

Release Bulletin Adaptive Server Enterprise Version 12.5 for Linux

Document ID: 78565-01-1250-02

Last revised: August, 2001

Topic	Page
1. Accessing current release bulletin information	2
2. Component summary	3
2.1 Installation kit	3
2.2 Server components	4
2.3 PC-client components	4
2.4 Operating system updates	5
3. Special installation instructions	5
3.1 Known installation issues	6
4. Special upgrade instructions	7
4.1 Migrating data between Adaptive Servers with different logical page sizes	7
4.2 Known upgrade issues	9
5. Changed functionality in this version	10
5.1 Changes affecting applications after upgrade	11
6. Known problems	11
6.1 Rebuilding system databases on upgraded servers	11
6.2 Highlighted known RPC-related problems	20
6.3 Highlighted known SSL problems	20
6.4 Highlighted known Sybase Central problems	22
6.5 Highlighted known alter table problems	23
6.6 Highlighted known select problems	24

Copyright 1989-2001 by Sybase, Inc. All rights reserved. Sybase, the Sybase logo, Data Workbench, InfoMaker, PowerBuilder, Powersoft, SQL Advantage, SQL Debug, Transact-SQL, Adaptive Server, Adaptive Server Anywhere, Adaptive Server Enterprise, Adaptive Server Enterprise Monitor, Adaptive Server IQ, AnswerBase, Anywhere Studio, Backup Server, ClearConnect, Client-Library, DB-Library, dbQueue, DirectConnect, Embedded SQL, Enterprise Client/Server, EnterpriseConnect, InformationConnect, Jaguar CTS, jConnect, KnowledgeBase, MainframeConnect, MAP, Net-Gateway, Net-Library, ObjectConnect, OmniConnect, OmniSQL Access Module, Open Client, Open ClientConnect, Open Client/Server, Open Gateway, Open Server, Open ServerConnect, PC DB-Net, PowerDesigner, Replication Agent, Replication Driver, Replication Server, Replication Server Manager, RW-Library, Secure SQL Server, Security Guardian, SQL Remote, SQL Server, SQL Server Manager, SQL Toolset, Sybase Central, Sybase SQL Desktop, Sybase SQL Workgroup, System 10, System 11, Watcom SQL, Web.SQL, Workgroup SQL Server, XA-Library, XA-Server, and XP Server are trademarks of Sybase, Inc. Other product names used herein may be trademarks or registered trademarks of Sybase or other companies. 3/01

Topic	Page
6.7 Turning off the O_SYNC flag	25
6.8 unichar does not work across platforms	26
6.9 sortkey() truncates result sets from wide columns	26
6.10 Restrictions on login trigger	26
6.11 Engine not exiting after ueshutdown()	26
6.12 Dynamic and host variables not allowed in DDL statements	27
7. Product compatibilities	27
8. Documentation updates and clarifications	29
8.1 Full Text Search Specialty Data Store User's Guide	29
8.2 Performance and Tuning Guide	32
8.3 Quick Reference Guide	39
8.4 Reference Manual Volume 1: Building Blocks	39
8.5 Reference Manual Volume 2: Commands	40
8.6 Reference Manual Volume 3: Procedures	45
8.7 Reference Manual Volume 4: Tables	53
8.8 System Administration Guide	55
8.9 Transact-SQL User's Guide	69
8.10 Utility Guide	70
8.11 What's New in Sybase Adaptive Server Enterprise?	71
9. Technical support	74
10. Other sources of information	74
10.1 Sybase certifications on the Web	74

1. Accessing current release bulletin information

A more recent version of this release bulletin may be available on the World Wide 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.

❖ To access 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 "Platform-Specific" collection for the product version you are interested in.

- 4 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. Component summary

Enclosed is Sybase® Adaptive Server® Enterprise version 12.5. 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™, jisql, and Ribo, see the jConnect documentation.

This release bulletin provides the latest information about Adaptive Server and its known problems, as well as documentation updates and clarifications.

Note Adaptive Server version 12.5 ships with the incorrect version of *ins_syn_sql*. This is the script that installs the *sp_syntax* system procedure. You can download the correct version from the Sybase Downloads Web site at <http://www.sybase.com/downloads>.

XA-Server and
XA-Library

If you use XA-Server™ or XA-Library™, you should migrate to the XA interface that is included with Open Client version 12.5, and which relies on the optional Adaptive Server Enterprise distributed transaction management (DTM) feature. Although XA-Library and XA-Server work with Adaptive Server version 12.5, they are usable only with version 11 of Open Client and will be discontinued. Contact your Sybase sales representative or customer service about obtaining a license for DTM.

2.1 Installation kit

The installation kit includes:

- The server CD – for contents list, see “Server components” on page 4.
- The PC-client CD – contains software client components to be installed on Windows 98 and Windows NT computers.
- *Release Bulletin Adaptive Server Enterprise Version 12.5 for Linux* (this document).
- *Installation Guide Adaptive Server Enterprise 12.5 for Linux*.

2.2 Server components

The Server CD includes the following components:

- Adaptive Server 12.5
- Adaptive Server Enterprise Monitor™ Server 12.5
- Monitor Client Library 12.5
- Historical Server 12.5
- Open Client™ 12.5 SDK
- Embedded SQL™/C 12.5 SDK
- SQL Remote™ 7.0.2
- Language Modules 12.5
- jConnect for JDBC 4.5
- jConnect for JDBC 5.5

Note Sybase no longer ships or supports the Cascade Web Server Gateway with jConnect.

- Java utilities:
 - jisql 2.0
 - Ribo 2.0
- Sybase Central™ 3.2

2.3 PC-client components

The PC-client CD includes the following components, which can be installed on Windows 98 and Windows NT client computers:

- Open Client 12.5 SDK
- Open Client Runtime 12.5
- Embedded SQL/C 12.5 SDK
- Embedded SQL/Cobol 12.5 SDK
- ODBC 3.7
- OLE DB 2.5

- jConnect 4.5 and related documentation
- jConnect 5.5 and related documentation
- Java utilities:
 - jsql 2.0
 - Ribo 2.0
- InfoMaker® 7.0.2
- PowerDynamo™ 3.5.2
- Language Modules 12.5
- XA Interface Library for ASE Distributed Data Manager 12.5
- Physical Architect™ 8.0
- SQL Remote 7.0.2
- Monitor Client Library 12.5

Note Java Central is installed with PowerDynamo, not Adaptive Server. The only plug-in that comes with Adaptive Server is the Adaptive Server Enterprise plug-in for Sybase Central Java Edition 12.5.

2.4 Operating system updates

Linux requires the following operating system kernel to run Adaptive Server 12.5 components:

- kernel-2.2.14-5.0

For best performance, use JRE version 1.1.6 or higher.

If your operating system requires patches, install them before you install Adaptive Server components.

3. Special installation instructions

This section provides information for installing Adaptive Server 12.5 that is not included or should be corrected in the installation guide.

3.1 Known installation issues

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

3.1.1 Studio Installer fails to load on Pentium 4 machines

Studio Installer does not work on Pentium 4 machines running any Windows operating systems. An error occurs in *java.exe* when the Studio Installer attempts to install Sybase products on a Pentium 4 machine. The error message may be similar to one of the following:

On Dr. Watson for Windows NT:

```
An application error has occurred and an application
error log is being generated. java.exe Exception: access
violation90xc0000005), Address: 0x500bf974
```

On MS-DOS:

```
Error: java.lang.ExceptionInInitializerError

java.lang.ExceptionInInitializerError
at java.util.TimeZone.getDefault<TimeZone.java:192>
at
at Logfile.println<Logfile.java:148>
at Installer.main<Compiled Code>
```

For a list of the products affected, a full description of the problem, and a workaround, see the Sybase Web site at <http://www.sybase.com/detail/1,3693,1013241,00.html>.

3.1.2 Input cursor not always visible in Studio Installer

[Bug #204713] Under some circumstances, no input cursor is visible in the text field of the dialog box. This error is caused by interactions between the Java libraries and the platform-specific Windows libraries. Usually, it is not a problem if you are directly connected to the system on which you are running the installer. The cursor is set to the first text field, so when you start typing, the characters are entered into that field.

Workaround: If no cursor is visible, click in the field in the dialog box before entering characters. Alternatively, you can minimize the Studio Installer window, then maximize it.

3.1.3 */etc/rc.d/init.d/sybase* incorrectly defines \$SYBASE variables

[Bug #241560] The file, */etc/rc.d/init.d/sybase* on Linux incorrectly configures the values for SYBASE_ASE and SYBASE_OCS. After installation you must redefine these to the following:

- SYBASE_ASE=ASE
- SYBASE_OCS=OCS

3.1.4 Using Adaptive Server with transaction services

To use Adaptive Server with a transaction monitor such as Tuxedo, Encina, or Microsoft MTS/DTC, you must:

- Obtain a license for DTM use.
- Configure Adaptive Server for DTM.
- Give the user `dtm_tm_role`.

See *Using Adaptive Server Distributed Transaction Management Features* for more information.

4. Special upgrade instructions

This section provides information for upgrading to Adaptive Server 12.5 that is not included or should be corrected in the installation guide.

4.1 Migrating data between Adaptive Servers with different logical page sizes

Adaptive Server version 12.5 supports multiple page sizes; however, you cannot upgrade directly from an earlier version of Adaptive Server with 2K logical page size, to a version 12.5 Adaptive Server with a larger logical page size.

A migration utility, `sybmigrate`, should be available in the near future to help you with this migration process.

Note The *Utility Guide* refers to `sybmigrate`, the data migration utility, as `dataxtr`.

In the meantime, you must manually rebuild existing servers and databases, and transfer data out of the 2K logical page size Adaptive Server and into the new and larger logical page size Adaptive Server using the bcp utility. See the *Utility Guide* for information on how to use bcp. See the *System Administration Guide*, and the *Performance and Tuning Guide* for more information about multiple page sizes.

The steps required to manually re-create an Adaptive Server installation for migration are nearly identical to those you perform when recovering from a corrupted server installation, as described in the *Troubleshooting and Error Messages Guide*.

❖ **To manually re-create an Adaptive Server installation for migration**

- 1 Save all device, login, and user information from the source Adaptive Server using bcp, using the -c command-line trace flag.
- 2 Save other server-wide information, such as character set information, attribute information, and server configuration information.
- 3 Build a new 12.5 Adaptive Server installation with a larger logical page size.
- 4 Install the server-wide information that you saved in step 2, on the new target server.
- 5 Save the data definition language (DDL) for all tables in the source databases. DDL may already be available at user sites. If not, use the ddlgen utility to re-create it for the entire destination Adaptive Server, as well as all the databases within it.

For more information on ddlgen, see the *Utility Guide*. See the documentation for Sybase Central for information on how to use the DDL generation feature.

Note Although ddlgen is a command-line utility, you must install Sybase Central to run it.

- 6 Re-create the database schema in the target Adaptive Server using the DDL saved in step 5 to create a skeletal database installation on the target Adaptive Server that does not include any user data.
- 7 In the source Adaptive Server, extract the data from the table to migrate using bcp out.
- 8 In the target Adaptive Server, load the previously extracted data into the new tables using bcp in.

- 9 Rebuild the indexes for all tables on the target Adaptive Server.
- 10 Reinstall user procedures, triggers, and other compiled objects on the target Adaptive Server.
- 11 Run dbcc to validate that the target database is working properly.

Note dump database and load database are not supported across different logical page size Adaptive Server platforms.

4.2 Known upgrade issues

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

4.2.1 Upgrading Monitor Server

[Bug #241321] When you upgrade Monitor Server using the sqlupgrade utility, you may see the following error message:

```
getMsg: unable to find message for code 528336
```

This message indicates that sqlupgrade successfully installed the *installmon* script; you can ignore this message, as it does not indicate an installation error.

4.2.2 OAM counts mismatch on *sysobjects* after upgrade

[Bug #239096] After upgrading to Adaptive Server version 12.5, you may see a message reporting an object allocation map (OAM) mismatch for the *sysobjects* table when you run dbcc checkalloc.

Workaround: First, put the database in single-user mode using sp_dboption. Then, run dbcc tablealloc with the fix option to correct the mismatch:

```
dbcc tablealloc(sysobjects, "full", "fix")
```

4.2.3 Using *diagserver* after upgrade

[Bug #241131] If you use *diagserver* after upgrading to Adaptive Server version 12.5, there may be assertion failures from the statistics code when you access *sysusers*, *sysalternates*, or *syslogins*. The assertion errors may also be present when you re-run *installmaster* as a part of the upgrade process for some of the system procedures.

If you use the `dataserver` binary, you might see some system stored procedures such as `sp_adduser`, `sp_addlogin`, `sp_addalias`, and `sp_dropuser` run slowly due to incorrect statistical information stored in `sysstatistics` on these catalogs.

Incorrect statistics are gathered on these catalogs due to the assertion failures. If you run `optdiag` against these catalogs after upgrade to 12.5, an error is returned:

```
Internal error: data length mismatch in ct_fetch().
CTLIB Message: - L4/O2/S6/N36/6/0:
ct_send(): protocol specific layer: internal Client
Library
error: There is a tds state machine error. An illegal
tds token sequence was received.
Optdiag failed. Check for error messages above.
```

The incorrect histogram information was saved in `sysstatistics` on these catalogs.

Workaround: Running `update statistics` does not fix this problem; use the following commands.

On the master database:

```
delete statistics master.dbo.syslogins
update statistics master.dbo.syslogins

delete statistics master.dbo.sysusers
update statistics master.dbo.sysusers

delete statistics master.dbo.sysalternates
update statistics master.dbo.sysalternates
```

On all user databases:

```
delete statistics dbname.dbo.sysalternates
update statistics dbname.dbo.sysalternates

delete statistics dbname.dbo.sysuser
update statistics dbname.dbo.sysuser
```

5. Changed functionality in this version

This section describes changed functionality with version 12.5 of Adaptive Server that is not included in *What's New in Adaptive Server 12.5?*

5.1 Changes affecting applications after upgrade

This section describes system changes introduced by Adaptive Server that may affect your applications if you are upgrading from an earlier version.

5.1.1 Increased optimization time for queries with many join keys

Queries having long chains of join keys may require additional time to optimize with Adaptive Server. If the time required to optimize such a query is unacceptable, consider using an abstract query plan for the query. For more information, see the *Performance and Tuning Guide*.

5.1.2 ANSI outer join syntax recommended

Adaptive Server provides support for ANSI outer joins. Sybase recommends that applications make use of the ANSI outer join syntax. For more information, see the *Transact-SQL User's Guide*.

6. Known problems

The following sections describe known problems and workarounds for Adaptive Server.

6.1 Rebuilding system databases on upgraded servers

The procedure described in this section is needed only for a server that has been upgraded to version 12.5 from an earlier server version. This procedure replaces the procedure in the *Error Messages and Troubleshooting Guide* for situations when the master database is corrupt, the master device is not affected, and one or both of the following conditions are true:

- There is no current backup of the master database.
- One or more user databases use space on the master device, and there are no current backups of those databases.

If you have current dumps of the master database and all databases that occupy space on the master device, you can use 12.5 `dataserver` to build a new master device. You can then load your dump of the master database and your database dumps.

Adaptive Server version 12.5 does not use the `buildmaster` binary to build the master device. Instead, Sybase has incorporated the `buildmaster` functionality in the `dataserver` binary.

The `dataserver` options you use to rebuild corrupt databases are:

```
-d master device path name
-b master device size
-w master | model
-z page size
-f[orcebuild]
```

See Chapter 1, “Building Servers Using `dataserver`” in the *Utility Guide* for more information, and the reference page for `dataserver` in the *Utility Guide* for information on how to use the options.

In Adaptive Server version 12.5, `dataserver` requires the following information to rebuild your databases:

- master device size when it was first built
- Logical page size of your current server’s installation

6.1.1 Understanding sizes

Adaptive Server version 12.5 stores the size of the master device in the config block. How you use the `dataserver -b` option depends on whether you are rebuilding a corrupted system database in a server installation that was:

- First built in version 12.5, or
- Upgraded to version 12.5.

With servers that were first built in version 12.5, `dataserver` retrieves and uses the master device size from the config block. If the config block is corrupt, use the `dataserver -b` option.

With servers that were upgraded to version 12.5, the master device size is written to the config block as part of the upgrade.

In master devices built in version 12.5 as well as for servers upgraded to version 12.5, Adaptive Server reports the size of the master device in the server’s error log that was created during the upgrade process to version 12.5, and looks like the following:

```
00:00000:00001:2001/06/07 19:27:46.95 server  WARNING:
*****
00:00000:00001:2001/06/07 19:27:46.96 server  SQL Server booted against master
device for different release.
This message is to be expected during an upgrade.
SQL Server version 12.5.0.0 Master device version 12.0.0.0
00:00000:00001:2001/06/07 19:27:46.96 server  WARNING:
*****
```

```
00:00000:00001:2001/06/07 19:27:46.98 server Master device size: 156.250000
megabytes, or 80000 virtual pages. (A virtual page is 2048 bytes.)
```

Once you obtain the size of the master device, include this information in the `-b` option when you use `dataserver` during the rebuild process.

Specifying logical page sizes for system databases

Adaptive Server version 12.5 supports 2K, 4K, 8K, and 16K logical page sizes. master devices built using Adaptive Server version 12.5 store information about its logical page size in the config block. The server's error log contains the system database's logical page size:

```
00:00000:00001:2001/06/07 19:27:44.80 server The
logical pagesize of the server is 2 Kb.
```

If you are rebuilding a corrupt system database in a server that was upgraded to version 12.5, `dataserver` uses the default logical page size of 2K.

If you are rebuilding a corrupt system database in a device built in 12.5, you do not need to specify the logical page size. The rewrite process retrieves the logical page size from the config block.

If the config block is corrupt, use the `dataserver -z page size` argument to specify the logical page size of the master device.

Obtain the page size of the master device you are rebuilding by checking a previous error log from its server, then specify that page size using the `-z` option. If the value you specify on the `-z` option differs from what the server finds in the config block, Adaptive Server reports a warning and does not perform the rewrite process. You can override this error using the `-f` option to force the rebuild.

Importance of specifying the correct *master* device and logical page sizes

During the rewrite master process, Adaptive Server examines each allocation page in the existing, and possibly corrupt, master device, to identify allocation units that belong to the master database. Adaptive Server uses the logical page size specified to create each allocation unit.

Therefore, if you provide an incorrect logical page size, data is read incorrectly while identifying allocation units. This results in a new master device that is rewritten with the wrong logical page size, potentially destroying previously existing model, tempdb, sybssystemdb, and other user databases.

If the specified master device size is smaller than the size of the initial master device, the search for allocation units that rightfully belong to the master database does not find space that should belong to the initial master device. If the specified master device size is too large, other databases that reside past the end of the initial master device are incorrectly scanned, which can cause corruption.

Validating user-specified size values against the config block

If you use the `-b` or `-z` option, Adaptive Server checks the sizes indicated in the options against the sizes indicated in the config block.

Warning! If the master device size and logical page size values differ from the values used when the installation was created, the rewrite master process corrupts other databases on your master device.

The following examples show the error checking that is done when an incorrect size value is specified:

Example 1

The master device size was 140MB when initially built, but the `-b` option specifies 200MB when rewriting the master database:

```
$SYBASE/$SYBASE_ASE/bin/dataserver -d/sybase/devices/master.dat
-c$SYBASE/ASE_125.cfg -e/sybase/devices/125_error.log -w master -b 200M
00:00000:00000:2001/06/11 09:21:28.79 kernel Use license file
/sybase/SYSAM-1_0/licenses/license.dat.
00:00000:00000:2001/06/11 09:21:28.79 kernel Checked out license ASE_SERVER
00:00000:00000:2001/06/11 09:42:15.02 kernel Using config area from primary
master device.00:00000:00000:2001/06/11 09:42:15.12 server This
installation's configured device size (71680 virtual pages) does not match
that specified on the command line (102400 virtual pages). To use the
configured size, omit the command line size; to use the command line size,
specify 'forcebuild'.
```

Example 2

The master device size had a logical page size of 2K when initially built, but the `-z` option specifies a 4K logical page when rewriting the master database:

```
$SYBASE/$SYBASE_ASE/bin/dataserver -d/sybase/devices/master.dat
-c$SYBASE/ASE_125.cfg -e/sybase/devices/125_error.log -w master -z 4k
00:00000:00000:2001/06/11 09:21:28.79 kernel Use license file
/sybase/SYSAM-1_0/licenses/license.dat.
00:00000:00000:2001/06/11 09:21:28.79 kernel Checked out license ASE_SERVER
00:00000:00000:2001/06/11 09:21:28.85 kernel Using config area from primary
master device.
```

```
00:00000:00000:2001/06/11 09:21:28.97 server The configured server page
size (2048 bytes) does not match that specified on the command line (4096
bytes). To use the configured size, omit the command line size; to use the
command line size, specify 'forcebuild'.
```

Note Due to the complexity and risk of the following workaround, Sybase strongly recommends that you dump the master database and all databases that use space on the master device after you complete an upgrade to 12.5. You should dump these databases regularly.

Building the *master* database

[Bug #209913] During this procedure, the following error messages may occur; you can ignore them: 3479, 5859, and 11245.

❖ To rebuild the *master* database

1 To rebuild the master database, use the 12.5 `dataserver` binary with the `-b` and `-w` flags. You can also use the `-z` flag to specify the logical page size. For example:

- To rebuild master database on a 200MB master device upgraded from a pre-12.5 server to 12.5, enter:

```
dataserver -d master.dat -w master -b 200M
```

The logical page size for upgraded servers is 2K, so the `-z` flag is not needed here.

- To rebuild a master device initially built using a 12.5 Adaptive Server of 250MB, with a logical page size of 8K, enter:

```
dataserver -d master.dat -w master
```

Assuming the config block is not corrupt, Adaptive Server retrieves the 250MB master device size and the 8K logical page size from the config block. The `-b` and `-z` flags are not necessary in this case.

- If the config block is not corrupt, and the user specifies individual sizes, enter:

```
dataserver -d master.dat -w master -b 250M -z 8K
```

Adaptive Server reads the required values from the config block, and validates them against the user-specified values. If these two values match, the master database is built at the specified location.

- If the config block is corrupt, enter:

```
dataserver -d master.dat -w master -b 250M -z 8K
-forcebuild
```

The user specifies the required sizes, and forces a rebuild of the master database, overriding the values in the config block with user-specified values.

The rewriting of the master database goes through an internal upgrade process to rebuild the corrupt master database. Adaptive Server shuts down when it has finished, and may produce an output that looks similar to the following:

```
00:00000:00001:2001/06/11 09:20:47.66 kernel libomni1 - Component
Integration Services: using 'Sybase Client-Library/12.5/P/SPARC/Solaris
2.8/1/DEBUG/Fri Mar 23 20:31:35 2001'
00:00000:00001:2001/06/11 09:20:47.70 server The logical pagesize of the
server is 2 Kb.
00:00000:00001:2001/06/11 09:20:49.00 server Database 'master' appears
to be at an older revision than the present installation; SQL Server will
assess it, and upgrade it as required.
00:00000:00001:2001/06/11 09:20:49.07 server Database 'master':
beginning upgrade step [ID 1]: Initialize disk and create empty
allocation units on master device.
00:00000:00001:2001/06/11 09:20:50.24 server Database 'master':
beginning upgrade step [ID 2]: Bootstrap basic system catalogs in
database.
00:00000:00001:2001/06/11 09:20:54.22 server Database 'master':
beginning upgrade step [ID 3]: creating index (table systypes, index
ncsystypes)
00:00000:00001:2001/06/11 09:20:55.43 server Database 'master':
beginning upgrade step [ID 4]: creating index (table sysobjects, index
ncsysobjects)
[...]
00:00000:00001:2001/06/11 09:20:57.38 server Database 'master':
beginning upgrade step [ID 81]: Load initial set of attribute-
definition rows into SYSATTRIBUTES.
00:00000:00001:2001/06/11 09:21:05.50 server Warning: Using only the
first 30 characters for password
00:00000:00001:2001/06/11 09:21:06.86 server No. of rows in
Master_sysmessages=4722
00:00000:00001:2001/06/11 09:21:06.86 server Total size of
Master_sysmessages=547085 bytes (0.52 M)
00:00000:00001:2001/06/11 09:21:11.59 server Database 'master':
beginning upgrade step [ID 251]: creating table (table syslogshold)
[...]
00:00000:00001:2001/06/11 09:21:11.97 server Database 'master':
beginning upgrade step [ID 602]: creating table (table sysusermessages)
[...]
00:00000:00001:2001/06/11 09:21:15.23 server Database 'master':
beginning upgrade step [ID 1345]: executing SQL statement (update
```



```

sysattributes set int_value=)
00:00000:00001:2001/06/11 09:21:15.28 server Database 'master':
beginning upgrade step [ID 1400]: noting the present database upgrade
level
00:00000:00001:2001/06/11 09:21:15.62 kernel ueshutdown: exiting

```

Note Adaptive Server uses a variant of the upgrade process to rebuild the master database. Do not be concerned that the rewrite master process upgrades the entire server again.

- 2 Edit a copy of the *RUN_servername* file in your 12.5 Adaptive Server *install* directory to add the *-m* flag and the *-T3608* trace flag. For example:

```

/ $SYBASE/$SYBASE-ASE/bin/dataserver -d/$SYBASE/data/master.dat \
-sTEST -e/$SYBASE/$SYBASE-ASE/install/TEST_errorlog \
-i/$SYBASE/interfaces -M/$SYBASE/$SYBASE-ASE -m -T3608

```

Use this file to start the server in single-user mode.

- 3 For all devices added with disk init, you must run disk reinit. Check the output from sysdevices, if available, for a list of devices. You must give the correct logical and physical names and sizes for the disk reinit command. Obtain physical device names from an error log of a prior successful start of the server. Obtain logical names and sizes from records you have kept regarding your installation.

For more information on disk reinit, see the *Reference Manual* or the *Error Messages and Troubleshooting Guide*.

- 4 Run disk refit. When disk refit completes, Adaptive Server automatically shuts down.
- 5 Restart the server using the *-m* flag and the *-T3608* trace flag.
- 6 Run:

```
dbcc checkalloc("master", fix)
```

You can ignore messages similar to the following:

```

EXTID:1536 (Alloc page: 1536) is initialized. Extent follows:
NEXT=0 PREV=0 OBJID=0 ALLOC=0x0 DEALL=0x0 INDID=1 STATUS=0x0
EXTID:1544 (Alloc page: 1536) is initialized. Extent follows:
NEXT=0 PREV=0 OBJID=0 ALLOC=0x0 DEALL=0x0 INDID==0 STATUS=0x0
EXTID:1552(Alloc page: 1536) is initialized. Extent follows:
...

```

- 7 Shut down the server, and restart the server in the usual way; that is, without the -m flag and the trace flag.
- 8 Run dbcc checkalloc on all databases. No error messages are expected at this time.
- 9 Run the following commands from your Adaptive Server 12.5 directory:

```
isql -Usa -Ppassword -i scripts/installmaster
isql -Usa -Ppassword -i scripts/instmsgs.ebf
```

If you are using jConnect, run:

```
isql -Usa -Ppassword -i scripts/installjconnect
```

If you are using two-phase commit or distributed transactions, run:

```
isql -Usa -Ppassword -i scripts/installcommit
```

- 10 Dump all databases.

All server-supplied system messages are reinstalled in U.S. English as part of this process. Localize these messages manually after this process.

The rebuild master process re-creates sysusermessages. Reinstall any messages that you installed in the master database.

Rebuilding the *model* database

If both your master and model databases are corrupt, you must first rebuild the master database as described in “Building the master database” on page 15. Rebuilding a model database when the master database is intact uses information from system catalogs in the master database to identify pieces of the existing model database in your server.

❖ To rebuild the *model* database

- 1 Run the 12.5 version of dataserver using the -w parameter:

```
dataserver -d master.dat -w model
```

The -b and -z parameters are ignored.

The new model database then goes through an internal upgrade process that rebuilds the corrupt model database. When Adaptive Server completes this process, it displays the following and shuts down:

```
00:00000:00001:2001/06/13 11:54:58.00 server The transaction log in the
database 'master' will use I/O size of 2 Kb.
00:00000:00001:2001/06/13 11:54:58.12 server Database 'model' appears
to be at an older revision than the present installation; SQL Server will
assess it, and upgrade it as required.
00:00000:00001:2001/06/13 11:54:58.14 server Database 'model':
beginning upgrade step [ID 1]: Initialize disk and create empty
```

```

allocation units on master device.
00:00000:00001:2001/06/13 11:54:59.13 server Database 'model':
beginning upgrade step [ID      2]: Bootstrap basic system catalogs in
database.
00:00000:00001:2001/06/13 11:54:59.63 server Database 'model':
beginning upgrade step [ID      3]: creating index (table systypes, index
ncsystypes)
[...]
00:00000:00001:2001/06/13 11:55:09.40 server Database 'model':
beginning upgrade step [ID 1400]: noting the present database upgrade
level
00:00000:00001:2001/06/13 11:55:09.47 kernel ueshutdown: exiting

```

- 2 Edit a copy of the *RUN_servername* file in your 12.5 Adaptive Server *install* directory to add the *-m* flag. For example:

```

/$SYBASE/$SYBASE-ASE/bin/dataserver -d/data/master.dat \
-sTEST -e/$SYBASE/$SYBASE-ASE/install/TEST_errorlog \
-i/$SYBASE/interfaces -M/$SYBASE/$SYBASE-ASE -m

```

Use this file to start the server in single-user mode.

- 3 Run:

```
dbcc checkalloc("model", fix)
```

You can ignore messages similar to the following:

```

EXTID:1536 (Alloc page: 1536) is initialized. Extent follows:
NEXT=0 PREV=0 OBJID=0 ALLOC=0x0 DEALL=0x0 INDID=1 STATUS=0x0
EXTID:1544 (Alloc page: 1536) is initialized. Extent follows:
NEXT=0 PREV=0 OBJID=0 ALLOC=0x0 DEALL=0x0 INDID==0 STATUS=0x0
EXTID:1552(Alloc page: 1536) is initialized. Extent follows:
...

```

- 4 Run:

```
dbcc checkalloc("model")
```

There should be no errors at this time.

- 5 Run the following command from your 12.5 Adaptive Server directory:

```
isql -Usa -Ppassword < scripts/installmodel
```

- 6 You can now shut down the server and restart it in your usual way; that is, without the *-m* flag.

6.2 Highlighted known RPC-related problems

This section discusses known problems with remote procedure calls (RPCs) and Adaptive Server version 12.5.

6.2.1 RPC compatibility with older versions of Adaptive Server

[Bug #239685] Adaptive Server version 12.5 fails when executing a server-to-server remote procedure call (RPC) against an earlier version of Adaptive Server if the remote server predates a fix made under CR #210796.

Workaround: To execute an RPC successfully from a version 12.5 Adaptive Server, your remote Adaptive Server must be one of the following:

- 12.5.x
- 11.9.2.2 or later within this release
- 11.5.1.1 or later within this release
- 11.0.3.4 or later within this release

6.2.2 Channel setup failure message when issuing RPCs

[Bug #239115] When you issue a remote procedure call (RPC) from a server that is aliased in `sys.servers` to itself, you may encounter an unexpected “channel setup failure” message.

Workaround: Restart your server using trace flag 8015:

```
dataserver -dd_master -T8015
```

6.3 Highlighted known SSL problems

This section discusses known problems related to Secure Sockets Layer (SSL) and Adaptive Server version 12.5.

6.3.1 Network memory requirements for using SSL for incoming connections

[Bug #273923] SSL-enabled client connections initiated from Adaptive Server, such as CIS/OMNI, DTM, RepAgent, CIS RPC handling, and so on, use up memory resources intended for incoming Adaptive Server SSL connections. When the memory reaches capacity, Adaptive Server returns this message to the error log:

```
ssl_alloc: Cannot allocate using ubfalloc(rnetmempool,
```

nnnn)

This affects both incoming connections to Adaptive Server and outgoing client connections initiated from Adaptive Server.

Workaround: Increase additional network memory to compensate for the memory requirements. You should add a minimum of 256K plus 40K for each outgoing connection in max cis remote connections.

If cis rpc handling is enabled, then the amount you need to add may go up to 600K to 900K, plus 40K for each number of user connections to handle peak memory requirements.

In addition, you must add an amount equal to or greater than the additional network memory to max memory to account for the additional memory requirements.

6.3.2 Lack of memory causes *ssl_handshake* failures

[Bug #225387] If a large number of connections are made to Adaptive Server, a memory leak can occur, causing *ssl_handshake* to fail and show an error message. The error log displays:

```
00:00000:00009:2001/05/17 16:49:41.63 kernel ssl_alloc:
Cannot allocate using ubfalloc(rnetmempool,####)
```

Workaround: Configure your server to give more memory to the network by using the additional network memory configuration parameter, found in Chapter 5, “Setting Configuration Parameters” of the *System Administration Guide*.

The workaround also requires that you occasionally shut down the server and restart it.

6.3.3 Harmless error messages when starting Linux with SSL enabled

[Bug #230232] When starting Adaptive Server on the Linux platform, you may receive intermittent errors such as:

```
ls: write error
df: write error: Broken pipe
```

Workaround: None. You can ignore the messages. The data server is running. The error messages are from third-party libraries.

6.3.4 SSL listener port hangs

[Bug #238820] If loading the certificate file fails due to an incorrect password while starting the server from a command line with the -y option for attended SSL server start-up, failover from SSL port to cleartext port does not work. The connection attempt hangs on the SSL listener socket.

Workaround: Terminate the connection attempt and restart the server with the correct private-key password after fixing the problem.

6.4 Highlighted known Sybase Central problems

The following sections describe known problems and workarounds for Sybase Central.

6.4.1 Unnecessary characters appear when executing Sybase Central

[Bug #240591] When you execute Sybase Central using scjview on Linux, you see messages that include unnecessary `\n` characters:

```
\nStarting Sybase Central Java Edition (displaying to  
rosella) ... \n
```

This occurs on Linux because of the behavior of the echo command.

Workaround: Ignore the `\n` characters that appear in the messages.

6.4.2 Error when viewing table properties in Sybase Central

[Bug #209782] When you select a user table and click Properties, then click the Miscellaneous tab, an error message appears. If you click OK, the error message dialog closes; however, if you then click the Properties button on the top of the Miscellaneous sheet, Sybase Central causes an application error and is shut down.

A similar error message also appears when you connect to a data server, select the *execution classes* folder, view the properties for one of the execution classes shown, then select the Bindings tab. An error dialog appears with a message similar to the one in the previous scenario; however, in this case Sybase Central does not cause an application error.

Workaround: To query the server for the table information, use isql:

```
use db_name  
go  
  
SELECT rowcnt( doampg ),
```

```
reserved_pgs( id, doampg ) + reserved_pgs( id, ioampg ),
data_pgs( id, doampg ), data_pgs( id, ioampg )
FROM sysindexes WHERE id = id
```

6.5 Highlighted known *alter table* problems

The following sections describe known problems and workarounds for the *alter table* command.

6.5.1 Sporadic error message from *alter table* on DOL tables

[Bug #240950] The following only occurs on servers that use a 4K page size.

If you use *alter table* on DOL tables with a clustered index to add non-NULL columns, or modify or drop existing columns, Adaptive Server may issue an error message similar to the following:

```
Msg 1530, Level 16, State 1: Line 1: Create index with
sorted_data was aborted because of row out of order.
Primary key of first out of order row is '27, "default
string"
```

This error occurs after *alter table* performs the data copy and is in the process of rebuilding the clustered index. This is a generic problem, and is highly data dependent, and only occurs on DOL tables with a clustered index.

If this error occurs, the entire operation is rolled back, so tables are not corrupted, and the original schema of the table being altered is reinstated.

Workaround: Drop the clustered index, alter the table to implement the schema change, and then rebuild the clustered index manually after the *alter table* operation has committed.

6.5.2 *alter table* and *sysstatistics* flaw

[Bug # 239627] When operating within big-endian architectures such as Digital UNIX, Linux, and NT, if the ID of the last column in the colidarray of a row in the *sysstatistics* table is a multiple of 256, then *alter table drop column* corrupts data in the colidarray column. This happens only if the columns dropped by the *alter table drop column* command do not appear in the colidarray, and if the ID of the last column minus the number of columns dropped is no longer a multiple of 256.

Workaround: If the above requirements are met and the colidarray column for the sysstatistics table is corrupt; delete sysstatistics, and then regenerate it. The correct values for the colidarray column will then be represented.

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

[Bug #199655] If a trigger contains an *if update()* clause, data modifications that should fire the trigger that are done after executing *alter table add*, *alter table drop*, *alter table lock*, or *alter table modify* may cause errors in the 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 will execute correctly.

6.6 Highlighted known *select* problems

The following sections describe known problems and workarounds for the *select* command.

6.6.1 *select* statements that include *order by* clauses

[Bug # 229518] Adaptive Server issues an error message if you perform a *select* from a table that:

- Includes a column, or columns, that are very close to the maximum row size for the server's logical page size (this problem does not occur if the sum of all the row sizes approach the maximum row size), and:
- Includes an *order by* clause, and:
- Uses the following format:

```
select substring(column_1, position, string_size)
from table_name
order by column_name
```

For example, if you create the following table:

```
create table documents(doc_id      int,
                      doc_text char(16215)
                      doc_author varchar(50))
insert documents values (1, "Is it OK?", "Jim O'Neal")
insert documents values (2, "No", "Judy Dermott")
```


And perform this select on it:

```
select substring(doc_text, 50, 20) from documents order
by doc_text
```

Adaptive Server issues this error message:

```
Attempt to update or insert row failed because resultant
row of size 16301 bytes is larger than the maximum size
(16298 bytes) allowed for this table
```

For this query, order by uses the column doc_text and not the substring() column. By default, substring() is defined to be nullable, which means that the result of the substring() is a nullable column. The order by requires a sort, and creation of a worktable, which looks like this:

```
select doc_text, substring(doc_text, 50, 20) as col2
```

The error occurs because the worktable is created as an all-pages-locked (APL) table. For APL row formats, Adaptive Server uses a row-length of 16295 bytes for a fixed-length column (in this case, column doc_text), followed by the varchar column doc_author. For the offset and adjust tables, the total row length for column c1 is 16366 bytes, which is larger than an APL row allows.

Workaround: Issue a select statement organized like the following, which orders the data by the substring() expression, and not column doc_text:

```
select substring(doc_text, 50, 20) from documents
order by 1
```

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

[Bug #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.

6.7 Turning off the O_SYNC flag

[Bug #241217] By default, Adaptive Server version 12.5 for Linux opens database devices with the O_SYNC flag to ensure recoverability in the event of power failure, operating system crash, or Adaptive Server crash. This O_SYNC flag impacts performance significantly.

Workaround: Turn off the O_SYNC flag by restarting the server using trace flag 1625:

```
dataserver -dd_master -T1625
```

6.8 unichar does not work across platforms

[Bug #241487] jConnect 4.5/5.5 does not properly convert unichar and univarchar data across platforms. When selecting data from a server on a platform that uses a different byte order than the client, such as a Windows client selecting data from a Sun server, jConnect does not properly convert the String data.

Workaround: Go to the the Sybase download area at <http://www.sybase.com/downloads> to download the latest EBF of jConnect 4.5/5.5. This bug has been fixed in the EBF.

6.9 sortkey() truncates result sets from wide columns

[Bug # 239293] sortkey() displays a maximum of 255 bytes of data and truncates result sets that are longer than this, which may lead to inaccurate comparison for wider columns.

6.10 Restrictions on *login trigger*

[Bug #234977] There are certain commands that are not allowed in a login trigger.

The login trigger is the last step for a login and is executed in the background, therefore, no result set is displayed. If any errors occur and the login fails, it is reported only to the error log.

The login failure displays only after you attempt to execute a command.

Workaround: Check the restrictions defined on dynamic SQL commands to verify the command will work with the login trigger.

6.11 Engine not exiting after *ueshutdown()*

[Bug #240673] If a timeslice error occurs in a multiple-engine Adaptive Server running on Linux, the engine on which the timeslice error occurred may not be terminated when you issue a ueshutdown() command.

Workaround: Terminate engine processes with the operating system command kill. Clear shared segments with the operating system command ipcrm.

6.12 Dynamic and host variables not allowed in DDL statements

[Bug # 234217] Dynamic and host variables in create view statements were allowed in earlier versions, but worked incorrectly when:

- create view statements were replicated, or
- Views containing dynamic and host variables were upgraded from text in the syscomments table.

To fix this problem, dynamic and host variables are no longer allowed in create view statements.

Workaround: To allow dynamic and host variables with this incorrect behavior in create view statements, start Adaptive Server with trace flag 108:

```
dataserver -dd_master -T108
```

7. Product compatibilities

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

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

- Sybase Character Sets 3.0
- Sybase Central viewer 3.2
- DirectConnect™ Anywhere 11.7
- DirectConnect for Informix 11.7
- DirectConnect for MVS 11.7
- DirectConnect for Oracle 12.0
- OpenSwitch 12.0
- jConnect for JDBC 4.5
- jConnect for JDBC 5.5
- Open Client/Open Server™ 12.5
- 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® 11.5.1, 12.0
- Security Guardian™ 11.1.1
- SQL Remote 6.0.2
- Physical Architect 8.0
- XA-Library for CICS/ENCINA 11.1.1
- XA-Server for Tuxedo 11.1.1

Note Backup Server 12.5 has been tested compatible with Backup Server 11.9.2.1 and 12.0. 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.

Replication Server compatibility

Replication Server version 12.1 does not currently support the following Adaptive Server version 12.5 features:

- Does not replicate extended data such as wide rows or wide columns, without truncating it.
- Does not replicate Unicode data.
- Does not use SSL when connecting to Adaptive Server, although Replication Server can still replicate data from and to an Adaptive Server that uses SSL.
- Does not use an LDAP server to retrieve server information, but instead relies on traditional interfaces files for this information. Replication Server can, however, still replicate data from and to an Adaptive Server that uses an LDAP server for this purpose.

DirectConnect and OpenSwitch compatibility

Direct Connect and OpenSwitch do not currently support the following Adaptive Server version 12.5 features:

- Extended limits, such as wide rows and wide columns
- Unicode
- SSL
- LDAP

8. Documentation updates and clarifications

This section describes changes to the following Adaptive Server documents:

- “Full Text Search Specialty Data Store User's Guide” on page 29
- “Performance and Tuning Guide” on page 32
- “Reference Manual Volume 1: Building Blocks” on page 39
- “Reference Manual Volume 2: Commands” on page 40
- “Reference Manual Volume 3: Procedures” on page 45
- “Transact-SQL User’s Guide” on page 69
- “Utility Guide” on page 70

8.1 Full Text Search Specialty Data Store User’s Guide

Version 12.5 of the Full-Text Search Specialty Data Store is the Enhanced Version.

Standard Full-Text Search (SFTS) is being discontinued. If you use SFTS, you should migrate to Enhanced Full-Text Search (EFTS), an option with Adaptive Server version 12.5. As an SFTS user, you can obtain a copy of EFTS through your Sybase sales representative or customer service.

Enhanced Full-Text Search (EFTS) was a standalone product that can now be purchased as an Adaptive Server option beginning with version 12.5. If you are an EFTS user on an existing support plan, you can also obtain a copy of the Adaptive Server option version of EFTS.

You do not need to pay EFTS support fees until your current Adaptive Server version 12.0 support plan expires.

8.1.1 Hardware and software requirements

To install and use the Enhanced Full-Text Search, you must install the following:

- Adaptive Server Enterprise version 12.5, which can be installed at the same time as Full-Text Search
- A valid license key (ASE-EFTS)

Enhanced Full-Text Search requires the same operating system and patches that are required by Adaptive Server Enterprise version 12.5.

Hardware requirements

- Minimum of 64MB of RAM
- 100MB of disk space for the Full-Text Search engine and Verity collections
- Twice the amount of space required by the text being indexed (for example, if you have 100MB of text, you need 200MB of space)

8.1.2 Upgrading from Adaptive Server version 11.9.2

Author comment: LINUX

Upgrade Full-Text Search from version 11.9.2:

- Shut down all servers and install Adaptive Server version 12.5 and Full-Text Search 12.5.
- Enter a valid license for Enhanced Full-Text Search 12.5 (ASE_EFTS).
- Upgrade the Adaptive Serve following instructions in the *Installation Guide* for your platform.
- Copy the Full-Text Search configuration file (*textserver_name.cfg*) from the old location to the new *\$SYBASE*, if applicable. You can skip this step if you have installed Enhanced Full-Text Search over the old location.
- Set the path to the old collections directory using the configuration file parameter *collDir*.
- Re-run the following installation scripts as outlined in the configuration section of the *Full-Text Search Specialty Data Store User's Guide*:
 - *installtextserver*
 - *installmessages*
 - *installevent*
- Start the Adaptive Server and run *sp_text_online* as described in the *Full-Text Search Specialty Data Store User's Guide*.

8.1.3 New functionality in this version

The following new features have been added to Full-Text Search since version 12.0:

- Supported data types – Adaptive Server introduces two new datatypes: `unichar` and `univarchar`. Enhanced Full-Text Search supports indexing of columns of these types.
- Licensing – there is a new configuration option in Adaptive Server 12.5, which enables Full-Text Search capability. Additionally, a license is required to enable this capability. Enhanced Full-Text Search does not accept connections from an Adaptive Server that does not have this feature enabled. To enable Full-Text Search capability, enter:

```
sp_configure 'enable full-text search', 1
```

- File system access – Adaptive Server 12.5 now allows mapping of proxy tables to files and directories. Enhanced Full-Text Search 12.5 supports indexing of columns of these tables.
- `batch_blocksize` – a new configuration parameter enables the Full-Text Search engine to retrieve data in the specified chunk sizes of up to 64K.
- `sp_check_text_index` – a new stored procedure you can use to report and fix consistency problems in the Full-Text Search index and source tables.
- IDENTITY column size limit increased – in earlier Full-Text Search versions, the IDENTITY column of the source table was limited to 10 digits. This restriction has been removed to allow IDENTITY column precision up to `numeric(38,0)`.
- Named pipes – this version supports named pipes, where allowed by Adaptive Server Enterprise.
- The highlighting feature is no longer supported.

8.1.4 Known problems

The following sections describe known problems with this version of Adaptive Server as it relates to Full-Text Search.

VDK language settings

When setting the VDK language in the configuration file to French, use *frenchx* as the configuration option.

Increase size of *model* database before installing scripts

Before installing the EFTS-12_5 scripts (*installtextserver*, *installevnts*, *installmessages*) to your Adaptive Server Enterprise, you may need to increase the size of the tempdb and the model databases from the default size to 3MB each when installing on a 64-bit platform.

Sample script

There is a typographical error in the sample script *EFTS-12_5/sample/scripts/sample_text_queries.sql* file. The first character in the script is “7.” It should be changed to a forward slash (/).

Use *interfaces* file

Enhanced Full-Text Search does not work with LDAP for this version; it requires an *interfaces* file.

Copy *startserver* to installation directory (Linux only)

After installing and configuring Enhanced Full-Text Search, copy the *startserver* file from the *\$\$SYBASE/\$SYBASE_ASE/install* directory to the *\$\$SYBASE/\$SYBASE_FTS/install* directory to use when restarting the text server.

8.1.5 Product compatibility

Version 12.5 requires Open Client 11.x or later. Earlier versions are not supported.

8.2 Performance and Tuning Guide

This section discusses updates and clarifications for the *Performance and Tuning Guide*.

8.2.1 Performance issues with using larger logical pages

The following sections describe performance issues when using Adaptive Server with larger logical page sizes.

Large logical pages and buffer pools

The default logical page size for Adaptive Server is 2K. If you are using larger logical pages, the memory for data caches and buffer pools is allocated in terms of logical pages. For example, if you are using a server with a 2K page size and the following cache configuration:

	Memory allocated	Number of buffers
Default data cache	20MB	10240
2K buffer pool	10MB	5120
8K buffer pool	5MB	640
16K buffer pool	5MB	320

The number of buffers in the 8K pool (640) is in terms of the number of 8K masses. Each mass in the 8K pool consists of four 2K buffers. The number of buffers in the 16K pool (320) gives the number of 16K masses.

For a server with page size of 8K, the basic buffer size is 8K. The larger-sized buffer pools are 8, 16, 32, or 16K.

The cache configuration above results in the following numbers of buffers available in each buffer pool.

	Memory allocated	Number of buffers
Default data cache	20MB	2560
8K buffer pool	10MB	1280
32K buffer pool	5MB	160
64K buffer pool	5MB	80

For this example, the 5MB memory allocated to the 32K and 64K buffer pool, the number of masses available are too few to set up the server's buffer pool.

Note When you are sizing the buffer pool, you should size according to the number of logical pages in a single buffer, rather than in terms of Kilobytes.

You must size the individual buffer pools in various caches according to the server's logical page size. If you do not size the caches and buffer pools properly, the buffer pool may not have enough buffers, causing excessive buffer washing, which negatively affects the performance of your queries.

Large I/O, buffer prefetch

Because of Adaptive Server version 12.5 larger logical pages, the term *Large I/O* implies the use of a mass of 8 logical pages, and is the amount of I/O to read an extent's worth of data. Depending on your server's logical page size, large I/O is 16, 32, 64, or 128K.

If your configuration file for a server using a 2K page size Adaptive Server specifies a 16K buffer pool, and you want to use large I/O on another server running with 16K logical page size, you must specify a buffer pool of size 128K in the latter server.

Specifying a 16K buffer pool in the 16K logical page size server creates a buffer pool of the basic page size, which will not support large I/Os.

Consider the following when configuring for large I/Os:

- The set prefetch on option allows Adaptive Server to prefetch data using large I/O only if large I/O buffer pools are available.
- If you are using larger logical pages, the prefetch specifier for select, delete, and update statements should specify the larger prefetch size based on the server's logical page size. Prefetch sizes that are less than the logical page size are ignored, and all I/O is performed using the buffers the size of the logical page.

Optimizing for concurrency on small DOL tables

For data-only-locked (DOL) tables, the table-level property `concurrency_opt_threshold` directs the optimizer to favor index scans on small tables. The default value for this is 15 logical pages. However, the data for some medium-sized tables may require more than 15 logical pages, although the same data would fit into far fewer pages on a server that uses larger logical pages.

This situation may cause the optimizer to favor an index scan on small tables in servers that use larger logical pages, whereas the same query on a server that uses 2K logical pages can result in the optimizer selecting a more efficient access method.

This potential change in behavior is data-dependent and must be analyzed on an individual query basis. You can use `sp_chgattribute` to lower the `concurrency_opt_threshold` value for affected tables to achieve the optimizer characteristics that are seen on servers that use 2K logical pages.

General guidelines about optimizer costing

All optimizer costing for physical and logical I/O is done in terms of page sizes. The cost estimates reported under various trace flags (such as 302, 310) are in terms of logical pages. Data clustering ratios like data page cluster ratio and Index page cluster ratio are all in terms of logical pages. It is important that you think in terms of logical pages as well as 2K pages when you configure the server.

For larger pages, the index height tends to be smaller than the index height for smaller logical pages. You may find differences in cost estimates, and choices of indexes on larger pages. Refer to the *Performance and Tuning Guide* for techniques to analyze these differences in query plan choices. In some instances, references to 2K logical pages in this document should be treated in terms of the current logical page size for your server.

8.2.2 Bulk copy utility

This updates information on the bulk copy (bcp) utility in Chapter 5, “Controlling Physical Data Placement” in the *Performance and Tuning Guide*:

Bulk copy utility

When you are using the bulk copy (bcp) to copy a very large number of rows, do *not* use -F for the first row and still use multiple input files. The -F option is for single processing and ad hoc copying, not the heavy-duty, multiprocess copying found in a production environment.

8.2.3 Union and view operators

Chapter 21, “Abstract Plans Language Reference” describes the use of the operators union and view. In versions earlier than Adaptive Server 12.5, a union defined the union of two or more derived tables. The view operator identifies a view that contains the base table to be scanned.

Abstract Plans now fully support unions in views. This example shows how this can be accomplished:

```
1> sp_helptext vua
2> go
```

```
-----
# Lines of Text
-----
```

```
1
(1 row affected)
text
-----
```

```

create view vua(v1, v2)
as
select r1, sum(s1) from r, s where r2 = s2 group by
    r1
union all
select t.t1, t.t2 from t

```

```

-----
(1 row affected)
(return status = 0)
-----

```

```

1> sp_helptext vud
2> go

```

```

-----
# Lines of Text
-----

```

```

1
(1 row affected)
text

```

```

-----
create view vud(v1, v2)
as
select r1, sum(s1) from r, s where r2 = s2 group by
    r1
union
select t.t1, t.t2 from t

```

```

-----
(1 row affected)
(return status = 0)
-----

```

```

1> select t1, max(t2) from vua, vud, t group by t1
2> go

```

```

AP Input Association :
select t1, max(t2) from vua, vud, t group by t1
Input Abstract Plan (id 544001938) :
( plan
  ( store Worktab3
    ( union
      ( plan
        ( store Worktab5
          ( nl_g_join
            ( t_scan s )
            ( t_scan r )
          )
        )
      ( t_scan ( work_t Worktab5 ) )
    )
  )
)

```

```

        )
        ( t_scan t )
    )
)
( store Worktab2
  ( union
    ( plan
      ( store Worktab6
        ( nl_g_join
          ( t_scan s )
          ( t_scan r )
        )
      )
      ( t_scan ( work_t Worktab6 ) )
    )
    ( t_scan t )
  )
)
( plan
  ( store Worktab1
    ( nl_g_join
      ( t_scan ( work_t Worktab3 )
      ( t_scan ( work_t Worktab2 ) )
      ( t_scan t )
    )
  )
  ( t_scan ( work_t Worktab1 ) )
)
)

```

Parsing the input Abstract Plan

Applying the Abstract Plan

Capturing the Abstract Plan.

AP Output Association :

```
select t1, max(t2) from vua, vud, t group by t1
```

Generating the output Abstract Plan.

Output Abstract Plan :

```

( plan
  ( store Worktab3
    ( union
      ( plan
        ( store Worktab5
          ( nl_g_join
            ( t_scan s )
            ( t_scan r )
          )
        )
      )
    )
  )
)

```

```

        ( t_scan ( work_t Worktab5 ) )
      )
    ( t_scan t )
  )
)
( store Worktab2
  ( union
    (( plan
      ( store Worktab6
        ( nl_g_join
          ( t_scan s )
          ( t_scan r )
        )
      )
      ( t_scan ( work_t Worktab6 ) )
    )
    (( t_scan t )
  )
)
( plan
  ( store Worktab1
    ( nl_g_join
      ( t_scan ( work_t Worktab3 ) )
      ( t_scan ( work_t Worktab2 ) )
      ( t_scan t )
    )
  )
  ( t_scan ( work_t Worktab1 ) )
)
)
)

```

8.2.4 Prefetch size and replacement strategy

This section includes changes to the *Performance and Tuning Guide* on the information for prefetch and replacement strategy.

In earlier versions, when a prefetch was done you could only specify one size, for example, 2K, 4K, and it had to be the same size for the table scan and possible index scan. This also applied to the replacement strategy.

In Adaptive Server 12.5, you can name more than one size for the table, the index, and the lru/mru. This example shows two prefetch sizes and two replacement strategies for the same table:

```
(i_scan i1 t1) (prop t1 (prefetch 8 16)(lru mru))
```

This allows you an 8K prefetch for table t1 and a 16K prefetch for any index scan for t1 (i1 in the example). It also allows you an lru replacement strategy for t1 and an mru for any index scan as above.

These changes also apply to the force index syntax in SQL, so the following query has the same effect:

```
select * from t1(i1 (prefetch 8 16)(lru mru))
```

8.3 Quick Reference Guide

This section discusses updates and clarifications for the *Quick Reference Guide*.

8.3.1 Range of values for *smallint* and *int*

The range of values for the *smallint* datatype is -2^{15} (-32,768) to $2^{15} - 1$ (32,767).

The range of values for the *int* datatype is -2^{31} (-2,147,483,648) to $2^{31} - 1$ (2,147,483,647).

8.4 Reference Manual Volume 1: Building Blocks

This section discusses updates and clarifications for *Reference Manual Volume 1: Building Blocks*.

8.4.1 Range of values for *smallint*

The range of values for the *smallint* datatype is -2^{15} (-32,768) to $2^{15} - 1$ (32,767).

8.4.2 Size of expressions and global variables

Expressions and global variables can be up to 16K in length.

8.4.3 *data_pgs* command includes *dbid* parameter

The *data_pgs* command includes an optional parameter, *dbid*. You must re-create any stored procedures that use *data_pgs*.

The new syntax is:

```
data_pgs([dbid], object_id, {data_oam_pg_id | index_oam_pg_id})
```

Where:

- *dbid* – is the *dbid* of the database that contains the data pages.
- The definitions for *object_id*, *data_oam_pg_id*, and *index_oam_pg_id* are in the *Reference Manual*.

For example:

```
select o.name,
Pages = data_pgs(4,i.id, i.doampg)
from pubs2..sysindexes i, pubs2.sysobjects o
where i.id = o.id
and i.id > 100
and (indid = 1 or indid = 0)
```

For more information about the `data_pgs` command, see the *Reference Manual*.

8.4.4 Varchars, cursors, and trailing spaces

This section discusses a clarification in the chapter “System and User-Defined Datatypes.” Insert the following as the first paragraph in “Length and storage size” in the section “Character datatypes”:

“Character variables strip the trailing spaces from strings when the variable is populated in a varchar column of a cursor.”

8.5 Reference Manual Volume 2: Commands

This section discusses updates and clarifications for *Reference Manual Volume 2: Commands*.

8.5.1 Minimum size that *alter database* extends a database

The description for 2-1 in the `alter database` section is incorrect, and should be replaced with the following:

“`alter database` extends a database by the following amount if you do not specify a value for the *size* parameter:

```
2K – 1M
4K – 1M
8K – 2M
16K – 4M”
```


8.5.2 *prefetch_size* values

This section discusses updates and clarifications for the *prefetch_size* parameter, used in the select, update, and delete commands.

New valid values for *prefetch_size*

When using *prefetch* and designating the *prefetch_size* (*prefetch_size*) the minimum is 2K and any power of two on the logical page size up to 16K. Usually, the *prefetch_size* is used in select, update and delete statements.

prefetch_size options in kilobytes:

Logical page size	Prefetch size options
2	2, 4, 8 16
4	4, 8, 16, 32
8	8, 16, 32, 64
16	16, 32, 64, 128

The *prefetch_size* specified in the query is only a suggestion. To allow the size specification configure the data cache at that size. If you do not configure the data cache to a specific size, the default *prefetch_size* is used.

To configure the data cache size, use `sp_cacheconfigure`.

8.5.3 Using *alter table* in *if...else* command

The reference pages for the *if... else* command includes this note, which is incomplete:

Note When a create table or create view command occurs within an *if...else* block, Adaptive Server creates the schema for the table or view before determining whether the condition is true. This may lead to errors if the table or view already exists.

The note should read:

Note When an alter table, create table, or create view command occurs within an *if...else* block, Adaptive Server creates the schema for the table or view before determining whether the condition is true. This may lead to errors if the table or view already exists.

8.5.4 Cursor scope description in *declare cursor* command incomplete

The reference pages for *declare cursor* command includes the following description of cursor scope, which is incomplete:

Cursor scope

- A cursor name must be unique within a given scope. Adaptive Server detects name conflicts within a particular scope only during run time. A stored procedure or trigger can define two cursors with the same name if only one is executed. For example, the following stored procedure works because only one `names_crsl` cursor is defined in its scope:

```
create procedure proc2 @flag int
as
if @flag > 0
    declare names_crsl cursor
    for select au_fname from authors
else
    declare names_crsl cursor
    for select au_lname from authors
return
```

Replace the section with the following:

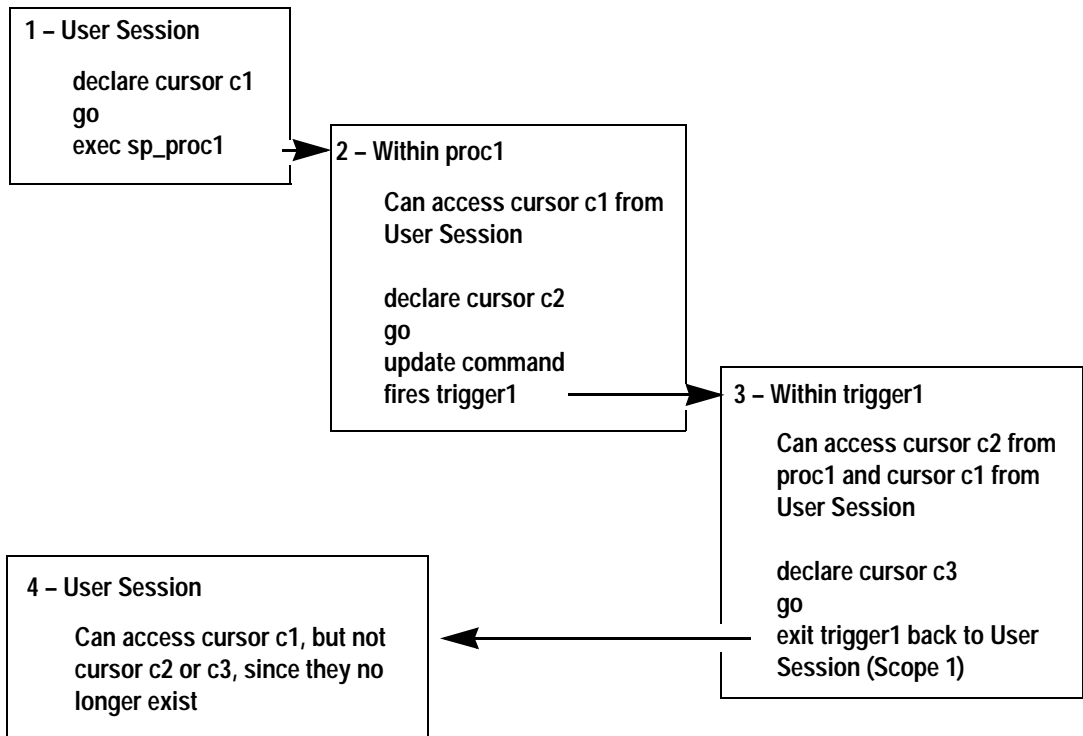
Cursor scope

- A cursor's existence depends on its *scope*. The scope refers to the context in which the cursor is used, that is, within a user session, within a stored procedure, or within a trigger.

Within a user session, the cursor exists only until the user ends the session. The cursor does not exist for any additional sessions started by other users. After the user logs off, Adaptive Server deallocates the cursors created in that session.

If a *declare cursor* statement is part of a stored procedure or trigger, the cursor created within it applies to stored procedure or trigger scope and to the scope that launched the stored procedure or trigger. Cursors declared inside a trigger on an inserted or a deleted table are not accessible to any nested stored procedures or triggers. However, cursors declared inside a trigger on an inserted or a deleted table *are* accessible within the scope of the trigger. Once the stored procedure or trigger completes, Adaptive Server deallocates the cursors created within it.

Figure 1 illustrates how cursors operate between scopes.

Figure 1: How cursors operate within scopes

- A cursor name must be unique within a given scope. Adaptive Server detects name conflicts within a particular scope only during run time. A stored procedure or trigger can define two cursors with the same name if only one is executed. For example, the following stored procedure works because only one `names_crshr` cursor is defined in its scope:

```
create procedure proc2 @flag int
as
if @flag > 0
    declare names_crshr cursor
    for select au_fname from authors
else
    declare names_crshr cursor
    for select au_lname from authors
return
```

8.5.5 Use of readpast may cause duplicates

Additional information is needed in the table describing isolation levels in the section “Using readpast” in the select description in *Reference Manual Volume 2: Commands*.

The additional information is for the line describing the *1, read committed* line:

Using readpast may produce duplicates and adding the distinct clause will not clear this problem.

To resolve this, when using readpast, use a group by clause *in addition to* a distinct clause to avoid duplicates.

8.5.6 revoke command with *cascade* option

This section is an addition to the “Comments” section of the revoke Transact-SQL command description:

You cannot use revoke with the cascade option to revoke privileges granted by the table owner. For example, the owner of a table (UserA) can grant privileges to another user (UserB) as in this scenario:

```
create table T1 (...)  
grant select on T1 to UserB
```

However, the System Administrator cannot revoke UserB’s privileges using the revoke privileges command with the cascade option as in this statement:

```
revoke select on T1 from UserA cascade
```

This statement revokes the select privileges of the table owner, but does not revoke those privileges from UserB.

By default, all data manipulation language (DML) operations are revoked implicitly for users other than the table owner. Because the sysprotects table contains no records indicating that the table owner has granted and then revoked privileges, the cascade option is not invoked.

You must revoke explicitly the select privilege from UserB.

8.5.7 Comments in *select* and *select into* regarding nullability are inaccurate

Comments section for the select and select into commands include this information:

“Because functions allow null values, any column in the select list that contains a function other than convert or isnull allows null values.”

Datatypes and nullability are implicitly assigned to literal values when the `select into` is used, such as:

```
select x = getdate() into mytable
```

This results in a non-nullable column, regardless of whether `allow nulls by default` is on or not. It depends upon how the `select` commands are used and with what other commands within the syntax.

The `convert()` syntax allows you to explicitly specify the datatype and nullability of the resulting column, not the default.

Workaround: Wrap `getdate()` with a function that does result in a null, such as:

```
select x = nullif(getdate(), "1/1/1900") into mytable
```

Or, use the new `convert()` syntax:

```
select x = convert(datetime null, getdate()) into  
mytable
```

8.5.8 *truncate table* command

The following comment for the `truncate table` command in its reference pages is incorrect:

“`truncate table` deallocates the distribution pages for all indexes; remember to run `update statistics` after adding new rows to the table.”

Replace the comment with the following:

truncate table

Adaptive Server no longer uses distribution pages; statistical information is now stored in the tables `sysstatistics` and `systabstats`.

During `truncate table`, statistical information is no longer deleted (deallocated), so it is unnecessary to run `update statistics` after adding data. `truncate table` does not delete statistical information for the table.

8.6 Reference Manual Volume 3: Procedures

This section discusses updates and clarifications for the *Reference Manual Volume 3: Procedures*.

8.6.1 *sp_audit* and SSL

This section updates the reference pages of `sp_audit` to include information about SSL.

Description	Allows a System Security Officer to configure auditing options.
Syntax	<code>sp_audit option, login_name, object_name [,setting]</code>
Parameters	<i>option</i> is the name of the auditing option to set. The valid auditing options include:

Option	Description
security	<p>Audits the following SSL security-relevant events:</p> <ul style="list-style-type: none"> • Issuing any of the following commands: <ul style="list-style-type: none"> • addcert • dropcert <p>Audits addcert and dropcert actions performed by <code>sp_ssladmin()</code>, and updates <code>sysattributes</code>.</p>

Example Initiates auditing for SSL security-relevant events. Both successful and failed events are audited:

```
sp_audit "security", "all", "all", "on"
go
```

```
Sample records added:
select * from sybsecurity..sysaudits_01 where event=99
go
```

Permissions Only a System Security Officer can execute `sp_audit`.

8.6.2 `sp_audit` incorrectly states that `cmdtext` works for roles

In *Reference Manual Volume 3: Procedures* under `sp_audit`, the description for the `cmdtext` auditing option incorrectly reads:

“Audits all actions of a particular user, or by users with a particular role.”

`sp_audit` checks if the second argument is a valid login, and raises an error message if this is not the case. For this reason, the paragraph should read:

“Audits all actions of a particular user.”

8.6.3 `sp_estspace`

Adaptive Server version 12.5 adds the *pagesize* parameter to `sp_estspace`. The full syntax for `sp_estspace` is:

```
sp_estspace table_name, no_of_rows, fill_factor,
cols_to_max, textbin_len, iosec, page_size
```

pagesize – allows you to estimate the space required for a given table – and all of its indexes – if you migrate the table to a server of the specified page size. You can either specify a page size (2, 4, 8, or 16K) or NULL to use your current page size. Because page allocation allocates the same size page for various objects, the *page_size* value applies to all page types (index, data, text and so on).

Example 1

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 6, null, null, 16, @i, "16k"
```

name	type	idx_level	Pages	Kbytes
blurbs	data	0	8	128
blurbs	text/image	0	6	96
blurbs_ind	clustered	0	1	16
blurbs_ind	clustered	1	1	16

Total_Mbytes

0.25

name	type	total_pages	time_mins
blurbs_ind	clustered	10	0
blurbs	data	6	0

This example is run after adding a clustered index to the blurbs table.

This example is run on a 2K server, and indicates that the blurbs table would require .25MB after it is migrated to a 16K server. Below is the same query run on a 16K server, which verifies the .25MB space requirement:

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 6, null, null, 16, @i, "16k"
```

name	type	idx_level	Pages	Kbytes
blurbs	data	0	8	128
blurbs	text/image	0	6	96
blurbs_ind	clustered	0	1	16
blurbs_ind	clustered	1	1	16

```
Total_Mbytes
```

```
-----
```

```
0.25
```

name	type	total_pages	time_mins
blurbs_ind	clustered	10	0
blurbs	data	6	0

Example 2

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 1000, null, null, 16, @i, "2k"
```

name	type	idx_level	Pages	Kbytes
blurbs	data	0	16	32
blurbs	text/image	0	1000	2000
blurbs_ind	clustered	0	1	2
blurbs_ind	clustered	1	1	2

```
Total_Mbytes
```

```
-----
```

```
1.99
```

name	type	total_pages	time_mins
blurbs_ind	clustered	18	0
blurbs	data	1000	0

This example estimates that, if the blurbs table had a thousand rows in it on a 2K server, it would require 1.99MB of space.

8.6.4 Additional parameter for *sp_unbindrule*

sp_unbindrule includes the *accessrule* parameter for unbinding an access rule. The full syntax for *sp_unbindrule* is:

```
sp_unbindrule objname [, futureonly [, "accessrule" | "all"]]
```

Where *accessrule* indicates that you are unbinding the access rule bound to *objname*, and *all* specifies that you are unbinding all rules bound to *objname*.

For example, this access rule is bound to the publishers table:

```
sp_bindrule empl_access, "publishers.pub_id"
```


To unbind this rule, issue the following:

```
sp_unbindrule "empl_access", NULL, "accessrule"
```

You can use the `all` parameter to unbind both access rules and domain rules. For example, to unbind all the access rules and domain rules on the `publishers` table:

```
sp_unbindrule publishers, null, "all"
```

To unbind the access rule from a user-defined datatype for subsequent uses of this datatype, issue:

```
sp_unbindrule def_ssn, futureonly, "accessrule"
```

To unbind both access rules and domain rules for subsequent uses of this datatype, issue:

```
sp_unbindrule def_ssn, futureonly, "all"
```

8.6.5 `sp_modifystats`

This stored procedure should be included in the *Reference Manual Volume 3: Procedures*.

Description	Allows the System Administrator to modify the density values of a column—or columns—in <code>sysstatistics</code>
Syntax	<pre>sp_modifystats [database].[owner].table_name, {"column_group" "all"}, MODIFY_DENSITY, {range total}, {absolute factor}, "value"</pre> <p>Or,</p> <pre>sp_modifystats [database].[owner].table_name, column_name, REMOVE_SKEW_FROM_DENSITY</pre>
Parameters	<p><i>table_name</i></p> <p>is the name of the table to change. Specify the database name if the table is in another database, and specify the owner's name if more than one table of that name exists in the database. The default value for <i>owner</i> is the current user, and the default value for <i>database</i> is the current database.</p> <p><i>column_group</i></p>

an ordered list of column names. To change a statistic for multiple columns (such as a density value), list the columns in the order used to create the statistic. Separate the column names with commas. For example, if your table has a density statistic on columns a1, a2, a3, a4:

- “a1” modifies column a1.
- “a1 , a2 , a3” modifies the column group a1,a2,a3,
- You can also use a wildcard character , %, with the `column_group` parameter to represent a range of characters. For example, “a1 , % , a3” modifies the groups a1,a2,a3 and a1, a4, a3, and so on; “a1 , %” modifies the groups a1,a2 and a1,a2,a3, and so on, but not a1; “a1%” modifies the groups a1,a2 and a1,a2,a3, and so on, as well as a1.

`all`

modifies all column group for this table. Because “all” is a keyword, it requires quotes.

`MODIFY_DENSITY`

allows you to modify either the range or total density of a column or column group to the granularity specified in the *value* parameter. Range cell density represents the average number of duplicates of all values that are represented by range cells in a histogram. See the *Performance and Tuning Guide* for more information.

`range`

modifies the range cell density.

`total`

modifies the total cell density.

`absolute`

ignore the current value and use the number specified by the *value* parameter.

`factor`

multiply the current statistical value by the *value* parameter.

value

is either the specified density value or a multiple for the current density. Must be between zero and one, inclusive, if `absolute` is specified.

column_name

is the name of a column in that table.

REMOVE_SKEW_FROM_DENSITY

allows the System Administrator to change the total density of a column to be equal to the range density which is useful when data skew is present. Total density represents the average number of duplicates for all values, those in both frequency and range cells. Total density is used to estimate the number of matching rows for joins and for search arguments whose value is not known when the query is optimized. See the *Performance and Tuning Guide* for more information.

REMOVE_SKEW_FROM_DENSITY also updates the total density of any composite column statistics for which this column is the leading attribute. Most commonly, a composite index for which this column is the leading attribute would produce these composite column statistics, but they can also be produced when you issue a composite update statistics command.

Examples

Example 1

Changes the range density for column group c00, c01 in table tab_1 to 0.50000000:

```
sp_modifystats "tab_1", "c00, c01", MODIFY_DENSITY,  
range, absolute, "0.5"
```

Example 2

The total density for column group c00, c01 in tab_1 is multiplied by .5. That is, divided in half:

```
sp_modifystats "tab_1", "c00,c01", MODIFY_DENSITY,  
total, factor, "0.5"
```

Example 3

The total density for all the columns in table tab_1 is multiplied by .5.

```
sp_modifystats "tab_1", "all", MODIFY_DENSITY, total,  
factor, "0.5"
```

Example 4

Total density for all column groups starting with c12 is changed to equal the range density.

```
sp_modifystats "tab_1", "c12" REMOVE_SKEW_FROM_DENSITY
```

Usage

- Allows the System Administrator to modify the density values of a column—or columns—in sysstatistics.

- Use `optdiag` to view a table’s statistics. See the *Performance and Tuning Guide* for more information about table density and using `optdiag`.
- Any modification you make to the statistics with `sp_modifystats` is overwritten when you run update statistics. To make sure you are using the most recent statistical modifications, you should run `sp_modifystats` after you run update statistics.
- Because `sp_modifystats` modifies information stored in the `sysstatistics` table, you should make a backup of statistics before execute running `sp_modifystats` in a production system.

Permissions

Only a System Administrator can execute `sp_modifystats`.

See also

Command	update statistics
System table	sysstatistics

8.6.6 Obtaining status information for commands invoked by `xp_cmdshell`

The reference page for the `xp_cmdshell` extended system procedure does not include the following information:

When running the `xp_cmdshell` extended system procedure, you can check to see whether `xp_cmdshell` itself was successful, as well as whether the command that `xp_cmdshell` invoked was successful.

To find out if `xp_cmdshell` itself was successful

To find out if `xp_cmdshell` was successful in spawning a command to the XP Server, enter:

```
@ret = exec xp_cmdshell command
```

where *command* is the name of the command you ran with `xp_cmdshell`.

Table 1: Return values and meanings of @ret = exec xp_cmdshell command

Value	What it means
0	<code>xp_cmdshell</code> was successful in spawning a command.
1	<code>xp_cmdshell</code> failed and did not spawn a command

To find out if the command invoked by `xp_cmdshell` was successful

To find out if the command you ran using `xp_cmdshell` was itself successful, enter:

```
@ret = exec xp_cmdshell command, return_status
```

where *command* is the name of the command you ran with xp_cmdshell.

Table 2: Return values and meanings of @ret = exec xp_cmdshell command, return_status

Value	What it means
0	The command that xp_cmdshell invoked to run in XP Server was successful.
1	The command that xp_cmdshell invoked to run in XP Server failed and was not successful.

Note Both @ret = exec xp_cmdshell *command* and @ret = exec xp_cmdshell *command*, return_status are backward-compatible. Old stored procedures that do not use the return_status parameter treat @ret = exec xp_cmdshell *command*, return_status as if it were @ret = exec xp_cmdshell *command*.

Also, you can still use the no_output parameter in combination with return_status, in any order.

8.7 Reference Manual Volume 4: Tables

This section discusses updates and clarifications for the *Reference Manual Volume 4: Tables*.

8.7.1 Datatype information for columns

The system table descriptions in *Reference Manual Volume 4: Tables* include erroneous information for the datatypes of some columns. The following describes the correct datatypes for these columns:

System table	Column	Datatype
syscolumns	colid	smallint
	length	int
syscurconfigs	message_num	varchar(20)
	apf_percent	varchar(10)
sysengines	osprocname	char(32)
	status	char(12)
sysindexes	status2	smallint
sysjars	jstatus	smallint
	jname	varchar(30)

System table	Column	Datatype
syskeys	key1 – key8 depkey1 – depkey8	smallint smallint
syslocks	class	varchar(30)
syslogins	password	varbinary(30)
syslogshold	xactid masterxactid xloid	binary(6) binary(6) int
sysmessages	error descriptions	int varchar(1024)
sysprocedures	version	smallint
sysprocesses	origsuid	int
sysprotects	action	smallint
sysroles	status	int
sysrvroles	srid	int
systabstats	spare3	smallint
systhresholds	suid	int
sysusages	crdate	datetime

8.7.2 Tracking create dates for columns

This section describes the crdate column of the system catalogs.

Adaptive Server version 12.5 includes the following new columns in the system catalogs:

- sysusages.crdate
- sysindexes.crdate

These system catalogs store creation dates for some objects, and are useful for tracking schema changes at a high level.

sysdatabases.crdate and sysusages.crdate

sysdatabases.crdate stores the database's creation date. sysusages.crdate includes information about when the databases were altered by alter database, which allows you to track the growth rate of your database.

sysobjects.crate and sysindexes.crate

sysobjects.crate reports the date that objects (tables, views, procedures, defaults, rules, constraints, and so on) were created. The sysobjects.crate remains fixed through the life of the object.

For example, the following query displays the creation date for objects, ordered by type:

```
select id, type, name, crdate from sysobjects order by type
      id      type  name                crdate
-----
288001026    P    sp_procxmode          Jun  8 2001  9:43AM
304001083    P    sp_validlang          Jun  8 2001  9:43AM
320001140    P    sp_getmessage         Jun  8 2001  9:43AM
...
240000855    U    spt_ijdbc_conversion  Jun  8 2001  9:43AM
272000969    U    ijdbc_function_escapes Jun  8 2001  9:43AM
```

Objects such as unique constraints, and referential constraints created as part of a create table statement have the same crdate in sysobjects and sysindexes.

You can track creation dates of constraints added by alter table subsequent to the creation of the table using the sysobjects.crate for the constraint's tableid, which you can retrieve from sysconstraints.

sysindexes.crate displays the date you re-created an index. For a newly created index, sysindexes.crate reports the date when you built the index. If you rebuild existing indexes, sysindexes.crate reports this date. For example, you drop and rebuild a clustered index, and then rebuilt all non-clustered indexes, the crdate value in sysindexes is the same for all indexes you rebuilt as part of this operation. Other DDL operations like reorg rebuild, alter table lock change, or alter table to modify a schema along with data copy are performed, will also cause indexes to be rebuilt.

For DOL tables, as the index ID entry 0 is never changed, the crdate value in sysobjects will be the same as that in sysindexes, unless the table's locking scheme was modified from allpages to datarows or datapages.

If the sysindexes.crate is more recent than sysobjects.crate, it means that the index was either built or rebuilt much after the initial table creation.

8.8 System Administration Guide

This section discusses updates and clarifications for the *System Administration Guide*.

8.8.1 Configuration parameters for user-defined login security

This section documents information about configuration parameters for user-defined login security (UDLS), and should go into Chapter 5, “Setting Configuration Parameters,” of the *System Administration Guide*.

maximum failed logins

This section provides information about maximum failed logins, a new configuration parameter.

Summary information

Name in pre-12.0 version	N/A
Default value	0
Range of values	0 – 32767
Status	Dynamic
Display level	10
Required role	System Security Officer

maximum failed logins allows you to set the server-wide maximum number of failed login attempts for logins and roles. For example, to set the system-wide maximum failed logins to 5, enter:

```
sp_configure "maximum failed logins", 5
```

Use create role to set maximum failed logins for a specific role or creation. To create the intern_role role with the password “temp244”, and set maximum failed logins for intern_role to 20, enter:

```
create role intern_role with passwd "temp244", maximum
failed logins 20
```

Use sp_modifylogin to set or change maximum failed logins for an existing login. To change maximum failed logins for the login “joe” to 40, enter:

```
sp_modifylogin "joe", @option="maximum failed logins",
@value="40"
```

Note The *value* parameter is a character datatype; therefore, quotes are required for numeric values.

To change the overrides for maximum failed logins for all logins to 3, enter:

```
sp_modifylogin "all overrides", "maximum failed
logins", "3"
```


To remove the overrides for maximum failed logins option for all logins, enter:

```
sp_modifylogin "all overrides", @option="maximum failed
logins", @value="-1"
```

Use `alter role` to set or change the maximum failed logins for an existing role. For example, to change the maximum failed logins allowed for `physician_role` to 5, enter:

```
alter role physician_role set maximum failed logins 5
```

To remove the overrides for maximum failed logins for all roles, enter:

```
alter role "all overrides" set maximum failed logins -1
```

check password for digit

This section provides information about `check password for digit`, a new configuration parameter.

Summary information

Name in pre-12.0 version	N/A
Default value	0
Range of values	1, 0
Status	Dynamic
Display level	10
Required role	System Security Officer

The System Security Officer can tell the server to check for at least one character or digit in a password using the server-wide configuration parameter `check password for digit`. If set, this parameter does not affect existing passwords. By default, checking for digits is off.

To activate `check password for digit` functionality, enter:

```
sp_configure "check password for digit", 1
```

To deactivate `check password for digit` functionality, enter:

```
sp_configure "check password for digit", 0
```

minimum password length

This section provides information about `minimum password length`, a new configuration parameter.

Summary information

Name in pre-12.0 version	minimum password length
--------------------------	-------------------------

Summary information

Default value	6
Range of values	0 – 30
Status	Dynamic
Display level	10
Required role	System Security Officer

minimum password length allows you to customize the length of server-wide password values or per-login or per-role password values to fit your personal needs. The per-login or per-role minimum password length value overrides the server-wide value. Setting minimum password length affects only the passwords you create after you have set the value; existing password lengths are not changed.

Use minimum password length to specify a server-wide value for minimum password length for both logins and roles. For example, to set the minimum password length for all logins and roles to 4 characters, enter:

```
sp_configure "minimum password length", 4
```

To set minimum password length for a specific login at creation, use `sp_addlogin`. For example, to create the new login “joe” with the password “Djdiek3”, and set minimum password length for “joe” to 4, enter:

```
sp_addlogin joe, "Djdiek3", minimum password length=4
```

To set minimum password length for a specific role at creation, use `create role`. To create the new role “intern_role” with the password “temp244” and set the minimum password length for “intern_role” to 0, enter:

```
create role intern_role with passwd "temp244", minimum
password length 0
```

The original password is seven characters, but the password can be changed to one of any length because the minimum password length is set to 0.

Use `sp_modifylogin` to set or change minimum password length for an existing login. `sp_modifylogin` only effects user roles, not system roles. For example, to change minimum password length for the login “joe” to 8 characters, enter:

```
sp_modifylogin "joe", @option="minimum password
length", @value="8"
```

Note The *value* parameter is a character datatype; therefore, quotes are required for numeric values.

To change the value of the overrides for minimum password length for all logins to 2 characters, enter:

```
sp_modifylogin "all overrides", "minimum password
length", @value="2"
```

To remove the overrides for minimum password length for all logins, enter:

```
sp_modifylogin "all overrides", @option="minimum
password length", @value="-1"
```

Use alter role to set or change the minimum password length for an existing role. For example, to set the minimum password length for “physician_role”, an existing role, to 5 characters, enter:

```
alter role physician_role set minimum password length 5
```

To override the minimum password length for all roles, enter:

```
alter role "all overrides" set minimum password length
-1
```

8.8.2 dbcc checkstorage

On data-only-locked (DOL) tables, there are additional types of returned codes for dbcc checkstorage. The dbcc_types tables in the dbccdb have all the codes and definitions.

Statistic Types

Code	Definition
5020	Deleted rows, the number of deleted rows in the object
5021	Forwarded rows, number of forwarded rows in the object
5022	The number of pages allocated but not containing data
5023	Pages with garbage, number of pages that could benefit from garbage collection
5024	Non-contiguous free space, number of bytes of non-contiguous free space

Fault parameter types:

Code	Definitions
10010	Data Only Locked data page header, the 44 byte page header for the page where the fault is located
10011	Data Only Locked btree leaf page header, the 44 byte page header for the page where the fault is located
10012	Data Only Locked btree page header, the 44 byte page header for the page where the fault is located

Fault types:

Code	Definitions
100033	Non-contiguous free space error, invalid or inconsistent value for the non-contiguous free space on the page
100034	Insert free space error, invalid or inconsistent value for the contiguous free space on the page
100035	spacebits mismatch, inconsistency in the page fullness indicator
100036	deleted row count error, invalid or inconsistent value for the deleted row count on the page
100037	Forwarded rows error, inconsistency between the forwarded rows indicator and the number of forwarded rows on the page
100038	Page header type error, Page header format indicator set incorrectly

8.8.3 Data structures used for storing *text* and *image* data

This section describes the data structures Adaptive Server uses for storing text or image datatypes.

When you allocate text or image data, a 16-byte text pointer is inserted into the row you allocated. Part of this text pointer is a pointer (to a text page number) for the head of the text or image data. This text pointer is known as the first text page (FTP). The FTP is the beginning of a doubly-linked list of text pages.

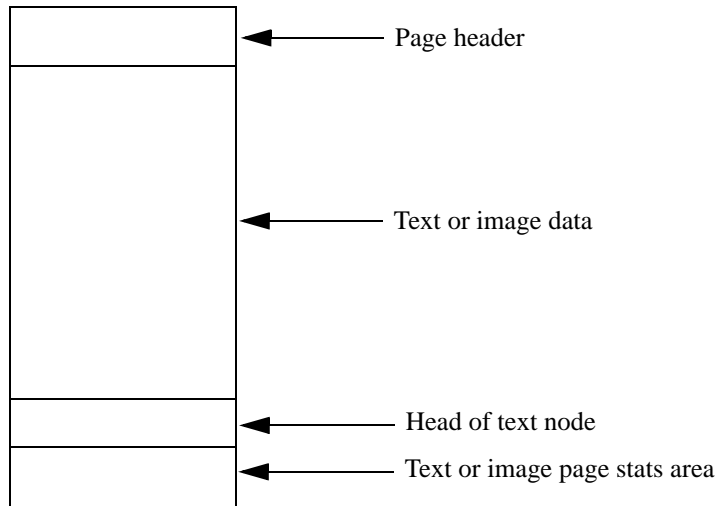
Once an FTP is allocated for a text or image data, it is never deallocated. If an update to an existing text or image data row results in fewer text pages than are currently allocated to this text or image data, Adaptive Server deallocates the extra text pages. If an update to a text or image data sets the FTP to NULL, all pages except the FTP are deallocated.

The data row and the text or image data have the following relationship:

- The page header contains Sybase page header information.
- The text or image data area contains the user text or image data.
- The L0-cache contains the head of the text-node structure (shown in Figure 4, below).
- The text or image page stats area contains information for managing the text or image data. This information includes multibyte translation information (for multibyte servers) and other status information.

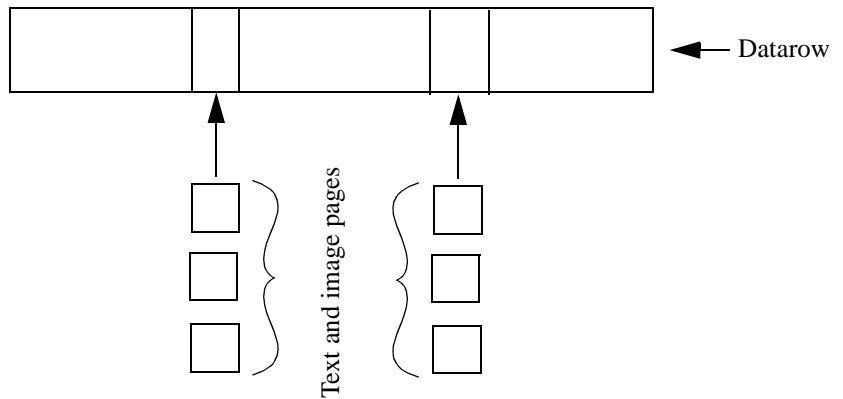
Figure 2 and Figure 3 describe this relationship. Figure 2 describes an individual page's layout, and Figure 3 describes how these pages relate to the text pointer and the data row.

Figure 2: Description of the text or image page layout



Each text page of text or image data has the following layout:

Figure 3: Relationship between the textpointer and datarows



Text nodes

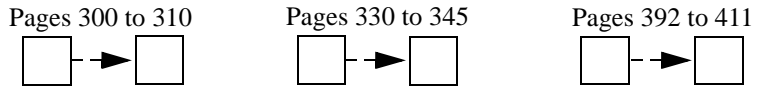
A text node is a hierarchical tree data structure that maps byte offsets (and character offsets for multibyte servers) to text pages for text data. Text nodes are used for:

- Text-page prefetch

- Indexing into text or image data when starting offsets are specified
- Indexing into text or image data when starting offsets are specified for readtext()

Text nodes are optimized for instances where text or image data are typically allocated with multiple runs of consecutive page numbers. For example:

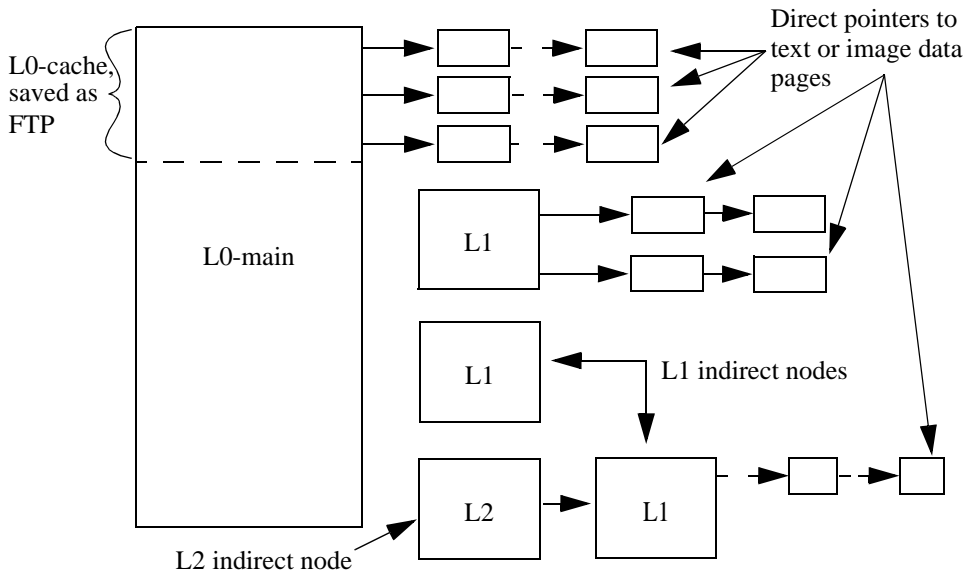
Figure 4: How text or image page numbers are allocated



In this example, the text or image data is made up of three separate runs of consecutive page numbers, (300 to 310), (330 to 345), and (392 to 411). For this case, only three text node entries are needed.

Figure 5 describes the structure of a text node. L0-cache is the text node, and L1 and L2 are indirect nodes that point to text or image data pages.

Figure 5: Structure of the text node



The first section of the text node stored in the L0-cache part of the FTP. Depending on the length of the text data, and how the text pages were allocated, Adaptive Server may allocate extra text pages to store the rest of the text node data structure. These extra text pages are each a text page chain, and are connected to the text data with the FTP of the text data value.

Whether or not the text data requires extra pages—and the amount of extra text pages needed—is dependent on the size of the text data, and how ‘consecutive’ the text pages are that are allocated for the text data.

8.8.4 Heap memory

The section titled “Heap memory” in Chapter 11, “Configuring Memory” should include the following:

The syntax for heap memory per user is:

```
sp_configure 'heap memory per user', amount_of_memory
```

By default, the amount of memory is in 2K pages. This example specifies 10 2K pages of heap memory per user:

```
sp_configure 'heap memory per user', 10
```

You can also specify the amount of memory in the number of bytes per user. For example, the following example specifies that each user connection is allocated 4K bytes of heap memory:

```
sp_configure 'heap memory per user', 0, "4K"
```

At the initial Adaptive Server configuration, 1MB is set aside for heap memory. Additional heap memory is allocated for all the user connections and worker processes for which the server is configured, so the following configuration parameters affect the amount of heap memory available when the server starts:

- number of user connections
- number of worker processes

The global variable @@heapmemsize reports the size of the heap memory pool, in bytes.

Calculating heap memory

To calculate how much heap memory Adaptive Server sets aside, perform the following:

$$((1024 * 1024) + (\text{heap memory in bytes}) * (\text{number of user connections} + \text{number of worker processes})) * 1.05$$

The initial value of (1024 * 1024) is the 1MB initial size of the heap memory pool. The final multiplier of 1.05 accounts for an approximate 5 percent overhead for internal structures.

For example, if your server is configured for:

- heap memory per user – 4K
- number of user connectins – 25 (the default)
- number of worker processes – 25 (the default)

@@heapmemsize reports 1378304 bytes.

And the estimated value using the formula above, is:

$$((1024 * 1024) + (4 * 1024 * 50)) * 1.05 = 1316044.80$$

Now, if you increase the number of user connections, the size of the heap memory pool increases accordingly:

```
sp_configure 'user connections', 100
```

@@heapmemsize reports 1716224 bytes.

The estimated value in this case comes out to be:

$$((1024 * 1024) + (4 * 1024 * (100 + 25))) * 1.05 = 1638604.80$$

If your applications were to fail with the following error message:

```
There is insufficient heap memory to allocate %ld bytes.
Please increase configuration parameter 'heap memory
per user' or try again when there is less activity on
the system.
```

You can increase the heap memory available to the server by increasing one of:

- heap memory per user
- number of user connections
- number of worker processes

Sybase recommends that you first try to increase the heap memory per user configuration option before you increase number of user connections or number of worker processes. Increasing the number of user connections and number of worker processes first consumes system memory for other resources, which may cause you to increase the server's max memory.

See Chapter 5, “Setting Configuration Parameters” in the *System Administration Guide* for more information on how to make memory related configuration option changes.

8.8.5 Unicode sort orders

Adaptive Server version 12.5 includes additional Unicode sort orders. The following is an updated version of Table 7-6 in Chapter 7, “Configuring Character Sets, Sort Orders, and Languages,” that includes this information:

Table 3: Default Unicode sort orders

Name	Description
defaultml	default Unicode ML ordering
nocasep	English, case insensitive, with preference
binary	default binary ordering
thaidict	Thai dictionary ordering
scandict	Scandinavian, dictionary
scannocp	Scandinavian, case insensitive
dict	English dictionary
nocase	English, case insensitive
noaccent	English, accent insensitive
espdict	Spanish, dictionary
espnocs	Spanish, case insensitive
espnoac	Spanish, accent insensitive
rusdict	Russian, dictionary
rusnocs	Russian, case insensitive
cyrdict	Cyrillic, dictionary
cyrnocs	Cyrillic, case insensitive
elldict	Greek, dictionary
hundict	Hungarian, dictionary
hunnoac	Hungarian, accent insensitive
hunnocs	hungarian, case insensitive
turdict	Turkish, dictionary
turknoac	Turkish, accent insensitive
turknocs	Turkish, case insensitive
sjisbin	Japanese, sjis binary
euojisbn	Japanese euojis
gb2312bn	Chinese gb2312
cp932bin	Japanese cp932

Name	Description
big5bin	Chinese b165
euckcsbn	Korean euckcs
utf8bin	matches Unicode UTF-8 binary sort order
gbpinyin	Chinese gb2312 pinyin

8.8.6 sp_help and sp_depend display access rule information

sp_help and sp_depends display access rule information for columns. The Rule_name column of the output displays the domain rule attached to the column, and the Access_Rule_name column displays the name of the access rule.

For example, after adding the following access rules to the pubs2 database:

```
create access rule empl_id_access as @empid < "1111"
```

and then binding them to the publishers table:

```
sp_bindrule empl_id_access, "publishers.pub_id"
```

sp_help displays the the following access rules for the publishers table (this example only displays the pertinent information for space reasons):

```
1> sp_help publishers
. . .
Name                Owner                Object_type
-----
publishers          dbo                   user table

Data_located_on_segment      When_created
-----
default                     Jun  8 2001 10:05AM

Column_name      Type                Length      Prec Scale Nulls Default_name
Rule_name        Access_Rule_name      Identity

-----
pub_id           char                4 NULL  NULL  0 NULL
pub_idrule       empl_id_access
pub_name         varchar              40 NULL  NULL  1 NULL
NULL            NULL
. . .
```

If you issue sp_help with only the access rule name, it describes if it an OR or an AND access rule:

```
sp_help name_access
```

Name	Owner	Object_type
empl_id_access	dbo	AND accessrule
Data_located_on_segment	When_created	
not applicable	Jun 12 2001	4:18PM

name_access access rule is an OR access rule in the publishers table.

sp_depends displays the dependencies for the publishers table, including the access rules bound to this table:

```
1> sp_depends publishers
```

Object doesn't reference any object and no objects reference it.
 Dependent objects that reference all columns in the table. Use sp_depends on each column to get more information.
 Columns referenced in stored procedures, views or triggers are not included in this report.

Column	Type	Object Names or Column Names
pub_id	accessrule	empl_id_access
pub_id	index	pubind (pub_id)
pub_id	logical RI	From titles (pub_id) To publishers (pub_id)
pub_id	logical RI	publishers (pub_id)
pub_id	rule	pub_idrule
pub_id	statistics	(pub_id)

8.8.7 Setting limits for *tempdb* space usage

Adaptive Server version 12.5 provides the tempdb_space resource limit, which restricts the number of pages a tempdb database can have during a single session. If a user exceeds the specified limit, the session can be terminated or the batch or transaction aborted.

tempdb_space is similar to other resource limit types such as io_cost, row_count, and elapsed_time, that help System Administrators (SAs) and Database Administrators (DBAs) prevent queries and transactions from monopolizing server resources. Resource limits are described in detail in Chapter 18, "Limiting Access to Server Resources," in the *System Administration Guide*.

For queries executed in parallel, the tempdb_space resource limit is distributed equally among the parallel threads. For example, if the tempdb_space resource limit is set at 1500 pages and a user executes the following with three-way parallelism, each parallel thread can create a maximum of 500 pages in tempdb:

```
select into #temptable from partitioned_table
```

The SA or DBA sets the tempdb_space limit using sp_add_resource_limit, and drops the tempdb_space limit using sp_drop_resource_limit.

The function and scope of tempdb_space is as follows:

Limit type	Use for queries that	Measuring resource using	Scope	Enforced during
tempdb_space	Use all space in tempdb when creating work or temporary tables.	Number of pages used in tempdb per session.	Query batch or transaction	Execution

8.8.8 Unicode sort orders

Adaptive Server version 12.5 includes additional Unicode sort orders. The following is an updated version of Table 7-6 in Chapter 7, “Configuring Character Sets, Sort Orders, and Languages,” that includes this information:

Table 4: Default Unicode sort orders

Name	Description
defaultml	Default Unicode ML ordering
nocasep	English, case insensitive, with preference
binary	Default binary ordering
thaidict	Thai dictionary ordering
scandict	Scandinavian, dictionary
scannocp	Scandinavian, case insensitive
dict	English dictionary
nocase	English, case insensitive
noaccent	English, accent insensitive
espdict	Spanish, dictionary
espnocs	Spanish, case insensitive
espnocac	Spanish, accent insensitive
rusdict	Russian, dictionary
rusnocs	Russian, case insensitive
cyrdict	Cyrillic, dictionary
cyrnocs	Cyrillic, case insensitive
elldict	Greek, dictionary
hundict	Hungarian, dictionary

Name	Description
hunnoac	Hungarian, accent insensitive
hunnocs	hungarian, case insensitive
turdict	Turkish, dictionary
turknoac	Turkish, accent insensitive
turknocs	Turkish, case insensitive
sjisbin	Japanese, sjis binary
ucjisbn	Japanese ucjis
gb2312bn	Chinese gb2312
cp932bin	Japanese cp932
big5bin	Chinese b165
euckcsbn	Korean euckcs
utf8bin	Matches Unicode UTF-8 binary sort order
gbpinyin	Chinese gb2312 pinyin

8.9 *Transact-SQL User's Guide*

This section discusses updates and clarifications for the *Transact-SQL User's Guide*.

8.9.1 System test can fail if too many aggregates in *compute* clause

The compute clause section of Chapter 3, “Using Aggregates, Grouping, and Sorting” should contain the following information:

A system test may fail because there are too many aggregates in the compute clause of a query. The number of aggregates that each compute clause can accommodate is limited to 127, and if a compute clause contains more than 127 aggregates, the system generates an error message when you try to execute the query.

Each avg() aggregate counts as two aggregates when you are counting toward the limit of 127, because an avg() aggregate is actually a combination of a sum() aggregate and a count() aggregate.

8.9.2 DDL commands allowed in transactions

This section discusses updates and clarifications for the alter table commands.

Table 18-1, “DDL commands allowed in transactions” incorrectly states that all alter table commands are allowed except with partition and unpartition table. This information is incorrect.

You cannot issue the alter table command with a partition or unpartition clause *within* a user-defined transaction.

8.10 Utility Guide

This section discusses updates and clarifications for the *Utility Guide*.

8.10.1 *bcp out* and partitioned tables

In Chapter 3, “Using *bcp* to Transfer Data to and from Adaptive Server,” the subsection titled “Using parallel bulk copy to copy data into a specific partition” in the section titled “*bcp* performance,” contains the following note:

Note When using parallel bulk copy to copy data out, you cannot specify which partitions *bcp* should use.

This note is incomplete, and should be replaced with the following:

bcp out and partitioned tables

bcp does not support parallel sessions of *bcp out* from a table that is partitioned across I/O devices. For this reason, you cannot specify which partitions *bcp* should use when you use parallel bulk copy to copy out data. If you perform a *bcp out* when a table is partitioned, *bcp* may create duplicate records. To ensure data integrity, you must unpartition a table before attempting to *bcp out* of it.

Example

The following sequence of *bcp* commands, issued against a partitioned table called *customer* in the *pubs2* database, risks the duplication of data that is copied out.

```
bcp pubs2..customer out customer.dat1 -Uuser -Ppwd
-c -F1 -L500 &
bcp pubs2..customer out customer.dat2 -Uuser -Ppwd
-c -F501 -L1000 &
bcp pubs2..customer out customer.dat3 -Uuser -Ppwd
-c -F1001 &
```

What does work correctly is a single process of *bcp out* that is running against a partitioned table. The following does not generate duplicate records:

```
bcp pubs2..customer out customer.dat -Uuser -Ppwd
-c &
```

8.10.2 Using bcp to transfer data to and from Adaptive Server

This updates information on the bulk copy (bcp) utility in Chapter 3, “Using bcp to transfer data to and from Adaptive Server” in the *Utility Guide*:

Bulk copy utility

When you are using the bulk copy (bcp) to copy a very large number of rows, do *not* use -F for the first row and still use multiple input files. The -F option is for single processing and ad hoc copying, not the heavy-duty, multiprocess copying found in a production environment.

8.10.3 bcp and row-level access rules

If Adaptive Server is enabled for row-level access, and you bulk-copy-out data, bcp copies out only the rows of data to which you have access. To copy out the entire table, you must first drop the access rules, then bcp out. Reinststate the access rules after you are done, if applicable.

If you bulk-copy-in data to a table that has access rules enabled, Adaptive Server may issue “uniqueness violation” errors. For example, if you load data from a bcp data file that was generated before the access rules were created on the table, and the bcp data file contains rows that were previously inserted into the table, you may receive this type of error.

If this happens, the table may look to the user like it does not include the rows that failed the bcp insert because of the uniqueness violation, but the user does not have access to the “missing” rows because of the access rules.

To copy in the entire table, drop the access rules, load the data, address any errors, then reinststate the access rules.

8.11 What’s New in Sybase Adaptive Server Enterprise?

The following are updates for the *What’s New in Adaptive Server 12.5?* guide.

8.11.1 Changed Transact-SQL commands

Table 5 summarizes the changes to existing Transact-SQL commands.

Table 5: Changed commands

Command	Change
disk init	Allows the following unit specifiers for the device size: 'k' or 'K' (kilobytes), 'm' or 'M' (megabytes), and 'g' or 'G' (gigabytes)
disk reinit	Allows the following unit specifiers for the device size: 'k' or 'K' (kilobytes), 'm' or 'M' (megabytes), and 'g' or 'G' (gigabytes)

Command	Change
create table	Supports the new <code>size_in_bytes</code> parameter, which allows you to specify the size of an in-row Java-SQL column.
create database	Allows the following unit specifiers for the device size: 'k' or 'K' (kilobytes), 'm' or 'M' (megabytes), and 'g' or 'G' (gigabytes)
alter database	Allows the following unit specifiers for the device size: 'k' or 'K' (kilobytes), 'm' or 'M' (megabytes), and 'g' or 'G' (gigabytes)
set	You can update as many as 1024 columns in the set clause using literals, variables, or expressions returned from a subquery
select...for browse	You cannot use the <code>select...for browse</code> option on tables containing more than 255 columns
select select_list into...at pathname	Allows you to specify a proxy table as the target of a <code>select into</code> command. The <code>at</code> indicates that the new table is a proxy table.
compute	If a compute clause includes a group by clause: <ul style="list-style-type: none"> • The compute clause cannot contain more than 127 aggregates. • The group by clause cannot contain more than 255 columns. Columns included in a compute clause cannot be longer than 255 bytes.
like	The character string indicated by the <code>like</code> keyword cannot be longer than the column length. The maximum column length is 16K.
declare cursor	You can include as many as 1024 columns in an update clause of a client's <code>declare cursor</code> statement
+ operator	Returns result strings up to a length of 16384 bytes
dump database	Includes compression syntax for making a compressed dump
load database	Includes compression syntax for loading a compressed dump
dump transaction	Includes compression syntax for making a compressed dump
load transaction	Includes compression syntax for loading a compressed dump
print statement	The total length of the format string + the output cannot exceed 1K. However, Adaptive Server does not issue an error message if you exceed this length.
create procedure (SQLJ)	Creates a SQLJ stored procedure, which is a Java method with a SQL name. Adaptive Server executes a SQLJ stored procedure in the same way it executes a Transact-SQL stored procedure.
create function (SQLJ)	Creates a user-defined function by adding a SQL wrapper to a Java static method. Can return a value defined by the method.
create access rule	Allows you to create a rule determining what data a user can access.
union in view	You are now allowed to use the <code>union</code> command in a view

New global variables

Adaptive Server version 12.5 adds the following global variables:

- `@@maxpagesize` – displays the logical page size for which the server is configured. The value is displayed in bytes.

- @@heapmemsize – displays the amount of heap memory for which the server is configured.

Changed global variables

The internal global variable, @@pagesize has changed to display the virtual page size.

8.11.2 Changed system procedures

Table 6 summarizes the changes made to existing system procedures.

Table 6: Changed system procedures

Procedure	Change
sp_helpjava	Supports the new depends keyword, which allows you to view a list of database objects that depend on the specified Java class or JAR.
sp_configure	sp_configure allows you to specify the value for configuration parameters in unit specifiers. The unit specifiers are p or P for pages, m or M for megabytes, and g or G for gigabytes. sp_configure also reports on whether the parameter is static or dynamic, and the units for each option.
sp_serveroption	Supports the new external engine auto start clause, which enables or disables automatic start-up of EJB Server whenever Adaptive Server starts up. The default is true.

8.11.3 Changed status in existing columns

The sysconstraints row is incorrect in Table 2-14 of the *What's New in Adaptive Server 12.5?* guide. The following replaces it:

Table 2-14: Changed status in existing columns

Table name	Column name	Old datatype	New datatype
sysconstraints	colid	tinyint	smallint

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 Adaptive Server Enterprise 12.5 Product Manuals CD and the Sybase Product Manuals Web site to learn more about your product:

- The Product Manuals CD, included with your software, contains product manuals in Portable Document Format (PDF).

Note Both the label and *README* file in the Product Manuals CD show Adobe Acrobat Reader as available on the CD, in the form of a tar file. This file does not exist on the CD. For this reason, follow the instructions in the *Installation Guide for Linux* file to obtain the software online.

- Sybase Product Manuals Web site is an HTML version of the Product Manuals CD that you can access using a standard Web browser. In addition to product manuals, you'll find links to the Technical Documents Web site, the Solved Cases page, and Sybase newsgroups.

Access the Sybase Product Manuals Web site at <http://www.sybase.com/support/manuals/>.

10.1 Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

❖ **For the latest information on product certifications**

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

❖ For the latest information on EBFs and Updates

- 1 Point your Web browser to Technical Documents at <http://www.sybase.com/support/techdocs/>.
- 2 Select EBFs/Updates. Enter user name and password information, if prompted (for existing web accounts) or create a new account (a free service).
- 3 Specify a time frame and click Go.
- 4 Select a product.
- 5 Click an EBF/Update title to display the report.

❖ To create 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.

