

## APPENDIX A

# Database Performance and Connection Properties

About this Appendix

This appendix contains information and tables relating to database performance monitoring and database connection parameters.

Contents

<b>Topic</b>	<b>Page</b>
Monitoring database performance	856
Connection properties	868

## Monitoring database performance


Adaptive Server Anywhere provides a set of statistics that can be used to monitor database performance. These are accessible from Sybase Central, and client applications can access the statistics as functions. In addition, these statistics are made available by the server to the Windows NT performance monitor.

This section describes how to access performance and related statistics from client applications, how to monitor database performance using Sybase Central, and how to monitor database performance using the Windows NT performance monitor.


### Obtaining database statistics from a client application

Adaptive Server Anywhere provides a set of system functions that can access information on a per-connection, per-database, or engine-wide basis. The kind of information available ranges from static information such as the server name to detailed performance-related statistics concerning disk and memory usage.

The performance-related statistics are also available, along with some other statistics, for the Windows NT engine and server in the Windows NT Performance Monitor.

 For more information on the Performance Monitor, see "Monitoring database statistics from the Windows NT Performance Monitor" on page 858.

This section illustrates how to use the functions.

 A complete list of system functions and of their properties is provided in the section "System functions" on page 276.

Functions that retrieve system information

The following functions are used to retrieve system information:

- ◆ **property** Provides the value of a given property on an engine-wide basis
- ◆ **connection\_property** Provides the value of a given property for a given connection, or for the current connection by default.
- ◆ **db\_property** Provides the value of a given property for a given database, or for the current database by default.

Supply as an argument only the name of the property you wish to retrieve, the functions return the value for the current server, connection, or database.

Examples

- ◆ The following statement sets a variable named **server\_name** to the name of the current server:

```
SET server_name = property( 'name' )
```

- ◆ The following query returns the user ID for the current connection:

```
SELECT connection_property( 'userid' )
```

- ◆ The following query returns the filename for the root file of the current database:

```
SELECT db_property( 'file' )
```

Improving query efficiency

For maximum efficiency, a client application monitoring database activity should use the **property\_number** function to identify a named property, and then use the number to repeatedly retrieve the statistic. The following set of statements illustrates the process from Interactive SQL:

```
CREATE VARIABLE propnum INT ;
CREATE VARIABLE propval INT ;
SET propnum = property_number( 'cacheread' );
SET propval = property( propnum )
```

Property names obtained in this way are available for many different database statistics, from the number of transaction log page write operations and the number of checkpoints carried out to the number of reads of index leaf pages from the memory cache.

Many of these statistics are made available in graphical form from the Sybase Central database management tool.

## Monitoring database statistics from Sybase Central

You can monitor database statistics from Sybase Central. The Sybase Central Performance Monitor is a graphing tool that can present database statistics as a line graph or a bar graph.

❖ **To start the Sybase Central Performance Monitor:**

- 1 Click the icon for the server you wish to monitor in the left panel.
- 2 Double-click the Statistics folder underneath the server.
- 3 Select a statistic to graph, and drag it to the Performance Monitor icon to start graphing that statistic.

About the performance monitor

The Performance Monitor uses the regular Adaptive Server Anywhere communication mechanisms to gather statistics. This means some statistics (most notably Cache Reads) are affected by Sybase Central. For example, graphing Cache Reads/sec in Sybase Central shows a steady rate, even when nothing apart from the monitoring is going on.

If you have a Windows NT client and server, the Windows NT Performance monitor is preferable since it offers more statistics, and is not intrusive: updating the statistics will not affect the measurements. The extra statistics the Windows NT performance monitor offers deal mainly with network communications—packets received, network buffers used, and so on.


## Monitoring database statistics from the Windows NT Performance Monitor

The Windows NT performance monitor is an application for viewing the behavior of objects such as processors, memory, and applications. Adaptive Server Anywhere provides many statistics for the performance monitor to display.

The Windows NT performance monitor allows unintrusive monitoring of statistics: updating the statistics does not affect the measurements.

❖ **To start the Windows NT performance monitor:**

- 1 Open the Administrative Tools program group.
- 2 Double click Performance Monitor.

 For information about Performance Monitor, see the Performance Monitor online Help.

❖ **To display database statistics:**

- 1 With Performance Monitor running, select Add To Chart from the Edit menu, or click the Plus sign on the toolbar.  
The Add To Chart dialog appears.
- 2 From the Object list, select Adaptive Server Anywhere.  
The Counter list then displays a list of the statistics provided.
- 3 From the Counter list, click a statistic to be displayed.
- 4 For a description of the selected counter, click Explain.
- 5 To display the counter, click Add.

- 6 When you have selected all the counters you wish to display, click Done.

Performance Monitor statistics

The statistics made available for Performance Monitor by Adaptive Server Anywhere are as follows:

Statistic	Description
Active Requests	Active Requests is the number of engine threads that are currently handling a request.
Asynchronous Reads/sec	Asynchronous Reads/sec is the rate at which pages are being read asynchronously from disk.
Asynchronous Writes/sec	Asynchronous Writes/sec is the rate at which pages are being written asynchronously to disk.
Bytes Received/sec	Bytes Received/sec is the rate at which network data (in bytes) are being received.
Bytes Transmitted/sec	Bytes Transmitted/sec is the rate at which bytes are being transmitted over the network.
Cache Hits/sec	Cache Hits/sec is the rate at which database page lookups are satisfied by finding the page in the cache.
Cache Index Internal Reads/sec	Cache Index Internal Reads/sec is the rate at which index internal-node pages are being read from the cache.
Cache Index Leaf Reads/sec	Cache Index Leaf Reads/sec is the rate at which index leaf pages are being read from the cache.
Cache Reads/sec	Cache Reads/sec is the rate at which database pages are being looked up in the cache.
Cache Table Reads/sec	Cache Table Reads/sec is the rate at which table pages are being read from the cache.
Cache Writes/sec	Cache Writes/sec is the rate at which pages in the cache are being modified (in pages/sec).
Checkpoint Flushes/sec	Checkpoint Flushes/sec is the rate at which ranges of adjacent pages are being written out during a checkpoint.
Checkpoint Log/sec	Checkpoint Log/sec is the rate at which the transaction log is being checkpointed.
Checkpoint Urgency	Checkpoint Urgency is expressed as a percentage.
Checkpoints/sec	Checkpoints/sec is the rate at which checkpoints are being performed.
Commit files/sec	Commit files/sec is the rate at which the engine is forcing a flush of the disk cache. On Windows NT and NetWare platforms, the disk cache does not need to be flushed because unbuffered (direct) I/O is used.

Statistic	Description
Commits/sec	Commits/sec is the rate at which Commit requests are being handled.
Context Switch Checks/sec	Context Switch Checks/sec is the rate at which the current engine thread is volunteering to give up the CPU to another engine thread.
Context Switches/sec	Context Switches/sec is the rate at which the current engine thread is being changed.
Continue Requests/sec	Continue Requests/sec is the rate at which "CONTINUE" requests are being issued to the engine.
Corrupt Packets/sec	Corrupt Packets/sec is the rate at which corrupt network packets are being received.
Current IO	Current IO is the current number of file I/Os issued by the engine which have not yet completed.
Current Reads	Current Reads is the current number of file reads issued by the engine which have not yet completed.
Current Writes	Current Writes is the current number of file writes issued by the engine which have not yet completed.
Cursor	Cursor is the number of declared cursors that are currently being maintained by the engine.
Dirty Pages	Dirty Pages is the number of pages in the cache which must be written out and which do not belong to temporary files.
Disk Index Internal Reads/sec	Disk Index Internal Reads/sec is the rate at which index internal-node pages are being read from disk.
Disk Index Leaf Reads/sec	Disk Index Leaf Reads/sec is the rate at which index leaf pages are being read from disk.
Disk Reads/sec	Disk Reads/sec is the rate at which pages are being read from file.
Disk SyncReads/sec	Disk SyncReads/sec is the rate at which pages are being read synchronously from disk.
Disk SyncWrite Other/sec	Disk SyncWrite Other/sec is the rate at which pages are being written synchronously to disk for a reason not covered by other "Disk SyncWrites ____/sec" counters.
Disk SyncWrites Checkpoint/sec	Disk SyncWrites Checkpoint/sec is the rate at which pages are being written synchronously to disk for a checkpoint.
Disk SyncWrites Extend/sec	Disk SyncWrites Extend/sec is the rate at which pages are being written synchronously to disk while extending a database file.
Disk SyncWrites Free Current/sec	Disk SyncWrites Free Current/sec is the rate at which pages are being written synchronously to disk to free a page that

Statistic	Description
	cannot remain in the in-memory free list.
Disk SyncWrites Free Push/sec	Disk SyncWrites Free Push/sec is the rate at which pages are being written synchronously to disk to free a page that can remain in the in-memory free list.
Disk SyncWrites Log/sec	Disk SyncWrites Log/sec is the rate at which pages are being written synchronously to the transaction log.
Disk SyncWrites Rollback/sec	Disk SyncWrites Rollback/sec is the rate at which pages are being written synchronously to the rollback log.
Disk SyncWrites/sec	Disk SyncWrites/sec is the rate at which pages are being written synchronously to disk. It is the sum of all the other "Disk SyncWrites ____/sec" counters.
Disk Table Reads/sec	Disk Table Reads/sec is the rate at which table pages are being read from disk.
Disk Waitreads/sec	Disk Waitreads/sec is the rate at which the engine is waiting synchronously for the completion of a read IO operation which was originally issued as an asynchronous read. Waitreads often occur due to cache misses on systems that support asynchronous IO.
Disk Waitwrites/sec	Disk Waitwrites/sec is the rate at which the engine is waiting synchronously for the completion of a write IO operation which was originally issued as an asynchronous write.
Disk Writes/sec	Disk Writes/sec is the rate at which modified pages are being written to disk.
Dropped Packets/sec	Dropped Packets/sec is the rate at which network packets are being dropped due to lack of buffer space.
Extend Database/sec	Extend Database/sec is the rate (in pages/sec) at which the database file is being extended.
Extend Temporary File/sec	Extend Temporary File/sec is the rate (in pages/sec) at which temporary files are being extended.
Free Buffers	Number of free network buffers.
Freelist Write Current/sec	Freelist Write Current/sec is the rate at which pages that cannot remain in the in-memory free list are being freed.
Freelist Write Push/sec	Freelist Write Push/sec is the rate at which pages that can remain in the in-memory free list are being freed.
Full compares/sec	Full compares/sec is the rate at which comparisons beyond the hash value in an index must be performed.
IO to Recover	IO to Recover is the estimated number of IO operations required to recover the database.
Idle Active/sec	Idle Active/sec is the rate at which the engine's idle thread

Statistic	Description
	becomes active to do idle writes, idle checkpoints, etc.
Idle Checkpoints/sec	Idle Checkpoints/sec is the rate at which checkpoints are completed by the engine's idle thread. An idle checkpoint occurs whenever the idle thread writes out the last dirty page in the cache.
Idle Waits/sec	Idle Waits/sec is the number of times per second that the server goes idle waiting for IO completion or a new request.
Idle Writes/sec	Idle Writes/sec is the rate at which disk writes are being issued by the engine's idle thread.
Index Fills	Index Fills is the number of times a new temporary merge index is created.
Index Merges	Index Merges is the number of times a temp index has been merged into a main index
Index adds/sec	Index adds/sec is the rate at which entries are being added to indexes.
Index lookups/sec	Index lookups/sec is the rate at which entries are being looked up in indexes.
Lock Table Pages	Lock Table Pages is the number of pages used to store lock information.
Main Heap Pages	Main Heap Pages is the number of pages used for global engine data structures.
Map Pages	Map Pages is the number of map pages used for accessing the lock table, frequency table, and table layout.
Maximum IO	Maximum IO is the maximum value that "Current IO" has reached.
Maximum Reads	Maximum Reads is the maximum value that "Current Reads" has reached.
Maximum Writes	Maximum Writes is the maximum value that "Current Writes" has reached.
Multi-packets Received/sec	Multi-packets Received/sec is the rate at which multi-packet deliveries are being received.
Multi-packets Transmitted/sec	Multi-packets Transmitted/sec is the rate at which multi-packet deliveries are being transmitted.
Open cursors	Open cursors is the number of open cursors that are currently being maintained by the engine.
Packets Received/sec	Packets Received/sec is the rate at which network packets are being received.
Packets Transmitted/sec	Packets Transmitted/sec is the rate at which network packets are being transmitted.



Statistic	Description
Page Relocations/sec	Page Relocations/sec is the rate at which relocatable heap pages are being read from the temporary file.
Pending requests/sec	Pending requests/sec is the rate at which the engine is detecting the arrival of new requests.
Ping1/sec	Ping1/sec is the rate at which ping requests which go all the way down into the engine are serviced.
Ping2/sec	Ping2/sec is the rate at which ping requests which are turned around at the top of the protocol stack are serviced.
Procedure Pages	Procedure Pages is the number of relocatable heap pages used for procedures.
Read Hints Used/sec	Read Hints Used/sec is the rate at which page-read operations are being satisfied immediately from cache thanks to an earlier read hint.
Read Hints/sec	A read hint is an asynchronous read operation for a page that the server is likely to need soon. Read Hints/sec is the rate at which such read operations are being issued.
Recovery Urgency	Recovery Urgency is expressed as a percentage.
Redo Free Commits/sec	A "Redo Free Commit" occurs when a commit of the transaction (redo) log is requested but the log has already been written (so the commit was done for "free").
Redo Rewrites/sec	Redo Rewrites/sec is the rate at which pages that were previously written to the transaction log (but were not full) are being written to the transaction log again (but with more data added).
Redo Writes/sec	Redo Writes/sec is the rate at which pages are being written to the transaction (redo) log.
Relocatable Heap Pages	Relocatable Heap Pages is the number of pages used for relocatable heaps (cursors, statements, procedures, triggers, views, etc.).
Remoteput Wait/sec	Remoteput Wait/sec is the rate at which the communication link must wait because it does not have buffers available to send information. This statistic is collected for NetBIOS (both sessions and datagrams) and IPX protocols only.
Requests/sec	Requests/sec is the rate at which the engine is being entered to allow it to handle a new request or continue processing an existing request.
Rereads Queued/sec	A reread occurs when a read request for a page is received by the database IO subsystem while an asynchronous read IO operation has been posted to the operating system but has not completed. Rereads Queued/sec is the rate at which this condition is occurring.

Statistic	Description
Rereceived Packets/sec	Rereceived Packets/sec is the rate at which duplicate network packets are being received.
Retransmitted Packets/sec	Retransmitted Packets/sec is the rate at which network packets are being retransmitted.
Rollback Log Pages	Rollback Log Pages is the number of pages in the rollback log.
Rollback/sec	Rollback/sec is the rate at which Rollback requests are being handled.
Adaptive Server Anywhere	The Adaptive Server Anywhere object provides information about the database server.
Sends Failed/sec	Sends Failed/sec is the rate at which the underlying protocol(s) failed to send a packet.
Statement	Statement is the number of prepared statements that are currently being maintained by the engine.
TotalBuffers	TotalBuffers number of network buffers.
Trigger Pages	Trigger Pages is the number of relocatable heap pages used for triggers.
Unscheduled requests	Unscheduled requests is the number of requests that are currently queued up waiting for an available engine thread.
View Pages	View Pages is the number of relocatable heap pages used for views.
Voluntary blocks/sec	Voluntary blocks/sec is the rate at which engine threads voluntarily block on pending disk IO.
Waitread Full Compare/sec	Waitread Full Compare/sec is the rate at which read requests associated with a full comparison (a comparison beyond the hash value in an index) must be satisfied by a synchronous read operation.
Waitread Optimizer/sec	Waitread Optimizer/sec is the rate at which read requests posted by the optimizer must be satisfied by a synchronous read operation.
Waitread Other/sec	Waitread Other/sec is the rate at which read requests from other sources must be satisfied by a synchronous read operation.
Waitread SysConnection/sec	Waitread SysConnection/sec is the rate at which read requests posted from the system connection must be satisfied by a synchronous read operation. The system connection is a special connection that is used as the context before a connection is made and for operations performed outside of a client connection.
Waitread	Waitread Temporary Table/sec is the rate at which read

<b>Statistic</b>	<b>Description</b>
Temporary Table/sec	requests for a temporary table must be satisfied by a synchronous read operation.





# Connection properties

The following table lists properties available for each connection.

## Examples

❖ **To retrieve the value of a connection property:**

- ◆ Use the **connection\_property** system function: The following statement returns the number of pages that have been read from file by the current connection.

```
select connection_property ( 'DiskRead' )
```

❖ **To retrieve the values of all connection properties:**

- ◆ Use the **sa\_conn\_properties** system procedure:

```
call sa_conn_properties
```

A separate row is displayed for each connection.

## Descriptions

Property	Description
<b>AllowNullsByDefault</b>	"ALLOW_NULLS_BY_DEFAULT option" on page 140
<b>AnsiBlanks</b>	"ANSI_BLANKS option" on page 140
<b>AnsiCloseCursorsOnRoll Back</b>	"ANSI_CLOSE_CURSORS_ON_ROLLBACK option" on page 141
<b>AnsiIntegerOverflow</b>	"ANSI_INTEGER_OVERFLOW option" on page 141
<b>AnsiPermissions</b>	"ANSI_PERMISSIONS option" on page 142
<b>AnsiNull</b>	"ANSINULL option" on page 142
<b>Async2Read</b>	The number of rereads. A reread occurs when a read request for a page is received by the database IO subsystem while an asynchronous read IO operation has been posted to the operating system but has not completed.
<b>AsyncRead</b>	The number of pages that have been read asynchronously from disk.
<b>AsyncWrite</b>	The number of pages that have been written asynchronously to disk.
<b>Automatic_timestamp</b>	"AUTOMATIC_TIMESTAMP option" on page 144
<b>Background_priority</b>	"BACKGROUND_PRIORITY option" on page 144

Property	Description
<b>BlockedOn</b>	If the current connection is not blocked this is zero. If it is blocked, the connection number on which the connection is blocked due to a locking conflict.
<b>CacheHits</b>	The number of successful reads of the cache
<b>CacheRead</b>	The number of database pages that have been looked up in the cache.
<b>CacheReadIndInt</b>	The number of index internal-node pages that have been read from the cache.
<b>CacheReadIndLeaf</b>	The number of index leaf pages that have been read from the cache.
<b>CacheReadTable</b>	The number of table pages that have been read from the cache.
<b>CacheWrite</b>	The number of pages in the cache that have been modified.
<b>Chained</b>	"CHAINED option" on page 146
<b>Checkpoint_time</b>	"CHECKPOINT_TIME option" on page 147
<b>CIS_option</b>	Reserved
<b>Close_on_EndTrans</b>	"CLOSE_ON_ENDTRANS option" on page 147
<b>Commit</b>	The number of Commit requests that have been handled.
<b>CommLink</b>	The communication link for the connection. This is one of the network protocols supported by Adaptive Server Anywhere, or is "local" for a same-machine connection.
<b>CommProtocol</b>	Returns CmdSeq for Adaptive Server Anywhere protocol (ODBC and Embedded SQL) or TDS for Open Client and jConnect connections.
<b>Continue_after_raiserror</b>	"CONTINUE_AFTER_RAISE_ERROR option" on page 148
<b>Conversion_error</b>	"CONVERSION_ERROR option" on page 149
<b>Cooperative_commit_timeout</b>	"COOPERATIVE_COMMIT_TIMEOUT option" on page 150
<b>Cooperative_commits</b>	"COOPERATIVE_COMMITS option" on page 149
<b>CurrTaskSwitch</b>	The number of current request context switches.
<b>Cursors</b>	The number of declared cursors that are currently being maintained by the server.
<b>CursorOpen</b>	The number of open cursors that are currently

Property	Description
	being maintained by the server.
<b>Date_format</b>	"DATE_FORMAT option" on page 150
<b>Date_order</b>	"DATE_ORDER option" on page 152
<b>DBNumber</b>	The ID number of the database.
<b>Default_timestamp_increment</b>	"DEFAULT_TIMESTAMP_INCREMENT option" on page 152
<b>Delayed_commit_timeout</b>	"DELAYED_COMMIT_TIMEOUT option" on page 153
<b>Delayed_commite</b>	"DELAYED_COMMITS option" on page 153
<b>DiskRead</b>	The number of pages that have been read from disk.
<b>DiskReadIndInt</b>	The number of index internal-node pages that have been read from disk.
<b>DiskReadIndLeaf</b>	The number of index leaf pages that have been read from disk.
<b>DiskReadTable</b>	The number of table pages that have been read from disk.
<b>DiskSyncRead</b>	The number of pages that have been read synchronously from disk.
<b>DiskSyncWrite</b>	The number of pages that have been written synchronously to disk. It is the sum of all the other "SyncWrite" counters.
<b>DiskWaitRead</b>	The number of times the server has waited synchronously for the completion of a read IO operation which was originally issued as an asynchronous read. Waitreads often occur due to cache misses on systems that support asynchronous IO.
<b>DiskWaitWrite</b>	The number of times the server has waited synchronously for the completion of a write IO operation which was originally issued as an asynchronous write.
<b>DiskWrite</b>	The number of modified pages that have been written to disk.
<b>Divide_by_zero_error</b>	"DIVIDE_BY_ZERO_ERROR option" on page 155
<b>Escape_character</b>	"ESCAPE_CHARACTER option" on page 155
<b>Fire_triggers</b>	"FIRE_TRIGGERS option" on page 156



Property	Description
<b>Float_as_double</b>	"FLOAT_AS_DOUBLE option" on page 156
<b>FullCompare</b>	The number of comparisons that have been performed beyond the hash value in an index.
<b>HintUsed</b>	The number of page-read operations that have been satisfied immediately from cache thanks to an earlier read hint.
<b>IndAdd</b>	The number of entries that have been added to indexes.
<b>IndLookup</b>	The number of entries that have been looked up in indexes.
<b>Java_heap_size</b>	"JAVA_HEAP_SIZE option" on page 160
<b>Java_namespace_size</b>	"JAVA_NAMESPACE_SIZE option" on page 160
<b>LastIdle</b>	The number of ticks between requests.
<b>LastReqTime</b>	The time at which the last request for the specified connection started.
<b>LockTablePages</b>	The number of pages used to store lock table info.
<b>LogFreeCommit</b>	The number of Redo Free Commits. A "Redo Free Commit" occurs when a commit of the transaction log is requested but the log has already been written (so the commit was done for "free").
<b>Login_mode</b>	"LOGIN_MODE option" on page 161
<b>Login_procedure</b>	"LOGIN_PROCEDURE option" on page 161
<b>LogRewrite</b>	The number of pages that were previously written to the transaction log (but were not full) that have been written to the transaction log again (but with more data added).
<b>LogWrite</b>	The number of pages that have been written to the transaction log.
<b>Max_cursor_count</b>	"MAX_CURSOR_COUNT option" on page 162
<b>Max_statement_count</b>	"MAX_STATEMENT_COUNT option" on page 162
<b>Name</b>	The name of the server.
<b>Nearest_century</b>	"NEAREST_CENTURY option" on page 163
<b>NodeAddress</b>	The node for the client in a client/server connection.
<b>Non_keywords</b>	"NON_KEYWORDS option" on page 164
<b>Number</b>	The ID number of the connection.

Property	Description
<b>Percent_as_comment</b>	"PERCENT_AS_COMMENT option" on page 167
<b>Port</b>	An application-specific number for each client machine, identifying the connection port.
<b>Precision</b>	"PRECISION option" on page 168
<b>Prefetch</b>	"PREFETCH option" on page 169
<b>PrepStmt</b>	The number of prepared statements that are currently being maintained by the server.
<b>ProcessTime</b>	The time since the start of the connection.
<b>Query_plan_on_open</b>	"QUERY_PLAN_ON_OPEN option" on page 169
<b>Quoted_identifier</b>	"QUOTED_IDENTIFIER option" on page 170
<b>ReadHint</b>	The number of read hints. A read hint is an asynchronous read operation for a page that the database server is likely to need soon.
<b>Recovery_time</b>	"RECOVERY_TIME option" on page 170
<b>Replicate_all</b>	"REPLICATE_ALL option" on page 171
<b>ReqType</b>	A string for the type of the last request.
<b>RI_Trigger_time</b>	"RI_Trigger_time option" on page 172
<b>RIbk</b>	The number of Rollback requests that have been handled.
<b>RollbackLogPages</b>	The number of pages in the rollback log
<b>Row_counts</b>	"ROW_COUNTS option" on page 170
<b>Scale</b>	"SCALE option" on page 172
<b>SQL_flagger_error_level</b>	"SQL_FLAGGER_ERROR_LEVEL option" on page 172
<b>SQL_flagger_warning_level</b>	"SQL_FLAGGER_WARNING_LEVEL option" on page 173
<b>String_rtruncation</b>	"STRING_RTRUNCATION option" on page 173
<b>SyncWriteChkpt</b>	The number of pages that have been written synchronously to disk for a checkpoint.
<b>SyncWriteExtend</b>	The number of pages that have been written synchronously to disk while extending a database file.
<b>SyncWriteFreeCurr</b>	The number of pages that have been written synchronously to disk to free a page that cannot remain in the in-memory free list.
<b>SyncWriteFreePush</b>	The number of pages that have been written

Property	Description
	synchronously to disk to free a page that can remain in the in-memory free list.
<b>SyncWriteLog</b>	The number of pages that have been written synchronously to the transaction log.
<b>SyncWriteRlbk</b>	The number of pages that have been written synchronously to the rollback log.
<b>SyncWriteUnkn</b>	The number of pages that have been written synchronously to disk for a reason not covered by other "SyncWrite" counters.
<b>TaskSwitch</b>	The number of times the current server thread has been changed.
<b>TaskSwitchCheck</b>	The number of times the current server thread has volunteered to give up the CPU to another server thread.
<b>Thread_count</b>	"THREAD_COUNT option" on page 174
<b>Time_format</b>	"TIME_FORMAT option" on page 175
<b>Timestamp_format</b>	"TIMESTAMP_FORMAT option" on page 175
<b>Tsql_hex_constant</b>	"TSQL_HEX_CONSTANT option" on page 177
<b>Tsql_variables</b>	"TSQL_VARIABLES option" on page 177
<b>UncommittedOp</b>	The number of uncommitted operations
<b>Userid</b>	The user ID for the connection.
<b>VoluntaryBlock</b>	The number of server threads that have voluntarily blocked on pending disk IO.
<b>Wait_for_commit</b>	"WAIT_FOR_COMMIT option" on page 178

Property	Description
<b>WaitReadCmp</b>	The number of read requests associated with a full comparison (a comparison beyond the hash value in an index) that must be satisfied by a synchronous read operation.
<b>WaitReadOpt</b>	The number of read requests posted by the optimizer that must be satisfied by a synchronous read operation.
<b>WaitReadSys</b>	The number of read requests posted from the system connection that must be satisfied by a synchronous read operation. The system connection is a special connection, used as the context before a connection is made and for operations performed outside of any client connection.
<b>WaitReadTemp</b>	The number of read requests for a temporary table that must be satisfied by a synchronous read operation.
<b>WaitReadUnkn</b>	The number of read requests from other sources that must be satisfied by a synchronous read operation.

**Compatibility**

The following table shows the Adaptive Server Enterprise system functions and their status in Adaptive Server Anywhere:

Function	Status
<b>Col_length</b>	Implemented
<b>Col_name</b>	Implemented
<b>Db_id</b>	Implemented
<b>Db_name</b>	Implemented
<b>Index_col</b>	Implemented
<b>Object_id</b>	Implemented
<b>Object_name</b>	Implemented
<b>Proc_role</b>	Always returns 0
<b>Show_role</b>	Always returns NULL
<b>Tsequal</b>	Implemented
<b>User_id</b>	Implemented
<b>User_name</b>	Implemented
<b>Suser_id</b>	Implemented

Function	Status
<b>Suser_name</b>	Implemented
<b>Datalength</b>	Implemented
<b>Curunreservedpgs</b>	Not implemented
<b>Data_pgs</b>	Not implemented
<b>Host_id</b>	Not implemented
<b>Host_name</b>	Not implemented
<b>Lct_admin</b>	Not implemented
<b>Reserved_pgs</b>	Not implemented
<b>Rowcnt</b>	Not implemented
<b>Used_pgs</b>	Not implemented
<b>Valid_name</b>	Not implemented
<b>Valid_user</b>	Not implemented

Notes

- ◆ Some of the system functions are implemented in Adaptive Server Anywhere as stored procedures.
- ◆ The **db\_id**, **db\_name**, and **datalength** functions are implemented as built-in functions.

The implemented system functions are described in the following table.

System function	Description
<b>Col_length</b> ( <i>table-name</i> , <i>column-name</i> )	Returns the defined length of column
<b>Col_name</b> ( <i>table-id</i> , <i>column-id</i> [, <i>database-id</i> ] )	Returns the column name
<b>Db_id</b> ( [ <i>database-name</i> ] )	Returns the database ID number
<b>Db_name</b> ( [ <i>database-id</i> ] )	Returns the database name
<b>Index_col</b> ( <i>table-name</i> , <i>index-id</i> , <i>key_#</i> [, <i>userid</i> ] )	Returns the name of the indexed column
<b>Object_id</b> ( <i>object-name</i> )	Returns the object ID
<b>Object_name</b> ( <i>object-id</i> [, <i>database-id</i> ] )	Returns the object name
<b>Tsequal</b> ( <i>timestamp</i> , <i>timestamp2</i> )	Compares timestamp values to prevent update on a row that has been modified since it was selected

System function	Description
User_id( [ user-name ] )	Returns an integer user identification number. This does not return the Adaptive Server Anywhere user ID
User_name( [user-id] )	Returns the user ID (user name in Adaptive Server Enterprise)
Suser_id( [user-name] )	Returns an integer user identification number
Suser_name( [user-id] )	Returns the user ID (server user name in Adaptive Server Enterprise)
Datalength( expression )	Returns the length of the expression, in bytes

## Properties available for the server

The following table lists properties that apply across the server as a whole.

### Examples

❖ **To retrieve the value of a server property:**

- ◆ Use the property system function: For example: the following statement returns the number of cache pages being used to hold the main heap:

```
select property ( 'MainHeapPages' )
```

❖ **To retrieve the values of all server properties:**

- ◆ Use the `sa_eng_properties` system procedure:

```
call sa_eng_properties
```

### Descriptions

Property	Description
ActiveReq	The number of server threads that are currently handling a request.
Async2Read	The number of rereads. A reread occurs when a read request for a page is received by the database IO subsystem while an asynchronous read IO operation has been posted to the operating system but has not completed.
AsyncRead	The number of pages that have been read asynchronously from disk.
AsyncWrite	The number of pages that have been written asynchronously to disk.

<b>Property</b>	<b>Description</b>
<b>CacheHits</b>	The number of database page lookups satisfied by finding the page in the cache.
<b>CacheRead</b>	The number of database pages that have been looked up in the cache.
<b>CacheReadIndInt</b>	The number of index internal-node pages that have been read from the cache.
<b>CacheReadIndLeaf</b>	The number of index leaf pages that have been read from the cache.
<b>CacheReadTable</b>	The number of table pages that have been read from the cache.
<b>CacheWrite</b>	The number of pages in the cache that have been modified.
<b>CheckpointUrgency</b>	The time that has elapsed since the last checkpoint, as a percentage of the checkpoint time setting of the database.
<b>Chkpt</b>	The number of checkpoints that have been performed.
<b>ChkptFlush</b>	The number of ranges of adjacent pages written out during a checkpoint.
<b>ChkptPage</b>	The number of transaction log checkpoints.
<b>CommitFile</b>	The number of times the server has forced a flush of the disk cache. On Windows NT and NetWare platforms, the disk cache does not need to be flushed if unbuffered (direct) IO is used.
<b>CompanyName</b>	The name of the company owning this software.
<b>ConnsDisabled</b>	The number of connections disabled
<b>ContReq</b>	The number of "CONTINUE" requests issued to the server.
<b>CurrIO</b>	The current number of file IOs that were issued by the server but have not yet completed.
<b>CurrRead</b>	The current number of file reads that were issued by the server but have not yet completed.
<b>CurrWrite</b>	The current number of file writes that were issued by the server but have not yet completed.
<b>DirtyPages</b>	The number of dirty pages in cache
<b>DiskRead</b>	The number of pages that have been read from disk.
<b>DiskReadIndInt</b>	The number of index internal-node pages that have been read from disk.
<b>DiskReadIndLeaf</b>	The number of index leaf pages that have been read

Property	Description
	from disk.
<b>DiskReadTable</b>	The number of table pages that have been read from disk.
<b>DiskSyncRead</b>	The number of pages that have been read synchronously from disk.
<b>DiskSyncWrite</b>	The number of pages that have been written synchronously to disk. It is the sum of all the "SyncWrite" counters.
<b>DiskWaitRead</b>	The number of times the server has waited synchronously for the completion of a read IO operation which was originally issued as an asynchronous read. Waitreads often occur due to cache misses on systems that support asynchronous IO.
<b>DiskWaitWrite</b>	The number of times the server has waited synchronously for the completion of a write IO operation which was originally issued as an asynchronous write.
<b>DiskWrite</b>	The number of modified pages that have been written to disk.
<b>ExtendDBWrite</b>	The number of pages by which the database file has been extended.
<b>ExtendTempWrite</b>	The number of pages by which temporary files have been extended.
<b>FreeWriteCurr</b>	The number of pages freed, of those that cannot remain in the in-memory free list.
<b>FreeWritePush</b>	The number of pages freed, of those that can remain in the in-memory free list.
<b>FullCompare</b>	The number of comparisons that have been performed beyond the hash value in an index.
<b>HintUsed</b>	The number of page-read operations that have been satisfied immediately from cache thanks to an earlier read hint.
<b>IdleCheck</b>	The number of times that the server's idle thread has become active to do idle writes, idle checkpoints, and so on.
<b>IdleChkpt</b>	The number of checkpoints completed by the server's idle thread. An idle checkpoint occurs whenever the idle thread writes out the last dirty page in the cache.
<b>IdleChkTime</b>	The number of 100ths of a second spent checkpointing during idle IO.



<b>Property</b>	<b>Description</b>
<b>IdleWrite</b>	The number of disk writes that have been issued by the server's idle thread.
<b>IndAdd</b>	The number of entries that have been added to indexes.
<b>IndLookup</b>	The number of entries that have been looked up in indexes.
<b>LegalCopyright</b>	The copyright string for the software.
<b>LegalTrademarks</b>	Trademark information for the software.
<b>LockTablePages</b>	The number of pages used to store lock information
<b>LogFreeCommit</b>	The number of Redo Free Commits. A "Redo Free Commit" occurs when a commit of the transaction log is requested but the log has already been written (so the commit was done for "free").
<b>LogRewrite</b>	The number of pages that were previously written to the transaction log (but were not full) that have been written to the transaction log again (but with more data added).
<b>LogWrite</b>	The number of pages that have been written to the transaction log.
<b>MainHeapPages</b>	The number of pages used for global server data structures.
<b>MapPages</b>	The number of map pages used for accessing the lock table, frequency table, and table layout.
<b>MaxIO</b>	The maximum value that Curr IO has reached.
<b>MaxRead</b>	The maximum value that Curr Read has reached.
<b>MaxWrite</b>	The maximum value that Curr Write has reached.
<b>Name</b>	The name of the server.
<b>PageRelocations</b>	The number of relocatable heap pages that have been read from the temporary file.
<b>PendingReq</b>	The number of new requests detected by the server.
<b>Platform</b>	The operating system on which the software is running.
<b>ProcedurePages</b>	The number of relocatable heap pages that have been used for procedures.
<b>ProductName</b>	The name of the software.
<b>ProductVersion</b>	The version of the software being run.
<b>ReadHint</b>	The number of read hints. A read hint is an asynchronous read operation for a page that the database server is likely to need soon.

Property	Description
<b>RelocatableHeapPages</b>	The number of pages used for relocatable heaps (cursors, statements, procedures, triggers, views, etc.).
<b>Req</b>	The number of times the server has been entered to allow it to handle a new request or continue processing an existing request.
<b>RollbackLogPages</b>	The number of pages in the rollback log.
<b>SyncWriteChkpt</b>	The number of pages that have been written synchronously to disk for a checkpoint.
<b>SyncWriteExtend</b>	The number of pages that have been written synchronously to disk while extending a database file.
<b>SyncWriteFreeCurr</b>	The number of pages that have been written synchronously to disk to free a page that cannot remain in the in-memory free list.
<b>SyncWriteFreePush</b>	The number of pages that have been written synchronously to disk to free a page that can remain in the in-memory free list.
<b>SyncWriteLog</b>	The number of pages that have been written synchronously to the transaction log.
<b>SyncWriteRlbk</b>	The number of pages that have been written synchronously to the rollback log.
<b>SyncWriteUnkn</b>	The number of pages that have been written synchronously to disk for a reason not covered by other "SyncWrite" counters.
<b>TriggerPages</b>	The number of relocatable heap pages used for triggers.
<b>UnschReq</b>	The number of requests that are currently queued up waiting for an available server thread.
<b>ViewPages</b>	The number of relocatable heap pages used for views.
<b>VoluntaryBlock</b>	The number of server threads that have voluntarily blocked on pending disk IO.

Property	Description
<b>WaitReadCmp</b>	The number of read requests associated with a full comparison (a comparison beyond the hash value in an index) that must be satisfied by a synchronous read operation.
<b>WaitReadOpt</b>	The number of read requests posted by the optimizer that must be satisfied by a synchronous read operation.
<b>WaitReadSys</b>	The number of read requests posted from the system connection that must be satisfied by a synchronous read operation. The system connection is a special connection, used as the context before a connection is made and for operations performed outside of any client connection.
<b>WaitReadTemp</b>	The number of read requests for a temporary table that must be satisfied by a synchronous read operation.
<b>WaitReadUnkn</b>	The number of read requests from other sources that must be satisfied by a synchronous read operation.

## Properties available for each database

The following table lists properties available for each database on the server.

### Examples

❖ **To retrieve the value of a database property:**

- ◆ Use the **db\_property** system function: For example, the following statement returns the page size of the current database:

```
select db_property ( 'PageSize')
```

❖ **To retrieve the values of all database properties:**

- ◆ Use the **sa\_db\_properties** system procedure:

```
call sa_db_properties
```

### Descriptions

Property	Description
<b>Alias</b>	The database name.
<b>CheckpointUrgency</b>	The time that has elapsed since the last checkpoint, as a percentage of the checkpoint time setting of the database.
<b>ConnCount</b>	The number of connections to the database.
<b>File</b>	The file name of the database root file, including path.

<b>Property</b>	<b>Description</b>
<b>FileVersion</b>	The version of the database file.
<b>LogName</b>	The file name of the transaction log, including path.
<b>Name</b>	The database name (identical to alias).
<b>PageSize</b>	The page size of the database, in bytes.
<b>RecoveryUrgency</b>	An estimate of the amount of time required to recover the database.