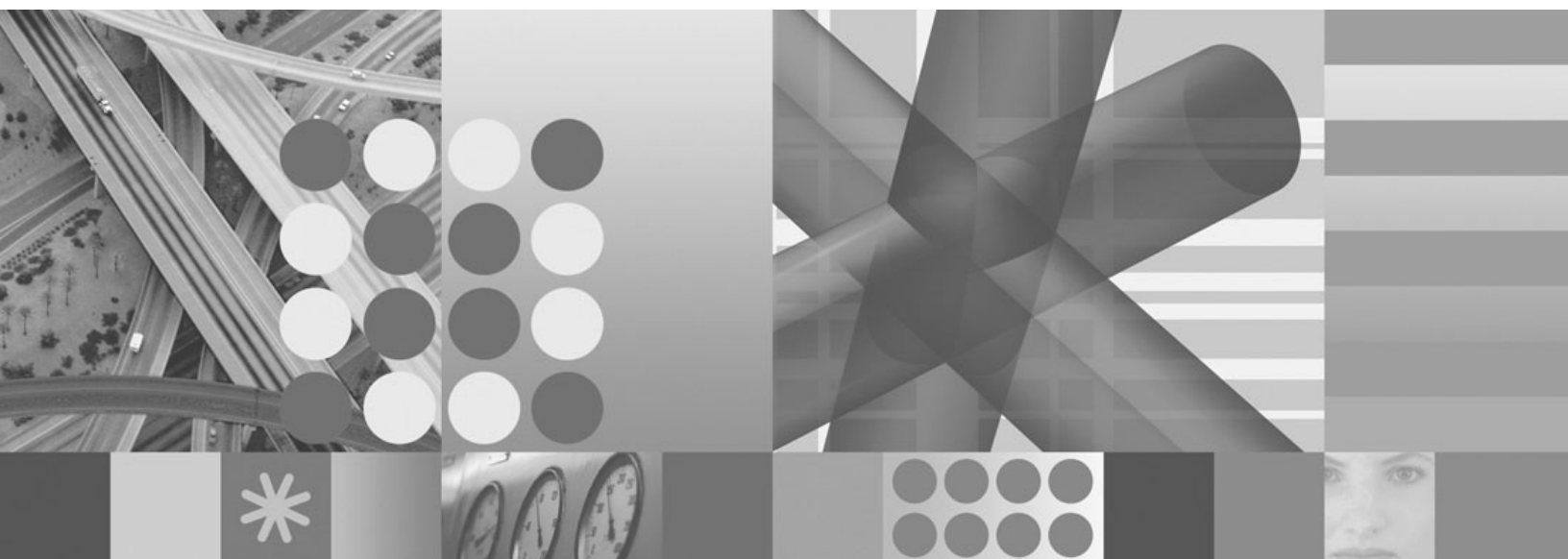




Administrator's Guide



Administrator's Guide

Note

Before using this information and the product it supports, read the information in "Notices" on page 263.

September 2009

This edition applies to version 6, release 2, modification 2, of IBM Tivoli Monitoring (product number 5724-C04) and to all subsequent releases and modifications until otherwise indicated in new editions. © Copyright International Business Machines Corporation 2005, 2009. US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© **Copyright International Business Machines Corporation 2005, 2009.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	vii
--------------------------	------------

Tables	ix
-------------------------	-----------

About this guide.	xi
------------------------------------	-----------

Chapter 1. Introduction 1

New in this release	1
New in Version 6.2.2.	1
New in Version 6.2.1.	4
New in Version 6.2.0.	6
IBM Tivoli Monitoring family of products	10
Tivoli Management Services components	10
Tivoli Enterprise Portal client	11
Desktop, Browser, and Java Web Start clients	12
Historical data collection	13
System administrator tasks	13

Chapter 2. Preparing your Tivoli Enterprise Portal environment 15

Browser client	15
Java runtime environment (JRE) versions	15
First time logon	15
Internet Explorer security settings	16
Windows write and delete privileges	16
Adding your company logo and URL	17
Starting the Tivoli Enterprise Portal client	17
Starting the desktop client	17
Starting the browser client	18
Using Web Start to download and run the desktop client	18
Installing the IBM JRE	19
Enabling tracing for the JRE	20
Downloading and running the desktop client	20
Manually creating a shortcut for the Web Start client	22
Starting the desktop client on another portal server	22
Starting the browser client on another portal server	23
Specifying the browser used for Launch Application and for online help	24
Adding operating platforms to the Navigator	26
Secure Socket Layer transmissions	26
Enabling TIP Web Service for Tivoli Integrated Portal charts	26

Chapter 3. Editing the portal configuration settings 29

Editing the portal client parameters	29
Editing client parameters	29
Portal client parameter list	30
Enabling the HTTP proxy server	35
Setting application properties for Linux and UNIX	35

Setting the environment variable when the hub is on z/OS	37
Editing the environment configuration	37
Editing the portal server environment file	38
Portal server environment variables	38
Controlling the number of logon attempts	40
Reducing processing load on the portal server.	40
Duper	41
Event management configuration	41
Federal Information Processing Standard enablement	42

Chapter 4. Setting up asymmetric encryption 43

Setting the JRE for GSKit and starting Key Manager	43
Creating a new key database	44
Creating a new public-private key pair and certificate request	44
Using a temporary self-signed certificate.	45
Receiving the CA-signed certificate	45
Save the password to a stash file	45

Chapter 5. Enabling user authentication 47

Configuring user authentication through the hub monitoring server	47
Prerequisites for configuring authentication on the hub monitoring server	48
Configuration procedures.	50
Running ldapsearch with LDAP configuration.	51
Configuring user authentication through the	54
Prerequisites for configuring authentication on the	54
Using single sign-on	56
Configuration procedures.	57
TEPS/e administration console	60
Mapping user IDs to LDAP distinguished names	60
Importing and exporting LTPA keys	61
Tivoli Enterprise Portal distinguished names	62
Reconfiguring the browser client for SSO	62
Migrating authentication from the monitoring server to the portal server	62

Chapter 6. User administration 65

Administer Users	65
Users and User Groups	66
Permissions	66
Applications	69
Navigator views.	69
Member Of and Members	70
Managing user IDs	70
Viewing and editing a user ID	70
Adding a user ID	71
Removing a user ID	72
Default user	73
Managing user groups.	73

Viewing user group memberships	74
Adding a user group	74
Reviewing and editing a user group	75
Removing a user group	75
Notes on user administration	76
Troubleshooting logon error messages	79

Chapter 7. Customizing event integration with Tivoli Enterprise Console 81

Default mapping of situation events to Tivoli Enterprise Console events	81
Expanding a generic event message situation description	83
Generic mapping for agent specific slots	83
Assigning severity for Tivoli Enterprise Console events	85
Localizing message slots	85
Situation event statuses and Tivoli Enterprise Console event generation	86
Synchronizing situation events	88
Checking the Tivoli Enterprise Console event cache	88
Changing the configuration of the event synchronization on the event server	89
Defining additional monitoring servers for the event synchronization on the event server	89
Closing sampled events	90
Changing rule set parameters for omegamon.rls	90
Tuning considerations	91
Using the Rules Check utility	92
Editing the Event Integration Facility configuration	93
Specifying EIF forwarding for a situation event	94
Customizing the event message	96
Updating the XML used by the MCS Attribute Service	96
Limitations on forwarding events to the Tivoli Enterprise Console	97
Using the NetView console through the Tivoli Enterprise Console event viewer	97

Chapter 8. Customizing event integration with Tivoli Netcool/OMNIbus 99

Default mapping of situation events to OMNIbus alerts	99
Expanding a generic event message situation description	102
Generic mapping for agent specific slots	102
Localizing alert summaries	103
Synchronizing situation events	104
Changing the configuration of the event synchronization on the event server	104
Defining additional monitoring servers for the event synchronization on the ObjectServer	104
Deleting or clearing sampled events	105
Customizing the OMNIbus configuration	105
Editing the Event Integration Facility configuration	106
Specifying situation events that send an OMNIbus event	107

Customizing the event message	108
---	-----

Chapter 9. Configuring connectors for the common event console 109

Common Event Console Configuration window	109
ITM Connector tab	110
TEC Connector tab	110
OMNIbus Connector tab	112
Names of Extra Columns tab	113
Best practices when event synchronization is used	115
Troubleshooting problems with connection to Tivoli Enterprise Console server on Linux systems	115

Chapter 10. Working with monitoring agents 117

Adding an agent through the Tivoli Enterprise Portal	117
Configuring an agent through the Tivoli Enterprise Portal	118
Starting, stopping, and recycling an agent through the Tivoli Enterprise Portal	119
Updating agents	120
Updating an agent through the Tivoli Enterprise Portal	120
Updating an agent through the command-line interface	121
Removing an agent through the Tivoli Enterprise Portal	122
Changing the monitoring server an agent connects to	122

Chapter 11. Agent autonomy 123

Autonomous capabilities	123
Configuration parameters for autonomous behavior	125
Situation limitations	129
Configuring Agent Management Services on Tivoli System Monitoring Agents	130
Private situations	131
Private situation operation	131
Private situation XML specification	133
Exported enterprise situation XML specification	137
Private situation examples	142
Private history	147
SNMP alerts	147
SNMP alert configuration	147
Configuring OMNIbus to receive SNMP alerts	149
Sample OMNIbus rules for SNMP alerts	151
SNMP message types for agent and situation state alerts	153
Trap configuration XML specification	159
SNMP PassKey encryption: itmpwdsnmp	167
Agent Service Interface	168
Starting the Agent Service Interface	168
Agent Service Interface - Agent Information	169
Agent Service Interface - Situations	170
Agent Service Interface - History	172
Agent Service Interface - Service Interface Request	172
Autonomous agent activity log	177

Chapter 12. Agent Management Services 179

Features of the Tivoli Agent Management Services	179
Component relationships	179
Component descriptions.	180
Installing and configuring Tivoli Agent Management Services	181
Monitoring the availability of agents	182
Managing the agent manually.	182

Chapter 13. Managing historical data 185

About historical data collection	185
Managing your historical data.	186
Historical data status and requests	187
Historical collection options	188
Performance impact of historical data requests	189
Impact on the Tivoli Enterprise Monitoring Server or the monitoring agent of large amounts of historical data	189
Requests for historical data from large tables	189
Scheduling the warehousing of historical data	190
Using a data mart to improve long or complex queries	190
Planning collection of historical data	193
Developing a strategy for historical data collection.	193
Warehousing your historical data.	199
Before you begin	199
Configuring your data warehouse	200
Collecting Agent Operations Log history	207
Converting short-term history files to delimited flat files	208
Converting history files to delimited flat files on Windows systems	208
Converting history files to delimited flat files on an i5/OS system	211
Converting history files to delimited flat files on UNIX Systems	212
Converting history files to delimited flat files on HP NonStop Kernel Systems	214
Converting history files to delimited flat files on z/OS systems	215

Chapter 14. Tivoli Common Reporting 219

Tivoli Common Reporting Users	220
Prerequisites.	221
Upgrading from a previous version	221
Limitations	222
Importing and running reports	222
Step 1: Ensure that historical reporting is enabled	222
Step 2: Import a report package	223
Step 3: Configure the data source.	224
Step 4: Generate a sample report	225

Chapter 15. Replicating the Tivoli Enterprise Portal Server database . . 227

Prerequisites.	227
------------------------	-----

Running the migrate-export script	227
Running the migrate-import script	228
Running migrate-import from source Windows to target Windows.	228
Running migrate-import from source Windows to target Linux or UNIX.	229
Running migrate-import from source Linux or UNIX to target Windows	230
Running migrate-import from source Linux or UNIX to target Linux or UNIX	231

Appendix A. Tivoli Enterprise Monitoring Web services 233

Configuring Tivoli Monitoring Web Services (SOAP Server)	233
Defining hubs	234
Adding users	235
Configuring IBM Tivoli Monitoring Web Services (SOAP Server) on UNIX and Linux	235
Tuning SOAP transaction performance on AIX	236
About the SOAP client	236
Using IBM Tivoli Monitoring Web services	236
User IDs	237
Starting the SOAP client and making a request	237
Using your browser	237
Using the SOAP client command-line utility (kshsoap).	238
Issuing SOAP requests as system commands	239
SOAP methods.	240
Issuing second-level requests	246
Sample CT_Get SOAP request.	247
IBM Tivoli Monitoring Web services scenarios	248
Generating daily logical operation summaries and charts	248
Obtaining data snapshots and offline table and charts	249
Sending alerts into an IBM Tivoli Monitoring platform	250
Collaborative automation using SA IOM	250
Acknowledging an event within an IBM Tivoli Monitoring platform	251
Report contents.	251

Appendix B. Using the Tivoli Management Services Discovery Library adapter 253

Documentation library 255

IBM Tivoli Monitoring library	255
Documentation for the base agents	256
Related publications	257
Other sources of documentation	257

Support information 259

Notices 263

Figures

1.	Tivoli Integrated Portal Web Services and the cross-product connections.	27	4.	Data snapshot chart and table	249
2.	Interactions of Agent Management Services components with IBM Tivoli Monitoring components	180	5.	Data Snapshot Table	250
3.	Tivoli Common Reporting environment	220	6.	Universal Message Console Showing Messages Received.	251
			7.	Message Log Details	251

Tables

1. File locations for changing application properties for UNIX and Linux	36
2. Setting up asymmetric encryption	43
3. Where to configure LDAP authentication	47
4. Tasks required for enabling user authentication through the hub monitoring server.	47
5. Tasks to complete before configuring authentication.	48
6. LDAP configuration parameters.	49
7. SSL parameters for communication between hub and LDAP server	49
8. ldapsearch command line options and corresponding monitoring server configuration parameters.	52
9. Tasks required for enabling user authentication through the	54
10. Tasks to complete before configuring authentication.	54
11. LDAP configuration parameters.	55
12. SSO parameters	55
13. Tasks for enabling single sign-on	56
14. Tivoli Enterprise Console event class attributes	82
15. Special characters for attribute groups and names in Tivoli Enterprise Console events generated from forwarded situation events.	84
16. Situation name suffix mapping to Tivoli Enterprise Console event severity	85
17. Supported Tivoli Enterprise Console event server configuration parameters for the event integration facility (EIF)	94
18. Tivoli Netcool/OMNIBus ObjectServer attributes.	100
19. Mapping of situation attributes to OMNIBus attributes	101
20. Special characters for attribute groups and names in EIF events generated from forwarded situation events	103
21. Supported OMNIBus EIF probe configuration parameters for the event integration facility (EIF)	107
22. Situation formula functions available when an enterprise agent is connected or disconnected, or when the situation is private.	129
23. SNMP trap variables for agentStatusEvent	153
24. SNMP trap variables for agentSitSampledEvent	155
25. SNMP trap variables for agentSitPureEvent	157
26. TrapDest element XML specification	159
27. TrapAttrGroup element XML specification	163
28. Situation element XML specification	163
29. Agent lifecycle status traps	166
30. StatTrap element XML specification	167
31. Agent Service Interface <AGENTINFO> request.	172
32. Agent Service Interface <REPORT> request	172
33. Agent Service Interface <AGENTINFO> request.	174
34. Agent Service Interface <APPLICATION> request.	174
35. Agent Service Interface <SNMP> request	174
36. Agent Service Interface <AGENTSTAT> request.	175
37. Agent Service Interface <SITCONTROL> request.	175
38. Agent Service Interface <SITSUMMARY> request.	176
39. Agent Service Interface <OVERRIDES> request.	176
40. Agent Service Interface <TRANSCON> request.	176
41. Customizable elements of a common agent package file in Agent Management Services	181
42. Summarization functions.	194
43. krarloff rolloff program parameters	211
44. DD names required	216
45. KPDXTRA parameters	217
46. TCP/IP Fields in Hub Specification Dialog	234
47. SNA Fields in Hub Specification Dialog	234
48. Predefined SOAP Methods	240
49. Example of CT_Get SOAP Request sent/data Received	247

About this guide

This guide describes the administration of your IBM® Tivoli® Monitoring infrastructure, Tivoli Management Services. The chapter topics cover the tasks for

- Configuring and maintaining the clients and server
- Maintaining user IDs and user groups
- Integrating the event activities between the Tivoli Enterprise Console Event Server or the Tivoli Netcool/OMNIBus ObjectServer and the hub Tivoli Enterprise Monitoring Server
- Configuring connectors for the event systems that send event information to the Tivoli Enterprise Portal
- Using the to maintain agents that support the remote agent deployment feature
- Configuring Tivoli Enterprise Monitoring Agents for autonomous operation
- Managing historical data collection and the Tivoli Data Warehouse
- Replicating the database to another computer or to keep as a backup
- Using IBM Tivoli Monitoring Web Services SOAP methods to query and control your monitored environment

Chapter 1. Introduction

This chapter reviews the new features and enhancements to the interface and Tivoli Management Services administrative features, followed by a list of the administrative tasks you can expect to perform.

For information on how to use the Version 6.2.2 features, please consult the integrated help (Help → and Index) or the *IBM Tivoli Monitoring Tivoli Enterprise Portal User's Guide*.

New in this release

Review the latest enhancements to the and to the Tivoli Management Services components that are relevant to the *Administrator's Guide*.

New in Version 6.2.2

This topic describes enhancements to the since the release of Version 6.2.1. Many of the changes are obvious as soon as you log on, such as the new toolbar icons. Others are changes in behavior or changes that are not apparent until you open workspaces or one of the editors.

The Tivoli Enterprise Portal client features are described in the online help and *IBM Tivoli Enterprise Portal User's Guide*.

User interface updates

The toolbar buttons and graphic icons have been updated and consolidated to further align the Tivoli user interfaces. Move the mouse pointer over a tool in the Tivoli Enterprise Portal to see its identity.







Tivoli Enterprise Portal Version 6.2.1

Desktop mode:



Browser mode:

Tivoli Enterprise Portal toolbar for Version 6.2.2

Some of the icons have been regrouped to align them with their function:  **Switch to Home Workspace** has moved to the first position in the toolbar, next to  **Back** in the desktop mode toolbar, and  **Save** in the browser mode toolbar; and the  **Situation Event Console**,  **Common Event Console**, and  **Tivoli Enterprise Console** views are grouped near the end of the toolbar.

Desktop mode:



Browser mode:



Workspace gallery for identification and selection

As well as the default workspace that opens when you click a Navigator item, additional workspaces are often available for the item and selectable

through the Navigator pop-up menu or the View menu. Now you have the Workspace Gallery tool for showing you the workspaces that you can open for the current Navigator item.

Manage Tivoli Enterprise Monitoring Server workspaces and situations

New self-monitoring workspaces and situations have been added and are accessible through the Enterprise Navigator item to help you monitor for and diagnose typical monitoring server configuration issues.

Configure historical data collection with distribution lists

The Historical Collection Configuration window has been redesigned. It looks similar to the Situation editor, with a tree on the left and user assistance on the right until you select a tree item. You can now have multiple collection configurations for an attribute group. There is also a new distribution method called **Managed System (TEMA)** that enables you to specify the managed systems that data collection will occur on. (With this method, the managed systems must connect to a V6.2.2 Tivoli Enterprise Monitoring Server.)

Granular data collection with historical configuration object groups

Object grouping was introduced in the previous release for situations and managed systems. Now you can create historical configuration object groups and assign historical collections to them.




Managed system lists renamed to managed system groups

The term *managed system list* has been renamed to *managed system group* to follow the naming used in the Object Group editor.

Modeling conditions for situations

Now you can capture data from a query-based view and use it to model possible threshold scenarios, then apply the results to a situation. You can also select a situation from the Manage Situations at Managed System window, compare the criteria with current and historical data samples, for modeling, and use the results to edit the situation thresholds.

Baselining added to charts for predictive analysis

The bar chart, plot chart, and area chart have a new  **Add Monitored Baseline** tool for selecting a situation to compare with the current sampling and anticipated values. The plot chart and area chart also have a new  **Add Statistical Baseline** tool with statistical functions. In addition, the plot chart has a new  **Add Historical Baseline** tool for comparing current samplings with a historical period.

Situation overrides can be assigned to subnodes

Situation overrides for dynamic thresholding has been extended to include subnodes. For situations that are distributed to managed systems that are subnodes of other managed systems, you can now apply expression overrides to the managed system subnodes.

Create private situations, which run locally and are independent of the monitoring server

Agent autonomous mode, which ensures that event information is retained when communications are interrupted between an agent and its monitoring server, was introduced in IBM Tivoli Monitoring V6.2.1. Now agents are autonomous by default, whereby agent startup is independent of the Tivoli Enterprise Monitoring Server and information is collected at the agent and maintained even if the agent is stopped and started again, ready for transfer to the monitoring server when a connection is made. You can also install and configure autonomous-only OS agents that have no

dependency on the Tivoli Enterprise Monitoring Server and never need to connect to it. The autonomous agent can run private situations that are defined in a situation configuration XML file. The autonomous agent can also send SNMP traps that are defined in a trap destination XML file. See Chapter 11, “Agent autonomy,” on page 123.

Send SNMP traps

Send SNMP traps directly to a receiver, without ever connecting to a monitoring server. See Chapter 11, “Agent autonomy,” on page 123.

Agent Service Interface

Agent autonomous mode, which ensures that event information is retained when communications are interrupted between an agent and its monitoring server, was introduced in IBM Tivoli Monitoring V6.2.1. Now agents are autonomous by default, whereby agent startup is independent of the Tivoli Enterprise Monitoring Server and information is collected at the agent and maintained even if the agent is stopped and started again, ready for transfer to the monitoring server when a connection is made. You can also install and configure autonomous-only OS agents that have no dependency on the Tivoli Enterprise Monitoring Server and never need to connect to it. The autonomous agent can run private situations that are defined in a situation configuration XML file. The autonomous agent can also send SNMP traps that are defined in a trap destination XML file. See Chapter 11, “Agent autonomy,” on page 123.

Agent autonomy

Agent autonomous mode, which ensures that event information is retained when communications are interrupted between an agent and its monitoring server, was introduced in IBM Tivoli Monitoring V6.2.1. Now agents are autonomous by default, whereby agent startup is independent of the Tivoli Enterprise Monitoring Server and information is collected at the agent and maintained even if the agent is stopped and started again, ready for transfer to the monitoring server when a connection is made. You can also install and configure autonomous-only OS agents that have no dependency on the Tivoli Enterprise Monitoring Server and never need to connect to it. The autonomous agent can run private situations that are defined in a situation configuration XML file. The autonomous agent can also send SNMP traps that are defined in a trap destination XML file. See Chapter 11, “Agent autonomy,” on page 123.

Proxy Agent Services renamed to Agent Management Services

The Proxy Agent Services have been renamed to Agent Management Services to better describe their function. More base agents are supported and there are new capabilities to report and manage agent instances.

TIP Web Service for Tivoli Integrated Portal charts

In the previous release, users who wanted to see query-based views in the Tivoli Integrated Portal administrative console needed to have workspace administrator authority. This is no longer a requirement. Users can now view through the administrative console any Tivoli Enterprise Portal workspaces that their user ID has permission to view. For example, if the allowed applications for your user ID include Linux OS application but not DB2, then any Linux OS workspaces are available from the administrative console but not the DB2 workspaces. See `tipwebadmin_tipwebsevice.dita`.

Optional agent restart

After reconfiguring an agent, you can now choose whether to restart the agent immediately for the changes to take effect or to leave the agent

running and restart it at a later time. The Tivoli Service Manager (Manage Tivoli Monitoring Services window) has a new column named **Configuration** to show this status: **out-of-sync** for configuration changes that have not been implemented; or **up-to-date** for configuration changes that have been implemented by restarting the agent.

Client environment variables

`cnp.browser.installdir` to specify a different path for the browser view files, required if users will be running multiple instances of the portal client and possibly logging on to different versions of the portal server. See “Portal client parameter list” on page 30.

New in Version 6.2.1

The Tivoli Enterprise Portal client features are described in the online help and *IBM Tivoli Monitoring: User's Guide*.

Dynamic thresholding with situation override

Dynamic thresholding for situations means that you can override the expression values of a situation formula for a specific managed system (or group of managed systems) or for a specific time period or both. This capability enables you to adjust situations for conditions that are specific to a particular managed system or managed system group or to a particular time period.

Long situation names

Situation names can now be as long as 256 bytes and are no longer restricted to only letters, numbers, and underscores.

Organize situations and managed systems into named groups

The Object Group editor is a general, consistent mechanism for creating named groups of objects. The object types that can be grouped are situations and managed systems. The managed system group is being retired in this release and all the functions of the managed system group editor are now provided by the new object group editor.

Customize the EIF slots in the Situation editor for events that get forwarded to an EIF receiver

Through the EIF (Enterprise Integration Facility) tab of the Situation editor, you can now map situation events to the EIF receiver and customize the forward events.

Area chart

The new area chart view is similar to the plot chart. The difference is that the area from the X-axis and Y-axis to the plot point for each data series is filled with a pattern or color or both.

Chart area thresholds and markers, collapsible legend

Visual indicators for value ranges (chart thresholds) and for specific values (chart markers) can be added to the bar, plot, and area charts. Chart legends can be kept in a collapsible panel and expanded and hidden as needed to give maximum viewing space to the plot area. As well as colored labels for the attributes, you can specify a fill pattern.

Zooming in on chart areas

On bar charts, plot charts, and area charts, you can click and drag over an area of the chart that you would like to zoom into for closer scrutiny, then press Esc to return to the previous size.

Historical navigation mode to synchronize workspaces

When you open a workspace, the query-based views retrieve the latest

values. Then, as you open or link to other workspaces, their assigned queries also retrieve the latest values. However, you might be performing analysis over multiple workspaces that requires review of a fixed time or time range. You can turn on *Historical navigation mode* with a time span that you specify. Then, all workspaces you open will align to that time period until you turn it off.

Options for more frequent warehouse intervals

When configuring attribute groups for historical data collection, you have more choices for the data rolloff from the history files to the Tivoli Data Warehouse. As well as 1 hour and 1 day, you now can select 15 minutes, 30 minutes, or 12 hours.

Find Navigator items

The Find feature for Navigator items enables you to search for and locate items by criteria such as product code or associated situation, and using formula functions. .

Terminal view links

You can now build contextual links from the table and chart views on an OMEGAMON XE workspace to a terminal view in another workspace.

Tabbed workspaces

When using the Tivoli Enterprise Portal browser client, you can open workspaces in new tabs if your browser enables them.

Agent deployment status workspaces

New workspaces have been added to the Enterprise Navigator item for showing deploy depot information and the status of the past, current, and scheduled agent deployments.

Single sign-on support with Java Web Start client

As well as the browser client, you can now use the Java Web Start client to launch into the Tivoli Enterprise Portal and out to other Tivoli Web-based and Web-enabled solutions without needing to re-enter your authentication credentials.

Tivoli Proxy Agent Services

New services can monitor the availability of agents and respond automatically (such as with a restart) if the agent operates abnormally or exits unexpectedly.

Configuring an HTTP proxy server for the browser view

The procedure for setting up an HTTP proxy server for portal browser view has been simplified and is the same for all Tivoli Enterprise Portal clients.

Enabling FIPS

The Tivoli Enterprise Portal Server has a new environment variable that can be enabled for conformance to the Federal Information Processing Standard (FIPS) 140–2 specification.

64-bit integers are now supported

Support for 64-bit integers has been added. Many of the Version 6.2.1 products have new attribute groups, attributes, situations, and workspaces that use the new 64-bit integer values. For example, there are workspaces with a superseded version that displays queries with a signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). You will also see similar 'superseded' notations

for attribute groups, attributes, and situations that have a 64-bit counterpart. See your product user's guide for details.

New schema publication tool simplifies generation of SQL statements for creating the Tivoli Data Warehouse

With the new schema publication tool, you can now generate the SQL statements needed to create the database objects (data warehouse tables, indexes, functions, views, and ID table inserts) required for initial setup of the Tivoli Data Warehouse. For details, see "Generating SQL statements for the Tivoli Data Warehouse: the schema publication tool" in the *IBM Tivoli Monitoring: Installation and Setup Guide*.

Tivoli Data Warehouse now supports DB2 on z/OS

You can now create your Tivoli Data Warehouse repository using DB2 running on z/OS. Although the Warehouse Proxy agent still runs only on Windows, Linux, or UNIX, the data warehouse itself is stored in DB2 on z/OS databases. Data communication is supported using either an ODBC or a JDBC connection. See "Tivoli Data Warehouse solution using DB2 on z/OS" in the *IBM Tivoli Monitoring: Installation and Setup Guide* for instructions on setting up your Tivoli Data Warehouse environment to run with a DB2 on z/OS repository.

Command Line Interface tacmds for new features

The CLI has dozens of new tacmds for many of the new Tivoli Enterprise Portal features and for features that are exclusive to the CLI, such as for exporting queries and custom Navigator views and their associated situations. See the *IBM Tivoli Monitoring: Command Reference* for details.

Setting traces

On Linux and UNIX, the file that contains the KBB_RAS1 parameter for setting a trace has moved from the .config file in the <itm_install_dir>/config directory to ms.ini. On distributed operating platforms the value is no longer enclosed in single quotes. See the *IBM Tivoli Monitoring: Troubleshooting Guide* for information on setting traces.

New in Version 6.2.0

Event type tag for SOAP methods

The CT_Alert, CT_Acknowledge, CT_Reset, and CT_Resurface methods were modified to support the <type> tag, which specifies the event type.

User groups

The Administer Users window has been significantly enhanced to enable the creation of user groups, including assigned permissions.

Lockout and lockout override

Logon Permitted is a new permission that enables the administrator to lock out a user ID, preventing the user from logging on to the portal server or to override an automatic lock out, which occurs after a set number of invalid logon attempts.

User authentication

Support has been added to enable external authentication of users with standards-based Lightweight Directory Access Protocol (LDAP) (LDAP) to shared registries. The hub monitoring server can be configured to validate user IDs and passwords using either the local system registry or a central LDAP authentication and authorization system. See also the "Single sign-on for launching to and from other Tivoli applications" entry in the changes for .

Flexible scheduling of summarization and pruning

The Summarization and Pruning agent configuration window has been enhanced to allow for flexible scheduling and to have the data warehouse and warehouse aggregation logs trimmed automatically after a specified number of days, months, or years.

The Defaults tab has been removed. Now, when you use the History Collection Configuration window to configure historical data collection for an attribute group, no summarization and pruning check boxes are selected for you by default.

Null values in summarized historical data

You now see null in a table cell or chart point when values that were identified as invalid were reported from a monitoring agent for a given summarization period.

More frequent intervals for historical data collection

One-minute and five-minute intervals have been added to the **Collection Interval** options, enabling you to save more frequently to the short-term history files at the monitoring agent or monitoring server.

There are no longer pre-selected check boxes for summarization and pruning when you use the configure historical data collection for an attribute group.

Common and Tivoli Enterprise Monitoring Server attributes

The common attribute groups, Local Time and Universal Time, have a new *Time* attribute for the time of the data sampling corrected for local time zone and daylight saving time.

Two of the Tivoli Enterprise Monitoring Server attribute groups also have new attributes: The Managed System Status attribute group adds a *Reason* attribute for the two-character reason code, if one exists, for the managed system status. The Situation Definition attribute group adds a new *Last Release* attribute to identify the release of the product to associate with the situation.

If your product was updated for this release of IBM Tivoli Monitoring, check the **New in this release** section of the product user's guide for a list of the new and updated attribute groups.

Some distributed products might require that you create new queries before you can see the new attributes in the query or queries for that attribute group or see no query for a new attribute group.

Seven event severities





The state of an event that opens for a true situation can be set to informational, warning, or critical. Now you have four additional states to choose from for associated situations, table view thresholds, and for filtering an event console view:  Unknown,  Harmless,  Minor, and  Fatal.

Table view threshold icons



The table view has a feature that highlights the background of any cell whose value exceeds a given threshold. Before this release, thresholding was limited to three background colors to indicate an informational, warning, or critical severity. Now, as well as having four more severities available, you can choose to display either an icon in the cell () or a background color () .

Table view style properties

A new option on the Style tab and in the workspace presentation cascading style sheet enables you to control the default font styling of table view header and footer text.

Common event console view

The common event console enables you to view and manage events from the Tivoli Enterprise Monitoring Server in the same way as the situation event console, plus it incorporates events from the Tivoli Enterprise Console Server and the Tivoli Netcool/OMNIBus ObjectServer if your managed environment is configured for those servers.

Situation editor EIF tab

The new EIF (event integration facility) tab has options for forwarding events that open for the situation to one or more EIF receivers and to specify the severity. The tecserver.txt mapping file that was used in version 6.1 is no longer needed.

Refresh Tivoli Enterprise Console information

A new CLI (Command Line Interface) tacmd refreshTECInfo command enables you to have the Tivoli Enterprise Console Event Forwarder reprocess updated event definitions, EIF configuration, and custom event mapping files without requiring a hub monitoring server recycle.


Situation event acknowledgement

Event acknowledgement has several new enhancements to facilitate quick acknowledgements, writing and reviewing notes, and attaching files to the acknowledgement.

Enterprise Status workspace

The first indication of the acknowledgement enhancements is in the Enterprise Status workspace, which adds a new view called My Acknowledged Events, as well as a new **Owner** column in the situation event console view that shows the ID of the user who acknowledged the situation event.

Home workspace

Initially, Enterprise Status is the first workspace to be displayed when you log on to the portal server. This is your home workspace. With the **Assign as Home Workspace** option, you can now establish another workspace, whether at this Navigator level or another and whether on this Navigator view or another, as your home workspace. The  **Home** tool opens the home workspace.

Topology view

Topology view is a new type of query-based view that enables you to create views from relational data sources. Attributes from the query are rendered as objects and connected to related objects.

Another topology view that is available at the Enterprise level of the Navigator is the TMS (Tivoli Management Services) infrastructure view, which visually expresses the relationships and linking of monitoring agents and other components to the hub monitoring server.


Self-Monitoring Topology workspace

The Enterprise Navigator item has a new Self-Monitoring Topology predefined workspace. The purpose of this workspace is to introduce the self-monitoring capabilities that are available through the Tivoli Enterprise Portal.

Dynamic linking

A new link type has been added to the workspace link feature that enables the link author to identify the target workspace by the host identifier. The *dynamic* link type adds more opportunities for workspace linking, such as to provide links to workspaces of other types of monitoring agents.

Navigator view icon in the status bar

When the Navigator view has been collapsed, you can now restore it or open to another Navigator view by a click or right-click of  <Navigator name> in the status bar.

Bar chart overlay

A new overlay feature has been introduced that allows one or more related attributes to be plotted against the bar chart. This can highlight the relationship of related values, and is useful for visualizing trends from historical data.

Plot chart overlay

In earlier releases the plot chart view was able to show data only from the first row of a data sampling. The plot chart properties have been enhanced for plotting multiple-row attribute groups (or historical data from a single-row attribute group) and multiple managed systems, and for controlling the refresh rate independent from the workspace as a whole.

The plot chart also has a new overlay feature that can be used to establish a secondary value axis.


Workflow editor

You can now launch the Situation editor from an activity in the Workflow editor to edit the situation that it references.

Application window

The banner artwork in the browser client has changed, the Navigator tabs have been modernized, as have the view title bars, which also have two new buttons for hiding or showing the view toolbar and for opening the Properties editor.

Creating a new view

After you click a view tool to create a new view, the mouse pointer adopts the view icon (instead of the  hand icon) on Windows® systems. And you can now press Escape or click somewhere in the toolbar if you then decide not to add the view.

Moving a view

You can now drag a view by its title bar to a new location in the workspace.

Searching in a view


The view toolbar for the table, notepad, and browser views has a new

 **Find** tool for quickly searching through text in the view.

Cell function

The  **See if Null (no value) has been detected** function can be used in the Filters and Thresholds formula editors to locate attributes for which no value has been retrieved.

operator

The  **Value of expression** function has a new comparison operator that can be used in situations and in table view Filters and Thresholds to highlight specific text values.

Browser view

The browser view now supports most types of Web content, such as JavaScript™, Applets, and PDF files.

IBM Tivoli Monitoring family of products

IBM Tivoli Monitoring products help you manage the performance and availability of distributed operating systems and applications. These products are based on a set of common service components, referred to collectively as Tivoli Management Services. Tivoli Management Services provides security, data transfer and storage, notification mechanisms, user interface presentation, and communication services in an agent-server-client architecture. These services are common to many product suites such as IBM Tivoli OMEGAMON XE mainframe monitoring, IBM Tivoli Composite Application Manager, and IBM Tivoli Performance Analytics for Tivoli Enterprise Portal.

After you have installed and initially configured Tivoli Management Services and the products that rely on them, consult this guide to apply further customization in a distributed environment. (*Configuring the Tivoli Enterprise Monitoring Server on z/OS®* is provided in the guide of the same name.) It also has general administrative information for the managed systems that share these common services. Product-specific administrative information is given in the guides for the individual products.

Tivoli Management Services components

The Tivoli Management Services provide the infrastructure for your Tivoli Enterprise Monitoring Agents.

Client The IBM Tivoli Monitoring client, Tivoli Enterprise Portal is a Java-based user interface for viewing and monitoring your enterprise network. Depending on how it was installed, you can start Tivoli Enterprise Portal as a desktop application or through your browser as a Web application.

Presentation server

The Tivoli Enterprise Portal client connects to the Tivoli Enterprise Portal Server. The Tivoli Enterprise Portal Server is a collection of software services for the client that enables retrieval, manipulation and analysis of data from the monitoring agents on your enterprise.

Management server

The Tivoli Enterprise Portal Server connects to the main, or *hub*, Tivoli Enterprise Monitoring Server. The monitoring server acts as a collection and control point for alerts received from the enterprise monitoring agents, and collects performance and availability data from them. The hub monitoring server correlates the monitoring data collected by monitoring agents and any remote monitoring servers and passes it to the portal server for presentation in the portal console.

Agents

Tivoli Enterprise Monitoring Agents are installed on the systems or subsystems whose applications and resources you want to monitor. An agent collects monitoring data from the *managed system* and passes it to the monitoring server to which it is connected. The client gathers the current values of the attributes and produces reports formatted into tables, charts, and relational table-based topology views. It can also test the values against a threshold and display an alert icon when that threshold is

exceeded or a value is matched. These tests are called *situations*. OS agents can be installed outside the enterprise as *Tivoli System Monitoring Agents*. They do not connect to nor have any reliance on the Tivoli Enterprise Monitoring Server. They can run *private situations*, which are independent of the monitoring server, save data samples for attribute groups as *private history*, and can send SNMP alerts to an Netcool/OMNIBus SNMP Probe.

Help server

IBM Eclipse Help Server is installed with the portal server and provides presentation and search features for the integrated help system.

Data warehouse

The Tivoli Data Warehouse is an optional component for storing historical data collected from agents in your environment. The data warehouse is located on a supported database (such as DB2®, Oracle, or Microsoft® SQL).

Event synchronization

The event synchronization component is optional. It is configured to send situation event updates that were forwarded to a Tivoli Enterprise Console Event Server or a Tivoli Netcool/OMNIBus ObjectServer back to the monitoring server.

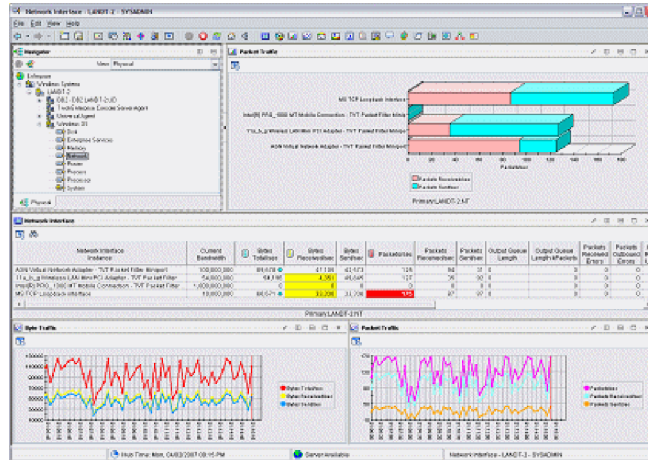
Tivoli Enterprise Portal client

Tivoli Enterprise Portal is the interface for your IBM Tivoli Monitoring products. In the same way you use your browser's home page as a starting point for navigating the Internet, you use Tivoli Enterprise Portal to get a high level overview of your network environment.

One section of the window displays the Navigator, a tree-like view of your monitored network, from the top level down to individual groupings of information collected by monitoring agents. The rest of the window is filled with views pertinent to the chosen item in the Navigator tree. From the top level or from your home workspace, you can navigate to specific locations to check activity and investigate problems.

This workspace was customized for the selected item in the tree. This workspace was designed with a bar chart, two plot charts, and a table that displays a background color for cell values that exceed a certain threshold. You can create and customize additional workspaces for every item in the tree.

The event indicators that display in the tree, or Navigator, are the results of tests, called situations, that run on your monitored systems. When the condition described in the situation is true, a colored icon overlays the affected items in the tree. Use the Situation editor to set up conditional alerts that monitor your environment automatically. Use the Workflow editor to set up policies to automate your environment.



Desktop, Browser, and Java Web Start clients

The Tivoli Enterprise Portal client can be deployed in three ways, as described briefly here and in more detail in the *IBM Tivoli Monitoring Installation and Setup Guide*:

Desktop

The desktop client requires that you load and run the installation software on each computer where the desktop client will be run. Users start Tivoli Enterprise Portal the same way they do their other locally installed applications. With the desktop client, you can also create multiple instances for connecting to different portal servers.

Browser

The browser client installation software resides on the Tivoli Enterprise Portal Server. The client software is downloaded from there to your computer the first time you log on to the portal server from your browser, and thereafter only when there are software updates.

You can start the browser client from any browser-enabled computer by entering the URL for the portal server. In this mode of operation, each portal workspace has a URL, so you can save a workspace to your Favorites list.

With the browser client you can launch from the Tivoli Enterprise Portal to other Tivoli Web-based and Web-enabled applications, and from those applications into the portal without re-entering your log-on credentials. This single sign-on solution uses a central LDAP-based user registry to authenticate sign-on credentials.

Java™ Web Start

With Java Web Start, like the browser client, the client software is accessed through a URL and downloaded from the portal server. Unlike the browser client, which is always run inside the browser, the Web Start client is run as a desktop application. Whenever updates to the client software are available, they are downloaded from the portal server automatically. References to *desktop client* behavior in this guide also assumes the Java Web Start client unless otherwise stated. Single sign-on is an example: As well as the browser client, you can use single sign-on with the Web Start client

Historical data collection

In addition to the real-time reports offered by Tivoli Enterprise Portal workspaces, you can configure historical data collection to store the data being collected by your monitoring agents for historical reports and situations. You can specify the following:

- Attribute groups for historical data collection
- Data collection interval.
- Data warehousing intervals if you choose to write data to the Tivoli Data Warehouse
- How data samples are grouped for pruning from the Tivoli Data Warehouse
- Pruning schedule of warehoused data.
- Storage location for the short-term history files before they are sent to the data warehouse. Data samples can be stored at the monitoring agent or on the Tivoli Enterprise Monitoring Server.

To ensure that data samplings are saved to populate your predefined historical workspaces, you must first configure and start historical data collection. Real-time workspaces are available whether you start historical collection or not.

System administrator tasks

A system administrator has the highest level of authority and can access all IBM Tivoli Monitoring features.

This list represents the types of tasks a system administrator might perform:

- Establishes user IDs and user groups with the appropriate permissions for their jobs.
- Designs workspaces for Navigator items and makes these workspaces available to users based on their established permissions.
- Defines queries that can be applied to table and chart views to specify the attributes and attribute value ranges to retrieve from the monitoring server
- Writes definitions for launching applications and makes them available to users based on their established permissions.
- Creates command line actions that can run at the specified managed system from the portal client, and makes them available to users who have been granted authority.
- Creates situations using the visual programming facilities
- Sets the severity of a situation for a particular Navigator item and what, if any, sound plays when the situation is true and an event opens
- Decides which situations apply to which managed systems, a process called distribution
- Provides expert advice to display when certain situations evaluate true
- Creates policy workflows, which are actions to take when situations evaluate true
- Creates, installs, upgrades, distributes and configures agents on remote hosts from a central location
- Starts, stops, and recycles agent processes

Chapter 2. Preparing your Tivoli Enterprise Portal environment

Review these topics for additional configuration of the Tivoli Enterprise Portal client environment.

Browser client

Users start the browser client by entering the URL for the integral HTTP server on the .

The advantages of the browser client are:

- Easy deployment. The browser client is installed the first time users log on to the URL for the Tivoli Enterprise Portal integral HTTP server.
- Software upgrades are automatic. When users log on, the browser client is checked against the one at the Tivoli Enterprise Portal Server and downloads a newer version from the server if one is detected.
- Global parameter settings are set for all users connected to the same Tivoli Enterprise Portal Server.
- Workspaces have identifying URLs that can be referenced in Web pages and when launching from another Web-enabled application.
- Includes a banner that can be customized with your company logo and URL.

Java runtime environment (JRE) versions

The Tivoli Enterprise Portal Server and client run Java-based software. When you install the portal server, a check is done for IBM Java 1.5 on your system and, if not found, is installed automatically. This check also takes place when you install the desktop client, Java WebStart client, or log on from a browser with this difference: Sun Java 1.5.0_xx through 1.6.0_xx is also recognized as a valid JRE for the client (but not 1.6.0_xx for Firefox on Linux).

If you have different versions of the Java Runtime Environment installed locally, they can coexist. Tivoli Enterprise Portal V6.2.1 and V6.2.2 require IBM Java V1.5 or Sun Java V1.5. The desktop client must be at the same version as the portal server it connects to. This is also true for the browser client, but Java versioning is controlled at the portal server and upgraded automatically when you connect to a newer portal server.

Running IBM Tivoli Monitoring V6.1 agents and IBM Tivoli Monitoring V6.2 agents on the same computer requires Java 1.4.2 and 1.5 on that computer.

First time logon

The first time the URL for Tivoli Enterprise Portal is entered from a system, the Java Plug-in transfers the necessary files from the portal server (on Windows, the files reside in the <itm_install_dir>\cnb branch; on operating systems such as UNIX, they are in the <itm_install_dir>/cw branch).

From then on the browser client software does not need to be downloaded again until a new version has been installed. The Java plug-in maintains the version levels of the files on users' computers and compares them with the version levels

on the integral HTTP server. If it detects files that are older than the ones on the HTTP server, it downloads the latest files.

Be sure you have sufficient free space for the downloaded files. If the disk runs out of space during the download, you are not warned.

Internet Explorer security settings

About this task

If you have the Internet Explorer security level set to high, you must adjust the settings to run Tivoli Enterprise Portal. Otherwise, Tivoli Enterprise Portal browser client cannot run.

Check the security settings

1. In Internet Explorer, select **Tools** → **Internet Options**
2. Select the **Security** tab.
3. Click **Internet** if you are running Tivoli Enterprise Portal through the Internet; or **Intranet** if you are running Tivoli Enterprise Portal through your intranet.
4. Change your security settings to **Default Level**
5. Click **OK** to save.

Keep current security settings

About this task

If you wish to keep your current security settings, you can add the Tivoli Enterprise Portal Web site to your Trusted Sites zone.

1. In Internet Explorer, select **Tools** → **Internet Options**
2. Select the **Security** tab.
3. Click **Trusted Sites** → **Sites**, and enter the URL for Tivoli Enterprise Portal.
4. Clear the check box that checks for (https:) for all sites at this zone, click **Add**
Choose the medium security level or lower for all sites in the **Trusted Sites** zone.
5. Click **OK** to save.

Windows write and delete privileges

Starting with Windows 2000, write and delete privileges for certain folders and registry keys were removed from the Users group. These privileges are required for anyone intending to use the Java Web Start client or the browser client. Otherwise, Java exception errors are encountered during attempts to start the product.

Before users can download the Web Start client or start the browser client, the Windows administrator must assign the required permissions to individual user IDs or the Users group, or create a new group with the necessary permissions and assign users to this group in addition to the Users group. The required permissions are:

- Write and Delete permissions on the directory where Windows is installed, such as C:\WINDOWS.
- Set Value, Create Subkey, and Delete permissions on registry key HKEY_LOCAL_MACHINE\SOFTWARE.

Note: The Windows permissions scheme affects the Tivoli Enterprise Portal browser mode and other third-party software installed through Internet Explorer.

Adding your company logo and URL

The Tivoli Enterprise Portal browser application looks much as it does in desktop mode, except that it also has a banner with a link to ibm.com. You can customize the Tivoli Enterprise Portal browser client by replacing the logo and URL with your organization's.

About this task

Take these steps to customize the portal client banner:

1. On the computer where you installed the Tivoli Enterprise Portal Server, open the following file in an HTML editor or text editor:

```
<install_dir>\cnb\bannerimage.html
```

2. Edit the HREF and IMG SRC tags for your organization's URL and logo graphic file:
 - a. Replace the **href** ' + URL + ' placeholder with your organization's URL.
 - b. Replace the **img src** ' + URL + ' placeholder with the name of your organization's logo GIF or JPG file.
 - c. Replace the **alt** ' + URL + ' placeholder with the text that should display when the mouse pointer is over the image, such as the URL.
3. Save the file and exit the editor.
4. Copy the logo graphic to the `<install_dir>\cnb\` directory.

Results

Users now see your logo on the right-hand side of the banner the next time they start browser mode.

Starting the Tivoli Enterprise Portal client

About this task

After you have successfully installed and configured all the components of your IBM Tivoli Monitoring environment, you can verify the installation and configuration by launching the Tivoli Enterprise Portal to view monitoring data. You can access the Tivoli Enterprise Portal using either the desktop client or the browser client.

Your monitoring server and portal server must be running for the portal client to start successfully.

Starting the desktop client

About this task

Follow these steps to start the desktop client:

On Windows:

1. Click **Start** → **Programs** → **IBM Tivoli Monitoring** → **Tivoli Enterprise Portal**.
2. Type your user ID and password in the logon window. The default user ID is `sysadmin`.

3. Click **OK**.

On Linux, run the following command to start the portal desktop client:

```
./itmcmd agent start cj
```

Starting the browser client

About this task

Follow these steps to start the browser client:

1. Start the browser.
2. Type the URL for the Tivoli Enterprise Portal into the **Address** field of the browser:
`http://systemname:1920///cnp/client`
where the *systemname* is the host name of the computer where the Tivoli Enterprise Portal Server is installed, and 1920 is the port number for the browser client. 1920 is the default port number for the browser client. Your portal server might have a different port number assigned.
3. Click **Yes** on the Warning - Security window.
4. Type your user ID and password in the logon window. The default user ID is `sysadmin`.
5. Click **OK**.

Using Web Start to download and run the desktop client

This section is reproduced from the *IBM Tivoli Monitoring: Installation and Setup Guide* for your convenience.

A desktop client obtained from the through IBM Web Start for Java benefits from centralized administration from the server. Like the browser client, it is automatically configured with the latest updates each time you start the client, and there is no need to configure application support.

Before you use IBM Web Start for Java to download the desktop client from the Tivoli Enterprise Portal Server:

- The Tivoli Enterprise Portal Server must be installed. (See the *IBM Tivoli Monitoring: Installation and Setup Guide*.)
- IBM 32-bit Runtime Environment for Windows, Java 2, version 5.0 must be installed on the computer to which you want to download the desktop client. You can download the IBM JRE installer from the Tivoli Enterprise Portal Server. The IBM JRE must be installed as the system JVM.

If you want to run the desktop client on a system that already has a Tivoli Management Services base component installed (such as a monitoring server or the portal server), there is no need to install the IBM JRE. The correct version of the IBM JRE is installed with the Tivoli Management Services component.

If you run the desktop client using Web Start instead of installing it from the installation media, you must configure the JRE to enable tracing for the desktop client.

Installing the IBM JRE

About this task

If you intend to download and run the desktop client using Web Start on a computer where no IBM Tivoli Monitoring base component is installed, you must first install IBM Java 1.5. You download an installer from the computer where the is installed:

Windows: Installing the IBM JRE

Install the IBM Java Runtime Environment on the computer where you plan to start the desktop client using Java Web Start.

About this task

Take these steps to download the IBM JRE installer from the Tivoli Enterprise Portal Server and install the JRE on a Windows computer:

1. Start the browser on the computer to which you want to download the installer.
2. Enter the following URL in the **Address** field of the browser, where `<TEPS_host_name>` is the fully qualified host name of the computer where the portal server is installed (for example, `myteps.itmlab.company.com`):
`http://<TEPS_host_name>:1920///cnp/kdh/lib/java/ibm-java2.exe`
3. When prompted, save the **java/ibm-java2.exe** file to a directory on your hard drive.
4. Change to the directory where you saved the **java/ibm-java2.exe** file and double-click the file to launch the JRE installer to start the installation program.
5. On the pop-up window, select the language from the drop-down list and click **OK**.
6. Click **Next** on the Welcome page.
7. Click **Yes** to accept the license agreement.
8. Accept the default location for installing the JRE or browse to a different directory. Click **Next**.
9. Click **NO** on the message asking if you want to install this JRE as the system JVM. Make Java 1.5 the system JVM only if there are no other JREs installed on the computer.
10. If another JRE is currently installed as the system JVM and you are prompted to overwrite the current system JVM, click **NO**. Overwriting the current system JVM might cause applications depending on the current JVM to fail.
11. Click **Next** on the Start Copying Files window to start installing the JRE.
12. On the Browser Registration window, select the browsers that you want the IBM JRE to be associated with. These would normally be the browsers that you want to use with the browser client.
13. Click **Next**.
14. Click **Finish** to complete the installation.

Linux: Installing the IBM JRE

About this task

Complete the following steps to download the IBM JRE installer from the Tivoli Enterprise Portal Server and install the JRE on a Linux computer.

1. Start the browser on the computer to which you want to download the installer.
2. Enter the following URL in the **Address** field of the browser:
`http://teps_hostname:1920///cnp/kdh/lib/java
/ibm-java2-i386-jre-5.0-7.0.i386.rpm`
 where *teps_hostname* is the fully qualified host name of the computer where the portal server is installed (for example, myteps.itmlab.company.com).
3. When prompted, save the installer to disk.
4. Change to the directory where you saved the **ibm-java2-i386-jre-5.0-7.0.i386.rpm** file and launch the installer to start the installation program using the following command:
`rpm -ivh ibm-java2-i386-jre-5.0-7.0.i386.rpm`

You can also install the JRE without downloading the installer by supplying the URL to the rpm in the command:

```
rpm -ivh http://teps_hostname:1920///cnp/kdh/lib/java  
/ibm-java2-i386-jre-5.0-7.0.i386.rpm
```

Enabling tracing for the JRE

Log files are not created for the desktop client launched through Web Start unless you enable tracing for the JRE.

Before you begin

The logs for the Web Start client are located in a different place than logs for the browser client and for the desktop client installed from the media. On Windows computers, the logs for the Web Start client are located in the C:\Documents and Settings\Administrator\Application Data\IBM\Java\Deployment\log directory. On Linux and UNIX computers, the logs are located in the .java/deployment directory of the home directory of the user ID under which the Java JRE was installed. Java Web Start will create a uniquely named trace file for every independent launch of the application. The files are named javaws.*nnnnn*.trace, where *nnnnn* is an arbitrary five-digit identifier.

About this task

Complete the following steps to enable tracing:

1. Launch the IBM Control Panel for Java.
 - On Windows, select **Start > Control Panel**, then double-click IBM Control Panel for Java. You must switch to the Classic view to see and select the Control Panel. Alternatively, you can launch the Control Panel by selecting Start > Run > "C:\Program Files\IBM\Java50\jre\bin\javacpl.exe".
 - On Linux, change to `<install_dir>/jre/<platform>/bin` and run Control Panel: `./Control Panel`
2. Select the **Advanced** tab.
3. Expand the Debugging node in the **Settings** tree and check **Enable Tracing**.
4. Click **OK** to save the setting and close the Java Control Panel.

Downloading and running the desktop client

The Tivoli Enterprise Portal can be launched as a desktop application or as a web application. You have three ways to install the desktop application: from a browser by entering the URL of the Java Web Start client on the Tivoli Enterprise Portal

Server, launching the desktop client from the IBM Java Control Panel, or launching the desktop client using Java Web Start from the command line.

Before you begin

These are the basic instructions for downloading and running the desktop client using Java Web Start. The complete instructions, with configuration notes are given in the *Installation and Setup Guide*.

About this task

Complete one of these steps to install and launch the desktop client using Java Web Start:

- Enter the URL of the portal server in a browser:
 1. Start the browser on the computer where you want to use the desktop client.
 2. Enter the following URL in the **Address** field of the browser, where `<TEPS_host_name>` is the fully qualified host name of the computer where the Tivoli Enterprise Portal Server is installed.
`http://TEPS_host_name:1920///cnp/kdh/lib/tep.jnlp`
 3. Click **Run** on the security message.
 4. If you want to create a shortcut on your desktop for the Tivoli Enterprise Portal, click **Yes** when prompted. The desktop client starts and displays the logon window. If IBM Java 1.5 is not the system JVM, you cannot use this shortcut. You must create your own, as described in the topic on “Manually creating a shortcut for the Web Start client” in *Installation and Setup Guide*.
 5. Enter the user ID and password to log on to the Tivoli Enterprise Portal or click **Cancel** if you do not want to log on at this time. The default user ID is `sysadmin`.

If you set the RAS trace option for the Tivoli Enterprise Portal client as documented in *IBM Tivoli Monitoring: Troubleshooting Guide*, when you recycle the client the `kcjras1.log` should be created in the location where the client was launched. On Windows this defaults to `\Documents and Settings\<userid>\Desktop`.

- Launch the desktop client from IBM Java Control Panel:
 1. Launch the IBM Java Control Panel:

Windows In the Windows control panel, double-click **IBM Java Control Panel**. You must be in the Classic view to see **IBM Java Control Panel**.
Linux Change to `<install_dir>/jre/<platform>/bin` directory and enter `./Control Panel`.
 2. On the **General** tab, in the Temporary Internet Files section, click **Settings**. The Temporary Files Settings window is displayed.
 3. Click **View Applications**.
 4. On the **User** tab, select Tivoli Enterprise Portal, then click **Launch Online**.

Web Start downloads and starts the desktop client. When the application is launched, you can close the Control Panel windows.

- Launch the desktop client using Web Start from the command line:
 1. Open a command prompt and change to the directory where Web Start is installed.

Windows
`C:\Program Files\IBM\Java50\jre\bin`

Linux

```
<install_dir>/jre/<platform>/bin
```

2. Enter the following command, where *<TEPS_host_name>* is the fully qualified host name of the computer where the Tivoli Enterprise Portal Server is installed.

Windows

```
javaws http://<TEPS_host_name>:1920///cnp/kdh/lib/tep.jnlp
```

Linux

```
./javaws http://<TEPS_host_name>:1920///cnp/kdh/lib/tep.jnlp)
```

Web Start downloads and launches the desktop client.

Manually creating a shortcut for the Web Start client

About this task

On Windows, the Web Start executable file for the default Java JVM is copied to the Windows\System32 directory. When you let Web Start create a short cut for launching the desktop client, it uses the file in the System32 directory as the target. If the default JVM is not IBM Java 1.5, the shortcut will not launch the desktop client. You must create a shortcut manually.

To create a shortcut to use to launch the desktop client using Web Start:

1. Right-click on the Windows desktop and select **New > Shortcut** from the popup menu.
2. In the Create Shortcut window, type the following path or click **Browse** and navigate to the executable as shown:
C:\Program Files\IBM\Java50\jre\bin\javaws.exe
3. Click **Next** and type a name for the shortcut in the Select a Title for the Program window. For example:
ITM Web Start client
4. Click **Finish**.

The shortcut appears on your desktop.

Starting the desktop client on another portal server

When installing the desktop client, you designate a home Tivoli Enterprise Portal Server. If your monitoring environment has a multiple portal servers, you can define a separate desktop instance to point to another portal server.

Before you begin

The typical scenario for having multiple portal servers is where there is a test and production portal server, or where there are multiple managed networks with a portal server connected to each hub monitoring server.

About this task

Take these steps to create another portal client instance that connects to a different portal server.

1. On the computer where the desktop client is installed, start Manage Tivoli Monitoring Services:

- **Windows** Select **Start** → **Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**.
 - **Linux** **UNIX** Change directory (cd) to `<install_dir>/bin` and enter `./itmcmd manage`.
2. Right-click **Tivoli Enterprise Portal – Desktop** and click **Create Instance**. If other instances of the Tivoli Enterprise Portal have been created, you see more than one in the list. **Create Instance** is disabled for all but the original Tivoli Enterprise Portal instance.
 3. In the Tivoli Enterprise Portal window, enter a name to identify the instance and click **OK**.
 4. In the Configure Application Instance window, enter the host name of the Tivoli Enterprise Portal Server that you want to connect to.
 5. Click **OK**.

Results

The new Tivoli Enterprise Portal instance is added to the list.

What to do next

You can now start the instance at any time by double-clicking its entry.

If you no longer need a Tivoli Enterprise Portal instance, you can delete it: right-clicking the entry and click **Remove Instance**.

Starting the browser client on another portal server

Start a separate instance of your browser and log onto the portal server of a different managed network to see two managed networks from the same computer.

Before you begin

Your managed network can have one Tivoli Enterprise Portal Server and one hub Tivoli Enterprise Monitoring Server. You can log on to the portal server through the Windows Internet Explorer or Mozilla Firefox. If your organization has multiple managed networks, you can start a separate instance of the browser and log on to a different portal server from the same computer. There is no limit, other than the practical limit imposed by resources, to how many portal server environments you can manage from one workstation as long as you start a new instance of your browser for each portal server. You cannot log on to a portal server in a browser window and then, from the same window, log on to another portal server.

About this task

Before starting the browser client instances, take these steps on each computer where a portal server that you want to connect to is installed.

- **Windows**
 1. In the Manage Tivoli Monitoring Services window, right-click the **Tivoli Enterprise Portal Browser** entry and click **Reconfigure**.
 2. In the Configure Tivoli Enterprise Portal Browser window that opens, double-click the `cnf.browser.installdir` parameter.

3. In the Edit Tivoli Enterprise Portal Browser Parm window that opens, enter the path to the directory where the browser files should be installed, for example, C:\\temp\\cnpBrowserFiles.
4. Select the ☒ **In Use** check box and click **OK**.
5. Click **OK** to save your changes.

- **Linux**

1. Change to the directory where applet.html is located: `<itm_install_dir>/platform/cw`, where `platform` is the current type of operating system.
2. Open `applet.html` in a text editor.
3. Find the line, `<!--END OF PARAMS-->` and add a new line above it.
4. On the new line, add this parameter where `browser_install_dir` is the path to the directory where the browser files are installed.

```
document.writeln( '<PARAM NAME= "cnp.browser.installdir" VALUE="browser_install_dir">' )
```
5. Save and close `applet.html`.

What to do next

If you are using Internet Explorer, launch each instance of the Tivoli Enterprise Portal client that you want.

If you are using the Firefox browser, you need to create a separate profile for each instance that you intend to start. The Mozilla support site has a topic on Managing Profiles (<http://support.mozilla.com/en-US/kb/Managing+Profiles>) that you can refer to for help with setting up profiles. After creating the profiles, launch each instance with this command `<full_path_to_firefox> -p <profile_name> -no-remote`

Related reference

“Portal client parameter list” on page 30

Specifying the browser used for Launch Application and for online help

If you are running the desktop client on Linux, or you want to view the online help with some browser other than the default, specify to the portal server the location of the browser you want to use.

About this task

Complete these steps to specify a different browser to use for the online help and launch application:

- **Windows**

1. Launch Manage Tivoli Monitoring Services (**Start > (All) Programs > IBM Tivoli Monitoring > Manage Tivoli Monitoring Services**).
2. In the Manage Tivoli Monitoring Services window, right-click the browser or desktop client and select **Reconfigure**. The Configure the Tivoli Enterprise Portal Browser window is displayed. (If you are configuring the desktop client, the Configure Application Instance window is displayed.)
3. Scroll down in the list of variables until you see the `kjr.browser.default` variable.
4. Double-click `kjr.browser.default`. The Edit Tivoli Enterprise Portal Browser Parm window is displayed.

5. In the Value field, type the path and the application name of the alternative browser application. For example, C:\Program Files\Mozilla Firefox\firefox.exe
6. Click **OK** to close the editing window and save the change.
7. Click **OK** to close the reconfiguration window.

- Linux UNIX

1. Go to the *install_dir/bin/cnp.sh* and edit the *cnp.sh* shell script.
2. Add your Web browser location to the last line of the file. In the example below, the Web browser location is */opt/foo/bin/launcher*.
`-Dkjr.browser.default=/opt/foo/bin/launcher` The line is very long and has various options on it, including several other `-D` options to define other properties. It is very important to add the option in the correct place.

If the last line of your *bin/cnp.sh* originally looked like the following:

```
{JAVA_HOME}/bin/java -showversion -noverify -classpath ${CLASSPATH}
-Dkjr.trace.mode=LOCAL -Dkjr.trace.file=/opt/IBM/ITM/logs/kcjas1.log
-Dkjr.trace.params=ERROR -DORBTcpNoDelay=true -Dcnp.http.url.host=
-Dvbroker.agent.enableLocator=false
-Dhttp.proxyHost=
-Dhttp.proxyPort=candle.fw.pres.CMWAplet 2>& 1 >> ${LOGFILENAME}.log
```

To set the browser location to */opt/foo/bin/launcher*, change the line to look like the following:

```
{JAVA_HOME}/bin/java -showversion -noverify -classpath ${CLASSPATH}
-Dkjr.browser.default=/opt/foo/bin/launcher
-Dkjr.trace.mode=LOCAL -Dkjr.trace.file=/opt/IBM/ITM/logs/kcjas1.log
-Dkjr.trace.params=ERROR -DORBTcpNoDelay=true -Dcnp.http.url.host=
-Dvbroker.agent.enableLocator=false
-Dhttp.proxyHost=
-Dhttp.proxyPort=candle.fw.pres.CMWAplet 2>& 1 >> ${LOGFILENAME}.log
```

- **Java Web Start:**

Java Web Start deployed applications are described in *jnlp* deployment files. For IBM Tivoli Monitoring, there is one deployment file that describes the core framework component and associated jar files, and one deployment file for each and every Tivoli Enterprise Portal-based monitoring solution that is installed.

The core deployment file is named *tep.jnlp*. The application deployment file is typically called *kxx_resources.jnlp* or *kxx.jnlp*, where *xx* is the application identifier (a product code, such as *nt*, *ux*, or *lz*). On a Windows computer where the is installed, the file is located in *<itminstall_dir>\CNB* (for example, *c:\IBM\ITM\CNB*). On a Linux computer where the Tivoli Enterprise Portal Server

The deployment file instances are generated whenever the is installed or reconfigured (for example, when adding a new monitoring solution to the environment). The contents of these files are based upon two template deployment files (*.jnlp*). The core template deployment file is called *tep.jnlp*. The application template deployment file is named *component.jnlp*. On a Windows computer where the is installed, the file is located in *<itminstall_dir>\Config* (for example: *c:\IBM\ITM\Config*). On UNIX, the file is located in *<itminstall_dir>/config* (for example, */opt/IBM/ITM/config*).

In order to add or modify JVM arguments (such as maximum heap size) or other *-based* properties (such as *RAS1* trace options), it is necessary to edit either the *tep.jnlp* deployment file or the *tep.jnlp* deployment template file. The deployment file is nothing more than XML syntax that describes the Web Start application being deployed. The *<resources>* element is used to define the JVM arguments, the properties, jar files, and references to component deployment files.

- Modify the `tep.jnlp` file if the change will be temporary (for example, setting a trace option for gathering further diagnostics).
- Modify the `tep.jnlp.t` file if the change will be permanent (for example, increasing the maximum heap size to accommodate a larger monitored environment or increased event load).

If you modify the deployment template file, make sure you then reconfigure the in order to regenerate the instance-level `.jnlp` deployment files with your changes.

To specify the location of the browser to use to display the online help, add the following property to the `<resources>` section of the appropriate file:

```
<property name="kjr.browser.default" value="<path where browser is located>" >
```

Adding operating platforms to the Navigator

About this task

The Navigator Physical view in the Tivoli Enterprise Portal shows the operating platform below the enterprise level. The operating platform name is followed by the word *Systems* as in Linux Systems or z/OS Systems. Some operating platforms can be aggregated further. If your environment has such platforms and you want each to have its own Navigator item, with all systems of that type contained there, you can edit the `osnames` file to add them.

Secure Socket Layer transmissions

Information transmitted over a network from one component of Tivoli Management Services to another can be encrypted. Changes to the Secure Socket Layer (SSL) configuration for the Tivoli Enterprise Portal Server can be made at anytime.

Security is enhanced by the use of the Global Security Toolkit (GSKit) for SSL communications and the iKeyMan utility for security certificates. A default certificate and key is provided with your installation. If you prefer to have a self-signed certificate, use the iKeyMan utilities to create and load the certificate and key database. A stash file provides the database password for unattended operation.

Enabling TIP Web Service for Tivoli Integrated Portal charts

If your product is based on the Tivoli Integrated Portal infrastructure and you want to build Tivoli charts with values from your Tivoli Monitoring environment, enable the ITMWebService.

Before you begin

Single sign-on must be enabled for users of the Tivoli Integrated Portal administrative console and Tivoli Enterprise Portal.

About this task

Complete these steps on the computer where the Tivoli Enterprise Portal Server is installed.

1. Copy the `kfwtipewas.properties` file to the portal server directory:

 From `<itm_installdir>\CNPS\SQLLIB\` to `<itm_installdir>\CNPS.`

Linux or **UNIX** From `<itm_installdir>/<platform>/cq/sqllib/` to `<itm_installdir>/<platform>/cq/`.

2. Reconfigure the Tivoli Enterprise Portal Server.

What to do next

In the previous release you needed to have workspace administrator authority for the user who wanted to see query-based views. As a Tivoli Enterprise Portal user you are entitled to see certain workspaces in the portal that belong to a particular monitored application based on your permissions. If you are entitled to see, say, Linux workspaces in the portal, then those workspaces will be available in the Tivoli Integrated Portal.

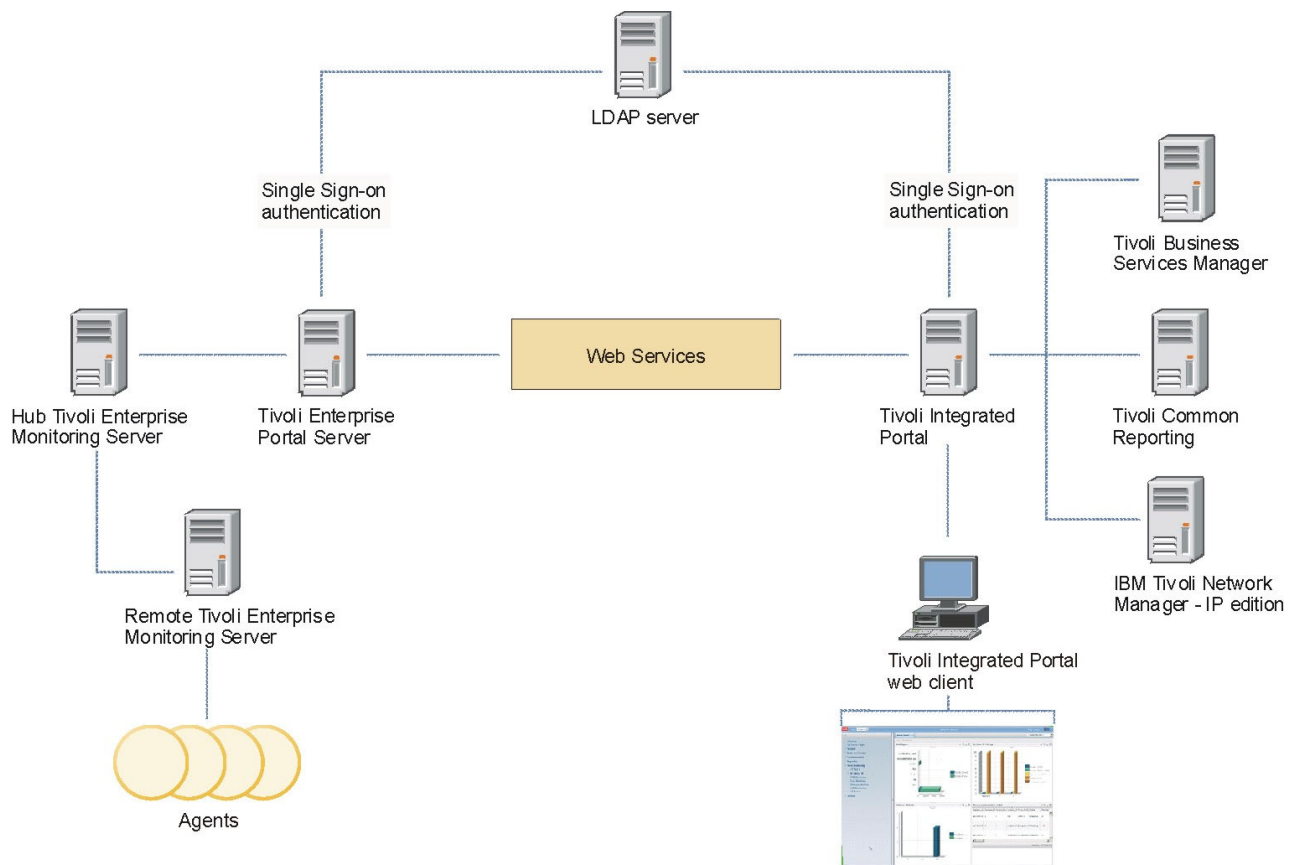


Figure 1. Tivoli Integrated Portal Web Services and the cross-product connections

Chapter 3. Editing the portal configuration settings

The Tivoli Enterprise Portal

Editing the portal client parameters

The portal client has parameters that affect its performance, such as the maximum size of files attached to event acknowledgements and for how long to keep the common event list in the cache.



Changes made to the browser client are applied globally because they are downloaded automatically through the HTTP server that is installed with the . If users are deploying the desktop client themselves through Java Web Start, the changes will also be applied globally. Otherwise, desktop client changes must be repeated on every computer where it is installed if the change should affect all users.

Editing client parameters

About this task

Changes you make to the browser client are applied globally because they are downloaded automatically through the HTTP server that is installed with the . If users are deploying the desktop client themselves through Java Web Start, the changes will also be applied globally. Otherwise, desktop client changes must be made on each computer where it is installed.

Complete these steps to adjust the client parameters:

1. Start Manage Tivoli Monitoring Services. For the browser client and Web Start, this is the computer where the is installed; otherwise, it is where the desktop client is installed.
 Click **Start** → **Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**.
 Change to the `<itm_install_dir>/bin` directory and enter: `./itmcmd manage`.
2. Right-click **Tivoli Enterprise Portal – Desktop** or **Tivoli Enterprise Portal – Browser**, and click **Reconfigure**.
The Configure Application Instance window is displayed for the desktop client (also used for Java Web Start); the Configure Tivoli Enterprise Portal Browser window is displayed for the browser client.
3. Double-click the parameter value you want to change. See → for a complete list with descriptions.
4. To activate the parameter, type a value and select **In Use** in the Edit Tivoli Enterprise Portal Parm window.
5. After you are finished editing the parameters, click **OK** to save your changes. Your changes will take effect the next time users log on to the . Users already logged on will see no change until they exit, and log on again.

Related reference

“Portal client parameter list”

Portal client parameter list

Most of the Tivoli Enterprise Portal client parameters are left unchanged from their default values. Edit the client parameters to effect a specific behavior.

Some parameters pertain only to the desktop client or browser client only and are noted as such.

browser.cache.memory.capacity

Indicates the maximum amount of memory in KB to be used to cache decoded images and other features by Browser views (a positive non-zero integer). Specify a value of 0 to disable memory caching. The default is -1 whereby the capacity value is automatically decided based on the total amount of memory.

Physical memory	Memory cache in KB
32 MB	2048
64 MB	4096
128 MB	6144
256 MB	10240
512 MB	14336
1 GB	18432
2 GB	24576
4 GB	30720
8 GB and beyond	32768

cnp.agentdeploy.timeout

This is the time that should pass before the agent deploy request times out. Default: 1800 seconds (30 minutes).

cnp.attachment.segment.maxsize

For transmission across the network, file attachments are broken into segments then reassembled at the Tivoli Enterprise Portal Server. For example, an 8 MB file is transmitted in eight segments of 1 MB. Adjust this parameter for the segment size that best suits your environment. Enter the maximum size in bytes, such as 250000 for 250 KB. Default: 1000000 (1 MB).

This parameter is also available as a portal server environment variable. See “Controlling the size of event attachments” on page 42.

cnp.attachment.total.maxsize

Use this parameter to set the maximum size of each file attached to an acknowledgement. Enter the maximum size in bytes, such as 2500000 for 2.5 MB. Default: 10000000 (10 MB).

This parameter is also available as a portal server

cnp.authentication.skip_dns

Value: “N”. This determines whether the server certificate validation tries to resolve and match the host DNS name.

cnp.browser.installdir

The WebRenderer API is used for browser view functionality in the Tivoli

Enterprise Portal. The first time a user creates a browser view, a

Windows `%HOMEPATH%\webrendererswing` or **Linux** **UNIX** `%HOME/.webrendererswing` subdirectory is created automatically on the user's computer. This subdirectory is where the browser jar files are extracted to and where certificates and other webrenderers artifacts are created for browser views. Use this parameter to specify a different path for the browser view files to be saved on user computers. A different path is required if users will be running multiple instances of the portal client and possibly logging on to different versions of the portal server.

cnp.commonevent.cache.timeout

Number of minutes to retain the cache for the common event console when the user has switched to a workspace that does not contain the common event console view (which means the cache is not being used). If this time period ends before the cache is used again, the cache is cleared. The cache is then rebuilt when it is needed by a common event console view.

A value of -1 means always retain the cache, even when it is not being used. A value of 0 means immediately clear the cache when the user has switched to a workspace that does not contain the common event console view. Default: 30.

cnp.databus.pageSize

In the portal user interface, the Properties editor has a field for adjusting the page size for individual query-based views. This parameter sets the number of rows to fetch in single logical page for all query-based views. Default: 100 rows. Although there is no limit to what you can set here, the larger the page size, the more memory required at the portal client and server.

You might, for example, want to set a larger page size for the searching in the table view over a larger number of rows. Or you might want fewer pages to scroll through when interacting with views that retrieve a large number of rows (or instances). You need to make sure, however, that you have sufficient resources on the portal client and server to handle the additional data being packaged, transported, and ultimately rendered as a result of increasing the page size value. Probably the best way to find the right number here is to increase it gradually (such as increments of 100) until response time across a good sampling of workspaces begins to suffer. At that point, you might want to reduce the number by the last increment (such as 100 rows fewer) as that will be close to the optimal value for the environment.

Another setting that affects query-based view response time is `KFW_REPORT_NODE_LIMIT`, which is a portal server environment variable.

cnp.drag.sensitivity

Number of pixels the mouse must move before drag operation begins. Default: 7.

cnp.encoding.codeset

String encoding code set identifier.

cnp.heartbeat.interval

Heartbeat ping interval between the portal client and server. An increase in the interval means that the client will take longer to detect when the is offline. A shorter interval means the client will be notified sooner but it also increases the traffic between client and server. Default: 30 seconds.

cnp.history.depth

Number of workspaces to maintain in the back / forward history navigation stack. Default: 20.

cnp.http.proxy.password

Desktop client only: Password used for proxy authentication using Browser view.

cnp.http.proxy.user

Desktop client only: Userid used for proxy authentication using Browser view.

cnp.http.url.host

Desktop client only: URL host for IOR fetch.

cnp.http.url.path

Desktop client only: URL path for IOR fetch.

cnp.http.url.port

Desktop client only: URL port for IOR fetch.

cnp.http.url.protocol

Desktop client only: URL protocol for IOR fetch.

cnp.http.url.DataBus

Desktop client only: The URL for the `cnps.ior` file, which is required for the to locate the graphic view images and style sheets. The default setting, which does not show, assumes the integral HTTP server. If it has been disabled for some reason, you must enter the URL for the integral HTTP server. See *IBM Tivoli Monitoring: Troubleshooting Guide* for details. When you set this parameter, it overrides the settings of the other `cnp.http.url` parameters for protocol, port, and path.

cnp.pipeline.factor

Databus to Server Pipeline monitoring factor (in Heartbeat cycles). Default: 2.

cnp.playsound.interval

Number of seconds before the same sound file can be played again. If events open frequently, this setting provides sound pause. Default: 10 seconds.

cnp.publishurl.delay

Browser mode only: When you make a workspace switch, allows the user interface rendering to complete before the browser initializes the new applet and destroys the old applet. Default: 1 second.

Important: Modify this parameter only after consulting IBM Software Support.

cnp.systemtray.offset

Tivoli Enterprise Portal factors in the Windows task bar at the bottom of the screen when sizing menus and windows for display. Default: true.

cnp.terminal.cache.entries

Maximum number of active terminal emulator sessions. Default: 50.

cnp.terminal.host

Default terminal emulator host name.

cnp.terminal.port

Default terminal emulator port number. Default: 23.

cnp.terminal.script.entries

Maximum number of user terminal emulator scripts that can be saved.
Default: 256.

cnp.terminal.type

Default terminal emulator type. When specifying a terminal type, enclose the terminal type within double quotes and enter one of these supported names:

IBM 3270 (24x80)
IBM 3270 (32x80)
IBM 3270 (43x80)
IBM 3270 (27x132)
IBM 5250 (24x80)
VT100 (24x80)

cnp.view.change_remove.warning

Warning message when the user is about to change or remove a view.

Default: True. The message is displayed. Change the setting to False to stop the message from being displayed.

cnp.workspace.switch.rate

The minimum amount of time that must pass before the workspace can be replaced by the next one selected. Default: 1000 (1 second).

cnp.workspace.render.delay

Browser mode only: Workspace post render delay in milliseconds.

http:agent

Defines the name of the integral HTTP server. If it or its proxy requires a different browser identity before it enables the browser view to access the Internet, you can enter a one-word name for the browser. It can be any name so long as it is not rejected by the proxy server. You normally do not need to add an http name definition unless users get an error when they attempt to access the Internet through a workspace browser view.

http.nonproxyhosts

When ☒ **Enable HTTP Proxy Server Requests** is selected, the servers in this list bypass the proxy. Separate each server name with a vertical line (|). See "Enabling the HTTP proxy server" on page 35.

http.proxyHost

Browser client: Used to specify the host name or IP address of the http proxy server if one is used.

http.proxyPort

Browser client: Used with the http.proxyHost parameter to specify the listening port number for the HTTP proxy server. Port 80 is the default for third-party HTTP servers.

kjr.browser.default

This is the path and name of the browser application to use when launching contextual help. To open the help with a specific browser or one other than the default, enter the path and the application name, such as C:\Program Files\Mozilla Firefox\firefox.exe.

kjr.trace.file

File name of RAS1 trace log if trace mode is LOCAL.

kjr.trace.mode

The RAS1 tracing option. Default: LOCAL.

kjr.trace.params

RAS1 trace options. Default: ERROR.

kjr.trace.qdepth

Sets the tracing thread queue depth to 15000 by default.

kjr.trace.thread

Determines whether trace calls are threaded. Default: true.

legacy_lifecycle

With Sun Java versions 1.6.0_10 or higher, a new plug-in architecture was introduced and established as the default plug-in. IBM Tivoli Monitoring browser clients do not run with this new plug-in architecture. To use the Sun 1.6.0_10 (or higher) JRE, set this parameter to **true**. You will also need disable the *next-generation Java plug-in* on the computer where the browser client is being run: Launch the Java Control Panel for the Sun JRE. In the **Advanced** tab, expand the **Java Plug-in** branch. Clear the ☐ **Enable the next-generation Java Plug-in (requires browser restart)** check box.

sun.java2d.noddraw

When the Tivoli Enterprise Portal is run as a client image in an emulation environment that does not support the DirectDraw screen-writing function, turn off the function by setting this variable to true in both the browser and desktop clients. Otherwise, users encounter conditions of high CPU usage because the Java process attempts to write to the screen. Default: true.

user.language

Specifies the language code of the user's locale preference (de, en, es, fr, it, ja, ko, pt, zh). As well as the language, the time, date, currency, and number separator formats are converted for the locale. You can create another instance of the desktop client and change this variable (and user.region) to another locale. In this way, you can have two or more instances of the desktop client running on the same computer, each in a different language. If you specify an unsupported locale, the failover is to en_US. Browser mode users can enter the text below directly into their Java plug-in runtime parameters if they do not want to change these environment variables or their operating system locale.

-Duser.language=xx

-Duser.region=XX

where xx is the language and XX is the locale: de_DE, en_US, en_GB, es_ES, fr_FR, it_IT, ja_JP, ko_KR, pt_BR, zh_CN, and zh_TW (such as pt_BR for Brazilian Portuguese and zh_TW for Traditional Chinese).

Note: The portal client uses cascading style sheets to render the application text. If no localized version of a style sheet, such as ws_press.css, is available, the English version will be used.

user.region

Specifies country code of user's locale preference (DE, US, UK, ES, FR, IT, JA, KR, BR, CN, TW). See also the description for user.language.

Related tasks

“Editing client parameters” on page 29

“Controlling the size of event attachments” on page 42

“Starting the browser client on another portal server” on page 23

Related reference

“Portal server environment variables” on page 38

Enabling the HTTP proxy server

Environments that use an HTTP proxy server require additional client configuration to enable URL access from the browser view in a Tivoli Enterprise Portal workspace.

About this task

To enable the HTTP proxy server, complete these steps on every computer where the Tivoli Enterprise Portal client is used that also uses an HTTP proxy for the browser view:

1. Open a workspace that contains a browser view or add a browser view to the current workspace.
2. In the browser view's address box, type: `about:config`
3. In the filter field that appears at the top of the page, enter the following to see the network proxy fields: `network.proxy`
4. Out of the reduced set shown, the following three entries are of interest. Double-click an entry or select it and press Enter to modify its values:

network.proxy.http

Enter the DNS identifier or the IP address of the proxy host to use for the HTTP protocol.

network.proxy.http_port

Enter 80, the default port number, or a different number used by the proxy host.

network.proxy.no_proxies_on

Append any fully qualified host names or IP addresses that should be accessed without the proxy. For example, this setting bypasses the proxy server for any files on your local system and on the portal server (`myteps.uk.ibm.com`) that are accessed from the browser view:
`localhost,127.0.0.1, myteps.uk.ibm.com`.

Results

After you click **OK** on the property edit panel, the change is saved on the Tivoli Enterprise Portal client.

Setting application properties for Linux and UNIX

About this task

To change a property such as the location of the Web browser that the Tivoli Enterprise Portal browser client launches in UNIX, update the shell script file or files that are run and the template that is used when the browser client is configured to create the script file or files that are run. You might have to update one or more of the following files:

Note: All file paths are relative to your *install_dir* directory where you installed IBM Tivoli Monitoring.

Table 1. File locations for changing application properties for UNIX and Linux

File location	Purpose of file
bin/cnp.sh	The default shell script that launches the Tivoli Enterprise Portal browser client.
bin/cnp_instance.sh	The shell script for a specific instance you have created, where <i>instance</i> is the name of the instance that launches the Tivoli Enterprise Portal browser client.
platform/cj/original/cnp.sh_template	<p>The template from which the bin/cnp.sh and bin/cnp_instance.sh shell scripts are generated during configuration, where <i>platform</i> is the code for the operating system platform on which IBM Tivoli Monitoring is installed. For example: <i>li6243</i> for Linux 2.4 on a 32-bit Intel® CPU).</p> <p>If you only change bin/cnp.sh or bin/cnp_instance.sh and do not change this template, the next time you configure the client, a new version of the script is created without the changes you made to bin/cnp.sh or bin/cnp_instance.sh.</p>

To change the location of the Web browser you must change the above file or files to include a new property:

1. Go to the `<itm_install_dir>/bin/cnp.sh` and edit the `cnp.sh` shell script.
2. Add your Web browser location to the last line of the file. In the example below, the Web browser location is `/opt/foo/bin/launcher`.
`-Dkjr.browser.default=/opt/foo/bin/launcher`

Important: The line is very long and has various options on it, including several other `-D` options to define other properties. It is very important to add the option in the correct place.

If the last line of your `bin/cnp.sh` originally looked like the following:

```
{JAVA_HOME}/bin/java -showversion -noverify -classpath ${CLASSPATH}
-Dkjr.trace.mode=LOCAL -Dkjr.trace.file=/opt/IBM/ITM/logs/kcjas1.log
-Dkjr.trace.params=ERROR -DORBtcpNoDelay=true -Dcnp.http.url.host=
-Dvbroker.agent.enableLocator=false
-Dhttp.proxyHost=
-Dhttp.proxyPort=candle.fw.pres.CMWApplet 2>& 1 >> ${LOGFILENAME}.log
```

To set the browser location to `/opt/foo/bin/launcher`, change the line to look like the following:

```
{JAVA_HOME}/bin/java -showversion -noverify -classpath ${CLASSPATH}
-Dkjr.browser.default=/opt/foo/bin/launcher
-Dkjr.trace.mode=LOCAL -Dkjr.trace.file=/opt/IBM/ITM/logs/kcjas1.log
-Dkjr.trace.params=ERROR -DORBtcpNoDelay=true -Dcnp.http.url.host=
-Dvbroker.agent.enableLocator=false
-Dhttp.proxyHost=
-Dhttp.proxyPort=candle.fw.pres.CMWApplet 2>& 1 >> ${LOGFILENAME}.log
```

You can also set instance name, browser, and Tivoli Enterprise Portal Server properties on Linux. Refer to the *Command Reference* for details.

Setting the environment variable when the hub is on z/OS

About this task

On z/OS, GSKit is known as the Integrated Cryptographic Service Facility, or ICSF. The monitoring server supports secure password encryption through ICSF, which provides a robust encryption and decryption scheme for stored passwords and is the preferred method of password encryption. (See *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS* Configuring IBM Tivoli Enterprise Monitoring Server on z/OS.)

If the hub is on a z/OS system that does not have ICSF installed, an alternative, less secure encryption scheme is used. The hub monitoring server and the portal server both must be using the same scheme. Therefore, if the hub system does not use ICSF, you must configure the Tivoli Enterprise Portal to use the less secure scheme (EGG1) as well. This involves editing the Tivoli Enterprise Portal Server

To add the new line to the environment file, complete the following steps:

Windows

1. On the system where the Tivoli Enterprise Portal Server is installed, select **Start** → **Programs** → **IBM Tivoli Monitoring vManage Tivoli Monitoring Services**.
2. Right-click Tivoli Enterprise Portal Server, point to **Advanced** and select **Edit ENV File** from the list.
3. If the Tivoli Enterprise Portal Server message displays, click **OK** to close it.
4. Add a new line: `USE_EGG1_FLAG=1`.
5. Click **Save**.
6. Click **Yes** to implement your changes and recycle the service.

Results

Linux OR UNIX

1. Change directory (cd) to `<itm_install_dir>/config`
2. Add the following line to the `cq.ini` file:
`USE_EGG1_FLAG=1`
3. Save the file.
4. Recycle the Tivoli Enterprise Portal Server.

Editing the environment configuration

The Tivoli Enterprise Portal Server runs a process called KfwServices, which has a set of environment variables that can be edited and enabled for certain configuration requirements. This can be done through the Manage Tivoli Monitoring Services application or at the command line using `itmcmd manage`. See *Command Reference*.

When you have security enabled, you can control the number of log in attempts before a user is locked out of the portal.

Several environment variables are used to control the way that event information is stored on the portal server.

If you want to set the application properties for advanced configuration functions on UNIX or Linux®, such as the location of the Web browser that the Tivoli Enterprise Portal browser client launches, this has to be done manually.

If the Tivoli Enterprise Portal Server connects to a monitoring server on a z/OS system that does not have the Integrated Cryptographic Service Facility (ICSF) installed, you need to edit the environment file to add a new line.

Editing the portal server environment file

Edit the Tivoli Enterprise Portal Server environment file, KFWENV to reconfigure the portal server parameters.

About this task

Take these steps to edit the portal server environment file:

1. Open the environment file on the computer where the portal server is installed:
 - **Windows** From Manage Tivoli Monitoring Services (**Start** → **Run** → **Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**), right-click **Tivoli Enterprise Portal Server** and click **Advanced** → **Edit ENV File** to open the kfwenv file.
 - **Linux** **UNIX** Change to the `<install_dir>/config` directory and open the `cq.ini` file in a text editor.
2. Edit the file to enable (delete # at the beginning of the line), disable (type # at the beginning of the line) or modify any of the environment variables.
3. Save kfwenv (Windows) or cq.ini (Linux and operating systems such as UNIX) and exit the text editor.
4. Click **Yes** when a message asks if you want to recycle the service. Click **No** if you prefer to have the changes you made take effect later by manually recycling the portal server.

Related reference

“Portal server environment variables”

Portal server environment variables

The environment configuration file for the Tivoli Enterprise Portal Server can be edited to add certain environment settings and to change the values of others.

The file shows a number of environment variables that have been enabled and others that are disabled by default or as a result of the way you configured the . Other variables in this list must be added manually to enable them.

KFW_AUTHORIZATION_MAX_INVALID_LOGIN=0

You can control the number of attempts a user can make to log on to the portal server by setting this environment variable to a value from 0 to 15. The default value, 0, indicates that there is no limit to the number of failed attempts a user can make before being locked out.

This configuration setting is effective only when you have enabled security through the hub monitoring server as described in the topic, “Controlling the number of logon attempts” on page 40.

KFW_CMW_DETECT_AGENT_ADDR_CHANGE=N

The Navigator function detects when the IP address for an agent is discovered. If the agent environment is constantly changing or has improper configurations that generate excessive Navigator tree rebuilding,

consider adding this environment variable to have any discovery of changes or additions of IP address ignored.

KFW_CMW_DETECT_AGENT_HOSTNAME_CHANGE=N

This variable is like the one for detect agent address change except that it prevents the Navigator rebuilding if an agent hostname is changed.

KFW_CMW_DETECT_AGENT_PROPERTY_CHANGE=N

This is like the detect agent address change except that it prevents the Navigator rebuilding if an agent affinity or affinity version changes.

KFW_CMW_SITUATION_ADMIN_SUPPRESS=N

When a situation is stopped, no message is sent to the situation event console. If you prefer to have the message written to the situation event console for each system the situation was distributed to, enable (remove the # at the beginning of the line) this environment variable. The Stopped message alerts the user that the situation has been stopped, thus, its state is unknown.

KFW_CMW_SPECIAL_HUB_ENTERPRISE

Associates situations to the .

KFW_ECLIPSE_HELP_SERVER_PORT=9999

The default port number for the Eclipse help server is 9999. If 9999 is already used by another device, add this variable and specify an port number from 1 to 65535. This value will be passed as a property from the to the client at logon time.

KFW_FIPS_ENFORCED=N

The monitoring server and agent components of the Tivoli Management Services are already FIPS compliant. This variable specifies whether the encryption methods used by the portal server should comply with the Federal Information Processing Standard (FIPS) 140-2 specification. If your environment must conform to the FIPS 140-2 standard, specify Y.

KFW_REPORT_NODE_LIMIT=200

When a workspace that contains a query-based view is opened or refreshed, the view's query requests data from the managed systems that are assigned to that Navigator item (unless you have edited the view's query definition to assign specific managed systems or managed system lists). The number of managed systems from which a query can retrieve data can be up to 200. This limitation is provided to keep traffic and resource usage of your managed environment at an acceptable level. You can adjust the maximum number with this variable but keep in mind that if you increase the maximum number of managed systems being queried, the longer it can take to render the view.

Consider creating filtered queries, managed system lists, or custom Navigator views with managed systems assignments on Navigator items that limit the number of managed systems to retrieve data from. These features are described in the online help and user's guide.

Another setting that affects query-based view response time is the `cnp.databus.pageSize` client parameter.

KFW_REPORT_TERM_BREAK_POINT=86400

Adjust this setting to change the point, in seconds, where a historical request selects from short-term or long-term history data. The default is for short-term history data to be collected from *now* to 24 hours ago, and long-term from 24 hours onward. Set to 0 to select only from long-term history data.

Related tasks

“Editing the portal server environment file” on page 38

Related reference

“Portal client parameter list” on page 30

Controlling the number of logon attempts

About this task

You can specify the number of attempts a user can make to log into the by setting the following environment variable in the kfwenv file (on Windows) or cq.ini (on operating systems such as UNIX):

KFW_AUTHORIZATION_MAX_INVALID_LOGIN=0

Specify a value between 0 and 15. The default value, 0, indicates that there is no limit to the number of failed attempts a user can make before they are locked out.

☑ Security: Validate User

The invalid login setting is effective only when you have enabled security through the hub monitoring server .

UNIX You must also enable the Login Lockout feature by turning on the validation setting in the monitoring server configuration file: KDS_VALIDATE_EXT="Y". The monitoring server configuration file is `<itm_install_dir>/config/<hostname>_ms_<address>.config`,

The monitoring server configuration file are named `<hostname>_ms_<address>.config` and `ms.ini`, and are located in the `<itm_install_dir>/config` directory.

Restoring user access

If a user is locked out, you have two options to restore their access to the :

- In the Tivoli Enterprise Portal
- On the computer where the is installed, run this command line utility to enable or disable access:

Windows Change directory to `<itm_install_dir>\cnps\` and enter
KfwAuthorizationAccountClient.exe ENABLE|DISABLE
user_id

For example, `KfwAuthorizationAccountClient.exe disable guest01` locks out the guest01 user until you re-enable the user ID.

Linux or **UNIX** Change directory to `<itm_install_dir>/bin` and enter
./itmcmd execute cq "KfwAuthorizationAccountClient
enable|disable user_name"

You can also use either of these procedures to disable a user from accessing the portal, regardless of the KFW_AUTHORIZATION_MAX_INVALID_LOGIN setting.

Reducing processing load on the portal server

If the portal server process kfwservices nears its memory capacity due to a large volume of data returning from hundreds or thousands of managed systems, you can reduce the memory usage.

Before you begin

Through the hub the receives data from any number of managed systems in the environment serviced by the hub. The process embeds some Java logic that can be externalized in a separate process, which has the effect of reducing memory usage of the process.

About this task

Take these steps to provide Java Virtual Machine (JVM) functionality in a separate process from the :

- **Windows** Locate the `kfwewas.properties` file in the `<install_dir>\cnps\sqllib` directory.
 1. Copy `kfwewas.properties` to the parent directory: `<install_dir>\cnps`
 2. In Manage Tivoli Monitoring Services, right-click the and click **Reconfigure**.
 3. Click **OK** for the two windows that display; if you are asked to configure the Tivoli Data Warehouse, click **No**.
 4. Start the .
- **Linux** or **UNIX**
 1. Locate the `kfwewas.properties` file in the `<install_dir>/platform/cq/sqllib` directory.
 2. Copy `kfwewas.properties` to the parent directory `<install_dir>/platform/cq`.
 3. In Manage Tivoli Monitoring Services, right-click the Tivoli Enterprise Portal Server and click **Configure**.
 4. Close the configuration windows.
 5. Start the Tivoli Enterprise Portal Server.

Duper

Event management configuration

Event pruning

About this task

Event information is stored in the KFW tables in the portal server database. Because this information can grow in the amount of space it consumes, it is automatically pruned.

By default, closed events are removed from the database one day after they are closed, within the hours of 12:00 AM and 4:00 AM on the local portal server.

You can control the pruning of this data by changing the following environment variables in the KFWENV configuration file:

KFW_PRUNE_START = *hh:mm*

The time of day to start pruning data, specified in 24-hour notation. For example, to begin pruning data at 11:00 PM, specify 23:00.

KFW_PRUNE_END = *hh:mm*

The time of day to stop pruning data, specified in 24-hour notation. For example, to end pruning data at 1:00 AM, specify 01:00.

KFW_EVENT_RETENTION = *d*

The number of days to keep a closed event. For example, to prune an event 2 days after it is closed, specify 2.

Controlling the size of event attachments

About this task

By default, the maximum size of each file attached to an event acknowledgement is 10 MB, and 1 MB for the size of information segments sent across the network. Environment variables are provided that enable you to change the maximum at the portal client or at the . The event attachment settings that are changed at the desktop client override those for the .

The parameters to change on the client are **cnp.attachment.total.maxsize** and **cnp.attachment.segment.maxsize**.

The environment variables to set on the are commented out in the kfwenv file:

```
#-----  
# Sample configuration for event attachments  
#  
#KFW_ATTACHMENT_MAX=10000000  
#KFW_ATTACHMENT_SEGMENT_MAX=1000000
```

Open the kfwenv file as described in Delete the # pound symbol at the beginning of the two KFW_ATTACHMENT lines and edit the settings as needed. Editing the portal server

Related reference

“Portal client parameter list” on page 30

Changing the KFWENV environment variables:

About this task

To change the attachment configuration variables at the monitoring environment level, edit the KFWENV configuration file and edit the following variables:

KFW_ATTACHMENT_MAX = *n*

Specify the new maximum file attachment size. The default value is 10 MB.

KFW_ATTACHMENT_SEGMENT_MAX = *n*

Specify the new maximum size for file segments. The default value is 1 MB.

Save and close the file.

Federal Information Processing Standard enablement

About this task

Chapter 4. Setting up asymmetric encryption

Setting up asymmetric encryption through the use of public-private key files requires the following steps:

Table 2. Setting up asymmetric encryption

Task
Create a new key database.
Create a request for a new public-private key pair and send that request to a trusted Certificate Authority.
Optionally use a temporary, self-signed key pair while you wait for your CA-signed certificate.
Add the CA-signed digital certificate to your key database.
Enable components to access the certificate by saving the key database password to a stash file on your computer.

For additional information on these procedures, see the IKeyMan user guide on IBM developerWorks®

Important: If you do not use the recommended names for the key database, stash file, and certificate label as described below, you must change the following environment variables in the KFWSERVICES file on the portal server:

- KDEBE_KEYRING_FILE=C:\IBM\ITM\keyfiles\keyfile.kdb
- KDEBE_KEYRING_STASH=C:\IBM\ITM\keyfiles\keyfile.sth
- KDEBE_KEY_LABEL=IBM_Tivoli_Monitoring_Certificate

Setting the JRE for GSKit and starting Key Manager

About this task

You need to set the path to the Java Runtime Environment before starting GSKit. Otherwise, you will get an error like "Failed to parse JAVA_HOME setting".

- **Windows**
 1. From the command prompt, run this script to get the IBM Java location:
`<install_dir>\Install\ITM\GetJavaHome.bat`
 2. Set the JAVA_HOME variable to point to the IBM Java location.
 3. Get the GSKit location by running this script:
`<install_dir>\Install\ITM\GetGSKitHome.bat`
 4. Change the directory to GSKit path\bin and run this command:
`gsk7ikm.exe`
 - 5.
- **Linux UNIX**
 1. From the console, run this script to get the IBM Java location:
`<install_dir>/bin/CandleGetJavaHome.sh`
 2. Export variable JAVA_HOME to point to the IBM Java path. For 64-bit, the gsk7ikm has to be 64-bit Java.
 3. Check the path for a local GSKit by looking in this file:
`<install_dir>/config/gskit.config`

GskitInstallDir points to a 32-bit GSKit and GskitInstallDir_64 points to a 64-bit GSKit.

4. Start GSKit Key Manager by running the command that corresponds to your system:
 - **HP 32-bit:** GskitInstallDir/bin/gsk7ikm_32
 - **Linux, Aix, or Solaris 32-bit:** GskitInstallDir/bin/gsk7ikm
 - **64-bit:** GskitInstallDir_64/bin/gsk7ikm_64

Creating a new key database

About this task

Use the following steps to create a new key database:

1. If you have not already done so, start iKeyman.
2. Click **Key Database File** → **New**.
3. Select **CMS** in the **Key database type** field.
4. Type keyfile.kdb in the **File Name** field.
5. Type the following location in the **Location field**: *<itm_installdir>/keyfiles*.
6. Click **OK**. The Password Prompt window is displayed.
7. Enter a password in the **Password** field, and confirm it again in the **Confirm Password** field. Click **OK**.
8. A confirmation window is displayed. Click **OK**.

The IBM Key Management window is displayed. This window reflects the new CMS key database file and your signer digital certificates.

Creating a new public-private key pair and certificate request

About this task

Use the following steps to create a new public-private key pair and certificate request:

1. If you have not already done so, start iKeyman.
2. Click **Key Database File** → **Open**.
3. Select the keyfile.kdb key database and click **Open**.
4. Type the password for the key database and click **OK**.
5. Select **Personal Certificate Requests** from the pull-down list and click **New**.
6. Click **New**.
7. Type IBM_Tivoli_Monitoring_Certificate in the **Key Label** field.
8. Type a **Common Name** and **Organization**, and select a **Country**. For the remaining fields, either accept the default values, or type or select new values.
9. At the bottom of the window, type a name for the file.
10. Click **OK**. A confirmation window is displayed, verifying that you have created a request for a new digital certificate.
11. Click **OK**.

The IBM Key Management window is displayed.

Send the file to a CA to request a new digital certificate, or cut and paste the request into the request forms on the CA's Web site.

Using a temporary self-signed certificate

About this task

It can take between two and three weeks to receive a CA-signed digital certificate. If you want to use a digital certificate other than the one provided with IBM Tivoli Monitoring and you have not yet received the CA-signed digital certificate, you can create a self-signed certificate on the portal server. A self-signed digital certificate is not as secure as a CA-signed certificate; this is strictly a temporary measure until the CA-signed certificate arrives.

Creating and using a self-signed certificate involves the following steps:

1. Create a CA key database.
2. Create the self-signed certificate.
3. Export the self-signed certificate.
4. Receive the self-signed certificate into the key databases on the portal server.

When you receive the CA-signed certificate, you need to delete the self-signed certificate.

Receiving the CA-signed certificate

About this task

After the CA returns your new digital certificate, save it on the computer where the portal server is running. Repeat for the client. If the CA returns the certificate as part of an e-mail message, copy and paste it from the e-mail into a text file.

Use the following steps to receive the digital certificate from the CA into key database on each computer.

1. If you have not already done so, start iKeyman.
2. Click **Key Database File** → **Open**.
3. Select the keyfile.kdb database and click **Open**.
4. Type the password for the database and click **OK**.
5. Select **Personal Certificates** from the pull-down list.
6. Click **Receive**.
7. Click **Data type** and select the data type of the new digital certificate, such as **Base64-encoded ASCII data**.
8. Type keyfile.sth for the **Certificate file name** and `<itm_installdir>/keyfiles` as the **Location** for the new digital certificate.
9. Click **OK**.
10. Type IBM_Tivoli_Monitoring_Certificate for the new digital certificate and click **OK**.

Save the password to a stash file

About this task

Because many of the IBM Tivoli Monitoring components work without user intervention, you need to save the key database password to a stash file on your computer. This enables the components to use SSL without requiring any intervention from you. Use the following steps to save the password to a stash file:

1. If you have not already done so, start iKeyman.
2. Select **Key Database File → Stash File**.
An information window is displayed telling you that the password was saved to a stash file.
3. Click **OK**.

Chapter 5. Enabling user authentication

Access to the Tivoli Enterprise Portal is controlled by user accounts defined to the portal server. In addition to defining the user IDs that are authorized to log onto the Tivoli Enterprise Portal, these accounts define the permissions that determine the Tivoli Enterprise Portal features a user is authorized to see and use, the monitored applications the user is authorized to see, and the Navigator views (and the highest level within a view) the user can access.

An initial **sysadmin** user ID with full administrator authority is provided at installation so you can log in to the Tivoli Enterprise Portal and add more user accounts. No password is required to log on to the Tivoli Enterprise Portal

User authentication can be enabled through either the hub or the Tivoli Enterprise Portal Server

User IDs authenticated through the hub monitoring server can be authenticated by either the local system registry or an external LDAP-enabled central registry. User IDs authenticated through the can be authenticated only by an external LDAP-enabled registry. User IDs that need to make SOAP Server requests (including user IDs that issue CLI commands that invoke SOAP server methods) must be authenticated through the hub monitoring server. User IDs that require single sign-on (SSO) capability must be authenticated through the portal server. LDAP authentication must be enabled through the before SSO can be configured.

Note: LDAP authentication is not supported for hub monitoring servers on z/OS.

Table 3. Where to configure LDAP authentication

	Hub monitoring server	Portal server
LDAP used for SOAP requests	X	
LDAP used for SSO		X
LDAP used with other registries		X

You can configure the and the Tivoli Enterprise Portal Server

If your hub monitoring server has already been configured to authenticate users, and you now want to migrate to authentication through the using the same LDAP registry, see “Migrating authentication from the monitoring server to the portal server” on page 62.

Configuring user authentication through the hub monitoring server

The following table lists the tasks required to configure authentication through the hub monitoring server and provides links to the required information.

Table 4. Tasks required for enabling user authentication through the hub monitoring server

Task
Perform all prerequisite tasks and obtain the information required during configuration.

Table 4. Tasks required for enabling user authentication through the hub monitoring server (continued)

Task
Configure authentication through the hub monitoring server.

ldapsearch is a command-line tool you can use to verify LDAP information before configuration and to troubleshoot problems you encounter with during configuration.

Prerequisites for configuring authentication on the hub monitoring server

About this task

This table lists the tasks that should be completed before user authentication is enabled on the hub monitoring server and provides links to the information needed to carry out those tasks.

Table 5. Tasks to complete before configuring authentication.

Task	Where to find information
Set up user accounts.	"Adding a user ID" on page 71
Set up user accounts in the authenticating registry.	See the documentation for setting up user accounts on the local operating system or LDAP directory server. For information on setting up users on z/OS, see <i>IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS</i> .
To use SSL (Secure Socket Layers) for communication between the hub and an LDAP server, set up a CMS key store and key store stash using GSKit and to import any required certificates.	Chapter 4, "Setting up asymmetric encryption," on page 43

If you intend to authenticate using the hub monitoring server, make sure that user accounts for the log-in IDs are set up in the authenticating registry before authentication is enabled. At a minimum, add the **sysadmin** user ID to the local registry on the hub computer, so that **sysadmin** can log in after authentication has been enabled.

Note: On Windows, the installer creates a **sysadmin** user account in the Windows registry and asks you to specify a password for that ID. The password is not required unless password authentication is enabled.

Tip: The Windows installer does not set the "Password never expires" option when it creates the **sysadmin** account. If you do not set this option, the password will expire according to the security policy on the hub computer, and you will not be able to log in to the portal server. Use the Windows Administrative Tools to ensure that the "Password never expires" option is selected for the **sysadmin** user account.

Before you enable authentication, obtain the following information

- If you are using an external LDAP server for authentication, obtain the information shown in this table from the LDAP administrator before configuring user authentication.

Table 6. LDAP configuration parameters

Parameter	Description
LDAP User Filter	<p>The attributes used to map Tivoli Enterprise Portal user IDs to LDAP log-in IDs. The attribute must contain the same name as the Tivoli Enterprise Portal log-in ID. The portal user ID will usually become the “ in the LDAP user filter. For example:</p> <p>IBM Tivoli Directory Server: (&(mail=%v@yourco.com) (objectclass=inetOrgPerson)) Microsoft Windows Active Directory: (&(mail=%v@yourco.com) (objectclass=user)) Sun Java System Directory Server: (&(mail=%v@yourco.com) (objectclass=inetOrgPerson))</p> <p>Not all LDAPs have the mail attribute for the person. For example, the LDAP administrator might only set the common name, in which case the filter would look like the following: (&(cn=%v) (objectclass=inetOrgPerson))</p> <p>The administrator should verify exactly which LDAP attribute must be used to search for the user. With Active Directory, for example, the cn equals the Full Name of the Active Directory user, and this <i>must</i> be exactly the same as the Tivoli Monitoring user, and cannot have spaces (for example, "S Smith" must be "SSmith").</p>
LDAP base	<p>The LDAP base node in the LDAP repository that which is used in searches for users. For example:</p> <p>IBM Tivoli Directory Server: dc=yourdomain,dc=yourco,dc=com Microsoft Windows Active Directory: dc=yourdomain,dc=yourco,dc=com Sun Java System Directory Server: dc=yourdomain,dc=yourco,dc=com</p>
LDAP bind ID	<p>The LDAP user ID for bind authentication, in LDAP notation. This LDAP user ID must be authorized to search for LDAP users. This value can be omitted if an anonymous user can search for LDAP users.</p>
LDAP bind password	<p>The password for LDAP bind authentication. This value can be omitted if an anonymous user can bind to your LDAP server. This value is encrypted by the installer.</p>
LDAP host name	<p>The LDAP server host name. This value can be omitted if your LDAP server is on the same host as the Tivoli Enterprise Monitoring Server. (The default is localhost.)</p>
LDAP port number	<p>The LDAP server port number. This value can be omitted if your LDAP server is listening on port 389.</p>

- If you intend to use SSL communication between the hub monitoring server and the LDAP server, obtain the information described in this table..

Table 7. SSL parameters for communication between hub and LDAP server

Parameter	Description
LDAP key store file	<p>The location of GSKit key store data base file. You can specify any location. For example: C:\IBM\ITM\keyfiles</p>
LDAP key store stash	<p>The location of the GSKit database password file. For example: C:\IBM\ITM\keyfiles\keyfile.sth</p>
LDAP key store label	<p>The key store label. For example: IBM_Tivoli_Monitoring_Certificate</p>
LDAP key store password	<p>The password required to access the key store.</p>

Configuration procedures

Configure user authentication on the Windows-, Linux-, or UNIX-based hub monitoring server.

About this task

For instructions for configuring authentication on a hub monitoring server installed on z/OS, see *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS*. Authentication by an external LDAP registry is not supported on a z/OS hub.

Windows: Configuring user authentication through the hub About this task

Complete the following steps to configure a hub monitoring server on Windows to authenticate users:

1. Select **Start** → **Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**
2. Right-click the hub monitoring server and select **Reconfigure**.
3. In the configuration window that displays, select **Security: Validate User**. The option **LDAP Security: Validate User with LDAP** becomes available.
4. If you want to use LDAP for user authentication, check the **Validate User with LDAP** option, then click **OK** to open the LDAP window. If you want to use the local registry, skip to step 7.
5. Specify the required LDAP values as appropriate for your site.
6. If you want to use SSL to secure communications between the hub and the LDAP server, check **LDAP SSL Communications: Use SSL?** and provide the appropriate values. If required, provide a password for the keystore.
7. Click **OK** The Hub TEMS Configuration window is displayed.
8. Click **OK** to accept the current settings.
9. In the Manage Tivoli Monitoring Services window, restart the hub monitoring server by right-clicking its name and selecting **Start**.

Linux or UNIX: Configuring user authentication through the hub

Configure user authentication for an environment in which the hub is installed on Linux or UNIX.

Configuring user authentication from the command line:

About this task

To configure the hub from the command line:

1. Change to the *install_dir/bin* directory and run the following command:

```
./itmcmd config -S -t tems_name
```

where *install_dir* is the installation directory for IBM Tivoli Monitoring, and *tems_name* is the name of the hub monitoring server. The default installation directory on Linux or UNIX is */opt/IBM/ITM*.

You see the following prompt:

Configuring TEMS...

2. Accept the defaults for the following prompts. The defaults should reflect the selections made during installation.
3. When you see the prompt:
Security: Validate User?

type 1 and press Enter.

4. If you do not want to use LDAP for authentication, press Enter to select the default (NO). If you want to use LDAP for authentication, type 1 and press Enter. Respond to following prompts by entering the values. To enable SSL communications between the hub and the LDAP server, provide the appropriate values.
5. Accept the defaults for the Tivoli Event Integration Facility and the Workflow Policy/Tivoli Emitter Agent Forwarding.
6. At the following prompt, type 6 (Save/exit) and press Enter:

```
Hubs
##      CMS_Name
1      ip.pipe:elsrmt1[4441]
```

7. Restart the hub monitoring server:

```
./itmcmd server stop tems_name
./itmcmd server start tems_name
```

Configuring authentication using Manage Tivoli Monitoring Services: About this task

To configure authentication using Manage Tivoli Monitoring services, complete the following steps:

1. Change to the *install_dir/bin* directory and run the following command:
./itmcmd manage [-h *install_dir*]

where *install_dir* is the installation directory for IBM Tivoli Monitoring. The default installation directory on Linux or UNIX is /opt/IBM/ITM.

The Manage Tivoli Monitoring Services window is displayed.

2. Right-click the hub monitoring server and click **Configure**.
3. Click the **Advanced Setting** tab. Select **Security: Validate User**.
4. If you want to use LDAP to authenticate users instead of the system registry, select **LDAP user authentication**.
5. Click **OK**.

If you selected the LDAP option, the LDAP configuration panel is displayed.

6. Specify the values, then click **OK**.
7. Click **OK**.
8. Restart the hub monitoring server, using one of the following methods:
 - In the Manage Tivoli Monitoring Services window, right-click the hub monitoring server and select **Recycle**.
 - From the command line, enter:

```
./itmcmd server stop tems_name
./itmcmd server start tems_name
```

Running ldapsearch with LDAP configuration

ldapsearch is an LDAP command-line tool available from many LDAP server vendors. You can save a lot of time by running **ldapsearch** to verify the LDAP

information before configuring a hub monitoring server for LDAP authentication. You can also use it to troubleshoot problems you encounter with the configuration. Ideally, **ldapsearch** is run by the LDAP administrator.

Note: Use this tool only if you are configuring LDAP authentication through the hub monitoring server. If you are configuring LDAP authentication through the , use the TEPS/e (extension server) administration console to verify configuration parameters.

The **ldapsearch** command operates something like the ping command. If the values you use as input to the command are correct, the command returns a version of the values you use in the search. If the values are not correct, the command returns either nothing, or an error message that can help you determine which value is involved, such as an incorrect password or a bad host name.

IBM Tivoli Directory Server (ITDS) **ldapsearch** is the best suited for Tivoli Monitoring. The ITDS **ldapsearch** supports GSKit SSL operations used in Tivoli Monitoring and has additional command-line options to support LDAP SSL searches. Tivoli Monitoring does not include ldapsearch with production installation. For information on IBM Tivoli Directory Server ldapsearch see <http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/topic/com.ibm.IBMDS.doc/commandref05.htm#ldapsrch>.

ldp.exe is a Microsoft Windows LDAP search tool which has the same basic features as **ldapsearch**. It can be downloaded from Microsoft Website for your version of windows. ldp.exe is included in the Windows Server 2003 CD support tools. For information on using Microsoft Windows **ldp** command, see <http://support.microsoft.com/kb/224543>.

ldapsearch command-line options

Table 8. ldapsearch command line options and corresponding monitoring server configuration parameters

Option	Description	Corresponding Parameter in TEMS configuration file
-h <i>host</i>	The host name of LDAP server.	KGL_LDAP_HOST_NAME
-p <i>port</i>	The LDAP port number.	KGL_LDAP_PORT
-D <i>dn</i>	The LDAP bind ID Do not use this command-line option if LDAP bind ID is not required.	KGL_LDAP_BIND_ID
-w <i>password</i>	The LDAP bind password Use the unencrypted value for the ldapsearch command-line option. Do not use this command-line option if LDAP bind ID is not required.	KGL_LDAP_BIND_PASSWORD
-b <i>base_dn</i>	The LDAP base.	KGL_LDAP_BASE
-K <i>keyfile</i>	The LDAP key store file (used only with LDAP SSL).	KGL_KEYRING_FILE
-P <i>key_pw</i>	The LDAP key store password (used only with LDAP SSL). Use the unencrypted value for the ldapsearch command-line option.	KGL_KEYRING_PASSWORD decrypted value

*Table 8. **ldapsearch** command line options and corresponding monitoring server configuration parameters (continued)*

Option	Description	Corresponding Parameter in TEMS configuration file
-N key_name	The LDAP key store label (used only with LDAP SSL).	KGL_KEYRING_LABEL
Filter	LDAP user filter. Replace %v with Tivoli Enterprise Portal, SOAP, or tacmd user ID.	KGL_LDAP_USER_FILTER

Sample ldapsearch command (no SSL)

For a configuration with the following values for which SSL is not enabled and no bind ID and password are required:

LDAP host name ldap.itm62.com
 LDAP port name 389
 LDAP base ou=itm62users,o=itm62.com
 LDAP user filter "(mail=%v@us.ibm.com)"

you would use the follow command:

```
ldapsearch -h ldap.itm62.com -p 389 -b "ou=itm62users,o=itm62.com"
-s sub "(mail=sysadmin@itm62.com)"
```

If the input values were correct, you would receive the following as output:

```
uid=12345678,ou=itm62users,o=itm62.com
objectClass=person
objectClass=organizationalPerson
...
mail=sysadmin@itm62.com
...
```

Sample ldapsearch command (with SSL)

For a configuration with SSL enabled and bind ID and password required, with the following values:

LDAP host name ldap.itm62.com
 LDAP port name 636
 LDAP bind ID uid=1,ou=itm62users,o=itm62.com
 LDAP bind password itm62
 LDAP base ou=itm62users,o=itm62.com
 LDAP key store C:\IBM\ITM\itm62keyfiles\keyfile.kdb
 LDAP key stash C:\IBM\ITM\itm62keyfiles\keyfile.sth
 LDAP keystore label BM_Tivoli_Monitoring_Certificate
 LDAP keystore password itm62
 LDAP user filter "(mail=%v@us.ibm.com)"

you would use the following command:

```
ldapsearch -h ldap.itm62.com -p 636 -D uid=1,ou=itm62users,o=itm62.com
-w itm62 -b "ou=itm62users,o=itm62.com" -s sub
-K C:\IBM\ITM\itm62keyfiles\keyfile.kdb -P itm62
-N "IBM_Tivoli_Monitoring_Certificate" "(mail=sysadmin@itm62.com)"
```

If the input values were correct, you would receive the following as output:

```
uid=12345678,ou=itm62users,o=itm62.com
objectClass=person
objectClass=organizationalPerson
...
mail=sysadmin@itm62.com
...
```

Configuring user authentication through the

Here are the tasks required to enable user authentication through the Tivoli Enterprise Portal Server

Table 9. Tasks required for enabling user authentication through the

Task
If you intend to enable SSO, read and understand the requirements for using SSO.
Perform all prerequisite tasks and obtain the information required during configuration.
Enable authentication and configure SSO (if desired).
Map user IDs to LDAP distinguished names.
Export LTPA key used to encrypt tokens and import keys from other participating SSO applications (if not done as part of configuring SSO).

Prerequisites for configuring authentication on the

These tasks should be completed before user authentication is enabled on the hub monitoring server.

Table 10. Tasks to complete before configuring authentication

Task	Where to find information
Set up user accounts.	"Adding a user ID" on page 71
Set up user accounts in the authenticating registry.	See the documentation for setting up user accounts on the local operating system or LDAP directory server. For information on setting up users on z/OS, see <i>IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS</i> .

If you intend to authenticate through the , add or verify user IDs in the registry, but do *not* create an account for **sysadmin** until after you have enabled authentication and are already logged in to the .

Note: On Windows, the installer creates a **sysadmin** user account in the Windows registry and asks you to specify a password for that ID. The password is not required unless password authentication is enabled.

Tip: The Windows installer does not set the "Password never expires" option when it creates the **sysadmin** account. If you do not set this option, the password will expire according to the security policy on the hub computer, and you will not be able to log in to the portal server. Use the Windows Administrative Tools to ensure that the "Password never expired" option is selected for the **sysadmin** user account.

Before you enable authentication, obtain the following information

- If you are using an external LDAP server for authentication, obtain the information shown in this table from the LDAP administrator before configuring user authentication.

Table 11. LDAP configuration parameters

Parameter	Description
LDAP Type	<p>One of the following types of LDAP servers that can be defined to the portal server using the IBM Tivoli Monitoring installation and configuration utilities:</p> <ul style="list-style-type: none"> • Active Directory Server 2000 • Active Directory Server 2003 • Tivoli Directory Server 6.0 • Tivoli Directory Server 6.1 <p>Other servers must be configured using the extension services administration console. On the SSO Configuration window, select Other from the drop-down menu, then configure the directory server to the portal server.</p>
LDAP base	<p>The LDAP base node in the LDAP repository that which is used in searches for users. For example:</p> <p>IBM Tivoli Directory Server: dc=yourdomain,dc=yourco,dc=com Microsoft Windows Active Directory: dc=yourdomain,dc=yourco,dc=com Sun Java System Directory Server: dc=yourdomain,dc=yourco,dc=com</p>
LDAP bind ID	The LDAP user ID for bind authentication, in LDAP notation. This LDAP user ID must be authorized to search for LDAP users. This value can be omitted if an anonymous user can search for LDAP users.
LDAP bind password	The password for LDAP bind authentication. This value can be omitted if an anonymous user can bind to your LDAP server. This value is encrypted by the installer.
LDAP host name	The LDAP server host name. This value can be omitted if your LDAP server is on the same host as the Tivoli Enterprise Monitoring Server. (The default is localhost.)
LDAP port number	The LDAP server port number. This value can be omitted if your LDAP server is listening on port 389.

- If you intend to configure SSO, obtain the information described in this table from the LDAP administrator.

Table 12. SSO parameters

Parameter	Description
Domain name	<p>The Internet or Intranet domain which for SSO should be configured. Only applications available in this domain or its subdomains are enabled for the SSO. For example:</p> <p>ibm.com</p>
Realm name	<p>A parameter shared across applications that are using the SSO implementation and are configured for SSO within the domain defined with the Domain Name parameter. For example:</p> <p>ibm_tivoli_sso</p> <p>Applications configured for the same domain name, but for a different realm name will not work as a part of the same SSO infrastructure.</p>

- If you intend to export the keys used to encrypt LTPA tokens generated by TEPS/e as part of the configuration process, you will need to provide a name

for the key file and password to use to encrypt the key. If you intend to import the keys used by other participating applications, you will need the name of the key file and the password required to decrypt the file.

The must be running when you import or export keys.

Using single sign-on

The Tivoli Enterprise Portal

Using a browser client or Java Web Start client, you can launch from the into another participating Tivoli Web application by using Launch Application or by entering the URL of the application into a browser view. You can also use a browser view in the desktop client to launch into another application using SSO, but you can only launch *into* the from another application through the browser client.

Note: If you are using SSO and you want to use the browser client on the same computer as the Tivoli Enterprise Portal Server, you must reconfigure the client to use the fully qualified name of the host computer.

For SSO to be enabled, authentication must be configured through the Tivoli Enterprise Portal Server

Authenticated credentials are shared among participating applications using LTPA (Lightweight Third Party Authentication) tokens. An LTPA token is an encrypted datum containing previously authenticated user credentials. Participating SSO applications pass LTPA tokens using browser cookies.

LTPA tokens are secure because they are created using secure cryptography. The tokens are both encrypted and signed. The server creating a LTPA token uses a set of cryptographic keys. The cryptographic keys are used to encode the token, so that the encoded token traveling to the user's browser cannot be decoded by someone who does not have the cryptographic keys. The cryptographic keys also are used to validate the token such that the token integrity is verifiable and tampering can be readily detected. When an SSO server receives an HTTP request and sees that the LTPA token is included, the server verifies the token using its copy of the shared cryptographic keys, and the information in the valid token allows the server to recognize the logged-in user.

Accordingly, LTPA keys must be exchanged among participating SSO servers. The Tivoli Enterprise Portal

After the user IDs available for SSO have been created in the LDAP repository, enabling SSO involves these tasks.

Table 13. Tasks for enabling single sign-on

Tasks
Verify that all prerequisites for enabling authentication and single sign-on have been met.
Define user accounts.
Enable LDAP authentication on the and configure single sign-on.
Map user IDs to LDAP distinguished names.
Export LTPA keys and import keys from participating Tivoli Web applications.

Configuration procedures

Configure the Tivoli Enterprise Portal to authenticate users against a central LDAP repository,

About this task

If you intend to use SSO, you must configure user authentication through the . Note that you can configure the to authenticate users against an LDAP repository and not configure SSO, but you cannot configure SSO without configuring the for LDAP authentication.

Note: If you want to export and import LTPA keys during the configuration process, start the before you begin configuration.

Windows: Configuring the to authenticate to an external LDAP repository

About this task

Complete the following steps to configure a on Windows to authenticate users against an external LDAP repository:

1. Select **Start → Programs → IBM Tivoli Monitoring → Manage Tivoli Monitoring Services**.

The Manage Tivoli Monitoring Services window is displayed.

2. Right-click the and select **Reconfigure...**

The TEP Server Configuration window is displayed.

3. Click **OK** to accept the existing configuration.

A second TEP Server Configuration window is displayed.

4. In the LDAP Security area, check **Validate User with LDAP?**

The option **Enable Single Sign On** becomes available.

5. If you want to enable SSO, check the option, then click **OK** to open the LDAP window.

6. Specify the required LDAP values as appropriate for your site.

If you want to use a directory server other than those listed, select **Other** for LDAP type and use the instructions in “TEPS/e administration console” on page 60 to complete the LDAP server configuration.

7. Click **OK**.

If you selected **Enable Single Sign On** on the TEP Server Configuration window, the Single Sign On window is displayed. Proceed to step 9.

If you did not select **Enable Single Sign On** on the TEP Server Configuration window, you see a message asking if you want to reconfigure the warehouse connection information for the . Proceed to step 13.

8. Enter the Domain and Realm name.

9. If you want to export the key used to encrypt and decrypt the LTPA tokens generated by the to other applications participating in SSO:

- a. Click **Export Keys**.

A Save window opens, directed to the *ITM_installdir\InstallITM* directory. If necessary, navigate to the directory in which you want to create the file.

- b. Type a name for the file in which the LTPA keys should be placed, and click **Save**.

The Export keys windows is displayed.

- c. Type a password to use to encrypt the file, and click **OK**.
You see a console window while the file is created and encrypted, and then you are returned to the Single Sign On window.
10. If you want to import keys used by other applications to encrypt their LTPA tokens:
 - a. Click **Import Keys**.
An Open window is displayed, directed to the *ITM_install_dir\Install\ITM* directory. If necessary, navigate to the directory in which the key file is located.
 - b. Type the name of the file you want to import, and click **Open**.
The Export keys windows is displayed.
 - c. Type the password required to decrypt the file, and click **OK**.
You see a console window while the file is created and encrypted, and then you are returned to the Single Sign On window. Repeat the import process to import keys from additional participating servers.
11. Click **OK** to accept the current settings.
You see a prompt asking if you want to reconfigure the warehouse connection information.
12. Click **NO**.
After some processing of the configuration settings, the Common Event Console Configuration window is displayed. Sometimes this window does not get foregrounded. If processing seems to be taking longer than expected, minimize other windows and look for the configuration window.
13. Click **OK**.
The Manage Tivoli Monitoring Services window is displayed.
14. Restart the by right-click its name and selecting **Start**.

To complete the configuration, the administrator must log on to the portal using the **sysadmin** user ID and map the Tivoli Enterprise Portal

Linux and UNIX: Configuring the to authenticate to an external LDAP repository

You can configure user authentication using the Manage Tivoli Monitoring Services utility or from the command line.

Configuring portal server authentication using Manage Tivoli Monitoring Services:

About this task

To configure authentication using Manage Tivoli Monitoring services, complete the following steps:

1. Change to the *itminstall_dir/bin* directory and run the following command:
`./itmcmd manage [-h install_dir]`

where *install_dir* is the installation directory for IBM Tivoli Monitoring. The default installation directory on Linux and UNIX is */opt/IBM/ITM*.

The Manage Tivoli Monitoring Services window is displayed.

2. Right-click the and click **Configure**.
The Configure window is displayed.
3. In the LDAP Security area of the **TEMS Connection** tab, check **Validate User with LDAP**.

The **Enable Single Sign On** option becomes available.

4. If you want to use SSO, check **Enable Single Sign On**.
5. Click **Save**.

The LDAP Configuration window is displayed.

6. Specify the values, then click **OK**.

If you want to use a directory server other than those listed, select **Other** for **LDAP type** and complete the configuration.

If you elected to enable SSO, the SSO Configuration window is displayed. Proceed to step 7.

If you did not enable SSO, proceed to step 10.

7. Enter the Domain and Realm name.
8. If you want to export the key used to encrypt and decrypt the LTPA tokens generated by the to other applications participating in SSO:
 - a. Click **Export Keys**.

A Save window opens, directed to the Root directory. If necessary, navigate to the directory in which you want to create the file.

- b. Type a name for the file in which the LTPA keys should be placed and select the type of file it should be saved as, then click **Save**.

The Export keys windows is displayed.

- c. Type a password to use to encrypt the file, and click **OK**.

You see a console window while the file is created and encrypted, and then you are returned to the Single Sign On window.

9. If you want to import keys used by other applications to encrypt their LTPA tokens:

- a. Click **Import Keys**.

An Open window is displayed, directed to the Root directory. If necessary, navigate to the directory in which the key file is located.

- b. Type the name of the file you want to import, and click **Open**.

The Export keys windows is displayed.

- c. Type the password required to decrypt the file, and click **OK**.

You see a console window while the file is created and encrypted, and then you are returned to the Single Sign On window. Repeat the import process to import keys from additional participating servers.

10. Click **OK**.

11. If necessary, recycle the portal server, using one of the following methods:

- In the Manage Tivoli Monitoring Services window, right-click the portal server and select **Recycle**.

- From the command line, enter:

```
./itmcmd agent stop cq  
./itmcmd agent start cq
```

Configuring portal server authentication from the command line: About this task

To configure the portal server from the command line:

1. Log on to the computer where the Tivoli Enterprise Portal Server is installed.
2. At the command line, change to the *ITMinstall_dir/bin* directory, where *ITMinstall_dir* is the directory where you installed the product.

3. Run the following command to start configuring the Tivoli Enterprise Portal Server:
`./itmcmd config -A cq`
 You see the following message:
 Agent configuration started...

 followed by the prompt:
 Edit "Common event console for IBM Tivoli Monitoring" settings?
 [1=Yes, 2=No] (default is: 1):
4. Enter 2.
 The following prompt is displayed:
 Edit 'ITM Connector' settings? [1=Yes, 2=No] (default is: 1):
5. Enter 2.
 The following prompt is displayed:
 Will this agent connect to a TEMS? [1=YES, 2=NO] (Default is: 1):
6. Accept the defaults for this prompt and the prompts that follow it until you see the following prompt. .
 LDAP Security: Validate User with LDAP ? (1=Yes, 2=No)(Default is: 2):

 The defaults should reflect the selections made during the original configuration
7. Enter 1 to begin configuration of LDAP authentication and provide the values for the LDAP parameters.
8. If you want to enable single sign-on as well as LDAP authentication, enter 1 at the following prompt, then provide the Realm name and Domain name.
 Enable Single Sign On ? (1=Yes, 2=No)(Default is: 2):

 After the installer has completed the configuration, the following message is displayed:
 Agent configuration completed...
9. Recycle the :
`./itmcmd agent stop cq`
`./itmcmd agent start cq`

 If you enable single sign-on, ensure that the Tivoli Enterprise Portal administrator exports the LTPA keys for exchange with other participating applications and import the keys from those applications.

TEPS/e administration console

The Tivoli Enterprise Portal Server extended services (TEPS/e) has an administration console. Use the TEPS/e console for configuring an LDAP server that is not supported by the Tivoli Management Services installation and configuration utilities. Even if you are configuring LDAP authentication through the portal server, you can use the administration console to verify the configuration parameters.

Mapping user IDs to LDAP distinguished names



About this task

One of the items of information passed in an LTPA token is a valid user name. Because the user name is being authenticated by a central LDAP repository that is shared by participating Tivoli applications, this name is the user's unique identifier

(UID) as known by the LDAP registry. This name is not necessarily the same as the user ID known to the . For this reason, Tivoli Enterprise Portal user IDs must be mapped to LDAP UIDs. This is done in the portal Administer Users window by a user with administrator authority.

Note: If authentication is being configured through the , user IDs are mapped using the Filter parameter.

Map user IDs to LDAP user IDs :

1. Log on to the portal using **sysadmin** or another user account with full administrative authority.
2. Click  **Administer Users**.
3. In the Administer Users window, right-click the row of the user ID you want to map and select  **Modify User**.
The Modify User window is displayed.
4. In the Distinguished Name field, type the LDAP distinguished name to be associated with the user account or use the **Find** button to locate it in the Distinguished Name List. For example:
UID=TEPUSER,0=SS
5. Click **OK** to save the mapping and return to the Administer Users window.
6. Repeat steps 3 through 5 until you have mapped all the users you want to authenticate against the configured LDAP registry.
7. Click **OK** to exit the Administer Users window.

Importing and exporting LTPA keys

About this task

The Manage Tivoli Monitoring Services window enables you to export the keys used to encrypt the LTPA tokens that are generated by the TEPS/e to other applications participating in SSO, and to import keys used by other applications for encryption. On Windows, AIX, and Linux, keys can also be imported and exported from the SSO Configuration window. On AIX and Linux, import and export scripts are also available.

When you request an export or import operation, you must provide the name of the key file to export or import and the password to use to encrypt or decrypt the file.

The must be running for import and export operations to be performed.

To import and export LTPA keys using Manage Tivoli Monitoring Services, right-click the Tivoli Enterprise Portal Server

To import and export LTPA keys during SSO configuration.

On AIX and Linux you can also use the `exportKeys.sh` and `importKeys.sh` scripts. The scripts are installed to `/opt/IBM/ITM/aix533/iw/scripts` on AIX, `/opt/IBM/ITM/li6263/iw/scripts` on Linux, and `/opt/IBM/ITM/1s3263/iw/scripts` on zLinux. The commands use the following syntax:

```
./exportKeys.sh <filename> <password>
./importKeys.sh <filename> <password>
```

Tivoli Enterprise Portal distinguished names

Starting with ITM V6.2 Fix Pack 1, the Tivoli Enterprise Portal Server requires distinguished names in addition to user IDs for each user account. The default distinguished name for a new user you create for the Tivoli Enterprise Portal will have the following structure:

```
UID=tep_userid,0=DEFAULTWIMITMBASEDREALM
```

Note: If you are upgrading from ITM V6.2, distinguished names are automatically created for existing users.

If you are creating new users, and you have already configured the portal server with LDAP and all of your users have entries in the LDAP registry, then you will have to map new users to their corresponding LDAP distinguished names in the Tivoli Enterprise Portal Administer Users window.

About distinguished names

A distinguished name (DN) is a unique name that unambiguously identifies a single entry in a tree-like structure called the Directory Information Tree (DIT). Each DN is constructed of a relative distinguished name (RDN), constructed from some attribute or attributes in the entry, following by the DN of the parent entry (for example, DN=UID=tep_user,0=DEFAULTWIMITMBASEDREALM).

Reconfiguring the browser client for SSO

About this task

By default, the Launch URL associated with the browser client running on the same computer as the Tivoli Enterprise Portal Server is localhost. If you want to use a browser client on the same computer as the Tivoli Enterprise Portal Server, this value must be the fully-qualified name of the computer, for example dev1.myco.com. The suffix myco.com is the domain value you enter in the SSO configuration panel. Using the suffix ensures that SSO tokens are visible only to the servers that are under the same domain suffix.

To reconfigure the browser:

1. Launch the Manage Tivoli Monitoring Services utility.
2. Right-click the Tivoli Enterprise Portal Browser entry and select **Reconfigure...** from the pop-up menu.
The Configure Enterprise Portal Browser window is displayed.
3. In the **Host** field beneath TEP Server, type the fully-qualified name of the computer. For example:
myhost.mycompany.com
4. Click **OK** to close the window and save the setting.

Migrating authentication from the monitoring server to the portal server

About this task

If your environment has already been configured for LDAP configuration using the hub monitoring server and you now want to use authentication through the , take

the following steps. Make sure that all users log off the before you begin the procedure and do not log on again until the procedure is completed.

1. Disable Tivoli Enterprise Monitoring Server

Use the Manage Tivoli Monitoring Services utility to reconfigure the monitoring server:

- a. Right-click the hub monitoring server, and select **Reconfigure...** (Windows) or **Configure...** (Linux or UNIX) from the popup menu.
- b. On the Tivoli Enterprise Monitoring Server
- c. Click **OK**.
- d. Click **OK** to accept the existing settings on the next window.
- e. Restart the monitoring server.

From Linux or UNIX command line:

- a. At the command line, change to the `/opt/IBM/ITM/bin` directory (or the directory where you installed IBM Tivoli Monitoring).
- b. Run the following command:

```
./itmcmd config -S -t tems_name
```



where *tems_name* is the name of your monitoring server (for example, HUB_itmdev17).


- c. Press Enter to accept the existing values until you see the prompt for **Security: Validate User**.
 - d. Enter `N0` to disable security.
 - e. Continue to press Enter until the configuration is complete.
 - f. Restart the monitoring server.
2. Rename the **sysadmin** UID in the LDAP registry (for example, **sysadmin_tems**).
3. Configure LDAP authentication through the :
- Use the Manage Tivoli Monitoring Services utility or the command line to reconfigure the .
4. Start the and log on as **sysadmin**.
 5. Adjust all user mapping to LDAP user IDs.
 6. Before logging off the , have the LDAP administrator rename the LDAP **sysadmin** account back to **sysadmin**, then map the Tivoli Enterprise Portal
 7. Save the changes and log off the .

At this point, the migration is complete. If you want to re-instate authentication through the for purposes of authenticating SOAP users, follow the steps in “Configuring user authentication through the hub monitoring server” on page 47.


Chapter 6. User administration



Every portal work session begins with a successful logon and connection to the Tivoli Enterprise Portal. The logon user IDs and user groups are created and profiled through the Administer Users window.

Administer Users is a multi-tabbed two-paned window. The top frame has two tabs:  **Users** and  **User Groups**, that list the user IDs, distinguished names if the portal server is configured for authentication to an LDAP repository, and the user groups that are stored on the portal server. The profile of the selected user or user group is reflected in the bottom frame:

 **Permissions** has a list of the portal features in the Authorities box. On the right are the possible operations for the selected feature. A selected check box means the selected user or user group has permission to perform that operation; a ☐ indicator next to the check box means the permission was added to a user group the user belongs to.

☐ **Applications** shows all the applications being monitored and that are available for assigning to the user or user group. One user or user group, for example, can be profiled to see only the OMEGAMON[®] applications, another to see only Linux and Oracle, middleware, and another to see all applications.

 **Navigator Views** shows all the Navigator views that are on the portal server and that are available for assigning to the user or user group. The user or user group can be restricted to seeing only a certain branch of a Navigator view rather than the entire hierarchy.


 **Member of**, when the Users tab is selected, or  **Members**, when the User Groups tab is selected, is a list of the groups the user belongs to or the user names in the group.


The User Administration function enables you to maintain user IDs and user groups on the portal server, and provides varying degrees of access to the features and views of your monitored environment to accommodate any combination of job roles, such as *operators* who respond to alerts and direct them to the appropriate person for handling and *administrators* who plan, design, customize, and manage the monitoring environment.

In some managed enterprises one person might assume all of these roles. In larger enterprises, the roles are often divided. You can choose to assign roles by individual user or by user type or both.

Administer Users

Your user ID and the user groups you are a member of are profiled with a set of permissions that determines which features you are authorized to see and use, a list of monitored applications you are authorized to see, and a list of Navigator views (and the highest level within a view) you can access.

Note: The Tivoli Enterprise Portal online help alerts you to limitations due to restricted permissions wherever you see the  User ID icon.

Clicking  **Administer Users** opens the Administer Users window. This is a two-paned window with Users and User Groups tabs in the top frame, and several tabs in the bottom frame. This arrangement enables the administrator to manage

user profiles by individual user, by user groups, or a combination of the two. You might create a user profile, then copy the profile for each additional user and change settings as needed (such as, for the Action feature, granting View permission to one user and granting Modify permission to a different user). Or you might create a user group with a particular profile and add users to the group. Then you can modify the permissions once for a group and apply to all members automatically.

Users and User Groups

The  **Users** and  **User Groups** tabs list the user IDs and the user groups that are stored on the .

After you select a user or user group from one of the lists, you can click any of the tabs in the lower half of the window to see the what permissions have been granted and what has been assigned. User groups enable the administrator to authorize the same set of functional permissions, applications, and Navigator views to multiple users at one time. Management of user authorization can be done by groups as well as individually. A user can be associated with one or more user groups; authorization by group will be by inclusion and not exclusion (nested groups are supported). Authorization will also be by global authority and by association with managed system and managed system lists. This security is not dependent on external authorization.


Permissions

You can authorize the same set of functional permissions multiple users, user group or each user ID at one time.

The following features are enabled or disabled individually for each user ID or user group.

Action

☒ **View** allows the user to see and run a take action command from the available list of commands in the Take Action view and in the pop-up menu for the Navigator item.

☒ **Modify** allows the user to create and save Take Action commands. When enabled,  **Edit Action** appears in the Navigator pop-up menu.


When issuing a take action command, you must be authorized on the relevant system for the requested command. For example, to issue a TSO command, your user ID must be both a valid TSO ID and a valid user ID on the portal server. The user ID must be typed with the same letter casing exactly as typed when logging on to the portal server (with the same letter casing).

Agent Management

☒ **Manage** allows the user to perform agent deployment throughout the managed network. This includes installing a monitored product, keeping the software revisions up-to-date, and removing an agent from the managed network. This permission also requires Action - Modify to be enabled.

☒ **Start/Stop** allows the user to start a monitoring agent or to stop it running.

Custom Navigator Views

☒ **Modify** allows the user to create new Navigator views, edit and delete them. With Modify cleared, the user will not see  **Edit Navigator View** in the Navigator toolbar.

Event ☒ **Attach** allows the user to attach a file (such as detailed notes) to the situation event. This permission requires that the user also have the Acknowledge and View permissions.

☒ **Close** lets you close a pure event or an event that was open before a situation was stopped manually. When it is enabled, ☒ **Close Situation Event** appears in the pop-up menu of the situation event flyover list, event Navigator item, and situation event console view when the selected event is a pure event or the situation has been stopped.


☒ **View** enables you to see situation event indicators in the Navigator when situations become true.

☒ **Acknowledge** allows you to acknowledge a situation event. When this permission is enabled, **Acknowledge Event** appears in the pop-up menu of the situation event flyover list, event Navigator item, and situation event console view.


Feature

☒ **Enable** is dimmed because you cannot change it. The access to this feature is determined by your organization's IBM Tivoli Monitoring license.

History

☒ **Configure** allows the user to open the History Collection Configuration window, configure history files and data rolloff, and start and stop data collection for the different attribute groups. When this permission is enabled,  **History Configuration** appears in the main toolbar.

Launch Application

☒ **Launch** allows the user to invoke any of the launch definitions available for the Navigator item, table view, chart view, or situation event console view. When this permission is enabled,  **History Configuration** appears in the main toolbar.




☒ **View** allows the user to see the composition of the selected launch definition.



☒ **Modify** allows the user to create, edit and delete launch definitions.





Managed system group

☒ **View** allows the user to access the Object group editor for viewing managed system groups. The user also needs Modify permission for the Object group editor tools to be available.


☒ **Modify** allows the user to open the Object group editor to create, edit and delete object groups and managed system lists.

Policy ☒ **View** allows the user to open the Workflows window to see the policies and their definitions. With View permission, the  **Workflow Editor** is available in the main toolbar and  **Manage Policies** is available in the Navigator pop-up menu at the  agent level.


☒ **Start/Stop** lets you start and stop policies. With this permission enabled,  **Start Policy** and  **Stop Policy** are available when you select a policy.

☒ **Modify** allows the user to open the Workflow editor to create and edit policies. With the Modify permission enabled,  **New Policy** is available after the user selects a policy, as are the other editing tools:  **Edit Workflow**,  **Copy Policy**, and  **Delete Policy**.





Query ☒ **View** allows the user to access the Query editor through the Properties editor and select a query for the selected table or chart. With the View permission enabled, the user can assign a query through the Query tab of the Properties editor.

☒ **Modify** allows the user to create, edit and delete queries in the Query editor. With the Modify permission enabled,  **Query Editor** is available from the main toolbar, as are the query editing tools.

Situation

☒ **View** allows the user to see situations in the Situation editor, including any expression overrides, and in the Manage Situations at Managed System window. With the View permission enabled,  **Situation Editor** is available in the main toolbar and in the Navigator item (except at the platform level) pop-up menu.

☒ **Modify** lets you create new situations and manage them. When the Modify permission has been granted, the situation editing tools and pop-up menu options are available in the Situation editor, as well as the **Override Formula** button in the Distribution tab for qualifying situations.

☒ **Start/Stop** lets you start or stop a situation and enable or disable a situation override. When this permission is enabled,  **Start Situation** and  **Stop Situation** are available in the situation event flyover list, situation event console view, Situation editor pop-up menu, and the Manage Situations at Managed System window; and  **Enable Situation Overrides** and  **Disable Situation Overrides** are available in the Situation editor pop-up menu.

Terminal Script

☒ **View** allows the user to run or stop running a terminal emulator script and to see them, but not to edit them. If View is disabled the user will be able only to run or stop a script.

☒ **Modify** allows the user to create or record new terminal emulator scripts, edit, and delete them.

User Administration

If you are viewing your own user ID, you will see that View and Modify are disabled; you cannot change your User Administration permissions.

☒ **Logon Permitted** enables log on to the portal server with this user ID. The administrator can clear the check box to deny a user access to the portal. This option works in conjunction with the KFW_AUTHORIZATION_MAX_INVALID_LOGIN (the default is 0, unlimited attempts are allowed) parameter in the Tivoli Enterprise Portal Server Environment Configuration file, *kfwenv*. When the value has been set and the invalid attempts have been exceeded, the check box is cleared automatically and the administrator must select the check box to reset the logon attempt count. See the *IBM Tivoli Monitoring Administrator's Guide* for details. ☐ **Modify** allows the editing of user IDs and removing them.

When this permission is enabled,  **Administer Users** is available in the main toolbar and the tools are available in the Administer Users window.

- ☒ **Author Mode Eligible** allows the user to enable or disable their Author Mode permission under **Workspace Administration** (see next authority), but not for any other user IDs.
- ☐ **View** allows the user to open the Administer Users window and see their user profile.
- ☒ **Administration Mode Eligible** allows the user to enable or disable their Administration Mode permission under **Workspace Administration** (see next authority), but not for any other user IDs.

Workspace Administration

- ☒ **Workspace Author Mode** allows the user to create and edit workspaces, links, and terminal emulator scripts. If Workspace Author Mode is disabled, the user cannot make any of these changes but can continue monitoring and responding to alerts; the tools can still be seen, but they are disabled.
- ☐ **Workspace Administration Mode** is available only for the SYSADMIN user ID and new IDs made from it in the Create Another User window. When administration mode is enabled, changes you make to workspaces affect all users who log on to the same portal server

WebSphere MQ Configuration Authorities

IBM Tivoli OMEGAMON XE for Messaging: WebSphere MQ Configuration installations will see this folder.

- ☒ **View** allows the user to see, but not change, your organization's WebSphere MQ configuration in the Navigator Configuration view.
- ☒ **Modify** allows the user to change your organization's WebSphere MQ configuration or to schedule updates in the Configuration view.

Applications


Your user ID is set so you can see some or all the application types being monitored. For example, one user might be able to see only mainframe applications, while another can see only middleware, and another sees all applications.

Allowed Applications

Shows the applications that you can access from Tivoli Enterprise Portal.

Available Applications




Shows the applications available for assignment to the selected user. If **<All Applications>** is on the **Allowed Applications** list, then no other entries can be added. You need to move it back to **Available Applications** before you can select a subset of applications to assign.

Select the applications you want to add, or select **<All Applications>**, and  move them to the **Allowed Applications** list. After selecting the first application, you can use Ctrl+click to select other applications or Shift+click to select all applications between the first selection and this one.


Navigator views

When a Navigator view is created, only the author is able to see the view, but it is available for the administrator to assign to users. An assigned Navigator view means the user can open it. For each assigned view, the user can be restricted to see only a certain branch rather than the entire hierarchy.

Assigned Views

Shows the Navigator views the user is able to see and access. The first Navigator view in this list is the default for the user; it displays automatically whenever the user logs on. You can select any views to which you do not want the user to have access, and click  right arrow to move them to the **Available Views** list. Select the appropriate entries and click  left arrow to move them to the **Assigned Views**. You can move a Navigator view to the top of the list to make it the default by clicking the  up arrow.

Available Views

Shows the Navigator views not assigned to the user and available for assignment. Select the Navigator views you want to add and move them to the **Assigned Views** list by using the  left arrow. After selecting the first view, you can use Ctrl+click to select other views or Shift+click to select all views between the first selection and this one.

Assigned Root

Shows the Navigator view chosen in Assigned Views, with the user's assigned Navigator root highlighted. The root is the top-most level of this Navigator view that the user can access. The user can access this item and all items below it, but no items parallel to or above it in the Navigator.

For example, you can assign UNIX Systems as the assigned root. The user sees the UNIX Systems workspaces and those below, but is not able to see the Enterprise workspaces or anything under Windows Systems.

Member Of and Members

When you select a user or user group from the list, the last tab on the bottom set of tabs reads either **Member Of** or **Members** (reflecting the selection of a User or User Group). Assignment of users to groups can be done in either tab.

Managing user IDs


Managing user IDs begins with planning the authorities to grant to users and whether they will belong to user groups.

The Administer Users window provides the tools for creating and maintaining user IDs, and adjusting permissions. This is also where user IDs are mapped to their unique identifier in the LDAP repository if user authentication through the portal server has been configured.


Viewing and editing a user ID






After a user has been added to the **Users** list in the Administer Users window, you can check and edit the profile settings at any time.



About this task

 To use this function, your user ID must have Modify permission for User Administration.

Use the following steps to edit a user ID:

1. Click  **Administer Users**.
2. Do one of the following in the **Users** list:
 - Click inside the **Name** or **Description** field to edit either of them.

- Double-click anywhere in a row to open the Modify User window for editing any of the fields.
 - Right-click the user profile you want to edit and click  **Modify User**.
3. Edit the **User Name**, **Distinguished Name** or **User Description**, then click **OK**. Distinguished Name is necessary if user authentication is through the portal server to an LDAP repository. If you have not yet added the DN, click **Find** to locate the name that matched the user ID. You cannot change the one-word User ID other than to change the letter casing. You must instead delete the user profile and create a new one.
 4. To change the  **Permissions**, select a function from the **Authorities** tree and select or clear each option as appropriate for all functions with permissions that you want to change.
 5. To assign access privileges to applications (managed system types), click the  **Applications** tab, select any applications you want to remove from the **Allowed Applications** list and click ; select the applications you want to add from the **Available Applications** list (or select **<All Applications>**), and click .

After selecting the first application, you can use Ctrl+click to select other applications or Shift+click to select all applications between the first selection and this one.
 6. To change any Navigator view assignments, click the  **Navigator Views** tab, then add or remove Navigator views from the **Assigned Views** list, and use  to place the one you want to be the default at the top of the list. For each Navigator view, change the **Assigned Root** as needed.
 7. When you are finished editing the user profile, save your changes with **Apply** if you want to keep the Administer Users window open, or **OK** if you want to close it. The next time the user logs on, the permission changes will be in effect.

Note: You can change your own user permissions except Create and Modify for User Administration.

Adding a user ID




Create a user ID for all users that should be able to log onto the Tivoli Enterprise Portal Server. You can use the default user profile or copy the profile of an existing user.

Before you begin









To use this function, your user ID must have Modify permission for User Administration.

About this task

Take these steps to add a new user:

1. Click  **Administer Users**.
2. Create a new user ID or create one from another:
 - To create a new user ID with the default user profile, click  **Create New User**.
 - To create a new user ID from an existing one, select the profile that you want to use from the **Users** list and click  **Create Another User**.
3. In the Create New User window, enter the user information:
 - **User ID:** The logon name. This name can be up to 10 characters and can contain no spaces. The name is limited to eight characters if user

authentication is at the hub monitoring server and uses RACF® (resource access control facility) security for z/OS.

- **User Name:** The name of the user or job classification or both. This name can include spaces and be up to 32 characters. The user name is displayed in **Users** list.
 - **Distinguished Name:** The unique identifier in the Lightweight Directory Access Protocol (LDAP) repository for the name given in the **User ID** field. Click **Find** to locate and insert the distinguished name, such as `UID=FRIDA,O=DEFAULTWIMITMBASEDREALM`
 - **User Description:** Optional description for the user. The text can include spaces and punctuation.
4. Click **OK** to close the window and see the new user ID arranged alphabetically in the **Users** list.
 5. To change the  **Permissions**, select a function from the **Authorities** tree and select or clear each option as appropriate for all functions with permissions that you want to change.
 6. To assign access privileges to applications (managed system types), click the ☐ **Applications** tab, then select **<All Applications>** or the individual applications the user should see, and click  to move them to the **Allowed Applications** list. After selecting the first application, you can use Ctrl+click to select other applications or Shift+click to select all applications between the first selection and this one.
 7. To assign Navigator views, click the  **Navigator Views** tab:
 - a. Select a Navigator view (or more with Ctrl + click and Shift + click) from the **Available Views** and click  to move it to the **Assigned Views**.
 - b. Use  to place the view that you want to be the default at the top of the list; use  and  to arrange the other Navigator views in the order that they should appear in the Navigator toolbar View  list.
 - c. For the selected Navigator view, change the **Assigned Root** as needed.
 8. When you are finished creating the user profile, save your changes with **Apply** if you want to keep the Administer Users window open, or **OK** if you want to close it.


What to do next

The Logon window has a field for entering a password. If you want the user ID to include a password, you must define the same user ID, including a password, to your network domain user accounts or to the operating system where the hub monitoring server is installed. Also, the monitoring server must be configured to validate users, which is the default on the Windows-based hub monitoring server. (In **Manage Tivoli Monitoring Services**, right-click **Tivoli Enterprise Monitoring Server** and click ☒ **Reconfigure**.)



Removing a user ID

You can remove a user ID as needed.

About this task

 To use this function, your user ID must have Modify permission for User Administration.


Use the following steps to remove a user ID:


1. Click  **Administer Users**.
2. Select the user ID that you want to delete. You can select additional user IDs with Ctrl+click, or with Shift+click to select all user IDs between the first selection and this one.
3. Click  **Remove Users** to delete the selected user ID and profile from the list.
4. When a message asks you to confirm the user ID removal, click **Yes**. The user is permanently removed from the user ID list. If the user is currently signed on, this does not affect their work session, but they will not be able to log on again.

Note: You cannot remove your user ID (the one you logged on with) or the <Default User> ID.

Default user

The first user ID in the **Users** list is <Default User>.

 To use this function, your user ID must have Modify permission for User Administration.





The Default User ID is used as the template ID for users created with  **Create New User**. Edit this user ID if you want to change any of the default settings. The initial defaults enable all the functions listed under Tivoli Enterprise Portal Authorities except User Administration Create and Modify. Any changes you make to the <Default User> ID apply to users created from this point on; they do not affect any existing user ID settings.


Managing user groups

User groups enable the administrator to authorize the same set of functional permissions, applications, and Navigator views to multiple users at one time. Management of user authorization can be done by groups as well as individually.

A user can be associated with one or more user groups. If a permission is granted to a user directly through their user ID, they maintain that permission even if a user group they belong to does not grant that permission. The reverse is also true, so that if an individual user ID is not granted a permission but the group ID is, the user will have the permission through their membership in the user group. Thus, the user's permission set is collected from what is given to the individual user ID and to any and all user groups that they belong to.

Authorization will also be by global authority and by association with managed system and managed system groups. This security is not dependent on external authorization.


When the active top tab is  **Users**, the last tab on the bottom set of tabs reads  **Member Of**. When the active top tab is  **User Groups**, you will also have a  **Members** tab. Assignment of users to groups can be done in either of these lower tabs.






Click the group in the details view at the top, then go to the  **Members** tab to see the list of users that belong to this group. likewise, to see the groups a user belongs to.

Viewing user group memberships

You can view both the groups a user ID belongs to, and the list of user IDs belonging to a user group.

About this task


 To use this function, your user ID must have Modify permission for User Administration.

1. Click  **Administer Users**. The Administer Users window is divided into two, with **Users** and **User Groups** tabs at the top, and **Permissions**, **Applications**, **Navigator Views**, and **Member Of** below.
2. To see the groups a user belongs to, select a name from the  **Users** list, then click the  **Member Of** tab. The groups the user belongs to are listed in the **Assigned Members Of** list.
3. To see the user IDs assigned to a group, select a name from the  **User Groups** list, then click the  **Members** tab. The users belonging to the group are in the **Assigned Members** list.








Adding a user group





You can create a new user group from the beginning or you can copy a group with similar permissions and user assignments to what you want, then modify the copy.

About this task

 To use this function, your user ID must have Modify permission for User Administration.

Complete these steps to add a user group:


1. Click  **Administer Users** to open the Administer Users window.
2. Click the  **User Groups** tab.
3. Do one of the following:
 - To create a new user group, click  **Create New Group**.
 - To copy an existing user group, select the group name from the list and click  **Create Another Group**.
4. In the Create New Group or Create Another Group window, enter the following user information:
 - a. **Group ID:** The group identifier. This name can be up to 10 characters and can contain no spaces. The name is limited to eight characters if the hub monitoring server uses RACF (resource access control facility) security for z/OS.
 - b. **Group Name:** The name or job classification for the user group. This name can include spaces..
 - c. **Group Description:** The text to describe the user group, such as their responsibilities. The description can include spaces and punctuation.
5. Click **OK** to close the window and see the new user group arranged alphabetically in the User Group list.
6. Add members to the group in the  **Members** tab by selecting one or more user IDs in the **Available Members** list and clicking  to move to the **Assigned Members** list.
7. To change the  **Permissions** for the group, select a function from the **Authorities** tree and select or clear each option check box for all functions.

8. To assign access privileges to applications (managed system types) for the group, click the  Applications tab, then select <All Applications> or the individual applications the user should see, and click  to move them to the **Allowed Applications** list. After selecting the first application, you can use Ctrl+click to select other applications or Shift+click to select all applications between the first selection and this one.
9. To assign Navigator views to the group, click the  Navigator Views tab, then add or remove Navigator views from the **Assigned Views** list, and use  to place the default view at the top of the list. For each Navigator view, change the **Assigned Root** as needed.
10. When you are finished creating the user group, save your changes with **Apply** to keep the Administer Users window open, or **OK** to close it.










Reviewing and editing a user group

After a user group has been added to the **User Groups** list in the Administer Users window, you can check and edit the profile settings at any time.

About this task

 To use this function, your user ID must have Modify permission for User Administration.

Use the following steps to edit a user ID:


1. Click  **Administer Users** to open the Administer Users window.
2. Click the  **User Groups** tab.
3. Right-click the user group to edit and click .
4. Edit the **Group Name** and **Group Description**, then click **OK**. You cannot change the one-word group ID. You must, instead, create another user group from this one and give it a new name, then delete this one.
5. To change the  Permissions, select a function from the **Authorities** tree and select or clear each option as appropriate for all functions with permissions that should change.
6. To change the group access privileges to applications (managed system types), click the  Applications tab, select any applications you want to remove from the **Allowed Applications** list and click ; select the applications you want to add from the **Available Applications** list (or select <All Applications>), and click . After selecting the first application, you can use Ctrl+click to select other applications or Shift+click to select all applications between the first selection and this one.
7. To change any Navigator view assignments for the group, click the  Navigator Views tab, then add or remove Navigator views from the **Assigned Views** list, and use  to place the one you want to be the default at the top of the list. For each Navigator view, change the **Assigned Root** as needed.
8. When you are finished editing the user group, save your changes with **Apply** to keep the Administer Users window open, or **OK** to close it. The user group changes are effective the next time each group member logs on.

Note: You can change the permissions, except Create and Modify for User Administration, of any groups you are a member of.




Removing a user group

You can remove a user group.

About this task

 To use this function, your user ID must have Modify permission for User Administration.

Use the following steps to remove a user ID:

1. Click  **Administer Users** to open the Administer Users window.
2. Click the  **User Groups** tab.
3. Select the user group to delete from the list and click  **Remove Selected Group**. You can select additional user IDs with Ctrl+click, or with Shift+click to select all user groups between the first selection and this one.
4. When a message asks you to confirm the user group removal, click **Yes**. The group is permanently removed from the user group list. Any members of this user group who receive permissions from the group will not be affected until they next log on to the portal server.

Notes on user administration

Read these notes to understand the user ID contribution to Tivoli Enterprise Portal functions and modes.

Workspace administration mode

Any changes you make to workspaces, links, and terminal host session scripts in the portal are available only to your user ID. The exception is while Workspace Administration Mode is enabled.

Workspace administration mode enables you to customize and add workspaces, links, and terminal emulator scripts that are shared with all users connected to the same .

SYSADMIN logon ID

Tivoli Enterprise Portal requires your logon ID whenever you start a work session. Every ID must first have been registered on the Tivoli Enterprise Portal Server. You can log onto Tivoli Enterprise Portal with **SYSADMIN** and register other user IDs through the Administer Users window. The initial user ID, **SYSADMIN**, has full access and complete administrator authority. The system administrator registers additional users and sets their access privileges and authority.

User ID and groups

Each user ID is stored at the Tivoli Enterprise Portal Server and contains:

- The user name
- Job description
- Permissions for viewing or modifying Tivoli Enterprise Portal functions
- Assigned Navigator views and which Navigator item in each view appears as the root (default is the first item)
- Access to specific monitoring applications
- The user groups the user belongs to and indicators to signify when a permission has been granted to the user by a user group

Each user group is also stored at the and has the same contents as for individual user IDs. But, instead of a list of user groups, it has a list of the user IDs assigned to the group.

Default user

The first user ID in the list is **<Default User>** and is used as the template ID for users created with Create New User. Edit this user ID if you want to change any of the default settings. The initial defaults enable all the functions listed under Tivoli Enterprise Portal Authorities, except the Modify permission for **User Administration**. Any changes you make to **<Default User>** ID apply to users created from this point on; they will not affect any existing user ID settings.

Granting access to a user

You set the authority privileges for each user when you create their user IDs. Giving users access to operational areas and customization options takes planning. Consider the job responsibilities of each user and the company security requirements when specifying authority privileges.

Important: Anyone with permission to create custom queries obtains access to all of the ODBC data source names (DSNs) created at the Tivoli Enterprise Portal Server. Add database user IDs, to be used in the DSN, to your database software, making sure to restrict user access to only those tables, columns, and so on, allowed by your organization's security policies.

Validating user access

The Tivoli Enterprise Portal Server verifies user IDs whenever users log on to Tivoli Enterprise Portal. If a user's job description changes and the user requires different access to the , you need to review and perhaps change the user's permissions.

The user ID for logging on to the might include a password. You do not establish passwords in the portal. Instead, you need to define a matching user ID with password to the network domain user accounts or to the operating system where the hub Tivoli Enterprise Monitoring Server resides:

- User Accounts on the Windows system
- Password file on the UNIX system
- RACF or ACF/2 host security system on the z/OS system

As well, the monitoring server must be configured to Validate User. When users log on to Tivoli Enterprise Portal, the monitoring server makes a request to the domain or the operating system to validate the user ID and password.

If the monitoring server has been installed on a distributed system, you can check if it has been configured to Validate User:

1. Start the Manage Tivoli Monitoring Services program:

Windows Start → Programs → IBM Tivoli Monitoring → Manage Tivoli Monitoring Services.

UNIX Change to the install_dir/bin directory and run the following command: ./itmcmd manage [-h install_dir] where install_dir is the installation directory (default is opt/IBM/ITM).

2. Right-click the row for TEMS1 (hub) and select **Reconfigure**.

3. In the Tivoli Enterprise Monitoring Server Configuration window, observe the setting of the ☒ **Security: Validate User** check box.

When this option is selected, the password is required whenever a user logs on to the ; when it is cleared, the user name is required to log on but no password is required.

Note: Be aware that passwords must follow the security requirements for your organization. If this includes periodic password changes, you might get a **Logon password has expired** message while attempting to log on to the . Should this happen, you need to change your system password before you can log on. For example, on Windows this means changing the password through the Administrative Tools User Accounts.

Launching into the portal from other applications

In addition to any security requirements for launching into the Tivoli Enterprise Portal (such as single sign-on requirements), the Tivoli Enterprise Portal user ID that receives control after a launch from an external application must be pre-authorized to access the target managed system and workspaces. The user ID also must be authorized to issue any necessary take action commands.

User ID for Take Action commands

When Tivoli Enterprise Portal sends a Take Action command to a managed system the user ID might or might not be checked for authority to perform the action. In the simplest case, the command is sent to the managed system and executed using the user ID under which the agent is running. The Tivoli Enterprise Portal user ID is sent along with the action command in these contexts:

- On-demand: user ID currently logged on
- Situation action: user ID of the last person to update the situation
- Workflow action: user ID of the last person to update the policy

However, the ID is ignored by the managed system unless told otherwise by a command prefix. These are command handlers implemented in the Tivoli Monitoring products to control whether the Tivoli Enterprise Portal user ID should be validated before passing the command to the agent for execution.

Command prefix

When a command prefix is present in the Take Action, the agent passes the command to the application handler rather than executing the command.

The syntax of the prefix and take action command is

productcode:CNPUserID:command and the agent routes it to the application for execution. The application is free to execute the command with whatever user ID is appropriate. In the case of OMEGAMON XE for WebSphere MQ, the Tivoli Enterprise Portal user ID is used.

If the special prefix is missing, the agent executes the command with the user ID under which the agent is running.

Most Tivoli Monitoring do not employ a command prefix. Tivoli Monitoring for WebSphere MQ does and, in fact, prepends any on-demand Take Action commands with a hidden **MQ:CNPUserID:** prefix, although you cannot see it.

UNIX setuid command

In addition to the command prefix and security exit, UNIX offers another option: a setuid command, which causes the process to dynamically


change its userid. Thus, the agent could be changed to set the ID to the value passed as a parameter, issue the command, then change the user ID back again after the command is issued.

Troubleshooting logon error messages


Logon prompts and progress messages are displayed in the Logon window status bar. If a user cannot log on, a message is displayed.

If a user cannot log on, one of the following messages is displayed:

Failed connection to Tivoli Enterprise Portal Server

1. On the system where the Tivoli Enterprise Portal Server is installed, click **Start -> Programs -> IBM Tivoli Monitoring ->  Manage Tivoli Monitoring Services**.
2. Optional: Right-click the Tivoli Enterprise Portal Server entry and click **Change Startup**. In the window that opens, select ☒ **System Account** and ☒ **Allow Service to Interact with Desktop** and click **OK**.
This opens a command line window when the Tivoli Enterprise Portal Server is started and displays the internal commands.
3. Ensure that the Tivoli Enterprise Portal Server is started:
 - If it is started, recycle it.
 - If it is stopped, start it.
4. If you are still unable to connect, review the following information. If it does not apply to your situation, contact IBM Software Support.

If you are running in browser mode and going across networks to reach the Tivoli Enterprise Portal Server, it is possible the host name cannot be resolved by the network system. If this is the case, doing the following should resolve the problem:

1. On the system where the Tivoli Enterprise Portal Server is installed, click **Start -> Programs -> IBM Tivoli Monitoring ->  Manage Tivoli Monitoring Services**.
2. Right-click the Tivoli Enterprise Portal Browser service and click **Reconfigure**.
3. Change the host name to the IP address in two places:
In the **Launch URL** field, change *hostname* in `http://hostname:1920///cnp/client` to the IP address of the Tivoli Enterprise Portal Server. For example, `http://10.21.2.166:1920///cnp/client`.
In the **Host** field, change the host name to the IP address of the Tivoli Enterprise Portal Server.
4. Click **OK**.
5. Start Tivoli Enterprise Portal browser mode using the IP address instead of the host name.
6. If you are still unable to connect, contact IBM Software Support.



Logon password has expired

If the hub is set to Validate Users, then passwords are required. Passwords must follow the security requirements of your organization. If this includes periodic password changes, you might get this message while attempting to log on to the . Should this happen, you need to change your system password before you can log on. For example, on Windows this means changing the password through the Administrative Tools User Accounts.

User authorization has failed -OR- Unable to process logon request

Tivoli Enterprise Portal uses the TEPS database to locally validate users. If your hub monitoring server is set for user validation (Windows default), the user ID is also validated at the monitoring server to verify the password.


The portal server did not validate the user credentials as entered. For the “Unable to process logon request” message, the portal server was able to validate the user credentials but did not complete the logon request. In either case, have the user try logging on again. If the message is displayed again, do the following:

1. On the system where the monitoring server is installed, ensure that the server is running in  Manage Tivoli Monitoring Services.
2. If the monitoring server is running, ensure that the user ID has been defined in Tivoli Enterprise Portal: Click  **Administer Users**, then find the ID in the **Users** list.
3. If the user has been defined, check if host level security was turned on for the hub monitoring server and that the user ID has been authorized to the host environment:

In  Manage Tivoli Monitoring Services, right-click **Tivoli Enterprise Monitoring Server**, and click **Reconfigure**. If host level security has been configured, the **Security: Validate User** box is selected.

If the monitoring server has been configured to Validate User, the user ID for Tivoli Enterprise Portal must also be added to the network domain user accounts or to the operating system where the monitoring server is installed, including a password.

4. Try logging on to Tivoli Enterprise Portal with the user ID in question.
5. If you cannot log on to Tivoli Enterprise Portal and the monitoring server is running properly, the problem might be with the Tivoli Enterprise Portal Server. Try recycling the portal server. If the user is still unable to log on, contact IBM Software Support.

This message is also displayed after a retry period of several minutes (the default is 10 minutes and can be changed through  Manage Tivoli Monitoring Services) where the status bar shows **Validating user credentials** continuously. This can be a symptom that the monitoring server is stopped.

Chapter 7. Customizing event integration with Tivoli Enterprise Console

If your monitoring environment includes the Tivoli Enterprise Console Event Server and situation event forwarding has been configured on the hub, you can forward situation events to that server. The *IBM Tivoli Monitoring: Installation and Setup Guide* provides the instructions to enable situation event forwarding: configuring the event server to receive the events, installing the event synchronization component on the event server, enabling situation forwarding on the hub monitoring server, and defining a default event integration facility (EIF) destination.

Default mapping of situation events to Tivoli Enterprise Console events

This section provides information about attribute mapping of situation events to Tivoli Enterprise Console events. You can use this mapping information when you forward a situation event to the Tivoli Enterprise Console and you want to write correlation rules in the Tivoli Enterprise Console.

The situation event forwarder generates a Tivoli Enterprise Console event with an event class based on the attribute group associated with the situation. When the situation event is forwarded to the event server the associated generated event class inherits event class attribute definitions (either directly or indirectly) from the parent: *Omegamon_Base* class. Because Tivoli Enterprise Console uses hierarchical event classes, use the *Omegamon_Base* parent class when you want to write a rule for all situation events that you forward to the event server.

Omegamon_Base is described as follows:

```
Omegamon_Base ISA EVENT
DEFINES {
    cms_hostname: STRING;
    cms_port: STRING;
    integration_type: STRING;
    master_reset_flag: STRING;
    appl_label: STRING;
    situation_name: STRING;
    situation_origin: STRING;
    situation_displayitem: STRING;
    situation_time: STRING;
    situation_status: STRING;
    situation_eventdata: STRING;
    situation_type: STRING;
    situation_thrnode: STRING;
    situation_group: STRING;
    situation_fullname: STRING; }; END;
```

In specialized cases where a situation event is mapped into an existing Tivoli Enterprise Console event class and the event hierarchy cannot be modified (*Omegamon_Base* cannot be added to the hierarchy) it is important that the slots from *Omegamon_Base* be included in the existing event class or in a class somewhere in the hierarchy. This mechanism is not preferred since it does not allow a rule to recognize the presence of *Omegamon_Base* in the event hierarchy.

As part of the generic mapping for these situations, the IBM Tivoli Monitoring event forwarder assigns associated values for attributed defined in the event class attributes when forwarding an event to the Tivoli Enterprise Console Event Server. In addition to these event class attributes, values are assigned to the following attributes inherited from the EVENT class, if available: source, hostname, fqhostname, origin, sub_origin, adapter_host, origin, severity, and message attributes that are inherited from the base EVENT class.

Table 14. Tivoli Enterprise Console event class attributes

Event class attributes	Values and meaning
adapter_host	Base EVENT class attribute. Same as hostname (see below). This is application-specific data related to the event, if any.
appl_label	Reserved for future use.
cms_hostname	TCP/IP host name of the Tivoli Enterprise Monitoring Server that forwards the event.
cms_port	The monitoring server port on which the web service is listening.
sqhostname	Base EVENT class attribute that contains the fully qualified hostname, if available.
hostname	Base EVENT class attribute that contains the TCP/IP hostname of the managed system where the event originates, if available.
integration_type	Indicator to help Tivoli Enterprise Console performance. <ul style="list-style-type: none"> • N for a new event, the first time the event is raised • U for update event, subsequent event status changes
master_reset_flag	Master reset indicator set for master reset events. Value is NULL for all other events: <ul style="list-style-type: none"> • R for Tivoli Enterprise Monitoring Server recycle master_reset • S for hotstandby master_reset
msg	Base EVENT class attribute that contains the situation name and formula.
origin	Base EVENT class attribute contained in the TCP/IP address of the managed system where the event originates, if available. The address is in dotted-decimal format.
severity	Base EVENT class attribute that contains the resolved severity.
situation_displayitem	Display item of associated situation, if available.
situation_eventdata	Raw situation event data starting from the second event data row, if any. Event data attributes are in key-value pair format. The event data can be truncated because the event integration facility (EIF) imposes a 2 KB size limit.
situation_group	One or more situation group names (up to 5) that the situation is a member of.
situation_fullname	Displayed name of the associated situation.
situation_name	Unique identifier given to the situation.
situation_origin	Managed system name where the situation event originated. It has the same value as sub_source.
situation_status	Current status of the situation event.
situation_time	Timestamp of the situation event.

Table 14. Tivoli Enterprise Console event class attributes (continued)

Event class attributes	Values and meaning
situation_type	Situation event type S for sampled event; P for pure event.
situation_thrnode	Reserved for future use.
source	Base EVENT class attribute that contains ITM
sub_origin	Base EVENT class attribute. This is the same as the managed system name for the associated situation_displayitem, if any.
sub_source	Base EVENT class attribute that contains the origin managed system name for the associated situation.

Expanding a generic event message situation description

The message slot gives you a descriptive way of looking at an event in the Tivoli Enterprise Console.

The situation name alone does not provide detailed event identification where there are large numbers of like-events from various sources. Rather, the situation name in the message slot sent from the hub monitoring server to the event server is expanded to include the following event attributes:

Situation-Name [(formula) ON Managed-System-Name ON DISPLAY-ITEM (threshold Name-Value pairs)]

where:

Situation-Name

The name of the situation.

formula

The formula tells how the situation is evaluated.

Managed-System-Name

The agent or the managed system.

DISPLAY-ITEM

The identifier that triggered the situation if there is more than one instance. This is optional and is used only if a display item is specified in the situation definition.

threshold Name-Value pairs

The raw data that the situation uses to evaluate whether it is triggered.

Examples:

```
NT_Critical_Process [(Process_CPU > 4 AND Thread_Count > 50)
ON IBM-AGX02:NT
(Process_CPU = 8 AND Thread_Count = 56)]
```

```
NT_Disk_Full [(Free_Megabytes < 1000000)
ON "IBM-AGX02:NT"
ON D: (Free_Megabytes = 100)]
```

Generic mapping for agent specific slots

Generic mapping identifies the target event class based on information out of a situation that is triggered and forwarded to the event server.

The event class name of the Tivoli Enterprise Console event is derived from the attribute group associated with the situation. It is a combination of **ITM_** plus the attribute group name associated with the situation. For example, a situation using the **NT_Process** attribute group will generate a Tivoli Enterprise Console event with class *ITM_NT_Process*.

Note: Some agents have very long attribute group names, which might cause the generated event class name to exceed the limit imposed by the event server. In these cases, the event class name will be a combination of **ITM_** plus the table name of the attribute group.

Additional event slot values are populated with situation attribute values from the situation event data. The slot names are the attribute names after special character processing.

For example, a situation using the **Process_CPU** attribute causes generation of a slot **process_cpu** in the Tivoli Enterprise Console event. In case the attribute name conflicts with the slot names in Tivoli Enterprise Console **EVENT** class or **Omegamon_Base** class, the *applname* associated with the attribute group, for example: *knt_*, is pre-pended to the attribute name to form the slot name.

For complex situations, the situation definition can involve more than one attribute group. In this case, the Tivoli Enterprise Console event class used is derived from the first attribute group encountered in the situation event data of the triggering situation. The exception is when the first attribute group found is **Local_Time** or **Universal_Time**; then it is passed over and the next different attribute group, if any, will be used.

For example, if a situation is written for the **NT_Process** and **NT_System** attribute groups, **NT_Process** being the first attribute group, the Tivoli Enterprise Console event class *ITM_NT_Process* is used. Additional event slots are generated based on the attributes of the attribute group selected.

Table 15. Special characters for attribute groups and names in Tivoli Enterprise Console events generated from forwarded situation events.

Character:	Converts to:
<uppercase> (applies only to attribute name)	<lowercase> (applies only to attribute name)
% percent sign	pct_
I/O	io
R/3	r3
/ forward slash	_per_
\ backward slash	_ (underscore)
<space>	_ (underscore)
(open parenthesis) close parenthesis	_ (underscore)
< open pointed bracket > close pointed bracket	_ (underscore)

Note: After special character processing, the leading and trailing underscore in the final event class or slot name, if any, will be removed.

Assigning severity for Tivoli Enterprise Console events

With the release of IBM Tivoli Monitoring Version 6.2, the `tecserver.txt` file is no longer used. If you are upgrading from a previous release, the information specified in an existing `tecserver.txt` file is automatically migrated into the situation definitions the first time the hub Tivoli Enterprise Monitoring Server is started with Tivoli Enterprise Console event forwarding enabled.

The severity of a Tivoli Enterprise Console event associated with a situation can be directly specified under the EIF tab of the Situation editor on the Tivoli Enterprise Portal. If no Tivoli Enterprise Console severity is specified for a situation, the event forwarder attempts to derive a severity from the suffix of the situation name using the following rule:

Table 16. Situation name suffix mapping to Tivoli Enterprise Console event severity

Situation name suffix	Assigned Tivoli Enterprise Console severity
Warn or _Warning	WARNING
Cri, _Crit, _Critical	CRITICAL
none of the above	UNKNOWN

Localizing message slots

Edit the `KMS_OMTEC_GLOBALIZATION_LOC` variable to enable globalization of the EIF event message slots that get mapped to alert summaries by the Tivoli Enterprise Console event server.

About this task

Some products ship with event mapping files and language bundles. The message slots for these defined Tivoli Enterprise Console events are globalized. The language selection is done through an environment variable called `KMS_OMTEC_GLOBALIZATION_LOC`.

By default, this variable is set to American English and the message slots are filled with the American English messages. Edit the variable to enable one of the language packs that are installed in your environment.

1. On the computer where the Hub Tivoli Enterprise Monitoring Server is installed, open the `KBBENV` file:
 - **Windows** Start Manage Tivoli Monitoring Services, right-click **Tivoli Enterprise Monitoring Server** and click **Advanced** → **Edit ENV file**.
 - **Linux** **UNIX** In a text editor, open the `<install_dir>/config/<tems_name>_ms_<address>.cfg` file, where `<tems_name>` is the value supplied during the monitoring server configuration, and `<address>` is the IP address or fully qualified name of the computer.
2. Locate (or add) the `KMS_OMTEC_GLOBALIZATION_LOC` environment variable and enter the desired language and country code, where `xx` is the language and `XX` is the country code: `de_DE`, `en_US`, `en_GB`, `es_ES`, `fr_FR`, `it_IT`, `ja_JP`, `ko_KR`, `pt_BR`, `zh_CN`, or `zh_TW` (such as `pt_BR` for Brazilian Portuguese or `zh_CN` for Simplified Chinese).
`KMS_OMTEC_GLOBALIZATION_LOC=xx_XX`
3. Save and close the monitoring server environment file.

Situation event statuses and Tivoli Enterprise Console event generation

The following table describes the meaning of the situation event statuses and the setting of the common slots in the generated Tivoli Enterprise Console event.

situation is true

integration_type: N, the first time the situation is true; U, all subsequent times
situation_status: Y
situation_name: Name of the situation
situation_display_item: Value of the attribute that was selected as the display item in the situation definition, if any.
master_reset_flag: None

situation reset (no longer true)

integration_type: U
situation_status: N
situation_name: Name of the situation
situation_display_item: Value of attribute selected as display item in the situation definition, if any.
master_reset_flag: None

acknowledge

integration_type: U
situation_status: A
situation_name: Name of the situation
situation_display_item: Value of the attribute that was selected as the display item in the situation definition, if any.
master_reset_flag: None

situation start

integration_type: None
situation_status: S
situation_name: Name of the situation
situation_display_item: None
master_reset_flag: None
No Tivoli Enterprise Console event is forwarded.

situation stop

integration_type: U
situation_status: P
situation_name: Name of the situation
situation_display_item: None
master_reset_flag: None
All opened situation events that originated from this monitoring server will be closed on the event server.

situation startup error

integration_type: None
situation_status: X

situation_name: Name of the situation
situation_display_item: None
master_reset_flag: None
No Tivoli Enterprise Console event is forwarded.

acknowledge expired

integration_type: U
situation_status: F
situation_name: Name of the situation
situation_display_item: Value of the attribute that was selected as the display item in the situation definition, if any.
master_reset_flag: None
Expiration that was specified in the acknowledge has expired.

resurface

integration_type: U
situation_status: E
situation_name: Name of situation
situation_display_item: Value of the attribute that was selected as the display item in the situation definition, if any.
master_reset_flag: None
The acknowledgement was removed before it had expired and the situation is still true.

hub start

integration_type: N
situation_status: None
situation_name: ""
situation_display_item: None
master_reset_flag: R
"Master reset" causes the event server to close all opened situation events from this hub monitoring server (cms_hostname value).

hub restart

integration_type: N
situation_status:
situation_name: ""
situation_display_item: None
master_reset_flag: R
"Master reset" causes the event server to close all opened situation events from this hub monitoring server (cms_hostname value).

hub Standby failover

integration_type: N
situation_status: None
situation_name: ""
situation_display_item: None
master_reset_flag: S

“Master reset” causes the event server to close all opened situation events from the old primary hub monitoring server. The name of the old primary hub is in the `situation_origin` slot.

Note: The `integration_type` value is solely used by the Tivoli Enterprise Console synchronization rule to improve its performance. It has no other meaning related with the event.

Synchronizing situation events

Checking the Tivoli Enterprise Console event cache

The event server rules event cache must be large enough to contain the volume of events expected at any given time. To check the rules cache size for a running event server, run the following the Tivoli Enterprise Console command:

```
wlsesvrcfg -c
```

To set this rules cache size, run the Tivoli Enterprise Console command:

```
wsetesvrcfg -c number_of_events
```

Note: For more information regarding these two commands, see the *Command and Task Reference* at the IBM Tivoli Enterprise Console information center.

If the rules event cache become full, the Tivoli Enterprise Console rules engine generates a `TEC_Notice` event, `Rule Cache full: forced cleaning`, indicating that 5 percent of the events from the cache were removed. Events are removed in order by age, with the oldest events removed first allowing newer events to be processed.

When the hub monitoring server forwards a status update for a situation event previously forwarded to the Tivoli Enterprise Console Event Server, if the original situation event is deleted from the rules event cache, then a `TEC_ITM_OM_Situation_Sync_Error` event is generated to indicate that the monitoring server and the event server are out of synchronization.

When using any Tivoli Enterprise Console viewer to acknowledge or close any situation event, if the situation event has been deleted from the rules event cache, the status change is not processed by the Tivoli Enterprise Console rules engine. Also, the situation event update is not forwarded to the originating Tivoli Enterprise Monitoring Server. This behavior results from the Tivoli Enterprise Console rules engine not processing any event status changes for any event not contained in the rules event cache. In this case, the event status change is updated only in the Tivoli Enterprise Console database.

Both situations can be remedied by performing a Tivoli Enterprise Console server configuration parameters analysis and performance analysis to determine the optimal configuration parameter settings and desired performance requirements. Refer to the IBM Tivoli Enterprise Console® information center, *IBM Tivoli Enterprise Console Rule Developer's Guide (Version 3.9)*, Rule Engine Concepts chapter for more information.

Changing the configuration of the event synchronization on the event server

About this task

If you want to change any of the settings for the event synchronization on the event server, use the **sitconfig.sh** command. You have two options for running this command:

- Manually modify the configuration file for event synchronization (named `situpdate.conf` by default and located in the and located in the `/etc/TME/TEC/OM_TEC` directory on operating systems such as UNIX, and the `%SystemDrive%\Program Files\TME\TEC\OM_TEC\etc` directory on Windows), and then run the following command:
`sitconfig.sh update <config_filename>`
- Run the **sitconfig.sh** command directly, specifying only those settings that you want to change. See *IBM Tivoli Monitoring: Command Reference* for the full syntax of this command.

After you change the configuration of the event synchronization, you must manually stop and restart the Situation Update Forwarder process from the `$BINDIR/TME/TEC/OM_TEC/bin` directory with the **stopSUF** and **startSUF** commands.

Defining additional monitoring servers for the event synchronization on the event server

About this task

For each monitoring server that is forwarding situation events to the event server, you must have the necessary server information defined so that the Situation Update Forwarder process forwards situation event updates to the originating monitoring server. Run the following command to add new monitoring server information:

```
sitconfsvruser.sh add serverid=server userid=user password=password
```

where:

serverid=server

The fully qualified host name of the monitoring server.

userid=user

The user ID to access the computer where the monitoring server is running.

password=password

The password to access the computer.

Repeat this command for each monitoring server that you want to add.

After you change the configuration of the event synchronization, you must manually stop and restart the Situation Update Forwarder process from the `$BINDIR/TME/TEC/OM_TEC/bin` directory with the **stopSUF** and **startSUF** commands (.cmd file extension on Windows; .sh on operating systems such as UNIX).

Closing sampled events

About this task

When a situation event from a sampled situation is forwarded to the Tivoli Enterprise Console Event Server and that event is subsequently closed in the event server, the behavior of event synchronization is to send a request to the Tivoli Enterprise Monitoring Server to acknowledge the situation with a specified timeout. The reason for this is because closing events from sampled situations causes problems with the situation's ability to fire after the close in IBM Tivoli Monitoring.

If the acknowledgement of the situation expires and the situation is still true, then a new situation event is opened in the Tivoli Enterprise Console. If the situation becomes false, then it resets itself in IBM Tivoli Monitoring and the event remains closed in the Tivoli Enterprise Console.

The default acknowledgement expiration time is 59 minutes. This can be changed in the situation timeouts configuration file on the event server (`sit_timeouts.conf`). Also, expiration times for individual situations can be configured in this file. After editing this file, you can have the expire times dynamically loaded into the Tivoli Enterprise Console rule using the `sitconfig.sh` refresh command in `$BINDIR/TME/TEC/OM_TEC/bin`.

Changing rule set parameters for `omegamon.rls`

The `omegamon.rls` rule set has parameters that you can edit, according to your environment, to tune performance or to set your own customized values. These parameters allow you to write and customize Tivoli Enterprise Console rules. During installation, you can choose the location of the rule base. Otherwise, you can use the `wrb -lscurrb -path` to find the current rule base.

Here are some reasons why you might want to change the behavior of the rule:

- For `omegamon.rls`, *omegamon_admin* is the name of the rule set but you can name your rule set after your administrator's name or some other value.
- Similarly, *sit_ack_expired_def_action* is set to REJECT by default. This means that whenever a situation event acknowledgement expires in the Tivoli Enterprise Portal and the event becomes OPEN in the portal, the Tivoli Enterprise Console event server rejects this action and re-acknowledges the event in the portal. You have the option of accepting the change that was initiated by the portal and change the status in the Tivoli Enterprise Console instead.

These are the user-configurable parameters:

omegamon_admin

This is the identifier used when a rule defined in this rule set closes an event. This identifier is used to differentiate close operations that were originated automatically rather than by the console operator.

omsync_timeout

This attribute sets the period in seconds that you must wait to distinguish between the synchronization of single or multiple events. The default timeout is 3 seconds.

omsync_maxentries

This attribute sets the maximum number of events allowed per batch. Default batch size is 100 events.

Warning: Setting this value less than 20 events might cause contentions within the Tivoli Enterprise Console task process, causing poor performance of events synchronized back to the Tivoli Enterprise Monitoring Server.

sit_resurface_def_action

This attribute determines the default action of the rules in case a situation update event arrives from Tivoli Enterprise Monitoring Server to resurface or reopen an event that has already been acknowledged. The two possible values are ACCEPT and REJECT. The default is ACCEPT.

sit_ack_expired_def_action

This attribute determines the default action of the rules in case a situation update event arrives from the Tivoli Enterprise Monitoring Server to reopen an event that has already been acknowledged. This happens when a situation's acknowledgement in the monitoring server expires and the situation event is reopened. The two possible values are ACCEPT and REJECT. The default is REJECT.

sf_check_timer

This attribute specifies the interval at which the state of the situation update forwarder is checked. It reads events from the cache files and send them to the Tivoli Enterprise Monitoring Server using Web Services. The default is 10 minutes.

After modifying any configuration parameters and saving `omegamon.rls`, you must recompile and reload the rule base and recycle the event server. To recompile the rule base, enter the following command, where `Rulebase_Name` is the name of the actively loaded rule base containing the `omegamon.rls` rule set:

```
wrb -comprules Rulebase_Name
```

To reload the rule base, issue the following command:

```
wrb -loadrb Rulebase_Name
```

To stop the Event server, issue the following command:

```
wstopesvr
```

To restart the Event server issue the following command:

```
wstartesvr
```

For more information regarding the **wrb**, **wstopesvr**, and **wstartesvr** commands, see the *Command and Task Reference* at the IBM Tivoli Enterprise Console information center.

Tuning considerations

Integration parameters supporting actions at the Tivoli Enterprise Console event console that are reflected at the Tivoli Enterprise Portal event console provide good response times with a reasonable system resource investment.

The tuning parameters to consider include:

- `omsync_timeout` in the `omegamon.rls` with a default of 3 seconds.
- `PollingInterval` in event synchronization with a default of 3 seconds.
- Tivoli Enterprise Console event console refresh interval with a default 60 seconds

- Tivoli Enterprise Portal

Note: Shorter intervals result in the consumption of more system resources.

The delivery time of situation changes from the Tivoli Enterprise Console event console to the event console results from `omsync_timeout` and `PollingInterval` settings working in parallel. To improve the response time, you can reduce these settings down to a minimum of 1 second. .

You can adjust the refresh interval for both consoles:

- For the Tivoli Enterprise Console, change the allowable range using the Tivoli Enterprise Console event console – Configuration. In the subsequent Event View displays, adjust the preferences.
- For the Tivoli Enterprise Portal

Using the Rules Check utility

The Rules Check utility provides you with the ability to assess the impact on an existing set of rules whenever the designs of BAROC (Basic Recorder of Objects in C) event classes are changed. This utility allows you to verify which rules might have been impacted by these event class definition changes.

There are two important sets of files that are used and required by the Rules Check utility to check the possible impacts of event classes design changes to the rules:

- BAROC Event Classes Definition files:

Tivoli Enterprise Console class definitions are hierarchical in nature with inheritances. One class can inherit from another class, and all attributes from the parent class are available in the child class. The EVENT class is the base Tivoli Enterprise Console class. The other classes usually derive from the Tivoli Enterprise Console EVENT class.

In Tivoli Enterprise Console, the BAROC Event Class Definition files (*.baroc files) are located in the actively loaded rule base's TEC_CLASSES subdirectory. They provide the event class definitions used by the Tivoli Enterprise Console Server. Although the tool is closely integrated with Tivoli Enterprise Console and uses the active rule base's TEC_CLASSES subdirectory by default input, the tool is not dependent on this subdirectory, and accepts as alternative input any other directory that contains the correct BAROC files and to which the user has read privileges.

- Rules files:

The Tivoli Enterprise Console product rule language also supports the inheritance nature of the Tivoli Enterprise Console class definitions. When a predicate in the Tivoli Enterprise Console rule is looking for a particular class, all classes that inherit from that particular class also satisfy the rule predicate.

In Tivoli Enterprise Console, the Ruleset files (*.rls files) are located in the actively loaded rule base's TEC_RULES subdirectory. They provide the rulesets and are deployed to the Tivoli Enterprise Console Server. Although the tool is closely integrated with Tivoli Enterprise Console and uses the active rule base's TEC_RULES subdirectory by default input, the tool is not dependent on this subdirectory. The tool accepts as an alternative input any other directory that contains the correct rulesets and to which the user has read privileges.

The Rules Check utility is shipped with IBM Tivoli Monitoring. This utility is installed in the \$BINDIR/TME/TEC/OM_TEC/bin directory as part of the Tivoli

Enterprise Console Event Synchronization Install. It does not require any specific directory configuration if the required privileges for access to the input and output files are granted.

To run the Rules Check command you must have:

- Read access to the *.rls and *.baroc files that are used as inputs.
- Write access to the output that is used to store the results of the check.
- Tivoli Enterprise Console administrator authority.
- When no -cd and -rd options are specified, the user issuing the command must have the proper TME® authorization, and verify the level of wrb subcommands that are required.

To run the Rules Check utility and see sample output, refer to the *Command Reference*.

Editing the Event Integration Facility configuration

Edit the **Tivoli Event Integration Facility** EIF file to customize the configuration such as to specify up to five failover EIF servers or to adjust size of the event cache.




Before you begin

When the **Tivoli Event Integration Facility** (EIF) has been enabled on the hub monitoring server and the default EIF server (Tivoli Enterprise Console Event Server or Netcool/OMNIBus EIF probe) and port number have been specified, the EIF configuration file is updated with the information. This is the default EIF receiver of forwarded situation events.

See the *Event Integration Facility Reference* at the IBM Tivoli Enterprise Console information center for more details on the parameters and values. See the *IBM Tivoli Monitoring Installation and Setup Guide* for instructions on configuring the monitoring server to enable the Tivoli Event Integration Facility.

About this task

Take these steps to edit the EIF configuration file:

1. Open the om_tec.config file:
 - a.  In the Manage Tivoli Monitoring Services window, right-click Tivoli Enterprise Monitoring Server and click **Advanced** → **Edit EIF Configuration**.
 - b.   Open <install_dir>/tables/host name/TECLIB/om_tec.config in a text editor.
2. Edit any of the event server configuration parameters for the event integration facility.
3. When you are finished editing om_tec.config, save the file.
4. You need to restart the monitoring server or, alternatively, you can use the **refreshTECinfo** command to complete the updates without having to restart the monitoring server. To use this command, log in to the command-line interface with **tacmd login**, then run **tacmd refreshTECinfo -t eif** to complete the EIF configuration. See the *IBM Tivoli Monitoring Command Reference*.

Results

Table 17. Supported Tivoli Enterprise Console event server configuration parameters for the event integration facility (EIF)

Parameters	Value	Remarks
ServerLocation=	tec_server_addr	host name or ip address of the event server. To provide event failover, you can indicate up to five default event servers, separating each with a comma. When the default event server is unavailable, the situation event goes to the next server in the list.
ServerPort=	[port:0]	The event server listening port, which is 5529 by default. Specify 0 if the event server uses the port mapper. If you specified multiple server locations, add the corresponding port numbers here, separating each with a comma.
EventMaxSize=	4096	Maximum number of characters allowed in the event. This is disabled by default. To enable it, remove the # (pound symbol) at the beginning of the line.
RetryInterval=	5	The number of times to retry connection with the event server before returning an error.
getport_total_timeout_usec=	50500	How many seconds to continue attempting to connect to the event server port before timing out. The default is 14 hours.
NO_UTF8_CONVERSION=	YES	Events are already in UTF8, no conversion is needed. Must be set to YES.
ConnectionMode=	co	The connection mode.
BufferEvents=	YES	Whether the EIF buffers the event. This must be set to YES.
BufEvtMaxSize=	4096	Maximum size of the event cache. The default is initially 4096 KB and you can change it here..
BufEvtPath=	./TECLIB/om_tec.cache	Path of the event cache file. The default is ./TECLIB/om_tec.cache.
FilterMode=	OUT	Enable event filtering. This is set to OUT by default.
Filter:	Class=ITM_Generic; master_reset_flag="";	To filter out specific classes, use this keyword. By default, situation events of the class ITM_Generic and those that send no master reset flag are not forwarded.

Specifying EIF forwarding for a situation event

When the Tivoli Enterprise Monitoring Server has been configured for the **Tivoli Event Integration Facility**, all situation events are forwarded to the event receiver. Use the Tivoli Enterprise Portal Situation editor to override this default for individual situations.

Before you begin

One of the Tivoli Enterprise Monitoring Server configuration options is **Tivoli Event Integration Facility**. When this option is enabled, the default EIF receiver is specified in the event server Location and Port Number window that opens (and described in the *IBM Tivoli Monitoring: Installation and Setup Guide*). Thereafter, all

situation events are forwarded to the EIF receiver by default, using the severity derived from the situation name or the  Critical severity if none can be derived.

You can override this default for individual situations through the EIF tab of the Situation editor in the .






Up to eight event destinations can be specified for a forwarded situation event. The event destination association can be done on the EIF tab of the Situation editor. The event destinations must be predefined with the `tacmd createEventDest` command. The commands are described in the *IBM Tivoli Monitoring: Command Reference*. Changes to the list of event destinations will not be effective until either the `tacmd refreshTECinfo` command is issued or the hub monitoring server is recycled.

Alternate event destinations that were specified in the `tecserver.txt` from earlier releases will be defined as valid event destinations automatically as part of the `tecserver.txt` file migration.

If no event destinations are specified for a Tivoli Enterprise Console event, it will be forwarded to all defined default destinations.

About this task

Complete these steps to specify the destination EIF receiver and severity for forwarded events:

1. In the Tivoli Enterprise Portal Navigator view, either click right-click the Navigator item that the situation is associated with and click  **Situations** or click  **Situation Editor** in the main toolbar.
2. Select the situation to forward.
3. Click the  **EIF** tab.
4. Select ☒ **Forward Events to an EIF Receiver** to specify that an EIF event is sent for each event that opens for this situation.
5. Select the **EIF Severity** to apply to forwarded events for this situation. <Default EIF Severity> uses the same severity as is used for the situation at this Navigator item.
6. Assign any other EIF receivers as well as or instead of the <Default EIF Receiver>:
 - To add a destination, select it from the **Available EIF Receivers** list and  move to the Assigned list. (After selecting the first destination, you can use Ctrl+click to select other destinations or Shift+click to select all destinations between the first selection and this one.)
 - To remove a destination, select it from the **Assigned EIF Receivers** list and  move to the Available list.

The **Available EIF Receivers** list shows all of the defined EIF destinations that were defined through Manage Tivoli Monitoring Services or with the `tacmd createEventDest` command.

7. Save the situation definition with your changes by clicking **Apply**, to keep the Situation editor open, or **OK**, to close the Situation editor.

Customizing the event message

From the Situation editor EIF tab, you can create map definitions for situation events sent to the EIF receiver. The EIF Slot Customization window, which is opened from the EIF tab, is used to customize how situation events are mapped to forwarded EIF events, thus overriding the default mapping between situation events and events forwarded to the Tivoli Enterprise Console Event Server.

When the Base Slot name is msg, the Literal value column is used for the message template. The message template consists of fix message text and variable substitution references, or *symbols*. The symbol can refer to common or event slot data or a special reference to the situation formula. Common slots are those that are included in all forwarded events, such as situation_name; event slots are those specific to the situation. Syntax:

- For an event slot, use the fully qualified attribute name (\$Attribute_Table.Attribute_Name\$)
- For a common slot, use the variable name that is not fully qualified (no . periods) unless it is the situation symbol
- For a situation formula, use \$formula\$

These characters are not supported: < less than, > greater than, " quote, ' single quote, and & ampersand. This column is available only if no value is selected in the Mapped attribute column. See the Tivoli Enterprise Portal online help or the *IBM Tivoli Monitoring: CandleNet Portal User's Guide* for more information.

Updating the XML used by the MCS Attribute Service

The default XML file used by the Multiple Console Support (MCS) Attribute Service includes only the event classes defined in the BAROC files within the TECLIB branch of the hub Tivoli Enterprise Monitoring Server installation. Generate a new XML file for EIF Slot Customization whenever a new type of agent is added into the Tivoli Management Services infrastructure or when a new event class has been added into the Tivoli Event Console Event Server.

Before you begin

If an event class specified for a rule is not found within the current event class definition set and you continue building the rule with the current definition set, any unrecognized event classes will be removed from the rule.

The EIF Event Customization facility uses the MCS Attribute Service to present a list of predefined event classes in the **Event class name** list of the EIF Slot Customization window (available through the EIF tab of the Situation editor). Only the event classes belonging to the OS agents are predefined and they are in an MCS Attribute Service jar file. When a new type of agent is added into the Tivoli Management Services infrastructure or a new event class is added, you need to generate a new MCS XML file and point the Tivoli Enterprise Portal Server to the new XML file before the new event classes will appear in the **Event class name** list.

Note: The definitions in MCS XML file supersede those defined in the shipped MCS Attribute Services jar file (they are not merged). To obtain a MCS XML file that contains both the event classes definitions of the OS agents as well as the new

agent, be sure all the BAROC definitions for the OS agents and new agent are loaded at the Tivoli Event Console Server before running the TEDGEN utility to generate the MCS XML file.

About this task

On the computer where the Tivoli Event Console Event Server is located, install TEDGEN from the tool directory of the event server installation media. Then complete these steps to create a new XML file with the TEDGEN tool:

1. Issue the wrb -imprbclass command to import the BAROC file that is installed with newly added agent, and the OS agents if it is not already installed:

```
wrb -imprbclass <class_file> [ -encoding <encoding> ]  
[-before <class_file> | -after <class_file>] [-force] <rule_base>
```

2. Issue the wrb -loadrb command to reload the rulebase:

```
wrb -loadrb <rule_base>
```

3. Stop and restart the event server by running these commands:

```
wstopesvr  
wstartesvr
```

4. Issue the TEDGEN command to generate the XML file:

```
tedgen [ -bcDir <baroc_classes_directory> | -rbName <rule_base_name> ]  
-id <server_id> -xmlPath <output_xml_file_path>
```

5. Copy the newly generated XML file to the computer where the Tivoli Enterprise Portal Server is installed.

6. Edit the portal server environment file to specify the path to the XML file:

- a. **Windows** In the Manage Tivoli Monitoring Services window, right-click **Tivoli Enterprise Portal Server** and click **Advanced** → **Edit ENV File** to open the kfwenv file in the text editor.

UNIX Open <itm_install_dir>/config/cq.ini in a text editor.

- b. Locate the KFW_MCS_XML_FILES environment variable and type = (equal sign) followed by the path to the XML file.
- c. Save and close the environment file.

Limitations on forwarding events to the Tivoli Enterprise Console

Situation events from the Tivoli Universal Agent that get forwarded to the Tivoli Enterprise Console have some limitations.

Using the NetView console through the Tivoli Enterprise Console event viewer

About this task

You can launch the Tivoli NetView® Java console from the Tivoli Enterprise Console views, navigating from an event row to the associated network topology and diagnostics. The selected event must contain a valid host name or IP address to support the topology display of the node associated with the event. Otherwise, the standard topology view is displayed without a specific node selected. Tivoli Enterprise Console rules automatically synchronize the events forwarded by Tivoli NetView to the Tivoli Enterprise Console server. The event status updates are reflected on the system where you launch the Netview event console.

Ensure that you have `netview.rls` and `netview baroc` files in the actively loaded rule base. For details, see the *Rule Set Reference* at the IBM Tivoli Enterprise Console information center.

If you want to use the NetView console through the Tivoli Enterprise Console view in the Tivoli Enterprise Portal, you must configure the `NVWC_HOME` variable in the shell script that launches the Tivoli Enterprise Portal client, to point to the installation directory of NetView Web Console.

To set the `NVWC_HOME` variable:

```
Windows <itm_install_dir>\cnp\cnp.bat
```

```
Linux or UNIX <itm_install_dir>/bin/cnp.sh
```

The NetView Web Console must be installed on the computer where the Tivoli Enterprise Portal client is running to launch the NetView console from Tivoli Enterprise Console view.

See the Tivoli Enterprise Console product documentation for more detailed information about using the NetView console.

Chapter 8. Customizing event integration with Tivoli Netcool/OMNIBus

Use the Tivoli Event Integration Facility (EIF) interface to forward enterprise situation events to OMNIBus. The events are received by the Netcool/OMNIBus Probe for Tivoli EIF, which maps them to OMNIBus events and then inserts them into the OMNIBus server.

Updates to those events are also sent to OMNIBus. When an OMNIBus user acknowledges, closes, or reopens a forwarded event, OMNIBus sends those changes back to the monitoring server that forwarded them.

The *IBM Tivoli Monitoring Installation and Setup Guide* provides the instructions to enable situation event forwarding: configuring the OMNIBus server for program execution from scripts, updating the OMNIBus db schema, configuring the EIF probe, enabling situation forwarding on the hub monitoring server, and defining a default event integration facility (EIF) destination.

Default mapping of situation events to OMNIBus alerts

This topic provides information about attribute mapping of situation events to OMNIBus alerts. You can use this mapping information when you forward a situation event to the Tivoli Netcool/OMNIBus ObjectServer and want to write probe rules or SQL procedures and triggers in the ObjectServer.

The situation event forwarder generates an event integration facility (EIF) event with an event class based on the attribute group used in the situation. When the situation event is forwarded to the ObjectServer, the Tivoli Netcool/OMNIBus EIF probe translates the EIF event into an OMNIBus alert format. The EIF event contains all of the attributes described by the parent *Omegamon_Base* class.

Omegamon_Base is described as follows:

```
Omegamon_Base ISA EVENT
DEFINES {
    cms_hostname: STRING;
    cms_port: STRING;
    integration_type: STRING;
    master_reset_flag: STRING;
    appl_label: STRING;
    situation_name: STRING;
    situation_origin: STRING;
    situation_displayitem: STRING;
    situation_time: STRING;
    situation_status: STRING;
    situation_eventdata: STRING;
    situation_type: STRING;
    situation_thrnode: STRING;
    situation_group: STRING;
    situation_fullname: STRING; }; END;
```

As part of the generic mapping for these situations, the IBM Tivoli Monitoring event forwarder assigns associated values for each of the event class attributes

when forwarding an event to OMNIBus. In addition to these event class attributes, values are assigned to the host name, origin, severity, and message attributes that are inherited from the base EVENT class.

Table 18. Tivoli Netcool/OMNIBus ObjectServer attributes.

Attributes	Values and meaning
appl_label	Application specific data related with the event, if any.
cms_hostname	TCP/IP host name of the that forwards the event.
cms_port	The monitoring server port on which the Web service is listening.
Hostname	Base EVENT class attribute that contains the TCP/IP hostname of the managed system where the event originates, if available.
integration_type	Indicator to help OMNIBus performance. <ul style="list-style-type: none"> • N for a new event, the first time the event is raised • U for update event, subsequent event status changes
master_reset_flag	Master reset indicator set for master reset events. Value is NULL for all other events: <ul style="list-style-type: none"> • R for monitoring server recycle master_reset • S for hotstandby master_reset
msg	Base EVENT class attribute that contains the situation name and formula.
origin	Base EVENT class attribute contained in the TCP/IP address of the managed system where the event originates, if available. The address is in dotted-decimal format.
severity	Base EVENT class attribute that contains the resolved severity.
situation_displayitem	Display item of associated situation, if available.
situation_eventdata	Event data attributes in key-value pair format. The event data can be truncated because the event integration facility (EIF)) imposes a 2 KB size limit.
situation_group	One or more situation group names (up to 5) that the situation is a member of.
situation_fullname	Displayed name of the associated situation.
situation_name	Unique identifier given to the situation.
situation_name	Name of the associated situation.
situation_origin	Managed system name where the situation event originated. It has the same value as sub_source.
situation_status	Current status of the situation event.
situation_time	Timestamp of the situation event.
situation_type	Indicator of whether the ITM situation which caused the event is a sampled or pure situation.
situation_thrnode	The hub or remote Tivoli Enterprise Monitoring Server through which the event was forwarded.
source	Base EVENT class attribute that contains "ITM"
sub_source	Base EVENT class attribute that contains the origin managed system name for the associated situation.

The Tivoli Netcool/OMNIBus EIF probe maps the attributes of the situation event into ObjectServer attributes, which are defined in the `alerts.status` table of the ObjectServer.

Table 19. Mapping of situation attributes to OMNIBus attributes

Situation attribute	OMNIBus Attribute
situation_name + situation_origin +situation_displayitem + event_class	(for ITMProblem)
situation_name + situation_origin + situation_displayitem + event_class + ITMResolution	Identifier (for ITMResolution)
situation_name	AlertKey
situation_origin	Node
situation_origin	NodeAlias
source	Agent
default	Type (20) (for ITMProblem)
situation_status = "P" and integration_type = "U"	Type (21) (for ITMResolution)
situation_status = "D" and integration_type = "U"	Type (21) (for ITMResolution)
situation_status = "N" and integration_type = "U"	Type (21) (for ITMResolution)
situation_displayitem	ITMDisplayItem
situation_status	ITMStatus
situation_thruNode	ITMThruNode
situation_time	ITMTime
situation_type	ITMSitType
situation_eventdata	ITMEventData
cms_hostname	ITMHostname
master_reset_flag	ITMResetFlag
integration_type	ITMIntType
event_class	AlertGroup
msg	Summary
"tivoli_eif probe on "+hostname()	Manager
6601	Class
severity FATAL / 60 = Critical CRITICAL / 50 = Critical MINOR / 40 = Minor WARNING / 30 = Warning UNKNOWN / 10 = Indeterminate	Severity
getdate	LastOccurrence/FirstOccurrence
date	TECDate
repeat_count	TECRepeatCount
fqhhostname	TECFQHostname
hostname	TECHostname

Expanding a generic event message situation description

The OMNIBus EIF probe maps the message slot in the EIF event sent from the Tivoli Enterprise Monitoring Server into the summary attribute of the ObjectServer. The summary attribute gives you a descriptive way of looking at an alert in OMNIBus.

The situation name alone does not provide detailed event identification where there are large numbers of like-events from various sources. The situation name in the summary attribute sent from the hub monitoring server to the ObjectServer is expanded to include the following event attributes:

Situation-Name [(formula) ON Managed-System-Name ON DISPLAY-ITEM (threshold Name-Value pairs)]

where:

Situation-Name

The name of the situation.

formula

The formula tells how the situation is evaluated.

Managed-System-Name

The agent or the managed system.

DISPLAY-ITEM

The identifier that triggered the situation if there is more than one instance. This is optional and is used only if a display item is specified in the situation definition.

threshold Name-Value pairs

The raw data that the situation uses to evaluate whether it is triggered.

Examples:

```
NT_Critical_Process [(Process_CPU > 4 AND Thread_Count > 50)
ON IBM-AGX02:NT
(Process_CPU = 8 AND Thread_Count = 56)]
```

```
NT_Disk_Full [(Free_Megabytes < 1000000)
ON "IBM-AGX02:NT"
ON D: (Free_Megabytes = 100)]
```

Generic mapping for agent specific slots

Generic mapping identifies the target event class based on information out of a situation that is triggered and forwarded to the OMNIBus EIF probe.

For situation events that do not have a mapping specified for the forwarded event, an event is generated with a unique class based on the attribute group used in the situation. The class name of the EIF event is a combination of **ITM_** plus the attribute group name associated with the situation.

For example, a situation using the **NT_Process** attribute group will generate a Tivoli Enterprise Console event with class **ITM_NT_Process**.

Additional event slot values are populated with situation attribute values from the situation event data. The slot names are the attribute names. These additional slot values can be used to write additional OMNibus EIF proberules.

For example, a situation using the Process_CPU attribute causes generation of a slot process_cpu in the EIF event forwarded to the OMNibus EIF probe. In case the attribute name conflicts with the slot names in Tivoli Enterprise Console EVENT class or Omegamon_Base class, the *applname* associated with the attribute group, for example: *knt_*, is prepended to the attribute name to form the slot name.

For complex situations, the situation definition can involve more than one attribute group. In this case, the EIF event class used is derived from the first attribute group encountered in the situation event data of the triggering situation. For example, if a situation is written for the NT_Process and NT_System attribute groups, NT_Process being the first attribute group, the EIF event class *ITM_NT_Process* is used. Additional event slots are generated based on the attributes of the attribute group only.

Table 20. Special characters for attribute groups and names in EIF events generated from forwarded situation events

Character:	Converts to:
<uppercase> (applies only to attribute name)	<lowercase> (applies only to attribute name)
% percent sign	pct_
I/O	io
/ forward slash	_per_
\ backward slash	_ (underscore)
<space>	_ (underscore)
(open parenthesis) close parenthesis	_ (underscore)
< open pointed bracket > close pointed bracket	_ (underscore)

All strings and time stamp types are mapped to STRING types, and all integer types are mapped to INTEGER in the event class definition. No default values are assigned to the attribute slots. Attributes that have a specified non-zero scale/precision value are mapped to the string type of REAL.

Note: If you are mapping from an attribute to a slot and the resulting slot name has a trailing underscore, the trailing underscore is removed in the final slot name, which never has a trailing underscore.

Localizing alert summaries

Edit the KMS_OMTEC_GLOBALIZATION_LOC variable to enable globalization of the EIF event message slots that get mapped to alert summaries by the OMNibus EIF probe.

About this task

By default, this variable is set to American English and the message slots are filled with the American English messages. Take these steps to edit the variable to enable

any language packs that are installed in your environment

1. On the computer where the Hub Tivoli Enterprise Monitoring Server is installed, open the KBBENV file:
 - **Windows** Start Manage Tivoli Monitoring Services, right-click **Tivoli Enterprise Monitoring Server** and click **Advanced** → **Edit ENV file**.
 - **Linux** **UNIX** In a text editor, open the `<install_dir>/config/<tems_name>_ms_<address>.cfg` file, where `<tems_name>` is the value supplied during the monitoring server configuration, and `<address>` is the IP address or fully qualified name of the computer.
2. Locate (or add) the `KMS_OMTEC_GLOBALIZATION_LOC` environment variable and enter the desired language and country code, where `xx` is the language and `XX` is the country code: `de_DE`, `en_US`, `en_GB`, `es_ES`, `fr_FR`, `it_IT`, `ja_JP`, `ko_KR`, `pt_BR`, `zh_CN`, or `zh_TW` (such as `pt_BR` for Brazilian Portuguese or `zh_CN` for Simplified Chinese).
`KMS_OMTEC_GLOBALIZATION_LOC=xx_XX`
3. Save and close the monitoring server environment file.

Synchronizing situation events

Changing the configuration of the event synchronization on the event server

About this task

If you want to change any of the settings for the event synchronization on the event server, use the **sitconfig.sh** command. You have two options for running this command:

- Manually modify the configuration file for event synchronization (named `situpdate.conf` by default and located in `<event_sync_installdir>/etc` and then run: **sitconfig.sh update <config_filename>**
- Run the **sitconfig.sh** command directly, specifying only those settings that you want to change. See *IBM Tivoli Monitoring: Command Reference* for the full syntax of this command.

After you change the configuration of the event synchronization, you must manually stop and restart the Situation Update Forwarder process from the `<event_sync_installdir>/bin` directory with `stopSUF.sh` and `startSUF.sh`.

Defining additional monitoring servers for the event synchronization on the ObjectServer

About this task

For each monitoring server that is forwarding situation events to the ObjectServer, you must have the necessary server information defined so that the Situation Update Forwarder process forwards situation event updates to the originating monitoring server. Run the following command to add new monitoring server information:

```
sitconfsvruser.sh add serverid=server userid=user password=password
```

where:

serverid=*server*

The fully qualified host name of the monitoring server.

userid=*user*

The user ID to access the computer where the monitoring server is running.

password=*password*

The password to access the computer.

Repeat this command for each monitoring server that you want to add.

After you change the configuration of the event synchronization, you must manually stop and restart the Situation Update Forwarder process from the `<event_sync_installdir>/bin` directory: On Windows, it is `stopSUF.cmd` and `startSUF.cmd`; on operating systems such as UNIX, it is `stopSUF.sh` and `startSUF.sh`.

Deleting or clearing sampled events

About this task

When a situation event from a sampled situation is forwarded to the Tivoli Netcool/OMNIbus ObjectServer and that event is subsequently deleted or cleared in the ObjectServer, the behavior of event synchronization is to send a request to the Tivoli Enterprise Monitoring Server to acknowledge the situation with a specified timeout. The reason for this is because closing sampled situations causes problems with the situation's ability to fire after the close in IBM Tivoli Monitoring.

If the acknowledgement of the situation expires and the situation is still true, then a new situation event is opened in the ObjectServer. If the situation becomes false, then it resets itself in IBM Tivoli Monitoring, and the event remains closed in the ObjectServer.

The default acknowledgement expire time for sampled situations is 59 minutes. This can be changed in the situation timeouts configuration file on the ObjectServer (`sit_timeouts.conf`). Also, expiration times for individual situations can be configured in this file. After editing this file, you can have the expire times dynamically loaded into the ObjectServer using the `sitconf.sh refresh` (UNIX) or `sitconf.cmd refresh` (Windows) command in `<event_sync_installdir>/bin`.

Customizing the OMNIbus configuration

The procedure `get_config_parms` in the `<event_sync_install_dir>/omnibus/itm_proc.sql` file defines three configuration parameters:

```
set sit_ack_expired_def_action = 'REJECT'
set sit_resurface_def_action = 'ACCEPT'
set situpdate_conf_file = 'situpdate.conf'
```

The variable `sit_ack_expired_def_action` defines the action to be taken for an event by the OMNIbus server when acknowledgement expiration information is received for an event from a monitoring server. The default action is to Reject the request. OMNIbus sends information to change the state of the event to Acknowledge back to the monitoring server. If you would like to change the action taken by the OMNIbus server to Accept the acknowledgement expiration, modify the statement to `set sit_ack_expired_def_action = 'ACCEPT'`.

The variable `sit_resurface_def_action` defines the action to be taken by the OMNIbus server when a situation event has resurfaced. The default action of the OMNIbus

server is to Accept this request and Deacknowledge the event. If you would like to change the action taken by OMNIBus server to Reject the resurface of the event, modify the statement to set `sit_resurface_def_action = 'REJECT'`. OMNIBus then sends information back to the monitoring server to change the state of the event back to Acknowledge.

The variable `situpdate_conf_file` specifies the name of the configuration file to be used by the SitUpdate Forwarder. If you would like to change the name of the configuration file, modify the statement to set `situpdate_conf_file = 'newname.conf'`.

After modifying `itm_proc.sql`, issue the following command:

Windows

```
%OMNIHOME%\..\bin\redist\isql -U <username>
-P <password>
-S <server_name>
< <path_to_file>\itm_proc.sql
```

Linux

or

UNIX

```
$OMNIHOME/bin/nco_sql -user <username>
-password <password>
-server <server_name>
< <path_to_file>/itm_proc.sql
```

Editing the Event Integration Facility configuration

Edit the **Tivoli Event Integration Facility** EIF file to customize the configuration such as to specify up to five failover EIF servers or to adjust size of the event cache.

Before you begin

When the **Tivoli Event Integration Facility** (EIF) has been enabled on the hub monitoring server and the default EIF server (Tivoli Enterprise Console Event Server or Netcool/OMNIBus EIF probe) and port number have been specified, the EIF configuration file is updated with the information. This is the default EIF receiver of forwarded situation events.

See the *IBM Tivoli Monitoring Installation and Setup Guide* for instructions on configuring the monitoring server to enable the Tivoli Event Integration Facility.

About this task

Take these steps to edit the EIF configuration file:

1. Open the `om_tec.config` file:
 - a. **Windows** In the Manage Tivoli Monitoring Services window, right-click Tivoli Enterprise Monitoring Server and click **Advanced** → **Edit EIF Configuration**.
 - b. **Linux** **or** **UNIX** Open `<install_dir>/tables/host name/TECLIB/om_tec.config` in a text editor.
2. Edit any of the event server configuration parameters for the event integration facility.
3. When you are finished editing `om_tec.config`, save the file.
4. You need to restart the monitoring server or, alternatively, you can use the **refreshTECinfo** command to complete the updates without having to restart the monitoring server. To use this command, log in to the command-line interface with **tacmd login**, then run **tacmd refreshTECinfo -t eif** to complete the EIF configuration. See the *IBM Tivoli Monitoring Command Reference*.

Results

Table 21. Supported OMNibus EIF probe configuration parameters for the event integration facility (EIF)



Parameters	Value	Remarks
ServerLocation=	tec_server_addr	host name or ip address of OMNibus EIF probe. To provide event failover, you can indicate up to five default event servers, separating each with a comma. When the default event server is unavailable, the situation event goes to the next server in the list.
ServerPort=	[port:0]	OMNibus EIF probe listening port, which is 5529 by default. Specify 0 if the OMNibus EIF probe uses the port mapper. If you specified multiple server locations, add the corresponding port numbers here, separating each with a comma.
EventMaxSize=	4096	Maximum number of characters allowed in the event. This is disabled by default. To enable it, remove the # (pound symbol) at the beginning of the line.
RetryInterval=	5	The number of times to retry connection with the event server before returning an error.
getport_total_timeout_usec=	50500	How many seconds to continue attempting to connect to the event server port before timing out. The default is 14 hours.
NO_UTF8_CONVERSION=	YES	Events are already in UTF8, no conversion is needed. Must be set to YES.
ConnectionMode=	co	The connection mode.
BufferEvents=	YES	Whether the EIF buffers the event. This must be set to YES.
BufEvtMaxSize=	4096	Maximum size of the event cache. The default is initially 4096 KB and you can change it here..
BufEvtPath=	./TECLIB/om_tec.cache	Path of the event cache file. The default is ./TECLIB/om_tec.cache.
FilterMode=	OUT	Enable event filtering. This is set to OUT by default.
Filter:	Class=ITM_Generic; master_reset_flag="";	To filter out specific classes, use this keyword. By default, situation events of the class ITM_Generic and those that send no master reset flag are not forwarded.




Specifying situation events that send an OMNibus event

When the Tivoli Enterprise Monitoring Server has been configured for the **Tivoli Event Integration Facility**, all situation events are forwarded to the Tivoli Netcool/OMNibus Probe for Tivoli EIF. Use the Tivoli Enterprise Portal Situation editor to override this default for individual situations.

About this task

Complete these steps to specify the destination EIF receiver and severity for a forwarded event:

1. In the Tivoli Enterprise Portal Navigator view, either click right-click the Navigator item that the situation is associated with and click  **Situations** or click  **Situation Editor** in the main toolbar.

2. Select the situation to forward.
3. Click the  EIF tab.
4. Select ☒ **Forward Events to an EIF Receiver** to specify that an EIF event is sent for each event that opens for this situation.
5. Select the **EIF Severity** to apply to forwarded events for this situation. <Default EIF Severity> uses the same severity as is used for the situation at this Navigator item.
6. Assign any other EIF receivers as well as or instead of the <Default EIF Receiver>:
 - To add a destination, select it from the **Available EIF Receivers** list and  move to the Assigned list. (After selecting the first destination, you can use Ctrl+click to select other destinations or Shift+click to select all destinations between the first selection and this one.)
 - To remove a destination, select it from the **Assigned EIF Receivers** list and  move to the Available list.

The **Available EIF Receivers** list shows all of the defined EIF destinations that were defined through Manage Tivoli Monitoring Services or with the **tacmd createEventDest** command.
7. Save the situation definition with your changes by clicking **Apply**, to keep the Situation editor open, or **OK**, to close the Situation editor.

Customizing the event message

From the Situation editor EIF tab, you can create map definitions for situation events sent to the EIF receiver. The EIF Slot Customization window, which is opened from the EIF tab, is used to customize how situation events are mapped to forwarded EIF events, thus overriding the default mapping between situation events and events forwarded to the Tivoli Netcool/OMNIBus ObjectServer.

When the Base Slot name is msg, the Literal value column is used for the message template. The message template consists of fix message text and variable substitution references, or *symbols*. The symbol can refer to common or event slot data or a special reference to the situation formula. Common slots are those that are included in all forwarded events, such as `situation_name`; event slots are those specific to the situation. Syntax:

- For an event slot, use the fully qualified attribute name (`$Attribute_Table.Attribute_Name$`)
- For a common slot, use the variable name that is not fully qualified (no . periods) unless it is the situation symbol
- For a situation formula, use `$formula$`

These characters are not supported: < less than, > greater than, " quote, ' single quote, and & ampersand. This column is available only if no value is selected in the Mapped attribute column.

See the Tivoli Enterprise Portal online help or the *IBM Tivoli Monitoring: CandleNet Portal User's Guide* for more information.

Chapter 9. Configuring connectors for the common event console

The *common event console* is a Tivoli Enterprise Portal view that provides a single, integrated display of events from multiple event systems. In one table, the common event console presents events from the event systems, and users can sort, filter, and perform actions on these events. The following event systems are supported:

- IBM Tivoli Monitoring
- IBM Tivoli Enterprise Console
- IBM Tivoli Netcool/OMNIBus

A *common event connector* (frequently called a *connector*) is software that enables the integrated display of events from multiple event systems in the common event console. A connector retrieves event data from an event system and sends user-initiated actions to be run in that event system. For example, if you perform an action on a Tivoli Enterprise Console or Netcool/OMNIBus event in the common event console, the associated common event console connector sends that action to the originating event system (Tivoli Enterprise Console or Netcool/OMNIBus) for execution. To have the events from a specific event system displayed in the common event console, you must configure a connector for that event system and set a variable in the environment file.

Common Event Console Configuration window

About this task

Use the Common Event Console Configuration window to configure a common event console connector for each of your event system instances. Because the connector for the IBM Tivoli Monitoring product is pre-configured when you install the product, the common event console includes situation events by default. However, to have IBM Tivoli Enterprise Console or IBM Tivoli Netcool/OMNIBus events included in the common event console, you must configure a connector for each of these event systems after you install the IBM Tivoli Monitoring product. This configuration includes specifying which event systems are used to obtain events for display in the common event console. You might also want to change some of the configuration values for the IBM Tivoli Monitoring connector.

To configure connectors, open the Common Event Console Configuration window by performing the following steps on the computer where the is installed:

Windows

1. Select **Start** → **Programs** → **IBM Tivoli Monitoring** → Manage Tivoli Monitoring Services.
2. In the Manage Tivoli Monitoring Services window, right-click **Tivoli Enterprise Portal Server**.
3. In the menu, click **Reconfigure**.
4. In the first TEP Server Configuration window, click **OK**.
5. In the second TEP Server Configuration window, click **OK**.

6. Click **No** in answer to the question “Do you want to reconfigure the warehouse connection information for the Tivoli Enterprise Portal Server?”

Linux

or

UNIX

1. At the command prompt, change directory (cd) to `<install_dir>/bin` and enter `./itmcmd manage`.
2. In the Manage Tivoli Monitoring Services window, right-click **Tivoli Enterprise Portal Server**.
3. In the pop-up menu, click **Configure**.

The stops and, after a moment, the Common Event Console Configuration window opens with the following tabs:

- ITM Connector
- TEC Connector
- OMNIBus Connector
- Names of Extra Columns

ITM Connector tab

Click the **ITM Connector** tab to view or change the information for the IBM Tivoli Monitoring connector. Because the Tivoli Monitoring event system has a single hub Tivoli Enterprise Monitoring Server, you configure only one IBM Tivoli Monitoring connector.

The following information defines the IBM Tivoli Monitoring connector:

Enable this connector

You can choose Yes or No. A value of Yes means that IBM Tivoli Monitoring events are available in the common event console.

Connector name

The name that is to be displayed in the common event console for this connector.

Maximum number of events for this connector

The maximum number of events that are to be available in the common event console for this connector.

View closed events

You can choose Yes or No. A value of Yes means that closed events for this connector are available in the common event console.

TEC Connector tab

Click the **TEC Connector** tab to view or change the information for an IBM Tivoli Enterprise Console connector. To have the events from a Tivoli Enterprise Console server displayed in the common event console, you must configure an IBM Tivoli Enterprise Console connector.

To configure a connector, click **New**. The resulting TEC Connector page contains the following information that defines an IBM Tivoli Enterprise Console connector:

Connector name

The name that is to be displayed in the common event console for this connector.

Maximum number of events for this connector

The maximum number of events that are to be available in the common event console for this connector.

Computer name of event system

The computer name of the event system that is associated with this connector.

Port number of event system

The object dispatcher (oserv) port number, typically 94. This is the port that the connector uses to retrieve events from the Tivoli Enterprise Console event system.

This is not the port used to connect to the Tivoli Enterprise Console event server (5529 by default).

User name for accessing event system

The user name that is used when accessing the event system that is associated with this connector.

Password

The password that is associated with the user name.

Event group that defines events for common event console

The Tivoli Enterprise Console event group that defines which events are available in the common event console.

If you do not specify an event group, all Tivoli Enterprise Console events are available in the common event console.

If you want to restrict events further, you can also define a clause in the **SQL WHERE clause that restricts events for common event console** field.

SQL WHERE clause that restricts events for common event console

This clause can be applied only to the part of an event that is built from the Tivoli Enterprise Console base attribute table. For example, status <> 30 causes all events with a status that is not equal to 30 to be available in the common event console.

If you do not define a clause, all Tivoli Enterprise Console events are available in the common event console, unless they are excluded by an event group that you specified in the **Event group that defines events for common event console** field.

View closed events

You can choose Yes or No. A value of Yes means that closed events for this connector are available in the common event console.

Time interval (in minutes) for polling event system

The number of minutes between each poll of the event system for new or changed events.

Time interval (in minutes) for synchronizing events

The number of minutes between each poll of the event system to determine which events have been deleted.

Time interval (in seconds) between reconnection attempts

The number of seconds of delay between reconnection attempts when the connector loses its connection to the event system.

Number of reconnection attempts

The maximum number of consecutive reconnection attempts to make if the connector loses its connection to the event system.

If this value is set to -1 and the connector loses its connection, the connector attempts to reconnect indefinitely.

Information for extra table columns

The common event console includes five extra table columns that you can customize. In the remaining fields on this TEC Connector page, you can define the Tivoli Enterprise Console attribute type and attribute name that identify the attribute that is to be mapped to each of these customizable columns.

For the attribute type, you can choose one of the following values:

- Base, which means that the attribute is from the Tivoli Enterprise Console base attribute table.
- Extended, which means that the attribute is from the Tivoli Enterprise Console extended attribute table.

OMNibus Connector tab

Click the **OMNibus Connector** tab to view or change the information for an IBM Tivoli Netcool/OMNibus connector. To have the events from a Tivoli Netcool/OMNibus ObjectServer displayed in the common event console, you must configure an IBM Tivoli Netcool/OMNibus connector.

To configure a connector, click **New**. The resulting OMNibus Connector page contains the following information that defines an IBM Tivoli Netcool/OMNibus connector:

Connector name

The name that is to be displayed in the common event console for this connector.

Maximum number of events for this connector

The maximum number of events that are to be available in the common event console for this connector.

Computer name of event system

The computer name of the event system that is associated with this connector.

Port number of event system

The ObjectServer port number (usually 4100), which this connector uses to retrieve events from the Tivoli Netcool/OMNibus event system.

User name for accessing event system

The user name that is used when accessing the event system that is associated with this connector.

Password

The password that is associated with the user name.

SQL WHERE clause that restricts events for common event console

This clause can be applied only to the part of an event that is built from the Tivoli Netcool/OMNibus alerts.status table. For example, Severity <> 0 causes all events with a severity that is not equal to 0 to be available in the common event console.

If you do not define a clause, all Tivoli Netcool/OMNibus events are available in the common event console.

View cleared events

You can choose Yes or No. A value of Yes means that cleared events for this connector are available in the common event console.

Time interval (in minutes) for polling event system

The number of minutes between each poll of the event system for new or changed events.

The Tivoli Netcool/OMNIbus ObjectServer automatically sends new or changed events to the common event console as they become available. Therefore, the primary purpose of this checking is to ensure that the server and the connection to the server are functioning properly.

Time interval (in seconds) between reconnection attempts

The number of seconds of delay between reconnection attempts when the connector loses its connection to the event system.

Number of reconnection attempts

The maximum number of consecutive reconnection attempts to make if the connector loses its connection to the event system.

If this value is set to 0 and the connector loses its connection, the connector remains inoperable indefinitely.

If this value is set to -1 and the connector loses its connection, the connector attempts to reconnect indefinitely.

Information for extra table columns

The common event console includes five extra table columns that you can customize. In the remaining fields on this page, you can define the Tivoli Netcool/OMNIbus field type and field name that identify the field that is to be mapped to each of these customizable columns.

For the field type, you can choose one of the following values:

- `alerts.status`, which means that the field contains data from the `alerts.status` table in the Tivoli Netcool/OMNIbus ObjectServer.
- `alerts.details`, which means that the field contains data from the `alerts.details` table in the Tivoli Netcool/OMNIbus ObjectServer.
- `Extended`, which means that the field contains extended attributes from a Tivoli Enterprise Console event that has been forwarded to the Tivoli Netcool/OMNIbus event system.

Names of Extra Columns tab

The common event console includes five extra table columns that you can customize. By default, the following names are used for these columns:

- Extra Column 1
- Extra Column 2
- Extra Column 3
- Extra Column 4
- Extra Column 5

Click the **Names of Extra Columns** tab to view or change the names of these columns.

When you define a Tivoli Enterprise Console or Tivoli Netcool/OMNIbus connector, you can define the information that is to be mapped to each of these customizable columns.

Purpose of extra table columns

The common event console displays only a basic set of information from the Tivoli Enterprise Console base attribute table and the Tivoli Netcool/OMNIBus alerts.status and alerts.details tables. If, for example, you want to see an additional attribute named “origin” from a Tivoli Enterprise Console event, you can perform the following steps:

1. In the **Attribute type for extra column 1** field on the TEC Connector page, choose the attribute type, for example, base.
2. In the **Attribute name for extra column 1** field on the TEC Connector page, enter the attribute name, for example, origin.
3. In the **Name of extra column 1** field on the Names of Extra Columns page, enter the name that you want to use for the column that you have customized. For example, you might enter Origin.

In the “Origin” column for each row that is a Tivoli Enterprise Console event, the common event console displays the value of the origin attribute.

TEC Connector tab: defining information for extra table columns

In the following fields on the TEC Connector page, you define the information that is to be mapped to the customizable columns:

- Attribute type for extra column 1
- Attribute name for extra column 1
- Attribute type for extra column 2
- Attribute name for extra column 2
- Attribute type for extra column 3
- Attribute name for extra column 3
- Attribute type for extra column 4
- Attribute name for extra column 4
- Attribute type for extra column 5
- Attribute name for extra column 5

OMNIBus Connector tab: defining information for extra table columns

In the following fields on the OMNIBus Connector page, you define the information that is to be mapped to the customizable columns:

- Field type for extra column 1
- Field name for extra column 1
- Field type for extra column 2
- Field name for extra column 2
- Field type for extra column 3
- Field name for extra column 3
- Field type for extra column 4
- Field name for extra column 4
- Field type for extra column 5
- Field name for extra column 5

Best practices when event synchronization is used

In your environment, if Tivoli Monitoring events are forwarded to the Tivoli Enterprise Console or Tivoli Netcool/OMNIBus event system for the purpose of event synchronization, you should configure the common event connectors to retrieve only one copy of the same event to avoid having duplicate event information in the common event console.

The following steps outline a way to restrict the common event console to include only the Tivoli Enterprise Console or Tivoli Netcool/OMNIBus events that do not originate as Tivoli Monitoring events:

When Tivoli Monitoring events are forwarded to Tivoli Enterprise Console event system

1. On the Tivoli Enterprise Console server, create an event group that defines only the Tivoli Enterprise Console events that do not originate as Tivoli Monitoring events and is named, for example, `All_but_ITM`.
2. When you configure a TEC Connector, type `All_but_ITM` in the **Event group that defines events for common event console** field.
3. When you configure the ITM Connector, click **Yes** in the **Enable this connector** field.

When Tivoli Monitoring events are forwarded to Tivoli Netcool/OMNIBus event system

1. When you configure an OMNIBus Connector, type `ITMStatus = ''` in the **SQL WHERE clause that restricts events for common event console** field, where `''` is two single quotes with no space between them. This clause restricts the Tivoli Netcool/OMNIBus events in the common event console to only those that do not originate as Tivoli Monitoring events.
2. When you configure the ITM Connector, click **Yes** in the **Enable this connector** field.

The resulting configuration causes the common event console to retrieve Tivoli Monitoring events directly from the Tivoli Monitoring event system rather than the Tivoli Enterprise Console or Tivoli Netcool/OMNIBus event system, which prevents you from having duplicate event information in the common event console.

Troubleshooting problems with connection to Tivoli Enterprise Console server on Linux systems

The following problem can occur on Linux systems:

Problem

The Tivoli Enterprise Console connector cannot connect to the Tivoli Enterprise Console server. Therefore, Tivoli Enterprise Console events are not available in the common event console.

Explanation

The `/etc/hosts` file on the computer where the Tivoli Enterprise Portal server is installed must include the local host with the correct IP address. The following line shows approximately what the default Linux configuration is:

```
127.0.0.1 my_hostname localhost
```

The default Linux configuration causes the connection request to be sent to the Tivoli Enterprise Console server with the 127.0.0.1 address, which is not the correct IP address of the computer where the Tivoli Enterprise Portal server is installed. For the Tivoli Enterprise Portal server to connect, it must be able to do a reverse lookup.

Solution

Ensure that the /etc/hosts file includes the local host with the correct IP address. The following two lines show approximately what the correct Linux configuration is, where *xxx.xxx.xxx.xxx* is the IP address of the computer where the Tivoli Enterprise Portal server is installed:

```
127.0.0.1 localhost
xxx.xxx.xxx.xxx my_hostname
```

Chapter 10. Working with monitoring agents

The Navigator Physical view in the Tivoli Enterprise Portal displays all the managed systems in your monitored network. From the Navigator pop-up menu, you can remotely deploy and manage Tivoli Enterprise Monitoring Agents that run on distributed operating systems and that connect to a Tivoli Enterprise Monitoring Server that runs on a distributed operating system.

Before you can remotely install and configure agents, each target computer must have an operating system (OS) agent installed. Monitoring agents that do not support the remote agent deployment feature will not display the **Add Managed System**, **Configure**, and **Remove** options in the Navigator pop-up menu. The types of managed systems that you can add to a computer depend on what agent bundles are in the *agent depot* on the monitoring server to which the OS agent is connected.

The *IBM Tivoli Monitoring: Installation and Setup Guide* tells how establish an agent depot on the monitoring server and an OS agent on each computer where agents will be deployed. After that has been done, use the topics here to start, stop, configure, and remove a monitored agent from the managed network.

Also described is how to change the monitoring server designation for an agent.

To manage agents through the Tivoli Enterprise Portal, your user ID requires **Manage** permission for the **Agent Management** authority.

Adding an agent through the Tivoli Enterprise Portal

Use the Tivoli Enterprise Portal client to add individual managed systems to the monitored network.

Before you begin


The types of agents that you can remotely install on a computer depend on what agent bundles are in the agent depot on the monitoring server to which the OS agent is connected. See the *IBM Tivoli Monitoring: Installation and Setup Guide* for more information.

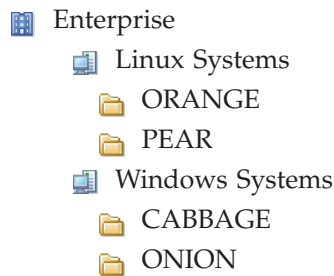
Before you can remotely install and configure distributed monitoring agents on a computer, the computer must have an OS monitoring agent installed for the operating system the monitoring product will run under. When the OS monitoring agents have been installed, the Navigator Physical view adds an item for each online managed system.




To use this feature, your user ID must have **Manage** permission for **Agent Management**.

About this task

Follow these instructions to install and configure managed systems through the Tivoli Enterprise Portal:

1. In the Navigator physical view, right-click the  system-level item for the computer where you want to install the monitoring agent. In this example, the computers named ORANGE, PEAR, CABBAGE, and ONION are available.



2. Click  **Add Managed System** to open the Select a Monitoring Agent window. The agents shown in this list are those available for the operating system on which this computer runs. The two-digit version number is followed by a two-digit release number and a modification number of up to five digits.
3. Highlight the name of the monitoring agent to install and click **OK**. The New Managed System Configuration window opens with an **Agent** tab. Any other tabbed pages are specific to the agent. Move the mouse pointer over a field to see hover help.
4. Complete the fields to configure the agent, clicking **Next** and **Back** to move among the tabbed pages.
5. On the **Agent** page, establish the operating system user ID under which the agent will run on the managed system. *Windows*: Either accept the default to start the managed system with your user ID (you can also select the check box to **Allow service to interact with desktop** to enable remote control); or select **Use this account** and fill in the user name and password under which the agent will run.
Non-Windows: Enter the **Username** under which the agent will run and the **Group name**.
6. Click **Finish** to complete the managed system configuration. If any of the information provided is invalid, you will receive an error message and be returned to the configuration window. Check your entries and edit as appropriate to configure correctly. Installation and setup begins and might take several minutes to complete depending on your Tivoli monitoring configuration, the location of the managed system, and the type of monitoring agent.
7. After the managed system has been added to the enterprise, click  **Apply Pending Updates** in the Navigator view toolbar. The new managed system (such as  Universal Database) is displayed below the system Navigator item.

Configuring an agent through the Tivoli Enterprise Portal



The Tivoli Enterprise Portal client offers a convenient feature for configuring individual managed systems. This does not include the OS agents because they are already configured and running.

Before you begin

To use this feature, your user ID must have Manage permission for Agent Management.

About this task

To configure your monitoring agents, complete the following steps.

1. Right-click the  Navigator item for the agent you want to configure or upgrade.
2. Click  **Configure** to open the Configure Managed System window. Any tabbed pages besides **Agent** are specific to the agent. Move the mouse pointer over a field to see hover help.
3. Edit the fields to configure the agent, clicking **Next** and **Back** to move among the tabbed pages.
4. On the **Agent** page, establish the user ID that will be used to maintain the agent:

Windows:

Accept the default ☒ **Use local system account** to use your Tivoli Enterprise Portal user ID. You can also select ☐ **Allow service to interact with desktop** to enable remote control. Or select ☐ **Use this account** and fill in the user name and password under which the agent will be controlled.

Non-Windows:

Enter the **Username** under which the agent will run and the **Group name**.


5. Click **Finish** to complete the managed system configuration. If any of the information provided is invalid, you will receive an error message and be returned to the configuration window. Check your entries and edit as appropriate to configure correctly.

Starting, stopping, and recycling an agent through the Tivoli Enterprise Portal



About this task

You can start an offline managed system, or recycle or stop it through the Tivoli Enterprise Portal.

All deployment commands are passed through the operating system agent that is installed at the target computer. If an operating system agent is not installed, you cannot start or stop the deployed agent.

 To use this feature, your user ID must have **Manage** permission for **Agent Management**.



To start a monitoring agent from the Tivoli Enterprise Portal

1. In the Navigator Physical view, right-click the Navigator item of the offline  agent.
2. Click  **Start**.

The request to start the monitoring agent is sent to the monitoring server to which it is connected. Depending on your monitoring configuration, it might take a few moments before the agent starts running and to see the Navigator item enabled.



If the monitoring agent does not start and you get an error message, the computer might be unavailable.

To stop a monitoring agent from the Tivoli Enterprise Portal

1. In the Navigator physical view, right-click the  agent to stop.
2. Click  **Stop**.

The agent goes offline and the Navigator item is dimmed. The agent does not come online until you start it manually or, if it is set to start automatically, after you restart the monitoring server to which it is connected.

To recycle a monitoring agent from the Tivoli Enterprise Portal

1. In the Navigator physical view, right-click the  agent to stop.
2. Click  **Restart** to stop, then start the monitoring agent. This might take a short time depending on the network traffic.

Updating agents

Whenever a new release of a distributed agent is available, you can use remote deployment to apply the updates. You can perform the updates through the Tivoli Enterprise Portal or at the command line.

- “Updating an agent through the Tivoli Enterprise Portal”
- “Updating an agent through the command-line interface” on page 121

Updating an agent through the Tivoli Enterprise Portal

Use the Configure Managed System window in the Tivoli Enterprise Portal client to apply a patch for a monitoring agent.



Before you begin

When a new version of a distributed monitoring agent is released, you can apply the new version locally or remotely to one managed system at a time, or to many simultaneously. This capability does not apply to the OS monitoring agents, z/OS-based agents, or any products that do not support the remote agent deployment feature. As well, the agents to be updated must have been originally installed using remote agent deployment. The types of managed systems that you can add to a computer depend on what agent bundles are in the agent depot on the monitoring server to which the OS agent is connected. See the *IBM Tivoli Monitoring: Installation and Setup Guide* for more information.

Before starting the update, you must install application support on the for any agent that you are going to deploy with the procedure that follows.

About this task

Complete these steps to apply a patch for a monitoring agent through the portal client:

1. Right-click the  Navigator item for the agent that you want to upgrade.
2. Click  **Configure** to open the Configure Managed System window.
3. Click the **Agent** tab.
4. Compare the installed version of the monitoring agent with any available product updates, then highlight the row of the agent to update and click **Install Updates**.

Results

Installation of the updates begins and might take several minutes to complete. The list that displays reflects the contents of the deployment depot. If **Install Updates** is disabled, one or more of the following conditions exist:

- The depot entry does not match the product type.
- The **VVRR** fields for the agent and the depot entry are the same, where **VV** is the version number and **RR** is the revision number. For example, an entry of **0610** would prevent you from applying a fix pack intended for a version 6.2 agent.
- The depot entry is at an older version than the agent.
- The host version field of the depot entry does not contain the host platform for the agent.
- The prereq field of the depot entry does not contain an agent of the same type as the agent itself. For example, if 6.1 UD (DB2 monitoring) is the selected agent, the prereq field in the depot entry must contain a deployment bundle notation such as **ud:061000000**, which is one way to denote a patch deployment bundle.

Updating an agent through the command-line interface

About this task

Updating agents involves stopping any that are running, applying the changes, and restarting them. After determining the specifics about monitoring agents that you want to update, including the type and version, run the **tacmd updateAgent** command from the command-line interface. If a version is not specified, the agent is updated to the latest version.

Complete the following steps at a command-line interface. For reference information about this command and related commands, see the *IBM Tivoli Monitoring: Command Reference, SC32-6045*.

- Use the **tacmd login** command to log into a Tivoli Enterprise Monitoring Server.

```
tacmd login {-s|--server} [{https|http}://]HOST[:PORT] }
[{-u|--username} USERNAME]
[{-p|--password} PASSWORD]
[{-t|--timeout} TIMEOUT] [-t TIMEOUT]
```

For example, to log in to the system *ms.austin.ibm.com* with the user name *Admin* and the password *log1n*, run the following command:

```
tacmd login -s ms.austin.ibm.com -u Admin -p log1n
```

- After logging in, use the **tacmd updateAgent** command to install an agent update to a specified node.

```
tacmd updateAgent {-t|--type} TYPE {-n|--node} MANAGED-OS
[{-v|--version} VERSION] [{-f|--force}]
```

For example, the following command updates a UNIX agent (type *UX*) on *itmserver*:

```
tacmd updateagent -t UX -n itmserver:KUX -v 6111
```


Note: Use only Tivoli provided **tacmd** commands to process bundles and to execute agent deployments. Manual manipulation of the depot directory structure or the bundles and files within it is not supported and might void your warranty.

Removing an agent through the Tivoli Enterprise Portal


About this task

You can also uninstall monitoring agents from the Tivoli Enterprise Portal by stopping the agent and removing its configuration settings. After you have removed the agent from the enterprise, you can completely uninstall the agent from the managed system. When you remove an agent, it is removed from any managed system lists to which it is assigned, any situation or policy distribution lists it was on, and any custom Navigator view items to which it was assigned.

Note: If the Manage Tivoli Monitoring Services utility is running when you uninstall the agent, it is shut down automatically by the uninstallation process.

 To use this feature, your user ID must have **Manage** permission for **Agent Management**.

Complete the following steps to remove and optionally uninstall an agent:

1. Right-click the  Navigator item for the agent you want to remove.
2. Click **Remove**.
3. Click **Yes** when you are asked to confirm the removal of the agent. If you are removing an agent that has subagents, another message will ask if you want them all removed.
4. When you are asked to confirm that you want to permanently uninstall the agent, click **Yes** to uninstall or **No** to leave the agent installed on your system.

Changing the monitoring server an agent connects to

A monitored environment with multiple Tivoli Enterprise Monitoring Servers can have all or some of the agents connect to remote monitoring servers. You can change the monitoring server an agent connects to by reconfiguring it.

About this task

Complete these steps to reassign a monitoring agent to a different monitoring server:

1. In the Manage Tivoli Monitoring Services window, right-click the monitoring agent, and click **Reconfigure**.
2. Click **OK** in the first Agent Advanced Configuration window.
3. In the second Agent Advanced Configuration window, enter the **Hostname or IP Address** of the monitoring server to connect to, and change the Port number if it is different from the default 1918.

What to do next

When reconfiguring a Universal Agent to connect to a different monitoring server, start and restart all the situations that are distributed to that managed system. Otherwise, the situations that are set to autostart will fail to start and an error will occur.

Chapter 11. Agent autonomy

A Tivoli Enterprise Monitoring Agent can run independently of the Tivoli Enterprise Monitoring Server. Different levels of autonomy are available depending on the functionality you want the agent to have, resource constraints, and how much dependency you want the agent to have to the monitoring server. A Tivoli System Monitoring Agent is an OS agent that is installed and configured to have no dependency on nor any connection to a monitoring server.

Tivoli Enterprise Monitoring Agents start independently of their Tivoli Enterprise Monitoring Server and they collect data, run situations, and register events when they are disconnected from their monitoring server. This is the default behavior, which can be adjusted for greater or less autonomy. Furthermore, you can configure special XML files to define and run situations locally, to collect and save historical data locally, and to emit SNMP alerts to a receiver without requiring connection to a monitoring server.

OS agents and Agent Builder agents can also be installed and configured as Tivoli System Monitoring Agents that never connect to a monitoring server. Autonomous agents are like any other monitoring agent except that any processing that can be done only through the monitoring server is not available. As well, an autonomous agent must not be installed on the same system as a Tivoli Management Services component or a Tivoli Enterprise Monitoring Agent.

Autonomous capabilities

In addition to the built-in autonomous capability of Tivoli Enterprise Monitoring Agents and Tivoli System Monitoring Agents, you can configure special XML files to define and run situations locally, to collect and save historical data locally, and to emit SNMP alerts to a receiver without a connection to a monitoring server.

Tivoli Enterprise Monitoring Agent

Tivoli Enterprise Monitoring Agents are configured for autonomous operation by default: The agent starts and continues to run with or without connection to its monitoring server. If connection is lost to the monitoring server, the agent can continue running situations autonomously; when the agent reconnects, it will upload situation events that took place while it was disconnected. This incurs use of additional disk space at the agent.

Some situations might not be able to be evaluated completely on the agent alone and are unable to run when there is no connection to the monitoring server. For example, situations using a group function in the formula must be evaluated at the monitoring server. Even if the agent or the host system is restarted, the events are persistently preserved and are uploaded on reconnect. This happens automatically on all agents that use the Tivoli Enterprise Monitoring Agent V6.2.2 framework. No configuration changes are required. If you do not want autonomous behavior enabled for an agent, you can disable it with the `IRA_AUTONOMOUS_MODE` agent configuration parameter.

Tivoli System Monitoring Agent

Tivoli System Monitoring Agents can be installed on a computer that has

no Tivoli Management Services components or Tivoli Enterprise Monitoring Agents installed other than agents built with Tivoli Monitoring Agent Builder V6.2.2 or higher.

The Tivoli System Monitoring Agent agent is an OS agent that never connects to a monitoring server. The autonomous version of the agent uses the same agent code that is installed for a full OS agent, but Java is not used for the installation process and the configuration user interface is not provided. The resulting installation is faster and has a small installed footprint. The agent configuration is stored in a local XML file and the agent can periodically poll an SNMP source for a new configuration file.

SNMP alerts

Prior to IBM Tivoli Monitoring V.6.2.2, enterprise situation events for a Tivoli Enterprise Monitoring Agent could be forwarded to the Netcool/OMNIBus Probe for Tivoli EIF (Event Integration Facility), which maps the situation events to OMNIBus events and then inserts them into the OMNIBus server. IBM Tivoli Monitoring V.6.2.2 enables you to emit SNMP alerts for situation events to the Netcool/OMNIBus SNMP Probe.

The two methods of sending events to OMNIBus can coexist and your monitored environment can be configured for any combination thereof:

- Forward enterprise situation events to receivers such as the Tivoli Enterprise Console event server and Netcool/OMNIBus Probe for Tivoli EIF.
- Emit SNMP alerts for private situation events to receivers such as the Netcool/OMNIBus SNMP Probe.

You can create a trap configuration XML file that enables an agent to emit SNMP alerts directly to the event receiver with no routing through the monitoring server. The agent must connect to the monitoring server at least once to receive enterprise situation definitions. The user needs to place an SNMP trap configuration file in the agent installation and restart the agent to enable this function.

Tivoli Enterprise Monitoring Agents and Tivoli System Monitoring Agents can also send SNMP alerts for private situations directly to a receiver such as the Netcool/OMNIBus SNMP Probe.

If you are forwarding enterprise situation events to the Netcool/OMNIBus Probe for Tivoli EIF and emitting SNMP alerts for enterprise situation events to the Netcool/OMNIBus SNMP Probe, be aware that events for the same situation sent to both probes will not be detected as the same event by OMNIBus deduplication.

Private situations

Monitoring agents can use locally defined situations to operate fully autonomously. These locally defined *private situations* are created in a private situation definition XML file. Private situations events come directly from the monitoring agent. You need to place an SNMP trap configuration file and a private situation configuration file in the agent installation and restart the agent to enable this function. Private situations can be used on a Tivoli Enterprise Monitoring Agent; the private situations have no interaction with or reporting of any kind to the monitoring server.

Private history

Just as you can create private situations for the agents installed locally, you can configure private history for collecting short-term historical data in the

same private situation configuration file using the HISTORY element. The resulting private history binary files can be viewed through the Agent Service Interface.

Agent Service Interface

The IBM Tivoli Monitoring Service Index utility provides links to the Agent Service Interface for each monitoring agent installed locally. After logging into the operating system, you can select one of these reports: agent information, situation, or history. Additionally, you can make a service interface request directly such as to get a report of situation activity or to recycle a situation.

Configuration parameters for autonomous behavior

Use the environment file that is provided with the agent framework services to control the autonomous behavior of the Tivoli System Monitoring Agent or of the Tivoli Enterprise Monitoring Agent when it is disconnected from the Tivoli Enterprise Monitoring Server.

Start and control agent autonomous mode

The *IBM Tivoli Monitoring Installation and Setup Guide* provides instructions for installing and configuring the Tivoli System Monitoring Agent. It also has a reference of the common agent environment variables in an appendix. The following configuration parameters start and control agent autonomous mode.

CTIRA_HEARTBEAT=10

The heartbeat interval for an agent. The default is **10** minutes and can be set as low as 1 minute.

CTIRA_MAX_RECONNECT_TRIES=0

This parameter is being deprecated. Use it to specify the number of consecutive times without success the agent attempts to connect to a monitoring server before giving up and exiting. The default value of **0** means that the agent will remain started regardless of its connection status with the monitoring server.

Prior to Tivoli Monitoring V6.2.2, the default value was **720**. Along with the CTIRA_RECONNECT_WAIT default setting of **600**, the agent tries to connect to the monitoring server for 432000 seconds (5 days) before giving up and exiting. If you prefer to have the agent shut down when the reconnect limit is reached, specify the number of retries. You must also disable the agent watchdog function (disarmWatchdog command), which is described in the agent user's guide. See also "Monitoring the availability of agents" on page 182.

CTIRA_RECONNECT_WAIT=600

The number of seconds for the agent to wait between attempts to register with a Tivoli Enterprise Monitoring Server. Consider setting this to a value equivalent to the CTIRA_HEARTBEAT setting, which is specified in minutes. The default of **600** seconds is the equivalent of the default heartbeat of 10 minutes.

IRA_AUTONOMOUS_LIMIT=50

This parameter determines the number of events that can be stored at the agent when it is in autonomous mode or allocates the amount of disk space that the events can occupy. When the event limit or disk space maximum has been reached, no further events are collected. The default is

50 events or **2MB**. Specify either the total number of events or the disk space limit, where *n* is the numeric value:

n = maximum number of events (sampled and pure) that can be saved.
To estimate the space for each event, add 1200 to the average application row size.

nKB = *n* times 1024 bytes.

nMB = *n* times 1,024,000 bytes.

nGB = *n* times 1,024,000,000 bytes

IRA_AUTONOMOUS_MODE=Y

This parameter controls autonomous mode operation. By default, autonomous mode is enabled. To disable it, set this parameter to **N**.

IRA_EVENT_EXPORT_CHECKUSAGE_INTERVAL=180

Specifies the preferred interval in seconds to check if the **IRA_AUTONOMOUS_LIMIT** has been reached. The default interval is **180** seconds (3 minutes); the minimum interval that can be specified is **60** seconds.

IRA_EVENT_EXPORT_SIT_STATS=Y

You can get a report of the situation operation statistics through the Agent Service Interface. This parameter enables (Y) or disables (N) the basic situation operation statistics data collection:

Situation Name

Situation Type - Enterprise or Private

Application Name

Table Name

Sample interval

Row data size

Time stamp First Time situation started

Time stamp First Time situation raised event (passed filter)

Time stamp Last Time situation started

Time stamp Last Time situation stopped

Time stamp Last Time situation evaluated to TRUE

Time stamp Last Time situation evaluated to FALSE

Number of times situation recycled

Number of times situation in autonomous operation

Default: **Y**.

IRA_EVENT_EXPORT_SIT_STATS_DETAIL=N

When set to Y, this parameters enables collection of the following event metrics from the agent:

True sample count

False sample count

True Sample ratio

False Sample ratio

Number of data rows counted in 24 hours

Number of true samples counted in 24 hours

Number of false samples counted in 24 hours

The agent keeps these metrics for eight days on disk, with roll-off daily at midnight. Default: **N**.

IRA_EVENT_EXPORT_SNMP_TRAP_CONFIG

By default, the agent looks to see if a `<install_dir>/localconfig/<pc>/<pc>_trapcnfg.xml` file exists. You can specify the complete path or the path relative to the local configuration directory.

z/OS For the SNMP trap configuration file that is a member of DDNAME RKANDATV, specify just the member name. For example, DDNAME RKANDATV member MYSNMP would be
IRA_EVENT_EXPORT_SNMP_TRAP_CONFIG=MYSNMP.

For the SNMP trap configuration file that is not a member in DDNAME RKANDATV, you need to add a DDNAME for the PDS. For example, DDNAME MYFILES is specified as 'TIVOLI.ITM622.TVT1006.MYFILES' and member MYSNMP would be
IRA_EVENT_EXPORT_SNMP_TRAP_CONFIG=MYSNMP.MYFILES.

IRA_LOCALCONFIG_DIR

The default local configuration directory path that contains locally customized configuration files such as threshold overrides, private situations, and SNNP trap configuration file is the `localconfig` subdirectory of the directory specified by the `CANDLE_HOME` environment variable; RKANDATV DD name on z/OS systems. Use this parameter to change the path.

Private situations

IRA_PRIVATE_SITUATION_CONFIG

Specifies the fully qualified private situation configuration file name. The default file on distributed systems resides in `<install_dir>/localconfig/<pc>/<pc>_SITUATIONS.xml` where `<pc>` is the two-character product code.

Windows `<install_dir>\TMAITM6`

Linux **UNIX** `<install_dir>/<platform>/<pc>/bin` where platform is the operating system (such as `li6263`).

z/OS A fully qualified path to the situation configuration file, such as 'TIVOLI.ITM622.TVT1006.RKANDATV(MYPSSIT)' where DDNAME RKANDATV is TIVOLI.ITM622.TVT1006.RKANDATV:
IRA_PRIVATE_SITUATION_CONFIG=MYPSSIT.

For a situation configuration file that is not a PDS member in DDNAME RKANDATV, specify 'TIVOLI.ITM622.TVT1006.MYFILES(MYPSSIT)' where DDNAME MYFILES is TIVOLI.ITM622.TVT1006.MYFILES:
IRA_PRIVATE_SITUATION_CONFIG=MYPSSIT.MYFILES.

Private history

IRA_PRIVATE_HISTORY_DIR

This is the default location for private history binary files:

Windows `<install_dir>\TMAITM6\logs`

Linux **UNIX** `<install_dir>/<arch>/<pc>/hist`

z/OS Dataset identified by PVTHIST DDNAME

Use this parameter to change the directory where private history binary files will be saved for the agent.

Agent Service Interface

These agent configuration parameters effect Service Interface operation:

IRA_SERVICE_INTERFACE_NAME

Specify the preferred agent service interface name that replaces the agent generated default name in the format of *product-codeagent*, such as kntagent or kmqagent. For example, specify uagent02 to identify the second installed Universal Agent instance on a system.

IRA_SERVICE_INTERFACE_DIR

Defines the path specification of the agent service interface HTML directory. In conjunction with the IRA_SERVICE_INTERFACE_DEFAULT_PAGE parameter, the agent constructs the file path to a specific, requested HTTP GET object. The default is *<itm_install_dir>/localconfig* on distributed systems.

Example: If IRA_SERVICE_INTERFACE_DIR="*\mypath\private*" and you enter *http://localhost:1920///kuxagent/kuxagent/html/myPage.htm* in your browser, *myPage.htm* is retrieved from *\mypath\private\html* instead of *<itm_install_dir>\localconfig\html*.

z/OS There is no directory path specification but instead a dataset represented by the JCL DD (Data Definition) name. Therefore, IRA_SERVICE_INTERFACE_DIR is not used but the IRA_SERVICE_INTERFACE_HTML specification is in effect. The default is RKANDATV DD name.

IRA_SERVICE_INTERFACE_DEFAULT_PAGE

Instructs the agent to open the named product-specific HTML page instead of the installed **navigator.htm** page upon log on to the agent service interface. The HTML file must exist in the agent installation HTML subdirectory: *<itm_install_dir>\localconfig\html* or as specified by IRA_SERVICE_INTERFACE_DIR.

IRA_SERVICE_INTERFACE_LOGON=N

Specify enabling (Y) or disabling (N) user system logon authentication. The default is N. On Linux and operating systems such as UNIX, enabling system logon requires the agent to run under root system ID authorization.

Diagnostics and troubleshooting

These parameters can be set for troubleshooting:

IRA_DEBUG_AUTONOMOUS=N

When set to Y, this parameter enables trace logging of all autonomous agent operation. The default setting is N.

IRA_DEBUG_EVENTEXPORT=N

When set to Y, this parameter enables trace logging of event export activity such a SNMP traps. The default setting is N.

IRA_DUMP_DATA=N

When set to Y, this parameter enables trace logging of all remote procedure call (RPC) data. The default setting is N.

IRA_DEBUG_PRIVATE_SITUATION=N

IRA_DEBUG_SERVICEAPI=Y

When set to Y, this parameter enables trace logging of all agent service interface processing. The default setting is N.

Related tasks

Editing the agent environment file

Situation limitations

Some formula functions are not available for private situations or do not evaluate when the agent is disconnected from the Tivoli Enterprise Monitoring Server.

Table 22. Situation formula functions available when an enterprise agent is connected or disconnected, or when the situation is private.









































































Formula function	Supported in enterprise situations	Connected to the monitoring server, evaluates at the agent and can emit alerts	Disconnected from the monitoring server, evaluates at the agent and can emit alerts	Supported in private situations
Cell functions				
CHANGE	 available	 available	 available	 not available
DATE	 available	 available	 available	 not available
MISSING	 available	 available	 available	 available
PCTCHANGE	 available	 available	 available	 not available
SCAN	 available	 available	 available	 not available
STR	 available	 available	 available	 not available
TIME	 available	 not available	 not available	 not available
VALUE	 available	 available	 available	 available
IN	 available	 available	 available	 not available
Group functions can be applied to multiple row attribute groups and to those configured for historical data collection. Table and chart views require that a time range be set to show a span of data samplings.				
AVG	 available	 not available	 not available	 not available
COUNT	 available	 not available	 not available	 not available
MAX	 available	 not available	 not available	 not available
MIN	 available	 not available	 not available	 not available
SUM	 available	 not available	 not available	 not available
Situation characteristics				
Embedded, including correlated situations	 available	 not available	 not available	 not available
Multiple attribute groups	 available	 not available	 not available	 not available
Persistence enabled	 available	 available ¹	 available ¹	 not available
Uses duper process	 available	 available	 not available ²	 not available

Table 22. Situation formula functions available when an enterprise agent is connected or disconnected, or when the situation is private. (continued)

Formula function	Supported in enterprise situations	Connected to the monitoring server, evaluates at the agent and can emit alerts	Disconnected from the monitoring server, evaluates at the agent and can emit alerts	Supported in private situations
¹ Situation persistence is not evaluated at the agent. Traps can be emitted in two modes: RC (Rising Continuous) whereby a trap is emitted every time the situation is true; HY (Hysteresis) whereby a trap is emitted the first time the situation is true and a clearing trap is emitted when the situation is no longer true. As well, persistence can be enabled at the trap destination by implementing a persistence rule. ² Traps are emitted but situations are not evaluated when the agent is disconnected from the monitoring server.				

Related reference

“Private situation operation” on page 131

Configuring Agent Management Services on Tivoli System Monitoring Agents

Configure the Agent Management Services for Tivoli System Monitoring Agents if you want to use the services to monitor and control agent availability.

Before you begin

Agent Management Services is configured differently in autonomous agent environments:

- Autonomous OS agents are managed by Agent Management Services by default. You suspend management by using the `disarmWatchdog` command, which disables the Agent Management Services watchdog for the autonomous OS agent and any agents created with Tivoli Monitoring Agent Builder on the same system. You resume management by the Agent Management Services by using the `rearmWatchdog` command, which enables the watchdog for the autonomous agents that are managed by the Agent Management Services. These commands are described in the agent user’s guide.
- Non-OS agents that are installed in an autonomous-only environment are not managed by the Agent Management Services watchdog by default. You can change whether the agent is managed by the watchdog.

About this task

After installing non-OS agents in an autonomous-only environment, take these steps to start or stop Agent Management Services management.

- While the watchdog process is running, move the common agent package (CAP) file named `k<pc>_default.xml` (where `<pc>` is the two-character product code) out of the CAP directory to a temporary location. The file is located in the `KCA_CAP_DIR` directory.

Windows `<itm_install_dir>\TMAITM6\CAP\`

Linux **UNIX** `<itm_install_dir>/config/CAP`

Removing the file from the CAP directory renders the agent invisible to the Agent Management Services.

- Modify all instances of `<managerType>` in the CAP file to enable or disable management:

- `<managerType>ProxyAgentServices</managerType>` to enable management.
- `<managerType>NotManaged</managerType>` to disable management.

A best practice is to rename the modified file to **k<pc>.xml** (where *<pc>* is the two-character product code). All CAP files located in the KCA_CAP_DIR are processed by Agent Management Services. If two or more CAP files share the same “subagent id” value, they are processed in sorted order. For example, kca.xml is used before kca_default.xml. Also, renaming the CAP file to k<pc>.xml ensures that your changes do not get overwritten during a future upgrade.

3. Save the updated file.
4. While the watchdog process (kcawd) is running, move or copy the updated CAP file back to KCA_CAP_DIR.

Results

The updated Agent Management Services settings are processed after the CAP file is placed in KCA_CAP_DIR.

Private situations

Define private situations for monitoring criteria and the resulting events that are pertinent to your local agent environment or to an event receiver and not relevant to the Tivoli Enterprise Monitoring environment. Private situations can be defined for Tivoli Enterprise Monitoring Agentautonomous agents and Tivoli System Monitoring Agents.

Private situation operation

Private situations are created in an XML formatted file that does not interact with the Tivoli Enterprise Monitoring Server. To use private situations effectively, you need to understand how they are different from enterprise situations.

Tivoli Management Services agent framework

Built into the agent framework of the Tivoli Management Services infrastructure is the ability to create situations that run locally and trigger events on the computer where you have either a Tivoli Enterprise Monitoring Agent or Tivoli System Monitoring Agent installed.

Enterprise situations and private situations

Enterprise situations are created with the Tivoli Enterprise Portal Situation editor or with the CLI **tacmd createSit** command. Enterprise situations send events to the monitoring server and can forward events to an Event Integration Facility receiver such as a Tivoli Enterprise Console event server or Netcool/OMNIBus Probe for Tivoli EIF when the hub monitoring server has been configured to forward events..



Private situations are created in a local private situation configuration XML file for the agent. Situation definitions that were exported from the monitored enterprise can also be added to the file to create situations. The events generated by private situations can remain local to your workstation or be sent as SNMP alerts to a receiver such as the Netcool/OMNIBus SNMP Probe. The private situation configuration file resides in the agent installation directory, one file per agent, and it contains all the private situation definitions for the agent.

Creating private situations

This example of a private situation configuration XML file for the Windows OS agent has two situations defined. You can create situations in the file by entering them manually. You can also create situations in this file by exporting existing enterprise situations from the monitoring server, using the CLI **tacmd bulkExportSit** and then copying the exported situations from their XML file to the agent's Private Situation configuration file.

```
<PRIVATECONFIGURATION>
  <PRIVATESIT>
    <SITUATION>NT_Missing_Scheduler_pr</SITUATION>
    <CRITERIA>
      <![CDATA[ *MISSING NT_Process.Process_Name *EQ ('schedule')]]>
    </CRITERIA>
    <INTERVAL>001000</INTERVAL>
  </PRIVATESIT>
  <PRIVATESIT>
    <SITUATION>NT_Paging_File_Critical_pr</SITUATION>
    <CRITERIA>
      <![CDATA[ *VALUE NT_Paging_File.%_Usage *GE 80 ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
  </PRIVATESIT>
</PRIVATECONFIGURATION>
```

The CRITERIA element contains the formula:

- ***VALUE** or ***MISSING** function name.  **Value of expression** and  **Check for Missing Items** are the only formula functions available for use in private situations.
- **attribute_group.attribute_name** as they are written:
 - in the **name** element of the agent .atr file, located in the `<install_dir>/TMAITM6/ATTRLIB` directory,
 - in the **<PDT>** element of the `<situation_name>.xml` file, which is the output of the `tacmd bulkExportSit` CLI command, or
 - in the **<PREDICATE>** element of the Situation Summary report that is generated through the Agent Service Interface.
- ***EQ**, ***LT**, ***GT**, ***LE**, or ***GE** Boolean operator.
- **Threshold** for the ***VALUE** function or comma-separated list for the ***MISSING** function.
- Multiple expressions can be connected by Boolean AND or OR logic, but not both. Up to nine expressions connected by AND are supported; and up to ten expressions connected by OR are supported.

Activation

When the agent is initialized, an XML parser examines and validates the private situation definitions. All XML parsing error messages are recorded in the agent operations log. (See the *IBM Tivoli Monitoring Troubleshooting Guide*.)

Private situations continue to run until the agent is shut down.

The events that are opened when a situation becomes true can be sent as SNMPv1v2 traps or SNMPv3 informs when an SNMP trap configuration file is created and a receiver such as the Netcool/OMNIbus SNMP Probe has been configured to receive them. As well, the Agent Service Interface provides a summary report of situation activity.

You create a private situation file named `<pc>_situations.xml` and save it to the `<install_dir>/localconfig/<pc>` (where `<pc>` is the product code). If you prefer to name the file differently or use a different path, agent environment parameters are provided for you to change the file name and path.

Summary

Private situations are agent monitoring requests defined by a local administrator with criteria that is pertinent to the local agent environment. This is a summary of private situation characteristics:

- Created at the agent locally through a simple editor.
- Emit results and events with agent SNMP traps.
- Have a separate name space to avoid naming conflicts with enterprise situations.
- Run from the time the agent starts until it stops regardless of monitoring server connectivity.
- Multiple expressions in a formula must have logic connectors that are uniformly conjunctive AND or disjunctive OR; a mix of the two connectors in a formula is not supported.
- Support up to nine expressions in the situation formula when connected by Boolean AND logic and up to ten expressions when connected by Boolean OR logic.
- All enterprise situation threshold operators are supported: equal (EQ), not equal (NE), greater than (GT), less than (LT), greater than or equal (GE), and less than or equal (LE).
- Support the reflex automation action command.
- Support for the VALUE and MISSING formula functions only; no support for group functions or other cell functions.
- No support for wildcards.
- Run concurrently with enterprise situations when the agent is in connected to the monitoring server.
- Can run on a Tivoli Enterprise Monitoring Agent, whether connected or autonomous, or Tivoli System Monitoring Agent.
- Remain unknown to the IBM Tivoli Monitoring centrally managed infrastructure. Tivoli Management Services is unaware of their existence, including their monitoring data and events. Therefore, private situations do not participate in event caching or persistence across agent restarts while the agent is disconnected from its monitoring server.

Related reference

“Situation limitations” on page 129

Private situation XML specification

Use the elements from the private situation XML specification to create private situations for an agent on your computer.

Elements

<PRIVATECONFIGURATION>

PRIVATECONFIGURATION is the root element identifying this as an agent private situation configuration document.

<PRIVATECONFIGURATION>

<PRIVATESIT>

<SITUATION NAME="Check_Process_CPU_Usage" INTERVAL="000500" /><SITUATION>

<CRITERIA>

```

<![CDATA[ *VALUE NT_Process.% Processor_Time *GE 65 *AND
*VALUE NT_Process.Priority_Base *NE 0 *AND
*VALUE NT_Process.Process_Name *NE _Total]]>
</CRITERIA>
<CMD><![CDATA[netstat >.\logs\netstat.dat]]></CMD>
<AUTOSOFT When="N" Frequency="N" />
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

<PRIVATESIT>

Enclose each situation definition in PRIVATESIT begin and end tags.

<SITUATION>

Within each set of PRIVATESIT begin and end tags, add a set of SITUATION begin and end tags. Within each set of SITUATION begin and end tags is the complete situation definition. Define the situation with these attributes:

NAME= The situation name, which must begin with a letter and can be up to 31 letters, numbers and _ underscores, such as "Missing_Process_Helper_Harmless".

INTERVAL= Specify the sampling interval in HHMMSS format. Default: **001500** (15 minutes). Alternatively, use the <INTERVAL> element.

CRITERIA= The situation formula. Alternatively, use the <CRITERIA> element.

```

<SITUATION NAME="High_CPU_Usage" INTERVAL="000500"
CRITERIA="*VALUE NT_Process.% Processor_Time *GE 65
*AND *VALUE NT_Process.Priority_Base *NE 0
*AND *VALUE NT_Process.Process_Name *NE _Total" /SITUATION>

```

<INTERVAL>

Specifies the situation sample interval in HHMMSS format. A value of 000000 (six zeroes) indicates a pure-event situation. The minimum interval is 000030 (30 seconds); the maximum 235959 (23 hours, 59 minutes, and 59 seconds). Default: **001500** (15 minutes). This element is required if INTERVAL is not specified in the SITUATION element.


<CRITERIA>

The situation criteria is specified within this element and the <![CDATA[]]> element. Each expression has three parts, starting with *VALUE or *MISSING, followed by **attribute-table-name.attribute-name**, the logical operator (such as *EQ), and the attribute threshold value or, for the MISSING function, a comma-separated list of names.

For the attribute, use the detailed attribute name in the format of attribute-table- name dot attribute-name. The product attribute file defines the agent product attribute tables and associated attributes, for example, **knt.atr** or **kux.atr** files residing in the ATTRLIB directory for a distributed agent installation.

The Operator defines logical operation of filter value and data. The supported operators are: *EQ for equal, *NE for not equal, *GE for greater than or equal to, *LE for less than or equal to, *LT for less than, and *GT for greater than.

For multiple expressions, use the *AND or *OR connector. All connectors in the formula must be the same, either all *AND or all *OR. Mixing logical *AND and *OR connectors is not supported. You can have up to 9 *AND connectors or up to 10 *OR connectors in a formula.

In a formula with multiple expressions, there can be no more than one *MISSING expression, it must be the last expression in the formula, and only *AND connectors can be used. (See the *Tivoli Enterprise Portal User's Guide* for a description of  **Check for Missing Items**.)

Wildcards are not supported. For example, *VALUE NT_Process.Process_Name *EQ S* to find all processes that start with "S" is invalid in a private situation. Likewise, wildcards in a *MISSING list are invalid, such as NT_Process.Process_Name *EQ ('DB2*') to find all processes beginning with DB2.

Examples:

```
<CRITERIA>
<![CDATA[ *VALUE NT_Process.%_Processor_Time *GE 65 *AND
          *VALUE NT_Process.Priority_Base *NE 0 *AND
          *VALUE NT_Process.Process_Name *NE _Total]]>
</CRITERIA>

<CRITERIA>
<![CDATA[ *MISSING NT_Process.Process_Name *EQ ('schedule','notepad')]]>
</CRITERIA>

<CRITERIA>
<![CDATA[ *VALUE Linux_Process.State *NE Running *AND
          *MISSING Linux_Process.Process_Command_Name *EQ ('MyHelp','myhelpw')]]>
</CRITERIA>
```

Enumerated attributes have a predefined set of values. You can specify either the enumeration symbol or the name. For example, both of these expressions with a process execution state of Stopped (T) are valid. If an SNMP alert is sent or an action taken, the symbol is used rather than the name:

```
<CRITERIA><![CDATA[ *IF *VALUE Process.Execution_State *EQ Stopped]]></CRITERIA>
<CRITERIA><![CDATA[ *IF *VALUE Process.Execution_State *EQ T]]></CRITERIA>
```

If the private situation uses any scaled attributes, their values must be normalized for proper evaluation. A scaled attribute value is used to specify how many positions to shift the decimal point to the left. For example, 55.255 is a valid value for an attribute that displays with a scale of 3. To normalize it, you would shift the decimal point right by three places to be 55255.

This example shows a hexadecimal integer as the comparison value:

```
<CRITERIA><![CDATA[ *IF *VALUE Disk.Mount_Point_U *EQ '/opt' *AND
          *VALUE Disk.Space_Used_64 *GT 0x80000000 ]]]></CRITERIA>
```

The <CRITERIA> element is required if CRITERIA is not specified in the <SITUATION> element.

<CMD>

Optional. Defines the action command or script to invoke when the situation is true. Enclose the command in the <![CDATA[]]> section.

Example:

```
<CMD><![CDATA[netstat >.\logs\netstat.dat]]></CMD>
```

<AUTOSOFT>

This is required if an action <CMD> is specified. It defines the action command execution options, WHEN (X), FREQUENCY (Y), WHERE (Z). The default is NNN:

WHEN= Optional. "Y" to run the command for each item; or "N" to run the command on the true item only. If the attribute group returns

multiple rows of data and more than one row satisfies the condition, you can choose to issue the command on only the first row that meets the criteria or once for each row that meets the criteria. Default: "N".

FREQUENCY= Optional. "Y" to issue the command on every true situation evaluation, even if no false states intervene; or "N" if you want the command to run only once and not every time incoming data matches the condition. If the situation remains true over successive intervals, the command runs only the first time the situation is true and not again until it becomes true from being false in the previous interval. Default: "N".

WHERE= "N" to run the command at the agent. Default: "N" Because there is only one possible setting for "where", you do not need to include it in the AUTOSOFT element.

```
<AUTOSOFT When="Y" Frequency="Y" />
```

<DISTRIBUTION>

Required for products with subnodes (subagents). Specifies a managed system name or a list of managed system names separated by a ; semicolon. The default is the agent managed system name or all known subagents.

<HISTORY>

Optional. Use the history element to specify each attribute group that you want to collect historical data for. It must be above or below the <PRIVATESIT> definitions. The agent does not support multiple <HISTORY> specifications for the same TABLE. The XML parser processes duplicated <HISTORY> as update scenarios. The final updated attribute value will be the value in effect and always output to agent's Operation Log.

TABLE= This parameter specifies the application attribute group name. If the agent cannot convert it to the SQL table name then the name is assumed to be the table name.

INTERVAL= Optional. This parameter specifies the historical data collection interval in minutes. The minimum collection interval is 1 minute and the maximum is 1440 (24 hours). Valid intervals are values that divide evenly into 60 or are divisible by 60: an interval below 60 could be 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, and 30; an interval greater than 60 could be 120, 180, 240, and so on, up to 1440. If you enter an invalid value, no history will be collected for the specified attribute group. Default: "15".

RETAIN= Optional. Retain defines the short-term history data retention period in hours. The default is 24 hours and the minimum retention period is one hour. There is no limit other than that imposed by storage space on the computer. After the retention limit has been reached, the oldest data samples are deleted as new samples arrive. Default: "24".

Examples:

The agent collects WTSYSTEM table data every 15 minutes and maintains 96 data rows (four times per hour for 24 hours) in the history file.

```
<HISTORY TABLE="NT_System" />
```

The agent collects UNIX OS table data every 5 minutes and maintains 3 days of short-term history.

```
<HISTORY TABLE="UNIXOS" Interval="5" RETAIN="72" />
```

The agent collects WTLOGCLDSK table data every minute. The history file keeps 24 hours history for a total of 1440 table row records.

```
<HISTORY TABLE="NT_Logical_Disk" INTERVAL="1" />
```

Related reference

“Private history” on page 147

“Agent Service Interface - History” on page 172

“Agent Service Interface - Situations” on page 170

Exported enterprise situation XML specification

Use the situation definitions from the <situation_name>.xml files that result from the CLI `tacmd bulkExportSit` and `tacmd viewSit` commands to populate the agent’s private situation configuration file.

If you already have enterprise situations for a Tivoli Enterprise Monitoring Agent, you can run the bulk export situation command or the view situation command to get situation definitions for the specified agent in the XML format that is acceptable to the private situation configuration file. Not all exported situations are valid; only those that use the *VALUE or *MISSING formula functions. See “Situation limitations” on page 129 for other restrictions.

Elements

<TABLE>

This is the root element that identifies this as a private situation configuration file.

<ROW>

This is the child element to follow TABLE.

<SITNAME>

Monitoring situation name. The situation name must begin with a letter and can be up to 31 letters and numbers and _ underscores. Example:

```
<SITNAME>Free_DiskSpace_Low</SITNAME>
```

<PDT>

The situation criteria is specified within the <PDT> element and the <![CDATA[]]> element. Each expression has three parts, starting with *VALUE or *MISSING, followed by **attribute-table-name.attribute-name**, the logical operator (such as EQ), and the attribute threshold value (for the MISSING function, this is a comma-separated list).

For the attribute, use the detailed attribute name in the format of attribute-table- name dot attribute-name. The product attribute file defines the agent product attribute tables and associated attributes, for example, KNT.atr or KUX.atr files residing in the ATTRIB directory for a distributed agent installation.

The Operator defines logical operation of filter value and data. The supported operators are: *EQ for equal, *NE for not equal, *GE for greater than or equal to, *LE for less than or equal to, *LT for less than, and *GT for greater than. Within the <PDT> element, the command is enclosed in Character Data tags to exclude it from XML parsing. This example shows a formula that triggers an alert when the available disk space is 35% or below:

```
<PDT> <![CDATA[*IF *VALUE NT_Logical_Disk.%_Free *LE 35]]> </PDT>
```

For multiple expressions, use the *AND and *OR connectors. All connectors in the formula must be the same, either all *AND or all *OR. A mix of logical *AND and *OR connectors is not supported. Example:

```
<PDT> <![CDATA[*VALUE NT_Process.%_Processor_Time *GE 65 *AND
*VALUE NT_Process.Priority_Base *NE 0 *AND
*VALUE NT_Process.Process_Name *NE _Total]]> </PDT>
```

Wildcards are not supported in private situations. For example, *VALUE NT_Process.Process_Name *EQ DB2* to find all processes that start with “DB2” is invalid. Exported enterprise situations with scaled attributes are not normalized when running as private situation. You must normalize the values manually. For example, this enterprise situation expression Avg Disk Queue Lenth is >= 0.004 is for a floating point attribute with a scale of 3. When the situation is exported with tacmd viewSit, the export monitoring criteria is shown as:

```
<PDT> <![CDATA[*IF *VALUE NT_Physical_Disk.Avg_Disk_Queue_Length
*GE 0.004]]> < PDT>
```

When this same definition is specified in a private situation, the value comparison value is being interpreted as a zero value.

```
<PRIVATECONFIGURATION>
<PRIVATESIT>
<SITUATION>SCALE_TEST</SITUATION>
<CRITERIA><![CDATA[ *IF *VALUE NT_Physical_Disk.Avg_Disk_Queue_Length
*GE 0.004 ]]></CRITERIA>
<INTERVAL>000030</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>
```

Normalize the value by shifting the decimal point to the right by three places: 0.004 would be 4 or a value such as shown here.

```
<CRITERIA><![CDATA[ *IF *VALUE NT_Physical_Disk.Avg_Disk_Queue_Length
*GE 4.123 ]]></CRITERIA>
```

SCAL (Scale)	Integer comparison value (example used is 5000)
Not defined (0)	5000
1	seen as 500 or 500.0 but represents 5000
2	seen as 50 or 50.00 but represents 5000
3	seen as 5 or 5.000 but represents 5000

The attribute description topics for your product should specify whether the value is scaled. For distributed agents, you can also review the attribute file for scal in the attribute definition. For example, khd.atr for the Warehouse Proxy agent has a work queue insertion rate attribute with scal 2. The k<pc>.atr files are located here:

Windows <install_dir>\TMAITM6\ATTRLIB
Linux **UNIX** <install_dir>/platform/<pc>/tables/ATTRLIB, where platform is the operating system and <pc> is the product code.

<CMD>

Optional. Defines the action command or script to invoke when the situation criteria are true. Within the <CMD> element, the command is enclosed in Character Data tags to exclude it from parsing. This example shows a system command that displays the timestamp in a message box at the agent when the situation becomes true. Without the CDATA tagging, the & ampersand and {} brackets would be considered an error by the XML parser.

```

<CMD>
<![CDATA[ net send &{Local_Time.Timestamp}  ]]>
</CMD>

```

tags.

<AUTOSOFT>

This is required if an action <CMD> is specified. It defines reflex automation action command execution options, in order XYZ, between begin and end tags. The default is NNN:

- **Only take action on first item**
- **Don't take action twice in a row (wait until situation goes false then true again)**
- **Execute the Action at the Managed System (Agent)**

X=Y Run command for each item.

X=N Run command on first item only.

Y=Y Run command for each sample interval.

Y=N Do not run command twice in a row.

Z=N This is always set to N for private situations, and means to run the command at the agent. If the exported option is set to Y, the setting will be ignored and be treated as N.

<DISTRIBUTION>

Required for products with subnodes (subagents). Specifies a managed system name or a list of managed system names separated by a ; semicolon. The default is the agent managed system name or all known subagents.

<LSTCCSID>

Optional. Specifies the IBM Code Character Set ID. **en_US** is the only value allowed.

<LSTDATE>

Optional. Situation last updated timestamp. If it is unspecified then the current data time is automatically generated. The format is CYYMMDDHHMMSSmmm (as in 1090715074501000 for July 15, 2009 at 07:45:01) where:

C = Century (1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

<LSTRELEASE>

Optional. Specifies the situation version.

<LSTUSRPRF>

Optional. This is the ID of the user who last updated this situation definition. If it is unspecified then the current logon user ID is used.

<SITINFO>

Optional. Defines the situation qualifiers. Within the <SITINFO> element,

enclose the situation formula in `<![CDATA[]]>` tagging. Alternatively, this defines qualifiers using parameters. Multiple qualifiers are delimited by a ; semicolon.

ATOM= Optional. EIF forwarding.

COUNT= Optional. EIF forwarding. The default value is 1.

SEV= Optional. EIF forwarding.

TFWD= Optional. EIF forwarding.

TDST Optional. EIF forwarding.

<TEXT>

Situation description. Within the `<TEXT>` element, enclose the situation formula in `<![CDATA[]]>` tagging.

<REEV_TIME>

Specifies the situation sample interval in HHMMSS format. A value of 0 zero indicates a pure-event situation. The default interval is 15 minutes, 001500; the minimum is 30 seconds, 000030; and the maximum is 23 hours, 59 minutes, and 59 seconds, 235959. Example:

```
<REEV_TIME>000500</REEV_TIME>
```

Ignored elements

These elements in the exported XML specification are not used:

```
<FULLNAME>
<ADVISE>
<AFFINITIES>
<ALERTLIST>
<AUTOSTART>
<DESTNODE />
<HUB />
<LOCFLAG />
<NOTIFYARGS>
<NOTIFYOPTS>
<OBJECTLOCK>
<PRNAMES>
<QIBSCOPE>
<REEV_DAYS>
<REFLEXOK>
<SENDMSGQ>
<SOURCE>
```

Exported enterprise situation example

The `NT_System_File_Critical` situation exported with **tacmd bulkExportSit** or **tacmd viewSit** is saved in the file, `NT_System_File_Critical.xml`:

```
<TABLE>
<ROW>
  <SITNAME>NT_System_File_Critical</SITNAME>
  <FULLNAME>
    <![CDATA[ ]]>
  </FULLNAME>
  <ADVISE>
    <![CDATA[ ADVISE("knt:"+$ISITSTSH.SITNAME$);]]>
  </ADVISE>
```

```

<AFFINITIES>%IBM.STATIC021 01000000000</AFFINITIES>
<ALERTLIST>*NO</ALERTLIST>
<AUTOSOFT>NNN</AUTOSOFT>
<AUTOSTART>*YES</AUTOSTART>
<CMD>
  <![CDATA[ *NONE  ]]>
</CMD>
<DESTNODE />
<HUB />
<LOCFLAG />
<LSTCCSID />
<LSTDATE>0961009010101000</LSTDATE>
<LSTRELEASE />
<LSTUSRPRF>IBM</LSTUSRPRF>
<NOTIFYARGS />
<NOTIFYOPTS />
<OBJECTLOCK />
<PDT>
  <![CDATA[ *IF *VALUE NT_System.File_Data_Operations/Sec *GE 100000 ]]>
</PDT>
<PRNAMES />
<QIBSCOPE>E</QIBSCOPE>
<REEV_DAYS>0</REEV_DAYS>
<REEV_TIME>001500</REEV_TIME>
<REFLEXOK />
<SENDMSGQ>*NONE</SENDMSGQ>
<SITINFO>
  <![CDATA[ SEV=Critical ]]>
</SITINFO>
<SOURCE />
<TEXT>
  <![CDATA[ Knt:KNT1359 ]]>
</TEXT>
<DISTRIBUTION>*NT_SYSTEM</DISTRIBUTION>
</ROW>
</TABLE>

```

In the private situation configuration file, a set of <PRIVATESIT> and </PRIVATESIT> tags are created, then the contents of NT_System_File_Critical.xml pasted inside the tags. This is an nt_situations.xml private situation configuration file after the exported NT_System_File_Critical situation definition was added above another private situation definition, Check_Process_CPU_Usage. The redundant elements (see “Ignored elements” earlier) and unused elements (AUTOSOFT and CMD, LSTCCSID, LSTRELEASE, DISTRIBUTION) from the exported situation were removed, although leaving them in the file does no harm because the XML parser ignores them:

```

<PRIVATECONFIGURATION>
<PRIVATESIT>
<TABLE>
<ROW>
  <SITNAME>NT_System_File_Critical</SITNAME>
  <LSTDATE>0961009010101000</LSTDATE>
  <LSTUSRPRF>IBM</LSTUSRPRF>
  <PDT>
    <![CDATA[ *IF *VALUE NT_System.File_Data_Operations/Sec *GE 100000 ]]>
  </PDT>
  <REEV_TIME>001500</REEV_TIME>
  <SITINFO>
    <![CDATA[ SEV=Critical ]]>
  </SITINFO>
  <TEXT>
    <![CDATA[ Knt:KNT1359 ]]>
  </TEXT>
</ROW>
</TABLE>

```

```

</PRIVATESIT>
<PRIVATESIT>
  <SITNAME>Check_Process_CPU_Usage</SITNAME>
  <PDT>
    <![CDATA[ *VALUE NT_Process.%_Processor_Time *GE 65 *AND
      *VALUE NT_Process.Priority_Base *NE 0 *AND
      *VALUE NT_Process.Process_Name *NE _Total]]>
    </PDT>
  <REEV_TIME>000300</REEV_TIME>
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

Private situation examples

Define private situations for monitoring criteria that is pertinent to your local agent environment and not dependent on or relevant to the enterprise environment. These examples can be used as a template for your private situations.

Linux OS lz_situations.xml

```

<PRIVATECONFIGURATION>
<!-- Situation Description: Percentage of time the processor is busy
is extremely high -->
<PRIVATESIT>
  <SITUATION>Linux_High_CPU_Overload_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_CPU.Idle_CPU *LT 10 *AND *VALUE Linux_CPU.CPU_ID
      *EQ Aggregate ]]>
    </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of packet collisions during data
transmission is high -->
<PRIVATESIT>
  <SITUATION>Linux_High_Packet_Collisions_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_Network.Collision_Percent *GT 10 ]]>
    </CRITERIA>
  <INTERVAL>000500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of available i-nodes is low -->
<PRIVATESIT>
  <SITUATION>Linux_Low_Pct_Inodes_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_Disk.Inodes_Used_Percent *GT 80 ]]>
    </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of space available on a filesystem
is low -->
<PRIVATESIT>
  <SITUATION>Linux_Low_Pct_Space_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_Disk.Space_Available_Percent *LT 15 ]]>
    </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Tests if the SSH Daemon, sshd, is up running -->
<PRIVATESIT>
  <SITUATION>Linux_Process_Missing_sshd_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *IF *MISSING Linux_Process.Process_Command_Name
      *EQ ('/usr/sbin/sshd') ]]>
    </CRITERIA>
  <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of Processor time used by

```

```

a process high -->
<PRIVATESIT>
  <SITUATION>Linux_Process_High_CPU_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_Process.Busy_CPU_Pct *GT 60 ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: High number of stopped processes on this system -->
<PRIVATESIT>
  <SITUATION>Linux_Process_Stopped_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_Process.State *NE Running *AND
*VALUE Linux_Process.State *NE Sleeping *AND
*VALUE Linux_Process.State *NE Disk *AND
*VALUE Linux_Process.State *NE Trace ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of rejected RPC server or
client calls is high -->
<PRIVATESIT>
  <SITUATION>Linux_RPC_Bad_Calls_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_RPC_Statistics.RPC_Client_Calls_Retransmitted *GT 30
*OR *VALUE Linux_RPC_Statistics.RPC_Server_Calls_Rejected *GT 30 ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: The swap space paging activity on this system
is extremely high -->
<PRIVATESIT>
  <SITUATION>Linux_System_Thrashing_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE Linux_System_Statistics.Pages_paged_out_per_sec *GT 400
*OR *VALUE Linux_System_Statistics.Pages_paged_in_per_sec *GT 400 ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

UNIX OS ux_situations.xml

```

PRIVATECONFIGURATION>
<!-- Situation Description: Reports High CPU processes -->
<PRIVATESIT>
  <SITUATION>UNIX_CMD_Runaway_Process_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *IF *VALUE Process.CPU_Utilization *GT 95 ]]>
  </CRITERIA>
  <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Process CPU utilization is greater than
or equal to 85% -->
<PRIVATESIT>
  <SITUATION>UNIX_CPU_Critical_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *IF *VALUE Process.CPU_Utilization *GE 85 *AND *VALUE
Process.Command *NE kproc *AND *VALUE Process.Command *NE swapper ]]>
  </CRITERIA>
  <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Notes typical I/O bound processor (NFS) -->
<PRIVATESIT>
  <SITUATION>UNIX_HD_Exces_IO_Wait_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE System.Wait_I/O *GT 20 ]]>

```

```

    </CRITERIA>
    <INTERVAL>000200</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Tests if the Internet Services Daemon, inetd,
is up running -->
<PRIVATESIT>
    <SITUATION>UNIX_Process_Missing_inetd_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *MISSING Process.Command *EQ ('/usr/sbin/inetd') ]]>
    </CRITERIA>
    <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Checks the System CPU, Idle, I/O Wait,
and Load Averages for the Busy state -->
<PRIVATESIT>
    <SITUATION>UNIX_System_Busy_Warning_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE System.System_CPU *GT 50 *AND
*VALUE System.Idle_CPU *GT 0 *AND *VALUE System.Wait_I/O *GT 0 *AND
*VALUE System.Load_Average_5_Min *GT 1 ]]>
    </CRITERIA>
    <INTERVAL>000200</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

Windows OS nt_situations.xml

```

PRIVATECONFIGURATION>
<!-- Situation Description: One of the NT Logs is close to capacity -->
<PRIVATESIT>
    <SITUATION>NT_Log_Space_Low_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Monitored_Logs_Report.%_Usage *GE 95 ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Test if the NT Scheduler process is running -->
<PRIVATESIT>
    <SITUATION>NT_Missing_Scheduler_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *MISSING NT_Process.Process_Name *EQ ('schedule') ]]>
    </CRITERIA>
    <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percent of the Page File in use is too high -->
<PRIVATESIT>
    <SITUATION>NT_Paging_File_Critical_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Paging_File.%_Usage *GE 80 ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percent of the Page File in use is rising -->
<PRIVATESIT>
    <SITUATION>NT_Paging_File_Warning_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Paging_File.%_Usage *GE 75 *AND
*VALUE NT_Paging_File.%_Usage *LT 80 ]]>
    </CRITERIA>
    <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percent of the time the disk drive is busy
is too high -->
<PRIVATESIT>
    <SITUATION>NT_Phys_Disk_Busy_Crit_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Physical_Disk.%_Disk_Time *GT 90 *AND

```

```

        *VALUE NT_Physical_Disk.Disk_Name *NE _Total ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percent of the time the disk drive is busy
is rising -->
<PRIVATESIT>
    <SITUATION>NT_Phys_Disk_Busy_Warn_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Physical_Disk.%_Disk_Time *GT 80 *AND
*VALUE NT_Physical_Disk.%_Disk_Time *LE 90 *AND
*VALUE NT_Physical_Disk.Disk_Name *NE _Total ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of processor time used is too high -->
<PRIVATESIT>
    <SITUATION>NT_Proc_CPU_Critical_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Process.%_Processor_Time *GE 65 *AND *VALUE
NT_Process.Priority_Base *NE 0 *AND *VALUE NT_Process.Process_Name
*NE _Total ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Percentage of processor time used is high -->
<PRIVATESIT>
    <SITUATION>NT_Proc_CPU_Warn_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Process.%_Processor_Time *GE 50 *AND
*VALUE NT_Process.%_Processor_Time *LT 65 *AND
*VALUE NT_Process.Priority_Base *NE 0 *AND
*VALUE NT_Process.Process_Name *NE _Total ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: A Service Error was reported -->
<PRIVATESIT>
    <SITUATION>NT_Service_Error_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_Event_Log.Source *EQ 'Service Control Manager'
*AND *VALUE NT_Event_Log.Type *EQ Error ]]>
    </CRITERIA>
    <INTERVAL>001000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Rate of operations to file system devices
per second is too high -->
<PRIVATESIT>
    <SITUATION>NT_System_File_Critical_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_System.File_Data_Operations/Sec *GE 100000 ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Rate of operations to file system devices per second
is rising -->
<PRIVATESIT>
    <SITUATION>NT_System_File_Warn_pr</SITUATION>
    <CRITERIA>
        <![CDATA[ *VALUE NT_System.File_Data_Operations/Sec *GE 10000 *AND
*VALUE NT_System.File_Data_Operations/Sec *LT 100000 ]]>
    </CRITERIA>
    <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

Tivoli Data Warehouse Summarization and Pruning sy_situations.xml

```
PRIVATECONFIGURATION>
<!-- Situation Description: No connectivity to Warehouse database -->
<PRIVATESIT>
  <SITUATION>KSY_DB_Connectivity_Fail_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KSY_CONNECTIVITY.DB_Connectivity *EQ No ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Failures occurred in pruning -->
<PRIVATESIT>
  <SITUATION>KSY_Pruning_Failures_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KSY_SUMMARIZATION_STATISTICS.Pruning_Failures *GT 0 ]]>
  </CRITERIA>
  <INTERVAL>000000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Failures occurred in summarization -->
<PRIVATESIT>
  <SITUATION>KSY_Summ_Failures_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KSY_SUMMARIZATION_STATISTICS.Summarization_Failures
      *GT 0 ]]>
  </CRITERIA>
  <INTERVAL>000000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: No connectivity to the
Tivoli Enterprise Portal Server -->
<PRIVATESIT>
  <SITUATION>KSY_TEPS_Conn_Fail_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KSY_CONNECTIVITY.TEPS_Connectivity *EQ No ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>
```

Tivoli Data Warehouse warehouse_situations.xml

```
PRIVATECONFIGURATION>
<!-- Situation Description: No connectivity to warehouse database -->
<PRIVATESIT>
  <SITUATION>KHD_DB_Connectivity_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KHD_DB_INFO.DB_Connectivity *EQ No ]]>
  </CRITERIA>
  <INTERVAL>001500</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Critical errors during the execution
of the Warehouse Proxy -->
<PRIVATESIT>
  <SITUATION>KHD_Error_Critical_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KHD_LAST_ERROR_DETAILS.Error_Severity *EQ Critical ]]>
  </CRITERIA>
  <INTERVAL>000000</INTERVAL>
</PRIVATESIT>
<!-- Situation Description: Fatal errors during the execution
of the Warehouse Proxy -->
<PRIVATESIT>
  <SITUATION>KHD_Error_Fatal_pr</SITUATION>
  <CRITERIA>
    <![CDATA[ *VALUE KHD_LAST_ERROR_DETAILS.Error_Severity *EQ Fatal ]]>
```

```

</CRITERIA>
<INTERVAL>000000</INTERVAL>
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

Private history

Private history is the collection and short-term storage of data from a local monitoring agent. Define historical collection in a private situation configuration file for an agent, then use the Agent Service Interface to view the short-term history.

Local historical data collection is defined in the local private situation configuration file for each attribute group that you want to save historical data for. You can define private history with or without private situations. There can be only one active history data collection per application table (attribute group).

All XML validation error messages are saved to the Agent Operation Log. The private history is completely separate and independent of historical data collection and the Tivoli Data Warehouse configuration within IBM Tivoli Management Services. Each private short-term history table data resides in its own history binary file.

The table name for an attribute group is also the history binary file name, with one unique history binary file per table. As part of the private history configuration, you can set the RETAIN attribute to manage the history file size. The agent outputs all private history files to this subdirectory:

Windows	<code><install_dir>\TMAITM6\logs</code>
Linux	<code><install_dir>/<arch>/<pc>/hist</code>
z/OS	Dataset identified by PVTHIST DDNAME

You can configure an alternative private history file location with the agent configuration parameter IRA_PRIVATE_HISTORY_DIR.

Related reference

“Private situation XML specification” on page 133

“Agent Service Interface - History” on page 172

SNMP alerts

Tivoli Enterprise Monitoring Agents can be configured to send Simple Network Management Protocol alerts to event receivers such as the Netcool/OMNIbus SNMP Probe.

SNMP alert configuration

Configure a Tivoli Enterprise Monitoring Agent to emit life cycle events or situation events to SNMP event receivers like Netcool/OMNIbus using the Netcool/OMNIbus SNMP Probe or IBM Tivoli NetView. Sample OMNIbus rules files are provided for immediate use with Netcool/OMNIbus.

Trap configuration file

You can configure situations in a trapcnfg.xml file that will emit SNMPv1/v2 traps or SNMPv3 informs to a receiver. When the agent is started, it will emit the traps that are defined in the file. The file is named `<pc>_trapcnfg.xml`, where `<pc>` is the 2-character product code of the agent and resides in the `<itm_install_dir>/localconfig/<pc>` directory.

Agent parameters

IRA_EVENT_EXPORT_SNMP_TRAP_CONFIG can be used to specify a different name and path to the trap configuration file. By default, the agent looks to see if a <pc>_trapcnfg.xml file exists in the agent's local configuration directory. You can specify the complete path or the path relative to the local configuration directory. On z/OS, to specify the complete path, the PDS should be listed at the end (or omitted and allowed to default to

RKANDATV).IRA_EVENT_EXPORT_SNMP_TRAP=N can be used to disable agent SNMP traps even if the TRAPCNFG.xml file is present.

XML specification

The trap configuration file can include these XML elements:

- SNMP
- TrapDest
- TrapAttrGroup
- Situation
- StatTrap

SNMP is the top-level xml element. TrapDest, TrapAttrGroup, and Situation are elements within the SNMP begin and end tags.

Sample trap configuration file

Review this sample nt_trapcnfg.xml for a Windows OS agent to see how a trap configuration file might be composed. It is placed in the <itm_install_dir>\localconfig\nt directory to enable trap emission for the Windows OS agent. The file is configured to send status traps to a Tivoli Universal Agent monitoring SNMPv1v2 traps on host nt2003infra and to send informs for all situation events to a Netcool/OMNIBus SNMP probe using SNMPv3 running on host 10.21.32.234.

```
<!--C:\IBM\ITM\localconfig\nt\nt_trapcnfg.xml /-->
<SNMP>
  <TrapDest name="UASatMon" Address=" nt2003infra " Version="v1"
    Community="{AES256:keyfile:a}P0hUrmUhCgfFwimS+Q6w+w==" Stat="Y" />

  <TrapDest name="Probe1" Version="v3" Address="10.21.32.234"
    SecLevel="authPriv" User="AuthPrivMD5DES" AuthType="MD5"
    AuthPassKey="{AES256:keyfile:a}yifHSbFcTKHBqvORpzxS6A=="
    PrivType="DES" PrivPassKey=
    "{AES256:keyfile:a}1le2Sx1jJR1M0Ii0EDIvig==" Stat="N" />

  <TrapAttrGroup Table="NT_Paging_File" TrapAttrList="Server_Name,
    %_Usage" />

  <Situation name="NT_Log_Space_Low_pr" sev="2" cat="0"
    mode="HY"
    target="Probe1" />
  <Situation name="NT_Missing_Scheduler_pr" sev="5" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_Paging_File_Critical_pr" sev="5" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_Paging_File_Warning_pr" sev="2" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_Phys_Disk_Busy_Critical_pr" sev="5" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_Phys_Disk_Busy_Warn_pr" sev="2" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_System_File_Warn_pr" sev="2" cat="0"
    mode="HY" target="Probe1" />
  <Situation name="NT_Proc_CPU_Critical_pr" sev="5" cat="0"
    mode="HY" target="Probe1" />
```

```

<Situation name="NT_Proc_CPU_Warn_pr" sev="2" cat="0"
mode="HY" target="Probe1" />
<Situation name="NT_Service_Error_pr" sev="2" cat="0"
mode="RC" target="Probe1" />
<Situation name="NT_System_File_Critical_pr" sev="5" cat="0"
mode="HY" target="Probe1" />
<Situation name="NT_System_File_Warn_pr" sev="2" cat="0"
mode="HY" target="Probe1" />

<StatTrap name="EE_HEARTBEAT" sev="1" interval="15" cat="3" />
<StatTrap name="EE_AUTO_ENTER" sev="1" cat="3" />
<StatTrap name="EE_AUTO_EXIT" sev="1" cat="3" />
<StatTrap name="EE_AUTO_USE_LIMIT" sev="5" cat="3" />
<StatTrap name="EE_TEMS_RECONNECT_LIMIT" sev="5" cat="3" />
<StatTrap name="EE_TEMS_CONNECT" sev="1" cat="4" />
<StatTrap name="EE_TEMS_DISCONNECT" sev="1" cat="4" />
<StatTrap name="EE_SIT_STOPPED" sev="1" cat="4" />
</SNMP>

```

Configuring OMNIBus to receive SNMP alerts

Configure the Netcool/OMNIBus SNMP Probe to accept the SNMP traps and informs of situation events from Tivoli Enterprise Monitoring Agents and Tivoli System Monitoring Agents. The IBM Tivoli Monitoring V6.2.2 Agents installation media has the Management Information Base (mib) and sample rules files that you add to the probe configuration.

Before you begin

Have the IBM Tivoli Monitoring V6.2.2 Agents DVD available. Verify that Tivoli Netcool/OMNIBus V7.x is installed and that the Netcool/OMNIBus SNMP Probe is installed.

Do not configure an enterprise situation for emitting SNMP alerts to the Netcool/OMNIBus SNMP Probe if the hub monitoring server is also configured to forward events for the same situation to the Netcool/OMNIBus Probe for Tivoli EIF because OMNIBus deduplication will not detect that they are the same event.

About this task

Complete these steps to prepare your OMNIBus environment to receive SNMP alerts for situation events from Tivoli monitoring agents.

1. Copy the Tivoli Monitoring rules file and lookup file.
 - a. Locate the mibs/sample_rules/omnibus directory on the Tivoli Monitoring V6.2.2 Agents installation media.
 - b. Copy the `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap` files to `$OMNIHOME/probes/arch/` on the computer where the Netcool/OMNIBus SNMP Probe is installed.
2. Reference the files in the rules that the Netcool/OMNIBus SNMP Probe is using.
 - a. Open the default rules file in a text editor. The default rules file is `$OMNIHOME/probes/arch/mttrapd.rules` unless specified otherwise in the `mttrapd` properties file (Step 3).
 - b. Add the lookup table reference as the first definition:


```
include "<path_to_lookup_file>/
ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.lookup"
```

Table definitions must appear at the start of a rules file, before any processing statements. If you are adding this statement to `mttrapd.rules`, position it after the comments at the head of the file and before the first processing statement. The fully qualified filename must be enclosed in double quotes. Environment variables like `%OMNIHOME%` or `$OMNIHOME` can be used. The Linux and UNIX filename convention, with the `/` forward slash to delimit the path, is also used by Windows.

- c. Add the rules reference in the order in which it should be processed.

```
include "<path_to_rules_file>/
ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules"
```

This statement should be added in the rules file in the location where it should be processed. For example, if adding the include to the default `mttrapd.rules` file, you would want the default rules to first "Check if an SNMPv2 trap and convert to SNMPv1 style tokens". The next block of code in the default `mttrapd.rules` handles Generic traps. The include statement for the `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules` should go after this, possibly as the last line of `mttrapd.rules`. You will best know where to include the rules if you are familiar with the Netcool/OMNIbus SNMP Probe and your event space.

3. Review and edit the Netcool/OMNIbus SNMP Probe properties file:
 - a. Open `$OMNIHOME/probes/arch/mttrapd.props` in a text editor.
 - b. Set the Protocol property to "UDP" or "ALL". Tivoli Monitoring SNMP alerts are sent using UDP.
 - c. Set the RulesFile property if the default rules file for the probe is not `mttrapd.rules`.
 - d. Set the MIBDirs property to the path where the mib files will reside.
4. Make the Tivoli Monitoring mib files available to the Netcool/OMNIbus SNMP Probe:
 - a. Locate the mibs/ directory on the Tivoli Monitoring installation media.
 - b. Copy `canbase.mib` and `cansysg.mib`, to the mib location specified in `mttrapd.props` by the MIBDirs property.
 - c. The `canbase.mib` and `cansysg.mib` include some common SNMP mibs. These mibs must also be available to the SNMP probe:
 - RFC1155-SMI
 - RFC1213-MIB
 - SNMPv2-TC
 - RFC-1212
 - RFC-1215
 If these mibs are not already present in the location specified in `mttrapd.props` by the MIBDirs property, they are publicly available and may be downloaded from the Internet.

Results

You should now have the `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap` rules file and lookup file and the `can*.mib` files that are provided on the Tivoli Monitoring installation media installed on the probe system, the rules and lookup files referenced in the `mttrapd` rules file, and any changes to the default paths or files referenced in the `mttrapd` properties file.

The `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules` file contains a sample mapping of the IBM Tivoli Monitoring SNMP trap variables to the Default alerts.status fields in OMNIBus.

The `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.lookup` file contains these tables:

SituationCategory, which maps the Tivoli Monitoring \$autoSit-Category to OMNIBus's @AlertGroup.

SituationSeverity, which maps the Tivoli Monitoring \$autoSit-Severity to OMNIBus's @Type: 1 - Problem; 2 - Resolution; and 13 - Information. It also changes the severity of an autoSit-Severity=0 clearing trap to 1 so that the OMNIBus generic_clear automation will correlate events.

SituationSource, which enumerates the \$agentSit-Source that identifies whether the situation was an enterprise situation defined at the Tivoli Enterprise Monitoring Server or a private situation defined in the Private Situation Configuration file located in the agent installation directory, `<tema_install_dir>/localconfig/k<pc>`. This table is also use to determine event Class.

What to do next

To activate the new rules and begin receiving alerts from Tivoli monitoring agents, recycle the Netcool/OMNIBus SNMP Probe.

Sample OMNIBus rules for SNMP alerts

The IBM Tivoli Monitoring V6.2.2 Agents installation media has a sample rules files that you add to the Netcool/OMNIBus SNMP Probe configuration.

Notes on creating the @Identifier & @AlertKey

The `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules` use the Tivoli Netcool/OMNIBus Deduplication Automation and Generic Clear Automation. These automations rely on several alert fields, including the Identifier and the AlertKey fields, each of which can be up to 255 characters. The Netcool/OMNIBus rules file standard for setting the Identifier alert field for an SNMP alert is:

```
@Identifier = @Node + " " + @AlertKey + " " + @AlertGroup + " " + @Type + " " + @Agent + " " + @Manager + " " + $specific-trap
```

Because the AlertKey is included in the information that is used to construct the Identifier, you might encounter truncation problems with 255-character AlertKeys used to create your Identifier.

As implemented in the `ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules`:

```
@Identifier = @Node + " " + @AlertKey + " " + $autoSit-Category + " " + @Type + " " + @Agent + " " + @Manager + " " + $specific-trap
```

\$autoSit-Category is an enumeration of the @AlertGroup (24 bytes) and is substituted for @AlertGroup to save 23 bytes in the final Identifier. These are the maximum field lengths of the components used to construct the Identifier:

Field	Size
@Node Max length	32
\$autoSit-Category fixed length	1
@Type Max length	2

Field	Size
@Agent Max length	31
@Manager fixed length	13
\$specific-trap fixed length	2
6 space delimiters	6
Total	87

This leaves 168 characters for the @AlertKey (255-87=168). If @AlertKey is defined as \$agentSit-Name + " (" + \$sitDisplayItem + ")", then \$sitDisplayItem must be less than 133 characters (168-35=133).

Field	Size
agentSit-Name	32
space delimiter	1
parentheses	2
Total	35

A best practice is to limit \$sitDisplayItem to 128 characters to maintain consistency with the IBM Tivoli Monitoring EIF probe rules. The sample rules enforce this limit using

```
$sitDisplayItem=substr($sitDisplayItem, 1, 128)
```

.

Situations written for attribute groups (such as Event Log) that generate pure events can be deduplicated by using the \$agentSit-Name, but many might require additional information to uniquely identify the event. Use the \$sitDisplayItem attribute to construct this additional data. The AlertKey will then be

```
$agentSit-Name + " (" + $sitDisplayItem + ")"
```

Use case statements based on the \$agentSit-Table field to identify all events based on a specific table.

Use case statements based on the \$agentSit-Name if individual situations need unique \$sitDisplayItems.

The **extract** command can be used to extract the value of any of the name value pairs from the \$sitAttributeList using regex pattern matching. This command extracts the value of the Description key and removes the quotes. An example is provided in the Sample rules for agentSitPureEvent traps based on the NTEVTLOG \$agentSit-Table.

```
$sitDisplayItem=extract($sitAttributeList,"Description=.(.+.;.*?")
```

Compatibility notes

@ExtendedAttr

OMNIbus V7.2 and greater defines the @ExtendedAttr column in the ObjectServer. The **nvp** functions are provided to allow manipulation of name-value pairs in the @ExtendedAttr alert field. The sitAttributeList varbind is formatted to allow direct mapping into the @ExtendedAttr, but this function is commented out to allow the rules to parse when the MTTRAPD probe connects to an OMNIbus ObjectServer V7.0 or V7.1. Uncomment the two lines in the ibm-TIVOLI-CANSYSSG-

MIB.include.snmptrap.rules file that set @ExtendedAttr if you are forwarding events to OMNIBus V7.2 or greater.

```
# @ExtendedAttr = $sitAttributeList
```

@Class

The @Class alert field is used to associate Tivoli Netcool/OMNIBus Tools with Events displayed in the Tivoli Netcool/OMNIBus EventList.

For Tivoli Netcool/OMNIBus 7.2x and below, see the Netcool/OMNIBus documentation for more information on creating and editing classes. By default, these class values are not defined in your ObjectServer.

Setting @Class to a value that is not defined in the OMNIBus ObjectServer causes no problems, but if you prefer to not set the @Class, uncomment this line in the ibm-TIVOLI-CANSYSSG-MIB.include.snmptrap.rules file to clear the @Class field before the event is forwarded to OMNIBus. #

```
@Class = ""
```

SNMP message types for agent and situation state alerts

Tivoli monitoring agents emit three types of SNMP messages: agentStatusEvent to convey agent operational status, agentSitSampledEvent for situations that sample at intervals and become true, and agentSitPureEvent for situations that receive unsolicited notifications. They are defined in the canbase.mib and cansysg.mib files, which are available on the IBM Tivoli Monitoring IBM Tivoli Monitoring Agents installation media.

The agent situation state SNMP traps are sent using enterprise 1.3.6.1.4.1.1667.1.3 (Candle-BASE-MIB::candle-Alert-MIB).

agentStatusEvent

The agentStatusEvent is a monitoring agent operational status information trap generated by the Tivoli Autonomous Agent SNMP Event Exporter to inform and notify about a specific agent operational event.

Specific trap: 20

Table 23. SNMP trap variables for agentStatusEvent

Variable	Description	Access	Status	OID
agentSit-Name	The situation name, up to 32 bytes, identifies the name and nature of the status event.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.3
agentSit-OriginNode	The name of the managed system where the situation was evaluated, up to 32 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.4

Table 23. SNMP trap variables for agentStatusEvent (continued)

Variable	Description	Access	Status	OID
agentSit-LocalTimeStamp	The timestamp when the situation state changed. The format is CYYMMDDHHMMSSmmm (such as 1090415094501000 for April 15, 2009 at 09:45:01) where: C = Century (1 for 21st) Y = Year M = Month D = Day H = Hour M = Minute S = Second m = millisecond	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.5
autoSit-Category	Assigned situation category. Valid values are: 0 - Threshold 1 - Network Topology 2 - Error 3 - Status 4 - Node Configuration 5 - Application Alert 6 - All Category 7 - Log Only 8 - Map 9 - Ignore	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.6
autoSit-Severity	Assigned situation severity. Valid values are: 0 - Cleared 1 - Indeterminate 2 - Warning 3 - Minor 4 - Major 5 - Critical	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.7
autoSit-StatusText	The agent status trap description message text, from 0 to 256 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.9
autoSit-Interval	The agent status trap interval.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.11

agentSitSampledEvent

A sampled situation event was detected. This trap was generated by the Tivoli Autonomous Agent SNMP Event Exporter in response to a situation threshold being exceeded at the time of the data sampling.

Specific trap: 21

Table 24. SNMP trap variables for agentSitSampledEvent

Attribute	Description	Access	Status	OID
agentSit-Application	This is the product application name, from 1 to 8 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.1
agentSit-Table	This is the name of the product application table (attribute group), from 1 to 12 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.2
agentSit-Name	The situation name, up to 32 bytes, identifies the name and nature of the status event.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.3
agentSit-OriginNode	The name of the managed system where the situation was evaluated, up to 32 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.4
agentSit-LocalTimeStamp	The timestamp when the situation state changed. The format is CYYMMDDHHMMSSmmm (such as 1091031183005000 for October 31, 2009 at 18:30:05) where: C = Century (1 for 21st) Y = Year M = Month D = Day H = Hour M = Minute S = Second m = millisecond	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.5
agentSit-Context	Unique situation context identifier, expressed as an integer (-2147483647 to 2147483647). This is the handle number identifying an agent running request. In an SNMP environment, trap-direct polling is typically used whereby a trap is received and the network manager polls the originating agent for additional detailed information. This identifier is used to supply context for the targeted agent to correlate the request to the problem source. Although agentSit-Context is sent, it is not used in this release.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.6

Table 24. SNMP trap variables for agentSitSampledEvent (continued)

Attribute	Description	Access	Status	OID
agentSit-SampleInterval	Sampled situation interval in seconds, from 0 to 86400.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.7
agentSit-Source	Situation current status. The valid values are: 0 - Undefined 1 - Enterprise, meaning the situation was defined on the Tivoli Enterprise Monitoring Server 2 - Private, meaning the situation was defined by the local private situation configuration file.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.20
autoSit-Category	Assigned situation category. Valid values are: 0 - Threshold 1 - Network Topology 2 - Error 3 - Status 4 - Node Configuration 5 - Application Alert 6 - All Category 7 - Log Only 8 - Map 9 - Ignore	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.6
autoSit-Severity	Assigned situation severity. Valid values are: 0 - Cleared 1 - Indeterminate 2 - Warning 3 - Minor 4 - Major 5 - Critical	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.7
autoSit-Predicates	This is the situation formula, up to 3210 bytes, in the form attributeName Operator CompareValue. When the formula has multiple expressions, their Boolean AND or OR connectors are shown.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.8
sitAttributeList	The attribute values for the situation that is assigned to the monitoring agent, from 0 to 3200 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.5

agentSitPureEvent

A pure situation event was detected. This trap was generated by the Tivoli Autonomous Agent SNMP Event Exporter in response to a situation threshold being exceeded. The variables in a pure event trap are identical to those for a sampled event trap except there is no agentSit-SampleInterval because pure events

are not sampled; rather the arrival of unsolicited data from the monitored attribute group causes the situation to become true. A situation created with an attribute group for a system log, for example, opens a pure event when a log entry arrives.

Specific trap: 22

Table 25. SNMP trap variables for agentSitPureEvent

Attribute	Description	Access	Status	OID
agentSit-Application	This is the product application name, from 1 to 8 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.1
agentSit-Table	This is the name of the product application table (attribute group), from 1 to 12 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.2
agentSit-Name	The situation name, up to 32 bytes, identifies the name and nature of the status event.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.3
agentSit-OriginNode	The name of the managed system where the situation was evaluated, up to 32 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.4
agentSit-LocalTimeStamp	The timestamp when the situation state changed. The format is CYYMMDDHHMMSSmmm (such as 1091031183005000 for October 31, 2009 at 18:30:05) where: C = Century (1 for 21st) Y = Year M = Month D = Day H = Hour M = Minute S = Second m = millisecond	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.5

Table 25. SNMP trap variables for agentSitPureEvent (continued)

Attribute	Description	Access	Status	OID
agentSit-Context	Unique situation context identifier, expressed as an integer (-2147483647 to 2147483647). This is the handle number identifying an agent running request. In an SNMP environment, trap-direct polling is typically used whereby a trap is received and the network manager polls the originating agent for additional detailed information. This identifier is used to supply context for the targeted agent to correlate the request to the problem source. Although agentSit-Context is sent, it is not used in this release.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.6
agentSit-Source	Situation current status. The valid values are: 0 - Undefined 1 - Enterprise, meaning the situation was defined on the Tivoli Enterprise Monitoring Server 2 - Private, meaning the situation was defined by the local private situation configuration file.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.10.1.20
autoSit-Category	Assigned situation category. Valid values are: 0 - Threshold 1 - Network Topology 2 - Error 3 - Status 4 - Node Configuration 5 - Application Alert 6 - All Category 7 - Log Only 8 - Map 9 - Ignore	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.6
autoSit-Severity	Assigned situation severity. Valid values are: 0 - Cleared 1 - Indeterminate 2 - Warning 3 - Minor 4 - Major 5 - Critical	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.7

Table 25. SNMP trap variables for agentSitPureEvent (continued)

Attribute	Description	Access	Status	OID
autoSit-Predicates	This is the situation formula, up to 3210 bytes, in the form attributeName Operator CompareValue. When the formula has multiple expressions, their Boolean AND or OR connectors are shown.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.8
sitAttributeList	The attribute values for the situation that is assigned to the monitoring agent, from 0 to 3200 bytes.	read-only	mandatory	1.3.6.1.4.1.1667.1.2.1.5

Trap configuration XML specification

Use these elements in SNMP XML files to configure traps for any agent type that you want to specify for the event receiver.

SNMP element

The SNMP element of the trap configuration XML specification is the top-level XML element. TrapDest, TrapAttrGroup, and Situation are elements within the SNMP begin and end tags.

```
<SNMP>
  <TrapDest name="OMNIbus2" Address="nswin21a" Stat="Y" />
  <situation name="*" target="OMNIbus2" />
</SNMP>
```

TrapDest element

Use TrapDest elements in a trap configuration XML file to define a trap receiver.

The TrapDest element requires the name and address attributes. Default values are used for any other attributes that are not specified.

```
<TrapDest name="LABEL" Address="HOSTNAME" />
```

Table 26. TrapDest element XML specification

Attribute	Description	Required	Default	SNMPv1/v2 or SNMPv3
Name=	Alphanumeric label that is used to identify the Trap Destination.	Required		
Address=	Trap receiver's TCP/IP address or hostname.	Required		All
IP=	ip protocol: "4" "6" 4 is IPv4; 6 is IPv6	Optional	"4"	All
Port=	Trap receiver TCP/IP trap listening port.	Optional	"162"	All

Table 26. TrapDest element XML specification (continued)

Attribute	Description	Required	Default	SNMPv1/v2 or SNMPv3
BindAddress=	Used to specify which local interface to use for SNMP traffic. The interface specified must match the IP setting.	Required if the host has multiple network interfaces defined. Otherwise the trap send might fail with error number 22.	First available	All
Version=	Specify SNMP trap version. Valid string values are (case insensitive) : v1 , v2 , v3	Optional	v1	All
Type=	Trap Inform Type must match the Version. Version= "v1" "v2" Type Must be "Trap" Version= "3" Type Must be "Inform"	Optional	Matches version	All
Stat=	Stat is used on a destination to send all status traps to that receiver when Stat="Y". Leave at the default "N" to disable sending. Also, you can override the status trap information and send specific ones to specific receivers.	Optional	"N"	All
Community=	Specify trap community name string. Should be encrypted using itmpwdsnmp, but clear-text is also allowed. (1-63 characters)	Optional	public	v1 and v2
SecModel=	Specify the security model. Only USM supported.	Optional	USM	v3

Table 26. *TrapDest* element XML specification (continued)

Attribute	Description	Required	Default	SNMPv1/v2 or SNMPv3
SecLevel=	Specify the Authentication and Privacy levels. The levels supported are: noAuthNoPriv - no authentication, and no privacy authNoPriv – authentication no privacy authPriv - authentication and privacy	Required for v3		v3
User=	Specify the account name	Required for v3		v3
AuthType=	Specify the authentication protocol. The protocols supported are: MD5 and SHA	Required for v3 SecLevel= authNoPriv or authPriv		v3
AuthPassKey=	Specify the authentication password Should be encrypted using itmpwdsnmp, but clear-text is also allowed. (1-63 characters)	Required for v3 SecLevel= authNoPriv or authPriv		v3
PrivType=	Specify the privacy protocol. The protocol supported are: DES	Required for v3 SecLevel= authPriv		v3
PrivPassKey=	Specify the privacy password. Should be encrypted using itmpwdsnmp, but clear-text is also allowed. (1-63 characters)	Required for v3 SecLevel= authPriv		v3
Timeout=	Specify the timeout (in seconds, integer) for the acknowledgement of SNMPv3 message (minimum 1)	Optional	2	v3

Table 26. TrapDest element XML specification (continued)

Attribute	Description	Required	Default	SNMPv1/v2 or SNMPv3
Retries=	Specify the number of retransmissions when a timeout occurs (min 0, max 5)	Optional	3	v3

Encrypting SNMP PassKeys in agent trap configuration files: The itmpwdsnmp uses GSKIT to either interactively encrypt a string or to encrypt all SNMP password strings in a trap configuration xml file.

itmpwdsnmp **[-b | -n]***your_agent_trapcnfg.xml***[-?]**

where:

no arguments specifies interactive mode

-b specifies to create a backup file. There is no prompting to delete the backup file.

-n specifies that no backup file is to be created.

your_agent_TRAPCNFG.xml is a trap configuration xml file that contains plaintext SNMP password strings.

-? displays usage

Windows `<itm_install_dir>\TMAITM6\itmpwdsnmp.exe`

Linux **UNIX** `<itm_install_dir>/bin/itmpwdsnmp.sh`

TrapAttrGroup element

Use the TrapAttrGroup element in a trapcnfg.xml file to specify which attributes from an attribute group to include in situation event traps.

In this syntax example, situations written for the Windows OS Paging File attribute group will send an SNMP trap with the server name, usage percentage and the usage peak values to the event receiver.

```
<TrapAttrGroup Table="NT_Paging_File" TrapAttrList="Server_Name,%_Usage,%_Usage_Peak" />
```

This element can be used to decrease the amount of attribute data sent in each trap request, reduce the possibility of trap fragmentation, and reduce the received data to include only what is relevant.

The TrapAttrGroup element sets the default attributes that will be sent for all situation that run against the Table. Individual situations may override the TrapAttrGroup settings by specifying a TrapAttrList attribute in the situation element.

If a TrapAttrGroup element is not defined for an attribute table, all attributes in the situation's data row are added to the sitAttributeList varbind of the traps sent for situations based on this attribute table. Attributes used in the situation predicate are added first and remaining attributes are added until the PDU maximum length of 1500 bytes is reached.

Table 27. TrapAttrGroup element XML specification

Attribute	Description
Table=	The name of the attribute table. For manually creating this file, you can look in the agent's attribute file, k<pc>.atr to identify the table names, where <pc> is the two-character product code.
TrapAttrList=	A comma delimited list of attributes to be included in the sitAttributeList varbind of the traps sent for situations based on this attribute table.

Situation element

Use situation elements in a trap configuration XML file to define the trap sent for the situation.

```
<situation name="Situation_ID" target="TrapDest_Name" />
```

The Situation element requires the name and target attributes. Default values are used for any other attributes that are not specified. The * asterisk wildcard can be specified for the situation name or target or both:

- Specifying the wildcard for situation name represents all situations. For example the following would send traps for all defined true situations to the defined TrapDest named trapProbe1 :

```
<situation name="*" target="trapProbe1" />
```

- Specifying the wildcard for target enables sending the situation specified in the situation name field to all defined targets:

```
<situation name="NT_Disk_Low" target="*" />
```

- Specifying the wildcard for both situation name and target would enable sending all traps to all defined trap receivers.

- Named situations have precedence over wildcard definitions. If a situation definition includes a wildcard and another situation definition names a situation or the target, the first occurrence of the named situation definition is honored.

Example:

```
<situation name="*" target="OMNibus2" />
<situation name="My_Missing_Process" target="MyReceiver" />
<situation name="NT_AA_Missing_Test" target="OMNibus1" />
<situation name="NT_AA_Missing_Test" target="OMNibus2" />
<situation name="NT_ABC_Missing_Test" target="*" />
```

The **My_Missing_Process** situation sends a trap to **MyReceiver** instead of **OMNibus2**. And **NT_ABC_Missing_Test** is sent to **MyReceiver**, **OMNibus1**, and **OMNibus2** instead of solely to **OMNibus2** because the situation is defined explicitly rather than using the wildcard.

If a situation is defined more than once, the first occurrence of a situation definition has precedence. Looking again at the example, **NT_AA_Missing_Test** is sent to **OMNibus1** and not **OMNibus2** because the first occurrence of the definition for the same situation specifies **OMNibus1**.

Table 28. Situation element XML specification

Attribute	Description	Required	Default
Name=	This is the ID or short name of the situation.	Required	

Table 28. Situation element XML specification (continued)

Attribute	Description	Required	Default
Target=	Specify a previously defined TrapDest. "*" implies send trap to all defined destinations.	Required	
Sev=	Specify trap severity. The standard trap severities are: 0 – Cleared 1 – Indeterminate 2 – Warning 3 – Minor 4 – Major 5 – Critical	Optional	2
Cat=	Specify trap category. The standard trap categories are: 0 - Threshold 1 – Network Topology 2 – Error 3 – Status 4 – Node Configuration 5 – Application Alert 6 – All Category 7 – Log Only 8 – Map 9 – Ignore	Optional	
Mode=	Used to specify behavior for SNMP trap emission on sampled situations. The standard modes are: RC – Rising Continuous, whereby traps are sent on each true evaluation of a situation. (Pure events are always RC.) No specific clearing trap will be sent. HY – Hysteresis, whereby a trap is sent the first time the sampled situation evaluates as true. A clearing trap will be sent once the sampled value no longer meets the criteria of the situation.	Optional	RC

Table 28. Situation element XML specification (continued)

Attribute	Description	Required	Default
Pred=	The situation predicate (formula) is sent in the trap's autoSit-Predicates varbind. The Pred attribute allows you to omit the situation predicate by setting Pred="N". This can be useful if you do not care to receive the predicate or if a complex predicate is taking up too much of the trap PDU, and you want more room to send situation attributes in the sitAttributeList varbind.	Optional	Y
Table=	Table name of the attribute group. Used with the TrapAttrlist to identify a subset of attributes used to construct the sitAttributeList varbind.	Required only if a TrapAttrList is used.	
TrapAttrList=	A comma delimited list of attributes to be included in the sitAttributeList varbind of the traps sent for situation. Values specified here will override any TrapAttrList values specified in a TrapAttrGroup element for the table that the situation is running against.	Optional	

Note: Situations for multiple-row attribute groups that include a display item are limited to sending one trap for the first row that evaluates to true, but not for any subsequent rows.

StatTrap

Use the StatTrap element in an SNMP trap configuration file to modify the default configuration of the predefined agent lifecycle status traps.

In this syntax example, the predefined trap for EE_HEARTBEAT was modified to specify severity 1 (Indeterminate) for the event, a 30-minute sampling interval, and trap category 3 (Status).

```
<StatTrap name="EE_HEARTBEAT" sev="1" interval="30" cat="3" />
```

There are eight predefined agent lifecycle traps and their default values are given in this table. By default, these traps are sent to all TrapDest trap destinations where the Stat attribute is "Y". If the Stat attribute is omitted from a TrapDest element the default value is "Y".

Table 29. Agent lifecycle status traps

Status Trap	Description	Severity	Category
EE_HEARTBEAT	A heartbeat indicates that the agent is running and events emitted can reach the trap destination. This is the only status trap with a set interval: 15 minutes.	1 – Indeterminate	3 – Status
EE_AUTO_ENTER	The agent has entered autonomous mode.	1 – Indeterminate	3 – Status
EE_AUTO_EXIT	The agent has exited autonomous mode.	1 – Indeterminate	3 – Status
EE_AUTO_USE_LIMIT	The agent has reached the storage limit specified by the IRA_AUTONOMOUS_LIMIT environment variable. Additional events generated while the agent is disconnected from the monitoring server may not be uploaded on reconnect.		3 – Status
EE_TEMS_RECONNECT_LIMIT	The agent has reached the retry limit specified by the CTIRA_MAX_RECONNECT_TRIES environment variable. The agent will no longer attempt to connect to a monitoring server and will shutdown. In ITM 6.2.2, the default value of CTIRA_MAX_RECONNECT_TRIES has been changed to 0, so the agent will never shutdown.		3 – Status
EE_TEMS_CONNECT	The agent has successfully connected to the monitoring server.	1 – Indeterminate	4 - Node Configuration
EE_TEMS_DISCONNECT	The agent has lost connection with the monitoring server.	1 – Indeterminate	4 - Node Configuration
EE_SIT_STOPPED	The situation has stopped	1 – Indeterminate	4 - Node Configuration

Use the StatTrap element to configure agent lifecycle traps.

Table 30. StatTrap element XML specification

Status Trap	Description	Required	Default
Name=	This must be the name of a predefined Life-Cycle status trap. EE_HEARTBEAT EE_AUTO_ENTER EE_AUTO_EXIT EE_AUTO_USE_LIMIT EE_TEMS_RECONNECT_LIMIT EE_TEMS_CONNECT EE_TEMS_DISCONNECT EE_SIT_STOPPED	Optional	
Target=	Specify a previously defined TrapDest. "*" implies send trap to all defined destinations. If no Target is defined, all TrapDest with Stat="Y" will receive the status trap.	Required	
Sev=	Specify trap severity. The standard trap severities are: 0 – Cleared 1 – Indeterminate 2 – Warning 3 – Minor 4 – Major 5 – Critical	Optional	Varies
Cat=	Specify trap category. The standard trap categories are: 0 - Threshold 1 - Network Topology 2 – Error 3 – Status 4 - Node Configuration 5 - Application Alert 6 - All Category 7 - Log Only 8 – Map 9 - Ignore	Optional	Varies
Interval=	Interval specifies in minutes how often the EE_HEARTBEAT status trap is emitted. Interval is ignored for the other status traps because they are pure events.	Optional	15

SNMP PassKey encryption: itmpwdsnmp

Use the itmpwdsnmp CLI command to interactively encrypt a password or add it to the SNMP trap configuration XML file to encrypt all SNMP passwords.

The itmpwdsnmp uses GSKIT to either interactively encrypt a string or to encrypt all SNMP password strings in a trap configuration xml file.

itmpwdsnmp **[[*-b* ^ | *-n*]*your_agent_trapcnfg.xml*][*-?*]**

where:

no arguments specifies interactive mode

-b specifies to create a backup file. There is no prompting to delete the backup file. The ^ is required to echo the pipe on windows.

-n specifies that no backup file is to be created.

your_agent_TRAPCNFG.xml is a trap configuration xml file that contains plaintext SNMP password strings.

-? displays usage

If a **-b** or **-n** backup option is not specified when encrypting a Trap Configuration xml file, you are prompted to delete the backup. The backup of the original input Trap Configuration xml file is created in the same directory as the original with a date and timestamp appended to the original file name.

Windows	<itm_install_dir>\TMAITM6\itmpwdsnmp.exe
Linux	UNIX <itm_install_dir>/bin/itmpwdsnmp.sh

CLI examples

This command will interactively encrypt a string:

```
itm_pwdsnmp
```

Enter string to be encrypted:

Confirm string:

{AES256:keyfile:a}GbHOIF7KPYZS80RripX4QQ==

You can now copy the encrypted string into the trap configuration file. This command will encrypt all SNMP password strings in the trap configuration file and then remove the backup of the original file:

```
itmpwdsnmp -n nt_trapcnfg.xml
```

Program Summary

Community strings encrypted 1

AuthPassKey strings encrypted 2

EncryptPassKey strings encrypted 1

Agent Service Interface

Use the agent service interface for retrieving information from an installed agent, whether it is a Tivoli Enterprise Monitoring Agent or Tivoli System Monitoring Agent.

The Agent Service Interface is accessed through the IBM Tivoli Monitoring Service Index utility. The interface operates as an Internet server, accepting and validating requests, dispatching requests to the agent for processing, and gathering and formatting reply data using the HTTP or HTTPS application protocol over TCP/IP.

Starting the Agent Service Interface

Start the Agent Service Interface from your browser to get a menu of choices for reporting agent information, getting situation status, displaying short-term history, and for making service requests in XML.

Before you begin

You need to have an administrator user ID for the operating system where the monitoring agent is installed to access the Agent Service Interface and its functions.

About this task

Follow these steps to start the IBM Tivoli Monitoring Service Index utility and then log onto the Agent Service Interface for the agent that you want to get information about.

1. Start the IBM Tivoli Monitoring Service Index by entering `http://<host name>:1920` or `https://<host name>:3661`, where *host name* is the fully qualified name or IP address of the computer where the agent is installed. A list of the started services is displayed.
2. Click the **<PC> Agent Service Interface** (where *<PC>* is the two-character component code) link for the application to work with.
3. As prompted, enter the administrator-level user name and password for the operating system.

Results

After you have been authenticated, the Agent Service Interface Navigator page is displayed with links to **Agent Information**, **Situations**, **History**, and **Service Interface Request**. The Navigator page is the `navigator.htm` file that is installed at this location by default:

Windows	<install_dir>\localconfig\html	
Linux	UNIX	<install_dir>/config/HTML
z/OS	RKANPARU DDNAME dataset	

Agent Service Interface - Agent Information

Select **Agent Information** from the Agent Service Interface menu to retrieve a report of pertinent data about the agent, including the environment file settings.

HOSTNAME

This is the fully qualified name of the computer, such as **myitm.raleigh.ibm.com**.

NODENAME

This is the name of the managed system, such as **Primary:MYITM:NT**.

SUBSYSID

If the agent has subnodes (subagents), this is the name. Otherwise, the subsystem ID is **Primary**.

NODEINFO

This is the type of system and operating platform, such as **Win2003~5.2-SP2**.

PRODUCT

This is the two-character product code of the agent, such as **NT**.

VERSION

This is the installed version of the agent, such as **06.22.00**.

LEVEL A=00:WINNT C=06.22.00.00:WINNT G=06.22.00.00:WINNT

PATCHLEVEL A=00:WINNT;C=06.22.00.00:WINNT;G=06.22.00.00:WINNT;

AFFINITY

This is value that identifies the affinity of the agent to the Tivoli Management Services components. For example, **%IBM.STATIC021000000000A00000u0a4**.

BOOTTIME

This is the day of the week, the calendar date and time when the agent completed startup, such as **Wed Jul 29 15:15:33 2009**.

ENVFILE

This is a list of the current parameter settings in the agent environment file. If you need to change any of the values, you can open the environment file through Manage Tivoli Monitoring Services or in a text editor on distributed systems.

Here is an example of the Windows OS environment file as it is displayed in Agent Information report:

```
* CANDLE_HOME=d:\IBM\ITM
* KBB_RAS1=ERROR
* KBB_VARPREFIX=%
* KBB_VARPREFIX=$
* KBB_RAS1_LOG=d:\IBM\ITM\tmaitm6\logs\$(computername)_nt_kntcma_$
(sysutcstart)-.log INVENTORY=d:\IBM\ITM\tmaitm6\logs\$(computername)
_nt_kntcma.inv COUNT=03 LIMIT=5 PRESERVE=1 MAXFILES=9
* TIMEOUT=600
* ITMDEPLOY_AGENTDEPOT=d:\IBM\ITM\tmaitm6\agentdepot
* ICCRTE_DIR=d:\IBM\ITM\GSK7
* CSV1_PATH=d:\IBM\ITM\GSK7\lib
* CSV2_PATH=d:\IBM\ITM\GSK7\bin
* KBB_VARPREFIX=$
* PATH!=$(CSV1_PATH);$(CSV2_PATH);$(PATH)
* KEYFILE_DIR=d:\IBM\ITM\keyfiles
* KDEBE_KEYRING_FILE=d:\IBM\ITM\keyfiles\keyfile.kdb
* KDEBE_KEYRING_STASH=d:\IBM\ITM\keyfiles\keyfile.sth
* KDEBE_KEY_LABEL=IBM Tivoli Monitoring Certificate
* KBB_IGNOREHOSTENVIRONMENT=Y
* JAVA_HOME=d:\IBM\ITM\java\java50\jre
* KBB_IGNOREHOSTENVIRONMENT=N
* PATH=d:\IBM\ITM\GSK7\LIB;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\
System32\Wbem;D:\IBM\SQLLIB\BIN;D:\IBM\SQLLIB\FUNCTION;D:\IBM\SQLLIB\
SAMPLES\REPL;d:\IBM\ITM\bin;d:\IBM\ITM\bin\dll;d:\IBM\ITM\InstallITM;
d:\IBM\ITM\TMAITM6;d:\IBM\ITM\InstallITM
```

Agent Service Interface - Situations

Select the **Situations** option of the Agent Service Interface to see the status and statistics of each situation, including private situations, for the monitoring agent.

The Situations report gives some vital statistics about each situation on the agent. The setting of the agent environment variable **IRA_EVENT_EXPORT_SIT_STATS** determines the level of detail given.

Situation name

Above each situation summary page is the name of the situation. If this is a private situation, the name will be appended with **_pr**.

TYPE Sampled or Pure. A situation is sampled if it samples data at regular intervals. Pure events are unsolicited notifications. The Windows Event Log and Windows File Change attribute are examples of attribute groups that report pure events.

INTERVAL

The interval between data samples, in seconds. If situations for this attribute group trigger pure events, there is no sampling interval and the value shows as 0.

ROWSIZE

This is the row size.

What is this? The character count of the formula?

FIRSTSTARTTIME

This is the day of the week, calendar day, and time when the situation is initially started after the agent starts.

LASTSTARTTIME

This is the day of the week, calendar day, and time when the situation was most recently started.

LASTSTOPTIME

This is the day of the week, calendar day, and time when the situation was most recently stopped.

FIRSTEVENTTIME

This is the day of the week, calendar day, and time of the first occurrence that the situation became true and opened an event since the situation was started.

LASTTRUETIME

This is the day of the week, calendar day, and time when the situation most recently became true and opened an event.

LASTFALSETIME

This is the day of the week, calendar day, and time when the situation state evaluated to false after an earlier sampling evaluated to true.

TIMESRECYCLED

This is the number of times the situation was stopped and started since the agent has been online.

TIMESAUTONOMOUS

This is the number times since startup that the situation entered autonomous state because the enterprise monitoring agent was disconnected from its monitoring server, followed by the DAY statistics:

DAY

DATE that the most recent statistical data was collected. If this is an enterprise situation, this is since the agent was most recently connected.

TRUESAMPLES is the number of times the situation evaluated to true while the agent was disconnected from the monitoring server.

FALSESAMPLES is the number of times the situation evaluated to false after a prior true while the agent was disconnected from the monitoring server.

TRUERATIO is the percentage of the number of times the situation evaluates to true compared with the false state.

FALSERATIO is the percentage of the number of times the situation evaluates to false compared with the true state.

HOURROWS is the number of rows of data that have been reported.

HOURTRUE is the number of hours that the situation remained true while the agent was disconnected from the monitoring server.

HOURFALSE is the number of hours that the situation remained false while the agent was disconnected from the monitoring server.

Related reference

“Private situation XML specification” on page 133

Agent Service Interface - History

Select **History** in the Agent Service Interface to display the private history data samples that have been saved for the selected attribute group table.

You can filter the report to show only the attributes that you are interested in by clearing the check box next to any unwanted attributes. Select a start date and time and an end date and time, then click **Report**. The report is displayed in a table below the attributes, showing historical data samples for the attribute group, one column per attribute and one row per sampling, for the time period specified, up to 5000 rows. If you do not see the rows you are interested in within the 5000 limit, you can generate another report after narrowing the time range.

Related reference

“Private history” on page 147

“Private situation XML specification” on page 133

Agent Service Interface - Service Interface Request

Select **Service Interface Request** in the Agent Service Interface to enter commands in XML format for information about the agent: attribute group definition; SNMP trap configuration file .

Agent properties information

This request agent identification information. The data retrieved is in three sections:

Agent ID – agent machine hostname, Managed System name, subnode list, and OS information.

Product ID – product name, version, maintenance and patch level data, product affinity and features.

Environment ID – active environment variable settings.

Table 31. Agent Service Interface <AGENTINFO> request.

Tag	Description
<AGENTINFO>	Enter begin and end AGENTINFO tags to make an agent property request..

Example:

```
<AGENTINFO>
</AGENTINFO>
```

Attribute group report

REPORT is used to request a report of the attribute group specified in the TABLENAME attribute, such as UNIXOS or NTPROCESS.

Table 32. Agent Service Interface <REPORT> request

Tag	Description
<REPORT>	Retrieve application table data.

Table 32. Agent Service Interface <REPORT> request (continued)

Tag	Description
<SQLTABLE>	The SQLTABLE begin and end tags enclose the TABLENAME tagging pair to identify the SQL table definition set.
<TABLENAME>	The TABLENAME begin and end tags enclose the table name to report. This is the name as it appears bracketed by begin and end tags. If you are not sure what the spelling is of the table, you can find it in the tabl field of the agent .atr file, located in the <install_dir>/TMAITM6/ATTRLIB directory.
<OUTPUT>	Optional. Use OUTPUT begin and end tags and their subordinate tags to filter and refine the report. <COLUMN> Define selected column name bracketed by begin and end tags. <FILTER> Define output data rows filter criteria with begin and end tags. The filter follows the same syntax as the private situation <CRITERIA> element. See "Private situation XML specification" on page 133.

Examples:

This command retrieves the contents of the entire Windows OS agent process table:

```
<REPORT>
  <SQLTABLE>
    <TABLENAME>UNIXOS</TABLENAME>
  </SQLTABLE>
</REPORT>
```

In this example, the Windows OS agent process data is retrieved for the specified attributes (columns):

```
<REPORT>
  <SQLTABLE>
    <TABLENAME>NTPROCESS</TABLENAME>
    <OUTPUT>
      <COLUMN>ORIGINNODE</COLUMN>
      <COLUMN>TIMESTAMP</COLUMN>
      <COLUMN>INSTCNAME</COLUMN>
      <COLUMN>IDPROCESS</COLUMN>
      <COLUMN>PCTPRCSTME</COLUMN>
      <COLUMN>THREADCNT</COLUMN>
      <COLUMN>WRKINGSET</COLUMN>
    </OUTPUT>
  </SQLTABLE>
</REPORT>
```

Agent properties information

This request agent identification information. The data retrieved is in three sections:

Agent ID – agent machine hostname, Managed System name, subnode list, and OS information.

Product ID – product name, version, maintenance and patch level data, product affinity and features.

Environment ID – active environment variable settings.

Table 33. Agent Service Interface <AGENTINFO> request

Tag	Description
<AGENTINFO>	Enter begin and end AGENTINFO tags to make an agent property request..

Example:

```
<AGENTINFO>
</AGENTINFO>
```

Agent application

Table 34. Agent Service Interface <APPLICATION> request

Tag	Description
<APPLICATION>	APPLICATION begin and end tags surround the application request definition.

Example:

```
<APPLICATION>
</APPLICATION>
```

SNMP trap configuration

Use the SNMP trap configuration request to see the contents of the SNMP trap configuration file that was created.

Table 35. Agent Service Interface <SNMP> request

Tag	Description
<SNMP>	Begin and end <SNMP> tags to display the SNMP trap configuration file definitions.

Example:

```
<SNMP>
</SNMP>
```

Example of the SNMP configuration file output:

```
<SNMP>
<TrapDest name="US-West" Address="myworkstation" Port="162"
Type="2" IP="4" STAT="Y" />
<situation name="Free_DiskSpace_Low" target="US-West" SEV="4" />
<situation name="Is_KFC_Running" target="US-West" sev="4" cat="5"
mode="HY" />
<StatTrap name="EE_HEARTBEAT" Interval="1" />
</SNMP>
```

Agent monitoring statistics

Use the AGENTSTAT command to report the activity of all situations or only those specified on this system.

Table 36. Agent Service Interface <AGENTSTAT> request

Tag	Description
<AGENTSTAT>	The root element to specify that this is a request for the situation monitoring statistics for the agents on this system. Enclose the SITUATION attribute in begin and end AGENTSTAT tags.
<SITUATION>	Enclose the situation selection properties within SITUATION begin and end tags. <NAME> Enter the name of the situation to report or *ALL to report all situations. Default: *ALL. <PERIOD> Optional. Specify the number of days to report, from 0 for today to 7 for the past week through today. Default: 0 <DETAIL> Optional. Yes specifies to report hourly detailed data; No to output state information only. Default: No.

Examples:

In this example, the AGENTSTAT command is used to retrieve today's situation statistics:

```
<AGENTSTAT>
<SITUATION>
  <NAME>*ALL</NAME>
</SITUATION>
</AGENTSTAT>
```

This example requests detailed situation statistics from the **Is_kfc150_Running** situation for the past 3 days:

```
<AGENTSTAT>
<SITUATION>
  <NAME>Is_kfc150_Running</NAME>
  <PERIOD>3</PERIOD>
  <DETAIL>Y</DETAIL>
</SITUATION>
</AGENTSTAT>
```

Situation stop or start

Use the SITCONTROL request to stop, start, or recycle a private situation or enterprise situation for the agent.

Table 37. Agent Service Interface <SITCONTROL> request

Tag	Description
<SITCONTROL>	Specify situation control request.
<SITUATION>	Enclose the target situation and the action to take within SITUATION begin and end tags. <NAME> Enter the name of the situation to affect. <ACTION> START – Start a known situation request; STOP – Stop an active situation; RECYCLE – Stop and restart an active situation <TYPE> Optional. P – Private situation; E – Enterprise situation. Default: P.

Example:

This request recycles the private situation named Check_DiskSpace_Low:

```
<SITCONTROL>
<SITUATION>
  <NAME>Check_DiskSpace_Low</NAME>
```

```

        <ACTION>RECYCLE</ACTION>
        <TYPE>P</TYPE>
    </SITUATION>
</SITCONTROL>

```

Situation summary

Use the situation summary command to request a listing of the private situations that are running on the agent.

Table 38. Agent Service Interface <SITSUMMARY> request.

Tag	Description
<SITSUMMARY>	Define dynamic threshold override specification.

```

<SITSUMMARY>
</SITSUMMARY>

```

The output from the request looks like the private situation configuration files shown in the “Private situation examples” on page 142.

Dynamic threshold configuration request

Table 39. Agent Service Interface <OVERRIDES> request

Tag	Description
<OVERRIDES>	Define dynamic threshold override specification.

Example:

```

<OVERRIDES>
</OVERRIDES>

```

File

Table 40. Agent Service Interface <TRANSCON> request

Tag	Description
<TRANSCON>	TRANSCON begin and end tags specify that this is a Transport Conduit service request.
<CONTROL>	Specify control action parameter bracketed by begin and end tag.
<INSTRUCTS>	Specify Transport Conduit request instruction. <FILENAME> is the name of the file being requested. <PATH> Optional. Enter the path to the filename if it is not in the current directory. <RECDLM> Optional. Specify the file record delimiter. When a delimiter is defined, the file records are separated by the delimiter, leading and trailing space characters are removed, and the output file is written in record mode. If a delimiter is not defined then all the received file data is reported as one binary data block.
<DATA>	Optional. Data tags bracket download file contents.

Examples:

This example specifies to download the configuration file to the agent:

```

<TRANSCON>
  <CONTROL>PUSH</CONTROL>
  <INSTRUCTS>
    <FILENAME>setenv150.bat</FILENAME>
  </INSTRUCTS>
</TRANSCON>

```

```

    <PATH>%candle_home%\tmaitm6</PATH>
    <RECDLM>$@$</RECDLM>
</INSTRUCTS>
<DATA>
    set path=.;%WINDIR%\system32\npp;%PATH%$@$
    set KFC_DEBUG_API=Y$@$
    set KFC_DEBUG_FILTER=N$@$
    set KFC_DEBUG_STORAGE=N$@$
    set KFC_DEBUG_STORAGE_STAT=N$@$
    set KFC_DEBUG_TIMESYNC=N$@$
    set KFC_TIME_SYNC_REQUIRED=N$@$
    set KFC_API_MEDIASERVER_LISTEN_PORT=12125$@$
    set kbb_ras1=ERROR$@$
    set kbb_ras1_log=.\logs\kfc1.log$@$';
</DATA>
</TRANSCON>

```

The command in this example uploads the trace file from the agent:

```

<TRANSCON>
<CONTROL>PUSH</CONTROL>
<INSTRUCTS>
    <FILENAME>kntcma-%seq%.log</FILENAME>
    <RANGE>3500-3520</RANGE>
    <PATH>%candle_home%\tmaitm6\logs</PATH>;
</INSTRUCTS>
</TRANSCON>

```

Autonomous agent activity log

An agent can run autonomously for an undetermined period of time, taking data samples and saving events. An audit trail log, the agent operations log, enables you to examine and review the agent's activities while it was running autonomously.

When the agent runs autonomously, audit trail records for all events and true sampled application data rows are written to the operations log. The autonomous agent leverages the existing Agent Operation Log facility and outputs audit trail records to it. The Agent Operation Log can be viewed on the Tivoli Enterprise Portal while the agent is online.

- On distributed systems, the agent creates the Operation Log file automatically in the agent installation directory, names it ComputerName_product.LG0 for the current running log file, and renames the previous log file ComputerName_product.LG1 (the backup file).
- On z/OS systems, the agent writes the Agent Operation log records to a SYSOUT class, saving portions of records in memory cache.

The agent operations log also shows the activity of private situations.

The autonomous activity log record contains these fields:

- Agent system name
- Message ID: KRAIRA005
- Global timestamps, showing the actual local time of the event activity
- The message, which shows the situation name, application table name, system name, filter column name, filter value, and actual sampled value or event value. If the situation filter criteria specify several threshold name and value pairs and thus the output exceeds the operation log's record size, then the agent outputs multiple log records.

To obtain an agent autonomous operation activity report, create an Agent Operation Log custom query in the Tivoli Enterprise Portal that filters on message KRAIRA005, then assign the query to a table view in a workspace at the agent level of the Navigator Physical view. Alternatively, you can assign the predefined query named *Agent Operations Log* to a table view and apply a post-filter through the Properties editor Filters tab filters out all rows except those with message KRAIRA005. shows a possible autonomous activity log that might result from such a query.

This is the result of a table view of the Agent Operations Log filtered to include only the agent autonomous messages: `Message == KRAIRA005`

Server Name	Message Number	Global Timestamp	Managed System Type
Primary:East:NT	KRAIRA005	02/16/2009 12:35:42	Situation NT_Process_CPU_Critical for KNT.WTPROCESS reset
Primary:East:NT	KRAIRA005	02/16/2009 12:34:43	Situation NT_Process_CPU_Critical for KNT.WTPROCESS triggered (03) Process_Name [_Total] value <kdsmain>
Primary:East:NT	KRAIRA005	02/16/2009 12:34:42	Situation NT_Process_CPU_Critical for KNT.WTPROCESS triggered (02) Priority_Base [0] value <8>
Primary:East:NT	KRAIRA005	02/16/2009 12:34:42	Situation NT_Process_CPU_Critical for KNT.WTPROCESS triggered (01) %_Processor_Time [65] value <66>
Primary:East:NT	KRAIRA005	02/16/2009 12:34:21	Situation NT_Log_Space_Low for KNT.WTPROCESS triggered %_Usage [95] value <100>
Primary:East:NT	KRAIRA005	02/16/2009 12:32:42	Situation NT_Process_Memory_Critical for KNT>WTPROCESS triggered (02) Working_Set [40000000] value <48832512>
Primary:East:NT	KRAIRA005	02/16/2009 12:32:41	Situation NT_Process_Memory_Critical for KNT>WTPROCESS triggered (01) Process_Name [_Total] value <Rtvscan>
Primary:East:NT	KRAIRA005	02/16/2009 12:31:21	Situation NT_System_CPU_Critical for KNT.WTSYSTEM triggered Operating_System_Version [5.0] value <5.1>
Primary:East:NT	KRAIRA005	02/16/2009 12:29:41	Situation CHECK_NETWORK_STAT for KNT.IPSTATS triggered (06) Datagrams_Received_Header_Errors [0] value <0>
Primary:East:NT	KRAIRA005	02/16/2009 12:29:41	Situation CHECK_NETWORK_STAT for KNT.IPSTATS triggered (05) Datagrams_Outbound_Header_Errors [0] value <0>

Chapter 12. Agent Management Services

Agent Management Services is a strategic approach to Tivoli monitoring agent management that provides these capabilities:

- Monitor the availability of other agents and respond automatically to abnormalities according to user policy.
- An automated method through policy settings and a manual method through take action commands to start, stop, restart, *manage*, and *unmanage* an agent manually.
- Agent management workspaces with views of the information being collected by the Agent Management Services for these base agents: Linux OS, UNIX Logs, UNIX OS, Windows OS, Warehouse Proxy, Warehouse Summarization and Pruning, and the Universal Agent.

Use the Agent Management Services to monitor the availability of agents and respond automatically (such as with a restart) if the agent becomes unhealthy or exits unexpectedly. By using these services, you can see improved agent availability ratings.

The Tivoli Enterprise Portal is the user interface for the services, with predefined take action commands for manually starting or stopping management of an agent by Agent Management Services, and for starting or stopping an agent when it is being managed by the Agent Management Services. AMS Start Agent Instance is intended for agents that are used only occasionally. These take action commands are available from the Agent Management Services workspace pop-up menus and can be referenced in situations for reflex automation.

Note: You can also continue use the familiar methods for starting and stopping an agent, such as through Manage Tivoli Monitoring Services and through the Tivoli Enterprise Portal Navigator pop-up menu.

Features of the Tivoli Agent Management Services

The Agent Management Services relies only on attributes common to all agents (such as file system installation location, file system log file location, and executable name) and APIs common to operating systems (such as enumerating the list of running processes). Using this information, the Agent Management Services improves agent availability and provides a simple, unified interface for the view and control of the agents' availability.

You can bring an agent under Agent Management Services management without making any changes to the agent. As additional agents are added to a system, they can easily be brought under Agent Management Services management.

Component relationships

The Agent Management Services uses three interfaces to communicate with other components in the OS agent process.

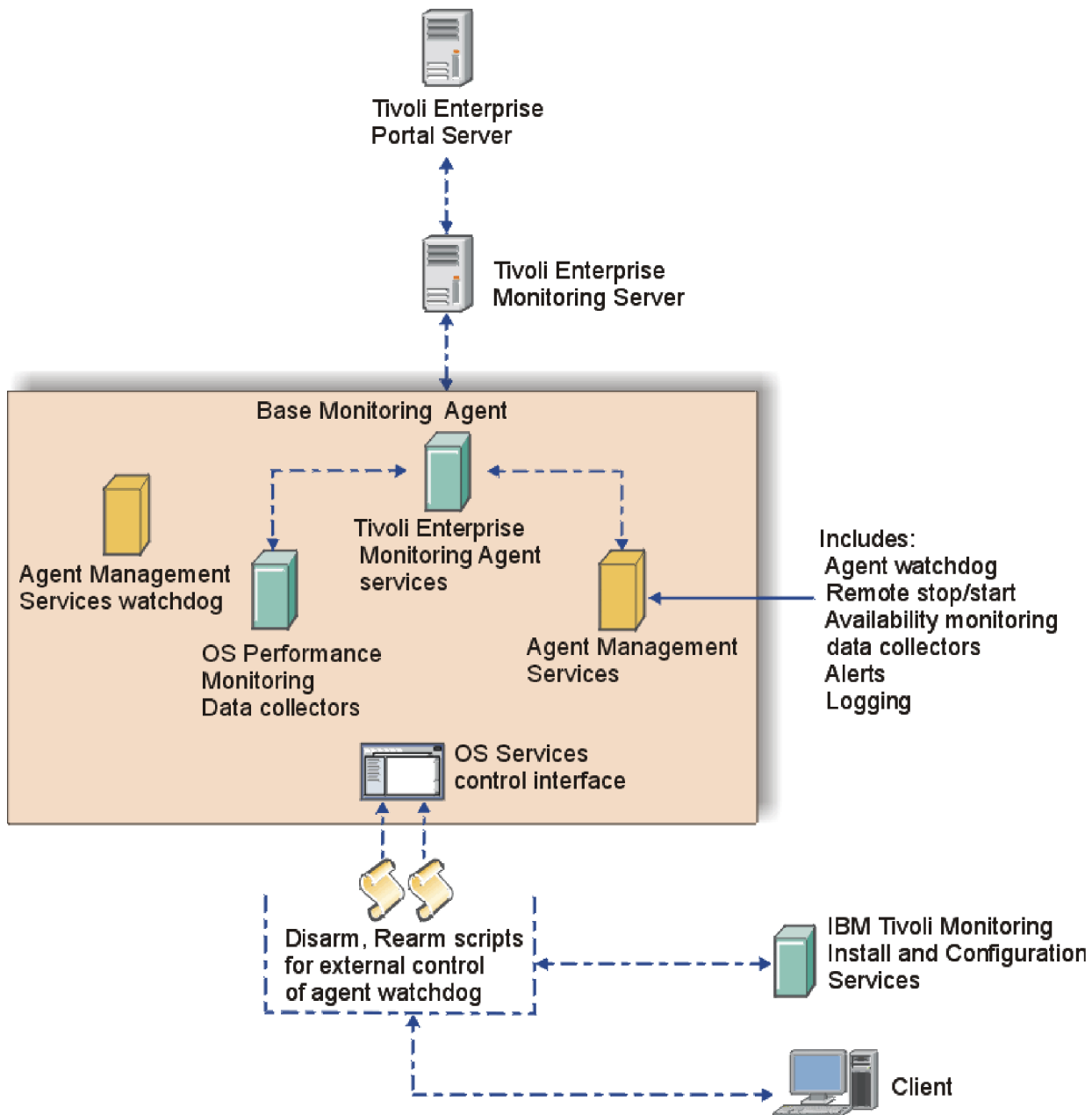


Figure 2. Interactions of Agent Management Services components with IBM Tivoli Monitoring components

Component descriptions

Agent watchdog

The agent watchdog performs specific availability monitoring actions against an agent based on the policy in the agent's *common agent package* (CAP) file. This component runs inside the OS agent process as a logical component. Other than the OS agent itself, the agent watchdog watches any monitoring agent that has an XML file in the CAP directory of the OS agent installation.

Agent Management Services watchdog

Who watches the watchdog? The *Agent Management Services watchdog* is the

same programmatically and behaves the same way as the agent watchdog within the OS agent, but it is used only to watch the OS agent. It is packaged as a stand-alone executable file with the OS agents and runs as process `kcawd` on Linux and operating systems such as UNIX, and as process `kcawd.exe` on Windows.

Installing and configuring Tivoli Agent Management Services

The Agent Management Services is installed automatically with the Linux OS agent, UNIX OS agent, and Windows OS agent, depending on the host platform. Application support files for these agents are also installed on the Tivoli Enterprise Monitoring Server and Tivoli Enterprise Portal Server.

The monitoring behavior of the Agent Management Services towards a particular agent is governed by settings in an XML-based policy files, referred to as a *Common Agent Package* (CAP) file. Every agent that can be managed by AMS installs a CAP file named `<ITM product code>_default.xml` into a directory defined by the `KCA_CAP_DIR` environment variable in the OS monitoring agent configuration file for the relevant platform. On Windows, the location is `<itm_install_dir>\TMAITM6\CAP`; on UNIX and Linux, it is `<itm_install_dir>\config\CAP`.

The CAP file installed by the agent is configured to be read-only and should not be directly modified. If you want to customize the policy settings of this file, create a copy of the file and name it with the convention `<ITM product code>.xml`. Note that because only one executable or “family” of executables is supported per CAP file, all multi-instance monitoring agent instances are governed by the same CAP file.

Table 41. Customizable elements of a common agent package file in Agent Management Services. These settings can be customized in a CAP file.

Name	Data type	Default	Description
checkFrequency	Multiples of 5 (units in seconds)	30	This is length of time between availability checks by Agent Management Services of the managed agent.
cpuThreshold	A positive integer from 0 to 100	no value	Maximum average percent of CPU time that the agent process can consume over a time interval equal to “checkFrequency” seconds before being deemed unhealthy and then restarted by Agent Management Services.
memoryThreshold	A string of characters between 0 and 9. The threshold must include a unit of one of KB, MB, GB, and TB.	no value	Maximum average amount of working set memory that agent process may consume over a time interval equal to “checkFrequency” seconds before being deemed unhealthy and then restarted by Agent Management Services.
managerType	Enumeration: NotManaged or ProxyAgentServices	NotManaged	The entity that performs availability monitoring of the agent.

Table 41. Customizable elements of a common agent package file in Agent Management Services (continued). These settings can be customized in a CAP file.

Name	Data type	Default	Description
maxRestarts	Positive integer.	4	This is the number of times per day an abnormally stopped or unhealthy agent should be restarted. Agents that do not need to be kept running can have a value of 0.

Monitoring the availability of agents

Agent Management Services responds to a stopped or reconfigured agent by restarting it. The Agent Management Services determines that the agent is stopped by querying the operating system for the running application using the value in the *<agentPath>* element of the common agent package (CAP) file.

If the operating system does not show the process in its list of running processes, Agent Management Services knows the process is down and will attempt to restart it using the command or script defined in the *<startScript>* element of the common agent package file. If there is no CAP file, the operating system is checked.

Managed agents that are configured but not started will be automatically started by the watchdog within 10 minutes of being configured. Managed agents whose configured instances are started by the user will be discovered immediately and appear in the Agents' Availability Status view.

If the number of connection attempts to the monitoring server exceeds CTIRA_MAX_RECONNECT_TRIES (default setting is 0), the agent attempts to shut down. If the Agent Management Services Watchdog is running, it immediately restarts the agent. If you want the agent to shut down when CTIRA_MAX_RECONNECT_TRIES is exceeded, this Watchdog process must be disabled. Use the AMS Stop Management action to disable this watchdog process.

Related tasks

Configuring Agent Management Services on autonomous agents

Managing the agent manually

From the *Agent Management Services* workspace for the agent, you can run these Take Action commands to start, stop, manage, and unmanage agents. The action taken will persist until you use the opposing action or start or stop an agent with another method (Tivoli Enterprise Portal, Manage Tivoli Monitoring Services, or at the command line). In the *Agents Management Status* table view, right-click the row of the agent whose status you want to change, then select the action:

AMS Start Agent

Use this action to start an agent that is under the management of the IBM Tivoli Monitoring Agent Management Services. For a multi-instance agent, use **AMS Start Agent Instance**.

AMS Stop Agent

Use this action to stop an agent that is under the management of the IBM Tivoli Monitoring Agent Management Services.

AMS Start Agent Instance

Use this action to start a particular instance of the monitoring agent.

AMS Start Management

Use this action to put an agent under the management of the IBM Tivoli Monitoring Agent Management Services. This is useful when the agent was taken offline intentionally and you are ready to resume running the agent and having it managed.

AMS Stop Management

Use this action to remove an agent from management by the IBM Tivoli Monitoring Agent Management Services. This is useful when you want to take an agent offline and not have it restarted automatically.

For example, to start managing the Universal Agent for Windows (shows in the Agent Management Services workspace, Agent Management Status view as *Unmanaged*), right-click the row and click **Take Action > Select**. Select the AMS Start Management action from the list of possible actions. The command reads, NT:AMS_Start_Manage "Universal Agent for Windows". Click **OK** to start managing the agent. After you click **Refresh**, the Universal Agent status changes to *Managed*.

For further information on each command and Take Action commands in general, see the *IBM Tivoli Monitoring: Windows OS Agent User's Guide*, *IBM Tivoli Monitoring: Linux OS Agent User's Guide*, *IBM Tivoli Monitoring: UNIX OS Agent User's Guide*, and *IBM Tivoli Monitoring: CandleNet Portal User's Guide*.

Chapter 13. Managing historical data

Tivoli Management Services provides tools for collecting data, displaying short-term historical reports, and uploading data to a relational database for long-term storage, summarization, and report generation.

About historical data collection

To make historical data available for reporting and analysis, you need to set up historical data collections. These collections are configured for each attribute group that you want to collect historical data for and distributed to the managed systems that you specify.

Historical data collection

Configuration programs allow you to specify the collection of historical data. The historical data is stored in short-term history files either at the Tivoli Enterprise Monitoring Server or at the monitoring agent. You can choose to specify that historical data to be sent to the Tivoli Data Warehouse database for long-term storage. The data model is the same across the long-term and short-term historical data.

Historical configuration object groups

Part of a historical collection definition is the distribution list, where the managed systems that will save historical data samples are specified. You can add the distribution directly to the historical collection or indirectly through a historical configuration object group.

- Direct distribution involves assigning individual managed systems or managed system groups or both to the historical collection. The advantage of this method is that the distribution applies only to this collection and you can easily add and remove managed systems as needed.
- Indirect distribution involves assigning managed systems or managed system groups or both to the historical configuration group that the historical collection is a member of. The advantage of this method is that you can establish one distribution list and apply it to multiple historical collections simply by adding those collections to the historical group membership.

Use historical configuration groups to combine related historical collections into groups. You can then control collection for the group rather than having to select collection definitions individually. This feature is available when the Tivoli Enterprise Portal Server and Tivoli Enterprise Monitoring Server are at Version 6.2.2 or later.

Warehouse proxy

The agents send the Tivoli Data Warehouse data through the warehouse proxy. The warehouse proxy is a multi-threaded server process that can handle concurrent requests from multiple agents. If the warehouse proxy is not reachable, the agent retries the transmission at the next warehouse interval (next hour or next day, depending on the setting). If, at the next interval, the warehouse proxy does not send back its status during transmission, the transaction is restarted. Then the data is resent to the

warehouse proxy after 15 minutes. If the warehouse proxy sends back a status indicating a failure, the transaction will be restarted at the next warehouse interval.

If you do not intend to save historical data to a data warehouse, you do not need to install and configure the warehouse proxy and the summarization and pruning agent. If the Tivoli Data Warehouse is not used, then it is necessary to use additional programs to trim short-term history files.

Warehouse schema

The Tivoli Data Warehouse has one or more tables for each product, with column names that relate to the data contents. This platform follows a simple data model that is based on the concept of attributes. An attribute is a characteristic of a managed object (node). For example, Disk Name is an attribute for a disk, which is a managed object. Attributes can be single-row or multiple-row. Single-row attributes gather only one set of data, such as the local time attributes because there is only one set of values for local time at any one time. Multiple-row attributes can gather multiple sets of data, such as the Avg_Queue attribute that returns one set of data for each queue that exists on the system. Each attribute belongs to an attribute group, and each attribute item stores data for a particular property of the attribute group.

A table is generated for each attribute group and the table names are used for collection of historical data. The individual monitoring agent user guides contain complete descriptions of the attribute groups specific to that agent.

Warehouse summarization and pruning

The warehouse summarization and pruning agent provides the ability to customize the length of time for which to save data (pruning) and how often to aggregate data (summarization) in the Tivoli Data Warehouse. With summarized data, the performance of queries can be improved dramatically. And with data summarization and data pruning working together, the amount of disk space used can be better managed.

Warehouse summarization is controlled on a per-table basis. How the rows in each table are summarized is determined by a set of attributes in each table that are designated as *primary keys*. There is always one primary key, the ORIGINNODE (often called Server Name or System Name), which means that data is summarized by the managed resource. One or more additional primary keys are provided to further refine the level of summarization for that table. For example, in an OS agent disk table, a primary key might be the logical disk name, which allows historical information be reported for each logical disk in the computer.

Managing your historical data

The Historical Collection Configuration window is available through the Tivoli Enterprise Portal. You can specify the collection and storage of historical data at either the Tivoli Enterprise Monitoring Server or at the remote system where the monitoring agent is installed.

The historical data collection can also be configured using the Command Line Interface **hist** tacmd commands:

```
histConfigureGroups
histcreatecollection
```

histdeletecollection
histeditcollection
histlistcollections
tacmd histviewcollection
histListAttributeGroups
histListProduct
histStartCollection
histStopCollection
histUnconfigureGroups
histViewAttributeGroup

If you have a test environment, you can write scripts that use tacmds for configuring historical data collections and run the script on other test computers or on the production system so that you do not need to repeat the same configuration for each system. For more information on these commands, see *IBM Tivoli Monitoring: Command Reference*.

After your configured historical data collections begin saving data samples, make provisions to manage it. Without additional action, the history data files can grow unchecked, using up valuable disk space.

If you have chosen to upload data through the warehouse proxy to the Tivoli Data Warehouse, then the short-term history files on the monitoring server or monitoring agent are automatically trimmed after upload. If you choose not to use the Tivoli Data Warehouse, then you must institute roll-off jobs to regularly convert and empty out the history data files. Roll-off programs are provided. In addition to trimming the history data files, these scripts produce flat files which can be used with third-party tools to produce trend analysis reports and graphics. There is also an environment variable for setting the maximum size of history files.

The Summarization and Pruning Agent is a mechanism for managing data in the Tivoli Data Warehouse. The data in the warehouse is a historical record of activity and conditions in your enterprise. Summarization of the data is the process of aggregating your historical data into time-based categories, for example, hourly, daily, weekly, and so on. Summarizing data allows you to perform historical analysis of the data over time. Pruning of the data keeps the database to manageable size and thus improves performance. Pruning of the database should be performed at regular intervals.

Important: You can run only one summarization and pruning agent even if you have multiple monitoring servers that are sharing a single Tivoli Data Warehouse database. Running multiple summarization and pruning agents causes conflicts because the multiple instances might attempt to prune the data in the tables simultaneously.

Historical data status and requests

Tivoli Enterprise Portal provides historical reporting capabilities that can be configured using the timespan function. There is information about this in the *Tivoli Enterprise Portal User's Guide*. When a data request is made by the Tivoli Enterprise Portal, data greater than 24 hours old is pulled from the Tivoli Data Warehouse database, and data under 24 hours is pulled from the Tivoli Enterprise Monitoring Server and Tivoli Enterprise Monitoring Agents short-term history files.

This action is transparent to the user; however, requests returning a large amount of data can negatively impact the performance of monitoring servers, monitoring agents, and your network.

Historical collection options

For flexibility in using historical data collection, you can:

- Start history data collection or stop it for the selected attribute groups for a product
- Start history data collection on specific Tivoli Enterprise Monitoring Servers
- Save the history file at the monitoring server or at the agent
- Define how frequently to send data samples to the short-term history file, from once a minute to once a day.
- Define the interval used to save data into the Tivoli Data Warehouse. It can be every 15 minutes, 30 minutes, hour, 12 hours, once a day, or off.
- Define how and when to summarize and prune the data that is stored in the data warehouse.

Historical data collection can be specified for individual monitoring servers, products, and tables. However, all agents of the same type that report directly to the same monitoring server must have the same history collection options. Also, for a given history table, the same history collection options are applied to all monitoring servers for which that history table's collection is currently enabled.

For scalability reasons, collect and store your historical data at the monitoring agent rather than at the monitoring server. However, there are some product and attribute group combinations that are only collected at a specific place, either the monitoring server or the monitoring agent. This is controlled by configuration files installed by the agent.

On OMEGAMON XE products, the persistent data store is used to store short-term history, so must be configured at the collection location. For any given agent, do not vary the collection location: all historical data for the product should be collected either at the TEMA or TEMS. For agents that are configured in the same address space as the monitoring server (required for OMEGAMON XE for z/OS and OMEGAMON XE for Storage on z/OS), configure the persistent data store in the same address space, and specify TEMS as the collection location.

Some agents do not enable collection of history data for all of their attribute groups. This is because the product development team for that agent has determined that collecting history data for certain attribute groups is not appropriate or might have a detrimental effect on performance. This might be because of the vast amount of data that is generated. Therefore, for each product, only tables that are available for history collection are listed in the History Collection Configuration window.

After you configure history data for a table and start history collection, if you still do not see history data for that table, there is a problem either with the agent collection of that data or with the history mechanism.

Performance impact of historical data requests

The impact of historical data collection and warehousing on Tivoli Enterprise Monitoring components is dependent on multiple factors, including collection interval, frequency of roll-off to the data warehouse, number and size of historical tables collected, amount of data, system size, and so on. This section describes some of these factors.

Impact on the Tivoli Enterprise Monitoring Server or the monitoring agent of large amounts of historical data

The default location for storing short-term historical data is at the monitoring agent, although in certain configurations the monitoring server might be preferable. This topic presents factors to consider when determining

- the attribute groups to collect historical data on
- where to save the short-term data files
- how frequently to send historical data samples to the short-term collection location
- whether to warehouse data from the attribute group and, if so, how frequently to send the data from short-term history files to the data warehouse

The collection location can be negatively impacted when large amounts of data are processed. This occurs because the warehousing process on the monitoring server or the monitoring agent must read the large row set from the short-term history files. The data must then be transmitted by the warehouse proxy to the data warehouse. For large datasets, this impacts memory, CPU resources, and, especially when collection is at the monitoring server, disk space.

Because of its ability to handle numerous requests simultaneously, the impact on the monitoring server might not be as great as the impact on the monitoring agent. Nonetheless, when historical collection is at the monitoring server, the history data file for one attribute group can contain data for many agents (all the agents storing their data at the monitoring server) thus making a larger dataset. As well, requests against a large dataset also impact memory and resources at the Tivoli Enterprise Portal Server.

When historical data is stored at the agent, the history file for one attribute group contains data only for that agent and is much smaller than the one stored at the monitoring server. The most recent 24 hours worth of data comes from short-term history files. Beyond 24 hours, the data is retrieved from the Tivoli Data Warehouse. If a query goes to the short-term history file and retrieves a large amount of data, this retrieval can consume a large amount of CPU and memory. You can experience low system performance while a large amount of short-term historical data is being retrieved.

When processing a large data request, the agent might be prevented from processing other requests until the this one has completed. This is important with many monitoring agents because a monitoring agent can typically process only one view query or situation at a time.

Requests for historical data from large tables

Requests for historical data from tables that collect a large amount of data have a negative impact on the performance of the Tivoli Enterprise Monitoring components involved. To reduce the performance impact on your system, set a longer collection interval for tables that collect a large amount of data. You specify

this setting in the History Collection Configuration window. To find out the disk space requirements for tables in your IBM Tivoli Monitoring product, see the specific agent's documentation.

While displaying a query-based view, you can set the Time Span interval to obtain data from previous samplings. Selecting a long time span interval for the report time span adds to the amount of data being processed, and might have a negative impact on performance. The program must dedicate more memory and CPU cycles to process a large volume of report data. In this instance, specify a shorter time span setting, especially for tables that collect a large amount of data.

If a report rowset is too large, the report request can drop the task and return to the Tivoli Enterprise Portal with no rows because the agent took too long to process the request. However, the agent continues to process the report data to completion, and remains blocked, even though the report data is not viewable.

There can also be cases where the historical report data from the z/OS Persistent Data Store might not be available. This can occur because the Persistent Data Store might be not be available while its maintenance job is running.

Scheduling the warehousing of historical data

The same issues with requesting large rowsets for historical reports apply to scheduling the warehousing of historical data only once a day. The more data being collected and stored, the more resources required to read data into memory and to transmit to the data warehouse. If possible, make the warehousing rowset smaller by spreading the warehousing load over each hour, that is, by setting the warehousing interval to one per hour, rather than one day.

Using a data mart to improve long or complex queries

Within the Tivoli Management Services infrastructure, the warehouse proxy regularly inserts new data from the short-term history files into the data warehouse tables. This detailed data is derived by queries from historical views to report this information and can be derived by queries from an external reporting tool. Any active datastore needs to balance read and write activity to maximize performance of the datastore. The data warehouse has periodic write activity balanced with frequent read activity for formatting and creating reports. Under some circumstances (especially formatting reports over long durations or executing complex queries), the database read and write activity can become unbalanced and result in abnormal wait times. Under these circumstances, you can significantly improve performance by adding a secondary datastore, commonly called a *data mart*, for reports from causing long or complex data queries.

Depending upon the reporting requirements, there are two mechanisms that can be used, exploiting the open interfaces delivered with the warehouse:

1. If the complete database is required, use the Database Replication Facilities of the Tivoli Data Warehouse RDBMS.
2. Write and schedule SQL extract scripts, similar to ETL Scripts in TDW V1.x, to extract desired data elements at a scheduled interval from the Tivoli Data Warehouse and populate a reporting database. This reporting database can be optimized for use by an external reporting tool, just like data marts were used in TDW V1.x. These scripts can be SQL Scripts, shell scripts, or PERL scripts.

Sample data mart SQL script for IBM Tivoli Monitoring

The following SQL script is an sample script of how you can create and populate a data mart. Your actual script needs to be revised to reflect your environment.

```

-----
-- Example data mart SQL Script for TDW 2.1
-----
-- This scripts demonstrates the creation and population
-- of a data mart (similar to the data marts in TDW 1.x)
-- starting from the "flat" tables in TDW 2.1.
-- This script can be run using the DB2 UDB CLP:
-- db2 -tvf myscript
-----

-- 1. Create hourly "flat" table from TDW 2.1 (simulated)
-- One row per hour per Windows system
drop table itmuser."Win_System_H";
create table itmuser."Win_System_H" (
    WRITETIME                CHAR( 16 ),
    "Server_Name"            CHAR( 64 ),
    "Operating_System_Type"  CHAR( 16 ),
    "Network_Address"        CHAR( 16 ),
    "MIN_%Total_Privileged_Time"  INTEGER,
    "MAX_%Total_Privileged_Time"  INTEGER,
    "AVG_%Total_Privileged_Time"  INTEGER,
    "MIN_%Total_Processor_Time"   INTEGER,
    "MAX_%Total_User_Time"       INTEGER,
    "AVG_%Total_User_Time"       INTEGER );

-- 2. Insert example data

insert into itmuser."Win_System_H" values (
    '1050917030000000', 'Primary:WinServ1:NT', 'Windows_2000', '8.53.24.170',
    20, 40, 30, 10, 30, 20 );

insert into itmuser."Win_System_H" values (
    '1050917040000000', 'Primary:WinServ1:NT', 'Windows_2000', '8.53.24.170',
    20, 40, 30, 10, 30, 20 );

insert into itmuser."Win_System_H" values (
    '1050917030000000', 'Primary:WinServ2:NT', 'Windows_2000', '8.53.24.171',
    20, 40, 30, 10, 30, 20 );

insert into itmuser."Win_System_H" values (
    '1050917040000000', 'Primary:WinServ2:NT', 'Windows_2000', '8.53.24.171',
    20, 40, 30, 10, 30, 20 );

-- 3. Create a dimension table for the hosts
-- primary key is Server_ID, a generated value
-- alternate key is Server_Name, Network_Address

drop table itmuser."D_Win_System";
create table itmuser."D_Win_System" (
    "Server_ID" INTEGER GENERATED ALWAYS AS IDENTITY
    PRIMARY KEY NOT NULL,
    "Server_Name"            CHAR( 64 ),
    "Operating_System_Type"  CHAR( 16 ),
    "Network_Address"        CHAR( 16 ) );

-- 4. Create an hourly fact table for the System facts
-- Server_ID is a foreign key to D_Win_System

drop table itmuser."F_Win_System_H";
create table itmuser."F_Win_System_H" (
    WRITETIME                CHAR( 16 ) NOT NULL,
    "Server_ID"              INTEGER NOT NULL,
    "MIN_%Total_Privileged_Time"  INTEGER,
    "MAX_%Total_Privileged_Time"  INTEGER,
    "AVG_%Total_Privileged_Time"  INTEGER,
    "MIN_%Total_Processor_Time"   INTEGER,
    "MAX_%Total_User_Time"       INTEGER,
    "AVG_%Total_User_Time"       INTEGER,

```

```

constraint SERVID foreign key ("Server_ID")
references itmuser."D_Win_System" ("Server_ID")
);

-- 5. Insert into the dimension table
-- only insert rows that do not already exist

insert into itmuser."D_Win_System" (
  "Server_Name",
  "Operating_System_Type",
  "Network_Address" )
select
  "Server_Name",
  min("Operating_System_Type") as "Operating_System_Type",
  "Network_Address"
from
  itmuser."Win_System_H" h
where
  not exists ( select 1 from
    itmuser."D_Win_System" d
    where d."Server_Name" = h."Server_Name"
    and d."Network_Address" = h."Network_Address"
  )
group by
  "Server_Name",
  "Network_Address"
;

-- 6. Check values in dimension table
select * from itmuser."D_Win_System"
;

-- 7. Insert into the fact table
-- only insert rows that do not already exist
insert into itmuser."F_Win_System_H"
select
  h.WRITETIME ,
  d."Server_ID" ,
  h."MIN_%_Total_Privileged_Time" ,
  h."MAX_%_Total_Privileged_Time" ,
  h."AVG_%_Total_Privileged_Time" ,
  h."MIN_%_Total_Processor_Time" ,
  h."MAX_%_Total_User_Time" ,
  h."AVG_%_Total_User_Time"
from
  itmuser."Win_System_H" h,
  itmuser."D_Win_System" d
where d."Server_Name" = h."Server_Name"
and d."Network_Address" = h."Network_Address"
and not exists ( select 1 from
  itmuser."F_Win_System_H" f
  where f.WRITETIME = h.WRITETIME
  and f."Server_ID" = d."Server_ID"
)
;

-- 8. Check values in fact table
select * from itmuser."F_Win_System_H"
;

-- 9. Repeat"5. Insert into the dimension table"
-- and "7. Insert into the fact table" on a daily basis

```

See the redbook, *Introduction to Tivoli Enterprise Data Warehouse* at <http://www.redbooks.ibm.com/> for references and additional sample SQL extract scripts.

Planning collection of historical data

Developing a strategy for historical data collection

This section can help you develop a strategy to improve the implementation of your historical data collection.

Factors to consider

When developing a strategy for historical data collection, determine the following factors:

- Whether to warehouse your data. If you plan to use the warehouse proxy, make the following decisions:
 - How often to store, summarize, and prune collected data
 - Whether to collect the data at the Tivoli Enterprise Monitoring Server or at the location where the monitoring agent is running
 - What data to collect
- Whether to schedule automatic or manual data conversion if you want to convert the data to delimited flat files
- Specific requirements for collecting history for Tivoli Monitoring Services for sysplex
- Data collection rules. Consider the following factors, which affect the frequency of historical data collection:
 - Amount of required disk storage for the data to be collected

Note: Data storage capacity requirements are defined in the agents' product guides. More detailed general information is available in the *IBM Tivoli Monitoring: Installation and Setup Guide* in the section about planning a large scale installation of IBM Tivoli Monitoring.

- Where the historical data is stored

Note: History is typically stored at the agent rather than at the Tivoli Enterprise Monitoring Server, so that the data does not have to traverse the network. But if you need to store the data at the monitoring server (for example, to circumvent a firewall), a remote is preferable to the hub. If you have agents connected to the hub, consider changing their connection to a remote monitoring server if you plan to collect historical data at the monitoring server.

- Usage plans for the collected data
- Planned frequency for collecting different attributes

Using the summarization and pruning agent

This topic gives you some planning information for using the summarization and pruning agent. The Tivoli Enterprise Portal enables you to set up summarization and pruning for selected attribute groups in the History Collection Configuration window or from the command line using `tacmd histConfigureGroups` (see *IBM Tivoli Monitoring: Command Reference*). For information about setting up data connections for the warehouse proxy and the summarization and pruning agent, see *IBM Tivoli Monitoring: Installation and Setup Guide*.

Planning to summarize and prune your collected data

If you have installed the Summarization and Pruning agent, the summarization and pruning configuration information is set to default values that are stored in an environment file during the installation of the Summarization and Pruning agent.

History Collection Configuration window

In the History Collection Configuration window in the Tivoli Enterprise Portal, you can configure summarization for each attribute group for which you want to collect data. You decide how to collect the data, how often to collect data, and how often to transfer the data to the Tivoli Data Warehouse.

Configure Summarization and Pruning Agent window

To see the utilization of resources or to determine the hours of peak loads, and so on, you can define a set of hours as *shifts*, for example 9 am to 5 pm.

To specify whether a particular day is a normal work day or a vacation day, you can classify the days that are not normal work days as *vacation days*.

If the Tivoli Data Warehouse and all the agents that are collecting data are not in the same time zone, the *Timezone Indicator* identifies the time zone to use. If you chose to use the Tivoli Data Warehouse time zone, all data is changed to reflect the time zone of the Tivoli Data Warehouse. If you choose the agent's time zone, the data stays unchanged, with the original time zone for that agent.

Summarization tables in the data warehouse

The following are names of the summarization tables. The *x* represents the original table name of the detailed data. The summarization interval that is chosen for the particular attribute group is appended to the original table name. Names can be different between the detailed data table and summarized table name due to database name length restrictions.

Yearly *x_Y*

Quarterly
 x_Q

Monthly
 x_M

Weekly
 x_W

Daily *x_D*

Hourly
 x_H

The table shows the names of the summarization columns in the detailed tables and what they mean. The *x* represents the original column name. The formula values are set by the agents and can be different for different attribute groups. Attribute names can be different between the detailed data table and summarized table due to database name length restrictions.

Table 42. Summarization functions

Name/meaning	Formula
Average	AVG_x
Delta high	HI_x
Delta low	LOW_x
Delta total	TOT_x

Table 42. Summarization functions (continued)

Name/meaning	Formula
Latest (based on the time that the historical data was collected at the Agent)	LAT_x
Maximum	MAX_x
Minimum	MIN_x
Sum	SUM_x

Names can be different between the detailed data table and summarized table name due to database name length restrictions.

Summarization and pruning metrics

The following example describes how the Summarization and Pruning agent calculates metrics that accumulate over time. You can use the results to manage your resources. In this example, the metric represents cache hits since last restart of server.

The total number of cache hits in the last hour is given by the **Total** value. The **Low** value represents the lowest number of cache hits in the hour based on all the detailed data values for the hour. The **High** value represents the highest number of cache hits in the hour based on all the detailed data values for the hour.

With these detailed data values in one hour: 9, 15, 12, 20, 22, delta-based processing has the following rules:

- If the current value is greater than or equal to the previous value, the output equals the previous value minus the current value.
- If the current value is less than the previous value, the output equals the current value.
- Because 15 is greater than 9, the output equals 6.
- Because 12 is less than 15, the output equals 12.
- Because 20 is greater than 12, the output equals 8.
- Because 22 is greater than 20, the output equals 2.
- The TOT_ value is 28, which is the total of outputs.
- The LOW_ value is 2, which is the lowest of outputs.
- The HI_ value is 12, which is the highest of outputs.

Null values in tables and charts of summarized and pruned data

If you see null as the value of a table cell or chart point, it means that no value was stored in the data warehouse. This happens when values that were identified as invalid are reported from a monitoring agent for a given summarization period. The agent support files might have been upgraded or some data cannot be computed on the summarized tables (for instance, counter and delta-based values cannot be calculated if only one value is present).

For example, assume that an invalid value for a particular attribute is -1. If the agent reports -1 for all the collection intervals (1, 5, 15, or 30 minutes; 1 hour; 1 day) up to the point when the summarization and pruning computation is done for a given summarization period (hourly, daily, weekly, monthly, quarterly, or yearly), then there is no data to perform calculations on and a null is written for the given summarization.

Capacity planning suggestions for historical data collection on your Tivoli Data Warehouse

Disk capacity planning is a prediction of the amount of disk space to be consumed for each attribute group whose historical data is being collected. Required disk storage is an important factor to consider when you are defining data collection rules and your strategy for historical data collection.

For more information about performance tuning for your DB2® database, go to the Tivoli Open Process Automation Library (OPAL) and search for part or all of this phrase: Relational database design and performance tuning for DB2 database servers. For more detailed information on capacity planning and scaling of the Tivoli Data Warehouse, see the *IBM Tivoli Monitoring: Installation and Setup Guide*.

Summarization after upgrading agent support

After support for an updated product has been applied to the , it is possible to get a request error message about a missing or unknown column name in the view's status bar after you set a time span with ☒ **Use summarized data** selected.

The resolution is to wait to view the summarized data until after the next scheduled summarization and pruning procedure has taken place. If need be, the summarization and pruning can be rescheduled to run sooner. More information is provided in the *Installation and Setup Guide* and in the Tivoli Enterprise Monitoring Agent User's Guide for your product.

Conversion process for using delimited flat files

If you chose not to warehouse your data, you must convert your collected data to delimited flat files. Data can be scheduled for conversion either manually or automatically. If you choose to continue to convert data to delimited flat files, schedule data conversion to be automatic. Perform data conversion on a regular basis even if you are collecting historical data only to support short-term history displayed in product reports.

If the KHD_TOTAL_HIST_MAXSIZE environment variable is used, the agent will no longer be able to write any historical data to the short term history files once the limit is reached. This variable is a limit for the agents.

Data conversion programs:

The conversion of short-term history files to delimited flat files is done by running a data rolloff program:

```
Linux      UNIX      Windows  krarloff
z/OS      KPDXTRA
```

Columns added to history data files and to meta description files:

Four columns are automatically added to the history data files and to the meta description files. These columns are:

- **TMZDIFF**. The time zone difference from Universal Time (GMT). This value is shown in seconds.
- **WRITETIME**. The CT time stamp when the record was written. This is a 16-character value in the format, where c is the century; yymmdd is the year, month, and day; and hhmmsssttt is hours, minutes, seconds, and milliseconds: cyyymmddhhmmsssttt

- **SAMPLES.** The SAMPLES column increments for every value collected during the same sample and then reset to its starting value again. Rows collected on the same sample have different SAMPLES column values.
- **INTERVAL.** The time between samples, shown in milliseconds.

Note: The warehousing process, using the Tivoli Data Warehouse, only adds two columns (TMZDIFF and WRITETIME), to the Tivoli Data Warehouse database.

Meta description files:

A meta description file describes the format of the data in the source files. Meta description files are generated at the start of the historical data collection process.

The various operating system environments use different file naming conventions. Here are the rules for some operating system environments:

- i5/OS® and HP NonStop Kernel: Description files use the name of the data file as the base. The last character of the name is 'M'. For example, for table QMLHB, the history data file name is QMLHB and the description file name is QMLHBM.
- z/OS: Description records are stored in the PDS facility, along with the data.
- UNIX and Linux: Uses the *.hdr file naming convention.
- Windows: Uses the *.hdr file naming convention.

*Sample *.hdr meta description file:*

```
TMZDIFF(int,0,4) WRITETIME(char,4,16)
QM_APAL.ORIGINNODE(char,20,128) QM_APAL.QMNAME(char,148,48)
QM_APAL.APPLID(char,196,12) QM_APAL.APPLTYPE(int,208,4)
QM_APAL.SDATE_TIME(char,212,16)
QM_APAL.HOST_NAME(char,228,48)
QM_APAL.CNTTRANPGM(int,276,4) QM_APAL.MSGSPUT(int,280,4)
QM_APAL.MSGSREAD(int,284,4) QM_APAL.MSGSBROWSD(int,288,4)
QM_APAL.INSIZEAVG(int,292,4) QM_APAL.OUTSIZEAVG(int,296,4)
QM_APAL.AVGMQTIME(int,300,4) QM_APAL.AVGAPPTIME(int,304,4)
QM_APAL.COUNTOFQS(int,308,4) QM_APAL.AVGMQGTIME(int,312,4)
QM_APAL.AVGMQPTIME(int,316,4) QM_APAL.DEFSTATE(int,320,4)
QM_APAL.INT_TIME(int,324,4) QM_APAL.INT_TIMEC(char,328,8)
QM_APAL.CNTTASKID(int,336,4) SAMPLES(int,340,4)
INTERVAL(int,344,4)
```

For example, an entry can have the form:

```
attribute_name(int,75,20)
```

where *int* identifies the data as an integer, 75 is the byte offset in the data file, and 20 is the length of the field for this attribute in the file.

Estimating space required to hold historical data tables:

The historical data tables for a product are defined in the product's documentation. Refer to the appropriate agent guide for assistance in determining the names of the tables where historical data is stored, their size, and the which are the default tables.

Limiting the growth of short-term history files:

Whether your environment includes a data warehouse or is set up for conversion of short-term history to delimited flat files, it is a good idea to set a maximum size for the history files.

Before you begin

Your operating system user ID must have write permission for this directory.

These agent environment variables are not available on z/OS.

About this task

When your configuration includes data roll-off to the Tivoli Data Warehouse, the size of the short-term history files is controlled by the amount of data being collected, the frequency of collection, and the frequency of roll-off to the data warehouse. Yet, it is possible for the warehouse proxy agent or data warehouse to become unavailable, which means the short-term history files can grow unchecked.

Set the KHD_TOTAL_HIST_MAXSIZE and KHD_HISTSIZE_EVAL_INTERVAL environment variables at every Tivoli Enterprise Monitoring Agent where historical data is collected or at the if data collection occurs there.

Complete these steps to specify a size limit for the directory where short-term history files are saved and how often that this check should take place:

1. Open the environment file for the agent:
 - **Windows** In the Manage Tivoli Monitoring Services window, right-click the component and click **Advanced** → **Edit ENV File**. (These are the `<install_dir>\TMAITM6\K<pc>ENV` files where `<pc>` is the two-character product code, such as `C:\IBM\ITM\TMAITM6\KNTENV`.)
 - **Linux** **UNIX** Change to the `<install_dir>/config` directory and open `<pc>.ini` in a text editor, where `<pc>` is the two-character product code. For example, `/opt/IBM/ITM/config/ux.ini` for the UNIX OS agent.

For a list of product codes see the “IBM Tivoli product codes” appendix of the *IBM Tivoli Monitoring: Installation and Setup Guide*.

2. Add two new lines to the file, where 5 is the maximum number of megabytes that the directory where the short-term history file is located can grow to; and where 900 (15 minutes) is the number of seconds between evaluation of the directory size:

```
KHD_TOTAL_HIST_MAXSIZE =5
KHD_HISTSIZE_EVAL_INTERVAL=900
```

3. Save and close the file.
4. Recycle the component.

Results

After you set a maximum and the directory limit is reached, no new records are written to the short-term history files, which causes gaps to occur in the data collected. However, if the data is warehoused, the warehouse proxy will trim the short term history files to contain only the last 24 hours of data. This can allow the agent to write historical data again; thus, the limit can be reached again and the process repeats. This process can also cause gaps to appear in the data.

What to do when the short-term history file directory size reaches its limit:

When the KHD_TOTAL_HIST_MAXSIZE and KHD_HISTSIZE_EVAL_INTERVAL environment variables have been set for the Tivoli Enterprise Monitoring Agent (or

at the Tivoli Enterprise Monitoring Server if data collection occurs there), no more historical data samples are added to the short-term history files if that maximum directory size has been reached.

You need to resolve the cause of the unchecked short-term history file growth before the saving of data samples to the history files can resume. When data is collected at the agent you can create a custom SQL query or a situation or both that reports when this condition occurs.

Here is an example of a custom SQL query that you can run:

```
SELECT ORIGINNODE, CATEGORY, SEVERITY, TABLE, TIMESTAMP, MESSAGE  
FROM 04SRV.KRAMESG WHERE ORIGINNODE = $NODE$
```

Warehousing your historical data

Several steps are required in order to store your historical data to a supported relational database. Other considerations must also be addressed. This chapter provides guidance on warehousing historical data.

Before you begin

As part of configuring historical data collection for an attribute group, you specify whether the data is stored at the agent or at the Tivoli Enterprise Monitoring Server. In most cases, store history at the agent because the data does not have to traverse the network.

On z/OS, if you store history in the PDS (Persistent Data Store), it can be converted but it must be written to a hard disk. This can use cycles to store and retrieve. Besides steady state cycles, large or multiple PDSs impact Tivoli Enterprise Monitoring Server restart time as they are completely read at startup.

You must have the following tasks completed:

1. Install one of the supported relational databases, DB2, Oracle, or MS SQL Server
2. Create a new relational database instance to store your collected data
3. Install ODBC (Windows) or JDBC drivers for the warehouse proxy
4. Install JDBC drivers for the Summarization and Pruning agent
5. For mainframe products, you must set up historical data collection by defining Persistent Data Store (PDS) datasets. You must also set up the required maintenance tasks to ensure the availability of these datasets. See *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS* for information about maintaining the PDS.

Refer to the *Installation and Setup Guide* for details on the prerequisites needed and how to install them.

Note: Using DB2 V.9 Fix Pack 2 for your data warehouse can cause the warehouse proxy and the summarization and pruning agent not to function properly. Upgrade to DB2 V.9 Fix Pack 3 or later.

Also, review *CandleNet Portal User's Guide*, and the online help in the History Collection Configuration window of the Tivoli Enterprise Portal, for more information about how to collect, summarize, and prune your data.

Configuring your data warehouse

Configure the Tivoli Data Warehouse, historical data collection, and the summarization and pruning for attribute groups.

Configuring the Tivoli Data Warehouse and short-term history

Information about how to install and configure your warehouse proxy is in the *IBM Tivoli Monitoring: Installation and Setup Guide*. Information about how to view short-term report information and workspaces is in the *IBM Tivoli Monitoring: CandleNet Portal User's Guide*.

This section addresses some of the short-term history configurations in relation to your Tivoli Data Warehouse database.

Naming of the Tivoli Data Warehouse history tables and columns

The history tables in the Tivoli Data Warehouse database have the same names as the group names of history tables and columns. For example, Windows NT history for group name NT_System is collected in a short-term file having the name WTSYSTEM. Historical data in this file, WTSYSTEM, is stored to the database in a table named NT_System.

The warehouse proxy uses the complete product attribute name to create DBMS table and column identifiers. This includes any special characters found in an attribute name.

When the length of an attribute name exceeds the maximum table or column name length supported by a DBMS product, the warehouse proxy uses the internal table and column names as defined in the product attribute file.

Tivoli Enterprise Monitoring data dictionary (WAREHOUSEID)

The WAREHOUSEID table resides in the Tivoli Data Warehouse database. It contains records that describe any attribute or table names that exceed the DBMS maximum name length and that have been converted to internal table or column names. You can query this table to find out the correct name for a table or a column that has been internally converted. Each attribute group name in this table has a RECTYPE value of "TAB". Only the TABLENAME and OBJECTNAME values are filled in. Each attribute column name has a RECTYPE value of "COL". All other column values in WAREHOUSEID are filled in.

The WAREHOUSE ID table has these definitions:

RECTYPE CHAR(3)

Indicates the type of record: "TAB" for table; "COL" for column.

TABLENAME CHAR(20)

Indicates an internal table name.

OBJECTNAME CHAR(140)

Indicates an attribute group name.

COLUMNNAME CHAR(20)

Indicates an internal column name.

ATTRNAME CHAR(140)

Indicates an attribute name.

Indexing for data tables

The warehouse proxy automatically creates an associated index for each data table in the Tivoli Data Warehouse database. The index is based on WRITETIME and ORIGINNODE (whose display name can be "Server_Name," "System_Name," and so on, depending on the table) and the TMZDIFF (time zone difference) columns. The index name is the short name of the table, with an "_IDX" suffix.

Use of double quotes to ensure correct access to all data

All Tivoli Data Warehouse table or column names for all major DBMS products are created by surrounding them with the DBMS-supported quoted identifier characters. When referencing historical data in the Tivoli Data Warehouse database, you must use the double-quote character to ensure correct access to that data. Some database products, such as Microsoft SQL Server, do not require the use of double quotes.

User-defined SQL queries or stored procedures

If you created SQL queries or stored procedures prior to IBM Tivoli Monitoring V6.2.1 for use with the previous version of the historical data collection program, these now might need to be modified. The SQL needs to take into account the fact that some relational database products (such as Oracle) require all table and column names to be surrounded by double quotation marks to access IBM history data, some agents might have changed their data characterizations or added new columns.

Warehouse proxy ATTRLIB directory

The ATTRLIB directory in the warehouse proxy is automatically created for you at product installation time. On a Windows system, this directory is located in `<itm_install_dir>\tmaitm6\attrlib`. On an operating system such as UNIX, this directory is located in `<itm_install_dir>/hd/tables`.

During installation, if the warehouse proxy is installed on the same computer where other agents are installed, the agent product attribute files that are accessible to the installation program are added to the ATTRLIB directory. The warehouse proxy uses the attribute file in only one specific condition: when the monitoring agent version is earlier than version 6.1.0.

The attribute file allows determination of the table or column internal name when the length of an attribute name exceeds the maximum table or column name length that a warehouse DBMS product supports. In that condition only, the attribute file must be in the ATTRLIB directory. If the warehouse proxy is installed on a separate computer and you have a monitoring agent that is not at the latest level, you must copy the attribute file of that agent to the ATTRLIB directory where the warehouse proxy is installed.

If you see an error message stating that an export failed because a particular product attribute file was missing from this directory, locate the missing product attribute file and copy it into the ATTRLIB directory.

Changes in the set of collected attributes

When changes are detected in the set of collected attributes, such as when a new version of an agent with added attributes is deployed, the historical program performs these functions:

1. If warehousing is specified in the current historical data collection request, all collected historical data for the table is exported to the data warehouse. If the warehousing operation is successful, all short-term history data and meta files are deleted.

If the operation fails (for example, if the warehouse proxy is not available), the short-term historical data and meta files are renamed. On the z/OS operating system environment, if a generic table is used to store the data, the short-term historical data for a table are deleted regardless of whether the warehousing operation is successful or not.

- Windows and UNIX operating system environments

On these operating system environments, the history data and meta files are renamed with the **.prv** and **.prvhdr** suffixes respectively.

- i5/OS operating system environment

On this operating system environment, the history data and meta files are renamed with the **P** and **Q** suffixes respectively.

If the renamed files already exist, they are deleted prior to the renaming operation (that is, only one generation of changed short-term history files is kept).

2. If warehousing is NOT specified in the current historical data collection request, the history data and meta file are renamed as described above. On z/OS, if a generic table is used to store the data, all short-term history data for a table together with its meta record are deleted.

Logging successful exports of historical data

Every successful export of historical data is logged in the Tivoli Data Warehouse in a table called WAREHOUSELOG. The WAREHOUSELOG contains information such as origin node, table to which the export occurred, number of rows exported, time the export took place, and so forth. You can query this table to learn about the status of your exported history data.

Furthermore, the WAREHOUSELOG table contains error messages that explain the reasons that export failed.

Error logging for stored data

About this task

Viewing errors in the Event Log

When the warehouse proxy is installed on Windows, if an error occurs during data roll-off, one or more entries are inserted into the Windows Application Event Log that is created. To view the Application Event Log, start the Event Viewer by clicking **Start → Programs → Administrative Tools → Event Viewer**. Select **Application** from the Log menu.

When the warehouse proxy is installed on an operating system such as UNIX, the errors can be seen in the `<itm_install_dir>/logs/*hd*.log` file.

On either platform, errors can also be seen in the WAREHOUSELOG table in the Tivoli Data Warehouse database.

Setting a trace option

You can set error tracing on to capture additional error messages that can be helpful in detecting problems. To activate the trace option:

1. In Manage Tivoli Monitoring Services, right-click **Warehouse Proxy** and select **Advanced Edit Trace Params**.
2. Select the RAS1 filters. The default setting is **ERROR**.
3. Accept the defaults for the rest of the fields.
4. Click **Yes** to recycle the service.

Viewing the trace log

To view the trace log containing the error messages:

1. In Manage Tivoli Monitoring Services, right-click **Warehouse Proxy** and select **Advanced → View Trace Log**. The Log Viewer window displays a list of log files for the warehouse proxy.
2. Select the appropriate log file in Select Log File. All logs are listed in this window, ordered by most recent file.
3. Click **OK**.

Summarization and pruning configuration

After installation of Tivoli Management Services is complete, one of the initial setup tasks is to configure the summarization and pruning agent for general behavior, such as the summarization and pruning schedule and frequency. As well, you need to specify summarization and pruning for the attribute groups that historical data is being collected for in your monitored application.

Summarized and pruned data availability:

The first time the summarization and pruning tool is run, you might not get the results you expect. Review the installation and configuration tasks that need to take place before you can expect to the data from the Tivoli Data Warehouse summarized and pruned.

The summarization and pruning procedure is dependent on having enough data in the data warehouse to work with, how the data collection and warehousing intervals are set, and whether the summarization and pruning specifications were set in the History Configuration Collection window. These installation and configuration tasks must be completed before summarized and pruned data is available from the warehouse:

1. Install the monitoring agent and add application support for it on the monitoring server and the portal server
2. Configure historical data collection for one or more attribute groups for the agent.
3. Distribute the historical collection to managed systems to start collecting data.
4. For each attribute group that has historical data collection taking place, configure the summarization and pruning intervals.
5. Wait for at least one warehouse interval. Check to make sure data is in the warehouse in the detailed tables. It is not sufficient to query historical data from the portal client because the first 24 hours comes from the short-term history files and not the data warehouse.
6. Configure the summarization and pruning agent, making sure that the test connection to the database works and that you schedule when the agent should perform work. You can configure the agent earlier, but wait for the scheduled run to complete before expecting the warehoused data to be summarized and pruned.

After the scheduled run time, you should have summary data in the warehouse.

Configuring summarization and pruning:

Configure summarization and pruning for the attribute groups that you have configured historical data collections for in order to aggregate data and keep the keep the Tivoli Data Warehouse size at a manageable level.

Results

The next time summarization and pruning takes place, the summarization and pruning agent applies the configuration to the long-term data stored in the data warehouse.

Changing global configuration settings:

Use the Configure Summarization and Pruning Agent window to change system-wide configuration settings for data summarization, pruning, or collection.

About this task

Complete these steps to edit the summarization and pruning agent configuration:

1. In Manage Tivoli Monitoring Services, right-click Summarization and Pruning agent.
2. Click on **Reconfigure**.
3. Click **OK** in the Warehouse Summarization and Pruning Agent: Advanced Configuration window.
4. Click **OK** in the next window.
5. Click **Yes** in the Warehouse Summarization and Pruning Agent window to configure the Summarization and Pruning Agent.
6. Enter the Tivoli Data Warehouse database and Tivoli Enterprise Portal
 - a. In the **JDBC drivers** field, click **Add** to invoke the file browser window to select your JDBC driver. Click **OK** to close the browser and add the JDBC drivers to the list. You can also highlight an entry in the JDBC drivers list and click **Delete** to remove a driver. This gives you the ability to collect JDBC drivers to communicate with your Tivoli Data Warehouse database. JDBC drivers are installed separately and each database provides a set of these JDBC drivers.

Note:

- If your Tivoli Data Warehouse database is on an operating system such as UNIX, find the directory where DB2 is installed and, in the jdbc drivers directory, select the db2jcc.jar and db2jcc_license_cu.jar files. For example, `<db2_installdir>/java/db2jcc.jar` and `<db2_installdir>/java/db2jcc_license_cu.jar`.
 - If your Tivoli Data Warehouse database is on MS SQL Server 2000 or 2005, install the MS SQL Server 2005 JDBC driver from the Microsoft SQL Server Web site. You will need the sqljdbc.jar file; see the installation instructions for your operating systems from Microsoft to locate the file.
 - If your Tivoli Data Warehouse database uses Oracle, use the ojdbc14.jar file. The location on Windows is `%ORACLE_HOME%\jdbc\lib`; the location on operating systems such as UNIX is `$ORACLE_HOME/jdbc/lib`.
- b. In the drop down list, select the type of database for your Tivoli Data Warehouse.
 - c. If not correct, enter the Tivoli Data Warehouse URL, Driver, Schema, User ID and password.

Important: During the configuration of the warehouse proxy, a database user (called ITMUser by default) is created. The User ID that you enter here must match that database user.

- d. Click **Test database connection** to ensure you can communicate with your Tivoli Data Warehouse database.
 - e. Enter the Tivoli Enterprise Portal Server Host and Port, if you do not want to use the defaults. The **TEP Server Port** field is numeric only.
7. Select the scheduling information in the **Scheduling** tab:
- a. Schedule the agent to run every x days.
 - b. Select the time of the day that you want the summarization to run.

The default is to run every day at 2:00 AM.

8. Specify the Shift Information and Vacation Settings in the **Work Days** tab:
- a. Select day in the **Week starts on**.
 - b. If you want to specify shifts, select **Specify shifts**. The default settings for this field are listed in the **Peak Shift Hours** box on the right side of the window. You can change these settings by selecting the hours you want in the **Off Peak Shift Hours** box and clicking the right arrow button to add them to the **Peak Shift Hours** box.



Note: Changing the shift information after data has been summarized can create an inconsistency in the data. Previous data collected and summarized can not be recalculated with the new shift values.

- c. If you want to change your vacation settings, select **Specify vacation days**. Click **Yes** or **No** to specify weekends as vacation days. Select **Add** to open a calendar, then select the vacation days to add.

Note: On Linux and operating systems such as UNIX, right-click to select the month and year.

The days selected display in the box below the **Select vacation days** field. If you want to delete any days you have previously chosen, select them and click **Delete**.

9. Select the desired options In the **Log Parameters** tab. This tab defines the parameters for pruning the log tables populated by the warehouse proxy and the summarization and pruning agent.
- a. Select ☐ **Keep WAREHOUSEAGGREGLOG data for** to prune the WAREHOUSEAGGREGLOG table, which is populated by the summarization and pruning agent. After enabling this option, specify the number of days, months, or years to keep data in the table. Data older than the specified time interval will be deleted by the summarization and warehouse pruning agent.
 - b. Select ☐ **Keep WAREHOUSELOG data for** to prune the WAREHOUSELOG table, which is populated by the warehouse proxy. After enabling this option, specify the number of days, months, or years, to keep the data in the table. Data older than the specified time interval will be deleted by the summarization and pruning agent.
 - c.
10. In the **Additional Parameters** tab select these options:
- a. Specify the maximum rows that can be deleted in a single database transaction. The values are 1 through n. The default is 1000.
 - b. Specify the age of the data that you want summarized in the **Summarize hourly data older than** and **Summarize daily data older than** fields. Values are 0 through n. The default is 1 for hourly data and 0 for daily data.

- c. Choose the time zone you want to use from the **Use timezone offset from** drop down list. If the Tivoli Data Warehouse and agents that are collecting data are all not in the same time zone, and all the data is stored in the same database, use this option to identify the time zone you want to use.
 - d. Specify the number of concurrent execution threads that will be used when the summarization and pruning agent is processing data in the **Number of Worker Threads**. The recommended value is twice the number of CPUs. More threads might allow the summarization and pruning agent to finish faster, but will use more resources on the machine that is running the summarization and pruning agent and will use more database resources such as connections and transaction log space.
 - e. The summarization and pruning caches the most recent errors that have occurred in memory. This information is provided in an attribute group and can be viewed in workspaces that are provided with the summarization and pruning agent. The **Maximum number of node errors to display** setting specifies the maximum number of errors to store in memory. Only the most recent errors are kept. Once the limit is reached, the oldest errors are dropped.
 - f. The summarization and pruning caches information about the most recent runs that were performed. This information is provided in an attribute group and can be viewed in workspaces that are provided with the summarization and pruning agent. The **Maximum number of summarization and pruning runs to display** setting specifies the maximum number of runs to store in memory. Only the most recent runs are kept. Once the limit is reached, the oldest runs are dropped.
 - g. The summarization and pruning agent periodically checks that it can communicate with the data warehouse database. The **database connectivity cache time** setting determines how often to perform this check.
 - h. To improve performance, the summarization and pruning agent batches updates to the data warehouse database. The **Batch mode** parameter specifies how the batching will be performed. The two options are **single managed system** and **multiple managed systems**.
11.   Click any of these buttons: **Save** after you have all your settings correct; **Reload** to reload the original values; or **Cancel**, at any time, to cancel out of the Configure Summarization and Pruning Agent window.

How to disable the Summarization and Pruning agent: About this task

If you want to disable summarization and pruning for your entire enterprise:

1. In Manage Tivoli Monitoring Services, right-click the Summarization and Pruning agent in the Service/Application column.
2. Select **Stop**.

If you want to turn off summarization and pruning for a particular product or set of attribute groups in the History Collection Configuration window:

1. In the Tivoli Enterprise Portal, click the History Collection Configuration window button that is located on the toolbar.
2. Select the **Product**.
3. Select one or more **Attribute Groups**.
4. Click the **Unconfigure Groups** button.

Collecting Agent Operations Log history

The Agent Operations Log collects the messages occurring on the distributed agents in your enterprise. This log is part of the Tivoli Management Services agent framework. On Windows, if your historical data collection configuration includes the Agent Operations Log attribute group (OPLOG table), you need to create directories for the historical data and edit each agent configuration file.

Before you begin

You need to manually create history data directories for all agents that are collecting historical data on the same computer and then edit each agent configuration file on the same computer to specify the new path for short-term data collection. This is necessary on Windows because all agent logs by default are stored in the same `<install_dir>\tmaitm6\logs\` directory and each agent creates an agent operations log file named OPLOG to store short term history data. Thus, the same OPLOG history file is being shared by all the agents; if more than one agent process attempts to warehouse history data from the same short term history binary file, the same data could get transferred to the Tivoli Data Warehouse more than once.

For example, the Windows OS and Active Directory monitoring agents are installed. Each process will create and store its operations log history data in a file named `C:\IBM\ITM\TMAITM6\logs\OPLOG`. Now there are at least two processes attempting to share the same history data file. The data from multiple agents can be written to the same file, but the warehouse upload process will encounter problems with this setup. One agent process is not aware that, at any given time, another agent process might be performing the same warehouse data upload from the same short-term history file. This can lead to duplicate history data transferred to the warehouse database.

About this task

For each agent that collects historical data on the Windows system, complete these steps:


1. Create a history child directory of `<itm_install_dir>\tmaitm6\logs\`.
2. Create a `k??` child directory of `<itm_install_dir>\tmaitm6\logs\history` where `??` is the two-character product code. For example, `c:\ibm\itm\tmaitm6\logs\history\k3z` would be the path to *IBM Tivoli Monitoring Agent for Active Directory* short-term history files. The system user ID for this agent must have read and write permission for this directory.
3. Open the `<itm_install_dir>\tmaitm6\k??cma.ini` agent configuration file (where `??` is the two-character product code) in a text editor. See your monitoring product user's guide for the name of the file used for agent configuration.
4. Locate the `CTIRA_HIST_DIR=@LogPath@` parameter and append with `\history\k??` (where `??` is the two-character product code). For example, `CTIRA_HIST_DIR=@LogPath@\history\knt` specifies `c:\ibm\itm\tmaitm6\logs\history\knt` for Windows OS agent historical data collection on this computer.
5. Save the `k??cma.ini` configuration file.
6. Copy the `<install_dir>\tmaitm6\logs\khdexp.cfg` warehouse upload status file to the `\history\k??` directory. If this file is not copied to the new agent

history directory, your existing history data might be warehoused more than once. It is possible that this file does not exist if the history warehousing option has never been enabled.

7. Copy any .hdr files and their base name counterparts (no file extension) for the agent to the new location. For example, the c:\ibm\itm\tmaitm6\logs\history\knt directory might look like this:

```
khdexp.cfg
netwrkin
netwrkin.hdr
ntprocssr
ntprocssr.hdr
wtlogcldsk
wtlogcldsk.hdr
wtmemory
wtmemory.hdr
wtphysdsk
wtphysdsk.hdr
wtserver
wtserver.hdr
wtsystem
wtsystem.hdr
```

Please note that you might be copying history data files from the tmaitm6\logs directory that are not managed by the target agent. For example, the directory might contain Oracle database history data, but you are copying the files to the new Windows OS agent history directory. The copied files that are not used by the Windows OS agent will not be needed and can safely be deleted.

8. In Manage Tivoli Monitoring Services, right-click the monitoring agent service and click **Reconfigure**, click **OK** twice to accept the settings in the configuration windows, then  **Start** the agent.

Converting short-term history files to delimited flat files

If you selected the option to store data to a data base, that option is mutually exclusive with running the file conversion programs described in this section. To use these conversion procedures, you must have specified **Off** for the Warehouse option in the History Collection Configuration window of the Tivoli Enterprise Portal.

The conversion procedure empties the history accumulation files and must be performed periodically so that the history files do not take up needless amounts of disk space.

Converting history files to delimited flat files on Windows systems

The history files collected using the rules established in the historical data collection configuration program can be converted to delimited flat files for use in a variety of popular applications to easily manipulate the data and create reports and graphs. Use the Windows **AT** command to schedule file conversion automatically. Use the krarloff rolloff program to manually invoke file conversion. For best results, schedule conversion to run every day.

Conversion process using the AT command

When setting up the process that converts the history files you have collected to delimited flat files, schedule the process automatically the Windows **AT** command,

or manually by running the krarloff rolloff program. History file conversion can occur whether or not the Tivoli Enterprise Monitoring Server or the agent is running.

Note: Run history file conversion every 24 hours.

Archiving procedure using the Windows AT command:

To archive historical data files on Tivoli Enterprise Monitoring Servers and on remote managed systems using the **AT** command, use the procedure that follows. To find out the format of the command, enter **AT /?** at the MS/DOS command prompt.

1. For the AT command to function, you must start the Task Scheduler service. To start the Task Scheduler service, select **Settings >Control Panel > Administrative Tools > Services**.

Result: The Services window displays.

2. At the Services window, select **Task Scheduler**. Change the service Start Type to Automatic. Click **Start**.

Result: The Task Scheduler service is started.

An example of using the AT command to archive the history files is as follows:

```
AT 23:30 /every:M,T,W,Th,F,S,Su c:\sentinel\cms\archive.bat
```

In this example, Windows runs the archive.bat file located in c:\sentinel\cms everyday at 11:30 pm. An example of the contents of archive.bat is:

```
krarloff -o memory.txt wtmemory
krarloff -o physdsk.txt wtphysdsk
krarloff -o process.txt wtprocess
krarloff -o system.txt wtsystem
```

Location of the Windows executables and historical data collection table files:

This section discusses the location of Windows programs needed for converting historical data.

The programs are in these locations:

- `<itm_install_dir>\cms` directory on the Tivoli Enterprise Monitoring Server.
- `<itm_install_dir>\tmaitm6` directory on the remote managed systems where the agents were installed.

If your agent history data has been configured to be stored at the agent computer and you want to store your history files on a disk that provides more storage capacity than the default history data file location provides, this location can be overridden using the existing environment variable `CTIRA_HIST_DIR` for your agent. This can not be done when history data is stored at the Tivoli Enterprise Monitoring Server.

If you have multiple instances of the same agent running on the same Windows system, the installer creates a separate directory for the process history files stored at the agent. The default location for agents running on the Windows operating system is `C:\IBM\ITM\TMAITM6\LOGS`. New directories are created under the `TMAITM6\LOGS` directory: `History\<3 character component code>(KUM, KUD, and so on)\<specified multi-process instance name>`.

For example, if you configure a second instance of the DB2 Monitoring agent called `UDBINST1` on the same Windows system, a directory called

C:\IBM\ITM\TMAITM6\LOGS\History\KUD\UDBINST1 is created to store the history data. This instance of the DB2 agent environment variable CTIRA_HIST_DIR is set to this value.

Location of Windows historical data table files:

The krarloff rolloff program needs to know the location of these files.

If you run the monitoring server and agents as processes or as services, the historical data table files are located in the:

- `<itm_install_dir>\cms` directory on the monitoring server
- `<itm_installdir>\tmaitm6\logs` directory on the managed systems

Converting files using the krarloff rolloff program

Overview:

You can also manually initiate the krarloff rolloff program as described in this section.

The krarloff rolloff program can be run either at the Tivoli Enterprise Monitoring Server or in the directory in which the monitoring agent is running, from the directory in which the history files are stored.

Note: The krarloff program will not output values that contain UTF8 data. All attributes that might contain UTF8 data will output the value BLANK.

Attributes formatting:

Some attributes need to be formatted for display purposes. For example, floating point numbers that specify a certain number of precision digits to be printed to the left of a decimal point. These display formatting considerations are specified in product attribute files.

When you use the krarloff rolloff program to roll off historical data into a text file, any attributes that require format specifiers as indicated in the attribute file are ignored. Only the raw number is seen in the rolled off history text file. Thus, instead of displaying 45.99% or 45.99, the number 4599 displays.

The warehouse proxy inserts data according to the type, length, and data precision specified in the attribute files. However, the Tivoli Data Warehouse database displays the correct attribute formatting *only* for those attributes that use integers with floating point number formats.

Using the krarloff rolloff program on a Windows system:

Run the krarloff rolloff program from the directory in which the Tivoli Enterprise Monitoring Server or the monitoring agent is run by entering the following at the command prompt:

```
krarloff [-h] [-d delimiter] [-g] [-m metafile] [-r rename-to-file]
[-o output-file] {-s source | source-file name}
```

where the square brackets denote the optional parameters, and the curly braces denote a required parameter.

Note: The command is on a single line when typed.

Krarloff rolloff program parameters:

Table 43. krarloff rolloff program parameters

Parameter	Default Value	Description
-h	off	Controls the presence or absence of the header in the output file. If present, the header is printed as the first line. The header identifies the attribute column name.
-d	tab	Delimiter used to separate fields in the output text file. Valid values are any single character (for example, a comma).
-g	off	Controls the presence or absence of the product group_name in the header of the output file. Add the -g to the invocation line for the krarloff rolloff program to include a group_name.attribute_name in the header.
-m	source-file.hdr	metafile that describes the format of the data in the source file. If no metafile is specified on the command-line, the default file name is used.
-r	source-file.old	Rename-to-filename parameter used to rename the source file. If the renaming operation fails, the script waits two seconds and retries the operation.
-o	source-file. <i>nnn</i> where <i>nnn</i> is Julian day	Output file name. The name of the file containing the output text file.
-s	none	Required parameter. Source short-term history file that contains the data that needs to be read. Within the curly brace, the vertical bar () denotes that you can either use an -s source option, or if a name with no option is specified, it is considered a source file name. No defaults are assumed for the source file.

Converting history files to delimited flat files on an i5/OS system

The history files collected using the rules established in the historical data collection configuration program can be converted to delimited flat files for use in a variety of popular applications to easily manipulate the data and create reports and graphs. Use the krarloff rolloff program to manually invoke file conversion.

Note: Run history file conversion every 24 hours.

Storing the historical data stored on an i5/OS system

User data is stored in the IFS directory set for the configuration variable *CTIRA_HIST_DIR*. The default value for this variable is *is/qibm/userdata/ibm/itm/hist*. For each table, there are two files stored on the i5/OS system that are associated with historical data collection.

For example, if you are collecting data for the system status attributes, these two files are:

- KA4SYSTS: This is the short-term data that is displayed as output by the i5/OS agent.
- KA4SYSTS.hdr: This is the metafile. The metafile contains a single row of column names.

The contents of both files can be displayed using `WRKLNK /qibm/userdata/ibm/itm.hist` command.

Conversion process on an i5/OS system

The `krarloff` rolloff program can be run either at the Tivoli Enterprise Monitoring Server or in the directory in which the monitoring agent is running from the directory in which the history files are stored.

Run the `krarloff` rolloff program by entering the following at the command prompt:

```
call qautomon/krarloff parm ([ ' -h' ] [ '-g' ] [ '-d' 'delimiter' ] [ '-m' metafile ]  
[ '-r' rename-source-file-to ] [ '-o' output-file ] { '-s' source-file | source-file } }
```

where the square brackets denote the optional parameters, and the curly braces denote a required parameter.

If you run the `krarloff` rolloff program from an i5/OS system in the directory in which the agent is running, replace `qautomon` with the name of the executable for your agent. For example, the MQ agent uses `kmqlib` in the command string.

Note: Enter the command on a single line.

After running the `krarloff` rolloff program

In using the system status example above, after running the `krarloff` rolloff program, file `KA4SYSTS` becomes `KA4SYSTSO`. A new `KA4SYSTS` file is generated when another row of data is available.

`KA4SYSTSM` remains untouched.

`KA4SYSTSH` is the file that is displayed as output by the `krarloff` rolloff program and that contains the data in delimited flat file format. This file can be transferred from the i5/OS to the workstation by means of a file transfer program (FTP).

Converting history files to delimited flat files on UNIX Systems

This topic explains how the UNIX **`itmcmd history`** script is used to convert the saved historical data contained in the history data files to delimited flat files. You can use the delimited flat files in a variety of popular applications to easily manipulate the data to create reports and graphs.

Understanding history data conversion

In the UNIX environment, you use the **`itmcmd history`** script to activate and customize the conversion procedure used to turn selected Tivoli Enterprise Monitoring short-term historical data tables into a form usable by other software products. The historical data that is collected is in a binary format and must be converted to ASCII to be used by third party products. Each short-term file is converted independently. The historical data collected by the Tivoli Enterprise Monitoring Server can be at the host location of the Tivoli Enterprise Monitoring Server or at the location of the reporting agent. Conversion can be run at any time, whether or not the Tivoli Enterprise Monitoring Server or agents are active.

Conversion applies to all history data collected under the current *install_dir* associated with a single Tivoli Enterprise Monitoring Server, whether the data was written by the Tivoli Enterprise Monitoring Server or by a monitoring agent.

See *IBM Tivoli Monitoring: Command Reference* for additional information about **itmcmd history**.

When you enter:

```
itmcmd history -h
```

at the command-line, this output displays:

```
itmcmd history [ -h install_dir ] -C [ -L nnn[Kb|Mb] ] [ -t masks*,etc ]  
[ -D delim ] [ -H|+H ] [ -N n ] [ -p cms_name ]  
prod_code itmcmd history -A?itmcmd history [ -h install_dir ]  
-A perday|0 [ -W days ] [ -L nnn[Kb|Mb] ] [ -t masks*,etc ]  
[ -D delim ] [ -H|+H ] [ -N n ]  
[ -i instance|-p cms_name ] prod_code
```

Note: Certain parameters are required. Items separated with a | vertical bar denotes mutual exclusivity (for example, Kb|Mb means enter either Kb or Mb, not both.) Typically, parameters are entered on a single line at the UNIX command prompt.

See the *Command Reference* for all of the parameters used with this command.

Performing the history data conversion

The **itmcmd history** script schedules the conversion of historical data to delimited flat files. Both the manual process to perform a one-time conversion and the conversion script that permits you to schedule automatic conversions are described here. See the *IBM Tivoli Monitoring: Command Reference* for a complete description of the syntax and options.

After the conversion has taken place, the resulting delimited flat file has the same name as the input history file with an extension that is a single numerical digit. For example, if the input history file table name is KOSTABLE, the converted file is named KOSTABLE.0. The next conversion is named KOSTABLE.1, and so on.

Performing a one-time conversion:

To perform a one-time conversion process, change to the *<itm_install_dir>/bin* and enter the following at the command prompt:

```
./itmcmd history -C prod_code
```

Scheduling basic automatic history conversions:

Use **itmcmd history** to schedule automatic conversions by the UNIX *cron* facility. To schedule a basic automatic conversion, enter the following at the command prompt:

```
./itmcmd history -A n prod_code
```

where *n* is a number from 1-24. This number specifies the number of times per day the data conversion program runs, rounded up to the nearest divisor of 24. The product code is also required.

For example, the following command means to run history conversion every three hours:

itmcmd history -A 7 ux

Customizing your history conversion:

You can use the **itmcmd history** script to further customize your history collection by specifying additional options. For example, you can choose to convert files that are above a particular size limit that you have set. You can also choose to perform the history conversion on particular days of the week.

See the *Command Reference* for a description of all of the history conversion parameters.

Converting history files to delimited flat files on HP NonStop Kernel Systems

If you selected the option to collect and store data to a data warehouse, that option is mutually exclusive with running the file conversion programs described in this chapter. To use these conversion procedures, you must have specified **Off** for the **Warehouse** option on the History Collection Configuration window of the Tivoli Enterprise Portal.

The history files collected using the rules established in the History Configuration program can be converted to delimited flat files for use in a variety of popular applications to easily manipulate the data and create reports and graphs. Use the **krarloff** rolloff program to manually invoke file conversion. For best results, schedule conversion to run every day.

Support is provided for IBM Tivoli Monitoring for WebSphere MQ Configuration and for IBM Tivoli Monitoring for WebSphere MQ Monitoring running on the HP NonStop Kernel operating system (formerly Tandem). For information specific to IBM Tivoli Monitoring for WebSphere MQ Monitoring relating to historical data collection, see the Customizing Monitoring Options topic found in your version of the product documentation.

Conversion process on HP NonStop Kernel Systems

When setting up the process that converts the history files you have collected to delimited flat files, schedule the process manually by running the **krarloff** rolloff program. Run history file conversion every 24 hours.

Using the **krarloff** rolloff program on HP NonStop Kernel:

The history files are kept on the DATA subvolume, under the default **<\$VOL>.CCMQDAT**. However, the location of the history files is dependent on where you start the monitoring agent. If you started the monitoring agent using **STRMQA** from the **CCMQDAT** subvolume, the files are stored on **CCMQDAT**.

You can run the **krarloff** rolloff program from the DATA subvolume by entering the following:

RUN <\$VOL>.CCMQEXE.KRARLOFF <parameters>

Note that **CCMQDAT** and **CCMQEXE** are defaults. During the installation process, you can assign your own names for these files.

Attribute formatting:

Some attributes need to be formatted for display purposes. For example, floating point numbers that specify a certain number of precision digits to be printed to the left of a decimal point. These display formatting considerations are specified in product attribute files.

When you use the *krarloff rolloff* program to roll off historical data into a text file, any attributes that require format specifiers as indicated in the attribute file are ignored. Only the raw number is seen in the rolled off history text file. Thus, instead of displaying 45.99% or 45.99, the number 4599 displays.

Converting history files to delimited flat files on z/OS systems

The history files collected by the rules established in the History Configuration program or by your definitions related to historical data collection during product installation can be converted to delimited flat files automatically as part of your persistent data store maintenance procedures to manually use a MODIFY command, or *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS* for more information. You can use the delimited flat file as input to a variety of popular applications to easily manipulate the data and create reports and graphs. For more details on the History Collection Configuration window, see the online help or *IBM Tivoli Monitoring: CandleNet Portal User's Guide*.

Data that has been collected and stored cannot be extracted since this data is deleted from the persistent data store. To use these conversion procedures, you must have set the **Warehouse Interval** to **Off** in the History Collection Configuration window.

Automatic conversion and archiving process on z/OS systems

When you customized your IBM Tivoli Monitoring environment, you were given the opportunity to specify the EXTRACT option for maintenance. Specification of the EXTRACT option ensures that scheduling of the process to convert and archive information stored in your history data tables is automatic. No further action on your part is required. As applications write historical data to the history data tables, the persistent data store detects when a given data set is full, launches the KPDXTTRA process to copy the data set, and notifies the Tivoli Enterprise Monitoring Server that the data set can be used again to receive historical information. Additional information about the persistent data store can be found in *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS*.

An alternative to the automatic scheduling of conversion is manually issuing the command to convert the historical data files.

Note: The KBDXTTRA process currently does not support UTF8 columns.

Converting files using the KPDXTTRA program:

The conversion program, KPDXTTRA, is called by the persistent data store maintenance procedures when the EXTRACT option is specified for maintenance. This program reads a data set containing the collected historical data and writes out two files for every table that has data collected for it. The processing of this data does not interfere with the continuous collection being performed. Because the process is automatic, a brief overview of the use of the KPDXTTRA program is provided here. For full information about the KPDXTTRA program, review the

sample JCL distributed with your IBM Tivoli Monitoring product. The sample JCL is found as part of the sample job the KPDXTTRA program contained in the sample libraries RKANSAM and TKANSAM.

Attribute formatting:

Some attributes need to be formatted for display purposes. For example, floating point numbers that specify a certain number of precision digits to be printed to the left of a decimal point. These display formatting considerations are specified in product attribute files.

When you use KPDXTTRA to roll off historical data into a text file, any attributes that require format specifiers as indicated in the attribute file are ignored. Only the raw number is seen in the rolled off history text file. Thus, instead of displaying 45.99% or 45.99, the number 4599 displays.

About KPDXTTRA:

KPDXTTRA program runs in the batch environment as part of the maintenance procedures. It is capable of taking a parameter that allows the default column separator to be changed. The z/OS JCL syntax for executing this command is:

```
// EXEC PGM=KPDXTTRA,PARM='PREF=dsn-prefix [DELIM=xx] [NOFF]'
```

Several files must be allocated for this job to run.

All datasets are kept in read/write state even if they are not active. This makes the datasets unavailable if the Tivoli Enterprise Monitoring Server is running. That is, jobs cannot be run against the active datasets and the inactive datasets must be taken offline. You can dynamically remove a data set from the Tivoli Enterprise Monitoring Server by issuing the modify command:

F stcname,KPDCMD QUIESCE FILE=DSN:data set

If you must run a utility program against an active data store, issue a SWITCH command prior to issuing this QUIESCE command.

DD names required to be allocated for KPDXTTRA:

The following is a summary of the DD names that must be allocated for the KPDXTTRA program. Refer to the sample JCL in the Sample Libraries distributed with the product for additional information.

Table 44. DD names required

DD name	Description
RKPDOUT	KPDXTTRA log messages
RKPDLOG	PDS messages
RKPDIN	Table definition commands file (input to PDS subtask) as set up by the configuration tool
RKPDIN1	PDS file from which data is to be extracted
RKPDIN2	Optional control file defined as a DUMMY DD statement

KPDXTTRA parameters:

The table that follows specifies the KPDXTTRA parameters.

Table 45. KPDXTTRA parameters

Parameter	Default Value	Description
PREF=	none	Required parameter. Identifies the high level qualifier where the output files are written.
DELIM=	tab	Specifies the separator character to use between columns in the output file. The default is a tab character X'05'. To specify some other character, specify the 2-byte hexadecimal representative for that character. For example, to use a comma, specify DELIM=6B.
QUOTE	NQUOTE	Optional parameter that puts double quotes around all character type fields. Trailing blanks are removed from the output. Makes the output format of the KPDXTTRA program identical in format to the output generated by the distributed krarloff rolloff program.
NOFF	off	Causes the creation (if set to ON) or omission (if set to OFF) of a separate file (header file) that contains the format of the tables. Also controls the presence or absence of the header in the output data file that is created as a result of the extract operation. If OFF is specified, the header file is not created but the header information is included as the first line of the data file. The header information shows the format of the extracted data.

KPDXTTRA program messages:

These messages can be found in the RKPDOOUT sysout logs created by the execution of the maintenance procedures:

```
Persistent datastore Extract program KPDXTTRA - Version V130.00
Using output file name prefix: CCCHIST.PDSGROUP
The following characters are used to delimit output file tokens:
Column values in data file.....: 0x05
Parenthesized list items in format file: 0x6b
Note: Input control file not found; all persistent data is extracted.
```

Table(s) defined in persistent datastore file CCCHIST.PDSGROUP.PDS#1:

Application Name	Table Name	Extract Status
PDSSTATS	PDSCOMM	Excluded
PDSSTATS	PDSDEMO	Included
PDSSTATS	PDSLOG	Included
PDSSTATS	TABSTATS	Included

Checking availability of data in data store file:

```
No data found for Appl: PDSSTATS Table: PDSDEMO . Table excluded.
No data found for Appl: PDSSTATS Table: TABSTATS . Table excluded.
```

The following 1 table(s) are extracted:

Application Name	Table Name	No. of Rows	Oldest Row	Newest Row
PDSSTATS	PDSLOG	431	1997/01/10 05:51:20	1997/02/04 02:17:54

Starting extract operation.
Starting extract of PDSSTATS.PDSLOG.
The output data file, CCCHIST.PDSGROUP.D70204.PDSLOG, does not exist; it is created.
The output format file, CCCHIST.PDSGROUP.F70204.PDSLOG, does not exist;
it is created.
Extract completed for PDSSTATS.PDSLOG. 431 data rows retrieved, 431 written.
Extract operation completed.

Location of the z/OS executables and historical data table files

The z/OS executables are located in the *&hilev.&midlev.RKANMOD* or *&hilev.&midlev.TKANMOD* library, where:

- *&hilev* is the library in which the Tivoli Enterprise Monitoring Server was installed
- *&midlev* is the name you have provided at installation time.

The z/OS historical data files created by the extraction program are located in the following library structure:

- *&hilev.&midlev.&dsnlolev.tablename.D*
- *&hilev.&midlev.&dsnlolev.tablename.H*

Manual archiving procedure

To manually convert historical data files on the Tivoli Enterprise Monitoring Server and on the remote managed systems, issue the following MODIFY command:

```
F stcname,KPDCMD SWITCH GROUP=cccccccc EXTRACT
```

where:

- *stcname* identifies the name of the started task that is running either the Tivoli Enterprise Monitoring Server or agents.
- *cccccccc* identifies the group name associated with the persistent data store allocations. The values for *cccccccc* can vary based on which products are installed. The standard group name is GENHIST.

When this command is run, only the tables associated with the group identifier are extracted. If multiple products are installed, each can be controlled by separate SWITCH commands.

This switching can be automated by using either an installation scheduling facility or an automation product.

You can also use the Tivoli Enterprise Portal's advanced automation features to run the SWITCH command. To do so, define a situation that, when it becomes true, runs the SWITCH command as the action.

Maintaining the Persistent Data Store

You have the option to run the PDS on the z/OS Tivoli Enterprise Monitoring Server or the agent. It provides the ability to record and retrieve tabular relational data 24 hours a day while maintaining indexes on the recorded data. See *IBM Tivoli Management Services on z/OS: Configuring the Tivoli Enterprise Monitoring Server on z/OS* on the Tivoli Monitoring and OMEGAMON XE Information Center for instructions on configuring the persistent datastore.

Chapter 14. Tivoli Common Reporting

The Tivoli Common Reporting tool is a reporting feature available to users of Tivoli products and provides a consistent approach to viewing and administering reports. Tivoli products can provide report packages designed for use with Tivoli Common Reporting, with reports that use a consistent look and feel.

Tivoli Common Reporting consists of several components:

- A *data store* for storing and organizing report designs, reports, and supporting resources. The data store is a location within the Tivoli Common Reporting infrastructure where all report-related files and reports are managed and maintained.
- A Web-based user interface for specifying report parameters and other report properties, generating formatted reports, and viewing reports.
- A command-line interface for working with objects in the data store and performing additional administrative functions.
- *Report packages*, archive files containing reports, documentation, graphics, and dynamic link libraries. Report packages for some monitoring agents are included as .zip files on the Application CD in the REPORTS directory, and the REPORTS directory is divided into subdirectories named with the three-character prefix that identifies the product. Report packages for some monitoring agents are available from the IBM Tivoli Open Process Automation Library (<http://www-18.lotus.com/wps/portal/topal>). You can search on “Tivoli Common Reporting” to find report packages on OPAL. A sample set of reports was provided with the Tivoli Common Reporting product. Other sets can be downloaded and installed using the Import facility. You can find additional report packages generated by other non-IBM users, business report templates, and the *Tivoli Common Reporting: Development and Style Guide* on the IBM developerWorks Web site: <http://www.ibm.com/developerworks/spaces/tcr>.
- The open-source Eclipse BIRT Report Designer that you can use to modify reports or create your own. This tool is not included with Tivoli Common Reporting, but can be downloaded from <http://www.eclipse.org/birt/phoenix/> or from the Tivoli Common Reporting page at IBM developerWorks (<http://www.ibm.com/developerworks/spaces/tcr>).

Get Tivoli Common Reporting

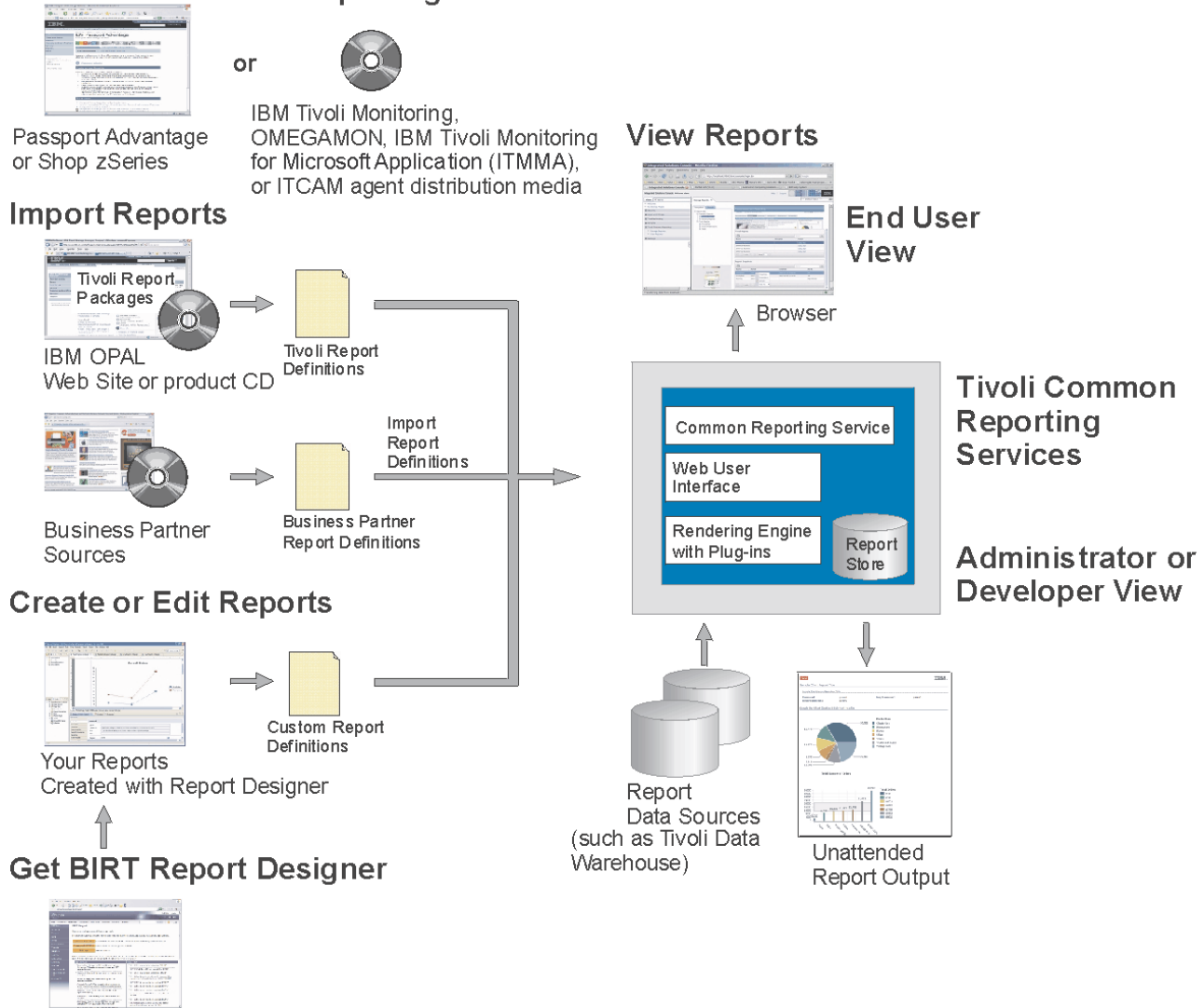


Figure 3. Tivoli Common Reporting environment

For more information about Tivoli Common Reporting, including information about installing and administering Tivoli Common Reporting and creating reports, refer to the *Tivoli Common Reporting User's Guide* or refer to the product information center: http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.tivoli.tcr.doc/tcr_welcome.html

Tivoli Common Reporting Users

This chapter covers information about Tivoli Common Reporting that is unique to products that run on the Tivoli Enterprise Portal and use the Tivoli Data Warehouse as the source of historical data for generating reports. This information is intended for the administrator who sets up Tivoli Common Reporting and installs report packages for use by these reports consumers within the customer enterprise:

- The network systems programmer who troubleshoots TCP/IP issues
- The application analyst or documentation manager
- The IT manager or service level advisor who validates service level agreements

- The capacity planner
- The service manager
- The system administrator
- The storage administrator

Prerequisites

To use the reports, you need the following components:

- IBM Tivoli Monitoring, version 6.2 Fix Pack 1 or later
- Tivoli Common Reporting version 1.1 or later

To run the reports provided with an OS monitoring agent, you must first install Tivoli Common Reporting version 1.1.1. For other monitoring agents, install Tivoli Common Reporting version 1.1 or later.

If you have not done so already, install and configure Tivoli Common Reporting, using the information found in the *Tivoli Common Reporting User's Guide*.

- Report packages
- Historical data stored in a database manager product supported by IBM Tivoli Monitoring 6.2 Fix Pack 1 or later

BIRT reports in this guide are historical reports, reporting against data collected in Tivoli Data Warehouse 6.2 Fix Pack 1 or later. For information about supported databases, refer to the most recent editions of the *IBM Tivoli Monitoring: Installation and Setup Guide*.

Note: Although not required, you can install the Eclipse BIRT Report Designer, version 2.2.1. Eclipse BIRT Report Designer, along with the *Tivoli Common Reporting: Development and Style Guide*, can be used to edit report templates or create new reports.

Download the report designer from this Web site: <http://www.eclipse.org/birt/phenix/> or from the Tivoli Common Reporting page at IBM developerWorks (<http://www.ibm.com/developerworks/spaces/tcr>). Download the development and style guide from the Tivoli Common Reporting information center: http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tivoli.tcr.doc/tcr_style_guide.pdf

For information about the software requirements for Tivoli Common Reporting, refer to the *IBM Tivoli Common Reporting User's Guide*.

For software requirements for running the BIRT Report Designer, refer to <http://www.eclipse.org/birt/phenix/>.

Upgrading from a previous version

OS monitoring agent reports continue to be delivered on OPAL; these reports run under Tivoli Common Reporting version 1.1.1. For other monitoring agents, which were previously delivered on OPAL and ran under Tivoli Common Reporting version 1.1.1, you can upgrade your Tivoli Common Reporting product level to the version included with the monitoring agent (version 1.2) without reinstalling the report packages downloaded from OPAL or from the product media.

However, Tivoli Common Reporting version 1.1.1 ran under the Integrated Solutions Console and installed into a different location from Tivoli Common Reporting version 1.2, which runs under the Tivoli Integrated Portal (TIP) and now

relies on that product for infrastructure support. Versions 1.1.1 and 1.2 can coexist on the same computer or you can migrate the reports you downloaded from OPAL to version 1.2. You do not need to reinstall the report packages. There are two options for migrating reports from version 1.1.1 to version 1.2:

- During installation of Tivoli Common Reporting version 1.2, the installer program detects if version 1.1.1 is installed and asks if you want to migrate these reports. Say **Yes**.

Note: If you migrated the report package you downloaded from OPAL to Tivoli Common Reporting version 1.2, be sure that the previously installed reports are overwritten. When you import the report package, click on **Advanced Options** in the **Import Report Package** text box and select the **Overwrite** check box.

- Migrate report packages manually.

Both of these options are explained in the version 1.2 *Tivoli Common Reporting User's Guide*.

Note: Tivoli Common Reporting now provides enhanced security that enables you to assign a security string to hypertext links in a report. The *Tivoli Common Reporting User's Guide* provides instructions for entering a security set.

Limitations

Be aware of these limitations for reports.

- The reports do not support the Tivoli Data Warehouse Summarization and Pruning Agent optional definition of shift hours. Customers can use shift hour support to flag collected data as being either Peak or Off-Peak periods. However, reports will include all data collected between the customer-selected report start and end times, whether that data was collected during Peak or Off-Peak periods. For additional information, see the *IBM Tivoli Monitoring: Administrator's Guide*.
- Reports that cover a long time period or a processing-intensive attribute might cause SQL arithmetic overflow. For more information about this limitation, see the agent reporting chapter or *Product reporting guide* for troubleshooting information.
- These reports run against the Tivoli Data Warehouse. DB2 limits the length of columns to 30 characters. Since the Tivoli Data Warehouse uses attribute group names as the column headers, attribute names longer than 30 characters in a DB2 warehouse are replaced with the internal column name, abbreviated database name for the attribute (for example, CPU_UTIL or DISK_UTIL rather than CPU Utilization or Disk Utilization).

Importing and running reports

Step 1: Ensure that historical reporting is enabled

About this task

The reports in this report package run against long-term historical data that is stored in the Tivoli Data Warehouse. Before you can run these reports, ensure that you have done the following:

1. Installed the Tivoli Data Warehouse and the Warehouse Proxy Agent according to the instructions in the *IBM Tivoli Monitoring: Installation and Setup Guide*.

2. Set up historical collection by allocating short-term data collection (the mainframe persistent data store) using the Configuration Tool and enabling short-term and long-term data collection using the Historical Collection window in the Tivoli Enterprise Portal.
3. Optionally enabled access to summarized data in the Tivoli Data Warehouse. The use of summarized data in reports can simplify analysis of displayed reports and improve the performance of generating the reports. Install the Summarization and Pruning Agent according to instructions in the *IBM Tivoli Monitoring: Installation and Setup Guide*. Use the Historical Collection window in the Tivoli Enterprise Portal to enable summarization. Refer to the *IBM Tivoli Monitoring: Administrator's Guide* for more information about enabling summarization. Enabling summarization for the default attribute groups ensures that summarization is enabled for all reports. Alternatively, you may enable summarization for a reduced set of attribute groups. The subset of attribute groups used to provide summary reports are identified in the *Agent user's guide* or *Product reporting guide* for the agent or product with which you are working.
4. Started the Tivoli Data Warehouse and the Warehouse Proxy and allowed the Tivoli Data Warehouse to collect the data for your requested report time period or the appropriate amount of data for a summarized report (that is, if you want a monthly report, you need at least a month's worth of data).

Step 2: Import a report package

Import the report package for a monitored application to get the necessary files for defining reports.

Before you begin

A *report package* is a .zip file containing all of the data necessary for defining one or more reports, including the required designs and resources and the hierarchy of report sets to contain the reports. The monitoring agent reports are included as .zip files on the agent image in the REPORTS directory. For example, on a Windows computer, if the image drive is labelled D:, reports are in directories such as: D:\REPORTS\kqb. See the agent reporting chapter or *Product reporting guide* for the location of the reports.

About this task

Take these steps to import a report package:

1. Launch the Tivoli Integrated Portal administrative console and log in.
2. In the menu area on the left, expand the **Reporting** item.
3. Under **Reporting**, click **Common Reporting**.
4. In the report navigation window, click on the **Navigation** tab (default).
5. Right-click the root node of the navigation tree (Report Sets) and click **Import Report Package**.
6. Specify the location and file name of the report package .zip file to import. You must specify the name of a security set. You can type directly in the entry field or click **Browse** to open a file selection window from which you can select the report package file. If you want to overwrite existing reports, select ☒ **Overwrite** to indicate that any existing file with the same name as an imported file is to be overwritten.

See the *Tivoli Common Reporting User's Guide* for more information about the Advanced Options and import window.

Results

The Navigation tree shows an item for the reports and items for subsets of the reports.

What to do next

Changing the data source in a report will change the data sources for all reports. You do not need to repeat the change for all reports.

Step 3: Configure the data source

All reports in a report package must point to the same data source. The data source pointer needs to be modified to point to your Tivoli Data Warehouse.

About this task

After you have installed Tivoli Common Reporting and imported your first set of reports, you or a user with Administrator authority must copy JDBC drivers from the local or remote database manager that you are using to run the Tivoli Data Warehouse into the Tivoli Common Reporting server directory. You specify these files in the **Edit Data Source** window. Perform the following steps to install these drivers.

1. Locate the JDBC driver files, db2jcc.jar and db2jcc.license_cu.jar:

- **Windows** C:\Program Files\IBM\SQLLIB\java
- **Linux** **UNIX** Linux UNIX Path.
- You can also download these files from this Web site: https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?lang=en_US&source=swg-dm-db2jdbcdriver

The JDBC drivers are typically found in this default DB2 installation path or in the java directory of whatever alternate path you specified for DB2 installation.

2. Copy db2jcc.jar and db2jcc.license_cu.jar to your Tivoli Common Reporting installation directory:
 - **Windows** <tc_r_install_dir>\tip\products \tcr\lib\birt-runtime-2_2_1\ReportEngine\plugins \org.eclipse.birt.report.data.oda.jdbc_2.2.1.r22x_v20070919\drivers
 - **Linux** **UNIX** Linux UNIX Path.
3. Right-click the name of a report and click **Data Sources**.
4. Click **Edit** in the Report Data Sources dialog. Open the Enabling a JDBC Driver list to see the location for the JDBC drivers.
5. In the **JDBC Driver** field, enter the path to the JDBC driver, com.ibm.db2.jcc.DB2Driver:
 - For a local database manager: jdbc:db2:WAREHOUS:currentSchema=ITMUSER;
 - For a remote database manager: jdbc:db2://<somehost.someomain>.com:50000/WAREHOUS :currentSchema=ITMUSER;
6. Change your username and password in the window to be the same as your database manager login ID (for example, your DB2 username).
7. Click **Save**.

What to do next

For additional information, refer to the JDBC driver section of the *IBM Tivoli Common Reporting: Development and Style Guide*.

For more information about Tivoli Data Warehouse connectivity issues, refer to the “Setting up data warehousing” section of the *IBM Tivoli Monitoring Installation and Setup Guide*.

Step 4: Generate a sample report

Tivoli Common Reporting report packages are organized by product. Select a report set to generate a report.

About this task

Take these steps to select a report set and run a report.

1. Launch the Tivoli Common Reporting Browser. From the Start menu, select **TCR→ TCR Browser**. The Tivoli Integrated Portal login panel is displayed. Log in.
2. In the menu area on the left, expand the **Reporting** item.
3. Click **Common Reporting**. All the reports available (all the report packages that you have imported) are displayed in the text area of the screen.
4. On the **Navigation** tab, expand the **Tivoli Products** item.
5. Select the Tivoli product whose reports you want to use from the list of available products.
6. If this is the first time you have run reports based on data from the Tivoli Data Warehouse, perform the following steps:
 - a. Define the Tivoli Data Warehouse as the data source for your reports. For information about data sources, refer to the *IBM Tivoli Common Reporting User's Guide* or the online help for Tivoli Common Reporting.
 - b. Copy the required JDBC drivers from the local or remote database manager that you are using to run the Tivoli Data Warehouse into the Tivoli Common Reporting server directory.

You might need to increase the default heap size for the Java Virtual Machine (JVM) on the Java command to start the Tivoli Common Reporting server. If you see these messages displayed when you create a report, default heap size might be your problem:

Processing has ended because of an unexpected error.
See the Tivoli Common Reporting log files for more information.

See *OMEGAMON XE and Tivoli Management Services on z/OS: Reports for Tivoli Common Reporting* for information on how to increase the default heap size.

7. Click the icon beside a report name to launch the parameters window and produce a report in the desired format. You can select from these report format by right-clicking the report name and selecting one of these report formats: HTML (the default), PDF, Microsoft Word, Microsoft Excel, or Adobe Postscript.

When you select a report from Tivoli Common Reporting, you are presented with a **Report Parameters** window that prompts you for information that will be used to generate the report. The title of the parameters window indicates the type of report that will be generated. For a description of the types of reports, see the *Agent user's guide* or *Product reporting guide* for the agent or product with which you are working.

A Report Parameters window contains some fields common to all reports (for example, timeframe). Other fields are specific to the agent running the reports. For most reports, you select a timeframe, resources, the summarization level of the data, and the attributes to graph, or press Enter to accept all displayed defaults.

8. Click **Run** to generate a report matching your parameter definitions.

Results

You will see an hour glass while Tivoli Common Reporting gathers report data and creates formatted output. After processing finishes, the report viewer opens in a new browser tab or instance, displaying the formatted report using the appropriate browser plug-in. You can view the report in your browser or save the formatted output using the browser or plug-in capabilities.

What to do next

If no report is generated or you see a message indicating that the requested data is unavailable, refer to the *IBM Tivoli Common Reporting User's Guide* for information about defining a data source.

If you are viewing an HTML or PDF report, you can also click any embedded links to open drill-through reports. Clicking a drill-through embedded link causes the report to link back to itself with the newly passed parameters or to a secondary (drill-down or summarized) report. Examples of drill-through links include clicking on a bar or line chart or on a table heading.

Chapter 15. Replicating the Tivoli Enterprise Portal Server database

With the exception of situations, policies and managed system lists, Tivoli Enterprise Portal customizations are stored at the Tivoli Enterprise Portal Server in the TEPS database. This includes user IDs, Navigator views, custom queries, custom workspaces, and local terminal scripts.

This chapter describes how to replicate the TEPS database, which is necessary for moving from a test environment to a production environment. You can also use this procedure for backing up the database as a precautionary measure before applying a fix pack or upgrading to a new version. When you upgrade to a new version of the , the TEPS database is updated with any new or changed predefined Navigator views, queries and workspaces. Custom Navigator views, queries, and workspaces that you created are not affected.

Note:

1. All workspaces previously created in the destination environment are replaced with those that were created in the source environment. Any existing user changes in the destination environment are also replaced.
2. You can also selectively copy workspaces from one to another. See *tacmd exportworkspaces* and *tacmd importworkspaces* in the *IBM Tivoli Monitoring: Command Reference*.
3. There are also commands for exporting situations and policies from one hub monitoring server and importing to another: *tacmd bulkExportSit*, *tacmd bulkImportSit*, *tacmd bulkExportPcy*, and *tacmd bulkImportPcy*.

Prerequisites

Prior to migrating your Tivoli Enterprise Portal Server you must ensure that you have satisfied the following requirements:

- The portal servers on the source and target computers must be configured to connect to the same hub monitoring server.
- The portal servers on the source and target computers must be at Version 6.2.1 and, ideally, both have been installed from the same CD image.
- The portal servers on the source and target computers must have been installed the same way:
 - The selected applications are the same. For example, if the source portal server has support for the UNIX, Windows Servers, and MQ Series applications installed, then the target portal server must have the same application support.
 - The same database program is used for the Tivoli Enterprise Portal Server database. For example, IBM DB2 UDB.

Running the migrate-export script

Export the Tivoli Enterprise Portal Server to create a copy of the TEPS data base for applying to another computer or to keep as a backup.

Before you begin

The portal server can be running or stopped when you initiate the migrate-export script. If the server is stopped, the script starts it temporarily in a limited mode to accomplish the export. Do not start the portal server manually until the migrate-export has completed.

About this task

On the computer where the source is installed, take these steps to create a copy of the TEPS database

- **Windows**

1. Open a command prompt window: **Start**→ **Run**, enter CMD.
2. Change to the `<install_dir>\CNPS` directory.
3. Enter: migrate-export

The migrate-export script generates a file named **saveexport.sql** in the `<install_dir>\CNPS\sqllib` subdirectory. It contains all the Tivoli Enterprise Portal Server data.

- **Linux** **UNIX**

1. On the source system, open a terminal window.
2. Change to the bin subdirectory of your IBM Tivoli Monitoring installation, such as: `cd /opt/IBM/ITM/bin`
3. Enter: `./itmcmd execute cq "runscript.sh migrate-export.sh"` Be sure to use the " double-quote symbol and not ' single-quote.

The migrate-export script generates a file named **saveexport.sql** in the `<install_dir>/$platform/cq/sqllib` subdirectory. It contains all the Tivoli Enterprise Portal Server data.

Running the migrate-import script

When you have a copy of the Tivoli Enterprise Portal Server database, named `saveexport.sql`, import it to a any portal server installation of the same version where you want duplicate settings.

Depending on the contents of the `saveexport.sql`, this process can completely replace the existing TEPS database.

Some of the tables included in the import script are applicable only to the CandleNet Portal Server, the predecessor to . If you are not importing a CandleNet Portal Server database, the migrate-import log file will contain SQL errors about an undefined name, such as `SQLExecDirect rc=-1: SQL_ERROR SQLSTATE: 42S02, ERR: -204, MSG: [IBM][CLI Driver][DB2/LINUX] SQL0204N "ITMUSER.TAGGROBJ" is an undefined name. SQLSTATE=42704 RC = -1 (also ITMUSER.TMANOBS, ITMUSER.TMANTMPL, ITMUSER.TTMPLSIT, ITMUSER.TTMPLSTA, ITMUSER.TSTUSERA)`. Ignore these errors.

Running migrate-import from source Windows to target Windows

Run the migrate-import script to import a copy of the Tivoli Enterprise Portal Server database from a Windows computer to another Windows computer.

Before you begin

This procedure overwrites the TEPS database on the target computer.

About this task

On the Windows computer where the target portal server is installed, take these steps to import the TEPS database that was copied from another Windows computer using migrate-export.

1. Stop the portal server on the target system.
2. On the source system, open a command prompt: Click **Start** → **Run**, and enter CMD.
3. Copy file **saveexport.sql** that was generated by the migrate-export.bat script from the source system to *<install_dir>\CNPS\sqllib* on the destination system, where *<mapped drive on destination system>* is the disk drive on the source system where this file resides. Example:

```
copy <mapped drive on destination system>:\IBM\ITM\CNPS\sqllib
    \saveexport.sql c:\ibm\itm\cnps\sqllib
```

If a drive is not already defined, you need to map a drive to the source system from the destination system with the net use command.

4. On the target system, change to the *<install_dir>\CNPS* directory and enter: migrate-import. Running the migrate-import process stops the portal server if it is currently running.
5. If you are using the migrate-import function to move the TEPS database from one release to another, perform this task after migrating the database to add application support:
 - a. Open *<install_dir>\CNPS\kfwalone* in a text editor.
 - b. Set KFW_MIGRATE_FORCE=Y, then save and close the file.
 - c. Invoke this script to apply the current portal server application support to the newly migrated TEPS database: *<install_dir>\CNPS\buildpresentation.bat*
6. Restart the portal server.

Running migrate-import from source Windows to target Linux or UNIX

Run the migrate-import script to import a copy of the Tivoli Enterprise Portal Server database from a Windows computer to a Linux or UNIX computer.

Before you begin

This procedure overwrites the TEPS database on the target computer.

About this task

On the Linux or UNIX computer where the target portal server is installed, take these steps to import the TEPS database that was copied from a Windows computer using migrate-export.

1. Stop the portal server on the target system.
2. On the source system, open a command prompt: Click **Start** → **Run**, and enter CMD.

3. Copy **saveexport.sql** that was generated by the migrate-export.bat script from the source Windows system to the target system's /opt/IBM/ITM/\$platform/cq/sqllib directory, where \$platform is li6243 for Intel Linux or ls3263 for zSeries® Linux on the destination system.
4. Open a terminal window on the target system.
5. Change to the bin subdirectory of the Tivoli Monitoring installation: `cd /opt/IBM/ITM/$platform/bin`
6. In the terminal window, enter: `./itmcmd execute cq "runscript.sh migrate-import.sh"` Be sure to use the " double-quote symbol and not ' single-quote. The script processes a file named saveexport.sql in the IBM/ITM/\$platform/cq/sqllib subdirectory. Depending on the contents of the saveexport.sql file, this process can completely replace the existing portal server data.
7. If you are using the migrate-import function to move the TEPS database from one release to another, perform this task after migrating the database to add application support:
 - a. Open `<itm_installdir>/cw/bin/lnxenvnocms` in a text editor.
 - b. Set `KFW_MIGRATE_FORCE=Y`, then save and close the file.
 - c. Invoke this script to apply the current portal server application support to the newly migrated TEPS database: `<install_dir>/cw/bin/InstallPresentation.sh`
8. Restart the portal server from the `<itm_installdir>/bin` directory: `./itmcmd agent start cq`.

Running migrate-import from source Linux or UNIX to target Windows

Run the migrate-import script to import a copy of the Tivoli Enterprise Portal Server database from a Linux or UNIX computer to a Windows computer.

Before you begin

This procedure overwrites the TEPS database on the target computer.

About this task

On the Windows computer where the target portal server is installed, take these steps to import the TEPS database that was copied from a Linux or UNIX computer using migrate-export.

1. Stop the portal server on the target system.
2. Copy file **saveexport.sql** that was generated by the migrate-export script from the source Linux or UNIX system (/opt/IBM/ITM/\$platform/cq/sqllib) to `<install_dir>\CNPS\sqllib` on the target system.
3. On the target system, change to the `<install_dir>\CNPS` directory and enter: `migrate-import`. Running the migrate-import process stops the portal server if it is currently running.
4. If you are using the migrate-import function to move the TEPS database from one release to another, perform this task after migrating the database to add application support:
 - a. Open `<install_dir>\CNPS\kfwalone` in a text editor.
 - b. Set `KFW_MIGRATE_FORCE=Y`, then save and close the file.

- c. Invoke this script to apply the current portal server application support to the newly migrated TEPS database: `<install_dir>\CNPS\buildpresentation.bat`
5. Restart the portal server.
6. Restart the Tivoli Enterprise Portal Server.

Running migrate-import from source Linux or UNIX to target Linux or UNIX

Run the migrate-import script to import a copy of the Tivoli Enterprise Portal Server database from a Linux or UNIX computer to another Linux or UNIX computer.

Before you begin

This procedure overwrites the TEPS database on the target computer.

About this task

On the Linux or UNIX computer where the target portal server is installed, take these steps to import the TEPS database that was copied from another Linux or UNIX computer using migrate-export.

1. Stop the portal server on the target system.
2. Copy file **saveexport.sql** that was generated by the migrate-export script from the source Linux or UNIX system (`/opt/IBM/ITM/$platform/cq/sqllib`) to the target system's `/opt/IBM/ITM/$platform/cq/sqllib` directory, where `$platform` is `li6243` for Intel Linux or `ls3263` for zSeries Linux on the destination system.
3. Open a terminal window on the target system.
4. Change to the bin subdirectory of the Tivoli Monitoring installation: `cd /opt/IBM/ITM/$platform/bin`
5. In the terminal window, enter: `./itmcmd execute cq "runscript.sh migrate-import.sh"` Be sure to use the " double-quote symbol and not ' single-quote. The script processes a file named `saveexport.sql` in the `IBM/ITM/$platform/cq/sqllib` subdirectory. Depending on the contents of the `saveexport.sql` file, this process can completely replace the existing portal server data.
6. If you are using the migrate-import function to move the TEPS database from one release to another, perform this task after migrating the database to add application support:
 - a. Open `<itm_installdir>/cw/bin/lnxenvnocms` in a text editor.
 - b. Set `KFW_MIGRATE_FORCE=Y`, then save and close the file.
 - c. Invoke this script to apply the current portal server application support to the newly migrated TEPS database: `<install_dir>/cw/bin/InstallPresentation.sh`
7. Restart the portal server from the `<itm_installdir>/bindirectory`: `./itmcmd agent start cq`.

Appendix A. Tivoli Enterprise Monitoring Web services

This appendix describes the Tivoli Enterprise Monitoring Web Services feature.

The Tivoli Enterprise Monitoring Web Services solution provides you with an industry-standard open interface into IBM Tivoli Monitoring solutions. This open interface provides easy access to Tivoli performance and availability data, allowing you to use this information for advanced automation and integration capabilities.

Tivoli Enterprise Monitoring Web Services implements a client/server architecture. The client sends Simple Object Access Protocol (SOAP) requests to the SOAP server. The server receives and processes the SOAP requests from the client.

Predefined SOAP methods let you perform many functions within the monitored environment. You can begin to use the SOAP methods immediately. You can also use these SOAP methods as templates in creating your own advanced methods.

SOAP works with any programming or scripting language, any object model and any Internet wire protocol. Tivoli SOAP methods can be invoked by PERL, Javascript, VBSCRIPT, JSCRIPT, C++, Java, and through a browser.

See the *IBM Tivoli Monitoring: CandleNet Portal User's Guide* for instructions on installing and configuring this product on the Tivoli Enterprise Monitoring Server.

Note:

1. Web Services does not support situation creation. Use the Tivoli Enterprise Portal Situation editor or the IBM Tivoli Monitoring command line `tacmd createSit` function for situation creation. The SOAP server can query only agent and managed system attributes.
2. If you will be issuing SOAP request on Internet Explorer version 5.0x, you must first install service pack MSXML 3.0. Otherwise, the SOAP requests will not succeed. Internet Explorer version 6.0x includes this service pack.

Configuring Tivoli Monitoring Web Services (SOAP Server)

By default, the SOAP server is installed on the hub . Use the following sections to configure SOAP server communication between hubs and to establish security on the SOAP server.

Note: You cannot make SOAP requests from IBM Tivoli Monitoring to earlier SOAP servers (such as those on an OMEGAMON platform V350).

The instructions in this chapter assume that you have a basic understanding of SOAP, XML and XML Namespaces, and the Web Services Description Language (WSDL).

These steps are required to configure SOAP:

- Define the hubs with which your SOAP Server communicates.
- Create users and grant them access.
- Verify that you have successfully configured SOAP.

Defining hubs

About this task

In this step you use the Manage Tivoli Monitoring Services to activate the SOAP server and define hubs with which the SOAP server communicates.

Use the following steps to define SOAP hubs:

1.

On the computer where the hub monitoring server is installed, start Manage Tivoli Monitoring Services:

Windows Click **Start** → **Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**.

Linux or **UNIX** Change directory to `<itm_install_dir>/bin` and enter:
`./itmcmd manage`

2. Right-click and click **Reconfigure**.

3. Select or clear the ☐ **Security: Validate User** field.

4. Open Manage Tivoli Monitoring Services.

5. Right-click **Tivoli Enterprise Monitoring Server**.

6. Click **Advanced** → **Configure SOAP Server Hubs**.

7. Click **Add Hub**. The Hub Specification window is displayed:

8. Select the communications protocol to be used with the from the **Protocol** menu.

9. Specify an alias name in the **Alias** field (for example: HUB2). Alias names can be a minimum of 3 characters and a maximum of 8 characters.

10. Do one of the following:

- If you are using TCP/IP or TCP/IP Pipe communications, complete the following fields:

Table 46. TCP/IP Fields in Hub Specification Dialog

Field	Description
Hostname or IP Address	The host name or TCP/IP address of the host computer.
Port	The TCP/IP listening port for the host computer.

- If you are using SNA communications, complete the following fields:

Table 47. SNA Fields in Hub Specification Dialog

Field	Description
Network Name	Your site SNA network identifier.
LU Name	The LU name for the monitoring server. This LU name corresponds to the Local LU Alias in your SNA communications software.
LU6.2 LOGMODE	The name of the LU6.2 logmode. Default: CANCTDCS.
TP Name	The Transaction Program name for the monitoring server.

Note: If you are connecting to a remote monitoring server, the protocol information must be identical to that used for the hub monitoring server.

11. Click OK. The server tree is displayed.

Adding users

About this task

In this step you define users on each hub and specify the access rights for each user (query or update).

Use the following steps:

1. Select the server (click anywhere within the server tree displayed), if necessary.
2. Under Add User Data, type the user name. User IDs must be identical to those specified for monitoring server logon validation. Access is restricted to only that monitoring server to which a user has access.

Note: If you do not supply a user ID, all users are given permission to update data.

3. Click the type of user access: **Query** or **Update**.
4. Click **Add User**. The server tree is updated, showing the user and type of access.
5. To delete a user: Select the user name from the tree and click **Delete Item**.
6. To delete a hub: Click anywhere within the hub's tree and click **Clear Tree**.

Configuring IBM Tivoli Monitoring Web Services (SOAP Server) on UNIX and Linux

About this task

Use the following steps to define SOAP hubs on UNIX or Linux using Manage Tivoli Monitoring Services:

1. Change to the `<itm_install_dir>/bin` directory and start Manage Tivoli Monitoring Services by entering the following command:

```
./itmcmd manage
```

The Manage Tivoli Monitoring Services Window is displayed.

2. Right-click **Tivoli Enterprise Monitoring Server** and select **Configure** from the popup menu.

The Configure TEMS window is displayed.

3. Click **Save**.

The SOAP Server Hubs Configuration window is displayed. If the current host is not displayed in the Hubs tree, define it before defining the hubs with which it communicates.

4. Confirm that the host name or IP address, port number, and protocol for the hub monitoring server are correct. If not, correct them.

If the name of the local hub does not appear in the tree, define the local hub before defining the hubs with which it communicates. The alias for the local hub must always be "SOAP".

5. To add another hub:
 - a. Type the name or IP address and port number of the host in the appropriate fields.
 - b. Specify an alias name in the **Alias** field.

- Alias names can be a minimum of 3 characters and a maximum of 8 characters (for example, HUB2).
- c. Select the communications protocol to be used with the hub from the **Transport** menu.
6. Click **Add Host**.
- The server tree is displayed, with the newly defined hub.

Tuning SOAP transaction performance on AIX

About this task

The default behavior on AIX[®] systems for Transmission Control Protocol (TCP) connections is to allow delayed acknowledgements (*Ack* packets) by up to 200 ms, and is controlled by the **tcp_nodelayack** network option. This delay allows the packet to be combined with a response and it minimizes system overhead. If you set **tcp_nodelayack** to **1**, the acknowledgement is immediately returned to the sender. With this setting, slightly more system overhead is generated but results in much higher network transfer performance when the sender is waiting for acknowledgement from the receiver.

To set this parameter, access a user account that has **root** privileges and issue the following command:

```
no -p -o tcp_nodelayack=1
```

The following output is typical:

```
Setting tcp_nodelayack to 1
Setting tcp_nodelayack to 1 in nextboot file
```

This is a dynamic change that takes effect immediately. The **-p** flag makes the change persistent, so that it is still in effect the next time you start the operating system.

To find out more about the **tcp_nodelayack** option, refer to the IBM System p and AIX Information Center.

About the SOAP client

Simple Object Access Protocol (SOAP) is a communications XML-based protocol that lets applications exchange information through the Internet. SOAP is platform independent and language independent. SOAP uses XML to specify a request and reply structure. It uses HTTP as the transport mechanism to drive the request and to receive a reply.

Important: Prior to using IBM's solution, you must have a basic understanding of SOAP, of Extensible Markup Language (XML) and XML Namespaces, and of the Web Services Description Language (WSDL).

Using IBM Tivoli Monitoring Web services

IBM provides numerous SOAP methods with IBM Tivoli Monitoring Web services. These methods allow you to dynamically query and control IBM Tivoli Monitoring environments.

Using IBM's SOAP methods, you can:

- Stop or start policies and situations

- Forward trapped messages from System Automation for Integrated Operations Management and display them on a Universal Message console
- Retrieve attribute data that you can display in charts or reports
- Open and close events
- Make real-time requests for data
- Issue SOAP requests as system commands in Tivoli Enterprise Portal

You can also use this product to test a request to ensure it works properly. You can then create a policy that submits multiple requests for processing. In addition, you can generate daily operation summaries.

You can store retrieved data in the Tivoli Data Warehouse, as described in the historical data collection guide.

Note: IBM Tivoli Monitoring Web Services provides XML data rows. Use IBM's SOAP methods in combination with your own scripts to display the data in charts and tables.

User IDs

At installation and configuration time, you are asked to supply user IDs for those who need access to monitoring server data. If no user IDs are supplied, all users are given permission to update data.

User IDs must be identical to those specified for monitoring server logon validation. Access is restricted to only that monitoring server to which a user has access.

You can also make changes at a later time to add or to remove users' access to monitoring server data. See the *IBM Tivoli Monitoring: Installation and Setup Guide* for details.

Starting the SOAP client and making a request

About this task

There are several ways of starting the SOAP client. Two are described here:

- Using Internet Explorer
- Using the SOAP client command-line utility (not available on z/OS systems)

When you use the SOAP client in conjunction with Internet Explorer to issue SOAP requests, you can modify, if needed, the tags or the text. In contrast, the command-line utility simply displays the output of the request at the command prompt.

Note: Before you can access newly created Universal Agent objects, the hub monitoring server where the SOAP server is running must be recycled. See the *IBM Tivoli Monitoring Installation and Setup Guide* for instructions on configuring the hub monitoring server.

Using your browser

Use Windows Internet Explorer or Mozilla Firefox to enter the URL for the SOAP service console.

About this task

After installing the Tivoli Monitoring Web Services SOAP client, perform these actions:

1. Start Internet Explorer version 5 or higher or Mozilla Firefox . Be sure to enable the **Access data sources across domains** option in Internet Explorer's security settings.
2. In the Address field, type the URL for the SOAP client, where localhost can be used literally when accessing the SOAP server running on the same system or changed to the proper host name or network address of a SOAP server running on a different system:

`http://localhost:1920///cms/soap/kshhsoap.htm`

The port number for the HTTP service is 1920.

Note: You can also route requests to a remote hub by replacing **soap** in the Address field with the alias name of the hub you want to access (**HUB_localhost** in the example below). The alias must have been previously defined to the SOAP server (for information about defining hub aliases, see the installation documentation). For example: `http://localhost:1920///cms/HUB_localhost/kshhsoap.htm`
The SOAP client HTML page is displayed.

3. Select a SOAP method from the list in the first field. After you select a method, the other fields fill in automatically.
4. Modify, if needed, the tags or the text in the "Edit Payload (XML)" area.
5. Click **Make SOAP Request**. The output of the request displays in the Your SOAP Response Payload area.

What to do next

When issuing a CT_Get request against a particular agent type, the monitoring server where the SOAP server is running must be configured and have the application support for that agent type. For example, when issuing a CT_Get request for a z/OS agent connected to an z/OS monitoring server, the monitoring server running the SOAP server must be configured and have the application support for that z/OS agent.

Using the SOAP client command-line utility (kshsoap)

The SOAP client command-line utility, kshsoap, is an http client. It issues direct SOAP requests. It does this by sending the SOAP request you specified in two text files and by displaying the output of your SOAP request at the command prompt.

Windows systems

Make a SOAP request and send it through the HTTP server with the kshsoap command-line utility in Windows.

About this task

Complete these steps to create a SOAP request file and a SOAP URL receiver file and send the request.

1. On the monitoring server system where the SOAP server is installed, change to the `<install_dir>\cms` directory.
2. Create a text file named SOAPREQ.txt and type the following SOAP request:

```
<CT_Get><object>ManagedSystem</object></CT_Get>
```

or, if security has been enabled:

```
<CT_Get><userid>logonid</userid><password>password</password><object>ManagedSystem</object></CT_Get>
```

3. Create another text file named `URLS.txt` containing the URLs to receive the SOAP request. In this example, **affiliatecompanylocalhost** is the name of the receiving system and where the hub monitoring server is installed:
`http://affiliatecompanylocalhost:1920///cms/soap`
4. At the command prompt, enter **kshsoap SOAPREQ.txt URLS.txt**

Results

The **kshsoap** utility processes the `SOAPREQ` file and displays the URL destination and request. It sends the SOAP request to each URL listed in the `URLS` file, then displays the URL and the response message received.

Systems with APPN installed

About this task

When running the **kshsoap** command on systems that have APPN installed, you might encounter an error message stating that an APPN file needs to be configured. To resolve this situation, modify the environment variable **KDE_WAPPC32**.

To modify this variable, from the command prompt window that you are going to run the **kshsoap** command in, enter this:

```
SET KDE_WAPPC32=none
```

UNIX systems

About this task

To invoke the SOAP command-line utility in UNIX, run the `CandleSOAPClient` script located in the `$install_dir\bin` directory. The `CandleSOAPClient` takes the same parameters as described above for running `kshsoap.exe` on Windows systems.

Issuing SOAP requests as system commands

In Tivoli Enterprise Portal you can use the Take Action feature to issue SOAP requests as system commands in policies or in situations. The SOAP requests are stored in a text file.

In Tivoli Enterprise Portal, you can issue a SOAP request in a situation using the Action tab of the Situation Editor, or in a policy using the Take action or Write message activity of the Workflow editor. See the *IBM Tivoli Monitoring: CandleNet Portal User's Guide* for more on the Situation editor and Workflow editor.

The soap command is:

```
soap:CT_Execute,filename=SOAPREQ
```

where:

- **CT_Execute** is the name of the SOAP method that allows you to run a SOAP request that is stored in a file

- **SOAPREQ** is the name of the file you created that contains the CT_EMail SOAP request.

For example, SOAPREQ might contain:

```
<CT_EMail><server>n-smtpmta</server>
<sender>soap@ibm.com</sender>
<receiver>jane_brown@ibm.com</receiver>
<subject>AFDATA untouched by human hands</subject>
<attachmenttitle>AFData.htm</attachmenttitle>
<request><attach>res.pfx</attach></request>
<request id="XMLID">
<CT_Redirect endpoint="http://sp22.ibm.com:18882">
<SOAP-ENV:Envelope xmlns:SOAP-ENV=
"http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" >
<SOAP-ENV:Body><AF_Execute><Exec>SOAP0002</Exec></AF_Execute></SOAP-ENV:Body>
</SOAP-ENV:Envelope></CT_Redirect></request>
<request><attach>res.sfx</attach></request></CT_EMail>
```

SOAP methods

Each SOAP method provided by IBM and its supported tags is described below.

Table 48. Predefined SOAP Methods

SOAP method	Supported tags	SOAP tag usage examples
CT_Acknowledge Send an event acknowledgement into the IBM Tivoli Monitoring platform.	<p><name>The name of the situation. This is required.<source> The source of the event (agent name or monitoring server name). The acknowledge applies to all the active sources of the named alert if the source is not supplied. <data> "No data was provided" is inserted if not provided. <item>Display item.</p> <p>Optional:<userid> The user ID to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided. <password> The password to access the hub monitoring server. Required for monitoring server/hub logon validation.<type> specifies the event type (sampled by default). The value can be "sampled" or "0", "pure" or "1", and "meta" or "2". <hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub.<expire> Expires the acknowledgement after the number of minutes entered here.</p>	<pre><CT_Acknowledge> <hub>z/OSPROD</hub><name>situation_from_CT </name> <source>CT_supported_system </source><data>Jack is taking care of this failure</data> <item>subsystem</item><userid>sysadmin</userid> <password>xxxxxxx</password><type>pure</type> </CT_Acknowledge><expire>60</expire></pre>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
CT_Activate Start a situation or a policy running on the IBM Tivoli Monitoring platform. Note that situations for agents connecting to a remote Tivoli Enterprise Monitoring Server cannot be started using this method.	<p><name> The name of the situation or policy. This is a required tag. <type> The type of object being activated. This tag is required. <userid> The user ID to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided. <password> The password to access the hub monitoring server. Required for monitoring server/hub logon validation. Optional:<hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub.</p>	<pre><CT_Activate> <hub>z/OSPROD</hub> <name>name_of_situation_or_policy</name><type> situation</type> <userid>sysadmin</userid><password>xxxxxxx</password> </CT_Activate></pre>
CT_Alert Send an event into the IBM Tivoli Monitoring platform.	<p><name> The name of the situation. This is required. <source> The source of the event (agent name or monitoring server name). This is a required tag. <data>No data was provided is inserted if not provided or if no optional object.attribute tag provided. <item>Display item. Optional:<userid> The user ID to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided. <password> The password to access the hub monitoring server. Required for monitoring server/hub logon validation. <hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub.</p> <p><data><object.attribute> Returns the value of the attribute (or attributes) specified to the Initial Attributes view of the Event results workspace. <type> specifies the event type (default is sampled). The value can be "sampled" or "0", "pure" or "1", and "meta" or "2".</p>	<pre><CT_Alert><hub>z/OSPROD</hub> <name>situation_from_XXX </name><source>XXX_supported_system </source><data><NT_Logical_Disk.Disk_Name> C:</NT_Logical_Disk.Disk_Name> </data><item>subsystem</item><userid>sysadmin</userid> <password>xxxxxxx</password></CT_Alert></pre> <p>Note: When you specify object.attribute in the data tag, leave out any non-alphanumeric characters other than the underscore (_). For example, NT_System.%_Total_Processor_Time is entered as NT_System.Total_Processor_Time.</p>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
CT_Deactivate Stop a situation or policy on the IBM Tivoli Monitoring platform. Note: Situations for agents connecting to a remote Tivoli Enterprise Monitoring Server cannot be stopped with this method.	<name> The name of the situation or policy. This is required.<type> The type of object (situation or policy). This is required.<userid>The user ID to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided.<password> The password to access the hub monitoring server. Required for monitoring server/hub logon validation. Optional: <hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub.	<CT_Deactivate> <hub>z/OSPROD</hub><name>name_of_situation_or_policy</name><type>situation</type><userid>sysadmin</userid><password>xxxxxxx</password></CT_Deactivate>
CT_EMail Send the output from another CT SOAP method, such as CT_Get, using e-mail through an SMTP server to a defined e-mail address.(not available on z/OS)	<server> smtp server name/network address is required.<sender> Sender's e-mail address is required.<receiver> Receiver's e-mail address is required.<subject> E-mail subject is optional.<message> E-mail message is optional.<attachmenttitle> Attachment title is optional.<request> When specifying a second-level request, such as CT_Get, each sub-request must be included within a <request> </request> tag. Optional: An id=" " element added to the <request> tag generates a <request id="XMLID"> element enclosing the corresponding response for that sub-request.	<CT_EMail> <server>smtp.server</server><sender>myemail@something.com</sender> <receiver>youremail@whatever.com</receiver> <subject>Here's your data.</subject><message>Table data supplied as attachment below. It is presented in csv format to be used by MS/Excel.</message><attachmenttitle>tabledata.csv</attachmenttitle> <request id="XMLID"><CT_Get><object>NT_Process</object><target>TIPrimary:DCSQLSERVER:NT</target><userid>sysadmin</userid><password>xxxxxxx</password></CT_Get></request></CT_EMail>
CT_Execute Runs the SOAP request that is stored in a file.	<filename> is required and specifies the file name that contains the SOAP request to be run.The file must reside in the \html directory. On z/OS, it must reside in RKANDATV.	<CT_Execute><filename>execute1.xml</filename></CT_Execute>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
CT_Export Send the output from another CT SOAP method, such as CT_Get, to a defined file.(not available on z/OS)	<p><filename> The name of the file to contain the exported data. This is a required tag.</p> <p>Note: When inserting the file name tag into a quoted string literal of certain programming languages, such as C++, back slashes need to be doubled.</p> <p><warehouse/> Specifies that data is to be exported to the Tivoli Enterprise Portal data warehouse through ODBC.<filename> and <warehouse/> are mutually exclusive, but one must be supplied.<request> When specifying a second-level request, such as CT_Get, each sub-request must be included within a <request> </request> tag.</p> <p>Optional: An id=" " element added to the <request> tag generates a <request id="XMLID"> element enclosing the corresponding response for that sub-request.</p>	<pre><CT_Export><filename>g:\exchange\excel\ ntprocess\$yymmddhhmmss\$.htm</filename> <request><attach>prefix.xml</attach> </request><request id="XMLID"><CT_Get><object>NT_Process</ object><target>Primary:DCSQLSE RVER:NT</target><userid>sysadmin</ userid><password>xxxxxxx</password></ CT_Get></request><request> <attach>suffix.xml</attach> </request></ CT_Export></pre>
CT_Export (continued)	<p>To the <filename> tag, you can add an optional date/time stamp variable. The variable is enclosed in dollar signs (\$) and can contain a combination of yy/mm/dd/hh/mm/ss (for year/month/day/hours/minutes/seconds). The date/time stamp attributes can be specified in any order, except mm must be preceded by yy or hh to identify it as either month (after year) or minutes (after hours).</p>	<pre><filename>g:\exchange\excel\ ntprocess\$yymmdd\$.htm</filename></pre>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
<p>CT_Get Receive a group of XML objects or individual XML objects from any IBM Tivoli Monitoring platform agent. You can use this to obtain real time data.</p> <p>Important: When issuing a CT_Get request against a particular agent type, the monitoring server where the SOAP server is running must be configured and seeded for that agent type.</p>	<p><object> The name of the object to be retrieved. Required (by default, retrieves all the public elements of an object).<userid>The user ID to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided. <password>The password to access the hub monitoring server. Required for monitoring server/hub logon validation.</p> <p>Optional: <target> Name of the agent.</p> <p>Caution: Defaults to "*ALL". Retrieves all available targets.<history>Y retrieves historical data if available.<results>PARSE retrieves status history event attributes. Only valid for Status_History object.Multiple: more than one can be specified.<attribute>Attribute name of object. This tag can be specified multiple times.hub> Specifies the alias name of a remote hub that has been configured in the hub's list. The SOAP request is routed to this hub.</p>	<pre><CT_Get> <hub>z/OSPROD</hub><object> NT_System</object><target> Primary:DCSQLSERVER:NT </target><userid>sysadmin</userid> <password></password> <history>Y</history><attribute>Server_Name</ attribute> <attribute>Processor_Queue_Length</attribute> Note: When you specify an attribute in the attribute tags, leave out any non-alphanumeric characters other than the underscore (_). For example, %_Total_User_Time is entered as Total_User_Time.</pre>
CT_Get (continued)	<p><afilter> Returns rows meeting filter criteria, such as attribute; operator; value operators: EQ, NE, GE, GT, LE, LT, LIKE. Like pattern characters: '%' matches any single character. '*' matches one to many characters. Only supported for character attributes. Multiple afilters are only supported as conjuncts, for example, using AND to join together.</p>	<pre><afilter>Write_Time;GT;1020804</afilter> <afilter>Write_Time;LT;1020805</afilter> </CT_Get></pre>
<p>CT_RedirectReroute a SOAP request to another registered SOAP method outside of the domain of the IBM Tivoli Monitoring platform.</p>	<p><request endpoint=" "> This is a required tag.The <request endpoint=" "> value must specify the target of the redirected SOAP request. The entire XML supplied as the value of the request element is sent to that endpoint. When CT_Redirect is specified within a second- level request, such as, CT_Export, the <endpoint=" "> attribute is specified <i>only</i> within the CT_Redirect method.</p>	<pre><CT_Redirect><request endpoint= \"http://services.xmethods.net:80/soap/servlet/ rpcrouter\"><SOAP-ENV:Envelope xmlns:SOAP-ENV=\"http:// schemas.xmlsoap.org/soap/envelope/\"><SOAP- ENV:Body><ns1:getTemp xmlns:ns1=\\ \"urn:xmethods-Temperature\"SOAP- ENV:encodingStyle=\\\"http:// schemas.xmlsoap.org/soap/encoding/\\ \"><zipcode>93117</zipcode> </ns1:getTemp> </SOAP-ENV:Body> </SOAP-ENV:Envelope> </request> </CT_Redirect></pre>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
CT_Reset Send an event reset (close event) to the IBM Tivoli Monitoring platform.	<p><name> The name of the situation. This is a required tag.<source> The source of the event (agent name or monitoring server name). The reset applies to all the active sources of the named alert if the source is not supplied.<item>Display item.</p> <p>Optional:<userid> The user ID used to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided.<password> The password used to access the hub monitoring server. Required for monitoring server/hub validation. <hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub.<type> Specifies the event type (default is sampled). The value can be "sampled" or "0", "pure" or "1", and "meta" or "2".</p>	<pre><CT_Reset> <hub>z/OSPROD</hub> <name>situation_from_CT </name><source> CT_supported_system </source><item>subsystem</ item><userid>sysadmin</ userid><password>xxxxxxx</password> </CT_Reset></pre> <p>Note: Sampled events can be closed only if the situation has been stopped or deleted. Use the <type> tag if CT_Reset will be closing a pure event.</p>
CT_Resurface Resurface an acknowledged event in the IBM Tivoli Monitoring platform.	<p><name> The name of the situation. This is required.<source> The source of the event (agent name or monitoring server name). The resurface applies to all the active sources of the named alert if the source is not supplied. <item>Display item.</p> <p>Optional:<userid> The user ID used to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided. <password>The password used to access the hub monitoring server. Required for monitoring server/hub validation. <hub> Specifies the alias name of a remote hub that has been configured in the hubs list. The SOAP request is routed to this hub. <type> Specifies the event type (default is sampled). The value can be "sampled" or "0", "pure" or "1", and "meta" or "2".</p>	<pre><CT_Resurface> <hub>z/OSPROD</hub> <name>situation_from_CT </name> <source> CT_supported_system </source> <item>subsystem</item> <userid>sysadmin</ userid> <password>xxxxxxx</password> </CT_Resurface></pre>

Table 48. Predefined SOAP Methods (continued)

SOAP method	Supported tags	SOAP tag usage examples
CT_WTO Send a Universal Message into the IBM Tivoli Monitoring Platform.	<p><code><data></code> The message to be sent. This is required.<code><category></code> This tag is optional/blank is the default. <code><severity></code>This is optional, blank is the default.<code><userid></code> The user ID used to access the hub monitoring server. "nnn.nnn.nnn.nnn" is inserted if not provided.<code><password></code> The password used to access the hub monitoring server. Required for monitoring server/hub validation.<code><hub></code> Specifies the alias name of a remote hub that has been configured in the hub's list. The SOAP request is routed to this hub.</p>	<pre><CT_WTO><hub>z/OSPROD</hub><data> This is Universal Message </data><category>Critical Messages </category><severity> High Severity </severity> <userid>sysadmin</ userid><password>xxxxxxx</password></ CT_WTO></pre>

Issuing second-level requests

Some second-level methods perform a particular function with the data retrieved, using imbedded lower-level methods. CT_Email and CT_Export are second-level methods that perform this function.

The lower-level methods are:

- `<CT_Get>`
- `<CT_Redirect>`
- `<attach>`
- `<insert>`

`<CT_Get>` and `<CT_Redirect>` are used as described in "SOAP methods" on page 240. `<attach>` allows you to load a file. The file must reside in the \ibm\itm\cms\html directory. `<insert>` allows you to load the imbedded text into the retrieved (output) data stream at a point corresponding to its position in the XML request.

The example shows how a second-level request might be used. Running this XML results in a file, named tabledata.htm, to be written with the data from file prefix.xls. This is followed by the imbedded data inside the `<insert>` tag, followed by data from the NT_System object. Since an ID=attribute is included with that request, the data tag `<XML id="NTDATA">` is wrapped around that particular request data. Following that is the response from a redirected SOAP request and the insertion of more file data.

```
<CT_Export>
  <filename>tabledata.htm</filename>
  <request>
    <attach>prefix.xls</attach>
  </request>
  <insert
    <insertelement>
      <insertdata>This data has been inserted complements of CT SOAP server.</insertdata>
    </insertelement>
  </insert>
  <request id="NTDATA">
```

```

<CT_Get>
  <userid>sysadmin</userid>
  <password></password>
  <object>NT_System</object>
  <target>*ALL</target>
</CT_Get>
</request>
<request>
<CT_Redirect endpoint="http://services.xmethods.net:80/soap/servlet/rpcrouter">
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
<ns1:getTemp xmlns:ns1="urn:xmethods-Temperature" SOAP-
ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<zipcode>93117</zipcode>
</ns1:getTemp>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
</CT_Redirect>
</request>
<request>
<attach>suffix.xml</attach>
</request>
</CT_Export>

```

Sample CT_Get SOAP request

The attached table shows a sample **CT_Get** SOAP request submitted and the response received.

Table 49. Example of CT_Get SOAP Request sent/data Received

SOAP Request sent to SOAP Endpoint, http://esada.ibm.com:19221/SOAP	SOAP Response from SOAP Endpoint, http://esada.ibm.com:19221/SOAP
<pre> <?xml.version="1.0" encoding="UTF-8" standalone="no"?> <SOAP- ENV:Envelope xmlns:SOAP- ENV="http:// schemas.xmlsoap.org/soap/ envelope/"> <SOAP- ENV:Body><CT_GET xmlns:m="http://ibm.com/Soap" SOAP-ENV:encodingStyle=""> <CT_Get> <Object>NT_System</ Object> <Source>Primary:ESADA:NT</ Source> </CT_Get> </SOAP-ENV:Body> </SOAP-ENV:Envelope> </pre>	<pre> <?xml version="1.0" encoding="ISO-8859-1"?> <SOAP-ENV:Envelope xmlns:SOAP-ENV="http:// schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http:// schemas.xmlsoap.org/soap/encoding/"> <SOAP-ENV:Body><SOAP-CHK:Success xmlns:SOAP-CHK = "http://soaptest1/soaptest/ "><PARMS> </PARMS><TABLE name="KNT.WTSYSTEM"> <OBJECT>NT_System</ OBJECT> <DATA> <ROW> <Server_Name >Primary:ESADA:NT</Server_Name> <Timestamp >1011127123323391</Timestamp> <User_Name >SYSTEM</User_Name> <Operating_System_Type >Windows_NT</Operating_System_Type> <Operating_System_Version >4.0</ Operating_System_Version> <Network_Address >10.21.2.154</Network_Address> <Number_of_Processors dt:dt="number">1</ Number_of_Processors> <Processor_Type dt:dt="number">586</Processor_Type> <Page_Size dt:dt="number">4096</Page_Size> <_Total_Privileged_Time dt:dt="number">1</ _Total_Privileged_Time> <_Total_Processor_Time dt:dt="number">7</_Total_Processor_Time> </pre>

Table 49. Example of CT_Get SOAP Request sent/data Received (continued)

SOAP Request sent to SOAP Endpoint, http://esada.ibm.com:19221/SOAP	SOAP Response from SOAP Endpoint, http://esada.ibm.com:19221/SOAP
	<pre> <_Total_User_Time dt:dt="number">6</ _Total_User_Time> <Context_Switches_Sec dt:dt="number">1745</Context_Switches_Sec> <File_Control_Bytes_Sec dt:dt="number">4500</ File_Control_Bytes_Sec> <File_Control_Operations_Sec dt:dt="number">98</File_Control_Operations_Sec> <File_Data_Operations_Sec dt:dt="number">28</ File_Data_Operations_Sec> <File_Read_Bytes_Sec dt:dt="number">800</File_Read_Bytes_Sec> <File_Read_Operations_Sec dt:dt="number">27</ File_Read_Operations_Sec> <File_Write_Bytes_Sec dt:dt="number">9772</File_Write_Bytes_Sec> <File_Write_Operations_Sec dt:dt="number">1</ File_Write_Operations_Sec> <Processor_Queue_Length dt:dt="number">0</Processor_Queue_Length> <System_Calls_Sec dt:dt="number">2368</ System_Calls_Sec> <System_Up_Time dt:dt="number">956388</System_Up_Time> <Total_Interrupts_Sec dt:dt="number">1076</ Total_Interrupts_Sec> </ROW> </DATA></TABLE> </SOAP-CHK:Success></SOAP-ENV:Body></SOAP- ENV:Envelope> </pre>

IBM Tivoli Monitoring Web services scenarios

Here are a few examples of how you might use IBM Tivoli Monitoring Web services. You can use these examples as suggestions for creating your own applications.

Note: These scenarios do not describe the actual code that was used to develop them. To produce the charts and tables shown in these examples, you need to develop your own scripts.

Generating daily logical operation summaries and charts

You can retrieve data from multiple agents, using the SOAP server against a live hub, to generate daily logical operation summaries. You can use the **CT_EMail** SOAP method to e-mail these summaries to management. You might want to add an **<insert>** tag into **CT_EMail**. This tag contains instructions for the preferred format for the summaries. Management can view these summaries at their desktops using Internet Explorer. Summaries provide an efficient and speedy look at problems that might have occurred during the night.

In addition to the general features, you might add to tables and charts:

- Transaction volumes/response times and whether they are meeting service levels can be plotted with respect to resource trends and error conditions.
- Charts can be plotted over multiple segments, making them easier to view and to print.
- The X-axis can use a variable scale to show the prime shift in greater detail.
- Multiple objects/attributes can be plotted from multiple sources and exceptions can be correlated by time, providing focus on problem areas.

Server Name	Total Processes	Total Context Switches	File Read Operations	File Writes
Primary-SM001-NT	3333	117	4	
Primary-TOR02-NT	2931	104	1691	
Primary-TOR02-NT	2036	39	3487	
Primary-TAD01-NT	4234	18	2503	
Primary-SM002-NT	4744	12	27	
Primary-STO02-NT	4894	26	3842	
Primary-SM001-NT	3994	10	1051	
Primary-SM002-NT	3479	20	2090	
Primary-RES01-NT	1761	74	2	
Primary-PR001-NT	2904	0	0	
Primary-PR002-NT	4222	4	179	

Figure 5. Data Snapshot Table

Sending alerts into an IBM Tivoli Monitoring platform

Using SOAP method **CT_Alert**, you can send a new alert into an IBM Tivoli Monitoring platform. For example, System Automation for Integrated Operations Management detects a problem on a HP NonStop Kernel system and generates an alert within an IBM Tivoli Monitoring platform. The IBM Tivoli Monitoring platform then displays alert information from the HP NonStop Kernel platform.

Collaborative automation using SA IOM

You can create a System Automation for Integrated Operations Management REXX application that calls JSCRIPT SOAP functions to forward any SA IOM trapped message and display it on a Universal Message console. You can use SA IOM scripts to trap and send any log messages, console messages, and so on, to IBM Tivoli Monitoring using SOAP methods.

You can create an application that provides these benefits:

- You can monitor devices, such as HP NonStop Kernel, by trapping VT100 messages and raising Universal Messages.
- You can send commands to SA IOM monitored Telnet sessions and send replies back to those commands.
- Source messages can be either excluded or included, based on any criteria using powerful regular expressions.
- A local log can keep audit information about the status of messages received and messages sent.
- A local log can keep information about the source hub connection/retry status.

The graphics that follow show a sample Telnet session, a Universal Message console showing messages received, and a sample message log.

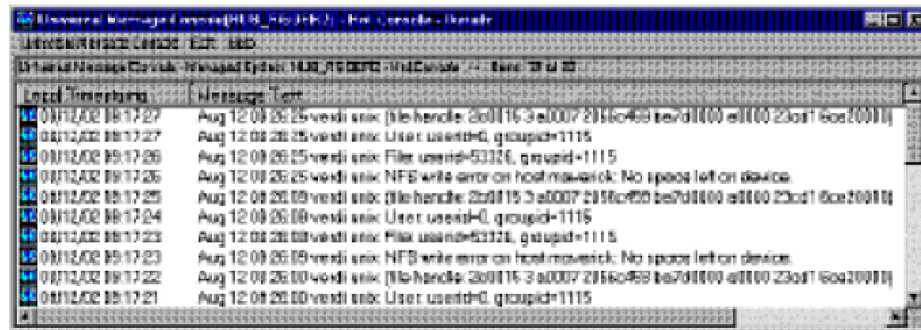


Figure 6. Universal Message Console Showing Messages Received

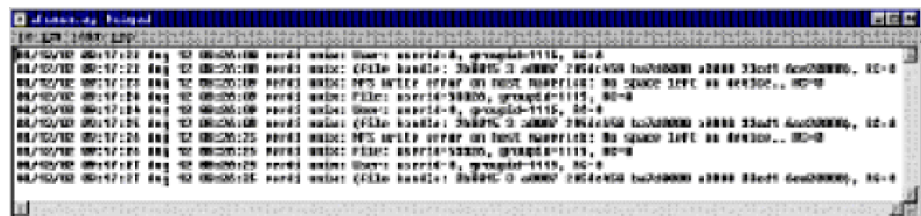


Figure 7. Message Log Details

Acknowledging an event within an IBM Tivoli Monitoring platform

You can acknowledge an event within the IBM Tivoli Monitoring platform. For example, in AF/Operator or System Automation for z/OS V3.2 (or higher):

1. a situation event is received from the hub Tivoli Enterprise Monitoring Server
2. a responsible party is paged who, in turn, sends back an acknowledgement
3. the acknowledgment of the alert is forwarded to the monitoring server

To accomplish this task, use the **CT_Acknowledge** SOAP method. This method enables you to control events in the IBM Tivoli Monitoring environment based upon information obtained and detected by IBM's automation solutions.

Report contents

You can design a report to contain both a table and a chart view. You might want to add a **Table/Chart** button that allows you to toggle between the chart and the table view.

Chart view features

Charts can have specific features to enable you to:

- View different types of charts, depending upon the data retrieved
- Choose the Y-axis by selecting additional attributes from the drop-down attribute list
- Change the title and instructions for the chart
- View the flyover text containing the name and value of the attribute plotted by placing your mouse over each plotted item

Table view features

Tables can have specific features. For example, you can design tables that allow you to:

- View the flyover text containing the name and value of the attribute plotted by placing your mouse over each plotted item
- Modify the table by filtering the attributes that display
- Remove attributes from a table by clicking the X button next to the attribute name

Appendix B. Using the Tivoli Management Services Discovery Library adapter

Tivoli Management Services includes a Discovery Library Adapter (DLA) program for scanning your IBM Tivoli Monitoring environment to identify the managed systems. You can then feed this information (an XML output file) into the Tivoli Application Dependency Discovery Manager's Change and Configuration Management Database. The DLA identifies all distributed and mainframe monitoring agents defined through the Tivoli Enterprise Portal.

The *tmsdla* program gathers information by querying the hub monitoring server for all managed systems and mapping them to Common Data Model resources based on the agent product code and managed system name format. The queries specified in the XML input file provided by each product are run and the results saved to a single output file.

For example, "IMN1:SYS1:IMS" is the managed system name for an OMEGAMON XE for IMS™ agent. The DLA discovers the following:

- A z/OS computer named "IMN1"
- An IMS subsystem named "SYS1"
- A relationship between the SYS1 z/OS computer and IMN1 IMS

For agents that use IP, IP.PIPE, or IP.SPIPE, the DLA can discover the IP address where the agent is running. As well, the DLA discovers the operating system for the computer where the agent is running, regardless of whether an OS monitoring agent is running on that computer.

Note: The monitoring servers and portal server must be running for these queries. Also, any managed systems that are not online will be ignored.

The DLA is run from the command line on the computer where the portal server is installed. The command is located in the *<itm_install_dir>\CNPS (/cq/bin on Linux or operating systems such as UNIX)* subdirectory. Use the following command:

tmsdla

The DLA generates the XML output file in the *<itm_install_dir>\CNPS\tmsdla* subdirectory on the portal server. The name of this file follows the standard Discovery Library file name format. To use this information in the CMDB, you must transfer the XML file to the Discovery Library File Store and then use the Discovery Library Bulk Loader.

In addition to discovering resources and relationships, the Tivoli Management Services DLA discovers information that the IBM Tivoli Change and Configuration Management Database uses to provide a contextual launch to the Tivoli Enterprise Portal. You can also view the status of the discovered managed systems while in IBM Tivoli Change and Configuration Management Database.

For more information about the Tivoli Change and Configuration Management Database, see the Application Dependency Discovery Manager documentation on the IBM IT Service Management information center.

Documentation library

This appendix contains information about the publication related to IBM Tivoli Monitoring and to the commonly shared components of Tivoli Management Services.

These publications are listed in the following categories:

- IBM Tivoli Monitoring library
- Related publications

See the *IBM Tivoli Monitoring Documentation Guide* for information about accessing and using the publications. You can find the *Documentation Guide* in the IBM Tivoli Monitoring and OMEGAMON XE Information Center at <http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp>. To open the *Documentation Guide* in the information center, select **Using the publication** in the **Contents** pane.

To find a list of new and changed publication, click **What's new** on the Welcome page of the IBM Tivoli Monitoring and OMEGAMON XE Information Center. To find publications from the previous version of a product, click **Previous versions** under the name of the product in the **Contents** pane.

IBM Tivoli Monitoring library

The following publications provide information about IBM Tivoli Monitoring and about the commonly shared components of Tivoli Management Services:

- *Quick Start Guide*, GI11-8058
Introduces the components of IBM Tivoli Monitoring.
- *Installation and Setup Guide*, GC32-9407
Provides instructions for installing and configuring IBM Tivoli Monitoring components on Windows, Linux, and UNIX systems.
- *Program Directory for IBM Tivoli Management Services on z/OS*, GI11-4105
Gives instructions for the SMP/E installation of the Tivoli Management Services components on z/OS.
- *Configuring the Tivoli Enterprise Monitoring Server on z/OS*, SC32-9463
Gives detailed instructions for using the Configuration Tool to configure Tivoli Enterprise Monitoring Server on z/OS systems. Includes scenarios for using batch mode to replicate monitoring environments across the z/OS enterprise. Also provides instructions for setting up security and for adding application support to a Tivoli Enterprise Monitoring Server on z/OS.
- *Administrator's Guide*, SC32-9408
Describes the support tasks and functions required for the Tivoli Enterprise Portal Server and clients, including Tivoli Enterprise Portal user administration.
- *High-Availability Guide for Distributed Systems*, SC23-9768
Gives instructions for several methods of ensuring the availability of the IBM Tivoli Monitoring components.
- Tivoli Enterprise Portal online help

Provides context-sensitive reference information about all features and customization options of the Tivoli Enterprise Portal. Also gives instructions for using and administering the Tivoli Enterprise Portal.

- *Tivoli Enterprise Portal User's Guide*, SC32-9409
Complements the Tivoli Enterprise Portal online help. The guide provides hands-on lessons and detailed instructions for all Tivoli Enterprise Portal features.
- *Command Reference*, SC32-6045
Provides detailed syntax and parameter information, as well as examples, for the commands you can use in IBM Tivoli Monitoring.
- *Troubleshooting Guide*, GC32-9458
Provides information to help you troubleshoot problems with the software.
- *Messages*, SC23-7969
Lists and explains messages generated by all IBM Tivoli Monitoring components and by z/OS-based Tivoli Management Services components (such as Tivoli Enterprise Monitoring Server on z/OS and TMS:Engine).
- *IBM Tivoli Universal Agent User's Guide*, SC32-9459
Introduces you to the IBM Tivoli Universal Agent, an agent of IBM Tivoli Monitoring. The IBM Tivoli Universal Agent enables you to use the monitoring and automation capabilities of IBM Tivoli Monitoring to monitor any type of data you collect.
- *IBM Tivoli Universal Agent API and Command Programming Reference Guide*. SC32-9461
Explains the procedures for implementing the IBM Tivoli Universal Agent APIs and provides descriptions, syntax, and return status codes for the API calls and command-line interface commands.
- *Agent Builder User's Guide*, SC32-1921
Explains how to use the Agent Builder for creating monitoring agents and their installation packages, and for adding functions to existing agents.

Documentation for the base agents

If you purchased IBM Tivoli Monitoring as a product, you received a set of base monitoring agents as part of the product. If you purchased a monitoring agent product (for example, an OMEGAMON XE product) that includes the commonly shared components of Tivoli Management Services, you did not receive the base agents.

The following publications provide information about using the base agents.

- Operating system agents:
 - *Windows OS Agent User's Guide*, SC32-9445
 - *UNIX OS Agent User's Guide*, SC32-9446
 - *Linux OS Agent User's Guide*, SC32-9447
 - *i5/OS Agent User's Guide*, SC32-9448
 - *UNIX Log Agent User's Guide*, SC32-9471
- Agentless operating system monitors:
 - *Agentless Monitoring for Windows Operating Systems User's Guide*, SC23-9765
 - *Agentless Monitoring for AIX Operating Systems User's Guide*, SC23-9761
 - *Agentless Monitoring for HP-UX Operating Systems User's Guide*, SC23-9763
 - *Agentless Monitoring for Solaris Operating Systems User's Guide*, SC23-9764

- *Agentless Monitoring for Linux Operating Systems User's Guide*, SC23-9762
- Warehouse agents:
 - *Warehouse Summarization and Pruning Agent User's Guide*, SC23-9767
 - *Warehouse Proxy Agent User's Guide*, SC23-9766
- System P agents:
 - *AIX Premium Agent User's Guide*, SA23-2237
 - *CEC Base Agent User's Guide*, SC23-5239
 - *HMC Base Agent User's Guide*, SA23-2239
 - *VIOS Premium Agent User's Guide*, SA23-2238
- Other base agents:
 - *Monitoring Agent for IBM Tivoli Monitoring 5.x Endpoint User's Guide*, SC32-9490

Related publications

You can find useful information about related products in the IBM Tivoli Monitoring and OMEGAMON XE Information Center at <http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp>.

Other sources of documentation

You can also obtain technical documentation about IBM Tivoli Monitoring and related products from the following sources:

- IBM Tivoli Open Process Automation Library (OPAL)

<http://www.ibm.com/software/tivoli/opal>

OPAL is an online catalog that contains integration documentation and other downloadable product extensions.
- Redbooks

<http://www.redbooks.ibm.com/>

IBM Redbooks and Redpapers include information about products from platform and solution perspectives.
- Technotes

Technotes provide the latest information about known product limitations and workarounds. You can find Technotes through the IBM Software Support Web site at <http://www.ibm.com/software/support/probsub.html>, or more directly through your product Web site, which contains a link to Technotes (under **Solve a problem**).
- Tivoli wikis on the IBM developerWorks Web site

Tivoli Wiki Central at <http://www.ibm.com/developerworks/wikis/display/tivoli/Home> is the home for interactive wikis that offer best practices and scenarios for using Tivoli products. The wikis contain white papers contributed by IBM employees, and content created by customers and business partners. Two of these wikis are of particular relevance to IBM Tivoli Monitoring:

 - Tivoli Distributed Monitoring and Application Management Wiki at <http://www.ibm.com/developerworks/wikis/display/tivolimonitoring/Home> provides information about IBM Tivoli Monitoring and related distributed products, including IBM Tivoli Composite Application Management products.
 - Tivoli System z Monitoring and Application Management Wiki at <http://www.ibm.com/developerworks/wikis/display/tivoliomegamon/>

Home provides information about the OMEGAMON XE products, NetView for z/OS, Tivoli Monitoring Agent for z/TPF, and other System z monitoring and application management products.

Support information

If you have a problem with your IBM® software, you want to resolve it quickly. IBM provides ways for you to obtain the support you need.

Online

Go to the IBM Software Support site at <http://www.ibm.com/software/support/probsub.html> and follow the instructions.

IBM Support Assistant

The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, go to <http://www.ibm.com/software/support/isa>.

Troubleshooting Guide

For more information about resolving problems, see the product's Troubleshooting Guide.

Using IBM Support Assistant

The IBM Support Assistant is a free, stand-alone application that you can install on any workstation. You can then enhance the application by installing product-specific plug-in modules for the IBM products you use.

The IBM Support Assistant saves you the time it takes to search the product, support, and educational resources. The IBM Support Assistant helps you gather support information when you need to open a problem management record (PMR), which you can then use to track the problem.

The product-specific plug-in modules provide you with the following resources:

- Support links
- Education links
- Ability to submit problem management reports

For more information, and to download the IBM Support Assistant, see <http://www.ibm.com/software/support/isa>. After you download and install the IBM Support Assistant, follow these steps to install the plug-in for your Tivoli product:

1. Start the IBM Support Assistant application.
2. Select **Updater** on the Welcome page.
3. Select **New Properties and Tools** or select the **New Plug-ins** tab (depending on the version of IBM Support Assistant installed).
4. Under **Tivoli**, select your product, and then click **Install**. Be sure to read the license and description.

If your product is not included on the list under **Tivoli**, no plug-in is available yet for the product.
5. Read the license and description, and click **I agree**.
6. Restart the IBM Support Assistant.

Obtaining fixes

A product fix might be available to resolve your problem. To determine which fixes are available for your Tivoli software product, follow these steps:

1. Go to the IBM Software Support Web site at <http://www.ibm.com/software/support>.
2. Under **Select a brand and/or product**, select **Tivoli**.
If you click **Go**, the **Search within all of Tivoli support** section is displayed. If you don't click **Go**, you see the **Select a product** section.
3. Select your product and click **Go**.
4. Under **Download**, click the name of a fix to read its description and, optionally, to download it.
If there is no **Download** heading for your product, supply a search term, error code, or APAR number in the field provided under **Search Support (this product)**, and click **Search**.

For more information about the types of fixes that are available, see the *IBM Software Support Handbook* at <http://techsupport.services.ibm.com/guides/handbook.html>.

Receiving weekly support updates

To receive weekly e-mail notifications about fixes and other software support news, follow these steps:

1. Go to the IBM Software Support Web site at <http://www.ibm.com/software/support>.
2. Click **My support** in the far upper-right corner of the page under **Personalized support**.
3. If you have already registered for **My support**, sign in and skip to the next step. If you have not registered, click **register now**. Complete the registration form using your e-mail address as your IBM ID and click **Submit**.
4. The **Edit profile** tab is displayed.
5. In the first list under **Products**, select **Software**. In the second list, select a product category (for example, **Systems and Asset Management**). In the third list, select a product sub-category (for example, **Application Performance & Availability** or **Systems Performance**). A list of applicable products is displayed.
6. Select the products for which you want to receive updates.
7. Click **Add products**.
8. After selecting all products that are of interest to you, click **Subscribe to email** on the **Edit profile** tab.
9. In the **Documents** list, select **Software**.
10. Select **Please send these documents by weekly email**.
11. Update your e-mail address as needed.
12. Select the types of documents you want to receive.
13. Click **Update**.

If you experience problems with the **My support** feature, you can obtain help in one of the following ways:

Online

Send an e-mail message to erchelp@ca.ibm.com, describing your problem.

By phone

Call 1-800-IBM-4You (1-800-426-4968).

Contacting IBM Software Support

IBM Software Support provides assistance with product defects. The easiest way to obtain that assistance is to open a PMR or ETR directly from the IBM Support Assistant.

Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have:

- For IBM distributed software products (including, but not limited to, Tivoli, Lotus, and Rational products, as well as DB2 and WebSphere products that run on Windows or UNIX operating systems), enroll in Passport Advantage in one of the following ways:

Online

Go to the Passport Advantage Web site at http://www-306.ibm.com/software/howtobuy/passportadvantage/pao_customers.htm.

By phone

For the phone number to call in your country, go to the IBM Software Support Web site at <http://techsupport.services.ibm.com/guides/contacts.html> and click the name of your geographic region.

- For customers with Subscription and Support (S & S) contracts, go to the Software Service Request Web site at <https://techsupport.services.ibm.com/ssr/login>.
- For customers with Linux, iSeries, pSeries, zSeries, and other support agreements, go to the IBM Support Line Web site at <http://www.ibm.com/services/us/index.wss/so/its/a1000030/dt006>.
- For IBM eServer software products (including, but not limited to, DB2 and WebSphere products that run in zSeries, pSeries, and iSeries environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM Technical Support Advantage Web site at <http://www.ibm.com/servers/eserver/techsupport.html>.

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States. From other countries, go to the contacts page of the *IBM Software Support Handbook on the Web* at <http://techsupport.services.ibm.com/guides/contacts.html> and click the name of your geographic region for phone numbers of people who provide support for your location.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Readers' Comments — We'd Like to Hear from You

IBM Tivoli Monitoring
Administrator's Guide
Version 6.2.2

Publication No. SC32-9408-03

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

E-mail address



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape

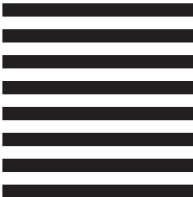


NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line



Printed in USA

SC32-9408-03

