

Examining Sybase instance statistics

This section includes the following topics:

- [About the Statistics tab](#)
- [How the Statistics tab is structured](#)
- [About the entities you can examine in the Statistics tab](#)
- [How the Statistics tab can help you identify performance problems](#)

About the Statistics tab

The Statistics tab displays statistical information on all Sybase instances. The tab can be used to monitor your system's historical statistical information.

The Statistics tab lets you provide answers to the following types of questions: "Does Sybase have too many or too few Adaptive Server engines?" or "There are a lot of task context switches in the system that affect performance. What is the main reason for those context switches?"

The Statistics tab provides hundreds of raw performance counters. It is possible to view status information in 15 minute intervals or historical information roll-up at a higher summary level.

Examining the ready-to-use set of graphs that display several related counters is more important than examining a single counter. This is now possible using the Statistics tab.

Use the Statistics tab to periodically monitor the health of your system - for example, view the statistics on CPU Utilization or Hit Ratios. Alternatively, you can use the Statistics tab to fully analyze a performance problem reported by the Collector agent.



Statistics tab data is based on the results of the statistics counters that are gathered by ASE. These counters are cleared when using `sp_sysmon` to view statistics counters. If `sp_sysmon` is run, statistics data gathered by Precise for Sybase will not be complete. This problem can be avoided by not running `sp_sysmon` or by using the `noclear` option of `sp_sysmon` (available in Sybase version 12.5.3 and higher).

How the Statistics tab is structured

The Statistics tab displays information on a selected entity and its associated entities. When you open the Statistics tab from the Dashboard tab or the Activity tab, the tab is launched, in-context, and depending upon your selection in the All Sybase Instances table (All or Instance), the information is displayed on the Tier or Instance level. Navigating between these tabs and the Statistics tab will change the chosen time frame. If you open the Statistics tab from another tab, the historical settings (meaning those settings which were selected when you left the tab, such as the last entity you drilled down to) are taken into account and the information displayed the last time you viewed this tab is displayed (similar to clicking the History control and returning to a previous tab).

The selected entity is always reflected in the Tab heading, which serves as a point of orientation. The highest-level entity you can view information for in the Statistics tab is the Tier level. You can select an instance from the Instance list.

The entities displayed in the Association area are related to the selected entity displayed in the Main area.

About the Main area in the Statistics tab

The Main area shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. For example, the Instance entity reports on Disk I/O, Memory and Lock Management.

About the Association area in the Statistics tab

The Association area displays relevant information on the entities associated with the selected entity (displayed in the Tab heading in the Main area) in a table format. For example, it is possible to associate to all Engines or Devices that are related to a specific Instance, by selecting an entity from the Association controls list. Notice that some entities, show additional information on the associated entities in different tabs. The tabs are located above the Association area table. Clicking on a tab displays different table columns showing different information for the associated entities.

If you want to view additional information on an associated entity, drill down to it, by selecting the entity's row. A drilldown affects the entire tab. When you drill down to another entity, the Tab heading reflects the new selection, the Main area displays information on the selected entity, and the Association area displays the entities that are now associated with the newly selected entity.

For example, when you want to view information on a specific device, choose Devices from the Association controls. The Association area changes to display information on the devices associated with the selected Instance. Note that the Tab heading and the Main area remain unchanged.

In the Association area, click the row of the device you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays statistical information on the device entity you drilled down to, and the Association area shows information on the associated performance counters.

About the entities you can examine in the Statistics tab

The Statistics tab displays information on the following different entities in the Association area, as described in the following sections:

- [About the Tier entity](#)

- [About the Instance entity](#)
- [About the Engine entity](#)
- [About the Device entity](#)
- [About the Data Cache entity](#)
- [About the Performance Counter or Counter Instance entity](#)
- [About the Wait Events entity](#)
- [About the Replication Server entity](#)
- [About the Replicate\(d\) Databases entity](#)

About the Tier entity

Choose All Instances from the Instances List to display information on the Tier entity. The Tier entity provides a view of all Sybase instances in the application.

The following views are available:

- Overview
- Instance Grouping

About viewing a summary of statistical information on top instance consumers

The Overview provides statistical information on the top six instance consumers of various resources and enables you to compare them.

The Tier entity shows the top six consumers as follows:

- **Top 6 instances (% Engine Utilization)**. Displays the top six instances for which the engine utilization was highest within a certain time frame.
- **Top 6 instances (Committed Transactions)**. Displays the top six instances with the highest number of committed transactions within a certain time frame.
- **Top 6 instances (Connection Opened – Sum)**. Displays the top six instances with the most connections opened within a certain time frame.

About viewing performance counters according to instance groups

The Instance Grouping view displays performance counters broken down into instance groups (the instance groups are defined by the user).

For an explanation on how to define instance groups, see the *Precise Installation Guide*. The table below describes the information displayed in the Instance Grouping table.

Table 1 Instance Grouping table

Item	Description
Group	Displays the name of the group.
Instances	Displays the number of instances linked to the group.
Engine Utilization	Displays the average Engine Utilization for the instances in the specified group.
Committed Transactions	Displays the SUM number of Committed Transactions for the instances in the specified group.
Connections Opened (Sum)	Displays the sum of opened connections for the instances in the specified group.

About viewing instances associated with an Tier in the Association area

Displays information on the Instances associated with the selected Tier in the Association area. The table below describes the information displayed in the tabs in the Association area.

Table 2 Viewing information on associated instances

Tab	Description
Overview	Displays counters reporting on CPU Usage, Memory Usage and amount of Locked Data.
Network	Displays counters reporting on the number of times an engine polls incoming and outgoing packets and bytes.
Disk I/O	Displays various counters reporting on the amount of disk I/O.
Transactions	Displays counters reporting on the number of rows changed in the instance, grouped by type (insert, update, and delete).

About the Instance entity

The instance entity displays statistical information for the entire instance. Precise for Sybase collects hundreds of counters that are relevant to the entire instance. The Instance entity displays predefined graphs that display several counters in each. Each graph contains counters that have some relation to each other and can indicate where performance problems lie.

The following views are available:

- Overview
- Disk I/O
- Data cache
- Memory
- Network
- Lock management
- Transaction log
- Transaction profile
- HouseKeeper management
- Wait events
- Replication Inbound
- Replication Outbound

About getting an overview of an instance based on statistical data

The Overview displays predefined overtime graphs, which can indicate where performance problems lie. The predefined graphs contain counters reported by Sybase and collected by the Precise for Sybase Collector agent.

The following information is displayed:

- **In Sybase.** Displays resource consumption counters collected by the Precise for Sybase Collector agent, such as Using CPU, I/O Wait, and Lock Wait.
- **CPU Yields.** Displays information on how often the engines yielded to the Operating System.
- **Engine Utilization.** Displays the percentage of time that the Adaptive Server Kernel is busy executing tasks.
- **Connections Opened.** Displays the number of opened connections (max and sum).
- **Task Context Switches - Reasons.** Displays the number of times each Adaptive Server engine switched context from one user task to another. Possible reasons are: a Network Packet sent and received, modify conflicts, last log page, writes group, committed sleeps, and so on.

About viewing Disk I/O activity

The Disk I/O view displays statistics that report on disk I/O activity, such as Percentage of Checks Returning I/O, number of Pages read and written, and so on.

The following information is displayed:

- **Checks Returning I/O Percentage.** Displays the percentage of checks that returned an I/O out of the total number of performed checks.
- **Page Requests.** Displays the number of pages read synchronous or asynchronous and the number of pages written.
- **Avg Disk I/O Returned.** Displays the average number of engine checks that returned an I/O.
- **Disk I/O Delays.** Displays the disk I/O delays due to the following reasons: the engine limit, the OS limit or the ASE limit.

About viewing Data Cache activity

The Data Cache view displays statistics that report on data cache activity, such as Hit Ratio, Utilization, and so on.

The following information is displayed:

- **Hit Ratio.** Displays the ratio between the number of times a page was found in the cache and the number of times caches were searched for a specific page.
- **Utilization.** Displays the number of times a requested page was found and not found in all caches (hits) and needed to be read from disk.
- **Wash Behavior.** Displays the number of buffers that were clean when they passed the wash marker, the number of times that the I/O was already active on a buffer when it entered the wash area, and the number of times that a buffer entered the wash area dirty and was not already in I/O.
- **Dirty Pages Requests.** Displays the average number of pages requested at isolation level 0 for a cache.

About viewing Memory activity

The Memory view displays statistics that report on memory activity, such as number of Memory Requests, Memory Pages, Worker Processes, and so on.

The following information is displayed:

- **Memory Requests.** Displays the number of successful/failed requests for memory allocation for the worker processes.
- **Worker Process.** Requests Displays the number of granted and denied worker process requests.
- **Memory Pages.** Displays the number of times that a new page was allocated/released in the memory.
- **Avg Memory Ever Used by Worker Process.** Displays the average memory used by the worker process.

About viewing Network activity

The Network view displays statistics that report on network activity, such as number of Packets, Bytes, Checks, and so on.

The following information is displayed:

- **Packets.** Displays the total number of packets received and sent.
- **Checks.** Displays the number of times ASE performed blocking and non-blocking network checks.
- **Bytes.** Displays the number of bytes received from the network and sent to the network

- **I/O Delays.** Displays the total number of times the network I/O was delayed.

About viewing Lock Management activity

The Lock Management view displays statistics that report on lock management activity, such as Lock Contention, Lock Waits, and so on.

The following information is displayed:

- **Lock Contention Percentage.** Displays the number of times that there were different lock contentions as a percentage of the total number of lock requests.
- **Lock Waits.** Displays the number of times that there was a lock contention, divided by types.

About viewing Transaction Log activity

The Transaction Log view displays statistics that report on transaction log activity, such as Log Writes, Log Allocations, and so on.

The following information is displayed:

- **ULC Flushes to Log.** Displays the number of times that the ULC was flushed to the transaction log, because of the following reasons: the ULC was full, a transaction ended, the database was changed, or a system transaction occurred within the user transaction.
- **Log Writes.** Displays the number of times the ASE server writes transaction log pages to disk.
- **Log Allocations.** Displays the number of times additional pages were allocated to the transaction log.
- **Avg Writes per Log Page.** Displays the average number of times each log page was written to disk.

About viewing a Transaction Profile

The Transaction Profile view displays statistics that report on the number of Inserts, Updates, and Deletes. The following information is displayed:

- **Inserts.** Displays the number of rows inserted into heap tables, clustered tables or data-only -lock tables.
- **Deletes.** Displays the number of rows deleted on the deferred mode, directly or from DOL tables.
- **Updates.** Displays the number of rows updated in the deferred mode, directly in-place, directly not in-place, directly expensive or deferred mode in DOL tables.
- **DOL Updates.** Displays the number of rows replaced in the DOL tables, rows shrunken while updated in the DOL tables, rows expanded while updated in the DOL tables, rows expanded and forwarded while updated in the DOL tables or rows that were forwarded already and now fit in original page and returned to the original page in the DOL tables.

About viewing Housekeeper activity

The Housekeeper Management view displays statistics that report on housekeeper activity, such as Buffer Cache Wash, Garbage Collection, and so on. The following information is displayed:

- **Buffer Cache Washes.** Displays the number of times that the housekeeper activity performs buffer cache washes and find it clean or dirty.
- **Garbage Collections.** Displays the number of times the housekeeper garbage collection activity checks for space that can be reclaimed.
- **Pages Processed in the Garbage Collection.** Displays the number of pages reclaimed by the housekeeper garbage collection activity.
- **Statistics Updates.** Displays the number of times the housekeeper chore activities checked to see if statistics needed to be written.

About viewing Wait events

The Wait Event view displays statistics on the top ten wait events for the selected instance.

About viewing Replication Inbound view graphs

The Replication Inbound view is a combination of a few predefined graphs related to the inbound processing phase (replication agent); the view helps you identify what may be causing the latency in the replication system. The information displayed in the Instance level is a summation of all the replicated databases of the instance.

Table 3 I/O vs Cmds Sent

Graph	Description
IO vs. Cmds Sent	Highlights The IO vs. Cmds Sent graph shows the number of commands processed by the RepAgent during the given period and the amount of time the RepAgent had to wait while writing to the inbound queue. Each bar represents a time period. The speed of the RepAgent is directly proportional to the speed of ASE processing the network that sends requests coupled with the speed of the Replication Server processing. What to do next • If the amount of I/O wait is significant (compared to the total time each bar represents), then the specific issue delaying the RepAgent must be isolated. The amount of time a RepAgent will have to wait while writing to the inbound queue depends on how much cache is available (exec_sqm_write_request_limit) as well as the values for init_sqm_write_delay/init_sqm_write_max_delay. For more information, check the online help available on the Sybase Web site. • If you see a shift in the trend of the Cmds Sent (fewer commands are sent even though the activity is the same), then perhaps something has changed in the ASE processing. Examine the Activity tab and see what was the ASE doing at that time and what, if anything, was holding it back.

Table 4 Truncation Point Movement

Graph	Description
Truncation Point Movement	Highlights The Truncation Point Movement graph shows the number of times the RepAgent has asked the RS for a new secondary truncation point and then moved the secondary truncation point in the log. If 'Moved' is more than one less than 'Gotten', the likely cause is that a large or open transaction exists from the Replication Server perspective (either it is indeed still open in ASE or the RepAgent has not forwarded the commit record yet). What to do next • Check for an open transaction in ASE. • To determine if the numbers are high - you can gauge the number by dividing the 'Replicated Log Records' (which can be seen in the Log Scan Summary graph) by the RepAgent 'scan_batch_size' configuration parameter and add one to count for the additional move when it reaches the end of the log. The number should be close to what is shown.

Table 5 Network

Graph	Description
Network	Highlights The Network graph shows information related to the network activity. If the Number of Full Packets is a considerable percentage out of the total package sent ($\geq 90\%$), then you may need to consider increasing the send_buffer_size. What to do next • Examine the send_buffer_size configuration parameters in the Objects tab.

Table 6 Log Scan Summary

Graph	Description
Log Scan Summary	Highlights The Log Scan Summary graph is an indicator of how much work the RepAgent is doing and how much information is being sent to the Replication Server. Replicated log records are records converted into LTL and sent to the RS. Not replicated log records are all log records that were scanned but not sent. What to do next • If the replicated log records number is small compared to the total log records scanned, it means that most of the transactions in the log are not aimed for replication — so a large part of the log is scanned needlessly. Consider moving the few tables that need to be replicated to another database so that the total log records scanned will be smaller.

Table 7 Replicated Commands Breakdown

Graph	Description
Replicated Commands Breakdown	Highlights The Replicated Commands Breakdown lists the number and type of commands that were replicated and which most affect the replication. • DDL. Refers to DDL statements that were replicated. Generally this number should be zero, with only minor increases in a Warm Standby when DDL changes are made. • Writetext. Shows how many Writetext operations are being replicated. • DML with Text/Image. Displays how many row images are processed. What to do next • If a large number of text rows are being replicated, you may want to investigate whether a text/image column was inappropriately marked or left at "always_replicate" instead of "replicate_if_changed".

About viewing Replication Outbound view graphs

The Replication Outbound view graphs view is a combination of a few predefined graphs related to the outbound processing phase. This enables you to understand what specific component(s) may be causing the latency in the replication system. The information displayed in the instance level is a summation of all the outbound processing performed for the specified instance by all the Replication servers attached to it.

Table 8 SQM Processing

Graph	Description
SQM Processing	Highlights The SQM Processing graph shows the space usage of SQM queue overtime versus the number of commands written to the queue. The SQM Commands Written indicates the amount of activity performed by the replication server. What to do next • Check the percentage used of the SQM and see if it is sufficient, over-allocated, or near maximum space capacity to accommodate the written commands. • Examine the trend of written commands, with special attention to trend changes. Is the activity evenly distributed, or does it peak at certain times? A deviation from normal trends can be acceptable, but it may also indicate a replication activity that is unauthorized or poorly defined. • If the number of commands are lower, then maybe there is a problem in inbound queue processing or in the network. Check the replication inbound statistics. • If the amount of free space is lower and there is no change in the number of written commands, this may indicate that the subscriber was unable to apply the transaction. For example, an insert statement may have failed because of a duplicate key. If this example occurs, the truncation point will indicate this failed transaction and will not proceed until it is cleared manually.

Table 9 DSI SQT Processing

Graph	Description
DSI SQT Processing	Highlights The DSI SQT Processing graph shows space usage of the outbound queue cache. The Cache size lists the size of the SQT cache. Cached Used shows SQT thread memory usage over time. Each command structure (allocated by an SQT thread) is freed when its transaction context is removed. Consequently, if no transactions are active in SQT, then SQT cache usage is zero. If Cached Used is near the maximum cache size capacity and Trans Removed is constantly greater than zero, you may decide to increase the SQT cache size by increasing dsi_sqt_max_cache_size. Don't increase the cache size if Trans Removed is occasionally greater than zero; this situation usually means that a transaction was removed from the cache because it was greater than the dsi_sqt_max_cache_size. For occasional increases of Trans Removed, increasing the cache size will have the opposite effect because the latency in processing transactions ahead of them will likely result in their being removed. The SQM Read Cached indicates how many 16K blocks of cache were read by the SQM Reader thread. This number should be as high as possible. If it is constantly hitting zero, most likely there is a latency in executing the SQL at the replicate; resulting in the DSI SQT cache to reach its maximum. What to do next • If Cached Used is near the maximum cache size capacity and Trans Removed is constantly greater than zero, you may decide to increase the SQT cache size by increasing dsi_sqt_max_cache_size. • For occasional increases of Trans Removed, increasing the cache size will have the opposite effect because the latency in processing transactions ahead of them will likely result in their being removed. • Examine dsi_sqt_max_cache_size and sqt_max_cache_size. The sqt_max_cache_size should usually be between 4-8 MB. For non-parallel DSI configuration and no Warm StandBy, the dsi_sqt_max_cache_size should be smaller than the sqt_max_cache_size. If set too high (or left at zero as default), the DSI-S thread will continuously try to fill the available DSI SQT cache from the outbound queue - often at the expense of yielding the CPU to the DSI EXEC. Consequently, the optimum size for dsi_sqt_max_cache_size is between 1-2 MB. • For Warm StandBy Configurations, the DSI SQT cache should normally be equal to the SQT cache or higher because it is the DSI SQT thread that is actually sorting the transactions into commit order. • For parallel DSI configurations, the DSI SQT cache should contain double the dsi_max_xacts_in_group transactions for each DSI EXEC thread. The DSI EXEC thread can effectively process many row modifications because the load can be distributed among several available DSIs. This can result in a situation where the DSI transaction rate is higher than the amount of rows read from the outbound queue. In these situations, raising the DSI SQT cache allows the DSI to "read ahead" into the queue and begin preparing transactions before they are needed.

Table 10 DSIE Transactions Processing

Graph	Description
-------	-------------

DSIE Transactions Processing	Highlights The DSIE Transactions Processing graph shows how long it took to process a transaction by a DSI/E thread breakdown to various phases. • FS Map Time. The amount of time used to translate the replicated row functions into SQL commands. If a large amount of time is spent in this area, this may indicate that there are relatively large customized function strings; if true, there may not be any options you can take. However, you may want to check that the STS cache is sized appropriately. • Sent Time. This represents the amount of time spent sending the command batch to the replicate data server. A long lag time may indicate inefficient batching, or a slow response to client applications from the replicate server. • SendRPCTimeAvg. This records the time spent sending RPCs to the RDB. • SendDTTimeAvg. This records time spent sending LOB data to the RDB. • Result Time. This calculated value can be used to determine the amount of time spent processing results from the replicate server. This value also includes the execution time, because RS does very little result processing. Often, these metrics will be among the highest. You need to speed-up the replicate DBMS to improve the RS throughput. • Commit Seq. Time. The amount of time spent waiting to commit. Here too, a high value may indicate a near-serial dsi-serialization_method, such as wait_for_commit. Alternatively, it may indicate a contention within the replicate server, possibly within the rs_threads_group. • Batch Seq. Time. This is the time spent trying to coordinate the sending of the first batch in parallel DSIs. A long time may indicate that the dsi_serialization_method is wait_for_commit and a previous transaction is running a long time, or that the DSI thread is simply too busy to respond to the Batch Sequencing message. • Other Time. The execution time by the replicate database outside of the replication server. What to do next • If Other Time is the largest chunk of time out of the total transaction time, tuning RS will not help. You must do one of the following: tune the replicate database by using parallel DSIs, or use minimal columns/repdefs to reduce the SQL execution time. • If FS Map Time is high, consider rewriting the queries and simplifying them, or work with a stored procedure. • If Commit Seq. Time is high, consider lowering the number of parallel threads.
------------------------------	--

Table 11 DSIE Commands Applied

Graph	Description
DSIE Commands Applied	Highlights The DSIE Commands Applied graph shows how many commands were successfully applied to the target database by DSI/E overtime (relevant for version 15 and higher). It can be used to measure the amount of work imposed by the replication server. You can also compare that number to the SQM Commands Written and check if there is a lag in processing commands inside the replication server. What to do next • Study the trend of applied commands. If it has suddenly dropped, this may indicate a problem in the subscriber ASE. • Go to the Activity tab and examine the resulting batches from replication (rep server program).

About the Engine entity

The Engine entity displays predefined graphs that shows several counters in one. This enables you to immediately view relevant counters, grouped according to topic.

The following views are available:

- Overview
- Network
- Disk I/O

About getting an overview of CPU yields and engine utilization

The Overview displays predefined overtime graphs, which represent CPU Yields and Engine Utilization. The following information is displayed:

- **CPU Yields.** Displays information about how often the engine yielded to the operating system.
- **Engine Utilization.** Displays the percentage of time that the Adaptive Server Engine is busy executing tasks.

About viewing Network activity

The Network view displays predefined overtime graphs, which represent the number of Bytes sent and received, and Packet transportation.

The following information is displayed:

- **Packets.** Displays the total number of packets received and sent by the engine.
- **Bytes.** Displays the total number of bytes received and sent by the engine.

About getting an overview of Disk I/O

The Disk I/O view displays predefined overtime graphs, which represent Checks Returning I/O and Average Returned Disk I/O.

The following information is displayed:

- **Checks Returning I/O.** Displays the percentage of I/O checks that found an I/O request completed.
- **Avg Disk I/Os Returned.** Displays the average number of engine checks that returned an I/O.

About the Device entity

The Device entity displays predefined graphs that report on device activity and performance. The following views are available:

- Overview
- Semaphores

About getting an overview on a Device activity

The Overview displays an overtime graph, which represents Page Requests (read and write) of the device. The following information is displayed:

- **Page Requests.** Displays the number of pages read synchronous or asynchronous and the number of pages that are written.

About viewing Semaphores

The Semaphores view displays an overtime graph, which represents Semaphore Hits and Misses. The following information is displayed:

- **Semaphores.** Displays the number of times that a request for a device semaphore was granted immediately and the number of times the semaphore was busy and the task had to wait for the semaphore to be released.

About the Data Cache entity

The Data Cache entity displays predefined graphs that report on the activity of the Data Cache. The following views are available:

- Overview
- Cache details

About getting an overview on Data Cache activity

The Overview displays statistics that report on the Data Cache Hit Ratio and Utilization. The following information is displayed:

- **Hit Ratio.** Displays the percentage of time a requested page was found in a cache and was found in the wash area.
- **Utilization.** Displays the percentage of searches using this cache, as a percentage of searches across all caches.

About viewing Cache Details

The Cache Details view displays statistics that report on the Data Cache Buffer Wash (Pages) and large I/Os. The following information is displayed:

- **Buffer Wash (Pages).** Displays the number of buffers that were clean when they passed the wash marker, the number of times that I/O was already active on a buffer when entered the wash area, and the number of times that a buffer entered the wash area dirty and was not already in I/O.
- **Large I/O.** Displays the number of times a large I/O was requested for a buffer and the number of times it was denied.

About the Performance Counter or Counter Instance entity

The Performance counter or Counter Instance entity displays statistical information on the selected counter. Precise for Sybase collects counters reported by Sybase. These counters determine which aspects of the objects to monitor. For example, the Sybase Memory Object contains the Mempages_allocated counter.

The Overview displays general details of the selected counter and overtime graph, which represent the counter value. In some of the selected counters, two overtime graphs are displayed: a graph that displays the number of transactions over time and another graph that displays the counter value per committed transaction.

About the Wait Events entity

Available for Sybase versions 12.5.0.3 and higher, the Wait Events entity displays the name of the wait event group, the number of waits experienced by this event group and the summed amount of time accumulated by wait events and summed occurrences.

The following view is displayed:

- Overview

About getting an overview of Wait Events

The Overview displays general details of the selected counter. The following information is available:

- **Event group.** Displays the name of the wait type.
- **Event name.** Provides a description of what is causing the event.

- **Time waited (avg).** Average time waited for wait events, over the selected time period.
- **Times waited graph.** Displays the time waited for wait events vs. the time waited for group events.
- **Wait graphs.** Displays the number of wait events vs. the number of wait events for group events.

About the Replication Server entity

The Replication Server entity displays predefined graphs which represent statistics related to the outbound processing phase and inbound (repagent statistics) processing phase for the selected Replication Server in the Selected time frame.



The Precise for Sybase Replication server collector agent collects statistics only for the Sybase ASEs which are monitored by Precise and use the specified replication server for replicating data. If the same RS also serves other ASEs that are not monitored by Precise or another RDBMS (such as: Oracle, SQL Server, etc.), the statistics of those instances will not be collected and displayed.

The following views are available:

- Replication Inbound
- Replication Outbound

See [About the Instance entity](#).

About the Replicate(d) Databases entity

The Replicate(d) databases entity displays predefined graphs which represent statistics related to the outbound processing phase and inbound (repagent statistics) processing phase for the selected database in the Selected time frame.



The Precise for Sybase Replication server collector agent collects statistics only for the Sybase ASEs that are monitored by Precise and use the specified replication server for replicating data. If the same RS also serves other ASEs that are not monitored by Precise or another RDBMS (such as: Oracle, SQL Server, etc.), the statistics of those instances will not be collected and displayed.

The following views are available:

- Replication Inbound
- Replication Outbound

See [About the Instance entity](#).

How the Statistics tab can help you identify performance problems

To determine whether Sybase is performing optimally there is a need to analyze performance measurements over time. The Statistics tab provides performance counters grouped into several predefined graphs that let you locate performance problems in your system (For example CPU, Paging, I/O, and Network Statistics). In addition, In Sybase states let you correlate between the performance counters and In Sybase states, to determine how an activity in the Sybase instance affects the system.

You can identify a performance problem by doing one or more of the following:

- [Examining the kernel utilization of Sybase over time](#)
- [Examining I/O statistics](#)
- [Examining network activity](#)
- [Examining memory bottlenecks](#)
- [Examining a specific counter over time](#)

Examining the kernel utilization of Sybase over time

You can examine a Sybase instance over time to confirm that the amount of Sybase engines configured in your system is optimal.

The Engine busy utilization graph in the Overview of the Instance entity reports the percentage of time the Sybase Kernel is busy executing tasks (rather than time spent idle). In other words, it measures how busy Sybase engines were during the time slice.

Viewing this graph in correlation with the CPU Yields graph can help you determine whether there are too many engines or if it is likely that response time and throughput can benefit from an additional engine.

The CPU Yields graph reports the number of times the engines yielded back to the Operating System.

If the Engine Busy Utilization graph data indicates a low engine utilization, use CPU Yields to determine whether the Engine Busy Utilization data reflects a truly inactive engine or one that is frequently starved out of CPU time by the Operating System.

When an engine is not busy, it yields to the CPU after a period of time. A high value for CPU Yields indicates that the engine yielded voluntarily.

The table below describes what the utilization status of an engine indicates.

Table 12 Utilization status

Engine Busy	CPU Yields	Status
Low	Low	Engine is CPU starved
Low	High	Engine is inactive
High	Low	Engine is busy
High	High	Engine is busy

To improve kernel utilization, it is recommended to check the Disk I/O and Network view. This is done to determine if the number of checks for I/O or Network that the engine performs, is ideal or are overhead.

For example, in the Disk I/O view, check the Checks Returning I/O Percentage graph. It reports the percentage that a Requested I/O had completed when an engine checked for Disk I/O. For example, if an engine checks for Expected I/O 100,000 times, this average indicates the percentage of time that there actually was I/O pending. If, of those 100,000 checks, I/O was pending 10,000 times, then 10% of the checks were effective, and the other 90% were overhead. However, you should also check the Average Disk I/O Returned graph and how busy the engines were during the sample interval.

If the sample includes idle time, or the I/O traffic is "bursty," it is possible that a high percentage of the checks were returning I/O during the busy period. If the results in this category seem low or high, you can configure the frequency of the checks.

Increasing the amount of time that Adaptive Server engines wait between checks may result in better throughput, because Adaptive Server engines can spend more time processing if they spend less time checking for I/O.

Examining I/O statistics

To examine I/O statistics

1. Open the Overview view of the specific instance entity.
2. Examine the I/O Wait value in the In Sybase graph to detect bottlenecks within the device subsystem.
3. Check for I/O Pacing and I/O Device Contention values in the Task Context Switches - Reasons graph.
4. If increased I/O Wait is the result of an activity in the instance, open the Activity tab to continue your analysis of the application's performance.

I/O Pacing reports how many times an I/O-intensive task switched off an engine, due to exceeding an I/O batch limit. This limit can be configured.

I/O Device Contention reports the number of times a task was put to sleep while waiting for a semaphore for a specific device. If there is significant contention for I/O Device Semaphores, try reducing it by redistributing the tables across devices or by adding devices and moving tables and indexes to them.

To determine which devices are having possible performance problems, drill down to the Device entity located in the Association area. In the Association area table you can see the Page Requests and the Sum of Semaphores for the different devices. Drilling down to one of the devices will enable you to see these counters over time.

In the Page Requests graph you can see the amount of I/O reads and writes.

High read requests may indicate the need for tuning a Select query. High write requests may indicate the need for tuning an Update query.

You can also use this information to check I/O distribution patterns over the disks and to make object placement decisions that can help balance Disk I/O across devices. For example, does the data show that some disks are more heavily used than others? If you see that a large percentage of all I/O went to a specifically named device, you can investigate the tables residing on the device and then determine how to remedy the problem.

Examining network activity

You can examine network traffic over a specified time period to detect bottlenecks. Network performance can be improved by one or more of the following:

- Enhancing the application
- Reducing the amount of data returned
- Configuring the network packet size parameter with a reasonable number that avoids many packets from being sent or received (causing overhead)
- Reconfiguring the network configuration

To examine network activity do one of the following

1. Examine the In Sybase graph and check the Network I/O Wait value.
2. Check the Task Context Switches - Reasons graph to see if Network Packet Sent is a major cause of task switching.
3. If so, check the applications on the problematic instance to see if it's possible to reduce the amount of data returned.

OR

1. Switch to the Network view.
2. Check the ratio between the Packets graph and the Bytes graph to see if the network packet size is used to its maximum capacity.
3. Consider configuring its size.

Examining memory bottlenecks

Lack of memory or a high memory consumption rate can cause paging. A page fault occurs when a process requires code or data that is not in its working set (its space in physical memory), thereby causing the page to be retrieved from disk and written to the physical memory. If there is not enough space in the physical memory, a chosen page needs to be written to disk to free space.

To examine memory bottlenecks

1. Examine the Memory view that reports the number of pages allocated and deallocated in each time slice.
2. If you spot a high value in one of the time slices, open the Activity tab to continue your analysis of the application's performance.
3. Switch to the Memory view to see information about the success and failure of Memory Requests for Worker Processed in the Memory Requests graph.
4. If some of the requests have failed you may need to increase the value of the Memory per Worker Process parameter.
5. Switch to Data Cache view to see information on how effective cache design is. A high Hit Ratio from the Hit Ratio graph and small amount of Page Request Misses from the Utilization graph may indicate effective cache design.
6. If you see a low Hit Ratio and a lot of Page Request Misses, investigate statistics for each of the Data Caches by selecting the Data Cache entity in the Association area.

Examining a specific counter over time

A wide range of performance counters generated by Sybase are collected by the Precise for Sybase collect instance statistics process. These performance counters are sampled every 15 minutes and displayed in this tab over time.

A performance group is a Sybase Performance Group resource, such as a Sybase Lock. Each performance group contains a few counters that determine which performance aspects to monitor. For example, the Sybase Memory Performance group contains counters called Pages Allocated (Sybase counter: mempages_allocated) and Pages Released (Sybase counter: mempages_freed).

Some performance counters have separate instances. An instance refers to a particular occurrence of a counter. For example, the Device-Counters Performance group has counters such as Disk Cache Misses (Sybase counter: p_misses) and Disk Cache Hits (Sybase counter: p_hits), but because several devices exist, each counter is divided into instances.

In relevant counters, the graphs appear with the counter value per transaction in addition to the counter value alone. In those counters, the number of transactions overtime graph appears as well.

To examine a specific counter

1. Select the associated required performance group from the Associations list.
2. Select the required counter.