名的情况下，将 CA 证书 cacert.pem 从该主机复制到主机 A 中的 /etc/pki/CA 目录。

scp kvmhost.company.org:/tmp/cacert.pem /etc/pki/CA/

c) 将客户机证书 clientcert.pem 文件从主机 B 复制到 /etc/pki/libvirt 目录。使用缺省文件名，并确保只有 root 用户能够访问专用密钥。

mkdir /etc/pki/libvirt/

scp kvmhost.company.org:/tmp/clientcert.pem /etc/pki/libvirt/.

chmod -R o-rwx /etc/pki/libvirt

切记: 如果密钥或证书的命名不正确，或者复制到错误的目录中，那么授权将失败。

d) 将客户机密钥 clientkey.pem 从该主机复制到 /etc/pki/libvirt/private 目录。使用缺省文件名，并确保只有 root 用户能够访问专用密钥。

mkdir /etc/pki/libvirt/private

scp kvmhost.company.org:/tmp/clientkey.pem /etc/pki/libvirt/private/.

chmod -R o-rwx /etc/pki/libvirt/private

切记: 如果密钥或证书的命名不正确，或者复制到错误的目录中，那么授权将失败。

e) 确认已正确放置文件：

ls -lR /etc/pki/libvirt

ls -lR /etc/pki/libvirt/private

5. 要编辑 libvirtd 守护程序配置，请完成下列步骤：
   a) 登录主机 B。
   b) 建立 /etc/sysconfig/libvirtd 文件和 /etc/libvirt/libvirtd.conf 文件的副本。
   c) 编辑 /etc/sysconfig/libvirtd 文件，并确保将 **--listen** 参数传递到 libvirtd 守护程序。此步骤确保 libvirtd 守护程序侦听网络连接。
   d) 编辑 /etc/libvirt/libvirtd.conf 文件，并在 libvirtd.conf 文件中使用 **tls_allowed_dn_list** 伪指令来配置一组允许的主体集。

   要点: 主体集中的字段顺序必须与用来创建证书的字段顺序相同。
   e) 重新启动 libvirtd 守护程序服务，以使更改生效：

   /etc/init.d/libvirtd restart
6. 要更改防火墙配置，请访问安全级别配置，并添加 TCP 端口 16514 作为可信端口。
7. 要确认远程管理正常工作，请在主机 A 上运行以下命令：

   virsh -c qemu+tls://kvmhost.company.org/system list --all

配置 TCP 协议
TCP 协议仅可用于测试。

关于此任务
假设： Linux KVM 代理程序安装在主机 A 上。您想要远程监视主机 B 上的系统管理程序。

过程

1. 登录主机 B。
2. 编辑 /etc/libvirt/libvirtd.conf 文件，确保启用 **listen_tcp** 参数，而且 **tcp_port** 参数的值设置为缺省值 16509。
3. 编辑 /etc/libvirt/libvirtd.conf 文件，以将 **auth_tcp** 参数设置为“none”。此步骤指示 TCP 不要认证连接。
4. 在主机 B 上，通过使用 **--listen** 标志来运行，或通过编辑 /etc/sysconfig/libvirtd 文件并取消注释 LIBVIRTD_ARGS="--listen" 行，以侦听方式重新启动 **libvirt** 守护程序。
5. 要验证连接，请运行以下命令：

   virsh -c qemu+tcp://kvmhost.company.org:port/system

   如果您未更改缺省 TCP 端口，请省略此命令的 **:port** 部分。

   要点: 如果 **virsh** 命令成功，Linux KVM 代理程序 会连接到系统管理程序。

下一步做什么
通过完成第 415 页的『配置 RHEVH 服务器连接』中描述的步骤，配置代理程序。

配置 RHEVM 服务器连接

要配置与 RHEVM 服务器的连接，您必须运行脚本并对提示做出响应。

开始之前

1. 下载位于以下路径的安全证书：

```
https://RHEVM-HOST:RHEVM-PORT/ca.crt
```

其中

RHEVM-HOST
主机的名称。

RHEVM-PORT
RHEVM 环境中使用的端口。

2. 使用 *keytool* 实用程序来导入安全证书文件，以生成本地密钥库文件：

   keytool -import -alias ALIAS -file CERTIFICATE_FILE -keystore KEYSTORE_FILE

   示例：**keytool -import -alias RHEVM36vmwt9 -file hjs495-vmw-t-9.cer -keystore RHEVM36KeyStore**

   其中

   ALIAS
   这是对添加到代理程序证书信任库的每个证书的唯一引用，例如，来自 *datasource.example.com* 的证书的相应别名为 *datasource*。

   CERTIFICATE_FILE
   要添加到信任库的数据源证书的完整路径和文件名。

   KEYSTORE_FILE
   要指定的密钥库文件的名称。

   提示: *keytool* 实用程序是随 Java 运行时环境 (JRE) 一起提供。这个密钥库文件存储在您运行此命令的位置。

3. 请确保连接至 RHEVM 的用户是具有 SuperUser 角色的管理员。您可使用具有此角色的现有用户标识，也可完成第 409 页的『创建用户和授予所需许可权』所述的步骤以创建新用户标识。

过程

1. 在命令行上，运行以下命令：

   install_dir/bin/linux_kvm-agent.sh config instance_name

   示例：**/opt/ibm/apm/agent/bin/linux_kvm-agent.sh config instance_name**

   其中

   instance_name
   要对实例指定的名称。

   install_dir
   代理程序的安装路径。

2. 对提示做出响应，并指定配置参数的值。

   有关配置参数的信息，请参阅第 415 页的『用于连接至 RHEVM 服务器的配置参数』。

3. 运行以下命令以启动代理程序：

   install_dir/bin/linux_kvm-agent.sh start instance_name

   示例：**/opt/ibm/apm/agent/bin/linux_kvm-agent.sh start instance_name**

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

配置 RHEVH 服务器连接

要配置与 RHEVH 服务器的连接，您必须运行脚本并对提示做出响应。

开始之前

- 请确保连接至 RHEVM 的用户是 root 用户。您可使用现有的用户标识，或者完成第 409 页的『创建用户和授予所需许可权』所述的步骤以创建新用户标识。
- 完成第 410 页的『配置协议』所述的步骤，配置要用来连接至 RHEVH 服务器的协议。

过程

1. 在命令行上，运行以下命令：

   install_dir/bin/linux_kvm-agent.sh config instance_name

   示例：**/opt/ibm/apm/agent/bin/linux_kvm-agent.sh config instance_name**

   其中

   instance_name
   要对实例指定的名称。

   install_dir
   代理程序的安装路径。
2. 对提示做出响应，并指定配置参数的值。

   有关配置参数的信息，请参阅第 416 页的『用于连接至 RHEVH 服务器的配置参数』。
3. 运行以下命令以启动代理程序：

   install_dir/bin/linux_kvm-agent.sh start instance_name

   示例：**/opt/ibm/apm/agent/bin/linux_kvm-agent.sh start instance_name**

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

用于连接至 RHEVM 服务器的配置参数

您可修改用于将代理程序连接到 RHEVM 服务器的配置参数的缺省值。

下表包含配置参数的详细描述。

表 *169.* 用于连接至 *RHEVM* 服务器的配置参数的名称和描述

参数名称	描述	必填字段
编辑 Monitoring Agent for Linux KVM 设置	指出您可开始编辑配置参数的缺省值。请输入 1 （Yes）（这也是缺省值）以继续。	是
最大数据提供程序日志文件数	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。	是
每个数据提供程序日志的最大大小 (KB)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。	是

表 *169.* 用于连接至 *RHEVM* 服务器的配置参数的名称和描述 *(续)*

参数名称	描述	必填字段
数据提供程序日志中的详细信息级别	可以包含在数据提供程序创建的日志文件中的详细信息级别。缺省值是 4 (Info)。以下值有效： · 1 = Off：不记录任何消息。 · 2 = Severe：仅记录错误。 · 3 = Warning：在 Severe 级别所记录的所有错误和消息，以及可能会导致不良行为的潜在错误。 · 4 = Info：在 Warning 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的高级别参考消息。 · 5 = Fine：在 Info 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的低级别参考消息。 · 6 = Finer：在 Fine 级别所记录的所有错误和消息，以及高度详细的参考消息，例如性能概要分析信息和调试数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 7 = Finest：在 Fine 级别所记录的所有错误和消息，以及最详细的参考消息，包括低级别编程消息和数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 8 = All：记录所有错误和消息。	是
编辑系统管理程序设置	指示您是否要编辑 RHEVH 服务器连接的参数。请输入 5 (Next)，因为您要配置 RHEVM 服务器连接。缺省值为 5 (Next)。	是
编辑 RHEVM 连接详细信息设置	指示您是否要编辑 RHEVM 服务器连接的参数。请输入 1 (Add) 以继续。缺省值为 5 (Next)。 **要点:** 指定所有配置参数的值之后，系统会再次提示您指示是否要继续编辑参数。请输入 5 (Exit)。	是
RHEVM 标识	对所要连接的 RHEVM 指定的唯一用户名。	是
主机	用于连接到 RHEVM 服务器的数据源的主机名或 IP 地址。	是
用户	具有足够特权来连接至 RHEVM 服务器的数据源用户名。	是
密码	用来连接至 RHEVM 服务器的用户名的密码。	是
重新输入密码	在**密码**字段中指定的密码。	是
端口	用来连接至 RHEVM 服务器的端口号。	是
域	用户所属的域。	是
密钥库路径	您使用 **keytool** 命令所创建的本地密钥库文件的文件路径和名称。	是

用于连接至 RHEVH 服务器的配置参数

您可修改用于将代理程序连接到 RHEVH 服务器的配置参数的缺省值。

下表包含配置参数的详细描述。

表 *170.* 用于连接至系统管理程序的配置参数的名称和描述

参数名称	**描述**	**必填字段**
编辑 Monitoring Agent for Linux KVM 设置	指出您可开始编辑配置参数的缺省值。请输入 1 （Yes）（这也是缺省值）以继续。	是
最大数据提供程序日志文件数	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。	是
每个数据提供程序日志的最大大小 (KB)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。	是
数据提供程序日志中的详细信息级别	可以包含在数据提供程序创建的日志文件中的详细信息级别。缺省值是 4 (Info)。以下值有效： · 1 = Off：不记录任何消息。 · 2 = Severe：仅记录错误。 · 3 = Warning：在 Severe 级别所记录的所有错误和消息，以及可能会导致不良行为的潜在错误。 · 4 = Info：在 Warning 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的高级别参考消息。 · 5 = Fine：在 Info 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的低级别参考消息。 · 6 = Finer：在 Fine 级别所记录的所有错误和消息，以及高度详细的参考消息，例如性能概要分析信息和调试数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 7 = Finest：在 Fine 级别所记录的所有错误和消息，以及最详细的参考消息，包括低级别编程消息和数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 8 = All：记录所有错误和消息。	是
编辑系统管理程序设置	指示您是否要编辑系统管理程序连接的参数。请输入 1 （Add）。缺省值为 5 （Next）。	是
系统管理程序标识	对所要连接的 RHEVH 指定的唯一用户名。	是
主机	用于连接到 RHEVH 服务器的数据源的主机名或 IP 地址。	是
用户	具有足够特权来连接至 RHEVM 服务器的数据源用户名。	是
远程传输	本地 `libvirt` API 用于连接到远程 `libvirt` API 的协议。缺省值为 1。下列值有效： · 1 = SSH · 2 = TLS · 3 = TCP（未加密 - 建议不要用于生产用途）	是
端口	传输协议在连接至 `libvirt` API 时所用的端口。缺省值是 22。 **要点:** 仅当标准端口（22 用于 SSH，16514 用于 TLS，16509 用于 TCP）已更改时，才需要此端口。	是
域	用户所属的域。	是

表 170. 用于连接至系统管理程序的配置参数的名称和描述 (续)		
参数名称	**描述**	**必填字段**
连接实例类型	指示本地 libvirt API 是连接至特许系统驱动程序，还是连接至各用户的非特许会话驱动程序。缺省值为 1。下列值有效: · 1 = 系统 · 2 = 会话	是
编辑 RHEVM 连接详细信息设置	指示您是否要编辑 RHEVM 服务器连接的参数。请输入 1 （Add）以继续。缺省值为 5 （Next）。 **要点:** 指定所有配置参数的值之后，系统会再次提示您指示是否要继续编辑参数。请输入 5 （Next）。	是

配置 MariaDB 监视

您必须配置 MariaDB 代理程序，以便代理程序能够收集数据，以监视 MariaDB 服务器资源的可用性和性能。请参阅以下先决条件以针对远程和本地监视配置 MariaDB 代理程序。

开始之前

确保在您的环境中满足 MariaDB 代理程序的系统需求。有关最新系统需求信息，请参阅 MariaDB 代理程序的软件产品兼容性报告 (SPCR)。

关于此任务

MariaDB 代理程序是单实例代理程序。安装此代理程序之后，您必须对其进行手动配置。可以在 Windows 和 Linux 操作系统上配置代理程序。代理程序需要实例名称和 MariaDB 服务器用户凭证以进行配置。“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。受管系统名称至多可以包含 32 个字符。指定的实例名称至多可包含 28 个字符（不包括主机名长度）。例如，如果指定 MariaDB 作为实例名称，那么受管系统名称为 MariaDB:hostname:MJ。

要点: 如果指定长实例名称，那么会截断受管系统名称，并且也不会显示代理程序代码。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置代理程序。更新配置值后，启动代理程序以应用更新后的值。

过程

要在 Windows 操作系统上配置代理程序，请完成下列步骤:

1. 单击**开始>所有程序>IBM Monitoring 代理程序>IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，完成以下步骤:
   a) 双击 **Monitoring Agent for MariaDB** 模板。
   b) 在 **Monitoring Agent for MariaDB** 窗口中，指定实例名称并单击**确定**。
3. 在 **Monitoring Agent for MariaDB** 窗口中，完成以下步骤:
   a) 在 **IP 地址**字段中，输入要远程监视的 MariaDB 服务器的 IP 地址。如果在要监视的服务器上安装代理程序，那么保留缺省值。
   b) 在 **JDBC 用户名**字段中，输入 MariaDB 服务器用户的名称。缺省值为 root。
   c) 在 **JDBC 密码**字段中，输入 JDBC 用户的密码。
   d) 在**确认 JDBC 密码**字段中，再次输入密码。
   e) 在 **JDBC JAR 文件**字段中，单击**浏览**并找到包含 MariaDB 连接器 Java 文件的目录，然后将其选中。
   f) 单击**下一步**。

g) 在 **JDBC 端口号**字段中，指定 JDBC 服务器的端口号。
缺省端口号为 3306。

h) 从 **Java 跟踪级别**列表中，选择 Java 的跟踪级别。
缺省值为 Error。

i) 单击**确定**。
该实例将显示在 **IBM Performance Management** 窗口中。

4. 右键单击 **Monitoring Agent for MariaDB** 实例，然后单击**启动**。

切记: 要重新配置代理程序，请在 **IBM Performance Management** 窗口中完成以下步骤：

a. 停止想要配置的代理程序实例。
b. 右键单击 **Monitoring Agent for MariaDB** 实例，然后单击**重新配置**。
c. 重复步骤 3 和 4。

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Linux 系统上配置代理程序

在 Linux 操作系统上，可以运行配置脚本并响应提示来配置代理程序。

过程

要在 Linux 操作系统上配置代理程序，请完成以下步骤：

1. 在命令行上，运行以下命令：

```
install_dir/bin/mariadb-agent.sh config instance_name
```

其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 MariaDB 代理程序的安装目录。

2. 当系统提示您输入以下参数的值时，可按 **Enter** 键以接受缺省值，或指定不同的值并按 **Enter** 键。

- IP 地址
- JDBC 用户名
- JDBC 密码
- 重新输入 JDBC 密码
- JDBC JAR 文件
- JDBC 端口号（缺省端口号为 3306。）
- Java 跟踪级别（缺省值为 Error。）

有关配置参数的更多信息，请参阅第 498 页的『使用静默响应文件来配置代理程序』。

3. 运行以下命令以启动代理程序：

```
install_dir/bin/mariadb-agent.sh start instance_name
```

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可以使用静默响应文件来配置 MariaDB 代理程序。在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

要使用静默响应文件配置代理程序，请完成以下步骤：

切记： 此过程假设此代理程序安装在以下缺省路径中：

Windows `C:\IBM\APM`

Linux `opt/ibm/apm/agent`

如果代理程序安装在其他位置，请替换指示信息中的路径。另外，编辑静默响应文件中的 **AGENT_HOME** 参数以指定代理程序的安装路径。

1. 在文本编辑器中，打开位于以下路径的响应文件：

   Linux *install_dir*`/samples/mariadb_silent_config.txt`

   Windows *install_dir*`\samples\mariadb_silent_config.txt`

   其中，*install_dir* 是 MariaDB 代理程序的安装目录
2. 在响应文件中，对下列参数指定值：
   - 对于 **Server Name** 参数，请指定要远程监视的 MariaDB 服务器的 IP 地址。否则，保留缺省值为 `localhost`。
   - 对于 **JDBC user name** 参数，保留缺省用户名 `root` 值，或者指定具有 INFORMATION_SCHEMA 表的查看权限的用户的名称。
   - 对于 **JDBC password** 参数，输入 JDBC 用户密码。
   - 对于 **JDBC Jar File** 参数，请保留缺省路径（如果此路径为 Java JAR 文件的 MariaDB 接口的路径）。否则，请输入正确的路径。接口位于下列缺省路径中：

     Linux `/usr/share/java/mariadb-connector-java.jar`

     Windows `C:\Program Files (x86)\MariaDB\mariadb-connector-java.jar`
   - 对于 **JDBC port number** 参数，请保留缺省端口号 3306 或指定另一个端口号。
   - 对于 **Java trace level** 参数，请保留缺省值 `Error` 或根据 IBM 支持指示信息指定另一个级别。
3. 保存并关闭响应文件，然后运行以下命令以更新代理程序配置设置：

   Linux *install_dir*`/bin/mariadb-agent.sh config` *instance_name install_dir*`/samples/mariadb_silent_config.txt`

   Windows *install_dir*`\BIN\mariadb-agent.bat config` *instance_name install_dir*`\samples\mariadb_silent_config.txt`

   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 MariaDB 代理程序的安装目录。

   要点： 请确保包含指向静默响应文件的绝对路径。否则，仪表板不显示代理程序数据。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

配置 Microsoft Active Directory 监视

完成安装后，系统将自动配置并启动 Monitoring Agent for Microsoft Active Directory。

开始之前

复审必备硬件和软件，请参阅 Microsoft Active Directory 代理程序的软件产品兼容性报告

要查看仪表板中所有属性的数据，请完成以下任务：

- 第 421 页的『以管理员用户身份运行 Microsoft Active Directory 代理程序』
- 第 421 页的『配置本地环境变量』

关于此任务

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

以管理员用户身份运行 Microsoft Active Directory 代理程序

必须具有管理权限才能运行 Microsoft Active Directory 代理程序。

关于此任务

所有数据集可用于作为 Administrators 组的成员的用户。在此任务中，您将创建用户，将管理员权限分配给该用户，并将代理程序的用户帐户更改为此用户。

过程

1. 单击**开始 > 所有程序 > 管理工具 > Active Directory 用户和计算机**。
2. 要展开您希望在其中创建用户的域，请单击域名旁边的加号 (+)。
3. 右键单击**用户**，然后单击**新建 > 用户**。
4. 要创建新用户，请打开**新建对象 - 用户**向导。
   缺省情况下，新用户是 Domain Users 组的成员。
5. 右键单击在 Domain Users 组中创建的新用户，并单击**属性**。这将显示“**用户名属性**”窗口。用户名为新用户的名称。
6. 在“**用户名属性**”窗口中，完成下列步骤：
   a) 单击**成员**选项卡。在**成员**区域中，添加 Administrators 组。
   b) 单击**应用**，然后单击**确定**。
7. 单击**开始 > 运行**，然后输入 `services.msc`。
8. 在“**服务**”窗口中，完成下列步骤：
   a) 右键单击 **Monitoring Agent for Active Directory 服务**，并单击**属性**。
   b) 在“**Monitoring Agent for Active Directory 属性**”窗口的**登录**选项卡上，单击**此帐户**。输入用户凭证。
   c) 单击**应用**，然后单击**确定**。
9. 重新启动代理程序服务。

配置本地环境变量

您必须为环境变量指定值以在仪表板中查看 Sysvol 复制数据。您也可以选择更新高速缓存时间间隔值来启用或禁用高速缓存。

过程

1. 在 **IBM Performance Management** 窗口中，从**操作**菜单中单击**高级 > 编辑 ENV 文件**。

2. 在 K3ZENV 文件中，更改以下环境变量的值。

   ADO_CACHE_INTERVAL
   确定是启动还是停止高速缓存，用于设置高速缓存时间间隔值。高速缓存时间间隔是连续两次数据收集间隔的持续时间，以秒为单位。您可以为高速缓存时间间隔指定任意正整数值以启动高速缓存。可以为高速缓存时间间隔指定值 0 以停止高速缓存。缺省情况下，启动高速缓存，并且高速缓存时间间隔值设置为 1200。

   ADO_SYSVOL_FORCE_REPLICATION_FLAG
   确定是启用还是禁用代理程序启动的强制复制。此变量的缺省值为 TRUE。要禁用强制复制，请将此变量的值更改为 FALSE。

   ADO_SYSVOL_REPLICATION_TEST_INTERVAL
   确定两次 Sysvol 复制测试之间的时间间隔，以分钟为单位。此变量的缺省值为 0 分钟。要完成 Sysvol 复制测试，请确保此变量值大于 0。

   ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL
   确定代理程序在完成 Sysvol 复制测试后等待验证 Sysvol 复制结果的时间量，以分钟为单位。

   ADO_SYSVOL_REPLICATION_TEST_INTERVAL 变量的值必须大于 **ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL** 变量的值。您可以将以下值用于这些变量：

   ADO_SYSVOL_REPLICATION_TEST_INTERVAL：1440
   ADO_SYSVOL_REPLICATION_TEST_VERIFICATION_INTERVAL：30

   为两个环境变量分配有效值后，Active Directory 代理程序在受管系统的 Sysvol 共享文件夹中创建一个文件，并初始化强制的 Sysvol 复制。这种强制的复制将从受管系统初始化到 Sysvol 复制合作伙伴的 Sysvol 共享文件夹。验证复制测试结果后，代理程序将移除从受管系统和 Sysvol 复制合作伙伴创建和复制的文件。
3. 可选： 在 K3ZENV 文件中，添加 **APM_ATTRIBUTES_ENABLE_COLLECTION** 环境变量，并将它的值设置为 Yes，以便在**属性详细信息**选项卡中查看下列数据集的数据。

   · 服务
   · 复制
   · 文件复制服务
   · 已移动或已删除的组织单元
   · LDAP
   · 安全帐户管理器
   · DFS
   · 地址簿
   · 事件日志
   · 密码设置对象

   切记: 如果要对这些数据集禁用数据收集，请将 **APM_ATTRIBUTES_ENABLE_COLLECTION** 环境变量的值设置为 No。
4. 重新启动 Microsoft Active Directory 代理程序。

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

以非管理员用户身份运行 Microsoft Active Directory 代理程序

您可以非管理员用户身份运行 Log File Agent。

关于此任务

您可以非管理员用户身份来运行 Active Directory 监视代理程序；但是，“信任拓扑”属性和“Sysvol 复制”属性可能不可用。这些属性仅适用于域用户。

要查看“信任拓扑”属性，非管理员用户必须具有下列注册表许可权：

- 授予对 HKEY_LOCAL_MACHINE\SOFTWARE\Candle directory 的完全访问权。
- 授予对 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Perflib 目录的读访问权。

要查看“Sysvol 复制”属性，非管理员用户必须对域中所有域控制器上的 Sysvol 文件夹具有完全访问权。

要点: 当 Microsoft Active Directory 代理程序以非管理员用户身份运行时，在 APM 用户界面上，“服务”属性组中的某些服务会将“当前状态”和“启动类型”属性的值显示为“未知”。

下表列出 Active Directory 代理程序的属性组，它们会向域用户和性能监视用户显示数据。

表 *171.* 域用户和性能监视用户的属性组

用户权限	属性组
域用户	· RID 池信息 · 服务 · 事件日志 · DNS · 集成式 DNS AD 详细信息 · 集成式 DNS AD · DHCP · 信任 · 组策略对象 · 丢失和找到的对象 · 交换目录服务 · 复制冲突对象 · LDAP 属性 · 根目录服务器 · 容器 · 复制合作伙伴 · 域控制器可用性 · 复制合作伙伴等待时间 · 森林拓扑

表 171. 域用户和性能监视用户的属性组 (续)	
用户权限	**属性组**
域用户和性能监视用户	针对域用户列出的所有属性组，以及下列额外的属性组： · 地址簿 · 复制 · 目录服务 · 知识一致性检查程序 · Kerberos 密钥分发中心 · 轻量级目录访问协议 · 本地安全权限 · 名称服务提供程序 · 安全帐户管理器 · 文件复制服务 · 分布式文件系统复制 · DFS 复制连接 · DFS 复制的文件夹 · DFS 服务卷 · 域控制器性能 · 远程访问服务器 · 直接访问服务器 · Netlogon 属性

注: 另外，对于隶属于 *Administrators* 组的用户，下列属性组会显示数据：

· Active Directory 数据库信息
· 已移动或删除的组织单位
· 密码设置对象

有关信息，请参阅第 421 页的『配置 Microsoft Active Directory 监视』

过程

1. 单击**开始 > 程序 > 管理工具 > Active Directory 用户和计算机**。
2. 通过单击域名旁边的加号 (+)，展开要在其中创建用户的域。
3. 右键单击**用户**，然后单击**新建 > 用户**。
4. 使用“**新建对象 - 用户**”向导创建新用户。缺省情况下，新用户是 **Domain Users** 组的成员。
5. 右键单击在 *Domain Users* 组中创建的新用户，并单击**属性**。“**Username 属性**”窗口将会打开，其中，*Username* 是新用户的名称。在“**Username 属性**”窗口中完成下列步骤：
   a) 单击**隶属于**选项卡。在“**隶属于**”区域中，添加 **Performance Monitor Users** 组。
   b) 单击**应用**，然后单击**确定**。
6. 转至 `Candle_Home` 目录。缺省路径为 `C:\IBM\APM`。
7. 右键单击 `APM` 文件夹，然后单击**属性**。“**APM 属性**”窗口随即打开。在“**APM 属性**”窗口中完成下列步骤：
   a) 在**安全**选项卡上，单击**编辑**。
   b) 单击**添加**以添加新用户，并将完全访问权授予此用户。
   c) 单击**应用**，然后单击**确定**。

8. 单击**开始 > 运行**，然后输入 services.msc。“**服务**”窗口随即打开。在“**服务**”窗口中完成下列步骤：
   a) 右键单击 **Monitoring Agent for Active Directory** 服务，并单击**属性**。
   b) 在“**Active Directory 属性**”窗口的**登录**选项卡上，单击**此帐户**。输入用户凭证。
   c) 单击**应用**，然后单击**确定**。
9. 重新启动代理程序服务。

为属性组 AD_Services_Status 配置域服务

确定 Server Status 时，可在 Services.properties 中配置
要使用或者要排除的 MS Active Directory 域服务。属性组 AD_Services_Status 及其情境适用于 Windows Server 2012 和更高版本。

关于此任务

Services.properties 文件包含以下缺省 MS Active Directory 域服务及其配置。
True 表示在确定“服务器状态”值时将考虑此服务。
False 表示在确定“服务器状态”值时将不会考虑此服务。

表 *172. MS Active Directory* 域服务和配置设置。

MS Active Directory 域服务	**缺省设置**
DFS 复制	true
远程过程调用 (RPC)	false
DNS 客户机	true
DNS 服务器	true
组策略客户机	false
站点间消息传递	true
Kerberos 密钥分发中心	true
NetLogon	true
Windows 时间	true
DHCP 客户机	false
Active Directory Web Service	false
Active Directory 联合服务	false

注: 需要重新启动代理程序才能在 Windows Server 2012 和更高版本上为属性组 AD_Services_Status 启用数据收集。

过程

1. 停止代理程序。
2. 查找要修改的 Services.properties 文件（如果有）。
   对于 32 位代理程序，Services.properties 文件位于 *CANDLE_HOME*\TMAITM6\ 中。
   对于 64 位代理程序，Services.properties 文件位于 *CANDLE_HOME*\TMAITM6_x64\ 中。
   CANDLE_HOME 是代理程序安装目录。
3. 如果希望在确定“服务器状态”时考虑“域服务”，请将其值设置为 true。
   如果不希望在确定“服务器状态”时考虑“域服务”，请将其值设置为 false。
   保存并关闭该文件。
4. 启动代理程序。

升级 Microsoft Active Directory 代理程序

您可将 MS Active Directory 代理程序升级到最新版本。

开始之前

确保在安装代理程序的机器中包含最新版本的安装程序中提供的 `installAPMAgents.bat` 文件。

关于此任务

要将代理程序升级至最新版本，请完成以下过程。

过程

1. 登录到已安装代理程序的机器。
2. 启动命令提示符，运行源自最新发行版的安装程序的 `installAPMAgents.bat` 文件。
3. 输入现有代理程序所在的安装目录，然后按 Enter 键。
4. 命令提示符会显示基本代理程序版本和要升级的目标代理程序版本。请按 Enter 键以继续。
5. 代理程序升级成功后，升级后的代理程序版本显示在 **IBM Performance Management** 窗口上。
6. 在 **IBM Performance Management** 窗口上，右键单击代理程序，并从下拉菜单中选择**重新配置**。
7. 要在**应用程序性能仪表板**上反映升级的代理程序版本，请登录到 APM 服务器，并使用以下命令重新启动 APM 服务器组件。
   a. **`apm stop_all`**
   b. **`apm start_all`**
8. 在 **IBM Performance Management** 窗口上，右键单击代理程序并从下拉菜单中选择**重新启动**。

结果

已升级的代理程序将反映在**应用程序性能仪表板**上。

注: 可能需耗时 30 分钟以上才能在**应用程序性能仪表板**上显示升级的代理程序。

配置 Microsoft Cluster Server 监视

您必须配置 Monitoring Agent for Microsoft Cluster Server，以便代理程序能够收集集群服务器数据。使用静默响应文件来配置代理程序。

开始之前

请确保完成了以下任务：

- 为代理程序创建空的资源组。
- 在 Windows Server 2008、2012、2016 和 2019 系统上的代理程序的资源组中创建通用服务集群资源。
- 确保连接到 Microsoft Cluster Server 环境或应用程序的用户具有管理员特权。使用具有管理员特权的现有用户或创建新的用户。通过向“管理员”组添加新用户，向新用户分配管理员特权。

  切记: 要配置 Microsoft Cluster Server 代理程序，可使用具有管理员特权的本地用户或域用户。

查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft Cluster Server 代理程序 的软件产品兼容性报告 (SPCR)。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

关于此任务

Microsoft Cluster Server 代理程序 是一个单实例代理程序。您必须在集群中的每个节点上以相同方式手动安装并配置代理程序。要配置代理程序，请参阅第 427 页的『使用静默响应文件来配置代理程序』。

在 Windows Server 2008、2012、2016 和 2019 系统上创建通用服务集群资源

您必须将集群代理程序服务作为资源进行添加，以便代理程序能够监视该集群服务器。

开始之前

请确保在集群中的每个节点上停止该代理程序。

过程

要创建通用服务集群资源，请完成下列步骤：

1. 在任意一个集群节点上打开“**故障转移集群管理器**”。
2. 完成下列其中一个步骤：
   - 对于 Windows Server 2008：

     在导航窗格中，右键单击**服务和应用程序**，然后单击**更多操作 > 创建空服务或应用程序**。新服务将显示在“服务和应用程序”列表中。将新创建的服务重命名。
   - 对于 Windows Server 2012：

     在导航窗格中，右键单击**角色**，然后单击**更多操作 > 创建角色**。新服务将显示在“角色”列表中。
   - 对于 Windows Server 2016 和 2019：

     在导航窗格中，右键单击**角色**，然后单击**配置角色**。这样会显示新服务。
3. 右键单击新服务，然后单击**添加资源 > 通用服务**。
4. 在“**新建资源向导**”窗口中，选择**监视 Agent for Microsoft Cluster Server** 并单击**下一步**。
5. 在后续窗口中单击**下一步**，直到您看到**完成**按钮为止。
6. 单击**完成**。

   代理程序服务已添加为资源。
7. 右键单击**监视 Agent for Microsoft Cluster Server** 资源，然后单击**使资源联机**。

结果

该代理程序已在首选节点上启动。

使用静默响应文件来配置代理程序

静默响应文件包含 Microsoft Cluster Server 代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件以使用不同的配置参数值来配置代理程序。

开始之前

创建包含要修改的配置参数的响应文件。如果要修改缺省配置参数，请编辑响应文件。

关于此任务

您可以使用静默响应文件来配置代理程序。

过程

1. 打开位于以下路径的静默响应文件：*install_dir*\samples\microsoft_cluster_server_silent_config.txt
2. 对于 **CTIRA_HOSTNAME** 环境变量，请指定集群名称作为值。
3. 在每个集群节点上，运行以下命令：**install_dir\BIN\microsoft_cluster_server-agent.bat config install_dir\samples\microsoft_cluster_server_silent_config.txt**

下一步做什么

将用户帐户从本地用户更改为域用户。

更改用户帐户

配置 Microsoft Cluster Server 代理程序 后，您可以将用户帐户从本地用户更改为域用户。

关于此任务

缺省情况下，代理程序使用本地用户帐户来运行。代理程序必须使用域用户来运行，这样才能从单个节点监视集群中的所有节点。

过程

要更改用户帐户，请完成下列步骤：

1. 打开 **IBM Performance Management** 窗口。
2. 右键单击代理程序，然后单击**更改启动**。
3. 输入域登录凭证。
4. 在一个节点上打开“**故障转移集群管理器**”，然后启动集群服务。

结果

代理程序已在该节点上启动。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Performance Management Console 的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

配置 Microsoft Exchange 监视

必须配置 Monitoring Agent for Microsoft Exchange Server 以监视 Exchange Server 的可用性和性能。

开始之前

在配置代理程序之前，请确保完成了以下任务：

- 第 429 页的『创建用户』
- 第 431 页的『向 Exchange Server 用户分配管理员权限』
- 第 433 页的『使 Exchange Server 用户成为本地管理员』
- 第 434 页的『配置 Exchange Server 以获取可达性』
- 第 435 页的『配置代理程序以在域用户下运行』
- 查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft Exchange Server 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

安装代理程序之后，可以启动 Microsoft Exchange Server 代理程序。但是，需要手动配置才能查看所有代理程序属性的数据。

- 要本地配置代理程序，请参阅第 435 页的『本地配置代理程序』。
- 要使用静默响应文件配置代理程序，请参阅第 438 页的『使用静默响应文件来配置代理程序』。

创建用户

您可以手动或通过运行新建用户实用程序为 Exchange Server 上的代理程序创建用户。必须在想要监视的每个 Exchange Server 上创建用户。

开始之前

安装 Microsoft Exchange Server 代理程序。要创建用户，您必须是具有 Microsoft Exchange Server 上的完全管理员权限的域管理员。

关于此任务

使用以下某个过程来创建用户：

- 第 429 页的『在 Exchange Server 2007 和 2010 上创建用户』
- 第 430 页的『在 Exchange Server 2013 上创建用户』
- 第 430 页的『通过运行“新建用户”实用程序创建用户』

在 Exchange Server 2007 和 2010 上创建用户

您必须为 Exchange Server 2007 和 2010 上的代理程序创建用户，从而使该代理程序可以与您想要监视的 Exchange Server 通信并向其认证。

过程

要创建用户，请完成以下步骤：

1. 单击**开始 > 程序 > Microsoft Exchange Server 2007 > Exchange 管理控制台**。“**Exchange 管理控制台**”窗口随即打开。
2. 在控制台树中，单击**收件人配置中的邮箱**。
3. 在“操作”窗格中，单击**新建邮箱**。“新建邮箱”向导会打开。
4. 在“**简介**”页面上，单击**用户邮箱**。
5. 在“**用户类型**”页面上，单击**新用户**。
6. 在“**用户信息**”页面上，指定以下信息：

   组织单位
   缺省情况下，将显示 Active Directory 中的用户容器。单击**浏览**以更改缺省组织单位。

   名字
   输入用户的名字。

   全名首字母缩写
   输入用户的全名首字母缩写。

   姓氏
   输入用户的姓氏。

   姓名
   缺省情况下，将在此字段中显示用户的名字、全名首字母缩写和姓氏。您可以修改姓名。

   用户登录名（用户主体名称）
   输入用户在登录邮箱时必须使用的名称。

   用户登录名（Windows 2000 或更早版本之前）
   输入兼容 Microsoft Windows 2000 Server 或更早版本的用户名。

   密码
   输入用户在登录邮箱时必须使用的密码。

   确认密码
   重新输入您在**密码**字段中所输入的密码。

   用户在下次登录时必须更改密码
   如果希望用户重置密码，请选中此复选框。
7. 在“**邮箱设置**”页面上，指定以下信息：

别名
缺省情况下，该字段的值等于在**用户登录名（用户主体名称）**字段中指定的值。

邮箱数据库
单击**浏览**以打开“**选择邮箱数据库**”窗口。选择想要使用的邮箱数据库，然后单击**确定**。

受管文件夹邮箱策略
选中此复选框以指定消息传递记录管理 (MRM) 策略。单击**浏览**以选择想要与此邮箱相关联的 MRM 邮箱策略。

Exchange ActiveSync 邮箱策略
选中此复选框以指定 Exchange ActiveSync 邮箱策略。单击**浏览**以选择想要与此邮箱相关联的 Exchange ActiveSync 邮箱策略。

8. 在“**新建邮箱**”页面上，复审配置摘要。单击**新建**以创建邮箱。在“**完成**”页面上，“摘要”部分将显示是否已创建该邮箱。
9. 单击**完成**。

下一步做什么

将管理员权限分配给创建的 Exchange 用户。

在 Exchange Server 2013 上创建用户

您必须为 Exchange Server 2013 上的代理程序创建用户，从而使该代理程序可以与您想要监视的 Exchange Server 通信并向其认证。

过程

要在 Exchange Server 2013 上创建用户，请完成以下步骤：

1. 使用管理员凭证登录到 Exchange 管理中心。
2. 在“**Exchange 管理中心**”页面上，单击**收件人**，然后单击**邮箱**。
3. 单击**邮箱**选项下面的加号 (+) 旁边的向下箭头，然后单击**用户邮箱**。
4. 在“新建用户邮箱”页面上，单击**新建用户**，然后指定其他字段的值。
5. 单击**保存**。

下一步做什么

将管理员权限分配给创建的 Exchange 用户。

通过运行“新建用户”实用程序创建用户

您可以运行“新建用户”实用程序以在 Exchange Server 2007 或更高版本上创建用户。通过运行此实用程序创建的用户具有运行代理程序的所有必需许可权。在安装代理程序时将安装此实用程序。

开始之前

确保已安装代理程序。要运行“新建用户”实用程序，您必须是具有 Exchange Server 上的完全管理员权限的域管理员。

关于此任务

在运行此实用程序时，用户将在 Active Directory 的“用户”组中创建，并且具有以下许可权：

- 在 Exchange Server 2007 上：
  - 本地管理员
  - 远程桌面用户
  - Exchange 收件人管理员
- 在 Exchange Server 2010 或更高版本上：
  - 本地管理员
  - 远程桌面用户

– Exchange Server 或公共文件夹管理。

过程

要运行“新建用户”实用程序，请完成下列步骤：

1. 双击在以下位置提供的 kexnewuser.exe 文件：
   install_dir\TMAITM6_x64，其中，*install_dir* 是代理程序的安装路径。
2. 在“**新建用户**”窗口中，完成以下步骤：
   a) 输入用户的**名字**和**姓氏**。
      限制: 名字和姓氏的长度不得超过 28 个字符。
   b) 在**用户登录名**字段中，输入用户在每次登录时必须输入的名称。
      限制: 用户登录名的长度不得超过 256 个字符。
   c) 在**密码**字段中，输入您的密码。
   d) 在**确认密码**字段中，重新输入密码。
   e) 如果希望用户在下次登录时重置指定的密码，请选择**用户在下次登录时必须更改密码**。
   f) 单击**下一步**。
   将验证指定的配置值，并且将针对不正确的值显示错误消息。
3. 从邮箱数据库列表，选择必需的邮箱数据库，然后单击**下一步**。
   这将显示配置值摘要。
4. 单击**完成**。

结果

这将保存设置，并创建用户。

向 Exchange Server 用户分配管理员权限

您为 Microsoft Exchange Server 代理程序创建的用户必须是在 Microsoft Exchange Server 上具有完全管理员权限的域管理员。需要管理员权限才能访问 Microsoft Exchange Server 代理程序组件。

开始之前

创建在受监视 Exchange Server 上有邮箱的 Exchange Server 用户。

关于此任务

使用以下某个过程来向该用户分配管理员权限：

- 第 431 页的『在 Exchange Server 2007 上分配管理员权限』
- 第 432 页的『分配 Exchange Server 2010 上的管理员权限』
- 第 432 页的『在 Exchange Server 2013 上分配管理员权限』
- 第 432 页的『在 Exchange Server 2016 上分配管理员权限』

在 Exchange Server 2007 上分配管理员权限

必须向 Exchange Server 2007 上的用户分配 Exchange 收件人管理员权限。

过程

1. 单击**开始 > 程序 > Microsoft Exchange Server 2007 > Exchange 管理控制台**。“**Exchange 管理控制台**”窗口随即打开。
2. 在控制台树中，单击**组织配置**。
3. 在“操作”窗格中，单击**添加 Exchange 管理员**。
4. 在“**添加 Exchange 管理员**”页面上，单击**浏览**。选择创建的新用户，然后选择 **Exchange 收件人管理员**角色。
5. 单击**添加**。

6. 在“**完成**”页面上，单击**完成**。

分配 Exchange Server 2010 上的管理员权限
在 Exchange Server 2010 上，必须将 Exchange Servers 或“公共文件夹管理”权限分配给用户。

过程

1. 以管理员权限登录 Exchange 服务器。
2. 单击**开始 > 管理工具 > 服务器管理器**。
3. 展开**工具**。
4. 单击 **Active Directory 用户和计算机**。
5. 展开**域**，单击 **Microsoft Exchange 安全组**。
6. 右键单击 **Exchange Server 或公共文件夹管理**，然后单击**属性**。
7. 在“**Exchange Server 属性或公共文件夹管理属性**”窗口中，转至**成员**，然后单击**添加**。
8. 从用户列表中，选择想要添加到该组的用户，然后单击**确定**。
9. 单击**确定**。

在 Exchange Server 2013 上分配管理员权限
在 Exchange Server 2013 上，必须将 Exchange Servers 或“公共文件夹管理”权限分配给用户。

过程

1. 以管理员权限登录 Exchange 服务器。
2. 单击**开始 > 管理工具 > 服务器管理器**。
3. 展开**工具**。
4. 单击 **Active Directory 用户和计算机**。
5. 展开**域**，单击 **Microsoft Exchange 安全组**。
6. 右键单击 **Exchange Server 或公共文件夹管理**，然后单击**属性**。
7. 在“**Exchange Server 属性或公共文件夹管理属性**”窗口中，转至**成员**，然后单击**添加**。
8. 从用户列表中，选择想要添加到该组的用户，然后单击**确定**。
9. 单击**确定**。

在 Exchange Server 2016 上分配管理员权限
在 Exchange Server 2016 上，必须将 Exchange Servers 或“公共文件夹管理”权限分配给用户。

过程

1. 以管理员权限登录 Exchange 服务器。
2. 单击**开始 > 管理工具 > 服务器管理器**。
3. 展开**工具**。
4. 单击 **Active Directory 用户和计算机**。
5. 展开**域**，单击 **Microsoft Exchange 安全组**。
6. 右键单击 **Exchange Server 或公共文件夹管理**，然后单击**属性**。
7. 在“**Exchange Server 属性或公共文件夹管理属性**”窗口中，转至**成员**，然后单击**添加**。
8. 从用户列表中，选择想要添加到该组的用户，然后单击**确定**。
9. 单击**确定**。

下一步做什么
使该用户成为安装 Exchange Server 的计算机的本地管理员。

使 Exchange Server 用户成为本地管理员

要访问 Exchange Server 数据，针对 Microsoft Exchange Server 代理程序创建的用户必须是安装 Exchange Server 的计算机的本地管理员。

开始之前

创建 Exchange Server 用户。

关于此任务

使用以下某个过程来使该用户成为本地管理员：

- 第 433 页的『使用户成为 Windows 2003 计算机上的本地管理员』
- 第 433 页的『使用户成为 Windows 2008 计算机上的本地管理员』
- 第 433 页的『使用户成为 Windows 2012 计算机上的本地管理员』
- 第 434 页的『使用户成为 Windows 2016 计算机上的本地管理员』

使用户成为 Windows 2003 计算机上的本地管理员

您必须使针对 Exchange Server 创建的用户成为在 Windows 2003 操作系统上运行且安装了 Exchange Server 的计算机的本地管理员。

过程

1. 在计算机桌面上右键单击**我的电脑**，然后单击**管理**。
2. 展开**本地用户和组**。
3. 单击**组**。
4. 双击 **Administrators** 以显示“**Administrators 属性**”窗口。
5. 单击**添加**。
6. 从**查找位置**列表中选择**整个目录**。
7. 选择创建的用户的名称，然后单击**添加**。
8. 单击**确定**。
9. 单击**确定**。

使用户成为 Windows 2008 计算机上的本地管理员

您必须使针对 Exchange Server 创建的用户成为在 Windows Server 2008 操作系统上运行且安装了 Exchange Server 的计算机的本地管理员。

过程

1. 单击**开始 > 管理工具 > 服务器管理器**。
2. 在导航窗格中，展开**配置**。
3. 双击**本地用户和组**。
4. 单击**组**。
5. 右键单击要将用户帐户添加到的组，然后单击**添加到组**。
6. 单击**添加**并输入用户帐户的名称。
7. 单击**检查名称**，然后单击**确定**。

使用户成为 Windows 2012 计算机上的本地管理员

您必须使针对 Exchange Server 创建的用户成为在 Windows Server 2012 操作系统上运行且安装了 Exchange Server 的计算机的本地管理员。

过程

1. 单击**开始 > 服务器管理器**。
2. 在“**服务器管理器仪表板**”页面上，单击**工具 > 计算机管理**。

3. 在“**计算机管理**”页面的导航窗格中，展开**本地用户和组**，然后单击**用户**。
4. 从用户列表中，右键单击要向其分配管理员权限的用户，然后单击**属性**。
5. 单击**隶属于**选项卡，然后单击**添加**。
6. 在“**选择组**”页面上，输入 `Administrators`，然后单击**确定**。
7. 单击**应用**，然后单击**确定**。

使用户成为 Windows 2016 计算机上的本地管理员
您必须使针对 Exchange Server 创建的用户成为在 Windows Server 2016 操作系统上运行且安装了 Exchange Server 的计算机的本地管理员。

过程

1. 单击**开始 > 服务器管理器**。
2. 在“**服务器管理器仪表板**”页面上，单击**工具 > 计算机管理**。
3. 在“**计算机管理**”页面的导航窗格中，展开**本地用户和组**，然后单击**用户**。
4. 从用户列表中，右键单击要向其分配管理员权限的用户，然后单击**属性**。
5. 单击**隶属于**选项卡，然后单击**添加**。
6. 在“**选择组**”页面上，输入 `Administrators`，然后单击**确定**。
7. 单击**应用**，然后单击**确定**。

配置 Exchange Server 以获取可达性

为了验证可达性，Microsoft Exchange Server 代理程序会向服务器发送电子邮件消息，并度量接收到自动响应的时间量。在启动代理程序之前，必须将 Exchange Server 配置为自动响应电子邮件消息。

开始之前
在配置 Exchange Server 之前，确保完成以下任务：

- 针对想要监视的 Exchange Server 上的用户创建邮箱。
- 针对代理程序创建的用户是域用户。
- 针对 Microsoft Exchange 组织中的服务器配置了服务器之间的邮件流。

过程

对于要验证其可达性的每个 Exchange Server，完成以下步骤：

1. 通过指定创建的用户的凭证，登录到 Microsoft Outlook。
2. 单击“**启动**”窗口上的**下一步**。
3. 选择**是**并单击**下一步**。
4. 在 **Microsoft Exchange Server** 字段中，输入 Exchange Server 的名称。
5. 在**邮箱**字段中，输入创建的用户的名称。
6. 单击**完成**。
7. 单击**确定**。
8. 单击**工具 > 规则和警报 > 新建规则**。
9. 选择**从空白规则开始**。
10. 选择**邮件到达时检查**，然后单击**下一步**。
11. 选择以下选项：
    - **我的姓名在“收件人”框中**
    - **主题或正文中包含特定词语**
12. 在窗口中的**第 2 步**下，单击**特定词语**。
13. 在**指定在主题或正文中要查找的字词或短语**字段中，输入 `AVAILABILITY CHECK`。
14. 单击**添加**。

15. 单击**确定**，然后单击**下一步**。
16. 选择**让服务器使用特定邮件回复**，然后单击**特定邮件**。
17. 在电子邮件消息编辑器内，在消息的“主题”字段中输入以下文本：
    `CHECK RECEIVED: MAILBOX AVAILABLE.`
18. 关闭电子邮件消息编辑器，然后单击**是**保存这些更改。
19. 单击**下一步**。
20. 向您询问例外时，请不要指定任何限制。
21. 单击**下一步**。
22. 单击**完成**，然后单击**确定**。

下一步做什么
配置 Microsoft Exchange Server 代理程序。

配置代理程序以在域用户下运行

缺省情况下，Microsoft Exchange Server 代理程序配置为在本地用户下运行。代理程序必须在创建的域用户下运行。

开始之前
确保：

· 创建的用户是具有本地管理员权限的域用户。
· 该用户具有安装代理程序的服务器上的管理员权限。

关于此任务
当代理程序在域用户下运行时，代理程序可以监视 Exchange Server 的所有组件。

过程

要更改运行代理程序的用户，请完成以下步骤：

1. 运行以下命令以验证用于启动代理程序的用户标识。
   `install_dir\InstallITM\KinCinfo.exe -r`
2. 如果监视代理程序是使用不属于 Administrators 组的用户标识启动的，请停止该代理程序。
3. 打开“**管理监视服务**”窗口。
4. 右键单击代理程序实例，然后单击**更改启动**。
5. 将标准用户标识指定为 `<Domain\Userid>`，然后指定密码。
6. 启动监视代理程序。

本地配置代理程序

您可以使用 IBM Cloud Application Performance Management 窗口来本地配置代理程序。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Microsoft Exchange Server**，然后单击**配置代理程序**。

   注意: 如果**配置代理程序**处于禁用状态，单击**重新配置**。
3. 在“**Monitoring Agent for Microsoft Exchange Server：代理程序高级配置**”窗口中，单击**确定**。
4. 在“**代理程序配置**”窗口中，完成以下步骤：
   a) 单击 **Exchange Server 属性**选项卡，并指定配置参数的值。在单击**确定**时，将验证指定的值。
   b) 单击 **Exchange 服务监视**选项卡，并指定配置参数的值。在单击**确定**时，将验证指定的值。

c) 单击**高级配置属性**选项卡，并指定配置参数的值。在单击**确定**时，将验证指定的值。

有关“**代理程序配置**”窗口的每个选项卡中配置参数的信息，请参阅以下主题：

- 第 436 页的『Exchange Server 属性的配置参数』
- 第 437 页的『Exchange 服务的配置参数』
- 第 437 页的『可达性的配置参数』

有关验证配置值的信息，请参阅第 438 页的『验证配置值』。

5. 重新启动代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

限制: 在 Cloud APM 仪表板上，“我的组件”下仅显示一个 Exchange 组件类型（Microsoft Exchange Server 或 Microsoft Exchange Server 2013）的实例。

Exchange Server 属性的配置参数

在“**代理程序配置**”窗口的 **Exchange Server 属性**选项卡中，您可以配置 Exchange Server 属性，例如，服务器名称、域名和用户名。

下表包含 **Exchange Server 属性**选项卡中配置设置的详细描述。

表 *173. Exchange Server* 属性选项卡中配置设置的名称和描述

参数名称	描述	必填字段	示例
Exchange Server 名称	Exchange Server 的名称。 在安装 Exchange Server 期间，缺省 Exchange Server 名称是 Windows Server 主机名。如果更改缺省 Exchange Server 名称，那么在配置 Exchange Server 代理程序时必须使用更改的名称。 **切记:** 在集群和分布式环境中，为 Exchange Server 2007 指定“邮箱服务器”名称。	是 **要点:** 如果代理程序安装在包含带有两个以上节点的单副本集群的服务器上，请勿指定值。	如果 Exchange Server 名称为 `popcorn`，那么请在 **Exchange Server 名称**字段中输入 `popcorn`。
Exchange 域名	安装 Exchange Server 的域的名称。	是	如果 Exchange Server 位于 `LAB.XYZ.com` 域中，请输入位于第一个点前面的名称，例如，`LAB`。
Exchange 用户名	配置为访问 Exchange Server 的用户的名称。 **切记:** 该用户必须在相同 Exchange Server 上具有邮箱。	是	
Exchange 用户密码	配置为访问 Exchange Server 的用户的密码。	是	
确认密码	针对 Exchange Server 用户指定的相同密码。	是	
Exchange MAPI 概要文件名称	MAPI 概要文件是访问 Exchange Server 所需的主配置设置。如果使用 64 位 Microsoft Exchange Server 代理程序 来监视 Exchange Server 2007 或更高版本，将禁用此字段。	否	
集群中的配置	如果希望在集群环境中配置 Microsoft Exchange Server 代理程序，请选择此复选框。	不适用	

表 173. *Exchange Server* 属性选项卡中配置设置的名称和描述 *(续)*			
参数名称	**描述**	**必填字段**	**示例**
Cluster Server 名称	Cluster Server 的名称。 在选中**集群中的配置**复选框时，将启用此字段。	是，如果启用了此字段。	SCCCluster
Exchange 子系统标识	Cluster Server 节点的名称。 在选中**集群中的配置**复选框时，将启用此字段。	是，如果启用了此字段。	node1
Exchange 代理程序历史数据目录	存储历史数据的磁盘位置。 在选中**集群中的配置**复选框时，将启用此字段。	是，如果启用了此字段。	c:\history

Exchange 服务的配置参数

在“**代理程序配置**”窗口的 **Exchange 服务监视**选项卡中，您可以选择 Exchange 服务以了解 Exchange Server 状态。

下表包含 **Exchange 服务监视**选项卡中配置设置的详细描述。

表 174. “*Exchange* 服务监视”选项卡中配置设置的名称和描述		
参数名称	**描述**	**必填字段**
Exchange 服务	从可用服务列表中选择 Exchange 服务，然后单击箭头以将选定的服务移动到**为服务器状态配置的服务**列表，这样 Microsoft Exchange Server 代理程序就可以对这些服务进行监视。 **切记:** 可用服务列表根据安装的 Exchange Server 版本和角色而更改。	不适用
为服务器状态配置的服务	此列表中可用的服务将确定 Exchange Server 的状态。这些服务是必需的并且不能从**为服务器状态配置的服务**列表移动到 **Exchange 服务**列表。您可以通过移动 **Exchange 服务**列表中的服务来向**为服务器状态配置的服务**列表添加更多的服务。也可以将添加的这些服务移回 **Exchange 服务**列表。	不适用

可达性的配置参数

在“**代理程序配置**”窗口的**高级配置属性**选项卡中，您可以配置与可达性相关的参数，例如，目标电子邮件地址和可达性时间间隔。

下表包含**高级配置属性**选项卡中配置设置的详细描述。

表 175. “高级配置属性”选项卡中配置设置的名称和描述		
参数名称	**描述**	**必填字段**
启用邮箱可达性监视	如果想要代理程序捕获可达性度量数据，请选中此复选框。	不适用
目标电子邮件地址	用于验证可达性的电子邮件地址。请使用分号 (;) 分隔多个电子邮件地址。 **限制:** 此字段中的字符总数不得超过 1023。	是，如果启用了此字段。
电子邮件传输时间间隔（秒）	Exchange Server 代理程序在相邻两次电子邮件发送操作之间的等待时间（秒）。	是，如果启用了此字段。
电子邮件传输超时（秒）	针对为测试邮箱服务器是否可达而发送的电子邮件，代理程序等待其响应的时间间隔（秒）。	否
启用邮箱详细信息监视	选择此复选框以收集邮箱详细信息度量的数据。	不适用

表 175. “高级配置属性”选项卡中配置设置的名称和描述 (续)		
参数名称	**描述**	**必填字段**
邮箱详细信息收集开始时间	收集邮箱详细信息度量的时间（格式为 hh:mm:ss）。	否
邮箱详细信息收集时间间隔（秒）	相邻两次邮箱详细信息度量收集之间的时间间隔（秒）。	否
事件日志收集时间（分钟）	代理程序收集事件记录的持续时间（分钟）。	否
最大事件数	收集事件记录的最大计数。在收集的事件记录数超过最大计数时，事件记录收集将停止。	否
收集时间间隔（秒）	代理程序周期之间的时间间隔（秒）。	否
Exchange 拓扑时间间隔（秒）	相邻两次拓扑详细信息收集之间的时间间隔（秒）。	否
消息跟踪收集时间间隔（小时）	收集消息跟踪日志的时间间隔（小时）。 **限制:** 时间间隔值必须在 1 - 12 范围内。如果指定大于 12 的时间间隔值，那么值将保存为 12。如果输入包含字母或特殊字符的无效值，那么值将保存为 0，这指示禁用了消息跟踪收集。 如果满足以下任何条件，将禁用此字段： · 未在 Exchange Server 上安装邮箱服务器角色或中心传输角色。 · 在 Exchange Server 上禁用了消息跟踪功能。	否

验证配置值
将验证配置代理程序时指定的值。此验证将确保为所有必需参数指定了值并且满足特定条件，例如，用户的本地管理员权限。

下表显示对指定的配置值执行的验证测试。

表 176. 验证测试	
验证测试	**验证是否**
Exchange Server 名称	用户的邮箱服务器名称匹配指定的 Exchange Server 名称。
Exchange Server 权限	用户具有必需的 Exchange Server 权限。在 Exchange Server 2007 上，用户必须具有收件人管理员权限，并且在 Exchange Server 2010 或更高版本上，用户必须具有收件人管理权限。
本地管理员	用户具有本地管理员权限。
代理程序服务登录	代理程序服务配置为以指定的用户帐户运行。

如果一个或多个验证测试失败，那么将生成错误消息。您必须为所有必需参数指定值。否则，无法保存配置的值。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数，其中某些参数定义有缺省值。您可以编辑静默响应文件以使用不同的配置参数值来配置代理程序。

关于此任务
在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

1. 打开位于 *install_dir*\samples 中的 msex_silent_config.txt 文件，并为所有必需参数指定值。

   您还可以修改其他参数的缺省值。
2. 运行以下命令：

   install_dir\BIN\msexch-agent.bat config install_dir\samples \msex_silent_config.txt

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

限制: 在 Cloud APM 仪表板上，“我的组件”下仅显示一个 Exchange 组件类型（Microsoft Exchange Server 或 Microsoft Exchange Server 2013）的实例。

配置代理程序的本地环境变量

您可以为 Microsoft Exchange Server 代理程序 配置本地环境变量，来为重复事件启用或禁用事件调速。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，从**操作**菜单中单击**高级 > 编辑 ENV 文件**。
3. 在 KEXENV 文件中，更改以下环境变量的值：

   EX_EVENT_THROTTLE_ENABLE
   此变量支持您调速重复事件。缺省值为 False。要启用事件调速以阻止触发重复事件情境，请将此变量的值设置为 True。

   EX_EVENT_THROTTLE_DURATION
   此变量提供事件调速的持续时间（分钟）。缺省值为 0 分钟。

配置 Microsoft Hyper-V 监视

在安装 Monitoring Agent for Microsoft Hyper-V Server 时，将使用缺省配置设置来自动配置和启动代理程序。使用静默响应文件可修改缺省配置设置。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft Hyper-V Server 代理程序 的软件产品兼容性报告 (SPCR)。
- 创建包含要修改的配置参数的响应文件。
- 要在“虚拟机”页面上查看虚拟机数据，请确保在每个虚拟机上安装集成组件和操作系统代理程序。对于在 Linux 系统上运行的虚拟机，请确保完成以下任务：
  - 升级 Linux 系统。
  - 在虚拟机上安装更新的 hypervkvpd 或 hyperv-daemons rpm 程序包。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

要点: 对于 Performance Management 8.1.3 发行版，因为不需要而移除了代理程序配置窗口。对于 Performance Management 8.1.2 或更早版本，代理程序配置窗口可用。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

过程

要配置代理程序，请完成以下步骤：

1. 打开位于 *install_dir*\samples 的 microsoft_hyper-v_server_silent_config.txt 文件，并未所有必需参数指定值。

   您还可以修改其他参数的缺省值。
2. 打开命令提示符，并输入以下命令：

   install_dir\BIN\microsoft_hyper-v_server-agent.bat config install_dir \samples\microsoft_hyper-v_server_silent_config.txt

   响应文件包含以下参数：

   · KHV_DIRECTOR_PORT

   · KHV_DIRECTOR_SERVER

   切记: 代理程序配置将组织到以下组中：

   IBM Systems Director 配置 (IBM_DIRECTOR_CONFIGURATION)
   该组中定义的配置元素在代理程序的配置中始终存在。该组定义适用于整个代理程序的信息。

   IBM Systems Director Server 端口号 (KHV_DIRECTOR_PORT)
   IBM Systems Director Server 的端口号。缺省值为 none。

   IBM Systems Director Server 主机名 (KHV_DIRECTOR_SERVER)
   管理环境的 IBM Systems Director Server 的主机名或 IP 地址。缺省值为 none。
3. 如果代理程序处于停止状态，将它启动。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

提供本地安全策略以便在 Windows 上以非管理员用户身份运行 Monitoring Agent for Microsoft Hyper-V Server

可以通过本地安全策略在 Windows 上以非管理员用户身份运行 Monitoring Agent for Microsoft Hyper-V Server。

关于此任务

通过结合使用以下两个本地安全策略，可以在 Windows 上以非管理员用户身份运行 Microsoft Hyper-V Server 代理程序。要让 Microsoft Hyper-V Server 代理程序启动或停止、配置以及验证数据，请使用以下两个策略。

· 调试程序

· 作为服务登录

另外，在 APM 门户网站上，下列属性组需要管理员权限才能获取数据：

· 可用性

· 迁移

· VM 迁移 WO 集群

· VM 存储迁移

请按提供的过程进行操作，将本地安全许可权用于非管理员用户。

过程

1. 以本地管理员身份安装 Microsoft Hyper-V Server 代理程序。
2. 在 `install_dir` 目录下添加该非管理员用户，并为其提供下列许可权：
   a) 提供对 `HKEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring` 注册表的完全访问权。
   b) 在 `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib` 注册表中，提供读访问权给该非管理员用户。
   c) 在 `install_dir` 目录中，提供完全访问权给该非管理员用户。
3. 转至**开始**菜单，并运行 **secpol.msc** 命令以打开本地安全策略。
4. 要在策略中添加非管理员用户，请参阅第 441 页的『授予本地安全策略许可权』。
5. 要在 Hyper-V Administrator Users 组中添加非管理员用户，请参阅第 442 页的『在 Hyper-V Administrator Users 组中添加非管理员用户』。
6. 要在 Performance Monitor Users 组中添加非管理员用户，请参阅第 442 页的『在 Performance Business Monitor Users 组中添加非管理员用户』。
7. 要修改非管理员用户的 DCOM 安全许可权，请参阅第 442 页的『修改 DCOM 许可权』。
8. 重新启动 Microsoft Hyper-V Server 代理程序，并在 APM 门户网站上验证数据。

授予本地安全策略许可权

要启动、停止或配置 Microsoft Hyper-V Server 代理程序以及验证数据，必须授予对以下两项本地安全策略的许可权：“调试程序”和“作为服务登录”。

授予“调试程序”许可权

关于此任务
要授予“调试程序”许可权，请完成以下过程。

过程

1. 单击**开始 > 控制面板 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开**本地策略**并单击**用户权限分配**。此时将打开策略列表。
3. 双击**调试程序**策略。“**调试程序属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**应用**，然后单击**确定**。

授予“作为服务登录”许可权

关于此任务
要授予“作为服务登录”许可权，请完成以下过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开**本地策略**并单击**用户权限分配**。此时将打开策略列表。
3. 双击**作为服务登录**策略。“**作为服务登录属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**应用**，然后单击**确定**。

修改 DCOM 许可权

您必须修改 DCOM 许可权，才能以非管理员用户访问权来运行 Microsoft Hyper-V Server 代理程序。

关于此任务

要修改 DCOM 许可权，请确认用户具有启动 DCOM 服务器所需的相应许可权。要修改许可权，请完成以下过程。

过程

1. 使用 **Regedit** 命令，转至 HKCR\Clsid*clsid value 注册表值。

   注: 当您以非管理员用户身份配置代理程序时，CLSID 值会以事件标识 10016 显示在事件查看器中。
2. 在“注册表编辑器”窗格中，双击**默认**。
3. 在“**编辑字符串**”对话框中，复制值数据字符串。
4. 单击**开始 > 控制面板 > 管理工具 > 组件服务**。
5. 在“**组件服务**”窗口中，展开**组件服务 > 计算机 > 我的电脑**，并双击 **DCOM**。
6. 在“DCOM 配置”窗格中，找到所复制的字符串（程序名），右键单击程序名，然后单击**属性**。
7. 在“**属性**”窗口中，选择**安全**选项卡。
8. 在“**启动和激活权限**”组框下，选择**自定义**，然后单击**编辑**。这将打开“**启动和激活权限**”窗口。
9. 单击**添加**，输入非管理员用户到权限列表中，然后单击**确定**。
10. 针对“本地启动”和“本地激活”选中**允许**复选框，然后单击**确定**。

在 Hyper-V Administrator Users 组中添加非管理员用户

必须在 Hyper-V Administrator Users 组中添加非管理员用户，才能在 APM 门户网站上获取数据。

关于此任务

要在 Hyper-V Administrator Users 组中添加非管理员用户，请完成以下过程。

过程

1. 单击**开始 > 控制面板 > 管理工具 > 计算机管理**。“**计算机管理**”窗口随即打开。
2. 单击**系统工具 > 本地用户和组 > 组**。此时将打开组列表。
3. 双击 **Hyper-V Administrators** 组。“**Hyper-V Administrators 属性**”窗口随即打开。
4. 单击**添加**。“**选择用户或组**”窗口随即打开。
5. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**应用**，然后单击**确定**。

在 Performance Business Monitor Users 组中添加非管理员用户

必须在 Performance Monitor Users 组中添加非管理员用户，才能在 APM 门户网站上获取数据。

关于此任务

要在 Performance Business Monitor Users 组中添加非管理员用户，请完成以下过程。

过程

1. 单击**开始 > 控制面板 > 管理工具 > 计算机管理**。“**计算机管理**”窗口随即打开。
2. 单击**系统工具 > 本地用户和组 > 组**。此时将打开组列表。
3. 双击 **Performance Monitor Users** 组。“**Performance Business Monitor Users 属性**”窗口将打开。
4. 单击**添加**。“**选择用户或组**”窗口随即打开。
5. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**应用**，然后单击**确定**。

配置 Microsoft IIS 监视

在安装 Monitoring Agent for Microsoft Internet Information Services 时，将使用缺省配置设置来自动配置并启动代理程序。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft IIS 代理程序 的软件产品兼容性报告(SPCR)。
- 确保连接到 Microsoft Internet Information Server 环境或应用程序的用户具有管理员特权。使用具有管理员特权的现有用户或者创建新用户。通过向“管理员”组添加新用户，将管理员特权分配给新用户。

  切记: 要配置 Microsoft IIS 代理程序，可使用具有管理员特权的本地用户或域用户。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

要配置代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

下一步做什么

配置代理程序后，可以将用户帐户从本地用户更改为域用户。有关更改用户帐户的步骤，请参阅第 444 页的『更改用户帐户』。

在 Windows 系统上配置代理程序

您可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置 Microsoft IIS 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

Microsoft IIS 代理程序会提供一些参数的缺省值。您可以为这些参数指定其他值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Microsoft Internet Information Services**，并单击**重新配置**。
3. 在 Monitoring Agent for Microsoft Internet Information Services 窗口中，完成下列步骤:
   a) 在 **HTTP 错误日志配置**选项卡上，指定用于保存日志文件的位置，然后单击**下一步**。

      注: 缺省情况下，此日志文件保存在以下位置：`C:\WINDOWS\system32\LogFiles\HTTPERR`。管理员可以更改日志文件的位置。

   b) 在**站点日志配置**选项卡上，指定用于保存日志文件的位置，然后单击**确定**。

      注: 缺省情况下，此日志文件保存在以下位置：`C:\inetpub\logs\LogFiles`。管理员可以更改日志文件的位置。
4. 在“**重新启动 Monitoring Agent for Microsoft IIS**”窗口中，单击**是**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

在安装 Microsoft IIS 代理程序 时，将使用缺省配置设置来自动配置并启动代理程序。使用静默响应文件可修改缺省配置设置。

开始之前

创建包含要修改的配置参数的响应文件。如果要修改缺省配置参数，请编辑响应文件。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

过程

要配置 Microsoft IIS 代理程序，请完成下列步骤：

1. 在命令行上，将路径更改为包含 `msiis-agent.bat` 文件的目录。
2. 输入以下命令：**`msiis-agent.bat config`** 响应文件的绝对路径。

   响应文件包含以下参数：

   KQ7_SITE_LOG_FILE
   C:\inetpub\logs\LogFiles

   KQ7_HTTP_ERROR_LOG_FILE
   C:\WINDOWS\system32\LogFiles\HTTPERR

   切记: 代理程序配置将组织到以下组中：

   站点日志配置 (SITE_LOG)
   该组包含与站点日志文件 (KQ7_SITE_LOG_FILE) 相关的配置参数。管理员可以指定用于保存日志文件的位置。缺省情况下，此日志文件保存在以下位置：`C:\inetpub\logs\LogFiles`

   HTTP 错误日志配置 (HTTP_ERROR_LOG)
   该组包含与 HTTP 错误日志文件 (KQ7_HTTP_ERROR_LOG_FILE) 相关的配置参数。管理员可以指定用于保存日志文件的位置。缺省情况下，此日志文件保存在以下位置：`C:\WINDOWS\system32\LogFiles\HTTPERR`
3. 如果代理程序处于“已停止”状态，请启动代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

更改用户帐户

配置 Microsoft IIS 代理程序 后，您可以将用户帐户从本地用户更改为域用户。

关于此任务

缺省情况下，Microsoft IIS 代理程序使用本地用户帐户来运行。

过程

1. 运行以下命令以验证用于启动代理程序的用户标识：

   `install_dir\InstallITM\KinCinfo.exe -r`

2. 如果监视代理程序是使用不属于 Administrators 组的用户标识启动的，请停止该代理程序。
3. 打开“**管理监视服务**”窗口。
4. 右键单击代理程序实例，然后单击**更改启动**。
5. 将标准用户标识指定为 <Domain\User ID>，然后指定密码。
6. 启动 Microsoft IIS 代理程序。

配置 Skype for Business Server（旧称 Microsoft Lync Server）监视

安装 Monitoring Agent for Skype for Business Server（旧称 MS Lync Server）后，该代理程序处于“已取消配置”状态。要启动该代理程序，必须对其进行配置。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 Skype for Business Server 代理程序 的软件产品兼容性报告 (SPCR)。
- 确保用于运行 Skype for Business Server 代理程序的用户是具有管理员特权的域用户，并且具有 Lync Server 或 Skype for Business Server 拓扑中列出的所有远程服务器的访问权。使用具有管理员特权的现有域用户，或者创建新的域用户并为新的域用户分配管理员特权。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

要配置代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。

下一步做什么

在配置代理程序后，可以将用户帐户从本地用户更改为域用户。有关更改用户帐户的步骤，请参阅第 447 页的『更改用户帐户』。

非管理员用户的许可权和许可权

您可以非管理员用户身份来运行 Skype for Business Server 代理程序 监视代理程序；但是，某些功能不可访问。

注册表许可权

要创建非管理员用户，请创建一个新用户（非管理员），并为这个新用户设置注册表许可权，如下所示。

- 对 KEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring 的完全访问权
- 对 CANDLE_HOME 目录的完全访问权

这个非管理员用户必须是 Performance Monitor Users 和 Performance Log Users 的成员。如果您为非管理员用户定义这些许可权，那么将显示基于 Perfmon 的所有属性组的数据。

查看从数据库收集的属性组数据

如果要查看从数据库收集的属性组数据，您必须为非管理员用户设置下列许可权。

- 用来运行 Skype for Business Server 代理程序 的非管理员用户帐户必须具有“调试程序”许可权，以便将调试器添加到任何进程。

  缺省情况下，“调试程序”许可权仅分配给管理员和“本地系统”帐户。要授予“调试程序”许可权，必须在 Lync Server 或 Skype for Business Server 上完成下列步骤：

  1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。

2. 展开**本地策略**，然后单击**用户权限分配**。此时将打开用户权限列表。
3. 双击**调试程序**策略。“**调试程序属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在“输入对象名称来选择”字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

· 授予“作为服务登录”许可权

要授予“作为服务登录”许可权，必须在 Lync Server 或 Skype for Business Server 上完成下列步骤：

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开**本地策略**，然后单击**用户权限分配**。此时将打开用户权限列表。
3. 双击**作为服务登录**策略。“**作为服务登录属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在“输入对象名称来选择”字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

对于 Administrators 组中的各个成员用户，“可用性”属性组将会显示数据。

在 Windows 系统上配置代理程序

您可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置 Skype for Business Server 代理程序（旧称 MS Lync Server 代理程序）。在更新配置值后，必须启动代理程序以保存更新的值。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

Skype for Business Server 代理程序会提供一些参数的缺省值。您可以针对这些参数指定不同的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Skype for Business Server**，然后单击**配置代理程序**。
3. 在 Monitoring Agent for Skype for Business Server 窗口中，完成下列步骤：
   a) 在 **Skype for Business 拓扑的 SQL 配置**选项卡中，要连接到 Microsoft Lync Server 或 Skype for Business Server Central Management Store，请指定配置参数的值，然后单击**下一步**。

      注: 您可以跳过此选项卡，因为 Skype for Business 拓扑的 SQL 配置不适用于 IBM Cloud Application Performance Management。

      要点: 合成事务配置为可选。如果需要合成事务数据，请指定**设置信息**和**调度程序配置**选项卡上的配置参数。
   b) 在**管理员登录凭证**选项卡上，指定管理员凭证，然后单击**下一步**。
   c) 在**设置信息**选项卡上，要运行合成事务的命令，请指定配置参数的值，然后单击**下一步**。
   d) 在**调度程序配置**选项卡上，要调度合成事务，请指定配置参数的值，然后单击**下一步**。
   e) 在 **Skype for Business 监视角色的 SQL Server 配置**选项卡中，要连接到 Microsoft Lync Server 或 Skype for Business Server 监视角色，请指定配置参数的值，然后单击**下一步**。

   有关配置参数的信息，请参阅第 447 页的『代理程序的配置参数』。
4. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Skype for Business Server**，然后单击**启动**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

安装 Skype for Business Server 代理程序（旧称 MS Lync Server 代理程序）后，该代理程序处于“已取消配置”状态。要启动该代理程序，必须对其进行配置。
静默响应文件包含代理程序配置参数，其中某些参数定义有缺省值。您可以编辑静默响应文件来为配置参数指定其他值。

开始之前

创建包含要修改的配置参数的响应文件。如果要修改缺省配置参数，请编辑响应文件。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

过程

要配置 Skype for Business Server 代理程序，请完成下列步骤：

1. 打开命令提示符。
2. 将路径切换到 `skype_for_business_server-agent.bat` 文件所在的目录。
3. 输入 **skype_for_business_server-agent.bat config** *absolute path to the response file* 命令。

   有关配置参数的信息，请参阅第 447 页的『代理程序的配置参数』。
4. 如果代理程序处于停止状态，将它启动。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

更改用户帐户

配置 Skype for Business Server 代理程序 后，您可以将用户帐户从本地用户更改为域用户。

关于此任务

缺省情况下，Skype for Business Server 代理程序使用本地用户帐户来运行。当代理程序在域用户下运行时，代理程序可从远程服务器收集数据。

过程

1. 运行以下命令以验证用于启动代理程序的用户标识：

   install_dir\InstallITM\KinCinfo.exe -r
2. 如果监视代理程序是使用不属于 Administrators 组的用户标识启动的，请停止该代理程序。
3. 打开“**管理监视服务**”窗口。
4. 右键单击代理程序实例，然后单击**更改启动**。
5. 将标准用户标识指定为 <Domain\User ID>，然后指定密码。
6. 启动 Skype for Business Server 代理程序。

代理程序的配置参数

在配置 Skype for Business Server 代理程序（旧称 MS Lync Server 代理程序）时，您可以更改配置参数的缺省值，例如，数据库服务器名称、数据库实例名称、数据库名称和其他参数。

下表包含 Skype for Business Server 代理程序 的配置参数的描述。

注: 所有字段中，“池 FQDN”字段在下表中为必填。

表 *177*. 代理程序的配置参数的名称和描述

参数名称	**描述**
数据库服务器名称（例如，PS6877）	· **Skype for Business 拓扑的 SQL 配置**选项卡：Lync Server 或 Skype for Business Server Central Management Store 安装所在数据库服务器的名称。 · **Skype for Business 监视角色的 SQL Server 配置**选项卡：监视角色安装所在数据库服务器的名称。
数据库实例名称	· **Skype for Business 拓扑的 SQL 配置**选项卡：缺省实例。 · **Skype for Business 监视角色的 SQL Server 配置**选项卡：监视角色安装所在数据库实例的名称。
数据库名称	数据库的名称。
数据库用户标识	数据库的用户标识。此用户必须具有必需的 Microsoft SQL Server 实例的访问权。此用户不能是 Active Directory 用户。
数据库密码	监视角色安装位置的数据库密码。
用户名（例如：skype\administrator）	管理员的用户标识。此用户必须是具有管理员特权的域用户，并且具有 Lync Server 或 Skype for Business Server 拓扑中列出的所有远程服务器的访问权。此用户的凭证也用于“综合事务”功能部件中。因此，应授权此用户在任务计划程序中创建 Windows 计划并执行综合事务命令。
密码	管理员的登录密码。
确认域密码	输入在“域密码”字段中指定的同一密码。
池 FQDN	对其执行了综合命令的 Skype 池的标准域名 (FQDN)。
地理位置	生产系统的地理位置。
测试用户 1（例如，user1@skype.com）	执行综合事务 cmdlet 时可以使用的第一个用户名。用户名的格式为 SAMAccountName@domain.com。 请勿提供 Sip 地址。
测试用户 1 密码	测试用户 1 的密码。
确认测试用户 1 密码	输入在**测试用户 1 密码**字段中指定的同一密码。
测试用户 2（例如，user2@skype.com）	执行综合事务 cmdlet 时可以使用的第二个用户名。用户名的格式为 SAMAccountName@domain.com。 请勿提供 Sip 地址。
测试用户 2 密码	测试用户 2 的密码。
确认测试用户 2 密码	输入在**测试用户 2 密码**字段中指定的同一密码。
用户代理程序配置值	如果要使用配置面板中提供的所有字段运行综合命令，请保持启用此字段。 禁用 New-CsHealthMonitoringConfiguration 设置的值。如果已禁用，那么**池 FQDN** 的值将用作 Get-CsHealthMonitoringConfiguration 的身份。确保提供有效的测试用户凭证以执行 **Test-CsMcxP2PIM** 命令。

表 177. 代理程序的配置参数的名称和描述 (续)	
参数名称	**描述**
频率	访存合成事务数据的调度实用程序的频率。频率可以具有以下值: · 每日 (DAY_FREQUENCY) · 每周 (WEEK_FREQUENCY) · 每月 (MONTHLY_FREQUENCY)
收集小时	选中以调度实用程序的时间戳记的小时部分，采用 24 小时时钟格式。
收集分钟	选择以调度实用程序的时间戳记的分钟部分。
开始日期 (YYYY-MM-DD)	激活调度程序的时间。
结束日期 (YYYY-MM-DD)	取消激活调度程序的时间。

配置 Microsoft .NET 监视

Monitoring Agent for Microsoft .NET 用于监视 .NET 应用程序。完成安装后，此代理程序将自动启动以收集资源监视数据。但是，要收集事务跟踪和诊断数据，必须完成一些配置任务。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft .NET 代理程序 的 软件产品兼容性报告 (SPCR)。

关于此任务

安装此代理程序之后，请完成下列配置任务，以使此代理程序可以收集事务跟踪和诊断数据:

1. 注册数据收集器
   数据收集器是 Microsoft .NET 代理程序的组件。它收集事务跟踪和诊断数据，并将该数据传递给 Microsoft .NET 代理程序。您必须注册数据收集器才能收集此数据。有关详细信息，请参阅第 450 页的『注册数据收集器』。
2. 配置事务跟踪和诊断数据的收集
   注册数据收集器之后，请在 Cloud APM 控制台上启用事务跟踪和诊断数据收集。您也可以使用 **configdc** 命令来启用诊断数据收集。有关详细信息，请参阅第 453 页的『启用事务跟踪和诊断数据收集』和第 453 页的『使用 configdc 命令来启用诊断数据的收集』。
3. 激活配置更新
   如果使用 **configdc** 命令来启用诊断数据收集，那么必须激活配置，以便将更新保存到配置文件中。有关激活配置更改的更多信息，请参阅第 455 页的『激活配置更新』。
4. 调整数据收集器的性能
   您可能需要完成一些任务，以微调数据收集器的性能。有关详细信息，请参阅第 455 页的『数据收集器的性能调整』。

代理程序共存

在代理程序共存环境中，您可以从 Cloud APM 控制台或 Tivoli® Enterprise Portal 查看事务跟踪数据。有关在代理程序共存环境中启用事务跟踪数据收集的信息，请参阅第 454 页的『在代理程序共存环境中启用事务跟踪』。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅代理程序版本。要访问先前代理程序发行版的文档，请参阅下表:

表 *178.* 代理程序版本和文档	
Microsoft .NET 代理程序版本	**文档**
8.1.3.2	IBM Cloud Application Performance Management
8.1.3 和 8.1.2	IBM Performance Management 8.1.3

该链接会打开本地 Knowledge Center 主题。

使用本地帐户或域帐户运行代理程序所需的许可权

仅属于管理员组成员的本地或域用户有权运行 Microsoft .NET 代理程序。如果本地或域用户不是管理员组成员，此主题提供了必须满足的条件。

用户必须对系统驱动器和代理程序安装驱动器具有下列许可权

1. 读
2. 写
3. 执行
4. 修改

用户必须对 HKEY_LOCAL_MACHINE 注册表键具有下列许可权

· 读

用户在受监视服务器上必须是下列组的成员

1. Users
2. IIS_IUSRS
3. Performance Monitor Users
4. Performance Log Users

注: 但是，建议使用隶属于本地 Administrators 组的本地用户或域用户来运行 Microsoft .NET 代理程序。

注册数据收集器

必须注册数据收集器才能收集事务跟踪和诊断数据。要收集资源监视数据，无需特定配置。

关于此任务

根据您是要让数据收集器收集事务数据、诊断数据还是这两类数据，请注册数据收集器的下列组件：

表 *179.* 数据收集器组件及其功能	
组件名称	**监视**
httpmodule	ASP.NET 事务，并收集请求响应时间和 CPU 时间
profiler	ADO.NET 事务，并收集用于诊断的方法、堆栈跟踪及请求上下文数据
isapi	ASP.NET 事务，并收集请求响应时间和 CPU 时间
soap	ASMX 或 WCF 服务事务，及 WCF 服务响应时间

切记:

· 使用 `isapi32` 在 64 位 Microsoft IIS Server 上过滤 32 位应用程序。

· 注册所有组件以跟踪所有事务，并查看完整的事务拓扑。

过程

1. 在安装此代理程序的服务器上，以管理员身份运行以下命令：

```
cd install_dir\qe\bin
configdc.exe registerdc [all|isapi|isapi32|profiler|httpmodule|soap]
```

切记：

- 如果运行 **configdc.exe registerdc** 命令时没有指定任何要注册的组件，那么将仅注册 httpmodule。
- 要注册所有组件，请运行 **configdc.exe registerdc all** 命令。
- 要同时注册任何组件，请运行以下命令：**configdc.exe registerdc** *component_name component_name*。例如，**configdc.exe registerdc** httpmodule profiler

2. 重新启动 .NET 应用程序。

下一步做什么

注册数据收集器之后，必须启用事务跟踪及诊断数据收集。有关启用数据收集的信息，请参阅第 453 页的『启用事务跟踪和诊断数据收集』。

如果您要停止监视 .NET 应用程序，请注销数据收集器。重复上述步骤，使用 **configdc.exe unregisterdc** 命令来注销数据收集器的所有组件。

使用 .NET 代理程序的“IIS 响应时间”模块

从 8.1.4.0.2 发行版开始，.NET 代理程序包含“IIS 响应时间”模块，该模块与 Response Time 代理程序配合工作，以显示 IIS 服务器的最终用户事务数据。

启用“响应时间”模块

“响应时间”模块必须先启用才能使用。

过程

要启用“响应时间”模块，请完成下列步骤：

1. 以管理员方式打开命令提示符。
2. 要停止 IIS，请运行以下命令：

```
iisreset /stop
```

3. 在命令提示符中，转至 install_dir\qe\bin 目录。
4. 要为 IIS 注册“响应时间”模块，请运行以下命令：

```
configdc registerdc rtmodule
```

5. 要启动 IIS，请运行以下命令：

```
iisreset /start
```

结果

“响应时间”模块现已启用。

配置 Response Time 代理程序以使其与 .NET 代理程序的“IIS 响应时间”模块配合工作

您必须配置 Response Time 代理程序，以使其与 .NET 代理程序的“IIS 响应时间”模块配合工作。

开始之前

安装 Response Time 代理程序 (V8.1.4)，有关更多信息，请参阅第 101 页的『第 6 章 安装代理程序』。

过程

完成下列步骤，从而配置 Response Time 代理程序以使其与 .NET 代理程序的“IIS 响应时间”模块配合工作：

1. 以管理员方式打开文本编辑器。
2. 在文本编辑器中打开以下文件：

   config_dir\TMAITM6_x64*host_name*_T5.config

   其中，*config_dir* 是 APM 主目录，*host_name* 是服务器名称。
3. 更新以下属性：

   { KT5DISABLEANALYZER=YES } { KT5ENABLEWEBPLUGIN=YES }
4. 在 SECTION=analyzerconfig [ ] 一节中添加以下属性：

   {KT5WEBPLUGINIPCNAME=KFC1}
5. 重新启动 Response Time 代理程序。
6. 登录 Performance Management 控制台，以在仪表板中验证代理程序收集的数据。有关使用 Performance Management Console 的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

为“IIS 响应时间”模块配置 JavaScript 注入

您必须配置 JavaScript (JS) 注入以与 .NET 代理程序的“Internet Information Services (IIS) 响应时间”模块配合工作。

过程

要配置 JS 注入以与 .NET 代理程序的“IIS 响应时间”模块配合工作，请遵循以下步骤：

1. 以管理员方式打开文本编辑器。
2. 在文本编辑器中打开以下文件：

   <APM_HOME>\qe\config\dotNetDcConfig.properties.inactive
3. 要对“响应时间”模块启用 JS 注入，请将 **RTModule.JSInjection.Enabled** 属性设置为 **true**。
4. 要对“响应时间”模块禁用 JS 注入，请将 **RTModule.JSInjection.Enabled** 属性设置为 **false**。
5. 在管理员方式下打开命令提示符并转至 <APM_HOME>\qe\bin 目录。
6. 运行以下命令：

   - configdc activateconfig
   - iisreset

禁用“IIS 响应时间”模块

您不想查看 IIS 服务器的最终用户事务数据时，可以将“IIS 响应时间”模块禁用。

过程

完成下列步骤，以禁用“IIS 响应时间”模块：

1. 以管理员方式打开命令提示符。
2. 要停止 IIS，请运行以下命令：

   iisreset /stop
3. 在命令提示符中，转至 *install_dir*\qe\bin 目录。
4. 要为 IIS 注销“响应时间”模块，请完成下列步骤：
   - 要为 IIS 注销“响应时间”模块，请运行以下命令：

     configdc unregisterdc rtmodule
   - 要注销数据收集器的所有组件，包括“响应时间”模块，请运行以下命令：

     configdc unregisterdc all
5. 要启动 IIS，请运行以下命令：

   iisreset /start

结果
“IIS 响应时间”模块现已禁用。

“IIS 响应时间”模块的局限性
“IIS 响应时间”模块的局限性列示如下。

· “IIS 响应时间”模块不会跟踪用户信息，用户名当前显示为“未知”。

启用事务跟踪和诊断数据收集

在“**代理程序配置**”页面上，您可以启用或禁用事务跟踪和诊断数据的收集。

开始之前
请确保您已注册数据收集器。有关详细信息，请参阅第 450 页的『注册数据收集器』。

过程

完成以下步骤为每个受管系统配置数据收集。

1. 登录到 Cloud APM 控制台。
2. 从导航栏，单击 **系统配置 > 代理程序配置**。
   “**代理程序配置**”页面随即显示。
3. 单击 **MS .NET** 选项卡。
4. 选中要对其配置数据收集的受管系统的相应复选框，然后完成**操作**列表中的下列任意一项操作。
   · 要启用事务跟踪，请单击**设置事务跟踪 > 已启用**。对于每个选中的受管系统，**事务跟踪**列中的状态将更新为“已启用”。
   · 要启用诊断数据收集，请选择**设置诊断方式**并单击想要设置的级别。**诊断方式**列中的状态将更新以显示每个所选受管系统的指定级别。
     – 级别 1：HTTP 模块收集请求摘要和请求实例数据。
     – 级别 2：HTTP 模块收集请求摘要和请求实例数据。概要分析程序收集方法数据和堆栈跟踪数据。
   · 要禁用事务跟踪，请单击**设置事务跟踪 > 已禁用**。对于每个选中的受管系统，**事务跟踪**列中的状态将更新为“已禁用”。
   · 要禁用诊断数据收集，请单击**设置诊断方式 > 已禁用**。对于每个选中的受管系统，**诊断方式**列中的状态将更新为“已禁用”。

结果
已配置每个受管系统的数据收集。

下一步做什么
登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的事务跟踪和诊断数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用 configdc 命令来启用诊断数据的收集

您也可以使用 **configdc** 命令来启用或禁用诊断数据的收集。此过程是可选的。

开始之前

· 请确保您已注册数据收集器。有关详细信息，请参阅第 450 页的『注册数据收集器』。
· 确保已完成第 453 页的『启用事务跟踪和诊断数据收集』上的过程。
· 确保 `qe_custom.properties` 文件在 `<APM_Home>\localconfig\qe` 上由 APM 服务器处理并具有以下属性：
  – transaction_tracking=ENABLED
  – diagnostic_mode=LEVEL2

过程

1. 运行以下命令：

```
cd install_dir\qe\bin configdc deepdivedc -tracelevel trace_level
```

其中，

install_dir
Microsoft .NET 代理程序的安装目录。

trace_level
跟踪级别，用于指示 .NET Data Collector 所收集的诊断数据量。请指定下列其中一个值：

0
禁用诊断数据的收集。

1
启用诊断数据的收集。HTTP 模块收集请求摘要和请求实例数据。

2
启用诊断数据的收集。HTTP 模块收集请求摘要和请求实例数据。概要分析程序收集方法数据和堆栈跟踪数据。

提示: 使用 **configdc.exe deepdivedc -tracelevel** 命令设置跟踪级别时，会在 dotNetDcConfig.properties 文件中设置 bci_dc.diagnose.level 参数的值。

2. 激活配置更改。
有关激活更改的信息，请参阅第 455 页的『激活配置更新』。

下一步做什么
登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的诊断数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在代理程序共存环境中启用事务跟踪

在代理程序共存环境中，您可以配置数据收集器以收集事务跟踪数据并传递到 Tivoli Enterprise Portal（这是 IBM Tivoli Monitoring 的一个组件）。

开始之前
您必须安装 Microsoft .NET 代理程序（作为 Cloud APM 的一部分交付），并且必须移除或注销 .NET Data Collector 组件（随 ITCAM for Microsoft Applications 一起交付）。使用 **configdc.exe unregisterdc** 命令取消注册数据收集器的所有模块。

过程

要配置数据收集器以收集事务跟踪数据并传递到 Tivoli Enterprise Portal，请完成下列步骤：

1. 转至 *install_dir*\localconfig\qe 目录，其中，*install_dir* 是 Microsoft .NET 代理程序 的安装目录。缺省路径为 C:\IBM\APM。
2. 打开 qe_default.properties 文件，并将 **transaction_tracking** 参数值设置为 ENABLED。
3. 保存并关闭 qe_default.properties 文件。
4. 转至 *install_dir*\qe\config 目录。
5. 在文本编辑器中打开 dotNetDcConfig.properties.inactive 文件。
6. 按如下所示设置 **TTDC.enabled** 和 **TTAS.enabled** 参数：

```
TTDC.enabled=true
TTAS.enabled=true
```

7. 要配置与事务收集器的连接，请将 **TTAS.Host** 和 **TTAS.Port** 参数的值设置为事务收集器的 IP 地址和端口号。
8. 运行以下命令以激活更改：

```
install_dir\qe\bin\configdc.exe activateconfig
```

9. 重新启动 .NET 应用程序，以使更改生效。

结果

现在，可以收集事务跟踪数据并在 Tivoli Enterprise Portal 中显示。

下一步做什么

要为 .NET Data Collector 禁用事务跟踪，请重复该过程并使用以下配置值：

- 在 qe_default.properties 文件中，设置 transaction_tracking=DISABLED。
- 在 dotNetDcConfig.properties.inactive 文件中，设置 TTDC.enabled=false 和 TTAS.enabled=false。

激活配置更新

必须激活您使用 configdc 命令在配置设置中进行的更新。此激活可确保将更新保存到 dotNetConfig.properties 文件。

关于此任务

使用 **configdc** 命令来更新配置设置时，将更新 dotNetConfig.properties 文件中的参数值。但是，如果此文件在使用中，导致无法修改，那么配置设置更新将保存在 dotNetDcConfig.properties.inactive 文件中。您必须激活配置，使这些更新保存在 dotNetConfig.properties 文件中。

过程

1. 转至以下路径：
   install_dir\qe\bin
   其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。
2. 运行以下命令：
   configdc activateConfig

下一步做什么

如果 Internet Information Service (IIS) 事务受到监视，并且数据收集器配置已更新，请重新启动 IIS 以激活配置。

如果 ASMX 或 WCF Web Service 受到监视，并且数据收集器配置已更新，请重新启动托管 Web Service 的进程。

数据收集器的性能调整

当您配置数据收集器以收集事务跟踪和诊断数据时，数据收集器的性能会受影响。为了提高性能，您可完成一些性能调整任务。

您可能需要完成下列任务，以提高数据收集器的性能：

- 过滤所要监视的 ADO.NET 接口。
- 对事务跟踪和诊断数据进行采样。
- 配置跟踪日志记录。

指定要监视的 ADO.NET 接口

您可以指定要启用事务跟踪的 ADO.NET 客户机接口。

开始之前

如果要查看 .NET Data Collector 支持的 ADO.NET 接口，请参阅数据收集器支持的名称空间功能.

要查看 .NET Data Collector 的配置值，请参阅 *install_dir*\qe\config 目录中的 dotNetDcConfig.properties 文件，其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。

关于此任务

缺省情况下，安装代理程序期间，所有受支持的 ADO.NET 接口都已启用事务跟踪。要指定数据收集器必须监视的接口，请为特定接口启用或禁用监视。

如果您禁用对某个接口的监视，那么任何关联应用程序域过滤器的设置将保留在数据收集器配置文件中。再次启用该接口时，将保留该过滤器。

过程

- 要启用对 ADO.NET 接口的监视，请完成下列步骤：
  a) 从 *install_dir*\qe\bin 目录，运行以下命令：

```
configdc enableMonitor all | adsi | db2 | ldap | odbc | oledb | oracle | sql
| http | web
[-appdomain appdomain filter list]
```

  b) 激活配置更改。
  有关激活更改的信息，请参阅第 455 页的『激活配置更新』。
- 要禁用对 ADO.NET 接口的监视，请完成下列步骤：
  a) 从 *install_dir*\qe\bin 目录，运行以下命令：

```
configdc disableMonitor all | adsi | db2 | ldap | odbc | oledb | oracle | sql
| http | web
```

  b) 激活配置更改。
  有关激活更改的信息，请参阅第 455 页的『激活配置更新』。

对事务跟踪和诊断数据进行采样

如果系统性能因为事务跟踪或诊断数据收集而受影响，那么您可对收集的数据启用采样以提高性能。

关于此任务

当系统性能由于事务跟踪和诊断数据收集而受到影响时，可以配置数据收集器以通过采样定期收集数据。启用采样后，数据收集器不会收集每个请求的数据，而是按多个请求的时间间隔来收集数据。您可以根据 DotNetProfilerService 进程的 CPU 使用率动态地更改采样率。



警告: 但是，采样可以节省系统资源，采样数据可能无法有效地诊断问题。启用数据采样之后，事务跟踪拓扑可能中断或丢失。因此，只有在性能严重受影响时，才应启用数据采样。

过程

要对事务跟踪和诊断数据收集启用采样，请完成下列步骤：

1. 转至以下目录：
   install_dir\qe\config
   其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。
2. 在文本编辑器中，打开 dotNetDcConfig.properties.inactive 文件。
3. 在文件中设置以下参数：

   bci_dc.sampling.Enabled
   指定数据收集器是否定期收集事务跟踪和诊断数据。有效值为 true 和 false。

   bci_dc.sampling.base
   指定数据采样基准。有效值为正数。例如，如果 **bci_dc.sampling.base** 参数值设置为 10，那么数据收集器会每 10 个请求收集一次事务跟踪和诊断数据。采样率为 1/10 个请求。数据收集器将收集第 1 个、第 11 个、第 21 个和第 31 个请求等请求的数据。

bci_dc.dynamic.sampling
指定采样率是常量还是动态变化。有效值为 on 和 off。在将 **bci_dc.dynamic.sampling** 参数值设置为 on 时，采样速率将根据 **bci_dc.dynamic.max_cpu_usage** 参数的值动态调整。

bci_dc.dynamic.max_cpu_usage
指定 DotNetProfilerService 进程的 CPU 使用率阈值。如果 DotNetProfilerService 进程的 CPU 使用率大于指定值的 110%，那么将减小采样率。如果 CPU 使用率小于指定值的 90%，那么将增大采样率。有效值在 1 到 100 范围内。

4. 保存并关闭 dotNetDcConfig.properties.inactive 文件。
5. 运行以下命令以激活更改：

```
install_dir\qe\bin\configdc.exe activateconfig
```

6. 重新启动 .NET 应用程序，以使更改生效。

对数据收集器启用跟踪日志记录

可以允许数据收集器生成跟踪日志。您可使用这些跟踪日志来解决收集事务跟踪和诊断数据时发生的问题。

关于此任务

要收集 ASP.NET 事务、ADO.NET 事务和诊断数据的日志，请对数据收集器的 httpmodule、profiler 和 isapi 组件启用跟踪日志。要收集 ASMX 和 WCF 事务的日志，请对数据收集器的 soap 组件启用跟踪日志。

要点: 启用跟踪日志记录可能会影响数据收集器的性能。因此，收集所需的跟踪日志之后，请禁用跟踪日志记录。

过程

1. 在安装此代理程序的服务器上，浏览到以下路径：
   install_dir\qe\bin
   其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。
2. 根据您想要启用的跟踪日志，完成下列任意一个过程，或完成这两个过程：
   - 要对 httpmodule、profiler、soap 组件和响应时间模块启用跟踪日志，请完成下列步骤：
     a. 运行以下命令：
        configdc logging -tracing on
     b. 重新启动 IIS 和 .NET 应用程序。
   - 要对 BCI 引擎启用跟踪日志，请完成下列步骤：
     a. 导航到下列路径：
        <APM_HOME>\qe\config
     b. 在文本编辑器中，打开 dotNetDcConfig.properties.inactive 文件。
     c. 对于属性 **bci_dc.trace.logging**，请将值指定为 on。
     d. 运行以下命令：
        configdc activateconfig
     e. 重新启动 IIS。

下一步做什么

要禁用跟踪日志，请执行以下操作：

- 要对 httpmodule、profiler、soap 组件和响应时间模块禁用跟踪日志，请完成下列步骤：
  - 运行以下命令：
    configdc logging -tracing off
  - 重新启动 IIS 和 .NET 应用程序。
- 要对 BCI 引擎禁用跟踪日志，请完成以下步骤：
  - 转至以下路径：

<APM_HOME>\qe\config

- 在文本编辑器中，打开 dotNetDcConfig.properties.inactive 文件。
- 对于 **bci_dc.trace.logging** 属性，请将值指定为 off。
- 运行以下命令：
  configdc activateconfig
- 重新启动 IIS 和 .NET 应用程序。

配置 Microsoft Office 365 监视

您必须配置 Microsoft Office 365 代理程序才能监视组织的 Microsoft Office 365 预订的可用性和性能。

开始之前

· 查看必备硬件和软件。

· 要收集 Office 365 用户的数据，必须在安装此代理程序的 Windows 客户机上安装下列模块：

- PowerShell 3.0 或更高版本
- Microsoft Online Services Sign-In Assistant PowerShell
- SharePoint Online Management Shell
- DotNetFrameworkVersion 4.5.2 或更高版本

配置此代理程序的用户必须具有管理特权，以及启用 PowerShell 的远程执行策略所需的特权。

· 要监视 Skype 合成事务，请完成下列任务：

- 在用户要执行 Skype 的合成事务的 Windows 客户机上，安装 Skype 2013 客户机。
- 将 Lync 和 Skype 的缺省视频设备设置为虚拟音频视频过滤器。

· 确保启动 Microsoft Office 365 的用户具有管理员权限。使用具有管理员特权的现有用户，或者创建新用户。通过向“管理员”组添加新用户，向新用户分配管理员特权。

查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft Office 365 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

产品版本和代理程序版本通常不同。以下指南针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 Microsoft Office 365 代理程序。要访问 V1.0.0 的文档，请参阅 IBM Cloud Application Performance Management Knowledge Center。

安装代理程序之后，可以启动 Microsoft Office 365 代理程序。但是，需要手动配置才能查看所有代理程序属性的数据。

要配置代理程序，您可以使用 IBM Cloud Application Performance Management 窗口或静默响应文件。

验证已配置的用户的可达性

为了验证可达性，Microsoft Office 365 代理程序会向已配置的用户发送电子邮件消息，并测量接收到自动响应的时间量。在启动代理程序之前，您必须配置所有用户（在 Office 365 代理程序邮箱可达性设置中配置），以自动响应电子邮件消息。

开始之前

在配置 Exchange Online 用户以实现可达性之前，确保完成以下任务：

· 对于 Exchange Online 上的每个要监视的用户，创建一个邮箱。

· 您为代理程序创建的用户是全局 Office 365 用户。

过程

对于要验证可达性的每个 Exchange Online 用户帐户，完成以下步骤：

1. 通过指定创建的用户的凭证，登录到 Microsoft Outlook。
2. 单击**工具 > 规则和警报 > 新建规则**。
3. 在**规则向导**窗口中的**从空白规则启动**下面，单击**对我接收的消息应用规则**并单击**下一步**。
4. 选择以下选项：
   - **来自人员或公共组**
   - **主题中包含特定词语**
5. 在窗口中的**第 2 步**下，单击**人员或公共组**。
6. 在**规则地址**窗口中，选择其发送的消息将被收到的用户（全局管理员）并单击**下一步**。
7. 在窗口中的**第 2 步**下，单击**特定词语**。
8. 在**指定在主题或正文中要查找的字词或短语**字段中，输入 `Test Reachability`。
9. 单击**添加**。
10. 单击**确定**，然后单击**下一步**。
11. 选择**让服务器使用特定邮件回复**，然后单击**特定邮件**。
12. 在电子邮件消息编辑器内，在消息的“主题”字段中输入以下文本：
    `Test Reachability`。
13. 在**收件人**列表中，添加全局管理员。
14. 关闭电子邮件消息编辑器，然后单击**是**保存这些更改。
15. 单击**完成**。
16. 单击**应用**，然后单击**确定**。

下一步做什么

配置 Microsoft Office 365 代理程序。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Microsoft Office 365 代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

Microsoft Office 365 代理程序为一些参数提供缺省值。您可以为这些参数指定不同的值。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Microsoft Office 365**，然后单击**配置代理程序**。
3. 在 **Monitoring Agent for Microsoft Office 365** 窗口中，完成下列步骤：
   a) 在 **Office365 预订详细信息**选项卡上，输入 Office 365 全局管理员的用户名和密码，然后单击**下一步**。
   b) 在**合成事务**选项卡上的**可达性电子邮件地址**字段中，输入以分号定界的电子邮件地址列表。
   c) 要启用 Skype QoS 度量值的数据收集，请选中 **Skype QoS** 复选框，然后单击**下一步**。
   d) 在**邮箱和 OneDrive 使用情况监视**选项卡上，从**收集时间间隔**列表选择收集时间间隔的持续时间（小时），然后单击**下一步**。
4. 在 **Monitoring Agent for Microsoft Office 365** 窗口中，单击**是**。

下一步做什么

- 配置 Skype 合成事务实用程序，以监视 Skype QoS 合成事务。有关监视 Skype QoS 的更多信息，请参阅第 461 页的『监视 Skype QoS』。
- 将用户帐户从本地用户更改为域用户。有关详细信息，请参阅第 461 页的『更改用户帐户』。
- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
- 如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

安装 Microsoft Office 365 代理程序时，在提供配置设置之后，必须手动配置并启动代理程序。请使用静默响应文件来配置定制设置。

开始之前

编辑响应文件，以修改下列缺省配置设置：

KMO_USER_NAME
Office 365 全局管理员的用户名。

KMO_PASSWORD
Office 365 全局管理员的密码。

KMO_MAIL_ADDRESSES1
目标电子邮件地址列表，用于验证邮箱可达性。电子邮件地址列表必须使用分号进行定界。

KMO_SKYPE
此参数用于启用 Skype QoS 合成事务收集。

KMO_DATA_COLLECTION_DURATION
在访存邮箱和 OneDrive 用量数据之前，此代理程序等待的持续时间（小时）。

响应文件位于下列位置：
`<CANDLEHOME>\samples`

关于此任务

当代理程序运行或停止时，您都可以配置它。代理程序会在配置完成后保持相同的状态。例如，如果代理程序正在运行，它会在配置完成后保持运行状态。

过程

要配置 Microsoft Office 365 代理程序，请完成下列步骤：

1. 在命令提示符处，将路径切换到 `microsoft_office365-agent.bat` 文件所在的目录。
2. 输入以下命令：`microsoft_office365-agent.bat` *absolute path to the response file*。

   响应文件包含以下参数：
3. 如果代理程序处于“已停止”状态，请启动代理程序。

下一步做什么

- 配置 Skype 合成事务实用程序，以监视 Skype QoS 合成事务。有关监视 Skype QoS 的更多信息，请参阅第 461 页的『监视 Skype QoS』。
- 将用户帐户从本地用户更改为域用户。有关详细信息，请参阅第 461 页的『更改用户帐户』。
- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
- 如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

更改用户帐户

配置 Microsoft Office 365 代理程序后，请将用户帐户从本地用户更改为域用户。

关于此任务

缺省情况下，Microsoft Office 365 代理程序使用本地用户帐户来运行。

过程

1. 运行以下命令以验证用于启动代理程序的用户标识：

   install_dir\InstallITM\KinCinfo.exe -r
2. 如果监视代理程序是使用不属于 Administrators 组的用户标识启动的，请停止该代理程序。
3. 打开“**管理监视服务**”窗口。
4. 在“**管理 Monitoring Services**”窗口中，右键单击代理程序实例，并单击**更改启动**。
5. 将标准用户标识指定为 <Domain\User ID>，然后指定密码。
6. 启动 Microsoft Office 365 代理程序。

监视 Skype QoS

要监视 Skype QoS，在安装了代理程序的 Windows 客户机上，或者在配置了 Skype for Business 的分布式环境中，用户必须配置 Skype 合成事务实用程序 kmoskypecaller.exe 和 Kmoskypereceiver.exe。

开始之前

要执行合成事务，必须以如下格式更新 <CANDLEHOME>\tmaitm6_x64\kmoskypecallerlist.properties 文件中的 Skype 调用者名称和 Skype 接收方名称：
Skype 调用者 = Skype 接收方
例如，john@xyz.com = alan@xyz.com
针对单个 Skype 调用者，可以使用以下格式添加多个 Skype 调用接收方：
Skype 调用者 = Skype 接收方列表
例如，john@xyz.com = alam@xyz.com;bill@xyz.com;chuk@xyz.com

切记: 如果您不想执行合成事务，而是想为实时用户监视 Skype QoS，那么无需更新 <CANDLEHOME>\TMAITM6_x64\kmoskypecallerlist.properties 文件。

关于此任务

配置并启动 Microsoft Office 365 代理程序时，将在 <CANDLEHOME>\TMAITM6_x64\ 中创建下列文件和文件夹：

- kmoskypecaller.properties
- kmoskypecallerlist.properties
- KMOSynthTransSkype.zip
- KMOSkypeTransReceiver.zip

并且，将以代理程序与 kmoskypecaller 实用程序之间用于通信的服务器 IP 和端口来更新 kmoskypecaller.properties 文件。

过程

要配置 Skype 调用者和 Skype 接收方并启动合成事务，例如即时消息传递、音频和视频呼叫以及应用程序共享会话，请完成下列步骤：

1. 启动 Office 365 代理程序。
2. 将 KMOSynthTransSkype.zip 文件从代理程序客户机复制到要从中启动 Skype 调用的 Windows 客户机。

3. 解压缩 KMOSynthTransSkype.zip 文件。
4. 将 kmoskypecaller.properties 文件从代理程序客户机复制到要从中启动 Skype 调用的 Windows 客户机上的解压缩的 KMOSynthTransSkype 文件夹。
5. 将 KMOSkypeTransReceiver.zip 文件从代理程序客户机复制到所有必须接收 Skype 调用的 Windows 客户机。
6. 在所有必须接收 Skype 调用的 Windows 客户机上解压缩 KMOSkypeTransReceiver.zip 文件，并运行 KMOSkypeTransReceiver.exe 以开始接收消息。
7. 要启动合成事务，请运行 KMOSynthTransSkype.exe 文件，此文件位于 Windows 客户机上解压缩的 KMOSynthTransSkype 文件夹中。Office 365 代理程序将从调用者客户机中开始接收 Skype 监视数据。

结果
代理程序开始监视 Skype QoS。

配置本地环境变量

可以配置本地环境变量以更改 Microsoft Office 365 代理程序的行为。

过程

1. 单击**开始> 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，从**操作**菜单中单击**高级 > 编辑 ENV 文件**。
3. 在环境变量文件中，输入环境变量的值。

   有关可配置的环境变量的信息，请参阅第 462 页的『本地环境变量』。

本地环境变量
您可以通过配置本地环境变量来更改 Microsoft Office 365 代理程序的行为。

用于为代理程序定义数据收集方法的变量

要为代理程序数据收集设置方法，请使用以下环境变量：

- **CDP_DP_INITIAL_COLLECTION_DELAY**：使用该变量设置一个时间间隔（秒），在此时间间隔后，线程池开始数据收集。
- **KMO_MAILBOX_REACHABILITY_INTERVAL**：使用该变量设置邮箱可访问性属性组的数据收集时间间隔（分钟）。
- **KMO_SKYPE_REPORT_INTERVAL**：使用该变量设置 Skype for Business 使用情况统计信息功能的数据收集时间间隔（小时）。
- **KMO_SERVICE_API_INTERVAL**：使用该变量设置 Office 365 服务运行状况功能的数据收集时间间隔（分钟）。
- **KMO_NETWORK_CONNECTION_INTERVAL**：使用该变量设置互联网连接功能的数据收集时间间隔（分钟）。
- **KMO_NETWORK_PERFORMANCE_INTERVAL**：使用该变量设置 Office 365 服务网络性能功能的数据收集时间间隔（分钟）。
- **KMO_SITE_CONNECTION_INTERVAL**：使用该变量设置 Office 365 连接功能的数据收集时间间隔（分钟）。
- **KMO_SPSITE_COLLECTION_INTERVAL**：使用该变量设置 SharePoint Sites 详细信息功能的数据收集时间间隔（分钟）。
- **KMO_UASGE_STATS_INTERVAL**：使用该变量设置 Office 365 服务使用情况和用户统计信息功能的数据收集时间间隔（小时）。
- **KMO_TENANT_INTERVAL**：使用该变量设置 Office 365 租户详细信息功能的数据收集时间间隔（分钟）。

- **KMO_ONEDRIVE_CONNECTIVITY_INTERVAL**：使用该变量设置 Office 365 OneDrive 连接功能的数据收集时间间隔（分钟）。
- **KMO_TENANT_DOMAIN**：使用该变量设置租户的域名。

配置 Microsoft SharePoint Server 监视

在安装 Monitoring Agent for Microsoft SharePoint Server 时，将使用缺省配置设置来自动配置和启动代理程序。使用静默响应文件可修改缺省配置设置。

开始之前

请确保完成了以下任务：

- 确保连接到 Microsoft SharePoint Server 环境或应用程序的用户具有管理员特权。使用具有管理员特权的现有用户或创建新的用户。通过向“管理员”组添加新用户，向新用户分配管理员特权。

  切记： 要配置 Microsoft SharePoint Server 代理程序，可使用具有管理员特权的本地用户或域用户。
- 编辑响应文件并修改缺省配置参数。

  响应文件包含以下参数：

  KQP_DB_User
  数据库的用户标识。

  KQP_DB_Password
  数据库的密码。

查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft SharePoint Server 代理程序 的软件产品兼容性报告 (SPCR)。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅代理程序版本。要访问先前代理程序发行版的文档，请参阅下表：

表 *180.* 代理程序版本和文档

Microsoft SharePoint Server 代理程序版本	**文档**
06.31.09.00、06.31.10.00	IBM Cloud Application Performance Management
06.31.09.00	IBM Performance Management 8.1.3 **注：** 该链接会打开本地 Knowledge Center 主题。
06.31.07.00	IBM Performance Management 8.1.2 **注：** 该链接会打开本地 Knowledge Center 主题。

过程

要配置 Microsoft SharePoint Server 代理程序，请完成下列步骤：

1. 打开命令提示符。
2. 将路径切换到 `ms_sharepoint_server-agent.bat` 文件所在的目录。
3. 输入以下命令：**`ms_sharepoint_server-agent.bat config`** *absolute path to the response file*
4. 如果代理程序处于“已停止”状态，请启动代理程序。

下一步做什么

配置代理程序后，可以将用户帐户从本地用户更改为域用户。有关更改用户帐户的步骤，请参阅第 464 页的『更改用户帐户』。

更改用户帐户

配置 Microsoft SharePoint Server 代理程序 后，您可以将用户帐户从本地用户更改为域用户。

关于此任务

如果使用域用户，那么代理程序可以监视 Microsoft SharePoint Server 代理程序的所有组件。

过程

要更改用户帐户，请完成下列步骤：

1. 运行以下命令以验证用于启动代理程序的用户标识。

   install_dir\InstallITM\KinCinfo.exe -r
2. 如果监视代理程序是使用不属于 Administrators 组的用户标识启动的，请停止该代理程序。
3. 打开“**管理监视服务**”窗口。
4. 右键单击代理程序实例，然后单击**更改启动**。
5. 将标准用户标识指定为 <Domain\Userid>，然后指定密码。
6. 启动监视代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

通过非管理员用户来运行 Monitoring Agent for Microsoft SharePoint Server

可以通过本地安全策略以非管理员用户身份运行 Monitoring Agent for Microsoft SharePoint Server。

关于此任务

通过结合使用以下两个本地安全策略，可以非管理员用户身份运行 Microsoft SharePoint Server 代理程序。

1. 调试程序。
2. 作为服务登录。

请按提供的过程进行操作，将本地安全许可权用于非管理员用户。

过程

1. 转到 TEMA，将 Microsoft SharePoint Server 代理程序启动更改为使用非管理员用户身份进行。
2. 在注册表键 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Office Server 目录下添加非管理员用户，并授予其读访问权。
3. 在注册表键 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\Web Server Extensions 下添加非管理员用户，并授予其读访问权。
4. 在注册表键 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\Web Server Extensions\16.0\Secure\ 下手动添加非管理员用户，并授予其读访问权。
5. 在注册表键 HKEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring 目录下添加非管理员用户，并授予其完全许可权。
6. 在注册表键 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion \Perflib 目录下添加非管理员用户，并授予其读访问权。
7. 在 SharePoint 代理程序安装文件夹（Candle 文件夹，例如 C:\IBM\APM）中添加非管理员用户，并授予其完全许可权。
8. 在**开始菜单**中运行 **secpol.msc** 命令，以打开“**本地安全策略**”。
9. 在“本地安全策略”中添加非管理员用户，请参阅第 465 页的『本地安全策略许可权』。
10. 将非管理员用户添加到 SQL Server Login 用户组。这个用户必须具有 SQL Server 上的 sysadmin SQL Server 角色许可权。

11. 重新启动 Microsoft SharePoint Server 代理程序。
12. 检查 Microsoft SharePoint Server 代理程序状态，并验证 APM 门户网站上的数据。
13. 对于 Administrators 组中的各个成员用户，下列属性组将会显示数据。
    a) 可用性
    b) Web Service

本地安全策略许可权

可以通过本地安全策略以非管理员用户身份运行 Microsoft SharePoint Server 代理程序。这些策略可帮助您启动或停止代理程序、配置代理程序以及对其执行数据验证。要以非管理员用户身份来运行 Microsoft SharePoint Server 代理程序，请完成两项本地安全策略工作。

授予“作为服务登录”许可权

您可授予“作为服务登录”许可权。

关于此任务

要授予“作为服务登录”许可权，请在 Microsoft SharePoint Server 代理程序上完成这里描述的过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 在导航窗格中，展开“**本地策略**”，然后单击**用户权限分配**。这样会打开用户权限列表。
3. 双击**作为服务登录**策略。“**作为服务登录属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

授予“调试程序”许可权

您可授予“调试程序”许可权。

关于此任务

要授予“调试程序”许可权，请在 Microsoft SharePoint Server 代理程序上完成这里描述的过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开“**本地策略**”，然后单击**用户权限分配**。这样会打开用户权限列表。
3. 双击**调试程序**策略。“**调试程序属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在“输入对象名称来选择”字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

配置 Microsoft SQL Server 监视

您必须配置 Monitoring Agent for Microsoft SQL Server，以便代理程序可从正受监视的应用程序收集数据。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Microsoft SQL Server 代理程序 的软件产品兼容性报告 (SPCR)。

通过使用命令提示符界面，可以本地安装和配置 Microsoft SQL Server 代理程序。确保在受监视的服务器上安装代理程序。

关于此任务

以下指南针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

Microsoft SQL Server 代理程序 是一个多实例代理程序；您必须手动配置和启动每个代理程序实例。

· 要配置代理程序，请完成以下任务：
  – 创建用户和授予所需许可权
  – 选择要监视的数据库
  – 配置本地环境变量
· 要在集群环境中运行代理程序，请完成“在集群环境中运行代理程序”主题中描述的步骤。

创建用户和授予许可权

在 Microsoft SQL Server 上，您必须创建一个用于运行代理程序的用户，并向该用户授予监视 Microsoft SQL Server 的许可权。授予许可权的过程对于 Microsoft SQL Server 2005 或更高版本是相同的。

开始之前

安装 Microsoft SQL Server 代理程序。要创建用户并将许可权授予该用户，您必须是具有 sysdamin 授权角色的数据库管理员。

关于此任务

使用以下过程来确定现有 SQL Server 用户是否具有足够的权限来监视 Microsoft SQL Server：

· Windows 第 466 页的『检查现有 SQL Server 用户的许可权』

使用下列过程之一来创建用户：

· Windows 第 467 页的『通过 Windows 认证创建 SQL Server 用户标识』
· Linux Windows 第 468 页的『通过 SQL Server 认证创建 SQL Server 用户标识』

请使用以下过程来授予许可权：

· Windows 第 468 页的『授予用于数据收集的最小许可权』
· Windows 第 469 页的『授予对 Perflib 注册表键的许可权以收集一些数据集的数据』

检查现有 SQL Server 用户的许可权

Windows 您可以运行实用程序工具 **koqVerifyPerminssions.exe**，检查现有 SQL Server 用户是否具有足够的与 SQL Server 数据库相关的许可权。

关于此任务

如果用户具有 **sysadmin** 角色或最低要求的许可权，那么实用程序工具 **koqVerifyPerminssions.exe** 返回消息 PASS。详细检查结果记录在 koqVerifyPermissions_log 中。

以下列示的是最低许可权：

· 服务器的许可权必须包括**查看服务器状态**、**查看任何数据库**和**查看任何定义**。

  这些服务器级别许可权是必需的。
· 有关用于监视的所有系统数据库和用户定义的数据库，数据库角色成员资格必须包括 **public** 和 **db_owner**。

  需要 **db_owner** 许可权来收集以下数据集的数据：
  – “服务器详细信息”数据集
  – “数据库详细信息”数据集
  – “数据库镜像”数据集

– “服务器摘要”数据集
– “作业摘要”数据集

· 有关 **msdb** 数据库，数据库角色成员资格必须包括 **db_datareader**、**SQLAgentReaderRole** 和 **SQLAgentUserRole**。“作业详细信息”数据集需要这些许可权。

过程

1. 启动命令提示符并切换到以下实用程序目录。
   · 对于 64 位代理程序，*Agent_home*\TMAITM6_x64
   · 对于 32 位代理程序，*Agent_home* \TMAITM6
   其中，*Agent_home* 是代理程序安装目录。
2. 通过提供以下参数运行 **koqVerifyPerminssions.exe**：

```
koqVerifyPermissions.exe -S Instance_name -U Username -P Password
```

   其中：
   · *Instance_name* 是 SQL Server 实例名称。
   · *Username* 是实用程序工具验证的用户名。
   · *Password* 是用户的密码。如果提供了 *username*，那么此参数为必需。
   注: 如果未提供 *username* 和 *password*，那么使用登录系统的缺省用户。示例：NT AUTHORITY \SYSTEM。

结果
koqVerifyPermissions_log 中的以下目录提供了详细检查结果：
· 对于 64 位代理程序，*Agent_home*\TMAITM6_x64\logs
· 对于 32 位代理程序，*Agent_home* \TMAITM6\logs
其中，*Agent_home* 是代理程序安装目录。

通过 Windows 认证创建 SQL Server 用户标识
Windows 通过 Windows 认证创建新用户，并向该用户分配所需角色和许可权。

过程

要创建用户，请完成下列步骤：
1. 在 **SQL Server Management Studio** 中，打开**对象资源管理器**。
2. 单击 ***Server_instance_name*** > **安全** > **登录**。
3. 右键单击 **Logins**，然后选择**新建登录名**。
4. 在**常规**页面上的**登录名**字段中，输入 Windows 用户的名称。
5. 选择 **Windows 认证**。
6. 根据您要对此用户分配的角色和许可权，完成下列任务之一：
   · 在“服务器角色”页面上，向新的登录标识分配 *sysadmin* 角色。
   · 如果您不想对该用户分配 *sysadmin* 角色，请完成第 468 页的『授予用于数据收集的最小许可权』中提及的步骤来向该用户分配最低许可权。
   要点: 缺省情况下，系统会向新登录标识分配 *public* 角色。
7. 单击**确定**。

结果
创建了具有缺省 *public* 角色和您分配的许可权的用户，它会显示在**登录** 列表中。

通过 SQL Server 认证创建 SQL Server 用户标识

Linux Windows 通过 SQL Server 认证创建新用户，并向该用户分配所需角色和许可权。

过程

要创建用户，请完成下列步骤：

1. 在 **SQL Server Management Studio** 中，打开**对象资源管理器**。
2. 单击 ***Server_instance_name*** > **安全** > **登录**。
3. 右键单击 **Logins**，然后选择**新建登录名**。
4. 在**常规**页面上的**登录名**字段中，输入新用户的名称。
5. 选择 **SQL Server 认证**。
6. 在**密码**字段中，输入用户的密码。
7. 在**确认密码**字段中，重新输入已在**密码**字段中输入的密码。
8. 根据您要对此用户分配的角色和许可权，完成下列任务之一：
   - 在“服务器角色”页面上，向新的登录标识分配 *sysadmin* 角色。
   - 如果您不想对该用户分配 *sysadmin* 角色，请完成第 468 页的『授予用于数据收集的最小许可权』中提及的步骤来向该用户分配最低许可权。

   要点: 缺省情况下，系统会向新登录标识分配 *public* 角色。
9. 单击**确定**。

结果

创建了具有缺省 *public* 角色和您分配的许可权的用户，它会显示在**登录** 列表中。

授予用于数据收集的最小许可权

Windows 除了缺省的 **public** 角色，您还可以向用户分配 **sysadmin** 角色或者向用户授予最低权限，使代理程序能够收集数据集的数据。

关于此任务

您可以通过用户界面或实用程序工具 **permissions.cmd** 授予许可权。

过程

- 要通过用户界面向用户授予最低权限，请完成下列步骤：
  a) 打开**服务器角色**页面，确认已选中 **public** 复选框。
  b) 打开**用户映射**页面，然后为要监视的所有系统数据库和用户定义数据库选择下列复选框：
  - **public**
  - **db_owner**

  对于 **msdb** 数据库，还要选择下列复选框：
  - **db_datareader**
  - **SQLAgentReaderRole**
  - **SQLAgentUserRole**

  c) 打开**安全对象**页面，然后为您监视的服务器实例选择下列复选框：
  - 查看数据库
  - 查看定义
  - 查看服务器状态
- 要使用实用程序工具 **permissions.cmd** 向用户授予最低权限，请完成下列步骤：
  a) 启动 Windows Explorer 并浏览到实用程序工具目录 *Agent_grant_perm_dir*：

- 对于 64 位代理程序，*Agent_grant_perm_dir* 是 *Agent_home*\TMAITM6_x64\scripts\KOQ\GrantPermission。
- 对于 32 位代理程序，*Agent_grant_perm_dir* 是 *Agent_home*\TMAITM6\scripts\KOQ\GrantPermission。
- *Agent_home* 是代理程序安装目录。

注意: 缺省情况下，实用程序工具 **permissions.cmd** 向所有数据库上的 **db_owner** 授权。要排除某些数据库，必须在 *Agent_grant_perm_dir*\exclude_database.txt 文件中添加数据库名称。必须通过符号别名 **@** 分隔数据库名称。

提示: 例如，要排除数据库 **MyDatabase1** 和 **MyDatabase2**，请在 exclude_database.txt 文件中添加以下条目：

```
MyDatabase1@MyDatabase2
```

b) 双击 **permissions.cmd** 启动实用程序工具。

c) 出现提示时输入预期参数值：

表 *181.* 参数

参数	**描述**
SQL Server 名称或 SQL Server 实例名称	输入您要向用户授予许可权的目标 SQL Server 名称或目标 SQL Server 实例名称。
现有的 SQL Server 用户登录名称	输入将变更其许可权的用户名。
许可权选项： **1** 授予 **db_owner** 许可权 **2** 授予 **db_datareader**、**SQLAgentReaderRole** 和 **SQLAgentUserRole** 许可权 **3** 授予所有必需的许可权	根据您的需求输入 **1** 或 **2** 或 **3**。
授予许可权的用户： **1** 当前登录到系统的用户 **2** 其他用户	输入 **1** 或 **2**。 如果选择 **2**，那么出现提示时输入目标用户名。 **注:** 用户必须有权向其他用户授予许可权。

下一步做什么

配置代理程序。

授予对 Perflib 注册表键的许可权以收集一些数据集的数据

Windows 要收集一些数据集的数据，您需要向用户授予对 Perflib 注册表键的读访问权。

关于此任务

您需要向 Windows 用户授予配置代理程序服务所使用的此访问权。在缺少 Perflib 许可权的情况下，有许多数据集受到影响，例如 MS SQL 数据库详细信息、MS SQL 内存管理器、MS SQL 锁定资源类型摘要、MS SQL 作业摘要、MS SQL Server 事务摘要、MS SQL Server 摘要等。

过程

要授予对 Perflib 注册表键的许可权，请完成下列步骤：

1. 要打开注册表编辑器，请单击**开始** > **运行** > **Regedit.exe**，然后按 **Enter** 键。

2. 转至 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib 注册表键。
3. 右键单击 **Perflib** 键，然后单击**许可权**。
4. 单击**添加**，输入安装和配置代理程序所使用的 Windows 用户名，然后单击**确定**。
5. 单击您添加的用户。
6. 通过选中该复选框，向用户授予读访问权。
7. 单击**应用**，然后单击**确定**。

本地环境变量

您可以通过配置本地环境变量来更改 Microsoft SQL Server 代理程序的行为。

用于检查 SQL Server 服务可用性的变量

要检查 SQL Server 服务的可用性，请使用以下环境变量：

- **COLL_MSSQL_RETRY_INTERVAL**：此变量提供检查 SQL Server 服务状态的重试时间间隔（以分钟为单位）。如果此值小于等于 0，那么变量会采用缺省值 1 分钟。
- **COLL_MSSQL_RETRY_CNT**：此变量提供 SQL Server 代理程序检查是否已启动 SQL Server 服务时的重试次数。如果在此变量所指定的重试次数后 SQL Server 服务未启动，那么收集器会停止工作。如果此变量值小于或等于 0，那么该变量会采用缺省值 3。

用于监视 SQL Server 错误日志文件的变量

要监视“MS SQL 错误事件详细信息”数据集，请使用以下环境变量：

- **COLL_ERRORLOG_STARTUP_MAX_TIME**：此变量提供代理程序启动之前的错误收集时间间隔 (T)。缺省值为 0 分钟。此变量可以采用以下值：

  T = 0
  代理程序从代理程序启动或重新启动时开始监视错误日志文件。代理程序不会读取在代理程序启动之前错误日志文件中记录的错误。

  T = 1
  代理程序根据为 **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** 变量设置的以下值监视错误日志文件，此变量由 R 表示：
  - 如果 R < 0，那么代理程序从代理程序启动或重新启动时开始监视错误日志文件。
  - 如果 R = 1，那么代理程序监视错误日志文件中记录的所有错误。
  - 如果 R > 1，且代理程序是首次安装的，那么代理程序会监视错误日志文件，直至监视到 R 个错误。如果 R > 1，且重新启动了代理程序，那么代理程序会监视先前遗漏的所有 R 个错误。

  T > 1
  代理程序从代理程序启动或重新启动时监视 N 分钟内记录的所有先前错误。代理程序监视还取决于针对 **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** 变量设置的以下值：
  - 如果 R ⩽ 0，那么代理程序从代理程序启动或重新启动时开始监视错误日志文件。
  - 如果 R = 1，那么代理程序在错误日志文件中监视 T 分钟内记录的所有错误。
  - 如果 R > 1，那么代理程序监视的错误数不超出过去 T 分钟内记录的 R 个错误。

- **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW**：此变量提供在代理程序启动时必须处理的最大错误数。缺省值为 0。您可以为此变量指定以下值：

  R = 0
  代理程序从代理程序启动或重新启动时开始监视错误日志文件。代理程序不会读取在代理程序启动之前错误日志文件中创建的错误。

  R = 1
  代理程序从代理程序启动或重新启动时监视过去 T 分钟内记录的错误。

R > 1
代理程序监视过去 T 分钟内记录的 R 个错误。

- **COLL_ERRORLOG_MAX_EVENT_ROW**：此变量提供错误行数。缺省值是 50。您可以为此变量指定以下值：

X = 0
代理程序不显示错误日志。

X > 0
代理程序显示 X 个错误行。

- **COLL_ERRORLOG_RECYCLE_WAIT**：此变量将提供当“MS SQL 错误事件详细信息”属性组的情境触发时，Microsoft SQL Server 代理程序在收集此属性组的数据之前所等待的时间间隔（以秒为单位）。您可将 1 到 30 范围内的值指定给此变量。如果此变量值小于 0，那么该变量会采用缺省值 0（秒）。如果此变量值大于 30，那么该变量会采用缺省值 30（秒）。

用于设置查询超时时间间隔的变量

要为 SQL Server 代理程序设置查询超时时间间隔，必须使用以下环境变量。

- **QUERY_TIMEOUT**：此环境变量定义 SQL Server 代理程序等待针对发送到 SQL Server 的查询接收响应的最大时间量（秒）。此变量的值必须小于 45 秒。然而，如果将此变量的值设置为 0 秒，那么 SQL Server 代理程序将无限期等待从 SQL Server 接收响应。如果 SQL Server 代理程序访问很多锁定的数据库，那么必须为此变量指定 10 - 20 秒范围内的值。如果未在设置的超时时间间隔内处理查询，那么 SQL Server 代理程序将跳过超时的查询并移至队列中的下一个查询。代理程序不会为已超时的查询显示数据。
- **QUERY_THREAD_TIMEOUT**：此环境变量定义 SQL Server 代理程序针对发送到 SQL Server 的查询，等待接收响应的最大时间量（秒）。此环境变量适用于那些使用线程化收集功能的属性组。例如，KOQDBD、KOQTBLD 和 KOQDEVD 等等。与 QUERY_TIMEOUT 变量不同，此变量的值没有任何限制。否则，它的工作方式与 QUERY_TIMEOUT 变量类似。

用于查看有关已启用作业的信息的变量

要查看有关“MS SQL 作业详细信息”数据集中已启用作业的信息，请使用 **COLL_JOB_DISABLED** 环境变量。如果将此变量的值设置为 1，那么 Microsoft SQL Server 代理程序 不会显示有关已禁用作业的信息。如果未指定此变量，那么可查看有关已启用和已禁用作业的信息。

用于限制“MS SQL 文件组详细信息”数据集中的行数的变量

要限制收集器服务对“MS SQL 文件组详细信息”数据集访存的行数，请使用 **COLL_KOQFGRPD_MAX_ROW** 环境变量。此环境变量用于定义收集器服务对“文件组详细信息”数据集访存的最大行数。如果未指定此变量的值，那么收集器服务会对“文件组详细信息”数据集访存 10,000 行。使用此环境变量可以修改 koqcoll.ctl 文件中缺省的最大行数限制。完成下列步骤以修改缺省限制：

1. 在 koqcoll.ctl 文件中为 KOQFGRPD 指定最大行数。
2. 添加 **COLL_KOQFGRPD_MAX_ROW** 环境变量，并确保此变量的值与您在 koqcoll.ctl 文件中指定的值相同。

如果 koqcoll.ctl 文件中的值小于 **COLL_KOQFGRPD_MAX_ROW** 环境变量中指定的值，那么 koqcoll.ctl 文件中的值将被视为最大行数的值。

如果 koqcoll.ctl 文件中的值大于 **COLL_KOQFGRPD_MAX_ROW** 环境变量中指定的值，那么 **COLL_KOQFGRPD_MAX_ROW** 环境变量中的值将被视为最大行数的值。

用于增强“MS SQL 文件组详细信息”数据集收集的变量

使用 **COLL_DBD_FRENAME_RETRY_CNT** 变量可以指定将 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_TEMP 文件移至 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_PREV 文件的尝试次数。

如果您未指定此变量的值，那么 Microsoft SQL Server 代理程序 会尝试移动该文件 3 次。

用于限制“MS SQL 设备详细信息”数据集中的行数的变量

要限制收集器服务对“MS SQL 设备详细信息”数据集访存的行数，请使用 **COLL_KOQDEVD_MAX_ROW** 环境变量。此环境变量用于定义收集器服务对“设备详细信息”数据集访存的最大行数。如果未指定此变量的值，那么收集器服务会对“设备详细信息”数据集访存 10,000 行。使用此环境变量可以修改 koqcoll.ctl 文件中缺省的最大行数限制。完成下列步骤以修改缺省限制：

1. 在 koqcoll.ctl 文件中为 KOQDEVD 指定最大行数。
2. 添加 **COLL_KOQDEVD_MAX_ROW** 环境变量，并确保此变量的值与您在 koqcoll.ctl 文件中指定的值相同。

如果 koqcoll.ctl 文件中的值小于 **COLL_KOQDEVD_MAX_ROW** 环境变量中指定的值，那么 koqcoll.ctl 文件中的值将被视为最大行数的值。

如果 koqcoll.ctl 文件中的值大于 **COLL_KOQDEVD_MAX_ROW** 环境变量中指定的值，那么 **COLL_KOQDEVD_MAX_ROW** 环境变量中的值将被视为最大行数的值。

用于增强“MS SQL 设备详细信息”数据集收集的变量

要增强“MS SQL 设备详细信息”数据集收集，请使用以下环境变量：

- **COLL_KOQDEVD_INTERVAL**：通过使用此环境变量，您可以指定“MS SQL 设备详细信息”数据集的两次连续收集之间的时间间隔（以分钟为单位）。

  注: 缺省情况下，“设备详细信息”数据集的数据收集基于需求。使用 **COLL_KOQDEVD_INTERVAL** 变量为“设备详细信息”数据集启动基于线程的收集并设置两次线程化收集之间的时间间隔。
- **COLL_DBD_FRENAME_RETRY_CNT**：使用此环境变量可以指定将 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_TEMP 文件移至 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_PREV 文件的尝试次数。

如果您未指定此变量的值，那么 Microsoft SQL Server 代理程序 会尝试移动该文件 1 次。

用于增强“MS SQL 数据库详细信息”数据集收集的变量

要增强“MS SQL 数据库详细信息”数据集收集，请使用以下环境变量：

- **COLL_KOQDBD_INTERVAL**：使用此环境变量可以指定“MS SQL 数据库详细信息”数据集的基于线程的两次连续收集之间的时间间隔（以分钟为单位）。如果未指定此变量的值或指定的时间间隔少于 3 分钟，那么 Microsoft SQL Server 代理程序缺省设置为 3 分钟的时间间隔。如果收集耗用更多时间或数据经常被看作 NOT_COLLECTED，那么您可以检查收集时间，方法是参阅数据库详细信息收集在 %d 秒内完成日志并将变量值设置为比日志中指定的收集时间长的值。
- **COLL_DBD_FRENAME_RETRY_CNT**：使用此环境变量可以指定将 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_TEMP 文件移至 %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_PREV 文件的尝试次数。

如果您未指定此变量的值，那么 Microsoft SQL Server 代理程序 会尝试移动该文件 1 次。

用于增强“MS SQL 审计详细信息”数据集收集的变量

要增强“MS SQL 审计详细信息”数据集收集，请使用以下环境变量：

- **COLL_AUDIT_TYPE**：使用此变量可以允许或禁止监视特定日志。此变量的缺省值为 [AL][FL][SL]。缺省情况下，代理程序将监视所有三种类型的日志，包括应用程序日志、审计文件和安全日志。对于每种日志类型，变量的值包含双字符代码：
  - [AL] 表示应用程序日志
  - [FL] 表示审计文件
  - [SL] 表示安全日志

  您可以更改此变量的值，以禁止监视特定日志类型。例如，如果您将该变量的值指定为 [AL][SL]，那么不会监视审计文件。如果未对此变量指定任何值，那么不会监视审计详细信息。

- **COLL_AUDIT_DURATION**：使用此变量可以报告您在此变量中指定的时间间隔内发生的审计事件。例如，如果您将此变量设置为 7，那么“审计详细信息”数据集会报告仅在过去 7 小时内发生的审计事件。**COLL_AUDIT_DURATION** 变量的缺省值为 24 小时。
- **COLL_AUDIT_COLLECTION_INTERVAL**：“审计详细信息”数据集中的线程化收集用于提供 SQL Server 实例中存在的所有数据库的规范。使用此变量可以设置该线程化收集的时间间隔。例如，如果您将此变量设置为 7，那么每 7 个小时后将从 SQL Server 实例中抽取一组全新的数据库规范。**COLL_AUDIT_COLLECTION_INTERVAL** 变量的缺省值为 24。

用于增强“MS SQL 进程详细信息”数据集收集的变量

要增强“MS SQL 进程详细信息”数据集收集，请将 **COLL_PROC_BLOCK_INTERVAL** 变量与下列值配合使用：

- 如果 **COLL_PROC_BLOCK_INTERVAL =** 0，那么“阻塞进程持续时间”属性的收集和“阻塞资源持续时间”属性处于禁用状态。
- 如果 **COLL_PROC_BLOCK_INTERVAL =** *x*，那么“阻塞进程持续时间”和“阻塞资源持续时间”属性的两次连续数据收集之间的时间间隔为 *x* 分钟。

如果未在 CANDLE_HOME 目录中设置 **COLL_PROC_BLOCK_INTERVAL** 变量，那么两次连续数据收集之间的时间间隔为 3 分钟。

用于从数据收集中排除锁定对象的变量

如果针对“数据库详细信息”、“文件组详细信息”、“数据库监视”和“设备详细信息”工作空间发送的查询的执行时间过长，请使用 **COLL_DBCC_NO_LOCK** 变量以值 WITH (NOLOCK) 运行查询。如果对其运行此查询的对象处于锁定状态，那么此变量会导致该查询不在队列中等待。

用于设置“表详细信息”数据集所返回的行的排序条件

“表详细信息”数据集所返回的行将以降序排序，具体取决于为 **COLL_TBLD_SORTBY** 变量设置的值。**COLL_TBLD_SORTBY** 变量的缺省值为 FRAG（分段百分比）。有效值包括：ROWS（表中的行数）、SPACE（表所使用的空间）和 OPTSAGE（表的优化器统计期）。

用于增强“MS SQL 问题详细信息”数据集和“问题摘要”数据集收集的变量

- **COLL_ALERT_SEV**：使用此变量可以设置“问题详细信息”数据集和“问题摘要”数据集中显示的错误消息的严重性级别。严重性级别等于或高于此变量中提及的值的错误消息将显示在“问题详细信息”数据集和“问题摘要”数据集中。例如，如果您将此变量的值设置为 10，那么严重性级别为 10 或以上的错误消息将显示在“问题详细信息”数据集和“问题摘要”数据集中。如果您未对此变量指定值，那么严重性级别等于或大于 17 的错误消息将显示在“问题详细信息”数据集和“问题摘要”数据集中。
- **COLL_SINCE_ERRORLOG_RECY**：使用此变量可以仅监视当前 ERRORLOG 文件中的高严重性错误。如果您未对此变量指定值，那么此变量的值为 0，这表示针对收集数据，“问题摘要”数据集还会考虑从先前的 ERRORLOG 文件中读取的高严重性错误。要仅监视当前 ERRORLOG 文件中的高严重性错误，请将此变量的值设置为 1。

用于设置超时时间间隔的变量

要设置 Microsoft SQL Server 代理程序 的超时时间间隔，您可以使用以下环境变量：

- **WAIT_TIMEOUT**：使用此变量可以设置 Microsoft SQL Server 代理程序 的等待超时时间间隔。如果任何数据集收集数据的时间超过 45 秒，那么代理程序可能已挂起或者可能错误地触发了情境。请检查收集数据花费的时间超过 45 秒的数据集的日志，然后使用 **WAIT_TIMEOUT** 变量延长代理程序进程和收集器进程之间的等待时间。
- **COLL_DB_TIMEOUT**：使用此变量可以定义任何请求（例如，对现有 SQL Server 连接运行查询）在返回到应用程序之前等待的时间间隔（以秒为单位）。如果您将此值设置为 0，那么表示不存在超时。如果您未对此变量指定值，那么代理程序将在返回到应用程序之前等待 15 秒。

用于设置收集器日志文件的属性的变量

要设置收集器日志文件的属性，您可以使用以下环境变量：

- **COLL_WRAPLINES**：使用此变量可以指定 col.out 文件中的最大行数。此变量的缺省值为 90,000 行（大于 2 MB）。
- **COLL_NUMOUTBAK**：使用此变量可以指定您要创建的收集器日志文件的备份副本数。缺省情况下，将创建五个收集器日志文件备份副本。备份文件名为 *.out。此备份文件已满时，它将重命名为 *.ou1，而最新的日志将写入 *.out 文件。通过此方式，对于这五个备份文件，最旧的日志在 *.ou5 文件中提供，而最新的日志在 *.out 文件中可用。

  通过在 **COLL_NUMOUTBAK** 变量中指定下列其中一个值，您可以创建五个以上的收集器日志文件备份副本：

  – 如果备份文件数小于 10，请指定您要在 **COLL_NUMOUTBAK** 变量中创建的备份文件数。例如，如果您在 **COLL_NUMOUTBAK** 变量中指定了 9，那么将创建 9 个备份文件。
  – 如果备份文件数大于 9 小于 1000，请在 **COLL_NUMOUTBAK** 变量中指定前面带连字符的备份文件数。例如，如果您在 **COLL_NUMOUTBAK** 变量中指定 -352，那么将创建 352 个备份文件。
- **COLL_DEBUG**：使用此变量可以通过将其值设置为 ddddddddd（10 乘以“d”）来启用对收集器的完全跟踪。

用于删除临时文件的变量

COLL_TMPFILE_DEL_INTERVAL：使用此变量可以指定应该删除 KOQ_<timestamp> 临时文件之前的时间间隔（以分钟为单位）。如果您未对此变量指定值，那么该变量的值为 0，这表示必须立即删除临时文件。

用于更改 MS SQL Server 代理程序使用的驱动程序的变量

要更改 Microsoft SQL Server 代理程序使用的驱动程序，请使用 **KOQ_ODBC_DRIVER** 环境变量。此变量指定 Microsoft SQL Server 代理程序用于连接到 SQL Server 的驱动程序。如果未指定此变量的值，那么代理程序使用 ODBC SQL Server 驱动程序作为缺省驱动程序。

注: 指定 Microsoft SQL Server 驱动程序时，请确保驱动程序名称正确并在数据源 (ODBC) 中的驱动程序选项下列出该驱动程序。

用于连接到 AlwaysOn 的变量启用了 SQL Server 数据库

KOQ_APPLICATION_INTENT：此变量用于在连接到 SQL Server 时指定连接选项。
KOQ_APPLICATION_INTENT 选项详细信息：

- **Readonly**：打开 **ApplicationIntent** 为 *readonly* 的连接。
- **Readwrite**：打开 **ApplicationIntent** 为 *readwrite* 的连接。
  当设置为 Readwrite 时，Microsoft SQL Server 代理程序不会使用连接执行任何写操作。

如果未设置此变量，那么会建立没有 **ApplicationIntent** 属性的连接。

注: 驱动程序由环境变量 **KOQ_ODBC_DRIVER** 指定。如果未设置此变量，那么将使用缺省 SQL Server 驱动程序。
如果驱动程序不支持 **ApplicationIntent**，那么会打开没有 **ApplicationIntent** 属性的连接。

代理程序的配置参数

您必须提供代理程序的必需配置参数。

关于此任务

下表包含配置参数的详细信息。请复查参数并确定每个参数的值。

参数名称	描述	缺省值	必填字段

用户名	用于建立代理程序与 SQL Server 之间的连接的用户名或登录名	NA	是
密码	用户名或登录名的密码	NA	是
数据库版本	要监视的 SQL Server 数据库的版本	NA	是
数据库服务器主目录	SQL Server 数据库的主路径	NA	是
错误日志文件路径	SQL Server 错误日志文件所在的位置	NA	是

在 Windows 系统上配置代理程序

您可以使用 IBM® Cloud Application Performance Management 窗口在 Windows 系统上配置代理程序。

开始之前

在配置代理程序之前，请确保完成了以下任务：

· 创建用户和授予所需许可权

· 复查本地环境变量

关于此任务

Microsoft SQL Server 代理程序 是一个多实例代理程序；您必须手动配置和启动每个代理程序实例。

· 要配置代理程序，请完成以下任务：

– 选择要监视的数据库

– 配置本地环境变量

选中要监视的数据库

可以通过使用“**配置数据库代理程序**”窗口选择要监视的数据库。

过程

1. 打开 **IBM Performance Management** 窗口。
2. 在“**IBM Performance Management**”窗口中，单击**任务/子系统**列、右键单击**模板**，然后单击**使用缺省值进行配置**。
3. 在“**配置数据库代理程序**”窗口中，从**可用数据库服务器**中选择要监视的数据库服务器，并将其移至**要监视的服务器**列表。
4. 在“**数据库服务器属性**”窗口中，将自动填写以下字段的值：

   · 服务器名称

   · 数据库版本

   · 主目录

   · 错误日志文件

   “**数据库服务器属性**”窗口中的以下字段为可选字段：

   · Windows 认证

   · 支持长期活动的数据库连接

· 扩展参数
· 监视所有数据库
· 频率（天）
· 频率（每周）
· 频率（每月）
· 收集开始时间
· 表详细信息连续收集

有关“**数据库服务器属性**”窗口中的配置参数的更多信息，请参阅第 476 页的『“数据库服务器”属性的配置参数』。

5. 如果不选择 **Windows 认证**字段，那么请仅使用 ASCII 字符在**登录名**和**密码**字段中输入登录名和密码。
6. 在**扩展参数**字段中，输入数据集名称以禁用数据收集，然后单击**确定**。

   例如：

   · 输入 `koqtbld` 以禁用“表详细信息”数据集的数据收集。
   · 输入 `koqdbd` 以禁用“数据库详细信息”数据集的数据收集。
   · 输入 `koqtbld,koqdbd` 以禁用“表详细信息”数据集和“数据库详细信息”数据集的数据收集。
7. 如果不选中**监视所有数据库**复选框，请在**数据库**组区域的字段中指定要为其启用或禁用监视的数据库列表。

   切记: 如果选中**监视所有数据库**复选框，并在**数据库**组区域中指定数据库，那么将优先选用**监视所有数据库**复选框的设置。
8. 指定“MS SQL 表详细信息”数据集的收集频率。可能的值为：每天 (daily)、每周 (weekly) 或每月 (monthly)。
9. 选中**表详细信息连续收集**复选框，以便为“MS SQL 表详细信息”数据集启用连续收集。如果选中**表详细信息连续收集**复选框，请在**两次连续收集之间的时间间隔（分钟）**字段中输入值。
10. 在“**配置数据库代理程序**”窗口中，单击**确定**，然后启动代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

“数据库服务器”属性的配置参数

在“**数据库服务器属性**”窗口中，您可以配置“数据库服务器”属性，例如，服务器名称、数据库版本和主目录。

下表包含“**数据库服务器属性**”窗口中的配置设置的详细描述。

表 *182.* “**数据库服务器属性**”窗口中配置设置的名称和描述

参数名称	描述	必填字段	示例
服务器名称	要监视的 Microsoft SQL Server 实例的名称。 使用 MSSQLSERVER 作为缺省实例的名称。 名称必须足够短以适合整个受管系统名称，其长度长度必须为 2 - 32 个字符。	是	如果要监视的 Microsoft SQL Server 实例是缺省 Microsoft SQL Server 实例，请在此字段中输入 `MSSQLSERVER`。 如果要监视的 Microsoft SQL Server 实例是命名实例，其中实例名称是 `mysqlserver`，主机名称是 `popcorn`，那么请在此字段中输入 `mysqlserver`。
登录	要用于连接到 Microsoft SQL Server 的 Microsoft SQL Server 用户标识。 仅当 **Windows 认证**参数设置为 `False` 时才需要用户标识。 请仅对用户标识使用 ASCII 字符。 通过在**登录名**字段中指定登录标识来配置 Microsoft SQL Server 代理程序时，代理程序使用此登录标识连接到 Microsoft SQL Server。 **要点:** 如果选中 **Windows 认证**复选框并在**登录名**字段中指定登录标识，在配置代理程序时，代理程序会优先使用“Windows 认证”。	否	
密码	Microsoft SQL Server 用户标识的密码。 仅当 **Windows 认证**参数设置为 `False` 时，才需要密码。 请仅对密码使用 ASCII 字符。	否	
数据库版本	SQL Server 实例的版本。	是	此 SQL Server 实例的数据库版本如下所示: · Microsoft SQL Server 2014 - 12.0.2000.8 · Microsoft SQL Server 2012 - 11.0.2100.60 · Microsoft SQL Server 2008 R2 - 10.50.1600.1 · Microsoft SQL Server 2008 - 10.0.1600.22 · Microsoft SQL Server 2005 - 9.0.1399.06

表 *182.* “**数据库服务器属性**”窗口中配置设置的名称和描述 *(续)*

参数名称	描述	必填字段	示例
主目录	SQL Server 安装目录。	是	缺省 Microsoft SQL Server 2005 实例的缺省主目录路径为 `C:\Program Files\Microsoft SQL Server\MSSQL`。 指定的 Microsoft SQL Server 2005 实例具有 `C:\Program Files\Microsoft SQL Server\MSSQL$`*instance_name* 格式的缺省主目录路径，其中，*instance_name* 是 Microsoft SQL Server 实例名称。
错误日志文件	SQL Server 错误日志的标准位置和名称。	是	缺省 Microsoft SQL Server 2005 实例的缺省错误日志路径为 `C:\Program Files\Microsoft SQL Server\MSSQL\LOG\ERRORLOG`。 指定的 Microsoft SQL Server 2005 示例具有 `C:\Program Files\Microsoft SQL Server\MSSQL$`*instance_name* `\LOG\ERRORLOG` 格式的缺省错误日志路径，其中，*instance_name* 是 Microsoft SQL Server 实例名称。

表 182. “**数据库服务器属性**”窗口中配置设置的名称和描述 (续)

参数名称	描述	必填字段	示例
Windows 认证	“Windows 认证”是配置代理程序服务所使用的 Windows 帐户，且是缺省配置选项。 如果选中 **Windows 认证**复选框，那么会将 Windows 凭证用于认证。 使用 Windows 认证来配置 Microsoft SQL Server 代理程序时，代理程序服务使用**本地系统帐户**或**此帐户**来登录到 Microsoft SQL Server。 · 如果代理程序服务配置为使用**本地系统帐户**登录，那么代理程序会使用 NT AUTHORITY \SYSTEM 用户标识访问 Microsoft SQL Server。 · 如果代理程序服务配置为使用**此帐户**登录，那么代理程序会使用各自的用户标识访问 Microsoft SQL Server。 **切记:** 如果未选中 **Windows 认证**复选框，那么您必须为**登录**和**密码**参数指定值。如果未指定这些参数并在“**数据库服务器属性**”窗口中单击了**确定**，那么弹出窗口中将显示一条错误消息，并且代理程序配置未完成。 **要点:** 如果通过选中 **Windows 认证**复选框并在**登录名**字段中指定登录标识来配置代理程序，那么代理程序会优先使用“Windows 认证”。	否	
支持长期活动的数据库连接	启用或禁用长期活动的数据库连接。以下数据集不使用长期存在的数据库连接: · MS SQL 文本 · MS SQL 文件组详细信息 · MS SQL Server 摘要	否	
扩展参数	禁用任何属性组的数据收集。	否	例如: 要禁用“表详细信息”数据集的数据收集，在**扩展参数**字段中输入 `koqtbld`。 要禁用“数据库详细信息”数据集的数据收集，在**扩展参数**字段中输入 koqdbd。 要禁用“表详细信息”和“数据库详细信息”数据集的数据收集，在**扩展参数**字段中输入 `koqtbld,koqdbd`。

表 *182.* “**数据库服务器属性**”窗口中配置设置的名称和描述 *(续)*

参数名称	描述	必填字段	示例

数据库	要选择数据库进行监视，请为此参数指定值。要对 SQL Server 实例上可用的所有数据库启用监视，请选中**数据库**组区域中的**监视所有数据库**复选框。缺省情况下，**监视所有数据库**复选框处于选中状态。
	
	要对特定数据库启用或禁用监视，请取消选中**监视所有数据库**复选框。
	· 要监视特定数据库，请从列表中选择**包含**，然后在列表旁边的文本字段中指定数据库的名称。
	· 要排除特定数据库而不进行监视，请从列表中选择**排除**，然后在列表旁边的文本字段中指定数据库的名称。
	
	使用文本字段以过滤想要监视的数据库。
	
	要指定数据库过滤器，首先必须选择一个分隔符。分隔符是区分数据库名称或数据库表达式与其他数据库名称或数据库表达式的字符。
	
	选择分隔符时，请确保数据库名称和分隔符表达式不包含选作分隔符的字符。在用于数据库名称或数据库表达式时，不得使用在 T-SQL 查询中常用的通配符（例如，%、_、[]、^、-）。
	
	当指定数据库过滤器时：
	· 数据库名称必须以一个分隔符开头。
	· 数据库表达式必须以 2 个分隔符开头。
	
	数据库表达式是可在 T-SQL 查询的 LIKE 部分中使用的有效表达式。但是，在指定数据库表达式时不能使用 T-SQL ESCAPE 子句。以下数据集受数据库过滤器的影响：“数据库详细信息”、“数据库摘要”、“设备详细信息”、“表详细信息”、“表摘要”、“文件组详细信息”和“其他数据库详细信息”
		否	过滤器示例：
			
			情况 1：% usage
			
			示例：
			
			`@@%m%`
			
			输出：将过滤名称中包含字符 m 的所有数据库。
			
			情况 2：_ usage
			
			示例：
			
			`@@____`
			
			输出：将过滤长度为四个字符的所有数据库。
			
			情况 3：[] usage
			
			示例：
			
			`@@[m]___`
			
			输出：将过长度为四个字符且名称以字符 m 开头的所有数据库。
			
			情况 4：[^] usage
			
			示例：
			
			`@@[^m]%`
			
			输出：将过滤名称以 *m* 开头的数据库之外的所有数据库（任意长度）。

表 *182.* “**数据库服务器属性**”窗口中配置设置的名称和描述 *(续)*

参数名称	描述	必填字段	示例

数据库（继续）	**切记:**
	· 如果您未选中**监视所有数据库**复选框，那么您必须在**数据库**组区域中出现的文本字段中指定您要对其启用或禁用监视的数据库的列表。如果在“**数据库服务器属性**”窗口中单击了**确定**，但未选中**监视所有数据库**复选框并指定了数据库列表，那么弹出窗口中将显示一条错误消息，并且代理程序配置未完成。
	· 如果选中了**监视所有数据库**复选框，并且还在**数据库**组区域中出现的文本字段中指定了要监视的数据库，那么将对**监视所有数据库**复选框的值指定优先级。将忽略您在文本字段中指定的数据库列表。
			情况 5：错误输入
			示例：
			`@%m%`
			输出：不过滤任何数据库。
			情况 6：缺省值
			示例：字段为空（不输入任何查询）
			输出：将过滤所有数据库。
			情况 7：混合模式
			示例：
			`@@[m-t]_d%`
			输出：将过滤名称以字符 m、n、o、p、q、r、s 或 t 开头后跟任意字符且第三个位置的字符为 d 的所有数据库（任意长度）。

频率（天）	使用此功能部件来定义收集“表详细信息”属性的数据的频率。值可以从 0 到 31。	否	
频率（每周）	使用此功能以指定用于收集“表详细信息”属性的数据的特定一天。值范围为 0 到 7。	否	
频率（每月）	使用此功能以定义月份特定一天的“表详细信息”属性的数据收集。可能的值为 1、2 和 3 等。	否	
收集开始时间	可以输入 HH:MM 格式的收集开始时间。 小时数的可能值介于 0 到 23。缺省值为 0。 分钟数的可能值介于 0 到 59。缺省值为 0。	否	
表详细信息连续收集	此功能用于“表详细信息”数据的连续后台收集。 缺省情况下，选中**表详细信息连续收集**复选框。	否	
两次连续收集之间的时间间隔（分钟）	指定两次收集之间的时间间隔时间（以分钟为单位）。最小时间间隔时间为 3 分钟。 您可选中**两次连续收集之间的时间间隔（分钟）**复选框，或者使用“调度”来指定“表详细信息”数据集的连续收集。如果选择**两次连续收集之间的时间间隔（分钟）**复选框，那么必须指定收集的时间间隔。如果使用“调度”来指定“表详细信息”数据集的连续收集，那么最短时间间隔为 1 天。 两个连续收集之间的缺省时间间隔为 3 分钟。	否	

代理程序按照频繁发生的数据收集的时间间隔来收集数据。例如，如果指定收集数据的所有频率（每日、每周和每月），那么代理程序会根据以下条件启动数据收集：

· 如果每日频率 ⩽ 7，那么选择每日频率设置，忽略每周和每月频率设置。

· 如果每日频率 > 7，那么选择每周频率设置，并忽略每日和每月频率设置。

切记: 如果选中**表详细信息连续收集**复选框，那么代理程序按照**两次连续收集之间的时间间隔（分钟）**字段中提及的时间间隔收集数据，而不是按照每日、每周或每月频率。

在 Windows 系统上配置本地环境变量

可以配置本地环境变量以更改 Microsoft SQL Server 代理程序的行为。

过程

1. 单击**开始> 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在“**IBM Performance Management**”窗口中的**操作**菜单中，单击**高级 > 编辑变量**。
3. 在 Monitoring Agent for Microsoft SQL Server 的“**覆盖局部变量设置**”窗口中，单击**添加**。
4. 在“**添加环境设置覆盖**”窗口中，输入变量和相应值。

   注: 请参阅第 470 页的『本地环境变量』，以获取可配置的环境变量的完整列表。

以非管理员用户身份运行

您可以非管理员用户身份来运行 Microsoft SQL Server 监视代理程序。

关于此任务

可以非管理员用户身份从“域用户”组运行 Microsoft SQL Server 代理程序。

过程

1. 启动 Windows 应用程序“Active Directory 用户和计算机”，并创建域用户。
   · 确保新用户是 *Domain Users* 组的成员。
   · 确保 SQL Server 是 *Domain Computers* 的成员。
2. 将新创建的域用户添加到 *SQL Server Login* 用户组。域用户应具有 SQL Server 上的 **sysadmin** SQL Server 角色许可权或者具有 https://www.ibm.com/support/knowledgecenter/SSMKFH/com.ibm.apmaas.doc/install/sql_config_agent_grant_permission_sqlserver.htm 中所述的许可权。
3. 以域管理员身份登录到 SQL Server。
4. 向 Microsoft SQL Server 代理程序访问的每个驱动器授予**修改**许可权。完成以下过程以降该许可权传播至所有子文件夹：
   a) 转至**我的电脑**。
   b) 右键单击**驱动器**。
   c) 单击**安全**选项卡。
   d) 添加新创建的用户。
   e) 向新创建的用户授予**修改**许可权。
   f) 单击**确定**。此过程需要几分钟时间以将许可权应用于所有子目录。
5. 通过使用 Windows 注册表，授予对 HKEY_LOCAL_MACHINE 的读访问权，并传播设置。完成下列步骤以传播设置：
   a) 右键单击 HKEY_LOCAL_MACHINE 目录并选择**许可权**。
   b) 添加新创建的用户。
   c) 选中新创建的用户。
   d) 选中**允许读取**复选框。
   e) 单击**确定**。此过程需要几分钟时间以将设置传播至整个 HKEY_LOCAL_MACHINE 树。
6. 通过使用 Windows 注册表，根据以下列表授予特定于代理程序的注册表许可权。

- 如果将 32 位代理程序安装在 32 位操作系统上，请授予对 KEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring 目录的完全访问权，然后传播设置。
- 如果将 32 位代理程序安装在 64 位操作系统上，请授予对 HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Candle 目录的完全访问权，然后传播设置。
- 如果将 64 位代理程序安装在 64 位操作系统上，请授予对 KEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring 目录的完全访问权，然后传播设置。

完成下列步骤以传播设置：

a) 右键单击您拥有其完全访问权的目录，并选择**许可权**。

b) 添加新创建的用户。

c) 选中新创建的用户。

d) 选中**允许完全控制**复选框。

e) 单击**确定**。此过程需要几分钟时间以将设置传播至整个 KEY_LOCAL_MACHINE\SOFTWARE\IBMMonitoring 树。

7. 将新的域用户添加至 **Performance Monitor Users** 组。
8. 确认域用户是 *Users* 组的成员。
9. 授予对 Windows 目录的下列许可权，以便以非管理员用户身份运行：
   - 如果 32 位代理程序安装在 32 位操作系统上，请授予对 OS_installation_drive:\Windows\system32 目录的读写访问权
   - 如果 32 位代理程序安装在 64 位操作系统上，请授予对 OS_installation_drive:\Windows\SysWOW64 目录的读写访问权

   注: 对于 Windows Server 2008、Windows Server 2008 R2 以及 Windows Server 2012、Windows Server 2012 R2 和 Windows Server 2016，不需要 Windows 目录的许可权。
10. 授予对 SQL Server 数据文件和日志文件的**修改**许可权：
    - SQL Server 数据文件的缺省路径为 *SQLServer_root_dir*\DATA，其中 *SQLServer_root_dir* 是 SQL Server 实例的根目录。例如，如果 SQL Server 实例的根目录是 C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL，那么数据文件路径为 C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\DATA。
    - SQL Server 日志文件的缺省路径为 *SQLServer_root_dir*\LOG，其中 *SQLServer_root_dir* 是 SQL Server 实例的根目录。例如，如果 SQL Server 实例的根目录是 C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL，那么日志文件路径为 C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\LOG。
11. 授予对 Candle_Home 目录的完全许可权。缺省路径为 C:\IBM\ITM。
12. 应用本地安全许可权，请参阅第 484 页的『本地安全策略许可权』。
13. 重新启动 SQL Server，以确保有效应用本地安全许可权。
14. 通过完成下列步骤，将 SQL Server 代理程序服务的登录设置更改为非管理员用户：

    a) 单击**开始 > 管理工具 > 服务**。

    b) 右键单击 **Monitoring Agent For SQL Server *instance_name***，并单击**属性**。“**SQL 服务属性**”窗口随即打开。

    c) 单击**登录**选项卡。

    d) 单击**此帐户**，并输入用户名。

    e) 在**密码**和**确认密码**字段中输入密码，然后单击**确定**。

    f) 针对 **Monitoring Agent For SQL Server Collector *instance_name*** 重复步骤 b - e，其中 *instance_name* 是 Microsoft SQL Server 实例名称。

本地安全策略许可权

本地安全策略用于管理系统及其安全策略。它在确保代理程序以及代理程序安装所在系统的安全方面扮演着重要的角色。此策略通过向用户授予访问权和许可权来工作。对于 Microsoft SQL Server 代理程序，请确保用户具有下列许可权，以符合本地安全许可权策略。

“作为服务登录”许可权

关于此任务

要授予“作为服务登录”许可权，请完成以下步骤。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 单击“**本地策略**”以展开列表。
3. 单击**用户权限分配**。这样会打开用户权限列表。
4. 双击**作为服务登录**策略。“**作为服务登录属性**”窗口随即打开。
5. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
6. 在**输入要选中的对象名称**字段中，输入要为其分配许可权的用户帐户名称，然后单击**确定**。
7. 单击**确定**。

“调试程序”许可权

关于此任务

要授予调试程序许可权，请在 Microsoft SQL Server 代理程序上完成以下过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 单击“**本地策略**”以展开列表。
3. 单击**用户权限分配**。这样会打开用户权限列表。
4. 双击**调试程序**策略。“**调试程序属性**”窗口随即打开。
5. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
6. 在**输入要选中的对象名称**字段中，输入要为其分配许可权的用户帐户名称，然后单击**确定**。
7. 单击**确定**。

身份验证后模拟客户端

关于此任务

要授予“身份验证后模拟客户端”许可权，请在 Microsoft SQL Server 代理程序上完成以下过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 单击“**本地策略**”以展开列表。
3. 单击**用户权限分配**。这样会打开用户权限列表。
4. 双击**身份验证后模拟客户端**策略。“**身份验证后模拟客户端属性**”窗口随即打开。
5. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
6. 在**输入对象名称来选择**字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
7. 单击**确定**。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

开始之前

在配置代理程序之前，请确保完成了以下任务：

· 复查本地环境变量

关于此任务

Microsoft SQL Server 代理程序 是一个多实例代理程序；您必须手动配置和启动每个代理程序实例。

过程

1. 在命令行中，更改代理程序安装目录的路径。
   示例：
   ```
   cd /opt/ibm/apm/agent/bin
   ```
2. 运行以下命令，其中 *instance_name* 是要对实例指定的名称：
   ```
   ./mssql-agent.sh config instance_name
   ```
3. 当命令提示符显示以下消息时，请输入 1 并输入：
   `Edit ‘Monitoring Agent for MSSQL setting? [1=Yes, 2=No]`
4. 出现提示时，指定配置参数的值。
   有关配置参数的信息，请参阅“代理程序的配置参数”。
5. 运行以下命令以启动代理程序：
   ```
   ./mssql-agent.sh start instance_name
   ```
6. 运行以下命令以停止代理程序：
   ```
   ./mssql-agent.sh stop instance_name
   ```

在 Linux 系统上配置本地环境变量

可以在 Linux 系统上配置本地环境变量以更改 Microsoft SQL Server 代理程序的行为。

过程

1. 启动终端或系统文件管理器，切换至代理程序安装目录：
   示例：
   ```
   /opt/ibm/apm/agent
   ```
2. 运行以下命令以停止代理程序：
   ```
   ./mssql-agent.sh stop instance_name
   ```
   其中，*instance_name* 是代理程序实例名称。
3. 打开以下配置目录中的 `.oq.environment` 文件：
   示例：
   ```
   install_dir/config
   ```
   其中，*install_dir* 是代理程序安装目录。
4. 按以下名称/值对格式在 `.oq.environment` 文件末尾添加所需的环境变量。
   `export VARIABLE_NAME=VARIABLE_VALUE`

示例：

```
export KOQ_ODBC_DRIVER=ODBC Driver 17 for SQL Server
```

注：

· 请参阅第 470 页的『本地环境变量』，以获取可配置的环境变量的完整列表。

· 代理程序升级后，不保留添加的定制的变量。

5. 保存该文件。
6. 从代理程序安装目录启动代理程序：

```
cd /opt/ibm/apm/agent/bin
```

```
./mssql-agent.sh start instance_name
```

使用静默响应文件来配置代理程序

您可使用静默响应文件来配置代理程序或代理程序的多个实例。

开始之前

要配置代理程序的多个实例，请确保静默响应文件中已指定所有代理程序实例的配置详细信息。

关于此任务

运行配置脚本以更改配置设置。您可以先编辑静默响应文件，然后再运行配置脚本。

过程

要配置代理程序，请完成以下步骤：

1. 启动文本编辑器，打开位于以下位置的静默响应文件：
   · Windows *install_dir*\samples/mssql_silent_config.txt
   · Linux *install_dir*/samples/mssql_silent_config.txt

   其中，*install_dir* 是代理程序安装目录。

   示例：
   · Windows C:\IBM\APM\samples\mssql_silent_config.txt
   · Linux /opt/ibm/apm/agent/samples/mssql_silent_config.txt

   注： 有关代理程序配置参数的信息，请参阅第 474 页的『代理程序的配置参数』。
2. 启动命令提示符，切换至以下目录：
   · Windows

   ```
   cd install_dir\bin
   ```

   · Linux

   ```
   cd install_dir/bin
   ```
3. 运行以下命令：
   · Windows

   ```
   mssql-agent.bat config install_dir\samples\mssql_silent_config.txt
   ```

   · Linux

   ```
   mssql-agent.sh config  instance_name install_dir/samples/mssql_silent_config.txt
   ```

4. 启动代理程序。

- Windows 在 **IBM Performance Management** 窗口中，右键单击已创建的代理程序实例，然后单击“启动”。
- Linux 运行以下命令：

```
cd /opt/ibm/apm/agent/bin
```

```
./mssql-agent.sh start instance_name
```

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

在集群环境中运行代理程序

Windows 您可以在集群环境中配置 Microsoft SQL Server 代理程序。Microsoft SQL Server 的多个实例以及 Microsoft SQL Server 代理程序 可以在单节点上运行。

安装和配置 Microsoft SQL Server 代理程序后，为在集群环境中运行该代理程序，请完成以下任务：

- 添加环境变量
- 更改代理程序服务和收集器服务的启动类型
- 将代理程序和收集器添加到集群环境中

您可以为 Microsoft SQL Server 的以下版本设置集群环境：

- Microsoft SQL Server 2005
- Microsoft SQL Server 2008
- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2012
- Microsoft SQL Server 2014
- Microsoft SQL Server 2016

要点: 在 Windows 系统上，此代理程序必须安装在 OS 代理程序所安装的同一目录中。在各个集群节点的节点系统磁盘上安装代理程序。

添加环境变量

必须对各个集群节点上安装的代理程序所使用的环境变量进行配置。

关于此任务

必须指定以下环境变量的值：

- *CTIRA_HOSTNAME*：此变量用于配置 Microsoft SQL Server 代理程序的各个实例。此变量的值限于 31 个字符，且通用于所有监视代理程序。将此变量的值设置为集群名称，以浏览至 应用程序性能仪表板 上该集群的所有监视代理程序。
- *CTIRA_NODETYPE*：此变量用于识别代理程序。缺省情况下，此变量的值设置为 Microsoft SQL Server 代理程序的 **MSS**。
- *CTIRA_SUBSYSTEMID*：此变量用于区分 Microsoft SQL Server 代理程序 的多个实例。缺省情况下，此变量的值设置为 Microsoft SQL Server 代理程序的 **Microsoft SQL Virtual Server**。
- *COLL_HOME*：此变量用于为属性组（使用共享位置的配置文件）收集数据和存储日志文件。将 变量值设置为 *X*:\shared-location，其中 *X* 是集群节点可访问的共享驱动。例如，为“MS SQL 表详细信息”属性组或“MS SQL 错误事件详细信息”属性组定义配置设置时，请设置 *COLL_HOME* 变量的值。

· *CTIRA_HIST_DIR*：此变量用于指定共享磁盘目录的路径。如果 Microsoft SQL Server 代理程序的历史记录配置为存储在监视代理程序上，那么该代理程序的各个实例必须配置了指向共享磁盘目录的通用 *CTIRA_HIST_DIR* 变量。

切记: 如果历史记录存储在 Cloud APM 服务器上，那么无需指定 *CTIRA_HIST_DIR* 变量的值。但是，将历史记录存储在 Cloud APM 服务器上会增加服务器的负载。

要添加这些变量，请参阅 第 482 页的『在 Windows 系统上配置本地环境变量』 中所描述的步骤。

下一步做什么

通过完成 第 488 页的『更改代理程序服务和收集器服务的启动类型』 中所描述的步骤，将代理程序服务和收集器服务的启动类型更改为**手动**。

更改代理程序服务和收集器服务的启动类型

缺省情况下，代理程序服务和收集器服务的启动类型为**自动**。将代理程 序服务和收集器服务的启动类型更改为**手动**，以使集 群资源可以控制监视代理程序的开始和停止

过程

要更改代理程序服务的启动类型，请完成下列步骤：

1. 单击**开始 > 运行**，输入 `services.msc` 命令，然后单击**确定**。
2. 右键单击此代理程序，然后单击**属性**。
3. 在“**Monitoring Agent for Microsoft SQL Server 属性**”窗口中，从**启动类型**列表中选择**手动**，然后单击**应用**，最后单击**确定**。

下一步做什么

· 使用相同的过程将收集器服务的启动类型更改为**手动**。

· 通过完成 第 488 页的『将代理程序和收集器添加到集群环境中 』 中所描述的步骤，将代理程序和收集器添加到集群环境中。

将代理程序和收集器添加到集群环境中

您必须将代理程序和收集器添加到集群环境中。

过程

1. 单击**开始 > 控制面板 > 管理工具 > 故障转移集群管理**。
2. 展开**故障集群管理**。
3. 展开**服务和应用程序**，然后右键单击要配置的 SQL 实例。
4. 单击**添加资源 > 通用服务**。此时将打开“新建资源”向导。
5. 在“选择服务”页面上，选择服务名称，然后单击**下一步**。

   Windows 服务名称示例：

   · `Monitoring Agent for Microsoft SQL Server: SQLTEST#INSTANCE1`

   · `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST#INSTANCE1`

   · `Monitoring Agent for Microsoft SQL Server: SQLTEST2#INSTANCE2`

   · `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST2#INSTANCE2`
6. 在“配置”页面上，检查详细信息，然后单击**下一步**。
7. 在“摘要”页面上，单击**完成**。此时已添加 Microsoft SQL Server 代理程序。

   切记: 使用相同的步骤将收集器添加到集群环境中。
8. 要使代理程序联机，请右键单击该代理程序，然后单击**使资源联机**。
9. 要使收集器联机，请右键单击该收集器，然后单击**使资源联机**。

结果
此时 Microsoft SQL Server 代理程序 正在集群环境中运行。

切记: 如果要重新配置代理程序，必须首先使代理程序和收集器脱机，或在运行代理程序和收集器的节点上编辑此代理程序的变量。完成代理程序配置后，重新使代理程序和收集器联机。

使用集群实用程序来配置代理程序

Windows 在集群环境中，可以使用集群实用程序将多个 Microsoft SQL Server 代理程序实例添加至集群组。

集群实用程序会自动将每个 Microsoft SQL Server 代理程序实例的代理程序服务和收集器服务作为通用服务资源添加至集群组。您可使用集群实用程序来完成下列任务:

- 将 SQL Server 代理程序实例添加至集群
- 更新集群中现有的 SQL Server 代理程序实例
- 从集群中移除 SQL Server 代理程序实例

使用集群实用程序的先决条件

您必须确保系统环境符合运行集群实用程序的先决条件。

确保满足下列先决条件:

- 在集群环境中至少有一个组的计算机上运行集群实用程序。
- 针对集群中的所有节点启动远程注册服务。
- 您必须具有集群管理员权限才能访问集群实用程序。
- 代理程序和收集器的服务名称必须在所有集群节点上都相同。

  例如，如果代理程序服务名称为 Monitoring Agent for Microsoft SQL Server: SQLTEST#INSTANCE1，收集器名称为 Monitoring Agent for Microsoft SQL Server: Collector SQLTEST#INSTANCE1，那么同一服务名称必须存在于集群的所有节点上。

将 Microsoft SQL Server 代理程序实例添加至集群

在集群环境中，可以使用集群实用程序将 Microsoft SQL Server 代理程序实例添加至集群组。

过程

1. 要运行该实用程序，请完成下列其中一个步骤:
   - 对于 64 位代理程序，请转至 *candle_home*\TMAITM6_x64 目录。
   - 对于 32 位代理程序，请转至 *candle_home*\TMAITM6 目录。
2. 要运行集群实用程序，请双击 KoqClusterUtility.exe。
3. 在**可用的 SQL Server 代理程序实例**区域中，选中 Microsoft SQL Server 代理程序实例，然后单击**添加**。
4. 在“**选择集群组名称**”窗口中，选择集群组。
   您选择的集群组必须是 Microsoft SQL Server 代理程序所监视的 SQL Server 实例。
5. 在“**选择共享位置的路径**”窗口中，浏览到存储代理程序日志及收集器日志的路径。
   如果您未选择路径，那么在缺省情况下，将会选择 CANDLEHOME/TMAITM6(_x64)/logs 位置来存储代理程序日志及收集器日志。
6. 要将 Microsoft SQL Server 代理程序实例添加到集群环境，请单击**确定**。
   集群实用程序的活动日志会显示在“**历史记录**”窗格中。

更新集群中现有的 Microsoft SQL Server 代理程序实例

您可使用集群实用程序来更新集群中 SQL Server 实例的代理程序日志及收集器日志的存储位置。

过程

1. 要更新现有的 Microsoft SQL Server 代理程序实例，请打开“**集群实用程序**”窗口。

2. 在**已配置的 SQL Server 代理程序实例**区域中，选中 Microsoft SQL Server 代理程序实例，然后单击**更新**。
3. 在“**设置共享位置的路径**”窗口中，浏览到存储代理程序日志及收集器日志的路径。
   如果您未选择路径，那么代理程序日志及收集器日志会存储于您在集群中添加 Microsoft SQL Server 代理程序实例时设置的位置。
4. 单击**确定**。
   集群实用程序的活动日志会显示在“**历史记录**”窗格中。

从集群中移除 Microsoft SQL Server 代理程序实例
您可使用集群实用程序从集群组中移除 Microsoft SQL Server 代理程序实例。

过程

1. 打开“**集群实用程序**”窗口。
2. 在**已配置的 SQL Server 代理程序实例**区域中，选中 Microsoft SQL Server 代理程序实例，然后单击**移除**。
3. 在**请确认操作**对话框中，单击**是**以从集群中删除该 Microsoft SQL Server 代理程序实例。
   集群实用程序的活动日志会显示在“**历史记录**”窗格中。

为 ERRORLOG 文件配置多个校对

Application Performance Management V8.1.4.0.4 的 Microsoft SQL Server 代理程序 V06.31.17.00 或更高版本支持 ERRORLOG 文件中的多个校对。您现在可以配置代理程序以解析**问题详细信息**属性组的 ERRORLOG 文件中的多个校对。请注意，ERRORLOG 文件中的多个校对不可应用于**错误事件详细信息**属性组。

开始之前

要配置代理程序的多个校对，请确保已安装代理程序。

关于此任务
缺省校对为英语。有关 SQL Server 的其他语言，代理程序将基于配置文件 `koqErrConfig.ini` 中的校对解析 ERRORLOG 文件。因此，必须添加 `koqErrConfig.ini` 文件中使用的校对。

过程

要为代理程序配置多个校对，请完成以下步骤：

1. 转至代理程序目录 *agent_directory*。

   Windows

   · 对于 64 位代理程序，*agent_directory* 是 *Agent_home*\TMAITM6_x64。

   · 对于 32 位代理程序，*agent_directory* 是 *Agent_home*\TMAITM6。

   Linux

   · 对于 64 位代理程序，*agent_directory* 是 *Agent_home*/TMAITM6_x64。

   其中，*Agent_home* 是代理程序安装目录。
2. 打开配置 `koqErrConfig.ini` 文件：
3. 移动到文件的末尾，添加新的校对。
   例如，要启用法语校对，在 `koqErrConfig.ini` 文件结尾处添加 **name-value** 对格式的下列校对设置。

```
[French]
Error = Erreur :
Severity = Gravité :
State = État :
```

注: *agent_directory*\koqErrConfigSample.ini 中提供了校对的样本列表。

其中 Windows

- 对于 64 位代理程序，*agent_directory* 是 *Agent_home*\TMAITM6_x64。
- 对于 32 位代理程序，*agent_directory* 是 *Agent_home*\TMAITM6。

Linux

- 对于 64 位代理程序，*agent_directory* 是 *Agent_home*/TMAITM6_x64。

其中，*Agent_home* 是代理程序安装目录。

如果 koqErrConfigSample.ini 中未提供目标校对，那么可以从 ERRORLOG 文件确定校对关键字值。在 koqErrConfig.ini 中配置校对设置时，请遵循以下校对格式。

```
[Section_name]
Error = Error_value
Severity = Severity_value
State = State_value
```

其中

- *Section_name* 是 SQL Server 校对名称。确保校对名称使用左方括号“**[**”和右方括号“**]**”括起来。
- *Error_value* 目标校对的 ERRORLOG 文件中找到的对应错误关键字。
- *Severity_value* 目标校对的 ERRORLOG 文件中找到的对应严重性关键字。
- *State_value* 目标校对的 ERRORLOG 文件中找到的对应状态关键字。

要点: 关键字值必须与 ERRORLOG 文件中找到的关键字值相同，包括特殊字符。

4. 保存配置 koqErrConfig.ini 文件。

   不需要重新启动代理程序。

   如果未提供配置文件 koqErrConfig.ini 或配置文件 koqErrConfig.ini 为空，那么 ERRORLOG 文件将显示缺省校对，因为英语错误消息的严重性级别大于缺省严重性级别（如有）。

   如果正确配置了配置文件 koqErrConfig.ini，那么 ERRORLOG 文件将显示严重性级别大于缺省严重性级别（如有）的对应错误消息。

   缺省值严重性级别为 17。

   注意: 代理程序升级期间 koqErrConfig.ini 文件中所作的更改未保留，您必须在执行代理程序升级前进行备份。

下一步做什么

检查 Application Performance Management 仪表板上的**错误日志警报**窗口小部件或**问题详细信息**属性组，作为校对设置的结果。

配置 MongoDB 监视

Monitoring Agent for MongoDB 需要实例名称。您必须手动配置和启动代理程序实例。MongoDB 代理程序支持本地及远程监视。请参阅配置 MongoDB 代理程序以进行远程及本地监视的下列先决条件。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 MongoDB 代理程序 的软件产品兼容性报告 (SPCR)。
- 确保配置 MongoDB 代理程序的用户具有所需的角色，以收集所有属性的数据。
  - 要在 MongoDB 数据库 V2.4 和 V2.6 上配置此代理程序，必须向用户分配 clusterAdmin、readAnyDatabase 和 dbAdminAnyDatabase 角色

– 要在 MongoDB 数据库 V3.x 和 V4.x 上配置此代理程序，必须向用户分配 clusterMonitor、readAnyDatabase 和 dbAdminAnyDatabase 角色

要了解需要这些用户角色的属性组的信息，请参阅第 492 页的表 183。

· 使用现有用户，或在管理员数据库中创建用户。

要点: 在创建用户并向该用户授予所需角色前，您必须连接到 MongoDB 数据库并将该数据库更改为管理员数据库。如果 mongod 或 mongos 过程正在认证方式下运行，请输入所需凭证以连接到 MongoDB 数据库。

1. 运行以下命令以连接到 MongoDB 数据库：

   `mongo IP:port`

   其中

   – *IP* 是 mongod 或 mongos 过程的 IP 地址
   – *port* 是 mongod 或 mongos 过程的端口号

2. 将该数据库更改为管理员数据库：

   `use admin`

3. 运行以下命令之一以在 MongoDB 管理员数据库中添加用户并向该用户分配所需角色：

   – 对于 MongoDB 数据库 V2.4，运行以下命令：

   ```
   db.addUser({ user: “username”, pwd: “password”, roles:
   [ 'clusterAdmin', 'readAnyDatabase', 'dbAdminAnyDatabase' ] })
   ```

   – 对于 MongoDB 数据库 V2.6，运行以下命令：

   ```
   db.createUser({user: “username”, pwd: “password”, roles:
   [ 'clusterAdmin', 'readAnyDatabase', 'dbAdminAnyDatabase' ] })
   ```

   – 对于 MongoDB 数据库 V3.x 和 4.x，运行以下命令：

   ```
   db.createUser({user: “username”, pwd: “password”, roles:
   [ 'clusterMonitor', 'readAnyDatabase', 'dbAdminAnyDatabase' ] })
   ```

4. 运行以下命令以验证用户是否添加到管理员数据库：

   `db.auth(“username”, “password”)`

   返回码 **1** 指示已添加用户，而返回码 **0** 指示用户添加失败。

下表包含有关用户角色以及需要这些用户角色的属性的信息：

表 *183.* 属性组及其所需用户角色

角色	**MongoDB 数据库版本**	**属性组**
dbAdminAnyDatabase	2.x、3.x 和 4.x	响应时间
readAnyDatabase	2.x、3.x 和 4.x	· Mongod 列表 · 常规共享信息 · 集合存储 · 数据库名称 · 共享详细信息 · 集合存储详细信息

表 183. 属性组及其所需用户角色 (续)

角色	MongoDB 数据库版本	属性组
clusterAdmin	2.x、3.x 和 4.x	· Mongo 实例信息 · Mongo 实例 IO 信息 · APMUI One 的 MII 副本 · APMUI Two 的 MII 副本 · Mongo 实例 DB 锁定 · 锁定数 · MongoDB 锁定数 · WiredTiger 详细信息 · MMAPv1 详细信息
clusterMonitor	2.x、3.x 和 4.x	· Mongo 实例信息 · Mongo 实例 IO 信息 · APMUI One 的 MII 副本 · APMUI Two 的 MII 副本 · Mongo 实例 DB 锁定 · 锁定数 · MongoDB 锁定数 · WiredTiger 详细信息 · MMAPv1 详细信息

· 要对 MongoDB 服务器进行远程监视，请参阅以下两项先决条件：

1. 由于 MongoDB 代理程序需要 mongo shell 来从 MongoDB 服务器远程收集信息，因此安装并配置了 MongoDB 代理程序的系统必须具有 MongoDB 服务器实例。代理机器上的 MongoDB 服务器的 mongo shell 用于连接到远程 MongoDB 服务器以进行监视。
2. 在托管该代理程序的系统上的 `/etc/hosts` 文件中，存在远程机器的条目。

关于此任务

受管系统名称包含您指定的实例名称。例如，可以将实例名称指定为 *instance_name*:*host_name*:*pc*，其中，*pc* 是代理程序的双字符产品代码。受管系统名称至多可以包含 32 个字符。实例名称最多包含 28 个字符，不包括主机名的长度。例如，如果指定 Mongo2 作为实例名称，那么受管系统名称为 `Mongo2:hostname:KJ`。

要点: 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法完整显示。

产品版本和代理程序版本通常不同。以下指南针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。

切记:

· 对于成功收集数据的代理程序，启动具有超级 (root) 用户的代理程序，或使用同一用户标识启动代理程序和 mongod 进程。

· 在 MongoDB 作为集群运行的环境中，确保在运行路由器进程的同一计算机上安装该代理程序。在具有该计算机的 IP 地址和端口号的同一计算机上配置该代理程序，并将 **TYPE** 设置为 1。

· 在 MongoDB 作为集群以认证方式运行的环境中，请确保在集群中所有 shard 上添加具有所需权利的同一用户标识。

您可通过使用缺省设置、通过编辑静默响应文件或通过响应提示来配置代理程序。

使用缺省设置配置代理程序

对于典型环境，请使用缺省设置来配置代理程序。将缺省设置用于代理程序配置时，代理程序不会以认证方式运行。

过程

1. 运行以下命令：
   install_dir/bin/mongodb-agent.sh config *instance_name install_dir*/samples/mongodb_silent_config.txt

   其中

   · *instance_name* 是针对唯一应用程序实例指定的名称。

   · *install_dir* 是 MongoDB 代理程序的安装目录。

   缺省安装目录为 /opt/ibm/apm/agent。
2. 运行以下命令以启动代理程序：
   install_dir/bin/mongodb-agent.sh start *instance_name*

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数，其中某些参数定义有缺省值。您可以编辑静默响应文件来为配置参数指定其他值，并配置代理程序。

开始之前

要以认证方式运行 MongoDB 数据库，请确保以对 MongoDB 数据库具有 clusterAdmin、readAnyDatabase 和 dbAdminAnyDatabase 角色的用户身份配置代理程序。

过程

1. 在文本编辑器中，打开位于以下路径的静默响应文件：
   install_dir/samples/mongodb_silent_config.txt。
2. 对于 **TYPE** 参数，输入以下其中一个值：

   · 1（对于集群）

   · 2（对于复制集）

   · 3（对于单独实例）

   缺省情况下，代理程序监视集群。
3. 对于**端口**参数，请指定 MongoDB 集群路由器或所监视的 MongoDB 复制集 mongod 实例的端口号。

   切记: 如果未指定任何端口号，那么代理程序自动发现在缺省接口上激活的相应 MongoDB 流程的端口号。如果在缺省接口上无活动的 MongoDB 流程，那么代理程序选择辅助接口上活动的相应 MongoDB 流程的端口号。
4. 对于 **HOST** 参数，指定 MongoDB 主机系统的 IP 地址。

   切记: 如果未指定任何 IP 地址，那么代理程序自动检测在缺省接口上活动的相应 MongoDB 流程的 IP 地址。如果缺省接口上无活动的 MongoDB 流程，那么代理程序检测在辅助接口上活动的相应 MongoDB 流程的 IP 地址。
5. 对于**认证**参数，指定是 以包含以认证方式运行的 mongoDB。缺省值为 NO，表示代理程序并非以认证方式运行。

切记: 当 MongoDB 数据库在认证方式下运行时，在没有凭证的情况下，MongoDB 代理程序 或任何 MongoDB 客户机无法连接到 MongoDB 数据库。要连接到在认证方式下运行的数据库，请为 **AUTHENTICATION** 参数指定 YES。

如果指定是，请完成以下步骤:

a) 对于 **User Name** 参数，指定路由器或 mongod 实例的用户名。请确保至少为用户指定最低限度的角色。有关用户角色的信息，请参阅第 492 页的表 183。

b) 对于**密码**参数，请指定密码。

6. 保存并关闭 mongodb_silent_config.txt 文件，然后运行以下命令:
   install_dir/bin/mongodb-agent.sh config *instance_name* *install_dir*/samples/mongodb_silent_config.txt

   其中

   · *instance_name* 是为实例指定的名称。

   · *install_dir* 是 MongoDB 代理程序的安装目录。
7. 运行以下命令以启动代理程序:
   install_dir/bin/mongodb-agent.sh start *instance_name*

   要点: 如果将代理程序升级到 V1.0.0.9 或更高版本，并想要以认证方式运行代理程序，必须再次配置代理程序以提供用户名和密码。要收集数据，必须在配置后停止并重新启动代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

通过对提示做出响应来配置代理程序

要使用定制设置配置代理程序，可在运行脚本时，在系统提醒的情况下为配置参数指定值。

过程

1. 运行以下命令:
   install_dir/bin/mongodb-agent.sh config *instance_name*

   其中

   · *instance_name* 是为实例指定的名称。

   · *install_dir* 是 MongoDB 代理程序的安装目录。
2. 当系统提示您提供 **TYPE** 参数的值时，可按 Enter 键以接受缺省值，或指定以下某个值，然后按 Enter 键:

   · 1（对于集群）

   · 2（对于复制集）

   · 3（对于单独实例）

   缺省情况下，代理程序监视集群。
3. 当系统提示您提供 **PORT** 参数的值时，按 Enter 键以接受缺省值，或为正在监视的 MongoDB 复制集的 MongoDB 集群或 mongod 实例指定路由器的端口号，然后按 Enter 键。

   切记: 如果未指定任何端口号，那么代理程序自动发现在缺省接口上激活的相应 MongoDB 流程的端口号。如果在缺省接口上无活动的 MongoDB 流程，那么代理程序选择辅助接口上活动的相应 MongoDB 流程的端口号。
4. 当系统提示您提供 **HOST** 参数的值时，按 Enter 键以接受缺省值，或指定 MongoDB 主机系统的 IP 地址，然后按 Enter 键。

切记: 如果未指定任何 IP 地址，那么代理程序自动检测在缺省接口上活动的相应 MongoDB 流程的 IP 地址。如果缺省接口上无活动的 MongoDB 流程，那么代理程序检测在辅助接口上活动的相应 MongoDB 流程的 IP 地址。

5. 当系统提示您提供 **AUTHENTICATION** 参数的值时，按 Enter 键以接受缺省值，或指定代理程序是否在认证方式下运行。

   缺省值为 NO，表示代理程序并非以认证方式运行。指定 YES 以指示 mongoDB 是否在认证方式下运行。

   切记: 当 MongoDB 数据库在认证方式下运行时，在没有凭证的情况下，MongoDB 代理程序 或任何 MongoDB 客户机无法连接到 MongoDB 数据库。要连接到在认证方式下运行的数据库，请为 **AUTHENTICATION** 参数指定 YES。

   如果指定是，请完成以下步骤:

   a) 对于 **User Name** 参数，指定路由器或 mongod 实例的用户名。请确保至少为用户指定最低限度的角色。有关用户角色的信息，请参阅第 492 页的表 183。

   b) 对于**密码**参数，请指定密码。

6. 运行以下命令以启动代理程序:
   install_dir/bin/mongodb-agent.sh start *instance_name*

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

配置 MySQL 监视

Monitoring Agent for MySQL 需要实例名称和 MySQL 服务器用户凭证。创建第一个代理程序实例后，可更改配置设置。

开始之前

- 确保在 MySQL 数据库中创建了用户以运行代理程序。该用户不需要对要监视的 MySQL 数据库具有任何特定特权。
- 查看必备硬件和软件。有关最新系统需求信息，请参阅 MySQL 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。受管系统名称至多可以包含 32 个字符。指定的实例名称至多可包含 28 个字符（不包括主机名长度）。例如，如果指定 MySQL2 作为实例名称，那么受管系统名称为 MySQL2:hostname:SE。

要点: 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法完整显示。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，完成以下步骤:
   a) 双击 **Monitoring Agent for MySQL** 模板。
   b) 在 **Monitoring Agent for MySQL** 窗口中，指定实例名称并单击**确定**。

3. 在 **Monitoring Agent for MySQL** 窗口中，完成以下步骤：
   a) 在 **IP 地址**字段中，输入要远程监视的 MySQL 服务器的 IP 地址。如果在要监视的服务器上安装代理程序，那么保留缺省值。
   b) 在 **JDBC 用户名**字段中，输入 MySQL 服务器用户的名称。缺省值为 root。
   c) 在 **JDBC 密码**字段中，输入 JDBC 用户的密码。
   d) 在**确认 JDBC 密码**字段中，再次输入密码。
   e) 在 **JDBC JAR 文件**字段中，单击**浏览**并找到包含 MySQL 连接器 Java 文件的目录，然后将其选中。
   f) 单击**下一步**。
   g) 在 **JDBC 端口号**字段中，指定 JDBC 服务器的端口号。
   缺省端口号为 3306。
   h) 从 **Java 跟踪级别**列表中，选择 Java 的跟踪级别。
   缺省值为 Error。
   i) 单击**确定**。
   该实例将显示在 **IBM Performance Management** 窗口中。
4. 右键单击 **Monitoring Agent for MySQL** 实例，然后单击**启动**。

   切记: 要重新配置代理程序，请在 **IBM Performance Management** 窗口中完成以下步骤：
   a. 停止想要配置的代理程序实例。
   b. 右键单击 **Monitoring Agent for MySQL** 实例，然后单击**重新配置**。
   c. 重复步骤 3 和 4。

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Linux 系统上配置代理程序

运行配置脚本在 Linux 系统上配置代理程序。

过程

1. 运行以下命令：

```
install_dir/bin/mysql-agent.sh config instance_name
```

   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 MySQL 代理程序的安装目录。
2. 当提示您输入以下参数的值时，请按 Enter 键以接受缺省值或指定另一个值然后按 Enter 键。
   · IP 地址
   · JDBC 用户名
   · JDBC 密码
   · 重新输入：JDBC 密码
   · JDBC Jar 文件
   · JDBC 端口号（缺省端口号为 3306。）
   · Java 跟踪级别（缺省值为 Error。）

   有关配置参数的信息，请参阅第 498 页的『使用静默响应文件来配置代理程序』。
3. 运行以下命令以启动代理程序。

```
install_dir/bin/mysql-agent.sh start instance_name
```

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置代理程序

使用静默响应文件配置代理程序，而无需对运行配置脚本时的提示进行响应。您可以使用静默响应文件在 Windows 和 Linux 系统上配置代理程序。

关于此任务

静默响应文件包含配置参数。您可以编辑响应文件中的参数值，并运行配置脚本来创建代理程序实例和更新配置值。

过程

1. 在文本编辑器中，打开位于以下路径的响应文件：

   Linux *install_dir*/samples/mysql_silent_config.txt

   Windows *install_dir*\samples\mysql_silent_config.txt

   其中，*install_dir* 是 MySQL 代理程序的安装目录。
2. 在响应文件中，对下列参数指定值：
   - 对于 **Server Name** 参数，请指定要远程监视的 MySQL 服务器的 IP 地址。否则，保留缺省值为 localhost。
   - 对于 **JDBC user name** 参数，保留缺省用户名 root 值，或者指定具有 INFORMATION_SCHEMA 表的查看权限的用户的名称。
   - 对于 **JDBC password** 参数，输入 JDBC 用户密码。
   - 对于 **JDBC Jar File** 参数，请保留缺省路径（如果此路径为 Java jar 文件的 MySQL 接口的路径）。否则，请输入正确的路径。接口位于下列缺省路径中：

     Linux /usr/share/java/mysql-connector-java.jar

     Windows C:\Program Files (x86)\MySQL\Connector J 5.1.26\mysql-connector-java-5.1.26-bin.jar
   - 对于 **JDBC port number** 参数，请保留缺省端口号 3306 或指定另一个端口号。
   - 对于 **Java trace level** 参数，请保留缺省值 Error 或根据 IBM 支持指示信息指定另一个级别。
3. 保存并关闭响应文件，然后运行以下命令以更新代理程序配置设置：

   Linux *install_dir*/bin/mysql-agent.sh config *instance_name install_dir*/samples/mysql_silent_config.txt

   Windows *install_dir*\BIN\mysql-agent.bat config *instance_name install_dir*\samples\mysql_silent_config.txt

   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 MySQL 代理程序的安装目录。

   要点: 请确保包含指向静默响应文件的绝对路径。否则，仪表板中不会显示任何代理程序数据。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

配置 NetApp Storage 监视

Monitoring Agent for NetApp Storage 使用 NetApp OnCommand Unified Manager、OnCommand API Services 和 OnCommand Performance Manager 来监视 NetApp 存储系统。

开始之前

- 查看必备硬件和软件。有关最新系统需求信息，请参阅 NetApp Storage 代理程序 的软件产品兼容性报告 (SPCR)。
- 确保在机器上安装下列组件：
  - OnCommand Unified Manager
  - OnCommand Performance Manager
  - OnCommand API Services

  有关安装这些组件的信息，请参阅 NetApp 文档。
- 确保 OnCommand API Services、OnCommand Unified Manager 和 OnCommand Performance Manager 的版本兼容。例如，要配置 OnCommand API Services V1.0，请将 OnCommand Unified Manager V6.2、V6.1 或 V6.0 与 OnCommand Performance Manager V1.1 配对。要了解兼容的产品版本，请参阅互操作性矩阵工具。
- 请确保连接至 OnCommand Unified Manager 的用户具有所要监视的 NetApp 存储系统的 GlobalRead 特权。请使用具有此特权的现有用户标识，或创建新的用户标识。有关在 NetApp 存储系统中创建用户标识的信息，请参阅 NetApp 文档。
- 请确保用来配置 OnCommand API Services 的用户是管理员或监视者。这些用户类型具有运行 REST API 所需的缺省许可权。
- 完成第 499 页的『下载和安装 NetApp 可管理性 SDK JAR 文件』所述的步骤，从 NetApp Web 站点下载 NetApp 可管理性 SDK JAR 文件 (`manageontap.jar`)，并将此文件安装在监视代理程序的 `lib` 目录中。

关于此任务

NetApp Storage 代理程序是多实例代理程序。必须创建第一个实例，然后手动启动代理程序。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅第 44 页的『变更历史记录』。

- 要在 Windows 系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。
- 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

下载和安装 NetApp 可管理性 SDK JAR 文件

NetApp Storage 代理程序需要 NetApp 可管理性 SDK JAR 文件，以便与 NetApp OCUM 服务器通信。

关于此任务

安装 NetApp Storage 代理程序之后，请从 NetApp Web 站点下载 NetApp 可管理性 SDK JAR 文件 (`manageontap.jar`)，并将此文件安装在监视代理程序的 `lib` 目录中。

过程

1. 从以下 Web 站点下载包含这个 JAR 文件的压缩文件：http://communities.netapp.com/docs/DOC-1152。
2. 将这个压缩文件解压缩，并将 `manageontap.jar` 文件复制到下列位置：
   - 对于 32 位 Windows 系统，请将此文件复制到 *`install_dir`*`/tmaitm6`
   - 对于 64 位 Windows 系统，请将此文件复制到 *`install_dir`*`/tmaitm6_x64`
   - 对于 32 位 Linux 系统，请将此文件复制到 *`install_dir`*`/li6263/nu/lib`
   - 对于 64 位 x86-64 Linux 系统，请将此文件复制到 *`install_dir`*`/lx8266/nu/lib`
   - 对于 64 位 zLinux 系统，请将此文件复制到 *`install_dir`*`/ls3266/nu/lib`

下一步做什么
完成代理程序配置。

在 Windows 系统上配置代理程序

可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

关于此任务

NetApp Storage 代理程序会提供一些参数的缺省值。您可以为这些参数指定不同的值。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for NetApp Storage**，然后单击**配置代理程序**。

   切记: 首次配置代理程序之后，**配置代理进程**选项将处于禁用状态。要再次配置代理程序，单击**重新配置**。
3. 在 Monitoring Agent for NetApp Storage 窗口中，完成下列步骤：
   a) 为 NetApp Storage 代理程序实例输入唯一名称，然后单击**确定**。
   b) 在**数据提供程序**选项卡上，指定配置参数的值，然后单击**下一步**。
   c) 在 **OnCommand Unified Manager** 选项卡上，指定配置参数的值，然后单击**下一步**。
   d) 在 **OnCommand API Service** 选项卡上，指定配置参数的值，然后单击**确定**。

   有关“Monitoring Agent for NetApp Storage”窗口的每个选项卡中配置参数的信息，请参阅以下主题：
   · 第 502 页的『数据提供程序的配置参数』
   · 第 503 页的『OnCommand Unified Manager 的配置参数』
   · 第 503 页的『OnCommand API Service 的配置参数』
4. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for NetApp Storage**，然后单击**启动**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

· 要以静默方式配置 NetApp Storage 代理程序，请完成下列步骤：
  a) 在文本编辑器中，打开位于以下路径的 netapp_storage_silent_config.txt 文件：
  – Linux *install_dir*/samples/netapp_storage_silent_config.txt

    示例：/opt/ibm/apm/agent/samples/netapp_storage_silent_config.txt

– Windows *install_dir*\samples\netapp_storage_silent_config.txt

示例：C:\IBM\APM\samples\netapp_storage_silent_config.txt

b) 在 netapp_storage_silent_config.txt 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

有关配置参数的信息，请参阅下列主题：

– 第 502 页的『数据提供程序的配置参数』
– 第 503 页的『OnCommand Unified Manager 的配置参数』
– 第 503 页的『OnCommand API Service 的配置参数』

c) 保存并关闭 netapp_storage_silent_config.txt 文件，然后运行下列命令：

– Linux *install_dir*/bin/netapp_storage-agent.sh config *instance_name* *install_dir*/samples/netapp_storage_silent_config.txt

示例：**/opt/ibm/apm/agent/bin/netapp_storage-agent.sh config instance_name /opt/ibm/apm/agent/samples/netapp_storage_silent_config.txt**

– Windows *install_dir*\bin\netapp_storage-agent.bat config *instance_name* *install_dir*\samples\netapp_storage_silent_config.txt

示例：**C:\IBM\APM\bin\netapp_storage-agent.bat config instance_name C:\IBM\APM\samples\netapp_storage_silent_config.txt**

其中

instance_name
要对实例指定的名称。

install_dir
代理程序的安装路径。

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

d) 运行以下命令以启动代理程序：

– Linux *install_dir*/bin/netapp_storage-agent.sh start *instance_name*

示例：**/opt/ibm/apm/agent/bin/netapp_storage-agent.sh start instance_name**

– Windows *install_dir*\bin\netapp_storage-agent.bat start *instance_name*

示例：**C:\IBM\APM\bin\netapp_storage-agent.bat start instance_name**

下一步做什么
登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

通过对提示做出响应来配置代理程序

要在 Linux 系统上配置代理程序，必须运行脚本并对提示做出响应。

过程

1. 在命令行上，输入以下命令：

   install_dir/bin/netapp_storage-agent.sh config *instance_name*

   示例：**/opt/ibm/apm/agent/bin/netapp_storage-agent.sh config instance_name**

   其中

instance_name
要对实例指定的名称。

install_dir
代理程序的安装路径。

2. 请参阅下列主题以响应提示：
   - 第 502 页的『数据提供程序的配置参数』
   - 第 503 页的『OnCommand Unified Manager 的配置参数』
   - 第 503 页的『OnCommand API Service 的配置参数』
3. 运行以下命令以启动代理程序：

   install_dir/bin/netapp_storage-agent.sh start *instance_name*

   示例：**/opt/ibm/apm/agent/bin/netapp_storage-agent.sh start instance_name**

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

数据提供程序的配置参数

在配置 NetApp Storage 代理程序时，可以更改数据提供程序的缺省参数值，例如最大数据提供程序日志文件数量、最大日志文件大小以及日志文件中包含的详细信息级别。

下表包含数据提供程序的配置参数的详细描述。

表 *184.* 数据提供程序的配置参数的名称和描述

参数名称	描述	必填字段
实例名称 (**KNU_INSTANCE_NAME**)	实例的名称。 **限制:** “实例名称”字段显示您首次配置代理程序时指定的实例名称。重新配置代理程序时，无法更改代理程序的实例名称。	是
最大数据提供程序日志文件数 (**KNU_LOG_FILE_MAX_COUNT**)	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。	是
每个数据提供程序日志的最大大小 (KB) (**KNU_LOG_FILE_MAX_SIZE**)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。	是

表 *184.* 数据提供程序的配置参数的名称和描述 *(续)*		
参数名称	**描述**	**必填字段**
数据提供程序日志中的详细信息级别 (**KNU_LOG_LEVEL**)	可以包含在数据提供程序创建的日志文件中的详细信息级别。缺省值是 4 (Info)。以下值有效： · 1 (Off)：不记录任何消息。 · 2 (Severe)：仅记录错误。 · 3 (Warning)：在 Severe 级别所记录的所有错误和消息，以及可能会导致不良行为的潜在错误。 · 4 (Info)：在 Warning 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的高级别参考消息。 · 5 (Fine)：在 Info 级别所记录的所有错误和消息，以及在数据提供程序处理时描述其状态的低级别参考消息。 · 6 (Finer)：在 Fine 级别所记录的所有错误和消息，以及高度详细的参考消息，例如性能概要分析信息和调试数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 7 (Finest)：在 Fine 级别所记录的所有错误和消息，以及最详细的参考消息，包括低级别编程消息和数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 8 (All)：记录所有错误和消息。	是

OnCommand Unified Manager 的配置参数

配置 NetApp Storage 代理程序时，可以更改 OnCommand Unified Manager (OCUM) 参数的缺省值，例如 OCUM 服务器 IP 地址、用户名和密码。

下表包含数据源的配置参数的详细描述。

表 *185. OnCommand Unified Manager* 的配置参数的名称和描述		
参数名称	**描述**	**必填字段**
服务器 (**KNU_DATASOURCE_HOST_ADDRESS**)	所要监视的 NetApp OCUM 服务器的主机名或 IP 地址。	是
用户 (**KNU_DATASOURCE_USERNAME**)	NetApp OCUM 服务器上具有足够特权来收集数据的用户名。缺省值为 admin。	是
密码 (**KNU_DATASOURCE_PASSWORD**)	**用户**参数中指定的用户的密码。	是
确认密码	**输入密码**参数中指定的密码。	是
协议 (**KNU_DATASOURCE_PROTOCOL**)	用于与 NetApp OCUM 服务器进行通信的协议。缺省值为 HTTPS。	是

OnCommand API Service 的配置参数

在配置 NetApp Storage 代理程序时，您可以更改 OnCommand API Service 参数的缺省值，例如主机地址、用户名和密码。

下表包含数据源的配置参数的详细描述。

表 186. *OnCommand API Service* 的配置参数的名称和描述		
参数名称	**描述**	**必填字段**
主机地址 (**KNU_API_SERVICES_HOST_ADDRESS**)	OnCommand API Service 的主机名或 IP 地址。	是
用户 (**KNU_API_SERVICES_USERNAME**)	具有足够特权来连接至 OnCommand API Service 的用户名。缺省值为 admin。	是
密码 (**KNU_API_SERVICES_PASSWORD**)	**用户**参数中指定的用户的密码。	
确认密码	**输入密码**参数中指定的密码。	是

配置 Node.js 监视

您可以使用 Node.js 代理程序或独立 Node.js 数据收集器来监视 Node.js 应用程序。如果需要更简单的安装过程和事务跟踪功能，请使用 Node.js 数据收集器。

开始之前

- 此处的方向针对的是此代理程序和数据收集器的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。
- 确保在您的环境中满足 Node.js 代理程序或 Node.js 数据收集器的系统需求。
  - 有关 Node.js 代理程序的最新系统需求的信息，请参阅 Node.js 代理程序 的软件产品兼容性报告 (SPCR)。
  - 有关 Node.js 数据收集器的最新系统需求的信息，请参阅 Node.js 数据收集器 的软件产品兼容性报告 (SPCR)。

关于此任务

以下过程是配置 Monitoring Agent for Node.js 和独立的 Node.js 数据收集器的路线图，其中包括必需步骤和可选步骤。

- 要监视本地应用程序，您可以配置独立的 Node.js 数据收集器或 Node.js 代理程序。如果要为 Node.js 应用程序启用事务跟踪，请配置独立的数据收集器。
- 要监视 IBM Cloud（原称为 Bluemix）或 Kubernetes 应用程序，请配置独立 Node.js 数据收集器。

请根据您的需要，完成以下步骤。

过程

- 配置 Node.js 代理程序以监视您的本地应用程序。
  a) 将一个代理程序数据收集器添加到 Node.js 应用程序，以使代理程序正常工作。请参阅第 505 页的『配置 Node.js 代理程序』。
  b) 可选： 要更改代理程序的监视行为，请参阅配置 Node.js 代理程序 数据收集器。
  c) 可选： 要配置诊断数据收集及显示，请参阅配置诊断数据收集器。
- 配置独立 Node.js 数据收集器以监视 IBM Cloud 应用程序。
  a) 要配置独立 Node.js 数据收集器，请参阅第 509 页的『针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器』。

b) 要更改独立 Node.js 数据收集器的行为，请参阅第 511 页的『针对 IBM Cloud 应用程序定制独立 Node.js 数据收集器』。

· 配置独立 Node.js 数据收集器以监视本地应用程序。

a) 要配置独立 Node.js 数据收集器，请参阅第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』。

b) 要更改独立 Node.js 数据收集器的行为，请参阅第 515 页的『为本地应用程序定制 Node.js 数据收集器』。

· 配置独立 Node.js 数据收集器以监视 Kubernetes 上的应用程序。

a) 要配置独立 Node.js 数据收集器，请参阅第 517 页的『为 Kubernetes 应用程序配置独立 Node.js 数据收集器』。

b) 要更改独立 Node.js 数据收集器的行为，请参阅第 518 页的『为 Kubernetes 应用程序定制独立 Node.js 数据收集器』。

配置 Node.js 代理程序

您必须先向 Node.js 应用程序添加数据收集器，并将其重新启动，代理程序才能开始监视应用程序。

开始之前

在同一版本内重新配置代理程序设置之前，使用以下步骤清除先前配置已创建的数据收集器文件：

1. 转至 *install_dir*/lx8266/nj/bin 目录。
2. 运行 ./uninstall.sh 命令，以移除现有的数据收集器文件。

关于此任务

Node.js 代理程序是单实例代理程序。它为每个受监视 Node.js 应用程序注册子节点。子节点的结构如下：

NJ:hostname_port:NJA

提示: 如果一个 Node.js 应用程序侦听多个端口号，那么会使用最小端口号。

您必须先向 Node.js 应用程序添加数据收集器，并将其重新启动，代理程序才能开始监视应用程序。代理程序数据收集器将收集转发至 Node.js 代理程序的数据。目前，提供了下列代理程序数据收集器：

· 资源数据收集器从 Node.js 应用程序收集资源监视数据。

· 诊断数据收集器从 Node.js 应用程序收集诊断数据和资源监视数据。

· 方法跟踪数据收集器从 Node.js 应用程序收集方法跟踪、诊断数据和资源监视数据。

过程

1. 确保用于运行应用程序服务器的用户标识对代理程序的 install_dir 目录具有完全许可权。
2. 移至 *install_dir*/bin 目录并运行以下命令：

```
./nodejs-agent.sh config
```

3. 遵循提示，以指定下列配置选项的值：

KNJ_NODEJS_RUNTIME_BIN_LOCATION
Node.js 运行时的 bin 文件夹目录。缺省目录为 /usr/local/bin。

KNJ_NPM_RUNTIME_BIN_LOCATION
npm 命令的 bin 文件夹的目录。缺省目录为 /usr/local/bin。

KNJ_NPM_LIB_LOCATION
npm 包全局安装目录的 lib 文件夹目录。缺省目录为 /usr/local/lib。例如，如果通过运行 npm install -g 命令来安装 npm 包，那么这个包将安装到 /*nodejs_home*/lib/node_modules 中，并且 **KNJ_NPM_LIB_LOCATION** 为 /*nodejs_home*/lib。

CP_PORT
代理程序侦听来自套接字客户端的数据时所用的端口。值为 0 表示将使用临时端口。缺省值为 63336。

注: 请勿使用系统中已使用的端口号。要检查该端口是否已在使用中，请运行 `netstat -apn | grep port_number` 命令。

4. 运行以下命令启动代理程序:

```
./nodejs-agent.sh start
```

5. 确认 Node.js 代理程序 顺利启动。如果此代理程序顺利启动，那么将生成 *KNJ_NPM_LIB_LOCATION*/node_modules/ibmapm 文件夹。
6. 根据您拥有的产品和需求，在 Node.js 应用程序的 .js 文件中插入下列其中一个条目以配置代理程序数据收集器:

   注: 只能将一个条目添加到 Node.js 应用程序，以启用代理程序数据收集器功能。并且，如果启用未包括在产品中的功能，那么可能会产生不必要的额外开销，这会降低应用程序执行效率。

   - 如果只有资源监视功能，那么可添加资源数据收集器。要添加资源数据收集器，请在 Node.js 应用程序文件开头插入下面这一行:

     ```
     require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm');
     ```

     如果环境中 **KNJ_NPM_LIB_LOCATION** 的值是 /usr/local/lib，那么此行是:

     ```
     require('/usr/local/lib/node_modules/ibmapm');
     ```

   - 如果不仅有资源层次的监视功能，还有诊断功能，那么可选择添加下列其中一个代理程序数据收集器:
     - 要添加方法跟踪数据收集器，请在 Node.js 应用程序文件开头插入下面这一行:

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm/methodtrace.js');
       ```

     - 要添加诊断数据收集器，请在 Node.js 应用程序文件开头插入下面这一行:

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm/deepdive.js');
       ```

     - 要添加资源监视数据收集器，请在 Node.js 应用程序文件开头插入下面这一行:

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibmapm');
       ```

     为了确保最佳性能，仅当要进行调试时，才应添加方法跟踪数据收集器。

   注: 插件代码从 Cloud APM2017 年 3 月开始更改。如果您从旧版升级代理程序，那么必须更新应用程序中现有数据收集器的代码，以使监视功能正常工作。
7. 重新启动 Node.js 应用程序，以启用代理程序数据收集器。

结果
您已成功配置 Node.js 代理程序。

下一步做什么
- 现在，您可验证 Node.js 代理程序 数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

  要点: 要在 Cloud APM 控制台中添加应用程序，请在应用程序编辑器中选择 **Node.js**。
- 您可以更改 Node.js 代理程序数据收集器的运行时行为。有关更多信息，请参阅配置 Node.js 代理程序数据收集器。
- 您可以配置诊断数据收集器，以启用诊断数据收集及显示。有关更多信息，请参阅配置诊断数据收集器。

配置 Node.js 代理程序数据收集器

通过在各 Node.js 代理程序数据收集器的配置文件中更改其运行时配置，可以更改其行为。

运行时配置文件

Node.js 数据收集器代码位于下列目录：

KNJ_NPM_LIB_LOCATION/node_modules/ibmapm

其中，*KNJ_NPM_LIB_LOCATION* 为 npm 包全局安装目录的 lib 文件夹目录。缺省目录为 /usr/local/lib。

在同一文件夹中，还有每一个代理程序数据收集器的运行时配置文件。代理程序数据收集器每分钟读取一次配置文件。

提示: 运行时配置文件以如下格式命名：

plugin_*application port number*_conf.json

更改配置文件的内容时，相关联代理程序数据收集器的行为会更改。在配置文件中，有两类可更改的信息：

- URL 过滤规则
- 代理程序数据收集器日志记录参数

URL 过滤规则

您可以更改运行时配置文件中的 URL 过滤规则。将使用正则表达式将 URL 路径名映射到用户定制的路径名。您可将 URL 映射到定制路径名，以满足下列需求：

- 聚集路径相似的 URL。例如，您有下列 URL 路径：

```
/demo/poll/1
/demo/poll/2
/demo/poll/3
...
```

  在 Web 服务器上，针对这些路径的请求可能是由公共例程来处理，因此您可使用以下示例所示的过滤器，将这些路径聚集到单一 URL 类型。

```
"filters":
[
    {
        "pattern": "/demo/poll/.+",
        "to": "/demo/poll/"
    }
```

  此过滤器会将针对诸如 "/demo/poll/xxx" 之类的 URL 路径的所有请求映射到 URL 路径类型 "/demo/poll"。然后，针对此类 URL 路径的所有请求的响应时间会取平均值为单个值。以这种方式汇总可帮助您更高效地利用可用资源。

- 忽略静态文件的 URL 路径，或过滤掉特定类型的请求。例如，如果某个 Web 页面包括会生成单独的服务器下载请求的图像，您可能不想查看这些类型的请求的响应时间。

  要过滤掉某种类型的请求，请将 "to" 值设置为空，格式如以下示例所示：

```
"filters":
[
    {
        "pattern": "GET /css/.+\\.css$",
        "to": ""
    }
```

  此过滤器会导致忽略用于获取 .css 文件的请求。因此，您可将可用的资源更有效率地用于需要监视的请求。

在配置文件中，URL 过滤规则是在名为 filters 的 JSON 数组中提供：

```
"filters":
[
```

```
    {
        "pattern": ".+\\.png$",
        "to": ""
    },
    {
        "pattern": ".+\\.jpg$",
        "to": ""
    },
    {
        "pattern": "GET /js/.+\\.js$",
        "to": ""
    },
    {
        "pattern": "GET /css/.+\\.css$",
        "to": ""
    }
]
```

数组中的每个成员为过滤规则。代理程序数据收集器接收到 HTTP 请求时，它会从该请求抽取 URL 路径名，并将其与每个 "pattern" 进行比较。如果路径名不匹配 "pattern"，那么原始 URL 路径名会保留并用于度量。

代理程序数据收集器日志记录参数

通过修改 *KNJ_NPM_LIB_LOCATION*/node_modules/ibmapm/etc 目录中的 config.properties 配置文件中的参数来更改日志记录行为。您可以更改以下日志记录参数：

日志级别

在配置文件中，用于指定日志级别的条目是 KNJ_LOG_LEVEL=info，这表示会在日志中打印有关操作的摘要信息。您可以通过更改 KNJ_LOG_LEVEL 的值来设置日志级别。缺省值为 info，且日志将作为标准输出进行打印。

支持以下五个日志级别：

off
不打印日志。

error
仅记录错误条件的信息。

info
当 Node.js 代理程序数据收集器正常运行时，记录信息。还记录了发送到代理程序的原始监视数据。

debug
在日志中打印调试、info 和错误信息，例如，收集的数据、发送到服务器的数据以及服务器响应。

all
在日志中打印所有信息。

配置 Node.js 代理程序诊断数据收集器

缺省情况下，已禁用对诊断数据收集的支持。如果具有诊断功能，必须设置和调整特定 Node.js 应用程序的数据收集。

过程

- 要修改正在运行的特定应用程序的数据收集器设置：
  1. 浏览到 *KNJ_NPM_LIB_LOCATION*/node_modules/ibmapm 目录，并在文本编辑器中打开文件 plugin_*port*_conf.json。

     提示: 有关 *KNJ_NPM_LIB_LOCATION* 的信息，请参阅 第 505 页的『KNJ_NPM_LIB_LOCATION』 的参数描述
  2. 使用下表以获取有关修改数据收集器设置的信息：

表 *187*. 数据收集器设置

诊断数据类别	描述	属性	操作
堆栈跟踪报告的最小时间增量	指定收集请求或方法调用的堆栈跟踪时所用的响应时间阈值。如果请求或方法调用的响应时间超过这个值，那么数据收集器将收集其堆栈跟踪。	`minClockStack`	设置为以毫秒为单位的值
报告请求的最小时间增量	指定收集请求实例的方法跟踪时所用的响应时间阈值。如果请求实例的响应时间超过此阈值，那么数据收集器将收集其方法跟踪。	`minClockTrace`	设置为以毫秒为单位的值
每个文件的最大事件数	指定要在 `.jso` 文件中记录的最大事件数目。`.jso` 文件将记录这些事件的诊断数据。	`eventsPerFile`	设置为最大事件数值
向一个文件报告所用的最大时间量	指定 `.jso` 文件要记录诊断数据的最大时间量。	`fileCommitTime`	设置为最长时间（秒）
删除最旧文件之前保留的最大文件数	指定在删除最旧的 `.jso` 文件之前，要保留的最大文件数。	`maxFiles`	设置为最大文件数
请求采样周期	指定请求的采样周期。	`sampling`	设置为所需采样周期。缺省值为 `10`。值 `10` 表示代理程序每 `10` 个请求收集其中一个。

- 可选： 如果您希望诊断数据收集器收集用户敏感信息（例如，cookie、HTTP 请求上下文和数据库请求上下文），请设置 *SECURITY_OFF* 环境变量。缺省情况下不收集此消息。

  在设置此变量时，请务必谨慎，因为它可能会导致信息泄漏。

  Linux 例如，要设置此环境变量，请发出以下命令：

  ```
  export SECURITY_OFF=true
  ```

结果

针对您所指定的正在运行的应用程序或针对所有应用程序，诊断数据收集器的配置已更改。

针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器

要收集 IBM Cloud 上的 Node.js 应用程序的相关信息，必须配置独立 Node.js 数据收集器。

开始之前

1. 确保 Node.js 应用程序可以在本地成功运行。独立 Node.js 数据收集器可以监视 Node.js V8.0.0 和未来修订包、V10.0.0 和未来修订包以及 V12.0.0 和未来修订包。

2. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

过程

1. 从数据收集器软件包中解压缩文件。已抽取的目录中包含 nodejs_datacollector_8.1.4.0.6.tgz 软件包。
2. 解压缩 nodejs_datacollector_8.1.4.0.6.tgz 文件，例如，通过运行以下命令进行解压缩：

```
tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
```

3. 通过运行以下命令，解压缩 nodejs_dc 文件夹中的 ibmapm.tgz 文件：

```
tar -zxf nodejs_dc/ibmapm.tgz
```

这将创建 ibmapm 文件夹。

4. 将从数据收集器软件包解压缩的 ibmapm 文件夹复制到应用程序的主目录，例如，通过运行以下命令进行复制：

```
cp -r directory_to_the_ibmapm_folder  home_directory_of_your_Node.js_application
```

提示: Node.js 应用程序主目录由用来启动 Node.js 应用程序的命令以及主文件所在的目录确定。如果您使用 **node app.js** 命令来启动 Node.js 应用程序，并且 app.js 主文件在 /root/nodejs_app 目录中，那么应用程序主目录是 /root/nodejs_app。

5. 在 Node.js 应用程序的 package.json 文件中，将以下行添加到 dependencies 小节：

```
 “ibmapm": "./ibmapm"
```

切记: 请勿遗漏文件中最后一行以外各行末尾的逗号，并保持 package.json 文件格式正确。

示例：

```
"dependencies": {
   "ibmapm": "./ibmapm",
   "cors": "^2.5.2",
   "helmet": "^1.3.0",
   "loopback": "^2.22.0",
   "loopback-boot": "^2.6.5",
   "loopback-datasource-juggler": "^2.39.0",
   "serve-favicon": "^2.0.1",
   "strong-error-handler": "^1.0.1"
 }
```

6. 在 Node.js 应用程序的主文件开头添加下面这一行：

```
require('ibmapm');
```

如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序主文件是 app.js。

7. 从 Node.js 应用程序 manifest.yml 文件所在的目录中，登录 IBM Cloud，然后运行以下命令：

```
cf push
```

提示: 要获取样本 manifest.yml 文件，请参阅第 160 页的『样本 manifest.yml 文件』。

结果

此数据收集器已配置完成，并已连接到 Cloud APM 服务器。

下一步做什么

您可以确认 IBM Cloud 应用程序的监视数据已显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

切记: 要将应用程序添加到 Cloud APM 控制台中，请在应用程序编辑器中选择 **Node.js 运行时**。

针对 IBM Cloud 应用程序定制独立 Node.js 数据收集器
您可以在 IBM Cloud 用户界面上添加环境变量，以定制针对 IBM Cloud 应用程序进行的监视。

Node.js 数据收集器的用户定义环境变量

您可使用下表中的信息来定制针对 IBM Cloud 的 Node.js 监视。

表 *188.* 针对 *IBM Cloud* 的 *Node.js* 监视所支持的用户定义环境变量

变量名称	重要性	值	描述
KNJ_SAMPLING	可选	请求的采样计数	采样所基于的请求数。 缺省值为 10，这表示在每 10 个请求中监视 1 个请求。 如果未设置此变量，那么将使用缺省值 10。
KNJ_MIN_CLOCK_TRACE	可选	收集方法跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其方法跟踪。 缺省值为 0。 如果未设置此变量，那么将使用缺省值 0。
KNJ_MIN_CLOCK_STACK	可选	收集堆栈跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其堆栈跟踪。 缺省值为 0。 如果未设置此变量，那么将使用缺省值 0。
KNJ_ENABLE_METHODTRACE	可选	· True · False	启用或禁用方法跟踪。 · 如果将此变量设置为 true，那么禁用请求的方法跟踪。 · 将此值设置为 false 表示启用请求的方法跟踪。这是缺省值。 如果未设置此变量，那么将使用缺省值 False，并启用请求的方法跟踪。
KNJ_ENABLE_DEEPDIVE	可选	· True · False	如果将此变量设置为 true，那么将向服务器发送诊断数据。缺省情况下，此值设置为 false，这表示不会向服务器发送诊断数据。
KNJ_ENABLE_TT	可选	· true · false	启用或禁用 AAR 事务跟踪。 · 将此变量设置为 true 表示启用 AAR 事务跟踪。 · 将此变量设置为 false 表示禁用 AAR 事务跟踪。 缺省情况下，未设置此值，表示禁用事务跟踪。

表 *188.* 针对 *IBM Cloud* 的 *Node.js* 监视所支持的用户定义环境变量 *(续)*

变量名称	重要性	值	描述
KNJ_AAR_BATCH_FREQ	可选	发送 AAR 数据的时间间隔（以秒计）	指定对 AAR 数据进行批处理并将其发送到服务器的时间间隔（以秒计）。 缺省值为 60，表示每分钟对 AAR 数据进行批处理并将其发送到服务器。 **注:** 此变量与 KNJ_AAR_BATCH_COUNT 配合使用，以确定何时对 AAR 数据进行批处理并发送到服务器。当满足两个变量之一设置的条件时，对 AAR 数据进行批处理并发送。当 AAR 数据包含的请求数达到最大数量（例如 100）时，在比设置的更短的时间间隔内，仍对数据进行批处理并立即发送。
KNJ_AAR_BATCH_COUNT	可选	批量 AAR 数据包含的最大请求数	指定批量 AAR 数据可包含的最大请求数，然后再将其发送到服务器。 缺省值为 100，表示当批量 AAR 数据包含的请求数达到 100 时，将此批量 AAR 数据发送到服务器。
KNJ_LOG_LEVEL	可选	在日志中显示的信息级别	控制在日志中打印的信息的级别。提供了以下级别: **off** 不打印日志。 **error** 仅记录错误条件的信息。 **info** 当 Node.js 代理程序数据收集器正常运行时，记录信息。还记录了发送到代理程序的原始监视数据。 **debug** 在日志中显示调试信息、参考信息和错误信息，例如，收集到的数据、发送到服务器的数据以及服务器响应。 **all** 在日志中打印所有信息。 缺省情况下，日志级别为 `info`，这表示在日志中打印有关数据收集器操作的摘要信息。日志将作为标准输出进行打印。

表 188. 针对 *IBM Cloud* 的 *Node.js* 监视所支持的用户定义环境变量 *(续)*			
变量名称	**重要性**	**值**	**描述**
SECURITY_OFF	可选	· true · false	启用或禁用用户敏感信息（例如，cookie、HTTP 请求上下文和数据库请求上下文）的收集。 · 如果将此变量设置为 true，那么收集用户敏感信息。 · 如果将此变量设置为 false，那么不收集用户敏感信息。这是缺省值。 未指定此变量时，将会使用缺省值 false 并且不收集用户敏感信息。

取消为 IBM Cloud 应用程序配置 Node.js 数据收集器

如果不需要监视 Node.js 环境，或者如果要升级独立 Node.js 数据收集器，首先必须取消配置独立 Node.js 数据收集器的先前设置。

过程

1. 从应用程序主文件中移除 `require('ibmapm');` 行。

   提示: 如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序的主文件是 app.js。
2. 从 package.json 文件中移除下列依赖项。

   ```
   “ibmapm": "./ibmapm"
   ```

   切记: 请勿移除应用程序需要的依赖项。
3. 从应用程序的主目录中移除 ibmapm 文件夹。

结果

您已成功取消配置独立 Node.js 数据收集器。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

为本地应用程序配置独立 Node.js 数据收集器

如果已在本地环境中安装 Node.js 应用程序，那么必须配置 Node.js 数据收集器以收集有关 Node.js 应用程序的信息。

开始之前

1. 确保 Node.js 应用程序可以在本地成功运行。独立 Node.js 数据收集器可监视 Node.js V8.0.0 和未来修订包、V10.0.0 和未来修订包以及 V12.0.0 和未来修订包。
2. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

过程

1. 从数据收集器软件包抽取文件。已抽取的目录中包含 nodejs_datacollector_8.1.4.0.6.tgz 软件包。
2. 确定应用程序的主目录。

- 对于典型 Node.js 应用程序，如果使用 **node app.js** 命令来启动 Node.js 应用程序，且 app.js 主文件位于 /root/nodejs_app 目录中，那么 /root/nodejs_app 是应用程序的主目录。
- 对于 IBM API Connect 环境中的集合成员，运行 **wlpn-server list** 命令以显示同一台机器上的所有集合成员的列表。集合成员的主目录采用以下格式：

  ```
  user_directory/collective-member_name/package
  ```

  例如，如果命令输出为 /root/wlpn/rock-8345a96-148538-1/package，那么 /root/wlpn 是用户目录，rock-8345a96-148538-1 是集合成员名称。
- 对于 IBM API Connect 环境中的开发人员门户网站应用程序，可以运行 **ps -ef | grep node** 命令来查找主目录。例如，假设获得以下命令输出，那么主目录是 /home/admin/bgsync，应用程序的主文件是 rest_server.js：

  ```
  admin 19085 1 0 Jun25 ?  00:06:53 /usr/local/bin/node /home/admin/bgsync/
  rest_server.js
  ```

3. 从应用程序的主目录，运行以下命令从数据收集器软件包中抽取文件：

   ```
   tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
   ```

4. 通过运行以下命令，解压缩 nodejs_dc 文件夹中的 ibmapm.tgz 文件：

   ```
   tar -zxf nodejs_dc/ibmapm.tgz
   ```

   这将创建 ibmapm 文件夹。
5. 运行以下命令将数据收集器安装到应用程序：

   ```
   npm install ./ibmapm
   ```

6. 将以下行添加到 Node.js 应用程序的主文件开头：

   ```
   require('ibmapm');
   ```

   - 如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序的主文件是 app.js。
   - 对于 IBM API Connect 环境中的集合成员，主文件在主目录或其子文件夹中的 package.json 文件中定义。缺省情况下，主文件是 *home_directory*/server/server.js，其中，*home_directory* 是集合成员的主目录。
   - 对于 IBM API Connect 环境中的开发人员门户网站应用程序，可以运行 **ps -ef | grep node** 命令来查找主文件。例如，假设获得以下命令输出，那么应用程序的主文件是 rest_server.js。

     ```
     admin 19085 1 0 Jun25 ?  00:06:53 /usr/local/bin/node /home/admin/bgsync/
     rest_server.js
     ```

7. 重新启动您的应用程序。

   提示：

   - 要重新启动集合成员，请运行 wlpn-server stop *collective_member_name* 命令。集合成员将在您运行该命令之后自动重新启动。如果没有启动，请运行 wlpn-server start *collective_member_name* 命令进行手动重新启动。
   - 要重新启动开发人员门户网站应用程序，首先运行 /etc/init.d/restservice stop 命令来停止应用程序，然后运行 /etc/init.d/restservice start 命令进行启动。

结果

此数据收集器已配置完成，并已连接到 Cloud APM 服务器。

下一步做什么

- 您可以验证应用程序的监视数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。
- 要查看 API Connect 环境的拓扑信息，启用事务跟踪。有关更多指示信息，请参阅第 515 页的『为本地应用程序定制 Node.js 数据收集器』中的 *KNJ_ENABLE_TT* 变量描述。

切记: 要将应用程序添加到 Cloud APM 控制台中，请在应用程序编辑器中选择 **Node.js 运行时**。

为本地应用程序定制 Node.js 数据收集器

通过修改数据收集器软件包中的文件，可设置环境变量，以定制针对 Node.js 应用程序进行的监视。

您可以通过定制环境变量或者编辑 `config.properties` 文件，设置变量。您可以在安装 Node.js 数据收集器的 `ibmapm/etc` 文件夹中找到 `config.properties` 文件。

表 *189.* 受支持的变量

变量名称	重要性	值	描述
KNJ_SAMPLING	可选	请求的采样计数	采样所基于的请求数。 缺省值为 `10`，这表示在每 10 个请求中监视 1 个请求。 如果未设置此变量，那么将使用缺省值 `10`。
KNJ_MIN_CLOCK_TRACE	可选	收集方法跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其方法跟踪。 缺省值为 `0`。 如果未设置此变量，那么将使用缺省值 `0`。
KNJ_MIN_CLOCK_STACK	可选	收集堆栈跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其堆栈跟踪。 缺省值为 `0`。 如果未设置此变量，那么将使用缺省值 `0`。
KNJ_ENABLE_METHODTRACE	可选	· `True` · `False`	启用或禁用方法跟踪。 · 如果将此变量设置为 true，那么禁用请求的方法跟踪。 · 将此值设置为 false 表示启用请求的方法跟踪。这是缺省值。 如果未设置此变量，那么将使用缺省值 `False`，并启用请求的方法跟踪。
KNJ_ENABLE_DEEPDIVE	可选	· `True` · `False`	如果将此变量设置为 true，那么将向服务器发送诊断数据。缺省情况下，此值设置为 false，这表示不会向服务器发送诊断数据。

表 *189.* 受支持的变量 *(续)*

变量名称	重要性	值	描述
KNJ_ENABLE_TT	可选	· `true` · `false`	启用或禁用 AAR 事务跟踪。 · 将此变量设置为 true 表示启用 AAR 事务跟踪。 · 将此变量设置为 false 表示禁用 AAR 事务跟踪。 缺省情况下，未设置此值，表示禁用事务跟踪。
KNJ_AAR_BATCH_FREQ	可选	发送 AAR 数据的时间间隔（以秒计）	指定对 AAR 数据进行批处理并将其发送到服务器的时间间隔（以秒计）。 缺省值为 60，表示每分钟对 AAR 数据进行批处理并将其发送到服务器。 **注:** 此变量与 KNJ_AAR_BATCH_COUNT 配合使用，以确定何时对 AAR 数据进行批处理并将其发送到服务器。当满足两个变量之一设置的条件时，对 AAR 数据进行批处理并发送。当 AAR 数据包含的请求数达到最大数量（例如 100）时，在比设置的更短的时间间隔内，仍对数据进行批处理并立即发送。
KNJ_AAR_BATCH_COUNT	可选	批量 AAR 数据包含的最大请求数	指定批量 AAR 数据可包含的最大请求数，然后再将其发送到服务器。 缺省值为 100，表示当批量 AAR 数据包含的请求数达到 100 时，将此批量 AAR 数据发送到服务器。
KNJ_LOG_LEVEL	可选	在日志中打印的信息的级别	控制在日志中打印的信息的级别。提供了以下级别： **off** 不打印日志。 **error** 仅记录错误条件的信息。 **info** 当 Node.js 代理程序数据收集器正常运行时，记录信息。还记录了发送到代理程序的原始监视数据。 **debug** 在日志中显示调试信息、参考信息和错误信息，例如，收集到的数据、发送到服务器的数据以及服务器响应。 **all** 在日志中打印所有信息。 缺省情况下，日志级别为 `info`，这表示在日志中打印有关数据收集器操作的摘要信息。日志将作为标准输出进行打印。

表 189. 受支持的变量 (续)			
变量名称	**重要性**	**值**	**描述**
SECURITY_OFF	可选	· true · false	启用或禁用用户敏感信息（例如，cookie、HTTP 请求上下文和数据库请求上下文）的收集。 · 如果将此变量设置为 true，那么收集用户敏感信息。 · 如果将此变量设置为 false，那么不收集用户敏感信息。这是缺省值。 未指定此变量时，将会使用缺省值 false 并且不收集用户敏感信息。

为本地应用程序取消配置独立 Node.js 数据收集器

如果不需要监视 Node.js 环境，或者如果要升级独立 Node.js 数据收集器，首先必须取消配置独立 Node.js 数据收集器的先前设置。

过程

1. 从应用程序主文件中移除 `require('ibmapm');` 行。

   提示: 如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序的主文件是 app.js。
2. 从 Node.js 应用程序 package.json 文件的 dependencies 节中移除 `"ibmapm": "./ibmapm"`。
3. 从应用程序主目录中删除 node_modules 文件夹。
4. 运行 `npm install` 命令安装应用程序相关软件。

结果

您已成功取消配置独立 Node.js 数据收集器。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

为 Kubernetes 应用程序配置独立 Node.js 数据收集器

如果您在 Kubernetes 上安装了 Node.js 应用程序，那么可以配置 Node.js 数据收集器 来收集有关 Node.js 应用程序的信息。

开始之前

1. 确保 Node.js 应用程序可以成功运行。独立 Node.js 数据收集器 可以监视 Node.js V8.0.0 和未来修订包、V10.0.0 和未来修订包，以及 V12.0.0 和未来修订包。
2. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

过程

1. 从数据收集器软件包抽取文件。已抽取的目录中包含 nodejs_datacollector_8.1.4.0.6.tgz 软件包。
2. 解压缩 nodejs_datacollector_8.1.4.0.6.tgz 文件，例如，通过运行以下命令进行解压缩：

   ```
   tar -zxf nodejs_datacollector_8.1.4.0.6.tgz
   ```
3. 通过运行以下命令，解压缩 nodejs_dc 文件夹中的 ibmapm.tgz 文件：

```
tar -zxf nodejs_dc/ibmapm.tgz
```

这将创建 ibmapm 文件夹。

4. 在 Node.js 应用程序的 package.json 文件中，将以下行添加到 dependencies 部分中：

```
"ibmapm": "./ibmapm"
```

切记: 请勿遗漏文件中最后一行以外各行末尾的逗号，并保持 package.json 文件格式正确。

5. 将以下行添加到 Node.js 应用程序的主文件开头：

```
require('./ibmapm');
```

如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序的主文件是 app.js。

6. 重新构建 Docker 映像。

注: 如果在其他 Docker 环境（例如，Docker Swarm 或 AWS Docker 服务）上运行 Node.js 应用程序，那么需要对这些步骤进行 Docker 化。

下一步做什么

如果要定制监视，那么可以在部署 yaml 文件中添加环境变量。有关详细信息，请参阅第 518 页的『为 Kubernetes 应用程序定制独立 Node.js 数据收集器』。

为 Kubernetes 应用程序定制独立 Node.js 数据收集器

您可以将环境变量添加到部署 yaml 文件来定制 Kubernetes 应用程序的监视。

Node.js 数据收集器的用户定义环境变量

您可使用下表中的信息来定制 Kubernetes 上的 Node.js 监视。

表 *190. Kubernetes* 上的 *Node.js* 监视所支持的用户定义的环境变量

变量名称	重要性	值	描述
KNJ_SAMPLING	可选	请求的采样计数	采样所基于的请求数。 缺省值为 10，这表示在每 10 个请求中监视 1 个请求。 如果未设置此变量，那么将使用缺省值 10。
KNJ_MIN_CLOCK_TRACE	可选	收集方法跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其方法跟踪。 缺省值为 0。 如果未设置此变量，那么将使用缺省值 0。
KNJ_MIN_CLOCK_STACK	可选	收集堆栈跟踪时采用的响应时间阈值（毫秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其堆栈跟踪。 缺省值为 0。 如果未设置此变量，那么将使用缺省值 0。

表 190. *Kubernetes* 上的 *Node.js* 监视所支持的用户定义的环境变量 *(续)*

变量名称	重要性	值	描述
KNJ_ENABLE_METHODTRACE	可选	· True · False	启用或禁用方法跟踪。 · 如果将此变量设置为 true，那么禁用请求的方法跟踪。 · 将此值设置为 false 表示启用请求的方法跟踪。这是缺省值。 如果未设置此变量，那么将使用缺省值 False，并启用请求的方法跟踪。
KNJ_ENABLE_DEEPDIVE	可选	· True · False	如果将此变量设置为 true，那么将向服务器发送诊断数据。缺省情况下，此值设置为 false，这表示不会向服务器发送诊断数据。
KNJ_ENABLE_TT	可选	· true · false	启用或禁用 AAR 事务跟踪。 · 将此变量设置为 true 表示启用 AAR 事务跟踪。 · 将此变量设置为 false 表示禁用 AAR 事务跟踪。 缺省情况下，未设置此值，表示禁用事务跟踪。
KNJ_AAR_BATCH_FREQ	可选	发送 AAR 数据的时间间隔（以秒计）	指定对 AAR 数据进行批处理并将其发送到服务器的时间间隔（以秒计）。 缺省值为 60，表示每分钟对 AAR 数据进行批处理并将其发送到服务器。 **注:** 此变量与 KNJ_AAR_BATCH_COUNT 配合使用，以确定何时对 AAR 数据进行批处理并将其发送到服务器。当满足两个变量之一设置的条件时，对 AAR 数据进行批处理并发送。当 AAR 数据包含的请求数达到最大数量（例如 100）时，在比设置的更短的时间间隔内，仍对数据进行批处理并立即发送。
KNJ_AAR_BATCH_COUNT	可选	批量 AAR 数据包含的最大请求数	指定批量 AAR 数据可包含的最大请求数，然后再将其发送到服务器。 缺省值为 100，表示当批量 AAR 数据包含的请求数达到 100 时，将此批量 AAR 数据发送到服务器。

表 *190. Kubernetes* 上的 *Node.js* 监视所支持的用户定义的环境变量 *(续)*

变量名称	重要性	值	描述
KNJ_LOG_LEVEL	可选	在日志中显示的信息级别	控制在日志中打印的信息的级别。提供了以下级别： **off** 不打印日志。 **error** 仅记录错误条件的信息。 **info** 当 Node.js 代理程序数据收集器正常运行时，记录信息。还记录了发送到代理程序的原始监视数据。 **debug** 在日志中显示调试信息、参考信息和错误信息，例如，收集到的数据、发送到服务器的数据以及服务器响应。 **all** 在日志中打印所有信息。 缺省情况下，日志级别为 `info`，这表示在日志中打印有关数据收集器操作的摘要信息。日志将作为标准输出进行打印。
SECURITY_OFF	可选	· `true` · `false`	启用或禁用用户敏感信息（例如，cookie、HTTP 请求上下文和数据库请求上下文）的收集。 · 如果将此变量设置为 true，那么收集用户敏感信息。 · 如果将此变量设置为 false，那么不收集用户敏感信息。这是缺省值。 未指定此变量时，将会使用缺省值 `false` 并且不收集用户敏感信息。

yaml 文件示例

```
spec:
      containers:
      - name: testapp
        image: mycluster.icp:8500/default/testapp:v1
        imagePullPolicy: Always
        ports:
        - containerPort: 3000
          protocol: TCP
        env:        - name: KNJ_LOG_LEVEL
          value: "debug"
        - name: KNJ_ENABLE_TT
          value: "true"
        - name: KNJ_ENABLE_DEEPDIVE
          value: "true"
```

为 Kubernetes 应用程序取消配置独立 Node.js 数据收集器

如果不需要监视 Node.js 环境，或者如果要升级独立 Node.js 数据收集器，首先必须取消配置独立 Node.js 数据收集器的先前设置。

过程

1. 从应用程序主文件中移除 `require('ibmapm');` 行。

   提示: 如果您通过运行 **node app.js** 命令来启动应用程序，那么应用程序的主文件是 `app.js`。
2. 从 `package.json` 文件中移除下列依赖项。

   ```
   "ibmapm": "./ibmapm"
   ```

   切记: 请勿移除应用程序需要的依赖项。

结果

您已成功取消配置独立 Node.js 数据收集器。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

配置 OpenStack 监视

您必须先配置 Monitoring Agent for OpenStack，然后该代理程序才能自动监视 OpenStack 代理程序环境。

过程

1. 通过对提示做出响应来配置代理程序。有关指示信息，请参阅第 521 页的『配置 OpenStack 代理程序』。
2. 如果要收集流程相关信息，请为 OpenStack 代理程序配置数据收集器。有关指示信息，请参阅第 523 页的『启用流程相关信息收集和 SSH 连接』。
3. 输入配置值才能运行该代理程序。有关指示信息，请参阅第 524 页的『添加配置值』。

配置 OpenStack 代理程序

对于典型环境，如果您希望 OpenStack 代理程序 自动监视 OpenStack 环境，您必须先配置代理程序。

开始之前

确保已安装在 Linux 系统上预安装中描述的所有必需软件。

过程

有两个选项可用于在 Linux 系统上配置 OpenStack 代理程序：

- 要通过运行脚本并响应提示来配置代理程序，请参阅第 521 页的『交互式配置』。
- 要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请参阅第 522 页的『静默配置』。

交互式配置

过程

1. 要配置代理程序，请运行以下命令：

   ```
   install_dir/bin/openstack-agent.sh config instance_name
   ```

   其中，*install_dir* 是 OpenStack 代理程序的安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。

2. 当出现输入实例名称提示时，请指定一个实例名称。

   要点: OpenStack 代理程序 是一个多实例代理程序，每个代理程序实例都需要一个实例名称。您指定的实例名称包含在受管系统名称 `instance_name:host_name:sg` 中。指定的实例名称长度限制为 28 个字符减去主机名长度。例如，如果指定 `OS1` 作为实例名称，那么受管系统名称为 `OS1:hostname:SG`。
3. 当出现`编辑 Monitoring Agent for OpenStack` 提示时，请按 `Enter` 键以继续。
4. 当出现`编辑 OpenStack 环境认证信息`提示时，请提供以下信息：

   ```
   OpenStack authentication url (default is: http://localhost:identity/v3):
   OpenStack username (default is: admin):
   Enter OpenStack password (default is: ):
   Re-type: OpenStack password (default is: ):
   OpenStack tenant name (default is: admin):
   ```
5. 当出现 `Python 可执行文件位置`提示时，请指定 Python 可执行文件位置，如 `/usr/bin/python`。

   您可以通过在您的环境中运行以下命令来找到标准路径：

   ```
   which python
   ```
6. 当出现`端口号`提示时，请接受缺省值或指定端口号。

   此端口用于监视 OpenStack 代理程序数据收集器和 OpenStack 代理程序之间的内部通信，这两项都只安装在本地服务器上。代理程序在此端口侦听来自数据收集器的数据。缺省值 0 指示在代理程序启动时使用临时端口。在对端口执行严格安全规则的服务器上，您可以为要使用的代理程序配置一个特定端口。此端口供代理程序内部使用并且与 OpenStack 环境无关。
7. 在您的系统上编辑 `/etc/hosts` 文件，为每个受监视的节点添加主机映射。

静默配置

过程

1. 在文本编辑器中打开 `sg_silent_config.txt` 文件。此文件位于 *`install_dir`*`/samples` 目录中，其中，*install_dir* 是 OpenStack 代理程序 的安装目录。
2. 为 OpenStack 代理程序编辑 `sg_silent_config.txt` 配置文件。
3. 为该文件中标识的参数指定值。响应文件包含用于定义可用参数以及要指定的值的注释。
4. 保存文件并退出。
5. 在您的系统上编辑 `/etc/hosts` 文件，为每个受监视的节点添加主机映射。
6. 从 *`install_dir`*`/samples` 目录，运行以下命令以配置代理程序：

   ```
   install_dir/bin/openstack-agent.sh config instance_name path_to_response_file
   ```

   其中，*instance_dir* 是要配置的实例的名称，*path_to_responsefile* 是静默响应文件的完整路径。指定此文件的绝对路径。

   例如，如果响应文件位于缺省目录中，请运行以下命令。

   ```
   /opt/ibm/apm/agent/bin/openstack-agent.sh config instance_name
   /opt/ibm/apm/agent/samples/sg_silent_config.txt
   ```

结果
代理程序已配置。

下一步做什么

· 在完成配置代理程序后，您可以运行以下命令来启动代理程序实例：

```
install_dir/bin/openstack-agent.sh start instance_name
```

其中，*instance_name* 是要配置的代理程序实例的名称。

· 要将 OpenStack 代理程序 连接到已启用 SSL 的 OpenStack 环境，请设置以下变量来指定 OpenStack 服务器 SSL 证书的目录：

```
OS_cert_path=directory of the certificate.crt file
```

OS_cert_path 变量位于 ksg_dc_*instance_name*.cfg 文件中的 OS_authentication_info 部分中。

· 如果要收集流程相关信息，通过完成第 523 页的『启用流程相关信息收集和 SSH 连接』中的步骤，为 OpenStack 代理程序配置数据收集器。

· 如果要更改代理程序的跟踪级别以进行故障诊断，请根据 *install_dir*/config/sg.environment 文件中的指示信息在该文件中编辑变量 **KBB_RAS1** 的值。

启用流程相关信息收集和 SSH 连接

如果要收集流程相关信息，请为 OpenStack 代理程序配置代理程序数据收集器并设置与目标 OpenStack 组件服务器的 SSH 连接。

关于此任务

在启动 OpenStack 代理程序前，您必须设置 SSH 连接以收集流程信息。要设置该连接，按照以下过程的描述，使用产品提供的 **ksg_setup_key.sh** 或 **ksg_ssh_setup.py** 帮助工具。

如果您很熟悉如何设置 SSH 连接，还可以使用 **ssh-keygen** 和 **ssh-copy-id** Linux 命令来设置连接

过程

1. 转至 *install_dir*/config 目录，其中 *install_dir* 是代理程序安装目录。
2. 编辑 ksg_dc_*instance_name*.cfg 文件，其中 *instance_name* 是为此代理程序实例指定的名称。

   此文件在代理程序实例启动后创建。如果该文件不存在，将 *install_dir*/lx8266/sg/bin/ksg_dc.cfg 复制到 *install_dir*/config 目录，并将文件名更改为 ksg_dc_*instance_name*.cfg。

   例如，如果实例名称为 OS1，请将名称更改为 ksg_dc_OS1.cfg。
3. 在 ksg_dc_*instance_name*.cfg 文件中，将参数 **collect_process_information** 的值设置为 YES。
4. 在 **OS_process_collection** 节中，按照以下示例的格式，使用 OpenStack 组件服务器的用户和主机名或 IP 地址指定 **ssh_user_host** 参数的值：

   ```
   ssh_user_host=root@9.112.250.248,user1@hostname
   ```
5. 保存设置。
6. 要使设置生效，请运行以下命令来重新启动代理程序实例：

   ```
   install_dir/bin/openstack-agent.sh stop instance_name
   install_dir/bin/openstack-agent.sh start instance_name
   ```
7. 使用以下某种方法设置与目标组件服务器的 SSH 连接：

   · 使用 ksg_setup_key.sh 脚本逐个设置连接：转至 *install_dir*/lx8266/sg/bin 目录，使用主机名或 IP 和用户运行 ksg_setup_key.sh 脚本，以构建与步骤 4 中指定的组件服务器的 SSH 连接。如果遵循步骤 4 中给定的示例，那么必须运行两次该脚本以逐一设置连接：

   ```
   ./ksg_setup_key.sh 9.112.250.248 root
   ./ksg_setup_key.sh hostname user1
   ```

   注: 您必须在第一次运行脚本时提供密码。不需要再次提供密码。

   · 使用 OpenStack 代理程序 在 *install_dir*/lx8266/sg/bin 中提供的 ksg_ssh_setup.py 工具，逐个或批量设置连接。在使用此工具前，您必须安装 Python pexpect 库。

   – 要逐一设置 SSH 连接，请运行以下命令：

```
python ksg_ssh_setup.py -single
```

此命令可帮助您设置与远程目标服务器的 SSH 连接。必须提供以下信息：

```
Enter the remote target machine host name orIP address: (Type 'END' to end input.)
Enter the account to access the remote machine(e.g. root):
Enter the above user's password:
```

– 要在批处理作业中设置 SSH 连接，请运行以下命令：

```
python ksg_ssh_setup.py -ssh SSH_file
```

其中，*SSH_file* 是包含目标服务器、用户和密码信息的文件。您必须根据 `ksg_dc_ssh_list.txt` 文件在与 Python 工具相同的目录中创建文件，并根据示例的格式在文件中指定主机和用户信息：

```
hostname root passw0rd
9.112.250.248 user1 passw0rd
```

注: 仅当目标服务器的用户名或密码更改时，才必须重新设置连接。在重新启动代理程序或更改代理程序配置后，无需重新设置连接。

结果

数据收集器已配置且 SSH 连接已正确设置。现在，您可以登录 Cloud APM 控制台并使用“应用程序”编辑器将 OpenStack 代理程序实例添加到应用程序性能仪表板。在添加代理程序实例时，从组件列表中选择 **OpenStack 环境**。

下一步做什么

- 当您单击 **API 端点摘要（按服务类型）** > **API 端点详细信息**时，将在 **API 检测故障时间历史记录**和 **API 检测故障百分比历史记录**组窗口小部件中看到无可用数据消息。单击在 **API 端点详细信息**中显示的 API 端点，您可以在两个组窗口小部件中查看监视数据。
- 当您单击**流程摘要（按组件）** > **流程详细信息**或 **SSH 服务器连接状态** > **流程详细信息**时，将在**流程 CPU 使用情况历史记录**和**流程内存使用情况历史记录**组窗口小部件中看到无可用数据消息。单击在**流程详细信息**中显示的流程，您可以在两个组窗口小部件中查看监视数据。

添加配置值

对于本地和远程配置，提供配置值以使代理程序运行。

当您使用交互方式来配置代理程序时，将显示一个面板，以便您能够输入每个值。如果存在缺省值，那么提前将此值输入到字段。如果一个字段表示一个密码，那么显示两个输入字段。您必须在每个字段中输入相同的值。您输入的值不会显示以帮助维护这些值的安全性。

当您使用静默方式来配置代理程序时，您可以编辑 *install_dir*/samples 目录中的 *response_file* 以添加配置值。在保存更改之后，根据步骤 第 522 页的『6』 中的指示信息，运行以下命令以使更改生效：

```
install_dir/bin/openstack-agent.sh start instance_name
```

其中，*instance_name* 是要配置的代理程序实例的名称。

在配置完成后，您可以在代理程序实例的 `.cfg` 文件（如 *hostname*`_sg_`*instance_name*`.cfg`）中找到已配置的值。

此代理程序的配置将组织到以下组中：

OpenStack 环境认证信息 (OPENSTACK_CONNECTION)

OpenStack 环境认证信息

该组中定义的配置元素在代理程序的配置中始终存在。

该组定义适用于整个代理程序的信息。

OpenStack 认证 URL (KSG_OPENSTACK_AUTH_URL)
OpenStack 环境的 auth_url

类型为字符串。

此值是必需的。

缺省值：`http://localhost:identity/v3`

OpenStack 密码 (KSG_OPENSTACK_PASSWORD)
管理员用户的密码

类型为密码。

此值是必需的。

缺省值：无

OpenStack 租户名称 (KSG_OPENSTACK_TENANT_NAME)
OpenStack 租户名称，也称为项目名称

类型为字符串。

此值是必需的。

缺省值：管理员

OpenStack 用户名称 (KSG_OPENSTACK_USERNAME)
管理员用户，用于登录 OpenStack 环境

类型为字符串。

此值是必需的。

缺省值：管理员

Python (KSG_PYTHON)
Python 可执行文件位置

该组中定义的配置元素在代理程序的配置中始终存在。

该组定义适用于整个代理程序的信息。

Python 可执行文件位置 (KSG_PYTHON_LOCATION)
Python 可执行文件，将用于运行 OpenStack 代理程序数据收集器。您可以通过在您的终端运行以下命令来查找标准路径：“which python”。

类型为字符串。

此值是必需的。

缺省值：无

套接字 (KSG_SOCKET)
套接字数据源

该组中定义的配置元素始终存在于代理程序的配置中。

该组定义适用于整个代理程序的信息。

端口号 (CP_PORT)
代理程序将用于侦听来自套接字客户机的数据的端口。值 0 指示将使用的临时端口。该端口不会响应应用程序使用的任何端口。此端口供代理程序内部使用。

类型为数字。

此值是可选的。

缺省值：0

配置 Oracle 数据库监视

Monitoring Agent for Oracle Database 提供针对 Oracle 数据库的可用性、性能和资源使用情况的监视功能。您可以配置多个 Oracle Database 代理程序实例，以监视不同的 Oracle 数据库。此代理程序还提供远程监视功能。

开始之前

- 配置 Oracle Database 代理程序之前，必须为 Oracle Database 代理程序使用的 Oracle 用户帐户授予特权。有关特权的更多信息，请参阅为 Oracle 数据库代理程序用户授予特权。
- 如果要以远程方式监视 Oracle 数据库，那么代理程序必须安装在安装有 Oracle 数据库软件或 Oracle Instant Client 的计算机上。

关于此任务

此处的方向针对的是代理程序的最新发行版，除非另有说明。有关如何检查环境中代理程序版本的信息，请参阅代理程序版本。

对于常规 Oracle 数据库性能监视，Oracle Database 代理程序还提供了对 Oracle 数据库的可用性、性能、资源使用情况和活动的监视，例如：

- 受监视 Oracle 数据库中的实例的可用性。
- 资源信息，例如，内存、高速缓存、分段、资源限制、表空间、撤销（回滚）、系统度量和系统统计信息。
- 活动信息，例如，操作系统统计信息、会话、争用和警报日志。

Oracle Database 代理程序 是多实例代理程序。您必须手动创建第一个实例并启动代理程序。另外，每个代理程序实例都可以监视多个数据库。

Oracle Database 代理程序的受管系统名称包含您所指定的数据库连接名称、您所指定的代理程序实例名称以及代理程序安装所在计算机的主机名。例如，`pc:connection_name-instance_name-host_name:SUB`，其中 *pc* 是两个字母的产品代码，*SUB* 是数据库类型（可能的值为 RDB、ASM 或 DG）。“受管系统名称”限制为 32 个字符。指定的实例名称限制为 23 个字符（减去主机名和数据库连接名称的长度之后）。例如，如果指定 **dbconn** 作为数据库连接名称，指定 **Oracle02** 作为代理程序实例名称，而且主机名为 *Prod204a*，那么受管系统名为 `RZ:dbconn-oracle02-Prod204a:RDB`。在 23 个可用于数据库连接名称、代理程序实例名称和主机名的字符中，此示例使用 22 个字符。

- 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法正确显示。
- *connection_name*、*instance_name* 和 *hostname_name* 变量的长度在超过 23 个字符时会截断。
- 要避免子节点名被截断，请通过设置以下环境变量来更改子节点命名约定：**KRZ_SUBNODE_INCLUDING_AGENTNAME**、**KRZ_SUBNODE_INCLUDING_HOSTNAME** 和 **KRZ_MAX_SUBNODE_ID_LENGTH**。
- 如果将 **KRZ_SUBNODE_INCLUDING_AGENTNAME** 设置为 NO，那么子节点名称的子节点标识部分不会包含代理程序实例名称。例如，
  - 缺省子节点名：*DBConnection-Instance-Hostname*
  - 环境变量设置为 NO 时的子节点名：*DBConnection-Hostname*
- 如果将 **KRZ_SUBNODE_INCLUDING_HOSTNAME** 设置为 NO，那么子节点名的子节点标识部分不会包含主机名。例如，
  - 缺省子节点名：*DBConnection-Instance-Hostname*
  - 环境变量设置为 NO 时的子节点名：*DBConnection-Instance*

过程

1. 要在 Windows 系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。

- 第 527 页的『在 Windows 系统上配置代理程序』.
- 第 533 页的『使用静默响应文件来配置代理程序』.

2. 要在 Linux 和 UNIX 系统上配置代理程序，您可以运行脚本并对提示做出响应，或者使用静默响应文件。
   - 第 530 页的『通过对提示做出响应来配置代理程序』.
   - 第 533 页的『使用静默响应文件来配置代理程序』.

下一步做什么

仅针对高级配置，Oracle 数据库管理员必须允许 Oracle 用户运行 `krzgrant.sql` 脚本以访问数据库，请参阅运行 krzgrant.sql 脚本。

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。这些日志的缺省路径如下所示：

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

在 Windows 系统上配置代理程序

可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置代理程序。更新配置值后，启动代理程序以保存更新后的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Oracle Database** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项处于禁用状态。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 在 Monitoring Agent for Oracle Database 窗口中，完成下列步骤：
   a) 为 Monitoring Agent for Oracle Database 实例输入唯一的实例名称，然后单击**确定**。
4. 在“**配置 ITCAM Extended Agent for Oracle Database**”窗口的“缺省数据库配置”窗格上，执行下列步骤：
   a) 输入**缺省用户名**。这是数据库连接的缺省数据库用户标识。
      此用户标识就是代理程序用于访问受监视数据库实例的标识。此用户标识必须对代理程序所需的动态性能视图和表具有 SELECT 特权。
   b) 输入**缺省密码**。这是与指定的缺省数据库用户标识相关联的密码。
   c) 如果 Oracle 代理程序版本为 8.0，请执行此步骤。
      1) 输入 **Oracle JDBC Jar 文件**。这是用来与 Oracle 数据库进行通信的 Oracle JDBC 驱动程序 Jar 文件的完整路径。在代理程序计算机上，必须要有支持 Oracle 代理程序所监视 Oracle 数据库版本的 Oracle Java 数据库连接 (JDBC) 驱动程序。
   d) 如果 Oracle 代理程序版本为 6.3.1.10，请执行这些步骤。
      1) 如果 Oracle Database 代理程序是安装在受监视的 Oracle 数据库服务器上，请选中**使用 Oracle 主目录中的库**，并输入 **Oracle 主目录**。（可选）要进行本地监视，可将 Oracle 主目录设置保留为空白，在此情况下，将使用 *ORACLE_HOME* 系统环境变量。
      2) 如果 Oracle Database 代理程序远离受监视的 Oracle 数据库服务器，请选中**使用 Oracle 即时客户机中的库**，并输入 **Oracle 即时客户机安装目录**。
   e) 如果您需要设置高级配置选项，请选中**显示高级选项**，否则前进到步骤 5。

f) `Net Configuration Files Directories` 可以保留为空并使用缺省目录。如果 Oracle 代理程序版本为 6.3.1.10，您可输入多个网络配置文件目录，并以分号 (;) 进行分隔。Oracle 代理程序 V8.0 只支持一个目录。

此设置包含一个或多个 Oracle 数据库网络配置文件。目录由每个 Oracle 数据库实例的 *TNS_ADMIN* 环境变量进行定义。缺省目录为 `%ORACLE_HOME%\NETWORK\ADMIN`。如果未配置这个项，那么将使用缺省目录。要禁止使用缺省目录，请将下列代理程序环境变量设置为 false：`KRZ_LOAD_ORACLE_NET=false`。

g) 将 `定制 SQL 定义文件名`保留为空。不使用该属性。

h) 选择是否在此工作站上配置缺省动态侦听器。

缺省动态侦听器是 `(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)`。如果在此工作站上配置了缺省动态侦听器，请将这个值设置为是。

i) 单击**下一步**。

5. 在“**配置 ITCAM Extended Agent for Oracle Database**”窗口的**实例配置**窗格上，执行下列步骤：

这是定义实际数据库连接实例的位置。您必须至少添加一个数据库连接实例。这也是您编辑和删除数据库连接实例的位置。如果存在多个数据库连接实例配置，请使用**数据库连接**选项来选择要编辑或删除的实例。

a) 按**数据库连接**部分中的**新建**。

b) 输入**数据库连接名称**，作为数据库连接的别名。

此别名用来代表数据库连接，可以随您选择，但有下列限制。只有字母、阿拉伯数字、下划线字符和减号可用于连接名称中。连接名称的最大长度为 25 个字符。

c) 选择**连接类型**

1) （可选）基本

缺省连接类型（也就是最常用的连接类型）是**基本**。如果您不确定需要何种连接类型，建议您选择此连接类型。

a) 当目标受监视数据库是单个实例（例如，标准文件系统实例或单个 ASM 实例）时，请选择**基本**连接类型。

b) 输入**主机名**，作为数据库的主机名或 IP 地址。

c) 输入该数据库所使用的**端口号**。

d) 选择**服务名称**或 **SID**。

i. 如果选择**服务名称**，请输入作为数据库逻辑表示的服务名称（即，本身是全局数据库服务名称的字符串）。

服务名称是数据库的逻辑表示，这是向客户端呈现数据库的方法。一个数据库可以呈现为多个服务，而一个服务可以实现为多个数据库实例。服务名称是本身为全局数据库名称的字符串，即，由数据库名称和域名所组成，并在安装或数据库创建期间输入的名称。如果您不确定全局数据库名称为何，您可从初始化参数文件中的 SERVICE_NAMES 参数值获取此名称。

ii. 如果选择 **SID**，请输入用于标识运行中数据库的特定实例的 Oracle 系统标识。

这是识别特定数据库实例的 Oracle 系统标识。

前进到步骤 5d。

2) （可选）TNS

a) 如果已设置 *ORACLE_HOME* 系统环境变量，且已在 `$ORACLE_HOME/network/admin/tnsnames.ora` 文件中定义目标受监视数据库的 TNS 别名，请选择 **TNS** 连接类型。

b) 输入 **TNS 别名**。

前进到步骤 5d。

3) （可选）Advanced

a) 当目标受监视数据库的多个物理节点中有多个 Oracle 实例时，请选择 **Advanced** 连接类型。例如，带有 Real Applications Cluster (RAC) 数据库的 ASM。

b) 输入 **Oracle 连接字符串**。

此属性支持所有 Oracle 网络命名方法，如下所示：

- 以下格式的 SQL Connect URL 字符串：`//host:port/service name`。例如，`//dlsun242:1521/bjava21`。
- Oracle Net 键/值对。例如，

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))
```

- **TNSNAMES** 条目，例如，**inst1**，其中设置了 *TNS_ADMIN* 或 *ORACLE_HOME* 环境变量且配置了配置文件。

前进到步骤 5d。

d) 选中此连接的**使用不同的用户名和密码**，以使用不同于步骤 4a 和步骤 4b 中所设置缺省凭证的凭证。否则，请前进到步骤 5g。

e) 输入此连接的**数据库用户名**。

此用户标识就是代理程序用于访问受监视数据库实例的标识。此用户标识必须对代理程序所需的动态性能视图和表具有 SELECT 特权。

f) 输入**数据库密码**。这是与指定的数据库用户标识相关联的密码。

g) 选择与授予数据库连接凭证的许可权匹配的**角色**。

角色是与连接关联的一组特权。对于已获授予 SYSDBA 系统特权的用户，请指定包括该特权的角色。对于 ASM 实例，请使用 **SYSDBA** 或 **SYSASM** 角色。

h) 如果您从这个代理程序实例监视远程 Oracle 警报日志，请选中**显示远程日志监视选项**，否则请前进到步骤 5k。

i) 输入路径或使用**浏览**以选择 **Oracle 警报日志文件路径**。

用于此数据库连接中远程数据库实例的已映射警报日志文件的绝对路径。代理程序通过读取这些文件监视警报日志。其通常位于 `$ORACLE_BASE/diag/rdbms/`*`DB_NAME`*`/`*`SID`*`/trace/alert_`*`SID`*`.log`。例如，如果 *DB_NAME* 和 *SID* 都是 db11g，而 *ORACLE_BASE* 是 `/home/dbowner/app/oracle`，那么警报日志将位于 `/home/dbowner/app/oracle/diag/rdbms/db11g/db11g/trace/alert_db11g.log`。

Windows 如果 Oracle Database 代理程序运行并通过网络读取警报日志文件，那么远程文件路径必须遵循 Windows 系统的通用命名约定。例如，`\\tivx015\path\alert_orcl.log`。

Windows

要点: 同时输入路径和警报日志文件名。警报日志路径不支持映射的网络驱动器。

Linux AIX 如果 Oracle Database 代理程序在远程服务器上，那么需要本地挂装的文件系统，才能监视其远程警报日志。

Windows 多个文件由分号 (;) 分隔。

Linux AIX 多个文件由冒号 (:) 分隔。

每个文件都使用 `alert_`*`instance`*`.log` 文件名模式与一个数据库实例匹配，如果不匹配，那么会将其忽略。

系统会自动发现本地数据库实例警报日志文件。

j) 选择或输入 **Oracle 警报日志文件字符集**。这是所映射的警报日志文件的代码页。

如果此参数为空白，那么会使用系统的当前语言环境设置，例如：

- ISO8859_1，ISO 8859-1 西欧编码
- UTF-8，UTF-8 Unicode 编码

- GB18030，简体中文 GB18030 编码
- CP950，繁体中文编码
- EUC_JP，日语编码
- EUC_KR，韩语编码

有关所有受支持代码页的完整列表，请参阅 ICU 受支持代码页。

k) 单击**应用**以将此数据库连接实例的设置保存在**数据库连接**部分中。

l) （可选）对新的数据库连接进行测试。

1) 在**数据库连接**部分中，选择新的数据库连接。

2) 单击**测试连接**。

3) 在“**测试连接**”结果窗口中观察结果。

- 成功**测试结果**示例：

```
正在测试连接 config1 ...
成功
```

- 不成功**测试结果**示例：

```
正在测试连接 config1 ...
KBB_RAS1_LOG；将 MAXFILES 设置为 1
ORA-12514：TNS：侦听器当前无法识别连接描述符中所请求的服务
失败
```

m) 单击**下一步**。

6. 阅读“**配置 ITCAM Extended Agent for Oracle Database**”窗口中**摘要**窗格上的信息，然后单击**确定**以完成代理程序实例的配置。
7. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Oracle Database**，然后单击**启动**。

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

通过对提示做出响应来配置代理程序

要在 Linux 和 UNIX 操作系统上配置此代理程序，请运行命令行配置脚本并对提示做出响应。

过程

1. 打开 *install_dir*/bin 目录，其中，*install_dir* 是 Oracle Database 代理程序的安装目录。
2. （可选）要列出任何已配置的现有代理程序实例的名称，请运行以下命令：**./cinfo -o rz**。
3. 要配置 Oracle Database 代理程序，请运行以下命令：**./oracle_database-agent.sh config** *instance_name*。
4. 当出现 `Edit 'Monitoring Agent for Oracle Database' settings` 提示时，请按 **Enter** 键。缺省值为 `Yes`。
5. 要输入 `Default Database Configuration` 信息，请执行下列步骤：

   注: `Default Database Configuration` 节不是数据库连接实例配置。这是一个模板节，用于设置当您从步骤 6 开始添加实际的数据库连接实例配置时，要用作缺省值的内容。

   a) 当出现 `Default Username` 提示时，请输入数据库连接的缺省数据库用户标识，并按 **Enter** 键。

   此用户标识就是代理程序用于访问受监视数据库实例的标识。此用户标识必须对代理程序所需的动态性能视图和表具有 SELECT 特权。

   b) 当出现 `Enter Default Password` 提示时，请输入与指定的缺省数据库用户标识相关联的密码，然后按 **Enter** 键。接着，如果出现提示，请确认密码。

   c) 如果 Oracle 代理程序版本为 8.0，请执行此步骤。

1) 输入 **Oracle JDBC Jar 文件**。这是用来与 Oracle 数据库进行通信的 Oracle JDBC 驱动程序 Jar 文件的完整路径。在代理程序计算机上，必须要有支持 Oracle 代理程序所监视 Oracle 数据库版本的 Oracle Java 数据库连接 (JDBC) 驱动程序。

d) 如果 Oracle 代理程序版本为 6.3.1.10，请执行这些步骤。

1) 当出现 Oracle Home Directory 提示时，如果 Oracle Database 代理程序是安装在受监视的 Oracle 数据库服务器上，请输入 Oracle 主目录，然后按 **Enter** 键。如果 Oracle Database 代理程序不是安装在将要监视的 Oracle 数据库服务器上，请将此设置保留为空，按 **Enter** 键，并执行下一步。如果要清除 Oracle Home Directory 目录值，请按空格键，然后按 **Enter** 键。

注: （可选）要进行本地监视，可将 Oracle Home Directory 和 Oracle Instant Client Installation Directory 保留为空，在此情况下，将使用 *ORACLE_HOME* 系统环境变量。

2) 如果 Oracle Database 代理程序远离受监视的 Oracle 数据库服务器，请输入 Oracle Instant Client Installation Directory 目录，然后按 **Enter** 键。如果已在步骤 第 531 页的『5.d.i』 中设置 Oracle Home Directory，那么将忽略此值。

e) Net Configuration Files Directories 可以保留为空并使用缺省目录。如果 Oracle 代理程序版本为 6.3.1.10，您可输入多个网络配置文件目录，并以 Windows “;” 或 Linux AIX “:” 进行分隔。Oracle 代理程序 V8.0 只支持一个目录。按 **Enter** 键。

此设置包含一个或多个 Oracle 数据库网络配置文件。目录由每个 Oracle 数据库实例的 *TNS_ADMIN* 环境变量进行定义。缺省目录为 Linux AIX $ORACLE_HOME/network/admin 或 Windows %ORACLE_HOME%\NETWORK\ADMIN。如果未配置这个项，那么将使用缺省目录。要禁止使用缺省目录，请将下列代理程序环境变量设置为 false：KRZ_LOAD_ORACLE_NET=false。

f) 选择是否在此工作站上配置缺省动态侦听器，并按 **Enter** 键。

缺省动态侦听器是 (PROTOCOL=TCP)(HOST=localhost)(PORT=1521)。如果在此工作站上配置了缺省动态侦听器，请将这个值设置为 True。

g) 将 定制 SQL 定义文件名保留为空。不使用该属性。

6. 在屏幕上看到下列输出之后，系统会显示 Edit 'Database Connection' settings 提示：

```
Instance Configuration :
Summary :
Database Connection :
```

注: 此步骤是定义实际数据库连接实例的位置。您必须至少添加一个数据库连接实例。这也是您编辑和删除数据库连接实例的位置。如果存在多个数据库连接实例配置，请使用 Next 选项来跳过不需要编辑或删除的实例，直至到达您需要编辑或删除的实例为止。

7. 要添加新的数据库连接，请输入 1，并按 **Enter** 键。
8. 要输入数据库连接信息，请执行以下步骤：

a) 当出现 Database Connection Name 提示时，请输入数据库连接的别名，并按 **Enter** 键。

此别名用来代表数据库连接，可以随您选择，但有下列限制。只有字母、阿拉伯数字、下划线字符和减号可用于连接名称中。连接名称的最大长度为 25 个字符。

b) 当出现 Connection Type 提示时，请选择下列其中一种连接类型：

1) （可选）Basic

缺省连接类型（也就是最常用的连接类型）是 **Basic**。如果您不确定需要何种连接类型，建议您选择此连接类型。

a) 当目标受监视数据库是单个实例（例如，标准文件系统实例或单个 ASM 实例）时，请选择**基本**连接类型。

b) 当出现 Hostname 提示时，请输入 Oracle 数据库的主机名或 IP 地址，并按 **Enter** 键。

c) 当出现 Port 提示时，请输入端口号，并按 **Enter** 键。

d) 输入随后两项设置中的一个。即，Service Name 或 SID。

i. （可选）当出现 Service Name 提示时，请输入作为数据库逻辑表示的服务名称（即，本身是全局数据库服务名称的字符串），按 **Enter** 键，然后前进到步骤 8c。

服务名称是数据库的逻辑表示，这是向客户端呈现数据库的方法。一个数据库可以呈现为多个服务，而一个服务可以实现为多个数据库实例。服务名称是本身为全局数据库名称的字符串，即，由数据库名称和域名所组成，并在安装或数据库创建期间输入的名称。如果您不确定全局数据库名称为何，您可从初始化参数文件中的 SERVICE_NAMES 参数值获取此名称。如果您在步骤 第 532 页的『8.b.i.4.b』 中设置 SID，那么此参数可保留为空白。

ii. （可选）当出现 SID 提示时，请输入用于标识运行中数据库的特定实例的 Oracle 系统标识，按 **Enter** 键，然后前进到步骤 8c。

此参数是识别特定数据库实例的 Oracle 系统标识。如果已在步骤 第 531 页的『8.b.i.4.a』 中定义 Service Name，那么可将这个项保留为空白。

2) （可选）TNS

a) 如果已设置 *ORACLE_HOME* 系统环境变量，且已在 $ORACLE_HOME/network/admin/tnsnames.ora 文件中定义目标受监视数据库的 TNS 别名，请选择 **TNS** 连接类型。

b) 输入 TNS 别名，按 **Enter** 键，然后前进到步骤 8c。

3) （可选）Advanced

a) 当目标受监视数据库的多个物理节点中有多个 Oracle 实例时，请选择 **Advanced** 连接类型。例如，带有 Real Applications Cluster (RAC) 数据库的 ASM。

b) 输入 Oracle 连接字符串，按 **Enter** 键，然后前进到步骤 8c。

此属性支持所有 Oracle 网络命名方法，如下所示：

- 以下格式的 SQL Connect URL 字符串：//host:port/service name。例如，//dlsun242:1521/bjava21。
- Oracle Net 键/值对。例如，

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))
```

- **TNSNAMES** 条目，例如，**inst1**，其中设置了 *TNS_ADMIN* 或 *ORACLE_HOME* 环境变量且配置了配置文件。

注: 命令行配置期间所显示的描述可能会在冒号 (\:) 和等号 (\=) 之前附加反斜杠。在连接字符串中，请勿输入反斜杠。在描述中显示反斜杠，是为了避开将等号解释为命令组成部分的一般行为，而改为将其仅仅解释为文本。

c) 前进到步骤 8c。

c) 当出现 Database Username 提示时，请输入连接的数据库用户标识，并按 **Enter** 键。

对于标准文件系统实例，此用户标识必须对代理程序所需的动态性能视图和表具有 SELECT 特权。

针对 ASM 实例，使用具有 **SYSDBA** 或 **SYSASM** 角色的帐户。例如，sys 帐户。

d) 当出现 Enter Database Password 提示时，请输入与指定的数据库用户标识关联的密码。

e) 当出现 Role 提示时，请选择与授予所指定用户标识的许可权匹配的角色，并按 **Enter** 键。

角色是与连接关联的一组特权。对于已获授予 SYSDBA 系统特权的用户，请指定包括该特权的角色。

对于 ASM 实例，请使用 **SYSDBA** 或 **SYSASM** 角色。

f) 当出现 Oracle Alert Log File Paths (including alert log file name) 提示时，请输入警报日志路径，并按 **Enter** 键。

此参数是指定此数据库连接中，远程数据库实例的映射警报日志文件的任何绝对文件路径。代理程序通过读取这些文件监视警报日志。其通常位于 $ORACLE_BASE/diag/rdbms/*DB_NAME*/*SID*/trace/alert_*SID*.log。例如，如果 *DB_NAME* 和 *SID* 都是 db11g，而 *ORACLE_BASE* 是 /home/dbowner/app/oracle，那么警报日志将位于 /home/dbowner/app/oracle/diag/rdbms/db11g/db11g/trace/alert_db11g.log。

Windows 如果 Oracle Database 代理程序运行并通过网络读取警报日志文件，那么远程文件路径必须遵循 Windows 系统的通用命名约定。例如，`\\tivx015\path\alert_orcl.log`。

要点: 同时输入路径和警报日志文件名。警报日志路径不支持映射的网络驱动器。

Linux AIX 如果 Oracle Database 代理程序运行，那么远程警报日志需要本地安装的文件系统。

Windows 多个文件由分号 (;) 分隔。

Linux AIX 多个文件由冒号 (:) 分隔。

每个文件都使用 `alert_`*instance*`.log` 文件名模式与一个数据库实例匹配，如果不匹配，那么会将其忽略。

本地数据库实例警报日志文件可以自动发现。

g) 当出现 **Oracle Alert Log File Charset** 提示时，请输入映射的警报日志文件的代码页，并按 **Enter** 键。

如果此参数为空白，那么会使用系统的当前语言环境设置，例如:

- ISO8859_1，ISO 8859-1 西欧编码
- UTF-8，UTF-8 Unicode 编码
- GB18030，简体中文 GB18030 编码
- CP950，繁体中文编码
- EUC_JP，日语编码
- EUC_KR，韩语编码

有关所有受支持代码页的完整列表，请参阅 ICU 受支持代码页。

9. 当再次出现 `Edit 'Database Connection' settings` 提示时，您会看到步骤 8a 中设置的数据库连接名称。您可再次进行编辑，或将其删除。如果您已配置多个数据库连接实例，请使用**下一步**来逐个处理这些实例。
10. （可选）要添加另一个数据库连接，以使用此代理程序实例来监视多个数据库实例，请输入 1，按 **Enter** 键，然后回到步骤 8。
11. 完成修改数据库连接之后，请输入 5，然后按 **Enter** 键以退出配置过程。
12. 要启动代理程序，请输入:
    install_dir`/bin/oracle_database-agent.sh start` *instance_name*.

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

1. 在文本编辑器中打开 `oracle_silent_config.txt` 文件。
   - Linux AIX *install_dir*`/samples/oracle_database_silent_config.txt`。

- Windows *install_dir*\samples\oracle_database_silent_config.txt

2. 对于 **Default Username**，请输入为此代理程序实例创建的数据库连接的缺省数据库用户的名称。例如，**KRZ_CONN_USERID=**user1。

   注: 此用户必须具有足够的特权，以完成此代理程序在连接到数据库时所执行的任务，例如，查询表。
3. 对于 **Default Password**，必须输入与指定的缺省数据库用户相关联的密码。例如，**KRZ_CONN_PASSWORD=**Password。
4. 如果 Oracle 代理程序版本为 8.0，请执行此步骤。
   a) 输入 **Oracle JDBC Jar 文件**。这是用来与 Oracle 数据库进行通信的 Oracle JDBC 驱动程序 Jar 文件的完整路径。

   在代理程序计算机上，必须要有支持 Oracle 代理程序所监视 Oracle 数据库版本的 Oracle Java 数据库连接 (JDBC) 驱动程序。
5. 如果 Oracle 代理程序版本为 6.3.1.10，请执行这些步骤。
   a) 如果 Oracle Database 代理程序是安装在受监视的 Oracle 数据库服务器上，请输入 Oracle 主目录。例如，**KRZ_ORACLE_HOME=**home_path。

   注: 对于可选参数（例如这个参数），请移除前导井号 (#) 以使用这些参数。

   如果 Oracle Database 代理程序不是安装在将要监视的 Oracle 数据库服务器上，请将此设置保留为空，并完成下一步。

   注: （可选）要进行本地监视，可将 Oracle Home Directory 和 Oracle Instant Client Installation Directory 保留为空（在静默配置文本文件中参数行的第一个位置，使用井号 (#) 注释掉该行），在此情况下，将使用 *ORACLE_HOME* 系统环境变量。

   b) 如果 Oracle Database 代理程序远离受监视的 Oracle 数据库服务器，请输入 Oracle Instant Client Installation Directory 目录。如果您在上一步已输入 Oracle Home Directory 目录，那么将忽略这个值。

   - Windows 定义 Oracle 调用接口 (OCI) 库文件所在的 **Oracle 主目录**的完整文件夹路径。如果 oci.dll 文件的完整路径为 C:\instantclient_10_2\oci.dll，那么必须定义此 C:\instantclient_10_2 路径。例如，**KRZ_INSTANT_CLIENT_LIBPATH=**C:\instantclient_10_2
   - AIX 定义 Oracle 调用接口 (OCI) 库文件所在的 **Oracle 主目录**的完整文件夹路径。如果 libocci.so.10.1 文件的完整路径为 /home/tivoli/oci/libocci.so.10.1，那么必须定义此 /home/tivoli/oci 路径。例如，**KRZ_INSTANT_CLIENT_LIBPATH=**/home/tivoli/oci

6. Net Configuration Files Directories 可以保留为空并使用缺省目录。Oracle Database 代理程序使用此文件路径以获取 tnsnames.ora 文件。此目录由每个 Oracle 数据库实例的 *TNS_ADMIN* 环境变量进行定义。缺省目录为 Linux AIX $ORACLE_HOME/network/admin 或 Windows %ORACLE_HOME%\NETWORK\ADMIN。如果您输入具有多个网络配置文件目录的该设置，请使用 “Windows ;” 或 “Linux AIX :” 来分隔目录。

   如果要从远端监视 Oracle 数据库，您可将网络配置文件从远程系统复制到安装代理程序的系统上。您还可以将远程系统上网络配置文件的内容，合并到安装代理程序的系统上的网络配置文件中。
7. 针对**动态侦听器**，请检查是否配置了缺省动态侦听器。缺省动态侦听器为 (PROTOCOL=TCP)(HOST=localhost)(PORT=1521)。如果配置了缺省动态侦听器，请将此值设置为 TRUE，如下所示：**KRZ_DYNAMIC_LISTENER=**TRUE。

   有效值为 TRUE 和 FALSE。
8. 将 定制 SQL 定义文件名保留为空。不使用该属性。
9. 从这里开始，定义实际的数据库连接实例。您必须至少添加一个数据库连接实例。在 oracle_silent_config.txt 中，提供了一个实例的条目，其实例名称为 *config1*。如果您更改实例名称，请务必更改所有引用。

此别名用来代表数据库连接，可以随您选择，但有下列限制。只有字母、阿拉伯数字、下划线字符和减号可用于连接名称中。连接名称的最大长度为 25 个字符。

10. 针对**连接类型**，指定以下某种连接类型：**基本**、**TNS** 或**高级**。例如，**KRZ_CONN_TYPE.config1=**Basic。
11. 针对上一步中选择的连接类型，指定必需参数：

    基本

    - 针对**主机名**，指定 Oracle 数据库的主机名或 IP 地址，例如：**KRZ_CONN_HOST.config1=** hostname。
    - 针对**端口**，指定 Oracle 数据库的侦听器端口，例如：**#KRZ_CONN_PORT.config1=** 1521。
    - 针对**服务名称**，通过将字符串用于全局数据库名称来指定数据库的逻辑表示，例如：**KRZ_CONN_SERVICE.config1=** orcl。

      要点: 如果未定义服务名称，那么必须指定 Oracle 系统标识 (SID)。

      针对 **Oracle 系统标识 (SID)**，指定用于标识运行中数据库的特定实例的 SID，例如：**KRZ_CONN_SID.config1=** sid。

    TNS

    针对 **TNS 别名**，从 tnsnames.ora 文件指定网络别名。例如，**KRZ_CONN_TNS.config1=** tnsalias。

    高级

    针对 **Oracle 连接字符串**，为 OCI 指定数据库连接字符串。例如，**KRZ_CONN_STR.config1=** //host:port/service

    此字符串支持所有 Oracle 网络命名方法，如此处所示。

    - 针对 SQL Connect URL 字符串：

      ```
      //host:[port][/service name]
      ```

    - 针对 Oracle Net 键/值对：

      ```
      "(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=1521))
      (CONNECT_DATA=(SERVICE_NAME=bjava21)))"
      ```

    此字符串还支持 **TNSNAMES** 条目，例如，**inst1**，其中，设置了 *TNS_ADMIN* 或 *ORACLE_HOME* 环境变量，并配置了配置文件。

    要点: 此属性仅适用于高级连接类型。
12. 针对**数据库用户名**，可以为连接指定数据库用户的名称，例如：**KRZ_CONN_USERID=**UserID。

    此用户必须具有足够的特权，以完成代理程序在连接到数据库时需要完成的任务，例如，创建、编辑和删除表。

    如果此字段为空，那么代理程序会使用缺省数据库配置部分中的缺省用户名。如果未配置**数据库用户名**，那么会将缺省用户名用于此连接。
13. 针对**数据库密码**，可以指定与指定数据库用户关联的密码，例如：**KRZ_CONN_PASSWORD=**Passsword。

    如果此字段为空，那么代理程序会使用缺省数据库配置部分中的缺省密码。如果未配置**数据库密码**，那么会将缺省密码用于此连接。
14. 针对**角色**，可以指定与连接关联的一组特权，例如：**KRZ_CONN_MODE.config1=**DEFAULT。

    有效值包括 *SYSDBA*、*SYSOPER*、*SYSASM* 和 *DEFAULT*。

    对于已被授予 *SYSDBA* 系统特权的用户，您可以指定包含该特权的连接。如果未定义此项，那么可以将 *DEFAULT* 角色分配给该用户。
15. 针对 **Oracle 警报日志文件路径**，当包含警报日志文件名时，可以为此数据库连接中的远程数据库实例指定已映射警报日志文件的绝对文件路径。例如，**KRZ_LOG_PATHS.config1=**AlertLogPath。

Windows 使用分号 (;) 来分隔多个文件。

Linux AIX 使用冒号 (:) 来分隔多个文件。

每个文件按 `alert_instance.log` 文件名模式匹配到数据库实例。或者，如果不匹配，会将其忽略。

将自动发现本地数据库实例警报日志文件。

如果未配置 **Oracle 警报日志文件路径**，那么警报日志不可用。

16. 针对 **Oracle 警报日志文件字符集**，可以指定已映射警报日志文件的代码页。例如，**KRZ_LOG_CHARSET.config1=**CharSet

    如果此字段为空，那么会使用系统的当前语言环境设置，如此处所示：

    ```
    ISO8859_1：ISO 8859-1 西欧编码
    UTF-8：UTF-8 Unicode 编码
    GB18030：简体中文 GB18030 编码
    CP950：繁体中文编码
    EUC_JP：日语编码
    ```

17. 保存并关闭 `oracle_database_silent_config.txt` 文件。然后输入：
    install_dir/bin/oracle_database-agent.sh config *instance_name install_dir*/samples/oracle_database_silent_config.txt
    其中，*instance_name* 是要对实例指定的名称。
18. 要启动代理程序，请输入：
    install_dir/bin/oracle_database-agent.sh start *instance_name*.

下一步做什么

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

为 Oracle 数据库代理程序用户授予特权

安装代理程序后，必须为 Oracle Database 代理程序 使用的 Oracle 用户帐户授予特权。

您可以为以下用户授予特权：

· 标准文件系统（非 ASM）实例用户

· 具有 RAC 的 ASM 的实例非 SYS 用户

为用户授予标准文件系统实例的特权

对于标准文件系统实例，Oracle Database 代理程序使用的 Oracle 用户标识必须对代理程序所需的动态性能视图、表和数据字典视图具有选择特权。它还必须具有运行某些数据库命令所必需的其他 Oracle 对象和系统特权。

过程

1. （可选）如 Oracle 数据库用户标识不存在，请通过使用 Oracle 工具并运行以下命令来创建此标识：
   `create user` *UserName* `identified by` *Password*
2. 通过运行 Oracle Database 代理程序随附的 **krzgrant.sql** 脚本，向所创建的 Oracle 用户标识授予对动态性能视图、表和数据字典视图的选择特权。必须在配置代理程序之前完成此步骤。有关如何定制和运行 **krzgrant.sql** 脚本的指示信息，请参阅第 537 页的『定制 krzgrant.sql 脚本』和第 537 页的『运行 krzgrant.sql 脚本』。

   注: 对动态性能视图、表和数据字典视图的选择特权依赖于特定应用程序环境中 Oracle 数据库的功能。针对 Oracle 数据库用户标识，您只能授予对 Oracle Database 代理程序使用的动态性能视图、表和数据字典视图的授权 Oracle 特权。
3. 通过使用 Oracle 工具为 Oracle Database 代理程序使用的 Oracle 用户标识授予其他 Oracle 对象特权和系统特权。

定制 *krzgrant.sql* 脚本

如果在 **krzgrant.sql** 脚本中，您不想允许对某些动态性能视图、表和数据字典视图的 Oracle 授权选择特权，那么可以在运行 **krzgrant.sql** 脚本之前，先对其进行定制。

注: 在代理程序启动时，代理程序实例会检查 **krzgrant.sql** 脚本中的所有缺省特权，并报告缺少特权的代理程序事件。您可以使用以下变量设置来禁用特权检查：KRZ_CHECK_ORACLE_PRIVILEGE=FALSE。GUI 配置的测试连接步骤会检查在 krzgrant.sql 文件中定义的所有 Oracle 特权。如果确认 Oracle 用户具有正确的特权，请忽略测试连接步骤中检查特权失败的情况。

在纯文本编辑器中编辑 krzgrant.sql 文件以在授权语句的开头移除或添加“--”前缀，从而跳过对未经授权的 Oracle 表或视图的授权执行。

例如，将以下行：

```
    execute immediate 'grant select on DBA_HIST_SNAPSHOT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQLSTAT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQLTEXT to '||userName;
    execute immediate 'grant select on DBA_HIST_SQL_PLAN to '||userName;
    execute immediate 'grant select on DBA_HIST_SYSMETRIC_SUMMARY to '||userName;
```

更改为以下行：

```
--    execute immediate 'grant select on DBA_HIST_SNAPSHOT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQLSTAT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQLTEXT to '||userName;
--    execute immediate 'grant select on DBA_HIST_SQL_PLAN to '||userName;
--    execute immediate 'grant select on DBA_HIST_SYSMETRIC_SUMMARY to '||userName;
```

为非 SYS 用户授予 ASM 实例的特权

必须连接到将 SYSDBA 和 SYSASM 角色用于用户的 ASM 实例。如果不希望使用 SYS 帐户连接到 ASM 实例，请创建一个用户帐户，并为其授予 SYSDBA 和 SYSASM 角色。

过程

1. 运行以下命令创建用户帐户并授予角色：
   - 使用 SYSASM 角色登录到 ASM 数据库来为代理程序创建新用户，然后授予 SYSDBA 角色或 SYSASM 角色：
     a. `create user UserName identified by Password`
     b. `grant sysdba to UserName`

     或者

     `grant sysasm to UserName`
2. 在配置窗口中创建 ASM 连接时，请指定 *UserName* 用户以及 SYSDBA 或 SYSASM 角色。

   注: 如果选择 SYSASM 角色来访问 ASM 数据库，那么必须通过使用 Oracle 主目录或 Oracle 即时客户机连接到 Oracle 数据库来配置代理程序实例。

运行 *krzgrant.sql* 脚本

开始之前

- 如果不运行 **krzgrant.sql** 脚本，将在代理程序事件工作空间中引发事件。
- 要完成安装过程，请参阅 第 101 页的『第 6 章 安装代理程序』。

安装后，您可以在以下目录中找到 **krzgrant.sql** 脚本：

- Windows *install_dir*\TMAITM6_X64
- Linux AIX *install_dir*/*architecture*/rz/bin

其中：

install_dir
Oracle Database 代理程序的安装目录。

architecture
IBM Application Performance Management 或 Cloud APM 系统体系结构标识。例如，lx8266 代表 Linux Intel V2.6（64 位）。要获取体系结构代码的完整列表，请参阅 *install_dir*/registry/archdsc.tbl 文件。

krzgrant.sql 脚本的用法如下：`krzgrant.sql` *user_ID temporary_directory*

其中：

user_ID
Oracle 用户的标识。在运行此 SQL 文件前必须创建此用户标识。示例值：*tivoli*。

temporary_directory
包含 **krzgrant.sql** 脚本的 krzagent.log 输出文件的临时目录的名称。在运行此 SQL 脚本前该目录必须存在。示例值：install_dir/tmp。

您必须具有此临时目录的 Oracle 数据库管理员 (DBA) 授权角色和写权限才能执行以下过程。

过程

1. 从命令行运行以下命令来设置环境变量。
   - Windows

     ```
     SET ORACLE_SID= sid
     SET ORACLE_HOME= home
     ```

   - Linux AIX

     ```
     ORACLE_SID = sid
     export ORACLE_SID
     ORACLE_HOME = home
     export ORACLE_HOME
     ```

   其中：

   sid
   区分大小写的 Oracle 系统标识。

   home
   受监视 Oracle 实例的主目录。
2. 从设置环境变量的命令行窗口启动 Oracle SQL Plus 或启动用于发出 SQL 语句的其他工具。
3. 以具有 Oracle DBA 特权的用户身份登录到 Oracle 数据库。
4. 转至包含 **krzgrant.sql** 脚本的目录并运行以下命令来授予选择特权：

   ```
   @krzgrant.sql user_ID temporary_directory
   ```

   输出结果将记录到临时目录中的 krzagent.log 文件。此日志会记录 Oracle Database 代理程序将对其授予选择特权的视图和表。

   成功授予权限后，可以配置并启动 Oracle Database 代理程序。

配置操作系统监视

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS 和 Monitoring Agent for Windows OS 代理程序自动进行配置。您可以针对操作系统代理程序配置日志文件监视，从而可监视应用程序日志文件。您可以非 root 用户身份运行操作系统代理程序。并且，还有其他配置选项可供 Linux 操作系统代理程序使用。

以非 root 用户身份运行操作系统代理程序

您可以非 root 用户身份运行 Monitoring Agent for Windows OS、Monitoring Agent for UNIX OS 和 Monitoring Agent for Linux OS。

要以非 root 用户身份运行 Windows OS 代理程序，请参阅第 539 页的『以非 root 用户身份使用 Monitoring Agent for Windows OS』。

要以非 root 用户身份运行 Monitoring Agent for UNIX OS 和 Monitoring Agent for Linux OS 代理程序，请参阅第 859 页的『以非 root 用户身份启动代理程序』。

限制:

以非 root 用户身份运行时，代理程序无法访问 `/proc/pid/status`，因此无法报告以下属性：

- - 用户 CPU 时间 (UNIXPS.USERTIME)
- - 系统 CPU 时间 (UNIXPS.SYSTEMTIM)
- - 总计 CPU 时间 (UNIXPS.TOTALTIME)
- - 线程计数 (UNIXPS.THREADCNT)
- - 子用户 CPU 时间 (UNIXPS.CHILDUTIME)
- - 子系统 CPU 时间 (UNIXPS.CHILDSTIME)
- - 总计子 CPU 时间 (UNIXPS.CHILDTIME)
- - 等待 CPU 时间 (UNIXPS.WAITCPUTIM)
- - 终端 (UNIXPS.USERTTY)

这些属性在 Cloud APM 控制台中不可见，但可用于创建阈值。

以非 root 用户身份使用 Monitoring Agent for Windows OS

您可以非 root 用户身份运行 Windows OS 代理程序。但是，一些功能不可用。

非 root 用户身份运行 Windows OS 代理程序 时，如果某些功能由管理员帐户唯一拥有，那么这些功能在以下属性组中不可用：

- 注册表
- 文件趋势
- 文件更改

由于需要管理员权限来安装新代理程序，因此无法远程部署其他代理程序。

对于 Agent Management Services，看守程序无法停止或启动其无权停止或启动的任何代理程序。

要创建非 root 用户，请创建新的限制（非 root）用户并为新用户设置注册表许可权，如以下示例中所示：

- 对 `HKEY_LOCAL_MACHINE\SOFTWARE\Candle` 的完整访问权
- 对 `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib` 的读访问权

启动 Monitoring Agent for Windows OS（主服务）的用户必须有权管理 Monitoring Agent for Windows OS（看守程序服务）。启动 Monitoring Agent for Windows OS（看守程序服务）的用户也必须有权管理由代理程序管理服务管理的任何服务，包括 Monitoring Agent for Windows OS（主服务）。要授予用户在 Windows 中管理系统服务的权限，请使用安全模板和组策略，或编辑 `Subinacl.exe` 文件。有关更多信息，请参阅以下 Microsoft 文档：http://support.microsoft.com/kb/325349(http://support.microsoft.com/kb/325349)。

以下示例说明如何通过使用安全模板授予用户权限来管理系统服务：

1. 单击**启动 > 运行**，在“打开”框中输入 `mmc`，然后单击**确定**。
2. 在**文件**菜单上，单击**添加/移除插入**。
3. 单击**添加 > 安全配置和分析**，然后再次单击**添加**。
4. 单击**关闭**，然后单击**确定**。

5. 在控制台树中，右键单击**安全配置和分析**，然后单击**打开数据库**。
6. 指定数据库的名称和位置，然后单击**打开**。
7. 在显示的“**导入模板**”对话框中，单击要导入的安全模板，然后单击**打开**。
8. 在控制台树中，右键单击**安全配置和分析**，然后单击**立即分析计算机**。
9. 在显示的“**执行分析**”对话框中，接受在“错误日志文件路径”框中显示的日志文件的缺省路径。或者，指定所需位置。单击**确定**。
10. 分析完成后，配置服务许可权，如下所示：
    a. 在控制台树中，单击**系统服务**。
    b. 在右窗格中，双击 Monitoring Agent for Windows OS（主服务）。
    c. 选中**在数据库中定义此策略**复选框，然后单击**编辑安全性**。
    d. 要为新用户或组配置许可权，请单击**添加**。
    e. 在“**选择用户、计算机或组**”对话框中，输入要为其设置许可权的用户或组的名称，然后单击**确定**。在“**用户或组的许可权**”列表中，选中**开始**旁的**允许**复选框。缺省情况下，已选择停止和暂停许可权，这样用户或组可启动、停止或暂停服务。
    f. 单击**确定**两次。
11. 重复步骤 10 以配置 Monitoring Agent for Windows OS（看守程序服务）的服务许可权。
12. 要将新安全设置应用到本地计算机，请右键单击**安全配置和分析**，然后单击**立即配置计算机**。

注: 还可使用 Secedit 命令行工具来配置和分析系统安全性。有关 Secedit 的更多信息，请单击**开始 > 运行**，输入 cmd，然后单击**确定**。在命令提示符处，输入 `secedit /?`，然后按 **ENTER**。使用此方法应用设置时，会重新应用模板中的所有设置。此方法可能会覆盖先前配置的其他文件、注册表或服务许可权。

以下示例显示如何使用 Windows 服务控制台将 Monitoring Agent for Windows OS 和看守程序服务设置为以非 root 用户身份登录：

1. 单击**开始 > 运行**，输入 `services.msc`，然后单击**确定**。
2. 选择 **Monitoring Agent for Windows OS - 主要**。
3. 右键单击**属性**。
4. 验证启动类型是否为“自动”。
5. 选择**登录**选项卡，然后选择**作为“此帐户”登录**，并提供标识和密码。单击**确定**。
6. 选择 **Monitoring Agent for Windows OS - 看守程序**。
7. 右键单击**属性**。
8. 验证启动类型是否为“手动”。
9. 选择**登录**选项卡，然后选择**作为“此帐户”登录**，并提供标识和密码。单击**确定**。

配置操作系统代理程序日志文件监视

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS 和 Monitoring Agent for Windows OS 代理程序自动进行配置。但是，您可以针对操作系统代理程序配置日志文件监视，从而可监视应用程序日志文件。

在代理程序过滤日志数据后，数据以日志事件的形式发送到 Cloud APM 控制台。

为操作系统代理程序添加或移除日志文件监视配置

您可以为操作系统代理程序添加日志文件监视配置，从而使操作系统代理程序可过滤日志文件数据。随后，您还可以移除操作系统代理程序的日志文件监视配置（如果需要）。

开始之前

操作系统代理程序现在包含样本 regex1.conf 文件和 regex1.fmt 文件，您可在配置 .conf 和 .fmt 文件之前查看这些文件。该文件位于此处：

- 在 UNIX/LINUX 上：`<install_dir>/samples/logfile-monitoring`
- 在 Windows 上：`<install_dir\samples\logfile-monitoring`

使用文本编辑器，可创建配置 .conf 文件和格式 .fmt 文件。有关这些文件内容的更多信息，请参阅第 545 页的『配置文件』和第 552 页的『格式文件』。必须确保在访问 Performance Management Console 的系统上保存这些文件，以便可以将这些文件上载到 Cloud APM 服务器。

关于此任务

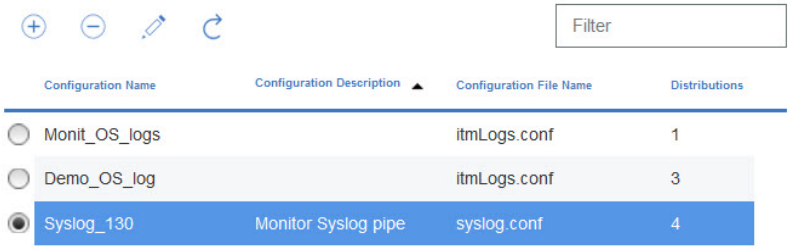
要启用操作系统代理程序来监视日志文件，则必须上载配置文件和格式文件并指定应用此配置的操作系统代理程序。操作系统代理程序下载 .conf 和 .fmt 文件，并且代理程序监视在配置中指定的日志文件。

过程

为操作系统代理程序添加日志文件监视配置

1. 单击**系统配置 > 代理程序配置**。
2. 根据想要监视日志文件的系统，单击 **Unix 操作系统**、**Linux 操作系统**或 **Windows 操作系统**选项卡。
3. 要创建新配置，请单击 (+) 图表以打开“**新建日志文件配置**”窗口。输入配置的名称以及配置的描述。
4. 要查看 .conf 和 .fmt 文件的内容，请单击**查看**。
5. 要使用 Cloud APM 服务器上载配置，请从打开 Performance Management Console 的相同系统选择 .conf 文件和 .fmt 文件，然后单击**完成**。
6. 在操作系统代理程序选项卡上，选择上载的配置。

   要点: 将分发到代理程序的 .conf 和 .fmt 文件重命名为定义的配置名称。



7. 要部署配置，请在**日志配置分发列表**表中，选择想要部署配置的代理程序，然后单击**应用更改**。

为操作系统代理程序移除日志文件监视配置

8. 选择配置名称。
9. 清空受管系统，然后单击**应用更改**。

   要点:

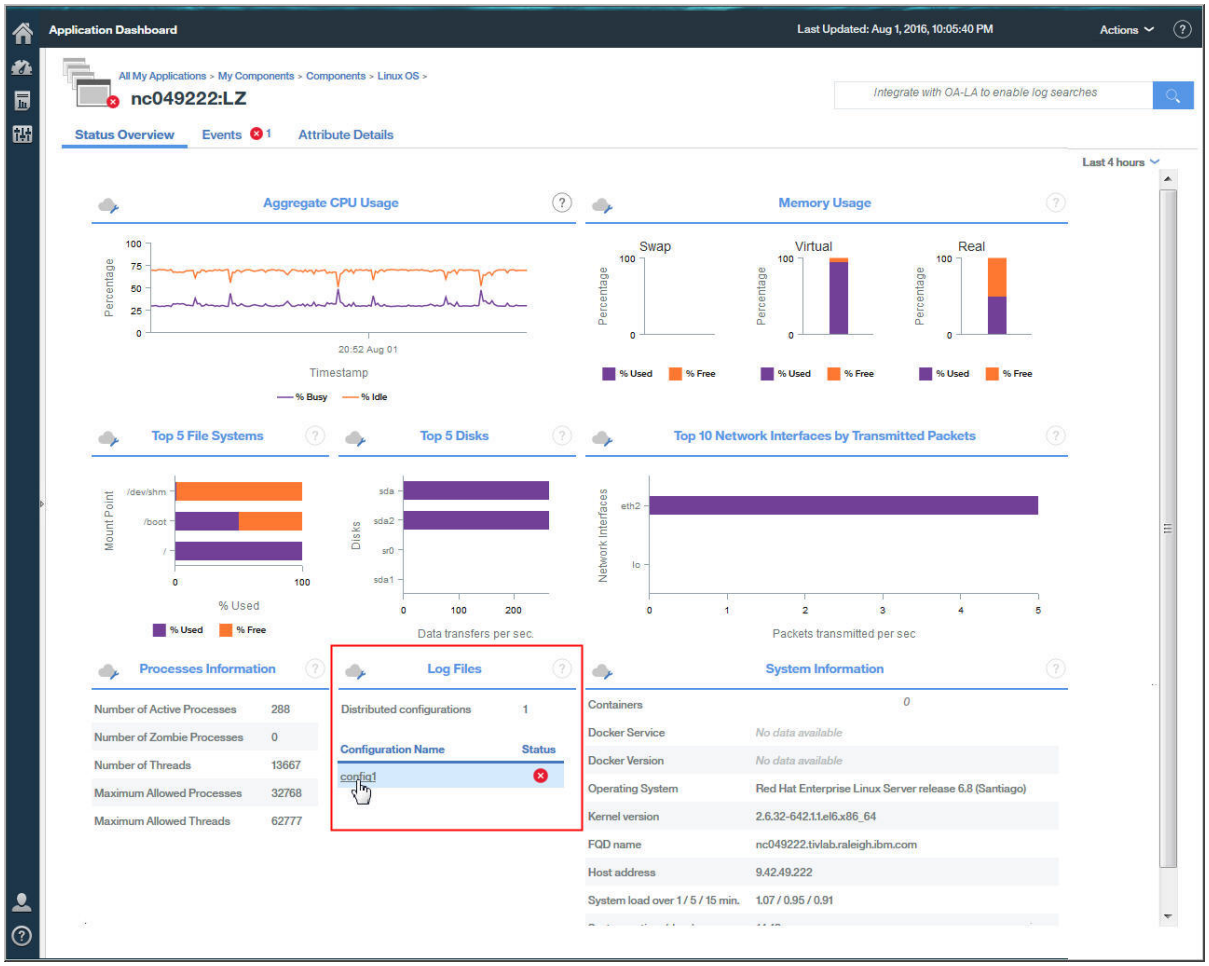
   在移除日志监视配置后，日志文件监视资源保留并且保持联机，直至重新启动操作系统代理程序。在**移除脱机系统延迟**选项中指定的时间后，清除脱机日志文件监视资源。

查看日志文件监视内容

您可以查看已部署到监视器日志文件的操作系统代理程序的日志文件监视配置。

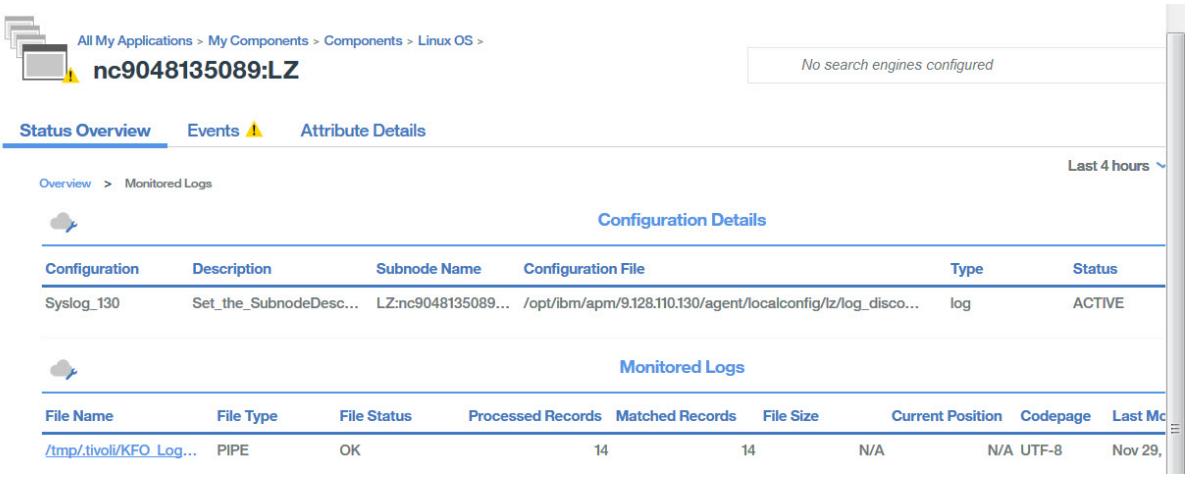
过程

1. 单击**性能 > 应用程序性能仪表板**，并选择包含部署了日志文件监视配置的操作系统代理程序的应用程序。
2. 向下钻取到操作系统代理程序仪表板，在“日志文件”窗口小部件中，单击概要文件以查看分发的日志监视配置以及监视的日志。

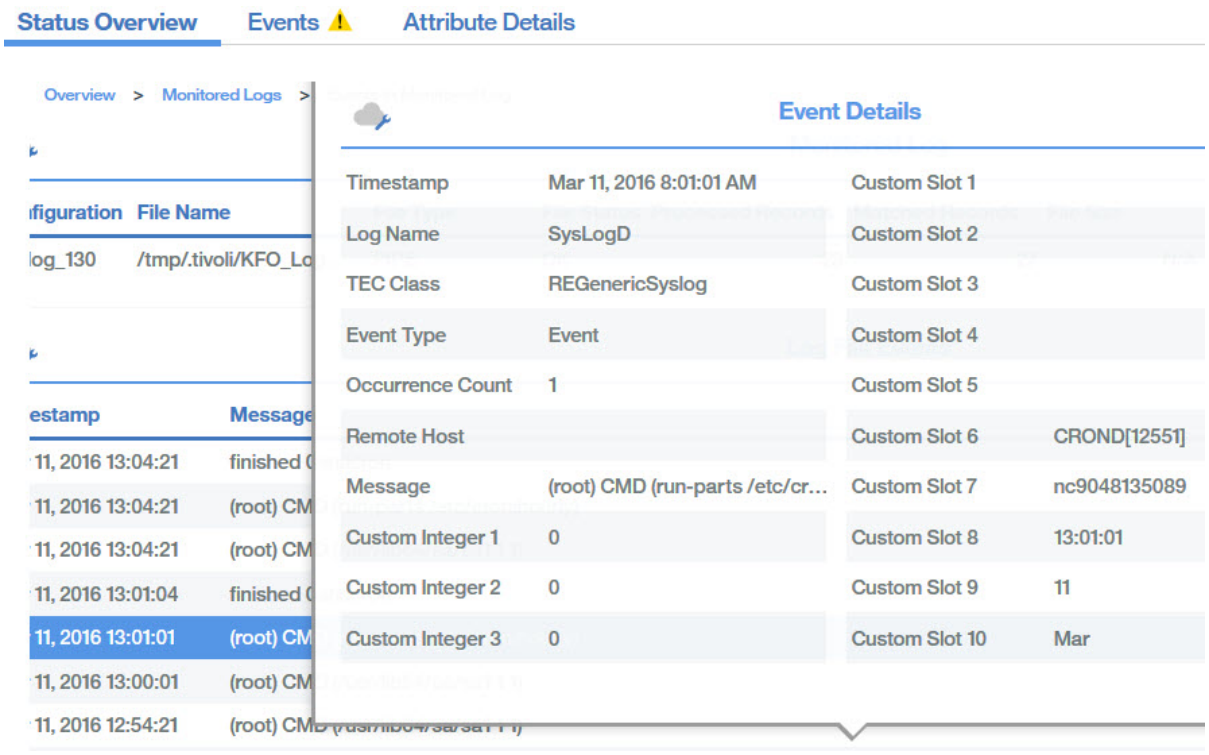


配置详细信息包含配置名称、描述、子节点、配置文件、状态和错误代码。

3. 单击日志文件名以查看与日志文件关联的所有日志文件事件。



4. 单击事件可查看事件详细信息，例如，在格式文件中定义的所有字段。



显示日志文件监视事件

配置操作系统代理程序以监视应用程序日志文件之后，您可创建阈值，以针对您想要获得相应提醒的日志文件条件引发警报。

过程

1. 在“导航栏”上，单击 **系统配置 > 阈值管理器**。
2. 选择**数据源类型**的目标操作系统。
3. 单击 “添加”以创建新阈值。
4. 设置超过此阈值的事件的严重性。
5. 选择要创建阈值的数据集。下列数据集适用于日志文件监视：
   - *Kpp* 日志文件正则表达式统计信息
   - *Kpp* 日志文件状态
   - *Kpp* 日志文件概要事件
6. 单击 **添加**以添加条件。在**添加条件**框中，选择属性和运算符，然后输入阈值。

   如有需要，您可重复此步骤，将更多条件添加到阈值。
7. 在“组分配”部分中，选择要将阈值分配到的资源组。
8. 单击**保存**。
9. 在“导航栏”上，单击 **系统配置 > 高级配置**。
10. 在 **UI 集成**类别中，将**启用子节点事件**值设置为 True。
11. 单击**保存**。

结果

当指定的条件变为 true 时，触发该警报的日志文件事件就会显示在“事件”选项卡中。

日志文件监视环境变量

您可以在操作系统代理程序环境文件中设置日志文件监视的环境变量。

设置以下环境变量并将 K*PC* 替换为操作系统代理程序代码，其中，*PC* 是两字符代理程序代码，例如，klz 是 Linux 操作系统代理程序的代码。

K*PC*_FCP_LOG
此变量仅在 *install_dir*/config/.*pc*.environment 文件中可用。缺省值为 True，并使用其来启用或禁用日志监视功能。

K*PC*_FCP_LOG_PROCESS_MAX_CPU_PCT
此设置是 1 分钟时间间隔内代理程序使用的所有系统 CPU 最大允许百分比。有效值为 5 - 100。缺省值为 100。此设置与 CPU 调速功能部件相关联。如果指定小于 5 的值，那么将使用最小值 5。

K*PC*_FCP_LOG_PROCESS_PRIORITY_CLASS
此设置是流程的操作系统调度程序优先级。A 最低，C 是操作系统缺省值，而 F 是最高优先级。设置为以下其中一个值：A、B、C、D、E、F。在 .conf 文件中指定的任何值将取代这些值。

K*PC*_FCP_LOG_SEND_EVENTS
缺省设置为 True，并且它由 OS 代理程序用于将事件发送到 Cloud APM 服务器。

K*PC*_FCP_LOG_SEND_EIF_EVENTS
缺省设置为 True。如果此选项设置为 Yes，那么代理程序将事件数据发送到 Cloud APM 服务器或任何 EIF 接收器，如 OMNIbus EIF 探测器。如果此选项设置为 No，代理程序不会发送事件数据。此选项的设置为全局，并且应用于所有监视概要文件。

注: EIF 接收方消耗事件，否则在代理程序高速缓存变满时，可能发生问题。

K*PC*_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN
具有日志文件事件监视的操作系统代理程序具有子节点限制。为管理日志文件事件，子节点 MSN 由以下结构组成：UX:*CITRAHOSTNAME_PROFILENAME*。子节点名的最大大小限制为 32 个字符。如果构建的子节点 MSN 名称太长并且超过 32 个字符，那么将被截断为 32 个字符。此名称与从“概要文件名称”获取的子串对应。

在操作系统代理程序配置文件中，使用以下变量来管理过长的概要文件名称：

- UNIX 操作系统代理程序：KUX_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true
- Linux 操作系统代理程序：KLZ_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true
- Windows 操作系统代理程序：KNT_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true

例如，如果具有长度为 20 个字符的名为 aixhost_nc123456789A 的代理程序，其 20 个字符为 CTIRAHOSTNAME=aixhost_nc123456789A。

具有以下两个概要文件：

```
ProfileLong12A ( 长度为 14 个字符 )
ProfileLong12B ( 长度为 14 个字符 )
```

预期具有以下相关子节点 MSN：

```
UX:aixhost_nc123456789A_ProfileLong12A ( 长度为 38 个字符 )
UX:aixhost_nc123456789A_ProfileLong12B ( 长度为 38 个字符 )
```

但是，子节点 MSN 将被截断为 32 个字符限制，因此针对以下两者，生成的名称相同：

```
UX:aixhost_nc123456789A_ProfileL
UX:aixhost_nc123456789A_ProfileL
```

要截断 CTIRAHOSTNAME 而不是概要文件名称，请设置 *Kpc_FCP_TRUNCATE_HOSTNAME_IN_LONG_SUBNODE_MSN=true* 变量。

例如，如果 *n* 是概要文件名称的长度（例如，14），那么与 *CTIRAHOSTNAME* 相关的 MSN 名称的子串会截断为 32-n-3 个字符，因此 *CTIRAHOSTNAME* 变量为：`aixhost_nc1234`。从而专用子节点 MSN 为：

```
UX:aixhost_nc1234_ProfileLong12A
UX:aixhost_nc1234_ProfileLong12B
```

配置文件
OS 代理程序使用一个在其启动时读取的配置文件。该文件包含配置选项和过滤器。您必须创建此配置文件并配置代理程序实例以使用此文件。

此后，系统每 60 秒监视一次配置文件的时间戳记是否有更改。如果文件的时间戳记发生更改，那么代理程序动态重新初始化其配置，而无需重新启动。有关更多信息，请参阅第 555 页的『更改代理程序配置和格式文件』。

操作系统代理程序的 `.conf` 文件接受以下选项：

codepage
此参数是受监视文件的代码页。当受监视文件的代码页与系统代码页不同时，使用配置文件中的此参数。指定受监视文件的代码页，例如，ibm-5348_P100-1997、UTF-16 或 UTF-8。

ConfigFilesAreUTF8=Y
此参数指定配置文件和格式文件采用 UTF-8。如果配置文件的编码为 UTF-8 而系统代码页不是时，使用此参数。缺省情况是代理程序假定系统编码。

DupDetectionKeyAttributes
Cloud APM 属性的逗号分隔列表，用于确定哪些事件重复。如果两个事件中所有指定的属性均相同，那么这两个事件被视为重复。此选项仅适用于事件。有关更多信息，请参阅第 863 页的『事件过滤和摘要』。

注：

1. 属性名称区分大小写，因此必须按照描述精确输入名称。
2. 如果未提供属性列表，那么该值缺省为 `Class` 和 `Logname`。

ENFORCE_STRICT_TEC_COMPATIBILITY
此参数指示日志数据中所有空格字符以确保考虑这些字符。例如，在使用诸如 `"%s %s"` 的格式从日志消息抽取信息时，操作系统代理程序不仅匹配字面值空格，而且还匹配存在的任何其他空格字符，例如，Tab 键或回车符。

如果未设置此参数，那么在匹配 Tivoli Enterprise Console® 样式格式字符串时，操作系统代理程序的缺省行为是匹配尽可能多的输入文本，而且按照从左到右的顺序处理格式。

例如，对于 `%s:%s` 格式字符串和 `one:two:three` 输入字符串，操作系统代理程序缺省将 `one.two` 分配给第一个参数（对应于第一个 `%s`）并且将 `three` 分配给第二个参数。

注：

1. 此参数不适用于使用正则表达式语法的格式语句。
2. 设置此参数会对性能产生影响。为更好地控制匹配的行为和性能，请避免设置此参数，而是使用正则表达式。

EventSummaryInterval
指定代理程序搜索重复事件以禁止的秒数。将此参数设置为正整数。此选项仅适用于事件。有关更多信息，请参阅第 863 页的『事件过滤和摘要』。

EventFloodThreshold
指定在检测到重复事件时发送的事件。将此参数设置为 `send_none`、`send_all`、`send_first` 或正整数。此选项仅适用于事件。有关更多信息，请参阅第 863 页的『事件过滤和摘要』。

EventMaxSize
指定生成的事件的最大大小（以字节为单位）。如果指定，那么将在两个位置中使用此参数：

1. 代理程序可使用参数来设置用于处理事件的缓冲区的大小。如果未设置，那么此缓冲区缺省为 16384 字节大小。如果将缓冲区设置的太小，那么事件将被截断并且可废弃。

2. EIF 发送方可使用参数来设置用于向 EIF 接收方（例如，OMNIbus EIF 探测器）发送事件的缓冲区的大小。如果未设置，那么此缓冲区缺省为 4096 字节大小。如果将缓冲区设置的太小，那么事件将被废弃。

FileComparisonMode
指定在多个日志文件匹配通配符模式时监视哪个文件。以下值可用：

CompareByAllMatches
该值是缺省行为。将监视匹配 LogSources 中指定的通配符模式的所有文件。

CompareByLastUpdate
将监视匹配 LogSources 中指定的通配符模式的文件中具有最新更新时间戳记的文件。

CompareBySize
将选择监视匹配文件名模式条件的两个或更多文件中较大的文件。请勿将 CompareBySize 用于将同时进行更新且文件大小增加的多个匹配文件。如果最大的文件频繁更改，那么监视可能在新选中的文件开始位置连续重新启动。相反，对于一组匹配文件，如果在任何特定时间只有一个文件活动且进行更新，那么使用 CompareBySize。

CompareByCreationTime
将监视匹配 LogSources 中指定的通配符模式的文件中具有最新创建时间戳记的文件。该值具有以下限制：

· 该值仅适用于 Windows 操作系统，因为 UNIX 和 Linux 操作系统不存储文件的真实创建时间。

· 使用安全 Shell (SSH) 文件传输协议监视的远程文件不支持该值。

提示: CompareByLastUpdate、CompareBySize 和 CompareByCreationTime 值均可用于滚动日志文件。CompareByLastUpdate 通常用于这些文件。

FQDomain
指定代理程序是否设置域名以及设置方式：

· 如果设置为 yes，那么代理程序确定系统域名。

· 如果设置为 no，那么代理程序不设置域名。fqhostname 属性指定空字符串。

· 如果设置为不包含 yes 或 no 值，那么接受域名作为值并且附加到主机名。

有关更多信息，请参阅第 552 页的『格式文件』。

IncludeEIFEventAttr
代理程序包含名为 *EIFEvent* 的大属性，这是在启用此功能时将通过“事件集成工具”发送的事件的表示。也可以在其他属性中找到 *EIFEvent* 属性中包含的信息。太大的大小使其成为问题，因此缺省情况下禁用。将此值设置为 y 可重新启用 EIFEvent 属性。

注: 如果您有大事件，那么使用此属性可能导致阈值失败。此上下文中的大事件是如下事件：包含所有属性及其名称的所有值所需的字节总数导致字符串超过 3600 字节。

LognameIsBasename
在设置为 y 时，Logname 属性的值是事件所在的日志文件的基本名称。此选项仅适用于 Performance Management 事件。将移除路径。例如，/data/logs/mylog.log 将变为 mylog.log。如果该值设置为 n，那么您可以获取完整路径。但是，因为属性限制为 64 个字符，将其设置为 n 意味着将截断较长的名称。因此，缺省值为 y。要查看较长属性中的完整路径名，您可以在 .fmt 文件中通过格式的映射部分指定，例如，filename FILENAME CustomSlot1。映射将使用事件所在的路径的完整路径填写名为 filename 的槽，并将其映射到长度为 256 个字符的 CustomSlot1。

LogSources
指定用于轮询消息的文本日志文件。必须指定每个文件的完整路径，并且必须用逗号分隔文件名。在每个文件名中，您还可以使用星号 (*) 来表示任何字符序列，或者使用问号 (?) 来表示任何单个字符。例如，mylog* 导致轮询名称以 mylog 开头的所有日志文件，而 mylog??? 导致轮询名称由 mylog 以及后跟的 3 个字符组成的所有日志文件。仅在文件名中支持这些通配符；必须显式指定路径。

如果想要在路径中使用正则表达式或模式匹配，请参阅 RegexLogSources 描述。

在启动代理程序时，无需存在日志文件源；将在创建时轮询日志文件。

NewFilePollInterval
指定代理程序检查新文件以进行监视的频率（秒）。例如，如果在代理程序启动时 *LogSources* 或 *RegexLogSources* 配置文件设置指定的文件名不存在，那么将在此时间间隔后重新检查文件是否存在。

NumEventsToCatchUp
在日志中指定代理程序开始的事件。如果监视的源是新的或者代理程序长时间停止，那么此选项可提供灵活性。以下值有效：

注： 对于文本文件，值 `0` 和 `-1` 适用。对于 Windows 事件日志，值 `0`、`-1` 和 n 适用。

0
从日志中的下一个事件开始。该值是缺省值。

-1
在设置为 `-1` 时，代理程序保存其在监视的文件中的位置。保存其位置，从而在代理程序停止并稍后重新启动时，可处理停止期间写入日志的任何事件。否则，在代理程序停止并从文件结束位置重新启动时，将忽略归档的事件。此设置不适用于管道或者 UNIX 和 Linux 系统上的系统日志监视。

n
设置为正整数。从日志中最新事件的第 *nth* 个事件开始；也就是，启动日志中最新事件的倒数第 *n* 个事件。如果 *n* 大于可用的事件数量，那么将处理可用的所有事件。

注： 您只能将 n 值用于 Windows 事件日志。在 `UseNewEventLogAPI` 设置为 *y* 时，将忽略 n 值。

PollInterval
指定轮询 `LogSources` 选项中列出的每个日志文件以查找新消息的频率（秒）。缺省值为 5 秒。

如果从先前发行版升级了 Windows 事件日志并且在 Windows 注册表中针对 `PollingInterval` 设置了值，那么必须使用 Windows 注册表中使用的相同值指定代理程序配置文件中的 `PollInterval` 选项。仅在替换注册表中已有值的 Tivoli Enterprise Console 操作系统代理程序时，此规则才适用。

ProcessPriorityClass
指定代理程序的进程优先级。如果代理程序处理大量事件并且使用太多处理器资源，那么可以调整该值以提高系统性能。可能的值为：

- `A` - 极低优先级
- `B` - 低优先级
- `C` - 普通优先级
- `D` - 高于普通优先级
- `E` - 高优先级
- `F` - 极高优先级
- `USE_CONF_FILE_VALUE` - 使用配置文件中指定的值。该值是缺省值。

RegexLogSources
指定用于轮询消息的文本日志文件。其不同于 LogSources 选项，在此正则表达式中，可以在文件名的基本名称部分以及文件名的一个子目录中使用元字符。在描述要在多个目录中监视的多个文件时，此差别比提供 LogSources 选项提供更大的灵活性。

例如，针对 LogSources 语句指定 `/var/log/mylog*` 等同于使用后跟星号 (`*`) 元字符的句点 (`.`) 元字符以在 RegexLogSources 语句中构成 `/var/log/mylog.*`。此类型的限定符导致轮询基本名称以 `mylog` 开始且后跟零或多个字符的 `/var/log` 目录中的所有日志文件。`/var/log/mylog.+` 限定符导致轮询名称以 `mylog` 开头且后跟一个或多个字符的 `/var/log` 目录中的所有日志文件。

与 LogSources 类似，必须指定每个文件的完整路径并且文件名必须以逗号分隔。但是，逗号也是正则表达式中的有效字符。要区分用作正则表达式一部分的逗号与用于分隔文件名的逗号，用作正则表达式一部分的逗号必须用反斜杠 (`\`) 字符进行转义。

例如，如果想要搜索匹配 `/logs/.*\.log` 或 `/other/logs/[a-z]{0,3}\.log` 正则表达式之一的日志，那么必须转义第二个表达式的 `{0,3}` 子句中的逗号，从而使代理程序不会搞错，新表达式开头为：`RegexLogSources=/logs/.*\.log,/other/logs/[a-z]{0\,3}\.log`

如果在路径名中使用了元字符，那么只能在路径的一个子目录中使用这些元字符。例如，可以指定 /var/log/[0-9\.]*/mylog.* 以在一个子目录中包含元字符。[0-9\.]* 将导致与仅包含数字和点 (.) 的 /var/log 任意子目录匹配。mylog.* 将导致与那些 /var/log 子目录中所有以 mylog 开头并且后跟零个或零个以上字符的任意文件名匹配。

某些操作系统使用反斜杠 (\) 作为目录分隔符，这可能会与正则表达式转义元字符混淆。为了避免此类混淆，必须始终使用正斜杠来指示目录。例如，指定为 C:\temp\mylog.* 的 Windows 文件可能表示 \t 是缩写的制表符。因此，对于所有操作系统目录分隔符，总是使用正斜杠 (/)。例如，C:/temp/mylog.* 表示 C:/temp 目录中以 mylog 开头的所有文件。

如果多个子目录包含元字符，那么还将使用跟则消息。例如，c:/[0-9\.]*/temp.files/mylog.* 具有使用元字符的两个子目录。[0-9\.]* 是使用元字符的第一个子目录，而 temp.files 是使用句点 (.) 元字符的第二个子目录。在此情况下，代理程序假定使用包含元字符的第一个子目录，并且忽略包含元字符的后续子目录。

SubnodeName
将向监视概要文件子节点分配可用于覆盖缺省名称的字符串值。缺省情况下，分配给监视概要文件的子节点名称对应于用于此概要文件的配置文件的基本名称。使用此设置可指定不同的子节点名称。

SubnodeDescription
可用于向 *LFAProfiles* 的 *Subnode Description* 属性分配值的字符串值。

UnmatchLog
指定文件以记录代理程序无法解析为事件类的废弃的事件。然后，可分析废弃的事件以确定是否需要修改代理程序格式文件。匹配使用 *DISCARD* 的模式的事件不会在不匹配的日志中显示，因为其不匹配模式。

此选项用于测试环境以供验证格式文件中的过滤器。如果长期保留此选项，那么将会填满文件系统。

使用 SSH 的远程日志文件监视的选项

与作为列表的 **SshHostList** 不同，所有选项只能有一个值，这适用于 **SshHostList** 中指定的所有远程主机。

仅支持文本日志文件。不支持 AIX 错误报告、系统日志和 Windows 事件日志。

提示: 您可以设置系统日志以将其输出写入到文本日志文件，然后使用操作系统代理程序远程监视此文本文件。

SshAuthType
必须设置为 *PASSWORD* 或 *PUBLICKEY*。如果设置为 *PASSWORD*，那么 **SshPassword** 的值将作为用于向所有远程系统进行 SSH 认证的密码。如果设置为 *PUBLICKEY*，那么 **SshPassword** 的值将作为控制专用密钥文件访问的口令。如果设置为 *PUBLICKEY*，那么还必须指定 **SshPrivKeyfile** 和 **SshPubKeyfile**。

SshHostList
要监视的远程主机的逗号分隔列表。将在此处列出的每个主机上监视在 **LogSources** 或 **RegexLogSources** 语句中指定的所有日志文件。如果 *localhost* 是指定的主机名之一，那么代理程序直接在本地系统上监视相同的文件集。在指定 *localhost* 时，不会使用 SSH 来访问本地系统上的文件；将直接读取日志文件。

SshPassword
在 **SshAuthType** 的值为 *PASSWORD* 时，该值是在 **SshUserid** 中指定的用户的帐户密码。您可以通过明文提供帐户密码，或者可以提供使用 IBM Tivoli Monitoring CLI **itmpwdsnmp** 命令加密的密码。有关使用 **itmpwdsnmp** 命令加密密码的更多信息，请参阅第 559 页的『远程日志文件监视：加密密码或口令』。

在 **SshAuthType** 的值为 *PUBLICKEY* 时，该值是解密 **SshPrivKeyfile** 参数指定的专用密钥的口令。您可以通过明文提供口令，或者可以提供使用 IBM Tivoli Monitoring CLI **itmpwdsnmp** 命令加密的口令。有关使用 **itmpwdsnmp** 命令加密密码的更多信息，请参阅第 559 页的『远程日志文件监视：加密密码或口令』。

注: 如果 **SshAuthType** 的值为 *PUBLICKEY*，并且配置的 SSH 不需要口令，那么必须将 **SshPassword** 设置为 null。要将 **SshPassword** 设置为 null，配置文件中的条目为：

```
SshPassword=
```

SSHPort
用于 SSH 连接的 TCP 端口。如果未设置，那么缺省为 *22*。

SshPrivKeyfile
如果 **SshAuthType** 设置为 *PUBLICKEY*，那么该值必须是包含在 **SshUserid** 中指定的用户的专用密钥的文件的完整路径，而且还必须设置 **SshPubKeyfile**。如果 **SshAuthType** 未设置为 *PUBLICKEY*，那么无需该值且将忽略。

SshPubKeyfile
如果 **SshAuthType** 设置为 *PUBLICKEY*，该值必须是包含在 **SshUserid** 中指定的用户公用密钥的文件的完整路径，而且还必须设置 **SshPrivKeyfile**。如果 **SshAuthType** 未设置为 *PUBLICKEY*，那么无需该值且将忽略。

SshUserid
远程系统上的用户名，代理程序将其用于 SSH 认证。

仅在 UNIX 和 Linux 系统上支持的选项

Linux AIX

AutoInitSyslog
如果此选项设置为 Yes，那么代理程序自动配置系统日志工具以将事件的标准集合写入代理程序监视的管道。通过启用此设置，您可以监视系统日志事件而不必维护和滚动日志文件。如果未在配置文件中设置此选项，那么与设置为 No 相同。

限制: 远程日志文件监视不支持此选项。

仅在 Windows 系统上支持的选项

Windows

NTEventLogMaxReadBytes
如果正在使用旧的 NT 事件日志界面（UseNewEventLogAPI 未设置为 y）来读取 Windows 系统上的事件日志数据，那么在每次检查事件日志以查找新数据时，代理程序都将读取至此字节数。将值设置为 0 会导致代理程序尝试读取所有新数据，如先前发行版中一样。在系统上存在大量事件时，此活动可能占据代理程序相当长的时间。缺省值为 655360。在设置时，代理程序可能不会在确切指定的值处停止，而是在内部缓冲区大小的倍数最接近该值处。

PreFilter
指定在代理程序处理之前过滤 Windows 事件日志中事件的方式。在过滤器确定将哪个事件从事件日志发送到代理程序时，PreFilterMode 使用 PreFilter 语句。在 PreFilter 语句中的每个 *attribute=value* 规范匹配事件日志中的事件时，事件匹配 PreFilter 语句。PreFilter 语句必须至少包含日志规范，并且最多可包含三个其他规范，这些规范均为可选：事件标识、事件类型和事件源。语句中属性的顺序并不重要。

PreFilter 语句具有以下基本格式：

```
PreFilter:Log=log_name;EventId=value; EventType=value;Source=value;
```

您可以为每个属性指定多个值，用逗号分隔每个值。

每个 PreFilter 语句必须位于单独一行上。

PreFilter 不是必需的。如果未指定预过滤器并且 PreFilterMode=OUT，那么会将所有 Windows 日志事件发送到代理程序。

PreFilterMode
此选项仅适用于 Windows 事件日志。此选项指定是发送 (PreFilterMode=IN) 还是忽略 (PreFilterMode=OUT) 匹配 PreFilter 语句的 Windows 系统日志事件。有效值为 IN、in、OUT 或 out。缺省值为 OUT。

PreFilterMode 为可选；如果未指定 PreFilterMode，那么仅将不匹配任何 PreFilter 语句的事件发送到代理程序。

注: 如果设置 PreFilterMode=IN，那么还必须定义 PreFilter 语句。

SpaceReplacement
缺省情况下，对于 Windows 事日志（仅限 Windows Server2008）设置为 TRUE，而非针对先前版本的事件日志。在 SpaceReplacement 为 TRUE 时，事件日志消息的安全标识、子源、级别和关键字字段中的任何空格都将替换为下划线 (_)。在 SpaceReplacement 为 FALSE 时，保留事件日志消息的安全标识、子源、级别和关键字字段中的任何空格不变。有关此选项的更多信息，请参阅第 865 页的『Windows 事件日志』。

UseNewEventLogAPI
在 Windows 系统上设置为 y 时，将新的 Windows 事件日志界面用于事件日志。仅在 Windows 2008 和更高版本上支持此选项。需要选项以访问在 Windows 2008 中调试的多个新事件日志以及在此系统上运行的应用程序。在先前版本的 Windows 以及 UNIX 和 Linux 上，忽略此选项。有关此选项的更多信息，请参阅第 865 页的『Windows 事件日志』。

WINEVENTLOGS
控制监视的 Windows 事件日志。

WINEVENTLOGS 语句是一个逗号分隔且不含空格的列表。有关更多信息，请参阅第 865 页的『Windows 事件日志』。

注: Windows 事件中的任何回车符、Tab 键或换行符都将替换为空格。

仅在 AIX 系统上支持的选项

AIX

AIXErrptCmd
可在此处提供代理程序运行的 **errpt**（错误报告）命令字符串。命令输出将供给到监视的日志数据流中。

例如，以下命令导致代理程序搜索 *mmddhhmmyy* 字符串并将其替换为启动时的实际日期和时间。将只替换此字符串的第一个实例。

```
AIXErrptCmd=errpt -c -smmddhhmmyy
```

虽然您可以提供自己的 errpt 命令，但是必须使用 -c（并行方式）选项，从而使命令连续运行。不能使用 -t 选项或导致详细输出的以下选项：-a、-A 或 -g。

数据流是 errpt 命令的标准输出，因此必须在 .fmt 文件中写入正则表达式以进行匹配。例如，数据输出可能为：

```
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
F7FA22C9   0723182911 I O SYSJ2      UNABLE TO ALLOCATE SPACE IN FILE SYSTEM
2B4F5CAB   1006152710 U U ffdc       UNDETERMINED ERROR
2B4F5CAB   1006152610 U U ffdc       UNDETERMINED ERROR
```

选取数据行（而非头）的样本格式为：

```
REGEX GenericErrpt
^([A-F0-9]{8}) +([0-9]{10}) ([A-Z]) ([A-Z]) (\S+) +(.*)$
Identifier $1 CustomSlot1
Timestamp  $2 CustomSlot2
T          $3 CustomSlot3
C          $4 CustomSlot4
Resource   $5 CustomSlot5
msg        $6
END
```

有关更多信息，请参阅《IBM Agent Builder 用户指南》中的迁移 *AIX* 二进制日志。

仅在将事件转发到 EIF 时适用的选项

要点: 这些选项适用于直接发送到 Operations Analytics - Log Analysis、OMNIbus 或任何其他通用 EIF 接收器的 EIF 事件。这些选项不适用于 Cloud APM 服务器。

BufferEvents
指定如何启用事件转发。可能的值为:

- **YES** - 在 BufEvtPath 选项指定的文件中存储事件（该值是缺省值）。
- **MEMORY_ONLY** - 在内存中缓存事件。
- **NO** - 不存储或缓存事件。

BufEvtPath
指定代理程序缓存文件的完整路径名。如果未调整此路径，那么缺省值为:

- AIX `/etc/Tivoli/tec/cache`
- Windows `\etc\Tivoli\tec\cache`

注: 如果将事件转发到多个服务器，那么必须针对每个转发通道指定 *BufEvtPath* 值。对于其他每个条目，将向 *BufEvtPath* 名称附加索引数字。例如，使用 *BufEvtPath1* 来指示转发到第一个额外的服务器的代理程序高速缓存文件的路径名。在每个 *BufEvtPath* 中设置的值必须唯一。

BufEvtMaxSize
指定代理程序缓存文件的最大大小 (KB)。缺省值为 64。在将 *BufferEvents* 选项设置为 Yes 时，缓存文件在磁盘上存储事件。文件的最小大小为 8 KB。将忽略小于此级别的指定的文件大小，并使用 8 KB。针对最大文件大小指定的值无上限。

注: 如果缓存文件存在，那么必须删除文件以使选项更改生效。

NO_UTF8_CONVERSION
指定 Event Integration Facility 是否以 UTF-8 编码事件数据。在此选项设置为 YES 时，EIF 不以 UTF-8 编码事件数据。在将数据传递到 EIF 时，假定数据已采用 UTF-8 编码。但是，将向标志添加前缀以指示数据采用 UTF-8 编码（如果，标志在事件数据的开头不存在）。缺省值为 NO。

MaxEventQueueDepth
该值指示排队进行转发的事件的最大数量。在到达限制时，队列中的每个新事件将取代队列中最旧的事件。如果未指定，那么缺省值为 1000。如果使用 *NumAdditionalServers*，那么此设置应用于所有转发通道。

NumAdditionalServers
如果想要将事件转发到多个 Netcool/OMNIbus ObjectServer，那么此条目是必需的。使用它的值来指示要将事件转发到的服务器的数量。有效值为 1 - 8。

ServerLocation
指定在其中安装事件服务器的主机的数量。指定主机名或 IP 地址。使用点分格式的 IP 地址。如果希望，可以为服务器指定故障转移值，例如，`ServerLocation1=2.3.4.5,2.3.4.6.` 。如果针对 *ServerLocation* 指定故障转移值，那么还必须针对每个 *ServerLocation* 指定额外的 *ServerPort* 值。

注: 如果将事件转发到多个服务器，那么必须针对每个服务器指定 *ServerLocation* 值。对于其他每个条目，将向 *ServerLocation* 名称附加索引数字。例如，使用 *ServerLocation1* 来指定安装第一个额外的服务器的主机的名称。

ServerPort
指定 EIF 接收方侦听事件的端口号。*ServerPort* 选项最多可包含 8 个值，用逗号分隔。如果针对 *ServerLocation* 指定故障转移值，那么必须设置相等的 *ServerPort* 值。在指定 *TransportList* 选项时，不使用 ServerPort。

注: 如果将事件转发到多个服务器，那么必须针对每个服务器指定 *ServerPort* 值。对于其他每个条目，将向 *ServerPort* 名称附加索引数字。例如，使用 *ServerPort1* 来指定 EIF 接收方侦听第一个额外的服务器的事件的端口号。

TransportList
指定用户提供的传输机制名称，使用逗号分隔。当传输机制对于发送方应用程序失败时，API 按照列表中指定的顺序使用以下传输机制。对于接收应用程序，API 创建并使用所有传输机制。必须使用 Type 和 Channels 关键字指定每个 *type_name* 的传输类型和通道：

***type_name*Type**

指定 *TransportList* 选项所制定的传输机制的传输类型。SOCKET 是唯一受支持的传输类型。

每个 channel_name 的服务器和端口由 *ServerLocation* 和 *ServerPort* 选项指定。

***type_name*Channels**

***channel_name*Port**
指定传输机制服务器侦听指定的通道的端口号（由 *Channel* 选项设置）。在此关键字设置为零时，将使用端口映射程序。此关键字是必需的。

***channel_name*PortMapper**
针对指定的通道启用端口映射程序。

***channel_name*PortMapperName**
指定端口映射程序的名称（如果启用端口映射程序）。

***channel_name*PortMapperNumber**
指定远程过程调用注册的标识。

***channel_name*PortMapperVersion**
指定端口映射程序的版本（如果启用端口映射程序）。

***channel_name*ServerLocation**
指定事件服务器的名称和针对指定的通道查找传输机制的服务器的区域。*Channel* 选项设置通道。此关键字是必需的。

在直接用于 OMNIbus 时，配置文件接受通用 EIF 选项。这些选项仅在到 OMNIbus 的 EIF 连接上生效。不会影响发送到 Cloud APM 服务器的事件。有关这些 EIF 选项的更多信息，请参阅 EIF 关键字。

格式文件
操作系统代理程序从系统日志消息抽取信息，然后将不同的日志消息与事件类相匹配。格式文件充当查找文件以将日志消息与事件类相匹配，告知事件类要读取的内容、要匹配的内容以及如何格式化数据。

在将格式个文件用作查找文件时，将从头到尾比较文件中的所有格式规范。在两个类匹配或者消息具有多个匹配类时，将使用此匹配末尾的第一个表达式。如果找不到匹配，那么将丢弃事件。如果在 `.conf` 文件中进行定义，那么会将废弃的事件写入不匹配日志。

将描述用于创建模式以匹配日志消息和事件的正则表达式语法。通过使用 International Components for Unicode (ICU) 库提供正则表达式过滤支持，以确定检查的属性值是否与指定模式匹配。

有关使用正则表达式的更多信息，请参阅 *ICU User Guide* 中的 Regular Expressions。

格式文件规范
格式文件描述代理程序查找以匹配受监视的日志中的事件的模式。格式文件包含一个或多个格式规范。

您可以在代理程序实例运行时更改格式文件。代理程序在启动时读取文件，并在此后每 60 秒监视一次其时间戳记更改。如果文件的时间戳记发生更改，那么代理程序动态重新初始化其配置，而无需重新启动。有关更多信息，请参阅第 555 页的『更改代理程序配置和格式文件』。

要创建新模式以匹配事件，请使用包含以下部分的新的正则表达式语法：

- 格式头
- 正则表达式
- 槽映射
- 结束语句

格式头包含 **REGEX** 关键字，其通知代理程序您使用正则表达式来匹配受监视的日志中的模式。

将此正则表达式分配给事件类，如以下示例中所示：

```
REGEX REExample
```

如果使用特殊的预定义事件类 *DISCARD* 作为事件类，那么将丢弃匹配关联的模式的任何日志记录，并且不会针对它们生成任何事件。例如：

```
REGEX *DISCARD*
```

在模式匹配时，不会将任何项写入不匹配日志。匹配的日志文件状态记录包含这些废弃的事件。

注: 您可以将多个事件定义分配给相同事件类或不同事件类。类名是任意项，您可以使用其来指示事件类型或者以各种方式分组事件。

在格式头后，格式内容在第一行包含一个正则表达式，后跟映射。每个映射都在单独一行上显示，以下示例描述了这些映射。

将选择匹配正则表达式的所有行并作为事件发送到监视服务器。正则表达式包含子表达式。您可以使用子表达式来匹配与事件集成工具中名为 *slot* 的变量相同的这些行的特定部分。

以下监视日志包含可能想要监视的三行：

```
Error:  disk failure
Error: out of memory
WARNING: incorrect login
```

例如，针对特定错误生成事件，例如，以 Error 开头的行，并忽略以 Warning 开头的行。正则表达式必须匹配以 Error 开头的行，而且还包含一个子表达式。子表达式用圆括号括起来，并且必须仅匹配想要分配给 *msg* 槽的输入文本。以下格式定义是仅包含一个子表达式的简单正则表达式：

```
REGEX REExample
Error: (.*)
msg $1
END
```

根据此格式规范以及前面的日志数据集，代理程序生成两个事件。这两个事件都指定 REEXample 事件类。在第一个事件中，将 disk failure 值分配给 *msg* 槽。另外，在第二个事件中，将 out of memory 值分配给 *msg* 槽。因为 Warning 行不匹配正则表达式，因此将忽略并且不生成任何事件。

在将值 $1 分配给 *msg* 槽时，将为其分配字一个子表达式的值。

如果您有包含以下错误的日志文本，那么可能想要将这些错误消息分配给它们自己的事件类从而立即通知您发生磁盘故障：

```
Error: disk failure on device /dev/sd0: bad sector
Error: disk failure on device /dev/sd1: temperature out of range
```

您可以在事件中包含发生错误的磁盘的描述以及更具体的磁盘错误。

以下正则表达式包含标识此信息的两个子表达式：

```
REGEX DiskFailure
Error: disk failure on device (/dev/sd[0-9]):(.*)
device $1 CustomSlot1
msg $2
END
```

将这两个子表达式分配给事件槽。生成的两个事件包含以下值：

```
"device=/dev/sd0" and "msg=bad sector"
"device=/dev/sd1" and "msg=temperature out of range"
```

如果使用 EIF 以生成第一个事件，那么将显示为在以下示例中显示：

```
DiskError;device='/dev/sd0';msg='bad sector';END
```

如果将事件发送到 Cloud APM 服务器，那么会将名为 *msg* 的槽分配给相同名称的 Performance Management 代理程序属性。但是，*device* 槽无预定义属性。

如果需要在 Cloud APM 控制台上查看直接分配给 *device* 的值，或者针对其编写阈值，那么必须将其分配给 Performance Management 属性。

操作系统代理程序包含以下 13 个预定义的属性：

· 范围从 *CustomSlot1* 到 *CustomSlot10* 的 10 个字符串类型属性

· 范围从 *CustomInteger1* 到 *CustomInteger3* 的三个整数类型属性

使用格式文件中的三个属性名称来填充相同名称的 Performance Management 属性。使用这些属性不会影响直接发送给 OMNIbus 的 EIF 事件的内容。

注: `CustomSlot` 和 `CustomInteger` 属性名称区分大小写，因此必须完全按照显示输入名称。

将槽从事件定义分配给格式文件中这些定制 Performance Management 属性之一。

将 *device* 槽分配给名为 *CustomSlot1* 的 Performance Management 字符串类型属性，如以下示例中所示：

```
REGEX DiskFailure
Error: disk failure on device (/dev/sd[0-9]):(.*)
device $1 CustomSlot1
msg $2
END
```

在 应用程序性能仪表板 中显示事件时，会将值分配给 *device* 槽的值分配给 Performance Management `CustomSlot1` 属性。在 Cloud APM 控制台中查看该值，或者使用其来定义阈值。您可以使用槽定义旁边的“`CustomSlot`*n*”以相同方式将事件定义中的任何槽分配给任何 10 个定制代理程序属性，其中，*n* 是 1 到 10 个一个数字。

在此示例中，第一个子表达式将专门定义为 `(/dev/sd[0-9])`，但是第二个子表达式通常定义为 `(.*)`. 请尽可能精确地定义正则表达式，从而提高性能。因此，如果输入搜索以在设备上搜索不匹配此处定义的特定错误消息的错误，在找不到错误时，搜索过程将立即停止。并不非浪费时间查找匹配。

END 关键字完成格式规范。格式头、正则表达式和 *END* 关键字必须在新行上开始，如以下示例中所示：

```
REGEX REExample
Error:
msg $1
END <EOL>
<EOF>
```

注: 对于文件中的最后一个格式，必须在 `END` 关键字后插入新行，如示例中所示。 否则，您将收到解析错误。

CustomInteger1 到 *CustomInteger3* 是 64 位定制整数属性。您可以通过与字符串类型 `CustomSlot` 属性相同的方式使用它们。您可以使用这些属性以将单个槽或子表达式从日志文件映射到单个 Cloud APM 属性。由于这些属性是数字，您可以对其使用算术比较，例如 < 和 >，这不适用于字符串属性。

注: 虽然 Cloud APM 服务器将这些值评估整数，但是针对 EIF 目的并且在格式文件中，它们仍被作为字符串。例如，要在 PRINTF 语句中使用整数槽，仍然使用“%s”而非“%d”对其进行识别。

以下示例阐述使用定制整数属性。假定收到定期 UNIX 系统日志消息，报告可用的文件系统的百分比，例如，以下假设的日志记录：

```
Oct 24 11:05:10 jimmy fschecker[2165]: Filesystem /usr is 97% full.
```

您可以在格式文件中使用以下语句来检查可用的文件系统的百分比：

```
REGEX FileSystemUsage
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2}) (.*?) (.*?):
Filesystem (.*?) is ([0-9]+)% full\.$
Month   $1 CustomSlot1
Date    $2 CustomSlot2
Time    $3 CustomSlot3
Host    $4 CustomSlot4
Service $5 CustomSlot5
Filesystem     $6 CustomSlot6
PctFull         $7  CustomInteger1
msg          PRINTF("%s: %s% full", Filesystem, PctFull)
END
```

注: 在前面的语句中，第二行和第三行上的 ^ 和 $ 符号之间的所有内容都必须位于一行上。

因为可能有其他事件在 *CustomInteger1* 中添加值，您可以使用 *Class* 属性的值来显示其对于事件的正确类型的影响，从而避免混淆不同的事件类型。例如，以下阈值公式导致阈值仅在 *FileSystemUsage* 事件类的事件在 *CustomInteger1* 中具有大于或等于 95 的值时才触发阈值:

```
(Class == 'FileSystemUsage' AND CustomInteger1 >= 95)
```

然后，不同事件可将 *CustomInteger1* 用于不同目的，而不是意外触发此阈值。

总而言之，现在您可以在 Performance Management 中编写在 `CustomInteger` 属性上使用算术运算符的阈值，这不适用于 `CustomSlots` 属性。

注: 如果将非整数数据映射到 `CustomInteger` 属性，那么生成值可能为零或某些意外的值。

更改代理程序配置和格式文件

操作系统代理程序在启动时读取配置文件（`.conf`）和格式文件（`.fmt`），并从此开始每 60 秒监视时间戳记一次。

如果配置文件或格式文件的时间戳记发生更改，代理程序将动态地重新初始化其配置，而不需要重新启动。重新初始化期间，监视会暂时中断。监视恢复时，代理程序必须在监视日志中确定重新启动的位置。因此，代理程序如完全停止后重新启动一样进行操作。

注: 配置文件或格式文件更改后重新初始化代理程序，重置 `Log File RegEx Statistics`、`Log File Status` 和 `Log File Event` 属性组中的信息。

缺省情况下，初始化完成后，代理程序将从文件的尾部启动监视。开始位置可能导致丢失监视中断期间发生的事件。为确保在监视恢复时收集到这些事件，请使用 `NumEventsToCatchUp=-1` 设置。

设置 `NumEventsToCatchUp=-1` 将维护位置文件。每次代理程序读取日志文件时都将更新位置文件。更新时会保存日志文件中代理程序的位置，以防代理程序重新启动。维护位置文件会对性能产生小的影响，因此只在必要时维护此文件。有关 `NumEventsToCatchUp` 的更多信息，请参阅第 545 页的『配置文件』。

注: 这些配置值不会出现在配置文件中，并且在初始化配置期间进行设置。如果更改这些值，必须重新启动代理程序。

继承

格式文件使用继承从预先定义的格式规范派生槽定义。

使用 FOLLOWS 关系通过继承从通用格式规范构建特定格式规范。

例如，首先，定义基类并将其命名为 `DiskFailure`，如下所示:

```
REGEX DiskFailure
Disk Failure on device (.*)
device $1 CustomSlot1
END
```

此正则表达式匹配监视日志中的 `Disk Failure on device/dev/sd0` 错误，从而将 `/dev/sd0` 值分配给 *device* 槽。

但是，您还可以查看监视日志中报告的此错误消息的扩展版本。

例如，您可能会看到 `Disk Failure on device /dev/sd0, error code: 13` 错误消息。

将此错误消息与以下示例中显示的槽相匹配:

```
REGEX DiskFailureError FOLLOWS DiskFailure
Disk Failure on device (.*), error code: ([0-9]*)
errcode $2 CustomSlot2
END
```

现在，事件包含 *device* 槽和 *errcode* 槽。因为 `DiskFailure` 事件类已针对设备名称定义槽，因此允许子类继承该槽，并且此继承可使您免除再次进行声明。槽定义为 `$1`，因此将正则表达式中的第一个子表达式分配给该槽。

但是，DiskFailureError 类还定义另一个子表达式。您可以将此子表达式分配给名为 errcode 的新槽并将其定义为 $2 以在正则表达式中引用第二个子表达式。在显示日志文本的先前示例中显示此类型的分配。

事件现在包含已分配值 /dev/sd0 的 device 槽和已分配值 13 的 errcode 槽。CustomSlot1 分配设备，而 CustomSlot2 分配错误代码。

还将继承 Performance Management 定制属性。有关 Performance Management 定制属性映射的更多信息，请参阅第 552 页的『格式文件规范』。

多行
使用多行语法将跨多行的记录与监视的日志中的模式相匹配。

将 \n 换行符指定为正则表达式的一部分以指示在监视日志中发生换行的位置。请参阅以下示例中此类型的语法：

```
REGEX REMultiLine
Line1:(.*)\nLine2(.*)
msg $1
second_msg $2
END
```

注： Windows 指定 \r\n 回车符和换行组合。

如果在日志文本中报告以下错误消息，那么将创建 REMultiLine 事件：

```
Line1: An error occurred
Line2: The error was "disk error"
```

msg 槽将分配值 An error occurred 而 second_msg 槽将分配值 The error was "disk error"。

映射
操作系统代理程序使用映射以确定系统日志消息的事件类。代理程序通过将消息与格式文件中的模式相匹配来确定事件类。

代理程序将日志消息转换为包含属性 name=value 对的事件类实例。然后，会将事件发送到事件服务器。

代理程序确定源的系统日志消息的事件类。代理程序通过将系统日志消息与格式文件中的模式相匹配来确定事件类。在使用此匹配过程来确定类后，必须将值分配到属性。

属性值来自各种源，例如：

· 代理程序提供的缺省值

· 在正则表达式中匹配特定子表达式的日志文本

映射语句包含在格式文件中并且由以下语法组成：

```
name    value CustomSlotn
```

此处，指定任何标识以描述槽的 name（也称为变量、属性或值标识）。然后，通过应用在第 557 页的『值说明符』中描述的任何值，指定值以分配到此槽。

使用定制槽以在 Performance Management 控制台中查看数据并定义阈值。在创建阈值时，所有定制槽值都是字符串。还需要定制槽以使重复检测生效，因为必须标识用于确定重复的槽。有关过滤事件的更多信息，请参阅第 863 页的『事件过滤和摘要』。msg 是特殊槽名，在事件表中具有自己的属性。您无需将定制槽用于 msg。

您可以限制槽的范围，从而使其仅存在于格式定义中。在定义槽时，槽名前置短划线，例如：

```
-name    value
```

通过此方式定义的任何槽不会包含在最终事件中。但是，您可以在格式定义中的任何位置引用槽，尤其是在 PRINTF 语句中。在下面的 REGenericSyslog 示例中，如果生成但未在 PRINTF 语句中引用 service 槽，那么将不包含此槽。其保留在定义时应用于原始槽的相同值，而无短划线。通过使用此过程，您可以使

用最终事件中未包含的格式定义中的临时变量。例如，您可以通过以下方式定义事件类 REGenericSyslog 以匹配通用 UNIX 系统日志事件：

```
REGEX REGenericSyslog
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2}) (.*?) (.*?): (.*)$
month $1
date $2
time $3
host $4
-service $5
msg $6
syslog_msg PRINTF("service %s reports %s", service, msg)
END
```

值说明符
格式规范中的映射向属性分配值。

格式规范的映射部分包含以下类型的说明符：

- $i
- 字符串常量
- PRINTF 语句

$i

i 指示格式字符串中子表达式的位置。每个子表达式的编号从 1 到格式字符串中子表达式的最大数量。

$i 值说明符（也称为变量、槽或属性）的值是对应的子表达式匹配的系统日志消息部分。

在以下示例中，日志代理程序将任何日志消息从 UNIX 系统日志工具转换为为其分配值的系统日志事件：

```
REGEX REGenericSyslog
^([A-Z][a-z]{2}) ([ 0-9][0-9]) ([0-9]{2}:[0-9]{2}:[0-9]{2})
 (.*?) (.*?): (.*)$
month   $1
date    $2
time    $3
host    $4
service $5
msg     $6
END
```

编号从 $1 到 $6 的每个子表达式匹配正则表达式中括号的项。

因此，以下系统日志事件：

```
Apr  6 10:03:20 jimmy syslogd 1.4.1: restart.
```

将分配以下值：

```
month=Apr
date=6
time=10:03:20
host=jimmy
service=syslogd 1.4.1
msg=restart.
```

例如，在系统日志事件中，10:03:20 值匹配正则表达式中括号内的第三项，因此将值分配给 *$3* 时间值。类似，jimmy 值匹配正则表达式中括号内的第四项，因此值分配给 *$4* 主机值。

字符串常量

字符串常量声明属性值是指定的字符串。如果属性值是不含空格的单个常量，那么指定而不用双引号（""）括起来，如以下示例中所示：

```
severity WARNING
```

否则，如果属性值中存在空格，那么必须使用双引号，如以下示例中所示：

```
component "Web Server"
```

PRINTF 语句

PRINTF 语句从其他属性值创建更复杂的属性值。PRINTF 语句包含关键字 PRINTF，后跟 printf() C 样式格式字符串以及一个或多个属性名。

格式字符串仅支持 %s 组件说明符。PRINTF 语句中使用的属性的值必须派生自 *$i* 值规范或常量字符串值规范（不能从其他 PRINTF 语句派生）。

使用参数属性的值以根据格式字符串编写新的常量字符串。这一新的常量字符串将成为属性值。

根据定义 REGenericSyslog 基类的前一个示例，以及 *service* 和 *msg* 槽，您可以使用 PRINTF 关键字定义名为 *syslog_msg* 的属性。

```
syslog_msg PRINTF("service %s reports %s", service, msg)
```

如果报告以下日志消息：

```
Apr  6 10:03:20 jimmy syslogd 1.4.1: restart.
```

将构成新常量字符串，其中包含来自格式字符串的属性值：

```
syslog_msg="service syslogd 1.4.1 reports restart."
```

关键字
在格式文件中，使用关键字以指定在运行时扩展的值。

在运行时扩展以下关键字：

- DEFAULT
- FILENAME
- LABEL
- REGEX

DEFAULT

使用 DEFAULT 关键字以将 DEFAULT 值分配给特定槽或属性。操作系统代理程序将内部缺省值分配给下表中描述的槽：

表 *191.* 槽和 *DEFAULT* 值

槽	**Description**
hostname	*hostname* 是代理程序运行所在的系统的短主机名。它不包含系统的域名。
origin	*origin* 是代理程序运行所在的系统的 IP 地址。
fqhostname	*fqhostname* 是代理程序运行所在的系统的标准主机名。其包含系统域名。
RemoteHost	当事件源自本地系统时，此特性为空。如果事件源自远程系统，*RemoteHost* 包含 *user@host:port* 格式的字符串，这指示发生事件的远程主机名以及将用于连接的用户和此主机上的端口。

分配给 *fqhostname* 的值受 .conf 文件中以下 FQDomain（可选）设置的影响：

- 如果将 FQDomain 设置为 yes，那么代理程序自己确定系统域名。

- 如果未设置 FQDomain 的值或者将值设置为 no，那么代理程序不会设置域名，并且会为 *fqhostname* 属性分配空字符串。
- 如果设置 FQDomain 从而不包含 yes 或 no 值，那么接受域名作为值并且附加到主机名。

在以下示例中，格式定义包含三个属性和槽：

- *hostname* DEFAULT
- *origin* DEFAULT
- *fqhostname* DEFAULT

如果在 .conf 文件中将 FQDomain 设置为 yes，那么使用以下属性在计算机上运行：

- *hostname*：myhost
- *IP address*：192.168.1.100
- *domainname*：mycompany.com

创建事件并为三个槽分配以下值：

```
"hostname=myhost", "origin=192.168.1.100", "fqhostname=myhost.mycompany.com"
```

FILENAME

FILENAME 关键字指示包含消息的日志文件的标准文件名（包括路径）。如果使用单个代理程序来监视多个日志文件，并且需要标识事件源，请使用此关键字以使用文件名填充事件属性。如果消息来自于系统日志，那么对于 Windows 操作系统代理程序映射设置为 EventLog，并且对于 UNIX 操作系统代理程序设置为 SysLogD。

注: 路径包含此关键字的属性。

LABEL
LABEL 关键字指定代理程序运行所在的系统的主机名。

REGEX
REGEX 关键字扩展为匹配消息且导致事件的正则表达式。

最大消息长度
该值是操作系统代理程序可接收而不截断消息的最大消息长度。

对于 Performance Management 和 Tivoli Netcool/OMNIbus，最大消息长度不同。

Performance Management
对于发送到 Performance Management 的事件，msg 属性限制为 2048 字节。将截断超过此长度的消息。

Tivoli Netcool/OMNIbus
对于通过 Probe for Tivoli EIF 发送到 Netcool/OMNIbus 的事件，事件的总大小（包括类名、所有槽及其值）不能超过 4096 字节。例如，在以下样本 EIF 事件中，不会针对 4096 字节限制计算 ;END。但是，针对限制计算所有其他项，包括语法元素，例如，分号、引号和等号。

```
Class;attr1=value1;attr2=value2;msg='Hello, world';END
```

远程日志文件监视：加密密码或口令
为提高安全性，在使用 远程日志文件监视 时，您可以加密传输到远程系统的密码和口令。

关于此任务
加密密码和口令存储在配置 (.conf) 文件中。有关配置文件的更多信息，请参阅第 545 页的『配置文件』。

过程

- 运行 **itmpwdsnmp** 命令并提供将加密的密码或口令：

- Linux AIX 此命令从 Cloud APM 安装目录运行。缺省安装路径为 opt/ibm/apm/agent，其中 *install_dir* 是安装代理程序的位置。
- Windows 缺省安装路径为 C:\IBM\APM。

Linux 在 Linux 系统上运行时命令的示例：

```
$ export install_dir=/opt/ibm/apm/agent/bin
$ /opt/ibm/apm/agent/bin

输入要加密的字符串：
mypassword

确认字符串：
mypassword

{AES256:keyfile:a}Z7BS23aupYqwlXb1Gh+weg==
$
```

在示例中，{AES256:keyfile:a}Z7BS23aupYqwlXb1Gh+weg== 命令的整个输出用于在代理程序配置文件中设置 **SshPassword**。{AES256:keyfile:a} 前缀告知代理程序密码加密。

要加密专用密钥文件的口令，请遵循相同过程。

配置操作系统代理程序定制脚本编制

Monitoring Agent for Linux OS、Monitoring Agent for UNIX OS 和 Monitoring Agent for Windows OS 代理程序自动进行配置。此功能允许用户定义脚本，以便按定义的频率在操作系统代理程序上运行。

缺省情况下，启用定制脚本编制功能。管理员可以通过在代理尘谷配置文件中设置新的环境变量 *KXX_FCP_SCRIPT*=true/false（缺省值为 true）来启用/禁用该功能，其中 XX 可以是以下值之一：

- LZ，对应 Monitoring Agent for Linux OS

- UX，对应 Monitoring Agent for UNIX OS

- NT，对应 Monitoring Agent for Windows OS

在以下部分中提供了详细信息。

定制脚本编制快速入门

为操作系统代理程序添加定制脚本编制，以定义脚本按定义的频率在操作系统代理程序上运行。

操作系统代理程序启动后，就会立即使用缺省值启用此功能。启动脚本编制功能的唯一操作是：

在缺省目录（在 Linux™ 或 UNIX™ 上，此目录为 install_dir/localconfig/*product code*/scripts_definitions，在 Windows™ 上，此目录为 install_dir\localconfig\nt\scripts_definitions）下通过使用提供的模板 script_property.txt 作为示例来创建属性文件。

只需以下两个属性：

ATTRIBUTE_NAME
用于唯一标识属性文件内部的脚本定义的任意名称。

SCRIPT_PATH_WITH_PARMS
含自变量的脚本的标准路径。

shell 脚本、Perl 以及其他类型的脚本均可供使用。指定在 SCRIPT_PATH_WITH_PARMS 属性中运行的完整命令。

例如，perl C:\IBM\scripts\Custom_Scripts\date.pl。在此示例中，通过代理程序环境中的 PATH 变量来确保此代理程序可解析 perl 的位置。指定以其他方式执行 perl 安装的位置的完整路径。

操作系统代理程序环境文件中的参数

您可以在操作系统代理程序环境文件中设置用于定制脚本编制的参数。

可通过设置操作系统代理程序环境文件中的参数来定制脚本编制功能：

***install dir*/config/lz.environment**
Monitoring Agent for Linux OS 的环境文件。

***install dir*/config/ux.environment**
Monitoring Agent for UNIX OS 的环境文件。

***install dir*\TMAITM6_x64\KNTENV**
64 位 Monitoring Agent for Windows OS 的环境文件。

***install dir*\TMAITM6\KNTENV**
32 位 Monitoring Agent for Windows OS 的环境文件。

KXX_FCP_SCRIPT
缺省情况下，启用脚本编制功能。要将其禁用，请设置：KXX_FCP_SCRIPT=false

其他参数可根据特定需求在代理程序环境文件内进行定义：

KXX_FCP_SCRIPT_DEFINITIONS
属性文件的存储位置。

Linux™ 或 UNIX™ 上的缺省位置为 *install dir*/localconfig/*PC*/scripts_definitions，在 Windows™ 上，位置为 *install dir*\localconfig\nt\scripts_definitions

KXX_FCP_SCRIPT_INTERVAL

操作系统代理程序使用该变量的值作为循环时间间隔（秒）以检查运行中脚本的执行情况，如果满足过滤条件，它就会发送事件。最小值为 30 秒，最大值为 300 秒。无效值将重置为缺省值。缺省值为 60 秒。

注: 如果 KXX_FCP_SCRIPT_SYNC_INTERVALS 设置为 USE_SCRIPT（参阅 KXX_FCP_SCRIPT_SYNC_INTERVALS 的定义），将忽略此参数。

KXX_FCP_SCRIPT_SYNC_INTERVALS

如果 KXX_FCP_SCRIPT_INTERVAL 定义的代理程序循环时间间隔大于脚本执行频率，那么部分脚本执行循环生成的数据可能会丢失。为避免发生此行为，可通过将 KXX_FCP_SCRIPT_SYNC_INTERVALS 设置为以下值来将脚本执行频率与代理程序循环时间间隔进行同步：

- USE_AGENT - 将每个脚本执行频率的值都强制设置为 KXX_FCP_SCRIPT_INTERVAL 与其属性文件中定义的 EXECUTION_FREQUENCY 之间的最大值。
- USE_SCRIPT - 代理程序循环时间间隔将动态设置为所有已定义的脚本之间的最小频率值（属性文件中的 EXECUTION_FREQUENCY）。这将忽略 KXX_FCP_SCRIPT_INTERVAL 设置的值。脚本的频率保持不变，仍为属性文件中定义的值。设置 USE_SCRIPT 时，每次添加、更改或移除脚本定义时，代理程序循环时间间隔都可能发生更改。无论如何，它都不能低于 KXX_FCP_OVERRIDE_MIN_FREQUENCY_LIMIT 设置的值，也不能大于 300 秒。
- NO - 不执行同步，某些执行结果可能会丢失。

KXX_FCP_SCRIPT_DEFINITIONS_CHECK_INTERVAL

在启动时以及由该变量定义的每个时间间隔内，操作系统代理程序都会检查脚本或属性文件中是否存在任何更改。请注意，如果 KXX_FCP_SCRIPT_DEFINITIONS_CHECK_INTERVAL 小于代理程序循环时间间隔，那么它将重置为代理程序循环时间间隔。允许的最大值为缺省值 300 秒。

KXX_FCP_USER

此参数仅在 Linux™ 或 UNIX™ 操作系统代理上才有效。它可定义用于创建 fcp_deamon 进程的 user，前提是该用户不同于操作系统代理程序进程用户；后者负责运行所有脚本。请注意，操作系统代理程序的所有者必须具备正确的许可权才能创建 fcp_daemon 进程。在 Windows™ 上，定义的用户必须不同于服务 Monitoring Agent for Windows OS 的登录用户“FCProvider”。此用户必须具有对代理程序安装目录和脚本存储库目录的“完全控制”许可权。有关更多信息，请参阅：

https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/install/install_linuxaix_agent_nonroot.html

KXX_FCP_MAX_CDP_DP_THREAD_POOL_SIZE
它可定义要运行的脚本的最大并行数量。最大值为 32。

KXX_FCP_MAX_DAEMON_RESTARTS

操作系统代理程序会监视 fcp_daemon：如果发生进程异常退出的情况，那么操作系统代理程序会将其重新启动。缺省值为 4。针对 KXX_FCP_MAX_DAEMON_RESTARTS 执行重新启动（每天多次）。必须使用值 0 以避免重新启动；如果设置了 -1，那么操作系统代理程序将不断重试重新启动 fcp_daemon。重新启动计数器将在操作系统代理程序重新启动时重置。

KXX_FCP_SEND_SCRIPT_RUNTIME_EVENTS

缺省值为 true。如果设置为 false，那么操作系统代理程序将停止发送对应每个脚本标准输出的每一行的事件。在此情况下，在控制台工作空间上可显示脚本输出，但不显示任何情境，并且不收集任何历史数据。

KXX_FCP_OVERRIDE_MIN_FREQUENCY_LIMIT

当 KXX_FCP_SCRIPT_SYNC_INTERVALS 设置为 USE_SCRIPT 时使用此设置。在此条件下，它会设置操作系统代理程序循环时间间隔的最小值。

针对操作系统代理程序循环时间间隔使用较低的值（小于 5 秒）是极具侵犯性的操作，可能影响操作系统代理程序的性能。如果需要频繁收集数据（例如，每一秒），那么建议对脚本进行定制。脚本会按所需的频率来缓存数据，并按较高的时间间隔（例如，每 60 秒）向操作系统代理程序返回收集的数据。

以下 Agent Builder（CDP）变量同样可用于控制 fcp_daemon 的行为：

CDP_DP_REFRESH_INTERVAL

（缺省值 60 秒）全局脚本调度开始时间。在脚本属性文件中频率未获得批准时使用。

CDP_DP_SCRIPT_TIMEOUT

（缺省值 30 秒）全局脚本执行最长时间。当脚本的执行时间超过此限制时，其 Status_Code 将设置为 TIMEOUT

CDP_DP_KILL_ORPHAN_SCRIPTS

（Y|N - 缺省值为 N）供 fcp_daemon 进程用于超时脚本的全局行为。设置为“Y”时，脚本将终止，否则将被放弃。如果在脚本属性文件中设置了 KILL_AFTER_TIMEOUT 键，那么针对特定脚本忽略该值。

CDP_MAXIMUM_ROW_COUNT_FOR_CPCI_DATA_RESPONSES

（缺省值为 1000）因性能原因添加全局值，用于限制脚本返回的最大输出行数。忽略超出此限制的行。允许的值为正整数。无效值将更改为无限制。

fcp_daemon 还支持用于控制 Agent Builder 代理程序的其他环境变量。有关完整列表，请参阅此处的《IBM Agent Builder 用户指南》： 第 35 页的『第 2 章 PDF 文档』.

属性文件中的参数

您可在属性文件中设置用于定制脚本编制的参数。

KXX_FCP_SCRIPT_DEFINITIONS 目录包含 *.properties 文件的列表。每个属性文件都包含脚本列表，用于配合 key=value 形式的相应属性一起运行。可定义（不区分大小写）的属性包括：

ATTRIBUTE_NAME

必需 - 字符串，长度不超过 256 个字符。名称由您选择，用于定义特定脚本及其属性。可用于 ATTRIBUTE_NAME 名称的字符包括字母数字字符和特殊字符（只能使用下划线作为特殊字符）。如果使用其他特殊字符（空格或空白），这些特殊字符会被转换为下划线 (_)。在同一个属性文件内列出多个脚本时，必须定义不同 ATTRIBUTE_NAME（每个脚本一个名称）。此名称必须是针对每个已定义的脚本指定的第一个值，并用于划定特定脚本的属性集开始位置到下一个 ATTRIBUTE_NAME 为止的界限。

SCRIPT_PATH_WITH_PARMS

必需 - 字符串，长度不超过 512 个字符。此参数用于定义指向含参数（以空格分隔）的脚本的完整路径。在脚本路径名中不得使用任何特殊字符。包含空格的值必须以单引号 (') 或双引号 (") 括起。可以传递变量，但是针对所有操作系统必须以 ${...} 括起。环境变量必须包含在操作系统代理程序进程上下文中。

EXECUTION_FREQUENCY

可选 - 缺省值为 60 秒。此参数用于定义脚本执行频率。

CUSTOM_NAME

可选 - 字符串，长度不超过 256 个字符。此参数可用于脚本的描述。

IS_ACTIVE
可选 - true|false，缺省值为 true。它用于激活脚本。如果值为 false，那么不会运行脚本。

DISABLE_USE_AGENT_SYNC
可选 - true|false，缺省值为 false。如果值为 true，那么会同时遵循脚本的 EXECUTION_FREQUENCY，即使全局变量 KXX_FCP_SCRIPT_SYNC_INTERVALS 设置为 USE_AGENT 也是如此。

KILL_AFTER_TIMEOUT
可选 - true|false，缺省值由 CDP_DP_KILL_ORPHAN_SCRIPTS 变量定义。当值为 true 时，脚本在超时后结束。当脚本执行时间大于操作系统代理程序配置文件中的 CDP_DP_SCRIPT_TIMEOUT 参数指定的值时，就会发生超时。否则，忽略该值。在这两种情况下，都不会收集数据。请注意，当设置 KILL_AFTER_TIMEOUT 时，属性文件中定义的唯一脚本将终止，但脚本创建的子进程（如果有）不会终止。Solaris™ 和 Windows™ 32 位操作系统代理程序不支持此功能，并且所有超时脚本都将被放弃。

对于脚本返回的输出行将进行解析。
脚本返回标准输出（称为第一个令牌）。当脚本在输出行中返回更多值时，会为其添加更多令牌。遵循预定义的语法，最多返回五个字符串、五个整数和五个浮点值。

OUTPUT_TYPE

STRING|INTEGER|FLOAT - 可选 - 缺省值为字符串 (string)。它可定义脚本每一行返回的第一个令牌的类型；OUTPUT_TYPE 可以是：

- STRING（缺省值）- 字符串，长度不超过 2048 个字符。使用此类型时，第一个令牌会填充 KXX_Custom_Scripts_Rtm_Smp 的“Standard_Output_String”属性。
- INTEGER - 允许获取数字值，范围介于 -9223372036854775806 和 9223372036854775806 之间。使用此类型时，第一个令牌会填充 KXX_Custom_Scripts_Rtm_Smp 的“Standard_Output_Integer”属性。
- FLOAT - 允许获取数字值，范围介于 -92233720368547758.06 和 92233720368547758.06 之间，精度为两位小数。使用此类型时，第一个令牌会填充 KXX_Custom_Scripts_Rtm_Smp 的“Standard_Output_Float”属性。

TOKEN_TYPES
STRING|INTEGER|FLOAT - 可选 - 用于定义第一个令牌后的其他令牌的输出类型。用户可以定义最多五个字符串、五个整数和五个浮点值。它显示为以逗号分隔的类型列表：<token_type>,<token_type>,... token_type 可为空或以下任一值（不区分大小写）：

- - STRING 或 S
- - INTEGER 或 I
- - FLOAT 或 F
- 如果 *TOKEN_TYPES* 为空，那么将跳过对应的令牌。

同样有效的布局示例如下所示：

- - TOKEN_TYPES=S,I,S,,,F,,F,F
- - TOKEN_TYPES=String,integer,S,,,Float,,f,FLOAT

TOKEN_LABELS
STRING - 可选 - 每个标签最多 16 个字符。它用于定义 TOKEN_TYPES 中定义的令牌标签。该值为以逗号分隔的令牌标签列表，必须与 TOKEN_TYPES 定义的令牌对应。例如：

- TOKEN_TYPES=S,I,S,,,F,,F,F
- TOKEN_LABELS=Cpu Name,Cpu number,Description,,,value 1,,value 2,value 3
- 如果未设置 TOKEN_TYPES，那么忽略 TOKEN_LABELS。

TOKEN_SEPARATOR
可选 - 缺省值为分号“;”，用于设置字符串，以用作为拆分令牌中的输出行的分隔符。如果未设置 TOKEN_TYPES，那么忽略该值。可接受空值（空格）作为分隔符，输出行中的多个连续空格被视为单个空格。

以下两个参数允许过滤脚本的行输出。操作系统代理程序仅将这两个参数应用于第一个令牌，并且两者必须配合一起使用。

FILTER_VALUE
可选。用于比较的值。如果定义 FILTER_OPERATOR，那么该值为必需。如果 OUTPUT_TYPE 为字符串，过滤值必须精确反映要过滤的脚本返回的字符串值，不含任何附加引号（不允许通配符）。

FILTER_OPERATOR
可选。用于比较的运算符。如果定义 FILTER_VALUE，那么该值为必需。可接受的 FILTER_OPERATOR 值包括：

- =（等于）
- !=（不等于）
- >（大于），仅适用于数字类型
- >=（不小于），仅适用于数字类型
- <（小于），仅适用于数字类型
- <= （不大于），仅适用于数字类型

属性文件示例
在属性文件中设置参数的示例。

#第一个脚本定义：脚本 ex_script1.sh 每隔 150 秒启动。它会返回浮点值，并且代理程序仅考虑等于 0.5 的输出行。

```
ATTRIBUTE_NAME=sample1
SCRIPT_PATH_WITH_PARMS=/opt/ibm/apm/agent/localconfig/lz/scripts_definitions/ex_script1.sh
EXECUTION_FREQUENCY=150
OUTPUT_TYPE=FLOAT
FILTER_VALUE=0.5
FILTER_OPERATOR==
```

#第二个脚本定义：脚本 ex_script2 每隔 60 秒启动。它会返回整数值，并且代理程序仅考虑不等于 0 的行。

```
ATTRIBUTE_NAME=ex_script2
SCRIPT_PATH_WITH_PARMS=${CANDLE_HOME}/tmp/check_out.sh
EXECUTION_FREQUENCY=60
OUTPUT_TYPE=INTEGER
FILTER_VALUE=0
FILTER_OPERATOR=!=
```

#第三个脚本定义：脚本 ex_script3.sh 每隔 120 秒启动，含三个输入参数（第一个输入参数为整数，第二个和第三个输入参数为字符串）。如果它挂起或者执行时间大于超时值，那么它就会终止。

```
ATTRIBUTE_NAME=ex_script3
SCRIPT_PATH_WITH_PARMS=/opt/scripts/ex_script3.sh 1 "second input parameter" "third input
parameter"
EXECUTION_FREQUENCY=120
OUTPUT_TYPE=STRING
KILL_AFTER_TIMEOUT=TRUE
```

#第四个脚本定义：脚本 cpu_mem_percentage.sh 每隔 50 秒启动，并返回 cpuid 作为标准输出字符串，返回两个浮点值（分别表示“空闲”和“已用 CPU 百分比”），并返回两个整数（分别表示“内存”和“虚拟内存”使用情况）。管道字符用作为解析输出的分隔符。必须由该脚本返回的行示例如下所示：

cpu2|35,5|65,5|3443|123800

```
ATTRIBUTE_NAME=cpu and mem Usage
SCRIPT_PATH_WITH_PARMS=${SCRIPT_HOME}/cpu_mem_percentage.sh
OUTPUT_TYPE=STRING
TOKEN_TYPES=F,F,I,I
TOKEN_LABELS= Idle CPU %, Used CPU %, Virt MEM used MB, MEM used MB
TOKEN_SEPARATOR=|
EXECUTION_FREQUENCY=50
```

已知问题和局限性
已知问题和局限性

- 在 Windows™ 2003 64 位系统上不支持“脚本编制功能”。

- 在 Solaris™ 和 Windows™ 32 位操作系统代理程序上，超时后终止无效。
- 如果某些脚本在超时周期内未完成，并且用户启用了密集跟踪，那么 fcp_daemon 可停止 Windows™ 32 位中运行的脚本。如果 fcp_daemon 停止运行脚本，控制台上报告的数据会反映上次运行脚本的时间。操作系统代理程序也可能停止返回数据。停止 fcp_daemon 进程允许代理程序恢复正常操作。
- 在 Solaris™ 上返回 SCRIPT_NONZERO_RETURN，而不是 SCRIPT_NOT_FOUND 或 SCRIPT_LAUNCH_ERROR。
- 脚本编制功能不提供完全全球化；在属性文件或脚本输出中使用本地化字符时可能会遇到问题。
- 在 Windows™ 操作系统代理程序上，无法运行驻留在映射的网络驱动器上的脚本。
- 升级 Windows™ 操作系统代理程序时，缺省情况下不启用脚本编制功能。请编辑 KNTENV 并将 `KNT_FCP_SCRIPT=FALSE` 更改为 `KNT_FCP_SCRIPT=TRUE`

对定制脚本编制进行故障诊断
对定制脚本编制进行故障诊断

标准 *KBB_RAS1* 变量适用于操作系统代理程序和 fcp_daemon 进程。要将特定跟踪设置仅应用于 fcp_daemon，请使用 *KXX_FCP_KBB_RAS1* 变量；如果已设置 *KXX_FCP_KBB_RAS1*，fcp_daemon 将忽略由 *KBB_RAS1* 指定的值。

要跟踪由该功能的操作系统代理程序核心线程记录的操作：

```
KBB_RAS1=ERROR (UNIT:factory ALL)
```

要从 APM 服务器跟踪脚本编制查询和发送到该服务器的事件，请添加以下条目：

在 Monitoring Agent for Linux OS 上

```
(UNIT:klz34 ALL) (UNIT:klz36 ALL)
```

在 Monitoring Agent for UNIX OS 上

```
(UNIT:kux48 ALL) (UNIT:kux50 ALL)
```

在 Monitoring Agent for Windows OS 上

```
(UNIT:knt84 ALL) (UNIT:knt86 ALL)
```

要查看 TEMA 跟踪以验证专用情境的执行情况，请添加以下条目：

```
(UNIT:kraavp all) (UNIT:kraapv all)
```

要查看脚本执行情况并了解当前正在解析的来自脚本的数据量，请设置：

```
KXX_FCP_KBB_RAS1=Error (UNIT:command ALL)
```

要对操作系统代理程序与 fcp_daemon 之间的通信中的问题进行故障诊断，请向 *KBB_RAS1* 和 *KXX_FCP_KBB_RAS1* 添加此跟踪级别：

```
(UNIT:cps_socket FLOW) (UNIT:cpci FLOW)
```

要详细查看操作系统代理程序进程与 fcp_daemon 之间的交互，请向 *KBB_RAS1* 和 *KXX_FCP_KBB_RAS1* 添加以下内容：

```
(UNIT:cps_socket ALL) (UNIT:cpci ALL)
```

快速入门方案
本部分描述了为示例场景配置定制脚本编制所需的最少步骤。

以下部分描述了配置 Monitoring Agent for Linux OS 以运行两个定制脚本所需的最少步骤。

定制脚本描述

在此示例中，用户具有两个脚本，位于 /scripts_repo 目录下：

checkDIRsize.sh - 此脚本用于检查作为输入参数传递的指定目录大小。输出为整数，例如：4594740

cpu_mem_usage.sh - 此脚本用于检查已用 CPU 百分比和已用交换内存 (MB)。输出以如下格式返回：cpu1|96,5|23800

其中，第一个令牌为 CPU 标识，第二个令牌为已用 CPU 百分比，第三个令牌为已用交换内存 (MB)。

定制需要 Monitoring Agent for Linux OS 运行这些脚本。

操作系统代理程序启动后，就会立即使用缺省值启用此功能：

在缺省目录 `install dir/localconfig/lz/scripts_definitions` 下创建属性文件 `AnyName.properties`。在此示例中，创建两个属性文件，分别对应名为 checkDIRsize.properties 的脚本和名为 cpu_mem_usage.properties 的脚本：

```
#CheckDIRsize.properties
ATTRIBUTE_NAME=OPT_DIR_SIZE
SCRIPT_PATH_WITH_PARMS=/scripts_repo/checkDIRsize.sh /opt
EXECUTION_FREQUENCY=20
OUTPUT_TYPE=INTEGER
```

```
#cpu_mem_usage.properties
ATTRIBUTE_NAME=cpu_mem_usage
SCRIPT_PATH_WITH_PARMS=/scripts_repo/cpu_mem_percentage.sh
OUTPUT_TYPE=string
TOKEN_TYPES=F,I
TOKEN_LABELS= Used CPU %, Swap MEM used MB
TOKEN_SEPARATOR=|
EXECUTION_FREQUENCY=10
```

添加（或更改）两个属性文件后，无需重新启动操作系统代理程序。操作系统代理程序会检查具有指定时间间隔（缺省值为 300 秒）的脚本定义目录。打开控制台，在“定制脚本”工作空间下，会显示脚本详细信息和结果。

配置 Linux OS Agent 文件系统数据收集

Monitoring Agent for Linux OS 采用自动配置。但您可以配置文件系统数据收集行为。

Monitoring Agent for Linux OS 具有对应文件系统数据收集的缺省行为。

缺省行为是仅监视来自 `/etc/fstab` 的文件系统。在 `lz.environment` 文件中定义了环境变量 *KBB_SHOW_MTAB_FS*，用于控制文件系统数据收集行为。如果要监视 `/etc/fstab` 和 `/etc/mtab` 中列出的所有文件系统，可以设置 KBB_SHOW_MTAB_FS=true。

KBB_SHOW_MTAB_FS
此变量在 *install_dir*`/config/.lz.environment` 文件中可用。缺省值为 `false`，用于定义代理程序仅监视来自 `/etc/fstab` 的文件系统。如果要监视 `/etc/fstab` 和 `/etc/mtab` 中列出的所有文件系统，请将值切换至 `true`。例如， *KBB_SHOW_MTAB_FS=true*。

配置 PHP 监视

您必须配置 Monitoring Agent for PHP，以便代理程序可从正受监视的 PHP 应用程序收集数据。

开始之前

1. 确保安装 PHP 流程包。如果使用 `yum install` 命令安装 PHP，那么运行 `yum install php-process` 命令以安装 PHP 流程包。
2. 配置代理程序之前，确保 Apache HTTPD 服务器已启动。

   打开 Apache HTTP Server `httpd.conf` 配置文件，并确保 `mod_status` 和 `ExtendedStatus On` 选项均已启用。例如：

   ```
   ExtendedStatus On
   <Location /server-status>
           SetHandler server-status
           Order deny,allow
   ```

```
        Allow from all
        Allow from 127.0.0.1
</Location>
```

在给定的示例中，`http://127.0.0.1/server-status` 必须工作正常才能使代理程序正常工作。

注: 您必须在 Linux 上安装 Lynx 或 Links，代理程序才能获取监视数据。

确保在不对 `apachectl` 命令进行任何代码更改的情况下，命令 `apachectl status` 在受监视的 Apache 服务器中工作正常。必须安装 Lynx，命令 `apachectl status` 才能正常工作。

关于此任务

要避免配置代理程序时发生许可权问题，请确保使用相同的 root 用户或用于安装代理程序的非 root 用户。如果作为所选用户身份安装代理程序并以另一用户身份配置代理程序，请参阅第 156 页的『以非 root 用户身份配置代理程序』。如果以一个选中的用户身份安装和配置了代理程序，并且想要以其他用户身份启动代理程序，请参阅第 859 页的『以非 root 用户身份启动代理程序』。

PHP 代理程序是一个多实例代理程序；必须手动创建第一个实例并启动代理程序。“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。“受管系统名称”限制为 32 个字符。指定的实例名称限制为 28 个字符（减去主机名长度后）。例如，如果指定 PHP2 作为实例名称，那么受管系统名称为 PHP2:hostname:PJ。

要点: 如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法正确显示。

过程

- 如果环境与缺省设置相同，那么可以使用缺省执行二进制路径、缺省 `php.ini` 文件以及缺省端口来配置代理程序:
  a) 输入:
  install_dir/bin/php-agent.sh config *instance_name* *install_dir*/samples/php_silent_config.txt
  其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 PHP 代理程序安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
  b) 要启动代理程序，请输入:
  install_dir/bin/php-agent.sh start *instance_name*
- 要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请完成以下步骤:
  a) 在文本编辑器中打开 *install_dir*/samples/php_silent_config.txt。
  b) 对于 **Location of PHP execution binary**，可以指定 PHP 执行所在的目录。缺省位置为 `/usr/local/bin`。
  c) 对于 **Location of PHP INI file**，可以指定 `php.ini` 文件所在的目录。缺省位置为 `/etc`。
  d) 对于 **Web server port**，可以指定运行 WordPress 的 Web 服务器的端口号。缺省值为 80。
  e) 对于 **Application DocumentRoot**，可以指定 PHP WordPress 应用程序的 DocumentRoot。使用冒号分隔多个记录。要允许代理程序找到所有记录，请使用缺省值 ALL。
  f) 保存并关闭 `php_silent_config.txt` 文件，然后输入:
  install_dir/bin/php-agent.sh config *instance_name* *install_dir*/samples/php_silent_config.txt
  其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 PHP 代理程序安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
  g) 要启动代理程序，请输入:
  install_dir/bin/php-agent.sh start *instance_name*
- 要通过运行脚本并响应提示来配置代理程序，请完成下列步骤:
  a) 输入:
  install_dir/bin/php-agent.sh config *instance_name*
  其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 PHP 代理程序安装目录。

b) 当出现 `Edit Monitoring Agent for PHP settings` 提示时，输入 1 以继续。

c) 当出现 `Location of PHP execution binary` 提示时，按 Enter 键以接受缺省位置或指定自己的位置。

d) 当出现 `Location of PHP INI file` 提示时，按 Enter 键以接受缺省位置或指定自己的位置。

e) 当出现 `Web server port` 提示时，按 Enter 键以接受缺省端口或指定其他端口号。

f) 当出现 `Application DocumentRoot` 提示时，按 Enter 键以接受缺省值或指定 PHP WordPress 应用程序的 DocumentRoot。可以使用冒号分隔多个记录。

g) 要启动代理程序，请输入：
install_dir`/bin/php-agent.sh start` *instance_name*

结果

代理程序仅评估 WordPress 应用程序中 PHP 请求的性能。不会对 CSS 和 JS 装入进行评估。代理程序不会使用 URL 参数来标识 URL。

下一步做什么

可以验证 PHP 代理程序数据是否显示在 Cloud APM 控制台中。

必须确保代理程序的 WordPress 插件已激活。要确保激活，请完成以下步骤：

1. 在 Web 浏览器中，输入以下 URL：`http://`*hostname*`:`*port*`/wp-admin/`。
2. 通过浏览到**插件** > **安装的插件**来访问管理页面。
3. 确保 PHP 代理程序插件已激活。PHP 代理程序插件会列为 **WordPress 代理程序**。通常，该插件已激活。如果尚未激活，请单击**激活**。

配置 PostgreSQL 监视

您必须配置 Monitoring Agent for PostgreSQL，以便代理程序可从正受监视的 PostgreSQL 数据库收集数据。

开始之前

必须先安装 PostgreSQL JDBC 驱动程序，然后再安装此代理程序。代理程序配置时需要此驱动程序的路径。

JDBC 第 4 类驱动程序是新版本，因此最好使用此版本。用户可以根据代理程序所使用的 JDK 版本来安装 JDBC 4 版本的子类型。要了解 JDBC 版本到 JDK 版本的映射，请从 https://jdbc.postgresql.org/download.html 获取更多信息。

代理程序收集的一些属性依赖于 `pg_stat_statements` 扩展。要添加 `pg_stat_statements`，请先安装 `postgresql-contrib` 程序包。必须修改 `postgresql.conf` 配置文件以使 PostgreSQL 服务器装入 `pg_stat_statements` 扩展。

1. 在文本编辑器中打开 `postgresql.conf` 文件，并更新 `shared_preload_libraries` 行：

```
shared_preload_libraries = 'pg_stat_statements'
pg_stat_statements.track_utility = false
```

除实用程序命令以外，监视 SQL 语句需要这些更改。

注: `pg_stat_statements.track_utility` 的状态仅由超级用户设置或修改。

2. 在更新并保存 `postgresql.conf` 后，重新启动 PostgreSQL 服务器。
3. 使用 psql 运行以下 SQL 命令，它应该与稍后将在用于 JDBC 连接的代理程序配置中提供的数据库相连接：

```
create extension pg_stat_statements;
select pg_stat_statements_reset();
```

注: 命令 `create extension` 和函数 `pg_stat_statements_reset()` 仅由超级用户运行。

视图 `pg_stat_statements` 需要对特定数据库启用，有关更多详细信息，请参阅 https://www.postgresql.org/docs/9.6/static/pgstatstatements.html。

`pg_hba.conf` 文件是包含认证设置的 PostgreSQL 数据库文件。在 `pg_hba.conf` 文件中将 `auth-method` 参数值设置为 `ident` 时，PostgreSQL 代理程序 无法连接到 PostgreSQL 数据库。确保 `auth-method` 参数的认证设置正确。例如，可以将 `auth-method` 参数的这些值设置为：`md5`、`trust` 或 `password`。

查看必备硬件和软件。有关最新系统需求信息，请参阅 PostgreSQL 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

PostgreSQL 代理程序是一个多实例代理程序；您必须手动创建第一个实例并启动代理程序。“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。“受管系统名称”限制为 32 个字符。指定的实例名称限制为 28 个字符（减去主机名长度后）。例如，如果指定 PostgreSQL2 作为实例名称，那么受管系统名称为 PostgreSQL2:hostname:PN。

要点: 如果指定长实例名称，那么会截断受管系统名称，并且也不会完全显示代理程序代码。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for PostgreSQL**，然后单击**配置代理程序**。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。
4. 在 **Monitoring Agent for PostgreSQL** 窗口中，完成以下步骤：
   a. 在 **IP 地址**字段中，输入要远程监视的 PostgreSQL 服务器的 IP 地址。如果在要监视的服务器上安装代理程序，那么保留缺省值。

      注:

      对于远程监视，**目前已用 CPU (%)** 和**已用物理内存 (MB)** 的数据在仪表板上将不可用。这些窗口小部件将显示**不适用**。
   b. 在 **JDBC 数据库名称**字段中，输入数据库名称以更改缺省数据库名称 `postgres`。
   c. 在 **JDBC 用户名**字段中，输入用户名以更改缺省名称 `postgres`。
   d. 在 **JDBC 密码**字段中，输入 JDBC 用户密码。
   e. 在**确认 JDBC 密码**字段中，重新输入密码。
   f. 在 **JDBC 端口号**字段中，输入端口号以更改缺省端口号 5432。
   g. 在 **JDBC JAR 文件**字段中，为 PostgreSQL 连接器输入 Java JAR 文件的路径，然后单击**下一步**。
   h. 在 **Java 跟踪级别**字段中，根据 IBM 支持指示信息输入跟踪级别。缺省跟踪级别为 `Error`。
   i. 单击**确定**。代理程序实例将显示在 IBM Performance Management 窗口中。
5. 右键单击 **Monitoring Agent for PostgreSQL** 实例，然后单击**启动**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

过程

1. 在命令行上，输入以下命令：
   install_dir/bin/postgresql-agent.sh config *instance_name*
2. 在提示编辑 PostgreSQL 设置的代理程序时，请输入 1 以继续。
3. 在提示输入以下参数的值时，请按 Enter 键以接受缺省值，或者指定其他值并按 Enter 键：
   - IP 地址

     注:

     确保要远程监视 PostgreSQL 服务器的 IP 地址。如果在要监视的服务器上安装代理程序，那么保留缺省值。

     对于远程监视，**目前已用 CPU (%)** 和**已用物理内存 (MB)** 的数据在仪表板上将不可用。这些窗口小部件将显示**不适用**。
   - JDBC 数据库名称
   - JDBC 用户名
   - JDBC 密码
   - JDBC 端口号
   - JDBC JAR 文件

   要点: JDBC JAR 文件的版本必须与监视的 PostgreSQL 数据库的版本相同。
4. 在提示输入 `Java trace level` 参数的值时，输入 2 以接收缺省值，或者根据 IBM 支持指示信息指定跟踪级别。
5. 运行以下命令以启动代理程序：

```
install_dir/bin/postgresql-agent.sh
start instance_name
```

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件来修改这些参数值，并运行配置脚本来创建实例和更新代理程序配置值。这种配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可使用静默响应文件来配置 PostgreSQL 代理程序。在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

- 要通过编辑静默响应文件并运行脚本而无需响应提示来配置代理程序，请完成以下步骤：

1. 在文本编辑器中，打开位于以下路径的静默响应文件：*install_dir*/samples/postgresql_silent_config.txt，其中，*install_dir* 是 PostgreSQL 代理程序的安装目录。缺省安装目录为 /opt/ibm/apm/agent。
2. 要编辑静默配置文件，请完成以下步骤：
   a. 对于 **IP 地址**参数，指定要远程监视的 PostgreSQL 服务器的 IP 地址。如果在要监视的服务器上安装代理程序，那么保留缺省值。

      注:

      对于远程监视，**目前已用 CPU (%)** 和**已用物理内存 (MB)** 的数据在仪表板上将不可用。这些窗口小部件将显示**不适用**。
   b. 对于 **JDBC database name** 参数，指定数据库名称以更改缺省数据库名称 postgres。
   c. 对于 **JDBC user name** 参数，指定用户名称以更改缺省名称 postgres。
   d. 对于 **JDBC password** 参数，输入 JDBC 用户密码。
   e. 对于 **JDBC port number** 参数，指定端口号以更改缺省端口号 5432。
   f. 对于 **JDBC JAR file** 参数，如果缺省路径不正确，针对 Java JAR 文件指定 PostgreSQL 连接器的路径。Java JAR 文件的缺省路径为：

      /opt/PostgreSQL/lib/postgresql-9.3-1100.jdbc4.jar

      要点: JDBC JAR 文件的版本必须与监视的 PostgreSQL 数据库的版本兼容。
   g. 对于 **Java trace level** 参数，根据 IBM 支持指示信息指定跟踪级别。缺省跟踪级别为 Error。
3. 保存并关闭静默响应文件，然后运行以下命令：

   ```
   install_dir/bin/postgresql-agent.sh config
   instance_name
   install_dir/samples/postgresql_silent_config.txt
   ```

   其中，*instance_name* 是要对实例指定的名称。
4. 要启动代理程序，请输入以下命令：

   ```
   install_dir/bin/postgresql-agent.sh
   start instance_name
   ```

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

配置 Python 监视

内部应用程序和 IBM Cloud Python 应用程序均可监视。根据应用程序类型完成相应的配置步骤。

关于此任务

配置 Python 数据收集器以监视您的内部应用程序和 IBM Cloud Python 应用程序。

过程

· 配置数据收集器以监视 IBM Cloud 应用程序。

  a) 配置 Python 数据收集器以收集并发送 IBM Cloud 应用程序的数据。有关指示信息，请参阅第 572 页的『针对 IBM Cloud 应用程序配置 Python 数据收集器』。

b) 可选： 定制 Python 数据收集器的监视功能。有关更多信息，请参阅第 573 页的『针对 IBM Cloud 应用程序定制 Python 数据收集器』。

- 配置数据收集器以监视内部应用程序。
  a) 配置数据收集器，以收集数据并将其发送到 Cloud APM 服务器。有关指示信息，请参阅第 576 页的『为本地应用程序配置 Python 数据收集器』。
  b) 可选： 定制 Python 数据收集器的监视功能。有关更多信息，请参阅第 577 页的『为本地应用程序定制 Python 数据收集器』。

针对 IBM Cloud 应用程序配置 Python 数据收集器

要收集 IBM Cloud 上的 Python 应用程序的相关信息，必须配置 Python 数据收集器。

开始之前

1. 请确保所要监视的 Python 应用程序具有唯一名称。Python 数据收集器会将两个具有相同名称的不同应用程序作为一个应用程序进行处理，这可能会导致 Cloud APM 控制台中出现数据显示问题。
2. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

关于此任务

要配置此数据收集器，请先部署 pypi 软件包服务器，然后将此数据收集器安装到 Python Django 应用程序。

过程

1. 从数据收集器软件包中解压缩文件。已抽取的目录中包含 `python_datacollector_8.1.4.0.tgz` 软件包。
2. 解压缩 `python_datacollector_8.1.4.0.tgz` 软件包，例如，通过运行以下命令进行解压缩：

```
tar -zxf python_datacollector_8.1.4.0.tgz
```

3. 在解压缩的目录中查找软件包服务器的 `manifest.yml` 文件，并按以下示例所示，在该文件中定义域、主机和名称：

```
domain: mybluemix.net
name: pythondc
host: pythondc
```

切记: *host* 值与 *name* 值必须相同，而且必须是唯一的。

4. 从 `python_dc` 目录中，通过运行以下命令，将 `pythondc` 应用程序推送到 IBM Cloud：

```
cf push
```

5. 在 Python 应用程序的 `requirements.txt` 文件中，添加下列各行：

```
cryptography==1.9.0
--extra-index-url https://<your_host_name_and_domain>/python-dc-repos/simple/
ibm_python_dc
```

6. 在 Python 应用程序的 `settings.py` 文件中，将 `ibm_python_dc.kpg_plugin.ResourceMiddleware` 添加到 `MIDDLEWARE_CLASSES` 小节的开头，例如：

```
MIDDLEWARE_CLASSES = (
    "ibm_python_dc.kpg_plugin.ResourceMiddleware",
    "mezzanine.core.middleware.UpdateCacheMiddleware",
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
```

7. 从 Python 应用程序 `manifest.yml` 文件所在的目录中，运行以下命令：

```
cf push
```

提示: 要获取样本 `manifest.yml` 文件，请参阅第 160 页的『样本 manifest.yml 文件』。

结果
此数据收集器已配置完成，并已连接到 Cloud APM 服务器。

下一步做什么

您可以验证 IBM Cloud 应用程序的监视数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

针对 IBM Cloud 应用程序定制 Python 数据收集器
您可以在 IBM Cloud 用户界面 (UI) 中添加环境变量，以定制针对 IBM Cloud 应用程序进行的监视。请根据您的需要，使用以下信息来添加变量。

Python 数据收集器的用户定义环境变量

您可使用下表中的信息来定制针对 IBM Cloud 的 Python 监视。

表 *192.* 针对 *IBM Cloud* 的 *Python* 监视所支持的用户定义环境变量

变量名称	重要性	值	描述
APM_BM_GATEWAY_URL	可选	· https://*<server ip or hostname>*:443 · http://*<server ip or hostname>*:80	目标内部部署服务器网关 URL。
APM_KEYFILE_PSWD	可选	密钥文件的已加密密码	与密钥文件配对的已加密密钥文件密码。如果您是 Linux 用户，那么可以使用 `echo -n <keyfile password> | base64` 命令对密码进行加密。 **注:** 只有在已将网关配置为使用 HTTPS 的情况下，才应设置此变量。
APM_KEYFILE_URL	可选	http://*<hosted http server>*:*<port>*/keyfile.p12	用来下载密钥文件的 URL。 **注:** 只有在已将网关配置为使用 HTTPS 的情况下，才应设置此变量。
KPG_ENABLE_DEEPDIVE	可选	· False · True	启用或禁用诊断数据收集。 · True：缺省值。将此变量设置为 `True` 表示收集诊断数据。 · False：将此变量设置为 `False` 表示不收集诊断数据。 未设置此变量时，将会收集诊断数据。

表 *192.* 针对 *IBM Cloud* 的 *Python* 监视所支持的用户定义环境变量 *(续)*

变量名称	重要性	值	描述
KPG_DD_CONFIG_FILE	可选	诊断监视配置文件的文件名。	诊断监视配置文件的文件名。缺省文件名为 `kpg_dd_config.xml`。 **注:** 定制此文件中的设置之后，必须将此文件放在应用程序根目录中。 如果未设置此变量，那么将使用数据收集器软件包中的缺省配置文件 `kpg_dd_config.xml`。
KPG_DD_APP_PATH	可选	Python 应用程序的路径。	要由数据收集器收集相应诊断数据的 Python 应用程序或模块的路径。请使用分号 `;` 来分隔所要监视的不同 Python 应用程序和模块的路径。 如果未设置此变量，那么数据收集器将收集应用程序所使用的请求及模块的数据。不会收集 Python 库中的请求数据。
KPG_DD_SECURITY_FILTER	可选	· True · False	· True：缺省值。如果将此变量设置为 `True`，那么值（例如密码）在 SQL 语句中将会隐藏，并且参数不会显示在组窗口小部件**请求上下文**中。 · False：如果将此变量设置为 `False`，那么值在 SQL 语句中不会隐藏，并且参数将显示在组窗口小部件**请求上下文**中。 未设置此变量时，值（例如密码）在 SQL 语句中将会隐藏，并且参数不会显示在组窗口小部件**请求上下文**中。
KPG_GC_STATS	可选	True	启用 Python 垃圾回收的所有统计功能。将此值设置为 `True` 相当于运行以下命令： `gc.set_debug(gc.DEBUG_STATS | gc.DEBUG_COLLECTABLE | gc.DEBUG_UNCOLLECTABLE | gc.DEBUG_INSTANCES | gc.DEBUG_OBJECTS )` 要禁用 KPG_GC_STATS，请删除此环境变量。请勿将其设置为 `False`。 **注:** 在正式的生产环境中，切勿设置 `KPG_SAVE_ALL=True`。此设置仅适用于调试方式。请确保分配足够的内存给应用程序。

表 *192.* 针对 *IBM Cloud* 的 *Python* 监视所支持的用户定义环境变量 *(续)*

变量名称	重要性	值	描述
KPG_LOG_LEVEL	可选	· DEBUG · ERROR · INFO	· DEBUG：在日志中仅显示有用的调试信息，例如，收集到的数据、发送到服务器的数据以及服务器响应。 · ERROR：在日志中仅显示有关异常和意外情况的信息。 · INFO：在日志中显示关于数据收集器的摘要信息，以使用户了解其正在执行的操作。
KPG_LOG_TOCONSOLE	可选	· Y · True · 任何其他并非 `False` 的值	日志将输出到控制台，您可通过运行 **`cf logs <appname>`** 命令来查看日志。
KPG_SAVE_ALL	可选	True	所有取消引用的对象都会保存到 `gc.garbage` 中，您必须每分钟清除 `gc.garbage` 一次（数据收集器会自动将其清除）。将此值设置为 `True` 相当于运行以下命令： `gc.set_debug(gc.SAVE_ALL)` 要禁用 KPG_SAVE_ALL，请删除此环境变量。请勿将其设置为 `False`。 **注:** 在正式的生产环境中，切勿设置 KPG_SAVE_ALL=True。此设置仅适用于调试方式。请确保分配足够的内存给应用程序。

取消为 IBM Cloud 应用程序配置 Python 数据收集器
如果不需要监视 Python 环境，或者如果要升级 Python 数据收集器，首先必须取消配置 Python 数据收集器的先前设置。

过程

1. 转至 Python 应用程序的主目录。
2. 从应用程序的 `requirements.txt` 移除以下行：

```
--extra-index-url https://<your_host_name_and_domain>/python_dc/static/python-dc-repos/
simple/
ibm-python-dc
```

3. 在 `settings.py` 文件中，从 `MIDDLEWARE_CLASSES` 节移除以下行：

```
ibm_python_dc.kpg_plugin.ResourceMiddleware
```

4. 运行以下命令来重新推送应用程序，以使更改生效：

```
cf push
```

结果
您已成功取消配置 Python 数据收集器。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

为本地应用程序配置 Python 数据收集器

要收集本地环境中运行的 Python 应用程序的相关信息，必须配置 Python 数据收集器。

开始之前

1. 请确保所要监视的 Python 应用程序具有唯一名称。Python 数据收集器会将两个具有相同名称的不同应用程序作为一个应用程序进行处理，这可能会导致 Cloud APM 控制台中出现数据显示问题。
2. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

关于此任务

数据收集器软件包是预先配置的软件包，其中包含预先配置的 `global.environment` 文件，以及复制到 `etc` 文件夹的 `keyfile.p12`。因此，数据收集器将自动连接到 Cloud APM 服务器。

以下过程使用缺省设置在 Python 应用程序中配置数据收集器。要定制数据收集器配置，请在数据收集器配置文件中使用环境变量。有关更多信息，请参阅第 577 页的『为本地应用程序定制 Python 数据收集器』。

过程

1. 从数据收集器软件包抽取文件。已抽取的目录中包含 `python_datacollector_8.1.4.0.tgz` 软件包。
2. 从数据收集器软件包中抽取文件，例如，通过运行以下命令来抽取：

   ```
   tar -zxf python_datacollector_8.1.4.0.tgz
   ```

3. 从 `python_dc` 目录，运行以下命令：

   ```
   python server.py
   ```

4. 运行以下命令：

   ```
   pip install ibm_python_dc --extra-index-url http://host name or ip:8000/
   python-dc-repos/simple/ --trusted-host host name or ip
   ```

   其中，*host name or ip* 是运行 Python 数据收集器存储库的主机的名称或 IP 地址。

   要点: 对于此命令中的 URL 和可信主机，请使用名称或 IP 地址来指定主机。例如，如果使用 IP 地址来指定主机，而且该 IP 地址为 `9.42.36.180`，那么命令如下所示：

   ```
   pip install ibm_python_dc --extra-index-url http://9.42.36.180:8000/
   python-dc-repos/simple/ --trusted-host 9.42.36.180
   ```

5. 在 Python 应用程序的 `settings.py` 文件中，按以下示例的格式，将 `ibm_python_dc.kpg_plugin.ResourceMiddleware` 添加到 `MIDDLEWARE_CLASSES` 小节：

   ```
   MIDDLEWARE_CLASSES = (
       "ibm_python_dc.kpg_plugin.ResourceMiddleware",
       "mezzanine.core.middleware.UpdateCacheMiddleware",
       'django.contrib.sessions.middleware.SessionMiddleware',
       'django.middleware.common.CommonMiddleware',
   ```

结果

数据收集器已使用缺省设置进行配置，并连接到 Cloud APM 服务器。

下一步做什么
现在，您可登录 Cloud APM 服务器以查看监视数据。

切记: 将 Python 应用程序添加到 Cloud APM 控制台之后，可在名为 Python 运行时应用程序的组件中查看其监视数据。

有关如何启动 Cloud APM 服务器的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

为本地应用程序定制 Python 数据收集器
通过修改数据收集器软件包中的文件，可设置环境变量，以定制针对 Python 应用程序进行的监视。

有两个文件可用来定制数据收集器设置，它们分别是 `global.environment` 和 `config.properties`。更改这些文件中的设置之后，请重新启动 Python 应用程序，以使更改生效。

通过修改 `global.environment` 文件，可定制数据收集器与 Cloud APM 服务器之间的连接。如果要使用另一个 Cloud APM 服务器来代替缺省服务器，或者密钥文件或其密码已更改，请修改 Cloud APM 服务器，以将数据收集器重新连接至 Cloud APM 服务器。

通过修改 `config.properties` 文件，您可根据需要来定制数据收集器行为，例如启用或禁用方法跟踪。

`global.environment` 配置文件

第 577 页的表 193 显示可以在 `global.environment` 配置文件中设置的环境变量以及相关描述。您可在 Python 数据收集器安装位置中的 `etc` 文件夹（例如，`/root/.pyenv/versions/3.5.2/lib/python3.5/site-packages/ibm_python_dc/etc` 目录）中找到 `global.environment` 文件。

表 *193. global.environment* 文件中支持的环境变量

变量名称	重要性	值	描述
APM_BM_GATEWAY_URL	可选	· https://*<server ip or hostname>*:443 · http://*<server ip or hostname>*:80	目标内部部署服务器网关 URL。
APM_KEYFILE_PSWD	可选	密钥文件的密码	与密钥文件配对的密钥文件密码。 **注:** 只有在已将网关配置为使用 HTTPS 的情况下，才应设置此变量。
APM_KEYFILE_URL	可选	http://*<hosted http server>*:*<port>*/keyfile.p12	用来下载密钥文件的 URL。 **注:** 只有在已将网关配置为使用 HTTPS 的情况下，才应设置此变量。

`config.properties` 文件

第 578 页的表 194 显示可以在 `config.properties` 配置文件中设置的环境变量以及相关描述。您可在 Python 数据收集器的安装目录（例如，`/root/.pyenv/versions/3.5.2/lib/python3.5/site-packages/ibm_python_dc` 目录）中找到 `config.properties` 文件。

表 *194. config.properties* 文件中支持的环境变量

变量名称	重要性	值	描述
KPG_ENABLE_DEEPDIVE	可选	· False · True	· False：缺省值。将此变量设置为 False 表示不收集诊断数据。 · True：将此变量设置为 True 表示收集诊断数据。 缺省级别为 True。 如果未设置此变量，那么不会收集诊断数据。
KPG_DD_CONFIG_FILE	可选	诊断监视配置文件的文件名。	诊断监视配置文件的文件名。缺省文件名为 kpg_dd_config.xml。 **注:** 定制此文件中的设置之后，必须将此文件放在应用程序根目录中。 如果未设置此变量，那么将使用数据收集器软件包中的缺省配置文件 kpg_dd_config.xml。
KPG_DD_APP_PATH	可选	Python 应用程序的路径。	要由数据收集器收集相应诊断数据的 Python 应用程序或模块的路径。请使用分号 ; 来分隔所要监视的不同 Python 应用程序和模块的路径。 如果未设置此变量，那么数据收集器将收集应用程序所使用的请求及模块的诊断数据。不会收集 Python 库中的请求数据。
KPG_DD_SECURITY_FILTER	可选	· True · False	· True：缺省值。如果将此变量设置为 True，那么将隐藏 SQL 语句中的值（例如密码），并且参数不会显示在组窗口小部件**请求上下文**中。 · False：如果将此变量设置为 False，那么不会隐藏 SQL 语句中的值，并且参数将显示在组窗口小部件**请求上下文**中。 如果未设置此变量，那么将隐藏 SQL 语句中的值（例如密码），并且参数不会显示在组窗口小部件**请求上下文**中。

表 194. *config.properties* 文件中支持的环境变量 *(续)*

变量名称	重要性	值	描述
KPG_GC_STATS	可选	True	启用 Python 垃圾回收的所有统计功能。将此值设置为 True 相当于运行以下命令： `gc.set_debug(gc.DEBUG_STATS | gc.DEBUG_COLLECTABLE | gc.DEBUG_UNCOLLECTABLE | gc.DEBUG_INSTANCES | gc.DEBUG_OBJECTS )` 要禁用 KPG_GC_STATS，请删除此环境变量。请勿将其设置为 False。 缺省值为 True。 **注:** 在正式的生产环境中，切勿设置 KPG_GC_STATS=True。此设置仅适用于调试方式。请确保向应用程序分配足够的内存。
KPG_LOG_LEVEL	可选	· DEBUG · ERROR · INFO	· DEBUG：在日志中仅输出有用的调试信息，例如，收集的数据、发送到服务器的数据以及服务器响应。 · ERROR：在日志中仅输出关于异常和非常意外的情况的信息。 · INFO：在日志中输出关于数据收集器的摘要信息，以使用户了解其正在执行的操作。 缺省值为 ERROR。
KPG_LOG_TOCONSOLE	可选	· Y · True · 任何其他并非 False 的值	日志将输出到控制台，您可通过运行 **cf logs <appname>** 命令来查看日志。 缺省值为 True。
KPG_SAVE_ALL	可选	True	所有取消引用的对象都将保存到 gc.garbage 中，您需要每分钟清除 gc.garbage 一次（数据收集器会自动将其清除）。将此值设置为 True 相当于运行以下命令： `gc.set_debug(gc.SAVE_ALL)` 要禁用 KPG_SAVE_ALL，请删除此环境变量。请勿将其设置为 False。 缺省值为 True。 **注:** 在正式的生产环境中，切勿设置 KPG_SAVE_ALL=True。此设置仅适用于调试方式。请确保向应用程序分配足够的内存。

表 194. config.properties 文件中支持的环境变量 (续)			
变量名称	**重要性**	**值**	**描述**
APM_GW_PROXY_CONNECTION	可选	http://<*server ip or hostname*>:port	这是 Python 数据收集器用来发送监视数据的 HTTP 或 HTTPS 代理。

取消配置内部部署应用程序的 Python 数据收集器
如果不需要监视 Python 环境，或者如果要升级 Python 数据收集器，首先必须取消配置 Python 数据收集器的先前设置。

过程

1. 转至 Python 应用程序的主目录。
2. 从应用程序的 `requirements.txt` 移除以下行：

```
--extra-index-url https://<your_host_name_and_domain>/python_dc/static/python-dc-repos/
simple/
ibm-python-dc
```

3. 在 `settings.py` 文件中，从 `MIDDLEWARE_CLASSES` 节移除以下行：

```
ibm_python_dc.kpg_plugin.ResourceMiddleware
```

4. 运行 `pip uninstall ibm_python_dc` 命令以从 Python 运行时卸载 Python 数据收集器。

结果
您已成功取消配置 Python 数据收集器。

下一步做什么
在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

配置 RabbitMQ 监视

Monitoring Agent for RabbitMQ 用于监视 RabbitMQ 集群资源（例如，集群的节点、队列和通道）的运行状况和性能。您必须配置 RabbitMQ 代理程序，以便代理程序可收集 RabbitMQ 数据。

开始之前

- 查看必备硬件和软件。
- 确保连接至节点的 RabbitMQ 用户具有读许可权，并对此用户启用监视、管理员或管理标记。
- 确保在集群的所有节点上启用 RabbitMQ 管理插件，这样集群的一个节点发生故障时，RabbitMQ 代理程序可连接至集群中的对等节点。

查看必备硬件和软件。有关最新系统需求信息，请参阅 RabbitMQ 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

RabbitMQ 代理程序是多实例代理程序。必须创建第一个实例，然后手动启动代理程序。

· 要在 Windows 系统上配置此代理程序，您可以使用 IBM Cloud Application Performance Management 窗口或静默响应文件。
· 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for RabbitMQ**，然后单击**配置代理程序**。

   切记: 首次配置代理程序之后，**配置代理进程**选项将处于禁用状态。要再次配置代理程序，单击**重新配置**。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。
4. 在“**Monitoring Agent for RabbitMQ**”窗口中，指定配置参数的值，然后单击**下一步**。

   有关配置参数的信息，请参阅以下主题：第 582 页的『代理程序的配置参数』
5. 右键单击 **Monitoring Agent for RabbitMQ** 实例，然后单击**启动**。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

过程

1. 在命令行上，输入以下命令：*install_dir*/bin/rabbitmq.sh config *instance_name*，其中 *instance_name* 是要对实例指定的名称：
2. 当提示您提供以下参数的值时，请按 Enter 键以接受缺省值或指定一个值然后按 Enter 键。
   · IP 地址
   · 用户名
   · 密码
   · 端口号
   · Java 主目录
   · Java 跟踪级别

   有关配置参数的信息，请参阅以下主题：第 582 页的『代理程序的配置参数』
3. 运行以下命令以启动代理程序：

```
install_dir/bin/rabbitmq.sh start instance_name
```

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可使用静默响应文件来配置 RabbitMQ 代理程序。在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

1. 打开以下路径中的静默响应文件：
   install_dir\samples\rabbitmq_silent_config.txt
2. 在 rabbitmq_silent_config.txt 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

   有关配置参数的信息，请参阅以下主题：第 582 页的『代理程序的配置参数』
3. 保存响应文件，然后运行以下命令：

   Linux AIX *install_dir*/bin/rabbitmq-agent.sh config *install_dir*/samples/rabbitmq_silent_config.txt

   Windows *install_dir*/bin/rabbitmq-agent.bat config *install_dir*/samples/rabbitmq_silent_config.txt
4. 启动代理程序：

   Linux AIX 运行以下命令：*install_dir*\bin\rabbitmq-agent.sh start

   Windows 右键单击 **Monitoring Agent for RabbitMQ**，然后单击**开始**。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

代理程序的配置参数

在配置 RabbitMQ 代理程序时，可以更改参数的缺省值，例如实例名称和 SSL 验证证书。

下表包含 RabbitMQ 代理程序的配置参数的详细描述。

表 *195. RabbitMQ* 代理程序的配置参数的名称和描述

参数名称	描述	必填字段
IP 地址	RabbitMQ 应用程序所安装在的节点的 IP 地址。	是
用户名	RabbitMQ 用户的用户名。	是
密码	用于连接至 RabbitMQ 管理用户界面的密码。	是
确认密码	在**密码**字段中输入的同一密码。	是
端口号	用于启用 RabbitMQ 管理插件的端口号。使用缺省端口号 15672，或指定另一个端口号。	否
Java 主目录	Java 插件的安装路径。使用缺省路径 C:\Program Files\IBM\Java50，或 java 插件的安装目录路径。	否

表 195. RabbitMQ 代理程序的配置参数的名称和描述 (续)		
参数名称	**描述**	**必填字段**
Java 跟踪级别	Java 提供程序的跟踪级别。有效跟踪级别值如下所示: · OFF · ERROR · WARN · INFO · DEBUG_MAX · ALL	否

配置响应时间监视

Response Time Monitoring 代理程序用于监视 HTTP 服务器上的 HTTP 和 HTTPS 事务。另外，还会监视基于浏览器的实际用户事务（浏览器计时）。

Response Time Monitoring 代理程序可用来查看下列级别的监视信息:

HTTP 和 HTTPS 事务监视

安装 Response Time Monitoring 代理程序后，HTTP 事务监视功能将自动可用。

根据所监视的 HTTP 服务器类型，HTTPS 事务监视功能可能会自动可用，也可能需要进行手动配置。有关更多信息，请参阅第 584 页的『Response Time Monitoring 组件』。

Response Time Monitoring 还监视与用户计数、会话计数和设备相关的数据。

数据将以用户的本地时间显示在“最终用户事务”仪表板中，而且还用于“请求和响应时间”窗口小部件。

实际最终用户事务监视（浏览器计时）

根据所监视的 HTTP 服务器类型，基于浏览器的计时可能会自动可用，也可能需要进行配置。有关更多信息，请参阅第 584 页的『Response Time Monitoring 组件』。

基于浏览器的计时是通过“JavaScript 注入”实现。

利用“JavaScript 注入”，您可在“最终用户事务”仪表板中查看更多窗口小部件和详细信息。“JavaScript 注入”确保从浏览器收集实际的最终用户响应时间。它会监视 HTTP 服务器所提供的 Web 页面的 HTTP 页面及嵌入对象的性能。另外，还会提供下列实际最终用户事务详细信息:

- “事务请求和响应时间”窗口小部件中的“客户机时间总计”
- “事务 - 排名前 10 位”窗口小部件中的“客户机事务的响应时间”
- 呈现时间细分

有关如何配置“JavaScript 注入”的信息，请参阅第 586 页的『JavaScript 注入』。

查看事务仪表板

在应用程序性能仪表板中查看事务数据。

提供有关事务上下文详细信息的应用程序性能仪表板中提供了若干窗口小部件。

良好请求响应时间少于 10 秒。慢请求响应时间多于 10 秒。用于确定良好响应时间与慢响应时间的 10 秒值不可配置。提供以下窗口小部件:

- 按用户排序最差项 - 排名前 5 的组窗口小部件
- 按设备排序最差项 - 排名前 5 的组窗口小部件
- “请求和响应时间”组窗口小部件
- 事务 - 排名前 10 的组窗口小部件

- “事务请求和响应时间”组窗口小部件
- “运行位置”组窗口小部件
- “子事务”组窗口小部件
- “事务实例”组窗口小部件
- “按位置划分用户”组窗口小部件
- “选定位置的用户”组窗口小部件
- 选定位置的用户会话 - 排名前 10 的组窗口小部件
- “用户请求和响应时间”组窗口小部件
- 用户会话 - 排名前 10 的组窗口小部件
- “设备请求和响应时间”组窗口小部件
- “会话”组窗口小部件
- “会话请求”组窗口小部件
- “会话实例”组窗口小部件
- “事务实例”组窗口小部件
- “中间件事务摘要”仪表板
- “中间件事务详细信息”仪表板
- 事务监视的事件阈值
- 交互汇总数据
- 事务汇总数据
- WRT 事务状态

Response Time Monitoring 组件

Response Time Monitoring 代理程序的基本功能包括：

- HTTP 事务监视
- HTTPS 事务监视
- 基于浏览器的计时（使用“JavaScript 注入”）监视

要查看此功能的更多详细描述，请参阅 第 583 页的『配置响应时间监视』。

根据所监视的 HTTP 服务器类型，Response Time Monitoring 代理程序的基本功能将通过下列其中一个组件来提供：

IBM HTTP Server 响应时间模块
IBM HTTP Server 响应时间模块只能监视以下类型的 HTTP 内容：text/html、application/xml 或 application/json。（无 Javascript 注入）

IBM HTTP Server 响应时间模块当前无法监视 JavaScript 工具压缩请求。

IBM HTTP Server 响应时间模块 Javascript 注入当前仅监视 text/html 类型的 HTTP 内容。

Packet Analyzer
Packet Analyzer 只能监视 text/html 类型的内容。Packet Analyzer 可监视 gzip 压缩请求。

如果要在 AIX 或 Linux 上监视 IBM HTTP Server 或 Apache HTTP Server，请使用 IBM HTTP Server 响应时间模块。可以使用 Packet Analyzer，但不建议这样做。IBM HTTP Server 响应时间模块在 Windows 上不受支持。在 Windows 环境中请使用 Packet Analyzer。

如果要监视任何其他 HTTP Server，请使用 Packet Analyzer。Packet Analyzer 在 Window、Linux 和 AIX 上都受支持。

规划安装

根据操作系统和 HTTP 服务器类型来规划 Response Time Monitoring 代理程序安装。

Response Time Monitoring 的基本功能可使用下列其中一个组件来提供：

- Packet Analyzer
- IBM HTTP Server 响应时间模块

您可根据下列事项来确定所要使用的组件：

- 正在其中安装 Response Time Monitoring 代理程序的 HTTP 服务器类型。
- HTTP 服务器安装在哪个操作系统上。

下列是随 Packet Analyzer 一起安装 Response Time Monitoring 代理程序时的注意事项：

- Packet Analyzer 在所有操作系统（Windows、Linux 和 AIX）上都受支持。
- 在所有操作系统上，Packet Analyzer 都在端口 80 上监视 HTTP 事务。
- HTTPS 事务监视不是自动的，必须进行手动配置。Response Time Monitoring 代理程序需要访问 SSL 证书，以便可以对来自 HTTP 服务器的 SSL 流量进行解密。有关更多信息，请参阅第 602 页的『监视 HTTPS 事务』。
- Packet Analyzer 在所有 HTTP 服务器上都受支持，但仅建议用于 Sun Java System Web Server 和 Microsoft Internet Information Services。
- AIX Linux Windows 要安装 Response Time Monitoring 代理程序以便与 IBM HTTP Server 或 Apache HTTP Server 上的 Packet Analyzer 配合工作，必须停止 HTTP Server。安装 Response Time Monitoring 代理程序时，如果 HTTP 服务器已停止，那么将自动启用 Packet Analyzer。
- AIX Linux Windows 虽然可以为 IBM HTTP Server 或 Apache HTTP Server 配置 Packet Analyzer，但建议不要这样做；建议使用 IBM HTTP Server 响应时间模块。
- Windows 必须先安装 WinPcap 4.1.3，然后才能安装 Response Time Monitoring 代理程序。
- Windows AIX Linux 如果将 Response Time Monitoring 代理程序安装在 Sun Java System Web Server 或 Microsoft Internet Information Services 上，那么将自动配置 Packet Analyzer。

下列是随 IBM HTTP Server 响应时间模块 一起安装 Response Time Monitoring 代理程序时的注意事项：

- IBM HTTP Server 响应时间模块是 HTTP Server 代理程序的组件。HTTP Server 代理程序必须在 Response Time Monitoring 之前进行安装，或者同时进行安装。有关更多信息，请参阅第 592 页的『IBM HTTP Server 响应时间模块』。
- IBM HTTP Server 响应时间模块 在所有操作系统（Windows、Linux 和 AIX）上都受支持。IBM HTTP Server 响应时间模块 支持 IBM HTTP Server V7、V8 和 V9。
- 在同一台机器上，安装 Response Time Monitoring 代理程序和 HTTP Server 代理程序。
- 在 AIX、Linux 和 Windows 上，IBM HTTP Server 响应时间模块在所有端口上监视 HTTP 和 HTTPS 请求。
- 只有 IBM HTTP Server 或 Apache HTTP Server 才支持 IBM HTTP Server 响应时间模块。
- 这两个代理程序都会自动启动，但您必须重新启动 HTTP Server。

下表描述可自动配置 Response Time Monitoring 代理程序的不同组合。

表 *196.* 可自动配置 *Response Time Monitoring* 代理程序的场景

HTTP 服务器与操作系统的组合	**Packet Analyzer**	**IBM HTTP Server 响应时间模块**
AIX 或 Linux 上的 Sun Java System Web Server 或 Microsoft Internet Information Services	自动	不支持
Windows 上的 Sun Java System Web Server 或 Microsoft Internet Information Services	自动	不支持
AIX、Linux 或 Windows 上的 IBM HTTP Server 或 Apache HTTP Server	自动（如果 HTTP 服务器已停止）。	自动（如果 HTTP 服务器存在并已配置完成）。

规划配置

安装 Response Time Monitoring 代理程序时，将自动启用 HTTP 监视。根据您的环境，可能需要手动配置 HTTPS 和“JavaScript 注入”。

HTTP 监视

如果您按安装指南执行操作，那么将为 Packet Analyzer 和 IBM HTTP Server 响应时间模块自动配置 HTTP 事务监视。请参阅第 584 页的『规划安装』。

HTTPS 监视

如果您按安装指南执行操作，那么将为 IBM HTTP Server 响应时间模块自动配置 HTTPS 事务监视。请参阅第 584 页的『规划安装』。

您需要为 Packet Analyzer 手动配置 HTTPS 事务监视。有关更多信息，请参阅第 599 页的『Packet Analyzer 路线图』。

基于浏览器的计时（使用“JavaScript 注入”）

系统会为 IBM HTTP Server 响应时间模块自动配置基于浏览器的计时（使用“JavaScript 注入”）。

您需要为 Packet Analyzer 手动配置基于浏览器的计时（使用“JavaScript 注入”）。有关更多信息，请参阅第 599 页的『Packet Analyzer 路线图』。

下表描述针对每个组件配置基本功能的方式：

表 *197.* 基本功能配置

	Packet Analyzer，位于 Sun Java System Web Server 或 Microsoft Internet Information Services 上（Windows、Linux 或 AIX）	**IBM HTTP Server 响应时间模块，位于 IBM HTTP Server 或 Apache HTTP Server 上（Windows、Linux 或 AIX）**
HTTP 事务监视	自动启用。	自动启用。
HTTPS 事务监视	必须手动配置。	自动启用。
使用 JavaScript 检测的实际最终用户事务监视（浏览器计时）	必须手动配置。	自动启用。

JavaScript 注入

您可以定制 Response Time Monitoring 代理程序为在“最终用户事务”仪表板中显示而收集的数据。

要确保基于 Web 的应用程序有良好的用户体验，您必须监视实际用户所感受到的性能。这意味着在浏览器级别进行监视。

为了能够在浏览器级别进行监视，需要将 JavaScript 监视代码注入到所要监视的页面中。这样，此代码将收集特定浏览器计时的数据。

这是通过在所要监视的 Web 页面和对象中使用“JavaScript 注入”来完成。根据 Response Time Monitoring 代理程序所安装在的 HTTP 服务器类型，可通过两种方法来收集实际的最终用户事务响应时间信息。

- 如果您使用的是 IBM HTTP Server 或 Apache HTTP Server，请使用 IBM HTTP Server 响应时间模块。IBM HTTP Server 响应时间模块将自动执行“JavaScript 注入”。IBM HTTP Server 响应时间模块是 HTTP Server 代理程序的组件。它是作为 HTTP Server 代理程序的组成部分进行安装和配置。有关更多信息，请参阅第 592 页的『IBM HTTP Server 响应时间模块』。
- 如果您使用的是任何其他受支持的 HTTP 服务器，请使用 Packet Analyzer。对于 Packet Analyzer，您必须手动检测 Web 页面以收集浏览器计时。有关更多信息，请参阅第 600 页的『将 JavaScript 监视组件添加到应用程序』。

下表显示您为 Packet Analyzer 或 IBM HTTP Server 响应时间模块配置环境时“应用程序性能仪表板”中提供的功能：

	Packet Analyzer	IBM HTTP Server 响应时间模块
排名前 10 位的事务	✓	✓
服务器时间	✓	✓
呈现时间细分	—	✓
AJAX 子事务	✓	✓
子事务表中的资源计时数据	—	✓
事务实例（排名前 10 位）	✓	✓
事务实例拓扑	✓	✓
应用程序拓扑	✓	✓
自动检测“JavaScript 注入”	不适用	✓

在 Windows 上重新配置 Response Time Monitoring

使用 `rt-agent` 交互式配置命令或 IBM Cloud Application Performance Management 实用程序来配置或重新配置代理程序。

开始之前

如果启用 HTTPS 事务监视，请确保未在相同机器上安装 Monitoring Agent for HTTP Server。否则，Response Time Monitoring 配置不会为 Packet Analyzer 更改 HTTPS 设置。

关于此任务

在安装之后，Response Time Monitoring 代理程序会自动进行配置。请按安装指南执行操作：第 584 页的『规划安装』。您可能需要进行重新配置，例如，要监视另一个端口或监视 HTTPS 事务时，就需要进行重新配置。

安装目录被称为 *install_dir*。缺省安装目录为：`C:\IBM\APM\`

作为使用 `rt-agent` 交互式配置命令的替代方法，您可以在 IBM Cloud Application Performance Management 实用程序中配置代理程序。有关更多信息，请参阅第 155 页的『在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口』。

过程

要定制数据设置，请完成下列步骤：

1. 在安装 Response Time Monitoring 代理程序的计算机上，停止此代理程序：
   install_dir`\BIN\rt-agent.bat stop`
2. 使用静默配置以配置代理程序：

   ```
   install_dir\BIN\rt-agent.bat config install_dir\samples\rt_silent_config.txt
   ```

   如果要启用 HTTPS 事务监视，请在静默配置文件中取消注释以下行。用于配置 Response Time Monitoring 代理程序以监视 Windows 上的 HTTPS 的 `rt_silent_config.txt` 的样本如下所示：

   ```
   # Monitor HTTPS transactions
   KT5MONITORHTTPSAPP=YES

   # HTTPS keystore (e.g. - /tmp/keys.kdb)
   KT5KEYSTORE=C:\keys\key.kdb

   # HTTPS server certificate map (eg - certAlias,9.48.152.1,443;...)
   KT5SERVERMAP=certalias,9.48.152.1,443
   # Monitor network traffic for the NIC hosts this IP address
   #KT5MONITORIP=9.48.152.1
   ```

3. 重新启动 Response Time Monitoring 代理程序，以使更改生效：

install_dir\BIN\rt-agent.bat start

结果

新源中的数据显示在与 Response Time Monitoring 相关联的仪表板中。

在 AIX 和 Linux 上重新配置 Response Time Monitoring

使用 rt-agent 配置命令可配置或重新配置 Response Time Monitoring 代理程序。

关于此任务

在安装之后，Response Time Monitoring 代理程序会自动进行配置。请按安装指南执行操作：第 584 页的『规划安装』。您可能需要进行重新配置，例如，要监视另一个端口时，就需要进行重新配置。

安装目录表示为 *install_dir*。缺省安装目录为：/opt/ibm/apm/agent。

请使用用于安装代理程序的 root 用户来启动、停止和配置代理程序。

过程

要进行重新配置，请完成下列步骤：

1. 在安装了 Response Time Monitoring 代理程序的计算机上，停止该代理程序：

   ```
   install_dir/bin/rt-agent.sh stop
   ```

2. 使用交互式或静默配置方法：

   a) 交互式配置：

   ```
   install_dir/bin/rt-agent.sh config
   ```

   b) 静默配置：

   ```
   install_dir/bin/rt-agent.sh config  install_dir/samples/rt_silent_config.txt
   ```

   如果要启用 HTTPS 事务监视，请在静默配置文件中取消注释以下行。用于配置 Response Time Monitoring 代理程序以监视 AIX 和 Linux 上的 HTTPS 的 rt_silent_config.txt 的样本如下所示：

   ```
   # Monitor HTTPS transactions
   KT5MONITORHTTPSAPP=YES

   # HTTPS keystore (e.g. - /tmp/keys.kdb)
   KT5KEYSTORE=/tmp/keys.kdb

   # HTTPS server certificate map (eg - certAlias,9.48.152.1,443;...)
   KT5SERVERMAP=certalias,9.48.152.1,443
   # Monitor network traffic for the NIC hosts this IP address
   #KT5MONITORIP=9.48.152.1
   ```

3. 重新启动 Response Time Monitoring 代理程序，以使更改生效：
   install_dir/bin/rt-agent.sh start

结果

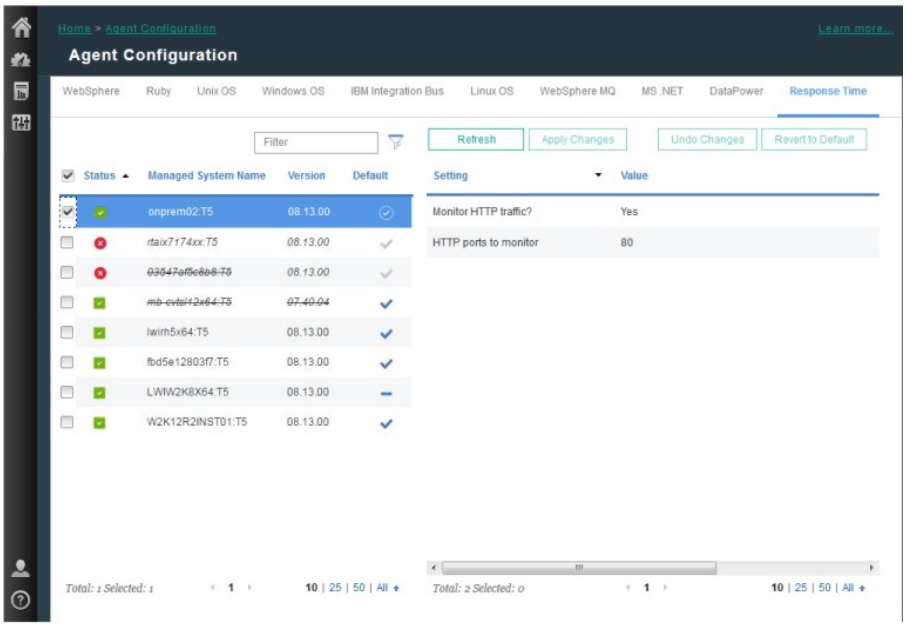
新源中的数据显示在与 Response Time Monitoring 相关联的仪表板中。

使用“代理程序配置”页面进行配置

您可使用 Cloud APM 控制台中的“**代理程序配置**”页面来查看已安装的代理程序。在适用的情况下，您可以禁用或启用 HTTP 事务监视，并设置 Response Time Monitoring 代理程序监视的端口。

代理程序配置

要访问“Response Time Monitoring **代理程序配置**”页面，请在 Cloud APM 控制台中，选择**系统配置** > **代理程序配置**，然后选择**响应时间**选项卡。



“**代理程序配置**”页面列出了环境中已安装 Response Time Monitoring 的系统。

对于安装了 Response Time Monitoring 代理程序的每个系统，“**代理程序配置**”页面显示了以下内容：

- 系统是联机（绿色背景的复选标记）还是脱机（红色背景的十字）。
- 安装的 Response Time Monitoring 代理程序版本。
- 如果中心配置无法确定代理程序的类型（即，将使用 Packet Analyzer 还是 IBM HTTP Server 响应时间模块来监视 HTTP 事务），那么仅代理程序带有删除线。通常，如果代理程序未通过 ASF 活动发送代理程序详细信息，那么无法确定其类型。
- 系统是使用缺省配置值还是设置了一些定制值。
- 在 Response Time Monitoring 代理程序使用 Packet Analyzer 监视 HTTP 事务时受监视的端口。

Setting	Value
Monitor HTTP traffic?	Yes
HTTP ports to monitor	80

- 是否将使用 IBM HTTP Server 响应时间模块和 HTTP Server 代理程序来监视 HTTP 事务。

Setting	Value
Is IBM HTTP Server Response Time module enabled?	Yes

提示: IBM HTTP Server 响应时间模块自动监视 HTTP 和 HTTPS 事务。无需进一步配置 Response Time Monitoring 代理程序。

选择代理程序以显示其配置设置。要查找特定代理程序，请在**过滤器**字段中输入安装该代理程序的系统的部分或完整名称。

在“**代理程序配置**”页面中执行的定制优先于任何其他定制并且优先于缺省值。

如果您针对所更改的设置改变主意，请单击**撤销更改**以还原为最近保存的设置，或者单击**还原为缺省值**以还原为缺省值。

新配置值会发送到 Central Configuration Services，然后将自动重新配置联机代理程序，而无需重新启动。如果代理程序脱机，它将在联机时下载新配置设置。刷新数据时，来自新端口的数据将显示在与 Response Time Monitoring 关联的仪表板中。

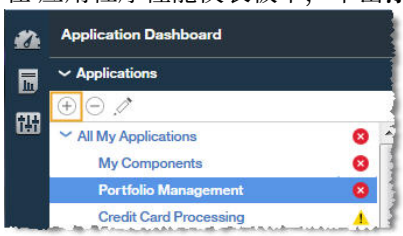
添加应用程序

安装 Response Time Monitoring 代理程序后，可能需要将您想监视的应用程序添加到 应用程序性能仪表板。

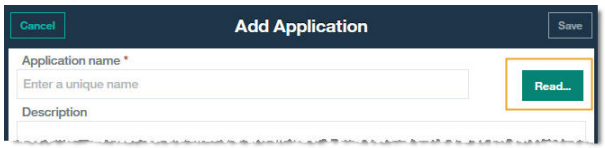
过程

要将应用程序添加到 应用程序性能仪表板，请执行以下操作：

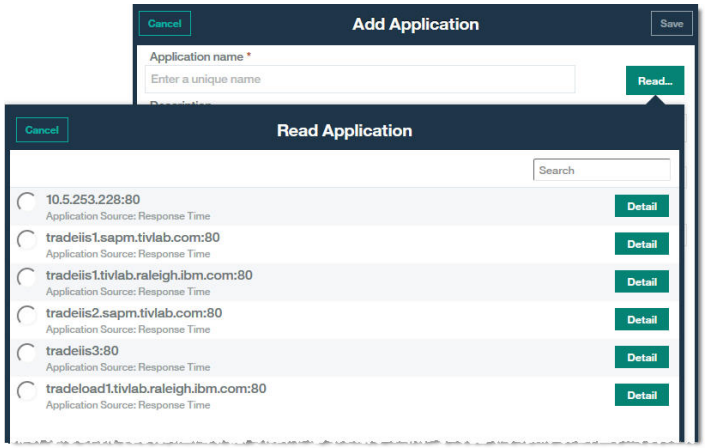
1. 在 应用程序性能仪表板中，单击**添加应用程序**。



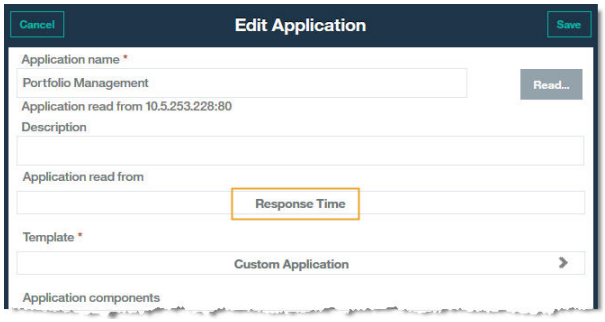
2. 选择**读取**以打开已发现应用程序的列表。



3. 选择要监视的应用程序。



Response Time 在**应用程序读取位置**字段中显示为源存储库，并且所有组件都在**应用程序组件**中列出。



4. 无需进一步配置即可在 应用程序性能仪表板 中显示 Response Time Monitoring 代理程序所监视的应用程序。单击“**添加应用程序**”窗口中的**保存**。

结果

Response Time Monitoring 代理程序检测到的应用程序将在 应用程序性能仪表板 的**我的所有应用程序**中列出。

配置 IBM HTTP Server 响应时间模块

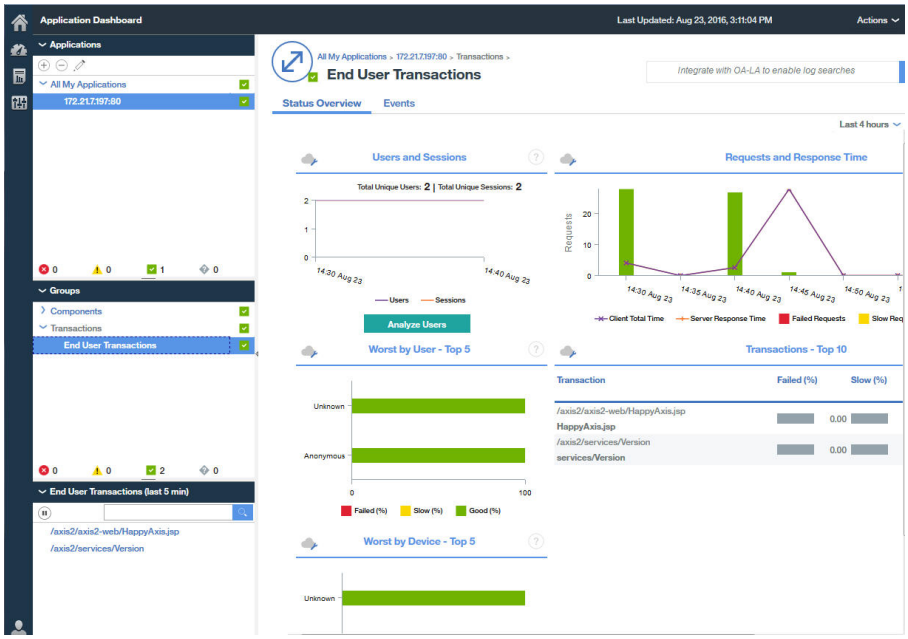
对于 IBM HTTP Server 和 Apache HTTP Server，使用 IBM HTTP Server 响应时间模块可查看 HTTP 页面的实际最终用户响应时间监视度量。

IBM HTTP Server 响应时间模块是作为 HTTP Server 代理程序的组成部分进行安装和配置。IBM HTTP Server 响应时间模块只能与 AIX、Linux 和 Windows 上的 IBM HTTP Server 和 Apache HTTP Server 配合工作。IBM HTTP Server 响应时间模块将在所有端口上监视 HTTP 和 HTTPS 请求。

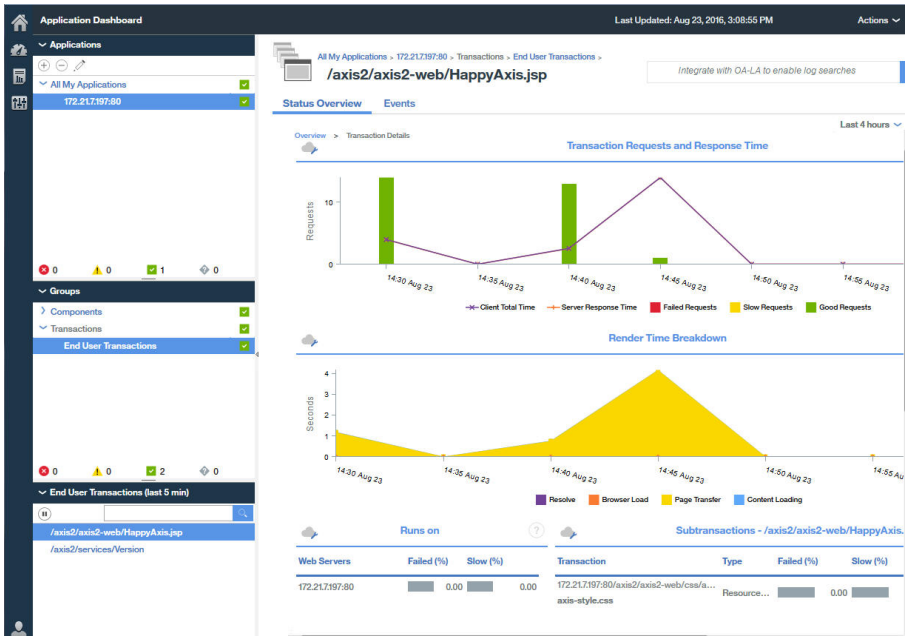
通过使用 JavaScript，IBM HTTP Server 响应时间模块可以将标头插入到 IBM HTTP Server 所提供的 Web 页面中，从而使 Response Time Monitoring 代理程序能够监视这些页面。将使用 Cookie 跟踪页面所装入的嵌入式对象。这样，IBM HTTP Server 或 Apache 所提供的 Web 页面中的事务信息将包括在“最终用户事务”仪表板中。

例如：

“最终用户事务”工作空间，在“事务 - 排名前 10 位”中显示从 IBM HTTP Server 响应时间模块收集的数据：



“最终用户事务”工作空间，在“**子事务**”表中显示从 IBM HTTP Server 响应时间模块收集的数据



IBM HTTP Server 响应时间模块

IBM HTTP Server 响应时间模块是 HTTP Server 代理程序的组成部分。但是，它与 Response Time Monitoring 代理程序配合工作，以监视受支持 HTTP 服务器上的应用程序事务。

当您安装 Response Time Monitoring 代理程序以便与 IBM HTTP Server 响应时间模块配合工作时，它将在所有端口上监视 HTTP 和 HTTPS 请求。

IBM HTTP Server 响应时间模块是 HTTP Server 代理程序的组成部分。HTTP Server 代理程序必须在 Response Time Monitoring 之前进行安装，或者同时进行安装。

HTTP Server 代理程序由两个插件组成：

1. khu_module - 这是 HTTP Server 代理程序。此插件负责所有与 HTTP Server 代理程序相关联的仪表板。有关更多信息，请参阅 HTTP Server 代理程序参考。
2. wrt_module - 这是 IBM HTTP Server 响应时间模块。

这两个插件在 HTTP Server 代理程序配置文件中指示。对于 Apache HTTP Server，HTTP Server 代理程序配置文件如下所示：

```
khu.usr.local.apache24.conf.httpd.conf
```

对于 IBM HTTP Server，此文件如下所示：

```
khu.opt.IBM.HTTPServer.conf.httpd.conf
```

此文件的命名规则是：khu.(HTTP 服务器配置文件的完整路径，将 / 更改为 .).conf

```
LoadModule khu_module

LoadModule wrt_module
```

为了让 IBM HTTP Server 响应时间模块正常工作，HTTP 服务器配置文件必须包含一个引用 HTTP Server 代理程序配置文件的 include 语句。例如：

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.conf
```

这个 include 语句将同时启用这两个插件。有关更多信息，请参阅第 231 页的『配置 HTTP Server 监视』。

安装和配置 IBM HTTP Server 响应时间模块

在 Response Time Monitoring 代理程序端，配置将自动进行。IBM HTTP Server 响应时间模块必须作为 HTTP Server 代理程序的组成部分进行安装和配置。Response Time Monitoring 代理程序将自动检测 IBM HTTP Server 响应时间模块并将其启用。

关于此任务

过程

1. 安装 HTTP Server 代理程序，这将自动安装 IBM HTTP Server 响应时间模块。
2. 配置 HTTP Server 代理程序。这将启用 IBM HTTP Server 响应时间模块。有关更多信息，请参阅第 231 页的『配置 HTTP Server 监视』。
3. 以 `root` 或 `Administrator` 用户身份安装 Response Time Monitoring 代理程序 代理程序，具体取决于您的操作系统。 有关详细的指示信息，请参阅第 101 页的『第 6 章 安装代理程序』。
4. 重新启动 IBM HTTP Server。当 Response Time Monitoring 安装程序检测到 HTTP Server 代理程序时，Response Time Monitoring 代理程序将自动启用 IBM HTTP Server 响应时间模块。

手动启用 IBM HTTP Server 响应时间模块

您可以手动启用 IBM HTTP Server 响应时间模块以监视 HTTP 页面性能和 IBM HTTP Server 所提供 Web 页面的嵌入式对象的性能。

关于此任务

安装并配置 HTTP Server 代理程序时，将自动启用 IBM HTTP Server 响应时间模块。但是，您可能想手动启用 IBM HTTP Server 响应时间模块。

过程

要在 Linux、AIX 或 Windows 上手动启用 IBM HTTP Server 响应时间模块，请完成下列步骤：

1. Linux AIX

   要在 Linux 或 AIX 上手动启用 IBM HTTP Server 响应时间模块，请完成下列步骤：

   a) 停止 Response Time Monitoring 代理程序。

   运行

   ```
   $AGENT_HOME/bin/rt-agent.sh stop
   ```

   其中，*$AGENT_HOME* 可以是 `/opt/ibm/apm/agent`（在 Linux 系统上）或 `/opt/ibm/ccm/agent`（在 AIX 系统上）。

   b) 运行配置命令并使用 *$AGENT_HOME*`/samples/rt_silent_config_ihs.txt` 以将装入模块添加到 Web 服务器配置文件，并针对 IBM HTTP Server 响应时间模块设置配置参数。

   c) 重新启动 Response Time Monitoring 代理程序。

2. Windows

   要在 Windows 上手动启用 IBM HTTP Server 响应时间模块，请完成下列步骤：

   a) 停止 Response Time Monitoring 代理程序。

   运行

   ```
   AGENT_HOME/bin/rt-agent.bat stop
   ```

   其中，*AGENT_HOME* 可以是 `C:\IBM\APM`（在 Windows 系统上）。

   b) 运行配置命令并使用 *AGENT_HOME*`\samples\rt_silent_config_ihs.txt` 以将装入模块添加到 Web 服务器配置文件，并针对 IBM HTTP Server 响应时间模块设置配置参数。

   例如，

   ```
   AGENT_HOME\bin\rt_agent.bat config AGENT_HOME\samples\rt_silent_config_ihs.txt
   ```

c) 重新启动 Response Time Monitoring 代理程序。

手动禁用 IBM HTTP Server 响应时间模块
要禁用 IBM HTTP Server 响应时间模块，并再次使用 Packet Analyzer，请重新配置代理程序并关闭 IBM HTTP Server 响应时间模块 监视。

关于此任务

使用以下过程来禁用 IBM HTTP Server 响应时间模块。

过程

要在 Linux、AIX 或 Windows 上以交互方式重新配置代理程序，请完成下列步骤：

1. Linux AIX
   要在 Linux 和 AIX 上以交互方式重新配置代理程序，请完成下列步骤：
   a) 运行 *install_dir*/bin/rt-agent.sh config
   其中，*install_dir* 是 Linux 和 AIX 上的 /opt/ibm/apm/agent。
   b) 重新启动 Response Time Monitoring 代理程序。
   或者，手动设置参数：
   a) 在文本编辑器中打开 *install_dir*/config/*hostname*_t5.cfg。
   其中，*install_dir* 是 Linux 和 AIX 上的 /opt/ibm/apm/agent。
   b) 设置以下参数：

   ```
   KT5DISABLEANALYZER=NO
   KT5ENABLEWEBPLUGIN=NO
   ```

   c) 重新启动 Response Time Monitoring 代理程序。
2. Windows
   要在 Windows 上以交互方式重新配置代理程序，请完成下列步骤：
   a) 手动设置相关参数，在文本编辑器中打开 *install_dir*\TMAITM6_x64*hostname*_t5.cfg。
   其中，*install_dir* 是 C:\IBM\APM（在 Windows 上）。
   b) 设置以下参数：

   ```
   KT5DISABLEANALYZER=NO
   KT5ENABLEWEBPLUGIN=NO
   ```

   c) 重新启动 Response Time Monitoring 代理程序。

IBM HTTP Server 响应时间模块的高级配置
IBM HTTP Server 响应时间模块有多个高级配置选项。

IBM HTTP Server 响应时间模块会自动进行配置，但是您可执行多个高级配置任务，以微调性能和功能。

禁用资源计时监视
将针对 HTTP Server 代理程序安装的所有 IBM HTTP Server 响应时间模块实例启用资源计时监视。

关于此任务

如果要减少 IBM HTTP Server 响应时间模块所监视的资源数，或者禁用资源计时监视以降低监视特定 IBM HTTP Server 所需的处理负载，请完成以下步骤：

过程

要编辑生成的 HTTP Server 代理程序配置文件，请完成下列步骤：

1. 在 HTTP Server 配置文件 (httpd.conf) 的末尾，附加

```
WrtMaxPostResourcesSize
```

2. 设置以下某个值：
   - `WrtMaxPostResourcesSize -1`，用于监视所有资源
   - `WrtMaxPostResourcesSize 0`，用于关闭资源监视
   - `WrtMaxPostResourcesSize` *n*，用于监视特定数量的资源，缺省为 `10`。例如，设置 `WrtMaxPostResourcesSize 2` 表示最多将 2 项资源发布到服务器。
3. 重新启动 HTTP Server。

禁用 *ARM* 相关因子生成

缺省情况下，将启用 ARM 相关因子生成，这允许 IBM HTTP Server 响应时间模块链接到拓扑中的任何后端服务器。如果需要，可以禁用 ARM 相关因子生成。

关于此任务

限制: 如果禁用 ARM 相关因子生成，那么 IBM HTTP Server 响应时间模块无法链接到后端服务器，例如 WebSphere Application Server。仅在 IBM 软件支持人员建议时才禁用 ARM 相关因子生成。

过程

要禁用 ARM 相关因子生成，请完成下列步骤：

1. 在 HTTP Server 配置文件 (httpd.conf) 的末尾，附加

```
WrtDisableArmCorr
```

2. 重新启动 HTTP Server

禁用针对客户机时间的响应时间监视（*JavaScript* 检测）

在 IBM Application Performance Management 中，将针对在所有计算机上运行的所有 IBM HTTP Server 安装启用针对客户机时间的响应时间监视（JavaScript 检测）。

关于此任务

过程

要手动禁用 JavaScript 注入，请完成下列步骤：

1. 打开位于以下位置的 HTTP Server 配置文件：*HTTP_Server_root*`/conf/httpd.conf`
2. 浏览到为 HTTP Server 代理程序添加的行，在该行之后附加以下行：

```
WrtDisableJSI
```

   例如，

```
include /opt/ibm/apm/agent/tmp/khu/khu.opt.IBM.HTTPServer.conf.httpd.confWrtDisableJSI
```

3. 保存 `httpd.conf` 文件，并重新启动 HTTP Server。

绕过 *WRTCorrelator cookie*

针对 WebSphere Portal 上的 Response Time Monitoring 代理程序启用 JavaScript 注入时，当您使用 WebSphere Portal 时 cookie（例如，WRTCorrelator cookie）可能导致问题。为了防止出现这些问题，您可以将 WRTCorrelator cookie 设置为忽略。

过程

1. 如有需要，请启动 `WebSphere_Portal` 服务器。
2. 登录 WebSphere® Integrated Solutions Console。
3. 转至**资源 > 资源环境 > 资源环境提供者**。

4. 选择 **WP 配置服务**。
5. 在其他属性下，选择**定制属性**。
6. 单击**新建**。
7. 指定此案例中属性 WRTCorrelator cookie 的名称，并将属性值设置为忽略。

   要将 WRTCorrelator cookie 设置为忽略，请输入以下代码：

```
cookie.ignore.regex =
digest\.ignore.*|LTPAToken|LTPAToken2|JSESSIONID|WASReqURL|WRTCorrelator
|PD_STATEFUL.*
```

8. 单击**应用**，然后保存更改。
9. 从 WebSphere® Integrated Solutions Console 注销。

结果

将 WRTCorrelator cookie 设置为忽略，就可以防止诸如循环的问题。

从客户机时间报告中排除页面

您可能想要从客户机时间报告中排除特定的浏览器页面。

关于此任务

您可以将参数添加到配置文件，以阻止 IBM HTTP Server 响应时间模块将 JavaScript 插入到与指定模式匹配的任何文件中。这样将阻止针对这些页面从浏览器执行客户机时间报告。

过程

要从浏览器中的客户机时间报告中排除特定页面，请完成下列步骤：

1. 在文本编辑器中打开以下文件：

   Linux AIX *install_dir*/config/*hostname*_t5.cfg，其中 *install_dir* 为 /opt/ibm/apm/agent

   Windows *install_dir*\TMAITM6_x64*hostname*_t5.cfg，其中 *install_dir* 为 C:\IBM\APM

2. 在 **advconfig** 部分中，添加

```
{KT5WEBPLUGIN_JSI_EXCLUDE_URI_WITH_PATTERNS=URL_path_pattern to_exclude}
```

   例如，

```
{KT5WEBPLUGIN_JSI_EXCLUDE_URI_WITH_PATTERNS=*/DoNotJSIMe.jsp,/absolutePath/index.jsp,/
skipThisDir/*}
```

   URL 路径的长度限制为 256 个字符。

   提示: 使用星号 (*) 作为前缀或后缀以与模式匹配。如果需要，将以逗号分隔的多个值添加到参数。

3. 保存并关闭 *hostname*_t5.cfg。
4. 重新启动 Response Time Monitoring 代理程序：

```
rt-agent.sh stop
rt-agent.sh start
```

以非 *root* 用户身份使用 *IBM HTTP Server* 响应时间模块

在进行谨慎设置后，您可以将 IBM HTTP Server 响应时间模块 与 root 以外的用户标识配合使用。请记住，如果您使用的是 Network Packet Analyzer，您必须使用 root 用户。

要使用 root 以外的用户标识，请遵循下列准则。

对于 Response Time Monitoring 代理程序

使用您运行 Response Time Monitoring 代理程序时要使用的用户标识将该代理程序安装至具有写访问权的目录。

- 对于此过程，Response Time Monitoring 代理程序用户标识为 *agentuser*。代理程序的安装目录为 *$AGENT_HOME*。
- 如果您以 root 用户身份安装 Response Time Monitoring 代理程序，然后以另一个用户身份运行该代理程序，那么将无法创建文件。

对于 IBM HTTP Server 响应时间模块

ServerRoot 的所有者必须为用于运行 apache start|stop 的用户标识。

- 在 apache start 执行期间，将在 ServerRoot 下创建一个 wrt 跟踪目录。因此，用户需要具备足够的权限才能在 ServerRoot 下创建文件和目录。
- 如果 IBM HTTP Server 响应时间模块 用户 *ihsuser* 不同于 *agentuser*，那么它需要对 $AGENT_HOME/tmp 的写访问权。

  $AGENT_HOME/tmp 是在安装代理程序期间创建的。*ihsuser* 需要具备在 $AGENT_HOME/tmp 中创建 kt5 目录的许可权。
- 可能存在多个版本的 ServerRoot，每个版本都由不同的用户管理。

对于 Response Time Monitoring 代理程序和 IBM HTTP Server 响应时间模块

agentuser 和 *ihsuser* 需要具备对下列目录的读/写访问权：

- $AGENT_HOME/tmp/kt5
- ServerRoot/wrt

通常，将先启动 IBM HTTP Server 响应时间模块，然后 Response Time Monitoring 会在其初次启动时自动创建 wrt 目录，具备对所有内容的读/写访问权。

ServerRoot/wrt 还由 camconfig 用于推送配置。*ihsuser* 将创建由 *agentuser* 选取的共享队列标识文件；*agentuser* 将从目录中读取队列标识并推送配置。

如果最初使用了 root 运行 apache start 并且 *ihsuser* 不是 root，请完成下列步骤：

1. 停止 Response Time Monitoring 代理程序。
2. 在使用 root 的情况下，运行 apachectl stop。
3. 删除下列目录：
   - $AGENT_HOME/tmp/kt5
   - ServerRoot/wrt
4. 使用 *ihsuser*，在具备正确许可权的情况下运行 apachectl start 重新创建目录。

有关重置 IBM HTTP Server 响应时间模块的 wrt 目录的许可权

您可以重置 IBM HTTP Server 响应时间模块 wrt 目录许可权以获取更多安全性。此更新涉及到在配置文件中添加参数，以限制非 root 用户在 IBM HTTP Server 安装期间创建的 wrt 目录的许可权。

在安装 IBM HTTP Server 响应时间模块 的系统上完成此过程，以将非 root 用户的 wrt 目录的许可权从 777 更改为 700：

1. 在文本编辑器中打开 IHS_HOME/conf/httpd.conf 文件。
2. 在文件末尾添加属性 **WrtDisableDirPermNonRoot**：
   - 当启用该属性时，只有已启动 httpd 且与创建了目录的用户标识匹配的用户标识才可用于创建具有许可权 700 的 wrt 目录。所有其他用户都被拒绝访问此目录。
   - 当未启用该属性时，使用缺省许可权 777 创建 wrt 目录。

3. 重新启动 Response Time Monitoring 代理程序：

```
rt-agent.sh stop
rt-agent.sh.start
```

Response Time Monitoring 配置数据和部分持久性文件存储在通信过程中使用的 `wrt` 目录中，并且每个连接进程都会在目录中创建一个 wrt 配置文件。将 **`WrtDisableDirPermNonRoot`** 添加到 `httpd.conf` 文件后，只有某个受限的用户可与 Response Time Monitoring 监视代理程序成功进行通信。

使用负载均衡器
如果正在环境中使用负载均衡器，那么可能需要某些其他定制。

过程

如果要使用负载均衡器，请遵循下列准则：

1. 关闭负载均衡器上的“URL 重写”。
2. 在要监视的每个 Web 服务器上安装一个 Response Time Monitoring 代理程序。
   请勿在负载均衡器上安装 Response Time Monitoring。

下一步做什么

如果在负载均衡器下运行 Response Time Monitoring 代理程序，那么可以配置负载均衡器以转发客户机的 IP 地址来优化监视性能。使用以下步骤作为一个示例：

1. 在 HTTP 头的 **`X-Forwarded-For`** 字段中设置客户机的 IP 地址。
2. 在 `$AGENT_HOME/config/`*`hostname`*`_t5.cfg` 文件中，将 `{KT5WEBPLUGIN_OVERRIDE_SOURCE_ADDR_HEADERS=X-Forwarded-For}` 添加到 `SECTION=advconfig` 部分。

   提示: 如果需要，将多个值添加到参数。例如，`{KT5WEBPLUGIN_OVERRIDE_SOURCE_ADDR_HEADERS=x-forwarded-for, iv-remote-address}`
3. 重新启动 Response Time Monitoring 代理程序。运行以下命令：

```
rt-agent.sh stop
rt-agent.sh start
```

限制用于监视 *IBM HTTP Server* 的 *CPU*
在饱和环境中，您可能希望限制 IBM HTTP Server 检测使用的 CPU 百分比。

关于此任务

指定 IBM HTTP Server 响应时间模块可以使用的 CPU 百分比。缺省情况下，不限制使用的 CPU 百分比。在安装代理程序的服务器上配置 CPU 百分比。

过程

要配置所使用的 CPU 的百分比，请完成下列步骤：

1. 以交互或手动方式重新配置代理程序：
   - 运行用于以交互方式配置的代理程序脚本：

     Linux AIX

     ```
     $AGENT_HOME/bin/rt-agent.sh config
     ```

     Windows

     ```
     install_dir\BIN\rt-agent.bat config
     ```

- 在文本编辑器中打开以下文件：

  Linux AIX

  ```
  /opt/ibm/apm/agent/config/hostname_t5.cfg
  ```

  Windows

  ```
  C:\IBM\APM\TMAITM6_x64\hostname_T5.cfg
  ```

2. 在 **advconfig** 部分中，添加以下参数并设置范围为 0 到 100 的值：

   ```
   KT5WEBPLUGIN_TARGET_CPU_PERCENTAGE=10
   ```

   其中，指定的值是 CPU 使用的百分比限制。缺省值 *0* 表示 CPU 使用不受限制。
3. 您还可以设置以下参数：

选项	描述
KT5WEBPLUGINCONFIGPOSTURL	对应于 IBM HTTP Server 安装的 URL 列表。 缺省值：`http://localhost/WrtUpdateConfig.dat`
KT5WEBPLUGIN_MAX_REQUESTS_PER_SECOND	每秒每个 IBM HTTP Server 安装监视的请求数。如果请求数量超出此数目，那么将不会监视后续请求。如果到达限制，那么 JavaScript 插入将停止，并且不会将任何数据发送回 Response Time Monitoring 代理程序。 缺省值：0（无最大值）
KT5WEBPLUGIN_CPUMAN_PERIOD_IN_SEC	检查 CPU 使用率以确定是否已超过目标的周期（以秒为单位）。 缺省值：60 秒
KT5WEBPLUGIN_CATCHUP_PERIOD_COUNT	缩减 CPU 前允许处于相同状态的周期的数目。例如，通过使用缺省值，如果 CPU 使用率较高，并且在 4 个周期后仍然较高，那么将缩减 CPU 使用率。 缺省值：3

结果
可用于 IBM HTTP Server 响应时间模块的 CPU 百分比设置为固定百分比。

Packet Analyzer 路线图

使用 Packet Analyzer 可监视 HTTP 事务。您必须手动配置 HTTPS 监视。您需要手动检测 Web 页面以收集浏览器计时。

要确定可使用 Packet Analyzer 的环境，请参阅 第 584 页的『Response Time Monitoring 组件』和第 584 页的『规划安装 』。

安装 Response Time Monitoring 代理程序时，将自动启用 Packet Analyzer，但您可能需要执行许多其他步骤和定制。

1. 您可以在“代理程序配置”窗口中定制 Packet Analyzer 设置，例如端口号。有关更多信息，请参阅第 600 页的『使用“代理程序配置”窗口来配置 Packet Analyzer』。
2. 要监视 HTTPS 事务，请手动检测 Web 页面以收集浏览器计时。有关更多信息，请参阅第 602 页的『监视 HTTPS 事务』。

3. 要启用浏览器计时，请将“JavaScript 注入”组件添加到应用程序，并将 JavaScript 监视组件与应用程序相关联，有关更多信息，请参阅第 600 页的『将 JavaScript 监视组件添加到应用程序』。
4. 在事务负载较高的环境中，可能需要执行一些高级调整步骤。有关更多信息，请参阅第 606 页的『Packet Analyzer 的高级配置』

使用“代理程序配置”窗口来配置 Packet Analyzer

您可使用“代理程序配置”窗口来配置 Packet Analyzer。

要使用 Packet Analyzer 监视特定系统的 HTTP 流量，请完成下列步骤：

1. 要访问“代理程序配置”页面，请在 APM UI 中选择**系统配置 > 代理程序配置**，然后选择**响应时间**选项卡。
2. 选择想要更新的一个或多个系统。如果想要将相同的 HTTP 设置用于多个系统中的每一个系统，请选择多个系统。

   如果选择的系统设置了不同的 HTTP 值，那么将显示 `Multiple Values` 或 `Multiple Lists`，而不是个别值。无法同时使用不同的值更新系统。
3. 在**是否监视 HTTP 流量?** 字段中，双击值并从列表中选择**是**。
4. 在**要监视的 HTTP 端口**字段中，双击值并输入除缺省端口 80 和已列出的任何其他端口之外您要监视的任何额外端口。

   要停止监视某个端口，请选择不想再监视的端口，然后单击**移除**。
5. 单击**应用更改**。

将 JavaScript 监视组件添加到应用程序

为帮助您了解浏览器中 Web 页面的性能和任何错误，Response Time Monitoring 代理程序需要能够从浏览器收集计时数据。要启用此功能，必须配置所要监视的应用程序。

关于此任务

您需要将 JavaScript 监视组件添加到应用程序的每个 Web 页面，然后才能监视 Web 页面内的交互。JavaScript 监视组件将捕获每个 Web 页面及其关联 JavaScript 交互的状态。将 JavaScript 监视组件添加到所要监视的应用程序。这将自动捕获相关的内容和操作，并将其发送到 Cloud APM 服务器以进行分析和关联。

过程

完成以下步骤以从浏览器启用实时用户监视数据的收集。这些步骤只需要完成一次，除非应用程序配置发生更改。

1. 将 JavaScript 监视组件添加到应用程序。您使用的过程取决于应用程序类型：
   a) 对于 Java EE 应用程序，请将 *install_dir*`/clienttime/ClientTime.war` 从安装包解压到 HTTP Server 可访问的目录。
   b) 对于非 Java EE 应用程序（例如，Ruby、.NET、Python 和 Node.js），将 *install_dir*`/clienttime/wrtInstrumentation.js` 从安装包保存到 HTTP Server 可访问的目录。

   将 *install_dir*`/clienttime/ClientTime.war` 文件解压到临时路径。必须将已解压的 `wrtTimingTarget.dat` 文件复制到文档根目录。文档根目录是在 HTTP Server（Apache、IIS 等）上设置的。它是用于存储文档的目录。缺省情况下，从该目录提取所有请求，但可通过使用符号链接和别名来指向其他位置。例如，Apache 的文档根目录为 `/opt/IBM/HTTPServer/htdocs`。

   `wrtInstrumentation.js` 文件可置于任意目录中。请确保在 HTML 头中更新指向 `wrtInstrumentation.js` 文件的路径位置。
2. 将 JavaScript 监视组件与应用程序相关联。

   此关联操作通常可通过修改应用程序标头脚本来完成。通常，针对要监视的每个组件或应用程序，仅需要修改一个标头脚本。

对于 Java EE 应用程序和非 Java EE 应用程序，将以下 JavaScript 添加到应用程序标头中任何其他 JavaScript 之前：

```
<script language="JavaScript" src="path/wrtInstrumentation.js"
type="text/JavaScript"></script>
```

其中，*path* 是 JavaScript 监视组件的相对路径。

例如：

```
<script language="JavaScript" src="/ClientTime/wrtInstrumentation.js"
type="text/JavaScript"></script>
```

结果

将对使用 JavaScript 监视组件检测的页面进行监视，并对这些页面中的数据进行分析，然后将数据显示在最终用户事务仪表板中。

启用浏览器计时

通过启用资源计时监视，Response Time Monitoring 代理程序可使用 Packet Analyzer 来处理 W3C 资源计时数据。通过启用此功能，您可以查看有关前端元素的详细性能信息。

关于此任务

必须先将资源计时监视组件添加到应用程序，并将资源计时监视组件与该应用程序相关联，然后才能监视资源计时数据。资源计时监视组件自动捕获前端元素的状态和交互，并将数据发送到 Cloud APM 服务器以进行分析。此分析的结果会显示在**子事务**仪表板中。

过程

完成以下步骤以启用资源计时监视功能。仅完成这些步骤一次，除非应用程序配置发生更改。

1. 将资源计时监视组件添加到应用程序。
   a) 从安装程序包解压缩 *install_dir*/clienttime/wrtInstrumentation.js 文件。
   b) 将 wrtInstrumentation.js 文件添加到应用程序的 JavaScript 目录。
2. 将下面这一行追加到应用程序标头中：

```
<script> var wrt_enableResourceTiming=true; </script>
```

例如，

```
<script language="JavaScript" src="path/wrtInstrumentation.js"
type="text/JavaScript"></script>
<script> var wrt_enableResourceTiming=true; </script>
```

结果

将使用资源计时监视组件来检测页面。此组件在缺省情况下已启用。将在**子事务**仪表板中分析和显示使用资源计时监视组件检测的页面的相关资源计时数据。

下一步做什么

如果想要禁用资源计时监视组件，请将 **wrt_enableResourceTiming** 参数设置为 false。

监视 HTTPS 事务

缺省情况下，Response Time Monitoring 监视 HTTP 事务。要监视 HTTPS 事务，Response Time Monitoring 需要访问 SSL 证书，从而可以解密来自于远程 Web 服务器的 SSL 流量。

开始之前

标识想要监视的 HTTPS Web 服务器，包括其 IP 地址和配置的端口。例如，192.168.1.23，端口 443。对于每个 HTTPS Web 服务器，检查 Response Time Monitoring 是否可读取其密码。Response Time Monitoring 支持 IBM Java 所支持的密码，包括以下密码。

- RSA_WITH_RC4_40_MD5
- RSA_WITH_RC4_128_MD5
- RSA_WITH_RC4_128_SHA
- RSA_WITH_RC4_40_SHA
- RSA_WITH_DES40_CBC_SHA
- RSA_WITH_DESC_CBC_SHA
- RSA_WITH_3DES_EDE_CBC_SHA
- RSA_WITH_AES_128_CBC_SHA
- RSA_WITH_AES_256_CBC_SHA
- RSA_EXPORT1024_WITH_RC4_56_MD5
- RSA_EXPORT1024_WITH_RC2_CBC_56_MD5
- RSA_EXPORT1024_WITH_DES_CBC_SHA
- RSA_EXPORT1024_WITH_RC4_56_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256

限制: Response Time Monitoring 无法对使用 Diffie-Hellman 密钥交换的流量进行解密。

过程

要启用 HTTPS 事务监视，请完成下列步骤：

1. 设置密钥库。有关更多信息，请参阅第 603 页的『设置密钥库』。
2. 通过运行以下一条命令并在提示时提供值，来配置 Response Time Monitoring 代理程序：
   例如：

```
Configuring Response Time Monitoring Agent
Edit 'Response Time Monitoring Agent' settings? [1=Yes,2=No](default is: 1): 1

Basic Configuration :   Specify basic monitoring configuration. Note: HTTP is
now configured centrally using the Response Time tab under Agent Configuration.

Specifies if HTTPS transactions should be monitored
Monitor HTTPS transactions [ 1=Yes, 2=No ] (default is:2): 1

This keystore contains the certificates of the HTTPS websites being monitored
HTTPS keystore (e.g. - /tmp/keys.kdb) (default is: ): /tmp/keys.kdb

This table maps HTTPS servers to the appropriate certificates (e.g. cert1,
server ip,server port; cert2,server2 ip,server2 port);...
HTTPS server certificate map (eg - certAlias,9.48.152.1,443;...)(default is: ):
 label1,10.0.0.1,9443;label1,9.185.150.71,443

Advanced Configuration :
Specify advanced monitoring configuration

The NIC card which has the selected IP address will be monitored.
IP address of the NIC to be monitored (default is: ): 10.0.0.1

Data Collection and Analysis Configuration :
Specify Configuration Information on how Data is Analyzed.
```

```
Configuration completed successfully.
Agent restart required to apply configuration changes.
```

其中：

- HTTPS 密钥库是步骤 1 中配置的密钥库
- HTTPS 服务器证书映射，指定：
  - `label 1` - 步骤 1 中配置的密钥标签
  - `server ip` - 服务器的 IP 地址，它必须与 IPv4 包头中的 Source/Destination 属性相匹配
  - `server port` - 服务器端口号，它必须与包的 TCP 头中的“源/目标端口”属性相匹配

  针对密钥标签相同的多个可能的服务器 IP 添加多个条目。
- IP address of the NIC to be monitored - 可查看包并且映射到 eth0 和 en0 等的接口。名称无需匹配 IPV4 或 TCP 包头的任何属性。如果 `10.0.0.1` 对应于 eth0，那么使用 `tcpdump -s0 -i eth0 ...` 以查看 Packet Analyzer 需要分析的所有包。

3. 重新启动 Response Time Monitoring 代理程序。

设置密钥库
要监视 HTTPS 事务，那么针对要监视的所有 Web 服务器，将密钥导入 KT5Keystore。

关于此任务

您可以从所监视的 Web 服务器中导出 SSL 证书，并使用 IBM Key Management (iKeyman) 将它们导入 HTTPS 密钥库，或者在 HTTPS 密钥库中指定 Web 服务器的密钥库隐藏项文件 (`.kdb`)。在安装或配置 Response Time Monitoring 时，将提示您输入 `keys.kdb` 文件的位置。

如果没有密钥库隐藏项文件（`.kdb` 和 `.sth`），那么确认在 Java 版本中启用了 CMS Provider，以便可使用 iKeyman 来设置密钥数据库：

1. 转至 *install_dir*`/ibm-jre/jre/lib/security` 目录。例如：
   - Linux `/opt/ibm/apm/agent/JRE/lx8266/lib/security`
   - Windows `C:\Program Files\IBM\APM\ibm-jre\jre\lib\security`
2. 在 `java.security` 文件中，按所示将以下语句添加到安全提供程序列表，其中 *number* 是列表中最后一个序号。

```
security.provider.number=com.ibm.security.cmskeystore.CMSProvider
```

提供者列表类似于以下示例：

```
## List of providers and their preference orders #
security.provider.1=com.ibm.jsse.IBMJSSEProvider
security.provider.2=com.ibm.crypto.provider.IBMJCE
security.provider.3=com.ibm.security.jgss.IBMJGSSProvider
security.provider.4=com.ibm.security.cert.IBMCertPath
security.provider.5=com.ibm.security.cmskeystore.CMSProvider
...
#
```

3. 保存并关闭该文件。

限制: Response Time Monitoring 无法使用 Diffie-Hellman 密钥交换对流量进行解密。

过程

要启用 HTTPS 事务监视，请从要监视的 Web 服务器中收集 SSL 证书，并使用 iKeyman 将证书和密钥库隐藏项文件导入 HTTPS 密钥库。以下示例使用 iKeyman 从 IBM HTTP Server 中导出证书，并将它们导入 HTTPS 密钥库：

1. 在要监视的每个 HTTPS Web 服务器上安装一个 Response Time Monitoring 代理程序。

2. 根据您的操作系统，通过运行下列其中一项命令来从 IBM Java bin 目录中运行 **IBM Key Management** (iKeyman)。
   - AIX Linux `/opt/ibm/apm/agent/JRE/lx8266/bin/ikeyman`

     注: 您的环境中必须有 X-Window 才能使 iKeyman 正常工作。
   - Windows `c:\IBM\APM\java\java80_x64\jre\bin\ikeyman`
3. 创建新的密钥库数据库。在**新建**对话框中，完成下列步骤：
   a) 从**密钥数据库类型**列表中，选择 **CMS**。
   如果列表中未提供 CMS，那么可能未启用 CMS Provider。请在 Java 安全性文件中启用 CMS Provider。
   b) 在**文件名**字段中，输入 HTTPS 密钥库文件的名称，然后单击**确定**。
   例如，`keys.kdb`。
4. 在**密码提示**对话框中，完成下列步骤：
   a) 在**密码**和**确认密码**字段中，输入并确认用于访问 `keys.kdb` 的密码。
   请不要设置到期时间，除非您希望定期重新创建密钥库数据库及重新启动 Response Time Monitoring 代理程序。
   b) 选择**将密码隐藏到文件?** 以在隐藏项文件 `keys.sth` 中通过加密格式存储 `keys.kdb` 的密码。

   注: “响应时间”代理程序仅支持隐藏的密码 V1。APM 8.1.4 后，运行以下命令以将 keys.kdb 密码存储在加密的隐藏项文件 keys.sth 中。

   在 Linux 上：

   ```
   cp keyfile.sth keyfile.sth.new-format
   ```

   ```
   cd /opt/IBM/ccm/agent/lx8266/gs/bin
   ```

   ```
   #export LD_LIBRARY_PATH=/opt/ibm/apm/agent/lx8266/gs/lib64:$LD_LIBRARY_PATH
   ```

   ```
   ./gsk8capicmd_64 -keydb -stashpw
   -db /opt/IBM/ccm/agent/keyfiles/keyfile.kdb -v1stash
   ```

   在 Windows 上：

   ```
   copy server.sth server.sth.backup
   ```

   ```
   set PATH=c:\IBM\APM\GSK8_x64\lib64;%PATH%
   ```

   ```
   C:\IBM\APM\GSK8_x64\bin\gsk8capicmd_64 -keydb
   -stashpw -db .\server.kdb -pw passw0rd -v1stash
   ```
5. 在 iKeyman 窗口的**密钥数据库内容**部分中，完成下列步骤：
   a) 选择**个人证书**。
   b) 单击**导入**。
   c) 在**导入密钥**对话框中，从**密钥文件类型**列表中，选择 **CMS**。
   d) 浏览至密钥库文件并单击**打开**，然后单击**确定**。
   e) 在**密码提示**对话框中，输入密钥库密码。
   f) 从列表中选择密钥，然后单击**确定**。
   g) 在**更改标签**对话框中，选择密钥标签名称。在**输入新标签**字段中，指定服务器的主机名，然后单击**应用**。

   注: 配置 Response Time Monitoring 时您需要这个值，因此请将其记下。
   h) 单击**确定**。
6. 保存 HTTPS 密钥库。

从 Internet Information Services 中导入密钥

要从 Internet Information Services 抽取密钥并将其导入到 KT5Keystore 中，请完成下列步骤：

1. 在要监视的每个 HTTPS Web 服务器上安装一个 Response Time Monitoring 代理程序。
2. 从 Internet Information Services 导出 .pfx 文件：
   a. 从 Windows **开始**菜单中，选择**管理工具 > Internet Information Services (IIS) Manager**。
   b. 选择要导出其专用密钥的 Web 服务器和站点，然后右键单击并从上下文菜单中选择**属性**。
   c. 选择**目录安全性**选项卡，然后在**安全通信**部分中选择**服务器证书**。
   d. 在 **IIS 证书向导**中，单击**下一步**。
   e. 选择**将当前证书导出到 .pfx 文件**，然后单击**下一步**。
   f. 输入路径和文件名称，然后单击**下一步**。
   g. 为密钥输入导出密码，然后单击**下一步**。
   h. 在所有后续页面上单击**下一步**，然后单击**完成**。
3. 从 .pfx 文件中抽取个人证书和签署者证书：
   a. 使用命令 `c:\IBM\APM\java\java80_x64\jre\bin\ikeyman`，从 IBM Java `bin` 目录运行 **IBM Key Management** (iKeyman)。确保设置了环境变量 `JAVA_HOME`。
   b. 在密钥库数据库中，选择**文件 > 打开**。
   c. 从**密钥数据库类型**列表中，选择 **PKCS12**。
   d. 输入您在上面创建的 .pfx 文件的名称和路径，然后单击**确定**。出现提示时，输入密码，然后单击**确定**。
   e. 选择**密钥数据库内容 > 个人证书**，然后单击**导出/导入**。
   f. 选择操作类型**导出密钥**和密钥文件类型 **PKCS12**。为导出的密钥输入文件名称和位置，然后单击**确定**。出现提示时，输入导出密码，然后再次单击**确定**。
   g. 如果个人证书由认证中心签署，请选择**密钥数据库内容 > 签署者证书**，然后单击**抽取**。选择缺省文件类型，为导出的证书输入文件名和位置，然后单击**确定**。
4. 抽取签署者 `.cer` 文件（如果需要）：
   a. 如果从 .pfx 文件中抽取签署者证书文件，请浏览至它的保存目录，并使用扩展名 `.cer` 生成新副本。双击新副本以使用 Windows 证书查看器将其打开。
   b. 在**证书路径**选项卡上，您可以查看签署者证书链。处于链中最底部的项目应该是个人证书。对于它上面的所有证书，请执行以下操作：
      1) 选择证书并单击**查看证书**。
      2) 选择**详细信息**并单击**复制到文件**。
      3) 在“证书导出”向导中接受所有缺省值并输入具有 `.cer` 扩展名的文件名。
5. 创建新的密钥库数据库。在**新建**对话框中，完成下列步骤：
   a. 从**密钥数据库类型**列表中，选择 **CMS**，然后输入文件名和位置。出现提示时，为新密钥库输入密码。

      注: 确保选择**将密码隐藏到文件**。
   b. 如果从 .pfx 中抽取签署者证书，请执行以下操作：
      1) 选择**密钥数据库内容 > 签署者证书**。
      2) 对于每个签署者证书，请单击**添加**并添加 `.cer` 文件。
   c. 选择**密钥数据库内容 > 个人证书**并单击**导入**。
   d. 选择密钥文件类型 **PKCS12**，以及 **.p12** 文件的名称和位置。在出现提示时输入密码。
   e. 保存密钥库并退出密钥管理实用程序。
   f. 在 Response Time Monitoring 设备机器上将 `.kdb` 和 `.sth` 文件复制到 KT5Keystore。

g. 在安全目录中放置 IBM Key Management 数据库文件 (.kdb) 和隐藏项文件 (.sth)，并确保它们只能被管理员或 root 用户（或用于安装 Response Time Monitoring 代理程序的用户标识）读取。

Packet Analyzer 的高级配置

Packet Analyzer 有多个高级配置选项。

配置 Packet Analyzer 之后，您可执行多个高级配置任务，以微调性能和功能。

使用负载均衡器

如果正在环境中使用负载均衡器，那么可能需要某些其他定制。

过程

如果要使用负载均衡器，请遵循下列准则：

1. 关闭负载均衡器上的“URL 重写”。
2. 在要监视的每个 Web 服务器上安装一个 Response Time Monitoring 代理程序。
   请勿在负载均衡器上安装 Response Time Monitoring。
3. 将 JavaScript 监视组件添加到应用程序。有关更多信息，请参阅第 600 页的『将 JavaScript 监视组件添加到应用程序』。

下一步做什么

如果在负载均衡器下运行 Response Time Monitoring 代理程序，那么可以配置负载均衡器以转发客户机的 IP 地址来优化监视性能。使用以下步骤作为一个示例：

1. 在 HTTP 头的 **X-Forwarded-For** 字段中设置客户机的 IP 地址。
2. 配置 Response Time Monitoring 代理程序以使用客户机的 IP 地址头。根据您的操作系统，在下列其中一个文件的 **KFC_OVERRIDE_SOURCE_ADDR_HEADER** 字段中设置客户机的 IP 地址头：
   - AIX Linux /opt/ibm/apm/agent/tmaitm6/wrm/kfcmenv
   - Windows C:\IBM\ITM\TMAITM6_x64\wrm\Analyzer\kfcmenv

   例如：

   ```
   KFC_OVERRIDE_SOURCE_ADDR_HEADER=x-forwarded-for
   ```

   或者，如果您使用的是 WebSEAL：

   ```
   KFC_OVERRIDE_SOURCE_ADDR_HEADER=iv-remote-address
   ```

配置 *CPU* 开销限制

如果要运行高事务负载环境，那么可以限制 Response Time Monitoring 代理程序所使用的监视资源。

关于此任务

此功能通过使用采样来仅监视并报告一部分 Web 流量，从而限制 Response Time Monitoring 代理程序的 CPU 使用率。缺省情况下，未配置 CPU 开销限制。您必须在安装代理程序的服务器上配置 CPU 开销限制。

过程

要配置 CPU 开销限制，请完成下列步骤：

1. 在文本编辑器中打开以下文件：
   Linux AIX /opt/ibm/apm/agent/tmaitm6/wrm/kfcmenv
   Windows C:\IBM\ITM\TMAITM6_x64\wrm\Analyzer\kfcmenv
2. 配置以下参数的值：

KFC_MAX_PROTOCOL_PACKETRATE
初始最大包速率。例如，如果将参数设置为 **KFC_MAX_PROTOCOL_PACKETRATE=2000**，那么最大包速率为每秒 2000 个包。此速率会根据 **KFC_CPUTHROTTLE_TARGET** 和当前 CPU 使用率的值动态变化。

KFC_CPUTHROTTLE_TARGET
总体 CPU 资源中 kfcmserver 进程可使用的百分比。例如，如果将参数设置为 **KFC_CPUTHROTTLE_TARGET=10.0**，那么 kfcmserver 进程最多可使用总体 CPU 资源的 10%。

注: 参数 **KFC_CPUTHROTTLE_TARGET** 的值是总体 CPU 资源中可用于 kfcmserver 进程的百分比。例如，如果您有 4 个 CPU 核心并且 **KFC_CPUTHROTTLE_TARGET** 设置为 10，那么 Windows 中的资源监视器会将 CPU 资源计为 400%。因此，kfcmserver 进程可使用 400% 总可用 CPU 资源的最多 40%。

结果
针对 Response Time Monitoring 代理程序配置了 CPU 开销限制。

重新配置以从 IBM HTTP Server Response Time 模块切换为 Packet Analyzer

您可能希望将监视环境从 IBM HTTP Server Response Time 模块更改为 Packet Analyzer。

过程

1. 卸载 HTTP Server 代理程序
   a) 注释 Response Time 插件行以编辑 /etc/httpd/conf/httpd.conf。例如:
   ```
   #include /opt/ibm/apm/agent/tmp/khu/khu.etc.httpd.conf.httpd.conf
   ```
   b) 卸载 HTTP Server 代理程序。例如:
   ```
   /opt/ibm/apm/agent/bin/http_server-agent.sh uninstall
   ```
   c) 在下一步中重新配置 Response Time 代理程序之前，请先打开新命令提示符窗口以清除系统变量。
2. 打开以下文件
   Linux AIX *install_dir*/config/*hostname*_t5.cfg
   其中，*install_dir* 为 /opt/ibm/apm/agent
   Windows *install_dir*\TMAITM6_x64*hostname*_t5.cfg
   ，其中 *install_dir* 为 C:\IBM\APM
3. 按如下所示设置参数:
   ```
   { KT5DISABLEANALYZER=NO } { KT5ENABLEWEBPLUGIN=NO }
   ```
4. 重新配置代理程序，如下所示:
   ```
   re-agent.bat config  install_dir\samples\rt_silent_config.txt
   ```

定制最终用户事务位置值

可以在“最终用户事务”仪表板中针对您的特定环境定制应用于特定 IP 地址或地址范围的位置。

开始之前

在**代理程序配置**中使用**地理位置**选项卡来定制位置值。

使用此功能可以设置仪表板中显示为**未知**的 IP 地址的位置。这些地址可能是内部地址（例如，192.168.x.x 或 10.x.x.x），也可能是未解析的外部 IP 地址。您还可以使用此功能来覆盖 IP 地址的错误位置。例如，如果您知道 IP 地址 9.1.1.1 位于洛杉矶，但它显示为旧金山，请覆盖该位置并将 9.1.1.1 设置为洛杉矶。

关于此任务

通过上载包含所需值的 CSV 文件，在“最终用户事务”仪表板中定制位置值。您可以在**地理位置**选项卡上查找样本 CSV 文件。

CSV 文件必须具有下列值作为标题。这些值可以处于任何顺序，并且条目必须与该顺序匹配。

```
IP_ADDRESS, COUNTRY, REGION, CITY
```

例如，

```
IP_ADDRESS, COUNTRY, REGION, CITY
10.0.5.0/24, Australia, WA, Perth
10.1.0.6, Australia, VIC, Melbourne
```

您可以指定单个 IPv4 地址或某个范围。如果指定了范围，请确保使用范围 1-32 的有效值。

过程

要定制“最终用户事务”仪表板中显示的位置值，请在应用程序性能仪表板中完成下列步骤：

1. 使用与位置匹配的 IP 地址来设置您的 CSV 文件。
2. 上载 CSV 文件。
   a) 转至**代理程序配置 > 地理位置**。
   b) 单击**上载 CSV**，选择要上载的文件，然后单击**打开**。
      · 确保 CSV 文件先列出大致的 IP 地址范围，然后再列出更具体的 IP 地址。
      · 如果需要，上载多个文件。
      · 如果一个文件中的值覆盖另一个文件中的值，那么较新的文件中的值覆盖第一个文件中的值。
3. 展开**上载结果**以检查错误。检查以下问题：
   · 覆盖
   · 无效的 IP 地址
   · 无效行
   · 长度超过 250 个字符的值

结果

请等待几分钟，并在“最终用户事务”仪表板中查看定制值。

下一步做什么

如果需要，您可以移除定制值。完成下列其中一个步骤：

· 要移除部分定制值，请选择要移除的 IP 地址，单击**清除所选条目**，然后单击**确定**以确认移除
· 要移除所有定制值，请单击**清除所有条目**，然后单击**确定**以确认移除

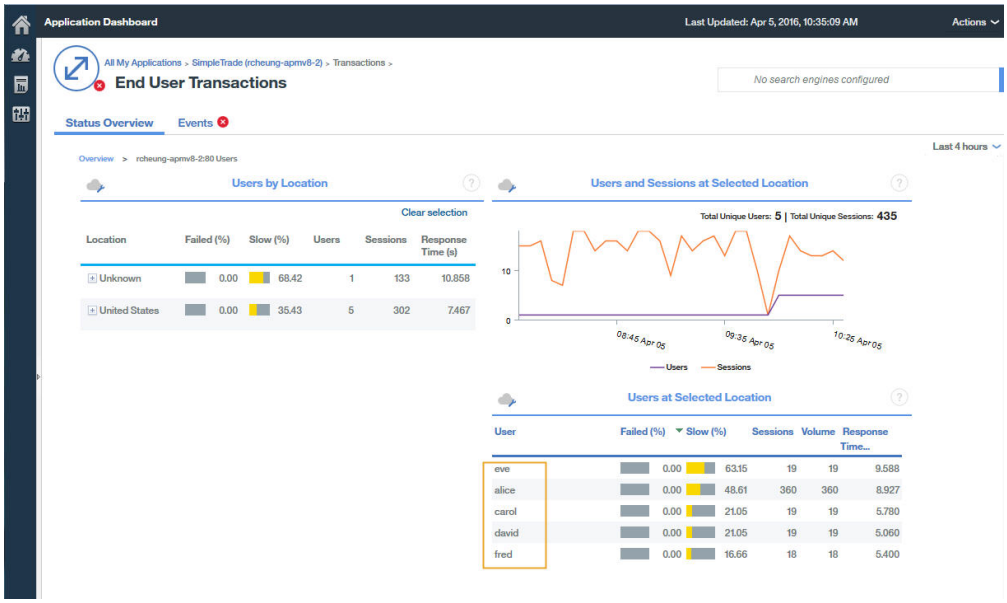
跟踪其他 Web 应用程序

除了缺省跟踪的 Web 应用程序之外，要跟踪其他 Web 应用程序，必须确定和配置用户和会话跟踪方法。

开始之前

如果您的应用程序不受缺省值的支持，那么仪表板不会包含用户和会话详细信息，用户名显示为 anonymous 或 unknown，并且没有可用的会话信息。

如果正确配置了用户跟踪方法，将抽取用户标识并列示在**最终用户事务 > 用户摘要 > 位于所选位置的用户**仪表板中。



注：用户跟踪基于会话跟踪。您需要首先在“响应时间”代理程序配置设置中为用户和会话跟踪方法设置正确的会话跟踪方法变量。

关于此任务

在 IBM Application Performance Management V8.1.4 和更高版本中，可以使用**代理程序配置 > 响应时间**页面来添加要由 Packet Analyzer 或 IBM HTTP Server 响应时间模块跟踪的应用程序。该页面上定义的值优先于 `WRT_Defaults.xml` 文件中的值。

要跟踪其他应用程序，必须先标识想要监视的应用程序的用户标识和会话标识方法与值。例如：

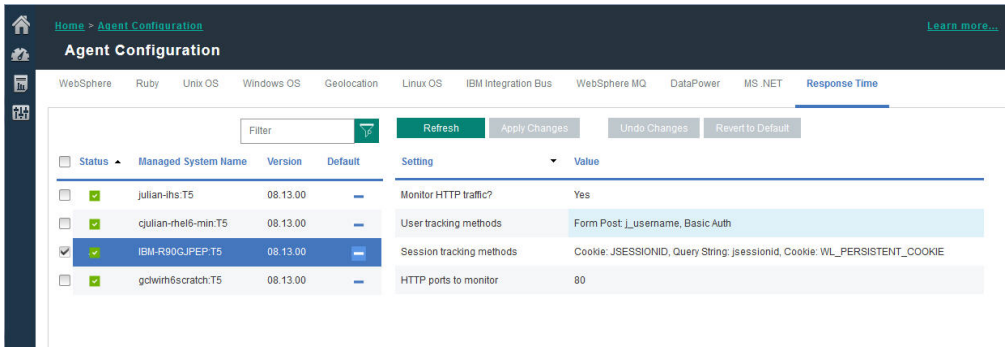
1. 打开浏览器的开发者工具，以便能够看到想要监视的应用程序的请求。
2. 在浏览器网络日志中选择最新请求，以便轻松确定测试请求。
3. 使用您能够是别的参数创建测试请求。例如，使用 `testuser` 登录 Web 站点。
4. 选择该测试请求并查看 `Headers`。
5. 确定请求日志中的会话标识。会话标识通常在 `cookie`、`POST`、`request/response header` 或 `query string` 中指定。如果 `cookie` 已在缺省概要文件中定义，无需在步骤 2 中进行添加。
6. 确定请求日志中的用户标识。用户标识可以在 `cookie`、`request header`、`POST` 或 `query string` 内容中指定。例如，搜索 `testuser` 将会为您提供用户标识的值。
7. 必须以客户的应用程序代码中使用的会话和用户的正确值名称来更新**用户跟踪方法**和**会话跟踪方法**。“会话”和“用户名”的值名称的识别方法取决于应用程序代码。以下是 8.1.4 中的“用户/会话”设置的缺省值。

```
Session tracking methods=cookie\:JSESSIONID,querystring\:jsessionid,cookie\
       :WL_PERSISTENT_COOKIE
User tracking methods=formpost\:j_username,formpost\:uid,formpost\
       :ctl00%24MainContent%24uid,basicauth\:Authorization\: Basic
```

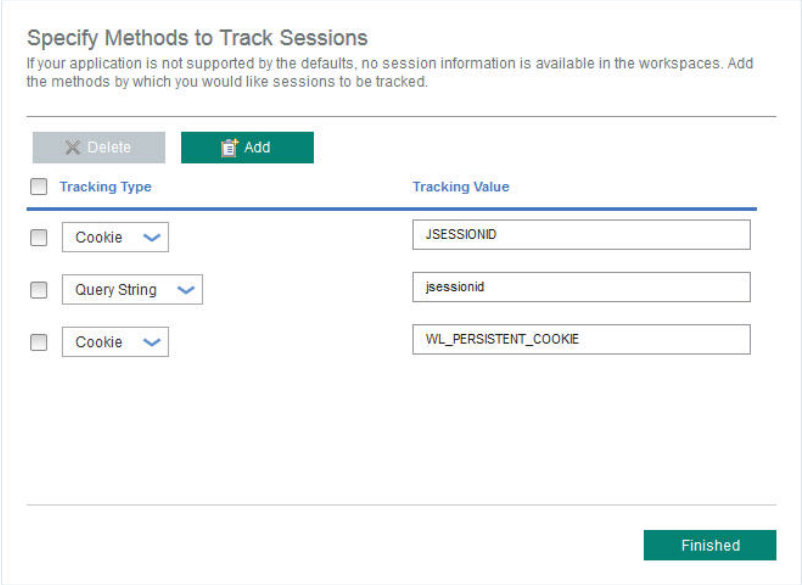
过程

在确定应用程序中使用的用户和会话跟踪方法与值之后，完成下列步骤：

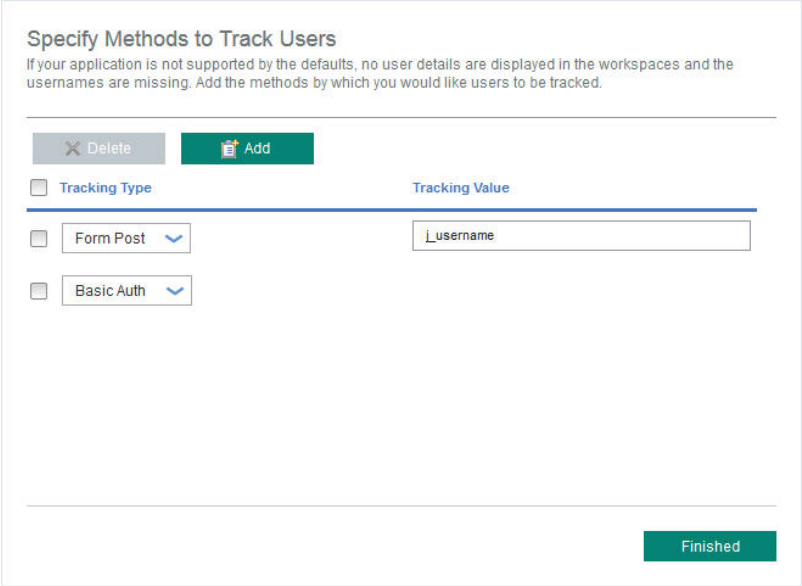
1. 转至**代理程序配置 > 响应时间**配置页面。



2. 选择想要更新的受管系统。
3. 如果需要，更新会话跟踪方法：
   a) 单击**会话跟踪方法**字段中的值。



   b) 在**指定用于跟踪会话的方法**窗口中，单击**添加**。
   c) 从**跟踪类型**列表中，选择跟踪类型。例如，`Cookie`。
   d) 在**跟踪值**字段中，指定值。例如，`WL_PERSISTENT_COOKIE`。
   e) 单击**完成**。
4. 如果需要，更新用户跟踪方法：
   a) 单击**用户跟踪方法**字段中的值。



b) 在**指定用于跟踪用户的方法**窗口中，单击**添加**。

c) 从**跟踪类型**列表中，选择跟踪类型。例如，`Header`。

d) 在**跟踪值**字段中，指定值。例如，`username`。

e) 单击**完成**。

5. 在代理程序配置页面中，单击**应用更改**。

结果

使用新指定跟踪方法的应用程序现在会显示在应用程序仪表板中。

下一步做什么

测试该应用程序中的用户标识和会话信息是否会显示在应用程序仪表板中。

为 Response Time Monitoring 代理程序指定唯一的受管系统名称

显示在 Cloud APM 控制台上的 Response Time Monitoring 代理程序实例名称也称为受管系统名称 (MSN)。您可以使用代理程序配置参数为每个代理程序实例指定唯一的 MSN。

关于此任务

Response Time Monitoring 代理程序的受管系统名称格式如下所示：

```
instancename:hostname:T5
```

T5 是 Response Time Monitoring 代理程序的产品代码。

过程

1. 停止全部现有的代理程序实例。如果不存在任何现有的代理程序实例，请前进到下一步。有关停止代理程序实例的更多信息，请参阅第 150 页的『使用代理程序命令』。
2. Linux AIX 更改 `runagent` 文件中的 **`CTIRA_SUBSYSTEM_ID`**。通常，一台机器上的所有代理程序实例使用同一个主机名值。

   a) 建立此文件的备份副本：

   Linux AIX
   ```
   install_dir/platform/t5/bin/runagent
   ```

   b) 编辑此文件。在 Linux 或 AIX 系统上，请添加 *newinstancename*。

Linux AIX **CTIRA_SUBSYSTEM_ID**=*newsubsystemid*

3. 启动代理程序的现有实例。
4. 启动 Cloud APM 控制台。修改应用程序，移除旧 MSN 下的代理程序实例，并添加新的代理程序实例。

配置 Ruby 监视

您可以监视本地和 IBM Cloud Ruby 应用程序。要监视本地 Ruby 应用程序，请配置 Ruby 代理程序。要监视 IBM Cloud Ruby 应用程序，请配置 Ruby 数据收集器。

关于此任务

依以下指示信息针对的是代理程序的最新发行版，除非另有说明。

以下过程是配置 Ruby 代理程序和 Ruby 数据收集器的路线图，其中包括必需步骤和可选步骤。请根据您的需要，完成配置步骤。

过程

- 要监视本地 Ruby 应用程序，请完成下列步骤以配置 Ruby 代理程序：
  a) 配置代理程序实例，以监视 Ruby 应用程序。请参阅配置 Ruby 代理程序以监视 Ruby 应用程序。
  b) 安装数据收集器，以收集要在 Cloud APM 控制台中显示的监视数据。请参阅安装数据收集器。
  c) 可选： 如果您是 Cloud APM, Advanced 用户，您可根据需要完成下列任务：
  - 要配置数据收集器以收集诊断数据，请参阅配置诊断数据收集器。
  - 要对请求启用方法跟踪，并调整“请求堆栈跟踪”窗口小部件中显示的文件路径参数的长度，请参阅启用方法跟踪和调整路径显示。
  - 要增加 JVM 堆大小，以避免内存不足错误，请参阅增加 JVM 堆大小。
  - 要禁用诊断，请参阅对 Ruby 应用程序禁用或启用诊断数据。
- 要监视 IBM Cloud Ruby 应用程序，请完成下列任务以配置 Ruby 数据收集器：
  a) 为 IBM Cloud 应用程序配置 Ruby 数据收集器。有关指示信息，请参阅第 618 页的『针对 IBM Cloud 应用程序配置 Ruby 数据收集器』。
  b) 可选： 要更改 Ruby 数据收集器的行为，请参阅第 619 页的『针对 IBM Cloud 应用程序定制 Ruby 数据收集器』。

配置 Ruby 代理程序

要使 Ruby 代理程序监视您的应用程序，请指定 Ruby 运行时。这样，您就可以使用该运行时，从 Ruby 应用程序收集数据以及配置此代理程序。

开始之前

确定您用来启动 Ruby 应用程序的服务器，以及此代理程序所使用的 Ruby 或 Rake 可执行文件的限定 bin 目录：

1. 要确定您所使用的应用程序服务器，请运行以下命令：

```
ps -ef | grep ruby
```

您将看到用来启动应用程序的服务器的名称。可能的服务器名称如下所示：

- Passenger
- Unicorn
- Puma
- Thin

如果命令输出未指出以上列表中显示的服务器名称，那么表示用来启动应用程序的服务器可能是 WEBrick。

要点: 如果使用多个 Web 服务器来启动 Ruby 应用程序，那么必须为每个应用程序 Web 服务器创建一个代理程序实例，例如，一个实例用于 PUMA，另一个用于 Unicorn。

2. 要确定 Ruby 代理程序所使用的 Ruby 或 Rake 可执行文件的限定 bin 目录，请运行以下命令:

```
which ruby
```

关于此任务

根据您的需要，您可重复此任务，以配置多个代理程序实例。

过程

1. 要配置代理程序，请运行以下命令:
   install_dir/bin/ruby-agent.sh config *instance_name*
   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 /opt/ibm/apm/agent。

   要点: 请勿指定长实例名称。主机名与代理程序实例名称的总长度不得超过 28 个字符。如果长度超过此限制，那么“受管系统名称”将截断，而且 Ruby 代理程序的产品代码无法正确显示。

   “受管系统名称”包括指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中 *pc* 是代理程序的双字符产品代码。例如，如果指定 Ruby2 作为实例名称，那么受管系统名称为 Ruby2:hostname:KM，其中 *KM* 是 Ruby 代理程序的双字符产品代码。
2. 出现提示 Edit 'Monitoring Agent for Ruby' settings 时，请输入 1 以继续。
3. 出现提示 Fully Qualified Rubies Bin Directory 时，请指定二进制目录。例如，如果您使用的是 Ruby Version Manager (RVM)，请输入 /usr/local/rvm/rubies/ruby-2.0.0-p247/bin。
4. 出现提示 Auto Detect Ruby Applications Flag 时，请输入 Y 以继续。代理程序将接收代理程序数据收集器所发送的数据。
5. 出现提示 Application Server Process name 时，请按 Enter 键以接受缺省值 ruby，或根据以下列表指定所用服务器的值:
   - 对于 WEBrick 服务器，请接受缺省值或指定 ruby；如果 Rails 是由 Ruby Stack 安装，请指定 .ruby.bin。
   - 对于 Passenger 服务器，请指定 passenger。
   - 对于 Unicorn 服务器，请指定 unicorn。
   - 对于 Puma 服务器，请指定 puma。
   - 对于 Thin 服务器，如果通过运行命令 thin start 来启动应用程序，请接受缺省值以使用 ruby；如果通过运行命令 thin start -d 来启动应用程序，请指定 thin；如果 Rails 是由 Ruby Stack 安装，并通过运行命令 thin start 来启动应用程序，请指定 .ruby.bin。
6. 出现提示 Socket Data Source 时，请按 Enter 键接受缺省值 0，以使用临时端口。
7. 出现提示 Edit 'Application' settings 时，请输入 5 以退出设置。
8. 要启动代理程序，请运行以下命令:
   install_dir/bin/ruby-agent.sh start *instance_name*

下一步做什么

安装数据收集器以使 Ruby 代理程序正常工作，并使数据显示在 Cloud APM UI 中。有关指示信息，请参阅 安装数据收集器

安装数据收集器

必须安装数据收集器才能使代理程序正常工作。在您安装数据收集器之后，监视数据将显示在“应用程序性能仪表板”上。

开始之前

如果使用非 root 帐户在 Linux 系统上安装了 Ruby on Rails 应用程序，并且计划收集诊断数据，那么该非 root 用户必须具有对诊断数据收集器主目录的访问权。请验证该非 root 用户在 *install_dir*/install-images/kkm 目录中具有读写访问权，其中 *install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 /opt/ibm/apm/agent。如有需要，请使用 chmod 777 命令来提供读写许可权。

过程

1. 停止 Ruby on Rails 应用程序。
2. 可选： 如果您正在将 Ruby 数据收集器升级到新版本，那么必须通过运行以下命令先卸载旧版本的数据收集器：

   ```
   gem uninstall stacktracer
   ```

3. 安装诊断数据收集器。输入 gem install --local *install_dir*/lx8266/km/bin/stacktracer-*version*.gem，其中，*version* 是版本号，*install_dir* 是 Ruby 代理程序的安装目录。在代理程序安装目录中，stacktracer-*version*.gem 文件名中的版本号指出您需要在此处输入的版本号。缺省安装目录为 /opt/ibm/apm/agent。

   要点: 以安装和运行 Ruby on Rails 应用程序时所使用的用户身份安装数据收集器。
4. 浏览到应用程序的主目录，打开其 Gemfile，并将下列各行添加到文件的末尾：gem 'stacktracer', '*version*'
   其中，*version* 是数据收集器的版本号。此版本号在 Ruby 代理程序的 *install_dir* 中 stacktracer-*version*.gem 文件的名称中指出。
   例如，如果您安装的是 Ruby 数据收集器 V1.0 FP8，您可在代理程序的安装目录中找到 stacktracer-01.00.08.00.gem 文件。然后，您可将 gem 'stacktracer', '01.00.08.00' 行添加到应用程序，以安装该数据收集器。

   注: 如果环境中只有一个版本的 stacktracer，那么将行 gem 'stacktracer' 添加到文件尾部。请勿在行中指定版本号。
5. 在应用程序的主目录中，输入 bundle install。
6. 重新启动 Ruby on Rails 应用程序。

结果

安装和配置了数据收集器，并启动了 Ruby on Rails 应用程序。

下一步做什么

- 如果您尚未登录，请按第 829 页的『启动 Cloud APM 控制台』中的指示信息执行操作。选择 **性能** > **应用程序性能仪表板**以打开**我的所有应用程序**仪表板，并向下钻取到 Ruby 应用程序资源监视仪表板和诊断仪表板，以观察 Ruby on Rails 应用程序从状态摘要到单个请求实例的各方面。
- 要查看和修改诊断数据收集器的设置，请继续下一个主题：第 615 页的『配置诊断数据收集器』。
- 要在 Cloud APM UI 中显示请求的方法跟踪数据，请参阅启用方法跟踪和调整路径显示。
- 当已启用方法跟踪，或数据请求较大时，您可能会收到内存不足错误。为了避免发生这些错误，您可以增加 JVM 堆大小。请参阅增加 JVM 堆大小。
- 您可以随时通过 Cloud APM 控制台禁用和启用一个或多个受管 Ruby on Rails 应用程序的诊断数据收集。请参阅第 618 页的『为 Ruby 应用程序禁用或启用诊断数据』。此功能不可用于资源监视。

配置诊断数据收集器

如果您是 Cloud APM, Advanced 用户，那么您可以继续配置数据收集器以收集诊断数据。缺省情况下，在数据收集器配置文件中已禁用诊断数据收集。

开始之前

您必须安装了诊断数据收集器，并配置了对诊断数据收集的支持，如第 614 页的『安装数据收集器』中所述。

关于此任务

在代理程序通过正确配置 Gemfile 注册 Ruby on Rails 应用程序的实例后，会出现 `instrumenter_settings.rb` 配置文件。Ruby 代理程序正在运行时可以修改此文件，且会自动获取更改。或者，可将更改应用到受监视的所有 Ruby on Rails 应用程序，这需要在编辑配置文件时停止这些应用程序。

过程

- 要修改正在运行的特定应用程序的数据收集器设置：

1. 浏览到 *install_dir*`/install-images/kkm/dchome/`*appClassName*`/config` 目录，其中，*appClassName* 是 Ruby 应用程序类名，而 *install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
2. 在文本编辑器中打开 `instrumenter_settings.rb`。
3. 修改数据收集器设置：

:instrumentation_enabled

要启用对诊断数据收集的支持，请设置 `:instrumentation_enabled => true`。

要禁用对诊断数据收集的支持，请设置 `:instrumentation_enabled => false`。

:sample_frequency

要修改请求的采样频率，请输入两次采样之间的请求数。

数据收集器仅为采样的请求收集诊断数据。例如，如果设置 `:sample_frequency => 10`，那么将针对每 10 个请求中的 1 个请求收集数据。

:max_methods_to_instrument

要禁用方法跟踪收集，或要启用方法跟踪收集并限制跟踪的方法数，请将值设置为零或输入要跟踪的最大方法数。

要禁用方法跟踪收集，请设置 `:max_methods_to_instrument => 0`。

要启用方法跟踪收集，请设置 `:max_methods_to_instrument => 10000`。值可以更大，但是大很多的值可能导致性能问题。收集方法数据时，对方法的调用将包含在“请求跟踪”仪表板的“方法跟踪”窗口小部件中，此窗口小部件显示所有请求实例及其嵌套请求。

:min_wallclock_to_include_in_trace

要修改用于确定应跟踪请求还是跟踪方法的阈值，请设置最小响应时间。例如，如果设置 `:min_wallclock_to_include_in_trace => 0.001`，那么仅跟踪其响应时间长于 1 毫秒的请求和方法。

切记: 在“**请求跟踪**”诊断仪表板中，可从“请求堆栈跟踪”组窗口小部件向下钻取到特定请求实例。为 `:min_wallclock_to_include_in_trace` 和 `:min_wallclock_to_include_stacks` 设置的过滤器（可能会排除一些数据）导致实例的总响应时间可能不正确。

:min_wallclock_to_include_stacks

要修改用于确定应收集请求还是方法的堆栈跟踪信息的阈值，请设置最小响应时间。

例如，如果设置 `:min_wallclock_to_include_stacks => 0.1`，那么会为其响应时间长于 100 毫秒的所有请求和方法收集堆栈跟踪信息。

:include_subclasses_of_these_modules
"请求跟踪"诊断仪表板帮助您识别对请求实例的嵌套请求和方法的一系列调用。数据收集器会抢先从类中过滤掉未包括在过滤器列表内的方法。如果要跟踪的操作未包括在方法堆栈跟踪中，可以在此处添加这些操作。

要指定需跟踪的方法，请添加其类名。

例如，假设您要在以下类型的 Ruby 代码中跟踪 Moped API：

```
session = Moped::Session.new(['ip:27017'])
session.use(:HR)
session[:profiles].insert({......})
session[:profiles].find({...}).remove
```

将这些 Moped API 的模块名称添加到以下属性：

```
:include_subclasses_of_these_modules => {"
ActionController" => true,
                      "ERB" => true,
                      "erb" => true,
                      "Arel" => true,
                      "Mongoid" => true,
                      "Moped" => true
                     },
```

限制: 方法跟踪不包含类方法和私有方法（类中定义的具有隐式或显式"private"访问说明符的方法）。

:include_sql_text
要收集方法的上下文数据，请将此属性设置为 `true`。

:num_samples_per_file
要修改将在每个文件中存储的最大已跟踪请求数，请输入诸如 `:num_samples_per_file => 1000` 之类的值。达到此处设置的限制后，会创建一个新文件。

如果要调整 `:num_samples_per_file` 配置以收集更多数据，请将其设置为较小的值。例如，将 `:include_subclasses_of_these_modules` 设置为跟踪更多类和方法可以增加数据收集量。此外，将以下任意属性设置为较小值也可以增加数据收集量：`:sample_frequency`、`:min_wallclock_to_include_in_trace` 和 `:min_wallclock_to_include_stacks`。

:verbose_request_instrumentation
:verbose_class_instrumentation
:verbose_method_instrumentation
要增大诊断数据收集器的日志记录级别，请将这些属性设置为 `true`。

提示: 如果明确需要跟踪的操作未包括在方法堆栈跟踪中，请设置 `:verbose_class_instrumentation => true`，并检查日志以确定是否已检测希望跟踪的类。如果未检测，那么将该类的模块名称的类名添加到 `:include_subclasses_of_these_modules` 属性。

4. 如果已编辑以下任一属性，那么重新启动相应 Ruby on Rails 应用程序以使更改生效：

```
:include_subclasses_of_these_modules
:max_methods_to_instrument
```

由于仅在启动应用程序以确定 Ruby 数据收集器将检测的类或方法时才会使用这些属性，因此必须执行重新启动。

- 要修改所有 Ruby on Rails 应用程序的数据收集器设置，请完成以下步骤：

1. 停止当前正在运行的任何 Ruby on Rails 应用程序。
2. 从 *install_dir*/install-images/kkm/dchome/*application_name*/config 目录移除 instrumer_settings.rb。
3. 修改 *Gem_dir*/gems/stacktracer-*version*/config/instrumenter_settings_template.rb 中的数据收集器设置，其中，*version* 是版本号（例如，

01.00.05.00），而 *Gem_dir* 是 `stacktracer-version.gem` 的安装目录（例如，`/usr/local/rvm/gems/ruby-2.1.4/`）。有关更多信息，请参阅该过程中用于修改特定应用程序的数据收集器设置的步骤 第 615 页的『3』。

4. 重新启动当前正在运行的任何 Ruby on Rails 应用程序。

结果

针对您所指定的正在运行的应用程序或针对所有应用程序，诊断数据收集器的配置已更改。

启用方法跟踪并调整路径显示

具有诊断数据的 IBM Cloud Application Performance Management, Advanced 使用户能够使用“**请求跟踪**”仪表板。如果收集方法数据，那么会显示对方法的调用。“**方法轨迹**”窗口小部件显示请求实例及其嵌套的请求。您可以启用方法跟踪将对方法的调用包含在嵌套请求中。您还可以调整“**请求堆栈跟踪**”窗口小部件的配置，以针对每个文件路径显示超过 50 个（缺省值）字符。

关于此任务

缺省情况下，方法跟踪处于禁用状态。完成第一个过程来启用方法跟踪以在“**请求跟踪**”仪表板中显示。您可以通过更改配置文件中的设置以启用方法跟踪。

完成第二个过程以调整“**请求堆栈跟踪**”窗口小部件中为文件路径显示的字符数。

过程

- 要启用方法跟踪，请编辑 `instrumenter_settings.rb` 设置：
  a) 在 Ruby 代理程序安装中找到 `instrumenter_settings.rb` 文件，例如，*`install_dir`*`/install-images/kkm/dchome/`*`appClassName`*`/config`，其中，*`appClassName`* 是 Ruby 应用程序类名，而 *install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
  b) 在文本编辑器中打开 `instrumenter_settings.rb`。
  c) 将以下属性设置为 10000。

  ```
  max_method_to_instrument
  ```

  值可以更大，但是大很多的值可能导致性能问题。（另请参阅第 617 页的『增大 JVM 堆大小』。）
  d) 重新启动 Ruby on Rails 应用程序以开始方法数据收集。
  有关所有 `instrumenter_settings.rb` 属性的更多信息，请参阅第 615 页的『配置诊断数据收集器』。
- 要在“**请求堆栈跟踪**”窗口小部件中调整文件路径显示大小，请编辑 `dfe.properties` 文件：
  a) 在 Ruby 代理程序安装中找到 `dfe.properties` 文件，例如，*`install_dir`*`/lx8266/km/bin/dfe.properties`
  其中，*install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
  b) 在文本编辑器中打开 `dfe.properties`。
  c) 通过调整以下属性的值，更改要在每个堆栈跟踪元素中显示的文件路径的最大大小：

  ```
  dfe.stacktrace.filepath.maxsize
  ```

  d) 重新启动 Ruby 代理程序。

增大 JVM 堆大小

当为 Ruby 诊断“**请求跟踪**”仪表板启用方法跟踪时，或数据请求很大时，可以增大 JVM 堆大小以避免出现内存不足错误。

关于此任务

Ruby 代理程序 是基于 Java 的代理程序，并且缺省 JVM 堆大小为 384 MB。执行这些步骤以增加堆大小，从而降低发生内存不足情况的可能性。在启用方法跟踪时，对于频繁的大数据请求，可能发生内存用尽情况。

过程

1. 查找 Ruby 代理程序安装目录中的 JVM 堆大小设置，例如，*install_dir*/lx8266/km/bin/runDeepDiveClient.sh
   其中，*install_dir* 是 Ruby 代理程序的安装目录。缺省安装目录为 /opt/ibm/apm/agent。
   缺省值为 -Xmx384m。
2. 增大值，例如，1024 MB，显示为 -Xmx1024m:
   ```
   export JAVA_OPT="-Djlog.common.dir=$CANDLEHOME/logs -DCONFIG_DIR=
   $DC_RUNTIME_DIR -Dkqe.cache.interval=60 -Xmx1024m -Dkqe.timespan=900 -
   Djlog.propertyFileDir.CYN=$CANDLEHOME/$ITM_BINARCH/$PRODUCT_CODE/bin"
   ```
3. 重新启动 Ruby 代理程序。

为 Ruby 应用程序禁用或启用诊断数据

如果您有 IBM Cloud Application Performance Management, Advanced，可在 Cloud APM 控制台中使用“**代理程序配置**”页面来随时为一个或多个受管系统禁用或启用诊断数据收集。

开始之前

· 必须在环境中安装 Cloud APM, Advanced。

· 您必须在虚拟机中安装和配置 Monitoring Agent for Ruby，如第 105 页的『安装代理程序』（在 AIX 系统上）或第 112 页的『安装代理程序』（在 Linux 系统上）以及第 612 页的『配置 Ruby 监视』中所述。

· 您必须安装诊断数据收集器并配置对诊断收集的支持，如第 614 页的『安装数据收集器』中所述。

关于此任务

在数据收集器配置中配置对诊断数据的支持后，缺省情况下会为每个受管系统禁用诊断数据收集。要在诊断仪表板中显示数据，必须为您所监视的每个受管系统启用诊断数据收集。

执行这些步骤来为每个受管系统启用和禁用诊断数据收集:

过程

1. 从导航栏，选择 **系统配置 -> 代理程序配置**。
   “**代理程序配置**”页面随即显示。
2. 单击 **Ruby** 选项卡。
3. 选中要对其禁用或启用诊断数据收集的受管系统的复选框。
4. 从**操作**列表，选择以下其中一个选项来为所选受管系统禁用或启用诊断数据收集:
   - 选择**禁用数据收集**。对于每个所选受管系统，“启用数据收集器”列中的状态会更新为“否”。
   - 选择**启用数据收集**。对于每个所选受管系统，“启用数据收集器”列中的状态会更新为“是”。

结果

您已为每个所选受管系统配置了诊断数据收集。

针对 IBM Cloud 应用程序配置 Ruby 数据收集器

要收集 IBM Cloud 上的 Ruby 应用程序的相关信息，必须配置 Ruby 数据收集器。

开始之前

1. 从 IBM Marketplace 下载数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。

过程

1. 从数据收集器软件包中解压缩文件。已抽取的目录中包含 ruby_datacollector_8.1.4.0.tgz 软件包。

2. 通过运行以下命令，将 ruby_datacollector_8.1.4.0.tgz 中的文件解压缩：

```
tar -zxf ruby_datacollector_8.1.4.0.tgz
```

这将创建 ibm_ruby_dc 文件夹。

3. 通过运行以下命令，将 ibm_ruby_dc 中的整个 etc 文件夹复制到 Ruby 应用程序的根文件夹：

```
cp -r directory to the etc folder  home directory of your Ruby application
```

以下命令将数据收集器解压缩到 /opt/ibm/ccm/ibm_ruby_dc/etc 目录，Ruby 应用程序的主目录是 /root/ruby_app/：

```
cp -r /opt/ibm/ccm/ibm_ruby_dc/etc /root/ruby_app/
```

4. 在 Ruby 应用程序主文件夹中的 Gemfile 中，添加以下小节：

```
gem 'logger',  '>= 1.2.8'
source 'https://maagemserver.ng.bluemix.net/' do
  gem 'ibm_resource_monitor'
  gem 'stacktracer'
end
```

5. 运行 bundle lock 命令，以重新生成 Gemfile.lock 文件。
6. 从 Ruby 应用程序 manifest.yml 文件所在的目录中，运行以下命令：

```
cf push
```

提示: 要获取样本 manifest.yml 文件，请参阅第 160 页的『样本 manifest.yml 文件』。

结果

数据收集器现已配置完成，并已连接到 Cloud APM 服务器。

下一步做什么

您可以验证 IBM Cloud 应用程序的监视数据是否显示在 Cloud APM 控制台中。有关如何启动 Cloud APM 控制台的指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

针对 IBM Cloud 应用程序定制 Ruby 数据收集器

您可以在 IBM Cloud 用户界面 (UI) 中添加环境变量，以定制针对 IBM Cloud 应用程序进行的监视。请根据您的需要，使用以下信息来添加变量。

Ruby 数据收集器的用户定义环境变量

您可使用下表中的信息来定制针对 IBM Cloud 的 Ruby 监视。

表 *198.* 针对 *IBM Cloud* 的 *Ruby* 监视所支持的用户定义环境变量

变量名称	重要性	值	描述
APM_BM_GATEWAY_URL	可选	· https://*<server ip or hostname>*:443 · http://*<server ip or hostname>*:80	目标内部部署服务器网关 URL。

表 198. 针对 *IBM Cloud* 的 *Ruby* 监视所支持的用户定义环境变量 *(续)*

变量名称	重要性	值	描述
APM_KEYFILE_PSWD	可选	密钥文件的已加密密码	与密钥文件配对的已加密密钥文件密码。如果您是 Linux 用户，那么可以使用 `echo -n <keyfile password> | base64` 命令对密码进行加密。 **注:** 只有在已将网关配置为使用 HTTPS 的情况下，才应设置此变量。
APM_KEYFILE_URL	可选	http://*<hosted http server>*:*<port>*/keyfile.p12	用来下载密钥文件的 URL。 **注:** 只有在将网关配置为使用 HTTPS 的情况下，才应设置此变量。
kkm_instrumentation_enabled	可选	· true · false	启用或禁用诊断数据收集。 true：将值设置为 `true` 表示收集诊断数据。 false：将值设置为 `false` 表示不收集诊断数据。 缺省值为 `true`。
kkm_max_methods_to_instrument	可选	跟踪的最大方法数	跟踪的最大方法数。 可以将值设置为 `0` 以禁用方法跟踪。 缺省情况下，值为 `10000`，即启用方法跟踪。 **注:** 建议您不要将值设置为大于 10000。远大于 10000 的值可能会导致应用程序执行效率下降。
kkm_sample_frequency	可选	请求采样率	建立一个采样请求所需的请求数，例如，如果将此值设置为 `10`，那么对于每 10 个请求，将收集一个请求的监视数据。 缺省值为 `10`。
kkm_min_wallclock_to_include_in_trace	可选	收集方法跟踪时采用的响应时间阈值（秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其方法跟踪。例如，如果将其设置为 0.001，那么将跟踪响应时间超过 1 毫秒的请求和方法。 缺省值为 `0`，这表示对所有请求和方法启用方法跟踪。

表 198. 针对 *IBM Cloud* 的 *Ruby* 监视所支持的用户定义环境变量 *(续)*			
变量名称	**重要性**	**值**	**描述**
kkm_min_wallclock_to_include_stacks	可选	收集堆栈跟踪时采用的响应时间阈值（秒）	如果请求实例的响应时间超过此变量的值，那么数据收集器将收集其堆栈跟踪。例如，如果将其设置为 0.001，那么将跟踪响应时间超过 1 毫秒的请求和方法。 缺省值为 `0`，这表示对所有请求和方法启用堆栈跟踪。

取消为 IBM Cloud 应用程序配置 Ruby 数据收集器
如果您不需要监视 Ruby 环境或者想要升级 Ruby 数据收集器，那么必须首先取消配置 Ruby 数据收集器的先前设置。

过程

1. 转到应用程序根文件夹。
2. 在 Ruby 应用程序的主文件夹中的 `Gemfile` 中，移除以下行：

```
gem 'logger',  '>= 1.2.8'
source 'https://maagemserver.ng.bluemix.net/' do
  gem 'ibm_resource_monitor'
  gem 'stacktracer'
end
```

3. 运行 `bundle lock` 命令。
4. 从应用程序主目录，运行以下命令以将应用程序重新推送到 IBM Cloud，以使更改生效。

```
cf push
```

结果
您已成功取消配置 Ruby 数据收集器。

下一步做什么
在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

配置 SAP 监视

要监视 SAP 系统，Monitoring Agent for SAP Applications 必须连接到要监视的系统中的应用程序服务器，以便代理程序可以访问随产品一起提供的 Advanced Business Application Programming (ABAP) 代码。

开始之前

· 查看必备硬件和软件，请参阅 SAP 代理程序的软件产品兼容性报告
· SAP 代理程序不支持非 Unicode SAP 系统。

关于此任务

SAP 代理程序是一个多实例代理程序；您必须手动创建第一个实例并启动代理程序。

· 要在 Windows 系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。
  – 第 622 页的『在 Windows 系统上配置代理程序』

– 第 624 页的『使用静默响应文件来配置代理程序』

· 要在 Linux 或 AIX 系统上配置代理程序，您可以运行脚本并对提示做出响应，或者使用静默响应文件。

– 第 623 页的『在 Linux 或 AIX 系统上配置代理程序』
– 第 624 页的『使用静默响应文件来配置代理程序』

在安装 SAP 代理程序后，您可以导入 SAP 系统上的 Advanced Business Application Programming (ABAP) 传输以支持 SAP 系统中的数据收集。有关更多信息，请参阅第 627 页的『在 SAP 系统上导入 ABAP 传输』。

在配置 SAP 代理程序后，必须验证代理程序配置。有关更多信息，请参阅第 633 页的『验证代理程序配置』。

配置 SAP 代理程序后，可以添加 OSLC（开放式源代码生命周期协作）合规性所需的数据库通信端口号。有关更多信息，请参阅第 636 页的『添加数据库通信端口号』。

要从 SAP 系统删除 ABAP 传输，必须将 delete 传输导入 SAP 系统。有关更多信息，请参阅第 632 页的『从 SAP 系统删除 ABAP 传输』。

缺省情况下，启用新的 CCMS 设计。条目出现在参数值设置为 YES. 的 isnewccmsdesign 参数的数据库表 / IBMMON/ITM_CNGF 中。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

在 Windows 系统上配置代理程序

使用 **IBM Performance Management** 窗口在 Windows 系统上配置 SAP 代理程序，以便代理程序可以收集受监视的 SAP Applications Server 的数据。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口的**任务/子系统**列下，右键单击**模板**，然后单击**使用缺省值配置**。

   这将打开 **Monitoring Agent for SAP Applications** 窗口。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。

   要点: 代理程序实例名称必须与受管 SAP Applications Server 的 3 位系统标识 (SID) 相匹配。例如，如果受管 SAP Applications Server 的 SID 为 PS1，请输入 PS1 作为实例名称。
4. 以“应用程序服务器”方式或“登录组”方式配置 SAP 代理程序。

   · 以“应用程序服务器”方式配置 SAP 代理程序，请完成下列步骤：

   a. 在**连接方式**字段中，选择**应用程序服务器方式**，然后单击**下一步**。
   b. 在**指定应用程序服务器信息**区域中，指定配置参数的值，然后单击**下一步**。
   c. 在**指定 SAP 系统的登录信息**区域中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 624 页的『代理程序的配置参数』

   · 以“登录组”方式配置 SAP 代理程序，请完成下列步骤：

   a. 在**连接方式**字段中，选择**登录组方式**，然后单击**下一步**。
   b. 在**指定登录组信息**区域中，指定配置参数的值，然后单击**下一步**。
   c. 在**指定 SAP 系统的登录信息**区域中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 624 页的『代理程序的配置参数』

   要点: 对于应用程序服务器模式，需要配置在配置了消息服务器或 ASCS 的 SAP 系统上具有分派器的对话框实例。对于登录组模式，无需配置在配置了消息服务器或 ASCS 的 SAP 系统上具有分派器的对话框实例。
5. 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

要点: 如果要创建 SAP 代理程序 的另一个实例，请重复步骤 1 到 6。对您要创建的每个 SAP 代理程序实例使用唯一的系统标识。

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
- 您必须编辑预定义的 R3_Alert_Crit 情境和 R3_Alert_Warn 情境，以将“警报状态”属性的条件设置为 `Alert Status!= DONE`，这样对于已关闭的 CCMS 警报就不会触发这些情境。

在 Linux 或 AIX 系统上配置代理程序

您可以在 Linux 或 AIX 系统上配置 SAP 代理程序，以便代理程序可以收集受监视的 SAP Applications Server 的数据。

过程

1. 在命令行中，更改代理程序安装目录的路径。
   示例：`/opt/ibm/apm/agent/bin`
2. 运行以下命令，其中 instance_name 是要对实例指定的名称：
   `./sap-agent.sh config instance_name`

   要点: 代理程序实例名称必须与受管 SAP Applications Server 的 3 位系统标识 (SID) 相匹配。例如，如果受管 SAP Applications Server 的 SID 为 PS1，请输入 PS1 作为实例名称。
3. 命令行显示以下消息时，请输入 1，然后按 Enter 键：`Edit 'Monitoring Agent for SAP Applications' setting? [1=Yes, 2=No]`
4. 在“应用程序服务器”方式下，或“登录组”方式配置 SAP 代理程序。
   - 以“应用程序服务器”方式配置 SAP 代理程序，请完成下列步骤：
     a. 命令行显示以下消息时，请输入 1，然后按 Enter 键：`Connection Mode [ 1=Application Server Mode, 2=Logon Group Mode ]`
     b. 为配置参数指定值。

     有关配置参数的信息，请参阅第 624 页的『代理程序的配置参数』
   - 以“登录组”方式配置 SAP 代理程序，请完成下列步骤：
     a. 命令行显示以下消息时，请输入 2，然后按 Enter 键：`Connection Mode [ 1=Application Server Mode, 2=Logon Group Mode ]`
     b. 为配置参数指定值。

     有关配置参数的信息，请参阅第 624 页的『代理程序的配置参数』

   要点: 对于应用程序服务器模式，需要配置在配置了消息服务器或 ASCS 的 SAP 系统上具有分派器的对话框实例。对于登录组模式，无需配置在配置了消息服务器或 ASCS 的 SAP 系统上具有分派器的对话框实例。
5. 运行以下命令以启动 SAP 代理程序 代理程序：
   `./sap-agent.sh start instance_name`

   要点: 如果要创建 SAP 代理程序 的另一个实例，请重复步骤 1 到 5。对您创建的每个 SAP 代理程序实例使用唯一的系统标识。

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
- 您必须编辑预定义的 R3_Alert_Crit 情境和 R3_Alert_Warn 情境，以将“警报状态”属性的条件设置为 `Alert Status!= DONE`，这样对于已关闭的 CCMS 警报就不会触发这些情境。

使用静默响应文件来配置代理程序

通过使用静默响应文件，您可以在 Windows、Linux 或 AIX 系统上配置 SAP 代理程序。

过程

1. 在文本编辑器中，打开 *install_dir*\samples 路径下的 sap_silent_config.txt 文件，并指定所有配置参数的值。

   Windows C:\IBM\APM\samples

   Linux AIX /opt/ibm/apm/agent/samples

   有关配置参数的信息，请参阅第 624 页的『代理程序的配置参数』
2. 在命令行中，将路径更改为 bin 目录。

   Windows *install_dir*\BIN

   Linux AIX *install_dir*\bin
3. 运行以下命令：

   Windows sap-agent.bat config *instance_name install_dir*\samples\sap_silent_config.txt

   Linux AIX sap-agent.sh config *instance_name install_dir*\samples\sap_silent_config.txt

   要点: 代理程序实例名称必须与受管 SAP Applications Server 的 3 位系统标识 (SID) 相匹配。例如，如果受管 SAP Applications Server 的 SID 为 PS1，请输入 PS1 作为实例名称。
4. 启动代理程序。

   Windows 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

   Linux AIX 运行以下命令：./sap-agent.sh start *instance_name*

   要点: 如果要创建 SAP 代理程序 的另一个实例，请重复步骤 1 到 4。对您创建的每个 SAP 代理程序实例使用唯一的系统标识。

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
- 您必须编辑预定义的 R3_Alert_Crit 情境和 R3_Alert_Warn 情境，以将“警报状态”属性的条件设置为 Alert Status!= DONE，这样对于已关闭的 CCMS 警报就不会触发这些情境。

代理程序的配置参数

配置 SAP 代理程序 时，您可以更改参数的缺省值，例如 SAP 主机名和 SAP 系统编号。

下表包含 SAP 代理程序 的配置参数的详细描述。

表 *199. SAP* 代理程序 的配置参数的名称和描述

参数名称	描述	必填字段	示例
SAP 主机名（主要）	此代理程序连接到的 SAP 应用程序服务器的主机名。如果 SAP 服务器通过专用 LAN 进行通信，那么托管这些服务器的计算机具有两张或更多网卡。对于主机名，输入可用于从外部系统到达应用程序服务器的名称，例如 SAPGUI 登录。请勿使用专用 LAN 主机名。缺省值是安装代理程序的主机的名称。	是	saphost.domain.com

表 *199. SAP* 代理程序 的配置参数的名称和描述 *(续)*

参数名称	描述	必填字段	示例
SAP 系统编号（主要）	用于连接到 SAP 主机服务器的 2 位数的 SAP 系统或实例编号。缺省值为 00。	是	
SAP 主机名（备用 1）	主要主机不可用时主机名的第二选择。	否	
SAP 系统编号（备用 1）	第一个备用主机名的系统编号。	否	
SAP 主机名（备用 2）	SAP 主机名（主要）和 SAP 主机名（备用 1）主机不可用时主机名的第三选择。	否	
SAP 系统编号（备用 2）	第二个备用主机名的系统编号。	否	
SAP 客户机编号	用于 SAP 的 RFC 登录的 SAP 客户机编号。缺省值为 000。如果使用了 ABAP 生成的 IBMMON_AGENT 用户，那么请输入在导入传输时指定的客户机编号。此编号与概要文件下的 nnn 客户机编号相同。	是	
SAP 用户标识	用于 SAP 的 RFC 登录的 SAP 用户标识。缺省值是导入期间创建的预定义用户标识 IBMMON_AGENT。	是	
SAP 用户密码	使用缺省密码或指定另一密码。	是	
确认 SAP 用户密码	**SAP 用户密码**字段中指定的密码。	是	
SAP 语言代码	指示代理程序连接到 SAP 系统时使用的语言的语言代码。指定的语言确定了查看 SAP 信息的语言，例如警报消息、系统日志消息和作业日志消息。 所有 SAP 系统均以英语和德语交付。如果需要其他语言，请与 SAP 管理员确认已在 SAP 系统上安装此语言。如果指定的语言不受支持，那么代理程序将无法连接到 SAP 系统。 支持以下语言和代码： · CS - 捷克语 · EN - 英语 · FR - 法语 · DE - 德语 · HU - 匈牙利语 · IT - 意大利语 · ES - 西班牙语 · JA - 日语 · KO - 韩国语 · PL - 波兰语 · PT - 葡萄牙语 · RU - 俄语 · ZH - 中文 · ZF - 繁体中文	是	

表 *199. SAP* 代理程序 的配置参数的名称和描述 *(续)*

参数名称	描述	必填字段	示例
RFC 跟踪	*SAPTRACE* 变量的远程函数调用 (RFC) 跟踪设置。如果选中此复选框，则激活 RFC 跟踪，而缺省值是无 RFC 跟踪。对于命令行，2 = 无跟踪，而 1 = 跟踪。因为 RFC 跟踪会生成详细的诊断信息，请谨慎使用。有关 RFC 跟踪的更多信息，请联系 IBM 支持人员。	否	
SAP 登录组	SAP 服务器登录组的名称。	是	
SAP 消息服务器名称	SAP 消息服务器的主机名。	是	
SAP 消息服务	SAP 消息服务器所在的服务的编号。 必须在以下操作系统服务文件中包含服务名称: · `/etc/services` · `\windows\system32\drivers\etc\services`	是	您可能会使用消息服务名称 sapmsTV1 或完整消息服务端口号 3601。
SAP 路由字符串	指定要使用 SAP 路由器访问 SAP 服务器时的 SAP 路由器字符串。	否	路由器字符串 `/H/host/H/` 必须使用以下格式：`/H/beagle/H/brittany/H/` 或 `/H/amsaix11.tivlab.raleigh.ibm.com/W/tivoli/H/amsaix25`
SNC	指定是要启用还是禁用安全网络通信 (SNC)。缺省值为禁用。	是	**sap_conn.sap_snc_mode** =`true` 或 `false`
SNC 安全级别	SNC 的安全级别。	是	**sap_snc_mode1.sap_snc_qop**=*QOP value*。 缺省值为 8。
客户机或代理程序 SNC 名称	客户机或代理程序的 SNC 名称。	是	**sap_snc_mode1.sap_snc_client**= *Client SNC Name*
合作伙伴或 SAP 服务器 SNC 名称	合作伙伴或 SAP 服务器的 SNC 名称。	是	**sap_snc_mode1.sap_snc_server**= *Server SNC Name*
SAP 加密库路径	SAP 加密库的路径。	是	**sap_snc_mode1.sap_snc_library**= *Crypto library path*

根据“Managed System Name”长度限制对 SAP 主机名进行剪裁

APM 控制台上发布的任意资源的 Managed System Name 长度均限制为 32 个字符。SAP 代理程序支持剪裁域名以形成符合此限制的 Managed System Name。

场景 1

类型为 **Sys** 的子节点的 Managed System Name 格式如下：

SID-DBHOST:**Sys**
其中：

· *SID* 是 SAP 系统标识。

· *DBHOST* 是 SAP 系统主机名。

示例：
鉴于 *SID* 为 **P27** 并且 *DBHOST* 为 **VPT02F90.mycorporation.co.in**，形成的 Managed System Name 的标准域名 (FQDN) 将为 **P27-VPT02F90.mycorporation.co.in:Sys**。

当 Managed System Name 长度超过 32 个字符时，SAP 代理程序 会对域名进行剪裁以形成 Managed System Name **P27-VPT02F90:Sys**。剪裁后的子节点 Managed System Name 将发布在 APM 控制台上。

注: 如果包含域名的 Managed System Name 长度小于或等于 32 个字符，那么将不会对 Managed System Name 的 FQDN 进行剪裁。Managed System Name 的 FQDN 将相应地发布在 APM 控制台上。为满足 Managed System Name 长度限制所需的域名剪裁适用于 SAP 代理程序发布的所有子节点类型。

场景 2

代理实例子节点 **mySAP** 的 Managed System Name 格式如下：

$SAPSYSTEMNAME-$dbhost:*$CTIRA_HOSTNAME*:**mySAP**
其中：

· *$SAPSYSTEMNAME* 是配置期间提供的代理程序实例名称。

· *$dbhost* 是 SAP 系统主机名。

· *$CTIRA_HOSTNAME* 是代理程序机器主机名。

示例：
鉴于 *$SAPSYSTEMNAME* 为 **SA2**，*$dbhost* 为 **VPT02F90. mycorporation.co**，并且 *$CTIRA_HOSTNAME* 为 **mysap1-v27.mycorp.co**，形成的 Managed System Name 的标准域名 (FQDN) 将为 **SA2-VPT02F90.mycorporation.co.in: mysap1-v27:mySAP**。

注: 代理机器主机名的缺省域名将被剪裁为 **mysap1-v27**。

Managed System Name 包含超过 32 个字符。首先，SAP 代理程序会对 SAP 系统主机名的域名进行剪裁以形成 **SA2-VPT02F90:mysap1-v27:mySAP**。
如果生成的 Managed System Name 仍超出 32 个字符，SAP 代理程序将对代理机器主机名的尾部字符进行剪裁，以使形成的 Managed SystemName 符合 32 个字符的限制。然后，子节点的 Managed System Name 将发布在 APM 控制台上。

在 SAP 系统上导入 ABAP 传输

您可以为每个 SAP 系统安装一个 SAP 代理程序，其中导入 Advanced Business Application Programming (ABAP) 传输请求以支持 SAP 系统中的数据收集。

开始之前

在 SAP 系统上导入 ABAP 传输之前，请确保满足以下先决条件：

· 要导入产品传输请求，需要 R3trans V01.07.04 或更新版本，因为 Dynpro 与“导出和导入”表不兼容。Dynpro 与“导入和导出”的不兼容问题不会影响代理程序的基本操作；只会影响 SAP 配置窗口。

- 在可使用 MAI 配置来监视 Solution Manager 系统的客户机上，必须确保导入 SAP 代理程序 V7.1.1 传输。要查看 PI 系统的功能部件，请在 PI 配置可用的 PI 系统客户机上导入 SAP 代理程序 V7.1.1 传输。
- 要查看 SLM 子节点下面的组窗口小部件中的数据，必须针对 PI 和 Solution Manager 完成 MAI 配置。您还必须配置业务流程监视，从而可以查看“BPM 警报”组窗口小部件中的数据。要查看“最新临界和高优先级警报”组窗口小部件的数据，请完成以下步骤：
  - 在 Solution Manager 7.1 中，运行 SOLMAN_SETUP 事务并选择**系统监视**，激活或启用第三方组件，然后添加**实施：警报反应的 BADI 定义**和第三方连接器。
  - 将作用域过滤器设置为**所有警报和度量**。
  - 确保“实施”状态为**活动**。

  有关更多信息，请参阅以下 Online Service System (OSS) Notes，包括必需的 SAP service pack 级别列表：
  - OSS Note 454321
  - OSS Note 330267
  - OSS Note 743155
- 要监视 SAP 系统，SAP 代理程序 需要 SAP 统计数据。在 SAP 7.0 系统上，必须设置 SAP 系统时区以匹配操作系统的时区，从而使用正确的时间戳记收集 SAP 统计信息。类似，更新 SAP 代理程序 的 SAP 系统时区，从而使代理程序可收集数据。有关此问题的更多信息，请参阅 SAP Note 926290。

关于此任务

有关导入 SAP 传输的信息，请参阅第 629 页的『导入 SAP 传输』。

导入 ABAP 传输前必须符合的“MAI 警报”相关先决条件

在导入 ABAP 传输之前，您必须验证“MAI 警报”相关先决条件。

transport.prop 文件中的配置设置

当您使用包括访存“MAI 警报”的新“MAI 警报”访存机制，但未配置电子邮件通知设置，且没有 BAdi 实现时，您必须修改 transport.prop 文件中的下列配置设置。

添加 SPLEVEL=X 行，其中 X 是 Solution Manager 系统的支持包 (SP) 级别。

例如，如果“系统标识”是 S10，支持包级别是 13，请添加 SPLEVEL=13。

要点: 对于 SP 级别为 10 或更高版本的 SAP 系统，当访存“MAI 警报”，但没有在 SAP Solution Manager 中配置电子邮件通知，而且没有 BAdi 实现时，在 SAP Solution Manager Dashboard 中的“评级为‘红色’的最新 MAI 警报”组窗口小部件上，不会填充“技术名称”(MEA) 属性的值。通过在 SAP Solution Manager 和 BAdi 实现中配置电子邮件通知来访存“MAI 警报”时，在 SAP Solution Manager Dashboard 中的“评级为‘红色’的最新 MAI 警报”组窗口小部件上，会填充“技术名称”(MEA) 属性的值。

根据 Solution Manager 支持包 (SP) 级别来确定访存“MAI 警报”的新旧机制

旧的“MAI 警报”访存机制
此机制基于配置电子邮件通知设置以及带有 IF_ALERT_DYN_COFIGURATION 接口的 /IBMMON/ITM_IMPL_ALRTINBX BAdi 实现，以收集“MAI 警报”并将其发送至 SAP 代理程序。

新的“MAI 警报”访存机制
此机制基于在不配置电子邮件通知设置，而且没有带 IF_ALERT_DYN_COFIGURATION 接口的 /IBMMON/ITM_IMPL_ALRTINBX BAdi 实现的情况下，访存“MAI 警报”。

您可使用下表来了解 transport.prop 文件的用法，及其对配置电子邮件通知设置的依赖关系。

表 *200. transport.prop* 文件的用法及其依赖关系

SAP 系统 SP 级别	**transport.prop 设置**		**电子邮件通知设置的配置**	**要使用的“MAI 警报”机制**
	MAI_CONFIGURED	**Solution Manager SP 级别**		
任何	无，或此文件不存在	不适用	已配置或未配置	SLM 子节点没有出现，而是出现 SOL 子节点。
SP 6 到 9	是	已指定	已配置	旧机制
SP 6 到 9	是	未指定	已配置	旧机制
SP 6 到 9	是	未指定	未配置	旧机制无法运作，因为配置电子邮件通知设置是必要条件。
SP 6 到 9	是	已指定	未配置	旧机制无法运作，因为配置电子邮件通知设置是必要条件。
SP 10 或更高版本	是	已指定	已配置	新机制
SP 10 或更高版本	是	已指定	未配置	新机制
SP 10 或更高版本	是	未指定	已配置	旧机制
SP 10 或更高版本	是	未指定	未配置	旧机制无法运作，因为配置电子邮件通知设置是必要条件。

导入 SAP 传输

SAP 代理程序提供一组 Advanced Business Application Programming (ABAP) 例程以支持 SAP 系统中的数据收集。此 ABAP 代码作为必须在要监视的每个 SAP 系统上安装的 SAP 传输交付。SAP 管理员安装传输。

关于此任务

ZITM_610AUTH 授权概要文件和 **ZITM_610AUT** 授权角色仅在 6.1 发行版之前有效。从 R6.2 或更新版本开始，将使用 **/IBMMON/AUTH** 授权概要文件。为避免未经授权的使用，SAP 系统中安装的 ABAP 代码在 SAP 系统中不可见。此外，无法修改或生成此代码。必须从 IBM 软件支持 Web 站点获取此代码的支持。

除了安装 ABAP 代码，传输还会安装翻译的语言文本元素以针对 SAP 传输文本元素提供多元文化支持。

要点: 在将传输导入 SAP 系统之前，不得启动配置为监视 SAP 系统的 SAP 代理程序实例。

在导入 SAP 传输时，会在 SAP 系统中隐式定义用户。

使用此过程以将 SAP 传输导入到 SAP 系统中。

过程

1. 从安装代理程序的计算机上的以下路径复制 IBM Tivoli Monitoring 传输文件。
   - 对于 Windows：*install_dir*\TMAITM6_x64\ABAP
   - 对于非 Windows：*install_dir*/*intrp*/sa/ABAP，其中，*intrp* 必须是 **lx8266** 或 **aix526**。
2. 将以下传输文件从步骤 1 中提及的路径复制到 SAP 环境：
   - K711_00xxxU.ITM 和 R711_00xxxU.ITM

这些文件是传输的 Unicode 版本。它们包含拉丁语代码页和双字节代码页文本字符串的 SAP 代理程序 ABAP 代码和 Unicode 支持。

- `K711_00xxx_DELETE.ITM` 和 `R711_00xxx_DELETE.ITM`

  这些文件移除 ABAP 代码。无需导入 DELETE 传输，除非停止使用整个产品并且想要从 SAP 系统中移除传输。请参阅：第 632 页的『从 SAP 系统删除 ABAP 传输』。

3. 如下所示将传输文件复制到 SAP Transport System 数据目录，请勿更改传输文件名：

   Unicode 传输

   a. 将 `K711_00xxxU.ITM` 文件复制到 `cofiles` 目录

   b. 将 `R711_00xxxU.ITM` 文件复制到 `data` 目录。

4. 要在 SAP 系统上安装单个 IBM Tivoli Monitoring 传输文件，请选择以下一个文件导入选项：

   - 对于 Solution Manager 7.1 Service Pack 6 级别或更高版本且配置了 MAI 的 SAP 系统，必须在 SAP 系统的 `usr/sap/SID/DVEBMGS`*`instancenumber`*`/work` 工作目录中创建 `transport.prop` 文件。如果 SAP 系统是带有 ABAP SAP Central Services (ASCS) 的分布式系统，那么在中心实例 (CI) `usr/sap/SID` 目录中创建 `transport.prop` 文件。然后，在此文件中添加 `MAI_CONFIGURED = YES` 条目。此条目会在 `/IBMMON/ITM_CNFG` 表中创建 `MAI_CONFIGURED = YES` 条目。现在，您可以在 SAP 系统上导入单个 IBM Tivoli Monitoring 传输文件。

     注： 在导入单个 IBM Tivoli Monitoring 传输文件之前，必须在 SAP 系统的 `usr/sap/SID/DVEBMGS`*`instancenumber`*`/work` 工作目录中创建 `transport.prop` 文件，并在该文件中添加 `MAI_CONFIGURED = YES` 条目。不得编辑 `/IBMMON/ITM_CNFG` 表中的对应条目。

   - 对于基本版本等于 7.0 或更高版本、具有 Solution Manager V7.1 且无 MAI 配置的所有其他 SAP 系统，必须直接导入单个 IBM Tivoli Monitoring 传输文件。

5. 运行以下命令以导入 SAP 传输：

   ```
   tp addtobuffer ITMK711_00xxxU SID
   pf=\usr\sap\trans\bin\PROFILE_NAME
   ```

   其中：

   SID
   目标 SAP 系统标识。

   PROFILE_NAME
   概要文件的名称。确保在从命令行导入代理程序传输文件时指定当前 tp 参数文件。tp 参数文件的名称通常为 `TP_DOMAIN_`*`SID`*`.PFL`。在 UNIX 系统上，此文件名区分大小写。

   nnn
   代理程序运行所在的目标客户机的编号，将针对其定义用户标识、IBMMON_AGENT、授权概要文件和 `/IBMMON/AUTH`。

   此外，您可以使用 SAP STMS 传输以导入 ITMK711_00xxxU.ITM 传输请求。确保在“**导入传输请求**”窗口的**导入选项**选项卡中选中以下选项。

   - **将传输请求保留在队列中以供稍后导入**
   - **重新导入传输请求**
   - **覆盖原始**
   - **覆盖处于未确认修复的对象**

   对于 SAP Basis 版本，如果启用**忽略有效的组件版本**选项，请确保选中。

结果
根据您的 SAP 发行版级别，在运行 **tp import** 命令时，可能会收到返回码 4，这并不表示发生问题。收到返回码 4 是 **import** 命令的预期结果。

SAP 代理程序所需的用户和授权

要阻止对 SAP 系统进行未授权访问，可向登录到 SAS 系统的用户分配授权。这些授权定义用户在 SAP 系统中的访问级别。

导入 ABAP 传输后，SAP 代理程序会在 SAP 系统中创建缺省用户标识 IBMMON_AGENT，其缺省密码为 ITMMYSAP。此用户是系统用户，且 /IBMMON/AUTH 授权概要文件与此用户关联。导入 ABAP 传输后，会创建 /IBMMON/AUTH 概要文件和 IBMMON_AGENT 用户。使用 /IBMMON/AUTH 概要文件，IBMMON_AGENT 用户可访问从 SAP 系统读取性能数据所需的事务。以下列出所使用的一些示例事务：

- CCMS 警报和管理
- 针对 PI/XI 消息监视的授权
- 解决方案管理器授权

您可以为代理程序创建任何其他系统类型用户。必须向此用户分配了 /IBMMON/AUTH 概要文件。

要查看和访问 SAP 组件的数据，请确保为此代理程序创建的用户具有下表中指定的所有授权：

表 *201.* 授权的列表

组件	授权对象	授权描述
包含以下组件的常规系统授权： · SAP 实例 · SAP 系统	S_ADMI_FCD	用于访问 SAP 系统
	S_BDS_DS -BC-SRV-KPR-BDS	用于访问文档集
	S_BTCH_JOB	用于针对后台作业执行操作
	S_CCM_RECV	用于传输中央系统存储库数据
	S_C_FUNCT	用于在 ABAP 程序中执行 C 内核函数调用
	S_DATASET	用于访问文件
	S_RFC	用于检查 RFC 访问权。S_RFC 授权对象包含以下两个子授权： · RFC1：用于为 RFC1 函数组提供授权。 · SDIFRUNTIME：用于为 SDIFRUNTIME 函数组提供授权。
	S_RFCACL	用于检查 RFC 用户的授权
	S_RZL_ADM	用于访问 R/3 系统管理的计算中心管理系统 (CCMS)
	S_TCODE	用于检查可启动为应用程序定义的事务的授权
	S_TOOLS_EX	用于在监视工具中显示外部统计信息记录
针对包含 SAP 流程集成的 PI 的授权	S_XMB_MONI	用于访问 XI 消息监视

表 *201.* 授权的列表 *(续)*		
组件	**授权对象**	**授权描述**
针对包含 SAP 解决方案管理器的 MAI 的授权	AI_DIAGE2E	用于限制 E2E 诊断功能
	AI_LMDB_OB	用于访问全景管理数据库 (LMDB) 对象
	SM_MOAL_TC	用于控制对 SAP 解决方案管理器中警报和监视功能的访问权
	SM_WC_VIEW	用于限制对解决方案管理器的工作中心中特定 UI 元素的访问权
	S_RFC_ADM	用于控制可管理 RFC 目标的权限
	S_RS_AUTH	用于在角色中指定分析授权
	SM_APPTYPE	用于访问解决方案管理器应用程序类型
	SM_APP_ID	用于访问工作中心中提供的应用程序

从 SAP 系统删除 ABAP 传输

如果选择从系统移除 SAP 代理程序，那么必须将 delete 传输导入 SAP 系统。Delete 传输将删除 SAP 代理程序目录对象和功能模块。

开始之前

在从 SAP 系统删除该传输之前，必须停止配置为监视 SAP 系统的 SAP 代理程序实例。

如果 SAP 系统为 V7.20 或更新版本，在导入 delete 传输之前，必须在传输概要文件中添加以下传输概要文件参数：**tadirdeletions=true**。此传输概要文件参数可用于 tp V375.57.68，以及 R3trans V6.14 R700 或更高版本。有关从 SAP 系统移除传输请求的更多信息，请参阅删除传输请求。

过程

1. 转至以下路径：
   - 对于 Windows：*install_dir*\TMAITM6_x64\ABAP
   - 对于非 Windows：*install_dir*/*intrp*/sa/ABAP，其中，*intrp* 必须是 **lx8266** 或 **aix526**。
2. 将传输文件复制到 SAP 环境。
3. 将 K711_00xxx_DELETE 和 R711_00xxx_DELETE 文件复制到 SAP Transport System 数据目录，如下所示：
   a) 将 K711_00xxx_DELETE 文件复制到 cofiles 目录。
   b) 将 R711_00xxx_DELETE 文件复制到 data 目录。
4. 运行以下命令以导入 delete 传输：
   a) **tp addtobuffer ITMK711_00xxx_DELETE SID pf=\usr\sap\trans\bin***PROFILE_NAME*
   b) **tp import ITMK711_00xxx_DELETE SID client=nnn U16 pf=\usr\sap\trans\bin** *PROFILE_NAME*，其中：

      SID
      目标 SAP 系统标识。

      PROFILE_NAME
      概要文件的名称。

      nnn
      代理程序运行所在的目标客户机的编号。

验证代理程序配置

在安装 SAP 代理程序 后，必须通过下载、复制并验证 NetWeaver RFC SDK V7.20 库来验证代理程序配置。您还必须带有 MAI_Monitoring 的 Solution Manager V7.1 配置，验证 MAI 警报，并验证特定于第三方组件的配置设置。

通过完成以下过程验证代理程序配置：

· 第 633 页的『下载 NetWeaver RFC SDK V7.20 库』
· 第 633 页的『复制 SAP 代理程序设置中的 NetWeaver RFC SDK V7.20 库』
· 第 634 页的『验证 NetWeaver RFC SDK V7.20 库』
· 第 634 页的『验证带有 MAI-Monitoring 的 Solution Manager V7.1 配置』
· 第 635 页的『验证 MAI 警报』
· 第 636 页的『验证特定于第三方组件的配置设置』

下载 NetWeaver RFC SDK V7.20 库

在完成安装 SAP 代理程序后，下载 NetWeaver RFC SDK V7.20 库。与 NetWeaver RFC SDK V7.20 库相关的所有文件都可从 SAP Web 站点下载。

过程

1. 使用以下 URL 登录到 SAP Marketplace：
   http://service.sap.com
2. 单击 **SAP Support Portal**。
3. 输入您的 Service Marketplace 用户名和密码。
4. 单击 **Software Downloads** 并展开 **Support Packages and Patches** 链接。
5. 单击 **Browse our Download Catalog**，然后单击 **Additional Components**。
6. 单击 **SAP NetWeaver RFC SDK**，然后单击 **SAP NetWeaver RFC SDK 7.20**。
7. 选择您具有 SAP 代理程序的操作系统。
8. 将 `*.SAR` 文件下载到计算机。
9. 要使用 SAP 提供的 SAPCAR 实用程序解压缩 `SAP Netweaver RFC SDK *.SAR` 文件，请运行以下命令：
   `sapcar -xvf SAP NetWeaver RFC SDK File Name.SAR`

   注： 您可以从 SAP Web 站点下载 SAPCAR 实用程序。
10. 浏览至解压文件夹中的 `lib` 文件夹。

下一步做什么

将 NetWeaver RFC SDK V7.20 库复制到 SAP 代理程序设置。

复制 SAP 代理程序设置中的 NetWeaver RFC SDK V7.20 库

NetWeaver RFC SDK V7.20 库包含必须在 SAP 代理程序设置位置中手动复制的文件。

过程

1. 浏览至下载 NetWeaver RFC SDK V7.20 库的目录。
2. 将文件复制到 SAP 代理程序设置位置。
   · 对于 Windows 64 位操作系统，必须复制以下文件：
   - `icuin34.dll`
   - `libicudecnumber.dll`
   - `libsapucum.dll`
   - `icudt34.dll`
   - `icuuc34.dll`

- sapnwrfc.dll

必须将这些文件复制到 install_dir\TMAITM6_x64 位置。

- 对于非 Windows 操作系统，必须将这些文件复制到 install_dir/*intrp*/sa/lib 位置，其中 *intrp* 是操作系统代码（aix526、li6263、sol606）。必须复制以下文件：
  - libsapnwrfc.so
  - ibicudecnumber.so
  - ibicuuc34.a
  - libicui18n34.a
  - libicudata34.a
  - libsapucum.so

下一步做什么

验证安装的 NetWeaver RFC SDK V7.20 库的版本。

验证 NetWeaver RFC SDK V7.20 库

在复制解压缩的文件后，必须验证文件版本。

过程

- Windows 要验证文件的版本，请完成下列步骤：
  a) 右键单击 sapnwrfc.dll，然后单击**属性**。
  b) 单击**版本**选项卡。
  c) 在**产品版本**部分中，确保具有以下版本：720, patch 514, changelist 1448293（或更高版本）。
- Linux AIX 要验证文件的版本，请完成下列步骤：
  a) 转至解压缩的 *.SAR 文件中的 lib 文件夹。
  b) 运行以下命令：**strings libsapnwrfc.so | grep SAPFileVersion**
  c) 必须看到以下消息：[root@IBMSAP2V6 lib]# strings libsapnwrfc.so | grep SAPFileVersion GetSAPFileVersion #[%]SAPFileVersion: 7200, 514, 22, 6206 .GetSAPFileVersion

注: 消息显示该库版本为 720，补丁为 514（或更高版本）。

验证带有 MAI-Monitoring 的 Solution Manager V7.1 配置

要接收 MAI 警报的数据，必须验证是否正确配置了 Solution Manager V7.1。

关于此任务

您可以将 Solution Manager V7.1 与 MAI-Monitoring 和 Alerting Infrastructure 一起使用来监视受管系统。Solution Manager V7.1 监视自身以及卫星系统。每个卫星系统都具有一个插件和多个诊断代理程序。诊断代理程序将访存主机或操作系统级别的数据。每个主机都可以具有多个诊断代理程序以供不同的 Solution Managers 监视主机。以下是在 Solution Manager MAI Monitoring 中使用的关键字：

- 度量：来自卫星系统的数据。
- 警报：超过某些可配置的阈值的通知。
- 事件：转换为凭单并分配给任何用户的警报。

要使用 MAI 监视验证 Solution Manager V7.1 的配置，必须验证基本设置、全局级别设置以及模板级别设置。

过程

1. 要验证基本设置，请输入事务代码：SOLMAN_SETUP，然后按 **Enter** 键。

确保以下选项卡中的所有指示灯呈绿色：

- 概述
- 基本配置
- 受管系统配置

注： 存在不同类别的受管系统，例如，技术系统、技术场景、主机、数据库、实例、PI 域、技术组件和连接。必须根据业务需求配置这些受管系统。MAI 警报基于配置的受管系统。

2. 输入事务代码：SE38，然后按 **Enter** 键。
3. 提供者名 RTCCTOOL 并运行报告。

   确保输出中的所有指示灯呈绿色。
4. 要验证全局级别设置，请输入事务代码：SOLMAN_WORKCENTER，然后按 **Enter** 键。

   确保以下选项卡中的所有指示灯呈绿色：

   - 概述
   - 配置基础结构
   - 先决条件
   - 配置
5. 验证**全局设置**的**通知**状态是否为**活动**。
6. 要验证模板级别设置，请输入事务代码：SOLMAN_SETUP，然后按 **Enter** 键。

   在**自动通知**列表的**技术设置**中，确保选中**活动**。

   注： 对于初始故障诊断，确保电子邮件通知活动。
7. 对于 MAI 系统监视，使用以下步骤验证 End-User Experience Monitoring (EEM) 的配置：
   a) 输入事务代码：SE37，然后按 **Enter** 键。
   b) 在**功能模块名称**字段中输入 **AI_EEM_LIST_ALL_SCENARIOS**，然后按 F8。
   必须存在 End-User Experience Monitoring (EEM) 条目。

验证 MAI 警报

要确保正确配置了 Solution Manager MAI 以监视“技术监视”中的“MAI 警报收件箱”，必须验证接收 MAI 警报作为输出。

过程

1. 输入事务代码 SOLMAN_WORKCENTER，然后按 **Enter** 键。检查是否可以在 Solution Manager“技术监视”下面的“MAI 警报收件箱”中查看 MAI 警报。
2. 使用以下步骤检查 BAdi 实施：
   a) 输入事务代码：SE19，然后按 **Enter** 键
   b) 在**增强功能实施**字段中输入 /IBMMON/ITM_IMPL_ALRTINBX。
   c) 单击**显示**并检查**运行时行为**部分中的 BAdi 实施是否活动。
3. 使用以下步骤检查数据库 /IBMMON/ITM_ALIX 是否包含 MAI 警报：
   a) 输入事务代码：SE16，然后按 **Enter** 键。
   b) 在**表名称**字段中，输入 /IBMMON/ITM_ALIX 并运行。确保收到表中的 MAI 警报。
4. 输入事务代码：SE37，然后按 **Enter** 键。
5. 在**功能模块名称**字段中，输入 /IBMMON/ITM_MAIALRT_INX 并按 F8。

   必须看到输出是 MAI 警报。

下一步做什么

如果无法查看 /IBMMON/ITM_ALIX 数据库中的 MAI 警报，那么必须验证第三方组件中的设置。

验证特定于第三方组件的配置设置

如果无法查看 MAI 警报，那么必须验证第三方组件中的设置。

过程

1. 验证第三方组件活动。
2. 验证在“**操作系统适配器**”的 **BAdi 实施**中，**警报反应**可用。如果**警报反应**不可用，那么请移除缺省设置，然后选择 **BAdi 实施 - 警报反应**。
3. 使用以下步骤检查模板设置：
   a) 验证用于将特定警报传输至第三方系统的设置，例如，SAP ABAP 7.0.0。
   b) 依次选择**专家方式**、**警报**，然后单击**第三方组件**。
   确保能够查看“警报反应 BAdi”名称。

   注: 确保已实施最新的 SAP Note。对于 Solution Manager V7.1 Service Pack 8，检查是否实施了以下 Note：

   · https://service.sap.com/sap/support/notes/1959978
   · https://service.sap.com/sap/support/notes/1820727
4. 如果无法查看 /IBMMON/ITM_MAIALRT_INX 数据库中的 MAI 警报，必须针对第三方组件运行以下 Solution Manager MAI 配置步骤：
   a) 输入事务代码：SOLMAN_SETUP，然后按 **Enter** 键。
   b) 在“**技术监视**”中，选择**系统监视**。
   c) 单击**配置基础结构**选项卡，然后单击**缺省设置**选项卡。
   d) 单击**第三方组件**选项卡，然后单击**编辑**。
   e) 从列表中选择**活动**。
   f) 确保对于选中的连接器，将作用域过滤器设置为**所有警报、事件和度量（包括内部事件）**。

   注: “操作系统命令适配器”是用于将数据推送到第三方连接器的一种方法。要配置“操作系统命令适配器”，请阅读“操作系统命令适配器”的“如何做”指南中的配置详细信息设置。

添加数据库通信端口号

数据库通信端口号对于在集成场景中唯一标识数据库实体非常重要。为了实现组件间协作，SCM AI 的组件包含了 OSLC（开放式源代码生命周期协作）。在 OSLC 合规性中，唯一标识协作组件非常重要。因此，数据库通信端口号很重要。

关于此任务

将相关 IBM Tivoli Monitoring 传输导入到 SAP 系统时，将自动创建 /IBMMON/ITM_PORT 数据库表。此表包含以下数据库字段：

· 系统标识
· 系统主机名
· 数据库通信端口号

过程

要添加 OSLC 合规性所必需的用于 SAP 代理程序的 SAP 数据库通信端口号，请执行以下步骤：

1. 转至 SE16 事务代码，并按 Enter 键。
2. 在**数据库表名**字段中，输入 /IBMMON/ITM_PORT，然后按 F7。
3. 显示 **/IBMMON/ITM_PORT** 数据库表选择屏幕时，按 F8 键。
   /IBMMON/ITM_PORT 数据库表包含以下三个数据库字段：
   · 系统标识
   · 系统主机名

· 数据库通信端口号

注: /IBMMON/ITM_PORT 数据库表中显示的 SAP 系统同时适用于 Java 和 ABAP 体系结构。

4. 在**数据库通信端口号**字段中，针对不同 SAP 系统标识和 SAP 系统主机名输入相关的 SAP 数据库端口号，然后保存更改。

   注: 如果未在 /IBMMON/ITM_PORT 数据库表的**数据库通信端口号**字段中输入任何值，那么缺省情况下，数据库通信端口号将为 0。

SAP 代理程序的高级安装和配置

这些是 SAP 代理程序特定的高级安装和配置。

描述了以下安装和配置主题:

· 第 637 页的『SAP 函数模块』
· 第 638 页的『SAP 用户标识』
· SAP 代理程序的实用程序
· 第 638 页的『SAP RFC 连接』
· 第 647 页的『“测试连接”功能部件』
· 第 640 页的『SAP 中的可选高级配置』
· 第 645 页的『CEN CCMS 报告』
· 第 646 页的『从 SAP 系统卸载高级业务应用程序编程 (ABAP) 传输』

注: SAP 代理程序的高级安装和配置包含对 IBM Tivoli Monitoring 的引用，从而使文档兼容 ABAP 传输定制事务代码 UI。

SAP 函数模块

当数据量在 SAP 服务器上较高时，您可能会遇到某些窗口小部件导致来自服务器的响应时间缓慢的问题。如果窗口小部件并非关键，那么可以禁用关联的 SAP 函数模块。

缺省情况下，SAP 代理程序函数模块处于启用状态。但是，以下函数模块在缺省情况下处于禁用状态:

· SYS 子节点下的 HTTP 服务（/IBMMON/ITM_HTTP_SRVS）
· PI/XI 子节点下的 XML 消息（/IBMMON/ITM_SXMB_MONI_NEW）
· PI/XI 子节点下的同步/异步通信（/IBMMON/ITM_SYN_ASYN_COMM）
· Sys 子节点下的 qRFC 入站队列详细信息（/IBMMON/ITM_QIN_QDETAILS）

禁用 SAP 函数模块后，如果选择窗口小部件，那么 IBM Application Performance Management 仪表板上不会显示数据。因此，您可避免任何与性能相关的问题。

启用 SAP 代理程序功能模块

如果先前禁用了 SAP 代理程序功能模块以解决性能问题，那么也可以启用此功能模块。

过程

1. 登录到 SAP 系统。
2. 运行 SE16 事务代码。
3. 输入表名称 /IBMMON/ITM_CNFG。
4. 选择要删除的行，然后按 **shift + F2** 以删除条目。
5. 单击**保存**。

禁用 SAP 函数模块

某些窗口小部件可能导致 SAP 服务器响应缓慢，因此您可以禁用 SAP 功能模块以提高服务器性能。

过程

1. 登录到 SAP 系统。

2. 运行 SE16 事务代码。
3. 输入表名称 /IBMMON/ITM_CNFG。
4. 按 **F5** 以创建新条目。
5. 在 **PARM NAME** 字段中输入 SAP 功能模块的名称。
6. 在 **VALUE CHAR** 字段中输入 No。
7. 单击**保存**。

SAP 用户标识
本部分提供有关 SAP 代理程序所需的 SAP 用户标识和许可权的信息。

用户标识支持以下用途：

· 第 638 页的『SAP RFC 连接』
· 第 638 页的『基本代理程序监视』

***SAP RFC* 连接**
SAP 代理程序使用远程函数调用 (RFC) 连接进行内部集中计算中心管理 (CCMS) 轮询和 CCMS 警报数据收集。此行为特定于 SAP RFC 体系结构。

SAP 代理程序打开一个与受代理程序监视的 SAP 系统的专用 RFC 连接。然后，SAP 系统在每个应用程序服务器上打开一个内部连接，以通过函数模块和程序进行数据收集。如果 CCMS 警报由代理程序收集，那么 SAP 系统将为此收集线程打开一个与各应用程序服务器的额外（系统内部）RFC 连接。当数据收集启动时，将会打开代理程序的一个 RFC 连接。然后，打开额外内部系统 RFC 连接所对应的高达两倍数量的 SAP 应用程序服务器。

您必须确保进行监视的实例可以容纳其他 RFC 会话，尤其是在具有 10 个或以上实例的大型系统中。当预计要监视的 RFC 负载可能对系统性能和容差造成不利影响时，请调整 SAP 概要文件参数。请联系 SAP 管理员并参阅以下 SAP 注释：

· 终端会话（缺省设置：200）22099
· 通信/网关/对话设置 887909 316877 384971

基本代理程序监视
在导入代理程序传输时，SAP 代理程序在 SAP 系统中创建一个 IBMMON_AGENT。

此用户标识为 IBMMON_AGENT，缺省密码为 ITMMYSAP。预先配置为仅限通信类型用户并且使用 /IBMMON/AUTH 授权概要文件。在传输导入时创建此概要文件，其中包含一组最低许可权以运行代理程序 Advanced Business Application Programming (ABAP) 代码。另外，此概要文件接受 SAP 系统上一组有限操作。

如果此用户标识名称不可接受，例如，如果其违反安装命名约定，那么可创建不同的用户标识。用户标识可以是任何允许的 SAP 用户标识，但是需要 /IBMMON/AUTH 概要文件中完整的许可权集。用户标识需要仅限通信类型用户访问权。

缺省用户标识提供仅满足以下目的的足够权限：

· 监视和数据收集
· 关闭计算中心管理系统 (CCMS) 警报
· 启用、禁用和重置网关统计信息
· 重置 Oracle 数据库统计信息

如果选择限制代理程序的操作功能，那么可移除某些操作许可权，例如，关闭 CCMS 警报。

要在 IBM Application Performance Management UI 门户网站上访问特定组件的数据，请确保具有相应的授权。下表列出访问不同子节点的数据所需的授权：

表 *202.* 授权的列表

子节点	授权对象	授权描述
包含以下子节点的常规系统授权： · Ins · Sys	S_ADMI_FCD	用于访问系统
	S_BDS_DS -BC-SRV-KPR-BDS	用于访问文档集
	S_BTCH_JOB	用于针对后台作业执行操作
	S_CCM_RECV	用于传输中央系统存储库数据
	S_C_FUNCT	用于在 ABAP 程序中执行 C 调用
	S_DATASET	用于访问文件
	S_RFC	用于检查 RFC 访问权。S_RFC 授权对象包含以下两个子授权： · RFC1：用于为 RFC1 函数组提供授权。 · SDIFRUNTIME：用于为 SDIFRUNTIME 函数组提供授权。
	S_RFCACL	用于 RFC 用户
	S_RZL_ADM	用于访问计算中心管理系统 (CCMS)：系统管理
	S_TCODE	用于在事务启动时检查事务代码
	S_TOOLS_EX	用于访问工具性能监视器
包含以下子节点的解决方案管理器的授权： · Lds · Sol	D_MD_DATA -DMD	用于查看主数据的数据内容
	D_SOLMANBU	用于访问解决方案管理器的会话类型
	D_SOLM_ACT	用于访问解决方案管理器中的解决方案
	D_SOL_VSBL	用于查看解决方案管理器中的解决方案
	S_CTS_SADM	用于查看特定于系统的管理（传输）
	S_TABU_RFC	用于查看客户机比较和复制：使用 RFC 的数据导出
包含 PI 子节点的 PI 的授权	S_XMB_MONI	用于访问 XI 消息监视

表 202. 授权的列表 (续)		
子节点	**授权对象**	**授权描述**
包含 Slm 子节点的 MAI 的授权	AI_DIAGE2E	用于访问解决方案诊断端到端分析
	AI_LMDB_OB	用于访问全景管理数据库 (LMDB) 对象
	SM_MOAL_TC	用于访问监视和警报
	SM_WC_VIEW	用于访问工作中心用户界面元素
	S_RFC_ADM	用于访问 RFC 目标的管理选项
	S_RS_AUTH	用于访问角色中的 BI 分析
	SM_APPTYPE	用于访问解决方案管理器应用程序类型
	SM_APP_ID	用于访问工作中心中提供的应用程序

使用“中央用户管理”(CUA)

“中央用户管理”(CUA) 用于监视 SAP 系统。

过程

要使用预定义的用户标识和授权角色来监视具备“中央用户管理”的 SAP 系统，请完成下列其中一个步骤：

- 将传输安装到“中央用户管理”父逻辑系统客户机中。
- 在要安装传输的客户机中手动创建用户标识或角色。该用户标识或角色位于安装（导入）传输的客户机中。
- 在“中央用户管理”父逻辑系统客户机中手动创建用户标识或角色。然后，将该用户标识或角色分发到运行代理程序的客户机。
- 在“中央用户管理”父逻辑系统客户机中手动创建用户标识或角色，并在此客户机中运行代理程序。

SAP 中的可选高级配置

您可以使用标准 SAP 或代理程序提供的 SAP 功能配置 SAP 代理程序。

在 SAP 中使用代理程序提供的事务以定制大量代理程序行为。在运行 `/n/IBMMON/ITM_CONFIG` 事务以访问 SAP 中的主配置菜单后，选择以下一个配置选项：

- 第 641 页的『复制、备份、复原功能部件和事务』
- 第 641 页的『使用事务复制、备份和复原数据』
- 第 642 页的『命令行实用程序工具』
- 第 642 页的『在 Windows 环境上运行命令行实用程序』
- 第 643 页的『在非 Windows 环境上运行命令行实用程序』
- 第 643 页的『警报维护』
- 第 644 页的『选择监视器集和监视器事务』
- 第 644 页的『配置 SAP 系统中的“对话框步骤响应阈值”』

注: 您必须为所有 `/IBMMON/ITM*` 事务前附加 `/n`。

SAP 代理程序将立即使用在这些事务中执行的配置更改，对主受管组执行的这些更改除外。在受管组配置更改时，SAP 代理程序将在下一次脉动信号时发现更改。

使用 SAP 标准函数以完成以下配置：第 644 页的『配置 SAP 系统中的“对话框步骤响应阈值”』

复制、备份、复原功能部件和事务

在您登录到 SAP 服务器并运行以下事务后，复制、备份和复原功能部件可供使用：/n/IBMMON/ITM_CONFIG。

通过复制、备份和复原操作，可以复制、备份和复原 IBM Tivoli Monitoring 配置数据。

使用此功能部件选择以下功能并保存 IBM Tivoli Monitoring 配置数据：

- **复制**

  使用此功能部件可将 IBM Tivoli Monitoring 配置设置从一个 SAP 服务器复制到另一个 SAP 服务器。例如，您可能希望将 IBM Tivoli Monitoring 配置设置从代理程序 **a1** 复制到 SAP 服务器实例 SAP2。此代理程序在系统 **m1** 上运行并针对 SAP 服务器实例 SAP 1 进行了配置。所有 IBM Tivoli Monitoring 配置设置（SAP 服务器实例监视设置除外）都复制到目标 SAP 系统。您通过使用命令行实用程序或 SAP GUI 来实施复制功能部件。

- **备份**

  您可以通过对系统进行备份来存储已在 SAP 服务器上完成的特定于代理程序的配置。使用此功能部件可在 SAP 系统上保存特定于 IBM Tivoli Monitoring 的配置设置。您使用 /IBMMON/ITM_CONFIG 事务来输入设置。备份文件存储在 SAP 服务器上的以下路径的工作目录中：/usr/sap//DVEBMGS/work。

- **复原**

  使用此功能部件可从工作目录复原 SAP 服务器上的 IBM Tivoli Monitoring 配置数据。您可以在完成此配置数据的备份过程所在的同一 SAP 服务器上或另一个 SAP 服务器上复原 IBM Tivoli Monitoring 配置数据。可以将 IBM Tivoli Monitoring 配置数据复制到特定 SAP 和 IBM Tivoli Monitoring 表。所存储的配置文件具有日期和时间戳记，因此可以选择要将文件复原到的点。

特定于代理程序的配置包含 SAP 中 /IBMMON/ITM_CONFIG 事务内的配置设置。您可以完成以下配置过程：

- 对警报的频率进行采样。
- 启用特定警报。
- 存储日志文件名。
- 管理组定义。
- 选择监视器集和监视器。
- 选择 SAP 实例以进行监视。

使用事务复制、备份和复原数据

在 SAP 用户界面上，您可以使用 /n/IBMMON/ITM_CONFIG 事务来复制、备份和复原数据。

开始之前

使用复制、备份和复原过程将 IBM Tivoli Monitoring 配置设置从一个 SAP 服务器复制到另一个 SAP 服务器。所有 IBM Tivoli Monitoring 配置设置（SAP 服务器实例监视设置除外）都复制到目标 SAP 系统。

过程

完成以下过程来复制、备份和复原 SAP 上的数据：

- **复制**
  a. 将目标 SAP 系统标识和现有文件名输入为 source system id__<filenam>date_time。

     /IBMMON/ITM_COPY 事务在工作目录中创建一个文件名为 SAP target SAP system id__<filename>_date_time 的 IBM Tivoli Monitoring 配置文件。

  b. 单击**执行**以将 IBM Tivoli Monitoring 配置数据复制到该文件。

  c. 单击**后退**或**取消**以返回到上一个 IBM Tivoli Monitoring 配置屏幕。

     预期的输入参数是要复制的 **Target System id** 和 **filename**。

- **备份**

a. 登录到 SAP 服务器并启动 /IBMMON/ITM_CONFIG 事务。

b. 选择**备份**。

c. 输入备份文件名。

该文件名存储为 sys_id_<filename>_date_time。

d. 单击**执行**以运行备份并将文件存储在应用程序服务器上。

注: 备份文件存储在应用程序的工作目录中。

e. 单击**后退**或**取消**以返回到上一个 IBM Tivoli Monitoring 配置屏幕。

· **复原**

a. 登录到 SAP 服务器并启动 /IBMMON/ITM_CONFIG 事务。

b. 选择**复原**。

c. 输入要复原的文件名，格式为 sys_id_<filename>_date_time。

d. 单击**执行**以复原 IBM Tivoli Monitoring 配置数据。

e. 单击**后退**或**取消**以返回到上一个 IBM Tivoli Monitoring 配置屏幕。

命令行实用程序工具

您可以使用命令行实用程序工具以在 SAP 服务器上复制、备份和复原 IBM Tivoli Monitoring 配置数据。

您可以在 Windows 和非 Windows 环境上运行命令行实用程序。请参阅第 642 页的『在 Windows 环境上运行命令行实用程序』和第 643 页的『在非 Windows 环境上运行命令行实用程序』。

· **复制**

运行 **backup** 命令以将 IBM Tivoli Monitoring 配置文件从代理程序目录 SAP 服务器实例 sap1 复制到 sap2。输入文件名和 sap1 作为来自 sap1 代理程序目录的源系统。然后，调用 ABAP 功能，此功能将 IBM Tivoli Monitoring 设置从此文件复制到 Sap2 的 IBM Tivoli Monitoring 配置文件。现在，从 sap1 代理程序目录实用程序工具中选择 **Copy**，并输入文件名和 sap2 作为目标 SAP 系统。

· **备份**

在运行命令行实用程序工具后，选择**备份**选项。然后，您需要输入文件名和 SAP 系统标识。工具调用 /IBMMON/ITM_BACKUP SAP 功能模块。功能模块读取存储在表中的特定 IBM Tivoli Monitoring 配置设置，并使用行和列分隔符存储它们。然后，命令行实用程序工具读取字符串并将数据写入到一个文件。生成的文件名具有以下格式：ID>_<filename>-<date&time>。此文件存储在存储实用程序的目录中。

· **复原**

在运行命令行实用程序工具后，输入要复原的文件名和想要复原文件的目标 SAP 文件。命令行实用程序工具从代理程序目录读取文件，并调用 /IBMMON/ITM_RESTORE SAP 功能模块。然后，工具作为一个字符串传递 IBM Tivoli Monitoring 配置。SAP 功能模块更新特定 IBM Tivoli Monitoring 表并复原特定 IBM Tivoli Monitoring 配置。

在 *Windows* 环境上运行命令行实用程序

您可以在 Windows 环境上运行命令行实用程序以完成复制、备份和复原过程。

过程

1. 根据操作系统，完成以下一个过程：
   · 对于 64 位操作系统，使用命令 **set CANDLE_HOME = C:\IBM\APM** 设置 CANDLEHOME 路径，然后从以下路径运行 **ksacopybackuprestore.bat** 命令：%candle_home%\ TMAITM6x64。
2. 要创建备份文件，请完成以下步骤：
   a) 选择**备份**，输入文件名和源 SAP 系统名称。
   b) 使用以下格式创建备份文件：SYS ID>_<filename>_<date&time>。
3. 要复原文件，请完成以下步骤：
   a) 选择**复原**，输入目标 SAP 系统名称。

b) 输入文件名。

4. 要复制文件，请完成以下步骤：
   a) 从源代理程序，选择**备份**并创建备份文件。
   b) 将备份文件从源代理程序目录复制到目标代理程序目录。
   c) 从源目录，运行命令行实用程序工具并选择**复制**。
   d) 输入文件名和目标 SAP 系统。

在非 *Windows* 环境上运行命令行实用程序

您可以在非 Windows 环境上运行命令行实用程序以完成复制、备份和复原过程。

过程

1. 从以下路径运行 **ksacopybackuprestore.sh** 命令：`/candle_home/<arch>/sa/shell`。
2. 要创建备份文件，请完成以下步骤：
   a) 选择**备份**，输入文件名和源 SAP 系统名称。
   b) 使用以下格式创建备份文件：SYS ID>_<filename>_<date&time>。
   备份文件将保存到此位置：`%candlehome% / arch /sa/bin`。
3. 要复原文件，请完成以下步骤：
   a) 选择**复原**，输入目标 SAP 系统名称。
   b) 输入文件名。
4. 要复制文件，请完成以下步骤：
   a) 从源代理程序，选择**备份**并创建备份文件。
   b) 将备份文件从源代理程序目录复制到目标代理程序目录。
   c) 从源目录，运行命令行实用程序工具并选择**复制**。
   d) 输入文件名和目标 SAP 系统。

警报维护

您可以通过更改 Tivoli Monitoring 所生成的警报的状态或阈值来修改这些警报。

此事务用于启用或禁用 Tivoli Monitoring 所生成的警报以及设置警告和紧急阈值。Tivoli Monitoring 所生成的所有警报都显示有其当前状态和阈值。

修改警报状态和阈值时，在下一个样本时间会使用已修改的值。

缺省样本周期维护

缺省样本周期提供有关特定属性组的实时报告的信息。

某些属性组具有组中每个记录的隐式日期和时间。例如，R/3_Abap_Dumps 属性组报告转储的创建时间，R/3_System_Log 属性组报告日志条目的创建时间。这些记录具有日期和时间字段。您可以获取表的简短历史记录而不只是最新信息的报告。此时间间隔是数据收集的时间范围，并在收集数据时用作实时时间间隔。/IBMMON/ITM_PERIOD 事务定义其中每个属性组的缺省样本周期（实时报告的时间范围）。样本周期确定从当前时间开始并回溯的数据样本周期的长度。

日志文件名维护

仅与实例匹配的特定日志文件包含在具有日志文件信息的 IBM Tivoli Monitoring 报告中。

此事务用于识别要考虑在含有日志文件信息的 IBM Tivoli Monitoring 报告中包含哪些日志文件。名称与指定实例上的指定名称模式匹配的所有日志文件都会在下一个数据收集时间间隔包含在该报告中。

受管组维护

受管组名事务监视并处理 SAP 系统中的特定事务。

此事务用于维护 IBM Tivoli Monitoring 受管组定义。所有受管组名都传递到 IBM Application Performance Management UI 门户网站并显示在受管系统选择列表中。在进行数据收集时，仅会将与属性选择条件匹配的数据发送到 SAP 代理程序。此数据显示在报告中或用于在情境和策略中进行评估。

您可以使用受管组来监视 SAP 系统中信息的子集。仅关注您感兴趣的 SAP 系统部分并忽略其他无关紧要的部分。例如，如果您仅对财务应用程序所包含的交易响应时间感兴趣，那么可以创建名为 Financials 的受管组。然后，在其中仅包含财务交易代码。只要 Tivoli Enterprise Portal 对 Financials 受管组进行了处理，那么就会在显示报告、评估情境或评估策略时仅考虑包含指定交易代码的信息。

选择监视器集和监视器事务
使用选择监视器集和监视器事务来编辑集中计算中心管理 (CCMS) 警报配置。例如，您可以完全关闭 CCMS 警报收集。

此事务用于选择 IBM Tivoli Monitoring 从中检索警报的 CCMS 监视器。缺省情况下，首次显示此窗口时选择“整个系统”监视器。您可以更改监视器集和/或监视器，然后保存配置。您最多可以选择三个要为其收集 CCMS 警报的监视器。

要完全关闭 CCMS 警报收集，请清除所有监视器的对应复选框并保存此配置。

已在运行的代理程序将读取此配置并收集选定监视器的 CCMS 警报。但是，在更改 CCMS 警报配置之前代理程序已收集的任何 CCMS 警报都属于代理程序和 IBM Tivoli Monitoring。

除选择监视器和监视器集以外，此事务还指定要检索的警报类型的出现次数。此外，它有助于决定是否自动关闭未检索的较旧的警报实例。

配置 *SAP* 系统中的“对话框步骤响应阈值”
您可以通过运行 SE16 事务，针对任何事务配置“对话框步骤响应阈值”。

过程

1. 在**表名称**字段中，输入 /IBMMON/ITM_TRSH，然后选择**表内容 (F7)** 以访问表。
2. 要查看当前阈值设置，请选择**执行 (F8)**。事务名称将在**工作负载**列下显示；阈值将在**阈值**列下显示。
3. 要添加新阈值设置，请选择**创建 (F5)**。在**工作负载**字段中输入事务名称。针对**工作负载**值接受以下通配符：
   - * 匹配多个字符
   - + 匹配任意单个字符
4. 在**阈值**字段中输入阈值（以毫秒为单位）。选择**保存**以保存此设置。新的和更改的阈值不会立即生效，而是在以下任何条件后生效：
   - 重新启动代理程序。
   - 代理程序打开其到 SAP 系统的 RFC 连接。此过程每隔 12 个脉动信号发生一次，缺省情况下，约为每 2 小时 10 分钟。

结果
将在 R/3_Transacation_Performance 属性组的“对话框步骤响应阈值”属性中返回针对**阈值**列输入的值。

批处理作业操作
您可以访存指定的时间间隔内的所有批处理作业。

过程

遵循第 627 页的『在 SAP 系统上导入 ABAP 传输』后的步骤。

切记: 针对所有批处理作业设置“严重”常量。

1. 要访存指定的时间间隔内的所有活动和已取消的批处理作业。
   在 /IBMMON/ITM_CNFG 表中添加以下条码。

表 *203. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES

2. 要访存指定的时间间隔内的所有已取消的作业和与时间间隔无关的所有活动作业。
   在 /IBMMON/ITM_CNFG 表中添加以下条码。

表 *204. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES_LONG_RUN

3. 要访存指定的时间间隔内的所有批处理作业和与时间间隔无关的所有活动批处理作业。
   在 /IBMMON/ITM_CNFG 表中添加以下条码。

表 *205. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
BATCH_JOBS_PERF	YES_ALL

注:

· 如果未添加配置参数，那么其访存指定时间间隔内未设置“严重”常量的所有批处理作业。
· 访存的行号总是等于在事务代码 /n/IBMMON/ITM_CONFIG 中设置的“严重”常量的值。

提高 */IBMMON/ITM_MAIALRT_INX* 函数模块的性能
您可以针对 SAP 代理程序 增强 /IBMMON/ITM_MAIALRT_INX 函数模块的性能。

过程

遵循步骤以提高 /IBMMON/ITM_MAIALRT_INX 函数模块的性能。

1. 登录到 SAP 代理程序 GUI。
2. 运行 SE16 事务代码并输入表名 /IBMMON/ITM_CNFG，然后按 F7。
3. 按 F5 或者单击**创建条目**，然后在 IBMMON/ITM_CNFG 表中添加以下条目。

表 *206. /IBMMON/ITM_CNFG*	
PARM_NAME	**VALUE_CHAR**
MAI_ALERTS_PERF	YES

注:

· 如果未在事务代码 - /N/IBMMON/ITM_CONFIG 中设置“严重”常量，那么缺省值为 2500。
· 此过程仅适用于从 PERIOD_START 和 PERIOD_END 为初始值的 SAP 系统访存 MAI 警报。

切记: 现在，函数模块 /IBMMON/ITM_MAIALRT_INX 访存 MAI 警报的数量，等效于事务代码 - /N/IBMMON/ITM_CONFIG 中设置的“严重”常量。

· 如果缺省情况下，未在 /IBMMON/ITM_CNFG 中创建此条目，那么将访存至少 2500 个 MAI 警报。
· 访存的行号总是等于在事务代码 /n/IBMMON/ITM_CONFIG 中设置的“严重”常量的值。

CEN CCMS 报告
集中式 (CEN) 计算中心管理系统 (CCMS) 是一项 SAP 监视功能。

使用此功能可以向监视中心报告多个 SAP 系统的 CCMS 警报。您从一个 CCMS 控制台监视 SAP 环境。在以下环境中，最适合使用集中式 CCMS 报告:

· CCMS 操作占主导地位，其中 CCMS 警报是唯一需要的监视数据。

- 集中式 CCMS 是 SAP 环境的组成部分。
- 具有许多 SAP 系统的大型 SAP 环境，例如，ISV 和 ISP。
- IBM Tivoli Monitoring V5.x 与 SAP 代理程序 CCMS 适配器集成。
- 收集来自非 ABAP SAP 组件和应用程序服务器的警报。

SAP 代理程序仅支持使用集中式 CCMS 来报告警报。然后，您将一个 SAP 代理程序 安装在集中式 SAP 系统上，即可查看整个 SAP 环境的 CCMS 警报。此支持是通过以下方式提供：

- 在报告 CCMS 警报时，代理程序检查这些警报是否与代理程序直接监视的 SAP 系统相关联。如果代理程序确定某警报属于另一个 SAP 系统，它就会采用集中式 CCMS，并自动创建额外的 R3_Group 受管系统。
- <local_SID>-All_CCMS_alerts:Grp 受管系统用于报告来自所有远程 SAP 系统的全部警报。<local_SID> 的值是直接监视的 SAP 系统的系统标识。例如，如果本地 SAP 系统为 QA1，那么此组名将是 QA1-All_CCMS_alerts:Grp。
- <local_SID>-<remote_SID>_CCMS_alerts:Grp 受管系统用于报告来自一个远程 SAP 系统的所有警报。<local_SID> 的值是直接监视的 SAP 系统的系统标识。<remote_SID> 的值是远程 SAP 系统的系统标识。例如，如果本地 SAP 系统为 QA1，远程 SAP 系统为 QA2，那么此组名将是 QA1-QA2_CCMS_alerts:Grp。
- 导航器树中的每个受管系统下都有完整的一组窗口小部件，但是仅“警报”窗口小部件包含有意义的数据。

SAP 代理程序在直接管理的 SAP 系统的“高级业务应用程序编程”(ABAP) 代码中维护其集中式 CCMS 组定义。如果要从中接收集中式警报的 SAP 系统也由 SAP 代理程序的另一个实例直接监视，您可能需要修改这些定义。您不会希望这些警报同时在这两个系统下报告。您可以限制集中式警报报告，如下所示：

- 使用 /IBMMON/ITM_CONFIG 事务来维护受管组。更改“所有 CCMS”警报组。通过编辑组定义以排除 (EXCLUDE) 远程系统标识，从此列表中移除远程系统。
- 使用 /IBMMON/ITM_CONFIG 事务来维护受管组。删除 <remote_SID> CCMS 警报组。例如，如果远程 SAP 系统为 QA2，那么此组名将是 QA2_CCMS_alerts。

或者，您可以使用集中式 CCMS 来报告来自所有 SAP 系统的警报，但是阻止本地安装的每个代理程序报告警报。请使用以下步骤来设置此配置：

- 配置 SAP 代理程序的实例以监视集中式 CCMS 系统。允许代理程序检测并报告来自所有远程 SAP 系统的所有警报。
- 配置 SAP 代理程序的实例以监视每个远程 SAP 系统。通过使用 /IBMMON/ITM_CONFIG 事务来选择监视器集和监视器，针对这些代理程序实例禁用警报收集和报告。在此功能中，请取消选中所有监视器的复选框，并保存此配置。

SAP 代理程序对集中式 CCMS 的支持适用于纯 CCMS 监视环境，您可在公共控制台上查看所有警报。另外，也可利用其全套功能，以提供用于远程 SAP 系统的情境和策略，以及“执行操作”命令。

从 SAP 系统卸载高级业务应用程序编程 (ABAP) 传输

如果您选择从系统中移除 SAP 代理程序，那么必须将 Delete 传输导入到 SAP 系统。Delete 传输将删除 SAP 代理程序目录对象和功能模块。

开始之前

如果 SAP 系统为 V7.20 或更新版本，在导入 delete 传输之前，必须在传输概要文件中添加以下传输概要文件参数：**tadirdeletions=true**。此传输概要文件参数可用于 tp V375.57.68，以及 R3trans V6.14 R700 或更高版本。有关从 SAP 系统移除传输请求的更多信息，请参阅删除传输请求。

过程

1. 转至产品 CD 上的 /ABAP 目录。
2. 将传输文件复制到 SAP 环境。
3. 将 K711_00xxx_DELETE 和 R711_00xxx_DELETE 文件复制到 SAP Transport System 数据目录，如下所示：

   a) 将 K711_00xxx_DELETE 文件复制到 cofiles 目录。

b) 将 R711_00xxx_DELETE 文件复制到 data 目录。

4. 运行以下命令：

a) **tp addtobuffer ITMK711_00xxx_DELETE SID pf=\usr\sap\trans\bin***PROFILE_NAME*

b) **tp import ITMK711_00xxx_DELETE SID client=nnn U16 pf=\usr\sap\trans\bin** *PROFILE_NAME*

其中：

SID
目标 SAP 系统标识

PROFILE_NAME
tp 概要文件的名称

nnn
代理程序将要运行所在的目标客户机的编号

SAP 实例定制

缺省情况下，所有 SAP 系统实例都受监视并显示在 IBM Application Performance Management UI 门户网站上。

作为管理员，您可以选择要监视的 SAP 实例。此外，作为管理员，还可关闭不希望监视的 SAP 实例。

/IBMMON/ITM_INSTANCE 定制事务链接到 /IBMMON/ITM_CONFIG 事务。

选择 **SAP 实例**选项以查看 SAP 服务器的可用实例。然后，选择要监视的实例。这些实例会显示在 IBM Application Performance Management UI 门户网站上。IBM Application Performance Management UI 门户网站上不会显示任何不活动或已清除的实例。

“测试连接”功能部件

通过“测试连接”功能部件，您可以验证是否能够将代理程序连接到受监视的 SAP 系统。

在 GUI 上输入参数以完成测试连接过程。如果成功连接到 SAP 系统，那么会显示成功消息。或者，如果连接失败，那么会显示失败消息。

测试连接按钮仅在 IBM Application Performance Management 窗口中可用。

启用 CCMS 设计

启用计算中心管理系统 (CCMS) 监视以从最新采样周期收集处于打开或关闭状态的 CCMS 记录。您可以配置采样周期，缺省情况下，其具有值 3 分钟。但是，必须确保 SAP 代理程序引用的传输文件和高级业务应用程序编程 (ABAP) 传输是相同版本。

过程

1. 登录到 SAP 系统。
2. 打开 SE16 事务并向事务添加 /IBMMON/ITM_CNFG 表名称。
3. 要运行 /IBMMON/ITM_CNFG ABAP 功能模块并针对 ABAP 程序提供配置，请按 **Enter** 键，然后按 **F8**。
4. 要创建向其添加新配置参数的新条目，请按 **F5**。
5. 要使用值 *YES* 在 SAP 服务器上创建名为 **ISNEWCCMSDESIGN** 的新配置参数，请在 **PARM NAME** 字段中，输入 ISNEWCCMSDESIGN，并在 **VALUE CHAR** 字段中，输入 YES。
6. 单击**保存**。
   您可以忽略 VALUE INT 字段。

修改警报的阈值

您可以修改与警报关联的**最大 ccms 警报**阈值。缺省情况下，值为 1000，意味着您可以在 IBM Application Performance Management 中查看 1000 个警报。系统会从高速缓存中移除较旧的警报

过程

1. 完成下列其中一个步骤：

- 在 Windows 操作系统上，打开 <cancle home>\tmaitm6\KSAENV 文件。
- 在非 Windows 操作系统上，打开 <candle home>/config/sa.ini 文件。

2. 将 *MAX_CCMS_ALERT_THRESHOLD=< Value>* 添加到文件末尾。

   注: 值必须大于 100。

禁用 CCMS 设计
您可以禁用计算中心管理系统 (CCMS) 设计。

过程

1. 登录到 SAP 系统。
2. 打开 SE16 事务并向事务添加 /IBMMON/ITM_CNFG 表名称。
3. 要运行 /IBMMON/ITM_CNFG ABAP 功能模块并针对 ABAP 程序提供配置，请按 **Enter** 键，然后按 **F8**。
4. 要删除现有条目，请选择并右键单击 **ISNEWCCMSDESIGN**，然后单击**删除**。

配置 SAP HANA 数据库监视

您必须配置 SAP HANA Database 代理程序，以便代理程序可收集正受监视的 SAP HANA 数据库服务器的数据。

开始之前

复审必备硬件和软件，请参阅 SAP HANA 数据库代理程序的软件产品兼容性报告

以下是配置 SAP HANA Database 代理程序之前的先决条件

1. 确保在 SAP HANA 系统的所有数据库（系统和租户）中创建具有以下特权的用户：
   - 角色：监视
   - 系统权限：监视管理

   系统数据库和租户数据库的用户名和密码必须相同。
2. 在 SAP HANA Database 代理程序系统上发生主连接到备用连接的切换时，代理程序使用需要在代理程序系统上解析的备用服务器的主机名。要将主机名解析为 IP 地址，需要在安装了代理程序的机器的主机文件中添加映射条目。

注: 如果使用主机来配置代理程序，请输入主机的的标准主机名或 IP 地址。如果用户使用备用主机来配置代理程序，请输入备用主机的标准主机名或 IP 地址。通过备用节点来配置代理程序时，主节点以及主机必须关闭。

关于此任务
SAP HANA Database 代理程序是多实例代理程序。必须手动创建第一个实例并启动代理程序。

过程

- Windows 要在 Windows 系统上配置代理程序，请完成下列步骤：
  a) 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
  b) 在 **IBM Performance Management** 窗口的**任务/子系统**列下，右键单击**模板**，然后单击**使用缺省值配置**。
  这将打开“**Monitoring Agent for SAP HANA 数据库**”窗口。
  c) 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。

  要点: 代理程序实例名称必须与 3 位数的 HANA 数据库系统标识 (SID) 相一致。例如，如果受管 SAP HANA 数据库的 SID 为 H01，请输入 H01 作为实例名称。
  d) 在“**Monitoring Agent for SAP HANA 数据库**”窗口中，指定以下字段的值：

实例名称
此字段中的缺省值与**输入唯一实例名称**字段中指定的值相同。

服务器名称
安装了系统数据库的 SAP HANA 服务器的标准主机名称或 IP 地址。

数据库名称
SAP HANA 数据库名称。

端口号
SAP HANA 数据库服务器系统数据库上的索引服务器服务的 SQL 端口号。

HANA DB 管理员
用于访问 SAP HANA 数据库服务器的用户名。

HANA DB 管理员密码
用于访问 SAP HANA 数据库服务器的密码。

确认 HANA DB 管理员密码
HANA DB 管理员密码字段中指定的密码。

e) 单击**确定**。

f) 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

- Linux AIX 要在 Linux 或 AIX 系统上配置代理程序，请完成下列步骤：

a) 在命令行中，更改代理程序安装目录的路径。
示例：`/opt/ibm/apm/agent/bin`

b) 运行以下命令，其中 instance_name 是要对实例指定的名称：
`./sap_hana_database-agent.sh config` *`instance_name`*

要点: 实例名称必须与 3 位数的 HANA 数据库系统标识 (SID) 相一致。如果受管 SAP HANA 数据库的 SID 为 H01，请输入 H01 作为实例名称。

c) 命令行显示以下消息时，请输入 1，然后按 Enter 键：`Edit 'Monitoring Agent for SAP HANA Database' setting? [1=Yes, 2=No]`

d) 指定以下代理程序参数的值：

服务器名称
安装了系统数据库的 SAP HANA 服务器的标准主机名称或 IP 地址。

数据库名称
SAP HANA 数据库名称。

端口号
SAP HANA 数据库服务器系统数据库上的索引服务器服务的 SQL 端口号。

HANA DB 管理员
用于访问 SAP HANA 数据库服务器的用户名。

HANA DB 管理员密码
用于访问 SAP HANA 数据库服务器的密码。

确认 HANA DB 管理员密码
HANA DB 管理员密码字段中指定的密码。

e) 运行以下命令以启动 SAP HANA Database 代理程序 代理程序：
`./sap_hana_database-agent.sh start` *`instance_name`*

- 要使用静默响应文件配置代理程序，请完成以下步骤：

a) 在文本编辑器中，打开 *`install_dir`*`\samples` 路径下的 `sap_hana_silent_config.txt` 文件，并指定所有参数值。

Windows `C:\IBM\APM\samples`

Linux AIX `/opt/ibm/apm/agent/samples`

b) 在命令行中，将路径更改为 *`install_dir`*`\bin`

c) 运行以下命令：

Windows `sap_hana_database-agent.bat config *instance_name* *install_dir*\samples\sap_hana_silent_config.txt`

Linux AIX `sap_hana_database-agent.sh config *instance_name* *install_dir*\samples\sap_hana_silent_config.txt`

d) 启动代理程序。

Windows 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

Linux AIX 运行以下命令：`./sap_hana_database-agent.sh start *instance_name*`

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

配置 SAP NetWeaver Java Stack 监视

您必须配置 SAP NetWeaver Java Stack 代理程序，以便代理程序可收集正受监视的 SAP NetWeaver Application Server 的资源监视数据。要监视事务跟踪和诊断数据，必须完成一些配置任务。

开始之前

复审必备硬件和软件，请参阅 SAP NetWeaver Java Stack 代理程序的软件产品兼容性报告

请确保先完成下列先决条件任务，然后再配置代理程序：

- 将以下 JAR 文件复制到 bin 目录：
  - sapj2eeclient.jar（包含 JMX 适配器的 SAP J2EE 引擎客户机 API）
  - logging.jar（日志记录库）
  - com_sap_pj_jmx.jar（SAP-JMX 库）
  - exception.jar（SAP 异常框架）

  bin 目录位于以下路径：

  Windows `candle_home\TMAITM6_x64`

  Linux `candle_home/interp/sv/bin`

  要点： 对于所有支持的操作系统，JAR 文件均相同。这些文件可从诊断代理程序补丁或软件更新管理器 (SUM) 中获取。
- 在“环境变量”中，将 *<candleHome>\svdchome\<build number>\toolkit\lib\win64\ttapi* 添加到路径变量中。
- 将 NWA_READONLY 角色分配给访客用户，以便收集事务跟踪和诊断数据。

关于此任务

SAP NetWeaver Java Stack 代理程序是多实例代理程序。必须手动创建第一个实例并启动代理程序。

- 要在 Windows 系统上配置此代理程序，您可以使用 GUI 或静默响应文件。
- 要在 Linux 或 AIX 系统上配置此代理程序，您可使用命令行或静默响应文件。

要配置事务跟踪和诊断数据的收集，请完成下列任务：

1. 配置数据收集器。有关详细信息，请参阅第 653 页的『配置数据收集器』。
2. 启用事务跟踪和诊断数据的收集。有关详细信息，请参阅第 654 页的『启用事务跟踪和诊断数据收集』。

除非另有说明，否则本主题中所述的指示信息适用于代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅代理程序版本。

在 Windows 系统上配置代理程序

可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置代理程序。

开始之前

确保第 650 页的『配置 SAP NetWeaver Java Stack 监视』主题的“准备工作”一节中所列的文件位于 bin 目录中。

关于此任务

SAP NetWeaver Java Stack 代理程序会提供一些参数的缺省值。您可以为这些参数指定其他值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口的**任务/子系统**列下，右键单击**模板**，然后单击**配置代理程序**。
   这将打开 **Monitoring Agent for SAP NetWeaver Java Stack** 窗口。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。

   要点: 代理程序实例名称必须与 3 位数的 SAP NetWeaver Java Stack 系统标识 (SID) 相匹配。例如，如果受管 SAP NetWeaver Java Stack 的 SID 为 PS1，请输入 PS1 作为实例名称。
4. 在 **Monitoring Agent for SAP NetWeaver Java Stack** 窗口中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 656 页的『代理程序的配置参数』。
5. 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**。

下一步做什么

- 登录 Cloud APM 控制台，以在仪表板中查看代理程序收集的资源监视数据。有关使用 Cloud APM 控制台的信息，请参阅启动 Performance Management 控制台。
- 要收集事务跟踪和诊断数据，请配置数据收集器并启用事务跟踪和诊断数据收集。

在 Linux 或 AIX 系统上配置代理程序

要在 Linux 或 AIX 操作系统上配置此代理程序，您必须运行脚本并对提示做出响应。

开始之前

确保第 650 页的『配置 SAP NetWeaver Java Stack 监视』主题的“准备工作”一节中所列的文件位于 bin 目录中。

过程

1. 在命令行中，更改代理程序安装目录的路径。

   Linux `/opt/ibm/apm/agent/bin`

   Linux AIX `/opt/ibm/apm/agent/bin`
2. 运行以下命令:

   ```
   ./sap_netweaver_java_stack-agent.sh config instance_name
   ```

   其中，*instance_name* 是要为实例指定的名称。

要点: 代理程序实例名称必须与 3 位数的 SAP NetWeaver Java Stack 系统标识 (SID) 相匹配。例如，如果受管 SAP NetWeaver Java Stack 的 SID 为 PS1，请输入 PS1 作为实例名称。

3. 当命令行显示以下消息时，请输入 1 并按 Enter 键：

   ```
   Edit 'Monitoring Agent for SAP NetWeaver Java Stack' setting? [1=Yes, 2=No]
   ```

4. 出现提示时，为配置参数指定值。

   有关配置参数的信息，请参阅第 656 页的『代理程序的配置参数』

5. 运行以下命令以启动代理程序：

   ```
   ./sap_netweaver_java_stack-agent.sh start instance_name
   ```

下一步做什么

- 登录 Cloud APM 控制台，以在仪表板中查看代理程序收集的资源监视数据。有关使用 Cloud APM 控制台的信息，请参阅启动 Performance Management 控制台。
- 要收集事务跟踪和诊断数据，请配置数据收集器并启用事务跟踪和诊断数据收集。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

开始之前

确保第 650 页的『配置 SAP NetWeaver Java Stack 监视』主题的“准备工作”一节中所列的文件位于 bin 目录中。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

1. 在文本编辑器中，打开以下路径中的 sap_netweaver_java_stack_silent_config.txt 文件，并指定配置参数的值。

   Windows `C:\IBM\APM\samples`

   Linux AIX `/opt/ibm/apm/agent/samples`

   有关配置参数的信息，请参阅第 656 页的『代理程序的配置参数』

2. 在命令行中，将路径切换到 install_dir\bin

3. 运行以下命令：

   Windows
   ```
   sap_netweaver_java_stack-agent.bat config instance_name
   install_dir\samples\sap_netweaver_java_stack_silent_config.txt
   ```

   Linux AIX
   ```
   ./sap_netweaver_java_stack-agent.sh config instance_name
   install_dir\samples\sap_netweaver_java_stack_silent_config.txt
   ```

4. 启动代理程序。

   Windows 在 IBM Cloud Application Performance Management 窗口中，右键单击您所创建的代理程序实例，然后单击**启动**。另外，也可以运行以下命令：`sap_netweaver_java_stack-agent.bat start instance_name`

   Linux AIX 运行以下命令：`./sap_netweaver_java_stack-agent.sh start instance_name`

下一步做什么

- 登录 Cloud APM 控制台，以在仪表板中查看代理程序收集的资源监视数据。有关使用 Cloud APM 控制台的信息，请参阅启动 Performance Management 控制台。
- 要收集事务跟踪和诊断数据，请配置数据收集器并启用事务跟踪和诊断数据收集。

配置数据收集器

您可以针对所要监视的每一个应用程序服务器实例来配置数据收集器。

开始之前

确保第 650 页的『配置 SAP NetWeaver Java Stack 监视』主题的“准备工作”一节中所列的文件位于 bin 目录中。

过程

要通过对提示做出响应来配置数据收集器，请完成下列步骤：

1. 在命令行上，将路径切换到 Windows `install_dir\svdchome\build no\bin\configNW` 或 Linux AIX `install_dir/svdchome/build no/bin/configNW` 目录，并运行下列脚本：

   Windows config.bat

   Linux AIX config.sh
2. 通过输入对应于要配置数据收集器的产品的编号，选择 NetWeaver Server 版本，然后按 Enter 键。
3. 提示输入用户名时，请输入在 SAP NetWeaver Application Server with Java Stack 上配置的用户名，然后按 Enter 键。
4. 提示输入密码时，请输入密码，然后按 Enter 键。
5. 提示重新输入密码时，请再次输入密码，然后按 Enter 键。
6. 提示输入 P4 端口号时，请输入本地计算机上 SAP NetWeaver Application Server 实例的 P4 端口号，然后按 Enter 键。

   要点: 请使用以下公式来计算 P4 端口号：50000 + (实例数*100) + 4
7. 提示选择 NetWeaver Server 实例编号时，请输入对应于所要配置的实例的编号，然后按 Enter 键。

   切记: 对于每个实例，您需要单独配置数据收集器。
8. 如果提示您输入 Java 主目录的路径，请使用来自 SAP 实例的 JAVA_HOME。例如，`E:\usr\sap\J01\J04\exe\sapjvm_6`。
9. 出现提示时，如果要启用事务跟踪数据收集，请输入 1。否则，输入 2，并按 Enter 键。
10. 出现提示时，如果要启用诊断数据收集，请输入 1。否则，输入 2，并按 Enter 键。

结果

此时将生成用于装入类文件的路径。

下一步做什么

1. 将生成的路径添加到相应的环境变量。

   Windows *PATH*

   Linux *LD_LIBRARY_PATH* 和 *LIBPATH*

   AIX *LD LIBRARY PATH* 和 *LIB PATH*

   切记:

   Windows 将生成的路径添加到 *PATH* 环境变量。

   Linux 将生成的路径以下列格式添加到 `/home/sidadm/.cshrc` 文件中的 *LD_LIBRARY_PATH* 和 *LIBPATH*。

```
setenv LD_LIBRARY_PATH /path
setenv LIBPATH /path
```

AIX 将生成的路径以下列格式添加到 `/etc/environment` 文件中的 *LD LIBRARY PATH* 和 *LIB PATH*。

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path
LIBPATH=$LIBPATH:/path
```

2. 重新启动应用程序服务器实例。
3. 启用事务跟踪及诊断数据收集。有关详细信息，请参阅第 654 页的『启用事务跟踪和诊断数据收集』。

启用事务跟踪和诊断数据收集

在“**代理程序配置**”页面上，可以启用或禁用事务跟踪或诊断数据收集。

开始之前

确保配置数据收集器。有关详细信息，请参阅第 653 页的『配置数据收集器』。

关于此任务

启用事务跟踪数据收集后，代理程序将收集下列组件的数据：

- Servlet JSP
- RemoteEJB
- JMS

过程

请完成以下步骤，以便为每个 SAP NetWeaver Application Server 实例配置数据收集。

1. 登录到 Cloud APM 控制台。
2. 从导航栏，单击 **系统配置** > **代理程序配置**。
   “**代理程序配置**”页面随即显示。
3. 单击 **NetWeaver** 选项卡。
4. 请选中要配置数据收集的 SAP NetWeaver Application Server 实例的相应复选框，并从**操作**列表完成下列任何操作。
   - 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。对于每个选中的 SAP NetWeaver Application Server 实例，**事务跟踪**列中的状态将更新为“已启用”。
   - 要启用诊断数据收集，请选择**设置诊断方式** > **仅已启用诊断方式**。对于每个选中的 SAP NetWeaver Application Server 实例，“诊断方式”列中的状态将更新为“已启用”。
   - 要启用诊断数据收集和方法跟踪，请选择**设置诊断方式** > **已启用诊断方式和方法跟踪**。对于每个选中的 SAP NetWeaver Application Server 实例，“诊断方式”和“方法跟踪”列中的状态将更新为“已启用”。
   - 要禁用事务跟踪，请单击**设置事务跟踪** > **已禁用**。对于每个选中的 SAP NetWeaver Application Server 实例，**事务跟踪**列中的状态将更新为“已禁用”。
   - 要禁用诊断数据收集，请单击**设置诊断方式** > **已禁用诊断方式和方法跟踪**。对于每个选中的 SAP NetWeaver Application Server 实例，**“诊断方式”和“方法跟踪”**列中的状态将更新为“已禁用”。

结果

您已经为每个 SAP NetWeaver Application Server 实例配置数据收集。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的事务跟踪和诊断数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

移除数据收集器配置

您可回滚您为 SAP Netweaver Application Server with Java Stack 实例配置数据收集器时进行的更改。

过程

要通过对提示做出响应来移除数据收集器配置，请完成下列步骤：

1. 在命令行上，将路径切换到 Windows `install_dir\svdchome\build no\bin\configNW` 或 Linux AIX `install_dir/svdchome/build no/bin/configNW` 目录，并运行下列脚本：

   Windows unconfig.bat

   Linux AIX unconfig.sh

   所有已配置有数据收集器的实例都会列出。
2. 输入要移除数据收集器配置的实例的相应编号，然后按 Enter 键。

   提示: 要移除多个实例的数据收集器配置，请输入对应于这些实例的编号，并以逗号进行分隔。要移除所有实例的数据收集器配置，您可执行下列脚本：

   Windows config.bat -a

   Linux AIX config.sh -a

下一步做什么

重新启动 SAP NetWeaver AS with Java Stack 实例。

复原 SAP NetWeaver Application Server 实例

使用复原实用程序可在配置 SAP NetWeaver 数据收集器后 SAP NetWeaver Application Server 实例无法启动时复原 JVM 参数，或者复原 SAP NetWeaver Application Server 实例。

过程

要通过响应提示来复原 SAP NetWeaver Application Server 实例，请完成下列步骤：

1. 在命令行上，将路径切换到 Windows `install_dir\svdchome\build no\bin\configNW` 或 Linux AIX `install_dir/svdchome/build no/bin/configNW` 目录，并运行下列脚本：

   Windows `restoreNW.bat`

   Linux AIX `restoreNW.sh`
2. 通过输入对应于要复原 JVM 参数的产品的编号并按 Enter 键，选择 NetWeaver Server 版本。
3. 提示输入用户名时，请输入 SAP NetWeaver Application Server 实例的用户名，并按 Enter 键。
4. 提示输入用户密码时，请输入 SAP NetWeaver Application Server 实例的用户密码，并按 Enter 键。
5. 提示输入 P4 端口号时，请输入本地计算机上 SAP NetWeaver Application Server 实例的 P4 端口号，然后按 Enter 键。

   如果使用所输入的 P4 端口找不到实例信息，系统会显示`无法连接至 SAP NetWeaver Server` 消息，并提示您提供 NetWeaver Server 实例主目录的路径。

   例如，`usr\sap\System_Name\Instance_Number`
6. 提示选择 NetWeaver Server 实例号时，请输入对应于所要复原的实例的编号，并按 Enter 键。

结果

此时将显示以下消息：

复原成功。请重新启动此实例。

代理程序的配置参数

配置 SAP NetWeaver Java Stack 代理程序 时，您可以更改参数的缺省值，例如，SAP_NETWEAVER_P4_HOSTNAME 和 SAP_NETWEAVER_P4_PORT。

下表包含 SAP NetWeaver Java Stack 代理程序 的配置参数的详细描述。您必须对所有字段指定值，因为这些字段必填。

表 *207*. 配置参数的名称和描述

参数名称	**描述**
实例名称	实例的名称。此字段中的缺省值与**输入唯一实例名称**字段中指定的值相同。
SAP_NETWEAVER_P4_HOSTNAME	SAP NetWeaver Application Server 的主机名或 IP 地址。
SAP_NETWEAVER_P4_PORT	SAP NetWeaver Application Server 的 P4 端口号。
SAP_NETWEAVER_P4_USERNAME	用于访问 SAP NetWeaver Application Server 的管理员的用户名。
SAP_NETWEAVER_P4_PASSWORD	用于访问 SAP NetWeaver Application Server 的管理员的密码。
Confirm SAP_NETWEAVER_P4_PASSWORD	为 **SAP_NETWEAVER_P4_PASSWORD** 参数指定的密码。

配置 Siebel 监视

Siebel 代理程序提供针对 Siebel 资源的中央监视点，其中包括 Siebel 统计信息、用户会话、组件、任务、应用程序服务器、Siebel Gateway Name Server、进程 CPU 和内存使用情况以及日志事件监视。

开始之前

- 请通读第 656 页的『配置 Siebel 监视』主题，确定完成配置所需的内容。
- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 Siebel 代理程序的系统需求。有关最新系统需求信息，请参阅 Siebel 代理程序的软件产品兼容性报告 (SPCR)。
- 在配置 Siebel 代理程序之前，您必须验证 Siebel 用户帐户，以供 Siebel 代理程序使用。

缺省情况下，“按组件统计信息监视”处于停用状态。您可启用“按组件统计信息监视”。

关于此任务

Siebel 代理程序 是多实例代理程序。必须创建第一个实例，然后手动启动代理程序。

过程

1. 要在 Windows 系统上配置此代理程序，您可以使用 IBM Performance Management 窗口或静默响应文件。
   - 第 658 页的『在 Windows 系统上配置代理程序』.
   - 第 663 页的『使用静默响应文件来配置代理程序』.
2. 要在 Linux 和 UNIX 系统上配置代理程序，您可以运行脚本并对提示做出响应，或者使用静默响应文件。
   - 第 662 页的『通过对提示做出响应来配置代理程序』.
   - 第 663 页的『使用静默响应文件来配置代理程序』.

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板页面以查看收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。这些日志的缺省路径如下所示：

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

验证 Siebel 用户帐户

您必须先验证用来运行 Siebel 代理程序的用户帐户，然后才能配置此代理程序。

关于此任务

用来运行 Siebel 代理程序的用户帐户必须有权运行 Siebel **srvrmgr** 命令行实用程序。要确认此用户帐户具有所需的许可权，请执行下列步骤：

过程

1. 以用来运行 Siebel 代理程序的用户帐户身份登录计算机。
2. 将目录切换到 Siebel 服务器的安装位置。
3. 读取 Siebel 环境文件：
   source siebenv.sh
4. 运行以下命令：

```
srvrmgr /s Siebel_server /g Siebel_gateway /e Siebel_enterprise
/u useraccount /p password
/c "list servers"
```

其中

Siebel_server
Siebel 应用程序服务器的名称。

Siebel_gateway
当前活动 Gateway Name Server 的名称。

Siebel_enterprise
Siebel Enterprise 的名称。

useraccount
用于登录计算机的用户帐户。

密码
与用户帐户关联的密码。

如果该用户帐户具有所需的许可权，那么您会看到类似于以下示例的输出，其中返回的字段限制为 3 个：

```
Connected to 1 server(s) out of a total of 1 server(s) in the enterprise

srvrmgr:s82win8> list servers show SBLSRVR_NAME, HOST_NAME, SBLSRVR_STATUS

SBLSRVR_NAME  HOST_NAME  SBLSRVR_STATUS
------------  ---------  --------------------------------
s82win8       s82win8    16.0.0.0 [23057] LANG_INDEPENDENT

1 row returned.
```

如果 **srvrmgr** 命令没有正确运行，请咨询服务器的 Siebel 管理员。请确保为该用户帐户设置必需的 Siebel 环境变量，并确保该用户帐户具有适当的许可权来运行 **srvrmgr** 命令。

启用“按组件统计信息监视”

缺省情况下，“按组件统计信息监视”处于停用状态。您可使用 KUY_ENABLE_COMP_STATS 环境变量来启用“按组件统计信息监视”。

开始之前

因为 Siebel V8.1 及更高版本的服务器有一个已知问题，收集 Siebel 组件统计信息可能会对 Siebel Gateway Server 的内存用量产生负面影响。这个问题已在 Oracle 发布的名为“Gateway Service on Siebel 8.1 或 8.2 Might Consume High Memory Consumption: Recovery（文档标识 1269177.1）”的技术说明中解决。在该文章中，已提供针对此问题的修订。该修订在 Siebel 服务器上实现。

如果环境中需要“按组件统计信息监视”，请先将该 Oracle 修订应用于 Siebel V8.1 和更高版本的网关服务器，然后再启用“按组件统计信息监视”。

关于此任务

应用该 Oracle 修订之后，请完成下列步骤，以便在 Siebel 代理程序中启用“按组件统计信息监视”：

过程

1. 转至 Siebel 代理程序的代理程序安装目录：
   - Windows *install_dir*\TMAITM6_x64
   - Linux AIX *install_dir*/config
2. 编辑 Siebel 代理程序配置文件，将 KUY_ENABLE_COMP_STATS 设置为 true。
   - Windows KUYENV_*instance_name*
   - Linux AIX `uy.environment`

   其中，*instance_name* 是 Siebel 代理程序的实例名称。
3. 重新启动代理程序。

   要点: 要使此设置成为所有新代理程序实例的缺省设置，请在配置模板文件中，将 KUY_ENABLE_COMP_STATS 设置为 true：
   - Windows KUYENV
   - Linux AIX 通过编辑步骤 2 中的 `uy.environment`，此设置已成为所有新代理程序实例的缺省设置。

在 Windows 系统上配置代理程序

通过使用 IBM Cloud Application Performance Management 窗口，您可以在 Windows 操作系统上配置 Siebel 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Siebel** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，**配置代理程序**选项处于禁用状态。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用字母、阿拉伯数字、下划线字符和减号字符。例如：`siebel01`.

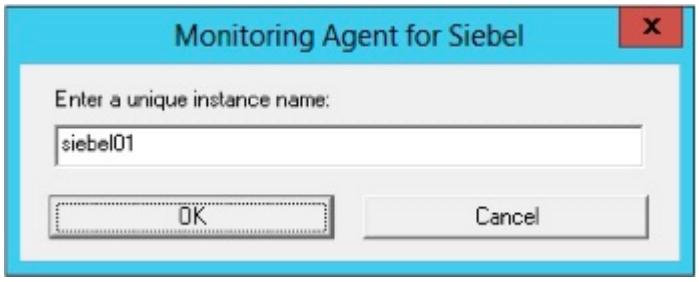


图 22. 用于输入唯一实例名称的窗口。

4. 选择服务器类型，为该服务器类型输入必填字段的值，然后单击**下一步**。
   请参阅第 664 页的表 208，以获取每个配置参数的说明。

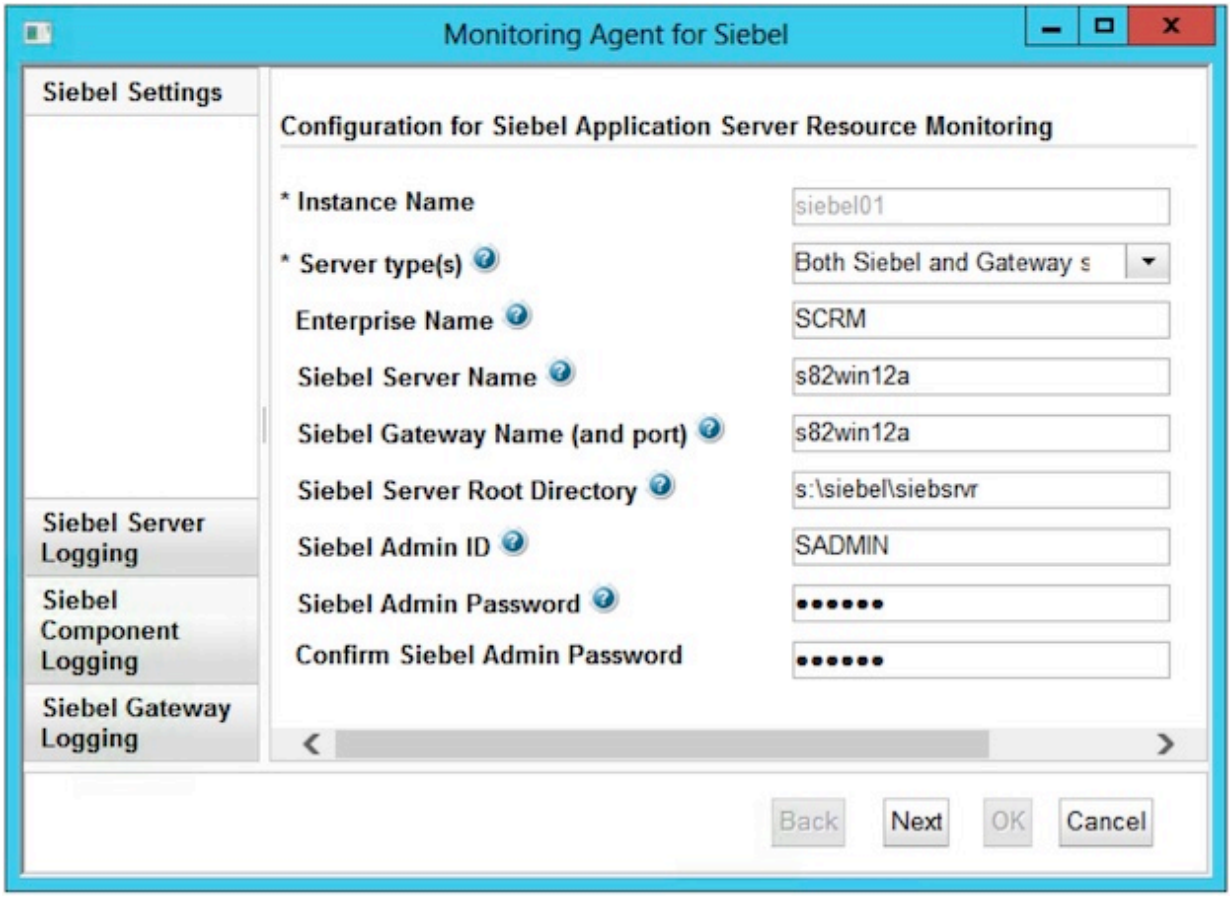


图 23. 安装在 *Siebel* 主机上的 *Siebel* 服务器类型的配置参数窗口

要点: 如果 Siebel 代理程序是安装在带有 Siebel Gateway Name Server 但没有 Siebel 服务器的计算机上，那么应用程序仪表板中显示的数据仅适用于此实例的 Siebel Gateway Name Server。所有其他 Siebel 代理程序视图均为空。

5. 可选： 编辑 Siebel 服务器日志记录值，然后单击**下一步**。
   请参阅第 665 页的表 209，以获取每个配置参数的说明。

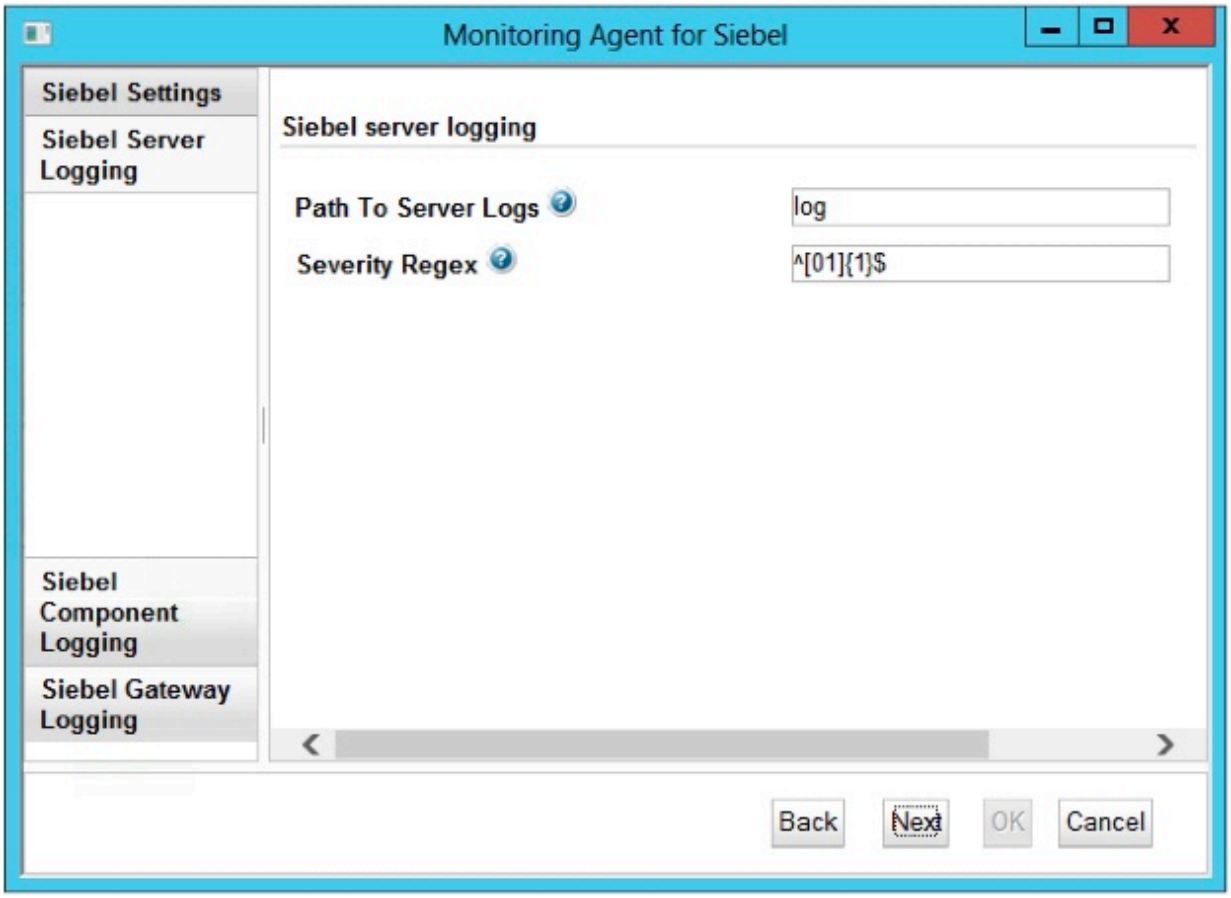


图 *24.* 用于指定 *Siebel* 服务器日志记录设置的窗口。

6. 可选： 编辑 Siebel 组件日志记录值，然后单击**下一步**。缺省情况下，第 666 页的表 212 中组件的日志受 Siebel 代理程序监视。要添加最多 10 个所要监视的其他组件日志，请指定相应的组件别名，例如，`SCBroker`。

   请参阅第 665 页的表 210，以获取每个配置参数的说明。

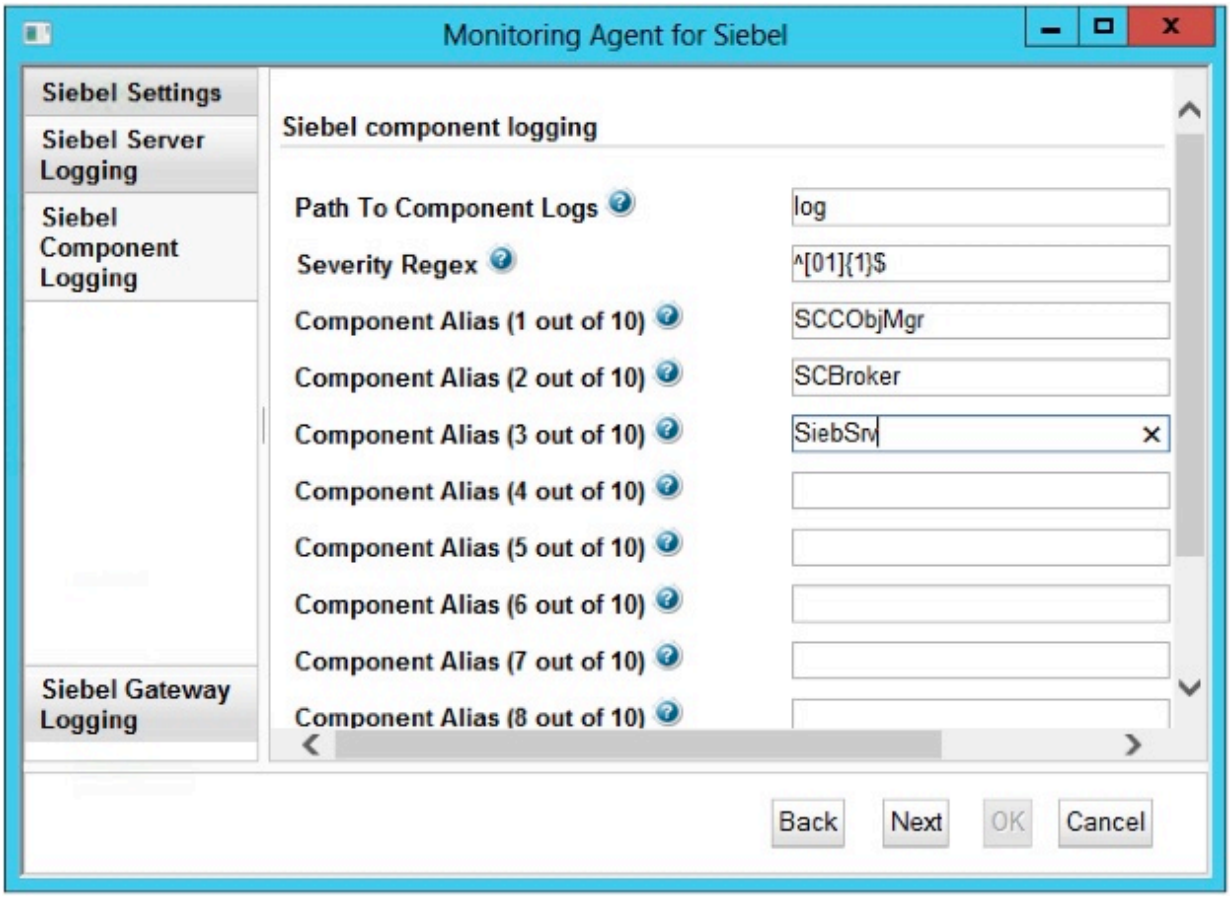


图 *25.* 用于指定所要监视的额外组件日志的窗口。

7. 可选： 编辑 Siebel 网关日志记录值。
   请参阅第 665 页的表 211，以获取每个配置参数的说明。

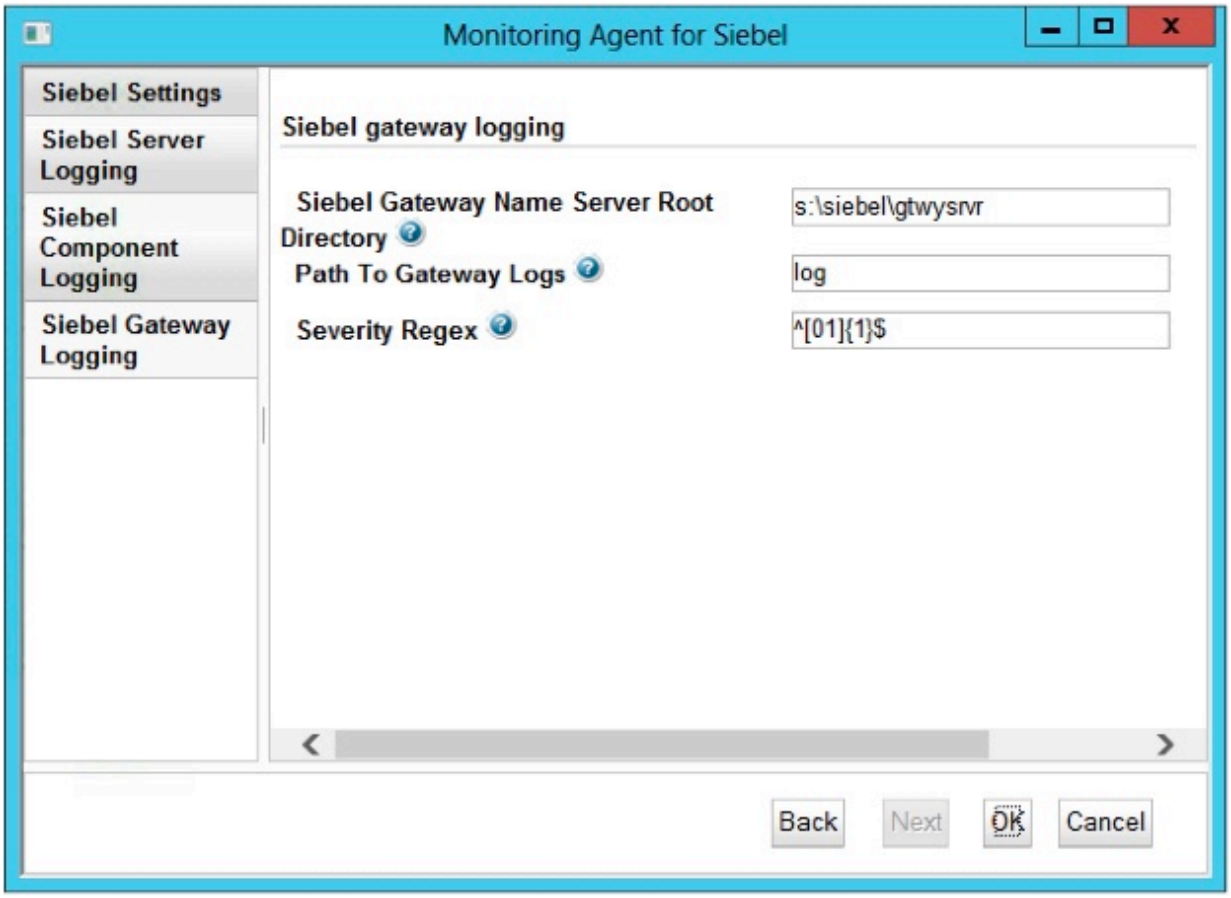


图 *26.* 用于指定 *Siebel* 网关日志记录设置的窗口。

8. 单击**确定**以完成配置。
9. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过对提示做出响应来配置代理程序

安装 Siebel 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 Siebel 代理程序是安装在本地 Linux 或 UNIX 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

· 完成下列步骤，通过运行脚本并对提示做出响应来配置 Siebel 代理程序。

a) 在命令行上，运行以下命令：

install_dir/bin/siebel-agent.sh config **instance_name**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/siebel-agent.sh config example-inst01
```

b) 对提示做出响应，以便为代理程序设置配置值。

请参阅第 664 页的『Siebel 代理程序的配置参数』，以获取每个配置参数的说明。

c) 运行以下命令以启动代理程序：

install_dir/bin/siebel-agent.sh start **instance_name**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

示例

```
/opt/ibm/apm/agent/bin/siebel-agent.sh start example-inst01
```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

· 要以静默方式配置 Siebel 代理程序，请完成下列步骤：

a) 在文本编辑器中，打开位于以下路径的 siebel_silent_config.txt 文件：

– Linux AIX *install_dir*/samples/siebel_silent_config.txt

– Windows *install_dir*\samples\siebel_silent_config.txt

其中，*install_dir* 是安装代理程序的路径。

示例

– Linux AIX /opt/ibm/apm/agent/samples/siebel_silent_config.txt

– Windows C:\IBM\APM\samples\siebel_silent_config.txt

b) 在 siebel_silent_config.txt 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

请参阅第 664 页的『Siebel 代理程序的配置参数』，以获取每个配置参数的说明。

c) 保存并关闭 siebel_silent_config.txt 文件，然后运行下列命令：

– Linux AIX *install_dir*/bin/siebel-agent.sh config **instance_name** *install_dir*/samples/siebel_silent_config.txt

– Windows *install_dir*\bin\siebel-agent.bat config **instance_name** *install_dir*\samples\siebel_silent_config.txt

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

示例

– Linux AIX /opt/ibm/apm/agent/bin/siebel-agent.sh config example-inst01 /opt/ibm/apm/agent/samples/siebel_silent_config.txt

– Windows C:\IBM\APM\bin\ siebel-agent.bat config example-inst01 C:\IBM\APM\samples\siebel_silent_config.txt

d) 运行以下命令以启动代理程序：

– Linux AIX *install_dir*/bin/siebel-agent.sh start **instance_name**

– Windows *install_dir*\bin\siebel-agent.bat start **instance_name**

其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

示例

- Linux AIX `/opt/ibm/apm/agent/bin/siebel-agent.sh start example-inst01`
- Windows `C:\IBM\APM\bin\siebel-agent.bat start example-inst01`

Siebel 代理程序的配置参数

下列各表显示 Siebel 代理程序的配置参数，并根据类别进行分组。

1. Siebel 设置 - 一般 Siebel 环境设置。
2. Siebel 服务器日志记录 - 专用于监视 Siebel 服务器日志的设置。
3. Siebel 组件日志记录 - 专用于监视 Siebel 组件日志的定制列表的设置。
4. Siebel 网关日志记录 - 专用于监视 Siebel 网关日志的设置。

表 *208. Siebel* 设置

参数名称	描述	需要此参数的服务器类型选项	静默配置文件参数名称
服务器类型	指示本地计算机上安装的服务器类型。	· **仅限 Gateway 服务器** · **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_SERVER_TYPE**
Enterprise 名称	Siebel Enterprise 的名称。	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_ENTERPRISE**
Siebel 服务器名称	要监视的 Siebel 服务器的名称。 **注:** 此名称并非服务器的主机名。这是运行 Siebel **srvrmgr** 命令时使用的服务器名称。	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_SERVER**
Siebel 网关名称	要监视的 Siebel Gateway Name Server，并选择性地指定端口，例如，`gtwysrvr` 或 `gtwysrvr:1234`。	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_GATEWAY**
Siebel 服务器根目录	Siebel Application Server 的基本安装目录。	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_INSTALL_ROOT**
Siebel 管理员标识	运行 **srvrmgr** 命令时，代理程序向 Siebel Enterprise 进行认证所用的 Siebel 特定用户标识。例如：`SADMIN`	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_ADMIN_ID**

表 *208. Siebel* 设置 *(续)*

参数名称	描述	需要此参数的服务器类型选项	静默配置文件参数名称
Siebel 管理员密码	Siebel 服务器管理员的密码。	· **仅限 Siebel 服务器** · **Siebel 和 Gateway 服务器**	**KUY_ADMIN_PASSWORD**

表 *209. Siebel* 服务器日志记录设置

参数名称	描述	静默配置文件参数名称
服务器日志路径	从“Siebel 服务器根目录”到服务器日志的相对路径。要禁止捕获 Siebel 服务器日志，请输入任何无效路径。例如：`xyz`。	**KUY_SERVER_LOGGING_PATH**
严重性正则表达式	用来捕获与严重性级别匹配的 Siebel 服务器日志的正则表达式。使用缺省值 `^[01]{1}$` 有助于捕获第 0 级和第 1 级错误。	**KUY_SERVER_LOGGING_SEVERITY_REGEX**

表 *210. Siebel* 组件日志记录设置

参数名称	描述	静默配置文件参数名称
组件日志路径	从“Siebel 服务器根目录”到服务器日志的相对路径。要禁止捕获 Siebel 服务器日志，请输入任何无效路径。例如：`xyz`。	**KUY_COMPONENT_LOGGING_PATH**
严重性正则表达式	用来捕获与严重性级别匹配的 Siebel 服务器日志的正则表达式。使用缺省值 `^[01]{1}$` 有助于捕获第 0 级和第 1 级错误。	**KUY_COMPONENT_LOGGING_SEVERITY_REGEX**
组件别名（第 *N* 个，共 10 个）	要监视其额外组件日志的组件别名。示例：`SCBroker`。其中，*N* 是 1 到 10 个可选组件。	**KUY_CUSTCOMPLOG_00** 到 **KUY_CUSTCOMPLOG_09**

表 *211. Siebel* 网关日志记录设置

参数名称	描述	静默配置文件参数名称
Siebel Gateway Name Server 根目录	Siebel Gateway Name Server 的基本安装目录。	**KUY_GATEWAY_ROOT**
网关日志路径	从“Siebel Gateway Name Server 根目录”到网关日志的相对路径。要禁止捕获 Gateway Name Server 日志，请输入任何无效路径。例如：`xyz`。	**KUY_GW_LOGGING_PATH**
严重性正则表达式	用来捕获与严重性级别匹配的 Siebel 服务器日志的正则表达式。使用缺省值 `^[01]{1}$` 有助于捕获第 0 级和第 1 级错误。	**KUY_GW_LOGGING_SEVERITY_REGEX**

始终受监视的 Siebel 组件日志

有 10 个 Siebel 组件的组件日志始终受监视。

表 212. 组件日志始终受监视的 *Siebel* 组件别名及名称。

组件别名	组件名称
SCCObjMgr	呼叫中心对象管理器
SMObjMgr	市场营销对象管理器
SSEObjMgr	销售对象管理器
CommInboundRcvr	入站通信接收方
CommOutboundMg	出站通信管理器
CommSessionMgr	通信会话管理器
WorkMon	工作流程监视代理程序
WfProcBatchMgr	工作流过程批处理管理器
WfProcMgr	工作流过程管理器
SiebSrvr	Siebel 服务器

配置 Sterling Connect Direct 监视

您必须配置 Sterling Connect Direct 代理程序以便代理程序可以从 Connect Direct 服务器收集数据，以监视 Connect Direct 服务器的文件传输及运行状况的统计信息。

开始之前

查看必备硬件和软件，请参阅 Sterling Connect Direct 代理程序的软件产品兼容性报告。

关于此任务

- 要在 Windows 系统上配置此代理程序，您可以使用 IBM Cloud Application Performance Management 窗口或静默响应文件。
- 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

在 Windows 系统上配置代理程序

您可以在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口配置代理程序。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口的**任务/子系统**列中，右键单击**模板**，然后单击**配置代理程序**。
3. 在**输入唯一实例名称**字段中，输入代理程序实例名称，然后单击**确定**。

   注: 限制代理程序实例名称的长度。最好是限制在 7-10 个字符的范围内。
4. 在 **Monitoring Agent for Sterling Connect Direct** 窗口的“**Connect Direct 服务器详细信息**”选项卡中，指定配置参数的值，然后单击**确定**。

   有关配置参数的信息，请参阅第 668 页的『代理程序的配置参数』。
5. 单击**下一步**。
6. 在“Java 参数”选项卡上，保留缺省值，然后单击**下一步**。
7. 在“Java API 客户机配置”选项卡上，单击**确定**。

8. 在 **IBM Performance Management** 窗口中，右键单击您创建的代理程序实例，然后单击**启动**以启动代理程序。

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

在 Linux 系统上配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

过程

1. 在命令行中，更改代理程序安装目录的路径。
   例如，`/opt/ibm/apm/agent/bin`
2. 运行 **/sterling_connect_direct-agent.sh config instance_name** 命令。

   注: *instance_name* 是要指定给代理程序实例的名称。
3. 命令行会显示以下消息：`Edit 'Monitoring Agent for Sterling Connect Direct' setting? [1=Yes, 2=No]`。
4. 输入 1 以编辑设置。
5. 出现提示时，指定配置参数的值。有关配置参数的信息，请参阅第 668 页的『代理程序的配置参数』。
6. 运行以下命令以启动代理程序：**./sterling_connect_direct-agent.sh start instance_name**

下一步做什么

登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。要了解有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

在 Linux 和 Windows 系统上，您可使用静默响应文件来配置 Monitoring Agent for Sterling Connect Direct。在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

1. 在文本编辑器中打开静默响应文件 *install_dir*`/samples/sterling_connect_direct_silent_config.txt`。
2. 在该文件中输入服务器名称、用户名、密码和安装目录，并保存该文件。
3. 在命令提示符中，转至 *install_dir*`/bin` 并运行以下命令：

   Linux AIX `./sterling_connect_direct-agent.sh config <Instance_name>` *install_dir*`/samples/sterling_connect_direct_silent_config.txt`。

   Windows `./sterling_connect_direct-agent.bat config <Instance_name>` *install_dir*`/samples/sterling_connect_direct_silent_config.txt`。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

代理程序的配置参数

在配置 Monitoring Agent for Sterling Connect Direct 时，您可设置配置参数的值。

下表包含 Monitoring Agent for Sterling Connect Direct 配置参数的详细描述。

表 *213.* 配置参数的名称和描述

参数名称	描述	必填字段
实例名称	此字段中的缺省值与**输入唯一实例名称**字段中指定的值相同。	是
服务器名称	Sterling Connect Direct 服务器的主机名或 IP。	是
服务器端口	Sterling Connect Direct 服务器的端口。对于 Sterling Connect Direct，缺省值为 1363。	是
用户名	用于连接 Sterling Connect Direct 服务器的用户名。	是
密码	用于连接 Sterling Connect Direct 服务器的密码。	是
Java 主目录	Java 安装文件夹的路径。	否
Java 跟踪级别	Java 提供程序所使用的跟踪级别。对于 Sterling Connect Direct，缺省值为 Error。	是
JVM 自变量	此参数可用来指定一列可选的自变量给 Java 虚拟机。	否
外部 JAR 文件的类路径	Java API 数据提供程序所需的那些并非随代理程序一起提供的 JAR 文件的路径。	否

配置 Sterling File Gateway 监视

Monitoring Agent for Sterling File Gateway 使用商家到商家 (B2B) REST API 和文件网关数据库来监视 IBM Sterling File Gateway 应用程序。您必须配置 Sterling File Gateway 代理程序以便代理程序可以从数据源收集数据，以监视 Sterling File Gateway 应用程序的统计信息及运行状况。可以在 Windows 和 Linux 系统上配置此代理程序。

开始之前

- 查看必备硬件和软件，请参阅 Sterling File Gateway 代理程序的软件产品兼容性报告。
- 确保 B2B REST API 安装在文件网关节点上。有关 B2B REST API 安装的更多信息，请参阅第 668 页的『安装 B2B REST API』。

关于此任务

Sterling File Gateway 代理程序是多实例代理程序。您必须手动创建第一个实例并启动代理程序。

- 要在 Windows 系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。
- 要在 Linux 系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

安装 B2B REST API

您可以在 Sterling File Gateway 节点上安装并配置商家到商家 (B2B) REST API。B2B REST API 在 B2B Integrator 安装程序 (V5.2.6.2) 中提供。

过程

1. 浏览到 `<install_dir>/bin` 目录。

   其中，*install_dir* 是 B2B Integrator 的代理程序安装程序目录。

2. 运行以下命令：

   · Linux ./InstallService.sh/*install_dir*/bin/b2bAPIs_10000602.jar

   其中，*<install_dir>* 是介质文件内容解压到的位置。

   · Windows ./InstallService.cmd/*install_dir*/bin/b2bAPIs_10000602.jar

   其中，*<install_dir>* 是 B2B 安装程序文件夹。

在 Windows 系统上配置 Sterling File Gateway 代理程序

在 Windows 操作系统上，您可以使用 **IBM Cloud Application Performance Management** 窗口来配置 Sterling File Gateway 代理程序。更新配置值之后，必须启动代理程序以保存更新的值。

关于此任务

Sterling File Gateway 代理程序会提供一些参数的缺省值。您可以为这些参数指定不同的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Sterling File Gateway**，然后单击**配置代理程序**。

   切记: 首次配置代理程序之后，**配置代理程序**选项不可用。要再次配置代理程序，请单击**重新配置**。
3. 在 Sterling File Gateway 代理程序窗口中，完成下列步骤：

   a) 为 Sterling File Gateway 代理程序实例输入唯一名称，然后单击**确定**。

   b) 在 **B2B API 详细信息**选项卡上，指定配置参数的值，然后单击**下一步**。

   c) 在**数据库详细信息**选项卡上，指定配置参数的值，然后单击**下一步**。

   d) 在 **Java API** 选项卡上，指定配置参数的值，然后单击**确定**。

   有关 Sterling File Gateway 代理程序窗口的每个选项卡中配置参数的更多信息，请参阅以下主题：

   · 第 672 页的『B2B API 详细信息的配置参数』

   · 第 673 页的『数据库详细信息的配置参数』

   · 第 673 页的『Java API 的配置参数』
4. 在 **IBM Performance Management** 窗口中，右键单击 **Sterling File Gateway 代理程序**，然后单击**启动**。

在 Linux 系统上配置 Sterling File Gateway 代理程序

在 Linux 操作系统上，您可运行配置脚本并对提示做出响应，以配置 Sterling File Gateway 代理程序。

过程

1. 转至命令行，并运行 **<install_dir>/bin/sterling_file_gateway-agent.sh config instance_name** 命令。

   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是代理程序安装目录路径。
2. 系统会提示您为所有必需配置参数提供值。您可以修改配置参数的缺省值。

   有关配置参数的更多信息，请参阅下列主题：

   · 第 672 页的『B2B API 详细信息的配置参数』

   · 第 673 页的『数据库详细信息的配置参数』

   · 第 673 页的『Java API 的配置参数』
3. 要启动代理程序，请运行 **<install_dir>/bin/sterling_file_gateway-agent.sh start instance_name** 命令。

使用静默响应文件来配置 Sterling File Gateway 代理程序

您可使用静默响应文件，无需对运行配置脚本时的提示进行响应，即可配置 Sterling File Gateway 代理程序。可以在 Windows 和 Linux 系统上配置使用静默响应文件的代理程序。静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值来配置代理程序。

过程

- 要以静默方式配置 Sterling File Gateway 代理程序，请完成下列步骤：

  a) 在文本编辑器中，打开以下路径中的 `sterling_file_gatway_silent_config.txt` 文件：

  - Linux *install_dir*/samples/sterling_file_gatway_silent_config.txt

    例如，**/opt/ibm/apm/agent/samples/sterling_file_gateway_silent_config.txt**

  - Windows *install_dir*\samples\sterling_file_gateway_silent_config.txt

    例如，**C:\IBM\APM\samples\sterling_file_gateway_silent_config.txt**

  b) 在 `sterling_file_gateway_silent_config.txt` 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

  有关配置参数的更多信息，请参阅下列主题：

  - 第 672 页的『B2B API 详细信息的配置参数』
  - 第 673 页的『数据库详细信息的配置参数』
  - 第 673 页的『Java API 的配置参数』

  c) 保存并关闭 `sterling_file_gateway_silent_config.txt` 文件，然后运行下列命令：

  - Linux *install_dir*/bin/sterling_file_gateway-agent.sh config *instance_name* *install_dir*/samples/sterling_file_gateway_silent_config.txt

    例如，**/opt/ibm/apm/agent/bin/sterling_file_gateway-agent.sh config instance_name /opt/ibm/apm/agent/samples/sterling_file_gateway_silent_config.txt**

  - Windows *install_dir*/bin/sterling_file_gateway-agent.bat config *instance_name* *install_dir*/samples/sterling_file_gateway_silent_config.txt

    例如，**C:\IBM\APM\bin\sterling_file_gateway-agent.bat config instance_name C:\IBM\APM\samples\sterling_file_gateway_silent_config.txt**

    其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是代理程序的安装路径。

  要点: 确保包括静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

  d) 运行以下命令以启动代理程序：

  - Linux *install_dir*/bin/sterling_file_gateway-agent.sh start *instance_name*

    例如，**/opt/ibm/apm/agent/bin/sterling_file_gateway-agent.sh start instance_name**

– Windows *install_dir*\bin\sterling_file_gateway-agent.bat start *instance_name*

例如，**C:\IBM\APM\bin\sterling_file_gateway-agent.bat start instance_name**

在 Linux 上配置用于数据提供程序的代理程序环境变量

在 Linux 操作系统上，您可配置用于数据提供程序的 Sterling File Gateway 代理程序环境变量。

关于此任务

Sterling File Gateway 代理程序会提供环境变量，您可以为数据提供程序配置这些变量。

过程

1. 转至 <install_dir>/agent/config 目录。
2. 在编辑器中打开 .fg.environment 文件，并编辑环境变量。

   有关您可配置的代理程序环境变量的更多信息，请参阅第 671 页的『数据提供程序的环境变量』。

在 Windows 上配置用于数据提供程序的代理程序环境变量

在 Windows 操作系统上，您可以使用 **IBM Performance Management** 窗口来配置用于数据提供程序的 Sterling File Gateway 代理程序环境变量。

关于此任务

Sterling File Gateway 代理程序会提供环境变量，您可以为数据提供程序配置这些变量。

过程

1. 单击**开始** > **所有程序** > > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击代理程序实例，然后单击**高级** > **编辑 ENV 文件**，并编辑环境变量的缺省值。

   有关您可配置的代理程序环境变量的更多信息，请参阅第 671 页的『数据提供程序的环境变量』。

数据提供程序的环境变量

配置 Sterling File Gateway 代理程序之后，您可以修改一些与代理程序数据收集相关的持续时间阈值。您可以在代理程序环境文件中指定这些值。

下表包含数据提供程序环境变量的详细描述。

表 *214.* 数据提供程序环境变量的名称和描述

参数名称	描述
文件传输收集持续时间（小时）(**KFG_FILE_ARRIVED_INTERVAL**)	代理程序收集文件传输数据的持续时间（小时）。缺省值为 24 小时。
以折线图形式显示的文件传输活动的收集时间间隔（小时）(**KFG_FILE_ACTIVITY_INTERVAL**)	代理程序收集文件传输活动数据的持续时间（小时）。缺省值为 1 小时。 例如，代理程序收集过去 1 小时内发生的文件传输活动。此数据以折线图形式显示在实例页面上。缺省值为 1 小时。
不活动伙伴的时间间隔阈值（天）(**KFG_INACTIVE_PARTNERS_INTERVAL**)	伙伴处于不活动状态或者未接收或上载任何文件时的持续时间阈值。缺省值为 10 天。 例如，如果任何伙伴在过去 10 天内未接收或传输任何文件，那么它在代理程序上将会显示为“不活动”。

表 *214.* 数据提供程序环境变量的名称和描述 *(续)*	
参数名称	**描述**
数据提供程序日志文件的最大数目 (**KFG_LOG_FILE_MAX_COUNT**)	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。
每个数据提供程序日志的最大大小 (KB) (**KFG_LOG_FILE_MAX_SIZE**)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。
数据提供程序日志中的详细信息级别 (**KFG_LOG_LEVEL**)	包括在数据提供程序所创建的日志文件中的详细信息级别。缺省值是 4 (Info)。以下值有效： · 1 (Off)：不记录任何消息。 · 2 (Severe)：仅记录错误。 · 3 (Warning)：在 Severe 级别记录的所有错误和消息，以及可能会导致不良行为的潜在错误。 · 4 (Info)：在 Warning 级别记录的所有错误和消息，以及在数据提供程序处理时描述其状态的高级别参考消息。 · 5 (Fine)：在 Info 级别记录的所有错误和消息，以及在数据提供程序处理时描述其状态的低级别参考消息。 · 6 (Finer)：在 Fine 级别记录的所有错误和消息，以及详细的参考消息，例如性能概要分析信息和调试数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 7 (Finest)：在 Fine 级别记录的所有错误和消息，以及最详细的参考消息，包括低级别编程消息和数据。选择此选项可能会对监视代理程序的性能产生负面影响。此设置只应当作与 IBM 支持人员一起进行问题确定的工具来使用。 · 8 (All)：记录所有错误和消息。
访存所有文件传输的事件 (**KFG_ALL_FGEVENTS**)	用于访存所有文件传输事件的标志。有效值为 Yes 或 No。缺省值为 No。如果将值设置为 No，那么代理程序访存用户可配置持续时间内的失败文件传输的事件。如果将值设置为 Yes，那么代理程序访存用户可配置持续时间内所有文件传输的事件。

B2B API 详细信息的配置参数

在配置 Sterling File Gateway 代理程序时，您必须指定商家到商家 (B2B) API 详细信息配置参数的值。

下表包含 B2B API 详细信息配置参数的详细描述。

表 *215. B2B API* 详细信息配置参数的名称和描述	
参数名称	**描述**
实例名称 (**KFG_Instance_Name**)	实例的名称。 **限制:** “实例名称”字段显示您首次配置代理程序时指定的实例名称。重新配置代理程序时，无法更改代理程序的实例名称。
服务器名称 (**KFG_API_SERVICES_Node_ADDRESS**)	B2B API 服务的主机名或 IP 地址。
服务器端口 (**KFG_API_SERVICES_PORT**)	B2B API 的端口。

表 215. B2B API 详细信息配置参数的名称和描述 (续)	
参数名称	**描述**
用户名 (**KFG_API_SERVICES_USERNAME**)	用来连接到 B2B API 服务的用户名。
密码 (**KFG_API_SERVICES_PASSWORD**)	用来连接至 B2B API 服务的用户名的密码。

数据库详细信息的配置参数

在配置 Sterling File Gateway 代理程序时，您必须指定数据库详细信息配置参数的值。

下表包含数据库详细信息配置参数的详细描述。

表 216. 数据库详细信息配置参数的名称和描述	
参数名称	**描述**
数据库服务器名称 (**KFG_DB_Node_ADDRESS**)	Sterling File Gateway 数据库服务器的主机名或 IP 地址。
数据库用户 (**KFG_DB_USERNAME**)	数据库用户的名称。
数据库密码 (**KFG_DB_PASSWORD**)	数据库的密码。
数据库端口 (**KFG_DB_PORT**)	数据库的端口。
数据库类型 (**KFG_DB_TYPE**)	数据库的类型。

Java API 的配置参数

在配置 Sterling File Gateway 代理程序时，您必须指定 Java API 配置参数的值。

下表包含 Java API 配置参数的详细描述。

表 217. Java API 配置参数的名称和描述	
参数名称	**描述**
外部 JAR 文件的类路径 (**KFG_CLASSPATH**)	要对相应数据库指定的数据库驱动程序 JAR 路径。

配置 Sybase 服务器监视

Sybase 代理程序 提供针对分布式数据库的集中管理点。它收集了数据库管理员和系统管理员所需的信息，用于检查 Sybase 服务器系统的性能，及早检测出问题并防止问题。数据库管理员和系统管理员可以设置必需的阈值级别和标记，以在系统达到这些阈值时触发警报。
必须配置 Monitoring Agent for Sybase Server 以监视 Sybase 服务器。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Sybase 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

Sybase 代理程序 是一个多实例代理程序，您必须手动配置和启动每个代理程序实例。

过程

1. 配置监视代理程序。
   - 第 675 页的『使用命令行界面来配置代理程序』
   - 第 677 页的『使用静默响应文件来配置代理程序』
2. 使用代理程序命令 **sybase-agent** 启动和停止监视代理程序。

   有关 **sybase-agent** 的更多信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm 中的“使用代理程序命令”。
3. 使用命令 **agent2server** 将监视代理程序连接到 Performance Management 服务器。

   有关 **agent2server** 的更多信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm 中的“使用代理程序命令”。

授予许可权

必须对用于监视 Sybase 服务器的用户标识授予许可权。

开始之前

安装 Sybase 代理程序。

您必须具有数据库管理员角色以授予许可权。

关于此任务

监视代理程序使用的用户标识必须能够访问 Sybase 表和已安装的监视表。

您可以执行下列任务：

- 为监视代理程序创建用户标识。
- 对新的用户标识和已安装的监视表授予许可权。

如果未以 `root` 用户身份运行 Sybase 代理程序，那么确保用户标识属于 `Sybase` 组并且具有 Sybase 日志文件的读访问权。

过程

1. 请输入您正在使用的操作系统的命令。
   - Windows

     ```
     cd install_dir\tmaitm6\SQLLIB
     ```
   - UNIX

     ```
     cd install_dir/misc
     ```

   其中，*install_dir* 是安装 Sybase 服务器的主目录。
2. 使用 **isql** 命令以用户 `sa` 身份登录 Sybase 服务器。
3. 运行以下命令配置 Sybase 代理程序使用的标识以与 Sybase 服务器进行通信：

   ```
   1>sp_addlogin user_name, password 2>g
   ```

   其中：
   - *user_name* 为用户标识。缺省情况下，它为 `tivoli`。

     如果用户标识不是 `tivoli`，请编辑 `koygrant.sql` 文件并将 `tivoli` 更改为正确的用户标识。
   - *password* 是用户的密码。

   注：

koygrant.sql 文件的位置：

- Windows \opt\ibm\apm\agent\misc\
- UNIX /opt/ibm/apm/agent/misc/

4. 运行以下命令，对授予数据库中的表授予许可权：

```
isql -U sa -P password -S servername -i koygrant_filepathkoygrant.sql
```

其中：

- *password* 是用户 sa 的密码。
- *servername* 是数据库服务器名称。
- *koygrant_filepath* 位于以下位置：

  注：

  – Windows \opt\ibm\apm\agent\misc\

  – UNIX /opt/ibm/apm/agent/misc/

5. 运行以下命令，创建用于已安装的监视表的代理表：

```
isql -U sa -P password -S servername
     -i $SYBASE/ASE-12_5/scripts/installmontables
```

其中：

- *password* 是用户 sa 的密码。
- *servername* 是数据库服务器名称。

下一步做什么

成功授予权限后，可以配置监视代理程序。

使用命令行界面来配置代理程序

您可以使用命令行界面配置 Monitoring Agent for Sybase Server。

开始之前

Sybase 代理程序 不支持远程配置。因此，您需要确保将 Sybase 服务器与 Sybase 代理程序 安装在同一主机上。

Sybase 代理程序 仅支持 Sybase Server V15.7 和 16.0。

创建用来连接至数据库服务器的用户标识。

关于此任务

Sybase 代理程序 是一个多实例代理程序，您必须手动配置和启动每个代理程序实例。

过程

1. 运行以下命令以配置代理程序。

   - Windows

     ```
     install_dir\bin\sybase-agent.bat instance_name
     ```

   - UNIX

     ```
     install_dir/bin/sybase-agent.sh instance_name
     ```

   其中：

· *install_dir* 是代理程序安装目录。

· *instance_name* 是 Sybase 服务器实例名。

2. 当提示您提供以下参数的值时，请按 Enter 键以接受缺省值或指定一个值然后按 Enter 键。

a) 对于 Home Directory 参数，输入 Sybase 服务器主目录的路径。

· Windows

Home Directory 的示例为 \opt\sybase。

· UNIX

Home Directory 的示例为 /opt/sybase。

b) 对于 ASE Directory 参数，输入数据库服务器 ASE 的路径。

· Windows

ASE Directory 的示例为 \opt\sybase\ASE-12_5。

· UNIX

ASE Directory 的示例为 /opt/sybase/ASE-12_5。

c) 对于 Open Client Directory 参数，输入 Sybase 开放客户机安装位置。

· Windows

Open Client Directory 的示例为 \opt\sap\ocs-16_0。

· UNIX

Open Client Directory 的示例为 /opt/sap/ocs-16_0。

d) 对于 USER ID 参数，输入监视代理程序使用的用户标识，用于连接到 Sybase 服务器。

缺省 USER ID 为 tivoli。

e) 对于 PASSWORD 参数，输入监视代理程序使用的用户标识的密码，用于连接到 Sybase 服务器。

f) 对于 VERSION 参数，输入 Sybase 服务器版本。

Sybase 代理程序 仅支持 Sybase Server V15.7 和 16.0。

g) 对于 ERROR LOG FILE 参数，输入 Sybase 服务器的错误日志文件的标准文件名。

· Windows

ERROR LOG FILE 示例为 \opt\sap\ASE-16_0\install*servername*.log。

· UNIX

ERROR LOG FILE 的示例为 /opt/sap/ASE-16_0/install/*servername*.log。

其中，*servername* 是 Sybase 服务器名称。

h) 对于 EXTENDED 参数，输入支持使用的扩展参数，以排除某些光标执行。（可选）请按 Enter 键而不指定任何值来排除所有光标。

EXTENDED 参数的选项为 DBD2,DBD15,KOYSEGD。

· DBD2 将执行数据集 Sybase_Database_Detail 和 Sybase_Database_Summary 的光标执行。

· DBD15 将执行数据集 Sybase_Database_Detail 的光标执行。

· KOYSEGD 将执行数据集 Sybase_Segment_Detail 的光标执行。

下一步做什么

配置完成时，您可以启动监视代理程序并将监视代理程序连接到 Performance Management 服务器。

要启动 Sybase 代理程序，请使用代理程序命令 sybase-agent 命令。

要将 Sybase 代理程序 连接到 Performance Management 服务器，请使用 agent2server 命令。

有关 sybase-agent 和 agent2server 的更多信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm 中的“使用代理程序命令”。

使用静默响应文件来配置代理程序

您可以使用静默响应文件配置 Monitoring Agent for Sybase Server。

开始之前

Sybase 代理程序 不支持远程配置。因此，您需要确保将 Sybase 服务器与 Sybase 代理程序 安装在同一主机上。

Sybase 代理程序 仅支持 Sybase Server V15.7 和 16.0。

创建用来连接至数据库服务器的用户标识。

关于此任务

Sybase 代理程序 是一个多实例代理程序，您必须手动配置和启动每个代理程序实例。

要配置监视代理程序，必须编辑静默响应文件，然后运行代理程序命令。

过程

1. 编辑静默响应文件。
   - Windows

     静默响应文件位于：*install_dir*\samples\sybase_silent_config.txt。
   - UNIX

     静默响应文件位于：*install_dir*/samples/sybase_silent_config.txt。

   其中，*install_dir* 是代理程序安装目录。

   a) 对于 Home Directory 参数，指定 Sybase 服务器主目录的路径。
      - Windows

        Home Directory 的示例为 \opt\sybase。
      - UNIX

        Home Directory 的示例为 /opt/sybase。

   b) 对于 ASE Directory 参数，指定数据库服务器 ASE 的路径。
      - Windows

        ASE Directory 的示例为 \opt\sybase\ASE-12_5。
      - UNIX

        ASE Directory 的示例为 /opt/sybase/ASE-12_5。

   c) 对于 Open Client Directory 参数，指定 Sybase 开放客户机安装位置。
      - Windows

        Open Client Directory 的示例为 \opt\sap\ocs-16_0。
      - UNIX

        Open Client Directory 的示例为 /opt/sap/ocs-16_0。

   d) 对于 USER ID 参数，指定监视代理程序使用的用户标识，用于连接到 Sybase 服务器。

      缺省 USER ID 为 tivoli。

   e) 对于 PASSWORD 参数，指定监视代理程序使用的用户标识的密码，用于连接到 Sybase 服务器。

f) 对于 VERSION 参数，指定 Sybase 服务器版本。

Sybase 代理程序 仅支持 Sybase Server V15.7 和 16.0。

g) 对于 ERROR LOG FILE 参数，指定 Sybase 服务器的错误日志文件的标准文件名。

- Windows

  ERROR LOG FILE 示例为 \opt\sap\ASE-16_0\install*servername*.log。

- UNIX

  ERROR LOG FILE 的示例为 /opt/sap/ASE-16_0/install/*servername*.log。

其中，*servername* 是 Sybase 服务器名称。

h) 对于 EXTENDED 参数，指定支持使用的扩展参数，以排除某些光标执行。（可选）将其保留为空以执行所有光标。

EXTENDED 的选项为 DBD2,DBD15,KOYSEGD。

- DBD2 将执行数据集 Sybase_Database_Detail 和 Sybase_Database_Summary 的光标执行。
- DBD15 将执行数据集 Sybase_Database_Detail 的光标执行。
- KOYSEGD 将执行数据集 Sybase_Segment_Detail 的光标执行。

2. 保存静默响应文件。
3. 运行以下代理程序命令以配置监视代理程序。

- Windows

```
install_dir\bin\sybase-agent.bat config instance_name
 install_dir\samples\sybase_silent_config.txt
```

- UNIX

```
install_dir/bin/sybase-agent.sh config instance_name
 install_dir/samples/sybase_silent_config.txt
```

其中：

- *install_dir* 是代理程序安装目录。
- *instance_name* 是 Sybase 服务器名称。

下一步做什么

配置完成时，您可以启动监视代理程序并将监视代理程序连接到 Performance Management 服务器。

要启动 Sybase 代理程序，请使用 **sybase-agent** 命令。

要将 Sybase 代理程序 连接到 Performance Management 服务器，请使用 **agent2server** 命令。

有关 **sybase-agent** 和 **agent2server** 的更多信息，请参阅 https://www.ibm.com/support/knowledgecenter/SSHLNR_8.1.4/com.ibm.pm.doc/welcome.htm 中的“使用代理程序命令”。

为查询禁用 dirty reads

缺省情况下，Sybase 代理程序为其查询执行启用 dirty reads 以防止锁定。
变量 COLL_USE_NOLOCK 用于启用或禁用查询 dirty reads。
当启用 dirty reads 时，会以隔离级别零来执行查询，以避免锁定。
如果要为代理查询禁用 dirty reads，可将变量 COLL_USE_NOLOCK 设置为零。

开始之前

要为代理查询禁用 dirty reads，请确保已安装该代理程序。

关于此任务

缺省情况下，Sybase 代理程序启用 `dirty reads`。要为代理程序查询禁用 `dirty reads`，请完成以下步骤。

过程

1. 停止代理程序。
2. 将变量 `COLL_USE_NOLOCK` 设置为零。
   - UNIX
     a. 在 *CANDLEHOME*`/config/.oy.environment` 文件中添加 **COLL_USE_NOLOCK=0**。
     b. 保存并关闭该文件。
   - Windows
     a. 查找代理程序实例文件 *CANDLEHOME*`\TMAITM6_x64\KOYENV_`*INSTANCENAME*。
     b. 在文件中添加以下行。

        COLL_USE_NOLOCK=0

     c. 保存并关闭该文件。

   CANDLEHOME 是代理程序安装目录。
   INSTANCENAME 是代理程序实例名称。
3. 启动代理程序。

配置 Synthetic Playback 监视

您必须配置 Synthetic Playback 代理程序，以便代理程序可收集内部 Web 应用程序的可用性和性能的相关数据。此数据显示在应用程序性能仪表板中。

关于此任务

通过运行脚本并对提示做出响应来配置 Synthetic Playback 代理程序。然后，启动脚本并验证脚本是否正在运行。

要点: 只有 IBM Website Monitoring on Cloud 附加组件的现有用户才能安装、配置和运行 Synthetic Playback 代理程序。Website Monitoring 已被 IBM Cloud Availability Monitoring 替换。有关更多信息，请参阅第 887 页的『关于 Availability Monitoring』。

过程

- 要通过运行脚本并对提示做出响应来配置代理程序，请完成以下步骤:
  a) 输入 *install_dir*`/bin/synthetic_playback-agent.sh config`，其中，*install_dir* 是 Synthetic Playback 代理程序 的安装目录。
  b) 当出现 `Edit Monitoring Agent for Synthetic Playback settings` 提示时，请输入 1 以继续。
  c) 提示您输入回放存在点的数据中心名称时，请输入用于标识代理程序位置的名称。

  要点: 选择回放存在点的描述性名称。完成代理程序安装时，您可以选择按名称列示的该位置作为合成事务的回放位置并在应用程序性能仪表板中查看该位置的事务数据。
  d) 出现 `Java 参数提示时，请选择 Java 跟踪级别`。按 Enter 键以选择缺省参数，或者输入 1 到 8 之间的某个数以指定跟踪级别。
  e) 出现外部 `JAR 的类路径提示时`，按 Enter 键以保留空白，否则请指定外部 JAR 的位置。
- 要使用静默响应文件配置代理程序，请完成以下步骤:

a) 在文本编辑器中，打开 *install_dir*/samples 路径下的 synthetic_playback_silent_config.txt 文件。
例如：

Linux /opt/ibm/apm/agent/samples

b) 在 synthetic_playback_silent_config.txt 文件中，取消注释并指定以下属性的值：

- 对于 LOCATION，请使该参数与数据中心的名称或描述代理程序在何处安装的名称相同。
- 对于 JAVA_TRACE_LEVEL，请使该参数与列出的某个跟踪级别相同，例如 JAVA_TRACE_LEVEL=ERROR。

保存该文件。

c) 在命令行中，将路径更改为 *install_dir*/bin。

d) 以静默方式配置代理程序，请运行以下命令：

```
synthetic_playback-agent.sh config install_dir/samples/
synthetic_playback_silent_config.txt
```

- 要启动 Synthetic Playback 代理程序，请输入：*install_dir*/bin/synthetic_playback-agent.sh start。
- 要验证 Synthetic Playback 代理程序是否正在运行，请输入：*install_dir*/bin/synthetic_playback-agent.sh status。有关更多信息，请参阅第 153 页的表 12。

下一步做什么

要查看内部 Web 应用程序的性能，您必须在 Synthetic Script Manager 中创建合成事务。有关更多信息，请参阅第 872 页的『使用 Web 站点监视管理合成事务和事件』。

为 Synthetic Playback 代理程序 启用上游代理支持

为 Synthetic Playback 代理程序 启用上游代理支持，以监视从内部 Web 应用程序到外部 Web 应用程序的 HTTP 请求。

开始之前

请确保当前运行的是 Synthetic Playback 代理程序 V01.00.05.08 或更高版本。要检查您运行的代理程序版本，请在命令行中输入 *install_dir*/bin/cinfo -t，其中 *install_dir* 是代理程序的安装位置。如果运行的是任何其他版本的 Synthetic Playback 代理程序，那么必须从 IBM Fix Central 下载并安装 IBM Cloud Application Performance Management, Private 8.1.4.0 Synthetic Playback 代理程序 临时修订 08（在**搜索**字段中输入合成，这样会显示 Synthetic Playback 代理程序 临时修订列表）。有关安装指示信息，请参阅 8.1.4.0-IBM-IPM-SYNTHETIC-PLAYBACK-AGENT-IF0008 自述文件。

关于此任务

企业防火墙后的内部 Web 应用程序需要上游代理才能访问外部 Web 资源。为 Synthetic Playback 代理程序配置代理设置，以允许您的代理程序支持上游代理，以便您可监视从内部 Web 应用程序到外部 Web 应用程序的 HTTP 请求。

过程

- 要为代理配置并启用上游代理支持，请完成下列步骤。

a) 作为 root 用户，在命令行中运行以下命令以配置代理设置。

```
cd install_dir/agent/lx8266/sn/bin
#./set_proxy.sh
```

出现提示时，输入代理程序安装路径，缺省路径为 /opt/ibm/apm/agent。输入要为 Synthetic Playback 代理程序配置的代理类型的编号。

例如：

```
# cd /install_dir/agent/lx8266/sn/bin/
#./set_proxy.sh
please input the agent install path, default is (/opt/ibm/apm/agent)
agent install path is:/opt/ibm/apm/agent
please input the number of proxy type:
1 system proxy
2 manual proxy
3 pac proxy
4 no proxy
```

b) 输入 *install_dir*/bin/synthetic_playback-agent.sh start 以重新启动代理程序。

- Linux 要为代理程序禁用上游代理支持，请重新运行 ./set_proxy.sh 命令，然后选择 4 no proxy。随后，重新启动代理程序。

配置 Tomcat 监视

您可使用缺省设置或定制设置来配置 Monitoring Agent for Tomcat，以监视 Tomcat 应用程序服务器的资源。此代理程序可在 Windows 和 Linux 系统上配置。

开始之前

查看必备硬件和软件。有关最新系统需求信息，请参阅 Tomcat 代理程序 的软件产品兼容性报告 (SPCR)。

关于此任务

Tomcat 代理程序是一个多实例代理程序；您必须手动创建第一个实例并启动代理程序。“受管系统名称”包含指定的实例名称，例如，*instance_name*:*host_name*:*pc*，其中，*pc* 是两字符产品代码。“受管系统名称”限制为 32 个字符。指定的实例名称限制为 28 个字符减去主机名长度。例如，如果指定 TOMCAT2 作为实例名称，那么受管系统名称为 TOMCAT2:hostname:OT。如果指定长实例名称，那么会截断“受管系统名称”，并且代理程序代码无法完整显示。

为避免在配置代理程序时发生权限问题，请确保使用用于安装代理程序的相同 root 用户或非 root 用户标识。如果以一个选中的用户身份安装了代理程序，并且想要以其他用户身份配置代理程序，请参阅第 156 页的『以非 root 用户身份配置代理程序』。如果以一个选中的用户身份安装和配置了代理程序，并且想要以其他用户身份启动代理程序，请参阅第 859 页的『以非 root 用户身份启动代理程序』。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

使用缺省设置来配置 Tomcat 代理程序

您可使用 Tomcat 代理程序的缺省设置来监视 Tomcat 服务器。您只需提供新的实例名称，无需提供其他配置信息。

开始之前

在使用缺省设置来配置此代理程序之前，请确保符合下列先决条件：

- 代理程序安装在缺省目录中。
- JMX 服务 URL 使用 8686 端口。
- Tomcat 服务器配置为不进行 JMX 授权。

关于此任务

切记： 当您使用缺省设置来配置代理程序时，不启用事务跟踪和深入诊断数据的收集。

过程

1. 运行以下命令：
   Linux *install_dir*/bin/tomcat-agent.sh config *instance_name install_dir*/samples/tomcat_silent_config.txt

Windows *install_dir*/bin/tomcat-agent.bat config *instance_name install_dir*/samples/tomcat_silent_config.txt

其中

install_dir
Tomcat 代理程序 的安装目录。

instance_name
要对实例指定的名称。

2. 运行以下命令以启动代理程序:
   Linux *install_dir*/bin/tomcat-agent.sh start *instance_name*
   Windows *install_dir*/bin/tomcat-agent.bat start *instance_name*

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

在 Windows 系统上配置代理程序

可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置代理程序。

开始之前

确保满足下列先决条件:

· Java 安装在安装代理程序的 Tomcat 服务器上。

· 在代理程序安装程序的安装提示符处设置 JDK 1.6 或更高版本。

· 为 Tomcat 服务器启用 JMX 远程。有关详细信息，请参阅启用 JMX Remote。

· Tomcat 服务器已启动并且正在运行。

关于此任务

您可以从命令提示符配置代理程序。有关详细信息，请按第 685 页的『在 Linux 系统上配置 Tomcat 代理程序』主题中提供的步骤进行操作，并使用 .bat 扩展而非 .sh 扩展来运行命令。下列过程说明使用代理程序配置面板来配置代理程序。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for Tomcat**。
3. 单击**配置代理程序**。

   注意: 如果**配置代理程序**不可用，请单击**重新配置**。
4. 在“**实例名称**”窗口中，为 Tomcat 代理程序实例指定唯一名称，然后单击**确定**。

   限制: MSN 的长度不得超过 32 个字符。
5. 在**服务器名称**字段中，输入唯一名称以识别受监视的 Tomcat 服务器。
6. 在“**Java 参数设置**”窗口中，完成下列其中一个步骤:
   · 单击**下一步**，以接受缺省的 Java 安装位置。缺省安装路径为 `C:\IBM\APM\java\java80_x64\jre`。
   · 如果 IBM Java 是安装在另一路径中，请在 **Java 主目录**字段中指定该路径。
7. 在“**JSR-160 相符服务器**”窗口中，指定下列参数的详细信息:
   a) 如果在 Tomcat 中已启用 JMX 授权，请在 **JMX 用户标识**字段中，指定用来连接到 Tomcat MBean 服务器的用户标识。
   b) 如果在 Tomcat 中已启用 JMX 授权，请在 **JMX 密码**字段中，指定 JMX 用户的密码。
   c) 在 **JMX 服务 URL** 字段中，输入用于连接到 Tomcat MBean 服务器的 URL。

URL 的格式为 `service:jmx:rmi:///jndi/rmi://host_name:port_number/jmxrmi`。当服务器是在本地主机上运行，而且使用 8686 端口作为 JMX 端口时，缺省 URL 有效。您可以修改 URL 中的主机名和端口号，但需保持格式不变。

d) 如果想要启用事务跟踪和深入数据的收集，请从**数据收集器配置**列表，选择“是”。

8. 在 **Monitoring Agent for Tomcat** 窗口中，右键单击 Tomcat 代理程序实例，然后单击**启动**。
9. 启用对事务跟踪和深入数据的收集并重新启动 Tomcat 服务器。

下一步做什么

如果 Tomcat 代理程序正作为服务运行，那么在 Windows 上配置代理程序之后，请配置 Tomcat Data Collector。有关更多信息，请参阅第 684 页的『配置 Tomcat Data Collector』。

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

提供本地安全策略以便在 Windows 上以非管理员用户身份运行 Monitoring Agent for Tomcat

可以通过本地安全策略在 Windows 上以非管理员用户身份运行 Monitoring Agent for Tomcat。

关于此任务

通过结合使用以下两个本地安全策略，可以在 Windows 上以非管理员用户身份运行 Tomcat 代理程序。对于 Tomcat 代理程序启动/停止、配置和数据验证，这两个策略都有效。

1. 调试程序。
2. 作为服务登录。

请按提供的过程进行操作，将本地安全许可权用于非管理员用户。

过程

1. 转到 TEMA，将 Tomcat 代理程序启动更改为使用非管理员用户身份进行。
2. 在 Tomcat 代理程序安装文件夹下添加非管理员用户，并向其授予完全许可权。
3. 在注册表键 HKEY_LOCAL_MACHINE 下添加非管理员用户，并单击**完全许可权**。
4. 在**开始菜单**中运行 **secpol.msc** 命令，以打开“本地安全策略”。
5. 然后，要在策略中添加非管理员用户，请参阅第 683 页的『本地安全策略许可权』。
6. 重新启动 Tomcat 代理程序。
7. 检查 Tomcat 代理程序状态，并验证 APM 门户网站上的数据。

本地安全策略许可权

授予“调试程序”许可权

关于此任务

要授予“调试程序”许可权，请在 Tomcat 代理程序上完成这里描述的过程。

过程

1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开“**本地策略**”，然后单击**用户权限分配**。这样会打开用户权限列表。
3. 双击**调试程序**策略。“**调试程序属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在“输入对象名称来选择”字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

授予“作为服务登录”许可权

关于此任务
要授予“作为服务登录”许可权，请在 Tomcat 代理程序上完成这里描述的过程。

过程
1. 单击**开始 > 管理工具 > 本地安全策略**。“**本地安全设置**”窗口随即打开。
2. 展开“**本地策略**”，然后单击**用户权限分配**。这样会打开用户权限列表。
3. 双击**作为服务登录**策略。“**作为服务登录属性**”窗口随即打开。
4. 单击**添加用户或组**。“**选择用户或组**”窗口随即打开。
5. 在“输入对象名称来选择”字段中，输入要向其分配许可权的用户帐户名称，然后单击**确定**。
6. 单击**确定**。

配置 Tomcat Data Collector
如果 Tomcat 代理程序作为服务运行，那么在 Windows 上配置代理程序后，请使用此处提供的指示信息来配置 Tomcat Data Collector。

关于此任务

配置并启动 Tomcat 代理程序实例后，它会在 /CANDLEHOME/setenv_<instance_name>.bat 中生成或更新 setenv.bat 文件。此文件包含配置 Tomcat Data Collector 所需的数据收集器配置参数。

过程
1. 打开“**Apache Tomcat 属性**”窗口并单击“**Java**”
2. 从 /CANDLEHOME/setenv_<instance_name>.bat 位置打开 setenv_<instanceName>.bat
3. 从 setenv_<instance_name>.bat 复制 **JAVA_OPTS** 参数的值，如以下代码块所示：

```
agentlib:am_ibm_16=C:\IBM\APM\otdchome\7.3.0.13.0\runtime\TOMTKWIN1
-Xbootclasspath/p:C:\IBM\APM\otdchome\7.3.0.13.0\toolkit\lib\bcm-bootstrap.jar
-Djava.security.policy=C:\IBM\APM\otdchome\7.3.0.13.0\itcamdc\etc\datacollector.policy
-Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=
C:\IBM\APM\otdchome\7.3.0.13.0\runtime\TOMTKWIN1\TOM_TK_1_DCManual.txt
-Dcom.ibm.tivoli.itcam.serverHome=C:\TOMCAT_9\apache-tomcat-9.0.5\apache-tomcat-9.0.5
-Dam.home=C:\IBM\APM\otdchome\7.3.0.13.0\itcamdc
-Dcom.ibm.tivoli.itcam.toolkit.runtime.dir=C:\IBM\APM\otdchome\7.3.0.13.0\runtime
```

4. 将该值粘贴到“**Apache Tomcat 属性**”的“**Java**”选项卡中标记为“**Java 选项**”的文本框中
5. 单击**应用**
6. 转至“**控制面板**”，单击**系统 > 高级 > 环境变量**
7. 在“**系统变量**”中，编辑 *PATH* 变量，追加文件路径 <OTDC_home>\toolkit\lib\win64;<OTDC_HOME>/ toolkit\lib\win64\ttapi，然后单击**确定**

   注: 将 <OTDC_home> 替换为工具箱安装目录的真实路径。例如，C:\IBM\APM\otdchome\7.3.0.13.0\toolkit\lib\win64;C:\IBM\APM\otdchome\7.3.0.13.0\toolkit\lib\win64\ttapi
8. 单击**新建**以添加 *RUNTIME_DIR* 变量。
9. 添加**变量名称**作为 *RUNTIME_DIR*，添加**变量路径**作为 C:\IBM\APM\otdchome\7.3.0.13.0\runtime。此路径在 setenv_<instancename>.bat 中提供
10. 重新启动 Windows。确保 Tomcat 服务启动设置为“自动”

在 Linux 系统上配置 Tomcat 代理程序

在 Linux 系统上，您可运行配置脚本并对提示做出响应，以配置 Tomcat 代理程序。

开始之前

· 为 Tomcat 服务器启用 JMX 远程。有关详细信息，请参阅启用 JMX Remote。

· Tomcat 服务器已启动并且正在运行。

过程

1. 运行以下命令：
   install_dir/bin/tomcat-agent.sh config *instance_name*
   其中，*instance_name* 是要对实例指定的名称。
2. 当系统提示您为 SERVER 指定值时，请指定唯一名称以标识正被监视的 Tomcat 服务器，然后按 Enter 键。
3. 当提示您为 Java home 指定值时，请按 Enter 键以接受 Java 虚拟机的缺省安装位置。缺省位置为 /opt/ibm/apm/agent/JRE/lx8266/jre。如果代理程序未安装在缺省目录中，请指定 *install_dir*/JRE/lx8266/jre。
4. 当提示您为 JMX user ID 指定值时，请指定连接至 Tomcat MBean 服务器的用户标识。如果未启用 JMX 授权，请按 Enter 键。
5. 当提示您为 JMX password 指定值时，请指定 JMX 用户的密码并加以确认。如果未启用 JMX 权限，请按 Enter 键。
6. 当提示您为 JMX service URL 指定值时，请按 Enter 键以接受缺省 URL，或指定另一个用来连接到 Tomcat MBean 服务器的服务 URL。
   URL 的格式为 service:jmx:rmi:///jndi/rmi://*host_name*:*port_number*/jmxrmi。当服务器是在本地主机上运行，而且使用 8686 端口作为 JMX 端口时，缺省 URL 有效。您可以修改 URL 中的主机名和端口，同时保持相同格式。
7. 当系统提示您为数据收集器配置指定值时，请指定 1 并按 Enter 键以启用事务跟踪和深入数据的收集。
8. 运行以下命令以启动代理程序：
   install_dir/bin/tomcat-agent.sh start *instance_name*
9. 启用对事务跟踪和深入数据的收集，并重新启动 Tomcat 服务器。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

使用静默响应文件来配置 Tomcat 代理程序

您可使用静默响应文件来配置 Tomcat 代理程序，而无需对提示做出响应。

过程

1. 在文本编辑器中，打开位于以下路径的 tomcat_silent_config.txt 文件：
   install_dir/samples
2. 对于 **KOT_SERVER** 参数，请指定唯一名称以标识正被监视的 Tomcat 服务器。
3. 对于 **Java home** 参数，请指定 Java 虚拟机的安装路径。缺省位置为 /opt/ibm/apm/agent/JRE/lx8266/jre。如果代理程序未安装在缺省目录中，请指定 *install_dir*/JRE/lx8266/jre。
4. 对于 **JMX user ID** 参数，请指定用于连接到 Tomcat MBean 服务器的用户标识。如果已在 Tomcat 中启用 JMX 授权，那么您必须为此参数指定值。
5. 对于 **JMX password** 参数，请指定 JMX 用户的密码。如果已在 Tomcat 中启用 JMX 授权，那么您必须为此参数指定值。
6. 对于 **JMX service URL** 参数，请指定用于连接到 Tomcat MBean 服务器的服务 URL。URL 的格式为 service:jmx:rmi:///jndi/rmi://*host_name*:*port_number*/jmxrmi。当服务器是在本地主

机上运行，而且使用 8686 端口作为 JMX 端口时，缺省 URL 有效。您可以修改 URL 中的主机名和端口号，同时保持相同格式。

7. 对于 **KOT_DCCONFIGURATION** 参数，如果您想要启用事务跟踪和深入数据的收集，请指定 Yes。
8. 保存并关闭 tomcat_silent_config.txt 文件，然后运行以下命令以更新代理程序配置设置：
   Linux *install_dir*/bin/tomcat-agent.sh config *instance_name install_dir*/samples/tomcat_silent_config.txt
   Windows *install_dir*/bin/tomcat-agent.bat config *instance_name install_dir*/samples/tomcat_silent_config.txt
   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 Tomcat 代理程序的安装目录。
9. 运行以下命令以启动代理程序：
   Linux *install_dir*/bin/tomcat-agent.sh start *instance_name*
   Windows *install_dir*/bin/tomcat-agent.bat start *instance_name*
10. 如果启用对事务跟踪和深度数据的收集，请重新启动 Tomcat Server。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

启用事务跟踪和诊断数据收集

在“**代理程序配置**”页面上，您可以启用或禁用事务跟踪和诊断数据的收集。

关于此任务

启用事务跟踪数据收集后，代理程序将收集下列组件的数据：

- Servlet JSP
- EJB 应用程序
- JMS

过程

完成以下步骤为每个受管系统配置数据收集。

1. 登录到 Cloud APM 控制台。
2. 从导航栏，单击 **系统配置** > **代理程序配置**。
   “**代理程序配置**”页面随即显示。
3. 单击 **Tomcat** 选项卡。
4. 选中要对其配置数据收集的受管系统的相应复选框，然后完成**操作**列表中的下列任意一项操作。
   - 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。对于每个选中的受管系统，**事务跟踪**列中的状态将更新为“已启用”。
   - 要启用诊断数据收集，请选择**设置诊断方式** > **仅已启用诊断方式**。对于每个选中的受管系统，“诊断方式”列中的状态将更新为“已启用”。
   - 要启用诊断数据收集和方法跟踪，请选择**设置诊断方式** > **已启用诊断方式和方法跟踪**。对于每个选中的受管系统，“诊断方式”和“方法跟踪”列中的状态将更新为“已启用”。
   - 要禁用事务跟踪，请单击**设置事务跟踪** > **已禁用**。对于每个选中的受管系统，**事务跟踪**列中的状态将更新为“已禁用”。
   - 要禁用诊断数据收集，请单击**设置诊断方式** > **已禁用诊断方式和方法跟踪**。对于每个选中的受管系统，**“诊断方式”和“方法跟踪”**列中的状态将更新为“已禁用”。

下一步做什么

登录 Cloud APM 控制台，以在仪表板中查看代理程序所收集的事务跟踪和诊断数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

更新或更改 Tomcat 应用程序服务器

要在配置 Tomcat 代理程序后更新或更改 Tomcat 应用程序服务器，请遵循本主题中给定的步骤。这些步骤对于通过 Windows 和 Linux 配置的 Tomcat 均通用。

过程

1. 停止 Tomcat 代理程序 实例和 Tomcat 服务器
2. 转至 <TOMCAT_SERVER>/bin 并在编辑器中打开 setenv.sh 文件
3. 从 setenv.sh 中移除数据收集器的所有启动参数。移除该文件中的以下行

```
export LD_LIBRARY_PATH="<CANDLE_HOME>/otdchome/7.3.0.13.0/toolkit/lib/lx8266"
export RUNTIME_DIR="<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime"
export JAVA_OPTS="-agentlib:am_ibm_16=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/
<Tomcat_Application_
Server> -
Xbootclasspath/p:<CANDLE_HOME>/otdchome/7.3.0.13.0/toolkit/lib/bcm-bootstrap.jar -
Djava.security.policy=<CANDLE_HOME>/otdchome/7.3.0.13.0/itcamdc/etc/datacollector.policy -
Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/
<Tomcat_
Application_Server>/<Agent_Instance>_DCManual.txt -
Dcom.ibm.tivoli.itcam.serverHome=<TOMCAT_HOME> -
Dam.home=<CANDLE_HOME>/otdchome/7.3.0.13.0/itcamdc -
Dcom.ibm.tivoli.itcam.toolkit.runtime.dir=<CANDLE_HOME>/otdchome/7.3.0.13.0/runtime"
```

4. 保存更改并启动 Tomcat 服务器
5. 重新配置 Tomcat 代理程序以更新或更改 Tomcat 应用程序服务器
6. 仅更新或更改 Tomcat 应用程序服务器，而不更改任何其他配置设置
7. 启动 Tomcat 代理程序实例
8. 检查 setenv.sh 文件是否使用数据收集器的启动参数中的新 Tomcat 应用程序服务器进行了更新
9. 重新启动 Tomcat 服务器
10. 验证对 Tomcat 应用程序服务器进行的更改是否反映在代理机器和 IBM Cloud Application Performance Management 仪表板中
    - 验证代理机器的 <CANDLE_HOME>/otdchome/7.3.0.13.0/runtime/<Tomcat_Application_Server> 位置的 Tomcat 应用程序服务器更改
    - 验证“**聚集事务拓扑**”页面上的 Tomcat 应用程序服务器更改以及 IBM Cloud Application Performance Management 仪表板上 *KOT_Server* 属性组下的 *appserver* 属性

配置 VMware VI 监视

安装 Monitoring Agent for VMware VI 之后，您必须创建第一个实例，并手动启动此代理程序，使此代理程序可以收集所监视 VMware Virtual Infrastructure 的数据。

开始之前

- 查看必备硬件和软件。
- 在 VMware Virtual Infrastructure 中创建用户标识。代理程序将使用这个用户标识来连接至 VMware vCenter，以监视 VMware Virtual Infrastructure。请确保您对所有受监视的 vCenter 和 ESX 服务器具有“System.View”和“System.Read”特权。有关如何创建用户标识的信息，请参阅有关管理用户、组、许可权和角色的 VMware 文档。
- 确定是否对 vCenter 配置了 SSL 通信。如果已配置，那么必须将 VMware VI 代理程序配置为使用 SSL 与 vCenter 进行通信。
  - 要确定 vCenter 是否使用 SSL 进行通信，请使用 https://*vCenterIPaddress* URL 来访问 vCenter。如果可访问 vCenter，那么表明 vCenter 使用 SSL 通过网络进行通信。
  - 要将 VMware VI 代理程序配置为使用 SSL 与 vCenter 进行通信，请完成第 689 页的『启用与 VMware VI 数据源的 SSL 通信』所述的步骤。

· 确定监视 VMware Virtual Infrastructure 所需的代理程序实例数目。有关根据监视环境来调整代理程序实例数目的信息，请参阅第 688 页的『缩放和规划 VMware VI 代理程序 部署』。

关于此任务

VMware VI 代理程序是一个多实例代理程序。与只能配置为监视一个受监视应用程序并收集其数据的单实例代理程序不同，VMware VI 代理程序 可以有多个已配置的实例，它们将连接到多个 vCenter 服务器，并从远端监视 VMware Virtual Infrastructure。

配置参数可定义所监视的 VMware VI 数据源，并可定义与 VMware vCenter、vCenter Server Appliance 或个别 VMware ESX 服务器的连接。要了解这些应用程序的哪些版本受支持，请参阅 VMware VI 代理程序 的软件产品兼容性报告。

产品版本和代理程序版本通常不同。此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

您必须手动配置代理程序才能查看所有代理程序属性的数据。

· 要在 Windows 操作系统上配置此代理程序，您可以使用 **IBM Performance Management** 窗口或静默响应文件。

· 要在 Linux 操作系统上配置此代理程序，您可以运行脚本并对提示做出响应，或使用静默响应文件。

缩放和规划 VMware VI 代理程序 部署

可以在单一系统上配置的代理程序实例数目取决于该系统上资源的可用性和利用率。

下表将 VMware 环境分类为各种规模，并指出了所需的 Java 堆大小：

表 *218. VMware* 环境和 *Java* 堆大小

VMware 环境规模	**ESX 服务器数目**	**Java 堆大小**
小型环境	最多可管理 125 个 ESX(i) 服务器及 300 到 1500 个访客的 vCenter 服务器。	**-Xmx2048m** (2 GB)
中型环境	可管理 125 到 250 个 ESX(i) 服务器及 1500 到 4000 个访客的 vCenter 服务器。	**-Xmx4096m** (4 GB)
大型环境	可管理 250 到 500 个 ESX(i) 服务器及 4000 到 7500 个访客的 vCenter 服务器。	**-Xmx8192m** (8 GB)
特大型环境	可管理超过 500 个 ESX(i) 服务器及超过 7500 个访客的 vCenter 服务器。	**-Xmx16384m** (16 GB)

要增大 Java 数据提供程序的堆大小，请完成第 693 页的『增加 Java 堆大小』所述的步骤。

为了让代理程序实例顺利监视环境，安装代理程序的服务器必须要有足够的内存资源，以容纳这些代理程序实例所收集的数据。单一 VMware VI 代理程序 实例大约需要 300 到 400 MB 来监视小型环境。请参阅下列准则，了解所要配置的代理程序实例数目：

· 使用单一实例来监视单一 vCenter。请勿使用同一个实例来监视多个 vCenter。

· 在非集群环境中，使用单一实例最多可监视 8 个小型 ESX 服务器（每个 ESX 服务器包含 100 到 200 个虚拟机）。请勿在单一代理程序实例下配置多个 ESX 服务器。

· 使用 VMware VI 代理程序的多个代理程序实例可监视包含多个 vCenter 的环境。在配置多个实例之前，请确保安装此代理程序的系统上有足够的内存资源。

启用与 VMware VI 数据源的 SSL 通信

将代理程序配置为使用 SSL 与其 VMware VI 数据源进行安全通信之前，您必须向代理程序的证书信任库添加数据源 SSL 证书。

关于此任务

要点: 以下信息仅当代理程序配置为验证 SSL 证书时适用。

如果关闭 SSL 证书验证，那么即使 VMware 数据源的 SSL 证书已到期、不可信或无效，VMware VI 代理程序 会连接到这些数据源。但是，关闭 SSL 证书验证可能不安全，必须谨慎实施。

如果 VMware 数据源使用常见认证中心（例如，Verisign、Entrust 或 Thawte）签署的 SSL 证书，那么无需将证书添加到 VMware VI 代理程序 证书信任库。但是，如果数据源使用未由常见认证中心签署的证书（缺省情况下即为这种情况），那么必须将该证书添加到信任库，以允许代理程序成功连接并收集数据。

注:

1. 缺省 VMware 证书文件名为 `rui.crt`。
2. 对于 Virtual Center，缺省情况下，SSL 证书文件位于以下路径中:

   ```
   C:\Documents and Settings\All Users\Application Data\VMware\VMware
   VirtualCenter\SSL
   ```

3. 对于 ESX Server，缺省情况下 SSL 证书文件位于 `/etc/vmware/ssl` 目录中。

过程

1. 将证书文件从数据源复制到代理程序计算机。
2. 将证书文件放置在代理程序计算机上您选择的目录中。不要覆盖证书文件。添加的每个证书都使用唯一的文件名和标签。
3. 使用 *keytool* 命令将数据源证书添加到代理程序的证书信任库:

   ```
   keytool -import -noprompt -trustcacerts -alias CertificateAlias -file
   CertificateFile -keystore Truststore -storepass TruststorePassword
   ```

   其中

   CertificateAlias

   这是对添加到代理程序证书信任库的每个证书的唯一引用，例如，来自 *datasource.example.com* 的证书的相应别名为 *datasource*。

   CertificateFile

   要添加到信任库的 VMware 数据源证书的完整路径和文件名。

   Truststore

   到 VMware VI 代理程序 证书数据库的完整路径和文件名。请使用以下路径和文件名:

   - Windows （64 位）: *install_dir*\tmaitm6_x64\kvm.truststore
   - Linux （64 位）: *install_dir*/lx8266/vm/etc/kvm.truststore

   TruststorePassword

   ITMVMWAREVI 是 VMware VI 代理程序 信任库的缺省密码。要更改此密码，请参阅 Java 运行时文档，以获取有关要使用的工具的信息。

   要点: 要使用 *keytool* 命令，Java 运行时 bin 目录必须位于路径中。请使用以下命令:

   - Windows （64 位）: `set PATH=%PATH%;install_dir\java\java70_x64\jre\bin`
   - Linux （64 位）: `PATH="$PATH":/opt/ibm/apm/agent/JRE/lx8266/bin`

4. 在添加完所有数据源证书之后，启动监视代理程序。

下一步做什么
完成代理程序配置。

在 Windows 系统上配置代理程序

可以使用 **IBM Performance Management** 窗口在 Windows 操作系统上配置代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

关于此任务

VMware VI 代理程序会提供一些参数的缺省值。您可以为这些参数指定不同的值。

过程

1. 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for VMware VI**，然后单击**配置代理程序**。

   切记: 首次配置代理程序之后，**配置代理进程**选项将会禁用。要再次配置代理程序，单击**重新配置**。
3. 在 Monitoring Agent for VMware VI 窗口中，完成下列步骤：
   a) 为 VMware VI 代理程序实例输入唯一名称，然后单击**确定**。
   b) 在**数据提供程序**选项卡上，指定配置参数的值，然后单击**下一步**。
   c) 在**数据源**选项卡上，指定配置参数的值，然后单击**下一步**。

   VMware VI 代理程序 是一个多数据源代理程序。您可以从同一代理程序监视多个数据源。

   · 如果您要配置新的数据源，请单击**新建**。
   · 如果您要删除现有数据源，请单击**删除**。

   有关“Monitoring Agent for VMware VI”窗口的每个选项卡中配置参数的信息，请参阅以下主题：

   · 配置数据提供程序的参数
   · 配置数据源的参数
4. 在 **IBM Performance Management** 窗口中，右键单击您配置的实例，然后单击**启动**。

下一步做什么

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

  如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。
· 如果正在监视具有超过 500 个 ESX 主机的大型 VMware 环境，那么可能需要提高 Java 数据提供程序的堆大小。有关更多信息，请参阅第 693 页的『增加 Java 堆大小』。

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件来为配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

· 要以静默方式配置 VMware VI 代理程序，请完成下列步骤：
  a) 在文本编辑器中，打开位于以下路径的 `vmware_vi_silent_config.txt` 文件：

– Linux *install_dir*/samples/vmware_vi_silent_config.txt

示例： /opt/ibm/apm/agent/samples/vmware_vi_silent_config.txt

– Windows *install_dir*\samples\vmware_vi_silent_config.txt

示例：C:\IBM\APM\samples\vmware_vi_silent_config.txt

b) 在 vmware_vi_silent_config.txt 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

有关配置参数的信息，请参阅下列主题：

– 第 693 页的『配置数据提供程序的参数』

– 第 692 页的『配置数据源的参数』

c) 保存并关闭 vmware_vi_silent_config.txt 文件，然后运行下列命令：

– Linux *install_dir*/bin/vmware_vi-agent.sh config *instance_name* *install_dir*/samples/vmware_vi_silent_config.txt

示例：**/opt/ibm/apm/agent/bin/vmware_vi-agent.sh config instance_name /opt/ibm/apm/agent/samples/vmware_vi_silent_config.txt**

– Windows *install_dir*\bin\vmware_vi-agent.bat config *instance_name* *install_dir*\samples\vmware_vi_silent_config.txt

示例：**C:\IBM\APM\bin\ vmware_vi-agent.bat config instance_name C:\IBM\APM\samples\vmware_vi_silent_config.txt**

其中

instance_name
要对实例指定的名称。

install_dir
代理程序的安装路径。

要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

d) 运行以下命令以启动代理程序：

– Linux *install_dir*/bin/vmware_vi-agent.sh start *instance_name*

示例：**/opt/ibm/apm/agent/bin/vmware_vi-agent.sh start instance_name**

– Windows *install_dir*\bin\vmware_vi-agent.bat start *instance_name*

示例：**C:\IBM\APM\bin\vmware_vi-agent.bat start instance_name**

下一步做什么

· 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。

· 如果正在监视具有超过 500 个 ESX 主机的大型 VMware 环境，那么可能需要提高 Java™ 数据提供程序的堆大小。有关更多信息，请参阅第 693 页的『增加 Java 堆大小』。

通过对提示做出响应来配置代理程序

要在 Linux 操作系统上配置代理程序，您必须运行脚本并对提示做出响应。

过程

· 要通过运行脚本并对提示做出响应来配置代理程序，请完成以下步骤：

a) 在命令行上，运行以下命令：

install_dir/bin/vmware_vi-agent.sh config *instance_name*

示例：**/opt/ibm/apm/agent/bin/vmware_vi-agent.sh config instance_name**

其中

instance_name
要对实例指定的名称。

install_dir
代理程序的安装路径。

b) 参考下列主题以对提示做出响应：

- 第 693 页的『配置数据提供程序的参数』
- 第 692 页的『配置数据源的参数』

c) 运行以下命令以启动代理程序：

install_dir/bin/vmware_vi-agent.sh start *instance_name*

示例：**/opt/ibm/apm/agent/bin/vmware_vi-agent.sh start instance_name**

下一步做什么

- 登录 Cloud APM 控制台可在仪表板中查看代理程序所收集的数据。有关使用 Cloud APM 控制台的信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

  如需故障诊断帮助，请参阅 developerWorks 上的 IBM Cloud APM 论坛。
- 如果正在监视具有超过 500 个 ESX 主机的大型 VMware 环境，那么可能需要提高 Java™ 数据提供程序的堆大小。有关更多信息，请参阅第 693 页的『增加 Java 堆大小』。

配置数据源的参数

配置 VMware VI 代理程序 时，您可以更改数据源参数（例如，数据源的地址、用户标识和密码）的缺省值。

下表包含数据源的配置参数的详细描述。

表 *219.* 数据源的配置参数的名称和描述

参数名称	描述	必填字段
数据源标识	数据源的标识。	是
数据源地址	数据源的地址。 如果您不需要代理程序验证 SSL 证书，请将值设置为受监视的 VMware Virtual Center 或 ESX Server 的主机名或 IP 地址。 如果您希望代理程序在使用 SSL 通过网络进行通信时验证 SSL 证书，请使用证书中提供的“主体替代名称”来配置代理程序。 要查看数据中心的“主体替代名称”，请完成下列步骤： 1. 打开证书。 2. 在**证书**窗口中，单击**详细信息**选项卡。 3. 选择**主体替代名称**，并使用 DNS 名称的值。例如，如果 DNS 名称的值为“ibmesx3v3vc.ITMfVS.com”，那么将“ibmesx3v3vc.ITMfVS.com”值用于主机名。	是
使用到数据源的 SSL 连接	指示代理程序是否使用 SSL 连接来连接至 VMware Virtual Infrastructure 的数据源。 如果代理程序使用 SSL 连接来连接至数据源，请指定 Yes。否则，指定 No。缺省值为 Yes。	是

表 *219.* 数据源的配置参数的名称和描述 *(续)*		
参数名称	**描述**	**必填字段**
数据源用户标识	具有收集监视数据的足够特权并且数据源已知的用户标识。	是
数据源密码	配置为访问数据源的用户标识的密码。	是
确认数据源密码	您在**数据源密码**字段涨指定的同一密码。	

配置数据提供程序的参数

在配置 VMware VI 代理程序时，可以更改数据提供程序的缺省参数值，例如最大数据提供程序日志文件数量、最大日志文件大小以及日志文件中包含的详细信息级别。

下表包含数据提供程序的配置参数的详细描述。

表 *220.* 数据提供程序的配置参数的名称和描述		
参数名称	**描述**	**必填字段**
实例名称	实例的名称。 **限制: 实例名称**字段显示首次配置代理程序时指定的实例名称。重新配置代理程序时，无法更改代理程序的实例名称。	是
有效 SSL 证书	指示代理程序是否在使用 SSL 通过网络进行通信时验证 SSL 证书。 如果您希望代理程序在使用 SSL 通过网络进行通信时验证 SSL 证书，请将此值设置为是。将值设置为否会阻止代理程序验证 SSL 证书。缺省值为 Yes。 有关向代理程序的证书信任库添加数据源 SSL 证书的信息，请参阅第 689 页的『启用与 VMware VI 数据源的 SSL 通信』。	是
最大数据提供程序日志文件数	在覆盖先前日志文件之前，数据提供程序创建的最大日志文件数。缺省值为 10。	是
每个数据提供程序日志的最大大小 (KB)	在数据提供程序创建新的日志文件之前，数据提供程序日志文件必须达到的最大大小 (KB)。缺省值为 5190 KB。	是
数据提供程序日志中的详细信息级别	可以包含在数据提供程序创建的日志文件中的详细信息级别。缺省值为 INFO。有效值为：OFF、SEVERE、WARNING、INFO、FINE、FINER、FINEST 和 ALL。	是
KEY_STORE_PASSWORD	**KEY_STORE_PASSWORD** 允许用户使用针对代理程序 JRE 设置的新密钥库密码来配置代理程序。请注意，此 Java 密钥库与 vCenter 的密钥库无任何关系。 并不强制在每个配置上都输入密码。如果该字段留空，那么代理程序假定使用代理程序 JRE 时应使用缺省 Java 密钥库密码。	否

增加 Java 堆大小

配置 VMware VI 代理程序之后，如果要监视大型 VMware Virtual Infrastructure 环境，那么可能需要增加 Java™ 数据提供程序的堆大小。

关于此任务

Java 数据提供程序的缺省最大堆大小为 256 兆字节。您必须将最大堆大小设置为适当的值，该值取决于 VMware 环境的规模。有关各种 VMware 环境所需的堆大小的信息，请参阅第 688 页的表 218。

要点: 您在其中安装并配置 VMware VI 代理程序的系统必须有足够的内存空间来容纳需要的堆大小。

如果发生下列任何问题，可能需要增加堆大小:

- Java 数据提供程序由于 javacore 问题而停止，并在 CANDLEHOME\tmaitm6_x64 目录中创建名为 javacore.*date.time.number*.txt 的文件。
- javacore.*date.time.number*.txt 文件包含字符串 java/lang/OutOfMemoryError。

过程

- Windows
  完成以下步骤以将值 1 GB 设置为堆大小：
  1. 打开 %CANDLE_HOME%\TMAITM6_x64\kvm_data_provider.bat 文件。
  2. 在以 KVM_JVM_ARGS="%KVM_CUSTOM_JVM_ARGS... 开头的行前面添加以下行：
     ```
     SET KVM_CUSTOM_JVM_ARGS=-Xmx1024m
     ```
  3. 重新启动代理程序。
- Linux
  完成以下步骤以将值 1 GB 设置为堆大小：
  1. 打开 $CANDLEHOME/lx8266/vm/bin/kvm_data_provider.sh 文件。
  2. 在以 KVM_JVM_ARGS="$KVM_CUSTOM_JVM_ARGS... 开头的行前面添加以下行：
     ```
     KVM_CUSTOM_JVM_ARGS=-Xmx1024m
     ```
  3. 重新启动代理程序。

配置 WebLogic 监视

Monitoring Agent for WebLogic 为您提供了集中监视点，用于监视 WebLogic Server 环境的运行状况、可用性和性能。此代理程序显示一组综合指标，用于帮助您做出有关 WebLogic 资源（包括 Java 虚拟机 (JVM)、Java 消息传递服务 (JMS) 和 Java 数据库连接 (JDBC)）的明智决定。

开始之前

- 此处的方向针对的是代理程序的最新发行版，除非另有说明。
- 请确保在您的环境中符合 WebLogic 代理程序的系统需求。有关最新系统需求信息，请参阅 WebLogic 代理程序 的软件产品兼容性报告 (SPCR)。
- 在配置 WebLogic 代理程序之前，必须先通过完成以下步骤来配置 Oracle WebLogic Server：

  注: 通过使用管理控制台（通常位于 http://*weblogic-server*:7001/console），完成大部分 Oracle WebLogic Server 配置。

  1. 在“监视者”组中建立一个监视用户。
     a. 选择要监视/编辑的域。
     b. 选择**安全域**。
     c. 选择您的安全域（或者如果不存在，创建一个）。
     d. 创建将用于通过 JMX 与 WebLogic 进行通信的用户。
     e. 将此用户添加到“监视者”组。
     f. 保存对域进行的更改。
  2. 启用“侦听端口”。
     a. 选择要监视/编辑的域。
     b. 在要监视的每台服务器上，单击**环境 > 服务器 > 选择服务器**。
     c. 确保已启用**侦听端口**并记下其端口号。
     d. 如果要启用 SSL，请确保已启用 **SSL 侦听器端口**并为 SSL 设置端口。

3. 启用 JMX MBean 服务器连接。
   a. 选择要监视/编辑的域。
   b. 选择**配置 > 高级**。
   c. 单击**已启用 Platform Mbean Server**。
   d. 保存更改。
4. 启用“IIOP 协议”选项。
   a. 选择要监视/编辑的域。
   b. 在要监视的每个服务器上，单击**环境 > 服务器**，然后选择服务器。
   c. 选择**协议选项卡 > 选择 *IIOP***。
   d. 在**高级**部分下，输入缺省 IIOP 用户名和密码。
   e. 保存更改。
5. 启用 SSL。
   a. 启用 HTTP 隧道。
      1) 转至**环境 > 服务器 > 选择服务器 > 协议 > 常规**。
      2) 选中**启用 HTTP 隧道**。
   b. 启用 SSL 侦听端口。
      1) 转至**环境 > 服务器 > 选择服务器 > 配置 > 常规**。
      2) 配置端口号。

关于此任务

WebLogic 代理程序 是一个多实例代理程序，也是一个多子节点代理程序。您可以创建一个具有多个子节点（每个 WebLogic Server 对应于一个子节点）的代理程序实例，也可以为每个 WebLogic Server 创建一个代理程序实例（一个子节点对应于该服务器）。或者可以创建各种类型配置的组合。配置代理程序实例后，必须手动启动每个代理程序实例。

过程

1. 要在 Windows 系统上配置代理程序，请使用 **IBM Performance Management** 窗口或静默响应文件与代理程序配置批处理文件。
   · 第 696 页的『在 Windows 系统上配置代理程序』.
   · 第 700 页的『使用静默响应文件来配置代理程序』.
2. 要在 Linux 和 UNIX 系统上配置此代理程序，请运行代理程序配置脚本并对提示做出响应，或使用静默响应文件。
   · 第 699 页的『通过对提示做出响应来配置代理程序』.
   · 第 700 页的『使用静默响应文件来配置代理程序』.
3. 可选： 要配置事务跟踪，请配置各个代理程序实例以提供事务跟踪数据，并配置应用程序性能仪表板以显示事务跟踪数据。
   a) 遵循此过程以第 702 页的『配置 WebLogic 代理程序的事务跟踪』。
   b) 遵循此过程以第 708 页的『配置应用程序性能仪表板以显示 WebLogic 代理程序的事务跟踪数据』。

   注: 事务跟踪功能可用于 Cloud APM, Advanced 产品中的 WebLogic 代理程序。对于具备 Cloud APM, Base 产品中提供的基本资源监视功能的 WebLogic 代理程序，请跳过此步骤。

下一步做什么

在 Cloud APM 控制台中，转至应用程序性能仪表板以查看已收集的数据。有关使用 Cloud APM 控制台的更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

如果您无法在代理程序仪表板中查看数据，请先检查服务器连接日志，再检查数据提供程序日志。这些日志的缺省路径如下所示：

- Linux AIX `/opt/ibm/apm/agent/logs`
- Windows `C:\IBM\APM\TMAITM6_x64\logs`

有关故障诊断的帮助，请参阅 Cloud Application Performance Management 论坛。

在 Windows 系统上配置代理程序

在 Windows 操作系统上，您可使用 IBM Cloud Application Performance Management 窗口来配置 WebLogic 代理程序。更新配置值之后，您必须启动代理程序以保存更新后的值。

过程

1. 单击**开始** > **所有程序** > **IBM Monitoring 代理程序** > **IBM Cloud Application Performance Management**。
2. 在 **IBM Performance Management** 窗口中，右键单击 **Monitoring Agent for WebLogic** 模板，然后单击**配置代理程序**。

   切记: 首次配置代理程序实例之后，将禁用**配置代理程序**选项。要再次配置该代理程序实例，请右键单击它，然后单击**重新配置**。
3. 输入唯一的实例名称，然后单击**确定**。在实例名称中，只能使用字母、阿拉伯数字、下划线字符和减号字符。例如：`weblogic01`。

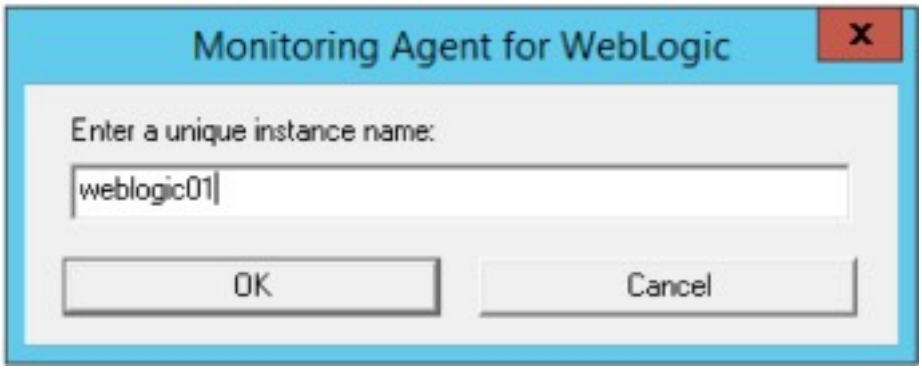


图 *27.* 用于输入唯一实例名称的窗口

4. 在**实例名称**代理程序配置面板上，单击**下一步**。

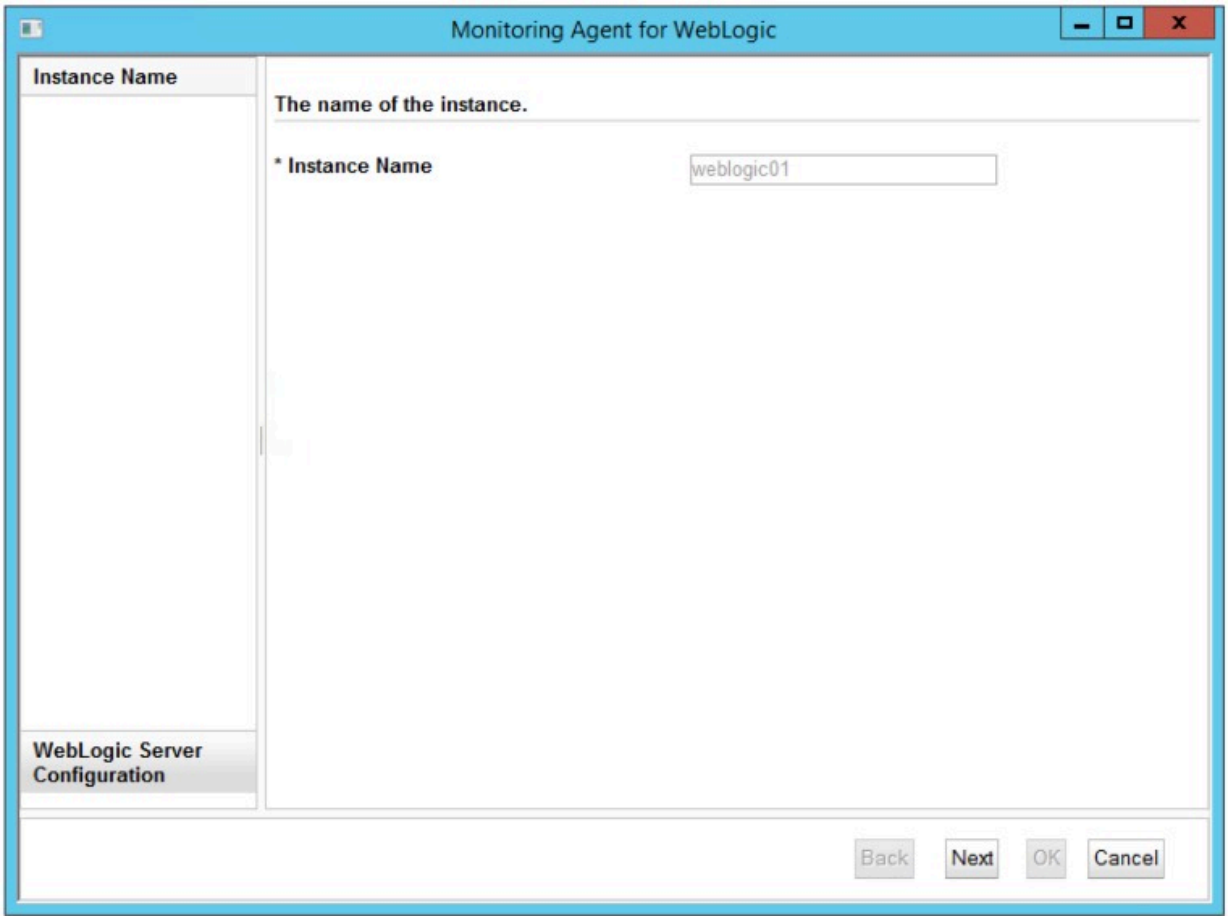


图 *28.* 显示代理程序实例名称的窗口

5. 输入 **WebLogic Server 配置**实例模板设置。

   注: 本节并非 WebLogic Server 连接实例配置。这是一个模板节，用于设置当您从步骤 6 开始添加实际的 WebLogic Server 连接实例配置时，要用作缺省值的内容。

   请参阅第 702 页的表 221，以获取每个配置参数的说明。

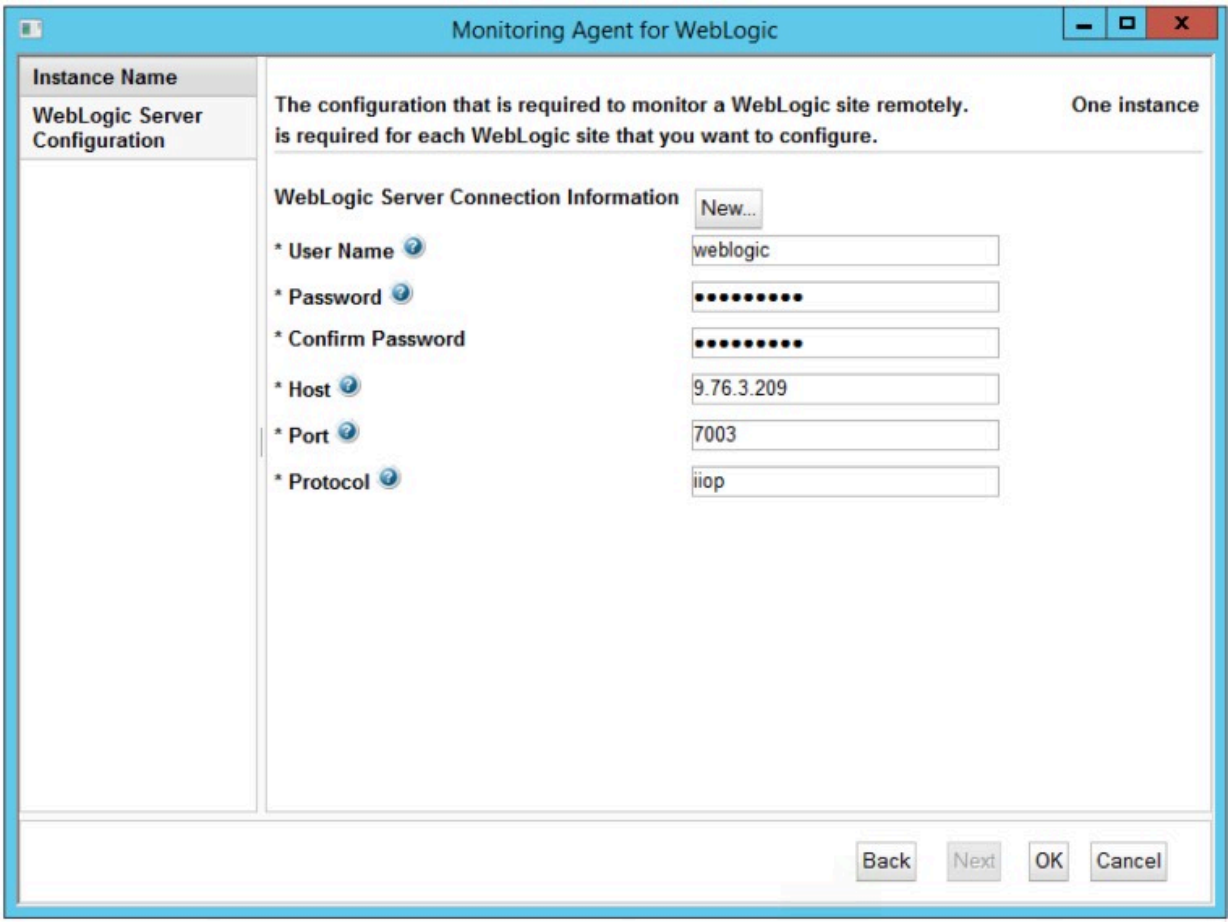


图 *29.* 用于指定 *WebLogic Server* 连接实例模板设置的窗口

6. 按**新建**并输入 WebLogic Server 连接实例设置，然后单击**下一步**。
   请参阅第 702 页的表 221，以获取每个配置参数的说明。

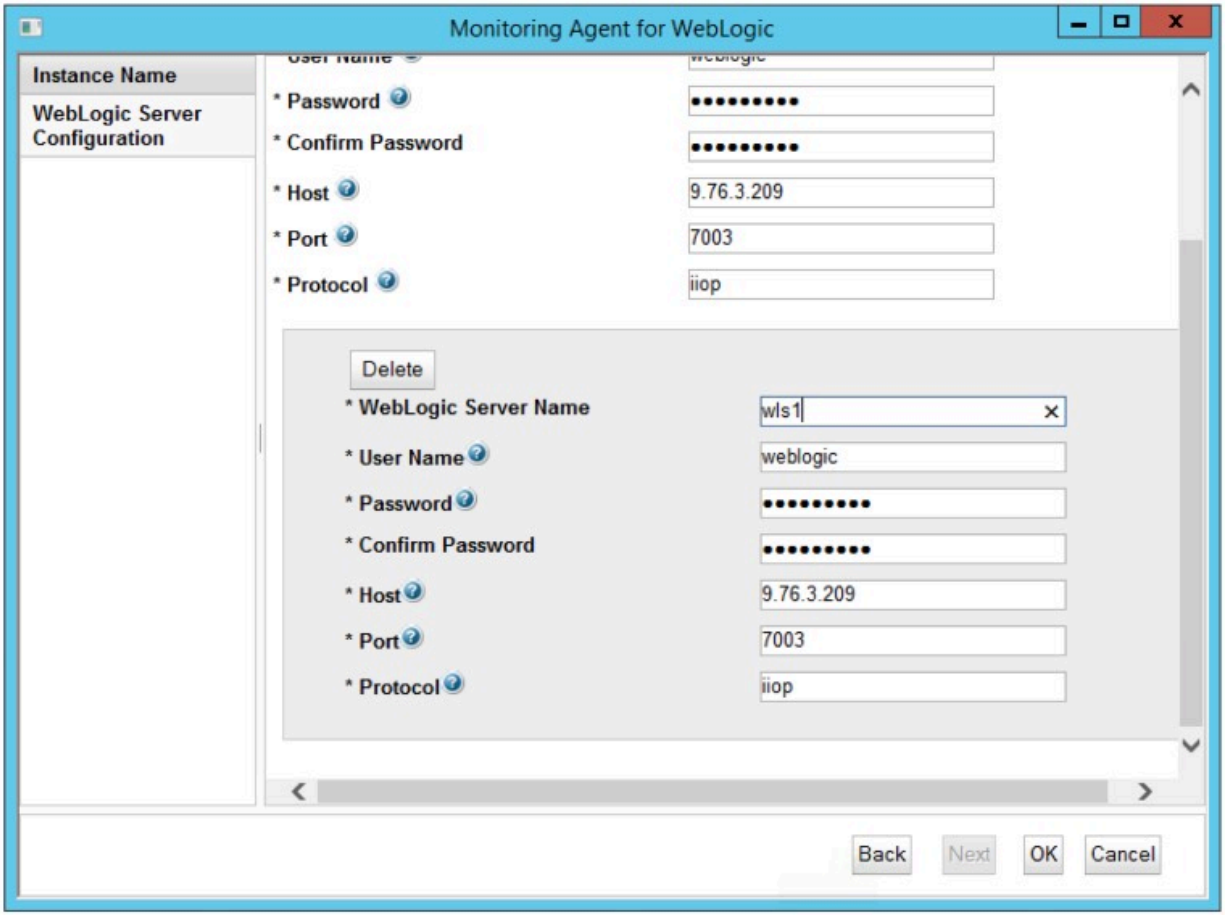


图 *30.* 用于指定 *WebLogic Server* 连接实例设置的窗口

7. 单击**确定**以完成配置。
8. 将 WebLogic 安全文件复制到 WebLogic 代理程序二进制目录。
   a. 查找 ORACLE_HOME 下面的 `wlclient.jar` 和 `wljmxclient.jar` 文件。例如，`C:\Oracle\Middleware\Oracle_Home\wlserver\server\lib`。
   b. 将文件从步骤 第 699 页的『8.a』 复制到 WebLogic 代理程序二进制目录。
      - Linux AIX *install_dir*/bin。
      - Windows *install_dir*\TMAITM6_x64

      其中，*install_dir* 是安装代理程序的路径。以下列出了缺省 *install_dir* 路径：
      - Linux AIX /opt/ibm/apm/agent
      - Windows C:\IBM\APM\TMAITM6_x64
9. 在 IBM Cloud Application Performance Management 窗口中，右键单击您所配置的实例，然后单击**启动**。

通过对提示做出响应来配置代理程序

安装 WebLogic 代理程序之后，必须先配置此代理程序，然后才能将其启动。如果 WebLogic 代理程序是安装在本地 Linux 或 UNIX 计算机上，您可按下列指示信息进行操作，通过命令行提示以交互方式进行配置。

关于此任务

切记: 如果您是对已配置的代理程序实例进行重新配置，那么对于每一项设置，将显示上次配置中设置的值。如果要清除现有的值，请在设置显示后按空格键。

过程

完成下列步骤，通过运行脚本并对提示做出响应来配置 WebLogic 代理程序。

1. 运行以下命令。

   install_dir/bin/weblogic-agent.sh config **instance_name**

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/weblogic-agent.sh config example-inst01
   ```

2. 对提示做出响应，以便为代理程序设置配置值。

   请参阅第 702 页的『WebLogic 代理程序的配置参数』，以获取每个配置参数的说明。

3. 将 WebLogic 客户机库文件复制到 WebLogic 代理程序二进制目录。

   a) 查找 ORACLE_HOME 下面的 wlclient.jar 和 wljmxclient.jar 文件。

   b) 将文件从步骤 第 700 页的『3.a』 复制到 WebLogic 代理程序二进制目录。

   install_dir/bin

   其中，*install_dir* 是安装代理程序的路径。

   示例

   ```
   /opt/ibm/apm/agent/bin
   ```

4. 运行以下命令以启动代理程序：

   install_dir/bin/weblogic-agent.sh start **instance_name**

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

   示例

   ```
   /opt/ibm/apm/agent/bin/weblogic-agent.sh start example-inst01
   ```

使用静默响应文件来配置代理程序

静默响应文件包含代理程序配置参数。您可以编辑静默响应文件以修改这些参数的值，然后运行配置脚本以创建实例并更新代理程序配置值。此配置方式称为静默方式。

关于此任务

静默响应文件包含代理程序配置参数，其中一些参数定义有缺省值。您可以编辑静默响应文件，以便对配置参数指定其他值。

在更新静默响应文件中的配置值之后，必须运行配置脚本以使用这些更新后的值配置代理程序。

过程

完成下列步骤以静默方式配置 WebLogic 代理程序。

1. 在文本编辑器中，打开位于以下路径的 weblogic_silent_config.txt 文件：

   - Linux AIX *install_dir*/samples/weblogic_silent_config.txt
   - Windows *install_dir*\samples\weblogic_silent_config.txt

   其中，*install_dir* 是安装代理程序的路径。

   示例

   - Linux AIX /opt/ibm/apm/agent/samples/weblogic_silent_config.txt
   - Windows C:\IBM\APM\samples\weblogic_silent_config.txt

2. 在 weblogic_silent_config.txt 文件中，指定所有必需参数的值。您还可以修改其他参数的缺省值。

   请参阅第 702 页的『WebLogic 代理程序的配置参数』，以获取每个配置参数的说明。
3. 保存并关闭 weblogic_silent_config.txt 文件，然后运行下列命令：

   - Linux AIX *install_dir*/bin/weblogic-agent.sh config **instance_name** *install_dir*/samples/weblogic_silent_config.txt
   - Windows *install_dir*\bin\weblogic-agent.bat config **instance_name** *install_dir*\samples\weblogic_silent_config.txt

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是想要指定给代理程序实例的名称。

   以下列出了缺省 *install_dir* 路径：

   - Linux AIX /opt/ibm/apm/agent
   - Windows C:\IBM\APM\TMAITM6_x64

   要点: 确保包含静默响应文件的绝对路径。否则，仪表板中不会显示代理程序数据。

   示例

   - Linux AIX /opt/ibm/apm/agent/bin/weblogic-agent.sh config example-inst01 /opt/ibm/apm/agent/samples/weblogic_silent_config.txt
   - Windows C:\IBM\APM\bin\weblogic-agent.bat config example-inst01 C:\IBM\APM\samples\weblogic_silent_config.txt
4. 将 WebLogic 客户机库复制到 WebLogic 代理程序二进制目录。

   a. 查找 ORACLE_HOME 下面的 wlclient.jar 和 wljmxclient.jar 文件。

   b. 将文件从步骤 第 701 页的『5.a』 复制到 WebLogic 代理程序二进制目录。

      - Linux AIX *install_dir*/bin。
      - Windows *install_dir*\TMAITM6_x64

   其中，*install_dir* 是安装代理程序的路径。以下列出了缺省 *install_dir* 路径：

   - Linux AIX /opt/ibm/apm/agent
   - Windows C:\IBM\APM\TMAITM6_x64
5. 运行以下命令以启动代理程序：

   - Linux AIX *install_dir*/bin/weblogic-agent.sh start **instance_name**
   - Windows *install_dir*\bin\weblogic-agent.bat start **instance_name**

   其中，*install_dir* 是代理程序的安装路径，*instance_name* 是代理程序实例的名称。

   以下列出了缺省 *install_dir* 路径：

   - Linux AIX /opt/ibm/apm/agent
   - Windows C:\IBM\APM\TMAITM6_x64

   示例

   - Linux AIX /opt/ibm/apm/agent/bin/weblogic-agent.sh start example-inst01
   - Windows C:\IBM\APM\bin\weblogic-agent.bat start example-inst01

WebLogic 代理程序的配置参数

下表显示 WebLogic 代理程序的配置参数。

1. WebLogic 代理程序设置 - WebLogic 代理程序环境设置。

表 *221. WebLogic* 代理程序设置

参数名称	描述	静默配置文件参数名称
WebLogic Server 名称	请提供一个名称，用于识别 WebLogic Server 代理程序实例。示例：*wls1* **注:** 此别名用来代表 WebLogic Server 代理程序实例，可以随您选择，但有下列限制。只有字母、阿拉伯数字、下划线字符和减号可用于连接名称中。连接名称的最大长度为 25 个字符。	下列每个参数都必须具有与代理程序实例的每个参数相同的实例名称后缀。新代理程序实例必须对其参数集使用唯一的实例名称。例如，一个代理程序实例可使用 *.wls1* 来替代以下参数名称中的 *.instance_name*，另一个代理程序实例可使用 *.wls2*。
用户名	这是用来向 WebLogic Server 进行认证的用户名。	**KWB_WLS_USERNAME.instance_name**
密码	这是用来向 WebLogic Server 进行认证的密码。	**KWB_WLS_PASSWORD.instance_name**
主机	这是用来向 WebLogic Server 进行认证的主机。输入 WebLogic Server 的标准主机名称或 IP 地址。	**KWB_WLS_HOST.instance_name**
端口	这是用来向 WebLogic Server 进行认证的端口。	**KWB_WLS_PORT.instance_name**
协议	这是用来向 WebLogic Server 进行认证的协议。受支持的协议为 *iiop* 和 *https*。	**KWB_WLS_PROTOCOL.instance_name**

配置 WebLogic 代理程序的事务跟踪

WebLogic 代理程序 的事务跟踪功能要求更改代理程序实例环境设置文件和 WebLogic Server 启动文件。还提供了有助于执行更改的脚本。

开始之前

Linux AIX 确保打开文件的资源限制大于 5,000，这样事务跟踪工具箱才能正常工作。

· 显示当前打开文件限制设置。**ulimit -n**

· 将打开文件限制设置为 5,056 的示例。**ulimit -n 5056**

在开始本过程之前，执行第 694 页的『配置 WebLogic 监视』 Windows 步骤 1 或 Linux AIX 步骤 2。

注: 事务跟踪功能可用于 Cloud APM, Advanced 产品中的 WebLogic 代理程序。对于具备 Cloud APM, Base 产品中提供的基本资源监视功能的 WebLogic 代理程序，请跳过此步骤。

必须将 WebLogic 代理程序 安装到使用事务跟踪功能进行监视的 WebLogic Server 本地。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权:

1. *WEBLOGIC_HOME* 目录。
2. *WEBLOGIC_HOME*/bin 目录和文件。
3. *install_dir*/config 目录。
4. *install_dir*/config/*hostname*_wb_*instance_name*.cfg 文件。

其中

WEBLOGIC_HOME
WebLogic Server 安装目录。

install_dir
代理程序的安装路径。这些日志的缺省路径如下。

- Linux AIX `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 696 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 700 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 701 页的『2』

过程

运行 **simpleConfig** 脚本。

1. 登录安装了 WebLogic 代理程序的 WebLogic Server。
2. 切换到代理程序安装目录。
   - Linux AIX *install_dir*
   - Windows *install_dir*\TMAITM6_x64

   其中，*install_dir* 是安装代理程序的路径。

   以下列出了缺省 *install_dir* 路径：
   - Linux AIX `/opt/ibm/apm/agent`
   - Windows `C:\IBM\APM\TMAITM6_x64`
3. 将目录切换到 `wbdchome/8.1.4.0.0/bin`。
4. 运行设置脚本。
   - Linux AIX `./simpleConfig.sh`
   - Windows `simpleConfig.bat`
5. 遵循提示，以便为环境输入参数：
   a) 从检测到的代理程序实例和子节点组合列表中选择要配置的 WebLogic 代理程序 *instance_name* 和子节点，其中 *instance_name* 是代理程序实例的名称。
   b) 输入 WebLogic Server 启动方法的编号。
   c) 输入 WebLogic 域根搜索路径。
   此路径用作用于搜索 WebLogic 域的基础。如果设置了 *WEBLOGIC_HOME* 环境变量，将提供该变量的值作为缺省值。
   d) 输入要配置的 WebLogic Server 的 WebLogic 域编号。
   e) 输入要配置的 WebLogic Server 名称得编号。

Linux 使用 WebLogic 启动方法 `WebLogic startup script` 的配置示例。

```
./simpleconfig.sh

The following agents and subnodes are not yet configured for transaction tracking:

1) wlinst1 Server1
2) wlinst1 Server2

Type the number that corresponds to the agent instance and subnode that you want to
```

```
configure.

Type your selection here (For example: 1): 1

The following WebLogic startup methods are supported:

1) WebLogic startup script
2) WebLogic Node Manager

Type your selection here (default is 1): 1

The path to begin looking for WebLogic domains.
WebLogic domain search root (default is: ): /home/wlsadmin

The found WebLogic domain paths are:

1) /home/wlsadmin/oracle/user_projects/domains/ttdd

Type the number that corresponds to the WebLogic domain containing the WebLogic server that
you want to configure.

Type your selection here (For examble: 1): 1

The following WebLogic servers are available for configuration:

1) AdminServer
2) Server1

Select a WebLogic server name (default is: 2): 2
INFO: [2000] Automatic configuration of agent environment file succeeded.
INFO: [3000] Automatic configuration of WebLogic start script succeeded.
INFO: [9000] Restart the WebLogic agent and WebLogic server for configuration to take effect.
```

6. 如果在步骤 第 703 页的『5.b』 中选择了 `WebLogic Node Manager` 作为 **WebLogic server startup method**，请执行以下步骤。否则，继续执行步骤 第 705 页的『7』。

   a) 打开步骤 第 703 页的『5』 输出文本的参考消息号 3011 中列出的 `weblogic_nodemanager_dc_opts` 文件。

   Windows 使用 WebLogic 启动方法 `WebLogic Node Manager` 的配置输出示例。

   ```
   INFO: [2000] Automatic configuration of agent environment file succeeded.
   INFO: [3010] Automatic configuration of WebLogic start script skipped.
   INFO: [3011] Please review C:\IBM\APM\TMAITM6_x64\wbdchome\8.1.4.0.0\runtime\ttdd_win
   \win_Server1\
   staging\weblogic_nodemanager_dc_opts.win for required WebLogic JVM start options.
   INFO: [9000] Restart the WebLogic agent and WebLogic server for configuration to take
   effect.
   ```

   b) 登录 WebLogic 控制台并选择**环境 > 服务器**。

   c) 选择要配置的服务器。

   d) 选择**配置 > 服务器启动**选项卡。

   e) 将服务器启动参数从 `weblogic_nodemanager_dc_opts` 文件复制到该服务器在 WebLogic 控制台中的**启动服务器**参数，然后保存更改。

   服务器启动参数是指 `weblogic_nodemanager_dc_opts` 文件中位于注释行 `# Add the following lines to the server start arguments` 之后的所有行。

   f) 确保在运行时，事务跟踪工具箱位于共享库路径中。

   选择方法。

   · 更新 Node Manager 启动脚本。

   注: Node Manager 启动的所有 WebLogic Server 都会将该库路径设置为包含事务跟踪工具箱对象文件库。

   1) 打开步骤 第 703 页的『5』 输出文本的参考消息号 3011 中列出的 `weblogic_nodemanager_dc_opts` 文件。

   2) 在 Node Manager 启动脚本中设置事务跟踪工具箱路径。用于设置该路径的命令是 `weblogic_nodemanager_dc_opts` 文件中位于注释行 `# Make sure that the`

executable path available to the WebLogic server includes the toolkit lib directory 后面的一行。

- Linux 将 LD_LIBRARY_PATH 行从生成的 weblogic_nodemanager_dc_opts.linux 文件复制并粘贴到 Node Manager 启动脚本中 export JAVA_OPTIONS 行的下面。例如，*WEBLOGIC_HOME*/user_projects/domains/*domain_name*/bin/startNodeManager.sh。
- Windows 将 PATH 从生成的 weblogic_nodemanager_dc_opts.win 文件复制并粘贴到 Node Manager 启动脚本中 export JAVA_OPTIONS 行的下面。例如，*WEBLOGIC_HOME*\user_projects\domains*domain_name*\bin\startNodeManager.bat。

其中，*WEBLOGIC_HOME* 是 WebLogic Server 安装目录，*domain_name* 是 WebLogic 域的名称。

· 更新启动 Node Manager 的用户帐户的环境。

注: 该用户帐户启动的所有应用程序都会将该库路径设置为包含工具箱对象文件库。

1) 编辑启动 Node Manager 的用户的环境设置。

- Linux AIX 编辑 shell 资源文件或 shell 概要文件。例如，在 bash shell 中，编辑 .bashrc 或 .bash_profile。
- Windows 编辑**控制面板 > 系统和安全 > 系统 > 高级系统设置 > 环境变量... > *user_name* 的用户变量 > 路径**，其中 *user_name* 是用于启动 WebLogic Server 的用户帐户的名称。

2) 设置用户帐户环境中的事务跟踪工具箱路径。用于设置该路径的命令是 weblogic_nodemanager_dc_opts 文件中位于注释行 # Make sure that the executable path available to the WebLogic server includes the toolkit lib directory 后面的一行。

- Linux AIX 从生成的 weblogic_nodemanager_dc_opts.linux 文件复制 export LD_LIBRARY_PATH 行。如果 export LD_LIBRARY_PATH 行不存在，请添加该行。如果存在，对其进行编辑，以使用正确的路径分隔符只将等号右侧的路径添加到现有路径。
- Windows 从生成的 weblogic_nodemanager_dc_opts.win 文件复制 set PATH 行。如果，**Path** 变量不存在于 ***user_name* 的用户变量**部分中（其中 *user_name* 是用于启动 WebLogic Server 的用户帐户的名称），输入 Path 作为变量名称，在等号右侧输入路径作为值进行添加。如果存在，对该值进行编辑，以使用正确的路径分隔符只将等号右侧的路径添加到现有路径。

3) 重新装入环境。

警告: startNodeManager 脚本由 WebLogic 配置实用程序生成。因此可能会在重新运行 WebLogic 配置时丢失更改。

7. 如果 WebLogic Server 和代理程序正在运行，请将它们重新启动。

结果

在事务跟踪配置期间更改的 WebLogic Server 文件：

· startManagedWebLogic 脚本。

- Linux AIX *WEBLOGIC_HOME*/bin/startManagedWebLogic.sh
- Windows *WEBLOGIC_HOME*\bin\startManagedWebLogic.cmd

其中，*WEBLOGIC_HOME* 是 WebLogic Server 安装目录。

事务跟踪功能的配置设置将更新到该文件中。配置标记插入到文件中以在禁用事务跟踪功能时使用。先在 *WEBLOGIC_HOME*/bin/bak/ 目录中保存备份文件，然后脚本再添加或移除事务跟踪功能更改。

在事务跟踪配置期间更改的代理程序文件：

· 代理程序实例配置文件

– Linux AIX *install_dir*/config/*hostname*_wb_*instance_name*.cfg

– Windows *install_dir*\TMAITM6_x64*hostname*_WB_*instance_name*.cfg

· 代理程序环境设置文件

– Linux AIX *install_dir*/config/wb_*instance_name*.environment

– Windows *install_dir*\TMAITM6_x64\KWBENV_*instance_name*

其中

install_dir
代理程序的安装路径。这些日志的缺省路径如下。

· Linux AIX /opt/ibm/apm/agent

· Windows C:\IBM\APM\TMAITM6_x64

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称:

· 在 Windows 系统上配置代理程序，步骤 第 696 页的『3』

· 通过对提示做出响应来配置代理程序，步骤 第 700 页的『1』

· 使用静默响应文件来配置代理程序，步骤 第 701 页的『2』

为 WebLogic 代理程序实例禁用事务跟踪

可移除 WebLogic 代理程序的事务跟踪功能。提供了一个脚本，用来移除代理程序实例的事务跟踪功能。

开始之前

确保 WebLogic Server 和 WebLogic 代理程序已关闭。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权:

1. *WEBLOGIC_HOME* 目录。
2. *WEBLOGIC_HOME*/bin 目录和文件。
3. *install_dir*/config 目录。
4. *install_dir*/config/*hostname*_wb_*instance_name*.cfg 文件。

过程

使用 **remove** 选项运行 **unconfig** 脚本。

1. 登录安装了 WebLogic 代理程序的 WebLogic Server。
2. 切换到代理程序安装目录。
   · Linux AIX *install_dir*
   · Windows *install_dir*\TMAITM6_x64
3. 将目录切换到 wbdchome/8.1.4.0.0/bin。
4. 使用 **remove** 选项运行 **unconfig**，并将代理程序实例名称命名为子节点名称。
   · 要禁用代理程序实例的一个子节点，请使用 *subnode_name* 参数。
   – Linux AIX ./unconfig.sh **remove** *instance_name subnode_name*
   – Windows unconfig.bat **remove** *instance_name subnode_name*
   · 要禁用代理程序实例的所有子节点，请省略 *subnode_name* 参数。

- Linux AIX `./unconfig.sh` **`remove`** *`instance_name`*
- Windows `unconfig.bat` **`remove`** *`instance_name`*

5. 启动 WebLogic Server 和代理程序。

其中

WEBLOGIC_HOME
WebLogic Server 安装目录。

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 696 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 700 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 701 页的『2』

subnode_name
在代理程序配置方法主题中指定给 **WebLogic Server 名称**参数的代理程序实例子节点名称：

- 在 Windows 系统上配置代理程序，步骤 第 698 页的『6』
- 通过对提示做出响应来配置代理程序，步骤 第 700 页的『2』
- 使用静默响应文件来配置代理程序，步骤 第 701 页的『2』

install_dir
代理程序的安装路径。这些日志的缺省路径如下所示。

- Linux AIX `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

卸载 WebLogic 代理程序的事务跟踪

可卸载 WebLogic 代理程序的事务跟踪功能。提供了一个脚本，用来移除所有代理程序实例中的事务跟踪功能，并移除事务跟踪工具箱。

开始之前

确保 WebLogic Server 和所有 WebLogic 代理程序实例已关闭。

运行此脚本的用户帐户必须对以下目录和文件具有写许可权：

1. *`WEBLOGIC_HOME`* 目录。
2. *`WEBLOGIC_HOME`*`/bin` 目录和文件。
3. *`install_dir`*`/config` 目录。
4. *`install_dir`*`/config/`*`hostname`*`_wb_`*`instance_name`*`.cfg` 文件。

过程

使用 **uninstall** 选项运行 **unconfig** 脚本。

1. 登录安装了 WebLogic 代理程序的 WebLogic Server。
2. 切换到代理程序安装目录。
   - Linux AIX *`install_dir`*
   - Windows *`install_dir`*`\TMAITM6_x64`
3. 将目录切换到 `wbdchome/8.1.4.0.0/bin`。
4. 使用 **uninstall** 选项运行 **unconfig**。

- Linux AIX `./unconfig.sh uninstall`
- Windows `unconfig.bat uninstall`

5. 启动 WebLogic Server 和所有代理程序实例。

其中

WEBLOGIC_HOME
WebLogic Server 安装目录。

hostname
安装代理程序的主计算机的名称。

instance_name
在代理程序配置方法主题中指定的代理程序实例的名称：

- 在 Windows 系统上配置代理程序，步骤 第 696 页的『3』
- 通过对提示做出响应来配置代理程序，步骤 第 700 页的『1』
- 使用静默响应文件来配置代理程序，步骤 第 701 页的『2』

install_dir
代理程序的安装路径。这些日志的缺省路径如下所示。

- Linux AIX `/opt/ibm/apm/agent`
- Windows `C:\IBM\APM\TMAITM6_x64`

配置应用程序性能仪表板以显示 WebLogic 代理程序的事务跟踪数据

查看由 WebLogic 代理程序的事务跟踪功能所收集的数据需要对应用程序性能仪表板进行配置更改。

开始之前

在遵循此过程前，执行第 702 页的『配置 WebLogic 代理程序的事务跟踪』。

过程

1. 如果您使用的是具备事务跟踪功能的 WebLogic 代理程序（该功能在 Cloud APM, Advanced 产品中提供），并且要启用事务跟踪功能，请在“应用程序性能仪表板”中启用事务跟踪数据。
   a) 从导航栏，单击 **系统配置** > **代理程序配置**。
   “**代理程序配置**”页面随即显示。
   b) 选择 **WebLogic** 选项卡。
   c) 选中要监视的 WebLogic Server 代理程序实例的相应复选框，然后从**操作**列表执行下列其中一项操作：
   - 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。**事务跟踪**列中的状态将更新为已启用。
   - 要禁用事务跟踪，请单击**设置事务跟踪** > **已禁用**。**事务跟踪**列中的状态将更新为已禁用。
2. 要查看 WebLogic 代理程序事务跟踪数据仪表板，请将 WebLogic 代理程序实例添加到“应用程序性能仪表板”中的应用程序。
   有关“应用程序”编辑器的更多信息，请参阅管理应用程序。
3. 确保将用户帐户分配给具有“诊断仪表板”许可权的角色，以访问下列 WebLogic 代理程序事务跟踪“应用程序仪表板”按钮。
   否则，在“应用程序仪表板”中，对于该用户，禁用这些按钮。
   a. **最慢的 5 个响应时间**窗口小部件上的**诊断**向下钻取按钮。
   b. **应用程序**窗口小部件上的**正在进行的请求**按钮。

配置 WebSphere Applications 监视

配置 WebSphere Applications 监视包括为您的应用程序服务器配置数据收集器。数据收集器可以是独立，也可以在其中嵌入 WebSphere Applications 代理程序。

嵌入式数据收集器

大多数 WebSphere Application Server 可以被嵌入式数据收集器监视，IBM Cloud 上的 Liberty 概要文件除外。嵌入式数据收集器可以提供所有可用监视功能。

要配置嵌入式数据收集器，您必须在运行该应用程序服务器的系统上先安装 WebSphere Applications 代理程序。然后，使用所提供的配置实用程序以交互方式或静默方式配置数据收集器。

独立的数据收集器

独立的数据收集器只适用于 Linux for System x 上的 WebSphere Application Server Liberty 以及 IBM Cloud 上的 WebSphere Liberty 概要文件。

如果选择配置独立的数据收集器，您可以跳过代理程序安装过程并在 Liberty 中直接配置数据收集器。

但是，独立的数据收集器将不收集一些按需诊断数据，例如，当前时间的堆转储或未完成的请求信息。这意味着您只能启用数据收集器，按指定的时间间隔自动收集堆转储信息，但无法随时使用 Cloud APM 控制台中的**获取快照**按钮来获取堆快照。所有未完成请求的相关仪表板（由嵌入式数据收集器提供）都不可供独立数据收集器使用。

使用第 709 页的表 222 来确定您的应用程序服务器的相应数据收集器。

表 *222. WebSphere* 应用程序和适用的数据收集器

要监视的应用程序	适用的数据收集器	文档
WebSphere Application Server（传统）	嵌入式数据收集器	第 709 页的『为 WebSphere Applications 代理程序 配置数据收集器』
WebSphere Application Server Liberty（本地）	· 嵌入式数据收集器 · 独立的数据收集器（仅限 Linuxfor System x）	· 第 709 页的『为 WebSphere Applications 代理程序 配置数据收集器』 · 第 749 页的『为本地应用程序配置 Liberty 数据收集器』
IBM Cloud 上的 WebSphere Liberty 概要文件	独立的数据收集器	第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』
Docker 容器中的 WebSphere Liberty 概要文件	嵌入式数据收集器	第 737 页的『监视 Docker 容器内部的 WebSphere Application Server Liberty』

为 WebSphere Applications 代理程序 配置数据收集器

除非您要更改缺省端口，否则 WebSphere Applications 代理程序在安装后无需任何配置。但是，您必须配置数据收集器（这是代理程序的一个组件），才能设置 WebSphere 环境监视。

关于此任务

此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增功能的详细信息，请参阅 第 44 页的『变更历史记录』。

过程

- （快速通道）如果您仅使用 WebSphere Applications 代理程序，而环境中没有旧式产品 ITCAM Agent for WebSphere Applications，要快速设置环境以进行监视，请参阅第 710 页的『快速通道：配置 WebSphere Applications 代理程序的数据收集器』以了解简化配置流。
- （简单配置）要了解纯 IBM Cloud Application Performance Management 环境的完整配置流，请参阅第 712 页的『使用简单配置实用程序来配置数据收集器』。

- （完全配置）要使用更多定制选项来配置数据收集器，请使用完全配置实用程序。有关指示信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。
- （静默配置）要为多个应用程序服务器实例部署相同的监视，请以静默方式配置数据收集器。有关指示信息，请参阅第 721 页的『以静默方式配置数据收集器』。
- (WebSphere Portal Server) 要监视 WebSphere Portal Server 实例，请使用高级配置过程。有关指示信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。
- （手动配置）如果您无法使用所提供的配置实用程序来为 WebSphere Applications 代理程序配置数据收集器，请在 WebSphere 管理控制台中手动配置数据收集器。有关指示信息，请参阅第 727 页的『在配置实用程序失败时手动配置数据收集器』。
- （代理程序共存）如果要配置数据收集器，让其在同时安装有 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications 的代理程序共存环境中工作，请参阅第 730 页的『（代理程序共存）配置 WebSphere Applications 代理程序和数据收集器』。
- （Docker 监视）要监视在 Docker 容器内运行的 WebSphere Application Server Liberty，请参阅第 737 页的『监视 Docker 容器内部的 WebSphere Application Server Liberty』。

快速通道：配置 WebSphere Applications 代理程序的数据收集器

WebSphere Applications 代理程序在代理程序安装后无需任何配置。但是，您必须配置数据收集器（这是代理程序的一个组件），才能设置 WebSphere 环境监视。

开始之前

1. 在安装并运行所要监视的应用程序服务器的系统上，安装 WebSphere Applications 代理程序。
2. 检查用户访问权需求。
   - Windows 使用用来安装应用程序服务器的管理员标识来配置数据收集器。请确保此用户标识对数据收集器主目录 (*install_dir*\dchome\7.3.0.14.08) 具有完全的写许可权。
   - Linux AIX 使用用来安装应用程序服务器的用户标识来配置数据收集器。请确保此用户标识对 *install_dir*/yndchome/7.3.0.14.08 中的下列子目录具有读写许可权：
     - bin
     - data
     - runtime

关于此任务

在此过程中，使用简式配置实用程序 simpleconfig 来提供数据收集器的基本配置。

simpleconfig 实用程序使用缺省设置来配置数据收集器。要使用更多定制选项来配置数据收集器，请使用同一目录中的完全配置实用程序 config。有关指示信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。

在大多数情况下，simpleconfig 实用程序已足够。在较复杂的环境中，您可使用 config 配置实用程序来配置数据收集器。如果 simpleconfig 实用程序失败，请改为使用 config 实用程序。

过程

1. 以用来安装应用程序服务器的用户标识身份登录系统。
2. 切换到数据收集器主目录中的 bin 目录。
   - Windows *install_dir*\dchome\7.3.0.14.08\bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
3. 运行以下简式配置实用程序：
   - Windows **simpleconfig.bat**
   - Linux AIX **./simpleconfig.sh**

4. 遵循提示，以继续执行数据收集器配置。

   根据应用程序服务器设置，您需要执行下列部分或全部操作：

   - 对于传统 WebSphere Application Server：
     - 选择自动发现的 WebSphere 安装目录，或手动指定安装目录。
     - 选择所要监视的 WebSphere Application Server 概要文件。
     - 选择要使用的安全属性概要文件，或提供 WebSphere 管理控制台的用户名和密码（如果已对应用程序服务器启用安全性）。
   - 对于 WebSphere Application Server Liberty：
     - 指定 `bin` 和 `servers` 目录所在 Liberty 主目录的完整路径。例如，`/opt/ibm/wlp`。
     - 指定 Liberty 所使用的 JRE 的主目录。
5. 数据收集器配置完成之后，重新启动应用程序服务器。
   a) 切换到应用程序服务器概要文件主目录下的 `bin` 目录。例如，`opt/IBM/WebSphere/AppServer/profiles/`*`profile_name`*`/bin`。
   b) 在命令控制台中输入 **stopServer** 命令，以停止应用程序服务器。
      - Linux AIX `./stopServer.sh` *`server_name`*
      - Windows `stopServer.bat` *`server_name`*
   c) 出现提示时，请输入 WebSphere 管理控制台管理员的用户标识及密码。
   d) 在命令控制台中输入 **startServer** 命令，以再次启动应用程序服务器。
      - Linux AIX `./startServer.sh` *`server_name`*
      - Windows `startServer.bat` *`server_name`*
6. 登录 Cloud APM 控制台，以在仪表板中查看数据。
   a) 使用提醒您服务已就绪的电子邮件中提供的链接来访问控制台。另外，也可以从 IBM Marketplace Web 站点访问控制台。有关详细的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
   b) 使用“应用程序”编辑器，将受监视应用程序服务器添加到 应用程序性能仪表板。您可将其作为新组件添加到现有应用程序，或创建应用程序以包含此组件。

      有关“应用程序”编辑器的更多信息，请参阅第 930 页的『管理应用程序』。

结果

数据收集器配置为监视应用程序服务器实例。请注意，数据收集会增加应用程序服务器上的开销。您可使用更高级的配置选项进行调整，以控制数据收集。

检查用户访问需求

WebSphere Applications 代理程序对用于配置数据收集器的用户标识有一些用户访问权需求。

关于此任务

将适当的许可权授予应用程序服务器安装标识之后，请使用用来安装应用程序服务器的标识来配置数据收集器。

过程

- Windows 使用用来安装应用程序服务器的管理员标识来配置数据收集器。请确保此用户标识对数据收集器主目录 (*`install_dir`*`\dchome\7.3.0.14.08`) 具有完全的写许可权。
- Linux AIX 使用用来安装应用程序服务器的用户标识来配置数据收集器。请确保此用户标识对 *`install_dir`*`/yndchome/7.3.0.14.08` 中的下列子目录具有读写许可权：
  - `bin`
  - `data`

- logs
- runtime

切记: 如果使用不同的用户标识来安装应用程序服务器，那么可能需要使用不同的用户标识来配置数据收集器。第一次配置数据收集器之后，您每一次使用不同的用户标识来配置数据收集器时，请授予下列文件的写许可权，其中 *profile_name* 是应用程序服务器概要文件名:

- *install_dir*/yndchome/7.3.0.14.08/data/findservers.inputlist
- *install_dir*/yndchome/7.3.0.14.08/data/*profile_name*.findservers.progress
- *install_dir*/yndchome/7.3.0.14.08/data/config_inputlist
- *install_dir*/yndchome/7.3.0.14.08/runtime/custom/connections.properties

处理应用程序服务器中现有的其他数据收集器
如果应用程序服务器中已存在数据收集器，您必须决定如何处理现有的数据收集器，使其不会与 WebSphere Applications 代理程序数据收集器发生冲突。

关于此任务

在要监视的应用程序服务器中，可能已存在下列类型的数据收集器:

· 安装在先前版本 IBM Cloud Application Performance Management 中的 WebSphere Applications 代理程序数据收集器
· 安装在旧 IBM® Tivoli® Monitoring 基础结构中的 ITCAM Agent for WebSphere Applications 数据收集器
· 任何其他并非由 IBM 提供的数据收集器

过程

采取适当的措施，以避免数据收集器冲突。

· 对于安装在先前版本 IBM Cloud Application Performance Management 中的先前版本 WebSphere Applications 代理程序数据收集器，选项如下:
  - 使用最新数据收集器主目录中的迁移实用程序来迁移该数据收集器。有关指示信息，请参阅第 968 页的『WebSphere Applications 代理程序：迁移数据收集器』。
  - 取消配置先前版本的数据收集器，然后使用最新数据收集器主目录中的配置实用程序重新配置该数据收集器。有关取消配置数据收集器的信息，请参阅第 126 页的『WebSphere Applications 代理程序：取消配置数据收集器』。
· 对于 ITCAM Agent for WebSphere Applications 的数据收集器，如果要在代理程序共存环境中部署监视，请完成下列步骤:
  a) 卸载 ITCAM Agent for WebSphere Applications 的数据收集器。
  b) 仅配置一个数据收集器，将数据同时发送到 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications。有关指示信息，请参阅第 730 页的『（代理程序共存）配置 WebSphere Applications 代理程序和数据收集器』。
· 对于并非由 IBM 提供的其他数据收集器，请评估是否有必要移除这些数据收集器。WebSphere Applications 代理程序数据收集器使用 Java 字节代码处理来收集数据。使用同一方式来收集数据的其他数据收集器可能会与 WebSphere Applications 代理程序数据收集器发生冲突。

使用简单配置实用程序来配置数据收集器
WebSphere Applications 代理程序在安装后自动启动，但是您必须手动配置数据收集器（代理程序的一个组件）才能监视应用程序服务器实例。

开始之前

· 请确保在您的环境中符合用户访问权需求。有关指示信息，请参阅第 711 页的『检查用户访问需求』。
· 如果所要监视的应用程序服务器中存在其他数据收集器，请采取相应的措施，以避免数据收集器冲突。有关指示信息，请参阅第 712 页的『处理应用程序服务器中现有的其他数据收集器』。

关于此任务

要点:

- 如果要配置数据收集器仅用于资源监视，或要设置额外选项，请使用完全配置过程。有关指示信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。
- 如果要更改监视用户界面中的服务器名称，请重新配置数据收集器，并指定服务器别名。有关指示信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。

对于 WebSphere Applications 代理程序，*dc_home* 变量指的是数据收集器的主目录。*dc_home* 变量在每个操作系统上的位置如下所示:

- Windows *install_dir*\dchome\7.3.0.14.08
- Linux AIX *install_dir*/yndchome/7.3.0.14.08

过程

1. 以用来安装应用程序服务器的用户标识身份登录系统。
2. 切换到数据收集器主目录中的 bin 目录。
   - Windows *install_dir*\dchome\7.3.0.14.08\bin
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin
3. 运行以下简式配置实用程序:
   - Windows simpleconfig.bat
   - Linux AIX ./simpleconfig.sh

   simpleconfig 实用程序会自动发现应用程序服务器的主目录。
4. 遵循提示，以继续执行数据收集器配置。

   根据应用程序服务器设置，您需要执行下列操作:

   - 对于传统 WebSphere Application Server:
     - 选择自动发现的 WebSphere 安装目录，或手动指定安装目录。
     - 选择所要监视的 WebSphere Application Server 概要文件。
     - 选择要使用的安全属性概要文件，或提供 WebSphere 管理控制台的用户名和密码（如果已对应用程序服务器启用安全性）。
   - 对于 WebSphere Application Server Liberty:
     - 指定 bin 和 servers 目录所在 Liberty 主目录的完整路径（例如，/opt/ibm/wlp）。
     - 指定 Liberty 所使用的 JRE 的主目录。
5. 如有可能，在数据收集器配置完成之后，请重新启动应用程序服务器实例。

   a) 切换到应用程序服务器概要文件主目录下的 bin 目录。例如，opt/IBM/WebSphere/AppServer/profiles/*profile_name*/bin。

   b) 在命令控制台中输入 **stopServer** 命令，以停止应用程序服务器。
   - Linux AIX ./stopServer.sh *server_name*
   - Windows stopServer.bat *server_name*

   c) 出现提示时，请输入 WebSphere 管理控制台管理员的用户标识及密码。

   d) 在命令控制台中输入 **startServer** 命令，以再次启动应用程序服务器。
   - Linux AIX ./startServer.sh *server_name*
   - Windows startServer.bat *server_name*

结果

- 数据收集器配置为监视概要文件中的所有实例，或者，对于 WebSphere Application Server Liberty，监视同一个目录中的一个或多个实例。要监视更多概要文件或实例，请重复此配置。
- 将在服务器实例中对数据收集器进行配置，从而提供最大的监视力度。
- 对于 Cloud APM, Base，启用资源监视。
- 对于 Cloud APM, Advanced，启用资源监视、事务跟踪和诊断数据。

已知限制: 在监视 WebSphere Application Server Liberty 时，数据收集器无法生成 Java 命名和目录接口 (JNDI) 事件。

下一步做什么

- 登录 Cloud APM 控制台，并使用“应用程序”编辑器，将受监视的应用程序服务器添加到“应用程序性能仪表板”。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关使用“应用程序”编辑器的信息，请参阅第 930 页的『管理应用程序』。

  切记: 如果 WebSphere Applications 代理程序配置为监视 WebSphere Portal Server，那么代理程序与 应用程序性能仪表板 上的 WebSphere Portal Server 应用程序组件相关，而不是与 WebSphere Application Server 相关。
- 如果 应用程序性能仪表板上的监视用户界面不显示应用程序服务器实例的任何信息，请通过完成下列步骤来重新启动 WebSphere Applications 代理程序的监视代理程序组件：
  - Windows

    ```
    cd install_dir\bin
    was-agent.bat stop
    was-agent.bat start
    ```
  - Linux AIX

    ```
    cd install_dir/bin
    ./was-agent.sh
    stop
    ./was-agent.sh start
    ```

使用完全配置实用程序来配置或重新配置数据收集器

要设置其他配置选项，可以使用完全配置实用程序（交互或静默）代替简单配置实用程序来配置数据收集器。您还可使用完全配置实用程序来重新配置已配置的数据收集器。此外，还需要使用完全配置实用程序来配置针对 WebSphere Portal Server 实例的监视。

开始之前

- 请确保在您的环境中符合用户访问权需求。有关指示信息，请参阅第 711 页的『检查用户访问需求』。
- 如果所要监视的应用程序服务器中存在其他数据收集器，请采取相应的措施，以避免数据收集器冲突。有关指示信息，请参阅第 712 页的『处理应用程序服务器中现有的其他数据收集器』。

关于此任务

配置和重新配置实用程序位于下列目录：

- Windows *install_dir*\dchome\7.3.0.14.08\bin
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin

过程

- 配置实用程序名为 **config**。在下列情况下，您可能需要使用完全配置实用程序来配置数据收集器：
  - **simpleconfig** 配置实用程序失败。
  - 您想配置针对 WebSphere Portal Server 实例的监视。

- 您想指定数据收集器配置期间要在监视用户界面中显示的服务器别名。
- 您想要进一步控制所要收集的数据。例如，您只想使用资源监视功能，而禁用诊断数据和事务跟踪。
- 您不想在同一个概要文件中同时配置所有应用程序服务器。
- 数据收集器并非配置在应用程序服务器中，您想对其进行重新配置。

有关交互式完全配置实用程序的信息，请参阅第 715 页的『以交互方式配置数据收集器』。

- 重新配置实用程序名为 **reconfig**。在下列情况下，可能需要重新配置数据收集器：
  - 在以交互方式或静默方式配置数据收集器之后，您想对其进行重新配置。

  有关交互式重新配置实用程序的信息，请参阅第 718 页的『以交互方式重新配置数据收集器』。
- 有关静默配置，请参阅第 721 页的『以静默方式配置数据收集器』。

以交互方式配置数据收集器

当 simpleconfig 实用程序失败时，您可使用交互式配置实用程序（config.sh 或 config.bat）来配置数据收集器。可以使用 config.sh 或 config.bat 实用程序为要监视的每个应用程序服务器实例配置数据收集器。

开始之前

如果您要配置数据收集器以监视 WebSphere Application Server Liberty，请将 **JAVA_HOME** 系统环境变量设置为用于应用程序服务器的 JVM。例如，在 Windows 系统上，将 **JAVA_HOME** 值设置为 C:\Program Files\IBM\java。或者，在 Linux 系统上，运行 export JAVA_HOME=/opt/IBM/java。

关于此任务

使用下列完全配置实用程序来配置数据收集器：

- Windows *install_dir*\dchome\7.3.0.14.08\bin\config.bat
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin/config.sh

过程

要通过对提示做出响应来配置数据收集器，请完成下列步骤：

1. 以用来安装应用程序服务器的用户标识身份登录系统。
2. 转至 *dc_home* 数据收集器主目录中的 bin 目录。
3. 通过发出以下命令启动配置实用程序：
   - Windows config.bat
   - Linux AIX ./config.sh

   配置实用程序将显示在本地计算机系统上找到的所有网卡的 IP 地址和主机名。
4. 输入对应于 IP 地址和主机名的编号。如果您要使用的 IP 地址和主机名不在列表中，请输入该 IP 地址或主机名。
5. 指定所要监视的应用程序服务器的主目录。
   - 对于传统 WebSphere Application Server，请输入对应于自动发现的应用程序服务器主目录的编号，或指定应用程序服务器主目录的完整路径。
   - 对于 WebSphere Application Server Liberty，请输入包含 bin 和 servers 目录的 WebSphere Application Server Liberty 主目录的完整路径，例如 /opt/ibm/wlp。
6. 如果您是在为 WebSphere Application Server Liberty 配置数据收集器，那么系统将提示输入 Java 主目录。请指定用于应用程序服务器的 Java 主目录。例如，/opt/IBM/java。
7. 在配置实用程序列出所指定应用程序服务器主目录下的所有概要文件之后，请输入对应于所要配置的应用程序服务器概要文件的编号。

- 对于传统 WebSphere Application Server，配置实用程序接着将指出是否已对指定的 WebSphere Application Server 概要文件启用 WebSphere 全局安全性。如果未启用全局安全性，请前进到步骤 第 716 页的『9』。
- 对于 WebSphere Application Server Liberty，请前进到步骤 第 716 页的『10』。

8. 如果已对 WebSphere Application Server 概要文件启用全局安全性，请指定是否从客户机属性文件检索安全设置。输入 1 以允许配置实用程序从相应的客户机属性文件中检索用户名和密码。否则，输入 2 以输入用户名和密码。

   数据收集器使用远程方法调用 (RMI) 或 SOAP 协议与 WebSphere Administrative Services 进行通信。如果为概要文件启用了全局安全性，那么您必须指定有权登录应用程序服务器概要文件的 WebSphere Application Server 管理控制台的用户的用户标识和密码。或者，您可以在配置数据收集器之前对用户名和密码进行加密，并将其存储在客户机属性文件中。必须将 `sas.client.props` 文件用于 RMI 连接，或将 `soap.client.props` 文件用于 SOAP 连接。

9. 提示输入 WebSphere 管理控制台的主机名时，请按 Enter 键以接受缺省值，或指定 WebSphere 管理控制台的主机名或 IP 地址。缺省值为 `localhost`。

   切记: 对于 Network Deployment 环境，输入 Deployment Manager 的主机名或 IP 地址。

10. 当配置实用程序列出所有尚未配置数据收集的服务器实例时，请从列表中选择一个或多个应用程序服务器实例。请输入要配置数据收集的应用程序服务器实例的相应编号，或者输入星号 (*) 以便针对所有应用程序服务器实例配置数据收集。要指定服务器的子集，请输入代表这些服务器的编号，以逗号隔开。例如，`1,2,3`。

    切记:

    - 对于单机环境，必须在配置期间运行应用程序服务器实例。（不需要运行 WebSphere Application Server Liberty 实例）。
    - 对于 Network Deployment 环境，Deployment Manager 必须正在运行。
    - 请确保您选择的应用程序服务器实例是托管了您想要监视的应用程序或服务的实际服务器。

11. 在 **Integration with Agent for WebSphere Applications** 部分中，指定您希望将数据收集器与 WebSphere Applications 代理程序进行集成。您必须输入 1 以选择此集成选项，然后按 Enter 键。

    将注册所选服务器以进行 PMI 资源监视。

12. 如果您是在为传统 WebSphere Application Server 配置数据收集器，请指定是否要在该应用程序服务器实例中配置数据收集器。

    - 输入 1 以在应用程序服务器中配置数据收集器。选择此选项将使该数据收集器与应用程序服务器集成，而这是全面收集操作监视和诊断数据所必需。但是，在应用程序服务器中配置数据收集器需要重新启动应用程序服务器。此外，数据收集器可能会影响服务器性能。
    - 输入 2 表示不在应用程序服务器中配置数据收集器，并前进到步骤 第 716 页的『14』。选择此选项将使数据收集器作为独立进程来运行，而且只能启用资源监视。

13. 出现提示时，请指定是否要允许数据收集器收集诊断数据。输入 1 表示启用诊断数据收集。缺省值为 2。

14. 提示输入 V8 监视代理程序的主机名时，请输入 WebSphere Applications 代理程序的主机名或 IP 地址，或者按 Enter 键以接受缺省值。缺省值对应于您在步骤 3 中选择的值。

    V8 监视代理程序是指随 IBM Cloud Application Performance Management 一起安装的 WebSphere Applications 代理程序。

15. 提示输入 V8 监视代理程序的端口号时，请输入 WebSphere Applications 代理程序的端口号，或者按 Enter 键以接受缺省值。缺省值为 63335。

16. 当询问您是否要为 WebSphere 应用程序配置 V6 监视代理程序时，请按 Enter 键以接受缺省值 No。

    V6 监视代理程序是指 ITCAM Agent for WebSphere Applications，它安装在旧的 IBM® Tivoli® Monitoring 基础结构中。只有在代理程序共存环境中，才需要配置 V6 监视代理程序。

17. 提示输入服务器别名时，请按 Enter 键以接受缺省值，或者输入另一个别名。如果要配置多个应用程序服务器实例，那么配置实用程序会提示输入每个实例的别名。

要点: 别名只能包含以下字符：A-Z、a-z、下划线 (_)、短划线 (-) 以及句点 (.). 请勿在别名中使用其他字符。

服务器别名是 Cloud APM 控制台中显示的代理程序实例名称的第一个限定符（又被称为 MSN）。缺省值是节点名与服务器名称的组合。例如，**node1server1** 别名表示节点 **node1** 中的服务器 **server1**。

18. 提示输入用于 PMI 资源监视的端口号时，请按 Enter 键以接受缺省值，或者输入新的端口号。缺省端口为 63355。

    此端口用于在同一主机上运行的组件之间的内部通信。如果缺省端口在使用中，您可设置另一个端口号。

19. 在 **Support for transaction tracking** 部分中，指定是否启用事务跟踪。输入 1 以启用对事务跟踪的支持。否则，输入 2 并跳至步骤 第 717 页的『22』。

20. 提示输入“事务框架扩展”的主机名或 IP 地址时，请按 Enter 键以接受缺省值，或者输入另一个主机名或 IP 地址。

    “事务框架扩展”是 WebSphere Applications 代理程序的内部组件，用于从数据收集器收集度量。

21. 提示输入数据收集器用于连接到“事务框架扩展”的端口号时，请按 Enter 键以接受缺省值，或者输入另一个端口号。缺省值为 5457。

22. 指定是否将数据收集器与 Application Performance Diagnostics Lite 进行集成。请按 Enter 键以接受缺省值 No。

23. 在 **Advanced settings** 部分中，指定是否要更改垃圾回收日志路径。输入 1 以选择垃圾回收日志路径。否则，输入 2 并跳至步骤 第 717 页的『25』。要使用已在应用程序服务器的 JVM 自变量中指定的日志路径，请输入 2。

24. 指定垃圾回收日志路径。输入带有完整路径的文件名。对于 WebSphere Application Server Liberty，请勿在路径中使用变量。数据收集器通过将服务器实例信息添加到日志文件名来对其进行自动修改。

    例如，如果指定 `gc.log` 作为文件名，那么对于每个已配置的应用程序服务器实例，实际名称将设置为 *`profile_name`*`.`*`cell_name`*`.`*`node_name`*`.`*`server_name`*`.gc.log`。

    要点: 在垃圾回收日志路径中，您可以使用 WebSphere 变量，例如 `${SERVER_LOG_ROOT}`。但是，请勿使用诸如 `%pid` 之类的模板。

25. 查看将应用于指定应用程序服务器实例的数据收集器配置的摘要。如果需要，在应用更改前重新配置数据收集器配置的某些部分。

26. 输入 a 以接受更改。

27. 出现提示时，指定是否要创建当前配置的备份。输入 1 以创建当前配置的备份。否则，输入 2。

    配置实用程序将应用更改并显示一条状态消息，指示针对概要文件的数据收集器配置操作已完成。

28. 如果您是在为传统 WebSphere Application Server 配置数据收集器，请重新启动应用程序服务器实例或重新启动代理程序，具体取决于您在步骤 第 716 页的『12』 中的选择。

    - 如果已在应用程序服务器中启用数据收集器，请按配置实用程序的指示重新启动应用程序服务器实例。
    - 如果已启用 PMI 资源监视，而未在应用程序服务器中启用数据收集器，请通过运行以下命令重新启动 WebSphere Applications 代理程序：
      - Windows

        ```
        cd install_dir\bin
        was-agent.bat stop
        was-agent.bat start
        ```

      - Linux AIX

        ```
        cd install_dir/bin
        ./was-agent.sh
        stop
        ./was-agent.sh start
        ```

    数据收集器配置将在应用程序服务器或代理程序重新启动后生效。

29. 登录 Cloud APM 控制台，以在仪表板中查看数据。
    a) 使用提醒您服务已就绪的电子邮件中提供的链接来访问控制台。另外，也可以从 IBM Marketplace Web 站点访问控制台。有关详细的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
    b) 使用“应用程序”编辑器，将受监视应用程序服务器添加到 应用程序性能仪表板。您可将其作为新组件添加到现有应用程序，或创建应用程序以包含此组件。
    有关“应用程序”编辑器的更多信息，请参阅第 930 页的『管理应用程序』。

下一步做什么

- 如果用来配置数据收集器的当前用户标识并非运行应用程序服务器的用户标识，请确认对于数据收集器主目录中的 `runtime` 和 `logs` 目录，用于配置数据收集器的用户标识具有读写许可权。这两个子目录是在服务器重新启动时，由运行应用程序服务器的用户标识创建。
- 登录到 Cloud APM 控制台以在仪表板中查看监视数据。如果监视数据尚不可用，请通过运行下列命令来重新启动 WebSphere Applications 代理程序：
  - Windows

```
cd install_dir\bin
was-agent.bat stop
was-agent.bat start
```

  - Linux AIX

```
cd install_dir/bin
./was-agent.sh
stop
./was-agent.sh start
```

- 更改服务器别名将更改向 Cloud APM 控制台注册的代理程序实例名称。如果这不是您第一次配置数据收集器，并且已更改服务器别名，那么必须通过完成下列步骤来清除一些高速缓存文件：
  1. 停止监视代理程序（如果它正在运行）。
  2. 使用文本编辑器打开下列目录中的 *hostname*_yn.xml 文件，其中 *hostname* 是安装了 WebSphere Applications 代理程序的主机的名称。
     - Windows *install_dir*\TMAITM6_x64（缺省值为 C:\IBM\APM\TMAITM6_x64）
     - Linux AIX *install_dir*/config（缺省值为 /opt/ibm/apm/agent/config）
  3. 找到以下列字符串开头并包含先前服务器名称的行。例如，was85.win4net01Cell02.win4net01Node02.AppSrv01.server1，其中 server1 是先前的应用程序服务器名称。

```
<!ENTITY was_product_code.cellname.nodename.profilename.servername
```

     其中，*was_product_code* 是 WebSphere Application Server 的产品代码；*cellname* 是单元格的名称；*nodename* 是节点名；*profilename* 是应用程序服务器概要文件的名称；*servername* 是先前的应用程序服务器名称。
  4. 在当前目录中找到该行中指示的 .XML 文件，并删除该文件。
  5. 从 *hostname*_yn.xml 文件中移除您在步骤 3 中找到的行。
  6. 在 *hostname*_yn.xml 文件的结尾，移除包含先前服务器名称的行。
  7. 保存并关闭该文件。
  8. 重新启动监视代理程序。

以交互方式重新配置数据收集器

将数据收集器配置为监视一个或多个应用程序服务器实例之后，可以使用重新配置实用程序（reconfig.sh 或 reconfig.bat）来重新配置数据收集器。

开始之前

如果您要配置数据收集器以监视 WebSphere Application Server Liberty，请将 **JAVA_HOME** 系统环境变量设置为用于应用程序服务器的 JVM。例如，在 Windows 系统上，将 **JAVA_HOME** 值设置为 `C:\Program Files\IBM\java`。或者，在 Linux 系统上，运行 `export JAVA_HOME=/opt/IBM/java`。

关于此任务

使用下列完全重新配置实用程序来配置数据收集器：

- Windows *install_dir*\dchome\7.3.0.14.08\bin\reconfig.bat
- Linux AIX *install_dir*/yndchome/7.3.0.14.08/bin/reconfig.sh

切记: 在下列情况下，**reconfig** 实用程序不适用。请改为使用 **config** 配置实用程序。虽然 **config** 实用程序会发出警告，指出服务器已配置完成，但它仍可以进行所需的任何更改。

- 数据收集器只是已完成资源监视配置，而您想要重新配置数据收集器。
- 您想要为 WebSphere Portal Server 重新配置数据收集器。

提示: 在请求代理程序配置设置的提示中，重新配置实用程序会提供当前配置的值作为缺省值。

过程

要通过对提示做出响应来重新配置数据收集器，请完成下列步骤：

1. 以用来安装应用程序服务器的用户标识身份登录系统。
2. 转至 *dc_home* 数据收集器主目录中的 bin 目录。
3. 通过发出以下命令启动重新配置实用程序：
   - Windows `reconfig.bat`
   - Linux AIX `./reconfig.sh`

   提示: 运行此重新配置实用程序相当于运行带 `-reconfig` 参数的 `config.bat` 脚本（在 Windows 系统上）或运行带 `-reconfig` 参数的 `config.sh` 脚本（在 Linux 或 AIX 系统上）。

   该重新配置实用程序显示本地计算机系统上找到的所有网卡的 IP 地址。
4. 输入与要使用的 IP 地址对应的编号。

   该重新配置实用程序显示此主机上为其配置了数据收集器的所有应用程序服务器实例，并提示您从列表中选择一个或多个应用程序服务器实例。
5. 从列表选择一个或多个应用程序服务器实例。请输入要重新配置数据收集的应用程序服务器实例的相应编号，或者输入星号 (*) 以便针对所有应用程序服务器实例重新配置数据收集。要指定服务器的子集，请输入代表这些服务器的编号（由逗号分隔）。例如：`1,2,3`。

   切记:

   - 对于单机环境，必须在配置期间运行应用程序服务器实例。（不需要运行 WebSphere Application Server Liberty 实例）。
   - 对于 Network Deployment 环境，Deployment Manager 必须正在运行。
   - 请确保您选择的应用程序服务器实例是托管了您想要监视的应用程序或服务的实际服务器。
6. 在 **Integration with Agent for WebSphere Applications** 部分中，指定您希望将数据收集器与 WebSphere Applications 代理程序进行集成。您必须输入 1 以选择此集成选项，然后按 Enter 键。
7. 如果您是在为传统 WebSphere Application Server 配置数据收集器，请指定是否要在该应用程序服务器实例中配置数据收集器。
   - 输入 1 表示在应用程序服务器中配置数据收集器。选择此选项将使该数据收集器与应用程序服务器集成，而这是全面收集操作监视和诊断数据所必需。但是，在应用程序服务器中配置数据收集器需要重新启动应用程序服务器。此外，数据收集器可能会影响服务器性能。
   - 输入 2 表示不在应用程序服务器中配置数据收集器，并前进到步骤 第 720 页的『9』。选择此选项将使数据收集器作为独立进程来运行，而且只能启用 PMI 资源监视。

8. 出现提示时，请指定是否要对数据收集器启用诊断数据收集。输入 1 表示“是”，或者输入 2 表示“否”。
9. 提示输入主机名时，请输入 WebSphere Applications 代理程序的主机名或 IP 地址，或者按 Enter 键以接受缺省值。缺省值对应于您在步骤 第 719 页的『4』 中所做的选择。
10. 提示输入端口号时，请输入监视代理程序的端口号，或者按 Enter 键以接受缺省值。缺省值为 63335。
11. 当询问您是否要为 WebSphere 应用程序配置 V6 监视代理程序时，请按 Enter 键以接受缺省值 No。

    V6 监视代理程序是指 ITCAM Agent for WebSphere Applications，它安装在旧的 IBM® Tivoli® Monitoring 基础结构中。只有在代理程序共存环境中，才需要配置 V6 监视代理程序。
12. 提示输入服务器别名时，请按 Enter 键以接受缺省值，或者输入另一个别名。如果要配置多个应用程序服务器实例，那么配置实用程序会提示输入每个实例的别名。

    要点: 别名只能包含以下字符：A-Z、a-z、下划线 (_)、短划线 (-) 以及句点 (.). 请勿在别名中使用其他字符。

    服务器别名是 Cloud APM 控制台中显示的代理程序实例名称的第一个限定符（又被称为 MSN）。缺省值是节点名与服务器名称的组合。例如，**node1server1** 别名表示节点 **node1** 中的服务器 **server1**。
13. 提示输入用于 PMI 资源监视的端口号时，请按 Enter 键以接受缺省值，或者输入新的端口号。缺省端口为 63355。

    此端口用于在同一主机上运行的组件之间的内部通信。如果缺省端口在使用中，您可设置另一个端口号。
14. 在 **Support for transaction tracking** 部分中，指定是否启用事务跟踪。输入 1 以启用对事务跟踪的支持。否则，输入 2 并跳至步骤 第 720 页的『17』。
15. 提示输入“事务框架扩展”的主机名或 IP 地址时，请按 Enter 键以接受缺省值，或者输入另一个主机名或 IP 地址。

    “事务框架扩展”是 WebSphere Applications 代理程序的内部组件，用于从数据收集器收集度量。
16. 提示输入数据收集器用于连接到“事务框架扩展”的端口号时，请按 Enter 键以接受缺省值，或者输入另一个端口号。缺省值为 5457。
17. 指定是否将数据收集器与 Application Performance Diagnostics Lite 进行集成。请按 Enter 键以接受缺省值 No。
18. 在 **Advanced settings** 部分中，指定是否要更改垃圾回收日志路径。

    输入 1 以选择垃圾回收日志路径。否则，输入 2 并跳至步骤 第 720 页的『20』。要使用已在应用程序服务器的 JVM 自变量中指定的日志路径，请输入 2。
19. 指定垃圾回收日志路径。输入带有完整路径的文件名。对于 WebSphere Application Server Liberty，请勿在路径中使用变量。数据收集器通过将服务器实例信息添加到日志文件名来对其进行自动修改。

    例如，如果指定 gc.log 作为文件名，那么对于每个已配置的应用程序服务器实例，实际名称将设置为 *profile_name*.*cell_name*.*node_name*.*server_name*.gc.log。

    要点: 在垃圾回收日志路径中，您可以使用 WebSphere 变量，例如 ${SERVER_LOG_ROOT}。但是，请勿使用诸如 %pid 之类的模板。
20. 查看将应用于指定应用程序服务器实例的数据收集器配置的摘要。如有需要，可在应用更改前重新配置数据收集器配置的某些部分。
21. 输入 a 以接受更改。
22. 出现提示时，指定是否要创建当前配置的备份。输入 1 以创建当前配置的备份。否则，输入 2。

    配置实用程序将应用更改并显示一条状态消息，指示针对概要文件的数据收集器配置操作已完成。
23. 如果您是在为传统 WebSphere Application Server 配置数据收集器，请重新启动应用程序服务器实例或重新启动代理程序，具体取决于您在步骤 第 719 页的『7』 中的选择。
    - 如果已在应用程序服务器中启用数据收集器，请按配置实用程序的指示重新启动应用程序服务器实例。
    - 如果已启用 PMI 资源监视，而未在应用程序服务器中启用数据收集器，请通过运行以下命令重新启动 WebSphere Applications 代理程序：

– Windows

```
cd install_dir\bin
was-agent.bat stop
was-agent.bat start
```

– Linux AIX

```
cd install_dir/bin
./was-agent.sh
stop
./was-agent.sh start
```

数据收集器配置将在应用程序服务器或代理程序重新启动后生效。

下一步做什么

· 更改服务器别名将更改向 Cloud APM 控制台注册的代理程序实例名称。如果您在重新配置过程中更改了服务器别名，那么必须通过完成下列步骤来清除一些高速缓存文件：

1. 停止监视代理程序（如果它正在运行）。
2. 使用文本编辑器打开下列目录中的 *hostname*_yn.xml 文件，其中 *hostname* 是安装了 WebSphere Applications 代理程序的主机的名称。
   – Windows *install_dir*\TMAITM6_x64（缺省值为 C:\IBM\APM\TMAITM6_x64）
   – Linux AIX *install_dir*/config（缺省值为 /opt/ibm/apm/agent/config）
3. 找到以下列字符串开头并包含先前服务器名称的行。例如，was85.win4net01Cell02.win4net01Node02.AppSrv01.server1，其中 server1 是先前的应用程序服务器名称。

   ```
   <!ENTITY was_product_code.cellname.nodename.profilename.servername
   ```

   其中，*was_product_code* 是 WebSphere Application Server 的产品代码；*cellname* 是单元格的名称；*nodename* 是节点名；*profilename* 是应用程序服务器概要文件的名称；*servername* 是先前的应用程序服务器名称。
4. 在当前目录中找到该行中指示的 .XML 文件，并删除该文件。
5. 从 *hostname*_yn.xml 文件中移除您在步骤 3 中找到的行。
6. 在 *hostname*_yn.xml 文件的结尾，移除包含先前服务器名称的行。
7. 保存并关闭该文件。
8. 重新启动监视代理程序。

以静默方式配置数据收集器

如果您希望配置多个应用程序服务器实例，那么以静默方式配置数据收集器可能更为方便。

关于此任务

如果您以静默方式配置数据收集器，那么首先在属性文件中指定配置选项。随配置实用程序一起打包了样本属性文件 sample_silent_config.txt。该文件位于以下目录中，其中 *dc_home* 是数据收集器的安装目录。有关 *dc_home* 目录的完整路径，请参阅为 WebSphere 应用程序代理程序配置数据收集器中的简介。

· Windows *dc_home*\bin

· Linux AIX *dc_home*/bin

有关此文件中所有可用配置属性的详细信息，请参阅第 723 页的『用于静默配置数据收集器的属性文件』。

过程

完成下列步骤以执行静默配置：

1. 在属性文件中指定配置选项。您可以复制样本属性文件，并更改必需选项。
2. 运行实用程序前，设置 Java 主目录的位置。
   例如：
   - Windows
     ```
     set JAVA_HOME=C:\Program Files\IBM\WebSphere\AppServer80\java
     ```
   - Linux AIX
     ```
     export JAVA_HOME=/opt/IBM/AppServer80/java
     ```

   要点： 如果您正在为 WebSphere Application Server Liberty 配置监视，那么必须使用用于应用程序服务器的 JVM 版本。否则，监视可能会失败。
3. 转至以下目录：
   - Windows *dc_home*\bin
   - Linux AIX *dc_home*/bin
4. 运行此命令，以静默方式配置数据收集器。
   - Windows 以安装 WebSphere Application Server 的管理员身份运行以下命令。
     ```
     config.bat -silent [dir_path]\silent file
     ```
   - Linux AIX 使用 root 用户特权运行以下命令。
     ```
     config.sh -silent [dir_path]/silent file
     ```

   提示： 如果使用 wsadmin 用户安装了应用程序务器，请以 wsadmin 用户身份或使用 root 用户特权运行 config 实用程序。
5. 在配置数据收集器以监视应用程序服务器实例后，如果已在应用程序服务器中启用数据收集器，那么必须重新启动这些实例。重新启动应用程序服务器实例之后，数据收集器配置会生效。
6. 如果已启用 PMI 资源监视，而未在应用程序服务器中启用数据收集器，那么可能需要重新启动 WebSphere Applications 代理程序以启动监视。如果监视数据尚不可用，请通过运行以下命令来重新启动监视代理程序：
   - Windows
     ```
     cd install_dir\bin
     was-agent.bat stop
     was-agent.bat start
     ```
   - Linux AIX
     ```
     cd install_dir/bin
     ./was-agent.sh
     stop
     ./was-agent.sh start
     ```

下一步做什么

完成静默配置后，要重新配置数据收集器，您有两个选项：

- 使用 **reconfig** 重新配置实用程序以交互方式进行重新配置。有关指示信息，请参阅第 718 页的『以交互方式重新配置数据收集器』。
- 以静默方式进行取消配置，然后再次使用同一过程以静默方式进行配置。有关指示信息，请参阅第 127 页的『以静默方式取消配置数据收集器』。

相关参考

第 723 页的『用于静默配置数据收集器的属性文件』

要静默配置数据收集器，首先在属性文件中指定配置选项，然后运行配置实用程序。

用于静默配置数据收集器的属性文件

要静默配置数据收集器，首先在属性文件中指定配置选项，然后运行配置实用程序。

创建属性文件后，请记住以下注意事项：

- 文件中以数字符号 # 开头的行将被视为注释，不予处理。如果行中其他地方使用到该数字符号，将不被视为注释的开始。
- 每个属性都在单独的行中描述，格式为：*property* = *value*。

property
属性的名称。第 723 页的表 223 中显示了可以配置的有效属性列表。

值
属性值。提供了部分属性的缺省值。您可以删除缺省值，以便将属性值保留为空。与使用缺省值相反，空值将视为未指定该属性。如果要使用缺省值，在文件中注释掉属性即可。

- 密码为纯文本。
- 属性及其值区分大小写。

第 723 页的表 223 描述以静默方式配置数据收集器时可用的属性。

要点: 如果正在配置 WebSphere Application Server Liberty 实例的数据收集器，那么不会使用某些属性。

表 *223.* 以静默方式运行配置实用程序时的可用属性

属性	**注释**
default.hostip	如果计算机系统使用多个 IP 地址，请指定供数据收集器使用的 IP 地址。
数据收集器与 ITCAM for Application Diagnostics Managing Server 的集成 **要点:** 仅当您有 ITCAM for Application Diagnostic 时，Managing Server 才可用。 对于 WebSphere Application Server Liberty 实例或在 Cloud APM 环境中，不使用这些属性。	
ms.connect	指定是否配置数据收集器以连接到 ITCAM for Application Diagnostics 环境中的 Managing Server。有效值为 `True` 和 `False`。
ms.kernel.host	指定 Managing Server 的标准主机名。
ms.kernel.codebase.port	指定 Managing Server 正在侦听的代码库端口。
ms.am.home	指定 Managing Server 的主目录。
ms.am.socket.bindip	指定数据收集器用于与 Managing Server 进行通信的 IP 地址或主机名。如果在数据收集器计算机系统上配置了多个网络接口或 IP 地址，请选择其中一个。
ms.probe.controller.rmi.port	如果数据收集器位于防火墙之后或者您对更改数据收集器的控制器 RMI 端口有特殊要求，请设置此端口号范围。在防火墙的许可内配置数据收集器主机的此端口号。例如： `ms.probe.controller.rmi.port=8300-8399` 或 ms.probe.controller.rmi.port=8300。
ms.probe.rmi.port	如果数据收集器位于防火墙之后，或者您对更改数据收集器的 RMI 端口有特殊要求，请设置此端口号范围。在防火墙的许可内配置数据收集器主机的此端口号。例如：`ms.probe.rmi.port=8200-8299` 或 `ms.probe.rmi.port=8200`。
对事务跟踪的支持 要查看事务跟踪信息，您必须在 Cloud APM 控制台中提供拓扑视图并在“代理程序配置”窗口中启用事务跟踪。	
ttapi.enable	指定数据收集器是否支持事务跟踪。有效值为 `True` 和 `False`。

表 223. 以静默方式运行配置实用程序时的可用属性 (续)

属性	注释
ttapi.host	指定事务框架扩展的主机，此事务框架扩展是 Monitoring Agent for WebSphere Applications 中从数据收集器收集度量的组件。使用本地主机值：127.0.0.1。
ttapi.port	指定事务框架扩展的端口。使用 5457。
数据收集器与 ITCAM for SOA 的集成 **要点:** 对于 WebSphere Application Server Liberty 实例或在 Cloud APM 环境中，不使用此属性。	
soa.enable	指定是否将数据收集器与 ITCAM for SOA 进行集成。必须安装 ITCAM for SOA 代理程序才能完成配置。
数据收集器与 Tivoli Performance Monitoring 的集成 **要点:** 对于 WebSphere Application Server Liberty 实例或在 Cloud APM 环境中，不使用此属性。	
tpv.enable	指定当数据收集器软件包含在 ITCAM for WebSphere Application Server V8.5 中时，是否将数据收集器与 Tivoli Performance Monitoring 进行集成。使用 WebSphere Application Server 管理控制台来访问 Tivoli Performance Monitoring。有效值为 *True* 和 *False*。
数据收集器与 Application Performance Diagnostics Lite 的集成 **要点:** 针对 WebSphere Application Server Liberty 实例，不会使用此属性。	
de.enable	指定是否收集 Application Performance Diagnostics 和 Application Performance Diagnostics Lite 所必需的诊断数据。有效值为 `True` 和 `False`。 如果您有 Application Diagnostics 或将来可能有此产品，请启用此集成。在此情况下，服务器启动时会启用诊断数据收集。否则，会在启动时将其禁用；您可以在用户界面中使用“代理程序配置”页面将其启用，但是如果服务器重新启动，那么会再次禁用诊断数据收集。 此设置还会启用与 Application Performance Diagnostics Lite 的集成，这是用于对 WebSphere Application Server 和 WebSphere Portal Server 上运行的应用程序进行诊断调查的工具。使用此工具，您可以实时分析数据，也可以将诊断信息保存到文件中以供将来分析。
PMI 资源和数据收集器监视 始终对所选服务器配置了资源 (PMI) 监视，而无需对应用程序服务器进行任何更改。此监视选项提供了有限的度量值并仅适用于 WebSphere Applications 代理程序，但不需要重新启动应用程序服务器，也不会影响性能。	
tema.appserver	指定您是否要在应用程序服务器实例中配置数据收集器。需要应用程序服务器实例中的数据收集器才能获取 WebSphere Applications 代理程序中的整套度量以及与任何其他产品集成。但是，配置数据收集器需要重新启动应用程序服务器。此外，数据收集器可能会影响服务器性能。有效值为 `True` 和 `False`。 如果此参数设置为 `False`，那么将忽略用于和 WebSphere Applications 代理程序以外的产品进行集成的数据收集器配置参数。当该参数设置为 `False` 时，诊断和事务跟踪功能不可用，只会收集资源监视数据。
tema.jmxport	用于资源监视的 TCP/IP 端口号。此端口用于在相同主机上运行的组件之间的内部通信。缺省端口为 63355；如果正在使用此端口，可以设置其他端口号。

表 223. 以静默方式运行配置实用程序时的可用属性 (续)

属性	注释
数据收集器与 WebSphere Applications 代理程序的监视代理程序组件的集成以及与 Application Performance Diagnostics Lite 的集成	
temaconnect	指定数据收集器是否连接到 WebSphere Applications 代理程序的监视代理程序组件。有效值为 `True` 和 `False`。 **要点:** 您必须使用 `True` 值才能使用 WebSphere Applications 代理程序。
tema.appserver	指定您是否要在应用程序服务器实例中配置数据收集器。需要应用程序服务器实例中的数据收集器才能获取 WebSphere Applications 代理程序中的整套度量以及与任何其他产品集成。但是，进行此配置需要重新启动应用程序服务器。此外，数据收集器可能会影响服务器性能。有效值为 `True` 和 `False`。 如果此参数设置为 `False`，那么将忽略用于将数据收集器与 WebSphere Applications 代理程序之外的产品进行集成的配置参数以及后续 tema.host 和 tema.port 参数。当该参数设置为 `False` 时，诊断和事务跟踪功能不可用，只会收集资源监视数据。
tema.host	指定 WebSphere Applications 代理程序的监视代理程序组件的标准主机名称或 IP 地址。使用本地主机地址 (127.0.0.1)。
tema.port	指定 WebSphere Applications 代理程序的监视代理程序组件的端口号。请勿更改缺省值 63335。
tema.jmxport	用于资源监视的 TCP/IP 端口号。此端口用于在相同主机上运行的组件之间的内部通信。缺省端口为 63355；如果正在使用此端口，可以设置其他端口号。
数据收集器与 ITCAM Agent for WebSphere Applications V6 的集成 使用下列属性可以将一个数据收集器配置为收集 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications V6 的数据。	
config.tema.v6	指定是否将数据收集器与 ITCAM Agent for WebSphere Applications V6 的监视代理程序组件进行集成。有效值为 `True` 和 `False`。缺省值为 `False`。
tema.host.v6	指定是否将数据收集器与 ITCAM Agent for WebSphere Applications V6 的监视代理程序组件进行集成。有效值为 `True` 和 `False`。缺省值为 `False`。
tema.port.v6	指定 ITCAM Agent for WebSphere Applications V6 的监视代理程序组件的端口号。请不要更改缺省值 63336。
WebSphere Application Server 备份	
was.backup.configuration	指定是否在应用新配置前对 WebSphere Application Server 配置的当前配置进行备份。有效值为 `True` 和 `False`。
was.backup.configuration.dir	指定备份目录的位置。
高级配置设置	
was.gc.custom.path	指定是否为垃圾回收日志设置定制路径。

表 223. 以静默方式运行配置实用程序时的可用属性 (续)

属性	注释
was.gc.file	指定到定制垃圾回收日志的路径。将此值设置为带有全路径的文件名。数据收集器通过将服务器实例信息添加到日志文件名来对其进行自动修改。例如，如果指定 gc.log 作为文件名，那么对于每个已配置的应用程序服务器实例，实际名称将设置为 *profile_name.cell_name.node_name.server_name*.gc.log。 **要点:** 在垃圾回收日志路径中，可以使用 WebSphere 变量，例如 ${SERVER_LOG_ROOT}。但是，请勿使用模板，例如 %pid。
WebSphere Administrative Services 连接设置	
was.wsadmin.connection.host	指定 wsadmin 工具正在连接到的主机名。在 Network Deployment 环境中将 wsadmin 连接指定给部署管理器。在单机环境中将 wsadmin 连接指定给该服务器。 **切记:** 如果 WebSphere 管理控制台位于同一系统上，将使用 localhost 的值进行连接。但是，在某些情况下，系统网络或安全设置决定了不允许使用 localhost 进行通信。在此情况下，您必须在静默响应文件中指定该参数。
was.wsadmin.connection.type	指定 wsadmin 工具连接到 WebSphere Application Server 必须使用的端口。
was.wsadmin.connection.port	指定 wsadmin 工具连接到 WebSphere Application Server 必须使用的端口。
WebSphere Application Server 全局安全性设置	
was.wsadmin.username	指定有权登录 IBM WebSphere Application Server 管理控制台的用户的用户标识。此用户在应用程序服务器上必须具有代理角色。
was.wsadmin.password	指定与在 was.wsadmin.username 属性中指定的用户对应的密码。
was.client.props	指定是否从客户机属性文件检索安全性设置。可能的值为 True 或 False。
WebSphere Application Server 设置	
was.appserver.profile.name	指定希望配置的应用程序服务器概要文件的名称。不用于 WebSphere Application Server Liberty。
was.appserver.home	指定 WebSphere Application Server 主目录。
was.appserver.cell.name	指定 WebSphere Application Server 单元名称。不用于 WebSphere Application Server Liberty。
was.appserver.node.name	指定 WebSphere Application Server 节点名。不用于 WebSphere Application Server Liberty。
WebSphere Application Server 运行时实例设置	
was.appserver.server.name	指定要配置的应用程序服务器概要文件内的应用程序服务器实例。 **提示:** · 静默响应文件可具有该属性的多个实例 · 添加第二个服务器时，取消注释第二个服务器（即 #[SERVER]），并添加服务器名称。

表 223. 以静默方式运行配置实用程序时的可用属性 (续)	
属性	**注释**
tema.serveralias	指定监视用户界面中包含此应用程序服务器实例的监视信息的节点的名称。缺省值是节点名与服务器名称的组合。 **要点:** 别名只能包含以下字符：A-Z、a-z、下划线 (_)、短划线 (-) 以及句点 (.). 请勿在别名中使用其他字符。 **提示:** 静默响应文件可具有该属性的多个实例。 **切记:** 更改服务器别名将更改向 Cloud APM 控制台注册的代理程序实例名称。如果这不是您第一次配置数据收集器，并且已更改服务器别名，那么您必须清除一些高速缓存文件。有关详细指示信息，请参阅清除具有旧服务器名称的高速缓存文件。

在配置实用程序失败时手动配置数据收集器

如果您无法使用所提供的配置实用程序来为 WebSphere Applications 代理程序配置数据收集器，您可在 WebSphere 管理控制台中手动配置数据收集器。

开始之前

- 安装 WebSphere Applications 代理程序。
- 了解数据收集器配置所需的数据收集器主目录。在 Linux 和 UNIX 系统上，缺省值为 `/opt/ibm/apm/agent/yndchome/7.3.0.14.08`；在 Windows 系统上，缺省值为 `C:\IBM\APM\dchome\7.3.0.14.08`。
- 如果您想要为 Liberty 服务器配置数据收集器，请了解 Liberty 服务器主目录。例如，`/opt/ibm/was/liberty/usr/servers/defaultServer`。
- 请确保 *dc_home*`/runtime/wsBundleMetaData` 文件夹中存在名为 `itcam_wsBundleMetaData.xml` 的文件，而且该文件中包含以下内容。如果该文件夹或文件不存在，请进行手动创建。

  切记: *plugins_dir_within_dc_home* 值必须设置为数据收集器主目录中 `plugins` 文件夹的绝对路径。在 Linux 和 UNIX 系统上，缺省值为 `/opt/ibm/apm/agent/yndchome/7.3.0.14.08/plugins`，而在 Windows 系统上，缺省值为 `C:\IBM\APM\dchome\7.3.0.14.08\plugins`。

```
<bundles>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.bundlemanager_7.2.0.jar</bundle>
  </directory>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.classicsca_7.2.0.jar</bundle>
  </directory>
  <directory path="plugins_dir_within_dc_home">
      <bundle>com.ibm.tivoli.itcam.toolkitsca.classicsca_7.2.0.jar</bundle>
  </directory>
</bundles>
```

关于此任务

要点:

- 必须以 WebSphere 管理用户身份手动更改 WebSphere Application Server 的数据收集器配置。
- 您必须是经验丰富的 WebSphere 管理员才可以针对数据数据对 WebSphere Application Server 进行手动更改。手动配置更改中的错误可能导致应用程序服务器无法启动。
- 手动配置数据收集器以监视应用程序服务器实例之后，无法使用取消配置实用程序来取消配置数据收集器。您必须改为手动取消配置数据收集器。

过程

- 要为 WebSphere Application Server 手动配置数据收集器，请参阅第 728 页的『为 WebSphere Application Server traditional 手动配置数据收集器』。
- 要为 Liberty 服务器手动配置数据收集器，请参阅第 729 页的『为 WebSphere Application Server Liberty 手动配置数据收集器』。

为 *WebSphere Application Server traditional* 手动配置数据收集器

过程

1. 以管理员身份登录 WebSphere 管理控制台。
2. 在导航窗格中，单击**服务器**，展开**服务器类型**，然后单击 **WebSphere Application Server**。
3. 在“配置”选项卡中的**服务器基础结构**部分下，展开 **Java 和进程管理**，并单击**进程定义**。
4. 在**其他属性**部分中，单击 **Java 虚拟机**。
5. 在**通用 JVM 参数**字段中，添加以下项。

   ```
   -agentlib:am_ibm_16=${WAS_SERVER_NAME} -Xbootclasspath/p:${ITCAMDCHOME}/
   toolkit/lib/bcm-bootstrap.jar -Djava.security.policy=${ITCAMDCHOME}/itcamdc/
   etc/datacollector.policy -verbosegc
   ```

   添加条目时，请注意以下事项：

   · 所有条目都必须位于单独一行上。

   · 在减号 (-) 前用空格分隔不同参数，其他位置请勿使用空格。
6. 单击**应用**，然后将更改保存至主配置。

   · 如果您不是处于 Network Deployment 环境中，那么单击**保存**。

   · 如果您处于 Network Deployment 环境中，请确保在**控制台首选项**选项中选中**与节点同步更改**，然后单击**保存**。
7. 在导航窗格中，单击**服务器**，展开**服务器类型**，单击 **WebSphere Application Server**，然后单击服务器名称。
8. 在“配置”选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **环境条目**。
9. 根据操作系统、硬件平台和应用程序服务器 JVM，设置以下环境条目。

表 *224.* 环境条目

平台	环境条目名称	环境条目值
AIX R6.1（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536
AIX R7.1（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536
Solaris 10（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/sol296
Solaris 11（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/sol296
Linux Intel R2.6（32 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/li6263
Linux x86_64 R2.6（64 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx8266
Linux on Power Little Endian（64 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpl266

表 *224.* 环境条目 *(续)*		
平台	**环境条目名称**	**环境条目值**
Linux on System z（32 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/ls3263
Linux on System z（64 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/ls3266
Windows（32 位 JVM）	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win32
Windows（64 位 JVM）	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win64

10. 单击**应用**，然后将更改保存至主配置。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
    - 如果您处于 Network Deployment 环境中，请确保在**控制台首选项**选项中选中**与节点同步更改**，然后单击**保存**。
11. 在导航窗格中，单击**环境** > **WebSphere 变量**。
12. 将作用域指定为相应的服务器级别，并添加 *ITCAMDCHOME* 变量。将 *ITCAMDCHOME* 变量值设置为数据收集器主目录。例如，/opt/ibm/apm/agent/yndchome/7.3.0.14.08。
13. 单击**应用**，然后将更改保存至主配置。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
    - 如果您处于 Network Deployment 环境中，请确保在**控制台首选项**选项中选中**与节点同步更改**，然后单击**保存**。
14. 重新启动应用程序服务器。

结果

现在，将 Cloud APM 控制台添加到应用程序之后，您可在这个应用程序组件中检查 WebSphere Applications 代理程序数据。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关如何添加或编辑应用程序的指示信息，请参阅第 930 页的『管理应用程序』。

下一步做什么

手动配置数据收集器之后，无法使用所提供的 unconfig 实用程序来取消配置数据收集器。请改为手动取消配置数据收集器。有关指示信息，请参阅第 131 页的『手动取消配置数据收集器』。

为 *WebSphere Application Server Liberty* 手动配置数据收集器

过程

1. 浏览至 Liberty 服务器主目录。例如，/opt/ibm/wlp/usr/servers/defaultServer。
2. 通过添加下列参数，编辑 jvm.options 文件，其中，*dc_home* 是数据收集器主目录，*server_name* 是 Liberty 服务器名称。如果 jvm.options 文件不存在，请使用文本编辑器进行创建。

```
-agentlib:am_ibm_16=server_name
-Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
-verbosegc
```

添加条目时，请注意以下事项：

- 每个条目都必须独占一行。
- 将 *server_name* 替换为实际的 Liberty 服务器名称。例如，defaultServer。

· 将 *dc_home* 替换为实际的数据收集器主目录。例如，`/opt/ibm/apm/agent/yndchome/7.3.0.14.08`。

3. 打开同一目录中的 `server.env` 文件，并根据操作系统，将下列路径添加到环境条目中，其中 *dc_home* 是数据收集器主目录。如果 `server.env` 文件不存在，请使用文本编辑器进行创建。

表 *225.* 环境条目

平台	环境条目名称	环境条目值
AIX R6.1（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
AIX R7.1（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/aix536
Solaris 10（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Solaris 11（64 位 JVM）	LIBPATH	/lib:*dc_home*/toolkit/lib/sol296
Linux x86_64 R2.6（64 位 JVM）	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/lx8266
Linux Intel R2.6（32 位 JVM）	LD_LIBRARY_PATH	/lib:*dc_home*/toolkit/lib/li6263
Windows（32 位 JVM）	PATH	/lib;*dc_home*/toolkit/lib/win32
Windows（64 位 JVM）	PATH	/lib;*dc_home*/toolkit/lib/win64

4. 打开同一目录中的 `server.xml` 文件并添加下列几行，以启用监视功能：

```
<featureManager>
              <feature>webProfile-7.0</feature>
              <feature>monitor-1.0</feature>
              <feature>usr:itcam-730.140</feature>
   </featureManager>
```

5. 重新启动 Liberty 服务器。

结果

现在，将 Cloud APM 控制台添加到应用程序之后，您可在这个应用程序组件中检查 WebSphere Applications 代理程序数据。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关如何添加或编辑应用程序的指示信息，请参阅第 930 页的『管理应用程序』。

下一步做什么

手动配置数据收集器之后，无法使用所提供的 `unconfig` 实用程序来取消配置数据收集器。请改为手动取消配置数据收集器。有关指示信息，请参阅第 131 页的『手动取消配置数据收集器』。

（代理程序共存）配置 WebSphere Applications 代理程序和数据收集器

在同时安装有 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications 的代理程序共存环境中，您必须为代理程序执行一些额外配置，并遵循不同的过程来配置数据收集器。

关于此任务

在代理程序共存环境中，您只需配置一个数据收集器，即可将数据同时发送到 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications。这两个代理程序必须使用不同的端口侦听来自数据收集器的请求。

过程

1. 如果环境中存在 ITCAM Agent for WebSphere Applications 的数据收集器（安装在旧 IBM® Tivoli® Monitoring 基础结构中），请将其卸载。
2. 安装 IBM Cloud Application Performance Management 8.1.3 或更高版本中提供的 WebSphere Applications 代理程序。这可确保安装代理程序共存所支持的数据收集器 V7.3.0.11.0 或更高版本。
3. 确保用于配置数据收集器的用户标识，以及安装应用程序服务器的用户标识，具有代理程序所需的适当用户特权。有关指示信息，请参阅第 711 页的『检查用户访问需求』。
4. 确保 WebSphere Applications 代理程序与 ITCAM Agent for WebSphere Applications 使用不同的端口号来侦听来自数据收集器的请求。这些端口号必须是唯一的，不可由环境中的任何其他组件使用。必要时，重新配置代理程序以更改端口。
   - 有关如何配置 WebSphere Applications 代理程序的信息，请参阅第 731 页的『配置 WebSphere Applications 代理程序』。
   - 有关如何配置 ITCAM Agent for WebSphere Applications 的信息，请参阅 ITCAM for Application Diagnostics 或 ITCAM for Applications 文档。
5. 使用提供的配置实用程序来配置数据收集器。有关指示信息，请参阅第 732 页的『针对代理程序共存环境配置数据收集器』。

   提示: 如果您熟悉数据收集器配置，那么也可以静默方式来配置数据收集器。有关指示信息，请参阅第 721 页的『以静默方式配置数据收集器』。

配置 *WebSphere Applications* 代理程序

在代理程序共存环境中，数据收集器同时由 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications 共享。这两个代理程序必须使用不同的端口侦听来自数据收集器的请求。必要时，您必须配置代理程序以更改端口。

关于此任务

- 在 Linux 或 AIX 系统上，您可以通过运行配置脚本并对提示做出响应来以交互方式配置代理程序，或者通过创建静默响应文件并运行配置脚本来以静默方式配置代理程序而不进行交互。
- 在 Windows 系统上，您可以通过创建静默响应文件并运行配置脚本或者使用所提供的“管理监视服务”实用程序来配置代理程序。有关如何在 Windows 系统上启动“管理监视服务”的信息，请参阅第 155 页的『在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口』。

过程

- 要通过编辑静默响应文件并运行无交互的脚本来配置代理程序，请完成以下步骤：

  a) 创建 `.txt` 文件作为静默响应文件。

  b) 在静默响应文件中指定下列参数。语法为 *`parameter_name=parameter_value`*。

  configure_type
  指定配置类型。此参数是必需的。

  对于代理程序配置，有效值为 `tema_configure`。

  KYN_ALT_NODEID
  指定用于标识代理程序的备用节点标识。此参数是必需的。

  有效值是最长 24 个字符的字母数字字符串。

  KYN_PORT
  指定代理程序所使用的侦听端口。这是代理程序用于侦听来自数据收集器的连接请求的 TCP 套接字。此参数是必需的。

  缺省值为 63335。

  切记: 在代理程序共存环境中，确保您在此处指定的端口号未由 ITCAM Agent for WebSphere Applications 使用。

例如，在您创建的 .txt 文件中添加以下行。

```
configure_type=tema_configure
KYN_ALT_NODEID=WASAgent
KYN_PORT=63335
```

c) 保存并关闭该文件，然后输入下列命令以运行配置脚本：

- Linux AIX *install_dir*/bin/was-agent.sh config *path_to_responsefile*
- Windows *install_dir*\bin\was-agent.bat config *path_to_responsefile*

其中，*install_dir* 是代理程序安装目录。缺省值为 C:\IBM\APM（在 Windows 系统上）和 /opt/ibm/apm/agent（在 Linux 系统和 AIX 系统上）。

d) 配置完成后，如果 WebSphere Applications 代理程序未在运行，请使用下列命令将其重新启动：

- Linux AIX *install_dir*/bin/was-agent.sh start
- Windows *install_dir*\bin\was-agent.bat start

· 要通过运行脚本并对提示做出响应来配置代理程序，请完成以下步骤：

a) 从命令行中，转至 *install_dir*/bin 目录，其中 *install_dir* 是代理程序安装目录。

缺省值是 /opt/ibm/apm/agent（在 Linux 系统和 AIX 系统上）。

b) 从目录中运行配置脚本：

```
./was-agent.sh config
```

c) 收到提示时，输入 1 并按 Enter 键以编辑 WebSphere Applications 代理程序的监视代理程序的设置。

d) 按 Enter 键，直到系统提示您输入用于标识监视代理程序的备用节点标识为止。

e) 提供节点标识并按 Enter 键。节点标识的有效格式为最长 24 个字符的字母数字字符串。

f) 系统提示您输入端口号时，请提供代理程序用于侦听来自数据收集器的连接请求的端口，然后按 Enter 键。

切记: 对于代理程序共存环境，确保指定的端口未由 ITCAM Agent for WebSphere Applications 使用。

g) 配置完成后，重新启动 WebSphere Applications 代理程序。

结果

您已配置 WebSphere Applications 代理程序。

下一步做什么

接下来，您必须配置数据收集器。在您配置数据收集器时，系统会要求您提供为 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications 配置的端口号。有关指示信息，请参阅第 732 页的『针对代理程序共存环境配置数据收集器』。

针对代理程序共存环境配置数据收集器

如果您的环境中同时安装了 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications，那么只能为这两个代理程序配置一个数据收集器。

开始之前

确保已完成第 730 页的『（代理程序共存）配置 WebSphere Applications 代理程序和数据收集器』所述的其他步骤。

关于此任务

使用所提供的交互配置实用程序针对 WebSphere Applications 代理程序和 ITCAM Agent for WebSphere Applications 同时存在并共享数据收集器的环境配置数据收集器。

限制: 对于 ITCAM Agent for WebSphere Applications，不支持将数据收集器与下列组件或产品进行集成:

- ITCAM for Application Diagnostics Managing Server
- ITCAM for Transactions
- Tivoli Performance Viewer

切记: ITCAM Agent for WebSphere Applications 不支持监视 WebSphere Application Server Liberty。要监视 WebSphere Application Server Liberty，请仅使用 WebSphere Applications 代理程序。有关针对 Liberty 监视的数据收集器配置，请参阅第 715 页的『以交互方式配置数据收集器』或第 721 页的『以静默方式配置数据收集器』。

过程

1. 以用来安装应用程序服务器的用户标识身份登录系统。
2. 从命令行中，转至 *dc_home* 目录中的 bin 目录。*dc_home* 目录如下所示:
   - Windows *install_dir*\dchome\7.3.0.14.08
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08
3. 运行下列命令以启动配置实用程序:
   - Windows config.bat
   - Linux AIX ./config.sh

   配置实用程序将启动并显示本地计算机系统上找到的所有网卡的 IP 地址。
4. 输入对应于所要使用的 IP 地址的编号，然后按 Enter 键。

   配置实用程序将显示系统上发现的 WebSphere Application Server 主目录。
5. 提示输入应用程序服务器的主目录时，请输入与 WebSphere Application Server 主目录或应用程序服务器主目录的完整路径对应的编号，然后按 Enter 键。

   配置实用程序将显示在指定主目录下发现的所有应用程序服务器概要文件。
6. 提示输入要配置的应用程序服务器概要文件时，请输入与 WebSphere Application Server 概要文件对应的编号，然后按 Enter 键。

   配置实用程序指示是否为您指定的 WebSphere Application Server 概要文件启用 WebSphere 全局安全性。如果未启用全局安全性，请跳至步骤 第 733 页的『8』。
7. 指定是否从客户机属性文件检索安全设置。输入 1 以允许配置实用程序从相应的客户机属性文件中检索用户名和密码。否则，输入 2 以输入 WebSphere 管理员用户名和密码。

   数据收集器使用远程方法调用 (RMI) 或 SOAP 协议与 WebSphere Administrative Services 进行通信。如果为概要文件启用了全局安全性，那么您必须指定有权登录应用程序服务器概要文件的 WebSphere Application Server 管理控制台的用户的用户标识和密码。或者，您可以在配置数据收集器之前对用户名和密码进行加密，并将其存储在客户机属性文件中。必须将 sas.client.props 文件用于 RMI 连接，或将 soap.client.props 文件用于 SOAP 连接。
8. 提示输入 WebSphere 管理控制台的主机名时，请按 Enter 键以接受缺省值，或指定 WebSphere 管理控制台的主机名或 IP 地址。缺省值为 localhost。
9. 当配置实用程序列出所有尚未配置数据收集的服务器实例时，请从列表中选择一个或多个应用程序服务器实例。请输入与要配置进行数据收集的应用程序服务器实例对应的编号，或者输入星号 (*) 以配置所有应用程序服务器实例进行数据收集，然后按 Enter 键。要指定服务器的子集，请输入代表这些服务器的编号，以逗号隔开。
   例如，1,2,3。

   切记:

   - 对于单机环境，必须在配置期间运行应用程序服务器实例。
   - 对于 Network Deployment 环境，Deployment Manager 必须正在运行。
   - 请确保您选择的应用程序服务器实例是托管了您想要监视的应用程序或服务的实际服务器。

配置实用程序为 WebSphere Applications 代理程序提供了用于集成数据收集器的选项。

10. 在 **Integration with Agent for WebSphere Applications** 部分中，指定您希望将数据收集器与监视代理程序进行集成。您必须输入 1 以选择此集成选项，然后按 Enter 键。

    将注册所选服务器以进行 PMI 资源监视。
11. 指定您是否要在应用程序服务器实例中配置数据收集器。您必须输入 1 以表示同意，然后按 Enter 键。
12. 指定是否要允许数据收集器收集诊断数据。输入 1 表示“是”，或者输入 2 表示“否”。
13. 提示输入 V8 监视代理程序组件的主机名时，请输入 WebSphere Applications 代理程序的主机名或 IP 地址，或者接受缺省值。
14. 提示输入 V8 监视代理程序的端口号时，请输入 WebSphere Applications 代理程序所使用的端口号。

    切记: 如果其他组件正在使用缺省值，那么缺省值可能不适合使用。您必须确保环境中的任何其他组件未使用此指定端口。
15. 指定要配置 V6 监视代理程序。输入 1 以配置 ITCAM Agent for WebSphere Applications，然后按 Enter 键。
16. 提示输入 V6 监视代理程序的主机名或 IP 地址时，请指定 ITCAM Agent for WebSphere Applications 的主机名或 IP 地址。
17. 提示输入 V6 监视代理程序的端口号时，请输入 ITCAM Agent for WebSphere Applications 的监视代理程序组件使用的端口号。

    切记: 如果其他组件正在使用缺省值，那么缺省值可能不适合使用。您必须确保环境中的任何其他组件未使用此指定端口。
18. 提示输入服务器别名时，请不要使用缺省值，并指定您要使用的唯一服务器别名。如果要配置多个应用程序服务器实例，那么配置实用程序会提示输入每个实例的别名。

    要点: 别名只能包含以下字符：A-Z、a-z、下划线 (_)、短划线 (-) 以及句点 (.). 请勿在别名中使用其他字符。

    服务器别名是 Cloud APM 控制台中显示的代理程序实例名称的第一个限定符（又被称为 MSN）。缺省值是节点名与服务器名称的组合。例如，**node1server1** 别名表示节点 **node1** 中的服务器 **server1**。
19. 提示输入用于 PMI 资源监视的 TCP/IP 端口号时，请按 Enter 键以接受缺省值，或者输入新的端口号。缺省端口为 63355。

    此端口用于在相同主机上运行的组件之间的内部通信。如果缺省端口在使用中，请设置另一个端口号。
20. 在 **Support for transaction tracking** 部分中，指定是否启用事务跟踪。输入 1 表示“是”，或者输入 2 表示“否”并跳至 22。

    切记: 要查看事务跟踪信息，您需要在 Cloud APM 控制台的“代理程序配置”页面上启用事务跟踪。
21. 接受“事务框架扩展”的缺省主机名或 IP 地址（“事务框架扩展”是 WebSphere Applications 代理程序的内部组件，用于从数据收集器收集度量）。
22. 接受数据收集器用于连接到“事务框架扩展”的缺省端口号。缺省值为 5457。
23. 指定是否将数据收集器与 Application Performance Diagnostics Lite 进行集成。请按 Enter 键以接受缺省值 No。
24. 在 **Advanced settings** 部分中，指定是否要更改垃圾回收日志路径。输入 1 以选择垃圾回收日志路径。否则，输入 2 并跳至步骤 第 734 页的『26』。
25. 指定垃圾回收日志路径。输入带有完整路径的文件名。

    例如，如果指定 gc.log 作为文件名，那么对于每个已配置的应用程序服务器实例，实际名称将设置为 *profile_name*.*cell_name*.*node_name*.*server_name*.gc.log。

    要点: 在垃圾回收日志路径中，您可以使用 WebSphere 变量，例如 ${SERVER_LOG_ROOT}。但是，请勿使用诸如 %pid 之类的模板。
26. 在 **Data collector configuration summary** 部分中，查看要应用于指定的应用程序服务器实例的数据收集器配置的相关摘要。如果需要，请修改配置设置。
27. 输入 a 以接受更改。
28. 出现提示时，指定是否要创建当前配置的备份。输入 1 以创建当前配置的备份。否则，输入 2。

29. 按配置实用程序的指示，重新启动应用程序服务器实例。
    a) 切换到应用程序服务器概要文件主目录下的 bin 目录。例如，opt/IBM/WebSphere/AppServer/profiles/*profile_name*/bin。
    b) 在命令控制台中输入 **stopServer** 命令，以停止应用程序服务器。
    - Linux AIX ./stopServer.sh *server_name*
    - Windows stopServer.bat *server_name*
    c) 出现提示时，请输入 WebSphere 管理控制台管理员的用户标识及密码。
    d) 在命令控制台中输入 **startServer** 命令，以再次启动应用程序服务器。
    - Linux AIX ./startServer.sh *server_name*
    - Windows startServer.bat *server_name*

    数据收集器配置将在应用程序服务器重新启动后生效。
30. 登录 Cloud APM 控制台，以在仪表板中查看数据。
    a) 使用提醒您服务已就绪的电子邮件中提供的链接来访问控制台。另外，也可以从 IBM Marketplace Web 站点访问控制台。有关详细的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。
    b) 使用“应用程序”编辑器，将受监视应用程序服务器添加到 应用程序性能仪表板。您可将其作为新组件添加到现有应用程序，或创建应用程序以包含此组件。

    有关“应用程序”编辑器的更多信息，请参阅第 930 页的『管理应用程序』。

下一步做什么

- 如果用来配置数据收集器的当前用户标识并非运行应用程序服务器的用户标识，请确认对于数据收集器主目录中的 runtime 和 logs 目录，用于配置数据收集器的用户标识具有读写许可权。这两个子目录是在服务器重新启动时，由运行应用程序服务器的用户标识创建。
- 对于 WebSphere Applications 代理程序，登录 Cloud APM 控制台以查看仪表板中的监视数据。对于 ITCAM Agent for WebSphere Applications，登录 Tivoli Enterprise Portal 以查看数据。如果监视数据尚不可用，请通过运行以下命令来重新启动监视代理程序：
  - Windows

```
cd install_dir\bin
was-agent.bat stop
was-agent.bat start
```

  - Linux AIX

```
cd install_dir/bin
./was-agent.sh
stop
./was-agent.sh start
```

- 更改服务器别名将更改向 Cloud APM 控制台注册的代理程序实例名称。如果这不是您第一次配置数据收集器，并且已更改服务器别名，那么必须通过完成下列步骤来清除一些高速缓存文件：
  1. 停止监视代理程序（如果它正在运行）。
  2. 使用文本编辑器打开下列目录中的 *hostname*_yn.xml 文件，其中 *hostname* 是安装了 WebSphere Applications 代理程序或 ITCAM Agent for WebSphere Applications 的主机的名称。
     - Windows *install_dir*\TMAITM6_x64，缺省值为 C:\IBM\APM\TMAITM6_x64（对于 WebSphere Applications 代理程序）或 C:\IBM\ITM\TMAITM6_x64（对于 ITCAM Agent for WebSphere Applications）
     - Linux AIX *install_dir*/config，缺省值为 /opt/ibm/apm/agent/config（对于 WebSphere Applications 代理程序）或 /opt/ibm/itm/agent/config（对于 ITCAM Agent for WebSphere Applications）

3. 找到以下列字符串开头并包含先前服务器名称的行。例如，`was85.win4net01Cell02.win4net01Node02.AppSrv01.server1`，其中 `server1` 是先前的应用程序服务器名称。

   ```
   <!ENTITY was_product_code.cellname.nodename.profilename.servername
   ```

   其中，*was_product_code* 是 WebSphere Application Server 的产品代码；*cellname* 是单元格的名称；*nodename* 是节点名；*profilename* 是应用程序服务器概要文件的名称；*servername* 是先前的应用程序服务器名称。
4. 找到该行中指示的位于当前目录中的 .XML 文件，并删除该文件。
5. 从 *hostname*`_yn.xml` 文件中移除您在步骤 3 中找到的行。
6. 在 *hostname*`_yn.xml` 文件的结尾，移除包含先前服务器名称的行。
7. 保存并关闭该文件。
8. 重新启动监视代理程序。

在 Cloud APM 服务器上更改产品类型的情况下重新配置数据收集器

如果已将 Cloud APM 服务器上安装的产品类型从 Cloud APM, Base 更改为 Cloud APM, Advanced，而且 WebSphere Applications 代理程序是随 Cloud APM, Base 产品一起进行安装和配置，以使用 Cloud APM, Advanced 产品中提供的代理程序高级功能，那么必须卸载先前的 WebSphere Applications 代理程序，并随 Cloud APM, Advanced 产品一起重新安装该代理程序。另外，也可以重新配置数据收集器，以使这些功能在新产品中可用。

关于此任务

根据使用哪个代理程序包安装代理程序，WebSphere Applications 代理程序的配置方式会不同。在 Cloud APM 服务器上更改产品类型后，您有两个选项来使代理程序功能在新产品中可用：

· 移除随先前产品安装的代理程序，然后安装新产品中的代理程序。

· 再次重新配置数据收集器以使用新产品中的功能。

过程

· 移除随先前产品安装的代理程序，然后安装新产品中的代理程序。

  a) 取消配置数据收集器。有关指示信息，请参阅第 126 页的『WebSphere Applications 代理程序：取消配置数据收集器』。

  b) 卸载使用先前产品的代理程序包安装的 WebSphere Applications 代理程序。有关指示信息，请参阅第 124 页的『卸载代理程序』。

  c) 使用新产品中的代理程序包安装 WebSphere Applications 代理程序并再次配置数据收集器。有关指示信息，请参阅第 712 页的『使用简单配置实用程序来配置数据收集器』。

· 再次重新配置数据收集器以使用新产品中的功能。

  a) 根据新产品类型，通过将 **IOFFERING** 值更改为以下其中一个值，编辑数据收集器主目录中的 `offering.id` 文件：

  BASE
  如果新产品类型为 Cloud APM, Base Private。

  ADVANCED
  如果新产品类型为 Cloud APM, Advanced Private。

  数据收集器主目录如下：

  – Windows *install_dir*`\dchome\7.3.0.14.08`

  – Linux AIX *install_dir*`/yndchome/7.3.0.14.08`

  b) 重新配置数据收集器，以根据新产品类型支持的功能在数据收集器中启用诊断和/或事务跟踪。有关如何配置数据收集器的指示信息，请参阅第 712 页的『使用简单配置实用程序来配置数据收集器』。

  c) 重新启动 WebSphere Application Server。

d) 针对 Cloud APM 控制台的任何页面，单击 **系统配置** > **代理程序配置**以打开“**代理程序配置**”页面。确保事务跟踪设置与新产品类型中可用的功能匹配。如果不匹配，请更新设置。

应该对 Cloud APM, Advanced 启用事务跟踪设置，但对 Cloud APM, Base 禁用。

监视 Docker 容器内部的 WebSphere Application Server Liberty

要监视 Docker 容器内部的 Liberty 概要文件，必须使用带有一些选项的 **docker run** 命令来配置数据收集器，然后才能启动 WebSphere Application Server Liberty。

开始之前

必须在 Docker 主机上安装 WebSphere Applications 代理程序。

关于此任务

每个在 Docker 中运行的 Liberty 概要文件都需要一个数据收集器以收集资源度量、事务度量和诊断数据，然后将数据传送到在 Docker 主机上运行的监视代理程序。在相同 Docker 主机上配置的所有数据收集器共享主机上相同的监视代理程序。

过程

要配置 Liberty 概要文件容器的数据收集器，请完成以下步骤：

1. 创建 `.txt` 静默响应文件，在文件中指定以下配置选项并保存。

   ```
   tema.host=agent_host
   was.appserver.server.name=liberty_profile_name
   ```

   其中，**tema.host** 用于指定监视代理程序主机的 IP 地址；**was.appserver.server.name** 用于指定 Liberty 概要文件的名称。

   提示: *<agent_install_dir>*/agent/yndchome/7.3.0.14.08/bin 目录中提供样本静默响应文件 (sample_silent_liberty_config.txt)。您可以基于此样本文件创建自己的响应文件。

2. 运行以下命令以启动新的 Docker 容器。请注意，必须通过将 **LICENSE** 参数设置为 accept 接受许可证才能完成配置。

   ```
   $docker run -d -e LICENSE=accept \
   -e JAVA_HOME=<java_home_dir> \
   -p <port_number>:<port_number> \
   -v <web_app_dir>:<liberty_install_dir>/usr/servers/<liberty_profile_name> \
   -v <agent_install_dir>/agent/yndchome:<agent_install_dir>/agent/yndchome
   websphere-liberty /bin/bash \
   -c "<agent_install_dir>/agent/yndchome/<dcversion>/bin/config.sh -silent
   <absolute_path_to_silent_response_file> && <liberty_install_dir>/bin/server
   run <liberty_profile_name>"
   ```

   其中：

   - *<java_home_dir>* 是 Liberty 概要文件使用的 JRE 的目录。例如，/opt/ibm/java/jre。
   - *<port_number>* 是容器和主机之间通信所用的端口号。
   - *<web_app_dir>* 是 Web 应用程序所在的目录。
   - *<liberty_install_dir>* 是 WebSphere Application Server Liberty 的安装目录。缺省值为 /opt/ibm/wlp。
   - *<liberty_profile_name>* 是 Liberty 概要文件的名称。
   - *<agent_install_dir>* 是 WebSphere Applications 代理程序的安装目录。缺省值为 /opt/ibm/apm。
   - *<dcversion>* 是 WebSphere Applications 代理程序的数据收集器的版本号。例如，7.3.0.14.08。
   - *<absolute_path_to_silent_response_file>* 是您创建的静默响应文件的绝对路径。

例如，以下命令配置名为 newitcam 的 Liberty 概要文件的数据收集器。WebSphere Applications 代理程序和 Liberty 概要文件都安装在缺省目录中。监视代理程序和数据收集器的版本是 7.3.0.14.08。

```
$docker run -d -e LICENSE=accept \
-e JAVA_HOME=/opt/ibm/java/jre \
-p 9082:9082 \
-v /home/kub/liberty-docker/newitcam:/opt/ibm/wlp/usr/servers/newitcam \
-v /opt/ibm/apm/agent/yndchome:/opt/ibm/apm/agent/yndchome websphere-liberty
/bin/bash \
-c "/opt/ibm/apm/agent/yndchome/7.3.0.14.08/bin/config.sh -silent
/opt/ibm/wlp/usr/servers/newitcam/silent_config.txt && /opt/ibm/wlp/bin/server
run newitcam"
```

结果

现在，您可以验证 WebSphere Applications 代理程序数据显示在 Cloud APM 控制台中。**WAS 信息**窗口小部件的**单元名称**列显示 Liberty 概要文件运行所在的 Docker 容器的 标识。

下一步做什么

要以交互方式取消配置数据收集器，请使用以下命令来启动取消配置实用程序：

```
docker exec -i container_id "<agent_install_dir>/yndchome/7.3.0.14.08/bin
/unconfig.sh"
```

手动配置数据收集器以监视动态集群服务器

您可以通过向用于创建动态集群服务器实例的服务器模板添加一些数据收集器配置参数来配置数据收集器以监视动态集群中的应用程序服务器实例。配置动态集群服务器实例的另一种方法是创建特定于 WebSphere Applications 代理程序的服务器模板。

关于此任务

要配置数据收集器以用于动态集群监视，必须创建两个设置文件，然后手动在 WebSphere 管理控制台中添加设置以修改动态服务器模板。在数据收集器启动时，将针对应用程序服务器实例自动创建 `runtime` 目录。请注意，对于服务器模板的任何升级都将擦除以此方式进行的这些更改。

要点:

- 集群名称不得包含空格。
- 必须以 WebSphere 管理用户身份手动更改 WebSphere Application Server 的数据收集器配置。
- 您必须是经验丰富的 WebSphere 管理员才可以针对数据数据对 WebSphere Application Server 进行手动更改。手动配置更改中的错误可能导致应用程序服务器无法启动。
- 如果手动配置数据收集器以监视应用程序服务器实例，那么无法使用取消配置实用程序来取消配置数据收集器。要取消对数据收集器的配置，必须手动更改回设置。

过程

1. 在数据收集器 `runtime` 目录中创建 `dcManualInput.txt` 文件。

   遵循第 739 页的『创建 dcManualInput.txt 文件』 中的指示信息。
2. 在数据收集器 `wsBundleMetaData` 目录中创建 `itcam_wsBundleMetaData.xml` 文件。遵循第 741 页的『创建 itcam_wsBundleMetaData.xml 文件』 中的指示信息。
3. 通过添加数据收集器配置参数，使用 WebSphere 管理控制台来修改动态服务器模板。遵循第 741 页的『使用 WebSphere 管理控制台添加设置』 中的指示信息。

   提示: 动态集群成员名用作 Cloud APM 控制台中显示的 WebSphere Applications 代理程序实例名称的中间限定符。有时，由于代理程序实例名称存在长度限制，因此集群成员名可能会被截断。在这种情况下，您可以通过添加变量 *${MEP_NAME}* 并将值设置为每个服务器实例的 JVM 名称来修改动态服务器模板。然后，您可以在 Cloud APM 控制台上按实际 JVM 名称来区分每个集群成员。有关指示信息，请参阅第 744 页的『可选：显示实际 JVM 名称以区分集群成员』。

创建 *dcManualInput.txt* 文件

关于此任务

dcManualInput.txt 文件包含数据收集器的初始配置所需的某些值。

过程

要创建 dcManualInput.txt 文件，请完成下列步骤：

1. 检查以下目录中是否存在名为 *platform*_Template.DCManualInput.txt 的文件。如果不存在，请创建。
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime
   - Windows *install_dir*\dchome\7.3.0.14.08\runtime

   文件名中的 *platform* 变量指示操作系统体系结构，例如，aix32、xLinux64。

   您可以任意命名文件。但是，在运行 configtemplate.sh 脚本创建文件时，*platform*_Template.DCManualInput.txt 遵循标准命名约定。在后续步骤中，您需要使用 WebSphere 管理控制台来为服务器模板指定此文件。
2. 将以下文件的内容复制到在先前步骤中找到或者创建的 .txt 文件。
   - Linux AIX *dc_home*/itcamdc/etc/was/dcInput_manual.properties
   - Windows *dc_home*\itcamdc\etc\was\dcInput_manual.properties
3. 编辑 .txt 文件的内容。必须根据第 739 页的表 226 中提供的描述，在文件的第 1 部分中设置参数。

   切记：

   - 请勿更改第 2 部分中的参数。
   - 数据收集器用于创建运行时目录的某些配置参数总是设置为 none。这是因为在动态集群监视中，数据收集器使用 WebSphere 服务器实例配置在 JVM 启动时创建目录。

表 *226.* 第 *1* 部分的配置参数

参数	值
local.hostname	本地系统的 IP 地址或标准域名。
was.profile.home	概要文件主目录。 对于动态集群服务器，总是设置为 none。
was.version	短版本号。 对于动态集群服务器，总是设置为 none。
itcam.home	数据收集器主目录。 示例：/opt/ibm/apm/agent/yndchome/7.3.0.14.08
was.nodename	节点名。 对于动态集群服务器，总是设置为 none。
was.servername	服务器名称。 对于动态集群服务器，总是设置为 none。
was.profilename	WebSphere 概要文件名称。 对于动态集群服务器，总是设置为 none。

表 226. 第 1 部分的配置参数 (续)

参数	值
am.camtoolkit.gpe.dc.operation.mode	数据收集器的操作方式。有效值是 WR、TT 和 DE 的任意组合，其中： **WR** 将数据收集器与 WebSphere Applications 代理程序相集成。 **TT** 向数据收集器与 ITCAM for Transactions 相集成。 **DE** 将数据收集器与 ITCAM Diagnostics Tool 相集成。ITCAM for Application Diagnostics 试用中提供此工具的预览。 必须仅指定需要的操作方式。例如，如果仅将数据收集器连接到 WebSphere Applications 代理程序，那么指定 WR。 使用逗号分隔多种操作方式。 示例： `am.camtoolkit.gpe.dc.operation.mode=WR, DE`
interp	平台代码。 示例：`interp=win64` 或 `interp=lx6266`
kwj.serveralias	WebSphere Application Server 别名。 对于动态集群服务器，总是设置为 none。
temagclog.path	（可选）垃圾回收日志文件路径名。输入带有完整路径的唯一文件名。路径名不得包含空格。
tema.host	WebSphere Applications 代理程序的主机名或 IP 地址。如果操作方式包含 WebSphere Applications 代理程序 (WR)，那么是必需的。通常，监视代理程序安装在数据收集器运行所在的系统上，并且可指定回送地址。 示例：`tema.host=127.0.0.1`
tema.port	用于与 WebSphere Applications 代理程序通信的端口。如果操作方式包含 WebSphere Applications 代理程序 (WR)，那么是必需的。缺省值为 63335。 示例：`tema.port=63335`
tt.connection.string	ITCAM for Transactions 事务收集器组件的主机名或 IP 地址以及端口号，格式为 `tcp:`*`host_name`*`(IP):`*`port`*。如果操作方式包含 ITCAM for Transactions (TT)，那么是必需的。 示例： `tt.connection.string=192.38.234.77:5455`

4. 向 `.txt` 文件的第 1 部分添加以下行。

```
bcm.helper=com.ibm.tivoli.itcam.was.bcm.websphere.DefaultWASBCMHelper
BCM_HELPER=@{bcm.helper}
```

5. 保存并关闭该文件。

创建 *itcam_wsBundleMetaData.xml* 文件

关于此任务

itcam_wsBundleMetaData.xml 文件包含数据收集器初始配置必需的一些值。

过程

要创建此文件，请完成以下步骤：

1. 在以下目录中创建名为 wsBundleMetaData 的目录：
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/runtime
   - Windows *install_dir*\dchome\7.3.0.14.08\runtime
2. 创建名为 itcam_wsBundleMetaData.xml 的文件并将为其复制以下文件的内容：
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc/etc/was/itcam_wsBundleMetaData_template.xml
   - Windows *install_dir*\dchome\7.3.0.14.08\itcamdc\etc\was\itcam_wsBundleMetaData_template.xml
3. 在 itcam_wsBundleMetaData.xml 文件中，将 *@{CONFIGHOME}* 变量替换为数据收集器主目录的完整路径。

   每个操作系统上的数据收集器主目录如下所示：
   - Linux AIX *install_dir*/yndchome/7.3.0.14.08
   - Windows *install_dir*\dchome\7.3.0.14.08
4. 将 itcam_wsBundleMetaData.xml 文件放置在步骤 1 中创建的 wsBundleMetaData 目录中。

使用 *WebSphere* 管理控制台添加设置

过程

完成以下步骤以使用 WebSphere 管理控制台修改动态服务器模板。

1. 登录 WebSphere 管理控制台。
2. 单击**服务器**。
3. 展开**集群**，然后选择**动态集群**。
4. 单击想要使用数据收集器配置的动态服务器集群的名称。
5. 在**其他属性**部分下，单击**服务器模板**。
6. 在**服务器基础结构**部分下，展开 **Java 和进程管理**，然后单击**进程定义**。
7. 在**其他属性**部分中，单击 **Java 虚拟机**。
8. 在**通用 JVM 参数**字段中，添加以下项。

```
-agentlib:am_$jvm-vendor_$jvm-version=${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
$platform_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
–Ditcamdc.dyncluster=true
```

在添加条目时，请注意以下事项：

- 所有条目都必须位于单独一行上。
- 在 - 符号前用空格分隔不同参数，其他位置请勿使用空格。
- 将以下变量替换为实际名称：

- *$jvm-vendor*：使用的 JVM 供应商。
- *$jvm-version*：JVM 版本信息，例如基于 Java 5 的 15、基于 Java 6 的 16 或基于 7 的 17。
- *$platform_Template_DCManualInput.txt*：在先前步骤中创建的 `.txt` 文件。

示例：

```
-agentlib:am_ibm_16=${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
aix64_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
–Ditcamdc.dyncluster=true
```

9. 单击**应用**。
10. 在“消息”对话框中，单击**保存**。
11. 在“保存到主配置”对话框中，完成以下步骤：
    - 如果您处于网络部署环境中，请确保选中**与节点同步更改**，然后单击**保存**。
    - 如果您不是处于网络部署环境中，那么单击**保存**。
12. 返回以展开**集群**，单击**动态集群**，然后单击相同的服务器名称。
13. 在**配置**选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **环境条目**。
14. 根据操作系统、硬件平台和应用程序服务器 JVM，设置以下环境条目：

表 *227.* 环境条目

平台	环境条目名称	环境条目值
AIX R6.1（32 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix533:${ITCAMDCHOME}/toolkit/lib/aix533/ttapi
AIX R6.1（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536:${ITCAMDCHOME}/toolkit/lib/aix536/ttapi
AIX R7.1（32 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix533:${ITCAMDCHOME}/toolkit/lib/aix533/ttapi
AIX R7.1（64 位 JVM）	LIBPATH	/lib:${ITCAMDCHOME}/toolkit/lib/aix536:${ITCAMDCHOME}/toolkit/lib/aix536/ttapi
Linux x86_64 R2.6（64 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx8266:${ITCAMDCHOME}/toolkit/lib/lx8266/ttapi
Linux Intel R2.6（32 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lx6263:${ITCAMDCHOME}/toolkit/lib/lx6263/ttapi

表 227. 环境条目 (续)		
平台	**环境条目名称**	**环境条目值**
Linux ppc R2.6（32 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpp263:${ITCAMDCHOME}/toolkit/lib/lpp263/ttapi
Linux ppc R2.6（64 位 JVM）	LD_LIBRARY_PATH	/lib:${ITCAMDCHOME}/toolkit/lib/lpp266:${ITCAMDCHOME}/toolkit/lib/lpp266/ttapi
Windows（32 位 JVM）	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win32;${ITCAMDCHOME}/toolkit/lib/win32/ttapi
Windows（64 位 JVM）	PATH	/lib;${ITCAMDCHOME}/toolkit/lib/win64;${ITCAMDCHOME}/toolkit/lib/win64/ttapi

15. 将 NLSPATH 环境条目名称设置为以下值：

```
${ITCAMDCHOME}/toolkit/msg/%L/%N.cat
```

16. 单击**应用**，然后单击**保存**。
17. 在“保存到主配置”对话框中，完成下列步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**，然后单击**保存**。
    - 如果您不是处于 Network Deployment 环境中，那么单击**保存**。
18. 返回以展开**集群**，单击**动态集群**，然后单击相同的服务器名称。
19. 在**配置**选项卡中，转至**服务器基础结构** > **Java 和进程管理** > **进程定义** > **Java 虚拟机** > **其他属性：定制属性**。
20. 单击**新建**以添加以下名称和值对，然后单击**应用**。
    - 创建 am.home 属性并将其值设置为以下目录：
      - Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc
      - Windows *install_dir*\dchome\7.3.0.14.08\itcamdc
    - 创建 am.orig.wascell 属性并将其值设置为单元目录。例如，am.orig.wascell =cellname1。
    - 创建 com.ibm.tivoli.itcam.toolkit.ai.runtimebuilder.enable.rebuild 属并将其值设置为 true。
    - 创建 ITCAM_DC_ENABLED 属性并将其值设置为 true。
    - 创建 TEMAGCCollector.gclog.path 属性。如果设置了通用 JVM verlogsegclog 参数，那么将 TEMAGCCollector.gclog.path 属性的值设置为相同值。否则，将 TEMAGCCollector.gclog.path 属性设置为 None。

      提示: 要标识 verlogsegclog 属性的值，请在**配置**选项卡中，单击**服务器基础结构** > **Java 和进程管理** > **进程定义** > **Java 虚拟机**。verlogsegclog 值位于**通用 JVM 参数**字段中。
21. 在“消息”对话框中，单击**保存**。
22. 在“保存到主配置”对话框中，完成下列步骤：
    - 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**。单击**保存**。

- 如果您不是处于 Network Deployment 环境中，那么单击**保存**。

23. 在导航窗格中，单击**环境** > **WebSphere 变量**。
24. 设置以下变量。对于每个变量，必须根据不同系统上的数据收集器安装目录选择相应的作用域级别。如果不同系统对于数据收集器有多个的安装目录，那么必须针对每个节点级别作用域正确设置这些变量。如果他们全都具有相同的安装目录，那么作用域可以更高，例如，在集群级别。

    · 将 *ITCAMDCHOME* 设置为以下目录：

    – Linux AIX *install_dir*/yndchome/7.3.0.14.08/itcamdc

    – Windows *install_dir*\dchome\7.3.0.14.08\itcamdc

    · 将 *ITCAMDCVERSION* 设置为数据收集器的版本，例如，7.3.0.14.08。
25. 单击**应用**，然后单击**保存**。
26. 在“保存到主配置”对话框中，完成下列步骤：

    · 如果您处于 Network Deployment 环境中，请确保选中**与节点同步更改**，然后单击**保存**。

    · 如果您不是处于 Network Deployment 环境中，那么单击**保存**。

    在修改模板后，值将与动态集群中的所有服务器实例进行同步。动态创建的任何新服务器也将具有相同的数据收集器配置参数。
27. 针对要激活的数据收集器重新启动应用程序服务器实例。数据收集器将读取设置文件并创建运行时目录。

可选：显示实际 *JVM* 名称以区分集群成员

关于此任务

在 Cloud APM 控制台上，WebSphere Applications 代理程序实例名称的格式为 *host_name*::*was_server_name*:KYNS 并且最大长度为 32 个字符。在动态集群环境中，动态集群成员名称将用于中间限定符 *was_server_name*。

有时，由于存在字符长度限制，因此集群成员名可能会被截断。在这种情况下，您可以在代理程序实例名称中指定要用于中间限定符的实际 JVM 名称。

过程

执行下列步骤以便在代理程序实例名称中显示实际 JVM 名称：

1. 登录 WebSphere 管理控制台，通过添加新环境变量 *${MEP_NAME}* 来更新通用 JVM 参数，如下所示：

```
-agentlib:am_$jvm-vendor_$jvm-version=${MEP_NAME}${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc -Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
$platform_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
-Ditcamdc.dyncluster=true
```

   示例：

```
-agentlib:am_ibm_16=${MEP_NAME}${WAS_SERVER_NAME}
-Xbootclasspath/p:${ITCAMDCHOME}/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=${ITCAMDCHOME}/itcamdc/etc/datacollector.policy
-verbosegc
-Dcom.ibm.tivoli.itcam.ai.runtimebuilder.inputs=${ITCAMDCHOME}/runtime/
aix64_Template_DCManualInput.txt
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.rmi.transport.connectionTimeout=300000
-Dws.bundle.metadata=${ITCAMDCHOME}/runtime/wsBundleMetaData
-Ditcamdc.dyncluster=true
```

2. 保存并应用更改。

3. 在导航窗格中，单击**环境** > **WebSphere 变量**为每个动态集群成员定义 *${MEP_NAME}* 变量。将值设置为集群成员的实际 JVM 名称。
4. 保存并应用更改。
5. 重新启动应用程序服务器实例。

   在 Cloud APM 控制台上，将显示其名称包含您刚才指定的 *${MEP_NAME}* 值的新 WebSphere Applications 代理程序实例。

在代理程序配置页面上动态配置数据收集

在数据收集器中启用对事务跟踪或诊断数据收集的支持后，使用“**代理程序配置**”页面动态启用或禁用数据收集。

开始之前

- 必须安装和配置 Monitoring Agent for WebSphere Applications。
- 要启用或禁用受监视的应用程序服务器的事务跟踪，必须安装“事务跟踪”。还必须在代理程序中启用对事务跟踪的支持，如第 715 页的『以交互方式配置数据收集器』中所述。如果遵循简单配置过程，那么将自动使用事务跟踪支持配置数据收集器。
- 要启用或禁用包括方法跟踪在内的诊断数据收集，您必须具有 Cloud APM, Advanced。还必须在数据收集器中启用对诊断和方法跟踪信息收集的支持，如第 715 页的『以交互方式配置数据收集器』中所述。（不适用于 Cloud APM, Base）。

提示: “**代理程序配置**”页面显示代理程序监视的所有服务器。如果缺少任何服务器，可能未对其进行正确监视。检查受监视系统上的代理程序日志文件中是否有错误消息，例如，连接错误。

切记: WebSphere Applications 代理程序仅支持 Db2 和 Oracle 作为数据源。对于其他类型的数据源，某些 KPI 值在“事务跟踪”仪表板和组窗口小部件上可能显示为空值。

过程

完成以下步骤来为每个服务器配置数据收集：

1. 从导航栏，单击 **系统配置** > **代理程序配置**。

   “**代理程序配置**”页面随即显示。
2. 单击 **WebSphere 应用程序**选项卡。
3. 选择要配置数据收集的服务器的复选框，然后从 **Actions** 列表执行以下一个操作：
   - 要启用事务跟踪，请单击 **Enable Transaction Tracking**。针对每个所选服务器，**Current Transaction Tracking** 列中的状态更新为 Yes。
   - 要仅启用诊断数据收集，请单击 **Enable Diagnostic Mode**。针对每个所选服务器，**Current Diagnostic Mode** 列中的状态将更新为 Yes。
   - 要同时收集诊断数据和方法跟踪信息，请单击 **Enable Diagnostic Mode and Method Trace**。对于每个选中的服务器，**Current Diagnostic Mode** 和 **Current Method Trace** 列中的状态将更新为 Yes。
   - 要禁用所选服务器的事务跟踪，请单击 **Disable Transaction Tracking**。针对每个所选服务器，**Current Transaction Tracking** 列中的状态更新为 No。
   - 如果仅针对所选服务器启用了诊断数据收集，那么要禁用数据收集，请单击 **Disable Diagnostic Mode**。针对每个所选服务器，**Current Diagnostic Mode** 列中的状态将更新为 No。
   - 如果针对所选服务器同时启用了诊断数据和方法跟踪数据，那么要禁用数据收集，请单击 **Disable Diagnostic Mode and Method Trace**。对于每个选中的服务器，**Current Diagnostic Mode** 和 **Current Method Trace** 列中的状态将更新为 No。

   切记:

   - 除非在数据收集器中配置了对事务跟踪或诊断数据收集的支持，否则“**代理程序配置**”页面上的操作不会启用数据收集，并且列值设置为 No。
   - 如果应用程序服务器概要文件已配置为使用 `127.0.0.1` 作为主机名，并且“**代理程序配置**”页面上的 **IP 地址**列将显示其中安装并运行了 WebSphere Applications 代理程序的主机的 IP 地址。

结果
您已为每个选中的服务器配置了数据收集。在启用数据收集之后，事务跟踪数据和诊断数据会显示在拓扑仪表板中。

要点: 如果应用程序服务器重新启动，那么可能需要再次为服务器启用事务跟踪或诊断数据收集。

启用内存泄漏监视
要让“内存分析”仪表板包含数据，必须对数据收集器启用内存泄漏监视。如果应用程序服务器所使用的 JRE 受支持，那么在缺省情况下，您启用诊断数据收集后，将启用内存泄漏监视功能。

开始之前

- 确保在应用程序服务器的 JVM 自变量中未定义 `-Xtrace:none`。
- 启用内存泄漏监视后，在应用程序服务器的 JVM 自变量中定义以下设置。如果已在当前 JVM 自变量中定义这些设置，请确保数据收集器配置能够对其进行更改。

```
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
```

- 确保应用程序服务器所使用的 JRE 是下列其中一个版本:
  - IBM JRE 1.6.0 SR16 FP3 或更高版本
  - IBM JRE 1.6.1 SR8 FP3 或更高版本
  - IBM JRE 1.7.0 SR8 FP10 或更高版本
  - IBM JRE 1.7.1 SR2 FP10 或更高版本
  - IBM JRE 1.8 或更高版本
  - 版本高于 1.6.0 SR7 且具有针对 APAR IV67574 的 iFix 的其他 IBM JRE

关于此任务

内存泄漏监视功能需要 IBM JRE 的 IBM Health Center 组件。您必须确保应用程序服务器所使用的 JRE 受此功能支持。

- 在 AIX 或 Linux 系统上，当您配置数据收集器以启用诊断数据收集时，如果当前 JRE 受支持，那么配置实用程序将自动检查 Health Center 组件是否合格，并在其不合格时予以升级。
- 在 Windows 系统上，如果 Health Center 组件的当前版本不受支持，那么您必须对其进行手动收集，这是因为配置实用程序无法替换运行中的 JRE 的文件。

切记: 只有在 Windows 系统上，才需要完成以下过程。对于 AIX 或 Linux 系统，要启用内存泄漏监视，您只需确保 JRE 版本受支持，并启用诊断数据收集。对于 Solaris 系统，IBM JRE 的 Health Center 不受支持，因此无法在 Solaris 系统上启用内存泄漏监视。

过程

1. 检查应用程序服务器所使用的 JRE 中包括的 IBM Health Center 版本。
   a) 在命令提示符处，切换到 JRE 主目录中的 `bin` 目录。
   b) 输入 `java -Xhealthcenter -version`，并按 Enter 键。
   此命令将返回 JRE 版本以及 IBM Health Center 版本。内存泄漏监视功能需要 IBM Health Center 3.0.11 或更高版本。
2. 如果 IBM Health Center 版本不合格，请将 JRE 升级到包含 IBM Health Center 3.0.11 或更高版本的版本。
3. 运行数据收集器配置实用程序或重新配置实用程序，以启用诊断数据收集。
   - 如果尚未配置数据收集器，请使用 `simpleconfig` 或 `config`。
   - 如果已配置数据收集器，请使用 `reconfig` 实用程序。

   切记: 即使升级 JRE 前已启用诊断数据收集，也仍然需要再次运行数据收集配置实用程序。

配置 PMI
要在操作监视仪表板中查看性能数据，必须将 WebSphere Application Server 上的性能监视基础结构 (PMI) 配置为收集性能数据。

关于此任务

使用 WebSphere Administrative Console 启用 PMI 并设置 WebSphere Application Server 上的 PMI 级别。

PMI 提供了以下四个预定义级别：

1. 无
2. 基本
3. 扩展
4. 全部

您可以使用定制选项选择性地启用或禁用个别统计信息。每个级别包括该级别以下的数据。

要在操作监视仪表板中显示数据，仪表板计算中所用的属性必须包含在所选级别中。

缺省情况下，WebSphere Applications 代理程序会将 PMI 级别设置得足以收集所需属性。

限制: 要查看某些“进程服务器”和“事务管理器”组窗口小部件中的数据，您必须手动设置 PMI 级别。有关更多信息，请参阅有关组窗口小部件的悬浮式帮助。

如果使用 WebSphere Administrative Console 修改 PMI 级别，那么您必须验证此级别是否足以收集所需数据。

过程

- 要在应用程序服务器上启用 PMI，请完成以下步骤：
  a) 在 WebSphere 管理控制台中，展开**监视和调整**，然后选择**性能监视基础结构 (PMI)**。
  b) 从服务器列表中，单击您服务器的名称。
  c) 单击“配置”选项卡，然后选中**启用性能监视基础结构 (PMI)** 复选框。
  d) 单击**应用**或**确定**。
  e) 单击**保存**以启用 PMI。
- 要在应用程序服务器上设置 PMI 级别，请完成以下步骤：
  a) 在 WebSphere 管理控制台中，展开**监视和调整**，然后选择**性能监视基础结构 (PMI)**。
  b) 从服务器列表中，单击您服务器的名称。
  c) 单击“配置”选项卡，然后选择要使用的统计信息集：“基本”、“扩展”、“全部”或“定制”。
  d) 单击**应用**或**确定**。
  e) 单击**保存**以设置 PMI 级别。

  有关每个属性所需的 PMI 级别的信息，请参阅 WebSphere Applications 代理程序参考中的“仪表板属性”部分。开启每个属性的收集之后，将显示所要承担的监视开销。

从备份复原应用程序服务器配置
如果已手动或者通过配置或迁移实用程序为数据收集配置独立应用程序服务器实例，且应用程序服务器启动失败，那么必须从备份复原应用程序服务器配置。如果未创建备份，请联系 IBM 支持人员。

关于此任务

在 Network Deployment 环境中，如果已手动或者通过配置或迁移实用程序配置应用程序服务器实例，且应用程序服务器启动失败，可选择执行以下操作：

- 可以从备份配置复原应用程序服务器配置。如果未创建备份，请联系 IBM 支持人员。
- 可以手动取消配置数据收集器。应用程序服务器上的 Deployment Manager 和 Node Agent 必须正在运行。有关更多信息，请参阅第 129 页的『从应用程序服务器实例手动移除数据收集器配置』。

此部分仅适用于 Windows、UNIX 和 Linux 操作系统。

过程

要使用 **restoreConfig** 命令应用备份配置，请使用以下其中一个过程：

- 在非 Network Deployment 环境中，完成以下步骤：
  a) 找到备份配置文件。

  缺省目录为 *dc_home*/data。如果存在多个备份文件，请检查文件的修改日期和时间。它必须为发生失败的配置的日期和时间。如果在一个配置失败后，未针对相同主机完成任何其他数据收集器配置，请使用此目录中的最新文件。

  b) 停止应用程序服务器的所有实例。

  c) 从 *appserver_home*/profiles/*profile_name*/bin 目录运行 **restoreConfig** 命令。

  命令语法如下所示：

  – Windows restoreConfig.bat *full_path_to_backup_file*

  – Linux AIX ./restoreConfig.sh *full_path_to_backup_file*

  有关 **restoreConfig** 命令的参数的更多信息，请参阅 WebSphere Application Server Knowledge Center 中的主题。

  d) 再次启动应用程序服务器的实例。
- 在 Network Deployment 环境中，完成以下步骤：
  a) 找到备份配置文件。

  缺省目录为 *dc_home*/data。如果存在多个备份文件，请检查文件的修改日期和时间；它必须为发生失败的配置的日期和时间。如果在一个配置失败后，未针对相同主机完成任何其他数据收集器配置，请使用此目录中的最新文件。

  b) 停止应用程序服务器的所有实例。

  c) 在任何方便的目录中创建临时目录 (*temp_directory*)。在 UNIX 或 Linux 系统上，在 /tmp 目录下创建此临时目录。

  d) 从 *appserver_home*/profiles/*profile_name*/bin 目录运行 restoreConfig 命令。

  命令语法如下所示：

  – Windows restoreConfig.bat *full_path_to_backup_file*

  – Linux AIX ./restoreConfig.sh *full_path_to_backup_file*

  restoreConfig 命令用于将原始应用程序服务器配置复原为临时目录。

  e) 将 server.xml、variables.xml 和 pmi-config.xml 文件从临时目录复制到 Deployment Manager 系统。

  – 源目录：*temp_directory*/*restored_configuration_home*/cells/*cell_name*/nodes/*node_name*/servers/*server_name*

  – 目标目录：*appserver_home*/profiles/*profile_name*/config/cells/*cell_name*/nodes/*node_name*/servers/*server_name*

  f) 从节点的 Deployment Manager 管理控制台完成节点同步。

  g) 在 Deployment Manager 管理控制台中，保存对主配置进行的更改。

  h) 启动应用程序服务器的实例。

为本地应用程序配置 Liberty 数据收集器

要在 Linux for System x 上监视 Liberty 概要文件，您可以将独立的数据收集器直接部署到本地 Liberty 目录，而不安装 WebSphere Applications 代理程序。

开始之前

1. 从 IBM Passport Advantage® Web 站点下载 `IBM_Data_Collectors_Install.tgz` 数据收集器软件包。有关详细的指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。
2. 如果防火墙规则不允许与外部主机建立透明出站 HTTPS 连接，您可以配置数据收集器，以将流量发送到转发代理。有关指示信息，请参阅第 138 页的『配置数据收集器以通过转发代理进行通信』。
3. 数据收集器需要 monitor-1.0 功能部件。您可以使用 **installUtility** 命令从 Liberty 功能存储库中下载此功能部件。有关指示信息，请参阅 WebSphere Application Server Network Deployment Knowledge Center 中有关下载资产的部分。
4. 要让“内存分析”仪表板包含数据，必须在配置期间对数据收集器启用内存分配收集。此诊断功能需要 IBM Health Center 3.0.8 或更高版本。如果 IBM Health Center 版本不合格，请将应用程序服务器使用的 JRE 升级到包含 IBM Health Center 3.0.8 或更高版本的版本。

   提示: 要检查应用程序服务器使用的 JRE 中包含的 IBM Health Center 版本，请切换至 JRE 主目录中的 `bin` 目录，然后发出 `java -Xhealthcenter -version`。

关于此任务

您可以选择手动配置数据收集器或使用提供的配置脚本来配置数据收集器。

过程

- 要以手动方式配置数据收集器，请从数据收集器软件包中获取数据收集器文件，然后修改 Liberty 服务器的一些本地文件。

  a) 运行以下命令，从数据收集器软件包中抽取文件。

  ```
  tar -xzf IBM_Data_Collectors_Install.tgz
  ```

  已抽取的目录中包含 `liberty_datacollector_8.1.4.0.tgz` 软件包。

  b) 使用以下命令将文件从 `liberty_datacollector_8.1.4.0.tgz` 软件包抽取到本地目录。已抽取的目录将变为数据收集器的主目录。

  ```
  tar -xzf liberty_datacollector_8.1.4.0.tgz
  ```

  例如，要将文件抽取到 `/opt/ibm/apm/` 目录，请发出以下命令:

  ```
  cd /opt/ibm/apm
  tar -xzf liberty_datacollector_8.1.4.0.tgz
  ```

  您可以在 `/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08` 目录中找到已抽取的文件。在以下步骤中该目录被称为数据收集器主目录 (*dc_home*)。

  c) 导航到 Liberty 服务器主目录。例如，`/opt/ibm/wlp/usr/servers/defaultServer`。

  d) 通过添加以下参数来编辑 `jvm.options` 文件。如果 `jvm.options` 文件不存在，请使用文本编辑器进行创建。

  ```
  -agentlib:am_ibm_16=server_name
  -Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
  -Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
  -Dliberty.home=liberty_home
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  -verbosegc
  -Xverbosegclog:absolute_path_to_log_file,1,10000
  ```

  添加条目时，请注意以下事项:

- 每个条目都必须独占一行。
- 将 *server_name* 替换为 Liberty 服务器的名称。
- 将 *dc_home* 替换为数据收集器的主目录。例如，/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08。
- 将 *liberty_home* 替换为 Liberty 安装目录的根目录。例如，/opt/ibm/wlp。
- 如果 Liberty 服务器要处理繁重的工作负载，请将 -Xmx 参数添加到数据收集器并分配额外 512M 堆大小。例如，-Xmx1024M。
- -Xhealthcenter:level=inprocess 和 -Xgc:allocationSamplingGranularity=10000 行可选。只有在想要启用内存分配收集（缺省情况下，禁用内存分配收集）的情况下，才添加这两行。“内存分析”仪表板需要启用内存分配收集以包含数据。
- -Xverbosegclog:*absolute_path_to_log_file*,1,10000 行可选，它将指定重定向垃圾回收日志文件的路径。如果未指定该路径，那么将日志写入一个文件，并每隔 10000 个分配故障进行循环操作。服务器运行时，原始 stdout 或 stderr 文件 (console.log) 可能非常大。如果要将垃圾回收输出日志文件保存到另一个目录，并且限制该日志文件数量和大小，请添加此行。如果指定的路径无效，那么该行不起作用，并且垃圾回收日志文件将保留 stdout 或 stderr 文件。

e) 打开同一目录中的 server.env 文件并将下列路径添加到环境条目中。如果 server.env 文件不存在，请使用文本编辑器进行创建。

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/lib:dc_home/toolkit/lib/lx8266:dc_home/
toolkit/lib/lx8266/ttapi
```

添加条目时，请注意以下事项：

- 每个条目都必须独占一行。
- 将 *dc_home* 替换为数据收集器的主目录。例如，/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08。

f) 修改同一目录中的 server.xml 以启用监视功能，方法是将以下行添加到 <featureManager> 部分：

```
<feature>monitor-1.0</feature>
```

g) 重新启动 Liberty 服务器。

· 要通过响应提示来配置数据收集器，请使用在数据收集器软件包中提供的配置脚本。

a) 运行以下命令，从数据收集器软件包中抽取文件。

```
tar -xzf IBM_Data_Collectors_Install.tgz
```

已抽取的目录中包含 liberty_datacollector_8.1.4.0.tgz 软件包。

b) 使用以下命令从 liberty_datacollector_8.1.4.0.tgz 软件包抽取文件。

```
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

例如，

```
cd /opt/ibm
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

已抽取的数据收集器文件位于 liberty_dc 目录中。

c) 通过运行以下命令，切换到 liberty_dc/.gdc/7.3.0.14.08/bin 目录并启动配置脚本：

```
./config_liberty_dc.sh
```

d) 出现提示时，请输入 Liberty 安装根目录或接受缺省值。例如，/opt/ibm/wlp。

e) 出现提示时，请输入应用程序服务器所使用的 JVM 主目录或接受缺省值。例如，/opt/ibm/java。

f) 配置程序可以自动发现和列出未在指定目录中配置的应用程序服务器。输入与要配置的 Liberty 服务器对应的编号。要选择多个服务器，请使用空格来分隔这些编号或输入 * 以全选。

g) 当配置程序完成更新所有 Liberty 服务器的文件后，请以手动方式更新 JVM 堆大小来为数据收集器分配额外的 512M。

h) 重新启动服务器以使配置生效。

结果

此数据收集器已配置完成，并已连接到 Cloud APM 服务器。资源监视、事务跟踪和诊断数据已启用。但是，堆收集和内存分配收集已禁用。如果需要“堆转储”和“内存分析”仪表板中的数据，那么可以使用数据收集器属性文件启用该数据。

下一步做什么

- 要查看 Liberty 服务器的监视数据，请启动 Cloud APM 控制台。有关指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

  切记: 当您在 应用程序仪表板 上添加 Liberty 数据收集器实例时，请从组件列表中选择 **Liberty Runtime**，而不是 **WebSphere Application Server**。
- 对于包含数据的“堆转储”仪表板和/或“内存分析”仪表板，您还需要启用堆快照收集器和/或内存分配收集的数据收集器，该操作可在数据收集器 `.properties` 文件中进行。请参阅第 759 页的『启用或禁用事务跟踪和诊断数据收集』。
- 如果密钥文件或 Cloud APM 服务器更改，请将数据收集器重新连接到 Cloud APM 服务器。有关指示信息，请参阅第 158 页的『将数据收集器重新连接到 Cloud APM 服务器』。

为内部部署应用程序取消配置数据收集器

如果不需要监视 Liberty 服务器，或者如果要将数据收集器升级到新版本，必须取消配置已经部署到 Liberty 服务器的数据收集器。

关于此任务

要取消配置已经部署到 Liberty 服务器的数据收集器，回滚在配置该数据收集器时所做的更改。您可以选择手动配置数据收集器，也可以选择使用提供的 `unconfig_liberty_dc` 脚本。

过程

- 要手动取消配置数据收集器，请完成以下步骤:

  a) 浏览至 Liberty 服务器主目录。例如，`/opt/ibm/wlp/usr/servers/defaultServer`。

  b) 编辑 `jvm.options` 文件以移除下列参数（如果有）。

  ```
  -agentlib:am_ibm_16=server_name
  -Xbootclasspath/p:dc_home/toolkit/lib/bcm-bootstrap.jar
  -Djava.security.policy=dc_home/itcamdc/etc/datacollector.policy
  -Dliberty.home=liberty_home
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  -verbosegc
  -Xverbosegclog:absolute_path_to_log_file,1,10000
  ```

  c) 编辑同一目录中的 `server.env` 文件以移除 **LD_LIBRARY_PATH** 的以下值

  ```
  /lib:dc_home/toolkit/lib/lx8266:dc_home/toolkit/lib/lx8266/ttapi
  ```

  其中，*dc_home* 是数据收集器的主目录。例如，`/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08`。

  d) 编辑同一目录中的 `server.xml` 文件以从 `<featureManager>` 部分中移除 `<feature>monitor-1.0</feature>`。

  e) 重新启动 Liberty 服务器。

- 要使用 `unconfig_liberty_dc.sh` 脚本取消配置数据收集器，请完成以下步骤:

a) 切换到 *dc_home*/bin 目录。例如，/opt/ibm/apm/liberty_dc/.gdc/7.3.0.14.08/bin。

b) 运行以下命令来启动取消配置脚本：

```
./unconfig_liberty_dc.sh
```

c) 出现提示时，请输入 Liberty 安装根目录或接受缺省值。例如，/opt/ibm/wlp。

d) 取消配置程序可以自动发现和列出在指定目录中配置的应用程序服务器。输入与要取消配置的 Liberty 服务器对应的编号。要选择多个服务器，请使用空格来分隔这些编号或输入 * 以全选。

e) 当取消配置程序完成更新所有 Liberty 服务器的文件后，请重新启动服务器以使更改生效。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

如果不再需要该数据收集器的主目录，您也可以将其删除。

为 IBM Cloud 应用程序配置 Liberty 数据收集器

要监视在 IBM Cloud 环境中运行的 Liberty 概要文件，您必须从 IBM Marketplace 下载数据收集器软件包，将数据收集器部署到您的本地应用程序文件，然后将更新推送到 IBM Cloud。

开始之前

假定使用 Cloud Foundry 命令将 Liberty 应用程序推送到 IBM Cloud 环境。manifest.yml 文件和 Liberty 服务器主目录（其中包含 server.xml 文件）已存在。

如果您的 Liberty 应用程序部署为 WAR 文件，必须修改某些本地文件，以通过推送包含 WAR 文件和数据收集器文件的本地目录来更新应用程序。以下提供了一个示例，说明如何在仅有 WAR 文件的情况下获得本地 Liberty 服务器主目录。

1. 发出以下命令以在本地运行 Liberty 应用程序：

   ```
   mvn install liberty:run-server
   ```

   在包含 Liberty WAR 文件的目录中，会创建一个子目录 /liberty/wlp/usr/servers/defaultServer。在以下过程中，该目录可以充当 Liberty 服务器主目录。
2. 从包含 Liberty WAR 文件的根目录，将整个 *application_name*-SNAPSHOT 文件夹复制到 /liberty/wlp/usr/servers/defaultServer 目录。
3. 在 /liberty/wlp/usr/servers/defaultServer 目录中，编辑 bootstrap.properties 文件以修改 **appLocation** 路径。**appLocation** 路径必须设置为相对于 IBM Cloud 中的应用程序目录的路径。
4. 移除 logs 和 workarea 文件夹。这两个文件夹无需推送到 IBM Cloud。
5. 修改 manifest.yml 文件中的 **path** 值以指向 defaultServer 目录。

   例如，path: target/liberty/wlp/usr/servers/defaultServer。

过程

完成下列步骤以配置 Liberty 数据收集器：

1. 从 IBM Marketplace 下载名为 IBM_Data_Collectors_Install.tgz 的数据收集器软件包。有关详细指示信息，请参阅第 86 页的『下载代理程序 以及数据收集器』。
2. 运行以下命令，从数据收集器软件包中抽取文件。

   ```
   tar -xzf IBM_Data_Collectors_Install.tgz
   ```

   已抽取的目录中包含 liberty_datacollector_8.1.4.0.tgz 软件包。

3. 将文件从 liberty_datacollector_8.1.4.0.tgz 软件包抽取到临时目录。

```
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

例如，

```
cd /root/tmp
tar -xzf liberty_datacollector_8.1.4.0.tgz
```

您可以在临时目录中的 liberty_dc 目录中找到已抽取的文件。

4. 将 .gdc 目录从 liberty_dc 目录复制到存储 server.xml 文件的 Liberty 服务器的主目录。Liberty 服务器主目录在以下步骤中被称为 *liberty_server_home*。

```
cp -rf temp_dir/liberty_dc/.gdc liberty_server_home
```

例如，

```
cp -rf /root/tmp/liberty_dc/.gdc /opt/liberty855/wlp/usr/servers/defaultServer/
```

5. 将 jvm.options 和 server.env 文件的内容从 liberty_dc/etc 目录复制或合并到 *liberty_server_home* 目录。
   - 如果 jvm.options 和 server.env 文件在 *liberty_server_home* 目录中不存在，请将这两个文件从 *temp_dir*/liberty_dc/etc 复制到 *liberty_server_home*。

     ```
     cp temp_dir/liberty_dc/etc/jvm.option liberty_server_home
     cp temp_dir/liberty_dc/etc/server.env liberty_server_home
     ```

   - 如果 jvm.options 或 server.env 文件在 *liberty_server_home* 目录中存在，请将该文件内容与 *temp_dir*/liberty_dc/etc 目录中的文件内容合并。
6. 如果由于网络或防火墙设置，IBM Cloud 应用程序无法直接连接到 Cloud APM 服务器，请配置数据收集器以通过转发代理发送流量。要执行此操作，请以下列其中一种方式来编辑 jvm.options 文件：
   - 如果不需要认证，那么将下列行添加到该文件：

     ```
     -Dhttp.proxyHost=http_proxy_host
     -Dhttp.proxyPort=http_proxy_port
     -Dhttps.proxyHost=http_proxy_host
     -Dhttps.proxyPort=http_proxy_port
     -Djava.net.useSystemProxies=true
     ```

   - 如果需要用户名和密码来访问转发代理服务器，那么将下列行添加到该文件：

     ```
     -Dhttp.proxyHost=http_proxy_host
     -Dhttp.proxyPort=http_proxy_port
     -Dhttp.proxyUser=http_proxy_user
     -Dhttp.proxyPassword=http_proxy_password
     -Dhttps.proxyHost=http_proxy_host
     -Dhttps.proxyPort=http_proxy_port
     -Dhttps.proxyUser=http_proxy_user
     -Dhttps.proxyPassword=http_proxy_password
     -Djava.net.useSystemProxies=true
     ```

7. 修改 Liberty 服务器主目录中的 server.xml 文件以启用监视功能，方法是将以下行添加到 <featureManager> 部分：

```
<featureManager>
                <feature>monitor-1.0</feature>
</featureManager>
```

8. 修改 Liberty 应用程序的 manifest.yml 文件以分配额外的 512M 内存。
9. 打开命令提示符，切换到 Liberty 服务器的包含 manifest.yml 文件的本地目录。例如，/opt/liberty855/。
10. 登录 IBM Cloud 并使用 **cf push** 命令更新 Liberty 概要文件。

结果
此数据收集器已配置完成，并已连接到 Cloud APM 服务器。资源监视、事务跟踪和诊断数据已启用。但是，堆收集和内存分配收集已禁用。如果需要“堆转储”和“内存分析”仪表板中的数据，那么可以使用数据收集器属性文件启用该数据。

下一步做什么

- 要查看 IBM Cloud 应用程序的监视数据，请启动 Cloud APM 控制台。有关指示信息，请参阅启动 Cloud APM 控制台。有关使用“应用程序”编辑器的信息，请参阅管理应用程序。

  切记: 当您要在 应用程序仪表板 上添加 Liberty 数据收集器实例时，请从组件列表中选择 **Liberty Runtime**，而不是 **WebSphere Application Server**。
- 对于包含数据的“堆转储”仪表板和/或“内存分析”仪表板，您还需要启用堆快照收集器和/或内存分配收集的数据收集器，该操作可在数据收集器 `.properties` 文件中进行。请参阅第 756 页的『使用属性文件定制数据收集器』。
- 如果密钥文件或 Cloud APM 服务器更改，请将数据收集器重新连接到 Cloud APM 服务器。有关指示信息，请参阅第 158 页的『将数据收集器重新连接到 Cloud APM 服务器』。

定制 Liberty 数据收集器的环境变量
要定制 IBM Cloud 应用程序的 Liberty 数据收集器，请使用 IBM Cloud UI 添加数据收集器所支持的环境变量。

提示: 要在 IBM Cloud UI 上添加环境变量，请先登录 IBM Cloud UI 并单击应用程序，然后单击**运行时** > **环境变量**。在**用户定义**部分中，添加环境变量。

- 使用第 754 页的表 228 中列出的变量来配置 Liberty 数据收集器和 Cloud APM 服务器之间的连接。
- 使用第 755 页的表 229 中列出的变量来启用或禁用 IBM Cloud 应用程序的方法跟踪。
- 启用方法跟踪后，使用第 755 页的表 230 中列出的变量来指定不同类型请求的阈值，以便可以收集不同级别的监视数据。

切记: 添加或修改环境变量后，重新启动应用程序以使更改生效。

表 *228.* 服务器连接的环境变量

变量名称	可能的值	描述
APM_BM_GATEWAY_URL	· `https://` *`server_ip_or_hostname`*`:443` · `http://` *`server_ip_or_hostname`*`:80`	目标 Cloud APM 服务器网关的 URL。
APM_KEYFILE_PSWD	密钥文件的已加密密码	与密钥文件配对的已加密密钥文件密码。如果您是 Linux 用户，可使用 `echo -n` *`<keyfile password>`* `| base64` 命令对密码进行加密。 **切记:** 只有在已将 *APM_BM_GATEWAY_URL* 变量配置为使用 HTTPS 的情况下，才应设置此变量。
APM_KEYFILE_URL	`http://` *`hosted_http_server`*`:`*`port`*`/keyfile.jks`	用来下载密钥文件的 URL。 **切记:** 只有在已将 *APM_BM_GATEWAY_URL* 变量配置为使用 HTTPS 的情况下，才应设置此变量。

表 *229.* 方法跟踪的环境变量		
变量名称	**可能的值**	**描述**
METHOD_TRACE_ENABLE	· true · false	使用此变量来启用或禁用方法跟踪。true 值将启用方法跟踪。 缺省值为 false。

启用方法跟踪之后，可以为不同类型的请求配置阈值，以定制方法跟踪。可以为每种请求类型配置下列阈值，它们将触发不同级别的监视数据收集：

主要阈值
如果配置请求类型的主要阈值，那么将捕获到此类请求的计时信息，例如此请求类型的 CPU 时间和响应时间。因此，当完成请求所花费的时间超过为主要阈值指定的时间时，将捕获到请求的计时。

次要阈值
如果配置请求类型的次要阈值，那么将捕获到深度上下文数据，例如数据库请求的堆栈跟踪和 SQL。捕获到的上下文数据随请求类型不同而有所变化。当完成请求所花费的时间超过为次要阈值指定的时间时，将捕获到其上下文数据。

不同请求阈值的环境变量命名为 *<request_type>_<threshold level>*。例如，要配置 JMS 请求的主要阈值，请添加 JMS_PRIMARY 变量，并设置它的值。

第 755 页的表 230 列出可以为不同请求类型添加的相应环境变量。值以毫秒为单位。

表 *230.* 不同请求阈值的环境变量	
变量名称	**缺省值（毫秒）**
SERVLET_PRIMARY	20
SERVLET_SECONDARY	50
JDBC_PRIMARY	20
JDBC_SECONDARY	50
JNDI_PRIMARY	20
JNDI_SECONDARY	50
EJB_PRIMARY	20
EJB_SECONDARY	50
WEBSERVICES_PRIMARY	20
WEBSERVICES_SECONDARY	50
APP_METHODS_PRIMARY （应用程序方法 - 非 J2EE）	50
APP_METHODS_SECONDARY	1000
JCA_PRIMARY	50
JCA_SECONDARY	80
JMS_PRIMARY	40
JMS_SECONDARY	70

使用属性文件定制数据收集器

缺省情况下会对数据收集器启用事务跟踪和方法跟踪。堆快照收集和内存分配收集处于禁用状态。您可以通过编辑数据收集器的 .properties 文件来定制数据收集或收集诊断数据的时间间隔。

关于此任务

数据收集器的属性文件位于 *dc_home* 目录中，例如，/opt/liberty855/wlp/usr/servers/defaultServer/.gdc/7.3.0.14.08。使用不同的属性来定制数据收集器以达到下列目的：

- 启用或禁用事务跟踪。
- 启用或禁用堆快照收集。
- 指定数据收集器获取堆转储快照的时间间隔。
- 启用或禁用内存分配监视。
- 指定数据收集器收集内存分配信息的时间间隔。
- 启用或禁用方法跟踪。

切记: 在修改 .properties 文件之后，使用 **cf push** 命令将更新推送到 IBM Cloud 环境。

过程

- 要启用或禁用事务跟踪，将以下文件中的 **com.ibm.tivoli.itcam.dc.bluemix.transaction.enabled** 属性设置为 true 或 false：

  dc_home/ldc/etc/ldc.properties

  如果启用事务跟踪，可以在拓扑中监视 IBM Java 应用程序堆栈。
- 要启用或禁用堆快照收集，将以下文件中的 **com.ibm.tivoli.itcam.hc.send.heap.enable** 和 **com.ibm.tivoli.itcam.hc.snapshot.automatic.enable** 属性设置为 true 或 false。

  dc_home/healthcenter/etc/hc.properties

  如果启用了堆快照收集，那么数据收集器可以在指定的时间间隔获取堆快照。堆转储信息可以显示在“堆转储”仪表板中。
- 要更改数据收集器获取堆快照的时间间隔，请将同一文件中的 **com.ibm.tivoli.itcam.hc.snapshot.automatic.interval** 属性设置为正整数。时间间隔单位是分钟，缺省值是 360。

  dc_home/healthcenter/etc/hc.properties
- 要启用或禁用内存分配收集，将以下文件中的 **com.ibm.tivoli.itcam.hc.events.collection.automatic.enable** 属性设置为 true 或 false。

  dc_home/healthcenter/etc/hc.properties

  切记: 要启用内存分配收集，还需要确保将以下两行添加到 Liberty 服务器的 jvm.options 文件。

  ```
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  ```

  在启用内存分配收集之后，“内存分析”仪表板中将有可用数据。
- 要指定收集内存分配信息的时间间隔，将同一文件中的 **com.ibm.tivoli.itcam.hc.events.collection.automatic.interval** 属性设置为正整数。时间间隔单位是分钟，缺省值是 15。

  dc_home/healthcenter/etc/hc.properties
- 要启用或禁用方法跟踪，将以下文件中的 **dfe.enable.methoddata** 属性设置为 true 或 false：

  dc_home/gdc/etc/gdc_dfe.properties

下一步做什么

- 在启用方法跟踪之后，可以使用环境变量来设置不同请求类型的阈值，这样可以针对不同的请求收集不同级别的监视数据。有关适用的环境变量，请参阅第 755 页的表 230。
- 如果禁用了内存分配收集，请记住从 Liberty 服务器的 `jvm.options` 文件移除以下行：

```
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
```

取消为 IBM Cloud 应用程序配置数据收集器

如果不需要监视 IBM Cloud 环境中的 Liberty 概要文件，或者如果要将数据收集器升级到新版本，必须取消配置先前部署的数据收集器。

关于此任务

要取消配置 IBM Cloud 环境中的 Liberty 概要文件的数据收集器，回滚 `jvm.options`、`server.env` 和 `server.xml` 文件中进行的更改，然后使用 **`cf push`** 命令更新 IBM Cloud 中的 Liberty 概要文件。

过程

1. 在 Liberty 概要文件的本地目录中，修改 `jvm.options` 文件以移除下列参数。如果更改后此文件为空，可以将其删除。

```
-agentlib:am_ibm_16=defaultServer
-Xbootclasspath/p:../../../../.gdc/7.3.0.14.08/toolkit/lib/bcm-bootstrap.jar
-Djava.security.policy=../../../../.gdc/7.3.0.14.08/itcamdc/etc/datacollector
.policy
-Dliberty.home=/home/vcap/app/.liberty
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
-verbosegc
-Xverbosegclog:/home/vcap/app/wlp/usr/servers/defaultServer/logs/gc.log,1,10000
```

2. 在 `server.env` 文件中，移除 **`LD_LIBRARY_PATH`** 环境变量的以下值。如果更改后此文件为空，可以将其删除。

```
LD_LIBRARY_PATH=:/lib:../../../../.gdc/7.3.0.14.08/toolkit/lib/lx8266
:../../../../.gdc/7.3.0.14.08/toolkit/lib/lx8266/ttapi
```

3. 修改 `server.xml` 文件，以通过将以下行移除到 `<featureManager>` 部分来移除 monitor-1.0 功能部件

```
<feature>monitor-1.0</feature>
```

4. 删除 Liberty 主目录中的 `.gdc` 目录。
5. 打开命令提示符，切换到包含 Liberty 服务器的 `manifest.yml` 文件的目录。
6. 登录 IBM Cloud 并使用 **`cf push`** 命令更新 Liberty 概要文件。

下一步做什么

在取消配置数据收集器后，Cloud APM 控制台 继续在已将数据收集器添加到的任何应用程序中显示数据收集器。Cloud APM 控制台将显示没有数据可用于应用程序并且不会指示数据收集器已脱机。有关如何从应用程序和资源组中移除数据收集器的信息，请参阅第 160 页的『从 Cloud APM 控制台中移除数据收集器』。

高级数据收集器配置

您可以修改数据收集器配置文件以更改其他监视设置。

Liberty 数据收集器的属性文件

为您提供了各种配置文件来进一步控制数据收集器的配置和行为。

将数据收集器软件包解压缩到本地目录后，数据收集器文件位于 *`local_dir`*`/liberty_dc/.gdc/7.3.0.14.08` 目录。例如，`/opt/ibm/apm/.gdc/7.3.0.14.08`。此文件夹成为数据收集器的主目录，表示为下列简化语句中的 *dc_home*。

数据收集器属性文件

数据收集器监视的每个应用程序服务器实例都具有其自己的属性文件。数据收集器会自动创建属性文件。文件名为 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/datacollector.properties。

为了方便以后升级，请不要更改此文件。

请改为将您要修改的设置添加到数据收集器定制属性文件中。此文件名为 *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/datacollector_custom.properties。数据收集器定制属性文件中的设置将覆盖数据收集器属性文件中的值。

要点: 如果 *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/datacollector_custom.properties 文件不存在，请在要进行更改时创建此文件。您可能还必须创建 custom 目录。

工具箱属性文件

数据收集器将在启动时使用各种输入文件自动创建工具箱属性文件。对于数据收集器监视的每个应用程序服务器实例，此文件是唯一的。其名称为 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/toolkit.properties。

由于数据收集器每次启动时都将重新创建此文件，因此**请不要对此文件进行任何更改**；如果进行了更改，那么这些更改将被覆盖。

请改为将您要修改的设置添加到工具箱定制属性文件中。此文件名为 *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties。工具箱定制属性文件中的设置将覆盖工具箱属性文件中的值。

您还可以为此数据收集器安装所监视的所有应用程序服务器实例设置工具箱属性。要执行此操作，请将设置添加到全局工具箱定制属性文件：*dc_home*/runtime/custom/toolkit_global_custom.properties。但是，如果在特定于实例的 toolkit_custom.properties 文件中设置了某个属性，那么此属性将覆盖此实例的全局文件中的值。

要点: 如果 *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties 或 *dc_home*/runtime/custom/toolkit_custom.properties 文件不存在，请在要进行更改时创建此文件。您可能还必须创建 custom 目录。

其他属性文件

在数据收集器属性文件和工具箱属性文件旁边，还存在特定于数据收集器监视的每个应用程序服务器实例的其他属性文件。

***dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties**
定义用于收集诊断和方法跟踪数据的详细信息。有关更改此文件的信息，请参阅第 760 页的『配置详细诊断信息的收集』。

***dc_home*/runtime/*appserver_version.node_name.server_name*/hc.properties**
定义堆快照集合和内存分配集合的详细信息。有关更改此文件的信息，请参阅第 760 页的『配置详细诊断信息的收集』。

***dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/cynlogging.properties**
定义日志文件名和数据收集器的 Java 部分的日志记录详细信息。

***dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/cyn-cclog.properties**

定义日志文件名和数据收集器的 C++ 部分的日志记录详细信息。

数据收集器跟踪文件

缺省情况下，数据收集器跟踪文件存储在以下位置：

- Windows *dc_home*\logs\CYN\logs。
- Linux AIX *dc_home*/logs/CYN/logs。

启用或禁用事务跟踪和诊断数据收集

缺省情况下会对数据收集器启用事务跟踪和方法跟踪。堆快照收集和内存分配收集处于禁用状态。您可以通过编辑数据收集器的 .properties 文件来定制数据收集或收集诊断数据的时间间隔。

关于此任务

数据收集器的属性文件位于 *dc_home* 目录中，例如，/opt/ibm/apm/.gdc/7.3.0.14.08。使用不同的属性来定制数据收集器以达到下列目的：

- 启用或禁用事务跟踪。
- 启用或禁用堆快照收集。
- 指定数据收集器获取堆转储快照的时间间隔。
- 启用或禁用内存分配监视。
- 指定数据收集器收集内存分配信息的时间间隔。
- 启用或禁用方法跟踪。

切记: 根据是否在数据收集器配置后重新启动了 Liberty 服务器，适用的 .properties 文件有所不同。如果在数据收集器配置之后重新启动了 Liberty 服务器，会在 *dc_home* 目录中创建 runtime 目录。之后，只能使用 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name* 目录中的 .properties 文件来定制每个应用程序服务器的数据收集器。

过程

- 要启用或禁用事务跟踪，将以下文件中的 **com.ibm.tivoli.itcam.dc.bluemix.transaction.enabled** 属性设置为 true 或 false：

  dc_home/runtime/*appserver_version.node_name.server_name*/ldc.properties（如果 runtime 目录不存在，请使用 *dc_home*/ldc/etc/ldc.properties）

  启用事务跟踪后，可以在拓扑中监视 IBM Java 应用程序堆栈。
- 要启用或禁用堆快照收集，将以下文件中的 **com.ibm.tivoli.itcam.hc.send.heap.enable** 和 **com.ibm.tivoli.itcam.hc.snapshot.automatic.enable** 属性设置为 true 或 false。

  dc_home/runtime/*appserver_version.node_name.server_name*/hc.properties （如果 runtime 目录不存在，请使用 *dc_home*/healthcenter/etc/hc.properties）

  启用堆快照收集后，数据收集器可以在指定的时间间隔获取堆快照。堆转储信息可以显示在“堆转储”仪表板中。
- 要更改数据收集器获取堆快照的时间间隔，请将同一文件中的 **com.ibm.tivoli.itcam.hc.snapshot.automatic.interval** 属性设置为正整数。时间间隔单位是分钟，缺省值是 360。

  dc_home/runtime/*appserver_version.node_name.server_name*/hc.properties （如果 runtime 目录不存在，请使用 *dc_home*/healthcenter/etc/hc.properties）

- 要启用或禁用内存分配收集，将以下文件中的 **com.ibm.tivoli.itcam.hc.events.collection.automatic.enable** 属性设置为 true 或 false。

  dc_home/runtime/*appserver_version.node_name.server_name*/hc.properties （如果 runtime 目录不存在，请使用 *dc_home*/healthcenter/etc/hc.properties）

  切记: 要启用内存分配收集，还需要确保将以下两行添加到 Liberty 服务器的 jvm.options 文件。

  ```
  -Xhealthcenter:level=inprocess
  -Xgc:allocationSamplingGranularity=10000
  ```

  在启用内存分配收集之后，“内存分析”仪表板中将有可用数据。
- 要指定收集内存分配信息的时间间隔，将同一文件中的 **com.ibm.tivoli.itcam.hc.events.collection.automatic.interval** 属性设置为正整数。时间间隔单位是分钟，缺省值是 15。

  dc_home/runtime/*appserver_version.node_name.server_name*/hc.properties （如果 runtime 目录不存在，请使用 *dc_home*/healthcenter/etc/hc.properties）
- 要启用或禁用方法跟踪，将以下文件中的 **dfe.enable.methoddata** 属性设置为 true 或 false：

  dc_home/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties（如果 runtime 目录不存在，请使用 *dc_home*/gdc/etc/gdc_dfe.properties）

结果

修改 .properties 文件之后，重新启动 Liberty 服务器以使更改生效。

有关每个应用程序服务器的数据收集器 .properties 文件的更多信息，请参阅第 757 页的『Liberty 数据收集器的属性文件』。

下一步做什么

- 在启用方法跟踪之后，可以设置不同请求类型的阈值，这样可以针对不同的请求收集不同级别的监视数据。有关指示信息，请参阅第 762 页的『定制请求阈值』。
- 如果禁用了内存分配收集，请记住从 Liberty 服务器的 jvm.options 文件移除以下行：

```
-Xhealthcenter:level=inprocess
-Xgc:allocationSamplingGranularity=10000
```

配置详细诊断信息的收集

如果您有 IBM Cloud Application Performance Management, Advanced，可使用数据收集器来收集有关受监视应用程序服务器实例的详细诊断信息。要配置诊断数据收集行为（包括数据收集器存储的诊断信息量），请定制 gdc_custom.properties 配置文件。

关于此任务

您可以在 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc 目录中查找 gdc_custom.properties 文件。

以下示例描述了如何使用 gdc_custom.properties 配置文件中的属性完成下列操作：

- 设置详细信息文件的大小和数目的限制
- 设置请求和方法诊断数据的完全或部分收集

您还可以在 gdc_custom.properties 文件中设置其他属性以定制诊断数据收集。请参阅文件中用于描述这些属性的注释。

切记: 编辑完 gdc_custom.properties 文件后，您必须重新启动受监视的应用程序服务器实例以使更改生效。

设置详细信息文件的大小和数目的限制

关于此任务

数据收集器在多个文件中存储诊断信息。缺省情况下，它会收集 100 个文件；如果已存储 100 个文件且创建了新文件，那么会删除最旧的文件。数据收集器每 15 分钟或当前文件大小超出 200 兆字节时会创建一个新文件。包含这些文件的目录的总大小超出 2 千兆字节时，数据收集器会删除最旧的文件。

过程

您可以在 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties 文件中更改以下设置：

- 要设置具有诊断信息的最大文件数，请设置 com.ibm.itcam.gdc.dfe.filelimit 属性。
  例如：

  ```
  com.ibm.itcam.gdc.dfe.filelimit=100
  ```

- 要设置在数据收集器创建新诊断数据文件之前经过的时间（以分钟计），请设置 com.ibm.itcam.gdc.dfe.frequency 属性。
  例如：

  ```
  com.ibm.itcam.gdc.dfe.frequency=15
  ```

- 要设置最大诊断数据文件大小（以兆字节计），请设置 dfe.file.maxlimit 属性。
  例如：

  ```
  dfe.file.maxlimit=200
  ```

  如果当前诊断数据文件达到此大小，那么数据收集器会创建一个新的诊断数据文件。
- 要设置所有数据文件的最大总大小（以字节计），请设置 trace.dir.size.limit 属性。
  例如：

  ```
  trace.dir.size.limit=2147483648
  ```

  如果所有诊断数据文件的大小总计超出此值，那么数据收集器会删除最旧的数据文件。最小总大小为 25 兆字节。

设置请求和方法诊断数据的完全或部分收集

关于此任务

数据收集器具有以下缺省设置：

- 数据收集器仅为所选请求收集诊断数据。对请求进行选择（采样）旨在包含所有错误和一些良好请求。
- 服务器启动时禁用了方法数据收集。
- 启用方法数据收集时，数据收集器仅收集某些请求（收集其诊断数据的请求）的方法数据。这种进一步选择（采样）同样旨在包含所有错误和一些良好请求。

要点： 更改这些设置会影响应用程序服务器的性能。在生产服务器上，性能降低可能很严重。

过程

您可以使用 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/gdc/gdc_custom.properties 文件中的属性来更改这些设置。

- 要启用方法数据收集，请按以下内容设置属性：

  ```
  dfe.enable.methoddata=true
  ```

  提示： 您还可以使用“**代理程序配置**”页面动态启用或禁用方法跟踪数据收集。
- 要为每个请求收集诊断数据，请按以下内容设置属性来禁用采样：

```
dc.sampling.enable=false
```

- 要为对其收集诊断数据的每个请求启用方法数据收集，请按以下内容设置属性：

```
dc.sampling.enable=false
dc.sampling.methsampler.enabled=false
```

切记： 仅在“代理程序配置”页面上或者通过 dfe.enable.methoddata 属性启用了方法数据收集时，dc.sampling.methsampler.enabled 属性才会生效。

定制请求阈值

如果缺省阈值较高，某些请求可能没有足够信息。您可以定制请求阈值，以便数据收集器能够捕获更多的请求或请求上下文数据。

关于此任务

每种请求类型都具有名为 **perfThreshold** 和 **secondaryPerfThreshold** 的两种阈值类型。仅当所用时间超过为 **perfThreshold** 阈值指定的时间时，数据收集器才捕获请求。仅当请求所用时间超过为 **secondaryPerfThreshold** 阈值指定的时间时，才会捕获诸如堆栈跟踪和 SQL 语句之类的上下文数据。您可以调整这些阈值以满足您的需求。

过程

1. 请转至 *dc_home*\gdc\etc 目录，其中，*dc_home* 为数据收集器的主目录。
2. 在文本编辑器中，打开要定制的请求类型的 XML 文件。您可以确定哪些文件适用于来自 XML 文件名的哪些请求类型。
   例如，ejb.xml 文件适用于 EJB 请求，custom.xml 文件适用于定制请求，appMethods.xml 文件适用于类和方法（启用方法跟踪后）。
3. 将 <collectContextData>、<collectStackTrace> 和 <createDataRow> 标记设置为 ifThresholdExceeded。

```
<collectContextData>ifThresholdExceeded</collectContextData>
<collectStackTrace>ifThresholdExceeded</collectStackTrace>
<createDataRow>ifThresholdExceeded</createDataRow>
```

4. 将 <perfThreshold> 和 <secondaryPerfThreshold> 标记设置为所需阈值。阈值的单位是毫秒。例如，ejb.xml 文件具有 EJB 请求的以下设置。因此，数据收集器仅捕获耗用时间超过 1 秒（1000 毫秒）的 EJB 请求。此外，仅当 EJB 请求所用时间超过 1.5 秒（1500 毫秒）时，才捕获与 EJB 请求有关的上下文数据（如堆栈跟踪和 EJB 主目录）。

```
<requestProbePoint id="EJB">
<interface>com.ibm.tivoli.itcam.toolkit.ai.boot.aspectmanager.ITurboEJBEventListener</
interface>
<family>EJB</family>
<collectContextData>ifThresholdExceeded</collectContextData>
<collectStackTrace>ifThresholdExceeded</collectStackTrace>
<perfThreshold>1000</perfThreshold>
<secondaryPerfThreshold>1500</secondaryPerfThreshold>
<dataToCollect>instanceAndSummary</dataToCollect>
<createDataRow>ifThresholdExceeded</createDataRow>
<requestType>EJB Method</requestType>
</requestProbePoint>
```

5. 保存您的更改并重新启动应用程序服务器。

禁用针对 Java EE API 的各种类型的字节代码检测

在字节代码检测 (BCI) 中，数据收集器会拦截各种类型 Java Platform Enterprise Edition (Java EE) API 的方法入口和出口调用，以创建每个应用程序请求的执行流。使用了一些资源来进行监视。您可以调整数据收集器，以便不监视某些 API，从而减少资源使用。

要禁用 Java EE API 的 BCI 监视，请将以下属性添加到工具箱定制属性文件。有关此文件的更多信息，请参阅第 758 页的『工具箱属性文件』。

表 *231.* 将行添加到工具箱定制属性文件

Java EE API 的类型	**要添加到 toolkit_custom.properties 文件的行**
Enterprise JavaBeans (EJB)	`com.ibm.tivoli.itcam.toolkit.ai.enableejb=false`
Java 连接器体系结构 (JCA)	`com.ibm.tivoli.itcam.toolkit.ai.enablejca=false`
Java 数据库连接 (JDBC)	`com.ibm.tivoli.itcam.toolkit.ai.enablejdbc=false`
Java 命名和目录接口 (JNDI)	`com.ibm.tivoli.itcam.toolkit.ai.enablejndi=false`
Java 消息服务 (JMS)	`com.ibm.tivoli.itcam.toolkit.ai.enablejms=false`
Servlet/JavaServer Pages (JSP) 的 Web 容器	`com.ibm.tivoli.itcam.dc.was.webcontainer=false`
HTTP 会话计数跟踪	`com.ibm.tivoli.itcam.toolkit.ai.enablesessioncount=false`
CICS® Transaction Gateway (CTG)	`com.ibm.tivoli.itcam.dc.ctg.enablectg=false`
IMS	`com.ibm.tivoli.itcam.dc.mqi.enableims=false`
Java 数据对象 (JDO)	`com.ibm.tivoli.itcam.dc.mqi.enablejdo=false`
消息队列接口 (MQI)	`com.ibm.tivoli.itcam.dc.mqi.enablemqi=false`
轴 Web service	`com.ibm.tivoli.itcam.toolkit.ai.axis.enablewebservice=false`
远程方法调用 (RMI)	`am.ejb.rmilistener.enable=false`
WebSphere Application Server EJB 容器	`com.ibm.tivoli.itcam.dc.was.enableEJBContainer=false`

针对特定的事务类型来禁用事务跟踪

为数据收集器启用事务跟踪之后，缺省情况下，将监视所有的事务类型。您可使用工具箱属性文件，针对特定的事务类型来禁用事务跟踪。

关于此任务

编辑 `toolkit_custom.properties` 文件可针对每一个应用程序服务器来定制事务跟踪，而编辑 `toolkit_global_custom.properties` 文件可以针对所有应用程序服务器实例来定制。

在下列过程中，使用的是单个应用程序服务器的 `toolkit_custom.properties` 文件。在 `toolkit_global_custom.properties` 文件中也支持这些属性。有关工具箱属性文件的更多信息，请参阅 Liberty 数据收集器的属性文件。

过程

1. 使用文本编辑器，打开应用程序服务器的 `toolkit_custom.properties` 文件。此文件可在下列目录中找到：

dc_home/runtime/*app_server_version*.*node_name*.*profile_name*.*server_name*/custom

2. 根据需要，指定下列一个或多个属性，并将属性值设置为 false，以针对特定的事务类型禁用事务跟踪。

   对于 CICS 请求
   com.ibm.tivoli.itcam.dc.ttapi.cics.enabled=false

   对于定制请求
   com.ibm.tivoli.itcam.dc.ttapi.ims.enabled=false

   对于 EJB 请求
   com.ibm.tivoli.itcam.dc.ttapi.ejb.enabled=false

   对于 HTTP 客户机调用
   com.ibm.tivoli.itcam.dc.ttapi.httpclient.enabled=false

   异常: 要针对 Apache HTTP 客户机调用来禁用事务跟踪，请指定 com.ibm.tivoli.itcam.toolkit.dc.enable.apache.httpclient=false。

   对于 IMS 请求
   com.ibm.tivoli.itcam.dc.ttapi.ims.enabled=false

   对于 JDBC 请求
   com.ibm.tivoli.itcam.dc.ttapi.jdbc.enabled=false

   对于 JMS 请求
   com.ibm.tivoli.itcam.dc.ttapi.jms.enabled=false

   对于 JNDI 请求
   com.ibm.tivoli.itcam.dc.ttapi.jndi.enabled=false

   对于 MQI 请求
   com.ibm.tivoli.itcam.dc.ttapi.mqi.enabled=false

   对于门户网站请求
   com.ibm.tivoli.itcam.dc.ttapi.portal=false

   对于 RMI-IIOP 请求
   com.ibm.tivoli.itcam.dc.ttapi.rmiiiop.enabled=false

   对于 Servlet 请求
   com.ibm.tivoli.itcam.dc.ttapi.arm.servlet.enabled=false

   对于 Web Service 请求
   com.ibm.tivoli.itcam.dc.ttapi.webservice.enabled=false

   提示: 有关这些属性的更多信息，请参阅 *dc_home*/ttdc/etc/ttdc.properties 文件。

3. 保存并关闭 toolkit_custom.properties 文件。
4. 重新启动应用程序服务器以使更改生效。

将类排除在监视之外
您可定制数据收集，将特定的类排除在监视之外。这项定制操作是使用工具箱属性文件来完成。

关于此任务

编辑 toolkit_custom.properties 文件可针对每一个应用程序服务器来定制事务跟踪，而编辑 toolkit_global_custom.properties 文件可以针对所有应用程序服务器实例来定制。

在下列过程中，使用的是单个应用程序服务器的 toolkit_custom.properties 文件。在 toolkit_global_custom.properties 文件中也支持这些属性。有关工具箱属性文件的更多信息，请参阅 Liberty 数据收集器的属性文件。

过程

1. 使用文本编辑器，打开应用程序服务器的 toolkit_custom.properties 文件。此文件可在下列目录中找到:

dc_home/runtime/*app_server_version*.*node_name*.*profile_name*.*server_name*/custom

2. 编辑该文件，添加下列属性并保存更改。

```
am.camtoolkit.gpe.customxml.exclude=excludes.xml
```

3. 在同一个 custom 目录中，创建包含以下内容的 excludes.xml 文件，并在 <exclude> 标记内指定要排除的类名。您可根据需要添加任意数量的类，而且支持星号通配符 (*)。

```
<gpe>
  <bci>
    <classExcludes>
      <exclude>class_name_to_be_exclued</exclude>
      <exclude>class_name_to_be_exclued</exclude>
    </classExcludes>
  </bci>
</gpe>
```

示例：

```
<gpe>
  <bci>
    <classExcludes>
      <exclude>org.apache.struts.action.ActionServlet</exclude>
      <exclude>com.company.package.*</exclude>
    </classExcludes>
  </bci>
</gpe>
```

4. 重新启动应用程序服务器。

下一步做什么

要确认已排除该类，请查看 toolkit.xml 文件，该类名应该在 <classExcludes> 小节中列出。

切记: toolkit.xml 文件包含运行时设置，而且会在应用程序服务器每次重新启动时更新。

定制请求信息映射

在某些情况下，您可能必须更改用于标识代理程序所监视的请求的信息。此信息包括请求名称，以及可对该请求显示的任何数据（例如，SQL 请求的查询文本）。要更改信息，请设置定制请求映射器配置。

关于此任务

要定制请求信息映射，您必须在 XML 文件中定义定制请求映射器配置。

在此文件中，某些内置符号表示来自请求的运行时上下文的值。您可以创建额外的符号，用于计算新值。计算可以包括原始请求值、表达式、Java 方法（包括受监视应用程序中的方法）调用、条件句以及对一组值的迭代。

然后，您可以将符号内容映射到为 Cloud APM 服务器提供的新请求数据。如果未映射请求数据中的特定变量，将保留原始值。

由于针对请求类型收集不同的数据，定制请求映射器配置必须特定于请求类型。您可以针对同一数据收集器上的不同请求类型配置不同的请求映射器。

过程

要针对一种请求类型设置定制请求映射器配置，请完成以下步骤：

1. 在 XML 文件中定义定制请求映射器配置。
   有关 XML 语法的信息，请参阅 第 766 页的『XML 文件语法』。
2. 将 XML 文件放在 *dc_home*/runtime/custom 目录中以将其用于所有应用程序服务器实例，或将 XML 文件放在 *dc_home*/runtime/*appserver_version*.*node_name*.*profile_name*.*server_name*/custom 目录中以将其用于一个应用程序服务器实例。
3. 在工具箱定制配置文件 toolkit_custom.properties 或 toolkit_global_custom.properties 中针对此类型启用定制请求映射。
   有关指示信息，请参阅第 774 页的『启用请求映射器』。

4. 从同一工具箱定制配置文件中引用您定义的 XML 文件。
   有关指示信息，请参阅第 774 页的『启用请求映射器』。

XML 文件语法

针对请求映射器配置创建的 XML 文件必须是有效的 XML 并且必须在使用配置时保持可用。将 XML 文件放在 *dc_home*/runtime/custom 目录中以将其用于所有应用程序服务器实例，或将 XML 文件放在 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom 目录中以将其用于一个应用程序服务器实例。

顶级

顶级标记是 <gpe>。在此标记中，使用 <runtimeConfiguration> 标记。这些标记没有属性。

在 <runtimeConfiguration> 标记中，创建 <requestMapperDefinition> 标记。此标记必须具有 type 属性。针对所需的请求类型，将其设置为请求映射器类型名称。有关更多信息，请参阅第 776 页的表 233。

在 <requestMapperDefinition> 标记中，以下两个标记必须存在：

<symbolDefinitions>
包含符号的所有定义。符号表示每次检测到此类型的请求时代理程序计算的值。

<selection>
包含上下文密钥到值的映射。密钥表示传递到代理程序的定制数据。针对每种请求类型对其进行预定义。映射可以是有条件的。

有关请求映射器启用属性和类型名称的更多信息，请参阅第 776 页的表 233。

此外，在 <runtimeConfiguration> 标记中，您可以创建 <requestMapperClassPath> 标记。在此标记中，您可以定义 JAR 文件。您可以在请求映射器定义中引用这些 JAR 文件中的 Java 类。

定义表达式
要定义符号，您必须使用表达式。代理程序对表达式进行求值，以对符号指定值。

在表达式中使用数据

表达式可以使用以下数据：

- 请求类型的输入数据符号
- 同一请求映射器定义中描述的其他符号
- 数字常数
- 字符串常量（以 " 定界，如 "string"）
- 布尔常量（true、TRUE、false、FALSE）
- null 常量

有关输入数据符号的更多信息，请参阅第 777 页的表 234。

如果符号的值是 Java 类的实例化，那么表达式可以包含对该类中定义的字段和方法的引用。要引用字段，请使用 *symbol.fieldname*。要引用方法，请使用 *symbol.methodname*(*parameters*)。方法调用必须返回一个值。例如，可以将 Java 字符串方法与具有字符串值的符号配合使用。

要引用类的静态字段或方法，您还可以使用 *classname.fieldname* 和 *classname.methodname*(*parameters*)。

如果符号引用数组对象，那么表达式可以选择一个元素 (*symbol*[*selector*]) 并确定数组长度 (*symbol*.length)

运算符

您可以在表达式中使用以下运算符：

- 布尔运算符：AND、&、OR、|、NOT、!
- 比较：==、!=、GT、>、LT、<、GE、>=、LE、<=

· 数字运算符：+、-、*、/

· 用于强制执行求值顺序的括号：（、）

要点: 必须对 XML 中的符号 <、> 和 & 进行转义。您也可以使用 GT（大于）、GE（大于或等于）、LT（小于）、LE（小于或等于）以及 AND 运算符。

通过使用 instanceof 运算符，表达式可以求出值是否是类的实例：

```
expression instanceof java.class.name
```

与 Java instanceof 运算符类似，此运算符生成布尔值。在此示例中，如果 *expression* 值所属的类满足任何以下条件，值为 true：

· 命名为 *java.class.name*

· 是由 *java.class.name* 标识的类的直接或间接子类。

· 直接或间接实现由 *java.class.name* 标识的接口。

通过使用 new 运算符，表达式还可以使 Java 类的新对象实例化。此运算符类似于 Java new 运算符：

```
new java.class.name(expression1, expression2, ... expressionN)
```

运算符优先级

运算符按优先级顺序进行求值。同一优先级顺序的运算符从左向右进行求值。您可以使用括号（和）来更改求值顺序。

优先级顺序如下所示：

1. . 运算符（方法调用或字段引用）
2. [ ]（数组元素选择器）
3. new
4. !、NOT
5. *, /
6. +, -
7. GT、>、LT、<、GE、>=、LE、<=、instanceof
8. ==, !=
9. AND、&
10. OR、|

示例

```
$s1 >= ( 2 * ($s2.sampMethod($s3, true) + 1))
```

此代理程序按以下方式对此表达式进行求值：

1. 对 $s1 符号进行求值。它必须生成数字值。
2. 对 $s2 符号进行求值。它必须生成 Java 对象。
3. 对 $s3 符号进行求值。
4. 对 $s2 求值产生的对象的 sampMethod 方法进行调用。$s3 的求值结果将作为第一个参数传递，布尔值 true 将作为第二个参数传递。对 sampMethod 的调用必须返回一个数字值。
5. 将 1 添加到步骤 第 767 页的『4』 的结果中。
6. 步骤 第 767 页的『5』 的结果乘以 2。
7. 将步骤 第 767 页的『1』 的结果与步骤 第 767 页的『6』 的结果进行比较。如果步骤 第 767 页的『1』 的结果大于或等于步骤 第 767 页的『6』 的结果，那么返回 true。否则，返回 false。

定义基本符号
要定义符号，您必须使用表达式。代理程序对表达式进行求值，以对符号指定值。

在 <symbol> 标记中，使用以下标记：

<name>
符号的名称。它是一个字符串，且必须以 $ 字符开头。

<eval>
代理程序必须对其进行求值以生成此符号的值的表达式。有关定义表达式的更多信息，请参阅第 766 页的『定义表达式』。

<type>
符号返回的值的类型。将此值指定为标准 Java 类名或 Java 原语。为符号指定类型是可选的。如果未定义，请求映射器尝试根据表达式建立字段类型。如果请求映射器无法在表达式求值前确定符号类型，将影响性能。因此，要获得最佳性能，最好指定类型。

<args>
符号的参数。此标记是可选的；如果已指定，必须提供用于对符号进行求值的参数。有关更多信息，请参阅第 768 页的『定义符号参数』。

示例

```
<symbol>
   <name>$doubles1</name>
   <eval>$s1*2</eval>
   <type>int</type>
</symbol>
```

此符号返回的值是另一符号 $s1 的值的两倍。

定义符号参数
在符号定义的 <args> 标记中，您可以定义符号的参数类型。

在此标记中，使用 <type> 标记可指定参数的类型。将此值指定为标准 Java 类名或 Java 原语。您可以指定任何数目的 <type> 标记；每个标记定义一个参数。

在这种情况下，通过将参数括在括号内来引用符号。

```
$symbol(argument1,argument2...)
```

参数的数目必须与参数类型定义的数目相同。

在符号定义中，引用第一个参数作为 $p0，引用第二个参数作为 $p1，以此类推。

具有参数的符号类似于 Java 方法。它使用输入参数，并返回依赖于这些参数值的值。

示例

```
<symbol>
   <name>$double</name>
   <eval>$p0*2</eval>
   <type>int</type>
   <args>
       <type>int</type>
   </args>
</symbol>
```

此符号返回的值是参数值的两倍。要对它进行求值，请提供数字参数：$double(2)、$double($s1)。

定义迭代符号
在 <symbolDefinitions> 标记中，您可以使用 <iterationSymbol> 标记定义迭代符号。迭代符号表示通过循环访问 Java 数组、枚举或集合中的一组对象获取的值。对于每个成员，请求映射器对一个或多个条件表达式进行求值。如果表达式返回 true，那么请求映射器使用成员来计算返回值。某个成员满足条件表达式时，请求映射器不会对其余成员进行求值。

在 <iterationSymbol> 标记中，使用以下标记。

<name>
符号的名称。它是一个字符串，且必须以 $ 字符开头。

<type>
符号返回的值的类型。将此值指定为标准 Java 类名或 Java 原语。为符号指定类型是可选的。如果未定义，请求映射器尝试根据表达式建立字段类型。如果请求映射器无法在表达式求值前确定符号类型，将影响性能。因此，要获得最佳性能，最好指定类型。

<args>
符号的参数。此标记是可选的；如果已指定，必须提供用于对符号进行求值的参数。有关更多信息，请参阅第 768 页的『定义符号参数』。

<iterate over="*expression*">
定义包含要循环访问的成员的对象（数组、枚举或集合）。表达式必须返回该对象。请求映射器对其成员进行迭代，直到任何一个成员导致条件表达式返回 `true`，或未保留更多成员。在此标记中定义标记中的该组迭代表达式：

<test>
在此标记中定义条件并返回表达式。<iterate> 标记可以包含多个 <test> 标记。在这种情况下，请求映射器对所有表达式进行求值。如果任何条件表达式都为 true，那么通过在同一 <test> 标记中使用结果表达式，符号将返回一个值，并且不进一步执行求值。

<castTo>
可选：如果存在此标记，请在其中指定 Java 类型的名称，作为标准 Java 类名或 Java 原语。在对条件进行求值和返回表达式前，请求映射器将迭代元素转换为此类型。如果此标记不存在，那么请求映射器将数组成员转换为数组基本类型，并将“枚举”或“集合”的成员转换为 `java.lang.Object`。对于数组成员，数组基本类型通常是正确的选择；因此，将此标记用于请求映射器以对枚举或集合进行迭代。

<condition>
必须生成布尔值的表达式。使用 `$iterElement` 来引用正在迭代的元素。

<return>
如果 <condition> 标记中的表达式返回 `true`，请求映射器将对 <return> 标记中的表达式进行求值。迭代符号将返回此表达式生成的值。使用 `$iterElement` 来引用正在迭代的元素。

<defaultValue>
可选。如果请求映射器已对此对象的所有成员进行迭代，但是没有条件表达式返回 `true`，请求映射器将对 <defaultValue> 标记中的表达式进行求值。迭代符号将返回此表达式生成的值。如果此标记不存在，缺省值为 `null`。

示例

```
<iterationSymbol>
  <name>$userNameCookieValue</name>
  <iterate over="$httpServletRequest.getCookies()">
     <test>
        <condition>$iterElement.getName().equals("userName")</condition>
        <return>$iterElement.getValue()</return>
     </test>
   </iterate>
</iterationSymbol>
```

此符号查找名为“username”的 cookie，并返回其值。`$httpServletRequest.getCookies()` 返回数组，因此不需要 <castTo> 元素。

```
<iterationSymbol>
  <name>$headerNameStartingWithA</name>
  <iterate over="$httpServletRequest.getHeaderNames()">
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.startsWith("A")</condition>
       <return>$iterElement</return>
    </test>
```

```
  </iterate>
</iterationSymbol>
```

此符号查找名称以“A”开头的头，并返回其名称。`$httpServletRequest.getHeaderNames()` 返回枚举，因此需要 <castTo> 元素。

```
<iterationSymbol>
  <name>$determined_gender</name>
  <iterate over="$children">
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.equals("male")</condition>
       <return>"It's a boy"</return>
    </test>
    <test>
       <castTo>java.lang.String</castTo>
       <condition>$iterElement.equals("female")</condition>
       <return>"It's a girl"</return>
    </test>
  </iterate>
  <defaultValue>"unknown"</defaultValue>
</iterationSymbol>
```

此符号对 `$children` 进行迭代，它必须是数组、枚举或字符串集合。如果任何一个字符串等于“male”，将返回“it's a boy”。如果任何一个字符串等于“female”，将返回“it's a girl”。最后，如果 `$children` 中的字符串既不等于“male”，也不等于“female”，符号返回“unknown”。

定义条件符号

在 <symbolDefinitions> 标记中，您可以使用 <conditionalSymbol> 标记定义条件符号。条件符号表示通过对一系列条件表达式求值得到的值。如果任何表达式返回 `true`，那么请求映射器使用成员来计算返回值。某个成员满足条件表达式时，请求映射器对相应的返回表达式进行求值，并返回结果。请求映射器找到要返回的结果后，不会再对任何表达式进行求值。

在 <conditionalSymbol> 标记中，使用以下标记。

<name>
符号的名称。它是一个字符串，且必须以 $ 字符开头。

<type>
符号返回的值的类型。将此值指定为标准 Java 类名或 Java 原语。为符号指定类型是可选的。如果未定义，请求映射器尝试根据表达式建立字段类型。如果请求映射器无法在表达式求值前确定符号类型，将影响性能。因此，要获得最佳性能，最好指定类型。

<args>
符号的参数。此标记是可选的；如果已指定，必须提供用于对符号进行求值的参数。有关更多信息，请参阅第 768 页的『定义符号参数』。

<if condition="*expression*">
`condition` 属性定义要求值的条件表达式。此表达式必须生成布尔值。如果值为 `true`，请求映射器使用 <if> 标记的内容尝试确定返回值。<if> 标记必须包含以下某项内容，但不能包含所有内容：

- <return> 标记。此标记包含一个表达式。如果条件表达式为 true，请求映射器对表达式进行求值并返回结果。
- 任何数目的 <if> 标记，嵌套到此 <if> 标记中。如果条件表达式为 true，请求映射器按处理顶级 <if> 标记的方式来处理嵌套的 <if> 标记。即它将对 `condition` 属性中的表达式进行求值，如果表达式为 true，使用标记的内容尝试并确定返回值。

要点: 如果确定了返回值，请求映射器不再对任何表达式进行求值。但是，如果 <if> 标记中的条件表达式为 true，但是它包含嵌套的 <if> 标记并且其条件表达式不为 true，那么不确定值。在这种情况下，请求映射器继续对后续表达式进行求值。

<defaultValue>
可选。如果请求映射器已对所有条件表达式进行求值，但是条件表达式不返回 `true`，请求映射器将对 <defaultValue> 标记中的表达式进行求值。条件符号返回此表达式生成的值。如果此标记不存在，缺省值为 `null`。

示例

```
<symbol>
  <name>$GET</name>
  <eval>"GET"</eval>
</symbol>
<symbol>
  <name>$PUT</name>
  <eval>"PUT"</eval>
</symbol>
<conditionalSymbol>
  <name>$sessionAttribute</name>
  <if condition="$httpServletRequest.getSession(false) != null>
     <if condition="$httpServletRequest.getSession(false).getAttribute($GET)
!= null">
       <return>$httpServletRequest.getSession(false).getAttribute($GET)</return>
     </if>
     <if condition="true">
       <return>$httpServletRequest.getSession(false).getAttribute($PUT)</return>
      </if>
  </if>
</conditionalSymbol>
```

假定此符号是 servlet 请求映射器的组成部分。首先，它将检查 servlet 是否存在 HTTP 会话；如果不存在，符号返回 `null`。如果存在会话，符号检查 servlet 是否具有 GET 属性，它将返回该属性的值。否则，返回 PUT 属性的值。第二个条件表达式为 true；此值用作 else 子句。如果第一个条件为 true，请求映射器不再对任何表达式进行求值；否则，它将继续对第二个表达式进行求值。

定义外部类符号

在 <symbolDefinitions> 标记中，您可以使用 <externalClassSymbol> 标记定义外部类。外部类符号表示外部 Java 类。外部类符号定义是可选的；您可以在表达式中直接使用外部 Java 类。但是，它可能会增强请求映射器配置的可读性。

在 <externalClassSymbol> 标记中，使用以下标记。

<name>
符号的名称。它是一个字符串，且必须以 $ 字符开头。

<className>
客户定义的类的名称。

要点: 要引用请求映射器配置中的任何 Java 类（无论是外部类符号定义中的还是任何表达式中的），您必须将包含此类的 JAR 文件的完整路径和名称添加到 <runtimeConfiguration> 标记中的 <requestMapperClassPath>标记。

定义外部符号后，您可以按照符号的名称来引用类。您也可以使用符号来引用类的静态方法和字段。

示例

```
<externalClassSymbol>
  <name>$rand</name>
  <className>user.class.Random</className>
</externalClassSymbol>
```

此符号引用了用户编写的类，生成随机数。必须在 <runtimeConfiguration> 标记中的 <requestMapperClassPath> 标记中提供包含此类的 JAR 文件的完整路径和名称。

要在表达式中引用静态方法 `user.class.Random.generate()`，您可以使用外部符号：

```
$rand.generate()
```

将值映射到上下文密钥

在 <requestMapperDefinition> 标记内，通过使用 <selection> 标记将值映射到上下文密钥。此映射提供监视信息中的更改。

您可以将值映射到针对此请求类型定义的输出密钥。有关更多信息，请参阅第 776 页的表 233。

如果在对请求映射器配置进行求值后未将任何值映射到密钥，ITCAM 将使用从请求抽取的原始值。

在 <selection> 标记中，使用以下标记。

<matchCriteria>
必须返回布尔值的表达式。仅当此表达式返回 `true` 时，才使用在此标题中定义的映射。

<mapTo>
定义密钥以及要映射到该密钥的值。在此标记中，<key> 标记包含密钥，<value> 标记包含值。

<selection>
您可以嵌套 <selection> 标记，将一个标记放入另一个标记中。

如果嵌套 <selection> 标记，那么仅当外部和嵌套 <matchCriteria> 表达式返回 `true` 时，才会使用嵌套映射。

您可以在 <requestMapperDefinition> 标记或另一个 <selection> 标记中使用多个 <selection> 标记。如果在同一嵌套级别上（即同一父标记中）的多个 <selection> 标记中多次映射同一密钥，那么使用 <matchCriteria> 表达式对其返回 `true` 的第一个映射。

不要在外部和嵌套 <selection> 标记中映射同一密钥。

通常，使用 <matchCriteria> 值 `true` 作为处于嵌套级别的最后一个选择标记的“else”值。如果要在不同情况下映射不同值，请在此外部标记中使用多个 <selection> 标记，每个标记都可以包含用于特殊情况的条件和值。具有值 `true` 的最后一个标记包括可用数据不满足任何条件的情况。

示例

```
<selection>
   <matchCriteria>true</matchCriteria>
   <mapTo>
      <key>Result</key>
      <value>$s1</value>
   </mapTo>
</selection>
```

在此映射配置中，`Result` 设置为符号 `$s1` 的值。

```
<matchCriteria>true</matchCriteria>
   <selection>
      <matchCriteria>$b1</matchCriteria>
      <mapTo>
         <key>Result</key>
         <value>1</value>
      </mapTo>
   </selection>
   <selection>
      <matchCriteria>true</matchCriteria>
      <mapTo>
         <key>Result</key>
         <value>2</value>
      </mapTo>
</selection>
```

在此映射配置中，符号 `$b1` 必须返回布尔值。如果 `$b1` 返回 `true`，`Result` 设置为 `1`，如果 `$b1` 返回 `false`，则设置为 `2`。如果 `$b1` 返回 `true`，请求映射器使用第一个 <selection> 标记中 `Result` 的映射；而不使用第二个标记中同一密钥的映射。

定义定制请求

缺省情况下，仅特定类型的 Java 类和方法由数据收集器作为请求进行监视。servlet、JSP、EJB 业务方法和特定的标准 Java EE API 将识别为请求。您可以将额外的类和方法指定为定制请求。

关于此任务

要按数据收集器启用定制请求监视，请在 XML 文件中定义定制请求，并在工具箱定制属性文件中设置 `am.camtoolkit.gpe.customxml.custom` 属性。

例如，缺省情况下，数据收集器不会将“Struts 操作”类识别为请求。但是，您可以设置定制请求定义，并使操作可识别为“嵌套的请求”。

过程

完成以下过程以启用定制请求的监视，并将一个或多个方法指定为定制请求：

1. 在临时位置中生成 *dc_home*/itcamdc/etc/custom_requests.xml 文件的副本。然后，在文本编辑器中打开此副本。
2. 修改文件中的参数。

   下表描述了您可修改的参数：

表 *232.* 定制请求配置文件的参数

标记名称	描述
edgeRequest	标识一个或多个应用程序方法，将对这些方法进行字节代码检测以用于定制请求处理。通过修改 edgeRequest 标记内的 requestName、Matches、type 和 methodName 标记，您可以定制所选内容。 每个 edgeRequest 标记都必须正好包含一个 methodName 标记以及一个或多个 Matches 标记。可以指定多个 edgeRequest 标记。
requestName	定义该请求的唯一名称。跟踪方法入口和出口时，会向用户显示请求名称。
Matches	标识包含特定方法的一个或多个类，将对这些方法进行字节代码检测以用于定制请求处理。一个 edgeRequest 标记内可能有多个 Matches 标记。
type	指示类是否必须为系统或应用程序类才能与 edgeRequest 标记匹配。
methodName	标识一些方法的名称，这些方法位于由 Matches 标记标识的要进行字节代码检测（以用于定制请求处理）的某个类中。每个 edgeRequest 标记内只能指定正好一个 methodName 标记。
requestMapper	可选。如果指定此标记，那么数据收集器会使用请求映射器来确定标识此请求的信息。您可以定义抽取此信息的非标准方式。有关启用和定义请求映射器的更多信息，请参阅第 765 页的『定制请求信息映射』。
切记: Matches 和 methodName 标记可以包含通配符。通配符的使用方式如下所述： · 星号 (*) 单独使用时，代表出现 0 次或更多次数的任何字符。当嵌入字符序列（例如， java.*.String）中时，它匹配出现 0 次或更多次数的、除程序包分隔符 (.) 之外的任意字符. · 可使用两个句点 (..) 来指定所有子软件包。它匹配以程序包分隔符 (.) 开头和结束的任何顺序的字符. 例如，java..String 与 java.lang.String 匹配，com.ibm..* 与以 com.ibm 开头的任何声明匹配。	

例如，程序包名为 com.mycompany.myapp 的应用程序具有以下需求：

· 在 Customer 类中，必须将 creditCheck() 方法视为名为 CreditCheck 的定制请求。

· 在 Supplier 类中，必须将 inventoryCheck() 方法视为名为 SupplyCheck 的定制请求。

实现这些需求的定制 custom_requests.xml 文件的内容如下所示：

```
<customEdgeRequests>
      <edgeRequest>
              <requestName>CreditCheck</requestName>
              <Matches>com.mycompany.myapp.Customer</Matches>
              <type>application</type>
              <methodName>creditCheck</methodName>
      </edgeRequest>
      <edgeRequest>
              <requestName>SupplyCheck</requestName>
              <Matches>com.mycompany.myapp.Supplier</Matches>
              <type>application</type>
              <methodName>inventoryCheck</methodName>
      </edgeRequest>
</customEdgeRequests>
```

3. 完成下列其中一个步骤：

- 将此文件保存在 *dc_home*/runtime/*app_server_version.node_name.profile_name.server_name*/custom 目录中。然后，在工具箱定制属性文件中，将 am.camtoolkit.gpe.customxml.custom 属性设置为您在步骤 第 773 页的『2』 中修改的文件的名称（不含路径）。
- 将文件保存到计算机上的任何目录。然后，在工具箱定制属性文件中，将 am.camtoolkit.gpe.customxml.custom 属性设置为您在步骤 第 773 页的『2』 中修改的文件的路径和名称。

有关工具箱定制属性文件的更多信息，请参阅第 758 页的『工具箱属性文件』。

启用请求映射器

要针对一种请求类型启用请求映射器，请编辑工具箱定制配置文件或工具箱全局定制配置文件。对于常见请求类型以及对于定制请求，过程将会不同。

开始之前

在 XML 文件中定义请求映射器配置。然后，将包含请求映射器配置的 XML 文件放在与工具箱属性文件相同的目录中。

- 要针对所有应用程序服务器实例启用请求映射器，请将其放在 *dc_home*/runtime/custom 目录中。
- 要针对一个应用程序服务器实例启用请求映射器，请将其放在 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/ 目录中。

有关 XML 文件语法的信息，请参阅 第 766 页的『XML 文件语法』。

关于此任务

编辑 toolkit_custom.properties 文件或 toolkit_global_custom.properties 文件以针对一个或所有应用程序服务器实例启用请求映射器。

过程

- 要针对常见请求启用请求映射器，请完成以下步骤：
  a) 在文本编辑器中，打开以下某个工具箱定制配置文件：
     - 要针对所有应用程序服务器实例启用请求映射器，请打开 *dc_home*/runtime/custom/toolkit_global_custom.properties 文件。
     - 要针对一个应用程序服务器实例启用请求映射器，请打开 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties 文件。
  b) 编辑工具箱属性文件，如下所示：
     - 添加可将此请求类型的启用属性设置为 true 的一行。有关更多信息，请参阅第 776 页的表 233。
     - 添加可将 am.camtoolkit.gpe.customxml.* 属性设置为映射器 XML 文件名称的一行。使用任何唯一值，而不是 * 符号。有关更多信息，请参阅第 766 页的『XML 文件语法』。
  c) 保存并关闭属性文件。

  示例：

  要针对 SQL 请求类型启用在 renameDataSource.xml 中定义的请求映射器，请将以下行添加到工具箱定制配置文件或工具箱全局定制配置文件：

  ```
  com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
  am.camtoolkit.gpe.customxml.renameDataSource=renameDataSource.xml
  ```

- 要针对定制请求启用请求映射器，请完成以下步骤：
  a) 在定制请求定义 XML 文件中的 <edgerequest> 标记下，创建 <requestMapper> 标记。在此标记中放置唯一的请求映射类型名称。有关定义定制请求的信息，请参阅第 772 页的『定义定制请求』。

b) 在请求映射器 XML 文件中 <requestMapperDefinition> 标记的 type 属性中使用唯一的请求映射类型名称。

c) 在文本编辑器中，打开以下某个工具箱定制配置文件：

- 要针对所有应用程序服务器实例启用请求映射器，请打开 *dc_home*/runtime/custom/toolkit_global_custom.properties 文件。
- 要针对一个应用程序服务器实例启用请求映射器，请打开 *dc_home*/runtime/*appserver_version.node_name.profile_name.server_name*/custom/toolkit_custom.properties 文件。

d) 编辑工具箱属性文件以添加可将 am.camtoolkit.gpe.customxml.* 属性设置为映射器 XML 文件名称的一行。使用任何唯一值，而不是 * 符号。有关更多信息，请参阅第 766 页的『XML 文件语法』。

e) 保存并关闭属性文件。

示例：

要针对 custom_requests.xml 文件中所定义的 SupplyCheck 定制请求类型启用在 customMapper.xml 中定义的请求映射器，请完成以下步骤：

1. 在 custom_requests.xml 文件中使用以下定义：

```
<customEdgeRequests>
  <edgeRequest>
    <requestName>SupplyCheck</requestName>
    <Matches>com.mycompany.myapp.Supplier</Matches>
    <type>application</type
    <methodName>inventoryCheck</methodName>
    <requestMapper>customMapper</requestMapper>
  </edgeRequest>
</customEdgeRequests>
```

2. 在 customMapper.xml 文件中，确保设置了类型名称：

```
<requestMapperDefinition type="customMapper">
```

3. 将以下行添加到工具箱定制配置文件或工具箱全局定制配置文件中：

```
am.camtoolkit.gpe.customxml.customMapper=customMapper.xml
```

请求映射器类型名称、输入和输出数据

下表列出了针对不同请求类型配置和启用请求映射器所必需的信息。

每个表头的含义解释如下所示：

请求类型
请求类型。

启用属性
要启用请求映射器，请在 toolkit_custom.properties 或 toolkit_global_custom.properties 文件中将此属性设置为 true。

要点： 如果从表中复制该值，请移除任何空格和换行符。

请求映射器类型名称
在请求映射器定义 XML 文件中将此值分配给 <requestMapperDefinition> 标记的 type 属性。

输入数据符号名称
表示请求信息的符号。您可以在请求映射器定义中的表达式中使用这些符号。有关更多信息，请参阅第 766 页的『定义表达式』。

输出数据上下文密钥
要提供监视信息中的更改，请为请求映射器定义中的这些密钥指定值。有关更多信息，请参阅第 771 页的『将值映射到上下文密钥』。

表 233. 请求映射器启用属性和类型名称

请求类型	启用属性	请求映射器类型名称
servlet	com.ibm.tivoli.itcam.toolkit.ai.enable.servletrequestmapper	servlet
JNDI	com.ibm.tivoli.itcam.toolkit.ai.enable.jndirequestmapper	jndiLookup
定制请求		由用户在 edgeRequest 定义中进行定义
EJB	com.ibm.tivoli.itcam.toolkit.ai.enable.ejbrequestmapper	ejb
JCA	com.ibm.tivoli.itcam.toolkit.ai.enable.jcarequestmapper	jca
JDBC 数据源	com.ibm.tivoli.itcam.toolkit.ai.enable.datasourcerequestmapper	dataSource
JDBC SQL 语句	com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper	sqlStatement
JMS	com.ibm.tivoli.itcam.toolkit.ai.enable.jmsrequestmapper	jms
JAX-RPC Web Service	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
轴 Web service	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
MQI	com.ibm.tivoli.itcam.toolkit.ai.enable.mqrequestmapper	mqi
EJB	com.ibm.tivoli.itcam.toolkit.ai.enable.ejbrequestmapper	ejb
JDBC 连接工厂	com.ibm.tivoli.itcam.toolkit.ai.enable.sqlconnectfactoryrequestmapper	connectionFactory
SCA	com.ibm.tivoli.itcam.toolkit.ai.enable.scarequestmapper	sca
JAX-WS Web Service	com.ibm.tivoli.itcam.toolkit.ai.enable.webservicerequestmapper	webServices
WebSphere Portal Server 门户网站（扩展 org.apache.jetspeed.portlet.Portlet 类）	com.ibm.tivoli.itcam.toolkit.ai.enable.portalrequestmapper	portalPortal

表 233. 请求映射器启用属性和类型名称 (续)		
请求类型	**启用属性**	**请求映射器类型名称**
WebSphere Portal Server V6.1、V7 和 8 门户网站（实现 `javax.portlet.Portlet` 接口）	`com.ibm.tivoli.itcam.toolkit.ai.enable.portal6requestmapper`	`Portal6Portal`

要点: 针对第 776 页的表 233 中未列出的请求类型，还没有有意义的方式来配置定制请求映射器。

表 234. 请求映射器输入和输出数据		
请求类型	**输入数据符号名称**	**输出数据上下文密钥**
servlet	有关更多信息，请参阅第 780 页的表 235。	**remappedURI** 定义已重命名的 URI **remappedURL** 定义已重命名的 URL **appName** 定义已重命名的应用程序名称 **userid** 定义请求的用户标识
JNDI	· **$jndiContext** 上下文对象 · **$lookup** 查找字符串 · **$context**: “JNDIlookup”	**renamedLookup** 定义已重命名的查找字符串
定制请求	· **$this** 定制请求方法的“this”对象 · **$0** 传递到定制请求方法的参数，指定为对象的数组 · **$className** 定制请求类名 · **$methodName** 定制请求方法名称 · **$context** edgeRequest 定义中的原始请求名称	**customRequestName** 定义已重命名的定制请求名称
EJB	· **$ejb** EJB 实现对象 · **$appName** 应用程序的名称 · **$ejbType** EJB 的类型 · **$className** EJB 实现对象的类名 · **$methodName** EJB 业务方法名称 · **$context**: “EJBBusinessMethod”	**appName** 定义已重命名的应用程序名称 **ejbType** 定义已重命名的 EJB 类型 **className** 定义已重命名的类名 **methodName** 定义已重命名的方法名称
JCA	· **$interaction** 交互对象 · **$interactionSpec** InteractionSpec 对象 · **$record** 记录对象 · **$context**: “J2Cexecute”	**lookupName** 是已重命名的 lookupName **productName** 是已重命名的产品名称 **productVersion** 是已重命名的产品版本

表 *234.* 请求映射器输入和输出数据 *(续)*

请求类型	输入数据符号名称	输出数据上下文密钥
JDBC 数据源	· **$this** 数据源或驱动程序对象 · **$dataSource** $this 对象，作为数据源 · **$driver** $this 对象，作为驱动程序 · **$dataSourceName** 是数据源的名称，如 java.lang.String · **$context** 指示请求的类型，“JDBCgetConnection”或“JDBCgetConnection FromDriver”	**dataSourceName** 是已重命名的数据源名称，前提是 $this 对象为数据源 **url** 是已重命名的驱动程序 URL，前提是 $this 对象为驱动程序
JDBC SQL 语句	· **$this** SQL 语句或 SQL 连接 · **$sqlText** 包含类似于 java.lang.String 的 SQL 文本，前提是 $this 对象为 SQL 语句 · **$sqlStatement** $this 对象，作为 SQL 语句 · **$sqlConnection** $this 对象，作为 SQL 连接 · **$dataSourceName** 数据源名称 · **$context** 指示请求的类型：“JDBCexecute”、“JDBCexecuteQuery”、“JDBCexecuteUpdate”、“JDBCcreateStatement”、“JDBCprepareStatement”	**dataSourceName** 是已重命名的数据源名称 **sqlText** 是已重命名的 SQL 文本
JMS	· **$this** 受检测方法的“this”对象。可以是 QueueBrowser、MessageConsumer、 MessageProducer 或 MessageListener · **$0** 队列对象（用于发送请求）或主题对象（用于发布请求） · **$queueBrowser** $this 对象，作为 QueueBrowser · **$messageConsumer** $this 对象，作为 MessageConsumer · **$messageProducer** $this 对象，作为 MessageProducer · **$messageListener** $this 对象，作为 MessageListener · **$queue** $0 对象，作为队列 · **$topic** $0 对象，作为主题 · **$context** 指示请求的类型：“JMSreceive”、“JMSsend”、“JMSbrowse”、“JMSpublish”、“JMSonmessage”	**queueName** 已重命名的队列名称 **providerURL** 已重命名的提供者 URL **topicName** 已重命名的主题名称

表 *234.* 请求映射器输入和输出数据 *(续)*

请求类型	输入数据符号名称	输出数据上下文密钥
JAX-RPC Web Service	· **$messageContext** IMessageContextWrapper · **$appName** 应用程序名称 · **$requestName** 缺省请求名称 · **$url** URL · **$context** 指示请求的类型：“WebServicesJaxRpc ProviderRequest”、“WebServicesJaxRpc ClientRequest”	**appName** 已重命名的应用程序名称 **requestName** 已重命名的请求名称 **url** 已重命名的 URL
轴 Web service	· **$messageContext** IMessageContextWrapper · **$appName** 应用程序名称 · **$requestName** 缺省请求名称 · **$url** URL · **$context** 指示请求的类型：“WebServicesAxisClient Request”、“WebServicesAxis ProviderRequest”	**appName** 已重命名的应用程序名称 **requestName** 已重命名的请求名称 **url** 已重命名的 URL
MQI	· **$queue** MQQueue 对象，前提是已知 · **$qmgr** MQQueueManager 对象，前提是已知 · **$message** MQMessage 或 MQMsg2 对象，前提是已知 · **$session** MQSESSION 对象，前提是已知 · **$getMsgOptions** MQGetMessageOptions 对象，前提是已知 · **$qmgrName** 队列管理器的名称 · **$queueName** 队列的名称 · **$context** MQ 请求的类型：“MQCONN”、“MQCONNX”、“MQDISC”、“MQBACK”、“MQBEGIN”、“MQCLOSE”、“MQCMIT”、“MQINQ”、“MQOPEN”、“MQSET”、“MQGET”、“MQPUT”、“MQPUT1”、“MQGETBROWSE”	**qmgrName** 已重命名的队列管理器名称 **qname** 已重命名的队列名称
EJB	· **$appName** 应用程序的名称 · **$ejbType** EJB 的类型 · **$className** EJB 实现对象的类名 · **$methodName** EJB 业务方法名称 · **$context**：“EJBBusinessMethod”	**appName** 定义已重命名的应用程序名称 **ejbType** 定义已重命名的 EJB 类型 **className** 定义已重命名的类名 **methodName** 定义已重命名的方法名称

表 *234.* 请求映射器输入和输出数据 *(续)*

请求类型	输入数据符号名称	输出数据上下文密钥
JDBC 连接工厂	· **$connectionFactory** 连接工厂 · **$dataSourceName** 数据源名称 · **$context**：“JDBCgetConnection”	**dataSourceName** 是已重命名的数据源名称
SCA	· **$uri** URI · **$operationName** 操作名称 · **$context** 指示请求的类型：“SCA_Generic”、“SCA_Ref”、“SCA_Target”	**uri** 是已重命名的 URI **operationName** 是已重命名的操作名称
JAX-WS Web Service	· **$messageContext** IMessageContextWrapper · **$appName** 应用程序名称 · **$requestName** 缺省请求名称 · **$url** URL · **$context** 指示请求的类型：“WebServicesJAXWS ClientRequest”、“WebServicesJAXWS ProviderRequest”、“WebServicesJAXWS AsyncRequest”	**appName** 已重命名的应用程序名称 **requestName** 已重命名的请求名称 **url** 已重命名的 URL
WebSphere Portal Server 门户网站（扩展 `org.apache.jetspeed.portlet.Portlet` 类）	· **$portletAdapter** PortletAdapter · **$portletRequest** PortletRequest · **$portletResponse** PortletResponse · **$portletName** portlet 名称 · **$pageTitle** 页面标题 · **$url** 请求的 URL · **$userid** 请求用户标识 · **$context**：“Portal.Portlet”	**portletName** 已重命名的 portlet 名称 **title** 已重命名的页面标题 **url** 已重命名的 URL **userid** 已重命名的用户标识
WebSphere Portal Server V6.1、V7 和 V8 门户网站（实现 `javax.portlet.Portlet` 接口）	· **$portlet** portlet · **$renderRequest** RenderRequest · **$renderResponse** RenderResponse · **$portletName** portlet 名称 · **$pageTitle** 页面标题 · **$url** 请求的 URL · **$userid** 请求用户标识 · **$context**：“Portal.Portlet”	**portletName** 已重命名的 portlet 名称 **title** 已重命名的页面标题 **url** 已重命名的 URL **userid** 已重命名的用户标识

对于 servlet 请求，提供了大量输入数据符号。

表 *235. servlet* 请求的输入数据符号名称

符号名称	值类型	符号内容
$context	String	“ServletMethod”

表 *235. servlet* 请求的输入数据符号名称 *(续)*

符号名称	值类型	符号内容
$servlet	javax.servlet.http.HttpServlet	与 servlet 请求关联的 HttpServlet 对象
$httpServletRequest	javax.servlet.http.HttpServletRequest	与 servlet 请求关联的 HttpServletRequest 对象
$httpServletResponse	javax.servlet.http.HttpServletResponse	与 servlet 请求关联的 HttpServletResponse 对象
$appName	java.lang.String	与 servlet 关联的应用程序名称
$URL	java.lang.StringBuffer	客户机用于提出请求的 URL
$RemoteUser	java.lang.String	提出此请求的用户的登录名，前提是已认证
$URI	java.lang.String	从协议名称到查询字符串的请求 URL 的部分
$ServletPath	java.lang.String	调用 servlet 的请求 URL 的部分。
$SessionID	javax.servlet.http.HttpSession	与此请求关联的当前会话
$QueryString	java.lang.String	请求 URL 中包含的位于路径之后的查询字符串。
$SessionAttribute	java.lang.String	此参数化符号返回会话属性值。它具有一个参数，属性名（必须为字符串）。 例如，`$SessionAttribute("attr1")` 返回名为 attr1 的属性的值。
$cookie	javax.servlet.http.Cookie	此参数化符号返回指定的 cookie。它具有一个参数，即 cookie 名称（必须为字符串）。 例如，`$cookie("cookie1")` 返回名为 cookie1 的属性的值。

示例请求映射器定义
以下示例说明了请求映射器功能的使用情况。

更改 *servlet* 应用程序名称
在此示例中，servlet 请求中的应用程序名称被 URI 和查询字符串替换。

dc_home`/runtime/changeAppname.xml` 文件包含以下请求映射器定义：

```
<gpe>
   <runtimeConfiguration>
         <requestMapperDefinition type="servlet">
               <selection>
                     <matchCriteria>true</matchCriteria>
                     <mapTo>
                           <key>appName</key>
                           <value>$URI +  “.”  + $QueryString</value>
                     </mapTo>
               </selection>
         </requestMapperDefinition>
   </runtimeConfiguration>
</gpe>
```

重命名数据源
在此示例中，SQL 请求中的数据源名称更改为用户可以更容易理解的版本。

dc_home/runtime/renameDataSource.xml 文件包含以下请求映射器定义：

```
<gpe>
 <runtimeConfiguration>
   <requestMapperDefinition type=” sqlStatement” >
      <selection>
         <matchCriteria>$dataSourceName != null</matchCriteria>
         <selection>
           <matchCriteria>$dataSourceName.equals("jdbc/TradeDataSource")
</matchCriteria>
           <mapTo>
             <key>dataSourceName</key>
             <value>"Daytrader Data Source"</value>
           </mapTo>
         </selection>
         <selection>
           <matchCriteria>$dataSourceName.equals("jdbc/LongDataSource")
</matchCriteria>
           <mapTo>
             <key>dataSourceName</key>
             <value>"Long term trader Data Source"</value>
           </mapTo>
         </selection>
      </selection>
   </requestMapperDefinition>
 </runtimeConfiguration>
<gpe>
```

第一个 <selection> 标记确保 $dataSourceName 不为空。第二个 <selection> 标记可以对 $dataSourceName.equals() 安全求值。

如果第一个 <selection> 标记不存在，并且传递了空的 $dataSourceName，那么请求映射器将生成异常。此异常可能会导致缺少监视信息。

要启用此请求映射器，文件 *dc_home*/runtime/toolkit_global_custom.properties 包含以下行：

```
com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
am.camtoolkit.gpe.customxml.renameDataSource=renameDataSource.xml
```

从 *SQL* 请求中移除敏感信息
在此示例中，应用程序在 SQL 请求中包含社会保险号。请求映射器从用户可以看到的请求版本中移除编号。

在 SQL 请求中，列出了社会保险号以及 SS 列名 SS = *number*。请求映射器查找 SS= 字符串并移除后面跟着的九个符号。

dc_home/runtime/removeSSN.xml 文件包含以下请求映射器定义：

```
<gpe>
    <runtimeConfiguration>
         <requestMapperDefinition type="sqlStatement">
             <symbolDefinitions>
                 <symbol>
                     <name>$offsetOfSS</name>
                     <eval>$sqlText.indexOf("SS = ")</eval>
                 </symbol>
                 <symbol>
                     <name>$sqlTextContainsSS</name>
<eval>$sqlText != null AND $offsetOfSS > 0 AND $sqlText.length() GE
$offsetOfSS+16</eval>
                 </symbol>
                 <conditionalSymbol>
                     <name>$sqlTextPriorToSSKeyword</name>
                     <type>java.lang.String</type>
                     <defaultValue>""</defaultValue>
                     <if condition="$sqlTextContainsSS">
                       <return>$sqlText.substring(0, $offsetOfSS+5)</return>
                     </if>
                 </conditionalSymbol>
                 <conditionalSymbol>
                     <name>$sqlTextAfterSS</name>
                     <type>java.lang.String</type>
                     <defaultValue>""</defaultValue>
```

```
                    <if condition="$sqlTextContainsSS">
                      <return>$sqlText.substring($offsetOfSS+16)</return>
                    </if>
                  </conditionalSymbol>
              </symbolDefinitions>
              <selection>
                   <matchCriteria>$sqlText != null AND $sqlText.length() >
0</matchCriteria>
                   <selection>
                        <matchCriteria>$sqlTextContainsSS</matchCriteria>
                        <mapTo>
                             <key>sqlText</key>
                             <value>$sqlTextPriorToSSKeyword + "?" +
$sqlTextAfterSS</value>
                        </mapTo>
                   </selection>
              </selection>
         </requestMapperDefinition>
    </runtimeConfiguration>
</gpe>
```

要启用此请求映射器，文件 *dc_home*/runtime/toolkit_global_custom.properties 包含以下行:

```
com.ibm.tivoli.itcam.toolkit.ai.enable.sqlrequestmapper=true
am.camtoolkit.gpe.customxml.renameDataSource=removeSSN.xml
```

配置 WebSphere Applications 代理程序以监视 WebSphere Extreme Scale

安装 WebSphere Applications 代理程序后，可以执行额外配置以在独立环境或 WebSphere Application Server 环境中监视 WebSphere Extreme Scale (WXS)。

关于此任务

根据 WebSphere Extreme Scale 的安装方式以及是否启用安全性，配置步骤不尽相同。运行配置过程之前，请执行以下步骤。

过程

1. 确认 WebSphere Extreme Scale 的安装方式。

   独立方式
   WebSphere Extreme Scale 安装在不含 WebSphere Application Server 的环境中。

   嵌入式 WAS 方式
   WebSphere Extreme Scale 安装在 WebSphere Application Server 环境中。
2. 确认针对 WebSphere Extreme Scale 是否启用安全性。如果在嵌入式 WebSphere Application Server 方式下使用了安全的 Java™ 客户机，必须完成安全性连接步骤。
3. 单击链接以获取指示信息。
   - 要在独立环境中配置 WebSphere Extreme Scale，请单击第 783 页的『在独立环境中配置 WebSphere Extreme Scale 监视』。
   - 要在不含安全性的嵌入式环境中配置 WebSphere Extreme Scale，请单击第 785 页的『在未启用安全性的 WebSphere 环境中配置 WebSphere Extreme Scale 监视』。
   - 要在启用安全性的嵌入式环境中配置 WebSphere Extreme Scale，请单击第 786 页的『在启用安全性的 WebSphere 环境中配置 WebSphere Extreme Scale 监视』。

在独立环境中配置 WebSphere Extreme Scale 监视

了解如何在不含 WebSphere Application Server 的环境中安装 WebSphere Extreme Scale 时配置 WebSphere Applications 代理程序。

过程

1. 停止 WebSphere Applications 代理程序。
   a) 转至安装 WebSphere Applications 代理程序的 *install_dir* 目录。
   b) 运行命令 bin/was-agent.sh stop。

2. 运行配置脚本。

```
install_dir/platform_code/yn/bin/wxs-agent-config.sh config
```

其中

- *install_dir* 是 WebSphere Applications 代理程序的安装目录。
- *platform_code* 是安装代理程序的平台代码，例如，lx8266 表示 Linux x86_64 R2.6（64 位），aix536 表示 AIX R5.3（64 位）。

示例命令：

```
/opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh config
```

```
/opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh config
```

3. 出现输入代理程序安装路径提示时，请指定 WebSphere Applications 代理程序的主目录。

   注: 此脚本会根据您指定的安装路径查找配置文件名称。缺省路径为 *install_dir*/config/${hostname}_yn.xml。如果提示此文件不存在，原因可能是执行配置时未启动 WebSphere Applications 代理程序。请启动 WebSphere Applications 代理程序，然后将其至少停止一次。
4. 出现 `WebSphere Extreme Scale Catalog Server connector type` 提示时，请按 1 继续。
5. 出现 `Input a node name to identify this agent node on UI` 提示时，请输入节点名称。

   节点名称用于识别受监视的 WebSphere Extreme Scale 区域，它显示在实例名称中，您可在“应用程序性能仪表板”UI 中看到此实例名称。
6. 出现 `WebSphere Extreme Scale Catalog Server security enabled?` 提示时，如果已启用安全性，请输入 1。然后，输入用户名和密码。如果未启用安全性，请输入 2。
7. 指定目录服务器的主机名和端口号。如果存在多个目录服务器，可以逐一添加。您还可以逐一添加多个区域。
   - 主机名即目录服务器所在位置的系统名称。请确保主机名可供访问。如果无法访问，请使用 IP 地址作为主机名。
   - 端口号即 WebSphere Extreme Scale 目录服务器的 **JMXServicePort** 编号。缺省值为 1099。在 WebSphere Extreme Scale Knowledge Center 中可找到有关端口号的更多详细信息。
8. 要启动代理程序，请运行以下命令。

```
install_dir/bin/was-agent.sh start
```

   注:
   - 代理程序配置存储在 *install_dir*/config/${hostname}_yn.xml 中。如果您要更改任意配置，请重新运行此脚本，或者直接修改 .xml 文件。
   - 先前配置将备份为 *install_dir*/config/${hostname}_yn.xml.bak。如果需要，您可复原先前配置。
   - 运行 *install_dir*/*platform_code*/yn/bin/wxs-agent-config.sh config 时，可按 Ctrl-C 退出脚本。现有配置不会发生更改。

在嵌入式 WebSphere 环境中配置 WebSphere Extreme Scale 监视

了解当 WebSphere Extreme Scale 安装在 WebSphere Application Server 环境中时，如何配置 WebSphere Applications 代理程序。

关于此任务

如果未针对 WebSphere Extreme Scale 服务器启用安全性，可以直接运行配置过程。否则，必须首先完成第 786 页的『在启用安全性的 WebSphere 环境中配置 WebSphere Extreme Scale 监视』。

在未启用安全性的 *WebSphere* 环境中配置 *WebSphere Extreme Scale* 监视

如果在未启用安全性的 WebSphere Application Server 环境中安装 WebSphere Extreme Scale，那么可以直接配置 WebSphere Applications 代理程序。

过程

1. 停止 WebSphere Applications 代理程序。
   a) 转至安装 WebSphere Applications 代理程序的 *install_dir* 目录。
   b) 运行命令 `bin/was-agent.sh stop`。
2. 运行配置脚本。

   ```
   install_dir/platform_code/yn/bin/wxs-agent-config.sh config
   ```

   其中

   · *install_dir* 是 WebSphere Applications 代理程序的安装目录。
   · *platform_code* 是安装代理程序的平台代码，例如，lx8266 表示 Linux x86_64 R2.6（64 位），aix536 表示 AIX R5.3（64 位）。

   示例命令：

   ```
   /opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh
   ```

   ```
   /opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh
   ```

3. 出现输入代理程序安装路径提示时，请指定 WebSphere Applications 代理程序的主目录。

   注: 此脚本会根据您指定的安装路径查找配置文件名称。缺省路径为 *install_dir*`/config/${hostname}_yn.xml`。如果提示此文件不存在，原因可能是执行配置时未启动 WebSphere Applications 代理程序。请启动 WebSphere Applications 代理程序，然后将其至少停止一次。
4. 出现 `WebSphere Extreme Scale Catalog Server connector type` 提示时，请按 2 继续。
5. 出现 `Input a node name to identify this agent node on UI` 提示时，请输入节点名称。

   节点名称用于识别受监视的 WebSphere Extreme Scale 区域，它显示在实例名称中，您可在“应用程序性能仪表板”UI 中看到此实例名称。
6. 出现 `WebSphere Extreme Scale Catalog Server security enabled?` 提示时，请按 2 继续。
7. 指定目录服务器的主机名和端口号。如果存在多个目录服务器，可以逐一添加。您还可以逐一添加多个区域。

   · 主机名即目录服务器所在位置的系统名称。请确保主机名可供访问。如果无法访问，请使用 IP 地址作为主机名。
   · 端口号用于指示 WebSphere Extreme Scale 目录服务器的 **JMXServicePort** 编号。它继承自每个 WebSphere Application Server 的 **BOOTSTRAP_ADDRESS** 值。在 WebSphere Extreme Scale Knowledge Center 中可找到有关端口号的更多详细信息。
8. 要启动代理程序，请运行以下命令。

   ```
   install_dir/bin/was-agent.sh start
   ```

   注:

   · 代理程序配置存储在 *install_dir*`/config/${hostname}_yn.xml` 中。如果您要更改任意配置，请重新运行此脚本，或者直接修改 `.xml` 文件。
   · 先前配置将备份为 *install_dir*`/config/${hostname}_yn.xml.bak`。如果需要，您可复原先前配置。
   · 运行 *install_dir*`/`*platform_code*`/yn/bin/wxs-agent-config.sh config` 时，可按 Ctrl-C 退出脚本。现有配置不会发生更改。

在启用安全性的 *WebSphere* 环境中配置 *WebSphere Extreme Scale* 监视

如果在启用安全性的 WebSphere Application Server 环境中安装 WebSphere Extreme Scale，必须先完成初始设置步骤，然后才能配置 WebSphere Applications 代理程序。

关于此任务

如果要在启用安全性的 WebSphere Application Server 环境中监视 WebSphere Extreme Scale 服务器，需要手动配置安全设置。

此过程适用于以下情况：

- WebSphere Extreme Scale 服务器必须部署在 WebSphere Application Server 应用程序服务器、节点代理程序或 DMGR 进程中。
- WebSphere Applications 代理程序必须部署在运行 WebSphere Extreme Scale 区域目录服务的节点上。在此节点下配置代理程序用于 WebSphere Extreme Scale 监视，并将其设置为连接到此目录服务实例。
- 每个代理程序实例只能用于监视一个 WebSphere Extreme Scale 区域。

过程

1. 如果 WebSphere Application Server 的 JDK 版本低于 1.7，必须重新配置 WebSphere Applications 代理程序，以使用与 WebSphere Application Server 相同的 JRE。
   a) 打开 *install_dir*/config/.yn.environment 文件。
   b) 将以下值添加到第一行。

   ```
   #JAVAHOME=/opt/IBM/WebSphere/AppServer/java/8.0/jre
   ```

2. 配置 WebSphere Applications 代理程序安全性属性文件。
   有关指示信息，请参阅第 786 页的『配置代理程序以处理 WebSphere Application Server 的 JAR 文件和安全属性』。
3. 可选： 如果使用了安全的 Java™ 客户机，那么必须确保正确配置认证。您必须编辑客户机属性文件和 SSL 属性文件。有关指示信息，请参阅第 787 页的『设置连接凭证』。
   注: 如果未使用 SSL 设置来保护密钥，只需输入密码和用户名，然后即可跳过此步骤。
4. 运行配置脚本以启动配置控制台。请参阅第 789 页的『运行配置』。

配置代理程序以处理 *WebSphere Application Server* 的 *JAR* 文件和安全属性

配置 WebSphere Applications 代理程序以处理 WebSphere Application Server JAR 文件和安全属性。

关于此任务

要执行此配置，请编辑 kynwb.properties 文件。

过程

1. 打开 *install_dir*/*platform_code*/yn/config/kynwb.properties 文件。如果此文件不存在，请创建此文件。
   - *install_dir* 是 WebSphere Applications 代理程序的安装目录。
   - *platform_code* 是安装代理程序的平台代码，例如，lx8266 表示 Linux x86_64 R2.6（64 位），aix536 表示 AIX R5.3（64 位）。
2. 在文件开头列出类路径。在现有行之前添加以下行。
   - 对于 WebSphere Application Server 9.0：

   ```
   appserver_home/plugins/com.ibm.ws.runtime.jar:\
   appserver_home/lib/bootstrap.jar:\
   appserver_home/runtimes/com.ibm.ws.admin.client_9.0.jar:\
   appserver_home/lib/wsogclient.jar:\
   ```

· 对于 WebSphere Application Server 8.5：

```
appserver_home/plugins/com.ibm.ws.runtime.jar:\
appserver_home/lib/bootstrap.jar:\
appserver_home/runtimes/com.ibm.ws.admin.client_8.5.0.jar:\
appserver_home/lib/wsogclient.jar:\
```

已改的类路径示例：

```
/opt/IBM/WebSphere/plugins/com.ibm.ws.runtime.jar:\
/opt/IBM/WebSphere/lib/bootstrap.jar:\
/opt/IBM/WebSphere/runtimes/com.ibm.ws.admin.client_8.5.0.jar:\
/opt/IBM/WebSphere/lib/wsogclient.jar:\
lib/kynwb.jar:\
lib/kynwxssec_api.jar:\
lib/itcam.cg.mbean.jar:\
wasdc/7.3/installer/lib/itcamfwas.jar:\
```

3. 在 *install_dir*/*platform_code*/yn/config/kynwb.properties 文件末尾，添加用于指示代理程序要使用的安全属性文件的行。通常，这些文件即 wsadmin 实用程序所使用的文件：

```
-Dcom.ibm.CORBA.ConfigURL=file:/appserver_profile/properties/sas.client.props
-Dcom.ibm.SSL.ConfigURL=file:/appserver_profile/properties/ssl.client.props
```

如果代理程序所需的安全性设置不同于 wsadmin 实用程序所使用的设置，请创建这些文件的独立副本，并改为提供这些副本的路径，例如：

```
-Dcom.ibm.CORBA.ConfigURL=file:/opt/IBM/ITM/config/sas.client.props
-Dcom.ibm.SSL.ConfigURL=file:/opt/IBM/ITM/config/ssl.client.props
```

注： 为 WebSphere Applications 代理程序安装修订包或临时修订时，会覆盖 yn.ini 和 kynwb.properties 文件中的更改。因此，安装修订包或临时修订后，必须重新在这两个文件中完成更改。

设置连接凭证
使用安全的 Java 客户机时，它需要读取包含 CSIv2 设置的属性文件。这些设置可确定客户机向服务器进行认证的方式。您必须确保正确配置认证。

关于此任务

通常，在 com.ibm.CORBA.ConfigURL JVM 属性中指定包含这些设置的文件。在 com.ibm.SSL.ConfigURL JVM 属性中指定的文件中可以找到更多 SSL 设置。

当 WebSphere Applications 代理程序配置为监视 WebSphere Application Server 中嵌入的 eXtreme Scale 服务器时，它会充当安全的 Java 客户机。因此，必须在 kynwb.properties 文件中指定 -Dcom.ibm.CORBA.ConfigURL 和 -Dcom.ibm.SSL.ConfigURL。

在大部分情况下，这些属性指向位于 *appserver_profile*/properties 目录中的 sas.client.props 和 ssl.client.props 文件。这些文件可供 wsadmin 或 xscmd 之类的工具使用。因此，如果不需要输入任何凭证即可使用其中一个工具来连接到 Extreme Scale 目录服务器，那么不需要定制设置。

如果连接失败或需要输入用户名或密码，那么必须完成额外配置。

修改客户机属性文件
编辑要供 WebSphere Applications 代理程序 使用的 sas.client.props 文件。

关于此任务
在 kynwb.properties 中的 -Dcom.ibm.CORBA.ConfigURL 属性中指定 sas.client.props 文件的完整路径。为运行已配置代理程序的目录服务实例的 WebSphere Application Server 实例提供连接和安全信息。

过程

1. 打开 *appserver_profile*/properties/sas.client.props 文件。

2. 将 com.ibm.CORBA.loginSource 属性的值更改为 properties：

```
com.ibm.CORBA.loginSource=properties
```

3. 将 com.ibm.CORBA.securityServerHost 属性设置为 WebSphere Extreme Scale 区域中的应用程序服务器的主机名。服务器可以是本地服务器或其他服务器。代理程序启动时，服务器必须始终可用。例如：

```
com.ibm.CORBA.securityServerHost=server.company.com
```

4. 将 com.ibm.CORBA.securityServerPort 属性设置为应用程序服务器概要文件的 RMI 端口，例如：

```
com.ibm.CORBA.securityServerPort=2819
```

5. 将 com.ibm.CORBA.loginUserid 属性设置为用于与应用程序服务器进行通信的登录名称，并将 com.ibm.CORBA.loginPassword 属性设置为密码，例如：

```
com.ibm.CORBA.loginUserid=admin
com.ibm.CORBA.loginPassword=password
```

6. 将以下属性设置为 true 或 false，对应于 WebSphere 管理控制台中的 **CSIv2 入站通信**设置：

```
com.ibm.CSI.performTLClientAuthenticationRequired
com.ibm.CSI.performTLClientAuthenticationSupported
com.ibm.CSI.performTransportAssocSSLTLSRequired
com.ibm.CSI.performTransportAssocSSLTLSSupported
```

   com.ibm.CSI.performTLClientAuthentication* 属性与 **Client certificate authentication** 设置相关。com.ibm.CSI.performTransportAssocSSLTLS* 与 **Transport** 设置相关。

7. 可选：如果不使用缺省 SSL 别名 (DefaultSSLSettings)，请在 com.ibm.ssl.alias 属性中设置 SSL 配置别名。
8. 保存文件，然后对 sas.client.props 文件中的密码进行加密。要加密密码，请运行以下命令：

   - 在 Linux 和 UNIX 系统上，运行 *appserver_profile*/bin/PropFilePasswordEncoder.sh sas.client.props com.ibm.CORBA.loginPassword

   要点：如果需要客户机证书认证并且启用基本认证，那么您可能还需要设置 com.ibm.CORBA.validateBasicAuth=false 属性。

修改客户机 *SSL* 属性文件
修改 WebSphere Applications 代理程序用于访问服务器证书的 SSL 属性文件。

关于此任务

编辑要供代理程序使用的 ssl.client.props 文件。在 kynwb.properties 文件中的 -Dcom.ibm.SSL.ConfigURL 属性中指定该文件的完整路径。为运行已配置代理程序的目录服务实例的 WebSphere Application Server 实例提供 SSL 信任库和密钥库信息。

您可以使用 WebSphere 管理控制台（**安全性** > **SSL 证书和密钥管理** > **密钥库和证书**）或者通过使用 iKeyman 工具来创建和管理证书。

过程

1. 打开 *appserver_profile*/properties/ssl.client.props 文件。
2. 更改 com.ibm.ssl.alias 属性的值以与 sas.client.props 文件中的相同属性的值相匹配。

   提示：ssl.client.props 文件可包含多个 SSL 配置。每个配置都以 com.ibm.ssl.alias 属性开头。
3. 将 com.ibm.ssl.enableSignerExchangePrompt 属性设置为 false。
4. 设置以下密钥库属性以支持客户机应用程序访问加密密钥：

com.ibm.ssl.keyStoreName
用于识别此密钥库的名称

com.ibm.ssl.keyStore
密钥库文件的完整路径和名称

com.ibm.ssl.keyStorePassword
密钥库的密码

com.ibm.ssl.keyStoreType
密钥库类型。使用缺省 PKCS12 类型，因为此类型与其他应用程序之间具有卓越的互操作性。

要点: 如果不需要客户机证书认证，那么密钥库可包含任何自签名密钥。否则，密钥库必须包含服务器信任库中的证书签署的密钥。

5. 设置以下信任库属性以支持客户机应用程序访问签署者证书：

com.ibm.ssl.trustStoreName
用于标识此信任库的名称

com.ibm.ssl.trustStore
信任库文件的完整路径和名称

com.ibm.ssl.trustStorePassword
信任库的密码

com.ibm.ssl.trustStoreType
信任库类型。使用缺省 PKCS12 类型，因为此类型与其他应用程序之间具有卓越的互操作性。

要点: 如果客户机要使用 SSL 连接，那么服务器签署者证书必须包含在其信任库中。

运行配置
检查环境和安全性后，即可运行配置过程。

过程

1. 停止 WebSphere Applications 代理程序。
   a) 转至安装 WebSphere Applications 代理程序的 *install_dir* 目录。
   b) 运行命令 `bin/was-agent.sh stop`。
2. 运行配置脚本。

```
install_dir/platform_code/yn/bin/wxs-agent-config.sh config
```

   其中

   · *install_dir* 是 WebSphere Applications 代理程序的安装目录。
   · *platform_code* 是安装代理程序的平台代码，例如，lx8266 表示 Linux x86_64 R2.6（64 位），aix536 表示 AIX R5.3（64 位）。

   示例命令：

```
/opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh config
```

```
/opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh config
```

3. 出现输入代理程序安装路径提示时，请指定 WebSphere Applications 代理程序的主目录。

   注: 此脚本会根据您指定的安装路径查找配置文件名称。缺省路径为 *install_dir*/config/${hostname}_yn.xml。如果提示此文件不存在，原因可能是执行配置时未启动 WebSphere Applications 代理程序。请启动 WebSphere Applications 代理程序，然后将其至少停止一次。
4. 出现 `WebSphere Extreme Scale Catalog Server connector type` 提示时，请按 2 继续。
5. 出现 `Input a node name to identify this agent node on UI` 提示时，请输入节点名称。

   节点名称用于识别受监视的 WebSphere Extreme Scale 区域，它显示在实例名称中，您可在“应用程序性能仪表板”UI 中看到此实例名称。

6. 出现 `WebSphere Extreme Scale Catalog Server security enabled?` 提示时，请按 `1` 继续。然后，输入用户名和密码。
7. 指定目录服务器的主机名和端口号。如果存在多个目录服务器，可以逐一添加。您还可以逐一添加多个区域。
   - 主机名即目录服务器所在位置的系统名称。请确保主机名可供访问。如果无法访问，请使用 IP 地址作为主机名。
   - 端口号用于指示 WebSphere Extreme Scale 目录服务器的 **JMXServicePort** 编号。它继承自每个 WebSphere Application Server 的 **BOOTSTRAP_ADDRESS** 值。在 WebSphere Extreme Scale Knowledge Center 中可找到有关端口号的更多详细信息。
8. 要启动代理程序，请运行以下命令。

```
install_dir/bin/was-agent.sh start
```

注:

- 代理程序配置存储在 *install_dir*`/config/${hostname}_yn.xml` 中。如果您要更改任意配置，请重新运行此脚本，或者直接修改 `.xml` 文件。
- 先前配置将备份为 *install_dir*`/config/${hostname}_yn.xml.bak`。如果需要，您可复原先前配置。
- 运行 *install_dir*`/bin/wxs-agent-config.sh config` 时，可按 Ctrl-C 退出现有脚本。现有配置不会发生更改。

取消配置 WebSphere Extreme Scale 监视

如果不想监视 WebSphere Extreme Scale，可以取消配置 WebSphere Applications 代理程序。

过程

1. 停止 WebSphere Applications 代理程序。
   a) 转至安装 WebSphere Applications 代理程序的 *install_dir* 目录。
   b) 运行命令 `bin/was-agent.sh stop`。
2. 运行取消配置脚本。

```
install_dir/{pc}/yn/bin/wxs-agent-config.sh unconfig
```

其中

- *install_dir* 是 WebSphere Applications 代理程序的安装目录。
- *platform_code* 是安装代理程序的平台代码，例如，lx8266 表示 Linux x86_64 R2.6（64 位），aix536 表示 AIX R5.3（64 位）。

示例命令:

```
/opt/ibm/apm/agent/lx8266/yn/bin/wxs-agent-config.sh unconfig
```

```
/opt/ibm/apm/agent/aix536/yn/bin/wxs-agent-config.sh unconfig
```

配置 WebSphere Infrastructure Manager 监视

配置 WebSphere Infrastructure Manager 代理程序以监视 WebSphere Deployment Manager 和 Node Agent 的性能。

关于此任务

WebSphere Infrastructure Manager 代理程序是一个多实例代理程序。必须手动创建第一个实例并启动代理程序。

过程

1. 要配置代理程序，请运行以下命令。

   ```
   install_dir/bin/wim-agent.sh config instance_name
   ```

   其中，*instance_name* 是要对实例指定的名称，而 *install_dir* 是 WebSphere Infrastructure Manager 代理程序的安装目录。缺省安装目录为 `/opt/ibm/apm/agent`。
2. 当出现 `Edit 'Monitoring Agent for WebSphere Infrastructure Manager' settings` 提示时，输入 `1` 以继续。
3. 当出现 `Java home` 提示时，指定安装 Java 的目录。

   缺省值为 `/opt/ibm/apm/agent/JRE/lx8266/jre`。
4. 当出现 `DMGR Profile Home` 提示时，指定 Deployment Manager 概要文件的主目录。

   缺省目录为 `/opt/IBM/WebSphere/AppServer/profiles/Dmgr01`。
5. 当出现 `JMX user ID` 提示时，指定用于连接到 MBean Server 的用户标识。
6. 当出现 `Enter JMX password` 提示时，指定用户的密码。
7. 当出现 `Re-type JMX password` 提示时，再次输入密码。
8. 要启动代理程序，请运行以下命令。

   ```
   install_dir/bin/wim-agent.sh start instance_name
   ```

结果

您创建了 WebSphere Infrastructure Manager 代理程序 实例，并启动了监视代理程序以开始收集数据样本进行资源监视。

配置 WebSphere MQ 监视

您必须先分配一个实例名称给此代理程序，并完成用户标识和受管系统名称的多项配置任务，然后才能启动此代理程序。您也可以选择性地对此代理程序启用事务跟踪。

开始之前

- 此处的方向针对的是此代理程序的最新发行版。有关如何检查环境中代理程序版本的信息，请参阅 代理程序版本命令。有关代理程序版本列表以及每个版本的新增内容的详细信息，请参阅第 44 页的『变更历史记录』。
- 请确保在您的环境中符合 WebSphere MQ 代理程序的系统需求。有关最新系统需求信息，请参阅 WebSphere MQ 代理程序 的详细系统需求报告。

关于此任务

依以下指示信息针对的是代理程序的最新发行版，除非另有说明。

要为 WebSphere MQ 代理程序设置环境，您必须先确保代理程序用户标识可访问 IBM MQ (WebSphere MQ) 对象，配置 IBM MQ (WebSphere MQ) 以启用数据，然后配置 WebSphere MQ 代理程序。

以下过程是配置 WebSphere MQ 代理程序的路线图，其中包括必需步骤和可选步骤。请根据您的需要，完成必要的步骤。

过程

1. 对用于配置、启动和停止代理程序的用户标识进行授权，使其可访问 IBM MQ (WebSphere MQ) 对象。请参阅第 792 页的『授权用户标识运行代理程序』。
2. 配置 IBM MQ (WebSphere MQ)，以启用您想要监视的数据。请参阅第 793 页的『配置 IBM MQ (WebSphere MQ) 以启用数据』。

3. 通过提供代理程序实例名称、队列管理器名称和代理程序名称（可选）来配置代理程序。请参阅第 795 页的『配置 WebSphere MQ 代理程序』。
4. 可选： 根据您的监视需求不同，可能需要唯一的受管系统名称，以区分不同的监视代理程序。请在 `mq-agent.sh config` 命令中使用 `Agent Name` 选项来指定受管系统名称的中间限定符。请参阅第 798 页的『为多个队列管理器指定唯一受管系统名称』。
5. 可选： 要配置代理程序以收集受监视队列管理器的事务跟踪数据，请使用“**代理程序配置**”页面。有关指示信息，请参阅第 799 页的『配置 WebSphere MQ 代理程序的事务跟踪』。
6. 可选： 使代理程序能够收集队列和通道的长期历史记录数据。有关指示信息，请参阅第 800 页的『为队列和通道长期历史记录启用数据收集』。
7. 可选： 要从远端监视 MQ 设备上的队列管理器，需要对代理程序和 IBM MQ (WebSphere MQ) 进行额外配置。有关指示信息，请参阅第 801 页的『远程监视 MQ 设备上的队列管理器』或第 802 页的『远程监视 MQ 设备上的 HA 队列管理器』。

授权用户标识运行代理程序

要使用户标识能够配置、启动和停止 WebSphere MQ 代理程序，该用户标识必须属于 **mqm** 组，该组对 IBM MQ (WebSphere MQ) 具有完整的管理特权。此外，对于非 root 用户或非管理员用户，必须使用 IBM MQ (WebSphere MQ) 控制命令授予用户对 IBM MQ (WebSphere MQ) 对象的访问权。

关于此任务

在 AIX 或 Linux 系统上，您必须将用户标识添加到 **mqm** 组，然后使用 **setmqaut** 命令授予该用户标识对 IBM MQ (WebSphere MQ) 对象的相应访问权。

在 Windows 系统上，您必须将用户标识添加到 **mqm** 组。如果用户标识不属于管理员用户组，那么还必须使用注册表编辑器来授予该用户标识启动或停止代理程序的许可权。

过程

- Linux AIX

  在 AIX 或 Linux 系统上，请完成下列步骤：

  a) 通过使用 root 标识登录到 AIX 或 Linux 系统。

  b) 将用于运行代理程序的用户标识添加到 **mqm** 组。

  c) （WebSphere MQ V7.5 或更高版本）： 如果用户标识是 AIX 或 Linux 系统上的非 root 用户，请运行以下命令为该用户标识授予访问 IBM MQ (WebSphere MQ) 对象的适当权限级别：

  ```
  setmqaut -m queue_manager -t qmgr -p user_ID +inq +connect +dsp +setid
  ```

  其中，*queue_manager* 是 WebSphere MQ V7.5 或更高版本的队列管理器的名称，*user_ID* 是运行代理程序的非 root 用户标识或非管理员用户标识。

- Windows

  在 Windows 系统上，完成下列步骤：

  a) 以系统管理员身份登录到 Windows 系统。

  b) 将用于运行代理程序的用户标识添加到 **mqm** 组。

  c) 如果用于启动、运行和停止代理程序的用户标识不是 Administrators 组的成员，请使用“注册表编辑器”来设置用户标识的许可权以确保能够成功启动和停止代理程序：

  a. 单击**开始** > **运行**，然后输入 `services.msc` 以打开注册表编辑器。

  b. 在注册表编辑器中，找到以下键：`HKEY_LOCAL_MACHINE\SOFTWARE\Candle`。

  c. 右键单击此键，并单击**权限**。

  d. 如果 WebSphere MQ 代理程序的用户标识不在组或用户名列表中，单击**添加**以将该用户标识添加到该列表中。

  e. 单击列表中的用户标识。

f. 在“*user-ID* 许可权”列表（其中，*user-ID* 是 WebSphere MQ 代理程序 的用户标识）中，在“允许”列中选择**完全控制**并单击**确定**。

g. 在注册表编辑器中，找到以下键：HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib。

h. 右键单击此键，并单击**权限**。

i. 如果 WebSphere MQ 代理程序的用户标识不在组或用户名列表中，单击**添加**以将该用户标识添加到该列表中。

j. 单击组或用户名列表中的用户标识。

k. 在“*user-ID* 许可权”列表（其中，*user-ID* 是 WebSphere MQ 代理程序 的用户标识）中，在“允许”列中选择**读**并单击**确定**。

l. 关闭注册表编辑器。

m. 找到 *install_dir* 目录，其中，*install_dir* 是代理程序安装目录。

n. 右键单击此目录，并单击**属性**。

o. 在“安全性”选项卡上，如果 WebSphere MQ 代理程序的用户标识不在组或用户名列表中，单击**编辑**，然后单击**添加**以将该用户标识添加到该列表中。

p. 单击组或用户名列表中的用户标识。

q. 在“*user-ID* 的权限”列表中，在“允许”列中选择**完全控制**，其中，**user-ID** 是 WebSphere MQ 代理程序 的用户标识。

r. 单击**确定**。

下一步做什么

下一个步骤是配置 IBM MQ (WebSphere MQ) 以启用数据。请参阅第 793 页的『配置 IBM MQ (WebSphere MQ) 以启用数据』。

配置 IBM MQ (WebSphere MQ) 以启用数据

在配置 WebSphere MQ 代理程序之前，建议您先配置 IBM MQ (WebSphere MQ)，以启用所要监视的数据。

关于此任务

确定要让 WebSphere MQ 代理程序监视的数据类型。如果队列管理器在缺省情况下不会生成这些数据，请在队列管理器上使用 MQSC 命令来启用这些数据。

切记: 在发出 MQSC 命令之前，必须对目标队列管理器启动 MQSC。要获取队列管理器列表，请从 IBM MQ (WebSphere MQ) 安装目录中的 bin 目录发出 **dspmq** 命令。要针对某个队列管理器来启动 MQSC，请从 bin 目录发出下列命令，其中 *<qmgr_name>* 是所要配置的队列管理器的名称。

```
runmqsc <qmgr_name>
```

过程

- 要查看队列中最早消息的存在时间，请完成第 793 页的『启用队列实时监视』中记载的步骤。
- 要监视队列管理器在缺省情况下不会生成的特定队列管理器事件，请完成第 794 页的『对队列管理器启用事件监视』所述的步骤。
- 要获取事务跟踪数据，请完成第 794 页的『启用 MQI 应用程序活动跟踪』所述的步骤。
- 要监视远程队列管理器，请确保 WebSphere MQ 代理程序可通过远程系统上的通道收集监视数据。有关更多信息，请参阅第 794 页的『远程监视的安全设置』。

启用队列实时监视

关于此任务

要查看队列中最早消息的存在时间（以秒计），您必须启用队列实时监视。

过程

使用以下命令在您的环境中启用队列实时监视。

- 要对 MONQ 属性设置为 QMGR 的所有队列启用实时监视，请发出以下命令：

  ```
  ALTER QMGR MONQ(collection_level)
  ```

  其中，*collection_level* 为队列指定监视数据的收集级别。您可以将其设置为低、中或高来满足环境需求。
- 要启用单个队列实时监视，请发出以下命令：

  ```
  ALTER QLOCAL(queue_name) MONQ(collection_level)
  ```

  其中，*queue_name* 是队列名称；*collection_level* 为队列指定监视数据的收集级别。您可以将其设置为低、中或高来满足环境需求。

结果

在 WebSphere MQ 代理程序启动后，数据可显示在“队列的最早消息存在时间”组窗口小部件中。

对队列管理器启用事件监视

关于此任务

事件监视是其中一项可用来监视 IBM MQ 网络的监视技术。启用队列管理器以发出特定类型的事件之后，事件发生时，事件消息将放置在事件队列中。这样，这些事件消息就可以由 WebSphere MQ 代理程序来监视并显示。

使用缺省的队列管理器配置时，不会监视和显示下列类型的事件。使用 **ALTER QMGR** 命令可允许队列管理器生成这些事件，以使这些事件可显示在应用程序性能仪表板上。

- 通道事件
- 性能事件

过程

使用下列命令可允许队列管理器生成您所关注的事件：

- 要生成通道事件，请发出 `ALTER QMGR CHLEV(ENABLED)`。
- 要生成性能事件，请发出 `ALTER QMGR PERFMEV(ENABLED)`。

结果

在 WebSphere MQ 代理程序启动后，受监视事件可以显示在“队列管理器事件”组窗口小部件中。

启用 MQI 应用程序活动跟踪

关于此任务

要让事务跟踪数据显示在中间件和拓扑仪表板中，必须在队列管理器上启用 MQI 应用程序活动跟踪。

过程

- 要启用 MQI 应用程序活动跟踪信息收集，请发出下列 MQSC 命令：

  ```
  ALTER QMGR ACTVTRC(ON)
  ```

远程监视的安全设置

关于此任务

要使用 WebSphere MQ 代理程序监视远程队列管理器，您必须确保 IBM MQ (WebSphere MQ) 的安全设置不阻止代理程序通过远程系统上的通道收集监视数据。

以下过程提供远程监视的简单安全设置的示例。要更准确地控制为通道级别连接系统授予的访问权，您可以使用通道认证记录。有关更多信息，请参阅 IBM MQ 安全性机制文档。

过程

1. 运行以下 MQSC 命令来禁用通道认证：

```
ALTER QMGR CHLAUTH(DISABLED) CONNAUTH(' ')
```

2. 按如下所示更改通道设置，其中 *channel_for_remote_monitor* 是用于远程监视的通道的名称。
   - Linux AIX

```
ALTER CHANNEL(channel_for_remote_monitor) CHLTYPE(SVRCONN) MCAUSER('mqm')
```

   - Windows

```
ALTER CHANNEL(channel_for_remote_monitor) CHLTYPE(SVRCONN) MCAUSER(MUSR_MQADMIN)
```

3. 刷新安全设置。

```
REFRESH SECURITY
```

配置 WebSphere MQ 代理程序

您必须为 WebSphere MQ 代理程序分配一个实例名称并配置此代理程序，它才能开始监视 IBM MQ (WebSphere MQ) 环境。

开始之前

- 确保代理程序用户标识具有相应的许可权，可访问 IBM MQ (WebSphere MQ) 对象。如果您尚未完成此任务，请按第 792 页的『授权用户标识运行代理程序』中的指示信息进行操作。
- 配置 IBM MQ (WebSphere MQ)，以启用所需的数据收集。如果您尚未完成此任务，请参阅第 793 页的『配置 IBM MQ (WebSphere MQ) 以启用数据』。
- 您必须提供 WebSphere MQ 代理程序所要监视的队列管理器名称。如果您不知道相应的队列管理器名称，请联系 IBM MQ (WebSphere MQ) 管理员。或者，从 IBM MQ (WebSphere MQ) 安装目录中的 `bin` 目录发出 **dspmq** 命令，以获取队列管理器列表。返回的 QMNAME 值就是您配置 WebSphere MQ 代理程序时必须提供的内容。

关于此任务

WebSphere MQ 代理程序是多实例代理程序；您必须创建第一个实例并手动启动此代理程序。

在 UNIX 或 Linux 系统上，您可以选择在进行或不进行交互的情况下配置此代理程序。在 Windows 系统上，只能在不进行交互的情况下配置此代理程序。

- 要通过交互配置代理程序，请运行配置脚本并对提示做出响应。请参阅第 795 页的『交互式配置』。
- 要不使用交互配置代理程序，请编辑静默响应文件，然后运行配置脚本。请参阅第 796 页的『静默配置』。

要点: 如果在安装 WebSphere MQ 代理程序（在 Cloud APM 中交付）的系统上也安装有 Monitoring Agent for WebSphere MQ（作为 ITCAM for Applications 产品的一个组件来交付），请勿使用它们来监视该系统上的同一个队列管理器。

交互式配置

过程

要通过运行脚本并响应提示来配置代理程序，请完成下列步骤：

1. 输入以下命令来创建代理程序实例：

```
install_dir/bin/mq-agent.sh config instance_name
```

其中 *instance_name* 是要对实例指定的名称。

2. 当出现 `Queue Manager Name` 提示时，请指定要监视的队列管理器名称。
3. 当出现 `Agent Name` 提示时，请指定用作受管系统名称中间限定符的代理程序名称。请不要按 Enter 键以跳过指定此参数。

   切记: 此代理程序名称不同于代理程序实例名称。代理程序配置文件名中将使用代理程序实例名称以区分不同代理程序的配置文件，例如，*hostname*_mq_*instancename*.cfg。代理程序名称用作创建唯一受管系统名称的短标识。要了解何时需要唯一受管系统名称，请参阅第 798 页的『为多个队列管理器指定唯一受管系统名称』。
4. 如果要监视远程队列管理器，请指定以下配置参数。如果要监视本地队列管理器，请按 Enter 键以继续操作。

   - `Connection Name`：用于远程监视的连接名称。格式为 *IP_address* (*port_number*)，例如，`127.0.0.1(1414)`。如果这是您首次配置代理程序实例，您可以按 Enter 键以接受缺省值空值。可自动发现相应的连接名称。
   - `Channel`：用于远程数据收集的通道的名称。如果这是您首次配置代理程序实例，您可以按 Enter 键以接受缺省值空值。将使用 SYSTEM.DEF.SVRCONN 通道。

   限制: 无法监视远程队列管理器的错误日志。当代理程序正在监视远程队列管理器时，“MQ 错误详细信息”仪表板不包含数据。
5. 提示输入 WebSphere MQ 库路径时，请按 Enter 键以接受缺省值，即 WebSphere MQ 代理程序自动发现的 IBM MQ (WebSphere MQ) 64 位库路径。如果没有显示缺省值，那么您必须提供 IBM MQ (WebSphere MQ) 的 64 位库路径才能继续。
   对于 Linux 系统，`/opt/mqm8/lib64` 是 64 位库路径的一个示例。
6. 要启动代理程序，请输入以下命令：

```
install_dir/bin/mq-agent.sh start instance_name
```

静默配置

过程

要通过编辑静默响应文件并在无交互情况下运行脚本来配置代理程序，请完成下列步骤：

1. 在文本编辑器中打开 `mq_silent_config.txt` 文件。

   - Linux AIX *install_dir*/samples/mq_silent_config.txt
   - Windows *install_dir*\tmaitm6_x64\samples\mq_silent_config.txt

   其中，*install_dir* 是 WebSphere MQ 代理程序的安装目录。
2. 必需： 对于 **QMNAME**，指定要监视的队列管理器的名称。
3. 必需： 对于 **AGTNAME**，指定用作受管系统名称中间限定符的代理程序名称。

   切记: 此代理程序名称不同于代理程序实例名称。代理程序配置文件名中将使用代理程序实例名称以区分不同代理程序的配置文件，例如，*hostname*_mq_*instancename*.cfg。代理程序名称用作创建唯一受管系统名称的短标识。要了解何时需要唯一受管系统名称，请参阅第 798 页的『为多个队列管理器指定唯一受管系统名称』。
4. 如果要监视远程队列管理器，请指定以下配置参数：

   - **CONNAME**：用于远程监视的连接名称。格式为 *IP_address* (*port_number*)，例如，`127.0.0.1(1414)`。
   - **CHANNEL**：用于远程数据收集的通道的名称。如果未指定，将使用 SYSTEM.DEF.SVRCONN 通道。

   限制: 无法监视远程队列管理器的错误日志。当代理程序正在监视远程队列管理器时，“MQ 错误详细信息”仪表板不包含数据。

5. 可选： 对于 **WMQLIBPATH**，请指定 IBM MQ (WebSphere MQ) 的 64 位库路径。例如，`/opt/mqm8/lib64`。如果未指定值，那么在代理程序配置期间可自动发现路径。
6. 保存并关闭 `mq_silent_config.txt` 文件，然后在命令行中运行以下命令：
   - Linux AIX *install_dir*`/bin/mq-agent.sh config` *instance_name path_to_responsefile*
   - Windows *install_dir*`\BIN\mq-agent.bat config` *instance_name* `"`*path_to_responsefile*`"`

   其中，*instance_name* 是配置的实例的名称，*path_to_responsefile* 是静默响应文件的完整路径。

   切记: 在 Windows 系统上，请勿省略将静默响应文件的路径括住的双引号 (""），在该路径包含特殊字符时尤其如此。

   例如，如果响应文件位于缺省目录中，请运行以下命令。
   - Linux AIX

     ```
     /opt/ibm/apm/agent/bin/mq-agent.sh config instance_name
     /opt/ibm/apm/agent/samples/mq_silent_config.txt
     ```
   - Windows

     ```
     C:\IBM\APM\BIN\mq-agent.bat config instance_name
     "C:\IBM\APM\tmaitm6_x64\samples\mq_silent_config.txt"
     ```
7. 要启动代理程序，请输入以下命令：
   - Linux AIX

     ```
     install_dir/bin/mq-agent.sh start instance_name
     ```
   - Windows

     ```
     install_dir\bin\mq-agent.bat start instance_name
     ```

结果

现在，您可登录 Cloud APM 控制台，并使用“应用程序”编辑器将 WebSphere MQ 代理程序实例添加到“应用程序性能仪表板”。有关如何启动 Cloud APM 控制台的指示信息，请参阅第 829 页的『启动 Cloud APM 控制台』。有关使用“应用程序”编辑器的信息，请参阅第 930 页的『管理应用程序』。

下一步做什么

- 如果已在队列管理器上启用 MQI 应用程序活动跟踪信息收集，请使用“**代理程序配置**”页面来配置 WebSphere MQ 代理程序，以收集受监视队列管理器的事务跟踪数据。请参阅第 799 页的『配置 WebSphere MQ 代理程序的事务跟踪』。如果该代理程序没有显示在“**代理程序配置**”页面上，请重新启动 Cloud APM 服务器。
- 根据您的监视需求不同，可能需要唯一的受管系统名称，以区分不同的监视代理程序。请在 `mq-agent.sh config` 命令中使用 `Agent Name` 选项来指定受管系统名称的中间限定符。请参阅第 798 页的『为多个队列管理器指定唯一受管系统名称』。
- 要配置 WebSphere MQ 代理程序以进行远程监视，在创建代理程序实例后必须执行一些手动配置。有关指示信息，请参阅以下主题：
  - 第 801 页的『远程监视 MQ 设备上的队列管理器』
  - 第 802 页的『远程监视 MQ 设备上的 HA 队列管理器』

为多个队列管理器指定唯一受管系统名称

有时需要唯一受管系统名称来区分连接到同一 Cloud APM 服务器的不同监视代理程序。在静默响应文件中使用 **AGTNAME** 参数或在 `mq-agent.sh config` 命令中使用 **Agent Name** 选项来指定在受管系统名称中使用的中间限定词。

关于此任务

当 WebSphere MQ 代理程序 启动时，它将注册下列受管系统：

```
monitoredqueuemanagername:agentname:MQ
```

其中

- *monitoredqueuemanagername* 是代理程序监视的队列管理器的名称。
- *agentname* 是受管系统名称的中间限定符。如果未指定 *agentname* 值，那么不会使用任何值。

在下列情况下，指定代理程序名称值非常有用：

- 如果您的站点包含具有相同名称的多个队列管理器（在不同节点上运行），请为每个队列管理器指定代理程序名称，以便 WebSphere MQ 代理程序可以创建唯一受管系统名称。
- 如果受管系统名称的长度超出 32 个字符，那么 2 个不同队列管理器名称由于截断可以解析为同一名称。要区别队列管理器的受管系统名称，请为每个队列管理器指定代理程序名称。
- 如果要按主机名和队列管理器名称以外的内容（如高可用性集群名称）来分组和识别队列管理器名称。
- 如果要让多个连接至同一 Cloud APM 服务器的代理程序，以监视不同主机上具有相同名称的队列管理器。

交互式配置

过程

要在 `mq-agent.sh config` 命令中使用 **Agent Name** 选项，请完成下列步骤：

1. 在命令行中，运行以下命令以开始配置 WebSphere MQ 代理程序。

   ```
   ./mq-agent.sh config instance_name
   ```

   其中，*instance_name* 是启动的实例的名称。
2. 遵循用于配置代理程序实例的选项。

   队列管理器的名称是必需的。对于其他选项，如果不需要任何更改，请使用缺省值。
3. `Agent Name` 选项显示时，请指定受管系统名称的中间限定符。

   切记： 完整受管系统名称是 *monitoredqueuemanagername*:*agentname*:MQ。完整的受管系统名称最长为 32 个字符，所以中间限定符 *agentname* 的最大长度取决于队列管理器名称的长度。如果为 `Agent Name` 选项指定的值超过最大长度，那么会将 *agentname* 的值截断为不超过 8 个字符。

   例如，在名为 LINUX2 的节点上存在名为 PERSONNEL 的队列管理器时，要监视 AIX1 节点上另一个也名为 PERSONNEL 的队列管理器，请首先对 AIX1 节点运行以下命令：

   ```
   ./mq-agent.sh config PERSONNEL
   ```

   然后在 *Agent Name* 选项显示时指定代理程序名称：

   ```
   Agent Name (default is: ): AIX1
   ```

   要同时监视 LINUX2 节点上的 PERSONNEL 队列管理器，请首先运行以下命令：

   ```
   ./mq-agent.sh config PERSONNEL
   ```

   然后指定代理程序名称：

   ```
   Agent Name (default is: ): LINUX2
   ```

切记: 用于代码样本中 `Agent Name` 选项的代理程序节点名仅供说明之用。您可以为 `Agent Name` 选项指定其他字符串。

静默配置

过程

要在静默响应文件中使用 **AGTNAME** 参数，请完成下列步骤：

1. 在文本编辑器中打开 `mq_silent_config.txt` 静默响应文件。
2. 为 **AGTNAME** 参数指定代理程序名称。

   切记: 完整受管系统名称是 *monitoredqueuemanagername*:*agentname*:MQ。完整的受管系统名称最长为 32 个字符，所以中间限定符 *agentname* 的最大长度取决于队列管理器名称的长度。如果为 **AGTNAME** 参数指定的值超过最大长度，那么会将 *agentname* 的值截断为不超过 8 个字符。
3. 保存并关闭 `mq_silent_config.txt` 文件，然后在命令行中运行以下命令：

   ```
   install_dir/BIN/mq-agent.sh config instance_name path_to_responsefile
   ```

   其中，*instance_name* 是配置的实例的名称，*path_to_responsefile* 是静默响应文件的完整路径。

下一步做什么

登录到 Cloud APM 控制台。如果具有先前 MSN 的代理程序实例仍显示为脱机，请编辑应用程序以将其移除，然后添加具有指定代理程序名称的新代理程序实例。

配置 WebSphere MQ 代理程序的事务跟踪

在 WebSphere MQ 代理程序的“**代理程序配置**”页面上启用数据收集之后，IBM MQ (WebSphere MQ) 的事务跟踪数据可以显示在中间件和拓扑仪表板中。

开始之前

- 确保在队列管理器上启用 MQI 应用程序活动跟踪信息收集。如果您在配置并启动 WebSphere MQ 代理程序之前没有完成此项操作，请按第 794 页的『启用 MQI 应用程序活动跟踪』中的指示信息进行操作，然后重新启动代理程序。
- 确保事务跟踪功能支持您要使用的 IBM MQ (WebSphere MQ) 的版本。有关受支持的 IBM MQ (WebSphere MQ) 的最新信息，请参阅 WebSphere MQ 代理程序 的详细系统需求报告中的先决条件声明。
- 确保 WebSphere MQ 代理程序已配置为监视队列管理器。有关指示信息，请参阅第 795 页的『配置 WebSphere MQ 代理程序』。

切记: 确保已将 WebSphere MQ 代理程序 升级到最新版本。 必须升级代理程序，并配置和启用事务跟踪以在某些窗口小部件（例如“消息量”窗口小部件）中查看数据。

过程

要配置 WebSphere MQ 代理程序 的事务跟踪，请完成下列步骤：

1. 从导航栏中，单击 **系统配置** > **代理程序配置**。

   “**代理程序配置**”页面随即显示。
2. 单击 **WebSphere MQ** 选项卡。
3. 选中要监视的队列管理器的相应复选框，然后执行**操作**列表中的下列其中一项操作：
   - 要启用事务跟踪，请单击**设置事务跟踪** > **已启用**。**事务跟踪**列中的状态将更新为“已启用”。

     提示: 缺省情况下，跟踪别名队列和远程队列已启用。要减少跟踪的数据量，您可以在**操作**列表中单击**设置别名队列跟踪** > **禁用**来禁止跟踪别名和远程队列。禁止跟踪别名和远程队列之后，在事务拓扑视图中将排除别名队列和远程队列。
   - 要禁用事务跟踪，请单击**设置事务跟踪** > **已禁用**。**事务跟踪**列中的状态将更新为“已禁用”。

结果

您已配置 WebSphere MQ 代理程序 以跟踪所选的队列管理器。事务跟踪数据可以显示在中间件和拓扑仪表板中。有关更多信息，请参阅第 81 页的『将中间件应用程序添加到应用程序性能仪表板』。

为队列和通道长期历史记录启用数据收集

缺省情况下，不收集队列长期历史记录和通道长期历史记录，也不会显示在任何预定义仪表板或组窗口小部件上。但是，可以启用代理程序以收集长期历史记录数据，然后使用**属性详细信息**选项卡来查询收集的数据。

开始之前

确保安装并配置 WebSphere MQ 代理程序。有关信息，请参阅第 795 页的『配置 WebSphere MQ 代理程序』。

关于此任务

通道长期历史记录或队列长期历史记录数据可以用于检测各个通道或队列的问题。

如果您是 Tivoli Data Warehouse 的用户，那么代理程序还可以将长期历史记录数据发送到 Tivoli Data Warehouse 进行进一步处理。

过程

完成以下步骤以使 WebSphere MQ 代理程序 能够收集队列长期历史记录和通道长期历史记录数据：

1. 使用文本编辑器打开以下代理程序环境文件。如果 `mq.environment` 文件不存在，请自行创建。
   - Linux AIX *install_dir*`/config/mq.environment`
   - Windows *install_dir*`\Config\KMQENV_`*instance*

   其中：
   - *install_dir* 是代理程序安装目录。缺省值是 `/opt/ibm/apm`（在 Linux 和 AIX 系统上）和 `C:\IBM\APM`（在 Windows 系统上）。
   - *instance* 是代理程序实例名称。
2. 将 **LH_COLLECTION** 值设置为 ENABLED 来启用数据收集。

   ```
   LH_COLLECTION=ENABLED
   ```
3. 可选： 如果您是 Tivoli Data Warehouse 用户并希望代理程序将收集到的数据发送到 Tivoli Data Warehouse，请将 **LH_PVTHISTORY** 值设置为 ENABLED。

   ```
   LH_PVTHISTORY=ENABLED
   ```

   切记: 仅当需要将收集到的数据发送到 Tivoli Data Warehouse 时才启用该选项。
4. 保存更改并重新启动代理程序。

结果

WebSphere MQ 代理程序 开始收集队列长期历史记录和通道长期历史记录数据。如果指定了 LH_PVTHISTORY=ENABLED，那么收集到的长期历史记录数据也会发送到 Tivoli Data Warehouse。

下一步做什么

使用**属性详细信息**选项卡以在仪表板上查看针对所配置代理程序实例收集到的数据。从**数据集**列表中选择 **Channel_Long-Term_History** 或 **Queue_Long-Term_History**。有关**属性详细信息**选项卡的更多信息，请参阅第 925 页的『创建定制图表或表页面』。

为 IBM MQ 的队列管理器启用队列统计信息监视

缺省情况下，不收集队列统计信息，并且不会在任意预定义的仪表板或组窗口小部件上显示这些统计信息。但您可以为代理程序启用收集队列管理器的统计信息的功能，然后查看收集的数据。

开始之前

确保安装并配置 WebSphere MQ 代理程序。有关信息，请参阅第 795 页的『配置 WebSphere MQ 代理程序』。

过程

完成以下步骤，启用 WebSphere MQ 代理程序，以收集统计信息数据：

1. 配置队列管理器以收集队列统计信息。运行以下 MQSC 命令：

```
ALTER QMGR STATQ(ON)
```

2. 设置收集记帐数据的时间间隔。运行以下命令：

```
ALTER QMGR STATINT(n)
```

其中，**n** 是收集记帐数据的秒数。

3. 为特定队列启用统计信息收集。运行以下 MQSC 命令：

```
ALTER QLOCAL(queue_name) STATQ(QMGR)
```

其中，**queue_name** 是您要收集其统计信息的队列的名称。

下一步做什么

使用以下其中一种方法来查看 MQ 队列统计信息监视数据：

- 在 **MQ_Queue_Statistics** 数据集的**属性详细信息**选项卡中查看监视数据。有关**属性详细信息**选项卡的更多信息，请参阅第 925 页的『创建定制图表或表页面』。
- 基于**已到期的消息计数**和 **MQ_Queue_Statistics** 的其他指标来定义阈值。有关阈值的更多信息，请参阅第 830 页的『阈值和资源组』。

远程监视 MQ 设备上的队列管理器

您可以使用 WebSphere MQ 代理程序来监视 MQ 设备环境上的远程队列管理器。

开始之前

- 在受支持的平台上安装 WebSphere MQ 代理程序。
- 安装 IBM MQ 客户机。MQ 客户机的版本必须与远程 MQ 队列管理器的版本相同。

过程

1. 设置到远程队列管理器的连接。在远程队列管理器上，定义用于与监视代理程序通信的服务器连接通道和侦听器。运行以下命令：

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_remote
> DEFINE LISTENER(listener) TRPTYPE(TCP) PORT(port_NO)
> DEFINE CHANNEL(chl_name)CHLTYPE(SVRCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_NO)') QMNAME(Qmgr_remote)
> END
```

其中：

- *qmgr_remote* 是远程队列管理器的名称。
- *listener* 是远程队列管理器上的侦听器的名称。
- *port_NO* 是用于侦听器的端口号。

· *chl_name* 是分配给服务器通道和客户机通道的名称。

· *host_IP* 是远程系统的 IP 地址。

2. 将侦听器配置为自动启动，然后在远程系统上运行以下命令以在远程队列管理器上启动侦听器：

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_remote
> ALTER LISTENER(listener) TRPTYPE(tcp) CONTROL(qmgr_remote)
> START LISTENER(listener)
> END
```

3. 针对用于启动 MQ 代理程序实例的用户标识，确保相应地配置其通道认证设置。有关更多信息，请参阅 IBM MQ Appliance Knowledge Center 中的设置队列管理器以接受客户机连接。
4. 遵循第 795 页的『配置 WebSphere MQ 代理程序』中的以下指示信息创建用于远程监视的 WebSphere MQ 代理程序实例，并在**远程监视设置**之后出现的提示中提供远程队列管理器连接信息。

```
Remote Monitoring Settings (For a local queue manager, just press Enter in
this section) :
Connection name for remote monitoring, for example: 192.168.1.1(1415)
Connection Name (default is: null):
Channel name for remote monitoring, SYSTEM.DEF.SVRCONN is as default.
Channels (default is: null):
```

5. 启动 WebSphere MQ 代理程序实例。

远程监视 MQ 设备上的 HA 队列管理器

要远程监视 MQ 设备上的 HA 队列管理器，您有两种选择。一种是使用单个代理程序实例连接到包含活动队列管理器的任意系统。另一种选择是对可能运行队列管理器的每个设备使用单独的代理程序实例。

关于此任务

本文仅说明第二种选择。要使用不同的代理程序实例，Linux 或 UNIX 系统上需要两个 WebSphere MQ 代理程序安装。在 Windows 系统上，只需要安装一个代理程序并创建不同的代理程序实例。

过程

· Linux AIX

执行以下步骤以使用 Linux 或 UNIX 系统上安装的 WebSphere MQ 代理程序进行远程监视：

a) 将 WebSphere MQ 代理程序安装在系统上的不同目录中。

b) 创建每个安装的 WebSphere MQ 代理程序的实例。有关指示信息，请参阅第 795 页的『配置 WebSphere MQ 代理程序』。

c) 修改每个代理程序实例的配置文件以将内容替换为以下行，从而启用远程监视：

```
SET GROUP NAME (GROUP1) -
DEFAULT(YES) -
RETAINHIST(120) -
COMMAND (YES) -
MSGACCESS(DESC) -
EVENTS(REMOVE) -
ACCOUNTINGINFO(REMOVE) -
STATISTICSINFO(REMOVE)

SET MANAGER NAME(qmgr_name) REMOTE(YES)
SET AGENT NAME(agentID)
SET QUEUE NAME(*) MGRNAME(qmgr_name) QDEFTYPE(PREDEFINED)
SET CHANNEL NAME(*) MGRNAME(qmgr_name)
PERFORM STARTMON SAMPINT(300) HISTORY(NO)
```

其中：

– *qmgr_name* 是 HA 队列管理器的名称。

– *agentID* 是用于标识队列管理器系统的的标识。通常是 HA 队列管理器运行所在的远程系统的主机名或 IP 地址。

配置文件名称和路径为 *install_dir*/config/*hostname*_mq_*qmgr_name*.cfg。

d) 在主队列管理器和 WebSphere MQ 代理程序之间以及辅助队列管理器和主队列管理器安装所在的远程系统上的 WebSphere MQ 代理程序之间创建一对客户机/服务器通道。

切记: 必须先运行以下所有命令，然后才能继续下一步。

a. 针对主队列管理器运行以下命令:

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_primary
>DEFINE LISTENER(listener_primary) TRPTYPE(TCP) PORT(port_no_primary)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_primary)') QMNAME(qmgr_primary)
```

其中:

- *qmgr_primary* 是主队列管理器的名称。
- *listener_primary* 是主队列管理器的侦听器的名称。
- *port_no_primary* 是侦听器使用的端口号。
- *chl_name_primary* 是想要分配给服务器通道和客户机通道的名称。
- *host_IP* 是主队列管理器安装所在的系统的 IP 地址。

b. 针对主队列管理器上的辅助队列管理器运行以下命令。这是为了将辅助队列管理器的连接信息添加到主队列管理器的客户机通道定义表文件。然后，同一个代理程序能够在主队列管理器故障转移时自动连接到辅助队列管理器。

```
>DEFINE LISTENER(listener_secondary) TRPTYPE(TCP) PORT(port_no_secondary)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_secondary)' ) QMNAME(qmgr_secondary)
```

其中:

- *qmgr_secondary* 是远程系统上的辅助队列管理器的名称。这与主队列管理器名称相同。
- *listener_secondary* 是辅助队列管理器的侦听器的名称。
- *port_no_secondary* 是侦听器使用的端口号。
- *chl_name_secondary* 是想要分配给服务器通道和客户机通道的名称。
- *host_IP* 是辅助队列管理器安装所在的系统的 IP 地址。

c. 最后，运行以下命令:

```
>END
>EXIT
```

e) 在第 1 个 MQ 设备中为 WebSphere MQ 代理程序实例创建客户机通道定义表文件 (AMQCLCHL.TAB)。

a. 使用 **runmqsc** 命令或 **runmqsc -n** 命令在第 1 个 MQ 设备上为队列管理器创建 AMQCLCHL.TAB 文件:

```
runmqsc -n
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance1(port_no_primary)') QMNAME(qmgr_name)
```

其中，*host_IP_appliance1* 是第 1 个 MQ 设备的 IP 地址；*chl_name_primary* 和 *port_no_primary* 与您在步骤 4 中定义的内容相同。

提示: 缺省情况下，AMQCLCHL.TAB 文件创建于 var/mqm/qmgrs/*qmgr_name*/@ipcc 目录中。

b. 将主 AMQCLCHL.TAB 文件移至为主队列管理器安装了 WebSphere MQ 代理程序的系统上的 *agent_install_dir*/*arch*/mq/bin 目录。

f) 在第 2 个 MQ 设备中为 WebSphere MQ 代理程序实例创建客户机通道定义表文件 (AMQCLCHL.TAB)。

a. 使用 **runmqsc** 命令或 **runmqsc -n** 命令在第 2 个 MQ 设备上为队列管理器创建 AMQCLCHL.TAB 文件：

```
runmqsc -n
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance2(port_no_secondary)') QMNAME(qmgr_name)
```

其中，*host_IP_appliance2* 是第 2 个 MQ 设备的 IP 地址；*chl_name_secondary* 和 *port_no_secondary* 与您在步骤 4 中定义的内容相同。

b. 将辅助 AMQCLCHL.TAB 文件移至为辅助队列管理器安装了 WebSphere MQ 代理程序的系统上的 *agent_install_dir*/*arch*/mq/bin 目录。

g) 针对用于在代理程序实例和队列管理器之间设置连接的用户标识，确保相应地配置其通道认证设置。

h) 对远程监视的队列管理器启动所有侦听器并启动所有 WebSphere MQ 代理程序实例。

- Windows

执行以下步骤以使用 Windows 系统上安装的 WebSphere MQ 代理程序进行远程监视：

a) 在 Windows 系统上安装 WebSphere MQ 代理程序。

b) 为每个 HA 队列管理器创建两个 WebSphere MQ 代理程序实例。

c) 修改每个代理程序实例的配置文件以将内容替换为以下行，从而启用远程监视：

```
SET GROUP NAME (GROUP1) -
DEFAULT(YES) -
RETAINHIST(120) -
COMMAND (YES) -
MSGACCESS(DESC) -
EVENTS(REMOVE) -
ACCOUNTINGINFO(REMOVE) -
STATISTICSINFO(REMOVE)

SET MANAGER NAME(qmgr_name) REMOTE(YES)
SET AGENT NAME(agentID)
SET QUEUE NAME(*) MGRNAME(qmgr_name) QDEFTYPE(PREDEFINED)
SET CHANNEL NAME(*) MGRNAME(qmgr_name)
PERFORM STARTMON SAMPINT(300) HISTORY(NO)
```

其中：

– *qmgr_name* 是 HA 队列管理器的名称。

– *agentID* 是用于标识队列管理器系统的的标识。通常是 HA 队列管理器运行所在的远程系统的主机名或 IP 地址。

提示: 配置文件名称和路径是 *install_dir*\TMAITM6_x64\mq_<*instance_name*>.cfg。

d) 在主队列管理器和 WebSphere MQ 代理程序之间以及辅助队列管理器和主队列管理器安装所在的远程系统上的 WebSphere MQ 代理程序之间创建一对客户机/服务器通道。

切记: 必须先运行以下所有命令，然后才能继续下一步。

a. 针对主队列管理器运行以下命令：

```
M2000# mqcli
M2000(mqcli)#runmqsc qmgr_primary
>DEFINE LISTENER(listener_primary) TRPTYPE(TCP) PORT(port_no_primary)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_primary)') QMNAME(qmgr_primary)
```

其中：

– *qmgr_primary* 是主队列管理器的名称。

– *listener_primary* 是主队列管理器的侦听器的名称。

– *port_no_primary* 是侦听器使用的端口号。

– *chl_name_primary* 是想要分配给服务器通道和客户机通道的名称。

– *host_IP* 是主队列管理器安装所在的系统的 IP 地址。

b. 针对主队列管理器上的辅助队列管理器运行以下命令。这是为了将辅助队列管理器的连接信息添加到主队列管理器的客户机通道定义表文件。然后，同一个代理程序能够在主队列管理器故障转移时自动连接到辅助队列管理器。

```
>DEFINE LISTENER(listener_secondary) TRPTYPE(TCP) PORT(port_no_secondary)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(SVRCONN) TRPTYPE(TCP)
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)
CONNAME('host_IP(port_no_secondary)' ) QMNAME(qmgr_secondary)
```

其中：

- *qmgr_secondary* 是远程系统上的辅助队列管理器的名称。这与主队列管理器名称相同。
- *listener_secondary* 是辅助队列管理器的侦听器的名称。
- *port_no_secondary* 是侦听器使用的端口号。
- *chl_name_secondary* 是想要分配给服务器通道和客户机通道的名称。
- *host_IP* 是辅助队列管理器安装所在的系统的 IP 地址。

c. 最后，运行以下命令：

```
>END
>EXIT
```

e) 为每个 WebSphere MQ 代理程序实例创建客户机通道定义表文件 (AMQCLCHL.TAB)。

a. 使用 **runmqsc** 命令或 **runmqsc -n** 命令在第 1 个 MQ 设备上为队列管理器创建 AMQCLCHL.TAB 文件：

```
runmqsc -n
>DEFINE CHANNEL(chl_name_primary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance1(port_no_primary)') QMNAME(qmgr_name)
```

其中，*host_IP_appliance1* 是第 1 个 MQ 设备的 IP 地址；*chl_name_primary* 和 *port_no_primary* 与您在步骤 4 中定义的内容相同。

b. 在第 2 个 MQ 设备上为队列管理器创建 AMQCLCHL.TAB 表：

```
runmqsc -n
>DEFINE CHANNEL(chl_name_secondary) CHLTYPE(CLNTCONN) TRPTYPE(TCP)+
>CONNAME('host_IP_appliance2(port_no_secondary)') QMNAME(qmgr_name)
```

其中，*host_IP_appliance2* 是第 2 个 MQ 设备的 IP 地址；*chl_name_secondary* 和 *port_no_secondary* 与您在步骤 4 中定义的内容相同。

f) 将两个 AMQCLCHL.TAB 文件重命名为不同的名称，例如，NODE1.TAB 和 NODE2.TAB。将这两个文件传输到目录 *install_dir*\TMAITM6_x64，其中，*install_dir* 是 WebSphere MQ 代理程序的安装目录。

g) 修改 kmqcma_*instance_name*.ini 文件以将 **MQCHLTAB** 值设置为每个代理程序实例的客户机通道定义表文件。
例如，在 kmqcma_instance1.ini 文件中设置 MQCHLTAB=NODE1.TAB，在 kmqcma_instance2.ini 文件中设置 MQCHLTAB=NODE2.TAB。

h) 打开 Windows 注册表编辑器，找到以下 **MQCHLTAB** 键，并将其从 AMQCLCHL.TAB 更改为每个代理程序实例的相应客户机通道定义表文件名。

- **HKEY_LOCAL_MACHINE\SOFTWARE\Candle\KMQ\Ver730*instance1*\Environment**

  MQCHLTAB=*NODE1.TAB*
- **HKEY_LOCAL_MACHINE\SOFTWARE\Candle\KMQ\Ver730*instance2*\Environment**

  MQCHLTAB=*NODE2.TAB*

i) 针对用于在代理程序实例和队列管理器之间设置连接的用户标识，确保正确地配置其通道认证设置。

j) 对远程监视的队列管理器启动所有侦听器并启动所有 WebSphere MQ 代理程序实例。

第 8 章 与其他产品和组件集成

您可以将其他产品和组件与 IBM Cloud Application Performance Management 集成，从而为您提供稳健的解决方案。

与 Cloud Event Management 集成

Cloud Event Management 提供跨服务、应用程序和基础结构的实时事件管理。在设置 Cloud Event Management 和 IBM Cloud Application Performance Management 之间的集成时，会将在 Cloud APM 中生成的所有事件发送到 Cloud Event Management。

关于此任务

在 Cloud Event Management 中配置 Webhook URL。然后，配置 Cloud APM 以使用 Webhook URL 以将事件发送到 Cloud Event Management。有关 Cloud Event Management 的其他信息，请参阅 IBM Cloud Event Management Knowledge Center。

过程

1. 单击 Cloud Event Management **管理**页面上的**集成**。
2. 单击**配置集成**。
3. 转至 **IBM Cloud Application Performance Management** 磁贴，然后单击 **配置**。
4. 输入集成的名称，然后单击**复制**以将生成的 Webhook URL 添加到剪贴板。确保已保存生成的 Webhook，从而可用于稍后的配置过程。例如，您可以将其保存到文件。
5. 要开始接收来自 Cloud APM 的警报信息，请确保在 Cloud Event Management 中将**启用来自此源的事件管理**设置为打开。
6. 单击**保存**。
7. 登录到 Cloud APM 预订。
8. 浏览至**系统配置** > **高级配置** > **事件管理器**。有关更多信息，请参阅高级配置。
9. 将 Webhook URL 粘贴到 **Cloud Event Management Webhook** 字段。
10. 单击**保存**。

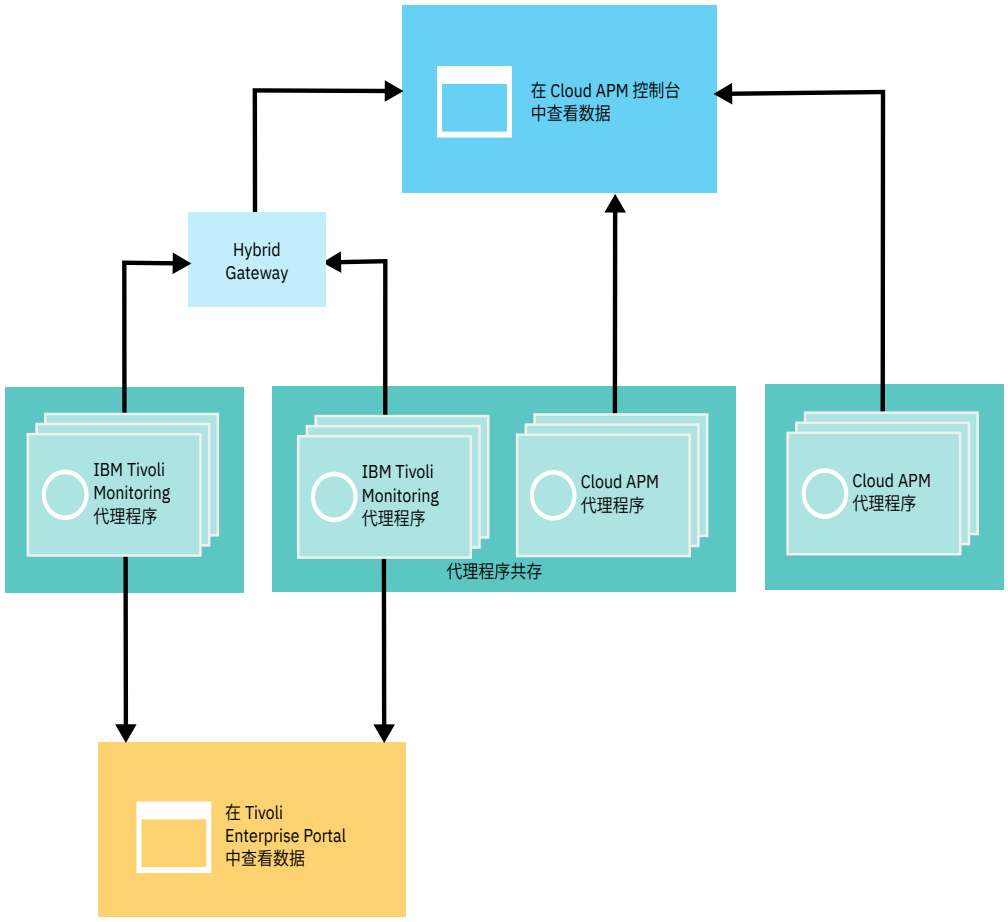
与 IBM Tivoli Monitoring V6.3 集成

在同时包含 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 产品的环境中，您可以通过多种方式共同使用这两种产品。

以下选项可用于与 IBM Tivoli Monitoring 集成：

- 您可以安装 IBM Cloud Application Performance Management Hybrid Gateway，以提供来自一个或多个 Tivoli Monitoring 域和 Cloud APM 域（应用程序性能仪表板 页面中）的受管系统的统一视图。 有关集成代理程序的更多信息，请参阅第 811 页的『Hybrid Gateway』。
- 您可以在同一系统上安装 Tivoli Monitoring 和 Cloud APM 代理程序。当代理程序共存于同一计算机上但位于不同目录中时，来自 Cloud APM 代理程序的数据在 Cloud APM 控制台中可用，来自 Tivoli Monitoring 代理程序的数据在 Tivoli Enterprise Portal 中可用。如果共存的代理程序正在监视相同资源，那么会应用一些限制。有关代理程序共存的更多信息，请参阅第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。

© Copyright IBM Corp. 2014, 2019



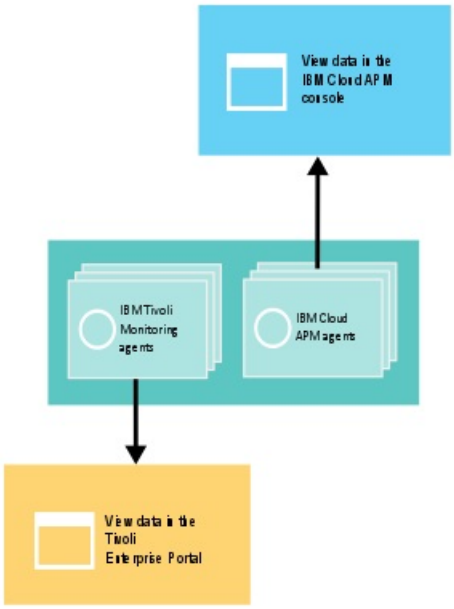
Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存

支持代理程序共存。您可以在安装 IBM Tivoli Monitoring 代理程序的同一计算机上安装 IBM Cloud Application Performance Management 代理程序。但是，这两种代理程序类型不能安装在同一目录中。

Cloud APM 代理程序指的是 V8 代理程序。Tivoli Monitoring 代理程序指的是 V6 或 V7 代理程序。

代理程序共存于同一台计算机上时，来自 V8 代理程序的数据可在 Cloud APM 控制台中使用，来自 V6 或 V7 代理程序的数据可在 Tivoli Enterprise Portal 中使用。

与 V8 代理程序共存于同一台计算机上并监视不同资源的 V6 或 V7 代理程序与 IBM Cloud Application Performance Management Hybrid Gateway 集成时，来自这两个代理程序的数据可在 Cloud APM 控制台中使用。有关更多信息，请参阅第 811 页的『Hybrid Gateway』。



下表列出了 Tivoli Monitoring 代理程序以及文档链接：

表 *236. Tivoli Monitoring* 代理程序的文档链接	
Tivoli Monitoring 代理程序	**IBM Knowledge Center 主题集合链接**
IBM Monitoring Agent for Citrix Virtual Desktop Infrastructure	IBM Tivoli Monitoring for Virtual Environments
IBM Tivoli Monitoring for Virtual Environments Agent for Cisco UCS	IBM Tivoli Monitoring for Virtual Environments
IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines	IBM Tivoli Monitoring for Virtual Environments
IBM Tivoli Monitoring for Virtual Environments Agent for VMware VI	IBM Tivoli Monitoring for Virtual Environments
IBM Tivoli Monitoring: HMC Base Agent	IBM Tivoli Monitoring
IBM Tivoli Monitoring： Linux OS Agent	IBM Tivoli Monitoring
IBM Tivoli Monitoring: UNIX OS 代理程序	IBM Tivoli Monitoring
IBM Tivoli Monitoring： Windows OS Agent	IBM Tivoli Monitoring
ITCAM Agent for Db2	ITCAM for Applications
ITCAM Agent for HTTP Servers	ITCAM for Applications
ITCAM Agent for J2EE	ITCAM for Applications
ITCAM for Microsoft Applications: Microsoft Active Directory Agent	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications： Microsoft Cluster Server 代理程序	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications： Microsoft Exchange Server 代理程序	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications： Microsoft Hyper-V Server 代理程序	ITCAM for Microsoft Applications

表 236. *Tivoli Monitoring* 代理程序的文档链接 *(续)*	
Tivoli Monitoring 代理程序	**IBM Knowledge Center 主题集合链接**
ITCAM for Microsoft Applications：Microsoft Internet Information Services 代理程序	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications：Skype for Business Server 代理程序	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications：Microsoft .NET Framework 代理程序	ITCAM for Microsoft Applications
ITCAM for Microsoft Applications：Microsoft SharePoint Server 代理程序	ITCAM for Microsoft Applications
Monitoring Agent for Microsoft SQL Server	ITCAM for Microsoft Applications
ITCAM Agent for SAP Applications	ITCAM for Applications
ITCAM Agent for WebSphere Applications	IBM Tivoli Composite Application Manager for Application Diagnostics（适用于 V7.1 和更低版本）以及 ITCAM for Applications（适用于 V7.2 和更高版本）。
ITCAM Agent for WebSphere DataPower Appliance	ITCAM for Applications
Monitoring Agent for WebSphere Message Broker	ITCAM for Applications
Monitoring Agent for WebSphere MQ	ITCAM for Applications Knowledge Center
ITCAM Extended Agent for Oracle Database	ITCAM for Applications
ITCAM Monitoring Agent for SAP HANA Database	ITCAM Monitoring Agent for SAP HANA Database 参考
ITCAM Web Response Time 代理程序	IBM Tivoli Composite Application Manager for Transactions

如果共存的代理程序正在监视相同资源，那么不支持以下场景：

- V6 或 V7 代理程序都与 Hybrid Gateway 集成，以在 Cloud APM 控制台上显示来自两个代理程序的数据。例如，如果 V6 或 V7 代理程序都通过 Hybrid Gateway 连接到同一 Cloud APM 服务器，请勿使用 V8 IBM Integration Bus 代理程序 和 V6 或 V7 Monitoring Agent for WebSphere Message Broker 来监视系统上的同一代理。

如果为在 Cloud APM 控制台中显示数据而与 Hybrid Gateway 集成的 Tivoli Monitoring 代理程序正在监视资源，并且您期望 Cloud APM 代理程序来监视该资源，请完成下列步骤：

1. 从包含 Tivoli Monitoring 代理程序的应用程序中将其移除。
2. 从 Tivoli Monitoring 管理的系统组中移除 Tivoli Monitoring 代理程序，Cloud APM 配置为使用该系统组。
3. 请至少等待 24 小时，然后安装 Cloud APM 代理程序，并将其添加到应用程序。

在同一个计算机上共存的多实例代理程序与 Hybrid Gateway 集成，并监视相同资源时，请对每个实例使用不同名称以在 Cloud APM 控制台中显示来自这两个代理程序的数据。

对于具有数据收集器的代理程序，支持相同类型的两个代理程序。 深入诊断、资源和事务跟踪数据显示在 Cloud APM 控制台上。资源数据显示在 Tivoli Enterprise Portal 上。以下代理程序共享一个数据收集器：

Monitoring Agent for HTTP Server

HTTP Server 代理程序是 Cloud APM 代理程序，ITCAM Agent for HTTP Servers 是 IBM Tivoli Monitoring 代理程序。如果环境中具有两个代理程序，那么您可以在同一 HTTP Server 上为两个代理程序配置数据收集器。 有关 HTTP Server 代理程序的更多信息，请参阅第 231 页的『配置 HTTP Server 监视』。

Microsoft .NET 代理程序

有关 Microsoft .NET 代理程序共存的更多信息，请参阅第 454 页的『在代理程序共存环境中启用事务跟踪』。

WebSphere Applications 代理程序

有关 WebSphere Applications 代理程序共存的更多信息，请参阅第 731 页的『配置 WebSphere Applications 代理程序』和第 732 页的『针对代理程序共存环境配置数据收集器』。

Hybrid Gateway

要在 Cloud APM 控制台中查看 IBM Tivoli Monitoring 和 OMEGAMON 代理程序的监视数据和事件，必须创建受管系统组并在 Tivoli Monitoring 域中安装 IBM Cloud Application Performance Management Hybrid Gateway，并在 Cloud APM 控制台“**Hybrid Gateway 管理器**”中配置通信。 查看背景信息以帮助您规划在 Tivoli Monitoring 和 Cloud APM 环境中安装和配置一个或多个 Hybrid Gateway。

安装 Hybrid Gateway 的位置

你可以在一个或多个 Tivoli Monitoring 域中安装 Hybrid Gateway：每个域一个中心 Tivoli Enterprise Monitoring Server。 有关 Hybrid Gateway 安装位置的详细信息，请参阅准备安装 Hybrid Gateway 信息。有关 Hybrid Gateway 系统需求，包括 Tivoli Enterprise Portal Server，请参阅 Hybrid Gateway 软件产品兼容性报告，**先决条件**选项卡。

受支持的 Tivoli Monitoring 和 OMEGAMON 代理程序

要使 Tivoli Monitoring 代理程序可用于 Hybrid Gateway，还必须在 Cloud APM，iOS 和 OMEGAMON 代理程序除外中支持该代理程序。有关可用 Tivoli Monitoring 代理程序和版本的列表，请参阅 Hybrid Gateway supported agents (APM Developer Center)。

要获取可在 Cloud APM 控制台中显示的 OMEGAMON 代理程序列表，请参阅 IBM Knowledge Center 上的 IBM OMEGAMON for Application Performance Management 主题集合中对应于您所使用的发行版的入门主题。

Cloud APM 控制台中的 Tivoli Monitoring 和 OMEGAMON 代理程序

在包含 Tivoli Monitoring 和/或 OMEGAMON 受管系统的 应用程序性能仪表板 中选择“我的组件” “” 预定义应用程序或或已定义应用程序后，您可以查看所有受管系统的摘要状态仪表板，并且可以查看单个受管系统实例的详细仪表板。您还可以在“**定制视图**”选项卡中创建仪表板页面

您可以在“**事件**”选项卡中查看这些代理程序的情景事件。但是，无法在阈值管理器中为 Tivoli Monitoring 和 OMEGAMON 代理程序创建新的阈值。改为在 Tivoli Monitoring 中创建新情景。

并非所有可能的 Tivoli Monitoring 和 OMEGAMON 事件都可在“事件”仪表板中使用。仅显示可以添加到应用程序的代理程序节点的事件。例如，对于 WebSphere Application Server 的 Tivoli Monitoring 代理程序，将显示与特定服务器实例关联的事件，但不显示来自代理程序的作为整体的事件。

查看来自每个 Tivoli Monitoring 域的多达 1500 个受管系统

可以查看来自 Tivoli Monitoring 域的受管系统的最大数量（包括子节点）为 1500。缺省情况下，该限制为 200 个系统。您可以进行规划以支持更多数量的系统。有关指示信息，请参阅第 818 页的『针对大量受管系统进行规划』。

所有 Tivoli Monitoring 域都必须限制在 Cloud APM 支持的最大值内。有关更多信息，请参阅第 37 页的『体系结构概述』。

仅限资源监视

资源监视可用于 Tivoli Monitoring 代理程序。有关资源监视的更多信息，请参阅第 39 页的『产品和附加组件』和第 46 页的『功能』。如果您有 IBM Cloud Application Performance Management, Advanced 预订，那么事务跟踪和诊断仪表板不可用于 Tivoli Monitoring 环境中的受管系统。

Tivoli Authorization Policy Server 影响 Tivoli Monitoring 受管系统的可用性

对于包含 Tivoli Authorization Policy Server 的 Tivoli Monitoring 环境，授权策略会影响通过 Hybrid Gateway 提供的受管系统。有关更多信息，请参阅 Tivoli Monitoring Knowledge Center 中的使用基于角色的授权策略。

有关视频演示，请参阅与 Tivoli Monitoring 集成 - Hybrid Gateway。

准备安装 Hybrid Gateway

要安装 IBM Cloud Application Performance Management Hybrid Gateway，必须先确保正确设置您的环境。复审信息以帮助您规划 Hybrid Gateway 安装。

安装 Hybrid Gateway 的位置

Hybrid Gateway 必须安装在具有与 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 的网络连接的 Red Hat Enterprise Linux V6.2（或更高版本）x86-64 系统上。

Hybrid Gateway 可以安装在与 Tivoli Enterprise Portal Server 相同的系统上，也可安装在与 Tivoli Enterprise Portal Server 不同的系统上，前提是这些系统都在 Red Hat Enterprise Linux 上运行。但是，Hybrid Gateway 无法安装在与 Cloud APM 服务器相同的系统上。

Tivoli Monitoring 域具有一个中心 Tivoli Enterprise Monitoring Server。如果 Tivoli Monitoring 环境由多个域组成，那么可以在多个域中安装 Hybrid Gateway。

要了解与 Hybrid Gateway 相关的系统需求，请单击 Hybrid Gateway Software Product Compatibility Report 中的**硬件**选项卡。

为 Hybrid Gateway 设置 Tivoli Enterprise Portal Server

对于门户网站服务器负载较重的 Tivoli Monitoring 环境，应安装单独的专用门户网站服务器来处理源自 Hybrid Gateway 的请求。如果设置单独的门户网站服务器：

- 如果门户网站服务器运行的是 Red Hat Enterprise Linux，可将同一台主机用于门户网站服务器和 Hybrid Gateway。
- 对于要在 Cloud APM 控制台中显示数据的代理程序，确保单独的门户网站服务器具有相应的应用程序支持。
- 确保 Tivoli Enterprise Portal 客户机未连接到单独的门户网站服务器，以完成创建定制工作空间、创建情境和创建受管系统组等管理任务。

Tivoli Enterprise Portal Server 必须为 V6.3 FP6 或更高版本。如果您的门户网站服务器版本较低，那么集成的 Tivoli Monitoring 代理程序可能无法在 Cloud APM 控制台中添加到应用程序。

必须在 Tivoli Enterprise Portal Server 上启用 IBM Tivoli Monitoring 仪表板数据提供程序。有关详细信息，请参阅 IBM Knowledge Center 上 IBM Tivoli Monitoring 主题集合中的验证仪表板数据提供程序是否已启用。

必须在 Hybrid Gateway 上打开 TCP 端口

在 Hybrid Gateway 上必须打开以下 TCP 端口。对于每个端口，一端发送请求，另一端提供响应。指示启动连接的一端。

- Hybrid Gateway 在端口 443 上与 Cloud APM 服务器进行单向连接，并发送 HTTPS 请求。

  如果 Hybrid Gateway 使用传递转发代理连接到 Cloud APM 服务器，请配置 Hybrid Gateway 以使用代理端口而非端口 443 来与 Cloud APM 服务器进行单向连接。有关指示信息，请参阅第 814 页的『使用转发代理与 Cloud APM 服务器进行通信』。
- 如果您使用 HTTP 与门户网站服务器进行通信，请打开端口 15200。如果您使用 HTTPS，请打开端口 15201。Hybrid Gateway 在端口 15200 或 15201 上启动与门户网站服务器的单向连接。要使用定制端口，请更新**门户网站服务器端口**设置的值。 有关更多信息，请参阅第 819 页的『Hybrid Gateway Manager』。

  另外，如果 Hybrid Gateway 使用传递转发代理来连接到门户网站服务器，请配置 Hybrid Gateway，以使用代理端口与门户网站服务器进行单向连接。设置**传递代理端口**设置的值。 有关更多信息，请参阅第 819 页的『Hybrid Gateway Manager』。
- 对于要侦听来自 Tivoli Enterprise Monitoring Server 的入站 EIF 事件的 Hybrid Gateway，请打开端口 9998。监视服务器将在端口 9998 上启动与 Hybrid Gateway 的单向连接。如果该端口未打开，安装实用程序将显示警告。

运行 Hybrid Gateway 安装脚本所需的 root 用户特权

您必须使用 root 用户特权来运行 Hybrid Gateway 安装脚本。有关受支持的操作系统的完整列表，请参阅 System requirements (APM Developer Center)

安装 Hybrid Gateway

下载并安装 IBM Cloud Application Performance Management Hybrid Gateway 以在 Cloud APM 控制台中查看来自 IBM Tivoli Monitoring 域的受管系统。

开始之前

请查看并完成准备安装 Hybrid Gateway 中的必需准备任务。

过程

完成以下步骤以在 Tivoli Monitoring 域中安装 Hybrid Gateway：

1. 下载 Hybrid Gateway 程序包。

   APM_Hybrid_Gateway_Install.tar 文件包含 Hybrid Gateway 以及安装脚本。

   a) 登录到您的帐户，然后转至 IBM Marketplace 上的产品和服务。

   b) 在 **IBM Cloud APM** 下，单击**更多操作**。

   c) 单击**显示更多页面**。

   d) 选择 **Hybrid Gateway**。如果需要，请向下滚动以查找此条目。

   e) 单击**下载**。
2. 如果需要，请将该文件传输至将运行 Hybrid Gateway 的系统。
3. 输入以下命令以解压缩其中的文件：

   ```
   tar -xf APM_Hybrid_Gateway_Install.tar
   ```

   归档文件包含用于部署 Hybrid Gateway 的脚本。安装脚本将解压缩到目录，而 Hybrid Gateway 文件将解压缩到子目录。
4. 切换到 Hybrid Gateway 目录，并使用 root 用户特权运行安装脚本：

   ```
   cd APM_Hybrid_Gateway_Install_version
   ./install.sh
   ```

   其中，*version* 是当前版本，例如 8.1.4.0。

   然后将针对您的环境启动先决条件扫描，该扫描可能需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。诸如磁盘空间不足等先决条件会停止安装。您必须解决故障，然后重新启动安装。您还可以如绕过先决条件扫描程序所述关闭先决条件检查。
5. 系统通过先决条件扫描之后，通过选择 1（表示接受）响应接受许可协议的提示。

   然后，一条消息指示您登录 Cloud APM 控制台并配置 Hybrid Gateway，然后才能继续。还将显示从主机名衍生的缺省概要文件名称。
6. 请按 Enter 键以接受缺省名称或输入概要文件名称。

   如果已针对此 Tivoli Monitoring 域创建了概要文件，那么使用 Hybrid Gateway 管理器中提供的相同名称。如果尚未创建概要文件，那么可以接受缺省名称或提供新名称，但请确保跟踪名称，因为稍后在创建概要文件时必须使用此名称。（请参阅第 816 页的『使用 Cloud APM 控制台配置 Hybrid Gateway』。）

   按 Enter 键之后，Hybrid Gateway 安装继续。

结果

Hybrid Gateway 安装在 /opt/ibm/hybridgateway 目录中，并会自动启动。安装日志文件位于 /opt/ibm/hybridgateway/logs/install-hybridgateway-*timestamp*.log。Hybrid Gateway 日志文件在 /opt/ibm/wlp/usr/servers/hybridgateway/logs 目录中。 请注意，在配置 Tivoli Enterprise Portal Server 的连接之后，才会记录连接失败。

下一步做什么

- 可以将 Hybrid Gateway 配置成使用转发代理与 Cloud APM 服务器进行通信。有关指示信息，请参阅第 814 页的『使用转发代理与 Cloud APM 服务器进行通信』。

- 您可以使用以下命令检查 Hybrid Gateway 的状态：*install_dir*/hybridgateway/bin/hybridgateway.sh status。有关更多选项，请参阅第 819 页的『管理 Hybrid Gateway』。
- 如果您已为 Hybrid Gateway 创建受管系统组，请遵循第 814 页的『创建受管系统组』中的指示信息。
- 如果您尚未针对 Tivoli Monitoring 域创建 Hybrid Gateway 概要文件，请遵循第 816 页的『使用 Cloud APM 控制台配置 Hybrid Gateway』中的指示信息。
- 如果 Tivoli Monitoring 环境具有多个中心域，那么可以在其他域中安装 Hybrid Gateway。重复此过程中的步骤以在其他 Tivoli Monitoring 域中安装 Hybrid Gateway。

使用转发代理与 Cloud APM 服务器进行通信

可以将 IBM Cloud Application Performance Management Hybrid Gateway 配置成使用转发代理与 Cloud APM 服务器进行通信。

过程

1. 在安装了 Hybrid Gateway 的主机上，编辑 /opt/ibm/wlp/usr/servers/hybridgateway/bootstrap.properties 文件：
   - 如果 Hybrid Gateway 使用 HTTP 与 Cloud APM 服务器进行通信，那么请添加行：

     ```
     http.proxyHost=proxy_host
     http.proxyPort=proxy_port
     ```

   - 如果 Hybrid Gateway 使用 HTTPS 与 Cloud APM 服务器进行通信，那么请添加行：

     ```
     https.proxyHost=proxy_host
     https.proxyPort=proxy_port
     ```

   其中，*proxy_host* 是代理的主机名或 IP 地址，可以从 Hybrid Gateway 主机上访问，*proxy_port* 是代理的端口。
2. 重新启动 Hybrid Gateway。

创建受管系统组

在 Tivoli Enterprise Portal 客户机中使用对象组编辑器创建想要在 Cloud APM 控制台中查看的受管系统的受管系统组。

开始之前

- 可以包括在受管系统组中的 IBM Tivoli Monitoring 及 OMEGAMON 代理程序类型必须是其中一种受支持的代理程序。例如，对于 Tivoli Monitoring，Monitoring Agent for Oracle Database 和 Monitoring Agent for Linux OS 是受支持的代理程序。

  有关受支持的 Tivoli Monitoring 代理程序的最新列表，请参阅 Hybrid Gateway supported agents (APM Developer Center)。要获取可在 Cloud APM 控制台中显示的 OMEGAMON 代理程序列表，请参阅 IBM Knowledge Center 上的 IBM OMEGAMON for Application Performance Management 主题集合中的“入门”主题。
- Tivoli Monitoring 和 OMEGAMON 代理程序必须连接至同一个 IBM Tivoli Monitoring 基础结构。如果您的环境具有多个 Tivoli Monitoring 域，那么为安装了 Hybrid Gateway 的每个中心 Tivoli Enterprise Monitoring Server 创建受管系统组。
- 缺省情况下，您最多可以向受管系统组添加 200 个受管系统以在应用程序性能仪表板中从 Tivoli Monitoring 域查看。您可以通过执行若干计划步骤，将此限制增大到多达 1500 个系统。有关更多信息，请参阅第 818 页的『针对大量受管系统进行规划』。如果您有多个 Hybrid Gateway 用于具有多个中心的 Tivoli Monitoring 环境，那么每个域的受管系统组必须在 Cloud APM 支持的最大值内。有关更多信息，请参阅体系结构概述。
- 在 2017 年 3 月的 Cloud APM 发行版中，已更改处理子节点的缺省方式。在前发行版中，如果存在具有子节点的代理程序，例如 WebSphere Applications 代理程序，那么必须将管理节点分配给受管系统组，而所有子节点将自动包括在内。虽然您将一个管理节点分配给受管系统组，但是任何子节点都会包括在计数中，直至达到受管系统的最大数目为止。

在 2017 年 3 月的 Cloud APM 发行版和更高版本中，要在 Cloud APM 控制台 中显示度量数据的子节点必须明确分配给受管系统组。如果管理代理程序的任何子节点明确分配给受管系统组，那么将自动发现管理代理程序。对于基于子节点的监视应用程序，Cloud APM 可能需要查询管理代理程序，以获取明确识别 Cloud APM 仪表板导航器中显示的监视资源所需的信息。由于这个原因，使用当前版本的发现方式，并且至少有一个关联的子节点分配给配置以供 Hybrid Gateway 使用的受管系统组时，会自动包括管理代理程序。当前的发现方式支持精确控制可在 Cloud APM 控制台中查看的子节点资源，而且更加符合 Cloud APM 应用程序的构造方式，对于涉及大量子节点资源实例的应用程序尤其如此。将 OMEGAMON 代理程序与 Cloud APM 集成时，请务必使用缺省当前发现方式。

您可以对外部属性 MSN_DISCOVERY_MODE 指定适当的值（Hybrid Gateway 将在初始化期间处理此属性），以指定 Hybrid Gateway 所使用的发现方式版本。要控制 Hybrid Gateway 使用何种发现方式，请在已安装 Hybrid Gateway 的系统上的下列属性文件中添加 MSN_DISCOVERY_MODE 属性（或更改其当前值），然后重新启动 Hybrid Gateway。

```
HG_install_dir/wlp/usr/servers/hybridgateway/bootstrap.properties
```

MSN_DISCOVERY_MODE 属性的可能值为：

- MSN_DISCOVERY_MODE=1 将强制 Hybrid Gateway 使用原始代理程序发现方式，在此方式下，对于分配给 Tivoli Monitoring 受管系统组的任何管理代理程序，将自动发现所有子节点。
- MSN_DISCOVERY_MODE=2 将强制 Hybrid Gateway 使用缺省的新代理程序发现方式，在此方式下，Hybrid Gateway 将仅查询明确分配给受管系统组的子节点。系统将自动发现一个或多个相关联的管理代理程序。

· 如果您更希望使用 **tacmd createsystemlist** 和 **tacmd editsystemlist** IBM Tivoli Monitoring 命令创建受管系统组，请参阅 *IBM Tivoli Monitoring* 命令参考 (https://www.ibm.com/support/knowledgecenter/SSTFXA_6.3.0/com.ibm.itm.doc_6.3/cmdref/itm_cmdref.htm) 以获取有关如何运行命令的信息。

过程

完成以下步骤以在 Tivoli Enterprise Portal 客户机中创建受管系统组。

1. 使用具有所有受管系统类型的完整访问权的用户标识和密码启动 Tivoli Enterprise Portal 客户机（“**允许的应用程序**”设置为用户标识的“**所有应用程序**”。）
2. 单击 **对象组编辑器**。
3. 展开 **受管系统**对象，并选择 **所有受管系统**以在受管系统组中组合多个代理程序类型（例如，Windows 操作系统和 Oracle）。如果希望受管系统组仅包含一个监视代理程序类型（例如，Linux 操作系统或 WebSphere Applications），请选择该代理程序类型。
4. 单击 **创建新组**，并输入受管系统组的名称。

   名称可以包含字母和数字，但不得包含空格、标点或除下划线 (_) 之外的特殊字符。

   在单击**确定**后，将在受管系统文件夹中显示新的受管系统组。
5. 从**可用受管系统**列表中选择受管系统，并通过 将它们移至**已分配**列表。

   您可以通过在按住 Ctrl 键的同时单击各受管系统来选中多个受管系统。在选中一个受管系统之后，可以通过按住 Shift 键并单击来选中此选择项与第一个选择项之间的所有受管系统。
6. 将受管系统添加到组之后，请单击**确定**以保存更改并关闭对象组编辑器。

下一步做什么

· 在创建受管系统组并安装 Hybrid Gateway 之后，必须在 Cloud APM 控制台中配置 Hybrid Gateway。

· 对于配置，指定创建的受管系统组名、允许访问所有代理程序类型的 Tivoli Enterprise Portal 用户标识以及 Tivoli Enterprise Portal Server 的主机名和端口。

· 有关安装指示信息，请参阅第 813 页的『安装 Hybrid Gateway』。

· 有关配置指示信息，请参阅第 816 页的『使用 Cloud APM 控制台配置 Hybrid Gateway』。

使用 Cloud APM 控制台配置 Hybrid Gateway

使用 Cloud APM 控制台中的 **Hybrid Gateway 管理器**页面来配置 IBM Cloud Application Performance Management Hybrid Gateway，以连接 Tivoli Enterprise Portal Server 并指定受管系统组。您可以为环境中的每个中心 Tivoli Enterprise Monitoring Server 创建 Hybrid Gateway 概要文件。

过程

请完成以下步骤，以在 Cloud APM 控制台中配置 Hybrid Gateway。

1. 如果您尚未登录 Cloud APM 控制台，请立即登录。
   （请参阅第 829 页的『启动 Cloud APM 控制台』。）
2. 单击 **系统配置** > **Hybrid Gateway Manager**。
   该页面显示为 Tivoli Monitoring 域配置的任何 Hybrid Gateway 的表。如果显示空白名称的概要文件，那么是在 2017 年 8 月发行版之前安装的 Hybrid Gateway。有关更多信息，请参阅第 819 页的『概要文件名称』。
3. 单击 **添加**以打开“**添加 Hybrid Gateway**”窗口，在**概要文件名称**字段中输入新的名称，并单击**添加**。
   如果已在 Tivoli Monitoring 域上安装了 Hybrid Gateway，那么确保使用在 Hybrid Gateway 安装期间提供或接受的同一名称。在安装期间输入的概要文件名称和在此处输入的名称必须完全匹配。
   打开“**编辑 Hybrid Gateway**”窗口。
4. 在**受管系统组名称**字段中，输入 Hybrid Gateway 的受管系统组的名称。
   这是在第 814 页的『创建受管系统组』中使用的名称。
5. 指定 Tivoli Enterprise Portal Server 的地址、端口和 Web 通信协议：

选项	描述
门户网站服务器主机名	输入门户网站服务器主机 IP 地址或标准主机名称或域名。
门户网站服务器端口	输入门户网站服务器用于 Web 通信的端口号。缺省端口为 15200（针对 HTTP）或 15201（针对 HTTPS）。值 0 将端口设置为缺省值 15200（针对 HTTP）或 15201（针对 HTTPS）。
门户网站服务器协议	选择 **HTTP** 因特网协议或安全 **HTTPS** 因特网协议以连接到门户网站服务器。

6. 使用用于启动 Tivoli Enterprise Portal 客户机的登录用户名和对应密码填写**门户网站服务器用户名**和**门户网站服务器密码**字段。
   用户标识必须具有所有监视代理程序类型的访问权（“**允许的应用程序**”设置为“**所有应用程序**”），如 `sysadmin` 标识。有关更多信息，请参阅 Tivoli Monitoring IBM Knowledge Center 中的管理用户。
7. 如果对门户网站服务器的访问要经过代理服务器，请指定地址、端口和 Web 协议：

选项	描述
传递代理主机名	输入代理主机系统的 IP 地址或标准名称。
传递代理端口	输入代理主机系统的端口号。
传递代理协议	选择用于通过代理进行通信的协议：HTTP 或 HTTPS

结果

在单击**保存**后，将建立与 Hybrid Gateway 服务的连接，且发现 Tivoli Monitoring 域的受管系统。对于资源监视数据，会每 5 分钟对受管系统组轮询一次。

下一步做什么

- 您必须配置 Tivoli Monitoring 以便与 Cloud APM 交互。有关指示信息，请参阅第 817 页的『配置 Tivoli Monitoring 以与 Cloud APM 集成』。

· 您可以重复这些步骤，以针对要从 Cloud APM 监视受管系统的每个 Tivoli Monitoring 域添加一个概要文件。

· 您可以使用“**Hybrid Gateway 管理器**”工具管理现有概要文件：

选择 Hybrid Gateway，然后单击 **编辑**以打开“**编辑 Hybrid Gateway**”窗口。

选择不再需要的 Hybrid Gateway，然后单击 **删除**。在确认要删除 Hybrid Gateway 后，永久移除概要文件。

单击列标题以按该列对表进行排序；按住 Ctrl 键并单击其他列以添加次要排序。

在过滤器文本框  中单击鼠标，然后输入要作为过滤依据的值的开头。输入时，将过滤掉不适合条件的行。要清除过滤器，请单击过滤器框  中的 ×，或者按退格键。

配置 Tivoli Monitoring 以与 Cloud APM 集成

要将 IBM Tivoli Monitoring 域与 Cloud APM 集成，必须完成配置 Tivoli Enterprise Portal Server 和配置中心 Tivoli Enterprise Monitoring Server 之类的任务。

过程

对于安装了 Hybrid Gateway 的每个 Tivoli Monitoring 域，请完成以下步骤：

1. 配置 Tivoli Enterprise Portal Server 以启用仪表板数据提供程序。

   需要数据提供程序才能将 Cloud APM 与 Hybrid Gateway 集成。有关指示信息，请参阅以下主题：

   · Windows Windows：安装门户网站服务器（步骤 16c）。

   · Linux AIX 在 Linux 或 AIX 上配置门户网站服务器：命令行过程（步骤 14）。

   如果使用了热备用功能，必须指定域覆盖。Hybrid Gateway 使用域名从两个中心监视服务器收集数据，而不考虑门户网站服务器连接的是哪个中心。

2. 如果想要在 Cloud APM 控制台 中查看来自 Tivoli Monitoring 代理程序的情境事件，请针对以下每种场景配置中心监视服务器：

   · 将事件仅发送到 Hybrid Gateway。

   · 将事件发送到 Hybrid Gateway 和其他 EIF 接收方，例如 Netcool/OMNIbus 服务器。

   根据适用的场景完成以下步骤之一：

   a) 要配置中心 Tivoli Enterprise Monitoring Server 以将事件仅发送到 Hybrid Gateway，完成配置中心监视服务器以转发事件主题中的步骤。

      为 **ServerPort** 参数指定端口号 9998。

      Cloud APM 不会将 Tivoli Monitoring 事件转发到 Netcool/OMNIbus。如果要在 Cloud APM 和 Netcool/OMNIbus 中查看 Tivoli Monitoring 事件，必须配置 Tivoli Monitoring 以将事件发送到这两个系统。

   b) 要配置中心 Tivoli Enterprise Monitoring Server 以将事件转发到 Hybrid Gateway 和另一个 EIF 接收方（例如 Netcool/OMNIbus 服务器），请使用配置中心监视服务器以转发事件主题中的步骤配置缺省 EIF 接收方。

      该主题还提供了有关使用 **tacmd createEventDest** 命令创建其他 EIF 目标的信息。指定端口 9998 作为 Hybrid Gateway 目标的 EIF 端口号。

   c) 配置针对 Hybrid Gateway 受管系统组中的代理程序而存在的任何情境，以确保情境事件发送到 Hybrid Gateway 的 EIF 目标。有关指示信息，请参阅指定哪些情境将事件转发至 Netcool/OMNIbus 主题。

下一步做什么

查看应用程序性能仪表板以确认 Tivoli Monitoring 域中的受管系统通过 Hybrid Gateway 传递：

1. 单击 **性能** > **应用程序性能仪表板**，以打开**我的所有应用程序**仪表板。
2. 在“我的组件”摘要框中“”，单击 **组件**以打开组件所管理的所有系统的状态摘要仪表板（WebSphere Applications 代理程序除外）。如果没有“我的组件”应用程序“”，请按第 930 页的『管理应用程序』中所述添加应用程序。

3. 查找来自 Tivoli Monitoring 域的受管系统，这类受管系统通过状态摘要组窗口小部件标题中的 **ITM** (IBM Tivoli Monitoring) 域图标指示。如果缺少任何受管系统，请转至 Cloud Application Performance Management 论坛，并搜索“Hybrid Gateway” “”。

您可以通过 Tivoli Monitoring 域中的受管系统创建应用程序，还可以包含 Cloud APM 域中的受管系统。有关更多信息，请参阅第 930 页的『管理应用程序』。

针对大量受管系统进行规划

您可以在 IBM Tivoli Monitoring 域中查看的最大受管系统数量为 1500。如果包含在您为 Hybrid Gateway 概要文件创建的受管系统组中具有子节点的代理程序，那么所有子节点以及该代理程序都计算在内，直到达到限制。缺省情况下，此限制为 200 个受管系统，但您可以执行一些规划步骤来提高此限制。所有 Tivoli Monitoring 域的限制都必须在 Cloud APM 支持的最大值之内。有关更多信息，请参阅第 37 页的『体系结构概述』。

- 将 Tivoli Enterprise Portal Server **KFW_REPORT_NODE_LIMIT** 环境变量的值设置为大于等于 Hybrid Gateway 受管系统数量的数字。缺省值为 200。有关指示信息，请参阅 Tivoli Enterprise Portal Server 配置设置。
  如果受管系统超过该设置，那么 Tivoli Enterprise Portal Server `KfwServices` 消息日志显示一条类似以下示例的消息：
  ```
  56C6246F.0000-10:ctreportmanager.cpp,2864,"CTReport::Manager::executeDefinitionDual") Query is targeting 1497 nodes which exceeds the current limit of 200 nodes.
  ```
- 如果在查看大量系统，根据代理程序类型、Hybrid Gateway 和受管系统之间的网络等待时间、每个代理程序监视的环境大小（收集和发布的数据量），可能会导致性能下降。为避免此影响，请仅选择提供必要数据的代理程序，并确保受监视系统和 Hybrid Gateway 主机之间具有较快的网络连接。
- 随着网络等待时间的延长，从给定数量代理程序收集数据的时间也会增加。Hybrid Gateway 尝试每 5 分钟从每个代理程序收集一次数据。如果从所有代理程序收集数据的时间超过 5 分钟，那么 Hybrid Gateway 将缺失数据样本，因此某些受管系统在 应用程序性能仪表板 页面上没有可用的度量值。
- 为抵消非常慢的网络速度带来的影响，您可以尝试增加 Hybrid Gateway 用于收集数据样本的线程数量。Hybrid Gateway `bootstrap.properties` 文件的 **MAX_COLLECTOR_THREADS** 参数用于控制线程数量。缺省值为 50。

卸载 Hybrid Gateway

如果不再希望在 Cloud APM 控制台中查看 IBM Tivoli Monitoring 受管系统，请卸载 IBM Cloud Application Performance Management Hybrid Gateway。

过程

1. 在 Hybrid Gateway *install_dir*/hybridgateway/bin 目录（例如，/opt/ibm/hybridgateway/bin）中，运行以下命令：
   ```
   ./hybridgateway.sh uninstall
   ```
   将移除 Hybrid Gateway，并有消息确认已成功将其卸载。如果您在包含混合代理程序的 Cloud APM 控制台中有任何应用程序，那么混合代理程序会继续显示，直到监视基础结构处理其删除。
2. 在 Cloud APM 控制台中，单击 **系统配置** > **Hybrid Gateway Manager**。
3. 选择不再需要的 Hybrid Gateway 概要文件，然后单击 **删除**。
   在确认要删除 Hybrid Gateway 后，永久移除概要文件。

下一步做什么

- 要在 Cloud APM 控制台中从应用程序移除任何混合代理程序受管系统，请遵循第 930 页的『管理应用程序』中有关编辑应用程序的指示信息。
- 如果并未成功移除软件，而是您收到类似于以下示例中所示的错误消息，那么请查看日志文件以查找可能的原因：
  ```
  error: Failed dependencies:
  ibm-java-x86_64-jre is needed by (installed) smai-kafka-00.08.00.00-1.el6.x86_64
  ```

```
Uninstallation failed. The uninstaller was unable to remove some of the components, please
inspect the
 log file ("/tmp/hybridgateway/logs/uninstall-hybridgateway-20150228080551.log") for more
information.
```

发生示例中所示的错误是因为系统上外部安装的某个程序包需要 ibm-java-x86_64-jre。安装程序不会移除该 JRE，因为这样做可能会导致其他程序包无法运作。变通方法是先卸载依赖 ibm-java-x86-64-jre 的产品，然后再卸载 Hybrid Gateway。

管理 Hybrid Gateway

使用可用于 IBM Cloud Application Performance Management Hybrid Gateway 服务的命令以启动或停止该服务、检查状态、卸载 Hybrid Gateway，以及收集日志文件（如果 IBM 支持人员指示进行收集）。

关于此任务

这些步骤假定 Hybrid Gateway 安装目录为 `/opt/ibm/`。在安装 Hybrid Gateway 的系统上，从命令提示符执行以下任何步骤：

过程

- 要启动 Hybrid Gateway 服务，请输入 **/opt/ibm/hybridgateway/bin/hybridgateway.sh start**。
- 要停止 Hybrid Gateway 服务，请输入 **/opt/ibm/hybridgateway/bin/hybridgateway.sh stop**。
- 要检查 Hybrid Gateway 服务的状态，请输入 **/opt/ibm/hybridgateway/bin/hybridgateway.sh status**。
- 要卸载 Hybrid Gateway，请输入 **/opt/ibm/hybridgateway/bin/hybridgateway.sh uninstall**。

  另请参阅第 818 页的『卸载 Hybrid Gateway』。
- 要检查 Hybrid Gateway 日志文件，请转至 `/opt/ibm/wlp/usr/servers/hybridgateway/logs`。
- 要收集 Hybrid Gateway 日志文件以提供给 IBM 支持人员，请输入 **/opt/ibm/hybridgateway/collectLogs.sh**。

  这将收集日志文件，并有消息显示压缩日志文件的位置，且要求您将它们返回给 IBM 支持人员。

Hybrid Gateway Manager

配置 IBM Cloud Application Performance Management Hybrid Gateway 以在 Cloud APM 控制台中查看来自 IBM Tivoli Monitoring 域的监视数据。您可以为您的环境中的每个中心 Tivoli Enterprise Monitoring Server 创建 Hybrid Gateway 概要文件。

单击 **系统配置** > **Hybrid Gateway Manager** 后，将显示包含已定义的 Hybrid Gateway 列表的页面。

此页面包含针对 Tivoli Monitoring 域配置的所有混合网关的表，并且包含用于管理 Hybrid Gateway 概要文件的工具：

- **添加**打开“**添加混合网关**”窗口来命名新概要文件。在输入名称并单击**添加**后，“**编辑混合网关**”窗口随即打开。
- 选择混合网关，然后单击 **编辑**以打开“**编辑混合网关**”窗口。
- 选择不再需要的混合网关，然后单击 **删除**。在确认要删除混合网关后，永久移除概要文件。
- 单击列标题以按该列对表进行排序；按住 Ctrl 键并单击其他列以添加次要排序。
- 在过滤器文本框 中单击鼠标，然后输入要作为过滤依据的值的开头。输入时，将过滤掉不适合条件的行。要清除过滤器，请单击过滤器框 中的 ×，或者按退格键。

配置 Hybrid Gateway 时必须填充的必填字段在“**编辑混合网关**”窗口中带有星号 (*) 标记。

概要文件名称

Hybrid Gateway 概要文件的给定名称，最大长度为 128 个字符、数字和下划线 (_)。

在 Hybrid Gateway 安装期间请求概要文件名称。如果您已在 Tivoli Monitoring 域上安装 Hybrid Gateway，请使用在 Hybrid Gateway 安装期间提供或接受的名称。

较旧版本的 Hybrid Gateway 不使用已命名的概要文件来访问其配置数据。如果在 Cloud APM 2017 年 8 月发行版之前安装了 Hybrid Gateway，那么将获得一个特殊的未命名（空白）的概要文件名称。只允许 Hybrid Gateway 的一个较旧版本连接到 Cloud APM 服务器。 如果您已配置较低版本的 Hybrid Gateway，那么未命名的概要文件显示已配置的值。如果您未配置较低版本的 Hybrid Gateway，那么未命名的概要文件显示缺省值。您可以保留未命名的概要文件，或者将其删除并在稍后根据需要重新添加，它只能用于 2017 年 3 月（或更低）版本的 Hybrid Gateway。

受管系统组名
为在 Cloud APM 控制台中查看受支持的监视代理程序所创建的 Tivoli Enterprise Portal Server 受管系统组。控制台中不会显示 Cloud APM 产品不支持的任何监视代理程序类型，无论它们是否包含在受管系统组中。
有关创建受管系统组以获取混合支持的指南和限制，请参阅。

门户网站服务器主机名
Tivoli Enterprise Portal Server 主机 IP 地址或标准域名。

门户网站服务器端口
Tivoli Enterprise Portal Server 用于通信的端口号。缺省端口是 `15200`（适用于 HTTP）或 `15201`（适用于 HTTPS）。值 0 将端口设置为缺省值 `15200`（适用于 HTTP）或 `15201`（适用于 HTTPS）。

门户网站服务器协议
确定是使用 HTTP 因特网协议还是安全 HTTPS 协议连接到 Tivoli Enterprise Portal Server。缺省值：`http`。

门户网站服务器用户名
用于启动 Tivoli Enterprise Portal 客户机的用户名。此用户标识必须具有所有监视代理程序类型的访问权（“**允许的应用程序**”设置为“**所有应用程序**”）。有关更多信息，请参阅 Tivoli Monitoring Knowledge Center 中的管理用户。）

门户网站服务器用户密码
与 Tivoli Enterprise Portal 登录用户名关联的密码。

传递代理主机名
如果 Tivoli Enterprise Portal Server 通过传递代理服务器进行通信，则使用。输入代理主机系统的 IP 地址或标准名称。

传递代理端口
如果 Tivoli Enterprise Portal Server 通过传递代理服务器进行通信，则使用。输入用于与代理通信的端口号。

传递代理协议
如果 Tivoli Enterprise Portal Server 通过传递代理服务器进行通信，则使用。输入用于通过代理通信的协议。缺省值：`http`。

在 Cloud APM 控制台中查看的 Tivoli Monitoring 代理程序位于 IBM Tivoli Monitoring 环境中。您可以在 应用程序性能仪表板 页面中对其进行查看，但是不能在“**阈值管理器**”中为这些代理程序创建阈值。

与 OMEGAMON 集成

您可通过购买 z Systems Extension Pack，并使用 Hybrid Gateway 将一个或多个已部署的 OMEGAMON 代理程序连接至 Cloud APM，从而在 Cloud APM 控制台中查看 OMEGAMON 应用程序组件的数据和事件。

开始之前

- 要使用 z Systems Extension Pack，您必须要有 IBM Cloud Application Performance Management, Advanced 或 IBM Cloud Application Performance Management, Base 产品。
- 在所监视的 z Systems LPAR 上，必须有一个或多个已获授权的 OMEGAMON 代理程序在运行。
- OMEGAMON 代理程序会连接至 IBM Tivoli Monitoring 基础结构。

要获取可在 Cloud APM 控制台中显示的 OMEGAMON 代理程序列表，请参阅 IBM Knowledge Center 上的 IBM OMEGAMON for Application Performance Management 主题集合中对应于您所使用的发行版的入门主题。

过程

要将 OMEGAMON 与 Cloud APM 集成，请完成下列步骤：

1. 将 z Systems Extension Pack 添加到 Cloud APM 产品后，，请完成下列 Hybrid Gateway 任务：
   a) 安装 Hybrid Gateway。
   b) 创建您要在 Cloud APM 控制台中查看的受管系统组。
   c) 在 Cloud APM 控制台中配置 Hybrid Gateway，以便将 Hybrid Gateway 连接至 Tivoli Enterprise Portal Server，并指定受管系统组。

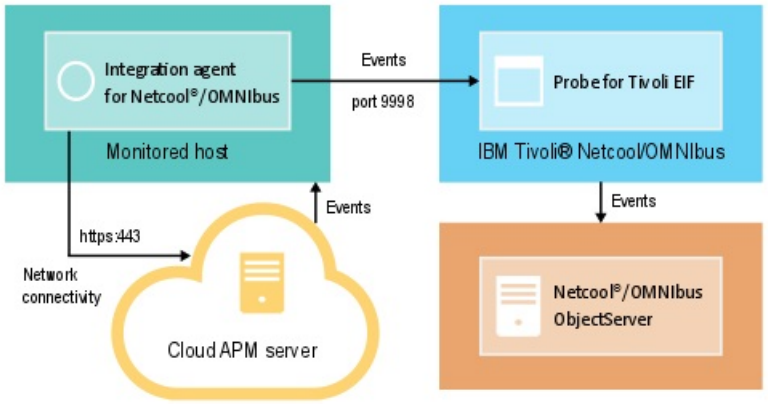
   有关更多信息，请参阅 第 811 页的『Hybrid Gateway』 部分中的所需主题。
2. 要在仪表板中查看应用程序的状态，请从浏览器登录 Cloud APM 控制台。有关更多信息，请参阅第 829 页的『启动 Cloud APM 控制台』。

与 Netcool/OMNIbus 集成

您可以将 IBM Cloud Application Performance Management 中的事件转发到内部部署 IBM Tivoli Netcool/OMNIbus 事件管理器。

过程

1. 要查看 Integration Agent for Netcool/OMNIbus，以及它如何在 Cloud APM 中与 Probe for Tivoli EIF 集成以将事件转发到 Netcool/OMNIbus，请参阅以下配置：



   Integration Agent for Netcool/OMNIbus 自动连接到 Cloud APM 服务器。该连接允许事件从服务器流向网络，而不需要任何入站网络连接。
2. 为 Netcool/OMNIbus 配置集成。

安装和配置 Integration Agent for Netcool/OMNIbus

要安装 Integration Agent for Netcool/OMNIbus，必须从 IBM Marketplace Web 站 点下载归档文件，解压缩代理程序安装文件，然后启动安装脚本。在安装后，代理程序将自动启动，但必须进行配置。

关于此任务

一次只能有一个 Integration Agent for Netcool/OMNIbus 实例可以将事件从 Cloud APM 服务预订的单个实例转发到 Netcool/OMNIbus 事件管理器。

过程

1. 下载 Cloud APM 集成归档文件，该文件包含 Integration Agent for Netcool/OMNIbus：
   a) 登录到您的帐户，然后转至 IBM Marketplace 上的产品和服务。
   b) 在 **IBM Performance Management** 下，单击**更多操作**。
   c) 单击**显示更多页面**。

d) 选择 **IBM Performance Management OMNIbus Integration on Cloud**。如果需要，请向下滚动以查找此项。

e) 单击**下载**。

2. 将文件保存到所选暂存目录。在具有 Tivoli Netcool/OMNIbus Probe for Tivoli Event Integration Facility (EIF) 的网络连接的任何系统上安装代理程序。如果需要，将安装归档文件传输至要监视的系统。归档文件包含代理程序和安装脚本。
3. 解压缩安装文件：

Linux

a. 在 Red Hat Enterprise Linux 系统上打开终端 shell 会话。

b. 转至归档文件所在的目录。

c. 通过使用以下命令解压缩安装文件：

```
tar -xf ./apm_integration_agents_xlinux_8.1.4.0.tar
```

Windows

解压缩 `apm_integration_agents_win_8.1.4.0.zip` 文件。

安装脚本解压缩到以归档文件和版本命名的目录。例如，`IPM_Agent_Install_8.1.3.2`。代理程序二进制和配置相关文件解压缩至该目录中的子目录。

4. 使用管理员特权从以归档文件和版本命名的目录运行安装脚本。

如果您要安装 Integration Agent for Netcool/OMNIbus 的系统与您的 Probe for Tivoli EIF 所在的系统相同，且 Probe for Tivoli EIF 正在使用缺省端口 9998，那么 Integration Agent for Netcool/OMNIbus 会自动配置为连接到您的 Probe for Tivoli EIF。

要点: 如果在不同于 Probe for Tivoli EIF 所在的系统上安装 Integration Agent for Netcool/OMNIbus，或者如果使用不用于 Probe for Tivoli EIF 的缺省值的端口号，那么在安装完成后必须配置 Integration Agent for Netcool/OMNIbus。

完成下列步骤以安装代理程序：

Linux `installAPMAgents.sh`

Windows `installAPMAgents.bat`

将提示您安装 Integration Agent for Netcool/OMNIbus。

然后将针对您的环境启动先决条件扫描，该扫描可能需要一些时间才能完成。如有任何需求未能得到满足，将显示一条消息指示您查看日志文件以了解故障原因。缺少先决条件（诸如缺少库或磁盘空间不足）会停止安装。您必须解决故障，然后重新启动安装。

将使用以下缺省设置来配置代理程序：

`Tivoli EIF 探测器主机：localhost`

`Tivoli EIF 探测器端口：9998`

安装后，会自动启动 Integration Agent for Netcool/OMNIbus。

监视代理程序会安装到指定的目录 (*install_dir*)。使用以下缺省目录：

Linux `/opt/ibm/apm/agent`

Windows `C:\IBM\APM\`

5. 如果要安装 Integration Agent for Netcool/OMNIbus 的系统不同于 Probe for Tivoli EIF 所在的系统，或者如果 Probe for Tivoli EIF 使用的端口号不同于缺省端口 9998，那么必须配置 Integration Agent for Netcool/OMNIbus 以连接到您的 Probe for Tivoli EIF。

注: 如果安装 Integration Agent for Netcool/OMNIbus 的系统与您的 Probe for Tivoli EIF 所在的系统相同，且 Probe for Tivoli EIF 正在使用缺省端口 9998，那么无需完成此步骤。

Linux 完成下列步骤以配置代理程序：

a. 运行以下命令：

```
install_dir/bin/omnibus-agent.sh config
```

b. 提示时，请提供您的 Probe for Tivoli EIF 主机名和端口号。

配置完成后，会自动启动 Integration Agent for Netcool/OMNIbus。

Linux 或者，可以使用以下步骤来复审并更改配置设置。

a. 在文本编辑器中打开 *install_dir*/samples/omnibus_silent_config.txt 响应文件。

b. 编辑此文件以设置或更改配置设置。确保取消注释配置行。

c. 保存并关闭响应文件。

d. 重新配置代理程序。指定您编辑的静默配置文件的标准路径来运行以下命令：

```
install_dir/bin/omnibus-agent.sh config install_dir/samples/omnibus_silent_config.txt
```

e. 重新启动代理程序以实施更改：

```
install_dir/bin/omnibus-agent.sh stop
install_dir/bin/omnibus-agent.sh start
```

Windows 完成下列步骤以配置代理程序：

a. 在文本编辑器中打开 *install_dir*\samples\omnibus_silent_config.txt 响应文件。

b. 编辑该文件以指定您的 Probe for Tivoli EIF 主机名和端口号。确保取消注释配置行。

c. 保存并关闭响应文件。

d. 通过指定您编辑的静默配置文件的标准路径来重新配置代理程序：

```
install_dir\BIN\omnibus-agent.bat config install_dir\samples\omnibus_silent_config.txt
```

e. 重新启动代理程序以实施更改：

```
install_dir\BIN\omnibus-agent.bat stop
install_dir\BIN\omnibus-agent.bat start
```

下一步做什么

遵循为 Netcool/OMNIbus 配置集成中的指示信息。

如果您要停止使用 Integration Agent for Netcool/OMNIbus 或要将代理程序移至其他系统，请使用以下命令卸载代理程序：

Linux *install_dir*/bin/omnibus-agent.sh uninstall

Windows *install_dir*\BIN\omnibus-agent.bat uninstall

为 Netcool/OMNIbus 配置集成

在安装 Integration Agent for Netcool/OMNIbus 后，必须将事件规则复制到 Probe for Tivoli EIF 并进行修改。您还必须更新 Netcool/OMNIbus ObjectServer 和数据库模式。

开始之前

在完成集成步骤之前，请通过使用以下命令停止 Integration Agent for Netcool/OMNIbus：

Linux *install_dir*/bin/omnibus-agent.sh stop

Windows *install_dir*\BIN\omnibus-agent.bat stop

install_dir is the default /opt/IBM/apm/agent or C:\IBM\APM directory or the directory that you specified during the Integration Agent for Netcool/OMNIbus installation.

关于此任务

在安装 Integration Agent for Netcool/OMNIbus 之后，以下配置文件位于 *install_dir*/localconfig/i0/omnibus 和 *install_dir*\localconfig\i0\omnibus 目录中：

- itm_apm_db_update.sql
- itm_event.rules
- itm_apm_event.rules

其中，*install_dir* 是缺省的 /opt/IBM/apm/agent 或 C:\IBM\APM，或者是您在 Integration Agent for Netcool/OMNIbus 安装期间指定的目录。

要点: 即使 Probe for Tivoli EIF 和 Netcool/OMNIbus ObjectServer 已与 IBM Tivoli Monitoring、 Probe for Tivoli EIF、IBM SmartCloud® Monitoring - Application Insight®、IBM SmartCloud Application Performance Management 或先前版本的 Cloud APM 集成，也必须完成这些步骤。

过程

在此过程中，访问 IBM Tivoli Monitoring 文档的链接时，仅完成链接页面上提供的步骤。

1. 将 Integration Agent for Netcool/OMNIbus itm_event.rules 和 itm_apm_event.rules 文件复制到 Probe for Tivoli EIF 安装目录。
   以下目录是缺省目录：

   Linux *install_dir*/tivoli/netcool/omnibus/probes/linux2x86
   Windows *install_dir*\Tivoli\Netcool\omnibus\probes\win32

   其中，*install_dir* 是缺省值。
2. 在文本编辑器中打开 Probe for Tivoli EIF tivoli_eif.rules 文件，并完成以下步骤之一：
   - 如果您是现有的 IBM Tivoli Monitoring 客户，并已经完成了 OMNIbus 集成，将此行添加到 itm_event.rules 文件：include "itm_apm_event.rules"。
   - 如果尚未设置 OMNIbus 集成，从引用 itm_event.rules 文件的行中移除注释。

   有关详细步骤，请参阅 IBM Tivoli Monitoring 文档中的更新 EIF 探测器的规则文件。
3. 如果使用的是多层 OMNIbus 解决方案，请完成 IBM Tivoli Monitoring 文档中的使用 IBM Tivoli Monitoring 属性、表和触发器来更新 Netcool/OMNIbus ObjectServer 部分中描述的所有任务。
4. 将 itm_apm_db_update.sql 文件装入数据库，更新 Netcool/OMNIbus ObjectServer 数据库模式：

   Linux

   ```
   $OMNIHOME/bin/nco_sql -user user_name -password password
   -server server_name < itm_apm_db_update.sql
   ```

   示例：

   ```
   $OMNIHOME/bin/nco_sql -user smadmin -password passw0rd -server NCOMS <
   /tmp/apm/itm_apm_db_update.sql
   ```

   Windows

   ```
   itm_apm_db_update.sql | %OMNIHOME%\..\bin\isql -U user_name
   -P password -S server_name
   ```

   示例：

   ```
   \temp\apm\itm_apm_db_update.sql | %OMNIHOME%\..\bin\isql -U smadmin
   -P passw0rd -S NCOMS
   ```

   在运行脚本时可能显示以下错误消息，但这些消息是无害的：

   - Object exists 和 Attempt to insert duplicate row，如果先前运行了脚本（例如，用于集成较早版本的 Cloud APM 或 Tivoli Monitoring）。

- 在 `itm_event_clear` 处或附近显示 `ERROR=Object not found on line 4 of statement "-- A workspace table for the ITM event clear automation..."`。
- 在 `itm_problem_events` 处或附近显示 `ERROR=Object not found on line 1 of statement "delete from alerts.itm_problem_events;..."`。
- 在 `itm_problem_events` 处或附近显示 `ERROR=Object not found on line 1 of statement "drop table alerts.itm_problem_events;..."`。

5. 重复步骤 5，以便将该文件装入 Object Server 两次，确保正确装入所有依赖关系。
6. 启动（或重新启动） Probe for Tivoli EIF。
7. 通过使用以下命令来重新启动 Integration Agent for Netcool/OMNIbus：

   Linux *install_dir*/bin/omnibus-agent.sh start

   Windows *install_dir*\BIN\omnibus-agent.bat start

与 Operations Analytics - Log Analysis 集成

环境包含 IBM Operations Analytics - Log Analysis 时，可以集成 SCALA 以支持在 Cloud APM 控制台中搜索应用程序日志。

关于此任务

与所安装的 Log Analysis 应用程序集成包括使用 URL 配置 Cloud APM 服务器。有关 Log Analysis 的更多信息，请参阅 IBM Operations Analytics - 开发人员社区。

您必须为 Log Analysis 安装提供顶级 URL，例如：

```
https://loganalysis.example.com:9987/Unity
```

Log Analysis URL 必须可从用户在其中使用 Cloud APM 控制台的主机进行访问。无需使其可从开放式因特网进行访问。

过程

1. 在 Cloud APM 控制台中，单击 **系统配置** > **高级配置**。
2. 选择 **UI 集成**类别。
3. 在 **Log Analysis URL** 字段中，输入用于启动 Log Analysis 应用程序的 URL。

结果

Log Analysis 应用程序集成完毕，并为您启用了从应用程序性能仪表板搜索应用程序日志的功能。

注:

不支持从 Cloud APM 控制台单点登录到 Log Analysis 应用程序。

下一步做什么

选择 **性能** > **应用程序性能仪表板** 。可以选择应用程序，然后使用 搜索框来搜索日志文件。缺省情况下搜索最后一小时的条目，但您可以更改此时间段。如果选择应用程序，那么仅搜索服务器上与该应用程序关联的日志。有关详细的指示信息，请参阅第 916 页的『搜索日志文件』。

与 Operations Analytics - Predictive Insights 集成

在将 IBM Cloud Application Performance Management 与 Operations Analytics - Predictive Insights 集成时，Operations Analytics - Predictive Insights 将分析 Cloud APM 收集的度量数据，并在数据中识别到异常时生成警报。

异常在 Cloud APM 仪表板中显示为事件，如第 941 页的『使用 Operations Analytics - Predictive Insights 调查异常』所述。然后，您可以向下钻取到 Operations Analytics - Predictive Insights 用户界面以查看有关异常的更多详细信息。

在将 Operations Analytics - Predictive Insights 添加到 Cloud APM 预订时，将自动配置为收集和分析性能指标。无需进一步配置。要将 Operations Analytics - Predictive Insights 添加到 Cloud APM 预订，请转到 IBM Support 并提交服务请求。

您可以将下列 Cloud APM 代理程序与 Operations Analytics - Predictive Insights 集成：

- Monitoring Agent for Db2
- Monitoring Agent for IBM Integration Bus
- Monitoring Agent for JBoss
- Monitoring Agent for Linux OS
- Monitoring Agent for Oracle Database
- Response Time Monitoring Agent
- Monitoring Agent for UNIX OS
- Monitoring Agent for VMware VI
- Monitoring Agent for WebSphere Applications
- Monitoring Agent for WebSphere MQ
- Monitoring Agent for Windows OS
- Monitoring Agent for Tomcat

与 Alert Notification 集成

为了获得比 Cloud APM 基本电子邮件转发功能更多的灵活性，可以与 Alert Notification 产品集成来扩展 Cloud APM 功能，以便在发生问题时通知用户。

与 Alert Notification 集成使您能够精细地控制哪些人能够收到通知以及如何接收通知。例如，每个用户可以决定自己是否想要接收电子邮件、SMS 或语音邮件。通知也可以路由到 Slack。不同的用户可以根据一天的具体时间、一周的具体日期等等接收不同类型的通知。每个用户都可以决定想要接收的警报类型。例如，数据库管理员可能只想接收达到警告或更高级别严重性的数据库警报。

开始之前

Alert Notification 自动集成在 IBM Cloud Application Performance Management 预订和试用预订中。

关于此任务

事件会自动配置为发送到 Alert Notification。然后，您可以为自己的预订创建通知策略，以确定要接收哪些警报的通知。

您可以将受监视应用程序添加到资源组。对于每个保存的资源组，可以配置一个或多个电子邮件地址。组中的任何受管系统的性能超过阈值时，将会向该组配置的地址发送电子邮件通知。

对于在分配到组的受管系统上打开的任何事件，Alert Notification 都会发送电子邮件通知。

过程

对于想要根据资源组路由其警报的任何资源组，完成步骤第 826 页的『1』 和第 827 页的『2』。

1. 如果想要根据资源组路由通知，请完成以下子步骤：
   a) 在 Cloud APM 控制台中，单击 **系统配置 > 资源组管理器**。
   b) 选择资源组，然后单击 **编辑**以打开“**资源组编辑器**”。

      要点: 必须先保存新资源组或编辑后的资源组，然后再打开 Alert Notification。在保存后，资源组编辑器将会关闭，如果要继续使用，必须重新打开。

如果不进行保存，那么在步骤 第 827 页的『2』 中配置电子邮件通知并返回到资源组编辑器之后，可能会收到错误消息，通知您不允许同步更新。

2. 完成下列步骤以配置电子邮件通知：

   a) 在 APM 控制台的资源组编辑器中，单击**配置电子邮件通知** URL 以在新的浏览器标签页或窗口中打开 IBM Alert Notification 应用程序。

   该操作会根据您在步骤 第 826 页的『1』 中选择的资源组自动创建新策略。

   注意: 如果您的浏览器不允许弹出窗口，则会阻止 Alert Notification 窗口打开。您必须将浏览器设置为允许 Alert Notification 窗口，才能将其打开，以针对资源组配置电子邮件通知。

   b) 在 Alert Notification 中，配置用户和组，并将其电子邮件地址与资源组关联起来以通过电子邮件接收事件通知。

   您可以为想要接收的警报类型设置自己的 Alert Notification 策略和通知。例如，Linux 管理员可能想接收所有 Linux 系统的电子邮件、SMS 消息或语音邮件。然后，他们想关闭严重的高严重性警报。策略允许他们过滤 Linux OS 代理程序所生成的所有事件。有关使用 Alert Notification 应用程序中的通知编辑器的信息，请参阅创建通知策略。

下一步做什么

有关 Alert Notification 的更多信息，请参阅 IBM Alert Notification 文档。

与 Control Desk 集成

您可以从 Cloud APM 配置事件以自动在 IBM Control Desk 中打开凭单。

关于此任务

可以使用 IBM Control Desk 的内部部署版本或云端版本来集成 IBM Cloud Application Performance Management。

过程

使用下列其中一个过程：

· 要在内部部署 IBM Control Desk V7.6 中提交凭单，请完成下列步骤：

1. 配置 IBM SmartCloud Notes® 电子邮件帐户，以使 Control Desk 能够使用 IMAP 邮件客户端。配置期间，必须确保选择**立即启用 IMAP 访问**。有关更多信息，请参阅 IBM Connections Social Cloud Knowledge Center 中的启用 IMAP 访问。
2. 在 Cloud APM 控制台中，单击 **系统配置** > **高级配置**，然后设置以下参数：

   目标电子邮件地址
   指定用于创建“服务请求”凭单的 SmartCloud Notes 电子邮件地址。

   电子邮件主题行
   指定电子邮件的主题行，例如，`PMaaS Event`。
3. 转至 Marketplace 支持，并选择**服务请求**来提交支持凭单，以完成启用。

   在凭单中提供以下信息：

   – SmartCloud Notes 电子邮件地址

     例如，`user@ibmserviceengage.com`。
   – SmartCloud Notes 电子邮件密码
   – SmartCloud Notes 电子邮件标准服务器名称

     例如，`imap.notes.na.collabserv.com`。
   – SmartCloud Notes 电子邮件端口号

     例如，`993`。

– 客户 IBM Control Desk URL

  按如下所示设置链接的格式：`https://<subscriber-id>.sccd.ibmserviceengage.com/maximo_t4hj/webclient/login/login.jsp?welcome=-true`

4. 要配置电子邮件侦听器，以解析电子邮件，以及在您想要分配凭单给 IBM Control Desk on Cloud 中的其他组时适当地处理电子邮件，请参阅配置电子邮件侦听器。

· 要在 IBM Control Desk 云环境中提交凭单，请完成下列步骤：

1. 转至 Marketplace 支持，并选择**服务请求**来提交支持凭单，以完成启用。

   在凭单中提供以下信息：

   – SmartCloud Notes 电子邮件地址

     例如，`user@ibmserviceengage.com`。

   – SmartCloud Notes 电子邮件密码

   – SmartCloud Notes 电子邮件标准服务器名称

     例如，`imap.notes.na.collabserv.com`。

   – SmartCloud Notes 电子邮件端口号

     例如，`993`。

   – 客户 IBM Control Desk URL

     按如下所示设置链接的格式：`https://<subscriber-id>.sccd.ibmserviceengage.com/maximo_t4hj/webclient/login/login.jsp?welcome=-true`

2. 要配置电子邮件侦听器，以解析电子邮件，以及在您想要分配凭单给 IBM Control Desk on Cloud 中的其他组时适当地处理电子邮件，请参阅配置电子邮件侦听器。

与 IBM Cloud 集成

您可以使用选择的数据收集器查看 IBM Cloud 环境中的应用程序的监视信息。

当配置为从 IBM Cloud 应用程序收集数据时，数据收集器支持将监视功能与 IBM Cloud 集成。数据收集器将关于 IBM Cloud 应用程序的资源监视和诊断数据传送到 Cloud APM 服务器。Cloud APM 服务器接收并处理数据收集器所收集的监视信息。可以监视下列类型的 IBM Cloud 应用程序：

· Liberty 应用程序
· Node.js 应用程序
· Python 应用程序
· Ruby 应用程序

正确配置数据收集器之后，您可在 Cloud APM 控制台上查看监视数据。有关配置指示信息，请参阅第 157 页的『用于配置数据收集器的常规过程』。

与 IBM Agent Builder 集成

您可以使用扩展 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境的监视功能的 Agent Builder 来创建、修改、调试和打包代理程序。定制代理程序使用上述任一环境来监视任何类型的内部或定制软件。

有关详细信息，请参阅《IBM Agent Builder 用户指南》。

第 9 章 管理

启动 Cloud APM 控制台

从浏览器登录到 Cloud APM 控制台，以在仪表板中查看应用程序的运行状态。

开始之前

- 通过使用初次注册服务时收到的确认电子邮件中提供的链接，激活您的帐户。
- 为了确保用户界面不出现截断问题，请至少使用 1280 x 1024 分辨率。
- 为实现最佳性能，请使用一种受支持的浏览器。要获取受支持的浏览器的列表，请转到 System requirements (APM Developer Center)。选择其中一个 IBM Cloud Application Performance Management 产品链接，然后单击“Server”链接；在显示的报告中，单击或向下滚动至“Web Browsers” “” “”。

过程

1. 要访问 Cloud APM 控制台，请使用提醒您服务已就绪的电子邮件中提供的链接。
2. 您还可以从 IBM Marketplace Web 站点访问控制台：
   a. 转至 IBM Marketplace Web 站点上的“产品和服务”。
   b. 以注册服务时所用的用户名和密码登录。
   c. 在 Cloud APM 服务器行中，单击**启动**。

结果

登录后，将显示“**入门**”页面，其中带有**用户任务**和**管理员任务**学习选项，以及**社区资源**链接。

下一步做什么

- 单击超文本链接以获取 Cloud APM 仪表板的教程，熟悉用户界面元素。观看用户任务和管理员任务的视频以帮助您开始使用和定制 Cloud APM 环境。
- 添加应用程序以供在逻辑分组中查看资源的仪表板，例如，“联机订购”。有关指示信息，请参阅第 930 页的『管理应用程序』。
- 创建阈值以测试满足时将导致打开事件的条件。例如，您可以具有在存储容量到达 90% 后打开一个事件的阈值。有关指示信息，请参阅第 838 页的『阈值管理器』。
- 向用户组和角色添加和分配用户以控制对 Cloud APM 控制台功能和受管资源的访问。有关更多信息，请参阅第 850 页的『管理用户访问权』。
- 要了解有关监视 IBM Java 应用程序堆栈和 IBM 集成堆栈的信息，请参阅第 73 页的『方案』。
- 如果浏览器未显示“入门”页面或应用程序性能仪表板，而是转至 IBM Web 站点，说明您的用户标识无权访问 Cloud APM 控制台。您必须从管理员处请求访问权。
- 如果未针对数据源显示任何度量，那么请参阅 developerWorks 上的 Cloud Application Performance Management 论坛。在论坛中搜索“dashboard”，回复条目以提出相关问题，或者创建新条目并描述症状。
- 如果要从 Internet Explorer 8、9 或 10 启动 Cloud APM 控制台，并收到此页面无法显示错误，可能需要启用安全选项 TLS 1.2。有关更多信息，请转至 Cloud Application Performance Management 论坛，并搜索“tls” “”。

© Copyright IBM Corp. 2014, 2019

阈值和资源组

阈值用于测试特定情况（例如，每分钟的事务数少于 100），并在满足条件时打开事件。使用阈值来监视受监视资源的实际和潜在问题。向资源组分配阈值以在属于该组的同一类型的所有受管系统上进行监视。

背景信息

查看背景信息以了解阈值、代理程序的预定义阈值、阈值分配到的资源组以及定制阈值。

预定义阈值

您的监视代理程序随附了在代理程序中启用和启动的预定义阈值。安装代理程序后首次打开“**阈值管理器**”时，针对所选数据源类型列出的阈值是预定义阈值。这些预定义的阈值分配到该代理程序的缺省系统资源组，并显示在**已分配的组**列中。

如果编辑了预定义阈值（例如更改了名称或条件），不会再将该阈值视为预定义阈值，而是视为定制阈值。但是，可以将预定义阈值的已分配资源组从缺省系统组更改为用户定义的组，而该阈值仍保持为预定义阈值。

如果不想使用预定义阈值，可以在“**高级配置**”页面中将其关闭（请参阅第 912 页的『阈值启用』）。禁用预定义阈值不会将其从“**阈值管理器**”中移除；而只会移除其组分配，从而使其显示为不活动。在禁用预定义阈值之后，可以打开“**阈值管理器**”，会看到所有预定义阈值的**已分配的组**列均为空（请参阅第 831 页的『禁用阈值的示例』）。

您可以通过将阈值分配给任何可用资源组以将其启用为定制阈值。

定制阈值

您创建的新阈值为定制阈值，在“**阈值管理器**”的**源**列中进行指示。如果编辑了预定义阈值，它也会成为定制阈值，其源将从“预定义”更改为“定制”。

执行命令

针对求值为 true 的阈值打开事件后，可以在打开了事件的受监视系统上自动运行命令或脚本。例如，您可能希望记录信息、触发蜂鸣声，或停止正在过度使用资源的作业。

此命令使用以下语法：

```
&{data_set.attribute}
```

其中，*data_set* 是数据集名称，*attribute* 是显示在阈值编辑器中的属性名称。如果数据集名称或属性名称中包含空格，请替换为下划线。

以下示例显示如何将磁盘名称参数传递至受管资源：

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

可以从数据集传入一个或多个属性。如果指定多个属性，那么各个属性将按顺序（$1、$2，等等）传递到命令中。

该命令将以用来启动代理程序的用户帐户从命令行运行。 例如，如果以 root 用户身份运行代理程序，那么 root 用户在受管系统上运行该命令。

以下选项控制命令运行的频率：

如果数据集返回多行，而您只想针对数据样本中出现的第一个事件运行命令，那么请选择 ☐**仅对第一个事件**。要对导致事件打开的每行都运行命令，请清除此复选框。

选择 ☐**对于每个连续 true 时间间隔**，从而每次阈值请求为 true 时运行命令。要在阈值为 true 时发出命令，但此后一直到阈值求值为 false 都不再执行此命令，直到后续时间间隔内再次求值为 true 时才再次发出命令，请清除此复选框。

资源组

资源组表示一组受管系统，用于控制阈值的分布方式。您可以将阈值分配到包含要运行该阈值的受管系统的资源组。

所有预定义阈值都有缺省资源组分配，即针对代理程序类型的系统定义组，例如，Db2 和 Microsoft IIS。

您可以创建定制资源组并选择要包含在每个组中的受管系统。定制资源组中可以包含多种代理程序类型；分配到该组的阈值仅分布到相同代理程序类型的受管系统。例如，使用 Linux OS 属性创建并分配到包含 Linux OS、MongoDB 和 Python 受管系统的资源组的阈值，只会分布到 Linux OS 受管系统。

有关更多信息，请参阅第 833 页的『资源组管理器』。

应用程序性能仪表板事件状态

应用程序性能仪表板中显示的状态严重性指示所选应用程序、组、子组和受管系统实例的最高事件严重性。

在从导航器或从**我的所有应用程序**仪表板中的摘要框中选择应用程序之后，将通过选项卡式仪表板呈现应用程序的不同方面。“**事件**”选项卡提供了有关所选导航器项的事件的信息，如第 939 页的『事件状态』所述。

阈值更改影响分配到相同监视代理程序的其他阈值

创建、修改或删除阈值定义或者更改分配给监视代理程序的阈值列表之后，对于阈值分配到的代理，将关闭所有采样事件。事件关闭后，监视代理程序会为求值为 true 的所有阈值条件重新打开事件。在 Cloud APM 控制台上，已关闭的事件将不显示在控制台中，直至通过新的**时间戳记**值重新打开这些事件为止。如果您收到事件的电子邮件通知，那么也会收到关闭事件和打开事件的电子邮件通知。

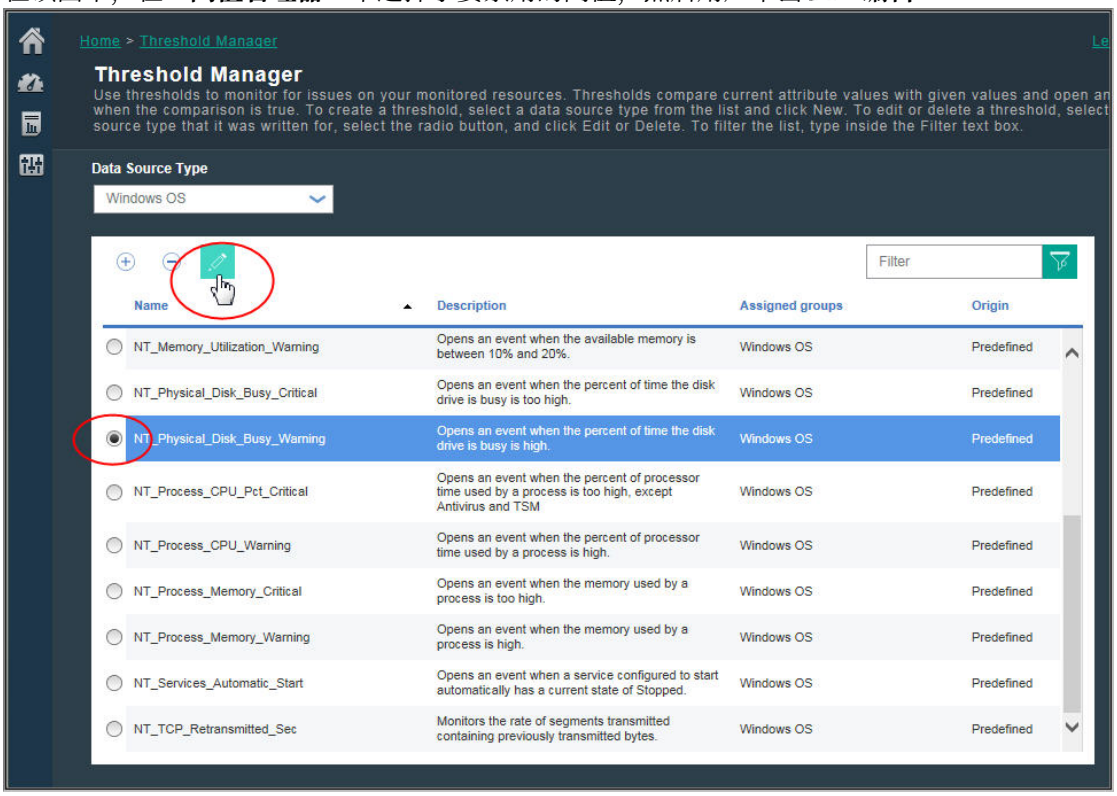
例如，假设您有名为“站点系统”的定制资源组，并且已为其分配 Linux 操作系统和 WebSphere Applications 阈值。您创建一个新的 Linux 操作系统阈值并将其分配给“站点系统”。那么将关闭该 Linux OS 代理程序上已分配给“站点系统”的任何打开的已采样事件。然后，如果阈值条件仍未 true，那么将重新打开采样事件。

禁用阈值的示例

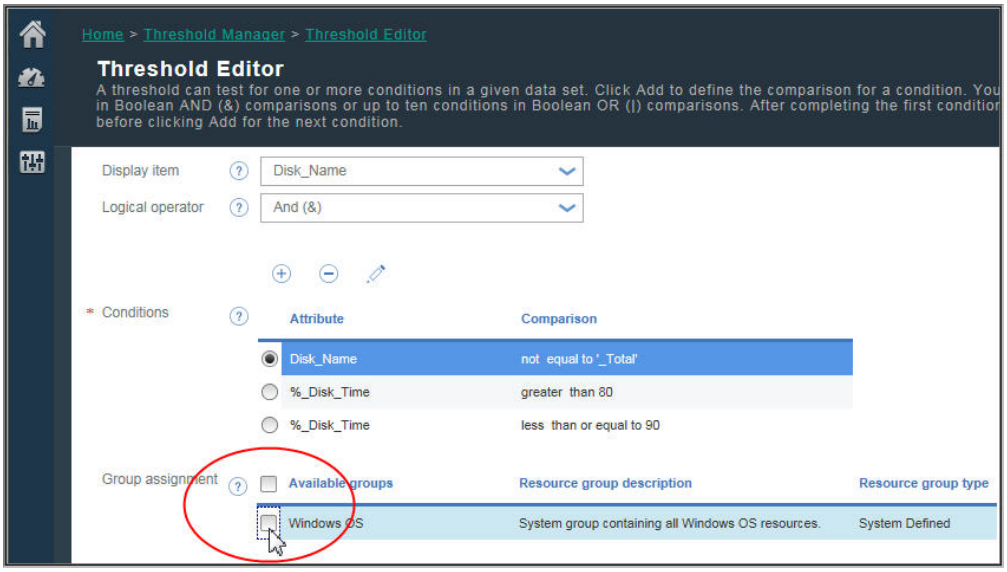
您可以禁用环境中所有代理程序的预定义阈值。也可以单独禁用阈值，无论是预定义还是定制阈值。阈值被禁用后，不会在受管系统上运行，且不会打开事件。您可以通过移除阈值的一个或多个资源组分配来禁用阈值。也可以使用“**高级配置**”设置来禁用所有代理程序的预定义阈值。

禁用单个阈值

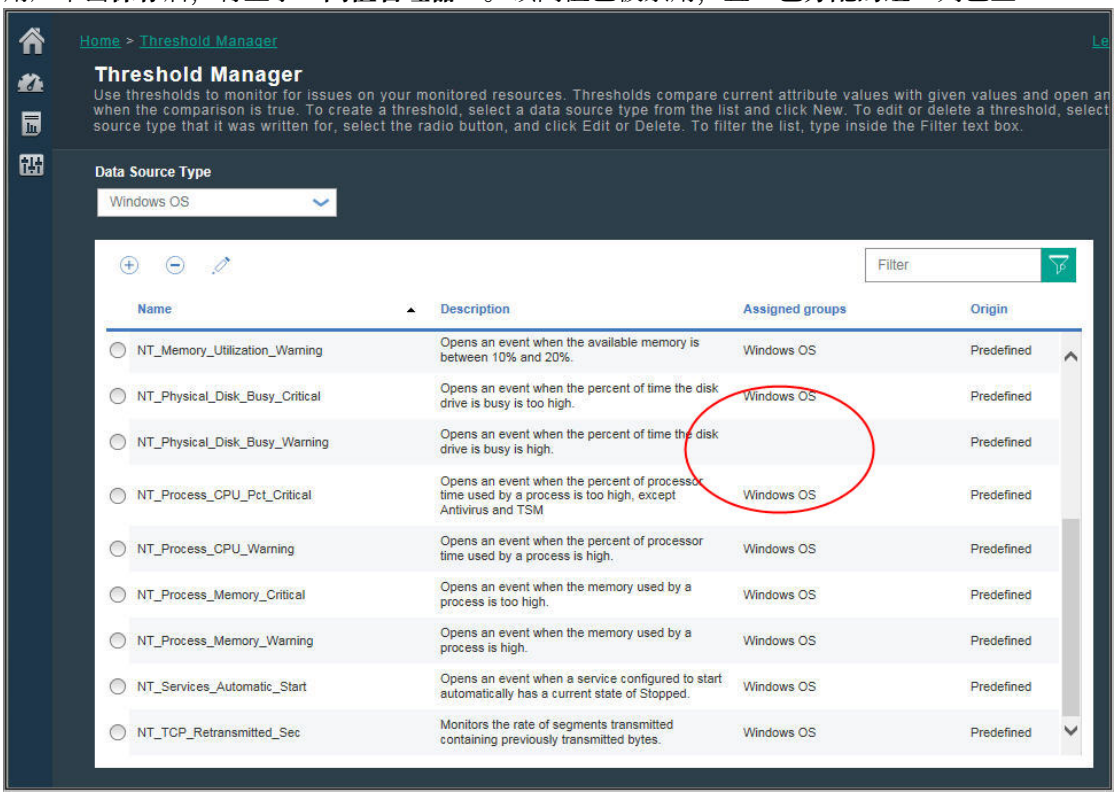
在该图中，在“**阈值管理器**”中选择了要禁用的阈值，然后用户单击“**编辑**”：



此时将在“阈值编辑器”中打开该阈值。用户清空**组分配**字段中已分配资源组的复选框：

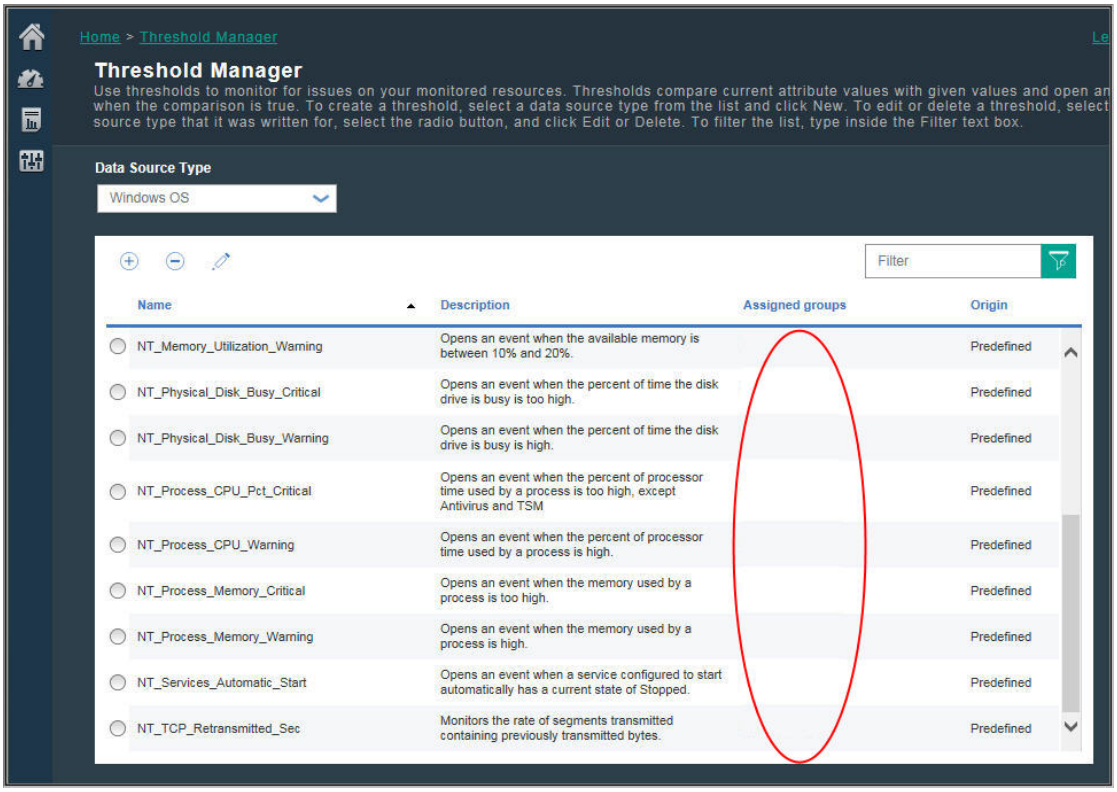


用户单击**保存**后，将显示“**阈值管理器**”。该阈值已被禁用，且“**已分配的组**”列已空：



禁用所有预定义的阈值

按照第 912 页的『阈值启用』所述，在“**高级配置**”页面中关闭所有监视代理程序的所有预定义阈值。下次打开“**阈值管理器**”时，所有预定义阈值的“**已分配的组**”列均为空，表示阈值处于不活动状态：



相关概念

第 830 页的『背景信息』
查看背景信息以了解阈值、代理程序的预定义阈值、阈值分配到的资源组以及定制阈值。

相关参考

第 838 页的『阈值管理器』

资源组管理器

您的受监视环境可能具有多个按用途分类的受管系统。此类系统通常具有相同的阈值需求。使用“**资源组管理器**”可将受管系统分组到可分配阈值的组。您也可以创建与“基于角色的访问控制”(RBAC) 策略关联的资源组。

单击 **系统配置 > 资源组管理器**后，将打开包含预定义资源组表的页面。最初，对于所安装的每个监视代理程序类型（例如 Windows 操作系统），显示一个预定义的系统组。. 每个系统组都包含代理程序的所有预定义阈值。

您对“**资源组管理器**”和资源组的访问权限由您的用户许可权控制。您必须具有资源组的查看许可权才能看到该资源组；必须具有修改许可权才能创建、编辑或删除资源组。

表包含管理资源组的工具：

- **新建**将打开“**资源组编辑器**”以分配受管系统和阈值。
- 选择资源组，在相邻窗格中查看对这个组分配的资源及阈值。
- 选择资源组，然后单击 **编辑**以打开“**资源组编辑器**”从而更改受管系统和阈值分配。
- 选择不再需要的资源组，然后单击 **删除**。确认删除后，必须将已分配到该组的任何阈值分配到其他组，才能继续在您的受管系统上运行这些阈值。
- 您可以在过滤器文本框 中单击鼠标并输入要过滤的值。输入时，将过滤掉不适合条件的行。要清除过滤器，请单击过滤器框 中的 ×，或者按退格键。

表将显示可用的资源组：

资源组名

预定义的组按其代理程序类型命名，而定制组由作者命名。

资源组描述
预定义组描述为受监视资源的系统组，而定制组由作者描述。

系统组（例如 Linux OS）包含代理程序的所有预定义阈值，以及所有已安装该代理程序的受管系统的所有预定义阈值。您可编辑系统组，以分配或移除阈值，但无法分配或移除受管系统。受管系统将自动分配给同一类型的系统组，包括 Tivoli Monitoring 域中的任何系统组（如果已配置 Hybrid Gateway）。

某些系统资源组与支持子节点的代理程序相关。根据代理程序类型，可以将子节点和/或代理程序节点添加至应用程序。如果只有子节点可添加至已定义的应用程序，则您无法查看任何已针对该代理程序节点定义的阈值的事件。但是，可以将这些事件转发至事件管理器，例如 Netcool/OMNIbus。并且，IBM Cloud Application Performance Management 订户可设置 Alert Notification。

资源组类型
预定义组具有系统定义类型。对于环境中安装的每种代理程序类型，都有一个预定义的组。

环境中由您或他人创建的定制组具有用户定义类型。

资源组编辑器

在单击 **新建**以添加组之后，或者在选择组并单击 **编辑**以编辑组之后，将显示带有以下字段的“**资源组编辑器**”：

组名
组的名称是必需的。您可以更改现有的定制组名，在保存更改之后会自动更新对该组的所有引用。

组描述
对于定制组，这是可选项。添加组组织的描述。描述在“**资源组管理器**”中显示。

资源分配
所有可添加至组的受管系统均显示在代理程序列表中，并按其受管系统名称、主机名、代理程序类型及它们的域排列。您可以单击列标题，以便按代理程序名称、主机名、类型或域对列表进行排序。

要填充该组，请选择一个或多个受管系统的复选框。

您可以选中 **仅显示选中的资源**，以隐藏未分配的受管系统。

如果您已配置 IBM Cloud Application Performance Management Hybrid Gateway，则可将 IBM Tivoli Monitoring 域中的受管系统添加到用户定义的资源组。无法将 Tivoli Monitoring 受管系统添加至系统定义的组，也无法针对其创建阈值。

阈值分配
所有预定义阈值或已通过“**阈值管理器**”添加的阈值都会显示在阈值列表中，并按其名称及代理程序类型排序。您可以单击列标题，以便对列表进行排序。

要向组添加阈值，请选择名称旁的复选框；要从组中移除阈值，请清除该复选框。 您必须具有“**阈值管理器**”的查看许可权才能添加或移除阈值。在向系统组添加阈值时，可用阈值限制为其数据集适用于该系统组的阈值。

您分配给组的阈值会分发到相同代理程序类型的组中的所有受管系统。虽然您可以将任何监视代理程序类型的阈值分配给组，但已分配的阈值仅分发到属于该组成员的相同类型的受管系统。例如，如果将 MySQL_Process_Down 阈值分配给组，那么它将包含在组中，但是仅分发到属于组的 Monitoring Agent for MySQL 受管系统。

您可以选择 **仅显示选中的阈值**，以隐藏未分配的阈值。如果已对列表进行过滤，那么单击过滤器框 中的 ×，即可清除过滤器并启用复选框。

您还可以从“**阈值管理器**”为资源组分配阈值。

配置电子邮件通知
可用于 IBM Cloud Application Performance Management：单击**配置电子邮件通知**以在新的浏览器选项卡或窗口中打开 IBM Alert Notification 应用程序。使用 Alert Notification 以创建用户并将其电子邮件地址与资源组相关联，以供通过电子邮件接收事件通知。

注意: 如果您的浏览器不允许弹出窗口，则会阻止 Alert Notification 窗口打开。您必须将浏览器设置为允许 Alert Notification 窗口，才能将其打开，以针对资源组配置电子邮件通知。

您单击**保存**后，资源组将随资源组列表一起保存，并显示在“**资源组管理器**”表中。

相关任务
第 826 页的『与 Alert Notification 集成』

第 910 页的『浏览 API』

相关参考

第 838 页的『阈值管理器』

教程：定义阈值

阈值是针对受管资源的潜在和实际问题的警报触发机制。 使用本教程来了解定义阈值以在发生条件时引发警报的基本步骤。

关于此任务

本教程使用 Linux 操作系统代理程序向您显示如何在“**阈值管理器**”中定义阈值，以及如何在应用程序性能仪表板中查看所引发的警报。您的用户标识必须具有“**阈值管理器**”的查看许可权才能完成以下步骤。

过程

1. 在导航栏中单击 **系统配置 > 阈值管理器**。
2. 单击**数据源类型**列表框并选择 **Linux 操作系统**数据类型。
   此时会在表中显示为 Linux 操作系统代理程序定义的阈值。
3. 单击 **新建**以打开“**阈值编辑器**”，用于定义阈值。
4. 定义阈值以在平均 CPU 低于 75% 时引发警报“严重性未知”：
   a) 在**名称**字段中，输入 CPU_average_below_75_percent。
   b) 在**描述**字段中，输入阈值教程
   c) 保留**严重性**、**时间间隔**和**所需连续样本**字段的缺省值。
   d) 在**数据集**字段中，选择 **KLZ CPU 平均值**。
   e) 在**条件**字段中，单击 **新建**并在弹出的对话框中添加比较：
      1) 在**属性**字段中，选择 **CPU_Usage_Current_Average**
      2) 在**运算符**字段中，选择**小于**
      3) 在**值**字段中，输入 75

      在单击**确定**之后，**条件**字段中将显示属性和比较。
   f) 在**组分配**字段中，选择 **Linux 操作系统**系统组。
   g) 单击**保存**以完成定义并返回到“**阈值管理器**”页面。
   此时应在为“Linux 操作系统”数据源定义的阈值列表中显示 CPU_average_below_75_percent。

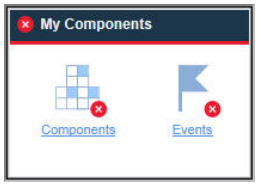
结果

您已定义当任何 Linux 操作系统受管系统的平均 CPU 使用率低于 75% 时引发警报的阈值。

下一步做什么

· 查看事件：

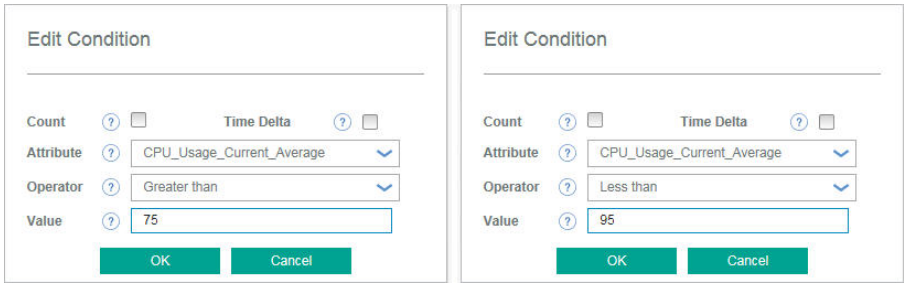
1. 在导航栏中，单击 **性能 > 应用程序性能仪表板**。
2. 在“**我的组件**”摘要框中，单击**事件**链接。

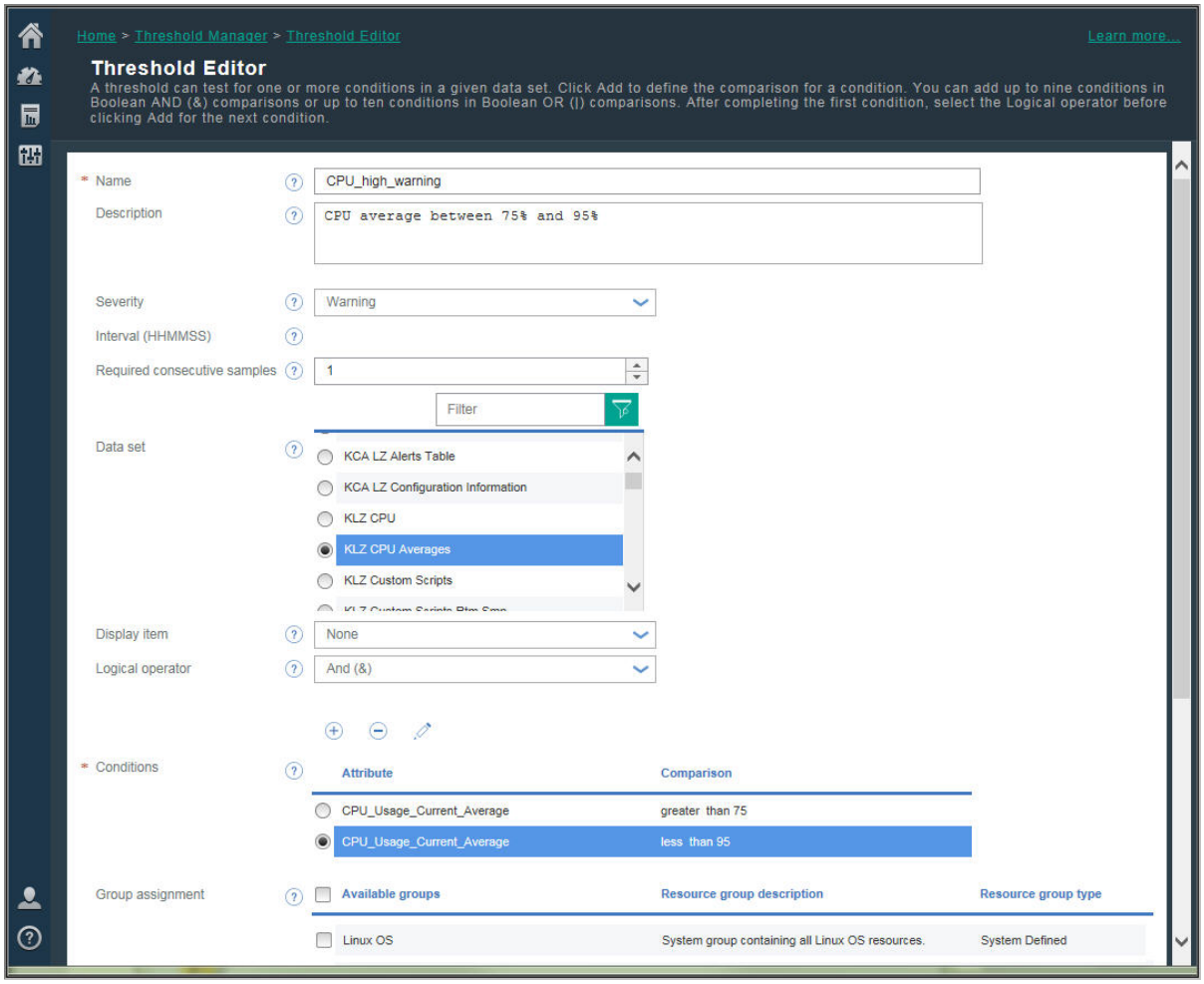


3. 在打开的“**事件**”选项卡中，从列表中查找 CPU_average_below_75_percent 阈值。可能需要 1 或 2 分钟才能引发警报。CPU 平均值高于 75% 时却没有引发警报。

· 编辑阈值：

1. 在导航栏中单击 **系统配置 > 阈值管理器**。
2. 单击**数据源类型**列表框并选择 **Linux 操作系统**数据类型。
3. 从列表中选择 CPU_average_below_75_percent 阈值，然后单击 **编辑**。
4. 更改要发出警报的条件的任何字段。例如，可以将**名称**更改为 CPU_high_warning，将**严重性**更改为 **Warning**，将条件编辑为**大于 75**，然后添加条件：**小于 95**。要编辑比较，请单击 **编辑**，将**运算符**更改为**大于**，然后单击**确定**。要添加新条件，请保留**逻辑运算符**字段的缺省值 **And (&)**，单击 **新建**以添加新条件，选择**运算符 Less than**，然后输入**值** 95。





· 复审代理程序的预定义阈值，并根据环境需要调整任何比较值。

· 创建新阈值以针对要发出警报的其他条件引发警报。

相关参考

第 838 页的『阈值管理器』

教程：定义阈值以便在受管资源上运行命令

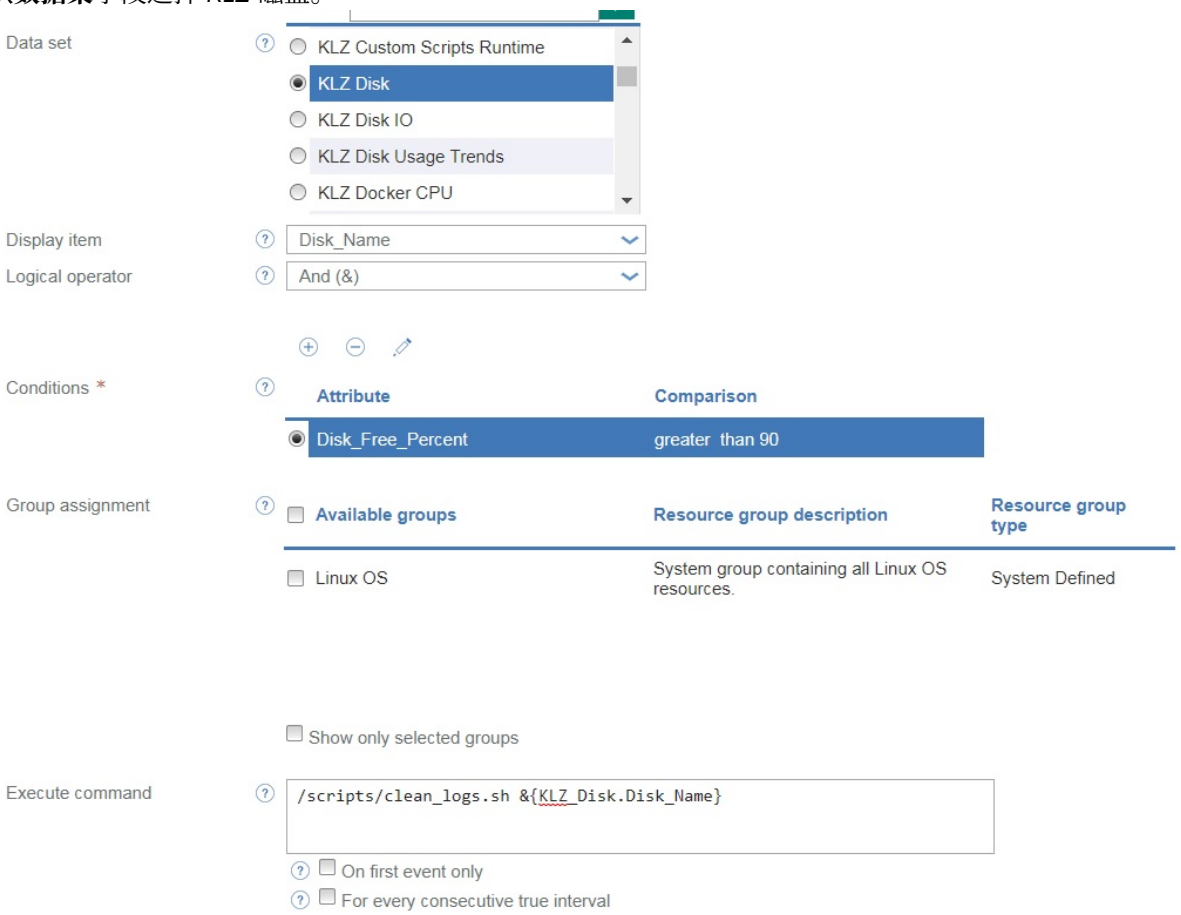
您可使用“**阈值编辑器**”将特定参数传递给代理程序。您可指定当事件触发时，要自动运行的命令或命令脚本。

关于此任务

本教程说明如何使用**执行命令**字段将参数传递给 IBM Cloud Application Performance Management 代理程序。

过程

1. 选择 **系统配置 > 阈值管理器**，以打开**阈值管理器**。
2. 从**数据源类型**字段中选择 *Linux OS*。
   此时会在表中显示为 Linux 操作系统代理程序定义的阈值。
3. 单击 **新建**打开“**阈值编辑器**”以定义阈值。
4. 通过指定各个参数的值，例如**名称**、**严重性**及**条件**，定义阈值和条件。
5. 从**数据集**字段选择 *KLZ* 磁盘。



6. 在**执行命令**字段中，输入下列命令：

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

   您必须将 *KLZ* 磁盘数据集名称中的空格替换为下划线。
   KLZ_Disk.Disk_Name 将传递到命令脚本中。

结果

您的命令脚本会设置为自动针对已定义的阈值来运行。这些命令会在触发事件的受监视系统上运行。

相关参考

第 838 页的『阈值管理器』

阈值管理器

使用“**阈值管理器**”可复查监视代理程序的预定义阈值以及创建和编辑阈值。阈值用于将属性的采样值与阈值中设置的值进行比较。如果采样值满足比较条件，那么将打开事件。当阈值比较不再满足时，事件将自动关闭。

在单击 **系统配置 > 阈值管理器**后，将显示包含针对选中的数据源类型定义的阈值表的页面。

单击**数据源类型**列表框时显示的数据类型针对的是受管环境中安装的监视代理程序和数据收集器的类型。请选择要为其创建或查看阈值的数据类型。

此表列出了针对选中的数据类型创建的所有阈值，其中包含用于管理阈值的工具：

- **新建**打开用于为所选数据类型定义阈值的“**阈值编辑器**”。
- 选择阈值，然后单击**编辑**以打开用于编辑定义的“**阈值编辑器**”。
- 选择不再需要的阈值，然后单击 **删除**。在确认要删除阈值后，将从列表以及将其分配到的任何资源组中移除阈值。然后关闭阈值的任何打开事件。
- 对于长列表，您可以在过滤器文本框 中单击鼠标，并输入要过滤器的值的开头部分。输入时，将过滤掉不适合条件的行。要清除过滤器，请单击过滤器框 中的 ×，或者按退格键。

有关表中显示的预定义阈值和定制阈值以及资源组分配（或缺少）的重要性的更多信息，请参阅第 830 页的『背景信息』。有关快速实践课程，请参阅第 835 页的『教程：定义阈值』。

阈值编辑器

在单击 **新建**，或选择阈值并单击 **编辑**后，将显示阈值编辑器以及以下字段：

名称

输入阈值的唯一名称。名称必须以字母开头，并且最多包含 31 个字母、数字和下划线，例如，“Average_Processor_Speed_Warning”。阈值名称将显示在应用程序性能仪表板的“**事件**”选项卡以及特定仪表板表中。

描述

可选。如果要记录用户可在“**阈值管理器**”中看到的阈值的用途，描述很有用。

严重性

从列表中选择相应的事件严重性： 致命、 紧急、 轻微、 警告、或 未知。

严重性将进行整合以显示在应用程序性能仪表板中：“致命”和“严重”事件显示为 ；“轻微”和“警告”事件显示为 ；而“未知”事件显示为 （请参阅第 939 页的『事件状态』）。

转发 EIF 事件?

如果在 **系统配置 > 高级配置**页面（第 911 页的『事件管理器』）中配置了事件转发，那么缺省情况下，打开事件将转发到配置的事件目标，例如，EIF 事件目标、Cloud Event Management 或 Alert Notification。如果不想要将此阈值的事件转发到任何事件目标，那么将设置更改为 **No**。

如果在 **系统配置 > 高级配置**页面（事件管理器）中配置了事件转发，那么缺省情况下，打开事件会转发到 EIF 接收器。如果不想要将此阈值的事件转发到 EIF 接收器，那么将设置更改为 **No**。

要定制阈值映射到转发事件的方式，从而覆盖转发到事件服务器的阈值和事件之间的缺省映射，请单击 **EIF 槽定制**。有关更多信息，请参阅第 841 页的『定制事件以转发到 EIF 接收器』。

时间间隔

使用 *HHMMSS* 格式输入或选择在两次获取数据样本之间要等待的时间，例如 00 15 00 表示 15 分钟。对于样本事件阈值，最小时间间隔为 000030（30 秒），最大时间间隔为 235959（23 小时 59 分 59 秒）。

值 000000（六个零）指示纯事件阈值。纯事件是自发通知。纯事件的阈值没有采样时间间隔，因此这些阈值没有可监视当前值的常量度量。纯事件会在 24 小时后关闭，或按照类别第 911 页的『事件管理器』中的“**高级配置**”页面的**纯事件关闭时间**字段中的设置关闭。

必需的连续样本

生成事件之前，指定有多少连续阈值样本必须求值为 true：对于任何设置为 1 且样本求值为 true 的阈值，会立即生成事件；而设置为 2 表示 2 个连续阈值样本求出的值必须为 true 后才能打开事件。

数据集

为要采样的数据类型选择数据集（属性组）。可包含在条件中的属性来自所选数据集。如果阈值具有多个条件，那么这些条件必须来自同一数据集。

要获取数据集的短描述，请将鼠标悬停在名称上。您可以单击悬浮式帮助中的“了解更多信息”链接以获取数据集和属性的完整描述。您还可以单击导航栏中的 **帮助** > **帮助内容** 或 **帮助** > **文档**，然后打开帮助或下载监视代理程序的参考。

某些代理程序归类为多节点代理程序，它们具有用于监视多个代理程序资源的子节点。多节点代理程序可能具有可在阈值中使用的数据集，但任何针对该阈值引发的事件都不会显示在应用程序性能仪表板中。系统会发出一条消息，向您通知该项限制。此类事件可以转发至 IBM Netcool/OMNIbus 事件管理器。

显示项

可选。仅适用于多行数据集。行求值导致事件打开后，在此事件关闭之前，不会在受监视系统上再为此阈值打开更多事件。通过选中显示项，即启用阈值以连续对数据样本中的其他行求值，并在其他行符合条件时，打开更多事件。此外，显示项在应用程序性能仪表板的“**事件**”选项卡中显示，因此您可以简单地区分打开其事件的行。此列表仅包含您可以指定为显示项的属性。

逻辑运算符

如果阈值只有一个条件，那么将忽略此字段。如果要测量多个条件，选择以下某个运算符，然后单击 **新建**以添加第二个或第三个（或更多）条件：

And (&)（如果希望同时满足运算符前后两个条件才违反阈值）
Or (|)（如果希望满足两个条件中的任何一个就违反阈值）

不支持混用多种逻辑运算符；请全部使用 And 运算符，或者全部使用 Or 运算符。使用 Or 运算符时，阈值最多可以有九个条件；使用 And 运算符时，阈值最多可以有 10 个条件。

如果要使用 `Missing` 函数（在随后的**运算符**一节中描述），那么在公式中只能使用 And 运算符。

条件

阈值定义可以通过逻辑方式同时包含多个阈值或条件。

单击 **新建**来添加条件。选择条件，然后单击 **编辑**来修改表达式，或者单击 **删除**来移除表达式。

在单击**新建**或 **编辑**后，填写打开的“**添加条件**”或“**编辑条件**”对话框中的字段：

计数

对于每个数据样本返回多行的数据集，您可以使每行都满足计数条件。在到达计数**值**并且满足公式中的任何其他条件后，引发事件。例如，如果“zombie”进程计数超过 10，将发出警报。

在以下示例中，在计数超过 10 行时，满足条件：**属性**为时间戳记，**运算符**为大于，**值**为 10。

选择**计数**复选框、要计数的**属性**、关系**运算符**和计数**值**。

如果公式具有多个条件，那么每个条件都必须使用 **And** 布尔运算符。**计数**和**时间增量**互斥：如果选择一个函数的复选框，那么将禁用另一个函数。属性不得是系统标识（例如，服务器名称或源代码）、不得指定为**显示项**，也不得来自于阈值为其打开纯事件的数据集。

时间增量

在条件中使用**时间增量**函数可以将采用时间戳记（例如，记录时间）与指定的时间差进行比较。

在选择**时间增量**复选框后，将显示“时间增量”字段，以供将 +（加号）或 -（减号）与天数、小时数、分钟数或秒数相组合。选择**采样时间**或**特定时间**作为要在比较中使用的值。

在以下“事件日志”示例中，公式将记录事件的时间与来自数据采样的时间戳记进行比较。如果事件发生在 七天以前，那么比较结果为 true。如果将关系运算符更改为“小于等于”，那么在 8 天、9 天及以后，比较结果将为 true：

属性 时间戳记
时间增量 -7 天
运算符 等于
值 输入时间

属性

请选择要在此条件中进行比较的属性。要查看属性的短描述，请在列表中的名称上悬停鼠标。

运算符
选择比较类型的关系运算符：

等于
不等于
大于
大于等于
小于
小于等于
正则表达式包含
正则表达式不包含

正则表达式包含和正则表达式不包含用于查找与表达式匹配的模式。越容易将字符串与表达式匹配，代理程序中的工作负载效率越高。表达式无需匹配整行；只需匹配表达式中的子字符串即可。例如，在 `See him run` 中，您想要知道字符串是否包含 `him`。可以使用 `him` 来编写正则表达式，但也可以使用 `.*him.*`。或者，如果要查找 `See`，可以输入 `See`，也可以输入 `^See` 以确认该单词位于行开头。输入 `.*` 通配符会导致搜索效率低下，并增加工作负载。有关正则表达式的更多信息，请参阅 developerWorks® technical library 主题或在浏览器中搜索 `regex`。

您还可以选择 `Missing` 函数，用于将指定度量的值与所提供的值列表进行比较。当该值与列表中的任何值均不匹配时，该条件为 true。如果您想要接收到系统中缺少某些内容的通知，那么可使用此函数。需求与限制：

1. 选中的度量必须为文本属性：不能使用时间属性和数字属性。
2. 请用逗号 (,) 分隔每个值，例如，`fred,mary,jean`。
3. 在每个阈值中只能包含一个 `Missing` 条件。
4. `Missing` 必须为公司中的最后一个条件。如果需要其他条件，请先输入这些条件，然后再添加 Missing 函数，并且，在公式中只能使用 **And (&)** 运算符。否则，将禁用所有后续行。

值
使用度量允许的格式来输入要比较的值，例如输入 20 表示 20%，或输入 120 表示 2 分钟。

组分配
分配资源组以将阈值分发到资源组中相同类型的受管系统。可用资源组为您有“修改”许可权的用户定义组以及您有“查看”许可权的系统组（针对代理程序类型）。可用系统组也限制为适用于所选数据集的那些组。

未分配任何组的阈值不会分发到任何受监视系统且保持停止，直至将其分发到资源组。

系统组（例如 `Linux OS` 或 `HTTP Server`）将阈值分发到代理程序安装所在的所有受管系统。缺省情况下，所有预定义阈值都会分配到该代理程序的系统组。（您可以按照第 912 页的『阈值启用』所述，在“**高级配置**”页面中禁用所有预定义阈值。）

IBM Tivoli Monitoring 域中的受管系统例外：必须使用 Tivoli Monitoring 环境中分发的情境来监视 Tivoli Monitoring 域中的受管系统。

要将组分配到阈值，请选择一个或多个资源组的复选框。如果分配的组的列表很长，那么可以选择 **仅显示选中的组**。

如果您看不到要对其分配阈值的资源组，则可保存阈值定义，然后在系统提示您确认要保存该阈值而不将其分配给任何组时，单击**确定**。然后，您可在“**资源组管理器**”中创建新组，并在“**资源组编辑器**”中分配阈值给这个新组。有关更多信息，请参阅第 833 页的『资源组管理器』。

执行命令
针对求值为 true 的阈值打开事件后，可以在打开了事件的受监视系统上自动运行命令或脚本。例如，您可能希望记录信息、触发蜂鸣声，或停止正在过度使用资源的作业。

此命令使用以下语法：

```
&{data_set.attribute}
```

其中，*data_set* 是数据集名称，*attribute* 是显示在阈值编辑器中的属性名称。如果数据集名称或属性名称中包含空格，请替换为下划线。

以下示例显示如何将磁盘名称参数传递至受管资源：

```
/scripts/clean_logs.sh &{KLZ_Disk.Disk_Name}
```

可以从数据集传入一个或多个属性。如果指定多个属性，那么各个属性将按顺序（$1、$2，等等）传递到命令中。

该命令将以用来启动代理程序的用户帐户从命令行运行。 例如，如果以 root 用户身份运行代理程序，那么 root 用户在受管系统上运行该命令。

以下选项控制命令运行的频率：

如果数据集返回多行，而您只想针对数据样本中出现的第一个事件运行命令，那么请选择 **仅对第一个事件**。要对导致事件打开的每行都运行命令，请清除此复选框。

选择 **对于每个连续 true 时间间隔**，从而每次阈值请求为 true 时运行命令。要在阈值为 true 时发出命令，但此后一直到阈值求值为 false 都不再执行此命令，直到后续时间间隔内再次求值为 true 时才再次发出命令，请清除此复选框。

在单击**保存**后，阈值将应用于分配的资源组中相同数据类型的所有受监视系统。

提示: 您可以通过“**高级配置**”页面中的**事件管理器**选项控制事件行为和事件转发。请参阅第 910 页的『高级配置』。

注: 要查看适合包含在阈值定义中的属性的列表，请使用计划使用的数据集创建表格。。

相关概念

第 830 页的『背景信息』
查看背景信息以了解阈值、代理程序的预定义阈值、阈值分配到的资源组以及定制阈值。

相关任务

第 835 页的『教程：定义阈值』
第 837 页的『教程：定义阈值以便在受管资源上运行命令』
您可使用“**阈值编辑器**”将特定参数传递给代理程序。您可指定当事件触发时，要自动运行的命令或命令脚本。

第 821 页的『与 Netcool/OMNIbus 集成』
您可以将 IBM Cloud Application Performance Management 中的事件转发到内部部署 IBM Tivoli Netcool/OMNIbus 事件管理器。

定制事件以转发到 EIF 接收器

您可以定制发送到事件集成工具 (EIF) 接收器的阈值事件，例如，Netcool/OMNIbus ObjectServer、Cloud Event Management 或 Alert Notification。使用“**EIF 槽定制**”窗口可定制转发到事件目标的事件内容，从而覆盖缺省映射。 您可以为发送到“事件集成工具”接收器的阈值事件创建映射定义。使用“**EIF 槽定制**”窗口可定制如何将事件映射到已转发的 EIF 事件，从而覆盖缺省映射。通过定制消息模板，可添加由事件和事件中特定数据确定的问题的相关信息。通过定制消息模板，可添加由事件确定的问题的相关信息并包含事件中特定数据。

关于此任务

您可以定制 EIF 基本槽，即向事件目标发送阈值公式的预定义 **msg** 槽。您还可以向事件添加一个或多个 EIF 定制槽。使用 Netcool/OMNIbus ObjectServer 时，如果想要在 Netcool/OMNIbus UI 中查看定制槽，那么必须更新 EIF 探测规则文件和 ObjectServer 触发器。

您可以定制 EIF 基本槽，即向 EIF 接收器发送阈值公式的预定义 **msg** 槽。您还可以添加一个或多个 EIF 定制槽，这需要对 EIF 接收器和探测规则文件进行更新。

过程

完成以下步骤以定制如何将当前阈值的事件映射至已转发的事件：

1. 如果“**阈值管理器**”未打开，请单击 **系统配置 > 阈值管理器**。

2. 单击**数据源类型**列表框并选择要使用的数据类型。
3. 如果这是新阈值，请单击 **新建**；否则，选择阈值，然后单击 **编辑**。
4. 要定制如何将此阈值的事件映射到已转发的事件，请确保将 **EIF 转发器**设置为是，单击 **EIF 槽定制**，然后执行以下某个步骤：
   - **EIF 基本槽**：要定制基本槽，请选中 **msg** 的单选按钮并单击 **编辑**。
   - **EIF 定制槽**：要添加定制槽，请单击 **添加**；要编辑定制槽，请选择槽的单选按钮并单击 **编辑**。

   将打开“**编辑槽**”或“**添加槽**”窗口。
5. 填写以下字段以定制槽值：

字段	描述	限制
槽名称	EIF 定制槽的名称，必须以字符开头。	EIF 基本槽是 **msg** 并且不能更改。
槽类型	EIF 定制槽的类型：**String Type** 或 **Number Type**。	EIF 基本槽是 **String Type** 并且不能更改。
子类型	分配给槽的值，与槽类型对应： · **Mapped Attribute** 启用**已映射属性**字段，以在事件发生时添加所选属性的值 · **Literal Value** 启用**字面值**字段，以将文本添加到消息模板 · **Literal Value + Mapped Attribute** 启用**字面值**和**已映射属性**字段以将文本值和属性值添加到消息模板，启用**添加**按钮以添加多个文本值和/或属性值。在每个字面值或属性的后面添加了空格。 EIF 基本槽 **msg** 的典型用法，即为消息模板指定 **Literal Value + Mapped Attribute**。	**Number Type** 槽可以只使用 **Mapped Attribute**。
添加	如果要在转发消息中发送多个字面值或属性值，单击**添加**以添加另外一组**字面值**和**已映射属性**字段。每当选择**添加**时，这些字段将添加到面板中。 要移除一组**字面值**和**已映射属性**字段，请先清除这两个字段，然后单击**确定**。 请参阅示例。	仅当**子类型**为 **Literal Value + Mapped Attribute** 时才启用。 最多 6 组**字面值**和**已映射属性**字段。 如果无法看到您添加的字段，请使用浏览器缩小功能 (Ctrl -) 来缩小布局以适应对话框。
字段值	要包含在消息模板中的文本。例如，包含映射属性 **%Memory Utilization** 的内存利用率的字面值在“**事件管理器**”用户界面中显示为内存利用率为 97.3%。 消息模板包含固定消息文本和变量替换引用，或符号。符号指的是通用或事件槽数据或者对阈值公式的特殊引用。通用槽是所有已转发的事件中包含的槽，如 *threshold_name*；事件槽是特定于阈值 msg 的槽。	当**子类型**为 **Mapped Attribute** 时禁用。
已映射属性	要将其值添加到消息模板的属性。这些可用属性来自为阈值选定的数据集。例如，对于监视高处理器时间的阈值，您可能想要映射用户时间百分比属性。	最多 6 个**已映射属性**字段。 当**子类型**为 **Literal Value** 时禁用。 当**槽类型**为 **Number Type** 时，只有数字属性可用。

字段	描述	限制
Multiplier	乘数是在定制原始映射属性数字值后通过乘以乘数定义的值：槽值 = *attribute1* * *n*。例如，如果在 EIF 事件中要将分钟转换为秒，那么将指定乘数 60。乘数值可以是分数，表示为小数，如 0.5 或 5.4。	仅对数字属性（**槽类型** 为 **Number Type**）启用。

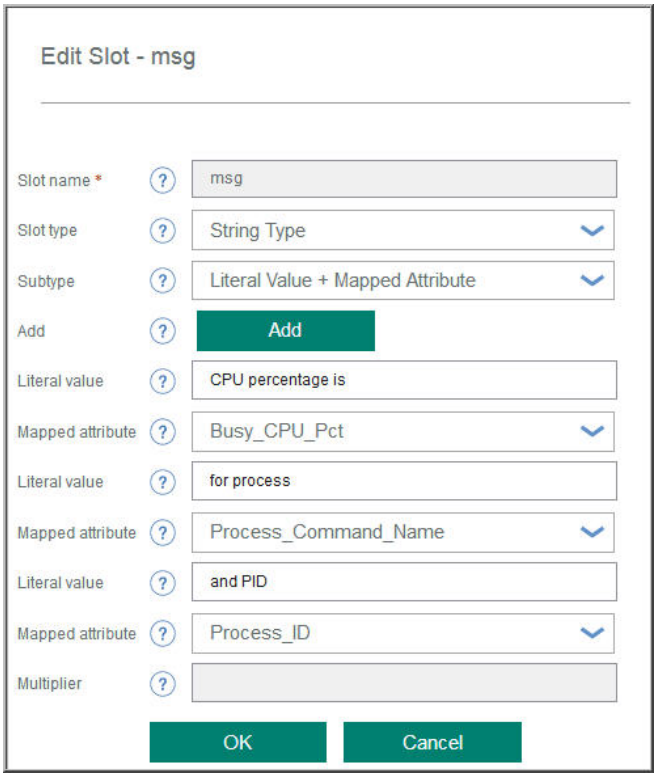
在单击**确定**以关闭窗口后，“**EIF 槽定制**”窗口列出槽名称以及该槽是否已定制。

6. 当完成编辑 EIF 基本槽或为阈值添加、删除或编辑 EIF 定制槽时，单击**确定**。
7. 完成编辑阈值后，单击**保存**。

有关更多信息，请参阅第 838 页的『阈值管理器』。

示例

`Linux_BP_ProcHighCpu_Critical` 阈值测定 CPU 使用量为 95% 或更高。为将繁忙 CPU 百分比、进程命令名和进程标识添加到摘要消息（包含在 msg 槽中），使用三组**字面值**和**已映射属性**字段定制了 msg 槽：



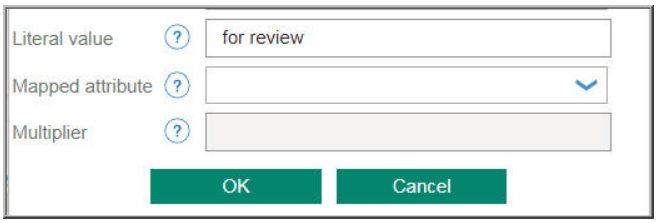
消息模板类似于以下内容：

```
CPU percentage is Busy_CPU_Pct for process Process_Command_Name and PID Process_ID
```

在事件管理器中查看的所产生的消息可能如下所示：

```
CPU percentage is 97 for process large.exe and PID 9876
```

您还可以添加**字面值**和**已映射属性**字段并将一个字段保留为空。例如，要将“for review”附加到消息模板，请单击**添加**并为**字面值**输入 `for review`。



消息模板类似于以下内容：

```
CPU percentage is Busy_CPU_Pct for process Process_Command_Name and PID Process_ID for review
```

在“**事件管理器**”中查看的所产生的消息可能如下所示：

```
CPU percentage is 96 for process big.exe and PID 5432 for review
```

下一步做什么

如果创建了新的 EIF 定制槽，您必须在 Netcool/OMNIbus ObjectServer 上的 `alerts.status` 表中识别新槽，然后更新在 Netcool/OMNIbus 与 Cloud APM 集成期间安装的 `itm_apm_event.rules` 配置文件。

将 EIF 定制槽添加到 Netcool/OMNIbus ObjectServer 数据库

在添加用于阈值的新 EIF 定制槽时，必须先在 EIF 接收方中标识这些槽，然后才能查看使用这些定制槽的转发事件。如果已将 Netcool/OMNIbus 与 Cloud APM 集成，更新 `alerts.status` 表以定义新槽。

关于此任务

在配置了 Netcool/OMNIbus 与 Cloud APM 的集成后，步骤 第 824 页的『4』会让您装入 `itm_apm_db_update.sql`。以下过程让您使用 SQL 交互式界面更新 `itm_apm_db_update.sql` 数据库中的 `alerts.status` 表。

过程

在 Netcool/OMNIbus ObjectServer 上完成这些步骤以定义您在“**阈值编辑器**”中创建的新 EIF 定制槽：

1. 启动 SQL 交互式界面编辑数据库：

   Linux

   ```
   $OMNIHOME/bin/nco_sql -user user_name -password password
   -server server_name > itm_apm_db_update.sql
   ```

   示例：

   ```
   $OMNIHOME/bin/nco_sql -user smadmin -password passw0rd -server NCOMS >
   /tmp/apm/itm_apm_db_update.sql
   ```

   Windows

   ```
   itm_apm_db_update.sql | %OMNIHOME%\..\bin\isql -U user_name
   -P password -S server_name
   ```

   示例：

   ```
   \temp\apm\itm_apm_db_update.sql | %OMNIHOME%\..\bin\isql -U smadmin
   -P passw0rd -S NCOMS
   ```

2. 对于每个 EIF 定制槽，按照下列各式输入包含定制槽名称和槽类型的 **ALTER TABLE** SQL 命令，按 Enter 键，然后输入 go 并按 Enter 键：

   - 对于字符串槽类型，

     ```
     alter table alerts.status add CustomSlotName varchar(512);
     ```

   - 对于数字槽类型，

     ```
     alter table alerts.status add CustomSlotName integer;
     ```

   其中，*CustomSlotName* 是 EIF 定制槽的名称，与“**阈值编辑器**”的“**添加槽**”窗口 **slotName** 字段中输入的完全相同。

示例

示例显示了用于添加 **BusinessApplication** 和 **GenericMetric** 定制槽的 **alter table** 命令。

```
alter table alerts.status add BusinessApplication varchar(512);
```

```
alter table alerts.status add GenericMetric integer;
```

下一步做什么

更新作为 Netcool/OMNIbus 与 Cloud APM 集成的一部分安装的 `itm_apm_event.rules` 配置文件。有关更多信息，请参阅第 845 页的『将 EIF 定制槽添加到 EIF 接收方事件规则』。

将 EIF 定制槽添加到 EIF 接收方事件规则

如果为阈值定义了新 EIF 定制槽，必须更新规则文件以向 EIF 接收方标识新槽。

关于此任务

以下步骤用于更新 Netcool/OMNIbus Probe for Tivoli EIF 上的 `itm_apm_event.rules` 文件以标识每个新的 EIF 定制槽。如果使用了另一个 EIF 接收方，根据接收方的要求更新规则文件。

过程

1. 在安装了 Probe for Tivoli EIF 的系统上，切换到安装目录。

   Linux cd *install_dir*/tivoli/netcool/omnibus/probes/linux2x86

   Windows cd *install_dir*\Tivoli\Netcool\omnibus\probes\win32

   其中，*install_dir* 是缺省目录 `/opt/IBM/` 或 `C:\IBM\` 或者在安装探测器时指定的目录。
2. 备份 `itm_apm_event.rules` 文件。
3. 在文本编辑器中打开 Probe for Tivoli EIF `itm_apm_event.rules` 文件。

   该文件包含三个部分，您要编辑这三个部分来添加已经创建的 EIF 定制槽。
4. 使用以下格式，通过新语句为每个 EIF 定制槽追加 **if** 语句：

   ```
   if(exists($CustomSlotName))
   {
       if(regmatch($CustomSlotName, "^'.*'$"))
       {
           $SourceType = extract($CustomSlotName, "^'(.*)'$")
       }
   }
   ```

   其中，*CustomSlotName* 是 EIF 定制槽的名称，与“**添加槽**”窗口 **slotName** 字段中输入的完全相同。
5. 使用以下格式，在新行中为每个 EIF 定制槽追加 `@` 条目列表：

   ```
   @CustomSlotName=$CustomSlotName
   ```

   其中，*CustomSlotName* 是 EIF 定制槽的名称。
6. 使用以下格式，在新行中为每个 EIF 定制槽追加 `$tmpEventData` 条目列表：

   ```
   $tmpEventData = nvp_remove( $tmpEventData, "CustomSlotName")
   ```

   其中，*CustomSlotName* 是 EIF 定制槽的名称。
7. 保存并关闭 `itm_apm_event.rules` 文件。
8. 重新启动 Probe for Tivoli EIF 以实施更新。

结果

规则文件已更新，并且 Probe for Tivoli EIF 现在能够处理使用新 EIF 定制槽并将事件详细信息转发到 Netcool/OMNIbus ObjectServer 的阈值事件。

示例

以下是从为添加这些 EIF 定制槽而编辑的 `itm_apm_event.rules` 中抽取的摘录：**BusinessApplication** 和 **GenericMetric**（斜体显示）。

```
# ------------------------------------------------------------
    if(exists($SourceID))
    {
        if(regmatch($SourceID, "^'.*'$"))
        {
            $SourceID = extract($SourceID, "^'(.*)'$")
        }
    }
    ...
    ...
    ...
    if(exists($ManagedSystemGroups))
    {
        if(regmatch($ManagedSystemGroups, "^'.*'$"))
        {
            $SourceType = extract($ManagedSystemGroups, "^'(.*)'$")
        }
    }
    if(exists($BusinessApplication))
    {
        if(regmatch($BusinessApplication, "^'.*'$"))
        {
            $SourceType = extract($BusinessApplication, "^'(.*)'$")
        }
    }
         if(exists($GenericMetric))
    {
        if(regmatch($GenericMetric, "^'.*'$"))
        {
            $SourceType = extract($GenericMetric, "^'(.*)'$")
        }
    }

 @SourceID=$SourceID
 @URL=$ManagementURL
 @Service=$Service
 @SourceType=$SourceType
 @SubscriberID=$TenantID
 @APMHostname=$apm_hostname
 @ManagedSystemGroups=$ManagedSystemGroups
 @BusinessApplication=$BusinessApplication
 @GenericMetric=$GenericMetric

 # ------------------------------------------------------------------
 # -  RTC 66157
 # ------------------------------------------------------------------
 if ( exists ( $appl_label ) )
 {
    if ( match($appl_label, "PI:A:S"))
    {
        @Class = 87723
    }
 }
 # ------------------------------------------------------------------
 # -  RTC 48775 - APM FP5 agents do not populate data in email of EMaaS Basic
 # ------------------------------------------------------------------
 if (match( $situation_eventdata, "~" ) )
 {
    # Dump all fields into the ITMEventData field
    $tmpEventData = nvp_add($*)
    # Remove the duplicated fields
    $tmpEventData = nvp_remove( $tmpEventData, "appl_label")
    $tmpEventData = nvp_remove( $tmpEventData, "control")
    ...
    ...
    ...
    $tmpEventData = nvp_remove( $tmpEventData, "ManagedSystemGroups")
    $tmpEventData = nvp_remove( $tmpEventData, "EventSeqNo")
    $tmpEventData = nvp_remove( $tmpEventData, "BusinessApplication")
    $tmpEventData = nvp_remove( $tmpEventData, "GenericMetric")
       @ITMEventData = $tmpEventData
```

发送电子邮件以响应事件

仅限 当受管环境包含 IBM Alert Notification 时，如果应用程序性能超出阈值，您可以提供电子邮件通知。

关于此任务

要配置电子邮件通知，必须启用 IBM Alert Notification，如 IBM Knowledge Center 上的“警报通知”主题集合所述。然后，将受监视应用程序添加到资源组。对于每个资源组，可以配置一个或多个电子邮件地址。组中的任何应用程序的性能超过阈值时，将会向您为该组配置的地址发送电子邮件通知。

过程

1. 单击 **系统配置** > **资源组管理器**。
2. 单击 **新建**以创建要为其配置电子邮件通知的资源组，或选择现有组，然后单击 **编辑**。
   资源组编辑器将打开。
3. 单击**配置电子邮件通知**以在新的浏览器选项卡或窗口中打开 IBM Alert Notification 应用程序。使用 Alert Notification 以创建用户并将其电子邮件地址与资源组相关联，以供通过电子邮件接收事件通知。

使用“资源组管理服务”API

使用资源组管理服务 API 可以从命令行管理受管系统组的生命周期。

关于此任务

完成资源组任务，例如创建、查看、更新和删除受管系统组。对定制组添加和移除各个系统。查看已添加到特定定制资源组的系统列表，以及查看自动添加到内置组（例如，系统资源组）的系统列表。

您可以创建脚本来自动执行一些任务，例如，定义资源组以及将代理程序分配到这些资源组。资源组可以是阈值分布和/或访问控制策略的目标。

您可在 API Explorer 中以及本主题末尾的“示例”中找到有关下列操作的描述。

· 返回所有资源组、代理程序，或特定资源组或代理程序。

· 创建定制资源组或更新现有组的定义

· 删除指定的定制资源组

· 将代理程序添加到定制资源组

· 从定制资源组中移除代理程序

过程

完成以下步骤以使用“资源组管理服务”API 定义和更改定制资源组。不能修改系统资源组和代理程序。

1. 完成探查 API 主题中的步骤 1 到步骤 9。
   步骤 10 和步骤 11 提供额外的详细信息。
2. 单击**使用**并选择密钥，例如，**Key1**。

   注: 单击**隐藏**以显示客户机标识和客户机密钥。请记下这些信息，因为在 API Explorer 外部使用外部工具执行 API 调用时，将需要这些信息。然后，单击**显示**进行隐藏。
3. 填充所有以星号标记的必填标头。

   X-IBM-Service-Location
   * 标头是您的预订的地理位置，例如 na 表示北美地区

   授权
   * 标头是 IBM 标识和密码的基本 64 位编码字符串。使用基本 64 位编码器工具对 IBM 标识和密码编码时，格式必须为 *IBMid:password*，例如，`Basic YXBtYWRtaW46YXBtcGFzcw==!`。
4. 必须在所有的 POST、PUT 和 DELETE 请求中包括 referer 标头。Referer 标头的值始终如下所示:

```
-H 'Referer: https://api.ibm.com'
```

5. 滚动以找到并单击**测试**。

结果

您在 API 中对定制资源组所做的更改会立即生效，并显示在“**资源组管理器**”中（请参阅第 833 页的『资源组管理器』）。

示例

此命令返回所有代理程序的名称、唯一标识、状态、主机名、版本和代理程序类型：

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes=Agent&_field=keyIndexName&
_field=online&_field=hostname&_field=version&_field=productCode&_field=description
```

此命令将返回所有 Linux OS 代理程序的列表：

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes=Agent&_filter=description=
"Linux OS"&_field=keyIndexName
```

此命令返回系统和定制组的列表：

```
GET /1.0/topology/mgmt_artifacts?_filter=entityTypes:AgentGroup,
AgentSystemGroup&_field=keyIndexName&_field=displayLabel
```

此命令返回代理程序的列表，它们将分配给具有唯一标识 {id} 的组：

```
GET /1.0/topology/mgmt_artifacts/{id}/references/to/contains
```

以下示例使用 curl 命令创建定制组。

```
POST /1.0/topology/mgmt_artifacts
```

注: POST 请求的主体必须包含定义该组的 JSON 对象，如 **-d** 参数所示。

```
curl -X POST \
  https://api.ibm.com/perfmgmt/run/1.0/topology/mgmt_artifacts \
  -H 'Referer: https://api.ibm.com' \
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' \
  -H 'content-type: application/json' \
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' \
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' \
  -d '{
  "keyIndexName": "customGroup",
  "description": "Custom group description",
  "displayLabel": "customGroupLabel",
  "entityTypes": [
    "AgentGroup"
  ],
  "arbitraryStringProperty": "Your custom property value"
}'
```

此命令将具有唯一标识 {otherid} 的代理程序添加到具有唯一标识 {id} 的定制组：

```
POST /1.0/topology/mgmt_artifacts/{id}/references/to/contains/{otherid}
```

此命令从具有唯一标识 {id} 的定制组中移除具有唯一标识 {otherid} 的代理程序：

```
DELETE /1.0/topology/mgmt_artifacts/{id}/references/to/contains/{otherid}
```

使用“阈值管理服务”API

使用阈值管理服务 API 可以从命令行管理监视阈值的生命周期。

关于此任务

完成阈值管理器任务，例如创建、查看、更新和删除阈值。将资源组分配到这些阈值。查看所有阈值和资源分配的列表。查看分配给特定资源组的所有阈值的列表。

您可以创建脚本来自动执行一些任务，例如，定义阈值以及将其分配给资源组。

您可在 API Explorer 中以及本主题末尾的“示例”中找到有关下列操作的描述。

- 返回所有阈值或获取特定阈值。可以使用以下属性来过滤请求：**label**，对应于阈值名称；**_appliesToAgentType**，对应于双字符产品代码；**_uiThresholdType**，对应于 Cloud APM 控制台的“阈值管理器”和“资源组”编辑器页面中显示的阈值类型。在获取阈值时可以使用 **_offset** 或 **_limit**
- 创建阈值或更新现有阈值的定义。对于更新请求，必须包括 X-HTTP-Method-Override 标头并设置为 PATCH
- 删除指定的阈值
- 返回所有资源分配或特定资源分配，这样会显示分配到每个资源组的阈值。您可以使用以下属性过滤请求：**resource._id** 和 **threshold._id**；并使用以下支持的运算符：**=**（等于）和 **!=**（不等于）
- 创建资源分配，这样会将单个阈值分配到单个资源组
- 删除资源分配，这样会从单个资源组移除单个阈值

过程

1. 完成探查 API 主题中的步骤 1 到步骤 9。
   步骤 10 和步骤 11 提供额外的详细信息。
2. 单击**使用**并选择密钥，例如，**Key1**。

   注: 单击**隐藏**以显示客户机标识和客户机密钥。请记下这些信息，因为在 API Explorer 外部使用外部工具执行 API 调用时，将需要这些信息。然后，单击**显示**进行隐藏。
3. 填充所有以星号标记的必填标头。

   X-IBM-Service-Location
   * 标头是您的服务器的地理位置，例如 na 表示北美地区

   Authorization
   * 这个标头是 IBMid 和密码的基本 64 位编码字符串。使用基本 64 位编码器工具对 IBMid 和密码编码时，格式必须为 *IBMid:password*。例如，`Basic YXBtYWRtaW46YXBtcGFzcw==!`。
4. 滚动以找到并单击**测试**。

示例

此命令返回注册到服务器的所有阈值：

```
GET /threshold_types/itm_private_situation/thresholds
```

此命令返回标签（名称）为 `My_threshold` 的阈值的信息。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=label%3DMy_threshold
```

此命令返回代理程序类型为 LZ（这是 Linux OS 代理程序的组件代码）的所有阈值。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=_appliesToAgentType%3DLZ
```

此命令的输出与上一命令相同，但给出了在 Cloud APM 控制台中显示的代理程序名称。

```
GET /threshold_types/itm_private_situation/thresholds?_filter=_uiThresholdType%3DLinux OS
```

此命令返回将阈值 123 分配到的所有资源组：

```
GET /resource_assignments?_filter=threshold._id=123
```

以下示例使用 curl 命令创建新阈值。

```
POST /1.0/thresholdmgmt/threshold_types/itm_private_situation/thresholds
```

切记: POST 请求的主体必须包含定义阈值的 JSON 对象，如 **-d** 参数所示。示例：

```
curl -X POST\
```

```
 https://api.ibm.com/perfmgmt/run/1.0/thresholdmgmt/threshold_types/itm_private_situation/
thresholds  \
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' \
  -H 'content-type: application/json' \
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' \
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' \
  -d '{
  "label": "Your_Linux_Threshold_Name",
  "description": "Your Linux Threshold Definition",
  "configuration": {
    "type": "json",
    "payload": {
      "formulaElements": [
        {
          "function": "*MKTIME",
          "metricName": "KLZ_CPU.Timestamp",
          "operator": "*EQ",
          "threshold": "1455767100000",
          "timeDelta": {
            "operator": "+",
            "delta": "3",
            "unit": "Hours"
          }
        }
      ],
      "period": "011500",
      "periods": "3",
      "severity": "Fatal",
      "matchBy": "KLZCPU.CPUID",
      "operator": "*OR",
      "actions": [
        {
          "name": "command",
          "command": "ps -ef",
          "commandFrequency": "Y",
          "commandWhen": "Y",
          "commandWhere": "N"
        }
      ]
    }
  }
}'
```

管理用户访问权

使用 Cloud APM 中“基于角色的访问控制”功能可授予用户其角色所需的访问权限。

Cloud APM 中的安全性基于角色。角色是控制您可以在 Cloud APM 中执行的操作的一组许可权。您可以在 Cloud APM 中创建定制角色。可以为定制的角色分配许可权，或者可以为现有缺省角色分配更多许可权。您可以将用户和用户组分配到现有缺省角色或定制角色。您可以将用户和用户组分配到多个角色。许可权是累积的，用户或用户组所分配到的许可权是他们分配到的所有角色的所有许可权。

Cloud APM 中的用户认证通过 IBM Marketplace 进行管理，或者可以通过 Collaborative Operations（如果有预订）进行管理。

Performance Management 中的用户认证需要 IBM 标识。请选择**登录 IBM** 页面上的**创建 IBM 标识**链接以创建 IBM 标识。要访问**登录 IBM** 页面，请转至 IBM Marketplace 上的**“产品和服务”** (http://ibm.biz/my-prodsvcs) 页面，这样您会重定向到**登录 IBM** 页面（您可能需要注销并返回到**“产品和服务”** (http://ibm.biz/my-prodsvcs) 页面）。

在获得 IBM 标识后，可以直接或从“产品和服务”登录 Cloud APM。访问 Cloud APM 实例的授权通过**“产品和服务”**或通过 Collaborative Operations（如果有预订）进行管理。缺省情况下，请求试用版或购买预订的用户具有管理员特权。

Cloud APM 预订的所有者是 Cloud APM 中的缺省用户。这是请求试用版或购买预订的用户。此缺省用户是“角色管理员”角色的成员，具有允许此用户添加新用户的管理员特权。后续用户缺省情况下会添加到“监视用户”角色。

如果您不是某个角色的成员，而尝试登录 Cloud APM，那么将收到**未经授权**消息。

要添加 Cloud APM 用户：

1. 转至 IBM Marketplace 上的“**产品和服务**”，并展开 **IBM Performance Management** 窗口小部件。
2. 单击**管理权限**并在**添加新用户**或**搜索现有用户**字段中输入 IBM 标识或电子邮件地址。单击**添加用户**以添加用户。

对于可以添加到 Cloud APM 预订的用户没有限制。

注: 不支持在 Cloud APM 中使用用户组。

有关角色的更多信息，请参阅第 851 页的『角色和许可权』。

角色和许可权

角色是控制您可以在 Cloud APM 中执行的操作的一组许可权。使用“基于角色的访问控制”页面来管理用户和角色，或者使用授权 API 从命令行完成基于角色的访问控制任务。有关更多信息，请参阅第 910 页的『浏览 API』。

Cloud APM 具有四个缺省角色:

角色管理员
此角色旨在用于其主要作业功能是为 Cloud APM 创建访问控制策略的用户。此角色具有所有许可权。如果更改缺省用户，那么新的缺省用户将自动成为“角色管理员”角色的成员。无法编辑此角色。会阻止“角色管理员”从“角色管理员”角色中移除自身。此限制移除意外移除“角色管理员”角色中的所有用户的风险。

监视管理员
此角色旨在用于其主要作业功能是使用 Cloud APM 监视系统的用户。“监视管理员”以下任务：添加监视应用程序、创建阈值、添加资源组，以及将阈值分发到这些资源组。无法编辑此角色。

系统管理员
此角色旨在用于其主要作业功能是执行 Cloud APM 系统的管理任务的用户。“系统管理员”执行配置事件管理器或配置 Hybrid Gateway 等任务。无法编辑此角色。

监视用户
此角色旨在用于其主要作业功能是配置和维护 Cloud APM 监视的系统的运行状况和状态的用户。无法编辑此角色。

下表描述可分配给角色的许可权，以及四个可用的缺省角色和关联的许可权:

表 *237.* 角色和许可权

	角色管理员		**监视管理员**		**系统管理员**		**监视用户**	
	查看	**修改**	**查看**	**修改**	**查看**	**修改**	**查看**	**修改**
系统配置许可权								
高级配置	✓	不适用	—	不适用	✓	不适用	—	不适用
代理程序配置	✓	不适用	✓	不适用	—	不适用	—	不适用
参考页面	✓	不适用	✓	不适用	✓	不适用	✓	不适用
搜索提供程序	✓	不适用	✓	不适用	—	不适用	—	不适用
使用情况统计信息	✓	不适用	✓	不适用	—	不适用	—	不适用
资源许可权								
应用程序性能仪表板	✓	✓	✓	✓	✓	—	✓	—
应用程序	✓	✓	✓	✓	—	—	✓	—
单个应用程序	第 854 页的『应用程序和资源组许可权』							
诊断仪表板	✓	不适用	—	不适用	—	不适用	—	不适用
资源组管理器	✓	不适用	✓	不适用	—	不适用	—	不适用
单个资源组	第 854 页的『应用程序和资源组许可权』							

表 237. 角色和许可权 (续)								
	角色管理员		监视管理员		系统管理员		监视用户	
	查看	修改	查看	修改	查看	修改	查看	修改
资源组	✓	✓	✓	✓	–	–	–	–
合成脚本管理器	✓	不适用	–	不适用	–	不适用	–	不适用
阈值管理器	✓	不适用	✓	不适用	–	不适用	–	不适用

其中

✓ 表示此角色成员具有此许可权

– 表示此角色成员不具有此许可权

N/A 指示此许可权不存在

注: 虽然**使用情况统计信息**显示在**系统配置许可权**的列表中，但是不再适用于 Cloud APM。

下表描述与每个许可权相关联的操作:

表 238. 许可权	
许可权	**描述**
高级配置	如果您具有查看许可权，那么可以执行下列任务: · 查看菜单栏中的 **系统配置 > 高级配置**。 · 在**高级配置**窗口中进行更改并保存。 · 查看菜单栏中的 **系统配置 > Hybrid Gateway Manager**。 · 在 **Hybrid Gateway Manager** 窗口中进行更改并保存。
代理程序配置	如果您具有查看许可权，那么可以执行下列任务: · 查看菜单栏中的 **系统配置 > 代理程序配置**。 · 在**代理程序配置**窗口中进行更改并保存。
参考页面	如果您具有查看许可权，那么可以执行下列任务: · 查看菜单栏中的 **入门**和 **帮助**。 **注:** 在**入门**页面打开时，如果清除**启动时显示此页面**，那么以后登录时，您将看到许可权被拒绝错误。但是，您仍能够浏览至**入门**页面以及具有许可权的任何其他区域。
搜索提供程序	如果您具有查看许可权，那么可以执行下列任务: · 查看菜单栏中的 **系统配置 > 配置搜索提供程序**。 · 在**配置搜索提供程序**页面中进行更改并保存更改。

表 *238.* 许可权 *(续)*	
许可权	**描述**
应用程序性能仪表板	如果您具有查看许可权，那么可以执行下列任务： · 查看菜单栏中的 **性能 > 应用程序性能仪表板**。 · 查看 应用程序性能仪表板 以及“**我的组件**”和“**我的事务**”预定义应用程序。 **注：** 要确定在“我的组件”应用程序中查看系统所需的许可权，请参阅第 854 页的『应用程序和资源组许可权』。 **注：** 仅在使用“Web 站点监视”时才显示“我的事务”应用程序。这将在“我的事务”应用程序中显示所有“Web 站点监视”合成事务。 · 在“定制视图”选项卡中打开定制仪表板页面。 · 在“属性详细信息”选项卡中创建视图并保存以供自己使用。 如果您具有修改许可权，那么可以执行下列任务： · 查看菜单栏中的 **性能 > 应用程序性能仪表板**。 · 查看 应用程序性能仪表板 以及“**我的组件**”和“**我的事务**”预定义应用程序。 **注：** 要确定在“我的组件”应用程序中查看系统所需的许可权，请参阅第 854 页的『应用程序和资源组许可权』。 **注：** 仅在使用“Web 站点监视”时才显示“我的事务”应用程序。这将在“我的事务”应用程序中显示所有“Web 站点监视”合成事务。 · 在“定制视图”选项卡中创建并保存定制仪表板页面。 · 在“属性详细信息”选项卡中创建视图并与其他人共享。 · 在组件页面中查看**操作 > 编辑**选项，此选项使您能够编辑在“组件”仪表板中显示的阈值以及组窗口小部件的其他设置。
应用程序	如果您具有查看许可权，那么可以执行下列任务： · 在应用程序仪表板中查看应用程序。 如果您具有修改许可权，那么可以执行下列任务： · 在应用程序仪表板中查看应用程序 · 使用 应用程序仪表板 中的 工具来创建、修改和删除应用程序。
单个应用程序	请参阅第 854 页的『应用程序和资源组许可权』。
资源组管理器	如果您具有查看许可权，那么可以执行下列任务： · 查看菜单栏中的 **系统配置 > 资源组管理器**。

表 *238.* 许可权 *(续)*	
许可权	**描述**
资源组	如果您具有查看许可权，那么可以执行下列任务： · 在“资源组管理器”中查看资源组以及其中的系统（如果您还具有“资源组管理器”查看许可权。 · 如果您还具有“应用程序性能管理”查看许可权或修改许可权，那么可以在“我的组件”预定义应用程序中查看系统 · 如果您还有权修改应用程序，那么可在“**添加应用程序**”窗口中查看系统。 如果您具有修改许可权，那么可以执行下列任务： · 在“资源组管理器”中查看资源组及其内容（如果您还具有“资源组管理器”查看许可权）。 · 如果您还具有 Application Performance Management 查看或修改许可权，那么在“我的组件”预定义应用程序中查看系统。 · 如果您还有权修改应用程序，那么可在“**添加应用程序**”窗口中查看系统。 · 在“资源组管理器”中创建、修改和删除资源组（如果您还具有“资源组管理器”查看许可权。要将阈值分配给资源组，您还需要成为具有“阈值管理器”的查看许可权的角色的成员。） **注:** “资源组管理器”用于按组组织受监视的系统，因此可以将阈值分配到这些组。如果您没有“阈值管理器”的查看许可权，那么无法查看针对“资源组”指定的阈值。如果向角色分配“修改资源组”许可权，那么您同时还需要向角色分配“查看阈值管理器”许可权。
单个资源组	请参阅第 854 页的『应用程序和资源组许可权』。
阈值管理器	如果您具有查看许可权，那么可以执行下列任务： · 查看菜单栏中的 **系统配置 > 阈值管理器**。 · 在阈值管理器中创建、修改和删除阈值。 · 如果您具有资源组的适当许可权，可以在阈值管理器中查看和编辑阈值的组分配。 · 或者，如果您有资源组管理器和资源的适当许可权以及阈值管理器的查看许可权，可以在资源组管理器中查看和编辑资源组的阈值分配。
Synthetic Script Manager	如果您具有查看许可权，那么可以执行下列任务： · 在合成事务管理器中创建、修改和删除合成事务。 **注:** 要在合成事务管理器中处理合成事务，还需要成为具有**代理程序配置**查看许可权的角色成员。
诊断仪表板	如果您有查看许可权，那么 WebSphere Applications 代理程序、Node.js 代理程序、Ruby 代理程序和 Microsoft .NET 代理程序的诊断仪表板上的“**诊断**”按钮处于启用状态。单击“**诊断**”按钮可以向下钻取到诊断仪表板。

应用程序和资源组许可权
可以向单个应用程序和资源组分配许可权。

应用程序许可权

在 Cloud APM 中，应用程序由一些组件以及这些组件中的实例构成。使用“**添加应用程序**”窗口以定义应用程序。有关如何定义应用程序的更多信息，请参阅管理应用程序。

要选择 Cloud APM 控制台 中的**性能 > 应用程序性能仪表板**，您必须分配有应用程序性能仪表板的查看或修改许可权。通过此许可权，您还可以查看“**我的组件**”和“**我的事务**”预定义应用程序。仅当您使用的是

“Web 站点监视”时，才会显示“**我的事务**”应用程序。要查看其他定制应用程序，您必须具有所有应用程序或个别应用程序的查看许可权或修改许可权。

注: 如果应用程序已重命名，那么不会保留许可权；您必须重新分配查看和修改许可权。

查看

应用程序的查看许可权优先于任何其他许可权。要查看应用程序，您无需成为具有应用程序中每个组件和组件实例的查看许可权的角色的成员。下表描述了具有应用程序查看许可权时可以执行的操作：

表 *239.* 应用程序的查看许可权

操作	**可用许可权**
查看此应用程序中的所有支持组件。	✓
在导航树中查看应用程序及其组件。	✓
在“我的组件”中查看应用程序组件。	✓
查看与应用程序关联的定制仪表板页面。	✓
向应用程序添加组件或从中移除组件。	—
指定应用程序组件的阈值。	—
在“资源组管理器”中查看应用程序的支持组件。	—

修改

如果您是具有单个应用程序修改许可权的角色的成员，那么您可以

- 删除此应用程序。
- 在“定制视图”选项卡中创建定制仪表板页面。请参阅第 942 页的『定制视图』。
- 使用“**编辑应用程序**”窗口添加或移除组件和组件实例。将基于您的角色许可权过滤“**编辑应用程序**”窗口中对您可用的组件和组件实例。可应用以下组件：
  - 具有直接访问系统资源组和定制资源组的许可权的组件
  - 基于具有修改许可权的其他应用程序间接继承许可权的组件

资源组许可权

使用资源组以按类型或用途将组件分组到一起。有关如何创建资源组的更多信息，请参阅第 833 页的『资源组管理器』。

要选择**系统配置 > 资源组管理器**，您必须分配有资源组管理器查看许可权。要在“**资源组管理器**”中查看资源组或在“**我的组件**”应用程序中查看资源组成员，您还必须分配有所有资源组或个别资源组的查看或修改许可权。

存在两种不同类型的资源组：定制资源组和系统资源组。

定制定义的资源组

在“资源组管理器”中创建定制资源组。使用定制资源组以按用途将组资源分组到一起。

下表描述在您具有定制资源组的查看许可权时可执行的操作：

表 *240.* 定制资源组的查看许可权

操作	**可用许可权**
在资源组管理器中查看定制资源组以及其中的资源。	✓
如果您还具有应用程序的修改许可权，请在“**添加应用程序**”窗口中查看属于定制资源组的资源。	✓

表 240. 定制资源组的查看许可权 (续)	
操作	**可用许可权**
如果您还具有应用程序性能仪表板许可权之一，请在“**我的组件**”预定义应用程序中查看属于定制资源组的资源。	✓
向定制资源组添加资源	–
删除定制资源组中的资源。	–

下表描述在您具有定制资源组的修改许可权时可执行的操作：

表 241. 定制资源组的修改许可权	
操作	**可用许可权**
在“阈值编辑器”将阈值分配给定制资源组。 **注:** 要分配阈值，您还需要成为具有“阈值编辑器”的查看许可权的角色的成员。	✓
向定制资源组添加资源	✓
删除定制资源组中的资源。	✓

系统资源组

系统资源组在 Cloud APM 环境设置中自动定义。无法手动创建、删除或定制系统资源组。仅查看许可权适用于系统资源组，修改许可权不适用。

在资源对 Cloud APM 服务器而言变为已知资源时，会为每种资源类型定义系统资源组。对于连接到 Cloud APM 服务器的每种资源类型都存在一个系统资源组。

Cloud APM 代理程序是一个资源示例。例如，第一次下载、安装和启动 Db2 代理程序时，将创建名为 Db2 的系统资源组。此组包含后续添加到 Performance Management 环境的所有 Db2 代理程序。

每种资源类型的系统资源组包含该类型的所有资源，包括 IBM Tivoli Monitoring。如果环境同时包含 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management，那么可以安装 IBM Cloud Application Performance Management Hybrid Gateway 以提供来自两个域的代理程序视图。系统定义的资源组包含来自两个域的代理程序。有关更多信息，请参阅第 807 页的『与 IBM Tivoli Monitoring V6.3 集成 』。

某些系统资源组基于子节点代理程序。您可以将阈值分配给基于子节点代理程序的系统资源组，事件不会在应用程序性能仪表板中显示。将阈值分配给基于用于事件转发的子节点代理程序的系统资源组。基于子节点代理程序的系统资源组在“资源组管理器”中具有以下描述：“该组的成员无法添加到应用程序并且不会在 Performance Management 控制台中显示事件”。有关更多信息，请参阅第 833 页的『资源组管理器』。

下表描述在具有系统资源组的查看许可权时可执行的操作：

表 242. 系统资源组的查看许可权	
操作	**可用许可权**
在“资源组管理器”中查看系统系统资源。	✓
如果您还具有应用程序的修改许可权，请在“**添加应用程序**”窗口中查看属于系统资源组的资源。	✓
如果您还具有应用程序性能仪表板许可权之一，请在“**我的组件**”预定义应用程序中查看属于系统资源组的资源。	✓
在“阈值编辑器”将阈值分配给系统资源组。	✓

表 242. 系统资源组的查看许可权 (续)	
操作	**可用许可权**
向系统资源组添加资源	—
删除系统资源组中的资源。	—

处理角色、用户和许可权

使用“基于角色的访问控制”页面以处理角色、用户和许可权。

开始之前

或者使用授权 API 从命令行完成基于角色的访问控制任务。有关更多信息，请参阅 第 910 页的『浏览 API』。

注: Cloud APM 中不支持用户组。

过程

- 要过滤在“基于角色的访问控制”页面中显示的角色、用户或用户组列表，请完成下列步骤:
  a) 选择 **系统配置 > 基于角色的访问控制**。
  b) 单击 **“过滤器”**文本框，然后输入部分或完整的过滤文本。
  在输入时，将从表中移除不包含在过滤器框中输入的内容的任何行。
  c) 要移除快速过滤，请删除值或者单击 “x”。
  d) 要应用过滤，请单击 。
- 要创建新的定制角色，请完成下列步骤:
  a) 选择 **系统配置 > 基于角色的访问控制**。
  b) 在**角色**选项卡中，单击 。**角色编辑器**页面随即显示。
  c) 在**将用户分配到角色**选项卡中，选择**用户组**选项卡或**单个用户**选项卡，然后选择想要添加到角色的用户和用户组。
  d) 在**将许可权分配到角色**选项卡中，选择**系统配置许可权**或**资源许可权**选项卡，然后选择想要分配到角色的许可权。
  e) 单击**保存**。
- 要编辑现有缺省或定制角色，请完成下列步骤:
  a) 选择 **系统配置 > 基于角色的访问控制**。
  b) 在**角色**选项卡中，单击 。**角色编辑器**页面随即显示。
  c) 在**将用户分配到角色**选项卡中，单击**用户组**选项卡或**单个用户**选项卡，然后选择想要添加到角色的用户或用户组。
  d) 在**将许可权分配到角色**选项卡中，选择**系统配置许可权**或**资源许可权**选项卡，然后选择想要分配到角色的许可权。
  e) 单击**保存**。
- 要删除角色，请完成下列步骤:
  a) 选择 **系统配置 > 基于角色的访问控制**。
  b) 在**角色**选项卡中，选择想要删除的角色，然后单击 。此时将显示一条确认消息，请单击**确定**。

  注: 在删除角色时，不会删除属于此角色的成员的用户。他们仍在**单个用户**选项卡中，并且角色管理员可将其分配到其他角色。
- 要编辑单个用户或用户组的许可权，请完成下列步骤:
  a) 选择 **系统配置 > 基于角色的访问控制**。
  b) 在**单个用户**或**用户组**选项卡中，选择想要编辑的用户或用户组，然后单击 。此时**单个用户编辑器**页面将打开。

c) 选择想要分配到用户的角色。

d) 单击**保存**。

- 要创建汇总用户或用户组的许可权的 csv 文件，请完成下列步骤：

a) 在**单个用户**或**用户组**选项卡中，选择必需的用户或用户组，然后单击 。此时**单个用户编辑器**或**用户组编辑器**页面将打开。

b) 单击**导出摘要**。

c) 选择**保存文件**，然后单击**确定**。

此时会将汇总用户或用户组许可权的 csv 文件保存到指定的位置。

结果

在单击**保存**时，角色和许可权分配将立即生效。

访问和使用基于角色的访问控制服务 API

使用基于角色的访问控制服务 API 可以从命令行管理基于角色的访问控制策略的生命周期。

关于此任务

完成基于角色的访问任务，例如创建、查看、更新和删除角色。对特定的角色添加和删除一组用户或用户组。将许可权授予特定角色。查看系统中定义的角色、用户、用户组和许可权的列表。

您可以创建脚本来自动执行一些任务，例如，定义新角色，以及将用户、用户组和许可权分配给这些角色。

过程

1. 完成探查 API 主题中的步骤 1 到步骤 9。
   步骤 10 和步骤 11 提供额外的详细信息。
2. 单击**使用**并选择密钥，例如，**Key1**。

   注: 单击**隐藏**以显示客户机标识和客户机密钥。请记下这些信息，因为在 API Explorer 外部使用外部工具执行 API 调用时，将需要这些信息。然后，单击**显示**进行隐藏。
3. 填充所有以星号标记的必填标头。

   X-IBM-Service-Location
   * 标头是您的预订的地理位置，例如 na 表示北美地区

   授权
   * 标头是 IBM 标识和密码的基本 64 位编码字符串。使用基本 64 位编码器工具对 IBM 标识和密码编码时，格式必须为 *IBMid:password*，例如，`Basic YXBtYWRtaW46YXBtcGFzcw==!`。
4. 必须在所有的 POST、PUT 和 DELETE 请求中包括 referer 标头。Referer 标头的值始终如下所示：

```
-H 'Referer: https://api.ibm.com'
```

5. 滚动以找到并单击**测试**。

示例

以下示例使用 curl 命令创建新角色。

```
POST /1.0/authzn/roles
```

注: POST 请求的主体必须包含定义该角色的 JSON 对象，如 **-d** 参数所示。

```
curl -X POST \
  https://api.ibm.com/perfmgmt/run/1.0/authzn/roles  \
  -H 'Referer: https://api.ibm.com' \
  -H 'authorization: Basic REPLACE_BASE64_ENCODED_STRING' \
  -H 'content-type: application/json' \
  -H 'x-ibm-client-id: REPLACE_KEY_VALUE' \
  -H 'x-ibm-client-secret: REPLACE_KEY_VALUE' \
```

```
  -d '{
    "description": "Your Role Description",
    "id": "/authzn/roles/Your_Role_Id",
    "label": "Your Role Name"
}'
```

管理代理程序

您的 IBM Cloud Application Performance Management 安装包含用于管理监视代理程序的工具。

其中一些工具也在受管系统的初始配置期间使用：第 150 页的『使用代理程序命令』、第 155 页的『代理程序配置页面』和第 155 页的『在 Windows 系统上使用 IBM Cloud Application Performance Management 窗口』。

以非 root 用户身份启动代理程序

如果希望以其他用户身份启动代理程序，请在系统上创建公共组，并使每个用户成为该组的成员。

开始之前

如果以相同的非 root 用户身份安装和配置了代理程序，并且想要以相同用户身份启动代理程序，那么无需特殊操作。如果以一个选中的用户身份安装和配置了代理程序，并且想要以其他用户身份启动代理程序，请在系统上创建一个公共组。使所有代理程序管理用户成为此公共组的成员。将所有代理程序文件和目录的所有权移交到该组。

关于此任务

自动启动脚本由安装、升级或配置生成。该脚本（名为 `ITMAgentsN` 或 `rc.itmN`，具体取决于 UNIX 操作系统）可以针对特定安装中的每个应用程序包含一个条目。缺省情况下，将以 root 用户访问权启动所有代理程序。要更新系统启动脚本，并以非 root 用户身份启动代理程序，必须编辑 `install_dir/config/kcirunas.cfg` 文件，该文件包含 XML 语法的超集。缺省情况下，`kcirunas.cfg` 文件中的每个 **productCode** 节都处于禁用状态。通过从 **!productCode** 中移除注释指示符来激活代理程序的 **productCode** 节。将忽略已注释或已取消激活的节。对于未安装的应用程序，也会忽略对应的已取消注释或已激活的节。

过程

1. 在 Linux 或 UNIX 上安装监视代理程序，如第 105 页的『安装代理程序』（在 AIX 系统上）或第 112 页的『安装代理程序』（在 Linux 系统上）所述。
2. 可选： 根据需要在 Linux 或 UNIX 上配置监视代理程序，请参阅第 135 页的『第 7 章 配置环境』。
3. 以非 root 用户的群组名来运行 `./secure.sh` 脚本，以保护文件的安全，并设置文件的文件组所有权。例如： **./secure.sh -g db2iadm1**
4. 要更新系统启动脚本，请完成以下步骤：
   a) 更新 `install_dir/config/kcirunas.cfg` 文件。激活代理程序的 **productCode** 节。对于不需要实例值的代理程序，请指定 product_code、实例和用户，其中 *product_code* 值是第 151 页的表 11 中指定的两字符代码。对于需要实例值的代理程序（例如 Db2 监视代理程序，产品代码为 ud），请指定 product_code、实例、用户和名称。
   例如：

   ```
   <productCode>ud</productCode>
   <instance>
   <name>db2inst1</name>
   <user>db2inst1</user>
   </instance>
   <instance>
   <name>db2inst2</name>
   <user>root</user>
   </instance>
   ```

   b) 以 root 用户身份或 sudo 用户访问权运行以下脚本：*install_dir*`/bin/UpdateAutoRun.sh`

下一步做什么

有关 **./secure.sh** 脚本的更多信息，请参阅保护代理程序安装文件。

将相同的用户标识用于代理程序安装和升级。

事务监视的事件阈值

您可以使用事件阈值立即监视环境。您还可以创建定制的事件阈值以测试特定条件，并在关键绩效指标超过阈值时发出事件。

Response Time Monitoring 事件

在 Web 事务超过**响应时间**阈值时创建“响应时间”事件。

在单击 **系统配置 > 阈值管理器**后，选择**响应时间**作为**数据源类型**。响应时间监视环境的所有事件阈值将应用于相同类型的所有受管系统。

以下预定义阈值可用于响应时间监视代理程序。

表 *243. Response Time Monitoring* 阈值

Threshold	描述	公式
Response_Time_Availability_Crit	高比例的 Web 事务失败。	如果 WRT 事务 Status.Percent_Failed 大于 10 并且 WRT 事务 Status.Transaction_Definition_Name 不等于“Ignore Resources”，那么 Response_Time_Availability_Crit 为 true
Response_Time_Availability_Warn	适中比例的 Web 事务失败。	如果 WRT 事务 Status.Percent_Failed 大于 0、WRT 事务 Status.Percent_Failed 小于 10 并且 WRT 事务 Status.Transaction_Definition_Name 不等于“Ignore Resources”，那么 Response_Time_Availability_Warn 为 true
Response_Time_Critical	响应时间缓慢的 Web 事务的百分比较高。	如果 WRT 事务 Status.Percent_Slow 大于 5、WRT 事务 Status.Percent_Available 等于 100 并且 WRT 事务 Status.Transaction_Definition_Name 不等于“Ignore Resources”，那么 Response_Time_Critical 为 true
Response_Time_Warning	响应时间缓慢的 Web 事务的百分比适中。	如果 WRT 事务 Status.Percent_Slow 大于 1、WRT 事务 Status.Percent_Slow 小于 5、WRT 事务 Status.Percent_Available 等于 100 并且 WRT 事务 Status.Transaction_Definition_Name 不等于“Ignore Resources”，那么 Response_Time_Warning 为 true

良好请求响应时间少于 10 秒。慢请求响应时间多于 10 秒。用于确定良好响应时间与慢响应时间的 10 秒值不可配置。

事务跟踪 事件

在中间件事务超过 事务跟踪 阈值时，创建 事务跟踪 事件

要查看缺省 事务跟踪 阈值，请单击 **系统配置 > 阈值管理器**，然后选择 **事务跟踪** 作为**数据源类型**。

提示: 您可以创建自己的 事务跟踪 阈值（如果需要）。

以下预定义阈值可用于中间件事务。

表 *244.* 事务跟踪 阈值

Threshold	**描述**	**公式**
Interaction_Avail_Critical	高比例的中间件交互失败。	如果 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED 大于 10，那么 Interaction_Avail_Critical 为 true
Interaction_Avail_Warning	中间件交互的适中比例失败。	如果 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED 大于 0 并且 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED 小于或等于 10，那么 Interaction_Avail_Warning 为 true
Interaction_Time_Critical	总时间缓慢的中间件交互的百分比较高。	如果 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW 大于或等于 5 并且 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED 等于 0，那么 Interaction_Time_Critical 为 true
Interaction_Time_Warning	总时间缓慢的中间件交互的百分比适中。	如果 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW 大于 1、KTE INTERACTION AGGREGATE DATA.PERCENTAGE_SLOW 小于 5 并且 KTE INTERACTION AGGREGATE DATA.PERCENTAGE_FAILED 等于 0，那么 Interaction_Time_Warning 为 true
Transaction_Avail_Critical	高比例的中间件事务失败。	如果 KTE 事务 AGGREGATE DATA.PERCENTAGE_FAILED 大于 10，那么 Transaction_Avail_Critical 为 true

表 244. 事务跟踪 阈值 (续)

Threshold	描述	公式
Transaction_Avail_Warning	适中比例的中间件事务失败。	如果 KTE 事务 AGGREGATE DATA.PERCENTAGE_FAILED 大于 0 并且 KTE 事务 AGGREGATE DATA.PERCENTAGE_FAILED 小于或等于 10，那么 Transaction_Avail_Warning 为 true
Transaction_Time_Critical	总时间缓慢的中间件事务的百分比较高。	如果 KTE 事务 AGGREGATE DATA.PERCENTAGE_SLOW 大于或等于 5 并且 KTE 事务 AGGREGATE DATA.PERCENTAGE_FAILED 等于 0，那么 Transaction_Time_Critical 为 true
Transaction_Time_Warning	总时间缓慢的中间件事务的百分比适中。	如果 KTE 事务 AGGREGATE DATA.PERCENTAGE_SLOW 大于 1、KTE 事务 AGGREGATE DATA.PERCENTAGE_SLOW 小于 5 并且 KTE 事务 AGGREGATE DATA.PERCENTAGE_FAILED 等于 0，那么 Transaction_Time_Warning 为 true

良好请求响应时间少于 10 秒。慢请求响应时间多于 10 秒。用于确定良好响应时间与慢响应时间的 10 秒值不可配置。

创建阈值以针对事务监视生成事件

使用“阈值管理器”来为事务创建阈值。阈值用于将属性的采样值与阈值中设置的值进行比较。如果采样值满足比较条件，那么将生成事务事件。

关于此任务

您可以使用阈值来监视应用程序何时报告特定条件。有关事务监视的缺省阈值的更多信息，请参阅第 860 页的『事务监视的事件阈值』。

您可以创建其他阈值以监视事务的其他方面。例如，您可以创建一个阈值来监视中间件事务事件速率下降。然后，如果事务事件速率下降地域阈值指定的值，那么将生成一个事件。

过程

要创建阈值并将其与一个或多个事务相关联，请完成以下任务：

1. 在“导航栏”上，单击 **系统配置 > 阈值管理器**。将**数据源类型**设置为**事务跟踪**。
2. 单击 “添加”以创建新阈值。
3. 设置超过此阈值的事件的严重性。
4. 要将阈值与事务相关联，请设置以下值：
   - 数据集 - KTE TRANSACTION AGGREGATE DATA
   - 显示项 - Resource_Value
   - 逻辑运算符 - And (&)
5. 或者，要将阈值与交互关联，请设置以下值：
   - 数据集 - KTE INTERACTION AGGREGATE DATA
   - 显示项 - Source_Resource_Value

· 逻辑运算符 - And （&）

6. 单击 **添加**以添加条件。在**添加条件**框中，选择一个属性和一个运算符，然后输入阈值。

   例如，要添加每分钟的事务数量低于 100 时生成事务事件的阈值条件，请设置以下值，然后单击**确定**：

   · 属性 - Transaction_Rate

   · 运算符 - Less than

   · 值 - 100

   重复此步骤以向阈值添加更多条件（如果需要）。
7. 在“组分配”部分中，选择事务跟踪以将阈值分配到此资源组。
8. 单击**保存**。

结果

您已创建阈值并将其与事务或交互关联。在满足阈值条件时，将生成一个事件。您可以在“应用程序性能仪表板”的“事件”选项卡中监视事件。

示例

除缺省值外，要针对 Response Time Monitoring 代理程序创建阈值以监视 Web 事务的其他方面：

1. 在“阈值管理器”中，将**数据源类型**设置为**响应时间**。
2. 在添加阈值时，请使用以下设置：

   · 数据集 - WRT Transaction Status

   · 显示项 - Application

   · 逻辑运算符 - And （&）

   · 组分配 - Web Response Time

管理操作系统代理程序事件

您可以配置操作系统代理程序以管理事件。

事件过滤和摘要

使用在配置 (.conf) 文件中设置的事件过滤和摘要选项以控制操作系统代理程序如何处理重复事件。

在监视日志时，事件可能快速显示多次。例如，在生成日志的应用程序遇到错误时可能发生此重复的日志记录，并且将持续记录此错误直至已解析阈值。在发生此类型的日志记录时，会将过多数量的事件发送到 Performance Management 基础结构。大量事件会对性能造成负面影响。

注: 仅在发送到 Performance Management 的事件上支持事件检测和摘要过程。您无法在 EIF 发送到 OMNIbus 的事件上完成这些过程。

检测并过滤重复事件

您可以配置操作系统代理程序以处理重复事件。

要缓解多个重复事件的问题，在 .conf 文件中使用 DupDetectionKeyAttributes 标记来定义构成重复事件的内容。在逗号分隔列表中，包含一个或多个定义的 Performance Management 属性，这些属性用于确定是否将事件视为重复事件。在以下示例中，具有相同消息和相同 CustomSlot1 的事件将被视为重复事件：

```
DupDetectionKeyAttributes=msg,CustomSlot1
```

从 Performance Management 属性检测重复事件。因此，如果想要基于定义的特定槽检测重复，请完成以下步骤：

1. 将槽值映射到 Performance Management 属性。
2. 将 Performance Management 属性映射到 .conf 文件中的 DupDetectionKeyAttributes 标记。

使用以下示例，其中，将重要槽 eventclass 和 eventid 映射到 *CustomSlot1* 和 *CustomSlot2*：

```
REGEX BaseAuditEvent
^([A-Z][a-z]{2} [0-9]{1,2} [0-9]{1,2}:[0-9]{2}:[0-9]{2}
 [0-9] {4}) [0-9] (\S+) (\S+) \
Microsoft-Windows-Security-Auditing (\S+) ([0-9]+) (.*)
timestamp $1
severity $2
eventclass $3 CustomSlot1
eventkeywords $4
eventid $5 CustomSlot2
msg $6
END
```

如果想要指定特定事件指定为重复事件，请在 .conf 文件中，将 Performance Management 属性映射到 DupDetectionKeyAttributes 标记，如下所示：

```
DupDetectionKeyAttributes=CustomSlot1,CustomSlot2
```

注:

1. CustomSlot 属性名称区分大小写，因此您必须输入与先前示例中完全相同的名称。
2. 如果未提供属性列表，那么值缺省为 Class 和 Logname。

这些属性匹配的事件将被代理程序视为重复事件。

因为重复检测为全局，最好的做法是选取一组定制槽以用作键并在所有格式语句中以此方式使用它们。例如，将槽 1 - 3 用于键。如果格式仅需要一个键但还需要更多槽，那么使用槽 1 以包含名称值，并使用槽 4 到 n 以包含其他数据。

摘要时间间隔

重复检测过程在称为“摘要时间间隔”的时间段内运行。

在此时间间隔内统计重复事件，然后在时间间隔到期时重置。计数器在每个新的摘要时间间隔开始时重新从 0 开始启动计数。

代理程序针对时间间隔内监视的每个事件集发送一个摘要事件。摘要事件包含匹配的第一个事件的属性值。摘要事件还包含一个计数，指示在摘要时间间隔内此事件发生的重复数。

在配置 (.conf) 文件中设置摘要时间间隔，如以下示例中所示：

```
EventSummaryInterval=300
```

分配给摘要时间间隔的值以秒为单位，因此在此示例中，摘要时间间隔为 5 分钟。

过滤事件

如果事件过滤正在运行，那么（.conf）文件中的 EventFloodThreshold 设置将通知代理程序何时发送事件。

下表显示 EventFloodThreshold 值。

表 *245. EventFloodThreshold* 值

EventFloodThreshold 值	**描述**
send_all	*send_all* 值是缺省值。将发送所有事件，即使这些事件是重复事件。
send_none	*send_none* 值意味着不发送任何单个事件。仅发送摘要事件。
send_first	使用 *send_first* 值以在遇到第一个事件时立即发送事件。如果在指定的时间内第一个事件重复出现，那么不会发送第一个事件的后续重复。有关更多信息，请参阅第 864 页的『摘要时间间隔』。

表 245. *EventFloodThreshold* 值 (续)	
EventFloodThreshold 值	**描述**
n 整数	使用 *n* 整数值以仅在特定时间内发送事件的第 *n* 次出现（例如，每隔五个重复）。有关更多信息，请参阅第 864 页的『摘要时间间隔』。

摘要属性
“事件类型”和“出现次数”属性用于帮助汇总事件。

在启用事件摘要时，“事件类型”和“出现次数”属性变得有意义。“事件类型”属性指示事件的类型，*Event* 或 *Summary Event*。与日志中找到的记录相对应的常规事件将一对一标记为 *Event*。在“摘要时间间隔”结束时发送的摘要事件将标记为 *Summary Event*。

“出现次数”属性指示在日志中针对此事件找到的重复记录的总量。摘要事件包含此计数，因为其显示在前一个摘要时间间隔内收到的匹配摘要事件的事件的数量。

阈值和摘要事件

无论第 864 页的『过滤事件』中描述的过滤器值，总是在每个摘要时间间隔结束时收到在此时间间隔期间至少发生一次的任何事件的摘要事件。如果不期望摘要事件，那么可能意外触发阈值。要避免意外触发阈值，请在阈值中包含子句 *Event Type== Event* 或 *Event Type!= Summary Event*。

Windows 事件日志
操作系统代理程序使用 .conf 文件以监视来自 Windows 事件日志的事件。

操作系统代理程序继续使用 WINEVENTLOGS 配置（.conf）文件选项来监视来自 Windows 事件日志的事件。代理程序监视逗号分隔的事件日志列表，如以下示例中所示：

```
WINEVENTLOGS=System,Security,Application
```

操作系统代理程序还继续使用 WINEVENTLOGS=All 设置。All 设置指示以下标准事件日志：安全性、应用程序、系统、目录、域名系统 (DNS) 和文件复制服务 (FRS)（Windows 2008 之前的版本随附）。但是，不会检查系统上的所有事件日志。

UseNewEventLogAPI 配置文件标记允许事件日志（Windows 事件日志 2008 或更高版本）访问 Microsoft 添加的任何新日志以及其他应用程序或用户创建的任何 Windows 事件日志。新日志通过 WINEVENTLOGS 关键字列出。

在以下示例中，UseNewEventLogAPI 标记设置为 y。

```
UseNewEventLogAPI=y
WINEVENTLOGS=Microsoft-Windows-Hyper-V-Worker-Admin
```

在此示例中，在具有 Hyper-V 角色的 Windows 系统上监视 Microsoft-Windows-Hyper-V/Admin。

在 Windows 事件日志中，每个事件按此顺序具有以下字段：

- 以下格式的日志：月、天、时间和年
- 整数形式的事件类别
- 事件级别
- Windows 安全标识。如果配置（.conf）文件中存在 SpaceReplacement=TRUE，那么 Windows 安全标识中的任何空格都将替换为下划线。

  注: 如果（.conf）文件中将 UseNewEventLogAPI 设置为 y（指定为使用事件日志），那么 SpaceReplacement=TRUE 为缺省值。
- Windows 源。如果配置（.conf）文件中存在 SpaceReplacement=TRUE，那么 Windows 源中的任何空格都将替换为下划线。
- Windows 事件日志密码。如果配置（.conf）文件中存在 SpaceReplacement=TRUE，那么 Windows 事件日志关键字中的任何空格都将替换为下划线。

注: 此处描述的关键字字段是 Windows 2008 版本的事件日志中的新增项。其不存在于先前的事件日志中，因此其存在会阻止您直接复用旧的“事件日志”格式语句。必须针对此额外的字段进行修改。

· Windows 事件标识

· 消息文本

例如，在管理用户登录到 Windows 2008 系统时，将在“安全性”日志中生成一个事件，指示分配给新用户会话的特权:

```
Mar 22 13:58:35 2011 1 Information N/A Microsoft-Windows-
Security-Auditing Audit_Success 4672 Special privileges assigned to new logon.
S-1-5-21-586564200-1406810015-1408784414-500    Account Name:
Administrator   Account Domain:     MOLDOVA     Logon ID:
0xc39cb8e    Privileges:        SeSecurityPrivilege
SeBackupPrivilege          SeRestorePrivilege
SeTakeOwnershipPrivilege            SeDebugPrivilege
SeSystemEnvironmentPrivilege            SeLoadDriverPrivilege
SeImpersonatePrivilege
```

要捕获 `Microsoft-Windows-Security-Auditing` 事件源创建的所有事件，请如下所示编写格式语句:

```
REGEX BaseAuditEvent
^([A-Z][a-z]{2} [0-9]{1,2} [0-9]{1,2}:[0-9]{2}:[0-9]{2} [0-9]
{4}) [0-9] (\S+) (\S+) Microsoft-Windows-Security-Auditing (\S+)
([0-9]+) (.*)
timestamp $1
severity $2
login $3
eventsource "Microsoft-Windows-Security-Auditing"
eventkeywords $4
eventid $5
msg $6
END
```

对于先前的示例事件，以下示例指示分配给槽的值:

```
timestamp=Mar 22 13:58:35 2011
severity=Information
login=N/A
eventsource=Microsoft-Windows-Security-Auditing
eventid=4672
msg="Special privileges assigned to new logon.
S-1-5-21-586564200-1406810015-1408784414-500    Account Name:
Administrator   Account Domain:     MOLDOVA     Logon ID:
0xc39cb8e    Privileges:        SeSecurityPrivilege
SeBackupPrivilege          SeRestorePrivilege
SeTakeOwnershipPrivilege            SeDebugPrivilege
SeSystemEnvironmentPrivilege            SeLoadDriverPrivilege
SeImpersonatePrivilege
```

因为很难精确预测这些事件的情况，编写正则表达式的有用方法是在文件中捕获实际事件。然后，您可以检查文件，选择想要代理程序捕获的事件，然后编写正则表达式以匹配这些事件。要从 Windows 事件日志捕获所有事件，请使用以下步骤:

1. 创建仅包含不匹配任何项的一个模式的格式文件，如以下示例中所示:

   ```
   REGEX NoMatch
   This doesn't match anything
   END
   ```

2. 将以下设置添加到配置（`.conf`）文件:

   ```
   UnmatchLog=C:/temp/evlog.unmatch
   ```

3. 运行代理程序并捕获某些样本事件。

事件映射

Tivoli Event Integration Facility (EIF) 接口用于将状态事件转发到 Tivoli Netcool/OMNIbus、Tivoli Enterprise Console 或 Operations Analytics - Log Analysis。

EIF 事件指定事件类，并将事件数据指定为标识事件槽的名称和值的“名称/值”对。事件类可以有子类。Performance Management 提供所有监视事件都包含的基本事件类定义和一组基本槽。代理程序会扩展基本事件类，以定义包含特定于代理程序槽的子类。对于 OS 代理程序日志文件事件，事件类与代理程序属性组对应，特定于代理程序的槽与属性组中的属性对应。

对于在 LFAProfiles 属性组中通过阈值生成的事件，将使用 ITM_KLO_LFAPROFILES 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· subnode_msn: STRING

· subnode_affinity: STRING

· subnode_type: STRING

· subnode_resource_name: STRING

· subnode_version: STRING

· subnode_config_file: STRING

· subnode_description: STRING

· subnode_description_enum: STRING

对于在 Log File RegEx Statistics 属性组中通过阈值生成的事件，将使用 ITM_KLO_LOG_FILE_REGEX_STATISTICS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· table_name: STRING

· attrib_name: STRING

· filter_number: INTEGER

· average_processor_time: REAL

· average_processor_time_enum: STRING

· total_processor_time: REAL

· total_processor_time_enum: STRING

· max_processor_time: REAL

· max_processor_time_enum: STRING

· min_processor_time: REAL

· min_processor_time_enum: STRING

· filter_count: REAL

· filter_count_matched: REAL

· filter_count_unmatched: REAL

· regex_pattern: STRING

· last_matched_time: STRING

· last_matched_time_enum: STRING

· last_unmatched_time: STRING

· last_unmatched_time_enum: STRING

· result_type: INTEGER

· result_type_enum: STRING

对于在 Log File Status 属性组中通过阈值生成的事件，将使用 ITM_KLO_LOG_FILE_STATUS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· table_name: STRING

· file_name: STRING

· regex_pattern: STRING

· file_type: INTEGER

· file_type_enum: STRING

· file_status: INTEGER

· file_status_enum: STRING

· num_records_matched: INTEGER

· num_records_not_matched: INTEGER

· num_records_not_matched_enum: STRING

· num_records_processed: INTEGER

· current_file_position: REAL

· current_file_position_enum: STRING

· current_file_size: REAL

· current_file_size_enum: STRING

· last_modification_time: STRING

· last_modification_time_enum: STRING

· codepage: STRING

对于在 LogfileEvents 属性组中通过阈值生成的事件，将使用 ITM_KLO_LOGFILEEVENTS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· klo_class: STRING

· logname: STRING

· eifevent: STRING

· klo_msg: STRING

· customslot1: STRING

· customslot2: STRING

· customslot3: STRING

· customslot4: STRING

· customslot5: STRING

· customslot6: STRING

· customslot7: STRING

· customslot8: STRING

· customslot9: STRING

· customslot10: STRING

· occurrence_count: INTEGER

· occurrence_count_enum: STRING

· event_type: INTEGER

· event_type_enum: STRING

· custominteger1: REAL

· custominteger1_enum: STRING

· custominteger2: REAL

· custominteger2_enum: STRING

· custominteger3: REAL

· custominteger3_enum: STRING

· remotehost: STRING

对于在 LogfileProfileEvents 属性组中通过阈值生成的事件，将使用 ITM_KLO_LOGFILEPROFILEEVENTS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· klo_class: STRING

· logname: STRING

· eifevent: STRING

· klo_msg: STRING

· customslot1: STRING

· customslot2: STRING

· customslot3: STRING

· customslot4: STRING

· customslot5: STRING

· customslot6: STRING

· customslot7: STRING

· customslot8: STRING

· customslot9: STRING

· customslot10: STRING

· occurrence_count: INTEGER

· occurrence_count_enum: STRING

· event_type: INTEGER

· event_type_enum: STRING

· custominteger1: REAL

· custominteger1_enum: STRING

· custominteger2: REAL

· custominteger2_enum: STRING

· custominteger3: REAL

· custominteger3_enum: STRING

· remotehost: STRING

对于在 Performance Object Status 属性组中通过阈值生成的事件，将使用 ITM_KLO_PERFORMANCE_OBJECT_STATUS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· query_name: STRING

· object_name: STRING

· object_type: INTEGER

· object_type_enum: STRING

· object_status: INTEGER

· object_status_enum: STRING

· error_code: INTEGER

· error_code_enum: STRING

· last_collection_start: STRING

· last_collection_start_enum: STRING

· last_collection_finished: STRING

· last_collection_finished_enum: STRING

· last_collection_duration: REAL

· average_collection_duration: REAL

· average_collection_duration_enum: STRING

· refresh_interval: INTEGER

· number_of_collections: INTEGER

· cache_hits: INTEGER

· cache_misses: INTEGER

· cache_hit_percent: REAL

· intervals_skipped: INTEGER

对于在 pro Performance Object Status 属性组中通过阈值生成的事件，将使用 ITM_KLO_PRO_PERFORMANCE_OBJECT_STATUS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· query_name: STRING

· object_name: STRING

· object_type: INTEGER

· object_type_enum: STRING

· object_status: INTEGER

· object_status_enum: STRING

· error_code: INTEGER

· error_code_enum: STRING

· last_collection_start: STRING

· last_collection_start_enum: STRING

· last_collection_finished: STRING

· last_collection_finished_enum: STRING

· last_collection_duration: REAL

· average_collection_duration: REAL

· average_collection_duration_enum: STRING

· refresh_interval: INTEGER

· number_of_collections: INTEGER

· cache_hits: INTEGER

· cache_misses: INTEGER

· cache_hit_percent: REAL

· intervals_skipped: INTEGER

对于在 Thread Pool Status 属性组中通过阈值生成的事件，将使用 ITM_KLO_THREAD_POOL_STATUS 事件类进行发送。此事件类包含以下槽：

· node: STRING

· timestamp: STRING

· thread_pool_size: INTEGER

· thread_pool_size_enum: STRING

· thread_pool_max_size: INTEGER

· thread_pool_max_size_enum: STRING

· thread_pool_active_threads: INTEGER

· thread_pool_active_threads_enum: STRING

· thread_pool_avg_active_threads: REAL

· thread_pool_avg_active_threads_enum: STRING

· thread_pool_min_active_threads: INTEGER

· thread_pool_min_active_threads_enum: STRING

· thread_pool_max_active_threads: INTEGER

· thread_pool_max_active_threads_enum: STRING

· thread_pool_queue_length: INTEGER

· thread_pool_queue_length_enum: STRING

· thread_pool_avg_queue_length: REAL

· thread_pool_avg_queue_length_enum: STRING

· thread_pool_min_queue_length: INTEGER

· thread_pool_min_queue_length_enum: STRING

· thread_pool_max_queue_length: INTEGER

· thread_pool_max_queue_length_enum: STRING

· thread_pool_avg_job_wait: REAL

· thread_pool_avg_job_wait_enum: STRING

· thread_pool_total_jobs: INTEGER

· thread_pool_total_jobs_enum: STRING

使用 Web 站点监视管理合成事务和事件

创建合成事务以监视内部应用程序、外部应用程序以及位于不同位置的面向公众的应用程序的性能和可用性。

在 Synthetic Script Manager 中创建合成事务。在 Synthetic Script Manager 中生成简单脚本以测试应用程序可用性，或使用 Selenium IDE 来记录用于对应用程序复制不同用户操作的合成脚本。然后，配置合成事务以按照特定时间间隔和回放位置回放脚本。

要点: 只有 IBM Website Monitoring on Cloud 附加组件的现有用户才能使用 Synthetic Playback 代理程序和 Synthetic Script Manager。Website Monitoring 已被 IBM Cloud Availability Monitoring 2017 年 8 月发行版替换。有关更多信息，请参阅第 887 页的『关于 Availability Monitoring』。

您的可用回放位置即安装了 Monitoring Agent for Synthetic Playback 以及为监视面向公众的 Web 应用程序提供的 15 个存在点 (PoPs) 的位置。 PoP 可用于下列位置：

· 阿姆斯特丹

· 金奈

· 达拉斯

· 法兰克福

· 香港

· 伦敦

· 墨尔本

· 墨西哥

· 巴黎

· 圣约瑟

· 圣保罗

· 新加坡

· 东京

· 多伦多

· 华盛顿

创建阈值和资源组以在应用程序变慢或不可用时发起事件和通知项目干系人。在应用程序性能仪表板中查看性能数据和生成历史报告。

如果您使用 Response Time 代理程序监视某个应用程序的最终用户响应时间，那么可以在应用程序性能仪表板中同时查看最终用户和合成事务的 KPI。将合成事务作为组件添加至您要使用 Response Time 代理程序监视的应用程序。

注: 要在 Synthetics Script Manager 中工作，您必须是对 Synthetic Script Manager 和代理程序配置具有查看许可权的角色的成员。有关更多信息，请参阅第 851 页的『角色和许可权』。

记录合成脚本

使用 Firefox Web 浏览器以及 Selenium IDE 附加组件来记录合成脚本。借助 Selenium IDE，您可以记录 Web 页面上的用户操作，例如，装入页面、单击链接或者选择对象。在 Selenium IDE 进行记录时，其在脚本中为每个用户操作生成一条命令。然后，您可以使用 Synthetic Script Manager 来配置脚本，从而按照设置的时间间隔在不同位置模拟用户在 Web 站点上的行为。

开始之前

在记录脚本时，必须使用 Firefox Web 浏览器

Selenium IDE 只能用作 Firefox 附加组件。如果未安装 Selenium IDE 或者未在运行，请完成以下步骤：

1. 请确保您正在运行支持 Selenium IDE 3.2.X 或 3.3.X 的 Firefox 60 版本或更高版本。如果您具有更高版本的 Selenium IDE，那么它不受支持；必须将其卸载并安装 V3.2.X 或 3.3.X。

   注: 缺省情况下，在安装 V3.2.X 或 3.3.X 后，Selenium IDE 会自动进行更新。关闭 Selenium ID 的自动更新以防止版本升级。
2. 从 **Selenium** 主页 (https://addons.mozilla.org/firefox/addon/selenium-ide/versions/) 下载并安装 Selenium IDE 3.2.X 或 3.3.X。允许 Selenium IDE 安装所有插件。
3. 安装 Selenium IDE 后，重新启动 Firefox。
4. 浏览至想要测试的 Web 页面，并关闭任何其他选项卡。要打开 Selenium IDE，单击**工具** > **Selenium IDE**。在 **Selenium IDE** 窗口中，确保**基本 URL** 字段包含显示的 Web 页面的 URL。Selenium IDE 在显示的 Web 页面上开始记录所有用户操作。

Selenium .side 脚本格式

使用较新版本的 Selenium 创建的脚本使用 `..side` 格式。使用 Selenium IDE 3.2.X 或 3.3.X，您可以导入使用 `.html` 格式创建的较旧脚本并保存为 `.side` 格式。有关更多信息，请参阅第 875 页的『从先前的 Selenium IDE 版本更新脚本』。

如果将使用 Selenium `.side` 脚本，那么首先必须安装以下更新：

- 安装了 Synthetic Playback 代理程序的系统上的 IBM Cloud Application Performance Management V8.1.4.0 Synthetic Playback 代理程序 临时修订 5 或更高版本。
- 联系 IBM 以确保您的 Cloud APM 预订已更新为 IBM IBM Cloud Application Performance Management, Private Cloud APM V8.1.4.0 Server 临时修订 8 或更高版本。
- 如果使用 Availability Monitoring 专用存在点 (PoP)，请通过从 PoP 安装目录中输入 **`cat build.info`** 命令，检查合成 PoP 构建号是否为 `APM_201903090832` 或更高版本。先前的构建版本不支持 `.side` 格式。

可从 IBM Support > Fix Central > IBM APM 8.1.4.0 下载临时修订或 Cloud APM V8.1.4.0。

关于此任务

在此任务中，您在 Web 页面上执行用户操作，并使用 Selenium IDE 在简单脚本中将这些操作录制为命令。可以使用脚本在应用程序性能仪表板中监视 Web 应用程序的性能和可用性。

过程

完成以下步骤以记录 Web 页面上用户操作的脚本：

1. 单击**录制**以开始录制脚本。在 Web 页面上执行用户操作，例如，单击链接。

   对于 Web 页面上的每个用户操作，Selenium IDE 记录一条命令并将其添加到脚本。

   例如，在脚本中完成下列操作，以在用户装入 IBM Marketplace Web 页面并浏览到 Cloud APM 的免费试用时进行记录：

表 *246.* 记录用户操作和 *Selenium IDE* 命令

用户操作	添加到脚本的命令
要添加有关 IBM Marketplace Web 站点上 Cloud APM Web 页面打开的记录，打开 IBM Marketplace Web 页面。右键单击已显示 Web 页面上的任何位置并选择**打开**。	`open`
为了确保脚本检查 Web 页面加载，请右键单击该 Web 页面的标题文本 (IBM Performance Management)，然后单击**显示所有可用命令** > **verifyTitle IBM Cloud Application Performance Management**。	`verifyTitle`

表 *246.* 记录用户操作和 *Selenium IDE* 命令 *(续)*	
用户操作	**添加到脚本的命令**
要添加有关用户单击某个链接以查看 Cloud APM 相关详细信息的记录，单击**详细信息**链接。这将装入“**详细信息**”页面。	clickAndWait
为确保脚本检查“**详细信息**”页面是否已经装入，右键单击“功能部件亮点”“”标题，并选择**显示所有可用命令** > **verifyText css=h2.heading--TERTIARY**。	verifyText
要添加有关用户单击链接以查看如何购买 Cloud APM 的详细信息的记录，请单击**购买**链接。这将装入“**购买**”页面。	clickandWait
要在用户单击按钮以注册 Cloud APM 的免费试用时进行记录，请单击**免费试用**按钮。	click

2. 在 Selenium IDE 窗口中，单击**录制**以停止录制。单击**保存项目**工具，为脚本提供有意义的名称，然后另存为 .side 文件（例如，open_webpage.side）。
3. 在 Selenium IDE 窗口中，复审您录制的脚本。单击**表**选项卡可以按表格式显示该脚本。在 Selenium IDE 窗口中，单击**播放当前测试用例**以测试录制的脚本的回放。

   在此示例中，Selenium IDE 在 IBM Marketplace Web 站点上显示用户操作的脚本，如步骤 1 中所述。

表 *247. Selenium IDE* 脚本记录 *IBM Marketplace Web* 站点上的用户操作的示例		
命令	**目标**	**值**
open	/	
verifyTitle	IBM Cloud Application Performance Management	
clickAndWait	css=ul > #details > a	
verifyText	css=h2.heading--TERTIARY	功能部件亮点
clickAndWait	css=ul > #purchase > a	
click	link=Try Free	

结果

您已录制可用于监视 Web 应用程序的性能和可用性的脚本。

下一步做什么

如果录制了复杂脚本，那么您可以将脚本组织到更简单的脚本中，其中每个脚本都表示 Web 应用程序中的特定业务流程或用户操作。

使用 Synthetic Script Manager 将脚本文件上载到新的或现有合成事务。

构造复杂的脚本

将复杂脚本组织到多个脚本中；然后，将脚本一起保存到一个名为测试套件 *(test suite)* 的集合脚本中。

关于此任务

如果创建负载脚本，那么可以将脚本组织到表示 Web 应用程序中不同业务或用户流程的简单脚本中。将这些脚本一起保存为测试套件。然后，您可以在应用程序性能仪表板中使用这些脚本监视 Web 应用程序在响应特定用户操作方面的性能和可用性。

应该只有一个测试套件，并且应该将所有测试添加到该测试套件中。

要点: 最佳实践是将复杂脚本组织到单独的脚本中，其中每个脚本都表示您要监视的一个典型用户或业务流程。例如，创建用于记录用户何时登录 Web 站点或搜索某个项的单独脚本。如果您根据用户或业务流程组织脚本，那么可以在应用程序性能仪表板中监视 Web 应用程序对这些特定流程的响应。

过程

要将复杂脚本组织到单独的脚本中，并将脚本保存为测试套件，请完成下列步骤：

1. 要针对在脚本中记录的每个用户流程创建一个单独的脚本，请在 Selenium IDE 中单击**测试** > **+**。为每个脚本指定一个描述用户流程的有意义名称，然后将每个脚本保存为 `.side` 文件，例如，`load_homepage.side`。

   有关更多信息，请参阅第 872 页的『记录合成脚本』。

   要点: 在 Selenium IDE 中针对脚本指定的名称是标识在应用程序性能仪表板中监视的录制的业务或用户流程的名称。
2. 在 Selenium IDE，打开您先前记录的复杂脚本。根据不同的用户操作，将脚本命令组织到单独的脚本中。在“**测试用例**”窗口中，从原始复杂脚本中**剪切**命令，然后将它们**粘贴**到不同的“**测试用例**”窗口中。

   例如，记录合成脚本中的复杂脚本示例包含用于三种不同用户流程的 Selenium IDE 命令。

   · 打开 IBM Marketplace Web 站点上的 Cloud APM 主页。

   · 在 IBM Marketplace 上打开“**详细信息**”页面。

   · 打开“**定价**”页面，并在用户打开免费试用注册页面时进行记录。

   然后，用户操作将组织到三个不同的脚本中。

表 *248.* 用于打开 *IBM Marketplace* 页面 *(load_homepage.side)* 的样本脚本

命令	目标	值
open	/	
verifyTitle	IBM Cloud Application Performance Management	

表 *249.* 用于在 *IBM Marketplace (load_products.side)* 上打开“**详细信息**”页面的样本脚本

命令	目标	值
clickAndWait	css=ul > #details > a	
verifyText	css=h2.heading--TERTIARY	功能部件亮点

表 *250.* 用于打开 *IBM Marketplace* 上的“**购买**”和试用注册页面的样本脚本 *(load_APM.side)*

命令	目标	值
clickAndWait	css=ul > #purchase > a	
click	link=Try Free	

3. 要将个别测试用例放入测试套件中，请切换到“**测试套件**”窗口，并根据业务逻辑顺序将测试添加到测试套件。最后，单击**保存项目**工具以将测试套件和测试套件中的所有测试保存到一个 `.side` 文件。例如，考虑逻辑顺序 `Load_URL`、`Select Manage inventory` 和 `Select IBM Machine Type`。在将这些测试用例添加到测试套件时，首先检查 `Load_URL`，然后是 `Select Manage inventory`，最后是 `Select IBM Machine Type`

结果

您已记录一组可用于监视 Web 应用程序的性能和可用性的脚本。使用 Synthetic Script Manager 将脚本的 `.side` 测试套件添加到新的或现有综合事务。

从先前的 Selenium IDE 版本更新脚本

受支持的 Selenium IDE V2.2.X 和 3.2.X 使用 `.side` 格式来记录合成脚本，而不是较旧版本的 Selenium IDE 使用的 `.html` 格式。如果具有现有 `.html` 脚本，那么仍可以使用这些脚本。使用较旧版本的 Selenium IDE 创建的脚本可能无法完全用于 IBM Cloud Availability Monitoring 使用的最新 Firefox 和 Selenium 驱动

程序。在某些实例中，您可能想要编辑 .html 脚本，以新的 .side 格式重新记录它们，或者导入 .html 脚本并保存为 .side 格式。

过程

- 例外：如果想要与 Select2 元素交互，请勿使用 **select** 命令（请参阅 https://github.com/SeleniumHQ/selenium-ide）。
  旧脚本为：

```
<td>select</td>
<td>id=country</td>
<td>label=United States</td>
```

  其应该更改为：

```
<tr>
   <td>runScript</td>
   <td>window.scrollTo(0,810)</td>
   <td></td>
</tr>
<tr>
   <td>click</td>
   <td>id=select2-country-container</td>
   <td></td>
</tr>
<tr>
   <td>click</td>
   <td>xpath=(//ul[@id='select2-country-results']/li[text() = 'United States'])</td>
   <td></td>
</tr>
```

- 限制：支持使用 Selenium IDE 3.2.X 或 3.3.X 记录的 .side 脚本；不支持 **linkText** 定位符。

管理合成事务
使用 Synthetic Script Manager 创建、配置和删除合成事务。

要显示 Synthetic Script Manager，请单击**系统配置**图标，然后选择 **Synthetic Script Manager**。要在 Synthetics Script Manager 中工作，您必须是对 **Synthetic Script Manger** 和**代理程序配置**具有查看许可权的角色的成员。有关更多信息，请参阅第 851 页的『角色和许可权』。

您可以查看面向外部的公共 Web 应用程序当月的合成事务使用情况的相关数据。在 Synthetic Script Manager 中的“每月回放用量”表中，查看本月已执行的回放实例数和基于当前配置的预测回放实例数。

要点: 要为内部专用 Web 应用程序和外部专用 Web 应用程序创建合成事务，必须在包含待监视 Web 应用程序的每个位置安装 Synthetic Playback 代理程序。

您可以使用 Synthetic Script Manager 执行以下任务：

· 创建和编辑合成事务。

· 配置合成事务变量。

· 删除合成事务。

创建和编辑合成事务
要查看有关 Web 应用程序的性能和可用性的数据，首先必须在 Synthetic Script Manager 中创建合成事务。

关于此任务
使用 Synthetic Script Manager 创建、编辑和配置合成事务。在 Synthetic Script Editor 中输入 Web 应用程序的 URL 来为合成事务生成简单脚本。要模拟复杂用户流程，在 Synthetic Script Editor 中将合成脚本上载到合成事务。然后，将合成事务配置为定期运行以及在不同位置运行。

过程

要创建事务或者编辑现有事务，请完成以下步骤：

1. 可选： 如果未显示 Synthetic Script Manager，请单击**系统配置** 图标，并选择 **Synthetic Script Manager**。

2. 要创建新事务，请单击**新建**图标 。要编辑现有事务，请单击**编辑**图标 。
3. 在 **Synthetic Script Editor** 中，单击**上载脚本**选项卡，然后在**事务名称**文本框中输入事务名称。在**描述**文本框中输入事务的描述。

   要点: 请勿为事务指定与过去 24 小时删除的某个事务相同的名称。如果为事务指定与最近删除的事务相同的名称，那么在应用程序性能仪表板中，来自这两个事务的数据都会错误地归属于该事务名称。
4. 要生成简单脚本以测试 Web 应用程序，请选择**输入 Web 页面的 URL 以进行测试**并输入 URL。Synthetic Script Manager 基于该 URL 生成简单合成脚本。
5. 要将先前创建的脚本文件分配给您的事务，选择**上载脚本文件**。单击**上载脚本**以浏览系统上的脚本。选择脚本，然后单击**打开**。

   要点: 合成脚本文件必须为以下文件类型之一:

   - `.html`
   - `.zip`

   将单个简单脚本（测试用例）保存为 `.html` 文件。将测试用例和测试套件一起压缩到 `.zip` 文件中。
6. 要配置同时或错开的合成事务回放，单击**调度脚本**选项卡。选择**同时**以同时从所有位置执行事务，或选择**错开**以在每个时间间隔从一个不同的位置执行事务。
7. 要选择脚本运行频率，请单击**调度脚本**选项卡。单击**时间间隔**文本框并根据想要监视 Web 应用程序的频率输入数字。选择 `1` 和 `60` 分钟之间的时间间隔长度。

   注: 大型或复杂脚本的运行时间可能较长。对于大型或复杂脚本，选择较长的时间间隔。
8. 要选择脚本的回放位置，请单击**调度脚本**选项卡，然后选择要在其中运行脚本的数据中心位置和代理程序安装位置。
9. 要设置合成事务和子事务的响应时间阈值，请单击**高级设置**选项卡；然后单击并展开合成事务以显示所有子事务。双击响应时间阈值并输入值。选择 `0` 和 `3600` 秒之间的值。如果不想设置阈值，请输入 `0`。缺省响应时间阈值为 `10` 秒。

   注: 某些命令可能需要比其他命令更长的时间。选择适合想要测试的命令的响应时间阈值。如果事务将测试 Web 页面打开所用的时间，请选择较长的响应时间。
10. 要完成创建或编辑事务，请单击**保存**。

结果
您已配置合成事务。合成事务将在 Synthetic Script Manager 中列出。

下一步做什么
您可以在应用程序性能仪表板中查看合成事务记录的度量值和 KPI。还可以将事务作为组件添加到应用程序，并查看与该应用程序关联的所有合成事务。

要点: 首次添加合成事务时，应用程序性能仪表板中的“**随时间变化的可用性**”组窗口小部件中可能会显示空白。当服务器接收到事务的第一个回放结果时，该空白迅速消失。

配置合成事务变量
使用 Synthetic Script Manager 可更新变量值，例如，存储在合成脚本中的用户名和密码，而无需编辑脚本文件。变量值对于每个回放位置都是唯一的。Web 应用程序在不同位置需要不同的变量值时，请为合成脚本配置变量。例如，如果您的 Web 应用程序不允许在不同位置使用相同的登录详细信息，请使用 Synthetic Script Manager 在每个位置提供不同的登录详细信息。通过在 Selenium-IDE 插件中使用 `store` 命令，您可以在合成脚本中创建多个变量。

关于此任务

在此任务中，使用 Synthetic Script Manager 配置存储在合成脚本中的变量。

过程

要配置合成事务的变量，请完成下列步骤:

1. 如果未显示 Synthetic Script Manager，请单击**系统配置** 图标，并选择 **Synthetic Script Manager**。从列表中选择合成事务，然后单击**编辑**图标。
2. 选择您的合成事务的回放位置。
3. 单击**高级设置**选项卡。如果合成脚本包含变量，那么您可以在“**配置不同位置的变量替换**”窗口中编辑这些变量。要编辑变量，请双击该值。要完成，请单击**保存**。

   例如，以下脚本包含变量 *username* 和 *password*。变量 `user1` 和 `pass` 的值使用 Selenium-IDE 中的 `store` 命令进行保存。这些变量在两个位置 Dallas 和 San Jose 上具有相同值。

表 *251.* 包含变量的脚本的示例

命令	**目标**	**值**
`store`	`user1`	*username*
`store`	`pass`	密码
`type`	`id=j_username`	*${username}*
`type`	`id=j_password`	*${password}*

   脚本变量的值显示在“**配置不同位置的变量替换**”窗口中。将位置 `Dallas` 的 *username* 的值从 `user1` 更改为 `admin1`，以便合成事务在不同的位置使用不同的登录详细信息。

表 *252.* 位于不同位置的脚本变量值

位置	**username**	**密码**
San Jose	user1	pass
Dallas	admin1	pass

结果
您已配置合成事务的变量。您现在可以使用该合成事务在不同位置测试 Web 应用程序的性能和可用性。

下一步做什么
您可以在应用程序性能仪表板中查看合成事务记录的度量值和 KPI。还可以将事务作为组件添加到应用程序，并查看与该应用程序关联的所有合成事务。

为合成事务过滤 *URL* 和域名
使用 Synthetic Script Manager 可以向用于测试的白名单和黑名单添加规则，以允许或阻止访问特定的 URL 和域。

关于此任务

您可控制将对所测试 Web 应用程序的响应时间产生影响的依赖项和资源。使用黑名单可过滤掉来自所指定域的请求，以从所测量的响应时间中移除这些请求。使用白名单可包括来自所指定域的请求，以将这些请求添加到所测量的响应时间。使用黑名单和白名单可过滤掉或包括与 Web 应用程序相关联的依赖项，例如第三方度量。

黑名单字段包含用于阻止访问所指定 URL 和域的一列规则。

白名单字段包含用于允许访问所指定 URL 和域的一列规则。

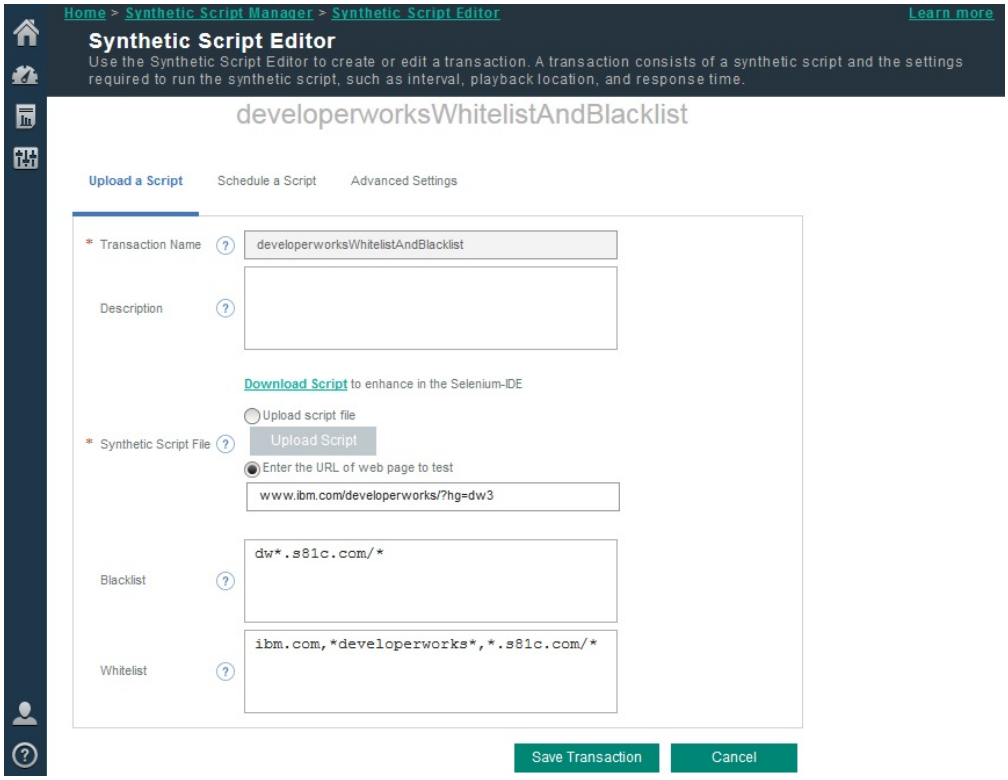
请使用逗号 (,) 来分隔黑名单和白名单中的规则。使用通配符 (*) 来过滤域名和 URL。例如，`ibm.com, *.bluemix.net, *developerworks*, *.profile.*.cloundfront.net/*.png`。

注: 如果为合成事务同时配置黑名单和白名单，那么黑名单的优先级较高。

过程

要为合成事务添加规则到黑名单或白名单，请完成下列步骤：

1. 如果未显示 Synthetic Script Manager，请单击**系统配置** 图标，并选择 **Synthetic Script Manager**。
2. 要为现有的事务来配置黑名单或白名单，请单击**编辑**图标。如果未列出任何事务，请单击**新建**图标以新建事务。
3. 在 **Synthetic Script Editor** 中，单击**上载脚本**选项卡，然后在**事务名称**文本框中输入事务名称。在**黑名单**和**白名单**文本框中添加规则。



结果
现在，您的合成事务将为 Web 应用程序过滤掉不需要的请求，并允许访问其他特定请求。

在 *Synthetic Script Manager* 中隐藏密码
在合成脚本中将密码存储为变量，以在 Synthetic Script Manager 中隐藏密码值。

开始之前
此过程要求您编辑合成脚本。使用 Selenium IDE 来记录合成脚本。有关更多信息，请参阅第 872 页的『记录合成脚本』。

关于此任务

在 Selenium IDE 中手动修改合成脚本以将密码存储为变量。然后，您可以使用 Synthetic Script Manager 中的隐藏密码来创建合成事务。在 Synthetic Script Manager 中隐藏的密码显示为星号。

要点: 建议将密码存储在合成脚本中，以使密码值不显示在 Synthetic Script Manager 中。通过阻止他人查看密码，隐藏的密码使您的 Web 应用程序更安全。

过程

1. 打开要在 Selenium IDE 中修改的脚本。使用 `store` 命令将密码分配到变量 *password*，遵循此步骤中所描述的示例；然后，保存脚本。

   要点: 您必须将密码存储为变量名 *password*，以使密码不显示在 Synthetic Script Manager 中。

   例如，以下合成脚本包含用户名 *test@example.com* 和密码值 *ibm4value*。

```
<tr>
   <td>type</td>
```

```
   <td>id=username</td>
   <td>test@example.com</td>
</tr>
<tr>
   <td>type</td>
   <td>id=password</td>
   <td>ibm4value</td>
</tr>
```

以下脚本显示如何将密码值 *ibm4value* 分配到变量 *password*，方法是使用 `store` 密码。

```
<tr>
   <td>store</td>
   <td>ibm4value</td>
   <td>password</td>
</tr>
<tr>
   <td>type</td>
   <td>id=username</td>
   <td>test@example.com</td>
</tr>
<tr>
   <td>type</td>
   <td>id=password</td>
   <td>$(password)</td>
</tr>
```

2. 可选： 要在脚本级别隐藏密码，请使用 `store` 命令将空白值分配到变量 *password*；然后，保存脚本。
   稍后可以在 Synthetic Script Manager 中设置密码。

   例如，以下脚本显示如何使用 `store` 密码将空白值分配到变量 *password*。

```
<tr>
   <td>store</td>
   <td></td>
   <td>password</td>
</tr>
<tr>
   <td>type</td>
   <td>id=username</td>
   <td>test@example.com</td>
</tr>
<tr>
   <td>type</td>
   <td>id=password</td>
   <td>$(password)</td>
</tr>
```

3. 登录 Cloud APM 控制台，然后打开 **Synthetic Script Manager**。创建事务并将脚本上载到该事务。单击**高级设置**选项卡。
   隐藏每个位置的密码。您可以为每个位置更改密码。有关更多信息，请参阅第 876 页的『管理合成事务』。

删除合成事务

使用 Synthetic Script Manager 以删除合成事务。

过程

要删除合成事务，请完成以下步骤：

1. 如果合成事务已分配给某个应用程序，那么首先必须先从该应用程序移除事务。在应用程序性能仪表板上，单击并展开**我的所有应用程序**，然后单击与想要删除的合成事务关联的应用程序。单击**编辑**图标。在“**编辑应用程序**”窗口中，从应用程序中移除合成事务组件。有关更多信息，请参阅管理应用程序。
   现在可删除合成事务。
2. 在导航栏上，单击**系统配置**图标，然后选择 **Synthetic Script Manager**。选择合成事务，然后单击**删除**图标。要确认您想要删除此合成事务，请单击**确定**。

结果
这将删除合成事务。

在应用程序性能仪表板中查看合成事务数据
在应用程序性能仪表板中查看合成事务数据。将合成事务与新的或现有应用程序关联，并在应用程序性能仪表板中统一查看所有关联的合成事务。

关于此任务

您可以在应用程序性能仪表板中的“**我的事务**”窗口中查看合成事务数据。

您还可以通过将事务与应用程序关联来创建合成事务组。使用应用程序性能仪表板中的“**添加应用程序**”或“**编辑应用程序**”工具将合成事务作为组件添加到新的或现有 Web 应用程序。然后，您可以在应用程序性能仪表板中统一查看与该应用程序关联的所有合成事务的数据。

如果已在使用 Response Time 代理程序来监视应用程序的用户响应时间，那么可以向此应用程序添加合成事务。然后，可以在应用程序性能仪表板只能怪查看该应用程序的更多度量和 KPI。

过程

- 要查看合成事务，请完成以下步骤：
  a) 单击**性能**图标，然后选择**应用程序性能仪表板**。在“**应用程序**”窗口中，展开**我的所有应用程序**，然后选择**我的事务**。在“**组**”窗口中，展开**事务**并选择**合成事务**。
  b) 单击合成事务以查看该事务的可用性和性能数据，以及定义的时间段内事务实例的响应时间图形。
- 要将合成事务与应用程序关联，请完成以下步骤：
  a) 单击**性能**图标，然后选择**应用程序性能仪表板**。选择并编辑现有应用程序或者创建新应用程序。有关更多信息，请参阅管理应用程序。
  b) 在“**添加应用程序窗口**”窗口中，单击**添加组件**图标 +，然后从组件列表中选择**合成事务**。在“**组件编辑器**”窗口中，选择一个合成事务，然后单击**添加**以将合成事务与应用程序相关联。
  c) 单击**后退**。单击**关闭**以关闭“**组件编辑器**”窗口。单击**保存**。要将其他合成事务添加为组件，请重复步骤 1 - 3。

结果
将合成事务与应用程序相关联。现在，您可以在应用程序性能仪表板中查看应用程序及其关联的合成事务。有关更多信息，请参阅管理应用程序。

注: 将合成事务与应用程序进行关联时，该应用程序的初始可用性未知。可能需要几分钟来更新状态。

管理合成事件
使用“阈值管理器”和“资源组管理器”来配置阈值并将它们分配给合成事务。事务属性值与阈值中定义的条件匹配时，将生成合成事件。您可以在“应用程序性能仪表板”中监视合成事件。

为合成事务创建阈值
使用“阈值管理器”来为合成事务创建阈值。阈值用于将属性值与阈值中设置的值进行比较。如果采样值满足比较条件，那么将生成事件。

关于此任务

阈值允许用户监视应用程序报告的特定情况。例如，您可以创建阈值以监视 Web 站点响应特定用户命令所用的时间。如果 Web 站点所用的时间超过阈值中指定的时间，那么将生成合成事件。

过程

要创建阈值并将其与一个或多个合成事务相关联，请完成以下步骤：

1. 在导航栏上，单击**系统配置**图标，然后选择**阈值管理器**。将**数据源**类型设置为**合成事务**。
2. 创建阈值。有关更多信息，请参阅第 838 页的『阈值管理器』。

3. 要将阈值与事务相关联，请选择 **KSO TRANSACTION** 作为**数据集**，然后选择 **TRANSNAME** 作为**显示项**。对于**逻辑运算符**，选择**和 (&)**。

   注: 必须选择 **TRANSNAME** 作为**显示项**。如果未选择 **TRANSNAME**，那么无法在“应用程序性能仪表板”上查看合成事件。
4. 要添加条件，请单击**新建条件**图标 ⊕。在“新建条件”框中，选择**属性**和**运算符**。然后，对于**值**输入阈值。要将此条件添加到阈值，请单击**确定**。

   例如，要添加在超过 50% 事务变慢时生成合成事件的阈值条件，请选择 **PSLOW** 作为**属性**，然后选择**大于**作为**运算符**。要设置用于生成事件的慢事务所占的百分比，请输入 50 作为**值**。
5. 要定义更多阈值属性，请向阈值添加更多条件。
6. 在完成时，单击**保存**。如果不想将阈值分配给资源组，请单击**确定**。

结果
您已创建阈值并将其与合成事务相关联。在满足阈值条件时，将生成一个事件。您可以在应用程序性能仪表板中的“**事件**”选项卡中监视事件。

下一步做什么
您可以将合成事务分组到各个资源组中。

为合成事务创建资源组
将合成事务组织到资源组中，并将阈值应用于该资源组中的所有事务。

开始之前
创建要应用于资源组中所有合成事务的阈值。

关于此任务

您可以将合成事务组织到资源组中，并将阈值应用于该资源组中的所有合成事务。使用资源组管理器以创建资源组，并为该资源组指定阈值。然后，将一个或多个合成事务子节点指定到该资源组。与该资源组关联的阈值现在可以应用于所有关联的合成事务。

过程

要为合成事务创建资源组，请完成以下步骤：

1. 单击**系统配置**图标 🎚，然后选择**资源组管理器**。创建资源组，或编辑现有资源组。有关更多信息，请参阅第 833 页的『资源组管理器』。

要为合成事务创建资源组，请完成以下步骤：

2. 为资源组指定名称和描述。在“**阈值分配**”表中，为此资源组指定阈值，然后单击**保存**。在“资源组管理器”中，再次选择此资源组，然后单击**编辑**图标 ✎。
3. 从“**资源分配**”表将资源组与合成事务子节点关联，然后单击**保存**。

   合成事务子节点的格式为 `SO:`*`TransactionName`*。例如，如果您有一个事务 `open_webpage`，那么可用子节点名为 `SO:open_webpage`。

结果
您已将合成事务组织到资源组中，并将阈值应用于该资源组中的所有事务。

针对同时和错开的合成事务创建临界阈值
使用“阈值管理器”针对同时和错开的合成事务创建临界阈值。

关于此任务

创建阈值以在错开的事务连续失败或者同时事务在所有回放位置处失败时通知项目干系人。有关更多信息，请参阅第 876 页的『创建和编辑合成事务』。

过程

要创建临界阈值以在错开的事务回放实例失败时创建事件，请完成以下步骤：

1. 在“阈值管理器”中为合成事务创建阈值。有关更多信息，请参阅第 881 页的『为合成事务创建阈值』。
2. 在“阈值管理器”中，选择**临界**作为**严重性**，并输入 1 分钟作为阈值**时间间隔 (HHMMSS)**。使用以下公式来确定**所需连续样本数量**：

   所需连续样本数量 = (回放时间间隔 * 预期连续失败次数) - 1

   例如，如果想要监视的合成事务的回放时间间隔为 5 分钟，且您希望检测到 8 次连续回放失败，那么必须将**所需连续样本数量**设置为 (5 * 8) -1 = 39。
3. 添加条件。在“新建条件”框中，选择 **LOCATION** 作为**属性**，选择**等于**作为**运算符**，然后输入 None 作为**值**。添加第二个条件，并将 **PFAILED** = 100。保存阈值。
4. 在导航栏中，打开**资源组管理器**。创建资源组。将一个或多个错开的合成事务分配到资源组，然后将您在步骤 1-3 中创建的阈值分配到资源组。保存资源组。有关更多信息，请参阅第 882 页的『为合成事务创建资源组』。

要创建临界阈值以在同时事务回放实例在若干位置失败时创建事件，请完成以下步骤：

5. 在“阈值管理器”中为合成事务创建阈值。有关更多信息，请参阅第 881 页的『为合成事务创建阈值』。
6. 在“阈值管理器”中，选择**临界**作为**严重性**，并输入 1 分钟作为**时间间隔 (HHMMSS)**。将**所需连续样本数量**设置为与要监视事务的回放时间间隔相同的值。

   例如，如果要监视的合成事务的回放时间间隔为 5 分钟，那么请将**所需连续样本数量**设置为 5。
7. 添加条件。在“新建条件”框中，选择 **LOCATION** 作为**属性**，选择**等于**作为**运算符**，然后输入 None 作为**值**。添加第二个条件，并将 **PFAILED** = 100。保存阈值。
8. 创建资源组。将一个或多个合成事务以及上述新增临界阈值添加到资源组。有关更多信息，请参阅第 882 页的『为合成事务创建资源组』。

结果

您已为错开的或同时合成事务创建了临界阈值。在满足阈值条件时，将生成一个事件。您可以在应用程序性能仪表板中的“**事件**”选项卡中监视事件。

管理合成事件的电子邮件通知

可以使用“资源组管理器”和 IBM Alert Notification 在应用程序性能超过阈值时生成电子邮件通知。

开始之前

要配置合成事件的电子邮件通知，您必须先为预订启用 Alert Notification。有关更多信息，请参阅 IBMKnowledge Center 中的 Alert Notification。

关于此任务

在“资源组管理器”中，使用 Alert Notification 配置电子邮件通知。在应用程序性能满足与资源组关联的阈值所设置的条件时将生成电子邮件通知。

过程

要管理电子邮件通知，请完成下列步骤：

1. 单击**系统配置**图标，然后选择**资源组管理器**。创建或编辑资源组。
2. 在“**阈值分配**”表中选择阈值并单击**保存**。然后，在“**资源分配**”表中，将资源组 Synthetic Playback 资源或 Synthetic Events 代理程序资源关联起来。单击**保存**。在“资源组管理器”中，再次选择此资源组，然后单击**编辑**图标。
3. 要在新标签页中启动 Alert Notification，请在“资源组管理器”中单击**配置电子邮件通知**。在 Alert Notification 中，单击导航栏中的**用户**以创建或编辑电子邮件通知的接收方。有关更多信息，请参阅 IBMKnowledge Center 中的 Alert Notification。

4. 在 Alert Notification 中，单击导航栏中的**通知策略**。系统将自动为“**通知编辑器**”供应新策略。该策略的名称和过滤器派生自资源组。单击**添加规则**以定义用于确定何时发送电子邮件通知的条件。有关更多信息，请参阅 IBMKnowledge Center 中的 Alert Notification。
5. 要完成电子邮件通知配置，请单击**保存**。您的策略将列示在**通知策略**选项卡中的表内。

最大化代理程序和服务器性能以进行日志文件监视的相关准则

为确保 OS 代理程序和 Performance Management 服务器达到最大性能，必须以格式 (.fmt) 文件形式定义正则表达式，还必须限制报告给 Cloud APM 控制台的日志文件监视事件数量。

以 .fmt 文件形式定义正则表达式的相关准则

.fmt 文件使用的正则表达式需要大量 CPU 处理。为提高代理程序和服务器性能，请使用以下准则，尽量缩短根据 .fmt 文件中的正则表达式检查监视源日志中的记录所花费的时间：

尽量减少多行模式的使用。

多行模式开销较大，因为代理程序必须确定有哪些记录，然后确定记录是否匹配。当您使用的多行模式为包含“\n”字符的正则表达式，或者为包含“%n”标记的 TEC 样式格式时，代理程序必须先对受监视文件进行断行处理，以形成不同大小的多条记录。然后，代理程序必须根据格式文件中的表达式检查记录。该过程要求检查两次正则表达式，因此处理较慢。如果您使用单行模式，那么会假设文件的每一行为一条记录，因此处理更快。

在某些情况下，可能会忽略某些行来达到更好的性能。例如，以下显示了 RAS1 跟踪日志中的单条记录：

```
(4D66DACB.0001-1:RAS1,400,"CTBLD")
```

```
+4D66DACB.0001 Component: ira
```

```
+4D66DACB.0001 Driver: agent_fac:15/4114877.7
```

```
+4D66DACB.0001 Timestamp: Feb 24 2011 13:18:54
```

```
+4D66DACB.0001 Target: sos510amdx6-d
```

在该示例中，如果您只想处理以下行：

```
+4D66DACB.0001 Driver: agent_fac:15/4114877.7
```

那么您可以编写以下单行模式：

```
^\+.*Driver: agent_fac:([0-9\.\/]+)$
```

该单行模式处理所需的重要值，而无需使用多行格式。其他四行逻辑记录被视为不匹配任何项的单行记录，并将被废弃。

按照监视日志中的出现频率对格式文件中的表达式排序。

代理程序根据格式文件中的表达式检查其从日志读取的每条记录，直到找到匹配项为止。先使用文件中的最后一个表达式，然后向上搜索。当找到匹配项时，将停止搜索。如果最常记录的表达式列在最后，那么当记录该表达式时，这将是唯一检查的表达式。

如果格式文件中有 100 个表达式，那么每当日志记录与格式文件中所列的第一个表达式匹配时，代理程序都必须先检查其他 99 个表达式，这样会降低处理速度。当从日志中读取的记录与格式文件中的任何模式都不匹配时，代理程序必须根据所有模式进行检查，然后才能知道该记录不匹配。该过程速度慢，开销大。

在正则表达式中包含尽可能多的常量数据。

例如，如果日志中返回了以下错误：

```
Disk error on device: /dev/sda1 Disk error on device: /dev/sdb2 yyy
```

您可以编写以下表达式：

```
^Disk.*: .*$
```

该表达式会产生匹配项，但会强制正则表达式引擎考虑其他相似行的更多可能性，而这些行最终并不匹配，例如，缺少冒号。

以下表达式更实用，因为该表达式更精确，会导致正则表达式引擎停止处理不匹配的错误：

```
^Disk error on device: /dev/sd[a-b][0-9]$
```

请勿使用您不需要的子表达式。

在以下示例中，在括号中显示的子表达式用于向正则表达式引擎通知您想要使用匹配数据中返回的值。这些子表达式导致额外处理，如果您不使用返回的值，则不需要使用这些子表达式。例如，如果日志中返回了以下错误：

```
write failure in writing to client 9.27.135.191. Error Broken pipe
```

如果在格式文件中包含以下正则表达式，会在结尾处捕获错误消息；但如果不使用返回的值，性能会受到负面影响：

```
REGEX
WriteFailure

^write failure in writing to client (.*)\. 错误
(.*)$

ClientAddr $1
CustomSlot1

END
```

在表达式中使用括号以用于分组目的。

您可以使用 **?** 运算符来通知正则表达式引擎不要捕获返回的值。因此，可以使用 **?** 运算符仅将返回的值分组到一起。该分组对性能有正面影响。例如，如果返回了以下日志数据：

```
Login succeeded on the first attempt for user Bob.
Login succeeded on the third attempt for user Joe.
```

要与返回的这两个值均匹配，必须考虑第一次或第三次登录尝试。如果并不介意哪次特定的登录尝试成功或哪个特定的用户成功，可以包含该表达式将返回的值分组到一起：

```
REGEX
LoginSuceeded

^login succeeded on the (?:[a-z]+) attempt for user ([A-Z][a-z]*)\.$

UserName $1
CustomSlot1
END
```

如果可能，请不要在表达式中使用 OR (|) 运算符。

| 运算符的处理开销较大。**|** 运算符会使正则表达式引擎完成备份，并尝试匹配最初不匹配的值。该过程的效率远远低于使用两个单独的表达式。例如，如果您有以下表达式：

```
REGEX DiskError
^.*disk error.*4|^.*disk failure.*4
END
```

使用以下两个表达式的效率远远高于上述示例：

```
REGEX DiskError
^.*disk error.*4
END

REGEX DiskError
^.*disk failure.*4
END
```

这些表达式返回相同的结果。

要点: 这些表达式违反了使用尽可能多的常量数据的准则，仅演示了使用 | 运算符的问题。

请勿使用模糊表达式。
模糊表达式强制正则表达式引擎备份和查找不同的表达式匹配方法。有关更多信息，请参阅性能提示。

如果包含用于将长记录切断成许多子表达式的表达式，可能会产生模糊表达式。在此问题的退化版本中，表达式在以下两个项之间具有空格（.*):

```
(.*) (.*)
```

在该退化版本示例中，正则表达式引擎查找以空格分隔的任意表达式的两个字符串。但是，* 还与空格匹配，因此正则表达式引擎可能会将出现的第一个空格分配给第一个 (.*)。如果到达输入记录结尾但没有找到其他空格，必须使用该空格作为表达式中需要的实际空格再次进行备份和尝试。

为提高性能，请仅使用具体表达式。您可以使用 Regex Pal 工具来检查您定义的格式文件是否与监视日志匹配。有关更多信息，请参阅 Regex Pal。

限制所报告的日志文件事件数量的相关准则。

以下准则限制可能导致 OS 代理程序或 Cloud APM 服务器性能变差的日志文件事件数量:

以 .fmt 文件形式编写特定格式。
以 .fmt 文件形式编写返回相关记录的特定格式。例如，您可以为特定错误生成事件，例如以 Error: 开头的行，并忽略以 Warning: 开头的行

```
Error:  disk failure
Error: out of memory
WARNING: incorrect login
```

请勿打开 .conf 文件中的 Unmatchlog 设置。
请确保不要打开 .conf 文件中的 **Unmatchlog** 设置，因为该设置将记录所有不匹配的文件，从而使文件系统超负载。

在 .fmt 文件中指定 *DISCARD* 事件类。
尝试在 .fmt 中指定预定义 *DISCARD* 事件类来故意废弃数据，从而限制代理程序的 CPU 用量。在使用 *DISCARD* 事件类时，不会为与 .fmt 文件中的模式匹配的日志记录创建事件。例如:

```
REGEX *DISCARD*
```

打开较长时间的重复事件检测。
您可以在 .conf 文件中使用以下键来打开重复事件检测:

- DupDetectionKeyAttributes
- EventSummaryInterval
- EventFloodThreshold

在该示例中，通过 *msg* 和 *CustomSlot1* 值识别重复行:

```
DupDetectionKeyAttributes=msg,CustomSlot1
EventSummaryInterval=300
EventFloodThreshold=send_first
```

如果您有许多重复事件，请将 *send_first* 或 *send_none* 阈值应用于事件。有关更多信息，请参阅第 863 页的『检测并过滤重复事件』。

编写特定阈值条件。
编写用于限制与阈值匹配的行集的特定阈值条件。例如，以下阈值公式导致阈值仅在 FileSystemUsage 事件类的事件在 CustomInteger1 中具有大于或等于 95 的值时才触发阈值:

```
( Class == ’ FileSystemUsage’  AND CustomInteger1 >= 95)
```

为代理程序提供正确的 .conf 和 .fmt 文件集。

确保为代理程序提供正确的 .conf 和 .fmt 文件集。例如，如果您在为 Windows 操作系统代理程序配置日志文件监视，请确保配置专门为 Windows 操作系统代理程序创建的 .conf 和 .fmt 文件。

查询 MongoDB 警报数据库以确定打开的事件数或事件速率。

· 完成以下步骤以查询 MongoDB 警报数据库，从而确定打开的事件数或事件速率：

1. 使用针对警报数据库的 MongoDB 查询创建 event-query.js 文件，例如：
   - 该查询计算具有以下阈值名称的所有打开和关闭事件的数量：

     ```
     UDB_DB_Pool_Hit_Rat_Pct_Crit_2 db.alarms.count
     ```

     ```
     ({"threshold_name" : "UDB_DB_Pool_Hit_Rat_Pct_Crit_2"})
     ```

   - 该查询计算 MongoDB 中打开和关闭的事件数量：

     ```
     db.alarms.count()
     ```

2. 运行此命令以获取 event-query.js 文件中的查询的结果：**/opt/ibm/mongodb/bin/mongo 127.0.0.1:27000/alarm -u user -p mongoUsrpasswd@08 <event-count.js**。

· 限制为日志监视指定的 CPU 量。有关更多信息，请参阅第 544 页的『日志文件监视环境变量』。

Availability Monitoring

使用 IBM Cloud Availability Monitoring，您可以在 Web 应用程序上创建、编辑、查看和删除模拟最终用户行为的合成测试。

Availability Monitoring 仪表板显示受监视的应用程序、URL 和 REST API 的可用性和响应时间信息。通过使用图形、明细表和映射视图，使用仪表板来监视与不同位置中的应用程序、URL 或 REST API 关联的警报和活动。

Availability Monitoring 可用于包含 Availability Monitoring 附加组件的 Cloud 产品上的 IBM Cloud Application Performance Management, Advanced 用户。

关于 Availability Monitoring

使用 Availability Monitoring 可创建昼夜不停地从不同公用和专用位置监视 Web 应用程序的可用性和性能的合成测试。

使用 Availability Monitoring 创建测试。配置您的测试以按照定义的时间间隔在所选位置运行。下载您自己的定制存在点 (PoP) 并在本地或专用服务器上对其进行部署。在以下位置从 15 个公用 PoP 运行测试：

亚洲
金奈、香港、新加坡和东京

澳洲
墨尔本

欧洲
阿姆斯特丹、法兰克福、伦敦和巴黎。

中美洲
墨西哥

北美洲
达拉斯、圣何塞、多伦多和华盛顿。

南美洲
圣保罗

当您已创建和配置您的测试时，您可以在 Availability Monitoring 仪表板上查看您的应用程序的可用性和性能数据。

Availability Monitoring 包含以下主要功能：

在 5 分钟内开始
在数分钟内轻松地创建单个操作测试以监视您的 Web 应用程序的性能和可用性。

最大程度地增加正常运行时间和提高用户满意度
从多个地理位置频繁监视应用程序的正常运行时间和响应时间。运行合成测试以度量 Web 站点装入和 API 调用的性能。从不同位置监视用于模拟用户流程的 Selenium 脚本。

主动
在问题影响用户之前，接收警告您注意问题的通知。您可以使用集成警报通知服务来创建减少警报噪声的警报策略。

准确快速地确定失败原因
瀑布分析可帮助您确定发生故障时的准确步骤以及失败原因；例如，链接中断、图像太大、查找速度慢或外部请求。屏幕捕获将自动创建以帮助您诊断浏览器故障和历史性能问题。针对您的测试，下载每月、每周和每日的平均可用性和响应时间报告。

要在 Availability Monitoring 中工作，您必须是对要监视的应用程序具有查看许可权的角色的成员。有关更多信息，请参阅第 851 页的『角色和许可权』。

访问 Availability Monitoring

应用程序的 Availability Monitoring **状态概述**选项卡显示有关测试的可用性和状态的摘要信息。. 您可以从 Availability Monitoring **状态概述**选项卡访问 Availability Monitoring 仪表板。

关于此任务

通过在 应用程序性能仪表板 上的“**所有我的应用程序**”窗格中单击合适的应用程序，访问 Availability Monitoring 摘要页面。从该摘要页面中，您可以为应用程序添加测试、查看应用程序的现有测试以及查看 Availability Monitoring 仪表板。

过程

要访问 Availability Monitoring，请完成以下步骤：

1. 单击**性能**图标 ；然后，单击**应用程序性能仪表板**。
2. 在“**我的所有应用程序**”窗格中，单击要监视的应用程序；然后单击**组**窗格中的**可用性监视**。
   如果未列出任何应用程序，您必须创建一个应用程序。确保您选择了**定制应用程序**作为您的**模板**。有关更多信息，请参阅第 930 页的『管理应用程序』。

   Availability Monitoring **状态概述**选项卡显示三个标尺，分别表示过去 24 小时的平均测试可用性、所有测试的当前测试状态以及当前计划的分配服务使用情况。

   通过选择要包含在**可用性计算**中的测试，您可以配置 Availability Monitoring 如何计算“平均测试可用性”。单击“平均测试可用性”标尺上的箭头图标 以查看您的测试卡；然后，单击**可用性计算**。单击一个卡以在计算中添加或移除该卡。已排除的测试卡将灰显。当您完成时，单击**我已完成**。“平均测试可用性”标尺将刷新。

   通过在“当前测试状态”标尺上单击箭头图标 ，您可以查看所有您的测试的状态。用于测试的测试卡将显示。单击某个卡以访问该特定测试的**明细**仪表板。
3. 单击**查看监视详细信息**以访问 Availability Monitoring 仪表板并查看您的应用程序的所有测试的数据。单击**查看所有测试**以在**合成测试**窗格中查看和编辑您的测试。单击**添加新测试**以创建测试。

   注: 首次运行 Availability Monitoring 时，您必须添加测试，才能查看 Availability Monitoring 仪表板中的任何数据。

创建和配置测试

创建和配置测试，以报告 Web 应用程序的可用性和性能。

创建和配置测试，可从多个地理位置频繁监视应用程序的可用性和响应时间。运行合成测试，可度量 Web 站点装入和 REST API 调用的性能。创建脚本化行为测试，可从不同位置运行并监视用于模拟用户流程的 Selenium 脚本。

创建 REST API 测试

通过使用以下 HTTP 方法创建 REST API 测试以测试 Web 应用程序的响应时间和可用性：GET、POST、PUT 和 DELETE。

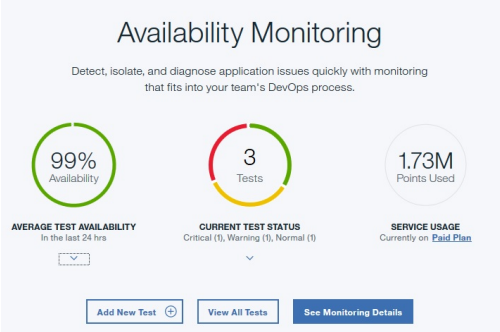
关于此任务

使用 REST API 测试来监视 Web 应用程序和其他 URL 的可用性和性能以对 REST 调用做出响应。

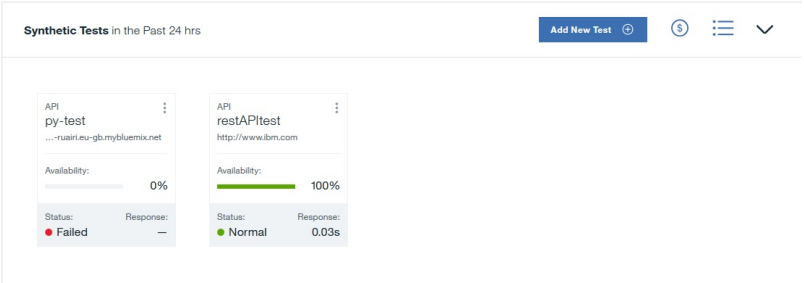
过程

要创建 REST API 测试，请完成以下步骤。

1. 如果要查看应用程序的 Availability Monitoring 摘要页面，请单击**添加新测试**。



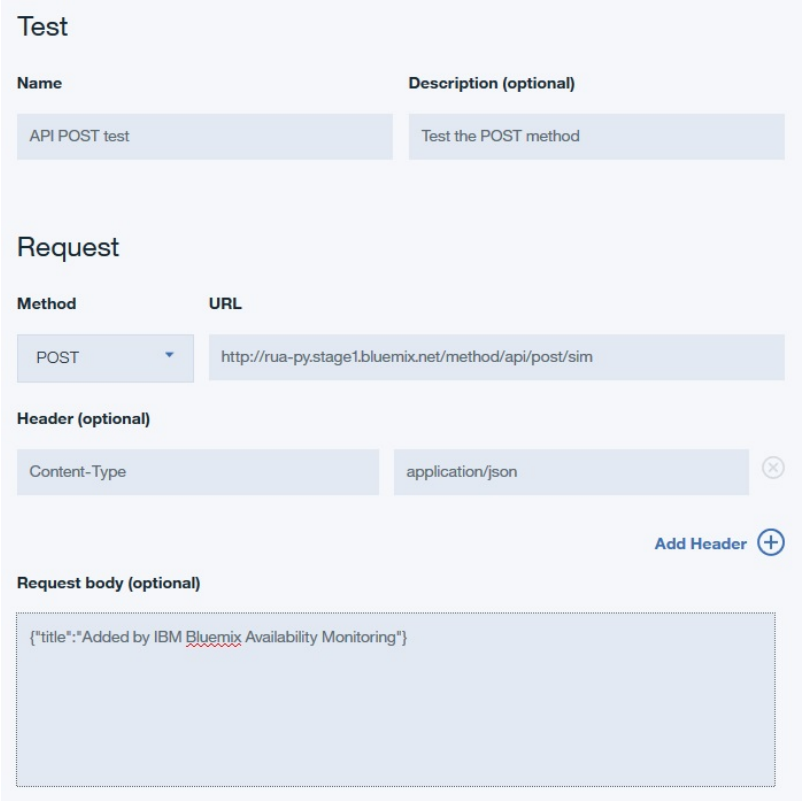
   如果要查看 Availability Monitoring 仪表板，请单击**合成测试**窗格上的**添加新测试**。



2. 单击“**监视设置**”页面上的**单个操作**；然后，单击“**单个操作**”页面上的 **REST API**。
3. 在**名称**字段中输入对您的测试有意义的名称。将测试目的描述添加到**描述**字段。
4. 在“**请求**”部分中，从**方法**列表中选择方法类型并输入要使用此方法测试的 **URL**。

   您可以选择 **GET**、**PUT**、**POST** 或 **DELETE**。如果选择 **PUT** 或 **POST** 方法，您可以在**请求主体（可选）**字段中输入要测试的主体内容。

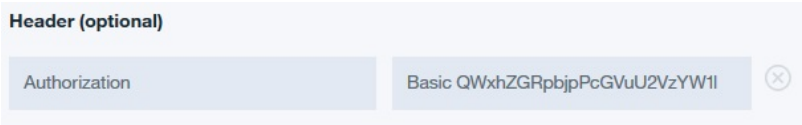
   例如，除了测试您的 Web 应用程序的可用性和性能，以下 REST API 测试使用 POST 方法来请求 Web 应用程序接受数据。



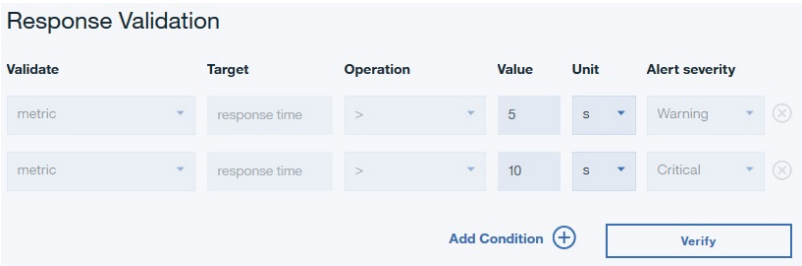
5. 可选： 配置您的测试以包含特定的头和值。在**头**字段中输入头名称和头值。

   如果要测试的 Web 应用程序需要用户登录名和密码，请将“`Authorization`”输入到**头名称**字段中。将单词“基本”、空格字符以及 *username:password* 的 base64 编码值输入到**头值**字段中。

   例如，如果您的用户名为 *Aladdin* 且密码为 *OpenSesame*，那么将单词“基本”、空格字符以及 *Aladdin:OpenSesame* 的 base64 编码值输入到**头值**字段中。



6. 在“**响应验证**”部分中为您的测试配置警告警报阈值和紧急警报阈值。为每行编辑**值**和**单位**。
   超出警告阈值和紧急阈值的响应时间触发警报。



7. 可选： 单击**添加条件**以定义并添加定制响应验证条件。
   定制响应验证条件被整体评估以生成警报。您最多可以定义并添加六个定制条件以用于测试。

   要点:

   在 Availability Monitoring 中，每个测试总共可生成最多三个警报。您的测试将报告严重性为最高的警报，直到所有造成警报的状况都解决为止。有关更多信息，请参阅第 898 页的『Availability Monitoring 中的警报生成』。

   您可以验证以下数据：

头响应代码
选择**头响应代码**以对一个或一系列 HTTP 响应代码进行测试。

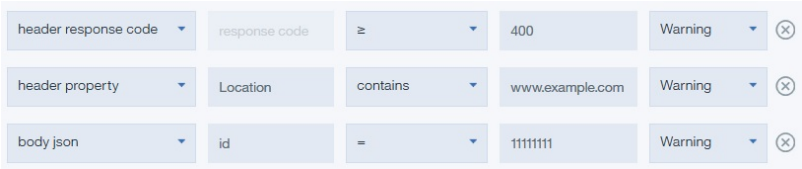
头属性
选择**头属性**以对特定 HTTP 头字段属性和值进行测试。

主体 JSON
选择**主体 JSON** 以对 JSON 主体中的特定属性进行测试。

对于每个条件，请在**目标**字段中输入要测试的属性，在**值**字段中输入要测试的值。从**操作**下拉菜单中选择一个操作员。最后，为您的条件选择**警报严重性**“警告”或“紧急”。

要点:

缺省情况下，您在**值**字段中输入的数字值将被视为数字，而不是字符串。要为响应验证条件输入**值**，请使用引号 “” 来区分字符串和数字。例如，要对字符串 123 进行测试，请在**值**字段中输入“123”。要检查数字 400，请输入不带任何引号的 400。



8. 单击**验证**以创建 REST API 测试并确定您的测试请求是否有效。

   Availability Monitoring 通过使用选定的 HTTP 方法以及为测试定义的任何请求头来确定测试有效性。在测试验证期间未发生响应验证。

   已验证的测试显示在“**已验证的项**”表中。您可以通过重复步骤 3 - 8 来添加更多 URL。
9. 要配置您的测试设置，请单击**下一步**。

   将显示测试配置的摘要。使用缺省设置时，将显示以下消息：

   测试将发生：每 15 分钟从 3 个公用位置和 0 个专用位置同时执行，以确定测试是否超过指定阈值。
10. 在**设置**窗格中，单击**编辑**以显示测试的当前设置。
    您可以更新以下设置：

    - **时间间隔**用来定义测试运行的频率。
    - **测试频率**用来确定测试是从所有位置同时运行，还是按每个时间间隔从不同位置运行。选择**同时**以同时从所有位置运行测试，或选择**错开**以在每个时间间隔从一个不同的选定位置运行您的测试。
    - **位置**用来确定运行测试的位置
11. 从**公用位置**列表中选择位置。要选择从中运行测试的专用位置，您必须在要从中运行测试的机器上先安装和配置专用 PoP。有关更多信息，请参阅第 895 页的『安装和配置专用 PoP 位置』。
12. 单击**保存**以完成配置您的测试；然后，单击**完成**。
    此时将显示 Availability Monitoring 仪表板。一分钟后，该仪表板显示用于新测试的信息和数据。

创建 Web 页面测试

创建 Web 页面测试可测试 Web 应用程序的可用性并监视打开该页面花费的时间。

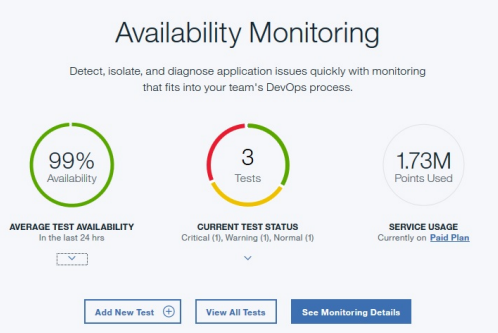
关于此任务

Web 页面测试报告用于装入 Web 应用程序 URL 的响应时间。创建 Web 页面测试以监视 Web 应用程序的可用性和响应时间。

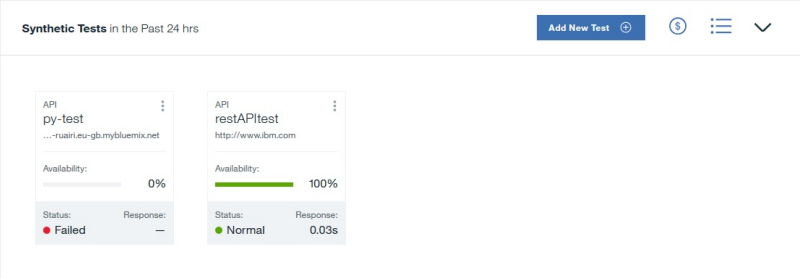
过程

要创建 Web 页面测试，请完成以下步骤。

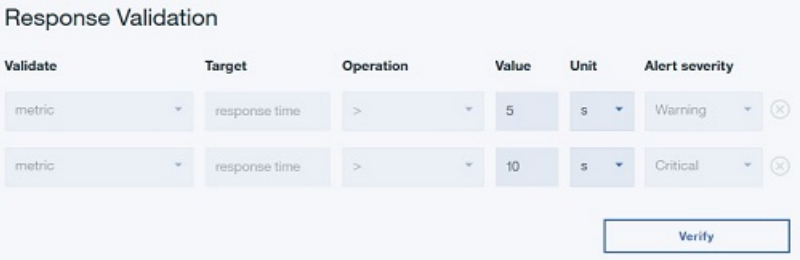
1. 如果要查看 Availability Monitoring 摘要页面，请单击**添加新测试**。



如果要查看 Availability Monitoring 仪表板，请单击**合成测试**窗格上的**添加新测试**。



2. 在“**监视设置**”页面上单击**单个操作**；然后，在“**单个操作**”页面上单击 **Web 页面**。
3. 在**名称**字段中输入对您的测试有意义的名称。将测试目的描述添加到**描述**字段。
4. 选择要监视的 Web 应用程序的 **URL**。
5. 在“**响应验证**”部分中为您的测试配置警告警报阈值和紧急警报阈值。为每行编辑**值**和**单位**。
   超出警告阈值和紧急阈值的响应时间触发警报。



6. 使用**黑名单**和**白名单**可指定要将请求发送至哪些 URL 和域，以及哪些 URL 和域会影响应用程序测试的度量值和状态。请将您要包含或阻止的 URL 和域添加到**白名单**和**黑名单**。
   有关更多信息，请参阅第 894 页的『使用白名单和黑名单进行阻止和过滤』。
7. 单击**验证**以创建 Web 页面测试并确定您的测试请求是否有效。

   Availability Monitoring 通过将 GET 请求发送到您的测试 URL 来确定测试有效性。在测试验证期间未发生响应验证。

   已验证的测试显示在“**已验证的项**”表中。您可以通过重复步骤 3 - 7 来添加更多 URL。
8. 要配置您的测试设置，请单击**下一步**。

   将显示测试配置的摘要。使用缺省设置时，将显示以下消息：

   测试将发生：每 15 分钟从 3 个公用位置和 0 个专用位置同时执行，以确定测试是否超过指定阈值。

   每个月的测试估算用量和估算数量是根据您的当前测试配置而显示。
9. 在**设置**窗格中，单击**编辑**以显示测试的当前设置。
   您可以更新以下设置：

   · **时间间隔**用来定义测试运行的频率。

- **测试频率**用来确定测试是从所有位置同时运行，还是按每个时间间隔从不同位置运行。选择**同时**以同时从所有位置运行测试，或选择**错开**以在每个时间间隔从一个不同的选定位置运行您的测试。
- **位置**用来确定运行测试的位置

从**公用位置**列表中选择位置。要选择从中运行测试的专用位置，您必须在要从中运行测试的机器上先安装和配置专用 PoP。有关更多信息，请参阅第 895 页的『安装和配置专用 PoP 位置』。

单击**保存**以完成配置您的测试。

10. 单击**完成**。

    此时将显示 Availability Monitoring 仪表板。一分钟后，该仪表板显示用于新测试的信息和数据。

从已上载的脚本创建脚本测试

上载 Selenium 脚本以创建脚本测试，用于测试 Web 应用程序的可用性和性能以对模拟用户行为做出响应。

开始之前

要创建脚本测试，您必须先创建 Selenium 脚本。有关创建 Selenium 脚本的更多信息，请参阅记录合成脚本。

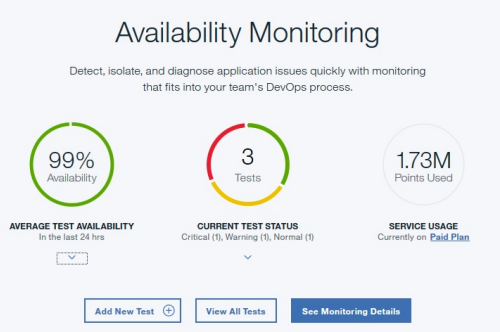
关于此任务

创建脚本测试以监视模拟用户与 Web 应用程序交互的 Selenium 脚本。如果您创建 Selenium 脚本来模拟登录您的应用程序的用户，那么您可以定期运行脚本测试对应用程序性能进行测试以响应模拟用户操作。

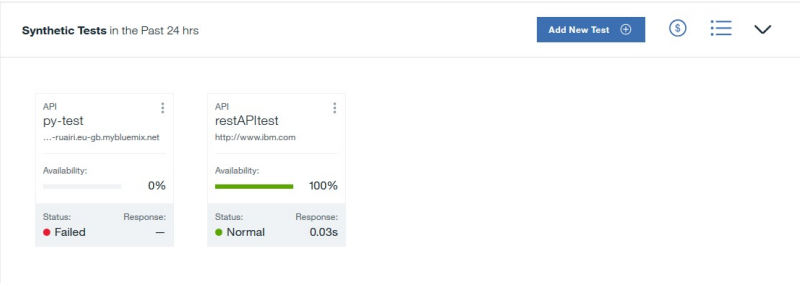
过程

要创建脚本测试，请完成以下步骤。

1. 如果要查看 Availability Monitoring 摘要页面，请单击**添加新测试**。



如果要查看 Availability Monitoring 仪表板，请单击**合成测试**窗格上的**添加新测试**。



2. 在“**监视设置**”页面上单击**脚本行为**。“**脚本行为设置**”页面随即显示。单击**上载文件**。

   如果稍后返回以编辑此测试，您可以下载已上载的脚本文件。单击**下载**图标 以下载您的脚本。
3. 在**名称**字段中输入对您的测试有意义的名称。将测试目的描述添加到**描述**字段。
4. 单击**浏览**以找到并上载脚本文件。
5. 使用**黑名单**和**白名单**可指定要将请求发送至哪些 URL 和域，以及哪些 URL 和域会影响应用程序测试的度量值和状态。请将您要包含或阻止的 URL 和域添加到**白名单**和**黑名单**。

   有关更多信息，请参阅第 894 页的『使用白名单和黑名单进行阻止和过滤』。

6. 要配置您的测试设置，请单击**下一步**。

   将显示测试配置的摘要。例如，使用缺省设置时，将显示以下消息：

   测试将发生：每 15 分钟从 3 个公用位置和 0 个专用位置同时执行，以确定测试是否超过指定阈值。

   每个月的测试估算用量和估算数量是根据您的当前测试配置而显示。
7. 在**设置**窗格中，单击**编辑**以显示测试的当前设置。

   您可以更新以下设置：

   - **时间间隔**用来定义测试运行的频率。
   - **测试频率**用来确定测试是从所有位置同时运行，还是按每个时间间隔从不同位置运行。选择**同时**以同时从所有位置运行测试，或选择**错开**以在每个时间间隔从一个不同的选定位置运行您的测试。
   - **临界阈值**用来为测试定义紧急警报的响应时间。
   - **警告阈值**用来为测试定义警告警报的响应时间。
   - **位置**用来确定运行测试的位置。

   请从缺省情况下显示的**公用位置**列表中选择位置。要选择从中运行测试的专用位置，您必须在要从中运行测试的机器上先安装和配置专用 PoP。有关更多信息，请参阅第 895 页的『安装和配置专用 PoP 位置』。

   如果需要，可输入在测试脚本中定义的变量的值。例如，如果您的脚本需要用户名和密码来连接到 Web 站点，那么您可以输入这些变量的值。您可以在**脚本变量**表中为不同位置的变量设置不同值。

   单击**保存**以完成配置您的测试。
8. 单击**完成**。

   此时将显示 Availability Monitoring 仪表板。一分钟后，该仪表板显示用于新测试的信息和数据。

使用白名单和黑名单进行阻止和过滤

使用白名单和黑名单可确定要将请求发送至哪些资源，以及哪些资源会影响应用程序测试的度量值和状态。白名单和黑名单仅适用于 Web 页面和脚本化行为测试。

白名单和**黑名单**字段定义您的测试可访问或无法访问的资源，以及影响测试度量值及状态的资源。白名单和黑名单控制哪些依赖项和资源会影响所测试 Web 应用程序的响应时间，例如第三方度量值。您可以在创建 Web 页面或脚本化行为测试时配置白名单和黑名单。

您可使用**白名单**来定义所允许的域和 URL；然后，使用**黑名单**来阻止所允许位置的特定元素。

语法

请使用逗号 (,) 来分隔黑名单和白名单中的各个项。使用通配符 (*) 来过滤每个 URL 或域的元素。

白名单

将要包括在请求和度量值计算中的 URL、模式或域添加到“白名单”字段中。在白名单中最多可以列出 10 个项。每个项的长度不可超过 200 个字符。所有与白名单中的项不匹配的域、模式和 URL 都将被阻止。

例如：`ibm.com, *developerworks*, *.s81c.com/*, https://www.ibm.com*, https://*`

注: 如果您的白名单 URL 过滤器包含 `http://` 或 `https://`，必须紧跟在 URL 之后包含通配符 (*)，例如，`https://www.ibm.com*`。

黑名单

将要排除在请求和度量值计算之外的 URL、模式或域添加到“黑名单”字段中。在黑名单中最多可以列出 20 个项。每个项的长度不可超过 200 个字符。

例如：`*.profile.*.cloudfront.net/*.png, http://*`

注: 如果您的黑名单 URL 过滤器包含 `http://` 或 `https://`，必须紧跟在 URL 之后包含通配符 (*)，例如，`https://www.ibm.com*`。

过滤和阻止行为

测试可以同时具有白名单和黑名单。确定要允许或阻止的位置时，黑名单始终会覆盖白名单。下表显示所有涉及白名单和黑名单的方案的过滤和阻止行为。

表 *253.* 白名单和黑名单的过滤和阻止行为

黑名单	白名单	行为	原因
空	空	允许访问	未输入任何过滤规则。
空	URL 与列表条目不匹配	阻止访问	URL 不在白名单中。
空	URL 与列表条目匹配	允许访问	URL 在白名单中。没有任何阻止访问的黑名单条目。
URL 与列表条目不匹配	空	允许访问	URL 不在黑名单中。没有阻止访问非白名单上 URL 的白名单条目。
URL 与列表条目匹配	空	阻止访问	URL 在黑名单中。
URL 与列表条目不匹配	URL 与列表条目不匹配	阻止访问	URL 不在白名单中。
URL 与列表条目不匹配	URL 与列表条目匹配	允许访问	URL 在白名单中。URL 不在黑名单中。
URL 与列表条目匹配	URL 与列表条目不匹配	阻止访问	URL 不在白名单中。URL 在黑名单中。
URL 与列表条目匹配	URL 与列表条目匹配	阻止访问	URL 在黑名单中。黑名单条目覆盖白名单条目。

安装和配置专用 PoP 位置

将专用 PoP 下载并安装到本地机器；然后配置专用 PoP 以用作 Availability Monitoring 中您的测试的位置。

开始之前

要安装专用 PoP，您的专用 PoP 的安装位置必须满足以下需求：

- Linux 随内核 V3.1.0 或更高版本一起安装。
- 已安装并启动 Docker 服务 V1.7.1 或更高版本。
- 可用磁盘空间为 4 GB 或更大。
- 可用内存为 2 GB 或更大。
- CPU 核心：
  - 如果仅在专用 PoP 中需要 REST API 回放，那么至少有 2 个可用 CPU 核心。
  - 如果想要在专用 PoP 中运行 Web 页面回放和脚本回放，那么对于每 1 个或 2 个测试有 1 个 CPU 核心以用于每分钟运行。
- 请在添加新测试之前和之后，以及应用包含更新版本的 Firefox 或 Selenium IDE 的专用 PoP 软件更新之后，检查专用 PoP CPU 和内存使用情况，因为更高版本可能具有更高的系统需求。

确定何时添加 CPU 核心的最佳实践是在专用 PoP 主机上运行需求最苛刻的进程以获取 CPU 使用率和内存使用情况：如果总 CPU 使用率大于 70% 并且 CPU 使用率排名最高的进程是 Firefox，那么添加 CPU 核心直至总 CPU 使用率低于 50%；如果专用 PoP 主机上的可用内存低于 500 MB，请增加内存。

如果没有更多硬件资源，但是想要专用 PoP 运行而无异常，请执行以下步骤以降低 Firefox 并行运行实例计数（这会导致测试运行时间间隔超过在 UI 中配置的值，因为您的硬件资源无法运行大量测试）：

1. 编辑 `start-pop.sh` 脚本添加 **MAX_TASKPOOL_SIZE** 环境变量，在专用 PoP 主机中输入可用 CPU 核心作为值，然后运行 **stop-pop.sh**，随后是 **start-pop.sh**。
2. 将测试设置为比 UI 中更长的时间间隔。

您必须对要安装专用 PoP 的机器的命令行界面 (CLI) 具有用户访问权。您还必须具有必需的用户访问权才能将软件包添加到 Docker。

要点:

- 确保想要运行专用 PoP 的机器的系统时间是标准时间，并且保持与标准时间同步。否则，测试实例在 Availability Monitoring 仪表板上显示不正确的时间戳记。
- 以下平台完全支持 Availability Monitoring 专用 PoP：Red Hat Enterprise Linux 7.4 和 CentOS Linux 7.4。

关于此任务

除了公用位置，在 Availability Monitoring 中创建或编辑测试时，您可以部署专用存在点 (PoP)。使用专用 PoP 来测试位于公司防火墙之后的应用程序，例如具有更高隐私或安全需求的应用程序。您最多可在 Availability Monitoring 中注册 50 个专用位置。下载预检查脚本和专用 PoP 软件包；然后，在要运行专用 PoP 的机器上保存该脚本和软件包。

过程

1. 创建测试或编辑现有测试。

   要创建测试，请单击**合成测试**窗格上的**添加新测试**。要编辑测试，请单击**操作** ⋮ ；然后单击**编辑**。如果要创建测试，请配置并验证您的测试。在“**设置**”部分中，单击**编辑**。

   有关更多信息，请参阅第 889 页的『创建 REST API 测试』、第 891 页的『创建 Web 页面测试』和第 893 页的『从已上载的脚本创建脚本测试』。
2. 单击“**设置**”部分中的**编辑**以显示“**位置**”部分；然后，单击**专用位置**。如果要编辑上一个测试，请单击“**位置**”部分中的**专用位置**。

   如果先前已安装一个或多个专用 PoP，那么显示所有已安装的专用 PoP 的列表。如果未安装和配置专用 PoP，那么 Availability Monitoring 可以指导您设置专用 PoP。
3. 单击**下载预检查**并将预检查脚本保存到要运行测试的机器。

   要点: 您必须从命令行界面 (CLI) 抽取和运行脚本以安装专用 PoP。专用 PoP 脚本和软件包可以安装在其他机器上，并通过该机器的 CLI 进行访问。当您正在使用专用 PoP 脚本或丢失任何未保存的测试设置时，请不要在浏览器中关闭或刷新 Availability Monitoring。

   为要查找专用 PoP 的机器打开 CLI。从 CLI 中，导航到保存预检查脚本的位置；然后运行预检查脚本，如下所示：

   ```
   ./precheck.sh
   ```

   确保您具有必需的许可权以在机器上执行 shell 脚本。

   预检查脚本显示检查结果。如果您的环境未能通过检查，请更新您的计算机以满足显示的需求。
4. 返回到 Availability Monitoring，单击**下载软件包**，然后保存软件包。将软件包移动到要运行测试的机器。从 CLI 中，导航到保存下载软件包的位置；然后，运行以下命令来抽取软件包：

   ```
   tar -xvf Availability_Monitoring_PoP.tar
   ```

   其中，*Availability_Monitoring_PoP.tar* 是包含您所下载的专用 PoP 软件包的 `.tar` 文件的名称。
5. 配置专用 PoP。从 CLI，运行以下脚本：

   ```
   ./config-pop.sh
   ```

   提示时，输入专用 PoP 的以下信息：

   - PoP 名称
   - 国家或地区位置
   - 城市位置
   - PoP 纬度
   - PoP 经度
   - PoP 描述

6. 如果任何 REST API 测试连接到使用自签名证书或者未由众所周知的 CA 证书提供者签名的服务器，请将 `.pem` 文件格式的任何可信 CA 证书放入 `keyfiles` 目录中。

   注:

   - 对 `.pem` 证书文件执行的任何更改都需要重新启动专用 PoP。
   - 要测试的服务器应在 TLS 握手期间发送所有证书（但根 CA 证书除外）；如果未发生这种情况，请更正服务器配置（如果可能）。否则，您可以将任何缺少的证书添加到 `keyfiles` 目录，如此步骤中所述。但是，PoP 测试可能无法反映其他客户机的体验。

7. 要配置专用 PoP 以在运行 Web 页面测试或脚本编制行为测试时使用代理服务器，请输入以下某个选项:

   要点: 使用手动或自动代理配置从专用 PoP 位置运行的 REST API 测试不使用该代理。只有 Web 页面和脚本编制行为测试可以使用代理服务器从专用 PoP 位置运行。

   no
   输入 no 以将专用 PoP 配置为在运行测试时不使用代理。

   manual
   输入 `manual` 以手动配置代理 IP 地址和端口号，以在运行测试时供专用 PoP 代理使用。脚本按照以下格式请求代理服务器 IP 地址和端口号: *`ip address:port number`*。您也可以创建无代理列表来阻止域元素、主机元素或 IPv4 地址元素。提示时，输入一个或多个域元素或 IPv4 地址元素。以空格或逗号（“,”）分隔各个列表项。不支持使用通配符 (*)。

   - 要阻止域和任何子域，请以点开头输入域后缀，例如: `.example.org, example.org`。
   - 要阻止网络，输入具有 CIDR 后缀的 IP 地址，以标识要阻止的 IP 地址范围，例如: `10.0.0.0/8`。

   pac
   输入 pac 以将专用 PoP 配置为使用自动代理配置 URL。当脚本提示时，输入自动代理配置 URL。

   您的专用 PoP 设置保存在 `pop.properties` 文件中。

8. 启动专用 PoP。从 CLI，运行以下脚本:

   ```
   ./start-pop.sh
   ```

   在运行专用 PoP 时，它可以被 Availability Monitoring 找到。

9. 返回到 Availability Monitoring，并单击  **刷新位置**以找到并显示新的专用 PoP。
   您的专用 PoP 将在表中列出。

10. 要选择专用 PoP 作为用于测试的位置，请选中包含专用 PoP 的表行的复选框。要删除专用 PoP，请完成以下步骤:

    a) 从 CLI，在专用 PoP 所在的机器上运行 **`./stop-pop.sh`** 脚本。

    b) 返回到 Availability Monitoring，并单击包含想要删除的专用 PoP 的表行上的  **删除**。

11. 重复步骤 3 - 10，向不同的机器添加更多专用 PoP 以在 Availability Monitoring 中作为位置选择。单击**完成**以保存并启动您的测试。
    此时将显示 Availability Monitoring 仪表板。在大约 1 分钟后，该仪表板显示用于新测试的信息和数据。

12. 可选： 要升级现有的专用 PoP，请完成以下步骤:

    a) 将新的专用 PoP 包下载到新文件夹；然后，使用 **`tar -xvf`** 命令将该新的 PoP 解压到该文件夹中。

    b) 从 CLI，将目录切换到旧的专用 PoP 所在的文件夹。运行以下脚本停止旧的专用 PoP:

    `./stop-pop.sh`

    c) 在旧的专用 PoP 所在的目录中，备份您的现有 `system.properties` 和 `pop.properties` 文件。

要点: system.properties 文件包含允许专用 PoP 连接到 Cloud APM 服务器的关键信息。pop.properties 文件包含您的专用 PoP 的配置数据。如果您想要保留此配置，请确保保留旧的专用 PoP 的 pop.properties 和 system.properties 文件，然后再升级您的专用 PoP。

d) 复制新的专用 PoP 文件夹中的所有文件（除 pop.properties 和 system.properties 以外），并替换旧的专用 PoP 所在的文件。

e) 如果需要重新配置升级后的专用 PoP，请从 CLI 运行以下脚本：

./config-pop.sh

f) 从 CLI 运行 **./start-pop.sh** 以启动升级后的专用 PoP。

g) 返回到 Availability Monitoring，并单击  **刷新位置**以找到并显示升级的专用 PoP。

Availability Monitoring 中的警报生成

在 Availability Monitoring 中，测试最多可生成共计三个警报。您的测试将报告严重性为最高的警报，直到造成警报的状况解决为止。

对三个不同的情况引发单独的警报。

- Web 应用程序或 URL 的响应时间超出为测试所设置的警告阈值或紧急阈值。缺省情况下，每个测试度量响应时间，并根据该测试的警告阈值和紧急阈值来引发警报。
- 测试所返回的 HTTP 响应代码指出 Web 应用程序或 URL 由于客户机或服务器错误而无法使用。缺省情况下，每个测试都会检查响应代码，以确定测试是成功还是失败。
- 您的测试确定满足一个还是多个定制条件，将引发具有由一个或多个定制条件定义的最高严重性的警报。Availability Monitoring 在确定是否引发警报时将整体考虑所有定制条件。在您的测试确定所有定制条件不再生成任何警告警报或紧急警报前，此警报仍然存在。

引发多个警报时，Availability Monitoring 将报告严重性为最高的警报，只要存在任何警报即如此。

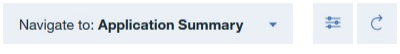
例如，如果添加引发紧急警报的定制条件以及引发警告警报的另一个定制条件，那么您的测试生成紧急警报。此警报在 Availability Monitoring 仪表板上可视。如果导致紧急警报的条件不再为 true，那么测试警报的严重性更改为“警告”。在所有条件不再引发警报前，警报仍然存在。

在“监视仪表板”上查看应用程序可用性和性能

您可以在 Availability Monitoring 仪表板上查看应用程序可用性和性能的详细信息，以及警报和任何相关测试。

Availability Monitoring 仪表板分为以下窗格：

- 应用程序摘要
- 警报频率
- 合成测试
- 响应时间和可用性

使用**导航到**下拉菜单 

以快速导航到任何窗格。

使用这些指南以帮助您了解 Availability Monitoring 的特性。要打开指南，请单击**帮助**图标  ；然后，单击要查看的指南。

视频教程库

视频教程库包含有关如何创建 Availability Monitoring 测试、使用 Selenium IDE 创建测试脚本以及发送警报的视频。

欢迎监视！

“欢迎监视”指南突出显示仪表板的区域并说明 Availability Monitoring 的每个功能。

您可以从“应用程序摘要”窗格、“警报频率”窗格、“合成测试”窗格或“响应时间和可用性”窗格访问**明细**仪表板。**明细**仪表板显示您的测试实例的关键统计信息。

您可以更改窗格顺序来满足您的需要。要移动一个窗格，请单击标题并将该窗格拖到其他位置。要保存这些更改，以便在您注销后仍存在，请单击**保存布局**。

您可以设置仪表板以便每分钟自动刷新一次。单击**配置**图标 ；然后，单击**刷新**滑动条以选择 **1 分钟**。要随时刷新您的页面，请单击**刷新**。

应用程序摘要

“应用程序摘要”窗格显示过去 24 小时的警报状态概述以及当前测试状态信息。

“应用程序摘要”窗格显示以下信息：

- **当前状态** 显示所有测试的最高严重性状态。严重性可以是正常、警告或紧急。
- **警报**显示开启的警报的数量并将其分为警告警报和紧急警报。
- **可用性报告**允许您下载 `.csv` 文件报告，其中包含应用程序的每月、每周和每日平均可用性和响应时间。

  单击**报告**图标 以下载报告。

警报频率

警报频率面板包含显示最新警报的图。警报按位置分组，并在**警报**表中列出。

“警报频率”图

警报频率图显示用于测试的所有公用和专用存在点 (PoP) 的概览信息。

使用缩放功能来放大图的任何区域，或者将其恢复至其原始大小。将鼠标指针悬停在每个位置以查看该位置的名称，以及该位置的警告警报和紧急警报的数量。您可以通过从**警报**下拉列表中选择**全部**、**已打开**或**已关闭**来过滤在图上显示的警报。

PoP 位置

PoP 位置图标 0 指示测试的 PoP 位置。每个 **PoP 位置**图标的颜色都表示每个位置中最新警报的严重性：正常 0、警告 1 或严重 1。**PoP 位置**动画图标指示在用于测试实例的所有位置中，此位置是否具有严重级别最高的最多警报。

将 PoP 位置添加到您选择的测试，方法是将鼠标指针悬停在**不活动的 PoP 位置**图标 上并单击**在此处测试**。对您所选的测试显示**测试编辑方式**页面。您可以从**响应时间**和**可用性**上的**测试**下拉菜单中选择测试。

由**专用 PoP 位置**图标 2 表示的专用 PoP 位置

警报数量

PoP 位置图标显示已打开的警报、已关闭的警报或在每个位置生成的所有警报的数量。**紧急**、**警告**和**正常**图标 **12** Critical **11** Warning **0** Normal 显示您所在位置的每个严重性的警报数。

失败的测试

发生失败测试的位置由 **PoP 位置**图标指示，该图标具有断开的边界 10。

使用缩放功能来放大图的任何区域，或者将其恢复至其原始大小。将鼠标指针悬停在每个位置以查看该位置的名称，以及该位置的警告警报和紧急警报的数量。您可以通过从**警报**下拉列表中选择**全部**、**已打开**或**已关闭**来过滤在图上显示的警报。

“警报”表

所有位置的警报显示在一个表中。



此表显示有关警报的以下信息：

- **严重性**将警报描述为紧急或警告。
- **时间戳记**显示创建警报的时间。
- **描述**总结了测试实例的性能。
- **触发者**显示触发警报的测试的名称。
- **位置**说明在哪里发生了问题。
- **状态**显示警报是打开的还是关闭的。

查看警报详细信息

表中的每个警报都包含指向**明细**仪表板的链接。使用明细仪表板可帮助您对引发警报的问题进行故障诊断。

过滤警报

要过滤特定位置的警报，请单击图上的 **PoP 位置**图标。要显示所有位置的警报，请单击图上除 **PoP 位置**图标以外的任意位置。

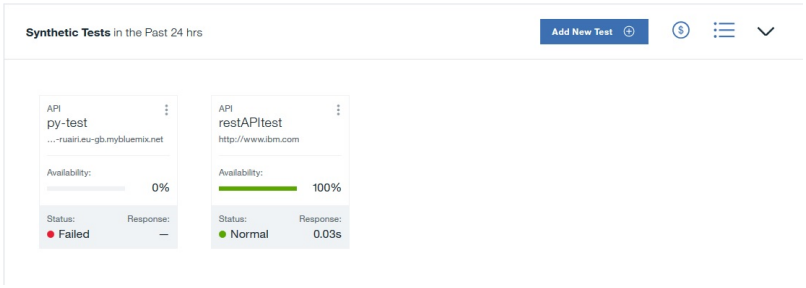
要按严重性过滤表中的警报，请单击**紧急**、**警告**或**正常**图标 12 Critical 11 Warning 0 Normal 。要移除过滤器并包含表中每个严重性的警报，请再次单击所选图标。

更改警报阈值

警报被您在创建测试时指定的阈值触发。在大多数情况下，生成警报是因为可用性失败或响应时间太长。要更改阈值设置，请单击在**合成测试**面板中生成警报的测试上的**操作**图标 并单击**编辑**。

合成测试

在**合成测试**窗格中，您可以创建、编辑、删除和查看用于监视应用程序的性能和可用性的合成测试。测试显示在**合成测试**窗格中的列表或卡视图中。



每一个测试卡都会显示该测试的相关信息：

可用性
显示该测试在过去 24 小时内的可用性百分比。

状态
显示该测试的当前状态。状态可以是“紧急”、“警告”、“正常”、“失败”、“不活动”或“未知”。

平均 响应
显示该测试在过去 24 小时内的平均响应时间。

您可以监视三种不同类型的测试：

REST API
报告 REST 调用的响应时间。支持所有 HTTP 请求格式，例如 GET、POST、PUT 和 DELETE。

Web 页面
报告用于在您的 URL 中装入 Web 站点的响应时间。

脚本化行为
监视您创建的用于模拟用户与 Web 站点交互的 Selenium 脚本。例如，您可以创建模拟登录应用程序的用户的 Selenium 脚本。定期运行此脚本对应用程序的性能进行测试以响应此脚本自动执行的用户操作。有关创建 Selenium 脚本的更多信息，请参阅第 872 页的『记录合成脚本』。

要添加另一个测试，请单击**添加新测试**。

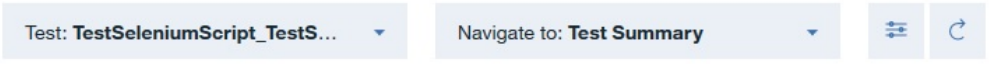
要停止、启动、删除或编辑合成测试，请单击**操作**图标 并单击所需的操作。要查看测试的**明细**详细信息，请单击测试。

要查看每个合成测试的具体使用情况，请单击**成本**图标 。如果预订了付费套餐，您的使用情况将以数据点的形式显示。

明细
明细仪表板显示测试的关键统计信息。该仪表板还汇总了过去 24 小时的可用性和响应时间信息、历史趋势以及测试性能数据。

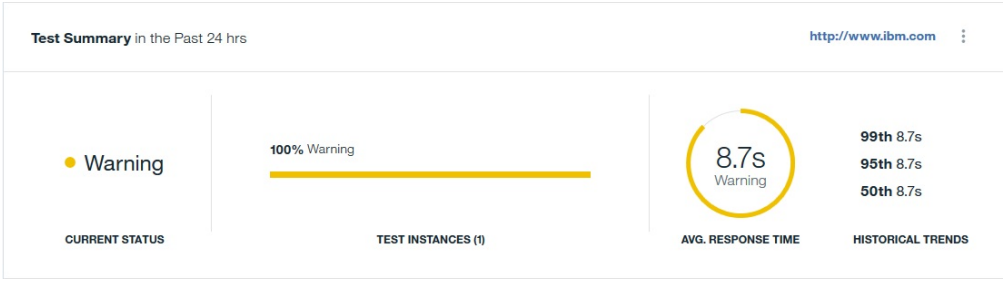
要查看测试的详细明细，请在**合成测试**窗格中单击测试。通过在“警报频率”窗格中的**警报**表中单击**明细**，您还可以打开**明细**仪表板。

使用**测试**下拉菜单来查看不同测试的明细。使用**导航到**下拉菜单来快速导航到任何窗格。



“明细”仪表板显示四个窗格。

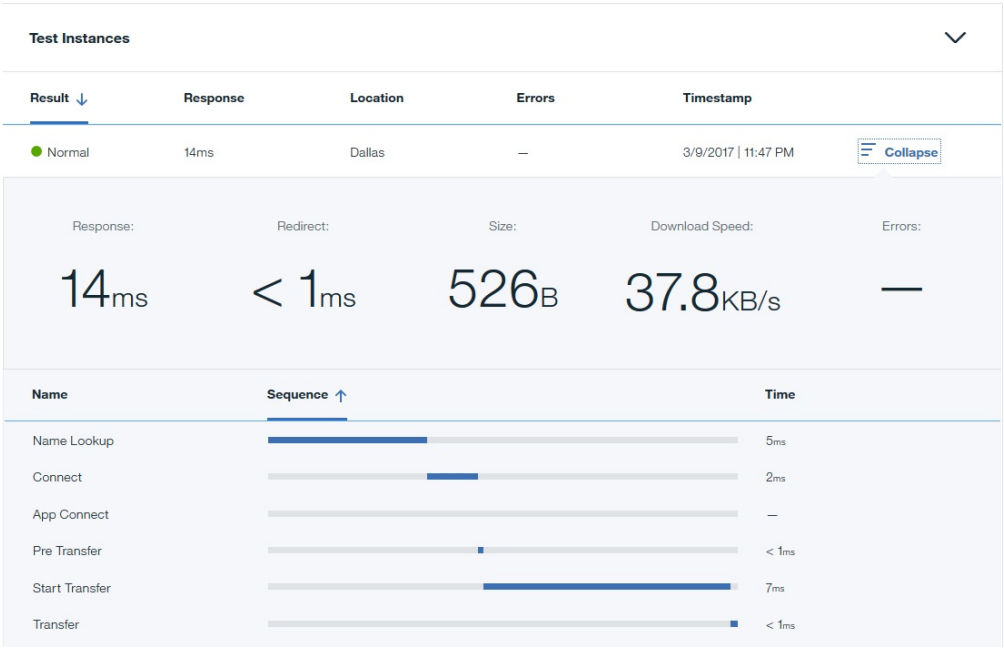
测试摘要



测试摘要窗格显示过去 24 小时内的以下测试信息：

- **当前状态**显示测试状态。
- **测试实例**显示正常、警告和紧急测试实例的百分比明细。
- **平均响应时间**显示测试的平均响应时间。
- **历史趋势**显示以秒或毫秒为单位的测试性能的历史趋势为 50%、95% 和 99%。

测试实例



测试实例表显示有关每个测试实例的详细信息，包括状态、响应时间、测试运行位置、错误数以及开始运行测试的时间戳记。要向下钻取到测试实例，请单击**展开**。将为测试实例中的每个步骤列出详细的响应信息。您可以对任何列排序，以帮助您快速确定发生减速或故障的确切步骤。查看错误、测试序列和响应时间可帮助您轻松地确定问题。

显示的信息取决于正在监视的合成测试的类型：

API

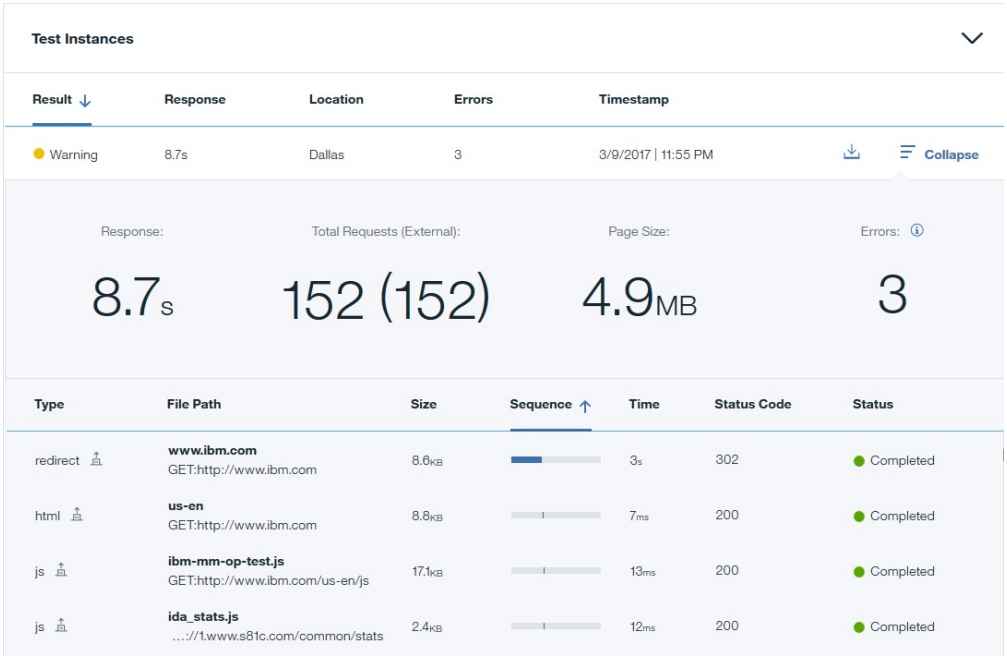
当您对 API 测试实例单击**展开**时，将显示以下详细信息的高级摘要：

- **响应**显示测试实例的总响应时间，包括重定向时间。
- **重定向**显示测试实例的重定向总时间。
- **大小**显示对象的大小。
- **下载速度**显示每个对象的下载速度。
- **错误数**显示在测试实例期间发生的错误数。要查看错误详细信息，请单击**信息**图标。

表显示 API 调用中的每个步骤，以及步骤名称、步骤顺序和每个步骤的响应时间。将显示以下步骤名称：

- **名称查找**表示测试实例解析对象名称所花费的时间。
- **连接**表示从步骤开始到与远程主机或代理的连接完成期间测试实例花费的时间。
- **应用程序连接**表示从步骤开始到与远程主机的 SSL 连接完成期间测试实例花费的时间。
- **传输前**表示从步骤开始一直到文件传输命令开始之前的这段时间内测试实例花费的时间。
- **开始传输**表示从步骤开始到接收到第一个字节期间测试实例花费的时间。
- **传输**表示测试实例传输文件花费的时间。

Web 页面



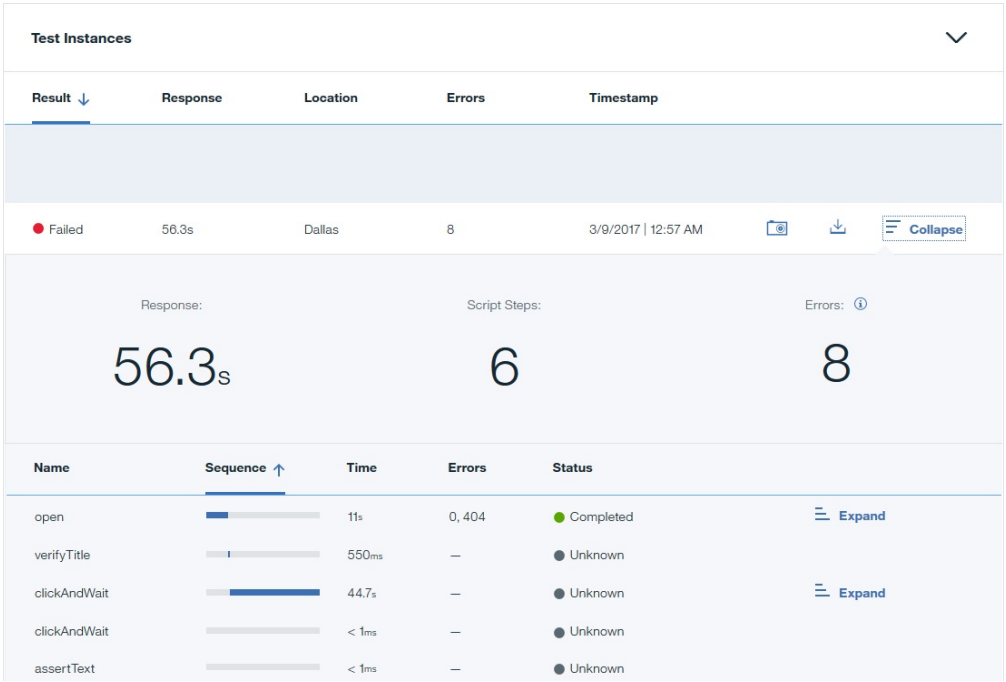
当您对 Web 页面测试实例单击**展开**时，将显示以下详细信息的高级摘要：

- **响应**说明测试实例的响应时间。
- **请求总数（外部）**显示测试实例的请求总数。外部请求数用于括号括起。
- **页面大小**显示 Web 页面的大小。
- **错误数**显示在测试实例期间发生的错误数。要查看错误详细信息，请单击**信息**图标。

还将显示一个表，其中列出由测试发出的每个请求的以下详细信息：

- **类型**显示请求类型，例如，HTML、CSS、JavaScript 或图像。外部和内部请求通过图标描述。
- **文件路径**描述请求的对象的位置。
- **大小**显示请求的对象的大小。
- **顺序**显示由测试发出的请求的顺序。
- **时间**显示每个请求花费的时间。
- **状态码**显示 HTTP 请求状态码。
- **状态**描述请求的结果，例如，已完成、未知或失败。

脚本

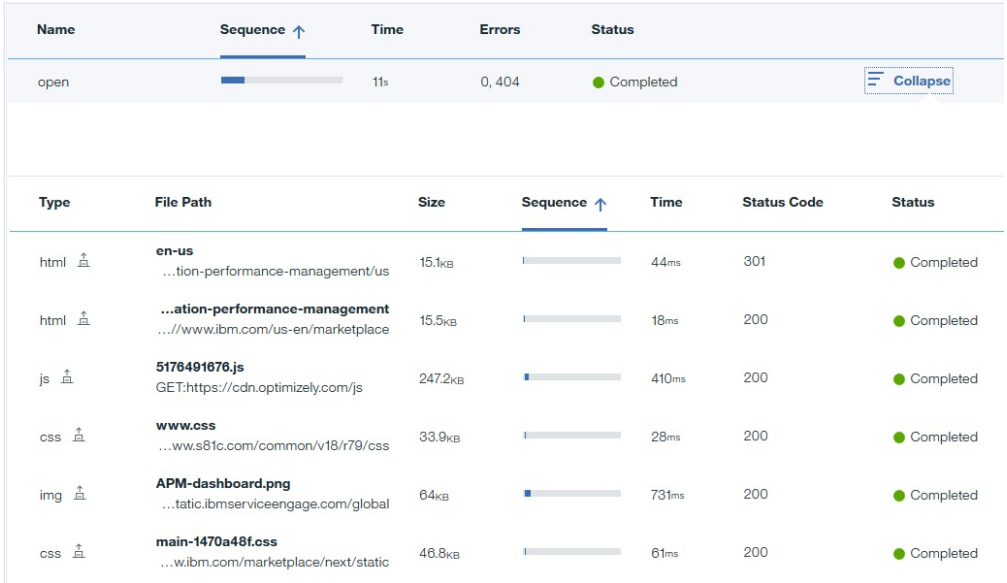


当您对脚本测试实例单击**展开**时，将显示响应时间、脚本步骤数和错误数。要查看错误详细信息，请单击**信息**图标。

每个脚本步骤的以下详细信息显示在一个表中：

- **名称**显示被测试实例调用的每个 Selenium 命令，例如，`Open`、`ClickAt` 或 `VerifyBodyText`。
- **顺序**显示从测试实例开始到结束期间脚本步骤的顺序。
- **时间**显示每个脚本步骤花费的时间。
- **错误**显示在执行每个脚本步骤期间发生的错误数。
- **状态**描述脚本步骤的结果，例如，已完成、未知或失败。

您可以向下钻取并查看由每个脚本步骤生成的请求的详细信息。



单击**展开**以查看包含以下详细信息的表：

- **类型**显示请求类型，例如，HTML、CSS、JavaScript 或图像。外部和内部请求通过图标描述。
- **文件路径**描述请求的对象的位置。
- **大小**显示请求的对象的大小。
- **顺序**显示由测试发出的请求的顺序。
- **时间**显示每个请求花费的时间。
- **状态码**显示 HTTP 请求状态码。
- **状态**描述请求的结果，例如，已完成、未知或失败。

如果 Web 页面装入失败或脚本中的步骤失败，那么 Availability Monitoring 可以自动创建屏幕捕获。例如，如果脚本中的某个步骤打开 Web 页面但是未装入该页面，那么 Availability Monitoring 自动创建屏幕捕获。要查看 Web 页面或脚本的屏幕捕获，请单击**截屏错误**图标 。此功能只能用于 Web 页面和脚本测试。它不会用于 REST API 测试。

通过单击**下载**图标 ，您还可以下载特定测试实例的网络流量记录作为 `.har` 文件。此功能可用于 Web 页面和脚本行为测试。

响应时间和可用性

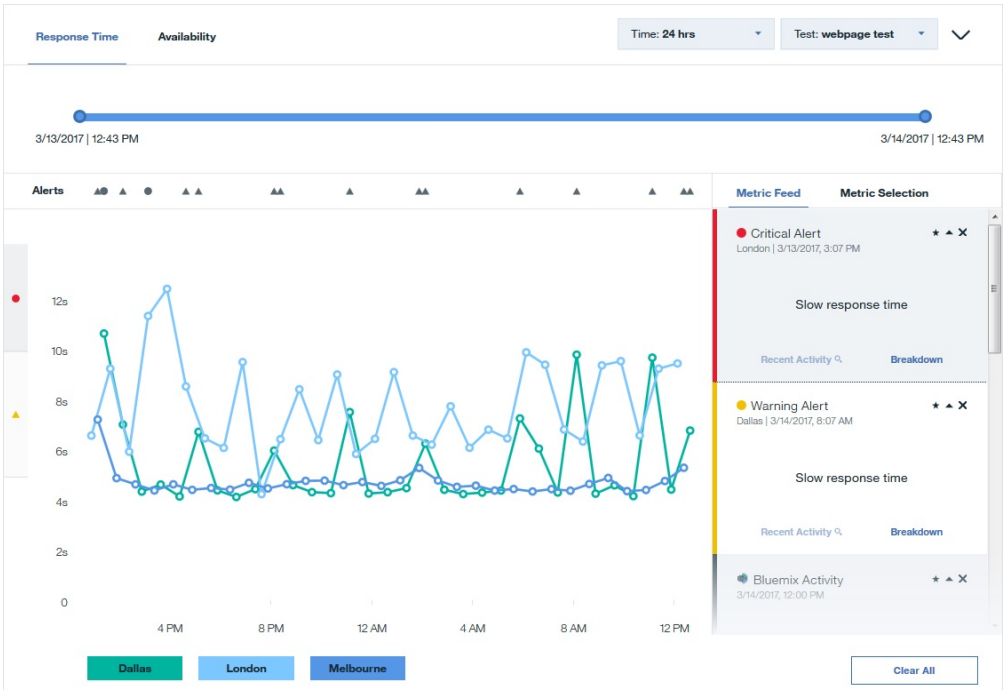
响应时间和可用性窗格显示在已定义的时段内针对测试实例的已度量响应时间和可用性的图形。有关更多信息，请参阅第 905 页的『响应时间和可用性』。

响应时间和可用性

使用**响应时间**和**可用性**窗格可帮助您将一段时间内的响应时间、可用性趋势和警告可视化。

响应时间图形

响应时间信息显示在折线图上。要查看该信息，请单击**响应时间**选项卡。



要点: 由 Availability Monitoring 度量的响应时间比用户经历的响应时间略长。Availability Monitoring 模拟实际用户行为，它将添加到响应时间度量中。响应时间较长是因为以下因素：

- Availability Monitoring 为每个测试创建一个新的 Firefox 实例，以阻止先前测试实例影响当前测试。因为浏览器高速缓存，实际用户可能会经历更快的响应时间。
- Availability Monitoring 在每次测试之前安装 Firefox Web 驱动程序插件。

测试的各响应时间通过折线图上的**响应点**图标 表示。不同颜色表示运行应用程序的不同地理位置。图形的 y 轴使用警报图标来确定警告阈值和紧急阈值范围。黄色警告图标 表示警告阈值范围，红色紧急图标 表示紧急阈值范围。单击黄色警告图标 或红色紧急图标 以轻松地确定在警告阈值和紧急阈值范围内显示的测试实例。要查看特定测试实例的详细信息，请单击图形上的**响应点**图标 。

过滤器

从**测试**下拉菜单中选择测试。您可以过滤 3 小时、24 小时、7 天、30 天和 12 个月的数据。对时间范围大于 24 小时的数据进行过滤时，将对在图形中显示的值求平均值。要查看更具体的信息，请单击该图形以向下钻取到各个警报和警告。您还可以使用滑块来缩小或扩大时间范围。

在“响应时间”图中，可以突出显示并隐藏特定 PoP 位置的数据。要突出显示特定位置的响应时间数据，可以将鼠标悬停于该 PoP 位置名称上；然后单击**突出显示位置**图标  。要隐藏某个位置的响应时间数据，将鼠标悬停于该 PoP 位置名称上；然后单击**隐藏位置**图标  。要在图形上恢复显示 PoP 位置数据，单击**添加更多位置** 或**度量值选择**选项卡；然后单击之前移除的 PoP 位置。

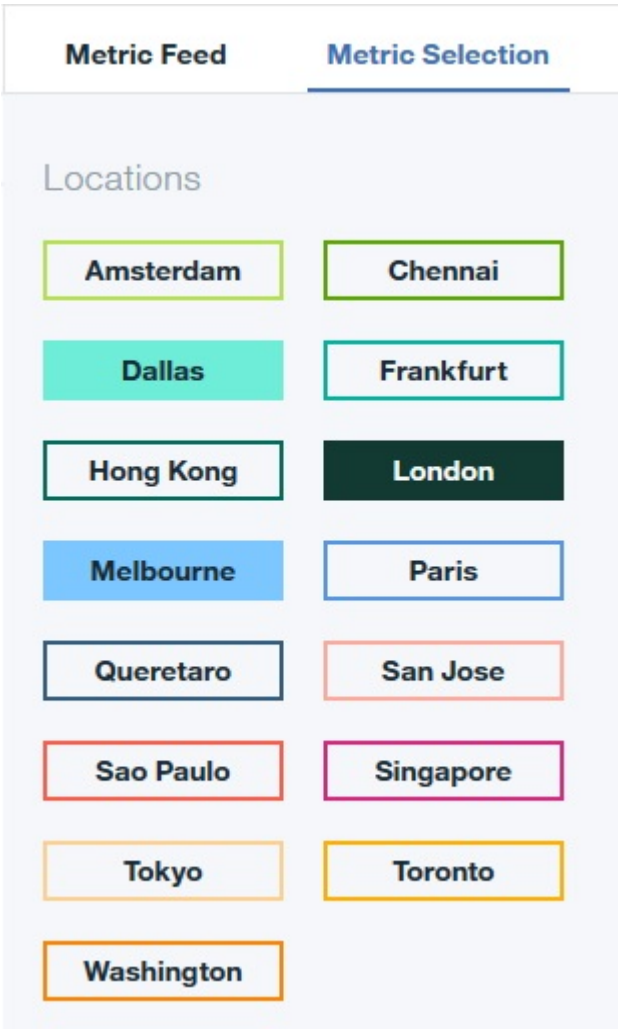
警报

您可以在“警报”行中轻松地确定警告警报和紧急警报。将鼠标光标悬停在**警报图标** 上以确定该警报的严重性和时间戳记。单击**警报图标**以在**度量订阅源**中显示该警报的详细信息。

如果在“警报”行上有多个邻近的图标，那么**数量图标**会显示在该时刻的警报数量。将鼠标悬停在**数量图标**上方即可显示个别警报，然后单击警报即可在**度量订阅源**中查看信息。

度量选择和度量订阅源

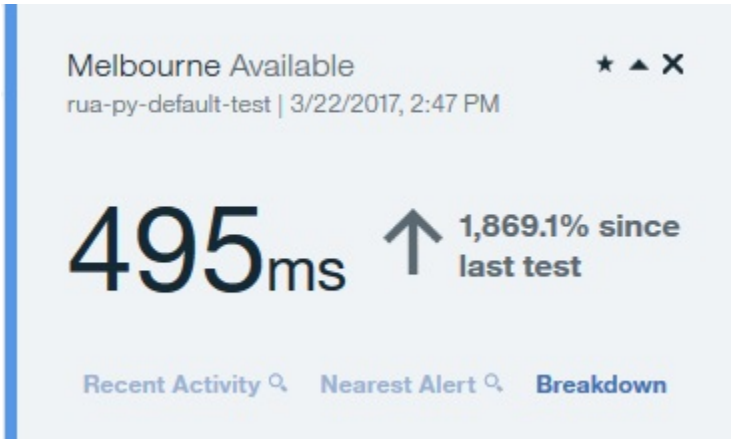
要按地理区域过滤度量，请单击**度量选择**。单击一个位置以在图形中添加或移除在该位置测量的度量值。单击**添加更多位置**以打开“测试编辑方式”页面，然后将 PoP 位置添加到所选测试。



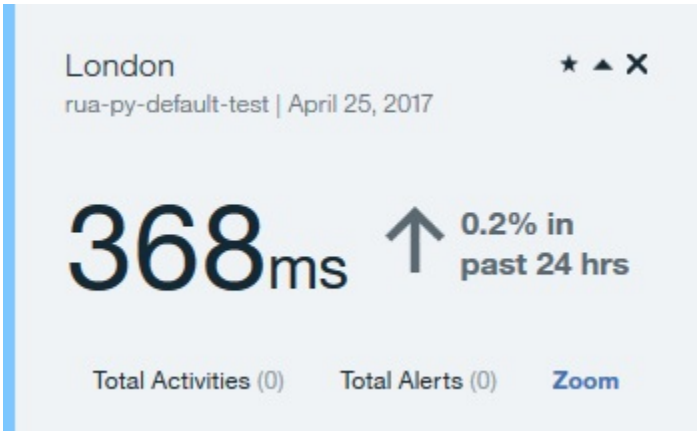
要查看度量详细信息的列表，请单击**度量订阅源**。**度量订阅源**显示满足度量的实例的列表。



在图上单击**警报图标**或**响应点** 以将该度量的详细信息添加到**度量订阅源**。



如果过滤时间范围大于 24 小时的“响应时间”图并单击**响应点**，那么您可以在**度量订阅源**中查看该天的汇总详细信息。

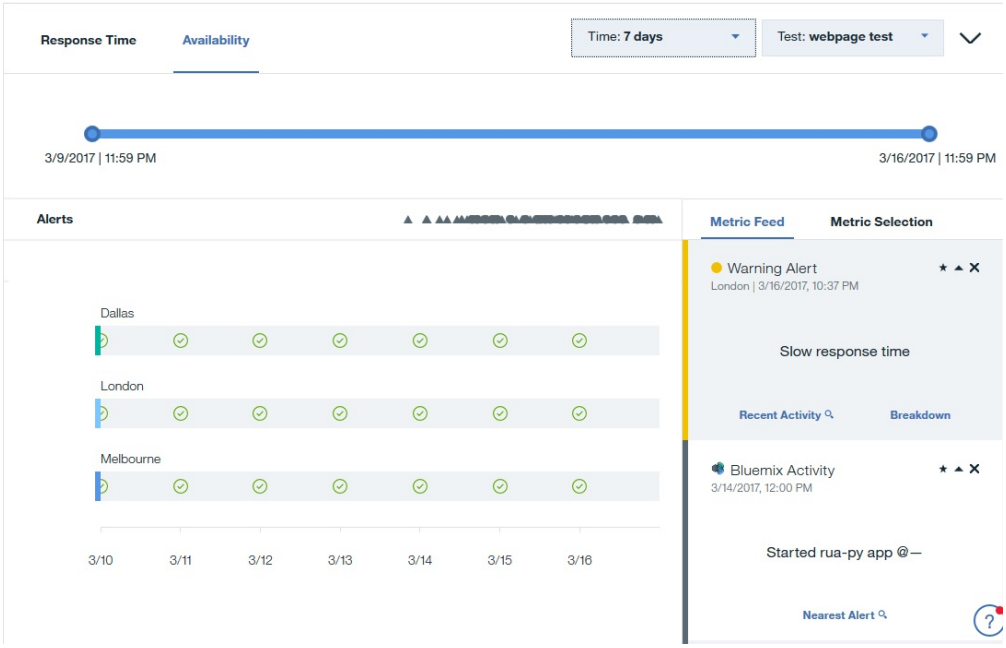


单击**缩放**以查看“响应时间”图形中针对那一天由测试生成的所有响应时间和警报。

要查看有关警报或测试响应时间的详细信息，请单击**度量订阅源**中的**明细**。如果发生了警报，请单击**最近的警报**以在**度量订阅源**中查看该测试实例的最近警报。

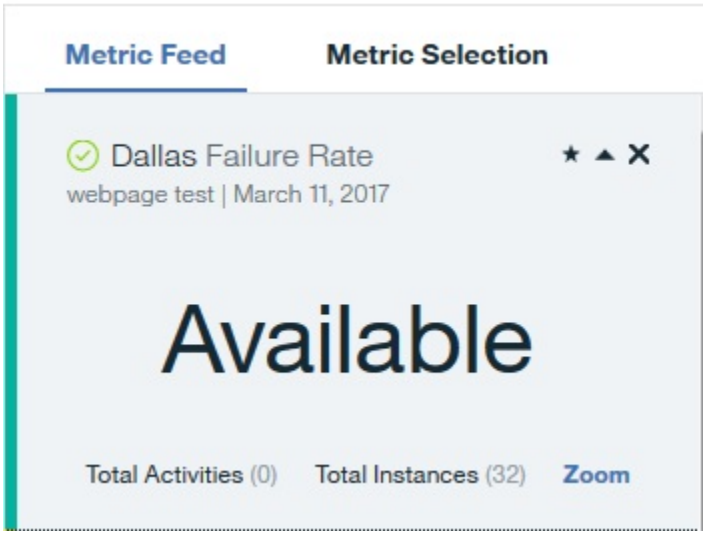
可用性

要查看您的应用程序的可用性信息，请单击**可用性**。“可用性图形”显示所选测试的每个存在点 (PoP) 的每日可用性。



在“可用性”图中，可以突出显示并隐藏特定 PoP 位置的数据。要突出显示特定位置的可用性数据，可以将鼠标悬停于该 PoP 位置名称上；然后单击**突出显示位置**图标  。要隐藏某个位置的可用性数据，将鼠标悬停于该 PoP 位置名称上；然后单击**隐藏位置**图标  。要在图形上恢复显示 PoP 位置数据，单击**度量值选择**选项卡；然后单击之前移除的 PoP 位置。

将鼠标指针悬浮在图形点上以显示特定一天和位置的故障率和测试实例数。单击图形点以在**度量订阅源**中显示此消息。



单击**缩放**对**可用性**图形和**响应时间**图形进行过滤以显示所选某天的信息。

Availability Monitoring 用量

您可在应用程序窗格上的**监视**选项卡上，以及在 Availability Monitoring 主要仪表板上，查看有关 Availability Monitoring 用量的详细信息。

要从 Availability Monitoring 主要仪表板查看用量概述，请单击**配置**图标  。如果您是 Availability Monitoring 试用版用户，那么用量会以条形图及百分比的形式，连同使用中的测试数量一起显示。如果您是付费套餐用户，那么用量会以数据点的形式显示。您可在**合成测试**窗格中，查看各个测试的用量详细信息。

用量是以数据点为单位进行测量。数据点估算数量是使用下列公式计算所得：

数据点估算数量 = T * L * (60/M * 24 * 30)/月

其中，T = 执行的合成测试数量，L = 位置数量，M = 测试之间的时间间隔（分钟）。

对于 Web 页面测试和 REST API 测试之类的简单测试，每一个测试使用 1 个数据点。对于 Selenium 脚本和 REST API 脚本之类的高级测试，每一个测试使用 100 个数据点。

浏览 API

使用 IBM Cloud Application Performance Management API 可创建脚本，以使 Cloud APM 环境自动上线。从 IBM developerWorks 上的 API Explorer 中 Cloud APM API 管理的服务供应项目中，您可访问和探索可用的资源组管理服务 API、阈值管理服务 API 以及基于角色的访问控制服务 API。

开始之前

您必须拥有活动的 Cloud APM 预订，以获取客户机标识密钥和运行 API 操作。

过程

1. 在浏览器中打开 API Explorer：https://developer.ibm.com/api。
2. 使用 IBM 标识登录。
3. 在**搜索所有 API** 字段中，输入 `performance management`，然后单击 。
4. 选择 IBM Cloud Application Performance Management API 图块。
5. 单击 API Explorer 窗口左侧的**文档**。
6. 选中特定的 API。
7. 选择子部分，以展开 API 操作列表。
8. 要继续操作，需要有效的 Cloud APM 预订。完成下列其中一个步骤：
   a) 如果您还没有 Cloud APM 预订，请注册免费的 30 天试用预订。
   b) 如果已有预订，请单击**我的 API** 登录，以在 API Explorer 中测试某些 API 操作。
   如果该预订有效，那么您登录 API Explorer 页面后，将会看到 API 预订列表。
9. 选择 API 操作以获取更多详细信息。
10. 从 API Explorer 页面顶部选择其中一种语言（例如，Shell 或 curl），以查看请求示例。
11. 检索客户端标识密钥和客户机密钥，并将它们存储在安全的位置以供外部使用。
    随每个 API 请求一起发送客户端标识密钥和客户机密钥。您必须要有 Cloud APM 预订才能完成此操作。

下一步做什么

有关如何运行 API 操作的更多信息，请参阅以下主题：

第 858 页的『访问和使用基于角色的访问控制服务 API』
第 847 页的『使用“资源组管理服务”API』
第 848 页的『使用“阈值管理服务”API』

高级配置

使用“**高级配置**”页面来控制通信设置和高级功能，例如事件转发。

在单击 **系统配置** > **高级配置**后，将在“高级配置”页面中显示以下配置类别。

UI 集成
对于与 Cloud APM 控制台集成的产品，可以添加或编辑 URL 以便于启动集成的应用程序。使用在集成配置过程中设置的 URL 填充这些字段。

- **Log Analysis URL** 用于启动 IBM Operations Analytics - Log Analysis 以便在应用程序性能仪表板的应用程序日志中进行搜索。有关更多信息，请参阅第 916 页的『搜索日志文件』。
- 对于具有子节点的代理程序，**启用子节点事件**控制子节点是否显示在“事件”选项卡中。当启用子节点事件时，将显示针对其打开了事件的节点和子节点。具体而言，如果想要在“事件”选项卡中显示日志文件监视情况，必须确保启用子节点事件。 缺省值：`False`。
- **仪表板刷新率**控制应用程序性能仪表板自动刷新的频率。您可以将该设置调整为 1 到 60 分钟内的任意值。该设置影响导航器和“状态概述”选项卡中显示的资源状态。不会影响“事件”选项卡条目。缺省值：1 分钟。

事件管理器

事件管理器使用 简单电子邮件传输协议 控制事件转发以及电子邮件通知。如果为“目标电子邮件地址”输入了值，将针对每个打开、关闭和停止事件发送电子邮件。您可以使用“**事件管理器**”字段从 Cloud APM 配置事件以自动在 IBM Control Desk 中打开凭单。有关其他配置任务，请参阅与 Control Desk 集成 第 827 页的『与 Control Desk 集成』。

如果将 SMTP 转发器配置为使用 SSL，必须将 SMTP 服务器的签名 CA 证书添加到 Cloud APM 服务器密钥库。使用 JVM 密钥工具命令将 CA 证书添加到缺省密钥库：

```
install_dir/java/jre/bin/keytool -importcert\
-noprompt \
-alias your_CA cert_alias \
-file path_to_your_CA_cert_file (*.cer)
 -keystore /install_dir/wlp/usr/servers/min/resources/security/key.jks \
-storepass ccmR0cKs! \
-storetype jks \
-trustcacerts
```

要查看样本电子邮件，请参阅第 912 页的『事件电子邮件』。

- **目标电子邮件地址**指定作为事件转发目标的电子邮件地址。请使用逗号 (,) 分隔各个地址，例如 `annette@ibm.com,jim@ibm.com,owen@ibm.com`。
- 将 **Alert Notification for ITMv6** 选项设置为值 `True` 即可启用 Alert Notification for ITMv6 事件。缺省值：`False`
- **Cloud Event Management Webhook** 是配置 IBM Cloud Application Performance Management 和 Cloud Event Management 之间的集成时在 Cloud Event Management 中生成的 Webhook URL。您必须在此处粘贴生成的 Webhook URL，从而可从 Cloud APM 转发事件。

您可使用 Alert Notification 代替事件管理器电子邮件功能，这样您就能够控制接收不同事件通知的人员以及接收通知的方式。有关更多信息，请参阅 第 826 页的『与 Alert Notification 集成』。

如果要将事件转发到 EIF（事件集成工具）接收器，那么可以定制 EIF 槽，例如将属性添加到 EIF 事件。有关更多信息，请参阅第 841 页的『定制事件以转发到 EIF 接收器』。有关将事件转发到 IBM Netcool/OMNIbus 事件管理器的信息，请参阅 第 821 页的『与 Netcool/OMNIbus 集成』。

跟踪分析服务

用于跟踪分析服务的设置。此设置仅适用于 Cloud APM, Advanced 产品以及在环境中配置事务跟踪。

- **连接池大小**是跟踪分析服务针对 “Top*N*” 查询在连接池中保留的并行 Db2 连接的数量。如果由于并行 Cloud APM 控制台用户过多而导致查询速度缓慢，请增大此值。缺省值：`10`。
- **伪节点**支持可视化未经过检测的服务。缺省值：`True`。
- **查询超时（秒）**是在每个“Top*N*”（其中，*N* 是数字，例如，“Top 5” 或 “Top 10”）查询超时前所用的秒数。如果需要，可针对具有较高工作负载和较大期望的查询时间的应用程序提高超时值。缺省值：`120` 秒。
- 正常情况下，无需更改**启用 DB2 查询重新优化**。该参数影响 Db2 查询优化器。在某些环境中，关闭优化器可提高某些事务集的性能。缺省值：`False`。

代理程序预订设施

代理程序预订设施包含代理程序 REST（表述性状态转移）接口和中央配置服务 HTTP 服务器。REST 接口由代理程序和数据收集器用于发送 Db2 服务器和阈值事件中持久存储的监视数据。中央配置服务

HTTP 服务器处理来自其配置文件的代理程序的请求，例如，阈值定义。使用这些参数来配置代理程序预订设施和 Cloud APM 服务器之间的通信。

- **错过轮询限制 (Fast Heart Beat)** 是脉动信号间隔为 60 秒或更低值的监视代理程序在被标记为脱机前无法连接的最大次数。缺省值：30 个时间间隔。
- **错过轮询限制 (Slow Heart Beat)** 是脉动信号间隔大于 60 秒的监视代理程序在被标记为脱机前无法连接的最大次数。缺省值：3 个时间间隔。
- **事务超时**是服务器等待对请求的响应的时间量（秒）。缺省值：120 秒。
- **移除脱机延迟系统**确定移除脱机受管系统的显示前等待的分钟数。在应用程序性能仪表板中，脱机受管系统由 未知状态指示符指示。即使您卸载了代理程序，受管系统也会继续显示，直到经过该延迟时间为止。 有关更多信息，请参阅第 934 页的『查看和移除脱机代理程序』。 缺省值：5760 分钟（4 天）。

阈值启用

您的每个监视代理程序都随附一组预定义的阈值，可以在代理程序中启用和启动。这些预定义的阈值分配到该代理程序的缺省系统资源组。

- **选择操作以定义预定义最佳实践阈值的策略**用于控制在缺省情况下，是启用还是禁用受管资源的预定义阈值。如果您不想运行预定义的阈值，请将此字段设置为**全部禁用**。**全部禁用**设置将移除所有预定义阈值到该系统组的分配。未分配任何组的阈值不会分发到任何受监视系统且保持停止，直至将其分发到资源组。如果您以后决定要开启预定义阈值，请将此字段设置为**全部启用**。

有关预定义阈值和定制阈值的更多信息，请参阅第 830 页的『背景信息』和第 831 页的『禁用阈值的示例』。

事件电子邮件

使用“高级配置”页面上的“事件管理器”字段以设置按电子邮件到地址列表的事件通知。

打开事件的电子邮件

当某个阈值条件为 true 时，将会打开一个事件，并且 Cloud APM 服务器发送的电子邮件包含适用于所有代理程序事件的基本属性以及从数据集第一行开始与阈值条件匹配的属性。situation_status 属性的值为 Y 表示打开事件。

关闭事件的电子邮件

当阈值条件不再为 true 时，将生成关闭事件。关闭事件的电子邮件仅包含适用于所有代理程序事件的基本属性，且 situation_status 值为 N。代理程序的属性不包含在此类电子邮件中，因为不满足阈值条件。

停止事件的电子邮件

当阈值停止时，将生成停止事件。停止事件的电子邮件仅包含适用于所有代理程序事件的基本属性，且 situation_status 值为 P。代理程序的属性不包含在此类电子邮件中，因为不满足阈值条件。

如果删除了阈值条件或者对分发到代理程序的任何阈值定义进行了更改，会停止代理程序的阈值。

以下样本显示了打开事件的电子邮件：

```
From:    noreply@apm.ibmserviceengage.com
To:         tester@us.ibm.com
Date:    10/25/2017 01:56 PM
Subject:    Linux_Disk_Space_Low on nc049048:LZ (Notification)
The text below lists the information received from the agent that triggered this event.
The IP and Agent values identify the agent that detected the event.
The Description and Severity values specify the name of the threshold definition and its
severity.
Below the Description are all of the attribute/value pairs present in the event, in their raw
form.
    Server IP  : 10.107.76.230 (SIDR26APAP1BLUE.test.ibm.com)
    Agent IP   : 9.42.49.48
    Agent      : nc049048:LZ
    Severity   : warning
    Description: Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10
                 AND FS_Type != nfs  AND FS_Type != iso9660 ]
      ITM_KLZ_Disk
      ManagedSystemGroups='*LINUX_SYSTEM'
      TenantID=F43E-D704-DADC-6270-1ED8-543E-A388-6513
      adapter_host=nc049048.tivlab.raleigh.ibm.com
      apm_hostname=SIDR26APAP1BLUE.test.ibm.com
      appl_label=A:P:S
      date=01/25/2017
```

```
      disk_free=5843
      disk_free_percent=20
      disk_name=/dev/sda2
      disk_used=22676
      disk_used_percent=80
      file_system_status=2
      file_system_status_enum=Up
      fqhostname=nc049048.test.ibm.com
      fs_type=ext4
      hostname=nc049048.test.ibm.com
      identifier=Linux_Disk_Space_Lownc049048:LZ/ITM_KLZ_Disk
      inodes_free=1721587
      inodes_free_percent=88
      inodes_used=232477
      inodes_used_percent=12
      integration_type=U
      mount_options=rw
      mount_point=/
      msg='Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10  AND FS_Type !
= nfs
     AND FS_Type != iso9660 ]'
      origin=9.42.49.48
      severity=WARNING
      situation_displayitem=/
      situation_eventdata='disk_name=/dev/
sda2;inodes_used_percent=12;mount_options=rw;fs_type=ext4;

system_name=nc049048:LZ;mount_point=/;disk_used_percent=80;disk_free=5843;file_system_status_enum=
Up;
     size=30040;disk_used=22676;inodes_used=232477;disk_free_percent=20;file_system_status=2;
     total_inodes=1954064;inodes_free=1721587;timestamp=1170125135553000;inodes_free_percent=88;~'
      situation_name=Linux_Disk_Space_Low
      situation_origin=nc049048:LZ
      situation_origin_uuid=09fb36afd6b3.22.02.09.2a.31.30.56.9d
      situation_status=Y
      situation_thrunode=nc049048:LZ
      situation_time='01/25/2017 13:55:55.000'
      situation_type=S
      size=30040
      source='ITM Agent: Private Situation'
      sub_origin=/
      sub_source=nc049048:LZ
      system_name=nc049048:LZ
      timestamp=1170125135553000
      tmz_diff=18000
      total_inodes=1954064
To unsubscribe from these emails: Log into the Cloud APM console and remove your email address
from
the list of target email addresses in the Event Manager category of the Advanced Configuration
page.
```

以下样本显示了关闭事件的电子邮件。

```
From:    noreply@apm.ibmserviceengage.com
To:      tester@us.ibm.com
Date:    01/25/2017 02:01 PM
Subject:    Linux_Disk_Space_Low on nc049048:LZ (Closed)
The text below lists the information received from the agent that triggered this event.
The IP and Agent values identify the agent that detected the event.
The Description and Severity values specify the name of the threshold definition and its
severity.
Below the Description are all of the attribute/value pairs present in the event in their raw
form.
    Server IP  : 10.107.76.230 (SIDR26APAP1BLUE-12f.test.ibm.com)
    Agent IP   : 9.42.49.48
    Agent      : nc049048:LZ
    Severity   : warning
    Description: Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10
                 AND FS_Type != nfs  AND FS_Type != iso9660 ]
      ITM_KLZ_Disk
      ManagedSystemGroups='*LINUX_SYSTEM'
      TenantID=F43E-D704-DADC-6270-1ED8-543E-A388-6513
      adapter_host=nc049048.tivlab.raleigh.ibm.com
      apm_hostname=SIDR26APAP1BLUE-12f.test.ibm.com
      appl_label=A:P:S
      date=01/25/2017
      fqhostname=nc049048.test.ibm.com
      hostname=nc049048.test.ibm.com
      identifier=Linux_Disk_Space_Lownc049048:LZ/ITM_KLZ_Disk
      integration_type=U
      msg='Linux_Disk_Space_Low[Disk_Free_Percent <= 20  AND Disk_Free_Percent > 10  AND FS_Type !
= nfs
```

```
     AND FS_Type != iso9660 ]'
      origin=9.42.49.48
      severity=WARNING
      situation_displayitem=/
      situation_eventdata=~
      situation_name=Linux_Disk_Space_Low
      situation_origin=nc049048:LZ
      situation_origin_uuid=09fb36afd6b3.22.02.09.2a.31.30.56.9d
      situation_status=N
      situation_thrunode=nc049048:LZ
      situation_time='01/25/2017 14:00:55.000'
      situation_type=S
      source='ITM Agent: Private Situation'
      sub_origin=/
      sub_source=nc049048:LZ
      tmz_diff=18000
To unsubscribe from these emails: Log into the Cloud APM console and remove your email address
from
the list of target email addresses in the Event Manager category of the Advanced Configuration
page.
```

第 10 章 使用仪表板

选择 **性能** > **应用程序性能仪表板**以获取应用程序的综合状态概述。您可以从最高级概述向下钻取至同一显示屏中的深层度量值。

使用仪表板上提供的工具来调查环境中的严重和警告条件，创建附加度量值视图以及执行一些操作，例如搜索跟踪日志和比较随着时间变化的度量值。

我的所有应用程序 - 应用程序性能仪表板

应用程序性能仪表板在**我的所有应用程序**中显示受监视域的摘要状态。对于每个用户定义的应用程序（如“库存管理”）和预定义的应用程序（如“我的组件”或“我的事务”，前提是环境中已包含）会显示一个摘要框。从该摘要框或导航器中，向下钻取到每个应用程序及其组成部分来查看详细度量值。

在选择项目时，将显示路径，您可以单击其中一个路径链接来返回到该视图。 您可以从任何 Cloud APM 控制台页面单击 **性能** > **应用程序性能仪表板** 以打开**我的所有应用程序**仪表板。通过从导航器中进行选择或者单击摘要框以向下钻取至下一级别，查看感兴趣的区域。

摘要框

我的所有应用程序针对每个定义的应用程序都包含一个摘要框。在标题栏中针对应用程序以及在摘要框中针对每个组，通过指示符显示最高状态严重性。根据定义的应用程序中包含的监视产品，提供了以下预定义的组：

- **Availability Monitoring** 没有子组
- **组件**针对支持您的应用程序的每个监视代理程序类型包含一个子组
- **事务**可以包含“**最终用户事务**”和“**合成事务**”（2017 年 8 月发行版之前的 IBM Website Monitoring on Cloud）子组

此外，每个摘要框都包含 **事件**，用于显示针对应用程序打开的最高严重性事件的严重性。您可以单击“事件”链接来调查任何打开的事件（请参阅第 939 页的『事件状态』）。

 单击摘要框标题栏以打开应用程序的“状态概述”选项卡。或者单击其中一个摘要框图标以打开“组件”组或者“用户”或“事务”子组的“状态概述”选项卡，或者打开应用程序组或子组的“事件”选项卡。

您可以折叠摘要框并通过选中或清除复选框进行过滤：

- 要仅显示摘要框标题栏以便于滚动浏览定义的应用程序，可清空**显示详细信息**复选框。
- 要过滤想要隐藏的严重性状态的摘要框，可清空计数器复选框，例如  。对于没有出现事件的严重性，对应的复选框将处于禁用状态。例如，在该图中，“严重”和“正常”的过滤器处于启用状态；“警告”和“未知”处于禁用状态，因为它们的计数为 0：  。

搜索

使用“搜索”字段来查找上一个小时内包含所输入文本的日志条目。您可以单击  来显示其他时间范围的结果。搜索文本将与导航器选择的关联日志条目进行比较，如有任何匹配，将显示在新浏览器标签页或窗口中。有关指示信息，请参阅第 916 页的『搜索日志文件』。搜索功能由 IBM Operations Analytics - Log Analysis 提供。

操作

操作菜单包含用于复制 URL，打开仪表板日志以及设置故障诊断跟踪的选项。有关更多信息，请参阅第 949 页的『复制仪表板 URL』和第 950 页的『设置跟踪』。

，使用**仪表板日志**选项可复审自上次重新启动服务器以来更新的代理程序仪表板的列表。

当选择**我的所有应用程序**主仪表板或其中一个应用程序时，“操作”菜单包含**启动至报告**以帮助您分析使用和性能趋势，但前提是可以使用基于 Cognos® 的报告并且环境包含 Tivoli Common Reporting。有关更多信息，请参阅第 951 页的『报告』。

© Copyright IBM Corp. 2014, 2019

当从导航器**组**部分或从摘要框中选择**组件**组时，**操作**菜单包含用于编辑“组件”状态概述仪表板的**编辑**选项。仅当登录用户具有“Performance Management 仪表板”的修改许可权以及“应用程序”的创建许可权时，**编辑**选项才可用。有关更多信息，请参阅第 923 页的『编辑“组件”仪表板组窗口小部件』。

帮助

打开弹出帮助可获取当前仪表板的简短描述，其中包含以下链接：**了解更多信息**打开本地 Cloud APM 帮助系统中的完整仪表板主题；**获取仪表板教程**启动 IBM Cloud APM 仪表板教程，该教程指导您使用各种功能并同时提供对仪表板元素的简单描述。

导航器

导航器以层次结构形式显示定义的应用程序、其用户、事务以及组件，反映了它们的组织方式。导航器针对每个应用程序层次结构级别包含一个部分。在导航器的每个级别处，仪表板度量值会更改为显示组件的数据。选择项目以将仪表板上下文切换为所选项。可查看内容范围取决于您的用户许可权。

每个导航器项都有 严重、警告、正常或 未知状态指示符（表示代理程序不可用）。每个导航器部分都显示与所选导航器项关联的每种严重性的事件计数。有关状态到阈值事件的映射，请参阅第 939 页的『事件状态』。

要为其他部分腾出更多空间，单击标题栏以折叠部分，再次单击可以恢复该部分。您还可以通过单击导航器边框处的 来完全隐藏导航器；单击 可以恢复显示，或者通过拖拽 边框可以调整宽度。

导航器包含三个部分：

- **应用程序**部分列出了您的域中定义或者您的用户角色允许的所有应用程序。
  - 在选择应用程序后，仪表板会更改为在**状态概述**选项卡中显示高级状态摘要，并在**事件**选项卡中显示最高严重性状态指示符。有关更多信息，请参阅第 917 页的『应用程序 - 应用程序性能仪表板』。
  - 您的域中已被监视基础结构发现的任何组件都会显示在名为“我的组件”的预定义应用程序中“”，该应用程序不能编辑或删除。
  - 如果您有管理员角色特权，那么“**应用程序**”工具栏将包含用于管理应用程序的 工具。有关指示信息，请参阅第 930 页的『管理应用程序』。
- 在选择应用程序之后，**组**部分显示了支持应用程序的组。有关更多信息，请参阅第 921 页的『组和实例 - 应用程序性能仪表板』。
- 在选择子组之后，将根据子组标题重命名**实例**部分，并将各个受管系统名称填充到该部分中。**状态概述**更改为显示所选子组的 KPI。在选择受管系统之后，将显示包含受管系统 KPI 的详细组窗口小部件。对于组件实例，还会看到**属性详细信息**选项卡，用于查看来自所选数据集属性的 KPI 表。有关更多信息，请参阅第 925 页的『查看和管理定制图表和表』。

如果在导航器中收到**网络错误**弹出消息，那么状态指示符可能更改为 正常，直至连接复原为止。届时，将重新处理任何打开事件，并且状态可能显示为正常，直至处理完成为止。

搜索日志文件

要查找用户遇到的问题（例如，变慢或故障）的根本原因，可以搜索与应用程序关联的日志数据。IBM Operations Analytics - Log Analysis 提供了搜索功能。应用程序日志数据和性能数据将集中在一起，以帮助您查找应用程序遇到的问题的根本原因并加快问题解决。

过程

执行以下步骤以找到可能与所调查问题（例如，CPU 使用率高）有关的日志条目。

1. 如果未显示应用程序性能仪表板，请从 **性能**菜单中进行选择。
2. 如果想要在应用程序内部搜索，请在“我的所有应用程序”仪表板中选择其中一个应用程序。例如，单击“我的组件”以搜索所有组件资源的日志。
3. 在 搜索框中输入要查找的日志文件文本。例如，输入 `rolled back` 以查找回滚到先前级别的 WebSphere Applications 代理程序资源。
4. 如果想要查找除了**上一小时**以外的时间范围内的数据，单击 并选择其他时段。
5. 单击。

结果
此时将在新浏览器标签页或窗口中显示当前浏览器级别上下文中包含搜索文本的所有日志条目。该浏览器窗口根据该上下文命名，例如“信用卡处理”应用程序。

下一步做什么
查看搜索结果。您可以选择其他应用程序来更改上下文的搜索结果。使用搜索字段来进一步优化结果。例如，如果搜索字段显示 `db2 AND (datasourceHostName:Pear* OR datasourceHostname:Persimmon* OR datasourceHostname:Pomegranate*)`，可以删除数据源以缩小结果范围：`db2 AND (datasourceHostname:Persimmon*)`。

有关更多信息，请参阅 IBM Knowledge Center 上的 IBMOperations Analytics Log Analysis 主题集合，或访问 IBM Operations Analytics - 开发人员社区。

应用程序 - 应用程序性能仪表板

在从导航器或从**我的所有应用程序**仪表板中的摘要框中选择应用程序之后，将通过选项卡式仪表板呈现应用程序的不同方面。 **状态概述**选项卡呈现应用程序的高级状态摘要。图表阈值和状态指示符给出总体运行状况和性能反馈。选择**事件**选项卡以查看对应用程序运行状况有影响的事件阈值。

有关导航器和条幅元素的描述，请参阅第 916 页的『导航器』、第 915 页的『搜索』、第 915 页的『操作』和第 916 页的『 帮助』。

状态概述

· 根据所选应用程序的组合，**状态概述**选项卡为评估较高级别的应用程序状态提供一个或多个透视图：

随时间变化的可用性
2017 年 8 月发行版之前的 IBM Website Monitoring on Cloud：如果应用程序包含 Synthetic Playback 代理程序（在“**事务**”导航器组和预定义的应用程序“**我的事务**”中），那么显“**一段时间内的可用性**”条形图。

每个绘制点表示一个事务样本，并以颜色指示状态 **正常**、**缓慢**或 **不可用**。

单击条形时间线上的任意位置可打开包含“**事务列表**”和“**位置列表**”表的弹出窗口。

请求和响应时间
如果应用程序包含 Response Time Monitoring Agent（“**事务**”导航器组中的“**最终用户事务**”），那么会显示“**请求和响应时间**”堆积条形图。

使用此图表来寻找性能的趋势模式。每个堆积条形图都绘制了响应时间正常并完成、响应时间缓慢并完成或未能完成的请求的百分比。重叠的折线图绘制了 5 分钟时间段内的平均响应时间。按照第 924 页的『随着时间变化调整和比较度量值』所述，使用时间选择器更改显示的时间范围。

汇总事务拓扑
当事务跟踪已启用，并且应用程序中包含下列任何代理程序或数据收集器时，将会显示“**聚集事务拓扑**”：

- DataPower 代理程序
- HTTP Server 代理程序
- IBM Integration Bus 代理程序
- J2SE 数据收集器
- JBoss 代理程序
- Liberty 数据收集器
- Microsoft .NET 代理程序
- Microsoft SQL Server 代理程序
- Node.js 数据收集器
- Response Time Monitoring 代理程序
- SAP NetWeaver Java Stack 代理程序

- Tomcat 代理程序
- WebLogic 代理程序（仅限 Linux 和 Windows）
- WebSphere Applications 代理程序
- WebSphere MQ 代理程序

除了 Response Time Monitoring 代理程序 之外，必须对所有其他代理程序手动启用事务跟踪。 对于数据收集器，事务跟踪会自动启用。 有关更多信息，请参阅第 155 页的『代理程序配置页面』。

“**汇总聚集事务**”显示与应用程序关联的资源以及它们之间的关系。页脚显示拓扑中的所选节点、资源、关系和任何过滤器的计数，以及最后一次刷新数据的时间。

如果将应用程序组件添加到业务应用程序并且该组件携带了多个应用程序的流量，那么针对这些业务应用程序显示的应用程序拓扑将包含所有应用程序的节点路径。

对于自动注入 JavaScript 的 IBM Java 应用程序堆栈，最高级别节点表示浏览器，最细粒度节点表示数据库。对于其他应用程序，最高级别节点表示应用程序，最细粒度节点表示受管系统实例。

每个节点都具有状态指示符和背景突出显示，用于显示该汇总级别的最高状态严重性。如果折叠导航器以腾出更多空间，仍可以在“**汇总事务拓扑**”中看到相同状态。节点的源环境会显示为 **云** (IBM Cloud Application Performance Management)、**ITM** (IBM Tivoli Monitoring)、**本地** (IBM Cloud Application Performance Management, Private) 、**私有云** (IBM Cloud Private) 或 **公共云** (IBM Cloud)。对于**其他**（受管资源来自其他环境），不会显示图标。

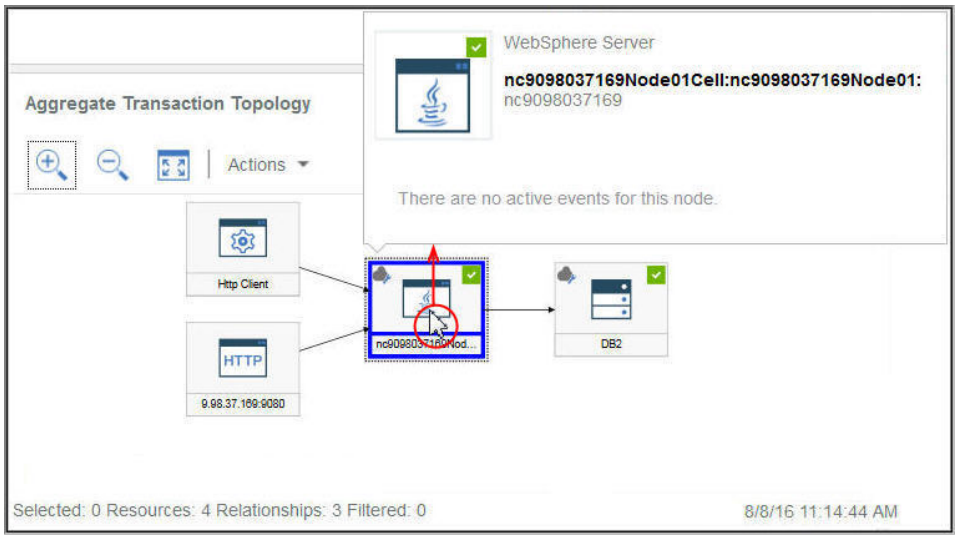
将鼠标悬停于节点上，打开快捷菜单，并选择节点以获取有关状态的更多信息，并帮助您确定问题的根本原因：

- 将鼠标悬停于节点上时，将显示弹出消息，提供 严重和 警告事件列表。
- 双击节点中的链接 URL 以打开包含组件或事务详细信息的对应仪表板。
- 右键单击节点并选择其中一个仪表板向下钻取选项：选定子组节点的**转至事务摘要页面**；实例节点的**转至组件实例页面**；或**属性**以查看资源名称、状态、受管系统名称和提供程序域（例如“Cloud”）。

使用此工具栏图标调整第 920 页的『使用“汇总事务拓扑”窗口小部件』中所述的显示和执行操作。

单击 工具以在此视图和下面所述“**当前组件状态**”之间进行切换。

没有拓扑信息的任何监视代理程序都不会显示在“汇总事务拓扑”窗口小部件中。



当前组件状态

“**当前组件状态**”堆积条形图显示应用程序中每种组件类型的严重、警告、正常和未知状态的百分比和计数。例如，假设有 5 个 Linux 系统支持所选应用程序。显示 40% 严重和 60% 正常的堆积条形图表示 2 个系统具有严重状态，3 个系统具有正常状态。

将鼠标悬停于条形段以在弹出窗口中读取状态：具有该状态的组件实例的百分比和计数。还会显示实例所在的一个或多个域，并为每个域显示状态计数：IBM Cloud、Cloud、On Premises、ITM 和其他。例如，5 个 Linux 系统中的 2 个位于 ITM 域中，3 个位于 Cloud 域中。如果其中一个严重系统位于 ITM 域中，另一个位于 Cloud 域中，那么将鼠标悬停于“严重 40%”条形段时，状态弹出窗口将显示 ITM 域中 1 个系统，Cloud 域中 1 个系统。

您可以单击某个条形来打开组件类型的状态摘要仪表板，其中针对每个受监视系统都包含一个组窗口小部件。

单击 ⊕ 工具可以切换该视图和先前描述的“**汇总事务拓扑**”。

Availability Monitoring
当已定义的应用程序只包含 Availability Monitoring 时，显示摘要仪表板，如 第 888 页的『访问 Availability Monitoring』 中所述。

- 从**组**部分中在选择子组之后，将根据子组标题重命名**实例**部分，并将各个实例名称填充到该部分中。有关导航器的组、子组和实例级别的仪表板的信息，以及有关选择受管系统后打开的**属性详细信息**选项卡的信息，请参阅第 921 页的『组和实例 - 应用程序性能仪表板』和第 925 页的『查看和管理定制图表和表』。
- 某些仪表板窗口小部件显示基于时间范围的度量值，而其他窗口小部件显示最新的度量值。如果显示了时间选择器条形，可以调整仪表板的时间范围，这样会影响其值派生自历史数据样本的任何图表或表。有关更多信息，请参阅第 924 页的『随着时间变化调整和比较度量值』。
  在查看图表时，可以单击绘制点来打开包含该绘制点值和其他相关信息的工具提示。在 Internet Explorer V11 浏览器中查看折线图后，在窗口上移动光标时，可能仍会看到显示工具提示。如果遇到此行为，可以在图表中单击几次来关闭工具提示。
- 控制台中的数据每分钟会自动刷新一次。此活动十分必要且不能暂停、停止或隐藏。
- 如果没有可用于图表或状态摘要框的数据，将显示参考消息。

事件

- 状态指示符显示在**事件**选项卡标题的旁边（例如 **14** **3**，显示以下所选导航器项的最高事件严重性的计数：：应用程序、组、子组或实例。阈值严重性按下表所示进行整合。例如，**事件** 表示最高严重性事件是“轻微”或“警告”。

“事件”选项卡	阈值严重性
严重	“致命”和“严重”
警告	“轻微”和“警告”
正常	未知

当受管环境包含 IBM Operations Analytics - Predictive Insights 且检测到异常时，将打开事件。菱形图标覆盖在状态指示符上（例如 ，用于通知您 Operations Analytics - Predictive Insights 至少检测到一个异常。例如，**事件** ，指示最高状态事件为 “警告”，并且至少打开了一个异常事件。

- 单击**事件**选项卡可以查看包含事件总数、每种严重性类型计数以及各个严重性百分比标尺的摘要。有关更多信息，请参阅第 939 页的『事件状态』。

定制视图

您创建并保存的页面将与选择的应用程序相关联。例如，Cloud APM Guided Demo 中的“库存管理”应用程序有下列监视代理程序：Linux OS、MySQL、Node.js、Hadoop 和 Ruby。您可在导航器的任何级别（从应用程序到实例）创建并保存定制页面，然后在其中创建的同一级别打开该页面。在特定级别创建的页面只能在同一级别打开。可用于窗口小部件的度量可以来自应用程序中的任何资源。使用“库存管理”示例，您可以创建一个页面，并使其包含一个来自 Ruby 代理程序的表，以及一个来自 Linux OS 代理程序的图表，等等。

- 从“**我的所有应用程序**”中选择应用程序之后，在导航器的任何级别都会提供**定制视图**选项卡。
- 打开**定制视图**选项卡之后，显示“**为定制页面选择模板**”窗口或显示缺省页面（如果已设置）。

– 在“**为定制页面选择模板**”窗口中，可以选择模板以创建页面。

– 在缺省页面上，可以单击  以创建新页面。

· 在缺省页面上，单击页面列表中的 ，然后从列表中选择其中一个保存的页面。

· 在“定制视图”选项卡中查看的选项会根据是否正在编辑或查看页面而有所变化。有关编辑页面的信息，请参阅第 942 页的『创建和管理定制页面』。有关查看页面的信息，请参阅第 947 页的『查看定制页面』。

使用“汇总事务拓扑”窗口小部件

使用“**汇总事务拓扑**”窗口小部件来查看所选应用程序中的资源层次结构。您可以调整和移动显示内容以查看每个组件的状态及其与其他组件的关系，并且可以打开节点的对应仪表板。

开始之前

在应用程序性能仪表板中选择应用程序之后，根据应用程序中包含的受监视资源，将显示包含一个或多个图表的**状态概述**选项卡。

针对以下支持事务跟踪的代理程序显示“**聚集事务拓扑**”：

· DataPower 代理程序
· HTTP Server 代理程序
· IBM Integration Bus 代理程序
· J2SE 数据收集器
· JBoss 代理程序
· Liberty 数据收集器
· Microsoft .NET 代理程序
· Microsoft SQL Server 代理程序
· Node.js 数据收集器
· Response Time Monitoring 代理程序
· SAP NetWeaver Java Stack 代理程序
· Tomcat 代理程序
· WebLogic 代理程序（仅限 Linux 和 Windows）
· WebSphere Applications 代理程序
· WebSphere MQ 代理程序

“**汇总事务拓扑**”针对支持拓扑功能的每个受监视资源显示节点对象。

过程

执行以下任意步骤以使用“**汇总事务拓扑**”窗口小部件，并打开与节点关联的仪表板：

· 要打开链接的仪表板，双击拓扑节点。 您也可以右键单击节点，并选择**转至事务摘要页面**或**转至组件实例页面**中的一个仪表板向下钻取选项，或者选择**属性**以查看有关受管系统的信息。
· 要放大拓扑显示大小，单击 **放大**。如果未选择节点，也可以单击**操作 > 放大**。
· 要缩小拓扑显示大小，单击 **缩小**。如果未选择节点，也可以单击**操作 > 缩小**。
· 要调整拓扑显示大小以适应当前窗口小部件空间，单击 **适应内容**。如果未选择节点，也可以单击**操作 > 适应内容**。
· 要过滤拓扑节点，在过滤栏中选择一个指示符。您可以关闭和打开过滤器以及选择多个过滤器。其属性与过滤器不匹配的任何节点都将灰显，与过滤器匹配的节点保持可见。
  · 正常、警告、严重或 未知（按节点状态过滤）。

- 要按环境过滤，请选择 **云** (IBM Cloud Application Performance Management)、**ITM** (IBM Tivoli Monitoring)、**本地** (IBM Cloud Application Performance Management, Private) 、**私有云** (IBM Cloud Private) 或 **公共云** (IBM Cloud)。
- 过滤以添加定制过滤器。

- 要为拓扑窗口小部件腾出更多空间，单击导航器或周围图表窗口小部件上的 **折叠部分**或者 拖动窗口小部件边框。

组和实例 - 应用程序性能仪表板

对选定的应用程序组、子组或实例使用仪表板以了解受管系统的高级状态。您可以向下钻取到包含选定实例的度量值的详细仪表板，并创建定制图表和表。

在应用程序性能仪表板中的**我的所有应用程序**下选择应用程序后，将显示“**状态概述**”和“**事件**”选项卡。

导航器**组**部分根据所定义应用程序的组成部分，列出一个或多个可能的组。

有关导航器和条幅元素的描述，请参阅第 916 页的『导航器』、第 915 页的『搜索』、第 915 页的『操作』和第 916 页的『 帮助』。

状态概述

组和子组

- 选择组或 展开组，然后选择子组以查看应用程序中每个受管系统的摘要组窗口小部件。在选择子组之后，“**状态概述**”选项卡中的摘要组窗口小部件特定于该子组。
- 根据安装的监视产品，提供了以下预定义的组：

Availability Monitoring
该组针对定制应用程序显示。 Availability Monitoring 的导航行为不同于“**组件**”和“**事务**”组。

Availability Monitoring 附加组件和仪表板在第 887 页的『Availability Monitoring』中进行了描述。

组件
该组对于所有应用程序（Response Time Monitoring Agent、Synthetic Playback 代理程序和 Availability Monitoring 除外）都显示。

“**组件**”针对支持选定应用程序的每个受监视软件组件包含一个子组。

事务
该组包含“**最终用户事务**”和“**合成事务**”（2017 年 8 月发行版之前的 IBM Website Monitoring on Cloud）子组。有关更多信息，请参阅 APM Developer Center 上代理程序度量/参考 PDF 下有关事务监视和 Synthetic Playback 代理程序 的帮助或其参考 PDF。

- 在选择**组件**或从**组**部分中选择子组之后，“**状态概述**”选项卡更改为显示摘要仪表板，其中针对每个受管资源包含一个组窗口小部件。源环境会显示为 **云** (IBM Cloud Application Performance Management)、**ITM** (IBM Tivoli Monitoring)、**本地** (IBM Cloud Application Performance Management, Private) 、**私有云** (IBM Cloud Private) 或 **公共云** (IBM Cloud)。“**实例**”部分会根据子组标题进行重命名，并以各个实例名称进行填充。

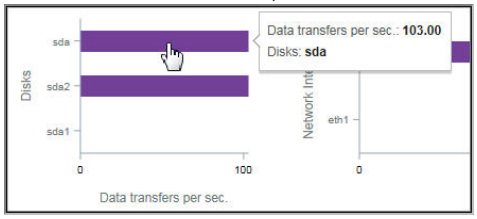


如果应用程序包含多个受管系统实例，将显示多个组窗口小部件。您可以滚动列表以查看所有实例。也可以从组件子组列表中选择受管系统类型（例如 Windows OS），以将显示内容限制为相同受管系统类型。您还可以过滤受管系统实例。

仅限 Firefox 浏览器：根据代理程序数量和带宽，在“组件”页面中向下滚动时，可能会看到弹出消息，指出用于装入资源页面的脚本可能需要较长时间才能完成。选择选项“不要再问我”以禁用该消息，并继续打开窗口小部件。或者，您可以在地址框中输入 `about:config`，搜索 **`dom.max_script_run_time`** 并增大超时值（以秒为单位）。值 0（零）禁用超时。

实例

- 单击组窗口小部件内部或从导航器中选择实例名称以打开受管资源的详细仪表板。
- 如果导航器中显示了许多实例，请使用“实例”工具栏中的搜索字段 。随着您的输入，将从显示内容中移除任何不匹配的实例。
- 要暂停应用程序性能仪表板自动刷新，请单击“实例”工具栏中的 **“暂停”**；要恢复自动刷新，请单击 **“恢复”**。
- 针对任何受管系统显示的窗口小部件和 KPI 可能取决于代理程序版本。如果受管系统上安装的代理程序版本较早，提供的信息可能无法像当前版本代理程序那么多。当没有可用数据时，将显示消息，而不是图表或表中一个或多个 KPI。原因可能只是在该时间范围内没有报告数据。可能与不支持数据集的低级代理程序有关，或者与图表或表中包含的某个属性有关。
  要查看自上次重新启动 Cloud APM 服务器以来更新的代理程序仪表板的列表，请选择**操作 > 仪表板日志**。
- 某些仪表板窗口小部件显示基于时间范围的度量值，而其他窗口小部件显示最新的度量值。如果显示了时间选择器条形，可以调整仪表板的时间范围，这样会影响其值派生自历史数据样本的任何图表或表。有关更多信息，请参阅第 924 页的『随着时间变化调整和比较度量值』。
  在查看图表时，可以单击绘制点来打开包含该绘制点值和其他相关信息的工具提示。在 Internet Explorer V11 浏览器中查看折线图后，在窗口上移动光标时，可能仍会看到显示工具提示。如果遇到此行为，可以在图表中单击几次来关闭工具提示。
- 查看缺少条形栏的图表，那么意味着此数据点的值为 0（零）。



- IBM Cloud Application Performance Management, Advanced 用户具有其他诊断仪表板，可通过单击详细信息仪表板的组窗口小部件中的**诊断**链接进行访问。

  限制: 要打开其诊断仪表板的受管系统必须位于 IBM Cloud APM 域中。如果受管系统位于 IBM Cloud 或 IBM Tivoli Monitoring 源域中，那么诊断仪表板不可用。另请参阅第 808 页的『Cloud APM 代理程序和 Tivoli Monitoring 代理程序的共存』。
- 如果您的环境包含 Synthetic Playback 代理程序，可以从**操作**菜单启动该代理程序实例的 Cloud APM 报告。

事件

- 状态指示符显示在**事件**选项卡标题的旁边（例如 **14** **3**，显示以下所选导航器项的最高事件严重性的计数：：应用程序、组、子组或实例。阈值严重性按下表所示进行整合。例如，**事件** 表示最高严重性事件是“轻微”或“警告”。

“事件”选项卡	阈值严重性
严重	“致命”和“严重”
警告	“轻微”和“警告”
正常	未知

  当受管环境包含 IBM Operations Analytics - Predictive Insights 且检测到异常时，将打开事件。菱形图标覆盖在状态指示符上（例如 ，用于通知您 Operations Analytics - Predictive Insights 至少检测到一个异常。例如，**事件** ，指示最高状态事件为 “警告”，并且至少打开了一个异常事件。
- 单击**事件**选项卡可以查看包含事件总数、每种严重性类型计数以及各个严重性百分比标尺的摘要。有关更多信息，请参阅第 939 页的『事件状态』。

定制视图

您创建并保存的页面将与选择的应用程序相关联。例如，Cloud APM Guided Demo 中的“库存管理”应用程序有下列监视代理程序：Linux OS、MySQL、Node.js、Hadoop 和 Ruby。您可在导航器的任何级别（从应用程序到实例）创建并保存定制页面，然后在其中创建的同一级别打开该页面。在特定级别创建的页面只能在同一级别打开。可用于窗口小部件的度量可以来自应用程序中的任何资源。使用“库存管理”示例，您可以创建一个页面，并使其包含一个来自 Ruby 代理程序的表，以及一个来自 Linux OS 代理程序的图表，等等。

- 从“**我的所有应用程序**”中选择应用程序之后，在导航器的任何级别都会提供**定制视图**选项卡。
- 打开**定制视图**选项卡之后，显示“**为定制页面选择模板**”窗口或显示缺省页面（如果已设置）。
  - 在“**为定制页面选择模板**”窗口中，可以选择模板以创建页面。
  - 在缺省页面上，可以单击 以创建新页面。
- 在缺省页面上，单击页面列表中的 ，然后从列表中选择其中一个保存的页面。
- 在“定制视图”选项卡中查看的选项会根据是否正在编辑或查看页面而有所变化。有关编辑页面的信息，请参阅第 942 页的『创建和管理定制页面』。有关查看页面的信息，请参阅第 947 页的『查看定制页面』。

属性详细信息

- 从导航器“**实例**”部分（已重命名为选定子组名称）中选择组件实例，或者单击摘要组窗口小部件内部时，将显示**属性详细信息**选项卡。
- 如果已针对代理程序保存了图表或表页面，那么将显示最近打开的页面，其中包含来自选定组件实例的度量值。单击标题 以从**我的页面** 或**共享页面** 中选择其他保存的页面。
- 您可以编辑图表或表，然后单击**预览结果**以显示包含选定属性的图表或表。有关更多选项，请参阅第 925 页的『创建定制图表或表页面』。

编辑“组件”仪表板组窗口小部件

您可以编辑“**组件**”仪表板（从导航器“**组**”部分中选择）中显示的组窗口小部件阈值。还可以控制显示哪些组窗口小部件及其位置，决定在确定组件状态时是否应包含窗口小部件阈值。

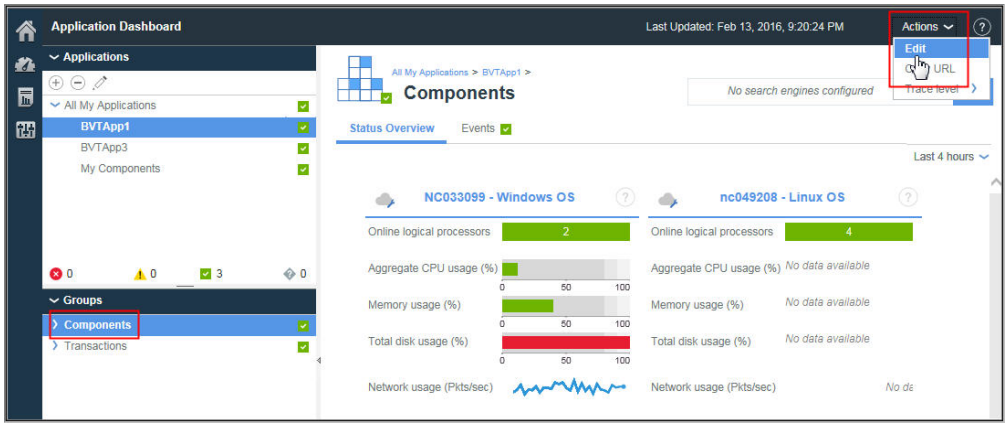
关于此任务

此任务包括编辑所定义应用程序的“组件”组仪表板及其组成部分摘要组窗口小部件。“组件”组编辑器不可用于“**我的组件**”预定义应用程序。有关定义的应用程序的更多信息，请参阅第 930 页的『管理应用程序』。

用户标识还必须具有对应用程序性能仪表板的修改许可权以及“应用程序”的创建许可权。有关更多信息，请参阅第 851 页的『角色和许可权』。

过程

1. 在从 **性能**菜单中打开应用程序性能仪表板之后，从“**我的所有应用程序**”仪表板中选择要编辑其摘要组窗口小部件的应用程序。
2. 在导航器的“**组**”部分中，单击**组件**以打开一个仪表板，其中显示该应用程序中所有组件的组窗口小部件。
3. 单击**操作 > 编辑**以打开“**组件**”组中组窗口小部件的编辑器。
   编辑选项仅在打开“**组件**”仪表板时显示。



4. 对组窗口小部件进行以下任意更改：
   - 要从视图中移除组窗口小部件，单击 。
   - 要修改窗口小部件的摘要阈值，单击 “**设置**”，选择“**阈值**”选项卡，并更改严重、警告或正常严重性的阈值。在编辑组窗口小部件的阈值之后，单击**完成**。
   - 要添加窗口小部件，单击 ，单击应用程序图标直到显示您所需的应用程序为止，然后单击组窗口小部件内部以将其选中，最后单击**添加**。
   - 要调整窗口小部件大小，拖动手柄图标 。调整窗口小部件大小不会更改文本大小或窗口小部件高度。
   - 要移动窗口小部件，可将其拖至新位置。
5. 要保存更改并关闭编辑器，单击**保存**；或者要废弃更改，单击**取消**。

结果

此时将显示具有新设置的选定应用程序的“**组件**”仪表板。

下一步做什么

有关在导航器中选择组或子组时的仪表板的更多信息，请参阅第 921 页的『组和实例 - 应用程序性能仪表板』；有关受监视组件仪表板的更多信息，请单击“**应用程序仪表板**”条幅中的 按钮。

随着时间变化调整和比较度量值

某些仪表板图表显示基于时间范围的度量值，而其他窗口小部件仅显示最新的度量值。 当在受管系统实例的“**状态概述**”选项卡中显示时间选择器时，可以调整从历史数据样本派生其值的图表的时间范围。对于收集多天数据且以折线图表示的属性，可以将今天的值与前一天进行比较。

开始之前

如果要与前一天的时间范围进行比较，那么可以回溯到多久以前取决于 Cloud APM 服务器已保存的天数以及页面上显示的数据的类型。。对于 Cloud APM 付费订户，对于大部分资源监视数据集，数据样本存储 8 天时间。准确数字发布在代理程序或数据收集器属性帮助和参考 PDF 中（请参阅 第 35 页的『第 2 章 PDF 文档』）。对于 Cloud APM 试用版订户，资源监视数据样本存储 2 天时间。只能显示 Response Time

Monitoring Agent 或中间件代理程序中过去 24 小时（或者，在某些情况下为过去 4 小时）的事务跟踪数据，并且无法更改其保留期。

过程

执行以下步骤来调整在受管资源实例的折线图中显示的时间范围或将这些值与前一天的同一时间范围进行比较。

1. 如果未显示应用程序性能仪表板，请从 **性能**菜单中进行选择。
2. 导航到显示历史折线图的实例的仪表板页面并单击**过去 4 小时**时间选择器。
3. 选择以下一个或多个选项：
   - 要更改显示的时间范围，选择**过去 4 小时**、**过去 12 小时**或**过去 1 天**。
   - 要将折线图中显示的时间范围与另一个日期的度量值进行比较，请选择**比较对象**，然后选择弹出日历中所显示天数内的一个更早的可用日期（将不可用的日期划掉）。
   - 要将时间范围应用于受监视环境中定义的所有应用程序的仪表板，选择**所有应用程序**。否则，保留设置**仅当前应用程序**以将时间范围仅应用于当前应用程序（例如“我的组件”）。**比较对象**选择仅对当前页面有效。

结果

- 如果要查看历史数据而不进行比较，那么当前应用程序（或所有应用程序）中的所有仪表板都会受此更改的影响。
- 如果要查看比较，那么仅影响当前页面中的折线图。将针对每个 KPI 绘制一个线条以显示所选日期的度量值。某些折线图不能用于比较，通过图表上的水位标记“无可用比较”进行指示。对于收集数据未达到指定日期的较新受管资源，可能会出现此情况。请尝试选择较近的日期进行比较。
- 未收集其历史数据的任何窗口小部件会继续显示最新值。
- 数据点沿着所选时间范围的完整图表长度分布。时间戳记显示在轴标上，以最早时间戳记作为起点，以最新时间戳记作为终点。检查最旧数据样本以确认显示的是部分历史数据还是完整的历史数据。
- 发送到图表和表的数据将根据 GMT（格林威治标准时间）进行规范化。**时间戳记**轴基于浏览器时区打印时间戳记。如果您的时区使用标准时间和夏令时，那么过渡时期显示的时间戳记会错位一小时。例如，假设您在西班牙查看折线图，时间从 2:00 AM 标准时间更改为 3:00 AM 夏令时。数据 GMT 和时间戳记本地时间的差异导致时间戳记产生从 2:00 AM 到 3:00 AM 的一个小时的时差。如果从 3:00 AM 夏令时到 2:00 AM 标准时间的切换期间内，您在新西栏查看同一个图表，那么从 2:00 AM 到 3:00 AM 的时间戳记将会重复。

查看和管理定制图表和表

应用程序性能仪表板提供了受管系统关键性能指标的预定义仪表板。在查看组件实例的仪表板时，使用**属性详细信息**选项卡来查看保存的图表或表页面以及创建和管理其他页面。

例如，在摘要仪表板中观察到临界指标，并向下钻取到出现此条件的实例。在此处，您可以添加绘制繁忙 CPU 速率的图表，从而查看随着时间所发生的事件。您可以查看有关选定组件实例的任何可用属性的详细信息，以及保存代理程序的定制图表或表，以在打开受管系统实例时进行显示。

一部分代理程序数据集和属性可以在定制图表和表中使用。这些属性是可在仪表板中显示的最有用的属性。所有属性都可在定制阈值中使用（请参阅第 838 页的『阈值管理器』）。

为了提高性能和减少冗余，代理程序限制了对“属性详细信息”视图中特定数据集显示的行数。代理程序帮助和参考 PDF 中的数据集描述指示缺省数据样本是否限制发送到 Cloud APM 服务器的行数。

对于有视力障碍的用户，能够创建历史表以替代辅助技术（如屏幕朗读器软件）不能解释的折线图。因此，**属性详细信息**选项卡可供 Response Time Monitoring 代理程序 和 Synthetic Playback 代理程序事务实例用于创建历史表。有关更多信息，请参阅第 928 页的『使用键盘控件创建定制表的示例』。

创建定制图表或表页面

在查看组件实例的应用程序性能仪表板时，您可以选择**属性详细信息**选项卡以查看保存的图表或表页面以及创建和管理其他页面。

关于此任务

从应用程序性能仪表板主页向下钻取到受管资源的实例后，会将**属性详细信息**选项卡添加到仪表板页面上的**状态概述**和**事件**选项卡中。

这些指示信息用于创建组件实例的定制图表和表。您可以遵循 Response Time Monitoring 代理程序 和 Synthetic Playback 代理程序事务实例的步骤，但是有以下限制：仅限于历史表（无图表）；不能过滤**数据集**或**属性**列表；选择所有属性（不能选择或取消选择单个属性）；不能保存页面；以及时间选择器**先前**选项不可用于 Synthetic Playback 代理程序。

过程

完成以下步骤以根据对所选组件实例提供的任何数据集构造图表或表：

1. 在从 **性能**菜单打开应用程序性能仪表板之后，向下钻取到受管资源的实例。
   所选系统将在导航器的“**实例**”部分中突出显示，它将根据组件类型命名，例如“**Ruby 应用程序**”。
2. 单击**属性详细信息**选项卡。
   如果没有为此监视代理程序类型保存图表或表页面，将选择**实时** **表**。“实时”适用于返回多行的数据集，并且仅可用于表。
3. 如果显示了保存的图表或表页面，单击 **新建**。
4. 在“标题”字段中输入图表或表页面的名称。
   请勿在标题中使用以下任何字符：！ " % & ' * ? < > } { \。
5. 如果希望查看随着时间获取的数据样本，请将类型更改为**历史**。
   图表选项已启用。
6. 如果选择了 **历史**，并希望以图表而非表来呈现，请单击 **图表**。
7. 从**数据集**列表中，选择想要查看的属性类型的单选按钮。如果列表较长，使用 过滤器框并输入数据集名称中肯定包含的文本来缩短列表。
   例如，针对 Linux OS 代理程序输入“config”会将数据集过滤为仅显示 **Linux_CPU_Config** 和 **Linux_OS_Config** 数据集。
8. 要在图表或表中包含某个属性，请选中**属性**列表中对应名称旁边的复选框；要包含所有属性，请选中列表开头的复选框。在 过滤器框中输入文本以找到特定属性，例如“percent” “”。
   例如在 KLZ_VM_Stats 数据集上输入“percent”会过滤该列表以显示“**可用虚拟存储器（百分比）**”和另 5 个“Percent”属性。
   图表可以仅绘制数字值；任何文本或时间属性都会禁用。
9. 单击**预览结果**以生成包含所选数据集的页面，一个表列或图表行分组表示一个属性，一个行或绘制点表示一个数据样本。
   您还会看到表示所有值的汇总的一行或一条绘制线。
10. 要隐藏图表线条或表行，执行以下步骤之一：

选项	描述
图表	清空图例中对应名称旁边的复选框来隐藏度量值（图表中的线条）。选择复选框可显示度量值。
表	在 过滤器框中输入作为过滤依据的值可以减少显示的行数。也可以按第 928 页的『定义表过滤器』所述创建高级过滤器。

11. 要调整时间范围，使用时间选择器：

选项	描述
实时	（仅限表）刷新并仅显示最新数据采样。
2 小时	针对过去 2 小时获取的每个数据样本按间隔绘制一个点或添加一行。
4 小时	针对过去 4 小时获取的每个数据样本按间隔绘制一个点或添加一行。
12 小时	针对过去 12 小时获取的每个数据样本按间隔绘制一个点或添加一行。
24 小时	针对过去 24 小时获取的每个数据样本按间隔绘制一个点或添加一行。

选项	描述
先前	选项子菜单，包含来自同一时间段**昨天**、**2 天前**的数据，或者来自最远不超过 **1 周前**的任何一天数据。例如，当前是 8 月 22 日下午 2:10，您将图表或表设置为显示过去 4 小时。通过选中**先前** > **1 周前**，您可以看到从今天上午 10:10 - 下午 2:10 的数据点和从 8 月 15 日上午 10:10 - 下午 2:10 的数据点。 选中先前某一天后，时间选择器会显示一个星号 (*)（如 Last 4 hours* ），选中的那一天的数据的表将重新生成：

- 只能查看当天以及过去一周内的数据，即使您已将最大历史数据保留期增加到超过 8 天也是如此。
- 历史图表会针对选择的时间范围（例如，过去 4 小时），从最早数据样本绘制到最新数据样本。选中过去的某一天时，您会看到该日所选时间范围内的数据，以及当天该时间范围内的数据。在该过往日期与当天之间的任何一天都会显示含时间戳记但不含任何数据样本的绘制点。
- 按时间降序绘制历史表。为 Response Time Monitoring 代理程序 选择前一天后，针对列标题中包含“前一天”的前一天复制每列“”。
- 无论选择的时间范围为何，最多只能显示 11,000 行。例如，如果您已选择显示 12 个小时的数据，而数据集在 2 小时内发送了 7,000 行，那么将返回不到 3 个小时的历史数据，并且会显示最早的数据样本。

12. 要保存或对图表/表进行其他更改，选择以下选项之一：

选项	描述
编辑	使您返回到编辑器以进行以下任何更改： · 编辑标题 · 更改为**实时**或**历史**数据样本 · 更改为 **图表**或 **表** · 选择其他**数据集**和/或**属性**
新建	放弃当前视图的任何未保存的更改，并返回至选择页面以创建新图表或表。
取消	取消当前图表或表页面的编辑会话。
删除	删除页面。**删除**仅在保存页面且取消了任何当前编辑之后可用。
为我保存	保存图表或表页面以仅供您的用户标识查看。其他用户不能查看该保存的页面。
保存以共享	保存图表或表页面以供登录到 Cloud APM 控制台的任何用户标识查看

您保存的视图在标题旁包含一个打开的锁 。其他用户保存的您无权编辑的视图具有关闭的锁图标。

结果

保存定制图表或表之后，会将其添加到保存的页面列表中。下次选择相同数据源类型（例如，WebSphere 应用程序）的实例，并选择**属性详细信息**选项卡时，将显示最近打开的保存的页面。单击标题 以从**我的页面** > 或**共享页面** > 中选择其他保存的页面。

下一步做什么

重复上述过程以创建和管理其他图表或表页面。

有关使用键盘控制代替鼠标点击来创建表的指南，请参阅第 928 页的『使用键盘控件创建定制表的示例』。

使用键盘控件创建定制表的示例

由于辅助技术（如屏幕朗读器软件）不能解释历史折线图，有视力障碍的用户可以使用**属性详细信息**仪表板选项卡来创建可访问的历史表进行替代。

关于此任务

以下示例阐明了使用键盘控件为 Synthetic Playback 代理程序报告的事务创建历史表。有关代理程序的更多信息，请参阅第 872 页的『使用 Web 站点监视管理合成事务和事件』。

在按下 Tab 键后，焦点按从左到右和从上到下的顺序移动到应用程序窗口的下一个字段或下一个部分。通过使用这些步骤，您可以将“**我的事务**”替换为包含代理程序的应用程序，从而为 Response Time Monitoring Agent 生成事务表，或者将“**我的事务**”替换为其他应用程序并选择“**组件**”组，从而为组件实例生成表。

过程

按照以下步骤，可使用键盘快捷键在**属性详细信息**选项卡中通过 Synthetic Playback 代理程序创建事务表。

1. 登录 IBM Cloud Application Performance Management。

   焦点位于导航栏上。
2. 要打开应用程序性能仪表板，请按向下箭头以将焦点移到**性能**菜单，按 Enter 键进行选择，然后按向下箭头将焦点移到**应用程序性能仪表板**选项并再次按 Enter 键。
3. 要打开“**合成事务**”仪表板页面，请重复按 Tab 键（大约 7 次），直到焦点移到导航器上，按向下箭头以将焦点移到“**我的事务**”预定义应用程序，然后按 Enter 键。
4. 要打开“**事务详细信息**”仪表板页面，请按 Tab 键（大约 10 次），直到焦点移到“**合成事务**”事务实例上的导航器“**实例**”部分，然后按 Enter 键。

   将在仪表板上显示**属性详细信息**选项卡。
5. 要打开**属性详细信息**选项卡，请按 Tab 键（大约 6 次），直到焦点位于**状态概述**选项卡上，然后按向右箭头，直到焦点位于**属性详细信息**选项卡上。

   将选择**历史** 表和“**随时间变化的事务可用性**”数据集。
6. 要生成表，请按 Tab 键（大约 18 次），直到焦点位于**预览结果**按钮上，然后按 Enter 键。

结果

“**随时间变化的事务可用性**”属性显示在表中，每列对应于过去 4 小时内的每个属性，每行对应于每个数据样本。

下一步做什么

- 要减少显示的行数，您可以按 Tab 键以将焦点移到 “**过滤器**”文本框，然后输入过滤所要依据的部分或全部文本或时间戳记值。
- 要更改时间范围，请将焦点移到时间选择器**最近 4 个小时** 下拉菜单并选择其他选项。有关更多信息，请参阅第 925 页的『创建定制图表或表页面』中的步骤 第 926 页的『11』。
- 要生成包含“**事务响应时间**”数据集的表，请按 Tab 键（大约 2 次）以将焦点移到 **新建**工具，然后按 Enter 键。将显示选择面板。寻找“**事务响应时间**”数据集和**预览结果**按钮。

定义表过滤器

您可以限制在仪表板**属性详细信息**选项卡中查看的表中的行，以仅显示特定类型的行或者具有特定文本或时间戳记属性值的行。虽然数字值（例如，百分比）不可用于过滤，但某些数字属性值会转换为表的显示值并作为文本来处理。您可以应用快速过滤，或打开编辑器以编写高级过滤。

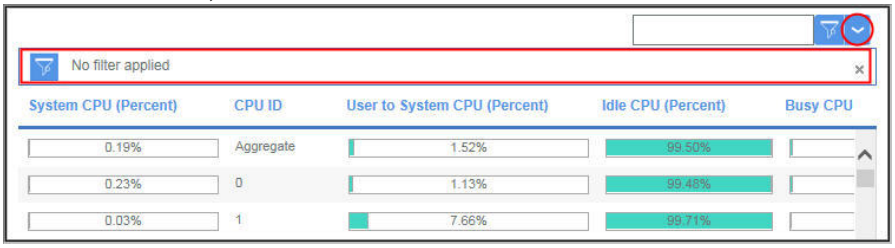
过程

完成以下步骤以按文本或时间戳记属性值过滤定制表。虽然数字值不可用于过滤，例如百分比，但是一些数字属性值已转化为表的显示值并被视为文本。

1. 在从 **性能**菜单打开应用程序性能仪表板之后，向下钻取到受管资源的实例。
2. 单击**属性详细信息**选项卡。

   最近保存的页面将显示，或者如果未保存页面，那么将显示**数据集**和**属性**选择列表。

3. 如果显示保存的表页面，那么请继续执行步骤 第 929 页的『5』，从下拉菜单 中选择其他已保存的表页面，或者单击 **添加** 以创建新的表。
4. 如果要创建新表或编辑已保存的表，请选择要使用的**数据集**和**属性**，然后单击**预览结果**。
5. 对于快速过滤，单击 **“过滤器”**文本框，然后输入部分或完整的过滤文本。
   随着您的输入，将从表中移除不包含所输入内容的任何行。要移除快速过滤，请删除值或者单击 “x”。
6. 对于高级过滤，单击 下拉菜单并选择**构建过滤器**，或者单击过滤器栏中的任意位置。
   此时将打开**“构建过滤器”**窗口，其中包含所定义的所有规则。



7. 要定义规则，填写以下字段：
   a) 将列设置保留为“任意列”，或者从列表中选择要作为过滤依据的属性。
   b) 将条件保留为“包含”，或者从列表中选择其他运算符，然后在文本框中输入 文本或时间戳记过滤值：

条件	**满足以下条件时，行包含在表中**
包含	在单元格中的某个位置找到过滤器值。
等于	单元格值与过滤器值完全匹配，包括字母大小写。
开头为	单元格值的开头字符与过滤器值相同。
结尾为	单元格值的结尾字符与过滤器值相同。
不等于	单元格值不与过滤器值完全匹配。
不包含	单元格值不包含与过滤器值相同的文本或数字。
开头不是	单元格值的开头字符与过滤器值不相同。
结尾不是	单元格值的结尾字符与过滤器值不相同。
为空	单元格未显示数据。

   c) 完成规则后，单击**过滤**以查看结果，单击 **添加过滤规则**以添加其他规则，或者转至下一步。
8. 如果过滤器包含多条规则，请执行以下任意步骤：
   - **匹配**初始设置为**所有规则**，意味着仅当行中的数据符合过滤器中的所有规则时才会显示该行。如果在任何规则后没有 文本或时间戳记值，那么将排除此行。 如果您有多条规则，并希望包含符合任意规则的行，请将设置更改为**任意规则**。
   - 要编辑规则，可以更改任意字段值。
   - 要删除规则，将其选中，然后单击 **移除规则**。
9. 完成规则定义后，单击**过滤**以关闭对话框并应用过滤器。

结果

不符合规则条件的组将从显示中移除，过滤栏将报告项数，例如，“显示 480 项，共 1200 项”。

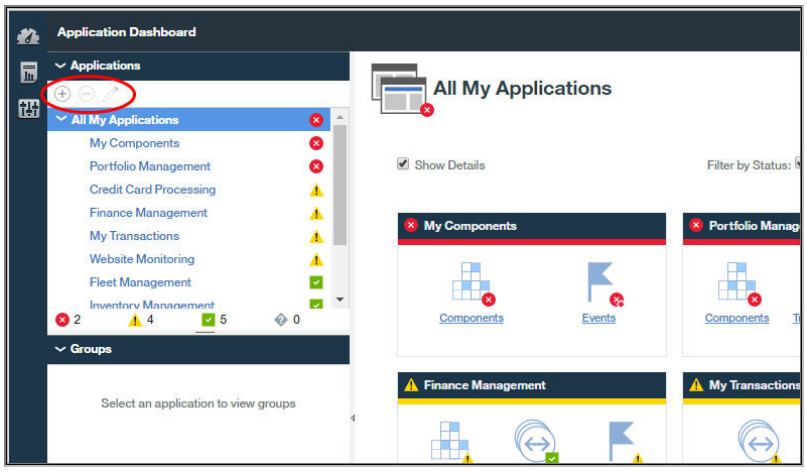
下一步做什么

- 将鼠标指针悬浮于过滤栏上以打开包含过滤条件的弹出窗口。您可以删除规则（单击 ×）或单击窗口内部以编辑过滤条件。
- 单击过滤栏中的**清除过滤器**或**“构建过滤器”**窗口中的**清除**以移除过滤器并显示所有行。

管理应用程序

使用应用程序性能仪表板中提供的工具将受管资源组成应用程序。

导航器**应用程序**工具打开用于创建或编辑应用程序以及应用可用受管资源的“应用程序”编辑器。



我的组件应用程序是预定义应用程序，包含 Cloud APM 服务器发现的受管系统。无法编辑或删除 **My Components**。

有关添加应用程序的视频演示，请观看 Application Performance Management - 定义应用程序。

有关创建应用程序以监视 IBM Java 应用程序堆栈的场景，请参阅第 75 页的『向应用程序性能仪表板添加 Web 应用程序 』和第 76 页的『将 IBM Java 应用程序堆栈与 Web 应用程序关联 』。

限制: 您必须具有“应用程序”的修改许可权才能使用“添加应用程序”工具。您必须具有“应用程序”或特定应用程序的修改许可权才能使用“移除”和“编辑”工具。有关更多信息，请参阅第 857 页的『处理角色、用户和许可权』。

添加应用程序

使用“应用程序”编辑器可创建新的应用程序并应用可用的受管资源，或从任何已发现的应用程序中选择一个。

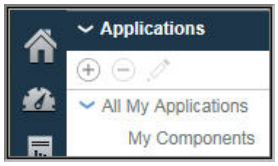
开始之前

您必须具有“应用程序”的修改许可权才能使用“添加应用程序”工具。有关更多信息，请参阅第 857 页的『处理角色、用户和许可权』。

过程

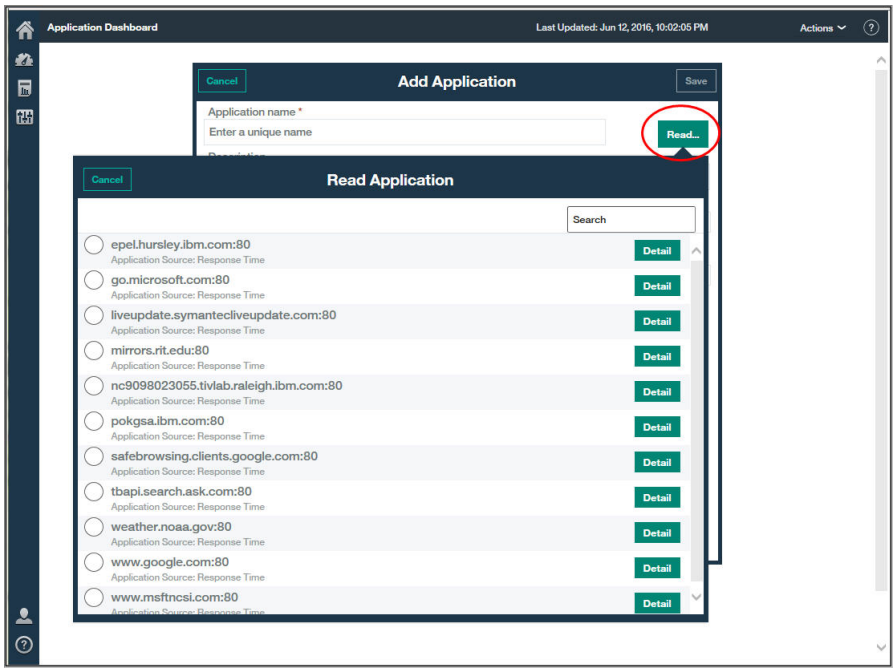
在 Cloud APM 控制台中完成以下步骤以将应用程序添加到应用程序性能仪表板中。

1. 如果应用程序性能仪表板未显示，请从 **性能**菜单中选择，或者如果在其他控制台页面上，请单击**主页**链接。
2. 在导航器的“**应用程序**”部分中，单击 ⊕。这将显示“**添加应用程序**”窗口。

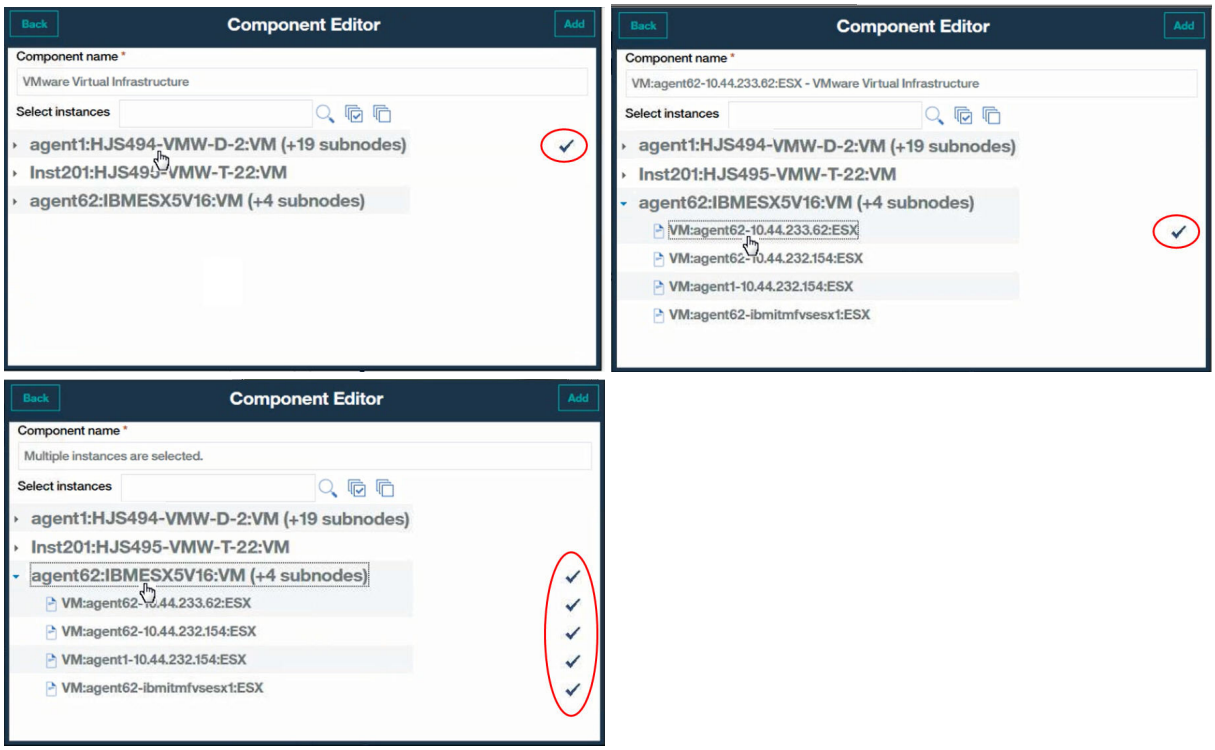


3. 在**应用程序名称**字段中输入应用程序的名称，（可选）在**描述**字段中输入描述。
   请勿在名称或描述中使用 ！ " % & ' * ? < > } { \ 符号。
   您可以看到应用程序名称的一些示例，例如，Guided Demo 中的“财务管理”和“信用卡处理” “” “” 。

4. 单击**读取**以打开包含任何发现的应用程序列表的“**读取应用程序**”窗口，然后执行以下一个或多个步骤：
   - 单击**详细信息**以查看应用程序的组件。
   - 选择要使用的应用程序，然后单击**保存**。“**读取应用程序**”窗口将关闭，并且**应用程序读取位置**字段中显示源存储库，**应用程序组件**中列出组件。
   - 单击**取消**以关闭窗口而不进行选择。



5. 在**模板**字段中，保留“**定制应用程序**”模板，或者使用 › 按钮选择其他模板，然后单击**保存**。
   任何关联的组件类型和实例将在**应用程序组件**列表中显示。
6. 单击 ⊕ **添加组件**，并在打开的“**选择组件**”窗口中，选择列表中的一个组件。
   这将显示“**组件编辑器**”。
7. 要找到并选择应用程序的代理程序节点和/或子节点实例，请执行以下一个或多个步骤：
   - 单击实例以进行选择。
   - 对于具有子节点的代理程序节点，在折叠树时单击名称以只选择节点，展开节点树（单击 ▸）并单击节点以选择节点和所有子节点，或者展开节点树并单击实例以选择各个子节点。
   - 使用 🔍 工具栏搜索包含搜索文本框中的文本的实例，以选择所有实例或清除所有实例。
   - 如果要更改导航器中的显示名称，请编辑组件名称。



V6-V7 如果添加 Tivoli Monitoring 代理程序实例但在可用实例列表中看不到该实例，请检查与 Hybrid Gateway 关联的 Tivoli Enterprise Portal Server 是否处于受支持的版本（请参阅 Hybrid Gateway supported agents (APM Developer Center)）。

8. 单击**添加**以将所选代理程序节点和子节点添加到应用程序，然后单击**返回**。
   应用程序组件列表将更新为新的组件名称。
9. 选择其他要添加实例的组件，然后重复 第 931 页的『6』、第 931 页的『7』 和 第 932 页的『8』，或单击**关闭**。
10. 如果其他实例与**应用程序组件**列表中的组件相关，那么将显示一个按钮以显示相关实例数，然后您可以执行以下步骤：
    a) 单击 按钮以在“**更新的详细信息**”窗口中查看相关实例。
    对于每种更新类型显示一栏，其中包括实例名称。例如，如果移除了其中一个组件，它会显示在“**已删除**”组件栏下方。
    b) 选择一个或多个实例，并单击**保存**以更新“应用程序资源”列表。
11. 在定义应用程序完成后，单击**保存**以关闭应用程序编辑器并保存更改，或单击**取消**以关闭应用程序编辑器并撤销更改。

结果

在保存更改后，您的应用程序更新将由 Cloud APM 服务器完成。可能需要几分钟才会在仪表板中显示更改。(如果您的更改要过很长时间才会显示，请尝试清除浏览器高速缓存。 新应用程序将在应用程序性能仪表板和导航器“**应用程序**”部分中显示。在选择应用程序时，组件将显示在“**组**”部分中。

编辑应用程序

使用“应用程序”编辑器可修改已定义的应用程序，以添加或移除作为应用程序组件的受管资源。

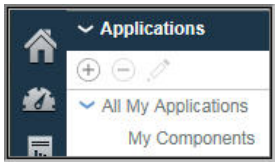
开始之前

您必须具有“应用程序”或特定应用程序的修改许可权才能使用“编辑”工具。有关更多信息，请参阅第 851 页的『角色和许可权』.

过程

在 Cloud APM 控制台中完成以下步骤以编辑应用程序。

1. 如果应用程序性能仪表板未显示，请从 **性能**菜单中选择，或者如果在其他控制台页面上，请单击**主页**链接。
2. 选择要从导航器中的“**我的所有应用程序**”列表编辑的应用程序并单击 。



   这将显示“**编辑应用程序**”窗口。
3. 可选： 编辑**应用程序名称**或**描述**。

   请勿在名称或描述中使用 ! " % & ' * ? < > } { \ 符号。
   如果应用程序“查看”或“修改”许可权适用于单个应用程序而不是所有应用程序，那么在应用程序重命名后，您可能无法在仪表板中查看应用程序或修改应用程序。此限制的原因是重命名的应用程序被视为新应用程序。您的角色管理员或监视管理员必须为您提供对已重命名的应用程序的“查看”或“修改”许可权。
4. 要向应用程序添加组件和实例，请执行以下步骤。
   a) 单击 ，然后在打开的窗口中从列表中选择一个组件。
      这将显示“**组件编辑器**”。
   b) 选择应用程序的代理程序节点和/或子节点实例：
      - 单击实例以进行选择。
      - 对于具有子节点的节点，在折叠树时单击名称以选择节点，展开节点树（单击 ）并单击节点以选择节点和所有子节点，或者展开节点树并单击实例以选择各个子节点。
      - 使用 工具栏搜索包含搜索文本框中的文本的实例，以选择所有实例或清除所有实例。
      - 如果要更改导航器中的显示名称，请编辑组件名称。
   c) 单击**添加**以添加实例，然后单击**返回**。
      应用程序组件列表将更新为新的组件名称。
   d) 您可以选择其他要添加实例的组件，或者单击**关闭**。

   应用程序组件列表将更新为新的组件名称。名称之后括号内的数字指示与组件相关联的实例数量。
5. 要编辑组件名称或者更改与之关联的实例，请从**应用程序组件**列表中选择该组件，然后单击 ：
   a) 要将不同的实例与组件相关联，请搜索并选择想要的实例。
   b) 要更改在导航器中用作此应用程序显示名称的组件名称，请编辑**组件名称**字段。
   c) 单击**保存**。

   “应用程序组件”列表将更新为所作更改。
6. 要从应用程序中移除组件或实例，请将其选中并单击 。单击**确定**以确认要移除。
7. 如果其他实例与**应用程序组件**列表中的组件相关，那么将显示一个按钮以显示相关实例数，然后您可以执行以下步骤：
   a) 单击 按钮以在“**更新的详细信息**”窗口中查看相关实例。
      对于每种更新类型显示一栏，其中包括实例名称。例如，如果移除了其中一个组件，它会显示在“**已删除**”组件栏下方。
   b) 选择一个或多个实例，并单击**保存**以更新“应用程序资源”列表。
8. 在编辑应用程序完成后，单击**保存**以关闭应用程序编辑器并保存更改或单击**取消**以关闭应用程序编辑器并撤销更改。

结果

在保存更改后，您的应用程序更新将由 Cloud APM 服务器完成。可能需要几分钟才会在仪表板中显示更改。

相关参考

第 851 页的『角色和许可权』

删除应用程序

当不再需要为在应用程序性能仪表板中显示而定义的应用程序时，您可以将其删除。删除应用程序不会卸载支持组件；只会删除它们所在的应用程序。相同的组件可用于添加到其他应用程序，不会从它们所属的其他应用程序中移除。

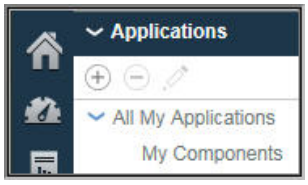
开始之前

您必须具有“应用程序”或特定应用程序的修改许可权才能使用“移除”工具。有关更多信息，请参阅第 857 页的『处理角色、用户和许可权』。

过程

请完成以下步骤以从应用程序性能仪表板中移除应用程序。

1. 如果应用程序性能仪表板未显示，请从 **性能**菜单中选择，或者如果在其他控制台页面上，请单击**主页**链接。



2. 在导航器的“**应用程序**”部分中，选择要从“**我的所有应用程序**”列表中删除的应用程序，然后单击 。

   您将收到一条消息，要求您进行确认。

3. 单击**是**以确认要删除应用程序；或者如果不确定，请单击**否**。

结果

在单击**是**之后，从应用程序性能仪表板中删除应用程序。

下一步做什么

针对希望删除的任何其他应用程序，重复此步骤。

查看和移除脱机代理程序

代理程序脱机四天后，从 Cloud APM 控制台中将其移除。复审如何指示脱机代理程序以及对资源组、拓扑视图和其他功能的作用。在四天结束前使用应用程序编辑器从仪表板中移除受管系统。。

关于此任务

在要管理的系统上安装代理程序后，它们将连接到 Cloud APM 服务器并将数据样本发送到应用程序性能仪表板以进行演示和阈值求值。如果代理程序脱机，在导航器和仪表板中显示 状态指示符。服务器等待特定时间间隔后，在显示代理程序不可用前没收到来自代理程序的响应。请参阅第 935 页的『脱机代理程序的示例』。

四天后，从用户界面中移除脱机代理程序，但是存在以下异常：如果代理程序支持事务跟踪，那么脱机代理程序继续在“汇总事务拓扑”和“事务实例拓扑”视图中显示。

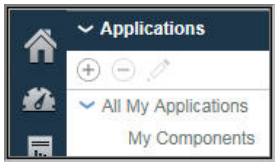
您可以从任何已定义的应用程序中移除脱机代理程序，将在四天等待周期结束前从 Cloud APM 控制台中将其移除。

过程

执行以下步骤从已定义的应用程序中移除脱机代理程序：

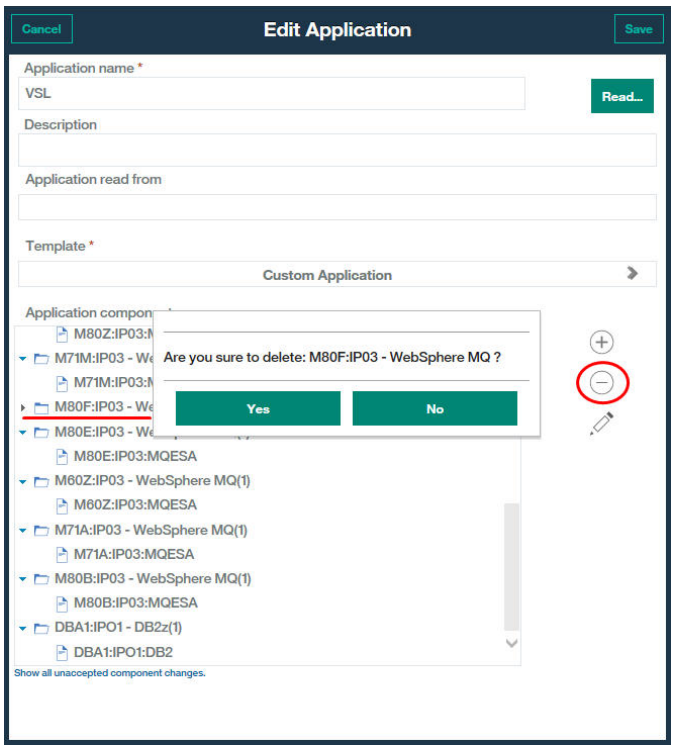
1. 如果应用程序性能仪表板未显示，请从 **性能**菜单中选择，或者如果在其他控制台页面上，请单击**主页**链接。

2. 在导航器的“**应用程序**”部分中，选择脱机代理程序是其组件的应用程序并单击 **编辑应用程序**。



3. 在“应用程序组件”列表中选择代理程序或代理程序子节点，然后单击 。

   对于具有子节点的代理程序，在折叠树时单击名称以只选择代理程序，展开节点树（单击 ）并单击节点以选择节点和所有子节点，或者展开节点树并单击实例以选择各个子节点。



4. 完成编辑应用程序以移除脱机代理程序或子节点后，单击**保存**。

结果

在保存更改后，您的应用程序更新将由 Cloud APM 服务器完成。从应用程序性能仪表板中移除脱机代理程序前可能需要几分钟时间。

脱机代理程序的示例

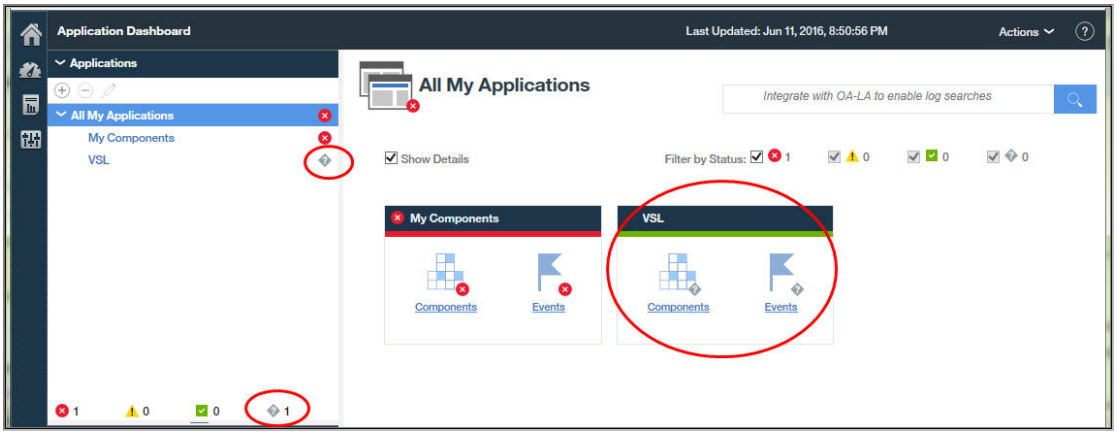
复审脱机代理程序如何在 Cloud APM 控制台中显示的示例。您可以从显示中移除不再希望监视的任何脱机代理程序。 如果代理程序稍后重新联机，监视恢复。

代理程序脱机后，不会向 Cloud APM 控制台发送数据，且“应用程序性能仪表板”将对该代理程序及其所属应用程序显示 状态指示符。代理程序不可用于添加到“应用程序”编辑器中已定义的应用程序或资源组管理器中的定制组，或者不可用于在“属性详细信息”选项卡中创建表和历史折线图。

应用程序性能仪表板 - “我的所有应用程序”

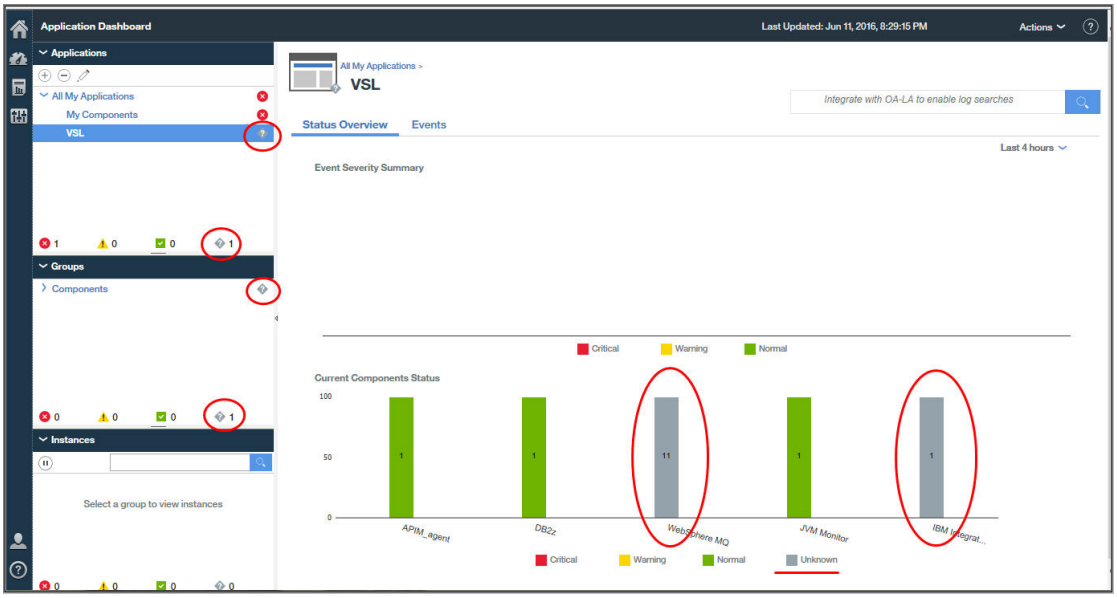
主仪表板页面“**我的所有应用程序**”提供了脱机状态的第一个指示。导航器“应用程序”部分中的计数器显示包含不可用资源的应用程序的数目。

摘要框显示“正常”事件状态，因为没有为任何应用程序的受管资源打开事件。



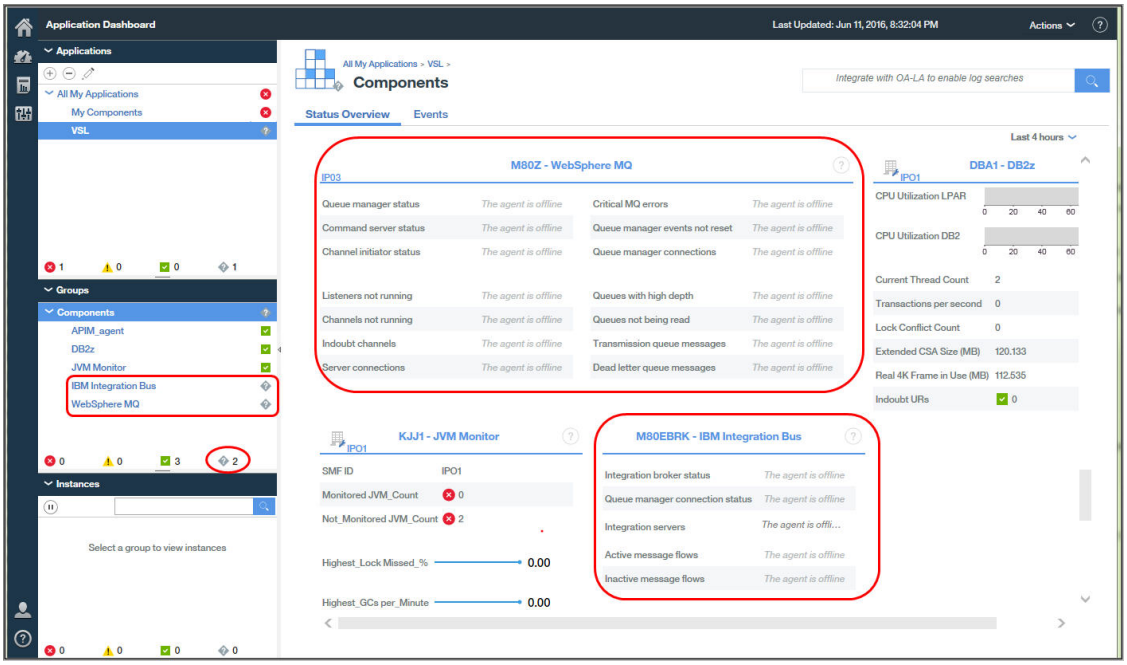
应用程序性能仪表板 - 应用程序

用户单击摘要框标题栏或从导航器中选择应用程序后，显示包含空的“**事件严重性摘要**”图表的“**状态概述**”选项卡。“**当前组件状态**”栏将脱机代理程序的状态显示为“未知”“”。



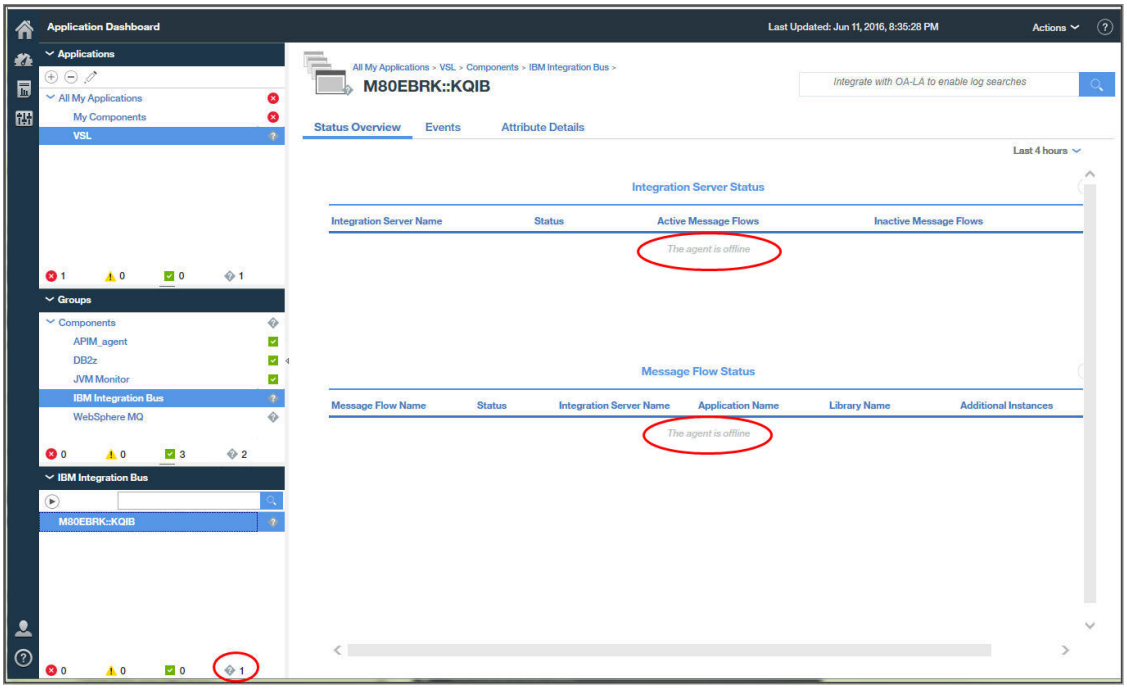
应用程序性能仪表板 - 组

用户在“**当前组件状态**”图表或导航器“**组件**”组中单击后，“**状态概述**”选项卡更改以显示每个受管资源的摘要组窗口小部件。不可用代理程序的组窗口小部件显示代理程序脱机的消息而不是 KPI。

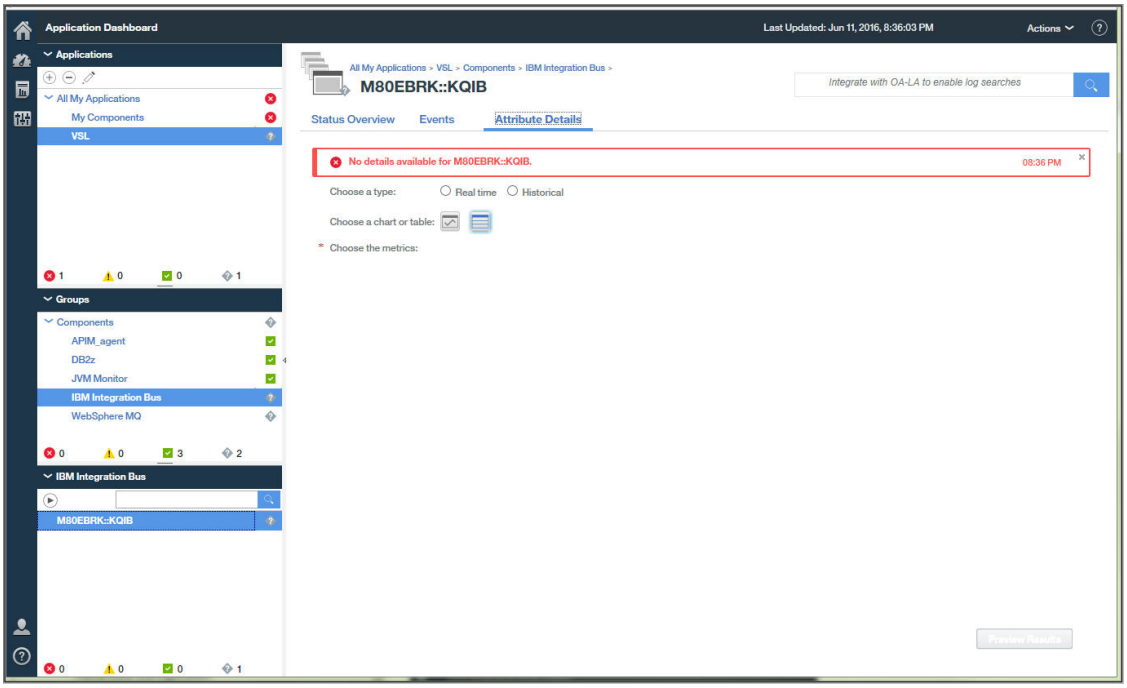


应用程序性能仪表板 - 实例

用户在某个脱机代理程序摘要组窗口小部件中单击后，“**状态概述**”选项卡显示所选代理程序的图表窗口小部件和表窗口小部件。但是，就像摘要组窗口小部件一样，只显示指示代理程序脱机的消息，而不显示代理程序实例的 KPI。



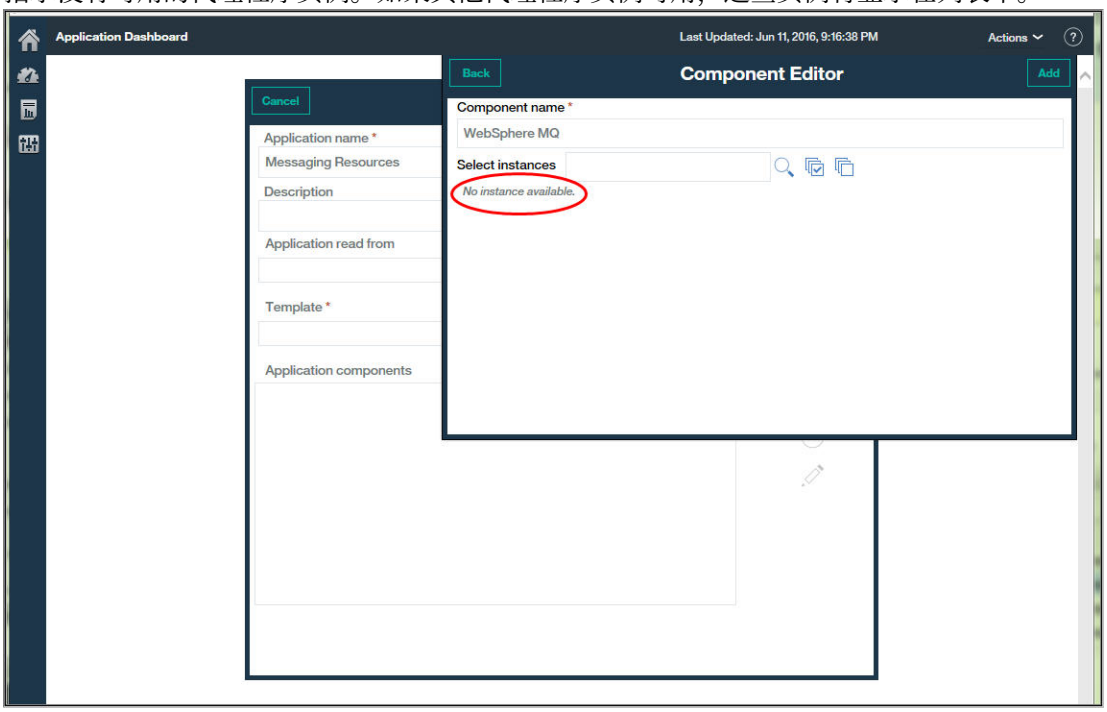
用户单击“**属性详细信息**”选项卡后，消息显示未提供代理程序实例的详细信息。无法为脱机代理程序实例创建定制图表或表。



应用程序编辑器

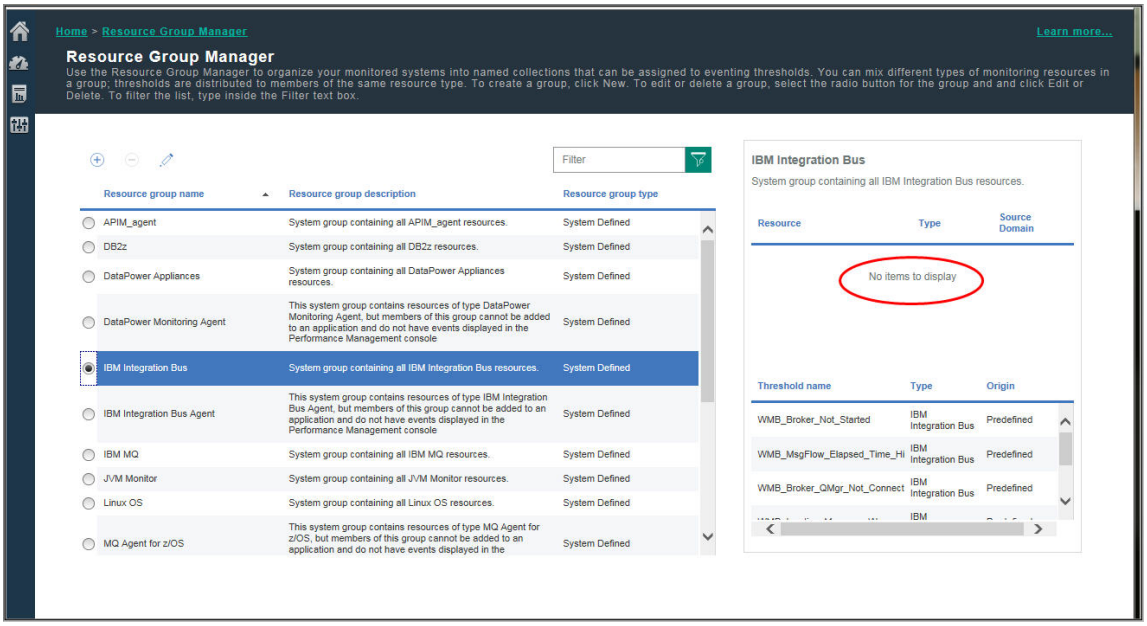
在应用程序性能仪表板中，用户单击导航器“应用程序” **添加应用程序** 或 **编辑应用程序** 工具后，“应用程序”编辑器窗口弹出。

用户单击 **添加组件** 并选择代理程序类型后，如果该类型的代理程序未安装或脱机，将向您发送消息，指示没有可用的代理程序实例。如果其他代理程序实例可用，这些实例将显示在列表中。



资源组管理器

用户选择 **系统配置 > 资源组管理器**后，将打开包含资源组表的页面。选择一个组时，将列出该组中的代理程序实例以及分配的阈值。如果所有代理程序实例都是脱机的，那么消息将显示没有可用的实例。



相关概念

第 930 页的『管理应用程序』
使用应用程序性能仪表板中提供的工具将受管资源组成应用程序。

第 915 页的『使用仪表板』

相关参考

第 833 页的『资源组管理器』
您的受监视环境可能具有多个按用途分类的受管系统。此类系统通常具有相同的阈值需求。使用“**资源组管理器**”可将受管系统分组到可分配阈值的组。您也可以创建与“基于角色的访问控制”(RBAC) 策略关联的资源组。

事件状态

使用“**事件状态**”获取所选导航器项的打开事件的摘要概述，并通过向下钻取到详细仪表板对具有严重或警告状态的事件做出响应。

状态指示符用于源自受管系统上运行的阈值的事件。如果配置了 Hybrid Gateway，事件也可能源自 IBM Tivoli Monitoring 环境中的受管系统上运行的情境。如果您的配置包含 IBM Operations Analytics - Predictive Insights，那么还会显示检测到的任何异常。

某些阈值事件不会显示在应用程序性能仪表板中。这些阈值使用未发布资源的属性，此情况可能出现在支持子节点的代理程序中。（要获取子节点的描述，请参阅 Agent Builder 主题，).

严重、警告和正常

- 这些状态指示符整合了源自阈值的事件严重性：
  - “严重”状态表示严重性为“致命”或“严重”的所有事件
  - “警告”状态表示严重性为“轻微”或“警告”的所有事件
  - “正常”状态表示严重性为“未知”的所有事件

  未知状态表示受管系统脱机。在脱机达到 4 天之后，该受管系统将从所有应用程序中移除，且不会再显示在仪表板中。
  要检查状态，停止或启动代理程序，请参阅第 150 页的『使用代理程序命令』中的**移除脱机系统延迟**选项
- V6-V7 当配置了一个或多个 Hybrid Gateway 时，对于源自 Tivoli Monitoring 情境的事件，状态指示符与源自阈值的事件相同，只有一点例外，即“正常”状态表示严重性为“无害”、“参考”或“未知”的事件。

- 当受管环境包含 IBM Operations Analytics - Predictive Insights 时，将通过状态指示符上方的菱形图标指示检测到的任何异常，例如 。有关更多信息，请参阅第 941 页的『使用 Operations Analytics - Predictive Insights 调查异常』。

“事件严重性摘要”百分比标尺

- “事件严重性摘要”标尺显示“严重”、“警告”和“正常”事件状态百分比。例如， 

显示 50% 的事件源自阈值且严重性为“轻微”或“警告”，50% 源自阈值且严重性为“致命”或“严重”。
- 此外还报告了事件总数和每个状态级别的事件数。
- 事件计数包括来自 Operations Analytics - Predictive Insights 的任何异常。例如，“总数为 8，包括 1 个异常”表示有 7 个阈值事件，1 个异常事件。

事件表

- 包含打开事件和状态的表由所选导航器项定义：应用程序、组、子组或实例。
- 事件按“**严重性**”列排序，严重性最高的显示在最前面。单击列标题以更改排序顺序。
- 每一行提供有关事件的以下信息：

阈值名称
提供给阈值的名称。

V6-V7 提供给情境的名称。

状态
事件的状态，例如“**打开**”。

严重性
事件的严重性值：“严重”（适用于“致命”和“严重”阈值严重性）、“警告”（适用于“轻微”和“警告”阈值严重性）或“正常”（适用于“未知”阈值严重性；对于 Tivoli Monitoring 事件，适用于“无害”、“参考”和“未知”严重性）。

未知状态表示受管系统脱机。在脱机达到 4 天之后，该受管系统将从所有应用程序中移除，且不会再显示在仪表板中。（要检查状态，停止并启动代理程序，请参阅第 150 页的『使用代理程序命令』。）

在受管环境包含 IBM Operations Analytics - Predictive Insights 时，应用于历史数据的分析可能会检测到异常并打开事件。针对检测到的异常打开的事件通过叠放在状态指示符上方的图标进行指示，例如 。单击**查看异常分析**链接以在新浏览器标签页或窗口中打开 Predictive Insights 的“**服务诊断**”视图。使用该“**服务诊断**”视图可以查看支持该应用程序的组件中的异常行为。

显示项
仅适用于多行数据集。显示项是针对阈值选择的键属性，用于区分针对同一个受管系统打开的事件。

源
从监视代理程序派生的系统主机名或其他名称，用于标识事件源。

时间戳记
发生事件或者源代理程序观察到条件的日期和时间，以 Cloud APM 控制台用户的时区表示。

如果代理程序重新启动，或者代理程序的阈值定义已修改，那么该代理程序所采样的事件将关闭，然后重新打开（如果阈值条件仍为 true）。在这些情况下，**时间戳记**值将更新为源代理程序重新打开该事件的时间。

对于纯事件，每当源代理程序确定阈值条件为 true 时，代理程序都会打开新事件，该事件会替换前一个事件实例。在阈值条件最近一次求值为 true 之后，纯事件将保持打开 24 小时（或者可配置的小时数）。在 Cloud APM 控制台上，仅显示纯事件的最新实例。

描述
针对阈值所写入的描述（如果有）。

- 单击某一行可以展开有关事件的详细信息：

节点
节点实例的受管系统名称。

对于包含子节点的代理程序，**Enable Subnode Events** 选项控制是否显示子节点。有关更多信息，请参阅第 910 页的『UI 集成』。

阈值标识
阈值标识。

全局时间戳记
Cloud APM 服务器从源代理程序接收到事件的日期和时间，以 Cloud APM 控制台用户的时区表示。

类型
指示事件是纯事件还是采样事件。纯事件是自发通知。纯事件的阈值没有采样时间间隔或可监视当前值的常量度量值。

描述
针对阈值所写入的描述（如果有）。

公式
在阈值编辑器中编写的公式。例如，`Percent Failed > 10.000 AND Transaction Definition Name != 'Ignore_Resources'`。

如果已使用 EIF 槽定制功能来定制 **msg** 基本槽的值，那么将显示定制后的 **msg** 槽值，而不是显示阈值公式。有关更多信息，请参阅阈值管理器中的第 838 页的『转发 EIF 事件？』以及第 841 页的『定制事件以转发到 EIF 接收器』。

您可以选择和展开其他行，或者再次单击以折叠行。当行处于展开状态时，可以向下钻取到受管系统的仪表板，使用此仪表板可以帮助确定事件原因。

使用 Operations Analytics - Predictive Insights 调查异常

仅限 IBM Cloud Application Performance Management：当受管环境包含 IBM Operations Analytics - Predictive Insights 时，应用于历史数据的分析可以检测异常并打开事件。 使用应用程序性能仪表板查找和查看 Operations Analytics - Predictive Insights 检测到的异常。

开始之前

Operations Analytics - Predictive Insights 必须与 Cloud APM 环境集成，才会在应用程序性能仪表板中向您发出异常警报。有关更多信息，请参阅第 825 页的『与 Operations Analytics - Predictive Insights 集成』。

关于此任务

应用程序性能仪表板显示域中的应用程序及其组件受管系统的状态摘要。**我的所有应用程序**仪表板摘要框中的事件状态显示 “严重”、“警告”和 “未知”严重性。如果事件包括 Operations Analytics - Predictive Insights 检测到的异常，那么状态指示符将包含异常图标：、或。 在您向下钻取应用程序、组和实例仪表板页面时还会在**事件**选项卡标题旁显示相同的“严重”和“警告”异常状态指示符：Events 2。要获取实践演示，请启动 IBM Cloud Application Performance Management Guided Demo，向下滚动“任务”列表，然后选择识别和诊断预测洞察异常情况。

过程

完成以下步骤以在 Operations Analytics - Predictive Insights 的“**服务诊断**”视图中标识和查看异常：

1. 单击 **性能** > **应用程序性能仪表板**，以打开**我的所有应用程序**仪表板。
2. 如果摘要框包含显示异常图标的 **事件**状态指示符，单击**事件**链接。
   此时应用程序仪表板会打开至**事件**选项卡。“**事件严重性摘要**”报告包括异常数量在内的事件总数。
3. 单击异常事件所在的表行，可以通过“**严重性**”列中的、或识别异常事件。
   该行将展开以显示事件详细信息。
4. 单击**查看异常分析** 以在新的浏览器标签页或窗口中打开 Operations Analytics - Predictive Insights “**服务诊断**”视图。

下一步做什么

- 使用该“**服务诊断**”视图可以查看支持该应用程序的组件中的异常行为。单击 以打开“**服务诊断**”视图的联机帮助。
- 返回到应用程序仪表板并查找受管系统上可能指示相关问题的任何其他事件。单击**状态概述**选项卡并向下钻取到发生该事件的受管系统实例以进行进一步调查。使用该信息确定需要执行的操作以避免发生 Predictive Insights 所标识的问题。
- 如果您认为应看到异常但没有显示异常，那么表示 Operations Analytics - Predictive Insights 训练时间可能不够长，因此无法产生异常。通常需要两周的训练时间。也可能是需要其他配置。

定制视图

通过定制自己的页面，使用 IBM Cloud Application Business Insights Universal View 增强预定义的应用程序性能仪表板页面已经提供的值。

Universal View 可用于显示资源监视数据。它不可用来显示综合事务数据、事务跟踪数据、响应时间代理程序数据和深入诊断数据。通过使用 Universal View，可以为应用程序快速构建监视页面并加以保存以供查看。在查看保存的定制仪表板页面时，可以自动刷新方式查看仪表板，将仪表板导出为原始数据文件，编辑仪表板，或者删除仪表板。

Cloud APM 中的四个缺省角色：角色管理员、监视管理员、系统管理员和监视用户具有不同的查看和修改仪表板页面的许可权。有关更多信息，请参阅表 1. 角色和许可权 。

在“**定制视图**”选项卡中提供的选项取决于页面是处于编辑状态还是查看状态。

创建和管理定制页面

使用“定制视图”选项卡，通过添加或更新填充了所选资源度量值的窗口小部件，创建或编辑所选应用程序/组/实例的仪表板页面。

关于此任务

您创建并保存的页面将与选择的应用程序相关联。例如，Cloud APM Guided Demo 中的“库存管理”应用程序有下列监视代理程序：Linux OS、MySQL、Node.js、Hadoop 和 Ruby。您可以在导航器的任何级别（从应用程序到实例）创建并保存定制页面，然后在创建该页面的级别将其打开。在特定级别创建的页面只能在同一级别打开。可用于窗口小部件的度量可以来自应用程序中的任何资源。使用“库存管理”示例，您可以创建一个页面，并使其包含一个来自 Ruby 代理程序的表，以及一个来自 Linux OS 代理程序的图表，等等。

过程

您创建并保存的页面将与选择的应用程序相关联。要创建并定制仪表板页面，请完成下列步骤：

1. 从 **性能**菜单打开“应用程序性能仪表板”之后，选择应用程序。
   “**定制视图**”选项卡显示在“**状态概述**”和“**事件**”选项卡之后。您也可以深入探索到导航器的组、子组或实例级别。
2. 单击**定制视图**选项卡。
   此选项卡显示“**为您的定制页面选择模板**”窗口或缺省页面（如果已设置缺省页面）。
   - 如果“**为您的定制页面选择模板**”窗口打开，请转至步骤 4。
   - 如果显示缺省页面，请转至步骤 3。
3. 单击 **添加**以创建新页面。
4. 从以下缺省模板选项中单击一个模板：
   - 1x1 模板
   - 1x2 模板
   - 2x2 模板

- 2x3 模板
- 3x3 模板
- 3x2 模板
- 2x1 模板
- 1x3 模板
- 3x1 模板

如果单击**返回**，将打开标记为收藏的页面或者列表中的第一页。如果不存在任何页面，那么将打开“**选择定制页面模板**”窗口。

5. 定制模板。有关详细信息，请参阅定制模板。
6. 创建窗口小部件。有关详细信息，请参阅第 945 页的『定义窗口小部件属性』。
7. 单击  **设置页面的缺省时间范围**并将页面的缺省数据保留期设置为 1、2、4、12 或 24 小时。
8. 当您准备好保存该页面时，请完成下列步骤：
   a) 在**页面名称**字段中，输入该页面的名称。

   要点: 页面名称字段中允许使用空格、下划线 (_) 和连字符 (-)。但是，不允许连字符后面跟着下划线 (-_)。例如，`System-_Overview` 是不允许的。

   b) 单击**保存**。

   在仪表板上可能会发生下列更改，或者可能会显示消息：

   - 将显示仪表板已保存消息。
   - 如果在**设置条件**中选择 *****，那么将显示以下消息：

     您已在“资源实例”或“设置条件”中选择 `*`，这会生成大量数据系列（例如图形上的线条）。这个数量的数据序列可能会引起页面可读性或性能问题。此图表的建议限制是 `50` 个数据序列。添加具体的值有助于将数据范围缩小到建议的限制之内，并改善用户体验。

   - 在  上会显示红色指示符，表示尚未选择该图表类型，您必须加以选择。
   - 在  上会显示红色指示符，表示尚未选择“选择度量”，您必须加以选择，然后会显示以下消息：

     必须保存度量才能保存图表。
9. 在页面标题栏中选择任何以下选项：

选项	描述
 **检索最新资源度量类型**	单击该选项可刷新度量类型。如果在应用任何代理程序补丁时度量类型或度量存在更改，您需要刷新度量类型。 两次刷新之间的时间间隔限制为 15 分钟。如果在上次刷新后 15 分钟内单击  **检索最新资源度量类型**，那么显示以下消息： 最近已刷新元数据高速缓存。请等待 *time_remaining* 分钟以将其重新装入。 装入元数据时，显示装入图像。 装入元数据时，显示以下消息： 元数据已成功地重新装入。 如果元数据刷新耗用的时间超过 30 秒，那么显示以下消息： 重新装入元数据高速缓存可能会耗用更多时间。要等待其完成吗？ 您可以单击**确定**或**取消**。
 **查看仪表板**	单击该选项可在仪表板中查看数据。

选项	描述
	要点: 针对每一项数据定义，返回的行数限制是 11,000 行。缺省情况下，超出限制时，将会显示最新的数据。数据量较大时，不会显示所选时间间隔内的全部数据。例如，对于数据量较大的数据源，如果您选择查看过去 24 小时的数据，那么可能会因为达到 11000 行的限制而仅显示过去 6 小时的数据。 如果图表中的数据序列超过 50 个，那么在窗口小部件中将会显示以下消息: 此图表无法装入，因为数据序列数目（例如图形上的线条数目）超过 50。数据序列数目当前为 *current_data_series*。您可减少选择度量或资源实例，或者优化每个度量的条件，以减少该数目。有关更多信息，请参阅“定义窗口小部件属性：对于 Cloud APM - http://ibm.biz/widgetprops 对于 Cloud APM Private - http://ibm.biz/widgetprops-private 如果装入图表所花费的时间超过 30 秒，那么在窗口小部件中将会显示以下消息: 因为数据量过大、网络等待时间过长或者存在连接问题，装入此图表所花费的时间过长。请减少资源实例数，或者优化每个度量的条件以缩小数据范围。有关更多信息，请参阅“定义窗口小部件属性：对于 Cloud APM - http://ibm.biz/widgetprops 对于 Cloud APM Private - http://ibm.biz/widgetprops-private
另存为	单击**保存**旁边的箭头，单击**另存为**，然后在**页面名称**字段中指定其他名称以使用其他名称保存页面。 **要点:** 如果指定与现有页面相同的页面名称，那么会覆盖现有页面。
删除	单击该选项可删除当前页面。
Back **返回**	单击以返回到上一页或收藏页面。

下一步做什么

查看定制页面，如第 947 页的『查看定制页面』中所述。

定制模板

您可以定制模板，方法是根据您的需求对窗口小部件占位符调整大小、进行移动或添加。

关于此任务

切记: 您可以定制现有的模板并加以使用。但是，您无法保存定制模板以供将来用于创建新仪表板。

过程

1. 在“定制视图”选项卡中，单击 **编辑模板**。
2. 选择窗口小部件占位符。
   您可以从所有侧边调整窗口小部件占位符的大小并将其拖动到其他位置。如果在对窗口小部件调整大小或拖动时它们相互重叠，那么将显示调整大小操作无效或移动操作无效消息。
3. 要将窗口小部件占位符添加到现有的模板，请完成下列步骤:
   a) 单击菜单选项中的**设置页面高度**，指定较高的行计数值，然后单击该菜单外部的任意位置以增加页面高度。
   b) 在页面上的空白区域中，通过定位指针并拖动以建立一个框，根据需要指定窗口小部件占位符。

4. 使用以下菜单选项可对模板完成不同操作：

选项	描述
撤销	撤销上一个操作。
重做	重做上一个操作。
删除选定框	要删除窗口小部件，请选择该窗口小部件并单击**删除选定框**图标。
重置	创建空白模板。 要在空白模板上指定窗口小部件占位符，请将指针放在“将模板拖到此处”区域并将其拖动以创建框。您可以在“将模板拖到此处”区域中创建不同大小的窗口小部件占位符。这些占位符可以移动或调整大小，但是不能相互重叠。
设置页面长度	设置页面的高度。您可以指定的行计数为 20 到 120 行。

5. 单击  **编辑模板**以使用创建的模板。

下一步做什么
创建窗口小部件。转至“创建和管理定制页面”主题中的步骤 6。

定义窗口小部件属性
为窗口小部件定义不同属性（如度量值和图表）以在窗口小部件中查看实时数据。

过程
要为窗口小部件定义属性，请完成以下步骤：

1. 在窗口小部件中，单击  以选择用于显示数据的图表类型。
   - 折线图
   - 区域
   - 条形图
   - 网格

   要点：对于折线图、面积图和条形图，如果存在 9 个以上的图例，那么图形颜色在第九个图例之后重复。图形颜色对于第 1 个和第 10 个图例是相同的，对于第 2 个和第 11 个图例是相同的。

   在  上会显示绿色指示符，表示选中该图表类型。
2. 为行、区域和条形图指定以下图标属性：
   - X 轴标签
   - Y 轴标签
   - 显示图注
   - 显示插值：所收集的要在图表上绘制的数据，可能包含某些空值。因此，当绘制该图表时，图表线条会在遇到空值时断开，因此图表中会出现多条断开连接的线条。如果选择插值，图表上的线条在遇到空值时不会显示为断开，而是连接到下一个可用的有效值。因此，选择插值时会获得一条始终连接的图表线条。

     注：对于 APM V8.1.4.0 IF0005 及更高版本，折线图和面积图不再对空值显示断开连接的线条。因此，不再需要“显示插值”功能，也不再支持此功能。

   要点：网格没有属性。
3. 单击  以选择度量内容。

选项	描述
资源类型	从**资源类型**列表中，选择资源。可用资源与应用程序关联。

选项	描述
	如果列表中未提供属于该应用程序的资源，那么找不到其资源定义。这是因为该资源定义尚未发布，或者受管系统未与 Cloud APM 服务器建立连接。
度量类型	从**度量类型**列表中，选择要包括在窗口小部件中的数据集。
度量	从**度量**列表中，选择要包括在视图中的属性。可用的属性来自选择的数据集。 要选择度量，请完成下列步骤： a. 单击**度量**列表。 这会打开一个弹出窗口，其中以字母顺序的升序列示各个度量。 b. 单击列示在“度量”下的属性（选中一个或多个），或单击**全部选中**。 **注:** 单击**全部选中**时，将会选中该列表中所有的度量。 c. 单击  可添加**选中的度量**列表下的度量。 d. 如果要删除**选中的度量**下的度量，请单击  。 e. 单击**资源实例**列表可关闭弹出窗口。 **要点:** 对于折线图、条形图和面积图，需要选择包含数字值的度量。包含字符串值的度量不能显示在这些图表中。 **提示:** 对于网格，请根据适合在 UI 上查看的输出来限制度量选择。
资源实例	初始选择为 *，表示从列表中的所有实例检索度量值。保留缺省选择，或从列表中选择实例。 **要点:** 如果选择实例，那么此窗口小部件不能用于显示任何其他代理程序或实例的数据。但是，最好指定实例以避免处理大量数据。
设置度量组的条件	如果选择的**度量类型**有多个元素，例如 CPU 或磁盘，那么**设置度量组的条件**将显示其他元素，供您从 **WHERE** 字段中选择。 **要点:** 请在 **WHERE** 字段中指定各个元素的值。请勿指定 ***** 以避免处理大量数据。 缺省情况下，WHERE 条件显示过去 4 小时的数据。系统管理员可以此将时间间隔更改为介于 1 与 24 小时之间的值，请参阅第 947 页的『更改 WHERE 条件数据的时间间隔』。
操作	单击  **保存**可保存度量。 单击 **编辑**可编辑度量。 单击  **删除**可删除度量。

4. 要添加另一个度量，请单击 **+ 添加另一个度量**。

   要点: 不适用于网格。
5. 在添加所有度量后关闭“**选择度量**”窗口。

   在关闭“**选择度量**窗口后自动保存所有度量。

   在仪表板上可能会发生下列更改，或者可能会显示消息：

   · 如果在任何度量的**资源实例**列表中选择 *****，那么将显示以下消息：

您已在“资源实例”或“设置条件”中选择 *，这会生成大量数据系列（例如图形上的线条）。这个数量的数据序列可能会引起页面可读性或性能问题。此图表的建议限制是 50 个数据序列。添加具体的值有助于将数据范围缩小到建议的限制之内，并改善用户体验。

- 在  上会显示绿色指示符，表明您已正确选择度量，将数据范围缩小到建议的限制之内。
- 在  上会显示橙色指示符，表明您未正确选择度量来将数据范围缩小到建议的限制之内。在**资源实例**或**设置条件**中选择了 ***。

6. 单击  以输入窗口小部件标题。
   如果未添加窗口小部件标题，那么自动将第一个度量名称指定为窗口小部件标题。

下一步做什么
同样地，将图表、度量和标题添加到所有窗口小部件，然后转至“创建和管理定制页面”主题中的步骤 7。

更改 WHERE 条件数据的时间间隔
系统管理员可以将时间间隔更改为介于 1 与 24 小时之间的值。

过程

要更改时间间隔，系统管理员可以完成下列步骤：

1. 登录构建所部署在的 APM 服务器。
2. 在命令行上，运行以下命令：

```
export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derbytools.jar:
export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derbyclient.jar:
export CLASSPATH=$CLASSPATH:/install_dir/gaian/lib/derby.jar:
java org.apache.derby.tools.ij
connect ’ jdbc:derby://localhost:port/
gaiandb;user=gaiandb;password=gaian_db_password;’ ;
```

   在这些命令中，*install_dir* 是指 APM 所部署在的目录，缺省情况下，这是 /opt/ibm。在 **connect** 命令中，*port* 是指配置数据库的端口值，*gaian_db_password* 是 Gaian 数据库密码。请联系 IBM 支持以获取此密码。
3. 连接数据库之后，请运行以下查询，以修改时间范围值：

```
UPDATE "OED_TOOL"."PREFERENCETABLE" SET PREFERENCES='-24H' WHERE
FIELD='TIMEINTERVAL';

Commit;

Exit;
```

   这里，时间范围值是在 SET PREFERENCES='-24H' 中指定。它可以设置为 1H 到 24H 的值。

查看定制页面

在“**定制视图**”选项卡中为应用程序、组、子组或实例创建并保存仪表板页面后，您可以随时进行查看。您可以选择的一些选项包括刷新页面、选择其他时间间隔、编辑页面以从不同资源检索数据以及将仪表板导出为原始数据文件。

过程

完成以下步骤以在仪表板的“**定制视图**”选项卡中查看已保存的页面。

1. 从 **性能**菜单打开“应用程序性能仪表板”之后，选择应用程序。
   “**定制视图**”选项卡显示在“**状态概述**”选项卡和“**事件**”选项卡之后。您也可以深入探索到导航器的组、子组或实例级别。
2. 选择“**定制视图**”选项卡。

此选项卡显示“**为您的定制页面选择模板**”窗口或缺省页面（如果已设置缺省页面）。

3. 单击页面列表中的 ，然后从列表中选择一个已保存的页面。
   可用页面就是您保存的或者被其他用户共享的页面。
   在您选择已保存的页面后，系统将在该页面中报告当前数据样本和历史数据样本。
4. 在页面标题栏中选择任何查看选项：

选项	描述
 **刷新**	表示自动刷新已关闭。单击以打开自动刷新。
 **刷新**	表示自动刷新已打开。单击以关闭自动刷新。 **要点:** 缺省刷新时间为 1 分钟。
 **导出 > 原始数据**	单击以将页面导出为 DAT 格式。因为导出多个 DAT 文件，它们会以 ZIP 格式下载到计算机。 **切记:** 如果下载的文件没有任何扩展名，请对其添加 zip 扩展名。 **要点:** 如果该文件未下载到您的计算机，请检查是否已启用拦截弹出广告窗口的软件。您可将此站点添加到例外列表中。 将下载的 ZIP 文件解压缩。解压缩后的文件是纯文本文件。该文件包含页面名称、持续时间、过滤器、日期、时间、时间间隔、图表标题和数据。数据分隔符是竖线。 您可使用适当的编辑器打开 DAT 文件，或者通过指定适当的值分隔符将这些文件导入到 EXCEL 中。
 **导出 > PDF**	单击以将页面导出为 PDF 格式。 该文件将包含页面名称、时间间隔、窗口小部件、创建者以及创建报告的时间。
 **编辑**	单击该选项可编辑当前页面。 您可以在窗口小部件中更改图表和度量，或添加新窗口小部件，更改该页面的缺省时间范围，或编辑该页面的名称。
 **删除**	单击该选项可删除当前页面。
 **添加**	单击以创建新页面。要定制该页面并进行保存，请参阅第 942 页的『创建和管理定制页面』。

5. 在窗口小部件中选择任何以下查看选项：

选项	描述
 **图表类型**	单击**图表类型**图标，然后从列表中选择相应的选项以更改现有图表类型。 · 对于折线图和面积图，**折线**和**面积**选项可用。 · 对于条形图，**簇状条形图**和**簇状柱形图**选项可用。 **要点:** 对于网格，没有图表类型选项。 对于网格中显示的数据，您可以过滤数据，如下所示： a. 单击  **定义过滤器**。“**过滤器**”窗口随即打开。

选项	描述
	b. 为**列**、**条件**和**值**指定值以添加过滤规则。 **注:** 您可以选择相应条件来过滤数字值和文本值。 c. 单击  **添加过滤规则**以添加其他过滤规则。您可以添加多个过滤规则。 d. 在**匹配**字段中，选择**所有规则**或**任何规则**来过滤数据。 如果想要根据在**值**字段中输入的文本大小写进行搜索，可以选择**匹配大小写**。 e. 单击**过滤**以过滤网格中显示的数据。 f. 单击**清除过滤**以清除过滤结果。 g. 单击**取消**以关闭“**过滤器**”窗口。
折叠	单击该选项可折叠窗口小部件。
展开	单击该选项可展开窗口小部件。
最大化	单击该选项可使窗口小部件最大化到页面大小。
复原	单击该选项可将窗口小部件复原到其原始大小。
图例	窗口小部件包含每个度量的复选框。选择或取消选中每个度量的复选框以查看特定度量或多个度量的数据。

6. 您可以使用**日期**、**时间**和**时间间隔**列表来过滤页面上的数据。您还可以为页面定义定制过滤器来显示所选日期和时间间隔的数据。要使用定制过滤器，请从**时间间隔**列表中选择**定制**，然后在“**时间段选择**”窗口中选择所需日期和时间间隔。

   注:

   - 从 APM V8.1.4.0 IF0005 开始，定制过滤选项可用。使用较低版本的 Cloud APM 创建的页面不显示定制过滤器选项。
   - 使用定制过滤器对最小时间间隔为 1 分钟且最大时间间隔为 24 小时的数据进行过滤。
   - 应用定制过滤器时，在“**时间段选择**”窗口中，如果您单击**取消**，那么在仪表板页面上**时间间隔**列表不显示先前已应用的时间间隔。
   - 如果将定制过滤器应用于页面，那么不自动刷新页面上的数据。

7. 要设置缺省页面，请单击页面列表中的 ，然后单击要设置为缺省页面的页面名称旁边的  **收藏夹**。

仪表板实用程序

使用可用选项可管理**应用程序性能仪表板**页面的外观和行为。

复制仪表板 URL

在应用程序层次结构中导航到某个位置后，浏览器地址框中的 URL 不会针对新视图而发生更改。 您可以复制所显示的应用程序性能仪表板页面的 URL。将该 URL 粘贴到新浏览器窗口中以打开该仪表板页面，或者使用该 URL 稍后访问该仪表板或分享给他人。

过程

1. 导航到想要记住的应用程序性能仪表板页面。

2. 单击**操作 > 复制 URL**。
3. 右键单击**链接到当前页面**超文本链接并选择用于复制 URL 的选项。

下一步做什么

保存 URL 副本或分享给受管环境中的其他用户。在将该 URL 粘贴到浏览器的地址框后，将在 Cloud APM 控制台中打开目标仪表板页面。

如果您没有登录到 Cloud APM 服务器，将提示输入用户标识和密码，然后才会显示目标仪表板页面。如果打开的是“**入门**”页面，而不是仪表板页面，请按 F5 键或单击浏览器的刷新工具栏按钮。通过清空“启动时显示此‘**入门**’页面”复选框，可以关闭将来工作会话的“**入门**”页面。

设置跟踪

调整跟踪设置以帮助管理员或 IBM 支持人员诊断应用程序性能仪表板的问题的原因。在使用导航器和“**状态概述**”选项卡时，可以使用多种跟踪级别。您可以在用户界面中遇到问题的准确位置启动详细级别的跟踪，然后在捕获必要日志数据后将跟踪恢复为较低级别。例如，如果特定仪表板行为异常，可以提升跟踪级别，然后再打开该仪表板以记录活动，之后再将跟踪日志记录恢复为正常级别。

关于此任务

请执行以下步骤以在想要增加或减少跟踪日志记录量时设置跟踪级别。

过程

1. 如果未打开应用程序性能仪表板，请从导航栏中的**性能**选项中选择。
2. 选择“我的所有应用程序”，或者从导航器或“状态概述”选项卡中选择应用程序。
3. 单击**操作 > 跟踪级别**，然后选择以下级别之一：
   - **详细**用于记录所有活动。“详细”跟踪级别包括“中等”、“较少”和“最少”跟踪日志记录。
   - **中等**用于记录变量更改，例如，传递了哪些参数以及进行了哪些计算。“中等”跟踪级别包含“较少”和“最少”跟踪日志记录。
   - **较少**用于记录错误和变量活动。如果您遇到没有返回数据但仪表板仍在运行之类的问题，可能需要将跟踪设置为此级别。“较少”跟踪级别包含“最少”跟踪日志记录。
   - **最少**是缺省设置，仅记录不可恢复错误。在收集特定活动序列之后，可以将跟踪级别恢复为“最少”。即使注销前设置了其他跟踪级别，下次登录时跟踪也始终重置为最低级别。
4. 如果希望将性能记录发送到公共日志记录文件，请选择**启用日志性能统计信息**。

   来自控制台的性能信息将写入服务器，在该服务器中，此性能信息与来自服务器的性能统计信息组合起来以提供端到端事务响应时间。所需性能信息包括功能启动的时间以及结束的时间。

结果

跟踪调整为所选级别。下次登录时，跟踪为**最少**，除非您再次对其进行更改。

为使通信流量保持最低，将批量传输日志消息。注销后将手动或一段时间后执行最后一次传输。（如果浏览器失败，将不会发送最后的日志记录。） 日志保存在服务器计算机上，并命名为 `itp.log`。每次重新启动服务器时都会创建新的 `itp.log`。

如果设置了**启用日志性能统计信息**，将在 *install_dir*`/usr/servers/apmui/logs/itp.log` 中保存类似于以下示例的记录：

```
<record>
    <date>2013-10-02T10:52:46</date>
    <millis>1380736366788</millis>
    <sequence>28008</sequence>
    <level>INFO</level>
    <class>StatusItemList</class>
    <method>tracing</method>
    <thread>96</thread>
    <message>BeginTrace:onSelectApp:272wt877d05</message>
</record>
<record>
    <date>2013-10-02T10:52:46</date>
    <millis>1380736366809</millis>
```

```
    <sequence>28009</sequence>
    <level>INFO</level>
    <class>StatusItemList</class>
    <method>tracing</method>
    <thread>96</thread>
    <message>EndTrace:onSelectApp:272wt877d05</message>
</record>
```

锁定 Cloud APM 控制台

您可以暂时锁定工作会话，而不需要从 Cloud APM 控制台注销。 会话锁定功能在 Apple iPad 上不可用。

过程

1. 登录 Cloud APM 控制台之后，请单击 ***apmadmin*** > **锁定会话**，其中 *apmadmin* 是用来登录的名称。
   将显示登录屏幕并锁定会话。
2. 要解锁会话，请输入您的用户标识的密码。
   工作会话恢复。

报告

在 Cloud APM 控制台中，将提供 Response Time Monitoring Agent、WebSphere Applications 代理程序和 Synthetic Playback 代理程序所收集的数据的历史报告。

您可以从**我的所有应用程序**仪表板运行报告。从任何 Cloud APM 控制台页面，单击 **性能** > **应用程序性能仪表板** 以打开**我的所有应用程序**仪表板。

注: 首次运行报告时，必须以有权运行 Cloud APM 报告的用户身份登录 Tivoli Common Reporting。IBM Cognos Viewer 是缺省报告输出查看器。

Response Time Monitoring Agent 报告

要查看**应用程序性能和使用情况**或**比较两个时间段的应用程序性能**报告，选择包含 Response Time Monitoring 代理程序 受管系统的应用程序，然后选择**操作** > **启动至报告**。

要查看**我的所有应用程序**或**比较多个应用程序的性能**报告，选择**我的所有应用程序**，然后选择**操作** > **启动至报告**。

WebSphere Applications 代理程序 报告

要查看任何 WebSphere Applications 代理程序报告，选择包含 WebSphere Applications 代理程序受管系统的应用程序，然后选择**操作** > **启动至报告**。

Synthetic Playback 代理程序

要查看任何 Synthetic Playback 代理程序报告，选择包含合成事务的应用程序，然后选择**操作** > **启动至报告**。

注: 如果**操作**菜单中缺少选项**启动至报告**，请检查 Cloud APM 报告是否正确安装。

有关支持用于查看 Synthetic Playback 代理程序、Response Time Monitoring 代理程序 和 WebSphere Applications 代理程序 报告的浏览器的信息，请参阅 Cognos 10.2.1.7 的软件产品兼容性报告 Cognos 10.2.1.7。

Response Time Monitoring Agent 报告

针对 Response Time Monitoring 代理程序 收集的数据提供了历史报告。Response Time Monitoring Agent 报告在 Cloud APM, Base 中不可用。它们仅在 Cloud APM, Advanced 中可用。

针对 Response Time Monitoring 代理程序 收集的数据提供了两种类型的报告：动态和简单。

动态报告

动态报告可在浏览器中以 MHTML 格式查看。缺省情况下，Internet Explorer 支持 MHTML。对于其他浏览器，可以安装 MHTML 支持插件。动态报告也称为脱机交互报告。

简单报告

简单报告可在 IBM Cognos Viewer 中进行查看。IBM Cognos Viewer 是缺省报告输出查看器。

针对 Response Time Monitoring 代理程序收集的数据提供了以下历史预定义报告：

表 *254.* 预定义的历史报告

报告	**Type**
我的所有应用程序	动态
应用程序性能和使用情况	动态
比较两个时间段的应用程序性能	简单
比较多个应用程序的性能	简单

报告数据存储在 DATAMART Db2 数据库中。报告显示分别针对 26 周、12 个月和 3 年保留的每日、每周和每月摘要数据。Cloud APM 不提供用于变更这些保留期的脚本或指示信息。

有关 Response Time Monitoring 代理程序 和 Performance Management 报告之间的映射的更多信息，请参阅 Response Time Monitoring 代理程序属性映射。

“我的所有应用程序”报告

使用“我的所有应用程序”报告可查看有关用户设备、数据量、响应时间和错误计数的信息。

在此报告中，可查看您所有应用程序的信息。将报告时间段指定为**昨天**（缺省值）、**上周**或**上个月**。按应用程序和所选时间段查看以下信息：

· 事务计数堆积柱形图

· 事务数据量柱形图

· 平均事务响应时间柱形图

· 错误计数堆积柱形图

“应用程序性能和使用情况”报告

使用此报告可查看单个应用程序的性能、可用性和用户设备信息。

在**选择应用程序**窗口中，选择应用程序。单击**下一步**。在**选择应用程序的关键事务**中，选择要按其过滤报告的一个或多个事务。单击**确定**。此报告具有三个选项卡 - 性能、 可用性和设备。缺省时间间隔为周。

在“性能”选项卡上，通过为当前正在“应用程序性能仪表板”中查看的应用程序选择时间间隔，可查看以下信息：

· 平均响应时间（按（关键）事务）折线图

· 平均事务响应时间（按成功、服务器错误和客户机错误）折线图

· 事务数据量条形图和折线图，条形显示事务数据量，折线显示事务数据量平均值

· 事务计数条形图和折线图，条形显示事务计数，折线显示多项式值和移动平均值

在“可用性”选项卡上，通过为当前正在“应用程序性能仪表板”中查看的应用程序选择时间间隔，可查看以下信息：

· 成功与失败事务百分比堆积条形图，其中失败细分为服务器错误和客户机错误

· 成功与失败事务计数（按设备类型）堆积条形图，其中失败细分为服务器错误和客户机错误

· 最常出现的错误代码饼图

在“设备”选项卡上，通过为当前正在“应用程序性能仪表板”中查看的应用程序选择时间间隔，可查看以下信息：

· 事务（按设备类型）条形图

· 事务（按设备操作系统）条形图

· 事务（按设备浏览器）条形图

· 按维度显示事务性能的表，您可以基于设备类型、设备操作系统、设备品牌和设备浏览器来过滤此表

“比较两个时间段的应用程序性能”报告
使用此报告可检查所选应用程序的应用程序性能。

在**选择应用程序和时间频率**窗口中，指定应用程序和时间频率（每周、每天或每月）。在**选择时间段**窗口中，选择与时间间隔相应的时间段，然后单击**确定**。

此报告针对所选应用程序和所选时间段按所选时间间隔显示以下图表：

· 设备类型的事务计数折线图
· 事务量折线图
· 平均响应时间折线图
· 错误计数折线图

“比较多个应用程序的性能”报告
使用此报告可比较同一时间段多个应用程序的性能。

在**选择应用程序和时间频率**中，指定应用程序和时间频率（每周、每天或每月）。单击**下一步**。选择与时间间隔相应的时间段。

此报告针对所选应用程序和所选时间段按所选时间频率显示以下图表：

· 事务计数折线图
· 事务数据量折线图
· 平均响应时间折线图
· 错误计数折线图

Response Time Monitoring Agent 属性映射
一些 Cloud APM 报告基于 Response Time Monitoring Agent 收集的数据。这些报告中的数据映射到 Response Time Monitoring 代理程序 属性。

下表提供了 Response Time Monitoring 代理程序 报告中的数据项到代理程序属性的映射：

表 *255. Response Time Monitoring* 代理程序 属性映射

报告数据项	**描述**	**ODI 属性名称**	**ODI 文件列**
应用程序名称	报告给 Cloud APM 控制台的受监视应用程序的名称	应用程序名称	T5TXCS.APPLICATIN
事务计数	监视代理程序在当前汇总时间间隔内观察到的请求和响应序列总数。	请求总数	T5TXINS.TOTREQ
客户机错误数	具有状态码 400-499 的 HTTP 请求数。	客户机错误数	T5TXCS.NUM4XX
服务器错误数	具有状态码 500-599 的 HTTP 请求数。	服务器错误数	T5TXCS.NUM5XX
事务名称	向应用程序管理控制台报告的事务名称。	事务名称	T5TXCS.TRANSACTN
事务状态	与事务关联的响应代码	状态码	T5TXCS.STATUSCODE
代码回复千字节数	数据时间间隔内请求的每个回复中的千字节总数。	回复字节数	T5TXCS.REPLYBYT

表 255. *Response Time Monitoring* 代理程序 属性映射 *(续)*			
报告数据项	**描述**	**ODI 属性名称**	**ODI 文件列**
请求千字节数	数据时间间隔内请求中的千字节总数。	请求千字节数	T5TXCS.REQBYTES
千字节总数	时间段内为所有请求传输的千字节总数。	字节总数	T5TXCS.TOTBYTES
对象总计数	时间段内嵌入到 Web 页面中的对象总数	对象总计数	T5TXCS.OBJCNT
对象总大小	时间段内嵌入到 Web 页面中的所有对象的总大小。	对象总大小	T5TXCS.OBJSIZE
响应时间（秒）	完成总体服务器事务所需的总秒数。	响应时间	T5TXCS.RESPTIME
呈现时间	在 Web 浏览器上使用嵌入式 JavaScript 标记完整呈现 Web 页面的耗用时间（秒）。	呈现时间	T5TXCS.RENDERTIME
客户机时间	当前监视时间间隔内事务在客户机上运行所用的平均耗用时间（秒）。	平均客户机时间	T5TXCS.CLIENTTIME
装入时间	从用户请求下载到完成 Web 对象下载的平均耗用时间（秒）。	平均装入时间	T5TXCS.LOADTIME
浏览器	显示 Web 页面的 Web 浏览器的描述。	浏览器描述	T5TXCS.BROWSEDESC
服务器	用于 TCP 事务的服务器的名称或 IP 地址。	服务器描述	T5TXCS.SERVERDESC
URL 主机名	URL 的 TCP/IP 主机名。	URL 主机名	T5TXCS.URLHOST
URL 方法	用于执行 HTTP 请求（GET、POST、HEAD、PUT、OPTIONS、DELETE、TRACE 或 CONNECT）的方法。	方法	T5TXCS.METHOD
URL 详细信息	托管 Web 页面的服务器上文件的 URL 路径。	URL 路径	T5TXCS.URLPATH

有关 Response Time Monitoring 代理程序的更多信息，请参阅事务监视参考。

生成 Synthetic Playback 代理程序报告

针对与合成事务关联的应用程序运行报告。

关于此任务

在应用程序性能仪表板中选择应用程序和关联的合成事务，然后根据选择生成报告。报告数据存储在 DATAMART Db2 数据库中。报告显示分别针对 371 天、53 周和 12 个月保留的每小时、每周和每月摘要数据。Cloud APM 不提供用于变更这些保留期的脚本或指示信息。

提供了五个多页报告：

事务总体情况

这是一个两页报告，显示在设定日期范围内，所选合成事务的响应时间和可用性比率。

第一页显示以下数据：

· 折线图：显示在设定日期范围内按设定时间间隔获取的选定合成事务的响应时间

· 表：以秒为单位显示设定日期范围内每个合成事务的平均响应时间

第二页显示以下数据：

· 折线图：显示在设定日期范围内按设定时间间隔获取的选定合成事务的可用性比率

· 表：显示设定日期范围内每个合成事务的平均可用性比率

您可以从“**事务总体情况**”报告访问两个额外的报告，即“**按事务划分的实时分析**”和“**按事务划分的 HTTP 度量值**”。

“**按事务划分的实时分析**”显示在设定日期范围内按设定时间间隔获取的选定合成事务的 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的选定合成事务的 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内选定合成事务的 HTTP 度量值

“**按事务划分的 HTTP 度量值**”显示在设定日期范围内按设定时间间隔获取的选定合成事务的 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的选定合成事务的 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内选定合成事务的 HTTP 度量值

按位置划分的事务详细信息

这是一个两页报告，按位置显示在设定日期范围内，选定合成事务和子事务的响应时间和可用性比率。

第一页显示以下数据：

· 折线图：按位置显示在设定日期范围内按设定时间间隔获取的选定合成事务和子事务的响应时间

· 表：以秒为单位显示在设定日期范围内从每个位置获取的所有合成子事务的平均响应时间

第二页显示以下数据：

· 折线图：按位置显示在设定日期范围内按设定时间间隔获取的选定合成事务和子事务的可用性比率

· 表：显示在设定日期范围内从每个位置获取的所有合成子事务的平均可用性比率

您可以从“**按位置划分的事务详细信息**”报告中访问四个额外报告，即“**按事务位置划分的实时分析**”、“**按事务位置划分的 HTTP 度量值**”、“**按子事务位置划分的实时分析**”和“**按子事务位置划分的 HTTP 度量值**”。

“**按事务位置划分的实时分析**”显示在设定日期范围内按设定时间间隔获取的按位置划分的合成事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的按位置划分的选定合成事务 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内按位置划分的选定合成事务的 HTTP 度量值

“**按事务位置划分的 HTTP 度量值**”显示在设定日期范围内按设定时间间隔获取的按位置划分的合成事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的按位置划分的选定合成事务 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内按位置划分的选定合成事务的 HTTP 度量值

“**按子事务位置划分的实时分析**”显示在设定日期范围内按设定时间间隔获取的按位置划分的子事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的按位置划分的选定子事务 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内按位置划分的选定子事务的 HTTP 度量值

“**按子事务位置划分的 HTTP 度量值**”显示在设定日期范围内按设定时间间隔获取的按位置划分的合成子事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的按位置划分的选定子事务 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内按位置划分的选定子事务的 HTTP 度量值

按子事务划分的事务详细信息

这是一个两页报告，显示在设定日期范围内按设定时间间隔获取的合成子事务的响应时间和可用性比率。

第一页显示以下数据：

· 折线图：显示在设定日期范围内按设定时间间隔获取的选定合成子事务的响应时间

· 表：以秒为单位显示设定日期范围内每个合成子事务的平均响应时间

第二页显示以下数据：

· 折线图：显示在设定日期范围内按设定时间间隔获取的选定合成子事务的可用性比率

· 表：显示设定日期范围内每个合成子事务的可用性比率

您可以从“**按子事务划分的事务详细信息**”报告访问两个额外的报告，即“**按子事务划分的实时分析**”和“**按子事务划分的 HTTP 度量值**”。

“**按子事务位置划分的实时分析**”显示在设定日期范围内按设定时间间隔获取的按位置划分的子事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的选定子事务的 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内选定子事务的 HTTP 度量值

“**按子事务位置划分的 HTTP 度量值**”显示在设定日期范围内按设定时间间隔获取的按位置划分的合成子事务 HTTP 度量值。该报告包含以下项目：

· 柱形图：显示在设定日期范围内按设定时间间隔获取的选定子事务的 HTTP 度量值

· 表：以毫秒为单位显示设定日期范围内选定子事务的 HTTP 度量值

事务趋势

这是一个四页报告，显示上周以及前 5 周的响应时间、可用性比率和 HTTP 度量值的趋势分析。

第一页显示有关合成事务的响应时间和可用性比率的趋势数据：

· 合并折线图：显示上周以及前 5 周的选定合成事务的平均响应时间

· 合并折线图：显示选定合成事务的可用性比率，比较的是上周可用性比率与前 5 周的可用性比率基线

· 表：显示上周以及前 5 周的日期范围内，合成事务的平均响应时间和可用性比率

· 合并折线图：显示上周以及前 5 周按位置划分的选定合成事务的平均响应时间

· 合并折线图：显示按位置划分的选定合成事务的可用性比率，比较的是上周可用性比率与前 5 周的可用性比率基线

· 表：显示上周以及前 5 周按位置划分的选定合成事务的平均响应时间和可用性比率

· 合并折线图：显示上周以及前 5 周选定合成事务的子事务的平均响应时间

· 合并折线图：显示选定合成事务的子事务的平均可用性比率，比较的是上周可用性比率与前 5 周的可用性比率基线

· 表：显示上周以及前 5 周选定合成事务的子事务的平均响应时间和可用性比率

第二页显示有关合成事务的 HTTP 度量值的趋势数据：

· 合并折线图：显示上周以及前 5 周的选定合成事务的平均阻止时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均 DNS 时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均 SSL 时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均连接时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均发送时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均接收时间
· 合并折线图：显示上周以及前 5 周的选定合成事务的平均呈现时间
· 表：显示上周以及前 5 周的合成事务的平均 HTTP 度量值

第三页显示以毫秒表示、按位置划分的合成事务 HTTP 度量值的相关趋势数据。以下图表和表比较的是上周平均 HTTP 度量值与前 5 周的平均度量值基线：

· 七个合并折线图：比较选定合成事务按位置划分的上周不同平均 HTTP 度量值和前 5 周的度量值基线
· 表：显示选定合成事务按位置划分的上周以及前 5 周的平均 HTTP 度量值

第四页显示以毫秒表示的子事务 HTTP 度量值的相关趋势数据。以下图表和表比较的是上周平均 HTTP 度量值与前 5 周的平均度量值基线：

· 七个合并折线图：比较选定合成事务按子事务划分的上周不同平均 HTTP 度量值和前 5 周的度量值基线
· 表：显示选定合成事务按子事务划分的上周以及前 5 周的平均 HTTP 度量值

子事务趋势

这是一个两页报告，显示子事务上周以及前 5 周的响应时间、可用性比率和 HTTP 度量值的趋势分析。

第一页显示子事务上周日、上周以及前 5 周的响应时间和可用性比率的相关趋势数据：

· 表：比较子事务的上周日、上周和前 5 周的响应时间和可用性比率
· 合并折线图：比较子事务上周平均响应时间和前 5 周的响应时间基线
· 合并折线图：比较子事务上周平均可用性比率和前 5 周的可用性比率基线
· 表：显示子事务的上周平均响应时间和可用性比率。
· 表：显示子事务的前 5 周平均响应时间和可用性比率。

第二页显示以毫秒表示的子事务 HTTP 度量值的相关趋势数据。以下图表和表显示子事务的上周平均 HTTP 度量值和前 5 周的平均度量值基线：

· 七个表：显示选定事务按子事务划分上周日、上周和前 5 周的平均 HTTP 度量值
· 七个合并折线图：比较选定事务以毫秒表示、按子事务划分的上周不同平均 HTTP 度量值和前 5 周可用性比率基线
· 表：显示子事务的上周平均 HTTP 度量值。
· 表：显示子事务的前 5 周平均 HTTP 度量值。

过程

要生成报告，请完成下列步骤：

1. 单击**性能**图标 ，然后选择**应用程序性能仪表板**。要选择应用程序，请展开**我的所有应用程序**并选择应用程序。要显示与所选应用程序关联的所有合成事务，请单击**组 > 事务 > 合成事务**。
2. 从名为“事务列表”的表中选择合成事务。要运行报告，请单击**操作 > 启动到报告**，然后选择下列其中一个报告：
   · **事务总体情况**
   · **按位置划分的事务详细信息**
   · **按子事务划分的事务详细信息**
   · **事务趋势**
   · **子事务趋势**

“配置”页面将在 Web 浏览器中的新选项卡中打开。

3. 要设置报告的日期范围，请选择预定义的日期范围，或者输入定制日期范围。
4. 要设置报告的时间间隔，请从**时间类型**中选择时间间隔。将报告设置为在设定日期范围内按**每小时**、**每天**或**每星期**的时间间隔显示合成事务和子事务的数据。要生成报告，请单击**完成**。
5. 要查看有关事务、子事务或位置的 HTTP 度量值的报告，必须在**事务总体情况**、**按位置划分的事务详细信息**或**按子事务划分的事务详细信息**报告中选择事务、子事务或位置。
   - 要查看“**按事务划分的实时分析**”，右键单击**事务总体情况**报告中的事务名称，并选择**转至 > 按时间显示的 HTTP 度量值分析**。
   - 要查看“**按事务划分的 HTTP 度量值**”，右键单击**事务总体情况**报告中的事务名称，并选择**转至 > HTTP 度量值汇总**。
   - 要查看“**按事务位置划分的实时分析**”，右键单击**按位置划分的事务详细信息**报告中的事务名称，并选择**转至 > 按时间显示的 HTTP 度量值分析**。
   - 要查看“**按事务位置划分的 HTTP 度量值**”，右键单击**按位置划分的事务详细信息**报告中的事务名称，并选择**转至 > HTTP 度量值汇总**。
   - 要查看“**按子事务位置划分的实时分析**”，右键单击**按位置划分的事务详细信息**报告中的子事务名称，并选择**转至 > 按时间显示的 HTTP 度量值分析**。
   - 要查看“**按子事务位置划分的 HTTP 度量值**”，右键单击**按位置划分的事务详细信息**报告中的子事务名称，并选择**转至 > HTTP 度量值汇总**。
   - 要查看“**按子事务划分的实时分析**”，右键单击**子事务趋势**报告中的子事务名称，并选择**转至 > 按时间显示的 HTTP 度量值分析**。
   - 要查看“**按子事务划分的 HTTP 度量值**”，右键单击**子事务趋势**报告中的子事务名称，并选择**转至 > HTTP 度量值汇总**。

WebSphere Applications 代理程序报告

针对 WebSphere Applications 代理程序收集的数据提供了预定义报告。

报告数据存储在 WAREHOUS Db2 数据库中。报告显示分别针对 1 个月、3 个月、1 年和 1 年保留的每小时、每天、每周和每月数据。Cloud APM 不提供用于变更这些保留期的脚本或指示信息。针对 WebSphere Applications 代理程序收集的数据提供了以下报告：

应用程序请求性能

描述

此报告在汇总级别分析应用程序服务器中的应用程序性能。通过饼图显示应用程序的汇总级别请求。通过条形图显示汇总级别的应用程序平均响应时间。通过两个时间序列折线图显示所有应用程序的平均响应时间和总请求计数趋势。要向下钻取至应用程序的单个请求，单击饼图分区或某个条形。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器类型”

使用的表

Request_Analysis_*V

DB 连接池

描述

此报告分析应用程序服务器中的数据库连接池。该表显示汇总级别所有连接池的关键统计信息。如果您选择特定的数据源，将通过两个趋势图显示关键统计信息的趋势。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器名称”

使用的表

DB_Connection_Pools_*V

EJB 性能

描述

此报告分析应用程序服务器中部署的 EJB 的性能。通过饼图显示 EJB 的汇总级别方法计数。通过条形图显示汇总级别的 EJB 平均方法响应时间。通过两个时间序列图显示所有 EJB 的方法调用计数趋势和平均方法响应时间趋势。通过单击列表中的行，可以按每个 EJB 过滤趋势线。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器名称”

使用的表

Enterprise_Java_Beans_*V

应用程序服务器的 GC 使用情况

描述

此报告分析垃圾回收。使用此报告确定垃圾回收是否会导致问题，或者堆的大小设置是否不正确。第一个图形显示使用的平均堆百分比以及随着时间变化的垃圾回收平均实时百分比。第二个图形显示垃圾回收运行的平均实时百分比和平均垃圾回收速率。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器类型”

使用的表

Garbage_Collection_Analysis_*V

应用程序服务器的 JVM 使用情况

描述

此报告分析应用程序服务器的 JVM 性能。通过堆积条形图显示 JVM 内存的使用和释放情况。通过双线折线图显示 JVM CPU 使用率和 JVM 内存使用率。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器类型”

使用的表

Application_Server_*V

线程池

描述

此报告分析应用程序服务器中的线程池。该表显示汇总级别所有线程池的关键统计信息。从列表中选择线程池之后，趋势图将显示所选线程池的关键统计信息的趋势。如果没有选择线程池，趋势将显示所有线程池的摘要。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间

必需参数：“摘要类型”和“应用程序服务器类型”

使用的表

Thread_Pools_*V

Web 应用程序性能

描述

此报告分析应用程序在应用程序服务器的 Web 容器中的性能（PMI 数据）。通过饼图显示应用程序的汇总级别请求。通过条形图显示汇总级别的应用程序平均响应时间。通过两个时间序列折线图显示所有应用程序的平均响应时间和总请求计数趋势。单击饼图分区/条形/线条可向下钻取至该应用程序的单个 servlet/jsps。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序服务器类型”

使用的表

Thread_Pools_*V

集群的应用程序请求性能

描述

此报告分析集群中的服务器性能。第一个图表显示所选时间间隔中每个集群成员完成的请求数量。第二个图表提供有关每个集群成员的平均响应时间趋势的信息。第三个图表中针对集群中的每个服务器都有单独的线条。单击线条可以向下钻取至单项服务器数据。这样会打开此服务器的“应用程序请求性能”报告

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“集群名称”

使用的表

Request_Analysis_*V

集群的 JVM 和 GC 使用情况

描述

此报告分析每个集群成员的 JVM 和垃圾回收使用情况趋势。第一个图表显示垃圾回收运行的平均实时百分比。第二个图表显示使用的平均堆百分比。最后一个图表显示 CPU 和 JVM 内存使用情况。所有这些图表将每个集群成员的数据显示为单独的一条线。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“集群名称”

使用的表

Garbage_Collection_Analysis_*V 或 Application_server_*V

在服务器中响应时间最慢的若干应用程序

描述

此报告在汇总级别分析所有应用程序服务器中的应用程序性能。通过条形图显示汇总级别的应用程序平均响应时间。

参数

日期范围：选择其中一个预定义的报告时段或从日历中选择准确的开始和结束时间。

必需参数：“摘要类型”和“应用程序数量”

第 11 章 升级

升级 代理程序和数据收集器以获得当前发行版中提供的最新功能部件和功能。

升级代理程序

将定期提供包含升级监视代理程序的新归档文件以供下载。这些归档文件位于以下位置：IBM Marketplace Web 站点上的 产品和服务。

开始之前

对于以下代理程序，必须先完成特定于代理程序的任务，然后再完成升级过程：

- 对于 AIX 上的代理程序，如果要以非 root 用户身份运行，必须从内存中清除其中一个库，然后再启动安装过程以升级代理程序。遵循第 964 页的『AIX 上的代理程序：在升级前停止代理程序并运行 slibclean』中的指示信息。
- 对于 HMC Base 代理程序 on AIX，如果您要以非 root 用户身份升级代理程序，首先必须停止 HMC Base 代理程序，然后从高速缓存中清除从属库。遵循第 964 页的『AIX 上的 HMC Base 代理程序：以非 root 用户身份停止代理程序并运行 slibclean，然后再进行升级』中的指示信息
- 针对 Microsoft .NET 代理程序，您必须在升级代理程序前从 .NET 应用程序中移除数据收集器。遵循第 966 页的『Microsoft .NET 代理程序：升级之前移除 .NET 数据收集器』中的指示信息。
- 针对 Node.js 代理程序，您必须在升级代理程序前从 Node.js 应用程序中移除数据收集器插件。遵循第 965 页的『Node.js 代理程序：升级之前移除数据收集器插件』中的指示信息。
- 对于 Ruby 代理程序，您必须在升级代理程序之前从 Ruby 应用程序中移除数据收集器。遵循第 967 页的『Ruby 代理程序：升级之前移除数据收集器插件』中的指示信息。
- 对于 HTTP Server 代理程序，您必须在升级代理程序前停止 HTTP Server。
- 对于 WebSphere MQ 代理程序，如果在前一个发行版中为代理程序启用了事务跟踪，必须先停止该代理程序实例，然后再升级代理程序。
- 对于 SAP NetWeaver Java Stack 代理程序，如果要从 V8.1.3.2 升级到 V8.1.4，请先停止所有配置有数据收集器的 SAP NetWeaver Java Stack 实例，再升级代理程序。
- 对于 Skype for Business Server 代理程序，如果从旧版本升级到 V8.1.4.0.2，那么在代理程序端，代理程序名称会更改为 Skype for Business Server。另外，在通过 SDA 升级支持之后，必须重新启动 APMUI 服务才能在 MIN 服务器端反映新的代理程序名称 (Skype for Business Server)，否则“MIN 服务器”仪表板会显示旧的代理程序名称 (MS Lync Server)。
- 对于 Tomcat 代理程序，如果要在 Windows 上升级 TEMA 核心框架，您必须停止代理程序和服务器。遵循 Tomcat 代理程序中的指示信息：在 Windows 上升级 TEMA 核心框架

关于此任务

如果新版本代理程序可用，那么运行安装脚本将自动升级代理程序。如果代理程序没有新版本可用，那么将显示一条消息，说明代理程序已经安装；已安装的代理程序不受影响。

要安装升级代理程序，请使用以下过程：

过程

- 第 102 页的『在 UNIX 系统上安装代理程序』
- 第 107 页的『在 Linux 系统上安装代理程序』
- 第 114 页的『在 Windows 系统上安装代理程序』

© Copyright IBM Corp. 2014, 2019

结果

代理程序将升级到最新版本。如果较新版本的监视代理程序不可用，那么将显示一条消息，说明代理程序已安装；已安装的代理程序不受影响。

下一步做什么

在升级 Windows 代理程序后，必须重新启动并非由 Windows Installer 自动配置和启动的任何代理程序。运行以下命令以检查代理程序状态：

```
./name-agent.bat status
```

使用以下某个方法启动此代理程序：

- 单击**开始 > 所有程序 > IBM Monitoring 代理程序 > IBM Performance Management**。右键单击一个代理程序，然后单击**开始**。
- 运行以下命令：

```
./name-agent.bat start
```

有关监视代理程序命令的信息，包括要使用的名称、如何检查代理程序状态以及其他信息，请参阅第 150 页的『使用代理程序命令』。有关哪些代理程序将会自动启动以及哪些需要手动启动的信息，请参阅 第 95 页的『第 5 章 代理程序 以及数据收集器部署』

- 对于 Hadoop 代理程序，从基于套接字的代理程序（8.1.2 FP2 或更早版本）升级到基于 REST API 的代理程序（8.1.3 或更高版本）之后，请完成下列步骤：

  1. 要阻止生成不必要的日志，请从所有 Hadoop 节点的 `hadoop-metrics2.properties` 文件中移除 17 行代码。
  2. 停止 Hadoop 服务。
  3. 删除代理程序安装程序从 Hadoop 集群中的所有节点复制的 `Plugin.jar` 文件。
  4. 启动 Hadoop 服务。

  有关 17 行代码和 `Plugin.jar` 文件的信息，请参阅配置 Hadoop 节点。
- 对于 HMC Base 代理程序，在将代理程序从 V6.2.2.6 升级到 V6.2.2.7 之后，必须重新配置代理程序并重新启动代理程序。有关指示信息，请参阅第 226 页的『配置 HMC Base 监视』。
- 对于 HTTP Server 代理程序，如果将代理程序从低于 1.0.0.4 的版本升级到 1.0.0.4 或更高版本，还必须更新 HTTP Server 所使用的 `.conf` 文件，以用新生成的文件替换先前的数据收集器配置文件。还必须将新的代理程序实例添加到控制台。有关指示信息，请参阅第 231 页的『配置 HTTP Server 监视』。
- 对于 Microsoft .NET 代理程序，在升级代理程序之后，配置数据收集器。有关指示信息，请参阅第 450 页的『注册数据收集器』。

  如果已将代理程序安装到新目录中，那么必须使用服务控制器 (sc) 命令来更改概要分析程序服务 bin 路径。例如，

  ```
  sc \\localhost config DotNetProfilerService binPath=
  "${install_dir}\qe\bin\DotNetProfilerService.exe
  ```

  其中，*install_dir* 是新安装目录。
- 对于 Node.js 代理程序，在升级代理程序之后，配置代理程序数据收集器。有关指示信息，请参阅第 505 页的『配置 Node.js 代理程序』。
- 对于 OpenStack 代理程序，要进一步配置代理程序以使用 OpenStack 身份 API V3，请重新配置所有代理程序实例，并更新代理程序数据收集器配置文件。有关指示信息，请参阅第 967 页的『OpenStack 代理程序：重新配置代理程序实例以使用 OpenStack 身份 API V3』。
- 对于 Ruby 代理程序，在升级代理程序之后，配置数据收集器。有关指示信息，请参阅第 615 页的『配置诊断数据收集器』。
- 针对 WebSphere Applications 代理程序，在升级代理程序之后，通过从新版本代理程序的安装目录运行命令 *dc_home*`/bin/migrate.sh/bat` 迁移数据收集器，并重新启动应用程序服务器实例。有关指示信息，请参阅第 968 页的『WebSphere Applications 代理程序：迁移数据收集器』。

- Linux 如果要升级在 /opt/ibm/ccm/agent 目录中安装的较旧版本的代理程序，那么必须在 Linux 系统上完成下列步骤：

  1. 如果确认要将代理程序配置从旧安装目录 /opt/ibm/ccm/agent 迁移到新安装目录（例如，/opt/ibm/apm/agent），那么必须在新安装位置启动代理程序。

     限制: 旧安装位置中的较旧版本代理程序将自动停止，但是新安装位置中的代理程序不会自动启动。
  2. 验证代理程序在新安装目录中正常工作后，必须从 /opt/ibm/ccm/agent 目录卸载较旧版本的代理程序。如果想要移除所有代理程序，请运行 **/opt/ibm/ccm/agent/bin/smai-agent.sh uninstall_all** 命令。

- Linux 如果您正在从 FP6 或前发行版升级代理程序，那么在完成代理程序到新目录的升级和代理程序的配置或重新配置之后，您可能想要移除旧安装目录。完成下列步骤：

  1. 在安装一个或多个监视代理程序的 VM 或系统上，启动命令行并切换到旧安装目录中的二进制文件夹 (/opt/ibm/ccm/agent/bin)。
  2. 要从旧安装目录卸载已安装的所有监视代理程序，请输入：./smai-agent.sh uninstall_all
  3. 删除旧安装目录。

保留代理程序配置更改

高级用户可以将覆盖值应用于组件定制。应用覆盖值可确保升级期间保留这些值。请首先在环境中测试更改，然后再全局应用更改。

关于此任务

- 以下指示信息适用于 Linux 和 AIX 代理程序。有关代理程序产品代码的列表以及用于停止和启动代理程序的命令，请参阅第 150 页的『使用代理程序命令』。
- Windows 代理程序进程设计为保留配置更改：k*pc*cma.ini 文件中更新的变量（其中，*pc* 是产品代码）保留在“覆盖本地设置”节中。这些变量在各个配置期间用于更新代理程序运行时使用的 Windows 注册表项。
- .*pc*.environment 文件和 .global.environment 文件中的定制设置在代理程序升级后将丢失。要保留您的设置，请在 *pc*.environment 和 global.environment 文件中进行定制更改。这些文件中的设置不会被代理程序升级覆盖。

过程

执行以下步骤以保存先前对环境文件的配置更改，并在代理程序升级后仍保留这些更改：

1. 根据需要创建或更新以下任何文件，其中，*install_dir* 是代理程序安装目录（例如 Linux 缺省 /opt/ibm/apm/agent/ 目录或 AIX 缺省 /opt/ibm/ccm/agent/ 目录）：

文件名	描述
***install_dir*/config/*pc*.environment**	文件名中的 *pc* 是代理程序产品代码，例如 mq 或 rz。
***install_dir*/config/global.environment**	将您希望影响所有代理程序类型的更改更新到全局环境文件中。

   例如，as.environment 是持久性 WebSphere Applications 代理程序环境文件。.as.environment 在代理程序升级到新版本时将被覆盖。

   以 *key*=*value* 格式定义变量，其中 *key* 是环境变量名，*value* 是值或设置（例如 **KDC_FAMILIES=${KDC_FAMILIES}HTTP:10001**）。
2. 在完成变量设置更新之后，请保存并关闭环境文件，然后重新启动受影响的代理程序。

结果

这些更新已应用到同一类型的所有代理程序，或者如果更新了全局环境文件，将应用到向 Cloud APM 服务器报告的所有代理程序。这些更改会随着代理程序版本升级持久存储。

AIX 上的代理程序：在升级前停止代理程序并运行 slibclean

如果要在 AIX 系统上以非 root 用户身份升级代理程序，那么必须完成本任务。在运行代理程序安装程序之前，必须停止代理程序并运行 **slibclean** 以清除 libkududp.a 库。

过程

1. 根据代理程序是否支持多个实例，运行以下一个命令，停止代理程序：
   - `./name-agent.sh stop`
   - `./name-agent.sh stop instance_name`

   请参阅第 150 页的『使用代理程序命令』。
2. 使用 root 用户特权运行以下命令。

   ```
   slibclean
   ```

   请参阅 IBM Knowledge Center 中的 slibclean Command。

结果

代理程序已停止并且已清除 libkududp.a 库。

下一步做什么

运行代理程序安装程序以将代理程序升级至已下载的发行版。请参阅第 101 页的『第 6 章 安装代理程序』。如果升级失败，请重新引导服务器，然后重复以上过程。

AIX 上的 HMC Base 代理程序：以非 root 用户身份停止代理程序并运行 slibclean，然后再进行升级

在 AIX 上以非 root 用户身份升级 HMC Base 代理程序之前，必须停止 HMC Base 代理程序并运行 **slibclean** 以从高速缓存中清空从属库。

关于此任务

过程

1. 以非 root 用户身份运行以下命令以停止代理程序。

   ```
   hmc_base-agent.sh stop
   ```

2. 使用 root 用户特权运行以下命令。

   ```
   slibclean
   ```

   请参阅 IBM Knowledge Center 中的 slibclean Command。

结果

HMC Base 代理程序已停止，并清除了从属库。

下一步做什么

运行代理程序安装程序以升级 HMC Base 代理程序。

Node.js 代理程序：升级之前移除数据收集器插件

升级 Node.js 代理程序 之前，必须从 Node.js 应用程序中移除监视插件。

关于此任务

基于 Node.js 代理程序版本，您需要完成不同的过程以从 Node.js 应用程序中移除监视插件。要找到代理程序版本，请参阅代理程序版本命令。

过程

1. 从 Node.js 应用程序文件开头移除数据收集器插件。
   - 如果将 Node.js 代理程序从 V01.00.12.00 升级到 V01.00.13.00，请完成以下过程：
     - 如果已启用资源数据收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_index.js');
       ```

       其中，*KNJ_NPM_LIB_LOCATION* 为 npm 包全局安装目录的 `lib` 文件夹目录。缺省目录为 `/usr/local/lib`。
     - 如果已启用资源数据收集和深入诊断数据收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_deepdive.js');
       ```

     - 如果已启用资源数据收集、深入诊断数据收集和方法跟踪收集，请从 Node.js 应用程序文件开头移除以下行：

       ```
       require('KNJ_NPM_LIB_LOCATION/node_modules/ibm-apm/knj_methodtrace.js');
       ```

   - 如果将 Node.js 代理程序从 V01.00.10.00 升级到 V01.00.13.00，请完成以下过程：
     - 如果已启用资源数据收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_index.js');
       ```

       ，其中，*install_dir* 是 Node.js 代理程序的安装目录。
     - 如果已启用资源数据收集和深入诊断数据收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_deepdive.js');
       ```

     - 如果已启用资源数据收集、深入诊断数据收集和方法跟踪收集，请从 Node.js 应用程序文件开头移除以下行。

       ```
       require('install_dir/lx8266/nj/bin/plugin/knj_methodtrace.js');
       ```

2. 重新启动 Node.js 应用程序以禁用数据收集器插件。
   - 如果当前 Node.js 代理程序版本为 V01.00.10.00，那么到目前为止，已成功移除数据收集器插件。
   - 如果当前 Node.js 代理程序版本为 V01.00.12.00，那么继续以下步骤。
3. 从 *install_dir*`/lx8266/nj/bin` 目录运行 `./uninstall.sh` 命令以移除先前的代理程序设置。

下一步做什么

升级 Node.js 代理程序。请参阅第 961 页的『升级代理程序』。

Response Time Monitoring 代理程序：升级 IBM HTTP Server 响应时间模块

如果您先前使用 IBM HTTP Server 响应时间模块或 HTTP Server 代理程序来监视 IBM HTTP Server，请升级安装。

关于此任务

下表显示可能与监视 IBM HTTP Server 的方式类似的一些安装场景。

Response Time Monitoring 代理程序	是否使用 IBM HTTP Server 响应时间模块	是否使用 Packet Analyzer	是否安装 HTTP Server 代理程序
AIX 和 xLinux: 08.11.00 和更高版本 Windows: 08.14.02 和更高版本	✓	–	✓
AIX 和 xLinux: 08.10.00	✓	–	–
AIX 和 xLinux: 08.10.00	✓	–	✓
7.40.07 或更低版本	–	✓	–
7.40.07 或更低版本	–	✓	✓

对于所有这些场景，安装过程类似：

过程

1. 在 AIX 或 Linux 上从 V8.1.1 或更高版本安装 HTTP Server 代理程序；在 Windows 上从 V8.1.4.02 或更高版本进行安装。

   IBM HTTP Server 响应时间模块自动随代理程序进行安装。
2. 配置 HTTP Server 代理程序。

   注: 如果先前使用 IBM HTTP Server 响应时间模块，那么使用新的 IBM HTTP Server 响应时间模块位置更新 Web 服务器配置文件 (`httpd.conf`)，并移除旧的装入模块配置文件 (`mod_wrt.so`)。

   注: Response Time Monitoring 代理程序 V8.1.1 和更高版本不能使用先前发行版的装入模块文件 (`mod_wrt.so`)。如果您尝试使用此文件的旧版本，会产生错误日志消息。可能仍然会跟踪事务，但不会显示事务实例数据。

   有关更多信息，请参阅 HTTP Server 代理程序参考 PDF（可从 http://ibm.biz/agent-httpserver 下载）。
3. 确保 IBM HTTP Server 和 HTTP Server 代理程序正在运行。如果 Response Time Monitoring 安装程序检测到 HTTP Server 代理程序，Response Time Monitoring 代理程序会启用 IBM HTTP Server 响应时间模块，而不是 Packet Analyzer。
4. 将 Response Time Monitoring 代理程序与 HTTP Server 代理程序安装到同一位置 `AGENT_HOME`。
   - Linux AIX 以 **root** 用户身份安装 V8.1.1 或更高版本。`AGENT_HOME` 示例：`/opt/ibm/apm/agent/`
   - Windows 使用管理员权限安装 V8.1.4.0.2 或更高版本。`AGENT_HOME` 示例：`C:\IBM\APM\`。
5. 如果您使用了较早版本的 Packet Analyzer，那么您可能必须禁用 Packet Analyzer 才能开始使用 IBM HTTP Server 响应时间模块 监视 IBM HTTP Server。
6. 重新启动 IBM HTTP Server。

Microsoft .NET 代理程序：升级之前移除 .NET 数据收集器

在升级 Microsoft .NET 代理程序 之前，您必须从 .NET 应用程序移除 .NET 数据收集器。

过程

1. 注销数据收集器的所有模块。

   作为管理员，输入：

   ```
   cd install_dir\qe\bin configdc unregisterdc all
   ```

   其中，*install_dir* 是 Microsoft .NET 代理程序的安装目录。
2. 重新启动 .NET 应用程序。

下一步做什么

升级 Microsoft .NET 代理程序。请参阅第 961 页的『升级代理程序』。

OpenStack 代理程序：重新配置代理程序实例以使用 OpenStack 身份 API V3

要升级 OpenStack 代理程序 以使用 OpenStack 身份 API V3，在安装最新版本的代理程序之后，必须重新配置所有代理程序实例，并更新数据收集器配置文件。

关于此任务

仅当您要升级代理程序以使用 OpenStack 身份 API V3 时，才必须执行此任务。

过程

1. 重新配置全部现有的代理程序实例。有关详细的指示信息，请参阅第 521 页的『配置 OpenStack 代理程序』。
2. 查找 `ksg_dc_instance_name.cfg` 代理程序数据收集器配置文件，其中 *instance_name* 是您对代理程序实例指定的名称。

   如果该文件不存在，将 *install_dir*`/lx8266/sg/bin/ksg_dc.cfg` 复制到 *install_dir*`/config` 目录，并将文件名更改为 `ksg_dc_instance_name.cfg`。

   例如，如果实例名称为 `OS1`，请将名称更改为 `ksg_dc_OS1.cfg`。
3. 在 `ksg_dc_instance_name.cfg` 文件中添加下面这一节：

   ```
   #OpenStack authentication information
   [OS_authentication_info]
   OS_project_domain_name=Default
   OS_user_domain_name=Default
   OS_cert_path=
   ```
4. 通过运行下列命令，重新启动代理程序实例：

   ```
   install_dir/bin/openstack-agent.sh stop instance_name
   install_dir/bin/openstack-agent.sh start instance_name
   ```

   其中，*instance_name* 是要配置的代理程序实例的名称。

Ruby 代理程序：升级之前移除数据收集器插件

升级 Ruby 代理程序之前，必须从 Ruby 应用程序中移除监视插件。

过程

1. 运行以下命令，移除旧版本的数据收集器。

   ```
   gem uninstall stacktracer
   ```
2. 浏览到应用程序的主目录，打开其 `Gemfile`，并移除以下行：`gem 'stacktracer', 'version'` 其中，*version* 是 Ruby 代理程序的版本号。
3. 在应用程序的主目录中，输入：`bundle install`

下一步做什么

升级 Ruby 代理程序。请参阅第 961 页的『升级代理程序』。

WebSphere Applications 代理程序：迁移数据收集器

在更新代理程序之后，您必须以交互方式或静默方式迁移数据收集器。

以交互方式迁移数据收集器

您可以使用迁移实用程序以交互方式迁移数据收集器的先前维护级别。

开始之前

Linux AIX 如果使用非 root 用户帐户安装了 WebSphere Application Server 或 WebSphere Portal Server，那么在运行配置实用程序之前，请验证此非 root 用户是否具有对 *install_dir*/yndchome/7.3.0.14.08 中以下代理程序目录的读写特权，其中，*install_dir* 是 WebSphere Applications 代理程序的安装目录：

- data
- bin
- runtime
- logs

如果需要，请使用 chmod 777 命令提供读和写许可权。此外，以用于安装应用程序服务器的用户身份登录。

关于此任务

您可以使用迁移实用程序以交互方式迁移数据收集器的先前维护级别。如果您希望迁移很多应用程序服务器实例，那么以静默方式使用迁移实用程序可能更为方便。

要点:

- 您只能迁移数据收集器的 7.3 版本的先前维护级别。数据收集器版本在数据收集器主目录路径中指示。
- 无法将数据收集器从 V7.3 迁移到 V7.3 FP1。请改为取消配置数据收集器，并卸载代理程序 V7.3。然后安装代理程序 V7.3 FP1，并再次配置数据收集器。

过程

1. Linux AIX 以用于安装应用程序服务器的用户身份登录。
2. 从最新版本的代理程序的安装目录启动迁移实用程序。

   Linux AIX 运行命令 *dc_home*/bin/migrate.sh

   Windows 运行命令 *dc_home*\bin\migrate.bat
3. 该实用程序显示在本地计算机系统上找到的所有网卡的 IP 地址。

   输入与要使用的 IP 地址对应的数字。
4. 此实用程序发现数据收集器的较旧维护级别所配置的所有服务器，并列示这些服务器。数据收集器将按维护级别分组。

   从列表选择一个或多个应用程序服务器实例。

   此列表可能同时包含了传统 WebSphere 服务器实例和 Liberty 服务器。传统 WebSphere 服务器实例可能位于其他概要文件下。

   提示:

   - 如果监视了一个概要文件下的多个实例，那么必须同时选择这些实例进行迁移。
   - 同时迁移 Liberty 概要文件下的所有服务器。部分迁移配置的服务器可能会导致不稳定。

   切记:

   - 针对单机环境，应用程序服务器实例必须正在运行。
   - 针对 Network Deployment 环境，Node Agent 和 Deployment Manager 必须正在运行。
   - 迁移期间，不需要运行 Liberty 服务器。

5. 输入与将要迁移其数据收集器的应用程序服务器实例对应的数字，或者输入星号 (*) 以迁移所有应用程序服务器实例的数据收集器。

   要指定服务器的子集，请输入表示服务器的数字（以逗号隔开）。例如：1,2,3。

   迁移实用程序自动将每个数据收集器与监视代理程序集成。监视代理程序主机和端口值将从现有配置文件进行检索。
6. 输入每个所选服务器的别名。

   缺省值为现有服务器别名。
7. 对于 Liberty 服务器实例，请在提示时输入 JVM 主目录。例如，/opt/IBM/java。
8. 该实用程序确定是否为在其中迁移数据收集的每个概要文件启用了 WebSphere 全局安全性。

   如果为一个或多个概要文件启用了 WebSphere 全局安全性，请指定是否从客户机属性文件检索安全设置：

   数据收集器使用 RMI 或 SOAP 协议与 WebSphere Administrative Services 进行通信。如果为概要文件启用了全局安全性，那么必须指定有权登录到概要文件的 IBM WebSphere Application Server 管理控制台的用户的用户标识和密码。

   或者，您可以在配置数据收集器之前对用户名和密码进行加密，并将其存储在应用程序服务器客户机属性文件中。必须使用 sas.client.props 文件来进行 RMI 连接或者使用 soap.client.props 文件来进行 SOA 连接。
9. 输入 1 以使实用程序从相应的客户机属性文件检索用户名和密码并跳至步骤 第 969 页的『11』。否则，输入 2 以输入用户名和密码。

   要点: 登录到 WebSphere Application Server 管理控制台可能需要一些时间。
10. 不管是否启用了 WebSphere 全局安全性，都要针对每个概要文件输入用户名和密码。
11. 该实用程序为每个所选应用程序服务器实例迁移数据收集。将显示一条状态消息，此消息指示是否已成功完成每个服务器的迁移。
12. 按实用程序的指示重新启动这些实例。重新启动应用程序服务器实例之后，数据收集器配置会生效。

结果

数据收集器将迁移到安装的最新维护级别。

下一步做什么

迁移实用程序将配置的设置保存在较旧版本的数据收集器中。要修改这些设置，可以从新数据收集器的 *dc_home*\bin 目录以交互或静默方式运行配置或重新配置实用程序。有关更多信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。

以静默方式迁移数据收集器

您可以使用迁移实用程序以静默方式迁移数据收集器的先前维护级别。

开始之前

Linux AIX 如果使用非 root 用户帐户安装了 WebSphere Application Server 或 WebSphere Portal Server，那么在运行配置实用程序之前，请验证此非 root 用户是否具有对 *install_dir*/yndchome/7.3.0.14.08 中以下代理程序目录的读写特权，其中，*install_dir* 是 WebSphere Applications 代理程序的安装目录：

- data
- bin
- runtime
- logs

如果需要，请使用 chmod 777 命令提供读和写许可权。此外，以用于安装应用程序服务器的用户身份登录。

关于此任务

随迁移实用程序打包了样本静默配置文件 `sample_silent_migrate.txt`。该文件在 *`install_dir`*`/yndchome/7.3.0.14.08/bin` 目录中提供。

在创建静默属性文件时，请记住以下注意事项：

- 文件中以数字符号 (#) 开头的行将视为注释而不予处理。如果行中其他地方使用到该数字符号，将不被视为注释的开始。这意味着您可以在密码或其他地方使用该数字符号。
- 每个属性都在单独的行中描述，格式为：property = value。

  property
  这是属性的名称。第 970 页的表 256 中显示了可以配置的有效属性列表。请勿在样本文件中修改或移除未在表中列出的属性。

  value
  这是属性的值。提供了部分属性的缺省值。您可以删除缺省值，将属性值留为空白。与使用缺省值相反，空值将视为未指定该属性。如果要使用缺省值，在文件中注释掉属性即可。
- 密码为纯文本。
- 属性及其值区分大小写。

第 970 页的表 256 描述以静默方式迁移数据收集器时可用的属性。

表 *256.* 以静默方式运行迁移实用程序时的可用属性

属性	注释
migrate.type	必须为 AD。
default.hostip	如果计算机系统使用多个 IP 地址，请指定供数据收集器使用的 IP 地址。
itcam.migrate.home	指定数据收集器的较旧维护版本的数据收集器主目录。迁移过程中未删除该目录。
was.wsadmin.connection.host	指定 wsadmin 工具正在连接到的主机名。在 Network Deployment 环境中将 wsadmin 连接指定给部署管理器。在单机环境中将 wsadmin 连接指定给该服务器。
was.wsadmin.username	指定有权登录 IBM WebSphere Application Server 管理控制台的用户的用户标识。此用户在应用程序服务器上必须具有代理角色。
was.wsadmin.password	指定与在 `was.wsadmin.username` 属性中指定的用户对应的密码。
was.appserver.profile.name	指定希望配置的应用程序服务器概要文件的名称。 **切记:** 对于 Liberty 概要文件，此属性不是必需的。
was.appserver.home	指定 WebSphere Application Server 主目录。
was.appserver.cell.name	指定 WebSphere Application Server 单元名称。 **切记:** 对于 Liberty 概要文件，此属性不是必需的。
was.appserver.node.name	指定 WebSphere Application Server 节点名。 **切记:** 对于 Liberty 概要文件，此属性不是必需的。
was.appserver.server.name	在应用程序服务器概要文件内指定应用程序服务器实例以迁移到新版本的数据收集器。静默属性文件可具有该属性的多个实例。

要点:

- 您只能迁移数据收集器的 7.3 版本的先前维护级别。数据收集器版本在数据收集器主目录路径中指示。
- 无法将数据收集器从 V7.3 迁移到 V7.3 FP1。请改为取消配置数据收集器，并卸载代理程序 V7.3。然后安装代理程序 V7.3 FP1，并再次配置数据收集器。

过程

1. 在静默迁移属性文件中指定配置选项。
2. 运行命令来从最新版本的代理程序的安装目录以静默方式启动迁移实用程序。
   - Linux AIX *dc_home*/bin/migrate.sh -silent *sample_silent_migration_filename*
   - Windows *dc_home*\bin\migrate.bat -silent *sample_silent_migration_filename*

结果
数据收集器将迁移到安装的最新维护级别。

下一步做什么
迁移实用程序将配置的设置保存在较旧版本的数据收集器中。要修改这些设置，可以从新数据收集器的 *dc_home*\bin 目录以交互或静默方式运行配置或重新配置实用程序。有关更多信息，请参阅第 714 页的『使用完全配置实用程序来配置或重新配置数据收集器』。

Tomcat 代理程序：升级 Windows 上的 TEMA 核心框架

要升级 Tomcat 代理程序的 Windows 上的 TEMA 核心框架，必须同时停止该代理程序和服务器以成功升级 TEMA 框架。

过程

1. 准备 Tomcat 服务器安装。
2. 安装并配置 Tomcat 代理程序。
3. 登录 IBM Cloud Application Performance Management 仪表板，转至**代理程序配置** > **Tomcat**，选择 Tomcat 代理程序实例，然后单击**启用 TT/DD**。
4. 重新启动 Tomcat 服务器。
5. 要应用 IBM APM 核心框架，请同时停止 Tomcat 代理程序和服务器。
6. 转至 TEMA/<IBM APM CORE FRAMEWORK_HOME>。运行命令 **apmpatch.bat <Tomcat Agent Installationdir>**。
   框架已升级。
7. 通过执行以下指示信息检查升级的 IBM APM 核心框架版本。
   Goto <TOMCAT_Agent_Install_Dir>\InstallITM
   Run: KinCInfo.exe -i。
8. 同时启动 Tomcat 服务器和代理程序。

升级数据收集器

将定期提供包含升级数据收集器的新归档文件以供下载。归档文件可以从 IBM Marketplace Web 站点上的产品和服务 获取。

开始之前

关于此任务
要升级数据收集器，请完成以下步骤：

过程

- 从本地和/或 IBM Cloud 应用程序取消配置数据收集器：
  - 对于 J2SE 数据收集器，不需要执行取消配置步骤。
  - 对于 Liberty 数据收集器，根据第 757 页的『取消为 IBM Cloud 应用程序配置数据收集器』和/或第 751 页的『为内部部署应用程序取消配置数据收集器』中的指示信息进行操作。
  - 对于 Node.js 数据收集器，根据第 513 页的『取消为 IBM Cloud 应用程序配置 Node.js 数据收集器』和/或第 517 页的『为本地应用程序取消配置独立 Node.js 数据收集器』中的指示信息进行操作。
  - 对于 Python 数据收集器，根据第 575 页的『取消为 IBM Cloud 应用程序配置 Python 数据收集器』和/或第 580 页的『取消配置内部部署应用程序的 Python 数据收集器』中的指示信息进行操作。
  - 对于 Ruby 数据收集器，根据第 621 页的『取消为 IBM Cloud 应用程序配置 Ruby 数据收集器』中的指示信息进行操作。
- 下载数据收集器软件包。
- 重新配置数据收集器以监视本地和/或 IBM Cloud 应用程序：
  - 对于 Node.js 数据收集器，在升级数据收集器之后，重新配置数据收集器。有关指示信息，请参阅第 509 页的『针对 IBM Cloud（以前称为 Bluemix）应用程序配置独立 Node.js 数据收集器』和/或第 513 页的『为本地应用程序配置独立 Node.js 数据收集器』。
  - 对于 Python 数据收集器，在升级数据收集器之后，重新配置数据收集器。有关指示信息，请参阅第 572 页的『针对 IBM Cloud 应用程序配置 Python 数据收集器』和/或第 576 页的『为本地应用程序配置 Python 数据收集器』。
  - 对于 Liberty 数据收集器，在升级数据收集器之后，重新配置数据收集器。有关指示信息，请参阅第 752 页的『为 IBM Cloud 应用程序配置 Liberty 数据收集器』和/或第 749 页的『为本地应用程序配置 Liberty 数据收集器』。
  - 对于 J2SE 数据收集器，在升级数据收集器之后，重新配置数据收集器。有关指示信息，请参阅第 388 页的『配置 J2SE 监视』。
  - 对于 Ruby 数据收集器，在升级数据收集器之后，重新配置数据收集器。有关指示信息，请参阅第 618 页的『针对 IBM Cloud 应用程序配置 Ruby 数据收集器』。

结果

数据收集器已升级到最新版本。

第 12 章 故障诊断和支持

请查看故障诊断条目，以了解您在安装、配置或使用 IBM Cloud Application Performance Management 时可能遇到的问题。

此 Knowledge Center 中提供故障诊断内容。先前，在 developerWorks 上的 Cloud Application Performance Management 论坛 中提供故障诊断内容。您可以继续在此论坛中搜索较旧的条目。搜索以“Troubleshooting”开头的条目。

有关 IBM Cloud Application Performance Management Hybrid Gateway 故障诊断，请参阅第 819 页的『管理 Hybrid Gateway』。

对代理程序进行故障诊断

对 代理程序安装和配置问题进行故障诊断。

我们正在将故障诊断内容从 developerWorks 中的 Cloud Application Performance Management 论坛 迁移到此 Knowledge Center。先前，在 developerWorks 上的 Cloud Application Performance Management 论坛 中提供故障诊断内容。您可以继续在此论坛中搜索较旧的条目。搜索以“Troubleshooting”开头的条目。

Internet Service Monitoring

您可以在此处查找有关 Internet Service Monitoring 已知问题的更多详细信息。

在“创建概要文件页面”保持打开超过 10 分钟后将不会创建概要文件，并且将不会创建相同名称的另一个概要文件

问题

在“**创建概要文件页面**”保持打开超过 10 分钟后将不会创建概要文件，并且将不会创建相同名称的另一个概要文件。

症状

在创建概要文件时，如果用户保持“**创建概要文件页面**”空闲（打开而无任何活动）超过 10 分钟，然后再尝试创建概要文件，那么将不会创建。此后，如果用户尝试使用相同名称再次创建概要文件，将不会创建概要文件。

原因

创建概要文件时在 MIN 端创建锁定文件，此文件将锁定其他用户的同一概要文件的概要文件创建活动。在概要文件创建完成后，删除锁定文件。但是，如果创建窗口空闲超过 10 分钟，那么创建事件将锁定，并且用户无法创建概要文件。

解决方案

- 在创建概要文件时，用户不应保持窗口空闲超过 10 分钟。
- 管理员可以从 `/opt/ibm/wlp/usr/servers/min/dropins/CentralConfigurationServer.war/data_source/is` 删除在 MIN 端创建的概要文件的锁定文件。
  例如，如果概要文件名称为 ABC，那么将创建锁定文件 `$$ABC$$1UjQ9wy1boIHTAQeoWSj1IU.lock`。

© Copyright IBM Corp. 2014, 2019

Microsoft Active Directory 监视

您可以在此处查找有关 Microsoft Active Directory 监视已知问题的更多详细信息。

Microsoft Active Directory 代理程序不显示更新的联机帮助内容

问题

不会使用最新的 Microsoft Active Directory 代理程序内容更新联机帮助页面。

症状

在 Microsoft Active Directory 代理程序 的 APM 仪表板 Eclipse 帮助上，对于以下新添加的属性组的数据收集时间间隔和保留期，缺少帮助内容：

· 目录服务
· Kerberos 一致性检查程序
· Kerberos 密钥分发中心
· 名称服务提供程序
· 交换目录服务

原因

由于构建服务器中的约束而发生此问题。

解决方案

用户可以在 APM 仪表板上各个属性组上下文帮助中查找帮助内容。

注: 此问题在 APM V8.1.4.10 发行版中出现。

Microsoft IIS 监视

您可以在此处查找有关 Microsoft Internet Information Services 已知问题的更多详细信息。

不会使用 Microsoft IIS APM 代理程序的最新内容更新联机帮助页面

问题

不会使用 Microsoft IIS APM 代理程序的最新内容更新联机帮助页面

症状

联机帮助内容中缺少新添加的属性组：

· WPROCESS
· MEMIISUS
· ASP 垃圾回收
· IISSVRINFO

原因

由于构建服务器问题而发生此问题。

变通方法

不可用。但是，您可以查看 APM 仪表板上特定属性组的帮助内容。

Microsoft .NET 监视

您可以在此处查找有关 Microsoft .NET 监视已知问题的更多详细信息。

Microsoft .NET 代理程序不显示更新的联机帮助内容

问题

不会使用最新的 Microsoft .NET 代理程序 内容更新联机帮助页面。

症状

在 Microsoft .NET 代理程序 的 APM 仪表板 Eclipse 帮助上，“数据库调用详细信息”属性组下缺少“请求名称”属性帮助内容。

原因

由于构建服务器中的约束而发生此问题。

解决方案

用户可以在 APM 仪表板上的“数据库调用详细信息”属性组上下文帮助中查找帮助内容。

注: 此问题在 APM V8.1.4.10 发行版中出现。

Microsoft SharePoint Server 监视

您可以在此处查找有关 Microsoft SharePoint Server 监视已知问题的更多详细信息。

Microsoft SharePoint Server 代理程序不显示更新的联机帮助内容

问题

不会使用最新的 Microsoft SharePoint Server 代理程序 内容更新联机帮助页面。

症状

在 Microsoft SharePoint Server 代理程序 的 APM 仪表板 Eclipse 帮助上，对于名为 `Last 1 Hour Trace Log Count` 和 `Trace Log Details` 的新添加的组窗口小部件，缺少帮助内容。

原因

由于构建服务器中的约束而发生此问题。

解决方案

用户可以在 APM 仪表板上各个组窗口小部件上下文帮助中查找帮助内容。

注: 此问题在 APM V8.1.4.10 发行版中出现。

PostgreSQL 监视

您可以在此处查找有关 PostgreSQL 监视已知问题的更多详细信息。

无活动连接的数据库的“缓冲区命中率百分比”窗口小部件不显示帮助内容

问题

无活动连接的数据库的信息不可用。

症状

不会在“缓冲区命中率百分比”窗口小部件中显示无活动连接的数据库。应该在窗口小部件帮助内容中显示信息。

原因

此限制是由于与 IBM Cloud App Management 的兼容性。

解决方案

不可用。用户应注意限制。

不在 SUSE15 平台上显示内存和 IP 地址值

问题

代理程序在 SUSE15 平台上本地监视 PostgreSQL Server 时，不显示内存和 IP 地址值。

症状

如果代理程序在 SUSE15 平台上监视 PostgreSQL Server 时，那么不显示内存和 IP 地址值。

原因

对于 SUSE15 平台上的代理程序，命令 **netstat** 失败。

解决方案

用户可以使用 SUSE12 平台以本地监视 PostgreSQL Server。

为 IBM 支持人员收集监视代理程序日志

使用问题确定收集工具（*pdcollect*）来收集 IBM 支持人员请求的监视代理程序的所需日志和其他问题确定信息。 PD 收集器工具会与每个监视代理程序一起安装。

开始之前

PD 收集器工具需要 root 或管理员许可权来从监视代理程序收集系统信息。您可以在以下文件夹中分别查看代理程序日志：

- Windows [64 位] *install_dir*\TMAITM6_x64\logs
- Windows [32 位] *install_dir*\TMAITM6\logs
- Linux AIX *install_dir*/logs

限制: 只能运行 pdcollect 脚本的一个实例。

关于此任务

install_dir 的缺省位置是：

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

要运行 PD 收集器工具，请完成下列步骤：

过程

1. 在命令行上，切换到代理程序目录：

- Linux AIX *install_dir*/bin
- Windows *install_dir*\BIN

2. 运行以下命令：
   - Linux AIX **./pdcollect**
   - Windows **pdcollect**

   文件名中具有时间戳记的文件会在 tmp 目录中生成，如 /tmp/pdcollect-nc049021.tar.Z。
3. 将输出文件发送给您的 IBM 支持代表。

下一步做什么

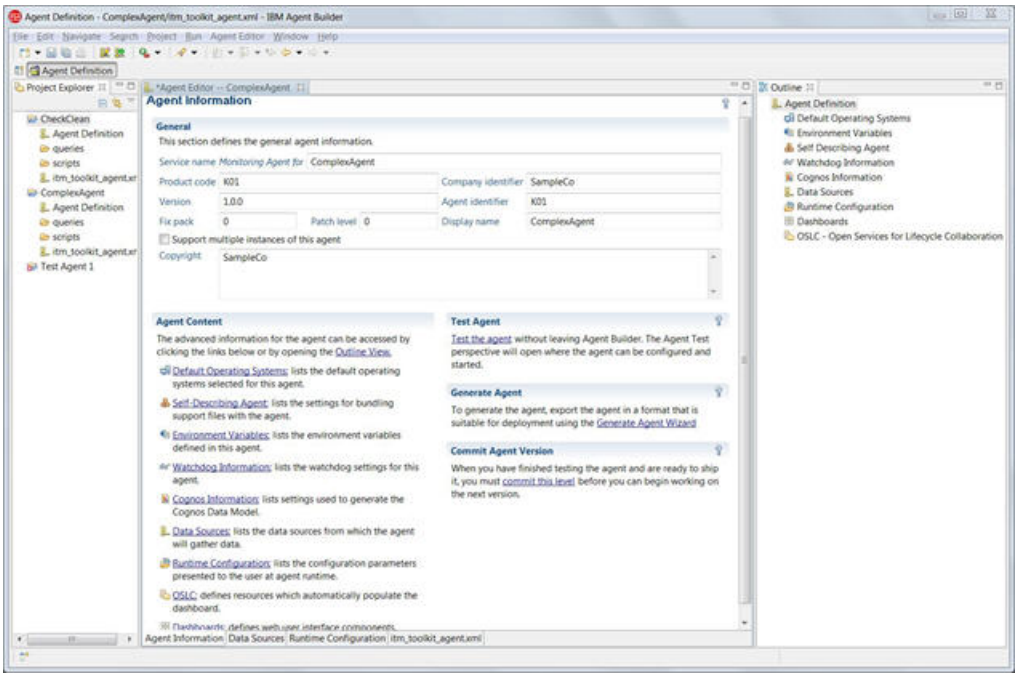
如果您已安装 Ruby 代理程序并已针对诊断仪表板对其进行配置，请在 Linux 系统上运行 kkm 收集器工具 *kkmCollector* 以收集配置文件、输出文件（例如，JSO 文件）和日志文件。

1. 切换到 *install_dir*/lx8266/km/bin 目录。
2. 运行命令 **./kkmCollector**

   文件名中具有时间戳记的文件会在 tmp 目录中生成，如 /tmp/kkm_dchome.tar.gz
3. 将输出文件发送给您的 IBM 支持代表。

第 13 章 Agent Builder

IBM Agent Builder 工具提供一个图形用户界面，以帮助创建、修改、调试和打包代理程序以供监视 IBM Cloud Application Performance Management 中的数据资源。



Agent Builder 概述

您可以使用 IBM Agent Builder 来创建和修改定制代理程序以扩展 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境的监视功能。定制代理程序使用上述任一环境来监视任何类型的内部或定制软件。

Agent Builder 基于 Eclipse，这是一种开放式源代码的集成开发环境。

Agent Builder 包含 Tivoli Monitoring 和 Cloud APM 环境的以下功能部件：

定义和修改代理程序
您可以创建和修改代理程序。代理程序收集并分析不同资源（例如，磁盘、内存、处理器或应用程序）的状态和性能的相关数据，并将此数据提供给监视环境。

测试和准备代理程序以用于部署
您可以在 Agent Builder 中测试代理程序，方法是在 Agent Builder 运行所在的主机上收集数据（某些情况下也可以从其他主机上收集信息）。您可以将代理程序打包以便于分发和部署。

以下其他功能部件可用于 Tivoli Monitoring：

定制工作空间、情境和“执行操作”命令
您可以使用 Agent Builder 对附加工作空间、情境和“执行操作”命令进行打包，以作为在 Tivoli Monitoring 环境中运行的新代理程序或现有代理程序的应用程序支持扩展

报告数据模型
您可以使用 Agent Builder 来生成 Cognos 数据模型，再用这些模型构建 Tivoli Common Reporting 报告。然后，可以将这些报告打包为代理程序映像的组成部分。

© Copyright IBM Corp. 2014, 2019

公共 Agent Builder 过程

下表列出可使用 Agent Builder 完成的主要过程。

您可以使用 Agent Builder 为 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 环境创建代理程序。您还可以用来为 Tivoli Monitoring 环境创建应用程序支持扩展。通过创建工作空间和情境可创建应用程序支持扩展，从而增强一个或多个现有代理程序。

Agent Builder 必须先安装才能使用。有关指示信息，请参阅第 983 页的『安装和启动 Agent Builder』。

要创建、测试和使用代理程序，请按列示顺序完成下表中的过程。

表 *257.* 创建代理程序的快速参考信息

目标	请参阅
使用“**代理程序**”向导创建代理程序。	· 第 988 页的『创建代理程序』
为代理程序创建数据源和属性。 **要点:** 对于 Cloud APM 环境，摘要仪表板最多可以显示大约 5 个属性；其中一个属性必须表示代理程序或子节点的总体状态。	· 第 1003 页的『编辑数据源和特性属性』
对于 Tivoli Monitoring 环境，请为代理程序创建工作空间和情境。 · 至少运行 Tivoli Monitoring V6.1 FP1 · 将 Tivoli Universal Agent 解决方案版本重置为“00” · 设置“AppTag”的值	· 第 1149 页的『创建工作空间、“执行操作”命令和情境』 · 第 1179 页的『导入应用程序支持文件』
对于 Cloud APM 环境，请为代理程序创建资源定义和仪表板。	· 第 1154 页的『为 Cloud APM 准备代理程序』
对于 Tivoli Monitoring 环境，请为代理程序创建用于报告的 Cognos 数据模型。	· 第 1245 页的『生成 Cognos 数据模型』
测试和调试所创建的代理程序，以确保监视信息的可用性。	· 第 1156 页的『在 Agent Builder 中测试代理程序』 · 第 1187 页的『命令行选项』 · 第 990 页的『使用 Agent Editor 修改代理程序』.
生成安装包，并在受监视主机上安装代理程序。	· 第 1163 页的『安装代理程序』
移除使用 Agent Builder 创建的代理程序。	· 第 1177 页的『卸载代理程序』

您还可以使用 Agent Builder 来打包定制工作空间、情境和“执行操作”命令以作为现有代理程序的应用程序支持扩展。这些功能仅可用于 Tivoli Monitoring 环境：

表 *258.* 其他功能的快速参考信息

目标	请参阅
创建定制工作空间、情境和“执行操作”命令。	· 第 1149 页的『创建工作空间、“执行操作”命令和情境』
将应用程序支持扩展打包。	· 第 1243 页的『为现有代理程序创建应用程序支持扩展』
构建定制捆绑包。	· 第 1263 页的『创建非代理程序文件捆绑包』

数据源和数据集

代理程序可以监视来自一个或多个数据源的信息。它以属性形式向监视基础结构呈现信息，属性组织为数据集。

在创建代理程序时，必须为其定义数据源。您可以添加更多数据源。数据源定义代理程序收集监视信息的方式。

您可以使用 Agent Builder 来创建代理程序，以使用监视下列数据提供程序的信息的数据源：

- 进程和服务可用性
- 网络系统可用性（使用 ICMP Ping）
- 命令返回码
- 脚本输出
- Windows 事件日志
- Windows Management Instrumentation (WMI)
- Windows 性能监视器 (Perfmon)
- 简单网络管理协议 (SNMP)
- SNMP 事件
- 超文本传输协议 (HTTP) 可用性和响应时间
- SOAP 或其他 HTTP 数据源
- Java 数据库连接 (JDBC)
- Java 应用程序编程接口 (API)
- Java 管理扩展 (JMX)
- 公共信息模型 (CIM)
- 日志文件
- AIX 二进制日志
- 套接字

您还可以使用其他开发工具来创建定制监视应用程序以通过日志、脚本输出和 Java API 数据源将信息传递给代理程序。

在添加数据源时，Agent Builder 会将对应的数据集添加到代理程序。数据集对呈现给监视环境的信息进行组织。在 IBM Tivoli Monitoring 中，数据集称为属性组。

数据集由若干属性构成，这些属性是由数据源提供的值。监视环境每次查询代理程序时，都会从数据源访存值，然后作为数据集中的属性返回这些值。

某些数据源可以在同一查询中返回若干行属性值（例如，如果数据源同时监视若干个服务）。

大多数数据源将信息呈现为一个数据集。SNMP 和 JMX 数据源根据配置可能会提供不同的信息集。为容纳这些信息，添加 SNMP 或 JMX 数据源时，Agent Builder 会创建多个数据集。

您可以编辑数据集来过滤数据和创建其他派生属性，即，使用公式根据现有属性计算而得的属性。您还可以合并数据集，从而使用两个或两个以上数据集中的信息创建一个新数据集。这样，用户可以查看从不同数据源合并的信息。

在 IBM Tivoli Monitoring 中，您可以查看所有属性内容。还可以创建工作空间，以定制视图呈现来自所有代理程序数据集的信息。您可以使用 IBM Tivoli Monitoring 来创建情境，当任何属性达到某个值时会触发这些情境。情境可以发出警报和调用系统命令。

在 IBM Cloud Application Performance Management 中，必须为代理程序定义摘要仪表板，最多可以选择五个属性显示在该仪表板中。您也可以定义详细信息仪表板，以将来自任何数据集的信息显示为表。您可以创建当任何属性达到某个值时触发的阈值；无需将此属性添加到仪表板。阈值可以发出警报。

监视多个服务器或一个服务器的多个实例

代理程序可以监视多个服务器，包括同一服务器的多个实例。可通过两种方法创建此类代理程序：代理程序的多个实例和代理程序中的若干子节点。

多个实例是监视应用程序服务器的标准方法，这些应用程序服务器在同一主机上拥有若干相似的实例。IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 中的许多标准代理程序支持多个实例。

通过多个实例，您可以在受监视主机上安装代理程序，然后为每个实例设置名称来配置一个或多个实例。为要监视的每个服务器实例配置一个代理程序实例。每个实例都是代理程序的一个独立且相同的副本，可以单独启动和停止。

您还可以在代理程序中定义一种或多种类型的子节点。每个类型所对应的资源类型都必须不同于代理程序能够监视的资源类型。子节点类型包含数据源和数据集；您还可以在任意子节点外部定义代理程序级别的数据源和数据集。在主机上安装代理程序时，可以对每种类型配置所需数量的子节点；对于每种子节点类型，可以独立设置子节点数量。对于 IBM Cloud Application Performance Management，您可以为代理程序创建仪表板，并且可以为每个子节点创建单独的仪表板。

子节点需要在受监视主机上执行不同的配置步骤。此外，要重新配置，需要添加或移除子节点。您必须停止并重新启动整个代理程序；可以在不影响其他实例的情况下重新配置、添加或移除某个实例。但是，子节点有若干优势：

- 通过子节点，您可以在使用较少资源的情况下监视大量服务器实例。有一条准则是，单个系统上支持的特定类型的代理程序实例数量是 10。但是，代理程序使用子节点最多可以监视 100 个本地或远程服务器。
- 一个代理程序可以包含用于几种不同种类的服务器的子节点类型。在受监视系统上，您可以对每种类型配置任意数量的子节点。您可以使用此功能来进一步节省资源。
- 使用子节点的代理程序可以在代理程序级别提供系统范围的数据。

您可以为同一代理程序同时定义多个实例和子节点。在此情况下，每个实例都可以包含若干子节点。您可以独立于其他实例停止和重新启动每个实例；一个实例中的所有子节点会一起停止和重新启动。

测试、安装和配置代理程序

您可以为代理程序创建安装包，然后将其安装在任意数量的受监视主机上。对于某些数据源，需要设置配置值来收集数据。

在为代理程序定义数据源和属性之后，可以在 Agent Builder 中运行该代理程序以进行测试。您可以测试单个数据集（属性组）或整个代理程序。

要扩大代理程序的测试范围并使用代理程序，可以创建安装映像。此映像提供了在任意受监视主机上安装和配置代理程序的脚本。

提示: 在安装代理程序之前，确保在主机上安装您的监视环境（IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management）的操作系统代理程序。

在安装代理程序之后，您需要配置代理程序。如果代理程序支持多个实例，那么必须配置代理程序以创建至少一个实例。

一些数据源需要额外的配置值；例如，对于 SNMP 数据源，必须使用 SNMP 协议配置所监视主机的 IP 地址。使用安装包部署的配置脚本来设置这些值。

或者，可以在创建安装映像之前，在 Agent Builder 中设置这些值。这样，您便无需在受监视主机上再次设置这些值。

提示: 对 Cloud APM 服务器进行升级之后，定制代理程序的帮助文件可能不会显示在帮助内容中。要显示帮助文件，请完成下列步骤：

1. 从 IBM Marketplace 中的 Cloud APM 预订下载最新版本的 IBM Agent Builder。
2. 重新创建定制代理程序。确保在“代理程序信息”页面中指定较高的版本号、修订包或补丁级别。
3. 将该定制代理程序安装在受监视主机上。
4. 在 Cloud APM 控制台中，单击导航栏中的**帮助** > **帮助内容**。这将显示您的定制代理程序帮助。

操作系统需求

Agent Builder 创建的代理程序在各种操作系统上受支持，是否受支持取决于监视环境以及创建代理程序时所选择的设置。

在 Tivoli Monitoring 环境中，Agent Builder 创建的代理程序可支持以下操作系统：

- AIX
- HP-UX
- Linux
- Solaris
- Windows

这些代理程序支持与操作系统代理程序相同的操作系统版本。有关详细信息，请访问 软件产品兼容性报告 Web 站点。搜索 `Tivioli Monitoring` 产品名称，然后选择 `OS Agents & TEMA (Tivoli Enterprise Monitoring Agent)` 组件复选框。

在 IBM Cloud Application Performance Management 环境中，Agent Builder 创建的代理程序可支持以下操作系统：

- AIX
- Linux
- Windows

这些代理程序支持与操作系统代理程序相同的版本。有关详细信息，请使用 System requirements (APM Developer Center) 的“组件报告”部分中的链接。

要在 Tivoli Monitoring 环境中运行监视代理程序，请在运行代理程序的每个受监视系统上安装相应的操作系统代理程序。

要在 IBM Cloud Application Performance Management 环境中运行监视代理程序，请在运行代理程序的每个受监视系统上安装 IBM Cloud Application Performance Management 随附的所有代理程序。

注: Agent Builder 浏览器对可从运行 Agent Builder 的系统访问的数据源和信息进行操作。请确保在以下一种系统上运行 Agent Builder：

- 运行级别与您为其开发代理程序的操作系统和受监视应用程序相同的系统
- 连接到另一个系统的系统，此另一个系统运行级别与您为其开发代理程序的操作系统和受监视应用程序相同

特定于 IBM Tivoli Monitoring 的功能部件

Agent Builder 提供了多个仅适用于 IBM Tivoli Monitoring 的功能部件。

您可以使用导航器组来组织代理程序显示在 IBM Tivoli Monitoring 导航器视图和工作空间中的数据。导航器组将来自若干属性组（数据集）的数据组合到一个视图中，同时向用户隐藏原始独立数据集。

您可以使用 Tivoli Enterprise Portal 为代理程序创建工作空间、情境和“执行操作”命令。然后，可以使用 Agent Builder 将工作空间、情境和“执行操作”命令保存为应用程序支持文件，并将这些文件捆绑到代理程序。此外，Agent Builder 还可以导入其他代理程序的工作空间、情境和“执行操作”命令，并为这些代理程序创建定制应用程序支持文件。

Agent Builder 可以为代理程序生成 Cognos 数据模型。使用此数据模型将代理程序信息导入到 Cognos Framework Manager 中（这是 IBM Tivoli Common Reporting 的一部分）以创建报告。

安装和启动 Agent Builder

在安装 IBM Agent Builder 之前，请确保系统满足先决条件。然后使用安装向导或静默安装过程来安装 Agent Builder。

提示: 有关安装或修改代理程序的信息，请参阅第 1163 页的『安装代理程序』。

安装和运行 Agent Builder 的先决条件

要安装并运行 Agent Builder，您的系统必须符合特定的要求。

要安装 Agent Builder，请确保满足以下条件：

- 至少具有 1 GB 可用磁盘空间的系统。您所开发的代理程序将需要额外磁盘空间。
- 受支持的操作系统。Agent Builder 可运行以下操作系统：
  - Windows Windows
  - Linux Linux（仅限 x86 64 位）
- Linux 如果使用 Linux 操作系统，必须安装 `libstdc++.so.5` 库。您可以安装提供此库的以下包：
  - 在 Red Hat Enterprise Linux 上为 `compat-libstdc++-33`
  - 在 SUSE Enterprise Linux 上为 `libstdc++-33`

Windows 在 Windows 系统上，必须能够以具有 Administrator 许可权的用户身份运行 Agent Builder。这些许可权可以确保 Agent Builder 的环境与用它开发的代理程序一致。

Linux 在 Linux 系统上，可以 root 用户身份或普通用户身份运行 Agent Builder。但是，如果以普通用户身份运行，代理程序测试将受到限制，并且在某些情况下可能不可用。

Agent Builder 的详细系统需求

使用 Software Product Compatibility Reports 来查看 Agent Builder 的详细系统需求。

访问 软件产品兼容性报告 Web 站点。搜索 `IBM Agent Builder` 产品名称。

安装 Agent Builder

您可以使用安装向导或静默安装过程来安装 Agent Builder。

提示: 在安装 Agent Builder 之前，先卸载任何先前版本。有关卸载的更多信息，请参阅（第 987 页的『卸载 Agent Builder』）。卸载时不会丢失任何现有代理程序信息。

使用安装向导来安装 Agent Builder

您可以使用安装向导来安装 IBM Agent Builder。

开始之前

确保您的系统满足先决条件。有关先决条件的信息，请参阅第 984 页的『安装和运行 Agent Builder 的先决条件』

过程

1. 如果您未登录到 IBM Marketplace，请使用您的 IBMid 和密码登录并转至“**产品和服务**”。
   “**产品和服务**”页面适用于活动订户。如果您有任何问题，请转至 Cloud Application Performance Management 论坛或 Marketplace 支持。
2. 下载 Agent Builder 安装归档文件：
   a) 在 Cloud APM 预订框中，单击**管理** > **下载**。
   b) 选择**多平台**作为操作系统。
   c) 选择 IBM Agent Builder 程序包。
   d) 单击**下载**并将 `IBM_Agent_Builder_Install.tar` 保存到系统。
3. 解压缩安装归档文件。
4. 在解压缩的映像目录中使用以下命令来启动安装：
   - Windows `setup.bat`
   - Linux AIX `./setup.sh`

   要点: 使用打算用来运行 Agent Builder 的相同用户标识来运行安装程序。

5. **IBM Agent Builder** 窗口打开后，请选择语言，然后单击**确定**。
6. 在“**简介**”页面上，单击**下一步**。
7. 在“**软件许可证协议**”页面上，单击**我接受许可协议中的条款**，然后单击**下一步**。
8. 在“**选择安装文件夹**”页面上，单击下列其中一个选项：
   - **下一步**，用于将 Agent Builder 安装到**您希望安装到什么位置？**字段中指定的目录。
   - **复原缺省文件夹**：用于将 Agent Builder 安装到缺省目录。
   - **选择**：用于选择另一个目录。

   注: 所选目录名称不得包含以下字符：

   ```
   !
   #
   %
   ;
   ```

   如果包含其中任意字符，可能无法启动 Agent Builder。
9. 在“**安装前摘要**”页面上，单击**安装**。
10. 在“**安装 IBM Agent Builder**”页面上，等待**安装完成**页面打开，然后单击**完成**。

结果

Windows 安装 Agent Builder 之后，“开始”菜单中会增加一个选项，并且桌面上会增加 Agent Builder 图标。安装日志文件位于 *install_dir*\IBM_Agent_Builder_InstallLog.xml 中。

Linux AIX 安装 Agent Builder 之后，Agent Builder 可执行文件命名为 *Install_Location*/agentbuilder。安装日志文件位于 *install_dir*/IBM_Agent_Builder_InstallLog.xml 中。

静默安装

您可以使用静默安装方法安装 Agent Builder。此方法不需要图形环境，可以在多台主机上轻松复制。

关于此任务

静默安装选项文件 installer.properties 包含在安装映像上的安装目录根目录中。您必须修改此文件以满足您的需要，然后运行静默安装程序。您可以将此文件复制到其他主机，并将 Agent Builder 快速安装到所有这些主机上。

过程

1. 如果您未登录到 IBM Marketplace，请使用您的 IBMid 和密码登录并转至“**产品和服务**”。
   “**产品和服务**”页面适用于活动订户。如果您有任何问题，请转至 Cloud Application Performance Management 论坛或 Marketplace 支持。
2. 下载 Agent Builder 安装归档文件：
   a) 在 Cloud APM 预订框中，单击**管理** > **下载**。
   b) 选择**多平台**作为操作系统。
   c) 选择 IBM Agent Builder 程序包。
   d) 单击**下载**并将 IBM_Agent_Builder_Install.tar 保存到系统。
3. 解压缩安装归档文件。
4. 创建 installer.properties 文件的副本，该文件位于安装映像目录中。
5. 编辑新文件以满足您的需要。以下是该文件的内容示例：

```
# ------------------------------------------------------------------
# IBM Agent Builder
#
# (C) Copyright IBM Corporation 2009.  All rights reserved.
#
# Sample response file for silent install
```

```
#
# To use this file, use the following command:
#
# Windows:
#    setup.bat -i silent -f <path>\installer.properties
#
# Linux or AIX:
#    setup.sh -i silent -f <path>/installer.properties
#
# Where
#    <path> is a fully-quailfied path to the installer.properties
#    file (including the drive letter or UNC path name on Windows).
#    <path> cannot contain spaces.
# -------------------------------------------------------------------


# -------------------------------------------------------------------
# This property indicates that the license has been accepted
# -------------------------------------------------------------------
# LICENSE_ACCEPTED=FALSE

# -------------------------------------------------------------------
# This property specifies the install directory
#
# On Windows, the default is:
#     C:\\Program Files (x86)\\IBM\\AgentBuilder
#
# On Linux, the default is:
#     /opt/ibm/AgentBuilder
# -------------------------------------------------------------------
#USER_INSTALL_DIR=C:\\Program Files (x86)\\IBM\\AgentBuilder
#USER_INSTALL_DIR=/opt/ibm/AgentBuilder
```

6. 通过在解压缩的安装映像目录中运行以下命令来启动静默安装：

   Windows `setup.bat -i silent -f path/installer.properties`

   Linux AIX `./setup.sh -i silent -f path/installer.properties`

   其中，*path* 是 `installer.properties` 文件的标准路径（在 Windows 上包括盘符或 UNC 路径名）。路径不得包含空格。

启动 Agent Builder

安装 Agent Builder 之后，可以将其启动。

过程

- 使用下列其中一种方法来启动 Agent Builder
  - Windows 在 Windows 系统上：
    - 从命令行输入：*Install_Location*\agentbuilder.exe。
    - 选择**开始** > **所有程序** > **IBM** > **Agent Builder**。
    - 单击 **Agent Builder 桌面图标**。
  - Linux 在 Linux 系统上，启动以下可执行文件：*INSTALL_DIR*/agentbuilder

注: 运行 Agent Builder 时，它会提示您输入工作空间目录的位置。用于创建代理程序的文件保存在该目录中。您可将任何目录指定为工作空间。

在 Agent Builder 中设置缺省浏览器

Linux 在 Linux 系统上，您可能需要设置 Agent Builder 缺省浏览器，以便显示帮助面板。

过程

1. 选择**窗口** > **首选项**以打开“**首选项**”窗口。
2. 选择并展开**常规**节点。
3. 选择 **Web 浏览器**。

4. 选择**使用外部 Web 浏览器**。
5. 选择要使用的浏览器。
6. 可选： 要添加 Web 浏览器，请完成下列步骤
   a) 单击**新建**。
   b) 在**名称**字段中，输入浏览器的描述性名称。
   c) 在**位置**字段中，输入浏览器可执行文件的完整路径。
   d) 单击**确定**。
7. 单击**确定**。

在 Agent Builder 中设置缺省时间戳记服务机构

您可以在 Agent Builder **首选项**窗口中为 JAR 文件设置时间戳记服务机构。如果缺省时间戳记服务机构签名证书到期，通过设置新服务机构，您可以继续验证 JAR 文件。

过程

1. 选择**窗口 > 首选项**以打开“**首选项**”窗口。
2. 选择并展开 **IBM Agent Builder** 节点。
3. 选择 **JAR 签名**。
4. 选择**将时间戳记添加到已签名的 JAR 文件**。
5. 输入时间戳记服务机构的 URL。
6. 单击**确定**。

卸载 Agent Builder

根据您的操作系统，可以使用不同的过程来卸载 Agent Builder。

过程

- Linux
  在 Linux 系统上，运行以下命令：
  a) *INSTALL_DIR*/uninstall/uninstaller
  其中 *INSTALL_DIR* 是安装 Agent Builder 的目录的名称。
- Windows
  在 Windows 7、Windows Server 2008 R2 和更高版本的 Windows 上，请完成以下步骤：
  a) 选择**开始 > 控制面板 > 程序 > 程序和功能**，打开“Windows 程序和功能”。
  b) 从已安装程序的列表中选择 **IBM Agent Builder**。
  c) 单击**卸载/更改**。
  d) 在“**卸载 IBM Agent Builder**”页面上，单击**卸载**。
  e) 在“**卸载完成**”页面上，单击**完成**。
  提示: 在 Windows 7 和 Windows Server 2008 R2 上，您还可以通过选择**开始 > 计算机 > 卸载或更改程序**来转至“**Windows 程序和功能**”窗口。然后，从步骤 第 987 页的『2』 继续操作。
- Windows
  在其他 Windows 系统上，完成以下步骤：
  a) 从 Windows 控制面板，选择**添加/删除程序**。
  b) 单击 **IBM Agent Builder**。
  c) 单击**更改/删除**。
- 在所有操作系统上，您还可以使用静默卸载方法。通过运行以下命令启动静默卸载：
  - Windows 在 Windows 系统上：*INSTALL_DIR*/uninstall/uninstaller.exe -i silent

- Linux 在 Linux 系统上：*INSTALL_DIR*/uninstall/uninstaller -i silent

静默卸载

您可以使用静默卸载方法执行卸载。

过程

- 通过运行以下命令启动静默卸载：

```
INSTALL_DIR/uninstall/uninstaller[.exe] -i silent
```

创建代理程序

要开始在 Agent Builder 中创建代理程序，请使用“新建代理程序”向导。通过此向导，您可以设置基本代理程序配置并创建一个数据源。然后，可以在 Agent Builder 中处理该代理程序以添加更多数据源和其他选项，包括子节点和导航器组。

对代理程序命名和配置

使用“**代理程序**”向导对代理程序命名，设置代理程序版本、受支持的操作系统和其他配置设置。

过程

1. 使用下列其中一种方法来启动“新建代理程序”向导：
   a) 单击工具栏上的 **创建新代理程序**图标。
   b) 从主菜单中，选择**文件** > **新建** > **代理程序**。
   c) 从主菜单中，选择**文件** > **新建** > **其他**。在“**选择向导**”页面中，双击 **Agent Builder** 文件夹，然后双击**代理程序**。
   “**代理程序**”向导将打开。
2. 单击**下一步**。
3. 在“**新建代理程序项目**”页面中的**项目名称**字段中，设置项目名称。Agent Builder 将此名称用于包含代理程序文件的文件夹。您可以选择更改以下设置：
   - 如果要将代理程序文件存储在其他位置，请取消选中**使用缺省位置**，并单击**浏览**以在**位置**字段中选择新目录。
   - 通过将资源添加到各种工作集，可更改 Eclipse Navigator View 显示资源的方式。有关更多信息，请参阅 Eclipse 帮助。要向 Eclipse 工作集添加代理程序，请选择**将项目添加至工作集**并单击**选择**按钮，将工作集添加到**工作集**字段。
4. 单击**下一步**。
5. 在“**常规信息**”页面中，配置以下设置：
   - 在**版权**字段中输入要用于新代理程序的版权声明。该声明必须符合对版权的法律要求。此版权声明会插入到为代理程序生成的所有文件中；您之后可以对其进行编辑。
   - 选择要为其构建代理程序的操作系统。

     要点: 如果想要在 Agent Builder 内部对代理程序运行全面测试（有关指示信息，请参阅第 1159 页的『代理程序全面测试』），请确保：

     – 如果在 Windows 上运行 Agent Builder，安装 32 位版本的操作系统。
     – 如果在 Linux 上运行 Agent Builder，安装 64 位版本的操作系统。

     要点: 在极少数情况下，您可能需要在仅安装有 32 位操作系统代理程序的 64 位系统上安装代理程序。在此情况下，请确保不要选择 64 位版本的操作系统，而选择 32 位版本。

要点: 使用 Agent Builder 创建的代理程序不支持 64 位 Windows Server 2003 R2 及更早版本的 Windows 系统。

6. 单击**下一步**。
7. 在“**代理程序信息**”页面中，配置以下设置：
   - 在**服务名称**字段中，为代理程序设置服务名称。此名称会显示在 IBM Tivoli Monitoring 环境中的“**管理 Tivoli Monitoring Services**”窗口以及 IBM Cloud Application Performance Management 中的“**管理 Monitoring Services**”实用程序和阈值编辑器中。在 Windows 系统上，这也是运行代理程序的 Windows 服务的名称。完整的服务名称始终以 `Monitoring Agent for` 开头。您负责输入名称的余下部分，这个部分通常用来描述此代理程序所监视的服务。名称可以包含字母、数字、空格和下划线。
   - 在**产品代码**字段中为代理程序设置三字符产品代码。IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 都需要产品代码。已保留一系列产品代码以用于 Agent Builder。允许的值为 `K00-K99`、`K{0-2}{A-Z}` 和 `K{4-9}{A-Z}`。

     要点: 这些值仅供内部使用，而不用于要在组织外部共享或销售的代理程序。如果要创建与他人共享的代理程序，必须向 toolkit@us.ibm.com 发送说明，以保留产品代码。请求产品代码必须包含要构建的代理程序的描述。然后，将为您分配、注册和返回产品代码。当您收到三个字母的产品代码时，系统将告知您如何使 Agent Builder 能够使用分配的产品代码。
   - 在**公司标识**字段中设置一个字符串，用于唯一标识开发代理程序的组织（`IBM` 是保留字）。您可以从公司的 URL 获取该标识；例如，如果公司 Web 站点是 `mycompany.com`，可使用文本 `mycompany`。
   - 在**代理程序标识**字段中设置一个字符串以唯一标识代理程序。在缺省情况下，Agent Builder 将代理程序标识设置为与产品代码相同。

     要点: 代理程序标识字段和**公司标识**字段的长度之和不能超过 11 个字符。
   - 在**版本**字段中设置代理程序版本。代理程序版本以 *V.R.R* 格式包含一个三位数，其中：

     V = 版本
     R = 发行版
     R = 发行版

     为了在监视环境中显示，*V.R.R* 值会转换为以下格式：`0V.RR.00.00`

     提示: 在 Agent Editor 中，提供了**补丁级别**字段。在为代理程序发布修订时，可以使用**补丁级别**字段，而无需更新版本。
   - 如果希望代理程序支持多个实例，请选中**支持此代理程序的多个实例**复选框。您可以使用代理程序的多个实例来监视同一主机上某个应用程序的若干实例，或者使用一个主机上安装的代理程序来监视不同主机上的若干软件服务器。在安装代理程序以支持多个实例时，可以创建所需数量的实例并进行配置。

下一步做什么

单击**下一步**，为代理程序定义初始数据源。有关更多信息，请参阅第 989 页的『定义初始数据源』

定义初始数据源

在创建代理程序时，请定义该代理程序要监视的初始数据。稍后，您可以在 Agent Editor 中添加更多数据源。

关于此任务

使用“**代理程序初始数据源**”页面来定义新代理程序要监视的数据源。有关根据各种数据提供程序创建数据源的详细指示信息，请参阅第 1025 页的『定义和测试数据源』。

过程

1. 在“**代理程序初始数据源**”页面上，选择其中一个**监视数据类别**及其中一个**数据源**。
2. 单击**下一步**。向导将引导您完成定义和配置任何所指定数据收集类型的过程。

提示: 您可以使用此向导来定义数据源，或者添加子节点或导航器组以组织代理程序。有关子节点的更多信息，请参阅第 1127 页的『使用子节点』。有关仅用于 IBM Tivoli Monitoring 的导航器组的更多信息，请参阅第 1126 页的『创建导航器组』。

3. 如果您已定义可以返回多个数据行的新数据源，那么将提示您选择键属性。有关更多信息，请参阅第 990 页的『选择键属性』。
4. 在定义首个数据源之后，将显示“**数据源定义**”窗口。要添加另一个数据源，请选择代理程序，或者选择子节点或导航器组（如果存在），然后单击**添加到选定项**按钮。
5. 要完成定义数据源，请单击**完成**。Agent Builder 会创建新代理程序并在 Agent Editor 中将其打开。

选择键属性

属性组返回多个数据行时，您必须选择键属性。

关于此任务

属性组可以返回多个数据行时，每一行代表一个正受监视的实体。每次对受监视数据采样时，监视环境都将行与受监视实体以及该实体的先前采样匹配。此匹配可通过键属性来完成。属性组中的一个或多个属性可以标识为键属性。这些键属性组合起来可以区分受监视实体。当键属性表示同一受监视实体时，不会从一个样本切换到下一个样本。

速率和变化量属性通过比较当前样本与先前样本计算得出。键属性相同可确保代理程序正在比较同一受监视实体的值。同样，Summarization and Pruning Agent 汇总具有相同键属性的样本。另外，在某种情境下，任何设置为键属性的属性还可以用作“显示项”。

您可以在“**代理程序初始数据源**”页面中指定新数据源的相关详细信息。如果所选数据源可能会返回多个数据行，那么 Agent Builder 有时可能会检测到键属性。否则，会提示您选择键属性。

过程

- 在“**选择键属性**”页面上，请执行下列其中一个步骤：
  - 从列表中单击作为该实体的键属性的一个或多个属性。要选择多个属性，请按住 Ctrl 键。
  - 如果此属性组仅返回一行，请选择**生成单一数据行**。如果选择了该选项，那么无需任何键属性，原因是该属性组中只会报告一个受监视实体。

使用 Agent Editor 修改代理程序

使用 Agent Editor 更改、保存和落实代理程序的版本。

您可以在 Agent Builder 中创建新代理程序；有关更多信息，请参阅第 988 页的『创建代理程序』。创建代理程序之后，可以使用 Agent Editor 修改此代理程序。

要在 Agent Editor 中打开您在 Agent Builder 中创建的代理程序，可在 **Project Explorer** 窗格中，找到该代理程序的名称并将其展开。在该代理程序名称下面，双击**代理程序定义**。或者，双击 `itm_toolkit_agent.xml` 文件名。

Agent Editor 是包含多个页面的 Eclipse 编辑器，您可以用于修改现有代理程序的属性。编辑器中的每个页面都对应于代理程序的一个特定功能。

可用页面列表显示在“大纲”视图中的**代理程序定义**节点下方。您可以通过在“大纲”视图中单击节点轻松切换到其他页面。如果没有“大纲”视图，或者此视图隐藏在另一视图下方，可以重置“代理程序定义”透视图。选择**窗口 > 重置透视图**可重置此透视图。另外，也可以右键单击**代理程序定义**选项卡，并从菜单中选择**重置**。

注: 有关创建代理程序的详细信息和过程，请参阅第 988 页的『创建代理程序』。

Agent Editor 中包含以下页面：

- 第 991 页的『“代理程序信息”页面』
- “数据源定义”页面
- “运行时配置信息”页面

· “代理程序 XML 编辑器”页面 (itm_toolkit_agent.xml)

注: 在查看 Editor 页面时，还可以通过单击其他页面的选项卡来切换到该页面。某些页面只有在“大纲”视图中选中后才显示选项卡。您可以强制使页面在未选中时显示选项卡。要强制使页面显示选项卡，请单击针形图标以使该图标中的针形指向该页面。

“代理程序信息”页面

“**代理程序信息**”页面是 Agent Editor 的主页。

“**代理程序信息**”页面包含以下信息:

· 一般代理程序信息，其中包括代理程序服务名称和产品代码。您可以单击**高级**，以设置不同的名称用于不同用途，但通常不需要进行此设置。

· 代理程序内容信息

  – **缺省操作系统**链接
  – **自描述代理程序**链接
  – **环境变量**链接
  – **看守程序信息**链接
  – **Cognos 信息**链接
  – **数据源**链接
  – **运行时配置**链接
  – **资源**链接
  – **仪表板**链接

· **测试代理程序**链接

· **生成代理程序向导**链接

· **落实代理程序版本**链接

配置瞬时错误消息的时间

Agent Editor 向导有时显示瞬时错误消息。消息会在向导标题中显示片刻（缺省情况下显示 3 秒）。您可以配置显示这些消息的持续时间。要更改此设置:

1. 从 Agent Builder 菜单栏中选择**窗口 > 首选项**。“**首选项**”窗口随即打开。
2. 选择 **Agent Builder**。
3. 设置**瞬时错误消息显示的时间（秒）**设置。
4. 单击**确定**。

缺省操作系统

使用**缺省操作系统**页面可更改构建代理程序时针对的操作系统。

过程

· 要打开**缺省操作系统**页面，请单击“**代理程序信息**”页面的“**代理程序内容**”部分中的**缺省操作系统**，或者单击“大纲”视图中的**缺省操作系统**节点。

· 在**缺省操作系统**页面中，选择您的代理程序一定可以支持的操作系统。

  在为代理程序生成安装程序包时，Agent Builder 会将所选操作系统的文件添加到该程序包。添加到代理程序而且并非特定于 Windows 操作系统的数据源在选择的所有操作系统上都可用。您可以更改任何特定数据源所适用于的操作系统，使其不同于此缺省选择。要更改可用于特定数据源的操作系统，请使用“**数据源定义**”页面的“**操作系统**”窗格。如果未选择缺省操作系统，那么您必须在“**数据源定义**”页面上针对每一个特定数据源选择操作系统。

  要点: 如果想要在 Agent Builder 内部对代理程序运行全面测试（有关指示信息，请参阅第 1159 页的『代理程序全面测试』），请确保:

– 如果在 Windows 上运行 Agent Builder，安装 32 位版本的操作系统。
– 如果在 Linux 上运行 Agent Builder，安装 64 位版本的操作系统。

要点: 在极少数情况下，您可能需要在仅安装有 32 位操作系统代理程序的 64 位系统上安装代理程序。在此情况下，请确保不要选择 64 位版本的操作系统，而选择 32 位版本。

自描述代理程序

对于 IBM Tivoli Monitoring 环境，请使用“**自描述代理程序**”页面来指定是否将代理程序的支持文件与代理程序捆绑在一起。对于 IBM Cloud Application Performance Management 环境，必须保持启用自描述代理程序。

过程

· 要打开“**自描述代理程序**”页面，请单击“**代理程序信息**”页面的“**代理程序内容**”部分中的**自描述代理程序**，或单击“大纲”视图中的**自描述代理程序**节点。

缺省情况下，针对使用 Agent Builder 6.2.3 或更高版本创建的所有新代理程序启用自描述。如果代理程序用于 IBM Cloud Application Performance Management 环境，必须启用自描述。

对代理程序启用自描述功能后，应用程序支持包将包括在代理程序映像中。包括在内将使代理程序能够植入 Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal Browser 的支持文件。有关自描述代理程序的更多信息，请参阅《IBM Tivoli Monitoring 安装与设置指南》和《IBM Tivoli Monitoring 管理员指南》。在 IBM Cloud Application Performance Management 环境中，自描述使代理程序能够将支持文件植入到 Cloud APM 服务器上；植入过程是此环境中的必需步骤。

注: 在 IBM Tivoli Monitoring 环境中，必须安装 Tivoli Monitoring V6.2.3 或更高版本才能使自描述代理程序功能工作，并且必须在 Tivoli Monitoring 中启用自描述。缺省情况下，自描述功能在 Tivoli Monitoring 中处于关闭状态。

注: 选中**针对此代理程序启用自描述**复选框不会阻止代理程序在先前版本的 Tivoli Monitoring 上工作。

环境变量

使用“**环境变量**”页面查看和修改代理程序在运行期间可以使用的环境变量。

开始之前

有关“**Agent Editor**”和“**代理程序信息**”页面的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

关于此任务

环境变量可以是您在脚本中为访问权定义的变量，也可以是使代理程序按照某种方法运行的预定义变量。请参阅第 993 页的『环境变量列表』以获取预定义变量列表。

过程

1. 要打开“**环境变量**”页面，请在“**代理程序信息**”页面的**代理程序内容**部分中单击**环境变量**。或者，单击“**大纲**”视图中的**环境变量**节点。
2. 在“**环境变量**”页面中，单击**添加**以添加新变量。或者，要编辑现有变量，请将其选中并单击**编辑**。
3. 在“**环境变量信息**”窗口中，设置以下值:
   · 在**名称**字段中，输入变量名称，或者从列表中选择预定义名称。
   · 在**值**字段中，如果要为代理程序设置变量，输入变量值。如果不输入值，代理程序会传播现有变量的值。
   · 在**描述**字段中，输入变量描述，或者保留预定义变量的现有描述。

a) 单击**确定**。
新增变量列示在“**代理程序信息**”页面的表中。

环境变量列表

使用环境变量来控制代理程序运行时行为。

通过使用“**环境变量**”页面，可以将环境变量构建到代理程序中。在 Windows 系统上，环境变量在代理程序的 K*XX*ENV 文件中定义。在 UNIX 和 Linux 系统上，可以在代理程序的 $CANDLEHOME/config/*XX*.ini 文件中定义这些变量。*XX* 是两字母产品代码。代理程序必须重新启动才能使新设置生效。

注: 未在运行 C Shell 的远程系统上正确设置环境变量。如果要使用环境变量，请使用其他 shell。

表 *259.* 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述

环境变量	缺省值	有效值	描述
CDP_*ATTRIBUTE_GROUP*_REFRESH_INTERVAL	不适用	任何非负整数	在后台更新指定属性组的时间间隔（秒）。此变量工作方式与 CDP_DP_REFRESH_INTERVAL 相同，除了其目标仅为指定属性组外。变量名中的属性组名称必须为大写，即便实际属性组名称不为大写也是如此。
CDP_DP_CACHE_TTL	55	大于等于 1 的任何整数。	在此秒数内，对所收集的属性组数据进行高速缓存。在该时间间隔内，对相同数据的多次请求将收到该数据的高速缓存副本。该值应用到代理程序中的所有属性组。
CDP_*ATTRIBUTE_GROUP*_CACHE_TTL	CDP_DP_CACHE_TTL 的值	大于等于 1 的任何整数。	在此秒数内，对所收集的指定属性组数据进行高速缓存。在该时间间隔内，对相同数据的多次请求将收到该数据的高速缓存副本。此值覆盖指定组的 CDP_DP_CACHE_TTL。变量名中的属性组名称必须为大写，即便实际属性组名称不为大写也是如此。
CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT	如果定义了子节点则为 5，否则不设置	任何正整数	在超时和返回高速缓存数据前（即使高速缓存数据是旧数据）等待数据收集的秒数。（如果高速缓存数据存在时间超过 CDP_DP_CACHE_TTL 秒，那么为旧数据）。如果未设置此变量，代理程序将等待，直到数据收集完成。等待有时会使 Tivoli Enterprise Portal 超时并放弃等待。如果未配置任何线程池，将忽略该变量并将同步进行数据收集。
CDP_DP_REFRESH_INTERVAL	如果定义了子节点则为 60，否则不设置	任何非负整数	在后台更新属性组的时间间隔（秒）。如果该变量未设置或者设置为 0，将禁用后台更新。如果配置了线程池（请参阅变量 CDP_DP_THREAD_POOL_SIZE），那么可以用并行方式刷新属性组。如果没有线程池，那么将按顺序方式进行更新，这可能会花费很长时间。在逻辑上等同于线程池大小为 1。

表 *259.* 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 *(续)*

环境变量	缺省值	有效值	描述
CDP_DP_THREAD_POOL_SIZE	如果定义了子节点则为 15，否则不设置	任何非负整数	为按照 CDP_DP_REFRESH_INTERVAL 定义的时间间隔运行后台数据收集而创建的线程数。如果该变量未设置或者设置为 0，表示没有线程池。 如果 CDP_DP_THREAD_POOL_SIZE 设置为大于 1 的值，且 CDP_DP_REFRESH_INTERVAL 设置为 0，那么将忽略 CDP_DP_THREAD_POOL_SIZE 的值并按需进行数据收集。 Thread Pool Status 属性组显示线程池是如何运行的。使用 Thread Pool Status 来调整线程池大小和刷新时间间隔以获得最佳结果。缺省情况下，该属性组的查询不会在代理程序导航器树上显示。您可能会忘记在代理程序的定制工作空间中包含查询。但是，可以通过将 Thread Pool Status 查询指定到基本代理程序级别工作空间视图进行轻松查看。
CDP_JDBC_MAX_ROWS	1000	任何正整数	JDBC 数据提供程序返回的最大数据行数。对于包含的行数超过该数量的结果集，最多仅处理此最大数量的数据行。可开发查询以避免将太多数据返回到 IBM Tivoli Monitoring。
CDP_NT_EVENT_LOG_GET_ALL_ENTRIES_FIRST_TIME	NO	YES 或 NO	如果设置为 YES，那么代理程序将针对 Windows 事件日志中的每个事件发送一个事件。如果设置为 NO，那么仅发送 Windows 事件日志中的新事件。
CDP_NT_EVENT_LOG_CACHE_TIMEOUT	3600	大于等于 300 的任何整数。	代理程序对 Windows 事件日志事件进行高速缓存的秒数。当查询事件日志属性组时，将返回所有高速缓存的事件。 **注:** 不再使用该变量。而使用 CDP_PURE_EVENT_CACHE_SIZE 变量。
CDP_PURE_EVENT_CACHE_SIZE	100	大于等于 1 的任何正整数。	针对以下项高速缓存的最大事件数：配置为处理新记录的日志文件数据源、Windows 事件日志属性组。以及 JMX 监视器和通知。 日志中的每个新记录都将导致发送事件。该环境变量定义代理程序在高速缓存中记住的事件数。当查询该属性组时，将返回高速缓存值。
CDP_DP_ACTION_TIMEOUT	20 秒	大于等于 1 的任何正整数。	等待代理程序正在处理的“执行操作”完成的秒数。
CDP_DP_SCRIPT_TIMEOUT	30 秒	大于等于 10 的任何正整数。	等待基于脚本的属性组所启动程序完成的秒数。

表 *259.* 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 *(续)*

环境变量	缺省值	有效值	描述
CDP_DP_PING_TIMEOUT	30 秒	大于等于 10 的任何正整数。	等待命令返回码启动的程序完成的秒数。 **注:** 此变量与 ICMP Ping 数据提供程序无关。
CDP_SNMP_MAX_RETRIES	2	任何正整数	发送 SNMP 请求重试次数。如果未收到响应，那么发送到 SNMP 代理程序的请求总数是该值加一。
CDP_SNMP_RESPONSE_TIMEOUT	2 秒	任何正整数	等待每个 SNMP 请求超时的秒数。 属性组中的每行表示一个单独的请求。此超时值是在进行重试前等待响应的秒数。单行数据的总超时为 (CDP_SNMP_MAX_RETRIES + 1) * CDP_SNMP_RESPONSE_TIMEOUT。总的缺省超时值是 (2+1) * 2 = 6 秒。
CDP_DP_HOSTNAME	安装的第一个网络接口的名称	IP 地址或主机名	在多接口系统上设置首选主机名（网络接口）。 如果代理程序将其侦听端口绑定到非缺省网络接口地址，那么请使用此环境变量。此环境变量由 SNMP 数据提供程序使用。 对于套接字数据源，如果同时设置了 CDP_DP_ALLOW_REMOTE，那么此变量也适用。
CDP_SNMP_ALLOW_ DECREASING_OIDS	NO	YES 或 NO	如果设置为 YES，SNMP 数据提供程序不会检查返回的 OID 是否在增大。设置为 YES 时需小心，因为受监视代理程序可能有问题，通常此检查会捕获这些问题。
KUMP_DP_COPY_MODE_SAMPLE_I NTERVAL	60	等待时间（秒）	对于日志文件数据提供程序，在将代理程序定义为**对文件进行采样时处理所有记录**时，指定其重新读取文件内容前的等待时间。以秒为单位指定此时间。
KUMP_MAXPROCESS	100%	5-100%	对于日志文件数据提供程序，指定要用来处理文件数据的最大处理器使用率。值的范围为 5-100%。缺省值为 100%。
KUMP_DP_SAMPLE_FACTOR	5	任何非负整数	对于日志文件数据提供程序，在 Agent Builder 上选择**对文件进行采样时处理所有记录**时，设置采样因子。此等待时间确保在针对模式记录扫描前写入跨多个记录的模式。
KUMP_DP_EVENT	5	任何非负整数	对于日志文件数据提供程序，设置事件数据的采样频率（秒）。
KUMP_DP_FILE_EXIST_WAIT	YES	YES 或 NO	对于日志文件数据提供程序，指定文件监视线程检测到受监视的文件不存在或者为空，该线程继续运行。该线程将等待文件出现，每隔几秒钟重新检查一次，并在文件可用时启动或重新启动监视。

表 259. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)

环境变量	缺省值	有效值	描述
KUMP_DP_FILE_SWITCH_CHECK_INTERVAL	600	任何非负整数	以秒为单位的频率，当启用动态文件名支持时，日志文件数据提供程序按此频率搜索要切换到的其他监视文件。
KUMP_DP_FILE_ROW_PAUSE_INCREMENT	无	任何非负整数	对于日志文件数据提供程序，指定在文件监视线程暂停前读取的文件记录数。暂停文件监视线程是以便处理之前的更新。仅当受监视文件突然收到大量新记录，并且您担心可能会丢失某些记录更新时，才使用此环境变量。
CDP_COLLECTION_TIMEOUT	60 秒	任何正整数	代理程序等待另一个进程启动的数据收集器做出响应的秒数。示例包括 JMX、JDBC、HTTP 和 SOAP 数据收集器。
CDP_SSH_TEMP_DIRECTORY	. （句点）	远程系统上的任何有效路径字符串	对于启用 SSH 的脚本数据提供程序，指定远程系统上的位置。随代理程序提供的脚本文件将更新至此位置。相对位置相对于用户的主目录。缺省值 . （句点）表示用户的主目录。
CDP_SSH_DEL_COMMAND	rm -Rf	远程系统上的任何有效删除命令字符串	对于启用 SSH 的脚本数据提供程序，指定命令以开始删除代理程序随附的已上载脚本文件。
CDP_SNMP_SEND_DELAY_FACTOR	0 毫秒	任何正整数	SNMP 初始发送从 0 延迟为指定的毫秒数。仅当同时启用了线程池时才会启用此变量。该延迟不会应用到所有发送，而只会应用到属性组进行的第一次发送。当受监视设备由于同时收到多个请求而难以做出正确响应时，此变量非常有用。
CDP_ICMP_PING_REFRESH_INTERVAL	60 秒	大于等于 1 的任何整数	在此时间间隔内已经对设备列表文件中的系统执行了 Ping 操作。如果 Ping 使用太长时间，那么再次开始 Ping 之前，总是至少延迟 CDP_PING_MIN_INTERVAL_DELAY 秒。数据的刷新频率不会超过此设置。可以根据设备列表文件中的条目数量和接收响应所用时间来降低刷新数据的频率。
CDP_ICMP_PING_MIN_INTERVAL_DELAY	30 秒	大于等于 1 且小于 CDP Ping 刷新时间间隔的任何整数	在对设备列表文件中的设备执行 Ping 之后，下一个 Ping 刷新时间间隔至少要在此秒数之后才会开始。
CDP_ICMP_PING_BURST	10	大于等于 0 的任何整数	在代理程序暂停时间达到 CDP_ICMP_PING_BURST_DELAY 变量指定的值之前发送的 Ping 数。值为 0 将禁用此功能。
CDP_ICMP_PING_BURST_DELAY	10	大于等于 0 的任何整数	在发送 CDP_ICMP_PING_BURST 变量所定义数量的一组 Ping 之后等待的时间量（毫秒）。值为 0 将禁用此功能。

表 *259.* 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 *(续)*

环境变量	缺省值	有效值	描述
CDP_ICMP_PING_TIMEOUT	2000 毫秒	大于等于 1 的任何整数	等待 Ping 响应的毫秒数。此设置适用于所做的每次 Ping 尝试。系统会对每个主机尝试 3 次 Ping 操作。如果 3 次尝试都未收到响应，那么等待应答的总时间为 CDP_ICMP_PING_TIMEOUT 乘以 3。在缺省情况下，此值为 6000 毫秒。更改 CDP_ICMP_PING_TIMEOUT 的值将导致当前响应时间属性的缺省 TIMEOUT 枚举不再适用。将 TIMEOUT 枚举更改为新值，即 CDP_ICMP_PING_TIMEOUT 乘以 3。
CDP_JDBC_CONNECTIONLESS	false	true 或 false	如果设置为 true，那么将在每次数据收集尝试后关闭 JDBC 连接。即，所有属性组都会在每次进行数据收集时尝试创建自己的连接。如果启用了此变量，那么不会复用连接。如果设置为 false，那么建立与数据库的一个连接后，属性组之间将共享该连接。
CDP_SSH_EXCLUDED_ ENVIRONMENT_VARIABLES	无	以逗号分隔的环境变量名称列表	对于启用了 SSH 的脚本数据提供程序，指定不允许在远程系统环境中设置的一组本地环境变量。
CDP_DP_EVENT_LOG_MAX_ BACKLOG_TIME	0 秒	0、1 或大于 1 的任何整数	如果设置为 0，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS 未设置为 1 或更大的整数，那么不处理在代理程序关闭期间生成的事件。0 是缺省值。 如果设置为 1，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS 未设置为大于 1 的整数，那么将处理在代理程序关闭期间生成的所有事件。 如果设置为大于 1，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS 未设置为大于 1，那么将处理以当前计算机时间为准，该秒数值内生成的事件。例如，如果值设置为 300，那么在启动时，代理程序处理当前时间 300 秒内生成的所有事件。 在针对 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 和 CDP_DP_EVENT_LOG_MAX _BACKLOG_EVENTS 变量输入大于 1 的值时，将处理此时间间隔内的事件或此数量的事件。选择哪个变量取决于哪一项最先匹配。
CDP_DP_EVENT_LOG_ *Windows_Event_Log*_MAX_BACKLOG_ TIME	0 秒（不处理代理程序关闭期间错过的事件）	0、1 或大于 1 的任何整数	如果设置为

表 *259.* 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 *(续)*

环境变量	缺省值	有效值	描述
CDP_DP_EVENT_LOG_ MAX_BACKLOG_EVENTS	0 个事件	0、1 或大于 1 的任何整数	如果设置为 0，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 变量未设置为 1 或更大的整数，不处理在代理程序关闭期间生成的事件。0 是缺省值。 如果设置为 1，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 变量未设置为大于 1 的整数，那么将处理在代理程序关闭期间生成的所有事件。 如果设置为大于 1，并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 未设置为大于 1，那么最多将处理在代理程序关闭期间生成的此数量的事件。例如，如果值设置为 200，那么在代理程序启动时，将处理启动前生成的最近 200 个事件。 在针对 CDP_DP_EVENT_LOG_MAX_BACKLOG _EVENTS 和 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 输入大于 1 的值时，将处理此数量的事件或此时间间隔内的事件。选择哪个变量取决于哪一项最先匹配。
CDP_DP_EVENT_LOG_ *Windows_Event_Log*_MAX_BACK LOG_ EVENTS	0 个事件（不处理代理程序关闭期间错过的事件）	0 或者大于等于 1 的任何整数	如果设置为
CDP_HTTP_READ_TIMEOUT	10	任何正整数	等待对 HTTP 请求的应答的秒数。
CDP_JAT_THREAD_POOL_SIZE	15	任何正整数	Java 提供者为处理数据收集请求而使用的线程数。JMX、JDBC、HTTP 和 SOAP 数据提供程序是可受益于此线程池的提供者。
CDP_HTML_OBJECTS_THREAD_ POOL_SIZE	10	任何正整数	这是一个线程数，这些线程用于下载在使用 HTTP 数据提供程序监视的 URL 中找到的页面对象。
CDP_HTTP_SOAP_MAX_ROWS	500	任何正整数	HTTP SOAP 数据提供程序返回的最大行数。
CDP_DP_ALLOW_REMOTE	NO	NO 和 YES	如果设置为 YES，代理程序将允许远程套接字连接。如果设置为 NO，代理程序将只允许来自本地主机的套接字连接。NO 是缺省值。
CDP_DP_INITIAL_COLLECTION_ DELAY	不固定	任何正整数	代理程序启动之后，线程池开始其调度的数据收集之前的秒数。

看守程序信息

使用“**看守程序信息**”页面以指定代理程序看守程序的配置信息。

关于此任务

要打开“**看守程序信息**”页面，请在“**代理程序信息**”页面的**代理程序内容**部分中单击**看守程序信息**。您还可以选择“大纲”视图中的**看守程序信息**节点。

可以为代理程序看守程序指定以下配置信息：

- **缺省情况下监视此代理程序**

  选中此复选框以在安装该代理程序后将代理程序交由 Agent Management Services 管理。将监视代理程序是否有不正常行为或异常终止，并且由看守程序重新启动。

- **检查频率（秒）**

  看守程序检查代理程序进程是否存在任何不正常行为或异常终止的频率。缺省值为每 180 秒一次。

- **最大重新启动次数**

  在向管理员发出问题警报前，在 24 小时周期内由于发生不正常行为或异常终止，看守程序重新启动代理程序的次数。该周期是从每天的午夜开始。因此，代理程序启动的第一个周期可能会“很短”。

  如果代理程序因任何原因关闭，将发生重新启动。如果因代理程序无反应或运行不正常，那么看守程序也会停止。例如，超出内存阈值时。缺省设置是在 24 小时周期内重新启动四次，该周期为从午夜到晚上 11:59。代理程序的每日重新启动计数在午夜自动归零。

- **内存阈值信息**

  这是代理程序进程的大小上限（以兆字节计），如果代理程序大小增大到超出此值，那么其看守程序会认定它运行不正常。Windows、Linux 和 UNIX 各有不同的值。如果代理程序进程大小超过此阈值，看守程序将停止并重新启动该进程。这些属性没有缺省值。如果不指定值，看守程序不会监视进程大小。此度量在 Windows 上使用工作集大小，在 UNIX 和 Linux 上使用用户内存大小。

如果看守程序停止代理程序，并且到达最大重新启动次数，那么看守程序发送代理程序超过其重新启动计数的警报，并停止执行自动重新启动。看守程序仍将报告代理程序是启动还是停止（假定其以另一种方式启动，例如，通过 Tivoli Enterprise Portal）。

必须使用 AMS 启动代理程序“执行操作”命令手动重新启动代理程序，以使重新启动计数不被重置。

此计数在以下某种情况下会重置（看守程序继续工作并报告状态，但不执行自动重新启动）：

- 时钟指向午夜 12 点。
- 用户使用 AMS 启动代理程序“执行操作”命令，此命令具有名为 **resetRestartCount** 的输入参数。如果输入值 1（表示“true”或“yes”），那么每日重新启动计数将重置为 0。

有关更多信息，请参阅 *IBM Tivoli Monitoring Administrator's Guide* 中的以下部分：

- 对于 Tivoli System Monitor Agent

  Configuring Agent Management Services on Tivoli System Monitor Agents

- 对于 Tivoli Enterprise Monitoring Agent

  Installing and configuring Tivoli Agent Management Services

Cognos 信息

使用“**Cognos 信息**”页面来指定为代理程序生成 Cognos 数据模型时使用的信息。此信息仅用于 IBM Tivoli Monitoring 环境。

过程

1. 要打开“**Cognos 信息**”页面，请单击“**代理程序信息**”页面的**代理程序内容**部分中的 **Cognos 信息**，或单击“大纲”视图中的 **Cognos 信息**节点。
2. 在**数据源**字段中，输入将 Tivoli Common Reporting 连接至 IBM Tivoli Data Warehouse 的数据源的名称。

缺省值为 TDW。

3. 在**模式**字段中，输入用于 Tivoli Data Warehouse 的数据库模式的名称，此名称用来对 Cognos 报告中的表名进行完全限定。

   缺省值为 ITMUSER。将生成的 Cognos 模型装入 Framework Manager 时，可以在 Framework Manager 中更改此值。

   “**数据源定义**”页面中的**向报告类别添加此属性组**复选框确定此属性组在 Cognos 模型中的放置位置。如果未选中此复选框，则会将此属性组放置在 Cognos 模型内的扩展属性文件夹中。如果选中此框，那么此属性组将放入“关键度量”文件夹中的所选子文件夹（“可用性”或“性能”）。有关数据源字段的更多信息，请参阅第 1003 页的表 260。

下一步做什么

您可以使用 Cognos 数据模型为代理程序创建 Tivoli Common Reporting 报告，请参阅 第 1245 页的『生成 Cognos 数据模型』。

“生成代理程序向导”链接

创建或编辑新的代理程序后，请使用“生成代理程序向导”准备安装。

过程

- 完成创建或编辑新代理程序之后，在 **Agent Editor** 的“**代理程序信息**”页面上，单击**生成代理程序向导**链接。

  通过“生成代理程序向导”，您可以：

  - 在本地系统上使用 Tivoli Monitoring 安装生成代理程序文件。有关指示信息，请参阅第 1164 页的『本地安装代理程序』。
  - 创建程序包以便可以在其他系统上安装该代理程序。有关指示信息，请参阅第 1165 页的『创建代理程序包』。

“数据源定义”页面

使用“**数据源定义**”页面来操作数据源。

关于此任务

“**数据源定义**”页面列出针对代理程序配置的数据源。当您在树中选择某个数据源或特性时，此页面将更新为显示选定对象的属性。使用这些字段可修改所选数据源或特性的属性。

注: 有关根据各种数据提供程序创建数据源的详细指示信息，请参阅第 1025 页的『定义和测试数据源』。

过程

- 要打开“**数据源定义**”页面，请单击“**代理程序信息**”页面的“**代理程序内容**”部分中的**数据源**，或单击“**大纲**”视图中的**数据源**节点。
- 通过单击**添加到选定项**或者在导航树中单击右键并选择其中一个选项，可以添加更多数据源。
- 通过右键单击数据源和特性并选择**移除**，可以将其移除。
- 您可以添加、修改和移除特性。有关指示信息，请参阅第 1003 页的『编辑数据源和特性属性』

通过使用“数据源定义”页面来复制数据源

使用“**数据源定义**”页面来复制数据源。

开始之前

转至“**数据源定义**”页面。有关更多信息，请参阅第 1000 页的『“数据源定义”页面』

关于此任务

您可以将产生属性组的数据源复制到剪贴板，然后粘贴回到此代理程序或者粘贴到另一代理程序。不产生属性组的数据源包括“可用性”及“Windows 事件日志”数据源。

过程

1. 选择要复制的属性组。
2. 使用下列其中一种方法剪切或复制属性组：
   - 单击菜单栏中的**编辑 > 剪切 > 编辑 > 复制**。
   - 右键单击其中一个已选择的项目，然后单击菜单中的**剪切**或**复制**。
   - 使用其中一个将会调用剪切或复制操作的操作系统或 Eclipse 击键。例如，在 Windows 系统上，按下 **Ctrl-C** 将调用复制操作。

   要将数据源从现有位置移除并放置到剪贴板中，请使用**剪切**。要将数据源保留在原位置并复制到剪贴板，请使用**复制**。
3. 选择属性组的父代（代理程序、子节点或导航器组），或选择现有属性组。
4. 通过使用下列其中一个选项，粘贴选择内容：
   - 从菜单栏中选择**编辑 > 粘贴**。
   - 在树中，右键单击要粘贴选择内容的节点，然后单击菜单中的**粘贴**。
   - 使用其中一个将会调用粘贴操作的操作系统或 Eclipse 击键。例如，在 Windows 系统上，按下 **Ctrl-V** 将调用粘贴操作。

结果

剪贴板中的属性组将放置在选择的父代中。或者，如果已选中属性组，那么剪贴板中的属性组将放置在选定属性组的父代中。

如果粘贴时与另一属性组发生名称冲突，那么会对粘贴的属性组名称稍作更改，以避免冲突。

“运行时配置信息”页面

“**运行时配置信息**”页面显示代理程序中的可配置变量。在受监视主机上安装代理程序时，您可以为变量设置值。

这些值可用于整个环境中的命令返回码和脚本。要打开“**运行时配置信息**”页面，请单击“**代理程序信息**”页面的“**代理程序内容**”部分中的**运行时配置**，或单击“大纲”视图中的**运行时配置**节点。Agent Builder 通过产品代码和标签自动构造环境变量的名称。

您可以使用“**运行时配置信息**”页面来添加和更改配置属性以及提供缺省值。

“代理程序 XML 编辑器”页面

“**代理程序 XML 编辑器**”页面显示代理程序定义的 XML。

代理程序定义 XML 包含 Agent Builder 所有其他部分中显示的信息。如果更改 XML，那么 Agent Builder 中显示的信息也会反映此更改。



注意: 请勿在 XML 中执行任何更改。此类更改可能导致错误，阻止生成代理程序或者对代理程序运行造成负面影响。

保存编辑和更改

在保存使用编辑器所做的更改之前，不会存储这些更改。

过程

- 通过以下某种方法来执行保存：
  - 选择**文件 > 保存**，这相当于选择保存（软盘）图标。
  - 按 **Ctrl+S**。

在保存时，系统将进行验证以确保信息完整。如果发生问题，那么 Eclipse 的“**问题**”视图中将显示关于错误的信息。如果此视图不可见，请选择**窗口 > 显示视图 > 问题**。如果尝试生成的代理程序有错误，将显示错误消息。

注: 必须先更正所有错误并保存更改，然后才能生成和安装代理程序。

落实代理程序的版本

如果您确定已完成了此版本的代理程序的开发，并且已准备好交付，请落实代理程序。

关于此任务

IBM Tivoli Monitoring 系统要求新版本代理程序包含之前在监视环境中使用的该代理程序先前版本中包含的所有信息。如果新代理程序安装在一些受监视主机上，但旧代理程序保留在其他主机上，那么必须包含先前版本的所有信息，工作空间、情境和查询才能够继续工作。

在完成代理程序开发和测试之后，必须将代理程序落实为某个版本号的最终版本。Agent Builder 确保在落实代理程序之后不会移除信息。后续构建代理程序时会使用新版本号。

最多只能有 1024 个版本。

切记: 如果您更改了要在 IBM Cloud Application Performance Management 环境中测试和运行的代理程序，则必须更改代理程序版本。

过程

1. 打开 **Agent Editor** 窗口的“**代理程序信息**”页面。
2. 在**落实代理程序版本**区域中，单击**落实此级别**。
3. 备份已落实的代理程序，或者将其检入到版本控制系统中。

下一步做什么

落实代理程序后，对代理程序的其他任何更改都成为新版本的一部分。您必须先输入新版本号，然后才能保存其他更改。对新版本的任何更改均不得中断与先前版本代理程序的兼容性。

落实代理程序后，不能对落实该代理程序之前就存在的对象执行下列操作：

· 从属性组中删除属性。
· 删除属性组。
· 对属性组中的现有属性重新排序。
· 重新组织现有属性组（使用导航器项）。
· 将属性组或导航器组移入或移出子节点。
· 重命名属性组。
· 重命名属性。
· 更改现有属性的数据类型。
· 更改子节点名称或类型（如果其中包含在落实代理程序前就存在的属性组）。
· 更改代理程序的公司标识或代理程序标识。
· 更改代理程序的产品代码。有关更多信息，请参阅第 1003 页的『更改产品代码』。

在落实代理程序后，可以执行下列操作：

· 将新属性添加到现有属性组。
· 添加新属性组。
· 对新属性重新排序。
· 使用导航器项组织新属性组。
· 创建新的子节点类型。
· 添加新查询。

· 添加新情境。

· 添加新工作空间。

为代理程序设置新版本号

要保存对已落实的代理程序的更改，您必须输入新的版本号。

过程

1. 打开 **Agent Editor** 窗口的“**代理程序信息**”页面。
2. 在“版本”提示后输入高于当前级别的版本、修订路径或补丁级别。
3. 对代理程序进行编辑。

提示: 如果您落实了代理程序但忘记更改代理程序版本，系统会在您保存任何更改时提示您输入新版本。

更改产品代码

如果更改产品代码，那么您的代理程序将与该代理程序的任何先前版本都不兼容。先前落实操作的所有记录都会丢失，并且您将开发新的代理程序。

任何从 IBM Tivoli Monitoring 中导出并导入到代理程序中的文件、情境、“执行操作”命令或工作空间都会从该代理程序中删除。

如果您尝试更改已落实的代理程序的产品代码，那么 Agent Builder 会显示警告并询问您是否要继续。

在“**代理程序产品代码**”窗口中单击**是**之后，系统会显示警告，指出代理程序支持文件的内容不再有效。系统还将发出警告，指出下一次保存代理程序时将移除这些文件。

编辑数据源和特性属性

在向代理程序添加数据源时，Agent Builder 会创建对应的数据集。您可以编辑数据集和其中的特性以提供必要的监视信息。

过程

要编辑或移除数据集（属性组）中的信息：

1. 在“**代理程序信息**”页面的“**代理程序内容**”区域中，单击**数据源**。

   “**数据源定义**”页面将打开。
2. 选择数据集（属性组）。

   该页面的属性组信息区域将更新以显示所选数据集的属性。

   注: 或者，如果您处于“**代理程序**”向导的最后一页，则可双击数据源以打开“**属性组信息**”窗口。此窗口包含与“**数据源定义**”页面的属性组信息区域相同的信息。

   第 1003 页的表 260 描述了适用于所有数据源的字段信息。使用这些字段可修改所选数据源或特性的属性。

表 *260.* 用于编辑数据源的字段

字段名称	描述	可接受的值和示例
属性组名称	Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示的数据源的名称	可接受的值：长度小于或等于 32 个字符的描述性字符串。它在代理程序中必须唯一。第一个字符必须是字母，其余字符可以是字母、数字或下划线。下划线显示为空格。请勿使用空格或特殊字符。
帮助文本	数据源的帮助文本	可接受的值：最多 256 个字符的字符串。

表 260. 用于编辑数据源的字段 (续)		
字段名称	**描述**	**可接受的值和示例**
Produces a single data row（产生单个数据行）	数据源返回 1 行数据。在所有采样数据源中均可编辑。	示例：如果要监视物理系统内存，请选择单行。系统通常在单个池中管理其所有内存；因此只能返回一行数据。
可生成多个数据行	该数据源可以返回任意数量的数据行。在所有采样数据源中均可编辑。	示例：如果要监视磁盘驱动器，请选择多个行，因为系统中可能有多个磁盘。对于键，请选择可区分磁盘的特性。对于磁盘，键特性是盘号、盘符、卷标或您环境中适用的任何特性。
生成事件	数据源返回基于事件的数据，每个事件返回 1 行数据。	示例：超出性能阈值时，基于 SNMP 事件的数据源发送通知（陷阱）。 **注:** 并非所有数据源都能够生成事件。
将此特性添加至报告类别	这是生成的 Cognos 模型中，将此属性组中的特性分配到的类别。	如果选中此复选框，那么此属性组将放入 Key Metrics 文件夹中的所选子文件夹（Availability 或 Performance）。如果未选中此复选框，那么属性组会放置在 Cognos 数据模型中的**扩展度量**文件夹中。
度量类别	这是此属性组中的特性将分配到的类别。	请选择**性能**或**可用性**。

注:

a. **生成单一数据行**和**可生成多个数据行**字段不影响事件数据源的数据。

b. 有关已采样和事件数据类型的更多信息，请参阅第 1021 页的『数据类型』。

c. 有关特定数据源字段的信息，请参阅第 1025 页的『定义和测试数据源』中的相关数据提供程序信息。

创建、修改和删除属性

您可以在数据集（属性组）中创建、修改或删除属性。

要处理属性，请打开“**数据源定义**”页面。有关更多信息，请参阅第 1000 页的『“数据源定义”页面』。

创建属性

您可以向数据集添加新属性。

过程

1. 右键单击该数据源，然后从菜单中选择**添加属性**。
   “**属性信息**”页面随即显示。

   注: 显示的页面取决于属性的数据源。
2. 在“**属性信息**”页面上指定新属性的选项。

   有关这些字段及选项的信息，请参阅第 1007 页的『用于定义特性的字段和选项』。
3. 要添加更多属性，请选择**添加其他属性**，并单击**下一步**。
4. 完成添加属性时，请单击**完成**。

复制属性

可以从“**数据源定义**”页面中复制属性。

过程

1. 在 Agent Editor 的“**数据源定义**”页面中，右键单击要复制的属性，然后单击**复制属性**。
2. 在“**复制属性**”窗口中的**名称**字段中，输入新属性的名称，然后单击**确定**。

编辑特性

您可以使用“**数据源定义**”页面编辑和更改特性信息。

过程

1. 选择要编辑的特性。

   该页面的“**特性信息**”窗格将更新以显示选定特性的属性。
2. 指定对新特性信息的选择。

   注: 在“**代理程序**”向导的最后一页（“**数据源定义**”页面）上，可以双击特性以打开“**特性信息**”窗口。该窗口包含与“**数据源定义**”页面的“特性信息”窗格相同的信息。

创建派生属性

您可以创建从其他属性派生值（而不是直接从数据源派生值）的属性。

关于此任务

在派生的属性中，可对源属性的值执行运算。例如，对数字属性执行基本的算术运算或对字符串属性执行字符串并置。

用于派生表达式的基本表达式语法包含函数。这些函数提供了较复杂的数据处理，包括短期聚集、从字符串到整数的转换以及访问配置属性和环境变量。另外，还提供了一个编辑器，可帮助您在构建表达式时进行显示。

过程

1. 在“**数据源定义**”页面上，右键单击数据源，然后单击**添加属性**。
2. 在“**属性信息**”页面上，输入属性名称和帮助文本。
3. 选择**从其他属性值派生**
4. 在**公式**字段中，输入公式文本，或者单击**编辑**以使用图形编辑器来输入公式。

   有关可以在公式中使用的运算符及函数的信息，请参阅第 1015 页的『公式运算符和函数』。

   注: 当您单击**编辑**时，公式编辑器将打开。有关编辑派生属性的信息，请参阅第 1006 页的『编辑派生属性』。
5. 可选： 选中或取消选中**特定于区间的计算**复选框，以确定计算函数时要使用哪两个属性样本值。

   如果您的公式使用了 `rate` 或 `delta` 函数，请使用此选项。有关**特定于区间的计算**的更多信息，请参阅第 1006 页的『特定于区间的计算』。有关 `rate` 和 `delta` 函数的更多信息，请参阅第 1015 页的『公式运算符和函数』。
6. 在**属性类型**区域中，单击属性类型。
7. 单击**确定**。

   数据源定义页面将再次显示，并像以前一样列出数据源。
8. 单击**完成**。

   要点: 如果您所创建的派生属性要引用另一个派生属性，请确保所引用的属性在新属性之前列出。如果属性引用在清单中稍后位置列出的另一个派生属性，那么代理程序将无法显示此属性的值。如果您创建这样的属性，那么 Agent Builder 将显示警告。

特定于区间的计算
在定义基于 rate 或 delta 函数的派生属性时，可以选择**特定于区间的计算**。

您可在“**属性信息**”页面的“**派生属性详细信息**”选项卡上选择**特定于区间的计算**。有关更多信息，请参阅第 1005 页的『创建派生属性』。

在使用**特定于区间的计算**选项时，了解变化量（属性值之间的差值）概念很重要。变化量是指属性最新值和属性先前值的差值。变化量由 delta 函数直接返回，并由 rate 函数用于计算结果。

delta 或 rate 函数必须始终以 last 函数作为其唯一自变量。last 函数指定使用属性的哪些值来确定变化量。如果未选择**特定于区间的计算**，则使用的先前值始终是第二新的值。如果选择了**特定于区间的计算**，则使用的先前值是存在时间（相对于最新值）等于请求者收集时间间隔的值。

例如，假设 CDP_DP_REFRESH_INTERVAL 设置为 120 秒，attribute A 具有以下采样值：

时间	**采样值**
最新	2800
2 分钟（120 秒）前	2600
4 分钟（240 秒）前	2499
6 分钟（360 秒）前	1500
8 分钟（480 秒）前	1200
10 分钟（600 秒）前	1000

如果未选择**特定于区间的计算**，那么 delta 函数始终返回 200，即两个最新值的差值，计算公式为 2800 - 2600。无论此值是显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中、用于情境还是历史数据收集，都返回相同值。

如果选择了**特定于区间的计算**，那么 delta 函数返回依赖于请求者收集时间间隔的值。

如果某派生属性与 delta 函数用于收集时间间隔为 4 分钟的情境，那么 delta 函数返回的值是 301，即最新值与 4 分钟前获取的值的差值，计算公式为 2800 - 2499。

如果某派生属性与 rate 函数用于收集时间间隔为 10 分钟（600 秒）的情境，那么 rate 函数返回的值是 3，即最新值与 10 分钟前获取的值的差值除以此时间间隔秒数，计算公式为 (2800 - 1000) / 600。

注: Tivoli Enterprise Portal 没有固有的收集时间间隔，因此 Tivoli Enterprise Portal 的变化量和速率计算始终使用最新属性值和第二新属性值，无论是否选择了**特定于区间的计算**，结果都相同。

为使 delta 和 rate 可以与**特定于区间的计算**正确地配合使用，

- 代理程序必须定期地在后台收集数据，而不是按需收集（CDP_DP_THREAD_POOL_SIZE 必须大于 0）。
- 使用属性的每个情境或历史数据收集时间间隔必须是后台刷新时间间隔 (CDP_DP_REFRESH_INTERVAL) 的倍数。
- count（last 函数的第二个自变量）的大小必须足以容纳情境或历史数据收集的最大的收集时间间隔。例如，如果代理程序必须支持 10 分钟（600 秒）的历史数据收集，并且 CDP_DP_REFRESH_INTERVAL 是 120 秒，那么 count 必须至少为 6，计算公式为 1+(600 / 120)。count 值为 6 可确保 last 函数返回最新的采样，并且多达过去 600 秒的采样。

注: 如果不满足这些条件，那么输入值很可能无效，并返回结果 0。

编辑派生属性
使用 Formula Editor 编辑派生属性。

“**属性信息**”页面上针对派生属性提供了 Formula Editor，如第 1005 页的『创建派生属性』所述。有关 Formula Editor 的更多信息，请参阅第 1011 页的『Formula Editor』

移除属性
您可以使用“**数据源定义**”页面移除一个或多个属性。

过程

· 要移除一个或多个属性，右键单击属性，然后从显示的菜单中选择**移除**。

注: 无法移除由派生属性使用的属性。必须先移除派生属性对您要移除的属性的引用。

用于定义特性的字段和选项
特性信息页面上适用于所有数据源的字段信息及选项的描述

有关每个数据源的特定字段信息的信息，请参阅每个数据源的相关文档。

表 *261.* 用于定义特性的字段和选项

字段名称/选项	**描述**	**可接受的值**
特性名称	特性的名称，此名称将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中	包含以下字符的字符串: · A-Z · _ · a-z · 0-9 **注:** 名称必须以 A-Z 或 a-z 开头。 特性名限长 63 个字符，属性组名限长 63 个字符
帮助文本	特性的帮助文本	字符串
隐藏 - 只能用于派生特性	如果选中此选项，那么此特性不会显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中。请参阅最后一行中的注释。	不适用
从其他特性值派生	特性值将根据其他特性的值进行计算	不适用
键特性	特性是表中的键。请检查此特性是否有助于唯一定义所报告的对象。 如果对数据进行仓储和摘要，那么将使用键特性在摘要表中进行数据汇总。	此选项不适用于 Perfmon 特性。
特性信息窗格	此选项卡的内容取决于此特性所属的数据源的类型。请参阅本章中有关您要监视的数据源的信息，以获取更多详细信息。 对于派生特性，请在**公式**字段中输入公式，以计算基于其他特性或常量的特性值。您可以在**公式**字段中输入公式，或者单击**编辑**以使用图形公式编辑器。请参阅第 1011 页的『Formula Editor』。	

表 *261.* 用于定义特性的字段和选项 *(续)*		
字段名称/选项	**描述**	**可接受的值**
特性类型	描述此特性在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中的显示方式。共有 3 种类型: · 字符串 · 数字 · 时间戳记 第 1008 页的『属性类型』包含有关特性类型的更多信息。	第 1009 页的表 262 包含数字特性类型值的描述。
枚举	这可以是不带小数位的数字或字符串值。	请通过使用第 1010 页的『指定属性的枚举』中的过程，将您的枚举项添加到表中。 在此特性中接收到来自代理程序的对应值时，枚举名称将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中。 此特性用于一组具有指定含义的特定值（例如，1 代表 UP，2 代表 DOWN）。
注: 在计算中将此特性与其他特性配合使用时，有多个理由支持不显示基本值。例如，表示字节计数的数字回绕太快，因此毫无用处。		

属性类型
共有三种属性类型

这三种属性类型是:

· 字符串

· 数字

· 时间戳记

字符串属性

当您选择**字符串**时，请使用**最大大小**字段来指定字符串的最大长度（字节）。缺省大小是 64 个字节。

字符串值可以包含任何 UTF-8 字符。最大大小是为了包含该字符串而分配的缓冲区的总长度（字节）。某些非 ASCII UTF-8 字符占用多个字节，因此您选择最大大小时必须考虑到此部分空间。仓库中的数据聚集将显示期间内收集到的最新值。

数字

当您指定**数字**时，可以设置多个选项。请参阅第 1009 页的表 262，以获取有关这些选项的信息。

时间戳记

时间戳记属性是遵循 CYYMMDDHHMMSSmmm 格式（其中 C 为 1 时表示 21 世纪）的字符串属性。对于脚本或套接字客户机，这 16 个字符必须全部使用。在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示时，时间戳记属性类型将以语言环境的正确格式显示。

在使用 WMI 的浏览功能时，Agent Builder 会自动将 CIM 类型为 CIM_DATETIME 的属性标记为时间戳记。数据提供程序会自动将 WMI 属性转换成此格式。

属性的数字方面
对属性的大小、用途、精度和范围方面的描述。

指定数字属性时，必须指定属性的大小、用途、精度和范围。有关更多信息，请参阅第 1009 页的表 262。

表 *262.* 数字属性选项

数字方面	选项和字段	描述
大小	**32 位** **64 位**	32 位数字值的范围可以从 -2147483648 到 2147483647（大约从 -2,000,000,000 到 2,000,000,000）。 64 位数字值的范围可以从 -9223372036854775808 到 9223372036854775807（大约从 $-9x10^{18}$ 到 $9x10^{18}$）
用途	**标尺**	返回的原始值大于或小于先前值的整数值。支持负值。此类型为整数的缺省类型。仓库中的数据聚集产生最小值、最大值和平均值。
	计数器	这是正整数值，包含通常随时间推移而增加的原始值。仓库中的数据聚集显示增量值的总值、上限值、下限值以及最新值。在以下基于增量的计算示例中（一个小时内的详细数据值为 9、15、12、20 和 22），基于增量的处理具有下列规则： · 如果当前值大于或等于先前值，那么输出等于先前值减去当前值 · 如果当前值小于先前值，那么输出等于当前值 · 由于 15 大于 9，因此输出等于 6 · 由于 12 小于 15，因此输出等于 12 · 由于 20 大于 12，因此输出等于 8 · 由于 22 大于 20，因此输出等于 2 · TOT_ 值为 28，这是输出总和 · LOW_ 值为 2，这是最小的输出 · HI_ 值为 12，这是最大的输出
	属性	这是不常变化的对象的属性。仓库中的数据聚集将显示在周期内收集到的最新值。
	变化量	此整数值表示此属性的当前值和先前值的差值。由于此属性在仓库中表示为标尺，因此仓库中的数据聚集将产生最小值、最大值和平均值。
	变化百分比	此整数值表示当前值和先前值的变化百分比。此类型的计算方法为：((new -old)*100)/old。由于此类型在仓库中表示为标尺，因此仓库中的数据聚集将产生最小值、最大值和平均值。
	变化率	这是一个整数值，表示的含义是当前值与先前值的差值除以采样间隔秒数。它将“值（例如字节数）”转换为“值/秒（字节数/秒）”。由于此类型在仓库中表示为标尺，因此仓库中的数据聚集将产生最小值、最大值和平均值。

表 262. 数字属性选项 (续)		
数字方面	**选项和字段**	**描述**
小数位	**小数位调整**	“小数位”确定数字中有多少小数位。每个小数位以因子 10 来缩小之前提到的范围。例如，小数点调整为 2 时，将显示两个小数位，而在 32 位数字中，允许的范围变成 -21474836.48 到 21474836.47。 指定非零的小数点调整时，会在内部将数字处理为浮点数。因此，较大的 64 位数字的精度可能会降低。
范围	**最小值** **最大**	“范围”给出值的预期范围。如果没有给出最小值或最大值范围，那么将使用先前描述的最大值。此范围用于在某些图形 Tivoli Monitoring 工作空间视图中产生更有用的初始视图。

指定属性的枚举
使用“**属性信息**”页面指定值枚举。

关于此任务
为属性指定枚举是一个简短的过程。当遇到定义了枚举的值时，Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中会显示此枚举名称，而不是值。

过程

1. 在“**属性信息**”页面的**属性类型**区域中，单击**数字**。
2. 在**枚举**区域中，单击枚举，然后单击**添加**。
   这将显示“**枚举定义**”窗口。
3. 在窗口的字段中输入枚举的名称和值。
4. 单击**确定**。
   然后，您可以添加更多的枚举。

指定用作状态指示符的特性的严重性
在 IBM Cloud Application Performance Management 环境中，摘要仪表板必须显示状态。您必须使用特性来提供状态值。对于此特性，必须指定值，以表示特定的状态严重性。

关于此任务
用于指示状态的特性必须是数字。请在“**仪表板设置**”向导中选择此特性；有关如何使用此向导的指示信息，请参阅第 1154 页的『为 Cloud APM 准备代理程序』。

您可以为特性指定对应于“正常”、“警告”和“严重”严重性的值。任何其他值均表示“未知”严重性状态；您还可以将某些值显式定义为“未定义”，对于这些值，用户界面会显示“未知”状态。

过程

1. 选择要编辑的特性。
   页面的“特性信息”窗格将更新，以显示所选特性的属性。
2. 在“特性信息”窗格中，单击**严重性**选项卡。
3. 选择必要的严重性（“正常”、“警告”、“严重”和“未定义”），并单击**编辑**。
4. 选择**范围**或**单个数字**，输入值范围或单个数字值，然后单击**确定**。
5. 可选： 如果需要为同一严重性添加其他值（例如，2 和 25 都表示警告），请单击**添加**，选择严重性，输入值，然后单击**确定**。

过滤属性组

您可以创建过滤器，以限制从返回采样数据的属性组返回的数据。

开始之前

如果存在属性组，打开“**数据源定义**”页面。有关更多信息，请参阅第 1000 页的『“数据源定义”页面』。

如果要创建属性组，请执行第 989 页的『定义初始数据源』中的步骤，然后单击初始数据源信息页面中的**高级**。

过程

1. 使用下列其中一个步骤以开始创建过滤器：
   - 如果要创建属性组，单击初始数据源信息页面中的**高级**。
   - 如果存在属性组，请在“**数据源定义**”页面中选择该属性组，然后在“**数据源定义**”页面中单击**高级**。
2. 在“**高级数据源属性**”页面中，输入选择公式。您输入的选择公式的求值结果必须是布尔值 true 或 false。

   在“**高级数据源属性**”页面中，您可以单击**编辑**，以使用公式编辑器来输入或修改公式。有关 Formula Editor 的更多信息，请参阅 第 1011 页的『Formula Editor』
3. 在完成过滤器选择公式输入之后，单击**确定**直到返回到“**数据源定义**”页面。

   创建过滤器后，代理程序会使用该过滤器对每行数据求值。当某一行数据的过滤器求值为 *true* 时，此数据将发送到 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management。当过滤器求值为 *false* 时，不发送此行数据，会将其废弃。

下一步做什么

您可以使用属性组的测试功能来验证过滤器是否按预期工作。有关属性组测试的更多信息，请参阅第 1156 页的『属性组测试』

Formula Editor

使用 Formula Editor 创建和更改 Agent Builder 中的公式。

Formula Editor 是一个图形工具，在执行以下任务时显示：

1. 创建或编辑派生属性，请参阅第 1005 页的『创建派生属性』和第 1006 页的『编辑派生属性』
2. 创建过滤属性组，请参阅第 1125 页的『创建过滤属性组』
3. 从属性组过滤数据，请参阅第 1011 页的『过滤属性组』



注意：

- 在创建派生属性时，所创建的公式必须产生与此属性类型匹配的数据类型。例如，如果派生属性类型是数字，那么所创建的公式必须求值为数字结果。
- 在创建过滤属性组或从属性组过滤数据时，所创建的公式必须产生布尔值“true”或“false”。

注：在下列视图中，显示了 Formula Editor 正在为派生属性创建公式。在将 Formula Editor 与过滤属性组或从属性组过滤数据结合使用时，这些视图是相同的。这些视图根据用途显示标题“**派生 Formula Editor**”或“**过滤器 Formula Editor**”。

显示 Formula Editor 时，当前公式会装入编辑器。如果公式不存在，可以通过在 **Formula Editor** 窗口中的公式空白处直接输入来输入公式。或者，可以单击**插入**以使用编辑器菜单选项开始输入公式。此编辑器在缺省窗口中包含两个公式视图，以及第三个视图的选项：

组件视图（缺省）
所编辑公式的组件显示在“**操作数**”区域和**运算符**字段中。可以使用选择菜单来编辑运算符以及它的两个操作数。

公式视图（缺省）
完整公式位于该窗口中的公式字段中。通过在此框中输入内容可以编辑公式。

公式层次结构树视图（选项）
可通过选中**显示公式层次结构**复选框来显示公式层次结构树。在以后调用 Formula Editor 时会记住该复选框的状态。

更改 Formula Editor 组件视图

更改 Formula Editor 中的组件视图。

关于此任务

可通过以下方法更改组件视图中显示的组件：

过程

- 将光标移到公式文本中。
- 选择公式层次结构树中的其他节点。
- 选择**上移一层**或其中一个编辑按钮。

组件类型

您可以使用 Formula Editor 来编辑当前组件或者该组件的任何操作数或函数自变量。某些组件在 Formula Editor 中选中后可能会有不同的显示方式。

Formula Editor Attribute 组件

使用 Formula Editor 中的 Attribute 组件可选择和操作公式中的属性。

关于此任务

您可以在 Formula Editor 的组件视图中，从属性列表选择属性组的属性。

过程

1. 要处理特定属性，从列表中选择该属性，然后单击**编辑**。
   这将显示“**编辑选定的属性**”窗口。
2. 可以通过以下方法操作所选属性：
   - 您可以选择**字符串**或**数字**，将属性替换为字符串或数字。属性列表将替换为输入字段，且其内容将不再与有效属性名列表进行比较。
   - 您可以单击**函数**将属性替换为函数。这样会在名称后面添加圆括号，并且列表中现在将包含可供选择的有效函数名。
   - 可以输入属性名称而不是选择一个名称。如果尚未定义此属性组中的所有属性，那么输入名称将非常有用。
     - 如果不存在具有所输入名称的属性，那么将显示警告。
     - 如果输入的字符不能用作属性名称的一部分，那么将显示错误。
     - 在更正警告或错误之前，**确定**按钮处于禁用状态。
   - 不会基于类型来过滤属性。如果选择或输入了错误类型的属性（或任何值），将显示警告消息。

Formula Editor Literal 组件

使用 Formula Editor 中的字符串和数字组件来操作公式中的字面值。

关于此任务

字面值是在公式中直接输入的任何值，它不来自属性值或函数。字面值可以是字符串或数字。

过程

- 您可以通过单击**属性**将文字串或数字替换为属性。在选择或输入有效的属性名称时不得带引号。

- 您可以通过单击**函数**将文字串或数字替换为函数。这样会在名称后面添加圆括号，并且选择列表将包含可供选择的有效函数名。
  - 如果在需要字符串的地方输入了数字，那么将显示警告，反之亦然。
  - 如果选择了**数字**，而字段内容不是数字，那么将显示错误。在更正错误之前，**确定**处于禁用状态。

Formula Editor Operator 组件

使用 Formula Editor 中的 Operator 组件来操作公式中的运算符。

关于此任务

Operator 组件显示运算符及其操作数。

过程

- 在 Formula Editor 组件视图中，从**运算符**列表中选择两个操作数之间的运算符。运算符 (%) 会将第一个操作数乘以 100，然后除以第二个操作数。
- 选择运算符 (+、 -、 *、 / 或 %) 。
  - 页面的**左操作数**部分位于运算符前面。
  - **右操作数**部分位于运算符后面。
  - 如 第 1012 页的『Formula Editor Attribute 组件』和 第 1012 页的『Formula Editor Literal 组件』所述，可以编辑简单的操作数（属性和字面值），而无需将所选组件更改为操作数。
  - 可通过单击**编辑**来编辑复杂操作数（由其他运算符或函数组成）。此操作会突出显示操作数组件而不是整个运算符。

Formula Editor 条件表达式组件

条件表达式组件显示一个条件、条件为 true 时返回的值以及条件为 false 时返回的值。

- “**条件**”部分中的表达式求值为 true 或 false。运算符 (==)、(!=)、(<)、(<=)、,(>)、(>=)、(&&)、(||)、(!) 可用于构成返回 true 或 false 的表达式。
- 如 第 1012 页的『Formula Editor Attribute 组件』和 第 1012 页的『Formula Editor Literal 组件』所述，可以编辑简单的操作数（属性和字面值），而无需将所选组件更改为操作数。
- 可通过单击 Edit 来编辑复杂操作数（由其他运算符或函数组成）。此操作会突出显示操作数组件而不是整个条件表达式。
- 请参阅 第 1014 页的『Formula Editor 通用选项』以了解有关使用以下选项的信息：**插入**、**移除**、**上移一层**和**编辑**。

相关概念

第 1011 页的『Formula Editor』
使用 Formula Editor 创建和更改 Agent Builder 中的公式。

Formula Editor Function 组件

使用 Formula Editor 中的 Function 组件可选择和操作公式中的函数组件。

关于此任务

Function 组件显示函数及其自变量。

过程

- 要使用函数，可从 Formula Editor 的列表中选择**函数名**。
  - 选定函数的描述将显示在函数后。
  - **函数自变量**部分显示在函数名后面。将显示选定函数的相应自变量数量。将显示特定于选定函数的描述。

- 如 第 1012 页的『Formula Editor Attribute 组件』和 第 1012 页的『Formula Editor Literal 组件』所述，可以编辑简单的自变量（属性和字面值），而无需将所选组件更改为操作数。
- 可通过单击**编辑**来编辑复杂的自变量（由运算符或其他函数组成）。此操作会突出显示自变量组件而不是整个函数。

· 对于自变量个数不固定的函数，除了 第 1014 页的『Formula Editor 通用选项』中描述的操作之外，还可以单击**插入**来添加自变量，或单击**移除**来移除自变量。

· 对于 getenv 函数，可通过单击**插入**来选择配置属性。如果选择“配置属性”选项，则会显示“**配置属性**”窗口。

Formula Editor 通用选项
您可以在 Formula Editor 的所有视图中使用某些选项

Formula Editor 通用选项包括：

· **插入**
· **卸下**
· **上移一层**
· **编辑**

插入

插入在组件前面插入运算符或函数。该组件将降级为运算符的一个操作数或函数的一个自变量。例如，如果在 sqrt(attr2) 函数前单击**插入**，系统将询问您要插入什么内容，并将显示以下选项：

· **以 sqrt(attr2) 作为其中一个运算符操作数的运算符**
· **以 sqrt(attr2) 作为函数第一个自变量的函数**
· **以 sqrt(attr2) 作为 true 或 false 值的条件表达式**

如果在 getenv 函数前单击**插入**，系统将询问您要插入什么内容，并将显示以下选项：

· **配置属性**：使用此选项来检索已为代理程序设置的配置属性的值，或者检索运行代理程序的主机上任何环境变量（例如，JAVA_HOME）的值。
· **以 attr2 作为其中一个运算符操作数的运算符**
· **以 attr2 作为函数第一个自变量的函数**
· **以 attr2 作为 true 或 false 值的条件表达式**

移除

移除仅可用于运算符和函数，与**插入**效果相反。您单击**移除**时，系统将询问您用什么来替换移除的运算符或函数。例如，sqrt(attr2) 函数前的**移除**将显示以下选项：

· **当前自变量 1，attr2**
· **新字符串、数字或特性引用**

选择**新字符串、数字或特性引用**将废弃待移除点之后的整个树，并将其替换为新特性或字面值。

单击**当前自变量**可提升所选操作数或自变量以替换移除的运算符或函数。如果有更多自变量或操作数，您可以单击其后的选项。将废弃任何其他操作数或自变量。

上移一层

单击**上移一层**将在树中上移。

编辑

单击复杂操作数或自变量前的**编辑**可使其成为待编辑组件。

在单击**编辑**之后单击**上移一层**可将当前组件复原为单击**编辑**之前的状态。

Formula Editor - 公式错误
更正 **Formula Editor** 中的公式错误

此组件视图在没有公式或无法解析输入的公式时有所不同。它不显示公式树。而是显示错误消息。

要更正含有解析错误的公式，可以直接在公式字段中进行输入，也可以单击**插入**将其替换为新公式。在此情况下，**插入**呈现以下选项：

- **属性**
- **字符串**
- **数字**
- **运算符**
- **条件表达式**
- **函数**

相关概念
第 1011 页的『Formula Editor』
使用 Formula Editor 创建和更改 Agent Builder 中的公式。

公式运算符和函数

可以在 Formula Editor 中使用的公式运算符和函数的参考（包括示例）。

派生的属性值是根据同一数据源中的常量和其他属性值对表达式求值的结果。表达式语法是常规数学表达式 - 操作数 运算符 操作数（以及用于分组的圆括号）。通过使用一般数学运算符 +、-、*、/ 和 % 来组合数字属性和其他数字属性或常量，% 用于将**左操作数**乘以 100，然后再除以**右操作数**。可使用 + 将字符串属性与其他字符串属性或常量组合。还可以使用以下描述的函数。使用以下格式输入函数：`function_name(argument_1, argument_2, argument_3)`。

属性通过其名称表示（与您在“**数据源**”信息树中看到的名称相同）。整数常量指定为数字。字符串常量用引号括起。

可以在公式中使用以下函数：

abs
返回某个数字的绝对值

atof
将字符串转换为浮点值

atoi
将字符串转换为整数值。其运算方法与一般 **C atoi** 的工作原理相同：在第一个非十进制字符处停止。

average
返回单个值，该值是一组值中的平均值。这组值来自函数的自变量。可以提供多个单独的值（例如属性名称或常量），每个值位于单独的自变量中。或者，也可将 last 函数作为此函数的唯一自变量（以计算属性的最近值的平均值）。

此函数的使用示例包括：

```
average (Attr_A, AttrB, Attr_C)
```

```
average (last (Attr_A, 10))
```

ceiling
返回不小于自变量的最小整数。

例如，如果 `attribute_a = 12.4`，那么 `ceiling(attribute_a)` 返回值 13。如果 `attribute_a = -12.4`，那么 `ceiling(attribute_a)` 返回值 -12。

delta
属性的最新值与先前针对该属性收集到的某个值的差。delta 的单一自变量必须是 `last` 函数，后者用于获取属性的当前值和先前值。常规用法如下所示：

```
delta (last(OtherAttribute, 2))
```

有关使用 last 函数中的哪些属性值来计算 delta 的更多信息，请参阅第 1006 页的『特定于区间的计算』。此函数仅适用于派生属性，而不适用于属性组过滤器。

floor
返回不大于自变量的最大整数。

例如，如果 `attribute_a = 12.4`，那么 `floor(attribute_a)` 返回值 12。如果 `attribute_a =-12.4`，那么 `floor(attribute_a)` 返回值 -13。

getenv
返回提供的环境或“配置变量”的值。

ipAddressToName
将 IP 地址转换为主机名。此函数需要一个自变量（采用点分十进制表示法的 IP 地址字符串）。如果无法解析该地址，那么会返回 IP 地址。

itoa
将整数转换为字符串。当您想要将数字值连接成字符串时，此函数非常有用。派生字符串 + 函数仅采用两个字符串自变量。

last
返回一列值以供 `min`、`max`、`average`、`stddev`、`rate` 和 `delta` 函数使用。它采用两个自变量：要收集的属性以及要在计算中使用的值的数量。如果所需的属性是“用字符串表示的整数值”属性，那么第一个自变量可包含 `atoi` 函数，例如 `atoi(numericalStringAttribute)`。第二个自变量必须为数字。它可以硬编码为常量，也可以是 `atoi(getenv("ENV_VAR"))` 表达式的结果。它不能引用任何属性值。

此函数的使用示例包括：

```
average (last (Attr_A, 10))
```

```
last (Attribute_A, ${K01_NUM_COLLECTIONS}))
```

限制: 在特定公式中，只能使用一次 `last` 函数。

matches
返回布尔值 true 或 false，以指示正则表达式是否与某个值匹配。此函数采用两个自变量，即字符串源以及结果要与此字符串进行比较的正则表达式。此函数对于过滤属性组而言非常有用。

max
返回单个值，该值是一组值中的最大值。这组值来自函数的自变量。可以提供多个单独的值（例如属性名称或常量），每个值位于单独的自变量中。或者，也可将 last 函数作为此函数的唯一自变量（以计算属性的最近值的最大值）。

min
返回单个值，该值是一组值中的最小值。这组值来自函数的自变量。可以提供多个单独的值（例如属性名称或常量），每个值位于单独的自变量中。或者，也可将 last 函数作为此函数的唯一自变量（以计算属性的最近值的最小值）。

nameToIpAddress
将主机名转换为 IP 地址。此函数需要一个自变量，即主机名字符串。如果无法解析该地址，那么会返回主机名。

NetWareTimeToTivoliTimestamp
将 Novell NetWare 十六进制时间值转换为 Tivoli Monitoring 时间戳记。此函数需要一个自变量，即一个特殊的 NetWare 十六进制时间值。属性类型为时间戳记。

rate
属性的最新值与先前针对该属性收集到的某个值的变化率（每秒）。rate 的单一自变量必须是 `last` 函数，后者用于获取属性的当前值和先前值。常规用法如下所示：

```
rate (last(OtherAttribute, 2))
```

有关使用 last 函数中的哪些属性值来计算 rate 的更多信息，请参阅第 1006 页的『特定于区间的计算』。此函数仅适用于派生属性，而不适用于属性组过滤器。

replaceFirst
将匹配正则表达式的第一个子串替换为替换字符串。此函数采用三个自变量。第一个：输入字符串。第二个：用于匹配输入字符串中的子串的正则表达式。第三个：替换字符串。有关正则表达式和替换字符串中允许使用的替换值的详细信息，请参阅（第 1259 页的『ICU 正则表达式』）。

replaceAll
将匹配正则表达式的所有子串替换为替换字符串。此函数采用三个自变量。第一个：输入字符串。第二个：用于匹配输入字符串中的子串的正则表达式。第三个：替换字符串。有关正则表达式和替换字符串中允许使用的替换值的详细信息，请参阅（第 1259 页的『ICU 正则表达式』）。

round
将数字四舍五入至最近的整数。

sqrt
返回数字的平方根

stddev
返回单个值，该值是一组值的标准差。这组值来自函数的自变量。可以提供多个单独的值（例如属性名称或常量），每个值位于单独的自变量中。或者，也可将 last 函数作为此函数的唯一自变量（以计算属性的最近值的标准差）。

StringToTivoliTimestamp
将日期和时间字符串转换为 Tivoli Monitoring 时间戳记。此函数需要两个自变量。第一个自变量是以自由格式字符串表示的时间戳记。第二个自变量是识别如何解析以自由格式字符串表示的时间戳记的格式字符串。（第 1017 页的表 263）描述了有效格式参数。属性类型为时间戳记。

表 *263. StringToTivoliTimestamp* 的有效格式参数

符号	含义	格式	示例
y	年份	yy yyyy	*96* *1996*
M	月份 **注:** 仅支持以英语表示的月份字符串。	M 或 MM MMM MMMM	*09* *Sept* *September*
d	日期	d dd	*2* *02*
E	周历日 **注:** 仅支持以英语表示的周历日字符串。	EE EEE EEEE	*Sa* *Sat* *Saturday*
h	以 AM 或 PM 表示的钟点 (1-12)	hh	*07*
H	一天中的钟点 (0-23)	HH	*00*

表 *263. StringToTivoliTimestamp* 的有效格式参数 *(续)*			
符号	**含义**	**格式**	**示例**
m	小时中的分钟	mm	*04*
s	分钟中的秒	ss	*05*
S	毫秒	S SS SSS	*2* *24* *245*
a	AM 或 PM 标记	a 或 aa	*am*
任何其他 ASCII 字符	跳过此字符	-（连字符） （空格） /（正斜杠） :（冒号） *（星号） ,（逗号）	

第 1018 页的表 264 提供了时间戳记的字符串表示和用于解析它们的格式字符串。

表 *264. StringToTivoliTimestamp* 示例. 下表列出并说明了时间戳记字符串表示的一些示例	
时间戳记的字符串表示	**格式字符串**
96.07.10 at 15:08:56	yy.MM.dd ** HH:mm:ss
Wed, August 10, 2010 12:08 pm	EEE, MMMM dd, yyyy hh:mm a
Thu 21/01/2010 14:10:33.17	EEE dd/MM/yyyy HH:mm:ss.SS

sum
返回单个值，该值是一组值的总和。这组值来自函数的自变量。可以提供多个单独的值（例如特性名称或常量），每个值位于单独的自变量中。或者，也可将 last 函数作为此函数的唯一自变量（以计算特性的最近值的总和）。

TivoliLogTimeToTivoliTimestamp
将 Tivoli 日志文件时间戳记转换为 Tivoli Monitoring 时间戳记。此函数需要一个自变量，即来自 Tivoli 日志文件的字符串时间戳记。特性类型为时间戳记。

tokenize
标记化的字符串的一个标记。此函数需要三个自变量。第一个自变量是要分割成标记的字符串。第二个自变量给出该字符串中用于分隔各个标记的一个或多个字符。任何时候出现此自变量中的任意字符时，都将用于识别和分隔第一个自变量中的标记。第三个自变量是要作为此函数结果返回的标记的索引。第一个标记为 index 0，第二个标记为 index 1，以此类推。此自变量还可以是返回最后一个标记的字符串 LAST。

UTCtoGMT
将世界标准时间转换为 GMT Tivoli Monitoring 时间戳记。此函数需要一个自变量，即整数 time_t 值。特性类型为时间戳记。

UTCtoLocalTime
将世界标准时间转换为本地 Tivoli Monitoring 时间戳记。此函数需要一个自变量，即整数 time_t 值。特性类型为时间戳记。

以下函数不采用任何自变量并返回数字。

count
保留一个计数器，首次调用时，起始值为 1，之后每调用一次，值递增 1。如果在同时使用 last 的表达式中使用该函数，它将与 last() 存储的元素数目匹配，但此过程仅持续到 last() 达到其最大值。此时，每增加一个新值，last() 就将删除最旧的值，从而保持值总数不变，而 count() 的值会不断增加。

cumulativeSum
返回由流量控制摘要事件表示的重复事件的自变量值总和。或者如果自变量是来自数据源的单个事件，返回该自变量。它采用单一数字自变量。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

eventThreshold
返回为生成事件的属性组配置的阈值。这是一个数字，具有以下三个枚举值：

- SEND_ALL (-3)
- SEND_FIRST (-2)
- SEND_NONE (-1)

括号中的数字是原始值。但是，Agent Builder 会定义枚举，这样在缺省情况下，文本版本在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中可见。如果您指定了一个实际的数字阈值，且并非上述三个预定义选项之一，那么此函数将返回该数字。该值是大于 0 的整数。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

isSummaryEvent
如果是来自数据源的单个事件，那么返回 0，如果事件是流量控制摘要事件，那么返回 1。如果对该函数使用缺省特性，那么显示的值是 Event 和 Summary Event。如果手动创建特性，那么除非您将名称定义为枚举值，否则显示的值是 0 和 1。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

occurrenceCount
返回由流量控制摘要事件表示的匹配事件数，如果是来自数据源的单个事件，那么返回 1。（流量控制摘要事件包括第一个事件）。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

summaryInterval
返回为生成事件的属性组配置的摘要时间间隔（以秒为单位）。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

示例

使用公式运算符和函数创建派生特性和过滤特性的示例

示例 1 - 派生特性

如果您拥有定义了以下特性类型的数据源：

名称	字符串
xBytes	数字
yBytes	数字
Virtual_Size	数字

您可以定义：

- 用于表示 xbytes 和 ybytes 之和的特性 totalbytes。输入公式 xBytes + yBytes。
- 表示 yBytes 占字节总数百分比的特性 yPercent，可以定义为 yBytes % (xBytes + yBytes) 或 yBytes % totalBytes。

示例 2 - 派生特性

以下公式返回 `Virtual_Size` 特性的最近收集值中的最大值。收集的样本数是配置变量 *K4P_COLLECTIONS_PER_HISTORY_INTERVAL*（通过 `getenv` 访问）的值，该值将（通过 `atoi`）转换为数字：

```
max(last(Virtual_Size,atoi(getenv("K4P_COLLECTIONS_PER_HISTORY_INTERVAL"))))
```

示例 3 - 派生特性

以下公式返回 `xbytes` 和 `ybytes` 特性值的平方总和的平方根：

```
sqrt(xBytes * xBytes + yBtyes * yBytes)
```

示例 4 - 派生特性

以下公式返回属性组的最近 20 个样本中 `xBytes` 特性的平均值。如果自代理程序启动以来收集的样本数少于 20，那么它将返回所有样本中 `xBytes` 特性的平均值：

```
average(last(xBytes,20))
```

示例 5 - 过滤特性

您的数据源返回下列数据：

```
Name    Type  Size  Used  Free
Memory  MEM      8     4     4
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

您只对磁盘使用情况感兴趣。解决方案是创建过滤器以限制返回的数据。要限制返回的数据，您可以创建返回布尔值 true 或 false 的简单过滤器，如下所示

磁盘过滤器：

```
Type=="DISK"
```

现在，当过滤器 `Type=="DISK"` 为 true 时，此属性组仅返回磁盘使用情况数据，例如：

```
Name    Type   Size  Used  Free
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

示例 6 - 过滤属性

您的数据源返回下列数据：

```
Name   Size  Used  Free
Memory    8     4     4
Disk1   300   200   100
Disk2   500   100   400
```

返回的数据与上一示例相似，但这次没有 `Type` 属性。在这里，您可以使用 matches 函数来查找名称属性值与 `"Disk"` 后跟一个数字匹配的任何数据行。

磁盘过滤器：

```
matches(Name, "Disk[0-9]*")
```

现在，仅当此过滤器在属性 Name 中与字符串 "Disk" 后跟一个数字匹配时，才会返回磁盘使用情况数据行：

```
Name   Size  Used  Free
Disk1   300   200   100
Disk2   500   100   400
```

指定操作系统

当您定义并非在代理程序所支持的所有操作系统上均可用的数据源时，必须指定运行数据源所在的操作系统。

关于此任务

缺省情况下，数据源将在代理程序级别定义的所有操作系统上提供数据，如第 991 页的『缺省操作系统』所述。您可以为每个数据源更改操作系统。

过程

1. 要打开“操作系统”部分，添加数据源时，请在“**数据源信息**”页面中单击**操作系统**。
2. 选择运行该数据源所在的操作系统。

   请选择单个操作系统、所有操作系统、特定类型的所有操作系统或者代理程序缺省操作系统。

配置和调整数据收集

在创建 Agent Builder 代理程序后，您可以配置和调整其数据收集以实现最佳结果。

对于不同的 Agent Builder 代理程序乃至一个代理程序中的不同属性组，代理程序的配置和调整方法可能各不相同。Agent Builder 代理程序可以包含两种数据，而且它们对最常见的数据类型支持两种基本数据收集方法。

数据类型

代理程序收集两种数据：

1. 大多数 Tivoli Monitoring 属性组表示数据的快照。有些属性组会请求并返回数据。代理程序使用此类数据来表示配置、性能、状态和其他信息，对于此类情况，一次收集一组数据较为合理。此数据称为采样数据。
2. 有些 Tivoli Monitoring 数据表示事件。在这种情况下，当事件发生时，代理程序必须将数据转发到 Tivoli Monitoring。事件示例包括 SNMP 陷阱、Windows 事件日志条目以及写入日志文件的新记录。为简单起见，这些类型的数据将进行分组并称为事件数据。

采样数据

需要采样数据时，将向代理程序发送针对特定属性组的请求。可通过单击 Tivoli Enterprise Portal 中的工作空间发出请求。可发起请求的其他事件包括正在运行的情境、Warehouse 的数据收集或 SOAP 请求。代理程序接收到该请求时，它将返回该属性组的最新数据。Tivoli Enterprise Portal 请求通过特定的受管系统名称 (MSN) 定位特定属性组。情境和历史请求则更有意思，尤其是在代理程序包含子节点时。当某个情境需要子节点中某个属性组的数据时，代理程序将接收到一个含有目标子节点列表的请求。代理程序必须先以所有子节点的请求目标属性组的所有数据进行响应，然后 Tivoli Monitoring 才能处理下一个请求。

代理程序满足请求的最直接方式是，在每次接收到来自 Tivoli Monitoring 的请求时收集数据。Agent Builder 代理程序并非每次都收集数据。这是因为收集数据通常需要花费时间或耗用资源。而许多情况下会在短时间内多次请求相同数据。例如，用户可能定义了多个按相同时间间隔对属性组运行的情境，而这些情境可能针对多种不同的情况发出信号。其中每一个情境都会产生一条发送给代理程序的请求，但您可能希望所有情境都得到相同的数据。因此，正因为每个情景都得到相同的数据，所以获取了更一致的结果，从而尽可能地降低监视代理程序对系统资源的需求。

代理程序开发者可以对代理程序进行配置，通过选择下列其中一种方式运行收集来优化数据收集：

1. **随需应变收集**：代理程序在收到请求时收集数据并返回该数据。
2. **预定收集**：代理程序在后台按调度的时间间隔运行数据收集，并在收到请求时返回收集的最新数据。

在这两种方式下，代理程序都使用短期高速缓存。如果在高速缓存有效时收到另一条数据请求，代理程序将返回高速缓存中的数据，而不会为每条请求收集新数据。使用高速缓存中的数据解决多个并发情景（和其他类型）请求所导致的问题。环境变量定义数据的有效期、调度的收集时间间隔、用于收集的线程数以及代理程序的运行方式（按需还是按调度运行）。您可以使用环境变量来调整每个代理程序，以使其在自己的环境中实现最佳运行状态。

请参阅以下示例，这些示例说明了代理程序在这两种方式下的工作方式：

- 代理程序 1（按需收集）：这是收集少量数据的一个简单代理程序，通常仅由情景或在 Tivoli Enterprise Portal 中偶尔访问。数据收集速度相当快，但可能会耗尽计算资源和网络资源。此代理程序通常定义为按需运行。如果没有任何情境处于运行状态，或者没有任何用户在 Tivoli Enterprise Portal 中单击，那么代理程序不执行任何操作。当需要数据时，将收集并返回数据。数据放置在短期高速缓存中，因此几乎同时的未来请求将返回相同数据。这种收集对此代理程序而言很可能是最高效的运行方式，这是因为，它仅在某人确实需要数据时收集数据。
- 代理程序 2（调度收集）：这是一个复杂的代理程序，它包含子节点并从受监视资源的多个副本收集数据。资源的多个副本可以由一个代理程序管理。为了监视受监视资源的状态和性能，相对频繁地对数据运行情境是很常见的。此代理程序定义为运行调度收集。运行调度收集的其中一个原因是 Tivoli Monitoring 代理程序评估情境的方式。由于情境在子节点中的属性组上运行，因此代理程序将同时从所有子节点接收到一个数据请求。在返回某个情境的所有数据之前，代理程序无法响应其他请求。如果代理程序在请求到达时收集了所有数据，那么当您在 Tivoli Enterprise Portal 中单击代理程序的一个工作空间时，代理程序将冻结。为了避免冻结代理程序，Agent Builder 自动将所有子节点代理程序定义为在调度收集方式下运行。代理程序开发者将调整线程数和刷新时间间隔，以按合理的时间间隔收集此数据类型的数据。例如，刷新时间间隔可以是一分钟一次，也可以是五分钟一次。

环境变量

代理程序根据一组环境变量的值来确定使用哪种方式和如何运行调度数据收集。可以在“**环境变量**”面板上的代理程序定义中设置这些环境变量。每个环境变量都在菜单中与缺省值一起列出。也可以通过编辑 Windows 上的环境变量 (env) 文件或者 UNIX 上的初始化 (ini) 文件，针对安装的代理程序设置或修改环境变量。控制采样属性组的数据收集的环境变量有：

- CDP_DP_CACHE_TTL=<高速缓存数据的有效期 - 缺省值为 55 秒>
- CDP_DP_THREAD_POOL_SIZE=<用于并行收集的线程数 - 对于子节点代理程序，缺省值为 15>
- CDP_DP_REFRESH_INTERVAL=<相邻两次收集间隔的秒数 - 对于子节点代理程序，缺省值为 60 秒>
- CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT=<有效期到期后，等待新数据的时间长度 - 缺省值为 5 秒>

这些变量中，最重要的是 CDP_DP_CACHE_TTL、CDP_DP_REFRESH_INTERVAL 和 CDP_DP_THREAD_POOL_SIZE。

如果 CDP_DP_THREAD_POOL_SIZE 的值大于或等于 1，或者代理程序包含子节点，代理程序会在调度收集方式下运行。如果未设置 CDP_DP_THREAD_POOL_SIZE 或者设置为 0，那么代理程序会在按需收集方式下运行。

如果代理程序在调度方式下运行，那么代理程序每隔 CDP_DP_REFRESH_INTERVAL 秒就自动收集所有属性组。它使用一组后台线程执行收集。线程数通过 CDP_DP_THREAD_POOL_SIZE 设置。CDP_DP_THREAD_POOL_SIZE 的正确值根据代理程序进行的操作而有所不同。例如：

- 如果代理程序使用 SNMP 从远程系统收集数据，那么 CDP_DP_THREAD_POOL_SIZE 的值最好与受监视的远程系统的数量相仿。通过将池大小设置为近似于受监视的远程系统的数量，代理程序将并行收集数据，但是会限制远程系统上的并行负载。SNMP 守护程序往往会在繁忙时丢弃请求。丢弃请求将迫使代理程序进入重试方式，从而导致耗用更多时间和更多资源来收集数据。
- 如果代理程序包含许多需要花费较长时间来收集的属性组，那么请使用足够多的线程以使费时较长的数据收集可以并行运行。对于其余属性组，您可以再添加一些线程。如果目标资源有足够的处理能力，请按此方式使用线程。如果脚本运行很长时间或者 JDBC 查询花费很长时间，那么在这些情况下将花费较长时间来收集属性组。

用较大的线程池运行代理程序将导致代理程序使用更多内存（主要用于为每个线程分配的堆栈）。但是这不会显著增加进程的处理器使用率，也不会显著增加进程的实际工作集大小。使用适合于工作负载的正确线程

池大小可以使代理程序更为高效。您可以适当调整线程池大小，以使特定环境中的特定代理程序的行为符合需要。

在收集数据时，数据存放在内部高速缓存中。此高速缓存用于满足收集新数据之前的未来请求。高速缓存的有效期由 CDP_DP_CACHE_TTL 控制。缺省情况下，有效期设置为 55 秒。当代理程序以调度方式运行时，最好将有效期值设置为与 CDP_DP_REFRESH_INTERVAL 相同的值。如果数据收集可能会持续很长时间，请将该值设置为较大的值。使用此方式设置有效期时，在下次调度收集之前数据被视为有效。

最后一个变量是 CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT。此变量仅当 CDP_DP_CACHE_TTL 在收集新数据前到期时才发挥作用。如果高速缓存在收集新数据前到期，代理程序会立即调度另一次数据收集。然后它会等待此收集完成，但最多等待 CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT 秒。如果该新收集操作完成，就更新高速缓存并返回最新数据。如果该新收集操作未完成，就返回现有数据。在 CDP_DP_CACHE_TTL 完成时代理程序不会清除高速缓存，从而避免发生 Universal Agent 所存在的问题。Universal Agent 总是会在有效期结束时清除其数据高速缓存。如果 Universal Agent 在下次收集完成前清除其数据高速缓存，该属性组的高速缓便是空的，在收集完成前不会返回数据。在情境运行时不返回数据就会造成问题。任何在高速缓存清除后且下次收集未完成时运行的情境都将得不到数据，任何触发的情境都会被清除。结果就是仅仅因为数据收集慢了一点，就有大量事件发生触发和清除。Agent Builder 代理程序不会引发此问题。如果“旧”数据导致某情境触发，通常同样的数据会使该情境保持同样状态。在下次收集完成后，该情境会获得新数据，并根据有效数据触发或清除。

属性组

Agent Builder 代理程序包含两个属性组，您可以使用这两个属性组来检查数据收集操作以及根据您的环境调整代理程序。这两个属性组是 Performance Object Status 和 Thread Pool Status。 当使用这些属性组调整数据收集性能时，最有用的数据是：

- Performance Object Status 的 Average Collection Duration 属性。此属性向您显示每个属性组收集数据所用的时间。通常，代理程序中较小百分比的属性组表示代理程序使用的大部分处理器使用率或时间。您也许能够优化一个或多个此类属性组的收集。或者，如果某些数据不需要像其他数据一样保持最新，那么可以修改一个或多个组的收集时间间隔。关于更多信息，请参阅（第 1024 页的『示例和高级调整』）。
- Performance Object Status 的 Intervals Skipped 属性。此属性显示代理程序尝试为属性组调度新收集，而发现先前的收集仍在排队、等待运行或已在运行的次数。在行为正常的代理程序中，此属性值对于所有属性组都是零。如果此数字开始增长，表示您应该调整数据收集，例如通过增加线程、延长收集的时间间隔或优化收集。
- Thread Pool Status 的 Thread Pool Avg Active Threads 属性。您可以将此值与 Thread Pool Size 属性组比较，以了解对线程池的利用情况。如果平均活动线程数是 5，分配大小为 100 个线程的线程池可能只会浪费内存。
- Thread Pool Status 的 Thread Pool Avg Job wait 和 Thread Pool Avg Queue Length 属性。这两个属性分别表示数据收集在队列中等待线程处理的平均时间和队列中的平均收集数。由于此数据的这种收集方式，即使空闲时，系统也指示至少平均有一个作业在队列中等待。如果等待的作业数更多或平均等待时间很长，表示收集需求未得到满足。您可以考虑增加线程，延长收集的时间间隔，或者优化一个或多个属性组的收集。

事件数据

Agent Builder 代理程序可以公开多种类型的事件数据。有些行为对于所有事件数据是相同的。代理程序将每个新事件作为单独的一行数据来接收。接收到事件数据行时，会立即将其发送到 Tivoli Monitoring 进行处理，并将其添加到代理程序中的内部高速缓存。将每一行发送到 Tivoli Monitoring 时，Tivoli Monitoring 都会执行情境和历史数据收集。高速缓存用于满足 Tivoli Enterprise Portal 或 SOAP 的数据请求。代理程序可以使用高速缓存来执行为属性组定义的重复检测、过滤和摘要。每个属性组的事件高速缓存的大小都由 CDP_PURE_EVENT_CACHE_SIZE 设置。此高速缓存包含最近的 CDP_PURE_EVENT_CACHE_SIZE 事件，其中最新返回的事件排在第一位。每个事件属性组都有独立的高速缓存。当某个属性组的高速缓存写满时，就会从列表中删除最早的事件。

Agent Builder 代理程序可以公开下列各项的事件：

- Windows 事件日志条目
- SNMP 陷阱或通知

· 添加到日志文件的记录

· JMX MBean 通知

· JMX 监视器

· 来自 Java API 提供者或套接字提供者的事件。

· 连接式属性组（其中一个数据源是事件数据源）

这些事件将按最适合每个源的方式处理。对于 SNMP 陷阱和通知、JMX 通知以及来自 Java API 和套接字提供者的事件，将以异步方式接收并立即转发到 Tivoli Monitoring。不需要调整这些收集器。代理程序使用 Windows 事件日志 API 进行预订，以从操作系统接收 Windows 事件日志条目。如果代理程序使用较旧的事件日志记录 API，它会使用线程池设置轮询系统以接收新事件。对于数据源之一是事件数据源的连接式属性组，没有适用于该连接式属性组的调整。但任何适用于事件源组的调整都可使连接式属性组受益。

文件监视则更复杂。代理程序必须监视文件是否存在以及新记录添加到文件的时间。代理程序可以配置为使用文件名或静态名称的模式监视文件。由于符合模式的文件集合可能随着时间变化，因此代理程序每隔 KUMP_DP_FILE_SWITCH_CHECK_INTERVAL 秒就会检查是否有新文件或更改的文件。此全局环境变量控制代理程序实例中的所有文件监视。当代理程序确定要监视的相应文件时，必须确定文件更改时间。在 Windows 系统上，代理程序使用操作系统 API 来侦听这些更改。代理程序会在文件更新时得到通知，并立即处理这些文件。在 UNIX 系统上，代理程序每隔 KUMP_DP_EVENT 秒检查一次文件更改。此全局环境变量控制代理程序实例中的所有文件监视。当代理程序注意到文件更改时，其处理文件中的所有新数据，然后等待下一次更改。

示例和高级调整

示例

在代理程序级别定义用于更高级调整的环境变量。您只要设置一次下列变量，它们就会应用于代理程序中的所有属性组：

· CDP_DP_CACHE_TTL

· CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT

· KUMP_DP_FILE_SWITCH_CHECK_INTERVAL

· KUMP_DP_EVENT

您可以将下列变量应用于个别属性组。它们仍有适用于代理程序中所有其他属性组的全局设置：

· CDP_DP_REFRESH_INTERVAL

· CDP_PURE_EVENT_CACHE_SIZE

如果您定义了包含下列六个属性组的代理程序：

· EventDataOne

· EventDataTwo

· EventDataThree

· SampledDataOne

· SampledDataTwo

· SampledDataThree

您可设置下列缺省变量：

· CDP_DP_CACHE_TTL=55

· CDP_DP_IMPATIENT_COLLECTOR_TIMEOUT=2

· CDP_DP_REFRESH_INTERVAL=60

· CDP_PURE_EVENT_CACHE_SIZE=100

因此，对于所有包含采样数据的属性组（SampledDataOne、SampledDataTwo 和 SampledDataThree），都是每 60 秒收集一次。每个事件属性组（EventDataOne、EventDataTwo 和 EventDataThree）将在其高速缓存中存储最近的 100 个事件。

这些设置可能效果非常理想，也可能因为一些原因而必须在更详细级别控制这些设置。例如，如果 EventDataOne 通常接收到的事件是 EventDataTwo 和 EventDataThree 的 10 倍，该怎么办？ 再复杂一点，EventDataOne 和 EventDataTwo 之间其实是有关联的。在针对 EventDataTwo 收到一个事件时，对于 EventDataOne 总是存在多个事件，并且用户想要关联这些事件。因此，没有一个适合所有属性组的高速缓存大小。最好让 EventDataOne 存储较多事件，让 EventDataTwo 存储较少事件。您可以通过将 CDP_PURE_EVENT_CACHE_SIZE 设置为适合大多数事件属性组的大小（100 似乎很合适）来实现这种存储方案。然后，您可以将 CDP_EVENTDATAONE_PURE_EVENT_CACHE_SIZE 设置为 1000。这样一来，所有对应事件在 Tivoli Enterprise Portal 中都可见。

对 CDP_DP_REFRESH_INTERVAL 可以进行同样处理。设置适合代理程序中大多数属性组的缺省值。然后，针对必须单独收集的属性组设置 CDP_*attributegroup name*_REFRESH_INTERVAL。为了优化收集，将缺省 CDP_DP_REFRESH_INTERVAL 设置为与 CDP_DP_CACHE_TTL 值匹配。CDP_DP_CACHE_TTL 是全局值，因此，如果将其设置为小于刷新时间间隔的值，那么可能会发生意外的收集。

定义和测试数据源

Agent Builder 支持多种数据提供程序。您可以从每种数据提供程序创建数据源。每种数据提供程序的数据源创建与测试过程都不相同。

对于大部分数据提供程序，当您创建数据源时，会向代理程序添加数据集（属性组）。该数据集包含此数据源收集的信息。

使用进程、Windows 服务或程序返回码数据提供程序的数据源使用特殊的“可用性”数据集。代理程序中只能创建一个“可用性”数据集。它包含此代理程序中，使用进程、Windows 服务或程序返回码数据提供程序的所有数据源收集的信息。

代理程序和子节点中的所有 Windows 日志数据源都将事件信息放置在一个“事件日志”数据集中。

为 Cloud APM 设置数据源

在 Cloud APM 中，您可以使用来自于“详细信息”仪表板的所有数据集中的数据，并使用阈值管理器来设置阈值。如果要在代理程序或子节点的摘要仪表板中使用来自数据集的信息（包括状态指示符）以及资源信息（服务名称、地址和端口），那么数据集必须只能产生一行。

对于大部分数据提供程序，可以在数据集配置中选择**生成单一数据行**。如果收集的信息将包含多行，您可单击**高级**，以设置一个可确保生成正确行的过滤器（有关指示信息，请参阅第 1011 页的『过滤属性组』）。您可以测试数据源以确保收集的信息产生所需的行。

对于某些数据提供程序，数据集必须产生多行。此外，进程、Windows 服务和命令返回码数据源会将数据放入产生多行的单个“可用性”数据集中。在此类情况下，必须创建产生单行的已过滤数据集。有关创建过滤型数据集（属性组）的指示信息，请参阅第 1125 页的『创建过滤属性组』。

其他的一些数据提供程序产生事件数据；针对每个新事件包含一行。请勿将这些数据提供程序用于 Cloud APM 中的摘要或资源信息。

以下数据提供程序必须产生包含多行的数据集：

- 进程（使用“可用性”数据集）
- Windows 服务（使用“可用性”数据集）
- 程序返回码（使用“可用性”数据集）
- SNMP 和 JMX（针对某些数据类型）
- 套接字和 Java API（取决于应用程序）

以下数据提供程序产生事件数据：

- SNMP 事件
- 日志文件
- AIX 二进制日志
- Windows 事件日志

· 套接字和 Java API（取决于应用程序）

数据集的其中一个属性必须提供状态值。Cloud APM 将此值用作整体状态指示符。如果该行不包含可用作状态指示符的属性，那么可以创建派生属性以计算状态。您必须配置状态严重性值；有关指示信息，请参阅第 1010 页的『指定用作状态指示符的特性的严重性』。

监视进程

您可以定义用于监视服务器上运行的一个或多个进程的数据源。这些进程与代理程序运行在同一主机上。对于每个进程，数据源都会向“可用性”数据集添加一行。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**进程**。
2. 在**数据源**区域中，单击**进程**。
3. 单击**下一步**。
4. 在“**进程监视器**”页面上的“**进程信息**”区域中，提供显示名称和进程名称。您可以手动输入进程名称，也可以通过单击**浏览**来获取进程名称。单击**浏览**将显示本地系统或远程系统上当前正在运行的进程列表。

   您可以通过选择**使用自变量匹配**和**匹配完整命令行**选项来进一步区分进程。例如，如果系统上正在运行同一进程的多个实例，那么可以使用这些选项来区分这些实例。

表 *265.* “**进程监视器**”页面上的字段. 下表列出了“**进程监视**”页面中的字段及其描述

字段名称	描述	可接受的值
Display name	这是由进程所实现的应用程序组件的描述性名称，显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中	描述性字符串
Process name	正在受监视的进程的名称	有效的可执行文件名
Use argument match	如果您想要对进程自变量进行匹配，请选择此选项。	On 或 Off
自变量	要对其进行匹配的自变量字符串。自变量匹配查找所提供的作为自变量子串的字符串。如果提供自变量的任意部分作为输入字符串，那么匹配成功。	字符串
Match full command line	指定可执行文件的完整名称（可能包含路径）	On 或 Off
Command line	将所提供的字符串与用来启动进程的标准命令名进行匹配。不包含命令自变量。“标准”表示必须包含该命令的路径。	字符串
Operating systems	选择运行该进程的操作系统	任意选择

5. 如果您单击**浏览**，则会打开“**进程浏览器**”窗口。该窗口初始包含有关 Agent Builder 系统上的每个进程的详细信息。该信息包括标识、进程名称和进程的完整命令行。选择一个或多个进程，或者通过以下一个或多个操作来使用“**进程浏览器**”窗口中的列表：
   a) 要对进程列表进行排序，请单击列标题。
   b) 要刷新窗口中的信息，单击**刷新**（闪电）图标。
   c) 要搜索特定进程，单击**搜索**（望远镜）图标。
      您可输入搜索短语并选择选项部分，以便按进程标识、名称和命令行执行搜索。

d) 要查看其他系统上的进程，请从**连接名称**列表中选择先前定义的系统。或者，单击**添加**以输入新系统的系统信息。

有关更多信息，请参阅第 1028 页的『定义用于进程浏览的连接』。您可以同时从多个系统装入进程，并且在为一个或多个连接装入进程时切换连接。

注: 在浏览远程系统时，仅当通过 Tivoli Enterprise Portal Server 浏览时，命令行详细信息才可用。

在以下示例中，选择 `svchost.exe` 之后，该进程将显示在**进程监视器**页面上的**进程名称**字段中（第 1027 页的图 31）。

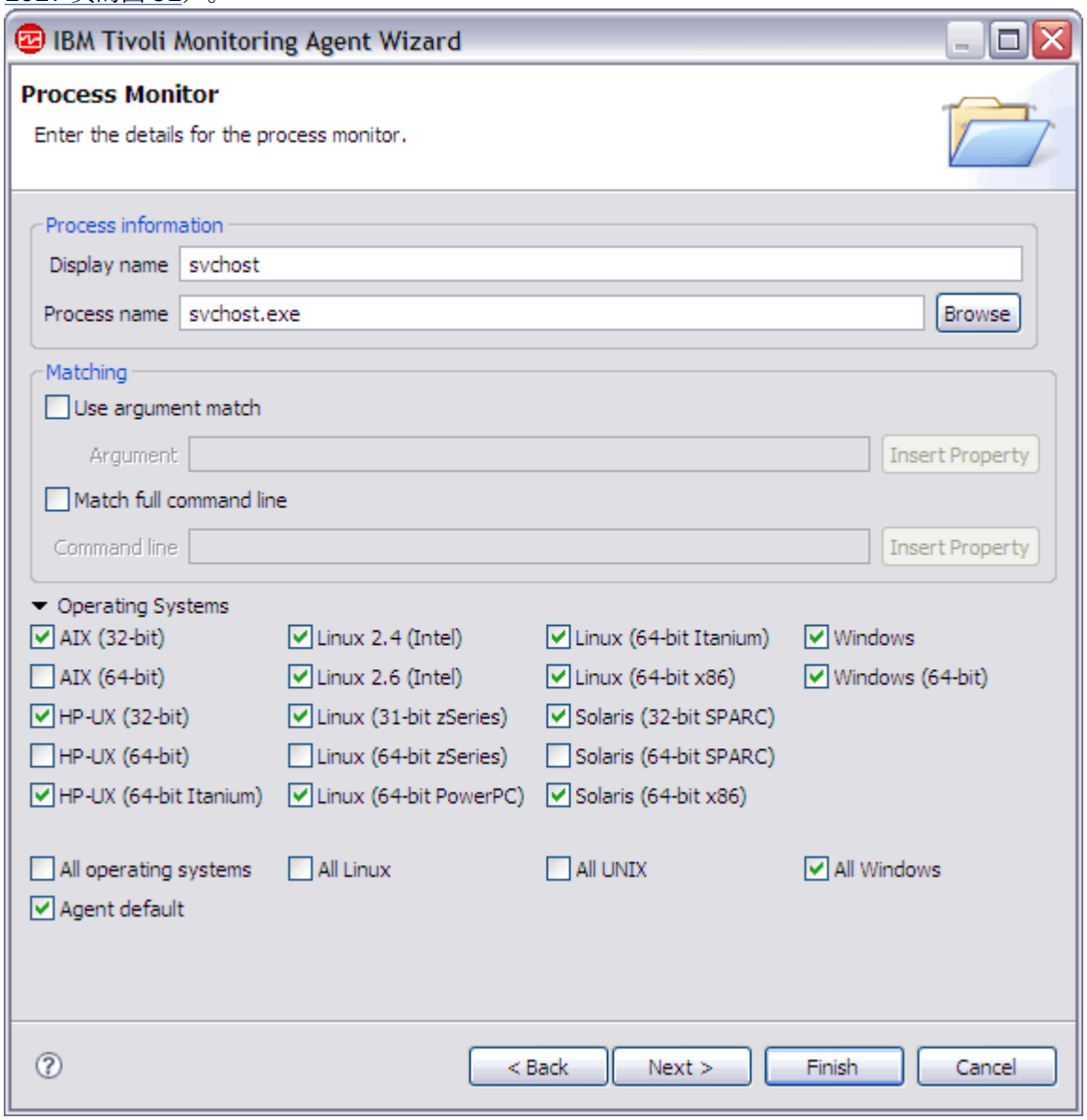


图 *31.* “进程监视器”页面示例

6. 使用第 1026 页的表 265 中的信息填写“**进程监视器**”页面。

注: 如果在此监视器中描述的进程仅适用于您的应用程序运行所在的部分操作系统，那么可能需要使用相同显示名称来创建一个或多个进程监视器以涵盖其他操作系统。一次添加一个进程监视器。请确保各个监视器的显示名称相同，但是可以在选择的操作系统上找到该进程名称。

7. 执行下列其中一个步骤:
   - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
   - 单击**完成**，以保存该数据源并打开 Agent Editor。

下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据，必须创建基于“可用性”数据集过滤的数据集（属性组），并将其配置为提供单行。使用“名称”字段来选择用于您的进程的行。

您可以使用“状态”字段来显示状态；DOWN 表示进程未在运行，UP 表示正在运行。在新的过滤属性组中，选择“状态”字段并为其指定严重性值。

如果有多个进程副本正在运行，“可用性”数据集中将显示具有该进程名称的若干行，且所有这些行都包含 UP 状态。您的过滤数据集必须配置为返回一行，这样可以返回其中任何行，但“状态”值在任何情况下均有效。

有关指示信息，请参阅：

- 第 1125 页的『创建过滤属性组』
- 第 1010 页的『指定用作状态指示符的特性的严重性』
- 第 1154 页的『为 Cloud APM 准备代理程序』

定义用于进程浏览的连接

在定义进程数据源时，可以查看和选择来自其他系统的进程。但是，代理程序运行时，将监视与代理程序在同一系统上运行的进程。

关于此任务

您必须具有其他系统的凭证，否则这些系统必须由 Tivoli Monitoring 操作系统代理程序监视。

过程

1. 要定义连接，请单击“**进程浏览器**”窗口中的**添加**。

   您可以选择连接类型（Secure Shell (SSH)、Windows 或 Tivoli Enterprise Portal Server Managed System），或者选择现有连接以用作模板。

   要添加 Managed System 连接，需要 Tivoli Enterprise Server 主机名、Tivoli Monitoring 用户名和密码。您还需要远程连接的受管系统名称。选定受管系统后，该表将列出远程系统上的进程。

   注： 操作系统代理程序必须在您正尝试浏览的系统上运行。代理程序还必须连接到正在运行的 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server。

   要添加 Secure Shell (SSH) 或 Windows 连接，您需要主机名、用户名和密码。
2. 添加连接时，可以从“**进程浏览器**”窗口的**连接名称**列表中选择连接。

   如果不保存建立连接所需的所有字段（例如，密码），那么将打开该连接的**连接属性**窗口。输入缺失的信息。对于 Tivoli Enterprise Portal Server Managed System 连接，必须先连接到 Tivoli Enterprise Portal Server，然后才能进入受管系统。
3. 输入用户名和密码，单击**刷新**（闪电）图标进行连接，然后再选择受管系统。

下一步做什么

要删除连接，请选择连接，然后单击**编辑**以打开“**连接属性**”窗口。选中**移除此连接**复选框，并单击**确定**。

监视 Windows 服务

您可以定义用于监视 Windows 系统上运行的一个或多个服务的数据源。这些服务与代理程序运行在同一主机上。对于每个服务，数据源都会向“可用性”数据集添加一行。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**进程**。
2. 在**数据源**区域中，单击 **Windows 服务**。
3. 单击**下一步**。

4. 在“**服务监视器**”页面的**显示名**字段中，输入描述。在**服务名称**字段中，提供服务应用程序的名称。您可以手动输入，也可以单击**浏览**以查看本地系统或远程系统上当前正在运行的服务列表。

   如果您单击**浏览**，则会打开“**服务浏览器**”窗口。此窗口初始状态下包含有关 Agent Builder 系统上每个服务的详细信息。此信息中包含服务的服务名称、显示名称、状态和描述。

   注: 如果 Agent Builder 不是在 Windows 系统上运行，那么不会显示本地服务。必须定义或选择远程 Windows 系统，请参阅（第 1029 页的『定义用于服务浏览的连接』）。

   注: 当您通过 Tivoli Enterprise Portal Server 或从 UNIX 或 Linux 系统进行浏览时，服务描述不可用。
5. 选择一个或多个服务，或者执行以下一个或多个步骤来使用“**服务浏览器**”窗口中的列表：
   - 要对服务列表进行排序，请单击列标题。
   - 要刷新窗口中的信息，单击**刷新**（闪电）图标。
   - 要搜索服务，请单击**搜索**（望远镜）图标以打开“**服务搜索**”窗口。您可按服务名称、显示名和描述进行搜索。
   - 要查看另一系统上的服务，请从**连接名称**列表中选择先前定义的系统，或者单击**添加**以输入系统信息。有关更多信息，请参阅第 1029 页的『定义用于服务浏览的连接』。您可以同时从多个系统装入服务并且在为一个或多个连接装入服务时切换连接。
6. 在选择或输入服务名称之后，完成下列其中一个步骤：
   - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
   - 单击**完成**，以保存该数据源并打开 Agent Editor。

下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据，必须创建基于“可用性”数据集过滤的数据集（属性组），并将其配置为提供单行。使用“名称”字段来选择用于您的进程的行。

在新的过滤属性组中，选择“功能测试状态”字段并为其指定严重性值。

有关指示信息，请参阅：

- 第 1125 页的『创建过滤属性组』
- 第 1010 页的『指定用作状态指示符的特性的严重性』
- 第 1154 页的『为 Cloud APM 准备代理程序』

定义用于服务浏览的连接

除了选择来自正在运行 Agent Builder 的系统的服务以外，您还可以选择来自其他 Windows 系统的服务。

关于此任务

要从其他 Windows 系统选择服务，请定义与远程系统的连接。您必须具有系统凭证，否则这些系统必须由 Tivoli Monitoring 操作系统代理程序监视。

过程

1. 要定义连接，请单击“**服务浏览器**”窗口中的**添加**。

   “**选择连接类型**”窗口随即打开。要添加 Managed System 连接，需要 Tivoli Enterprise Server 主机名、Tivoli Monitoring 用户名和密码以及受管系统名称。选定受管系统后，该表将列出远程系统上的服务。

   注: 操作系统代理程序必须在您尝试浏览的系统上运行，还必须连接到正在运行的 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server。

   添加 Windows 连接需要主机名、用户名和密码。
2. 选择连接类型（Windows 或 Tivoli Enterprise Portal Server Managed System），或者选择现有连接以用作模板。

   “**连接属性**”窗口随即打开。

3. 填写“连接属性”。
4. 单击**完成**。
5. 添加连接时，可以从“**服务浏览器**”窗口的**连接名称**列表中选择连接。
   如果没有保存建立连接所需的字段（例如，密码），将打开**连接属性**窗口，您可以输入缺少的信息。
   a) 对于 Tivoli Enterprise Portal Server Managed System 连接，您必须先连接到 Tivoli Enterprise Portal Server，然后才能进入受管系统。输入用户名和密码，单击**刷新**（闪电）图标进行连接，然后再选择受管系统。
6. 要删除连接，请执行以下步骤：
   a) 在“**服务浏览器**”窗口中选择连接。
   b) 单击**编辑**以打开“**连接属性**”窗口。
   c) 选中**移除此连接**复选框。
   d) 单击**确定**。

监视来自 Windows Management Instrumentation (WMI) 的数据

您可以定义数据源以从代理程序运行所在的系统或远程系统上的 Windows Management Instrumentation (WMI) 收集数据。数据源监视单个 WMI 类，并将此类中的所有值放在它所产生的数据集中。如果类提供多个实例，那么数据集包含多行；您可以按实例名称进行过滤以确保数据集仅包含一行。

开始之前

如果代理程序使用 Windows Management Instrumentation (WMI) 从远程系统收集数据，那么它需要具有访问该远程系统上的 WMI 数据的许可权。当您提供了具有远程系统上 WMI 数据的访问许可权的帐户凭证时，代理程序可以访问该系统上的 WMI 数据。管理员帐户具有必需的许可权。在随后过程中，您可以提供管理员凭证或具有必需许可权的其他用户的凭证。有关创建具有 WMI 数据浏览许可权的用户帐户的更多信息，请参阅第 1146 页的『创建具有 Windows Management Instrumentation (WMI) 权限的用户』。

要通过 Windows API 收集度量值，代理程序必须在 Windows 操作系统上托管。远程系统上必须启用了远程注册表管理。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **WMI**。
3. 单击**下一步**。
4. 在“**Windows Management Instrumentation (WMI) 信息**”页面上，完成下列其中一个步骤：
   - 在这些字段中输入 WMI 名称空间的名称和 WMI 类名的名称。然后，转至步骤 第 1031 页的『9』
   - 单击**浏览**来查看系统上的所有 WMI 类。

     要浏览远程系统，请从列表中选择系统（如果已定义）。或者，单击**添加**以添加 Windows 系统主机名。提供具有远程系统上 WMI 数据的访问许可权的用户帐户凭证，或提供远程系统的管理员凭证。此页面将更新以显示远程系统的信息。仅当在 Windows 系统上运行 Agent Builder 时才能进行浏览，并且只能浏览 Windows 系统。

5. 单击类旁边的加号 (+) 以展开类并显示特性。
6. 从列表中，选择类以及您要为其指定的关联特性，然后单击**确定**。

   注: 您可以单击**搜索**（望远镜）图标，以查找列表中的所选项。请在**搜索短语**字段中输入短语；通过单击**按名称搜索**、**按类描述搜索**或**按类属性搜索**字段指定首选项；然后单击**确定**。如果找到所搜索的项，请将其选中，然后单击**确定**。

   此时会再次打开向导的“**WMI 信息**”页面，其中显示所选的 WMI 类信息。
7. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1031 页的『测试 WMI 属性组』

8. 可选：通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关过滤来自属性组的数据的更多信息，请参阅第 1011 页的『过滤属性组』
9. 单击**下一步**。

   注: 如果手动输入了 WMI 名称空间和 WMI 类名，则您将转到“**特性信息**”页面，您可以在这里填写特性信息。在“**特性信息**”页面上，如果要添加更多特性，您可选择**添加其他特性**。单击**完成**以完成操作。
10. 在“**选择键特性**”页面上，选择键特性或指示此数据源仅生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
11. 执行下列其中一个步骤：
    - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
    - 单击**完成**，以保存该数据源并打开 Agent Editor。
12. 您可以添加特性并提供其相关信息。有关更多信息，请参阅第 1004 页的『创建属性』。

    WMI 数据源的“**特性信息**”页面除了包含适用于所有数据源的字段（第 1007 页的表 261）以外，还包含以下字段：

    度量名称
    要收集的类的属性名
13. 如果要为数据源设置全局选项，请单击**全局选项**。

    如果要包括此选项，请选中**包括远程 Windows 配置属性**复选框，并单击**确定**。

    有关 Windows 数据源的 Windows 远程连接配置的信息，请参阅（第 1145 页的『配置 Windows 远程连接』）。

测试 WMI 属性组

如果是在 Windows 系统上运行 Agent Builder，那么可以在 Agent Builder 中测试 WMI 属性组。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**WMI 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**WMI 测试**”窗口。
2. 可选：在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。
3. 单击**启动代理程序**。

   此时将显示一个窗口，表明代理程序正在启动。
4. 要模拟对代理程序数据的监视环境请求，请单击**收集数据**。

   代理程序会查询 WMI 以获取数据。“**WMI 测试**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
5. 可选：如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据。
6. 通过单击**停止代理程序**停止代理程序。
7. 单击**确定**或**取消**以退出“**WMI 测试**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』

使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视 Windows 性能监视器 (Perfmon)

您可以定义数据源以从 Windows 性能监视器 (Perfmon) 收集数据。数据源监视 Perfmon 对象。该对象中的计数器放置在所产生数据集中的属性中。如果类提供多个实例，那么数据集包含多行；您可以按实例名称进行过滤以确保数据集仅包含一行。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **Perfmon**。
3. 单击**下一步**。
4. 在“**Perfmon 信息**”页面上，请完成下列其中一个步骤：
   - 在**对象名称**字段中输入对象名称，然后单击**下一步**以定义属性组中的第一个属性。

     注: 如果您为 Windows 性能监视器对象输入名称，那么此名称必须是英语名称。
   - 单击**浏览**以查看 Perfmon 对象列表。

     当性能监视器 (Perfmon) 的“对象浏览器”窗口最初打开时，此窗口中填充了来自本地系统的信息。要浏览远程系统，请从列表中选择系统（如果已定义），或者单击**添加**以添加 Windows 系统主机名。提供管理员标识和密码。此窗口将更新以包含远程系统的信息。仅当在 Windows 系统上运行 Agent Builder 时才能进行浏览，并且只能浏览 Windows 系统。例如，不能添加 Linux 或 Solaris 系统的主机名来执行远程浏览。

     – 您单击对象名称时，该对象中的可用计数器将显示在窗口中。

       - 要对 Windows 性能监视器 对象或计数器进行排序，请单击列标题。
       - 要刷新窗口中的信息，请单击**刷新**。
       - 要搜索特定的对象或计数器，请单击**搜索**（望远镜）图标，以打开“**性能监视器搜索**”窗口。您可以搜索对象名称和/或计数器名称。搜索操作执行子串匹配，并且不区分大小写。
       - 选择对象，然后单击**确定**。
       - “**Perfmon 信息**”页面将打开，其中的**对象名称**字段会显示选定对象的名称。
   - 如果要为数据源设置全局选项，请单击**全局选项**。

     如果要包括此选项，请选中**包括远程 Windows 配置属性**复选框，并单击**确定**。

     有关 Windows 数据源的 Windows 远程连接配置的信息，请参阅（第 1145 页的『配置 Windows 远程连接』）。
5. 如果所选的 Windows 性能监视器对象返回多个实例，并且您希望根据实例名称来过滤结果：
   a) 在“**Perfmon 信息**”页面上，选中**按 Perfmon 实例名称过滤**复选框。
   b) 在 **实例名称**字段中，输入要过滤的实例名称，或者单击**浏览**以列出可用的实例。
   c) 要浏览远程系统，从列表中选择一个系统，或者单击**添加**以添加 Windows 系统主机名。选择主机之后，请提供管理员标识和密码。该表将更新以包含远程系统上的实例列表。

   注: 您还可以按属性组进行过滤，请参阅步骤 第 1032 页的『9』
6. 如果所选的 Windows 性能监视器对象要返回多个实例，并且您希望返回实例名称，请选择“**Perfmon 信息**”页面上的**返回实例名称**。

   选中该选项将会向数据源中添加一个属性，该属性不会显示在属性列表中。该属性包含实例名称。

   注: 如果浏览过选定对象，并且该对象定义为具有多个实例，那么将自动选中该复选框。
7. 如果未选中用于返回实例名称的选项，那么将打开“选择键属性”页面。在“选择键属性”页面上，选择键属性或指示此数据源仅生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
8. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1033 页的『测试 Perfmon 属性组』
9. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。

有关过滤来自属性组的数据的更多信息，请参阅步骤第 1011 页的『过滤属性组』

注: 您还可以按实例名称进行过滤，请参阅 第 1032 页的『5』

10. 执行下列其中一个步骤：
    - 如果您正在使用“新建代理程序”向导，请单击**下一步**。
    - 单击**完成**，以保存该数据源并打开 Agent Editor。

    Agent Editor 的“**数据源定义**”页面会显示一个列表，其中包含对象以及对象的相关信息。
11. 您可以添加属性并提供其相关信息。有关更多信息，请参阅第 1004 页的『创建属性』。

    数据源的“**Perfmon 属性信息**”页面除了包含适用于所有数据源的字段以外，还包含以下字段：

    度量名称
    特定对象的计数器的名称。

下一步做什么

有关 Perfmon 数据源的 Windows 远程连接配置的信息，请参阅第 1145 页的『配置 Windows 远程连接』。

测试 Perfmon 属性组

如果您在 Windows 系统上运行 Agent Builder，可以测试所创建的 Perfmon 属性组。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**Perfmon 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**Perfmon 测试**”窗口。
2. 可选： 在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。
3. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
4. 要模拟来自监视环境的代理程序数据请求，请单击**收集数据**。

   代理程序会查询性能监视器以获取数据。“**Perfmon 测试**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。

   注: 您可能直到再次单击**收集数据**之后，才会看到所有属性的有用数据。原因是某些性能监视器属性返回的是变化量值，因此需要上一个值才能计算变化量值。
5. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据。
6. 通过单击**停止代理程序**停止代理程序。
7. 单击**确定**或**取消**以退出“**Perfmon 测试**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视来自简单网络管理协议 (SNMP) 服务器的数据

您可以定义数据源以监视 SNMP 服务器。数据源监视来自单个 SNMP 对象标识 (OID) 和单个主机的所有数据。如果选择注册其他对象的 OID 注册树下的某个元素，将针对每组不同的标量或表值创建数据集。如果对象返回标量数据，那么数据集只包含一行。如果对象返回表格数据，那么数据集将包含多行。

关于此任务

代理程序支持简单网络管理协议 V1、V2C（请注意，此版本是 V2C 而不只是 V2）和 V3。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **SNMP**。
3. 单击**下一步**。
4. 在“简单网络管理协议 (SNMP) 信息”页面上，输入显示名称，或者单击**浏览**以查看系统上的所有对象。

   定义数据源之后，可以添加属性。这些属性的 OID 可能很长且难以正确输入。使用 Browse 选项可方便地输入正确的 OID。

   注: 浏览器不浏览活动系统，而是读取定义，即管理信息库 (MIB)。

   注: 单击**刷新**图标可以清除已解析 MIB 文件的内存中版本，并重新解析工作空间高速缓存中的文件。高速缓存位于以下位置：*workspace_directory*\.metadata\.plugins\com.ibm.tivoli.monitoring.agentkit\mibs

   其中:

   workspace_directory
   标识在初始运行 Agent Builder 时指定的工作空间目录，请参阅（第 986 页的『启动 Agent Builder』）。

   a) 如果未装入定义所需对象的 MIB，单击**管理定制 MIB** 以打开“管理定制 MIB”对话框。

   b) 单击**添加**以浏览到要添加的 MIB 文件。要从高速缓存删除 MIB，请将其选中，并单击**移除**。

   c) 单击**确定**以更新高速缓存。

   如果解析 MIB 时发生任何错误，那么“管理定制 MIB”对话框保持打开状态。该对话框允许您添加或移除 MIB 以消除错误。

   单击**取消**会将 MIB 高速缓存恢复到打开此对话框时的状态。

   Agent Builder 包含一组 MIB:

   - hostmib.mib
   - rfc1213.mib
   - rfc1243.mib
   - rfc1253.mib
   - rfc1271.mib
   - rfc1286.mib
   - rfc1289.mib
   - rfc1315.mib
   - rfc1316.mib
   - rfc1381.mib
   - rfc1382.mib
   - rfc1443.mib
   - rfc1461.mib
   - rfc1471.mib
   - rfc1493.mib
   - rfc1512.mib
   - rfc1513.mib
   - rfc1516.mib
   - rfc1525.mib
   - rfc1573a.mib

- rfc1595.mib
- rfc1650.mib
- rfc1657.mib
- rfc1659.mib
- rfc1666.mib
- rfc1695.mib
- rfc1747.mib
- rfc1748.mib
- rfc1757.mib
- rfc1903.mib
- rfc1907.mib
- rfc2011.mib
- rfc2021.mib
- rfc2024.mib
- rfc2051.mib
- rfc2127.mib
- rfc2128.mib
- rfc2155.mib
- rfc2206.mib
- rfc2213.mib
- rfc2232.mib
- rfc2233.mib
- rfc2238.mib
- rfc2239.mib
- rfc2320.mib
- rfc3411.mib

所有这些 MIB 都是由 IETF 定义的标准 MIB。包括这些 MIB 的原因是，它们表示对监视可能有用的公共定义。此外，还有许多 MIB 也是必需的，这样定制 MIB 才能解析所导入的符号。

d) 从列表中选择对象。

单击对象旁的加号 (+) 以展开并显示层次。

e) 从列表中，选择要指定的对象，然后单击**确定**。

然后，新数据源将列示在“**数据源定义**”页面上。

注: 如果选择了定义其他对象（嵌套在第一个对象下的各个对象）的对象，那么所有这些对象都将变成数据源。如果选择高级对象，那么会添加许多数据源。

5. 在“**简单网络管理协议信息**”页面上，选择操作系统。
6. 可选： 您可以通过单击“**简单网络管理协议信息**”页面上的**测试**来测试一个或多个数据源。

有关测试的更多信息，请参阅第 1037 页的『测试 SNMP 属性组』

7. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关过滤来自属性组的数据的更多信息，请参阅第 1011 页的『过滤属性组』
8. 单击**下一步**。
9. 在“**属性信息**”页面上，指定属性信息。
10. 执行下列其中一个步骤:

- 如果您正在使用“新建代理程序”向导，请单击**下一步**。
- 单击**完成**，以保存该数据源并打开 Agent Editor。

11. 有关添加属性和提供属性信息的更多信息，请参阅第 1004 页的『创建属性』。

    SNMP 数据源的“**属性信息**”页面除了包含适用于所有数据源的字段以外，还包含以下字段：

    度量名称
    任意字符串

    Object identifier
    注册给对象的完整 OID，不包括索引值

下一步做什么

您可以使用代理程序运行时配置来设置受监视主机。

为使 Agent Builder 能够生成 64 位数据类型，并能够为 32 位无符号的 MIB 属性处理最大值，请参阅第 1036 页的『SNMP MIB 解析选项』。

SNMP MIB 错误
处理 SNMP MIB 中的错误。

在添加 SNMP MIB 时，经常会发现错误。单击“**Agent Builder 错误**”窗口中的**详细信息 >>** 可查看 MIB 错误详情。

一个最常见的错误是缺少在其他 MIB 中所作的定义。可以同时导入多个 MIB 以解决此问题，也可以逐次添加 MIB，直到所有缺少定义问题全部解决。Agent Builder 可以使用已解决相关问题的任何定义。因此，如果某些错误仅影响 MIB 中不打算使用的那一部分，您可以选择忽略此类错误。MIB 的顺序无关紧要，因为先将它们全部装入，然后才解析引用。

SNMP MIB 解析选项
设置 SNMP MIB 的解析首选项

过程

1. 在 Agent Builder 中，选择**窗口 > 首选项**以打开“**首选项**”窗口。
2. 在导航窗格中，展开 **IBM Agent Builder**。
3. 单击 **MIB 解析**以打开“**MIB 解析**”窗口。

   Agent Builder 使用的 MIB 解析器使用 ASN.1 定义的语法来解析 MIB。某些 MIB 未正确遵循该语法。解析器允许放宽特定规则以适应最常见的错误。通过放宽这些规则，您可以解析不合规的 MIB。

   允许类型以小写字母开头
   允许用户在 MIB 中编写的类型，例如值

   允许以数字命名的编号
   允许以大写字母开头的编号

   允许在值名称中使用下划线
   允许使用下划线字符

   允许值以大写字母开头
   允许以大写字母开头的值

   忽略重复的 MIB
   关闭重复 MIB 模块警告

4. 可选： 如果选中**针对 32 位无符号 MIB 属性创建 64 位特性**复选框，Agent Builder 就能够生成 64 位数据类型来处理 32 位无符号 MIB 属性的最大值。选择此选项不会更改任何现有代理程序字段定义。您必须浏览至 MIB 文件才能为这些属性创建新的数据源。
5. 完成首选项编辑时，请单击**确定**。

测试 SNMP 属性组

您可以对在 Agent Builder 中创建的 SNMP 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**简单网络管理协议信息**”页面上的**测试**。

     注：

     如果选择的 SNMP 对象包含多个属性组，系统将提示您选择要测试的属性组。
   - 创建代理程序之后，在 **Agent Editor** 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述其中一步中单击**测试**之后，将打开“SNMP 测试设置”窗口。
2. 从**连接名称**中选择现有连接，或者单击**添加**，然后系统将提示您选择连接类型。另外，也可使用“**创建连接向导**”来选择现有连接以用作模板。
3. 在选择连接类型或现有连接之后，单击**下一步**以填写 SNMP 连接属性。完成后，单击**完成**以返回到“SNMP 测试设置”窗口。
4. 可选： 在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。
5. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
6. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。代理程序会查询已配置的 SNMP 连接以获取数据。
7. “**测试设置**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
8. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据。
9. 通过单击**停止代理程序**停止代理程序。
10. 单击**确定**或**取消**以退出“**测试设置**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视来自简单网络管理协议事件发送方的事件

您可以定义数据源以从 SNMP 陷阱和通知事件收集数据。您必须在代理程序运行时配置中设置端口，并配置服务器以通过此端口向代理程序主机发送事件。所有受监视的事件都作为行放置在数据集中。

关于此任务

代理程序支持简单网络管理协议 (SNMP) V1、V2C（请注意，此版本名称是 V2C 而不只是 V2）和 V3。代理程序可以接收和处理 SNMP 陷阱和通知。此提供者接收到的数据将作为事件传递到监视环境。

有关 SNMP 事件属性组的更多信息，请参阅（第 1220 页的『“SNMP 事件”属性组』）。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **SNMP 事件**。
3. 单击**下一步**。
4. 在“**简单网络管理协议事件信息**”窗口中，执行下列其中一个步骤：
   - 单击**所有事件**以创建属性组，用来为任何收到的 SNMP 事件发送事件。
   - 单击**一般事件**以创建属性组，用来为与任何所选通用事件类型匹配的所接收到的任何通用 SNMP 事件发送事件。

· 单击**定制事件**以创建一个或多个属性组，用来为特定于企业的 SNMP 事件发送事件。单击**浏览**以选择要监视的事件。

在“**简单网络管理协议 (SNMP) 管理信息库 (MIB) 浏览器**”窗口中，按照事件定义所在的 MIB 模块组织选择窗格中的事件。展开 SNMP 对象以显示该 MIB 模块中的事件。在列表中，单击要指定的对象，然后单击**确定**。

如果某个陷阱配置文件包含陷阱的静态数据，请选中**包括用于显示陷阱配置文件中所定义信息的属性**复选框。有关 SNMP 陷阱配置文件的更多信息，请参阅（第 1269 页的『SNMP 陷阱配置』）。

如果要将属性以及收到的所有变量绑定 (VarBind) 数据包括在陷阱协议数据单元 (PDU) 中，请选中**包括变量绑定 (VarBind) 数据属性**复选框。有关此属性的更多信息，请参阅属性定义（第 1220 页的『“SNMP 事件”属性组』）。

注:

a. 浏览器不浏览活动系统；而是读取定义和管理信息库 (MIB)。第 1033 页的『监视来自简单网络管理协议 (SNMP) 服务器的数据』中定义了 Agent Builder 所包含的 MIB 列表。任一 SNMP 数据提供程序装入的 MIB 在两者中都可使用。

b. 如果选择 MIB 模块或个别事件，那么该模块中的所有事件都将转换为单独的数据源。为事件中定义的每个变量添加一个属性。如果您希望所选模块或陷阱的所有事件都到达单个事件源，请选中**收集单一属性组中的事件**复选框。如果选择单个陷阱，并且选择了**收集单一属性组中的事件**标记，那么会为每个事件中定义的每个变量（忽略重复变量）添加一个属性。如果您选择模块，那么不会添加变量属性。

c. 如果要输入自己的过滤器，请使用以下语法：

OID（对象标识）元素的值用于确定为此属性组处理哪些陷阱。

– **陷阱匹配**：global_snmp_event_settings_for_group 元素的 OID 属性可以是逗号定界的标记列表。单个标记语法如下：

```
[enterpriseOID][-specificType]
```

– **示例：**“1.2.3.5.1.4,1.2.3.4.5.6.7.8.9-0”。第一个标记与企业 OID 为 1.2.3.5.1.4 的任何陷阱匹配。第二个标记与企业 OID 为 1.2.3.4.5.6.7.8.9 且特定 OID 为 0 的任何陷阱匹配。由于这些标记共同列示在同一个属性组中，因此与任一标记匹配的已接收事件都会由该属性组处理。

d. 接收到的每个事件仅由与接收到的事件匹配的第一个属性组处理。先处理子节点属性组，然后再处理基本属性组。代理程序开发者必须确保以正确的方式定义组以便在预期属性组中接收事件。

5. 在“**SNMP 事件信息**”窗口中，选中**子节点主机匹配**复选框以将事件与子节点匹配。如果 SNMP 事件属性组是子节点的一部分，可以选中**子节点主机匹配**复选框来控制事件是否必须来自受监视的 SNMP 代理程序。

例如：您已让一个代理程序监视路由器，其中每个子节点实例代表一个特定路由器。您开发了一个使用 SNMP 数据收集器从路由器收集数据的代理程序。您还定义了一个接收该路由器所发送的 SNMP 事件的属性组。每个路由器实例都包含为事件过滤器定义的相同数据。因此，您需要其他方法来确保来自路由器的事件显示在该路由器的属性组中。

选中 Subnode Host Matching 时，由路由器发送的事件将与为 SNMP 数据收集器定义的主机进行比较。如果 SNMP 数据收集器正在使用的主机与发送所接收事件的主机相同，那么该子节点实例会处理 SNMP 事件。否则，会将事件传递到下一个子节点实例。地址匹配仅适用于子节点。基本代理程序中的 SNMP 事件属性组不执行任何地址匹配。要使用地址匹配，子节点定义必须包含至少一个 SNMP 属性组。SNMP 用于该子节点实例的 SNMP 主机是用于匹配的主机。

如果取消选中**子节点主机匹配**复选框，您的子节点实例不会执行这一额外的比较。必须允许用户为此情况中的各个子节点配置其他 OID 过滤器。否则，您无需在子节点定义中包含 SNMP 事件属性组。

6. 在“**SNMP 事件信息**”窗口中，选择操作系统。
7. 可选： 您可以单击“**SNMP 事件信息**”窗口中的**测试**以启动并测试代理程序。

有关更多信息，请参阅第 1041 页的『测试 SNMP 事件属性组』

8. 可选：

在“**SNMP 事件信息**”窗口中，单击**高级**，以选择**事件过滤和摘要选项**。有关更多信息，请参阅第 1180 页的『事件过滤和摘要』。

a) 完成选择**事件过滤和摘要选项**之后，返回到“**SNMP 事件信息**”窗口。如果先前在“**SNMP 事件信息**”窗口中选择了**定制事件**，请单击**下一步**以选择键属性，否则跳过下一步。

b) 在“选择键属性”页面上，为属性组单击一个或多个键属性，或者单击**生成单一数据行**。

9. 单击**下一步**，如果使用“新建代理程序”向导来保存代理程序，请单击**完成**，并打开 Agent Editor。
10.

下一步做什么

有关添加更多属性的信息，请参阅（第 1004 页的『创建属性』）。

SNMP 事件配置属性

将 SNMP 事件属性组添加到代理程序时，自动创建某些配置属性

添加数据源之后，配置将显示在 Agent Editor 的“**运行时配置信息**”页面上。例如，第 1040 页的图 32 显示了配置节，以及在将 SNMP 事件属性组添加到代理程序时自动创建的一些配置属性。

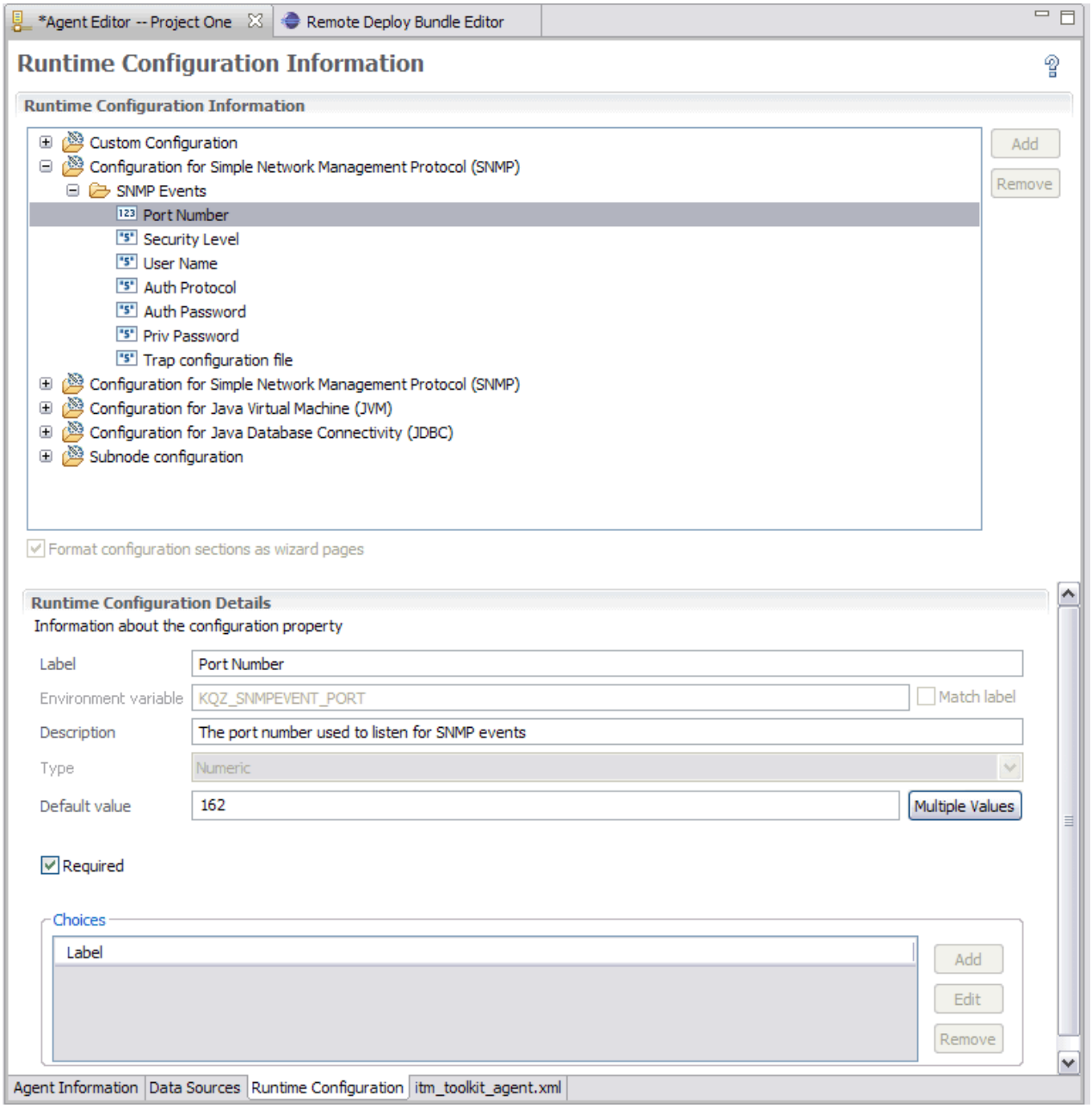


图 *32.* “运行时配置”页面

预定义配置属性的标签、描述和缺省值可以更改，但变量名称和类型不能更改。SNMP 事件配置节包含以下属性：

表 *266. SNMP* 事件配置属性

姓名	**有效值**	**是否必需**	**描述**
端口号	正整数	是	用于侦听事件的必需端口号
安全级别	noAuthNoPriv、authNoPriv 和 authPriv	否	SNMP V3 安全级别
用户名	字符串	否	SNMP V3 用户名
认证协议	MD5 或 SHA	否	SNMP V3 认证协议
认证密码	字符串	否	SNMP V3 认证密码

表 266. SNMP 事件配置属性 (续)			
姓名	**有效值**	**是否必需**	**描述**
专用密码	字符串	否	SNMP V3 隐私密码
陷阱配置文件	包括路径的文件名	否	陷阱配置文件的位置。如果使用此配置属性未找到该文件，会尝试在代理程序 bin 目录查找 `trapcnfg` 文件。

V1 或 V2C 事件不需要配置。无论是指定源名称还是共用名，都将处理所有 V1 或 V2C 事件。唯一支持的隐私协议为 DES，所以没有指定隐私协议的选项。没有必需的 SNMP V3 配置选项（可以选择指定任何选项）。如果要指定这些选项，那么必须为所选的安全级别指定适当的值。

测试 SNMP 事件属性组

您可以对在 Agent Builder 中创建的 SNMP 事件属性组进行测试。

开始之前

要测试 SNMP 事件属性组，请使用测试程序，或用于生成 SNMP 事件的应用程序。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**SNMP 事件信息**”窗口中的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述两个步骤中的某个步骤中单击**测试**之后，将打开“**测试事件设置**”窗口
2. 可选： 在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。有关 SNMP 事件配置属性的更多信息，请参阅第 1039 页的『SNMP 事件配置属性』。
3. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。

   当代理程序启动时，它会根据其配置侦听 SNMP 事件。

   注: 启动的代理程序是简化的版本，其中包含要测试的一个属性组。
4. 要测试代理程序的数据收集，需生成与代理程序配置匹配的 SNMP 事件。可以使用应用程序或事件生成器执行此操作。

   当代理程序接收到与其配置匹配的 SNMP 事件时，会将这些事件添加到其内部高速缓存。
5. 要模拟来自监视环境的代理程序数据请求，请单击**收集数据**。

   “**测试事件设置**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部事件。第 1042 页的图 33 中显示了示例数据收集

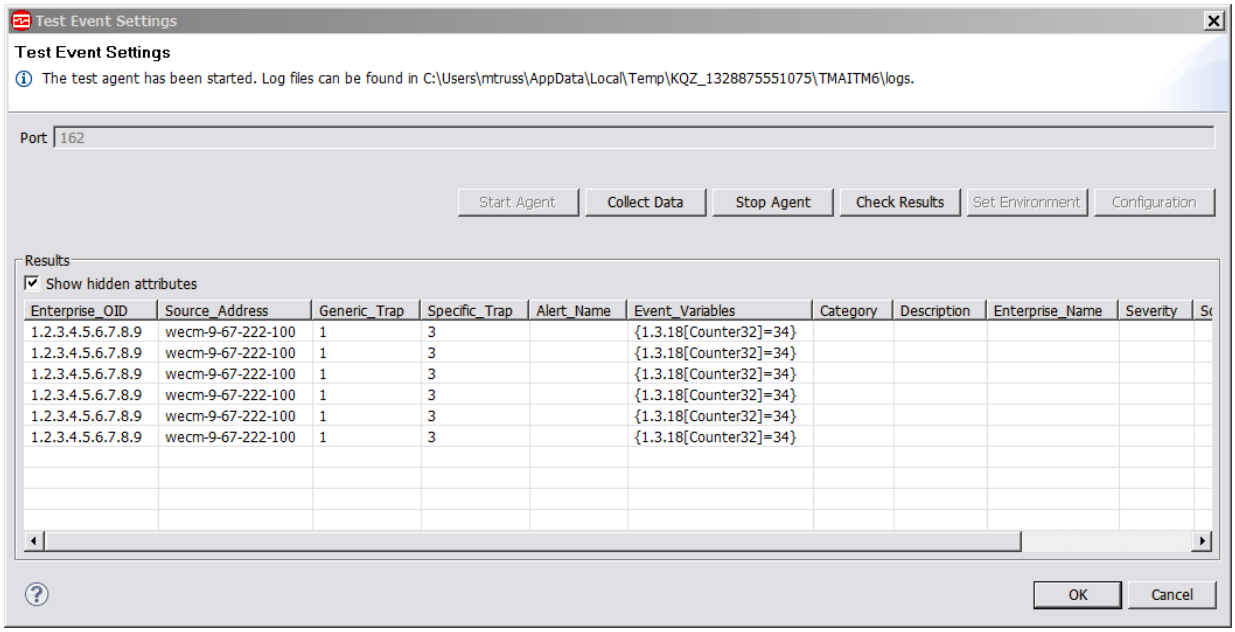


图 *33.* 显示所收集 *SNMP* 事件数据的“**测试事件设置**”窗口

6. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。（第 1042 页的图 34）中显示了示例。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据。

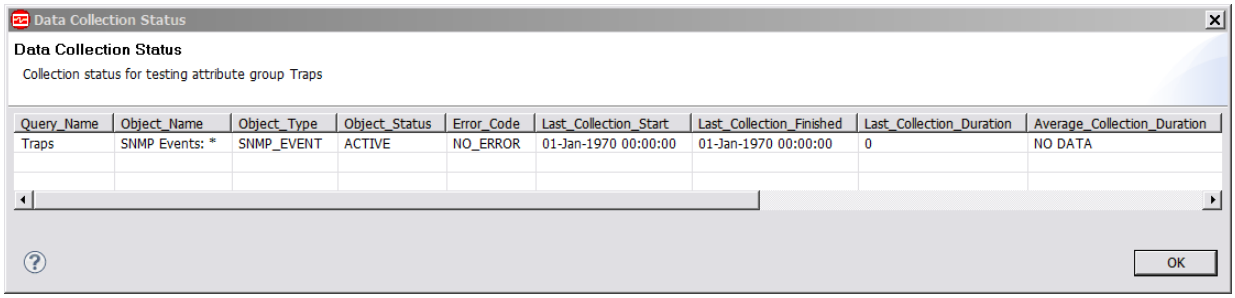


图 *34.* “**数据收集状态**”窗口

7. 通过单击**停止代理程序**停止代理程序。
8. 单击**确定**或**取消**以退出“**测试事件设置**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视 Java 管理扩展 (JMX) MBean

您可以定义数据源以从 JMX MBean 收集数据。将来自于每个受监视的 MBean 的数据放置在一个数据集中。根据 MBean，数据集可生成一行或者多行。

关于此任务

您定义的每个 JMX 数据源都必须标识单个 MBean（单个实例）或特定类型的 MBean（多个实例）。您必须知道 MBean 的对象名称或一种 MBean 类型（包含要收集的数据）的对象名称模式。对象名称模式只能用于标识一组相似的 MBean。与该模式匹配的这组 MBean 必须都能提供您要在监视表中查看的数据。典型的对象名称模式类似于 `*:j2eeType=Servlet,*`。此对象名称模式与具有 j2eeType 的 servlet 的所有 MBean 相匹配。您可能希望与该模式匹配的任何 MBean 都具有一组类似的、可添加到数据源的公开特性和操作。使用该模式的数据源将从与该模式匹配的每个 MBean 收集数据。您为此数据源定义的特性必须可供与此数据源对象名称模式相匹配的任何 MBean 使用。

支持 Java V5 或更高版本。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **JMX**。
3. 单击**下一步**。
4. 在“**JMX 信息**”页面上，单击**浏览**以查看 MBean Server 上的所有 JMX MBean。

   定义数据源之后，可以使用 Browse 功能预填充特性列表。然后可以添加、移除或修改浏览器插入的特性。这些特性的名称可能很长且难以正确输入。使用 Browse 选项可方便地输入正确的名称。

   注: 您可以手动创建 JMX 数据源，方法是指定对象名称并单击**下一步**，而无需使用浏览器。手动创建 JMX 数据源将创建两个数据源。将创建一个包含 JMX 通知的预定义特性的事件数据源。另外，还将定义集合数据源，其中包含必须在向导中指定的一个特性。

   MBean pattern
   显示 MBean 模式。

   Global JMX Options
   显示支持级别。

   为以下 JMX 服务器提供了支持:

   - Java 5 操作系统 MBean Server。通过使用 JSR-160 接口建立连接。支持通知和监视器。
   - WebSphere Application Server V6 和更高版本。对 SOAP 和 RMI 协议都提供了接口。由于远程代理程序无法创建 MBean，因此不支持 JMX 监视器。
   - WebSphere Community Edition 和其他基于 Apache Geronimo 的应用程序服务器。通过标准 JSR-160 接口建立连接。JMX 通知和监视器在 V1.1 和更高版本中受支持。
   - JBoss Application Server V4.0 和更低版本。
   - JBoss Application Server JSR-160 连接。
   - WebLogic Server V9 和更新版本。为 T3 协议提供了接口。
5. 在首次运行 JMX Browser 时，**MBean Server** 向下滚动菜单中没有任何项。要添加连接，请单击**添加**按钮。

   使用**编辑**按钮可修改或删除您已定义并在向下滚动菜单中选中的连接。连接定义存储在工作空间中，这样在创建连接时就会记住该定义。完成以下步骤来创建连接。如果您已有连接，请跳至下一步。

   a) 要创建与 MBean Server 的连接，请单击**添加**以添加连接，或者编辑现有连接。
      如果尚未定义任何连接，此时将显示“**Java 管理扩展 (JMX) 浏览器**”窗口。
   b) 在单击**添加**以添加连接之后，将打开“**选择连接类型**”页面。
   c) 使用“MBean Server 连接”向导来连接到 MBean Server。页面上所列的新连接是为创建连接可以选择的选项。您可以将现有连接作为模板，使用现有连接列表来创建新连接。选择其中一种新连接类型，然后单击**下一步**以开始创建连接。
   d) 在选择连接类型之后，可能会要求您选择更具体的连接类型。此时将显示基于**标准 JMX 连接 (JSR-160)**连接类型的两个模板。选择最合适您的 MBean Server 的模板，然后单击**下一步**。

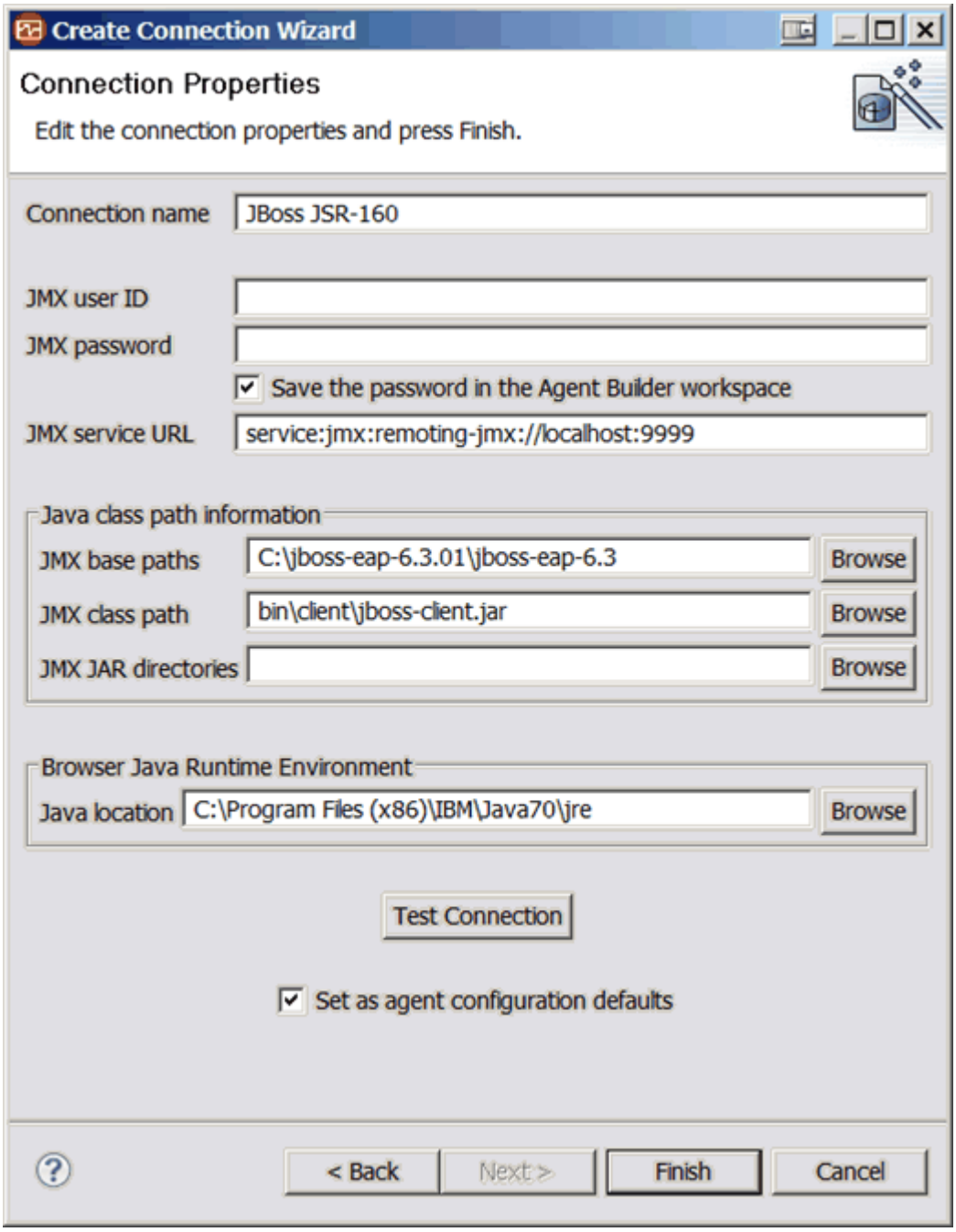


图 *35. JMX* 连接属性

“**连接属性**”页面（第 1044 页的图 35）包含有关如何连接到 MBean Server 的详细信息。您必须将 MBean Server 的相关详细信息填写到该页面中。

要点: 如果数据源连接到远程 WebSphere Application Server，请确保 WebSphere Application Server 也安装在运行 Agent Builder 的主机上，并将 **Java 位置**设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

e) 如果要保存此连接的密码，请选中**将密码保存在 Agent Builder 工作空间中**复选框。

f) 可选： 如果要从这些连接属性中复制 JMX 缺省值，请选中**设置为代理程序配置缺省值**。
例如，在第 1044 页的图 35 中，缺省的 **JMX 基本路径**是 `C:\jboss-eap-6.3.01\jboss-eap-6.3`，**JMX 服务 URL** 是 `service:jmx:remoting-jmx://localhost:9999`，而 **Java 位置**是 `C:\Program Files\IBM\Java70\jre`

1) 在指定连接所需的属性之后，请单击**测试连接**以确保可以建立此连接。如果连接失败，请更正必需的属性。

2) 连接成功后，请单击**完成**以返回到使用所配置连接的浏览器。

“**连接属性**”页面中的 Java 类路径信息包含三个字段。必须根据需要填写这三个字段才能连接到其所需 Java 类不在 Java 运行时环境中的 MBean Server。通常，您要连接的 MBean Server 必须与 Agent Builder 安装在同一系统上。在此情况下，请在 **JMX 基本路径**字段中指定包含 MBean Server 的应用程序的安装目录。然后，**JMX Jar 目录**字段将列出“基本路径”目录的相对目录，这些目录包含连接到 MBean Server 所需的 JAR 文件。**JMX 类路径**字段可以用于包含特定 JAR 文件。**JMX JAR 目录**字段中所列的 JAR 文件不需要单独列示在 **JMX 类路径**字段中。

任意字段都可包含多个引用；以分号分隔条目即可。这些值是配置代理程序时所需的值。有关更多信息，请参阅第 1049 页的『JMX 配置』。

6. 选择连接后，JMX Browser 将从 JMX 服务器下载有关 MBean 的信息。此信息显示在“**JMX 浏览器**”窗口的以下四个区域中（第 1046 页的图 36）：

从“Java 管理扩展 (JMX) 浏览器”窗口开始到 Agent Editor 的**运行时配置**选项卡的屏幕方向：从“**JMX 信息**”页面中，选择**浏览**。从浏览器（未选择连接的“JMX 浏览器”）中，选择**添加**。从“**JMX 连接选择**”页面中，选择 **JBoss**，然后选择**下一步**。从“**JMX 连接属性**”页面中，定制以下两个连接属性：JBoss 提供程序 URL 的值为 `jnp://wapwin3.tivlab.raleigh.ibm.com:1099/`，**JBoss Jar 目录** 的值为以下 JAR 文件所在目录的完整路径：`jbossall-client.jar`、`jboss-jmx.jar`、`jboss-jsr77-client.jar` 和 `jboss-management.jar`。选择**完成**。此配置将设置 JBoss 连接，这样您会看到类似以下所示的屏幕。

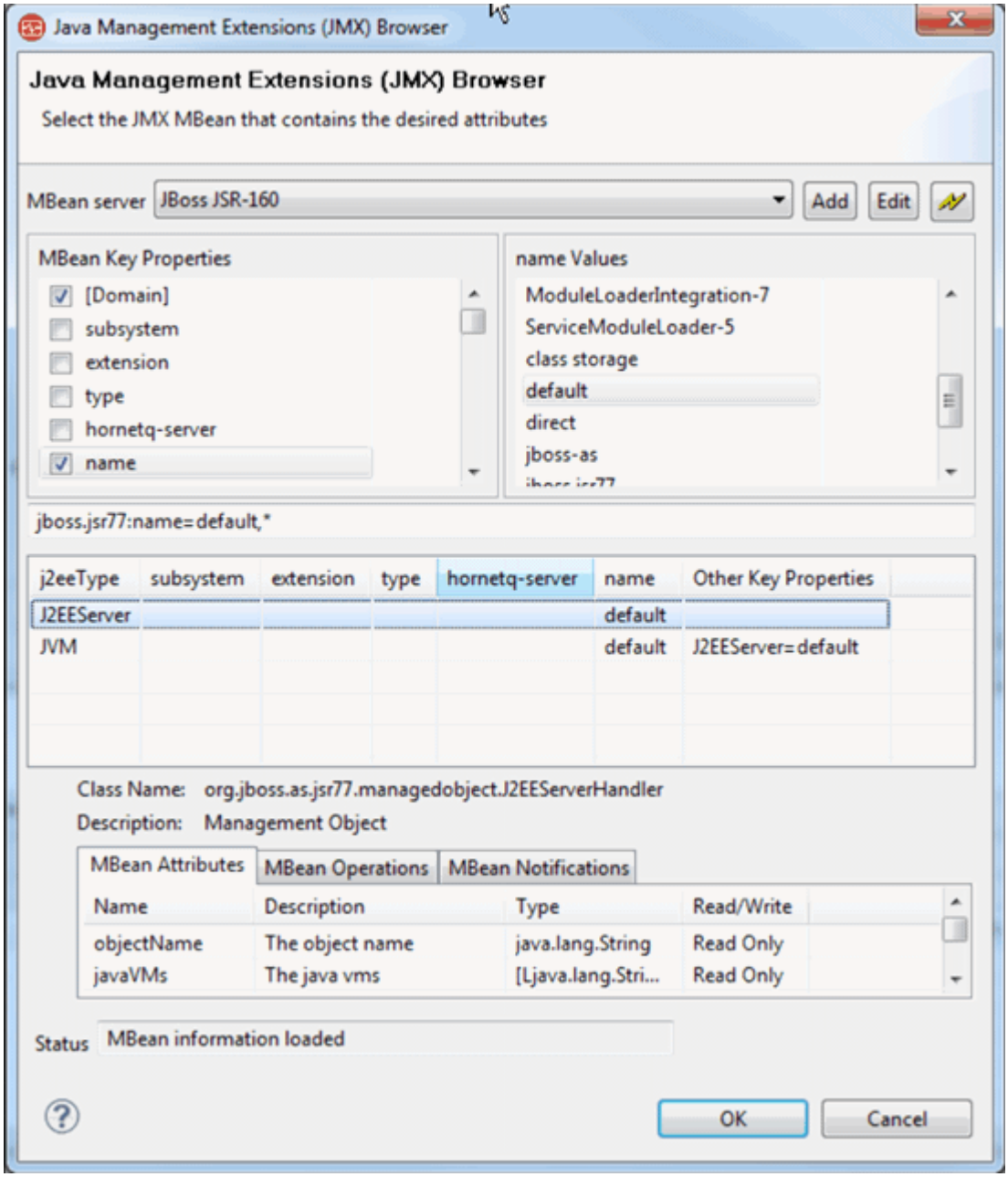


图 *36. Java Management Extensions (JMX) Browser* 窗口

- **MBean 键属性**区域：此区域是从服务器上所有 MBean 中发现的所有唯一对象名称键的集合。**[Domain]** 是特殊条目，因为它并不是键。但是，**[Domain]** 条目被视为 MBean 域值的隐式键。从此列表选择项，并找到包含该键属性的 MBean。键属性值列表显示在**选择的键属性值**列表中。当您勾选一个键属性后，它将包含在数据源的对象名称模式中。
- **选择的键属性值**区域：此区域显示当前从所有 MBean 中选择的“MBean 键属性”值。选择其中一个值也将勾选对应的 MBean 键属性。此选择还会将此 MBean 键属性的名称和值更新到消息字段中显示的对象名称模式。
- 一个列出所有与对象名称模式匹配的 MBean 的表：当您从 MBean Key Properties 和 Selected Key Property Values 列表中选择键属性和值后，您将看到对象名称模式更新。您还将看到该表中的 MBean 列表更改为反映与选定模式匹配的 MBean 列表。如果具有与任何 MBean 都不匹配的模式，那么您可以清除 MBean Key Properties 列表中的条目。通过单击模式所使用的键旁边的复选框并移除复选标记来清除条目。您也可以手动编辑模式，以查找需要的 MBean。模式 *:* 将选择所有 MBean。

  您可以使用此表来浏览服务器中的 MBean，然后确定哪些包含了您要监视的数据。要帮助浏览可能非常多的 MBean，可以按任意键特性排序（使用菜单或单击列标题）。您还可以从菜单中选择**显示键属**

性来显示任意列中的任意键特性。当您在表中看到标识了要监视的 MBean 的键属性时，右键单击该值并从菜单中选择**仅选择具有键属性的 MBean**。

· 包含有关选定 MBean 的详细信息的表：JMX Browser 向您显示单个 MBean 的相关信息。要查看某个 MBean 的详细信息，请从显示与当前过滤器匹配的 MBean 列表的表中选择该 MBean。有关 MBean 的键信息是其定义的特性、操作和通知的列表。

要从 JMX Browser 创建数据源，可使用之前描述的四个面板来构建对象名称模式。构建与一组 MBean（其中每个 MBean 都包含您要收集的监视数据）匹配的对象名称模式。例如，如果要监视来自所有 ThreadPool MBean 的数据，请使用以下步骤：

a) 从“**MBean 键属性**”面板选择**类型**。选择**类型**将导致更新“**选择的键属性值**”中的值，列出任意 MBean 的此类型键的所有唯一值。

b) 从此类型键的值列表中选择**线程池**。选择**线程池**之后，将在“**MBean 键属性**”面板中选中此类型键属性名称，并且对象名称模式将更新为 *:type=ThreadPool,*。MBean 列表也将更新，以仅显示与该模式匹配的 MBean。

c) 从 MBean 列表选择一个 MBean，以查看对该 MBean 可用的特性、操作和通知。如果 MBean 列表包含的 MBean 超出要监视的数量，那么您必须继续执行选择键属性和值的这一过程。继续该过程，直到您得到可识别要监视的 MBean 集合的对象名称模式。您也可以在 MBean 列表中打开一个菜单，以使用表中显示的键属性值更新对象模式。

7. 当对象名称模式正确时，从表选择一个 MBean。

   选定 MBean 的所有特性在新的 JMX 数据源中作为初始特性。某些特性可能不包含数据。创建 JMX 数据源后，请复审特性并移除所有不太重要的特性。如果选定的 MBean 没有任何特性，将警告您即将创建没有任何特性的数据源。如果选定的 MBean 包含通知，那么还将创建事件数据源以接收来自 MBean 的通知。

   要点: 对于每个 MBean 特性，Agent Builder 会在新数据集中创建一个特性。对于数字 MBean 特性，Agent Builder 会创建一个数字特性。对于任何对象类型，包括 `String`，Agent Builder 将创建一个字符串特性，其中包含值的字符串表示。如果 MBean 特性中对象的类型为 `javax.management.openmbean.CompositeData`，而且 Agent Builder 浏览器可读取该对象本身，那么它会创建多个特性，即 `CompositeData` 对象中嵌入的每个对象都有一个相应的特性。要包括非 `CompositeData` 对象的对象内部的值（字段或方法返回值），您需要创建一个具有更复杂度量名称的特性，如第 1055 页的『Java 管理扩展 (JMX) MBean 的特定字段』所述。

8. 在填写完毕的“JMX 信息”页面中，单击**完成**。

   数据源是根据在上一步中选定的 MBean 创建的。如果未选择任何 MBean，那么将创建一个没有任何特性的属性组。此时将显示警告，以允许您选择 MBean。通知数据源的数据源名称开头含有 **Event** 一词，用以与显示特性的数据源进行区分。

9. 要更改代理程序的其他 JMX 选项，可单击**全局 JMX 选项**。使用这些选项，您可以执行以下操作：

   a) 选择此代理程序是否支持 JMX 监视器。如果要创建 JMX 监视器属性组和“执行操作”命令，请选择**包括 JMX 监视器属性组及执行操作**。

   请参阅下一章节以获取 JMX 监视器的描述。

   b) 选择代理程序部署时要连接的 MBean 服务器类型。

   将列出若干特定于供应商的服务器类型，以及基于标准的服务器的通用 JSR-160 兼容服务器。您可以根据需要选择多个类型，但必须只选择支持要监视的 MBean 的服务器类型。必须至少选择一个类型。如果选择多个类型，那么将在配置代理程序时提示您指定要连接的服务器类型。

10. 选择所需选项之后，单击**确定**。
11. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅（第 1057 页的『测试 JMX 属性组』）
12. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关过滤来自属性组的数据的更多信息，请参阅第 1011 页的『过滤属性组』
13. 单击**下一步**。
14. 在“**选择键属性**”页面上，选择键属性或指示此数据源仅生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
15. 单击**下一步**。

“**JMX 代理程序范围的选项**”窗口将显示 Agent Builder 支持的应用程序服务器的类型。如果之前在“**连接属性**”页面上选择了**设置为代理程序配置缺省值**，那么将自动选择您所浏览到的应用程序服务器的类型。

16. 在“**JMX 代理程序范围的选项**”窗口（第 1048 页的图 37）中，选择您希望代理程序能够连接到的任何其他类型的应用程序服务器。

    注: 在所示示例中，选择 **JBoss Application Server JSR-160 连接**等同于选择**符合 JSR-160 标准的服务器**，不同之处在于所提供的缺省值不同。

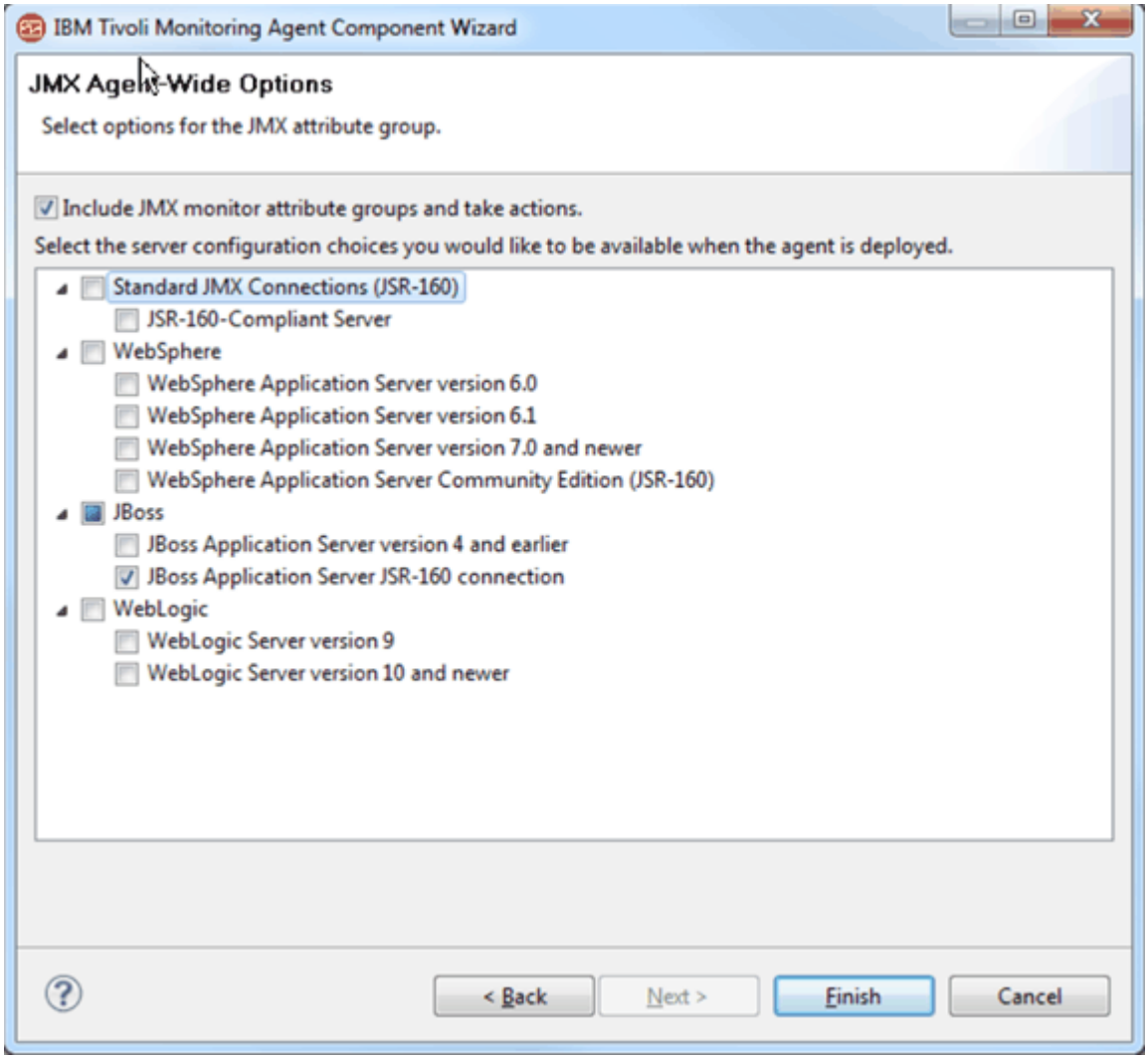


图 *37. JMX Agent-Wide Options* 窗口

17. 执行下列其中一个步骤:
    - 如果您正在使用“新建代理程序”向导，请单击**下一步**。
    - 单击**完成**，以保存该数据源并打开 Agent Editor。
18. 在创建代理程序之后，如果要更改可以连接的应用程序服务器类型，请单击 **JMX 数据源信息**区域中的**全局 JMX 选项**。
19. 在“**JMX 代理程序范围的选项**”窗口中，更改所需的任何选择。
20. 单击**确定**。
21. 要查看自动生成的配置节和属性，可单击 Agent Editor 的**运行时配置**选项卡。

    JBoss 基本路径属性的缺省值与 JMX Browser 中输入的值相同。

下一步做什么

有关 JMX 事件属性组的更多信息，请参阅 第 1221 页的『“JMX 事件”属性组』。

JMX 配置

在代理程序中定义 JMX 数据源时，将为您创建部分配置属性。

JMX 运行时配置是唯一的，因为它使您能够对所显示配置量进行一些控制。代理程序的 JMX 客户机可以连接到多种不同类型的应用程序服务器。但是，无需在任一代理程序中支持所有类型的应用程序服务器。您可以确定要支持哪些类型的应用程序服务器，且在代理程序中将不包含不必要的配置节。

在大多数情况下，代理程序设计为监视一种 JMX 应用程序服务器类型。在创建 JMX 数据源，可以使用 JMX Browser。在使用 JMX Browser 时，会自动向代理程序添加用于浏览 MBean Server 的 JMX 服务器配置选项。在创建代理程序之后，要更改可以连接的应用程序服务器类型，请单击“**JMX 信息**”区域中的**全局 JMX 选项**。在“**JMX 代理程序范围的选项**”页面中，更改所需的任何选择。

您可以设计用于监视多种类型的 JMX 应用程序服务器的通用代理程序。在此情况下，可以在“**JMX 代理程序范围的选项**”页面上选择多个 JMX 服务器配置选项。如果支持多种 JMX 连接类型，那么运行时配置会提示您选择将用于代理程序实例的连接类型。

注: 一个代理程序实例只能连接一种 JMX 应用程序服务器。在一个代理程序实例内，子节点可用于连接相同类型的不同 JMX 应用程序服务器。要连接多种 JMX 应用程序服务器，必须为每种 JMX 应用程序服务器类型配置至少一个代理程序实例。

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息，请参阅第 1145 页的『使用 Agent Editor 更改配置属性』。如果在子节点中定义了 JMX 数据源，那么还能指定“子节点配置覆盖”。有关指示信息，请参阅第 1134 页的『子节点配置』。

如果在代理程序中定义了 JMX 数据源，那么该代理程序必须使用 Java 来连接 JMX 应用程序服务器。Java 配置属性将自动添加到该代理程序中。

下列 Java 配置属性特定于代理程序运行时配置：

Java 主目录
指向 Java 安装目录的标准路径

将代理程序配置为使用的 JVM 与您所监视的应用程序使用的 JVM 相同，对于 WebLogic Server 和 WebSphere Application Server 尤其应这样配置。

JVM 自变量
向 Java 虚拟机指定可选自变量列表。

跟踪级别
定义要写入 Java 跟踪文件的信息量。缺省值为只将错误数据写入日志文件。

注: Agent Builder 不需要这些属性，因为它使用自己的 JVM 以及日志记录，这是通过 JLog 插件配置的。

如果在代理程序中定义 JMX 数据源，那么将向代理程序中自动添加以下必需的公共配置字段：

连接
与 MBean Server 的连接的类型

用户标识
用于向 MBean Server 认证的用户标识。

密码
用户标识的密码。

基本路径
用于在其中搜索**类路径**所指定 JAR 文件的目录，或者是 **JAR 目录**中指定的目录（非标准目录）。在 Windows 上使用分号（;）分隔目录名称，在 UNIX 系统上使用分号（;）或冒号（:）分隔目录名称。

类路径
代理程序将搜索的显式指定的 JAR 文件。在找到 JAR 文件之前，所有非标准文件都将追加到每个 Base Paths 中。

JAR 目录
要在其中搜索 JAR 文件的目录。在 Windows 上使用分号（;）分隔目录名称，在 UNIX 系统上使用分号（;）或冒号（:）分隔目录名称。不需要显式标识这些目录中的 JAR 文件；之所以找到这些文件是因为它

们确实在其中一个目录中。但不会搜索这些目录的子目录。在找到目录之前，所有非标准目录名都将追加到每个 Base Paths。

注: 对于远程监视，这些 JAR 文件及其所有从属 JAR 文件必须以本地方式安装在代理程序运行所在的计算机上。这些 JAR 文件是连接到受监视应用程序所必需的文件。这些 JAR 文件必须在 **JAR 目录**中配置，并且必须位于**基本路径**和**类路径**中。此外，以本地方式安装所监视应用程序支持的 JVM，并在 **Java 主目录配置**字段中指定路径。

示例:

- 对于 WebLogic 10，类路径为 `server/lib/wlclient.jar;server/lib/wljmxclient.jar`。基本路径指向 `server/lib` 目录所在的 WebLogic 应用程序服务器目录。
- 对于 WebSphere，基本路径指向 WebSphere Application Server 的安装位置。此示例中列出了多个基本路径，为 Windows 和 UNIX 提供了缺省值。类路径列出了相对于基本路径的 JAR 文件。**JAR 目录**字段的相对值 `lib` 导致装入基本路径下此目录中的所有 JAR 文件。
  - **基本路径**: `C:\Program Files\IBM\WebSphere\AppServer;/opt/IBM/WebSphere/AppServer`
  - **类路径**: `runtimes/com.ibm.ws.admin.client_6.1.0.jar;plugins/com.ibm.ws.security.crypto_6.1.0.jar`
  - **JAR 目录**: `lib`

根据在 JMX Agent-Wide Options 页面中选择的 JMX 服务器类型，将添加以下部分或全部配置属性。缺省值将由 Agent Builder 提供，且可以修改:

特定于 JSR-160 兼容服务器连接的配置属性:

JMX 服务 URL
要连接以进行监视的 JMX 服务 URL。

特定于 WebSphere Application Server V6.0 及更高版本连接的配置属性:

主机名
要监视的应用程序服务器所在系统的主机名。对于本地监视，此名称是本地系统名称。对于远程监视，此名称是应用程序服务器所在的系统的主机名。

端口
在要监视的主机名中使用的端口号。

连接器协议
监视连接要使用的接口协议。支持 RMI 和 SOAP。

概要文件名称
用于配置连接的概要文件的名称。

特定于 JBoss Application Server（非 JSR-160）连接的配置属性:

JNDI 名称
用于查找 MBean Server 的 JNDI 名称。

提供程序 URL
要连接以进行监视的 JMX 服务提供者 URL。

特定于 WebLogic Server 连接的配置属性:

服务 URL
要连接以进行监视的 JMX 服务提供者 URL（包含 JNDI 名称）。

注: 如果启用了 WebSphere 管理安全，那么必须确保在相应的客户机连接属性文件中禁用客户机登录提示。对于 RMI 连接，要阻止客户机提示用户，必须修改 WebSphere Application Server 概要文件属性目录中 `sas.client.props` 文件内的 *com.ibm.CORBA.loginSource* 属性。对于 SOAP 连接，必须修改同一目录中 `soap.client.props` 文件内的 *com.ibm.SOAP.loginSource* 属性。在这两种情况下，*loginSource* 属性都必须设置为不包含值。

您可以使用 Agent Editor 来查看、添加和更改配置属性。请参阅（第 1145 页的『使用 Agent Editor 更改配置属性』）。如果子节点中定义了 Windows 数据源，您还可以指定“子节点配置覆盖”。请参阅第 1134 页的『子节点配置』。

JMX 通知
除了应请求提供监视数据外，一些 MBean 还提供通知。

通知是由 MBean 所生成的对象，在发生事件时传递到注册侦听器。

Agent Builder 所构建的代理程序可以定义一些属性组以包含来自通知（而非 MBean）的值。

当代理程序启动后，通知侦听器会注册到与属性组的 MBean 模式匹配的所有 MBean。然后，属性组将显示收到的通知，每行显示一个通知。每列将包含一个来自通知的数据项。来自通知的所需数据由列值定义，方法与为 MBean 定义列数据类似。

对于并非基于事件的属性组，将在需要时收集数据。对于基于事件的属性组，代理程序将维护最近收到的 100 个事件的高速缓存。这些事件用于响应来自 Tivoli Enterprise Portal 的请求。这些事件将立即转发，以供情境进行分析以及进行仓储。

JMX 监视器
除了应请求提供监视数据外，一些 MBean 还提供监视。

JMX 提供者支持代理程序创建 JMX 监视器的能力。JMX 监视器是 JMX 代理程序在 JMX 服务器上创建的 MBean。它监视其他 MBean 的属性值并在该值符合某些条件时发送通知。定义的阈值使监视器能报告特定属性值。

并非所有应用程序服务器都支持从 JMX 客户机创建监视器，这对于当前发行版的 WebSphere Application Server 也是如此。可以将 JMX 监视器和“执行操作”命令包括在代理程序中，方法是选择“**全局 JMX 选项**”下的**包括 JMX 监视器属性组及执行操作**。

任何报告其他 MBean 属性的 MBean 都可视为监视器。实际上，JMX 定义三个具体的监视器类，这三个类都是创建的监视器类型。将创建以下具体的监视器类型：

- 字符串监视器 - 观察字符串属性，报告字符串是否相等。
- 标尺监视器 - 观察可变数字属性，报告超出阈值的上溢或下溢活动。
- 计数器监视器 - 观察增大数字属性，当达到阈值或增加了特定量时进行报告。

以下属性组可自动添加到代理程序，以收集或表示 JMX 监视器通知：

- 已注册的监视器

  此属性组显示用户添加的所有 JMX 监视器。
- 计数器通知

  此属性组报告从计数器监视器收到的所有通知。
- 标尺通知

  此属性组报告从标尺监视器收到的所有通知。
- 字符串通知

  此属性组报告从字符串监视器收到的所有通知。

针对 *JMX* 监视器的“执行操作”命令
监视器通过运行“执行操作”命令创建。

将定义三个“执行操作”命令，每一个命令用于创建一种类型的监视器，还将定义第四个“执行操作”，用于删除现有监视器。对“执行操作”命令应用了 256 个字符的限制。

监视器属性组是所构建的每个 JMX 代理程序（包括 Agent Builder 构建的所有代理程序）的一部分。这四个“执行操作”命令对所有代理程序都可用，但是它们不能用于非 JMX 代理程序。

JMX 添加字符串度量观察程序
使用此“执行操作”命令来创建监视器以观察字符串属性。

参数

MBean 模式
与此模式匹配的所有 MBean 受此监视器监视。

受观测属性
所观察的 MBean 字符串属性的名称。

通知匹配
当受监视字符串与参考值匹配时，如果将发送通知，那么为 true，否则为 false（缺省值为 false）。

通知差异
当受监视字符串与参考值不匹配时，如果将发送通知，那么为 true，否则为 false（缺省值为 true）

参考值
要与受观测属性进行比较的字符串。

缺省值表示未指定自变量。

示例：请求当服务停止时进行通知

```
STRING_METRIC_WATCHER [*:type=Service,*] [StateString] [true] [false] [Stopped]
```

其中：

`*:type=Service,*`
MBean 模式：监视具有键属性指定类型值 Service 的任意 MBean。

`StateString`
受观测属性：对所有类型为 `Service` 的 MBean 通用的字符串属性。

true
通知匹配：您希望在 `StateString` 属性与参考值 Stopped 匹配时，向代理程序发送通知。

false
通知差异：您不希望在 Service 属性与 Stopped 不匹配时收到通知。

Stopped
参考值：当 `StateString` 属性更改为值 Stopped 时，发送通知。

JMX 添加标尺度量观察程序
使用此“执行操作”命令来创建监视器以观察标尺属性。

参数

MBean 模式
与此模式匹配的所有 MBean 受此监视器监视。

受观测属性
所观察的 MBean 字符串属性的名称。

差异方式
如果受监视值等于属性的实际当前值与先前值的差，那么为 true。如果受监视值等于属性的实际当前值，那么为 false（缺省值为 false）。

阈值上限通知
当增大的受监视值超出阈值上限时，如果将发送通知，那么为 true，否则为 false（缺省值为 true）。

阈值下限通知
当减小的受监视值超出阈值下限时，如果将发送通知，那么为 true，否则为 false（缺省值为 true）。

阈值上限
受观测属性应保持低于此值。

阈值下限
受观测属性应保持高于此值。

示例：请求可用内存低于 10 MB 时进行通知

```
GAUGE_METRIC_WATCHER [ServerInfo] [FreeMemory] [false] [false] [true] [30000000] [10000000]
```

其中：

***:type=ServerInfo**
MBean 模式：监视名称中包含单个键属性指定类型值 ServerInfo 的任意 MBean。

FreeMemory
受观测属性：上下波动的数字属性，该属性指示应用程序服务器中的可用内存量。

false
差异方式：监视实际属性值，而不是某一观测值与另一观测值之间的差异。

false
阈值上限通知：可用内存增加时不发送通知。

true
阈值下限通知：可用内存变得过低时不发送通知。

30000000
阈值上限：即使您并不在意是否超过阈值上限，仍需要合理的阈值上限。在属性值达到或超过阈值上限之前，不会再次发出阈值下限通知。

10000000
阈值下限：希望得到有关通知的阈值下限。

JMX 添加计数器度量观察程序
使用此“执行操作”命令来创建监视器以观察计数器属性。

参数

MBean 模式
与此模式匹配的所有 MBean 受此监视器监视。

受观测属性
所观察的 MBean 字符串属性的名称。

初始阈值
用来与受观测属性比较的值。

偏离量
这是超过阈值后对阈值增加的值，用来创建一个更改的阈值。

模数
这是计数器的最大值，达到此值之后计数器回滚为 0。

差异方式
如果受监视值等于属性的实际当前值与先前值的差，那么为 true。如果受监视值等于属性的实际当前值，那么为 false（缺省值为 false）。此方式将有效开启更改率监视。

频率
执行测量的频率（缺省值为 20 秒）。如果差异方式为 true，此项将最重要

示例：请求当任意服务器包含三个或以上错误时进行通知

```
COUNTER_METRIC_WATCHER [*:j2eeType=Servlet,*] [errorCount] [3] [4] [] [diff] [gran]
```

其中：

:j2eeType=Servlet,
MBean 模式：监视名称中包含单个键属性指定类型值 `ServerInfo` 的任意 J2EE servlet MBean

errorCount
受观测属性：不断增大的数字属性，该属性指示 servlet 的错误数。

3
初始阈值：您希望在 `errorCount` 达到或超过 3 时收到通知。

4
偏移量：得到出现三个错误的通知时，将对先前阈值 3 加 4 以得到新的阈值 7。在 errorCount 达到 7 时，将发送第二个通知；达到 11 时，将发送第三个通知；达到 15 时，将发送第四个通知，以此类推。0 或“无”为无效值，因为预期计数器始终是增加的，而不增加偏移量对计数器没有任何意义。

模数：
errorCount 没有已构造的最大值，因此使用非常大的值。

false
差异方式：您关注的是绝对错误计数。如果所关注的是 errorCount 的增加率，差异将为 true。

频率：未设置，因此使用缺省频率 20 秒。频率对所有监视器类型可用。但是，它随计数器监视器一起显示，以便能够确定有意义的更改率（差异方式 = true）。

JMX 删除度量观察程序
使用此“执行操作”可删除监视器。

参数

编号
REGISTERED_MONITORS 表中所显示的监视器编号

示例：删除 2 号监视器

```
DELETE_WATCHER [2]
```

其中：

2=
要删除的监视器的编号。

***JMX* 操作**
除了应请求提供监视数据外，一些 MBean 还提供操作。

具有 JMX 数据源的代理程序包含 JMX_INVOKE“执行操作”命令，您可使用该命令来针对所监视的服务器运行 JMX 操作。

“执行操作”命令语法

该操作语法如下：

```
JMX_INVOKE [MBean pattern] [Operation name] [Argument 1] [Argument 2]
[Argument 3] [Argument 4]
```

其中：

MBean pattern
用来选择要对其运行操作的 MBean 的 MBean 查询。如果该模式与多个 MBean 相匹配，那么该操作在每个匹配的 MBean 上运行。

Operation name
要运行的 MBean 操作的名称。

Argument 1、Argument 2、Argument 3 和 Argument 4
可以提供给 MBean 操作的可选自变量。自变量必须是简单数据类型，如字符串或整数。

如果操作成功运行，那么 JMX invoke“执行操作”命令返回“成功”。如果操作返回一个值，该值将写入 JMX 数据提供程序日志文件。

示例：启动用于重置计数器的操作

此操作对 Threading MBean 运行 resetPeakThreadCount 操作：

```
JMX_INVOKE [*:type=Threading,*] [resetPeakThreadCount][] [] [] []
```

其中：

:type=Threading,
MBean 模式：此模式与类型为 Threading 的所有 MBean 相匹配。

resetPeakThreadCount
操作名称：在与该模式相匹配的每个 MBean 上运行的操作。

[] [] [] []
自变量 1、2、3 和 4：这些自变量并非此操作需要的自变量。指定它们只是为了符合该操作的语法。

示例：带自变量启动操作

此操作对 Threading MBean 运行 getThreadCpuTime 操作。结果记录到 JMX 数据提供程序跟踪文件中。

```
JMX_INVOKE [*:type=Threading,*] [getThreadCpuTime] [1] [] [] []
```

其中：

:type=Threading,
MBean 模式：此模式与类型为 Threading 的所有 MBean 相匹配

getThreadCpuTime
操作名称：在与该模式相匹配的每个 MBean 上运行的操作。

1
自变量 1：被查询的线程标识。

[] [] []
自变量 2、3 和 4：这些自变量并非此操作需要的自变量。将它们指定为空自变量是为了符合“执行操作”命令语法。

运行 JMX_INVOKE“执行操作”命令

代理程序开发者无法预见到用户会运行 JMX_INVOKE“执行操作”命令。而是必须开发更多运行 JMX_INVOKE“执行操作”的操作。如果可能，在这些操作中向用户隐藏一些详细信息，如操作名称和 MBean 模式。

启动和停止 *JMX* 监视器

JMX 监视器在启动和停止代理程序与 JMX 服务器时始终存在。

如果代理程序检测到 JMX 服务器已重新启动，将重新注册监视器。如果代理程序重新启动，也将重新注册监视器。监视器定义存储在名为 `default_`*`instanceName`*`.monitors` 的文件中，其中 *instanceName* 是代理程序实例名称，如果只有一个代理程序实例，这是缺省名称。该文件位于以下目录中（请注意，*xx* 表示两个字符的产品代码）：

- Windows 系统：`TMAITM6/k`*`xx`*`/config`
- UNIX 和 Linux 系统：*`architecture`*`/`*`xx`*`/config`（请参阅第 1170 页的『系统上的新文件』以了解有关确定体系结构值的信息）

如果重新启动代理程序，它将使用监视器定义文件来恢复监视器。

Java 管理扩展 (JMX) MBean 的特定字段

在“**特性信息**”窗口上进行指定时，JMX 属性组的度量名称语法必须遵循某些规则。

JMX 属性组的度量名称语法包含以句点分隔的标记。这些标记组成了主要的值以及可选的辅助值：

- **主值**：这是在表的特定行中从 MBean 或“通知”直接获取的值。来自 MBean 的主要值可从 MBean 特性或通过 MBean 操作调用（方法调用）获取。来自“通知”的主要值可从“通知”对象上的字段或方法调用获取。主要值可以是基本类型，也可以是 Java 对象。
- **辅助值**：这是通过进一步处理主要值或其他辅助值获取的值。辅助值是在引擎内部处理的，且不会涉及对 JMX 服务器的调用。如果主要值（或其他辅助值）是 Java 对象，那么辅助值将是访存该对象公共字段的结果。辅助值还可以是该对象的方法调用的结果。此类辅助值通过使用主要（或其他辅助）Java 对象的 Java 自省获得。如果主要（或其他辅助）值是形式为 MBean 名称的 Java 字符串，那么辅助值可以是域。辅助值还可以是组成该 MBean 名称的任意属性。

以下语法描述**度量名称**字段的格式：

```
Metric Name    =    PrimaryValue [ .SecondaryValue ]
PrimaryValue    =    Attribute.attributeName |
        Method.methodName |
        Domain |
        Property.propertyName |
        Field.fieldName |
        Name
SecondaryValue    =    Field.fieldName |
        Method.methodName |
        Domain |
        Property.propertyName |
        Explode |
        ElementCount
```

```
propertyName    =    the name of a key property in an MBean ObjectName
attributeName    =    the name of an MBean attribute
methodName    =    a zero-argument operation of an MBean or a zero-argument method
of a Notification or other Java object.
methodName(argument) = A single-argument operation of an MBean or a
single-argument method of a Notification or other Java object. The
argument will be passed to the method as a string.
fieldName    =    the name of a public instance variable in a Notification or
other Java object
notificationMethod    =    the name of a public zero-argument method of a
Notification object
```

如果在度量名称定义中仅包含主要值，那么收集的数据可以是以下任意项：

- MBean 域
- MBean 字符串值
- MBean 名称的键属性
- MBean 特性的数字或字符串特性值（包含另一 MBean 的全名）。来自 MBean 操作的数字或字符串返回值。
- “通知”对象中的数字或字符串公共实例变量的值
- 来自“通知”操作的数字或字符串返回值。

通过向度量定义添加辅助值，您可以向下钻取到 Java 对象的主要值。此外，还可以启动公共方法或访存公共实例变量。

通过向度量定义中另一辅助值添加辅助值，您可以向下钻取到辅助值对象。MBean 或“通知”内部嵌套了多少对象，您就可以向下钻取到所对应的深度。

组成主要值和辅助值的标记可以是关键字或名称。在大多数情况下，关键字标记后跟名称标记。下表显示了某些示例：

度量名称样本	**属性组类型**	**返回数据的描述**
域	MBean	MBean 的域部分（冒号之前的部分）。
Name	MBean	MBean 的完整字符串表示法。
Attribute.serverVendor	MBean	MBean 特性 serverVendor。
Method.getHeapSize	MBean	MBean 上由 getHeapSize() 返回的值。
Property.j2eeType	MBean	从 MBean 名称抽取的 j2eeType 的值。
Field.Message	事件（通知）	通知中的**消息**字段。

关键字 Attribute、Method 和 Field 可以返回包含其他数据的 Java 对象。您可以通过追加辅助值定义对那些对象运行操作。更多示例：

度量名称样本	**属性组类型**	**返回数据的描述**
Attribute.deployedObject.Method.getName	MBean	从 MBean 获取 deployedObject 特性，并得到 getName() 方法的结果。

度量名称样本	属性组类型	返回数据的描述
Attribute.eventProvider.Method.getException.Method.getDescription	MBean	此过程分三步：假设一个名为 eventProvider 的特性是具有 getException() 方法的对象。此方法通过 getDescription() 方法返回一个对象。调用此方法，并将返回值放至列中。
Attribute.HeapMemoryUsage.Method.get(used)	MBean	从 MBean 获取 HeapMemoryUsage 特性，并获取 get(String value) 方法的结果。所使用的字符串作为自变量传递到此方法。只能提供一个自变量，并且该自变量必须是文字串值。 显示如何从开放式 MBean 组合数据结构收集数据。

如果上一个值返回格式为 MBean 名称的字符串，那么 `Domain` 和 `Property` 可用作辅助值中的关键字。例如：

度量名称样本	属性组类型	返回数据的描述
`Attribute.jdbcDriver.Property.name`	MBean	特性 jdbcDriver 返回 MBean 名称，键属性 name 从该 MBean 名称中抽取。
`Attribute.jdbcDriver.Domain`	MBean	特性 jdbcDriver 返回 MBean 名称，domain 从该 MBean 名称中抽取。

`ElementCount` 和 `Explode` 关键字对数组或数据集合运行操作。

· `ElementCount` - 返回数组中的元素数量。

· `Explode` - 将一行拆分为多行，数组的每个元素对应一个新行。

每个关键字的示例：

度量名称样本	属性组类型	返回数据的描述
Attribute.deployedObjects.ElementCount	MBean	MBean 特性 deployedObjects 是一个数组，且该列包含数组中元素的数量。
Attribute.deployedObjects.Explode.MBean.Property.j2eeType	MBean	将使所部署对象的每个元素在表中都有一个对应的行。此列包含部署对象的 j2eeType。
Attribute.SystemProperties.Method.values.Explode.Method.get(key)	MBean	使开放式 MBean 表格数据结构中的每个条目都能获得一个新行。每个表格数据结构都包含组合数据结构，此结构具有返回的名为 key 的项。

测试 JMX 属性组

您可以对在 Agent Builder 中创建的 JMX 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   · 在创建代理程序期间，单击“**JMX 信息**”页面上的**测试**。
   · 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**JMX 测试**”窗口
2. 从**连接名称**下提供的列表中选择连接，或者单击**添加**以添加连接，并执行第 1042 页的『监视 Java 管理扩展 (JMX) MBean』中详细描述的过程。

3. 可选： 在开始测试之前，您可以设置环境变量、配置属性和 Java 信息。
   有关更多信息，请参阅第 1156 页的『属性组测试』。有关 JMX 配置的更多信息，请参阅 第 1049 页的『JMX 配置』。
4. 单击**启动代理程序**。
   此时将显示一个窗口，表明代理程序正在启动。
5. 单击**收集数据**可模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求。
   代理程序会监视 JMX 服务器以获取数据。“**JMX 测试**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
6. 可选： 如果返回的数据未如预期，请单击**检查结果**。
   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据
7. 通过单击**停止代理程序**停止代理程序。
8. 单击**确定**或**取消**以退出“**JMX 测试**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视来自公共信息模型 (CIM) 的数据

您可以定义数据源以从公共信息模型 (CIM) 数据源接收数据。数据源监视单个 CIM 类，并将此类中的所有值放在它所产生的数据集中。如果类提供多个实例，那么数据集包含多行；您可以按实例名称进行过滤以确保数据集仅包含一行。

关于此任务

此任务描述了配置公共信息模型 (CIM) 数据源的步骤。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**来自服务器的数据**。
2. 在**数据源**区域中，单击 **CIM**。
3. 单击**下一步**。
4. 在“**公共信息模型 (CIM) 信息**”页面的 **CIM 信息**区域中，选择下列其中一个选项：
   - 填写**名称空间**和 **CIM 类名**字段，以指定您想要收集的数据。
   - 单击**浏览**，以浏览特定系统上的 CIM 存储库。

     这将显示“**公共信息模型 (CIM) 类浏览器**”窗口。此浏览器会连接至 CIM 服务器，并为您提供有关该服务器上存在的类的信息。

     要浏览远程系统，请从**主机名**列表中选择系统（如果已定义）。或者，单击**添加**以添加 CIM 服务器所在的系统的主机名。

     用于指定主机名的语法为 `http[s]://hostname:port`。如果您只提供了主机名，那么公共信息模型 (CIM) 类浏览器将使用缺省 URL `http://hostname:5988` 进行连接。

     如果您提供了协议，但未指定端口，那么将使用 5988 作为缺省值（对于 `http`），或使用 5989 作为缺省值（对于 `https`）。

     如果提供了端口但未指定协议，那么 `http` 将与提供的端口一起使用。

     提供帐户的用户标识和密码，该帐户需对您想浏览的名称空间中的对象具有读许可权。将使用远程系统的信息对此窗口进行更新。

     Agent Builder 将尝试发现 CIM 服务器上可用的名称空间。发现的名称空间将显示在**名称空间**列表中。但是，Agent Builder 可能只能发现该服务器上的部分可用名称空间。如果您想要浏览未在**名称空间**列表中列出的名称空间，请单击**名称空间**列表旁边的加号 (+) 图标。在字段中输入该名称空间的

名称，然后单击**确定**。如果 CIM 服务器上存在该名称空间，那么将列示该名称空间中定义的类。您输入的名称空间将保存下来，并在您下一次浏览该特定 CIM 服务器时放入**名称空间**列表。

您从**名称空间**列表中选择名称空间时，Agent Builder 会收集该特定名称空间的所有类信息。然后，Agent Builder 将缓存该信息，以使您能够在名称空间之间快速切换。如果您要强制 Agent Builder 重新收集特定名称空间的类信息，请选择该名称空间，然后单击**连接**。单击**连接**将删除全部已缓存的信息，并导致 Agent Builder 重新收集类信息。

您可以单击**搜索**（望远镜）图标，以查找列表中的所选项。请在**搜索短语**字段中输入短语；通过单击**按名称搜索**或**按类属性搜索**字段指定首选项；然后单击**确定**。如果找到所搜索的项，请将其选中，然后单击**确定**。

5. 在“公共信息模型 (CIM) 信息”页面上的**操作系统**区域中，选择要执行此收集的操作系统。
6. 如果在 **CIM 信息**区域中输入了名称空间和 CIM 类名，请执行下列步骤：
   a) 单击**下一步**以显示“**属性信息**”页面，并定义属性组中的第一个属性。
   b) 指定“**属性信息**”页面的相关信息，然后单击**完成**。
7. 如果您已浏览 CIM 信息，那么将显示“选择键属性”页面。在“选择键属性”页面上，选择键属性或指示该数据源只生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
8. 如果您已浏览到 CIM 信息，请单击**完成**。
9. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1060 页的『测试 CIM 属性组』。
10. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息，请参阅第 1011 页的『过滤属性组』
11. 执行下列其中一个步骤：
    a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。
    b) 单击**完成**，以保存该数据源并打开 Agent Editor。

CIM 配置
有关 CIM 配置属性的详细信息。

如果在代理程序中定义 CIM 数据源，CIM 配置属性将自动添加到代理程序。您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息，请参阅第 1145 页的『使用 Agent Editor 更改配置属性』。如果在子节点中定义了 CIM 数据源，那么请指定“子节点配置覆盖”。有关指示信息，请参阅第 1134 页的『子节点配置』。

以下特定于连接的配置属性在 CIM 配置页面上：

CIM 本地或远程
向 CIM 服务器进行本地或远程认证。“本地/远程”缺省值为远程

CIM 用户标识
用于访问 CIM 服务器的用户标识

CIM 密码
用于访问 CIM 服务器的密码

CIM 主机名
要访问以获取 CIM 数据的主机名

使用 SSL 进行 CIM 通信
使用 SSL 与 CIM 服务器通信。选项为“是”和“否”。缺省值为“否”。

CIM 端口号
用于非安全通信的端口号。

CIM SSL 端口号
用于安全通信的端口号。缺省值为 5989。（Solaris 8 的缺省值通常不同。）

测试 CIM 属性组

您可以对在 Agent Builder 中创建的 CIM 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**CIM 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**测试设置**”窗口
2. 可选： 请先设置环境变量及配置属性，然后再开始测试。

   有关更多信息，请参阅第 1156 页的『属性组测试』。
3. 选择或添加**主机名**。

   有关添加**主机名**的更多信息，请参阅第 1058 页的『监视来自公共信息模型 (CIM) 的数据』。
4. 单击**启动代理程序**。

   这将打开一个窗口，指示代理程序正在启动。
5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。

   代理程序将查询 CIM Server 以获取数据。“**测试设置**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
6. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据
7. 通过单击**停止代理程序**停止代理程序。
8. 单击**确定**或**取消**以退出“**测试设置**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视日志文件

您可以定义数据源以从文本日志文件接收数据。代理程序定期解析添加日志文件的行，并基于这些行生成事件信息。您可以配置代理程序将日志解析为事件的方式。您还可以配置代理程序以过滤和汇总数据。产生的事件放置在数据集中。

开始之前

注: 代理程序监视的日志文件与代理程序运行所使用的语言环境和代码页相同。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**记录的数据**。
2. 在**数据源**区域中，单击**日志文件**。
3. 单击**下一步**。
4. 在“**日志文件信息**”页面上的**日志文件信息**区域中，输入要监视的日志文件的名称。

   文件名必须是标准文件名。
   a) 可选： 日志文件名的一部分可能取自运行时配置属性。要创建日志文件名，可单击**插入配置属性**并选择配置属性。
   b) 可选： 此文件也可以是动态文件名。有关更多信息，请参阅第 1266 页的『动态文件名支持』。
5. 在**字段标识**区域中，单击下列其中一个选项：

   Fixed number of characters
   如果选择此选项，将限制字符的数量。

通过使用此选项，将为每个特性分配它可容纳的日志文件中的最大字符数。例如，如果有三个特性 A、B 和 C（按此顺序），并且每个特性是最大长度为 20 的字符串。那么日志记录的前 20 个字节将进入 A，接下来的 20 个字节进入 B，再接下来的 20 个字节进入 C。

制表符分隔符
如果选择此选项，那么可使用制表符分隔符。

Space separator
如果选择此选项，那么多个连续空格将用作一个分隔符。

Separator Text
如果选择此选项，需输入分隔符文本。

Begin and End Text
如果选择此选项，需输入开始和结束文本。

XML in element
如果选择此选项，输入要用作记录的 XML 元素的名称，或者单击**浏览**以定义元素。

如果您已单击**浏览**，则会显示“**XML 浏览器**”窗口。如果使用 Browse 功能，Agent Builder 将通过查看子标记及其特性来识别记录的所有可能特性。

注: 除非单击**高级**并填写该窗口中的信息，否则对于您所填写的信息，将作出以下假设：

· 一次仅监视一个日志文件。

· 日志文件的每行都包含填写要定义的特性所必需的所有字段。

有关日志文件解析和分隔符的更多信息，请参阅（第 1067 页的『日志文件解析和分隔符』）。

6. 可选： 在“**日志文件信息**”页面上单击**高级**，使用“**高级数据源属性**”页面完成以下操作：

· 监视多个文件，或监视不同操作系统上不同名称的文件，或监视名称与正则表达式匹配的文件。

· 从日志文件中的多行提取一组字段。

· 选择**事件过滤和摘要选项**。

· 生成输出摘要信息。该摘要在每个时间间隔都会再生成一个属性组。有关此属性组的更多信息，请参阅第 1206 页的『日志文件摘要』。此功能已被 Event Information 选项卡中提供的选项取代。

a) 要监视多个日志文件，请单击**添加**并输入名称。
如果列出了多个文件，那么必须为每个文件输入唯一的标签。标签可作为特性显示以指示哪个文件生成了记录。标签不得包含空格。

b) 可选： 要选择待监视的每个日志文件所在的操作系统，请执行下列步骤：

1) 单击日志文件的**操作系统**列。

2) 单击**编辑**。

3) 在“**操作系统**”窗口中，选择操作系统。

4) 单击**确定**以保存更改，并返回到“**高级数据源属性**”页面。

c) 可选： 如果您提供的文件名是一个正则表达式，而该正则表达式用于查找文件，而不是用作文件名，请选择**文件名与正则表达式匹配**。
有关更多信息，请参阅第 1259 页的『ICU 正则表达式』。如果未选中此复选框，那么此名称必须是实际的文件名。或者，它必须是遵循第 1267 页的『动态文件名语法』中所述文件名模式规则的模式。

d) 可选： 选择**一个目录元素与正则表达式匹配**以将文件名路径的一个子目录与正则表达式匹配。
如果在上一步中选择了**文件名与正则表达式匹配**，则可同时选择此选项。

如果在路径名中使用了正则表达式元字符，那么只能在路径的一个子目录中使用这些元字符。例如，可以指定 `/var/log/[0-9\.]*/mylog.*` 以在一个子目录中包含元字符。`[0-9\.]*` 将导致与仅包含数字和点 (`.`) 的 `/var/log` 任意子目录匹配。`mylog.*` 将导致与那些 `/var/log` 子目录中所有以 `mylog` 开头并且后跟零个或零个以上字符的任意文件名匹配。

某些操作系统使用反斜杠 (`\`) 作为目录分隔符，这可能会与正则表达式转义元字符混淆。为了避免此类混淆，必须始终使用正斜杠来指示目录。例如，指定为 `C:\temp\mylog.*` 的 Windows 文件

可能表示 \t 是缩写的制表符。因此，在所有操作系统上，请始终使用正斜杠 (/) 作为目录分隔符。C:/temp/mylog.* 示例表示 C:/temp 目录中所有以 mylog 开头的文件。

e) 在**当多个文件匹配时**列表中，选择下列其中一个选项：

- **文件名中的数字值最大的文件**
- **最大的文件**
- **更新时间最近的文件**
- **创建时间最近的文件**
- **所有匹配的文件**

注: 在选择**所有匹配的文件**时，代理程序将识别目录中与动态文件名模式匹配的所有文件。代理程序以并行方式监视所有文件的更新。所有文件中的数据会在数据收集过程中混和。最好是在**记录字段信息**中选择**日志文件名**以添加特性，从而将日志消息和包含该日志消息的日志文件关联起来。确保与动态文件名模式匹配的所有文件都可以采用一致的方式分割成多个特性。如果无法一致地解析所选日志文件，那么最佳做法是在**记录字段信息**中选择**整个记录**来定义单个特性。有关为特性指定**记录字段信息**的更多信息，请参阅步骤 第 1063 页的『8』。

f) 选择处理文件的方式。

通过**对文件进行采样时处理所有记录**，每次为日志监视器定义的采样时间间隔到期时，您都可以处理整个文件中的所有记录。缺省时间间隔为 60 秒。您可以使用 *KUMP_DP_COPY_MODE_SAMPLE_INTERVAL* 环境变量（指定以秒为单位的值）来修改此时间间隔。除非从文件中移除这些记录，否则每次都会报告相同的记录。如果使用该选项，那么向文件中写入新记录时不会生成事件数据。通过**处理追加到文件中的新记录**，您可以处理在代理程序运行期间追加到文件的新记录。对于添加到文件中的每个记录，都会生成一个事件记录。如果替换了文件（第一个记录以任何方式发生更改），那么将处理文件，并且对于文件中的每条记录生成一个事件。

注: 如果将记录追加到 XML 日志文件，那么追加记录必须包含您选作**字段标识**的 XML 元素中定义的一组完整元素。

g) 如果选择处理追加到文件中的新记录，那么还可以选择检测新记录的方式。

通过**记录计数增大时检测新记录**，当文件中的记录数增加时，无论文件大小是否更改，都可以检测到新记录。在向日志文件中写入任何记录之前预先分配整个文件时，该功能很有用。对于非预先分配的文件也可以选择该选项，但与监视文件的大小相比，此选项效率较低。通过**文件大小增大时检测新记录**，您可以确定何时通过典型方法向文件追加新条目。在识别受监视文件被替换时，可能会有一个短暂的延时。

h) 如果选择了**文件大小增大时检测新记录**，则还可以选择如何处理监视代理程序启动时存在的文件。

忽略现有记录禁止在代理程序启动时为文件中的任意记录产生事件。**处理文件中的 ___ 个现有记录**指定在代理程序启动时，为文件结尾处固定数量的记录产生事件。**处理代理程序先前未处理的记录**：针对要由监视代理程序维护的重新启动数据指定，以便代理程序知道其上次运行时处理了哪些记录。对于自上次代理程序运行以来追加到文件中的任何记录都生成事件。每次将记录添加到文件时，此选项都会涉及一些额外的处理。

i) 如果选择了**处理代理程序先前未处理的记录**，则可以选择在代理程序启动时明显替换了现有文件的情况下要执行的操作。

在文件替换后处理所有记录：如果有关受监视文件的信息和重新启动数据信息不匹配，那么会为文件中的所有记录产生事件。不匹配的示例包括：文件名不同、文件创建时间不同、已减小文件大小、文件上次修改时间早于原先的时间。**在文件替换后不处理记录**：如果有关受监视文件的信息和重新启动数据信息不匹配，那么会禁用对文件中现有记录的处理。

j) 单击**记录标识**选项卡可将日志文件中的多行解释为单个逻辑记录。

注: 如果在“**日志文件信息**”页面上选择**元素中的 XML** 作为字段标识，那么不会显示**记录标识**选项卡。

- **单一行**：将每个行解释为单个逻辑记录。
- **分隔行**：当您输入一串字符时，此行标识用于分隔记录的行。

  注: 分隔符行不是上一记录或下一记录的组成部分。

· **规则**：标识构成记录的最大行数，并可以选择性地标识一串字符以指出记录的开头或结尾。通过**规则**，您可以指定以下属性：

– **最大非空白行数**：定义规则可以处理的最大非空白行数。

– **规则类型**可以是下列其中一项：

- **不比较文本**（每个记录的最大行数指示单个逻辑记录）。
- **识别记录开头**（标记单个逻辑记录的开头）。
- **识别记录结尾**（标记单个逻辑记录的结尾）。

– **偏移量**：指定行中必须出现“比较字符串”的位置。

– **比较测试**：可以是**等于**（要求在特定偏移位置存在字符序列匹配项）或 **不等于**（指示在特定偏移位置不出现特定字符序列）。

– **比较字符串**：定义要比较的字符序列。

· **正则表达式**：标识用于指示记录开头或结尾的模式。通过使用**正则表达式**，可以指定以下属性：

– **比较字符串**：定义要匹配的字符序列。

或者

– 记录的开头或结尾：

- **识别记录开头**标记单个逻辑记录的开头。
- **识别记录结尾**标记单个逻辑记录的结尾。

k) 如果您之前已选择**对文件进行采样时处理所有记录**，请单击**过滤表达式**选项卡。通过单击**过滤表达式**，您可以根据一个或多个特性的值和/或配置变量的值来过滤返回为行的数据。

如果您之前已选择**处理追加到文件中的新记录**，那么无法创建过滤表达式。有关过滤来自属性组的数据的更多信息，请参阅（第 1011 页的『过滤属性组』）。

l) 如果您之前已选择**处理追加到文件中的新记录**，请单击**事件信息**选项卡，以选择**事件过滤和摘要选项**。

有关更多信息，请参阅第 1180 页的『事件过滤和摘要』。

注: 如果代理程序是使用先前版本的 Agent Builder 创建的，那么可能存在 Summary 选项卡。现已由 Event Information 选项卡取代了该 Summary 选项卡

7. 可选： 在“**日志文件信息**”页面上单击**单击测试日志设置**，以启动并测试数据源。 . 在选择日志源选项之后，请单击**测试日志文件设置**。在测试日志文件数据源和提供日志文件内容时，Agent Builder 会基于解析日志的结果自动在组中创建特性。有关测试的更多信息，请参阅第 1068 页的『测试日志文件属性组』。
8. 如果之前并未使用测试功能，并且在“**日志文件信息**”页面上的**日志文件信息**区域中输入了日志文件名，请使用以下步骤：

a) 单击**下一步**以显示“**特性信息**”页面，并定义属性组中的第一个特性。

b) 在“**特性信息**”页面上指定信息，然后单击**完成**。

注: 在缺省的 Tivoli Monitoring 最低版本 (6.2.1) 或更高版本中向代理程序添加了日志文件属性组时，将包含日志文件状态属性组。有关日志文件状态属性组的更多信息，请参阅第 1235 页的『“日志文件状态”属性组』。

日志文件数据源的“**特性信息**”页面不仅包含适用于所有数据源的字段，还在**记录字段信息**区域中包含一些附加字段。

记录字段信息字段包括：

下一个字段
显示使用来自属性组的定界符（或来自 Advanced 对话框的此特性的特殊定界符）解析之后的下一个字段。

记录的其余部分
显示在解析先前的特性之后记录的其余部分。除了可能存在的日志文件名或日志文件标签外，该特性是最后一个特性。

整个记录
显示整条记录，除了可能存在的日志文件名或日志文件标签外，这可能是唯一的特性。

日志文件名
显示日志文件的名称。

日志文件标签
在高级面板上显示指定给文件的标签。

注: 仅当需要派生特性，而不需要直接来自日志文件的特性时使用**派生特性详细信息**选项卡。

9. 单击**记录字段信息**区域中的**高级**，以显示“**高级日志文件特性信息**”页面。
   a) 在**特性过滤器**部分中，指定包括或排除数据的条件。
   对特性进行过滤可以减少处理的数据量，从而提高解决方案的性能。单击以下一个或多个特性过滤器:
   - **包括**：指示特性过滤器集是一个接受过滤器，即表示如果过滤成功，记录将通过过滤器，并将其输出。
   - **排除**：指示特性过滤器集是一个拒绝过滤器，即表示如果特性过滤成功，那么将拒绝记录，并且不会输出此记录。
   - **与所有过滤器匹配**：指示定义到过滤器的所有过滤器必须都与特性记录匹配，过滤才能成功。
   - **与任何过滤器匹配**：指示如果定义到过滤器的任何过滤器与特性记录匹配，即过滤成功。

   b) 使用**添加**、**编辑**和**移除**为特性过滤器集定义各个过滤器。
   c) 要添加过滤器，请执行以下步骤:
      1) 单击**添加**，并按如下所示填写“**添加过滤器**”窗口中的选项:
         a) **过滤条件**部分定义过滤器的基本特征，包括以下属性:
            - **开始偏移量**：定义特性字符串中比较的起始位置。
            - **比较字符串**：定义作为特性定义依据的模式字符串。

              输入代理程序用来过滤从文件中读取的数据的字符串、模式或正则表达式。与过滤模式匹配的记录将从返回到监视环境的记录中移除，或者它们是唯一返回的记录。结果取决于您选择的是 Inclusive 还是 Exclusive 过滤器。
            - **匹配整个值**：检查特性字符串中与比较字符串完全一致的匹配项。从起始偏移位置开始查找。
            - **匹配值的任何部分**：在特性字符串中的任意位置检查是否存在比较字符串。从起始偏移位置开始查找。

         b) **比较字符串是正则表达式**：指示比较字符串是可以针对特性字符串应用的正则表达式模式。

            通过使用 International Components for Unicode (ICU) 库提供正则表达式过滤支持，以确定检查的特性值是否与指定模式匹配。

            要有效使用正则表达式支持，必须熟悉 ICU 实现正则表达式的具体方法。此实现方法与 `Perl`、`grep`、`sed` 和 Java 正则表达式中的正则表达式支持的实现方法以及其他实现方法不同。请参阅 第 1259 页的『ICU 正则表达式』 以了解创建正则表达式过滤器的相关准则。

         c) **定义覆盖过滤器**：指示您要提供更具体的过滤器比较以覆盖先前定义的基本特征。此额外比较字符串用于逆转过滤结果。如果是**包括**类型的过滤器，那么覆盖会充当过滤表达式的排除限定词。如果是**排除**类型的过滤器，那么覆盖会充当过滤表达式的包含限定词。（有关**包括**和**排除**的更多信息，请参阅步骤 第 1064 页的『9』 以及随附示例）。覆盖过滤器具有以下属性:
            - **开始偏移量**：定义特性字符串中比较的起始位置。
            - **比较字符串**：定义作为特性匹配依据的模式字符串。

              输入代理程序用来过滤从文件中读取的数据的正则表达式。与过滤模式匹配的记录将从返回到监视环境的记录中移除，或者它们是唯一返回的记录。结果取决于您选择的是 Inclusive 还是 Exclusive 过滤器。

         d) **替换值**可用于将原始特性字符串变更为新值。请参阅 第 1259 页的『ICU 正则表达式』 以了解有关可使用的特殊字符的更多详细信息。

e) **替换第一个实例**将比较字符串匹配的第一个匹配项替换为新文本。

f) **替换所有实例**将比较字符串匹配的所有匹配项替换为新文本。

2) 单击**确定**。

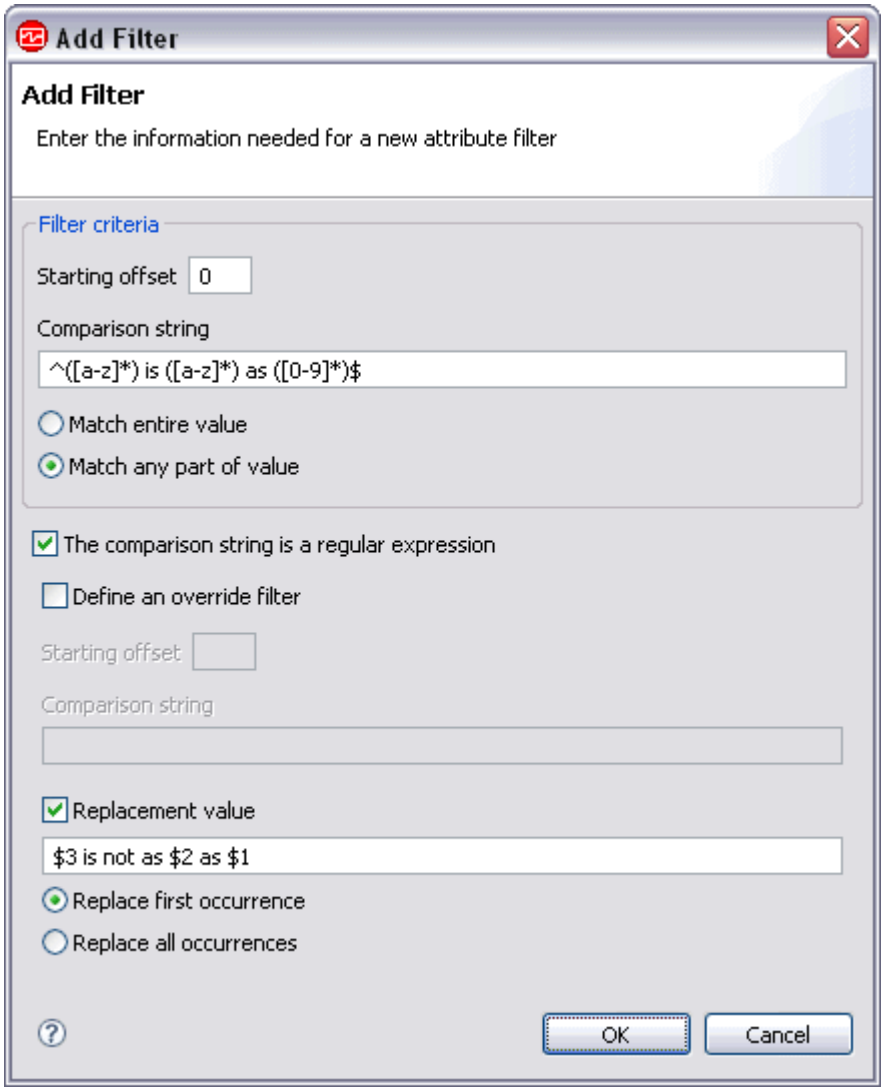


图 *38.* “**添加过滤器**”示例 *1*

如果特性字符串是 `abc is easy as 123`，那么在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示的已被替换的字符串是 `123 is not as easy as abc`。

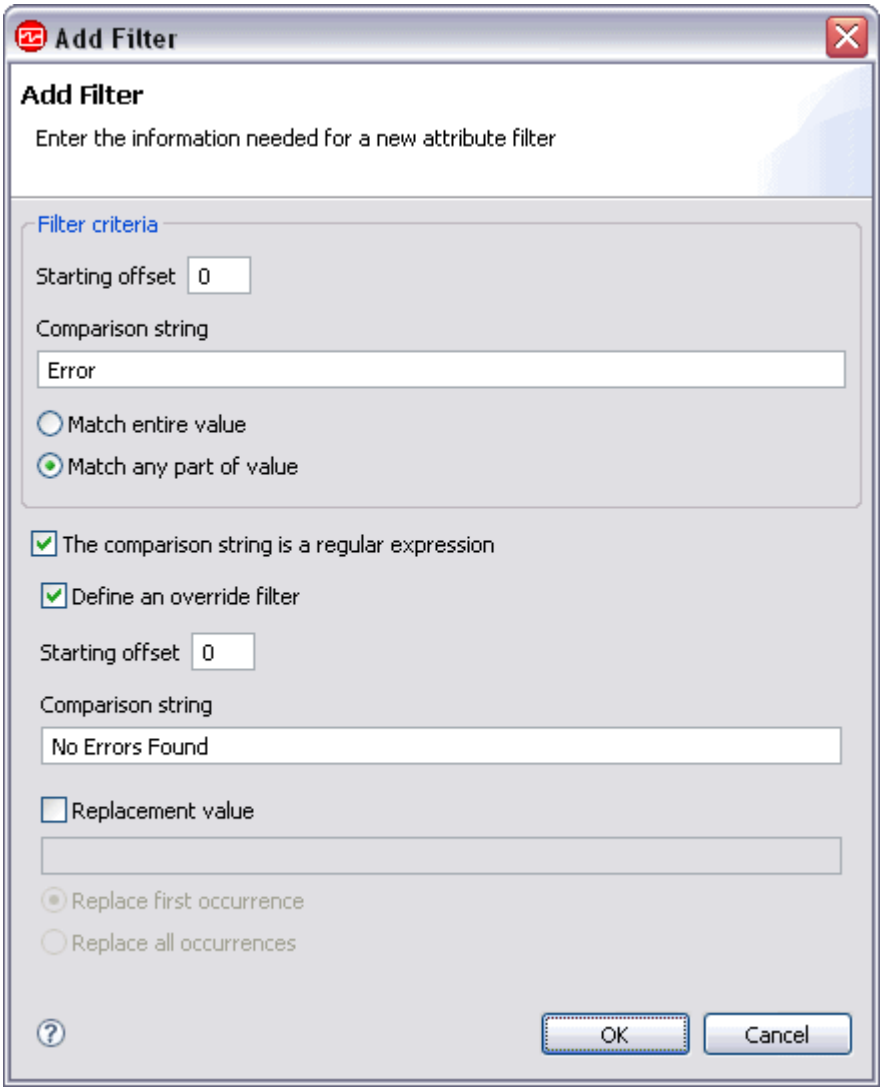


图 *39.* “**添加过滤器**”示例 *2*

如果特性字符串是 `Unrecoverable Error reading from disk`，并且使用**包括**类型的过滤器，那么该特性将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中。如果特性字符串是 `No Errors Found during weekly backup`，并且使用**包括**类型的过滤器，那么不会显示此特性。

d) 在“**高级日志文件特性信息**”页面的**字段标识**部分中，指定如何仅针对这一个特性覆盖属性组字段分隔符。单击某个特性过滤器，并填写该选项的必填字段：

- **字符数**：输入字符数限制。
- **制表符分隔符**：指定使用制表符分隔符。
- **分隔符文本**：输入要使用的分隔符文本。
- **开始及结束文本**：请同时输入**开始**文本和**结束**文本。

e) 在“**高级日志文件特性信息**”页面的**摘要**部分中，单击**将特性包括在摘要属性组中**复选框以将特性添加到摘要属性组。

当用户开启了日志特性摘要时将生成此属性组。

f) 单击**确定**。

10. 如果在步骤 第 1063 页的『7』 中使用了测试功能，那么将显示“**选择键特性**”页面。在“**选择键属性**”页面上，选择键属性或指示此数据源仅生成一个数据行。

    有关更多信息，请参阅第 990 页的『选择键属性』。

11. 执行下列其中一个步骤：

- 如果您正在使用“新建代理程序”向导，请单击**下一步**。
- 单击**完成**，以保存该数据源并打开 Agent Editor。

注: 在缺省的 Tivoli Monitoring 最低版本 (6.2.1) 或更高版本中向代理程序添加了日志文件属性组时，将包含日志文件状态属性组。有关日志文件状态属性组的更多信息，请参阅第 1235 页的『“日志文件状态”属性组』。

日志文件解析和分隔符

您可以更改在日志文件记录中用于分隔一个或多个属性的缺省分隔符。

在创建日志文件属性组时，缺省情况下会分配一个分隔符。缺省分隔符是制表符。代理程序使用该分隔符对数据行中每个属性的数据进行解析和定界。您可以将缺省属性分隔符更改为:

- 固定数量的字符
- 空格
- 其他一个或多个字符
- 特定的开始和结束文本
- XML 元素。

您可以通过以下方法更改用于组中所有属性的缺省分隔符:

1. 在创建属性组时，在“**日志文件信息**”页面中更改。
2. 创建属性组之后，可通过打开 **Agent Editor** > **数据源**选项卡，选择属性组，然后在“**字段标识**”区域中选择分隔符。

您还可以选择为一个或多个个别属性指定特定分隔符。您可以将个别属性的特定分隔符指定为使用:

- 固定数量的字符。
- 制表符分隔符
- 空格分隔符
- 其他一个或多个字符
- 特定的开始和结束文本。

您可以通过以下方法更改用于个别属性的分隔符:

1. 创建属性时，在“**属性信息**”页面上选择**高级**进行更改。
2. 通过打开 **Agent Editor** > **数据源**选项卡，选择属性，并在“**日志文件属性信息**”选项卡上选择**高级**。

示例 1 - 简单日志文件输出

某些日志文件记录具有明确的常规分隔符，例如:

```
one,two,three
```

此处的“,”字符就是明确的常规分隔符，位于这一行的三段数据之间。在此情况下，选择**分隔符文本**，并指定“,”作为属性组的缺省分隔符。无需更改或定义其他分隔符。

以下输出显示了如何为日志文件定义此分隔符，此日志文件包含本示例之前显示的数据行:

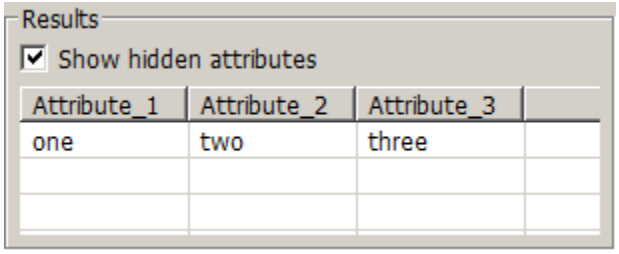


图 *40.* 代理程序解析简单日志文件数据行时的示例属性值输出。

示例 2 - 复杂日志文件输出

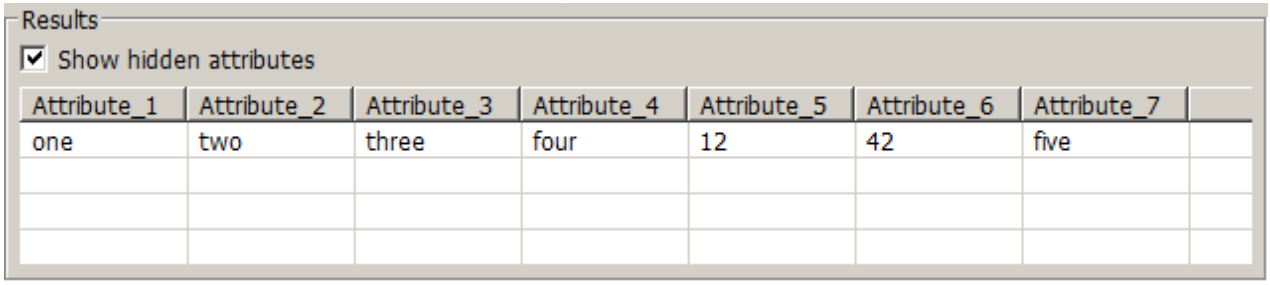
有些日志文件可能包含使用非常规或会变化的分隔符的数据行，例如：

```
one,two,three,[four]12:42,five
```

在此示例中，可以用于为属性定义指定分隔符的过程如下：

1. 在上一个示例中，将缺省分隔符设置为“,”。此分隔符用于所有属性，除非您使用特定分隔符将其覆盖。在此示例中，仍然使用缺省分隔符“,”来分隔此行中的前三个属性是正确的。
2. 对于第四个属性，假设“[”和“]”之间的字符串是您要抽取的值。在此情况下，在定义第四个属性时，为分隔符类型**开始和结束文本**指定开始和结束文本值“[”和“]”。
3. 对于第五个属性，假设您要抽取“]”和“:”字符之间的值。在此情况下，在定义第五个属性时，将分隔符类型**分隔符文本**设置为“:”。
4. 对于第六个属性，又可以使用缺省属性组分隔符“,”。
5. 对于第七个属性，无需指定分隔符，因为这是最后一个属性。

以下输出显示了如何为日志文件定义这些分隔符，此日志文件包含本示例之前显示的数据行：


Results

Show hidden attributes

Attribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5	Attribute_6	Attribute_7
one	two	three	four	12	42	five

图 *41.* 代理程序解析复杂日志文件数据行时的示例属性值输出。

第 1060 页的『监视日志文件』步骤 第 1060 页的『5』 中描述了如何定义属性分隔符的过程。

测试日志文件属性组

您可以使用 Agent Builder 测试所创建的日志文件数据集（属性组）。如果没有为组定义任何属性，测试过程将自动进行定义。

开始之前

如果已为该数据集定义了任何属性，并想要在测试期间自动定义属性，请使用代理程序编辑器从数据集中移除所有现有的属性。有关指示信息，请参阅第 1007 页的『移除属性』。

过程

1. 可以通过以下方法启动测试过程：
   - 代理程序创建期间，在“**日志文件信息**”页面上单击**测试日志文件设置**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试日志文件设置**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试日志文件设置**之后，将打开“**解析日志**”窗口。
2. 选择要测试的日志数据的源：
   - **使用属性组测试**：使用数据源中指定的文件名和位置。缺省情况下，数据源仅处理启动测试过程后添加到日志文件的信息。如果日志文件实时更新，您可以使用此选项。
   - **指定样本文件**：提供样本日志文件。通过此设置，测试过程将解析日志文件的整个内容。通过此选项，您可以根据现有样本测试数据源并立即为其创建属性。在**日志文件名**字段中指定文件的路径和名称，或者使用**浏览**按钮来选择文件。
3. 可选： 在开始测试之前，您可以设置环境变量和配置属性。

   有关更多信息，请参阅第 1156 页的『属性组测试』。

4. 单击**启动代理程序**。

   此时会打开一个窗口，指示代理程序正在启动。当代理程序启动时，它会监视已配置的日志文件中是否有新记录

5. 要测试代理程序数据收集，请在受监视日志文件中生成新记录。

   在将新记录添加到日志文件时，代理程序将根据其配置解析这些记录，并更新其高速缓存中对应的属性值。

6. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。

   “**解析日志**”窗口将收集和显示自代理程序上次启动以来高速缓存中的任何新属性值。第 1069 页的图 42 中显示了示例数据收集

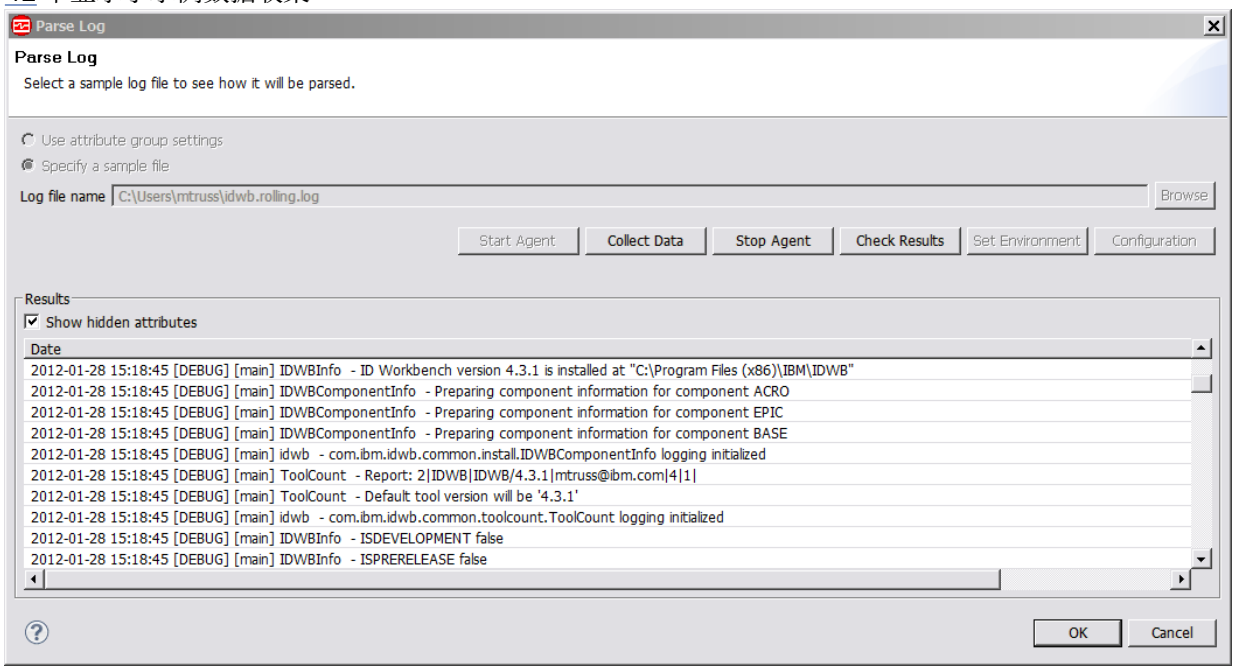


图 *42.* 显示已解析的日志文件属性值的“解析日志”窗口

7. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据

8. 通过单击**停止代理程序**可停止代理程序。
9. 单击**确定**或**取消**以退出“**解析日志**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视 AIX 二进制日志

您可以定义数据源以通过 `errpt` 命令监视 AIX 二进制错误日志。您还可以对其进行配置以过滤和汇总数据。产生的事件放置在数据集中。

关于此任务

日志监视功能支持通过 `errpt` 命令监视 AIX 二进制错误日志。`errpt` 命令根据错误日志中的条目生成错误报告。它提供了用于选择与特定条件相匹配的错误的标志。对通过 `errpt` 命令监视 AIX 二进制错误日志的此支持，是以 Tivoli Monitoring UNIX Logs Agent（产品代码为 `kul` 或 `ul`）中的同一功能支持为基础而设计。

向 Agent Builder 提供 **`errpt`** 命令字符串后，它将处理由于运行此命令而发生的事件。Agent Builder 将对此命令实施 Monitoring Agent for UNIX Logs 所实施的约束。尤其是，必须使用 **`-c`**（并发方式）选项以使此命令持续运行，并且不得使用 **`-t`** 选项或下列将产生详细输出的选项：**`-a`**、**`-A`** 或 **`-g`**。

监视 AIX **errpt** 命令的 Agent Builder 代理程序将自动包括 Monitoring Agent for UNIX Logs 所包括的信息。有关 AIX 二进制错误日志的属性组的更多信息，请参阅第 1208 页的『“AIX 二进制日志”属性组』。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**记录的数据**。
2. 在**数据源**区域中，单击 **AIX 二进制日志**。
3. 单击**下一步**。
4. 在“**二进制日志信息**”页面中，输入 `errpt` 命令。

   缺省值为：

   ```
   errpt -c -smmddhhmmyy
   ```

   启动时，代理程序将搜索“mmddhhmmyy”字符串并将其替换为实际日期和时间。将只替换此字符串的第一个实例。

   您可以提供自己的 `errpt` 命令，但 Agent Builder 将对此命令实施 Monitoring Agent for UNIX Logs 所实施的约束。尤其是，必须使用 **-c**（并发方式）选项以使此命令持续运行，并且不得使用 **-t** 选项或下列将产生详细输出的选项：**-a**、**-A** 或 **-g**。
5. （可选）单击**高级**以选择事件的过滤及摘要选项。有关更多信息，请参阅第 1181 页的『控制重复事件』。
6. 执行下列其中一个步骤：
   - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
   - 单击**完成**，以保存该数据源并打开 Agent Editor。

相关参考

第 1208 页的『“AIX 二进制日志”属性组』
“AIX 二进制日志”属性组显示来自 AIX 二进制日志的事件，这些事件由提供的 `errpt` 命令字符串选择。

监视 Windows 事件日志

您可以定义数据源以从 Windows 事件日志收集数据。您可以对其进行配置以过滤数据。产生的事件放置在“事件日志”数据集中。

关于此任务

您可以使用事件类型、源或标识从 Windows 事件日志收集数据。您可以使用这些参数来过滤 Windows 系统收集的日志事件。代理程序将受监视事件日志中的每个新事件与指定的过滤器进行比较。如果事件与过滤器中指定的某个事件类型、事件源以及事件标识相匹配，那么该事件将通过。

例如，如果事件日志过滤器针对的是应用程序日志，请指定**错误**作为事件类型。此选项与记录到应用程序日志并且事件类型值为 `error` 的所有事件匹配。如果添加 **Diskeeper** 和 **Symantec AntiVirus** 事件源，那么代理程序将与来自其中任一源的所有错误事件匹配。可添加特定事件标识来进一步优化过滤器。事件类型、事件源和事件标识之间不存在直接关联。如果其中某个值各自与事件相匹配，事件即匹配。

缺省情况下，仅处理代理程序启动之后生成的事件。但是，您可以允许代理程序在重新启动时处理其关闭期间生成的日志事件。有关使代理程序能够处理在代理程序关闭期间生成的事件的更多信息，请参阅步骤 第 1071 页的『6』。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**记录的数据**。
2. 在**数据源**区域中，单击 **Windows 事件日志**。
3. 单击**下一步**。
4. 在“**Windows 事件日志**”页面上，从 **Windows 事件日志名称**列表中选择其中一个日志的名称，或者为事件日志输入名称。

   该列表由当前系统上的日志集合构成，例如：

应用程序
安全性
系统

5. 在“**Windows 事件日志**”页面中，指定您是要使用以下其中一种还是多种机制来过滤结果：
   - 第 1071 页的『按事件类型过滤』
   - 第 1072 页的『按事件源过滤』
   - 第 1072 页的『按事件标识过滤』

   注: 必须至少选择其中一个过滤条件。
6. 要处理在代理程序关闭期间生成的日志事件，请在代理程序重新启动时，单击“**Windows 事件日志**”页面上的 **脱机事件设置**。

   “**Windows 事件日志书签设置**”窗口随即打开。
7. 选择下列其中一个书签选项：

   注: 下列选项将应用于所有受监视的 Windows 事件日志。
   - **不收集任何脱机事件**：不处理在代理程序关闭期间生成的事件。此选项为缺省选项。
   - **收集所有脱机对象**：处理在代理程序关闭期间生成的所有事件。
   - **指定定制收集设置**：您可以输入值以根据时间值和/或事件数量来调整旧事件的处理速度。通过使用此选项，可确保在代理程序启动时，监视环境不会因为事件过多而超负荷。

     例如，如果在**要收集的最大事件数**字段中输入了 100，在**根据时间间隔限制收集（秒）**字段中输入了 30。那么处理的事件数量为代理程序启动之前生成的最后 100 个事件，或者是代理程序启动 30 秒内生成的任何事件。其结果取决于首个匹配的变量。

     如果输入了要收集的最大事件数的值，那么将添加 *CDP_DP_EVENT_LOG_MAX_BACKLOG_EVENTS* 环境变量。在输入值以根据时间间隔限制收集时，会添加 *CDP_DP_EVENT_LOG_MAX_BACKLOG_TIME* 环境变量。添加任意一个或同时添加这两个变量时，会创建 *eventlogname_productcode_instancename_subnodename*.rst 文件，其中包含为事件日志处理的最后一个事件记录。此文件位于 *%CANDLE_HOME%*\tmaitm6\logs 目录中，并在重新启动代理程序以处理代理程序关闭期间生成的旧事件时使用。
8. 如果要为数据源设置全局选项，请单击“**Windows 事件日志**”页面上的**全局选项**。

   “**全局 Windows 数据源选项**”窗口随即打开。
9. 如果要包括此选项，请选中**包括远程 Windows 配置属性**复选框，并单击**确定**。

   有关 Windows 数据源的 Windows 远程连接配置的信息，请参阅第 1145 页的『配置 Windows 远程连接』。
10. 在指定过滤器并单击**确定**之后，请在“**Windows 事件日志**”页面上执行下列其中一个步骤：
    - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
    - 单击**完成**，以保存该数据源并打开 Agent Editor。新 Windows 事件日志的名称将显示在 **Agent Editor** 的“**数据源定义**”页面上。

下一步做什么

有关 Windows 事件日志数据源的 Windows 远程连接配置的信息，请参阅第 1145 页的『配置 Windows 远程连接』。

按事件类型过滤

按事件类型过滤 Windows 事件日志结果

过程

1. 在“**Windows 事件日志**”页面中，选择**按事件类型过滤**。
2. 选择以下一个或多个事件类型：
   - **参考**

- **警告**
- **错误**
- **成功审计**
- **失败审计**

3. 单击**完成**以完成操作。

按事件源过滤
按事件源过滤 Windows 事件日志结果

过程

1. 选择**按事件源过滤**，并在“**Windows 事件日志**”页面的**事件源**区域中单击**添加**。
   “**事件源**”窗口随即打开。
2. 选择执行下列操作之一。
   - 输入事件源名称，然后单击**确定**。
   - 单击**浏览**以查找并从列表中选择事件源，然后单击**确定**。

     您选择的名称会显示在“**事件源**”窗口中。

     注:

     a. 要对事件源列表进行排序，请单击列标题。
     b. 要刷新窗口中的信息，单击**刷新**图标。
     c. 要搜索特定事件源，单击**搜索**（望远镜）图标。

3. 单击**确定**后，您将在“**Windows 事件日志**”窗口的“事件源”列表中看到这个新增的事件源过滤器。

按事件标识过滤
对于 Windows 事件日志数据源，您可以按事件标识过滤事件。

关于此任务

要按事件标识进行过滤，请使用以下过程：

过程

1. 选择**按事件标识过滤**，并在“**Windows 事件日志**”窗口的**事件标识**区域中单击**添加**。
   这将显示“**事件标识**”窗口。
2. 如果您要监视来自应用程序的特定事件，请在应用程序定义事件时指定事件的编号。输入整数以作为事件标识，然后单击**确定**。
   此时，该新事件过滤器的数字标识显示在“**Windows 事件日志**”的“事件标识”列表中。

   注: 必须逐个定义每个事件标识。
3. 如果要修改某个 Windows 事件日志，请将其选中，然后单击**编辑**。
4. 如果要删除某个 Windows 事件日志，请将其选中，然后单击**移除**。
5. 您可以继续向列表添加事件日志，也可以单击**完成**。

监视命令返回码

您可以定义数据源以使用命令返回码来监视应用程序或系统。代理程序运行命令，收集返回码，并将结果添加到“可用性”数据集。

关于此任务

用户创建的脚本、可执行文件、查询或系统命令能够返回代码。命令返回码是用于确定应用程序或受监视的系统是否可用的特定于应用程序的机制。代理程序将运行指定的命令并通过检查返回码确定应用程序或受监视的系统的状态。

命令必须对每种描述性状态都显示一个唯一返回码。命令还必须对其中每个返回码都定义要由代理程序使用的消息。命令可以在用户创建的脚本、可执行文件、查询或系统命令内使用环境和配置变量。以命令行方式调用命令时，该命令不能使用环境或配置变量，仅下列变量除外：*AGENT_BIN_DIR*、*AGENT_ETC_DIR*、*AGENT_LIB_DIR*、*CANDLE_HOME* 和 *CANDLEHOME*。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，选择**监视数据类别**区域中的**命令或脚本**。
2. 在**数据源**区域中，单击**命令返回码**。
3. 单击**下一步**。
4. 在“**命令返回码**”页面的**命令返回码信息**区域中，输入显示名。
5. 使用以下子步骤来定义和描述您希望命令返回码使用的命令行。

   注: 请为代理程序支持的每一款操作系统定义一个命令。这些命令可以共享，但所有命令对应的操作系统的总集合必须与代理程序支持的操作系统集合相同。

   a) 在“**命令返回码**”窗口的**命令**区域中单击**添加**以打开“**命令信息**”窗口。

   b) 输入命令行，并从“**命令信息**”窗口的**操作系统**区域中的列表中选择操作系统。

      注:

      1) 对于 Windows 命令，必须输入命令的全名。例如，`command_to_run.bat`，而不仅仅是 `command_to_run`。

      2) 将名称加上引号，以免命令解释器对其进行语法分析。例如，请输入 “`this is a test.bat`” `argument`，而不要输入 `this is a test.bat argument`。

      3) 您可以单击命令，然后单击**编辑**对其进行修改，或者单击**删除**将其删除。

   c) 在“**命令信息**”窗口的 **Return Codes** 区域中，单击**添加**。

   d) 在“**返回码定义**”窗口中，从显示的列表中选择返回码类型。

      您可以指定下列状态以测试返回码:

      - `ALREADY_RUNNING`
      - `DEPENDENT_NOT_RUNNING`
      - `GENERAL_ERROR`
      - `NOT_RUNNING`
      - `OK`
      - `PREREQ_NOT_RUNNING`
      - `WARNING`

   e) 为选择的返回码类型输入数字值。

      返回码的值是一个整数，用于指定针对命令返回码定义的返回码。为了能够在操作系统之间进行移植，请使用 `0 - 255` 范围内的返回码值。对于仅在 Windows 上运行的命令，返回码值可以从 `-2147483648 - 2147483647`。

   f) 请为每个返回码定义一条消息，以便可以同时显示消息和代码。单击**浏览**以设置消息文本。

      消息窗口列出代理程序中定义的消息。“**消息**”（列表）窗口将打开。

      注:

      1) 通过在消息文本列表中进行选择，即可选择先前输入的文本，而无需单击**浏览**。然后，转到步骤 5k。

      2) 在定义消息之前，列表保留为空。您可以使用**编辑**来变更已定义的消息，并可以使用**移除**来删除一条或多条已定义的消息。

   g) 在“**消息**”（列表）窗口中，单击**添加**。

   “**消息定义**”窗口随即打开。

注: 系统会为您自动生成消息标识。

h) 在**消息文本**字段中输入一些文本，用于描述新消息的含义。

i) 单击**确定**。

“**消息**”（列表）窗口将打开，并显示新消息。

j) 要验证该消息并使其永久化，请从列表中选择该消息，然后单击**确定**。

新的返回码类型、值和文本将显示在“**返回码定义**”窗口中。

k) 如果您希望此返回码可用于此命令返回码的其他操作系统上的其他命令，请选中**全局返回码适用于所有命令**。如果您想要让此返回码仅可用于此命令，请将**本地返回码仅适用于此命令**保留为选中状态。

l) 在“**返回码定义**”窗口中，单击**窗口**。

m) 在离开“**命令信息**”窗口之前，请至少定义两个返回码。一个返回码用于指示没有可用性问题，另一个用于指示是否发生问题。如果要添加另一个返回码，请返回到步骤 c。

n) 可选： 如果您想选择一个或多个脚本或可执行文件以供代理程序运行，请在“**命令信息**”窗口的**命令文件**区域中单击**添加**。

一个或多个文件将复制到代理程序的项目文件夹的 `scripts/`*`operating system`* 之下，其中 *operating system* 是一个变量，它依赖于您在“**命令信息**”窗口的**操作系统**区域中选择的内容。这些文件也与代理程序一起打包和分发。要编辑现有命令文件的定义，或者编辑已复制至项目的原始命令文件，请选中该文件，然后单击**编辑**。请参阅第 1075 页的『编辑命令文件定义』。

o) 在“**命令信息**”窗口中，单击**确定**。

注: 命令文件表是定义要包含在代理程序包中的所有外部文件的位置。这些文件将复制到项目目录中，并随代理程序一起打包，以进行分发。

6. 如果存在其他尚未定义的返回码，请定义并描述命令返回码可以使用的全局返回码。

a) 在“**命令返回码**”页面的“**全局返回码**”区域中，单击**添加**。

注: 此处定义的返回码是全局返回码。这表示这些返回码适用于所有针对命令返回码定义的命令。（这些返回码不能在命令返回码之间共享。） 另外，您可以在输入命令信息时定义返回码。在这里定义的返回码可以是全局返回码或局部返回码。本地返回码仅适用于此特定命令。如果具有跨所有操作系统都相同的返回码，则该层次结构很有用。（例如，返回码 0 表示一切正常。您可以在全局级别对它进行定义，然后所有已定义命令都按这种方式来解释 0。） 如果其他所有操作系统都未返回 5，那么可以仅为 Windows 命令定义返回码 5。如果在本地命令级别定义的返回码与已在全局级别定义的返回码相同，那么使用该命令级别。您可以使用此方法来覆盖特定操作系统上的返回码。例如，即使返回码 2 在所有 UNIX 操作系统上都表示某个含义，但是在 Windows 上，它表示另一种含义。您可以在全局级别按 UNIX 操作系统的需要定义返回码 2。然后，在 Windows 的命令中，可以针对 Windows 上的含义重新定义返回码 2。

b) 在“**返回码定义**”窗口中，从显示的列表中选择返回码类型。

您可以指定下列状态以测试返回码：

- `ALREADY_RUNNING`
- `DEPENDENT_NOT_RUNNING`
- `GENERAL_ERROR`
- `NOT_RUNNING`
- `OK`
- `PREREQ_NOT_RUNNING`
- `WARNING`

c) 为选择的返回码类型输入数字值。返回码的值是一个整数，用于指定针对命令返回码定义的返回码。

d) 单击**浏览**以设置消息文本及其相关含义。必须为每个返回码定义一条消息，以便可以同时显示消息和代码。

“**消息**”窗口列出代理程序中定义的消息。

注:

1) 在定义消息之前，列表保留为空。您可以使用**编辑**来变更已定义的消息，并可以使用**移除**来删除一条或多条已定义的消息。

2) 通过在**消息文本**列表中进行选择，即可选择先前输入的文本，而无需单击**浏览**。然后，转到步骤 6h。

e) 在“**消息**”（列表）窗口中，单击**添加**以查看“**消息定义**”窗口，您可以在该窗口中输入文本以描述新消息的含义。

f) 单击**确定**。

g) “**消息**”（列表）窗口将打开，并显示新消息。要验证该消息并使其永久化，请从列表中选择该消息，然后单击**确定**。

h) 当新的文本、类型和值显示在“**返回码定义**”窗口中之后，单击**确定**。

i) 在“命令返回码”页面上，当您完成针对所有受支持操作系统定义返回码及命令时，执行下列其中一个步骤：

- 如果您正在使用“新建代理程序”向导，请单击**下一步**，或者单击**完成**以保存该数据源并打开 Agent Editor。
- 如果您正在使用“新建代理程序组件”向导，请单击**完成**以返回到 Agent Editor。

下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据，必须创建基于“可用性”数据集过滤的数据集（属性组），并将其配置为提供单行。使用“名称”字段来选择用于您的进程的行。

在新的过滤属性组中，选择“状态”字段并为其指定严重性值。

有关指示信息，请参阅：

- 第 1125 页的『创建过滤属性组』
- 第 1010 页的『指定用作状态指示符的特性的严重性』
- 第 1154 页的『为 Cloud APM 准备代理程序』

编辑命令文件定义

您可以对导入到项目中的命令文件进行更改，也可以将针对现有命令文件的更改导入到项目中。

过程

1. 在“**命令信息**”窗口的“**命令文件**”区域中选择文件。
2. 单击**编辑**以打开“**导入命令文件**”窗口。

   从“**导入命令文件**”窗口中，您可以了解该命令文件的状态。您还可以更改初始源文件的位置，并将源文件重新复制到代理程序中。
3. 选择下列某个步骤：
   - 单击**确定**，以将文件复制安排在下一次保存代理程序时执行。
   - 单击**立即复制**以复制该文件，而不首先保存代理程序。

   注: 从“新建代理程序”向导中访问“**导入命令文件**”窗口时，**立即复制**选项不可用。

文件拆分与合并

您可以使用“拆分与合并”功能将文件移入和移出代理程序中特定于操作系统的文件夹。

文件首次添加到代理程序时，将在 `scripts/all_windows` 文件夹、`scripts/all_unix` 文件夹或 `scripts/common` 文件夹中添加一个副本。如果在 Windows 和 UNIX 上都使用该文件，那么将使用 `scripts/common` 文件夹。

要在不同操作系统上放置此文件的不同副本（例如，二进制可执行文件），请单击**编辑**，然后单击**分隔**。系统会将该文件从公共文件夹中移除，并将其复制到特定于操作系统的文件夹中。然后，可以将该文件的各个副本替换为适合于特定操作系统的副本。

注: Java 资源文件必须保留在 `scripts/common` 文件夹中。您无法通过单击**分隔**来针对个别操作系统建立 Java 资源文件的单独副本。

如果您已将文件分隔到操作系统文件夹中，那么可以使用**合并**将其移回到公共文件夹。如果您是在不支持公共文件夹的 Agent Builder 版本中创建代理程序，请使用**合并**将这些文件移回到公共文件夹。如果该文件的任何副本与其他副本不同，系统会提示您选择用作公共文件的文件。将废弃所有其他副本。

监视脚本的输出

您可以定义数据源以从脚本或外部程序收集数据。当通过标准管理界面无法使用应用程序数据或需要在单个行中提供多行数据摘要时，使用该数据源。代理程序运行脚本并收集其输出。脚本输出中的每一行都会解析为所产生数据集中的一行。

可以从本地或远程系统收集数据。脚本或程序输出只能包含属性组中每个属性的值。要返回多行数据，必须使用换行符来分隔各行数据。每个数据行中的属性都由您定义的分隔符分隔。有关分隔符的更多信息，请参阅第 1076 页的『脚本解析和分隔符』

此命令可以在用户创建的脚本、可执行文件、查询或系统命令中使用环境变量和配置变量。以命令行方式调用此命令时，此命令无法使用除下列变量以外的环境变量或配置变量：`AGENT_BIN_DIR`、`AGENT_ETC_DIR`、`AGENT_LIB_DIR`、`CANDLE_HOME` 和 `CANDLEHOME`。

代理程序将监视以代理程序运行环境的语言环境和代码页写入的脚本输出。

从远程系统收集脚本数据

为了从远程系统收集脚本或程序数据，Agent Builder 使用 Secure Shell (SSH)

为了从远程系统收集数据，Agent Builder 会创建一个 Secure Shell (SSH) 会话，并在远程系统上启动脚本或外部程序。代理程序建立并登录到 SSH 会话。然后，代理程序将脚本上载到远程系统，启动脚本或外部程序，并检索输出。可以对代理程序进行配置，以便该会话保持打开状态或者为每个调用重新建立该会话。如果该会话保持打开状态，那么可以对每个调用重复使用或上载脚本。在缺省情况下，将使用单个 SSH 会话并对每个调用重复使用脚本。

Agent Builder 仅支持将 SSH 协议 V2 与 Rivest、Shamir 和 Adleman (RSA) 或 Digital Signature Algorithm (DSA) 密钥一起使用。该代理程序通过用户名和密码或公用密钥进行认证。公用密钥的生成和分发是一项必须在代理程序和 Agent Builder 外部执行的管理任务。

要在远程系统上针对启用了 Secure Shell (SSH) 的脚本数据提供程序运行所编写的“执行操作”命令，请参阅 第 1273 页的『SSHEXEC 操作』。

限制: 如果您的代理程序是使用 Agent Builder V6.3 之前的版本构建的，并且其脚本数据提供程序使用 SSH，那么在与 IBM Tivoli Monitoring V6.3 或更高版本一起运行时，该提供者将失败。要解决此问题，请使用 Agent Builder 的当前版本重新构建代理程序。

存在此限制是因为 IBM Tivoli Monitoring V6.3 使用较新版本的全局安全工具箱 (GSKit) API。您必须使用 Agent Builder 6.3 或更高版本来重新构建代理程序，才能将其与 IBM Tivoli Monitoring V6.3 或更高版本一起运行。如果使用 Agent Builder 6.3 构建代理程序，那么它也可以与较低版本的 IBM Tivoli Monitoring 一起运行。

脚本解析和分隔符

您可以为一个或多个属性更改和指定特定的脚本分隔符。

在创建脚本属性组时，缺省情况下将指定单字符文本分隔符。该缺省分隔符是“;”。代理程序使用该分隔符对数据行中每个属性的数据进行解析和定界。您可以更改缺省分隔符以使用其他字符。您还可以为一个或多个个别属性指定特定分隔符。

您可以为个别属性指定特定分隔符：

- 使用输出中固定数量的字节。
- 以定制分隔符分隔相邻属性，该分隔符可以由多个字符组成。
- 在属性值的开头和结尾用字符串为该值定界。
- 将其余文本作为属性值返回（无论它是否包含嵌入的分隔符）。

您可以使用一个或多个此类分隔符从数据行中抽取属性值。

示例 1 - 简单脚本输出

有些脚本可以输出带有明确的常规分隔符的数据行，例如：

```
Row One;1;2
Row Two;3;4
Row Three;5;6
```

此处的“;”字符就是明确的常规分隔符，位于每一行的三段数据之间。在此情况下，使用缺省分隔符即可，因此无需更改或定义其他分隔符。不难想象使用其他字符作为分隔符的类似脚本输出，如同以下示例。

```
Row One-1-2
Row Two-3-4
Row Three-5-6
```

在此示例中，分隔符从“;”字符更改为“-”字符。在此情况下，在定义属性时，更改缺省分隔符以使用“-”字符。

示例 2 - 复杂脚本输出

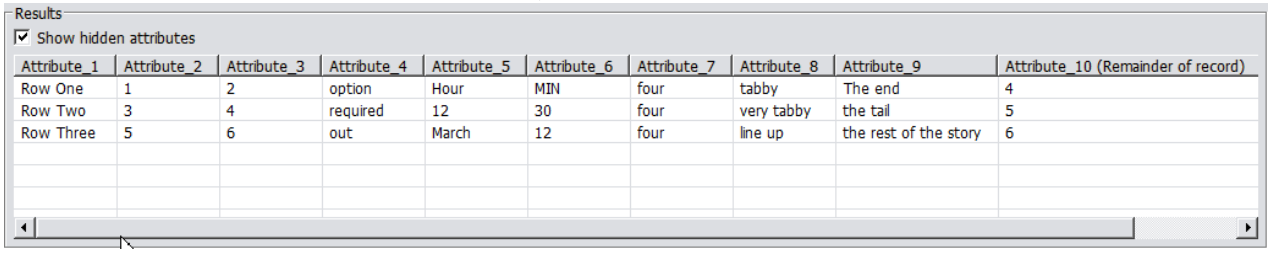
有些脚本可能输出使用非常规或会变化的分隔符的数据行，例如：

```
Row One;1;2;[option]Hour:MIN;fourtabby     The end;4
Row Two;3;4;[required]12:30;fourvery tabby the tail;5
Row Three;5;6;[out]March:12;fourline up    the rest of the story;6
```

在此示例中，可以用于为属性定义指定分隔符的过程如下：

1. 最初可使用缺省分隔符“;”分隔每个数据行中的前三个属性。在此情况下，将分隔符类型**分隔符文本**设置为“;”，这样在定义每个属性时，此设置是缺省设置。
2. 对于第四个属性，假设“[”和“]”之间的字符串是您要抽取的值。在此情况下，在定义第四个属性时，为分隔符类型**开始和结束文本**指定开始和结束文本值“[”和“]”。
3. 对于第五个属性，假设您要抽取“]”和“:”字符之间的值。在此情况下，在定义第五个属性时，将分隔符类型**分隔符文本**设置为“:”。
4. 对于第六个属性，又可以使用缺省分隔符“;”，因此请接受缺省值。
5. 对于第七个属性，假设您要抽取接下来的四个字符组成的字符串“four”。在此字符串结尾没有明确的分隔符。您可以指定一定数量的字符来定义其与下一个属性的分界。您可以分配分隔符类型**字符数**，并指定长度为 4 个字符。
6. 对于第八个属性，您要抽取字符串 tabby、very tabby 和 line up。在这种情况下，您可以假定所有这些字符串后面都跟着一个制表符。此时，分配类型为**制表符分隔符**的分隔符。
7. 对于第九个属性，您应该再次使用缺省分隔符类型，以将剩余文本抽取到此属性。
8. 对于第十个属性，指定**记录的其余部分**以将数据行的剩余部分分配给此属性。

以下输出显示了如何在脚本中定义这些分隔符，此脚本输出本示例之前显示的数据行：


Results

Show hidden attributes

Attribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5	Attribute_6	Attribute_7	Attribute_8	Attribute_9	Attribute_10 (Remainder of record)
Row One	1	2	option	Hour	MIN	four	tabby	The end	4
Row Two	3	4	required	12	30	four	very tabby	the tail	5
Row Three	5	6	out	March	12	four	line up	the rest of the story	6

图 *43.* 代理程序解析复杂脚本输出时的属性值输出示例。

第 1078 页的『监视脚本输出的步骤』步骤 第 1079 页的『10』 中描述了如何定义属性分隔符的过程。

监视脚本输出的步骤

配置代理程序以接收来自脚本数据源的数据。

开始之前

请参阅：第 1076 页的『监视脚本的输出』。

关于此任务

使用以下过程可监视脚本输出：

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，选择**监视数据类别**区域中的**命令或脚本**选项。
2. 在**数据源**区域中，单击**脚本输出**。
3. 单击**下一步**。
4. 在“**命令列表**”页面上，单击**添加**以显示“**命令信息**”窗口。

   注: 选中**允许使用 SSH 进行数据收集**复选框可为此属性组启用 SSH。如果此复选框未处于选中状态，此属性组以本地方式运行。

   注: 如果有可以在 Agent Builder 运行所在的操作系统上运行的命令，将启用**测试**选项。您可以使用**测试**来测试所定义的命令。
5. 在“**命令信息**”窗口的**命令信息**区域中，在**命令**字段中输入命令名和必需自变量，在**分隔符**字段中输入分隔符。

   注:

   a. Windows 中的脚本频繁启动而不在命令行中指定 `.bat` 或 `.cmd` 扩展名。对于远程执行，必须安装 shell 环境，并且必须在脚本数据源命令中指定 `.bat` 或 `.cmd`，这样脚本才能运行。`Cygwin` 就是一个可用于 Windows 的 shell 环境示例。Linux、Red Hat 和 AIX。要验证 shell 环境是否存在，请通过 SSH 访问远程主机或登录远程主机并输入以下命令：

   ```
   PATH=$PATH:. <command>
   ```

   如果此命令运行成功，那么表明存在 shell 环境。

   b. 将名称加上引号，以免命令解释器对其进行语法分析。例如，`this is a test.bat argument` 变成：

   ```
   "this is a test.bat" argument
   ```

   c. 环境变量和配置变量可以在用户提供的脚本中使用，但不能作为启动该脚本的命令行的组成部分。此规则不适用于以下变量：

   AGENT_BIN_DIR
   这是代理程序用于存放二进制文件或脚本的目录

   AGENT_ETC_DIR
   这是代理程序用于存放配置文件的目录

   AGENT_LIB_DIR
   这是代理程序用于存放共享库或动态链接库的目录

   CANDLEHOME
   Linux 或 UNIX Tivoli Monitoring 安装目录

   CANDLE_HOME
   Windows Tivoli Monitoring 安装目录

   d. 如果正在使用 SSH 数据收集选项，将以远程系统用户主目录的相对路径运行命令行。如果您将脚本或可执行文件上载到远程系统，那么它们将复制到代理程序的环境变量 *CDP_SSH_TEMP_DIRECTORY* 中指定的位置。缺省位置为远程系统上的用户主目录。在某些系统上，您可能需要使用相对路径（例如，`./Script.sh`）定义命令行。

6. 在**操作系统**区域中，选择一个或多个操作系统。使用 SSH 从远程系统收集数据时，“操作系统”是安装代理程序的系统的属性。而不是远程系统的“操作系统”。在使用 SSH 数据收集功能时，建议选中**所有操作系统**复选框。
7. 可选： 如果一个或多个用户定义文件是运行命令所必需的，请单击“命令文件”区域中的**添加**以指定系统中的文件。

   文件复制到代理程序项目文件夹的 `scripts/operating system` 下，其中，*operating system* 是一个变量，取决于您在“**命令信息**”窗口中选择的内容。这些文件也与代理程序一起打包和分发。如果要编辑已添加的命令文件定义，或者更改其内容，请选择该文件并单击**编辑**。请参阅第 1075 页的『编辑命令文件定义』。
8. 单击**确定**。“**命令列表**”页面随即显示。
9. 要测试命令，请使用以下步骤:
   a) 单击**测试**以打开命令信息并显示“**测试命令**”窗口。要在远程系统上测试脚本，请从**连接名称**列表中选择系统，或者单击**添加**以添加系统主机名。
   b) 使用“**测试命令**”窗口来更改命令、缺省分隔符和特性分隔符，以及查看这些更改对返回的数据有何影响。
      1) 在字段中输入命令和分隔符（若还未输入）。

         注: 您可以指定其他分隔符，方法是在创建特性时使用“**特性信息**”窗口，或者使用 Agent Editor 来修改现有特性。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』，有关操作数据源和特性的更多信息，请参阅第 1003 页的『编辑数据源和特性属性』
      2) 在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。
      3) 单击**确定**，以返回到“**测试设置**”窗口。
      4) 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
      5) 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。Agent Builder 会运行您的命令。如果已指定远程系统，请提供用户标识和密码。即使返回码不是 0，Agent Builder 仍将以与代理程序相同的方式解析命令的结果。
      6) “**测试设置**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。特性的初始名称是 **Attribute_1 和 Attribute_2**，依此类推；但是，您可以通过单击相应的列标题来修改特性的属性。
      7) 单击**检查结果**可查看命令返回码、未解析的数据以及返回的任何错误消息。
      8) 通过单击**停止代理程序**可停止代理程序。
      9) 单击**确定**，以返回到“**命令信息**”窗口。

         如果更改了命令或分隔符，那么将更新相应命令来反映这些更改。

         如果在创建脚本数据源时打开过此窗口，那么特性已添加到新脚本数据源。

         如果该窗口是从现有脚本数据源打开的，那么对特性的任何更改都会应用于该脚本数据源。将添加任何附加特性，但不会移除任何多余的特性。这些选项仅影响从脚本输出解析的特性。不会影响任何派生的特性。如果其中任何特性根据其引用的特性而变为无效，那么可以手动更新或移除派生的特性。显示的是派生特性公式而不是实际的结果值。

   注: 如果属性组存在，那么请完成以下过程来启动测试
   a. 在 **Agent Editor** 的“**数据源定义**”页面上选择属性组。
   b. 从“命令列表”中选择要测试的脚本
   c. 单击**测试**，并执行步骤 第 1079 页的『9』 的过程。
10. 如果跳过步骤 (第 1079 页的『9』) 中的命令测试，请使用以下步骤:
    a) 在填写了命令信息的“**命令列表**”页面上，单击**下一步**。
    b) 在“**特性信息**”页面上，使用第 1007 页的表 261 来填写特性名称和类型信息。选择**添加其他特性**以添加其他特性。
    c) 在“**特性信息**”页面上，使用“**脚本特性信息**”选项卡为此特性选择特定数据分隔符。

缺省情况下已选择标准分隔符 ;。您可以选择多种其他分隔符，例如字符串、一定数量的字符、制表符或空格。您也可以选择使用分别用于数据开头和结尾的不同字符串分隔符。最后，还可以选择**记录的其余部分**来向特性分配剩余记录。有关脚本解析和分隔符的更多信息，请参阅第 1076 页的『脚本解析和分隔符』。

11. 执行下列其中一个步骤：
    - 如果您正在使用“**代理程序**”向导，请单击**下一步**。
    - 单击**完成**，以保存该数据源并打开 Agent Editor。
12. 您可以添加特性并提供其相关信息。有关更多信息，请参阅第 1004 页的『创建属性』。

    脚本数据源的“**数据源定义**”页面除了包含适用于所有数据源的字段（如第 1007 页的『用于定义特性的字段和选项』所述）以外，还包含以下选项：

    命令列表
    提供对命令和脚本的访问以在数据收集过程中启动。

    添加
    允许用户添加可由此属性组启动的命令。

    编辑
    允许用户编辑现有命令条目。

    移除
    允许用户删除现有命令条目。

    测试
    允许用户访问此属性组的测试环境。

    允许使用 SSH 进行数据收集
    选中此复选框将对此属性组启用 SSH。如果此复选框未处于选中状态，此属性组以本地方式运行。

    有关脚本数据源的 SSH 远程连接配置的信息，请参阅第 1148 页的『配置安全 Shell (SSH) 远程连接』。

监视来自 Java 数据库连接 (JDBC) 的数据

您可以定义数据源以从 JDBC 数据库接收数据。代理程序运行 SQL 查询以从数据库收集数据。查询返回的每个列都将成为所产生数据集中的一个特性。

关于此任务

JDBC 数据提供程序支持以下数据库服务器：

- IBM Db2 9.x 和 8.x
- Microsoft SQL Server 2008、2005 和 2000
- Oracle Database 11g 和 10g

Agent Builder 不包含这些数据库的 JDBC 驱动程序。JDBC 驱动程序是与数据库建立 JDBC 连接所必需的一组 JAR 文件，并由供应商提供。为方便起见，下面提供了到可下载的驱动程序的链接：

- IBM Db2：在数据库服务器安装的 Db2 主安装目录下名为 java 的子目录中包含了 JDBC 驱动程序。
- Microsoft SQL Server Web 站点 (www.microsoft.com)
- Oracle 数据库：Oracle 数据库 JDBC (http://www.oracle.com/technetwork/database/features/jdbc/index.html)

注: 您要记住的一个重要事项是，JDBC 数据提供程序可以远程监视数据库服务器。Java 运行时环境以及所连接数据库服务器的 JDBC 驱动程序 JAR 文件必须位于运行代理程序的系统上。

以下 Java 版本受支持：

- Oracle Corporation Java V5 或更高版本
- IBM Corporation Java V5 或更高版本

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **JDBC**。
3. 单击**下一步**。
4. 在“**JDBC 信息**”页面的 **JDBC 信息**区域中，单击**浏览**以连接至数据库并构建 SQL 查询。

   使用 JDBC 浏览器来连接到数据库并查看其表，以使您能够构建用于收集所需数据的 SQL 查询。当您选择表和列时，系统将生成一个查询，并针对该查询所返回的每个列添加特性。您可以修改和测试所生成的查询，以确保返回所需的数据。

   注: 您也可以手动创建 JDBC 数据源，而无需单击**浏览**。如果您想手动创建数据源，请指定查询并单击**下一步**。您必须针对该查询所返回的每个列，按返回列的顺序定义特性。

   您可以使用 JDBC 数据提供程序来对数据库运行 SQL 查询和存储过程以收集监视数据。在指定 SQL 查询以收集数据时，可以在 SQL 语句中包括 WHERE 子句以便对返回的数据进行过滤。SQL 语句还可以连接多个表中的数据。除了 SQL select 语句之外，JDBC 数据提供程序还可以运行存储过程。有关运行存储过程的信息，请参阅第 1084 页的『存储过程』。
5. Browser 第一次打开时，Java Database Connectivity (JDBC) Browser 窗口将指示未选择任何连接。您必须添加连接。单击**添加**，并执行添加连接的步骤。

   如果您已定义连接，那么将使用该连接，并且您可以前进到步骤 第 1081 页的『6』。

   注: 状态字段显示当前连接的状态。

   请使用以下步骤添加连接:

   a) 在“**JDBC 连接**”页面上，单击 **JDBC 连接**，然后单击**下一步**。

   b) 在“**连接属性**”页面上填写字段，如下所示:

   连接名称
   JDBC 连接的名称。输入此连接的唯一名称。该名称用于引用浏览器中的连接。

   数据库类型
   数据库的类型。选择要连接到的数据库产品。例如，要连接至 IBM Db2 数据库，请选择 **DB2**。

   用户名
   必须定义为至少对数据库具有读访问权，但不必是数据库管理员

   密码
   必须定义为至少对数据库具有读访问权，但不必是数据库管理员

   主机名
   运行数据库服务器的主机的名称。使用 JDBC 可以监视远程数据库，因此您并不限于仅监视本地系统上的数据库。

   端口
   该主机名的主机上数据库服务器正在侦听的端口。

   数据库
   要连接到的数据库的名称。

   Jar 目录
   包含用于连接到数据库的 JDBC JAR 文件的目录。请输入路径名，或者单击**浏览**以查找目录。

   c) 可选： 如果要保存此连接的密码，请选中**将密码保存在 Agent Builder 工作空间中**复选框。

   d) 可选： 如果您希望从这些属性复制此应用程序服务器类型的缺省值，请选中**设置为代理程序配置缺省值**复选框。

   如果您在与受监视系统类似的系统上构建代理程序，最好选中此框。如果未选中此框，配置该代理程序的用户将看到空的字段。然后用户必须确定没有缺省值的所有信息的值。

   e) 单击**测试连接**，以创建一个使用所指定配置参数的数据库连接。

   在“**连接属性**”页面上将显示一条消息，指示连接是否成功。

   f) 当您有了正常工作的连接之后，请单击**完成**。
6. 在在“**Java 数据库连接 (JDBC) 浏览器**”窗口中，将与配置的数据库建立连接。该数据库中包含的表将显示在**数据库表**区域中。请选择数据库表，以便在**选定表中的列**区域中查看该表中包含的列。

注:

a. 单击望远镜图标，以便在**数据库表**列表中搜索表。

b. 缺省情况下将显示所有表。可以通过选择另一个过滤选项对显示的表进行过滤。第 1082 页的表 267显示了可用的过滤选项。

表 *267.* 过滤选项

过滤选项	**描述**
全部	显示所有表
用户	只显示用户表
系统	只显示系统表
查看	只显示数据库视图

注: 如果要检索特定列，请仅选择这些列。如果您选择表，那么 Agent Builder 将自动构建一个查询来收集该表中所有的列，并针对当前存在于该表中的所有列创建特性。

可以通过以下方法来选择列:

· 选择表将获得对所有列的缺省查询。

· 选择列将只获取那些列。

7. 可选： 修改“**特性信息**”页面中为错误、缺失数据和 No 值设置的枚举值。
   修改这些值以避免与可能从数据库表列返回的合法值发生任何重叠。
8. 可选： 在“**Java 数据库连接 (JDBC) 浏览器**”窗口中单击**测试**，以测试并修改 SQL 语句。
   “**运行 SQL 语句**”窗口随即打开。
   a) 在 **SQL 语句**字段中输入或修改 SQL 语句。
   b) 单击**运行**以运行该 SQL 语句。
   结果将显示在“**结果**”区域中。请继续修改并测试该语句，直到您对返回的数据满意为止。
   c) 单击**确定**以保存该语句、创建正确的特性并返回到“**JDBC 信息**”窗口。
9. 可选： 在“**JDBC 信息**”窗口中单击**测试**，以便在更现实的代理程序环境中测试该属性组。有关测试 JDBC 属性组的更多信息，请参阅第 1086 页的『测试 JDBC 属性组』。如果在此测试期间更改了 JDBC 语句，那么还必须调整特性，以确保对于该 JDBC 语句返回的每个列，都存在一个相应的特性且顺序正确。
10. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息，请参阅第 1011 页的『过滤属性组』
11. 在“**JDBC 信息**”页面的**操作系统**部分中，选择操作系统，并单击**下一步**。有关应选择哪个操作系统的信息，请参阅第 1021 页的『指定操作系统』。

    注: 单击**插入配置属性**以选择要插入的属性。有关更多信息，请参阅第 1143 页的『定制代理程序配置』。
12. 在“**选择键特性**”页面上，选择键特性或指示此数据源仅生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
13. 如果您想测试先前定义的数据源，请在 Agent Editor 窗口中选择**数据源**选项卡，并选择 JDBC 数据源。在“**JDBC 属性组信息**”区域中，单击**测试**。有关测试的更多信息，请参阅第 1086 页的『测试 JDBC 属性组』。
14. 如果您想查看自动生成的配置节，请单击 Agent Editor 的**插入配置属性**选项卡。
    您可以更改这些属性的标签或缺省值，以便与用户最初配置代理程序时看到的缺省值相匹配。
15. 可选： 填写“**特性信息**”页面；有关详细信息，请参阅第 1007 页的『用于定义特性的字段和选项』。如果您已选择手动创建 JDBC 数据源，而不在步骤 第 1081 页的『4』 中单击 Browse，请执行此步骤。

Agent Builder JDBC 数据源支持从大多数 SQL 类型收集数据。第 1083 页的表 268 中的信息描述了当 JDBC Browser 检测到其中一种类型的列时所创建的特性类型。这些数据类型是支持用于监视代理程序的类型。

表 *268.* 支持用于监视代理程序的 *SQL* 数据类型

SQL 数据类型	**创建的 IBM Tivoli Monitoring 特性**
BIGINT	此数据类型是 IBM Tivoli Monitoring 中的 64 位标尺值。如果您选择了 IBM Tivoli Monitoring V6.2 兼容性，那么这是 32 位标尺。
DECIMALDOUBLEFLOATNUMERICREAL	这些 SQL 类型在 IBM Tivoli Monitoring 中将创建为 64 位标尺特性。如果数据库元数据包含小数位数值，那么将使用该值；否则，小数位数设为 1。如果您选择了 IBM Tivoli Monitoring V6.2 兼容性，那么此特性为 32 位标尺。
BITINTEGERSMALLINTTINYINT	下列 SQL 类型在 IBM Tivoli Monitoring 中将创建为 32 位标尺特性。
BOOLEAN	此值在 IBM Tivoli Monitoring 中是 32 位标尺，并具有 TRUE 和 FALSE 枚举。
TIMESTAMP	此类型的列中的数据将转换为 16 字节 IBM Tivoli Monitoring 时间戳记特性。
TIMEDATECHARLONGVARCHARVARCHAR	浏览器将这些 SQL 类型均视为字符串特性。列大小将用作特性大小，最大大小为 256，这是 JDBC 浏览器的缺省字符串特性大小。

注: 如果从未列出的数据类型收集数据，那么缺省情况下使用字符串特性。代理程序还尝试从数据库以字符串形式收集数据。

修改“**特性信息**”页面中为错误、缺失数据和 No 值设置的枚举值（如果必要）。修改这些值以避免与可能从数据库表列返回的合法值发生任何重叠。

JDBC 配置

在代理程序中定义 JDBC 数据源时，将为您创建部分配置属性。

如果在代理程序中定义了 JDBC 数据源，那么该代理程序必须使用 Java 来连接 JDBC 数据库服务器。Java 配置属性将自动添加到该代理程序中。下列 Java 配置属性特定于代理程序运行时配置：

- *Java* 主目录：指向 Java 安装目录的标准路径
- *JVM* 自变量：使用此参数为 Java 虚拟机指定一个可选自变量列表。
- 跟踪级别：：此参数定义要写入 Java 跟踪日志文件的信息量。缺省设置为只将错误数据写入日志文件。

注: Agent Builder 不需要这些 Java 属性，因为它使用自己的 JVM 以及日志记录，这是通过 JLog 插件配置的。

如果在代理程序中定义 JDBC 数据源，那么将向该代理程序中自动添加以下必需的公共配置字段：

- *JDBC database type*：要连接的数据库的类型，即 IBM Db2、Microsoft SQL Server 或 Oracle Database Server。
- *JDBC user name*：用于向数据库服务器认证的用户名。
- *JDBC password*：用于向数据库服务器认证的密码。
- *Base paths*：要在其中搜索 *Class Path* 字段所指定 JAP 文件的目录列表，或者为 *JAR directories* 字段所指定的非标准目录的列表。目录名称在 Windows 系统上以分号（;）分隔，在 UNIX 系统上以分号（;）或冒号（:）分隔。
- *Class path*：代理程序要搜索的显式指定 JAR 文件。在找到 JAR 文件之前，所有具有非标准名称的文件都将追加到每个“基本路径”中。

· *JAR directories*：要在其中搜索 JAR 文件的目录的列表。目录名称在 Windows 系统上以分号（;）分隔，在 UNIX 系统上以分号（;）或冒号（:）分隔。您不必显式指示这些目录中的 JAR 文件；这些 JAR 文件会因为它们在其中一个目录中而被找到。但不会搜索这些目录的子目录。在找到目录之前，所有具有非标准名称的目录都将追加到每个“基本路径”中。

运行时配置还需要您指定一些其他详细信息才能连接到数据库。可以选择如何指定其余配置项，指定为 JDBC URL 或基本配置属性（缺省设置）：

· URL 配置选项

  – JDBC 连接 URL：特定于供应商的连接 URL，提供有关数据库所在的主机以及连接到的端口号的详细信息。URL 格式通常如下所示：

```
jdbc:identifier://server:port/database
```

  请参阅 JDBC 驱动程序供应商文档以了解不同的 URL 格式。

· JDBC 基本属性选项（缺省设置）

  JDBC 服务器名称：正在运行数据库服务器的主机名。
  JDBC 数据库名称：建立了连接的主机上的数据库的名称。
  JDBC 端口号：数据库服务器正在侦听的端口号。

注: 借助 JDBC 数据提供程序，可以使用子节点在同一个代理程序中监视多种数据库类型。要使用该方法进行监视，必须仔细定义“子节点配置覆盖”。如果监视多个数据库类型，以下配置设置很可能不同：

· JDBC 数据库类型

· JDBC 用户名

· JDBC 密码

如果您正在使用基本配置选项，那么还必须在“**子节点配置覆盖**”页面上针对下列属性定义覆盖项：

· JDBC 服务器名称

· JDBC 端口号

· JDBC 数据库名称

要为子节点定义配置覆盖项，请参阅第 1127 页的『使用子节点』，以获取有关访问“**子节点配置覆盖**”页面的更多详细信息。当您在运行时配置代理程序时，必须针对创建的每个新子节点实例配置所有这些属性。

除了配置覆盖项，您的代理程序还必须指向各个数据库类型的 JDBC 驱动程序，您将在子节点中连接到该驱动程序。*JAR* 目录参数是指向 JDBC 驱动程序最方便的方式。使用分号来分隔每个目录，以列出包含 JDBC 驱动程序的目录。例如，如果要使用代理程序连接到 Db2 和 Oracle 数据库，您必须指定类似于以下示例的 *JAR* 目录值：`C:\Program Files\IBM\SQLLIB\java;C:\oracle\jdbc`。

存储过程
可以与 JDBC 数据提供程序配置使用的示例 SQL 和 Db2 存储过程。

JDBC 数据提供程序可以处理存储过程返回的结果集。字符串或整数输入参数可以传递给存储过程。以下语法运行存储过程：

```
call[:index] procedureName [argument] ...
```

其中：

index
可选整数，指定数据提供程序将使用哪个结果集。当存储过程返回多个结果集而您只需要收集其中一个结果集中的值时，此参数很有用。如果不指定 index，将收集并返回每个结果集的数据。

procedureName
JDBC 数据提供程序将运行的存储过程的名称。

argument
存储过程的输入参数。必须使用空格分隔多个自变量。如果自变量包含空格字符，请使用双引号将整个自变量括起来。如果自变量可以解析为整数，那么它将作为整数自变量传递到存储过程。任何括在双引号内的自变量都将作为字符串自变量进行传递。

SQL Server 样本
`call sp_helpdb`

运行过程 `call sp_helpdb`，此过程不需要自变量。返回的所有结果集中的数据都将包括在数据提供程序所返回的数据中。

`call:2 sp_helpdb master`

使用 master 自变量运行过程 `sp_helpdb`。此自变量是一个字符串输入自变量。只有此存储过程所返回的第二个结果集中的数据，才会包括在数据提供程序所返回的数据中。

如果不指定索引，将收集返回的所有结果集中的数据。必须确保这些情况下返回的数据与您定义的属性兼容。Agent Builder 根据返回的第一个结果集创建属性，任何其他结果集都应该与第一个结果集兼容。

Db2 存储过程

以下是以 SQL 编写的样本 Db2 函数。此函数演示如何返回可以由 Agent Builder JDBC 数据提供程序处理的结果：

```
-- Run this script as follows:
-- db2 -td# -vf db2sample.sql

-- Procedure to demonstrate how to return a query from
-- a DB2 stored procedure, which can then be used by
-- an Agent Builder JDBC provider. The stored procedure
-- returns the following columns:
-- Name               Description              Data Type
-- current_timestamp  The current system time  timestamp
-- lock_timeout       The lock timeout         numeric scale 0
-- user               The user for the session String 128 characters long
DROP procedure db2sample#

CREATE PROCEDURE db2sample()
  RESULT SETS 1
  LANGUAGE SQL
BEGIN ATOMIC

  -- Define the SQL for the query
  DECLARE c1 CURSOR WITH HOLD WITH RETURN FOR
  SELECT CURRENT TIMESTAMP as current_timestamp,
CURRENT LOCK TIMEOUT as lock_timeout, CURRENT USER as user
  FROM sysibm.sysdummy1;

  -- Issue the query and return the data
  OPEN c1;
END#
```

从 Agent Builder 中，可以使用与针对其他存储过程定义的相同语法来调用此函数。在这种情况下，请将 `call db2sample` 定义为 JDBC 语句以运行此存储过程。

Oracle 存储过程

Oracle 存储过程不返回结果集。您必须编写返回 Oracle 引用游标的函数。以下是以 PL/SQL 编写的样本 Oracle 函数，此函数演示如何返回可以由 Agent Builder JDBC 数据提供程序处理的结果：

```
CREATE OR REPLACE FUNCTION ITMTEST
RETURN SYS_REFCURSOR
IS
        v_rc SYS_REFCURSOR;
BEGIN
  OPEN v_rc FOR SELECT * FROM ALL_CLUSTERS;
  RETURN v_rc;
END;
```

从 Agent Builder 中，可以使用与针对其他存储过程定义的相同语法来调用此函数。在这种情况下，请将 `call ITMTEST` 定义为 JDBC 语句来运行此存储过程。由于该 Oracle 函数必须返回光标引用，所以 Oracle 函数只能处理一个结果集。这意味着 Oracle 不支持索引选项，因为无法返回多个结果集。

测试 JDBC 属性组

您可以对在 Agent Builder 中创建的 JDBC 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**JDBC 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**测试 JDBC 语句**”窗口。
2. 可选： 在开始测试之前，您可以设置环境变量、配置属性和 Java 信息。

   有关更多信息，请参阅第 1156 页的『属性组测试』。有关 JDBC 配置属性的更多信息，请参阅（第 1083 页的『JDBC 配置』）。
3. 单击**启动代理程序**。

   此时将显示一个窗口，表明代理程序正在启动。
4. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。

   代理程序会使用指定的 SQL 查询对数据库进行查询。“**测试 JDBC 语句**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。

   注: 返回数据的顺序至关重要；例如，第一个返回列中的数据值始终分配给第一个属性。如果更改 JDBC 语句，那么必须对属性进行添加、移除或重新排序，以与此语句返回的列相匹配。
5. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据
6. 通过单击**停止代理程序**停止代理程序。
7. 单击**确定**或**取消**以退出“**测试 JDBC 语句**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

使用 Ping 来监视系统可用性

您可以定义数据源以使用因特网控制报文协议 (ICMP) 回传 Ping 测试网络设备列表。要测试的设备的主机名或 IP 地址列示在一个或多个设备列表文件中。单独的 Ping 配置文件指定每个设备列表文件的路径。然后，在代理程序运行时配置中设置 Ping 配置文件的名称。这些结果包括每个网络设备的状态。

开始之前

创建设备列表文件和 Ping 配置文件（请参阅第 1087 页的『配置文件』）。

关于此任务

网络管理其中一部分职能便是能够决定系统是否对因特网控制报文协议 (ICMP) Ping 进行回应。使用此数据源以监视环境中一组服务器或其他关键设备的基本联机或脱机状态。用 Ping 进行监视不仅简单，而且开销低。要监视一列设备，请向代理程序添加 Ping 数据收集器。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**网络管理数据**。
2. 在**数据源**区域中，单击 **Ping**。
3. 单击**下一步**。
4. 在 **Ping 信息**窗口中的**操作系统**区域中，选择操作系统。
5. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1088 页的『测试 Ping 属性组』

6. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关过滤来自属性组的数据的更多信息，请参阅第 1011 页的『过滤属性组』
7. 执行下列其中一个步骤：

   a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。

   b) 单击**完成**，以保存该数据源并打开 Agent Editor。
8. 有关添加属性的更多信息，请参阅（第 1004 页的『创建属性』）。

结果

有关 Ping 属性组的更多信息，请参阅 第 1223 页的『Ping 属性组』。

配置文件

使用配置文件为代理程序提供要执行 Ping 操作的设备的列表。

代理程序需要两种类型的配置文件。

设备列表文件

包含要执行 Ping 操作的设备的列表。如果有许多设备，您可以在多个设备列表文件中对其进行划分。代理程序对每个设备列表文件启动单独的线性并且以并行方式循环通过这些文件。该线程以 60 秒的时间间隔循环通过每个文件，或以 30 秒加上 Ping 设备列表所花时间的时间间隔循环通过，以两者中时间间隔长的为准。

设备列表文件的语法如下所示：

```
LISTNAME=list_name
device_name or host_name
device_name or host_name
device_name or host_name device_name or host_name
```

其中，*list_name* 是该文件中的设备的描述。如果未定义任何列表名称，那么将使用设备列表文件的名称。列表名称不需要是文件中的第一个条目。但是，如果文件有多个列表名称定义，那么使用最后的定义。

设备列表文件中可以包含的设备数量没有限制。然而，包含过多条目则无法实现定义关键设备目标列表的目的，并会增加总的工作负载。在 60 秒监视时间间隔内检索每个设备的状态可能更难。

在每个循环开始时，代理程序会检查设备列表文件的最后修改时间。如果该文件的最近一次修改时间晚于代理程序最近一次读取该文件的时间，那么代理程序将重新读取该文件，而不需要重新启动。

Ping 配置文件

指定每个设备列表文件的位置。使用 Ping 配置文件位置的标准路径或相对路径。Ping 配置文件作为运行时配置参数传递到代理程序。

示例

在以下示例中，设备分为两个文件。
`/data/retailList.txt` 文件包含以下条目：

```
LISTNAME=Retail
frontend.mycompany.com
productdb.mycompany.com
```

`/data/manufacturingList.txt` 文件包含以下条目：

```
LISTNAME=Manufacturing systems
manufloor.mycompany.com
stats.supplier.com
```

Ping 文件 `/data/pinglists.txt` 包含以下条目：

```
/data/retailList.txt
/data/manufacturingList.txt
```

“网络管理”配置属性

添加 Ping 数据源之后，配置将显示在 Agent Editor 的“**运行时配置信息**”页面上。

“**运行时配置信息**”页面的**网络管理**配置部分包含以下属性：

表 *269*. “网络管理”配置属性

姓名	有效值	是否必需	描述
Ping 配置文件	文件的路径	否。如果未提供此文件，那么将从代理程序 bin 目录使用 KUMSLIST 文件。	包含一列文件（每个文件都包含一列要使用 ICMP Ping 监视的主机）的文件的路径。

测试 Ping 属性组

您可以对在 Agent Builder 中创建的 Ping 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**Ping 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将打开“**测试设置**”窗口
2. 可选： 在开始测试之前，您可以设置环境变量和配置属性。有关更多信息，请参阅第 1156 页的『属性组测试』。
3. 单击**浏览**以选择 Ping 配置文件。有关 Ping 配置文件的更多信息，请参阅 第 1087 页的『配置文件』
4. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
5. 要模拟来自监视环境的代理程序数据请求，请单击**收集数据**。代理程序会对 Ping 配置文件中引用的设备列表文件中所指定的设备执行 Ping 操作。
6. “**测试设置**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
7. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据。
8. 通过单击**停止代理程序**停止代理程序。
9. 单击**确定**或**取消**以退出“**测试设置**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视 HTTP 可用性和响应时间

您可以配置数据源以监视所选 URL 的可用性和响应时间。使用配置文件来定义 URL 列表。在代理程序运行时配置中设置文件名称。在 IBM Tivoli Monitoring 中，也可以使用“执行操作”命令来添加和移除受监视的 URL。每个 URL 的状态作为一行添加到所产生的数据集中。

关于此任务

对于所监视的每个 URL，结果将提供有关对 HTTP 请求作出的 HTTP 响应的一般信息。结果包括能否对其进行检索、检索所耗时间以及响应大小。如果响应内容为 HTML，那么还将提供有关 URL 中的页面对象的信息。

您可以监视使用 HTTP、HTTPS、FTP 和文件协议的 URL。请在 HTTP URL 文件中指定所要监视的 URL，或者通过“执行操作”选项进行指定。

要点: 在发布时，“执行操作”命令在 IBM Cloud Application Performance Management 环境中不可用。它们仅在 Tivoli Monitoring 环境中可用。

此数据源需要 Java 运行时环境。以下 Java 版本受支持：

· Oracle Corporation Java V5 或更高版本

· IBM Corporation Java V5 或更高版本

使用以下过程来创建用于监视 URL 列表的属性组：

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **HTTP**。
3. 单击**下一步**。
4. 在“**HTTP 信息**”页面上的**操作系统**区域中，选择一个或多个操作系统。
5. 可选： 单击**测试**以测试此属性组。有关测试的更多信息，请参阅第 1094 页的『测试 HTTP 属性组』。
6. 可选： 单击**高级**以创建过滤器，从而限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息，请参阅第 1011 页的『过滤属性组』
7. 执行下列其中一个步骤：

   a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。

   b) 单击**完成**，以保存该数据源并打开 Agent Editor。

结果

HTTP 数据源创建两个属性组：“受管 URL”和“URL 对象”。您可以添加、修改或删除属性。

相关任务

第 1004 页的『创建属性』
您可以向数据集添加新属性。

相关参考

第 1225 页的『HTTP 属性组』
受管 URL 和 URL 对象这两个 HTTP 属性组用于接收来自 URL 及其中对象的信息。

HTTP 表

有关缺省 HTTP 属性组的参考信息。

HTTP 数据源所创建的两个属性组为：

受管 URL
“受管 URL”表提供有关每个受监视 URL 的可用性和响应时间数据。

URL 对象
对于嵌入的每个对象，“URL 对象”表都包含一个单独的 URL 条目。例如，“受管 URL”报告中所列的 Web 站点中可能使用的 `.gif` 和 `.jpg` 文件。

有关“受管 URL”和“URL 对象”表中使用的语法的信息，请参阅第 1090 页的『HTTP 属性的特定字段』。

如果要监视 Web 站点中特定对象的响应时间和可用性，请查看“URL 对象”表的内容。“URL 对象”表用于监视在下载的 HTML 文件中检测到的对象的特定列表。下表列示要在其中搜索所要监视的对象的 HTML 元素以及这些元素中引用了对象的属性：

表 *270.* 要在其中搜索所要监视的对象的 *HTML* 元素

HTML 元素	**包含所要监视的对象的属性**
`img`	`src`
`script`	`src`
`embed`	`src`

表 270. 要在其中搜索所要监视的对象的 HTML 元素 (续)	
HTML 元素	**包含所要监视的对象的属性**
object	codebase or data
body	background
input	src

在以下 HTML 片段示例中，受监视对象是 img 元素的 src 属性所引用的图像。

```
<img src="/v8/images/id-w3-sitemark-simple.gif" alt="" width="54" height="33" />
```

此图像的完整 URL 将根据源文档的 URL 计算。

注: 如果您不想监视在网页中找到的对象，请在“URL 监视”配置部分中，将**页面对象收集**属性设置为**否**。

HTTP 属性的特定字段

在“**属性信息**”页面中，有两个用于 HTTP 属性的字段，它们定义从 URL 收集数据的方式。**属性类型**字段可以是列表中的任意值，用于控制所返回的 URL 相关信息。某些属性类型要求**类型值**字段包含值。

下表描述“受管 URL”属性组的所有属性类型及类型值（如果需要）：

表 271. HTTP 属性信息 - 受管 URL				
属性类型	**描述**	**类型值**	**返回的数据类型**	**与 FTP 和文件协议的差别**
XPath Query	对通过 URL 连接返回的内容运行 XPath 查询。此查询必须编写为返回对属性有用的数据，而不是节点列表。	要对通过 URL 连接获取的内容运行的 XPath 查询。	返回的数据可以是字符串、数字或时间戳记值。如果数据采用 XML 日期时间格式，那么您可以将时间戳记指定为属性类型。代理程序将值转换为 Candle Timestamp。	无
Response Time	从所请求的 URL 下载内容所耗用的时间（毫秒）。	无	整数（毫秒数）	无
Response Message	服务器返回的 HTTP 响应消息。	无	String	仅当此 URL 使用 HTTP 或 HTTPS 协议时，响应消息才适用。
Response Code	服务器返回的 HTTP 响应代码。	无	整数	仅当此 URL 使用 HTTP 或 HTTPS 协议时，响应代码才适用。对于文件或 FTP URL，此代码始终为 0。
Response Length	从所请求的 URL 下载的内容的大小（以字节计）	无	整数（以字节计的大小）	无

表 271. HTTP 属性信息 - 受管 URL (续)				
属性类型	**描述**	**类型值**	**返回的数据类型**	**与 FTP 和文件协议的差别**
Response Header	可用于从其中一个 URL 响应头字段检索值的响应头。自变量指定所要请求的字段。	要收集的响应头。	String	通常，FTP 和文件协议没有任何可收集的头。
Request URL	这是所连接的 URL。所有响应关键字都将提供有关与此 URL 的连接的信息。您可以使用 XPath 查询从通过访问此 URL 返回的内容中获取信息。	无	String	无
页面对象数	在受监视 HTML 页面上发现的受“URL 对象”属性组监视的对象数。	无	整数	无
对象总大小	在此 Web 页面的“URL 对象”属性组中监视的对象的大小总计。	无	整型（字节）	无
别名	用户对此 URL 指定的别名。	无	String	无
用户	用户对此 URL 指定的数据。	无	String	无

下表描述“URL 对象”属性组的属性类型：

表 272. HTTP 属性信息 - URL 对象				
属性类型	**描述**	**类型值**	**返回的数据类型**	**与 FTP 和文件协议的差别**
URL	在“受管 URL”表中监视的 URL。	无	String	无
对象名称	在 **HTML** 页面中监视的对象的 URL。	无	String	无
对象大小	从对象名 URL 下载的内容的大小（以字节计）。	无	数字	无
对象响应时间	下载页面对象所用的时间（毫秒）。	无	数字	无

监视 URL

通过将 URL 包括在 URL 文件中，或者通过使用“HTTP URL 添加执行操作”选项，可以对任何 URL 启动监视。

URL 文件

配置中指定的 URL 文件可以位于任意目录中。如果此文件不存在或为空，那么您可以使用“执行操作”来启动 URL 监视。有关更多信息，请参阅第 1092 页的『“执行操作”选项』。如果您已有使用 Tivoli Universal Agent HTTP Data Provider 的 Tivoli Universal Agent，那么可以复用 KUMPURLS 文件。配置此代理程序时，请指向 KUMPURLS 文件。

下表根据添加 URL 的方法提供如何将 URL 输入到 URL 文件的示例。

表 *273. URL* 文件条目

URL	**添加者**

`www.bbc.co.uk`
`http://weather.com`
`www.ibm.com`
	以手动方式在此文件中添加条目。如果未指定协议（如 `www.ibm.com` 示例所示），那么将采用 HTTP。

`ftp://userid:password@ftpserver/index.html`	使用文件传输协议 (FTP) 以手动方式添加
`http://www.ibm.com USER=ibm ALIAS=ibm`	使用“HTTP URL 添加”执行操作
`file:/tmp/samples.html USER=samples \` `ALIAS=samples`	使用通过 FTP 执行的“HTTP URL 添加”执行操作
`http://google.com INTERVAL=60 CACHE=50 \` `USER=google ALIAS=search`	Tivoli Universal Agent KUMPURLS 文件中的示例

如果您直接编辑 URL 文件，那么所作的更改将在此代理程序下次收集数据时实现。

“执行操作”选项

另外，还可以通过名为“HTTP URL 添加”的执行操作选项指定要监视的 URL。

限制: 此选项在 IBM Cloud Application Performance Management 的当前发行版中不可用，这是因为，您无法手动启动“执行操作”命令。

选择此选项后，系统将显示一个窗口，您可以在该窗口中指定下列参数:

URL
这是一个必需参数，用于表示 URL 自身。您可以在指定或不指定 `http://` 或 `https://` 前缀的情况下输入此参数。

别名
这是一个可选参数，您可以通过指定此参数使更有意义的名称与 URL 相关联。在此参数中，不允许输入空格。如果未填写此参数，那么“别名”将缺省为空。

`User_Data`
这是一个可选参数，您可以通过指定此参数输入关于 URL 的数据。如果未填写此参数，那么 `User_Data` 将缺省为 `INITCNFG`。

填写信息并关闭窗口后，请对与代理程序相关联的目标受管系统指定 HTTP URL 添加操作。针对新 URL 进行的监视将立即开始。此 URL 还将添加到 URL 文件中，因此，代理程序重新启动后将继续对其进行监视。

相应的“执行操作”选项名为 HTTP URL 移除。使用 HTTP URL 移除操作立即停止对特定 URL 的监视。这还会从 URL 文件中删除所移除的 URL。“**HTTP URL 移除**”窗口仅要求提供 URL 和 `User_Data` 值。URL 和 `User_Data` 值必须与 Tivoli Enterprise Portal 中显示的值匹配，否则“移除”操作将失败。例如，即使在“添加”操作的 URL 字段中省略了 `http://`，也必须将其包括在“移除”操作的 URL 字段中。如果未指定 `User_Data`，那么必须按 Tivoli Enterprise Portal 中显示的那样指定 `INITCNFG`。

如果已将 URL 手动添加到 URL 文件中，那么可使用“执行操作”将其删除。如果使用“执行操作”进行删除，那么必须按 Tivoli Enterprise Portal 中显示的那样指定值。例如，如果在 URL 文件中添加了 `www.ibm.com`，那么 Tivoli Enterprise Portal 将显示 `http://www.ibm.com` 作为 URL，并显示

INITCNFG 作为 User_Data。要使用“执行操作”来移除此 URL，必须使用 Tivoli Enterprise Portal 中显示的值。

填写信息并关闭窗口后，请对与代理程序相关联的目标受管系统指定 HTTP URL 移除操作。

监视 https:// URL
HTTP 数据源仅监视不需要脚本式访问或交互式提示的安全的 https:// URL。

如果可使用标准 HTTP Get 调用来检索 https:// URL，那么可对其进行监视。

代理服务器
如果正在运行代理程序的系统需要代理才能访问 SOAP 数据提供程序，那么必须指定代理服务器配置属性。

有关更多信息，请参阅第 1093 页的『代理服务器配置』。

HTTP 配置
有关 HTTP 配置的参考信息。

添加 HTTP 数据源之后，配置将显示在 Agent Editor 的“**运行时配置**”页面上。对于 URL 监视、代理服务器认证和 Java，添加了相应的配置部分。

“URL 监视”配置

“URL 监视”配置节包含以下属性：

表 *274.* “*URL* 监视”配置属性

姓名	有效值	是否必需	描述
HTTP URL 文件	文件的路径	是	包含 URL 列表的文件的路径。
页面对象收集	“是”和“否” 缺省值为“是”。	否	是否下载在 Web 页面中找到的对象并从中收集数据。

代理服务器配置

代理服务器配置节包含以下属性：

表 *275.* 代理服务器配置属性

姓名	有效值	是否必需	描述
代理主机名	String	否	要用于 HTTP 连接的代理主机名。
代理用户名	String	否	代理服务器的用户名。
代理端口	正整数 缺省值为 80。	否	代理服务器的 HTTP 端口号。
代理密码	密码	否	代理服务器的密码。

注: 如果**代理主机名**属性为空白，那么将不使用代理。

Java 配置

如果在代理程序中定义了 HTTP 数据源，那么该代理程序必须使用 Java 来连接到 HTTP 服务器。Java 配置属性将自动添加到代理程序中。下列 Java 配置属性特定于代理程序运行时配置。Agent Builder 不需要这些 Java 属性，因为它使用自己的 JVM 以及日志记录，这是通过 JLog 插件配置的：

表 276. Java 配置属性			
姓名	**有效值**	**是否必需**	**描述**
Java 主目录	目录的标准路径	否	指向 Java 安装目录的标准路径。
跟踪级别	选项 （缺省值为“错误”）	是	请使用此属性来指定 Java 提供程序所使用的跟踪级别。
JVM 自变量	String	否	请使用此属性来指定 Java 虚拟机的可选自变量列表。

测试 HTTP 属性组
您可以对在 Agent Builder 中创建的 HTTP 属性组进行测试。

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**HTTP 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**HTTP 测试**”窗口。
2. 单击**浏览**以选择 HTTP URL 文件。有关 URL 文件的更多信息，请参阅第 1091 页的『URL 文件』。
3. 可选： 在开始测试之前，请设置环境变量、配置属性及 Java 信息。
   有关更多信息，请参阅第 1156 页的『属性组测试』。有关 HTTP 配置的更多信息，请参阅 第 1093 页的『HTTP 配置』。
4. 单击**启动代理程序**。
   此时将显示一个窗口，表明代理程序正在启动。
5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。
   代理程序会监视 HTTP URL 文件中定义的 URL。“**HTTP 测试**”窗口将显示所返回的全部数据。
6. 可选： 如果返回的数据未如预期，请单击**检查结果**。
   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』 对 Data collection Status 窗口所收集并显示的数据作了描述。
7. 通过单击**停止代理程序**停止代理程序。
8. 单击**确定**或**取消**以退出“**HTTP 测试**”窗口。单击**确定**将保存您所作的全部更改。

相关概念
第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

监视来自 SOAP 或其他 HTTP 数据源的数据

您可以定义数据源，以接收来自 HTTP 服务器的数据（例如，使用 SOAP 协议来接收数据）。该数据源将 HTTP 请求发送到 URL，并将 XML、HTML 或 JSON 格式的响应解析为所产生数据集的属性。您可以选择要从该请求中检索的数据。

关于此任务

通过使用 SOAP 数据源，您可以指定 HTTP URL 并发送 GET、POST 或 PUT 请求。对于 POST 或 PUT 请求，您可以指定关联的 POST 数据。系统将检索并解析 XML、HTML 或 JSON 响应，并通过属性将数据呈现给监视环境。您可以将属性定义为特定元素中的所有值。您还可以定义定制 XPath 值，以指定如何填充各个属性。也可以结合使用这两种机制。

使用以下过程从 URL 中收集并解析 XML、HTML 或 JSON 响应：

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**来自服务器的数据**区域中的**日志数据**。
2. 在**数据源**区域中，单击 **SOAP**。
3. 单击**下一步**。
4. 在“**SOAP 信息**”页面上，输入 URL。
   缺省值为：

   ```
   http://${KQZ_HTTP_SERVER_NAME}:${KQZ_HTTP_PORT_NUMBER}
   ```

   注: 可以使用一个或多个解析为 URL 的配置变量。单击**插入配置属性**以选择要插入的属性。有关更多信息，请参阅第 1143 页的『定制代理程序配置』。
5. 选择请求类型。缺省请求类型为 Get。对于 Post 和 Put 请求，请输入要处理的数据。

   注: 对于 Post 和 Put 请求，将启用**插入配置属性**。单击**插入配置属性**以在要处理的数据中包括配置变量。有关更多信息，请参阅第 1143 页的『定制代理程序配置』。
6. 单击**浏览**。

   注: 如果输入 URL 并选择了请求类型，但不想使用 SOAP 浏览器来构建定义，请输入**行选择 XPath**。您可在“**SOAP 信息**”窗口中输入**行选择 XPath**。然后为属性组定义所有属性。
7. 在“**SOAP 浏览器**”窗口中，执行下列步骤：
   a) 输入 URL 和选择请求类型（如果尚未执行此操作）。
   b) 单击**配置**以设置在 URL 或其他字段中引用的任何配置属性。
   c) 单击**连接**，以从 SOAP 提供者获取数据。

   连接至 URL 时，文档对象模型 (DOM) 树将显示此 URL 的 XML 元素列表。HTML 或 JSON 响应将转换为 XML，并显示为 DOM 树。有关将 JSON 响应转换为 XML 的详细信息，请参阅第 1098 页的『JSON 数据的 XML 表示』。在（第 1096 页的图 44）中的 WebSphere Application Server 示例中，输入了以下 URL：

   ```
   http://nc053011.tivlab.raleigh.ibm.com:9080/wasPerfTool/servlet
   /perfservlet?module= \threadPoolModule
   ```

   这将显示 XML 元素 `PerformanceMonitor`。此元素是 SOAP 提供者返回的 XML 文档中的顶级 XML 元素。

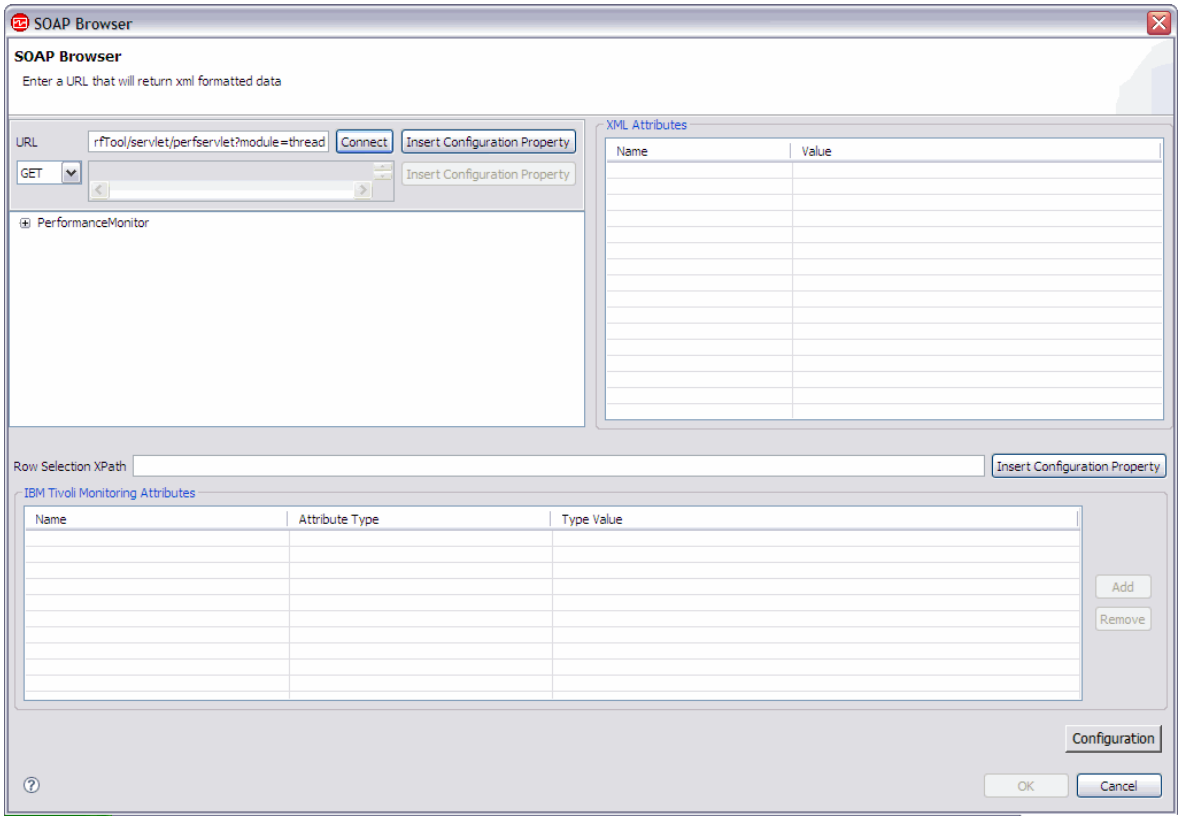


图 *44.* “***SOAP* 浏览器**”窗口

d) 在 DOM 树中，查找并选择要设置为**行选择 XPath** 的 XML 节点。

在第 1097 页的图 45 中的 WebSphere Application Server 示例中，选择了 `PerformanceMonitor/Node/Server/Stat/Stat/Stat` 节点。此节点表示属性组中的一行数据。在 DOM 树中选择节点并单击**添加**时，您将获得在树中该节点上定义的所有属性和元素。（单击**代理程序属性**区域中的**添加**。）

选择节点时，**XML 属性**区域将显示为所选节点定义的任何 XML 属性。选择 XML 属性，然后单击**添加**以将此属性包含在“代理程序属性”列表中。

注: 如果预期会有多行数据，那么 XPath 必须映射至节点集。当 Row Selection XPath 返回仅设置了一个项的节点时，属性组仅包含一行。

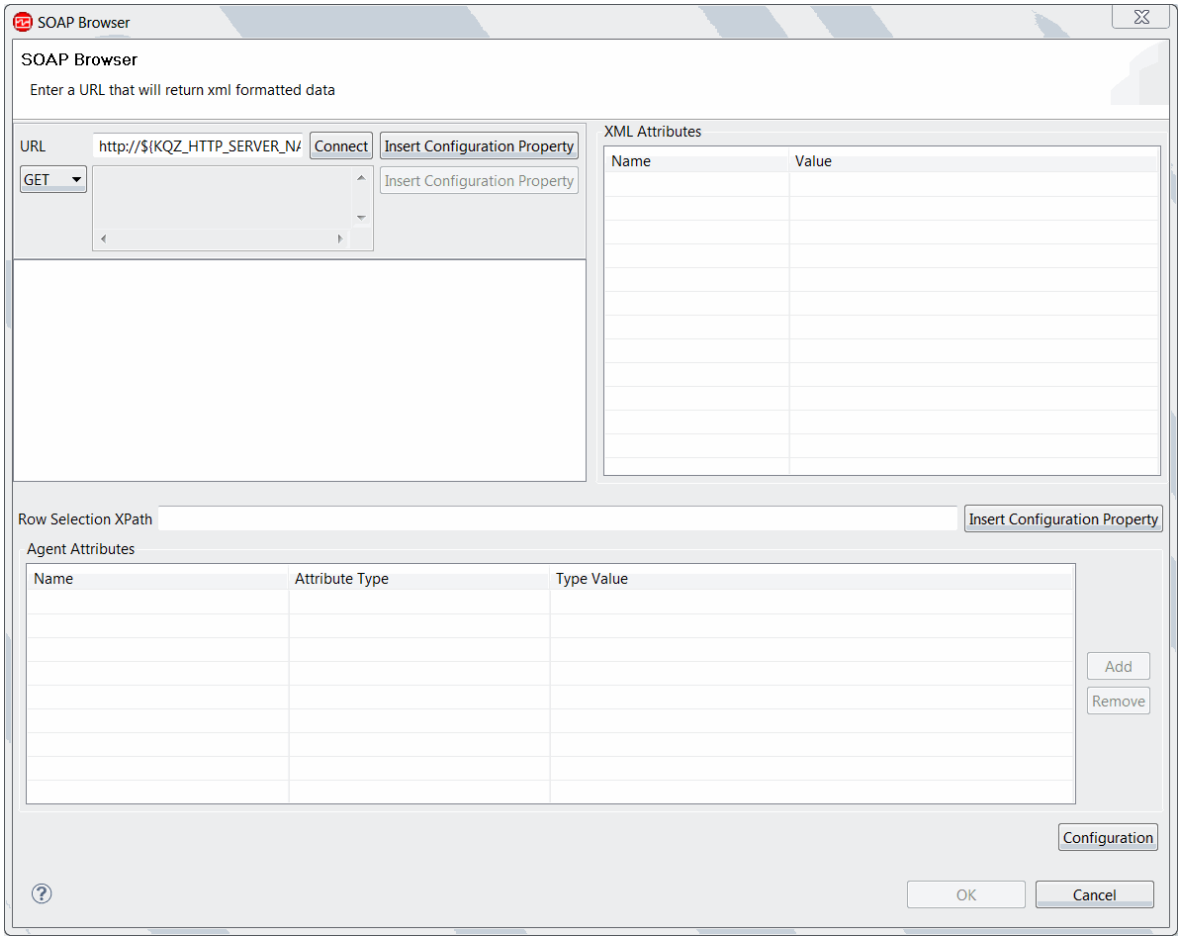


图 *45.* “***SOAP* 浏览器**”窗口

e) 单击“代理程序属性”区域中的**添加**。

此时会显示代理程序属性列表，并填充了**行选择 XPath** 字段。

每个代理程序属性的 XPath 用于将 XML 节点或元素映射到代理程序属性。在第 1098 页的图 46 中的 WebSphere Application Server 示例中，代理程序属性列表中的第一个属性 `Stat` 无用，会将其移除。

您可以在**类型值**字段中编辑代理程序属性的名称和 XPath。有关使用 XPath 的更多信息，请参阅 第 1100 页的『XPath 选项』

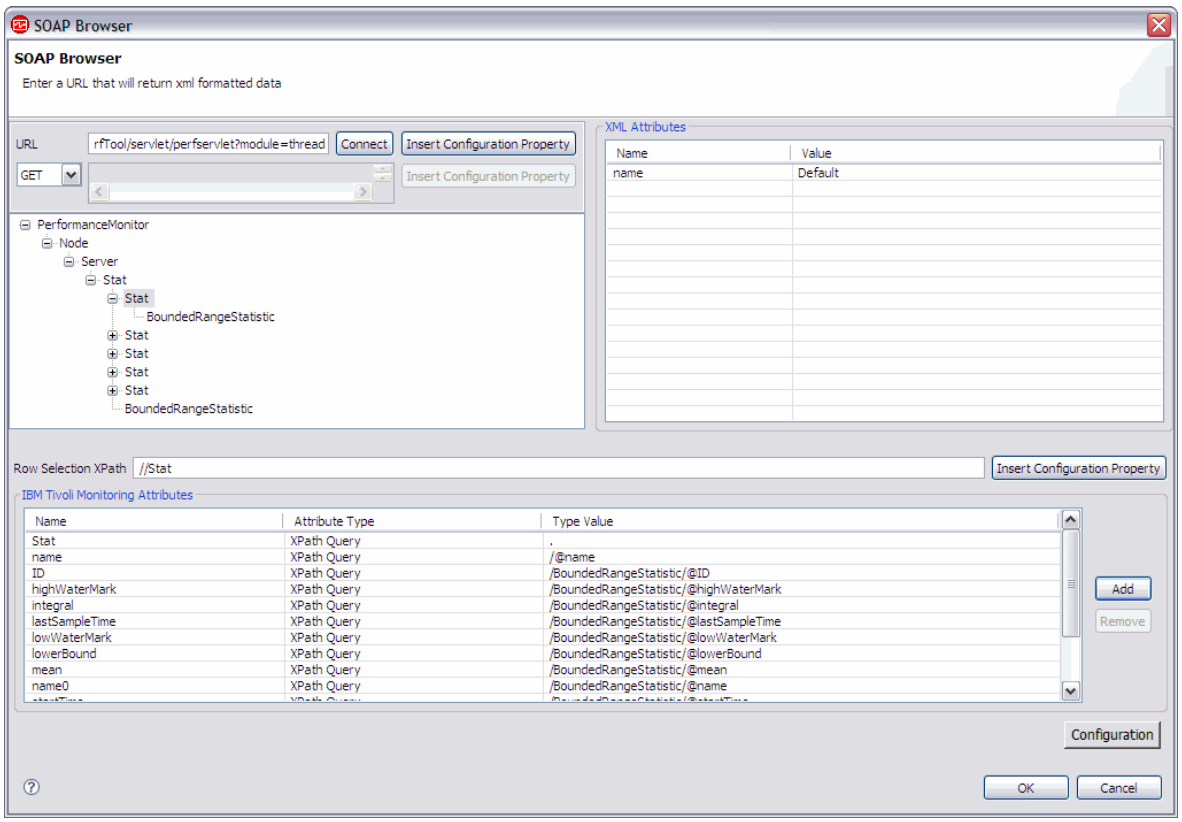


图 *46.* “***SOAP* 浏览器**”窗口

f) 在“**SOAP 浏览器**”窗口中，单击**确定**以保存更改并返回到“**SOAP 信息**”窗口。

8. 在“**SOAP 信息**”窗口中，单击**下一步**。
9. 如果您之前未使用**浏览**，而是在“**SOAP 信息**”窗口中输入了 **URL** 和**行选择 XPath**，那么将显示“**属性信息**”页面。在“**属性信息**”页面上指定第一个属性的信息，然后单击**完成**。然后，可以使用 Agent Editor 指定更多属性。有关创建属性的更多信息，请参阅（第 1004 页的『创建属性』）。
10. 如果您在步骤 第 1095 页的『6』 中使用了**浏览**功能，那么将显示“**选择键属性**”页面。在“**选择键属性**”页面上，选择键属性或指示此数据源仅生成一个数据行。有关更多信息，请参阅第 990 页的『选择键属性』。
11. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1101 页的『测试 SOAP 属性组』
12. 可选： 通过单击**高级**，您可以创建过滤器，以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息，请参阅第 1011 页的『过滤属性组』
13. 执行下列其中一个步骤：

    a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。

    b) 单击**完成**，以保存该数据源并打开 Agent Editor。

JSON 数据的 XML 表示

如果 HTTP 请求返回 JSON 数据，那么数据提供程序会将该数据转换为 XML。

数据提供程序会将 JSON 属性名称转换为元素名称。对于简单类型的 JSON 属性，它会将值转换为元素中的文本数据。嵌入式 JSON 对象会转换为嵌入式 XML 元素。任何下级属性都会转换为下级元素。

根 XML 元素是 `JSON_document`。

如果 JSON 属性名称包含不适用于元素名称的字符，那么数据提供程序会进行修改，以生成有效的元素名称。数据提供程序还会向该元素添加一个 `JSON_name` 属性。该属性的值是原始 JSON 属性名称。

针对 JSON 数组的每个元素，数据提供程序会创建一个 `JSON_`*xxxx*`_array_element` XML 元素，其中 *xxx* 是数组名称。该数组元素的值会转换为 XML 元素中的文本。系统会向每个 XML 元素添加一个 `JSON_index` 属性；该属性的值是数组中数组元素的索引。

数据提供程序会向每个元素添加下列属性：

· JSON_level：节点在 JSON 文件中所处的级别。由 JSON_document 标记所代表的树根处于级别 1。

· JSON_type：JSON 节点的类型（对象、数组、字符串或数字）。

SOAP 属性的特定字段

在“**属性信息**”窗口中，有两个 SOAP 属性字段，用于定义如何从 SOAP 响应收集数据。

属性类型字段可以是列表中的任意值，用于控制所返回的响应相关信息。某些属性类型要求**类型值**字段包含值。缺省属性类型为 XPath Query，它对 SOAP 服务器响应内容运行 XPath 查询。此类型值是所运行的 XPath 查询。下表描述所有属性类型及类型值（如果需要）：

表 *277. SOAP* 属性信息

属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议的差别
XPath Query	对通过 URL 连接返回的内容运行 XPath 查询。此查询必须编写为返回对属性有用的数据，而不是节点列表。	要对通过 URL 连接获取的内容运行的 XPath 查询。如果定义了行选择查询，那么此 XPath 查询必须相对于行选择查询。	返回的数据可以是字符串、数字或时间戳记值。用于 SOAP 的 Agent Builder 浏览器通常从所浏览的数据检测属性的正确数据类型。如果数据采用 XML 日期时间格式，那么您可以将时间戳记指定为属性类型，代理程序会将该值转换为 Candle 时间戳记。	无
Response Time	从所请求的 URL 下载内容所耗用的时间（毫秒）。	无	整数（毫秒数）	无
Response Message	服务器返回的 HTTP 响应消息。	无	字符串	仅当此 URL 使用 HTTP 或 HTTPS 协议时，响应消息才适用。
Response Code	服务器返回的 HTTP 响应代码。	无	整数	仅当此 URL 使用 HTTP 或 HTTPS 协议时，响应代码才适用。对于文件或 FTP URL，此代码始终为 0。
Response Length	从所请求的 URL 下载的内容的大小（以字节计）	无	整数（以字节计的大小）	无
Response Header	可用于从其中一个 URL 响应头字段检索值的响应头。自变量指定所要请求的字段。	要收集的响应头字段。	字符串	通常，FTP 和文件协议没有任何可收集的头。

表 277. SOAP 属性信息 (续)				
属性类型	**描述**	**类型值**	**返回的数据类型**	**与 FTP 和文件协议的差别**
Request URL	已建立与此 URL 的连接。所有响应关键字都将提供有关与此 URL 的连接的信息。您可以使用 XPath 查询从通过访问此 URL 返回的内容中获取信息。	无	字符串	无

XPath 选项

通过使用 XML 路径语言，您可以从 XML 文档选择节点。SOAP 数据源的 XPath 的一些可能用法包括：

- 在 XPath 中使用谓词来标识对应于 IBM Tivoli Monitoring 属性组中的数据行的 XML 元素。您可以在用于将 XML 元素或属性映射至 Tivoli Monitoring 属性的 XPath 中使用谓词，如以下示例所示：

  ```
  Stat[@name="URLs"]/CountStatistic[@name="URIRequestCount"]/@count
  ```

  其中，XPath 包含多个位置步骤，每个位置步骤可以包含一个或多个谓词。这些谓词可能比较复杂并包含布尔值或公式运算符。例如：

  ```
  //PerformanceMonitor/Node/Server[@name="server1"]/Stat/Stat/Stat[@name=
  "Servlets"]/Stat
  ```

- 在 XPath 中包括节点集函数，前提是某一行包含多个同一类型的 XML 元素。并且节点列表中 XML 元素的位置确定了此元素所映射到的 Tivoli Monitoring 属性。节点集函数的示例包括 `position()`、`first()`、`last()` 和 `count()`。
- 执行简单数据变换，例如 substring。如果指定以下 substring：

  ```
  substring(myXMLElement,1,3)
  ```

  此 XPath 将返回 XML 元素 myXMLElement 的前三个字符。

您可以使用两个句点（`..` 在 Row Selection XPath 上下文外部指定元素，如以下示例所示：

```
/../OrganizationDescription/OrganizationIdentifier
```

SOAP 配置

添加 SOAP 数据源之后，配置将显示在 Agent Editor 的“**运行时配置**”页面上。

对于 HTTP 服务器、代理服务器和 Java，添加了相应的配置节。有关代理服务器配置的信息，请参阅（第 1093 页的『代理服务器配置』）。有关 Java 配置的信息，请参阅 第 1093 页的『Java 配置』。

HTTP 服务器

“HTTP 服务器”配置部分包含以下属性：

表 278. HTTP 服务器配置属性			
姓名	**有效值**	**是否必需**	**描述**
HTTP 用户名	字符串	否	HTTP 用户
HTTP 密码	密码	否	HTTP 服务器密码
HTTP 服务器名称	字符串 （缺省值为 localhost）	否	HTTP 服务器的主机或 IP 地址

表 278. HTTP 服务器配置属性 (续)			
姓名	**有效值**	**是否必需**	**描述**
HTTP 端口号	数字 （缺省值为 80）	否	HTTP 服务器的主机或 IP 地址
启用证书验证	True 或 False （缺省值为 True）	是	禁用证书验证可能不安全
HTTP 信任库文件	文件的路径	否	HTTP 信任库文件
HTTP 信任库密码	HTTP 信任库密码	否	HTTP 信任库密码

代理服务器

如果正在运行代理程序的系统需要代理才能访问 SOAP 数据提供程序，那么必须指定代理服务器配置属性。有关更多信息，请参阅第 1093 页的『代理服务器配置』。

测试 SOAP 属性组

您可以对在 Agent Builder 中创建的 SOAP 属性组进行测试

过程

1. 可以通过以下方法启动测试过程：
   - 在创建代理程序期间，单击“**SOAP 信息**”页面上的**测试**。
   - 在创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**测试 SOAP 收集**”窗口。
2. 可选： 在开始测试之前，您可以设置环境变量、配置属性和 Java 信息。

   有关更多信息，请参阅第 1156 页的『属性组测试』。有关 SOAP 配置的更多信息，请参阅 第 1100 页的『SOAP 配置』。
3. 更改 URL、Row Selection XPath 和请求类型。
4. 单击**启动代理程序**。

   此时将显示一个窗口，表明代理程序正在启动。
5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。此操作将填充 Results 表，您可以预览数据在 Tivoli Enterprise Portal 中的列中的解析和显示方式。

   在 Results 区域中，可以更改属性定义，然后重新装入数据以查看这些更改如何影响属性组。您可以在列结果区域中右键单击以显示用于编辑属性的选项。属性编辑选项为：

   - **编辑属性**
   - **隐藏属性**
   - **在前面插入属性**
   - **在后面插入属性**
   - **移除**
   - **移除后续属性**
   - **全部移除**
6. 可选： 如果返回的数据未如预期，请单击**检查结果**。

   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了“**数据收集状态**”窗口收集和显示的数据。
7. 通过单击**停止代理程序**停止代理程序。
8. 单击**确定**或**取消**以退出“**测试 SOAP 收集**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』

使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

使用套接字来监视数据

您可以定义数据源以使用 TCP 套接字从外部应用程序收集数据。应用程序必须启动与代理程序的 TCP 连接，并以结构化 XML 格式发送数据。根据应用程序，数据源可能产生包含单个行、多行或事件数据的数据集。

关于此任务

使用套接字数据源从外部应用程序向代理程序提供数据，该外部应用程序与代理程序运行相同的系统。这个外部应用程序可以根据需要随时将数据发送到代理程序。例如，您可以开发一个命令行界面，以便允许用户在该界面运行时将数据发布到某个属性组。另一种方法是修改受监视的应用程序，使其将更新发送到代理程序。代理程序不会启动或停止正在将数据发送到套接字的应用程序；该操作由用户进行控制。

以下是一些使用套接字数据源的限制：

- 缺省情况下，只能连接到本地主机 (127.0.0.1)。有关配置代理程序以接受来自远程主机的连接的更多信息，请参阅第 1108 页的『远程套接字端口连接』。
- 套接字 API 未提供使客户机能够确定哪些子节点可用的机制。客户机可以针对特定子节点发送数据，但它必须已经知道该子节点的名称。

您可以使用以下过程来创建属性组，以便通过传输控制协议 (TCP) 套接字收集数据。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**定制程序**。
2. 在**数据源**区域中，单击**套接字**。
3. 单击**下一步**。
4. 在“**套接字信息**”页面上，输入属性组名称。
5. 输入此属性组的帮助文本。
6. 选择这个属性组是**生成单一数据行**、**可生成多个数据行**还是**生成事件**。有关更多信息，请参阅第 1104 页的『发送数据』。
7. 在“套接字信息”部分中，选择**代码页**。有关更多信息，请参阅第 1106 页的『字符集』。
8. 可选： 单击**高级**，以修改此属性组的高级属性。当您选择属性组**可生成多个数据行**或**生成事件**时，**高级**选项处于活动状态。
9. 单击**下一步**。
10. 在“**特性信息**”页面上，指定此属性组的第一个特性。有关创建特性的更多信息，请参阅第 1004 页的『创建属性』。
11. 单击**下一步**。
12. 可选： 在“**全局套接字数据源信息**”页面上的**错误码**部分中，定义套接字客户机在无法收集数据时可以发送的错误代码。有关更多信息，请参阅第 1105 页的『发送错误而不是数据』。要定义错误代码，请使用以下步骤：

    a) 在**错误码**部分中，单击**添加**。错误代码限长 256 个字符。只允许使用 ASCII 字母、数字和下划线。不允许使用空格。

    b) 在“**套接字错误码定义**”窗口中，输入 **Performance Object Status** 属性组中显示的值。

    c) 输入内部值。内部值必须是介于 1,000 与 2,147,483,647 之间的整数。

    d) 必须为每个错误定义消息文本。您可以通过从列表中进行选择来使用先前输入的消息文本。单击**确定**，以返回到“**全局套接字数据源信息**”页面。该消息文本在代理程序日志文件中使用。

    如果没有合适的可用消息文本，请单击**浏览**以设置消息文本。此时将打开 Messages（列表）窗口。Messages 窗口列出代理程序中定义的消息。在定义消息之前，该列表保持空白。您可以使用**编辑**来变更已定义的消息，并可以使用**移除**来删除一条或多条已定义的消息。

    e)

在“消息”（列表）窗口中，单击**添加**以查看“**消息定义**”窗口。在“**消息定义**”窗口中，输入用于描述新消息含义的文本，并选择消息类型。

注: 系统会为您自动生成消息标识。

f) 单击**确定**。

g) 此时将打开 Messages（列表）窗口，其中显示了新消息。要验证消息并返回到“**全局套接字数据源信息**”页面，请单击**确定**。

13. 可选： 在“**全局套接字数据源信息**”页面的**补充文件**部分中，您可以添加与代理程序打包在一起的文件。这些文件会在安装该代理程序时复制到代理程序系统。

文件类型列描述每个文件的预期用法。下表描述了三种可能的用途：

表 *279.* 补充文件的文件类型

文件类型	**描述**
Executable	如果要随代理程序一起提供可执行文件，请选择此选项。代理程序不使用这些文件。
Library	如果要随代理程序一起提供库，请选择此选项。代理程序不使用这些文件。
Java resource	如果要随代理程序一起提供 Java 资源，请选择此选项。代理程序不使用这些文件。

有关随代理程序一起安装的补充文件安装位置的信息，请参阅（第 1170 页的『系统上的新文件』）。

单击**编辑**可以对导入的文件进行编辑。有关更多信息，请参阅（第 1075 页的『编辑命令文件定义』）。

14. 可选： 您可以通过单击**测试**来测试此属性组。有关测试的更多信息，请参阅第 1109 页的『测试套接字属性组』

15. 可选： 如果已对数据源采样，您可以单击**高级**来创建过滤器以限制此属性组返回的数据。如果没有在“**套接字信息**”页面上选择“生成事件”，那么会对数据源采样。有关过滤来自属性组的数据的更多信息，请参阅第 1011 页的『过滤属性组』

16. 请执行以下步骤之一：

a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。

b) 单击**完成**，以保存该数据源并打开 Agent Editor。

在“**套接字提供程序设置**”页面的**操作系统**部分中，选择代理程序侦听来自套接字客户机的数据时所在的操作系统。要打开此页面，请在大纲视图中单击**套接字提供程序设置**，或者在 Agent Editor 中的任何套接字属性组页面上单击**全局设置**。

注: 可以在“**套接字提供程序设置**”页面的**错误码**和**补充文件**部分中更新错误码及补充文件。

将套接字信息发送到代理程序

当代理程序包含一个或多个套接字属性组时，该代理程序将打开一个套接字并侦听来自客户机的数据。

将套接字数据发送到代理程序的应用程序连接到该代理程序中已定义的端口。该端口是代理程序配置属性设置的值，或者是按照 TCP/IP 自动分配的临时端口。有关套接字端口和配置的更多信息，请参阅第 1107 页的『套接字配置』。

所接收的数据必须遵循结构化 XML 格式。使用套接字数据源可以实现以下 XML 信息流：

- 对于采样属性组，将一个或多个数据行发送到代理程序
- 对于生成事件的属性组，将一个数据行发送到代理程序
- 将错误代码发送到代理程序，而非数据。
- 将任务前缀注册发送到代理程序
- 接收来自代理程序的任务请求
- 将任务响应发送到代理程序

发送数据

定义属性组以接收采样数据或事件数据。在创建属性组时，您将指定一个选项以指示要接收的数据应执行以下哪种操作：

· 生成单一数据行

· 生成多个数据行

· 生成事件

如果您选择**生成单一数据行**或**可生成多个数据行**，则为采样属性组。如果选择**生成事件**，那么每次收到行时，属性组都会向监视环境发送事件。

在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中查看采样数据时，会看到最新收集的一组数据行。针对事件属性组显示的数据是代理程序维护的本地高速缓存的内容。对于事件数据，代理程序会向高速缓存添加新条目，直到达到此大小为止，此时会删除最旧的条目。对于采样数据，您每次发送数据时，代理程序都将替换高速缓存的内容。

如果选择了**生成事件**或**生成单一数据行**，则必须在每条消息中针对该属性组，向代理程序仅发送一行数据。您可以根据需要发送任意数目的事件，并在独立的消息中发送各个事件。

通常，采样数据按请求由代理程序收集，但套接字客户机将按自己的时间表提供已更新的样本。您可以根据需要按任意频率更新采样属性组（单行或多行）。在 Tivoli Monitoring 或 IBM Cloud Application Performance Management 请求数据时，代理程序会提供最新的数据。

如果 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中缺少套接字属性组的数据行，请检查日志文件中是否存在错误。此外，如果该属性组中的数据与期望不符，也请检查日志文件中是否存在错误。套接字数据源将尝试处理它所能够处理的任何输入内容。例如，如果客户机发送了三个格式正确的行和一个无效的行（例如，格式不正确的 XML），那么您将看到：

· 在属性组中有三个数据行

· 在代理程序的日志文件中，针对格式不正确的行记录了错误

· 由于返回了有效的行，因此 Performance Object Status 将显示 NO_ERROR 状态

对于事件数据和采样数据，数据作为单一 XML 数据流从套接字客户机发送到代理程序。从套接字客户机发送的数据必须始终以换行符“\n”结尾。代理程序将读取换行符之前的所有数据，然后尝试处理接收到的数据。接收到的任何无法处理的数据都将被废弃。以下样本说明如何将名为 abc 的属性组的两个数据行发送到代理程序：

```
<socketData><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/></in><in> \
<a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>\n
```

此样本将两个数据行发送到代理程序，并且每个数据行都包含三个属性。这些属性的顺序很重要，并且必须遵循属性组中定义的顺序。唯一的例外情况是，必须跳过派生的属性，而不必考虑这些属性在属性组中的位置。

如果在子节点中定义此属性组，那么向代理程序发送数据时，必须标识子节点实例标识。通过使用 socketData 元素中的子节点属性来标识子节点实例标识。在配置供套接字客户机使用的子节点实例标识时必须采用一种约定，因为客户机无法查询实例标识或配置属性。发送到未配置的子节点的数据将被忽略。

以下是一个样本：

```
<socketData subnode="app1"><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/>
</in><in> \<a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>\n
```

在此示例中，将数据发送到实例标识为 "app1" 的子节点。"app1" 不是受管系统名称，而是在配置子节点实例时指定的实例标识。

套接字数据由下列 XML 元素构成：

socketData
根元素。它包含一个可选属性，名为 subnode，用于指定子节点实例标识。

attrGroup
此元素用于标识套接字数据的相应属性组。name 属性是必需的，用于指定属性组名。

in
此元素是必需的，用于标识新的数据行。一个数据行的所有属性值都必须是同一个 in 元素的子代。

a
a 元素用于标识属性值。v 属性是必需的，用于指定属性值。

发送错误而不是数据

有时，发布套接字数据的应用程序可能无法收集属性组所必需的数据。在这种情况下，可以返回错误代码，而不是向代理程序发送数据。错误代码为您提供了如何将问题告知监视环境的方法。以下是一个错误示例：

```
<socketData><attrGroup name="abc"/><error rc="1000"/></attrGroup></socketData>\n
```

必须在代理程序中的一个所有套接字属性组公用的列表中定义错误代码。当代理程序收到错误代码时，将在代理程序日志文件中记录定义的错误消息。另外，名为 Performance Object Status 的属性组还包含一个“错误代码”属性，此属性将通过“错误代码类型”更新。该“错误代码类型”是针对发送的错误代码定义的。

对于上述示例，您必须已在代理程序中定义错误代码值 1000。请参阅以下错误代码定义样本：

表 *280.* 样本错误代码

错误代码值	错误代码类型	消息
1000	APP_NOT_RUNNING	The application is not running

发送此错误代码时，将在代理程序日志文件中记录一条类似于以下示例的消息：

```
(4D7FA153.0000-5:customproviderserver.cpp,1799,"processRC") Received error code 1000
 from client. \Message: K1C0001E The application is not running
```

如果从 Tivoli Enterprise Portal 选择了 Performance Object Status 查询，那么行 **abc** 属性组的**错误码**列将显示该表中的值 APP_NOT_RUNNING。

向采样属性组发送错误将清除先前针对该属性组收到的任何数据。向该属性组发送数据将导致此错误代码不再显示在 Performance Object Status 属性组中。另外，您也可以发送错误代码 0 以清除该表中的此错误代码。

向产生事件的属性组发送错误并不会从高速缓存中清除先前发送的事件。

处理“执行操作”请求

套接字客户机可以进行注册，以便在操作命令与特定前缀相匹配时接收来自代理程序的“执行操作”请求。任何不匹配的操作都由代理程序处理。该前缀不得与代理程序应该处理的操作相冲突，因此，请使用代理程序产品代码作为前缀。随 Agent Builder 一起提供的“执行操作”根据“执行操作”使用的数据源进行命名。例如，“执行操作”JMX_INVOKE 对 JMX 数据源进行操作。另一个示例是使用 SSH 脚本数据提供程序的“执行操作”SSHEXEC。由于这些操作不使用产品代码，因此，可以安全地将产品代码用作“执行操作”前缀。

套接字客户机必须长时间运行，并且必须保持套接字处于打开状态。它必须针对前缀发送注册请求，并侦听来自套接字的请求。即使数据未在流动，代理程序也能确保不会发生长时间运行客户机的套接字超时。以下是注册请求样本：

```
<taskPrefix value="K42"/>\n
```

在此样本中，代理程序收到的并以 "K42" 开头的任何“执行操作”命令将转发到发起注册的套接字客户机。以下样本显示套接字客户机可能收到的“执行操作”请求：

```
<taskRequest id="1"><task command="K42 refresh" user="sysadmin"/></taskRequest>\n
```

id 是一个唯一标识，代理程序用来跟踪发送到客户机的请求。当套接字客户机响应该任务时，它必须在 taskResponse 元素的 id 属性中提供此标识。

套接字客户机必须处理操作并发送响应。以下是一个响应样本：

```
<taskResponse id="1" rc="1"/>\n
```

如果操作成功完成，那么将返回 rc 属性值 0。rc 的值必须是整数，其中，任何非 0 值都将被视为失败。任务返回码值将记录到代理程序日志文件中，并显示在代理程序所包含的 Take Action Status 查询中。运行操作后在 Tivoli Enterprise Portal 上显示的对话框没有显示返回码。该对话框将指示“执行操作”命令是返回了成功还是失败。如果发生失败，您必须查看代理程序日志或 Take Action Status 查询以确定实际返回码。

代理程序开发人员负责记录、创建和导入与代理程序一起使用的套接字客户机所支持的任何操作。如果用户向套接字客户机发送了不受支持的操作，那么必须将客户机开发为能够以适当的方式处理这些场景。如果用户定义了更多以注册前缀开头的更多操作，这些操作将传递给客户机。必须将客户机开发为能够以适当的方式处理这些场景。

代理程序在等待来自套接字客户机的响应时，等待时间由超时控制。该设置是在代理程序中定义的环境变量，名为 CDP_DP_ACTION_TIMEOUT，其缺省值是 20 秒。

注: 为套接字数据源属性组定义的错误代码消息不用于“执行操作”。您可以返回相同的返回码值。但是，代理程序不会记录定义的消息，也不会影响 Performance Object Status 属性组中的错误代码字段。

套接字数据的编码

套接字客户机对发送到代理程序的数据进行编码。

了解套接字客户机对要发送至代理程序的数据进行编码的方式非常重要。

特殊字符

发送到代理程序的数据不得包含任何换行符（在每个事件或数据样本的末尾除外）。属性值中出现的换行符必须替换为其他字符或（第 1106 页的表 281）中编码的字符。另外，您还必须确保属性值不违反 XML 语法。下表显示属性值中进行编码的字符：

表 *281.* 属性值中要进行编码的字符

字符	头
&	&
<	<
>	>
“	"
'	'
\n

注: 代理程序使用换行符来分隔从客户机接收到的响应。意外出现的换行符导致无法正确地解析数据。

代理程序不包含全功能型 XML 解析器，因此您不得对不在（第 1106 页的表 281）中的字符使用特殊编码。例如，不要编码 ¢ 或 &cent 以代替分币符号 ¢。

字符集

除了要对特殊字符编码，代理程序还必须了解用于对数据进行编码的代码页。在定义每个套接字属性组时，应指出您是要将数据作为 **UTF-8** 数据还是**本地代码页**数据发送到代理程序。需了解客户机发送数据的方式。如果使用以 Java 编写的客户机，请在写程序中指定 **UTF-8** 编码以用于向代理程序发送数据。请指定 **UTF-8** 作为属性组的**代码页**。**本地代码页**表示代理程序的本地代码页。如果通过远程套接字来发送数据，那么必须遵循代理程序的本地代码页或者使用 UTF-8。

数字数据

应了解您的数字属性值的格式设置。发送到代理程序的数字值不得包含任何特殊字符。千位分隔符就是特殊字符的一个示例。其他示例包括货币符号或者用于描述值单位的字符。如果代理程序在解析数字数据时遇到问题，将记录错误以指示该问题。解析属性失败时，不会设置 Performance Object Status 错误代码。以下是代理程序日志中的错误消息示例：

```
(4D3F1FD6.0021-9:utilities.cpp,205,"parseNumericString") Invalid characters :00:04 \
found getting numeric value from 00:00:04, returning 0.000000
```

注: 有关时间戳记属性必须设置为何种格式的信息，请参阅（第 1008 页的『时间戳记』）。

套接字错误
对于从套接字客户机接收到的数据所发生的问题，系统会将对应错误写入代理程序日志文件。

记录的其他错误是返回非 0 值的执行操作。套接字客户机发送的错误值将随着错误代码的关联消息一起记录。

当套接字客户机向代理程序发送错误返回码时，会将属性组设置为 Performance Object Status。除代理程序定义的值以外，还会看到一些其他值。下表描述了您使用套接字属性组时可能会遇到的其他“错误代码”值:

表 *282. Performance Object Status* 值

错误代码	描述
NO_ERROR	未发生错误。表示属性组没有任何问题。一行采样数据出现问题并不会导致状态由 NO_ERROR 更改为其他值。即使看到错误代码 NO_ERROR，也必须验证显示的行数和属性值。
NO_INSTANCES_RETURNED	对于采样属性组，套接字客户机未发送数据行。并不是错误。这表示此属性组未监视任何资源实例。
XML_PARSE_ERROR	代理程序未能解析从客户机收到的数据。有关更多详细信息，请参阅代理程序日志。
OBJECT_CURRENTLY_UNAVAILABLE	客户机向代理程序发送了在错误代码的全局列表中未定义的错误代码。
GENERAL_ERROR	从客户机收集数据时发生问题，这通常是因为客户机未在超时时间间隔内应答请求。有关更多详细信息，请参阅代理程序跟踪日志。 另外，客户机也可以指定 GENERAL_ERROR 作为错误代码，但最好定义更详细的错误代码。

套接字配置
向代理程序添加套接字数据源之后，可以将代理程序配置为接受来自指定套接字端口的数据。

关于此任务

添加套接字数据源之后，配置将显示在 Agent Editor 的“**运行时配置**”页面上。套接字配置节包含以下属性:

表 *283.* 套接字配置属性

姓名	有效值	是否必需	描述
端口号	0 或任何正整数 缺省值为 0	是	这是代理程序用于侦听来自套接字客户机的数据的端口。值为 0 表示使用临时端口。

代理程序会将正在使用的端口值写入一个文件。以后，在代理程序计算机上运行的套接字客户机可以读取此文件以确定所要连接的端口。端口所写入的文件命名为 *kxx_instanceName*_cps.properties，其中 *kxx* 是代理程序的三字符产品代码，*instanceName* 是多实例代理程序的代理程序实例名。如果此代理程序不是多实例代理程序，那么名称中将不包含这一部分，因此文件名为 *kxx*_cp.properties。

在 Windows 中，此文件写入 32 位安装的 %CANDLE_HOME%\TMAITM6 目录，或者 64 位安装的 %CANDLE_HOME%\TMAITM6_x64。在 UNIX 中，此文件写入 /tmp。

过程

1. 可选： 如果您的系统有多个接口，将环境变量 CDP_DP_HOSTNAME 设置为您的网络接口的主机名或 IP 地址：
   a) 转至 Agent Editor 的“**代理程序信息**”视图并选择**环境变量**。
   b) 单击**添加**，然后使用“名称”字段从环境变量列表中选择 CDP_DP_HOSTNAME。
   c) 在 Value 字段中设置主机名或 IP 地址。
2. 启动代理程序。

   代理程序启动后，它将与 CDP_DP_HOSTNAME 环境变量所定义的接口绑定。如果未设置 CDP_DP_HOSTNAME，那么代理程序将与缺省主机名绑定。

   如果要将代理程序绑定到定义的端口，而不是临时端口，可以设置配置属性**端口号** (CP_PORT)。

   要设置端口号配置属性，请使用以下步骤：

   a) 转至 Agent Editor 的“**运行时配置**”视图。
   b) 在“**运行时配置信息**”窗格中，选择**套接字配置 > 套接字 > 端口号**。
   c) 在 **Default value** 中输入端口号值。
   如果不输入值，那么将使用值 0。值为 0 表示使用临时端口。

远程套接字端口连接

您可以将代理程序配置为接受来自远程套接字端口的数据。代理程序必须在一个与远程系统建立了网络接口连接的系统上运行。

过程

1. 通过完成以下步骤将环境变量 CDP_DP_ALLOW_REMOTE 的值设置为 YES。
   a) 转至 Agent Editor 的“**代理程序信息**”页面并选择**环境变量**。
   b) 单击**添加**，然后使用“名称”字段从环境变量列表中选择 **CDP_DP_ALLOW_REMOTE**。
   c) 将 **Value** 字段设置为 YES。
2. 遵循第 1107 页的『套接字配置』中详细描述的过程。

   限制:

   · 套接字应用程序和代理程序之间发送的数据：
   – 必须符合为套接字数据提供程序定义的 XML 语法。有关更多信息，请参阅第 1106 页的『套接字数据的编码』。
   – 必须以 UTF-8 编码。
   – 明文（未加密）发送。如果此数据包含敏感信息，必须在代理程序外部通过 SSH 通道或其他机制来保护通信安全。
   · 代理程序处理从任何远程主机收到的数据，因此必须使用适当的防火墙或网络流量过滤器保护环境安全。

结果

您可以在能够连接到代理程序运行所在系统的任何系统上运行实施套接字数据提供程序的代码。

套接字的样本脚本

此样本脚本演示如何编写套接字客户机。

Perl 样本

以下样本 Perl 脚本连接至套接字并发送数据。此样本针对在 UNIX 上运行的代理程序而编写，使用的产品代码为 k00，并且有一个名为 SocketData 的属性组。

```
#!/usr/bin/perl -w
# SocketTest.pl
# A simple Agent Builder Socket client using IO:Socket
#------------------------

use strict;
use IO::Socket;

# Initialize socket connection to the agent
#-----------------------
my $host = '127.0.0.1';
my $port = 0;
# This sample is for an agent with the k00 product code.  The product code is
# used in the following line to find the file containing the port number to use.
open PORTFILE, "/tmp/k00_cps.properties" || die "Port file not found $!\n";
while (<PORTFILE>) {
    if (/^CP_PORT=([0-9]+)/) {
        $port = $1;
    }
}

if ($port == 0) {
    die "Could not find port to use to connect to agent.\n";
}

my $sock = new IO::Socket::INET( PeerAddr => $host, PeerPort => $port,
Proto => 'tcp'); $sock or die "no socket :$!";

# The following call sends 2 rows of data to the agent.  Each row contains 1
# String attribute and 3 numeric attributes.
syswrite $sock, "<socketData><attrGroup name=\"SocketData\"><in><a v=\"A message
from perl\"/>  \<a v=\"1\"/><a v=\"2\"/><a v=\"123\"/></in><in><a v=\"More from
perl\"/><a v=\"456\"/>  \<a v=\"123\"/><a v=\"789\"/></in></attrGroup>
</socketData>\n";

close $sock;
```

测试套接字属性组

您可以对在 Agent Builder 中创建的套接字属性组进行测试。

开始之前

要测试此属性组，您需要一个用于发送数据的套接字客户机。可以在第 1109 页的『套接字的样本脚本』中查看使用 Perl 脚本编写的示例套接字客户机

限制: 与大多数其他属性组不同的是，您无法在创建期间测试套接字属性组。您可以在完成此属性组的创建工作后对其进行测试。

过程

1. 创建代理程序之后，请在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**测试套接字客户机**”窗口。
2. 可选： 在开始测试之前设置环境变量和配置属性。

   有关更多信息，请参阅第 1156 页的『属性组测试』。
3. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
4. 当代理程序启动时，会根据其配置侦听套接字数据。
5. 要测试代理程序的数据收集，您现在应该生成与代理程序配置匹配的套接字数据。

   您可以使用套接字客户机生成套接字数据。

当代理程序接收到与其配置匹配的套接字数据时，会将这些数据添加到其内部高速缓存。

6. 要模拟来自 Tivoli Enterprise Portal 的代理程序数据请求，请单击**收集数据**。
   “**测试套接字客户机**”窗口会收集并显示自代理程序上次启动以来，代理程序高速缓存中的全部数据。
7. 如果某些操作似乎未按预期工作，请单击**检查结果**。
   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据
8. 通过单击**停止代理程序**停止代理程序。
9. 单击**确定**或**取消**以退出“**测试套接字客户机**”窗口。单击**确定**将保存您所作的全部更改。

相关概念

第 1156 页的『在 Agent Builder 中测试代理程序』
使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

使用 Java API 来监视数据

您可以定义数据源以使用 Java API 与 Java 平台上长期运行的应用程序交互。代理程序在启动时启动应用程序并定期与其进行交互。在构建代理程序时，Agent Builder 创建应用程序的源代码。您必须定制代码以收集正确的数据。根据代码，数据源可以生成多个数据集，可包含一行、多行或事件数据。

关于此任务

使用 Java API 数据源和 Java 编程语言来收集使用其他 Agent Builder 数据源无法收集的数据。代理程序将启动此 Java 应用程序并在到时间关闭时发送关闭请求。此 Java 应用程序只能在接收到退出请求时退出。

包含 Java API 属性组的代理程序会与 Java 应用程序进程进行交互。Java 应用程序使用 Java 提供程序客户机 API 与代理程序进行交互。有关此 API 的信息，请参阅 Tivoli Monitoring Knowledge Center 上的 Javadoc。您可以使用 Java API 来执行下列操作：

- 连接到代理程序进程，并针对 Java 应用程序所支持的属性组进行注册
- 接收并应答对采样的数据发出的请求
- 以异步方式为生成事件的属性组发送数据
- 对于收集数据失败的属性组，发送错误
- 在配置了子节点实例的子节点中支持属性组
- 接收并回复“执行操作”请求

请使用以下过程来创建在 Java 应用程序中收集数据并使用 Java API 发送数据的属性组。该过程显示如何创建样本 Java 应用程序作为您自己的 Java 应用程序的基础。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面中，单击**监视数据类别**区域中的**定制程序**。
2. 在**数据源**区域中，单击 **Java API**。
3. 单击**下一步**。
4. 在 **Java API 信息**页面上，输入属性组名称。
5. 输入此属性组的帮助文本。
6. 选择这个属性组是**生成单一数据行**、**可生成多个数据行**还是**生成事件**。此选项将影响在向导末尾创建的样本 Java 应用程序。有关更多信息，请参阅第 1104 页的『发送数据』。
7. 可选： 单击**高级**，以修改此属性组的高级属性。当您选择属性组**可生成多个数据行**或**生成事件**时，**高级**可用。
8. 单击**下一步**。
9. 在“**特性信息**”页面上，指定此属性组的第一个特性。有关创建特性的更多信息，请参阅（第 1004 页的『创建属性』）。
10. 选择**添加其他特性**，并单击**下一步**以便向代理程序添加其他特性。对特性的引用将合并到在向导结束时创建的样本 Java 应用程序中。
11. 单击**下一步**。

12. 在**全局 Java API 数据源信息**页面上，输入类名及 JAR 文件名。

    这是一个标准类名，在 Java 启动时，将调用这个类的 main 方法。将使用此类中的主 Java 方法创建样本 Java 应用程序。

    此 JAR 文件是一个归档，它包含构成该 Java 应用程序的 Java 类。此 JAR 文件将随代理程序一起打包和安装。

13. 可选： 在“**全局 Java API 数据源信息**”页面上的**错误码**部分中，定义 Java 应用程序可以发送的错误码。这些错误码由 Java 应用程序在其无法收集数据时发送。

    限制: 错误代码限长 256 个字符。只允许使用 ASCII 字母、数字和下划线。不允许使用空格。

    a) 单击“错误码”部分中的**添加**。

    b) 在 **Java API 错误码定义**窗口中，输入显示值。

    c) 输入内部值。内部值必须是介于 1,000 与 2,147,483,647 之间的整数。

    d) 定义每个错误的消息文本。您可以通过从列表中进行选择来使用先前输入的消息文本。单击**确定**，以返回到**全局 Java API 数据源信息**页面。

    此消息将记录在代理程序日志文件中。

    e) 如果没有合适的可用消息文本，请单击**浏览**以设置消息文本。

    这将显示“消息”（列表）窗口。“消息”窗口列出代理程序中定义的消息。在定义消息之前，列表保留为空。您可以使用**编辑**来变更已定义的消息，并可以使用**移除**来删除一条或多条已定义的消息。

    f) 在“消息”（列表）窗口中，单击**添加**以查看“**消息定义**”窗口。在“**消息定义**”窗口中，您可以输入文本以描述新消息的含义并选择消息类型。

    注: 系统会为您自动生成消息标识。

    g) 单击**确定**。

    h) 这将显示包含新消息的“消息”（列表）窗口。要验证消息并返回到“**全局 Java API 数据源信息**”页面，请单击**确定**。

14. 可选： 在**全局 Java API 数据源信息**页面的**补充文件**部分中，可以添加文件，这些文件将与代理程序一起打包并在代理程序安装时复制到代理程序系统。Java 提供程序客户机 API JAR 文件未在此处列出；此文件自动复制到代理程序系统。**文件类型**列描述每个文件的预期用法。下表（第 1111 页的表 284）描述了三种可能的用法。单击**编辑**可以对导入的文件进行编辑。有关更多信息，请参阅第 1075 页的『编辑命令文件定义』。

表 *284.* 补充文件的文件类型

文件类型	**描述**
Executable	如果要随代理程序一起提供可执行文件，请选择此选项。代理程序不使用此文件，但此文件将保存在路径中供 Java 应用程序使用。
Library	如果要随代理程序一起提供库，请选择此选项。代理程序不使用此文件，但此文件将保存在库路径中供 Java 应用程序使用。
Java resource	如果要随代理程序一起提供 Java 资源，请选择此选项。代理程序不使用此文件，但此文件将保存在类路径中供 Java 应用程序使用。

    注: 当 Java 资源补充文件添加到 Agent Builder 时，该文件将自动添加到项目类路径中。Java 编译器使用这个补充文件将代码中包含的所有引用解析为资源中的类。

    有关补充文件随代理程序一起安装于何处的信息，请参阅第 1170 页的『系统上的新文件』。

15. 可选： 创建过滤器，以限制此属性组所返回的数据（如果对数据进行取样）。通过单击**高级**创建过滤器。

    注: 如果在“**Java API 信息**”页面上未选中**生成事件**，那么将对数据进行取样。

有关对来自属性组的数据进行过滤的更多信息，请参阅第 1011 页的『过滤属性组』。

16. 可选： 为子节点添加配置属性

    如果您正将此数据源添加到子节点，那么将显示“**子节点配置覆盖**”页面，因此您可以向该子节点添加配置属性。要创建样本 Java 应用程序，在子节点下需要至少一个配置属性。因为样本使用配置属性来区分不同的子节点实例，所以至少需要一个配置属性。

17. 执行下列其中一个步骤：

    a) 如果您正在使用“**代理程序**”向导，请单击**下一步**。根据需要完成向导。

    b) 否则，请单击**完成**，以保存该数据源并打开 Agent Editor。然后，在主菜单中，选择**文件 > 保存**。

    此时，Agent Builder 将创建监视应用程序的源代码。此代码位于项目目录的 `src` 子目录中。请编辑此代码，以创建监视应用程序。

下一步做什么

在“**Java API 设置**”页面上，选择正确的操作系统。如果要运行此属性组和 Java 应用程序的操作系统不同于您为代理程序定义的操作系统，请进行此选择。要打开此页面，请在大纲视图中单击 **Java API 设置**，或者在 Agent Editor 中的任何 Java API 属性组页面上单击**全局设置**。

注: 稍后，可以在 **Java API 设置**页面的**错误码**和**补充文件**部分中更新错误码及补充文件。

运行 Java 应用程序

有关 Java 应用程序及其依赖关系的初始化的信息

初始化 Java 应用程序

代理程序将在启动和初始化期间启动 Java 应用程序。配置设置用于控制使用哪个 Java 运行时来启动该进程。另外，您还可以在配置中指定 Java 虚拟机参数和 Java 日志记录级别。有关 Java API 配置的更多信息，请参阅 第 1119 页的『Java API 配置』。Java 进程将继承为代理程序定义的环境变量。运行时配置设置也被置于环境中，并且，您可以使用 API 调用来查询这些设置。

Java 应用程序必须是长时间运行的进程。除非它接收到来自 API 的关闭请求，否则不能终止。如果 Java 应用程序在向代理程序注册后终止，那么代理程序将尝试重新启动该 Java 应用程序，并且最多尝试三次。如果成功地恢复数据收集，那么此重新启动计数将重置。在 Java 应用程序终止时以及在发起重新启动时，代理程序将记录错误。

注: 如果 Java 应用程序在属性组注册完成之前终止，那么将不会尝试重新启动。

依赖项

Java 应用程序必须使用 Java 运行时环境。以下 Java 版本受支持：

- Oracle Corporation Java V5 或更高版本
- IBM Corporation Java V5 或更高版本

必须先在代理程序系统上安装 Java，然后再配置并启动代理程序。包含用于与代理程序进行通信的 API 的 JAR 文件将随代理程序运行时一起提供，并且包括在 JVM 的类路径中。必须将 Java 应用程序所需的任何其他 JAR 文件都定义为 Java API 属性组的补充文件。任何文件类型为 *Java* 资源的补充文件都将随 Java API JAR 文件一起自动添加到 Java 应用程序的基本类路径中。

任何未随代理程序一起提供但却是 Java 应用程序运行时操作所必需的 JAR 文件都必须包括在外部 *JAR* 的类路径配置设置中。

生成的样本 Java 应用程序

此参考资料描述 Agent Builder 生成的代码以及您必须针对所要监视的资源添加或替换的代码。

当您创建具有一个或多个 Java API 数据源的代理程序时，Agent Builder 将生成 Java 应用程序源代码。该代码将在代理程序项目中生成并遵循代理程序的结构。您必须向生成的应用程序添加自己的 Java 代码。您的代码收集采样属性组的数据、处理要向基于事件的属性组发布的事件、在遇到任何问题时报告错误以及运行任务。生成的应用程序为代理程序提供数据，但这是样本数据，将替换为从您要监视的资源获取的数据。

假设存在一个具有下列特征的样本代理程序：

- 产品代码：K91
- Java API 主类：agent.client.MainClass
- 第 1113 页的图 47 所示的代理程序数据源结构：

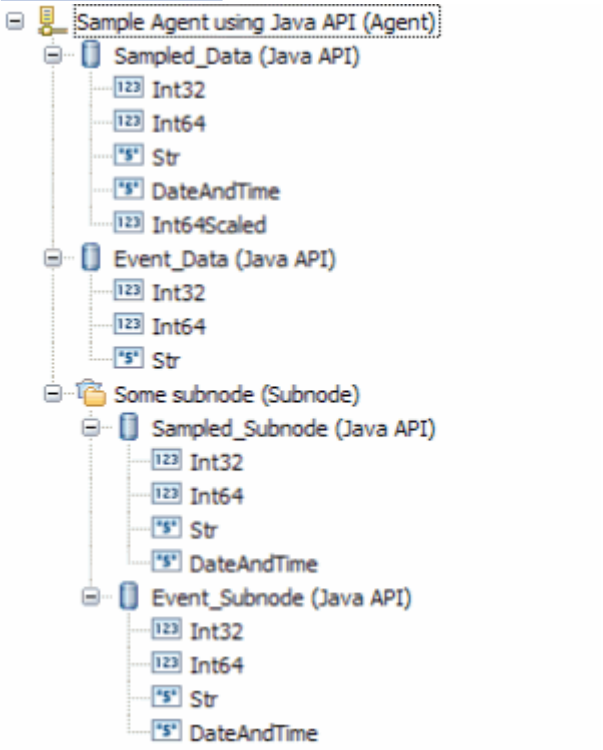


图 *47.* 样本代理程序结构

- 某个子节点配置属性：K91_INSTANCE_KEY

类结构

在生成的 Java 应用程序中，与代理程序进行交互的代码在很大程度上独立于与您所监视的资源进行交互的代码。它包含您修改的文件以及不修改的文件。

下列 Java 类由 Agent Builder 创建：

MainClass (agent.client 包)
您在“**全局 Java API 数据源信息**”页面上指定的类。这个类包含 main 方法以及用于处理执行操作请求的方法。这个类继承自下面描述的帮助程序类。您必须修改这个类，以便与要监视的资源以及要执行的操作进行交互。

MainClassBase (agent.client 包)
这是一个帮助程序类，用于初始化服务器连接、注册属性组并等待来自服务器的请求。请不要修改这个类。

Sampled_Data、Sampled_Subnode、Event_Data 和 Event_Subnode 类 (agent.client.attributeGroups 包)
每个 Java API 属性组都有一个类，用于处理该属性组的数据收集请求或者为该属性组生成事件。这些类中的每个类都继承自下面描述的其中一个帮助程序类。您必须修改这些类，以便从要监视的资源收集数据。

Sampled_DataBase、Sampled_SubnodeBase、Event_DataBase 和 Event_SubnodeBase 类 (agent.client.attributeGroups 包)
这些是帮助程序类（每个 Java API 属性组都有一个对应的帮助程序类），用于定义组属性在内部类中的结构。请不要修改这些类。

ICustomAttributeGroup 接口 (agent.client.attributeGroups 包)
此接口用于定义每个属性组类中的公用方法。请不要修改此接口。

您可以修改的类永远不会被 Agent Builder 覆盖。仅当这些类不存在时，Agent Builder 才会创建这些类。

每次保存 Agent Builder 时，帮助程序类和接口都将被覆盖。对代理程序进行更改并保存该代理程序时，帮助程序类将更新以反映对 Java API 属性组所作的任何结构更改。接口和帮助程序类的头部包含警告，提醒您不要修改该文件。

初始化与清除

代理程序启动时，将调用 `MainClass` 中的 main 方法。此方法将创建 `MainClass` 实例，然后进入用于接收并处理代理程序请求的长时间运行方法。

大部分初始化和清除代码必须添加到 `MainClass` 中。在构造函数中，请添加创建或访问资源所需的初始化代码。您可能想打开远程资源连接、创建句柄或初始化数据结构。

在代理程序终止之前，将调用 `stopDataCollection` 方法。如果要在 Java 应用程序结束前关闭连接或执行清除操作，请在 `stopDataCollection` 方法中添加相应的代码。

如果只需要针对特定属性组进行初始化，那么可以在该属性组类的构造函数中添加初始化代码。同样，如果只需要针对特定属性组执行任何清除操作，那么可以在该属性组的 `stopDataCollection` 方法中添加相应的清除代码。

Java 应用程序中的任何代码都可以使用记录器对象来写入日志条目。（主帮助程序类将在它的构造函数中创建一个受保护的记录器对象。属性组帮助程序对象将在它们的构造函数中创建对该记录器的受保护引用）。此记录器对象将使用 Java 跟踪日志实用程序。可以从记录器所创建的跟踪日志中获取错误及详细跟踪信息。跟踪信息对于诊断提供程序存在的问题而言非常重要。

在调用 `stopDataCollection` 时，如果将清除工作传递给另一个线程，请先等待该线程完成，然后再从 `stopDataCollection` 方法返回。否则，在进程结束时，清除工作可能会由于主线程已完成而突然终止。

在代理程序配置设置中，其中一个是 Java 跟踪级别设置。下表显示可以在 `JAVA_TRACE_LEVEL` 配置属性中设置的值。如果 API 创建了记录器，那么下表将显示该记录器使用的级别。

表 *285. Java* 跟踪级别选项

配置的跟踪级别	**Java 日志记录跟踪级别**	**描述**
关	OFF	不执行任何日志记录。
错误	SEVERE	跟踪 Java 应用程序中发生的问题。
警告	WARNING	跟踪错误和潜在错误。
参考	INFORMATION	跟踪关于 Java 应用程序的重要信息。
最低级别调试	FINE	跟踪分析 Java 应用程序行为所必需的高级别详细信息。
中等级别调试	FINER	跟踪关于 Java 应用程序的程序流的详细信息。
最高级别调试	FINEST	跟踪关于 Java 应用程序的所有详细信息。
全部	ALL	跟踪所有消息。

在此示例中，Java 应用程序创建的日志文件的名称为 `k91_trace0.log`。对于多实例代理程序，实例名称将包括在日志文件名中。

注: 请不要将消息写至标准错误或标准输出。在 Windows 系统上，这些消息将丢失。在 UNIX 和 Linux 系统上，此数据将写入一个不回绕的文件。

收集采样属性组数据

采样属性组（收集一个或多个数据行的属性组）的类包含 `collectData` 方法，例如 `Sampled_Data.collectData`。每当代理程序请求获取数据时，都将调用此方法。

属性组的帮助程序类定义了一个名为 Attributes 的内部类。对于该属性组中定义的每个属性，这个类都有一个对应的字段。由于派生属性由代理程序进行计算，因此未包括在内。属性字段的数据类型是 Tivoli Monitoring 属性类型的 Java 对等项，如第 1115 页的表 286 所示。

表 *286.* 属性字段的数据类型及其 *IBM Tivoli Monitoring* 属性类型对等项

Tivoli Monitoring 类型	**属性字段的数据类型**
字符串	String
32 位数字（不进行小数调整）	int
64 位数字（不进行小数调整）	long
数字（非零小数调整）	double
时间戳记	Calendar

`collectData` 方法必须完成下列任务：

1. 从受监视的资源收集相应的数据。
2. 创建 Attributes 对象。
3. 将数据添加到该 Attributes 对象的字段。
4. 调用 `Attributes.setAttributeValues` 方法，以便将数据复制到内部缓冲区。
5. 根据需要，对每个数据行重复第 1 步到第 4 步。（可以同时跳过第 1 步到第 4 步，而不返回任何行。在这种情况下，Performance Object Status 表的“错误码”列的值为 NO_INSTANCES_RETURNED。有关错误码的更多信息，请参阅第 1117 页的『错误码』。
6. 调用 `AgentConnection.sendData` 以便将数据发送到代理程序，或者调用 `sendError` 以废弃通过调用 `setAttributeValues` 复制的数据并改为发送错误码。

您必须从资源收集数据（第 1 步），从而替换在生成的应用程序中使用的样本数据。

要填充该 Attributes 对象，您可以使用 Attributes 构造函数来传递数据（正如生成的应用程序中的做法）。另外，也可以使用不带自变量的构造函数来创建 Attributes 对象，然后将收集到的属性值赋予该 Attributes 对象的字段。字段与属性同名，但字段名称以小写字母开头。

收集子节点的采样数据

如果采样属性组位于子节点中，那么您可能正在监视多个资源（对于每个子节点，都存在不同的资源）。您必须确定要从哪些资源收集数据。必须有一个或多个用于标识所监视资源的配置属性。

对于本示例，假定配置属性 `K91_INSTANCE_KEY` 中包含的值指示必须从中收集数据的资源。

请通过下列步骤来查找正确的资源：

1. 通过调用 `AgentConnection.getConfiguredSubnodeInstanceIDs`，获取所有已配置的子节点的实例标识。每个已配置的子节点都具有唯一的实例标识。
2. 对于每个实例标识，通过调用 `AgentConnection.getSubnodeConfigurationProperty` 获取配置属性 `K91_INSTANCE_KEY`。
3. 找到 `K91_INSTANCE_KEY` 中的值所表示的资源。

这些步骤可以在第 1115 页的『收集采样属性组数据』中详细描述的步骤序列之前，在 `collectData` 方法中完成。

另外，您也可能想在属性组类构造函数中执行这些步骤，并建立从实例标识到资源的直接映射。`Sampled_Subnode` 构造函数是属性组类构造函数的一个示例。此过程还使您有机会创建句柄或打开可以在代理程序的整个生命周期内使用的连接。创建句柄或打开连接可以更有效地访问资源。

生成的代码将在构造函数中创建类型为 MonitoredEntity 的样本资源对象，并将这些对象添加至 configurationLookup 映射。您必须移除 MonitoredEntity 内部类，并将 MonitoredEntity 对象替换为用于访问您自己的资源的对象。如果您选择在 collectData 方法中执行整个查找过程，那么可以从该类中移除 configurationLookup 映射。

如果您选择使用构造函数，从而将子节点实例标识映射到资源，那么 collectData 方法中的步骤如下所示：

1. 通过调用 Request.getSubnodeInstanceID，从请求参数中检索子节点的实例标识。
2. 从构造函数中创建的映射中检索资源对象。
3. 执行第 1115 页的『收集采样属性组数据』中详细描述的步骤序列，以便将数据发送到代理程序。

在 Agent Builder 示例中选择了任意子节点属性（在本例中，这是 K91_INSTANCE_KEY）。如果选择了错误的属性，或者需要多个属性才能标识正确的资源，那么您必须选择这些属性以标识该资源。

发送事件

对于生成事件的属性组而言，不会定期调用 collectData 方法。在资源发布事件时，您的应用程序将发送这些事件。

下面是生成事件的一个示例：针对基于事件的属性组生成的代码创建并启动一个线程，该线程从名为 SampleEventClass 的内部类中运行。此示例中使用的基于事件的属性组为 Event_Data 类。此线程将定期唤醒并发送事件。如果您希望定期轮询资源以查找事件，那么可以使用 Event_Data 类在生成时具有的结构：

1. 在 Event_Data 构造函数中，创建并启动一个线程。
2. 在此线程的 run 方法中，进行循环，直到代理程序终止为止。
3. 休眠一段时间，然后查找事件。您可能想将轮询时间间隔 5000 毫秒更改为对代理程序有意义的值。
4. 确定是否发生了一个或多个事件。生成的应用程序不会检查是否发生了事件，而会始终发布单一事件。
5. 对于每个必须发布的事件，获取要发布的事件数据。
6. 创建并填充 Attributes 对象（与 collectData 方法对采样属性组进行的处理相似）。
7. 调用 Attributes.sendEventData 方法。事件由单一的行组成，因此，每次只能发送单一事件。

另外，如果您正在使用通过自己的线程来报告事件的 Java API，那么可以在 Event_Data 构造函数中初始化该线程。您还可以向资源的事件处理机制注册您自己的事件处理对象。在事件处理程序中，使用以下步骤：

1. 获取要发布的事件数据。
2. 创建并填充 Attributes 对象。
3. 调用 Attributes.sendEventData 方法。

在这种情况下，不必在 Event_Data 类中创建自己的线程，也不需要 SampleEventClass 类。

在子节点中发送事件

检测到子节点属性组的事件时，Java 应用程序必须将该事件发布到正确的子节点。

在本示例中，假定配置属性 K91_INSTANCE_KEY 中包含的值指示可以生成事件的资源实例。并且，假定 K91_INSTANCE_KEY 属性的值与要在事件中发布的数据一起检索。要检索属性和数据，Java 应用程序应执行以下步骤：

1. 获取要发布的事件数据以及“实例键”。
2. 创建并填充 Attributes 对象。
3. 通过调用 AgentConnection.getConfiguredSubnodeInstanceIDs，获取所有已配置的子节点实例标识的列表。
4. 对于每个子节点实例，通过调用 AgentConnection.getSubnodeConfigurationProperty 访存 K91_INSTANCE_KEY 的值。

5. 发现 K91_INSTANCE_KEY 的值与随事件数据获取的值匹配时，记录相应的子节点实例标识。
6. 在传递所记录的子节点实例标识的情况下，调用 Attributes.sendSubnodeEventData。

生成的应用程序不会执行第 4 步和第 5 步中描述的查找操作，而是将事件发布到每个子节点的属性组。对于生产代理程序而言，这可能不是正确的行为。

执行操作命令

您可以在 Tivoli Enterprise Portal 中定义执行操作命令，也可以使用 tacmd createaction 命令来定义这些操作。并且，可以将这些操作导入到代理程序的 Agent Builder 项目中，以便在安装代理程序时创建这些操作。有关导入执行操作命令的更多信息，请参阅第 1179 页的『导入应用程序支持文件』。

生成的 Java 应用程序将针对任何以代理程序产品代码开头的操作（例如 K91Refresh）进行注册。此注册在主帮助程序类 (MainClassBase) 的 registerActionPrefix 方法中完成。如果要注册其他前缀或者完全不针对任何操作进行注册，请覆盖 (MainClassBase) 中的 registerActionPrefix。

当代理程序要运行以代理程序所注册的前缀开头的操作时，将调用 MainClass.takeAction 方法。您可以添加代码以调用 Request.getAction() 并执行适当的操作，然后调用 AgentConnection.sendActionReturnCode 以发送操作返回码。返回码为 0 表示操作成功，任何其他返回码均表示操作失败。

处理异常

collectData 和 takeAction 方法可以抛出任何 Java 异常，因此，您可以允许收集代码抛出异常而不捕获这些异常。当帮助程序类接收到异常时，将调用 handleException 方法（对于 collectData）或 handleActionException 方法（对于 takeAction）。

对于 collectData 异常，您必须在异常发生时或者在收集数据期间出现任何问题时调用 AgentConnection.sendError。生成的应用程序传递错误码 GENERAL_ERROR。但是，您必须将此错误码替换为由代理程序定义且最能说明所遇到问题的错误码。有关添加错误码的更多信息，请参阅第 1111 页的『13』。

对于 takeAction 异常，您必须调用 AgentConnection.sendActionReturnCode，且使用非零返回码。

某些 AgentConnection 方法会抛出派生自 com.ibm.tivoli.monitoring.agentFactory.customProvider.CpciException 的异常。如果在收集数据期间抛出了 CpciException，那么不会调用 handleException 方法，这是因为帮助程序类将处理该异常。

注: 如果您选择在 collectData 方法内捕获异常，而不使用 handleException 方法，请确保重新抛出任何 CpciException。确保重新抛出 CpciException 以使其可以由基类进行处理。

错误码

对异常或其他资源错误的典型响应是，通过调用 AgentConnection.sendError 方法将错误码发送到代理程序。基于事件的属性组的错误随时都可以发送。采样属性组的错误只能作为对收集数据请求的响应进行发送，并且替代 sendData 调用。

如果将错误发送到代理程序，那么将发生下列情况:

1. 错误消息将记录在代理程序跟踪日志中。此错误消息包含错误码以及针对该错误码定义的消息。
2. 存在 Performance Object Status 查询，您可以查看此查询以获取属性组的状态信息。“错误码”列将设置为针对所发送错误定义的错误码类型。在代理程序成功接收到此属性组的数据后，错误状态将清除。如果您使用 sendData 调用来应答收集数据请求，但未包括任何数据行，那么“错误码”列将显示 NO_INSTANCES_RETURNED。

下表描述了一些在某些情况下有可能出现的代理程序内部错误码:

表 *287.* 代理程序的内部错误码	
错误码	**描述**
NO_ERROR	此属性组目前没有任何问题。
NO_INSTANCES_RETURNED	Java 应用程序已响应数据收集请求，但未提供任何数据。未提供数据不是错误。通常，这表示此属性组未监视任何资源实例。
OBJECT_NOT_FOUND	代理程序试图收集未通过客户机 API 注册的属性组的数据。此错误可能表示应用程序未能启动，或者应用程序在代理程序尝试收集数据时未启动属性组注册。
OBJECT_CURRENTLY_UNAVAILABLE	应用程序向代理程序发送了未在全局错误码列表中定义的错误码。
GENERAL_ERROR	从应用程序收集数据时发生问题，这通常是因为应用程序未在超时时间间隔内对请求作出应答。有关更多详细信息，请参阅代理程序跟踪日志。 另外，应用程序也可以指定 GENERAL_ERROR 作为错误码，但最好定义更详细的错误码。

对代理程序所作的更改

对代理程序所作的某些更改要求对 Java 应用程序进行相应更改。如果结构更改非常复杂，那么可以先删除任何或全部 Java 源文件，然后再保存代理程序。如果您想在未进行定制的情况下重新开始，那么也可以删除这些文件。

下表描述了在 Agent Builder 中进行某些更改之后，在保存该代理程序时需要对 Java 应用程序源文件进行的修改。

表 *288.* 要求对 *Java* 源代码进行修改的代理程序更改		
代理程序更改	**Agent Builder 执行的操作**	**需要对 Java 源代码进行的手动更改**
更改主类的包名	· 在新的包结构中生成所有的类。 · 从旧包中移除所有的帮助程序类。	· 将主类及属性组类的内容从旧包中的类移植到新包中的类。 · 在迁移完成后，从旧包中移除这些类。
更改主类的名称	· 创建新的主类。 · 移除旧的主帮助程序类。	· 将主类的内容移植至新类。 · 更新属性组类中对类名的引用。
添加 Java API 属性组	· 针对新的属性组创建类。 · 在主帮助程序类中，针对新的属性组添加注册。	使用属性组类中的定制逻辑覆盖样本代码。
移除 Java API 属性组	从主帮助程序类中移除注册。	· 移除属性组类，或者将定制逻辑移植到另外某个类。 · 移除属性组帮助程序类。

表 *288.* 要求对 *Java* 源代码进行修改的代理程序更改 *(续)*

代理程序更改	Agent Builder 执行的操作	需要对 Java 源代码进行的手动更改
重命名 Java API 属性组	· 针对属性组的新名称创建类。 · 在主帮助程序类中，更新已重命名的属性组的注册。	· 将具有旧名称的属性组类中的定制逻辑移植至具有新名称的属性组类。 · 移除具有旧名称的属性组类。 · 移除具有旧名称的属性组帮助程序类。
向 Java API 属性组添加属性	在属性组帮助程序类中更新 Attributes 内部类。	在属性组类中收集新属性的数据。
从 Java API 属性组中移除属性	在属性组帮助程序类中更新 Attributes 类。	在属性组类中移除先前属性的数据收集。
重命名 Java API 属性组中的属性	在属性组帮助程序类中更新 Attributes 类中的属性名称。	更新 Attributes 类中的所有对该属性名称的引用（由于使用了具有位置参数的 Attributes 构造函数，因此通常不存在任何引用）。
对 Java API 属性组中的属性进行重新排序	在属性组帮助程序类中更新 Attributes 类中的属性顺序。	更新全部 Attributes 构造函数调用中的自变量顺序。

如果使用“Eclipse 重构 - 重命名”操作，那么上表提及的某些更改可以简化。请先对所有受影响的名称（包括帮助程序类名）使用此操作，然后再保存所更改的代理程序。

Java API 的使用

在生成的 Java 应用程序中，将始终使用 Java API 与代理程序进行通信。通常，您与 Java API 进行的唯一直接交互是修改现有方法调用的参数。例如，将发布的错误码由 GENERAL_ERROR 更改为代理程序中定义的错误码。

如果要使用 Java API 进行更广泛的编码，您可以从 Eclipse 文本编辑器中查看 Javadoc。在编辑 Java 代码时，可以通过执行下列步骤查看 Javadoc：

1. 突出显示 API 中的包名、类名或方法名。
2. 按下 **F1** 以打开 Eclipse 帮助视图。
3. 选择 Javadoc 链接。

您也可以通过将鼠标悬停在类名或方法名上查看 Javadoc 中的简要描述。此 API 的 Javadoc 也可以在 Tivoli Monitoring Knowledge Center 上找到，请参阅 Javadoc。

Java API 的类包含在 `cpci.jar` 中。在创建包含 Java API 属性组的代理程序时，`cpci.jar` 文件将自动添加到项目的 Java 构建路径。在导入包含 Java API 属性组的代理程序时，也将添加该文件。将 Java API 属性组添加到现有代理程序时，也将添加该文件。另外，`cpci.jar` 还将自动地与每个包含 Java API 属性组的代理程序打包在一起，并添加到 Java 应用程序的 CLASSPATH。

Java API 配置

在代理程序中定义 Java API 数据源时，将为您创建部分配置属性。

如果在代理程序中定义了 Java API 数据源，那么该代理程序必须使用 Java 来连接到 Java API 服务器。Java 配置属性将自动添加到该代理程序中。下列 Java 配置属性特定于代理程序运行时配置：

表 289. Java 配置属性			
姓名	**有效值**	**是否必需**	**描述**
Java 主目录	目录的标准路径	否	指向 Java 安装目录的标准路径。
Java 跟踪级别	选项	是	请使用此属性来指定 Java 提供程序所使用的跟踪级别。
JVM 自变量	字符串	否	请使用此属性来指定 Java 虚拟机的可选自变量列表。
外部 JAR 文件的类路径	字符串	否	包含所有未随代理程序一起提供但却是运行时客户机操作所必需的 JAR 文件的路径。

这些配置变量位于 Agent Editor 中“**运行时配置信息**”页面上的 **Java 虚拟机 (JVM) 配置**和 **Java API 配置**之下。

测试 Java 应用程序属性组
您可以对在 Agent Builder 中创建的 Java 应用程序属性组进行测试。

开始之前

限制: 与大部分其他属性组不同，您在创建 Java 应用程序属性组时，无法对其进行测试。您可以先将此属性组添加到代理程序并保存代理程序，然后再测试此属性组。保存代理程序将致使生成此属性组的 Java 代码。

过程

1. 创建代理程序之后，请在 **Agent Editor** 的“**数据源定义**”页面上选择属性组，然后单击**测试**。

   有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』

   在上述两个步骤中的某个步骤中单击**测试**之后，将显示“**测试 Java 客户机**”窗口。
2. 可选： 在开始测试之前，请设置环境变量、配置属性及 Java 信息。有关更多信息，请参阅第 1156 页的『属性组测试』。有关缺省 Java 运行时配置属性的更多信息，请参阅 第 1119 页的『Java API 配置』。
3. 单击**启动代理程序**。此时将显示一个窗口，表明代理程序正在启动。
4. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求，请单击**收集数据**。
   代理程序会监视 Java 客户机的数据。“**测试 Java 客户机**”窗口将显示所返回的全部数据。
5. 可选： 如果返回的数据未如预期，请单击**检查结果**。
   “**数据收集状态**”窗口将会打开，并显示有关该数据的更多信息。第 1195 页的『Performance Object Status 节点』 对 Data collection Status 窗口所收集并显示的数据作了描述。
6. 通过单击**停止代理程序**停止代理程序。
7. 单击**确定**或**取消**以退出“**测试 Java 客户机**”窗口。单击**确定**将保存您所作的全部更改。

相关概念
第 1156 页的『在 Agent Builder 中测试代理程序』

使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

根据现有来源创建数据集

至少存在一个数据集时，您可使用现有数据集中的数据来创建新数据集。

“**代理程序初始数据源**”页面和“**数据源位置**”页面都提供了创建新数据集的选项。您可以通过下列方法使用现有数据源来创建数据集：

1. 连接来自两个现有数据集（属性组）的数据。有关更多信息，请参阅第 1121 页的『连接两个属性组』。
2. 过滤来自某个现有数据集（属性组）的数据。有关更多信息，请参阅第 1125 页的『创建过滤属性组』。

提示: 只有在创建两个或两个以上数据集之后，连接两个数据集的选项才可用。

连接两个属性组

从两个其他属性组创建一个属性组。

关于此任务

代理程序从两种不同类型的数据源收集数据时，连接式属性组非常有用。例如，代理程序可以收集数据 WMI 和 PerfMon，或 SNMP 和脚本数据源。各组属性在一个 Tivoli Enterprise Portal 视图中搭配使用时可能更加有用。

例如，假设您的属性组定义如下：

```
First_Attribute_Group
    index integer
    trafficRate    integer
    errorCount    integer
```

```
Second_Attribute_Group
    index2    integer
    name string
    traffic string
```

一个定义向您提供了计数器（例如：Perfmon），另一个向您提供了识别信息。没有属性组本身对您有用。但是，如果您可以使用索引将这两个属性组组合起来以与每个属性组的相应行进行匹配，那么会得到一个更有用的属性组。您可以使用组合的属性组来同时显示名称、类型和度量。

此机制还能用于将标记添加到通过普通属性组收集的信息内。然后在检测到问题时能够在事件系统中轻松地与信息进行关联。例如，某公司想通过收集公共数据，并使用公共情境来监视服务器的运行状况，从而管理其所有服务器。此公司还希望能通过更多的信息（指示正在特定服务器上运行哪些应用程序）来识别服务器。此公司想要控制每台服务器上使用的值，但是不想为各个应用程序创建不同的代理程序。它可以在其单个代理程序中创建一个附加属性组来实现此控制，如下所示：

```
Application_Information
    application_type     integer
    application_name    string
    application_group    string
```

此属性组可定义为脚本属性组，用于从代理程序配置收集值。您可以为每个代理程序实例指定不同的值，并使用一个代理程序来管理他们的所有系统。此属性组随即将连接至所有可能需要该应用程序信息的源属性组。然后该信息将在 Tivoli Enterprise Portal、情境、事件和已仓储数据中可用。

连接两个属性组时，将创建第三个属性组。此属性组包含源属性组中包含的所有属性。

连接操作的结果会因各个源属性组支持的行数不同而不同。如果两个属性组都定义为只返回一行数据，那么生成的连接式属性组将有一行数据。单个行包含来自这两个源属性组的所有属性。

表 *290.* 源属性组 *1*（单行）

属性 1	**属性 2**	**属性 3**
16	部分文本	35

表 *291.* 源属性组 *2*（单行）

属性 4	**属性 5**	**属性 6**	**属性 7**
5001	更多数据	56	35

表 *292.* 生成的连接

属性 1	**属性 2**	**属性 3**	**属性 4**	**属性 5**	**属性 6**	**属性 7**
16	部分文本	35	5001	更多数据	56	35

假设一个源属性组定义为只返回一行（单行），而另一个源属性组可返回多行（多行）。生成的连接式属性组包含与多行源属性组相同的行数。来自单行属性组的数据将添加到多行属性组中的各行。

表 *293.* 源属性组 *1*（单行）

属性 1	**属性 2**	**属性 3**
16	部分文本	35

表 *294.* 源属性组 *2*（多行）

属性 4	**属性 5**	**属性 6**	**属性 7**
user1	path1	56	35
user2	path2	27	54
user3	path3	44	32

表 *295.* 生成的连接

属性 1	**属性 2**	**属性 3**	**属性 4**	**属性 5**	**属性 6**	**属性 7**
16	部分文本	35	user1	path1	56	35
16	部分文本	35	user2	path2	27	54
16	部分文本	35	user3	path3	44	32

最后，假设两个源属性组都定义为返回多行。您必须识别要执行连接操作的每个源属性组的属性。生成的属性组将包含多个数据行，其中，第一个属性组的属性值与第二个属性组的属性值相匹配。

表 *296.* 源属性组 *1*（多行）

属性 1	**属性 2**	**属性 3**
16	部分文本	35
27	更多文本	54
39	其他字符串	66

表 *297.* 源属性组 *2*（多行）

属性 4	**属性 5**	**属性 6**	**属性 7**
user1	path1	56	35
user2	path2	27	54
user3	path3	44	32

表 298. 生成的连接（按属性 3 和属性 7 连接）						
属性 1	**属性 2**	**属性 3**	**属性 4**	**属性 5**	**属性 6**	**属性 7**
16	部分文本	35	user1	path1	56	35
27	更多文本	54	user2	path2	27	54

如果代理程序中定义了任何可用性过滤器，还可以使用 Agent Builder 将用户定义的属性组连接到“可用性”属性组。有关“可用性”属性组中包含的数据的更多信息，请参阅第 1190 页的『可用性节点』。

通过单击右键访问数据源树的菜单并选择**连接式属性组**，您可以创建此类型的属性组。

过程

1. 在“**数据源定义**”页面上，右键单击其中一个要连接的属性组，然后选择**连接式属性组**。
   仅当至少定义了两个属性组时才显示该选项。定义了可用性过滤器即认为是定义了属性组。
   “**属性组信息**”页面随即显示。

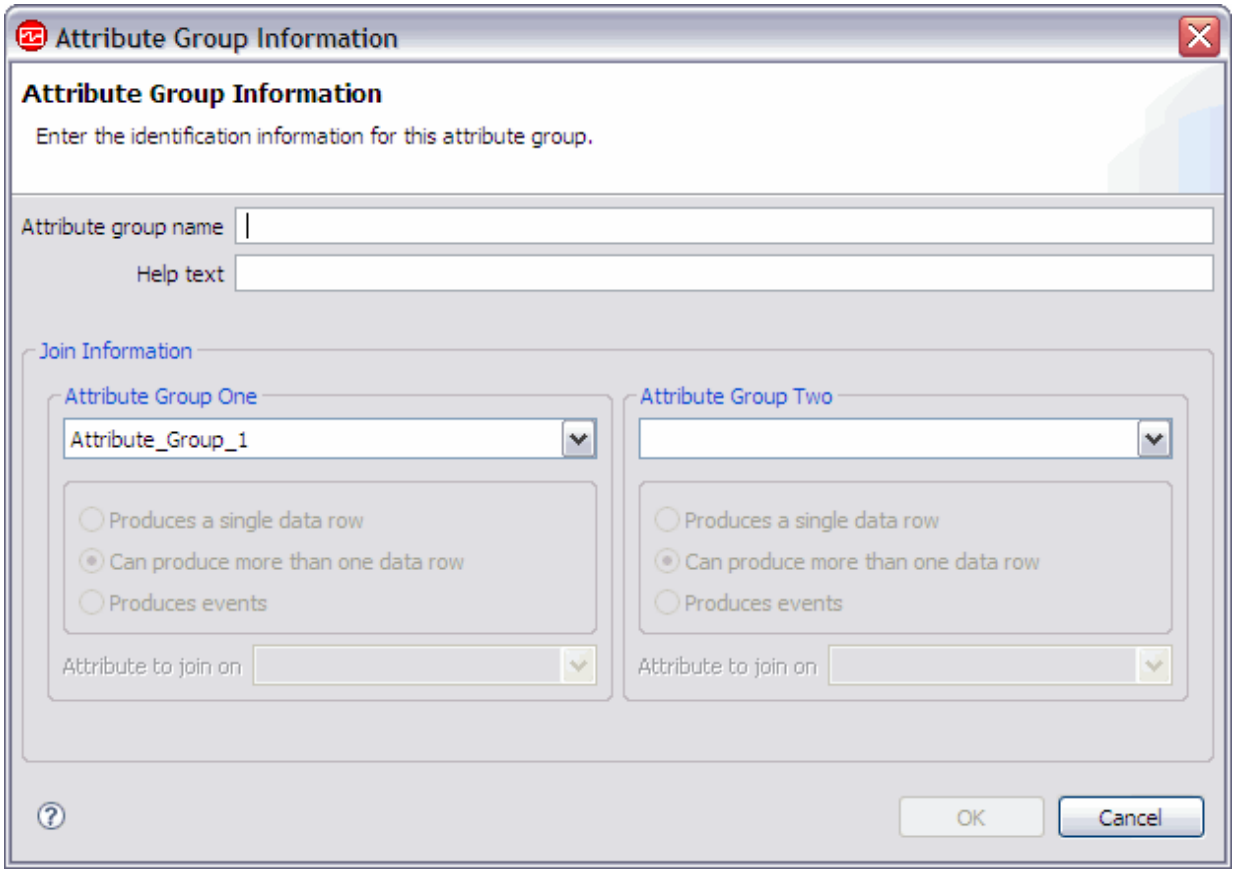


图 48. “**属性组信息**”页面“属性组信息”窗口

2. 在“**连接信息**”区域中，选择要连接的两个属性组。通过从**属性组一**和**属性组二**列表中的群组中进行选择，选择属性组。
   对于每个属性组，系统将为您选择**生成单一数据行**或**可生成多个数据行**。此选择已锁定并取决于最初如何定义这些源属性组。

   注: 对于可以连接式属性组有以下限制：

   · 不能将一个子节点类型中的属性组连接到另一个子节点类型中的属性组。

   · 只能将事件属性组连接到单行非事件属性组。

   a) 当两个属性组在**连接所依据的属性**下都显示了**可生成多个数据行**时，请为每个属性组选择要用作连接依据的属性。

属性组名称和**帮助**字段将使用所选属性组的信息进行填写。您可以根据需要来更改这些条目。

3. 单击**确定**。

结果

您创建的连接式属性组将添加到“**数据源定义**”页面的“**属性组信息**”区域中。

处理连接式属性组中的属性

使用连接式属性组中的属性可以对如何处理这些属性实施规则。

删除属性组

如果某个属性组在连接式属性组中被引用，那么除非连接式属性组也要被删除，否则不能删除该属性组。

删除属性

如果某个属性的父属性组在连接式属性组中被引用，并且符合以下某项，则无法删除该属性：

· 该属性被定义为连接式属性组中的连接式属性。

· 该属性在连接式属性组中的任何派生属性中使用。

无法删除连接式属性。只能从连接式属性组删除派生属性（如果添加了这些派生属性）。

重新排序属性

连接式属性的顺序固定为源属性的顺序。无法对连接式属性列表重新排序。只能对派生属性（如果有）重新排序。

落实代理程序的版本后，源属性和派生属性就无法重新排序，并且不可移除。添加在新版本的代理程序中的属性（源或派生属性）将跟在所有落实的属性之后。有关更多信息，请参阅第 1002 页的『落实代理程序的版本』。

添加属性

无法显式添加新的连接式属性。只能显式创建派生属性。

移除可用性过滤器

如果在连接式属性组中引用了“可用性”属性组，那么不能移除最后一个可用性过滤器。

连接式属性

处理与连接式属性有关的信息

过程

· 您可以更改连接式属性的属性名称和帮助文本，以使其与源属性不同：

  a) 在**数据源定义**页面的**属性组信息**窗格中，选择连接式属性组中的属性。

  b) 输入新名称和帮助文本。

· 通过选中或取消选中**在 Tivoli Enterprise Portal 中显示属性**复选框，可以使连接式属性显示或不显示在 Tivoli Enterprise Portal 中。此复选框位于**数据源定义**页面的**连接式属性信息**部分中。此选项与源属性是否显示在 Tivoli® Enterprise Portal 中无关。

· 通过选中**键属性**复选框，可以将 Tivoli Enterprise Portal 中显示的任何属性或属性组合标记为键属性。此选项与属性是否是源属性组中的键属性无关。此选项与源属性是否显示在 Tivoli® Enterprise Portal 中也无关。

· 连接式属性的属性类型信息是从源属性取得并且无法在连接式属性中更改。在 Agent Editor 的 **连接式属性组信息**部分（第 1125 页的图 49）中，单击**找到源属性**即可转到源属性。

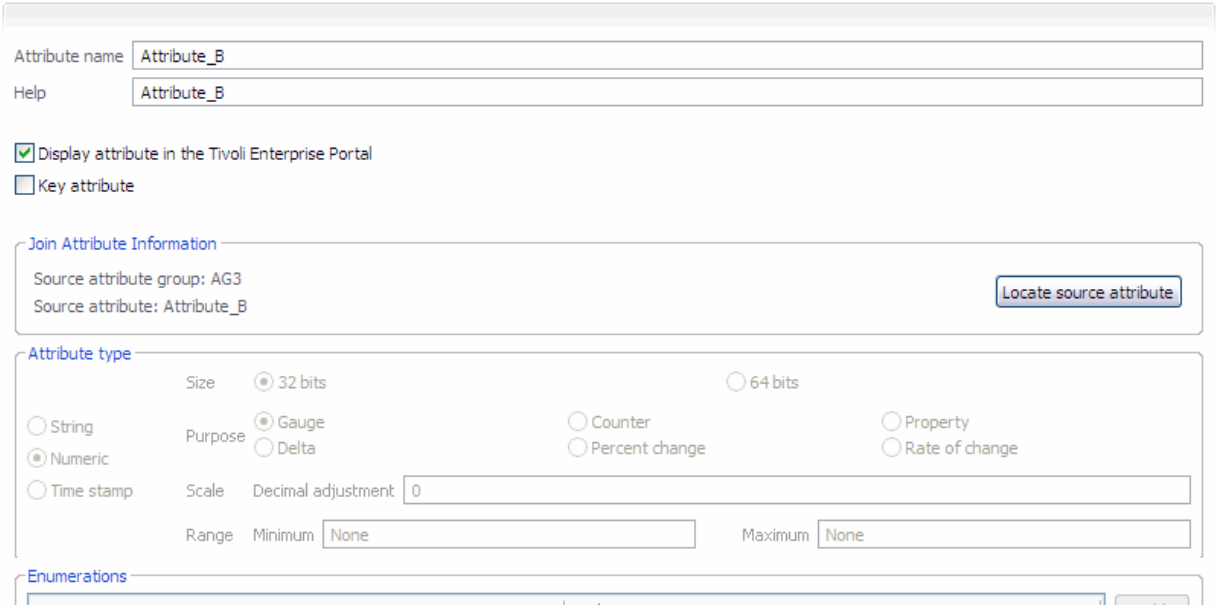


图 *49.* 找出源属性信息

对源属性组所作的更改会反映到连接式属性中。如果源属性组发生更改，那么连接式属性组的那些属性将自动更新。如果将另一个属性组设置为源属性组，那么也会发生此自动更新。对源属性类型所做的更改将复制到连接式属性。对源属性名称或帮助文本所作的更改将复制到连接式属性。但是，在更改连接式属性的名称或帮助文本后，不会复制这些源属性更改。

创建过滤属性组

通过从现有属性组中过滤数据行来创建过滤属性组（数据集）。如果现有数据集返回多行，可以创建返回单行的已过滤组以用于 IBM Cloud Application Performance Management。

关于此任务

过滤属性组包含与源属性组相同的列，但排除了其中一些行。它使用选择公式来确定要包含的行。

要为 Cloud APM 提供状态和摘要信息，需要使用返回单行的数据集。有关详细信息，请参阅第 1154 页的『为 Cloud APM 准备代理程序』。如果源信息位于返回多行的数据集中，可以创建返回单行的过滤属性组。

例如，进程、Windows 服务和命令返回码数据源在单个“可用性”数据集中以行的形式提供信息。您可以在选择公式中使用“名称”字段来创建过滤属性组。该组包含必要应用程序的状态。将其定义为返回单行。然后，可以使用该属性组作为 Cloud APM 的摘要数据集。

如果您希望将基本数据源查询返回的数据分到不同的组中，那么过滤属性组也很有用。Windows 性能监视器、SNMP 和 WMI 是此类数据源的示例。

例如，假设数据源可返回下列数据：

```
Name    Type  Size  Used  Free
Memory  MEM      8     4     4
Disk1   DISK   300   200   100
Disk2   DISK   500   100   400
```

这是报告系统上现有存储器（包括内存和磁盘空间）的表。您可能希望将该表拆分成两个独立的表，分别对应内存和磁盘。您可以创建两个基本属性组来拆分该表。每个基本属性组收集相同的数据并过滤掉不需要的行。但是，这并不是效率最高的做法。您可以改为定义一个同时返回内存和磁盘使用情况数据的基本属性组。然后，定义两个过滤属性组。这两个组都使用同一个基本表作为源。其中一个包含 `Type=="MEM"` 的过滤器，另一个包含 `Type=="DISK"` 的过滤器。

在示例中，对于 `Type=="MEM"` 的过滤属性组，返回的数据是：

```
Name    Type  Size   Used  Free
Memory  MEM      8      4     4
```

对于 Type=="DISK"，返回的数据是：

```
Name    Type  Size  Used  Free
Disk1   DISK  300   200   100
Disk2   DISK  500   100   400
```

注: 数据基于事件的属性组无法用来创建过滤型属性组。只能使用包含样本数据的属性组。

过程

1. 在“**代理程序初始数据源**”页面或“**数据源位置**”页面上的**监视数据类别**区域中，单击**现有数据源**。

   注:

   - 您可以使用“新建代理程序”向导转到“**代理程序初始数据源**”页面。有关更多信息，请参阅第 988 页的『创建代理程序』。
   - 您可以通过在 **Agent Editor** 的“**数据源定义**”页面中右键单击代理程序并选择**添加数据源**，转到“**数据源位置**”页面。
2. 在**数据源**区域中，选择**过滤属性组的数据行**。
3. 单击**下一步**。

   “**过滤器信息**”页面随即显示。
4. 从列表中选择**源属性组**。
5. 输入**选择公式**，以便对来自所选属性组的数据进行过滤。
   例如，在前面显示的“**过滤器信息**”页面中，选择公式根据 Type 属性等于 "DISK" 对数据行进行过滤。Type 属性与 "DISK" 不匹配的数据行将被废弃。您输入的选择公式的求值结果必须是布尔值 true 或 false。

   注: 在“**过滤器信息**”页面中，您可以单击**编辑**，以使用公式编辑器来输入或修改公式。有关公式编辑器的更多信息，请参阅第 1011 页的『Formula Editor』。
6. 单击**下一步**。
7. 选择**生成单一数据行**或**可生成多个数据行**。

   a) 如果选择了**可生成多个数据行**，请从列表中选择一个或多个键属性。
8. 单击**完成**。

创建导航器组

在 IBM Tivoli Monitoring 环境中，使用导航器组将若干相关数据源（属性组）分组到一起，以便能够创建工作空间来显示合并了数据源的视图。在基本代理程序级别使用“新建代理程序”向导创建代理程序时，可以创建导航器组。使用“新建代理程序组件”向导定义子节点时也可以创建导航器组。

关于此任务

例如，您也许能够从多个数据源收集文件系统数据。创建一个显示不同数据源的所有文件系统数据视图的工作空间可能十分有用。

要在 Tivoli Enterprise Portal 中隐藏数据源，导航器组也是一种很好的方法。您可能决定如果将两个数据源合并以创建一个新的组合数据源，那么从这两个数据源收集的度量值最有用。您可能只想在合并数据源中看到组合的数据。您可以创建包含全部三个数据源的导航器组，并创建一个工作空间，其中包含只显示合并数据源的视图。可以有效地在 Tivoli Enterprise Portal 中隐藏两个原始数据源。请参阅第 1121 页的『根据现有来源创建数据集』以了解有关合并数据源的信息。

注: 将数据源分组到导航器组时，Tivoli Monitoring 不会使查询与该导航器组相关联。而是假设您为导航器组定义了缺省工作空间，从而以有用的格式显示数据源。

导航器组可以在基本代理程序或子节点中定义。导航器组不得包含其他导航器组。

导航器组在 IBM Cloud Application Performance Management 环境中不起作用。

过程

1. 执行下列其中一个步骤：
   - 使用“**代理程序**”向导创建新代理程序时，在“**代理程序初始数据源**”页面上，请单击**监视数据类别**区域中的**数据源分组**。
   - 对于现有代理程序，请在 Agent Editor 中执行以下步骤：
     a. 单击**数据源**选项卡，以打开“**数据源定义**”页面。
     b. 选择代理程序并单击**添加至选定项**。
     c. 在“**数据源位置**”页面上的**监视数据类别**区域中，单击**数据源分组**。
2. 在**数据源**区域中，单击**导航器组**。
3. 单击**下一步**。
4. 在“**导航器组信息**”页面上，输入导航器组名称以及要与此名称关联的帮助文本，然后单击**下一步**。

   注: 在某些情况下，Agent Builder 将自动创建导航器组。以下导航器组名称是保留名称：

   · 可用性

   .
5. 在“**第一个导航器组数据源**”页面上，选择新导航器的第一个监视数据源。在**监视数据类别**列表中单击某个类别，并在**数据源**列表中单击某个数据源。然后单击**下一步**。

   提示: 您可以照常创建数据源。或者，单击**现有数据源**并选择将已创建的一个或多个数据源移入导航器组。
6. 如果要在导航器组中创建数据源，请在“**数据源定义**”页面上选择该导航器组，然后单击**添加至选定项**。
7. 如果要将现有数据源移入导航器组，请在“**数据源定义**”页面上选择该导航器组，单击**添加至选定项**，然后在“**导航器组数据源**”页面上选择**现有数据源**。在“**当前定义的数据源**”页面中，选择数据源。
8. 如果要从导航器组移除数据源，请在“**数据源定义**”页面上执行下列其中一个步骤：
   - 选择数据源，然后将其拖至数据源树的根。
   - 选择数据源，然后单击**移除**。
9. 如果要创建导航器组，请在“**数据源定义**”页面上执行下列其中一个步骤：
   - 单击**添加至代理程序**。
   - 选择子节点，然后单击**添加至选定项**。

使用子节点

您可以使用子节点从单个代理程序实例监视多个应用程序组件。

通过使用子节点，您可以构建单个代理程序来完成以下任务：

- 监视系统上运行的每个软件服务器实例，而无需针对每个软件服务器实例使用一个单独的代理程序实例。
- 监视若干个不同的远程系统，而无需针对每个远程系统使用一个单独的代理程序实例。
- 从一个代理程序监视多种不同类型的资源，而无需构建并部署若干个不同的代理程序。
- 在 IBM Tivoli Monitoring 中的 Tivoli Enterprise Portal 物理导航树中显示额外的级别，以允许进一步分组和定制。而且，您可以定义受管系统组，按情境进行进一步细分。
- 在 IBM Cloud Application Performance Management 中，提供若干不同的资源，显示不同的摘要和详细信息仪表板。子节点资源可以显示为代理程序资源的对等项或子组件。您可以将这些资源单独包括在应用程序中。

您可以在 Agent Builder 中创建子节点类型。每种类型必须对应于代理程序所能够监视的不同资源类型。请针对特定的受监视资源，将数据源和数据集添加到子节点类型。

将代理程序部署到受监视主机并进行配置时，可以为每种子节点类型创建一个或多个实例。子节点的每个实例都必须对应于服务器实例、远程系统实例或指定让该子节点类型监视的任何资源实例。单一子节点类型的

所有子节点实例都有相同形式的属性组和工作空间。但是，每个子节点实例都有来自特定受监视资源的数据。

在受监视主机上配置代理程序时，您可确定子节点实例的数量。某些配置数据可作为整体应用于代理程序，而其他配置数据应用于单个子节点实例。请对每个子节点实例采用与其他子节点实例不同的配置方式，确保它们不会监视完全相同的资源，也不会显示完全相同的数据。

在 IBM Tivoli Monitoring 环境中，子节点实例显示在 Tivoli Enterprise Portal 的导航器树中的代理程序下。工作空间显示子节点实例所生成的数据，情境可以分发到子节点的一个或多个实例。将会自动创建包含子节点所有实例的受管系统列表，就像为代理程序创建的“受管系统列表”一样。

在 IBM Cloud Application Performance Management 环境中，代理程序实例和子节点实例都可以显示为受监视资源。每个子节点实例都会成为单独的资源。有关详细信息，请参阅第 1133 页的『IBM Cloud Application Performance Management 中的子节点』。

由于使用 Agent Builder 构建的代理程序会根据配置值来创建子节点实例，因此这些子节点的寿命与代理程序相同。仍然只有一个针对代理程序的脉动信号，而不存在针对每个子节点的单独脉动信号。因此，通过使用子节点，可以显著加大监视环境的潜在规模。替代方法是使用多个代理程序实例，但是这样会限制 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境的潜在规模。

添加或移除子节点要求重新配置代理程序。要重新配置代理程序，您需要将其停止并重新启动，包括所有子节点。您可以将代理程序定义为多实例代理程序；在这种情况下，可以启动并停止单个实例，让其他实例保持运行。

除子节点中的数据集以外，代理程序还可以定义代理程序级别的数据集，它们位于子节点外部。

在 Tivoli Enterprise Portal 的导航器树中，子节点类型显示在代理程序名称下，子节点实例显示在子节点类型下。与代理程序一样，子节点由受管系统名称 (MSN) 识别，例如 `94:Hill.cmn`。

例如，在第 1129 页的图 50 所示的导航器树中，**Watching Over Our Friends** 是有三个资源（**Boarders**、**Common Areas** 和 **Kennel Runs**）以及两种子节点类型（**Common Area** 和 **Kennel Run**）的代理程序。其中两个资源（**Common Area** 和 **Kennel Run**）具有针对其定义的子节点类型。第三个资源 (**Boarder**) 不需要子节点，它由表中基本代理程序级别的单一行表示。Common Area 子节点类型有三个子节点实例：`94:Hill:cmn`、`94:Meadow:cmn` 和 `94:Tree:cmn`，它们表示内核中的三个公共区域。Kennel Run 子节点类型有四个子节点实例：`94:system1:run`、`94:system2:run`、`94:system4:run` 和 `94:system5:run`，它们表示四个 Kennel Run。

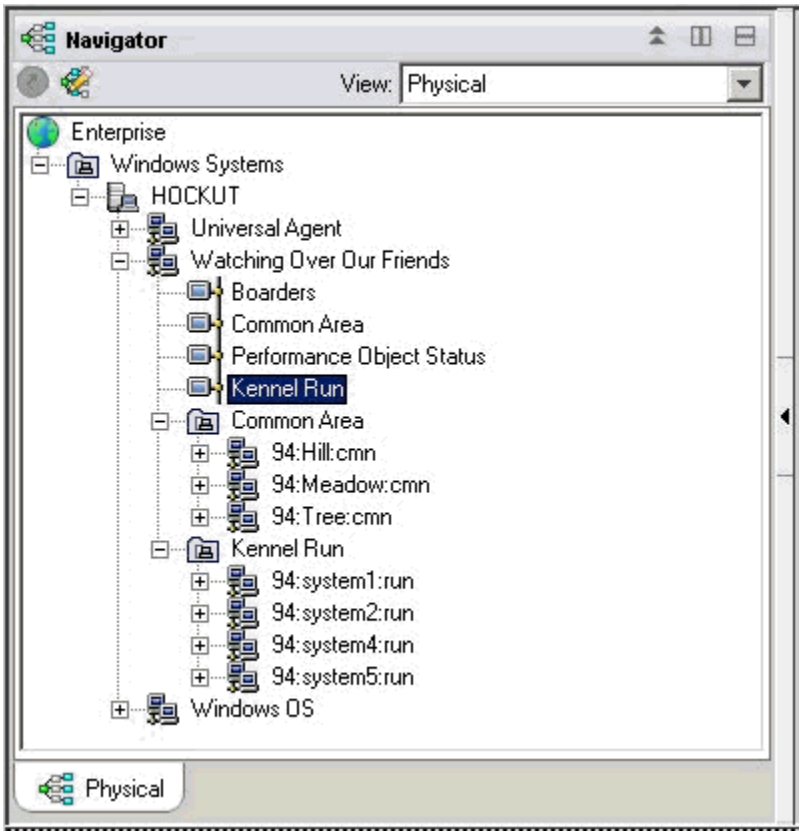


图 *50.* 导航器树中的子节点

单个代理程序可以两种方式使用子节点：

· 代理程序可以有类型相同的不同子节点。

· 代理程序可以有类型不同的子节点。

用于来自不同来源的同一数据的子节点

您可以使用同一类型的子节点来表示受监视资源类型的多个实例。同一类型的每个子节点包含相同的属性组和特定受监视资源实例的正确值。子节点的数量随代理程序配置不同而异。第 1130 页的图 51 中的示例显示不同系统的监视。

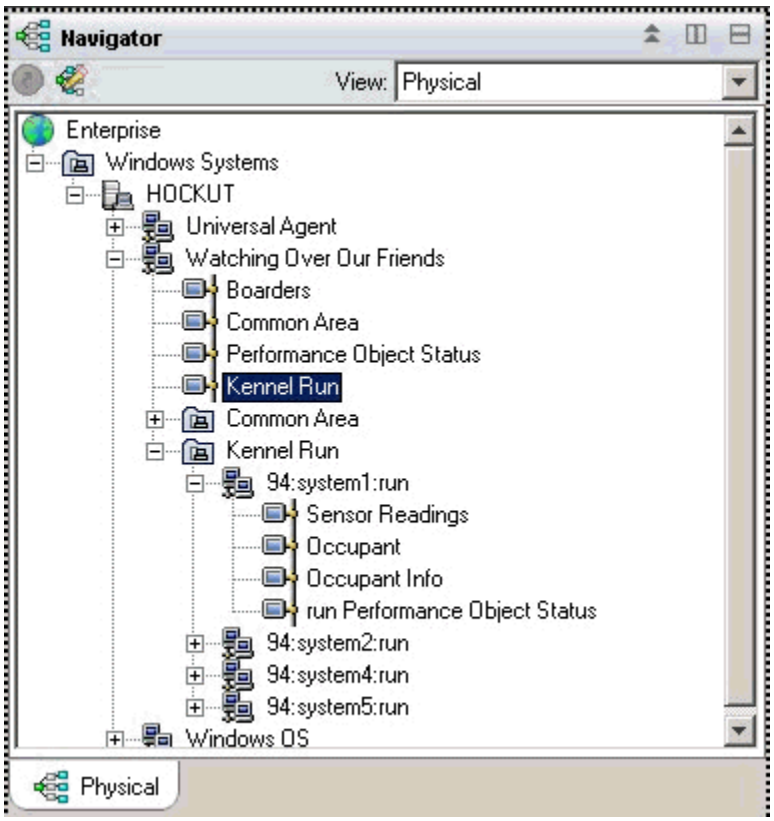


图 *51.* 监视不同系统的子节点

用于多类数据的子节点

当一个代理程序监视多种类型的受监视资源时，您可以为每种资源类型创建一种子节点类型。每个子节点都会包含该子节点类型中定义的信息。以下示例显示两种子节点类型。每种类型都监视一种不同类型的资源，并针对每种资源提供不同类型的数据：

- Common Area
- Kennel Run

第 1131 页的图 52 中的代理程序运行每种子节点类型的一个副本。特定代理程序可以创建已定义的代理程序的任意子集。子节点可以用于模拟 Tivoli Monitoring V5 概要文件。

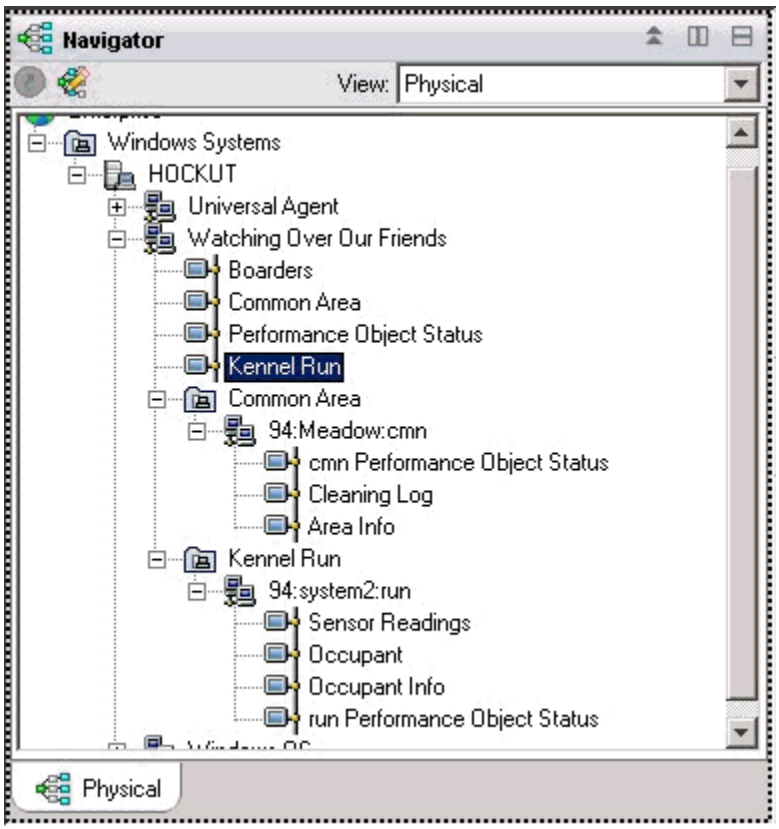


图 *52.* 导航器树中的子节点类型

这两种使用子节点的方法可以在同一代理程序中使用，其中每种类型都可以有多个子节点实例。

第 1131 页的图 52 显示监视以下两类资源的两类子节点：Common Area 和 Kennel Run。另外，还为每种类型定义了若干子节点。Common Area 类型的有三个子节点；这些子节点的标识如下：Meadow、Hill 和 Tree。另外还有四个 Kennel 类型的子节点，它们分别从专用于 Kennel Run 的不同系统收集数据；这些子节点的标识如下：system1、system2、system4 和 system5。

注: 子节点标识的前 24 个字符对于 IBM Tivoli Monitoring 安装中该子节点类型的所有实例而言必须是唯一的。

子节点中的数据提供程序

子节点可以包含来自不同数据提供程序类型的任意混合数据。可以在子节点中使用最新的 Agent Builder 数据提供程序，包括以下数据提供程序：

- WMI
- Perfmon
- Windows 事件日志
- SNMP
- SNMP 事件
- JMX
- ICMP Ping
- 脚本
- 日志
- CIM
- JDBC

· HTTP

· SOAP

· 套接字

· Java API

子节点还可以包含连接属性组，用于组合来自相同子节点的两个其他属性组或代理程序级别属性组的数据。

子节点状态

有两种方式可用来确定子节点代理程序的状态。第一种方式是查看“性能对象状态”属性组中显示的数据。该属性组显示了代理程序中同一级别上的其他所有属性组的状态。代理程序级别的“性能对象状态”属性组显示了代理程序级别上的其他属性组的收集状态。而每个子节点中的“性能对象状态”属性组显示了该子节点中的属性组的收集状态。

Agent Builder 还将为每个子节点类型创建一个属性组，该属性组为该类型的每个已配置子节点显示一行。在（第 1132 页的图 53）中的示例中，四个节点正在运行以收集数据。

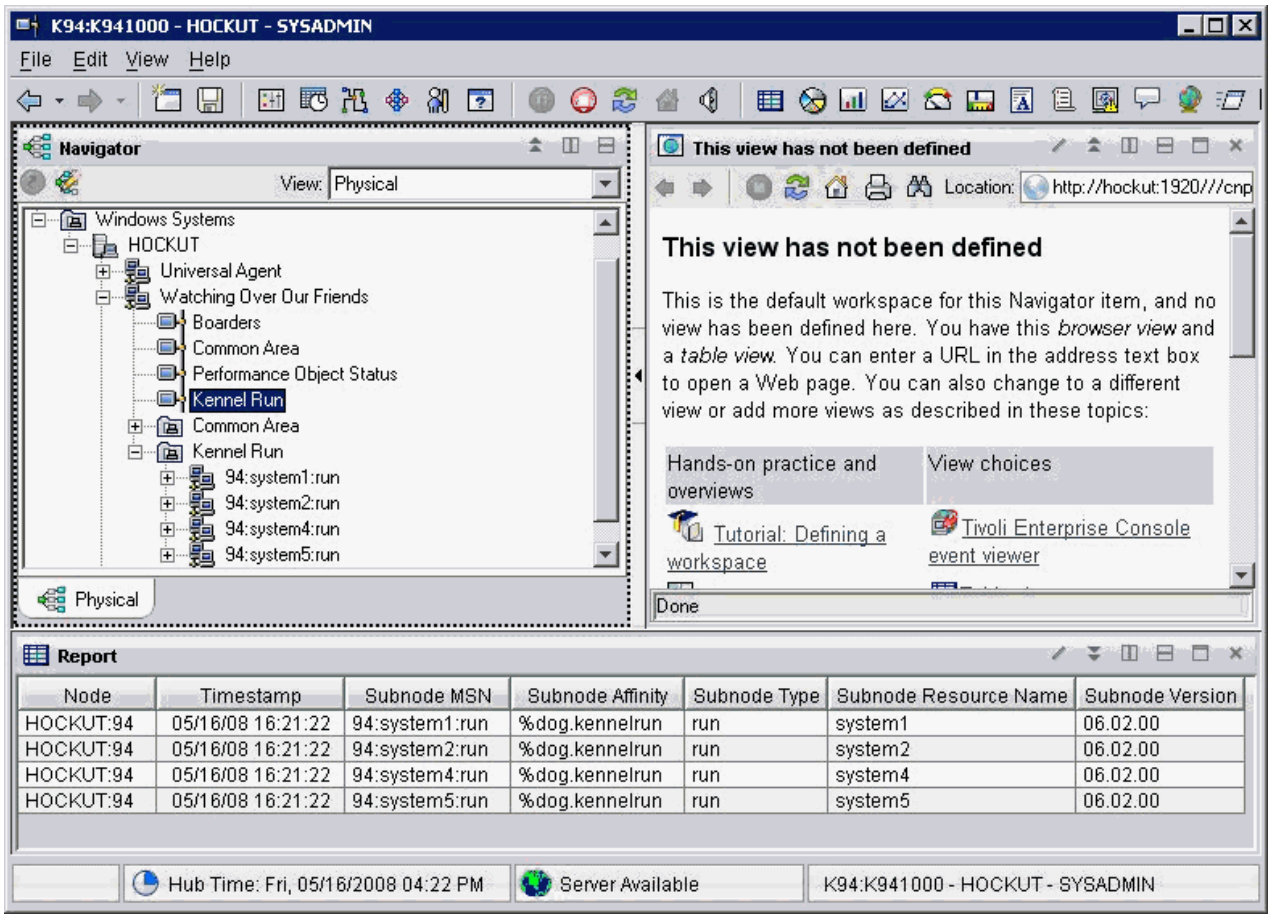


图 *53.* 监视相同子节点类型的多个子节点实例

在 IBM Tivoli Monitoring 环境中，**性能对象子节点**子节点包含导航器树中可见的数据，并具有一些用于监视其他数据收集状态的情境。

在 IBM Cloud Application Performance Management 环境中，您可以创建阈值来监视**性能对象状态**数据。

第 1133 页的图 54 中的示例显示数据收集失败的情况（由于找不到脚本 shell 命令而失败）。通常除 NO_ERROR 外的所有值都表示存在问题。对于子节点中定义的每个数据收集器，表中都有对应的一行。

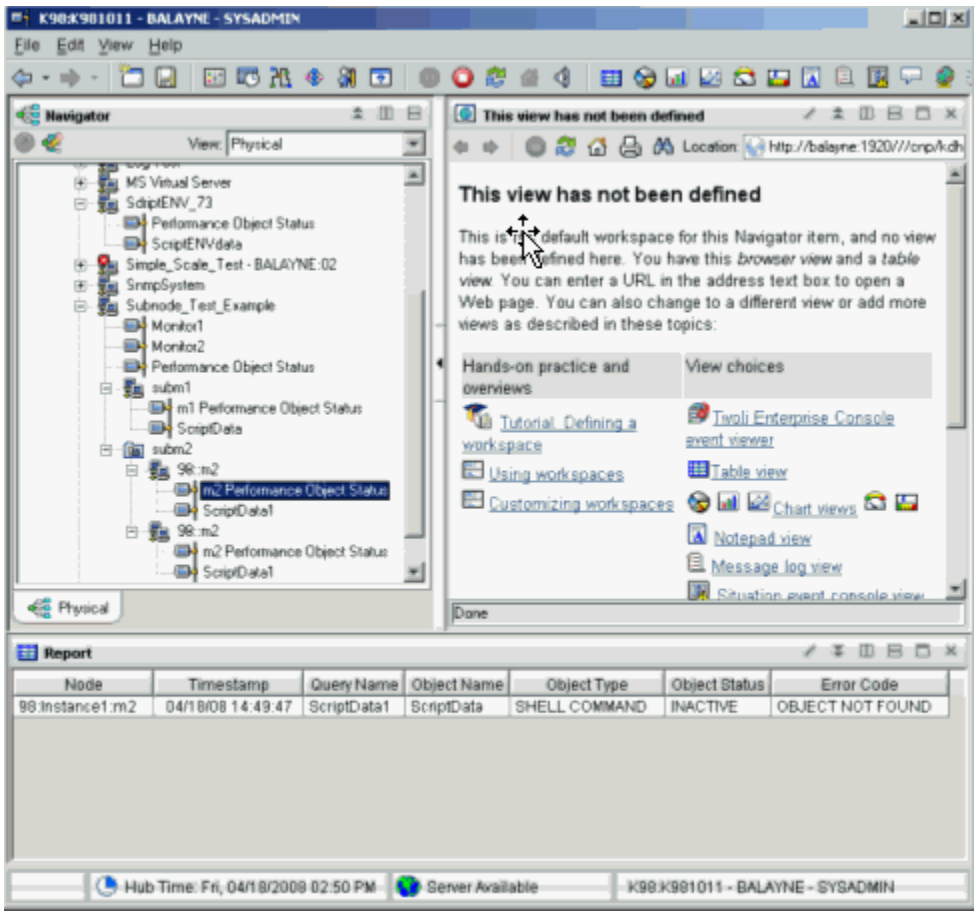


图 *54.* 示例：子节点中的数据收集

IBM Cloud Application Performance Management 中的子节点

在 IBM Cloud Application Performance Management 中，您可以将代理程序实例和/或子节点实例定义为受监视资源，每个资源对应于一个摘要仪表板。

子节点仪表板无法显示代理程序级别的数据。要在此环境中显示代理程序级别的数据，请为代理程序定义摘要仪表板。

根据所选设置，代理程序和子节点资源可以显示在同一级别，没有层次结构之分，或者将子节点资源列示为代理程序资源的子代。

有关配置代理程序和子节点资源的指示信息，请参阅第 1154 页的『为 Cloud APM 准备代理程序』。

创建子节点

您可以在创建或编辑代理程序时创建子节点。

过程

1. 执行下列其中一个步骤：
   - 使用“**代理程序**”向导创建新代理程序时，在“**代理程序初始数据源**”页面上，单击**监视数据类别**区域中的**数据源分组**。
   - 对于现有代理程序，请在 Agent Editor 中执行以下步骤：
     a. 单击**数据源**选项卡，以打开“**数据源定义**”页面。
     b. 选择代理程序并单击**添加至选定项**。
     c. 在“**数据源位置**”页面上的**监视数据类别**区域中，单击**数据源分组**。
2. 在**数据源**区域中，单击**子节点定义**。

3. 单击**下一步**。
4. 按如下所示填写“**子节点信息**”页面，以定义新的子节点：
   a) 在**名称**字段中，输入要创建的子节点的名称。
   b) 在**类型**字段中，输入 1 到 3 个字符（使用数字和/或字母），以指示所创建子节点的类型。
   c) 在**描述**字段中，输入要创建的子节点的描述。
   d) 单击**针对此类子节点显示节点属性组**复选框，以隐藏或显示可用性属性组。有关此属性组的更多详细信息，请参阅第 1190 页的『可用性节点』。
   e) 单击**下一步**。
5. 填写“**初始子节点数据源**”页面，以选择一个数据源作为新子节点中的第一个项。在**监视数据类别**列表中单击类别，并在**数据源**列表中单击数据源。然后，单击**下一步**。

   提示: 您可以照常创建数据源。另外，您也可以将已创建的一个或多个数据源移入导航器组。要移动数据源，请单击**现有数据源**，并在“**当前定义的数据源**”页面中选择数据源。

   要点: 不能将进程、Windows 服务或命令返回码数据源包括在子节点中。作为变通方法，您可以编写脚本来确定必要的进程或服务信息，并使用脚本输出数据源。
6. 如果代理程序包含定制配置属性，或者所选数据源需要配置，请使用“**子节点配置覆盖**”页面来选择配置属性。

   在“**子节点配置覆盖**”页面中，选择要在代理程序级别用于子节点的配置属性。然后，请选择要随每个子节点而不同的配置属性。

   使用**移动**、**复制**和**移除**，按照第 1134 页的『配置子节点』所述指定配置属性。
7. 单击**下一步**。
   “**数据源定义**”页面随即显示。

子节点配置

定义子节点类型时，将会针对该子节点专门定义一个配置节。

子节点配置节与其他配置节有几点不同：

- 子节点节中的属性集可以复制，因此有多组属性。每组属性构成其自己的节。所有节的布局都相同，但可以在每个节中输入不同的值。

  与此相反，其他节（即代理程序级别的节）中的属性在运行时配置期间仅显示一次。它们不会构成子节，且无法复制或移除。

  请参阅第 1137 页的『子节点配置示例』，以获取配置子节点的 GUI 和命令行示例。
- 对于在运行时配置期间创建的子节点节的每个副本，代理程序将创建单独的子节点实例。所有子节点实例的类型相同。
- 子节点节中的属性名可以与代理程序级别的节中的属性名重复。出现重复名称时，子节点属性值会覆盖代理程序级别的属性值。
- 在 IBM Tivoli Monitoring V6.2.1 和更高版本中，子节点节可以具有缺省属性值，应用于该类型的子节点的所有实例。此功能允许对单个属性值执行三层查找，如下所示：
  1. 代理程序从子节点实例子节获取属性值。
  2. 如果在子节点实例级别未配置任何值，那么将从子节点缺省级别获取属性值。
  3. 如果在上述两个级别均未配置值，那么将从代理程序级别的节获取属性值。

请参阅第 1137 页的『子节点配置示例』，以获取配置子节点的 GUI 和命令行示例。

配置子节点

使用**子节点配置覆盖**页面可配置子节点数据源。

开始之前

使用第 1133 页的『创建子节点』中的步骤来创建子节点。

关于此任务

将数据源添加到子节点时，如果该数据源需要配置，那么会显示“**子节点配置覆盖**”页面。它会显示定制的配置属性，以及任何其他适用于子节点类型的配置属性。

过程

- 在“**子节点配置覆盖**”窗口中，选择要在代理程序级别用于子节点的配置属性。另外，请选择要随每个子节点而不同的配置属性。
- 使用**复制 >>** 来复制配置属性，使它们同时存在于代理程序级别和子节点级别。

  代理程序先在子节点级别查找值，如果找不到，接着在代理程序级别上查找。如果某个同时存在于这两个级别的属性是必需属性，那么它只是在代理程序级别上必需，而在子节点级别是可选项。
- 使用**移动 >>** 将属性从代理程序级别移至子节点级别。对于代理程序级别数据源或其他类型子节点所需的属性，**移动 >>** 不可用。
- 使用**移除**来移除两个列表中的一个。属性只有同时在代理程序级别和子节点级别列出时才可移除。此功能不能用于完全移除某个属性。
- 使用 **<< 复制**将属性从子节点级别复制到代理程序级别。
- 使用 **<< 移动**将属性从子节点级别移至代理程序级别。

下一步做什么

您可使用 Agent Editor 更改现有子节点的配置。

子节点配置覆盖

使用“子节点配置覆盖”可将代理程序配置属性覆盖为特定于子节点的属性。

第 1134 页的『配置子节点』中的过程描述如何针对自动生成的属性管理子节点配置。管理定制配置属性与此相似。所定义的任何定制配置属性都会显示在**子节点配置覆盖**窗口中。

在将定制属性从子节点级别复制或移动到代理程序级别时，系统会提示您指定要将其放入哪个节。您可以选择现有的定制节，也可以输入新定制节的名称。

选择子节点配置属性

如果没有子节点，那么数据源类型的所有实例将共享配置参数。例如，所有 SNMP 属性组将使用相同的共用名来连接到同一主机。如果有子节点，且在子节点级别指定 SNMP_HOST 属性，那么子节点的各个实例可以连接到不同的主机。

开发代理程序时，选择要在子节点级别覆盖的属性是一项重要的考虑事项。选择了过多属性时，子节点配置节会变得杂乱，难以管理。如果选择的属性过少，那么用户希望某个属性因子节点不同而有变化时，代理程序功能可能会受限。

以下属性无法复制到子节点级别。（所有子节点及基本代理程序中的所有属性组都必须使用相同的 SNMP 版本和 JMX 连接类型）：

- SNMP 版本
- JMX MBean Server 连接类型
- Java 主目录
- Java 跟踪级别
- JVM 自变量
- 外部 JAR 文件的类路径
- 套接字数据源端口号
- JMX 或 JDBC 类路径设置

高级子节点配置

使用高级子节点配置可以在子节点中覆盖代理程序配置属性。

关于此任务

在 IBM Tivoli Monitoring V6.2.1 和更高版本代理程序中有一个选项，启用此选项即可在子节点实例中覆盖来自任意代理程序级别配置节的属性。在“**子节点配置覆盖**”页面上，有一个标注为**允许在任何子节点中覆盖任何配置属性**的复选框。有关更多信息，请参阅第 1135 页的『子节点配置覆盖』。要启用此选项，在对代理程序命名（第 988 页的『对代理程序命名和配置』）时，必须选择 **6.2.1** 作为**最低 ITM 版本**。如果选择此选项，那么每个子节点实例都可以覆盖任何代理程序级别配置节的任何属性。但是，只能从 GUI 覆盖此属性，而无法从 **itmcmd** 命令行执行覆盖。

过程

允许在任何子节点中覆盖任何配置属性选项会在每个子节点配置面板上显示**高级**字段，其中包含一个列表。**高级**字段中的初始选择提供了简要说明：**请选择一个节以覆盖值**。

- 单击该列表时，它会列出所有包含配置属性的非子节点节。
- 选择一个节。

  该节的属性将临时添加到子节点面板中。所更改的任何属性值将添加到针对该子节点定义的属性集。子节点中的数据源会先在子节点中查找属性值，然后再从代理程序级别的节中查找。.

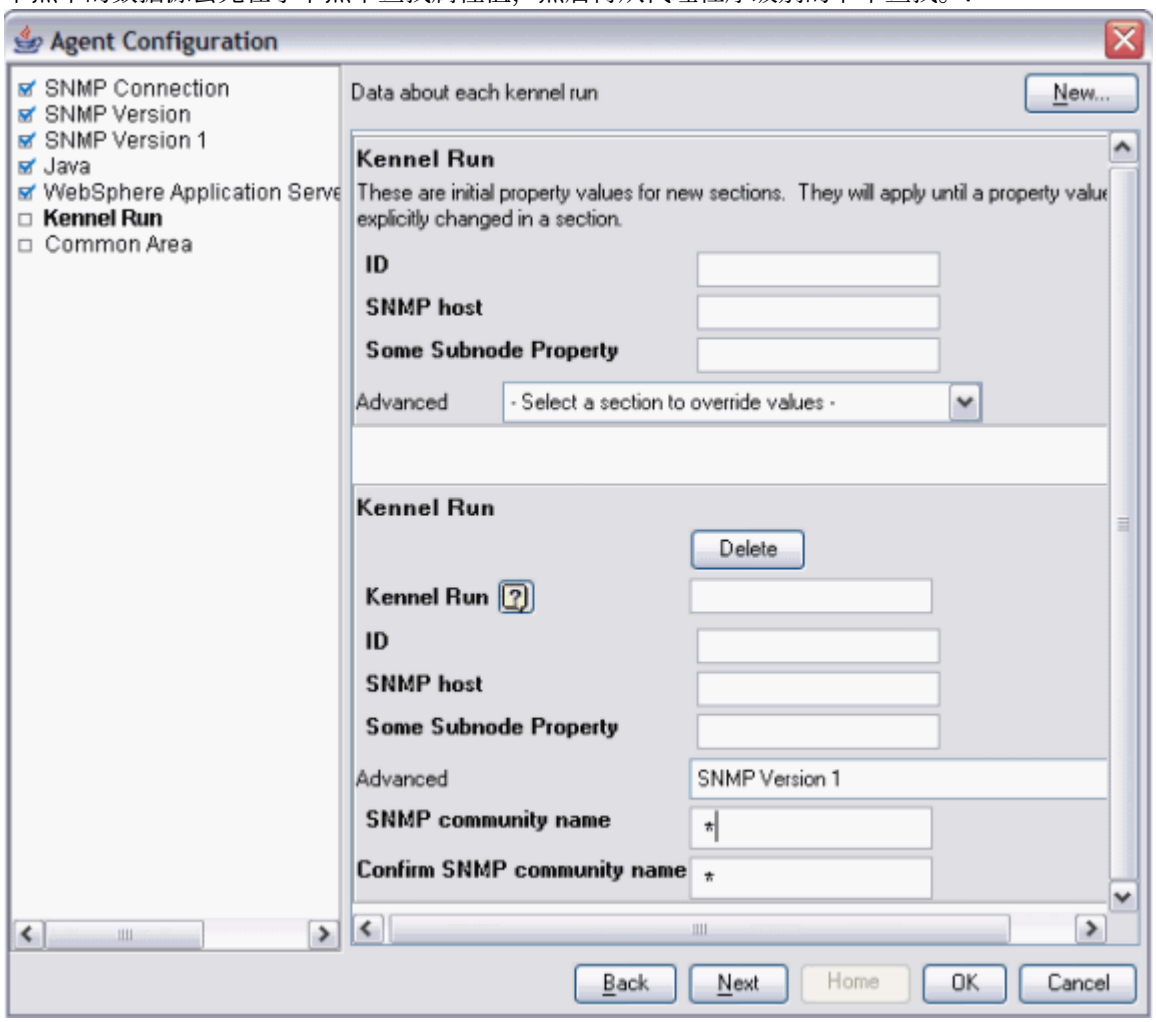


图 *55.* 展开的 *SNMP V1* 属性

下列进一步的信息适用于覆盖来自代理程序级别节的属性：

- 在**高级**列表中选择代理程序级别的节时，不会显示之前复制到子节点节的属性。例如，在第 1136 页的图 55 中，**SNMP 主机**不会显示在**高级**列表之后，因为它已复制到子节点属性，并已显示。
- 在**高级**列表中，不可选择不包含可覆盖属性的节。
- 系统会保留您为一个节输入的覆盖值，即使您选择另一个节以显示其他属性也是如此。
- 选中**允许在任何子节点中覆盖任何配置属性**，以便在代理程序中启用此功能。

从命令行配置子节点

在 IBM Tivoli Monitoring 环境中，还可以使用命令行来配置子节点。

开始之前

有关子节点配置的更多信息，请参阅第 1134 页的『子节点配置』。

关于此任务

过程

· 要从命令行配置子节点实例，请使用以下命令：

```
tacmd configureSystem -m HOSTNAME:00 -p
section_name:subnode_instance_id.property_name=value
```

其中：

section_name
与子节点类型相同

subnode_instance_id
配置期间定义的子节点标识。

property_name
配置属性的名称

value
属性的值

子节点配置示例

本节说明如何配置具有一个已定义的子节点的样本代理程序。

示例：

此示例说明如何配置一个样本代理程序，它有一个名为 Example Subnode 且类型为 exs 的子节点以及下列三个配置属性：

· Agent Cfg（实际属性名为 K00_AGENT_CFG）仅在代理程序级别定义。

· Subnode Cfg（实际属性名为 K00_SUBNODE_CFG）仅在 Example Subnode 中定义。

· Overridable Cfg（实际属性名为 K00_OVERRIDABLE_CFG）在代理程序级别定义，并已复制到 Example Subnode。

第 1138 页的图 56 显示 Agent Editor 中“**运行时配置信息**”页面上的这些配置属性。

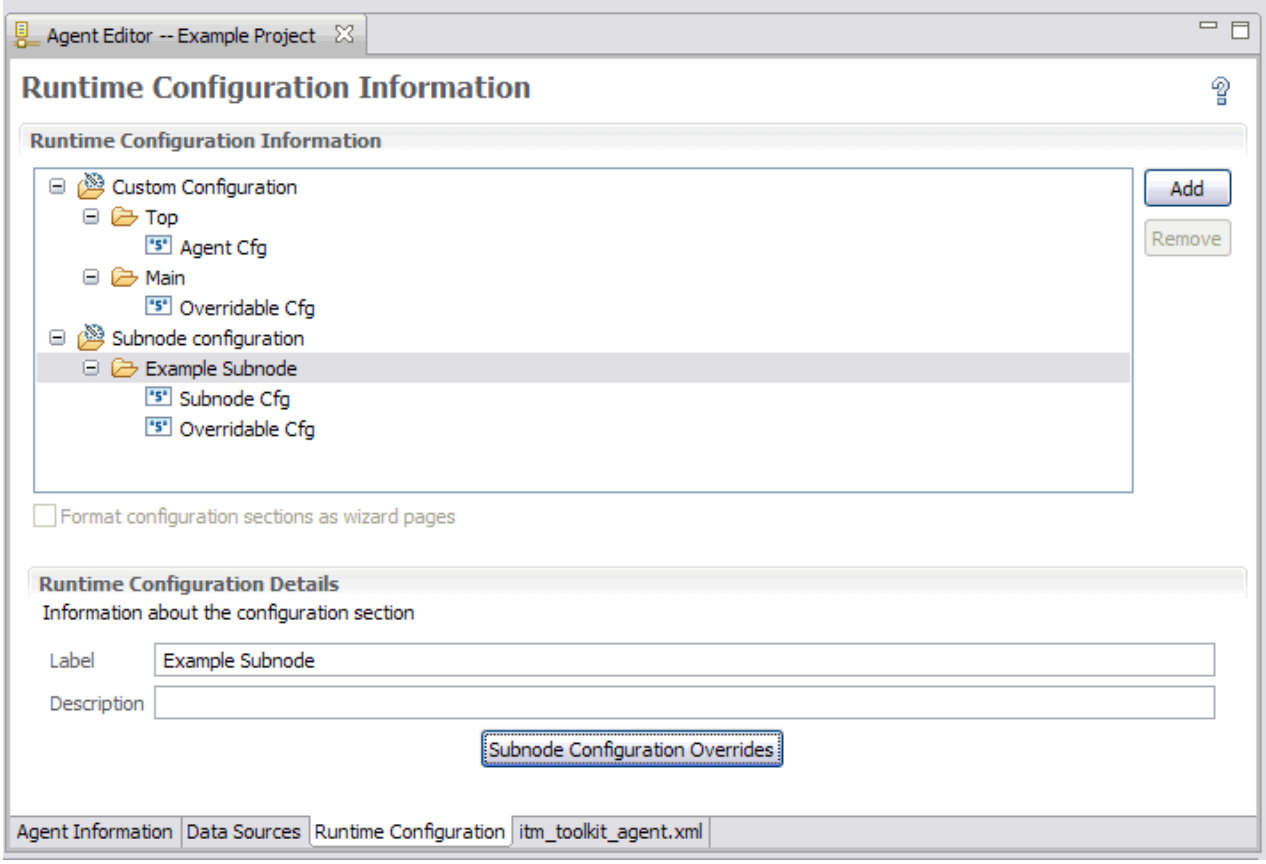


图 *56. Agent Builder* 中的配置属性定义

配置此示例代理程序时，显示的第一个页面是 **Top** 节，其中包含第 1139 页的图 57 所示的 **Agent Cfg** 属性。此属性是代理程序级别的属性，因此在代理程序配置期间会显示一次。Example Subnode 的任何实例都可以看到该属性值，但所有实例看到的是同一个值。

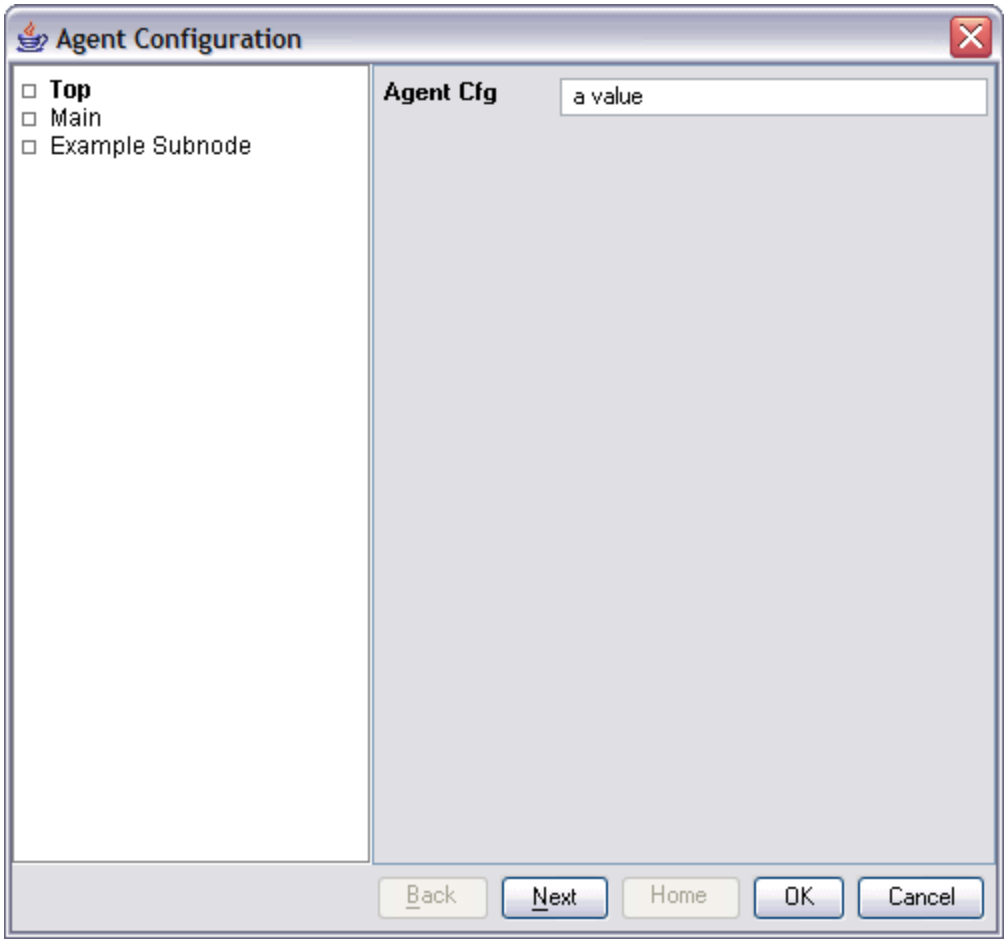


图 *57.* 包含 ***Agent Cfg*** 属性的代理程序级别配置的 ***Top*** 节

如果要从 Tivoli Enterprise Monitoring Server 命令行进行配置，可以使用以下命令来设置 **Agent Cfg** 属性：

```
tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00_AGENT_CFG=a value"
```

显示的下一个节是第 1140 页的图 58 中所示的 **Main** 节。这也是代理程序级别的节，其中包含代理程序级别的 **Overridable Cfg** 属性。此属性不同于 **Agent Cfg** 属性，它已复制到 Agent Builder 中的 Example Subnode。这表示可以在 **Main** 页面上输入此属性的缺省值。但是，任何 Example Subnode 实例都可以将此处输入的值覆盖为另一个值。

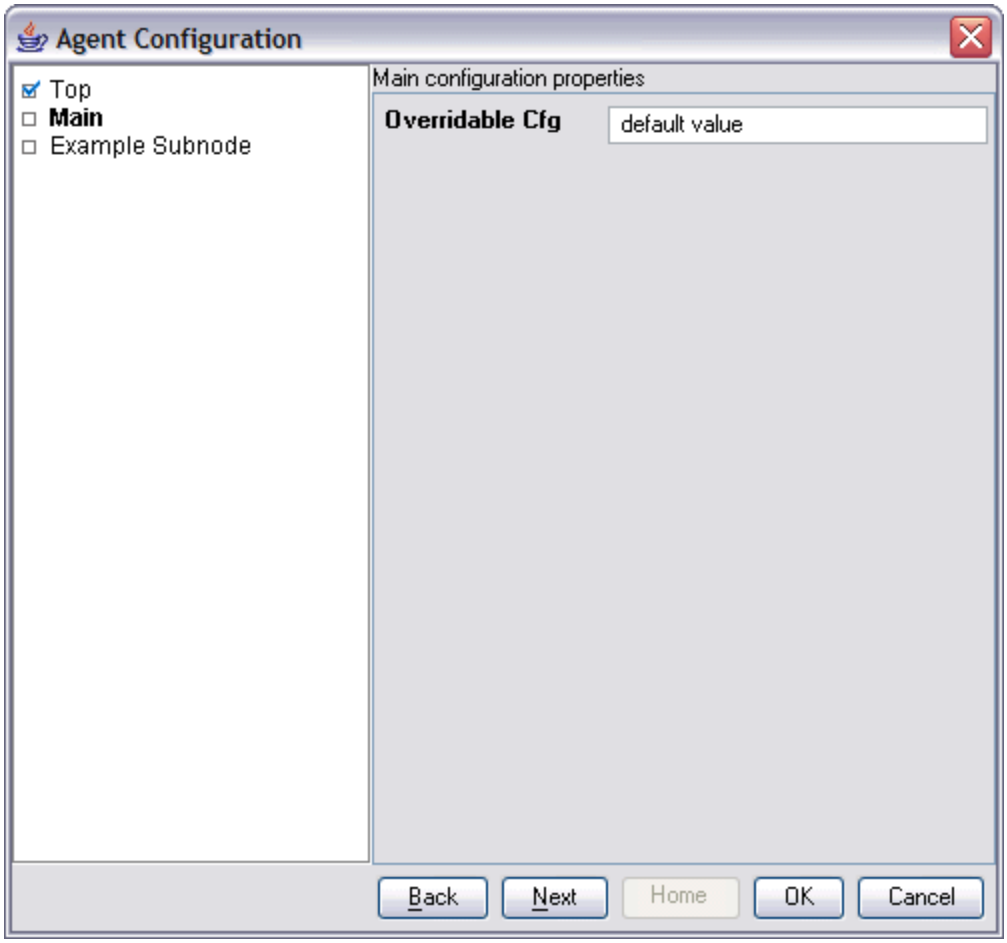


图 *58.* 包含 ***Overridable Cfg*** 属性的代理程序范围缺省值的 ***Main*** 节

如果要从 Tivoli Enterprise Monitoring Server 命令行进行配置，可以使用以下命令设置此属性：

```
tacmd configureSystem -m HOSTNAME:00 -p "MAIN.K00_OVERRIDABLE_CFG=default value"
```

可以将这两个属性放入同一个代理程序级别的节中。您可以决定要创建多少个代理程序级别的定制节，以及如何将定制属性分配到这些节中。

显示的下一个节是第 1141 页的图 59 中所示的 **Example Subnode** 节。这是首次配置此代理程序，因此没有已定义的子节点实例，也没有显示子节点实例子节。将会显示初始属性值子节，尽管它是可选项，而且某些子节点类型可能不显示该子节。由于显示了初始属性值子节，因此可以输入任何配置属性的缺省值。**Overridable Cfg** 属性已具有缺省值，该值获取自代理程序级别的同名属性。

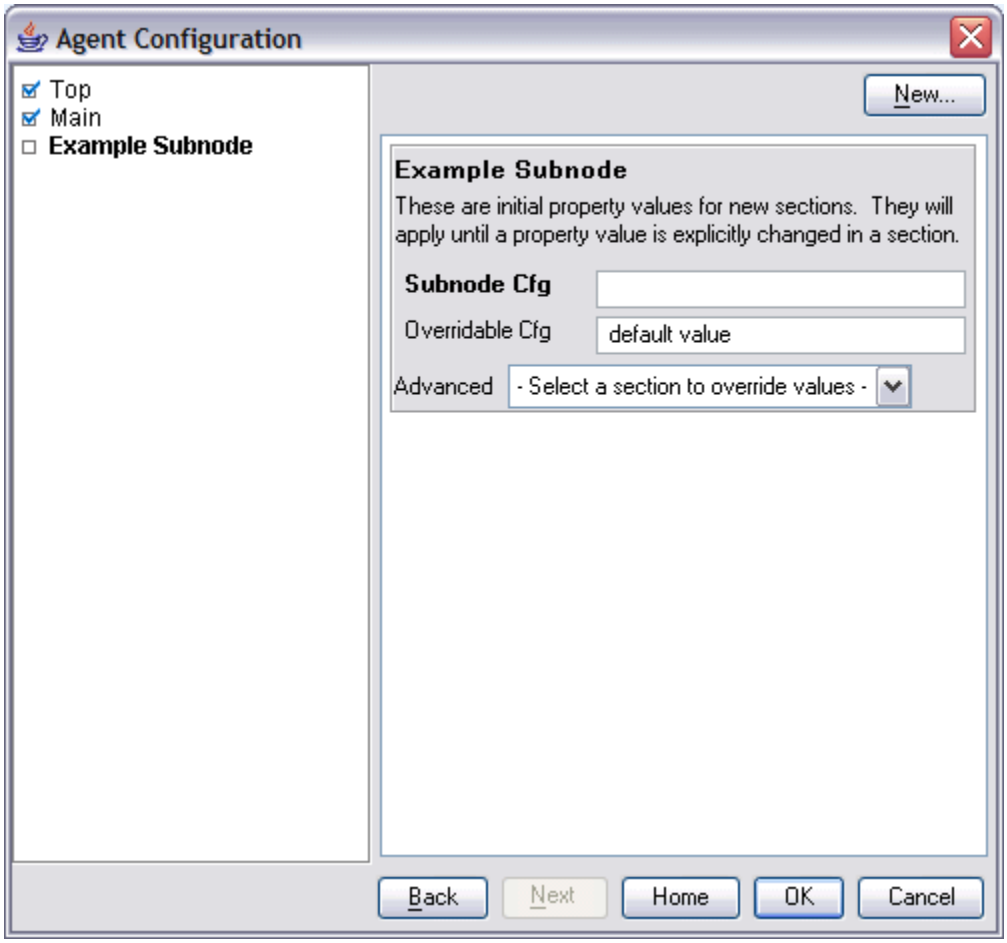


图 *59.* 没有子节点的 ***Example Subnode*** 节页面

通过在空的 **Example Subnode** 节页面（第 1142 页的图 60）上执行以下操作，可以定义子节点实例：

1. 在初始 **Example Subnode** 节中的 **Subnode Cfg** 字段中，为该属性输入以下缺省字符串：`sub-default value`。
2. 单击**新建**。**Example Subnode** 子节随即显示在初始属性子节之后。
3. 在 **Example Subnode** 字段中，输入以下子节点实例标识：`do`。
4. 单击**新建**。另一个 **Example Subnode** 子节随即显示在第一个子节之后。
5. 在第二个 **Example Subnode** 字段中，输入以下子节点实例标识：`re`。
6. 在 **Subnode Cfg** 字段中，为 **Subnode Cfg** 属性输入以下值：`sc override`。
7. 在 **Overridable Cfg** 字段中，为 **Overridable Cfg** 属性输入以下值：`oc override`。

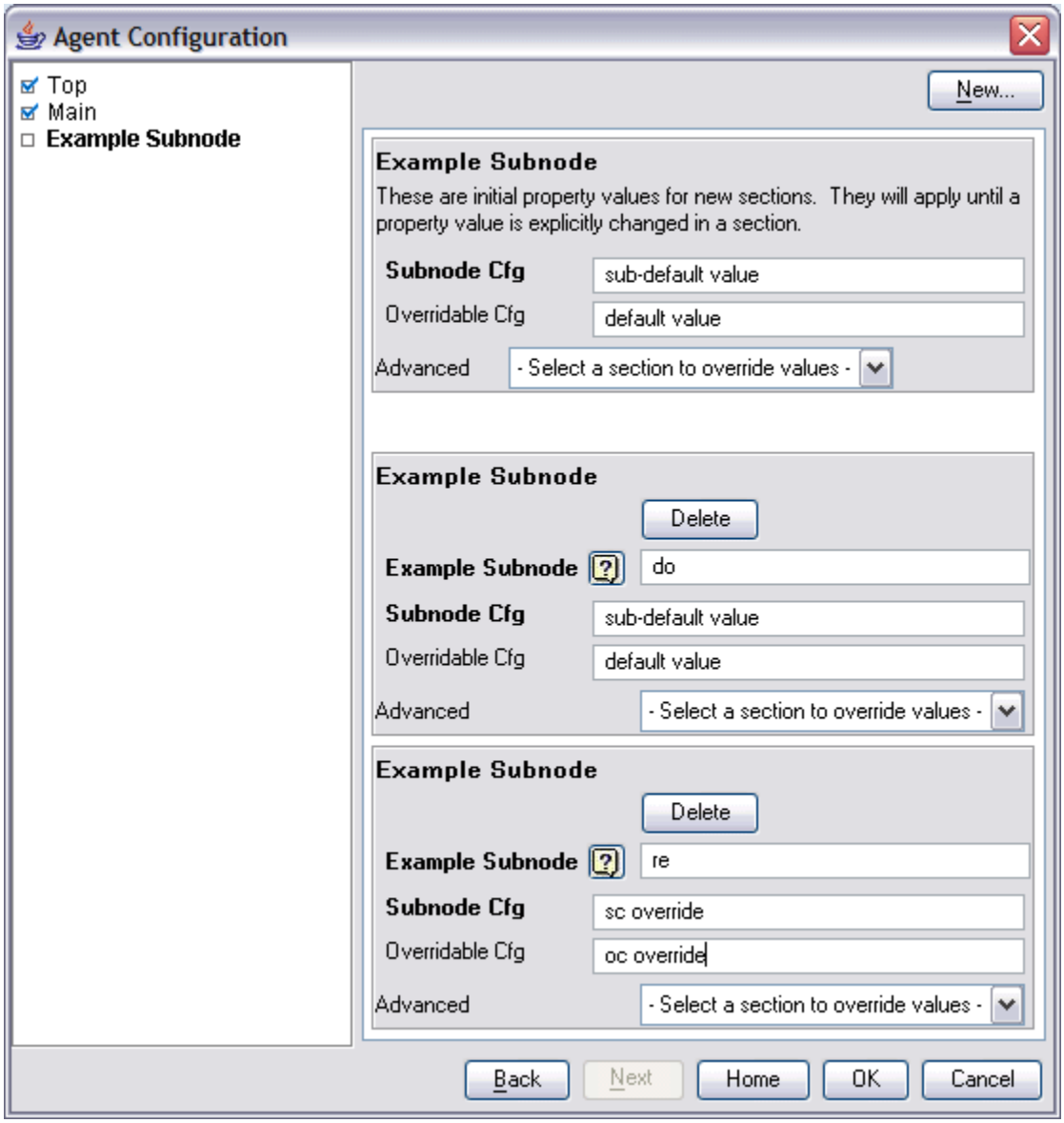


图 *60.* 定义了两个子节点实例的 ***Example Subnode*** 节页面

两个新的子节会使得代理程序在启动时创建两个子节点实例。由于 **do** 子节点子节的属性未更改，因此该子节点实例会使用缺省属性值。因为在 **re** 子节为属性输入了不同的值，**re** 子节点实例会使用输入的那些值。

您可以从 Tivoli Enterprise Monitoring Server 命令行使用以下命令来设置缺省值：

```
tacmd configureSystem -m HOSTNAME:00 -p "exs.K00_SUBNODE_CFG=sub-default value"
```

除了节名称用来识别子节点节之外，设置子节点缺省值的格式与设置代理程序级别属性的格式完全相同。

您可以从 Tivoli Enterprise Monitoring Server 命令行使用以下命令来创建子节点实例：

```
tacmd configureSystem -m HOSTNAME:00 -p "exs:do.K00_OVERRIDABLE_CFG=default value" \
 "exs:re.K00_SUBNODE_CFG=sc override" "exs:re.K00_OVERRIDABLE_CFG=oc override"
```

子节点实例标识将插入到节名称与属性名之间。使用命令行创建子节点实例时，即使所有属性都使用缺省值，也必须至少指定一个属性。否则，定义子节点实例时，无需在命令行上指定缺省值。

可以在一条命令中设置所有的代理程序配置属性。以下命令等同于上述所有命令的总和：

```
tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00_AGENT_CFG=a value" \
 "MAIN.K00_OVERRIDABLE_CFG=default value" \
 "exs.K00_SUBNODE_CFG=sub-default value" \
```

```
"exs:do.K00_OVERRIDABLE_CFG=default value" \
"exs:re.K00_SUBNODE_CFG=sc override" "exs:re.K00_OVERRIDABLE_CFG=oc override"
```

子节点与 Windows 数据源
选择是否将 Windows 远程连接属性包括在代理程序中。

关于此任务

如果代理程序在代理程序级别有 Windows 数据源，而在子节点中没有这些数据源，那么在代理程序中包括 Windows 远程连接配置属性这一操作是可选的。Windows 数据源包括 Windows 事件日志、Windows Management Instrumentation 和 Windows 性能监视器。如果未包括配置属性，那么在缺省情况下，这些数据源会监视本地 Windows 系统，且无需进行任何配置。缺省情况下，不会在任何子节点中包括 Windows 数据源。

要选择是否在代理程序中包括 Windows 远程连接属性，请执行以下步骤：

过程

1. 在“**Windows Management Instrumentation (WMI) 信息**”页面上，当数据源属性显示时，单击**全局选项**。在创建数据源时选择**全局选项**，或者从 Agent Editor 的“**数据源**”页面选择此选项。
2. 在“**全局 Windows 数据源选项**”窗口中，如果要将这些属性包括在代理程序中，请选择**包括 Windows 远程连接配置**。

子节点和脚本数据源
在子节点脚本中访问子节点实例配置属性，就像在代理程序级别的脚本中访问一样。

脚本可以访问代理程序级别的所有配置属性，以及所有子节点实例配置属性。如果代理程序级别的属性在子节点级别被覆盖，那么脚本只能访问子节点级别的属性值。

定制代理程序配置

定制进程、日志文件和脚本数据源的配置。

开始之前

如果要向代理程序添加 SNMP、JMX、CIM、JDBC、HTTP 和 SOAP 数据源，请按如下章节所述配置这些数据源：

- 第 1033 页的『监视来自简单网络管理协议 (SNMP) 服务器的数据』
- 第 1042 页的『监视 Java 管理扩展 (JMX) MBean』
- 第 1058 页的『监视来自公共信息模型 (CIM) 的数据』
- 第 1080 页的『监视来自 Java 数据库连接 (JDBC) 的数据』
- 第 1088 页的『监视 HTTP 可用性和响应时间』
- 第 1094 页的『监视来自 SOAP 或其他 HTTP 数据源的数据』

关于此任务
使用本任务可以定制进程、日志文件和脚本数据源的配置，以使代理程序能够访问其所监视的应用程序。

所有代理程序都必须先进行配置，然后才能启动。所有代理程序都必须具有基本配置信息，例如连接至 Tivoli Enterprise Monitoring Server 的方法。在许多情况下，代理程序必须掌握更多配置信息才能访问运行该代理程序的系统所特有的信息。例如，如果必须知道某个软件产品的安装位置，那么可以添加配置属性来提示输入该信息。您可能提示输入的信息还可以是用于访问某个界面的用户标识和密码。

定制配置由代理程序开发者定义。并非所有代理程序需要定制配置，但是定制配置可以用于数据收集的以下方面：

- 与“进程监视器”中的自变量匹配
- 与“进程监视器”中的命令行匹配

· 构成日志文件路径或名称

· 在脚本中定义环境变量

注: 某些数据源（例如 JMX 和 SNMP）会自动添加此配置。

注: 当 Agent Builder 自动添加特定于数据源的配置时，将只添加此配置的英语版。

在数据源定义期间，如果代理程序需要某个数据收集区域的特定于系统的信息，那么将显示**插入属性**或**插入配置属性**。

例如，在创建用于监视日志文件的属性组时，将显示**插入配置属性**。

过程

1. 单击**插入配置属性**以显示“**配置属性**”窗口。
2. 在“**配置属性**”窗口中，单击属性，然后单击**添加**。

   注: 最初，没有为代理程序定义任何配置属性。
3. 在“**运行时配置属性**”窗口中，填写下列字段：

   a) 在“**部分**”区域中，填写下列字段：

   标签
   描述属性的文本

   描述
   （可选）属性的描述

   b) 在“**属性**”区域中，填写下列字段：

   标签
   在代理程序配置面板中显示的文本，它标识必须输入的信息。

   环境变量
   环境变量显示在**环境变量**字段中，当您在“标签”字段中输入内容时，环境变量将进行更新。Agent Builder 通过产品代码和标签自动构造环境变量的名称。如果您想独立于标签来更改环境变量，那么可以取消选中**匹配标签**。

   描述
   对所定义属性的描述（可选）。

   类型
   要收集的信息类型，这可以是下列其中一个选项：

   字符串
   适用于必须收集的任何字母信息（例如，安装位置、用户名和主机名）。

   密码
   适用于任何在存储时必须加密的信息。除了提供数据加密之外，还使用星号遮盖您在文本框中输入的数据。此外，您需要输入该信息两次，以验证数据。

   数字
   适用于任何数字信息（例如端口号）。

   选项
   适用于特定值的列表。此选项将启用“选项”表。您可以通过单击**添加**来定义特定的值。输入的值将作为一组选项显示在代理程序配置面板上，您只能从这组选项中选择一项。

   只读文本
   在配置代理程序时显示文本，但不收集信息。

   分隔符
   显示水平分隔符，但不收集信息。

   文件浏览器
   收集文件名字符串。请单击**浏览**来浏览系统文件，以查找所需的文件。

   缺省值
   （可选）指定运行时第一次配置代理程序时在配置面板中显示的值。如果您想使用与 Windows 缺省值不同的 UNIX/Linux 缺省值，请单击**多个值**。

在“**配置属性缺省值**”窗口中，指定要用于 Windows 系统以及 UNIX 和 Linux 系统的缺省值。

注: 多缺省值支持是仅在 IBM Tivoli Monitoring V6.2.1 及更高版本中才受支持的功能。如果您的代理程序与 IBM Tivoli Monitoring V6.2 兼容，那么会有一个提示来提醒您此项需求，并且您可以取消，或者在启用 V6.2.1 兼容性的情况下继续。

必需
如果配置代理程序时用户必须输入值，请选中此字段。如果用户可以选择是否输入值，请取消选中此字段。

c) 要添加选项，请单击**添加**。

4. 在“**配置属性值**”窗口中，请填写**标签**和**值**字段。
“标签”显示为其中一个选项。如果选中此选项，那么该值将成为属性值。
5. 单击**确定**。
新配置部分及属性会显示在“**配置属性**”窗口中的**定制配置**下。
6. 可选： 要将另一个属性添加至现有部分，请选择该部分或者该部分中的现有属性，然后单击**添加**。请在“**配置属性**”窗口的运行时配置树中进行选择。
7. 请填写新属性的字段。（请填写步骤 第 1144 页的『3』 中的那些字段。）
8. 单击**确定**。此时将选中您最近添加的属性。
9. 保留选择，或者选择要插入到日志文件名中的属性。
10. 单击**确定**。此属性将插入到日志文件名中。
然后，您可以继续执行向导，以完成定义日志文件属性组。

注: 尽管配置属性是在日志文件名的上下文中进行定义，但可用于其他位置。例如，另一接受配置属性的位置为脚本数据源。这种灵活性意味着，您可以使用脚本变量 *$K00_APPLICATION_LOG_FILE*（如果产品代码为 K00）来访问配置元素**文件信息**的值。您还可以使用 Windows 批处理文件变量 *%K00_APPLICATION_LOG_FILE%*。

使用 Agent Editor 更改配置属性

使用 Agent Editor 更改代理程序的配置属性。

关于此任务

本任务提供有关使用 Agent Editor 来查看、添加和更改配置属性的信息。

过程

1. 单击**运行时配置**选项卡。
2. 选择配置节，然后单击**添加**。

添加的工作方式与其在第 1143 页的『定制代理程序配置』中的工作方式相同。没有**编辑**选项，这是因为，选中配置节或属性后即可对其进行编辑。

3. 选择配置属性，以显示“**运行时配置详细信息**”区域。
4. 在“**运行时配置详细信息**”区域中，编辑字段以配置属性。

配置 Windows 远程连接

关于配置 Windows 远程连接的信息

关于此任务

Windows Management Instrumentation (WMI)、“Windows 性能监视器 (Perfmon)”及“Windows 事件日志”数据源可以监视已安装代理程序的系统上的数据。这些数据源也可以监视远程 Windows 系统上的数据。这三种数据源类型称为 Windows 数据源。如果这些 Windows 数据源正在以远程方式监视数据，那么它们将共享定义这些数据源的代理程序级别的“Windows 远程连接”配置属性。

如果您在代理程序的基本级别定义 Windows 数据源，那么“Windows 远程连接”配置属性不会自动添加至代理程序。为了保持与可能使用 Windows 数据提供者的先前版本代理程序的兼容性，在启用远程监视之

前，不会添加这些配置属性。代理程序中的 Windows 数据源监视已安装该代理程序的本地 Windows 系统上的数据。

如果您在代理程序中的某个子节点中定义了 Windows 数据源，那么“Windows 远程连接”配置属性会自动添加至该代理程序。如果 Windows 数据源是在子节点中，那么它必须支持“Windows 远程连接”。除非从代理程序的所有子节点中移除所有 Windows 数据源，否则无法取消选中此选项。子节点的每个实例都可以配置为监视不同的远程 Windows 系统。子节点中的所有 Windows 数据源共享“Windows 远程连接”配置属性。

要配置基本代理程序，以便对单个远程 Windows 系统进行远程监视，请完成以下过程。

过程

1. 在 Agent Editor 的“**数据源定义**”窗口中，单击**全局选项**。
   “**全局 Windows 数据源选项**”窗口随即打开。
2. 选择**包括 Windows 远程连接配置**。
3. 单击**确定**。

结果

在 Agent Editor 的“**运行时配置信息**”页面中，可以通过选择 **Windows 远程访问配置** > **Windows 远程连接**访问下列特定于连接的配置属性。

远程 Windows 主机
远程 Windows 计算机的主机名

远程 Windows 密码
远程 Windows 的密码

远程 Windows 域\用户名
远程 Windows 主机的用户名

下一步做什么

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息，请参阅第 1145 页的『使用 Agent Editor 更改配置属性』。如果子节点中定义了 Windows 数据源，您还可以指定“子节点配置覆盖”。有关指示信息，请参阅第 1134 页的『子节点配置』。

创建具有 Windows Management Instrumentation (WMI) 权限的用户

在 Windows 系统上，您可以添加并配置具有权限的用户，以允许进行 WMI 浏览。

关于此任务

如果代理程序通过使用 Windows Management Instrumentation (WMI) 从远程系统收集数据，那么它必须有权访问远程系统上的 WMI 数据。您提供有权访问远程系统上 WMI 数据的帐户的凭证后，代理程序便能够访问该系统上的 WMI 数据。此过程适用于 Windows 7、Windows 2008 Server 和 Windows Vista。

注: 代理程序还可以使用“Windows 性能监视器 (Perfmon)”和“Windows 事件日志”数据源来访问远程 Windows 系统上的数据。但是，对于“Windows 性能监视器 (Perfmon)”和“Windows 事件日志”数据源，您必须提供远程系统的管理员凭证。

过程

1. 创建用户帐户：
   a. 转至 Windows **开始** > **管理工具** > **计算机管理**。“**计算机管理**”窗口随即打开。
   b. 展开**本地用户和组**。
   c. 右键单击**用户**文件夹并选择**新用户**。
   d. 填写用户详细信息，并单击**创建**和**关闭**。
2. 配置新用户帐户的组成员资格：

a. 在“**计算机管理**”窗口中，选择**用户**文件夹。
b. 右键单击新用户帐户，然后选择**属性**。
c. 单击**隶属于**选项卡。
d. 单击**添加**。
e. 单击**高级**。
f. 单击**立即查找**。
g. 选择下列组：
   · Distributed COM Users
   · Performance Log Users
   · Remote Desktop Users

   提示: 如果要选择多个组，请按住 Ctrl 键，然后单击。
h. 单击**确定**，直至返回到“**计算机管理**”窗口为止。
i. 选择**文件 > 退出**以退出“**计算机管理**”窗口。

3. 分配分布式组件对象模型 (DCOM) 权限：
   a. 转至 Windows **开始 > 管理工具 > 组件服务**。“**组件服务**”窗口随即打开。
   b. 展开**组件服务 > 计算机 > 我的电脑**。
   c. 右键单击**我的电脑**并选择**属性**。“**我的电脑属性**”窗口随即打开。
   d. 单击 **COM 安全**选项卡。
   e. 在“**访问权限**”区域中，单击**编辑限制**
   f. 在 **Distributed COM Users** 中，确认已选中**本地访问**和**远程访问**。
   g. 单击**确定**以保存设置。
   h. 在“**我的电脑属性**”窗口中的“**启动和激活权限**”区域中，单击**编辑限制**
   i. 在 **Distributed COM Users** 中，确认已选中**本地启动**、**远程启动**、**本地激活**和**远程激活**。
   j. 单击**确定**以保存设置，然后再次单击**确定**以关闭“**我的电脑属性**”窗口。
   k. 选择**文件 > 退出**以退出“**组件服务**”窗口。

4. 配置 WMI 名称空间安全分配
   a. 转至 Windows **开始 > 运行...**。
   b. 输入 `wmimgmt.msc` 并单击**确定**。
   c. 右键单击 **WMI 控件（本地）**并选择**属性**。
   d. 单击**安全**选项卡。
   e. 单击**安全设置**。
   f. 单击**添加**。
   g. 单击**高级**。
   h. 单击**立即查找**。
   i. 选择新用户帐户，并单击**确定**，直至返回到“**安全设置 Root**”窗口为止。
   j. 单击**高级**并选择新添加的用户帐户。
   k. 单击**编辑**。
   l. 从**应用于：**菜单选项中，选择**这个名称空间和子名称空间**。
   m. 在**执行方法**中，确认已选中**启用帐户**、**远程启用**和**读取安全性**。
   n. 单击**确定**，直至返回到 **wmimgmt** 窗口为止。
   o. 选择**文件 > 退出**以退出 **wmimgmt** 窗口。

下一步做什么

有关从远程系统收集 WMI 数据的更多信息，请参阅第 1030 页的『监视来自 Windows Management Instrumentation (WMI) 的数据』。

配置安全 Shell (SSH) 远程连接

有关配置 SSH 远程连接的信息

关于此任务

脚本数据源可以监视安装了代理程序的系统上的数据，还可以监视远程系统上的数据。如果这些脚本数据源以远程方式监视数据，那么它们将共享定义这些数据源的代理程序级别的 SSH 远程连接配置属性。在启用远程监视之前，代理程序的早期版本可能使用了脚本数据提供程序。为了保持与代理程序的早期版本的兼容性，SSH 远程连接配置属性不会自动添加到代理程序。代理程序中的脚本数据源将监视安装了该代理程序的本地系统上的数据。

如果您在子节点中定义脚本数据源，并选中了**允许使用 SSH 进行数据收集**，那么可以配置每个子节点实例以监视不同的远程系统。子节点中的所有脚本数据源都将共享 SSH 远程连接配置属性。

如果要让代理程序以远程方式监视远程系统，请完成以下过程。

过程

在 Agent Editor 中脚本数据源的“**数据源定义**”窗口中，选中**允许使用 SSH 进行数据收集**。

结果

在 **Agent Editor** 的“**运行时配置信息**”页面中，可以通过选择 **Secure Shell (SSH) 的配置** > **SSH 远程连接** 访问下列特定于连接的配置属性。

网络地址
远程计算机的 IP 地址或主机名。

SSH 端口号
运行 SSH 服务器的 IP 端口号。缺省值是 22。

认证类型
登录到远程 SSH 服务器时要使用的认证类型。您可以选择 Password 或 Public Key。

在每个收集间隔后从远程系统中断开连接
此选项确定在收集数据后，脚本数据提供程序是否断开与远程系统的登录会话。在缺省情况下，值为 No。

在每个收集间隔后从远程系统中移除脚本
此选项用于在每个数据收集间隔之后，从远程系统中删除脚本。在缺省情况下，值为 No。

如果“认证类型”设置为“密码”，那么从 **Agent Editor** 的“**运行时配置信息**”页面中，可以通过选择 **Secure Shell (SSH) 的配置** > **密码**访问下列配置属性：

用户名
远程系统的用户名称

密码
远程系统的密码

如果“认证类型”设置为“公用密钥”，那么从 **Agent Editor** 的“**运行时配置信息**”页面中，可以通过选择 **Secure Shell (SSH) 的配置** > **公用密钥**访问下列配置属性：

用户名
与公用密钥文件相关联的用户名

公用密钥文件
与用户相关联的公用密钥文件

专用密钥文件
与用户相关联的专用密钥文件

密码
用于将专用密钥文件解锁的密码

下一步做什么

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息，请参阅第 1145 页的『使用 Agent Editor 更改配置属性』。如果 SSH 远程连接配置属性包含在子节点中，那么您还可以指定“子节点配置覆盖”。有关指示信息，请参阅第 1134 页的『子节点配置』。

创建工作空间、“执行操作”命令和情境

在 IBM Tivoli Monitoring 环境中安装代理程序之后，您可以为监视解决方案创建工作空间、查询、“执行操作”命令及情境。

您创建的情境、工作空间、“执行操作”命令和查询可以包括在安装程序包中。为了使情境、工作空间和代理程序本身使用同一个安装映像，情境文件和工作空间文件必须与代理程序位于同一个项目中。Agent Builder 提供了在代理程序项目中创建适当文件的向导。有关导入应用程序支持文件的信息，请参阅第 1179 页的『导入应用程序支持文件』。

创建情境、“执行操作”命令和查询

查找信息以帮助创建情境、“执行操作”命令和查询。

要创建情境、“执行操作”命令和查询，请使用 Tivoli Enterprise Portal 和嵌入式情境编辑器。有关如何创建情境的详细信息，请参阅《Tivoli Enterprise Portal 用户指南》。您还可以使用与 Tivoli Enterprise Portal Server 一起安装的帮助文档。Agent Builder 监视代理程序能够识别一组特定的“执行操作”命令，并对其执行特殊处理。有关这些特殊的“执行操作”命令的更多信息，请参阅第 1272 页的『“执行操作”命令参考』。

系统监视器代理程序的情境的创建方式与使用 Tivoli Enterprise Portal 情境编辑器或 **tacmd createSit** 命令创建的企业情境不同。对于系统监视器代理程序，将在代理程序的本地专用情境配置 XML 文件中创建专用情境。有关为系统监视器代理程序创建情境的信息，请参阅 *IBM Tivoli Monitoring Administrator's Guide* 的“Agent Autonomy”一章中的“Private situations”。

创建工作空间

将 Tivoli Enterprise Portal 置于“管理员”方式，以创建可以导出并包括在解决方案中的工作空间。

关于此任务

在使用工作空间的环境中构建工作空间。在构建工作空间时，请更改计算机的显示设置，以环境中通常使用的最低分辨率来构建工作空间。在较高分辨率下构建工作空间可能会创建杂乱的视图，这些视图在较低分辨率下无法正常使用。

要创建可以导出并包括在解决方案中的工作空间，必须将 Tivoli Enterprise Portal 置于“管理员”方式。要将 Tivoli Enterprise Portal 置于“管理员”方式，请完成下列步骤：

过程

1. 转至 *ITM_INSTALL*/CNP 目录并打开 cnp.bat 文件。

   如果您使用的是缺省安装，那么该目录为 C:\IBM\ITM\CNP。在 cnp.bat 文件中，必须更新 set _CMD= %_JAVA_CMD% 行以包括选项 -Dcnp.candle.mode="$_KCJ_$"。

   如果您想要在 Linux 或 AIX 系统上创建扩展，请使用以下路径：

   ```
   /opt/IBM/ITM/li263/cj/bin/cnp.sh
   ```

   其中，*li263* 是运行 Tivoli Enterprise Portal 的操作系统。

更新后的 set_CMD=%_JAVA_CMD % 与以下示例类似：

```
set _CMD= %_JAVA_CMD% -Dcnp.candle.mode="$_KCJ_$" -Xms64m -Xmx256m -showversion -noverify
 -classpath %CPATH% -Dkjr.trace.mode=LOCAL -Dkjr.trace.file=C:\IBM\ITM\CNP\LOGS\kcjras1.log
-Dkjr.trace.params=ERROR -DORBtcpNoDelay=true -Dibm.stream.nio=true
-Dice.net.maxPersistentConnections=16 -Dice.net.persistentConnectionTimeout=1
-Dcnp.http.url.host=SKINANE -Dvbroker.agent.enableLocator=false  -Dnv_inst_flag=%NV_INST_FLAG
%
-Dnvwc.cwd=%NVWC_WORKING_DIR% -Dnvwc.java=%NVWC_JAVA% candle.fw.pres.CMWApplet
```

注: 在此处，此命令仅因为格式编排原因而显示为多行。

2. 打开新的 Tivoli Enterprise Portal Client，并以 sysadmin 用户标识登录。
3. 将 "sysadmin" 用户标识置于“管理员”方式。在 Tivoli Enterprise Portal 中，选择**编辑 > 管理用户**。选择 **sysadmin**，然后在**许可权**选项卡下，选择**工作空间管理**。选中**工作空间管理方式**复选框。

   如果您正确地进行选择，那么桌面标题栏中将显示“***管理方式***”。

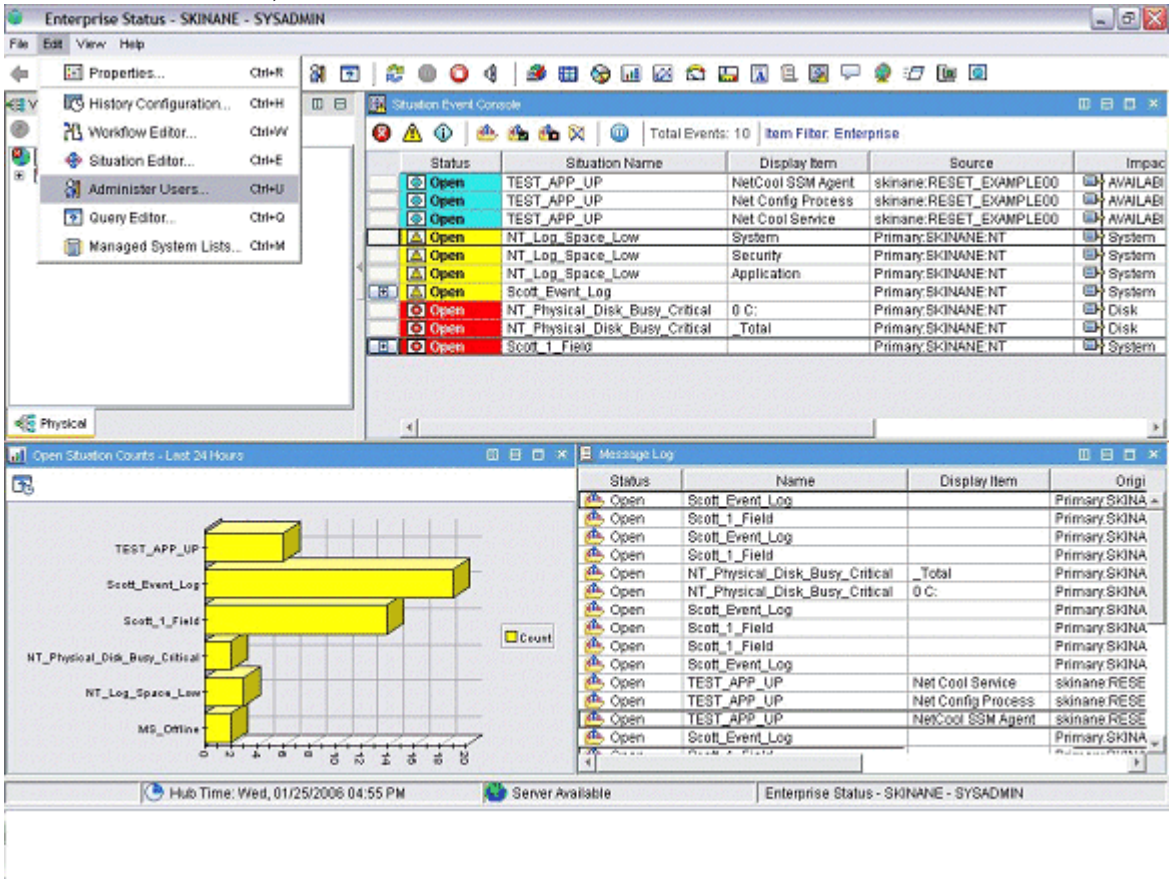


图 *61.* 设置 *sysadmin* 用户标识

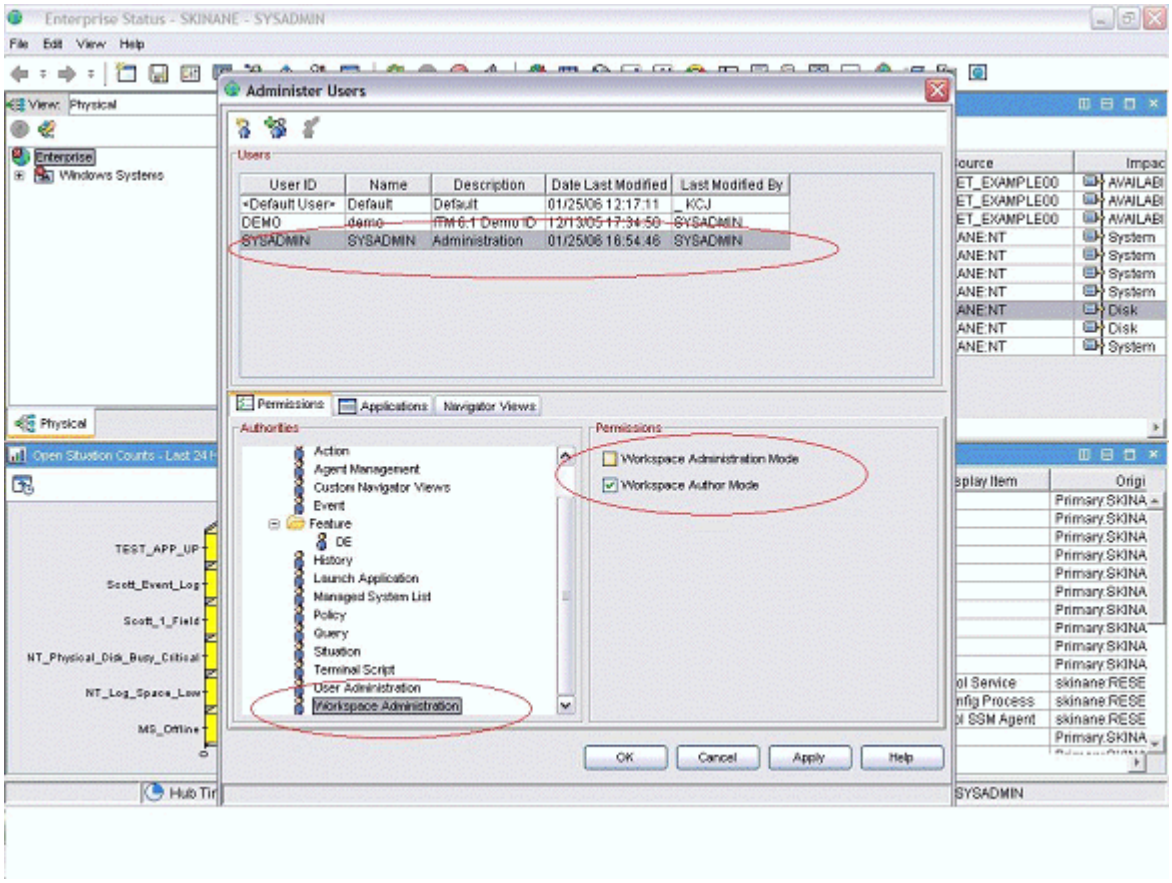


图 62. 设置 *sysadmin* 用户标识（续）

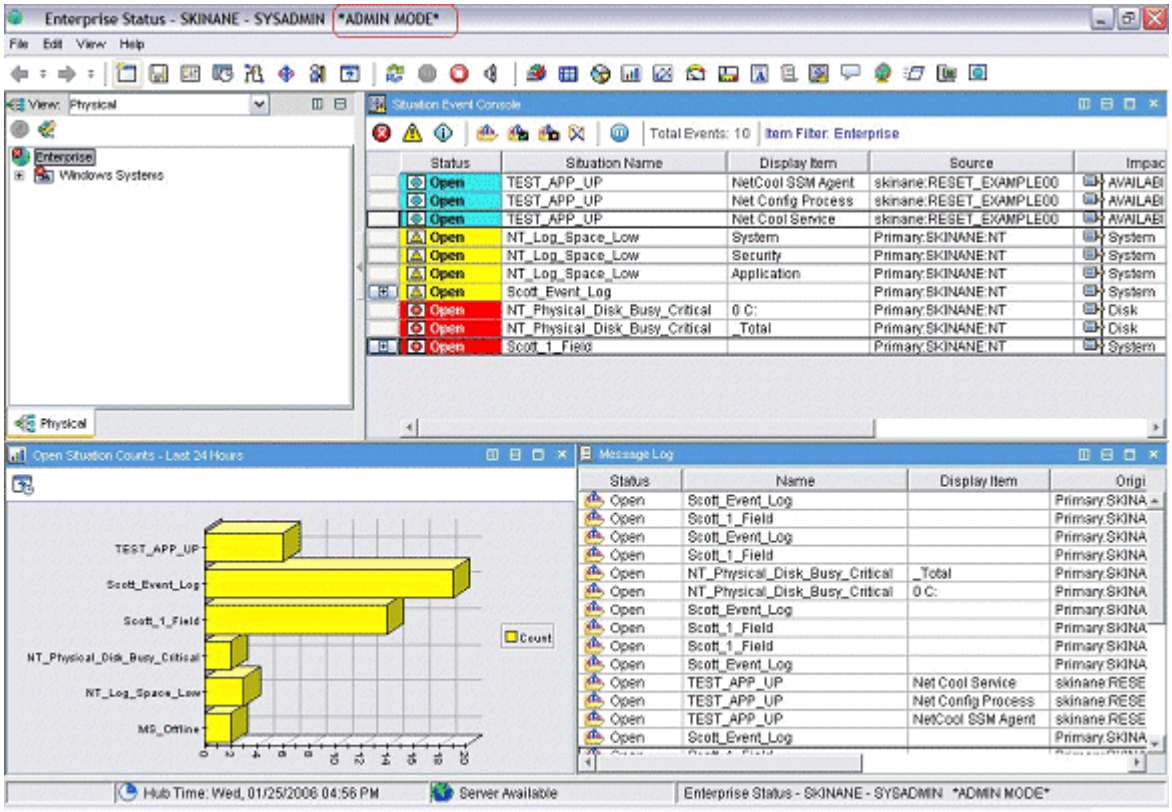


图 63. 设置 *sysadmin* 用户标识（续）

下一步做什么

按第 1152 页的图 63 中的描述进入“管理员”方式之后，您现已准备好为应用程序创建工作空间。有关如何定制和创建工作空间的信息，请参阅《Tivoli Enterprise Portal 用户指南》。另外，也可以使用与 Tivoli Enterprise Portal 组件一起安装的帮助文档。

如果您希望工作空间处于“只读”状态，而不可由客户删除，请为每个工作空间设置“不可编辑”和“不可删除”属性。在工作空间属性中，您必须选择以下属性：

- **不允许修改**
- **IBM 提供的产品（标记为不可删除）**

通过查看工作空间或单击具有控件的图标可转至这些属性。您也可以转至某个视图属性页面，然后转至属性树中的工作空间级别。如果每个导航器项具有多个工作空间，请记住设置用于每个工作空间的属性。如以下示例截屏所示：

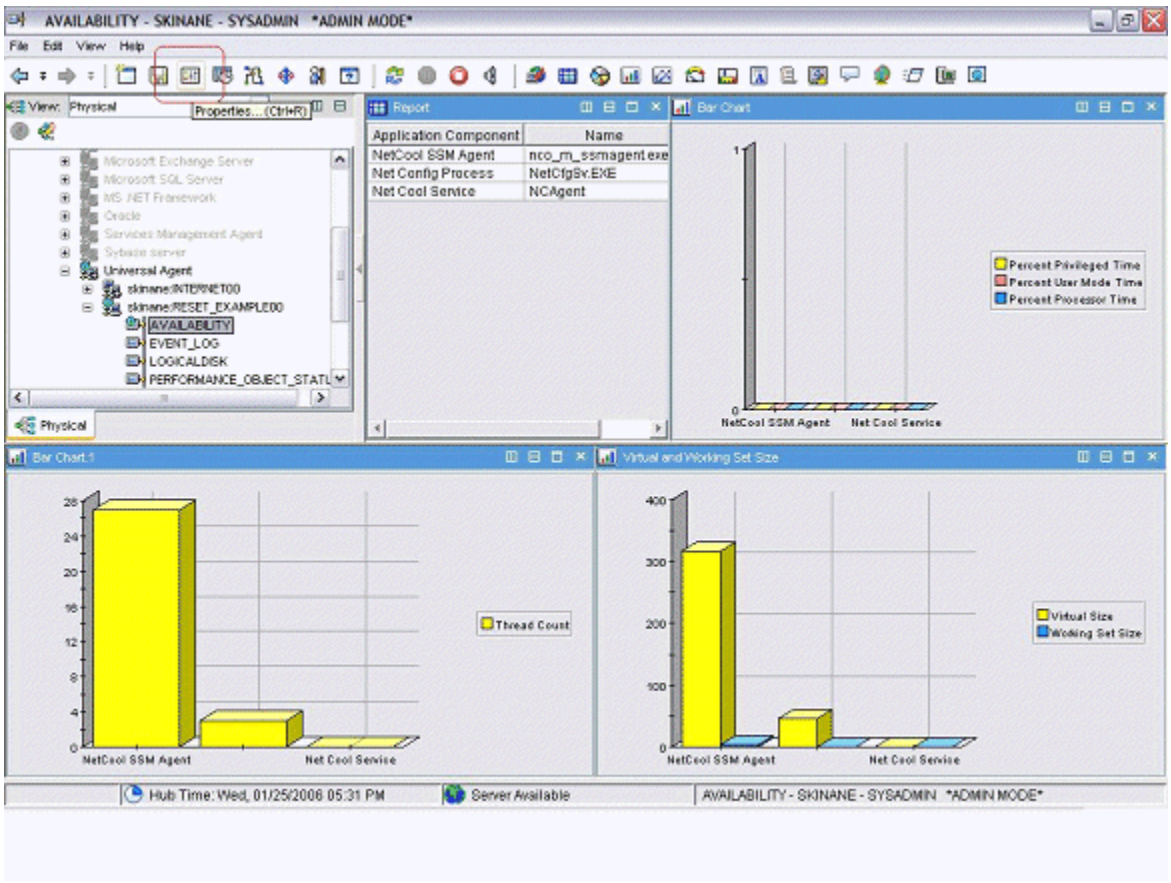


图 *64.* 设置工作空间属性

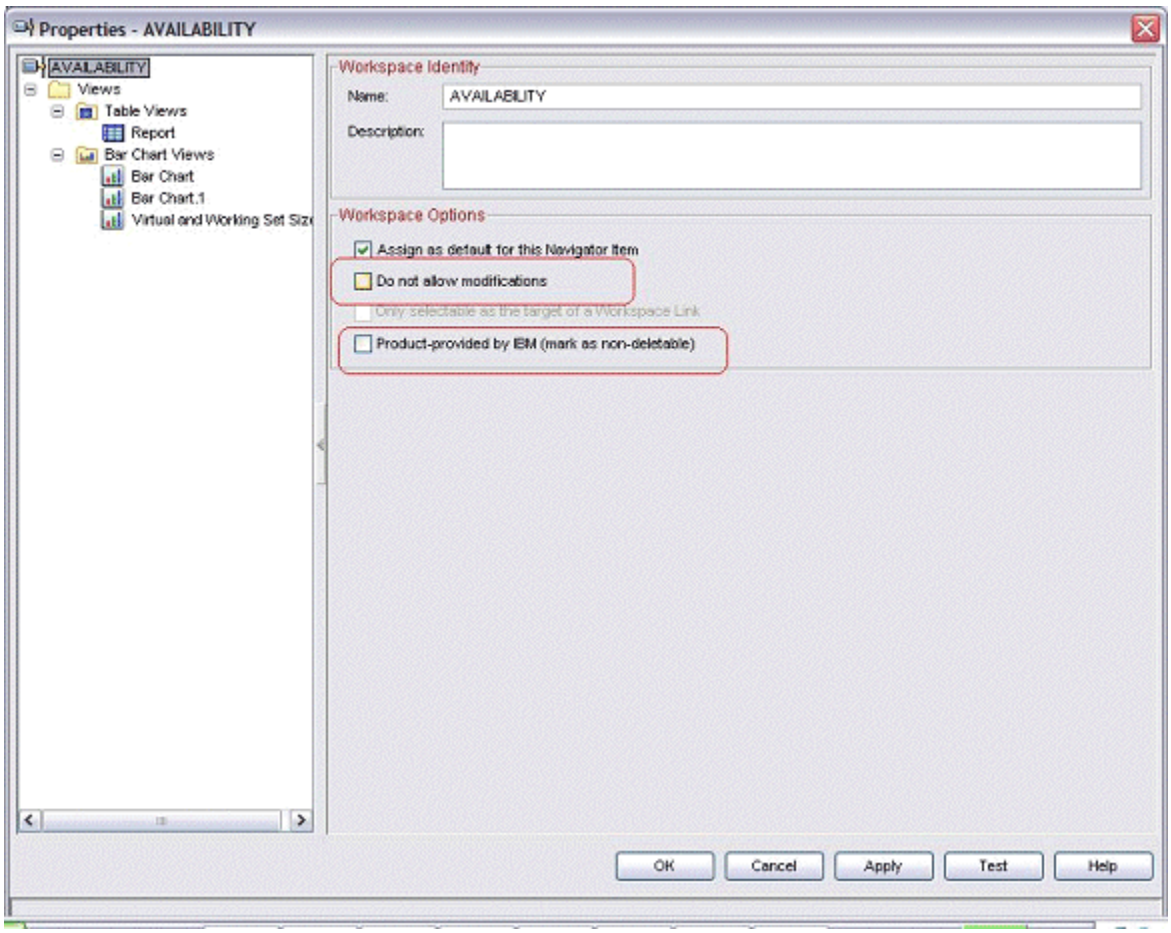


图 65. 设置工作空间属性（续）

为 Cloud APM 准备代理程序

如果要将代理程序与 IBM Cloud Application Performance Management 配合使用，您需要使用“**仪表板设置**”向导来准备该代理程序。此向导将配置信息，您可在 Cloud APM 中的摘要仪表板和详细信息仪表板中查看这些信息。它还设置 Cloud APM 对代理程序所必需的资源信息。

开始之前

为了成功地为 Cloud APM 准备代理程序，您需要确保该代理程序提供下列数据：

- 一个或多个生成一行数据的数据集（属性组）。您可使用这些数据集中的属性来填充摘要仪表板。

  要点: 要在摘要仪表板中包含任何信息，需要在产生单行数据的数据集中提供该信息。一些数据源创建产生多行数据的数据集；例如，进程、Windows 服务和命令返回码数据源会将数据放入产生多行的单个“可用性”数据集中。在此类情况下，需要创建产生单行的已过滤数据集，以便将该数据包含在摘要仪表板中。有关指示信息，请参阅第 1125 页的『创建过滤属性组』。
- 其中一个数据集中的数字属性，用于指示受监视服务的状态（正常、警告、严重或其他类似状态值）。您必须为这个属性定义状态严重性值。有关定义状态严重性值的指示信息，请参阅第 1010 页的『指定用作状态指示符的特性的严重性』。
- 如果受监视应用程序用来提供服务的端口号固定不变，那么您必须知道这个端口。如果这个端口在不同部署之间可能有变化，那么其中一个生成一行数据的数据集必须包含指示此端口的数字字段。
- 如果该代理程序可安装在一台主机上，以监视正在另一主机上运行的服务器，那么其中一个数据集中的某个字符串属性需指示服务器 IP 地址。如果该代理程序始终监视其运行所在的主机，那么不需要这样的属性。

提示: 如果有提供主机名的属性可供使用，那么您可使用 nameToIpAddress 函数为 IP 地址创建派生属性。有关创建派生属性的信息，请参阅第 1005 页的『创建派生属性』。有关此函数的信息，请参阅 第 1016 页的『ipAddressToName』。

如果代理程序包含子节点，那么这些需求适用于要为其创建仪表板的每个子节点。

关于此任务

Cloud APM 监视资源。资源对应于代理程序的实例，或有时对应于子节点。要定义资源，您需要提供应用于受监视服务的资源类型名称、服务器名称、IP 地址和端口号。

Cloud APM 显示每个受监视资源的摘要仪表板。摘要仪表板包含状态指示器；使用这个指示器（通常显示为绿色、黄色或红色，以指示正常、警告或严重状态），用户一眼就可以看到资源状态。同一个仪表板可包含其他一些高级别运行状况度量。

在摘要仪表板上，数据显示为单个的项。因此，包含此数据的数据集必须仅生成一行。

针对代理程序，可以选择性低提供详细信息仪表板。用户单击摘要仪表板即可查看详细信息仪表板。详细信息仪表板可以显示表，因此可以在该仪表板上使用来自任何数据集的数据。

您必须选择摘要仪表板及详细信息仪表板上显示的属性（包括状态指示器）。

要点: 所选属性中的数据每分钟自动从代理程序传递至 Cloud APM 服务器一次。指定过多的数据可能会导致网络、服务器或受监视主机过载。请仅选择所需的属性。例如，如果必须显示连接数据集或派生属性，请勿同时指定源属性。

要点: 除这些属性以外的数据不会传递至 Cloud APM。您在 Cloud APM 中无法查看或使用其他数据，但是在代理程序级别监视的阈值除外。如果在阈值中使用其他数据，则可能无法在 Cloud APM 控制台中查看阈值状态。

过程

1. 从“**代理程序信息**”视图，单击**仪表板**链接。
2. 在**仪表板组件**下，选择**在仪表板中显示代理程序组件**。

   提示: 或者，如果要创建代理程序以专用于 IBM Tivoli Monitoring，可以选择**此代理程序不存在仪表板**。在此情况下，请勿完成此过程的后续步骤。不能将此类代理程序安装在 Cloud APM 环境中。
3. 单击**仪表板设置向导**链接。
4. 如果代理程序包含子节点，定义代理程序和子节点资源在 Cloud APM 中的布局：
   - 选择**基本代理程序实例**以将基本代理程序（子节点之外的数据）显示为资源。
   - 对于每个子节点，选择**子节点“*name*”实例**以将此节点显示为资源。
   - （可选）对于选定的任何子节点，选择**显示为代理程序的子代**。在此情况下，子节点资源会作为代理程序资源下的子代显示在 Cloud APM 控制台中的列表中。

   Cloud APM 会针对您选择的每个组件显示摘要及详细信息仪表板。

   要点: 如果再次运行此向导并取消选择代理程序或子节点，那么不会自动移除此代理程序或子节点的资源。要移除这些资源，请在“大纲”视图中展开**资源**，选择要删除的资源，并按键盘上的 Delete 键。
5. 在**属性选择 - 状态**页面中，选择指示受监视服务的状态的属性。可以使用返回单一数据行的组中的数字属性。

   提示: 或者，如果不想在仪表板中显示状态，取消选择**为此代理程序提供状态**。
6. 在同一个页面中，您可选择是否要在摘要及详细信息仪表板中显示其他数据：
   - 要在摘要仪表板中显示其他高级别运行状况度量，请确保选中**选择其他要在此代理程序的摘要信息中显示的属性**框。否则，请取消选中这个框。
   - 要在详细信息仪表板中显示其他数据，请确保选中**选择其他要在此代理程序的详细信息中显示的属性**框。否则，请取消选中这个框。（通常，请选中这个框，因为需要详细信息仪表板来显示足够多的数据，以使监视代理程序有意义）。

   单击**下一步**。

7. 如果您已选中**选择其他要在此代理程序的摘要信息中显示的属性**，请在**属性选择 - 摘要**页面中最多选择 4 个要包括在摘要仪表板中的其他属性。可以使用返回单一数据行的组中的属性。单击**下一步**。
8. 如果您已选中**选择其他要在此代理程序的详细信息中显示的属性**，请在**属性选择 - 详细信息**页面中选择要包含在详细信息仪表板中的属性。代理程序中的所有属性均可用；为避免发生性能问题，请尽可能包含最少的属性。单击**下一步**。
9. 在**资源类型**页面中，输入要监视的服务器类型，例如 `Email server` 或 `SampleCo Database Server`。单击**下一步**。
10. 在**属性选择 - 软件服务器名称**页面中，在**固定名称**字段中输入固定的软件服务器名称，或者选择代理程序中提供软件服务器名称的属性。此名称会显示给此特定受监视实例的用户，例如 JBoss 应用程序服务器实例的名称。单击**下一步**。

    要点: 在同一台受监视主机上，请勿执行两个或两个以上具有相同软件服务器名称的监视代理程序、代理程序实例或子节点。如果代理程序具有实例或子节点，请确保针对每一个实例或子节点生成唯一的软件服务器名称。如果两个不同的代理程序生成相同的软件服务器名称，请勿将其安装在同一台受监视主机上。
11. 在**属性选择 - IP 地址**页面中，选择代理程序中指定受监视服务器或应用程序使用的主要接口连接 IP 地址（并非主机名）的属性。例如，HTTP 服务器的 HTTP 连接或者数据库服务器的数据库客户机连接。或者，选中**使用代理程序的 IP 地址**，以使用运行代理程序的主机的地址。单击**下一步**。
12. 在**属性选择 - 端口**页面中，输入受监视应用程序用来提供服务的端口，或选择代理程序中指定这个端口的数字属性。单击**完成**。
13. 如果同时选择了代理程序和一个或多个子节点作为资源，请单击**下一步**来输入下一个组件（代理程序或子节点）的仪表板和资源信息。如果**下一步**按钮处于禁用状态，表示您已输入所有必要组件的信息；请单击**完成**来完成向导。

结果

在受监视主机上安装代理程序时，可以在**状态概述**选项卡中查看摘要和详细信息仪表板。

要点: 从安装代理程序到仪表板可用最多可能有 30 分钟延迟，尤其是首次在环境中安装此代理程序类型和版本的情况下。

单击这个代理程序的摘要仪表板，即可查看详细信息仪表板。缺省情况下，详细信息仪表板中的所有信息都显示为表。

您可使用**属性详细信息**选项卡将这些信息的定制显示配置为表和图表。

在 Agent Builder 中测试代理程序

使用 Agent Builder 创建代理程序之后，可以在 Agent Builder 中测试该代理程序。

测试代理程序可确保您所期望的监视数据就是显示的数据。通过测试代理程序，您可以了解如何修改或调整代理程序设置，以确保显示的数据有用且准确。

您可使用以下方法在 Agent Builder 中测试代理程序：

1. 首先，使用 Agent Builder 的属性组测试功能来测试各个属性组，每次测试一个。有关更多信息，请参阅第 1156 页的『属性组测试』。
2. 完成属性组测试之后，可以使用 Agent Builder 的代理程序测试功能来一起测试代理程序中所有的属性组。有关更多信息，请参阅第 1159 页的『代理程序全面测试』。

要点: 在 Agent Builder 中测试代理程序时，您会看到以下数字属性的特殊值：

- -1：一般错误
- -2：缺少数据
- -3：没有值（例如，数据库返回 NULL）

属性组测试

您可以使用属性组测试来测试使用 Agent Builder 创建的代理程序的属性组，每次测试一个属性组。在完成属性组定义之前，您可以测试多个属性组。例如，您可以在定义新代理程序的属性组时，从“**IBM Tivoli**

Monitoring 代理程序向导”启动测试。另外，也可以在向现有代理程序添加属性组时，从“**IBM Tivoli Monitoring 代理程序组件向导**”启动测试。

开始之前

在开始测试属性组之前，您可以选择性地执行下列操作：

· 设置属性组测试首选项。有关更多信息，请参阅第 1158 页的『属性组测试 - 首选项』。

· 设置环境变量、配置属性以及适用的 Java 信息。有关更多信息，请参阅第 1158 页的『属性组测试 - 配置』。

关于此任务

对于大部分数据源，Agent Builder 支持属性组测试功能

过程

· 可以通过以下方法启动测试过程：

1. 在创建代理程序或属性组期间，单击相关数据源信息页面上的**测试**。
2. 创建代理程序之后，在 Agent Editor 的“**数据源定义**”页面上选择属性组，然后单击**测试**。有关 Agent Editor 的更多信息，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

在上述任一步骤中单击**测试**之后，“属性组测试”窗口随即显示。对于不同的数据源，此窗口有所不同。

对于大部分数据源，Agent Builder 支持属性组测试功能。

有关特定属性组的测试过程的更多信息，请参阅随后的“测试”章节：

– Windows Management Instrumentation (WMI)：有关 WMI 测试过程的更多信息，请参阅第 1031 页的『测试 WMI 属性组』

– Windows Performance Monitor (Perfmon)：有关 Perfmon 测试过程的更多信息，请参阅第 1033 页的『测试 Perfmon 属性组』

– 简单网络管理协议 (SNMP)：有关 SNMP 测试的更多信息，请参阅第 1037 页的『测试 SNMP 属性组』

– 简单网络管理协议 (SNMP) 事件发送方：有关 SNMP 事件测试过程的更多信息，请参阅第 1041 页的『测试 SNMP 事件属性组』

– Java 管理扩展 (JMX)：有关 JMX 测试过程的更多信息，请参阅第 1057 页的『测试 JMX 属性组』

– 公共信息模型 (CIM)：有关 CIM 测试过程的更多信息，请参阅第 1060 页的『测试 CIM 属性组』

– 日志文件：有关日志文件测试过程的更多信息，请参阅第 1068 页的『测试日志文件属性组』

– 脚本：有关脚本测试过程的更多信息，请参阅第 1078 页的『监视脚本输出的步骤』

– Java 数据库连接 (JDBC)：有关 JDBC 测试过程的更多信息，请参阅第 1086 页的『测试 JDBC 属性组』

– 因特网控制报文协议 (ICMP) Ping：有关 ICMP 测试过程的更多信息，请参阅第 1088 页的『测试 Ping 属性组』

– 超文本传输协议 (HTTP) 可用性：有关 HTTP 测试过程的更多信息，请参阅第 1094 页的『测试 HTTP 属性组』

– SOAP：有关 SOAP 测试过程的更多信息，请参阅第 1101 页的『测试 SOAP 属性组』

– 传输控制协议 (TCP) 套接字：有关套接字测试过程的更多信息，请参阅第 1109 页的『测试套接字属性组』

– Java 应用程序编程接口 (API)：有关 Java API 测试过程的更多信息，请参阅第 1120 页的『测试 Java 应用程序属性组』

有些数据源不具备属性组测试功能，例如：

– 您可以使用 Agent Builder 浏览器来查看系统上的即时数据。例如，您可以查看当前正在系统上运行的进程。此外，还可以查看安装在系统上的服务（Windows 服务），以及存在的 Windows 事件日志。

– 您在代理程序中没有或基本没有可执行的定制（AIX 二进制日志、命令返回码）。
– 无法使用属性组测试功能来测试连接的属性组以及经过滤的属性组，因为这些组基于多个属性组。

注:

1. 使用代理程序全面测试功能，可以对使用属性组测试功能无法测试的数据源进行测试。有关代理程序全面测试的更多信息，请参阅第 1159 页的『代理程序全面测试』。
2. 在测试数据源时，单击**收集数据**后，数据可能完全不显示，或者可能不是第一次单击后的最新数据。在此类情况下，再次单击**收集数据**即可显示当前数据。

· 调试:

所测试的每个数据源都有一个测试目录，该目录由 Agent Builder 创建。此目录用于该数据源的测试运行时环境。对该数据源运行的测试的相关日志文件存储在该目录下。这些日志文件可以帮助您调试在测试期间发现的问题。

注:

1. 您单击**启动代理程序**以及单击**停止代理程序**之后，测试日志文件的位置就会以状态消息的形式显示在“**测试**”窗口中。
2. 关闭 Agent Builder 时，所有的测试数据源目录随即删除。

属性组测试 - 首选项
在测试属性组之前设置首选项。

关于此任务
在开始测试属性组之前，您可以选择性地设置部分首选项，以确定测试期间处理属性的方式。

过程

1. 从 Agent Builder 菜单栏选择**窗口 > 首选项**。
   “**首选项**”窗口随即打开。
2. 选择 **Agent Builder**。
   与测试属性组相关联的首选项随即显示:

   测试时显示“数据类型已更改”对话框
   选中此选项时，Agent Builder 会提供对属性数据类型的更改建议。当属性的数据类型与该属性的测试所返回的数据不匹配时，Agent Builder 就会提供更改建议。例如，对属性定义的字符串长度太短，无法容纳测试所返回的值。在此示例中，Agent Builder 会建议重新定义属性，增加其字符串长度。取消选中此选项时，Agent Builder 在测试期间不会检查或推荐数据类型。此选项在缺省情况下已选中。

   创建的脚本或日志属性的最大数目
   在此字段中输入的值确定 Agent Builder 在日志文件或脚本属性组的初始测试期间解析的最大属性数。缺省值为 25。
3. 完成设置首选项后，请单击**确定**以保存设置并关闭“**首选项**”窗口。
   如果要复原缺省设置，请先单击**复原缺省值**，然后单击**确定**。

属性组测试 - 配置
在测试属性组之前，请设置环境变量、配置属性和 Java 信息。

关于此任务
在开始测试属性组之前，您可从数据源的“测试”窗口选择性地设置环境变量、配置属性和适用的 Java 信息。Java 信息是配置数据的子集。某些环境变量具有用于属性组测试的特殊值，这些值在缺省情况下已设定。如需进一步了解具有用于属性组测试的特殊值的环境变量，请参阅第 1162 页的『测试环境变量』。

过程

1. 可选： 在数据源的“**测试**”窗口中，单击**设置环境**。

“**环境变量**”窗口随即打开。填写完成后，“**环境变量**”窗口会列出运行测试期间使用的所有环境变量。“环境变量”窗口的初始视图包含代理程序中定义的任何现有环境变量。另外，还会列出您从该代理程序的先前测试中添加的任何环境变量。

a) 单击**添加**或**编辑**可以添加或编辑个别变量。

b) 单击**移除**可以移除个别变量，单击**复原缺省值**可以复原缺省变量并移除所有其他变量。

c) 单击**确定**以保存更改并返回至“**测试**”窗口。

2. 可选： 在数据源的“**测试**”窗口中，单击**配置**。“**运行时配置**”窗口随即打开。

a) 单击**编辑代理程序配置**，即可使用“**配置属性**”窗口添加配置属性或编辑现有的代理程序配置属性。

b) 选择一个配置属性，然后单击**编辑**即可编辑与所要测试的属性组相关的现有配置属性。

c) 选择一个配置属性，然后单击**复原缺省值**可将配置属性复原为它的缺省值。

要点: 如果 JMX 数据源连接到远程 WebSphere Application Server，请确保安装本地 WebSphere Application Server，且 Java 位置设置为该服务器所使用的 JRE。有关设置连接的详细信息，请参阅第 1042 页的『监视 Java 管理扩展 (JMX) MBean』。

3. 单击**确定**以保存更改并返回至“**测试**”窗口。

4. **注:** 可以针对下列类型的属性组设置 Java 信息：

· Java 管理扩展 (JMX)

· Java 数据库连接 (JDBC)

· 超文本传输协议 (HTTP) 可用性

· SOAP

· Java 应用程序编程接口 (API)

Java 信息是步骤 第 1159 页的『2』 中描述的配置数据的子集

可选： 在数据源的“**测试**”窗口中，单击 **Java 信息**。

“**Java 信息**”窗口随即打开。

a) 输入 Java 信息。
例如，浏览至 Java 运行时环境 (JRE) 的位置或输入该位置，选择 **Java 跟踪级别**，或者输入 **JVM 自变量**。

b) 单击**确定**以保存更改并返回至“**测试**”窗口。

代理程序全面测试

使用代理程序全面测试可以同时测试代理程序的所有属性组。您还可以使用代理程序全面测试对使用属性组测试功能无法测试的数据源进行测试。

关于此任务

您可以使用代理程序全面测试以 IBM Tivoli Monitoring 中运行代理程序的方式来运行代理程序，而不需要安装 IBM Tivoli Monitoring。

要点: 在 Windows 系统上，如果要在 Agent Builder 中对代理程序运行全面测试（请参阅第 1159 页的『代理程序全面测试』），请确保在“代理程序信息”窗口中选择运行 Agent Builder 的 32 位版本的操作系统，即 32 位的 Windows。在 Linux 系统上，必须选择 64 位版本。

过程

1. 打开“**代理程序测试**”透视图：

a) 在 Agent Editor 中，打开**代理程序信息**选项卡。

b) 单击**测试代理程序**。

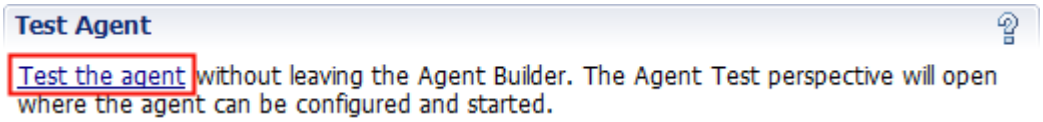


图 66. ***Agent Editor*** 的“**代理程序信息**”页面的“**测试代理程序**”部分。

或者，从 Agent Builder 菜单中选择**窗口 > 打开透视图 > 其他**，选择**代理程序测试**，然后单击**确定**。这样将打开“**代理程序测试**”透视图（第 1161 页的图 68）。“**代理程序测试**”视图显示您在 Agent Editor 中打开的代理程序；您可以测试其中的任何代理程序。还会显示“**属性组测试**”视图；此视图初始为空。“**属性组测试**”视图显示在代理程序运行时从所选属性组收集的数据。

提示: 如果未在编辑任何代理程序，那么“**代理程序测试**”透视图为空。要填充该视图，请转至 **IBM Tivoli Monitoring** 透视图，并在 **Agent Editor** 中打开代理程序。在 **Agent Editor** 中打开代理程序后，请返回到“**代理程序测试**”透视图以测试该代理程序。

2. 可选： 在开始测试前，配置环境变量和配置属性。

   您可以通过两种方法从“**代理程序测试**”视图访问“**环境变量**”和“**运行时配置**”窗口：

   - 在“**代理程序测试**”视图中右键单击代理程序以打开选择菜单。您可以从菜单中选择**设置环境**，以打开“**环境变量**”窗口。您可从菜单中选择**配置**，以打开“**运行时配置**”窗口。
   - 单击“**代理程序测试**”视图工具栏上的“视图菜单”图标  ，以访问与上一选项相同的**设置环境**和**配置**菜单项。

   有关使用“**环境变量**”和“**运行时配置**”窗口的更多信息，请参阅第 1156 页的『属性组测试』。

   要点:

   a. 在代理程序中，将自动填入与测试的每个属性组相关的最近一组配置。

   b. 对于属性组测试和代理程序全面测试，部分环境变量可能具有不同的缺省值。有关将特殊值用于属性组测试的环境变量的更多信息，请参阅（第 1162 页的『测试环境变量』）。

   c. 如果 JMX 数据源连接到远程 WebSphere Application Server，请确保安装本地 WebSphere Application Server，且 Java 位置设置为该服务器使用的 JRE。有关设置连接的详细信息，请参阅第 1042 页的『监视 Java 管理扩展 (JMX) MBean』。

   d. 在 Java API、JDBC、JMX、HTTP 或 SOAP 数据源中，您可以使用 **Java > JVM 自变量**设置来控制代理程序跟踪日志记录。设置以下值：

   ```
   -DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
   ```

   其中，*files* 是保留的最大跟踪日志文件数量（缺省值为 4），*size* 是以千字节为单位的最大日志文件大小（缺省值为 5000）。例如，您可以设置以下值：

   ```
   -DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
   ```

   在此情况下，代理程序将 100 千字节写入第一个日志文件，然后切换到第二个日志文件，以此类推。在写入七个 100 千字节的日志文件之后，将覆盖第一个日志文件。

   e. 如果代理程序包含子节点，那么在已安装版本中，您可以为不同的子节点设置不同的配置值，并可以单独为基本代理程序属性组设置配置值。但是，在全面代理程序测试配置中，每个配置值只能设置一次；该设置应用于基本代理程序以及任何子节点。只能测试每个子节点的一个实例。

3. 在“**代理程序测试**”视图中，选择要测试的代理程序，然后单击  **启动代理程序**图标。

   此时将显示一个窗口，表明代理程序正在启动。代理程序启动后，其属性组会作为代理程序的子代显示在“**代理程序测试**”视图中。属性组通过“属性组”图标  指示。

   用于提供代理程序相关信息的状态属性组（**Performance Object Status**、**Thread Pool Status** 和 **Take Action Status**）也会作为代理程序子代显示在“**代理程序测试**”视图中。状态属性组通过 信息图标指示。

   您可以同时启动并运行多个代理程序。

当代理程序启动时，  **停止代理程序**图标变为可用。

如果代理程序包含子节点或导航器组，它们会作为节点显示在“**代理程序测试**”视图中。子节点定义显示在代理程序下方。子节点实例节点显示在子节点定义节点下方。属性组和导航器组将显示在子节点实例节点下方。例如：

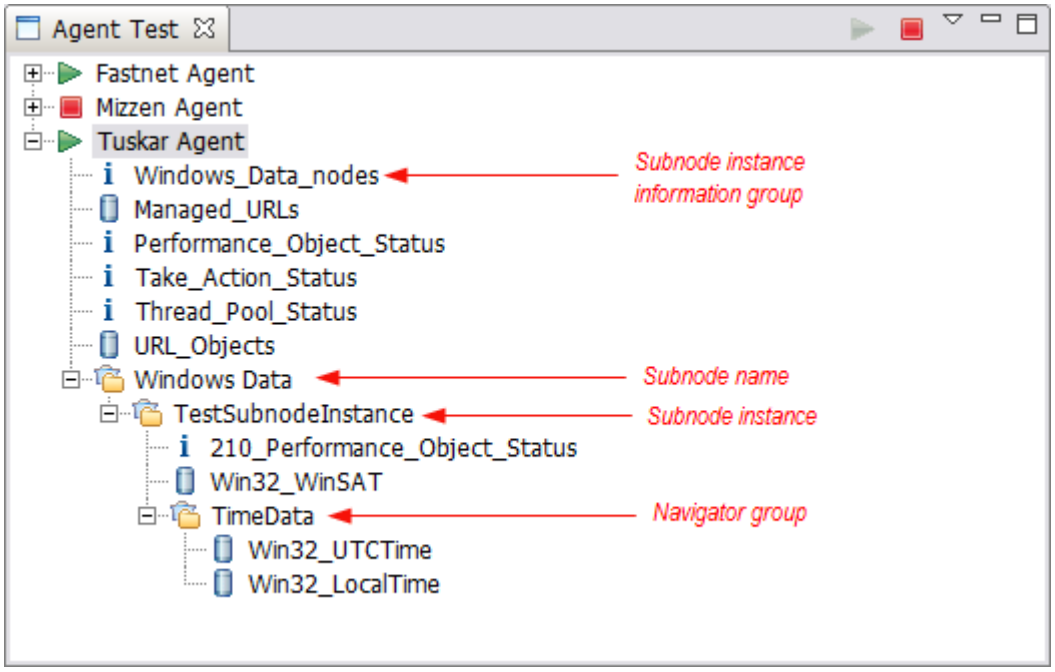


图 67. 突出显示了示例子节点和导航器组的“**代理程序测试**”视图。

您可以右键单击“**代理程序测试**”视图中的任意节点，以访问**编辑**和**停止代理程序**之类的菜单选项。**编辑**将针对 **Agent Editor** 中的选定节点打开“**数据源定义**”。

注: 直到运行中的代理程序停止并重新启动后，您使用 **Agent Editor** 所做的更改才会显示在该代理程序中。

4. 在“**代理程序测试**”视图中，选择要测试的第一个属性组。

选择属性组后，将为所选属性组开始数据收集。如果收集需要一些时间，那么窗口将指示数据收集正在进行中。数据收集完成时，所收集的数据会显示在“**属性组测试**”视图中，例如：

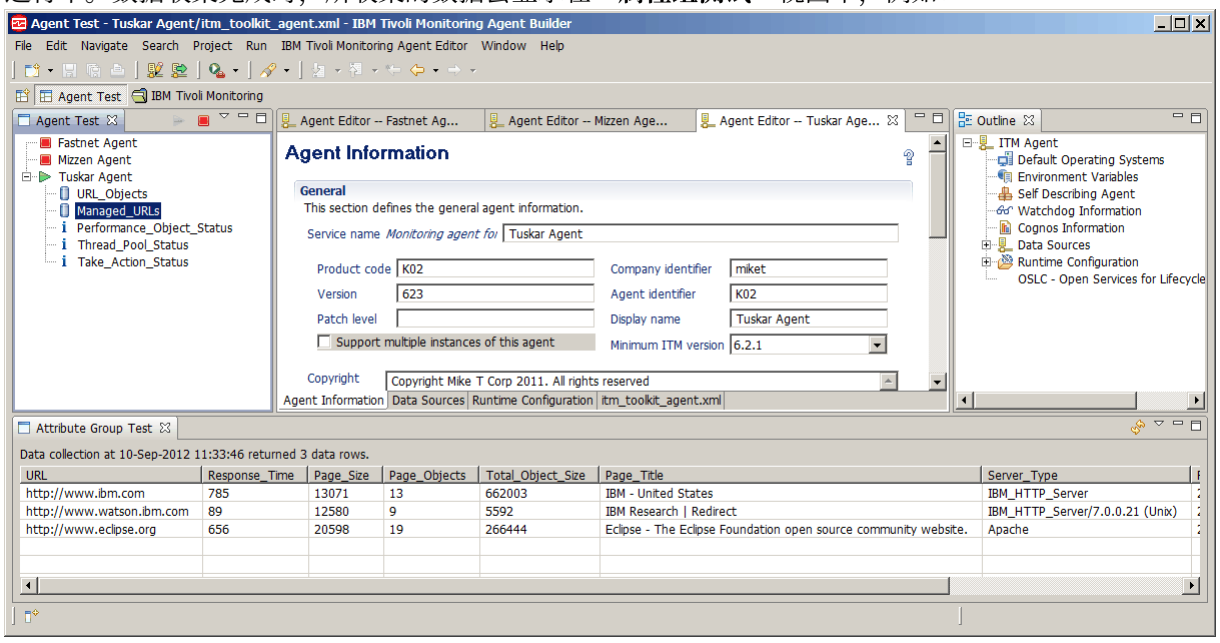


图 68. “**代理程序测试**”透视图

如果未显示数据，则“**属性组测试**”视图中会显示以下消息：返回了 0 个数据行。导致代理程序不返回数据的原因有多种。这些原因包括：

- 没有数据
- 定义不正确
- 配置不正确

您可以通过查看 **Performance Object Status** 属性组中的 **Error_Code** 值来检查未返回数据的原因。有关查看 **Performance Object Status** 属性组的更多信息，请参阅步骤 第 1162 页的『9』。

要收集运行中的代理程序中另一属性组的数据，请选择所需的属性组。

在“**代理程序测试**”视图中选择属性组时，对应的属性组会显示在 **Agent Editor** 视图中。

5. 可选： 对特定属性组类型运行首次数据收集之后，再次运行数据收集以获取有用的数据值。

   要运行数据收集，请单击“**属性组测试**”视图中的“收集数据”图标  。

   如果收集需要一些时间，那么窗口将指示数据收集正在进行中。数据收集完成时，新收集的数据显示在“**属性组测试**”视图中。

6. 可选： 单击“**属性组测试**”视图中的特性列标题以打开 **Agent Editor** 的“**数据源定义**”选项卡中的“**特性信息**”。您也可以通过右键单击表中的任意数据单元格，并从菜单中选择**编辑**来访问同一个“**特性信息**”。

   您可以按常规方式编辑特性的属性。直到运行中的代理程序停止并重新启动后，您所做的更改才会显示在该代理程序中。

7. 可选： 同时打开多个“**属性组测试**”视图。

   要再打开一个“**属性组测试**”视图，请单击“**属性组测试**”视图工具栏上的“视图菜单”图标  ，然后选择**打开属性组的视图**。

   注: 另外打开一个“**属性组测试**”视图时，它将显示与原始“**属性组测试**”视图相同的特性信息。然后，您可以在“**代理程序测试**”视图中选择另一个属性组，以在原始“**属性组测试**”视图中显示不同的属性组信息。首次打开另一个“**属性组测试**”视图时，它会在与原始视图相同的位置处打开，但具有其自己的选项卡。如果您希望同时查看这两个视图，可以将该选项卡拖动到工作空间中的另一位置。

8. 可选： 选择子节点实例信息属性组（如果代理程序具有子节点），以查看子节点在代理程序中的列示方式（第 1161 页的图 67）。选择子节点实例信息属性组会在“**属性组测试**”视图中显示子节点实例信息（针对所选类型的所有联机子节点）。

9. 可选： 要查看有关代理程序运行的更多信息，可以在“**代理程序测试**”视图中选择 **Performance Object Status** 和 **Thread Pool Status** 属性组。这些状态属性组通过“信息”图标 指示。选择这些组可以查看有关属性组的先前数据收集的状态信息。

   例如：

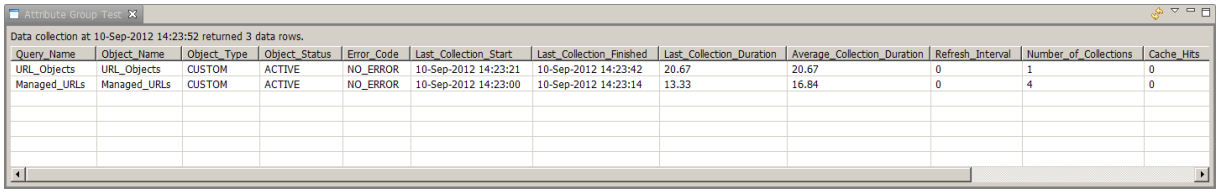


   图 *69.* “**属性组测试**”视图，显示与 ***Managed_URLs*** 和 ***Managed_Nodes*** 属性组的数据收集有关的更多信息 *(Performance Object Status)*。

10. 在完成代理程序测试时，单击“停止代理程序”图标  。

测试环境变量

使用以下环境变量可以控制代理程序在测试期间的行为。

环境变量是动态的具名值，用于确定代理程序的运行方式。对于属性组测试，部分代理程序环境变量设置为特殊值。使用特殊值的目的，是让代理程序以适合测试单个属性组的方式进行响应。对于代理程序全面测试，不使用特殊值，而是使用缺省值。缺省值意味着代理程序的行为与平常一样，这样更适用于代理程序全面测试。

下表汇总具有用于属性组测试的特殊值的环境变量。有关所有代理程序环境变量的更多信息，请参阅第 993 页的『环境变量列表』。有关设置环境变量的更多信息，请参阅第 992 页的『环境变量』。

表 *299.* 环境变量

环境变量	缺省值（代理程序全面测试）	属性组测试值	更改属性组测试值的原因
CDP_DP_INITIAL_COLLECTION_DELAY	不固定	1	此值适用于带有线程池的代理程序。此值是将初始数据收集请求发送到数据提供程序之前，线程池等待的时间（秒）。 **注:** 如果未设置 CDP_DP_INITIAL_COLLECTION_DELAY，那么线程池将等待 CDP_DP_REFRESH_INTERVAL 或 CDP_ATTRIBUTE_GROUP_REFRESH_INTERVAL 所指定的时间。此等待时间就是线程池在两次数据收集之间等待的时间，对于等待第一次数据收集来说可能太长。
CDP_DP_CACHE_TTL	55	1	设置为 1 时，“收集数据”请求极有可能导致数据提供程序立即收集数据。否则，它可能会返回过去最多 60 秒的高速缓存数据。

在监视基础结构中安装代理程序以供测试和使用

在 Agent Builder 中测试代理程序之后，可以将代理程序安装到现有的 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境中，以供进一步测试和使用。

在监视基础结构中安装和测试代理程序有下列优势:

· 可以配置和测试同时运行的多个代理程序实例。

· 可以配置和测试同时运行的多个子节点实例。

· 在 Tivoli Monitoring 环境中，可以在 Tivoli Enterprise Portal 中构建工作空间、情境、操作和查询。

要点: 将代理程序的初始版本部署到监视基础结构的测试版本中。在 Tivoli Monitoring 上，请使用单独的监视服务器和门户网站服务器。在 Cloud APM 上，请使用测试云帐户，或使用本地监视服务器的单独测试部署。将代理程序的最终版本部署在生产基础结构上。

如果将代理程序的某个版本部署在生产监视基础结构上，然后更改该代理程序中的任何数据集，那么新版本可能会与服务器上的旧版本发生冲突。在这种情况下，可能无法使用该代理程序的任何版本。

安装代理程序

可以使用两种方法来安装使用 Agent Builder 创建的代理程序。

1. 要使用与 Agent Builder 在同一系统上运行的监视基础结构来测试代理程序，可以将代理程序安装到本地 Tivoli Monitoring 或 Cloud APM 安装。
2. 要使用不与 Agent Builder 在同一系统上运行的 Tivoli Monitoring 或 Cloud APM 系统来测试或使用代理程序，可以生成能够传输到其他系统的压缩文件（代理程序包）并进行部署。

注:

1. 对于 Tivoli Monitoring，在安装代理程序之后，可以在 Tivoli Enterprise Portal 表中查看性能指标。有关情境或工作空间的支持，请参阅第 1179 页的『导入应用程序支持文件』。
2. 对于 Tivoli Monitoring，在安装代理程序之后，可以使用 Tivoli Enterprise Portal 验证来自代理程序的数据。有关更多信息，请参阅第 1173 页的『Tivoli Enterprise Portal 中的更改』。如果在 Tivoli Enterprise Portal 中查看数据之后要修改代理程序，请参阅第 990 页的『使用 Agent Editor 修改代理程序』。

3. 对于支持 Linux 或 UNIX 的代理程序，请在 Linux 或 UNIX 系统上生成安装程序映像，因为 Linux 或 UNIX 系统会创建具有相应许可权的文件。

本地安装代理程序
将代理程序安装到运行 Agent Builder 的本地系统上的监视环境中。

关于此任务

完成以下步骤以将代理程序安装到本地系统上的监视环境中：

1. 使用下列其中一种方法，在 Agent Builder 的 Project Explorer 导航树中单击 `itm_toolkit_agent.xml` 文件：
   a. 右键单击 `itm_toolkit_agent.xml` 文件并选择 **IBM** > **生成代理程序**。
   b. 选择 `itm_toolkit_agent.xml` 文件，并选择工具栏上的 **生成代理程序**图标。
   c. 双击 `itm_toolkit_agent.xml` 文件并选择 **Agent Editor** > **生成代理程序**。
2. 在“**生成代理程序向导**”窗口的**本地安装代理程序**部分中，输入监视基础结构的安装目录。Agent Builder 填写 `CANDLE_HOME` 环境变量中找到的值。如果未设置此变量，那么将显示 Windows 的缺省值 `C:\IBM\ITM`。

   各个复选框按如下方式启用：

   安装代理程序
   如果 Agent Builder 在指定位置中检测到相应的 Tivoli Enterprise Monitoring Agent 或 IBM Cloud APM 代理程序，那么启用。相应的代理程序是指支持本地操作系统并且处于正确的最低版本的代理程序。

   安装 TEMS 支持
   如果 Agent Builder 在指定位置中检测到 Tivoli Enterprise Monitoring Server，那么在 Tivoli Monitoring 环境中启用。

   安装 TEPS 支持
   如果 Agent Builder 在指定位置中检测到 Tivoli Enterprise Portal Server，那么在 Tivoli Monitoring 环境中启用。
3. 选择要安装的组件（代理程序、Tivoli Enterprise Monitoring Server 支持和 Tivoli Enterprise Portal Server 支持）。
4. 在 Tivoli Monitoring 环境中，如果 Tivoli Enterprise Monitoring Server 或 Tivoli Enterprise Portal Server 安装在本地计算机上，并且您要安装这些服务器的支持文件，那么可以选择是否重新启动服务器。

   在此情况下，“生成代理程序向导”的**本地安装代理程序**部分中的**重新启动 TEMS 而无凭证**和**重新启动 TEPS** 复选框处于活动状态。您可以将这些复选框取消选中，以便在不重新启动服务器的情况下安装此支持。

   在清除**重新启动 TEMS 而无凭证**复选框时，将提示输入 Tivoli Enterprise Monitoring Server 用户标识和密码。输入这些详细信息，然后单击**登录**。如果要在关闭安全性的情况下运行 Tivoli Monitoring，请输入“sysadmin”作为用户标识，密码留空，然后单击**登录**。

   或者，您也可以继续操作而不输入凭证，不指定用户标识和密码并单击**登录**，或者单击**取消**。如果完成这些步骤，那么 Tivoli Enterprise Monitoring Server 将重新启动。

   要点: 要在不重新启动 Tivoli Enterprise Monitoring Server 的情况下安装支持文件，请确保 Tivoli Enterprise Monitoring Server 处于正在运行状态。
5. 选择要生成的代理程序组件。您可以选择**基本代理程序**和/或 **Cognos Reporting**。
6. 在 IBM Cloud APM 环境中，您可以为自描述代理程序提供安全签名。单击**编辑所有 JAR 签名首选项**。您可以将时间戳记添加到签名 JAR 文件并指定添加时间戳记权限。指定有关 Java 密钥库文件的详细信息。

   注: 您必须使用 Java 工具来创建 Java 密钥库文件。例如，要使用 Java 密钥库文件中的对应公用密钥来生成专用密钥和证书，可以运行以下命令：

   ```
   ·ab_install_path/jre/bin/keytool -genkeypair -keystore keystore_file_path -
    storepass key_store_password -alias key_store_alias -dname "CN=common_name,
   ```

```
OU=organizational_unit, L=city_or_locality, ST=state_or_province,
C=country" -keypass key_password
```

其中：

- *ab_install_path* 是 Agent Builder 的安装位置
- *keystore_file_path* 是现有 JKS 密钥存储库的路径，或者是要在其中创建 JKS 密钥存储库的路径
- *key_store_password* 是访问此密钥库中的任何项所需的密码
- *key_store_alias* 是用于在密钥库中识别此密钥的名称（缺省为“mykey”）
- *key_password* 是访问此特定密钥所需的密码（缺省为 key_store_password）

证书必须包括在服务器的密钥库中。

7. 填写完“**JAR 签名**”详细信息时，单击**确定**。
8. 单击**完成**。
9. 配置并启动代理程序。有关更多信息，请参阅 IBM Cloud APM 环境中的第 1167 页的『在 IBM Tivoli Monitoring 环境中配置和启动代理程序』或第 1168 页的『配置代理程序』和第 1169 页的『启动和停止代理程序』。

   对于 Tivoli Monitoring V6.2 FP1 或更高版本，您无需重新启动服务器即可安装 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 支持。在此情况下，“生成代理程序向导”的**本地安装代理程序**部分中的**重新启动 TEMS 而无凭证**和**重新启动 TEPS** 复选框处于活动状态。您可以将这些复选框取消选中，以便在不重新启动服务器的情况下安装此支持。在清除**重新启动 TEMS 而无凭证**复选框时，将提示您输入 Tivoli Enterprise Monitoring Server 用户标识和密码。请输入 Tivoli Enterprise Monitoring Server 用户标识和密码并单击**登录**。如果要在关闭安全性的情况下运行 Tivoli Monitoring，请输入“sysadmin”作为用户标识，密码留空，然后单击**登录**。您也可以继续操作而不输入凭证（不指定用户标识和密码并单击**登录**，或者单击**取消**。这样将导致重新启动 Tivoli Enterprise Monitoring Server）。

   注: 要在不重新启动 Tivoli Enterprise Monitoring Server 的情况下安装支持文件，Tivoli Enterprise Monitoring Server 必须处于运行状态。

创建代理程序包

您可以使用 Agent Builder 来创建压缩代理程序安装包。

关于此任务

代理程序包中包含运行代理程序所需的所有文件以及安装和配置脚本。此安装包还包含监视环境的支持文件。

您可以使用代理程序包将代理程序安装到 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 环境中。

过程

1. 使用下列其中一种方法，在 Agent Builder 的 **Project Explorer** 导航树中单击 itm_toolkit_agent.xml 文件：
   - 右键单击 itm_toolkit_agent.xml 文件并选择 **IBM** > **生成代理程序**。
   - 选择 itm_toolkit_agent.xml 文件，并选择工具栏上的 **生成代理程序**图标。
   - 双击 itm_toolkit_agent.xml 文件并选择 **Agent Editor** > **生成代理程序**。
2. 在**生成代理程序映像**部分中，输入用于存放输出（压缩包或解压缩文件）的目录的名称。
3. 选中**保留中间文件**复选框，以便将生成的解压缩文件与 zip 或 tar 文件分开存放。
4. 选中**创建 ZIP 文件**复选框，以便在指定的目录中创建压缩文件。缺省情况下，对于 Windows 系统，此压缩 zip 文件名为 smai-*agent_name-version*.zip。
5. 选中**创建 TAR 文件**复选框，以便在指定的目录中创建 tar 文件。缺省情况下，对于 UNIX 和 Linux 系统，此压缩 tar 文件名为 smai-*agent_name-version*.tgz。
6. 选择要生成的代理程序组件。您可以选择**基本代理程序**和/或 **Cognos Reporting**。

要点: 对于 IBM Cloud Application Performance Management 环境，请勿选择 **Cognos 报告**，这是因为这些报告目前不受支持，而包括这些报告会增加包的大小。

7. 您可以选择性地提供代理程序应用程序文件的安全签名。如果您想要提供安全签名，请选择**签署自我描述支持 JAR**。单击**编辑所有 JAR 签名首选项**。您可以将时间戳记添加到签名 JAR 文件并指定添加时间戳记权限。指定有关 Java 密钥库文件的详细信息。

   要点: 您可以使用 Java 工具来创建 Java 密钥库文件。例如，要使用 Java 密钥库文件中的对应公用密钥来生成专用密钥和证书，可以运行以下命令：

   - `ab_install_path/jre/bin/keytool -genkeypair -keystore keystore_file_path -storepass key_store_password -alias key_store_alias -dname "CN=common_name, OU=organizational_unit, L=city_or_locality, ST=state_or_province, C=country" -keypass key_password`

   其中：

   - *ab_install_path* 是 Agent Builder 的安装位置
   - *keystore_file_path* 是现有 JKS 密钥存储库所在的路径，或者是要在其中创建 JKS 密钥存储库的路径
   - *key_store_password* 是访问此密钥库中任何项所需的密码
   - *key_store_alias* 是用于在密钥库中唯一标识此密钥的名称（缺省为“mykey”）
   - *key_password* 是访问此特定密钥所需的密码（缺省为 key_store_password）

   请将此证书包括在服务器密钥存储库中。

8. 单击**完成**。

在 IBM Tivoli Monitoring 环境中安装软件包

要在 IBM Tivoli Monitoring 环境中测试和使用代理程序，请使用生成的软件包在受监视的系统、中心 Monitoring Server 系统和 Portal Server 系统上安装代理程序。

开始之前

在受监视的系统上安装代理程序之前，请确保 Tivoli Monitoring 操作系统代理程序已存在并且正常工作。有关安装 Tivoli Monitoring 代理程序的信息，请参阅 Tivoli Monitoring Knowledge Center 中的安装监视代理程序。

要点: 要在 Tivoli Enterprise Portal 中显示代理程序信息，必须安装以下组件：

- 将代理程序安装在所有受监视的系统上
- 将 Tivoli Enterprise Monitoring Server 支持文件安装在中心 Tivoli Enterprise Monitoring Server 上
- 将 Tivoli Enterprise Portal Server 支持文件安装在 Tivoli Enterprise Portal Server 上
- 将 Tivoli Enterprise Portal 支持文件安装在 Tivoli Enterprise Portal Server 和（如果适用）任何 Tivoli Enterprise Portal 桌面客户机上

过程

1. 将缺省情况下名为 *product_code*.zip（针对 Windows 系统）或 *product_code*.tgz（针对 UNIX 和 Linux 系统）的压缩文件复制到要安装代理程序的系统上。
2. 将此文件解压缩至临时位置。

   注: Linux AIX 对于 UNIX 和 Linux 系统，此临时位置不得为 /tmp/*product_code*，其中产品代码为小写。

   可以通过使用压缩文件来远程安装代理程序。

   - Linux 在 Linux 系统上，使用以下命令来解压缩 .tgz 文件：

     ```
     tar -xvzf filename
     ```

- UNIX 在 AIX 系统上，使用以下命令来解压缩 .tgz 文件：

```
gunzip filename
tar -xvf filename
```

3. 运行相应的安装脚本。
   - 要同时安装代理程序、Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal 支持文件：

```
InstallIra.bat/.sh itm_install_location [[-h Hub_TEMS_hostname] -u
HUB_TEMS_username -p Hub_TEMS_password]
```

   - 要在不安装支持文件的情况下安装代理程序：

```
installIraAgent.bat/.sh itm_install_location
```

   - 要安装 Tivoli Enterprise Monitoring Server 支持：

```
installIraAgentTEMS.bat/.sh itm_install_location [[-h Hub_TEMS_hostname] -u
HUB_TEMS_username -p Hub_TEMS_password]
```

   - 要安装 Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal 支持文件：

```
installIraAgentTEPS.bat/.sh itm_install_location
```

   安装位置 *itm_install_location* 必须是第一个自变量，并且在所有脚本中均为必需：`installIra.bat/.sh`、`installIraAgent.bat/.sh`、`installIraAgentTEMS.bat/.sh` 和 `installIraAgentTEPS.bat/.sh`。这是此系统上安装 Tivoli Monitoring 组件的位置。

   其他自变量为可选。

   如果安装 Monitoring Server 支持文件并且不提供用户标识，那么 Tivoli Enterprise Monitoring Server 将重新启动。
4. 配置并启动代理程序，请参阅第 1167 页的『在 IBM Tivoli Monitoring 环境中配置和启动代理程序』。

下一步做什么

如果更改了代理程序的布局，导致移动或移除了导航器项，请重新启动 Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal。重新启动可确保正确识别您所作的更改。

在 IBM Tivoli Monitoring 环境中配置和启动代理程序

在 IBM Tivoli Monitoring 中的受监视系统上安装代理程序后，请配置并启动此代理程序。

过程

1. 打开**管理 Tivoli Monitoring Service**。
   这将显示新条目 **Monitoring Agent for *agent_name***。
2. 右键单击该条目并选择**使用缺省值进行配置**。如果出现提示，请单击**确定**以接受缺省值。

   要点:

   a. 在 UNIX 系统上，应选择**配置**选项。

   b. 在配置多实例代理程序时，将提示输入实例名称。

   提示: 如果您的代理程序使用 JMX 数据源来连接到远程 WebSphere Application Server，请确保 WebSphere Application Server 也安装在运行该代理程序的主机上，并将 `Java home` 设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

   提示: 对于 Java API、JDBC、JMX、HTTP 或 SOAP 数据源，您可以使用 **Java** > **JVM 自变量**设置来控制代理程序跟踪日志记录。在此设置中设置以下值：

```
-DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
```

其中，*files* 是保留的最大跟踪日志文件数量（缺省值为 4），*size* 是以千字节为单位的最大日志文件大小（缺省值为 5000）。例如，您可以设置以下值：

```
-DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
```

在此情况下，代理程序将 100 千字节写入第一个日志文件，然后切换到第二个日志文件，以此类推。在写入七个 100 千字节的日志文件之后，将覆盖第一个日志文件。

如果已将运行时配置元素添加到代理程序，或者已选择数据源，那么系统将显示配置面板。您可以使用这些面板来收集代理程序所需的信息。

3. 右键单击代理程序条目，然后选择**启动**
4. 打开 Tivoli Enterprise Portal 并转至新代理程序。

在 IBM Cloud Application Performance Management 环境中安装和使用代理程序

要在 IBM Cloud Application Performance Management 环境中测试和使用代理程序，请使用生成的软件包在所有受监视的系统上安装代理程序。在某些情况下，您需要先配置代理程序，然后才能将其启动。您可以根据需要启动和停止代理程序。

安装代理程序

使用 Agent Builder 准备的安装程序包在所有受监视系统上安装代理程序。

开始之前

请确保受监视系统上已存在用于 IBM Cloud Application Performance Management 的代理程序（通常是操作系统代理程序），并确保该代理程序正常工作。

Windows 在 Windows 系统上，使用管理员命令行 shell 来安装和配置代理程序。要启动管理员 shell，请从 Windows“程序”菜单选择**命令提示符**并单击右键，然后单击**以管理员身份运行**。

过程

1. 将程序包解压缩到临时目录并切换到此目录。
2. 根据操作系统不同，使用以下命令来安装代理程序：
   - Windows 在 Windows 系统上：`installIraAgent.bat` *agent_install_location*
   - Linux AIX 在 Linux 和 UNIX 系统上：`./installIraAgent.sh` *agent_install_location*

   其中，*agent_install_location* 是现有代理程序的安装位置。缺省位置为：
   - Windows 在 Windows 系统上：`C:\IBM\APM`
   - Linux 在 Linux 系统上：`/opt/ibm/apm/agent`
   - AIX 在 AIX 系统上：`/opt/ibm/apm/agent`

   要点： 如果您已在 Agent Editor 的“**运行时配置**”窗口中添加了任何定制配置属性，如果代理程序支持多个实例，或者如果代理程序使用任何需要配置（例如，用户标识和密码）的预定义数据源，那么您必须先配置代理程序，然后它才能启动。如果代理程序不需要配置，那么完成安装之后，它将自动启动。

配置代理程序

如果您已在 Agent Editor 的“运行时配置”窗口中添加了任何定制配置属性，如果代理程序支持多个实例，或者如果代理程序使用任何需要配置（例如，用户标识和密码）的预定义数据源，那么您必须先配置代理程序，然后它才能启动。

开始之前

Windows 在 Windows 系统上，使用管理员命令行 shell 来安装和配置代理程序。要启动管理员 shell，请从 Windows“程序”菜单选择**命令提示符**并单击右键，然后单击**以管理员身份运行**。

关于此任务

在配置过程中，您可以执行以下操作：

· 如果代理程序支持多个实例，可以设置实例名称以创建或更改实例。

· 设置任何对代理程序可用的配置属性。

· 如果代理程序支持子节点，创建和配置子节点。

Windows 在 Windows 系统上，要设置任何配置属性或创建任何子节点，必须使用静默配置过程。样本静默配置响应文件位于 *install_dir*\samples 目录中，名为 *agentname*_silent_config.txt。创建此文件副本，并根据需要设置配置变量。

Linux AIX 在 Linux 和 UNIX 系统上，您可以选择使用静默配置过程。或者，可以使用交互式过程。如果不使用响应文件名称来启动配置命令，配置实用程序将提示输入配置值。

过程

1. 切换到 *install_dir*/bin 目录。
2. 运行以下命令以配置代理程序：
   · 如果代理程序不支持多个实例：
     – Windows 在 Windows 系统上：*name*-agent.bat config [*response_file*]
     – Linux AIX 在 Linux 和 UNIX 系统上：./*name*-agent.sh config [*response_file*]
   · 如果代理程序支持多个实例：
     – Windows 在 Windows 系统上：*name*-agent.bat config *instance_name* [*response_file*]
     – Linux AIX 在 Linux 和 UNIX 系统上：./*name*-agent.sh config *instance_name* [*response_file*]

   其中：

   · *instance_name* 是实例名称。如果不存在具有此名称的实例，将创建此实例。如果已存在此实例，将重新配置此实例。您必须至少创建一个实例来使用代理程序。

   · *response_file* 是静默配置响应文件的名称。

   提示： 如果您的代理程序使用 JMX 数据源来连接到远程 WebSphere Application Server，请确保 WebSphere Application Server 也安装在运行该代理程序的主机上，并将 Java home 设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

   提示： 对于 Java API、JDBC、JMX、HTTP 或 SOAP 数据源，您可以使用 **Java** > **JVM 自变量**设置来控制代理程序跟踪日志记录。在此设置中设置以下值：

   ```
   -DJAVA_TRACE_MAX_FILES=files -DJAVA_TRACE_MAX_FILE_SIZE=size
   ```

   其中，*files* 是保留的最大跟踪日志文件数量（缺省值为 4），*size* 是以千字节为单位的最大日志文件大小（缺省值为 5000）。例如，您可以设置以下值：

   ```
   -DJAVA_TRACE_MAX_FILES=7 -DJAVA_TRACE_MAX_FILE_SIZE=100
   ```

   在此情况下，代理程序将 100 千字节写入第一个日志文件，然后切换到第二个日志文件，以此类推。在写入七个 100 千字节的日志文件之后，将覆盖第一个日志文件。

启动和停止代理程序

要监控系统，请确保在系统上已启动此代理程序。您可以随时启动和停止代理程序。如果代理程序支持多个实例，可以单独启动和停止每个实例。

过程

1. 切换到 *install_dir*/bin 目录。
2. 运行以下命令以启动代理程序：

- 如果代理程序不支持多个实例：
  - Windows 在 Windows 系统上：`name-agent.bat start`
  - Linux AIX 在 Linux 和 UNIX 系统上：`./name-agent.sh start`
- 如果代理程序支持多个实例：
  - Windows 在 Windows 系统上：`name-agent.bat start instance_name`
  - Linux AIX 在 Linux 和 UNIX 系统上：`./name-agent.sh start instance_name`

3. 运行以下命令以停止代理程序：
   - 如果代理程序不支持多个实例：
     - Windows 在 Windows 系统上：`name-agent.bat stop`
     - Linux AIX 在 Linux 和 UNIX 系统上：`./name-agent.sh stop`
   - 如果代理程序支持多个实例：
     - Windows 在 Windows 系统上：`name-agent.bat stop instance_name`
     - Linux AIX 在 Linux 和 UNIX 系统上：`./name-agent.sh stop instance_name`

生成及安装代理程序之后的结果

安装 Agent Builder 代理程序会创建和更改系统上的某些文件。在 IBM Tivoli Monitoring 环境中，您还会看到 Tivoli Enterprise Portal 中的更改。

系统上的新文件

生成并安装使用 Agent Builder 创建的代理程序之后，您在代理程序系统上可以看到下列新文件：

注: *xx* 表示两个字符的产品代码。

Windows

Windows 系统:

TMAITM6\k*xx*agent.exe
代理程序二进制文件

TMAITM6\K*xx*ENV
环境变量设置

TMAITM6\K*xx*.ref
代理程序提供程序配置

TMAITM6\SQLLIB\k*xx*.his
代理程序属性信息的 SQL 描述

TMAITM6\SQLLIB\k*xx*.atr
代理程序属性信息

TMAITM6*xx*_dd_*version*.xmll
产品描述

TMAITM6*xx*_dd.properties
产品名称

TMAITM6\kxxcma.ini
代理程序服务定义文件

TMAITM6*your files*
通过 Java API 或套接字数据源附带的补充文件，其文件类型为“可执行文件”或“库”。通过脚本或命令返回码数据源包括的脚本。

Linux AIX

UNIX/Linux 系统:
registry/*xxarchitecture*.ver
内部版本和必备文件

***architecture*/*xx*/bin/*xx*_dd_*version*.xml**
产品描述

***architecture*/*xx*/bin/k*xx*agent**
代理程序二进制文件

***architecture*/*xx*/bin/*xx*_dd.properties**
产品名称

***architecture*/*xx*/work/k*xx*.ref**
代理程序提供程序配置

***architecture*/*xx*/tables/ATTRLIB/k*xx*.atr**
代理程序属性信息

***architecture*/*xx*/hist/k*xx*.his**
代理程序属性信息的 SQL 描述

architecture*/*xx*/bin/*your files
通过 Java API 或套接字数据源附带的补充文件，其文件类型为“可执行文件”。通过脚本或命令返回码数据源包括的脚本。

architecture*/*xx*/lib/*your files
通过 Java API 或套接字数据源附带的补充文件，其文件类型为“库”。

config/.*xx*.rc
内部设置文件

config/*xx*.environment
环境设置

config/*xx*_dd_*version*.xml
产品描述

config/*xx*_dd.properties
产品名称

config/.ConfigData/k*xx*env
环境变量设置

注: 请运行以下命令，以确定系统的体系结构:

```
cinfo -p xx
```

其中 *xx* 是两个字符的产品代码。

例如，对于以产品代码 19 来运行代理程序 的 64 位 Solaris 8 系统，输出如下所示:

```
# /opt/ibm/apm/agent/bin/cinfo -p 19

*********** Fri Aug 17 11:23:58 EDT 2007 ******************
User : root Group: other
Host name : guadalajara  Installer Lvl:06.20.00.00
CandleHome: /opt/IBM/ITM
************************************************************
Platform codes:
sol286 : Current machine
sol286 : Product (19)
tmaitm6/sol286 : CT Framework (ax)
```

以粗体显示的行是相关行。冒号之前的字符串 sol286 指示用于此代理程序的体系结构。对于操作系统与计算机硬件的不同组合，此字符串有所不同。只有在先前已安装代理程序的情况下，此功能才有效。

下列文件用于基于 Java 的数据源。只有在代理程序包含 JMX、JDBC、HTTP 或 SOAP 数据源的情况下，才会创建这些文件:

- cpci.jar
- jlog.jar

- common/jatlib-1.0.jar

下列文件用于 JMX 运行时支持。只有在代理程序包含 JMX 数据源的情况下，才会创建这些文件：

- common/jmx-1.0.jar
- common/connectors/jboss/connJboss-1.0.jar
- common/connectors/jsr160/connJSR160-1.0.jar
- common/connectors/was/connWas-1.0.jar
- common/connectors/weblogic/connWeblogic-1.0.jar

下列文件用于 JDBC 运行时支持。只有在代理程序包含 JDBC 数据源的情况下，才会创建这些文件：

- common/jdbc-1.0.jar

下列文件用于 HTTP 或 SOAP 运行时支持。只有在代理程序包含 HTTP 或 SOAP 数据源的情况下，才会创建这些文件：

- http-1.0.jar

下列文件用于 Java API 运行时支持。只有在代理程序包含 Java API 数据源的情况下，才会创建这些文件：

- cpci.jar
- custom/*your JAR file*：此 JAR 文件的名称在 Java API 数据源的**全局设置**中指定。
- custom/*your JAR file*：文件类型为 Java 资源的补充文件。

对于基于 Java 的数据源，在 Windows、UNIX 和 Linux 系统上存在相同的文件，但它们位于不同的目录中：

- Windows Windows 路径：TMAITM6\kxx\jars
- Linux AIX UNIX/Linux 路径：*architecture*/*xx*/jars

下列文件用于日志文件监视运行时支持。只有在代理程序包含日志文件数据源的情况下，才会创建这些文件：

- Windows 在 Windows 系统上：TMAITM6\kxxudp.dll
- Linux 在 Solaris/Linux 系统上：*architecture*/*xx*/lib/libkxxudp.so
- 在 HP-UX 系统上：*architecture*/*xx*/lib/libkxxudp.sl
- UNIX 在 AIX 系统上：*architecture*/*xx*/lib/libkxxudp.a

下列文件用于 SSH 脚本监视运行时支持。只有在代理程序包含已启用 SSH 收集的脚本数据源的情况下，才会创建这些文件：

- Windows 在 Windows 系统上：TMAITM6\kxxssh.dll
- Linux 在 Solaris/Linux 系统上：*architecture*/*xx*/lib/libkxxssh.so
- 在 HP-UX 系统上：*architecture*/*xx*/lib/libkxxssh.sl
- UNIX 在 AIX 系统上：*architecture*/*xx*/lib/libkxxssh.a

管理 Tivoli Enterprise Monitoring Services 窗口中的更改

在 IBM Tivoli Monitoring 环境中安装代理程序之后，您可以在“**管理 Tivoli Enterprise Monitoring Services**”窗口中看到一个代理程序条目。该条目名称为 **Monitoring Agent for *agent_name***。

要点：“**管理 Tivoli Enterprise Monitoring Services**”在 IBM Cloud Application Performance Management 环境中不受支持。

Windows 在 Windows 系统上，此条目包含**任务/子系统**列，用于标识代理程序是否支持多个实例：

- 单实例代理程序将在“**管理 Tivoli Enterprise Monitoring Services**”窗口中显示新的应用程序。此应用程序名称为 **Monitoring Agent for *agent_name***。针对此代理程序，将创建一个服务（第 1173 页的图 70）。**任务/子系统**列包含值**主要**。

· 多实例代理程序将在“**管理 Tivoli Enterprise Monitoring Services**”窗口中显示新的应用程序模板。此模板名称为 **Monitoring Agent for *agent_name***。根据此模板创建此代理程序的实例之后，才会为此代理程序创建服务。**任务/子系统**列包含值**模板**，以指示此条目是用于创建代理程序实例的模板。

Linux AIX 在 Linux 和 UNIX 系统上，无论代理程序是否支持多个实例，代理程序的条目都相同。

注: 下列屏幕用于 Windows 系统。在 UNIX 和 Linux 系统上，存在类似的屏幕。

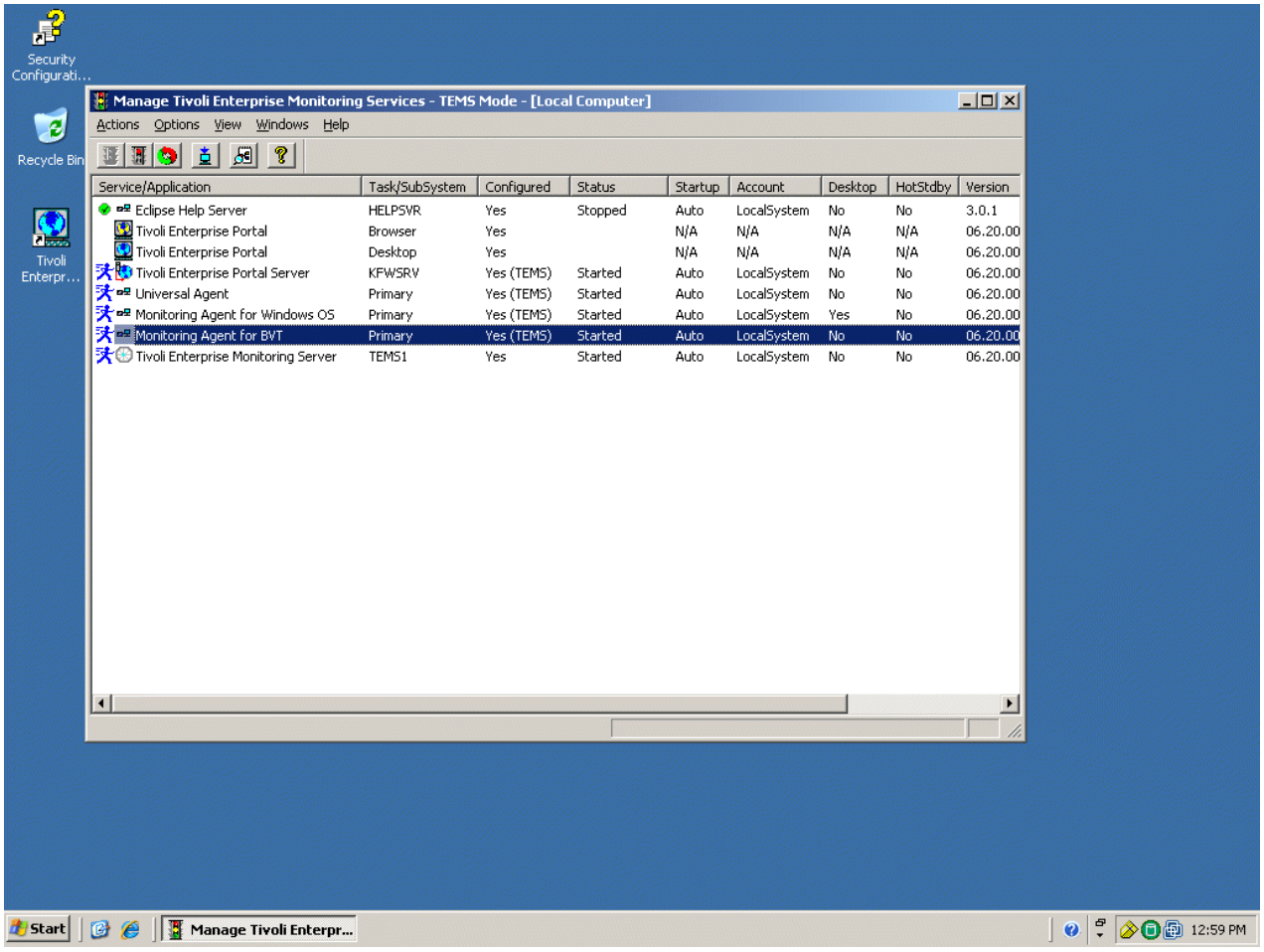


图 *70.* “管理 *Tivoli Enterprise Monitoring Services*”窗口

Tivoli Enterprise Portal 中的更改

在 IBM Tivoli Monitoring 环境中，安装并启动代理程序后，单击 Tivoli Enterprise Portal 中的绿色**刷新**图标。然后，您就可以查看新的代理程序。您在门户网站中可以看到以下更改:

· Tivoli Enterprise Portal 物理视图中新的代理程序子节点。

· 您使用 Agent Builder 定义的每个导航器组及数据源的对应节点（第 1174 页的图 71）。

注: 对于每个导航器项，您必须定义一个缺省查询。

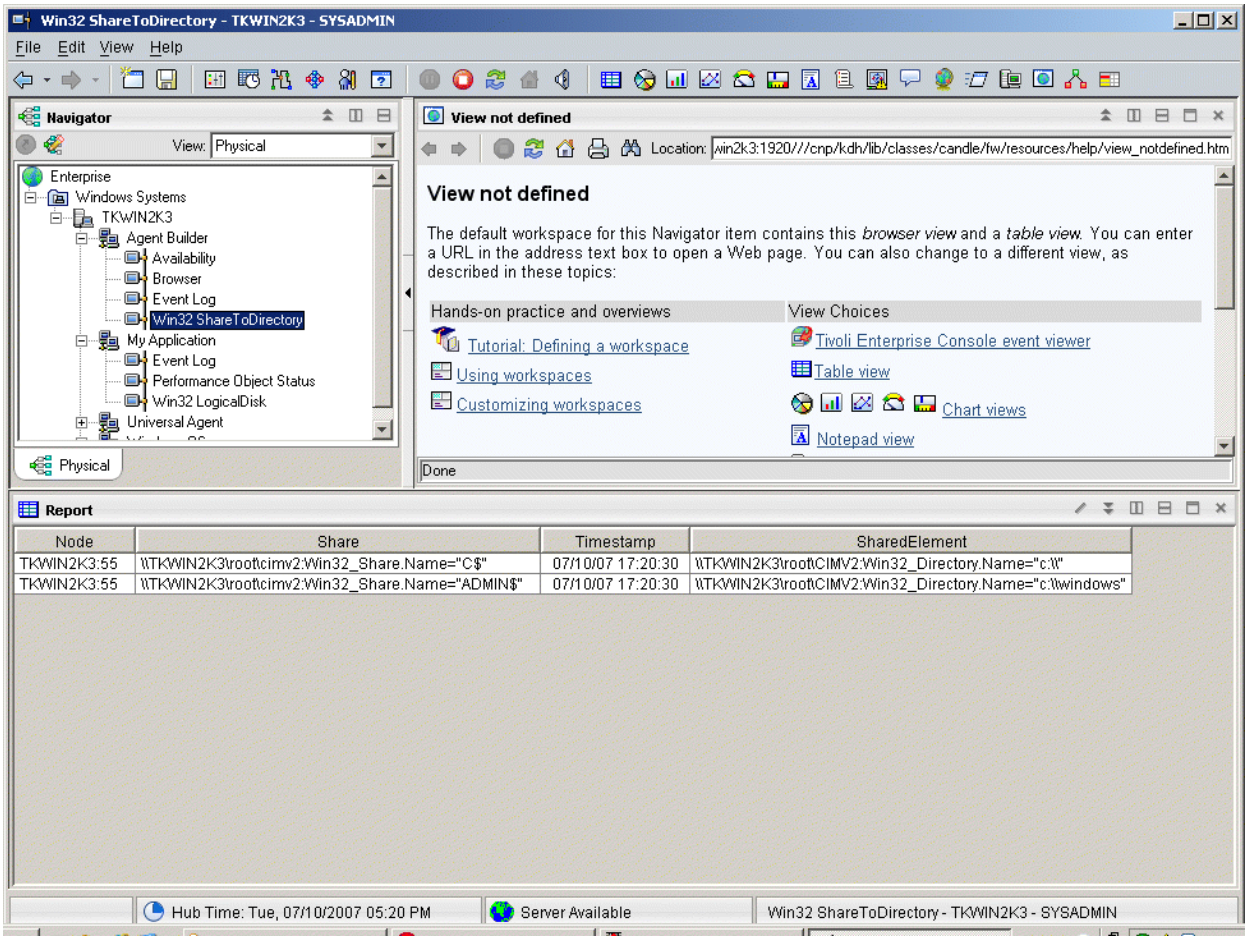


图 71. 新代理程序中属性组的节点。

- 如果代理程序包含子节点，那么对于代理程序中定义的每个子节点，都表现为一个可展开的节点。在可展开的节点之下，将显示下列节点：
  - xxx 性能对象状态，其中 xxx 是以 3 个字母表示的子节点类型
  - 您在子节点中定义的每个导航器组和数据源的对应节点
  - xxx 事件日志节点（如果有事件日志）
  - xxx JMX 监视器节点（如果存在 JMX 且包括了 JMX 监视器）
- 下列自动节点：
  - 可用性节点（如果代理程序包含可用性数据源）（第 1175 页的图 72）

    注: 根据代理程序的内容不同，此节点的行为也会有所变化。如果代理程序仅监视可用性，那么可用性节点表示可用性数据源。如果代理程序监视可用性及性能，则可用性节点会变为导航器项，该项代表可用性及性能对象状态数据源。

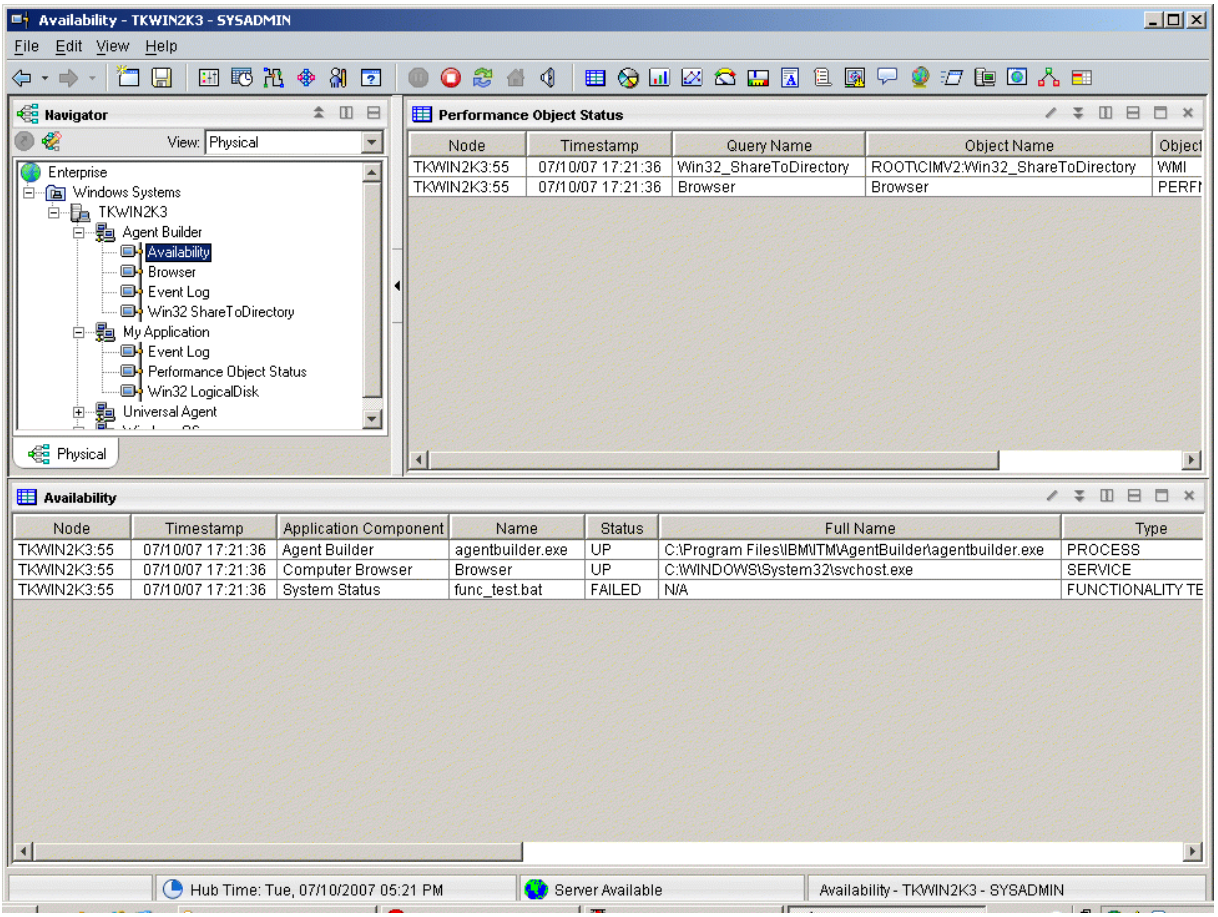


图 72. “可用性”节点

– 性能对象状态，前提是代理程序包含性能监视（非可用性）数据源（第 1176 页的图 73）

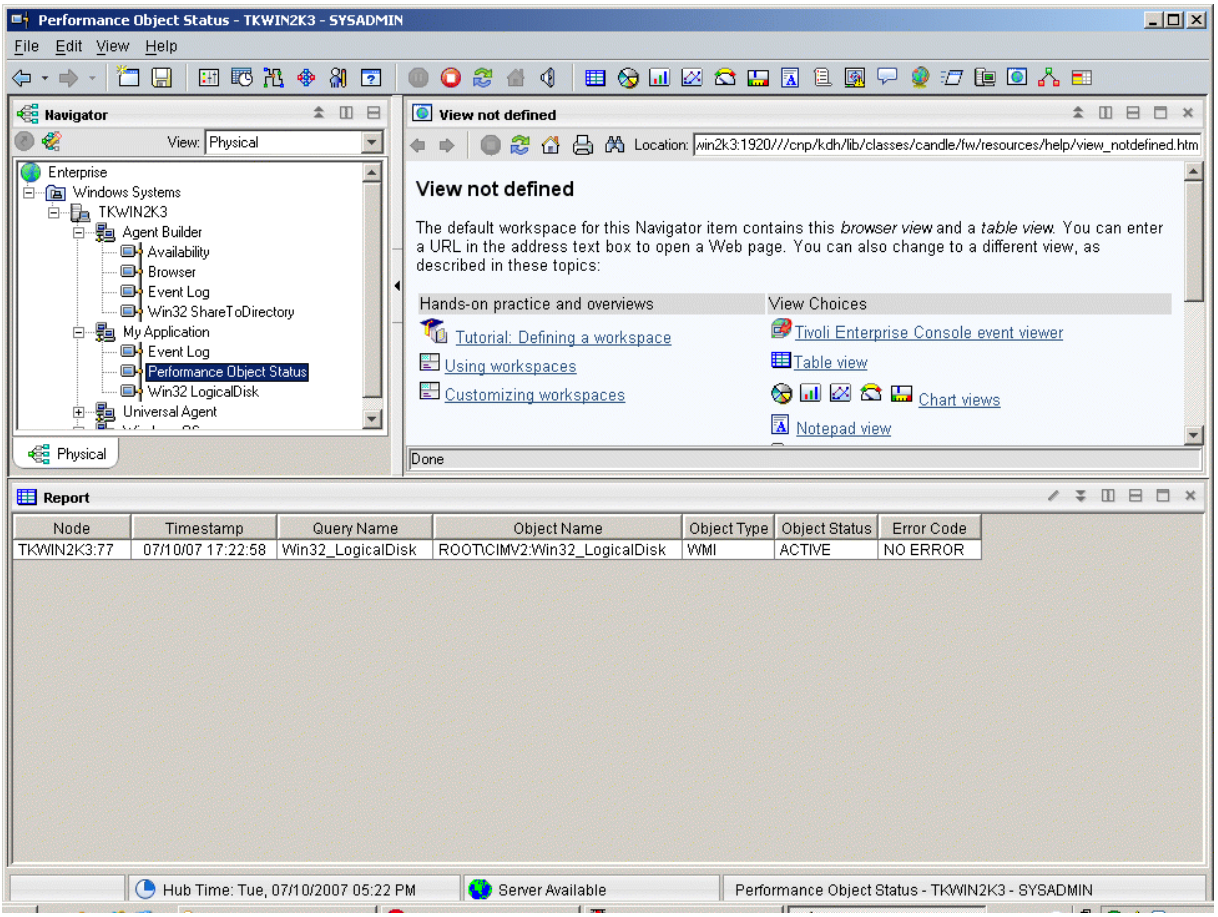


图 73. “性能对象状态”节点

– 事件日志，前提是代理程序包含数据源生成日志数据（第 1177 页的图 74）

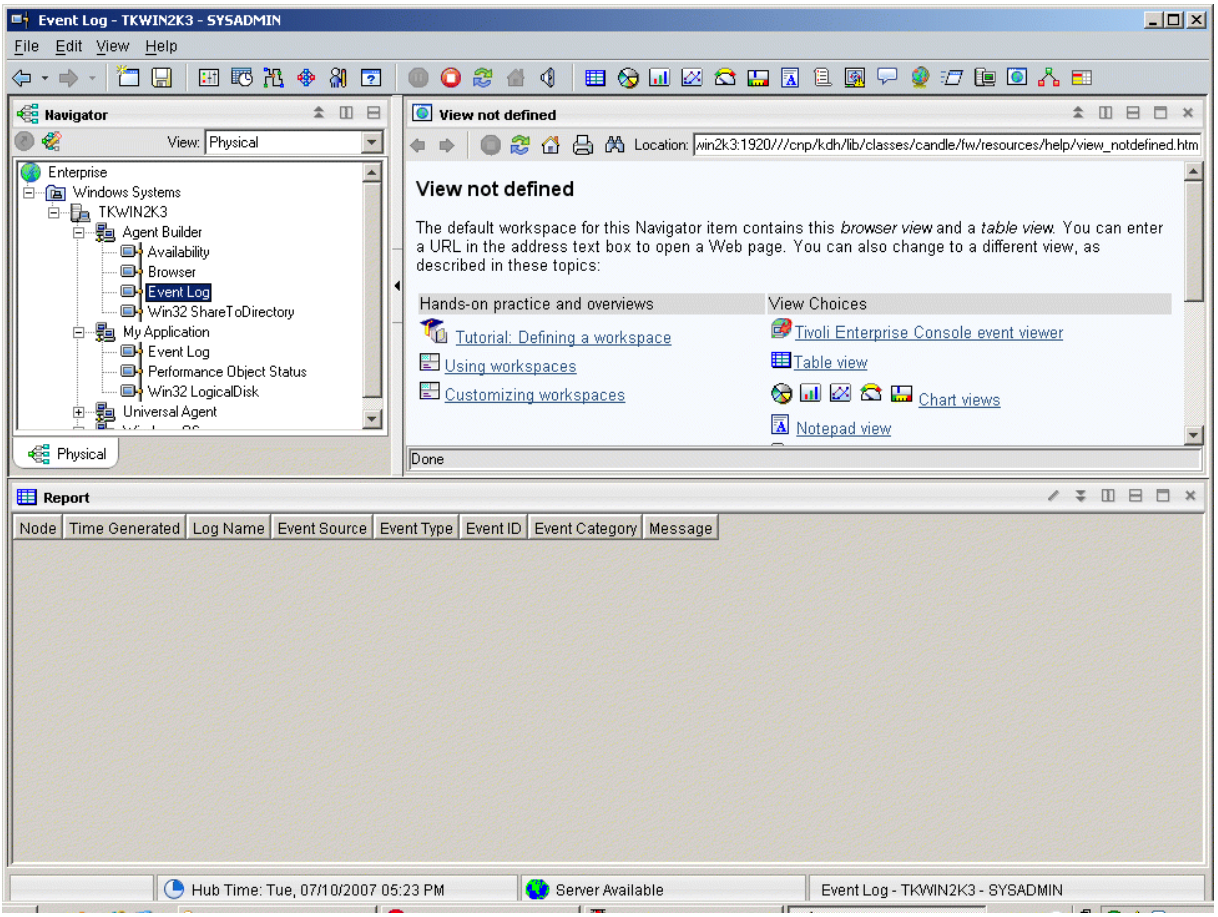


图 74. “事件日志”节点

请参阅第 1190 页的『属性参考』，以获取 Agent Builder 的属性组及属性的相关描述。

卸载代理程序

您可以从受监视的主机移除 Agent Builder 生成的代理程序。

关于此任务

卸载过程只会卸载代理程序系统中的代理程序。此过程不会卸载任何其他代理程序或任何监视基础结构。

在 IBM Tivoli Monitoring 环境中，您可以使用以下一个过程来移除 Agent Builder 生成的代理程序：

- 第 1177 页的『使用 Tivoli Enterprise Portal 移除 Tivoli Monitoring 代理程序』
- 第 1178 页的『在不使用 Tivoli Enterprise Portal 的情况下移除 Tivoli Monitoring 代理程序』

在使用上述任意过程移除代理程序后，使用以下过程从 Tivoli Enterprise Portal 中清除该代理程序：第 1178 页的『从 Tivoli Enterprise Portal 清除 Tivoli Monitoring 代理程序』。

在 IBM Cloud Application Performance Management 环境中，使用以下过程：第 1178 页的『卸载 IBM Cloud Application Performance Management 代理程序』。

使用 Tivoli Enterprise Portal 移除 Tivoli Monitoring 代理程序

在 IBM Tivoli Monitoring 环境中，您可以使用 Tivoli Enterprise Portal 移除代理程序。

开始之前

操作系统代理程序必须正在运行，才能移除所创建的代理程序。

过程

要使用 Tivoli Enterprise Portal 移除代理程序，请完成下列步骤：

- 在 Tivoli Enterprise Portal 导航树中，右键单击代理程序，然后选择**移除**。

在不使用 Tivoli Enterprise Portal 的情况下移除 Tivoli Monitoring 代理程序
如果 Tivoli Enterprise Portal 在 IBM Tivoli Monitoring 环境中不可用，那么可以使用操作系统脚本和命令来移除代理程序。

过程

要在不使用 Tivoli Enterprise Portal 的情况下从目标系统移除 Agent Builder 生成的代理程序，可以完成以下任意步骤：

- Windows
  在 Windows 系统上，使用以下命令：

  ```
  cd ITM_INSTALL/TMAITM6
  kxx_uninstall.vbs ITM_INSTALL
  ```

  其中 *xx* 是代理程序的产品代码
- Windows
  或者， 在 Windows 系统上，可以使用 `cscript.exe` 命令来运行卸载脚本。此命令是 vbs 脚本的命令行界面解析器，不显示窗口；消息将显示在控制台上：

  ```
  cd ITM_INSTALL/TMAITM6
  cscript.exe kxx_uninstall.vbs ITM_INSTALL
  ```

- Linux AIX
  在 Linux 或 UNIX 系统上，使用在 *ITM_INSTALL*`/bin` 中找到的 `uninstall.sh` 文件：

  ```
  uninstall.sh [-f] [-i] [-h ITM_INSTALL] [product platformCode]
  ```

从 Tivoli Enterprise Portal 清除 Tivoli Monitoring 代理程序
在 IBM Tivoli Monitoring 环境中，移除代理程序后，来自代理程序的信息的空字段可能保留在 Tivoli Enterprise Portal 中。要移除这些字段，请从 TivoliEnterprise Portal 清除该代理程序。

过程

1. 确保 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 已启动并正在运行。
2. 登录到 Tivoli Enterprise Portal 客户机。
3. 从 Tivoli Enterprise Portal 客户机的“物理导航器”视图中，右键单击**企业**，然后选择**工作空间 > 受管系统状态**。
   此时将显示“受管系统状态”工作空间。
4. 选择代理程序的所有 IBM Tivoli Managed System。
5. 单击右键并选择**清除脱机条目**，这样会从表中清除所有上述条目。

卸载 IBM Cloud Application Performance Management 代理程序
您可以从 IBM Cloud Application Performance Management 环境中的任何受监视系统卸载自己的代理程序。

过程

1. 在安装了代理程序的系统上，启动命令行并切换到 *install_dir*`/bin` 目录，其中 *install_dir* 是监视代理程序的安装目录。
2. 要卸载特定监视代理程序，请输入代理程序脚本名称和卸载选项，其中 *name* 是代理程序脚本名称：
   - 在 Windows 系统上：*name*`-agent.bat uninstall`
   - 在 Linux 或 AIX 系统上：`./`*name*`-agent.sh uninstall`

导入应用程序支持文件

如果代理程序要在 IBM Tivoli Monitoring 环境中使用，那么可以在安装程序包中包含定制情境、工作空间、“执行操作”命令和查询。

关于此任务

为了使情境、工作空间和代理程序使用一个安装映像，情境文件和工作空间文件必须与代理程序位于同一个项目中。Agent Builder 提供了在代理程序项目中创建适当文件的向导。

还可以在安装程序包中包含与代理程序关联的定义。这些定义的内容对于企业监视环境和系统监视器环境中使用的代理程序是不同的。企业监视代理程序映像可以包含定制情境、工作空间、“执行操作”命令和查询。系统监视器代理程序映像可以包含专用情境、陷阱定义和代理程序配置信息。

为将适合的定义和代理程序本身包含在一个安装程序包中，这些文件必须与代理程序位于同一个项目中。Agent Builder 提供了为企业监视安装创建适当文件的向导。可以使用 *IBM Tivoli Monitoring Administrator's Guide* 中的 *Agent Autonomy* 一章所描述的过程来创建系统监视器代理程序环境的文件。生成的文件将复制到该代理程序的 Eclipse 项目的根目录。

导出和导入 Tivoli Enterprise Monitoring Agent 的文件

关于此任务

在 Tivoli Enterprise Portal 中创建情境、工作空间、查询和“执行操作”命令之后，可以将它们一起导出并导入到 Tivoli Monitoring V6.2 环境中。有关创建情境和工作空间的更多信息，请参阅第 1149 页的『创建工作空间、“执行操作”命令和情境』。使用以下步骤可抽取情境、工作空间、“执行操作”命令和查询:

过程

1. 从 **Project Explorer** 选项卡，右键单击代理程序项目文件夹。
2. 选择 **IBM Corporation** > **导入应用程序支持文件**。
3. 输入 Tivoli Enterprise Portal Server 的主机名。
4. 输入要连接到的 Tivoli Monitoring 环境的用户名和密码，然后单击**完成**。
5. 如果为代理程序定义了情境，将呈现一个对话框，其中列出了为代理程序定义的情境。
6. 从列表中选择要导出的情境，单击 **<<** 以添加到“选定的情境”表，然后单击**确定**。

   导入过程可能需要一段时间。任务完成时，将在代理程序项目中的相应文件夹下显示 SQL 文件。
7. 如果已为代理程序定义了“执行操作”命令，那么将显示一个对话框，其中呈现已定义的“执行操作”命令。从列表中选择要导出的“执行操作”命令，单击 **>>** 以添加到“选择的执行操作”表，然后单击**确定**。

   导入过程可能需要一段时间。任务完成时，将在代理程序项目中的相应文件夹下显示 SQL 文件。
8. 如果已为代理程序定义了定制查询，那么将显示一个对话框，其中呈现已定义的“查询”。从列表中选择要导出的查询，单击 **<<** 以添加到“选定的查询”表，然后单击**确定**。

   导入过程可能需要一段时间。任务完成时，将在代理程序项目中的相应文件夹下显示 SQL 文件。工作空间将自动导入。

下一步做什么

重新创建定制代理程序，在受监视的主机上安装代理程序，并安装 Tivoli Enterprise Portal 支持。

导出和导入 Tivoli System Monitor Agent 的文件

关于此任务

系统监视器代理程序定义包含在以下三种类型的文件中:

· 专用情境在名为 *xx*`_situations.xml` 的文件中定义，其中 *xx* 是两个字符的产品代码

- 陷阱配置信息在名为 *xx*_trapcnfg.xml 的文件中定义，其中 *xx* 是两个字符的产品代码
- 对于需要配置的代理程序，将为代理程序的每个实例在一个文件中定义配置。如果代理程序是单实例代理程序，那么该文件名为 *xx*.cfg。如果代理程序是多实例代理程序，那么每个实例都有一个对应文件。文件名为 *xx_instance name*.cfg，其中 *xx* 是两个字符的产品代码，*instance name* 是代理程序实例的名称。

过程

- 通过使用 *IBM Tivoli Monitoring Administrator's Guide* 中的 *Agent Autonomy* 一章所描述的过程来创建文件。手动将这些文件复制到项目目录的根目录，或者使用 Eclipse 导入功能来选择要导入的文件：**文件** > **导入** > **常规** > **文件系统**。

  这些文件包含在代理程序映像中并由安装程序安装。

  安装代理程序时，安装过程会：

  - 将包含的文件复制到相应的位置。
  - 在代理程序上运行的 pc_situations.xml 文件中定义的任何专用情境。
  - pc_trapcnfg.xml 中定义的陷阱定义用于根据情境转发陷阱。
  - 在以下情况下，将自动配置和启动代理程序：
    - 代理程序是单实例代理程序，并且没有定义为代理程序一部分的配置。
    - 代理程序是单实例代理程序，具有定义为代理程序一部分的配置，且映像包含 pc.cfg 文件。
    - 代理程序是多实例代理程序（所有多实例代理程序都需要配置）：安装程序将为每个 pc_*inst*.cfg 文件启动一个代理程序实例。

事件过滤和摘要

属性组被定义为纯事件或采样属性组。纯事件属性组包含异步出现的数据行。每个新数据行到达时，Tivoli Monitoring 都会立即对其进行处理。采样属性组会在每次请求数据时收集最新的一组数据行。以下属性组说明了它们之间的不同：

- 创建了 SNMPEvent 属性组以表示发送至代理程序的所有 SNMP 陷阱和通知。陷阱或通知在由受监视系统发送时异步到达。每个事件到达时，都将传递到 Tivoli Monitoring。
- 创建了 Disk 属性组以表示系统上所有磁盘的相关信息。磁盘信息是定期收集的。每次收集磁盘信息时，代理程序都会返回若干数据行（每个磁盘对应一行）。

纯事件属性组和采样属性组之间的不同会影响 Tivoli Monitoring 的各个方面。这些方面包括：情境、仓库数据和 Tivoli Enterprise Portal 视图。

每个情境指定（或分发）给一个或多个受管系统，以监视一组条件的特定情况。如果必须根据特定时间间隔的观察结果来确定事件，那么此类事件称为采样事件。如果事件是自然发生的，那么此类事件称为纯事件。因此，采样事件的情境具有一个关联的时间间隔，而纯事件情境没有。采样事件的另一个特征是导致事件的条件可以更改，从而使其不再为 true。纯事件无法更改。因此，针对采样事件发起的警报可以从 true 更改为 false，而纯事件在发生时保持为 true。

采样事件的示例是进程数 > 100。当进程数超过 100 时事件变为 true，而稍后当此计数回落至小于等于 100 时将再次变为 false。监视用户的无效登录尝试的情境是一个纯事件；检测到无效登录尝试时该事件发生，且不会变为 False 事件。虽然可创建根据采样属性组的特定时间间隔进行评估的情境，但此类评估不可用于纯事件属性组。

类似地，对于历史数据，可以配置收集采样数据的频率。但是，当开启对纯事件数据的收集时，将在每次收集时获取一行。

Tivoli Enterprise Portal 中针对采样数据显示的数据是最新收集的一组行。针对纯事件属性组显示的数据是代理程序维护的本地高速缓存的内容。它不一定与传递至 Tivoli Monitoring 以进行情境评估和历史数据收集的数据相匹配。

控制重复事件

使用事件过滤和摘要选项来控制如何将重复事件发送至 Tivoli Monitoring。

开始之前

有关事件过滤和摘要的更多信息，请参阅第 1180 页的『事件过滤和摘要』。

关于此任务

Agent Builder 定义了属性组，这些属性组在 Tivoli Monitoring 中将事件数据表示为纯事件。这些属性组包括日志文件、AIX 二进制日志、SNMP 事件和 JMX 通知。这些属性组可产生多个重复事件。您可以控制向 Tivoli Monitoring 发送这些重复事件的方式。在**高级**窗口中**高级数据源属性**下的**事件信息**选项卡中，您可以针对日志文件、SNMP 事件和 JMX 通知属性组激活这些控件。

是否将事件视为其他事件的重复事件取决于您在属性组中定义的键特性。当事件中所有键特性的值与现有事件中相同键特性的值匹配时，发生重复事件。启用事件过滤和摘要时，会自动添加 `isSummary`、`occurrenceCount`、`summaryInterval` 和 `eventThreshold` 函数的特性。

过程

- 在**事件过滤和摘要选项**区域中，选择下列其中一个选项：
  - **不进行事件过滤或摘要**：发送没有任何事件过滤或摘要的所有事件。此选项为缺省选项。
  - **过滤并摘要事件**：为具有重复事件的每个事件以及基于键特性的每个唯一事件创建摘要记录。选择此选项还会选择事件过滤选项。在**摘要选项**区域中，输入摘要时间间隔。您可以输入以秒为单位的值，或插入配置属性。

    事件过滤选项如下：

    - **仅发送摘要事件**：仅发送指定时间间隔的摘要记录。
    - **发送所有事件**：发送所有事件和摘要记录。
    - **发送第一个事件**：对于每个事件，仅发送在所指定摘要时间间隔收到的第一个事件，不发送重复事件。此选项还会发送摘要记录。
    - **事件阈值**：当时间间隔中收到的重复事件数可以被此阈值整除时，向 Tivoli Monitoring 发送事件。例如，如果将事件阈值设置为 5，并且在时间间隔中收到的重复事件（包括第一个事件）少于 5，那么不会向 Tivoli Monitoring 发送事件。如果收到 5、6、7、8 或 9 个重复事件，那么将发送一个事件。如果收到 10 个重复事件，那么发送 2 个事件。在**事件阈值**字段中，您可以输入数字或插入配置属性。此选项还会发送摘要记录。

在 Tivoli Enterprise Portal 中查看事件过滤和摘要

关于如何根据事件过滤和摘要选项处理数据的示例。

代理程序将保留最近接收到的事件的高速缓存。在缺省情况下，此高速缓存大小为 100。如果启用代理程序事件过滤和摘要，那么高速缓存中的事件数与发送到 IBM Tivoli Monitoring 的数量可能不同。高速缓存中的其他事件可能不会达到指定的发送阈值。或者，如果选择了**发送所有事件**选项，那么高速缓存中的事件数可能较少。如果设置**发送所有事件**选项，那么在每次发生重复时将发送事件。但是，仅在高速缓存中保留事件的一个副本，并且每次发生事件时出现次数都将递增。要查看发送至 IBM Tivoli Monitoring 的事件，请创建历史视图。有关创建历史视图的信息，请参阅《Tivoli Enterprise Portal 用户指南》中的『历史报告』。您可以将此视图与 Tivoli Enterprise Portal 中的实时高速缓存视图进行比较。还可以使用情境来进行此比较。

下列示例说明如何根据您对事件过滤和摘要进行的选择（如果有）处理相同的日志数据。创建的示例代理程序用于描述不同的行为。每个属性组都定义为监视同一个日志文件。在每个示例中都显示历史视图和实时（高速缓存）视图。Tivoli Enterprise Portal 中的节点名称反映了所选设置。在缺省情况下，历史视图将最新事件显示在最后。高速缓存的缺省实时视图将最新事件显示在最前。在这些示例中，历史视图显示过去 1 小时的事件。

当有新事件到达时，您会在高速缓存视图中看到这些事件。当有重复事件到达时，现有行内的数据将进行更新。经过摘要时间间隔后，现有事件将转换为摘要事件并发送。然后将为下一个摘要时间间隔添加新行。

（第 1182 页的图 75）显示没有启用事件过滤或摘要时的历史视图和高速缓存视图。这两个视图都显示相同数据，但顺序相反。为了显示对应事件，历史视图向下滚动，而实时（高速缓存）视图向上滚动。

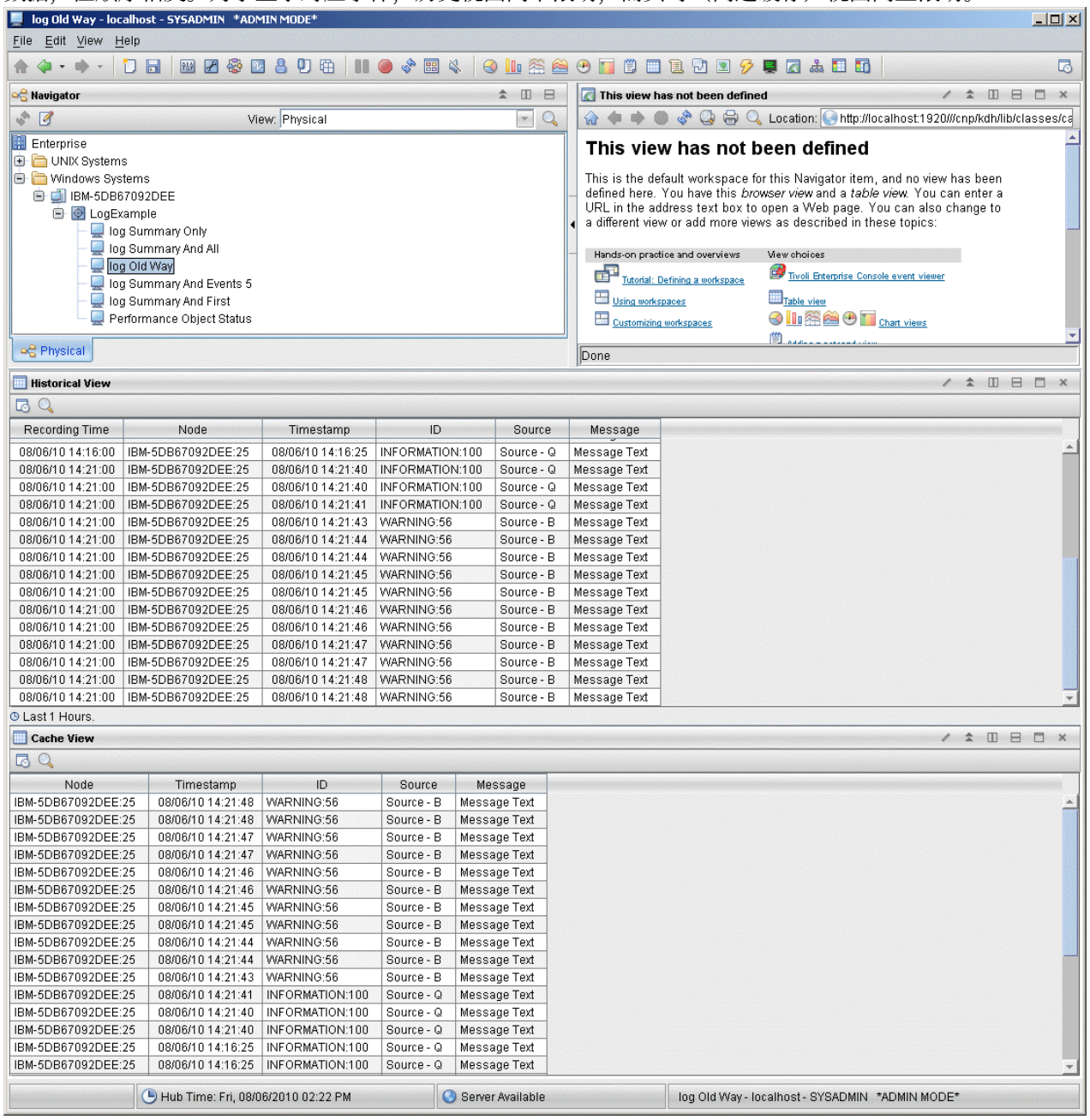


图 75. 未启用事件过滤或摘要时的历史视图和高速缓存视图

第 1183 页的图 76 显示在“**事件信息**”选项卡中选择了**仅发送摘要事件**选项时的历史视图和高速缓存视图。摘要事件将同时显示在这两个视图中，但新事件只显示在实时（高速缓存）视图中。

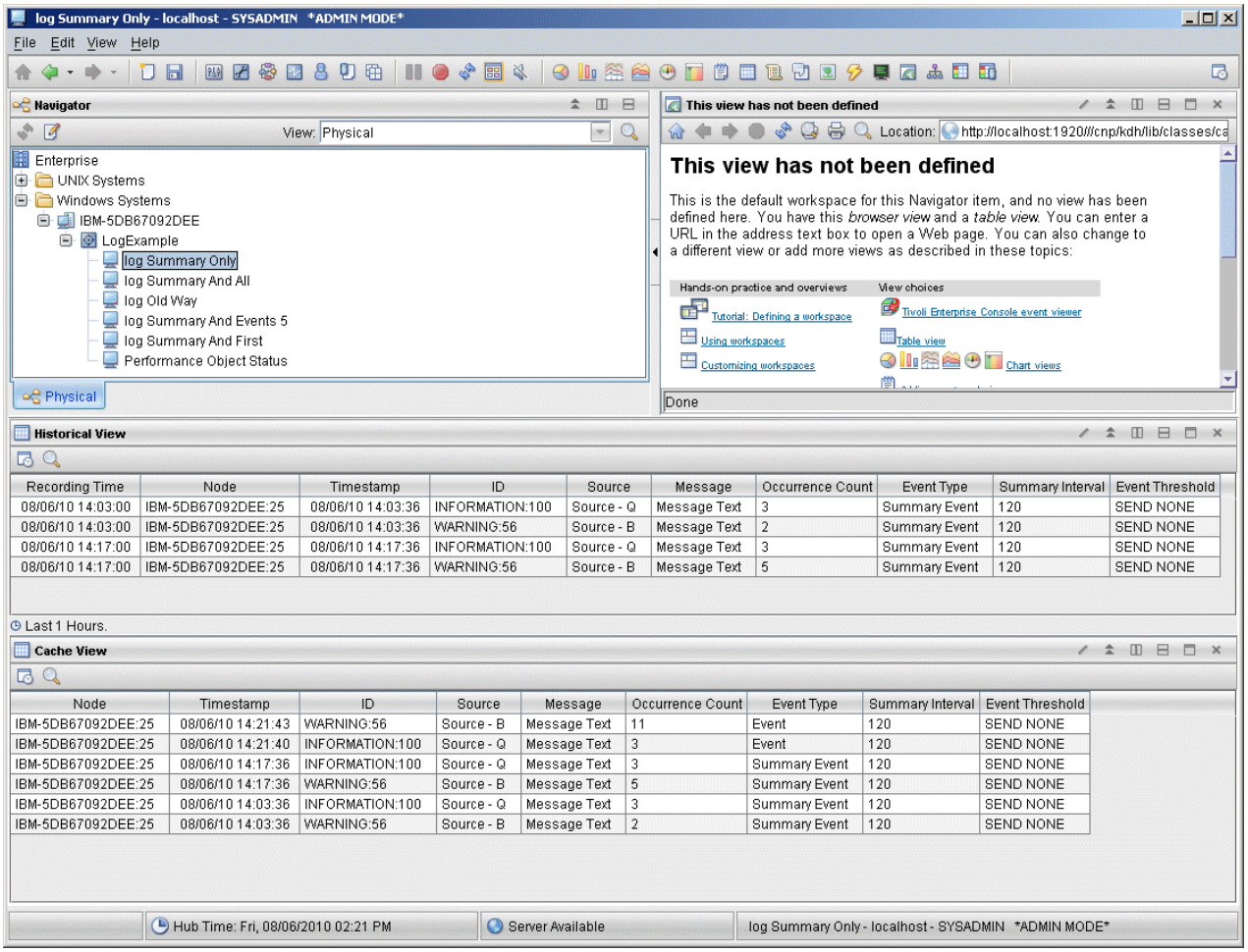


图 76. 选择了**仅发送摘要事件**时的历史视图和高速缓存视图

第 1184 页的图 77 显示在“**事件信息**”选项卡中选择了**发送所有事件**选项时的历史视图和高速缓存视图。两个视图都显示所有事件，但您还可以看到每个时间间隔结束时创建的摘要事件。实时视图随时间间隔流逝而变化。现有事件将转换为摘要记录，然后添加新事件。还添加了另外两个可用的事件属性，这两个属性用于显示摘要时间间隔（在本例中为 120 秒）和 *SEND ALL* 阈值。

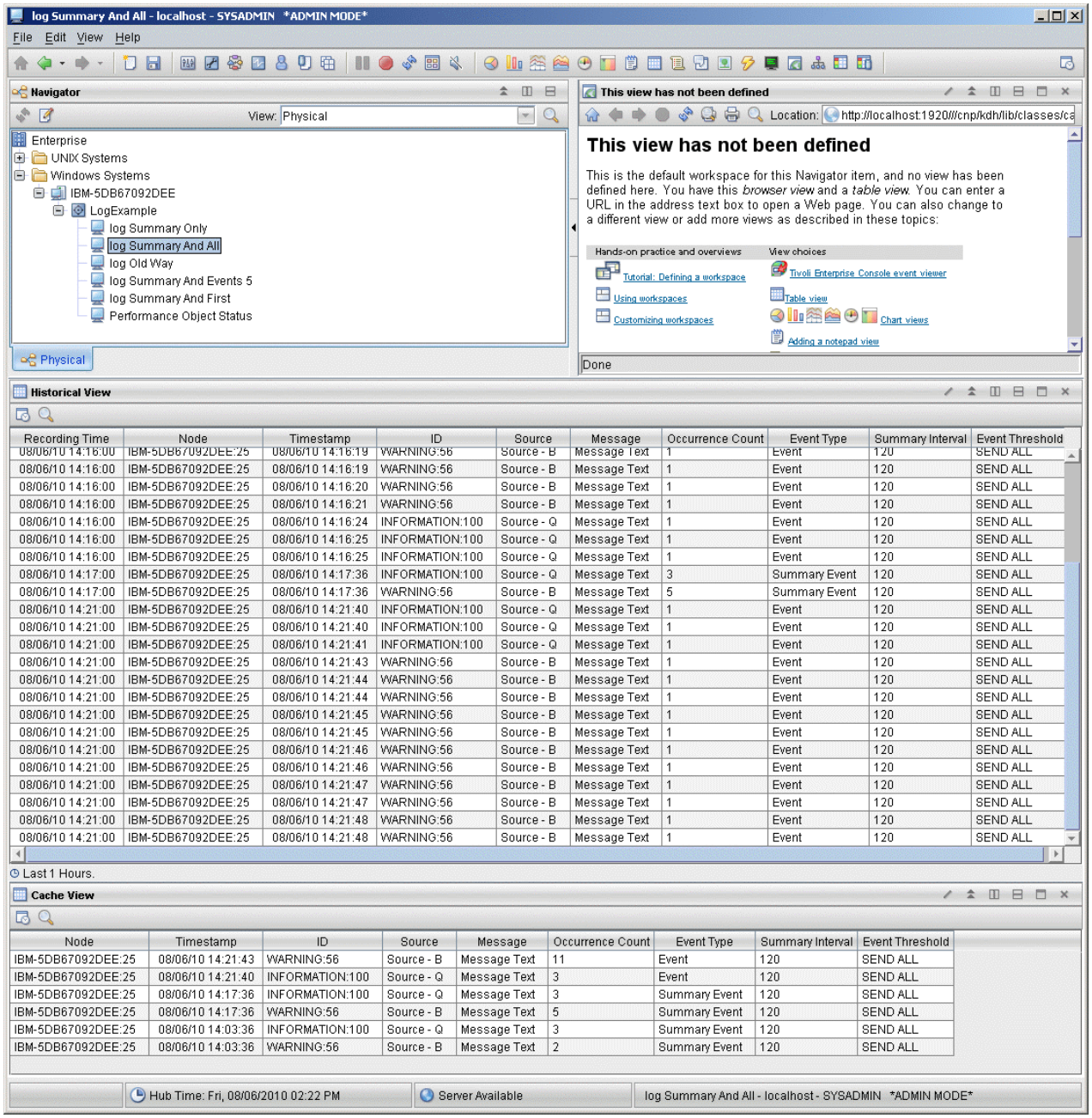


图 77. 选择了**发送所有事件**时的历史视图和高速缓存视图

第 1185 页的图 78 显示在“**事件信息**”选项卡中选择了**发送第一个事件**选项时的历史视图和高速缓存视图。摘要事件将同时显示在这两个视图中，但所有新事件只显示在实时（高速缓存）视图中。对于每个事件，历史视图仅显示在时间间隔内收到的第一个事件，不显示重复事件。

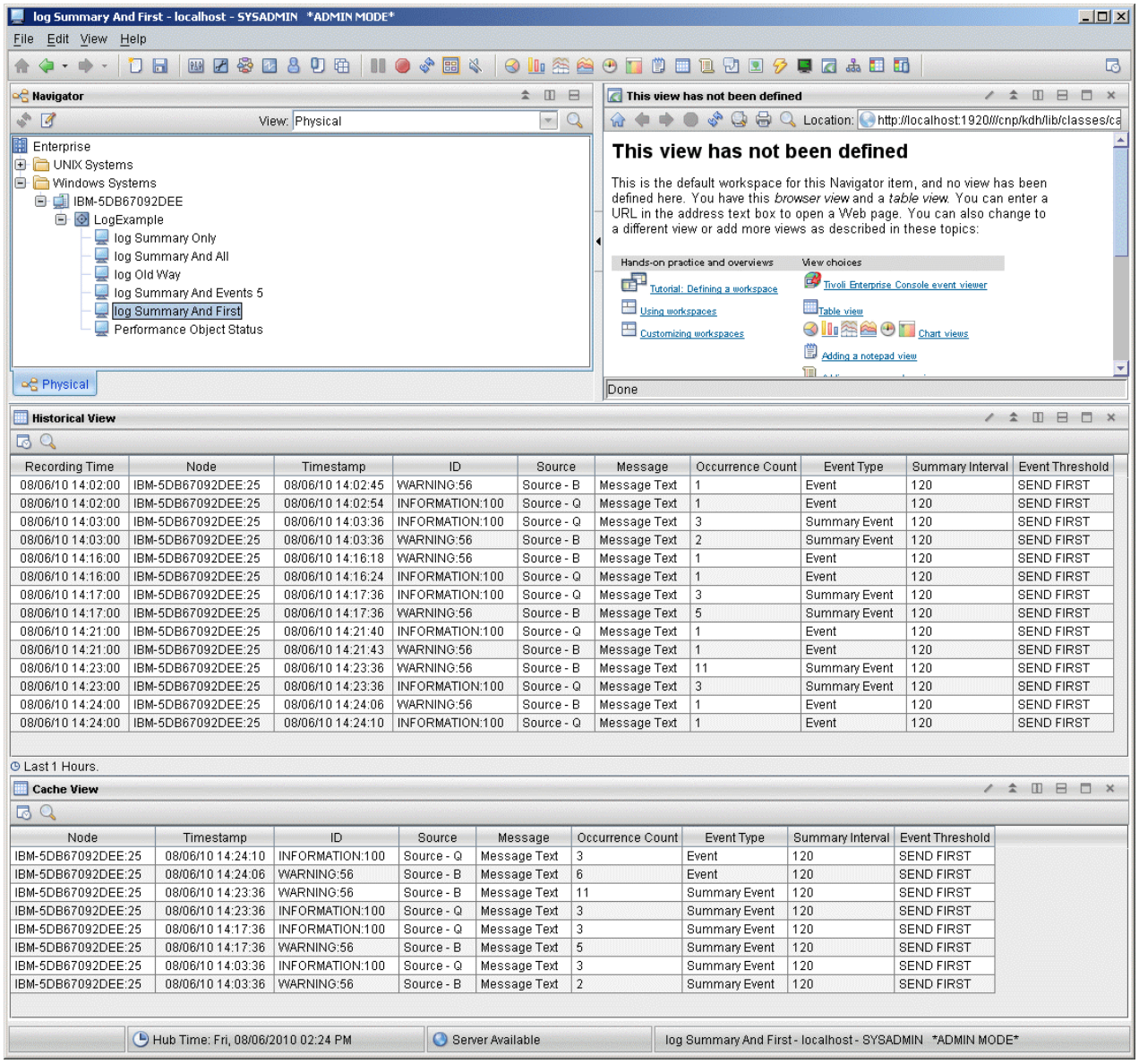


图 *78.* 选择了**发送第一个事件**时的历史视图和高速缓存视图

第 1186 页的图 79 显示选择了**事件阈值**选项并输入值 5 时的历史视图和高速缓存视图。摘要事件将同时显示在这两个视图中，但所有新事件只显示在实时（高速缓存）视图中。本示例中指定的阈值为 5。仅当在时间间隔内接收到一个事件的五个重复事件（包括第一个事件）时，历史视图才会显示事件。如果接收到的重复事件不足 5 个，那么不显示任何事件。如果在此时间间隔内收到 6、7、8 或 9 个重复事件，那么只显示一个事件。如果收到 10 个重复事件，那么显示 2 个事件。

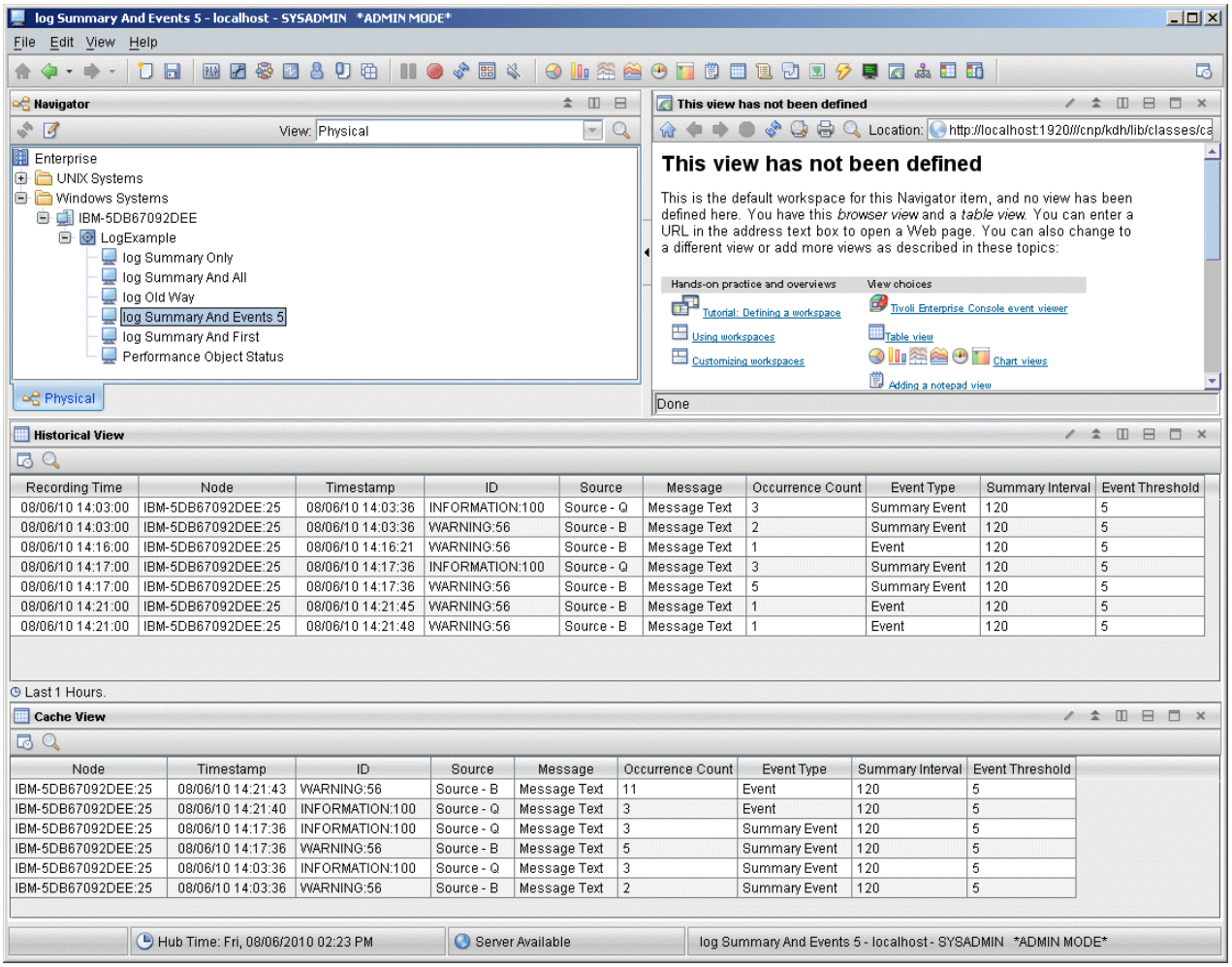


图 79. 选择了**事件阈值**时的历史视图和高速缓存视图

相关概念

第 1180 页的『事件过滤和摘要』

故障诊断和支持

请查看故障诊断信息，以了解您在安装、配置或使用 IBM Agent Builder 时可能遇到的问题。

有关在 IBM Cloud Application Performance Management 环境中开发、安装或使用定制代理程序时的问题故障诊断帮助，请参阅 developerWorks 上的 Cloud Application Performance Management 论坛。您可以搜索“agent_builder”标记，回复提出相关问题的条目，或使用您的问题创建新条目。

有关日志记录和消息参考信息以及与 IBM Tivoli Monitoring 环境有关的问题的故障诊断帮助，请参阅 IBM Agent Builder 故障诊断参考。

共享项目文件

与他人共享 IBM Tivoli Monitoring 代理程序项目。

过程

1. 获取文件。您需要与工作空间目录中的项目同名目录的全部内容。

例如，如果工作空间目录为 `c:\Documents and Settings\User1\workspace`，并且您想要共享名为 `TestProject` 的项目。必须使目录 `c:\Documents and Settings\User1\workspace\TestProject` 及其所有内容可供系统访问。

2. 选择**文件** > **导入**。
3. 打开 **IBM Tivoli Monitoring**。
4. 选择 **IBM Tivoli Monitoring Agent** 并单击**下一步**。
5. 输入代理程序 XML 文件的完整路径，或者单击**浏览**以浏览到该文件。
6. 单击**完成**。

结果

在向导完成时，您将会在工作空间中看到新的 IBM Tivoli Monitoring 代理程序项目。

共享解决方案安装程序项目

与他人共享解决方案安装程序项目

过程

1. 获取文件。您必须具有与工作空间目录中解决方案安装程序项目同名的目录的全部内容。
   例如，如果工作空间目录为 `c:\Documents and Settings\User1\workspace`，并且您想要共享名为 `TestProject Installer` 的解决方案安装程序项目。必须使目录 `c:\Documents and Settings\User1\workspace\TestProject Installer` 及其所有内容可供系统访问。
2. 单击**文件** > **导入**。
3. 打开**常规**。
4. 选择**现有项目到工作空间**，并单击**下一步**。
5. 输入解决方案安装程序项目根目录的完整路径，或者单击**浏览**以浏览到解决方案安装程序项目的根目录。（在此示例中为 `TestProject Installer` 目录。） 该目录中的项目显示在 Projects 列表中并且缺省情况下处于选中状态。
6. 可选： 单击**将项目复制到工作区中**。
7. 单击**完成**。

命令行选项

Agent Builder 命令行界面 (CLI) 中可用的命令。

Tivoli Monitoring Agent Builder 包含命令行界面 (CLI)，您可以使用该界面来生成 Tivoli Monitoring Agent，而无需启动 Eclipse 图形用户界面 (GUI)。可以作为构建的一部分而生成代理程序，例如：

在 Windows 系统上，可以使用下列目录中的批处理文件来访问 CLI：

```
install_location\agenttoolkit.bat
```

在 UNIX 及 Linux 系统上，可以使用下列目录中的脚本来访问 CLI：

```
install_location/agenttoolkit.sh
```

本文档中描述的命令针对 Windows 系统编排格式，在这些系统上，目录路径使用反斜杠 (\)。

对于 UNIX® 或 Linux® 系统，请使用与 Windows 系统相同的命令，但是进行下列更改：

· 在目录路径中，请使用正斜杠 (/)，而不要使用反斜杠 (\)。

· 请使用 `agenttoolkit.sh` 脚本，而不要使用 `agenttoolkit.bat` 脚本。

命令

第 1188 页的表 300 列出文本命令的每个命令选项的名称及用途说明：

表 *300.* 命令快速参照表	
命令	**用途**
generatelocal	装入并验证 `itm_toolkit_agent.xml` 文件，并生成用于运行 Tivoli Monitoring Agent 的文件。将会安装至本地 Tivoli Monitoring 环境。
generatemappingfile	创建映射文件，用于将定制 IBM Tivoli Monitoring V5.x 资源模型移植至 IBM Tivoli Monitoring V6 代理程序。
generatezip	生成名为 *productcode*`.zip` 或 *productcode*`.tgz` 的压缩文件。

从表中引用的命令描述说明了如何运行命令，其中包含下列信息：

用途
列示命令的用途。

格式
指定在命令行输入的语法。语法包含命令名称和命令的参数列表。在命令名称之后，跟着每个参数的定义。

示例
命令的示例包含示例的简短描述和语法示例。

用法
提供命令的说明及其用法。

注释
提供能够传达更多信息的命令或文本。

命令 - generatelocal

使用此命令可以装入并验证 XML，并生成用于运行 Tivoli Monitoring Agent 的文件。

用途

装入并验证 `itm_toolkit_agent.xml` 文件，并生成用于运行 Tivoli Monitoring Agent 的文件。将会安装至本地 Tivoli Monitoring 环境。

格式

对于 Windows 系统：

```
install_location\agenttoolkit.bat project_dir -generatelocal itm_install_dir
```

其中：

install_location
Agent Builder 的安装目录

project_dir
`itm_toolkit_agent.xml` 文件所在目录的名称

itm_install_dir
Tivoli Monitoring 的安装位置（例如 `c:\IBM\ITM`）

示例

在下列 Windows 示例中，将验证 `C:\ABCAgent` 中的代理程序定义，且会在 `C:\IBM\ITM` 中生成运行 `ABCAgent` 所需的文件：

```
install_location\agenttoolkit.bat C:\ABCAgent -generatelocal C:\IBM\ITM
```

命令 - generatemappingfile

使用此命令可以将定制 IBM Tivoli Monitoring V5.x 资源模型迁移至 IBM Tivoli Monitoring V6 代理程序。

用途

此命令会创建映射文件，用于将定制 IBM Tivoli Monitoring V5.x 资源模型迁移至 IBM Tivoli Monitoring V6 代理程序。

格式

对于 Windows 系统：

```
install_location\agenttoolkit.bat project_dir -generatemappingfile output_dir
   itm5_interp_list
```

其中：

install_location
Agent Builder 的安装目录

project_dir
`itm_toolkit_agent.xml` 所在目录的名称

output_dir
映射文件所写入的目录的名称

itm5_interp_list
运行定制资源模型的 ITM 5x 操作系统的逗号分隔列表。允许的值如下所示：

- `aix4-r1`
- `hpux10`
- `linux-ix86`
- `linux-ppc`
- `linux-s390`
- `os2-ix86`
- `os400`
- `solaris2`
- `solaris2-ix86`
- `w32-ix86`

示例

对于 Windows 系统

```
install_location\agenttoolkit.bat c:\ABCAgent -generatemappingfile c:\output
   linux-ix86,linux-ppc,linux-s390
```

命令 - generatezip

使用此命令可以装入并验证 XML 以及生成可用于在其他系统上安装代理程序的压缩文件。

用途

装入并验证 `itm_toolkit_agent.xml` 文件以及生成名为 *productcode*`.zip` 或 *productcode*`.tgz` 的压缩文件。所生成的压缩文件可用于在其他系统上安装代理程序。根据您的环境，可能会同时生成这两种类型的文件。

格式

对于 Windows 系统：

```
install_location\agenttoolkit.bat project_dir -generatezip output_dir
```

其中：

project_dir
`itm_toolkit_agent.xml` 文件所在目录的名称

output_dir
压缩文件所写入的目录的名称

示例

在下列 Windows 示例中，将验证 `C:\ABCAgent` 中的代理程序定义，且会在 `C:\Output` 中生成压缩文件，其中包含运行 `ABCAgent` 所需的文件：

```
install_location\agenttoolkit.bat\ C:\ABCAgent -generatezip C:\Output
```

属性参考

提供 Agent Builder 中每一个生成的属性组的属性描述。

可用性节点

“可用性”属性组包含应用程序的可用性数据。

此表提供了用于表示应用程序可用性的通用格式，包括应用程序以下三个方面的相关信息：服务（仅限 Windows）、进程和命令返回码。

以下列表包含关于“可用性”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“应用程序组件”属性 - 此属性是键属性

描述
应用程序某个组成部分的描述性名称

类型
字符串

名称

属性名称
Application_Component

列名
COMPONENT

“名称”属性

描述
进程、服务或功能测试的名称。此名称与进程可执行文件名、服务短名称或用于测试应用程序的进程的名称相匹配。

类型
字符串

名称

属性名称
Name

列名
NAME

“状态”属性

描述
应用程序组件的状态。

- 对于进程，值为 UP、DOWN、WARNING 或 PROCESS_DATA_NOT_AVAILABLE。对于一个进程，如果匹配的进程正在运行，但无法收集该进程的资源使用信息，那么将对该进程显示 PROCESS_DATA_NOT_AVAILABLE。
- 对于服务，值为 UP、DOWN 或 UNKNOWN。如果未安装该服务，那么将显示 UNKNOWN。
- 对于命令返回码，值为 PASSED 或 FAILED。

类型
字符串

名称

属性名称
Status

列名
STATUS

“全名”属性

描述
进程的全名，其中包含依赖于进程的信息。此名称可能包含完整路径（如果进程是以这种方式启动）。此名称也可以包含不完整的路径，甚至可以包含进程更改过的路径。

类型
字符串

名称

属性名称
Full_Name

列名
FULLNAME

“类型”属性

描述
指示应用程序组件的类型。组件是进程、服务或命令返回码。

类型
整数（标尺）

名称

属性名称
Type

列名
TYPE

“虚拟大小”属性

描述
进程的虚拟大小 (MB)

类型
整数（标尺）

名称

属性名称
Virtual_Size

列名
VIRTSIZE

“每秒的缺页故障数”属性

描述
进程的缺页故障率，以每秒故障数测量。此值仅包含对进程有效的数据。

类型
整数（标尺）

名称

属性名称
Page_Faults_Per_Sec

列名
PAGEFAULTS

“工作集大小”属性

描述
进程的工作集大小 (MB)。此值仅包含对进程有效的数据。

类型
整数（标尺）

名称

属性名称
Working_Set_Size

列名
WORKSET

“线程计数”属性

描述
此进程当前分配的线程数。此值仅包含对进程有效的数据。

类型
整数（标尺）

名称

属性名称
Thread_Count

列名
THREADS

PID 属性

描述
与进程相关联的进程标识。此值仅包含对进程有效的数据。

类型
整数（标尺）

名称

属性名称
PID

列名
PID

“特权时间百分比”属性

描述
此进程用于进行特权操作的可用处理器时间百分比

类型
整数（标尺）

名称

属性名称
Percent_Privileged_Time

列名
PERCPRIV

“用户方式时间百分比”属性

描述
此进程用于进行用户方式操作的可用处理器时间百分比

类型
整数（标尺）

名称

属性名称
Percent_User_Mode_Time

列名
PERCUSER

“处理器时间百分比”属性

描述
此进程使用处理器来运行指令所耗用的时间百分比

类型
整数（标尺）

名称

属性名称
Percent_Processor_Time

列名
PERCPROC

“命令行”属性

描述
启动进程时在命令行中指定的程序名和任何自变量。在运行服务或功能测试时，此属性的值为 *N/A*。

类型
字符串

名称

属性名称
Command_Line

列名
CMDLINE

“功能测试状态”属性

描述
功能测试的返回码。受监视的应用程序正常运行时，将返回 SUCCESS。当该应用程序未正常运行时，将返回 NOT_RUNNING。当该行并非表示功能测试时，将返回 N/A。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串，仓库和查询将返回数字。已定义的值为：N/A(1)、SUCCESS (0)、GENERAL_ERROR (2)、WARNING (3)、NOT_RUNNING (4)、DEPENDENT_NOT_RUNNING (5)、ALREADY_RUNNING (6)、PREREQ_NOT_RUNNING (7)、TIMED_OUT (8)、DOESNT_EXIST (9)、UNKNOWN (10)、DEPENDENT_STILL_RUNNING (11) 或 INSUFFICIENT_USER_AUTHORITY (12)。任何其他值在 Tivoli Enterprise Portal 中都显示数字值。

名称

属性名称
Functionality_Test_Status

列名
FUNCSTATUS

“功能测试消息”属性

描述
与“功能测试状态”对应的文本消息。此属性只对命令返回码有效。

类型
字符串

名称

属性名称
Functionality_Test_Message

列名
FUNCMSG

Performance Object Status 节点

使用 Performance Object Status 属性组可以查看组成此代理程序的所有属性组的状态。每个属性组都由此表中的一行或者其他类型视图表示。属性组的状态反映该属性组的最后一次数据收集尝试或最后一个数据接收事件的结果。通过检查状态信息，可以确定该代理程序是否正常运行。如果您的代理程序不收集数据，而是接收数据（事件数据），那么与采样数据相关的属性不会包含有用的数据。只有所列出的前七个属性与事件数据相关。

“历史”组

此属性组 适合与 Tivoli Data Warehouse 配合使用。

属性描述

以下列表包含关于 Performance Object Status 属性组中每个属性的信息：

“节点”属性：此属性是键属性。

描述

代理程序的受管系统名称。

类型

字符串

仓库名称

NODE

“时间戳记”属性

描述

收集数据时代理程序的本地时间。

类型

字符串

仓库名称

TIMESTAMP

“查询名称”属性：此属性是键属性。

描述

属性组的名称。

类型

字符串

仓库名称

QUERY_NAME 或 ATTRGRP

“对象名称”属性

描述

性能对象的名称。

类型

字符串

仓库名称

OBJECT_NAME 或 OBJNAME

“对象类型”属性

描述

性能对象的类型。

类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值：

· WMI (0)
· PERFMON (1)
· WMI ASSOCIATION GROUP (2)
· JMX (3)
· SNMP (4)
· SHELL COMMAND (5)
· JOINED GROUPS (6)
· CIMOM (7)
· CUSTOM (8)
· ROLLUP DATA (9)
· WMI REMOTE DATA (10)
· LOG FILE (11)
· JDBC (12)
· CONFIG DISCOVERY (13)
· NT EVENT LOG (14)
· FILTER (15)
· SNMP EVENT (16)
· PING (17)
· DIRECTOR DATA (18)
· DIRECTOR EVENT (19)
· SSH REMOTE SHELL COMMAND (20)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
OBJECT_TYPE 或 OBJTYPE

“对象状态”属性

描述
性能对象的状态。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值：

· ACTIVE (0)
· INACTIVE (1)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
OBJECT_STATUS 或 OBJSTTS

“错误代码”属性

描述
与查询关联的错误代码。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值：

· NO ERROR (0)
· GENERAL ERROR (1)

· OBJECT NOT FOUND (2)
· COUNTER NOT FOUND (3)
· NAMESPACE ERROR (4)
· OBJECT CURRENTLY UNAVAILABLE (5)
· COM LIBRARY INIT FAILURE (6)
· SECURITY INIT FAILURE (7)
· PROXY SECURITY FAILURE (9)
· NO INSTANCES RETURNED (10)
· ASSOCIATOR QUERY FAILED (11)
· REFERENCE QUERY FAILED (12)
· NO RESPONSE RECEIVED (13)
· CANNOT FIND JOINED QUERY (14)
· CANNOT FIND JOIN ATTRIBUTE IN QUERY 1 RESULTS (15)
· CANNOT FIND JOIN ATTRIBUTE IN QUERY 2 RESULTS (16)
· QUERY 1 NOT A SINGLETON (17)
· QUERY 2 NOT A SINGLETON (18)
· NO INSTANCES RETURNED IN QUERY 1 (19)
· NO INSTANCES RETURNED IN QUERY 2 (20)
· CANNOT FIND ROLLUP QUERY (21)
· CANNOT FIND ROLLUP ATTRIBUTE (22)
· FILE OFFLINE (23)
· NO HOSTNAME (24)
· MISSING LIBRARY (25)
· ATTRIBUTE COUNT MISMATCH (26)
· ATTRIBUTE NAME MISMATCH (27)
· COMMON DATA PROVIDER NOT STARTED (28)
· CALLBACK REGISTRATION ERROR (29)
· MDL LOAD ERROR (30)
· AUTHENTICATION FAILED (31)
· CANNOT RESOLVE HOST NAME (32)
· SUBNODE UNAVAILABLE (33)
· SUBNODE NOT FOUND IN CONFIG (34)
· ATTRIBUTE ERROR (35)
· CLASSPATH ERROR (36)
· CONNECTION FAILURE (37)
· FILTER SYNTAX ERROR (38)
· FILE NAME MISSING (39)
· SQL QUERY ERROR (40)
· SQL FILTER QUERY ERROR (41)
· SQL DB QUERY ERROR (42)
· SQL DB FILTER QUERY ERROR (43)
· PORT OPEN FAILED (44)
· ACCESS DENIED (45)

· TIMEOUT (46)
· NOT IMPLEMENTED (47)
· REQUESTED A BAD VALUE (48)
· RESPONSE TOO BIG (49)
· GENERAL RESPONSE ERROR (50)
· SCRIPT NONZERO RETURN (51)
· SCRIPT NOT FOUND (52)
· SCRIPT LAUNCH ERROR (53)
· CONF FILE DOES NOT EXIST (54)
· CONF FILE ACCESS DENIED (55)
· INVALID CONF FILE (56)
· EIF INITIALIZATION FAILED (57)
· CANNOT OPEN FORMAT FILE (58)
· FORMAT FILE SYNTAX ERROR (59)
· REMOTE HOST UNAVAILABLE (60)
· EVENT LOG DOES NOT EXIST (61)
· PING FILE DOES NOT EXIST (62)
· NO PING DEVICE FILES (63)
· PING DEVICE LIST FILE MISSING (64)
· SNMP MISSING PASSWORD (65)
· DISABLED (66)
· URLS FILE NOT FOUND (67)
· XML PARSE ERROR (68)
· NOT INITIALIZED (69)
· ICMP SOCKETS FAILED (70)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
ERROR_CODE 或 ERRCODE

“最近一次收集开始时间”属性

描述
最近一次对该组启动数据收集的时间。

类型
具有枚举值的时间戳记。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值:

· NOT COLLECTED (0691231190000000)
· NOT COLLECTED (0000000000000001)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
LAST_COLLECTION_START 或 COLSTRT

“最近一次收集完成时间”属性

描述
最近一次对该组完成数据收集的时间。

类型
具有枚举值的时间戳记。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值：

· NOT COLLECTED (0691231190000000)

· NOT COLLECTED (0000000000000001)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
LAST_COLLECTION_FINISHED 或 COLFINI

“最近一次收集持续时间”属性

描述
最近一次对该组完成数据收集的持续时间（秒）。

类型
精确到小数点后 2 位的实数（32 位计数器）

仓库名称
LAST_COLLECTION_DURATION 或 COLDURA

“平均收集持续时间”属性

描述
该组的所有数据收集的平均持续时间（秒）。

类型
具有枚举值且精确到小数点后 2 位的实数（32 位计数器）。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值：

· NO DATA (-100)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

仓库名称
AVERAGE_COLLECTION_DURATION 或 COLAVGD

“刷新时间间隔”属性

描述
对该组进行刷新的时间间隔（秒）。

类型
整数（32 位计数器）

仓库名称
REFRESH_INTERVAL 或 REFRINT

“收集次数”属性

描述
自代理程序启动以来对该组进行收集的次数。

类型
整数（32 位计数器）

仓库名称
NUMBER_OF_COLLECTIONS 或 NUMCOLL

“高速缓存命中次数”属性

描述
通过高速缓存满足该组外部数据请求的次数。

类型
整数（32 位计数器）

仓库名称
CACHE_HITS 或 CACHEHT

“高速缓存未命中次数”属性

描述
未能通过高速缓存满足该组外部数据请求的次数。

类型
整数（32 位计数器）

仓库名称
CACHE_MISSES 或 CACHEMS

“高速缓存命中百分比”属性

描述
通过高速缓存满足该组外部数据请求的百分比。

类型
精确到小数点后 2 位的实数（32 位计数器）

仓库名称
CACHE_HIT_PERCENT 或 CACHPCT

“跳过的时间间隔数”属性

描述
由于应该开始下一次收集时上一次收集仍在运行而跳过后台数据收集的次数。

类型
整数（32 位计数器）

仓库名称
INTERVALS_SKIPPED 或 INTSKIP

Thread Pool Status 属性组

Thread Pool Status 属性组包含反映内部线程池（用于以异步方式收集数据）状态的信息。

以下是构成此属性组的属性列表。粗体名称显示了属性在 Tivoli Enterprise Portal 中的显示方式。

以下列表包含关于 Thread Pool Status 属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
当构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时，从代理程序系统收集到的时间。或者进行存储以记录历史数据。它表示代理程序系统的本地时区。

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“线程池大小”属性

描述
线程池中当前存在的线程数。

类型
整数

名称

属性名称
Thread_Pool_Size

列名
THPSIZE

“线程池最大大小”属性

描述
线程池中允许存在的最大线程数。

类型
整数

名称

属性名称
Thread_Pool_Max_Size

列名
TPMAXSZ

“线程池活动线程数”属性

描述
线程池中当前正在执行工作的线程数。

类型
整数

名称

属性名称
Thread_Pool_Active_Threads

列名
TPACTTH

“线程池平均活动线程数”属性

描述
线程池中同时在执行工作的平均线程数。

类型
整数

名称

属性名称
Thread_Pool_Avg_Active_Threads

列名
TPAVGAT

“线程池最小活动线程数”属性

描述
线程池中同时在执行工作的最小线程数。

类型
整数

名称

属性名称
Thread_Pool_Min_Active_Threads

列名
TPMINAT

“线程池最大活动线程数”属性

描述
线程池中同时在执行工作的最大线程数。

类型
整数

名称

属性名称
Thread_Pool_Max_Active_Threads

列名
TPMAXAT

“线程池队列长度”属性

描述
线程池队列中当前正在等待的作业数。

类型
整数

名称

属性名称
Thread_Pool_Queue_Length

列名
TPQLGTH

“线程池平均队列长度”属性

描述
此次运行期间线程池队列的平均长度。

类型
整数

名称

属性名称
Thread_Pool_Avg_Queue_Length

列名
TPAVGQL

“线程池最小队列长度”属性

描述
线程池队列达到的最小长度。

类型
整数

名称

属性名称
Thread_Pool_Min_Queue_Length

列名
TPMINQL

“线程池最大队列长度”属性

描述
线程池队列达到的最大长度。

类型
整数

名称

属性名称
Thread_Pool_Max_Queue_Length

列名
TPMAXQL

“线程池平均作业等待时间”属性

描述
作业等待线程池队列所花费的平均时间。

类型
整数

名称

属性名称
Thread_Pool_Avg_Job_Wait

列名
TPAVJBW

“线程池作业总数”属性

描述
自代理程序启动以来池中所有线程完成的作业数。

类型
整数

名称

属性名称
Thread_Pool_Total_Jobs

列名
TPTJOBS

“事件日志”属性节点

“事件日志”属性组包含所有与应用程序相关的最近事件日志条目。

缺省情况下，代理程序只显示在代理程序启动后发生的事件。事件发生 1 小时后，将从“事件日志”视图中移除。

以下列表包含关于“事件日志”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“日志名称”属性

描述
事件日志 - 应用程序日志、系统日志、安全日志或特定于应用程序的日志

类型
字符串

名称

属性名称
Log_Name

列名
LOGNAME

“事件源”属性

描述
应用程序所定义的事件源

类型
字符串

名称

属性名称
Event_Source

列名
EVTSOURCE

“事件类型”属性

描述
事件类型 - Error(0)、Warning(1)、Informational(2)、Audit_Success(3)、Audit_Failure(4) 和 Unknown(5)

类型
整数

名称

属性名称
Event_Type

列名
EVTTYPE

“事件标识”属性

描述
事件的标识

类型
整数

名称

属性名称
Event_ID

列名
EVTID

“事件类别”属性

描述
事件的类别

类型
字符串

名称

属性名称
Event_Category

列名
EVTCATEG

“消息”属性

描述
事件消息

类型
字符串

名称

属性名称
Message

列名
MESSAGE

“生成时间”属性

描述
生成事件的时间

类型
时间

名称

属性名称
Time_Generated

列名
TIMESTAMP

日志文件摘要

在数据源的高级属性中选择对应选项后，此属性组的特性将包含在摘要属性组中。

如果在数据源的高级属性中选择了**将特性包含在摘要属性组中**，那么将针对每个“日志文件”数据源创建一个“摘要”节点。摘要节点的名称为末尾添加 Summary 的数据源名称。

以下列表包含关于“日志文件摘要”属性组中每个缺省特性的信息。这些特性始终包含在摘要属性组中。如果选中**将特性包含在摘要属性组中**（请参阅第 1060 页的『监视日志文件』中的步骤 第 1064 页的『9』），那么该日志属性组的摘要属性组还将包含您选择的每个特性。值为日志文件属性组中对应特性的副本。

所有添加的特性共同组成一个键，每一组唯一键在摘要表中都有对应的一行。此行指示在时间间隔内接收到的日志记录数（在这些记录中，提供的所有键都与相应特性中报告的值匹配）。

“节点”特性 - 此特性是键特性

描述
代理程序的受管系统名称

类型
字符串

名称

特性名称
Node

列名
ORIGINNODE

“时间戳记”特性

描述
收集数据时代理程序的本地时间

类型
时间

名称

特性名称
Timestamp

列名
TIMESTAMP

“时间间隔单位”特性

描述
两次生成摘要特性之间间隔的秒数

类型
整数（标尺）

名称

特性名称
_Interval_Unit

列名
IU

“时间间隔”特性

描述
当前时间间隔在下一个更大时间单位内的偏移量（例如，在 1 小时中的分钟数）

类型
整数（标尺）

名称

特性名称
_Interval

列名
INV

“发生次数”特性

描述
在时间间隔内记录的发生次数

类型
整数（标尺）

名称

特性名称
_Occurrences

列名
OCC

“本地时间戳记”特性

描述
生成摘要数据的时间

类型
时间戳记

名称

特性名称
_LocalTimeStamp

列名
LTS

“日期时间”特性

描述
生成摘要数据的时间

类型
字符串

名称

特性名称
_Date_Time

列名
DT

“时间间隔单位名称”特性

描述
时间间隔单位的文字描述

类型
字符串

名称

特性名称
_Interval_Unit_Name

列名
IUN

“AIX 二进制日志”属性组

“AIX 二进制日志”属性组显示来自 AIX 二进制日志的事件，这些事件由提供的 `errpt` 命令字符串选择。

以下列表包含关于“AIX 二进制日志”属性组中每个属性的信息：

注: Agent Builder 不允许移除 `Identifier`、`ErrptTimestamp`、`Type`、`Class`、`ResourceName` 和 `Description` 属性，以及对它们重新排序或更改大小。代理程序将根据文本行中的列解析 `errpt` 命令所返回的数据。这些列由 `Identifier`、`ErrptTimestamp`、`Type`、`Class`、`ResourceName` 及 `Description` 特性的顺序及大小定义。移除这些特性、将它们重新排序或更改它们的大小将更改各列所对应的特性。这将导致 Tivoli Monitoring 中显示的结果行不正确。

但是，您可将这些特性重命名。

“节点”特性 - 此特性是键特性

描述
代理程序的受管系统名称

类型
字符串

名称

特性名称
Node

列名
ORIGINNODE

Identifier 属性 - 此属性是键属性

描述
`errpt` 所报告的事件标识

类型
字符串

名称

特性名称
Identifier

列名
IDENTIFIER

ErrptTimestamp 特性

描述
`errpt` 所报告的事件记录时间。

注: 此特性在运行时处于隐藏状态。此特性包含原始值。从此特性派生的其他特性以更易使用的形式显示值。为此，此属性可以在 Agent Builder 中使用，而在 Tivoli Monitoring 环境中，运行时将缺省处于不可见状态。如果您希望此特性可见，请在 Agent Editor 中的“**数据源定义**”页面中选中此特性，然后选中**在 Tivoli Enterprise Portal 中显示特性**。

类型
字符串

名称

特性名称
ErrptTimestamp

列名
ERRPTTIMES

类型

描述
errpt 所报告的单字符事件类型，即，下列其中一项：I(NFO)、P(END/ERF/ERM)、T(EMP) 和 U(NKN)

类型
字符串

名称

特性名称
Type

列名
TYPE

“类”特性 - 此特性是键特性

描述
errpt 所报告的事件类，即，下列其中一项：Hardware、Software、Operator 和 Undertermined。这些是枚举值。与情境配合使用的原始值为 H、S、O 和 U。

类型
字符串

名称

特性名称
Class

列名
CLASS

ResourceName

描述
errpt 所报告的资源名称，用于标识错误记录的来源

类型
字符串

名称

特性名称
ResourceName

列名
RESOURCENA

“描述”特性

描述
errpt 所报告的描述，这通常是描述错误性质的简短文本消息

类型
字符串

名称

特性名称
描述

列名
DESCRIPTIO

LogFile 特性

描述
二进制 `errpt` 日志的全名（包括路径）。

注: 此特性在运行时处于隐藏状态。此特性包含原始值。从此特性派生的其他特性以更易使用的形式显示值。为此，此属性可以在 Agent Builder 中使用，而在 Tivoli Monitoring 环境中，运行时将缺省处于不可见状态。如果您希望此特性可见，请在 Agent Editor 中的“**数据源定义**”页面中选中此特性，然后选中**在 Tivoli Enterprise Portal 中显示特性**。

类型
字符串

名称

特性名称
LogFile

列名
LOGFILE

“系统”特性

描述
在其中收集错误的系统的主机名

类型
字符串

名称

特性名称
System

列名
SYSTEM

LogName 特性

描述
从中收集记录的二进制 `errpt` 日志的基本名称

类型
字符串

名称

特性名称
LogName

列名
LOGNAME

LogPath 特性

描述
从中收集记录的二进制 `errpt` 日志所在目录的名称

类型
字符串

名称

特性名称
LogPath

列名
LOGPATH

EntryTime 特性

描述
errpt 所报告的事件记录时间（采用 Tivoli 时间戳记格式）。此时间不一定与代理程序接收事件的时间（记录在 **Timestamp** 字段中）完全相同。

类型
时间戳记

名称

特性名称
EntryTime

列名
ENTRYTIME

“监视器和通知”属性组

“监视器和通知”属性组的定义。

前 4 项特定于监视器，最后 1 项特定于通知（全部都与 JMX 相关）。

列出的每一项都带有指示，指出其是否基于事件。对于不基于事件的属性组，将在需要时收集数据。对于基于事件的属性组，代理程序将保留最近收到的 100 个事件的高速缓存。这些事件用于响应来自 Tivoli Enterprise Portal 的请求。这些事件将立即转发，以用于按情境分析和进行仓储。

计数器通知
“计数器通知”属性组是不基于事件的属性组，用于发送所有计数器监视器所接收的事件。

以下列表包含关于“计数器通知”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“通知类型”属性

描述
接收到的通知类型。描述 MBean 的受观测属性如何触发通知。

类型
字符串

名称

属性名称
Notification_Type

列名
NOTIFICATI

“监视器标识”属性

描述
生成此通知的监视器的监视器标识

类型
整数

名称

属性名称
Monitor_ID

列名
MONITOR_ID

“受观测 MBean”属性

描述
正在监视其属性的 MBean

类型
字符串

名称

属性名称
Observed_MBean

列名
OBSERVED_M

“受观测属性”属性

描述
“受观测 MBean”中监视的属性的名称

类型
字符串

名称

属性名称
Observed_Attribute

列名
OBSERVED_A

“阈值”属性

描述
监视器的当前阈值

类型
字符串

名称

属性名称
Threshold

列名
THRESHOLD

“偏移量”属性

描述
此属性每次超出阈值时，向阈值增加的值。此值构成新的阈值。

类型
字符串

名称

属性名称
Offset

列名
OFFSET

“模数”属性

描述
此属性的最大值。达到此值时，此属性将回滚并从 0 开始重新计数。

类型
整数

名称

属性名称
Modulus

列名
MODULUS

“计数器值”属性

描述
触发通知的计数器值

类型
整数

名称

属性名称
Counter_Value

列名
COUNTER_VA

“通知时间戳记”属性

描述
触发通知的时间

类型
时间

名称

属性名称
Notification_Time_Stamp

列名
NOTIFICAT0

“通知消息”属性

描述
通知中的消息

类型
字符串

名称

属性名称
Notification_Message

列名
NOTIFICAT1

标尺通知

“标尺通知”属性组是并非基于事件的属性组，用于发送所有标尺监视器所接收的事件。

以下列表包含关于“标尺通知”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“通知类型”属性

描述
接收到的通知类型。描述 MBean 的受观测属性如何触发通知。

类型
字符串

名称

属性名称
Notification_Type

列名
NOTIFICATI

“监视器标识”属性

描述
生成此通知的监视器的监视器标识

类型
整数

名称

属性名称
Monitor_ID

列名
MONITOR_ID

“受观测 MBean”属性

描述
正在监视其属性的 MBean

类型
字符串

名称

属性名称
Observed_MBean

列名
OBSERVED_M

“受观测属性”属性

描述
“受观测 MBean”中监视的属性的名称

类型
字符串

名称

属性名称
Observed_Attribute

列名
OBSERVED_A

“阈值下限”属性

描述
受监视器监视以待受观测属性跨越的阈值

类型
字符串

名称

属性名称
Low_Threshold

列名
LOW_THRESH

“阈值上限”属性

描述
受监视器监视以待受观测属性跨越的阈值

类型
字符串

名称

属性名称
High_Threshold

列名
HIGH_THRES

“标尺值”属性

描述
触发通知的标尺值

类型
字符串

名称

属性名称
Gauge_Value

列名
MODULUSGAUGE_VALU

“通知时间戳记”属性

描述
触发通知的时间

类型
时间

名称

属性名称
Notification_Time_Stamp

列名
NOTIFICAT0

“通知消息”属性

描述
通知中的消息

类型
字符串

名称

属性名称
Notification_Message

列名
NOTIFICAT1

已注册的监视器

“已注册的监视器”属性组是基于事件的属性组，它显示由代理程序创建的所有 JMX 监视器的列表。

以下列表包含关于“已注册的监视器”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
String

名称

特性名称
Node

列名称
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

特性名称
Timestamp

列名称
TIMESTAMP

“监视器标识”属性 - 此属性是键属性

描述
监视器的唯一整数标识

类型
整数

名称

特性名称
Monitor_ID

列名称
MONITOR_ID

“监视器参数”属性

描述
这是用于创建监视器的参数

类型
String

名称

特性名称
Monitor_Parameters

列名称
MONITOR_PA

“监视器名称”属性

描述
监视器 MBean 的 JMX 对象名

类型
String

名称

特性名称
Monitor_Name

列名称
MONITOR_NA

字符串通知
“字符串通知”属性组是不基于事件的属性组，用于发送所有字符串监视器所接收的事件。

以下列表包含关于“字符串通知”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“通知类型”属性

描述
接收到的通知类型。描述 MBean 的受观测属性如何触发通知。

类型
字符串

名称

属性名称
Notification_Type

列名
NOTIFICATI

“监视器标识”属性 - 此属性是键属性

描述
监视器的唯一整数标识

类型
整数

名称

属性名称
Monitor_ID

列名
MONITOR_ID

“受观测 MBean”属性

描述
正在监视其属性的 MBean

类型
字符串

名称

属性名称
Observed_MBean

列名
OBSERVED_M

“受观测属性”属性

描述
“受观测 MBean”中监视的属性的名称

类型
字符串

名称

属性名称
Observed_Attribute

列名
OBSERVED_A

“比较字符串”属性

描述
比较操作中使用的字符串

类型
字符串

名称

属性名称
Compare_String

列名
COMPARE_ST

“字符串值”属性

描述
触发通知的属性值

类型
字符串

名称

属性名称
String_Value

列名
STRING_VAL

“通知时间戳记”属性

描述
触发通知的时间

类型
时间

名称

属性名称
Notification_Time_Stamp

列名
NOTIFICAT0

“通知消息”属性

描述
通知中的消息

类型
字符串

名称

属性名称
Notification_Message

列名
NOTIFICAT1

“SNMP 事件”属性组

“SNMP 事件”属性组用于接收陷阱和通知。这些属性组是基于事件的属性组

以下列表包含关于“SNMP 事件”属性组中每个属性的信息：

注: 您可以更改这些属性的缺省显示名称。这些显示名称与各个属性的内部标识不同。

Enterprise_OID
生成陷阱的企业 OID。

Source_Address
发送陷阱的 SNMP 代理程序的主机名或 IP 地址。

Generic_Trap
从接收到的陷阱中抽取的通用陷阱号。可能的值包括：

```
0 ColdStart
1 WarmStart
2 LinkDown
3 LinkUp
4 Authentication Failure
5 EGPNeighborLoss
```

Specific_Trap
从接收到的陷阱中抽取的特定于企业的陷阱号。仅当 Generic_Trap = 6 时适用。

Alert_Name
陷阱配置文件的定义中指定的陷阱名称。

Category
陷阱配置文件的定义中指定的陷阱类别。

描述
陷阱配置文件的定义中指定的陷阱描述。最长描述长度为 256 个字符。

Enterprise_Name
在陷阱配置文件中指定，并通过陷阱对象标识确定的陷阱企业名称。

Source_Status
生成陷阱的代理程序在发送陷阱后所处的状态（在陷阱配置文件中的陷阱定义中指定）。

Source_Type
生成陷阱的代理程序的类型（在陷阱配置文件的陷阱定义中指定）。

Event_Variables
在陷阱协议数据单元 (PDU) 中接收到的变量绑定 (VarBind) 数据。此字符串的构造如下：

```
{OID[type]=value}{OID[type]=value}{oid[type]=value}...
```

其中：

oid
MIB 变量对象标识

type
SMI 数据类型

value
变量值

{}
每个三元组都括在花括号 ({}) 内。

注: 属性 Alert Name、Category、Description、Enterprise_Name、Source_Status 和 Source_Type 提供更多信息。在“**SNMP MIB 浏览器**”窗口中，选中**包括用于显示陷阱配置文件所定义信息的属性**复选框即可包括这些属性。

“JMX 事件”属性组

“JMX 事件”属性组用于接收来自 MBean Server 的通知。

这些属性组是不基于事件的属性组，使用以下可由代理程序开发者编辑的属性生成。

以下列表包含关于“JMX 事件”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“类型”属性

描述
通知类型

类型
字符串

名称

属性名称
Type

列名
TYPE

“源”属性

描述
这是导致发送通知的 MBean

类型
字符串

名称

属性名称
Source

列名
SOURCE

“序号”属性

描述
通知对象中的序号

类型
字符串

名称

属性名称
Sequence_Number

列名
SEQUENCE_N

“消息”属性

描述
通知消息

类型
字符串

名称

属性名称
Message

列名
MESSAGE

“用户数据”属性

描述
通知中的用户数据对象

类型
字符串

名称

属性名称
User_Data

列名
USER_DATA

Ping 属性组

Ping 属性组包含发送到多列设备的 ICMP Ping 的结果。

以下列表包含关于 Ping 属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称。

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
当构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时，从代理程序系统收集到的时间。或者进行存储以记录历史数据。它表示代理程序系统的本地时区。

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“地址”属性 - 此属性是键属性

描述
受监视主机的 IP 地址。

类型
具有枚举值的字符串。如果 IP 地址未知，那么会显示值 UNKNOWN_ADDRESS。对于此枚举，仓库和查询将返回 0.0.0.0。任何其他 IP 地址值均按原样显示。

名称

属性名称
Address

列名
PNGADDR

“设备条目”属性 - 此属性是键属性

描述
此节点在设备列表文件中的条目。

类型
字符串

名称

属性名称
Device_Entry

列名
PINGDEVC

“当前响应时间”属性

描述
受管节点的 ICMP 请求的当前网络响应时间（毫秒）。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的值为 `TIMEOUT(-1)` 和 `SEND_FAILURE(-2)`。所有其他值均显示数字值。

名称

属性名称
Current_Response_Time

列名
PINGRSTM

“名称”属性

描述
受管节点的主机名。如果无法通过 DNS 解析节点地址，那么将显示点分十进制 IP 地址。

类型
具有枚举值的字符串。如果主机名未知，那么会显示值 UNKNOWN_HOSTNAME。对于此枚举，仓库和查询将返回 0.0.0.0。任何其他主机名值均按原样显示。

名称

属性名称
Name

列名
PNGNAME

“节点描述”属性

描述
受管节点的描述。

类型
字符串

名称

属性名称
Node_Description

列名
PNGDESC

“节点状态”属性

描述
受管节点的当前运行状态。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的值为 `INVALID(-2)`、`UNKNOWN(-1)`、`INACTIVE(0)` 和 `ACTIVE(1)`。

名称

属性名称
Node_Status

列名
PNGSTAT

“节点类型”属性

描述
受管节点的类型。如果节点处于联机状态，那么类型为 IP Node。如果节点处于脱机状态，那么类型为 Unknown。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的值为 `UNKNOWN(0)` 和 `IP NODE(1)`。

名称

属性名称
Node_Type

列名
PNGTYPE

状态时间戳记

描述
最近一次检查节点的日期和时间。

类型
时间

名称

属性名称
Status_Timestamp

列名
PNGTMSP

HTTP 属性组

受管 URL 和 URL 对象这两个 HTTP 属性组用于接收来自 URL 及其中对象的信息。

有关“受管 URL”和“URL 对象”表中使用的语法的信息，请参阅第 1090 页的『HTTP 属性的特定字段』。

受管 URL

以下列表包含关于“受管 URL”属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

URL 属性 - 此属性是键属性

描述
受监视的 URL。

类型
字符串

名称

属性名称
URL

列名
HTTPURL

“响应时间”属性

描述
下载响应所用的时间（毫秒）。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串，仓库和查询返回数字。已定义的值为 `TIMEOUT (-1)`。

名称

属性名称
Response_Time

列名
HTTPURL

“页面大小”属性

描述
HTTP 请求所返回页面的大小。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串，仓库和查询返回数字。已定义的值为 `NO_RESPONSE_RECEIVED(-1)`。

名称

属性名称
Page_Size

列名
PAGESZ

“页面对象数”属性

描述
与受监视页面相关联的对象总数。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串，仓库和查询返回数字。已定义的值为 `NOT_COLLECTED(-1)`。

名称

属性名称
Page_Objects

列名
PGOBJS

“对象总大小”属性

描述
HTTP 请求所返回页面的大小。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串，仓库和查询返回数字。已定义的值为 `NOT_COLLECTED(-1)`。

名称

属性名称
Total_Object_Size

列名
TOTOSZ

“页面标题”属性

描述
所接收 URL 页面的页面标题。

类型
字符串

名称

属性名称
Page_Title

列名
PAGETTL

“服务器类型”属性

描述
在目标 URL Web 站点上使用的服务器的类型。

类型
字符串

名称

属性名称
Server_Type

列名
SRVTYP

“响应代码”属性

描述
HTTP 请求的响应代码。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串，仓库和查询返回数字。已定义的值为 `NO_RESPONSE_RECEIVED(-1)`。

名称

属性名称
Response_Code

列名
CODE

“状态”属性

描述
当前受管 URL 的状态（OK 或状态描述）。

类型
字符串

名称

属性名称
Status

列名
STATUS

“URL 别名”属性

描述
URL 的用户指定别名。

类型
字符串

名称

属性名称
URL_Alias

列名
ALIAS

“用户数据”属性

描述
随 URL 一起指定的用户数据。

类型
字符串

名称

属性名称
User_Data

列名
USER

URL 对象

以下列表包含关于“URL 对象”属性组中每个属性的信息:

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

URL 属性 - 此属性是键属性

描述
受监视的 URL。

类型
字符串

名称

属性名称
URL

列名
HTTPURL

“对象名称”属性

描述
目标 URL 内的页面对象的名称。

类型
字符串

名称

属性名称
Object_Name

列名
ONAME

“对象大小”属性

描述
目标 URL 内的页面对象的大小（字节）。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的值为 NOT_COLLECTED （-1）和 OBJECT_NOT_FOUND （-2）。所有其他值均显示数字值。

名称

属性名称
Object_Size

列名
SIZE

“对象响应时间”属性

描述
下载对象所用的时间（毫秒）。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的值为 NOT_COLLECTED （-1）、NO_RESPONSE_RECEIVED （-2）和 STATUS_CODE_ERROR （-3）。所有其他值均显示数字值。

名称

属性名称
Object_Response_Time

列名
ORTIME

发现属性组

此属性组代表针对子节点类型定义的子节点实例集合

创建子节点类型时，将创建一个属性组，它代表针对该子节点类型定义的子节点实例集合。其中每个属性组包含相同的属性集合。

以下列表包含关于“发现”属性组中的每个属性的信息。粗体字名称显示属性在 Tivoli Enterprise Portal 中的显示方式：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称

类型
字符串

名称

属性名称
Node

列名称
ORIGINNODE

“时间戳记”属性

描述
构建数据行并将其发送到 Tivoli Enterprise Monitoring Server（或进行存储以用于历史记录用途）时，从代理程序系统收集的时间。它表示代理程序系统的本地时区。

类型
时间

名称

属性名称
Timestamp

列名称
TIMESTAMP

“子节点 MSN”属性

描述
子节点代理程序的“受管系统名称”。

类型
字符串

名称

属性名称
Subnode_MSN

列名称
SN_MSN

“子节点亲缘关系”属性

描述
子节点代理程序的亲缘关系。

类型
字符串

名称

属性名称
Subnode_Affinity

列名称
SN_AFFIN

“子节点类型”属性

描述
此子节点的节点类型。

类型
字符串

名称

属性名称
Subnode_Type

列名称
SN_TYPE

“子节点资源名称”属性

描述
子节点代理程序的资源名称。

类型
字符串

名称

属性名称
Subnode_Resource_Name

列名称
SN_RES

“子节点版本”属性

描述
子节点代理程序的版本。

类型

名称

属性名称
Subnode_Version

列名称
SN_VER

Take Action Status 属性组

Take Action Status 属性组包含代理程序已处理的操作的状态。

此属性组基于事件，并包含关于 Take Action Status 属性组中每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称。

类型
字符串

名称

属性名称
Node

列名
ORIGINNODE

“时间戳记”属性

描述
这是构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时，或进行存储以记录历史数据时，从代理程序系统收集的时间。它表示代理程序系统的本地时区。

类型
时间

名称

属性名称
Timestamp

列名
TIMESTAMP

“操作名称”属性

描述
已运行的操作的名称

类型
字符串

名称

属性名称
Action_Name

列名
TSKNAME

“操作状态”属性

描述
操作的状态。

类型
具有枚举值的整数。值包括：正常 (0)、不适用 (1)、一般错误 (2)、警告 (3)、不在运行中 (4)、依赖项不在运行中 (5)、已在运行中 (6)、必备项不在运行中 (7)、已超时 (8)、不存在 (9)、未知 (10)、依赖项仍在运行中 (11)、用户权限不足 (12)

名称

属性名称
Action_Status

列名
TSKSTAT

“操作应用程序返回码”属性

描述
操作所启动的应用程序的返回码。

类型
整数

名称

属性名称
Action_App_Return_Code

列名
TSKAPRC

“操作消息”属性

描述
与操作返回码关联的消息。

类型
字符串

名称

属性名称
Action_Message

列名
TSKMSGE

“操作实例”属性

描述
与运行操作所生成的输出关联的实例。如果操作是系统命令，那么实例是命令输出的行号。

类型
字符串

名称

属性名称
Action_Instance

列名
TSKINST

“操作结果”属性

描述
运行操作所生成的输出。

类型
字符串

名称

属性名称
Action_Results

列名
TSKOUTP

“操作命令”属性

描述
操作所运行的命令。

类型
字符串

名称

属性名称
Action_Command

列名
TSKCMND

“操作节点”属性

描述
运行操作的节点。

类型
字符串

名称

属性名称
Action_Node

列名
TSKORGN

“操作子节点”属性

描述
运行操作的子节点。

类型
字符串

名称

属性名称
Action_Subnode

列名
TSKSBND

“操作标识”属性

描述
操作的标识。

类型
整数

名称

属性名称
Action_ID

列名
TSKID

“操作类型”属性

描述
操作的类型。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将会显示字符串，仓库和查询将返回数字。已定义的值为：未知 (0)、自动化 (1)。

名称

属性名称
Action_Type

列名
TSKTYPE

“操作所有者”属性

描述
启动操作的情境或用户的名称。

类型
字符串

名称

属性名称
Action_Owner

列名
TSKOWNR

“日志文件状态”属性组

“日志文件状态”属性组包含的信息反映此代理程序所监视的日志文件的状态。

如果您具有日志属性组并且代理程序处于缺省最低的 Tivoli Monitoring V6.2.1 或更高版本，那么包含“日志文件状态”属性组。“日志文件状态”属性组包含两个定义为 64 位数字的属性，以便能够处理大文件。64 位数字属性支持由 Tivoli Monitoring V6.2.1 或更高版本提供。

以下列表包含关于“日志文件状态”属性组中的每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称。

类型
字符串

名称

属性名称
节点

列名
ORIGINNODE

“时间戳记”属性

描述
此值是构建数据行并从代理程序发送到 Tivoli Enterprise Monitoring Server 时，从代理程序系统进行收集时的时间。或进行存储以记录历史数据。它表示代理程序系统的本地时区。

类型
时间

名称

属性名称
时间戳记

列名
TIMESTAMP

“表名”属性 - 此属性是键属性

描述
正在其中监视此日志的表的名称

类型
字符串

名称

属性名称
表名

列名
TBLNAME

“文件名”属性 - 此属性是键属性

描述
正被监视的文件的名称

类型
字符串

名称

属性名称
文件名

列名
FILNAME

“正则表达式模式”属性 - 此属性是键属性

描述
这是导致此文件被监视的正则表达式模式（如果有）

类型
字符串

名称

属性名称
正则表达式模式

列名
REPATRN

“文件类型”属性

描述
此文件的类型（常规文件或管道）

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。已定义的值为 UNKNOWN(0)、REGULAR FILE(1) 和 PIPE(2)

名称

属性名称
文件类型

列名
FILTYPE

“文件状态”属性

描述
正被监视的文件的状态

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。已定义的值为：OK(0)、PERMISSION DENIED(1)、FILE DOES NOT EXIST(2)、INTERRUPTED SYSTEM CALL(4)、I/O ERROR(5)、NO SUCH DEVICE(6)、BAD FILE NUMBER(9)、OUT OF MEMORY(12)、ACCESS DENIED(13)、RESOURCE BUSY(16)、NOT A DIRECTORY(20)、IS A DIRECTORY(21)、INVALID ARGUMENT(22)、FILE TABLE OVERFLOW(23)、TOO MANY OPEN FILES(24)、TEXT FILE BUSY(26)、FILE TOO LARGE(27)、NO SPACE LEFT ON DEVICE(28)、ILLEGAL SEEK ON PIPE(29)、READ-ONLY FILE SYSTEM(30)、TOO MANY LINKS(31) 和 BROKEN PIPE(32)

名称

属性名称
文件状态

列名
FILSTAT

“匹配的记录数”属性

描述
此日志中已处理并与其中一种指定模式相匹配的记录的数目

类型
整数

名称

属性名称
匹配的记录数

列名
RECMTCH

“不匹配的记录数”属性

描述
发送到 UnmatchLog 的已处理记录的数目；这些记录与任何模式都不匹配

类型
整数

名称

属性名称
不匹配的记录数

列名
RECUNMT

“已处理的记录数”属性

描述
自代理程序启动以来，在此日志中处理的记录数（包含并非匹配项/事件的记录）

类型
整数

名称

属性名称
已处理的记录数

列名
RECPROC

“当前文件位置”属性

描述
在所监视的文件中的当前位置（以字节计）。此位置之前的数据已处理，此位置之后的数据未处理。不适用于管道。

类型
整数

名称

属性名称
当前文件位置

列名
OFFSET

Current® “当前文件大小”属性

描述
这是受监视文件的当前大小。不适用于管道。

类型
整数

名称

属性名称
当前文件大小

列名
FILESIZE

“上次修改时间”属性

描述
上次写入所监视文件的时间。不适用于管道。

类型
时间戳记

名称

属性名称
上次修改时间

列名
LASTMOD

“代码页”属性

描述
所监视文件的语言代码页

类型
字符串

名称

属性名称
代码页

列名
CODEPG

“日志文件正则表达式统计信息”属性组

“日志文件正则表达式统计信息”属性组包含的信息显示了日志文件正则表达式搜索表达式的统计信息。

您可以使用正则表达式对记录进行过滤或定义记录。此属性组显示这两种类型的相关信息。如果“结果类型”属性包含 INCLUDE 或 EXCLUDE，那么将使用过滤器对记录进行过滤。如果“结果类型”属性包含 BEGIN 或 END，那么将使用过滤器来定义记录。CPU 测量是基于操作系统所公开数据的粒度的近似值。如果正则表达式的求值时间非常短，这些测量可能会产生值 0.00。使用 CPU 时间来确定正则表达式的相对成本并优化特定正则表达式的行为。

如果您具有日志属性组并且代理程序处于 Tivoli Monitoring V6.2.1 或更高版本，那么包含“日志文件正则表达式统计信息”属性组。“**代理程序信息**”页面上将选择最低的 Tivoli Monitoring 版本。有关更多信息，请参阅第 988 页的『对代理程序命名和配置』。“日志文件正则表达式统计信息”属性组包含定义为 64 位数字的属性，以便能够处理较长的持续时间。对 64 位数字属性的支持由 Tivoli Monitoring V6.2.1 或更高版本提供。

以下列表包含关于“日志文件正则表达式统计信息”属性组中的每个属性的信息：

“节点”属性 - 此属性是键属性

描述
代理程序的受管系统名称。

类型
字符串

名称

属性名称
节点

列名
ORIGINNODE

“时间戳记”属性

描述
收集数据时代理程序的本地时间。

类型
时间

名称

属性名称
时间戳记

列名
TIMESTAMP

“表名”属性 - 此属性是键属性

描述
这是日志文件属性组的名称。

类型
字符串

名称

属性名称
表名

列名
TBLNAME

“属性名称”属性 - 此属性是键属性

描述
这是要对其应用此过滤器的属性的名称。

类型
字符串

名称

属性名称
属性名称

列名
ATRNAME

过滤器编号

描述
这是用于此属性的过滤器的序号（起始于零）。

类型
整型（数字属性）

名称

属性名称
过滤器名称

列名
FLTRNUM

“结果类型”属性

描述
结果类型可以是 INCLUDE 或 EXCLUDE，分别用于在过滤器匹配时接受或拒绝此属性。结果类型可以是 BEGIN 或 END，这两个值分别用于指定多行记录的记录开头和末尾。

类型
具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。如果使用过滤器来过滤记录，已定义的值为 INCLUDE(1) 或 EXCLUDE(2)。如果使用过滤器来定义记录，已定义的值为 BEGIN(3) 或 END(4)。

名称

属性名称
结果类型

列名
RSTTYPE

“平均处理器时间”属性

描述
这是用于处理此属性的过滤器的平均处理器秒数。平均处理器时间是处理器总秒数除以过滤器计数而得。

类型
整数（标尺）

名称

属性名称
平均处理器时间

列名
CPUTAVG

“处理器时间”属性

描述
这是用于处理此属性的过滤器的总处理器秒数。处理器时间是累计时间，并且将进行截断而不是四舍五入。与 Linux /proc/<pid>/task/*thread*/stat 文件类似。

类型
整型（计数器）

名称

属性名称
处理器时间

列名
CPUTIME

“最长处理器时间”属性

描述
这是用于单一过滤器处理的最大处理器秒数。最大时间有可能为零，发生这种情况的原因是，此过滤器从未被使用，或者每个过滤器处理耗用的时间都不足 0.01 秒。

类型
整数（标尺）

名称

属性名称
最长处理器时间

列名
CPUTMAX

“最短处理器时间”属性

描述
这是用于单一过滤器处理的最小处理器秒数。最小时间有可能为零，发生这种情况的原因是，过滤器处理的耗用时间不足 0.01 秒。

类型
整数（标尺）

名称

属性名称
最短处理器时间

列名
CPUTMIN

“过滤器计数”属性

描述
过滤器运行的次数。将此值与总处理器时间配合使用可以计算平均处理器时间。

类型
整型（计数器）

名称

属性名称
过滤器计数

列名
COUNT

“过滤器匹配次数”属性

描述
这是此过滤器运行且属性匹配的次数。

类型
整型（计数器）

名称

属性名称
过滤器匹配次数

列名
COUNTMA

“过滤器不匹配次数”属性

描述
这是此过滤器运行且属性不匹配的次数。

类型
整型（计数器）

名称

属性名称
过滤器不匹配次数

列名
COUNTUN

“正则表达式模式”属性 - 此属性是键属性

描述
这是用于匹配的正则表达式。

类型
字符串

名称

属性名称
正则表达式模式

列名
REGXPAT

“上次匹配时间”属性

描述
上次使用此过滤器且结果匹配的时间。

类型
时间

名称

属性名称
上次匹配时间

列名
LASTMAT

“上次不匹配时间”属性

描述
上次使用此过滤器且结果不匹配的时间。

类型
时间

名称

属性名称
上次不匹配时间

列名
LASTUMA

为现有代理程序创建应用程序支持扩展

对于 IBM Tivoli Monitoring 环境，您可以构建用于分发所创建的定制工作空间、情境、查询和“执行操作”命令的可安装包，作为对现有代理程序的应用程序支持扩展。

开始之前

有关如何创建定制情境、工作空间、“执行操作”命令和查询的更多信息，请参阅第 1149 页的『创建工作空间、“执行操作”命令和情境』。

关于此任务

要点: 此任务并不是向所要构建的代理程序添加应用程序支持。要向所要构建的代理程序添加应用程序支持，请参阅第 1179 页的『导入应用程序支持文件』。

过程

1. 从 Agent Builder 中，选择**文件** > **新建** > **其他**。
2. 选择 **Agent Builder** 下的 **Agent Builder 应用程序支持扩展**。
3. 单击**下一步**以转到“**IBM Tivoli Monitoring 应用程序支持扩展**”向导的欢迎页面。
4. 在欢迎页面上，单击**下一步**。
5. 输入项目名称，然后单击**完成**。

创建“应用程序支持扩展”项目

使用 Agent Builder 创建“应用程序支持扩展”项目。

过程

1. 从 Agent Builder 中，选择**文件** > **新建** > **其他**。
2. 选择 **Agent Builder** 下的 **Agent Builder 应用程序支持扩展**。
3. 单击**下一步**以转到 **IBM Tivoli Monitoring 应用程序支持扩展**向导的欢迎页面。
4. 在欢迎页面上，单击**下一步**。
5. 输入项目名称，然后单击**完成**。

将支持文件添加至项目

将支持文件添加至“应用程序支持扩展”项目

开始之前

创建“应用程序支持扩展”项目。有关更多信息，请参阅第 1243 页的『创建“应用程序支持扩展”项目』。

过程

1. 右键单击“应用程序支持扩展”项目，并选择 **IBM Tivoli** > **导入应用程序支持扩展**
2. 在“**导入信息**”窗口中，选择 Tivoli Enterprise Portal Server 所在主机的名称，或者单击**添加**以添加主机。
3. 在**应用程序**字段中，输入代理程序产品代码。
4. 输入正在为其创建定制应用程序支持的代理程序的亲缘关系。

   代理程序亲缘关系是 Tivoli Monitoring 内部标识，用于使工作空间、查询和其他的项与代理程序相关联。它在 Tivoli Monitoring 安装中必须是唯一的。请单击**浏览**以打开“**节点类型**”窗口，并从列表中选择此信息，而不要输入此信息。
5. 您认为导入信息无误时，请单击**完成**。
6. 在“**情境**”窗口中，从“可用的情境”列表选择要导入的情境。

单击 **<<** 以将其添加到“选择的情境”列表，并单击**确定**。此时在项目下将创建一个新文件夹，其中包含安装工作空间、情境和查询所必需的文件。

7. 在“**查询**”窗口中，从“可用的查询”列表选择要导入的查询。
   单击 **<<** 以将其添加到“选择的查询”列表，并单击**确定**。
8. 在“**执行操作**”窗口中，从“可用的执行操作”列表选择要导入的“执行操作”命令。
   单击 **<<** 以将其添加到“选择的执行操作”列表，并单击**确定**。代理程序的支持文件将放置在项目的相应文件夹中。

下一步做什么

您可以根据需要对任意数目的不同代理程序重复此过程。Agent Builder 会根据“应用程序支持扩展”项目中的所有支持文件创建单个安装映像。

生成 Application Support Extension 安装映像

生成 Application Support Extension 安装映像。

过程

1. 右键单击“应用程序支持扩展”项目，并选择 **IBM Tivoli** > **创建应用程序支持扩展安装映像**。
2. 在“**应用程序支持扩展信息**”窗口中，输入映像所在的目录。
3. Application Support Extension 必须具有自己的产品代码。请输入新代理程序的注册产品代码。您可以使用保留供 Agent Builder 使用的其中一个产品代码。允许的值是 `K00-K99`、`K{0-2}{A-Z}` 和 `K{4-9}{A-Z}`。

   注: 这些值仅供内部使用，而不用于要共享或销售的代理程序。如果要创建与他人共享的代理程序，必须向 toolkit@us.ibm.com 发送说明，以保留产品代码。请求产品代码必须包含要构建的代理程序的描述。然后，将为您分配、注册和返回产品代码。当您收到三个字母的产品代码时，系统将告知您如何使 Agent Builder 能够使用分配的产品代码。
4. 输入 Application Support Extension 的名称。
5. 输入 Application Support Extension 的描述。
6. 以 VVRRMMFF 格式输入 Application Support Extension 的版本，其中 `vv` = 版本号；`rr` = 发行版号；`mm` = 修改号（修订包号）；`ff` = 临时修订号。
7. 单击**完成**。

安装 Application Support Extension

安装 Application Support Extension

过程

1. 将映像传输到 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 服务器。
2. 要安装 Tivoli Enterprise Monitoring Server 支持，运行以下命令之一：
   - 在 Windows 上：`install`*`KXX`*`TEMSSupport.bat`
   - 在 UNIX 上：`install`*`KXX`*`TEMSSupport.sh`

   此命令的格式如下所示：

```
installKXXTEMSSupport[.bat | .sh] <ITM Install Directory> [-s tems_host]
 [-u tems_user]  \[-p tems_password]
```

3. 要安装 Tivoli Enterprise Portal Server 支持，运行以下命令之一：
   - 在 Windows 上：`install`*`KXX`*`TEPSSupport.bat`
   - 在 UNIX 上：`install`*`KXX`*`TEPSSupport.sh`

   此命令的格式如下所示：

```
installKXXTEPSSupport[.bat | .sh] <ITM Install Directory> [-r]
```

其中 `-r` 指示在安装后必须重新启动 Tivoli Enterprise Portal Server

将解决方案安装项目转换为“应用程序支持扩展”项目

将现有**解决方案安装项目**转换为“应用程序支持扩展”项目

关于此任务

如果存在要转换为“应用程序支持扩展”项目的现有**解决方案安装项目**，请完成下列步骤：

注: 在**解决方案安装项目**中，将仅迁移支持文件。

过程

1. 右键单击**解决方案安装项目**并选择 **IBM Tivoli > 转换解决方案安装项目**。
2. 输入新的“应用程序支持扩展”项目的名称，或者从列表中选择现有的项目
3. 单击**完成**。

生成 Cognos 数据模型

Agent Builder 可以为每个代理程序生成一个 Cognos 数据模型。使用该数据模型可将代理程序信息导入到 Cognos Framework Manager 中，以便创建报告。

您可以在 Framework Manager 中打开并查看此 Cognos 数据模型，而 Framework Manager 将构建要发布到 Tivoli Common Reporting 的模型包。在发布之前，还可以在 Framework Manager 中对此数据模型进行定制或修改。

创建报告后，Agent Builder 还允许将最终的报告包导入到 Agent Builder 项目中。此功能使您将来可以使用代理程序包中已包含的报告来生成代理程序项目。作为代理程序安装映像组成部分打包的报告可以导入到生产环境中的 Tivoli Common Reporting 中。

注: 在本文档中，请注意下列约定：

· K*xx* 或 k*xx* 是指对代理程序指定的产品代码，例如 k99。

· *dbType* 是指 Tivoli Data Warehouse 所使用的数据库，例如 Db2。

生成 Cognos 数据模型的先决条件

在生成 Cognos 数据模型之前，请完成下列任务

关于此任务

注:

· 这些步骤仅需完成一次，这是因为，将来使用 Agent Builder 生成的所有数据模型都将使用此环境。

· 您最好创建一个隔离的开发环境，用于测试代理程序和创建报告。

过程

1. 安装并配置 第 1246 页的『Tivoli Data Warehouse』。
2. 在 Tivoli Data Warehouse 中创建表及过程。
   a) 第 1246 页的『在 Tivoli Data Warehouse 中创建表和过程』.
   b) 第 1248 页的『以 Tivoli Reporting and Analytics Model 填充 Tivoli Data Warehouse』.
3. 安装并配置 第 1248 页的『Tivoli Common Reporting』。
4. 安装并配置 第 1249 页的『Framework Manager』。

Tivoli Data Warehouse
关于 Tivoli Data Warehouse。

要创建报告，您需要在环境中安装并配置 Tivoli Data Warehouse、Warehouse Proxy Agent 和 Summarization and Pruning Agent。有关更多信息，请参阅《IBM Tivoli Monitoring 安装与设置指南》。

在 *Tivoli Data Warehouse* 中创建表和过程
在 Tivoli Data Warehouse 中创建或变更 ManagedSystem 表和存储过程

关于此任务

生成的 Cognos 数据模型包含用于定义 ManagedSystem 维度的 ManagedSystem 表。ManagedSystem 维度支持创建可以关联受管系统的报告。例如，对于子节点代理程序，可以使用该维度来确定特定代理程序实例的子节点。

ManagedSystem 表并非由 Tivoli Data Warehouse 创建。因此，在 Agent Builder 中生成代理程序时，将为每个数据库平台生成 SQL 脚本，这些脚本将执行下列操作：

- 创建 ManagedSystem 表。如果 Tivoli Data Warehouse 中不存在该表，请使用此脚本。
- 编辑 ManagedSystem 表。如果 Tivoli Data Warehouse 中存在该表，请使用此脚本。其他报告产品可以创建 ManagedSystem 表，但它们创建的表中没有全部的必需列。
- 创建一个根据 Tivoli Data Warehouse 中的表填充 ManagedSystem 表的存储过程。

这些脚本只需运行一次。

运行 *Db2* 脚本以在 *Tivoli Data Warehouse* 中创建表和过程
对于 Db2 数据库，请使用以下脚本在 Tivoli Data Warehouse 中创建表

开始之前

适用于 Db2 的脚本位于以下目录中：

```
reports/db2/Kxx/reports/cognos_reports/itmkxx/db_scripts
```

过程

1. 生成的脚本（create_table.sql、alter_table.sql 和 create_procedure.sql）全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中，如果 Tivoli Data Warehouse 用户标识并非 *itmuser*，请将出现的所有 *itmuser* 更改为正确的用户标识。
2. 以 Tivoli Data Warehouse 用户身份连接到 Tivoli Data Warehouse：

   ```
   db2 connect to <Tivoli Data Warehouse alias name> user
   <Tivoli Data Warehouse user id> using  <password>
   ```

3. 确定 ManagedSystem 表是否存在：

   ```
   db2 "select count(*) from sysibm.systables where name = 'MANAGEDSYSTEM'
   and creator=upper ('<Tivoli Data Warehouse user id>')"
   ```

4. 创建或变更该表。
   - 如果以上查询返回 1，那么表明该表已存在。请运行变更脚本：

     ```
     db2 -tvf alter_table.sql
     ```

   - 如果以上查询返回 0，那么表明该表不存在。请运行创建脚本：

     ```
     db2 -tvf create_table.sql
     ```

5. 运行脚本以创建存储过程：

   ```
   db2 -td@ -f create_procedure.sql
   ```

运行 *Oracle* 脚本以在 *Tivoli Data Warehouse* 中创建表和过程
对于 Oracle 数据库，请使用以下脚本在 Tivoli Data Warehouse 中创建表

开始之前

适用于 Oracle 的脚本位于以下目录中：

```
reports/oracle/K<i>xx</i>/reports/cognos_reports/itmk<i>xx</i>/db_scripts
```

过程

1. 生成的脚本（create_table.sql、alter_table.sql 和 create_procedure.sql）全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中，如果 Tivoli Data Warehouse 用户标识并非 *itmuser*，请将出现的所有 *itmuser* 更改为正确的用户标识。
2. 启动 sqlplus：

   ```
   sqlplus <IBM Tivoli Monitoring user ID>/<password>@
   <Tivoli Data Warehouse SID>
   ```

3. 确定 ManagedSystem 表是否存在：

   ```
   select count(*) from user_tables where table_name = 'MANAGEDSYSTEM';
   ```

4. 创建或变更该表。
   - 如果以上查询返回 1，那么表明该表已存在。请运行变更脚本：

     ```
     @<path to alter_table.sql>;
     ```

   - 如果以上查询返回 0，那么表明该表不存在。请运行创建脚本：

     ```
     @<path to create_table.sql>;
     ```

5. 运行脚本以创建存储过程：

   ```
   @<path to create_procedure.sql>;
   ```

运行 *SQL Server 2005* 和 *2008* 脚本以在 *Tivoli Data Warehouse* 中创建表和过程

开始之前

适用于 SQL Server 的脚本位于以下目录中：

```
reports/mssql/K<i>xx</i>/reports/cognos_reports/itmk<i>xx</i>/db_scripts
```

过程

1. 生成的脚本（create_table.sql、alter_table.sql 和 create_procedure.sql）全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中，如果 Tivoli Data Warehouse 用户标识并非 *itmuser*，请将出现的所有 *itmuser* 更改为正确的用户标识。
2. 确定 ManagedSystem 表是否存在：

   ```
   osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
   <Tivoli Data Warehouse database name> -Q "Select count(*)
   from INFORMATION_SCHEMA.TABLES where table_name = 'ManagedSystem'"
   ```

3. 创建或变更该表。
   - 如果以上查询返回 1，那么表明该表已存在。请运行变更脚本：

     ```
     osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
     <Tivoli Data Warehouse database name> -I -n -i <path to alter_table.sql>
     ```

- 如果以上查询返回 0，那么表明该表不存在。请运行创建脚本：

```
osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d
<Tivoli Data Warehouse database name> -I -n -i <path to create_table.sql>
```

4. 运行脚本以创建存储过程：

```
osql -S <Server> -U <Tivoli Data Warehouse user ID> -P
<password> -d <Tivoli Data Warehouse database name>
-I -n -i <path to create_procedure.sql>
```

以 *Tivoli Reporting and Analytics Model* 填充 *Tivoli Data Warehouse*

使用提供的数据库脚本来填充 Tivoli Data Warehouse

关于此任务

Tivoli Reporting and Analytics Model (TRAM) 包含所有报告包通用的基本知识集。TRAM 通过每个数据库所特有的一组脚本进行安装。用于填充每个受支持数据库的必需脚本在代理程序安装映像的 reports 目录中提供。使用以下过程可以在 Tivoli Data Warehouse 中创建 Tivoli Reporting and Analytics Model 公共维度。

过程

1. 浏览到 Tivoli Reporting and Analytics Model 数据库脚本。
2. 解压缩代理程序包。
   - 在 Windows 系统上，代理程序包是 k*xx*.zip。
   - 在 Linux 和 UNIX 系统上，代理程序包是 k*xx*.tgz。
3. 转到相应的数据库脚本。
   - Db2 脚本位于代理程序包中的以下位置：

     ```
     reports/db2/K<xx>/reports/cognos_reports/itmk<xx>/db_scripts
     ```

   - Oracle 脚本位于代理程序包中的以下位置：

     ```
     reports/oracle/K<xx>/reports/cognos_reports/itmk<xx>/db_scripts
     ```

   - Microsoft SQL Server 脚本位于代理程序包中的以下位置：

     ```
     reports/mssql/K<xx>/reports/cognos_reports/itmk<xx>/db_scripts
     ```

4. 运行数据库脚本，以在 Tivoli Data Warehouse 中生成公共维度。每个脚本集都提供有包含用法指示信息的自述文件。
5. 确认这些脚本已将下列各表添加到 Tivoli Data Warehouse：

```
"Computer System", WEEKDAY_LOOKUP, MONTH_LOOKUP, TIMEZONE_DIMENSION, TIME_DIMENSION
```

Tivoli Common Reporting

Tivoli Common Reporting 包含 Cognos Business Intelligence 引擎，其中包含用于帮助您创建代理程序报告的元素。

您必须安装 Tivoli Common Reporting 并为其配置一个连接到 Tivoli Data Warehouse 的数据源。

安装 Tivoli Common Reporting

您必须安装 Tivoli Common Reporting。支持 V1.3、V2.1、V2.1.1 或更高版本。有关安装 Tivoli Common Reporting 的信息，请参阅安装 Tivoli Common Reporting。

配置 Tivoli Common Reporting

您必须配置 Tivoli Common Reporting。有关配置 Tivoli Common Reporting 的信息，请参阅配置 IBM Tivoli Common Reporting。

在 Tivoli Data Warehouse 与 Tivoli Common Reporting 之间创建数据源。有关更多信息，请参阅配置数据库连接。单击相应的数据库类型。请记录对该数据源指定的名称。缺省值为 **TDW**。

注: 数据源名称必须与“**Cognos 信息**”页面的**数据源**字段中的名称匹配。有关“**Cognos 信息**”页面的更多信息，请参阅 第 999 页的『Cognos 信息』。

Framework Manager
Framework Manager 是随 Tivoli Common Reporting 应用程序一起提供的应用程序，但必须单独安装和配置。

Framework Manager 用于查看和修改数据模型以及将数据模型发布到 Tivoli Common Reporting

安装 Framework Manager

您必须安装 Framework Manager。支持 V8.4、V8.4.1 或更高版本。

Framework Manager 随 Tivoli Common Reporting 一起提供，但必须手动安装。Tivoli Common Reporting 1.3 随附了 Framework Manager 8.4。Tivoli Common Reporting 2.1 和 2.1.1 随附了 Framework Manager 8.4.1。有关安装 Framework Manager 的信息，请参阅 *Tivoli Common Reporting User's Guide* 中的 Installing Framework Manager。

配置 Framework Manager

您必须配置 Framework Manager。有关配置 Framework Manager 的信息，请参阅 *Tivoli Common Reporting User's Guide* 中的 Configuring Framework Manager。

创建报告

使用 Framework Manager 发布代理程序模型，并使用 Report Studio 开始创建报告。

开始之前

代理程序完成后，它必须安装到 Tivoli Monitoring 环境中。另外，必须配置代理程序历史记录收集，并使该代理程序至少运行一个仓库上载时间间隔。必须配置摘要，并且 Tivoli Monitoring 中选择的摘要设置选项必须与 Agent Builder 中选择的摘要选项完全相同。将代理程序的数据上载到仓库之后，必须至少运行一次 Summarization and Pruning Agent。

1. 安装、配置并启动代理程序。
2. 对于每个要为其创建报告的属性组，请创建相应的历史记录收集并将其分发到代理程序。

   注: 仓库上载时间间隔缺省为“每天”。但是，您可能希望缩短此时间间隔。

   有关配置历史记录收集的信息，请参阅 *IBM Tivoli Monitoring Administrator's Guide* 中的 Managing historical data。
3. 在 Tivoli Monitoring 中，针对所有在步骤 2 创建了历史记录收集的属性组，配置摘要。

   注: 当您配置历史记录收集及摘要时，必须等待足够长的时间才能得到摘要表中的数据。

   注: 缺省情况下，Summarization and Pruning Agent 配置为每天凌晨 2 点运行一次。您可能想更改此设置。例如，可以将其配置为每小时运行一次。有关配置 Tivoli Data Warehouse 的信息，请参阅 *IBM Tivoli Monitoring Installation and Setup Guide* 中的 Setting up data warehousing。

关于此任务

在 Agent Builder 中生成代理程序将创建完整的 Framework Manager 项目，其中包含数据模型以及 Framework Manager 项目文件。Framework Manager 可以直接打开项目文件，这将打开数据模型以进行修改、定制或发布。

过程

注: 生成的代理程序数据模型包含每个属性组的所有摘要时间维度：每小时、每日、每周、每月、每季度和每年。只有为代理程序配置摘要及修剪之后，这些维度才会存在于代理程序的 Tivoli Data Warehouse 中。如果已选择这些维度，并且 Summarization and Pruning Agent 已创建并填充了表，也会如此。您可以定义

使用不存在的维度的报告，并将其发布至 Tivoli Common Reporting。此类报告直到 Summarization and Pruning Agent 创建摘要表之后才有效。

1. 在 Framework Manager 中打开代理程序数据模型：
   a) 打开 Framework Manager。
   b) 在“**欢迎**”页面中，单击**打开项目**。

      提示: 如果您是在 Framework Manager 中，请单击**文件**菜单中的**打开**。
   c) 浏览至代理程序数据模型。
      - 对于 Db2：

        ```
        reports/db2/Kxx/model/
        ```
      - 对于 Oracle：

        ```
        reports/oracle/Kxx/model/
        ```
      - 对于 Microsoft SQL Server：

        ```
        reports/mssql/Kxx/model/
        ```
   d) 选择代理程序项目文件 `Kxx.cpf`。

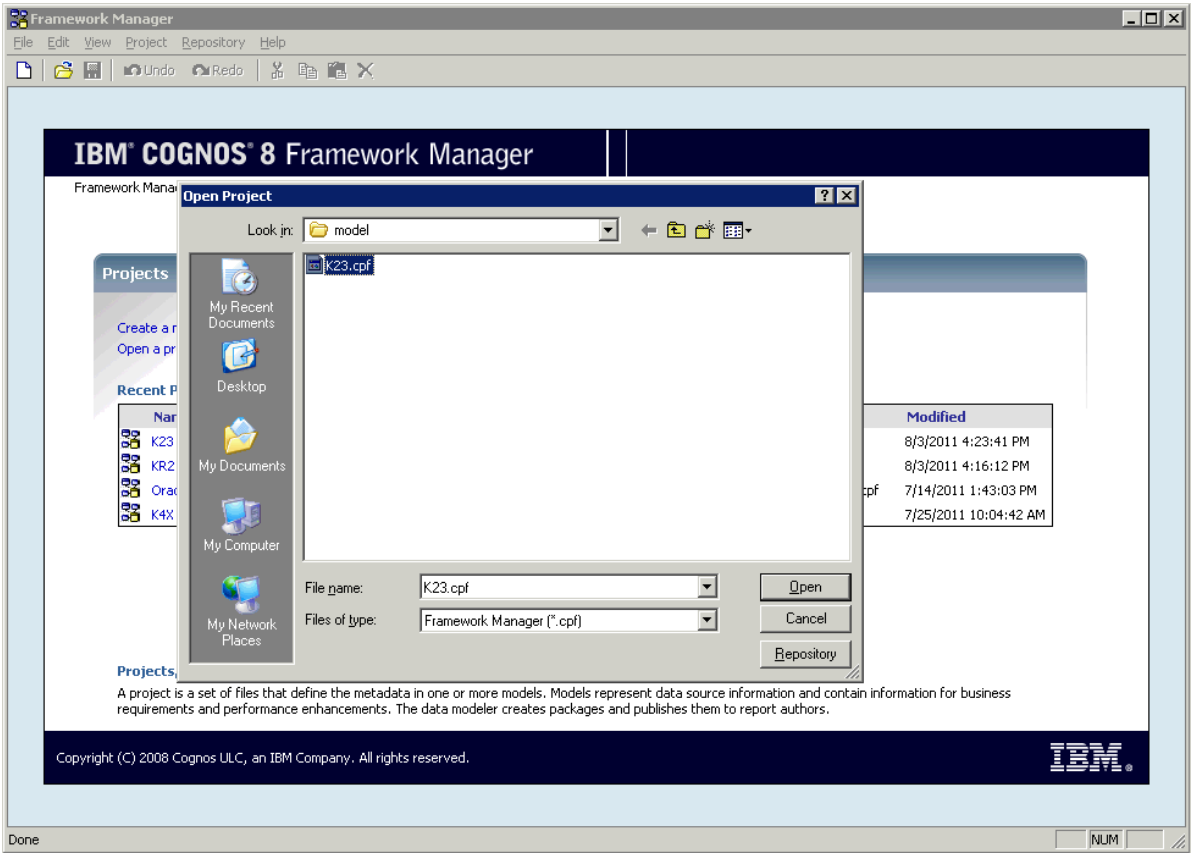


图 *80.* 选择代理程序项目文件

注: 在 Framework Manager 中打开代理程序项目时，代理程序名称将在“最近的项目”下列出。

2. 填写“受管系统表”。有关更多信息，请参阅第 1254 页的『填充 ManagedSystem 表』
3. 使用 Framework Manager 将代理程序模型发布至 Tivoli Common Reporting
   a) 打开 Framework Manager。
   b) 打开代理程序项目。
   c) 在导航树中，展开**包**。

d) 右键单击该代理程序包，然后选择**发布包**。

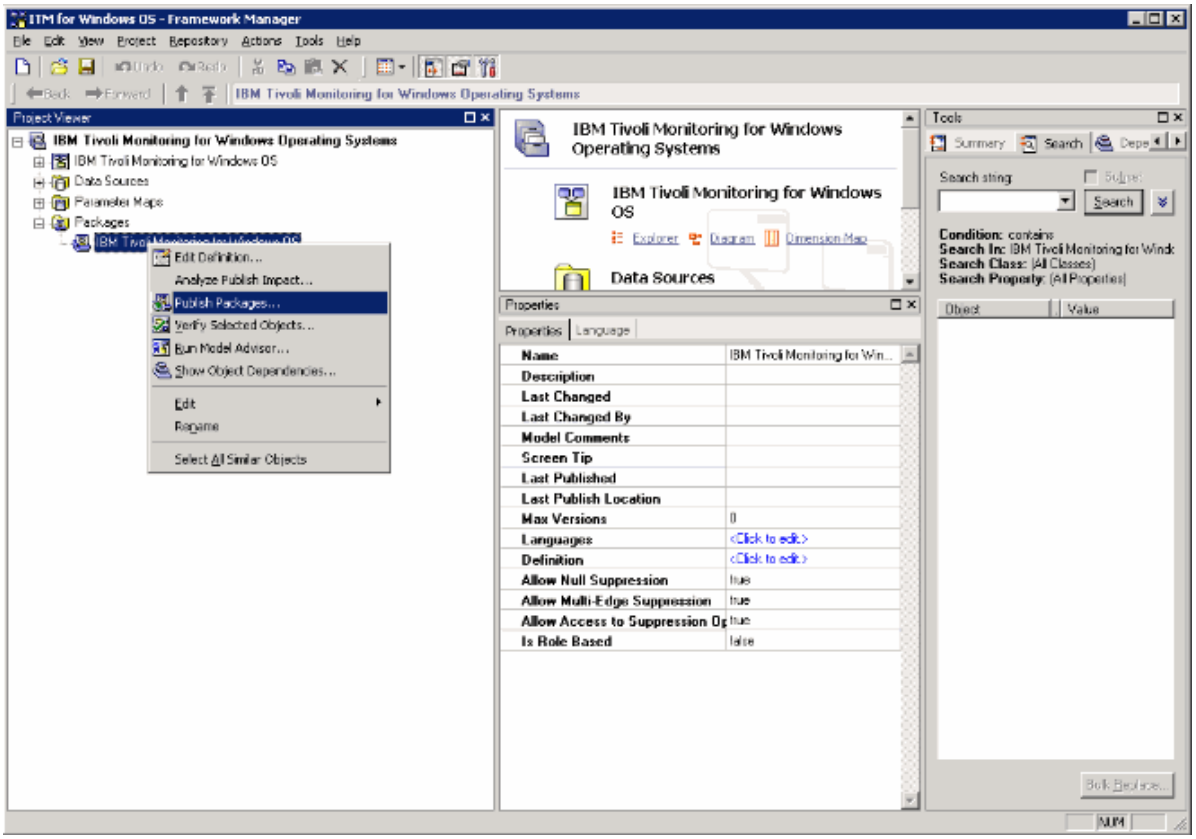


图 *81.* 选择发布包

4. 使用 Report Studio 创建新的报告或模板。
   a) 登录 Tivoli Common Reporting。
   b) 浏览至“公用文件夹”，在导航面板中展开**报告**，并选择 **Common Reporting**。

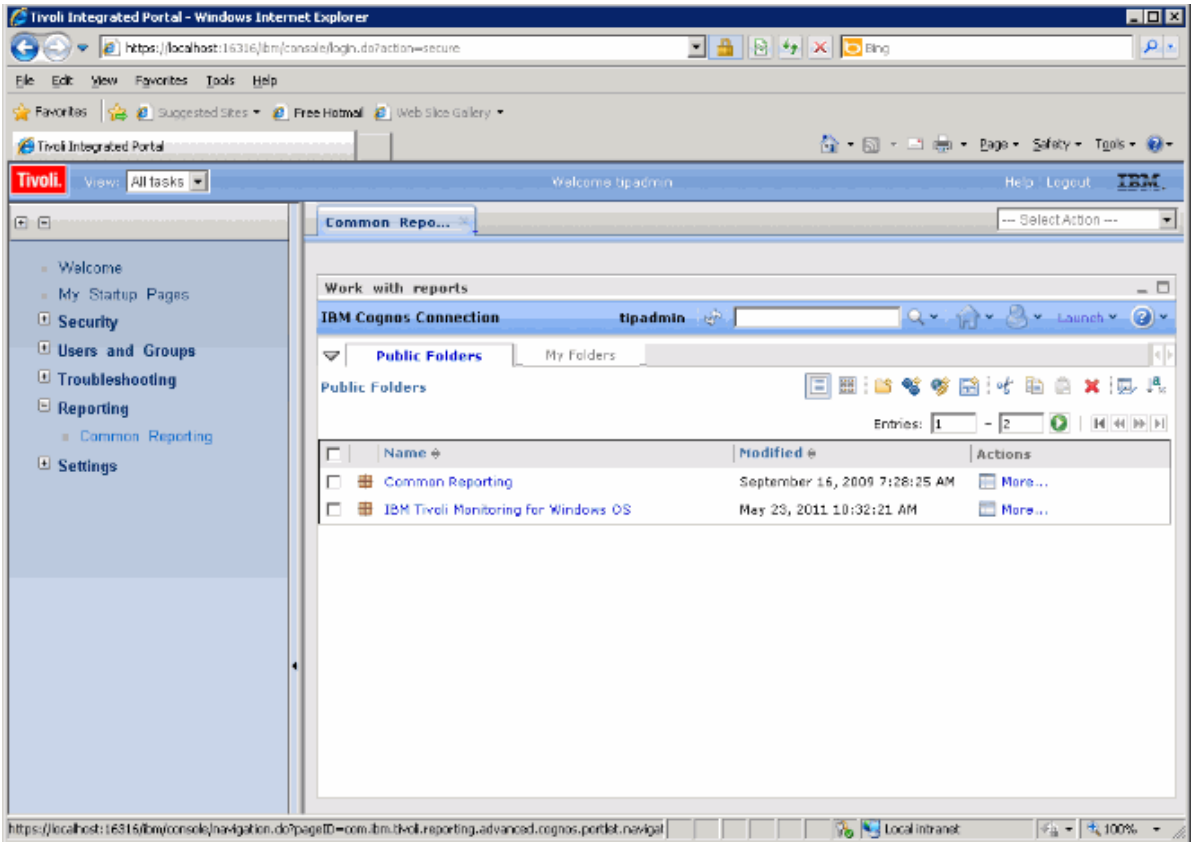


图 *82.* 选择 *Common Reporting*

c) 从提供的列表中选择 Tivoli Monitoring 代理程序。

d) 通过单击“启动”菜单并选择 **Report Studio** 或 **Query Studio**，打开报告创建工具。

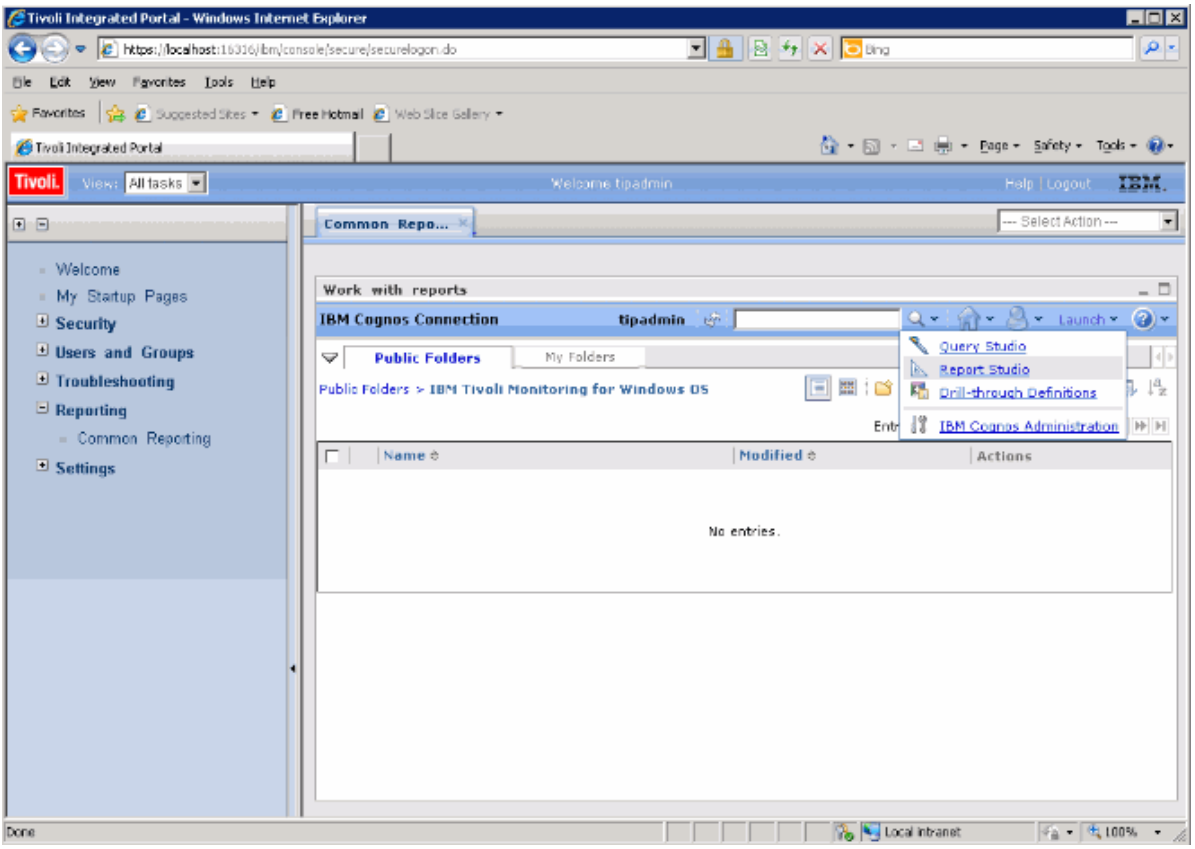


图 *83.* 选择 *Report Studio*

下一步做什么

您可以使用 Report Studio 来创建新的报告或模板，也可以对现有的报告或模板进行修改。

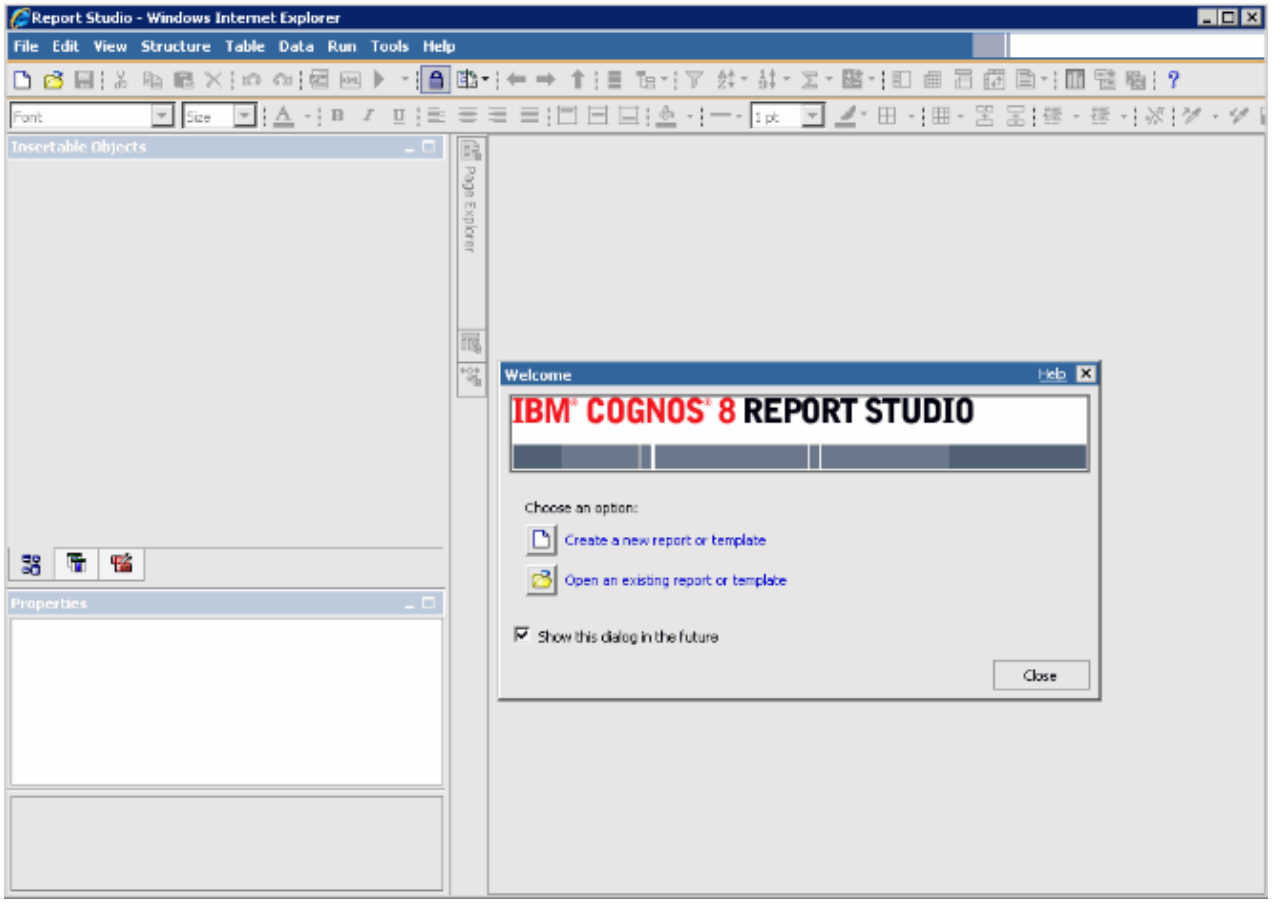


图 *84. Report Studio*

有关更多信息，请参阅 IBM Knowledge Center 上的 Tivoli Common Reporting 主题集合。

填充 ManagedSystem 表

ManagedSystem 表通过 kqz_populate_msn 存储过程进行填充。

有关更多信息，请参阅第 1255 页的『运行 Db2 存储过程』。此过程必须定期运行，以使 ManagedSystem 表包含受管系统名称的最新列表。

此存储过程将读取 Tivoli Data Warehouse 中的下列历史表（如果它们存在）：

- 代理程序的 Performance Object Status 表
- 代理程序的可用性表。用于监视进程或服务的代理程序具有可用性表。
- 代理程序的发现表。子节点代理程序将创建发现表。

必须对特定一组属性组启动历史数据收集。将生成一组脚本，这些脚本将对这些属性组创建并启动历史数据收集。如果您不希望使用脚本，那么属性组列表将列示在脚本的注释头块中。

创建了样本脚本，用于说明必须启用历史数据收集的表：

- `reports/configuretdw.sh`
- `reports/configuretdw.bat`

下表描述必需的自变量：

注: 必须指定 `-n` 或 `-m`，但不能同时指定这两者。

表 *301.* 必需自变量	
参数	**描述**
-h *candle_home*	Tivoli Monitoring 安装路径。
-u *teps_user*	您创建历史数据收集时用于进行登录的 Tivoli Enterprise Portal Server 用户。
-n *tems_name*	必须在其中启动收集的 Tivoli Enterprise Monitoring Server。通过使用以空格分隔的列表，可以指定多个 Tivoli Enterprise Monitoring Server。如果指定了多个 Tivoli Enterprise Monitoring Server，请将列表括在引号内。例如，-n “tems1 tems2”
-m *managed_system_group_or_managed_system*	必须对其启动收集的受管系统组或受管系统的名称。可以使用以空格分隔的列表来指定多个受管系统组或受管系统。如果指定了多个受管系统组或受管系统，请将列表括在引号内。例如，-m “msg1 msg2”

下表描述可选的自变量：

表 *302.* 可选自变量	
参数	**描述**
-s *teps_host*	Tivoli Enterprise Portal Server 的主机名或 IP 地址。如果未指定，那么将使用缺省值 localhost。
-p *teps_password*	使用 -u 选项指定的 Tivoli Enterprise Portal Server 用户的密码。如果未指定，那么脚本会提示您输入密码
-c *historical_collection_interval*	启动历史数据收集时使用的历史数据收集时间间隔。如果未指定，那么将使用缺省值 1h（1 小时）。有效值为：15m、30m、1h、12h 或 1d，其中 m 是分钟，h 是小时，d 是天。
-r *pruning_interval*	用于历史数据的修剪时间间隔。必须对历史数据进行修剪，以使表不会持续增大。如果未指定，那么将使用缺省值 2d（2 天）。使用 d 表示天，m 表示月，y 表示年。

启动历史数据收集后，必须定期运行 kqz_populate_msn 存储过程。此存储过程定期运行，以使 ManagedSystem 表包含 Tivoli Monitoring 环境中受管系统的最新列表。

运行 *Db2* 存储过程

对 Db2 运行存储过程。

关于此任务

请执行下列步骤，以便对 Db2 运行存储过程：

过程

1. 以仓库用户身份连接至 Tivoli Data Warehouse 数据库：

```
connect to <Tivoli Data Warehouse database alias> user
<Tivoli Data Warehouse user id> using <password>
```

2. 运行存储过程：

```
db2 "call <Tivoli Data Warehouse schema>.kqz_populate_msn
('<three letter product code for the agent>')"
```

运行 *Oracle* 存储过程

在 Oracle 上运行存储过程。

关于此任务

请执行下列步骤，以便对 Oracle 运行存储过程：

过程

1. 启动 sqlplus：

```
sqlplus <Tivoli Data Warehouse user id>/<password>@
<Oracle SID>
```

2. 运行存储过程：

```
execute kqz_populate_msn('<three letter product code for the agent>');
```

在 *SQL Server 2005* 和 *2008* 上运行存储过程

在 SQL Server 上运行存储过程。

关于此任务

请执行下列步骤，以便对 SQL Server 2005 和 2008 运行存储过程：

过程

运行存储过程：

```
osql -S <server> -U <Tivoli Data Warehouse id> -P
<Tivoli Data Warehouse password> -d
<Tivoli Data Warehouse database name> -Q "EXEC
[<Tivoli Data Warehouse schema>].[kqz_populate_msn]
@pv_productcode = N'<three letter product code>'"
```

从 Tivoli Common Reporting 中导出报告和数据模型

从 Tivoli Common Reporting 中导出报告和数据模型。

过程

1. 登录到 Tivoli Common Reporting。
2. 转至“公用文件夹”，在导航面板中的**报告**下，选择 **Common Reporting**。
3. 在“使用报告”部分中，单击**启动**菜单并选择 **IBM Cognos 管理**。
4. 单击**配置**选项卡。
5. 单击**内容管理**。

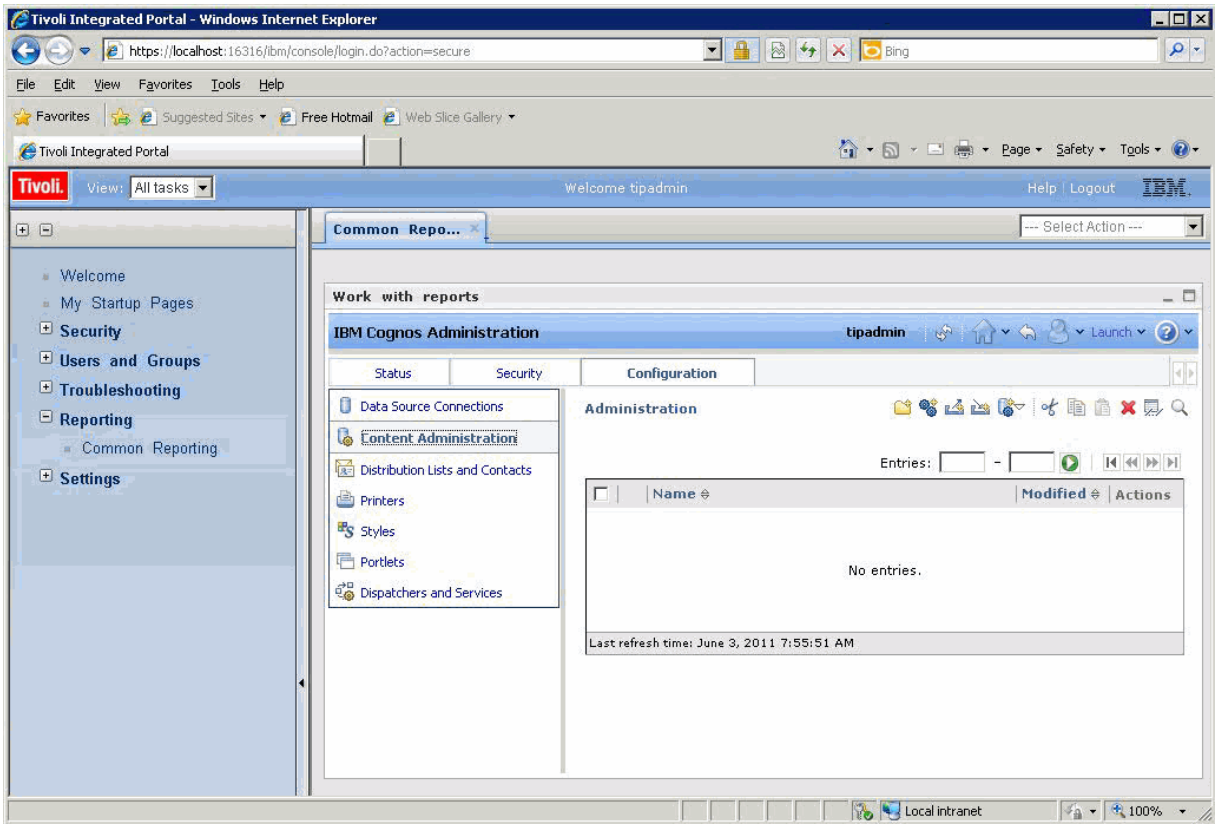


图 *85.* “内容管理”选项卡

6. 单击**新建导出**图标以导出新包。
7. 对这个包进行命名。（可选）您可以添加屏幕提示和描述。
8. 选择**选择公用文件夹和目录内容**。
9. 在“公用文件夹”对话框中，单击**添加**链接。
10. 将代理程序包移到**选定的条目**。
11. 在向导的最后一页上，选择**仅保存**。在向导完成时，此报告包会列示在“内容管理”选项卡上。
12. 在“内容管理”选项卡上，单击绿色箭头（运行）以创建压缩 `.zip` 文件。

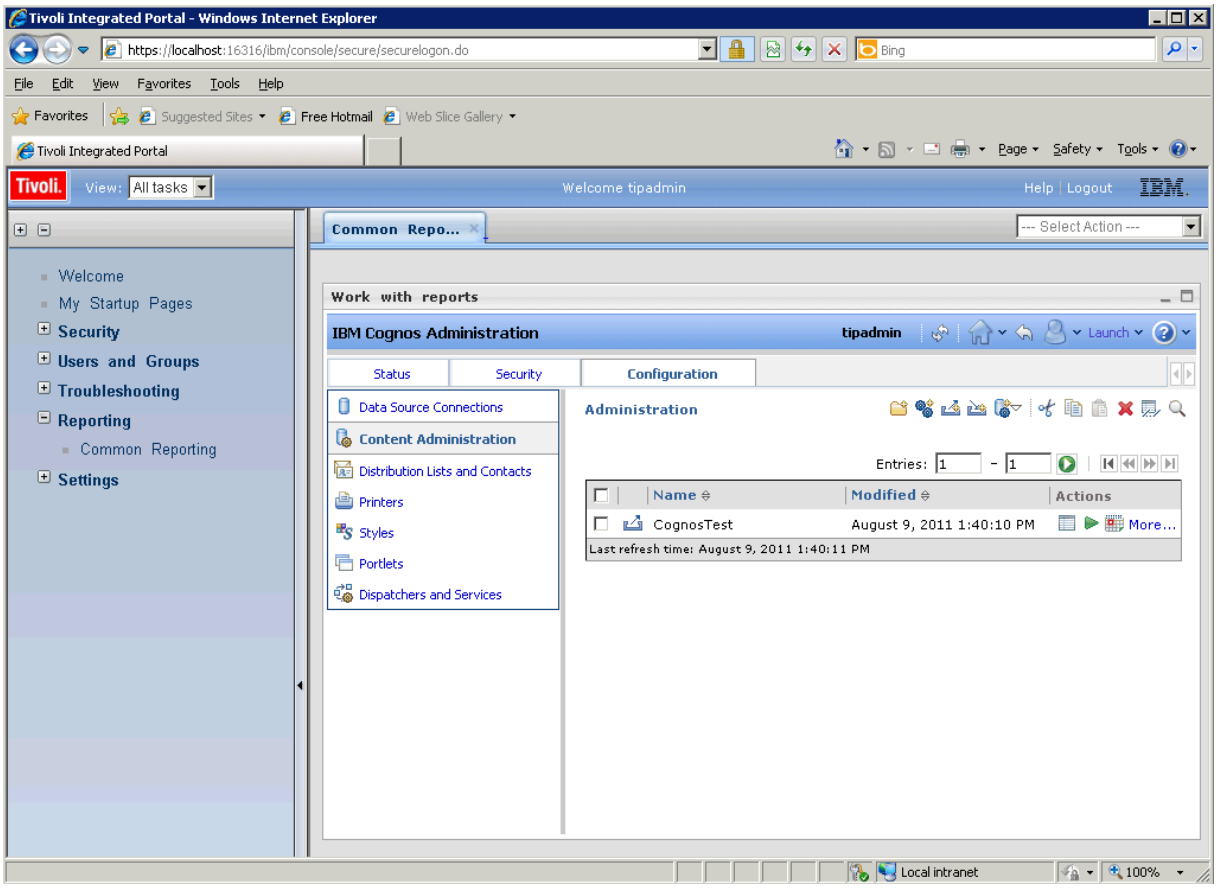


图 *86.* 列出代理程序包的“内容管理”选项卡

结果

导出过程创建的压缩 `.zip` 文件位于部署目录中。

- Tivoli Common Reporting V1.3 的目录路径如下所示：

```
C:\IBM\tivoli\tip\products\tcr\Cognos\c8\deployment
```

- Tivoli Common Reporting V2.1 或更高版本的目录路径如下所示：

```
C:\IBM\tivoli\tipv2Components\TCRComponent\cognos\deployment
```

下一步做什么

有关导出报告的更多信息，请参阅 *Tivoli Common Reporting User's Guide* 中的 Exporting Cognos report packages。

将报告导入到 Agent Builder 中

从 Tivoli Common Reporting 中导出报告包之后，可以将其导入到 Agent Builder 项目中。然后，可以在代理程序安装映像中包括报告包。

过程

1. 在 Agent Builder 中，右键单击代理程序项目。
2. 选择 **IBM** > **导入报告包**。
3. 在“**导入报告包**”窗口中，选择创建报告包时所要基于的**数据库类型**。
4. 输入报告包的标准路径，或者单击**浏览**以选择报告包。
5. 单击**确定**。
6. 报告包现在显示在代理程序项目中的 `reports/dbtype` 目录下。

注: 如果创建特定于数据库的报告包，必须将每个包导入到 Agent Builder 中。

将代理程序包中的报告安装到 Tivoli Common Reporting 中
将代理程序中的报告包导入到 Tivoli Common Reporting 中

过程

1. 按照向导中的步骤，从代理程序映像导入新的包。
   在代理程序映像中，报告位于：`reports/dbType/Kxx/reports/cognos_reports/itmkxx/packages`
2. 将报告压缩 `zip` 文件复制到 Tivoli Common Reporting 部署目录。
   - Tivoli Common Reporting V1.3 的目录路径为：`C:\IBM\tivoli\tip\products\tcr\Cognos\c8\deployment`
   - Tivoli Common Reporting V2.1 或更高版本的目录路径为：`C:\IBM\tivoli\tipv2Components\TCRComponent\cognos\deployment`
3. 登录到 Tivoli Common Reporting。
4. 转至“公用文件夹”，在导航面板中的**报告**下，选择 **Common Reporting**。
5. 在“使用报告”部分中，单击**启动**菜单并选择 **IBM Cognos 管理**。
6. 转至**配置**选项卡并打开**内容管理**部分。
7. 单击**新建导入**以创建包导入。
8. 选择代理程序的报告包。
9. 选择要导入的公共文件夹。
10. 选择 Save。
11. 单击绿色（运行）箭头以导入。

结果

有关更多信息，请参阅《Tivoli Common Reporting 用户指南》中的『登录报告界面』。

ICU 正则表达式

ICU 正则表达式实施细节的描述。

此引用内容摘自 *ICU User Guide*。此内容描述 ICU 正则表达式实施细节。如果您在使用 Agent Builder 正则表达式功能，那么此信息很重要，因为不同编程语言实施正则表达式的方式略有不同。

表 *303.* 正则表达式元字符

字符	描述
\a	与响铃符 \u0007 匹配
\A	在输入内容的开头处匹配。与 ^ 的不同之处是 \A 不在输入内容中的换行后匹配。
\b，在 [Set] 外	如果当前位置是单词边界就匹配。边界在单词 (\w) 和非单词 (\W) 字符间的过渡处出现，忽略组合标记。有关单词边界的更多信息，请参阅“ICU 边界分析”。
\b，在 [Set] 内	与退格符 \u0008 匹配。
\B	如果当前位置不是单词边界就匹配。
\cX	与 Ctrl-X 字符匹配。

表 303. 正则表达式元字符 (续)

字符	描述
\d	与具有 Unicode 常规类别 Nd（N 表示数字，d 表示十进制数字）的任何字符匹配。
\D	与任何非十进制数字的字符匹配。
\e	与转义符 \u001B 匹配。
\E	终止由 \Q ... \E 引起来的序列。
\f	与换页符 \u000C 匹配。
\G	如果当前位置是先前匹配的末尾就匹配。
\n	与换行符 \u000A 匹配。
\N{UNICODE CHARACTER NAME}	与指定字符匹配。
\p{UNICODE PROPERTY NAME}	与任何具有指定 Unicode 属性的字符匹配。
\P{UNICODE PROPERTY NAME}	与不具有指定 Unicode 属性的任何字符匹配。
\Q	在 \E 之前的所有字符两边加引号。
\r	与回车符 \u000D 匹配。
\s	与空格字符匹配。空格定义为 [\t\n\f\r \p{Z}]。
\S	与非空格字符匹配。
\t	与横向制表符 \u0009 匹配。
\uhhhh	与具有十六进制值 hhhh 的字符匹配。
\Uhhhhhhhh	与具有十六进制值 hhhhhhhh 的字符匹配。必须正好提供八个十六进制数字，即使最大的 Unicode 代码点为 \U0010ffff。
\w	与单词字符匹配。单词字符为 [\p{Ll}\p{Lu} \p{Lt}\p{Lo}\p{Nd}]。
\W	与非单词字符匹配。
\x{hhhh}	与具有十六进制值 hhhh 的字符匹配。可以提供 1 到 6 个十六进制数字。
\xhh	与具有 2 位数十六进制值 hh 的字符匹配。
\X	与语素簇匹配。
\Z\	如果当前位置在输入的结尾处、而在最后一行的终止符（如果存在）之前就匹配。
\z	如果当前位置在输入内容结尾处就匹配。
\n	反向引用。与和第 n 个捕获组匹配的任何对象匹配。n 必须为 > 1 且 < 模式中捕获组总数的数字。 **注:** ICU 正则表达式中不支持八进制转义符（如 \012）。
[pattern]	与集内的任意一个字符匹配。有关模式中可能出现的内容的完整描述，请参阅 UnicodeSet

表 *303.* 正则表达式元字符 *(续)*

字符	**描述**
.	与任意字符匹配。
^	在行开头处匹配。
$	在行结尾处匹配。
\	在以下字符两边加引号。必须加引号才能被视为字面值的字符有：`* ? + [ ( ) { } ^ $ | \ . /`

表 *304.* 正则表达式运算符

运算符	**描述**
|	交替。A|B 与 A 或 B 匹配。
*	匹配 0 次或 0 次以上。匹配次数尽可能地多。
+	匹配 1 次或 1 次以上。匹配次数尽可能地多。
?	匹配 0 次或 1 次。首选 1 次。
{n}	精确匹配 n 次
{n,}	至少匹配 n 次。匹配次数尽可能地多。
{n,m}	匹配 n 到 m 次。匹配次数尽可能地多，但不超过 m 次。
*?	匹配 0 次或 0 次以上。匹配次数尽可能地少。
+?	匹配 1 次或 1 次以上。匹配次数尽可能地少。
??	匹配 0 次或 1 次。首选 0 次。
{n}?	精确匹配 n 次
{n,}?	至少匹配 n 次，但是不超过整体模式匹配所需的次数
{n,m}?	匹配 n 到 m 次。匹配次数尽可能地少，但不少于 n 次。
*+	匹配 0 次或 0 次以上。首次遇到时，匹配的次数尽可能地多，即使总体匹配失败，重试次数也不减少（占有匹配）
++	匹配 1 次或 1 次以上。占有匹配。
?+	匹配 0 次或 1 次。占有匹配。
{n}+	精确匹配 n 次
{n,}+	至少匹配 n 次。占有匹配。
{n,m}+	匹配 n 到 m 次。占有匹配。
(...)	捕获括号。与括起的子表达式匹配的输入范围在该匹配后可用。
(?: ...)	非捕获括号。将包含的模式进行分组，但是不提供对匹配文本的捕获。比捕获括号的效率更高。

表 *304.* 正则表达式运算符 *(续)*	
运算符	**描述**
(?> ...)	原子匹配括号。与括起子表达式的首次匹配是唯一尝试的一次匹配。如果此次匹配未导致整体模式匹配，备份与“(?>”前面的位置匹配的搜索。
(?# ...)	自由格式注释 (?# comment)。
(?= ...)	先行断言。如果括起的模式在当前输入位置匹配而没有提前输入位置，那么为 True。
(?! ...)	否定先行断言。如果括起的模式在当前输入位置不匹配，那么为 True。不提前输入位置。
(?<= ...)	后顾断言。如果括起的模式与当前输入位置前面的文本匹配，那么为 True。匹配的最后一个字符正好是位于当前位置前面的输入字符。不变更输入位置。后顾模式匹配的可能字符串长度必须受限（无 * 或 + 运算符）。
(?<!...)	否定后顾断言。如果括起的模式与当前输入位置前面的文本不匹配，那么为 True。匹配的最后一个字符正好是位于当前位置前面的输入字符。不变更输入位置。后顾模式匹配的可能字符串长度必须受限（无 * 或 + 运算符）。
(?ismx-ismx: ...)	标志设置。在启用或禁用指定标志的情况下对括起的表达式求值。
(?ismx-ismx)	标志设置。更改标志设置。更改应用于设置后面的模式部分。例如，(?i) 更改为不区分大小写的匹配。

替换文本

“查找与替换”操作的替换文本可以包含对来自查找的捕获组文本的引用。引用的形式为 $n，其中 n 是捕获组数量。

表 *305.* 替换文本字符	
字符	**描述**
$n	用位置捕获组 n 的文本替换 $n。n 必须 >= 0，并且不大于捕获组数量。$ 后不跟数字不表示任何特殊意义，在替换文本中将按原样显示为 $。
\	将此字符视为字面值，不表示任何特殊含义。替换文本中的反斜杠转义仅针对“$”和“\”是必需的，但可以用于任何其他字符，而不会产生反作用。
$@n	用捕获组 n 的文本替换与捕获组 n 匹配的正则表达式。n 必须 >= 0，但不大于捕获组数量。$@ 后不跟数字不表示任何特殊意义，在替换文本中将按原样显示为 $@。
$#n	用匹配捕获组 n 的文本替换 $#n。n 必须 >= 0，并且不得大于匹配的捕获组数量。$# 后不跟数字不表示任何特殊意义，在替换文本中将按原样显示为 $#。

标志选项

以下标志用于控制正则表达式匹配的各个方面。可以在将表达式编译为 RegexPattern 对象时指定标志值。或者，可以使用 (?ismx-ismx) 模式选项在模式自身内部指定。

表 *306*. 标志选项

标志（模式）	标志（API 常量）	描述
i	UREGEX_CASE_INSENSITIVE	如果设置了此标志，将以不区分大小写的方式进行匹配。
x	UREGEX_COMMENTS	如果设置了此标志，那么可以在模式中使用空格和 #comments。
s	UREGEX_DOTALL	如果设置了此标志，那么模式中的“.”将与输入文本中的行终止符匹配。缺省情况下不匹配。文本中的回车符/换行对等效于单个行终止符，并且与 RE 模式中的单个“.”匹配
m	UREGEX_MULTILINE	控制模式中“^”和“$”的行为。缺省情况下，这些模式分别仅在输入文本的开头和结尾处匹配。如果设置了此标志，那么“^”和“$”还在输入文本内部每一行的开头和结尾处匹配。

创建非代理程序文件捆绑包

您能够创建可以放入 Tivoli Monitoring 库的文件捆绑包。然后，在您的环境中，可以将这些文件捆绑包部署到目标系统。

关于此任务

借助此功能，您可以采用远程方式来配置没有远程配置选项的产品。要使用此功能，请将进行了预填充的配置文件放入库中，并将其发送到所需的系统。

过程

1. 从 Agent Builder 中，选择**文件** > **新建** > **其他**。
2. 在 **Agent Builder** 下面，选择**非代理程序远程部署捆绑包**。
3. 单击**下一步**。
4. 在**项目名称**字段中，输入项目名称。
5. 单击**下一步**。
6. 在“**远程部署捆绑包信息**”窗口中填写信息：
   a) 在**捆绑包标识**字段中键入标识，此标识为 3 - 31 个字符的唯一字母数字字符串。此字符串可以包含连字符。此字符串必须以字母开头，但不能以 K 或连字符开头。
   b) 在**捆绑包描述**字段中，键入捆绑包描述。
   c) 在**版本**字段中，键入 VVRRMMFFF 格式的捆绑包版本。其中，vv 为版本号，rr 为发行版号，mm 为修改号（修订包号），而 fff 为临时修订号。
7. 在**操作系统**区域中，选择此捆绑包可以部署到的操作系统。
8. 单击**完成**以便在工作空间中创建项目，并打开**远程部署捆绑包编辑器**。

远程部署捆绑包编辑器

远程部署捆绑包编辑器用于生成帮助部署文件捆绑包的命令。

远程部署捆绑包编辑器提供有关项目捆绑包的信息。

捆绑包标识信息部分包含以下信息:

Bundle identifier
捆绑包的唯一标识

Bundle description
捆绑包的描述

Bundle version
捆绑包的版本

Build
捆绑包的构建标识。此处输入构建号。如果未指定任何构建号，那么将根据生成捆绑包的日期和时间生成一个编号。

针对捆绑包中的文件创建复制命令复选框
单击该复选框以生成在部署捆绑包时运行的一组缺省复制命令。这些文件将复制到**复制位置**文本框中指定的位置。缺省位置是 *INSTALLDIR*。在命令行部署中进行以下设置来指定此远程部署变量: KDY.*INSTALLDIR*=...

操作系统部分显示可以将捆绑包部署到的操作系统。

命令部分显示部署捆绑包时要运行的命令。

必备捆绑包部分显示为使此捆绑包工作而必须存在的捆绑包。

使用远程部署捆绑包编辑器来选择一组缺省复制命令，以便将捆绑包中的文件复制到指定位置。如果选择了此选项，那么会为捆绑包项目中的各个文件生成一个复制命令。缺省复制位置为 *INSTALLDIR*。如果未在部署命令行中设置特殊远程部署变量，那么缺省为 *CANDLEHOME*。要更改 *INSTALLDIR* 指定的位置，请在运行 **addSystem** 命令时指定 **KDY.INSTALLDIR** 属性。

捆绑包项目中指定的相同目录结构将复制到 *INSTALLDIR*。例如，如果捆绑包项目中名为 config 的文件夹有一个名为 myprod.config 的文件，那么生成的复制命令会在部署捆绑包时将该文件复制到 *INSTALLDIR*/config/myprod.config。

向捆绑包添加命令

您可以指定更多要在部署期间运行的命令。

关于此任务

通过使用**远程部署捆绑包编辑器**，您可以指定要在部署期间运行的其他命令。

过程

1. 要指定部署期间运行的其他命令，请在**远程部署捆绑包编辑器**的**命令**部分中单击**添加**。
2. 在“**命令**”窗口中，选择命令类型**预安装**、**安装**、**后安装**或**卸载**，然后指定要运行的命令。
   必须指定要运行的命令的标准路径。方便起见，远程部署提供了一组预定义变量。要在命令中引用变量，请在变量两侧加竖线，例如 |DEPLOYDIR|。有关命令的预定义变量的更多信息，请参阅第 1264 页的表 307。

表 *307.* 命令的预定义变量

变量	描述
DEPLOYDIR	这是端点上的一个临时目录，在部署期间，捆绑包存储在此目录中。例如，如果要运行捆绑包中包含的 myscript.sh 脚本，可以指定以下命令: |DEPLOYDIR|/myscript.sh

表 *307.* 命令的预定义变量 *(续)*	
变量	**描述**
INSTALLDIR	*CANDLEHOME*，或者 *KDY.INSTALLDIR* 的值（如果已在 **addSystem** 命令中指定）。
CANDLEHOME	Tivoli Monitoring 安装目录。

3. 最后，选择要在其中运行命令的**系统**。

向捆绑包添加必备项

使用**远程部署捆绑包编辑器**可以指定捆绑包的必备项。

过程

1. 要添加必备项，请在**远程部署捆绑包编辑器**的“**捆绑包信息**”页面的**必备捆绑包**部分中单击**添加**。
2. 在“**新建必备项**”窗口中，输入此捆绑包所依赖的捆绑包标识以及所需的最低版本。
3. 选择需要此必备项的操作系统。
4. 单击**确定**以完成操作并退出。

向捆绑包添加文件

通过使用**远程部署捆绑包编辑器**，可以向文件捆绑包添加文件。

过程

1. 要将文件添加至远程部署捆绑包，请执行下列其中一个过程：
   - 在捆绑包编辑器中，单击**将文件添加至捆绑包**。
   - 在导航器树中，右键单击项目，然后单击 **IBM Tivoli Monitoring 远程部署 > 将文件添加至捆绑包**

   这两项操作都将显示“**导入捆绑包文件**”窗口：
2. 在**文件信息**区域中，指定个别文件或包含文件的目录。
3. 单击**完成**。
   所指定的文件或目录将复制到项目目录中。您构建远程部署捆绑包时，项目中的目录结构保持不变。如果要让 Agent Builder 生成缺省的复制命令，请确保文件所处的目录结构正确，可以进行部署。

生成捆绑包

使用 Agent Builder 为代理程序远程部署生成捆绑包。

过程

1. 要生成远程部署捆绑包，请使用以下其中一个过程来显示“**生成最终远程部署捆绑包**”窗口。
   - 在**远程部署捆绑包编辑器**中，单击**生成最终远程部署捆绑包**。
   - 在导航器树中，右键单击项目，然后单击 **IBM Tivoli Monitoring 远程部署 > 生成远程部署捆绑包**。
2. 现在，您可以用以下两种方法生成捆绑包：
   - 如果运行 Agent Builder 的系统上存在 Tivoli Enterprise Monitoring Server，请单击**将远程部署捆绑包安装到本地 TEMS 库中**。

     Agent Builder 尝试确定 Tivoli Monitoring 安装位置并将其输入到**目录**字段中。如果未设置 *CANDLE_HOME*，将使用缺省位置 `C:\IBM\ITM` 或 `/opt/ibm/itm`。请确保安装位置正确，然后再继续操作。

     必须提供 Tivoli Enterprise Monitoring Server 登录信息才能安装捆绑包。
   - 要将捆绑包生成到系统上的目录中，请单击**在本地目录中生成远程部署捆绑包**。

此过程完成之后，必须将此目录传输到 Tivoli Enterprise Monitoring Server 系统，并使用 `tacmd addbundles` 命令将此捆绑包添加到库。

下一步做什么

在部署捆绑包时，必须使用 `tacmd addSystem` 命令。例如：

```
tacmd addsystem -t MONITORINGCOLLECTION -n Primary:ITMAGT:NT
```

其中 `-t` (type) 是下面的 `tacmd viewDepot` 命令返回的产品代码：

```
>tacmd viewDepot
Product Code : MONITORINGCOLLECTION
Version : 010000003
Description : MonitoringCollectionScripts
Host Type : WINNT
Host Version : WINNT
Prerequisites:
```

注: 您无法从 Tivoli Enterprise Portal Desktop 或 Browser 远程部署。从 Tivoli Enterprise Portal Desktop 或 Browser 远程部署会产生 KFWITM219E 消息。

请参阅 Tivoli Monitoring 文档以了解更多详细信息。

为 Tivoli Netcool/OMNIbus 探测器创建可部署捆绑包

您可以使用 Agent Builder 来创建包和配置捆绑包，这些包及配置捆绑包可用来将 Tivoli Netcool/OMNIbus 探测器部署到远程计算机。

关于此任务

为了支持探测器远程部署，在部署探测器之前，您还可以创建可部署至远程计算机的 Tivoli Netcool/OMNIbus 捆绑包。

过程

1. 从 Agent Builder 中，选择**文件** > **新建** > **其他**。
2. 在 **IBM Tivoli OMNIbus 向导**下面，选择**软件包捆绑**。
3. 单击**下一步**。

下一步做什么

接下来，使用 **OMNIbus 安装捆绑包**向导来创建捆绑包。有关使用此向导的信息，请参阅 Tivoli Netcool/OMNIbus 文档。

动态文件名支持

使用动态文件名支持来指定文件名模式而非实际文件名。

某些应用程序可以创建遵从更改的输出文件名。名称根据当前日期、月份、年份或包含递增序号的文件名等特定条件进行更改。在这些情况下，您可以指定文件名模式而非实际文件名。在指定文件名模式时，有两种可以识别的模式格式：

- 正则表达式（首选）。
- IBM Tivoli Universal Agent 动态文件名语法（不推荐）。

正则表达式文件名模式

要指定文件名模式，您可以根据（第 1259 页的『ICU 正则表达式』）中所记录的 International Components for Unicode (ICU) 语法来使用正则表达式。要使用此功能，您必须在“**高级日志文件属性组信**

息”页面上选中**文件名与正则表达式匹配**复选框。在指定正则表达式模式时，还必须从“**高级日志文件属性组信息**”页面上的**当多个文件匹配时**列表中选择选项，以指定用于选择最新匹配文件的准则。

注: 正则表达式是指定文件名模式的首选方法。

有关如何配置高级日志文件属性组属性的更多信息，请参阅（第 1060 页的『监视日志文件』）中的步骤 (第 1061 页的『6』)。例如，如果指定了文件名模式:

```
d:\program files\logs\tivoli.*
```

此模式将搜索 d:\program files\logs 目录中以 tivoli 开头的文件名。只能对文件名部分而不是路径名指定正则表达式。

动态文件名语法

如果使用动态文件名语法，那么一次只能监视一个文件。文件数据提供程序检查所指定的路径位置中的所有文件，查找与所定义的模式相匹配的文件。文件数据提供程序始终监视最新的匹配文件，判断依据是具有最大数字或日期时间值的匹配文件名。要监视的相应文件由文件名确定，而不是由文件创建或其他条件确定。

可以指定文件名的模式，其可以具有任意数量的组成部分。例如，Log{###} 与只含有一个部分的文件名（例如 Log010 或 Log456）匹配。在含有多个部分的文件名中，可以在文件名的任何部分或多个部分中指定模式字符。例如，aaa.bbb{???}.ccc 是有效模式，aaa.bbb{???}.ccc{###} 也有效。

注: 指定文件名模式的首选方法是正则表达式，而不是动态文件名语法，有关正则表达式的更多信息，请参阅第 1266 页的『正则表达式文件名模式』

以下示例说明了文件名模式规范:

{##########}.abc
与长度为 8 且文件扩展名为 *.abc* 的数字文件名相匹配，例如 10252006.abc 或 10262006.abc。将监视文件 10262006.abc，因为 10262006 大于 10252006。

{##########}.*
与长度为 8 的数字文件名相匹配，并忽略文件扩展名。示例包括 20061025.log、20061101.log 和 10252006.abc。将监视文件 20061101.log，因为 20061101 是最大数字。

{#######??}.abc
与长度为 8 且文件扩展名为 .abc 的数字文件名相匹配，并忽略名称部分的最后两位。示例包括 02110199.abc、02110200.abc 和 021101AZ.abc。将监视文件 02110200.abc，因为 021102 是最大数字。

Console.{#######}
与名称部分包含 *Console* 且扩展名为六位数字的文件名相匹配。示例包括 Console.000133、Console.000201 和 Console.000134。将监视文件 Console.000201。

IN{#######}.log
与以 IN 开头，后跟 6 个数字，且文件扩展名为 .log 的文件名相匹配。示例包括 IN021001.log、IN021002.log 和 IN021004.log。将监视文件 IN021004.log。

PS{###}FTP.txt
与以 PS 开头，后跟 3 个数字，然后再跟 FTP，且扩展名为 .txt 的文件名相匹配。示例包括 PS001FTP.txt、PS005FTP.txt 和 PS010FTP.txt。将监视文件 PS010FTP.txt。

请遵循以下准则来确立文件名模式:

- 使用花括号 {} 括起文件名中的模式字符。花括号内存在模式字符表示使用了文件名模式。
- 使用星号 (*) 作为通配符以忽略文件扩展名或文件名中的任何尾随字符。例如，Myapp{###}.log* 指定以 Myapp 开头，后跟三个数字，然后再跟 .log（无论后面再跟什么字符）的任何文件名都是匹配项。

  星号必须在花括号（{ }）后面指定，而不能用于文件名开头。在文件扩展名中使用星号时，星号必须独立使用。

  通配符（*）正确用法示例:

  err{??}.*

error{$}.*

通配符 (*) 错误用法示例:

error.20*
星号 (*) 之前没有花括号。

error*.{###}
星号不能在文件名结尾使用。

error.*
星号 (*) 之前没有花括号。

- 如果定义了特定的文件扩展名，那么只考虑具有相同扩展名的文件。
- 使用井号来表示文件名的每个数字元素。
- 使用问号来排除命名约定中不作为搜索条件来决定合适文件名的各个元素。
- 使用美元符号 ($) 来表示任意字符或没有字符。例如，如果要匹配名为 Log 和 LogA 的两个文件，请指定 Log{$}。美元符号有几个使用限制。在使用一个或多个美元符号作为文件名前缀时（例如 {$$$$$$}_abc.log），美元符号的数量必须与文件名中该位置处的字符数完全相等。另外，也不能在文件名模式中的多个位置指定美元符号，例如 {$$$}b{$$$}.log 将不能匹配 abc.log。由于存在这些美元符号限制，因此如果文件名中的字符数量不确定，请使用正则表达式文件名模式。
- 花括号括起的井号和问号的总数很重要。它必须与文件名的部分完全匹配。例如，模式 AA{#####} 指示文件数据提供程序查找类似 AA0001 的文件名。将不考虑诸如 AA001 或 AA00001 之类的文件名。
- 精确文件名模式、常量和数字部分必须与文件名完全匹配。例如，模式 AA{###} 指示文件数据提供程序检查文件 AA101。将不考虑诸如 XAA101、AA222X 和 AA55555 之类的文件名。
- 使用保留的模式字符串 {TIVOLILOGTIME} 来替换 Tivoli Monitoring 代理程序或服务器日志文件中的十六进制时间戳记和文件序列号。在对 Tivoli Monitoring 组件实施自监视时，此模式字符串很有用。例如，如果要监视 /opt/IBM/ITM/logs 目录中的最新 监视服务器 日志，可以指定文件名模式:

```
/opt/IBM/ITM/logs/Host1_ms_{TIVOLILOGTIME}.log
```

如果 /logs 目录中存在 Host1_ms_452053c0-01.log、Host1_ms_451f11f4-01.log、Host1_ms_45205946-01.log 和 Host1_ms_451f11f4-02.log，将选择监视 Host1_ms_45205946-01.log 文件。

要精确指定由各个日期部分（年、月和日）组成的文件名，使用大写字母 Y、M 和 D。必须在括号内指定这些字母；否则，它们将被视为文件名中的文字字符。

请参阅以下示例:

{YYYYMMDD}.log
指定诸如 20060930.log 或 20061015.log 之类的文件名。

{MMDDYY}.log
指定诸如 101106.log 或 110106.log 之类的文件名。

{DDMMYYYY}.log
指定诸如 01092006.log 或 15082006.log 之类的文件名。

{DDMMMYY}.log
指定诸如 24Jan07 或 13Sep06 之类的文件名。

{MM-DD-YY}.log
指定诸如 11-02-06 或 04-29-07 之类的文件名。日期字段中的 (-) 分隔符会被忽略，因此不需要使用问号模式字符跳过此字符。

MY{YYDDD}.log
指定诸如 MY06202.log、MY06010.log 或 MY04350.log 之类的文件名。

存在一些复杂的情况，例如，较长的文件名中嵌入了日期字段，那么先前示例中的日期模式不足以满足需求。对于复杂情况，请创建混合使用井号和问号的模式，并且这些模式仍执行选择最新文件进行监视的数字比较。例如，模式 ABC{?####?##?##?##?##?##?}XYZ.TXT 可用于 ABC 2006-04-20 11_22_33

XYZ.TXT 之类的文件名。在此示例中，您仅对 #- 标记的数字感兴趣，问号充当用于忽略文件名中其他字符的占位符。

文件数据提供程序会定期检查目标路径位置中与定义的文件模式匹配的新文件。当检测到与模式匹配的较新文件时，文件数据提供程序自动将应用程序监视切换至该新文件。在以下情况下，文件数据提供程序会搜索最佳匹配文件：

· 文件数据提供程序第一次启动时。

· 由于可能发生重命名或删除，当前受监视的文件不再存在。

· 现有文件内容由于可能发生重写而已更改。

· 检查时间间隔已到期。缺省时间间隔为 10 分钟。您可以通过指定环境变量来增大或减小时间间隔

```
KUMP_DP_FILE_SWITCH_CHECK_INTERVAL=number-of-seconds
```

SNMP 陷阱配置

这是对配置文件的描述，SNMP 数据提供程序使用此配置文件以更容易阅读的形式呈现陷阱信息。该文件还用于向陷阱分配类别、严重性、状态和源标识。

此外还包含有关如何修改缺省文件或替换您自己的配置文件的指示信息。

SNMP 陷阱配置文件 trapcnfg

在启动时，SNMP 数据提供程序读取名为 trapcnfg 的配置文件。该文件的一个用途是将 SNMP 陷阱信息转换为更容易阅读的形式。另一个用途是向特定陷阱分配类别、严重性、状态和源标识，因为 SNMP 未定义这些类别。

您可以通过添加新的陷阱或企业定义或更改现有的陷阱或企业定义来修改 trapcnfg 文件以满足特定于站点的需要。还可以使用自己的配置文件。

使用 HP OpenView trapd.conf 文件

trapcnfg 文件与 HP OpenView Network Node Manager 陷阱配置文件 trapd.conf 在格式上相似，但并不完全相同。您可以在必要的情况下复制 OpenView 文件并复用许多定义语句。

记录类型

trapcnfg 包含三种类型的记录或记录块：

注释
注释记录以数字符号 (#) 开头。

企业定义
企业定义由两个空白定界标记组成，其中第一个标记是名称；第二个标记是两边为大括号 ({ }) 的对象标识 (OID)。

陷阱定义
陷阱定义由 8 个空白定界标记组成。陷阱定义是块记录，因为每个定义可能由多条记录构成。

第一种类型是自我解释型。（第 1270 页的图 87）显示了第二种和第三种类型的示例。

第 1270 页的图 87 中第一个示例显示的企业定义记录将企业 OID 1.3.6.1.4.1.311.1.1.3.1.1 定义为 Microsoft Windows NT。

第二个示例显示的陷阱定义记录将 trapName MSNTCOLD 定义成与企业 OID 1.3.6.1.4.1.311.1.1.3.1.1 关联的一般陷阱号 0 和特定陷阱号 0。请注意，严重性是以十进制形式表示，而类别是以文本形式表示。严重性在显示之前转换为其文本形式。类型 3 记录块中的下一条记录是 Agent Builder 不使用的简短描述。Agent Builder 使用定界符 SDESC 和 EDESC 之间括起的详细描述。

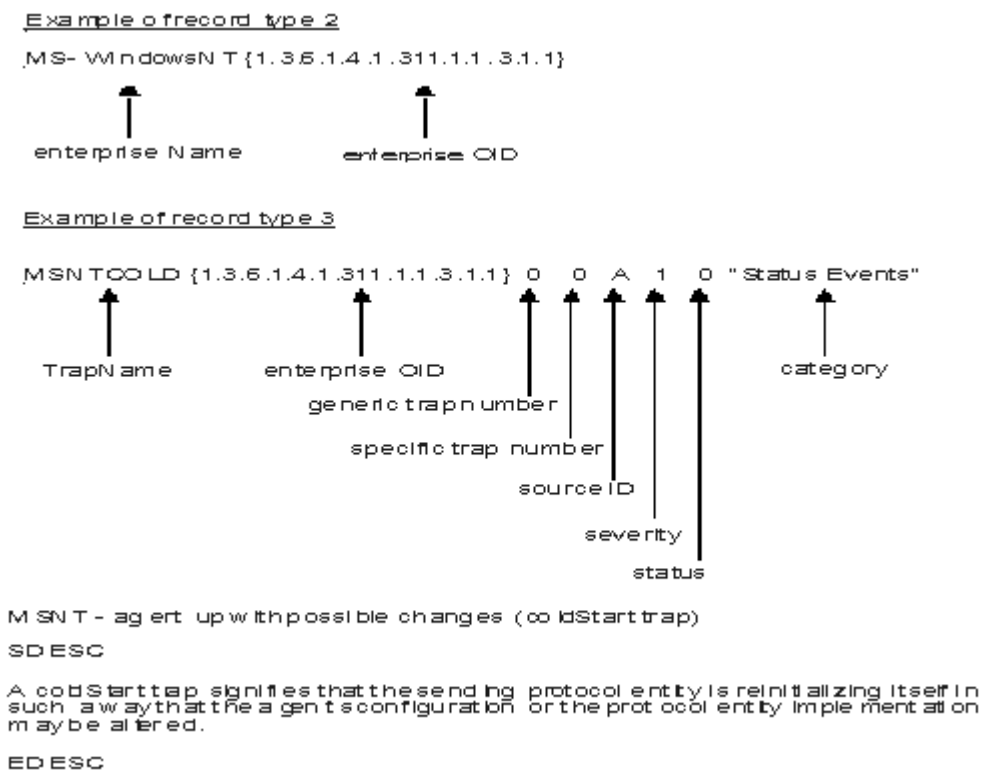


图 *87.* 第二和第三种类型的配置记录示例

trapcnfg 文件的缺省值

以下各表列出了 SNMP 数据提供程序支持的缺省值。

受支持的类别

（第 1270 页的表 308）显示了 Agent Builder 支持的类别。

表 *308. SNMP* 数据提供程序支持的类别

类别	文本说明
0	阈值事件
1	网络拓扑事件
2	错误事件
3	状态事件
4	节点配置事件
5	应用程序警报事件
6	所有类别事件
7	仅日志事件
8	映射事件
9	忽略事件

（第 1271 页的表 309）列出了 Agent Builder 支持的严重性。

表 *309. SNMP* 数据提供程序支持的严重性	
严重性	**文本说明**
0	正常
1	不确定
2	警告
3	轻微错误
4	严重
5	主要错误

受支持的状态

（第 1271 页的表 310）显示 Agent Builder 配置文件中定义的状态。

表 *310. SNMP* 数据提供程序支持的状态	
状态	**文本说明**
0	不变
1	未知
2	运行
3	不佳
4	关闭
5	未受管
6	确认
7	用户 1
8	用户 2

受支持的源标识

（第 1271 页的表 311）列出了 `trapcnfg` 支持的源标识。

表 *311. SNMP* 数据提供程序支持的源标识	
源标识	**描述**
a	应用程序
A	代理程序
C	`Xnmcollect`
d	演示
D	数据收集器
E	`Nvevents`
如果	`Ipmap`
L	`LoadMIB`
m	`Shpmon`
M	`IP` 拓扑

表 311. SNMP 数据提供程序支持的源标识 (续)	
源标识	**描述**
n	netmon 相关
N	netmon 生成的陷阱
O	OSI SA
P	非 IP 陷阱
r	Tralertd
s	Spappld
S	安全代理程序
t	Xnmtrap
T	Trapd
V	供应商相关
?	未知

“执行操作”命令参考

“执行操作”命令概述、有关“执行操作”命令的参考资料以及特殊“执行操作”命令的描述。

关于“执行操作”命令

“执行操作”命令可包括在 Agent Builder 监视代理程序中。“执行操作”命令可以从门户网站客户机运行，或者包括在情境或策略中。包括在情境中时，这些命令会在该情境变为 true 时运行。情境中的“执行操作”命令也称作反射自动化。在情境中启用“执行操作”命令后，就会自动响应系统状况。例如，可以使用“执行操作”命令来发送一条命令，以便在受管系统上重新启动进程。另外，还可以使用“执行操作”命令向手机发送短信。

高级自动化使用策略来运行操作、调度工作以及自动完成手动任务。策略由一系列的自动化步骤组成，这些步骤称为活动，它们相互连接以构成工作流程。完成一项活动之后，Tivoli Enterprise Portal 会收到返回码反馈，高级自动化逻辑会以该反馈所指定的后续活动进行响应。

基本“执行操作”命令会在操作完成后显示的消息框或日志文件中显示操作返回码。关闭此窗口后，不会提供有关此操作的更多信息。

关于“执行操作”命令的更多信息

有关使用“执行操作”命令的更多信息，请参阅《*Tivoli Enterprise Portal* 用户指南》。

要获取此监视代理程序的“执行操作”命令的列表和描述，请参阅第 1272 页的『特殊“执行操作”命令』。另请参阅该章节中每个命令的信息。

特殊“执行操作”命令

Agent Builder 监视代理程序可以识别一组“执行操作”命令，并为其执行特殊处理：

· SSHEXEC

有关创建这些命令以及将它们包括在 Agent Builder 监视代理程序项目中的更多信息，请参阅第 1149 页的『创建工作空间、“执行操作”命令和情境』。

SSHEXEC 操作

开始之前

有关“执行操作”命令的更多信息，请参阅第 1272 页的『“执行操作”命令参考』。

关于此任务

SSHEXEC 操作是针对至少有一个 SSH 脚本属性组的受监视应用程序而识别。它表示 SSHEXEC 关键字后的命令要在 SSH 目标系统上远程启动。该命令将会使用为监视 SSH 目标系统而配置的用户的凭证和特权来启动。该命令将在“受管系统名称”所表示的远程系统上运行。

过程

要将“执行操作”命令包括在情境或工作流策略中，请使用系统命令的以下语法：

```
SSHEXEC [Command]
```

例如：

```
SSHEXEC [ls &path]
```

注： 在调用“执行操作”期间，通过将“执行操作”自变量选项与 *Command* 配合使用，可以对命令或命令中的某些部分进行定制。

注： 如果 *Command* 包含多个自变量，请考虑添加方括号，以支持通过 **tacmd** 命令行界面调用“执行操作”命令。

辅助功能

辅助功能帮助那些身体残障（例如行动有障碍或视力有障碍）的用户成功地使用信息技术内容。

辅助功能

IBM Cloud Application Performance Management 基于 Web 的界面是 Cloud APM 控制台。该控制台包含以下主要的辅助功能：

- 使用户能够使用辅助技术（例如屏幕朗读器软件和数字发音合成器）来听取屏幕上显示的内容。有关将这些技术用于本产品的详细信息，请查阅辅助技术的产品文档。
- 使用户能够仅使用键盘操作特定的或等效的功能部件。
- 传达与颜色无关的所有信息。[1]

Cloud APM 控制台使用最新的 W3C 标准 WAI-ARIA 1.0 (http://www.w3.org/TR/wai-aria/)，以确保符合 US Section 508 (http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards) 和 Web Content Accessibility Guidelines (WCAG) 2.0 。要利用辅助功能，请结合使用最新发行版的屏幕朗读器与该产品支持的最新 Web 浏览器。

IBM Knowledge Center 中的 Cloud APM 控制台联机产品文档支持辅助功能选项。IBM Knowledge Center 的辅助功能在 IBM Knowledge Center 发行说明中进行了描述 .

键盘导航

本产品使用标准的导航键。

界面信息

Cloud APM 控制台 Web 用户界面不依靠级联样式表来正确呈现内容和提供适用的体验。但是，产品文档依靠级联样式表。IBM Knowledge Center 为视力低下用户提供了等效方法来使用其定制显示设置，包括高对比度模式。您可以使用设备或浏览器设置来控制字体大小。

Cloud APM 控制台 Web 用户界面包括您可以用于快速导航到应用程序中的功能性区域的 WAI-ARIA 导航标志。

Cloud APM 控制台用户界面不包含每秒闪烁 2 - 55 次的内容。

相关辅助功能选项信息

除了标准的 IBM 帮助热线和支持 Web 站点，IBM 还建立了 TTY 电话服务，以供耳聋或由严重听力障碍的客户用于获取销售和支持服务：

TTY 服务 800-IBM-3383 (800-426-3383)（北美地区）

IBM 与辅助功能

有关 IBM 对辅助功能选项所作承诺的更多信息，请参阅 IBM Accessibility (www.ibm.com/able)。

[1] 但 Performance Management 控制台的某些**代理程序配置**页面除外。

© Copyright IBM Corp. 2014, 2019

声明

本信息是为在美国国内供应的产品和服务而编写的。IBM 可能提供本资料的其他语言版本。但是，您可能必须具有该语言版本的产品副本或产品版本才能对其进行访问。

IBM 可能在其他国家或地区不提供本文档中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并不意味着授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
美国

有关双字节字符集 (DBCS) 信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

International Business Machines Corporation“按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些管辖区域在某些交易中不允许免除明示或默示的保证，因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本出版物中描述的产品进行改进和/或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i) 允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及 (ii) 允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
美国

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处讨论的性能数据是在特定操作条件下得出的。实际结果可能会有差异。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

© Copyright IBM Corp. 2014, 2019

本信息仅用于规划的目的。在所描述的产品上市之前，此处的信息会有更改。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，示例中可能会包括个人、公司、品牌和产品的名称。所有这些名字都是虚构的，若与现实人员或企业相似，纯属巧合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口 (API) 进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。这些实例程序“按现状”提供，不附有任何种类的保证。对于因使用样本程序所引起的任何损害，IBM 概不负责。

凡这些实例程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：
©（贵公司的名称）（年）。此部分代码是根据 IBM Corp. 公司的样本程序衍生而来。© Copyright IBM Corp. 2014, 2015.

商标

IBM、IBM 徽标和 ibm.com 是 International Business Machines Corp., 在全球许多管辖区域的商标或注册商标。其他产品和服务名称可能是 IBM 或其他公司的商标。IBM 商标的最新列表可从 Web 上的“Copyright and trademark information”处获得，网址为：www.ibm.com/legal/copytrade.shtml。

Linux 是 Linus Torvalds 在美国和/或其他国家或地区的商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和/或其他国家或地区的商标。

UNIX 是 The Open Group 在美国和/或其他国家或地区的注册商标。



Java 和所有基于 Java 的商标和徽标是 Oracle 和/或其分公司的商标或注册商标。

产品文档的条款和条件

根据下列条款和条件授予对这些出版物的使用许可权。

适用性

这些条款和条件是对 IBM Web 站点的任何使用条款的补充。

个人使用

您可以为了个人使用而非商业性使用复制这些出版物，但前提是保留所有专有权声明。未经 IBM 的明确许可，您不得分发、显示这些出版物或其中部分出版物，也不得制作其演绎作品。

商业使用

您仅可在贵公司内部复制、分发和显示这些出版物，但前提是保留所有专有权声明。未经 IBM 明确同意，您不可以制作这些出版物的演绎作品，或者在您的企业外部复制、分发或展示这些出版物或其中的任何部分。

权利

除了本许可权中明确授权，未授予对这些出版物或此处包含的任何信息、数据、软件和其他知识产权的任何其他许可权、许可证或权利。

只要 IBM 认为这些出版物的使用会损害其利益或者 IBM 判定未正确遵守上述指示信息，IBM 将有权撤销本文授予的许可权。

只有您完全遵循所有适用的法律和法规，包括所有的美国出口法律和法规，您才可以下载、出口或再出口该信息。

IBM 对于这些出版物的内容不作任何保证。这些出版物“按现状”提供，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关适销性、非侵权和适用于某种特定用途的保证。

IBM 网上隐私声明

包括“软件即服务”解决方案在内的 IBM 软件产品（“软件产品”）可能会使用 cookie 或其他技术来收集产品使用信息，以帮助改善最终用户体验、定制与最终用户的交互或用于其他用途“”。在许多情况下，软件产品不收集任何个人可标识信息。某些软件产品可帮助您收集个人可标识信息。如果本软件产品使用 cookie 收集个人可标识信息，那么以下段落阐述了有关本产品使用 cookie 方面的特定信息。

根据部署的配置，此软件产品可能会使用会话 cookie 来收集每个用户的用户名，以用于会话管理、认证和单点登录配置。可禁用这些 cookie，但是禁用也可能会消除它们所启用的功能。

如果为此软件产品部署的配置使您作为使用者能够通过 Cookie 和其他技术从最终用户收集个人可标识信息，那么您应该征求有关适用于这种数据收集的任何法律的法律建议，包括对通知和同意的任何要求。

有关将各种技术（包括 cookie）用于这些目的的更多信息，请参阅 IBM 隐私策略 (http://www.ibm.com/privacy) 和 IBM 在线隐私声明 (http://www.ibm.com/privacy/details) 中标题为为“Cookie、Web Beacon 和其他技术”“”的部分以及“IBM Software Products and Software-as-a-Service Privacy Statement”“”(http://www.ibm.com/software/info/product-privacy)。

