IBM Tivoli Composite Application Manager for Microsoft Applications: Microsoft SQL Server Agent

Version 6.3.1 Fix Pack 10

Installation and Configuration Guide

SC27-5660-01

IBM Tivoli Composite Application Manager for Microsoft Applications: Microsoft SQL Server Agent

Version 6.3.1 Fix Pack 10

Installation and Configuration Guide

SC27-5660-01

Note

Before using this information and the product it supports, read the information in “Notices” on page 49.

This edition applies to version 6.3.1.10 of IBM Tivoli Composite Application Manager for Microsoft Applications: Microsoft SQL Server Agent (product number 5724-U17) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Tables

Chapter 1. Overview of the agent

The IBM Tivoli Composite Application Manager for Microsoft Applications: Microsoft SQL Server Agent (product code OQ) provides you with the capability to monitor Microsoft SQL Server. You can also use the agent to take basic actions with the Microsoft SQL Server.

IBM® Tivoli® Monitoring is the base software for the Microsoft SQL Server agent.

The Microsoft SQL Server agent monitors the following functions:
- Availability and resources
- Performance
- Error and event log
- Historical data

IBM Tivoli Monitoring

IBM Tivoli Monitoring provides a way to monitor the availability and performance of all the systems in your enterprise from one or several designated workstations. It also provides useful historical data that you can use to track trends and to troubleshoot system problems.

You can use IBM Tivoli Monitoring to achieve the following tasks:
- Monitor for alerts on the systems that you are managing by using predefined situations or custom situations.
- Establish your own performance thresholds.
- Trace the causes leading to an alert.
- Gather comprehensive data about system conditions.
- Use policies to take actions, schedule work, and automate manual tasks.

The Tivoli Enterprise Portal is the interface for IBM Tivoli Monitoring products. You can use the consolidated view of your environment as seen in the Tivoli Enterprise Portal to monitor and resolve performance issues throughout the enterprise.

See the IBM Tivoli Monitoring publications listed in "Prerequisite publications" on page 47 for complete information about IBM Tivoli Monitoring and the Tivoli Enterprise Portal.

Features of the monitoring agent

The Microsoft SQL Server agent offers a central point of management for distributed databases. The monitoring agent provides a comprehensive means for gathering exactly the information you need to detect problems early and prevent them. Information is standardized across all systems so you can monitor hundreds of servers from a single workstation. You can easily collect and then analyze specific information by using the Tivoli Enterprise Portal interface.

The Microsoft SQL Server agent is an intelligent, remote monitoring agent that is installed on managed systems. It assists you in anticipating trouble and warns system administrators when critical events take place on systems. With the Microsoft SQL Server agent, database and system administrators can set threshold levels and can set flags to alert them when the system reaches these thresholds.

This monitoring agent provides the following benefits:

- Simplifies application and system management by managing applications, operating systems, and resources across your environment.
- Increases profits by providing you with real-time access to reliable, up-to-the-minute data that allows you to make faster, better-informed operating decisions.
- Scales and ports to new Windows operating systems.
- Improves system performance by letting you integrate, monitor, and manage your system, network, console, and mission-critical applications. The monitoring agent sends an alert when conditions on the system network meet threshold-based conditions. These alerts notify your system administrator to limit and control database usage. You can view data gathered by monitoring agents in reports and charts for the status of your distributed database systems.
- Enhances efficiency by monitoring different Microsoft SQL Server versions on separate systems and networks from a single PC screen. Depending on your configuration, you can collect and monitor data across systems. The monitoring agent gathers and filters status information at the managed system rather than at the hub, eliminating unnecessary data transmission and sending only data that is relevant to changes in status conditions. The Microsoft SQL Server agent helps you to monitor and to gather the consistent, accurate, and timely information you need to effectively perform your job.

New in this release

For version 6.3.1.10 of the Microsoft SQL Server agent, the following enhancements have been made since version 6.3.1, including the fix packs:

- Added support to monitor the Microsoft SQL Server 2016
- Modified the threshold value of the MS_SQL_Fragmentation_Crit situation to 90
- Modified attribute groups:
  - MS SQL Blocked Queries Details
  - MS SQL Database Detail
  - MS SQL Device Detail
  - MS SQL Filegroup Detail
  - MS SQL Server Properties
  - MS SQL Services Detail
  - MS SQL Table Detail
- Added the **COLL_TBLD_SORTBY** variable to set the sorting criteria for the rows that are returned by the Table Details attribute group
- Added the **Monitor All Databases** check box in the Database Server Properties window to monitor all the databases of an SQL Server instance.

Components of the IBM Tivoli Monitoring environment

After you install and set up the Microsoft SQL Server agent, you have an environment that contains the client, server, and monitoring agent implementation for Tivoli Monitoring.

This Tivoli Monitoring environment contains the following components:

Tivoli Enterprise Portal client
The portal has a user interface based on Java™ for viewing and monitoring your enterprise.

Tivoli Enterprise Portal Server
The portal server is placed between the client and the Tivoli Enterprise Monitoring Server and enables retrieval, manipulation, and analysis of data from the monitoring agents. The Tivoli Enterprise Portal Server is the central repository for all user data.

Tivoli Enterprise Monitoring Server
The monitoring server acts as a collection and control point for alerts received from the

monitoring agents, and collects their performance and availability data. The Tivoli Enterprise Monitoring Server is also a repository for historical data.

Tivoli Enterprise Monitoring Agent, Microsoft SQL Server agent
This monitoring agent collects data and distributes the data to the Tivoli Enterprise Monitoring Server, Tivoli Enterprise Portal Server, Tivoli Enterprise Portal, Tivoli Data Warehouse, and Tivoli Integrated Portal.

Multiple copies of this agent can run on the same system.

IBM Tivoli Netcool/OMNIbus
Tivoli Netcool/OMNIbus is an optional component and the recommended event management component. The Netcool/OMNIbus software is a service level management (SLM) system that delivers real-time, centralized monitoring of complex networks and IT domain events. Event information is tracked in a high-performance, in-memory database and presented to specific users through individually configurable filters and views. The software includes automation functions that you can use to perform intelligent processing on managed events. You can use this software to forward events for Tivoli Monitoring situations to Tivoli Netcool/OMNIbus.

IBM Tivoli Enterprise Console®
The Tivoli Enterprise Console is an optional component that acts as a central collection point for events from various sources, including events from other Tivoli software applications, Tivoli partner applications, custom applications, network management platforms, and relational database systems. You can view these events through the Tivoli Enterprise Portal (by using the event viewer), and you can forward events from Tivoli Monitoring situations to the Tivoli Enterprise Console component. If you do not already use Tivoli Enterprise Console and need an event management component, you can choose to use IBM Tivoli Netcool/OMNIbus.

IBM Tivoli Common Reporting
Tivoli Common Reporting is a separately installable feature available to users of Tivoli software that provides a consistent approach to generating and customizing reports. Some individual products provide reports that are designed for use with Tivoli Common Reporting and have a consistent look and feel.

IBM Tivoli Application Dependency Discovery Manager (TADDM)
TADDM delivers automated discovery and configuration tracking capabilities to build application maps that provide real-time visibility into application complexity.

IBM Tivoli Business Service Manager
The Tivoli Business Service Manager component delivers real-time information to help you respond to alerts effectively based on business requirements. Optionally, you can use this component to meet service-level agreements (SLAs). Use the Tivoli Business Service Manager tools to help build a service model that you can integrate with Tivoli Netcool/OMNIbus alerts or optionally integrate with data from an SQL data source. Optional components provide access to data from other IBM Tivoli applications such as Tivoli Monitoring and TADDM.

Tivoli Integrated Portal
Tivoli Integrated Portal helps the interaction and secure passing of data between Tivoli products through a common portal. Within the same dashboard view, you can launch from one application to another and research different aspects of your managed enterprise. This component is installed automatically with the first Tivoli product that uses the Tivoli Integrated Portal framework. Subsequent products can install updated versions of Tivoli Integrated Portal. After version 2.2, this component is replaced by the Dashboard Application Services Hub.

User interface options

Installation of the base IBM Tivoli Monitoring software and other integrated applications provides various interfaces that you can use to work with your resources and data.

The following interfaces are available:

Tivoli Enterprise Portal user interface
You can run the Tivoli Enterprise Portal as a desktop application or a browser application. The client interface is a graphical user interface (GUI) based on Java on a Windows or Linux workstation. The browser application is automatically installed with the Tivoli Enterprise Portal Server. The desktop application is installed by using the Tivoli Monitoring installation media or with a Java Web Start application. To start the Tivoli Enterprise Portal browser client in your Internet browser, enter the URL for a specific Tivoli Enterprise Portal browser client installed on your Web server.

Command-line interface
You can use Tivoli Monitoring commands to manage the Tivoli Monitoring components and their configuration. You can also run commands at the Tivoli Enterprise Console event server or the Tivoli Netcool/OMNIbus ObjectServer to configure event synchronization for enterprise situations.

Manage Tivoli Enterprise Monitoring Services window
You can use the window for the Manage Tivoli Enterprise Monitoring Services utility to configure the agent and start Tivoli services not designated to start automatically.

IBM Tivoli Netcool/OMNIbus event list
You can use the Netcool/OMNIbus event list to monitor and manage events. An event is created when the Netcool/OMNIbus ObjectServer receives an event, alert, message, or data item. Each event is made up of columns (or fields) of information that are displayed in a row in the ObjectServer alerts.status table. The Tivoli Netcool/OMNIbus web GUI is also a web-based application that processes network events from one or more data sources and presents the event data in various graphical formats.

IBM Tivoli Enterprise Console
You can use the Tivoli Enterprise Console to help ensure the optimal availability of an IT service for an organization. The Tivoli Enterprise Console is an event management application that integrates system, network, database, and application management. If you do not already use Tivoli Enterprise Console and need an event management component, you can choose to use Tivoli Netcool/OMNIbus.

IBM Tivoli Common Reporting
Use the Tivoli Common Reporting web user interface for specifying report parameters and other report properties, generating formatted reports, scheduling reports, and viewing reports. This user interface is based on the Dashboard Application Services Hub for Tivoli Common Reporting 3.1 and on Tivoli Integrated Portal for earlier versions.

IBM Tivoli Application Dependency Discovery Manager
The Discovery Management Console is the TADDM client user interface for managing discoveries.

IBM Tivoli Business Service Manager
The Tivoli Business Service Manager console provides a graphical user interface that you can use to logically link services and business requirements within the service model. The service model provides an operator with a second-by-second view of how an enterprise is performing at any moment in time or how the enterprise performed over a time period.

Tivoli Integrated Portal
Web-based products that are built on the Tivoli Integrated Portal framework share a common user interface where you can launch applications and share information. After version 2.2, this interface is replaced by the Dashboard Application Services Hub.

Microsoft SQL Server agent data collection

The Microsoft SQL Server agent uses different algorithms to collect data from specific data sources.

The agent collects data in the following ways:

- Data collection at workspace refresh, at situation sampling of attributes, or at regular intervals
- Data collection from the Microsoft SQL Server SELECT statements and stored procedures
- Data collection from operating system or MS SQL Server files

Collection processes

The Microsoft SQL Server agent gathers data when it is requested by a workspace refresh, a situation sampling of attributes, or historical data collection.

The monitoring agent gathers all attributes in the attribute groups that make up a workspace or situation. The default refresh or sampling intervals do not put a significant load on the system or the Microsoft SQL Server as it gathers the data. Most of the attributes gathered by the Microsoft SQL Server agent come from monitoring data provided by Windows Performance Monitor (Perfmon). This data is acquired though the use of requests for Perfmon data and SQL database SELECT statements. The Perfmon and SQL data are gathered on a defined cycle while most other attributes are gathered on demand when requested through a workspace refresh, a situation sampling of attributes, or historical data collection. Other attribute groups contain data that is collected from system information or external Microsoft SQL Server data such as the database instance log. This data is primarily gathered on demand, as it is requested. The Microsoft SQL Server agent attempts to lessen the possibility of over sampling by maintaining a cache of attribute data for a set length of time. If the data is sampled within a shorter duration than the interval set for the cache, the cache data is considered fresh and the collector does not acquire new data.

Table 1 provides information about when the attribute data is gathered (on demand or cyclically) and the cache duration for each Microsoft SQL Server agent attribute group.

Table 1. Data collection for attribute groups

Attribute group	When data is collected (cycle time or on demand)	Cache duration in seconds
MS SQL Audit details	On demand	60
MS SQL Batch Stats	Every 180 seconds	No caching
MS SQL Configuration	On demand	90
MS SQL Customized SQL Query Details	On demand	No caching
MS SQL Customized SQL Query Result	On demand	No caching
MS SQL Database Detail	User-defined interval	User-defined interval
MS SQL Database Mirroring	On demand	10
MS SQL Database Summary	Every 180 seconds	No caching
MS SQL Device Detail	On demand —OR— User-defined interval	60 —OR— User-defined interval
MS SQL Enterprise Log Shipping DB Details	On demand	No caching
MS SQL Filegroup	On demand	No caching
MS SQL Job Detail	Every 180 seconds	No caching
MS SQL Job Summary	On demand	No caching

Table 1. Data collection for attribute groups (continued)

Attribute group	When data is collected (cycle time or on demand)	Cache duration in seconds
MS SQL Lock Conflict Detail	On demand	No caching
MS SQL Lock Detail	On demand	No caching
MS SQL Lock Resource Type Summary	Every 300 seconds	No caching
MS SQL Lock Summary	On demand	No caching
MS SQL Log Shipping DB Details	On demand	No caching
MS SQL Log Shipping Errors	On demand	No caching
MS SQL Log Shipping Jobs Detail	On demand	No caching
MS SQL Memory Manager	On demand	No caching
MS SQL Problem Detail	Every 180 seconds	No caching
MS SQL Process Detail	On demand	10
MS SQL Process Summary	Every 180 seconds	No caching
MS SQL Problem Summary	Every 180 seconds	No caching
MS SQL Remote Servers	Every 60 seconds	No caching
MS SQL Resource Pool Stats	On demand	No caching
MS SQL Server Detail	On demand	60
MS SQL Server Enterprise View	On demand	No caching
MS SQL Server Summary	Every 180 seconds	No caching
MS SQL Services Detail	On demand	No caching
MS SQL Statistics Detail	On demand	300
MS SQL Statistics Summary	Every 60 seconds	No caching
MS SQL Table Detail	On demand —OR— User-defined interval	120 —OR— User-defined interval
MS SQL Text	On demand	No caching
MS SQL Workload Group Stats	On demand	No caching
VMS SQL Remote Servers	On demand	No caching
VMS SQL Server Enterprise View	On demand	No caching
MS SQL Server Transaction Summary	On demand	10
MS SQL Service Broker Activation	On demand	10
MS SQL Service Broker Statistics	On demand	10
MS SQL Service Broker Transport	On demand	10
MS SQL Table Summary	On demand	120
MS SQL Integration Service Details	On demand	No caching
MS SQL Analysis Services Data Mining	On demand	No caching
MS SQL Analysis Services Memory statistics	On demand	No caching
MS SQL Analysis Services Storage Engine Query	On demand	No caching

Table 1. Data collection for attribute groups (continued)

Attribute group	When data is collected (cycle time or on demand)	Cache duration in seconds
MS SQL Analysis Services MDX Processing	On demand	No caching
MS SQL Analysis Services Rows Processing	On demand	No caching
MS SQL Running Queries Details	On demand	30
MS SQL Individual Queries Details	On demand	30
MS SQL Blocked Queries Details	On demand	10
MS SQL FileTable Detail	On demand	120
MS SQL Additional Database Details	On demand	60
MS SQL Availability Groups Details	On demand	60
MS SQL Availability Replicas Details	On demand	60
MS SQL Availability Replicas Status	On demand	60
MS SQL Availability Replicas Status Summary	On demand	60
MS SQL Availability Groups Summary	On demand	60
MS SQL Availability Replicas Statistics	On demand	60
MS SQL Availability Database Details	On demand	60
MS SQL Availability Database Statistics	On demand	60
MS SQL Availability Database Summary	On demand	60
MS SQL Availability Group Listener Details	On demand	60
MS SQL Availability Groups Details In Cluster	On demand	60
MS SQL Availability Replicas Details In Cluster	On demand	60
MS SQL Availability Database Details In Cluster	On demand	60
MS SQL Additional Availability Details	On demand	60
MS SQL Server Properties	On demand	10

The Microsoft SQL Server agent maintains long-running processes for the monitoring agent that communicates with the Tivoli Enterprise Monitoring Server and the collector that drives data collection. Short-running processes are used to access system data, access database instance log file data, issue and process the output of Microsoft SQL Server stored procedures, and perform other database instance interactions.

Data collection from SELECT statements and stored procedures

Some of the attribute groups of Microsoft SQL Server agent gather data that is returned by Microsoft SQL Server SELECT statements, stored procedures, or both.

Table 2 on page 8 provides information about how each attribute group collects data (SELECT statements or stored procedures). The table also provides the following information:

- For attribute groups that use SELECT statements to gather data, the number of individual SELECT statements issued for the group and the tables that are accessed are listed.

- For attribute groups that use stored procedures to gather data, the names of the stored procedures are listed.

Table 2. Key aspects of data gathering for attribute groups

Attribute group	Primary collection mechanism	Number of SELECT statements issued	Tables accessed	Name of stored procedures
MSSQL Audit Details	SELECT	4	sys.server_audit_specifications, sys.server_audit_specification_details, sys.dm_audit_actions, sys.database_audit_specifications, sys.database_audit_specification_details, sys.dm_audit_actions, sys.server_audits, sys.dm_server_audit_status	sp_MSForEachDB
MSSQL Batch Stats	Perfmon	0	None	None
MSSQL Server Summary	SELECT and Perfmon	5	sysprocesses, sysjobhistory	serverproperty ('processid'), DBCC
MSSQL Server Detail	SELECT and Perfmon	5	sysprocesses, sysjobhistory	serverproperty ('processid'), DBCC
MSSQL Database Summary	SELECT and Perfmon	4	sysdatabases, sysfiles	sp_replcounters
MSSQL Database Detail	SELECT and Perfmon	4	sysdatabases, sysfiles	sp_replcounters
MSSQL Database Mirroring	SELECT and Perfmon	2	sys.database_mirroring	None
MSSQL Device Detail	SELECT and Perfmon	4	sysfiles, sysdevices, sysusages, spt_values	None
MSSQL Enterprise Log Shipping DB Details	SELECT	1	log_shipping_monitor_primary, log_shipping_monitor_history_detail, log_shipping_monitor_secondary	None
MSSQL Lock Detail	SELECT	2	syslockinfo, sysobjects	None
MSSQL Lock Conflict Detail	SELECT	1	sysprocesses	None
MSSQL Log Shipping DB Details	SELECT	1	log_shipping_monitor_primary, log_shipping_monitor_history_detail, log_shipping_monitor_secondary	None
MSSQL Log Shipping Errors	SELECT	1	log_shipping_monitor_error_detail	None
MSSQL Log Shipping Jobs Detail	SELECT	1	log_shipping_monitor_history_detail, sysjobhistory	None
MSSQL Text	SELECT	1	sysprocesses	None
MSSQL Process Summary	SELECT	2	sysprocesses	None
MSSQL Process Detail	SELECT	3	sysprocesses	None

Table 2. Key aspects of data gathering for attribute groups (continued)

Attribute group	Primary collection mechanism	Number of SELECT statements issued	Tables accessed	Name of stored procedures
MSSQL Statistics Summary	SELECT	1	@@cpu_busy, @@io_busy, @@idle, @@connections, @@pack_sent, @@pack_received, @@packet_errors, @@total_read, @@total_write, @@total_errors,	None
MSSQL Statistics Detail	SELECT	1	@@cpu_busy, @@io_busy, @@idle, @@connections, @@pack_sent, @@pack_received, @@packet_errors, @@total_read, @@total_write, @@total_errors,	None
MSSQL Remote Servers	SELECT	1	sysservers	None
MSSQL Resource Pool Stats	Perfmon	0	None	None
MSSQL Configuration	SELECT	1	sysconfigurations, sysscurconfigs, sysconfigures	None
MSSQL Server Enterprise View	SELECT	1	sysprocesses, sysjobhistory, syslockinfo	None
MSSQL Lock Summary	SELECT and Perfmon	16	syslockinfo	None
MSSQL Memory Manager	Perfmon	0	None	None
MSSQL Job Summary	SELECT	3	syscategories, sysjobs, sysjobhistory	
MSSQL Job Detail	SELECT	3	syscategories, sysjobs, sysjobhistor	
MSSQL Filegroup Detail	SELECT and Perfmon	2	sysdatabases, sysfiles	None
MSSQL Lock Resource Type Summary	Perfmon	0	None	None
MSSQL Workload Group Stats	Perfmon	0	None	None
MS SQL Server Transaction Summary	Perfmon	0	None	None
MS SQL Service Broker Activation	Perfmon	0	None	None
MS SQL Service Broker Statistics	Perfmon	0	None	None
MS SQL Service Broker Transport	Perfmon	0	None	None
MS SQL Table Summary	SELECT	3	sys.tables, sysindexes, sys.dm_db_ index_physical_stats, sys.dm_db_ partition_stats	None
MS SQL Integration Service Details	Perfmon	0	None	None

Table 2. Key aspects of data gathering for attribute groups (continued)

Attribute group	Primary collection mechanism	Number of SELECT statements issued	Tables accessed	Name of stored procedures
MS SQL Analysis Services Data Mining	Perfmon	0	None	None
MS SQL Analysis Services Memory Statistics	Perfmon	0	None	None
MS SQL Analysis Services Storage Engine Query	Perfmon	0	None	None
MS SQL Analysis Services MDX Processing	Perfmon	0	None	None
MS SQL Analysis Services Rows Processing	Perfmon	0	None	None
MS SQL Running Queries Details	SELECT	1	sys.dm_exec_requests, sys.dm_exec_sessions, sys.dm_exec_sql_text	None
MS SQL Individual Queries Details	SELECT	1	sys.dm_exec_ connections, sys.dm_exec_ requests, sys.dm_exec_sessions, sys.dm_exec_sql_text	None
MS SQL Blocked Queries Details	SELECT	3	sys.dm_os_sys_info, sys.configurations, sys.dm_exec_ query_stats, sys.dm_exec_requests, sys.dm_exec_sql_text, sys.dm_exec_ plan_attributes	None
MS SQL FileTable Detail	SELECT	3	sys.databases, sys.filetables, sysindexes, sys.dm_filestream_non_ transacted_handles	None
MS SQL Additional Database Details	SELECT	1	sys.databases, sys.database_ filestream_options	None
MS SQL Availability Groups Details	SELECT	3	sys.availability_groups, sys.dm_hadr_ availability_group_states, sys.dm_hadr_ database_replica_states, sys.dm_hadr_ availability_replica_ states, sys.availability_replicas	None
MS SQL Availability Replicas Details	SELECT	2	sys.availability_replicas, sys.availability_groups, sys.dm_hadr_ availability_group_ states	None

Table 2. Key aspects of data gathering for attribute groups (continued)

Attribute group	Primary collection mechanism	Number of SELECT statements issued	Tables accessed	Name of stored procedures
MS SQL Availability Replicas Status	SELECT	2	sys.dm_hadr_ availability_replica_ states, sys.availability_groups, sys.dm_hadr_ database_replica_states	None
MS SQL Availability Replicas Status Summary	SELECT	2	sys.dm_hadr_ availability_replica_ states, sys.availability_groups, sys.dm_hadr_database_ replica_states	None
MS SQL Availability Groups Summary	SELECT	3	sys.availability_groups, sys.dm_hadr_ availability_group_ states, sys.dm_hadr_ database_replica_ states, sys.dm_hadr_ availability_replica_ states, sys.availability_replicas	None
MS SQL Availability Replicas Statistics	Perfmon	0	None	None
MS SQL Availability Database Details	SELECT	1	dm_hadr_database_ replica_states, sys.availability_groups	None
MS SQL Availability Database Statistics	Perfmon	0	None	None
MS SQL Availability Database Summary	SELECT	1	dm_hadr_database_ replica_states, sys.availability_groups	None
MS SQL Availability Group Listener Details	SELECT	1	sys.availability_group_ listener_ip_addresses, sys.availability_group_ listenerssavg_list, sys.availability_groups	None
MS SQL Availability Groups Details In Cluster	SELECT	1	sys.availability_groups_ cluster	None
MS SQL Availability Replicas Details In Cluster	SELECT	1	sys.dm_hadr_ availability_replica_ cluster_states, sys.availability _groups_cluster, sys.dm_hadr_ availability_replica_ cluster_nodes,	None
MS SQL Availability Database Details In Cluster	SELECT	1	sys.dm_hadr_database_ replica_cluster_states, sys.availability_ databases_cluster, sys.availability_groups_ cluster	None

Table 2. Key aspects of data gathering for attribute groups (continued)

Attribute group	Primary collection mechanism	Number of SELECT statements issued	Tables accessed	Name of stored procedures
MS SQL Additional Availability Database Details	SELECT	1	sys.dm_hadr_database_replica_states, sys.dm_hadr_availability_replica_states	None
MS SQL Server Properties	SELECT	1	SERVERPROPERTY	None
MS SQL Customized SQL Query Result	SELECT	Depends on SQL Server query issued by user	Depends on SQL Server query issued by user	None

Data collection from operating system or SQL Server files

Microsoft SQL Server agent collects data from operating system or MS SQL Server files for some attribute groups.

Table 3. Data that is accessed by the attribute groups that gather data from operating system or Microsoft SQL Server files

Attribute group	Data accessed
MS SQL Error Event Details	MS SQL server error log file
MS SQL Problem Summary	MS SQL server error log file
MS SQL Problem Detail	MS SQL server error log file
MS SQL Services Detail	Windows Services API
MS SQL Integration Details	Windows Services API
MS SQL Analysis service Data Mining	Windows Services API
MS SQL Analysis Service Memory Statistics	Windows Services API
MS SQL Analysis Service Storage Engine Query	Windows Services API
MS SQL Analysis Service MDX Processing	Windows Services API
MS SQL Analysis Service Rows Processing	Windows Services API
MS SQL Customized SQL Query Details	KoqCusSQL.properties file
MS SQL Audit Details	Windows applications and event logs

Chapter 2. Agent installation and configuration

Agent installation and configuration requires the use of the *IBM Tivoli Monitoring Installation and Setup Guide* and agent-specific installation and configuration information.

To install and configure Microsoft SQL Server agent, use the procedures for installing monitoring agents in the *IBM Tivoli Monitoring Installation and Setup Guide* along with the agent-specific installation and configuration information.

If you are installing silently by using a response file, see "Performing a silent installation of IBM Tivoli Monitoring" in the *IBM Tivoli Monitoring Installation and Setup Guide*.

With the self-describing agent capability, new or updated IBM Tivoli Monitoring agents using IBM Tivoli Monitoring V6.2.3 or later can become operational after installation without having to perform additional product support installation steps. To take advantage of this capability, see "Enabling self-describing agent capability at the hub monitoring server" in the *IBM Tivoli Monitoring Installation and Setup Guide*. Also, see "Self-describing monitoring agents" in the *IBM Tivoli Monitoring Administrator's Guide*.

Requirements

Before installing and configuring the agent, make sure your environment meets the requirements for the IBM Tivoli Composite Application Manager for Microsoft Applications: Microsoft SQL Server Agent.

For the most up-to-date information about system requirements, see the Software product compatibility reports (http://www-969.ibm.com/software/reports/compatibility/clarity/index.html). Search for the ITCAM for Microsoft Applications product.

Installing language packs

You can install language packs for your agent support files on the Tivoli Enterprise Monitoring agent and the Tivoli Enterprise Portal Server. There are separate procedures for installing language packs on Windows and on UNIX or Linux systems.

Before you begin

Ensure that you have installed the product in English before you install a language pack for the agent support files on the Tivoli Enterprise Monitoring Server, and the Tivoli Enterprise Portal Server.

Installing language packs on Windows systems

You can install the language packs on a Windows system.

Procedure

1. Double-click `lpinstaller.bat` in the language pack CD to start the installation program.
2. Select the language of the installer and click **OK**.
3. Click **Next** on the Introduction panel.
4. Click **Add/Update** and click **Next**.
5. Select the folder in which the National Language Support package (`NLSPackage`) files are located.

   Note: Usually the `NLSPackage` files are located in the `nlspackage` folder where the executable installer is located.
6. Select the language support for the agent of your choice and click **Next**.

Note: You can select multiple languages by pressing the Ctrl key.

7. Select the languages that you want to install and click **Next**.
8. Examine the installation summary page and click **Next** to start the installation.
9. Click **Finish** after the installation completes.
10. Restart the Tivoli Enterprise Portal, Tivoli Enterprise Portal Server, and Eclipse Help Server if any of these components are installed.

Installing language packs on UNIX or Linux systems

You can install the language packs on a UNIX or Linux system.

Before you begin

Ensure that you have installed the product in English.

Procedure

1. Run the following command to create a temporary directory on the computer. Ensure that the full path of the directory does not contain any spaces:

   ```
   mkdir dir_name
   ```
2. Mount the language pack CD to the temporary directory that you have created.
3. Run the following command to start the installation program:

   ```
   cd dir_name
   lpinstall.sh ITM Home Directory
   ```

   where *ITM Home Directory* is the location where you have installed IBM Tivoli Monitoring. Usually it is `/opt/IBM/ITM` for AIX® and Linux systems.
4. Select the language of the installer and click **OK**.
5. Click **Next** on the Introduction panel.
6. Click **Add/Update** and click **Next**.
7. Select the folder in which the National Language Support package (`NLSPackage`) files are located.

   Note: Usually, the `NLSPackage` files are located in the `nlspackage` folder where the installer executable is located.
8. Select the language support for the agent of your choice and click **Next**.

   Note: You can select multiple languages by pressing the Ctrl key.
9. Select the languages that you want to install and click **Next**.
10. Examine the installation summary page and click **Next** to start the installation.
11. Click **Finish** after the installation completes.
12. Restart the Tivoli Enterprise Portal, Tivoli Enterprise Portal Server, and Eclipse Help Server if any of these components are installed.

Silent installation of language packs on Windows, UNIX, or Linux systems

You can use the silent-mode installation method to install the language packs. In silent mode, the installation process obtains the installation settings from a predefined response file. It does not prompt you for any information.

Before you begin

First, make sure that you installed the product in the English language.

Procedure

1. Copy and paste the `ITM_Agent_LP_silent.rsp` response file template as shown in “Response file example.”
2. Change the following parameter settings:

   NLS_PACKAGE_FOLDER

   Folder where the National Language Support package (NLSPackage) files are located. Typically, the NLSPackage files are located in the `nlspackage` folder, for example: `NLS_PACKAGE_FOLDER = //tmp//LP//nlspackage`.

   PROD_SELECTION_PKG

   Name of the language pack to install. Several product components can be included in one language package. You might want to install only some of the available components in a language pack.

   BASE_AGENT_FOUND_PKG_LIST

   Agent for which you are installing language support. This value is usually the same as *PROD_SELECTION_PKG*.

   LANG_SELECTION_LIST

   Language you want to install.
3. Enter the command to install the language pack with a response file (silent installation):
   - For Windows systems:

     `lpinstaller.bat -f` *`path_to_response_file`*
   - For UNIX or Linux systems:

     `lpinstaller.sh -c` *`candle_home`* `-f` *`path_to_response_file`*

     where *`candle_home`* is the IBM Tivoli Monitoring base directory.

Response file example

```
#     IBM Tivoli Monitoring Agent Language Pack Silent Installation Operation
#
#This is a sample response file for silent installation mode for the IBM Tivoli
#Monitoring Common Language Pack Installer.
#.
#This file uses the IBM Tivoli Monitoring Common Agent Language Pack with the
#install package as an example.
#Note:
#This response file is for the INSTALLATION of language packs only.
#This file does not support UNINSTALLATION of language packs in silent mode.
#-------------------------------------------------------------------------------
#-------------------------------------------------------------------------------
#To successfully complete a silent installation of the the example of Common Agent
#localization pack, complete the following steps:
#
#1.Copy ITM_Agent_LP_silent.rsp to the directory where lpinstaller.bat or
#lpinstaller.sh is located (IBM Tivoli Monitoring Agent Language Pack build
#location).
#
#2.Modify the response file so that it is customized correctly and completely for
#your site.
#  Complete all of the following steps in the response file.
#
#3.After customizing the response file, invoke the silent installation using the
#following command:
#For Windows:
#    lpinstaller.bat -f <path_to_response_file>
#For UNIX and Linux:
#    lpinstaller.sh -c <candle_home> -f <path_to_response_file>
#Note:<candle_home> is the IBM Tivoli Monitoring base directory.
#--------------------------------------------------------------------------------
```

```
#-------------------------------------------------------------------------------
#Force silent install mode.
#-------------------------------------------------------------------------------
INSTALLER_UI=silent
#--------------------------------------------------------------------------------
#Run add and update actions.
#--------------------------------------------------------------------------------
CHOSEN_INSTALL_SET=ADDUPD_SET
#--------------------------------------------------------------------------------
#NLS Package Folder, where the NLS Packages exist.
#For Windows:
#   Use the backslash-backslash(\\) as a file separator (for example,
#C:\\zosgmv\\LCD7-3583-01\\nlspackage).
#For UNIX and Linux:
#   Use the slash-slash (//) as a file separator (for example,
#//installtivoli//lpsilenttest//nlspackage).
#--------------------------------------------------------------------------------
#NLS_PACKAGE_FOLDER=C:\\zosgmv\\LCD7-3583-01\\nlspackage
NLS_PACKAGE_FOLDER=//tmp//LP//nlspackage
#--------------------------------------------------------------------------------
#List the packages to process; both variables are required.
#Each variable requires that full paths are specified.
#Separate multiple entries with a semicolon (;).
#For Windows:
#           Use the backslash-backslash(\\) as a file separator.
#For Unix and Linux:
#           Use the slash-slash (//) as a file separator.
#--------------------------------------------------------------------------------
#PROD_SELECTION_PKG=C:\\zosgmv\\LCD7-3583-01\\nlspackage\\KIP_NLS.nlspkg
#BASE_AGENT_FOUND_PKG_LIST=C:\\zosgmv\\LCD7-3583-01\\nlspackage\\KIP_NLS.nlspkg
PROD_SELECTION_PKG=//tmp//LP//nlspackage//kex_nls.nlspkg;//tmp//LP//nlspackage//
koq_nls.nlspkg
BASE_AGENT_FOUND_PKG_LIST=//tmp//LP//nlspackage//kex_nls.nlspkg;//
tmp//LP//nlspackage//koq_nls.nlspkg
#-------------------------------------------------------------------------------
#List the languages to process.
#Separate multiple entries with semicolons.
#-------------------------------------------------------------------------------
LANG_SELECTION_LIST=pt_BR;fr;de;it;ja;ko;zh_CN;es;zh_TW
```

Prerequisites checking

The prerequisite checker utility verifies whether all the prerequisites that are required for the agent installation are met. The prerequisite checker creates a log file that contains a report of all the prerequisites checks when the prerequisite checker was run.

For the Microsoft SQL Server agent, the prerequisite checker verifies the following requirements:

- Memory
- Disk
- Operating systems
- SQL Server versions

Additionally, the prerequisite checker verifies whether the user, who installs the agent, is a member of the Administrators group.

For detailed information about installation prerequisites, see the Software product compatibility reports (http://www-969.ibm.com/software/reports/compatibility/clarity/index.html).

You can run the prerequisite checker in stand-alone mode or remotely. For more information about the prerequisite checker, see "Prerequisite Checking for IBM Tivoli Monitoring Agents" in the *IBM Tivoli Monitoring Installation and Setup Guide*.

Running as a non-administrator user

You can run the monitoring agent for Microsoft SQL Server as a non-administrator user.

About this task

The Microsoft SQL Server agent can be run as a non-administrator user. This process supports Domain Users only.

Procedure

1. Start Windows application Active Directory Users and Computers and create a domain user.
   - Make sure that the new user is a member of the *Domain Users* group.
   - Make sure that the SQL Server that you monitor is a member of *Domain Computers*.
2. Add the newly created domain user in the *SQL Server Login* user group. The domain user must have sysadmin SQL Server role permissions on the SQL Server.
3. Log on to the SQL Server with a domain administrator account.
4. Use **File Systems**, to give the Modify permission to every drive that the Microsoft SQL Server agent accesses and propagate permissions to all subdirectories. Complete the following steps to propagate permissions:
   a. Go to **My Computer**.
   b. Right-click the *drive*.
   c. Click the **Security** tab.
   d. Add the newly created user.
   e. Give modify permissions to the newly created user.
   f. Click **OK**. This procedure takes a few minutes to apply permissions throughout the file system.
5. By using the Windows Registry, grant read access to `HKEY_LOCAL_MACHINE`, and propagate settings. Complete the following steps to propagate settings:
   a. Right-click the `HKEY_LOCAL_MACHINE` directory and select **Permissions**.
   b. Add the newly created user.
   c. Select the newly created user.
   d. Select the **Allow Read** check box.
   e. Click **OK**. This procedure takes a few minutes to propagate through the entire `HKEY_LOCAL_MACHINE` tree.
6. By using the Windows Registry, grant the agent-specific registry permissions according to the following list:
   - If you installed a 32-bit agent on a 32-bit operating system, grant full access to the `HKEY_LOCAL_MACHINE\SOFTWARE\Candle` directory, and then propagate the settings.
   - If you installed a 32-bit agent on a 64-bit operating system, grant full access to the `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Candle` directory, and then propagate the settings.
   - If you installed a 64-bit agent on a 64-bit operating system, grant full access to the `HKEY_LOCAL_MACHINE\SOFTWARE\Candle` directory, and then propagate the settings.

   Complete the following steps to propagate settings:

   a. Right-click the directory for which you have granted full access and select **Permissions**.
   b. Add the newly created user.
   c. Select the newly created user.
   d. Select the **Allow Full Control** check box.
   e. Click **OK**. This procedure takes a few minutes to propagate through the entire `HKEY_LOCAL_MACHINE\SOFTWARE\Candle` tree.

7. Add a new Domain User to the *Performance Monitor Users* group.
8. Verify that Domain Users are members of the *Users* group.
9. Grant the following permissions to the Windows directory while running as a non-administrator user:
   - If a 32-bit agent is installed on a 32-bit operating system, grant read and write access to the `OS_installation_drive:\Windows\system32` directory
   - If a 32-bit agent is installed on a 64-bit operating system, grant read and write access to the `OS_installation_drive:\Windows\SysWOW64` directory
   - If a 64-bit agent is installed on a 32-bit or a 64-bit operating system, grant read and write access to the `OS_installation_drive:\Windows\system32` directory

   Note: You do not need to grant the permissions to the Windows directory if you are using Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012.
10. Grant the following permissions to the SQL Server data file and log file:
    - The default data file path is *`SQLServer_root_dir`*`\DATA`, where *SQLServer_root_dir* is the root directory of the SQL Server instance. For example, if the root directory of the SQL Server instance is `C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL`, the data file path is `C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\DATA`.
    - The default log file path is *`SQLServer_root_dir`*`\LOG`, where *SQLServer_root_dir* is the root directory of the SQL Server instance. For example, if the root directory of the SQL Server instance is `C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL`, the log file path is `C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\LOG`.
11. Grant full permissions to the `Candle_Home` directory. The default path is `C:\IBM\ITM`.
12. Restart the SQL Server to ensure that local security permissions are applied.
13. Change the logon settings for the SQL Server agent services to the non-administrator user by completing the following steps:
    a. Click **Start > Administrative Tools > Services**.
    b. Right-click the **Monitoring Agent For SQL Server *instance_name***, and click **Properties**. The SQL Service Properties window opens.
    c. Click the **Log On** tab.
    d. Click **This account** and type the user name.
    e. In the **Password** and **Confirm Password** fields, enter the password, and click **OK**.
    f. Repeat steps b - e for the **Monitoring Agent For SQL Server Collector *instance_name***, where *instance_name* is the Microsoft SQL Server instance name.

Agent-specific installation and configuration

You can install and configure the Microsoft SQL Server agent locally or remotely by using a GUI or command line.

When you install the agent support files on the Tivoli Enterprise Portal Server, the composite workspaces are also installed.

Never attempt to start the monitoring agent until you have completed the configuration steps appropriate to the installation of the monitoring agent.

Permissions required for configuring the monitoring agent

The user ID on the SQL Server must be granted specific permissions for configuring the monitoring agent.

If the Microsoft SQL Server agent is configured for Windows authentication, the agent uses the run-as user ID to access the Microsoft SQL Server. Otherwise, the agent uses the configured SQL Server user ID. Further reference in this section to Microsoft SQL Server user ID or SQL Server ID refers to either the run-as user ID or the Microsoft SQL Server user ID.

The Microsoft SQL Server user ID must have access to the Microsoft SQL Server agent. The procedure described in this section includes creating a Microsoft SQL Server user ID and granting permission to the new user ID, which is the minimum authority required.

The SQL Server ID used to configure this monitoring agent must have the following SQL Server authorities:

- Required authorization roles
  - Database roles: Public access is required for each database that is being monitored.
  - Server roles: No special Server Role is required.
- Optional authorization roles
  - Each Take Action command has a separate set of authorization roles that are required for the SQL Server credentials to pass to the Take Action command. For more information about these authorization roles, see the "Take Action commands reference" topic in the *Microsoft SQL Server agent: Reference*.

The monitored attributes in Table 4 on page 20 require additional authorization for the SQL Server ID used to configure the Microsoft SQL Server agent. If you want to monitor any of these attributes, configure the SQL Server ID used by the monitoring agent with the corresponding authorization.

Table 4. Additional authorization

Attribute group	Navigation item, workspace, view	Situation	Authorization
MS SQL Job Detail	Navigator item: Jobs	None predefined	**Server Role: Server Administrators**This authorization enables data collection for all SQL Server jobs. —OR— **Database Role (msdb database): public**An SQL Server Agent proxy account must be defined. The agent can collect job data only on the jobs that this SQL Server ID owns. —OR— **Database Role (msdb database): SQLAgentUserRole (SQL Server 2005 only)**The agent can collect job data only on the jobs that this SQL Server ID owns. —OR— **Database Role (msdb database): SQLAgentReaderRole (SQL Server 2005 only)**The agent can collect job data on all SQL Server jobs. —OR— **Database Role (msdb database): SQLAgentOperatorRole (SQL Server 2005 and 2008)**The agent can collect job data on all SQL Server jobs.
	Workspace: Job Detail		
	View: Job Status, Job Detail		
MS SQL Job Summary	Navigator item: Jobs		
	Workspace: Job Summary		
	View: Running Job Current Interval, Job Summary		
MS SQL Lock Detail MS SQL Lock Summary	Navigator item: Server Locking	None predefined	VIEW SERVER STATE permission is required on the SQL Server
	Workspace: Server Locking, Lock Detail		
	View: Log Detail, Lock Detail		

Table 4. Additional authorization (continued)

Attribute group	Navigation item, workspace, view	Situation	Authorization
MS SQL Server Detail	Navigator item: Servers	• MS_SQL_Proc_Buffs_Active_Warn • MS_SQL_Proc_Buffs_Active_Crit • MS_SQL_Proc_Buffs_Used_Warn • MS_SQL_Proc_Buffs_Used_Crit • MS_SQL_Proc_Cache_Active_Warn • MS_SQL_Proc_Cache_Active_Crit • MS_SQL_Proc_Cache_Used_Warn • MS_SQL_Proc_Cache_Used_Crit	**Database Role: db_owner (on default database associated with the SQL Server ID)** —OR— **Server Role: System Administrator**
	Workspace: Server Detail		
	View: Server Detail-Status, Server Detail-Statistics		
MS SQL Server Summary	Navigator item: Servers	None predefined	VIEW SERVER STATE permission is required on the SQL Server
	Workspace: Server Summary		
	View: CPU Utilization, Server Summary		
MS SQL Statistics Summary	Navigator item: Servers	• MS_SQL_Client_Cnt_Pct_Used_Crit • MS_SQL_Client_Cnt_Pct_Used_Warn • MS_SQL_IOERR_Startup_Warning • MS_SQL_IOError_Curintvl_Warning • MS_SQL_LogonPct_Crit • MS_SQL_LogonPct_Warning • MS_SQL_Network_Read_Rate_Crit • MS_SQL_Network_Read_Rate_Warn • MS_SQL_Network_Write_Rate_Crit • MS_SQL_Network_Write_Rate_Warn • MS_SQL_Pct_IO_Warning	VIEW SERVER STATE permission is required on the SQL Server
	Workspace: Server Statistics		
	View: Statistics Summary Chart view, IO Summary, Statistic Summary table view		

Table 4. Additional authorization (continued)

Attribute group	Navigation item, workspace, view	Situation	Authorization
MS SQL Table Detail	Navigator item: Databases Workspace: Table Detail View: Table Detail, Fragmentation (greater than 30%) by Index Name	• MS_SQL_Fragmentation_Warn • MS_SQL_Fragmentation_Crit • MS_SQL_Opt_Stats_Age_Warn • MS_SQL_Opt_Stats_Age_Crit	**Database Role: db_owner (for each database)** —OR— **Server Role: System Administrator** —OR— VIEW DATABASE STATE and VIEW SERVER STATE permissions are required on the SQL Server
MS SQL Audit Details	Navigator item: Servers Workspace: Audit Details View: Audit Details	None predefined.	**Server Role: Server Administrator** —OR— CONTROL SERVER permission is required on the SQL Server
MS SQL Database Detail	Navigator item: Databases Workspace: Databases Information View Database Detail	None predefined	**Server Role: Server Administrator** —OR— CONTROL SERVER permission is required on the SQL Server
MS SQL Filegroup Detail	Navigator item: Databases Workspace: Filegroup Detail View: Filegroup Detail	None predefined	**Server Role: Server Administrator** —OR— CONTROL SERVER permission is required on the SQL Server
MS SQL Device Detail	Navigator item: Databases Workspace: Device Detail View: Device Detail	None predefined	**Server Role: Server Administrator** —OR— CONTROL SERVER permission is required on the SQL Server

Granting permissions for SQL Server

Before installing the Microsoft SQL Server agent, you must grant permissions to the Microsoft SQL Server. The process of granting permissions is the same for SQL Server 2005, or later.

Before you begin

Ensure that you have the Database administrator authorization role to grant permissions. If you are configuring the Microsoft SQL Server agent locally, complete this procedure between the steps for installing and configuring the monitoring agent. If you are configuring the Microsoft SQL Server agent remotely, complete this procedure after installing and configuring the monitoring agent.

Perform the appropriate installation procedures in the *IBM Tivoli Monitoring Installation and Setup Guide*.

Procedure

1. Click **Start > Programs > Microsoft SQL Server 2005 > SQL Server Management Studio**.
2. In the **Tree** tab, select **Logins** in the `Security` folder (**Console Root > WindowsName > Security > Logins**).
3. Right-click **Logins** and select **New Login**.
4. Select the **General Folder** option and type the SQL Server user ID if you are using SQL Server authentication. Type the Windows user ID if you are using Windows authentication.
5. Select one of the following authentication methods:
   - SQL Server
   - Windows
6. Type a password in the **Password** field if you are using the SQL Server authentication method.
7. Select the **User Mapping** option.
8. In the **Specify which databases can be accessed by this login** area, select the check box for each database that you currently have to give permission to each selected database.
9. Select the **Server Roles** option.
10. If you are running a Take Action command, you might need additional authority. See the Take Action command descriptions in Predefined Take Action commands.
11. Click **OK** to display the Confirm Password window.
12. Retype the password that you typed for the user ID.
13. Click **OK** to display the new user ID in the **Logins** list.

What to do next

Configure the Microsoft SQL Server agent to start the monitoring agent and begin monitoring your Microsoft SQL Server application.

Local configuration

You can install and configure the agent locally.

For more information, see the "Installing monitoring agents" topic in the *IBM Tivoli Monitoring Installation and Setup Guide*. Also, use the agent-specific configuration information in this section and in Table 5 on page 25 for the Manage Tivoli Enterprise Monitoring Services window.

Configuration parameters in the Database Server Properties window

Use the Configure Database Agents window to select the database that you want to monitor.

1. Click **Start** > **All Programs** > **IBM Tivoli Monitoring**.
2. Click **Manage Tivoli Enterprise Monitoring Services**. The Manage Tivoli Enterprise Monitoring Services window opens.
3. In the Manage Tivoli Enterprise Monitoring Services window, right-click **Monitoring Agent for Microsoft SQL Server** (for **Template**).
4. Click **Configure Using Defaults**. The Configure Database Agents window opens.
5. In the Configure Database Agents window, from the **Database Servers Available** list, select the server that you want to monitor, and move the server to the **Server to Monitor** list. The Database Server Properties window opens.

The following fields are populated in the Database Server Properties window:

- Server Name
- Database Version
- Home Directory
- Error Log File

The following fields in the Database Server Properties window are optional:

- Windows Authentication
- Support Long Lived Database Connections
- Extended Parms
- Monitor all Databases
- Include
- Day(s) Frequency
- Weekly Frequency
- Monthly Frequency
- Collection Start Time
- Table Detail Continuous Collection

6. If you do not select the **Windows Authentication** field, enter your user ID and password in the **Login** and **Password** fields by using only ASCII characters.
7. In the **Extended Parms** field, type one of the following options to disable the data collection, and then click **OK**.
   - `koqtbld` for the Table Detail attribute group.
   - `koqdbd` for the Database Detail attribute group.
   - `koqtbld,koqdbd` for disabling the Table Detail and Database Detail attribute groups.
8. If you clear the **Monitor All Databases** check box, specify the list of databases for which you want to enable or disable monitoring, in the text field that is present in the **Databases** group area.

   Remember: If you select the **Monitor All Databases** check box and also specify the databases that you want to monitor in the text field that is present in the **Databases** group area, then priority is given to the value of the **Monitor All Databases** check box. The list of the databases that you specify in the text field is ignored.
9. Specify a daily, weekly, or monthly frequency for the collection of the MS SQL Table Detail attribute group.
10. Select the **Table Detail Continuous Collection** check box to enable continuous collection of the MS SQL Table Detail attribute group. If you select the **Table Detail Continuous Collection** check box, enter a value in the **Interval Between Two Continuous Collection (in minutes)** field.
11. In the Configure Database Agents window, click **OK**, and then start the agent.

Configuration settings

You can specify the SQL Server agent configuration settings in the interfaces that include the Manage Tivoli Enterprise Services window, the Tivoli Enterprise Portal, and the tacmd command line.

Table 5 on page 25 contains a list of the configuration settings for each of the interfaces where you can specify these settings and a description of each setting.

Table 5. Names and descriptions of configuration settings for each interface

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**
Server Name	Database Server Instance Name[1]	`INSTANCE=`*`InstanceName`*	Name of the SQL Server instance that is to be monitored. Use MSSQLSERVER as the instance name for the default instance. The name must be short enough to fit within the total managed system name, which must be between 2 and 32 characters in length.	If the SQL Server instance being monitored is the default SQL Server instance, enter `MSSQLSERVER` in this field. If the SQL Server instance being monitored is a named instance with the instance name of `mysqlserver` and the host name is `popcorn`, enter `mysqlserver` in this field.
Login	Database Server User Id[1]	`DBSETTINGS.db_login =`*`UserId`*	SQL Server user ID to be used to connect to the SQL Server. See "Permissions required for configuring the monitoring agent" on page 18 for more information. Login is required only when **`Windows Authentication`** parameter is set to `False`. Use only ASCII characters.	
Password	Password[1]	`DBSETTINGS.db _password=`*`Password`*	Password for the SQL Server user ID. Password is required only when **`Windows Authentication`** parameter is set to `False`. Use only ASCII characters. **Important:** A password cannot be blank. If you specify a login name, you must enter a password.	

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**
Database Version	The version of the SQL Server instance.	This parameter need not be specified in the **tacmd** command.	The database versions for the SQL Server.	The database versions for the SQL Server instance are as follows: • Microsoft SQL Server 2016 - 13.0.1601.5 • Microsoft SQL Server 2014 - 12.0.2000.8 • Microsoft SQL Server 2012 - 11.0.2100.60 • Microsoft SQL Server 2008 R2 - 10.50.1600.1 • Microsoft SQL Server 2008 - 10.0.1600.22
Home Directory	Database Server Home Directory Path[1]	`DBSETTINGS.db_home=`*`HomeDirPath`*	Install directory Server instance monitored.	The default home directory path for the default SQL Server 2005 instance is `C:\Program Files\Microsoft SQL Server\MSSQL`. A named SQL Server 2005 instance has a default home directory path in the format `C:\Program Files\Microsoft SQL Server\MSSQL$`*`instance_name`*, where *instance_name* is the SQL Server instance name.
Error Log File	Database Server Error Log File[1]	`DBSETTINGS.db_errorlog=`*`ErrorlogPath`*	Fully qualified location and name of the SQL Server Error Log	The default error log path for the default SQL Server 2005 instance is `C:\Program Files\Microsoft SQL Server\MSSQL\LOG\ERRORLOG`. A named SQL Server 2005 instance has a default error log path in the format `C:\Program Files\Microsoft SQL Server\MSSQL$`*`instance_name`*`\LOG\ERRORLOG`, where *instance_name* is the SQL Server instance name.

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**

Windows Authentication	Windows Authentication	`DBSETTINGS.db _winauth= 1/0`	Enables or disables Windows Authentication
			If the **Windows Authentication** check box is selected, windows credentials will be used for authentication.
			Remember: If you do not select the **Windows Authentication** check box, you must specify values for the **Login** and **Password** parameters. If you do not specify these parameters and click **OK** in the Database Server Properties window, an error message is displayed in a pop-up window and the agent configuration does not finish.
				`DBSETTINGS.db_winauth=1`
				—OR—
				`DBSETTINGS.db_winauth=0`

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**

Support Long Lived Database Connections	Support Long Lived Database Connections	`DBSETTINGS.db _lldbconn= 1/0`	Enables or disables long lived database connections The following Take Action commands do not use Long Lived Database Connections:
			• Database Check-up and Tune-up
			• Dump Database
			• Dump Transaction Log
			• Rebuild Fragmented Indexes
			• Refresh Query Optimizer Statistics
			• Update Space Usage Information
			The following attribute groups do not use Long Lived Database Connections:
			• MS SQL Text
			• MS SQL Filegroup Detail
			• MS SQL Server Summary
				`DBSETTINGS.db_lldbconn=1`
				—OR—
				`DBSETTINGS.db_lldbconn=0`

Extended Parms	Extended Parms	`DBSETTINGS.db _extparms="KOQTBLD"`	Disables capture of Table Detail attributes.	To disable the data collection for the Table Details attribute group, enter `koqtbld` in the **Extended Parms** field. **Remember:** To disable data collection for the Table Detail and Database Detail attribute groups, type `koqtbld,koqdbd` in the **Extended Parms** field.
		`DBSETTINGS.db _extparms="KOQDBD"`	Disables capture of Database Detail attributes.	To disable the data collection for the Table Details attribute group, enter `koqtbld,koqdbd` in the **Extended Parms** field. **Remember:** To disable data collection for the Table Detail and Database Detail attribute groups, type `koqtbld,koqdbd` in the **Extended Parms** field.

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**
Database	Include or Exclude Database name(s)	`DBSETTINGS.db_include _state=`*x*	To select the databases for monitoring, specify a value for this parameter. • To monitor particular databases, select **Include** from the list, and then specify the names of the databases in the text field next to the list. • To exclude particular databases from being monitored, select **Exclude** from the list, and then specify the names of the databases in the text field next to the list. **Note:** To enable or disable the monitoring of particular databases, clear the **Monitor All Databases** check box.	While configuring the agent remotely, use one of the following values in the **`DBSETTINGS.db_include _state=`***x* command: • If the value of *x* is 0, the database names that are specified in the text field are monitored. • If the value of *x* is 1, the database names that are specified in the text field are excluded from being monitored.

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**
Database (Continued)	Monitor all databases	`DBSETTINGS.db_monitorall_state=`*x*	To monitor all the databases of an SQL server instance, ensure that the **Monitor All Databases** check box in the **Databases** group area is selected. If you clear the **Monitor All Databases** check box, then you must specify the list of databases for which you want to enable or disable monitoring, in the text field under the **Databases** group area. If you do not specify the databases to be monitored, an error message is displayed in a pop-up window when you click **OK**, and the agent configuration fails.	In the **`DBSETTINGS.db_monitorall_state=`***x* command, specify one of these values when you configure the agent: • The value 0 for *x* indicates that the database names in the text field are monitored. • The value 1 for *x* indicates that all the databases are monitored.
Database (Continued)			**Remember:** If you select the **Monitor All Databases** check box and also specify the databases to monitor in the text field under the **Databases** group area, then priority is given to the value of the **Monitor All Databases** check box. The list of databases that you specify in the text field is ignored.	

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**

Database
(Continued)
	Database list to include or exclude	`DBSETTINGS.db_list`	Use the text field to filter databases that you want to monitor.
			To specify database filter, you must first select a separator. A separator is a character that distinguishes a database name or database expression from the other database name or database expression.
			While selecting a separator, ensure that database names and database expression do not contain the character that you choose as a separator. You must not use the wildcard characters that are typically used in the T-SQL query (for example, %, _, [ ], ^, -) if they are used in the database names or database expression.
				Examples of filters:
				Case 1: `% usage`
				Example:
				`@@%m%`
				Output: All the databases that have the character m in their names are filtered.
				Case 2: `_ usage`
				Example:
				`@@____`
				Output: All the databases that are of length four characters are filtered.
				Case 3: `[] usage`
				Example:
				`@@[m]___`
				Output: All the databases of length four characters and whose names start with the character m are filtered.
				Case 4: `[^] usage`
				Example:
				`@@[^m]%`
				Output: All the databases (of any length) except those whose names start with the character *m* are filtered.

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**

Database
(Continued)
			When specifying database filter:
			• Database names must start with a separator.
			• Database expression must start with two separators.
			Note: Database expression is a valid expression that can be used in the `LIKE` part of the `T-SQL` query. However, you cannot use the `T-SQL ESCAPE` clause while specifying the database expression.The following workspaces are affected by database filter:
			• Database Information
			• Enterprise Database Summary
			• Device Detail
			• Table Detail
			• Filegroup Detail
				Case 5: Wrong input
				Example:
				`@%m%`
				Output: None of the databases are filtered.
				Case 6: Default
				Example: Field left blank (No query is typed)
				Output: All the databases are filtered.
				Case 7: Mixed patterns
				Example:
				`@@[m-t]_d%`
				Output: All the databases (of any length) whose names start with the characters m, n, o, p, q, r, s, t, followed by any character, with the character d in the third place are filtered.

Day(s) Frequency	Table Detail Collection Day(s) Frequency[3]	`DBSETTINGS.db_tbld_daily`	Use this feature to define the frequency of collecting data of Table Detail attributes. The values can be from zero to 31.	**`DBSETTINGS.db_tbld_daily=1`**
Weekly Frequency	Table Detail Collection Weekly Frequency[3]	`DBSETTINGS.db_tbld_weekly`	Use this feature to specify a particular day for collecting data for Table Detail attributes. The values can be from zero to seven.	**`DBSETTINGS.db_tbld_weekly=1`**

Table 5. Names and descriptions of configuration settings for each interface (continued)

Interfaces where configuration settings are specified				
Manage Tivoli Enterprise Monitoring Services window	**Tivoli Enterprise Portal**	**tacmd command line**	**Description**	**Examples**
Monthly Frequency	Table Detail Collection Monthly Frequency[3]	`DBSETTINGS.db_tbld _monthly`	Use this feature to define the data collection of Table Detail attributes on a particular day of the month. The possible values are 1, 2, 3, and so on.	**`DBSETTINGS.db_tbld_monthly=5`**
Collection Start Time	Collection Start Time Hour	`DBSETTINGS.db_tbld _coll_stime_hour`	The possible values are zero to 23. The default value is zero.	**`DBSETTINGS.db _tbld_coll_stime_hour=1`**
	Collection Start Time Minute	`DBSETTINGS.db_tbld _coll_stime_minute`	The possible values are from zero to 59. The default value is zero.	**`DBSETTINGS.db _tbld_coll_stime_minute=59`**
Table Detail Continuous Collection	Table Detail Continuous Collection	`DBSETTINGS.db_tbldconcoll`	Use this feature for the continuous background collection of Table Detail data. The **Table Detail Continuous Collection**check box is selected by default.	**`DBSETTINGS.db_tbldconcoll=1`**
Interval Between Two Continuous Collection (in min.)	Interval Between Two Continuous Collection (in min.)	`DBSETTINGS.db_tbldcoll _time`	Specify the time for the interval (in minutes) between two collections. For data collection methods, see "Data collection for the Table Details attribute group"	**`DBSETTINGS.db_tbldcoll _time=10`**

- [1] **Database Server Properties** tab
- [2] **Agent** tab **Run as** area

Data collection for the Table Details attribute group

There are three methods to configure the agent to collect data for the Table Details attribute group:

Continuous collection

Choose this method of data collection if the SQL Server contains many tables, few large tables, or tables that are frequently updated. By using the continuous collection method, you can configure the agent to continuously collect data in the background. To enable continuous collection, select the **Table Details Continuous Collection** check box in the configuration window. Specify the time interval between two collections in the **Interval Between Two Continuous Collection (in min.)** field. The default and minimum value of this interval is 3 minutes.

Important: If you select the **Table Details Continuous Collection** check box, you must specify a value in the **Interval Between Two Continuous Collection (in min.)** field.

Scheduled collection

Choose this method of data collection if there are no frequent updates in SQL Server tables. By using the scheduled collection method, you can configure the agent to collect data at the time interval for which data collection occurs frequently. For example, if you specify all three Table Details Collection configuration settings (Day, Weekly, and Monthly), the agent starts the data collection according to the following conditions:

- If Day(s) Frequency ≤ 7, the Day(s) Frequency settings are selected, and the Weekly and Monthly frequency settings are ignored.
- If Day(s) Frequency > 7, the Weekly Frequency settings are selected, and the Day(s) and Monthly frequency settings are ignored.

To enable scheduled collection, clear the **Table Details Continuous Collection** check box and specify values for the **`Collection Start Time`**, **`Day(s) Frequency`**, **`Weekly Frequency`**, and **`Monthly Frequency`** parameters in the configuration window. The minimum time interval between two collections is 1 day.

Demand based collection

Choose this method of data collection if there are few small tables on the SQL Server. The agent collects data when the agent receives a data collection request. If the SQL Server contains many tables or few large tables, then the data collection takes time. If the data collection does not complete in the specified time frame, then no data is sent to the Tivoli Enterprise Monitoring Server. To enable demand based data collection, clear the **Table Details Continuous Collection** check box, and do not specify any value for the **`Collection Start Time`**, **`Day(s) Frequency`**, **`Weekly Frequency`**, and **`Monthly Frequency`** parameters in the configuration window.

Remember: If you select the **Table Detail Continues Collection** check box, priority is given to continuous data collection method and agent collects the data at the interval that is mentioned in the **Interval Between Two Continuous Collection (in min.)** field. The values that you have specified the daily, weekly, or monthly frequencies are ignored.

Configuring local environment variables

You can configure local environment variables to change the behavior of the Microsoft SQL Server agent.

Procedure

1. In the Manage Tivoli Enterprise Monitoring Services window, from the **Actions** menu, click **Advanced > Edit Variables**.
2. In the Tivoli Enterprise Monitoring Server: Override Local Variable Settings window, click **Add**.
3. In the Add Environment Setting Override window, enter the variable and the corresponding value.

   Note: See "Local environment variables" to view the list of environment variables that you can configure.

Local environment variables

You can change the behavior of the Microsoft SQL Server agent by configuring local environment variables.

Variables for checking SQL Server service availability

To check the availability of the SQL Server service, add the following environment variables:

- **COLL_MSSQL_RETRY_INTERVAL**: This variable provides the retry interval (in minutes) to check the SQL Server service status. If the value is less than or equal to zero, then the variable takes the default value of 1 minute.
- **COLL_MSSQL_RETRY_CNT**: This variable provides the number of retries that the SQL Server agent makes to check whether SQL Server service is started or not. If the SQL Server service is not started after the number of retries that are specified in this variable, then the collector stops working. If the value of the variable is less than or equal to zero, then the variable takes the default value of `3`.

Variables for monitoring the SQL Server error log file

To monitor the MS SQL Error Event Details attribute group, add the following environment variables:

- **COLL_ERRORLOG_STARTUP_MAX_TIME**: This variable provides the time interval (T) for error collection before agent startup. The default value is `0` minutes. This variable can take the following values:

  T = 0 The agent starts monitoring the error log file from the time the agent starts or is restarted. The agent does not read the errors that were logged in the error log file before the agent was started.

  T = 1 The agent monitors the error log file according to the following values that are set for the **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** variable, which is represented by R.
  - If R < 0, the agent starts monitoring the error log file from the time the agent starts or is restarted.
  - If R = 1, the agent monitors all the errors that are logged in the error log file.
  - If R > 1 and the agent is installed for the first time, the agent monitors the error log file until R errors are monitored. If R > 1 and the agent is restarted, the agent monitors all the previously missed R errors.

  T > 1 The agent monitors all previous errors that were logged up to T minutes from the time the agent starts or is restarted. The agent monitoring also depends on the following values that you set for the **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW** variable:
  - If R ≤ 0, the agent starts monitoring the error log file from the time the agent starts or is restarted.
  - If R = 1, the agent monitors the error log file for all the errors that are logged up to T minutes.
  - If R > 1, the agent monitors not more than R errors that are logged in last T minutes.
- **COLL_ERRORLOG_STARTUP_MAX_EVENT_ROW**: This variable provides the maximum number of errors that must be processed at agent startup. The default value is `0`. This variable can take the following values:

  R = 0 The agent starts monitoring the error log file from the time the agent starts or restarts. The agent does not read errors that were created in the error log file before the agent was started.

  R = 1 The agent monitors the errors that were logged in the last T minutes from the time the agent starts or restarts.

  R > 1 The agent monitors R errors logged in the last T minutes.
- **COLL_ERRORLOG_MAX_EVENT_ROW**: This variable provides the number of error rows to be displayed on the Tivoli Enterprise Portal. The default value is `50`. This variable can take the following values:

  X = 0 The agent does not display error logs on the Tivoli Enterprise Portal.

  X > 0 The agent displays X error rows on the Tivoli Enterprise Portal.

Variable for query timeout interval

To set the query timeout interval for the SQL Server agent, use the **QUERY_TIMEOUT** variable. This environment variable defines the maximum amount of time (in seconds) that the SQL Server agent waits to receive a response for a query that is sent to the SQL Server. The value for this variable must be less than 45 seconds. However, if you set the value for this variable as 0 seconds, the SQL Server agent waits indefinitely to receive a response from the SQL Server. If the SQL Server agent accesses many locked databases, you must assign a value to this variable in the range of 10 - 20 seconds. If the query is not processed within the set timeout interval, the SQL Server agent skips the timed out query and moves to the next query in the queue. The agent does not display data on the Tivoli Enterprise Portal for the query that has timed out.

Variable for viewing information about enabled jobs

To view the information about enabled jobs in the MS SQL Job Detail attribute group, use the **COLL_JOB_DISABLED** environment variable. If you set the value of this variable as 1, the Microsoft SQL Server agent does not display information about disabled jobs. If you do not specify this variable, you can view information about enabled and disabled jobs.

Variable for limiting rows in the MS SQL Filegroup Detail attribute group

To limit the number of rows the that the collector service fetches for the MS SQL Filegroup Detail attribute group, use the **COLL_KOQFGRPD_MAX_ROW** environment variable. This environment variable defines the maximum number of rows that the collector service fetches for the Filegroup Detail attribute group. If you do not specify a value for this variable, the collector service fetches 10,000 rows for the Filegroup Detail attribute group. Use this environment variable to modify the default limit of maximum rows in the koqcoll.ctl file. Complete the following steps to modify the default limit:

1. Specify the maximum number of rows for KOQFGRPD in the koqcoll.ctl file.
2. Add the **COLL_KOQFGRPD_MAX_ROW** environment variable, and ensure that the value of this variable is the same as the value that you have specified in the koqcoll.ctl file.

If the value in the koqcoll.ctl file is less than the value that is specified in the **COLL_KOQFGRPD_MAX_ROW** environment variable, the value in the koqcoll.ctl file is treated as the value for the maximum number of rows.

If the value in the koqcoll.ctl file is greater than the value that is specified in the **COLL_KOQFGRPD_MAX_ROW** environment variable, the value in the **COLL_KOQFGRPD_MAX_ROW** environment variable is treated as the value for the maximum number of rows.

Variables for enhancing the collection for the MS SQL Filegroup Detail attribute group

Use the **COLL_DBD_FRENAME_RETRY_CNT** variable to specify the number of attempts that can be made to move the %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_TEMP file to the %COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__FGRP_PREV file.

If you do not specify a value for this variable, the Microsoft SQL Server agent makes 3 attempts to move the file.

Variable for limiting rows in the MS SQL Device Detail attribute group

To limit the number of rows that the collector service fetches for the MS SQL Device Detail attribute group, use the **COLL_KOQDEVD_MAX_ROW** environment variable. This environment variable defines the maximum number of rows that the collector service fetches for the Device Detail attribute group. If you do not specify a value for this variable, the collector service fetches 10,000 rows for the Device Detail attribute group. Use this environment variable to modify the default limit of maximum rows in the koqcoll.ctl file. Complete the following steps to modify the default limit:

1. Specify the maximum number of rows for KOQDEVD in the koqcoll.ctl file.
2. Add the **COLL_KOQDEVD_MAX_ROW** environment variable, and ensure that the value of this variable is the same as the value that you have specified in the koqcoll.ctl file.

If the value in the `koqcoll.ctl` file is less than the value that is specified in the **`COLL_KOQDEVD_MAX_ROW`** environment variable, the value in the `koqcoll.ctl` file is treated as the value for the maximum number of rows.

If the value in the `koqcoll.ctl` file is greater than the value that is specified in the **`COLL_KOQDEVD_MAX_ROW`** environment variable, the value in the **`COLL_KOQDEVD_MAX_ROW`** environment variable is treated as the value for the maximum number of rows.

Variables for enhancing the collection for the MS SQL Device Detail attribute group

To enhance the MS SQL Device Detail attribute group collection, add the following environment variables:

- **`COLL_KOQDEVD_INTERVAL`**: This environment variable enables you to specify a time interval (in minutes) between two consecutive collections of the MS SQL Device Detail attribute group.

  Note: By default, the data collection for the Device Detail attribute group is demand based. Use the **`COLL_KOQDEVD_INTERVAL`** variable to start a thread based collection for the Device Detail attribute group and to set the time interval between two threaded collections.
- **`COLL_DBD_FRENAME_RETRY_CNT`**: Use this environment variable to specify the number of attempts that can be made to move the `%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_TEMP` file to the `%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DEVD_PREV` file.

If you do not specify a value for this variable, the Microsoft SQL Server agent makes 1 attempt to move the file.

Variables for enhancing the collection for the MS SQL Database Detail attribute group

To enhance the MS SQL Database Detail attribute group collection, use the following environment variables:

- **`COLL_KOQDBD_INTERVAL`**: Use this environment variable to specify a time interval (in minutes) between two consecutive collections of the MS SQL Database Detail attribute group. If you do not specify a value for this variable, the Microsoft SQL Server agent sets the time interval for collecting the threaded collections of the database details to 3 minutes.
- **`COLL_DBD_FRENAME_RETRY_CNT`**: Use this environment variable to specify the number of attempts that can be made to move the `%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_TEMP` file to the `%COLL_HOME%_tmp_%COLL_VERSION%_%COLL_SERVERID%_%COLL_SERVERID%__DBD_PREV` file.

If you do not specify a value for this variable, the Microsoft SQL Server agent makes 1 attempt to move the file.

Variables for enhancing the collection for the MS SQL Audit Details attribute group

To enhance the MS SQL Audit Details attribute group collection, use the following environment variables:

- **`COLL_AUDIT_TYPE`**: Use this variable to enable or disable the monitoring of specific logs. The default value of the variable is [AL][FL][SL]. By default, the agent monitors all three types of logs that include the application logs, audit files, and the security logs. The value of the variable includes two character code for each log type:
  - [AL] for application logs
  - [FL] for audit files
  - [SL] for security logs

You can change the value of the variable to disable the monitoring of specific log type. For example, if you specify the value of the variable as [AL][SL] the audit files are not monitored. If no value is specified for the variable, audit details not monitored.

- **COLL_AUDIT_DURATION**: Use this variable to report the audit events that occurred during the time interval that you specify in this variable. For example, if you set this variable to 7, the audit events that occurred only in last 7 hours are reported by the Audit Details attribute group. The default value of the **COLL_AUDIT_DURATION** variable is 24 hours.
- **COLL_AUDIT_COLLECTION_INTERVAL**: The threaded collection in the Audit Details attribute group provides specifications of all the database that are present on the SQL server instance. Use this variable to set the interval for this threaded collection. For example, if you set this variable to 7, a fresh set of database specifications is extracted from the SQL server instance after every 7 hours. The default value of the **COLL_AUDIT_COLLECTION_INTERVAL** variable is 24 hours.

Variable for enhancing the collection for the MS SQL Process Detail attribute group

To enhance the MS SQL Process Detail attribute group collection, use the **COLL_PROC_BLOCK_INTERVAL** variable with the following values:

- If **COLL_PROC_BLOCK_INTERVAL** = *0*, the collection for the Blocking Process Duration attribute, and the Blocking Resource Duration attribute is disabled.
- If **COLL_PROC_BLOCK_INTERVAL** = *x*, the interval between the two consecutive data collections is *x* minutes.

If the **COLL_PROC_BLOCK_INTERVAL** variable is not set in the CANDLE_HOME directory, the interval between the two consecutive data collections is 3 minutes.

Variable for excluding the locked objects from the data collection

If the queries that are sent for the Database Detail, Filegroup Details, Database Mirroring, and Device Detail workspaces take long to execute, use the **COLL_DBCC_NO_LOCK** variable to run a query with the value **WITH (NOLOCK)**. This variable causes the query not to wait in the queue when an object on which the query is run is locked.

Variable for setting the sorting criteria for the rows returned by the Table Details attribute group

The rows that are returned by the Table Details attribute group are sorted in a descending order depending on the value that is set for the **COLL_TBLD_SORTBY** variable. The default value for the **COLL_TBLD_SORTBY** variable is FRAG (fragmentation percent). The valid values are: ROWS (number of rows in a tables), SPACE(space used by the table), and OPTSAGE (the optimizer statistics age of the table).

Variable for enhancing the collection for the MS SQL Problem Detail and Problem Summary attribute groups

- **COLL_ALERT_SEV**: Use this variable to set the severity level of the error messages that are displayed in the Problem Detail and Problem Summary attribute groups. Error messages, which have a severity level that is equal to or greater than the value mentioned in this variable, are displayed in the Problem Detail and Problem Summary attribute groups. For example, if you set the value of this variable to 10, the error messages with severity level 10 or greater are displayed in the Problem Detail and Problem Summary attribute groups. If you do not specify a value for this variable, the error messages, which have a severity level that is equal to or greater than 17, are displayed in the Problem Detail and Problem Summary attribute groups.
- **COLL_SINCE_ERRORLOG_RECY**: Use this variable to monitor only the high severity errors in the current ERRORLOG file. If you do not specify a value for this variable, the value of the variable is 0, which means that for collecting the data, the Problem Summary attribute group also considers the high severity

errors that are read from the previous ERRORLOG file. To monitor only the high severity errors in the current ERRORLOG file, set the value of this variable to 1.

Variables for setting the timeout interval

To set the timeout interval for the Microsoft SQL Server agent, you can use the following environment variables:

- **WAIT_TIMEOUT**: Use this variable to set the wait timeout interval for the Microsoft SQL Server agent. If any attribute group takes more than 45 seconds to collect data, then the agent might hang or situations might be incorrectly triggered. Check the log for the attribute groups that take more than 45 seconds to collect the data, and use the **WAIT_TIMEOUT** variable to increase the wait time between the agent process and the collector process.
- **COLL_DB_TIMEOUT**: Use this variable to define the wait interval (in seconds) for any request such as running a query on the existing SQL server connection to complete before returning to the application. If you set this value to 0, then there is no timeout. If you do not specify a value for this variable, the agent waits 15 seconds before returning to the application.

Variables for setting the properties of the collector log files

To set the properties of the collector log files, you can use the following environment variables:

- **COLL_WRAPLINES**: Use this variable to specify the maximum number of lines in a col.out file. The default value of this variable is 90,000 lines (about 2 MB).
- **COLL_NUMOUTBAK**: Use this variable to specify the number of backup copies of the collector log files that you want to create. By default, five backup copies of the collector log file are created. The backup file is named *.out. When this backup file is full, the file is renamed to *.ou1 and the latest logs are written in the *.out file. In this manner, for five backup files, the oldest logs are available in the *.ou5 file and the latest logs are available in the *.out file.

  You can create more than five backup copies of the collector log files by specifying one of the following values in the **COLL_NUMOUTBAK** variable:

  - For less than 10 backup files, specify the number of backup files that you want to create in the **COLL_NUMOUTBAK** variable. For example, if you specify 9 in the **COLL_NUMOUTBAK** variable, nine backup files will be created.
  - For more than 9 and less than 1000 backup files, in the **COLL_NUMOUTBAK** variable, specify the number of backup files preceded by a hyphen. For example, if you specify -352 in the **COLL_NUMOUTBAK** variable, three hundred and fifty-two backup files will be created.
- **COLL_DEBUG**: Use this variable to enable full tracing of the collector by setting the value of this variable to dddddddddd (10 times"d").

Variable for deleting the temporary files

COLL_TMPFILE_DEL_INTERVAL: Use this variable to specify the interval (in minutes) after which the KOQ_<timestamp> temporary files should be deleted. If you do not specify a value for this variable, the value of the variable is 0, which means that the temporary files must be deleted immediately.

Enabling cursors in the koqcoll.ctl file

On the **Koqcoll** tab in the SQL Agent Trace Configuration utility, you can enable or disable one or more cursors in the koqcoll.ctl file.

Procedure

1. To run the utility, complete one of the following steps:
   - For a 64-bit agent, go to the *candle_home*\TMAITM6_x64 directory.
   - For a 32-bit agent, go to the *candle_home*\TMAITM6 directory.

2. To run the SQL Agent Trace Configuration utility, double-click the `koqCtlEditUtility.exe` file.
3. To enable a cursor, click the **Koqcoll** tab, and select a cursor in the **Disabled** area.
4. To enable the cursor, click **Enable**.

   Tip: You can enable multiple cursors simultaneously. To do a multiple selection, press Ctrl and select the cursors that you want to enable. To do a contiguous selection, click a cursor from the list, press Shift and click the last cursor in the list.
5. To save the configuration settings, click **Apply**.
6. To restore the configuration values of the cursors in the `koqcoll.ctl` file to the values that existed before running the utility for the first time, click **Restore Default**.

Remote installation and configuration

You can remotely install and configure the agent.

To remotely install and configure the agent, see the "Deploying non-OS agents" topic in the *IBM Tivoli Monitoring Installation and Setup Guide*. Also, use the agent-specific configuration information in this section and in Table 5 on page 25 for the following interfaces:

- Tivoli Enterprise Portal
- **`tacmd`** command line

In the New Managed System Configuration window, use the settings in Table 5 on page 25 for the **Tivoli Enterprise Portal Database Server Properties** tab and the **Agent** tab **Run as**information.

To remotely deploy this monitoring agent by using the command line, use the procedure “Deploying through the command line,” in the *IBM Tivoli Monitoring Installation and Setup Guide*. To use the **`tacmd addSystem`** command, see the agent-specific configuration information in Table 5 on page 25. The *IBM Tivoli Monitoring Command Reference* has complete information about the **`tacmd addSystem`** command.

Use the **`-t`** or **`--type TYPE`** parameter to specify the Microsoft SQL Server agent that you are configuring: OQ.

Specify the properties with the **`-p`** or **`-property`** option.

For example:

```
tacmd addSystem -t OQ -n Primary:myhostname:NT -p
DBSETTINGS.db_login=sa
DBSETTINGS.db_password=sapwd
DBSETTINGS.db_home="C:\Program Files\Microsoft SQL Server\MSSQL"
DBSETTINGS.db_errorlog="C:\Program Files\Microsoft SQL Server\MSSQL\LOG\ERRORLOG"
DBSETTINGS.db_winauth=1
DBSETTINGS.db_lldbconn=0
DBSETTINGS.db_include_state=0
DBSETTINGS.db_monitorall=1
DBSETTINGS.db_list=#master##temp%
INSTANCE=MyServer
DBSETTINGS.db_tbld_weekly=1
DBSETTINGS.db_tbld_coll_stime_hour=23
DBSETTINGS.db_tbld_coll_stime_minute=59
```

The parameters in the example are shown on separate lines for clarity. When typing the command, type all of the parameters on one line.

Reconfiguration

You can reconfigure the agent locally or remotely.

If you need to reconfigure the Microsoft SQL Server agent, ensure that the steps for installing the monitoring agent in the *IBM Tivoli Monitoring Installation and Setup Guide* are completed.

Local reconfiguration

If you are reconfiguring an instance locally, use the Manage Tivoli Enterprise Monitoring Services window. See Table 5 on page 25 for the configuration settings.

Remote reconfiguration

If you are reconfiguring remotely, use one of the following interfaces:

- tacmd command line

  Use the **configureSystem** command. See the *IBM Tivoli Monitoring Command Reference Guide* for complete information about this command.

  When reconfiguring, enter the information for the property that you are changing as well as the INSTANCE property.
- Tivoli Enterprise Portal

See Table 5 on page 25 for the configuration settings.

Starting and stopping the Microsoft SQL Server agent

You can start and stop the Microsoft SQL Server agent locally or remotely.

Table 6 shows which interfaces you can use on Windows operating systems locally or remotely to start the monitoring agent.

Table 6. Interfaces for starting and stopping the Microsoft SQL Server agent locally and remotely

Local	Remote
• Manage Tivoli Enterprise Monitoring Services • tacmd startAgent • tacmd stopAgent • tacmd restartAgent	• Tivoli Enterprise Portal • tacmd startAgent • tacmd stopAgent • tacmd restartAgent

When you use the Manage Tivoli Enterprise Monitoring Services window, ensure that you enter only ASCII characters in the fields.

For information about using the Tivoli Enterprise Portal to start and stop the agent, see the "Starting and stopping components" topic in the *IBM Tivoli Monitoring Installation and Setup Guide*.

In the following examples, the **tacmd** command is used to start, stop, or restart the Microsoft SQL Server agent instances that are configured for the Microsoft SQL Server on Windows operating systems, where *myhostname* is the host name of the computer where the monitoring agent is running.

- Local
  - `tacmd startAgent -t oq`
  - `tacmd stopAgent -t oq`
  - `tacmd restartAgent -t oq`
- Remote
  - `tacmd stopagent -t oq -m InstanceName:hostname:MSS`
  - `tacmd startagent -t oq -m InstanceName:hostname:MSS`
  - `tacmd restartagent -t oq -m InstanceName:hostname:MSS`

The command for restarting the monitoring agent remotely can be `tacmd restartagent -t oq -n` *`INST1`*`:`*`ESX1-PG3-2`*`:MSS`, where *INST1* is the Microsoft SQL Server agent instance name, and *ESX1-PG3-2* is the host name of the computer where the monitoring agent is running.

For information about using the **`tacmd`** commands, see the *IBM Tivoli Monitoring Command Reference*.

Setting up the Microsoft SQL Server agent in a cluster environment

You can setup the Microsoft SQL Server agent in a cluster environment.

The *IBM Tivoli Monitoring Installation and Setup Guide* contains an overview of clustering.

The SQL Server and the Microsoft SQL Server agent both support multiple instances running on the same node. One agent does not interfere with another while running on the same node.

When the SQL Resource Group is moved from node to node, the server down situation event fires. This event is caused by the Microsoft SQL Server agent coming online faster than the SQL Server. When the SQL Server comes online, the event clears.

Do not use the Start and Stop Take Action commands for the Microsoft SQL Server agent because they conflict with the actions that are taken by the cluster server.

You can install and configure the monitoring agent in a cluster environment by completing any of the following steps:

- "Installing and configuring the monitoring agent manually"
- "Installing and configuring the agent by using the cluster utility" on page 45

Requirements

You must ensure that all the requirements for setting up the agent in a cluster environment are met.

You can set up a cluster environment for the following versions of the Microsoft SQL Server:

- Microsoft SQL Server 2005
- Microsoft SQL Server 2008
- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2012
- Microsoft SQL Server 2014
- Microsoft SQL Server 2016

On Windows systems, IBM Tivoli Monitoring requires that monitoring agents are installed in the same directory path as the OS agent. Therefore, each node in a cluster must have installed all monitoring agents (on the nodes system disk) that are required to support the cluster applications that can run on that cluster node.

Installing and configuring the monitoring agent manually

You can install the Microsoft SQL Server agent on each node of a cluster where it is possible for the Microsoft Virtual Servers to run.

Before you begin

In addition to installing and setting up the Microsoft SQL Server agent, you must ensure that the following prerequisites for cluster environment are met:

- Setting the **CTIRA_HOSTNAME** variable to a common value for all monitoring agents (usually the cluster name)

- Setting the **CTIRA_HIST_DIR** variable to a common disk location if history is stored at the monitoring agents (if history for the Microsoft SQL Server agent is configured to be stored at the monitoring agent)
- Creating a monitoring agent cluster resource in the Resource Group of the Virtual Server 3

Procedure

1. Setting **CTIRA_HOSTNAME**
   - Because there can be multiple instances of the Microsoft SQL Server agent, each instance must be configured with a **CTIRA_HOSTNAME** . Set the **CTIRA_HOSTNAME** environment variable to the name of the Microsoft Cluster Server cluster for all monitoring agents running in that cluster. By setting the **CTIRA_HOSTNAME** variable for all agents in the cluster to the same name, you can navigate to all of the monitoring agents for that cluster in the Tivoli Enterprise Portal.
   - When deciding on the value for the **CTIRA_HOSTNAME** variable, consider that the managed system name is comprised of three parts: the **CTIRA_SUBSYSTEM_ID** variable, the **CTIRA_HOSTNAME** variable, and the **CTIRA_NODETYPE** variable. Also, the name is limited to 31 characters. By default for the Microsoft SQL Server Agent, the **CTIRA_NODETYPE** variable is set to `MSS`, and **`CTIRA_SUBSYSTEM_ID`** is set to the *Microsoft SQL Virtual Server* name. The **CTIRA_SUBSYSTEM_ID** variable is used to distinguish the multiple instances of the Microsoft SQL Server agent.
2. Setting the **CTIRA_HIST_DIR** variable
   - If history for the Microsoft SQL Server agent is configured to be stored at the monitoring agent, each instance of the monitoring agent must be configured with a common **CTIRA_HIST_DIR** variable that points to a shared disk directory.
   - If history is stored at the Tivoli Enterprise Management Server, setting the **CTIRA_HIST_DIR** variable is not required. Storing history at the Tivoli Enterprise Management Server puts a higher burden on that server.
3. Setting the **COLL_HOME** variable

   If you want to collect data for attribute groups that use configuration files at a shared location, and these files are used by multiple cluster nodes, set the **`COLL_HOME`** variable to `*X*:\shared-location`, where *X* is a shared drive that is accessible to the cluster nodes. For example, you can set the value for the **COLL_HOME** variable when you are defining the configuration settings for the MS SQL Table Detail attribute group or MS SQL Error Event Details attribute group.
4. Creating a monitoring agent cluster resource
   - Set each Microsoft SQL Server agent startup parameter to manual so the cluster resource can control the starting and stopping of the monitoring agent.
   - After these parameters are set for each Microsoft SQL Server agent instance, you must create the cluster resources to control the monitoring agents. Each Microsoft SQL Server agent is comprised of two Windows Services: `KOQAGENT`*x* and `KOQCOLL`*x*, where *x* is the agent instance number.

Example

Examples of Windows Services names:

- `Monitoring Agent for Microsoft SQL Server: SQLTEST`
- `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST`
- `Monitoring Agent for Microsoft SQL Server: SQLTEST2`
- `Monitoring Agent for Microsoft SQL Server: Collector SQLTEST2`

Creating a resource

You can create a resource for the Windows Server 2003 operating system.

Procedure

1. Click **Start > Administrative Tools > Cluster Administrator**.
2. Select the group for the **Instance that is being worked > SQLTEST**.

3. Right-click the group, and then click **New > Resource**.
4. In the New Resource window, complete the fields as follows:

   Name: `KOQAGENT_`*`instance_name`*

   Description: *`resource_description`*

   Resource type: `Generic Service`

   Group: `SQLTEST`
5. In the Possible Owners window, accept the default of **all Available Nodes**.
6. In the Dependencies window, do not add any dependencies on history disk.
7. In the Generic Service Parameters window, complete the fields as follows:

   Service name: `KOQAGENT_`*`instance_name`*

   Start Parameters: `-Hkey "KOQ\610\SQLVS1"`
8. Click **Finish**.
9. Click **Advanced**.
10. Clear the **Affect the group** check box.
11. Right-click the group, and then click **New > Resource**.
12. Complete the following information:

    Name: `KOQCOLL_`*`instance_name`*

    Resource Type: `Generic Service`

    Group: `SQLTEST`
13. In the Possible Owners window, accept the default of **all Available Nodes**.
14. In the Dependencies window, do not add any dependencies on history disk.
15. In the Generic Service Parameters window, complete the fields as follows:

    Name: `KOQCOLL_`*`instance_name`*

    Start Parameters: `-Hkey "KOQ\610\SQLVS1"`
16. Click **Finish**.
17. Click **Advanced**.
18. Clear the **Affect the group** check box.
19. Bring the two agent resources online.
20. Repeat these steps for the other instances of the monitoring agent in the cluster environment.

Creating a service

You can create a service for Windows 2008 systems.

Procedure

1. Click **Start > Administrative Tools > Failover Cluster Management**.
2. Expand **Failover Cluster Management**.
3. Expand **Services and Applications** and right-click the SQL instance that you want to configure.
4. Click **Add a resource > Generic Service**. The New Resource Wizard opens.
5. On the Select Service page, select **Monitoring Agent for Microsoft SQL Server**, and click **Next**.

   Note: To add the Microsoft SQL Server- Collector service, select **Monitoring Agent for Microsoft SQL Server- Collector *instance_name*** on the Select Service page, and click **Next**.
6. On the Confirmation page, check the details, and click **Next**.
7. On the Summary page, click **Finish**.
8. Right-click the service that you added, and click **Properties**.
9. In the **Dependencies** tab, select **SQL Server** or **Monitoring Agent for Microsoft SQL Sever** from the **Resources** list.

10. Click **Apply** > **OK**.
11. Right-click the service that you created, and click **Bring this resource online**.

Configuration changes after installing the agent

After installing the agent in a cluster environment, you can make the configuration changes.

After control of the agent cluster resource is given to the cluster server, you must take the agent cluster resource offline to make configuration changes, or edit the agent variables on the node from which the agent cluster resource runs. If the agent cluster resource is not offline when the agent configuration utility attempts to take the agent offline, the cluster server notices that the monitoring agent went offline and attempts to bring the monitoring agent back online. When finished with the configuration changes for the monitoring agent, bring the agent cluster resource back online.

Installing and configuring the agent by using the cluster utility

You can use the cluster utility to add multiple SQL Server agent instances to a cluster group in a cluster environment.

The cluster utility automatically adds the agent service and the collector service of each SQL Server agent instance as a generic service resource to the cluster group. You can use the cluster utility to complete the following tasks:

- Adding an SQL Server agent instance to the cluster
- Updating an existing SQL Server agent instance in a cluster
- Removing an SQL Server agent instance from a cluster

Prerequisites for using the cluster utility

You must ensure that your system environment meets the prerequisites for running the cluster utility.

Ensure that the following prerequisites are met:

- Run the cluster utility on a computer that has at least one group in the cluster environment.
- Start the remote registry service for all nodes in the cluster.
- You must have the cluster manager authorization to access the cluster utility.

Adding an SQL Server agent instance to the cluster

You can use the cluster utility to add an SQL Server agent instance to a cluster group in a cluster environment.

Procedure

1. To run the utility, complete one of the following steps:
   - For a 64-bit agent, go to the `*candle_home*\TMAITM6_x64` directory.
   - For a 32-bit agent, go to the `*candle_home*\TMAITM6` directory.
2. To run the Cluster Utility, double-click the `KoqClusterUtility.exe`.
3. In the SQL **Server Agent Instances Available** area, select an SQL Server agent instance, and click **Add**.
4. In the Select cluster group name window, select a cluster group. The cluster group that you select must be the SQL Server instance that is monitored by the SQL Server agent.
5. In the Select Path for Shared Location window, navigate to the path where the agent and collector logs are stored. If you do not select the path, by default, the `CANDLEHOME/TMAITM6(_x64)/logs` location is selected for storing the agent and collector logs.
6. To add the SQL Server agent instance to the cluster environment, click **OK**. The activity logs of the cluster utility are displayed in the History pane.

Updating an existing SQL Server agent instance in a cluster

You can use the cluster utility to update the location where the agent and collector logs are stored for an SQL Server instance in a cluster.

Procedure

1. To update an existing Microsoft SQL Server agent instance, open the Cluster Utility window.
2. In the **SQL Server Agent Instances Configured** area, select an SQL Server agent instance, and click **Update**.
3. In the Set Path for Shared Location window, navigate to the path where the agent and collector logs are stored. If you do not select the path, the agent and collector logs are stored at the location that was set while adding the SQL Server agent instance in a cluster.
4. Click **OK**. The activity logs of the cluster utility are displayed in the History pane.

Removing an SQL Server agent instance from a cluster

You can use the cluster utility to remove an SQL Server agent instance from a cluster group.

Procedure

1. Open the Cluster Utility window.
2. In the **SQL Server Agent Instances Configured** area, select an SQL Server agent Instance, and click **Remove**.
3. In the **Please Confirm Action** dialog box, click **Yes** to delete the SQL Server agent instance from the cluster. The activity logs of the cluster utility are displayed in the History pane.

Appendix. ITCAM for Microsoft Applications documentation library

Various publications are relevant to the use of ITCAM for Microsoft Applications.

For information about how to access and use the publications, see **Using the publications** (http://www.ibm.com/support/knowledgecenter/SSTFXA_6.3.0.1/com.ibm.itm.doc_6.3/common/using_publications.htm).

To find publications from the previous version of a product, click **Previous versions** under the name of the product in the **Contents** pane.

Documentation for this product is in the ITCAM for Microsoft Applications Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSDKXQ_6.3.1/com.ibm.itcamms.doc_6.3.1/welcome_msapps631.html)

- Quick Start Guides
- Offering Guide
- Download instructions
- Links to Prerequisites
- Installation and Configuration Guide for each agent
- Link to Reference information for each agent
- Link to Troubleshooting Guide for each agent

Prerequisite publications

To use the information about the agents effectively, you must have some prerequisite knowledge.

See the following information at the IBM Tivoli Monitoring Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSTFXA_6.3.0.2/com.ibm.itm.doc_6.3fp2/welcome.htm) to gain prerequisite knowledge:

- *IBM Tivoli Monitoring Administrator's Guide*
- *IBM Tivoli Monitoring Installation and Setup Guide*
- *IBM Tivoli Monitoring High Availability Guide for Distributed Systems*
- IBM Tivoli Monitoring: Installation and Configuration Guides for the following agents: Operating System agents and Warehouse agents
- IBM Tivoli Monitoring: User's Guides for the following agents: Agentless OS monitors, Log file agent, System p agents, Systems Director base agent
- *IBM Tivoli Monitoring Agent Builder User's Guide*
- *IBM Tivoli Monitoring Command Reference*
- *IBM Tivoli Monitoring: Messages*
- *IBM Tivoli Monitoring Troubleshooting Guide*
- IBM Tivoli Monitoring: References for the following agents: Operating System agents and Warehouse agents
- IBM Tivoli Monitoring: Troubleshooting Guides for the following agents: Operating System agents and Warehouse agents
- *Tivoli Enterprise Portal User's Guide*

Related publications

The publications in related knowledge centers provide useful information.

See the following knowledge center at IBM Tivoli Monitoring Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSTFXA_6.3.0.2/com.ibm.itm.doc_6.3fp2/welcome.htm):

- Tivoli Monitoring
- Tivoli Application Dependency Discovery Manager
- Tivoli Business Service Manager
- Tivoli Common Reporting
- Tivoli Enterprise Console
- Tivoli Netcool/OMNIbus

Tivoli Monitoring Community on Service Management Connect

Service Management Connect (SMC) is a repository of technical information that is organized by communities.

Access Service Management Connect at https://www.ibm.com/developerworks/servicemanagement.

For information about Tivoli products, see the Application Performance Management community (http://www.ibm.com/developerworks/servicemanagement/apm/index.html).

Connect, learn, and share with Service Management professionals. Get access to developers and product support technical experts who provide their perspectives and expertise. You can use SMC for these purposes:

- Become involved with transparent development, an ongoing, open engagement between other users and IBM developers of Tivoli products. You can access early designs, sprint demonstrations, product roadmaps, and prerelease code.
- Connect one-on-one with the experts to collaborate and network about Tivoli and the (enter your community name here) community.
- Read blogs to benefit from the expertise and experience of others.
- Use wikis and forums to collaborate with the broader user community.

Other sources of documentation

You can obtain additional technical documentation about monitoring products from other sources.

See the following sources of technical documentation about monitoring products:

- IBM Integrated Service Management Library (http://www.ibm.com/software/brandcatalog/ismlibrary/) is an online catalog that contains integration documentation as well as other downloadable product extensions.
- IBM Redbook publications (http://www.redbooks.ibm.com/) include Redbooks® publications, Redpapers, and Redbooks technotes that provide information about products from platform and solution perspectives.
- Technotes (http://www.ibm.com/support/entry/portal/software), which are found through the IBM Software Support website, provide the latest information about known product limitations and workarounds.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (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.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

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 jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide 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 Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

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.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

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-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 actual people or business enterprises 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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

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. 2005, 2016.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth in the following paragraphs.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's user name for purposes of session management, authentication, and single sign-on configuration. These cookies can be disabled, but disabling them will also likely eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at http://www.ibm.com/software/info/product-privacy.

Index

Printed in USA

SC27-5660-01