

Release Bulletin Adaptive Server[®] Enterprise Version 12.5.3a for Windows

Document ID: DC71335-01-1253-02

Last revised: October 3, 2005

Topic	Page
1. Accessing current release bulletin information	3
2. Product summary	3
2.1 Installation kit	4
2.2 Server components	4
2.3 PC-client components	5
2.4 Operating system updates	6
3. Special installation instructions	7
3.1 Post installation instructions	7
3.2 Running dbcc checkcatalog	8
3.3 Installing Adaptive Server version 12.5.3a with previous releases	9
3.4 General configuration steps for Kerberos	9
3.5 Known installation issues	9
4. Special upgrade instructions	10
4.1 Upgrading Job Scheduler	10
4.2 Upgrade support for Adaptive Server version 12.5.3a	11
4.3 Upgrading to ADO.NET 1.1	11
5. Downgrading from 12.5.3a release	11
5.1 Basic downgrade steps	12
5.2 Post-downgrade tasks	12
5.3 Downgrading to Adaptive Server version 12.5.0.x	14

Copyright 1987-2005 by Sybase, Inc. All rights reserved. Sybase, the Sybase logo, ADA Workbench, Adaptable Windowing Environment, Adaptive Component Architecture, Adaptive Server, Adaptive Server Anywhere, Adaptive Server Enterprise, Adaptive Server Enterprise Monitor, Adaptive Server Enterprise Replication, Adaptive Server Everywhere, Adaptive Warehouse, Afaria, Answers Anywhere, Anywhere Studio, Application Manager, AppModeler, APT Workbench, APT-Build, APT-Edit, APT-Execute, APT-Translator, APT-Library, AvantGo Mobile Delivery, AvantGo Mobile Inspection, AvantGo Mobile Marketing Channel, AvantGo Mobile Pharma, AvantGo Mobile Sales, AvantGo Pylon, AvantGo Pylon Application Server, AvantGo Pylon Conduit, AvantGo Pylon PIM Server, AvantGo Pylon Pro, Backup Server, BizTracker, ClearConnect, Client-Library, Client Services, Convoy DM, Copernicus, Data Pipeline, Data Workbench, DataArchitect, Database Analyzer, DataExpress, DataServer, DataWindow, DataWindow .NET, DB-Library, dbQueue, Developers Workbench, DirectConnect, DirectConnect Anywhere, Distribution Director, e-ADK, E-Anywhere, e-Biz Impact, e-Biz Integrator, E-Whatever, EC Gateway, ECMAP, ECRTIP, eFulfillment Accelerator, Embedded SQL, EMS, Enterprise Application Studio, Enterprise Client/Server, Enterprise Connect, Enterprise Data Studio, Enterprise Manager, Enterprise SQL Server Manager, Enterprise Work Architecture, Enterprise Work Designer, Enterprise Work Modeler, eProcurement Accelerator, EWA, Financial Fusion, Financial Fusion Server, Gateway Manager, GlobalFIX, iAnywhere, iAnywhere Solutions, ImpactNow, Industry Warehouse Studio, InfoMaker, Information Anywhere, Information Everywhere, InformationConnect, InternetBuilder, iScript, Jaguar CTS, jConnect for JDBC, M2M Anywhere, Mach Desktop, Mail Anywhere Studio, Mainframe Connect, Maintenance Express, Manage Anywhere Studio, M-Business Channel, M-Business Network, M-Business Server, MDI Access Server, MDI Database Gateway, media.splash, MetaWorks, mFolio, Mirror Activator, MySupport, Net-Gateway, Net-Library, New Era of Networks, ObjectConnect, ObjectCycle, OmniConnect, OmniSQL Access Module, OmniSQL Toolkit, Open Biz, Open Client, Open ClientConnect, Open Client/Server Interfaces, Open Server, Open ServerConnect, Open Solutions, Optima++, PB-Gen, PC APT Execute, PC DB-Net, PC Net Library, PocketBuilder, Pocket PowerBuilder, Power++, power.stop, PowerAMC, PowerBuilder, PowerBuilder Foundation Class Library, PowerDesigner, PowerDimensions, PowerDynamo, PowerScript, PowerSite, PowerSocket, Powersoft, PowerStage, PowerStudio, PowerTips, Powersoft Portfolio, Powersoft Professional, PowerWare Desktop, PowerWare Enterprise, ProcessAnalyst, QAnywhere, Rapport, RemoteWare, RepConnector, Replication Agent, Replication Driver, Replication Server, Replication Server Manager, Replication Toolkit, Report-Execute, Report Workbench, Resource Manager, RFID Anywhere, RW-DisplayLib, RW-Library, S-Designer, SDF, Search Anywhere, Secure SQL Server, Secure SQL Toolkit, Security Guardian, SKLS, smart.partners, smart.parts, smart.script, SOA Anywhere, SQL Advantage, SQL Anywhere Studio, SQL Code Checker, SQL Debug, SQL Edit, SQL Edit/FTP, SQL Everywhere, SQL Modeler, SQL Remote, SQL Server, SQL Server Manager, SQL SMART, SQL Toolkit, SQL Server/CFI, SQL Server/DBM, SQL Server/SNMP SubAgent, SQL Station, SQL STEP, SupportNow, S.W.I.F.T. Message Format Libraries, Sybase Central, Sybase Client/Server Interfaces, Sybase Financial Server, Sybase Gateway, Sybase IQ, Sybase MPP, Sybase SQL Desktop, Sybase SQL Lifecycle, Sybase SQL Workgroup, Sybase User Workbench, SybaseWare, Syber Financial, SyberAssist, SybFlex, SyBooks, System 10, System 11, System XI (Dogo), SystemTools, Tabular Data Stream, TradeForce, Transact-SQL, Translation Toolkit, UltraLite, UltraLite.NET, UNIBOM, Unilib, Uninull, Unisep, Unistring, URK Runtime Kit for UniCode, VisualWriter, VQL, WarehouseArchitect, Warehouse Control Center, Warehouse Studio, Warehouse WORKS, Watcom, Watcom SQL, Watcom SQL Server, Web Deployment Kit, Web.PB, Web.SQL, WebSights, WebViewer, WorkGroup SQL Server, XA-Library, XA-Server, XcelleNet, and XP Server are trademarks of Sybase, Inc. 06/05

Topic	Page
5.4 Additional steps required if you used 12.5.3a features before downgrading	15
6. Known problems	19
6.1 Highlighted encrypted columns issues	20
6.2 Starting replication agent with the recovery option	21
6.3 SSL and Replication Agent	21
6.4 Connecting to Adaptive Server with the default character set	22
6.5 Blank spaces in an ldap server entry	22
6.6 alter table and triggers with if update() clauses	22
6.7 Truncation error using select with power() function and numeric argument	23
6.8 Some messages returned incorrectly on UTF-8 servers	23
6.9 Cross platform bcp of parsed XML images	23
6.10 Using DSEDIT to edit sql.ini files	24
6.11 Starting the Sybase License Manager Utility	24
6.12 Starting Sybase Central	24
6.13 Memory allocation failure	24
6.14 Migration of stored procedures	25
6.15 cis connect timeout and enable SNMP are not implemented	25
6.16 XP Server returns incorrect results	25
6.17 Network license daemon support	26
6.18 Configuring connectivity for clients accessing wide data	26
6.19 Upgrading check constraints	26
6.20 Specifying default directory services	27
6.21 Handling multibyte character sets during migration	27
6.22 Dynamic configuration in Adaptive Server and Performance Monitor	27
7. Product compatibilities	27
7.1 Known compatibility issues	29
8. Documentation updates and clarifications	29
8.1 Web Services User's Guide	30
8.2 Updates for Using Sybase Failover in a High Availability System	31
8.3 Utility Guide	32
8.4 Transact-SQL User's Guide	38
8.5 Reference Manual: Building Blocks	38
8.6 Reference Manual: Commands	39

Topic	Page
8.7 Reference Manual: Procedures	42
8.8 System Administration Guide	57
8.9 Performance and Tuning Guide	58
8.10 XA Interface Integration Guide for CICS, Encina, and TUXEDO	59
9. Technical support	59
10. Other sources of information	59
10.1 Sybase certifications on the Web	60
10.2 Sybase EBFs and software maintenance	60

1. Accessing current release bulletin information

A more recent version of this release bulletin may be available on the Web. To check for critical product or document information added after the release of the product CD, use the Sybase Technical Library Product Manuals Web site.

❖ Accessing release bulletins at the Technical Library Product Manuals Web site

- 1 Go to Product Manuals at <http://www.sybase.com/support/manuals/>.
- 2 Follow the links to the appropriate Sybase product.
- 3 Select the Release Bulletins link.
- 4 Select the Sybase product version from the Release Bulletins list.
- 5 From the list of individual documents, select the link to the release bulletin for your platform. You can either download the PDF version or browse the document online.

2. Product summary

Enclosed is Sybase[®] Adaptive Server[®] Enterprise version 12.5.3a. Server and client components are distributed on separate CDs.

For details on system requirements, including disk space and RAM, see the installation guide for your platform.

For more information specific to jConnect[™] for JDBC[™], jsql, and Ribo, see the jConnect documentation.

2.1 Installation kit

Note The installation guide for Adaptive Server version 12.5.3a is the same installation guide as for Adaptive Server version 12.5.1.

The installation kit includes:

- The server CD – for contents list, see “Server components” on page 4.
- The PC-client CD – contains software client components for installation on Windows 2000, Windows XP Pro, and Windows 2003 computers.
- *Release Bulletin for Adaptive Server Enterprise Version 12.5.3a for Windows* (this document).
- *Installation Guide for Adaptive Server Enterprise 12.5.1 on Windows*.

2.2 Server components

The Server CD includes the following components:

- Sybase Servers:
 - Adaptive Server 12.5.3a
 - Adaptive Server Enterprise Monitor™ Server 12.5.3a
 - Historical Server 12.5.3a
 - SQL Remote™ 7.0.4
 - Adaptive Server Enterprise Diagnostic Server 12.5.2
- Language Modules 12.5.3a
- Adaptive Server Administration Tools
 - Sybase Central™ 4.1
 - Adaptive Server plug-in 12.5.3a
 - Job Scheduler Templates and Utilities 2.2
 - SQL Advantage
 - Microsoft Cluster Server Resource Type
- Software Developer’s Kit (SDK) 12.5.1, which includes the following:
 - Connectivity:

- Open Client™ 12.5.1
- Embedded SQL™/C 12.5.1
- Embedded SQL/Cobol 12.5.1
- Monitor Client Library 12.5.1
- XA Interface for Adaptive Server Distributed Transaction Manager 12.5.1
- Adaptive Server Data Providers:
 - ADO.NET1.1
- jConnect for JDBC 5.5 and related utilities and documentation
- jConnect for JDBC 6.0 and related utilities and documentation

2.3 PC-client components

The PC-client CD includes the following components, which can be installed on Windows:

- Software Development Kit (SDK) 12.5.1, which includes the following:
 - Open Client 12.5.1
 - Embedded SQL/C 12.5.1
 - Embedded SQL/Cobol 12.5.1
 - Monitor Client Library 12.5.1
 - XA Interface for Adaptive Server Distributed Transaction Manager 12.5.1
- Adaptive Server Data Providers:
 - ADO.NET 1.1
 - OLE DB Provider by Sybase 12.5.1
 - ODBC Driver by Sybase 12.5.1
- Language Modules 12.5.3
- jConnect 5.5 for JDBC and related utilities and documentation
- jConnect 6.0 for JDBC and related utilities and documentation
- Adaptive Server Administration Tools:
 - Sybase Central 4.1

- Adaptive Server Plug-in
- Job Scheduler Templates and Utilities 2.2
- SQL Advantage
- Microsoft Cluster Server Administration Extensions
- InfoMaker 9.0
- PowerDesigner 10.0
- PowerTransfer 10.0

2.4 Operating system updates

Windows requires the following operating system service pack to run Adaptive Server version 12.5 and later components:

- Windows 2000 – Service Pack 3 or later.
- Windows XP Professional – There is no service pack required.
- Windows 2003 – There is no service pack required.

Note When you are running IPv6 on Windows platforms, Adaptive Server must be running on Windows 2003.

If your operating system requires this service pack, install it before installing Adaptive Server.

To determine which service packs have been installed on your system, see “Viewing installed patches,” next.

Contact your operating system representative for a complete list of service packs. Do not use a service pack that is earlier than the version suggested for your operating system. Use the service pack recommended by the operating system vendor, even if it supersedes the service pack listed here.

Viewing installed patches

To determine which service pack operating system level and build number are installed on your system, enter:

- 1 Start Windows Explorer.
- 2 Select Help | About Windows.
- 3 Find the Service Pack number on the Version line.

In the following example, *xxxx* is the build number and *n* is the service pack number.

```
Version 4.0 (Build xxxx: Service Pack n)
```

3. Special installation instructions

This section provides information for installing Adaptive Server version 12.5.3a and later that is not included or should be corrected in the installation guide.

3.1 Post installation instructions

Sybase recommends that you perform all of the steps below after installing Adaptive Server on Windows platforms. These steps should be performed as the “sybase” user that installed the Adaptive Server product. Verify that the “sybase” user is a member of the Windows system-defined group “Administrators” so the user has the access privileges that allow them to complete the steps described below.

- 1 Stop all Adaptive Server services.

Using the Administrative Tools, open Services, select each of the Sybase Adaptive Server entries and stop them. There are separate entries in Services for Sybase Adaptive Server SQLServer, XPServer, BCKServer, MONServer, and SYSAM.

- 2 Identify the drives and folders where the Adaptive Server software is installed, and identify Adaptive Server’s disk devices. Use the Windows command line tool `cacls` or Windows Explorer to set access permissions on the identified drives and folders. For detailed instructions on the use of `cacls` or Windows Explorer, see your Windows documentation or your Windows Operating System administrator.

Sybase recommends granting full control to the user “sybase” and denying access to others on all identified drives, files, and folder hierarchies.

If users other than “sybase” need access to files (configuration files, programs, libraries, and so on) in the folder hierarchy, limit their access to read and execute. Granting this access and any further access permissions should be carefully reviewed by the administrator to ensure they are required.

Note `cacls` is a command line utility available on Windows 2000 and later versions. On earlier Windows versions, this utility is available from the NT Resource Kit.

- 3 Adaptive Server services should start as the user “sybase”:
 - Using the Administrative Tools, open Services and select each of the Sybase Adaptive Server entries installed.
 - Set the “Log On” properties for each of the entries to use “sybase” and the password for this account. This action results in the user “sybase” being granted ‘log on as a service’ rights by Windows.
- 4 Start Adaptive Server services. Using the Administrative Tools, open Services and select each of the Sybase Adaptive Server entries that were stopped in Step 1 and start them.

3.2 Running dbcc checkcatalog

When any column value is accessed in a database table, Adaptive Server has been enhanced to validate the length of the row and the length of the column value within the row, using information stored within the system catalogs in that database. Any discrepancy is reported in the error log with the message:

```
Invalid column length: <length>. Value must be between  
0 and <max col length> for '<row type>' row with minimum  
row length of <min row length>.
```

This check requires accurate information within the system catalogs. It is important to verify the accuracy of this information before any database activity occurs after the upgrade to Adaptive Server to prevent any false alarms.

To verify the information, run `dbcc checkcatalog(<database name>)` for all databases. If any errors are reported, fix them by running `dbcc checkcatalog(<database name>, "fixall")`.

Dump the database after the fixes have been made. If an older dump is loaded into Adaptive Server, then any fixes that dbcc checkcatalog has applied to system catalogs is lost by the load. Rerun dbcc checkcatalog after the load has completed.

3.3 Installing Adaptive Server version 12.5.3a with previous releases

You can install this release of Adaptive Server on top of an existing 12.5 or later release without having to do an upgrade. Just shutdown your servers, install the release on top of the existing release area and do not configure any servers, and then start your servers back up. There is no upgrade.

For more information about post installation tasks, see the Technote at the Sybase Web site at <http://my.sybase.com/detail?id=1013610>.

3.4 General configuration steps for Kerberos

For information about installing Kerberos security software, see the white paper on the Sybase Web site at <http://www.sybase.com/detail?id=1029260>. This helps you get started with using Kerberos with Adaptive Server, OpenClient, OpenServer and jConnect applications.

3.5 Known installation issues

This section describes known problems or issues that you may encounter during the installation process.

3.5.1 Adding a local server name

[CR #348555] As part of the installation, you should have a local server name for the server. If you do not, sp_sysmon "cachewizard" does not work.

Workaround: To add a local server, enter:

```
sp_addserver <local servername>, local
```

Restart the server for this change to take effect.

3.5.2 Using Adaptive Server with MTS and DTC

If you have difficulty using Adaptive Server with transactionally enabled Microsoft MTS components and Microsoft DTC, make sure you:

- 1 Configure Adaptive Server for DTM.
- 2 Give the user `dtm_tm_role`.
- 3 Confirm that `%SYBASE%\%SYBASE%_OCS\ddl` is in the system path and contains `libxadtm.dll` so it is picked up by DTC.

If you change the system path, restart Windows. In most situations, the installation sets the system path correctly.
- 4 Set the ODBC DSN Distributed Transaction Support value on the Advanced tab to Native OLE.

If Adaptive Server is running on a platform other than Windows, or if you want to use the XA interface, make sure the ODBC DSN Distributed Transaction Support value under the Advanced tab is set to XA Protocol, which is the default value.
- 5 Confirm that the DTC service is running. Click the Services icon in the Control Panel, and look for “MSDTC.”

For more information, see *Using Adaptive Server Distributed Transaction Management Features*.

4. Special upgrade instructions

This section provides information for upgrading to Adaptive Server version 12.5.3a that is not included or should be corrected in the installation guide.

4.1 Upgrading Job Scheduler

[CR #371574] If you are upgrading from Adaptive Server versions earlier than 12.5.2, you must run the `installjsdb` script anytime you upgrade Job Scheduler.

❖ Upgrading Job Scheduler

- 1 Disable Job Scheduler before you upgrade Adaptive Server:

```
sp_configure "enable job scheduler", 0
```
- 2 Upgrade Adaptive Server. See the installation guide for details on upgrading.
- 3 Run the `installjsdb` script:

```
isql -Usa -Psa_password -Sservername -i
```

```
$SYBASE_ASE/scripts/installjsdb
```

Note You must have the directory with the location of the `isql` executable (`$SYBASE_OCS/bin`) in your path.

The `installjsdb` script looks for the `sybmgmtdb` database. The script updates Job Scheduler tables and stored procedures.

- 4 Enable Job Scheduler:

```
sp_configure "enable job scheduler", 1
```

- 5 To start Job Scheduler, you can either restart the server or enter the following:

```
use sybmgmtdb
go
sp_js_wakeup "start_js",1
go
```

4.2 Upgrade support for Adaptive Server version 12.5.3a

Sybase supports upgrades from Adaptive Server versions 11.9.x, 12.0.x, 12.5, 12.5.1, 12.5.2, and 12.5.3 to Adaptive Server version 12.5.3a.

4.3 Upgrading to ADO.NET 1.1

Adaptive Server version 12.5.2 and later includes ASE ADO.NET Data Provider 1.1. The Common Language Runtime (CLR) bytecode for applications built against previous versions of the Data Provider is not compatible with this version of the data provider. To use the updated version of the Data Provider, your applications should be rebuilt against the new Data Provider. Multiple versions of the Data Provider can coexist on the client machine and therefore your old application can continue to use the previous version of the Data Provider. See the *Adaptive Server ADO.NET Data Provider User's Guide* for more information.

5. Downgrading from 12.5.3a release

This section describes instructions to downgrade from Adaptive Server 12.5.3a to Adaptive Server 12.5.3 or an earlier 12.5.x release. These include:

- Basic downgrade steps

- Post-downgrade tasks, whether or not 12.5.3a features are enabled
 - Downgrading to Adaptive Server version 12.5.0.x
 - Information regarding downgrading individual features
 - Returning to Adaptive Server version 12.5.3a
-

Note Adaptive Server 12.5.3a is not an upgrade release, and you may return to previous versions after using the new features in 12.5.3a. Sybase recommends that you do not enable the new features offered in Adaptive Server version 12.5.3a until you are committed to staying with 12.5.3a.

5.1 Basic downgrade steps

Use these basic steps to return to Adaptive Server version 12.5.3 or earlier. Further steps are necessary if you have enabled 12.5.3a features.

- 1 Overlay the 12.5.3 or earlier binary as you did when installing 12.5.3a.
- 2 Restart the server with the old 12.5.x binary.
- 3 Run the 12.5.x version of *installmaster* to return stored procedures to their original version.

These are the minimum tasks needed for rollback. Additional steps are described in the rest of the section.

Note Returning to 12.5.3 ESD #1 is preferable, because 12.5.3 ESD #1 contains fixes for certain security vulnerabilities. Should you return to a pre-12.5.3 ESD #1 version, your site may be at a greater risk of attack.

5.2 Post-downgrade tasks

After you perform the basic downgrade steps, changes from the 12.5.3a upgrade still persist.

5.2.1 System stored procedures

After you run *installmaster* on your downgraded server, any changed system stored procedures are returned to their 12.5.x status. Any new stored procedures introduced to Adaptive Server in releases subsequent to 12.5.x are not removed. Attempts to execute such stored procedures against an older binary will have unpredictable results.

5.2.2 Unrecognized tokens or objects

The on-disk structures of stored procedures, triggers, and views may contain statement identity tokens, datatypes and object references that are not understood by the earlier Adaptive Server versions. You must drop all compiled objects that use features introduced into Adaptive Server subsequent to the release to which you are downgrading.

If you are returning to a pre-12.5.1 version of Adaptive Server, Sybase recommends that you return to 12.5.0.3. This is because 12.5.0.3 includes error handling code for unrecognized tokens or objects pertaining to features in 12.5.1. If the 12.5.0.3 Adaptive Server encounters these unknown objects, it handles them with a meaningful error message.

Should you return to a pre-12.5.0.3 version of Adaptive Server, no such soft landing is provided. Using the new features can result in inexplicable behavior, misleading error messages and stack traces.

5.2.3 System tables

After returning to a pre-12.5.3a binary:

- The new columns added to *syscolumns* remain but are not used.
- When returning to 12.5.0.x or earlier, the new datatypes *date*, *date null*, *time*, and *time null* remain in *systypes* but you cannot use them in 12.5.0.3 and attempts to use them in 12.5.0.2 or earlier will cause a stack trace. See the special procedure for downgrading to Adaptive Server version 12.5.0.x to avoid problems with new datatypes.
- *sp_values* is dropped and recreated when you run the 12.5.x *installmaster*, so any new types are eliminated from this table.
- *sysmessages* still contains all the new error messages, but that is not a problem because they are not used.

5.2.4 Configuration parameters

New configuration parameters have been added in 12.5.3a. When starting the previous version of the Adaptive Server, use the old *.cfg* configuration file.

If you forget to run the 12.5.x *installmaster* before starting Adaptive Server, an error message displays at startup.

Running the 12.5.x version of *installmaster* removes configuration parameters that belong to 12.5.3a, by deleting *sysconfigures* rows for configuration parameters that do not exist in *syscurconfigs*. After running *installmaster* the error messages will no longer appear when you start the server.

5.2.5 Keywords

Two keywords, *func* and *function*, were removed in Adaptive Server 12.5.1, making it possible to create identifiers using these names. If you did not use these as identifiers under 12.5.3a, no action is needed. If you did, the parser generates a syntax error under 12.5.0.x since it interprets these words as keywords.

See *Additional steps required if you used 12.5.3a features before downgrading* for more information.

5.3 Downgrading to Adaptive Server version 12.5.0.x

Perform these steps only when you are downgrading to 12.5.0.x. Returning to 12.5.0.3 is preferable, because these versions can handle a number of error conditions that may arise as a result of downgrading. Older version of Adaptive Server cannot handle these situations and results are unpredictable.

Although it is not necessary to restore *systypes* and *syscharsets* to their previous state, Sybase provides the store procedure *sp_twelve_five_one* to do so.

After returning to Adaptive Server version 12.5.0.x, remove the new datatypes from *systypes* and *syscharsets* and return these tables to their 12.5.0.x state by performing the following steps:

- 1 Enable updates to system tables:

```
sp_configure "allow updates", 1
```

- 2 Remove the 12.5.1 datatypes:

```
sp_twelve_five_one "remove"
```

- 3 Disable updates to system tables:

```
sp_configure "allow updates", 0
```

This procedure also marks all procedural objects for recompiling, to insure that the pre-12.5.1 Adaptive Server does not encounter 12.5.1 objects or datatypes.

5.4 Additional steps required if you used 12.5.3a features before downgrading

If you are rolling back after having used any of the 12.5.3a features, additional steps can be necessary. You should take steps before you downgrade to pre-12.5.3a, and others should be performed immediately after downgrading. Areas of concern are listed for each new feature.

In general, no additional steps are required when you are returning to an Adaptive Server version in which the feature was already available. When returning to a server version earlier than the version in which the feature was introduced, you must be aware of the areas of concern listed below.

See the encrypted columns section in the *New Feature Bulletin for Adaptive Server Enterprise version 12.5.3a*.

5.4.1 Encrypted Columns

If you have used the encrypted columns feature of 12.5.3a, you must take steps in 12.5.3a before you downgrade. Perform the following:

- 1 Drop all stored procedures, views, and triggers that use the SQL extensions for encrypted columns.
- 2 Remove all encrypted data by dropping tables with encrypted columns or using `alter table` to decrypt the data or drop the encrypted columns.
- 3 Run `sp_encryption remove_catalog` to remove the `sysencryptkeys` system catalog in each database.

5.4.2 Real Time Messaging

Drop all stored procedures, views, and triggers that use the messaging built-ins for the Real Time Messaging feature.

For more information about Real Time Messaging, see the *Messaging Services User's Guide*.

5.4.3 New sort orders

Upgrading from 12.5.0.3 to 12.5.1 or later modifies `syscharsets`. All of the sort orders that were (formerly) destined for use with `unichar` was rewritten in terms of ID and CSID, although the names remain the same. Other sort orders (that is, any that you loaded with the `charset` utility or the `sqlloc` GUI front end) remain unmodified.

While running under 12.5.3a, you may configure UTF-8 with a non-binary sort order. If you return to 12.5.0.3 that server fails to start, with a message saying that the character set/sort order combination is not supported. Versions earlier than 12.5.0.3, however, do not have the error handling code; you do not get the message but instead see inexplicable behavior. The most likely symptom is that the server cannot find anything using indexes on the system tables.

To get around this:

- Restart the 12.5.3a server.
- Configure UTF-8 with binary sort order `bin_utf8` (ID 50). Do not use any other binary sort order (there are three to choose from); `bin_utf8` is the only one known to 12.5.0.3 and earlier servers.

Now when you start 12.5.0.3, `syscharsets` is still in the 12.5.3a state. If you run `sp_helpsort`, rewrite `syscharsets` to a close approximation of what it was before the upgrade.

Note If you execute `sp_helpsort` now, and then decide to go back to 12.5.3a, be aware that although `syscharsets` is in the 12.5.0.3 state, the upgrade does not run again since it has already run once. Execute `sp_helpsort` again on return to 12.5.3a to update the `syscharsets` table.

5.4.4 Unicode parser

If you have taken advantage of unicode parser feature through explicit or implicit use of the `U&'syntax'` (in a stored procedure, for example), the text of the stored procedure is saved in `syscomments`. If you roll back to 12.5.0.3 and the server needs to recompile from query text, you get syntax errors.

On another note, if you take advantage of the fact that in 12.5.3a `unichar` no longer requires UTF-8, you may have stored procedure plans that contain `unichar`-to-`char` conversions. In 12.5.1 this is handled correctly on a non-UTF-8 server. In 12.5.0.3 and earlier, however, the assumption was UTF-8; anything else causes the conversion to throw an error and abort the transaction.

5.4.5 date and time datatypes

When you start the 12.5.3a server, the new system types are installed in `systypes`. If you returned to 12.5.0.3:

- The new types remain in `systypes`, but server operation is not affected provided you did not use the new types on the 12.5.3a server
- You cannot create using date or time because error checks were added
- If the server encounters date/time tokens, it raises error messages
- If the server encounters a 12.5.3a procedure that accesses a table with date/time types, a stack trace results.

If you returned to a pre-12.5.0.3 server, nothing prevents you from creating objects using date and time types (since no error handling code exists for this purpose), but this is not advisable because any operations on those objects such as selects or inserts result in stack traces.

If you have created 12.5.3a tables with date or time columns, or procedures with date or time parameters, you need to remove references to these types before rolling back to a pre-12.5.1 server. You can run the following, which identifies the offending tables and procedures, advising you to remove references to them or use alter table to modify the types:

```
sp_date_time_type
```

If `sp_date_time_types` does not find any date/time objects, it marks procedural objects for recompile, removing all internal references to these new types

5.4.6 SQL derived tables

If you used SQL derived tables before returning to a pre-12.5.1 version server, and you:

- Create views with SQL derived tables using version 12.5.3a roll back to ASE 12.5.0.3, then try to select from the view, you get an error that derived tables are not supported in that version.
- Create stored procedures with SQL derived tables in 12.5.3a roll back to ASE 12.5.0.3 or earlier versions and run the stored procedure, you will see an error that the object (SQL derived table) does not exist.
- Create views with SQL derived tables in 12.5.3a return to a pre-12.5.0.3 version, you get stack trace errors.

5.4.7 Native XML

The XML Services feature of Adaptive Server includes the following new built-in functions and clauses:

- `xmlextract` built-in function
- `xmlparse` built-in function
- `xmltest` predicate
- `for xml` clause of `select` commands

If you create views or stored procedures using one of these features in 12.5.3a, then return to 12.5.0.3, you get an error that the feature is not supported in that version. If you store XML documents in:

- Their original text form, they can be processed on both Adaptive Server 12.5.3a and ASE 12.5.0.3.
- The parsed form generated by the `xmlparse` built-in function, then return to 12.5.0.3, the parsed form is not accessible. You must first regenerate the text form of the XML documents in 12.5.3a.

For example, if a column named `xmlindexed` contains data generated in Adaptive Server 12.5.3a by the `xmlparse` built-in function, you can regenerate the text form of the document in the `xmlsource` column with the following SQL statement that uses the Adaptive Server 12.5.3a `xmlextract` built-in function:

```
update xmltab
set xmlsource = xmlextract("/", xmlindexed)
```

You may then do one of the following:

- Process the `xmlsource` column directly with the Java-based XQL processor, using the `com.sybase.xml.xql.Xql.query` Java-based method, or
- Update the `xmlindexed` column with the parsed form suitable for processing with the Java-based XQL processor, using the following statement:

```
update xmltab
set xmlindexed = com.sybase.xml.xql.Xql.parse(xmlsource)
```

If you do not want to add the `xmlsource` column, you can combine these steps with the following SQL statement in ASE 12.5.3a:

```
update xmltab
set xmlindexed
= com.sybase.xml.xql.Xql.parse
(xmlextract("/", xmlindexed))
```

Prior to execution of this update statement, the `xmlindexed` column contains the parsed form of the documents generated by the `xmlparse 12.5.3a` built-in function. After the update statement, that column contains the parsed form of the documents, generated by the `com.sybase.xml.xql.Xql.parse` Java-based method, suitable for processing in Adaptive Server 12.5.0.3 with the `com.sybase.xml.xql.Xql.query` Java-based method.

A configuration parameter was added to enable XML in 12.5.1. When you first restart with the prior version, you may see error message 5859 advising you that the parameter is unknown. After running *installmaster* from the pre-12.5.1 version this warning should no longer appear.

5.4.8 Automatic database expansion

If you installed the 12.5.1 automatic database expansion procedures using `installdbextend`, and then applied the threshold procedure to one or more database segments, the thresholds might not work properly when applied to the log segment after a downgrade to 12.5.0.3.

To clear all auto-expansion thresholds that might exist on one or more segments before downgrading, use the command:

```
sp_dbextend 'clear', 'threshold'
```

Alternatively, you can disable the entire automatic expansion feature server-wide without changing any existing rules or clearing any thresholds. Execute the following commands using `sa_role`:

```
use master
go
sp_dbextend 'disable', 'database', 'server-wide'
go
```

This prevents threshold procedures from doing any work even if they were fired at runtime.

Sybase recommends that you leave all the policies and thresholds in place, and simply disable the entire feature server-wide before the downgrade. This simplifies re-enabling automatic expansion if you return to 12.5.1 later.

6. Known problems

This section describes known problems and workarounds for Adaptive Server.

6.1 Highlighted encrypted columns issues

This section covers issues on encrypted columns.

6.1.1 Residual *cleartext* data on disk

[CR #392841] Encryption of data through `alter table` copies the table to new data pages. The old (cleartext) values remain in the unused pages until those pages are re-used by the server. A similar situation exists with the transaction log. Log records of updates and deletes on tables prior to encrypting the data (through `alter table`) will also persist until those log pages are re-used.

Workaround: To destroy residual data pages, build a brand new device. Use `bcp` to copy all the data out from the old device and back in to the new device. The `-C` option on `bcp` will leave the data in ciphertext format during the transfer. Destroy the old device. For the case of plaintext values that may remain in the transaction log, normal database activity replaces these pages.

6.1.2 Using referential integrity with encrypted columns

[CR #369088] Referential integrity and encryption may not apply to the same column in Adaptive Server version 12.5.3a. For example, if you have a primary key with a foreign key dependency, you may not encrypt the primary key. Similarly, you may not encrypt a foreign key.

Workaround: There is no workaround for this issue.

6.1.3 Executing `sp_encryption 'remove catalogs'` fails to give error message

[CR #395745] When executing `sp_encryption 'remove catalogs'` to prepare for downgrade, you should get an error when a given key has been used to encrypt a column. No error is issued if the column and the key are in the same database, and `sp_encryption` removes `sysencryptkeys` even where there are encrypted columns in the database.

Workaround: With `sso_role` permissions, run:

```
sp_encryption help
```

This command prints a list of keys in the current database. For each key listed, the System Security Officer runs:

```
sp_encryption help, <keyname>, display_cols
```

The command lists the names of any columns encrypted by the key. Then use `alter table` to decrypt these columns before executing `sp_encryption 'remove_catalogs'`.

6.1.4 Views and encrypted columns used with bcp

[CR #395749] If a view uses an encrypted column in the `where` clause, such as `bcp out` using `-C` fails:

```
create view v as select custid from customer where
ssn = '123 44 5678'
```

Workaround: Put the results of the view into an intermediate table using `select into` or `insert-select`. You may use `bcp out` with `-C` on the intermediate table.

6.1.5 Encrypted column license display

[CR #395247] 12.5.3a Adaptive Server Plug-in does not correctly display encrypted column license information. Even if encrypted columns are licensed and enabled, Adaptive Server Plug-in does not display them as part of the license information.

Workaround: There is no workaround for this issue. This does not affect Adaptive Server functionality.

6.2 Starting replication agent with the recovery option

[CR #387036] Starting a replication agent with the `recovery` option to apply a procedure to recover a primary database from old logs fails.

Workaround: Use alternatives to synchronize primary and replicated databases, depending on the environment. You can use:

- `rs_subcmp` utility
- `dump database` and `load database`
- set autocorrection in replication server

For more information, see the Replication Server manuals.

6.3 SSL and Replication Agent

[CR #391581] An error occurs when you attempt to establish an SSL connection between Replication Agent Thread version 12.5.3.a and Replication Server version 12.6 ESD #5.

Workaround: When you configure Adaptive Server for SSL, Replication Server creates a primary connection, but the Replication Agent Thread fails on the initial connection attempt. However, when you restart the Replication Agent Thread, it establishes a successful SSL connection.

6.4 Connecting to Adaptive Server with the default character set

[CR #311604] ASE plug-in is sometimes unable to connect to Adaptive Server using the default character set.

Workaround: If you cannot connect to Adaptive Server using the default character set, try to connect using `iso_1`. If you are unable to connect using `iso_1`, change the unicode setting to 1 using the following command:

```
sp_configure 'enable unicode conversions', 1
```

6.5 Blank spaces in an *ldap* server entry

[CR #333307] If you have a blank space after your `ldap` server entry, `dscp` defaults to using the interfaces driver and does not connect to an `ldap` server.

Workaround: When you want to use `dscp` to make an entry into an `ldap` server, enable `ldap` by editing the `$SYBASE/OCS-12_5/config/libtcl.cfg` file to add the `ldap` server you plan to use.

6.6 *alter table* and triggers with *if update()* clauses

[CR #199655, 343165] If a trigger contains an `if update()` clause, data modifications that should fire the trigger that are performed after executing `alter table add`, `alter table drop`, `alter table lock`, or `alter table modify` may cause errors in column references. Triggers on the altered table that use an `if update()` clause in the body of the trigger to reference a column may not fire, or may fire incorrectly.

Workaround: After the `alter table` operation has completed, drop and re-create all triggers on the altered table. Doing so causes the `if update()` clause in the triggers to correctly reference the new columns by their new column offsets, so the trigger code executes correctly.

6.7 Truncation error using *select* with *power()* function and numeric argument

[CR #236723] If you use the *select* command with the function *power()*, and use a numeric argument, the result is a truncation error.

Workaround: Convert the numeric argument to a float.

6.8 Some messages returned incorrectly on UTF-8 servers

[CR #271646] When the Adaptive Server default character set is UTF-8, the ASE Replicator procedures *sp_addreplicatart* and *sp_addreplicateconn* return some error messages incorrectly.

This problem does not affect the function of these procedures.

Note Even though the error messages are not returned correctly, the errors are logged correctly in the ASE Replicator log files.

Workaround: There is no workaround for this issue.

6.9 Cross platform *bcp* of parsed XML images

[CR #332012] You cannot use *bcp* or replicate parsed XML image data between a big-endian platform and a little-endian platform, for example between a Solaris platform (big-endian) and a Linux or Windows platform (little-endian).

The term “parsed XML image data” refers to data of datatype *image* that is generated by the *xmlparse* built-in function.

Workaround: When you transfer XML data between platforms, transfer the character form of the XML documents rather than the parsed XML form. If you have not stored the character form of the XML documents, you can regenerate it from the parsed form. For example, if column *xmlindexed* of the *xmldocuments* table is an *image* column containing parsed XML image data, the following two commands append a new *text* column to the table and populate it with a character form of the documents contained in the *xmlindexed* column:

```
alter table xmldocuments add xmltext text
update xmldocuments set xmltext = xmlextract('/', xmlindexed)
```

Then you can use *bcp* or replicate the *xmltext* column from one platform to the other.

6.10 Using *DSEEDIT* to edit *sql.ini* files

[CR #350610] Sybase often recommends that you use different installation directories on the same Windows machine when installing multiple Sybase products. However, *dsedit* used to edit the *sql.ini* file only allows you to work with the *sql.ini* file in the system's *%SYBASE%* location.

dsedit cannot be used to work with different *sql.ini* files at two different locations.

Workaround: Either manually edit the non-*%SYBASE%* *sql.ini* file using a plain text editor, or consolidate all *sql.ini* entries into the same *sql.ini* file.

6.11 Starting the Sybase License Manager Utility

[CR #351387] If the *DISPLAY* environment variable is not set, executing *\$\$SYBASE/SYSAM-1_0/bin/lmgr* to start the Sybase License Manager Utility causes errors.

Workaround: You must set the *DISPLAY* environment variable by executing:

In Bourne shell where *host name* is the name of the system where you want the installer screens to display:

```
set DISPLAY=<host name>:0.0
export DISPLAY
```

In C shell:

```
setenv DISPLAY <host name>:0.0
```

6.12 Starting Sybase Central

[CR #351275] In some cases, when using Sybase Central, the help menu does not launch the help window.

Workaround: If you have installed the ASE Plug-in version 12.5.2, and cannot launch Sybase Central help, download and run "KeyHH" from the Keyworks Software Web site at <http://www.keyworks.net/keyhh.htm>.

Select the download link, and run the downloaded program. This program fixes Windows HTML Help system and allows Sybase Central to run properly.

6.13 Memory allocation failure

[CR #350856] If there is insufficient memory in the heap class Adaptive Server hangs.

Workaround: Double the heap class size.

6.14 Migration of stored procedures

[CR #328131] `sybmigrate` does not migrate stored procedures that have a space character in their name, or belong to a procedure group.

Workaround: There is no workaround for this issue.

6.15 `cis connect timeout` and `enable SNMP` are not implemented

[CR #323177] Configuration parameters `cis connect timeout` and `enable SNMP` are not currently implemented.

Workaround: Do not use these configuration parameters.

6.16 XP Server returns incorrect results

[CR #280073] On all systems, using an extended stored procedure (ESP) with parameters can result in wrong data being sent to XP Server. Because of this, XP Server could return incorrect results.

This happens when there is no parameter declaration list in the ESP, but one or more parameters are supplied at execution time. Before CR #280073, all parameters supplied at execution time were sent to the XP Server, causing incorrect results. CR #280073 fixes this problem by silently ignoring all parameters supplied at execution time when the parameter declaration list is empty.

This fix does not cause any behavior change when the parameter declaration list in the ESP matches the parameters supplied at execution time.

If you are using ESPs without a parameter declaration list, and you supply parameters at execution time, drop and re-create the ESP in Adaptive Server. Make sure the parameter declaration list in the ESP matches the parameter list supplied at execution time

You can use trace flag 523 to revert to pre-CR280073 behavior. This is a server-wide trace flag you set when you restart Adaptive Server.

6.17 Network license daemon support

Adaptive Server version 12.5.2 and higher has upgraded the SySAM daemons and libraries (FlexLM) to version 8. The SySAM daemons (Imgrd and SYBASE) are backward compatible; the libraries are not. As a result, Adaptive Server 12.5.2 cannot check out licenses from earlier versions of the SySAM license daemons.

Workaround: Sites using the Network License Model should make sure that the central SySAM license daemons are upgraded first.

6.18 Configuring connectivity for clients accessing wide data

[CR #271822] If you are using wide data (that is, data other than text and image that is greater than 255 bytes), and you are using the bulk library APIs, make sure you configure the connectivity capability for wide data. Failure to do so may result in memory corruption.

6.19 Upgrading check constraints

[CR #272941] When you upgrade to Adaptive Server 12.5.0.1 or later, check constraints created with the following criteria as they may not have been upgraded correctly:

- The check constraint was created as a named check constraint
- Quoted identifiers were set to on while the constraint was created
- Quote identifiers were used around the entire constraint name
- The constraint name was also a key word, or not a legal identifier

You see parser syntax errors after you have upgraded to Adaptive Server 12.5.0.1 and later, and when you first add data to the constrained column, or when you run `dbcc upgrade_object`.

To determine whether this problem occurs on your upgraded databases, run this command to check for syntax errors on your constraints and rules:

```
dbcc upgrade_object (dbname, rule)
```

Workaround: There are two workarounds to this problem:

- Issue `alter table drop constraint_name` to drop a constraint that was not upgraded successfully. Then issue `alter table add constraint_name`, and re-create the constraint.

- As System Administrator, manually update *syscomments.text* to add quoted identifiers to the constraints that were not successfully upgraded.

6.20 Specifying default directory services

[CR #256822] When you specify the directory services using *-i*, all Adaptive Servers that are started are affected rather than the specified Adaptive Server. *-i* overrides *libtcl.cfg*.

Workaround: Do not use *-i*; only allow the *libtcl.cfg* file in the default location to specify all directory services.

6.21 Handling multibyte character sets during migration

[CR #353079] If you configure multibyte character sets after migrating data from system catalogs *sysattributes* and *sysxtypes*, the text columns in these catalogs are inconsistent with the multibyte character sets.

Workaround: Manually run *dbcc fix_text* on *sysattributes* and *sysxtypes*, to make the text columns consistent with the multibyte character sets.

6.22 Dynamic configuration in Adaptive Server and Performance Monitor

[CR # 234103] When you reconfigure Adaptive Server to increase the number of user connections, number of open databases, or number of engines at startup, these new instances are not visible in the PerfMon instance window.

Workaround: To monitor Adaptive Server using PerfMon, restart Adaptive Server after reconfiguring number of user connections, number of open databases, or number of engines at startup.

7. Product compatibilities

This section lists the Sybase components that are compatible with Adaptive Server Enterprise version 12.5 and later. For information about operating system requirements, see individual component documentation.

Note DirectConnect 12.6 for Informix, MSSS, UDB does not support LDAP.

The following components have been tested for compatibility with Adaptive Server 12.5 and later:

- Sybase Character Sets 3.0
- Sybase Central viewer 4.x
- ECDA option for ODBC 12.6
- ECDA option for Informix 12.6
- MainframeConnect DirectConnect for OS/390 12.6
- ECDA option for Oracle 12.6
- ECDA option for Microsoft SQL Server 12.6
- ECDA option for DB2 Universal Database 12.6
- ECDA option for AS/400 12.6
- OpenSwitch 12.5

Note OpenSwitch 12.5 does not support SSL.

- jConnect for JDBC 4.5
- jConnect for JDBC 5.5
- Open Server™ 12.5.1
- SDK 12.5.1
- Open ServerConnect™ (CICS, IMS/MVS) 4.0
- Open ClientConnect™ (CICS, IMS/MVS) 4.0
- ODBC Driver for Adaptive Server 12.5
- OLE DB Provider for Adaptive Server 12.5
- InfoMaker 7.0.3
- Replication Server® 12.5, 12.6
- SQL Remote 7.0.02
- Physical Architect 8.0
- XA-Library for CICS/ENCINA 11.1.1
- XA-Server for Tuxedo 11.1.1

- ADO.NET 1.1

Note Backup Server 12.5 is compatible with Backup Server 11.5.1 and later. Either server can be local or remote. Backup Server 12.5 is compatible with versions of Backup Server earlier than 11.9.2.1 *only* if the Backup Server 12.5 is the remote server.

7.1 Known compatibility issues

7.1.1 Installing Replication Server with other Sybase software

You have to install the RepServer 12.5 in a separate product directory. If the RepServer is 12.6 or later, a separate product directory is not necessary. This includes the Replication Server Manager.

7.1.2 Using Job Scheduler templates

Sybase recommends upgrading to the Job Scheduler templates provided with Adaptive Server 12.5.3a. The templates contain an important change that handles the 12.5.3a version number. To upgrade, install the XML from the `$Sybase/JS-12_5/Templates/xml` directory and the template stored procedures from the `$Sybase/JS-12_5/Templates/sprocs` directory. For more information on installing the templates, see *Job Scheduler User's Guide*.

This upgrade should not interfere with any existing jobs as there were no interface or functional changes. Essentially all that happens is that existing jobs will use the updated template stored procedures, and will run as they have in the past.

Note If you have existing jobs created from templates delivered with a pre-12.5.2 Adaptive Server, you will need to refer to the 12.5.2 Release Bulletin for your platform and follow the upgrade steps there for the Job Scheduler templates. Then you can upgrade Job Scheduler templates using the above information.

8. Documentation updates and clarifications

This section describes changes and additions to the Adaptive Server documentation set.

8.1 Web Services User's Guide

The following are updates for Adaptive Server for *Adaptive Server Enterprise Web Services User's Guide*.

8.1.1 Remote server class definition setting

Include the following in the “Troubleshooting Issues” section of Chapter 7, “Troubleshooting” in the *Adaptive Server Enterprise Web Services User's Guide*.

Issue

The `sp_webservices add` command may return the following error when generating proxy tables:

```
Warning: Row size (3347 bytes) could exceed row size limit, which is 1962
bytes.
Msg 208, Level 16, State 1:
Server 'JMALVARADO', Line 1:
tempdb..ws_4338e6e122cd4ef0a not found. Specify owner.objectname or uses to
check whether the object exists (sp_help may produce lots of output).
No proxy tables were created for the WSDL URL:
[http://www.xignite.com/xquotes.asmx?WSDL]
(return status = 0)
```

This error occurs because the remote server representing the Web Services Consumer has been added using `sp_addserver` with a class other than “sds.” To verify that this is so, use `sp_helpserver` in isql:

```
1> sp_helpserver ws
2> go
```

Here, `ws` is the name of the Web Services Consumer. This is the default. The remote server class is returned in the indicated column of the result:

```
name network_name class ...
---- -
ws ws null ...
```

User action

Change the class of the remote server to “sds” by using `sp_dropserver` and `sp_addserver` in isql:

```
1> sp_dropserver ws_name
2> go
...
1> sp_addserver ws_name, sds, ws_name
2> go
```

Here, `ws_name` is the name chosen for the Web Services Consumer.

8.2 Updates for Using Sybase Failover in a High Availability System

The following are updates for Adaptive Server for *Using Sybase Failover in a High Availability System*.

8.2.1 Configuring Adaptive Server resource groups

Step 2 of the section titled “Configuring Adaptive Server resource groups” in Chapter 11, “Active-Active Configuration for Sun Cluster 3.0” and in Chapter 12, “Active-Passive Configuration for Sun Cluster 3.0” is replaced with:

If you use another file at a different location, specify the full path for the resource extension property *Dataserver_login_file* when configuring the *SYase* resource. Create or edit a file that contains Adaptive Server login information for system administrator and the user you added for the fault monitor. The default file is

```
$$SYBASE/$SYBASE_ASE/SC-3_0/etc/ase_login_file.
```

The file consists of two lines; the first line is the login and password of the System Administrator, and the second line is the login and password of *monitor_user*. The fault monitoring program, *ase_monitor*, performs the thorough probe as user *monitor_user*.

```
login_type <tab> login_string  
login_type <tab> login_string
```

Valid values for login type are “encrypted” and “normal”. If you set *login_type* to “normal”, the value of the *login_string* is in the form “login_name/password”. If you set *login_type* to “encrypted”, the value of *login_string* is the encrypted string you get from the *haisql* utility (located in *\$\$SYBASE/\$SYBASE_ASE/bin*). Sybase recommends usage of “encrypted” *login_type* so the sensitive information in the file is well protected. To use *haisql* to generate the encrypted login string:

- 1 Run *haisql* with no arguments to generate the encrypted string for a given *login_name* and *password*:

```
/$SYBASE/ASE-12_5/bin/haisql  
Enter Username: sa  
Enter Password:  
TWA58n1jSF2gBsvayUlw97861.cyTKaS1YhavBRQ2qKcJwtX.TmFBarGS2K1553WDR7g8m5  
vrf86t@K4CU62HEccm4zkeexSP9E=FeuvX
```

- 2 Copy and then paste the encrypted string to the *ase_login_file* file.

The following is an example of the *ase_login_file* using the “encrypted” login type:

encrypted

TWAS8n1jSF2gBsvayUlW97861.cyTKaS1YhavBRQ2qKcJwtx.TmFBarGS2K1553WDR7g8m5vrf86t@K4CU62HEccm4zkeexsP9E=FeuvX

encrypted

rX2S8n1jSF2gBuD0q=AXEXKCZvzGcK5K3kWnp_P+e4avf=67kYVSzy7+h640@97FSP_dlKH_oV2Zi ma5+7tUyHnsm4zmSIHIUnKSTPoTD

The following is an example of the *ase_login_file* file using “normal” login type:

```
normal    sa/sa_password
normal    monitor_user/monitor_user_password
```

Note The two lines of the *ase_login_file* may use different login types.

You should protect the *ase_login_file* file with proper access permissions, particularly if you are not using the encrypted login strings. Perform the following to make the file readable only to the root user after editing the file with proper *login_type* and *login_string* values:

```
chmod 400 ase_login_file
chown root ase_login_file
chgrp sys ase_login_file
```

8.3 Utility Guide

The following are updates for Adaptive Server for the *Utility Guide*.

8.3.1 bcp modes

Correction In the section “bcp modes” in Chapter 3, “Using bcp to Transfer Data to and from Adaptive Server” of the *Utility Guide*, the description should read:

bcp in works in one of two modes:

- Slow bcp – logs each row insert that it makes, used for tables that have one or more indexes or triggers.
- Fast bcp – logs only page allocations, copying data into tables without indexes or triggers at the fastest speed possible.

Version 12.5.1 of the book erroneously reversed the descriptions.

8.3.2 ddlgen

ddlgen, the Java-based tool that generates definitions for server- and database-level objects in Adaptive Server, has expanded its list of filters to include indexes, triggers, and constraints out of table and database definitions in the DDL of database- as well as table-level objects. The valid filters are:

-F filters out indexes, triggers, and constraints out of table and database definitions in the DDL of table- and database-level objects. The valid filters are:

- *For tables* – [% | I | RI | KC | TR]
- *For databases* – [% | SGM | GRP | USR | R | D | UDD | U | V | P | XP | I | RI | KC | TR]

The filter options are:

Filter option	Filters out:
%	Everything, and retrieves the schema-only definition of a database or table.
SGM	Segments
GRP	Groups
USR	Users
R	Rules
D	Defaults
UDD	User-defined datatypes
U	User tables
V	Views
P	Stored procedures
XP	Extended stored procedures
I	Indexes
RI	Referential integrity constraints
KC	Primary- and unique-key constraints
TR	Triggers

Examples

Generates DDL for the `authors` table in the `pubs2` database, filtering for all indexes (I), and referential integrity constraints (RI), primary and unique key constraints (KC), and triggers (TR) from the DDL of a table:

```
ddlgen -Uroy -Proyl23 -TU -Nauthors -Dpubs2 -F%
```

Alternatively, you can specify each of the filters individually:

```
ddlgen -Ulogin -Ppassword -TU -Ndbname.owner.table
```

-FI, RI, KC, TR

Note When you use the -F parameter to filter a table, then indexes, referential integrity, key constraints and triggers automatically get filtered, as they are a subset of table object.

The following generates the definition of *table_name* while filtering out foreign keys and primary-unique keys:

```
ddlgen -Ulogin -Ppassword -TU -Ntable_name -Ddbname
      -FRI, KC
```

Both of these generate foreign keys for a specified user in the entire database:

```
ddlgen -Ulogin -Ppassword -TRI -N%.%.% -Ddbname
```

Or:

```
ddlgen -Ulogin -Ppassword -TRI -Ndbname%.%.%
```

Both of these generate DDL for the primary and unique keys of all the tables in a database that begin with “PK”:

```
ddlgen -Ulogin -Ppassword -TKC -Ndbname.%.%.PK%
```

Or:

```
ddlgen -Ulogin -Ppassword -TKC -N%.%.PK% -Ddbname
```

Note Although you can normally generate all indexes only for a table, the -T object type parameter with the RI and KC value allows you to generate foreign keys as well as primary and unique keys for an entire database.

The following generates schema-only definition of a database:

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
      -F%
```

Alternatively, you can specify each of the filters individually:

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
      -FSGM, GRP, USR, R, D, UDD, V, P, XP, I, RI, KC, TR
```

The following generates the database DDL skipping the compiled object:

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
      -FTR, D, XP, V, R
```

The following generates database definition without a table definition:

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname  
-FU
```

ddlgen update

Servers The existing description of the -S parameter in *ddlgen* is:

- *-Shost_name : port_number*

specifies the host name or IP address of Adaptive Server, as well as its port number. Separate *host_name* and *port_number* with a colon, without spaces before or after it.

Note You must use the -S option because *ddlgen* does not connect to a default server.

The updated description is as follows:

- *-Sserver* – specifies the name of the Adaptive Server. *ddlgen* looks this name up in the interfaces file or LDAP configuration file. If you specify:
 - *-S [host_name:port_number]* – *ddlgen* uses the *host_name* and *port_number* provided, and neither interfaces nor LDAP configuration files are read or parsed.
 - *-S[server] -l* – *ddlgen* parses the interfaces file specified at the user location for the server name (see the -l parameter description for more information).
 - *-S[server]* – without specifying an interfaces file, *ddlgen*:
 - 1 Tries to read the LDAP configuration file from the standard location.
 - 2 If the LDAP file does not exist, or exists but does not contain an Adaptive Server entry, then the interfaces file is parsed at its standard location for the server name.
 - 3 If the LDAP file exists, then *ddlgen* uses it to search the server name. The interfaces file is not parsed, and the LDAP configuration file is parsed.

Separate *host_name* and *port_number* with a colon, without spaces before or after it.

Note You must use the `-S` option because `ddlgen` does not connect to a default server.

Interfaces files `-I` – specifies the interfaces file name, and corresponds to `$$SYBASE/interfaces` for UNIX, and `$$SYBASE%\ini\sql.ini` for Windows. Use this optional parameter with `-S`.

Indexes You can now run the `ddlgen` utility for all indexes of all tables in a database in the following command:

```
ddlgen -Ulogin -Ppassword -Sserver:port -TI
-Ndbname.%.%.%
```

For example, this generates DDL for all indexes for all tables in the `pubs2` database:

```
ddlgen -Usa -P -SHARBOR:1955 -TI -Npubs2.%.%.%
```

8.3.3 sp_sproc_columns

There is a new column in the result set for the `sp_sproc_columns` catalog stored procedure called `mode`:

Column	Datatype	Description
<code>mode</code>	<code>varchar(20)</code>	The parameter mode information stored in <code>syscolumns</code> that contains: <ul style="list-style-type: none"> • For <i>SQL procedures</i> – in, out, or “return value”. • For <i>SQLJ procedures (Java)</i> – in, out, inout, or “return value”.

8.3.4 sqldbgr

Add the following to the description of the `sqldbgr` SQL debugger utility:

Note You do not have the ability to view `sqldbgr` version strings.

8.3.5 dataserver clarification

In the `dataserver` reference page in the *Utility Guide*, Example 2 is incorrect. It currently reads:

Example 2 rewrites a corrupt model database:

```
dataserver -d my_master_device -w model
```

The correct example is:

Example 2 rewrites a corrupt model database:

```
dataserver -d d_master -w model -s server_name
```

8.3.6 Omission from *sybmigrate*

The following information was omitted from the “sybmigrate” Chapter, in the *Utility Guide*.

sybmigrate requires allow resource limits be set to 0.

Note If metadata already exists on the target Adaptive Server, you cannot migrate server data.

SYBASE_JRE – defines the location of the Java runtime environment. This is generally set to `$SYBASE/shared-1_0/jre-1_3` in the Adaptive Server release area. This environment variable overrides JAVA_HOME.

SYBASE_JRE defaults to `$SYBASE/shared-1_0/jre-1_3`.

8.3.7 Utilities moved

The following three utilities have moved from `$SYBASE/$SYBASE_ASE/bin` to `$SYBASE/$SYBASE_OCS/bin`:

- certauth
- certpk12
- certreq

8.3.8 *bcp* clarification

In the section “bcp modes” in Chapter 3, “Using bcp to Transfer Data to and from Adaptive Server” in the *Utility Guide*, the definitions of bcp in were reversed and incorrectly stated as:

bcp in works in one of two modes:

- Fast bcp – logs each row insert that it makes, used for tables that have one or more indexes or triggers.

- Slow bcp – logs only page allocations, copying data into tables without indexes or triggers at the fastest speed possible.

The correct definitions are as follows:

bcp in works in one of two modes:

- Slow bcp – logs each row insert that it makes, used for tables that have one or more indexes or triggers.
- Fast bcp – logs only page allocations, copying data into tables without indexes or triggers at the fastest speed possible.

8.4 Transact-SQL User's Guide

In “Sorting Query results” in Chapter 3, the word “aggregates” has been dropped from this sentence:

“Adaptive Server does not allow subqueries, variables and constant expressions in the order by list.” A successful example in the following section, order by and group by, uses the aggregate function avg(price) in an order by clause:

```
order by avg(price)
```

8.5 Reference Manual: Building Blocks

8.5.1 *str_replace*

The *str_replace* function now accepts NULL in the third parameter and treats it as an attempt to replace *string_expression2* with NULL, effectively turning *str_replace* into a “string cut” operation.

For example, the following returns “abcghijklm”:

```
str_replace("abcdefghijklm", "def", NULL)
```

8.5.2 *@@version_as_integer* update

In Chapter 3, “Global Variables” in *Reference Manual: Building Blocks*, the definition for @@version_as_integer states:

“Returns the version of the current release of Adaptive Server as an integer.”

This information should read:

“Returns the number of the last upgrade version of the current release of Adaptive Server as an integer. For example, @@version_as_integer returns 12500 if you are running Adaptive Server version 12.5, 12.5.0.3, or 12.5.1.”

8.5.3 set_appcontext update

In the reference page for set_appcontext in Chapter 2, “Transact-SQL Functions” in *Reference Manual: Building Blocks*, the parameter for *attribute_value* is incorrectly stated as being a char(2048). The correct datatype is char(30), and should appear as follows:

attribute_value – is a row that specifies and application attribute value. It is saved as the datatype char(30).

8.6 Reference Manual: Commands

8.6.1 where clause syntax

The following syntax is no longer applicable in the where clause.

```
where [not] time_period1 overlaps time_period2
```

The following example in the where clause reference pages is also no longer applicable:

Example 9 Compares two time periods and determines whether they overlap each other. The first period begins March 16, 1994 and lasts for one month. The second period begins March 31, 1994 and lasts until December 31, 1994. The predicate returns a value of TRUE because the two periods have points in common:

```
where (date "1994-03-16", interval +"1" month) overlaps
      (date "1994-03-31", date "1994-12-31")
```

8.6.2 dbcc upgrade object has a new parameter “check”

The following description of the dbcc check command should be included in *Reference Manual: Commands*.

Syntax

```
dbcc upgrade_object( {dbid | dbname},
                    { compiled_object_name |
                      "view" | "procedure" | "default" |
                      "rule" | "default" | "trigger" },
                    [ "force" | "check" ] )
```

Keywords and options	check – checks syntax for the specified <i>compiled_object</i> in syscomments in the specified database. Does not raise errors on occurrences of select.
Comments	<ul style="list-style-type: none">• check is used to detect syscomments text corruption which was caused by Adaptive Server defects that occurred before the upgrade. This syscomments text corruption is serious because it causes upgrade to fail.• If any error is reported by <code>dbcc upgrade_object</code> with check option, you must drop and re-create the <i>compiled_object</i>.

8.6.3 create index clarification

The create index reference page in Chapter 1, “Commands” of *Reference Manual: Commands* incorrectly states that the `max_rows_per_page` and `fillfactor` parameters are exclusive.

“`max_rows_per_page` – `max_rows_per_page` and `fillfactor` are mutually exclusive. Unlike `fillfactor`, the `max_rows_per_page` value is maintained until it is changed with `sp_chgattribute`.”

It should read:

`max_rows_per_page` – limits the number of rows on data pages and the leaf level pages of indexes. Unlike `fillfactor`, the `max_rows_per_page` value is maintained until it is changed with `sp_chgattribute`.

8.6.4 bcp reference page clarification

In the bcp reference page in Chapter 8, “Utility Commands Reference” of the *Reference Manual: Commands*, the description of the `-b batchsize` parameter has been clarified. It used to read:

“`-b batchsize` is the number of rows per batch of data copied (default is to copy all the rows in one batch).”

The text now reads:

“`-b batchsize` is the number of rows per batch of data copied. By default, bcp in copies `n` rows in one batch, where `n` is equal to the batch size.”

8.6.5 set command option *explicit_transaction_require* not supported

The set command option `explicit_transaction_require` is not currently supported, and should not appear in the *Adaptive Server Reference Guide* version 12.5.1.

8.6.6 New set parameters *proc_output_params* and *set proc_return_status*

The following new parameters should be included in the set command reference section of *Reference Manual: Commands*.

Parameters

- *proc_output_params* – controls sending of output parameters that a stored procedure generates back to the client. `set proc_output_params off` suppresses sending the output parameters back to the client. The default for this parameter is `on`.
- *proc_return_status* – controls sending of a return status TDS token back to the client. `set proc_return_status off` suppresses sending the return status token to the client, and `isql` client does not display the `(return status = 0)` message. The default for this parameter is `on`.

Warning! If the client application that executes a procedure relies on the success or failure of the procedure based on the return status, then do not use the `set proc_return_status off` option.

Example 1

Suppresses the output of both parameters and the return status TDS token:

```
1> set proc_output_params OFF
2> go

1> set proc_return_status OFF
2> go

1> declare @x int
2> select @x = 2
3> exec sp_pout @x output
4> print "Value of @x returned from sproc is: %!", @x
5> go

(1 row affected)
Value of @x returned from sproc is: 3
(1 row affected)
```

In addition, you can also suppress the lines reporting the number of rows affected to generate output with no extra messages using the `set nocount on` option before running this batch.

Example 2

Suppresses the output of parameter information:

```

1> create procedure sp_pout (@x int output) as select
  @x = @x + 1
2> go

1> set proc_output_params off
2> go
1> declare @x int
2> select @x = 1
3> exec sp_pout @x output
4> print "Value of @x returned from sproc is: %1!", @x
5> go

(1 row affected)
(return status = 0)

```

Value of @x returned from sproc is: 1

If you do not perform `set proc_output_params off`, the output after `(return status = 0)` includes the following:

Return parameters:

```

-----
                2

```

8.7 Reference Manual: Procedures

8.7.1 sp_monitorconfig

There are numerous changes and corrections to `sp_monitorconfig`. Replace the reference pages in *Reference Manual: Procedures* for `sp_monitorconfig` with this text.

Description

Displays cache usage statistics regarding metadata descriptors for indexes, objects, and databases. `sp_monitorconfig` also reports statistics on auxiliary scan descriptors used for referential integrity queries, and usage statistics for transaction descriptors and DTX participants.

Syntax

```
sp_monitorconfig "configname" [ , "result_tbl_name" ] [ , "full" ]
```

Parameters

- *configname*

is either all, or part of the configuration parameter name with the monitoring information that is being queried. Valid configuration parameters are listed in the “Usage” section. Specifying all displays descriptor help information for all indexes, objects, databases, and auxiliary scan descriptors in the server.

- “*result_tbl_name*”

is the name of the table you create, in which to save the stored procedure results. This is an optional parameter.

- “full”

returns a set of values for the configname that you specify. The values are:

- *config_val* – reports the configured value
- *system_val* – reports the systems default value when there's no value configured
- *total_val* – reports the actual value used

Examples

Example 1 Shows all items that are open:

```
sp_monitorconfig "open"
```

Configuration option is not unique.

option_name	config_value	run_value
current change w/ open cursors	1	1
number of open databases	12	12
number of open indexes	500	500
number of open objects	500	500
open index hash spinlock ratio	100	100
open index spinlock ratio	100	100
open object spinlock ratio	100	100

Example 2 Shows the status for all configurations:

```
sp_monitorconfig "all"
```

Usage information at date and time: Oct 25 2002 10:36AM.

Name	num_free	num_active	pct_act	Max_Used	Reused
additional network memory	0	0	0.00	0	NA
audit queue size	100	0	0.00	0	NA
heap memory per user	4096	0	0.00	0	No
max cis remote connection	0	0	0.00	0	NA

max memory	12404	21388	63.29	21388	NA
max number network listen	3	2	40.00	2	NA
max online engines	4	1	20.00	1	NA
memory per worker process	1024	0	0.00	0	NA
number of alarms	31	9	22.50	9	NA
number of aux scan descri	200	0	0.00	0	NA
number of devices	9	1	10.00	1	NA
number of dtx participant	500	0	0.00	0	NA
number of java sockets	0	0	0.00	0	NA
number of large i/o buffers	6	0	0.00	0	NA
number of locks	4673	327	6.54	408	NA
number of mailboxes	30	0	0.00	0	NA
number of messages	64	0	0.00	0	NA
number of open databases	6	6	50.00	6	No
number of open indexes	492	8	1.60	8	No
number of open objects	482	18	3.60	18	No
number of open partitions	447	53	10.60	0	NA
number of remote connections	20	0	0.00	0	NA
number of remote logins	20	0	0.00	0	NA
number of remote sites	10	0	0.00	0	NA
number of sort buffers	500	0	0.00	9	NA
number of user connection	23	2	8.00	2	NA
number of user processes	0	0	0.00	0	NA
partition groups	1024	0	0.00	0	NA
permission cache entries	15	0	0.00	0	NA
procedure cache size	2567	704	21.52	810	No
size of global fixed heap	150	0	0.00	0	NA
size of process object heap	1500	0	0.00	0	NA
size of shared class heap	1536	0	0.00	0	NA
size of unilib cache	0	0	0.00	0	NA
txn to pss ratio	16	0	0.00	0	NA

(return status = 0)

Example 3 Shows 283 active object metadata descriptors, with 217 free. The maximum used at a peak period since Adaptive Server was last started is 300:

```
sp_monitorconfig "open objects"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
```

Name	num_free	num_active	pct_act	Max_Used	Reused
-----	-----	-----	-----	-----	-----
number of open	217	283	56.60	300	No

You can then reset the size to 330, for example, to accommodate the 300 maximum used metadata descriptors, plus space for 10 percent more:

```
sp_configure "number of open objects", 330
```

Example 4 Shows the maximum number of index metadata descriptors, which is 44:

```
sp_monitorconfig "open indexes"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
Name          num_free  num_active  pct_act    Max_Used   Reused
-----
number of open 556      44          7.33      44         No
```

You can reset the size to 100, the minimum acceptable value:

```
sp_configure "number of open indexes", 100
```

Example 5 Shows the number of active scan descriptors as 30, though Adaptive Server is configured to use 200. Use the number of aux scan descriptors configuration parameter to reset the value to at least 32. A safe setting is 36, to accommodate the 32 scan descriptors, plus space for 10 percent more:

```
sp_monitorconfig "aux scan descriptors"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
Name          num_free  num_active  pct_act    Max_Used   Reused
-----
number of aux s 170      30          15.00     32         No
```

Example 6 Adaptive Server is configured for five open databases, all of which have been used in the current session.

```
sp_monitorconfig "number of open databases"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
Name          num_free  num_active  pct_act    Max_Used   Reused
-----
number of open 0         5           100.00    5          Yes
```

However, as indicated by the `Reused` column, you need to open an additional database. If all 5 databases are in use, an error may result, unless the descriptor for a database that is not in use can be reused. To prevent an error, reset `number of open databases` to a higher value.

Example 7 Only 10.2 percent of the transaction descriptors are currently being used. However, the maximum number of transaction descriptors used at a peak period since Adaptive Server was last started is 523:

```
sp_monitorconfig "txn to pss ratio"
```

Usage information at date and time: Apr 22 2002 2:49PM.

Name	num_free	num_active	pct_act	Max_Used	Reused
number of open	784	80	10.20	523	NA

Example 8 Using the optional parameter *result_tbl_name* to create a user table, saves the `sp_monitorconfig` result to this table:

```
create table sample_table(
    Name varchar(35), Num_free int,
    Num_active int, Pct_act char(6),
    Max_Used int, Reuse_cnt int,
    Date varchar(30))
```

The name of the table created becomes the second parameter of `sp_monitorconfig`:

```
sp_monitorconfig "number of alarms", sample_table
```

```
-----
(return status = 0)
```

```
select * from sample_table
```

```
-----
Name                Num_free  Num_active  Pct_act  Max_Used  Reuse_cnt  Date
-----
number of alarms    29       11          27.50    11        -1 Dec 4 2002 10:20AM
(1 row affected)
```

```
sp_monitorconfig "number of devices", sample_table
```

```
-----
(return status = 0)
```

```
select * from sample_table
```

```
-----
Name                Num_free  Num_active  Pct_act  MaxUsed  Reuse_cnt  Date
-----
number of alarms    29       11          27.50    11        -1 Dec 4 2002 10:20AM
number of devices   9        1           10.00    1         -1 Dec 4 2002 10:20AM
(2 rows affected)
```

The result set saved to the table accumulates until you delete or truncate the table.

Note If `sample_table` is in another database, you must provide its fully qualified name in quotes.

Example 9 Displays the `configure_value`, `system_value`, and `run_value` columns of all the configurations:

```
sp_monitorconfig "all", null, "full"
go
```

Usage information at date and time: Mar 23 2004 5:15PM.

Name	Num_active	Pct_act	Configure Value Max_Used	System Value Num_Reuse	Run Value	Num_free
additional network memory			0	496156	496156	334844
	161312	32.51	161312	0		
audit queue size			100	0	100	100
	0	0.00	0	0		
disk i/o structures			256	0	256	256
	0	0.00	55	0		
heap memory per user			4096	563	4096	4096
	0	0.00	0	0		
max cis remote connection			0	100	100	100
	0	0.00	0	0		
max memory			33792	0	33792	3452
	30340	89.78	30340	0		
max number network listen			5	0	5	4
	1	20.00	1	0		
max online engines			1	0	1	0
	1	100.00	1	0		
memory per worker process			1024	68	1024	1024
	0	0.00	0	0		
number of alarms			40	0	40	30
	10	25.00	10	0		
number of aux scan descri			200	25	200	200
	0	0.00	0	0		
number of devices			10	0	10	9
	1	10.00	1	0		
number of dtx participant			500	0	500	500
	0	0.00	0	0		
number of java sockets			0	50	50	50

	0	0.00	0	0		
number of large i/o buffers	6		0	0	6	6
	0	0.00	1	0		
number of locks	5000		334	0	5000	4905
	95	1.90	446	0		
number of mailboxes	30		1	0	30	29
	1	3.33	1	0		
number of messages	64		1	0	64	64
	0	0.00	0	0		
number of open databases	12		0	0	12	6
	6	50.00	6	0		
number of open indexes	500		0	0	500	481
	19	3.80	45	0		
number of open objects	500		0	0	500	135
	365	73.00	367	0		
number of open partitions	500		0	0	500	447
	53	10.60	56	0		
number of remote connections	20		0	0	20	20
	0	0.00	0	0		
number of remote logins	20		0	0	20	20
	0	0.00	0	0		
number of remote sites	10		0	0	10	10
	0	0.00	0	0		
number of sort buffers	500		0	0	500	500
	0	0.00	9	0		
number of user connections	25		0	0	25	24
	1	4.00	3	0		
number of worker processes	0		0	0	0	0
	0	0.00	0	0		
partition groups	1024		0	0	1024	1024
	0	0.00	0	0		
permission cache entries	15		0	0	15	15
	0	0.00	0	77		
procedure cache size	3271		0	0	4727	0
	4727	100.00	8225	277		
size of global fixed heap	150		0	0	150	150
	0	0.00	0	0		
size of process object heap	1500		0	0	1500	1500
	0	0.00	0	0		
size of shared class heap	1536		0	0	1536	1536
	0	0.00	0	0		
size of unilib cache	0		119386	0	119386	118922
	464	0.39	464	0		
txn to pss ratio	16		0	0	16	400
	0	0.00	0	0		
(return status = 0)						

Usage

- If the `max cis remote connections` configuration parameter has a `config_value`, the `system_val` reports a value of zero (0).
- If you reconfigure a resource using a value that is smaller than the original value it was given, the resource does not shrink, and the `Num_active` configuration parameter can report a number that is larger than `Total_val`. The resource shrinks and the numbers report correctly when Adaptive Server restarts.
- `sp_monitorconfig` displays cache usage statistics regarding metadata descriptors for indexes, objects, and databases, such as the number of metadata descriptors currently in use by the server.
- `sp_monitorconfig` also reports the number of auxiliary scan descriptors in use. A scan descriptor manages a single scan of a table when queries are run on the table.
- `sp_monitorconfig` monitors the following resources:
 - additional network memory
 - audit queue size
 - heap memory per user
 - max cis remote connection
 - max memory
 - max number network listeners
 - memory per worker process
 - max online engines
 - number of alarms
 - number of aux scan descriptors
 - number of devices
 - number of dtx participants
 - number of java sockets
 - number of large i/o buffers
 - number of locks
 - number of mailboxes
 - number of messages
 - number of open databases
 - number of open indexes
 - number of open objects
 - number of open partitions
 - number of remote connections
 - number of remote logins
 - number of remote sites
 - number of sort buffers
 - number of user connections

number of worker processes
partition groups
permission cache entries
procedure cache size
size of global fixed heap
size of process object heap
size of shared class heap
size of unilib cache
txn to pss ratio

- The columns in the `sp_monitorconfig` output provide the following information:
 - `num_free` – specifies the number of available metadata or auxiliary scan descriptors not currently used.
 - `num_active` – specifies the number of metadata or auxiliary scan descriptors installed in cache (that is, active).
 - `pct_active` – specifies the percentage of cached or active metadata or auxiliary scan descriptors.
 - `Max_Used` – specifies the maximum number of metadata or auxiliary scan descriptors that have been in use since the server was started.
 - `Reused` – specifies whether a metadata descriptor was reused in order to accommodate an increase in indexes, objects, or databases in the server. The returned value is `Yes`, `No` or `NA` (for configuration parameters that do not support the reuse mechanism, such as the number of `aux` scan descriptors).
- Use the value in the `Max_Used` column as a basis for determining an appropriate number of descriptors; be sure to add about 10 percent for the final setting. For example, if the maximum number of index metadata descriptors used is 142, you might set the number of open indexes configuration parameter to 157.

- If the `Reused` column states `Yes`, reset the configuration parameter to a higher value. When you need to reuse descriptors, there can be performance problems, particularly with open databases. An open database contains a substantial amount of metadata information, which means that to fill up an open database, Adaptive Server needs to access the metadata on the disk many times; the server can also have a spinlock contention problem. To check for spinlock contention, use the system procedure `sp_sysmon`. For more information, see the *Performance and Tuning Guide*. To find the current number of indexes, objects, or databases, use `sp_countmetadata`.
- To get an accurate reading, run `sp_monitorconfig` during a normal Adaptive Server peak time period. You can run `sp_monitorconfig` several times during the peak period to ensure that you are actually finding the maximum number of descriptors used.
- `result_tbl_name` creates a table using the following syntax. All the result information is saved in this table, which returns no standard output.

```
create table table_name(
    Name varchar(35), Num_free int,
    Num_active int, Pct_act char(6),
    Max_Used int, Reuse_cnt int,
    Date varchar(30))
```

- Some configuration parameter, such as *number of sort buffers* and *txn to pss ratio*, are dependent on the number of configured user connections, while other configuration parameters, such as *max number of network listeners*, are per engine.
- The output of `sp_monitorconfig` uses the number of user connections and online engines to calculate the values for the columns `num_free`, `num_active`, `pct_act` and `max_used`.
- The updates on the internal monitor counters are done without using synchronization methods because of performance reasons. For this reason, a multi-engine Adaptive Server under heavy load might report numbers in the `sp_monitorconfig` output that are not a completely accurate.
- You might see the number of active locks as greater than 0 on an idle system. These “active” locks are reserved and used internally.

Permissions

Only a System Administrator can execute `sp_monitorconfig`.

See also

System procedures `sp_configure`, `sp_countmetadata`, `sp_helpconfig`, `sp_helpconstraint`, `sp_sysmon`

8.7.2 sp_autoformat

sp_autoformat is a stored procedure introduced in Adaptive Server version 12.5.1.

Description A utility stored procedure that produces readable result set data, sp_autoformat reformats the width of variable-length character data to display only non-blank characters. Trailing blanks are truncated in the output.

Syntax sp_autoformat @fulltabname [, @selectlist , @whereclause , @orderby]

Parameters @fulltabname
 specifies the name of table from which data is being selected. Use owner names if the object owner is not the user running the command.

@selectlist
 specifies the comma-separated list of columns selected in the result set. Columns in the table can be renamed using the <name> = <column> notation. See examples. If @selectlist is not provided, all columns in the table specified are output in column ID order.

@whereclause
 is a search predicate, specified as a where clause, that filters out rows from the table being selected.

@orderby
 is an optional order by clause that specifies the order in which the output result set is presented.

Examples **Example 1** Returns a result set from a select statement similar to select id, colid, name from syscolumns where id = 3, where the character columns are autoformatted:

```
1> sp_autoformat "syscolumns", "id, colid, name", "where id = 3"
2> go
```

```

id          colid  name
-----
          3      1 id
          3      2 number
          3      3 colid
          3      4 status
          3      5 type
          3      6 length
          3      7 offset
          3      8 usertype
          3      9 cdefault
          3     10 domain
          3     11 name
```

```

3      12 printfmt
3      13 prec
3      14 scale
3      15 remote_type
3      16 remote_name
3      17 xstatus
3      18 xtype
3      19 xdbid
3      21 accessrule
3      22 status2

```

Example 2 Renames the output columns using the following syntax:

```
[ < AS-Name label of Column> ][ ]*=[ ]*<column name>
```

<AS-Name label of Column> is optional, and you can use white spaces around the = separator:

```

1> sp_autoformat syscolumns, "'Object Id' = id, 'Column Name'=name,
   'Column ID'=colid", "where id = 3"
2> go

```

Object Id	Column Name	Column ID
-----	-----	-----
3	id	1
3	number	2
3	colid	3
3	status	4
3	type	5
3	length	6
3	offset	7
3	usertype	8
3	cdefault	9
3	domain	10
3	name	11
3	printfmt	12
3	prec	13
3	scale	14
3	remote_type	15
3	remote_name	16
3	xstatus	17
3	xtype	18
3	xdbid	19
3	accessrule	21
3	status2	22

(1 row affected)

Example 3 Uses the `@orderby` parameter to specify an ordering in the result output:

```
sp_autoformat @fulltabname = 'syscolumns',
              @selectlist = "id, name",
              @orderby = "ORDER BY name"
```

Example 4 To generate an autoformatted result when you select from multiple tables, or if you have a complex SQL `select` statement with expressions in the `select` list, you must:

- 1 Use temporary tables to generate the result set:

The following generates the list of the columns with the highest column ID on all system catalogs:

```
select o.id, o.name, c.colid, c.name
from sysobjects o, syscolumns c
where o.id < 100 and o.id = c.id
      and c.colid = (select max(c2.colid) from syscolumns c2
                    where c2.id = c.id)
order by o.name
```

The following generates the same result set with auto-formatting of character data using a temporary table to produce readable output, and includes minor changes to provide column names in the temporary table:

```
select o.id, ObjectName = o.name, c.colid, ColumnName = c.name
into #result
from sysobjects o, syscolumns c
where o.id < 100 and o.id = c.id
      and c.colid = (select max(c2.colid) from syscolumns c2
                    where c2.id = c.id)
```

- 2 Use `sp_autoformat` on that temporary table to produce formatted output:

The `order by` clause in the original `select` statement is skipped when generating the temporary table, and is instead added to the call to `sp_autoformat` when generating the output result.

```
1> exec sp_autoformat @fulltabname = #result, @orderby = "order by
      ObjectName"
2> go
```

```
      id      ObjectName      colid ColumnName
-----
      11 sysalternates      2 altsuid
```

```

21 sysattributes          13 comments
55 syscertificates      6  suid
45 syscharsets          8  sortfile
3  syscolumns           22  status2
6  syscomments          8  status
37 sysconfigures        9  value4
17 sysconstraints       7  spare2
38 syscurconfigs        15  type
30 sysdatabases         19  status4
12 sysdepends            10  readobj
35 sysdevices           7  mirrorname
43 sysengines           12  starttime

```

...

```

(1 row affected)
(return status = 0)

```

You can further process the temporary table to report only on the required output for selected tables, as shown below:

```

1> exec sp_autoformat #result, "id, 'Object Name' = ObjectName,
   'Column Name' = ColumnName", "where id < 5"
2> go

```

```

id      Object Name Column Name
-----
1  sysobjects  loginame
2  sysindexes  crdate
3  syscolumns  status2
4  systypes    accessrule

```

Usage

- `sp_autoformat` looks for an object only in the current database. To use `sp_autoformat` on temporary tables, issue the procedure from `tempdb`.
- `sp_autoformat` does not validate that the columns referenced in any of the parameters actually exist in the table specified by the `@fulltablename` parameter. `sp_autoformat` fails if you reference any nonexistent columns.
- Provide only one instance of a column in the `select` list.

Return codes

- 0 – successful completion
- 1 – internal error, or usage error in invocation
- Other – any other errors raised by Adaptive Server during the execution of the generated SQL statement are returned back to the caller.

Restrictions

- `sp_autoformat` uses internal SQL variables to generate SQL statements that are then executed using `execute immediate`. The length of the generated SQL statement is limited to 2K bytes. Auto-formatting result sets for a large column list, or columns with long names can sometimes cause an error due to insufficient size of the buffer for the generated SQL statement.
- Quoted identifiers are not supported for either the table or column names. If you have result sets that use quoted identifiers and that need autoformatting, you must:
 - a Generate the required data in a temporary table, where the columns in the temporary table do not have any quoted identifiers.
 - b Use `sp_autoformat` to produce the required output using the temporary table.
 - c Rename the columns in the `@selectlist` in the desired output format.

Permissions

Any user can execute `sp_autoformat`. However, users selecting from the tables must have appropriate `select` privileges.

8.7.3 sp_helpdb

The `sp_helpdb` stored procedure now includes the `order` parameter. Add the following to the reference page for `sp_helpdb` in *Reference Manual: Procedures*. The full syntax of `sp_helpdb` is:

```
sp_helpdb [dbname, order]
```

The default order of the output is by `lstart`, which is the order in which the databases were created or altered. Use `device_name` along with `dbname` to display the output of `sp_helpdb` ordered by `device_name`.

Example Specifies `device_name` for the `order` parameter to display the device fragments for `mydb` in alphabetical order, overriding the default sort order of `sp_helpdb`.

```
sp_helpdb mydb, device_name
```

```
name                db_size  owner  dbid  created          status
-----
mydb                4.5 MB   sa     5     Feb 27, 2003    no options set
```

```
(1 row affected)
```

```
device_fragments  size      usage          created          free kbytes
-----
A                 1.5 MB    data only     Feb 27 2003    7:50AM         1530
```


B	1.0 MB	log only	Feb 27 2003	7:50AM	not applicable
C	2.0 MB	data only	Feb 27 2003	7:50AM	846

8.8 System Administration Guide

8.8.1 Table information incomplete

In the *System Administration Guide*, Chapter 12, “Auditing,” Table 12-6, Values in event and extrainfo columns, is incomplete.

Table 12-6 should include these lines for the create index entry under:

“Other Information: Index name”

8.8.2 Size information incorrect in procedure

In the *System Administration Guide*, Chapter 26, “Checking Database Consistency,” some size information is incorrect in the “Creating the dbccdb database” procedure. The procedure should read:

```
* In step 1, output:
Recommended size for dbccdb is 23MB (data = 21MB, log
= 2MB)
* In step 3, user input:
...
size = "4096"
...
size="1024"
* In step 7 user input:
use dbccdb|
go|
sp_dbcc_createws dbccdb, scanseg, scan_pubs2, scan,
"64K"|
sp_dbccvreatews dbccdb, textseg, text_pubs2, text,
"64K"|
```

8.9 Performance and Tuning Guide

8.9.1 OAM-scans on DOL tables

OAM-pages are used when performing a table scan on a DOL-table. These OAM-pages refer to allocation-units, which must then be scanned to find the extents on the pages for the table being scanned. When the scan is started, a list of pages is generated the information in the allocation-page. Any newly allocated pages produced by another process in this allocation-unit are not seen by the scanning process as they were not allocated when the scan began. This is due to the way DOL-tables are implemented. If you want to include the newly allocated pages, perform one of the following:

- Use isolation level 3 type of scans (select with holdlock)
- Add an index to the DOL-table which can be used to prevent a table-scan
- Convert the DOL-table to an APL-table

8.9.2 Multiple temporary databases

The information on multiple tempdbs was inadvertently included in the *Performance and Tuning Guide*. This information is being moved to the *System Administration Guide* in the next update.

Login bindings to multiple tempdbs

In prior versions only a user with the “sa” role could be bound to a tempdb.

As of this version of Adaptive Server, any user can also be bound to an application at the time of login. For example:

If there is a binding for login “user1” and for application isql, when “user1” logs in with application isql the “user1” binding determines the tempdb assigned to the session, not the isql binding.

8.9.3 Optimizer and Abstract Plans

Correction: In Chapter 11, “Multiple Temporary Databases,” the “Unchanged yet notable procedures” section noted the following for the sp_changedowner stored procedure:

You cannot map system databases, including tempdb to a default location. However, you can map user-created temporary databases.

This information is incorrect.

8.10 XA Interface Integration Guide for CICS, Encina, and TUXEDO

In Chapter 4, “Application Programming Guidelines in the *XA Interface Integration Guide for CICS, Encina, and TUXEDO*, the introductory paragraph for the section, “Tightly coupled transactions,” has been revised. The new text is emphasized.

8.10.1 Tightly coupled transactions

The XA environment treats each thread or process that works on a transaction as a transaction branch. Each transaction branch is assigned a different `xid` and works independently of the other branches. However, all branches are committed or rolled back as a unit.

Note This applies to MTS/COM+ environments only, and only if you are using cllib-based drivers that are older than version 3.6.

9. Technical support

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support. If you have any questions about this installation or if you need assistance during the installation process, ask the designated person to contact Sybase Technical Support or the Sybase subsidiary in your area.

10. Other sources of information

Use the Sybase Getting Started CD, the Sybase Technical Library CD, and the Technical Library Product Manuals Web site to learn more about your product:

- The Getting Started CD contains release bulletins and installation guides in PDF format, and may also contain other documents or updated information not included on the Technical Library CD. It is included with your software. To read or print documents on the Getting Started CD, you need Adobe Acrobat Reader (downloadable at no charge from the Adobe Web site, using a link provided on the CD).
- The Technical Library CD contains product manuals and is included with your software.

Refer to the *Technical Library Installation Guide* in your documentation package for instructions on installing and starting the Technical Library.

- The Technical Library Product Manuals Web site is an HTML version of the Technical Library CD that you can access using a standard Web browser. In addition to product manuals, the site includes links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and the Sybase Developer Network.

To access the Technical Library Product Manuals Web site, go to Product Manuals at <http://www.sybase.com/support/manuals/>.

10.1 Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

❖ Finding the latest information on product certifications

- 1 Point your Web browser to Technical Documents at <http://www.sybase.com/support/techdocs/>.
- 2 Select Products from the navigation bar on the left.
- 3 Select a product name from the product list and click Go.
- 4 Select the Certification Report filter, specify a time frame, and click Go.
- 5 Click a Certification Report title to display the report.

❖ Creating a personalized view of the Sybase Web site (including support pages)

Set up a MySybase profile. MySybase is a free service that allows you to create a personalized view of Sybase Web pages.

- 1 Point your Web browser to Technical Documents at <http://www.sybase.com/support/techdocs/>.
- 2 Click MySybase and create a MySybase profile.

10.2 Sybase EBFs and software maintenance

❖ Finding the latest information on EBFs and software maintenance

- 1 Point your Web browser to the Sybase Support Page at <http://www.sybase.com/support>.
- 2 Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.
- 3 Select a product.

- 4 Specify a time frame and click Go. A list of EBF/Maintenance releases is displayed.

Padlock icons indicate that you do not have download authorization for certain EBF/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the “Technical Support Contact” role to your MySybase profile.

- 5 Click the Info icon to display the EBF/Maintenance report, or click the product description to download the software.

