



# APPENDIX **A**

## Performance Objects and Counters for the System

---

This appendix provides information on system-related objects and counters. For information on specific counters, click the blue text in the following list to go to the object:

- [Cisco Tomcat Connector, page A-2](#)
- [Cisco Tomcat JVM, page A-3](#)
- [Cisco Tomcat Web Application, page A-4](#)
- [Database Change Notification Client, page A-5](#)
- [Database Change Notification Server, page A-6](#)
- [Database Change Notification Subscription, page A-6](#)
- [Database Local DSN, page A-7](#)
- [DB User Host Information Counters, page A-7](#)
- [Enterprise Replication DBSpace Monitors, page A-7](#)
- [Enterprise Replication Perfmon Counters, page A-7](#)
- [IP, page A-8](#)
- [Memory, page A-9](#)
- [Network Interface, page A-10](#)
- [Number of Replicates Created and State of Replication, page A-11](#)
- [Partition, page A-12](#)
- [Process, page A-13](#)
- [Processor, page A-14](#)
- [System, page A-15](#)
- [TCP, page A-16](#)
- [Thread, page A-16](#)



**Tip**

For the latest performance monitoring counters, objects, and counter descriptions that are available for system monitoring, access the performance monitoring counters in the Cisco Unified Real-Time Monitoring Tool. In RTMT, you can review a counter description, as described in the [“Using Performance Queries to Add a Counter”](#) section on page 6-3.

---

# Cisco Tomcat Connector

The Tomcat Hypertext Transport Protocol (HTTP)/HTTP Secure (HTTPS) Connector object provides information about Tomcat connectors. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related web pages are accessed. The Secure Socket Layer (SSL) status of the URLs for web applications provides the basis for the instance name for each Tomcat HTTP Connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL. [Table A-1](#) contains information on the Tomcat HTTP connector counters.

**Table A-1** Cisco Tomcat Connector

| Counters       | Counter Description  |
|----------------|--|
| Errors         | This counter represents the total number of HTTP errors (for example, 401 Unauthorized) that the connector encountered. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL. |
| MBytesReceived | This counter represents the amount of data that the connector received. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL.   |
| MBytesSent     | This counter represents the amount of data that the connector sent. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL.   |
| Requests       | This counter represents the total number of request that the connector handled. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL.   |

**Table A-1** Cisco Tomcat Connector (continued)

| Counters     | Counter Description  |
|--------------|--|
| ThreadsTotal | This counter represents the current total number of request processing threads, including available and in-use threads, for the connector. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL.  |
| ThreadsMax   | <p>This counter represents the maximum number of request processing threads for the connector. Each incoming request on a Cisco Unified Communications Manager related window requires a thread for the duration of that request. If more simultaneous requests are received than the currently available request processing threads can handle, additional threads will be created up to the configured maximum shown in this counter. If still more simultaneous requests are received, they accumulate within the server socket that the connector created, up to an internally specified maximum number. Any further simultaneous requests will receive connection refused messages until resources are available to process them.</p> <p>A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</p> |
| ThreadsBusy  | This counter represents the current number of busy/in-use request processing threads for the connector. A Tomcat Connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when web pages that are related to Cisco Unified Communications Manager are accessed. The Secure Sockets Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL.  |

## Cisco Tomcat JVM

The Cisco Tomcat Java Virtual Machine (JVM) object provides information about the Tomcat JVM, which represents, among other things, a pool of common resource memory that Cisco Unified Communications Manager related web applications such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, Cisco Unity Connection Administration, and more use. [Table A-2](#) contains information on the Tomcat JVM counters.

Table A-2 Tomcat JVM

| Counters          | Counter Description  |
|-------------------|--|
| KBytesMemoryFree  | This counter represents the amount of free dynamic memory block (heap memory) in the Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration create. When the amount of free dynamic memory is low, more memory gets automatically allocated, and total memory size (represented by the KbytesMemoryTotal counter) increases but only up to the maximum (represented by the KbytesMemoryMax counter). You can determine the amount of memory in use by subtracting KBytesMemoryFree from KbytesMemoryTotal. |
| KBytesMemoryMax   | This counter represents the amount of free dynamic memory block (heap memory) in the Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration create.  |
| KBytesMemoryTotal | This counter represents the current total dynamic memory block size, including free and in-use memory, of Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration create.   |

## Cisco Tomcat Web Application

The Cisco Tomcat Web Application object provides information about how to run Cisco Unified Communications Manager web applications. The URLs for the web application provide basis for the instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (<https://<IP Address>:8443/ccmadmin>) gets identified by ccmadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccuser, Cisco Unity Connection Administration (<https://<IP Address>:8443/cuadmin>) gets identified by cuadmin, and URLs that do not have an extension, such as <https://<IP Address>:8443> or <http://<IP Address>:8080>, get identified by \_root. [Table A-3](#) contains information on the Tomcat Web Application counters.

**Table A-3** *Tomcat Web Application*

| Counters       | Counter Description  |
|----------------|--|
| Errors         | This counter represents the total number of HTTP errors (for example, 401 Unauthorized) that a Cisco Unified Communications Manager related web application encountered. The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://<IP Address>:8443/ccadmin) gets identified by ccadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccuser, Cisco Unity Connection Administration (https://<IP Address>:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://<IP Address>:8443 or http://<IP Address>:8080), get identified by _root.          |
| Requests       | This counter represents the total number of requests that the web application handles. Each time that a web application is accessed, its Requests counter increments accordingly. The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://<IP Address>:8443/ccadmin) gets identified by ccadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccuser, Cisco Unity Connection Administration (https://<IP Address>:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://<IP Address>:8443 or http://<IP Address>:8080), get identified by _root. |
| SessionsActive | This counter represents the number of sessions that the web application currently has active (in use). The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://<IP Address>:8443/ccadmin) gets identified by ccadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccuser, Cisco Unity Connection Administration (https://<IP Address>:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://<IP Address>:8443 or http://<IP Address>:8080), get identified by _root.  |

## Database Change Notification Client

The Database Change Notification Client object provides information on change notification clients. [Table A-4](#) contains information on the Database Change Notification Client counters.

**Table A-4** *Database Change Notification Client*

| Counters           | Counter Descriptions  |
|--------------------|---|
| MessagesProcessed  | This counter represents the number of database change notifications that have been processed. This counter refreshes every 15 seconds.  |
| MessagesProcessing | This counter represents the number of change notification messages that are currently being processed or are waiting to be processed in the change notification queue for this client. This counter refreshes every 15 seconds. |

**Table A-4 Database Change Notification Client (continued)**

| Counters         | Counter Descriptions   |
|------------------|--|
| QueueHeadPointer | This counter represents the head pointer to the change notification queue. The head pointer acts as the starting point in the change notification queue. To determine the number of notifications in the queue, subtract the head pointer value from the tail pointer value. By default, this counter refreshes every 15 seconds.  |
| QueueMax         | This counter represents the largest number of change notification messages that will be processed for this client. This counter remains cumulative since the last restart of the Cisco Database Layer Monitor service.   |
| QueueTailPointer | This counter represents the tail pointer to the change notification queue. The tail pointer represents the ending point in the change notification queue. To determine the number of notifications in the queue, subtract the head pointer value from the tail pointer value. By default, this counter refreshes every 15 seconds. |
| TablesSubscribed | This counter represents the number of tables in which this client has subscribed.  |

## Database Change Notification Server

The Database Change Notification Server object provides information on different change-notification-related statistics. [Table A-5](#) contains information on the Database Change Notification Server counters.

**Table A-5 Database Change Notification Server**

| Counter                | Counter Descriptions  |
|------------------------|---|
| Clients                | This counter represents the number of change notification clients (services/servlets) that have subscribed for change notification.   |
| Queue Delay            | <p>This counter provides the number of seconds that the change notification process has messages to process but is not processing them. This condition is true if:</p> <ul style="list-style-type: none"> <li>either Change Notification Requests Queued in Database (QueuedRequestsInDB) and Change Notification Requests Queued in Memory (QueuedRequestsInMemory) are non-zero, or</li> <li>the Latest Change Notification Messages Processed count is not changing.</li> </ul> <p>This condition gets checked every 15 seconds.</p> |
| QueuedRequestsInDB     | This counter represents the number of change notification records that are in the DBCNQueue (Database Change Notification Queue) table via direct TCP/IP connection (not queued in shared memory). This counter refreshes every 15 seconds.   |
| QueuedRequestsInMemory | This counter represents the number of change notification requests that are queued in shared memory.  |

## Database Change Notification Subscription

The Database Change Notification Subscription object displays the names of tables where the client will receive Change Notifications.

The SubscribedTable object displays the table with the service or servlet that will receive change notifications. Because the counter does not increment, this display occurs for informational purposes only.

## Database Local DSN

The Database Local Data Source Name (DSN) object and LocalDSN counter provide the DSN information for the local machine. [Table A-6](#) contains information on the Database local DSN.

**Table A-6** Database Local Data Source Name

| Counters            | Counter Descriptions  |
|---------------------|---|
| CcmDbSpace_Used     | This counter represents the amount of Ccm DbSpace that is being consumed                            |
| CcmtempDbSpace_Used | This counter represents the amount of Ccmtemp DbSpace that is being consumed.                       |
| CNDbSpace_Used      | This counter represents the percentage of CN dbspace consumed.                                      |
| LocalDSN            | This counter represents the data source name (DSN) that is being referenced from the local machine. |
| SharedMemory_Free   | This counter represents total shared memory that is free.   |
| SharedMemory_Used   | This counter total shared memory that is used.  |
| RootDbSpace_Used    | This counter represents the amount of RootDbSpace that is being consumed.                           |

## DB User Host Information Counters

The DB User Host Information object provides information on DB User Host.

The DB:User:Host Instance object displays the number of connections that are present for each instance of DB:User:Host.

## Enterprise Replication DBSpace Monitors

The enterprise replication DBSpace monitors object displays the usage of various ER DbSpaces. [Table A-7](#) contains information on the enterprise replication DB monitors.

**Table A-7** Enterprise Replication DBSpace Monitors

| Counters         | Counter Descriptions  |
|------------------|---|
| ERDbSpace_Used   | This counter represents the amount of enterprise replication DbSpace that was consumed. |
| ERSBDbSpace_Used | This counter represents the amount of ERDbSpace that was consumed.                      |

## Enterprise Replication Perfmon Counters

The Enterprise Replication Perfmon Counter object provides information on the various replication counters.

The ServerName:ReplicationQueueDepth counter displays the server name followed by the replication queue depth.

## IP

The IP object provides information on the IP statistics on your system. [Table A-8](#) contains information on the IP counters.

**Table A-8** IP

| Counters         | Counter Descriptions   |
|------------------|--|
| Frag Creates     | This counter represents the number of IP datagrams fragments that have been generated at this entity.  |
| Frag Fails       | This counter represents the number of IP datagrams that were discarded at this entity because the datagrams could not be fragmented, such as datagrams where the Do not Fragment flag was set.   |
| Frag OKs         | This counter represents the number of IP datagrams that were successfully fragmented at this entity.   |
| In Delivers      | This counter represents the number of input datagrams that were delivered to IP user protocols. This includes Internet Control Message Protocol (ICMP).  |
| In Discards      | This counter represents the number of input IP datagrams where no problems were encountered, but which were discarded. Lack of buffer space provides one possible reason. This counter does not include any datagrams that were discarded while awaiting reassembly.   |
| In HdrErrors     | This counter represents the number of input datagrams that were discarded with header errors. This includes bad checksums, version number mismatch, other format errors, time-to-live exceeded, and other errors that were discovered in processing their IP options.  |
| In Receives      | This counter represents the number of input datagrams that were received from all network interfaces. This counter includes datagrams that were received with errors   |
| In UnknownProtos | This counter represents the number of locally addressed datagrams that were received successfully but discarded because of an unknown or unsupported protocol.   |
| InOut Requests   | This counter represents the number of incoming IP datagrams that were received and the number of outgoing IP datagrams that were sent.   |
| Out Discards     | This counter represents the number of output IP datagrams that were not transmitted and were discarded. Lack of buffer space provides one possible reason.   |
| Out Requests     | This counter represents the total number of IP datagrams that local IP user-protocols (including ICMP) supply to IP in requests transmission. This counter does not include any datagrams that were counted in ForwDatagrams.  |
| Reasm Fails      | This counter represents the number of IP reassembly failures that the IP reassembly algorithm detected, including time outs, errors, and so on. This counter does not represent the discarded IP fragments because some algorithms, such as the algorithm in RFC 815, can lose track of the number of fragments because it combines them as they are received. |



**Table A-8** *IP (continued)*

| Counters    | Counter Descriptions   |
|-------------|--|
| Reasm OKs   | This counter represents the number of IP datagrams that were successfully reassembled.                         |
| Reasm Reqds | This counter represents the number of IP fragments that were received that required reassembly at this entity. |

## Memory

The memory object provides information about the usage of physical memory and swap memory on the server. [Table A-9](#) contains information on memory counters.

**Table A-9** *Memory*

| Counters             | Counter Descriptions   |
|----------------------|--|
| % Mem Used           | This counter displays the system physical memory utilization as a percentage. The value of this counter equals (Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes) / Total KBytes, which also corresponds to the Used KBytes/Total KBytes.   |
| % Page Usage         | This counter represents the percentage of active pages.  |
| % VM Used            | This counter displays the system virtual memory utilization as a percentage. The value of this counter equals (Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes + Used Swap KBytes) / (Total KBytes + Total Swap KBytes), which also corresponds to Used VM KBytes/Total VM KBytes. |
| Buffers KBytes       | This counter represents the capacity of buffers in your system in kilobytes.   |
| Cached KBytes        | This counter represents the amount of cached memory in kilobytes.  |
| Free KBytes          | This counter represents the total amount of memory that is available in your system in kilobytes.  |
| Free Swap KBytes     | This counter represents the amount of free swap space that is available in your system in kilobytes.   |
| Faults Per Sec       | This counter represents the number of page faults (both major and minor) that the system made per second (post 2.5 kernels only). This does not necessarily represent a count of page faults that generate I/O because some page faults can get resolved without I/O.  |
| Low Total            | This counter represents the total low (non-paged) memory for kernel.   |
| Low Free             | This counter represents the total free low (non-paged) memory for kernel.  |
| Major Faults Per Sec | This counter represents the number of major faults that the system has made per second that have required loading a memory page from disk (post 2.5 kernels only).   |
| Pages                | This counter represents the number of pages that the system paged in from the disk plus the number of pages that the system paged out to the disk.   |
| Pages Input          | This counter represents the number of pages that the system paged in from the disk.  |

**Table A-9** *Memory (continued)*

| Counters             | Counter Descriptions  |
|----------------------|---|
| Pages Input Per Sec  | This counter represents the total number of kilobytes that the system paged in from the disk per second.  |
| Pages Output         | This counter represents the number of pages that the system paged out to the disk.  |
| Pages Output Per Sec | This counter represents the total number of kilobytes that the system paged out to the disk per second.   |
| Shared KBytes        | This counter represents the amount of shared memory in your system in kilobytes.  |
| Total KBytes         | This counter represents the total amount of memory in your system in kilobytes.   |
| Total Swap KBytes    | This counter represents the total amount of swap space in your system in kilobytes.   |
| Total VM KBytes      | This counter represents the total amount of system physical and memory and swap space (Total Kbytes + Total Swap Kbytes) that is in use in your system in kilobytes.  |
| Used KBytes          | This counter represents the amount of system physical memory that is in use on the system in kilobytes. The value of the Used KBytes counter equals Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes. The Used KBytes value differs from the Linux term that displays in the top or free command output. The Used value that displays in the top or free command output equals the difference in Total KBytes - Free KBytes and also includes the sum of Buffers KBytes and Cached KBytes. |
| Used Swap KBytes     | This counter represents the amount of swap space that is in use on your system in kilobytes.  |
| Used VM KBytes       | This counter represents the system physical memory and the amount of swap space that is in use on your system in kilobytes. The value equals Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes + Used Swap KBytes. This corresponds to Used Mem KBytes + Used Swap KBytes.  |

## Network Interface

The network interface object provides information about the network interfaces on the system.

[Table A-10](#) contains information on network interface counters.

**Table A-10** *Network Interface*

| Counters   | Counter Descriptions  |
|------------|---|
| Rx Bytes   | This counter represents the number of bytes, including framing characters, that were received on the interface.   |
| Rx Dropped | This counter represents the number of inbound packets that were chosen to be discarded even though no errors had been detected. This prevents the packet from being delivered to a higher layer protocol. Discarding packets to free up buffer space provides one reason. |

**Table A-10**      *Network Interface (continued)*

| Counters      | Counter Descriptions   |
|---------------|--|
| Rx Errors     | This counter represents the number of inbound packets (packet-oriented interfaces) and the number of inbound transmission units (character-oriented or fixed-length interfaces) that contained errors that prevented them from being deliverable to a higher layer protocol.     |
| Rx Multicast  | This counter represents the number of multicast packets that were received on this interface.  |
| Rx Packets    | This counter represents the number of packets that this sublayer delivered to a higher sublayer. This does not include the packets that were addressed to a multicast or broadcast address at this sublayer.   |
| Total Bytes   | This counter represents the total number of received (Rx) bytes and transmitted (Tx) bytes.  |
| Total Packets | This counter represents the total number of Rx packets and Tx packets.   |
| Tx Bytes      | This counter represents the total number of octets, including framing characters, that were transmitted out from the interface.  |
| Tx Dropped    | This counter represents the number of outbound packets that were chosen to be discarded even though no errors were detected. This action prevents the packet from being delivered to a higher layer protocol. Discarding a packet to free up buffer space represents one reason. |
| Tx Errors     | This counter represents the number of outbound packets (packet-oriented interfaces) and the number of outbound transmission units (character-oriented or fixed-length interfaces) that could not be transmitted because of errors.   |
| Tx Packets    | This counter represents the total number of packets that the higher level protocols requested for transmission, including those that were discarded or not sent. This does not include packets that were addressed to a multicast or broadcast address at this sublayer.         |
| Tx QueueLen   | This counter represents the length of the output packet queue (in packets).  |

## Number of Replicates Created and State of Replication

The Number of Replicates Created and State of Replication object provides real-time replication information for the system. [Table A-11](#) contains information on replication counters.

**Table A-11**      **Number of Replicates Created and State of Replication**

| Counters                     | Counter Descriptions   |
|------------------------------|--|
| Number of Replicates Created | This counter displays the number of replicates that were created by Informix for the DB tables. This counter displays information during Replication Setup.  |
| Replicate_State              | <p>This counter represents the state of replication. The following list provides possible values:</p> <ul style="list-style-type: none"> <li>• 0—Initializing. The counter equals 0 when the server is not defined <i>or</i> when the server is defined but realizes the template has not completed.</li> <li>• 1—Replication setup script fired from this node. Cisco recommends that you run <code>utils dbreplication status</code> on the CLI to determine the location and cause of the failure.</li> <li>• 2—Good Replication.</li> <li>• 3—Bad Replication. A counter value of 3 indicates replication in the cluster is bad. It does not mean that replication failed on a particular server in the cluster. Cisco recommends that you run <code>utils dbreplication status</code> on the CLI to determine the location and cause of the failure.</li> <li>• 4—Replication setup did not succeed.</li> </ul> |

## Partition

The partition object provides information about the file system and its usage in the system. [Table A-12](#) contains information on partition counters. These counters are also available for the spare partition, if present.

**Table A-12**      **Partition**

| Counters        | Counter Descriptions   |
|-----------------|--|
| % CPU Time      | This counter represents the percentage of CPU time that is dedicated to handling I/O requests that were issued to the disk. This counter is no longer valid with the counter value -1.   |
| % Used          | This counter represents the percentage of disk space that is in use on this file system.   |
| % Wait in Read  | Not Used. The Await Read Time counter replaces this counter. This counter is no longer valid with the counter value -1.  |
| % Wait in Write | Not Used. The Await Write Time counter replaces this counter. This counter is no longer valid with the counter value -1.   |
| Await Read Time | This counter represents the average time, measured in milliseconds, for Read requests that are issued to the device to be served. This counter is no longer valid with the counter value -1.   |
| Await Time      | This counter represents the average time, measured in milliseconds, for I/O requests that were issued to the device to be served. This includes the time that the requests spent in queue and the time that was spent servicing them. This counter is no longer valid with the counter value -1. |

**Table A-12** *Partition (continued)*

| Counters            | Counter Descriptions  |
|---------------------|---|
| Await Write Time    | This counter represents the average time, measured in milliseconds, for write requests that are issued to the device to be served. This counter is no longer valid with the counter value -1. |
| Queue Length        | This counter represents the average queue length for the requests that were issued to the disk. This counter is no longer valid with the counter value -1.                                    |
| Read Bytes Per Sec  | This counter represents the amount of data in bytes per second that was read from the disk.   |
| Total Mbytes        | This counter represents the amount of total disk space in megabytes that is on this file system.  |
| Used Mbytes         | This counter represents the amount of disk space in megabytes that is in use on this file system.   |
| Write Bytes Per Sec | This counter represents the amount of data that was written to the disk in bytes per second.  |

## Process

The process object provides information about the processes that are running on the system. [Table A-13](#) contains information on process counters.

**Table A-13** *Process*

| Counters         | Counter Descriptions  |
|------------------|---|
| % CPU Time       | This counter, which is expressed as a percentage of total CPU time, represents the tasks share of the elapsed CPU time since the last update.   |
| % MemoryUsage    | This counter represents the percentage of physical memory that a task is currently using.   |
| Data Stack Size  | This counter represents the stack size for task memory status.  |
| Nice             | This counter represents the nice value of the task. A negative nice value indicates that the process has a higher priority while a positive nice value indicates that the process has a lower priority. If the nice value equals zero, do not adjust the priority when you are determining the dispatchability of a task. |
| Page Fault Count | This counter represents the number of major page faults that a task encountered that required the data to be loaded into memory.  |
| PID              | This counter displays the task-unique process ID. The ID periodically wraps, but the value will never equal zero.   |

Table A-13 Process (continued)

| Counters            | Counter Descriptions   |
|---------------------|--|
| Process Status      | This counter displays the process status: <ul style="list-style-type: none"> <li>• 0—Running</li> <li>• 1—Sleeping</li> <li>• 2—Uninterruptible disk sleep</li> <li>• 3—Zombie</li> <li>• 4—Stopped</li> <li>• 5—Paging</li> <li>• 6—Unknown</li> </ul>  |
| Shared Memory Size  | This counter displays the amount of shared memory (KB) that a task is using. Other processes could potentially share the same memory.  |
| STime               | This counter displays the system time (STime), measured in jiffies, that this process has scheduled in kernel mode. A jiffy corresponds to a unit of CPU time and gets used as a base of measurement. One second comprises 100 jiffies.  |
| Thread Count        | This counter displays the number of threads that are currently grouped with a task. A negative value (-1) indicates that this counter is currently not available. This happens when thread statistics (which includes all performance counters in the Thread object as well as the Thread Count counter in the Process object) are turned off because the system total processes and threads exceeded the default threshold value. |
| Total CPU Time Used | This counter displays the total CPU time in jiffies that the task used in user mode and kernel mode since the start of the task. A jiffy corresponds to a unit of CPU time and gets used as a base of measurement. One second comprises 100 jiffies.   |
| UTime               | This counter displays the time, measured in jiffies, that a task has scheduled in user mode.   |
| VmData              | This counter displays the virtual memory usage of the heap for the task in kilobytes (KB).   |
| VmRSS               | This counter displays the virtual memory (Vm) resident set size (RSS) that is currently in physical memory in kilobytes (KB). This includes the code, data, and stack.   |
| VmSize              | This counter displays the total virtual memory usage for a task in kilobytes (KB). It includes all code, data, shared libraries, and pages that have been swapped out: Virtual Image = swapped size + resident size.   |
| Wchan               | This counter displays the channel (system call) in which the process is waiting.   |

## Processor

The processor object provides information on different processor time usage in percentages. [Table A-14](#) contains information on processor counters.

**Table A-14 Processor**

| Counters           | Counter Descriptions   |
|--------------------|--|
| % CPU Time         | This counter displays the processors share of the elapsed CPU time, excluding idle time, since the last update. This share gets expressed as a percentage of total CPU time.   |
| Idle Percentage    | This counter displays the percentage of time that the processor is in the idle state and did not have an outstanding disk I/O request.   |
| IOwait Percentage  | This counter represents the percentage of time that the processor is in the idle state while the system had an outstanding disk I/O request.   |
| Irq Percentage     | This counter represents the percentage of time that the processor spends executing the interrupt request that is assigned to devices, including the time that the processor spends sending a signal to the computer. |
| Nice Percentage    | This counter displays the percentage of time that the processor spends executing at the user level with nice priority.   |
| Softirq Percentage | This counter represents the percentage of time that the processor spends executing the soft IRQ and deferring task switching to get better CPU performance.  |
| System Percentage  | This counter displays the percentage of time that the processor is executing processes in system (kernel) level.   |
| User Percentage    | This counter displays the percentage of time that the processor is executing normal processes in user (application) level.   |

## System

The System object provides information on file descriptors on your system. [Table A-15](#) contains information on system counters.

**Table A-15 System**

| Counters        | Counter Descriptions   |
|-----------------|--|
| Allocated FDs   | This counter represents the total number of allocated file descriptors.                              |
| Being Used FDs  | This counter represents the number of file descriptors that are currently in use in the system.      |
| Freed FDs       | This counter represents the total number of allocated file descriptors on the system that are freed. |
| Max FDs         | This counter represents the maximum number of file descriptors that are allowed on the system.       |
| Total CPU Time  | This counter represents the total time in jiffies that the system has been up and running.           |
| Total Processes | This counter represents the total number of processes on the system.                                 |
| Total Threads   | This counter represents the total number of threads on the system.                                   |

# TCP

The TCP object provides information on the TCP statistics on your system. [Table A-16](#) contains information on the TCP counters.

**Table A-16**      *TCP*

| Counters      | Counter Description  |
|---------------|--|
| Active Opens  | This counter displays the number of times that the TCP connections made a direct transition to the SYN-SENT state from the CLOSED state.   |
| Attempt Fails | This counter displays the number of times that the TCP connections have made a direct transition to the CLOSED state from either the SYN-RCVD state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYS-RCVD state. |
| Curr Estab    | This counter displays the number of TCP connections where the current state is either ESTABLISHED or CLOSE- WAIT.  |
| Estab Resets  | This counter displays the number of times that the TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.  |
| In Segs       | This counter displays the total number of segments that were received, including those received in error. This count only includes segments that are received on currently established connections.  |
| InOut Segs    | This counter displays the total number of segments that were sent and the total number of segments that were received.   |
| Out Segs      | This counter displays the total number of segments that were sent. This count only includes segments that are sent on currently established connections, but excludes retransmitted octets.  |
| Passive Opens | This counter displays the number of times that TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.  |
| RetransSegs   | This counter displays the total number of segments that were retransmitted because the segment contains one or more previously transmitted octets.   |

# Thread

The Thread object provides a list of running threads on your system. [Table A-17](#) contains information on the Thread counters.

**Table A-17**      *Thread*

| Counters   | Counter Description  |
|------------|--|
| % CPU Time | This counter displays the threads share of the elapsed CPU time since the last update. This counter expresses the share as a percentage of the total CPU time. |
| PID        | This counter displays the threads leader process ID.   |



# Where to Find More Information

## Related Topics

- [Understanding Performance Monitoring](#)
- [Working with Performance Queries](#)

