Tivoli® Monitoring: Linux OS Agent

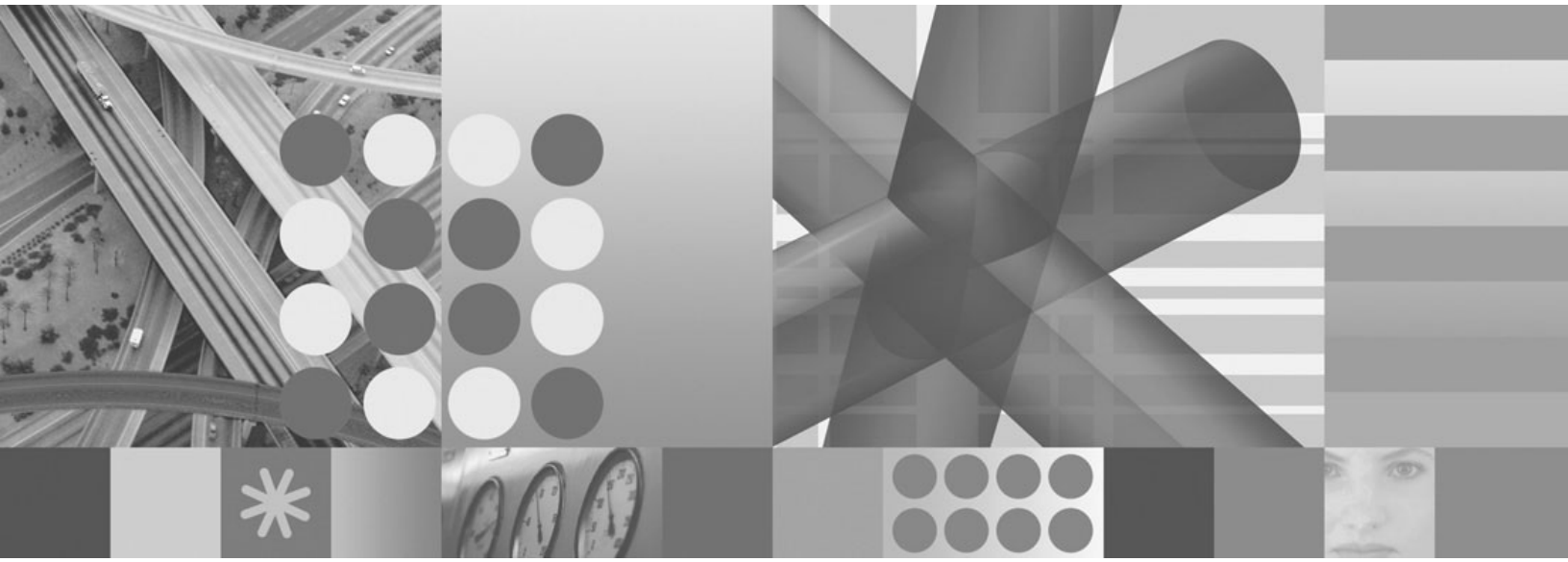
Version 6.2.2

User's Guide

SC32-9447-03

Tivoli® Monitoring: Linux OS Agent

Version 6.2.2



User's Guide

SC32-9447-03

Note
Before using this information and the product it supports, read the information in "Notices" on page 243.

This edition applies to version 6.2.2 of the IBM Tivoli Monitoring: Linux OS Agent (5724-C04) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Tables

Chapter 1. Overview of the Monitoring Agent for Linux OS

The Monitoring Agent for Linux OS provides you with the capability to monitor Linux, and to perform basic actions with Linux. This chapter provides a description of the features, components, and interface options for the Monitoring Agent for Linux OS.

IBM Tivoli Monitoring overview

IBM Tivoli Monitoring is the base software for the Monitoring Agent for Linux OS. 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 do the following:

- 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 perform actions, schedule work, and automate manual tasks.

The Tivoli Enterprise Portal is the interface for IBM Tivoli Monitoring products. By providing a consolidated view of your environment, the Tivoli Enterprise Portal permits you to monitor and resolve performance issues throughout the enterprise.

See the IBM Tivoli Monitoring publications listed in Appendix F, “Documentation library,” on page 237 for complete information about IBM Tivoli Monitoring and the Tivoli Enterprise Portal.

Features of the Monitoring Agent for Linux OS

As part of the Tivoli Enterprise Portal for Distributed Systems, the Monitoring Agent for Linux OS offers a central point of management of Linux-based environments. It provides a comprehensive means for gathering exactly the information you need to detect problems early and to prevent them. Information is standardized across all systems, and you can monitor servers from a single workstation. The Tivoli Enterprise Portal lets you easily collect and analyze specific information.

The Monitoring Agent for Linux OS is an intelligent, remote monitoring agent that resides on managed resources. It assists you in anticipating trouble and warns systems administrators when critical events take place on their systems. With the Monitoring Agent for Linux OS, systems administrators can set threshold levels and flags as desired to alert them when the system reaches these thresholds.

For Tivoli Enterprise Portal, information appears in named workspaces. Tivoli Enterprise Portal refers to this tabular format for information as a table view. Information can also be displayed in the workspace as charts, graphs, or other formats that you can specify.

The Monitoring Agent for Linux OS provides the following benefits:

- Simplifies application and system management by managing applications, platforms, and resources across your environment.
- Helps to increase 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 platforms by supporting a wide variety of platforms.
- Improves system performance by letting you integrate, monitor, and manage your system, network, console, and mission-critical applications. A monitoring agent alerts the Tivoli Enterprise Monitoring Server when conditions on the system network meet threshold-based conditions. These alerts notify your systems administrator to limit and control database usage. You can view data gathered by the Tivoli Enterprise Monitoring Server in tables and charts for the status of your distributed database systems.
- Enhances efficiency by monitoring diverse platforms and networks from a single PC screen. Depending on your Tivoli Enterprise Portal configuration, you can collect and monitor data across platforms. Management agents gather and filter status information at the managed resource rather than at the hub, eliminating unnecessary data transmission and sending only data that is relevant to changes in status conditions. The Monitoring Agent for Linux OS helps you monitor and gather the consistent, accurate, and timely information you require to effectively perform your job.

New in this release

For version 6.2.1 of the Monitoring Agent for Linux OS, the following enhancements have been made:

- For the migration of agents to dynamic affinities, there are new silent installation parameters, and changes to the command line tools and the user interface. For more information see the *IBM Tivoli Monitoring Installation and Setup Guide*.
- New Take Action:
  - AMS Start Agent Instance

Monitoring Agent for Linux OS components

After you install the Monitoring Agent for Linux OS (product code "klz" or "lz") as directed in the *IBM Tivoli Monitoring Installation and Setup Guide*, you have an environment with a client, server, and monitoring agent implementation for IBM Tivoli Monitoring that contains the following components:

- Tivoli Enterprise Portal client with a Java-based user interface for viewing and monitoring your enterprise.
- Tivoli Enterprise Portal Server that is placed between the client and the Tivoli Enterprise Monitoring Server and enables retrieval, manipulation, and analysis of data from the monitoring agents.
- Tivoli Enterprise Monitoring Server, which acts as a collection and control point for alerts received from the monitoring agents, and collects their performance and availability data.
- Monitoring Agent for Linux OS, which collects and distributes data to a Tivoli Enterprise Monitoring Server. This component also embeds the Agent Management Services function.
- Operating system agents and application agents installed on the systems or subsystems you want to monitor. These agents collect and distribute data to the Tivoli Enterprise Monitoring Server.

- Tivoli Data Warehouse for storing historical data collected from agents in your environment. The data warehouse is located on a DB2®, Oracle, or Microsoft® SQL database. To collect information to store in this database, you must install the Warehouse Proxy agent. To perform aggregation and pruning functions on the data, install the Warehouse Summarization and Pruning agent.
- Tivoli Enterprise Console event synchronization component for synchronizing the status of situation events that are forwarded to the event server. When the status of an event is updated because of IBM® Tivoli Enterprise Console® rules or operator actions, the update is sent to the monitoring server, and the updated status is reflected in both the Situation Event Console and the Tivoli Enterprise Console event viewer. For more information, see *IBM Tivoli Monitoring Installation and Setup Guide*.

User interface options

Installation of the base software and other integrated applications provides the following interfaces that you can use to work with your resources and data:

Tivoli Enterprise Portal browser client interface
The browser interface is automatically installed with Tivoli Enterprise Portal. To start Tivoli Enterprise Portal in your Internet browser, enter the URL for a specific Tivoli Enterprise Portal browser client installed on your Web server.

Tivoli Enterprise Portal desktop client interface
The desktop interface is a Java-based graphical user interface (GUI) on a Windows® workstation.

IBM Tivoli Enterprise Console
Event management application

Manage Tivoli Enterprise Monitoring Services window
The window for the Manage Tivoli Enterprise Monitoring Services utility is used for configuring the agent and starting Tivoli® services not already designated to start automatically.

Chapter 2. Requirements for the monitoring agent

This chapter contains information about the following topics and procedures relevant to the installation and configuration of the Monitoring Agent for Linux OS.

In addition to the requirements described in the *IBM Tivoli Monitoring Installation and Setup Guide*, the Monitoring Agent for Linux OS has the requirements listed in Table 1 on page 6.

Table 1. System requirements for the Monitoring Agent for Linux OS

Operating system	**Linux®**
Operating system versions	Linux: • Linux on zSeries – RedHat Enterprise Linux AS 3 (31-bit or 64-bit) – RedHat Enterprise Linux AS 4 (31-bit or 64-bit) – RedHat Enterprise Linux AS 5 (31-bit or 64-bit) – SUSE Linux Enterprise Server 8 (31-bit or 64-bit) – SUSE Linux Enterprise Server 9 (31-bit or 64-bit) – SUSE Linux Enterprise Server 10 (31-bit or 64-bit) • Linux on Intel® (32-bit) – RedHat Enterprise Linux AS/ES 3 – RedHat Enterprise Linux AS/ES 4 – RedHat Enterprise Linux AS/ES 5 – SUSE Linux Enterprise Server 8 – SUSE Linux Enterprise Server 9 – SUSE Linux Enterprise Server 10 – RedFlag 4.1 – Asian Linux 2.0 • Linux on pSeries – RedHat Enterprise Linux AS 4 – RedHat Enterprise Linux AS 5 – SUSE Linux Enterprise Server 9 – SUSE Linux Enterprise Server 10 • Linux on IA64 (Itanium®) – RedHat Enterprise Linux AS 4[1] – RedHat Enterprise Linux AS 5[1] – SUSE Linux Enterprise Server 9 [1] – SUSE Linux Enterprise Server 10[1] – Asian Linux 2 • Linux on x86-64 – RedHat Enterprise Linux AS 4[1] – RedHat Enterprise Linux AS 5[1] – SUSE Linux Enterprise Server 9[1] – SUSE Linux Enterprise Server 10[1] – Asian Linux 2 The Linux version must support the Korn shell (ksh) and Motif Window Manager (libmotif) for installation of the monitoring agent.
Memory	• 30 MB RAM for the Monitoring Agent for Linux OS

Table 1. System requirements for the Monitoring Agent for Linux OS (continued)

Operating system	**Linux**®
Disk space	The Monitoring Agent for LINUX OS needs 180 MB of disk space in the file system where it is to be installed through the local install method. It needs 135 MB of disk space in the /tmp filesystem and 185 MB of disk space in the file system where the agent is to be installed through the **tacmd createNode** command. It needs 268 MB of disk space when it is updated using the command **tacmd updateAgent**. For historical data disk space information, see "Disk capacity planning for historical data" on page 114.
Other requirements	• IBM Tivoli Monitoring v6.2.2 agents require at least a v6.2.2 hub monitoring server and portal server. IBM Tivoli Monitoring v6.2.1 hub monitoring servers and portal servers do not support v6.2.2 monitoring agents. IBM Tivoli Monitoring v6.2.1 monitoring agents work with both v6.2.1 and v6.2.2 environments. • The monitoring agent must have the permissions necessary to perform requested actions. For example, if the user ID you used to log onto the system to install the monitoring agent (locally or remotely) does not have the permission to perform a particular action being monitored by the monitoring agent (such as running a particular command), the monitoring agent will be unable to perform the requested action. • Linux versions require some compatibility libraries to be installed for the agent to work correctly. The latest versions of libstdc++, libgcc, and compat-libstdc++, are required for the agent to work correctly. [2]

Notes:

1. In native 64-bit mode, not tolerance mode.
2. See Table 2 for the minimum version required for these libraries.

Table 2. Required Linux libraries

Architecture	**libstdc++**	**libgcc**	**compat-libstdc++**
li6243/li6246 32bit agent for Linux Intel kernel 2.4 (RHEL3,SLES8)	libstdc++-2.96-98	N/A	compat-libstdc++-6.2-2.9.0.16
li6263/li6266 32bit agent for Linux Intel kernel 2.6 (RHEL4, RHEL5, SLES9,SLES10)	libstdc++-3.3.3-43.41	libgcc-4.1-4.1.2_20070115-0.2	N/A
lx8266 64bit agent for Linux x64 kernel 2.6	libstdc++-3.4.4-2	libgcc-3.4.4-2	compat-libstdc++-33-3.2.3-47.3
lia266 64bit agent for Linux IA64 kernel 2.6	libstdc++-3.2.2-23	libgcc-3.2.2-23	N/A
lpp266 64bit agent for Linux PPC kernel 2.6	libstdc++-3.3.3-43.41	libgcc-3.3.3-43.41	N/A

Table 2. Required Linux libraries (continued)

Architecture	libstdc++	libgcc	compat-libstdc++
ls3243 31bit agent for zLinux kernel 2.4 (RHEL3,SLES8)	libstdc++-3.2.2-54	libgcc-3.2.2-54	N/A
ls3246 64bit agent for zLinux kernel 2.4 (RHEL3,SLES8)	libstdc++-3.2.2-54	libgcc-3.2.2-54	N/A
ls3263 31bit agent for zLinux kernel 2.6 (RHEL4, RHEL5, SLES9,SLES10)	libstdc++-3.3.3-43.34	libgcc-3.3.3-43.34	N/A
ls3266 64bit agent for zLinux kernel 2.6 (RHEL4, RHEL5, SLES9,SLES10)	libstdc++-3.3.3-43.34	libgcc-3.3.3-43.34	N/A

Note: For the most current information about the operating systems that are supported, see the following URL:

`http://www-306.ibm.com/software/sysmgmt/products/support/Tivoli_Supported_Platforms.html`

When you get to that site, click **Tivoli platform and database support matrix link** at the bottom of the window.

Silent installation: If you are performing a silent installation using a response file, see the IBM Tivoli Monitoring Installation and Setup Guide, "Performing a silent installation of IBM Tivoli Monitoring."

Naming instances

If you have multiple instances of a monitoring agent, you must decide how to name the monitoring agents. This name is intended to uniquely identify that monitoring agent. The agent's default name is composed of three qualifiers:

- Optional instance name
- Machine network hostname
- Agent product node type

An agent name truncation problem can occur when the network domain name is included in the network hostname portion of the agent name. For example, instead of just the hostname myhost1 being used, the resulting hostname might be myhost1.acme.north.prod.com. Inclusion of the network domain name causes the agent name in the example above to expand to SERVER1:myhost1.acme.north.prod.com:KXX. This resulting name is 39 characters long. It is truncated to 32 characters resulting in the name SERVER1:myhost1.acme.north.prod.

The agent name truncation is only a problem if there is more than one monitoring agent on the same system. In this case, the agent name truncation can result in collisions between agent products attempting to register using the same truncated name value. When truncated agent names collide on the same system, this can lead to Tivoli Enterprise Monitoring Server problems with corrupted EIB tables. The

agent name collision in the Tivoli Enterprise Monitoring Server might cause a registered name to be associated with the wrong product.

In general, create names that are short but meaningful within your environment. Use the following guidelines:

- Each name must be unique. One name cannot match another monitoring agent name exactly.
- Each name must begin with an alpha character.
- Do not use blanks or special characters, including $, #, and @.
- Each name must be between 2 and 32 characters in length.
- Monitoring agent naming is case-sensitive on all operating systems.

See "Unique names for monitoring components" on page 223 for more information about creating unique names.

Running as a non-Administrator user

The Monitoring Agent for Linux OS can be run by a non-Administrator user (a non-root user), however some functionality becomes unavailable. The Machine BIOS information uses the dmidecode executable to extract the relevant information. This Linux provided executable must be run by the Administrator user to extract BIOS information. This attribute group does not report data if the agent is not run by the Administrator user. This information is also used by Tivoli Application Dependency Discovery Manager.

A non-Administrator user can only access the directories that it has permissions to read. Therefore, functionality of the File Information attribute group might be reduced.

For Agent Management Services, the watchdog cannot stop or start any agent that it does not have privileges to stop or start.

Using Agent Management Services

There are two watchdog monitors that run as part of the Monitoring Agent for Linux. One monitor runs as part of the OS Monitoring Agent process, which is referred to as the *Agent Watchdog*. The other watchdog monitor runs as a separate process named 'kcawd'. The kcawd process is also called the *Agent Management Services Watchdog*. This is the watchdog that watches the OS Agent. It does this out-of-the-box, so as long as its Availability Status is showing 'Running' in the Agents' Runtime Status view of the Agent Management Services workspace. There is no setup or configuration required.

The Agent Watchdog monitors agent processes other than the OS Agent itself. Using the communication facility of the OS Agent, it is able to respond to Tivoli Enterprise Portal Desktop queries and Take Actions performed against these other agent processes. This is the data that is seen in the Agent Management Services workspace. In the Tivoli Enterprise Portal Desktop, the Agent Management Services workspace lists the agents that can be monitored by this watchdog running as part of the OS Agent. These are non-OS agents, so the Monitoring Agent for Linux is not listed in the workspace, except for in the Agents' Management Definitions view. One of the agents listed in the workspace is the Agent Management Services Watchdog. Its purpose is to monitor the OS Agent's availability.

The Agent Management Services Watchdog monitor is responsible for watching just the OS Monitoring Agent and restarting it if it goes down. It is enabled by default and does not need to be configured. It is started automatically when the Monitoring Agent for Linux is started. This watchdog does not have a communication facility, so it cannot report information to the Tivoli Enterprise Portal or respond to Take Actions. It is not an agent per se, but a separate process that always monitors the OS Monitoring Agent.

You can temporarily disable the Agent Management Services Watchdog by using the *InstallDir*/bin/itmcmd execute lz disarmWatchdog.sh command. This disables the Watchdog process for the OS Monitoring Agent and all Agent Management Services managed agents. If there is local administrative work to be performed, and you do not want the auto-restart of the agents to interfere with it, run the *InstallDir*/bin/itmcmd execute lz disarmWatchdog.sh command before proceeding. When the work is complete, recycle the OS Monitoring Agent to reenable Agent Management Services, or use the *InstallDir*/bin/itmcmd execute lz rearmWatchdog.sh command.

If you use the itmcmd interface to stop or start an Agent Management Services managed agent, its watchdog will be disabled if stopping the agent and enabled if starting the agent.

Chapter 3. Workspaces reference

This chapter contains an overview of workspaces, references for detailed information about workspaces, and descriptions of the predefined workspaces included in this monitoring agent.

About workspaces

A workspace is the working area of the Tivoli Enterprise Portal application window. At the left of the workspace is a Navigator that you use to select the workspace you want to see.

As you select items in the Navigator, the workspace presents views pertinent to your selection. Each workspace has at least one view. Some views have links to workspaces. Every workspace has a set of properties associated with it.

This monitoring agent provides predefined workspaces. You cannot modify or delete the predefined workspaces, but you can create new workspaces by editing them and saving the changes with a different name.

More information about workspaces

For more information about creating, customizing, and working with workspaces, see the *IBM Tivoli Monitoring User's Guide*.

For a list of the predefined workspaces for this monitoring agent and a description of each workspace, refer to thePredefined workspaces section below and the information in that section for each individual workspace.

Predefined workspaces

The following list shows the organization of the predefined workspaces provided with IBM Tivoli Monitoring: Linux OS Agent.

This agent also includes the following linked workspaces:

- Historical Summarized Availability
- Historical Summarized Availability Daily
- Historical Summarized Availability Hourly
- Historical Summarized Availability Weekly
- Historical Summarized Capacity
- Historical Summarized Capacity Daily
- Historical Summarized Capacity Hourly
- Historical Summarized Capacity Weekly
- Historical Summarized Performance
- Historical Summarized Performance Daily
- Historical Summarized Performance Hourly
- Historical Summarized Performance Weekly

Some predefined workspaces are not available from the Navigator tree item, but are accessed by selecting the link indicator next to a row of data in a view. Left-clicking a link indicator selects the default workspace associated with that link. Right-clicking a link indicator displays all linked workspaces that can be selected.

The remaining sections of this chapter contain descriptions of each of these predefined workspaces, which are organized alphabetically within the group.

Agent Management Services workspace

The Agent Management Services workspace contains views of data collected by the Agent Management Services component of the Monitoring Agent for Linux.

This workspace includes an Agents' Management Status view, an Agents' Runtime Status view, an Agents' Alerts view, and an Agents' Management Definitions view.

Agents' Management Log workspace

The Agents' Management Log workspace contains a list of monitoring agent log entries filtered on the Agent Management Services component. Use this workspace to see the operations being executed by Agent Management Services. They include:

- Agent added to system - CAP file found.
- Agent removed from system - CAP file removed.
- Agent now managed.
- Agent now unmanaged.
- Agent stop command received.
- Agent start command received.
- Agent restart failed.
- Agent started successfully.
- Agent stopped abnormally.
- Agent stopped successfully.

- Agent manual stop failed.
- Agent exceeded restart tries.
- Agent manual start failed.
- Agent not found.
- Agent exceeded policy defined memory threshold.
- Agent exceeded policy defined CPU threshold.

This workspace includes an Agents' Management Log view.

All Files workspace

The All Files workspace is reached by right-clicking the File Information navigator item in the Tivoli Enterprise Portal. The views are:

- File Size - Top Ten (bar chart)
- All Files (table view)

The File Size - Top Ten bar chart displays the sizes of the largest files. The All Files table provides file information.

Capacity Usage Information workspace

The Capacity Usage Information workspace reflects the “health” of your system by providing CPU, disk, and swap space usage statistics. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus ‘superseded’) with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- Disk Usage Averages (table view)
- Disk Space Usage (bar chart)
- Disk Usage Averages (bar chart)

The Disk Usage Averages table provides information on the system’s current disk usage. The Disk Space Usage bar chart displays the system's current disk usage. The Disk Usage Averages bar chart displays average disk usage information. With the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

CPU Averages workspace

The CPU Averages workspace is reached by right-clicking the Capacity Usage Information navigator item in the Tivoli Enterprise Portal. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus ‘superseded’) with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). The workspace is comprised of 3 views. The views are Current Overall CPU Usage bar chart, CPU Averages (Hourly Updates) chart, and CPU Usage Trends table.

Disk I/O Extended Rate workspace

The Disk I/O Extended Rate workspace is reached by right-clicking the System Information navigator item in the Tivoli Enterprise Portal. The Disk I/O Extended Rate workspace provides detailed input/output statistics and "calculations", including the queue length and size in sectors of read and write requests, the rate of those requests, and wait times associated with requests. This workspace has a superseded version that displays queries with signed 32-bit maximum value

(2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of two views. The views are:

- Disk I/O Extended Rate (table view)
- Disk Service Time (bar chart)
- Disk Activity (bar chart)

The Disk I/O Extended Rate table details the input/out data and calculated values associated with disk activity. The Disk Service Time chart displays average services time in minutes. The Disk Activity chart displays read and write sectors in seconds. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Note: The attributes associated with this workspace are only available for systems with a 2.4 (or higher) kernel.

Disk I/O Rate workspace

The Disk I/O Rate workspace is reached by right-clicking the System Information navigator item in the Tivoli Enterprise Portal. The Disk I/O Rate workspace provides input/output statistics, including the transfer rates, block read rates, and block write rates of your monitored systems. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of two views. The views are:

- Disk I/O Rate (table view)
- Disk I/O Rate (bar chart)

The Disk I/O Rate table includes transfer rates, block read rates, and block write rates for your monitored systems. The Disk I/O Rate chart provides "at a glance" rate details associated with disk reads, writes, and transfers. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Note: The attributes associated with this workspace are only available for systems with a 2.4 (or higher) kernel.

Disk Usage workspace

The Disk Usage workspace reflects the health of storage space within your monitored systems. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of four views. The views are:

- Space Used Percent (bar chart)
- Inodes Used Percent (bar chart)
- Disk Space (bar chart)
- Disk Usage (table view)

The three charts that comprise this workspace provide "at a glance" percentages of the space used, percentages of the inodes used, and amounts of disk space used/available for each monitored disk. The Disk Usage table captures this

information, as well as mount point and file system data, in tabular form. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

File Information workspace

The File Information workspace displays file information statistics. The views are:

- File Size - Top Ten (bar chart)
- File Size - Top Ten (table view)

Historical Summarized Availability workspace

The Historical Summarized Availability workspace shows the percentage of time that a managed resource was available during the number of months that you specify in the Time Span dialog. This workspace consists of the following two graphical views:

- Availability (average over months), which shows the percentage of time that managed resources were available, grouped by server
- Process Summary (average over months), which shows the percentage of time per system that each process was used by the server

Historical Summarized Availability Daily workspace

The Historical Summarized Availability Daily workspace shows availability information, a process summary, and a system summary for a managed server by day. This workspace consists of the following three graphical views:

- Availability (daily), which shows the percentage of the day that the server was available, summarized by day
- Process Summary (daily), which shows details such as memory and processor usage for processes that were running on the server, summarized by day
- System Summary (daily), which shows system information for the server, such as the operating system type, name, version, and manufacturer, summarized by day

Historical Summarized Availability Hourly workspace

The Historical Summarized Availability Hourly workspace shows availability information, a process summary, and a system summary for a managed server by hour. This workspace consists of the following three graphical views:

- Availability (hourly), which shows the percentage of the hour that the server was available, summarized by hour
- Process Summary (hourly), which shows details such as memory and processor usage for processes that were running on the server, summarized by hour
- System Summary (hourly), which shows system information for the server, such as the operating system type, name, version, and manufacturer, summarized by hour

Historical Summarized Availability Weekly workspace

The Historical Summarized Availability Weekly workspace shows availability information, a process summary, and a system summary for a managed server by week. This workspace consists of the following three graphical views:

- Availability (weekly), which shows the percentage of system time that the server was available, summarized by week

- Process Summary (weekly), which shows processes that kept the server busy, summarized by week
- System Summary (weekly), which shows system information such as the operating system type, name, version, and manufacturer, summarized by week

Historical Summarized Capacity workspace

The Historical Summarized Capacity workspace shows usage of system resources during the time span that you specify in the Time Span dialog. This workspace consists of the following five graphical views:

- Network Interface Activity (average over months), which shows network traffic for the server for all network interfaces on the system during the time span that you specify in the Time Span dialog
- Processor Utilization (average over months), which shows CPU usage, including idle CPU time, for all processors that are associated with the server during the specified time period
- Memory Utilization (average over months), which shows memory used, free memory, and swapped memory use during the specified time period
- Disk Utilization (maximum over months), which shows the maximum percentage of space used on the system's logical disks during the specified time period
- Disk Capacity (minimum over months), which shows information about the remaining number of days until the disk is full based on the current rate of disk usage, and the remaining number of days until the disk is full based on peak rate of disk usage, for all disks that are associated with the server

Historical Summarized Capacity Daily workspace

The Historical Summarized Capacity Daily workspace shows system usage summarized by day. This workspace consists of the following four graphical views:

- Network Interface Activity, which shows network traffic for the server, including packet collision rates, during the specified time period, summarized by day
- Processor Utilization, which shows CPU usage (including an idle, busy, or waiting CPU), for all processors that are associated with the server during the specified time period, summarized by day
- Memory Utilization, which shows memory used, free memory, and swapped memory use during the specified time period, summarized by day
- Disk Utilization, which shows percentage of space used or available on the system's logical disks during the specified time period, summarized by day

Historical Summarized Capacity Hourly workspace

The Historical Summarized Capacity Hourly workspace shows system resources used, summarized by hour. This workspace consists of the following four graphical views:

- Network Interface Activity, which shows network traffic, including collisions, packet transmittal and count transmittal for the server during the specified time period, summarized by hour
- Processor Utilization, which shows average CPU usage (idle, busy, and waiting), for all processors that are associated with the server during the specified time period, summarized by hour
- Memory Utilization, which shows memory used, free memory, and swapped memory use during the specified time period, summarized by hour

- Disk Utilization, which shows percentages of space used and available on all the system's logical disks during the specified time period, summarized by hour

Historical Summarized Capacity Weekly workspace

The Historical Summarized Capacity Weekly workspace shows system resources used, summarized by week. This workspace consists of the following five graphical views:

- Network Interface Activity, which shows network traffic for the server during the specified time period, summarized by week
- Processor Utilization, which shows CPU usage, especially idle CPU time, for all processors that are associated with the server during the specified time period, summarized by week
- Maximum Memory Utilization, which shows maximum memory used, free memory, and swapped memory during the specified time period, summarized by week
- Average Memory Utilization, which shows average memory that the server used during the specified time period, summarized by week
- Disk Utilization, which shows the maximum percentage of space used on all the system's logical disks during the specified time period, summarized by week

Historical Summarized Performance workspace

The Historical Summarized Performance workspace shows the average performance of system resources for the time span that you specify in the Time Span dialog. This workspace consists of the following five graphical views:

- Network Activity (maximum over months), which shows (in the sample period) percentages of errors and collisions in network traffic for all networks that are associated with the system during the time span that you specify in the Time Span dialog
- System Load (average over months), which shows the system workload during the specified time period
- Disk I/O Traffic (average over months), which shows the average percentage of time that the disk was busy during the specified time period
- Memory Page Faults (average over months), which shows the average rate of page in and page out for the system during the specified time period
- Processor Performance (average over months), which shows the average percentage of usage that users consumed and the average processor waiting time for the server during the specified time period

Historical Summarized Performance Daily workspace

The Historical Summarized Performance Daily workspace shows the performance of system resources, summarized by day. This workspace consists of the following five graphical views:

- Network Activity (daily), which shows the average network activity for a server, including transmittals, packet collisions, carrier losses, and so on, summarized by day
- System Load (daily), which shows the system workload during the specified time period, summarized by day
- Disk I/O Traffic (daily), which shows the average percentage of time that the disk was busy during the specified time period, summarized by day
- Memory Page Faults (daily), which shows the average rate of page in and page out for the system during the specified time period, summarized by day

- Processor Performance (daily), which shows the percentage of processor time that users consumed, as well as the waiting time that the CPU spent during the specified time period, summarized by day

Historical Summarized Performance Hourly workspace

The Historical Summarized Performance Hourly workspace shows the performance of system resources, summarized by hour. This workspace consists of the following five graphical views:

- Network Activity (hourly), which shows the network activity for a server, including transmittals, packet collisions, carrier losses, and so on, summarized by hour
- System Load (hourly), which shows the system workload during the specified time period, summarized by hour
- Disk I/O Traffic (hourly), which shows the average percentage of time that the disk was busy during the specified time period, summarized by hour
- Memory Page Faults (hourly), which shows the average rate of page in and page out for the system during the specified time period, summarized by hour
- Processor Performance (hourly), which shows the percentage of processor time that users consumed, as well as the waiting time that the CPU spent during the specified time period, summarized by hour

Historical Summarized Performance Weekly workspace

The Historical Summarized Performance Weekly workspace shows the performance of system resources, summarized by week. This workspace consists of the following five graphical views:

- Network Activity (weekly), which shows the network activity for a server, including errors and packet collisions, for all networks associated with the server, summarized by week
- System Load (weekly), which shows the system workload during the specified time period, summarized by week
- Memory Page Faults (weekly), which shows the average rate of page in and page out for the system during the specified time period, summarized by week
- Disk I/O Traffic (weekly), which shows the average percentage of time that the disk was busy during the specified time period, summarized by week
- Processor Performance (weekly), which shows the percentage of processor time that users consumed, as well as the waiting time that the CPU spent during the specified time period, summarized by week

Linux workspace

The Linux workspace reflects the health of the system. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- System CPU Usage (bar chart)
- Disk IO Transfers (bar chart)
- System Load Averages (bar chart)

Network workspace

The Network workspace reflects the health of the network components within your monitored systems. This workspace has a superseded version that displays queries

with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of four views. The views are:

- Network Errors (bar chart)
- Network Activity (bar chart)
- Network Devices (table view)
- IP Addresses (table view)

The Network Errors chart shows the number of input errors, output errors, and collisions for the sampling period. The Network Activity chart shows the number of packets received and transmitted per second. The Network Devices table reflects your network's performance based on its transmission, reception, and collision data. The IP Addresses table shows the IP addresses of the network interface names. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

NFS Statistics workspace

The NFS Statistics workspace is reached by right-clicking the Network navigator item in the Tivoli Enterprise Portal. The NFS Statistics workspace provides statistics on the operations involving the Network File System, such as the number and type of calls being made, and the percentages those types of calls make up in relation to total calls. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). The views are:

- Network Errors (bar chart)
- RPC Network Activity (bar chart)
- NFS Statistics (table view)

Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Process workspace

The Process workspace reflects the health of specific processes within your monitored systems. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- Process CPU Percent Usage (bar chart)
- Process + Child CPU Percent Usage (bar chart)
- Process Information Detail (table view)

The Process CPU Percent Usage chart displays the percent of CPU time spent in kernel mode and spent in user mode by process. The Process + Child CPU Percent Usage chart displays the cumulative percent of CPU time spent in kernel mode and spent in user mode. The Process Information Detail table lists in tabular form a wide range of process characteristics such as data set size, kernel scheduling priority, the number of pages of memory, and the number of page faults. Based on

the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Process User Information workspace

The Process User Information workspace is reached by right-clicking the Process navigator item in the Tivoli Enterprise Portal. The Process User Information workspace identifies process owners of your monitored Linux system and details their usage. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- Process CPU Percent Usage (bar chart)
- Process + Child CPU Percent Usage (bar chart)
- Process User Information (table view)

The Process CPU Percent Usage chart displays the percent of CPU time spent in kernel mode and spent in user mode by process. The Process + Child CPU Percent Usage chart displays the cumulative percent of CPU time spent in kernel mode and spent in user mode. The Process User Information table provides in tabular form the names of effective groups, file system groups, real groups, and saved groups for your monitored systems. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

RPC Statistics workspace

The RPC Statistics workspace is reached by right-clicking the Network navigator item in the Tivoli Enterprise Portal. The RPC (remote procedure call) workspace provides statistics on the number and type of calls being made to the server and clients, including statistics on the number of calls that are not valid or had to be retransmitted. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). The views are:

- Network Errors (bar chart)
- RPC Network Activity (bar chart)
- RPC Statistics (table view)

Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Sockets Information workspace

The Sockets Information workspace is reached by right-clicking the Network navigator item in the Tivoli Enterprise Portal. The Sockets Information workspace reflects the health of the socket connections within your monitored systems. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- Sockets Used by Protocol (bar chart)

- Network Activity (bar chart)
- Socket Services Information (table view)

The Sockets Used by Protocol chart shows a count of the sockets currently in use and the high water mark for each protocol during the sampling period. The Network Activity chart shows the number of packets received and transmitted per second. The Socket Services Information table provides a detailed perspective of each socket that you are monitoring. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Specific File Information workspace

The Specific File Information workspace can be accessed by right-clicking the link on either the File Information workspace or the All Files workspace. The Specific File Information workspace contains detailed information about a specific file or directory. You can access this information down through the lowest directory structure. This workspace is comprised of two views. The views are:

- File Information (table view)
- Take Action view

System Configuration workspace

The System Configuration workspace is reached by right-clicking the System Information workspace in the Tivoli Enterprise Portal. The System Configuration workspace displays information about CPU usage, the processor's configuration, and operating system level. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). It contains three views:

- CPU Usage (bar chart)
- Processor Configuration Information (table view)
- OS Version Information (table view)

System Information workspace

The System Information workspace reflects the health of your monitored systems by displaying data associated with system loads, context switching, and process creation. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of four views. The views are:

- CPU Usage (bar chart)
- Paging Rates (bar chart)
- System Load (bar chart)
- Virtual Memory Statistics (bar chart)
- System Statistics (table view)

The CPU Usage bar chart shows the percentage of idle CPU time, system CPU time, user CPU time, and user nice CPU time of the monitored processor. The System Load chart depicts the load on your monitored system's processor during the previous one, five, and fifteen minutes. The paging rates chart displays information about paging in and out as well as swapping in and out trends in seconds. The Virtual Memory Statistics chart depicts the current usage and

availability of a variety of memory categories (buffered, cached, shared, and swapped). The System Statistics table lists in tabular form the source data of these charts and gauge. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

The System Configuration workspace is reached by right-clicking the System Information navigator item in the Tivoli Enterprise Portal.

Users Workspace

The Users workspace identifies logged in users. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of three views. The views are:

- Process User Information (table view)
- Total User Logins (needle gauge)
- User Login Information (table view)

The Process User Information table provides in tabular form the names of effective groups, file system groups, real groups, and saved groups for your monitored systems. The Total User Logins gauge displays the number of users logged into the monitored system during the monitoring period. The User Login Information table lists users, their login time, and their idle time. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Virtual Memory Statistics workspace

The Virtual Memory Statistics workspace is reached by right-clicking the System Information navigator item in the Tivoli Enterprise Portal. The Virtual Memory Statistics workspace provides a snapshot of your monitored systems memory usage. This workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). This workspace is comprised of four views. The views are:

- Context Switches Percent Change (needle gauge)
- Context Switches Per Second (needle gauge)
- Virtual Memory Statistics (bar chart)
- Virtual Memory Information (table view)

The Context Switches Percent Change gauge reflects the percent change in the number of context switches per second. The Context Switches Per Second gauge shows the number of context switches per second. The Virtual Memory Statistics chart depicts the current usage and availability of a variety of memory categories (buffered, cached, shared, and swapped). The Virtual Memory Information table presents the Virtual Memory Usage information in tabular form. Based on the information that this workspace provides, you can recommend changes, set up situations, and verify that your recommended changes improve performance.

Virtual Memory Usage Trends workspace

The Virtual Memory Usage Trends workspace is reached by right-clicking the Capacity Usage Information navigator item in the Tivoli Enterprise Portal. This

workspace has a superseded version that displays queries with signed 32-bit maximum value (2,147,483,647) and a version with the same name (minus 'superseded') with queries that support values up to signed 64-bit max (9,223,372,036,854,775,807). The views are:

- Current Virtual Memory Usage (bar chart)
- Virtual Memory Averages (bar chart)
- Swap Space Usage Trends (table view)

The Current Virtual Memory Usage bar chart displays memory usage information. The Virtual Memory Averages bar chart displays virtual memory usage trend information. The Swap Space Usage Trends table provides several types of swap space information.

Chapter 4. Attributes reference

This chapter contains information about the following topics:

- Overview of attributes
- References for detailed information about attributes
- Descriptions of the attributes for each attribute group included in this monitoring agent
- Disk space requirements for historical data

About attributes

Attributes are the application properties being measured and reported by the Monitoring Agent for Linux OS, such as the amount of memory usage or the message ID. Some agents have fewer than 100 attributes, while others have over 1000.

Attributes are organized into groups according to their purpose. The attributes in a group can be used in the following two ways:

- Chart or table views

  Attributes are displayed in chart and table views. The chart and table views use queries to specify which attribute values to request from a monitoring agent. You use the Query editor to create a new query, modify an existing query, or apply filters and set styles to define the content and appearance of a view based on an existing query.
- Situations

  You use attributes to create situations that monitor the state of your operating system, database, or application. A situation describes a condition you want to test. When you start a situation, the Tivoli Enterprise Portal compares the values you have assigned to the situation attributes with the values collected by the Monitoring Agent for Linux OS and registers an *event* if the condition is met. You are alerted to events by indicator icons that appear in the Navigator.

Some of the attributes in this chapter are listed twice, with the second attribute having a "(Unicode)" designation after the attribute name. These Unicode attributes were created to provide access to globalized data.

More information about attributes

For more information about using attributes and attribute groups, see the *IBM Tivoli Monitoring User's Guide*.

For a list of the attributes groups, a list of the attributes in each attribute group, and descriptions of the attributes for this monitoring agent, refer to the Attribute groups and attributes section in this chapter.

Groups of attributes

Each attribute belongs to an attribute group. The attribute group includes attributes that are related. Each attribute item stores data for a particular property of an attribute group.

The following are the attribute groups for IBM Tivoli Monitoring: Linux OS Agent. The groups are collected in attribute tables that are designated in brackets [ ] after the group name.

- Agent Availability Management Status [KLZPASMGMT]
- Agent Active Runtime Status [KLZPASSTAT]
- Alerts Table [KLZPASALRT]
- All Users Group [LNXALLUSR]
- Configuration Information [KLZPASCAP]
- CPU [KLZCPU]
- CPU (superseded) [LNXCPU]
- CPU Averages [KLZCPUAVG]
- CPU Averages (superseded) [LNXCPUAVG]
- CPU Configuration [LNXCPUCON]
- Disk [KLZDISK]
- Disk (superseded) [LNXDISK]
- Disk IO [KLZDSKIO]
- Disk IO (superseded) [LNXDSKIO]
- Disk Usage Trends [KLZDU]
- Disk Usage Trends (superseded) [LNXDU]
- File Comparison Group [LNXFILCMP]
- File Information [LNXFILE]
- File Pattern Group [LNXFILPAT]
- Linux Group [LNXGROUP]
- I/O Ext [KLZIOEXT]
- I/O Ext (superseded) [LNXIOEXT]
- IP Address [LNXIPADDR]
- Linux Host Availability [LNXPING]
- Machine Information [LNXMACHIN]
- Network [KLZNET]
- Network (superseded) [LNXNET]
- NFS Statistics [KLZNFS]
- NFS Statistics (superseded) [LNXNFS]
- OS Configuration [LNXOSCON]
- Process [KLZPROC]
- Process (superseded) [LNXPROC]
- Process User Info [KLZPUSR]
- Process User Info (superseded) [LNXPUSR]
- RPC Statistics [KLZRPC]
- RPC Statistics (superseded) [LNXRPC]
- Sockets Detail [KLZSOCKD]
- Sockets Detail (superseded) [LNXSOCKD]
- Sockets Status [KLZSOCKS]
- Sockets Status (superseded) [LNXSOCKS]
- Swap Rate [KLZSWPRT]
- Swap Rate (superseded) [LNXSWPRT]

- System Statistics [KLZSYS]
- System Statistics (superseded) [LNXSYS]
- User Login [KLZLOGIN]
- User Login (superseded) [LNXLOGIN]
- VM Stats [KLZVM]
- VM Stats (superseded) [LNXVM]

IBM Tivoli Monitoring provides other attribute groups that are available to all monitoring agents, for example Universal Time and Local Time. The attributes in these common attribute groups are documented in the Tivoli Enterprise Portal Help.

Attribute groups and attributes for the Monitoring Agent for Linux OS

The following sections contain descriptions of these attribute groups, which are listed alphabetically. Each description contains a list of attributes in the attribute group.

IBM Tivoli Monitoring provides other attribute groups that are available to all monitoring agents, for example Universal Time and Local Time. The attributes in these common attribute groups are documented in the Tivoli Enterprise Portal Help.

Note: Some of the attributes have the enumerations, Value Exceeds Maximum and Value Exceeds Minimum. The Tivoli Enterprise Monitoring Server allows only signed integers, so the maximum is 9,223,372,036,854,775,807 and the minimum is -9,223,372,036,854,775,808. If the agent has a value bigger or smaller than these, it is capped with these enumerations.

Agent Availability Management Status Attributes

Use Agent Availability Management Status attributes to view the current management status of an agent relative to Agent Management Services.

Agent Management Status The watched agent management status. Valid values include the following: Unmanaged (0), Managed (1), Watchdog (2). A value of 'Managed' means that the agent is under the management of Agent Management Services. A value of 'Unmanaged' means it is known, but not under the management of Agent Management Services.

Agent Name The watched agent name.

Agent Type The watched agent type. The following are valid values: Unknown (0), ITM_Unix (1), Console (2), Windows_Service (3), Discover_ITM (4), Discover_Bin (5), Linux_Service (6), ITM_Windows (7).

Agent Version The VRM information for the agent.

Build Date The build date information for the agent. Superseded by the Build Number attribute.

Build Number The build number information for the agent.

Manager Type The enum defining the manager type. Valid values include the following: Unknown (0), Not_Managed (1), Agent_Management Services (2),

Watchdog (3), External (4). A value of 'Agent Management Services' means that Agent Management Services is responsible. A value of 'NotManaged' means that the agent is not under availability monitoring by any application. A value of 'Externally' means that some other application besides Agent Management Services is responsible for availability monitoring of the agent, for example Tivoli System Automation.

Operating System The operating system identification. The following are valid values: Win2000 (0), Win2003 (1), Win2008 (2), AIX (3), Linux (4), UNKNOWN (5), NA (-1).

Server Name The origin node of the collecting agent.

Service Name The service name.

Timestamp The date and time the Tivoli Enterprise Monitoring Server samples the data. Standard 16-character date/time format (CYYMMDDHHMMSSmmm), where:

C	Century (0 for 20th, 1 for 21st)
YY	Year
MM	Month
DD	Day
HH	Hour
MM	Minute
SS	Second
mmm	Millisecond

Use simple text strings as described above. For example, 1101009130500000 expresses October 9, 2010, 1:05:00 pm.

Agent Active Runtime Status Attributes

Use the Agents'Active Runtime Status attributes to view the current availability status of an agent: Running, Not present, Unknown, Stopped, Manually Stopped. You can view the frequency at which the agent's availability and runtime properties are queried and also the agent's Daily Restart Count.

Agent Availability Status The watched agent availability status. Valid values include the following: Unknown (0), Not_found (1), Stopped (2), Start_Pending (3), Running (4), Manually_Stopped (5), Stop_Pending (6), Not_Configured (7). For agents that have an Availability Status of 'Running', use the attribute group to see runtime properties of the agent such as its Process ID and Thread Count.

Agent Host Name The hostname of the agent.

Agent Name The watched agent name.

Agent Type The watched agent type. The following are valid values: Unknown (0), ITM_Unix (1), Console (2), Win_Service (3), Discover_ITM (4), Discover_Bin (5), Linux_Service (6), ITM_Windows (7).

Check Frequency The frequency to check status in seconds.

Command Line The command line.

Daily Restart Count The restarts within a period of a day.

Instance Name The instance name of the running IBM Tivoli Monitoring agent.

IP Address The IP address of the agent.

Last Health Check The last health check timestamp.

Number of Threads The thread count.

Operating System The operating system identification. The following are valid values: Unknown (0), Windows (1), Linux (2).

Page Faults Per Second The total page faults.

Parent Process ID The parent process ID.

Process ID The process ID.

Process Name The process name.

Process System CPU (Percent) The system CPU.

Process User CPU (Percent) The user CPU time.

Resident Size The process resident size.

Server Name The origin node of the collecting agent.

Timestamp The date and time the Tivoli Enterprise Monitoring Server samples the data. Standard 16-character date/time format (CYYMMDDHHMMSSmmm), where:

C	Century (0 for 20th, 1 for 21st)
YY	Year
MM	Month
DD	Day
HH	Hour
MM	Minute
SS	Second
mmm	Millisecond

Use simple text strings as described above. For example, 1101009130500000 expresses October 9, 2010, 1:05:00 pm.

Total Size (Pages) The total memory size in pages.

User Name The user name of running managed agent.

Alerts Table Attributes

Use the Alerts Table attributes to view exceptional Warning and Critical level events surfaced by Agent Management Services. These events have to do with the operation of Agent Management Services or conditions affecting its ability to manage agents. They include the following:

- Agent stopped abnormally.
- Agent restart failed.
- Agent exceeded restart tries.

- Agent not found.
- Agent exceeded policy defined memory threshold.
- Agent exceeded policy defined CPU threshold.
- Agent manual stop failed.
- Agent removed from system - CAP file removed.

Agent Name The watched agent name.

Agent Status The agent status. Valid values include the following: Unknown (0), Not_found (1), Stopped (2), Start_Pending (3), Running (4), Manually_Stopped (5), Stop_Pending (6), Not_Configured (7).

Agent Type The watched agent type. The following are valid values: Unknown (0), ITM_Unix (1), Console (2), Windows_Service (3), Discover_ITM (4), Discover_Bin (5), Linux_Service (6), ITM_Windows (7).

Alert Details The alert message details.

Alert Message The alert message. Valid values include the following: Availability_policy_removed (1), Managed_agent_removed_from_system (2), Unmanaged_agent_removed_from_system (3), Agent_abnormally_stopped (4), Agent_exceeded_restart_count (5), Agent_restart_failed (6), Agent_overutilizing_memory (7), Agent_overutilizing_CPU (8), Agent_manual_stop_failed (9).

Operating System The operating system identification. The following are valid values: Unknown (0), Windows (1), Linux (2).

Process ID The process ID.

Process Name The process name.

Server Name The origin node of the collecting agent.

Timestamp The date and time the Tivoli Enterprise Monitoring Server samples the data. Standard 16-character date/time format (CYYMMDDHHMMSSmmm), where:

C	Century (0 for 20th, 1 for 21st)
YY	Year
MM	Month
DD	Day
HH	Hour
MM	Minute
SS	Second
mmm	Millisecond

Use simple text strings as described above. For example, 1101009130500000 expresses October 9, 2010, 1:05:00 pm.

All Users Attributes

The All Users attributes refer to user characteristics such as name, user sessions, and user ID.

Duplicate User Name True if the user name is listed more than once in /etc/passwd. The valid values are False and True.

Name The full name of a user.

No Password True if no password is assigned to the user. The valid values are Unknown (-1), False (0), and True (1).

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

User ID The numeric ID the system assigned to a user. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

User ID (Superseded) The numeric ID the system assigned to a user. Valid values can include the value Value_Exceeds_Maximum=2147483647.

User Sessions The number of login sessions this user currently has established. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Configuration Information Attributes

Use Configuration Information attributes to monitor agent configuration like Memory Threshold and Operating System.

Agent Name The sub agent name.

Agent Path The fully qualified path to agent.

Agent Type The watched agent type. The following are valid values: Unknown (0), ITM_Unix (1), Console (2), Windows_Service (3), Discover_ITM (4), Discover_Bin (5), Linux_Service (6), ITM_Windows (7).

Check Frequency The frequency to check status in seconds.

Configuration Script The agent configuration script.

% CPU Threshold The maximum CPU allowed.

Dependencies The dependent agents.

Manager Type The enum defining the manager type. Valid values include the following: Unknown (0), Not_Managed (1), Agent_Management Services (2), Watchdog (3), External (4).

Maximum Daily Restarts The maximum number of restarts allowed. The clock begins at midnight.

Memory Threshold The maximum memory allowed.

Memory Unit The maximum memory allowed units. Valid values include the following: Bytes (0), KB (1), MB (2), GB (3).

Operating System The operating system identification. The following are valid values: Unknown (0), Windows (1), Linux (2).

Operating System Name The operating system name.

Operating System Version The operating system version.

PAS_ID The PAS sub agent ID.

Policy File Timestamp The date and time of CAP file.

Process Name The process name of the managed agent.

Server Name The origin node of the collecting agent.

Service Name The service name.

Startup Script The agent startup script.

Status Script The agent status script.

Stop Script The agent stop script.

Timestamp The date and time the Tivoli Enterprise Monitoring Server samples the data. Standard 16-character date/time format (CYYMMDDHHMMSSmmm), where:

C	Century (0 for 20th, 1 for 21st)
YY	Year
MM	Month
DD	Day
HH	Hour
MM	Minute

SS Second
mmm Millisecond

Use simple text strings as described above. For example, 1101009130500000 expresses October 9, 2010, 1:05:00 pm.

CPU Attributes

The CPU attributes refer to processor characteristics such as idle time, system CPU time, and user CPU time.

Busy CPU (Percent) The percentage of time the CPU was busy. Valid entry is an integer. Valid entry is an integer in the range 0 to 100.

CPU ID The processor ID. Valid entry is an integer in the range 0 to 999. Use this attribute to determine the processor ID. In a SMP system with more than one processor, the CPU report will show CPU ID as "aggregate" on the first row. This means the data row return aggregated CPU statistics. Valid values can include the value Aggregate=-1.

Idle CPU (Percent) Percent of idle CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine how efficiently the entire system or each processor of the SMP system is operating. The Idle CPU value must be low if the system load is heavy, and high if the system load is light. If the system load is heavy and the Idle CPU value is high, an I/O problem might exist. If the Idle CPU value is small, or zero, and the User percent is larger (greater than 30%), the system might be compute-bound or in a loop.

I/O Wait (Percent) The percentage of time the CPU was in a wait input/output state. Valid entry is an integer in the range of 0 to 100.

System CPU (Percent) Percent of system CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine the percent of system or per processor CPU time devoted to executing Linux system kernel code. System CPU time includes time spent executing system calls and performing administrative functions.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

User CPU (Percent) Percent of user CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine the percent of system or per processor CPU time devoted to user processes. User CPU time includes time spent executing both user program and library functions. It does not include CPU time spent executing system calls. The ratio between user and system CPU time varies, depending on the kinds of programs executing. If user CPU is extremely high and adversely affecting system performance, you might want to determine which user programs are preventing the CPU from functioning at its normal speed.

User Nice CPU (Percent) Percent of user nice CPU time during the sampling period. Valid entry is an integer in the range 0 to 100.

User to System CPU (Percent) Of the total CPU time, the percentage consumed by users. The range of possible values for this attribute is -10000 to + 10000.

CPU Attributes (superseded)

The CPU attributes refer to processor characteristics such as idle time, system CPU time, and user CPU time. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Busy CPU (Percent) The percentage of time the CPU was busy. Valid entry is an integer. Valid entry is an integer in the range 0 to 100. (Superseded.)

CPU ID The processor ID. Valid entry is an integer in the range 0 to 999. Use this attribute to determine the processor ID. In a SMP system with more than one processor, the CPU report will show CPU ID as “aggregate” on the first row. This means the data row return aggregated CPU statistics. Valid values can include the value Aggregate=-1. (Superseded.)

Idle CPU (Percent) Percent of idle CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine how efficiently the entire system or each processor of the SMP system is operating. The Idle CPU value must be low if the system load is heavy, and high if the system load is light. If the system load is heavy and the Idle CPU value is high, an I/O problem might exist. If the Idle CPU value is small, or zero, and the User percent is larger (greater than 30%), the system might be compute-bound or in a loop. (Superseded.)

I/O Wait (Percent) The percentage of time the CPU was in a wait input/output state. Valid entry is an integer in the range of 0 to 100. (Superseded.)

System CPU (Percent) Percent of system CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine the percent of system or per processor CPU time devoted to executing Linux system

kernel code. System CPU time includes time spent executing system calls and performing administrative functions. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

User CPU (Percent) Percent of user CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. Use this attribute to determine the percent of system or per processor CPU time devoted to user processes. User CPU time includes time spent executing both user program and library functions. It does not include CPU time spent executing system calls. The ratio between user and system CPU time varies, depending on the kinds of programs executing. If user CPU is extremely high and adversely affecting system performance, you might want to determine which user programs are preventing the CPU from functioning at its normal speed. (Superseded.)

User Nice CPU (Percent) Percent of user nice CPU time during the sampling period. Valid entry is an integer in the range 0 to 100. (Superseded.)

User to System CPU (Percent) Of the total CPU time, the percentage consumed by users. The range of possible values for this attribute is -10000 to + 10000. (Superseded.)

CPU Averages Attributes

The CPU Averages attributes refer to CPU usage, System CPU time, idle CPU time, user CPU time, and user nice CPU time characteristics.

Estimated Days Until CPU Upgrade The number of days until CPU Usage Moving average hits 100% Rate. Valid entry is an integer. Note: -1 indicates Not Available and -2 indicates Not Collected.

Idle CPU Moving Average (Percent) The moving average of the idle CPU time for the system, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

Idle CPU (Percent) The current average of the idle CPU time for the system, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

System CPU Current Average (Percent) The current average of the System CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

System CPU Moving Average (Percent) The moving average of the System CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total CPU Used Current Average (Percent) The current average of CPU usage, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

Total CPU Used Moving Average (Percent) The moving average of CPU usage, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

User CPU Current Average (Percent) The current average of the user CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

User CPU Moving Average (Percent) The moving average of the user CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

User Nice CPU Current Average (Percent) The current average of the user nice CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

User Nice CPU Moving Average (Percent) The moving average of the user nice CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly.

Wait CPU Moving Average (Percent) The moving current average of the wait CPU time, expressed as a percentage. Valid entry is an integer between 0 and 100. This average is calculated hourly.

Wait CPU (Percent) The current average of the wait CPU time, expressed as a percentage. Valid entry is an integer between 0 and 100. This average is calculated hourly.

CPU Averages Attributes (superseded)

The CPU Averages attributes refer to CPU usage, System CPU time, idle CPU time, user CPU time, and user nice CPU time characteristics. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Estimated Days Until CPU Upgrade The number of days until CPU Usage Moving average hits 100% Rate. Valid entry is an integer. Note: -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Idle CPU Moving Average (Percent) The moving average of the idle CPU time for the system, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

Idle CPU (Percent) The current average of the idle CPU time for the system, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

System CPU Current Average (Percent) The current average of the System CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

System CPU Moving Average (Percent) The moving average of the System CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ. (Superseded.)

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total CPU Used Current Average (Percent) The current average of CPU usage, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

Total CPU Used Moving Average (Percent) The moving average of CPU usage, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

User CPU Current Average (Percent) The current average of the user CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

User CPU Moving Average (Percent) The moving average of the user CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

User Nice CPU Current Average (Percent) The current average of the user nice CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

User Nice CPU Moving Average (Percent) The moving average of the user nice CPU time, expressed as a percent. Valid entries: integers between 0 and 100, such as 85 for 85%. This average is calculated hourly. (Superseded.)

Wait CPU Moving Average (Percent) The moving current average of the wait CPU time, expressed as a percentage. Valid entry is an integer between 0 and 100. This average is calculated hourly. (Superseded.)

Wait CPU (Percent) The current average of the wait CPU time, expressed as a percentage. Valid entry is an integer between 0 and 100. This average is calculated hourly. (Superseded.)

CPU Configuration Attributes

The CPU Configuration attributes refer to configuration characteristics such as CPU ID, CPU Family, and Clock Speed.

Model Name The process model name.

Processor Cache Size (KB) The processor cache size (Kb). Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Processor Clock Speed (MHz) The processor clock speed (MHz). Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Processor Family Number The process family number. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Processor ID The processor ID.

Processor Model Number The process model number. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Processor Vendor ID The Processor Vendor ID.

Disk Attributes

The Disk attributes refer to disk characteristics such as inode size, inodes used, mount point, and space available.

Disk Free (MB) The amount of free space on a disk, expressed in megabytes. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Disk Free Percent The amount of free space on a disk, expressed as a percentage. Note: the value -1 indicates Not Available and the value -2 indicates Not Collected.

Disk Name The name of the physical disk partition where the filesystem is mounted. This is the physical location of the disk. Valid entry is an alphanumeric text string, with a maximum length of 32 characters.

Disk Used (MB) The amount of used space on a disk, expressed in megabytes. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Disk Used Percent The amount of used space on a disk, expressed as a percentage. Note: the value -1 indicates Not Available and the value -2 indicates Not Collected.

File System Type The file system type, such as hsfs, nfs, tmpfs, and ufs. Valid entries are up to eight letters or numbers.

Inodes Free The number of inodes currently available on your filesystem. Use this attribute to avoid a pending crisis. Corrective action might include freeing up unneeded space or deleting temporary files. If the value for Inodes Free is less than 100, this is a critical condition. Notify your system administrator immediately. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Inodes Free Percent The number of inodes currently available on your filesystem, expressed as a percentage. Note: the value -1 indicates Not Available and the value -2 indicates Not Collected.

Inodes Used The number of inodes currently allocated to files on the filesystem. This value equals the Total Inodes value minus the Inodes Free value. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Inodes Used Percent The percent of inodes currently allocated to files, calculated by dividing the Inodes Used value by the Total Inodes value. Valid entries: integers between 0 and 100, such as 85 for 85%. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Mount Point The path name of the directory to which a filesystem is mounted. This is the virtual name for the directory. Valid entries are up to 256 letters or numbers representing a directory path.

Size (MB) The total size of a filesystem, expressed in megabytes. For example, 1000 represents one gigabyte. Valid entry is an integer of up to 99999999. Note: the

value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Inodes The number of inodes allocated on a filesystem. For example, a value of 163817 indicates that the number of inodes allocated is 163,817. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Use this attribute when a filesystem needs additional or fewer inodes assigned to it. Viewing the current number of inodes assigned helps you determine the number of inodes you need to add or subtract to optimize performance in your system.

Disk Attributes (superseded)

The Disk attributes refer to disk characteristics such as inode size, inodes used, mount point, and space available. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Disk Mount Point The path name of the directory to which a filesystem is mounted. This is the virtual name for the directory. Valid entries are up to 32 letters or numbers representing a directory path. (Superseded.)

Disk Name The name of the physical disk partition where the filesystem is mounted. This is the physical location of the disk. Valid entry is an alphanumeric text string, with a maximum length of 32 characters. (Superseded.)

File System Type The file system type, such as hsfs, nfs, tmpfs, and ufs. Valid entries are up to eight letters or numbers. (Superseded.)

Inodes Available Percent The percent of inodes currently available. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Inodes Free The number of inodes currently available on your filesystem. Use this attribute to avoid a pending crisis. Corrective action might include freeing up unneeded space or deleting temporary files. If the value for Inodes Free is less than 100, this is a critical condition. Notify your system administrator immediately. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Inodes Used The number of inodes currently allocated to files on the filesystem. This value equals the Total Inodes value minus the Inodes Free value. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Inodes Used Percent The percent of inodes currently allocated to files, calculated by dividing the Inodes Used value by the Total Inodes value. Valid entries: integers between 0 and 100, such as 85 for 85%. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Mount Point (Unicode) The path name of the directory to which a filesystem is mounted. (Superseded.)

Size (MB) The total size of a filesystem, expressed in megabytes. For example, 1000 represents one gigabyte. Valid entry is an integer of up to 99999999. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Space Available (MB) The amount of unused space currently available to non-superusers on a filesystem, expressed in megabytes. For example, 40000 represents 40 megabytes. Valid entry is an integer of up to 99999999. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

This disk space does not include any space which is reserved for superuser. A low value in this column, relative to the disk size, alerts you to critical disk space conditions.

If this value is low for one or more filesystems, relative to the disk size, you might need to evaluate reconfiguring the filesystem to distribute the files more evenly across disks.

Space Available Percent The percent of space available. Valid entry is an integer between 0 and 100, such as 10 for 10%. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Space Used (MB) The amount of disk space currently in use on a filesystem, expressed in megabytes. For example, 5000 represents five gigabytes. Valid entry is an integer of up to 99999999. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Space Used Percent The space currently used on the file system, expressed as a percent of the sum of used and available space. The Space Used Percent reflects the percent of disk space which is available to non-superusers. A high value in this column alerts you to critical disk space conditions. Valid entries: integers between 0 and 100, such as 80 for 80%. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Inodes The number of inodes allocated on a filesystem. For example, a value of 163817 indicates that the number of inodes allocated is 163,817. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Use this attribute when a filesystem needs additional or fewer inodes assigned to it. Viewing the current number of inodes assigned helps you determine the number of inodes you need to add or subtract to optimize performance in your system.

Disk IO Attributes

The Disk IO attributes refer to disk input/output characteristics, including transfer rates, block read rates, and block write rates.

Note: These attributes are only available for systems with a 2.4 (or higher) kernel.

Blocks Reads Per Second Indicates the amount of data read from the drive expressed in a number of blocks per second. A block is of indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Blocks Written Per Second Indicates the amount of data written to the drive expressed in a number of blocks per second. A block is of indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Blocks Read The total number of blocks read. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Blocks Written The total number of blocks written. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Device Major Number Major number of the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Device Minor Number Distinctive minor number for device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Device Name Name of the device as appears under the /dev directory.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Transfers Per Second Indicates the number of transfers per second that were issued to the device. A transfer is an I/O request to the device. Multiple logical requests can be combined into a single I/O request to the device. A transfer is of

indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Disk IO Attributes (superseded)

The Disk IO attributes refer to disk input/output characteristics, including transfer rates, block read rates, and block write rates. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Note: These attributes are only available for systems with a 2.4 (or higher) kernel.

Block Reads Per Second Indicates the amount of data read from the drive expressed in a number of blocks per second. A block is of indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Blocks Written Per Second Indicates the amount of data written to the drive expressed in a number of blocks per second. A block is of indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Blocks Read The total number of blocks read. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Blocks Written The total number of blocks written. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Device Major Number Major number of the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Device Minor Number Distinctive minor number for device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Device Name Name of the device as appears under the /dev directory. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Transfers Per Second Indicates the number of transfers per second that were issued to the device. A transfer is an I/O request to the device. Multiple logical requests can be combined into a single I/O request to the device. A transfer is of indeterminate size. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Disk Usage Trends Attributes

The Disk Usage Trends attributes refer to disk usage characteristics, such as high water / low water usage rates and days until the disk is full.

Disk Name The name of the physical disk partition where the filesystem is mounted. This is the physical location of the disk. Valid entry is an alphanumeric text string, with a maximum length of 32 characters.

Days Until Full Disk Current Rate The number of days until the disk is full based on the current rate of disk usage. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Days Until Full Disk Moving Avg The number of days until the disk is full based on the moving average rate of disk usage. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Days Until Full Disk Low Water Mark The number of days until the disk is full based on the disk usage rate that represents the low water mark. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Days Until Full Disk Peak Rate Days until full disk based on the Peak Rate. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Disk Usage Moving Avg (Bytes/Hr) The bytes/hour of disk usage averaged over all previous samples. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Disk Usage Rate (Bytes/Hr) The bytes/hour of disk usage over the last hour. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

High Water Mark Disk Usage Rate (Bytes/Hr) The bytes/hour rate that represents the highwater mark of disk usage. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

High Water Mark Time Stamp The date and time that the disk usage reaches a highwater mark. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Space Free (MB) The amount of unused space currently available to non-superusers on a filesystem, expressed in megabytes. For example, 40,000 represents 40 megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

This disk space does not include any space which is reserved for superuser. A low value in this column, relative to the disk size, alerts you to critical disk space conditions.

If this value is low for one or more filesystems, relative to the disk size, you might need to evaluate reconfiguring the filesystem to distribute the files more evenly across disks.

Space Used (MB) The amount of disk space currently in use on a filesystem, expressed in megabytes. Valid entries For example, 5000 represents five gigabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Disk Usage Trends Attributes (superseded)

The Disk Usage Trends attributes refer to disk usage characteristics, such as high water / low water usage rates and days until the disk is full. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Disk Name The name of the physical disk partition where the filesystem is mounted. This is the physical location of the disk. Valid entry is an alphanumeric text string, with a maximum length of 32 characters. (Superseded.)

Days Until Full Disk Current Rate The number of days until the disk is full based on the current rate of disk usage. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Days Until Full Disk Moving Avg The number of days until the disk is full based on the moving average rate of disk usage. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Days Until Full Disk Low Water Mark The number of days until the disk is full based on the disk usage rate that represents the low water mark. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Days Until Full Disk Peak Rate Days until full disk based on the Peak Rate. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Disk Usage Moving Avg (Bytes/Hr) The bytes/hour of disk usage averaged over all previous samples. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Disk Usage Rate (Bytes/Hr) The bytes/hour of disk usage over the last hour. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

High Water Mark Disk Usage Rate (Bytes/Hr) The bytes/hour rate that represents the highwater mark of disk usage. Valid entry is an integer. Valid values can include the value Value_Exceeds_Minimum=-2147483648. (Superseded.)

High Water Mark Time Stamp The date and time that the disk usage reaches a highwater mark. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Space Available (MB) The amount of unused space currently available to non-superusers on a filesystem, expressed in megabytes. For example, 40,000 represents 40 megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

This disk space does not include any space which is reserved for superuser. A low value in this column, relative to the disk size, alerts you to critical disk space conditions.

If this value is low for one or more filesystems, relative to the disk size, you might need to evaluate reconfiguring the filesystem to distribute the files more evenly across disks.

Space Used (MB) The amount of disk space currently in use on a filesystem, expressed in megabytes. Valid entries For example, 5000 represents five gigabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

File Comparison Group Attributes

File Comparison Group Attributes refer to File Comparison Group characteristics. This attribute group is not available for historical data collection.

File Compare Option The File compare option is used to specify which type of comparison is used. The valid values include: Plain (1), Ignore_Whitespace (2), Ignore_Case (3), Ignore_Case_Whitespace (4), and Binary (5). The default is Plain.

File Compare Result The result of the file comparison between File_Name_1 and File_Name_2. Valid values include Same (0) and Different (1). Note: -1 indicates Not_Available and -2 indicates Not_Collected.

File Name 1 Fully-qualified file name of one of the files to be compared. This attribute is required.

File Name 2 Fully-qualified file name of the other of the files to be compared. This attribute is required.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

File Information Attributes

The File Information attributes refer to file information characteristics. This attribute group is not available for historical data collection.

Access The access rights of the file expressed as 4-digit octal number.

Attribute Last Change Time The date and time of the last file attributes change.

Checksum Checksum or hash string based on hashing algorithm. The default algorithm is CRC32.

Checksum Algorithm Only used in situations in conjunction with the Checksum attribute to specify the algorithm to be used to calculate the hash string. Note: -1 indicates Not_Applicable. Other possible values are CRC32 (0), MD5 (1), and SHA1 (2), Not_Available (-1). The default is CRC32.

File The name of file or directory. If the file is a symbolic link, the link name is shown in Link_Name attribute.

File Content Changed A numeric indicator that the content of a file has changed. It is equivalent to noting a change in checksum between two samples. Valid values include No (0), Yes (1), and Not Available (-1).

File Mode Mode is the string representation of the access rights of the file. This is related to the Access attribute. The access attribute is the octal representation of the access rights of the file. The mode of a file would be rwxr-xr-x if the access was 755.

Group The logical group to which the file belongs.

Last Accessed Time The date and time of the last file access.

Last Changed Time The date and time of the last change to a file.

Link Name The name of the file for which this file is a symbolic link. If this field is blank, the file is not a link.

Links The number of links to a file.

Owner The name of the file owner.

Path The full path containing a particular file or directory.

Size (MB) The size, in MB, of the file. This attribute displays as a floating point with a scale of 3. For example 55.255. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Size (MB) (Superseded) The size, in MB, of the file. This attribute displays as a floating point with a scale of 3. For example 55.255.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Type The type of file. Possible values are:

- Dir (= directory)
- DirLink (=directory link)
- File (= file)
- FileLink (=file link)
- Sock (= socket)
- Link (= link)
- Spec (= special file)
- Unknown (=unknown)

File Pattern Group Attributes

The File Pattern Group attributes refer to file pattern group characteristics. This attribute group is not available for historical data collection.

File Name Fully qualified file name which will be searched for lines matching a pattern.

Match Count The number of matches for the specified pattern in the specified file. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Match Option Options that affect how the search is performed. The valid values include: Normal (1), Ignore_Case (2), Inverse_Search (3), and Match_Whole_Words (4).

Match Pattern The grep regular expression used to search for matching lines in File Name.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Linux Group Attributes

The Linux Group attributes refer to group characteristics.

Duplicate Group Name True if the group name is listed more than once in /etc/group. The valid values include False (0) and True (1).

Group ID The ID of this group. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Group ID (Superseded) The ID of this group. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Group Name The name of the group.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

I/O Ext Attributes

The I/O Ext attributes refer to a wide variety of disk input/output characteristics, including read request rates, write request rates, and service time measures.

Note: These attributes are only available for systems with a 2.4 (or higher) kernel.

Average Request Queue Length The average queue length of the requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Average Request Size (Sectors) The average size (in sectors) of the requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Average Service time (ms) The average service time (in milliseconds) for I/O requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Average Wait Time (ms) The average time (in milliseconds) for I/O requests issued to the device to be served. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Bytes Transferred Per Second The number of bytes transferred per second. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Device Name Name of the device as appears under the /dev directory. Valid entry is an alphanumeric text string, with a maximum length of 64 characters.

Disk Read Percent The percentage of time spent in read operations.

Disk Write Percent The percentage of time spent in write operations.

Percent CPU Time Used Percentage of CPU time during which I/O requests were issued to the device. Saturation occurs at 100%.

Read Bytes Per Second The number of bytes read from the device per second. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Read Requests Per Second The number of read requests that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Read Requests Merged Per Second The number of read requests merged, per second, that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Read Sectors Per Second The number of sectors read, per second, from the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Write Bytes Per Second The number of bytes written to the device per second. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Write Requests Per Second The number of write requests that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Write Requests Merged Per Second The number of write requests merged that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Write Sectors Per Second The number of sectors written to the device, per second. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

I/O Ext Attributes (superseded)

The I/O Ext attributes refer to a wide variety of disk input/output characteristics, including read request rates, write request rates, and service time measures. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Note: These attributes are only available for systems with a 2.4 (or higher) kernel.

Average Request Queue Length The average queue length of the requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Average Request Size (Sectors) The average size (in sectors) of the requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Average Service time (ms) The average service time (in milliseconds) for I/O requests that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Average Wait Time (ms) The average time (in milliseconds) for I/O requests issued to the device to be served. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Bytes Transferred Per Second The number of bytes transferred per second. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Device Name Name of the device as appears under the /dev directory. Valid entry is an alphanumeric text string, with a maximum length of 64 characters. (Superseded.)

Disk Read Percent The percentage of time spent in read operations. (Superseded.)

Disk Write Percent The percentage of time spent in write operations. (Superseded.)

Percent CPU Time Used Percentage of CPU time during which I/O requests were issued to the device. Saturation occurs at 100%. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Read Bytes Per Second The number of bytes read from the device per second. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Read Requests Per Second The number of read requests that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Read Requests Merged Per Second The number of read requests merged, per second, that were issued to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Read Sectors Per Second The number of sectors read, per second, from the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Write Bytes Per Second The number of bytes written to the device per second. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Write Requests Per Second The number of write requests that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Write Requests Merged Per Second The number of write requests merged that were issued, per second, to the device. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Write Sectors Per Second The number of sectors written to the device, per second. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

IP Address Attributes

The IP Address attributes refer to network characteristics, including IP address and network interface name.

DNS Name The Domain Name Server (DNS) entry associated with the IP network address. Valid entry is an alphanumeric text string, with a maximum length of 384 characters. Note that the value No_DNS_Entry indicates NO_DNS_ENTRY.

IP Address An IP address associated with the network interface. Valid entry is an alphanumeric text string, with a maximum length of 46 characters.

IP Version An indicator as to whether the IP address is version 4 or version 6. Valid values include the following:

- IPv4=4
- IPv6=6

Network Interface Name The name of the network interface. Valid entry is an alphanumeric text string, with a maximum length of 32 characters.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Linux Host Availability Attributes

The Linux Host Availability attributes refer to Linux host availability characteristics. The attributes in this group can only be used in a situation. Historical information is available for the Host Availability table for users interested in trending server response times. However, to enable history collection for this attribute group, a list of monitored (pinged) servers must be specified. The list is specified through an environment variable - "KLZ_PINGHOSTLIST" in the lz.ini file in the IBM Tivoli Monitoring config directory. For example:

```
KLZ_PINGHOSTLIST='/opt/ibm/itm/config/klzpinghosts'

sample content of klzpinghosts:
```

```
#
# hosts pinged for availability from this agent
#
server1.domain.com
server2
server4
```

Host Status Result of the ″ping″ operation. Valid values include: Successful (1), Unsuccessful (0), and Error (-1).

Server Response Time Ping operation response time in milliseconds. Note: -1000 indicates Not Available.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Target Host The hostname or IP Address of the target of the ping operation. Valid entry is an alphanumeric text string, with a maximum length of 128 characters.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Machine Information attributes

The Machine Information attribute group contains various items required by other Tivoli products. They include system hardware information.

Note: This Monitoring Agent for Linux gathers the following attributes in this group using the command /usr/sbin/dmidecode:

- BIOS Version
- BIOS Release
- Hardware Brand
- Hardware Model

- Machine Serial Number

  The Monitoring Agent for Linux must be running as root in order to execute this command. If not, ″Unknown″ is returned for the dmidecode metrics. Further, this program is not available for zLinux or p-series systems. Hardware Brand will report as "IBM." Hardware Model will report as "zSeries," and the remaining metrics will report as "Unknown." Further information on dmidecode is available at the following website:

  `http://www.nongnu.org/dmidecode`

BIOS Release The BIOS vendor release date. Note: the value unknown = UNKNOWN.

BIOS Version The BIOS vendor version. Note: the value unknown = UNKNOWN.

Hardware Brand The brand of hardware on which the agent is running. Note: the value unknown = UNKNOWN.

Hardware Model The specific hardware model underlying the monitored operating system. Note: the value unknown = UNKNOWN.

Machine Serial Number The serial number of the machine. Note: the value unknown = UNKNOWN.

Number of Processors Configured The number of processors configured for this machine. This number excludes secondary processor contexts, but might include virtual processors in some virtual environments. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Number of Processors Online The number of processors online the machine. This number excludes secondary processor contexts, but might include virtual processors in some virtual environments. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Server Host Name The hostname for the machine. Note: the value unknown = UNKNOWN.

System Board UUID The Universally Unique Identifier burned in to the system board.

System Name The managed system name. The form should be *hostname*:*agent_code*. Note: the value unknown = UNKNOWN.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system.

Network Attributes

The Network attributes refer to network characteristics such as received count, sent count, network interface name, and interface status.

Bytes Received Per Second The number of bytes received per second by the interface. Valid entry is an integer in the range 0 to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Bytes Transmitted Per Second The number of bytes transmitted per second by the interface. Valid entry is an integer in the range 0 to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Carrier Losses The number of carrier losses that occurred in the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Collisions (Percent) Of the total number of packets transmitted in this sample period, the percentage involved in a collision. Valid entry is an integer.

Collisions Per Minute The number of times a packet collided with another packet per minute. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Device Type The device type. Valid Values include: NETROM, ETHER, EETHER, AX25, PRONET, CHAOS, IEEE802_TR, ARCNET, APPLETLK, DLCI, ATM, METRICOM, IEEE1394, SLIP, CSLIP, SLIP6, CSLIP6, RSRVD, ADAPT, ROSE, X25, HWX25, PPP, HDLC, LAPB, DDCMP, RAWHDLC, TUNNEL, TUNNEL6, FRAD, SKIP, LOOPBACK, LOCALTLK, FDDI, BIF, SIT, IPDDP, IPGRE, PIMREG, HIPPI, ASH, ECONET, IRDA, FCPP, FCAL, FCPL, FCFABRIC, IEEE802, IEEE80211, UNKNOWN.

Errors (Percent) Of the total number of packets received and transmitted, the percentage that were in error during this sample period. Valid entry is an integer.

This information can help you determine the data transfer capabilities of various network interfaces, and alleviate bottlenecks by re-routing traffic from devices that appear to be overloaded, to other network interfaces that might be able to handle additional data traffic.

Input Error (Percent) The number of input packet errors as a percentage of the total number of packets received in this sample.

Input Errors The number of packets with errors received on the interface. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Input Errors Per Minute The number of packets with errors received per minute by the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Input FIFO Buffer Overruns The number of input FIFO buffer overruns that occurred during the sampling period. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Input Packets Dropped The number of input packets dropped by the device driver. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Example: www.company.com indicates that the DNS will resolve the name www.company.com to mean the IP address for the interface.

IPv4 Address The Internet Protocol (IP) address of the network interface. A gateway system might have more than one interface, each with a separate address. Valid entries: Internet protocol addresses in the form a.b.c.d. where a, b, c, and d are integers in the range 0 to 255.

Example: 197.128.55.55 indicates the network interface uses the IP address 197.128.55.55.

Interface Status This attribute indicates if a network interface is currently available. Valid entries for each Network interface:

UP	Indicates the interface is in service
DOWN	Indicates the interface is not in service
UP_NOT_RUNNING	Indicates the interface is in service but not running
UNKNOWN	Indicates the interface is in unknown

These values are case-sensitive.

Example:**UP** means an interface is in service.

MAC Address The MAC address of the Network Interface Card. NOT_AVAILABLE is a valid value. It is typically 6 bytes, but can be up to 14. The value is formatted with a colon between each byte.

Maximum Transmission Unit The maximum packet size (in bytes) for the specified network interface. This is a fixed value. Valid entry is an integer in the range 0 to 99999999. Use this attribute to determine the minimum, maximum or average packet size used by a network interface. This information can help you determine the size used by a network interface.

Network Interface Name Identifies the network interface adapter. Valid entries are simple text string, alphanumeric comprised of "Interface Name, Unit Number" where:

- The name is a two-character representation of the adapter, based on the hardware, operating system, and installation procedure.
- The unit represents the physical adapter number installed in the system with a typical range 0 to 7.

Output Errors The number of packet transmission errors by the network interface. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Output Error (Percent) The total number of output errors as a percentage of the total number of packets transmitted in this sample.

Output Errors Per Minute The number of packet transmission errors per minute during the monitoring interval. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Output FIFO Buffer Overruns The number of output FIFO buffer overruns that occurred during the sampling period. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Output Packets Dropped The number of output packets dropped by the device driver. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Packet Framing Errors The number of packet framing errors that occurred in the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Packets Received The number of packets received by the interface during the sampling period. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Packets Received Per Second The number of packets received per second by the interface. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Packets Transmitted The number of packets transmitted by the interface during the sampling period. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Packets Transmitted Per Second The number of packets transmitted per second by the interface. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Received Count (KB) The number of kilobytes received since the network interface was configured. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Example: If a low number of packets are being received, data traffic might need to be re-routed.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Collisions The number of times during the sampling period that a packet transmitted by the network interface collided with another packet. This occurs when another interface on the same local network transmits a packet at nearly the same time. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Use this attribute to determine if a network interface has an unacceptable number of packet collisions. Packet collisions cause the interface to retransmit the packet. With this increased traffic, the likelihood of future collisions increases. This can result in a steady increase of network traffic to critical levels.

Transmitted Count (KB) The number of kilobytes transmitted by an interface since boot time. Valid entry is an integer in the range zero to 9223372036854775807. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Example: A high value might indicate an overloaded interface. A low value might indicate a device that is not being used much, which can carry an additional load, if required.

Network Attributes (superseded)

The Network attributes refer to network characteristics such as received count, sent count, network interface name, and interface status. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Bytes Received Per Second The number of bytes received per second by the interface. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Bytes Transmitted Per Second The number of bytes transmitted per second by the interface. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Minimum=-2147483648 and the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Carrier Losses The number of carrier losses that occurred in the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Collisions (Percent) Of the total number of packets transmitted in this sample period, the percentage involved in a collision. Valid entry is an integer. (Superseded.)

Collisions Per Minute The number of times a packet collided with another packet per minute. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Device Type The device type. Valid Values include: NETROM, ETHER, EETHER, AX25, PRONET, CHAOS, IEEE802_TR, ARCNET, APPLETLK, DLCI, ATM, METRICOM, IEEE1394, SLIP, CSLIP, SLIP6, CSLIP6, RSRVD, ADAPT, ROSE, X25, HWX25, PPP, HDLC, LAPB, DDCMP, RAWHDLC, TUNNEL, TUNNEL6, FRAD, SKIP, LOOPBACK, LOCALTLK, FDDI, BIF, SIT, IPDDP, IPGRE, PIMREG, HIPPI, ASH, ECONET, IRDA, FCPP, FCAL, FCPL, FCFABRIC, IEEE802, IEEE80211, UNKNOWN. (Superseded.)

Errors (Percent) Of the total number of packets received and transmitted, the percentage that were in error during this sample period. Valid entry is an integer. (Superseded.)

This information can help you determine the data transfer capabilities of various network interfaces, and alleviate bottlenecks by re-routing traffic from devices that appear to be overloaded, to other network interfaces that might be able to handle additional data traffic.

Input Error (Percent) The number of input packet errors as a percentage of the total number of packets received in this sample. (Superseded.)

Input Errors The number of packets with errors received on the interface. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Input Errors Per Minute The number of packets with errors received per minute by the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Input FIFO Buffer Overruns The number of input FIFO buffer overruns that occurred during the sampling period. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Input Packets Dropped The number of input packets dropped by the device driver. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Example: www.company.com indicates that the DNS will resolve the name www.company.com to mean the IP address for the interface.

IPv4 Address The Internet Protocol (IP) address of the network interface. A gateway system might have more than one interface, each with a separate address. Valid entries: Internet protocol addresses in the form a.b.c.d. where a, b, c, and d are integers in the range 0 to 255. (Superseded.)

Example: 197.128.55.55 indicates the network interface uses the IP address 197.128.55.55.

Interface Status This attribute indicates if a network interface is currently available. (Superseded.) Valid entries for each Network interface:

UP	Indicates the interface is in service
DOWN	Indicates the interface is not in service
UP_NOT_RUNNING	Indicates the interface is in service but not running
UNKNOWN	Indicates the interface is in unknown

These values are case-sensitive.

Example:**UP** means an interface is in service. (Superseded.)

MAC Address The MAC address of the Network Interface Card. NOT_AVAILABLE is a valid value. It is typically 6 bytes, but can be up to 14. The value is formatted with a colon between each byte. (Superseded.)

Maximum Transmission Unit The maximum packet size (in bytes) for the specified network interface. This is a fixed value. Valid entry is an integer in the range 0 to 99999999. Valid values can include the value Value_Exceeds_Maximum=2147483647. Use this attribute to determine the minimum, maximum or average packet size used by a network interface. This information can help you determine the size used by a network interface. (Superseded.)

Network Interface Name Identifies the network interface adapter. (Superseded.) Valid entries are simple text string, alphanumeric comprised of "Interface Name, Unit Number" where:

- The name is a two-character representation of the adapter, based on the hardware, operating system, and installation procedure.
- The unit represents the physical adapter number installed in the system with a typical range 0 to 7.

Output Errors The number of packet transmission errors by the network interface. Valid entries are integers in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Output Error (Percent) The total number of output errors as a percentage of the total number of packets transmitted in this sample. (Superseded.)

Output Errors Per Minute The number of packet transmission errors per minute during the monitoring interval. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Output FIFO Buffer Overruns The number of output FIFO buffer overruns that occurred during the sampling period. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Output Packets Dropped The number of output packets dropped by the device driver. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Packet Framing Errors The number of packet framing errors that occurred in the interface. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Packets Received The number of packets received by the interface during the sampling period. Valid entry is an integer in the range 0 to 99999999. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Packets Received Per Second The number of packets received per second by the interface. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Packets Transmitted The number of packets transmitted by the interface during the sampling period. Valid entry is an integer in the range 0 to 99999999. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Packets Transmitted Per Second The number of packets transmitted per second by the interface. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Received Count (KB) The number of kilobytes received since the network interface was configured. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Example: If a low number of packets are being received, data traffic might need to be re-routed.

System Name The managed system name. The form should be *hostname*:*agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Collisions The number of times during the sampling period that a packet transmitted by the network interface collided with another packet. This occurs

when another interface on the same local network transmits a packet at nearly the same time. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Use this attribute to determine if a network interface has an unacceptable number of packet collisions. Packet collisions cause the interface to retransmit the packet. With this increased traffic, the likelihood of future collisions increases. This can result in a steady increase of network traffic to critical levels.

Transmitted Count (KB) The number of kilobytes transmitted by an interface since boot time. Valid entry is an integer in the range 0 to 2147483647. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Example: A high value might indicate an overloaded interface. A low value might indicate a device that is not being used much, which can carry an additional load, if required.

NFS Statistics Attributes

Use NFS Statistics to monitor characteristics of Network File System (NFS) such as the number of calls, lookups, and operations. This agent currently reports only on NFS version 2 and 3 statistics.

Access Calls The number of access calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Access Calls Percent Of the total number of calls made to the NFS server, the percentage that were access calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Commit Calls The number of file commit calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Commit Calls Percent Of the total number of calls made to the NFS server, the percentage that were file commit calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

File Creates The number of file create calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

File Creates Percent Of the total number of calls made to the NFS server, the percentage that contained file creation operations. Valid entry is an integer in the range of 0 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

File System Info Calls The number of file system information calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

File System Info Calls Percent Of the total number of calls made to the NFS server, the percentage that were calls to obtain information about the file system. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

File System Statistics Calls The number of calls made to the NFS server which requested statistics of the file system. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

File System Statistics Calls Percent Of the total number of calls made to the NFS server, the percentage that involved a request for file system statistics. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Get Attribute Calls The number of calls made to the NFS server which contained a get attribute (getattr) operation. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Get Attribute Calls Percent Of the total number of calls made to the NFS server, the percentage that contained get attribute (getattr) operations. Valid entry is an integer in the range of 0 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Link Calls The total number of link calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Link Calls Percent Of the total number of calls made to the NFS server, the percentage that were link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Location The location of the origin of the call in the Network File System. Valid entry is an integer. A value of 0 indicates unknown, the value of 1 represents the server, and a value of 2 represents the client. Note: the value -1 indicates Not Available and the value -2 indicates Not Collected.

Lookups The number of lookups made on the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Lookups Percent Of the total number of calls made to the NFS server, the percentage that were lookups. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Make Directory Calls The number of make directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Make Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were make directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Make Node Calls The number of make node (mknod) calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Make Node Calls Percent Of the total number of calls made to the NFS server, the percentage that were make node (mknod) calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

NFS Calls The total NFS server or client calls. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

NFS Version The software version associated with the NFS server. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Null Calls The number of calls made to the NFS server from NFS clients which contained no data. Valid entry is an integer in the range of 0 to 100. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Null Calls Percent Of the total number of calls made to the NFS server, the percentage that contained no data. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Path Conf Calls The number of calls made to the NFS server that involved path configuration (pathconf) calls to obtain configuration values for files. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Path Conf Call Percent Of the total number of calls made to the NFS server, the percentage that involved use of the pathconf command to obtain configuration values for files. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Read Calls The number of read calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Read Calls Percent Of the total number of calls made to the NFS server, the percentage that were read calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Read Dir Plus Calls The number of read directory plus (readdirplus) calls made to the NFS server to return the name, the file ID, attributes, and file handle. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Read Dir Plus Calls Percent Of the total number of calls made to the NFS server, the percentage that were read directory plus (readdirplus) calls. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Read Directory Calls The number of read directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Read Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were read directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Read Link Calls The number of read link calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Read Link Calls Percent Of the total number of calls made to the NFS server, the percentage that were read link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Remove Directory Calls The number of remove directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Remove Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were remove directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Remove File Calls The number of file removal calls made to the NFS server. Valid entry is an integer. Note: -1 indicates Not_Available, -2 indicates Not_Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum..

Remove File Calls Percent Of the total number of calls made to the NFS server, the percentage that were file removal calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Rename File Calls The number of file rename calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Rename File Calls Percent Of the total number of calls made to the NFS server, the percentage that were file rename calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Root Calls The number of calls made to the NFS server which contained root calls. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Root Calls Percent Of the total number of calls made to the NFS server, the percentage that were root calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Set Attribute Calls The number of calls made to the NFS server which contained a set attribute (setattr) operation. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Set Attribute Calls Percent Of the total number of calls made to the NFS server, the percentage that contained a set attribute (setattr) operation. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Symbolic Link Calls The total number of symbolic link calls. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Symbolic Link Calls Percentage Of the total number of calls made to the NFS server, the percentage that were symbol link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Write Cache Calls The number of write cache calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Write Cache Calls Percent Of the total number of calls made to the NFS server, the percentage that were write cache calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Writes The number of write calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Writes Percent Of the total number of calls made to the NFS server, the percentage that were write calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

NFS Statistics Attributes (superseded)

Use NFS Statistics to monitor characteristics of Network File System (NFS) such as the number of calls, lookups, and operations. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Access Calls The number of access calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Access Calls Percent Of the total number of calls made to the NFS server, the percentage that were access calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Commit Calls The number of file commit calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Commit Calls Percent Of the total number of calls made to the NFS server, the percentage that were file commit calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

File Creates The number of file create calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

File Creates Percent Of the total number of calls made to the NFS server, the percentage that contained file creation operations. Valid entry is an integer in the range of 0 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

File System Info Calls The number of file system information calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

File System Info Calls Percent Of the total number of calls made to the NFS server, the percentage that were calls to obtain information about the file system. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

File System Statistics Calls The number of calls made to the NFS server which requested statistics of the file system. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

File System Statistics Calls Percent Of the total number of calls made to the NFS server, the percentage that involved a request for file system statistics. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Get Attribute Calls The number of calls made to the NFS server which contained a get attribute (getattr) operation. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Get Attribute Calls Percent Of the total number of calls made to the NFS server, the percentage that contained get attribute (getattr) operations. Valid entry is an integer in the range of 0 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Link Calls The total number of link calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Link Calls Percent Of the total number of calls made to the NFS server, the percentage that were link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Location The location of the origin of the call in the Network File System. Valid entry is an integer. A value of 0 indicates unknown, the value of 1 represents the server, and a value of 2 represents the client. Note: the value -1 indicates Not Available and the value -2 indicates Not Collected. (Superseded.)

Lookups The number of lookups made on the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Lookups Percent Of the total number of calls made to the NFS server, the percentage that were lookups. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Make Directory Calls The number of make directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Make Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were make directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Make Node Calls The number of make node (mknod) calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Make Node Calls Percent Of the total number of calls made to the NFS server, the percentage that were make node (mknod) calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

NFS Calls The total NFS server or client calls. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

NFS Version The software version associated with the NFS server. Valid entry is an integer. A value of 2 represents version 2, 3 represents version 3, 4 represents version 4. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Null Calls The number of calls made to the NFS server from NFS clients which contained no data. Valid entry is an integer in the range of 0 to 100. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Null Calls Percent Of the total number of calls made to the NFS server, the percentage that contained no data. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Path Conf Calls The number of calls made to the NFS server that involved path configuration (pathconf) calls to obtain configuration values for files. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Path Conf Call Percent Of the total number of calls made to the NFS server, the percentage that involved use of the pathconf command to obtain configuration values for files. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Read Calls The number of read calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Read Calls Percent Of the total number of calls made to the NFS server, the percentage that were read calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Read Dir Plus Calls The number of read directory plus (readdirplus) calls made to the NFS server to return the name, the file ID, attributes, and file handle. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Read Dir Plus Calls Percent Of the total number of calls made to the NFS server, the percentage that were read directory plus (readdirplus) calls. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Read Directory Calls The number of read directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Read Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were read directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Read Link Calls The number of read link calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Read Link Calls Percent Of the total number of calls made to the NFS server, the percentage that were read link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Remove Directory Calls The number of remove directory calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Remove Directory Calls Percent Of the total number of calls made to the NFS server, the percentage that were remove directory calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Remove File Calls The number of file removal calls made to the NFS server. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Remove File Calls Percent Of the total number of calls made to the NFS server, the percentage that were file removal calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Rename File Calls The number of file rename calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Rename File Calls Percent Of the total number of calls made to the NFS server, the percentage that were file rename calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Root Calls The number of calls made to the NFS server which contained root calls. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Root Calls Percent Of the total number of calls made to the NFS server, the percentage that were root calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Set Attribute Calls The number of calls made to the NFS server which contained a set attribute (setattr) operation. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Set Attribute Calls Percent Of the total number of calls made to the NFS server, the percentage that contained a set attribute (setattr) operation. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Symbolic Link Calls The total number of symbolic link calls. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Symbolic Link Calls Percentage Of the total number of calls made to the NFS server, the percentage that were symbol link calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Write Cache Calls The number of write cache calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Write Cache Calls Percent Of the total number of calls made to the NFS server, the percentage that were write cache calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Writes The number of write calls made to the NFS server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Writes Percent Of the total number of calls made to the NFS server, the percentage that were write calls. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

OS Configuration Attributes

The OS Configuration attributes refer to configuration characteristics such as OS Name and OS Version.

GCC Version The version of the GNU Compiler with which the kernel was compiled.

OS Name The operating system name.

OS Vendor Information The operating system information.

OS Version The operating system version.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Vendor ID The Processor Vendor ID.

Process Attributes

The Process attributes refer to process characteristics such as data set size, kernel scheduling priority, the number of pages of memory, and the number of page faults.

Command Line The process command line string. Valid entry is a text string, with a maximum length of 768 characters.

Cumulative Busy CPU (Percent) The summation of user CPU and system CPU for this process and children.

Cumulative Process System CPU (Percent) The percent of cumulative CPU time spent in kernel mode by process. Valid entry is an integer between 0 and 100.

Cumulative Process User CPU (Percent) The percent of cumulative CPU time spent in user mode by process. Valid entry is an integer between 0 and 100.

Data Resident Set (Pages) The size of the data set based on the number of pages. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Data Size (KB) The data size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807 Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Data Size (MB) The data size (in megabytes) of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Executable Size (KB) The executable size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Executable Size (MB) The executable size (in megabytes) of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Kernel Priority The kernel scheduling priority. Valid entry is an integer between -100 - 100 (-100 is the highest). Real-time processes can have priorities that are negative.

Library Size (KB) The library size (in kilobytes) of the virtual memory. This measurement represents all pages, including unused. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Library Size (MB) The library size (in megabytes) of the virtual memory. This measurement represents all pages, including unused. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Nice Value The standard UNIX nice level (-20 represents the highest level). Valid entry is an integer in the range -20 to 19.

Number of Threads The number of threads started for this process. (Valid only on 2.6 kernel and above.) Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Process Busy CPU (Percent) The summation of User CPU Percent and System CPU Percent for this process.

Process Command Name The name of the process command. Valid entry is an alphanumeric text string, with a maximum length of 32 characters.

Process Count The count of processes with the same name. The name is selected by using the Command Line (UNICODE), CMDLINEU, attribute. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process CPU ID The ID of the process CPU. Valid entry is an integer. Note: -1 indicates Not Available.

Process Dirty Pages Pages that have been modified (dirty) in buffer (main memory), but not yet copied to the cache. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Process ID The identifier of the process. Valid entry is an integer between 0 and 999. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process Parent ID The identifier for the parent process. Valid entry is an integer between 0 and 999. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process Short Term Avg Busy CPU (Percent) The summation of Proc System CPU Norm and Proc User CPU Norm for this process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Process Short Term Avg System CPU (Percent) The short term average of the percentage of CPU time spent in kernel mode by the process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Process Short Term Avg User CPU (Percent) The short term average of the percentage of CPU time spent in user mode by the process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Process State The state of the process (Sleeping, Disk, Running, Zombie, Trace, Dead, or N/A). Valid entry is an integer between -1 and 5, where:

0 = Sleeping

1 = Disk

2 = Running

3 = Zombie

4 = Trace

5 = Dead

-1 = Not_Available

Process System CPU (Percent) The percent of CPU time spent in kernel mode by process. Valid entry is an integer between 0 and 100.

Process User CPU (Percent) The percent of CPU time spent in user mode by process. Valid entry is an integer between 0 and 100.

Resident Set Size (Pages) The number of pages the process has in real memory. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Session ID The session ID. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Shared Lib Resident Set (Pages) The number of pages of shared library set (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Shared Memory (Pages) The number of pages of shared (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Stack Size (KB) The stack size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Stack Size (MB) The stack size (in megabytes) of the virtual memory. Valid entry is an integer. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Text Resident Set (Pages) The number of pages of text resident (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Major Faults The total number of major page faults (including child processes) since the start of the process. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Total Minor Faults The total number of minor page faults (including child processes) since the start of the process. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Total Size (Pages) The number of pages that the process has in real memory. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=9223372036854775807 and Not_Collected=-2.

User to System CPU (Percent) Of the total system CPU usage, the percentage that was user CPU usage. For example, 500% means that user CPU usage is 5 times the system CPU usage. Valid entry is an integer between -10,000 and 10,000.

VM Locked Pages (KB) The size (in kilobytes) of locked pages of the virtual memory. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

VM Locked Pages (MB) The size (in megabytes) of locked pages of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

VM Size (KB) The size (in kilobytes) of the virtual memory. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. Valid values can include Value_Exceeds_Maximum=9223372036854775807.

VM Size MB Virtual memory size in megabytes. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Process Attributes (superseded)

The Process attributes refer to process characteristics such as data set size, kernel scheduling priority, the number of pages of memory, and the number of page faults. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Command Line The process command line string. Valid entry is an alphanumeric text string, with a maximum length of 256 characters. (Superseded.)

Command Line (Unicode) The process command line string. Valid entry is a text string, with a maximum length of 512 bytes. This attribute is globalized (Unicode). (Superseded.)

Cumulative Busy CPU (Percent) The summation of user CPU and system CPU for this process and children. (Superseded.)

Cumulative Process System CPU (Percent) The percent of cumulative CPU time spent in kernel mode by process. Valid entry is an integer between 0 and 100. (Superseded.)

Cumulative Process User CPU (Percent) The percent of cumulative CPU time spent in user mode by process. Valid entry is an integer between 0 and 100. (Superseded.)

Data Resident Set (Pages) The size of the data set based on the number of pages. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Data Size (KB) The data size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Data Size (MB) The data size (in megabytes) of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Executable Size (KB) The executable size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Executable Size (MB) The executable size (in megabytes) of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Kernel Priority The kernel scheduling priority. Valid entry is an integer between -100 - 100 (-100 is the highest). Real-time processes can have priorities that are negative. (Superseded.)

Library Size (KB) The library size (in kilobytes) of the virtual memory. This measurement represents all pages, including unused. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Library Size (MB) The library size (in megabytes) of the virtual memory. This measurement represents all pages, including unused. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Nice Value The standard Linux nice level (-20 represents the highest level). Valid entry is an integer in the range -20 to 19. (Superseded.)

Number of Threads The number of threads started for this process. (Valid only on 2.6 kernel and above.) Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Parent Process ID The identifier for the parent process. Valid entry is an integer between 0 and 999. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Process Busy CPU (Percent) The summation of User CPU Percent and System CPU Percent for this process. (Superseded.)

Process Command Name The name of the process command. Valid entry is an alphanumeric text string, with a maximum length of 32 characters. (Superseded.)

Process Command Name (Unicode) The name of the process command. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Process Count The count of processes with the same name. The name is selected by using the Command Line (UNICODE), CMDLINEU, attribute. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Process CPU ID The ID of the process CPU. Valid entry is an integer. Note: -1 indicates Not Available. (Superseded.)

Process Dirty Pages Pages that have been modified (dirty) in buffer (main memory), but not yet copied to the cache. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Process ID The identifier of the process. Valid entry is an integer between 0 and 999. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Process Short Term Avg Busy CPU (Percent) The summation of Proc System CPU Norm and Proc User CPU Norm for this process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Process Short Term Avg System CPU (Percent) The short term average of the percentage of CPU time spent in kernel mode by the process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Process Short Term Avg User CPU (Percent) The short term average of the percentage of CPU time spent in user mode by the process. CPU percentages are normalized to account for multiple online processors; percentages are normalized to a maximum of 100 percent. This metric is only available through situations and only when the Process name is also specified within the situation predicate. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Process State The state of the process (Sleeping, Disk, Running, Zombie, Trace, Dead, or N/A). (Superseded.) Valid entry is an integer between -1 and 5, where:

0 = Sleeping

1 = Disk

2 = Running

3 = Zombie

4 = Trace

5 = Dead

-1 = Not_Available

Process System CPU (Percent) The percent of CPU time spent in kernel mode by process. Valid entry is an integer between 0 and 100. (Superseded.)

Process User CPU (Percent) The percent of CPU time spent in user mode by process. Valid entry is an integer between 0 and 100. (Superseded.)

Resident Set Size (Pages) The number of pages the process has in real memory. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Session ID The session ID. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Shared Lib Resident Set (Pages) The number of pages of shared library set (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Shared Memory (Pages) The number of pages of shared (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Stack Size (KB) The stack size (in kilobytes) of the virtual memory. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Stack Size (MB) The stack size (in megabytes) of the virtual memory. Valid entry is an integer. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Text Resident Set (Pages) The number of pages of text resident (mmap) memory. mmap is a system API that lets you map a file or device into memory. The mapped pages might be shared so that other processes can access them. Valid entry is an integer. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Major Faults The total number of major page faults (including child processes) since the start of the process. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Total Minor Faults The total number of minor page faults (including child processes) since the start of the process. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Total Size (Pages) The number of pages that the process has in real memory. Valid entry is an integer. (Superseded.) Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

User to System CPU (Percent) Of the total system CPU usage, the percentage that was user CPU usage. For example, 500% means that user CPU usage is 5 times the system CPU usage. Valid entry is an integer between -10,000 and 10,000. (Superseded.)

VM Locked Pages (KB) The size (in kilobytes) of locked pages of the virtual memory. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

VM Locked Pages (MB) The size (in megabytes) of locked pages of the virtual memory. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value -2147483648 indicates Value_Exceeds_Minimum. (Superseded.)

VM Size (KB) The size (in kilobytes) of the virtual memory. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. Valid values can include Value_Exceeds_Maximum=2147483647. (Superseded.)

VM Size MB Virtual memory size in megabytes. This attribute displays as a floating point with a scale of 1. For example 5.2. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Process User Info Attributes

The Process User Info attributes refer to characteristics associated with effective groups, file system groups, real groups, and saved groups.

Effective Group ID The identifier of the effective group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Effective Group Name The effective group name. Valid entry is a text string, with a maximum length of 64 bytes.

Effective User ID The identifier of the effective user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Effective User Name The name of the effective user. Valid entry is a text string, with a maximum length of 64 bytes.

File System Group Name The name of the file system group. Valid entry is a text string, with a maximum length of 64 bytes.

File System Group ID The identifier of the file system group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647.

File System User ID The identifier of the file system user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

File System User Name The name of the file system user. Valid entry is a text string, with a maximum length of 64 bytes.

Process Command Line The Command Line string for the process.

Process Command Name Command name of the process.

Process ID The identifier associated with the process. Valid entries: integers. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process Parent ID The Parent Process ID. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process State The state of the process (Sleeping, Disk, Running, Zombie, Trace, Dead, or N/A). Valid entry is an integer between -1 and 5, where:

0 = Sleeping

1 = Disk

2 = Running

3 = Zombie

4 = Trace

5 = Dead

-1 = Not_Available

Real Group ID The identifier of the real group. Valid entries: simple text string, alphanumeric with a maximum length 16 characters. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Real Group Name The name of the real group. Valid entries: simple text string, with a maximum length 64 bytes.

Real User ID The identifier of the real user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Real User Name The name of the real user. Valid entry is a text string, with a maximum length of 64 bytes.

Saved Group ID The identifier of the saved group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Saved Group Name The name of the saved group. Valid entry is a text string, with a maximum length of 64 bytes.

Saved User ID The identifier of the saved user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Saved User Name The name of the saved user. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode).

Session ID The session ID. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Terminal Device Name of the terminal device that started a process.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

VM Size (MB) Virtual Memory Size in Megabytes. This attribute displays as a floating point with a scale of 1. For example 5.2. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Process User Info Attributes (superseded)

The Process User Info attributes refer to characteristics associated with effective groups, file system groups, real groups, and saved groups. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Command Line (Unicode) Command Line string of the process. (Superseded.)

Effective Group ID The identifier of the effective group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Effective Group Name The effective group name. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Effective Group Name (Unicode) The effective group name. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Effective User ID The identifier of the effective user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Effective User Name The name of the effective user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Effective User Name (Unicode) The name of the effective user. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

File System Group Name The name of the file system group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

File System Group Name (Unicode) The name of the file system group. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

File System Group ID The identifier of the file system group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

File System User ID The identifier of the file system user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

File System User Name The name of the file system user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

File System User Name (Unicode) The name of the file system user. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Process Command Name (Unicode) The Process Command name (Unicode). (Superseded.)

Process ID The identifier associated with the process. Valid entries: integers. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Process Parent ID The Parent Process ID. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Process State The state of the process (Sleeping, Disk, Running, Zombie, Trace, Dead, or N/A). (Superseded.) Valid entry is an integer between -1 and 5, where:

0 = Sleeping

1 = Disk

2 = Running

3 = Zombie

4 = Trace

5 = Dead

-1 = Not_Available

Real Group ID The identifier of the real group. Valid entries: simple text string, alphanumeric with a maximum length 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Real Group Name The name of the real group. Valid entries: simple text string, alphanumeric with a maximum length 16 characters. (Superseded.)

Real Group Name (Unicode) The name of the real group. Valid entries: simple text string, with a maximum length 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Real User ID The identifier of the real user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Real User Name The name of the real user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Real User Name (Unicode) The name of the real user. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Saved Group ID The identifier of the saved group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Saved Group Name The name of the saved group. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Saved Group Name (Unicode) The name of the saved group. Valid entry is a text string, with a maximum length of 64 bytes. (Superseded.)

Saved User ID The identifier of the saved user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Saved User Name The name of the saved user. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Saved User Name (Unicode) The name of the saved user. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Session ID The session ID. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Terminal Device Name of the terminal device that started a process. (Superseded.)

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

VM Size (MB) Virtual Memory Size in Megabytes. This attribute displays as a floating point with a scale of 1. For example 5.2. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

RPC Statistics Attributes

Use RPC Statistics to monitor remote procedure call (RPC) characteristics, such as the number of RPC server calls (including the number of rejected calls), packets that are not valid, and client calls.

RPC Calls Retransmitted The number of client calls that needed to be transmitted again. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Client Calls The number of calls to the server made by the clients of the server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Packets with Malformed Header The number of packets that were received at the server with header records that were not properly formatted. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Server Call Authorization Failures The number of packets that were received at the server with authorizations that were not valid. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Server Calls Rejected The number of calls made to the server, which were rejected. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Server Invalid Client Requests The number of packets that were received at the server, which had client requests that were not valid. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

RPC Total Server Calls Received The total number of calls made to the server (both valid and not valid). Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Times Authentication Refreshed The number of times the authentication of a client was refreshed. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

RPC Statistics Attributes (superseded)

Use RPC Statistics to monitor remote procedure call (RPC) characteristics, such as the number of RPC server calls (including the number of rejected calls), packets that are not valid, and client calls. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

RPC Calls Retransmitted The number of client calls that needed to be transmitted again. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Client Calls The number of calls to the server made by the clients of the server. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Packets with Malformed Header The number of packets that were received at the server with header records that were not properly formatted. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Server Call Authorization Failures The number of packets that were received at the server with authorizations that were not valid. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Server Calls Rejected The number of calls made to the server, which were rejected. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Server Invalid Client Requests The number of packets that were received at the server, which had client requests that were not valid. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

RPC Total Server Calls Received The total number of calls made to the server (both valid and not valid). Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

System Name The managed system name. The form should be *hostname*:*agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Times Authentication Refreshed The number of times the authentication of a client was refreshed. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Sockets Detail Attributes

The Sockets Detail attributes refer to characteristics associated with socket details, including user ID, local and foreign addresses, socket states, and socket protocols.

Foreign Address The address of the remote end of the socket. Like "netstat" * indicates that the address is unassigned/unavailable. Valid entry is an alphanumeric text string, with a maximum length of 16 characters.

Foreign Port The number of the foreign port. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Local Address The address of the local end of the socket, presented as a dotted ip address. Valid entry is an alphanumeric text string, with a maximum length of 16 characters.

Local Port The local port number. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Local Service Name The local port number translated to service name from /etc/services. Valid entry is an alphanumeric text string, with a maximum length of 64 characters.

Receive Queue (Bytes) The count of bytes not copied by the user program connected to this socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Send Queue (Bytes) The count of bytes not acknowledged by the remote host. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Socket Inode The inode used by the socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Socket Owner Name The user name associated with the user ID that owns or started the socket connection. Valid entry is a text string, with a maximum length of 64 bytes.

Socket Protocol Indicates the sockets using this protocol. "Total" includes UNIX® domain sockets not displayed here. Valid entry is an integer, where:

0 = TCP

1 = UDP

2 = RAW

3 = UNIX

-1 = Not Available

-2 = Not Collected

Socket State The state of the socket. Valid entry is an integer, where

1 = ESTABLISHED

2 = SYN_SENT

3 = SYN_RECV

4 = FIN_WAIT1

5 = FIN_WAIT2

6 = TIME_WAIT

7 = CLOSED

8 = CLOSED_WAIT

9 = LAST_ACK

10 = LISTEN

11 = CLOSING

12 = UNKNOWN

Socket UID The user ID of the owner of the socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Sockets Detail Attributes (superseded)

The Sockets Detail attributes refer to characteristics associated with socket details, including user ID, local and foreign addresses, socket states, and socket protocols. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Foreign Address The address of the remote end of the socket. Like “netstat” * indicates that the address is unassigned/unavailable. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Foreign Port The number of the foreign port. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Local Address The address of the local end of the socket, presented as a dotted ip address. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Local Port The local port number. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Local Service Name The local port number translated to service name from /etc/services. Valid entry is an alphanumeric text string, with a maximum length of 64 characters. (Superseded.)

Receive Queue (Bytes) The count of bytes not copied by the user program connected to this socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Send Queue (Bytes) The count of bytes not acknowledged by the remote host. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Socket Inode The inode used by the socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Socket Owner Name (Unicode) The user name associated with the user ID that owns or started the socket connection. Valid entry is a text string, with a maximum length of 64 bytes. This attribute is globalized (Unicode). (Superseded.)

Socket Protocol Indicates the sockets using this protocol. “Total” includes UNIX domain sockets not displayed here. (Superseded.) Valid entry is an integer, where:

0 = TCP

1 = UDP

2 = RAW

3 = UNIX

-1 = Not Available

-2 = Not Collected

Socket State The state of the socket. (Superseded.) Valid entry is an integer, where

1 = ESTABLISHED

2 = SYN_SENT

3 = SYN_RECV

4 = FIN_WAIT1

5 = FIN_WAIT2

6 = TIME_WAIT

7 = CLOSED

8 = CLOSED_WAIT

9 = LAST_ACK

10 = LISTEN

11 = CLOSING

12 = UNKNOWN

Socket UID The user ID of the owner of the socket. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Name The managed system name. The form should be *hostname*:*agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Sockets Status Attributes

The Sockets Status attributes refer to characteristics associated with the status of the Linux system sockets, including protocol names and high water marks used by protocols.

Highest Sockets Used The high water mark of sockets used by this protocol. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Socket Protocol Indicates the sockets using this protocol. "Total" includes UNIX domain sockets not displayed here. Valid entry is an integer, where:

0 = TCP

1 = UDP

2 = RAW

3 = UNIX

4 = FRAG

-1 = TOTAL

-2 = NOT_AVAILABLE

Sockets in Use Sockets in use by protocol. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Sockets Status Attributes (superseded)

The Sockets Status attributes refer to characteristics associated with the status of the Linux system sockets, including protocol names and high water marks used by protocols. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Highest Sockets Used The high water mark of sockets used by this protocol. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Socket Protocol Indicates the sockets using this protocol. "Total" includes UNIX domain sockets not displayed here. (Superseded.) Valid entry is an integer, where:

0 = TCP

1 = UDP

2 = RAW

3 = UNIX

4 = FRAG

-1 = TOTAL

-2 = NOT_AVAILABLE

Sockets in Use Sockets in use by protocol. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

System Name The managed system name. The form should be *hostname*:*agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Swap Rate Attributes

The Swap Rate attributes feature swap space characteristics, including usage rates and days till full data.

Days Until Swap Space Full The predicted number of days till swap space is completely used (moving average). Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Low Water Mark for Free real memory (KB) The lowest level that Free real memory has reached, expressed in kilobytes. Valid entry is an integer. Note: -1 indicates Not Available and -2 indicates Not Collected. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Minimum Days to Swap Full The minimum number of days till swap space is completely used (peak rate based). Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647.

Peak Swap Space Used (MB) The peak swap space used based on snap shots, expressed in megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Swap Space Used (MB) (Moving Average) The moving average of swap space used, expressed in megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Swap Space Used (bytes per hour) The swap space usage rate, expressed in bytes per hour. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Name The managed system name. The form should be *hostname*:*agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Swap Space (MB) (Moving Average) The moving average of total swap space, expressed in megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Swap Rate Attributes (superseded)

The Swap Rate attributes feature swap space characteristics, including usage rates and days till full data. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Days Until Swap Space Full The predicted number of days till swap space is completely used (moving average). Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Low Water Mark for Free real memory (KB) The lowest level that Free real memory has reached, expressed in kilobytes. Valid entry is an integer. Note: -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Minimum Days to Swap Full The minimum number of days till swap space is completely used (peak rate based). Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Peak Swap Space Used (MB) The peak swap space used based on snap shots, expressed in megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Swap Space Used (MB) (Moving Average) The moving average of swap space used, expressed in megabytes. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Swap Space Used (bytes per hour) The swap space usage rate, expressed in bytes per hour. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Swap Space (MB) (Moving Average) The moving average of total swap space, expressed in megabytes. Valid entry is an integer. (Superseded.)

System Statistics Attributes

The System Statistics attributes refer to characteristics associated with system performance such as the number of logged in users, the number of processes per second, and system load statistics.

Context Switches Per Second The number of context switches per second. Calculated on a 30 second interval. Valid entry is an integer. Valid values can include the value Value_Exceeds_Minimum=-9223372036854775808 and the value Value_Exceeds_Maximum=9223372036854775807.

Major Page Faults Per Second Number of major faults per second, these are page faults that directly require the loading of pages from disk. (Kernel 2.6 and greater.) Calculated on a 30 second interval. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Number of Processes in Zombie State Number of processes currently in Zombie State. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Number of User Logins The current number of users logged in. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Page Faults Per Second The total number of page faults per second (major and minor). (Kernel 2.6 and above only.) Calculated on a 30 second interval. Note: the

value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Pages Paged In Per Second The pages paged in per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Pages Paged Out Per Second The pages paged out per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Pages Swapped In The pages swapped in. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Pages Swapped In Per Second The pages swapped in per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Pages Swapped Out The pages swapped out. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Pages Swapped Out Per Second The pages swapped out per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Percent Change Context Switches Per Second The percent change in the number of context switches per second. Valid entry is an integer in the range -100 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Percent Change Processes Created The percent change in the number of processes created per second. Valid entry is an integer in the range -100 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected.

Processes Created Per Second The number of processes created per second. Calculated on a 30 second interval. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. Valid values can include the value Value_Exceeds_Minimum=-9223372036854775808 and the value Value_Exceeds_Maximum=9223372036854775807.

System Load Last 1 Minute The load on the system for the last minute. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Load Last 5 Minutes The load on the system for the last five minutes. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Load Last 15 Minutes The load on the system for the last fifteen minutes. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

System Name The managed system name. The form should be *hostname*:*agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

System Uptime The System Uptime in seconds, however it displays as a time counter on the Tivoli Enterprise Portal. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Number of Processes The total number of processes. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

Total Pages Paged In The total pages paged in. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Total Pages Paged Out The total pages paged out. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

System Statistics Attributes (superseded)

The System Statistics attributes refer to characteristics associated with system performance such as the number of logged in users, the number of processes per second, and system load statistics. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Context Switches Per Second The number of context switches per second. Calculated on a 30 second interval. Valid entry is an integer. Valid values can include the value Value_Exceeds_Minimum=-2147483648 and the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Major Page Faults Per Second Number of major faults per second, these are page faults that directly require the loading of pages from disk. (Kernel 2.6 and greater.)

Calculated on a 30 second interval. (Superseded.) Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum.

Number of Processes in Zombie State Number of processes currently in Zombie State. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Number of User Logins The current number of users logged in. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Page Faults Per Second The total number of page faults per second (major and minor). (Kernel 2.6 and above only.) Calculated on a 30 second interval. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Pages Paged In Per Second The pages paged in per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Pages Paged Out Per Second The pages paged out per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Pages Swapped In The pages swapped in. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Pages Swapped In Per Second The pages swapped in per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Pages Swapped Out The pages swapped out. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Pages Swapped Out Per Second The pages swapped out per second. Calculated on a 30 second interval. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Percent Change Context Switches Per Second The percent change in the number of context switches per second. Valid entry is an integer in the range -100 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Percent Change Processes Created The percent change in the number of processes created per second. Valid entry is an integer in the range -100 to 100. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

Processes Created Per Second The number of processes created per second. Calculated on a 30 second interval. Valid entry is an integer. Note: -1 indicates Not_Available and -2 indicates Not_Collected. (Superseded.)

System Load Last 1 Minute The load on the system for the last minute. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Load Last 5 Minutes The load on the system for the last five minutes. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Load Last 15 Minutes The load on the system for the last fifteen minutes. Valid entry is an integer in the range 0 to 100. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

System Uptime The System Uptime in seconds, however it displays as a time counter on the Tivoli Enterprise Portal. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Number of Processes The total number of processes. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

Total Pages Paged In The total pages paged in. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Total Pages Paged Out The total pages paged out. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

User Login Attributes

The User Login attributes refer to user characteristics such as idle time, user name, location, and login time.

Hostname (From) The hostname associated with the login for the user. Valid entry is an alphanumeric text string, with a maximum length of 256 characters.

Idle Time The number of minutes that have passed since a user last entered a command. Valid entry is a numeric value expressed as minutes in the range 0 to 20160. Use this attribute to check idle time.

Line The terminal device type or line to which the user is connected. Valid entry is an alphanumeric text string, with a maximum length of 16 characters.

Login Time The date and time the user logged in. Valid entry is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Example: To express November 6, 1998, 1:05 p.m., enter 0981106130500000.

System Name The managed system name. The form should be *hostname*:*agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

User Login PID The login ID of the user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=9223372036854775807.

User Name The full name of a user. Valid entry is an alphanumeric text string, with a maximum length of 96 characters.

User Login Attributes (superseded)

The User Login attributes refer to user characteristics such as idle time, user name, location, and login time. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Hostname (From) The hostname associated with the login for the user. Valid entry is an alphanumeric text string, with a maximum length of 256 characters. (Superseded.)

Idle Time The number of minutes that have passed since a user last entered a command. Valid entry is a numeric value expressed as minutes in the range 0 to 20160. Use this attribute to check idle time. (Superseded.)

Line The terminal device type or line to which the user is connected. Valid entry is an alphanumeric text string, with a maximum length of 16 characters. (Superseded.)

Login Time The date and time the user logged in. (Superseded.) Valid entry is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Example: To express November 6, 1998, 1:05 p.m., enter 0981106130500000.

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

User Login PID The login ID of the user. Valid entry is an integer. Valid values can include the value Value_Exceeds_Maximum=2147483647. (Superseded.)

User Name The full name of a user. Valid entry is an alphanumeric text string, with a maximum length of 32 characters. (Superseded.)

User Name (Unicode) The name of the user logging in to access the system. Valid entry is a text string up to 64 bytes. This attribute is globalized (Unicode). (Superseded.)

VM Stats Attributes

The VM Stats attributes refer to memory characteristics such as the size of cached, free, and shared memory.

Free Virtual Storage (MB) The available virtual storage (in megabytes). Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Free Virtual Storage (Percent) Available Virtual Storage in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Memory Cached (MB) The size (in megabytes) of physical memory cached. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Memory Free (MB) The size (in megabytes) of physical memory free. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Memory Free (Percent) The available real memory in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Memory in Buffers (MB) The size (in megabytes) of physical memory in buffers. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Memory Used (MB) The size (in megabytes) of physical memory used. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Memory Used (Percent) The used real memory in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Shared Memory (MB) The size (in megabytes) of physical memory shared. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Swap Space Free (MB) The size (in megabytes) of swap space free. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Swap Space Free (Percent) Available Swap Space (Percent). Note: the value -1 indicates Not Available and -2 indicates Not Collected.

Swap Space Used (MB) The size (in megabytes) of swap space used. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Swap Space Used (Percent) Used Swap Space (Percent). Note: the value -1 indicates Not Available and -2 indicates Not Collected.

System Name The managed system name. The form should be *hostname:agent_code*.

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Memory (MB) The total size (in megabytes) of physical memory. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Total Swap Space (MB) The total size (in megabytes) of swap space. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Total Virtual Storage (MB) The total virtual storage (real plus swap storage) in MB. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Used Virtual Storage (MB) The used virtual storage in MB. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 9223372036854775807 indicates Value_Exceeds_Maximum.

Used Virtual Storage (Percent) The used virtual storage in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected.

VM Stats Attributes (superseded)

The VM Stats attributes refer to memory characteristics such as the size of cached, free, and shared memory. This attribute group is superseded. There is a new attribute group with the same name that replaces it.

Available Virtual Storage (MB) The available virtual storage in MB. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Available Virtual Storage (Percent) The available virtual storage in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Memory Cached (MB) The size (in megabytes) of physical memory cached. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Memory Free (MB) The size (in megabytes) of physical memory free. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Memory in Buffers (MB) The size (in megabytes) of physical memory in buffers. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Memory Used (MB) The size (in megabytes) of physical memory used. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Real Memory Available (Percent) Available Real Memory in Percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Real Memory Used (Percent) Used Real Memory (Percent). Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Shared Memory (MB) The size (in megabytes) of physical memory shared. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Swap Space Available (Percent) Available Swap Space (Percent). Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Swap Space Free (MB) The size (in megabytes) of swap space free. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Swap Space Used (MB) The size (in megabytes) of swap space used. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Swap Space Used (Percent) Used Swap Space (Percent). Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

System Name The managed system name. The form should be *hostname:agent_code*. (Superseded.)

Examples include spark:KLZ or deux.raleigh.ibm.com:KLZ.

In workspace queries, this attribute should be set equal to the value $NODE$ in order to populate the workspace with data. This attribute is generally not included in situations, unless there is a need to customize the situation for a specific managed system.

Time Stamp The date and time the agent collects information as set on the monitored system. (Superseded.) The timestamp format for SCAN and STR functions is CYYMMDDHHMMSSmmm (as in 1020315064501000 for 03/15/02 06:45:01) where:

C = Century (0 for 20th, 1 for 21st)

Y = Year

M = Month

D = Day

H = Hour

M = Minute

S = Second

m = millisecond

Total Memory (MB) The total size (in megabytes) of physical memory. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Total Swap Space (MB) The total size (in megabytes) of swap space. Valid entry is an integer. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Total Virtual Storage (MB) The total virtual storage (real plus swap storage) in MB. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Used Virtual Storage (MB) The used virtual storage in MB. Note: the value -1 indicates Not Available, the value -2 indicates Not Collected, and the value 2147483647 indicates Value_Exceeds_Maximum. (Superseded.)

Used Virtual Storage (Percent) The used virtual storage in percent. Note: the value -1 indicates Not Available and -2 indicates Not Collected. (Superseded.)

Disk capacity planning for historical data

Disk capacity planning for a monitoring agent is a prediction of the amount of disk space to be consumed for each attribute group whose historical data is being collected. Required disk storage is an important factor to consider when you are defining data collection rules and your strategy for historical data collection.

Expected number of instances is a guideline that can be different for each attribute group, because it is the number of instances of data that the agent will return for a given attribute group, and depends upon the application environment that is being monitored. For example, if your attribute group is monitoring each processor on your machine and you have a dual processor machine, the number of instances is 2.

Calculate expected disk space consumption by multiplying the number of bytes per instance by the expected number of instances, and then multiplying that product by the number of samples.Table 3 on page 115 provides the following information required to calculate disk space for the Monitoring Agent for Linux OS:

- *Bytes per instance (agent)* is an estimate of the record length for each row or instance written to the agent disk for historical data collection. This estimate can be used for agent disk space planning purposes.
- *Database bytes per instance (warehouse)* is an estimate of the record length for detailed records written to the warehouse database, if the attribute group is configured to be written to the warehouse. Detailed records are those that have been uploaded from the agent for long-term historical data collection. This estimate can be used for warehouse disk space planning purposes.

- *Aggregate bytes per instance (warehouse)* is an estimate of the record length for aggregate records written to the warehouse database, if the attribute group is configured to be written to the warehouse. Aggregate records are created by the Summarization agent for attribute groups that have been configured for summarization. This estimate can be used for warehouse disk space planning purposes.

The IBM Tivoli Monitoring Installation and Setup Guide contains formulas that can be used to estimate the amount of disk space used at the agent and in the warehouse database for historical data collection of an attribute group.

Table 3. Capacity planning for historical data logged by component

Table	Attribute group	Bytes per instance (agent)	Database bytes per instance (warehouse)	Aggregate bytes per instance (warehouse)
KLZCPU	KLZ_CPU	140	216	598
KLZCPUAVG	KLZ_CPU_Averages	160	313	1001
KLZDISK	KLZ_Disk	492	505	870
KLZDSKIO	KLZ_Disk_IO	220	248	492
KLZDU	KLZ_Disk_Usage_Trends	212	219	584
KLZIOEXT	KLZ_IO_Ext	288	441	1294
KLZNFS	KLZ_NFS_Statistics	412	459	2107
KLZNET	KLZ_Network	385	411	1256
KLZPROC	KLZ_Process	1236	1446	2998
KLZPUSR	KLZ_Process_User_Info	1600	1632	1720
KLZRPC	KLZ_RPC_Statistics	172	176	429
KLZSOCKD	KLZ_Sockets_Detail	324	332	455
KLZSOCKS	KLZ_Sockets_Status	128	127	207
KLZSWPRT	KLZ_Swap_Rate	156	159	364
KLZSYS	KLZ_System_Statistics	264	361	1289
KLZLOGIN	KLZ_User_Login	516	522	559
KLZVM	KLZ_VM_Stats	228	338	1119
LNXALLUSR	Linux_All_Users	180	182	219
LNXCPU	Linux_CPU	184	266	699
LNXCPUAVG	Linux_CPU_Averages	208	380	1170
LNXCPUCON	Linux_CPU_Config	328	335	372
LNXDISK	Linux_Disk	516	523	872
LNXDSKIO	Linux_Disk_IO	240	273	493
LNXDU	Linux_Disk_Usage_Trends	232	232	581
LNXFILCMP	Linux_File_Comparison	1652	1660	1697
LNXFILE	Linux_File_Information	3608	3653	3792
LNXFILPAT	Linux_File_Pattern	1652	1660	1697
LNXGROUP	Linux_Group	172	172	209
LNXPING	Linux_Host_Availability	244	255	343
LNXIOEXT	Linux_IO_Ext	276	474	1327

Table 3. Capacity planning for historical data logged by component (continued)

Table	Attribute group	Bytes per instance (agent)	Database bytes per instance (warehouse)	Aggregate bytes per instance (warehouse)
LNXIPADDR	Linux_IP_Address	574	578	615
LNXMACHIN	Linux_Machine_Information	792	801	838
LNXNFS	Linux_NFS_Statistics	352	392	1740
LNXNET	Linux_Network	345	364	1025
LNXOSCON	Linux_OS_Config	468	460	497
LNXPROC	Linux_Process	1172	1393	2849
LNXPUSR	Linux_Process_User_Info	1432	1469	1557
LNXRPC	Linux_RPC_Statistics	180	177	334
LNXSOCKD	Linux_Sockets_Detail	340	341	456
LNXSOCKS	Linux_Sockets_Status	160	152	228
LNXSWPRT	Linux_Swap_Rate	176	172	365
LNXSYS	Linux_System_Statistics	232	350	1194
LNXLOGIN	Linux_User_Login	552	552	589
LNXVM	Linux_VM_Stats	220	371	1152

For more information about historical data collection, see the *IBM Tivoli Monitoring Administrator's Guide*.

Note: The Linux Process attribute group is eligible for historical collection by default since the Linux Availability Historical workspaces require historical collection to be turned on for this attribute group. However, turning on historical collection for this attribute group is not recommended for all customers - customers who have large number of processes running on systems should weigh the costs (disk space, CPU, etc.) of collecting historical information on this attribute group.

Chapter 5. Situations reference

This chapter contains an overview of situations, references for detailed information about situations, and descriptions of the predefined situations included in this monitoring agent.

About situations

A situation is a logical expression involving one or more system conditions. Situations are used to monitor the condition of systems in your network. You can manage situations from the Tivoli Enterprise Portal by using the Situation editor.

The IBM Tivoli Monitoring agents that you use to monitor your system environment are shipped with a set of predefined situations that you can use as-is or you can create new situations to meet your requirements. Predefined situations contain attributes that check for system conditions common to many enterprises.

Using predefined situations can improve the speed with which you can begin using the Monitoring Agent for Linux OS. You can examine and, if necessary, change the conditions or values being monitored by a predefined situation to those best suited to your enterprise.

Note: The predefined situations provided with this monitoring agent are not read-only. Do not edit these situations and save over them. Software updates will write over any of the changes that you make to these situations. Instead, clone the situations that you want to change to suit your enterprise.

You can display predefined situations and create your own situations using the Situation editor. The left frame of the Situation editor initially lists the situations associated with the Navigator item that you selected. When you click a situation name or create a new situation, the right frame opens with the following tabs:

Formula
Condition being tested

Distribution
List of managed systems (operating systems, subsystems, or applications) to which the situation can be distributed.

Expert Advice
Comments and instructions to be read in the event workspace

Action
Command to be sent to the system

Until Duration of the situation

More information about situations

The *IBM Tivoli Monitoring User's Guide* contains more information about predefined and custom situations and how to use them to respond to alerts.

For a list of the predefined situations for this monitoring agent and a description of each situation, refer to the Predefined situations section below and the information in that section for each individual situation.

Predefined situations

This monitoring agent contains the following predefined situations:

The remaining sections of this chapter contain descriptions of each of these predefined situations. The situations are organized alphabetically.

- Linux_AMS_Alert_Critical
- Linux_Fragmented_File_System
- Linux_Fragmented_File_System_2
- Linux_High_CPU_Overload
- Linux_High_CPU_Overload_2
- Linux_High_CPU_System
- Linux_High_CPU_System_2
- Linux_High_Packet_Collisions
- Linux_High_Packet_Collisions_2
- Linux_High_RPC_Retransmit
- Linux_High_RPC_Retransmit_2
- Linux_High_Zombies
- Linux_High_Zombies_2
- Linux_Low_Pct_Inodes
- Linux_Low_Pct_Inodes_2
- Linux_Low_percent_space
- Linux_Low_percent_space_2
- Linux_Low_Space_Available
- Linux_Low_Space_Available_2
- Linux_Network_Status
- Linux_Network_Status_2
- Linux_NFS_Buffer_High
- Linux_NFS_Buffer_High_2
- Linux_NFS_Getattr_High
- Linux_NFS_Getattr_High_2
- Linux_NFS_rdlink_high
- Linux_NFS_rdlink_high_2
- Linux_NFS_Read_High
- Linux_NFS_Read_High_2
- Linux_NFS_Writes_High
- Linux_NFS_Writes_High_2
- Linux_Packets_Error
- Linux_Packets_Error_2
- Linux_Process_High_Cpu
- Linux_Process_High_Cpu_2
- Linux_Process_stopped
- Linux_Process_stopped_2
- Linux_RPC_Bad_Calls
- Linux_RPC_Bad_Calls_2
- Linux_System_Thrashing

- Linux_System_Thrashing_2

Linux_AMS_Alert_Critical situation

Monitors to determine if one of the following conditions is true:

- A managed agent has exceeded its restart count for the day as configured in the 'maxRestarts' field of its Common Agent Package file.
- A managed agent is overutilizing the available CPU resources as configured in the 'cpuThreshold' field of its Common Agent Package file.
- A managed agent is overutilizing the available system memory resources as configured in the 'memoryThreshold' field of its Common Agent Package file.
- An attempt at auto-restarting a managed agent failed.
- An attempt at starting a stopped or manually stopped managed agent failed.

The formula for this situation is as follows:

```
Alert Message=='Agent exceeded restart count' OR
Alert Message=='Agent overutilizing CPU' OR
Alert Message=='Agent overutilizing memory'  OR
Alert Message=='Agent restart failed'
```

Linux_Fragmented_File_System situation

This situation has been superseded by Linux_Fragmented_File_System_2. Monitors the percentage of i-nodes to disk space. An exception condition occurs when the percentage of i-nodes to disk space used is high, which could indicate high disk fragmentation on the disk.

This situation has the following formula.

```
IF VALUE Linux_Disk.Space_Used_Percent LT 85 AND
VALUE Linux_Disk.Inodes_Used_Percent GT 80
```

Linux_Fragmented_File_System_2 situation

Monitors the percentage of i-nodes to disk space. An exception condition occurs when the percentage of i-nodes to disk space used is high, which could indicate high disk fragmentation on the disk.

This situation has the following formula.

```
IF VALUE KLZ_Disk.Disk_Used_Percent LT 85 AND
VALUE KLZ_Disk.Inodes_Used_Percent *GT 80
```

Linux_High_CPU_Overload situation

This situation has been superseded by Linux_High_CPU_Overload_2. Monitors the percentage of time the processor is busy. An exception condition occurs when the percentage is extremely high.

This situation has the following formula.

```
IF VALUE Linux_CPU.Idle_CPU LT 10.0 AND VALUE Linux_CPU.CPU_ID EQ Aggregate
```

Linux_High_CPU_Overload_2 situation

Monitors the percentage of time the processor is busy. An exception condition occurs when the percentage is extremely high.

This situation has the following formula.

```
IF VALUE KLZ_CPU.Idle_CPU LT 10.0 AND VALUE KLZ_CPU.CPU_ID EQ
Aggregate
```

Linux_High_CPU_System situation

This situation has been superseded by Linux_High_CPU_System_2. Monitors the percentage of processor time that is used for system calls to check for runaway processes. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_CPU.CPU_ID EQ Aggregate AND VALUE
Linux_CPU.System_CPU GT 80.0
```

Linux_High_CPU_System_2 situation

Monitors the percentage of processor time that is used for system calls to check for runaway processes. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_CPU.CPU_ID EQ Aggregate AND VALUE KLZ_CPU.System_CPU
GT 80.0
```

Linux_High_Packet_Collisions situation

This situation has been superseded by Linux_High_Packet_Collisions_2. Monitors the percentage of packet collisions during data transmission. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_Network.Collision_Percent GT 10
```

Linux_High_Packet_Collisions_2 situation

Monitors the percentage of packet collisions during data transmission. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_Network.Collision_Percent GT 10
```

Linux_High_RPC_Retransmit situation

This situation has been superseded by Linux_High_RPC_Retransmit_2. Monitors the percentage of retransmits because of RPC Server calls. An exception condition occurs when the percentage is extremely high.

This situation has the following formula.

```
IF PCTCHANGE Linux_RPC_Statistics.RPC_Client_Calls_Retransmitted GT 10
```

Linux_High_RPC_Retransmit_2 situation

Monitors the percentage of retransmits because of RPC Server calls. An exception condition occurs when the percentage is extremely high.

This situation has the following formula.

```
IF PCTCHANGE KLZ_RPC_Statistics.RPC_Client_Calls_Retransmitted GT
 10
```

Linux_High_Zombies situation

This situation has been superseded by Linux_High_Zombies_2. Monitors the number of processes in zombie state. An exception condition occurs when the number is high.

This situation has the following formula.

```
IF VALUE Linux_Process.State EQ Zombie AND COUNT Linux_Process.State GT 20
```

Linux_High_Zombies_2 situation

Monitors the number of processes in zombie state. An exception condition occurs when the number is high.

This situation has the following formula.

```
IF VALUE KLZ_Process.State EQ Zombie AND COUNT
KLZ_Process.Parent_Process_ID GT 20
```

Linux_Low_Pct_Inodes situation

This situation has been superseded by Linux_Low_Pct_Inodes_2. Monitors the percentage of available i-nodes. An exception condition occurs when the number is low.

This situation has the following formula.

```
IF VALUE Linux_Disk.Inodes_Used_Percent GT 80
```

Linux_Low_Pct_Inodes_2 situation

Monitors the percentage of available i-nodes. An exception condition occurs when the number is low.

This situation has the following formula.

```
IF VALUE KLZ_Disk.Inodes_Used_Percent GT 80
```

Linux_Low_percent_space situation

This situation has been superseded by Linux_Low_percent_space_2. Monitors the percentage of space available on a file system. An exception condition occurs when the percentage is low.

This situation has the following formula.

```
IF VALUE Linux_Disk.Space_Available_Percent LT 15
```

Linux_Low_percent_space_2 situation

Monitors the percentage of space available on a file system. An exception condition occurs when the percentage is low.

This situation has the following formula.

```
IF VALUE KLZ_Disk.Disk_Free_Percent LT 15
```

Linux_Low_Space_Available situation

This situation has been superseded by Linux_Low_Space_Available_2. Monitors the available space on a file system. An exception condition occurs when the amount of space is low.

This situation has the following formula.

```
IF VALUE Linux_Disk.Space_Available LT 7
```

Linux_Low_Space_Available_2 situation

Monitors the available space on a file system. An exception condition occurs when the amount of space is low.

This situation has the following formula.

```
IF VALUE KLZ_Disk.Disk_Free LT 7
```

Linux_Network_Status situation

This situation has been superseded by Linux_Network_Status_2. Monitors whether the Network Interface Card is up or not. An exception condition occurs when the network interface card is not up.

This situation has the following formula.

```
IF VALUE Linux_Network.Interface_Status NE UP
```

Linux_Network_Status_2 situation

Monitors whether the Network Interface Card is up or not. An exception condition occurs when the network interface card is not up.

This situation has the following formula.

```
IF VALUE KLZ_Network.Interface_Status NE UP
```

Linux_NFS_Buffer_High situation

This situation has been superseded by Linux_NFS_Buffer_High_2. Monitors the number of RPC retransmissions with no duplicate acknowledgements. An exception condition occurs when the number of retransmissions is high.

This situation has the following formula.

```
IF VALUE Linux_RPC_Statistics.RPC_Client_Calls_Retransmitted GT 60 AND
 PCTCHANGE Linux_RPC_Statistics.RPC_Client_Times_Authentication_Refreshed GT 5
```

Linux_NFS_Buffer_High_2 situation

Monitors the number of RPC retransmissions with no duplicate acknowledgements. An exception condition occurs when the number of retransmissions is high.

This situation has the following formula.

```
IF VALUE KLZ_RPC_Statistics.RPC_Client_Calls_Retransmitted GT 60
AND PCTCHANGE KLZ_RPC_Statistics.RPC_Client_Times_Authentication_Refreshed
GT 5
```

Linux_NFS_Getattr_High situation

This situation has been superseded by Linux_NFS_Getattr_High_2. Monitors the percentage of NFS server calls to read client attributes. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_NFS_Statistics.NFS_Get_Attribute_Calls_Pct GT 40
```

Linux_NFS_Getattr_High_2 situation

Monitors the percentage of NFS server calls to read client attributes. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_NFS_Statistics.NFS_Get_Attribute_Calls_Pct GT 40
```

Linux_NFS_rdlink_high situation

This situation has been superseded by Linux_NFS_rdlink_high_2. Monitors the percentage of NFS server calls for read link operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_NFS_Statistics.NFS_Read_Link_Pct GT 10
```

Linux_NFS_rdlink_high_2 situation

Monitors the percentage of NFS server calls for read link operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_NFS_Statistics.NFS_Read_Link_Pct GT 10
```

Linux_NFS_Read_High situation

This situation has been superseded by Linux_NFS_Read_High_2. Monitors the percentage of NFS server calls for read operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_NFS_Statistics.NFS_Read_Calls_Pct GT 30
```

Linux_NFS_Read_High_2 situation

Monitors the percentage of NFS server calls for read operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_NFS_Statistics.NFS_Read_Calls_Pct GT 30
```

Linux_NFS_Writes_High situation

This situation has been superseded by Linux_NFS_Writes_High_2. Monitors the percentage of NFS server calls for write operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_NFS_Statistics.NFS_Writes_Pct GT 15
```

Linux_NFS_Writes_High_2 situation

Monitors the percentage of NFS server calls for write operations. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_NFS_Statistics.NFS_Writes_Pct GT 15
```

Linux_Packets_Error situation

This situation has been superseded by Linux_Packets_Error_2. Monitors the percentage of network packets in error. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_Network.Total_Error_Percent GT 10
```

Linux_Packets_Error_2 situation

Monitors the percentage of network packets in error. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_Network.Total_Error_Percent GT 10
```

Linux_Process_High_Cpu situation

This situation has been superseded by Linux_Process_High_Cpu_2. Monitors the percentage of processor time used by a process. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE Linux_Process.Busy_CPU_Pct GT 60.0
```

Linux_Process_High_Cpu_2 situation

Monitors the percentage of processor time used by a process. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF VALUE KLZ_Process.Busy_CPU_Pct GT 60.0
```

Linux_Process_stopped situation

This situation has been superseded by Linux_Process_stopped_2. Monitors the number of stopped processes on the system. An exception condition occurs when the number is high.

This situation has the following formula.

```
IF VALUE Linux_Process.State NE Running AND
VALUE Linux_Process.State NE Sleeping
```

Linux_Process_stopped_2 situation

Monitors the number of stopped processes on the system. An exception condition occurs when the number is high.

This situation has the following formula.

```
IF VALUE KLZ_Process.State NE Running AND VALUE
KLZ_Process.State NE Sleeping
```

Linux_RPC_Bad_Calls situation

This situation has been superseded by Linux_RPC_Bad_Calls_2. Monitors the percentage of rejected RPC server or client calls. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF ( ( VALUE Linux_RPC_Statistics.RPC_Client_Calls_Retransmitted GT 30
) OR ( VALUE Linux_RPC_Statistics.RPC_Server_Calls_Rejected GT 30 ) )
```

Linux_RPC_Bad_Calls_2 situation

Monitors the percentage of rejected RPC server or client calls. An exception condition occurs when the percentage is high.

This situation has the following formula.

```
IF ( ( VALUE KLZ_RPC_Statistics.RPC_Client_Calls_Retransmitted
GT 30 ) OR ( VALUE KLZ_RPC_Statistics.RPC_Server_Calls_Rejected GT 30 )  )
```

Linux_System_Thrashing situation

This situation has been superseded by Linux_System_Thrashing_2. Monitors the swap space paging activity on the system. An exception condition occurs when the activity is extremely high.

This situation has the following formula.

```
IF ( ( VALUE Linux_System_Statistics.Pages_paged_out_per_sec GT 400.0 )
OR ( *VALUE Linux_System_Statistics.Pages_paged_in_per_sec GT 400.0 ) )
```

Linux_System_Thrashing_2 situation

Monitors the swap space paging activity on the system. An exception condition occurs when the activity is extremely high.

This situation has the following formula.

```
IF ( ( VALUE KLZ_System_Statistics.Pages_paged_out_per_sec GT 400
.0 ) OR ( VALUE KLZ_System_Statistics.Pages_paged_in_per_sec
GT 400.0) )
```

Chapter 6. Take Action commands reference

This chapter contains an overview of Take Action commands, references for detailed information about Take Action commands, and a description of the Take Actions commands included in this monitoring agent.

About Take Action commands

Take Action commands can be run from the desktop or included in a situation or a policy.

When included in a situation, the command executes when the situation becomes true. A Take Action command in a situation is also referred to as reflex automation. When you enable a Take Action command in a situation, you automate a response to system conditions. For example, you can use a Take Action command to send a command to restart a process on the managed system or to send a text message to a cell phone.

Advanced automation uses policies to perform actions, schedule work, and automate manual tasks. A policy comprises a series of automated steps called activities that are connected to create a workflow. After an activity is completed, Tivoli Enterprise Portal receives return code feedback, and advanced automation logic responds with subsequent activities prescribed by the feedback.

More information about Take Action commands

For more information about working with Take Action commands, see the *IBM Tivoli Monitoring User's Guide*.

Predefined Take Action commands

This monitoring agent contains the following Take Action commands:

- AMS Start Agent
- AMS Start Agent Instance
- AMS Stop Agent
- AMS Start Management
- AMS Stop Management
- Sample kill Process

The remaining section of this chapter contains a description of this Take Action command. The following information is provided about the Take Action command:

Description
Which actions the command performs on the system to which it is sent

Arguments
List of arguments, if any, for the Take Action with a short description and default value for each one

Destination systems
Where the command is to be executed: on the Managed System (monitoring agent) where the agent resides or on the Managing System (Tivoli Enterprise Monitoring Server) to which it is connected

Usage notes
Additional relevant notes for using the Take Actions

AMS Start Agent action

Description
Use this action to start an agent that is under the management of Agent Management Services. The action includes an optional input field for resetting the Daily Restart Count back to 0. This is helpful when an agent has exceeded its maxRestartCount for the day.

Arguments
Agent Name
The name of the agent as it appears in the Agents' Runtime Status View's Agent Name column.

Daily Restart Count
Value indicating whether to reset the daily restart count. The value 1 indicates True, and the value 0 (default) indicates False.

Process Name
The name of the process representing the agent instance as it appears in the Agents' Runtime Status View's Process Name column.

Destination systems
Managed system

Usage notes
You cannot target the Monitoring Agent for Linux OS with this action. Only the other agents being managed by Agent Management Services running inside of the Monitoring Agent for Linux OS can be targeted with this action.

AMS Start Agent Instance action

Description
Use this action to start a monitoring agent instance of type ITM Windows or ITM UNIX that is under the management of Agent Management Services. The action includes an optional input field for resetting the Daily Restart Count back to 0. This is helpful when an agent instance has exceeded its maxRestartCount for the day.

Arguments
Agent Name
The name of the agent as it appears in the Agents' Runtime Status View's Agent Name column.

Daily Restart Count
Value indicating whether to reset the daily restart count. The value 1 indicates True, and the value 0 (default) indicates False.

Process Name
The name of the process representing the agent instance as it appears in the Agents' Runtime Status View's Process Name column.

Instance Name
The name of the monitoring agent instance as it appears in the Agents' Runtime Status View's Instance Name column.

Destination systems
Managed system

Usage notes
You cannot target the Monitoring Agent for Linux OS with this action. Only the other agents being managed by Agent Management Services running inside of the Monitoring Agent for Linux OS can be targeted with this action.

AMS Stop Agent action

Description
Use this action to stop an agent that is under the management of Agent Management Services. The action will put a running instance of an agent into the 'Manually Stopped' state, meaning that Agent Management Services will not perform any auto-restarts. To prompt Agent Management Services to commence auto-restarting, use the AMS Start Agent command or the AMS Start Agent Instance command to manually put the agent back into a Running state.

Arguments
Process ID
By default, this argument is populated with the Process ID of the particular agent instance selected from the Tivoli Enterprise Portal. To stop all instances of an agent, such as by using the tacmd executeaction AMS Stop Agent command, leave this argument blank.

Destination systems
Managed system

Usage notes
You cannot target the Monitoring Agent for Linux OS with this action. Only the other agents being managed by Agent Management Services running inside of the Monitoring Agent for Linux OS can be targeted with this action.

AMS Start Management action

Description
Use this action to put an agent under the management of Agent Management Services. This management is what provides auto-restart capability.

Destination systems
Managed system

Usage notes
You cannot target the Monitoring Agent for Linux OS with this action. Only the other agents being managed by Agent Management Services running inside of the Monitoring Agent for Linux OS can be targeted with this action.

AMS Stop Management action

Description
Use this action to remove an agent from management by Agent Management Services. The action will cause the Agent Management Services watchdog to stop performing health checks and auto restarts.

Destination systems
Managed system

Usage notes
You cannot target the Monitoring Agent for Linux OS with this action. Only the other agents being managed by Agent Management Services running inside of the Monitoring Agent for Linux OS can be targeted with this action.

Sample_kill_Process action

Description
Kills the process named in the parameter supplied and enables you to issue ad-hoc commands from the Tivoli Enterprise Portal that the Monitoring Agent for Linux OS will execute on your behalf.

Arguments
Process ID
The Process ID (PID) of the process you would like to kill.

Destination systems
Managed system

Usage notes
The kill command is executed directly by the remote Monitoring Agent for Linux OS. Because it is easy to kill processes unintentionally, you need to exercise caution if the monitoring agent is run as superuser (root).

Chapter 7. Policies reference

This chapter contains an overview of policies and references for detailed information about policies.

About policies

Policies are an advanced automation technique for implementing more complex workflow strategies than you can create through simple automation.

A *policy* is a set of automated system processes that can perform actions, schedule work for users, or automate manual tasks. You use the Workflow Editor to design policies. You control the order in which the policy executes a series of automated steps, which are also called activities. Policies are connected to create a workflow. After an activity is completed, Tivoli Enterprise Portal receives return code feedback and advanced automation logic responds with subsequent activities prescribed by the feedback.

Note: For monitoring agents that provide predefined policies, predefined policies are not read-only. Do not edit these policies and save over them. Software updates will write over any of the changes that you make to these policies. Instead, clone the policies that you want to change to suit your enterprise.

More information about policies

For more information about working with policies, see the *IBM Tivoli Monitoring User's Guide*.

For information about using the Workflow Editor, see the *IBM Tivoli Monitoring Administrator's Guide* or the Tivoli Enterprise Portal online help.

For a list of the policies for this monitoring agent and a description of each policy, refer to the "Predefined policies" section below and the information in that section for each individual policy.

Predefined policies

There are no predefined policies for this monitoring agent.

Appendix A. Upgrading for warehouse summarization

The Monitoring Agent for Linux OS made changes to the warehouse collection and summarization characteristics for some agent attribute groups. These changes correct and improve the way warehouse data is summarized, producing more meaningful historical reports. This appendix explains those changes and the implications to your warehouse collection and reporting.

Note: This upgrade is only available from IBM Tivoli Monitoring v6.1.0 to v6.2.1, and is not available for upgrading from IBM Tivoli Monitoring v6.2 to v6.2.1.

Warehouse summarization is controlled on a per-table basis. How the rows in each table are summarized is determined by a set of attributes in each table that are designated as primary keys. There is always one primary key representing the monitored resource, and data is minimally summarized based on this value. For all agents, this primary key is represented internally by the column name, ORIGINNODE; however, the external attribute name varies with each monitoring agent.

One or more additional primary keys are provided for each attribute group to further refine the level of summarization for that attribute group. For example, in an OS agent disk attribute group, a primary key might be specified for the logical disk name that allows historical information to be reported for each logical disk in a computer.

Tables in the warehouse

For a monitoring agent, there are two main types of warehouse tables:

- Raw tables:

  These tables contain the raw information reported by a monitoring agent and written to the warehouse by the Warehouse Proxy agent. Raw tables are named for the attribute group that they represent, for example, lnxallusr.
- Summary tables:

  These tables contain summarized information based on the raw tables and written to the warehouse by the Summarization and Pruning agent. Summarization provides aggregation results over various reporting intervals, for example, hours, days, and so on. Summary table names are based on the raw table name with an appended suffix, for example, lnxallusr_H, lnxallusr_D, and so on.

Effects on summarized attributes

When tables are summarized in the warehouse, the summary tables and summary views are created to include additional columns to report summarization information. Table 4 contains a list of the time periods and the suffixes for the summary tables and views.

Table 4. Time periods and suffixes for summary tables and views

Data collection time period	Summary table suffixes	Summary view suffixes
Hourly	_H	_HV

Table 4. Time periods and suffixes for summary tables and views (continued)

Data collection time period	Summary table suffixes	Summary view suffixes
Daily	_D	_DV
Weekly	_W	_WV
Monthly	_M	_MV
Quarterly	_Q	_QV
Yearly	_Y	_YV

Table 5 shows the expansion to summary columns of some of the most commonly used attribute types.

Table 5. Additional columns to report summarization information

Attribute name	Aggregation type	Additional summarization columns
MyGauge	GAUGE	MIN_MyGauge MAX_MyGauge SUM_MyGauge AVG_MyGauge
MyCounter	COUNTER	TOT_MyCounter HI_MyCounter LO_MyCounter LAT_MyCounter
MyProperty	PROPERTY	LAT_Property

These additional columns are provided only for attributes that are not primary keys. In the cases when an existing attribute is changed to be a primary key, the Summarization and Pruning agent no longer creates summarization values for the attributes, but the previously created column names remain in the table with any values already provided for those columns. These columns cannot be deleted from the warehouse database, but as new data is collected, these columns will not contain values. Similarly, when the primary key for an existing attribute has its designation removed, that attribute has new summarization columns automatically added. As new data is collected, it is used to populate these new column values, but any existing summarization records do not have values for these new columns.

The overall effect of these primary key changes is that summarization information is changing. If these changes result in the old summarization records no longer making sense, you can delete them. As a part of warehouse upgrade, summary views are dropped. The views will be recreated by the Summarization and Pruning agent the next time it runs. Dropping and recreating the views ensure that they reflect the current table structure.

Upgrading your warehouse with limited user permissions

The IBM Tivoli Monitoring warehouse agents (Warehouse Proxy and Summarization and Pruning agents) can dynamically adjust warehouse table definitions based on attribute group and attribute information being loaded into the warehouse. These types of table changes must be done for this monitoring agent for one or both of the following conditions:

- The monitoring agent has added new attributes to an existing attribute group and that attribute group is included in the warehouse.

- The monitoring agent has added a new attribute group and that attribute group is included in the warehouse.

For the warehouse agents to automatically modify the warehouse table definitions, they must have permission to alter warehouse tables. You might not have granted these agents these permissions, choosing instead to manually define the raw tables and summary tables needed for the monitoring agents. Or, you might have granted these permissions initially, and then revoked them after the tables were created.

You have two options to effect the required warehouse table changes during the upgrade process:

- Grant the warehouse agents temporary permission to alter tables

  If using this option, grant the permissions, start historical collection for all the desired tables, allow the Warehouse Proxy agent to add the new data to the raw tables, and allow the Summarization and Pruning agent to summarize data for all affected tables. Then, remove the permission to alter tables
- Make the warehouse table updates manually

  If using this option, you must determine the table structures for the raw and summary tables. If you manually created the tables in the earlier warehouse definition, you already have a methodology and tools to assist you in this effort. You can use a similar technique to update and add new tables for this warehouse migration.

  For a method of obtaining raw table schema, refer to the IBM Redbook,*Tivoli Management Services Warehouse and Reporting*, January 2007, SG24-7290. The chapter that explains warehouse tuning includes a section on creating data tables manually.

Types of table changes

The following types of table changes affect warehouse summarization:

Case 1 - New attribute added to an attribute group and defined as a primary key.

Case 2 - Existing attribute defined as a primary key or had primary key designation removed.

Case 3 - Moving some tables from 4K tablespaces to 8K tablespaces when using DB2 as the warehouse database.

Case 1 and Case 2 are primary key changes. In both cases, new summarization records will not match existing summarized data:

- A new attribute was added to an attribute group and that attribute was defined as a primary key:

  New summarization records will provide more accurate summarization or greater granularity than previous records. Existing summarization records are still available but contain less granular detail if default values are not assigned for the new primary keys.
- An existing attribute was defined as a primary key or the primary key designation was removed:

  If a new key was added, then the new summarization records will provide more accurate summarization or greater granularity than previous records. If a key was removed, then the new summarization records will provide less granularity than previous records, but with the intent of providing more meaningful summarization. Existing summarization records are still available.

Case 3 requires that you move some tables from 4K tablespaces to 8K tablespaces when using DB2 as the warehouse database to avoid errors during summarization and pruning processing.

Table summary

Table 6 provides information to help you determine the effects of primary key and warehouse changes for this monitoring agent. The table shows each attribute group, the current primary keys (in addition to ORIGINNODE) for the attribute group, primary keys that were removed, and whether this table is being included in warehouse reporting.

Table 6. Primary key and warehouse changes for the Monitoring Agent for Linux OS

Attribute group (the attribute group name as it appears in the Tivoli Enterprise Portal)	Current primary keys	Removed primary keys	Warehoused
KLZ_CPU_Averages			Yes
KLZ_CPU	CPU_ID		Yes
KLZ_Disk_IO	Dev_Name		Yes
KLZ_Disk_Usage_Trends	Disk_Name		Yes
KLZ_Disk	Mount_Point Disk_Name		Yes
KLZ_IO_Ext	Device_Name		Yes
KLZ_NFS_Statistics	NFS_Version Location		Yes
KLZ_Network	Network_Interface_Name		Yes
KLZ_Process_User_Info	Process_ID		Yes
KLZ_Process	Process_ID		Yes
KLZ_RPC_Statistics			Yes
KLZ_Sockets_Detail	Socket_Inode		Yes
KLZ_Sockets_Status	Socket_Protocol		Yes
KLZ_Swap_Rate			Yes
KLZ_System_Statistics			Yes
KLZ_User_Login	Login_PID User_Name		Yes
KLZ_VM_Stats			Yes
Linux_All_Users	User_ID_64 User_ID		Yes
Linux_CPU_Averages			Yes
Linux_CPU_Config	CPU_ID		Yes
Linux_CPU	CPU_ID		Yes
Linux_Disk_IO	Dev_Name		Yes
Linux_Disk_Usage_Trends	Disk_Name		Yes
Linux_Disk	Mount_Point_U Disk_Name		Yes
Linux_File_Comparison	File_Name_2 File_Name_1		No
Linux_File_Information	File_Name_U Path_U		No
Linux_File_Pattern	File_Name		No
Linux_Group	Group_ID_64 Group_ID		Yes
Linux_Host_Availability	Target_Host		No

Table 6. Primary key and warehouse changes for the Monitoring Agent for Linux OS (continued)

Attribute group (the attribute group name as it appears in the Tivoli Enterprise Portal)	Current primary keys	Removed primary keys	Warehoused
Linux_IO_Ext	Device_Name		Yes
Linux_IP_Address	IP_Address Network_Interface_Name		No
Linux_Machine_Information			Yes
Linux_NFS_Statistics	NFS_Version Location		Yes
Linux_Network	Network_Interface_Name		Yes
Linux_OS_Config	OS_Name		Yes
Linux_Process_User_Info	Process_ID		Yes
Linux_Process	Process_ID		Yes
Linux_RPC_Statistics			Yes
Linux_Sockets_Detail	Socket_Inode		Yes
Linux_Sockets_Status	Socket_Protocol		Yes
Linux_Swap_Rate			Yes
Linux_System_Statistics			Yes
Linux_User_Login	User_Name_U Login_PID		Yes
Linux_VM_Stats			Yes

Upgrading your warehouse for primary key and tablespace changes

Upgrading your warehouse includes making the following types of changes:

- Case 1 - New attribute is added and is designated as a primary key
  - New attribute and a default value must be added to the raw table and the summarization tables.

    If the attribute group name is not too large for the underlying database, the table name corresponds to the attribute group name. If the attribute group name is too long, a short name is used. The mapping of attribute group names to table names is stored in the WAREHOUSEID table.
  - Case-1 scripts that perform the following actions are provided to assist in this change:
    - Alter existing raw tables
    - Alter existing summary tables
    - Drop existing summary views
  - These changes must be done before the monitoring agent is started and begins exporting data to the Warehouse Proxy agent.
- Case-2 - Existing attributes are changed to either add or remove primary key designation.
  - Existing data is of limited value and should be deleted.
  - Case-2_Truncate scripts that perform the following actions are provided to assist in this change:
    - Remove all records from existing summary tables, preserving existing table definitions
    - Delete the raw data marker allowing raw data to be resummarized

  - Case-2_Drop scripts that perform the following actions are provided to assist in this change:
    - Drop existing summary views
    - Drop existing summary tables
    - Delete the raw data marker allowing raw data to be resummarized
  - These changes are optional, but result in more accurate summarized information.
- Case 3 - Move tables from 4K tablespace to 8K tablespace for selected agents
  - Special processing for selected agents, to move tables from a 4K tablespace to an 8K tablespace.
  - Individual scripts are provided for each summary table to be changed.

Affected attribute groups and supporting scripts

Table 7 shows the attribute groups and summary tables affected for this monitoring agent, the names of the SQL scripts provided to assist in the upgrade process, the types of warehouse databases for which the scripts must be run, and the types of changes (cases) to which the scripts apply.

Table 7. Scripts for affected attribute groups and summary tables for the Monitoring Agent for Linux OS

Attribute group or summary table	File	DB2	Oracle	MS SQL Server	Case 1	Case 2
Linux_All_Users	klz_61migr_Linux_OS_Agent_Case-1.sql	X	X	X	X	
Linux_Group	klz_61migr_Linux_OS_Agent_Case-1.sql	X	X	X	X	

The following types of warehouse objects are affected by these scripts. Review the scripts before running them:

- Case-1.sql

  These scripts affect raw tables, summary tables, and summary views.
- Case-2_Drop.sql

  These scripts affect the summary tables, summary views, and the Summarization and Pruning agent WAREHOUSEMARKER table.
- Case-2_Truncate.sql

  These scripts affect the summary tables and the Summarization and Pruning agent WAREHOUSEMARKER table.

Procedures

The warehouse can be hosted on any of three databases: DB2, Oracle, or Microsoft SQL Server. There are different sets of script files for each type of database. These scripts are provided as part of the monitoring agent Tivoli Enterprise Portal Server support file installation. After installing the Tivoli Enterprise Portal Server support files for the monitoring agent, the files are located on the Tivoli Enterprise Portal Server computer in *install_dir*/CNPS/SQLLIB/WAREHOUSE. There is a subdirectory for each type of database: DB2 for DB2, Oracle for Oracle, and SQLServer for Microsoft SQL Server.

The scripts provide commands for all affected tables and views. If you do not have summarization enabled for some periods, for example, quarterly or yearly, you will not have the corresponding summary tables (_Q, _Y) and summary views (_QV, _YV) in your warehouse database. If you run the scripts that are provided, the

database reports errors for these missing objects. The scripts continue to run the remaining commands. Similarly, if you rerun the scripts, all commands are attempted. If the objects do not exist, or the command cannot be run (especially for the ALTER commands), the scripts continue processing the remaining commands.

DB2 warehouse database procedure

1. Stop *all* running Warehouse Proxy agent instances and the Summarization and Pruning agent.
2. Back up your warehouse database.
3. Copy the scripts from the Tivoli Enterprise Portal Server in one of the following directories to a temporary directory on the system where the warehouse database is located:
   - Windows:

     install dir\CNPS\SQLLIB\WAREHOUSE\DB2
   - UNIX and Linux:

     install dir/*arch*/cq/sqllib/WAREHOUSE/DB2
4. On the system where the warehouse database is located, change to the directory where you placed the script files in Step 3. Then, connect to the warehouse database through the DB2 command line with a user ID that has the authorization to load and alter tables and drop views. Run commands based on the following example to connect, set the schema, and save the script to an output file:

   ```
   db2 connect to WAREHOUS user ITMUSER using ITMPASS
   db2 set current schema="ITMUSER"
   db2 -tv -z log/script.sql.log -f script.sql
   ```

   These parameters are used in the example:
   - WAREHOUS is the database name.
   - ITMUSER is the user name used by the Warehouse Proxy agent.
   - ITMPASS is the password used by the Warehouse Proxy agent.
   - *script.sql* is the name of the script file. See Table 7 on page 138 for the script file names.
   - *script.sql*.log is the name of the output file.

   Notes®: You might receive error messages such the following from DB2:
   - `SQL0204N "schema name.table name" is an undefined name. SQLSTATE=42704`

     This message indicates that the table named *table name* does not exist and cannot be altered or dropped. This happens if you do not have warehousing or summarization enabled for the given table. For example if you only have hourly and daily summarization enabled, you see this message for the weekly, monthly, quarterly, and yearly summarization tables because these tables do not exist.
   - `SQL3304N The table does not exist.`

     This message indicates that the table does not exist and cannot be loaded. This happens if you do not have warehousing or summarization enabled for the given table. For example if you only have hourly and daily summarization enabled, you see this message for the weekly, monthly, quarterly, and yearly summarization tables because these tables do not exist.

Oracle warehouse database procedure

1. Stop *all* running Warehouse Proxy agent instances and the Summarization and Pruning agent.
2. Back up your warehouse database.
3. Copy the scripts from The Tivoli Enterprise Portal Server in one of the following directories to a temporary directory on the system where the warehouse database is located:
   - Windows

     install dir\CNPS\SQLLIB\WAREHOUSE\Oracle
   - UNIX and Linux

     install dir/*arch*/cq/sqllib/WAREHOUSE/Oracle
4. On the system where the warehouse database is located, change to the directory where you placed the script files in Step 3. Then, connect to the warehouse database through the Oracle command line with the same user that the Warehouse Proxy agent uses to connect to the warehouse, and run the script. To run the script, the user ID must have authorization to alter tables and drop views, or to drop tables when using Case 2 Drop, or truncate tables when using Case 2 Truncate. The output is saved to a file named *script name*.log. Run the following command:

   ```
   sqlplus ITMUSER/ITMPASS@WAREHOUS @script.sql
   ```

   These parameters are used in the example:
   - WAREHOUS is the connect identifier.
   - ITMUSER is the user name used by the Warehouse Proxy agent.
   - ITMPASS is the password used by the Warehouse Proxy agent.
   - *script.sql* is the name of this script file. See Table 7 on page 138 for the script file names.

   Note: You might receive error messages such as the following from Oracle:
   ```
   ORA-00942: table or view does not exist
   ```

   This message indicates that the table does not exist and cannot be altered, dropped, or truncated. This happens if you do not have warehousing or summarization enabled for the given table. For example if you only have hourly and daily summarization enabled, you see this message for the weekly, monthly, quarterly, and yearly summarization tables because these tables do not exist.

MS SQL warehouse database procedure

1. Stop *all* running Warehouse Proxy agent instances and the Summarization and Pruning agent.
2. Back up your warehouse database.
3. Copy the scripts from the Tivoli Enterprise Portal Server in the one of the following directories to a temporary directory on the system where the warehouse database is located:
   - Windows:

     install dir\CNPS\SQLLIB\WAREHOUSE\SQLServer
   - UNIX and Linux:

     install dir/*arch*/cq/sqllib/WAREHOUSE/SQLServer
4. On the system where the warehouse database is located, change to the directory where you placed the script files in Step 3. Then, connect to the

warehouse database through the SQL Server command line with the same user that the Warehouse Proxy agent uses to connect to the warehouse, and run the script. To run the script, the user ID must have authorization to alter tables and drop views, or to drop tables when using Case 2 Drop, or truncate tables when using Case 2 Truncate. The output is saved to a file named *script name*.log. Run the following command:

```
osql -I -S SQLHOST[\SQLINST] -U ITMUSER -P ITMPASS -d WAREHOUS
   -m-1 -n -o log/script.sql.log -i script.sql
```

These parameters are used in the example:

- WAREHOUS is the database name.
- ITMUSER is the user name used by the Warehouse Proxy agent.
- ITMPASS is the password used by the Warehouse Proxy agent.
- *script.sql* is the name of this script file.
- SQLHOST is the SQL server name.
- SQLINST is the optional SQL instance name.

Note: You might receive error messages from the SQL Server such as the following: `Msg 4902, Level 16, State 1, Server ENTERPRISE, Line 1 Cannot find the object "`*`table name`*`" because it does not exist or you do not have permissions.`

This message indicates that the table named *table name* does not exist and cannot be dropped or truncated. This happens if you do not have warehousing or summarization enabled for the given table. For example if you only have hourly and daily summarization enabled, you see this message for the weekly, monthly, quarterly, and yearly summarization tables because these tables do not exist.

Appendix B. IBM Tivoli Enterprise Console event mapping

Specific event mapping is provided for those monitoring agents that support Distributed Monitoring migration. The specific event mapping creates Distributed Monitoring events for Distributed Monitoring migrated situations. For a list of these situations and their related event classes, see Table 8.

Generic event mapping provides useful event class and attribute information for situations that do not have specific event mapping defined. Each event class corresponds to an attribute group in the monitoring agent. For a description of the event slots for each event class, see Table 9 on page 145. For more information about mapping attribute groups to event classes, see the *IBM Tivoli Monitoring Administrator's Guide*.

BAROC files are found on the Tivoli Enterprise Monitoring Server in the installation directory in TECLIB (that is, *install_dir*/cms/TECLIB for Windows systems and *install_dir*/tables/*TEMS_hostname*/TECLIB for UNIX systems). IBM Tivoli Enterprise Console event synchronization provides a collection of ready-to-use rule sets that you can deploy with minimal configuration. Be sure to install IBM Tivoli Enterprise Console event synchronization to access the correct Sentry.baroc, which is automatically included during base configuration of IBM Tivoli Enterprise Console rules if you indicate that you want to use an existing rulebase. See the *IBM Tivoli Monitoring Installation and Setup Guide* for details.

Table 8. Overview of Distributed Monitoring migrated situations

Situation	IBM Tivoli Enterprise Console event class
LZ_USInodes*	Sentry2_0_inodes Sentry2_0_inodesused
LZ_USIUsPct*	Sentry2_0_inodesusedpct
LZ_USDkUPct*	Sentry2_0_diskusedpct
LZ_USDskAva*	Sentry2_0_diskavail
LZ_USDskUsd*	Sentry2_0_diskused
LZ_USTvDBSp*	Sentry2_0_tivdbspace
LZ_USDIORtK*	Sentry2_0_diskioratek
LZ_USRCPTmo*	Sentry2_0_rpctmout
LZ_USNtInEr*	Sentry2_0_netinerr
LZ_USNtInEX*	Sentry2_0_netinerrx
LZ_USNetIn*	Sentry2_0_netinerr
LZ_USNetInX*	Sentry2_0_netinx
LZ_USBadNFS*	Sentry2_0_badnfs
LZ_USBadNFS*	Sentry2_0_badnfs
LZ_USNetCol*	Sentry2_0_netcoll
LZ_USNCPct*	Sentry2_0_netcollpct
LZ_USNCPctX*	Sentry2_0_netcollpctx
LZ_USNetOEr*	Sentry2_0_netouterr
LZ_USNetOEX*	Sentry2_0_netouterrx

Table 8. Overview of Distributed Monitoring migrated situations (continued)

Situation	IBM Tivoli Enterprise Console event class
LZ_USNetOut*	Sentry2_0_netouterr
LZ_USNetOX*	Sentry2_0_netoutx
LZ_USBadRPC*	Sentry2_0_badrpc
LZ_USSwpAva*	Sentry2_0_swapavail
LZ_USCPUIdl*	Sentry2_0_cpuidle
LZ_USCPUSys*	Sentry2_0_cpusys
LZ_USCPUUsr*	Sentry2_0_cpuusr
LZ_USCPUSdu*	Sentry2_0_cpusdu
LZ_USCPUSpu*	Sentry2_0_cpuspu
LZ_USZombie*	Sentry2_0_zombies
LZ_USLdAv15*	Sentry2_0_loadavgfifteenm
LZ_USLdAv5*	Sentry2_0_loadavgonem
LZ_USLdAv1*	Sentry2_0_loadavgonem
LZ_USPgIns*	Sentry2_0_pageins
LZ_USPgOuts*	Sentry2_0_pageouts
LZ_USACPUBu*	Sentry2_0_avgcpubusy
LZ_UDskAva*	universal_diskavail
LZ_UDskUsd*	universal_diskused
LZ_UDskUPct*	universal_diskusedpct
LZ_UIndsFre*	universal_diskusedpct
LZ_UIndsUsd*	universal_diskusedpct
LZ_ULoadAvg*	universal_loadavg
LZ_UPageOut*	universal_pageouts
LZ_USwapAva*	universal_swapavail

To determine what event class is sent when a given situation is triggered, look at the first referenced attribute group in the situation predicate. The event class that is associated with that attribute group is the one that is sent. This is true for both pre-packaged situations and user-defined situations. See the table below for attribute group to event classes and slots mapping information.

For example, if the situation is monitoring the No Password attribute from the All Users Group attribute group, the event class that is sent once the situation is triggered is ITM_Linux_All_Users.

Note: There are cases where these mappings generate events that are too large for the Tivoli Enterprise Console. In these cases, the event class names and the event slot names are the same, but some of the event slots are omitted.

Each of the event classes is a child of KLZ_Base. The KLZ_Base event class can be used for generic rules processing for any event from the Monitoring Agent for Linux OS.

Table 9. Overview of attribute groups to event classes and slots

Attribute group	event classes and slots
Linux_User_Login	ITM_Linux_User_Login event class with these slots: • system_name: STRING • timestamp: STRING • user_name: STRING • login_pid: INTEGER • login_pid_enum: STRING • line: STRING • login_time: STRING • idle_time: STRING • from_hostname: STRING • user_name_u: STRING • linux_vm_id: STRING
Linux_Disk	ITM_Linux_Disk event class with these slots: • system_name: STRING • timestamp: STRING • disk_name: STRING • mount_point: STRING • size: INTEGER • size_enum: STRING • space_used: INTEGER • space_used_enum: STRING • space_available: INTEGER • space_available_enum: STRING • total_inodes: INTEGER • total_inodes_enum: STRING • inodes_used: INTEGER • inodes_used_enum: STRING • inodes_free: INTEGER • inodes_free_enum: STRING • space_used_percent: INTEGER • space_used_percent_enum: STRING • inodes_used_percent: INTEGER • inodes_used_percent: STRING • fs_type: STRING • space_available_percent: INTEGER • space_available_percent_enum: STRING • mount_point_u: STRING • linux_vm_id: STRING • inodes_available_percent: INTEGER • inodes_available_percent_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Disk_Usage_Trends	ITM_Linux_Disk_Usage_Trends event class with these slots: • system_name: STRING • timestamp: INTEGER • disk_name: STRING • space_used: INTEGER • space_used_enum: STRING • space_available: INTEGER • space_available_enum: STRING • disk_usage_rate: INTEGER • disk_usage_rate_enum: STRING • highwater_du_rate: INTEGER • highwater_du_rate_enum: STRING • highwater_time: STRING • disk_usage_moving_average: INTEGER • disk_usage_moving_average_enum: STRING • days_until_full_disk: INTEGER • days_until_full_disk_enum: STRING • days_full_disk_curr: INTEGER • days_full_disk_curr_enum: STRING • low_water_full_disk_curr: STRING • low_water_full_disk_curr_enum: STRING • days_full_disk_peak: INTEGER • days_full_disk_peak_enum: STRING • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Network	ITM_Linux_Network event class with these slots: • system_name: STRING • timestamp: INTEGER • network_interface_name: STRING • interface_ip_address: STRING • interface_dns_name: STRING • interface_status: INTEGER • interface_status_enum: STRING • transmission_unit_maximum: INTEGER • transmission_unit_maximum_enum: STRING • kbytes_received_count: INTEGER • kbytes_received_count_enum: STRING • bytes_received_per_sec: INTEGER • bytes_received_per_sec_enum: STRING • kbytes_transmitted_count: INTEGER • kbytes_transmitted_count_enum: STRING • bytes_transmitted_per_sec: INTEGER • bytes_transmitted_per_sec_enum: STRING • packets_received_count: INTEGER • packets_received_count_enum: STRING • packets_received_per_sec: INTEGER • packets_received_per_sec_enum: STRING • input_errors: INTEGER • input_errors_enum: STRING • output_errors: INTEGER • output_errors_enum: STRING • packets_transmitted_per_sec: INTEGER • packets_transmitted_per_sec_enum: STRING • input_errors: INTEGER • input_errors_enum: STRING • output_errors: INTEGER • output_errors_enum: STRING • collisions: INTEGER • collisions_enum: STRING • collision_rate: INTEGER • collision_rate_enum: STRING • collision_percent: INTEGER

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Network (Continued)	• input_error_rate: INTEGER • input_error_rate_enum: STRING • output_error_rate: INTEGER • output_error_rate_enum: STRING • total_error_percent: INTEGER • input_packets_dropped: INTEGER • input_packets_dropped_enum: STRING • output_packets_dropped: INTEGER • output_packets_dropped_enum: STRING • input_fifo_buffer_overruns: INTEGER • input_fifo_buffer_overruns_enum: STRING • output_fifo_buffer_overruns: INTEGER • output_fifo_buffer_overruns_enum: STRING • packet_framing_errors: INTEGER • packet_framing_errors_enum: STRING • carrier_losses: INTEGER • carrier_losses_enum: STRING • linux_vm_id: STRING • input_error_percent: INTEGER • output_error_percent: INTEGER • device_type: INTEGER • device_type_enum: STRING • mac_address: STRING • mac_address_enum: STRING
User	ITM_Linux_CPU event class with these slots: • system_name: STRING • timestamp: STRING • cpu_id: INTEGER • cpu_id_enum: STRING • user_cpu: REAL • user_nice_cpu: REAL • system_cpu: REAL • idle_cpu: REAL • busy_cpu: REAL • wait_io_cpu: REAL • user_sys_pct: INTEGER • steal_time_cpu: REAL • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_CPU_Averages	ITM_Linux_CPU_Averages event class with these slots: • system_name: STRING • timestamp: STRING • days_to_cpu_upgrade: REAL • days_to_cpu_upgrade_enum: STRING • cpu_usage_current_average: REAL • cpu_usage_moving_average: REAL • user_nice_cpu_current_average: REAL • user_nice_cpu_moving_average: REAL • user_cpu_current_average: REAL • user_cpu_moving_average: REAL • system_cpu_current_average: REAL • system_cpu_moving_average: REAL • idle_cpu_current_average: REAL • idle_cpu_moving_average: REAL • wait_cpu_current_average: REAL • wait_cpu_moving_average: REAL • steal_cpu_current_average: REAL • steal_cpu_current_average_enum: STRING • steal_cpu_moving_average: REAL • steal_cpu_moving_average_enum: REAL • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Process	ITM_Linux_Process event class with these slots: • system_name: STRING • timestamp: INTEGER • process_id: REAL • parent_process_id: INTEGER • process_command_name: STRING • state: INTEGER • state_enum: STRING • proc_system_cpu: REAL • proc_user_cpu: REAL • tot_proc_system_cpu: REAL • tot_proc_user_cpu: REAL • priority: INTEGER • nice: INTEGER • total_size_memory: INTEGER • total_size_memory_enum: STRING • resident_set_size: INTEGER • resident_set_size_enum: STRING • shared_memory: INTEGER • shared_memory_enum: STRING • text_resident_size: INTEGER • text_resident_size_enum: STRING • shared_lib_set_size: INTEGER • shared_lib_set_size_enum: STRING • data_set_size: INTEGER • data_set_size_enum: STRING • dirty_pages: INTEGER • dirty_pages_enum: STRING • vm_size: INTEGER • vm_size_enum: STRING • vm_lock: INTEGER • vm_lock_enum: STRING • vm_data: INTEGER • vm_data_enum: STRING • vm_stack: INTEGER • vm_stack_enum: STRING • vm_exe_size: INTEGER • vm_exe_size_enum: STRING • vm_lib_size: INTEGER • vm_lib_size_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Process (continued)	• tot_minor_faults: INTEGER • tot_minor_faults_enum: STRING • tot_major_faults: INTEGER • tot_major_faults_enum: STRING • proc_cmd_line: STRING • proc_cmd_line_u: STRING • proc_cpu: INTEGER • proc_cpu_enum: STRING • linux_vm_id: STRING • user_sys_cpu_pct: INTEGER • process_command_name_u: STRING • total_busy_cpu_pct: REAL • busy_cpu_pct: REAL • vm_size_mb: REAL • vm_size_mb_enum: STRING • vm_lock_mb: REAL • vm_lock_mb_enum: STRING • vm_data_mb: REAL • vm_data_mb_enum: STRING • vm_stack_mb: REAL • vm_stack_mb_enum: STRING • vm_exe_size_mb: REAL • vm_exe_size_mb_enum: STRING • vm_lib_size_mb: REAL • vm_lib_size_mb_enum: STRING • threads: INTEGER • threads_enum: STRING • session_id: INTEGER • session_id_enum: STRING • proc_system_cpu_norm: REAL • proc_system_cpu_norm_enum: STRING • proc_user_cpu_norm: REAL • proc_user_cpu_norm_enum: STRING • proc_busy_cpu_norm: REAL • proc_busy_cpu_norm_enum: STRING • process_count: INTEGER • process_count_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Process_User_Info	ITM_Linux_Process_User_Info event class with these slots: • system_name: STRING • timestamp: STRING • process_id: INTEGER • process_id_enum: STRING • real_user_name: STRING • eff_user_name: STRING • saved_user_name: STRING • fs_user_name: STRING • real_group: STRING • eff_group: STRING • saved_group: STRING • file_sys_group: STRING • real_user_id: INTEGER • real_user_id_enum: STRING • eff_user_id: INTEGER • eff_user_id_enum: STRING • saved_user_id: INTEGER • saved_user_id_enum: STRING • fs_user_id: INTEGER • fs_user_id_enum: STRING • real_group_id: INTEGER • real_group_id_enum: STRING • eff_group_id: INTEGER • eff_group_id_enum: STRING • saved_group_id: INTEGER • saved_group_id_enum: STRING • file_sys_group_id: INTEGER • file_sys_group_id_enum: STRING • real_user_name_u: STRING • eff_user_name_u: STRING • saved_user_name_u: STRING • fs_user_name_u: STRING • real_group_u: STRING • eff_group_u: STRING • saved_group_u: STRING • file_sys_group_u: STRING • linux_vm_id: STRING • session_id: INTEGER • session_id_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Process_User_Info (Continued)	• parent_process_id: INTEGER • parent_process_id_enum: STRING • state: INTEGER • state_enum: STRING • proc_cmd_line_u: STRING • process_command_name_u: STRING • vm_size_mb: REAL • vm_size_mb_enum: STRING • terminal_device: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_System_Statistics	ITM_Linux_System_Statistics event class with these slots: • system_name: STRING • timestamp: STRING • ctxt_switches_per_sec: INTEGER • ctxt_switches_per_sec_enum: STRING • pct_change_ctxt_switches: REAL • processes_per_sec: INTEGER • pct_change_processes: REAL • pct_change_processes_enum: STRING • number_of_users: INTEGER • number_of_users_enum: STRING • system_load_1min: REAL • system_load_1min_enum: STRING • system_load_5min: REAL • system_load_5min_enum: STRING • system_load_15min: REAL • system_load_15min_enum: STRING • system_uptime: INTEGER • system_uptime_enum: STRING • linux_vm_id: STRING • pages_paged_in: INTEGER • pages_paged_in_enum: STRING • pages_paged_in_per_sec: REAL • pages_paged_in_per_sec_enum: STRING • pages_paged_out: INTEGER • pages_paged_out_enum: STRING • pages_paged_out_per_sec: REAL • pages_paged_out_per_sec_enum: STRING • pages_swapped_in: INTEGER • pages_swapped_in_enum: STRING • pages_swap_in_per_sec: REAL • pages_swap_in_per_sec_enum: STRING • pages_swapped_out: INTEGER • pages_swapped_out_enum: STRING • pages_swap_out_per_sec: REAL • pages_swap_out_per_sec_enum: STRING • page_faults_per_sec: INTEGER • page_faults_per_sec_enum: STRING • major_faults_per_sec: INTEGER • major_faults_per_sec_enum: STRING • total_number_processes: INTEGER • total_number_processes_enum: STRING • processes_zombie_count: INTEGER • processes_zombie_count_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Swap_Rate	ITM_Linux_Swap_Rate event class with these slots: • system_name: STRING • timestamp: STRING • moving_total_swap_space: INTEGER • moving_total_swap_space_enum: STRING • swap_space_used: INTEGER • swap_space_used_enum: STRING • swap_usage_rate: INTEGER • swap_usage_rate_enum: STRING • days_to_swap_space_full: INTEGER • days_to_swap_space_full_enum: STRING • peak_swap_space_used: INTEGER • peak_swap_space_used_enum: STRING • days_to_peak_space_full: INTEGER • days_to_peak_space_full_enum: STRING • low_free_memory: INTEGER • low_free_memory_enum: STRING • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_VM_Stats	ITM_Linux_VM_Stats event class with these slots: • system_name: STRING • timestamp: STRING • total_swap_space: REAL • total_swap_space_enum: STRING • swap_space_used: REAL • swap_space_used_enum: STRING • swap_usage_free: REAL • swap_usage_free_enum: STRING • total_memory: REAL • total_memory_enum: STRING • memory_used: REAL • memory_used_enum: STRING • memory_free: REAL • memory_free_enum: STRING • shared_memory: REAL • shared_memory_enum: STRING • memory_in_buffers: REAL • memory_in_buffers_enum: STRING • memory_cached: REAL • memory_cached_enum: STRING • linux_vm_id: STRING • total_virtual_storage: REAL • total_virtual_storage_enum: STRING • used_virtual_storage: REAL • used_virtual_storage_enum: STRING • available_virtual_storage: REAL • available_virtual_storage_enum: STRING • virtual_storage_pct_avail: INTEGER • virtual_storage_pct_avail_enum: STRING • virtual_storage_pct_used: INTEGER • virtual_storage_pct_used_enum: STRING • real_memory_pct_used: INTEGER • real_memory_pct_used_enum: STRING • real_memory_pct_avail: INTEGER • real_memory_pct_avail_enum: STRING • swap_pct_used: INTEGER • swap_pct_used_enum: STRING • swap_pct_avail: INTEGER • swap_pct_avail_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Sockets_Status	ITM_Linux_Sockets_Status event class with these slots: • system_name: STRING • timestamp: STRING • socket_protocol: INTEGER • socket_protocol_enum: STRING • sockets_in_use: INTEGER • sockets_in_use_enum: STRING • highest_sockets_used: INTEGER • highest_sockets_used_enum: STRING • linux_vm_id: STRING
Linux_Sockets_Detail	ITM_Linux_Sockets_Detail event class with these slots: • system_name: STRING • timestamp: STRING • socket_protocol: INTEGER • socket_protocol_enum: STRING • receive_queue: INTEGER • receive_queue_enum: STRING • send_queue: INTEGER • send_queue_enum: STRING • local_address: STRING • local_port: INTEGER • local_port_enum: STRING • local_service: STRING • foreign_address: STRING • socket_state: INTEGER • socket_state_enum: STRING • socket_uid: INTEGER • socket_uid_enum: STRING • socket_inode: INTEGER • socket_inode_enum: STRING • foreign_port: INTEGER • foreign_port_enum: STRING • socket_owner_name_u: STRING • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Disk_IO	ITM_Linux_Disk_IO event class with these slots: • system_name: STRING • timestamp: STRING • transfers_per_sec: REAL • transfers_per_sec_enum: STRING • blk_rds_per_sec: REAL • blk_rds_per_sec_enum: STRING • blk_wrtn_per_sec: REAL • blk_wrtn_per_sec_enum: STRING • blks_read: INTEGER • blks_read_enum: STRING • blks_wrtn: INTEGER • blks_wrtn_enum: STRING • dev_major: INTEGER • dev_major_enum: STRING • dev_minor: INTEGER • dev_minor_enum: STRING • dev_name: STRING • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_IO_Ext	ITM_Linux_IO_Ext event class with these slots: • system_name: STRING • timestamp: STRING • device_name: STRING • read_reqm_per_sec: REAL • read_reqm_per_sec_enum: STRING • write_reqm_per_sec: REAL • write_reqm_per_sec_enum: STRING • read_req_per_sec: REAL • read_req_per_sec_enum: STRING • write_req_per_sec: REAL • write_req_per_sec_enum: STRING • read_sect_per_sec: REAL • read_sect_per_sec_enum: STRING • write_sect_per_sec: REAL • write_sect_per_sec_enum: STRING • avg_req_size: REAL • avg_req_size_enum: STRING • avg_req_queue_length: REAL • avg_req_queue_length_enum: STRING • avg_wait_time: REAL • avg_wait_time_enum: STRING • avg_svc_time: REAL • avg_svc_time_enum: STRING • cpu_util: REAL • cpu_util_enum: STRING • linux_vm_id: STRING • disk_read_percent: REAL • disk_write_percent: REAL • read_bytes_per_sec: REAL • read_bytes_per_sec_enum: STRING • write_bytes_per_sec: REAL • write_bytes_per_sec_enum: STRING • transfers_bytes_per_sec: REAL • transfers_bytes_per_sec_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_RPC_Statistics	ITM_Linux_RPC_Statistics event class with these slots: • system_name: STRING • timestamp: STRING • rpc_server_total_calls: INTEGER • rpc_server_total_calls_enum: STRING • rpc_server_calls_rejected: INTEGER • rpc_server_calls_rejected_enum: STRING • rpc_server_packets_bad_auth: INTEGER • rpc_server_packets_bad_auth_enum: STRING • rpc_server_packets_bad_clt: INTEGER • rpc_server_packets_bad_clt_enum: STRING • rpc_server_packets_with_malformed_ header: INTEGER • rpc_server_packets_with_malformed_ header_enum: STRING • rpc_client_calls: INTEGER • rpc_client_calls_enum: STRING • rpc_client_calls_retransmitted: INTEGER • rpc_client_calls_retransmitted_enum: STRING • rpc_client_times_authentication_refreshed: INTEGER • rpc_client_times_authentication_refreshed_ enum: STRING • linux_vm_id: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_NFS_Statistics	ITM_Linux_NFS_Statistics event class with these slots: • location: INTEGER • location_enum: STRING • nfs_version: INTEGER • nfs_version_enum: STRING • nfs_null_calls: INTEGER • nfs_null_calls_enum: STRING • nfs_null_call_percentage: INTEGER • nfs_null_call_percentage_enum: STRING • nfs_get_attribute_calls: INTEGER • nfs_get_attribute_calls_enum: STRING • nfs_get_attribute_calls_pct: INTEGER • nfs_get_attribute_calls_pct_enum: STRING • nfs_set_attribute_calls: INTEGER • nfs_set_attribute_calls_enum: STRING • nfs_set_attrib_calls_pct: INTEGER • nfs_set_attrib_calls_pct_enum: STRING • nfs_root_calls: INTEGER • nfs_root_calls_enum: STRING • nfs_root_calls_pct: INTEGER • nfs_root_calls_pct_enum: STRING • nfs_lookups: INTEGER • nfs_lookups_enum: STRING • nfs_lookups_pct: INTEGER • nfs_lookups_pct_enum: STRING • nfs_read_link_calls: INTEGER • nfs_read_link_calls_enum: STRING • nfs_read_link_pct: INTEGER • nfs_read_link_pct_enum: STRING • nfs_read_calls: INTEGER • nfs_read_calls_enum: STRING • nfs_read_calls_pct: INTEGER • nfs_read_calls_pct_enum: STRING • nfs_write_cache_calls: INTEGER • nfs_write_cache_calls_enum: STRING • nfs_write_cache_calls_pct: INTEGER • nfs_write_cache_calls_pct_enum: STRING • nfs_writes: INTEGER • nfs_writes_enum: STRING • nfs_writes_pct: INTEGER • nfs_writes_pct_enum: STRING • nfs_file_creates: INTEGER • nfs_file_creates_enum: STRING • nfs_file_creates_pct: INTEGER

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_NFS_Statistics (continued)	• nfs_file_creates_pct_enum: STRING • nfs_remove_file_calls: INTEGER • nfs_remove_file_calls_enum: STRING • nfs_remove_file_calls_pct: INTEGER • nfs_remove_file_calls_pct_enum: STRING • nfs_rename_file_calls: INTEGER • nfs_rename_file_calls_enum: STRING • rename_file_calls_pct: INTEGER • rename_file_calls_pct_enum: STRING • nfs_link_calls: INTEGER • nfs_link_calls_enum: STRING • link_calls_pct: INTEGER • link_calls_pct_enum: STRING • nfs_symbolic_link_calls: INTEGER • nfs_symbolic_link_calls_enum: STRING • symbolic_link_calls_pct: INTEGER • symbolic_link_calls_pct_enum: STRING • nfs_make_directory_calls: INTEGER • nfs_make_directory_calls_enum: STRING • nfs_make_directory_calls_pct: INTEGER • nfs_make_directory_calls_pct_enum: STRING • nfs_remove_directory_calls: INTEGER • nfs_remove_directory_calls_enum: STRING • remove_directory_calls_pct: INTEGER • remove_directory_calls_pct_enum: STRING • nfs_read_directory_calls: INTEGER • nfs_read_directory_calls_enum: STRING • read_directory_calls_pct: INTEGER • read_directory_calls_pct_enum: STRING • nfs_file_system_statistics_calls: INTEGER • nfs_file_system_statistics_calls_enum: STRING • file_system_statistics_calls_pct: INTEGER • file_system_statistics_calls_pct_enum: STRING • nfs_access: INTEGER • nfs_access_enum: STRING • access_pct: INTEGER • access_pct_enum: STRING • nfs_make_node_calls: INTEGER • nfs_make_node_calls_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_NFS_Statistics (continued)	• make_node_calls_pct: INTEGER • make_node_calls_pct_enum: STRING • nfs_read_dir_calls_plus: INTEGER • nfs_read_dir_calls_plus_enum: STRING • read_dir_calls_plus_pct: INTEGER • read_dir_calls_plus_pct_enum: STRING • nfs_file_system_info: INTEGER • nfs_file_system_info_enum: STRING • file_system_info_pct: INTEGER • file_system_info_pct_enum: STRING • nfs_path_conf_calls: INTEGER • nfs_path_conf_calls_enum: STRING • path_conf_calls_pct: INTEGER • path_conf_calls_pct_enum: STRING • nfs_commit: INTEGER • nfs_commit_enum: STRING • nfs_commit_pct: INTEGER • nfs_commit_pct_enum: STRING • system_name: INTEGER • timestamp: STRING • linux_vm_id: STRING • nfs_total_calls: INTEGER • nfs_total_calls_enum: STRING
Linux_CPU_Config	ITM_Linux_CPU_Config event class with these slots: • system_name: STRING • timestamp: STRING • cpu_id: INTEGER • vendor_id: STRING • cpu_family: INTEGER • cpu_family_enum: STRING • cpu_model: INTEGER • cpu_model_enum: STRING • model_name: STRING • clock_speed: REAL • clock_speed_enum: STRING • cache_size: INTEGER • cache_size_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_OS_Config	ITM_Linux_OS_Config event class with these slots: • system_name: STRING • timestamp: STRING • os_name: STRING • os_version: STRING • gcc_version: STRING • os_vendor: STRING
Linux_File_Information	ITM_Linux_File_Information event class with these slots: • system_name: STRING • timestamp: STRING • path_u: STRING • file_name_u: STRING • size_mb: REAL • size_mb_enum: STRING • owner_u: STRING • group_u: STRING • last_changed_time: STRING • last_accessed_time: STRING • links: INTEGER • access: INTEGER • type: STRING • type_enum: STRING • link_name_u: STRING • mode: STRING • last_attr_chg_time: STRING • checksum_algorithm: INTEGER • checksum_algorithm_enum: STRING • checksum: STRING • file_content_changed: INTEGER • file_content_changed_enum: STRING
Linux_Host_Availability	ITM_Linux_Host_Availability event class with these slots: • system_name: STRING • timestamp: STRING • target_host: STRING • host_availability: INTEGER • host_availability_enum: STRING • response_time: REAL • response_time_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_File_Pattern	ITM_Linux_File_Pattern event class with these slots: • system_name: STRING • timestamp: STRING • file_name: STRING • match_pattern: STRING • match_option: INTEGER • match_option_enum: STRING • match_count: INTEGER • match_count_enum: STRING
Linux_File_Comparison	ITM_Linux_File_Comparison event class with these slots: • system_name: STRING • timestamp: STRING • file_name_1: STRING • file_name_2: STRING • file_compare_option: INTEGER • file_compare_option_enum: STRING • file_compare_result: INTEGER • file_compare_result_enum: STRING
Linux_All_Users	ITM_Linux_All_Users event class with these slots: • system_name: STRING • timestamp: STRING • name: STRING • user_id: INTEGER • user_id_enum: INTEGER • password_null: INTEGER • password_null_enum: STRING • user_duplicated: INTEGER • user_duplicated_enum: STRING • user_sessions: INTEGER • user_sessions_enum: STRING
Linux_Group	ITM_Linux_Group event class with these slots: • system_name: STRING • timestamp: STRING • group_name: STRING • group_id: INTEGER • group_id_enum: STRING • group_duplicated: INTEGER • group_duplicated_enum: STRING

Table 9. Overview of attribute groups to event classes and slots (continued)

Attribute group	event classes and slots
Linux_Machine_Information	ITM_Linux_Machine_Information event class with these slots: • system_name: STRING • timestamp: STRING • klz_hostname: STRING • klz_hostname_enum: STRING • hardware_brand: STRING • hardware_brand_enum: STRING • hardware_model: STRING • hardware_model_enum: STRING • number_of_processors_online: INTEGER • number_of_processors_online_enum: STRING • number_of_processors_configured: INTEGER • number_of_processors_configured_enum: STRING • bios_version: STRING • bios_version_enum: STRING • bios_release: STRING • bios_release_enum: STRING • machine_serial: STRING • machine_serial_enum: STRING • system_board_uuid: STRING
Linux_IP_Address	ITM_Linux_IP_Address event class with these slots: • system_name: STRING • timestamp: STRING • network_interface_name: STRING • ip_address: STRING • dns_name: STRING • dns_name_enum: STRING • ip_version: INTEGER; • ip_version_enum: STRING;

Appendix C. Monitoring Agent for Linux OS data collection

In general, the Monitoring Agent for Linux OS gathers data when requested to satisfy a workspace refresh, situation sampling of attributes, or historical data collection. All attributes in the attribute groups that make up a workspace or situation are gathered at that time. The default refresh/sampling intervals were chosen such that the agent will not put a significant load on the system as it gathers the data.

The following table shows each Linux attribute group.

Table 10. Mechanisms used to gather attributes

Attribute group	Attribute name	Collection method
KLZLOGIN	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	USRNAME	getutent API; struct utmp.ut_user
	USRPID	getutent API; struct utmp.ut_pid
	LINE	getutent API; struct utmp.ut_line
	LOGINTIME	getutent API; struct utmp.ut_tv.tv_sec
	IDLETIME	stat API on /dev/*ut_line* to get last access time & substract from current time
	FROMHOST	getutent API; struct utmp.ut_host

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZDISK	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DSKNAME	getmntent API; struct mntent.mnt_fsname
	MOUNTPT	getmntent API; struct mntent.mnt_dir
	FSTYPE	statfs API; struct statfs elements: f_blocks * (f_bsize / 1024) / 1024
	DSKSIZE	statfs API; struct statfs elements: (f_blocks * (f_bsize / 1024)) / 1024
	DSKUSED	statfs API; struct statfs elements: ((f_blocks - f_bfree) * (f_bsize / 1024)) / 1024
	DSKUSEDPCT	DSKUSED * 100.0 / (DSKUSED + DSKFREE)
	DSKFREE	statfs API; struct statfs elements: ((f_blocks - f_bfree) * (f_bsize / 1024)) / 1024
	DSKFREEPCT	100 - DSKCUSEDPCT
	INDSIZE	statfs API; struct statfs element: f_files
	INDUSED	statfs API; struct statfs elements: f_files - f_ffree
	INDFREE	statfs API; struct statfs element: f_ffree
	INDFREEPCT	100 - INDUSEDPCT
	INDUSEDPCT	INODEUSED * 100.0 / f_files

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZDU	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DSKNAME	getmntent API; struct mntent.mnt_fsname
	SPCUSED	statfs API; struct statfs elements: ((f_blocks - f_bfree) * (f_bsize / 1024)) / 1024
	SPCFREE	statfs API; struct statfs elements: (f_bavail * (f_bsize / 1024)) / 1024
	DURATE	Calculated from ″N″ and ″N - 1″ samples of SPCUSED
	HWDURATE	Larger of ″N″ and ″N - 1″ samples of DURATE
	HWTIME	Timestamp associated with the HWDURATE sample
	DUMVAVG	Average of all DURATE values
	DAYSDSK	(SPCAVAIL * 1024 * 1024) / (DUMVAVG * 24)
	DAYSCURR	(SPCAVAIL * 1024 * 1024 / (DURATE * 24)
	LWCURR	Smaller of ″N″ and ″N - 1″ samples of DAYSCURR
	DAYSPEAK	(SPCAVAIL * 1024 * 1024) / (HWDURATE * 24)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZNET	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	FNAME	Read from /proc/net/dev
	FIPADDR	socket, ioctl & inet_ntoa APIs
	FSTATUS	socket & ioctl APIs
	FMTU	socket & ioctl APIs
	FIKBYTES	Read from /proc/net/dev & divided by 1024
	RECBPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FOKBYTES	Read from /proc/net/dev & divided by 1024
	TRANSBPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FIFRAMES	Read from /proc/net/dev
	RPACKPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FOFRAMES	Read from /proc/net/dev
	TPACKPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FIERRORS	Read from /proc/net/dev
	FOERRORS	Read from /proc/net/dev
	FCOLLSNS	Read from /proc/net/dev
	FCOLLSNRT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60 / sample_interval
	FCOLLSPCT	Read from /proc/net/dev; for this sample period: (collisions / (frames sent + frames rcved)) * 100
	FIERRORT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60 / sample_interval
	FOERRORT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60 / sample_interval
	FIOERRPCT	Read from /proc/net/dev; for this sample period: (input_errors + output_errors) / (frames_sent + frames_rcved)) * 100

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZNET	FIDROP	Read from /proc/net/dev
	FODROP	Read from /proc/net/dev
	FIFOINOVR	Read from /proc/net/dev
	FIPKTFRAM	Read from /proc/net/dev
	FCARRIER	Read from /proc/net/dev
	FIERRPCT	FIOERRPCT * (FIERRORT / (FIERRORT + FOERRORT))
	FOERRPCT	FIOERRPCT - FIERRPCT
	DEVTYPE	socket & ioctl APIs
	MACADDRESS	socket & ioctl APIs
KLZCPU	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	CPUID	Read from /proc/stat
	USRCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	USRNCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	SYSCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	IDLECPU	10000 - BUSYCPU
	BUSYCPU	USRCPU + USRNCPU + SYSCPU + WAITCPU
	WAITCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	USRSYSCPU	((USRNCPU + USRCPU) * 100) / SYSCPU

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZCPUAVG	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DAYSCPU	Read from /proc/stat; total_moving_used_cpu / (previous_moving_idle - current_moving_idle); converted to days.
	CPUCURAVG	USRNCURAVG + USRCURAVG + WAITCUR + SYSCPUCUR
	CPUMOVAVG	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	USRNCURAVG	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_ sample_interval * 10000
	USRNMOVCPU	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	USRCURAVG	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	USRMOVCPU	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	SYSCPUCUR	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZCPUAVG (Continued)	SYSCPUMOV	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	IDLECUR	10000 - CPUCURAVG
	IDLEMOV	10000 - (USRNMOVCPU + USRMOVCPU + WAITMOV+ SYSCPUMOV)
	WAITCUR	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	WAITMOV	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_ the_sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZPROC	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	PID	Read from /proc; PID is the subdirectory name
	PPID	Read from /proc/PID/stat
	CMD	Read from /proc/PID/stat
	CMDLINE	Read from /proc/PID/cmdline
	STATE	Read from /proc/PID/stat
	PSYSCPU	Read from /proc/PID/stat; converted from jiffies
	PUSRCPU	Read from /proc/PID/stat; converted from jiffies
	TSYSCPU	Read from /proc/PID/stat; converted from jiffies
	TUSRCPU	Read from /proc/PID/stat; converted from jiffies
	INTPRI	Read from /proc/PID/stat
	NICE	Read from /proc/PID/stat
	SIZE	Read from /proc/PID/statm
	RSS	Read from /proc/PID/statm
	SHAREMEM	Read from /proc/PID/statm
	TRS	Read from /proc/PID/statm
	LRS	Read from /proc/PID/statm
	DRS	Read from /proc/PID/statm
	DIRTPG	Read from /proc/PID/statm
	VMSIZE	Read from /proc/PID/status
	VMLOCK	Read from /proc/PID/status
	VMDATA	Read from /proc/PID/status
	VMSTACK	Read from /proc/PID/status
	VMEXESZ	Read from /proc/PID/status
	VMLIBSZ	Read from /proc/PID/status
	CMINFLT	Read from /proc/PID/stat
	CMAJFLT	Read from /proc/PID/stat
	CPUAFF	Read from /proc/PID/stat
	USRSYSCPU	(TUSRCPU / TSYSCPU) * 100
	TBUSYCPU	TSYSCPU + TUSRCPU

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZPROC (Continued)	BUSYCPU	PSYSCPU + PUSRCPU
	VMSIZEMB	Read from /proc/PID/status; converted to MB
	VMLOCKMB	Read from /proc/PID/status; converted to MB
	VMDATAMB	Read from /proc/PID/status; converted to MB
	VMSTACKMB	Read from /proc/PID/status; converted to MB
	VMEXESZMB	Read from /proc/PID/status; converted to MB
	VMLIBSZMB	Read from /proc/PID/status; converted to MB
	PROCTHRD	Read from /proc/PID/status
	SESSIONID	Read from /proc/PID/stat
	PSYSNORM	Read from /proc/PID/stat; converted from jiffies
	PUSRNORM	Process user-mode time read from /prod/PID/stat; Nbr of CPUs read from sysconf API; (current_user_mode - previous_user_mode) / (elapsed_time * nbr_of_ cpus)
	PBUSYNORM	Process kernel-mode time read from /prod/PID/stat; Nbr of CPUs read from sysconf API; (current_kernel_mode - previous_kernel_mode) / (elapsed_time * nbr_of_ cpus)
	PROCCOUNT	Generated; count of processes with same CMDLINE

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZPUSR	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	PID	Read from /proc; PID is the subdirectory name
	RUSERID	Read from /proc/PID/status
	EUSERID	Read from /proc/PID/status
	SUSERID	Read from /proc/PID/status
	FSUSERID	Read from /proc/PID/status
	RGRPID	Read from /proc/PID/status
	EFFGRPID	Read from /proc/PID/status
	SGRPID	Read from /proc/PID/status
	FSGRPID	Read from /proc/PID/status
	RUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	EUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	SUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	FSUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	RGRP	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	EGRP	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	SGRP	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	FSGRP	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	SESSIONID	Read from /proc/PID/stat
	PPID	Read from /proc/PID/stat
	STATE	Read from /proc/PID/stat

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZPUSR (Continued)	CMD	Read from /proc/PID/stat
	CMDLINE	Read from /proc/PID/cmdline
	VMSIZEMB	Read from /proc/PID/status; converted to MB
	TTY	Read from /proc/PID/stat; converted to string by internal method

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZSYS	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	CSWSEC	Read from /proc/stat; samples_(″N″ - ″N - 1″) / sample_interval
	RATECSW	Read from /proc/stat; ((current_CSWSEC - previous_CSWSEC) / previous_CSWSEC) * 100
	PROCSEC	Read from /proc/stat; samples_(″N″ - ″N - 1″) / sample_interval
	RATEPROC	Read from /proc/stat; ((current_PROCSEC - previous_PROCSEC) / previous_PROCSEC) * 100
	CURUSRS	getutent API; count of entries in utmp database
	LOAD1MIN	Read from /proc/loadavg * 100
	LOAD5MIN	Read from /proc/loadavg * 100
	LOAD15MIN	Read from /proc/loadavg * 100
	SYSUPTIME	Read from /proc/uptime
	PGPGIN	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	PGPGINPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGPGOUT	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	PGPGOUTPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGSWAPIN	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	SWAPINPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZSYS (Continued)	PGSWAPOUT	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	SWAPOUTPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGFLTPS	Read from /proc/vmstat (2.6 kernel) samples_(″N″ - ″N - 1″) / sample_interval * 100; N/A on 2.4 kernel
	MAJFLTPS	Read from /proc/vmstat (2.6 kernel) samples_(″N″ - ″N - 1″) / sample_interval * 100; N/A on 2.4 kernel
	TOTPROCS	Count process subdirs in /proc
	ZOMBCNT	Count process subdirs in /proc in zombie state
KLZSWPRT	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	MOVSWPTOT	Read from /proc/meminfo; (last MOVSWAPTOT + SwapTotal) / 2
	SWAPUSED	Read from /proc/meminfo; (last SWAPUSED + (SwapTotal - SwapFree)) / 2
	SWPRATE	Read from /proc/meminfo; (last SWAPRATE + ((SwapTotal - SwapFree) - previous_SWAPUSED)) / 2
	SWAPDAYS	Read from /proc/meminfo; SwapTotal / (24 * SWAPRATE)
	PKSWPUSD	Read from /proc/meminfo; larger of last two (SwapTotal - SwapFree)
	MINSWPDAYS	Read from /proc/meminfo; smaller of last two SWAPDAYS
	LOWMEM	Read from /proc/meminfo; LowFree

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZVM	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SWPTOT	Read from /proc/meminfo; (SwapTotal / 1024) * 100
	SWPUSED	Read from /proc/meminfo; ((SwapTotal - SwapFree) / 1024) * 100
	SWPUSEDPCT	Read from /proc/meminfo; ((SwapTotal - SwapFree) / SwapTotal) * 100
	SWPFREE	Read from /proc/meminfo; (SwapFree / 1024) * 100
	SWPFREEPCT	100 - SWPUSEDPCT
	MEMTOT	Read from /proc/meminfo; (MemTotal / 1024) * 100
	MEMUSED	Read from /proc/meminfo; ((MemTotal - MemFree) / 1024) * 100
	MEMUSEDPCT	Read from /proc/meminfo; ((MemTotal - MemFree) / MemTotal) * 100
	MEMFREE	Read from /proc/meminfo; (MemFree / 1024) * 100
	MEMFREEPCT	100 - MEMUSEDPCT
	MEMSHARED	Read from /proc/meminfo; (MemShared / 1024) * 100
	MEMBUFF	Read from /proc/meminfo; (Buffers / 1024) * 100
	MEMCACHE	Read from /proc/meminfo; (Cache / 1024) * 100
	VSTOT	MEMTOT + SWPTOT
	VSUSED	SWPUSED + MEMUSED
	VSUSEDPCT	100 - VSFREEPCT
	VSFREE	VSTOT - VSUSED
	VSFREEPCT	(VSFREE / VSTOT) * 100
KLZSOCKS	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SCKPROTO	Read from /proc/net/sockstat
	SCKINUSE	Read from /proc/net/sockstat
	SCKHWUSED	Read from /proc/net/sockstat

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZSOCKD	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SCKPROTO	Generated TCP/UDP indicator
	RECVQ	Read from /proc/net/tcp or /proc/net/udp
	SENDQ	Read from /proc/net/tcp or /proc/net/udp
	LOCLADDR	Read from /proc/net/tcp or /proc/net/udp
	LOCLPORT	Read from /proc/net/tcp or /proc/net/udp
	LOCLSVC	Read from /proc/net/tcp or /proc/net/udp & getservbyport API; struct servent.s_name
	FORNADDR	Read from /proc/net/tcp or /proc/net/udp
	STATE	Read from /proc/net/tcp or /proc/net/udp
	SOCKUID	Read from /proc/net/tcp or /proc/net/udp
	SCKINOD	Read from /proc/net/tcp or /proc/net/udp
	REMOTPORT	Read from /proc/net/tcp or /proc/net/udp
	RUSER	Read from /proc/net/tcp or /proc/net/udp & getpwuid API; struct passed.pw_name

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZDSKIO	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	TPS	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads + writes; samples_(″N″ - ″N - 1″) / sample_interval
	BLKRDSSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors read; samples_(″N″ - ″N - 1″) / sample_interval
	BLKWRTNSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors written; samples_(″N″ - ″N - 1″) / sample_interval
	BLKSRD	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); total sectors read
	BLKSWRTN	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); total sectors written
	DEVMAJOR	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	DEVMINOR	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	DKNAME	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZIOEXT	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DKNAME	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	RDRQMSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads merged; samples_(″N″ - ″N - 1″) / sample_interval
	WRTRQMSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); writes merged; samples_(″N″ - ″N - 1″) / sample_interval
	RDRQSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads; samples_(″N″ - ″N - 1″) / sample_interval
	WRTREQSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); writes; samples_(″N″ - ″N - 1″) / sample_interval
	RDSECTSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors read; samples_(″N″ - ″N - 1″) / sample_interval
	WRSECTSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors written; samples_(″N″ - ″N - 1″) / sample_interval
	AVGRQSZ	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); (sectors_read + sectors_written) / (totals_reads + total_writes)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZIOEXT (Continued)	AVGRQQUSZ	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); I/O in progress; samples_(″N″ - ″N - 1″) / sample_interval
	AVGWAITTM	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); (time_reading + time_ writing) / (totals_reads + total_writes)
	AVGSVCTM	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); time_spent_on_I/O / (totals_reads + total_writes)
	IOUTIL	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); time_spent_on_I/O / monitoring_interval
	IUTIL	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″);IOUTIL / (total_reads / (totals_reads + total_writes))
	OUTIL	IOUTIL - OUTIL
	RDBYTESEC	RDSECTSEC converted to bytes
	WRBYTESEC	WRSECTSEC converted to bytes
	TOTBYTSEC	WRBYTESEC + RDBYTESEC

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZRPC	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	RSCALLS	Read from /proc/net/rpc/nfsd
	RSBADCALL	Read from /proc/net/rpc/nfsd
	RSBADAUTH	Read from /proc/net/rpc/nfsd
	RSBADCLT	Read from /proc/net/rpc/nfsd
	RSXDRCALL	Read from /proc/net/rpc/nfsd
	RCCALLS	Read from /proc/net/rpc/nfs
	RCRETRAN	Read from /proc/net/rpc/nfs
	RCAREF	Read from /proc/net/rpc/nfs

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZNFS	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	LOCORIG	Generated client/server indicator
	NFSVER	Generated version indicator
	NFSNULL	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	NULLPCT	NFSNULL * 100 / NFSTOT
	NFSGETATT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	GETATTPCT	NFSGETADD * 100 / NFSTOT
	NFSSETATT	
	SETATTPCT	NFSSETATT * 100 / NFSTOT
	NFSROOT	
	NFSROOTPCT	NFSROOT * 100 / NFSTOT
	NFSLOOKUP	
	LOOKUPPCT	NFSLOOKUP * 100 / NFSTOT
	NFSRDLINK	
	RDLINKPCT	NFSRDLINK * 100 / NFSTOT
	NFSREAD	
	READPCT	NFSREAD * 100 / NFSTOT
	NFSWRCACH	
	WRCACHPCT	NFSWRCACH * 100 / NFSTOT
	NFSWRITES	
	NFSWRTPCT	NFSWRITES * 100 / NFSTOT
	NFSCREATES	
	CREATESPCT	NFSCREATES * 100 / NFSTOT
	NFSREMOVE	
	REMOVEPCT	NFSREMOVE * 100 / NFSTOT
	NFSRENAME	
	RENMPCT	NFSRENAME * 100 / NFSTOT
	NFSLINK	

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
KLZNFS (Continued)	LINKPCT	NFSLINK * 100 / NFSTOT
	NFSSYMLNK	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	SYMLNKPCT	NFSSYMLNK * 100 / NFSTOT
	NFSMKDIR	
	MKDIRPCT	NFSMKDIR * 100 / NFSTOT
	NSRMDIR	
	RMDIRPCT	NFSRMDIR * 100 / NFSTOT
	NFSRDDIR	
	RDDIRPCT	NFSRDDIR * 100 / NFSTOT
	NFSFSSTAT	
	FSSTATPCT	NFSFSSTAT * 100 / NFSTOT
	NFSACCESS	
	ACCSSPCT	NFSACCESS * 100 / NFSTOT
	NFSMKNOD	
	MKNODPCT	NFSMKNOD * 100 / NFSTOT
	RDDIRPLUS	
	RDIRPLSPCT	RDDIRPLUS * 100 / NFSTOT
	NFSFSINFO	
	FSINFOPCT	NFSFSINFO * 100 / NFSTOT
	NFSPTHCONF	
	PTHCONFPCT	NFSPTHCONF * 100 / NFSTOT
	NFSCOMMIT	
	COMMITPCT	NFSCOMMIT * 100 / NFSTOT
	NFSTOT	NFSNULL + NFSGETATT + NFSSETATT + NFSROOT + NFSLOOKUP + NFSRDLINK + NFSREAD + NFSWRCACH + NFSWRITES + NFSCREATES + NFSREMOVE + NFSRENAME + NFSLINK + NFSSYMLNK + NFSMKDIR + NFSRMDIR + NFSRDDIR + NFSFSSTAT

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXLOGIN	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	USRNAME	getutent API; struct utmp.ut_user
	USRPID	getutent API; struct utmp.ut_pid
	LINE	getutent API; struct utmp.ut_line
	LOGINTIME	getutent API; struct utmp.ut_tv.tv_sec
	IDLETIME	stat API on /dev/*ut_line* to get last access time & substract from current time
	FROMHOST	getutent API; struct utmp.ut_host
	USRNAMEU	getutent API; struct utmp.ut_user
LNXDISK	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DSKNAME	getmntent API; struct mntent.mnt_fsname
	MOUNTPT	getmntent API; struct mntent.mnt_dir
	DSKSIZE	statfs API; struct statfs elements: f_blocks * (f_bsize / 1024) / 1024
	SPCUSED	statfs API; struct statfs elements: ((f_blocks - f_bfree) * (f_bsize / 1024)) / 1024
	SPCAVAIL	statfs API; struct statfs elements: (f_bavail * (f_bsize / 1024)) / 1024
	INODESIZE	statfs API; struct statfs element: f_files
	INODEUSED	statfs API; struct statfs elements: f_files - f_ffree
	INODEFREE	statfs API; struct statfs element: f_ffree
	PCTSPCUSED	SPCUSED * 100.0 / (SPCUSED + SPCAVAIL)
	PCTINDUSED	INODEUSED * 100.0 / f_files
	FSTYPE	getmntent API; struct mntent.mnt_type
	PCTSPCAV	100 - PCTSPCUSED
	MOUNTPTU	getmntent API; struct mntent.mnt_dir
	PCTINDAVAL	100 - PCTINDUSED

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXDU	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DSKNAME	getmntent API; struct mntent.mnt_fsname
	SPCUSED	statfs API; struct statfs elements: ((f_blocks - f_bfree) * (f_bsize / 1024)) / 1024
	SPCAVAIL	statfs API; struct statfs elements: (f_bavail * (f_bsize / 1024)) / 1024
	DURATE	Calculated from ″N″ and ″N - 1″ samples of SPCUSED
	HWDURATE	Larger of ″N″ and ″N - 1″ samples of DURATE
	HWTIME	Timestamp associated with the HWDURATE sample
	DUMVAVG	Average of all DURATE values
	DAYSDSK	(SPCAVAIL * 1024 * 1024) / (DUMVAVG * 24)
	DAYSCURR	(SPCAVAIL * 1024 * 1024 / (DURATE * 24)
	LWCURR	Smaller of ″N″ and ″N - 1″ samples of DAYSCURR
	DAYSPEAK	(SPCAVAIL * 1024 * 1024) / (HWDURATE * 24)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXNET	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	FNAME	Read from /proc/net/dev
	FIPADDR	socket, ioctl & inet_ntoa APIs
	FSTATUS	socket & ioctl APIs
	FMTU	socket & ioctl APIs
	FIKBYTES	Read from /proc/net/dev & divided by 1024
	RECBPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FOKBYTES	Read from /proc/net/dev & divided by 1024
	TRANSBPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FIFRAMES	Read from /proc/net/dev
	RPACKPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FOFRAMES	Read from /proc/net/dev
	TPACKPS	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) / sample_interval
	FIERRORS	Read from /proc/net/dev
	FOERRORS	Read from /proc/net/dev
	FCOLLSNS	Read from /proc/net/dev
	FCOLLSNRT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60/ sample_interval
	FCOLLSPCT	Read from /proc/net/dev; for this sample period: (collisions / (frames sent + frames rcved)) * 100
	FIERRORT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60/ sample_interval
	FOERRORT	Read from /proc/net/dev; samples_(″N″ - ″N - 1″) * 60/ sample_interval
	FIOERRPCT	Read from /proc/net/dev; for this sample period: (input_errors + output_errors) / (frames_sent + frames_rcved)) * 100

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXNET (Continued)	FIDROP	Read from /proc/net/dev
	FODROP	Read from /proc/net/dev
	FIFOINOVR	Read from /proc/net/dev
	FIFOUTOVR	Read from /proc/net/dev
	FIPKTFRAM	Read from /proc/net/dev
	FCARRIER	Read from /proc/net/dev
	FIERRPCT	FIOERRPCT * (FIERRORT / (FIERRORT + FOERRORT))
	FOERRPCT	FIOERRPCT - FIERRPCT
	DEVTYPE	socket & ioctl APIs
	MACADDRESS	socket & ioctl APIs
LNXCPU	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	CPUID	Read from /proc/stat
	USRCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	USRNCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	SYSCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	IDLECPU	10000 - BUSYCPU
	BUSYCPU	USRCPU + USRNCPU + SYSCPU + WAITCPU
	WAITCPU	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_sample _interval * 10000
	USRSYSCPU	((USRNCPU + USRCPU) * 100) / SYSCPU

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXCPUAVG	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DAYSCPU	Read from /proc/stat; total_moving_used_cpu / (previous_moving_idle - current_moving_idle); converted to days.
	CPUCURAVG	USRNCURAVG + USRCURAVG + WAITCUR + SYSCPUCUR
	CPUMOVAVG	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	USRNCURAVG	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_ sample_interval * 10000
	USRNMOVCPU	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_the _sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	USRCURAVG	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_ over_the_sample_interval * 10000
	USRMOVCPU	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_ the_sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	SYSCPUCUR	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over _the_sample_interval * 10000

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXCPUAVG (Continued)	SYSCPUMOV	Read from /proc/stat; metric_moving_average / moving_total_CPU_over_ the_sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2
	IDLECUR	10000 - CPUCURAVG
	IDLEMOV	10000 - (USRNMOVCPU + USRMOVCPU + WAITMOV+ SYSCPUMOV)
	WAITCUR	Read from /proc/stat; samples_(″N″ - ″N - 1″) / total_CPU_over_the_ sample_interval * 10000
	WAITMOV	Read from /proc/stat; metric_moving_average / moving_total_CPU_over _the_sample_interval * 10000; moving average of a metric is (previous_moving_average + samples_(″N″ - ″N - 1″)) / 2

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXPROC	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	PID	Read from /proc; PID is the subdirectory name
	PPID	Read from /proc/PID/stat
	CMD	Read from /proc/PID/stat
	STATE	Read from /proc/PID/stat
	PSYSCPU	Read from /proc/PID/stat; converted from jiffies
	PUSRCPU	Read from /proc/PID/stat; converted from jiffies
	TSYSCPU	Read from /proc/PID/stat; converted from jiffies
	TUSRCPU	Read from /proc/PID/stat
	INTPRI	Read from /proc/PID/stat
	NICE	Read from /proc/PID/statm
	SIZE	Read from /proc/PID/statm
	RSS	Read from /proc/PID/statm
	SHAREMEM	Read from /proc/PID/statm
	TRS	Read from /proc/PID/statm
	LRS	Read from /proc/PID/statm
	DRS	Read from /proc/PID/statm
	DIRTPG	Read from /proc/PID/statm
	VMSIZE	Read from /proc/PID/status
	VMLOCK	Read from /proc/PID/status
	VMDATA	Read from /proc/PID/status
	VMSTACK	Read from /proc/PID/status
	VMEXESZ	Read from /proc/PID/status
	VMLIBSZ	Read from /proc/PID/status
	CMINFLT	Read from /proc/PID/stat
	CMAJFLT	Read from /proc/PID/stat
	CMDLINE	Read from /proc/PID/cmdline
	CMDLINEU	Read from /proc/PID/cmdline
	CPUAFF	Read from /proc/PID/stat
	USRSYSCPU	(TUSRCPU / TSYSCPU) * 100
	CMDU	Read from /proc/PID/stat
	TBUSYCPU	TSYSCPU + TUSRCPU

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXPROC (Continued)	BUSYCPU	PSYSCPU + PUSRCPU
	VMSIZEMB	Read from /proc/PID/status; converted to MB
	VMLOCKMB	Read from /proc/PID/status; converted to MB
	VMDATAMB	Read from /proc/PID/status; converted to MB
	VMSTACKMB	Read from /proc/PID/status; converted to MB
	VMEXESZMB	Read from /proc/PID/status; converted to MB
	VMLIBSZMB	Read from /proc/PID/status; converted to MB
	PROCTHRD	Read from /proc/PID/status
	SESSIONID	Read from /proc/PID/stat
	PSYSNORM	Read from /proc/PID/stat; converted from jiffies
	PUSRNORM	Process user-mode time read from /prod/PID/stat; Nbr of CPUs read from sysconf API; (current_user_mode - previous_user_mode) / (elapsed_time * nbr_of_ cpus)
	PBUSYNORM	Process kernel-mode time read from /prod/PID/stat; Nbr of CPUs read from sysconf API; (current_kernel_mode - previous_kernel_mode) / (elapsed_time * nbr_of_ cpus)
	PROCCOUNT	Generated; count of processes with same CMDLINE

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXPUSR	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	PID	Read from /proc; PID is the subdirectory name
	RUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	EUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	SUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	FSUSER	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	RGRP	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	EGRP	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	SGRP	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	FSGRP	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	RUSERID	Read from /proc/PID/status
	EUSERID	Read from /proc/PID/status
	SUSERID	Read from /proc/PID/status
	FSUSRID	Read from /proc/PID/status
	RGRPID	Read from /proc/PID/status
	EFFGRPID	Read from /proc/PID/status
	SGRPID	Read from /proc/PID/status
	FSGRPID	Read from /proc/PID/status

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXPUSR (Continued)	RUSERU	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	EUSERU	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	SUSERU	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	FSUSERU	Read from /proc/PID/status; converted to string using the getpwuid API; struct passwd.pw_name
	RGRPU	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	EGRPU	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	SGRPU	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	FSGRPU	Read from /proc/PID/status; converted to string using the getgrgid API; struct group.gr_name
	SESSIONID	Read from /proc/PID/stat
	PPID	Read from /proc/PID/stat
	STATE	Read from /proc/PID/stat
	CMDLINEU	Read from /proc/PID/cmdline
	CMDU	Read from /proc/PID/stat
	VMSIZEMB	Read from /proc/PID/status; converted to MB
	TTY	Read from /proc/PID/stat; converted to string by internal method

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXSYS	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	CSWSEC	Read from /proc/stat; samples_(″N″ - ″N - 1″) / sample_interval
	RATECSW	Read from /proc/stat; ((current_CSWSEC - previous_CSWSEC) / previous_CSWSEC) * 100
	PROCSEC	Read from /proc/stat; samples_(″N″ - ″N - 1″) / sample_interval
	RATEPROC	Read from /proc/stat; ((current_PROCSEC - previous_PROCSEC) / previous_PROCSEC) * 100
	CURUSRS	getutent API; count of entries in utmp database
	LOAD1MIN	Read from /proc/loadavg * 100
	LOAD5MIN	Read from /proc/loadavg * 100
	LOAD15MIN	Read from /proc/loadavg * 100
	SYSUPTIME	Read from /proc/uptime
	PGPGIN	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	PGPGINPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGPGOUT	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	PGPGOUTPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGSWAPIN	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	SWAPINPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXSYS (Continued)	PGSWAPOUT	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel)
	SWAPOUTPS	Read from /proc/vmstat (2.6 kernel) or /proc/stat (2.4 kernel); samples_(″N″ - ″N - 1″) / sample_interval * 100
	PGFLTPS	Read from /proc/vmstat (2.6 kernel) samples_(″N″ - ″N - 1″) / sample_interval * 100; N/A on 2.4 kernel
	MAJFLTPS	Read from /proc/vmstat (2.6 kernel) samples_(″N″ - ″N - 1″) / sample_interval * 100; N/A on 2.4 kernel
	TOTPROCS	Count process subdirs in /proc
	ZOMBCNT	Count process subdirs in /proc in zombie state
LNXSWPRT	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	MOVSWPTOT	Read from /proc/meminfo; (last MOVSWAPTOT + SwapTotal) / 2
	SWAPUSED	Read from /proc/meminfo; (last SWAPUSED + (SwapTotal - SwapFree)) / 2
	SWPRATE	Read from /proc/meminfo; (last SWAPRATE + ((SwapTotal - SwapFree) - previous_SWAPUSED)) / 2
	SWAPDAYS	Read from /proc/meminfo; SwapTotal / (24 * SWAPRATE)
	PKSWPUSD	Read from /proc/meminfo; larger of last two (SwapTotal - SwapFree)
	MINSWPDAYS	Read from /proc/meminfo; smaller of last two SWAPDAYS
	LOWMEM	Read from /proc/meminfo; LowFree

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXVM	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SWAPTOT	Read from /proc/meminfo; (SwapTotal / 1024) * 100
	SWAPUSED	Read from /proc/meminfo; ((SwapTotal - SwapFree) / 1024) * 100
	SWAPFREE	Read from /proc/meminfo; (SwapFree / 1024) * 100
	MEMTOT	Read from /proc/meminfo; (MemTotal / 1024) * 100
	MEMUSED	Read from /proc/meminfo; ((MemTotal - MemFree) / 1024) * 100
	MEMFREE	Read from /proc/meminfo; (MemFree / 1024) * 100
	MEMSHARED	Read from /proc/meminfo; (MemShared / 1024) * 100
	MEMBUFF	Read from /proc/meminfo; (Buffers / 1024) * 100
	MEMCACHE	Read from /proc/meminfo; (Cache / 1024) * 100
	VSTOT	MEMTOT + SWAPTOT
	USEDVS	SWAPUSED + MEMUSED
	AVAILVS	VSTOT - USEDVS
	AVALVSPCT	(AVAILVS / VSTOT) * 100
	USEDVSPCT	100 - USEDSWPPCT
	USEDRLPCT	Read from /proc/meminfo; ((MemTotal - MemFree) / MemTotal) * 100
	AVALRLPCT	100 - USEDSWPPCT
	USEDSWPPCT	Read from /proc/meminfo; ((SwapTotal - SwapFree) / SwapTotal) * 100
	AVALSWPPCT	100 - USEDSWPPCT
LNXSOCKS	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SCKPROTO	Read from /proc/net/sockstat
	SCKINUSE	Read from /proc/net/sockstat
	SCKHWUSED	Read from /proc/net/sockstat

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXSOCKD	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	SCKPROTO	Generated TCP/UDP indicator
	RECVQ	Read from /proc/net/tcp or /proc/net/udp
	SENDQ	Read from /proc/net/tcp or /proc/net/udp
	LOCLADDR	Read from /proc/net/tcp or /proc/net/udp
	LOCLPORT	Read from /proc/net/tcp or /proc/net/udp
	LOCLSVC	Read from /proc/net/tcp or /proc/net/udp & getservbyport API; struct servent.s_name
	FORNADDR	Read from /proc/net/tcp or /proc/net/udp
	STATE	Read from /proc/net/tcp or /proc/net/udp
	SOCKUID	Read from /proc/net/tcp or /proc/net/udp
	SCKINOD	Read from /proc/net/tcp or /proc/net/udp
	REMOTPORT	Read from /proc/net/tcp or /proc/net/udp
	RUSERU	Read from /proc/net/tcp or /proc/net/udp & getpwuid API; struct passed.pw_name

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXDSKIO	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	TPS	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads + writes; samples_(″N″ - ″N - 1″) / sample_interval
	BLKRDSSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors read; samples_(″N″ - ″N - 1″) / sample_interval
	BLKWRTNSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors written; samples_(″N″ - ″N - 1″) / sample_interval
	BLKSRD	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); total sectors read
	BLKSWRTN	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); total sectors written
	DEVMAJOR	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	DEVMINOR	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	DKNAME	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXIOEXT	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	DKNAME	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel)
	RDRQMSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads merged; samples_(″N″ - ″N - 1″) / sample_interval
	WRTRQMSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); writes merged; samples_(″N″ - ″N - 1″) / sample_interval
	RDRQSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); reads; samples_(″N″ - ″N - 1″) / sample_interval
	WRTREQSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); writes; samples_(″N″ - ″N - 1″) / sample_interval
	RDSECTSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors read; samples_(″N″ - ″N - 1″) / sample_interval
	WRSECTSEC	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); sectors written; samples_(″N″ - ″N - 1″) / sample_interval
	AVGRQSZ	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); (sectors_read + sectors_written) / (totals_reads + total_writes)

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXIOEXT (Continued)	AVGRQQUSZ	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); I/O in progress; samples_(″N″ - ″N - 1″) / sample_interval
	AVGWAITTM	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); (time_reading + time_ writing) / (totals_reads + total_writes)
	AVGSVCTM	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); time_spent_on_I/O / (totals_reads + total_writes)
	IOUTIL	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″); time_spent_on_I/O / monitoring_interval
	IUTIL	Read from /proc/diskstats (2.6 kernel) or /proc/partitions (2.4 kernel); samples_(″N″ - ″N - 1″);IOUTIL / (total_reads / (totals_reads + total_writes))
	OUTIL	IOUTIL - OUTIL
	RDBYTESEC	RDSECTSEC converted to bytes
	WRBYTESEC	WRSECTSEC converted to bytes
	TOTBYTSEC	WRBYTESEC + RDBYTESEC

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXRPC	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	RSCALLS	Read from /proc/net/rpc/nfsd
	RSBADCALL	Read from /proc/net/rpc/nfsd
	RSBADAUTH	Read from /proc/net/rpc/nfsd
	RSBADCLT	Read from /proc/net/rpc/nfsd
	RSXDRCALL	Read from /proc/net/rpc/nfsd
	RCCALLS	Read from /proc/net/rpc/nfs
	RCRETRAN	Read from /proc/net/rpc/nfs
	RCAREF	Read from /proc/net/rpc/nfs

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXNFS	LOCORIG	Generated client/server indicator
	NFSVER	Generated version indicator
	NFSNULL	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	NULLPCT	NFSNULL * 100 / NFSTOT
	NFSGETATT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	GETATTPCT	NFSGETADD * 100 / NFSTOT
	NFSSETATT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	SETATTPCT	NFSSETATT * 100 / NFSTOT
	NFSROOT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	NFSROOTPCT	NFSROOT * 100 / NFSTOT
	NFSLOOKUP	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	LOOKUPPCT	NFSLOOKUP * 100 / NFSTOT
	NFSRDLINK	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	RDLINKPCT	NFSRDLINK * 100 / NFSTOT
	NFSREAD	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	READPCT	NFSREAD * 100 / NFSTOT
	NFSWRCACH	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	WRCACHPCT	NFSWRCACH * 100 / NFSTOT
	NFSWRITES	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	NFSWRTPCT	NFSWRITES * 100 / NFSTOT

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXNFS (Continued)	NFSCREATES	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	CREATESPCT	NFSCREATES * 100 / NFSTOT
	NFSREMOVE	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	REMOVEPCT	NFSREMOVE * 100 / NFSTOT
	NFSRENAME	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	RENMPCT	NFSRENAME * 100 / NFSTOT
	NFSLINK	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	LINKPCT	NFSLINK * 100 / NFSTOT
	NFSSYMLNK	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	SYMLNKPCT	NFSSYMLNK * 100 / NFSTOT
	NFSMKDIR	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	MKDIRPCT	NFSMKDIR * 100 / NFSTOT
	NSRMDIR	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	RMDIRPCT	NFSRMDIR * 100 / NFSTOT
	NFSRDDIR	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	RDDIRPCT	NFSRDDIR * 100 / NFSTOT
	NFSFSSTAT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	FSSTATPCT	NFSFSSTAT * 100 / NFSTOT
	NFSACCESS	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	ACCSSPCT	NFSACCESS * 100 / NFSTOT

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXNFS (Continued)	NFSMKNOD	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	MKNODPCT	NFSMKNOD * 100 / NFSTOT
	RDDIRPLUS	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	RDIRPLSPCT	RDDIRPLUS * 100 / NFSTOT
	NFSFSINFO	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	FSINFOPCT	NFSFSINFO * 100 / NFSTOT
	NFSPTHCONF	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	PTHCONFPCT	NFSPTHCONF * 100 / NFSTOT
	NFSCOMMIT	Read from /proc/net/rpc/nfs (client) or /proc/net/rpc/nfsd (server)
	COMMITPCT	NFSCOMMIT * 100 / NFSTOT
	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	NFSTOT	″NFSNULL + NFSGETATT + NFSSETATT + NFSROOT + NFSLOOKUP + NFSRDLINK + NFSREAD + NFSWRCACH + NFSWRITES + NFSCREATES + NFSREMOVE + NFSRENAME + NFSLINK + NFSSYMLNK + NFSMKDIR + NFSRMDIR + NFSRDDIR + NFSFSSTAT″
LNXCPUCON	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	CPUID	Read from /proc/cpuinfo
	VENDID	Read from /proc/cpuinfo
	CPUFAMILY	Read from /proc/cpuinfo
	CPUMODEL	Read from /proc/cpuinfo
	MODELNM	Read from /proc/cpuinfo
	CPUCLK	Read from /proc/cpuinfo
	CACHESZ	Read from /proc/cpuinfo

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXOSCON	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	OSNAME	Read from /proc/version
	OSVER	Read from /proc/version
	GCCVER	Read from /proc/version
	OSVEND	Read from /proc/version
LNXFILE	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	PATHU	stat64, opendir and readdir APIs
	FILEU	stat64, opendir and readdir APIs
	SIZEMB	lstat64 API; struct stat64.st_size / (1024 * 1024) * 1000
	OWNERU	lstat64 API; struct stat64.st_uid & getpwuid API - struct passed.pw_name
	GROUPU	lstat64 API; struct stat64.st_gid & getgrgid API - struct group.gr_name
	CHANGEDTM	lstat64 API; struct stat64.st_mtime
	ACCESSEDTM	lstat64 API; struct stat64.st_atime
	LINKS	lstat64 API; struct stat64.st_nlinks
	ACCESS	lstat64 API; struct stat64.st_mode
	TYPE	lstat64 API; struct stat64.st_mode
	LINKNAMEU	readlink API
	MODE	lstat64 API; struct stat64.st_mode
	STATUSTM	lstat64 API; struct stat64.st_ctime
	HASHALGO	Passed to agent as parameter
	HASHSUM	CRC32, MD5 or SHA1 calculation - internal functions
	FCCHANGED	Generated; true if HASHSUM has changed since last monitoring interval
	SIZEMB64	lstat64 API; struct stat64.st_size / (1024 * 1024) * 1000

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXPING	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	TGTSERV	Passed to agent as situation parameter or from KLZ_PINGHOSTLIST file
	SERVUP	Result from /bin/ping command
	HOSTRESP	Result from /bin/ping command
LNXFILPAT	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	FILENAME	Passed to agent as parameter
	MATCHPAT	Passed to agent as parameter
	MATCHOPT	Passed to agent as parameter
	MATCHCNT	Result from grep cmd
LNXFILCMP	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	FILENAME1	Passed to agent as parameter
	FILENAME2	Passed to agent as parameter
	COMPOPT	Passed to agent as parameter
	COMPRESULT	Result from /usr/bin/cmp or /usr/bin/diff commands
LNXALLUSR	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	USERNAME	getpwent API; struct passwd.pw_name
	UID	getpwent API; struct passwd.pw_uid
	PWNULL	getspnam API; generated; true if struct spwd.sp_pwdp is empty
	USERDUP	Generated; true if duplicate USERNAME or UID is detected
	USERSES	getpwent & getutxent APIs; generated; matches of struct paswd.pw_name & struct utmpx.ut_user fields
	UID64	getpwent API; struct passwd.pw_uid

Table 10. Mechanisms used to gather attributes (continued)

Attribute group	Attribute name	Collection method
LNXGROUP	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	GRPNAME	getgroupent API; struct group.gr_name
	GRPID	getgroupent API; struct group.gr_gid
	GRPDUP	Generated; true if duplicate GRPNAME or GRPID is detected
	GRPID64	
LNXMACHIN	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	HOSTNAME	gethostname API
	BRAND	dmidecode program, where supported; hardcoded otherwise
	MODEL	dmidecode program, where supported; hardcoded otherwise
	ONLNCPU	sysconf API
	CONFCPU	sysconf API
	BIOSVER	dmidecode program, where supported; hardcoded otherwise
	BIOSREL	dmidecode program, where supported; hardcoded otherwise
	MACSERIAL	dmidecode program, where supported; hardcoded otherwise
	UUID	dmidecode program, where supported; hardcoded otherwise
LNXIPADDR	ORIGINNODE	Short host name + ″:LZ″
	TIMESTAMP	Current time
	INTFNAME	Read from /proc/net/dev
	IPADDRESS	IPv4: socket, ioctl & inet_ntoa APIs. IPv6: read from /proc/net/if_inet6
	DNSNAME	getaddrinfo and getnameinfo APIs
	IPVERSION	Harcoded based on IP type

Appendix D. Troubleshooting

This appendix explains how to troubleshoot the IBM Tivoli Monitoring: Linux OS Agent. Troubleshooting, or problem determination, is the process of determining why a certain product is malfunctioning.

Note: You can resolve some problems by ensuring that your system matches the system requirements listed in Chapter 2, "Requirements for the monitoring agent," on page 5.

This appendix provides agent-specific troubleshooting information. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information. Also see "Support information" on page 231 for other problem-solving options.

Gathering product information for IBM Software Support

Before contacting IBM Software Support about a problem you are experiencing with this product, gather the following information that relates to the problem:

Table 11. Information to gather before contacting IBM Software Support

Information type	Description
Log files	Collect trace log files from failing systems. Most logs are located in a `logs` subdirectory on the host computer. See "Trace logging" on page 214 for lists of all trace log files and their locations. See the *IBM Tivoli Monitoring User's Guide* for general information about the IBM Tivoli Monitoring environment.
Linux information	• Version number and patch level • Sample application data file (if monitoring a file)
Operating system	Operating system version number and patch level
Messages	Messages and other information displayed on the screen
Version numbers for IBM Tivoli Monitoring	Version number of the following members of the monitoring environment: • IBM Tivoli Monitoring. Also provide the patch level, if available. • IBM Tivoli Monitoring: Linux OS Agent
Screen captures	Screen captures of incorrect output, if any.
Core dump files	If the system stops on UNIX or Linux systems, collect core dump file from *install_dir*/bin directory, where *install_dir* is the directory path where you installed the monitoring agent.

Built-in troubleshooting features

The primary troubleshooting feature in the IBM Tivoli Monitoring: Linux OS Agent is logging. *Logging* refers to the text messages and trace data generated by the IBM Tivoli Monitoring: Linux OS Agent. Messages and trace data are sent to a file.

Trace data captures transient information about the current operating environment when a component or application fails to operate as designed. IBM Software Support personnel use the captured trace information to determine the source of an error or unexpected condition. See "Trace logging" on page 214 for more information.

Problem classification

The following types of problems might occur with the IBM Tivoli Monitoring: Linux OS Agent:

- Installation and configuration
- General usage and operation
- Display of monitoring data
- Take Action commands

This appendix provides symptom descriptions and detailed workarounds for these problems, as well as describing the logging capabilities of the monitoring agent. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information.

Trace logging

Trace logs capture information about the operating environment when component software fails to operate as intended. The principal log type is the RAS (Reliability, Availability, and Serviceability) trace log. These logs are in the English language only. The RAS trace log mechanism is available for all components of IBM Tivoli Monitoring. Most logs are located in a `logs` subdirectory on the host computer. See the following sections to learn how to configure and use trace logging:

- "Principal trace log files" on page 215
- "Examples: using trace logs" on page 216
- "Setting RAS trace parameters" on page 217

Note: The documentation refers to the RAS facility in IBM Tivoli Monitoring as "RAS1".

IBM Software Support uses the information captured by trace logging to trace a problem to its source or to determine why an error occurred. The default configuration for trace logging, such as whether trace logging is enabled or disabled and trace level, depends on the source of the trace logging. Trace logging is always enabled.

Log file management is described in the following table:

Table 12. Log file management on UNIX compared to log file management on Windows

Location of logs	Description
• On a Windows monitoring server • On a Windows computer where the monitoring agent is running • On a UNIX or Linux computer where the monitoring agent is running	On Windows, the log file is overwritten each time the component starts. There is no automated method to archive previous RAS1 log files. **Note:** To prevent the log files from consuming too much disk space, you can stop and start the component. This action automatically creates a new log file. Save a backup of log files if your company policy requires archiving of log files.

Table 12. Log file management on UNIX compared to log file management on Windows (continued)

Location of logs	Description

• On a UNIX or Linux monitoring server
• On a UNIX or Linux computer where the monitoring agent is running
	On UNIX or Linux systems, because of the use of the &Timestamp; variable in the log file names, multiple RAS1 logs are normally stored the `logs` subdirectory. The file name for a trace log is a copy of a related file that includes the process ID of the agent. The two files have the same timestamp as in these examples from a computer with a host name **f50pa2b**. The **1112097194** part of the name is the timestamp:
	`f50pa2b_lz_1112097194.log`
	`f50pa2b_lz_1112097194.pid60420`
	
	where *lz* is the unique, two-character code for Monitoring Agent for Linux OS.

Note: When you communicate with IBM Software Support, you must capture and send the RAS1 log that matches any problem occurrence that you report. Table 13 can help you identify files that are relevant to your troubleshooting efforts.

Principal trace log files

Table 13 contains locations, file names, and descriptions of trace logs that can help determine the source of problems with agents.

Table 13. Trace log files for troubleshooting agents

System where log is located	File name and path	Description
On the computer that hosts the monitoring agent	The *hostname*`_lz_`*instance*`.log` file is located in the *install_dir*`/logs` path.	Traces activity of the monitoring agent.
	The `*.LG0` file is located in the following subdirectory of the *install_dir* path: `/logs`.	Shows whether agent was able to connect to the monitoring server. Shows which situations are started and stopped, and shows other events while the agent is running. A new version of this file is generated every time the agent is restarted. IBM Tivoli Monitoring generates one backup copy of the `*.LG0` file with the tag `.LG1`. View `.LG1` to learn the following details regarding the *previous* monitoring session: • Status of connectivity with the monitoring server. • Situations that were running. • The success or failure status of Take Action commands.
	The *take_action_name*`.log` file (where *take_action_name* is the name of the Take Action command) is located in the `/logs` subdirectory of the *install_dir* path.	Traces activity each time a Take Action command runs. For example, when a hypothetical **start_command** Take Action command runs, IBM Tivoli Monitoring would generate a `start_command.log` file.

Table 13. Trace log files for troubleshooting agents (continued)

System where log is located	File name and path	Description
On the Tivoli Enterprise Monitoring Server	The `candle_installation.log` file in the *`install_dir`*`/logs` path.	Provides details about products that are installed. **Note:** Trace logging is enabled by default. A configuration step is not required to enable this tracing.
	The `Warehouse_Configuration.log` file is located in the following path on Windows: *install_dir*\InstallITM.	Provides details about the configuration of data warehousing for historical reporting.
	The name of the RAS log file is as follows: • **On Windows:** *install_dir*\logs\ *`hostname`*`_ms_`*`timestamp`*`.log` • **On UNIX or Linux:** *`hostname`*`_ms_`*`timestamp`*`.log` and *`hostname`*`_ms_`*`timestamp`*`.pid`*`nnnnn`* in the *install_dir/logs* path, where *nnnnn* is the process ID number.	Traces activity on the monitoring server.
On the Tivoli Enterprise Portal Server	The name of the RAS log file is as follows: • **On Windows:** *install_dir*\logs\ *hostname*_cq_*timestamp*.log • **On UNIX or Linux:** *`hostname`*`_cq_`*`timestamp`*`.log` and *`hostname`*`_cq_`*`timestamp`*`.pid`*`nnnnn`* in the *install_dir/logs* path, where *nnnnn* is the process ID number.	Traces activity on the portal server.
	The `TEPS_ODBC.log` file is located in the following path on Windows: *`install_dir`*`\InstallITM`.	When you enable historical reporting, this log file traces the status of the warehouse proxy agent.
Definitions of variables: *timestamp* is timestamp whose format includes year (y), month (m), day (d), hour (h), and minute (m), as follows: **yyyymmdd hhmm** *install_dir* represents the directory path where you installed the IBM Tivoli Monitoring component. *install_dir* can represent a path on the computer that host the monitoring system, the monitoring agent, or the portal. *instance* refers to the name of the database instance that you are monitoring. *hostname* refers to the name of the computer on which the IBM Tivoli Monitoring component runs.		

See the *IBM Tivoli Monitoring Installation and Setup Guide* for more information on the complete set of trace logs that are maintained on the monitoring server.

Examples: using trace logs

Typically IBM Software Support applies specialized knowledge to analyze trace logs to determine the source of problems. However, you can open trace logs in a text editor to learn some basic facts about your IBM Tivoli Monitoring environment.

Example one

This excerpt shows the typical log for a failed connection between a monitoring agent and a monitoring server with the host name **server1a**:

```
(Thursday, August 11, 2005, 08:21:30-{94C}kdcl0cl.c,105,"KDCL0_ClientLookup") status=1c020006,
    "location server unavailable", ncs/KDC1_STC_SERVER_UNAVAILABLE
(Thursday, August 11, 2005, 08:21:35-{94C}kraarreg.cpp,1157,"LookupProxy") Unable to connect to
    broker at ip.pipe:: status=0, "success", ncs/KDC1_STC_OK
(Thursday, August 11, 2005, 08:21:35-{94C}kraarreg.cpp,1402,"FindProxyUsingLocalLookup") Unable
    to find running CMS on CT_CMSLIST <IP.PIPE:#server1a>
```

Example two

The following excerpts from the trace log *for the monitoring server* show the status of an agent, identified here as "Remote node." The name of the computer where the agent is running is **SERVER5B**:

```
(42C039F9.0000-6A4:kpxreqhb.cpp,649,"HeartbeatInserter") Remote node SERVER5B:LZ is ON-LINE.
. . .
(42C3079B.0000-6A4:kpxreqhb.cpp,644,"HeartbeatInserter") Remote node SERVER5B:KLZ is OFF-LINE.
```

Key points regarding the preceding excerpt:

- The monitoring server appends the **LZ** product code to the server name to form a unique name (`SERVER5B:LZ`) for this instance of Monitoring Agent for Linux OS. This unique name enables you to distinguish multiple monitoring products that might be running on **SERVER5B**.
- The log shows when the agent started (ON-LINE) and later stopped (OFF-LINE) in the environment.
- For the sake of brevity an ellipsis (...) represents the series of trace log entries that were generated while the agent was running.
- Between the ON-LINE and OFF-LINE log entries, the agent was communicating with the monitoring server.
- The ON-LINE and OFF-LINE log entries are always available in the trace log. All trace levels that are described in "Setting RAS trace parameters" provide these entries.

Setting RAS trace parameters

Objective

Pinpoint a problem by setting detailed tracing of individual components of the monitoring agent and modules.

Background Information

Monitoring Agent for Linux OS uses RAS1 tracing and generates the logs described in Table 13 on page 215. The default RAS1 trace level is ERROR.

Before you begin

When you are troubleshooting, follow these guidelines to ensure that you capture and analyze the correct log files: Because of the use of the &Timestamp; variable in the log file names on UNIX or Linux systems, there are typically multiple RAS1 logs in the `logs` subdirectory. When you forward log files to IBM Software Support, you must send the RAS1 log that matches the problem occurrence that the log files are reporting.

After you finish

On UNIX or Linux, periodically prune the trace logs in the `logs` subdirectory so that there is available disk space for new logging.

Note: The **KDC_DEBUG** setting and the Maximum error tracing setting can generate a large amount of trace logging. Use them only temporarily, while you are troubleshooting problems. Otherwise, the logs can occupy excessive amounts of hard disk space.

Procedure

Specify RAS1 trace options in the *install_dir*`/config/lz.ini` file. The basic format for setting tracing options is as follows:

```
KBB_RAS1=ERROR (UNIT:klz options)
```

Use one of the following methods to modify trace options:

- **Manually edit the configuration file to set trace logging**
  1. Open the trace options file: */install_dir*/config/lz.ini.
  2. Edit the line that begins with **KBB_RAS1=** to set trace logging preferences.

     For example, if you want detailed trace logging, set the Maximum Tracing option:

     ```
     export KBB_RAS1='ERROR (UNIT:klz ALL) (UNIT:kra ALL)'
     ```
  3. Restart the monitoring agent so that your changes take effect.

Problems and workarounds

The following sections provide symptoms and workarounds for problems that might occur with Monitoring Agent for Linux OS:

Note: You can resolve some problems by ensuring that your system matches the system requirements listed in Chapter 2, "Requirements for the monitoring agent," on page 5.

This appendix provides agent-specific troubleshooting information. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information.

Installation and configuration troubleshooting

This section provides tables that show solutions for installation, configuration, and uninstallation problems.

Agent upgrade and restart using non-root

The monitoring agent can run using a non-root user ID on UNIX and Linux systems. This can be done by running the **itmcmd agent start** command while logged in as a non-root user, and this can be done remotely by deploying the agent using the **Run As** option on the GUI or using the **_UNIX_STARTUP_.Username** option on the **tacmd addSystem** command line. If the agent is running using a non-root user ID, and then the agent is upgraded, restarted remotely, restarted as a result of a system reboot, or the **itmcmd agent start** is run using the root user ID, then the monitoring agent subsequently runs as the root user. To confirm the user ID that the monitoring agent is using, run the following command:

```
itm_install/bin/cinfo -r
```

If the agent is using root, and that is not the desired user ID, then use the following steps to restart the agent:

1. Log in as root.
2. Run the **itmcmd agent stop** command.
3. Log in (or 'su') to the user ID that you want the agent to run as.
4. Run the **itmcmd agent start** command.

If the agent was running as root because of a system reboot, then edit the startup file using the following steps so that the appropriate user ID is used the next time the system is rebooted:

1. Look at *install_dir*/registry/AutoStart, and get *NUM*.
2. Edit the autostart for your operating system:

   The location of the startup file is platform dependent as follows:
   - AIX®: /etc/rc.itm*NUM*
   - HP-UX: /sbin/init.d/ITMAgents*NUM*
   - Linux: /etc/init.d/ITMAgents*NUM*
   - Solaris: /etc/init.d/ITMAgents*NUM*
3. Add entries for your operating system using the following command:

   ```
   /usr/bin/su - instancename
   -c "install_dir/bin/itmcmd agent
   -h install_dir
   -o instancename
   start product_code"
   ```

   Where:

   instancename
   Name of the instance

   install_dir
   Name of the directory

   product_code
   2-character product code for the agent, for example, lz for the Monitoring Agent for Linux OS

   Examples:
   - For AIX, add entries with the following format:

     ```
     su - USER -c " /opt/IBM/ITM/bin/itmcmd agent
     -o INSTANCE start lz"
     ```

     Where:

     USER Name of the user

     INSTANCE
     Name of the instance
   - For Linux, HP_UX, and Solaris, add entries with the following format:

     ```
     /bin/su - USER -c " /opt/IBM/ITM/bin/itmcmd agent
     -o INSTANCE start lz >/dev/null 2>&1"
     ```

     Where:

     USER Name of the user

     INSTANCE
     Name of the instance
4. Repeat Steps 1 through 3 for all occurrences of stop.
5. Save the file.

Table 14. Problems and solutions for installation and configuration

Problem	Solution
When you upgrade to IBM Tivoli Monitoring, you might need to apply fixpacks to Candle®, Version 350, agents.	Fixpacks for Candle, Version 350, are delivered as each monitoring agent is upgraded to IBM Tivoli Monitoring. **Note:** The IBM Tivoli Monitoring download image or CD provides application fixpacks for the monitoring agents that are installed from that CD (for example, the agents for operating systems such as Windows, Linux, UNIX, and i5/OS®). The upgrade software for other agents is located on the download image or CDs for that specific monitoring agent, such as the agents for database applications. If you do not upgrade the monitoring agent to IBM Tivoli Monitoring, the agent continues to work. However, you must upgrade to have all the functionality that IBM Tivoli Monitoring offers.
install.sh fails with a JVMJ9VM011W error.	The SELINUX parameter in the /etc/sysconfig/selinux file must be set to ″disable″. Then, reboot the system.
Presentation files and customized OMEGAMON® screens for Candle monitoring agents need to be upgraded to a new Linux on z/Series system.	The upgrade from version 350 to IBM Tivoli Monitoring handles export of the presentation files and the customized OMEGAMON screens.
Installation of Monitoring Agent for Linux OS on the Linux S390 R2.6 64-bit operating system fails with a message similar to the following: `LINUX MONITORING AGENT V610R`*nnn* `unable to install agent`, where *nnn* is the release number.	Solve this problem as follows: 1. Run the following command before running any installation or configuration command for the agent: `export JAVA_COMPILER=NONE` 2. Install the following two RPM (Red Hat Package Manager) files: • **compat-libstdc++-295-2.....s390x.rpm** • **compat-libstdc++-33-3.....s390x.rpm** It requires the two s390x.rpm files, in addition to the s390.rpm files. You can obtain the required RPM files from the CD for Red Hat As 4.0 s390x.
During a command-line installation, you choose to install a component that is already installed, and you see the following warning: `WARNING - you are about to install the SAME version of "`*component*`"` where *component* is the name of the component that you are attempting to install. **Note:** This problem affects UNIX command-line installations. If you monitor only Windows environments, you would see this problem if you choose to install a product component (for example, a monitoring server) on UNIX.	You must exit and restart the installation process. You cannot return to the list where you selected components to install. When you run the installer again, do not attempt to install any component that is already installed.
The product fails to do a monitoring activity that requires read, write, or execute permissions. For example, the product might fail to run a Take Action command or read a log.	The monitoring agent must have the permissions necessary to perform requested actions. For example, if the user ID you used to log onto the system to install the monitoring agent (locally or remotely) does not have the permission to perform a monitoring operation (such as running a command), the monitoring agent is not able perform the operation.

Table 14. Problems and solutions for installation and configuration (continued)

Problem	Solution
While installing the agent from a CD, the following message is displayed and you are not able to continue the installation: `install.sh warning: unarchive of "/cdrom/unix/cienv1.tar" may have failed`	This error is caused by low disk space. Although the install.sh script indicates that it is ready to install the agent software, the script considers the size of *all* tar files, not the size of all the files that are contained within the tar file.Run the **df -k** command to check whether the file systems have enough space to install agents.
Cannot locate the **KDCB0_HOSTNAME** setting.	Go to *install_dir*/config and edit the corresponding **.ini** file. Set the **KDCB0_HOSTNAME** parameter followed by the IP address. If you use multiple network interface cards (NICs), give the Primary IP address of the network interface.
The Monitoring Agent for Linux OS repeatedly restarts.	You can collect data to analyze this problem as follows: 1. Access the *install_dir*/config/lz.ini file, which is described in "Setting RAS trace parameters" on page 217. 2. Add the following line: `KBB_SIG1=trace –dumpoff`
Agents in the monitoring environment use different communication protocols. For example, some agents have security enabled and others do not.	Configure both the monitoring server and the Warehouse proxy server to accept multiple protocols, as described in the *IBM Tivoli Monitoring Installation and Setup Guide*.
Creating a firewall partition file: The partition file enables an agent to connect to the monitoring server through a firewall.	**How it works:** When the agents start, they search KDCPARTITION.TXT for the following matches: • An entry that matches the partition name **OUTSIDE**. • An entry that also includes a valid external address. For more information, see the *IBM Tivoli Monitoring Installation and Setup Guide*.
You see the following error: `Hub not registered with location broker. Error-code 1195.`	Confirm that the password within the Tivoli Enterprise Monitoring Server is correct.

Table 14. Problems and solutions for installation and configuration (continued)

Problem	Solution

The Monitoring Agent for Linux OS is started and running but not displaying data in the Tivoli Enterprise Portal.	Perform the following steps:
	1. Open the Manage Tivoli Enterprise Monitoring Services window.
	2. Right-click the name of the monitoring server.
	3. Select **Advanced > Add TEMS Application Support** in the pop-up menu. Add application support if any for any agent that is missing from the list. See in IBM Tivoli Monitoring Installation and Setup Guide for more information on adding application support.
	4. Check the log files to see whether there are connection problems.
	5. If there are no connection problems, check whether the agent has terminated.
	6. If the agent is not terminated, confirm that you have added application support for the Monitoring Agent for Linux OS in the Tivoli Enterprise Monitoring Server as follows:
	• Verify that the following entries are available in the *install_dir*\candle_installation.log file:*install_dir*\Install\IBM Tivoli Monitoring *timestamp*.log
	`... Browser Client support for ITM Agent for Linux`
	`... Desktop Client support for ITM Agent for Linux`
	If the **candle_installation.log file** does not have the above entries for Monitoring Agent for Linux OS, add application support for this monitoring agent. See in IBM Tivoli Monitoring Installation and Setup Guide for more information on adding application support.
	• Verify that the following files are available in the directory:
	`install_dir\ATTRLIB\klz.atr`
	`install_dir\CNPS\CMSATR\klz.atr`
	`install_dir\SQLLIB\klz.sql`
	`install_dir\CNPS\SQLLIB\klz.sql`

You successfully migrate an OMEGAMON monitoring agent to IBM Tivoli Monitoring, Version 6.2.0. However, when you configure historical data collection, you see an error message that includes, `Attribute name may be invalid, or attribute file not installed for warehouse agent`.	Install the agent's application support files on the Tivoli Enterprise Monitoring Server, using the following steps: 1. Open the Manage Tivoli Enterprise Monitoring Services window. 2. Right-click the name of the monitoring server. 3. Select **Advanced > Add TEMS Application Support** in the pop-up menu. Add application support if any for any agent that is missing from the list. See in IBM Tivoli Monitoring Installation and Setup Guide for more information on adding application support. Ensure that the agent's application support files are pushed to the system that houses the Warehouse Proxy Agent. The Warehouse Proxy must be able to access the short attribute names for tables and columns. That way, if the longer versions of these names exceed the limits of the Warehouse database, the shorter names can be substituted.

Table 14. Problems and solutions for installation and configuration (continued)

Problem	Solution
You receive the following error: `/data/itm/li6263/lz/bin/klzagent: error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory`	Ensure that the libstdc++.so.5 library is installed.

Table 15. General problems and solutions for uninstallation

Problem	Solution
The way to remove inactive managed systems (systems whose status is OFFLINE) from the Enterprise navigation tree in the portal is not obvious.	When you want to remove a managed system from the navigation tree, complete the following steps: 1. Click **Enterprise** in the navigation tree. 2. Right-click **Workspace -> Managed System Status**. 3. Right-click the offline managed system and select **Clear offline entry**.

Unique names for monitoring components

IBM Tivoli Monitoring might not be able to generate a unique name for monitoring components due to the truncation of names that the product automatically generates.

IBM Tivoli Monitoring automatically creates a name for each monitoring component by concatenating the host name and product code separated by colons (*hostname*:LZ).

Note: When you monitor a multinode system, such as a database, IBM Tivoli Monitoring adds a subsystem name to the concatenated name, typically a database instance name.

The length of the name that IBM Tivoli Monitoring generates is limited to 32 characters. Truncation can result in multiple components having the same 32-character name. If this problem happens, shorten the *hostname* portion of the name as follows:

1. Open the configuration file for the monitoring agent, which is located in the following path: *install_dir*`/config/lz.ini`.

   Note: When you modify the **lz.ini** file, your configuration changes affect only the instance Monitoring Agent for Linux OS that is running on the computer. If you want your configuration changes to affect all agents that run on the computer, modify the *install_dir*`/config/env.config` file.
2. Find the line the begins with **CTIRA_HOSTNAME=**.
3. Type a new name for host name that is a unique, shorter name for the host computer. The final concatenated name including the subsystem name, new host name, and LZ, cannot be longer than 32 characters.

   Note: You must ensure that the resulting name is unique with respect to any existing monitoring component that was previously registered with the Tivoli Enterprise Monitoring Server.
4. Save the file.
5. Restart the agent.

6. If you do not find the files mentioned in Step 1, perform the workarounds listed in the next paragraph.

If you cannot find the **CTIRA_HOSTNAME** environment variable, you must add it to the configuration file of the monitoring agent:

- **On UNIX and Linux:** Add the variable to the config/*product_code*.ini file.

Agent troubleshooting

This section lists problems that might occur with agents.

This appendix provides agent-specific troubleshooting information. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information.

Table 16. Agent problems and solutions

Problem	Solution
A configured and running instance of the monitoring agent is not displayed in the Tivoli Enterprise Portal, but other instances of the monitoring agent on the same system do appear in the portal.	Tivoli Monitoring products use Remote Procedure Call (RPC) to define and control product behavior. RPC is the mechanism that allows a client process to make a subroutine call (such as GetTimeOfDay or ShutdownServer) to a server process somewhere in the network. Tivoli processes can be configured to use TCP/UDP, TCP/IP, SNA, and SSL as the desired protocol (or delivery mechanism) for RPCs. "IP.PIPE" is the name given to Tivoli TCP/IP protocol for RPCs. The RPCs are socket-based operations that use TCP/IP ports to form socket addresses. IP.PIPE implements virtual sockets and multiplexes all virtual socket traffic across a single physical TCP/IP port (visible from the netstat command). A Tivoli process derives the physical port for IP.PIPE communications based on the configured, well-known port for the HUB Tivoli Enterprise Monitoring Server. (This well-known port or BASE_PORT is configured using the 'PORT:' keyword on the KDC_FAMILIES / KDE_TRANSPORT environment variable and defaults to '1918'.) The physical port allocation method is defined as (BASE_PORT + 4096*N) where N=0 for a Tivoli Enterprise Monitoring Server process and N={1, 2, ..., 15} for a non-Tivoli Enterprise Monitoring Server. Two architectural limits result as a consequence of the physical port allocation method: • No more than one Tivoli Enterprise Monitoring Server reporting to a specific Tivoli Enterprise Monitoring Server HUB can be active on a system image. • No more that 15 IP.PIPE processes can be active on a single system image. A single system image can support any number of Tivoli Enterprise Monitoring Server processes (address spaces) provided that each Tivoli Enterprise Monitoring Server on that image reports to a different HUB. By definition, there is one Tivoli Enterprise Monitoring Server HUB per monitoring Enterprise, so this architecture limit has been simplified to one Tivoli Enterprise Monitoring Server per system image. No more that 15 IP.PIPE processes or address spaces can be active on a single system image. With the first limit expressed above, this second limitation refers specifically to Tivoli Enterprise Monitoring Agent processes: no more that 15 agents per system image. This limitation can be circumvented (at current maintenance levels, IBM Tivoli Monitoring V6.1 Fix Pack 4 and later) if the Tivoli Enterprise Monitoring Agent process is configured to use EPHEMERAL IP.PIPE. (This is IP.PIPE configured with the 'EPHEMERAL:Y' keyword in the KDC_FAMILIES / KDE_TRANSPORT environment variable). There is no limitation to the number of ephemeral IP.PIPE connections per system image. If ephemeral endpoints are used, the Warehouse Proxy Agent is accessible from the Tivoli Enterprise Monitoring Server associated with the agents using ephemeral connections either by running the Warehouse Proxy Agent on the same computer or by using the Firewall Gateway feature. (The Firewall Gateway feature relays the Warehouse Proxy Agent connection from the Tivoli Enterprise Monitoring Server computer to the Warehouse Proxy Agent computer if the Warehouse Proxy Agent cannot coexist on the same computer.)
The Monitoring Agent for Linux OS running on a Linux system does not communicate with the Tivoli Enterprise Monitoring Server running on a Z/OS system.	The procedure for seeding the Tivoli Enterprise Monitoring Server running on a Z/OS system for an instance of the Monitoring Agent for Linux OS running on a Linux system can be found in *Configuring Tivoli Enterprise Monitoring Server on z/OS®*.

Table 16. Agent problems and solutions (continued)

Problem	Solution
The agent's process, **klzagent** uses a large amount of system resources.	In most cases, the problem occurs during the backup. Any one of the following scenarios can cause this problem. **The agent is running during the backup** After backing up, the agent is started during system startup. **Multiple agents are running at the same time.** The computer that hosts the Tivoli Enterprise Monitoring Server was rebooted and the agent has been installed by the `root` user account. **The agent is running during the backup** During the backup, some of the service might be interrupted or not be available or locked for some amount of time. While the backup process is going on, the Monitoring Agent for Linux OS, which is running parallel, might wait for resources to be freed by the backup process. When the backup is completed and you are viewing the agent, high CPU at this point is expected, because the agent is in an uncertain state (backup usually stops several kernel services that could cause this state). For this reason, it is advisable to stop all agents before the backup run, because there might be lost information, file, or API connections. Stop the agent before the backup process starts. **The agent is started during system boot up:** If you use scripts to stop and start the agent, do not start the agent from an **init** process script when you restart the system. The computer that hosts the Tivoli Enterprise Monitoring Server was rebooted and the agent has been installed by the `root` user account. Verify whether the log file has the following information: `Unable to find running Tivoli Enterprise Monitoring Server on CMSLIST`
Attributes do not allow non-ASCII input in the situation editor.	None. Any attribute that does not include "(Unicode)" might support only ASCII characters. For example "Attribute (Unicode)" will support unicode but "Attribute" without "(Unicode)" might only support ASCII characters.
In the User workspace, data does not show up in the User Login Information (table view).	This problem arises when you install the agent on a 64-bit zLinux operating system, but run the agent in 32-bit mode. The workspace is unable to access user login data. Run the agent in 64-bit mode.

Tivoli Enterprise Portal troubleshooting

Table 17 lists problems that might occur with the Tivoli Enterprise Portal. This appendix provides agent-specific troubleshooting information. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information.

Table 17. Tivoli Enterprise Portal problems and solutions

Problem	Solution
Historical data collection is unavailable because of incorrect queries in the Tivoli Enterprise Portal.	The column, Sort By, Group By, and First/Last functions are not compatible with the historical data collection feature. Use of these advanced functions will make a query ineligible for historical data collection. Even if data collection has been started, you cannot use the time span feature if the query for the chart or table includes any column functions or advanced query options (Sort By, Group By, First / Last). To ensure support of historical data collection, do not use the Sort By, Group By, or First/Last functions in your queries. See the *IBM Tivoli Monitoring Administrator's Guide* or the Tivoli Enterprise Portal online Help for information on the Historical Data Collection function.

Table 17. Tivoli Enterprise Portal problems and solutions (continued)

Problem	Solution
When you use a long process name in the situation, the process name is truncated.	Truncation of process names in the portal display is the expected behavior. 64 bytes is the maximum name length.
You see the following message: `KFWITM083W Default link is disabled for the selected object; please verify link and link anchor definitions.`	You see this message because some links do not have default workspaces. Right-click the link to access a list of workspaces to select.

Troubleshooting for remote deployment

Table 18 lists problems that might occur with remote deployment. This appendix provides agent-specific troubleshooting information. See the *IBM Tivoli Monitoring Troubleshooting Guide* for general troubleshooting information.

This section describes problems and solutions for remote deployment and removal of agent software Agent Remote Deploy:

Table 18. Remote deployment problems and solutions

Problem	Solution
The removal of a monitoring agent fails when you use the remote removal process in the Tivoli Enterprise Portal desktop or browser.	This problem might happen when you attempt the remote removal process immediately after you have restarted the Tivoli Enterprise Monitoring Server. You must allow time for the monitoring agent to refresh its connection with the Tivoli Enterprise Monitoring Server before you begin the remote removal process.

Situation troubleshooting

This section provides information about both general situation problems and problems with the configuration of situations. See the *IBM Tivoli Monitoring Troubleshooting Guide* for more information about troubleshooting for situations.

General situation problems

Table 19 lists problems that might occur with specific situations.

Table 19. Specific situation problems and solutions

Problem	Solution
You want to change the appearance of situations when they are displayed in a Workspace view.	1. Right-click an item in the Navigation tree. 2. Select **Situations** in the pop-up menu. The Situation Editor window is displayed. 3. Select the situation that you want to modify. 4. Use the **Status** pull-down menu in the lower right of the window to set the status and appearance of the Situation when it triggers. **Note:** This status setting is not related to severity settings in IBM Tivoli Enterprise Console.

Table 19. Specific situation problems and solutions (continued)

Problem	Solution
Situations are triggered in the Tivoli Enterprise Monitoring Server, but events for the situation are not sent to the Tivoli Enterprise Console server. The Tivoli Enterprise Monitoring Server is properly configured for event forwarding, and events for many other situations are sent to the event server.	This condition can occur when a situation is only monitoring the status of other situations. The event forwarding function requires an attribute group reference in the situation in order to determine the correct event class to use in the event. When the situation only monitors other situations, no attribute groups are defined and the event class cannot be determined. Because the event class cannot be determined, no event is sent. This is a limitation of the Tivoli Enterprise Monitoring Server event forwarding function. Situations that only monitor other situations do not send events to the event server.
Monitoring activity requires too much disk space.	Check the RAS trace logging settings that are described in "Setting RAS trace parameters" on page 217. For example, trace logs grow rapidly when you apply the **ALL** logging option.
A formula that uses mathematical operators appears to be incorrect. For example, if you were monitoring Linux, a formula that calculates when **Free Memory** falls under 10 percent of **Total Memory** does not work: `LT #'Linux_VM_Stats.Total_Memory' / 10`	This formula is incorrect because situation predicates support only logical operators. Your formulas cannot have mathematical operators. **Note:** The Situation Editor provides alternatives to math operators. Regarding the example, you can select **% Memory Free** attribute and avoid the need for math operators.
If you are running a Version 350 Monitoring Agent for Linux OS and you choose to alter the views to include a Version 610 UNICODE attribute, be aware that data for this attribute is not displayed and you see a blank column in this view.	To enable Unicode and other features, upgrade the monitoring agent to IBM Tivoli Monitoring, Version 6.1.0.
IBM Tivoli Monitoring is configured to provide data to the optional product IBM Tivoli Enterprise Console. However, a situation displays the severity UNKNOWN in IBM Tivoli Enterprise Console.	For a situation to have the correct severity in TEC for those situations which are not mapped, you need to ensure that one of the following is true: • Specify the severity in the SITINFO column of the O4SRV.TSITDESC table. For example use the values 'SEV=Critical' and 'SEV=Warning' for the SITINFO column in your kxx.sql file, which adds application support to the monitoring product. —OR— • Have the name of the situation ends with '_Warn' or '_Warning' for WARNING severity and '_Cri' or '_Critical' for Critical severity
You see the 'Unable to get attribute name' error in the Tivoli Enterprise Monitoring Server log after creating a situation.	Install the agent's application support files on the Tivoli Enterprise Monitoring Server, using the following steps: 1. Open the Manage Tivoli Enterprise Monitoring Services window. 2. Right-click the name of the monitoring server. 3. Select **Advanced > Add TEMS Application Support** in the pop-up menu. Add application support if any for any agent that is missing from the list. See in IBM Tivoli Monitoring Installation and Setup Guide for more information on adding application support.
Events received at the Tivoli Enterprise Console server from IBM Tivoli Monitoring do not have values for all event attributes (slots) even though the values are visible in workspace views.	The problem is due to a limitation in the IBM Tivoli Monitoring interface code that generates Tivoli Enterprise Console events from situations. The situation results are provided in a chain of buffers of 3000 bytes each. The interface code currently extracts event information from only the first buffer. When situations or agent table data expands into a second buffer, this additional data is not examined, and it is not included in events sent to the Tivoli Enterprise Console server.

Table 19. Specific situation problems and solutions (continued)

Problem	Solution
Tivoli Enterprise Console events from IBM Tivoli Monitoring 6.2 for IBM Tivoli Monitoring 5.x migrated situations receive parsing errors in the Tivoli Enterprise Console server.	Complete the following two steps: 1. Ensure that you have the IBM Tivoli Monitoring 6.2 Event Sync installed on your Tivoli Enterprise Console server. 2. Obtain updated baroc files from IBM Tivoli Monitoring 6.2 for the monitoring agent's events. Updated baroc files are on the Tivoli Enterprise Monitoring Server in the *CANDLEHOME*/CMS/TECLIB/itm5migr directory.
You are receiving Tivoli Business Systems Management events that cannot be associated due to application_oid and application_class not being set.	The problem is due to IBM Tivoli Monitoring 6.2 sending Tivoli Enterprise Console events for IBM Tivoli Monitoring 5.x migrated situations. These events are not able to set the cited slot values. Replace the *agent_name*_forward_tbsm_event_cb.sh script on the Tivoli Enterprise Console server with the version of this file from the Tivoli Enterprise Monitoring Server in the *CANDLEHOME*/CMS/TECLIB/itm5migr directory.

Problems with configuration of situations

Table 20 lists problems that might occur with situations.

This section provides information for troubleshooting for agents. Be sure to consult the *IBM Tivoli Monitoring Troubleshooting Guide* for more general troubleshooting information.

Table 20. Problems with configuring situations that you solve in the Situation Editor

Problem	Solution
Note: To get started with the solutions in this section, perform these steps: 1. Launch the Tivoli Enterprise Portal. 2. Click **Edit > Situation Editor**. 3. In the tree view, choose the agent whose situation you want to modify. 4. Choose the situation in the list. The Situation Editor view is displayed.	
The situation for a specific agent is not visible in the Tivoli Enterprise Portal.	Open the Situation Editor. Access the All managed servers view. If the situation is absent, confirm that application support for Monitoring Agent for Linux OS has been added to the monitoring server. If not, add application support to the server, as described in the *IBM Tivoli Monitoring Installation and Setup Guide.*
The monitoring interval is too long.	Access the Situation Editor view for the situation that you want to modify. Check the **Sampling interval** area in the **Formula** tab. Adjust the time interval as needed.
The situation did not activate at startup.	Manually recycle the situation as follows: 1. Right-click the situation and choose **Stop Situation**. 2. Right-click the situation and choose **Start Situation**. **Note:** You can permanently avoid this problem by placing a check mark in the **Run at Startup** option of the Situation Editor view for a specific situation.
The situation is not displayed.	Click the **Action** tab and check whether the situation has an automated corrective action. This action can occur directly or through a policy. The situation might be resolving so quickly that you do not see the event or the update in the graphical user interface.
An Alert event has not occurred even though the predicate has been properly specified.	Check the logs, reports, and workspaces.
A situation fires on an unexpected managed object.	Confirm that you have distributed and started the situation on the correct managed system.

Table 20. Problems with configuring situations that you solve in the Situation Editor (continued)

Problem	Solution
The product did not distribute the situation to a managed system.	Click the **Distribution** tab and check the distribution settings for the situation.
The situation does not fire. Incorrect predicates are present in the formula that defines the situation. For example, the managed object shows a state that normally triggers a monitoring event, but the situation is not true because the wrong attribute is specified in the formula.	In the **Formula** tab, analyze predicates as follows: 1. Click the *fx* icon in the upper-right corner of the Formula area. The Show formula window is displayed. a. Confirm the following details in the **Formula** area at the top of the window: • The attributes that you intend to monitor are specified in the formula. • The situations that you intend to monitor are specified in the formula. • The logical operators in the formula match your monitoring goal. • The numerical values in the formula match your monitoring goal. b. (*Optional*) Click the **Show detailed formula** check box in the lower left of the window to see the original names of attributes in the application or operating system that you are monitoring. c. Click **OK** to dismiss the Show formula window. 2. (*Optional*) In the Formula area of the **Formula** tab, temporarily assign numerical values that will immediately trigger a monitoring event. The triggering of the event confirms that other predicates in the formula are valid. **Note:** After you complete this test, you must restore the numerical values to valid levels so that you do not generate excessive monitoring data based on your temporary settings.

Table 21. Problems with configuration of situations that you solve in the Workspace area

Problem	Solution
Situation events are not displayed in the Events Console view of the workspace.	Associate the situation with a workspace. **Note:** The situation does not need to be displayed in the workspace. It is sufficient that the situation be associated with any workspace.
You do not have access to a situation.	**Note:** You must have administrator privileges to perform these steps. 1. Select **Edit** > **Administer Users** to access the Administer Users window. 2. In the Users area, select the user whose privileges you want to modify. 3. In the Permissions tab, Applications tab, and Navigator Views tab, select the permissions or privileges that correspond to the user's role. 4. Click **OK**.
A managed system seems to be offline.	1. Select Physical View and highlight the Enterprise Level of the navigator tree. 2. Select **View** > **Workspace** > **Managed System Status** to see a list of managed systems and their status. 3. If a system is offline, check network connectivity and status of the specific system or application.

Table 22. Problems with configuration of situations that you solve in the Manage Tivoli Enterprise Monitoring Services window

Problem	Solution
After an attempt to restart the agents in the Tivoli Enterprise Portal, the agents are still not running.	Check the system status and check the appropriate IBM Tivoli Monitoring logs.

Table 22. Problems with configuration of situations that you solve in the Manage Tivoli Enterprise Monitoring Services window (continued)

Problem	Solution
The Tivoli Enterprise Monitoring Server is not running.	Check the system status and check the appropriate IBM Tivoli Monitoring logs.
The managed objects you created are firing on incorrect managed systems.	Check the managed system distribution on both the situation and the managed object settings sheets.

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:

Online

Go to the IBM Software Support site at http://www.ibm.com/software/support/probsub.html and follow the instructions.

IBM Support Assistant

The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, go to http://www.ibm.com/software/support/isa.

Appendix E. Discovery Library Adapter for the monitoring agent

This chapter contains information about the Discovery Library Adapter (DLA) for the Monitoring Agent for Linux.

About the DLA

The Tivoli Management Services DLA discovers resources and relationships and creates a Discovery Library Book file. The Book follows the Discovery Library IdML schema and is used to populate the Configuration Management Database (CMDB) and Tivoli Business System Management products. The Tivoli Management Services DLA discovers Linux resources. For all Linux systems that are active and online at the Tivoli Enterprise Portal Server, information is included in the discovery book for those resources. The Tivoli Management Services DLA discovers active resources. It is run on-demand and can be run periodically to discover resources that were not active during previous discoveries.

The DLA discovers Linux components.

More information about DLAs

The following sources contain additional information about using the DLA program with all monitoring agents:

- The *IBM Tivoli Monitoring Administrator's Guide* contains information about using the Tivoli Management Services Discovery Library Adapter.
- For information about using a DLA with Tivoli Application Dependency Discovery Manager (TADDM), see the information center at http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.taddm.doc_7.1/cmdb_welcome.html

Linux data model class types represented in CDM

This section contains information about how the various source application data objects map to classes in the Common Data Model (CDM) for the Monitoring Agent for Linux.

The following information is provided for each class where appropriate:

Relationships
CDM relationships (hierarchical) between currently identified model objects

CDM attributes, agent attributes, descriptions, and examples
CDM and agent attributes that are required to create an instance of a resource, descriptions of the attributes, and examples of the attributes

Linux class

The following information describes the Linux class.

CDM class name
sys.linux.Linux

Relationships

- installedOn
- runsOn

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: ManagedSystemName
  Agent attribute: Managed System Name
  Description: MSYSN
- CDM attribute: OS Version
  Agent attribute: OS Version
  Description: OS_VERSION
- CDM attribute: Name
  Agent attribute: OS Name
  Description: FQHN
- CDM attribute: Fqdn
  Agent attribute: Fully Qualified Domain Name
  Description: PRI_DNS_NAME

ComputerSystem class

The following information describes the ComputerSystem class.

CDM class name

sys.ComputerSystem

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: ManagedSystemName
  Agent attribute: Managed System Name
  Description: MSYSN
- CDM attribute: Name
  Agent attribute: OS Name
  Description: FQHN
- CDM attribute: Signature
  Agent attribute: IP Address
  Description: PRI_IP_ADDR
- CDM attribute: PrimaryMACAddress
  Agent attribute: MAC Address
  Description: PRI_MAC_ADDR
- CDM attribute: Type
  Agent attribute: Manufacturer
  Description: ComputerSystem
- CDM attribute: SystemBoardUUID
  Agent attribute: System Board UUID
  Description: SYS_UUID
- CDM attribute: SerialNumber
  Agent attribute: Serial Number
  Description: SERIAL
- CDM attribute: Model
  Agent attribute: Model
  Description: MODEL

IpInterface class

The following information describes the IpInterface class.

CDM class name
net.IpInterface

Relationships

- contains

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: ComputerSystem
  Description: IF_IP_ADDR

IpV4Address class

The following information describes the IpV4Address class.

CDM class name
net.IpV4Address

Relationships

- bindsTo

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: DotNotation
  Description: IF_IP_ADDR
- CDM attribute: Label
  Description: IF_IP_ADDR

IpV6Address class

The following information describes the IpV6Address class.

CDM class name
net.IpV6Address

Relationships

- bindsTo

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: DotNotation
  Description: IF_IP_ADDR
- CDM attribute: Label
  Description: IF_IP_ADDR

Fqdn class

The following information describes the Fqdn class.

CDM class name
net.Fqdn

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: Fqdn
  Agent attribute: Fully Qualified Domain Name
  Description: IF_DNS_NAME

TMSAgent class

The following information describes the TMSAgent class.

CDM class name
app.TMSAgent

Relationships

- installedOn
- monitors

CDM attributes, agent attributes, descriptions, and examples

- CDM attribute: ManagedSystemName
  Agent attribute: Managed System Name
  Description: MSYSN
- CDM attribute: ManagedObjectName
  Description: MSYSN
- CDM attribute: SoftwareVersion
  Description: PRODVER
- CDM attribute: ProductCode
  Description: PRODUCT
- CDM attribute: Affinity
  Description: PRODAFF
- CDM attribute: Label
  Description: MSYSN

Appendix F. Documentation library

This appendix contains information about the publications related to IBM Tivoli Monitoring and to the commonly shared components of Tivoli Management Services. These publications are listed in the following categories:

- IBM Tivoli Monitoring library
- Related publications

See *IBM Tivoli Monitoring and OMEGAMON XE Products: Documentation Guide*, SC23-8816, for information about accessing and using the publications. You can find the *Documentation Guide* in the IBM Tivoli Monitoring and OMEGAMON XE Information Center at http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/. To open the *Documentation Guide* in the information center, select **Using the publications** in the **Contents** pane.

To find a list of new and changed publications, click **What's new** on the Welcome page of the IBM Tivoli Monitoring and OMEGAMON XE Information Center. To find publications from the previous version of a product, click **Previous versions** under the name of the product in the **Contents** pane.

IBM Tivoli Monitoring library

The following publications provide information about IBM Tivoli Monitoring and about the commonly shared components of Tivoli Management Services:

- *Quick Start Guide*, GI11-8058

  Introduces the components of IBM Tivoli Monitoring.
- *Installation and Setup Guide*, GC32-9407

  Provides instructions for installing and configuring IBM Tivoli Monitoring components on Windows, Linux, and UNIX systems.
- *Program Directory for IBM Tivoli Management Services on z/OS*, GI11-4105

  Gives instructions for the SMP/E installation of the Tivoli Management Services components on z/OS.
- *Configuring the Tivoli Enterprise Monitoring Server on z/OS*, SC32-9463

  Gives detailed instructions for using the Configuration Tool to configure Tivoli Enterprise Monitoring Server on z/OS systems. Includes scenarios for using batch mode to replicate monitoring environments across the z/OS enterprise. Also provides instructions for setting up security and for adding application support to a Tivoli Enterprise Monitoring Server on z/OS.
- *Administrator's Guide*, SC32-9408

  Describes the support tasks and functions required for the Tivoli Enterprise Portal Server and clients, including Tivoli Enterprise Portal user administration.

- *High-Availability Guide for Distributed Systems*, SC23-9768

  Gives instructions for several methods of ensuring the availability of the IBM Tivoli Monitoring components.
- Tivoli Enterprise Portal online help

  Provides context-sensitive reference information about all features and customization options of the Tivoli Enterprise Portal. Also gives instructions for using and administering the Tivoli Enterprise Portal.
- *Tivoli Enterprise Portal User's Guide*, SC32-9409

  Complements the Tivoli Enterprise Portal online help. The guide provides hands-on lessons and detailed instructions for all Tivoli Enterprise Portal features.
- *Command Reference*, SC32-6045

  Provides detailed syntax and parameter information, as well as examples, for the commands you can use in IBM Tivoli Monitoring.
- *Troubleshooting Guide*, GC32-9458

  Provides information to help you troubleshoot problems with the software.
- *Messages*, SC23-7969

  Lists and explains messages generated by all IBM Tivoli Monitoring components and by z/OS-based Tivoli Management Services components (such as Tivoli Enterprise Monitoring Server on z/OS and TMS:Engine).
- *IBM Tivoli Universal Agent User's Guide*, SC32-9459

  Introduces you to the IBM Tivoli Universal Agent, an agent of IBM Tivoli Monitoring. The IBM Tivoli Universal Agent enables you to use the monitoring and automation capabilities of IBM Tivoli Monitoring to monitor any type of data you collect.
- *IBM Tivoli Universal Agent API and Command Programming Reference Guide*, SC32-9461

  Explains the procedures for implementing the IBM Tivoli Universal Agent APIs and provides descriptions, syntax, and return status codes for the API calls and command-line interface commands.
- *Agent Builder User's Guide*, SC32-1921

  Explains how to use the Agent Builder for creating monitoring agents and their installation packages, and for adding functions to existing agents.

Documentation for the base agents

If you purchased IBM Tivoli Monitoring as a product, you received a set of *base* monitoring agents as part of the product. If you purchased a monitoring agent product (for example, an OMEGAMON XE product) that includes the commonly shared components of Tivoli Management Services, you did not receive the base agents.

The following publications provide information about using the base agents.

- Operating system agents:
  - *Windows OS Agent User's Guide*, SC32-9445
  - *UNIX OS Agent User's Guide*, SC32-9446
  - *Linux OS Agent User's Guide*, SC32-9447
  - *i5/OS Agent User's Guide*, SC32-9448
  - *UNIX Log Agent User's Guide*, SC32-9471

- Agentless operating system monitors:
  - *Agentless Monitoring for Windows Operating Systems User's Guide*, SC23-9765
  - *Agentless Monitoring for AIX Operating Systems User's Guide*, SC23-9761
  - *Agentless Monitoring for HP-UX Operating Systems User's Guide*, SC23-9763
  - *Agentless Monitoring for Solaris Operating Systems User's Guide*, SC23-9764
  - *Agentless Monitoring for Linux Operating Systems User's Guide*, SC23-9762
- Warehouse agents:
  - *Warehouse Summarization and Pruning Agent User's Guide*, SC23-9767
  - *Warehouse Proxy Agent User's Guide*, SC23-9766
- System P agents:
  - *AIX Premium Agent User's Guide*, SA23-2237
  - *CEC Base Agent User's Guide*, SC23-5239
  - *HMC Base Agent User's Guide*, SA23-2239
  - *VIOS Premium Agent User's Guide*, SA23-2238
- Other base agents:
  - *Monitoring Agent for IBM Tivoli Monitoring 5.x Endpoint User's Guide*, SC32-9490

Related publications

You can find useful information about related products in the IBM Tivoli Monitoring and OMEGAMON XE Information Center at http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/.

Other sources of documentation

You can also obtain technical documentation about IBM Tivoli Monitoring and related products from the following sources:

- IBM Tivoli Open Process Automation Library (OPAL)

  http://www.ibm.com/software/tivoli/opal

  OPAL is an online catalog that contains integration documentation and other downloadable product extensions.
- Redbooks

  http://www.redbooks.ibm.com/

  IBM Redbooks® and Redpapers include information about products from platform and solution perspectives.
- Technotes

  Technotes provide the latest information about known product limitations and workarounds. You can find Technotes through the IBM Software Support Web site at http://www.ibm.com/software/support.
- Tivoli wikis on the IBM developerWorks Web site

  Tivoli Wiki Central at http://www.ibm.com/developerworks/wikis/display/tivoli/Home is the home for interactive wikis that offer best practices and scenarios for using Tivoli products. The wikis contain white papers contributed by IBM employees, and content created by customers and business partners.

Two of these wikis are of particular relevance to IBM Tivoli Monitoring:

- Tivoli Distributed Monitoring and Application Management Wiki at http://www.ibm.com/developerworks/wikis/display/tivolimonitoring/Home provides information about IBM Tivoli Monitoring and related distributed products, including IBM Tivoli Composite Application Management products.
- Tivoli System z Monitoring and Application Management Wiki at http://www.ibm.com/developerworks/wikis/display/tivoliomegamon/Home provides information about the OMEGAMON XE products, NetView for z/OS, Tivoli Monitoring Agent for z/TPF, and other System z monitoring and application management products.

Appendix G. Accessibility

Accessibility features help users with physical disabilities, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in this product enable users to do the following:

- Use assistive technologies, such as screen-reader software and digital speech synthesizer, to hear what is displayed on the screen. Consult the product documentation of the assistive technology for details on using those technologies with this product.
- Operate specific or equivalent features using only the keyboard.
- Magnify what is displayed on the screen.

In addition, the product documentation was modified to include the following features to aid accessibility:

- All documentation is available in both HTML and convertible PDF formats to give the maximum opportunity for users to apply screen-reader software.
- All images in the documentation are provided with alternative text so that users with vision impairments can understand the contents of the images.

Navigating the interface using the keyboard

Standard shortcut and accelerator keys are used by the product and are documented by the operating system. Refer to the documentation provided by your operating system for more information.

Magnifying what is displayed on the screen

You can enlarge information on the product windows using facilities provided by the operating systems on which the product is run. For example, in a Microsoft Windows environment, you can lower the resolution of the screen to enlarge the font sizes of the text on the screen. Refer to the documentation provided by your operating system for more information.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

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

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

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.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Cell Broadband Engine and Cell/B.E. are trademarks of Sony Computer Entertainment, Inc., in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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.

Other company, product, and service names may be trademarks or service marks of others.

Index

M

N

O

P

Q

R

S

T

U

W

Printed in USA

SC32-9447-03