SYBASE®

What's New In Adaptive Server Enterprise?

Adaptive Server® Enterprise
15.0

DOCUMENT ID: DC37429-01-1500-01

LAST REVISED: September 2005

Copyright © 1987-2005 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase, the Sybase logo, ADA Workbench, Adaptable Windowing Environment, Adaptive Component Architecture, Adaptive Server, Adaptive Server Anywhere, Adaptive Server Enterprise, Adaptive Server Enterprise Monitor, Adaptive Server Enterprise Replication, Adaptive Server Everywhere, Adaptive Warehouse, Afaria, Answers Anywhere, Anywhere Studio, Application Manager, AppModeler, APT Workbench, APT-Build, APT-Edit, APT-Execute, APT-Translator, APT-Library, AvantGo Mobile Delivery, AvantGo Mobile Inspection, AvantGo Mobile Marketing Channel, AvantGo Mobile Pharma, AvantGo Mobile Sales, AvantGo Pylon, AvantGo Pylon Application Server, AvantGo Pylon Conduit, AvantGo Pylon PIM Server, AvantGo Pylon Pro, Backup Server, BizTracker, ClearConnect, Client-Library, Client Services, Convoy/DM, Copernicus, Data Pipeline, Data Workbench, DataArchitect, Database Analyzer, DataExpress, DataServer, DataWindow, DataWindow .NET, DB-Library, dbQueue, Developers Workbench, DirectConnect, DirectConnect Anywhere, Distribution Director, e-ADK, E-Anywhere, e-Biz Impact, e-Biz Integrator, E-Whatever, EC Gateway, ECMAP, ECRTP, eFulfillment Accelerator, Embedded SQL, EMS, Enterprise Application Studio, Enterprise Client/Server, Enterprise Connect, Enterprise Data Studio, Enterprise Manager, Enterprise SQL Server Manager, Enterprise Work Architecture, Enterprise Work Designer, Enterprise Work Modeler, eProcurement Accelerator, EWA, Financial Fusion, Financial Fusion Server, Gateway Manager, GlobalFIX, iAnywhere, iAnywhere Solutions, ImpactNow, Industry Warehouse Studio, InfoMaker, Information Anywhere, Information Everywhere, InformationConnect, InternetBuilder, iScript, Jaguar CTS, jConnect for JDBC, M2M Anywhere, Mach Desktop, Mail Anywhere Studio, Mainframe Connect, Maintenance Express, Manage Anywhere Studio, M-Business Channel, M-Business Network, M-Business Server, MDI Access Server, MDI Database Gateway, media.splash, MetaWorks, mFolio, Mirror Activator, MySupport, Net-Gateway, Net-Library, New Era of Networks, ObjectConnect, ObjectCycle, OmniConnect, OmniSQL Access Module, OmniSQL Toolkit, Open Biz, Open Client, Open ClientConnect, Open Client/Server, Open Client/Server Interfaces, Open Gateway, Open Server, Open ServerConnect, Open Solutions, Optima++, PB-Gen, PC APT Execute, PC DB-Net, PC Net Library, PocketBuilder, Pocket PowerBuilder, Power++, power.stop, PowerAMC, PowerBuilder, PowerBuilder Foundation Class Library, PowerDesigner, PowerDimensions, PowerDynamo, PowerScript, PowerSite, PowerSocket, Powersoft, PowerStage, PowerStudio, PowerTips, Powersoft Portfolio, Powersoft Professional, PowerWare Desktop, PowerWare Enterprise, ProcessAnalyst, QAnywhere, Rapport, RemoteWare, RepConnector, Replication Agent, Replication Driver, Replication Server, Replication Server Manager, Replication Toolkit, Report-Execute, Report Workbench, Resource Manager, RFID Anywhere, RW-DisplayLib, RW-Library, S-Designor, SDF, Secure SQL Server, Secure SQL Toolset, Security Guardian, SKILS, smart.partners, smart.parts, smart.script, SQL Advantage, SQL Anywhere, SQL Anywhere Studio, SQL Code Checker, SQL Debug, SQL Edit, SQL Edit/TPU, SQL Everywhere, SQL Modeler, SQL Remote, SQL Server, SQL Server Manager, SQL SMART, SQL Toolset, SQL Server/CFT, SQL Server/DBM, SQL Server SNMP SubAgent, SQL Station, SQLJ, STEP, SupportNow, S.W.I.F.T. Message Format Libraries, Sybase Central, Sybase Client/Server Interfaces, Sybase Financial Server, Sybase Gateways, Sybase IQ, Sybase MPP, Sybase SQL Desktop, Sybase SQL Lifecycle, Sybase SQL Workgroup, Sybase User Workbench, SybaseWare, Syber Financial, SyberAssist, SybFlex, SyBooks, System 10, System 11, System XI (logo), SystemTools, Tabular Data Stream, TradeForce, Transact-SQL, Translation Toolkit, UltraLite, UltraLite.NET, UNIBOM, Unilib, Uninull, Unisep, Unistring, URK Runtime Kit for UniCode, VisualWriter, VQL, WarehouseArchitect, Warehouse Control Center, Warehouse Studio, Warehouse WORKS, Watcom, Watcom SQL, Watcom SQL Server, Web Deployment Kit, Web.PB, Web.SQL, WebSights, WebViewer, WorkGroup SQL Server, XA-Library, XA-Server, XcelleNet, and XP Server are trademarks of Sybase, Inc. 02/05

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names used herein may be trademarks or registered trademarks of their respective companies.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

About This Book ..... **vii**

CHAPTER 1 **New Features in Adaptive Server Version 15.0** **1**
- Partition support 2
- Row-locked system catalogs 3
- Query Processor 3
- Large identifiers 4
- Computed columns 5
- Scrollable cursors 6
- Unitext support 6
- big int support 7
- Unsigned int support 7
- Integer identity 8
- Enhancements to XML services 8
  - XML schema support 8
  - Internationalization (I18N) support 8
  - for xml enhancements 9
- Adaptive Server Plug-in enhancements 9
- Interactive SQL 10
- User-defined web services 10
- Very large storage support 10
- Automatic update statistics 11
- SySAM license management 12
- Query processing metrics (qp metrics) 13
- Updates to abstract plans 14
- showplan changes 14
- Secure Socket Layer Uses FIPS 140-2 16

CHAPTER 2 **System Changes in Adaptive Server Version 15.0** **17**
- System changes in version 15.0 17
  - Changes to configuration parameters 17
  - Changes to Transact-SQL commands 19
  - New and changed system procedures 22

Changes to utility programs 23
Changes to databases and system tables 24
New and changed monitoring tables 28
New reserved words 28
New or changed global variables 29

CHAPTER 3 New Features in Adaptive Server 12.5.2 and 12.5.3 31
New features in Adaptive Server 12.5.3 31
Cross platform dump and load 31
Importing statistics for proxy tables 32
Additional support for historical server 32
Additional support for SSL 32
Native XML 33
HP-UX support 33
Enhancements to Real Time Data Services 33
Resource Governor enhancements 34
Page allocation for partitioned DOL tables 34
User connections enhancements 35
dtdValidation support 35
Migration tool enhancements 36
top n functionality 36
New language support 36
Monitor counter concurrency 37
System changes in 12.5.3 37
New features in Adaptive Server 12.5.2 38
Setting the statement cache 39
XML services 39
Real Time Messaging Services 40
Web Services 41
IPv6 support 41
Changes to Enhanced Full-Text Search Capabilities 42
Security changes 43
Password-protected backups 48
Making compressed database dumps 49
Linux Adaptive Server for Failover on Veritas 2.1 49
Large Memory Support for 32-Bit Linux 50
Additions to dbcc complete_xact 50
System changes in 12.5.2 50

CHAPTER 4 New Features in Adaptive Server Version 12.5.1 57
Dynamic data cache 58
Automatic database expansion 58
Improvements to the Adaptive Server plug-in for Sybase Central . 60

Dynamic listeners 61
Globalization enhancements 62
UTF-8 sort order 62
Unicode parser 62
SQL derived tables 62
Date and time datatypes 63
XML mapping 63
Native XML processor 64
Native XML query 65
LDAP user authentication 66
HA agent on VCS 3.5 67
Fast recovery 68
Database mount and unmount 68
Job Scheduler 69
New cache wizard section for sp_sysmon 70
Web Services for Adaptive Server 71
Enhanced compatibility with Microsoft SQL extensions 71
Adaptive Server 12.5.1 backward compatibility 72
System changes in version 12.5.1 72
New and changed configuration parameters 72
New and changed Transact-SQL commands 73
New functions 74
New and changed system procedures 74
New and changed system tables 75
New databases 76
New reserved words 76
Changes to the documentation 76

CHAPTER 5 New Features in Adaptive Server Version 12.5.0.x 79
New features in version 12.5.0.3 79
Monitoring system tables in Adaptive Server 80
Index selection 82
Active-passive configuration for Sun Cluster 3.0 82
Asynchronous log service and optimistic index locking 83
Using sampling for update statistics 84
Multiple temporary databases 84
Housekeeper enhancements 85
Chinese character set (GB18030) support 86
SQLJ aliases for XQL 86
SQLX mapping functions 86
New configuration parameters in sp_monitorconfig 86
Faster checkpoints 87
Changes to checkstorage 87
Changes to CIS 88

Improved performance for select into 89
New set parameters 90
Improved performance for create database, alter database, load database, and dbcc checkalloc 90
Number of databases per transaction 90
System changes 90
New global variables 97
New functions 97
New features in version 12.5.0.1 98
Options for Adaptive Server version 12.5.0.1 98
Using grant dbcc and revoke dbcc 99
Increasing the size of devices with disk resize 99
ASE Replicator 100
Synchronizing EJB Servers 100
XQL result sets 101
XQL query engine 102
Proxy table space 103
Creating files and directories with the insert command 104
Migration utility 104
High availability on Sun Cluster 3.0 105
New character sets 105
System changes 105

Index 109

About This Book

Audience

This manual is for customers who are upgrading to Adaptive Server® version 15.0.

How to use this book

What's New in Adaptive Server Enterprise? is an introduction to the new Sybase® Adaptive Server Enterprise features and the commands, system procedures, system tables, and documentation that supports them.

This manual consists of the following chapters:

- Chapter 1, "New Features in Adaptive Server Version 15.0," describes the features added to the 15.0 version of Adaptive Server
- Chapter 2, "System Changes in Adaptive Server Version 15.0," describes new and changed configuration parameters, Transact-SQL commands, system procedures, databases, system tables, and reserved words
- Chapter 3, "New Features in Adaptive Server 12.5.2 and 12.5.3," describes the features added to the 12.5.2 and 12.5.3 versions of Adaptive Server
- Chapter 4, "New Features in Adaptive Server Version 12.5.1," describes the features added to the 12.5.0.3 version of Adaptive Server.
- Chapter 5, "New Features in Adaptive Server Version 12.5.0.x," describes the features added to the 12.5.0.x versions of Adaptive Server.

Related documents

The Sybase® Adaptive Server® Enterprise documentation set consists of the following:

- The release bulletin for your platform – contains last-minute information that was too late to be included in the books.

  A more recent version of the release bulletin may be available on the World Wide Web. To check for critical product or document information that was added after the release of the product CD, use the Sybase Technical Library.

- The *Installation Guide* for your platform – describes installation, upgrade, and configuration procedures for all Adaptive Server and related Sybase products.
- *What's New in Adaptive Server Enterprise?* – describes the new features in Adaptive Server version 15.0, the system changes added to support those features, and changes that may affect your existing applications.
- *ASE Replicator User's Guide* – describes how to use the Adaptive Server Replicator feature of Adaptive Server to implement basic replication from a primary server to one or more remote Adaptive Servers.
- *Component Integration Services User's Guide* – explains how to use the Adaptive Server Component Integration Services feature to connect remote Sybase and non-Sybase databases.
- The *Configuration Guide* for your platform – provides instructions for performing specific configuration tasks for Adaptive Server.
- *Full-Text Search Specialty Data Store User's Guide* – describes how to use the Full-Text Search feature with Verity to search Adaptive Server Enterprise data.
- *Glossary* – defines technical terms used in the Adaptive Server documentation.
- *Historical Server User's Guide* – describes how to use Historical Server to obtain performance information for SQL Server® and Adaptive Server.
- *Java in Adaptive Server Enterprise* – describes how to install and use Java classes as data types, functions, and stored procedures in the Adaptive Server database.
- *Job Scheduler User's Guide* – provides instructions on how to install and configure, and create and schedule jobs on a local or remote Adaptive Server using the command line or a graphical user interface (GUI).
- *Messaging Service User's Guide* – describes how to useReal Time Messaging Services to integrate TIBCO Java Message Service and IBM WebSphere MQ messaging services with all Adaptive Server database applications.
- *Monitor Client Library Programmer's Guide* – describes how to write Monitor Client Library applications that access Adaptive Server performance data.
- *Monitor Server User's Guide* – describes how to use Monitor Server to obtain performance statistics from SQL Server and Adaptive Server.

- *Performance and Tuning Guide* – is a series of four books for Adaptive Server version 12.5.x that explains how to tune Adaptive Server for maximum performance:
  - *Basics* – the basics for understanding and investigating performance questions in Adaptive Server.
  - *Locking* – describes how the various locking schemas can be used for improving performance in Adaptive Server.
  - *Optimizer and Abstract Plans* – describes how the optimizer processes queries and how abstract plans can be used to change some of the optimizer plans.
  - *Monitoring and Analyzing* – explains how statistics are obtained and used for monitoring and optimizing performance.
- *Quick Reference Guide* – provides a comprehensive listing of the names and syntax for commands, functions, system procedures, extended system procedures, datatypes, and utilities in a pocket-sized book.
- *Reference Manual* – is a series of four books that contains the following detailed Transact-SQL® information:
  - *Building Blocks* – Transact-SQL datatypes, functions, global variables, expressions, identifiers and wildcards, and reserved words.
  - *Commands* – Transact-SQL commands.
  - *Procedures* – Transact-SQL system procedures, catalog stored procedures, system extended stored procedures, and `dbcc` stored procedures.
  - *Tables* – Transact-SQL system tables and `dbcc` tables.
- *System Administration Guide* – provides in-depth information about administering servers and databases. This manual includes instructions and guidelines for managing physical resources, security, user and system databases, and specifying character conversion, international language, and sort order settings.
- *System Tables Diagram* – illustrates system tables and their entity relationships in a poster format. Available only in print version.
- *Transact-SQL User's Guide* – documents Transact-SQL, Sybase's enhanced version of the relational database language. This manual serves as a textbook for beginning users of the database management system. This manual also contains descriptions of the `pubs2` and `pubs3` sample databases.

- *Using Adaptive Server Distributed Transaction Management Features* – explains how to configure, use, and troubleshoot Adaptive Server DTM features in distributed transaction processing environments.
- *Using Sybase Failover in a High Availability System* – provides instructions for using Sybase's Failover to configure an Adaptive Server as a companion server in a high availability system.
- *Unified Agent and Agent Management Console* – Describes the Unified Agent, which provides runtime services to manage, monitor and control distributed Sybase resources.
- *Utility Guide* – documents the Adaptive Server utility programs, such as isql and bcp, which are executed at the operating system level.
- *Web Services User's Guide* – explains how to configure, use, and troubleshoot Web Services for Adaptive Server.
- *XA Interface Integration Guide for CICS, Encina, and TUXEDO* – provides instructions for using the Sybase DTM XA interface with X/Open XA transaction managers.
- *XML Services in Adaptive Server Enterprise* – describes the Sybase native XML processor and the Sybase Java-based XML support, introduces XML in the database, and documents the query and mapping functions that comprise XML Services.

Other sources of information

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

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

- The SyBooks CD contains product manuals and is included with your software. The Eclipse-based SyBooks browser allows you to access the manuals in an easy-to-use, HTML-based format.

  Some documentation may be provided in PDF format, which you can access through the PDF directory on the SyBooks CD. To read or print the PDF files, you need Adobe Acrobat Reader.

  Refer to the *SyBooks Installation Guide* on the Getting Started CD, or the README.txt file on the SyBooks CD for instructions on installing and starting SyBooks.

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

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

Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

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

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

❖ **Finding the latest information on component certifications**

1. Point your Web browser to Availability and Certification Reports at http://certification.sybase.com/.
2. Either select the product family and product under Search by Product; or select the platform and product under Search by Platform.
3. Select Search to display the availability and certification report for the selection.

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

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

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

Sybase EBFs and software updates

❖ **Finding the latest information on EBFs and software maintenance**

1. Point your Web browser to the Sybase Support Page at http://www.sybase.com/support.
2. Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.
3. Select a product.
4. Specify a time frame and click Go. A list of EBFs/Maintenance releases is displayed.

   Padlock icons indicate that you do not have download authorization for certain EBFs/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the "Technical Support Contact" role to your MySybase profile.
5. Click the Info icon to display the EBFs/Maintenance report, or click the product description to download the software.

Conventions

The following style conventions are used in this manual:

- In a sample screen display, commands you should enter exactly as shown are given in:

  ```
  this font
  ```

- In a sample screen display, words which you should replace with the appropriate value for your installation are shown in:

  ```
  this font
  ```

- In the regular text of this document, the names of files and directories appear in italics:

  /usr/u/sybase

- The names of programs, utilities, procedures, and commands appear in bold type:

  bcp

- Commands for both the C and Bourne shells are provided in this document when they differ. The initialization file for the C shell is called *.cshrc*. The initialization file for the Bourne shell is called *.profile*. If you are using a different shell, such as the Korn shell, refer to your shell-specific documentation for the correct command syntax.

The conventions for syntax statements in this manual are as follows:

Table 1: SQL syntax conventions

Key	Definition
command	Command names, command option names, utility names, utility flags, and other keywords are in Arial bold.
variable	Variables, or words that stand for values that you fill in, are in Italics.
{ }	Curly braces indicate that you choose at least one of the enclosed options. Do not include braces in your option.
[ ]	Brackets mean choosing one or more of the enclosed options is optional. Do not include brackets in your option.
( )	Parentheses are to be typed as part of the command.
|	The vertical bar means you can select only one of the options shown.
,	The comma means you can choose as many of the options shown as you like, separating your choices with commas to be typed as part of the command.

Accessibility features

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

Adaptive Server HTML documentation has been tested for compliance with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

Note You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at http://www.sybase.com/accessibility. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

CHAPTER 1

New Features in Adaptive Server Version 15.0

This chapter describes the new features introduced with Sybase Adaptive Server Enterprise version 15.0.

Topic	Page
Partition support	2
Row-locked system catalogs	3
Query Processor	3
Large identifiers	4
Computed columns	5
Scrollable cursors	6
Unitext support	6
big int support	7
Unsigned int support	7
Integer identity	8
Enhancements to XML services	8
Adaptive Server Plug-in enhancements	9
Interactive SQL	10
User-defined web services	10
Very large storage support	10
Automatic update statistics	11
SySAM license management	12
Query processing metrics (qp metrics)	13
Updates to abstract plans	14
showplan changes	14
Secure Socket Layer Uses FIPS 140-2	16

Partition support

Partitioning is particularly useful in managing large tables and indexes by dividing them into smaller, more manageable pieces. Partitions, like a large-scale index, provide faster and easier access to data.

Partitions are database objects and can be managed independently. You can, for example load data, and create index cannot be done at a partition level.. Yet partitions are transparent to the end user, who can select, insert, and delete data using the same commands whether the table is partitioned or not.

Adaptive Server 15.0 supports horizontal partitioning, in which a selection of table rows can be distributed among partitions on different disk devices. Individual table or index rows are assigned to a partition according to a semantic or to a round-robin partitioning strategy.

Semantic partitioning strategies use the data values in specified, key columns in each row to determine the partition assignment of that row. The round-robin partitioning strategy assigns rows randomly without reference to data values.

Partitioning strategies are:

- *Hash partitioning* (semantic) – a system-supplied hash function determines the partition assignment for each row.
- *List partitioning* (semantic) – values in key columns are compared with sets of user-supplied values specific to each partition. Exact matches determine the partition assignment.
- *Range partitioning* (semantic) – values in key columns are compared with a user-supplied set of upper and lower bounds associated with each partition. Key column values falling within the stated bounds determine the partition assignment.
- *Round-robin partitioning* – rows are assigned randomly to partitions in a round-robin manner so that each partition contains a more or less equal number of rows. This is the default strategy.

You can:

- Create partitions when you create a table or index using the create table and create index commands.
- Alter a table's partitioning strategy using the alter table command.
- Add a partition to an existing table with add partition.

- You can use partitioning to expedite the loading of large amounts of table data—even when the table eventually will be used as an unpartitioned table.

Row-locked system catalogs

Adaptive Server version 15.0 converts most system catalogs to a datarows locking scheme. These system catalogs continue to use allpages locking scheme:

- Materialized tables such as syslocks and sysprocesses. These tables are generated during run-time and their locking schemes are irrelavent for concurrency.
- sysmessages and sysusermessages, which are read-only tables
- Auditing tables in sybsecurity, which are write-once and read many times.

Adaptive Server's internal upgrade process converts the system table locking schemes during an installation, upgrade, or load upgrade.

Because DDLs in Adaptive Server release 15.0 use the same table-level locks as 12.5.x and earlier versions, there is no concurrency improvement when you run DDLs.

Query Processor

The Adaptive Server version 15.0 query processor is self-tuning, requiring fewer interventions than earlier versions. This version of Adaptive Server has less reliance on worktables for materialization between steps since the engine supports data flow between steps. However, more worktables could be used in cases where Adaptive Server determines that hash and merge operations are effective.

New features include support for:

- Both vertical and horizontal parallelism for query processing
- Improved index selection, especially for joins with OR clauses and joins and search arguments (SARGs) with mismatched but compatible datatypes

- More efficient algorithms
- Improved costing, using join histograms for joins with data skews in joining columns
- Improved query plan selection that enhances performance through:
  - New index union and index intersection strategies for queries with and/or predicates on different indexes
  - On-the-fly grouping and ordering using in-memory sorting and hashing for queries with group by and order by clauses
  - Cost-based pruning and timeout mechanisms that use permutation search strategies for large, multi-way joins, and for star and snowflake schema joins
- Improved problem diagnosis and resolution using:
  - Searchable XML format trace outputs
  - Diagnostic output from new set commands
- Joins involving a large number of tables
- Data and index partitioning, which are especially beneficial for very large data sets

Partitioning is the basic building block for parallelism.

Adaptive Server release 15.0 provides roundrobin partitioning. Round robin partitionin is equivalent to the 12.5 style of partitioning. During the upgrade to Adaptive Server release 15.0, all existing partitioned tables are unpartitioned and automatically converted to 1-way round robin partitioned tables.

Large identifiers

There are new limits for the length of object names or identifiers: 255 bytes for regular identifiers, and 253 bytes for delimited identifiers. The new limit applies to most user-defined identifiers including table name, column name, index name and so on. Due to the expanded limits, some system tables (catalogs) and built-in functions have been expanded.

For variables, “@” count as 1 byte, and the allowed name for the variable i 254 bytes.

Computed columns

This chapter describes an enhancement that provides easier data manipulation and faster data access, by allowing you to create computed columns, computed column indexes, and function-based indexes.

- Computed columns – defined by an expression, whether from regular columns in the same row, functions, arithmetic operators,or path names.
- Indexes on computed columns, or computed column indexes – indexes that contain one or more computed columns as index keys.
- Function-based indexes – indexes that contain one or more expressions as index keys.
- Deterministic property – a property assuring that an expression always returns the same results from a specified set of inputs.

Computed columns and function-based indexes similarly allow you to use an expression or a function as the basis for a more complex function.

Differences between computed columns and function-based indexes

Computed columns and function-based indexes differ in some respects:

- A computed column provides both shorthand for an expression and indexability, while a function-based index provides no shorthand; it allows you to index the expression directly.
- A computed column can be either deterministic or nondeterministic, but a function-based index must be deterministic. “Deterministic” means that if the input values in an expression are the same, the return values must also be the same.

Differences between materialized and not materialized computed columns

- Computed columns can be materialized or not materialized. Columns that are materialized are preevaluated and stored in the table when base columns are inserted or updated. The values associated with them are stored in both the data row and the index row. Any subsequent access to a materialized column does not require reevaluation; its preevaluated result is accessed. Once a column is materialized, each access to it returns the same value.
- Columns that are not materialized are also called virtual columns; virtual columns become materialized when they are accessed. If a column is virtual, or not materialized, its result value must be evaluated each time the column is accessed. This means that if the virtual computed column is expression-based on, or calls a nondeterministic expression, it may return different values each time you access it.You may also encounter run-time exceptions, such as domain errors, when you access virtual computed columns.

Scrollable cursors

Adaptive Server Enterprise allows both scrollable and nonscrollable cursors, which can be either semi-sensitive or insensitive.

"Scrollable" means that you can scroll through the cursor result set by fetching any, or many, rows, rather than one row at a time; you can also scan the result set repeatedly. You must use Transact-SQL or JDBC to declare a scrollable cursor, and you must have the query engine provided in Adaptive Server 15.0 or later. A scrollable cursor allows you to set the position of the cursor anywhere in the cursor result set for as long as the cursor is open, by specifying the option first, last, absolute, next, prior, or relative in a fetch statement.

To fetch the last row in a result set, enter:

```
fetch last [from] <cursor_name>
```

Or, to select a specific row in the result set, in this case the 500th row, enter:

```
fetch absolute 500 [from] <cursor_name>
```

"Insensitive" or "semi-sensitive" refers to the extent to which data changes from outside the cursor are visible to the cursor. A cursor can be semi-sensitive but not scrollable.

All scrollable cursors are read-only. All update cursors are nonscrollable.

Unitext support

Adaptive Server version 15.0 includes the variable-length unitext datatype, which can hold up to 1,073,741,823 Unicode characters (2,147,483,646 bytes). You can use unitext anywhere you use the text datatype, with the same semantics. unitext columns are stored in UTF-16 encoding, regardless of the Adaptive Server default character set.

The benefits of unitext include:

- Large Unicode character data. Together with unichar and univarchar datatypes, Adaptive Server provides complete Unicode datatype support, which is best for incremental multilingual applications.
- unitext stores data in UTF-16, which is the native encoding datatype for Windows and Java environments.

For more information about unitext support, see the *System Administration Guide*.

big int support

Adaptive Server version 15.0 includes the exact numeric datatype bigint

This is the range of numbers allowed by the bigint datatype:

Table 1-1: Ranges for bigint datatype

Datatype	Range of signed datatypes
bigint	Whole numbers between -2^{63} and $2^{63} - 1$ (from -9,223,372,036,854,775,808 to +9,223,372,036,854,775,807, inclusive.

Adaptive Server's bigint support also adds the hextobigint, biginttohex, and count_big functions. For more information about these functions, see Chapter 2, "System Changes in Adaptive Server Version 15.0," and the *Reference Manual: Blocks*.

Unsigned int support

These unsigned integer datatypes allow you to extend the range of the positive numbers for the existing integer types without increasing the required storage size. That is, the signed versions of these datatypes extend both in the negative direction and the positive direction (for example, from -32 to +32). However, the unsigned versions extend only in the positive direction. Table 1-2 describes the range of the signed and unsigned versions of these datatypes.

Table 1-2: Ranges for signed and unsigned datatypes

Datatype	Range of signed datatypes	Range of unsigned datatypes
bigint	Whole numbers between -2^{63} and $2^{63} - 1$ (from -9,223,372,036,854,775,808 to +9,223,372,036,854,775,807, inclusive	Whole numbers between 0 and 18,446,744,073,709,551,615
int	Whole numbers between -2^{31} and $2^{31} - 1$ (-2,147,483,648 and 2,147,483,647), inclusive	Whole numbers between 0 and 4,294,967,295
smallint	Whole numbers between -2^{15} and 2^{15} -1 (-32,768 and 32,767), inclusive	Whole numbers between 0 and 65535

Integer identity

Adaptive Server release 15.0 allows you to use the following datatypes as identity values:

- bigint
- int
- numeric
- smallint
- tinyint
- unsigned bigint
- unsigned int
- unsigned smallint

Enhancements to XML services

XML enhancements in 15.0 include XML schema support, Unicode (I18N) support, and for xml clause enhancements.

XML schema support

You can validate XML documents against either a DTD or an XML schema. The DTD or schema can be specified either in the xmlvalidate command or in the document itself.

You can parse, store, and query XML documents with XML schema declarations.

Internationalization (I18N) support

The I18N extensions fall into three categories:

- I18N support in the for xml clause. The columns of the result set you map to XML can contain non-ASCII data.Such data can be represented in the generated SQLX XML document either as plain characters or as numeric character representations (NCRs).
- I18N in xmlparse and xmlvalidate, to store and validate documents containing non-ASCII data.
- I18N in xmlextract and xmltest, to process XML documents and queries containing non-ASCII data.

for xml enhancements

In Transact-SQL, an expression subquery is a parenthesized subquery. It has a single column, the value of which is the expression subquery result, and must return a single row. You can use an expression subquery almost anywhere you can use an expression. For more information about subqueries, see the *Transact-SQL® User's Guide*.

The for xml subqueries feature allows you to use any subquery containing a for xml clause as an expression subquery. For the syntax of for xml subqueries, see the *XML Services Guide*.

Adaptive Server Plug-in enhancements

Adaptive Server release 15.0 adds the following to the Adaptive Server Plug-in to enhance its efficiency and convenience:

- An enterprise view that includes Server Discovery (which enables you to find available servers on the system) and automatic server status.
- The ability to update servers, administrate remote servers, and manage server logs.
- SQL Preview and Job Scheduler integration.
- A graphical query plan viewer.
- The ability to integrate external tools.

Interactive SQL

Interactive SQL allows you to execute SQL statements, build scripts, and display database data to the server. You can run nteractive SQL individually or from the Adaptive Server Plug-in. It has been integrated in the Adaptive Server Plug-in as the standard query tool.

You can use Interactive SQL to:

- Browse the information in a database.
- Test SQL statements that you plan to include in an application.
- Load data into a database and carrying out administrative tasks.

In addition, Interactive SQL can run command files or script files. For example, you can build repeatable scripts to run against a database and then use Interactive SQL to execute these scripts as batches.

User-defined web services

In addition to the Web methods provided by the Adaptive Server Web Services Engine, Web Services enables you to create Web services and execute SQL commands in Adaptive Server Enterprise using either a Web browser or a SOAP client. These user-defined Web services use existing security and auditing control inherent in Adaptive Server Enterprise.

You can create a user-defined Web service with the create service command, which enables you to specify the SQL to be executed, create a first-class object for which permissions can be controlled with the grant command, and control whether the service can be invoked with a Web browser or a SOAP client. The ASE Web Services Engine automatically generates WSDL for user-defined Web services. For details on creating and using user-defined Web services, see the Adaptive Server Enterprise Web Services *User's Guide*.

Very large storage support

Adaptive Server 15.0 greatly extends the allowable number of disk devices and the allowable number of 2K blocks for each device.

In pre-15.0 releases of Adaptive Server, a virtual page is described internally in a 32-bit integer: the first byte holds the device number (vdevno) and the succeeding three bytes describe the page offset within the device in units of 2K bytes (the virtual page number). This architecture limits the number of devices to 256 and the size of each device to 32 gigabytes—for a maximum storage limit of 8 terabytes in the entire server.

With Adaptive Server 15.0, the device number and the page offset are stored in separate 32-bit integers. The new architecture allows you to create up to 2,147,483,647 disk devices, each of which can be as large as 2,147,483,648 2K blocks or 4 terabytes.

Note Because of schema changes to the sysdevices and sysusages system tables, you may need to modify scripts and stored procedures that access these tables. The device identifier must now be obtained from the vdevno columns of sysdevices and sysusages. The high, low, and vstart columns of these tables no longer store the device and virtual page number—they store only the virtual page numbers.

Automatic update statistics

Instead of manually running update statistics at a certain time, you can set update statistics to run automatically at the time that best suits your site and avoid running it at times that hamper your system. The best time for you to run update statistics is based on the feedback from the datachange function. datachange also helps to ensure that you do not unnecessarily run update statistics. You can use these templates to determine the objects, schedules, priority, and datachange thresholds that trigger update statistics, which ensures that critical resources are used only when the query processor generates more efficient plans.

Because it is a resource intensive task, the decision to run update statistics should be based on a specific set of criteria. Some of the key parameters that can help you determine a good time to run update statistics are:

- How much has the data characteristics changed since you last ran update statistics? This is known as the “datachange” parameter.

- Are there sufficient resources available to run update statistics? These include resources such as the number of idle cpu cycles and making sure that critical online activity does not occur during update statistics.

Datachange is a key metric that helps you measure the amount of altered data since you last ran update statistics, and is tracked by the datachange function. Using this metric and the criteria for resource availability, you can automate the process of running update statistics. The Job Scheduler provides the mechanism to automatically run update statistics. Job Scheduler includes a set of customizable templates that determine when update statistics should be run. These inputs include all parameters to update statistics, the datachange threshold values, and the time when to run update statistics. The Job Scheduler runs update statistics at a low priority so it does not affect critical jobs that are running concurrently.

SySAM license management

The Sybase Software Asset Management (SySAM) implementation has changed for this release of Adaptive Server. The changes include:

- Asset management and reporting tools are provided with SySAM version 2.0. These tools allow you to monitor license usage and compliance.
- A single installation method supports all Adaptive Server editions.
- SySAM configuration is no longer optional.
- Flexible SySAM configuration options are provided.
- SySAM licenses are no longer shipped along with order fulfillment. You must obtain license certificates from the Sybase Product Download Center (SPDC).
- SySAM license keys include information about the support plan you purchased. You must update these licenses whenever you renew your support plan.
- Licensing policies are strictly and consistently enforced.

- Adaptive Server can function under grace periods if it is not able to obtain a license. These grace periods allow customers reasonable time to respond to the issues causing license failure. Adaptive Server continues to function normally during the grace period. Adaptive Server features or the server itself will shut down at the end of the grace period if the licensing issues are not resolved.
- You can receive real-time e-mail notifications about licensing events.
- Licenses issued from SPDC include information about the host machine where the licenses will be deployed. These licenses cannot be used on another machine without being reissued from SPDC.

These changes affect the Adaptive Server installation and configuration process. See the SySAM Configuration chapter of the *Configuration Guide* for details on SySAM configuration and deployment options. See the *Adaptive Server Installation Guide* for your platform on pre-installation planning and SySAM installation information.

Plan your SySAM deployment before installing Adaptive Server.

Warning! SySAM provides for grace periods when it encounters licensing problems. When Adaptive Server enters such a grace period, the Adaptive Server error log is updated with this information. Optionally, e-mail notifications can be configured for such events. You must fix the problems causing Adaptive Server to go into grace. While Adaptive Server functions normally during this grace period, it may shutdown or disable the licensed features if the problem causing license failure is not fixed within the grace period.

Query processing metrics (qp metrics)

Query processing (QP) metrics identify and compare empirical metric values in query execution. When a query is executed, it is associated with a set of defined metrics that are the basis for comparison in QP metrics.

The metrics captured include:

- CPU execution time – the time, in milliseconds, it takes to execute the query.

- Elapsed time – the difference in milliseconds between the time the command started and the current time, as taken from the operating system clock.
- Logical IO (LIO) reads – the number od Logical IO reads.
- Physical IO (PIO) reads – the number of Physical IO reads.
- Count – the number of times a query is executed.
- Abort count – the number of times a query is aborted by the resource governor due to a resource limit being exceeded.

Each metric has three values: minimum, maximum, and average. Count and abort count are not included.

Updates to abstract plans

For a description of the new and changed abstract plans, see the *Query Processing Guide*.

showplan changes

The Adaptive Server 15.0 version of showplan better represents the steps performed by the query processor Adaptive Server 15.0 changes the format of the showplan messages to convey the shape of the query plan. Instead of the showplan messages displayed in a vertical format:

```
delete
from authors
where au_lname = "Willis"
and au_fname = "Max"

QUERY PLAN FOR STATEMENT 1 (at line 1).

The type of query is DELETE.
        The update mode is direct.

        FROM TABLE
           authors
        Nested iteration.
```

```
        Using Clustered Index.
        Index : au_names_ix
        Forward scan.
        Positioning by key.
        Keys are:
           au_lname  ASC
           au_fname  ASC
        Using I/O Size 2 Kbytes for index leaf pages.
        With LRU Buffer Replacement Strategy for index leaf pages.
        Using I/O Size 2 Kbytes for data pages.
        With LRU Buffer Replacement Strategy for data pages.
        TO TABLE
           authors
```

The Adaptive Server 15.0 version of showplan displays a series of "pipes" (the "|" symbol) to distinguish each of the steps performed by the operators.

In the following query, there are three operators, EMIT, DELETE, and SCAN, so this query includes three sets of pipes to display this organization:

```
delete
from authors
where au_lname = "Willis"
and au_fname = "Max"

QUERY PLAN FOR STATEMENT 1 (at line 1).
2 operator(s) under root

The type of query is DELETE.
ROOT:EMIT Operator
  |DELETE Operator
  |  The update mode is direct.
  |
  |  |  |SCAN Operator
  |  |  |  FROM TABLE
  |  |  |  authors
  |  |  |  Index : aumind
  |  |  |  Forward Scan.
  |  |  |  Positioning by key.
  |  |  |  Keys are:
  |  |  |       au_lname ASC
  |  |  |       au_fname ASC
  |  |  |  Using I/O Size 8 Kbytes for index leaf pages.
  |  |  |  Using LRU Buffer Replacement Strategy for index leaf pages
  |  |  |  Using I/O Size 8 Kbytes for data pages.
  |  |  |  With LRU Buffer Replacement Strategy for data pages.
  |
  |  TO TABLE
```

```
|   authors
|   Using I/O Size 8 Kbytes for data pages.
```

Note This version of Adaptive Server also includes the ability to display showplan messages in XML.

Secure Socket Layer Uses FIPS 140-2

Secure Socket Layer (SSL) is the standard for securing the transmission of sensitive information – such as credit card numbers, stock trades, and banking transactions – over the Internet. SSL relies on public key and secret key cryptography.

The SSL used in Adaptive Server release 15.0 uses cryptographic modules validated for FIPS 140-2, level 1. The cryptographic modules are Certicom Security Builder GSE for Adaptive Server products running on Windows, Solaris, AIX and HPUX operating systems. For more information, see validation certificate #542, dated June 2, 2005 at NIST website, http://csrc.nist.gov/cryptval/140-1/1401val.htm.

CHAPTER 2

System Changes in Adaptive Server Version 15.0

This chapter describes the system changes introduced in Adaptive Server version

Feature	Page
Changes to configuration parameters	17
Changes to Transact-SQL commands	19
New and changed system procedures	22
Changes to utility programs	23
Changes to databases and system tables	24
New and changed monitoring tables	28
New reserved words	28
New or changed global variables	29

System changes in version 15.0

Changes to configuration parameters

This section describes the new and changed configuration parameters in version 15.0.

New configuration parameters

Table 2-1 summarizes the new configuration parameters.

Table 2-1: New configuration parameters

Function	Description
enable metrics capture	Enables Adaptive Server to capture metrics at the server level.

Function	Description
enable semantic partitioning	Enables semantic (hash-, list-, range-) partitioning of tables and indexes at a licensed site.
enable web services	Enables web services
enable xml	Enables the XML services
max native threads per engine	Defines the maximum number of native threads the server spawns per engine
max partition degree	Configures the amount of dynamic repartitioning Adaptive Server requires, which enables Adaptive Server to use horizontal parallelism
max resource granularity	Sets the maximum percentage of the system's resources a query can use
number of devices	Specifies the number of database devices Adaptive Server can use
number of dump threads	Controls the number of threads that Adaptive Server spawns to perform a memory dump
number of open partitions	Specifies the number of partitions that Adaptive Server can access at one time.
optimization goal	Allows you to configure for three optimization goals, which you can specify at three tiers: server level, session level, and query level
optimization timeout limit	Specifies the amount of time Adaptive Server can spend optimizing a query as a percentage of the total time spent processing the query
rtm thread idle wait period	Defines the time a native thread used by Adaptive Server waits when it has no work to do
sysstatistics flush interval	Determines the length of the interval (in minutes) between flushes of sysstatistics
statement cache size	Increases the server allocation of procedure cache memory and limits the amount of memory from the procedure cache pool used for cached statements. The statement cache feature is enabled server-wide.

Changes to existing configuration parameters

Table 2-2 summarizes changes to existing configuration parameters.

Table 2-2: Changed configuration parameters

Parameter	Change
default network packet size	Previous versions of Adaptive Server used a default network packet size of 512. As of Adaptive Server version 15.0 the default network packet size is 2048.

Changes to Transact-SQL commands

This section describes the new and changed Transact-SQL commands in version 15.0.

New Transact-SQL commands

Table 2-3 summarizes the new Transact-SQL commands.

Table 2-3: New commands

Command	Function
create service	For creating a user-defined Web service.
drop service	For creating a user-defined Web service.
update table statistics	Updatex systabstats statistics for a table or a partition.

Changed Transact-SQL commands

Table 2-4 summarizes the changes to existing Transact-SQL commands.

Table 2-4: Changed commands

Command	Change
alter table	Syntax added to support computed and materialized or non-materialized columns. Adds support for paritions.
create index	Enhanced to allow computed columns to be used as index keys, in the same way as regular columns, and to create function-based indexes. Adds support for paritions.
create table	Syntax added to support computed and materialized or non-materialized columns. Adds support for paritions.
dbcc	Adds support for partitions.
declare cursor	Syntax added for scrollable cursors.
delete statistics	Adds support for partitions.
disk init	The size parameter can be specified in terabytes. Adds the directio parameter, which allows you to configure Adaptive Server to transfer data directly to disk, bypassing the operating system buffer cache
disk reinit	Adds the directio parameter, which allows you to configure Adaptive Server to transfer data directly to disk, bypassing the operating system buffer cache
declare cursor	Syntax added to support semi_sensitive, insensitive, and scrollable cursors.
fetch	fetch_orientation options added to support scrollable cursors: next, prior, first, last, absolute, and relative.
reorg	Adds support for partitions.
select	for xml clauses added to support XML services. Adds support for partitions.
truncate table	Adds support for partitions.

Command	Change
update all statistics	Adds support for partitions.
update statistics	Adds support for partitions.
update partition statistics	Obsoleted

New *set* command options

Table 2-5 summarizes the new options for the set command.

Table 2-5: New set command options

set Option	Description
set delayed_commit	Allows you to determine when log records are written to disk. With the delayed_commit parameter set to true, the log records are asynchronously written to the disk and control is returned to the client without waiting for the IO to complete
set plan optgoal	Sets the optimization goals at the session level.
set plan opttimeoutlimit	Sets the limit the time taken by long-running and complex queries at the session level.
set metrics_capture on/off	Activates QP metrics at the session level.

New Transact-SQL functions

Table 2-6 summarizes the new Transact-SQL functions.

Table 2-6: New Transact-SQL functions

Function	Description
biginttohex	Returns the platform-independent hexadecimal equivalent of the specified integer
count_big	Returns the number of (distinct) non-null values or the number of selected rows as a bigint
datachange	Measures the amount of change in the data distribution since update statistics
data_pages	Returns the number of pages used by the specified table, index, or a specific
hextobigint	Returns the bigint value equivalent of a hexadecimal string
is_quiesced	
partition_id	Returns the partition id of the specified data or index partition name.
partition_name	The explicit name of a new partition, partition_name returns the partition name
reserved_pages	Reports the number of pages reserved to a table, index or a specific partition.
row_count	Returns an estimate of the number of rows in the specified table.
showplan_in_xml	
ssel_message	
tran_dumpable_status	
used_pages	Reports the number of pages used by a table, an index, or a specific partition.
xmlvalidate	Validates XML documents, including those containing non-ASCII characters (I18N). Described in *XML Services*.

Changes to functions

Table 2-7 describes Adaptive Server 15.0 functions that replace obsoleted functions from earler releases.

Table 2-7: Obsoleted functions

Obsoleted function	New function
data_pgs	data_pages
used_pgs	used_pages
reserved_pgs	reserved_pages
rowcnt	row_count
ptn_data_pgs	data_pages

New and changed system procedures

This section describes the new system procedures added and changes made to existing system procedures.

New system procedures

Table 2-8 summarizes the new system procedures.

Table 2-8: New system procedures

System Procedure	Function
sp_helpcomputedcolumn	Reports information on all the computed columns in a specified table
sp_version	Returns the version information of the installation scripts (*installmaster*, *installdbccdb*, and so on) that was last run and whether it was successful.

Changed system procedures

Table 2-9 summarizes the changes made to existing system procedures.

Table 2-9: Changed system procedures

Procedure	Change
sp_checksource	Checks the existence of computed columns source text.
sp_help	Reports information on computed columns, function-based indexes, and partitions.
sp_helpartion	Adds detailed partition information to its output.
sp_helpindex	Reports information on computed column indexes, function-based indexes, and partitions.
sp_helptext	Displays the source text of computed columns, function-based index definitions, and partitions.
sp_hidetext	Hides the text of computed columns, function-based index keys, and partition condition.
sp_modifylogin	Adds option "enable logins during recovery".
sp_webservices	Adds addalias, deploy, dropalias, listudws, listalias, and undeploy options to support user-defined Web Services.
sp_monitorconfig	Supports the number of open partitions configuration parameter.
sp_countmetadata	Supports the number of open partitions configuration parameter.
sp_helpsegment	Prints segment bindings for objects and partitions.
sp_objectsegment	Displays segment information for all partitions for an object.
sp_placeobject	Enables future allocations for a partition from a new segment.
sp_dbcc_faultreport	Creates reports for a specific OPID or fault type.
sp_sysmon	Reports information related to open partitions Metadata Cache Management section of the configuration file.

Changes to utility programs

Table 2-10 summarizes the new utility programs added to Adaptive Server version 12.5.

Table 2-10: Changes to utility programs

Utility	Change
bcp	Adds new parameters --sho-fi and --hide-vcc, to support computed columns and functional indexes. Add new parmeter --maxconn to support for parallel loading into partitioned tables. bcp interface has changed to now allow you to run bcp in and bcp out to and from specific partitions.
dataserver	Specifies the -b *master_database_size* parameter in terabytes.
sqlsrvr	Specifies the -b *master_database_size* parameter in terabytes.
ddlgen	Adds the WS object type for the -T *object_type* parameter to support user-defined Web services.
preupgrade	preupgrade includes options to perform incremental checks for various upgrade checks and is enhanced to run on a single database that is undergoing an upgrade using load database

Changes to databases and system tables

This section describes the changes made to databases and system tables to support version 15.0 features.

New databases

There are no new databases for this release.

New system tables

Table 2-11: New system tables

Table	Description
syspartitions	syspartitions is completely changed from the pre-15.0 version of the table. All columns are new. syspartitions supports both semantic and round-robin partitioning of tables and indexes.
syspartitionkeys	Contains a row for each column in a partition key for each hash-, range-, and list-partitioned table.

Changed system tables

Most system tables are row-locked for Adaptive Server release 15.0. During upgrade, most system tables are upgraded to the datarows-locking mode.

Adaptive Server release 15.0 provides the necessary row-locked catalog infrastructure to support enhanced, multi-user-concurrent data-definition language (DDL) operations. However, this release does not change the catalog locking behavior for DDL operations. Applications that perform heavy multi-user DDL operations (for example, by creating or dropping tables in tempdb, will not see any change in behaviour in this release for catalog blocking, or any increased DDL concurency.

Table 2-12 summarizes the system tables that have been changed in this release.

Table 2-12: Changed system tables

Table	Change
syscolumns	New fields: • computedcol • status3 New columns: • encrtype – Type of encryption • encrlen – Length of encrypted column • encrkeydid – Encryption key id • encrkeydb – Database name containing encryption key • encrdate – Date column was encrypted. New bits in status2 field: • Hex: 0x00000010, Decimal 16 – the column is a computed column. • Hex: 0x00000020, Decimal 32 – the column is a materialized computed column. • Hex: 0x00000040, Decimal 64 – the column is a computed column in a view.
sysconstraints	New internal bit in status field: Hex 0x0100, decimal 265 – indicates a computed column object.
sysdevices	New columns: • vdevno – device identification number • crdate – date device created • resizedate – date size of device changed • status2 – Additional status2 bits.
sysusages	New column: vdevno – device identification number
sysstatistics	New columns: • indid – index ID of the data partition. Always 0. • partitionid – ID of the data partition • ststatus – Internal status bits Unique placement index on id, indid, partitionid, statid, colidarry, formatid, sequence

Table	Change
systabstats	New columns: • partitionid – ID of data or index partition • statmoddate – Date when statistics were last modified on disk. • unusedpgcnt – Number of unused pages. • oampagecnt – Number of OAM pages for each partition.
syspartitions	syspartitions is completely changed from the pre-15.0 version of the table. All columns are new. syspartitions supports both semantic and round-robin partitioning of tables and indexes.
syscomments	New column: partitionid – ID of data or index partition Table enhanced to store the text of computed column or function-based index key expression.
sysindexes	New columns: • partitiontype – partitioning strategy: 1 – range, 2 – hash, 3 – round-robin, 4 – list • conditionid – ID of the partition condition New rows: contains one row for each function-based index or index created on a computed column. One new internal status bit added to the status2 field: Hex 0x8000, decimal 32768 – the index is a function-based index.
syslocks	New columns: • nodeid – Reserved for future use. • partitionid – ID of data or index partition. Reserved for future use. Always 0.
sysobjects	New object in type column: N – partition condition New column: • identburnmax – For an identity column, maximum burned identity value • spacestates – Number of space states being tracked. (Only applies for DOL tables.) • erlchgts – Timestamp when expected row length was last changed. (Only applies to DOL tables.) New row: one row for each computed column and function-based index key object • type field: type “C” added to the type field, when the object is a computed column • status2 field: new bit added to indicate that the table contains one or more function-based indexes.
sysprocedures	Stores a sequence tree for each computed column or function-based index definition, in binary form

For more information about system tables, see the *Reference Manual Volume 4: System Tables*.

Changed status in existing columns

Table 2-13 summarizes the changes in existing system table columns in this release:

Table 2-13: Changes to system table columns

System catalog	Changed column	Datatype changes	Identifier name
sysattributes	object_cinfo char_info	varchar(30) null to varchar(255) null varchar(255) to varchar(768)	Identifier for the object
sysaudits01-sysaudits08	objname	varchar(30) not null to varchar(255) not null	Object name
syscolumns	name	varchar(30) not null to varchar(255) not null	Column name
	remote_name	varchar(30) null to varchar(255) null	Maps local names to remote names
sysconfigures	name	varchar(80) null to varchar(255) null	
sysindexes	name	varchar(30) null to varchar(255) null	Index for the table name
sysjars	jname	varchar(30) null to varchar(255) null	JAR name
sysobjects	name	varchar(30) not null to varchar(255) not null	Object name
sysprocesses	hostname	char(10) not null to varchar(30) null	Host computer name
	program_name	char(16) not null to varchar(30) null	Name of front_end module
	hostprocess	char(8) not null to varchar(30) null	Host process ID number
	cmd	char (16) not null to varchar(30) null	Command or process currently being executed. Evaluation of a conditional statement, such as an if or while loop, returns.
systimeranges	name	varchar(30) not null to varchar(255) not null	Unique name of the time range
systypes	name	varchar(30) to varchar(255)	Datatype name
sysdatabases:	def_remote_loc	varchar(255) null to varchar(349) null.	

New and changed monitoring tables

Table 2-14 describes the new monitoring table.

Table 2-14: New monitoring tables

Monitoring table	Description
monOpenPartitionAcitivity	Provides monitoring information for partitions

Changed monitoring tables

Table 2-14 describes the changed monitoring tables.

Table 2-15: New monitoring tables

Monitoring table	Changes
monEngine	New columns for house keeper GC task
monCachedObject	New columns for partitions
monProcessObject	New columns for partitions

New reserved words

The following are the reserved words added version 15.0 Adaptive Server:

- insensitive – supports scrollable cursors
- xmlextract – supports XML services
- xmlparse – supports XML services
- xmltest – supports XML services

As for all reserved words, the workaround is to use quoted identifiers:

```
set quoted_identifier on
create table "xmlextract"("xmlparse" int)
```

You must change all database names that are new reserved words before you can upgrade from an earlier release of the server. You can change table, view, and column names or use delimited identifiers. Once you upgrade to version 15.0, you cannot use database objects whose names are new reserved words until you modify your procedures, SQL scripts, and applications.

Reference Manual Volume: 1 Building Blocks lists all of the reserved words for version 12.5. Reserved words cannot be used as object names or column names.

New or changed global variables

Table 2-16 summarizes the global variables for the server limits of logins, users, and groups

Table 2-16: New global variables

Name of variable	What it displays	Value
@@rowcount	Enhanced to display the number of rows moved by a cursor, scrollable or nonscrollable. In a nonscrollable cursor, the rows are fetched from the underlying tables to the client. In a scrollable cursor, the rows counted are fetched from the current result set, not from the underlying tables.	The value of the global variable *@@rowcount* is affected by the specified cursor type. The default, non-scrollable cursor moves forward one row at a time; the maximum value is the number of rows in the result set. In a scrollable cursor the value of continues to increment, whatever the direction of the fetch command; there is no maximum value.
@@fetch_status	The status of a fetch command used for a scrollable cursor.	0 – fetch statement successfully executed. -1 – either the fetch statement failed, or the row requested is outside the result set. -2 – value reserved.
@@cursor_rows	The total number of rows in the cursor result set.	0 – No cursors are open, or no rows qualify for the last open cursor. -1 – Semi-sensitive and scrollable, but the scrolling worktable is not yet populated. The number of rows that qualify the cursor is unknown. *n* – The last opened or fetched cursor result set is fully populated; the value returned is the total number of rows in the result set.

CHAPTER 3

New Features in Adaptive Server 12.5.2 and 12.5.3

This chapter describes the system changes introduced in Adaptive Server versions 12.5.2 and 12.5.3.

Feature	Page
Cross platform dump and load	31
Importing statistics for proxy tables	32
Additional support for historical server	32
Additional support for SSL	32
Native XML	33
HP-UX support	33
Enhancements to Real Time Data Services	33
Resource Governor enhancements	34
Page allocation for partitioned DOL tables	34
User connections enhancements	35
dtdValidation support	35
Migration tool enhancements	36
top n functionality	36
New language support	36
Monitor counter concurrency	37
System changes in 12.5.3	37

New features in Adaptive Server 12.5.3

Cross platform dump and load

Adaptive Server Enterprise version 12.5.2 supported the dump and load of databases across platforms with the same endian architecture.

With Adaptive Server version 12.5.3, you can now dump and load databases across platforms with different endian architecture. This means you can perform dump database and load database from either a big endian platform to a little endian platform, or from a little endian platform to a big endian platform.

Platforms supported:

Big-endian	Sun Solaris	IBM AIX	Silicon Graphics	HP-UX on HPPA, HPIA	MAC OS X
Little-endian	Linux IA	Windows	HP True 64*	Sun Solaris x86	

* On True64, the XPDL feature does not work as the backupserver dump is incompatible with other platforms. This issue will be fixed in a future maintenance release.

Importing statistics for proxy tables

In Adaptive Server version 12.5.3, when you perform update statistics on a remote server proxy table, if the relevant table and index statistics are available, the table catalogs are imported to the local systabstats and sysstatistics.

Additional support for historical server

Adaptive Server version 12.5.3 allows you to send monitoring data from Historical Server to a database on a specific Adaptive Server.

Additional support for SSL

Adaptive Server Enterprise version 12.5.3 has added the following changes for Secure Sockets Layer (SSL):

- New cipher suites using the Advanced Encryption Standard (AES) algorithm

- New options to sp_ssladmin to set preferences for cipher suites accepted by Adaptive Server
- New global variable *@@ssl_ciphersuite* to tell the client which cipher suite was chosen by the SSL handshake

These enhancements improve the ability of the System Security Officer to manage SSL, and for client applications to determine the encryption algorithms used on their connection.

Native XML

As of Adaptive Server version 12.5.3, XML services are enabled for the HP-UX 32 bit and HP-UX 64 bit platforms.

HP-UX support

This platform includes the following for the release of Adaptive Server:

- Feedback Optimization (FBO) – On the Itanium platform, Adaptive Server Enterprise supports the FBO server on HP-11.23 Itanium, which improves performance.
- High Availability – On the Itanium platform, the High Availability option is available on HP-UX 11.23 with HP ServiceGuard A.11.15. For more information, see *Using Sybase Failover in A High Availability System* for more information.
- For HP-UX, XML services are available for the HP-UX (on HPPA platform) 32-bit and 64-bit platforms (the XML Management Option needs to be licensed separately).

Enhancements to Real Time Data Services

In Adaptive Server version 12.5.3, use sp_configure to set the number of native threads and the wait time for messaging.

max native threads per engine

Summary information	
Default value	50
Maximum values	1000

Summary information	
Status	Dynamic
Display level	Intermediate
Required role	System Administrator

Use to define the maximum number of native threads the server spawns per engine. When the limit for the native threads is reached, Adaptive Server sessions that require a native thread, sleep until another session releases a native thread.

rtm thread idle wait period

Summary information	
Default value	50 seconds
Maximum value	4026531839 seconds
Status	Dynamic
Display level	Intermediate
Required role	System Administrator

Use to define the time a native thread used by Adaptive Server waits when it has no work to do. When the time set for a native thread is reached, the thread automatically fades out.

Resource Governor enhancements

In Adaptive Server Enterprise version 12.5.3, when the System Administrator modifies a resource limit, all users logged in the session see the change, including the System Administrator.

Page allocation for partitioned DOL tables

In prior versions of Adaptive Server Enterprise, inserts into a partitioned data-only locked table with a placement index may cause wasted space. This waste is more pronounced in tables with a higher number of partitions and also for databases with larger page sizes.

Adaptive Server Enterprise version 12.5.3 avoids wasting extra space by filling up existing allocated extents in the target allocation page even though these extents are assigned to other partitions. The net effect is that we only allocate new extents when there are no free extents in the target allocation page.

To turn off the new space allocation mechanism, activate command line trace flag 646 (-T646).

User connections enhancements

User enhancements done for Adaptive Server Enterprise version 12.5.3 include an updated error message and a correction on reserved sockets.

Number of user connections

Earlier versions of Adaptive Server reserved one third of available sockets to provide for the Enterprise Java Beans (EJB) server, whether or not EJB was configured. Once reserved, these sockets were unavailable for Adaptive Server.

With Adaptive Server version 12.5.3, ESD #2, no sockets are automatically reserved for EJB.

In Adaptive Server version 12.5.3, if the EJB server is enabled and HBC sockets are not available, the message `"hbc_ninit: No sockets available for HBC"` is reported.

User disconnections

An enhancement to error 1608 now displays the host name and login name when a client connection is disconnected abnormally from Adaptive Server and the extended error information provides additional diagnostics. The System Administrator can now analyze and confirm the cause of the disconnect.

dtdValidation support

For Adaptive Server Enterprise version 12.5.3, the dtdValidate option expanded to include:

- dtdValidate='no'

  No validation is performed whether the document has an embedded DTD or references an external DTD. If the document contains an embedded DTD, dtdValidate verifies that it is correct but does not validate.

- dtdValidate='yes'

Validates only if there is an embedded DTD or a reference to an external DTD.

- dtdValidate='strict'

  The document must contain an embedded or reference to an external DTD, and validates against the DTD.

Migration tool enhancements

The sybmigrate tool allows you to migrate across versions of Adaptive Server Enterprise and supports source servers from 12.0 through 12.5.3.

When migrating a database or server from a source server with Adaptive Server Enterprise versions 12.0 and greater but earlier than 12.5.0.1, you will be asked to specify the size and location of a work database on the target server.

top n functionality

top n functionality implements compatibility with MicroSoft SQL server, Adaptive Server Anywhere, and ASIQ.

Use the top n clause to limit the number of rows in the result set to the number of rows specified by the integer. The integer can be any unsigned 32-bit value, in the range 0 through 2^{32}-1 (4GB-1 or 4,294,967,295). Zero indicates “no” rows.

Adaptive Server Enterprise, version 12.5.3 supports the top n clause in outer query select statements, but not in the select list of a subquery. This differs from the MicroSoft SQL server. Any attempt to use the top n clause in a subquery yields a syntax error.

Use top n with select...into statements to limit the number of rows inserted in the target table. This is different from set rowcount, which is ignored during a select...into.

New language support

The Enhanced Full-Text Search Specialty Data Store (EFTS) now includes language support for:

- Traditional Chinese on the Windows and Solaris platforms

- Arabic, Hebrew, Thai, and Russian on the Linux platform

Monitor counter concurrency

Adaptive Server version 12.5.3 tracks the number of applications using the monitor counters. Adaptive Server does not disable or stop data collection by the monitor counters as long as one or more applications are known to be using them. This allows applications such as sp_sysmon and the Monitor Server to operate concurrently.

System changes in 12.5.3

This section describes the system changes in Adaptive Server release 12.5.3.

Changes to commands

Table 3-1 summarizes the new Transact-SQL functions.

Table 3-1: Changes to commands

Function	Description
dbcc monitor	Adds increment, decrement, and reset. increment and decrement increase and decrease usage counts for the monitor counters in the specified group by 1. reset sets the usage count for the monitor counters in the specified group to zero. This turns off collection of monitoring data for this group.
alter table, create table	Adds the id parameter, which sets the identity column to int, smallint, tinyint, or numeric

New Transact-SQL functions

Table 3-2 summarizes the new Transact-SQL functions.

Table 3-2: New Transact-SQL functions

Function	Description
getutcdate	Returns a date and time whose value is in Universal Coordinated Time (UTC). getutcdate is calculated each time a row is inserted or selected

New system procedures

Table 3-3 summarizes the new system procedures.

Table 3-3: New system procedures

System Procedure	Function
sp_post_xpload	checks and rebuilds indexes after a cross-platform database load where the endian types are different

Changed system procedures

Table 3-4 summarizes the changes made to existing system procedures.

Table 3-4: Changed system procedures

Procedure	Change
sp_sysmon	Adds the noclear parameter, which allows you to run multiple concurrent sessions of the sp_sysmon and other monitoring applications, such as Monitor Server, Historical Server or other sp_sysmon sessions
sp_help_resource_limit	Adds the verbose parameter, which lists all resource limits in verbose mode.

New features in Adaptive Server 12.5.2

This section summarizes the version 12.5.2 changes:

- "Setting the statement cache" on page 39
- "XML services" on page 39
- "Real Time Messaging Services" on page 40
- "Web Services" on page 41
- "IPv6 support" on page 41
- "Changes to Enhanced Full-Text Search Capabilities" on page 42

- "Security changes" on page 43
- "Password-protected backups" on page 48
- "Making compressed database dumps" on page 49
- "Linux Adaptive Server for Failover on Veritas 2.1" on page 49
- "Large Memory Support for 32-Bit Linux" on page 50
- "Additions to dbcc complete_xact" on page 50
- "System changes in 12.5.2" on page 50

Setting the statement cache

Adaptive Server 12.5.2 adds the statement cache. The statement cache allows Adaptive Server to store the text of ad hoc SQL statements. Adaptive Server compares a newly received ad hoc SQL statement to cached SQL statements and, if a match is found, uses the plan cached from the initial execution. In this way, Adaptive Server does not have to recompile SQL statements for which it already has a plan.

The statement cache is a server-wide resource, which allocates and consumes memory from the procedure cache memory pool. Set the size of the statement cache dynamically using the statement cache size configuration parameter. The syntax is as follows, where *size_of_cache* is the size, in 2K pages:

```
sp_configure "statement cache size", size_of_cache
```

For example, to set your statement cache to 5000 2K pages, enter:

```
sp_configure "statement cache size", 5000
```

XML services

The changes in XML Services enhance three main areas:

- Extended XML query language for the xmlextract built-in function and xmltest predicate.
  - Support for XPath string functions:
    - toupper
    - tolower

        - normalize-space
        - concat
    - Support for parenthesized XPath expressions
    - XPath standards convergence:
        - Removal of nonstandard subscript expressions
        - Interpretation of the "//" operator
- Enhanced datatype support in the for xml clause of select statements:
    - text, image, binary, and varbinary
    - java.lang.String
- Enhanced datatype support for xmlextract, xmlparse, and xmlrepresentation:
    - xmlextract – can return the result XML fragment as java.lang.String.
    - xmlparse – return the parsed document as a varbinary datatype.
    - xmlrepresentation – can accept varbinary XML documents.
- Enhanced Java-XML sample code:
    - A Java-based ForXmlTree function to map SQL data to a hierarchic XML document
    - A Java-based OpenXml function to extract a SQL table from a hierarchic XML document
    - Java-based SQLX mapping functions for SQL queries that return multiple result sets

For a complete description of these changes, see *XML Services in Adaptive Server Enterprise*.

Real Time Messaging Services

Adaptive Server Enterprise 12.5.2 includes a messaging functionality with the Real Time Data Services (RTDS) option package. This option simplifies the development of application that interact with messaging systems and databases.

RTDS allows you to capture transactions (data changes) in an Adaptive Server database and deliver them as events to external applications in real time. These data changes, or events, are delivered to applications through a Java Messaging Service (JMS) message bus, provided by TIBCO Enterprise™ for JMS. You can send messages to – or retrieve messages from – the message provider using Transact-SQL, provided by Adaptive Server.

For a complete description of the Real Time Messaging Services option, see the *Real Time Data Service User's Guide*.

Web Services

A Web service is a self-contained, modular application that can be accessed through a network connection. Using a Web Service, the user trades performance for increased interoperability enforced by adherence to the Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), HTTP, and Extensible Markup Language (XML) open standards.

Regardless of the programming language in which it has been implemented, a Web service can be accessed from many different platforms and operating systems, thus greatly enhancing the ability for diverse applications to share data. By using many discrete Web services, each handling a limited set of specific tasks, business enterprises can dynamically and incrementally integrate by exposing their existing software in a secure and controlled environment. By providing a standardized means to invoke remote applications, Web services reduce the amount of code required for infrastructure. By enabling users to extract implementation from exposed interfaces (WSDL), Web services provide the tools needed to build a service-oriented architecture (SOA).

For a complete description of the Web Services option, see the *Web Services User's Guide*.

IPv6 support

Adaptive Server 12.5.2 supports IPv6.

Changes to Enhanced Full-Text Search Capabilities

For a complete description of the Enhances Full-Text Search option, see the *Enhanced Full-Text Search Specialty Data Store User's Guide*.

Changes to installation

Adaptive Server version 12.5.2 implements several changes and enhancements to EFTS. Because of these changes the following issues must be kept in mind when upgrading to EFTS for Adaptive Server version 12.5.2:

- Collections created with earlier version of EFTS are not compatible with this version of EFTS.
- Previously existing collections must be dropped before upgrading to the latest version of EFTS.
- Collections must be recreated after the upgrade.

The 12.5.2 version of EFTS is installed into a directory named *EFTS-12_5_2* and does not overwrite the existing *EFTS-12_5* directory. Once the old collections are dropped, you must edit the *SYBASE.csh* and *SYBASE.sh* files to change the value of the SYBASE_FTS environment variable to point to the *EFTS-12_5_2* directory.

New style files directory

The directory containing the Verity style files used by EFTS has changed. EFTS no longer uses the style files contained in *$SYBASE/$SYBASE_FTS/verity/common/style*.

Instead, EFTS now uses the style files contained in *$SYBASE/$SYBASE_FTS/verity/common/styles/txtsvr*.

New features

Adaptive Server version 12.5.2 introduces many new EFTS features.

Permissions for shutdown

Only users with sa_role can shut down EFTS.

New pseudo column *total_docs*

Starting with Adaptive Server version 12.5.2, EFTS returns an integer value for the total number of documents that match a search criteria. This value is returned in a new pseudo column called total_docs.

This is also useful when using the max_docs column, which limits the number of results returned.

index_any clauses support up to 16000 bytes

In earlier versions of Adaptive Server, EFTS did not support index_any clauses greater than 255 bytes. In Adaptive Server version 12.5.2, EFTS supports index_any clauses up to 16000 bytes.

The column definition for the proxy table is not altered, because the definition does not affect the size of the clause that is sent to EFTS.

Primary keys

In Adaptive Server version 12.5.2 and later, primary keys can be used as the text id column. You can create text indexes on tables that contain an identity column or a suitable primary key. Primary keys defined on a single decimal, numeric, int, smallint, or tinyint column are also eligible. Decimal and numeric columns must have a scale of zero

When creating a text index, EFTS first looks for an IDENTITY column in the source table to use as the ID column of the text index. If an IDENTITY column is not found, EFTS looks for a suitable primary key that it can use as the ID column of the text index.

Security changes

For a complete description of Adaptive Server security, see the *System Administration Guide*.

Security features in Adaptive Server are grouped into the following four categories:

- Identification and authentication (I&A)
- Access control
- Accountability
- Encryption technologies

Identification and authentication

Identification and authentication refers to features used by Adaptive Server to positively identify a user. Once a user has been identified, access control mechanisms and individual accountability is enforced.

Adaptive Server 12.5.2 supports the following new and enhanced I&A features:

- Enhanced Kerberos
- LDAP user authentication
- PAM user authentication
- Enhanced login controls

Kerberos

Kerberos is a network authentication protocol that uses secret key cryptography so that a client can prove its identity to a server across a network connection. User credentials are obtained when the user logs in to the operating system, or by executing an authentication program. These credentials are then used by each application to perform authentication. Users only have to log in once, instead of having to log in to each application.

Adaptive Server 12.5.2 supports Kerberos through:

- CyberSafe Kerberos libraries on the following platforms:
  - Sun Solaris 32-bit
  - Sun Solaris 64-bit (new to Adaptive Server version 12.5.2)
  - Windows
  - AIX 32-bit
- MIT Kerberos libraries version 1.3.1 on the following platforms (new to Adaptive Server version 12.5.2):
  - Sun Solaris 32-bit
  - Sun Solaris 64-bit
  - Linux 32-bit
- Native libraries on the following platforms (new to Adaptive Server version 12.5.2):
  - Sun Solaris 32-bit

- Sun Solaris 64-bit
- Linux 32-bit

Note To enable Kerberos security options, you must have ASE_SECDIR, the "Security and directory services" package.

LDAP user authentication

LDAP externalizes authentication. When you are using LDAP, authentication decisions are based on whether Adaptive Server can successfully bind to a specified LDAP server on behalf of the user. To bind to an LDAP server, Adaptive Server uses a distinguished name (DN) extracted from the specified LDAP URL.

Note When LDAP is enabled, password management is delegated to the LDAP service providers.

As of Adaptive Server version 12.5.2, LDAP-authenticated users must already exist as valid logins in Adaptive Server. To create new Adaptive Server logins for LDAP-authenticated users automatically, issue:

```
sp_maplogin, LDAP, NULL, "create login"
```

Alternatively, LDAP-authenticated users can be mapped to existing Adaptive Server users. For example:

```
sp_maplogin NULL, "externuser", "aseuser"
```

Pluggable Authentication Module (PAM) support

Adaptive Server version 12.5.2 introduces Pluggable Authentication Modules (PAM) support, which allows multiple authentication service modules to be stacked and made available without modifying the applications that require the authentication.

PAM integrates Adaptive Server more closely with Sun and Linux operating systems and simplifies the management and administration of user accounts and authentication mechanisms. PAM reduces the total cost of ownership through this closer integration. An additional benefit is that users can customize or write their own authentication and authorization modules.

Note PAM support is currently available on Linux and on Solaris platforms. For more information on PAM user authentication, see your operating system documentation.

Enhanced login controls

Adaptive Server version 12.5.2 introduces several new ways to control authentication:

- authenticate with – the authentication mechanism is defined when you log in. enable pam user auth and enable ldap user auth enable PAM and LDAP respectively. You can also force a login to use a specific authentication process by using the new options to sp_modifylogin and sp_addlogin. For more information, see the Reference Manual: Procedures.
- sp_maplogin – allows you to map external users to Adaptive Server logins.
- sp_helpmaplogin – displays mapping information
- @@authmech – specifies the current authentication mechanism.

Access control

The access control changes to Adaptive Server version 12.5.2 are discussed in this section.

Improved granularity for set proxy

In earlier versions of Adaptive Server, set proxy allowed you to switch your server user identity to any other server login, but it did not allow you to limit use of set proxy based on the target login roles. When you granted set proxy to a user, this user could become any other server user.

As of Adaptive Server version 12.5.2, you can grant set proxy...restricted role, which allows you to restrict which roles cannot be acquired when switching identities.

Grant revoke on administration commands

Adaptive Server version 12.5.2 allows you to grant and revoke permissions for users, roles, and groups for the update statistics, delete statistics, and truncate table commands. Table owners can also provide permissions through an implicit grant by adding update statistics, delete statistics, and truncate table to a stored procedure and then granting execute permissions on that procedure to a user or role.

You cannot grant or revoke permissions for update statistics at the column level. You must have the sso_role to run update statistics or delete statistics on sysroles, syssrvroles, and sysloginroles security tables.

By default, users with the sa_role have permission to run update statistics and delete statistics on system tables other than sysroles, syssrvroles and sysloginroles, and can transfer this privilege to other users.

Restricted permission on system catalogs

Adaptive Server version 12.5.2 adds the default permissions parameter to the grant and revoke commands, which allows you to grant and revoke the default permissions for the system tables listed below. The partial syntax is:

```
grant default permissions on system tables
revoke default permissions on system tables
```

where default permissions on system tables specifies that you grant or revoke the default permissions for the following system tables when you issue it from any database:

sysalternates	sysindexes	sysprocedures	systabstats
sysattributes	sysjars	sysprotects	systhresholds
syscolumns	syskeys	sysqueryplans	systypes
syscomments	syslogs	sysreferences	sysusermessages
sysconstraints	sysobjects	syssegments	sysusers
sysdepends	syspartitions	sysstatistics	sysxtypes

default permissions on system tables also makes the following changes:

- Revokes sysobjects(audflags) permissions from public
- Grants permissions for sysobjects to sso_role

Accountability

audit_event_name is a new function that returns a description of an audit event.

Encryption

Adaptive Server uses encryption technology to secure communications with clients. Beginning with version 12.5.2, Adaptive Server uses FIPS-certified algorithms for SSL encryption. Adaptive Server version 12.5.2 can also protect backups using passwords.

FIPS certified algorithms for SSL encryption

SSL is the standard for securing the transmission of sensitive information, such as credit card numbers, stock trades, and banking transactions over the Internet. It relies on public key cryptography.

SSL implementation now uses FIPS 140-2 certified algorithms. These algorithms are available through:

- FIPS-140-2 Certified Crypto Module on Windows (Certicom's SB/GSE – NIST Certificate #316 with validation dates of 05/13/2003 and 06/30/2003)
- FIPS certified algorithms on Solaris 32- and 64-bit platforms through Certicom's SB Crypto-C.

Password-protected backups

You can protect your database dump from unauthorized loads using the password parameter of the dump database command. If you include the password parameter when you make a database dump, you must also include this password when you load the database.

The partial syntax for the password-protected dump database and load database commands are:

dump database *database_name* to *file_name* [with passwd = *password*]

load database *database_name* from *file_name* [with passwd = *password*]

where:

- *database_name* – is the name of the database that is being dump or loaded.
- *file_name* – is the name of the dump file.
- *password* – is the password you provide to protect the dump file from unauthorized users.

Making compressed database dumps

The compression parameter of the dump command allows you to reduce your space requirements for your archived databases. Earlier versions of dump database using the compression API allowed you only to compress your database dumps to a local file. With Adaptive Server 12.5.2, the compression parameter enables you to compress your dumps to a remote machine.

You need not include the compression level when you load the database dump. However, you can issue load with listonly=full to determine the compression level at which the dump was made.

The partial syntax for dump database is:

> dump database *database_name* to *file_name* [with compression = *compress_level*]

where:

- *database_name* – is the name of the database from which you are copying data. The database name can be specified as a literal, a local variable, or a stored procedure parameter.
- *file_name* – is the name of the dump file. The name cannot exceed 17 characters, and must conform to operating system conventions for file names.
- *compress_level* – is a number between 1 and 9, with 9 providing the highest level of compression. There is no default compression level; if you do not specify a *compress_level*, Adaptive Server does not compress the dump.

For example, the following dumps the pubs2 database to the remote machine called “remotemachine” and uses a compression level of 4:

```
dump database pubs2 to "/Syb_backup/mydb.db" at remotemachine
with compression = "4"
```

Linux Adaptive Server for Failover on Veritas 2.1

Adaptive Server release 12.5.2 supports Veritas 2.1 on RedHat Enterprise Linux 2.1.

Large Memory Support for 32-Bit Linux

Adaptive Server large memory support on 32-bit Enterprise Linux operating systems increases the amount of available memory in Adaptive Server from 2.7GB to as much as 64GB. Increasing the amount of memory available to Adaptive Server improves performance by significantly reducing the number of times the server must access the disk.

Additions to dbcc complete_xact

Adaptive Server 12.5.2 includes the 1pc parameter to the dbcc complete_xact command. 1pc heuristically completes a transaction that was subject to a one-phase commit protocol optimization—instead of the regular two-phase commit protocol—by the external transaction manager that was coordinating its completion. Heuristically committing such a transaction requires that the transaction is in a "done" state (as reported by sp_transactions).

The partial syntax for dbcc complete_xact is:

```
dbcc complete_xact("<xid>", "commit", "1pc")
```

System changes in 12.5.2

This section describes the system changes in Adaptive Server release 12.5.2.

XML language changes

This section describes the XML language changes in Adaptive Server release 12.5.2.

New XML query language

Table 3-5 summarizes the new commands for the XML query language.

Table 3-5: New commands

Function	Description
toupper	Returns the argument values in uppercase.
tolower	Returns the argument values in lowercase.
normalize-space	Makes two changes when it returns its argument value: • Removes leading and trailing white-space characters. • Replaces all substrings of two or more white-space characters that are not leading characters with a single white-space character.
concat	Returns the string concatenation of the argument values. It has zero or more parameters.

Changes to the XML query language

Table 3-6 summarizes the changes to the XML query language.

Table 3-6: Changes to commands

Function	Description
xmlextract	The returns clause of xmlextract now supports java.lang.String.
xmlparse	xmlparse now includes an optional returns clause that lets you specify the datatype of the returned parsed XML document.
xmlrepresentation	In addition to the image datatype, xmlrepresentation now supports binary and varbinary datatypes for the XML document operand.

New XML functions

Table 3-7 summarizes the new XML functions.

Table 3-7: New Transact-SQL functions

Function	Description
ForXmlTree	Maps a set of SQL tables or result sets to a tree-structured XML document.
OpenXml	Extracts repeating data from a tree-structured XML document to a SQL table.
forxmlmultiplej	Maps result sets of a SQL query, that can contain multiple result sets, to an XML document.

Changes to commands

Table 3-8 summarizes the new Transact-SQL functions.

Table 3-8: Changes to commands

Function	Description
dbcc	• prsqlcache parameter allows you to print summaries of cached SQL statements, and the purgesqlcache parameter purges all statement cache entries, except those that are currently in use. • dbcc checkcatalog performs sysindexes consistency checking and adds the fix parameters to fix any errors, if any. • complete_xact adds 1pc, which heuristically completes a transaction that was subject to a one-phase commit protocol optimization—instead of the regular two-phase commit protocol—by the external transaction manager that was coordinating its completion. • Adds the stackused parameter, which reports the maximum amount of stack memory used since the server first started.
update statistics, delete statistics	Allow you to grant and revoke permissions for users, roles, and groups.
grant, revoke	Add the default permissions parameter, which allows you to grant and revoke the default permissions for some system tables.
dump, load database	Include a password parameter, allows you to password-protect your database dumps.
union	The maximum number of subqueries within a single side of a union has increased from 16 to 50.
kill	Adds the statusonly parameter, which reports on the progress of a server process ID (spid) in rollback status. It does not terminate the spid.
select *	Descriptions for queries using select * are fully expanded to describe the columns selected.
grant dbcc, set proxy	Issue the following warning when they are issued while set fipsflagger option is enabled.
update, delete	Queries that use update and delete on views with a union all clause are always resolved using worktables in tempdb.
create procedure (SQLJ)	Allows you to define a default value so that you can execute the SQLJ procedure without providing a parameter value.

New and changed *set* command options

Table 3-9 summarizes the new and changed options for the set command.

Table 3-9: New and changed set command options

set option	Description
statement cache	enables or disables the statement cache at the session level. If the statement cache is configured at the server level, by default it is enabled at the session level
set proxy...restricted role	Allows you to restrict which roles cannot be acquired when switching identities

New Transact-SQL functions

Table 3-10 summarizes the new Transact-SQL functions.

Table 3-10: New Transact-SQL functions

Function	Description
audit_event_name	Returns a description of an audit event
audit_event_name	Returns a description of an audit event.

New system procedures

Table 3-11 summarizes the new system procedures.

Table 3-11: New system procedures

System Procedure	Function
sp_webservices	Creates and manages the proxy tables used in the Consumer component of Web Services
sp_maplogin	Allows you to map external users to Adaptive Server logins
sp_helpmaplogin	Displays mapping information
sp_monitor	Adds the following parameters: connection, (cpu | diskio | elapsed time), event, (summary | detail) , enable, disable, statement, (cpu | diskio | elapsed time), help
sp_dropalias	Adds the force parameter, which allows you to drop an alias even if it owns database objects.
sp_audit	Allows you to audit extended stored procedures.
sp_helptext	Truncates trailing spaces when displaying the source text from syscomments.
sp_ldapadmin	Creates an LDAP URL search string, lists an LDAP URL search string, or verifies an LDAP URL search string or login

Changed system procedures

Table 3-12 summarizes the changes made to existing system procedures.

Table 3-12: Changed system procedures

Procedure	Change
sp_ldapadmin	Adds: • set_access_acct – allows users to specify an account that Adaptive Server can use for administrative searches • set_dn_lookup_url – specifies an alternative authentication algorithm
sp_helpconfig	Adds support for large memory support for 32-bit Linux.
sp_sysmon	Displays performance information about the secondary data cache

Changes to utility programs

Table 3-13 summarizes the new utility programs added to Adaptive Server version 12.5.

Table 3-13: Changes to utility programs

Utility	Change
optdiag	Displays the statistic, sampling percent last used, which indicates that statistics are gathered with a user-specified sampling percent

New configuration parameters

This section discusses new configuration parameters for Adaptive Server version 12.5.2.

Table 3-14: New configuration parameters

Configuration parameter	Description
extended cache size	Specifies the size of the global secondary data cache for 32-bit Enterprise Linux operating systems.
histogram tuning factor	Controls the number of steps Adaptive Server analyzes per histogram for update statistics, update index statistics, update all statistics, and create index
number of dump threads	Controls the number of threads that Adaptive Server spawns to perform a memory dump. Using the appropriate value for number of dump threads can reduce the amount of time the engines are halted during the memory dump

New global variables

Table 3-15 summarizes the global variables for the server limits of logins, users, and groups

Table 3-15: New global variables

Name of variable	What it displays
@@authmech	Specifies the current authentication mechanism
@@monitors_active	Reduces the number of messages displayed by sp_sysmon

CHAPTER 4

New Features in Adaptive Server Version 12.5.1

This chapter describes the new features and system changes introduced with Sybase Adaptive Server Enterprise version 12.5.1.

Feature	Page
Dynamic data cache	58
Automatic database expansion	58
Improvements to the Adaptive Server plug-in for Sybase Central	60
Dynamic listeners	61
Globalization enhancements	62
SQL derived tables	62
Date and time datatypes	62
XML mapping	63
Native XML processor	64
LDAP user authentication	66
HA agent on VCS 3.5	67
Fast recovery	68
Database mount and unmount	68
Job Scheduler	69
New cache wizard section for sp_sysmon	70
Web Services for Adaptive Server	71
Enhanced compatibility with Microsoft SQL extensions	71
Adaptive Server 12.5.1 backward compatibility	72
System changes in version 12.5.1	72

Dynamic data cache

Adaptive Server 12.5.1 allows you to dynamically alter the data cache without having to restart the server, as was required in earlier versions. Using dynamic sp_cacheconfig actions, you release memory so that it can be reallocated immediately.

Some of the actions you can perform dynamically with sp_cacheconfig are:

- Adding a new cache
- Adding memory to an existing cache
- Deleting a cache, either log or mix
- Changing a cache type

If Adaptive Server cannot allocate all the memory requested, it allocates the available memory. The additional memory is allocated when you restart Adaptive Server.

Reducing cache size is not a dynamic action. Rather than using sp_cacheconfig to reduce cache size as a static action, remove the cache and recreate it with the correct size.

For more information, see the *System Administration Guide*.

Automatic database expansion

Adaptive Server Enterprise version 12.5.1 provides automatic expansion of databases and devices. Databases can be configured to expand automatically when they run out of space.

The automatic database expansion stored procedure sp_dbextend allows you to install thresholds that identify which devices have room and then appropriately alter the database—and the segment where the threshold was fired—on these devices.

sp_dbextend is the user interface for the automatic expansion process. DBAs use it to:

- Define site-specific rules for individual segments and devices
- Change user-specified policies and system-supplied default behaviours
- Examine the current state of the rules

DBAs can also use this interface to execute trial simulations of the database expansion machinery, and if necessary, to manually perform a database expansion without waiting for the automatic expansion process to be scheduled.

After you set up the database for automatic expansion, when a database grows to its free space threshold, internal mechanisms fire, increasing the size of the database by the amount of space specified by expansion policies. The automatic expansion process measures the amount of room left on all devices bound to the database. If there is sufficient room on the device, the database continues to grow. If any devices are configured for expansion, those devices are expanded next, followed by expansion of the database on those devices. The automatic expansion process runs as a background task and generates informational messages in the server's error log about its progress.

The automatic expansion process requires new installation procedures. You install it using the *installdbextend* script, which loads new rows into master.dbo.sysattributes—rows that describe defaults for autoexpansion in a database or in a device. You can install the automatic expansion process in one or more databases or one or more devices.

You can configure automatic expansion to run with server-wide default expansion policies, or you can customize it for individual segments in specified databases.You can install thresholds on key segments on which tables with critical data reside, allowing you to have a fine degree of control over how Adaptive Server meets the data space requirements for different kinds of tables. If your site has key tables with large volumes of inserts, you can bind these tables to specific segments, with site-specific rules of extending that segment. This enables you to avoid outages that can occur in a production environment due to large loads to such key tables.

Automatic expansion procedures allow you to simulate the expansion processes based on your chosen policies. This simulation allows you to make sure the expansion processes work without jeopardizing your production environment.

Note The automatic expansion procedure does not create new devices; it only alters the size of the database and segment on existing devices to which the segment currently maps.

For more information, see Chapter 30, "Automatic Database Expansion," in the *System Administration Guide*.

Improvements to the Adaptive Server plug-in for Sybase Central

The Adaptive Server plug-in for Sybase Central has been enhanced for 12.5.1. The Adaptive Server plug-in, along with Sybase IQ and Adaptive Server Anywhere, now runs on Sybase Central version 4.1. This version of Sybase Central provides better performance, requires less memory, and improves the user interface.

The 12.5.1 Adaptive Server plug-in:

- Lets you view and export data from a table or a view using the new “Data” panel
- Lets you view SQL code of compiled objects—stored procedures, views, rules, and so on—using the new “Code” panel
- Lets you log SQL commands in a window or file so that a record exists of actions by the plug-in
- Provides robust interfaces file support
- Provides an enhanced table editor to support column properties and constraint bindings
- Lets you manage multiple tempdbs
- Provides Quiesce and Mount/Unmount Database wizards
- Provides an enhanced Proxy Table wizard for managing proxy tables on data sources other than Adaptive Server
- Lets you manage the Job Scheduler
- Lets you shut down Adaptive Server
- Lets you resize disks
- Lets you use the Monitor Server graphics interface on Windows platforms

The Adaptive Server plug-in for 12.5.1 is backward compatible to Adaptive Server version 12.0. Sybase recommends that users managing pre-12.5.1 servers upgrade Sybase Central to version 4.1 and the Adaptive Server plug-in to 12.5.1.

Dynamic listeners

This version of Adaptive Server adds functionality to sp_listener that allows you to manage network listeners. sp_listener allows you to:

- Start additional listeners (the maximum number of listeners is 32)
- Stop listeners
- Suspend listeners
- Resume suspended listeners

The syntax for sp_listener is:

sp_listener "*command*", "*server_name*", *engine* | *remaining*

Or:

sp_listener "*command*", "*[protocol:]machine:port*", *engine* | *remaining*

Where:

- *command* is start, stop, suspend, resume, or status.
- *server_name* is the name of Adaptive Server.
- *engine* specifies the number of the engine affected by this command (this parameter is ignored by Windows NT). *engine* can be a single-engine number in quotes ("2"), a list ("3,5,6"), a range ("2-5"), or mix ("2,3-5,7").
- remaining specifies that the command is to take effect on all engines on which it can be meaningfully applied (that is, where the listener is in a state in which the command can take effect).
- *protocol* is the protocol used (tcp, tli, ssltcp, ssltli, winsock, sslnlwnsck, or sslwinsock).
- *machine:port* is the machine name and port number to which the listener listens (as specified in the interfaces file).

The number of listener ports is determined by the dynamic configuration parameter number of network listeners. The default value of this parameter is 1. The semantics for sp_listener is atomic: if a command cannot be completed successfully, it is aborted.

For more information, see the *Performance and Tuning Guide*.

Globalization enhancements

Adaptive Server 12.5.1 supports the UTF-8 sort order and the Unicode parser.

UTF-8 sort order

In versions of Adaptive Server earlier than 12.5.1, when using UTF-8, the two sort order choices were binary and no-case (ASCII-only). Adaptive Server version 12.5.1 provides the ability to perform nonbinary sort orders in UTF-8. As such, all sort orders available for the unichar and univarchar datatypes can be used for char or varchar data when the default character set is set to UTF-8.

Unicode parser

In versions of Adaptive Server earlier than 12.5.1, you had to set the server's default character set to UTF-8 to use the unichar and univarchar datatypes.

This restriction has been removed in version 12.5.1. You can now use unichar and univarchar datatypes with any default character set without first having to configure your default character set to UTF-8.

By incorporating syntax from the new SQL2002 standard, you can now use escape sequences to specify any Unicode character in a string literal. If a string literal (whether or not it makes use of the new syntax) contains characters that cannot be represented in the server's default character set, the literal is promoted from the varchar datatype to the univarchar datatype.

See Chapter 7, "Configuring Character Sets, Sort Orders, and Languages" in the *System Administration Guide* for more information.

SQL derived tables

A SQL derived table is created with a nested select statement, as in the following example:

```
select * from (select * from table_1) derived_table_1
```

An equivalent group of SQL statements that uses a view instead of a SQL derived table requires both create and drop statements:

```
create view view_1 as select * from table_1
select * from view_1
drop view view_1
```

The group of SQL statements using a view is more complicated, requiring three statements. Also, the benefits of creating a view for only one query are offset by the overhead of making a system catalog entry. The SQL derived tables feature eliminates this overhead by enabling queries to spontaneously create non-persistent tables without needing to drop the tables or make insertions into the system catalog. This makes a SQL derived table more desirable than a group of SQL statements using a view for ad hoc queries. For repeated queries, a SQL derived table used multiple times performs comparably to a SQL statement using a view with a cached definition. A SQL derived table differs from a temporary table in that a SQL derived table exists only for the duration of the query, while a temporary table exists until the server is brought down.

A system catalog entry occurs for a view that uses a SQL derived table:

```
create view view_1 as
select column_1 from
(select column_1 from table_1) derived_table_1
```

This is likewise true for a stored procedure that uses a SQL derived table:

```
create proc sp_foo @name varchar(40) as
select column_1 from
(select column_1 from table_1) derived_table_1
```

Date and time datatypes

Adaptive Server has various ways to identify date and time. Before version 12.5.1, only datetime and smalldatetime were available. As of version 12.5.1, date and time have been added as separate datatypes.

XML mapping

The for xml clause in select statements and the forxmlj function map SQL result sets to SQLX-XML documents, using the SQLX-XML format defined by the ANSI SQLX standard. For example:

```
                        1> select name, type from systypes
                        2> where name like "%var%"
                        3> for xml
                        4> go
------------------------------
<resultset xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
       <row>
               <name>nvarchar</name>
       </row>
       <row>
               <name>univarchar</name>
       </row>
       <row>
               <name>varbinary</name>
       </row>
       <row>
               <name>varchar</name>
               <type>39</type>
       </row>
</resultset>
```

For more information, see *XML Services in Adaptive Server Enterprise*.

Native XML processor

The native XML processor supports SQL extensions that perform XML query functions on XML documents. The native XML processor can query both stored XML documents and XML documents mapped from SQL by the for xml clause or forxmlj function.

Integration of the native XML processor in Adaptive Server 12.5.1 provides remarkable performance improvements over the Java-based XQL processor of earlier versions of Adaptive Server.

The native XML processor supports standard XML documents and standard XPath queries, which are a subset of the new XQuery language. *XML Services in Adaptive Server* describes the subsets of those standards that the native XML processor supports.

The SQL extensions supported by the native XML processor are:

- xmlextract – a built-in function that applies an XML query expression to an XML document and returns the select result.

- xmltest – a SQL predicate that applies an XML query expression to an XML document and returns the Boolean result.
- xmlparse – a built-in function that parses an XML document for more efficient processing.
- xmlrepresentation – a built-in function that determines whether a given image column contains a parsed XML document.

For more information, see *XML Services in Adaptive Server Enterprise*.

Native XML query

The native XML query integrates the 12.5 XML/XPATH capabilities into Adaptive Server. The early prototypes have already shown remarkable performance improvements over the 12.5.0. Java processor. Completion of this project will enable Sybase to

- Capitalize on such performance improvements
- Remove the dependency on Java for using XML
- Integrate the SQL and XML language levels

The XML query functions support the XML 1.0 standard for XML documents and the XPath 1.0 standard for XML queries. The book *XML Services in Adaptive Server* describes the subsets of those standards that "XML Services in Adaptive Server" supports.

Table 4-1 describes the SQL extensions for accessing and processing XML documents in SQL statements.

Table 4-1: XML query functions

Function	Description
xmlextract	A built-in function that applies an XML query expression to an XML document and returns the selected result.
xmltest	A SQL predicate that applies an XML query expression to an XML document and returns the Boolean result.
xmlparse	A built-in function that parses an XML document for more efficient processing.
xmlrepresentation	A built-in function that determines whether a given image column contains a parsed XML document.

LDAP user authentication

LDAP is an industry standard for accessing directory services over a network. With version 12.5.1, Adaptive Server extends its LDAP support to include storage of user information.

With LDAP services enabled:

- Adaptive Server authenticates clients with data from an LDAP server.

  Users authenticate with passwords stored on an LDAP server rather than in the syslogins catalog. The LDAP server provides a centralized location for login accounts—both names and passwords.

- Adaptive Server servers share user login data stored on the LDAP server.

  Information formerly stored in syslogins is now managed and stored on an LDAP server. It is cached locally to preserve referential integrity and for other, database-specific uses.

- With LDAP enabled, users have a single login and password throughout the enterprise.

Adaptive Server support for LDAP requires an ASE_DIRS license. To set up accounts, you must:

- Add user login accounts to the LDAP server.
- Construct an LDAP URL search string for user authentication to Adaptive Server using sp_ldapadmin.
- Set the Adaptive Server configuration parameter enable ldap user auth to authorize the use of LDAP.

After a user account is added to the LDAP server, Adaptive Server can modify local characteristics of that account. A System Administrator or a System Security Officer can add a row in syslogins using sp_addlogin to set login-specific values—such as a default database or the granting of roles.

To aid migration of user accounts to the LDAP server, enable ldap user auth provides a setting that allows authentication through either the LDAP server or syslogins.

For more information, see the *System Administration Guide*.

HA agent on VCS 3.5

Adaptive Server version 12.5.1 introduces a new HA agent that adds support for active-active high availability on Veritas Cluster 3.5.

The new agent has these components:

- Resource definition file
- Agent binary with monitor function
- Agent scripts
- Agent installation tool

The new HA agent delivers several benefits:

- It provides a complete, independent solution and requires only an Adaptive Server HA license. Enterprise Agent for Sybase, developed by Veritas, is not necessary.
- It is easily installed using the included installation tool.
- It eliminates the need to overwrite Enterprise Agent scripts for Sybase products.
- It is smoothly and tightly integrated with Adaptive Server.

For more information, see *Using Sybase Failover in a High Availability System*.

Fast recovery

During a server restart after a planned or unplanned shutdown, or during HA failover, a significant amount of time can be spent on database recovery. The fast recovery feature enables databases to be available earlier and minimizes database downtime. It can aid in recovery from planned shutdowns, unplanned crashes, and HA failover.

Fast recovery enables the following changes to Adaptive Server:

- Brings engines online earlier after all system databases are recovered.
- Enhances performance of database recovery.
- Recovers multiple databases in parallel by making use of available server resources. This is controlled by the configuration parameter max concurrently recovered db.
- Provide for multiple checkpoint tasks at runtime that can run concurrently to minimize the work that needs to be done at recovery time. This is controlled by the configuration parameter number of checkpoint tasks.
- New global variable @@recovery_state to determine if Adaptive Server is in recovery.
- New option strict | relax to the stored procedure sp_dbrecovery_order.

For more information see "Fast Recovery" in the *System Administration Guide*.

Database mount and unmount

The purpose of the mount and unmount commands is to make it possible to transport databases from one source Adaptive Server to another destination Adaptive Server.

When you copy a database you must use a command outside of Adaptive Server, such as UNIX dd or ftp, to create a byte-for-byte copy of all pages in a set of one or more databases.

When you move a set of databases from a source Adaptive Server to a destination Adaptive Server, you are physically moving the underlying devices.

For more information, see Chapter 22, "Database Mount and Unmount," in the *System Administration Guide*.

Job Scheduler

Job Scheduler eases the management of Adaptive Server by providing the ability to define and schedule database administration tasks. With Job Scheduler, jobs that normally require interaction from a database administrator can be scheduled to run unattended at the appropriate times, freeing the database administrator to attend to other issues.

Job Scheduler allows you to create and schedule jobs, and also share jobs and schedules. One database administrator can create a job and other database administrators can then schedule and run that job on another server. Jobs can be created from scratch, command line or GUI, loaded from a SQL batch file, or generated from a predefined template.

Job Scheduler captures the results and output of jobs, and records that information in log tables. Job Scheduler keeps a history of scheduled jobs. However, to keep a limit on the size of the history table, Job Scheduler is self-monitoring and removes outdated, unnecessary history records.

Job Scheduler is comprised of the following components:

- An internal Adaptive Server task
- An external process called the JS Agent
- The *sybmgmtdb* database and stored procedures
- The graphical user interface
- Predefined templates from which the database administrator may create and schedule useful, time-saving jobs

The internal ASE task determines when scheduled jobs should run and creates a historical record of jobs that are run. It starts the JS Agent process and feeds JS Agent the necessary information to retrieve job information and run the job on the specified ASE.

The JS Agent retrieves the job information from Job Scheduler's own database, called *sybmgmtdb*. Then it logs in to the target ASE and issues the job commands. When the job completes, JS Agent logs any result or output to the log tables in the *sybmgmtdb* database.

All the job, schedule, and scheduled job information, and data needed by the JS task for internal processing is stored in the *sybmgmtdb* database. Most access to data in the *sybmgmtdb* database is via stored procedures. The stored procedures make the data available to the GUI, the JS Agent and the command-line interface. Only the JS task accesses data directly from the *sybmgmtdb* database.

The GUI assists the user in creating and scheduling jobs, viewing job status and job history and in controlling jobs. The GUI also provides an administration feature to turn on and off the ASE internal task and therefore the ability of Job Scheduler to process and execute scheduled jobs.

Templates are an important tool in defining tasks for self-management of the database, such as database backups, reorganization rebuilds, modification of configuration parameters, and statistics updates and monitoring. They are implemented as batch Transact-SQL commands for which parameter values can be provided. Database administrators can use templates to generate jobs, which may then be scheduled to run at desired times.

For more information, see the *Job Scheduler User's Guide for Adaptive Server Enterprise*.

New cache wizard section for *sp_sysmon*

sp_sysmon for Adaptive Server 12.5.1 has been enhanced. A new cache wizard parameter aids in the monitoring and configuring of data caches for enhanced performance. The cache wizard allows you to identify:

- "Hot" objects (objects that are often accessed). The output is ranked by the number of logical reads in a named cache or default data cache.
- The usage of the cache and buffer pools.
- The percentage of hits at a cache, buffer pool, and object level.
- The effectiveness of large I/O.
- The effectiveness of Asynchronous Prefetch (APF).
- The cache occupancy by the various objects.

The cache wizard section appears in the sp_sysmon output only when you include the cache wizard parameter. You can include two parameters with cache wizard, topN and filter. topN ranks the objects while printing the report and filter prints cache names containing this pattern.

For more information about the cache wizard, see *Performance and Tuning: Monitoring and Analyzing*.

Web Services for Adaptive Server

A Web service is a self-contained, modular application that can be accessed and used over a network connection using the SOAP, WSDL, and XML open standards. Adaptive Server Enterprise Web Services enables Adaptive Server Enterprise to both provide and use Web services.

Adaptive Server Enterprise Web Services consists of a Producer component, which enables client applications to access SQL and stored procedures in Adaptive Server using SOAP.

For more information, see the *Web Services User's Guide for Adaptive Server Enterprise*.

Enhanced compatibility with Microsoft SQL extensions

Adaptive Server 12.5.1 provides enhanced compatibility with Microsoft SQL extensions.

Newly supported features let you:

- Use default in the insert statement. For example:

```
create table test_table (col1 tinyint,
   col2 tinyint default 10, col3 tinyint)
insert test_table values (11, default, 43)
```

- Use set for variable and value assignments. For example:

```
declare @a tinyint ; set @a = 10
```

- Use square brackets surrounding identifier names. For example:

```
create table [toto2] (a tinyint)
```

- Use the new built-in function cast() and those added in version 12.5.0.3: len(), left(), day(), month(), year(), str_replace(), newid(), square().
- Use derived tables, which can be used wherever a view can be used, and which can be used in the from list of select, select into, create view, and insert statements.
- Migrate more easily between Adaptive Server and MS-SQL Server.

Adaptive Server 12.5.1 backward compatibility

Before using the new Adaptive Server 12.5.1 features, make sure that you read the Technote "Adaptive Server 12.5.1 Backward Compatibility," which is available at http://www.sybse.com/support/techdocs.

You can downgrade to previous 12.5.x versions of Adaptive Server, but to do so you must take the precautions described in the technical note *before* using any of the new 12.5.1 features.

System changes in version 12.5.1

This section provides a summary of changes to configuration parameters, command, functions, and so on to Adaptive Server version 12.5.1.

New and changed configuration parameters

This section describes the new and changed configuration parameters in version 12.5.1.

Table 4-2 summarizes the new configuration parameters:

Table 4-2: New configuration parameters

Parameter	Function
enable ldap user auth	Enables the storage of user login information on the LDAP server.
enable job scheduler	Enables the Job Scheduler.

Table 4-3 summarizes the changes to existing configuration parameters.

Table 4-3: Changed configuration parameters

Parameter	Change
number of user connections	One connection is reserved for temporary administrative tasks, and has a total login duration of 15 minutes. The reserved connection ensures the availability of connections for database administrators when there are no free connections and number of user connections needs to be increased.

New and changed Transact-SQL commands

This section describes the new and changed Transact-SQL commands in version 12.5.1.

Table 4-4 summarizes the new Transact-SQL commands.

Table 4-4: New commands

Command	Function
mount	Use the mount command to add information for devices and other attributes for the database to the destination or secondary Adaptive Server. The mount command decodes the information in the manifest file and makes the set of databases available online.
unmount	The unmount command shuts down the database and drops it from the Adaptive Server. The devices are also deactivated and dropped.

Table 4-5 summarizes the changes to existing Transact-SQL commands:

Table 4-5: Changed commands

Command	Change
select	A SQL-derived table can be used in the from clause of a select statement. This includes a select statement that appears in an insert or create view statement. For more information, see the *Adaptive Server Enterprise Reference Manual*.
checkpoint	Enhancements: • Allows specification of an optional list of target databases or the keyword all. • Performs an automatic checkpoint for sp_dboption, which requires checkpoint to continue. • Permission to execute checkpoint is granted to the oper_role.

New functions

Table 4-6 describes new function for 12.5.1.

Table 4-6: New functions

Function	Description
current_date	Returns the current date as a date datatype.
current_time	Returns the current time as a time datatype.

New and changed system procedures

This section describes the new system procedures added and changes made to existing system procedures.

Table 4-7 summarizes the new system procedures:

Table 4-7: New system procedures

System procedure	Function
sp_dbextend	User interface for the automatic expansion process.
sp_ldapadmin	Creates an LDAP URL search string, lists an LDAP URL search string, or verifies an LDAP URL search string or login.
sp_listener	Dynamically starts and stops listeners on Adaptive Server on any given port on a per-engine basis.

Table 4-8 summarizes the changes made to existing system procedures.

Table 4-8: Changed system procedures

Procedure	Change
sp_addserver	*Component Integration Services only* – You can use *pname* to specify the hostname or IP address and the port of the server you wish to connect to.
sp_bindcache and sp_unbindcache	Although you can still use sp_bindcache and sp_unbindcache on a system tempdb, the binding of the system tempdb is now non-dynamic.
sp_cacheconfig	Was a static and required a restart of Adaptive Server to take effect. Now a dynamic option.
sp_dbcc_evaluatedb	The output for sp_dbcc_evaluatedb is now more compact than before.
sp_dbcc_faultreport	Provides a new parameter @hard_only, which specifies a report of hard faults only.
sp_dboption	Now includes the checkpoint command. Users no longer need to execute checkpoint after executing sp_dboption.
sp_serveroption	The procedure has the following additional options: • external engine auto start • server cost • server login
sp_sysmon	Now includes the cache wizard option, which allows you to monitor and configure data caches.

New and changed system tables

There are no new or changed system tables for Adaptive Server 12.5.1. For more information about system tables, see the *Adaptive Server Reference Manual*.

New databases

Table 4-9 lists the databases that are new in this version.

Table 4-9: New databases

Database	Function
sybmgmtdb	Job Scheduler has its own database, called sybmgmtdb. Any results or output are logged to the log tables in the sybmgmtdb database. All the job, schedule, scheduled job information, and data needed by the Job Scheduler Task for internal processing is stored in the sybmgmtdb database. Most access to data in the sybmgmtdb database is through stored procedures. The stored procedures make the data available to the GUI, the Job Scheduler Agent and the command-line interface. Only the Job Scheduler Task accesses data directly from the sybmgmtdb database.

New reserved words

There are no new reserved words for Adaptive Server 12.5.1.

The *Adaptive Server Reference Manual* lists all of the reserved words for version 12.5.1. Reserved words cannot be used as object names or column names.

You must change all database names that are new reserved words before you can upgrade from an earlier version of Adaptive Server. You can change table, view, and column names, or use delimited identifiers. Once you upgrade to version 12.5.1, you cannot use database objects whose names are new reserved words until you modify your procedures, SQL scripts, and applications.

Changes to the documentation

Adaptive Server version 12.5.1 includes the following new books, which describe separately licensed Adaptive Server features:

- *Web Services User's Guide* – describes the Web Services feature, and provides instructions for configuration, administration, and troubleshooting.

- *XML Services in Adaptive Server Enterprise* – describes the Sybase native XML processor and the Sybase Java-based XML support, introduces XML in the database, and documents the query and mapping functions that comprise XML Services. This book includes XML information previously described in *Java in Adaptive Server*.
- *Job Scheduler User's Guide* – provides instructions on installing and configuring, and creating and scheduling jobs on a local or remote Adaptive Server using the command line or a graphical user interface (GUI).

CHAPTER 5

New Features in Adaptive Server Version 12.5.0.x

This chapter describes the new features and system changes introduced with Sybase Adaptive Server Enterprise version 12.5.0.x.

Name	Page
Monitoring system tables in Adaptive Server	80
Index selection	82
Active-passive configuration for Sun Cluster 3.0	82
Asynchronous log service and optimistic index locking	83
Using sampling for update statistics	84
Multiple temporary databases	84
Housekeeper enhancements	85
Chinese character set (GB18030) support	86
SQLJ aliases for XQL	86
SQLX mapping functions	86
New configuration parameters in sp_monitorconfig	86
Faster checkpoints	87
Changes to checkstorage	87
Changes to CIS	88
Improved performance for select into	89
New set parameters	90
Improved performance for create database, alter database, load database, and dbcc checkalloc	90
Number of databases per transaction	90
System changes	90

New features in version 12.5.0.3

The new features in Adaptive Server version 12.5.0.3 are:

Monitoring system tables in Adaptive Server

Adaptive Server 12.5.0.3 includes a set of system tables that contain monitoring and diagnostic information. The information in these tables provides you with a statistical snapshot of the state of Adaptive Server, which allows you to analyze the server for performance improvements. You can query these system tables in much the same way you currently query any other tables in Adaptive Server. For example, to display statistical information about I/O on Sybase devices:

```
select * from monDeviceIO
Reads    APFReads     Writes   DevSemaphoreRequests  DevSemaphoreWaits IOTime
LogicalName PhysicalName
-------- -----------  -------- --------------------  ----------------- -------
--------------------------   -------------------------------
1563    7           7891    3                     0             134900
master                         /dev/vx/rdsk/sybase/master_vol01
59       0             15       2                        0                 800
engcomdb_data_vol01            /dev/vx/rdsk/sybase/engcomdb_data_vol01
5        0           13      2                     0             100
engcomdb_log_vol01             /dev/vx/rdsk/sybase/engcomdb_log_vol01
126255  59657       8604    2                     0             1408700
qts_db_data_vol01              /dev/vx/rdsk/sybase/qts_db_data_vol01
31       0           9879    2                     0             128400
qts_db_log_vol01               /dev/vx/rdsk/sybase/qts_db_log_vol01
51       0           19      2                     0             400
sadb_data_vol01                /dev/vx/rdsk/sybase/sadb_data_vol01
5        0           12      2                     0             200
sadb_log_vol01                 /dev/vx/rdsk/sybase/sadb_log_vol01
56       0             25       2                        0                 900
scratchdb_vol01                /dev/vx/rdsk/sybase/scratchdb_vol01
0        0           0       2                     0              0
rmdb_data_vol01                 /dev/vx/rdsk/sybase/rmdb_data_vol01
0         0           0       2                      0               0
rmdb_log_vol01                 /dev/vx/rdsk/sybase/rmdb_log_vol01
52658    424          99512   2                      0               2231300
sysprocsdev                    /dev/vx/rdsk/sybase/sybsystemprocs_vol01
146      0           3569    2                      0               13700
tempdb_data                    /tmp/tempdb_data
4        0           814     2                      0               400
tempdb_log                          /tmp/tempdb_log
```

Where monDeviceIO is the system table that contains statistical information about disk I/O (physical name, reads, writes, and so on). To perform this query, the user does not need to know if Monitor Server or any other monitoring agent is executing on the same host server; the monDeviceIO table contains all the information they need.

The monitoring tables are not created by default, but must be created using the *installmon* script in the *$SYBASE/ASE-12_5/scripts* directory.

Note You must have the mon_role role to query these tables.

Using Transact-SQL to monitor performance

Providing monitoring information as tables enables you to use Transact-SQL to monitor Adaptive Server. For example, to identify the Transact-SQL statements that are currently consuming CPU, enter:

```
select s.SPID, s.CpuTime, t.LineNumber, t.SQLText
from monProcessStatement s, monProcessSQLText t
where s.SPID=t.SPID
order by s.CpuTime, s.SPID, t.LineNumber desc
```

You can use this same query to find the SQL statements that are using the most physical I/O by substituting CpuTime for PhysicalReads.

The information in each monitoring table can be sorted, selected, joined, inserted into another table, and treated much the same as the information in a regular Adaptive Server table.

The tables are read-only because they are in-memory tables that are generated as they are queried.

Access to these tables is restricted to users with the mon_role role.

The definitions for the monitoring tables have a similar syntax to CIS definitions, which allow remote procedures to be defined as local tables. For example, the following syntax is used to create the monNetworkIO table on a server named “loopback:

```
create existing table monNetworkIO (
    PacketsSent int,
    PacketsReceived int,
    BytesSent int,
    BytesReceived int,
)
external procedure
```

```
at "loopback...$monNetworkIO"
```

See the *Performance and Tuning Guide* for more information.

Index selection

Index selection allows you to determine which indexes are actively being used and those that are rarely used. The index selection feature is used with the monitoring tables feature.

See the *Performance and Tuning Guide* for more information.

Active-passive configuration for Sun Cluster 3.0

Adaptive Server for Sun Cluster 3.0 offers either an active-passive or an active-active setup.

An active-active setup includes two Adaptive Servers installed on two nodes of the cluster that are configured as companion servers. When the primary companion fails, the secondary companion takes over its workload. The active-active setup for Sun Cluster 3.0 is described in Chapter 11 of *Using Sybase Failover in a High Availability System.*

If Adaptive Server, or the node on which it is running, fails, the server and all its devices fails over to a secondary node. At the time of failover, all existing client connections are lost. Clients must reestablish their connections and resubmit their uncommitted transactions as soon as Adaptive Server is running on the secondary node. The failover of client connections (that is, reestablishing their connections after failover) can be done automatically by using the self-referencing *hafailover* entry in the interfaces file and setting the *CS_HAFAILOVER* connection property.

If your site includes more than two nodes in an active-passive setup, the server running on the primary node can fail over to any of the secondary nodes, as long as it can successfully host Adaptive Server and all its associated resources. The node list determines the order in which Adaptive Server fails over to the other nodes in the cluster.

Table 5-1 discusses the differences between an active-active setup and an active-passive setup.

Table 5-1: Differences between active-active and active-passive

Active-active	Active-passive
Setup: Two Adaptive Servers are configured as companion servers, each with an independent resource group. These companions run on the primary and secondary nodes, respectively, as individual servers until failover.	*Setup*: One Adaptive Server, running either on the primary node or on the secondary node. Only one Adaptive Server resource group exists, which is hosted by the primary node before failover and the secondary node after failover.
Failover: When failover occurs, the secondary companion takes over the devices, client connections, and so on from the primary companion. The secondary companion services the failed-over clients, as well as any new clients, until the primary companion fails back and resumes its activities.	*Failover*: When failover occurs, the Adaptive Server along with its associated resources and resource group is relocated to, and restarted on, the secondary node.
Failback: Failback is a planned event during which the primary companion takes back its devices and client connections from the secondary companion to resume its services.	*Failback*: Failback is a planned failover or relocation of the Adaptive Server resource group to the another node. Failback is not required, but can be used for administration.
Client Connection failover: During failover, clients connect to the secondary companion to resubmit their transactions. During failback clients connect to the primary companion to resubmit their transactions. Clients with the failover property reestablish their connections automatically	*Client Connection failover*: During failover and failback, clients connect to the same Adaptive Server to resubmit uncommitted transactions. Clients with the failover property reestablish their connections automatically.

Asynchronous log service and optimistic index locking

Complex, multitier applications that consolidate the various parts of an enterprise system increase the load on high-volume database servers. When you address this by upgrading database hardware, from 4 to 8 CPU systems to 16, 32, or 64 CPU systems, you can drastically increase contention on such key resources as the log, the spinlocks that guard the log, and the address locks. Asynchronous log service (ALS) and optimistic index locking help resolve contention, increasing Adaptive Server scalability in systems using four or more online engines.

Optimistic index locking can resolve increased contention on some important resources, such as the spinlocks that guard address locks on the root page of an index.

Applications where this amount of contention might occur are typically those in which:

- Access to a specified index constitutes a significant portion of the transaction profile, and many users are concurrently executing the same workload.
- Different transactions, such as ad hoc and standardized queries, use the same index concurrently.

Using sampling for *update statistics*

The Adaptive Server query optimizer uses statistics to optimize queries.

Run the update statistics commands on tables (data sets) to update information about the distribution of key values in specified indexes or columns, for all columns in an index, or for all columns in a table. The commands revise histograms and density values for column-level statistics.

Multiple temporary databases

Adaptive Server version 12.5.0.3 allows you to create and manage multiple temporary databases in addition to the system tempdb, which was the only temporary database in the server in earlier versions of Adaptive Server.

Multiple temporary databases, also referred to as **tempdbs**, reduce contention on system catalogs and logs in the system tempdb. They allow you to:

- Create temporary databases on fast-access devices
- Drop a temporary database to reclaim storage
- Partition tasks that create temporary objects into using specific tempdbs, which prevents these tasks from interfering with other sessions that need to use temporary database space

The multiple temporary database feature is fully enabled for:

- New installations
- Installations that are upgrading from an Adaptive Server earlier than version 12.5
- Databases that are loaded from an Adaptive Server earlier than version 12.5

The tempdb database is the system-created temporary database. Before Adaptive Server version 12.5.0.3, tempdb was the only temporary database in the server. Temporary tables and work tables are created in tempdb.

Housekeeper enhancements

Adaptive Server version 12.5.0.3 includes the following enhancements to the housekeeper utility:

- In versions earlier than 12.5.0.3, one housekeeper performed five tasks. The work is now divided among three separate tasks:
  - Housekeeper wash task
  - Housekeeper chores task
  - Housekeeper garbage collection task

  The output for all three tasks appears in the output for sp_who.
- Washing buffers is an optional task and runs only at idle times. You can turn off this task using the configuration parameter housekeeper free write percent. The housekeeper wash task is the only housekeeper task for which you use this configuration parameter.
- HK_CHORES runs at idle times only and does not use a common configuration parameter. It manages miscellaneous chores, such as:
  - Automatically flushing table statistics.
  - Automatically flushing account statistics.
  - Handling timeout of detached transactions. You can turn off this chore using the configuration parameter dtm detach timeout period.
  - Checking licence usage. You can turn this task off using the configuration parameter license information.
- There are two forms of garbage collection, lazy and aggressive. These terms describe two distinct tests for finding empty pages.
  - Lazy garbage collection refers to an inexpensive test to find empty pages. This test may not be effective during long-running transactions, and empty pages may accumulate. Lazy garbage collection is inexpensive to use, but can lower performance. Performance is affected by the fragmentation of space allocated to a table, and by the accumulation of empty pages that must be evaluated during queries.

- Aggressive garbage collection refers to a sophisticated test for empty pages. This test is more expensive than the lazy garbage collection test, because it checks each deleted row in a page to determine whether that deleting transactions are committed.

  Both the delete command and the housekeeper garbage collection task can be configured for aggressive or lazy garbage collection, through the configuration parameter enable housekeeper GC.

  The aggressive housekeeper garbage collection self-tunes the frequency with which the housekeeper garbage collection task examines the housekeeper list, so that the frequency of examination matches the rate at which the application generates empty pages.

Chinese character set (GB18030) support

Adaptive Server 12.5.0.3 supports the Chinese character set (GB18030).

SQLJ aliases for XQL

Adaptive Server includes a query processor for the XML query language (XQL), written in Java. The SQLJ alias feature simplifies SQL references to Java static methods by allowing you to create SQLJ alias names for them, rather than requiring you to reference them by their fully qualified Java names.

SQLX mapping functions

Adaptive Server Enterprise 12.5.0.3 includes a set of SQLX mapping functions, for mapping between SQL result sets and XML documents. These mapping functions are implemented with a set of Java classes that you can invoke in a client environment, by placing the *jar* file on your CLASSPATH, or in Adaptive Server, by installing the classes in the server.

New configuration parameters in *sp_monitorconfig*

sp_monitorconfig adds the all parameter. The new syntax for sp_monitorconfig is:

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

or

```
sp_monitorconfig "all" [, "result_tbl_name"]
```

configname is all or part of the configuration parameter name whose monitoring information is being queried. Until Adaptive Server 12.5, the only valid configuration parameters were open indexes, open objects, open databases, aux scan descriptors, txn to pss ratio, number of dtx participants, and all. Specifying all displays descriptor help information for all indexes, objects, databases, and auxiliary scan descriptors in the server.

The number of configuration parameters sp_monitorconfig supports has been increased from 6 to 34.

Faster checkpoints

In Adaptive Server Enterprise 12.5.0.3, the checkpoint task in a specified database executes up to 200% faster, according to internal benchmarks.This increase in checkpoint speed can cause significant gains in database recovery speeds. The checkpoint feature enhancement is automatic, and requires no action on the user's part.

Issuing a larger batch I/O of dirty buffers causes this increase in checkpoint speed. The new default value of the I/O batch size is 100.

Changes to *checkstorage*

Adaptive Server 12.5.0.3 provides the following changes to dbcc checkstorage. You can:

- Execute dbcc stored procedures from any database.
- Generate reports that provide recommended actions to address faults reported by checkstorage.
- Generate reports on aborted checkstorage and checkverify operations.
- Limit checkverify processing to a specific object.
- Exclude tables, faults, and combinations of faults and tables from processing by checkverify and sp_dbcc_faultreport.
- Create default configuration values for the dbcc configuration parameters and view them with sp_dbcc_configreport.

- Delete values of the dbcc configuration parameters.
- Avoid configuring an explicit cache for checkstorage operations.
- Avoid creating scan or text workspaces for every database on which checkstorage reports. You can create both or either one, the scan or the workspaces. Default scan and text workspaces are created when you run the *installdbccdb* script.
- Avoid running sp_dbcc_updateconfig to reregister a workspace that you have dropped and re-created.
- Avoid manually expanding the dbccdb workspaces.

Changes to CIS

CIS for Adaptive Server version 12.5.0.3 includes the discussed changes in this section.

set bulk array size

set bulk array size establishes the number of rows that will be buffered in local server memory before they are transferred with the bulk copy interface. The syntax for set bulk array size is:

set bulk array size *number*

The value of *number* indicates the number of rows to buffer. If the rows being transferred contain text, image, or Java Abstract Data Types (Java ADTs), the bulk copy interface ignores the current setting for array size and uses a value of 1. Also, the array size actually used never exceeds the value of *@@bulkbatchsize*. If *@@bulkbatchsize* is smaller than the array size, then the smaller value is used.

set bulk batch size

set bulk batch size establishes the number of rows transferred to a remote server via select into *proxy_table* when the bulk interface is used. The bulk interface is available to all Adaptive Server versions, as well as DirectConnect for Oracle version 12.5.1.

This set command is used with Component Integration Services for transferring rows to a remote server using select into.

Global variables

There are several new global variables for CIS.

@@bulkarraysize

@@bulkarraysize shows the current setting of bulk array size. The following isql session illustrates the use of *@@bulkarraysize*:

```
1> select @@bulkarraysize
2> go

-----------
50    <== default value

(1 row affected)
```

@@bulkbatchsize

@@bulkbatchsize shows the current setting of bulk batch size. The following isql sessions illustrates the use of *@@bulkbatchsize*:

```
1> select @@bulkbatchsize
2> go

-----------
0    <== default value

(1 row affected)
```

Improved performance for *select into*

Adaptive Server Enterprise version 12.5.0.3 enables large-page allocation for the target table in select into, by allocating pages by extent rather than individual page, and thus issuing fewer logging requests for the target table. Instead of logging and updating information for each new page in sysindexes, Adaptive Server updates only once, to the last page in sysindexes, as the last operation in select into.

Although select into statements use a 2K I/O buffer pool by default, the optimizer can choose the largest pool possible, depending on the size of the buffer pools available in the target database cache. If you configure Adaptive Server with large buffer pools, it uses large I/O buffer pools for writing the target table pages to disk.

New *set* parameters

Adaptive Server 12.5.0.3 includes the identity_update parameter for the set option, which supports updating the value of the identity column.

Improved performance for *create database*, *alter database*, *load database*, and *dbcc checkalloc*

The size of large I/O disk buffers used by create database, alter database, load database, and dbcc checkalloc to issue multiple parallel I/O is now one allocation, (256 pages), not one extent (8 pages). This increase means that the server requires more memory allocation to issue large buffers. The server now requests that more memory be configured, but this memory is not allocated until a command requests large I/O buffers. A disk buffer that required memory for 8 pages in earlier versions now needs memory for 256 pages. The limit on the configuration parameter number of large io buffers, with a default value of 6, has increased from 32 to 256. At the same time, the internal limit on the number of large I/O buffers a single command can create has been increased from 8 to 32. These increases allow you to both create larger buffers, and use more buffers per command. Whatever value you assign to the configuration parameter dynamic allocation on demand, memory for large I/O buffers is always allocated on demand. This memory pool is shrinkable: if the number of large I/O buffers decreases, memory requirements per buffer decrease accordingly.

Number of databases per transaction

In Adaptive Server Enterprise version 12.5.0.3, the maximum number of databases Adaptive Server allows per user transaction is the maximum number of databases that can be created within Adaptive Server. Adaptive Server returns an error message if it is not configured for more than 16 databases within a transaction.

System changes

This section describes the system changes in Adaptive Server version 12.5.0.3.

New and changed configuration parameters

Table 5-2 summarizes the new configuration parameters in version 12.5.0.3.

Table 5-2: New configuration parameters

Parameter	Function
deadlock pipe active	Controls whether Adaptive Server collects deadlock messages.
deadlock pipe max messages	Determines the number of deadlock messages Adaptive Server stores.
enable monitoring	Controls whether Adaptive Server collects the monitoring table data.
errorlog pipe active	Controls whether Adaptive Server collects error log messages.
errorlog pipe max messages	Determines the number of error log messages Adaptive Server stores.
max sql text monitored	Specifies the total number of bytes Adaptive Server allocates for each user task to store SQL text.
object lockwait timing	Controls whether Adaptive Server collects timing statistics for requests of locks on objects.
per object statistics active	Controls whether Adaptive Server collects statistics for each object.
plan text pipe active	Determines whether Adaptive Server collects query plan text.
plan text pipe max messages	Determines the number of query plan text messages Adaptive Server stores.
process wait events	Controls whether Adaptive Server collects statistics for each wait event for every task.
sql text pipe active	Controls whether Adaptive Server collects SQL text.
sql text pipe max messages	Specifies the number of SQL text messages Adaptive Server stores.
statement pipe active	Controls whether Adaptive Server collects statement-level statistics.
statement pipe messages	Determines the number of statement statistics messages Adaptive Server stores.
statement statistics active	Controls whether Adaptive Server collects the monitoring table statement-level statistics.
sql batch capture	Controls whether Adaptive Server collects SQL text.
wait event timing	Controls whether Adaptive Server collects statistics for indiviual wait events.

Table 5-3 summarizes changed configuration parameters in 12.5.0.3.

Table 5-3: Changed configuration parameters

Parameter	Changes
enable housekeeper GC	Provides these parameter values: • 0 – disables the housekeeper garbage collection task. • 1 – enables lazy garbage collection. • 2 – reserved for future use. • 3 – reserved for future use. • 4 – enables aggressive garbage collection for both the housekeeper garbage collection task and the delete command. • 5 – enables aggressive garbage collection for the housekeeper garbage collection task, and lazy garbage collection by the delete command.

New and changed Transact-SQL commands

Table 5-4 summarizes the new Transact-SQL commands in version 12.5.0.3.

Table 5-4: New commands

Command	Description
set identity_update	Updates the value of the IDENTITY column on a table.

Table 5-5 summarizes the changes to existing Transact-SQL commands in version 12.5.0.3.

Table 5-5: Changed commands

Command	Change
update statistics	Adds the sampling=*percent* percent option, which specifies a sampling percent.
dbcc	Adds these new parameters: • pravailabletempdbs – prints the global list of available temporary databases. • addtempdb – adds a temporary database to the global list of available temporary databases.
create database	Adds the temporary parameter, which creates a temporary database.

New and changed system procedures

Table 5-6 summarizes the changes made to existing system procedures.

Table 5-6: Changed system procedures

Procedure	Change
sp_monitorconfig	Adaptive Server 12.5.0.3 adds the all parameter, which displays descriptor help information for all 34 indexes, objects, databases, and auxiliary scan descriptors in the server.

Procedure	Change
sp_dboption	• Adds the async log service option, which enables, disables, or configures the asynchronous log service feature. • Can prevent user-created temporary databases from being set to single-user mode. •
sp_helpdb	• Adds a column to display asynchronous log service information. • Reports whether or not a database is a user-created temporary database in the status column.
sp_help	Adds a column that displays optimistic index lock information.
sp_chgattribute	Adds the optimistic_index_lock option, which acquires an optimistic index lock on a table.
sp_bindcache	Can precent bvinding of individual tables to a named cache in user-created temporary databases.
sp_dbcc_summary_report	Adds the display_recommendations parameter, which enables reporting the recommendations generated by sp_dbcc_recommandations.
sp_dbcc_faultreport	Displays faults in ordered list by table and index. Adds these new parameters: • exclusion_mode – ignores or applies exclusion lists. • exclusion_faults – list of fault types to be excluded from reporting. • exclusion_tables – list of tables to be excluded from reporting. • exclusion_combo – list of fault/table combinations to be excluded from reporting. • display_recommendations – enables reporting the recommendations generated by sp_dbcc_recommendations.
sp_dbcc_checkverify	Adds these new parameters: • *ignore_exclusions* – enables or disables the exclusion list. • *tblename* – limits checkverify processing to a specific .object.
sp_dbcc_updateconfig	Provides default configuration values for dbcc configuration values. Accepts delete as a value for the str1 parameter. Disables enable automatic workspace expansion for databases that are enabled by default.
sp_dbcc_configreport	Adds the defaults parameter, which enables only the display of configured default values.

New system procedures

Table 5-7 summarizes the new system procedures.

Table 5-7: New system procedures

Procedure	Change
sp_helptext	Displays the **source text** of a **compiled object**.
sp_tempdb	Creates the default temporary database group and binds temporary databases to the default temporary database group.

Procedure	Change
sp_dbcc_recommendations	Analyzes faults reported by the checkstorage operation and summarizes them.
sp_dbcc_help_fault	Provides a description of the specified fault type and the recommended fix.
sp_dbcc_exclusions	Allows the user to create and manage persistent exclusion lists for use by checkverify and sp_dbcc_faultreport.

New and changed utility programs

Table 5-8 summarizes changed utility programs in Adaptive Server version 12.5.0.3.

Table 5-8: Changed utility programs

Utility	Function
sybmigrate	In Adaptive Server 12.5.0.3, sybmigrate selects dependent objects for migration when you use the auto-select feature. These changes have been made to the command line configuration: • -rn diff – a difference report feature that compares source and target database objects. • -rn password – a new report feature to create a file for the changed passwords.

New global variables

Table 5-9 describes new global variables for 12.5.1.

Table 5-9: New global variables

Global variable	Definition
@@datefirst	Set using set datefirst n where n is a value between 1 and 7. Returns the current value of *@@datefirst*, indicating the specified first day of each week (1 for Sunday, 2 for Monday, and so on), expressed as tinyint. The default value in Adaptive Server is 1, Sunday (based on the us_language default).
@@lock_timeout	Set using set lock wait n. Returns the current *lock_timeout* setting, in milliseconds. *@@lock_timeout* returns the value of n. The default value is no timeout. If no set lock wait n is executed at the beginning of the session, *@@lock_timeout* returns -1.
@@max_precision	Returns the precision level used by decimal and numeric datatypes set by the server. This value is a fixed constant of 38.
@@tempdbid	Returns a valid temporary database ID of the session's assigned temporary database.

New functions

Table 5-10 describes the functions added for Adaptive Server version 12.5.0.3.

Table 5-10: New functions

Global variable	Definition
year	Returns an integer that represents the year in the datepart of a specified date.
month	Returns an integer that represents the month in the datepart of a specified date.
day	Returns an integer that represents the day in the datepart of a specified date.
str_replace	Replaces any instances of the second string expression (*string_expression2*) that occur within the first string expression (*string_expression1*) with a third expression (*string_expression3*.
square	Returns the square of a specified value expressed as a float.
left	Returns a specified number of characters on the left end of a character string.
len	Returns the number of characters, not the number of bytes, of a specified string expression, excluding trailing blanks.
newid	Generates human-readable, globally unique IDs (GUIDs) in two different formats, based on arguments you provide. The length of the human-readable format of the GUID value is either 32 bytes (with no dashes) or 36 bytes (with dashes).
tempdb_id	Reports the temporary database to which a given session is assigned.
next_identity	Retrieves the next identity value that is available for the next insert.
identity_burn_max	Tracks the identity burn max value for a given table.

New and changed system tables

This section describes new system tables to support version 12.5.0.3 features. Table 5-11 describes new system monitoring tables.

Table 5-11: New system tables

System table	Description
monTables	Provides a description of all monitoring tables. No configuration options are required.
monTableParameters	Provides a description of all the optional parameters for each monitoring table. No configuration options are required.
monTableColumns	Describes all the columns for each monitoring table. No configuration options are required.
monState	Provides information regarding the overall state of Adaptive Server.
monEngine	Provides statistics regarding Adaptive Server engines. Requires the enable monitoring configuration parameter to be enabled.
monDataCache	Returns statistics relating to Adaptive Server data caches. Requires the enable monitoring configuration parameter to be enabled.
monProcedureCache	Returns statistics relating to Adaptive Server procedure cache. Requires the enable monitoring configuration parameter to be enabled.
monOpenDatabases	Provides state and statistical information pertaining to davases that are currently in use.
monSysWorkerThread	Returns server-wide statistics related to worker threads. Requires the enable monitoring configuration parameter to be enabled.

System table	Description
monNetworkIO	Returns network I/O statistics. Requires the enable monitoring configuration parameter to be enabled.
monErrorLog	Returns the most recent error messages from the Adaptive Server error log.
monLocks	Returns a list of all locks that are being held, and those that have been requested, by any process, for every object.
monDeadLock	Provides information pertaining to the most recent deadlocks that have occurred in Adaptive Server.
monWaitClassInfo	Provides a textual description for all of the wait classes. All events have been grouped into wait classes that classify the type of event that a process is waiting for.
monWaitEventInfo	Provides a textual description for every possible situation where a process is forced to wait within Adaptive Server.
monCachedObject	Returns statistics for all objects and indexes with pages currently in a data cache.
monCachePool	Provides statistics for all pools allocated for all caches.
monOpenObjectActivity	Provides statistics for all open objects.
monIOQueue	Provides device I/O statistics broken down into data and log I/O for normal and temporary databases on each device.
monDeviceIO	Returns statistical information relating to devices.
monSysWaits	Provides a server-wide view of where processes are waiting for an event.
monProcess	Provides detailed statistics about processes that are currently executing or waiting.
monProcessLookup	Provides information enabling processes to be tracked to an application, user, client machine, and so on.
monProcessActivity	Provides detailed statistics about process activity.
monProcessNetIO	Provides the network I/O activity for each process.
monProcessObject	Provides statistical information regarding objects that have been accessed by processes.
monProcessWaits	Provides a server-wide view of where processes are waiting for an event.
monProcessStatement	Provides information for currently executing statements.
monProcessSQLText	Provides the SQL text that is currently being executed.
monSysPlanText	Provides the most recent generated text plan.
monSysStatment	Provides statistics pertaining to the most recently executed statements.
monCachedProcedures	Provides statistics for all procedures currently stored in the procedure cache.
monSysSQLText	Provides the most recent SQL text that has been executed, or is currently being executed.
monProcessProcedures	Returns a list of procedures that are being executed by processes.

Table 5-12 describes changes to system tables to support Adaptive Server 12.5.0.3.

Table 5-12: Changed system tables

System table	Changes
sysattributes	Displays information about the representation of temporary database groups and bindings.
sysdatabases	Adds a new bit which indicates the temporary status of a database by the value of 0x00000100 (256 decimal) in the status3 field.

New global variables

Adaptive Server 12.5.0.3 adds the following global variables.

Table 5-13: New global variables

Global variable	Definition
@@datefirst	Set using set datefirst n where n is a value between 1 and 7. Returns the current value of *@@datefirst*, indicating the specified first day of each week (1 for Sunday, 2 for Monday, and so on), expressed as tinyint. The default value in Adaptive Server is 1, Sunday (based on the us_language default).
@@lock_timeout	Set using set lock wait n. Returns the current *lock_timeout* setting, in milliseconds. *@@lock_timeout* returns the value of n. The default value is no timeout. If no set lock wait n is executed at the beginning of the session, *@@lock_timeout* returns -1.
@@max_precision	Returns the precision level used by decimal and numeric datatypes set by the server. This value is a fixed constant of 38.

New functions

The functions added for Adaptive Server version 12.5.0.3 are:

- year
- month
- day
- str_replace
- square
- left
- len
- next_identity

- identity_burn_max

New features in version 12.5.0.1

The new features in Adaptive Server version 12.5.0.1 are:

Feature
“Options for Adaptive Server version 12.5.0.1” on page 98
“Using grant dbcc and revoke dbcc” on page 99
“Increasing the size of devices with disk resize” on page 99
“ASE Replicator” on page 100
“Synchronizing EJB Servers” on page 100
“XQL result sets” on page 101
“XQL query engine” on page 102
“Proxy table space” on page 103
“Creating files and directories with the insert command” on page 104
“Migration utility” on page 104
“High availability on Sun Cluster 3.0” on page 105
“New character sets” on page 105
“System changes” on page 105

Options for Adaptive Server version 12.5.0.1

Earlier versions of Adaptive Server were sold either at the Enterprise Level or the Workplace Level. Beginning with Adaptive Server version 12.5.0.1, Adaptive Server is available in the Enterprise Edition, the Small Business Edition, and the Developer’s Edition. The Enterprise Edition of Adaptive Server is a full-featured server that can run all optional features. The Small Business Edition of Adaptive Server includes the features required by most small businesses, but excludes some of the more advanced features. The Developer’s Edition is designed for you to design and build applications for Adaptive Server in a development environment.

In addition to offering the three editions of Adaptive Server, Sybase also offers a series of options. Some of these option are intended for an enterprise-level site, and are not available in the Small Business Edition. Table 5-14 describes the availability of each option. For a full description of each option, see your sales representative.

Table 5-14: Options available for Adaptive Server

Options	Enterprise Edition	Small Business Edition	Developer's Edition	Description
Security and Directory Services Edition	Yes	Yes	Yes	Systems continuously available with fault resilience
XML Management Edition	Yes	Yes	Yes	Semi-structured and unstructured-data handling
Content Management Edition	Yes	Yes	Yes	Semi-structured and unstructured-data management
e-business Edition	Yes	No	Yes	Data-storage, access, replication, and security via an e-business platform
High Availability	Yes	No	Yes	Systems continuously available
Data Transfer Management (DTM)	Yes	No	Yes	
Enhanced Full-Text Search (EFTS)	Yes	Yes	No	
SQL Expert	No	No	No	

Using *grant dbcc* and *revoke dbcc*

System Administrators can grant the execution of database consistency check (dbcc) commands to users and roles that do not have System Administrator-level privileges in Adaptive Server. This discretionary access control allows System Administrators to control access to database objects or to certain database- and server-level actions.

Increasing the size of devices with *disk resize*

The disk resize command allows you to dynamically increase the size of your database devices, rather than initializing a new device. For example, if /sybase/testdev.dat requires an additional 10MB of space, you can run disk resize and allocate this amount of space to the device. The create and alter database commands can use this added space.

You can use disk resize to increase the size for devices on raw partitions and for file systems. The minimum amount of space that you can increase a device is 1MB or an allocation unit, whichever is greater.

You cannot use disk resize on dump or load devices.

Any properties that are set on the device continue to be set after you increase its size. That is, if a device has dsync set before you increase its size, it has dsync set afterwards. Also, any access rights that were set before you increased the size of the device remain set.

Only users with the sa role can execute the disk resize command.

You can use audit trails on disk resize to track the number of times a device is resized. The device being resized is always online and available for users during the resize operation.

ASE Replicator

ASE Replicator extends Adaptive Server Enterprise capabilities and provides basic replication from a primary database to one or more replicate databases.

ASE Replicator:

- Replicates both tables and stored procedures
- Maintains database integrity and transactional consistency at multiple sites
- Provides guaranteed delivery of replicated data
- Manages database objects with a publish-and-subscribe model

ASE Replicator uses Component Integration Services (CIS) to handle transaction and RPC distribution to replicate databases.

Synchronizing EJB Servers

Synchronization replicates application files and configuration information between EJB Servers. It is a useful alternative to importing and exporting packages.

You can use the synchronization option to move repository information between servers. For example, you may want to move a development EJB Server's repository to a test server. In this case, the synchronization command is similar to export/import options without the JAR file; instead of moving packages from one server to another, you are moving repository information.

You can synchronize at the server or package level. The level you choose determines the available synchronization options. Options that are not available are dimmed, and cannot be selected.

XQL result sets

The Extensible Query Language (XQL) result set feature allows you to access query results as objects, rather than as strings, and to extract SQL base types easily from an XML document, without parsing results or writing code. The XQL engine is written in Java, so creating a Java object is easy. The engine encapsulates results in the Java class com.sybase.xml.xql.XqlResults.SybXqlResult and its methods.

You can:

- Iterate through XQL result sets to extract XML fragments
- Convert XML into SQL data, for insertion into database tables
- Get the results as a DOM (Document Object Model) object

Table 5-15 describes the methods added for 12.5.0.3.

Table 5-15: com.sybase.xml.xql.Xql.results.SybXqlResult specified methods

Method	Action
next()	Moves marker to the next item in the result set.
getXML()	Gets the current item in the result set and returns it as a java.lang.String object. If the internal marker is set to 0, this method returns the whole result set as java.lang.String.
getDOM()	Gets the current item in the result set and returns it as an org.w3c.Document getDOM() object. If the internal marker is set to 0, this method returns the whole result set as org.w3c.Document.
getString()	Gets the current item in the result set and returns it as a java.lang.String object. If the current item has nested elements, this method generates an exception.
getInt()	Gets the current item in the result set and returns it as an integer. If the current item has nested elements or cannot be cast as an integer, this method generates an exception.
getDouble()	Gets the current item in the result set and returns it as a double. If the current item has nested elements or cannot be cast as a double, this method generates an exception.

Method	Action
getFloat()	Gets the current item in the result set and returns it as a float. If the current item has nested elements or cannot be cast as a float, this method generates an exception.
reset()	Resets the internal marker to 0, which means that immediate calls to getDOM() or getXML() yield the entire result set as either java.lang.String or org.w3c.Document object. Calling reset() reinitializes the result set marker.
isFirst()	Returns true if the current item is the first item in the result set.
isLast()	Returns true if the current item is the last item in the result set.

XQL query engine

The XQL character processing engine now supports storage and querying of XML documents containing strings in native character sets.

The XML parser accepts XML documents as a stream of bytes containing characters in native code sets. The XML parsing engine accepts XML documents as either java.io.InputStream or as java.lang.String. The XML parser checks XML documents and indexes the XML document in any language or character set the XML standard supports.

The query engine parses queries made against the XML document, processes them, and returns query results as java.lang.String, which is Unicode. You can use the XQL engine either as part of Adaptive Server or as a standalone engine.

The processing engine returns results as java.lang.String, which is Unicode. You should then convert the results of your query from Java strings back to your native code set.

The query engine allows you to:

- Set the character set encoding. The XQL query engine in Adaptive Server mode processes documents whose character set encoding is the same as the character set of the Adaptive Server. If you use the query engine in a standalone server, character set encoding is set using the Java method com.sybase.xml.xql.Xql.setEncoding().

  Note com.sybase.xml.xql.Xql.setEncoding() is not supported in Adaptive Server. Use it only in standalone mode.

- Parse your XML document. Your XML document heading specifies the native character set of the data you enter, in java.lang.String or java.io.InputStream. The parser generates events while it parses an XML document. These events pass their information to the indexing layer as java.lang.String.

  The new flexibility of the query engine requires no change in your parsing process if you use the Xerces parser. If you use a different parser, make sure your parser supports the native character set encoding of your XML document, which is specified in the encoding directive of the XML document heading.

- Query. The XQL query engine accepts queries as java.io.InputStream or java.lang.String. The java.io.InputStream parameter allows character set queries to be fed into the query parser without conversion.

  Internally, the query engine translates the incoming stream of bytes to Unicode, using either the character encoding of Adaptive Server, or com.sybase.xml.xql.Xql.setEncoding() in standalone mode.

- Obtain results. You can use the sample code provided in this document to convert the results of your query from java.lang.string back to native code sets.

Proxy table space

Proxy tables in earlier versions of Adaptive Server occupy one extent (8 pages), as well as one extent for each index on a proxy table. In a pre-12.5.0.1 server with 16K logical page size, each proxy table uses 128K worth of space.

In Adaptive Server 12.5.0.1, proxy tables and indexes do not use extents; they use space only in the system catalogs, which Sybase estimates to be 1MB per 100 proxy tables (assuming an average of two indexes per table). The normal upgrade does not reclaim this unused space. To reclaim this space, first drop, then re-create the proxy table.

Creating files and directories with the *insert* command

You can use the insert command to create a new file directory. To do so, use only the filename, filetype, and content columns. Specify "DIR" as the file type to create filename as a directory. Otherwise, filename is created as a regular file. If filename contains subdirectories, and the subdirectories do not exist, they are created automatically.

Also:

- The value of *@@identity* is updated when an insert completes, if you created a regular file.
- If you attempt to create a file that already exists, an error is raised. You must first delete the file before you can overwrite the contents with an insert command.

You can use insert to transfer the contents of one directory to another, as in the following examples:

```
create proxy_table srctab external directory at '/work/sourcedir;R'
create proxy_table dsttab external directory at '/work1/destdir;R'
insert into dsttab (filename, filetype, content)
Select filename, filetype, content from srctab
```

Migration utility

Adaptive Server version 12.5 and later allows you to configure the page size of your server to use 2K, 4K, 8K, or 16K. However, traditional Adaptive Server upgrade procedures do not allow you to change the page size. Therefore, to convert your current Adaptive Server from one page size to another page size, you must first install a second Adaptive Server with the desired page size, and then use sybmigrate to transfer both schema and data from your original (source) Adaptive Server to the new (target) Adaptive Server.

Note sybmigrate runs only on Adaptive Server 12.5.0.1 and later.

Benefits of *sybmigrate*

sybmigrate:

- Aids users in changing the page sizes of their database applications.
- Provides a manageable and smooth migration process.
- Allows customers to take advantage of the variable page size feature for existing databases with user data, thus realizing the full benefit of Adaptive Server versions 12.5 and later.

High availability on Sun Cluster 3.0

Adaptive Server version 12.5.0.1 adds support for high availability for Sun Cluster 3.0.

New character sets

Adaptive Server version 12.5.0.1 adds support for the following character sets:

- roman9
- iso15
- big5hk

System changes

New configuration parameters

Table 5-16 summarizes the new configuration parameters in version 12.5.0.1.

Table 5-16: New configuration parameters

Parameter	Function
number of histogram steps	Specifies the number of histogram steps. If statistics do not exist on the column, the default value is 20. If statistics for a column already exist in sysstatistics, the default value is the current number of steps.

New and changed Transact-SQL commands

Table 5-17 summarizes the new Transact-SQL commands in version 12.5.0.1.

Table 5-17: New commands

Command	Description
grant dbcc	Allows the System Administrator to grant access on certain dbcc commands.
revoke dbcc	Allows the System Administrator to revoke access on some dbcc commands.

Table 5-18 summarizes the changes to existing Transact-SQL commands in version 12.5.0.1.

Table 5-18: Changed commands

Command	Change
insert	You can use the insert command to create a new file directory. To do so, use only the filename, filetype, and content columns. Specify "DIR" as the filetype, to create filename as a directory.

Changed system procedures

Table 5-19 summarizes the changes made to existing system procedures.

Table 5-19: Changed system procedures

Procedure	Change
sp_helptext	Adds the number parameter, which is an integer identifying an individual procedure, when *objname* represents a group of procedures. This parameter tells sp_helptext to display the source text for a specified procedure in the group.
sp_monitorconfig	sp_monitorconfig can now monitor more than 30 resources compared to the 6 resources it monitored in earlier versions.

New utility programs

Table 5-20 summarizes the new utility programs added to Adaptive Server version 12.5.0.1.

Table 5-20: New utility programs

Utility	Function
sybmigrate	Enables you to migrate database from a server using 2K logical pages to a server using 4, 8, or 16K logical pages.

Changes to databases and system tables

There are no new databases or system tables for Adaptive Server 12.5.0.1.

Index

A

access control 46
accessing query results as objects 101
accountability 47
active-active 83
  failover in Sun Clusters 83
active-passive
  failover in Sun Clusters 83
  fall back in Sun Clusters 83
  Sun clusters 82
Adaptive Server Enterprise Web Services 71
Adaptive Server Plug-in 9
  Interactive SQL 9
aggressive garbage collection 85
allocation
  space 34
ASE Replicator 100
ASE Web Services 71
asynchronous log service 83
@@lock_timeout 94
@@max_precision 94
automatic database expansion 58
automatic **update statistics** **11**

B

backward compatibility 72
big int support 6
Bourne shell xiii
buffer pool, larger 89
bytes, converting to Unicode 103

C

C shell xiii
cache wizard, for **sp_sysmon** 70
changed Transact-SQL commands 19
character processing engine, XQL 102
checkpoint command 74
checkpoints 87
checkstorage 87
Chinese character set 86
client connection
  failover 83
  Sun Clusters failover 83
cluster
  active-active setup 82
  active-passive setup 82
  synchronization 100
commands
  disk resize 99
Component Integration Services (CIS) 88, 100, 104
compressed database dumps
  syntax 49
computed column
  deterministic 5
  differences from function-based index 5
  indexability 5
  materialized or non-materialized 5
computed columns 5
  materialized and non materialized differences 5
configuration parameter
  deadlock pipe active 91
  deadlock pipe max messages 91
  dtm detach timeout period 85
  dynamic allocation on demand 90
  enable ldap user auth 72
  enable monitoring 91
  errorlog pipe active 91
  errorlog pipe max messages 91
  housekeeper free write percent 85
  max native threads per engine 34
  max sql text monitored 91
  number of large io buffers increased 90
  number of network listeners 61
  number of user connections 73
  object lockwait timing 91

per object statistics active 91
plan text pipe active 91
plan text pipe max messages 91
process wait events 91
rtm thread idle wait period 34
SQK text pipe active 91
sql batch active 91
sql text pipe max messages 91
statement pipe active 91
statement pipe messages 91
statement statistics active 91
wait even timing 91
configuration parameters
enable metrics capture 17
enable semantic partitioning 18
enable web services 18
enable xml 18
max native threads per engine 18
max partition degree 18
max resource granularity 18
number of devices 18
number of dump threads 18
number of open partitions 18
optimization goal 18
optimization timeout limit 18
rtm thread idle wait period 18
statement cache size 18
sysstatistics flush interval 18
create database command 92
.cshrc file xiii
cursors, scrollable 6

D

data in native character set 103
database dumps
compressed 49
password-protected 48
database expansion 58
datatypes
big int 6
date 63
time 63
unsigned int 7
datatypes, unitext 6
dbcc command 92
described 99
discretionary access control 99
delete statistics syntax 47
deterministic property
computed column 5
function-based index 5
disconnection
user 35
discretionary access control of **dbcc** commands 99
disk resize 99
document heading, XML 103
DOM
getting results as 101
dtdValidate 35
dtm detach timeout period configuration parameter 85
dump and load database
across platforms 32
dump database syntax 48, 49
dynamic listeners 61

E

EJB server
reserved sockets 35
enable job scheduler configuration parameter 72
enable ldap user auth configuration parameter 66, 72
enable metrics capture configuration parameter 17
enable semantic partitioning configuration parameter 18
enable web services configuration parameter 18
enable xml configuration parameter 18
encryption 48
FIPS certified algorithms for SSL encryption 48
engine
query processing 102
query, XQL 102
XQL character processing 102
Enhanced Full-Text Search
collections 42
environment variable 42
INDENTITY column 43
index_any 43
installation directory 42

installing 42
primary keys 43
searching documents 43
shutdown permissions 42
style files directory 42
enhanced login controls 46
extracting SQL base types
from XML document 101

F

fallback in Sun Clusters 83
fbo. *See* HP-UX Itanium
feedback optimization
HP-UX Itanium 33
files
.cshrc xiii
.profile xiii
FIPS 140-2 and SSL 16
for xml clause
generating non-ASCII data 9
function based indexes 5
function-based index
deterministic property 5
different from computed column 5
indexability 5
materialized 5
functions
current_date 74
current_time 74
day 95
identity_burn_max 95
left 95
len 95
month 95
newid 95
next_identity 95
square 95
str_replace 95
tempdb_id 95
year 95
functions, changed 21
functions, new 20

G

getDOM() Java method 101
getDouble() Java method 101
getFloat() Java method 102
getInt() Java method 101
getString() Java method 101
getXML(), Java method 101
global variables 89, 94, 97
89, 89
@@datefirst 94, 97
@@lock_timeout 94, 97
@@max_precision 94, 97
@@tempdbid 94
global variables, new and changed 29
grant dbcc command 106
granting and revoking permissions for users and roles
47

H

high availability, on Sun Cluster 3.5 67
Historical Server 32
housekeeper
chores 85
configuration parameter license information 85
wash 85
housekeeper free write percent configuration
parameter 85
HP-UX 33
HP-UX Itanium
feedback optimization 33
XML 33

I

identity values and integer identity 8
import statistics
proxy tables 32
increase, configuration parameter **number of large io**
buffers 90
index_any
length of 43
indexability
computed column 5

function-based index 5
indexes
Unicode-compliant 102
insert command 106
Component Integration Services 104
installdbextend script 59
installing EFTS 42
integer identity, datatypes as identity values 8
Interactive SQL 10
isFirst() Java method 102
isLast() Java method 102

J

Java class
com.sybase.xml.xql.XqlResults.SybXqlResult 101
Java exception
double 101, 102
Java method
getXML() 101
java.io.InputStream 103
java.lang.String 103
query(), invoking with literal 103
getDOM() 101
getDouble() 101
getFloat() 102
getInt() 101
getString() 101
isFirst() 102
isLast() 102
next 101
reset() 102
setEncoding 103
Java methods in result set 101
java.io.InputStream 102
java.lang.String 102
Job Scheduler 69

K

Kerberos 44
CyberSafe Kerberos libraries 44
licenses 45
MIT Kerberos libraries 44
Native libraries 44
keywords
new for release 12.5 28
Korn shell xiii

L

languages 36
large I/O buffers, improvement 90
large identifiers 4
large memory support for Linux 50
larger buffer pool 89
lazy garbage collection 85
LDAP user authentication 45
creating logins 45
password management 45
license information, configuration parameter 85
limiting **set proxy** use for roles 46
Linux
large memory support 50
operating system 50
list 2
load database syntax 48

M

materialized or non-materialized
function-based index 5
max native threads per engine configuration parameter 18, 34
max partition degree configuration parameter 18
max resource granularity configuration parameter 18
memory allocation 58
migration 36
mon_role 81
monCachedObject system table 96
monCachedProcedures system table 96
monCachePool system table 96
monDataCache system table 95
monDeadLock system table 96
monDeviceIO system table 96
monEngine system table 95
monErrorLog system table 96
monIOQueue system table 96

monitor counter concurrency 37
monitoring indexes 82
monitoring tables 80–81
  CIS and 81
  mon_role 81
  Using Transact-SQL to monitor performance 81
monitoring tables, changed 28
monitoring tables, new 28
monLocks system table 96
monNetworkIO system table 96
monOpenDatabase system table 95
monOpenObjectActivity system table 96
monProcedureCache system table 95
monProcess system table 96
monProcessActivity system table 96
monProcessLookup system table 96
monProcessNetIO system table 96
monProcessObject system table 96
monProcessProcedures system table 96
monProcessSQLText system table 96
monProcessStatement system table 96
monProcessWait system table 96
monState system table 95
monSysPlanText system table 96
monSysSQLText system table 96
monSysStatement system table 96
monSysWait system table 96
monSysWorkerThread system table 95
monTableColumns system table 95
monTableParameters system table 95
monTables system table 95
monWaitClassinfo system table 96
monWaitEventinfo system table 96
mount Transact-SQL command 73
multiple temporary databases. See tempdbs.

N

native character sets
  of data 103
  querying in 102
native code sets
  converting results back to 103
native threads
  sp_configure 33
native XML 65
native XML processor 64
new global variables 94, 97
new Transact-SQL commands 19
next Java method 101
non-ASCII data
  generating in **for xml** clause 9
  processing in XML documents and queries 9
  storing in **xmlparse** 9
number of databases per transaction 90
number of devices configuration parameter 18
number of dump threads configuration parameter 18
number of histogram steps configuration parameter 106
number of open partitions configuration parameter 18
number of user connections configuration parameter 73

O

objects
  accessing query results as 101
optimistic index locking 83
optimization goal configuration parameter 18
optimization timeout limit configuration parameter 18
option packages 98
  Content Management Edition 99
  descriptions 99
  e-Business Edition 99
  Security and Directory Services Edition 99
  XML Management edition 99
overview of cluster synchronization 100

P

parallelism in Adaptive Server 3
parser, XML, Xerces 103
partitioning 2
  and query processor 3
  hash 2
  range 2
  round-robin 2

partitioning support 2
password-protected database dumps 48
performance improvements 89
Pluggable Authentication Module (PAM) 45
primary keys 43
.profile file xiii
proxy table 103
proxy tables
  import statistics 32
  update statistics 32

Q

queries
  in native character sets 102
  XML documents 102
query engine
  XQL 102
query functions
  table of 66
query plans 3
  statement cache storage 39
query processing metrics 13
query processor 3
query results, accessing as objects 101

R

Real Time Messaging Services
  overview 40
replication
  with ASE Replicator 100
repositories, moving information with synchronization 101
reserved sockets
  EJB server 35
reserved words, new 28
reset() Java method 102
result set
  Java methods in 101
results
  converting to native code sets 103
  obtaining in Java string 103
revoke dbcc command 106
row-locked catalogs 3
row-locked system catalogs 3
rtm thread idle wait period configuration parameter 18, 34

S

sample code
  for XML results 103
sampling
  statistics 84
scrollable cursors 6
Secure socket layer (SSL) and FIPS 140-2 16
security
  access control 43
  accountibility 43
  encryption technologies 43
  identification and authentication 43
  Kerberos 44
select command 74
select into command 89
set bulk array size 88
set bulk batch size 88
set command
  identity_update 90
set identity_update command 92
set options, new 20
set tablesyntax 46
shells xiii
showplan changes 14
shutdown permissions
  for EFTS 42
source Adaptive Server, for migration 104
sp_addlogin system procedure 67
sp_addserver system procedure 75
sp_bindcache system procedure 75, 93
sp_cacheconfig system procedure 75
sp_chgattribute system procedure 93
sp_configure
  native threads 33
sp_dbcc_checkverify system procedure 93
sp_dbcc_configreport system procedure 93
sp_dbcc_exclusions system procedure 94
sp_dbcc_faultreport system procedure 75, 93

sp_dbcc_help_fault system procedure 94
sp_dbcc_recommendations system procedure 94
sp_dbcc_summary_report system procedure 93
sp_dbcc_updateconfig system procedure 93
sp_dbccevaluatedb system procedure 75
sp_dbextend system procedure 58, 75
sp_dboption system procedure 75, 93
sp_help system procedure 93
sp_helpdb system procedure 93
sp_helptext system procedure 93, 106
sp_ldapadmin system procedure 75
sp_listener system procedure 61, 75
sp_monitorconfig system procedure 86, 92
sp_serveroption system procedure 75
sp_sysmon system procedure 75
sp_tempdb system procedure 93
sp_unbindcache system procedure 75
space
  allocation 34
SQL
  base types, extracting from XML document 101
  converting XML to 101
  derived tables 62
  extensions, query functions as 65
SQL compatibility 71
SQL mapping functions 86
SQL2002 standard 62
SQLJ aliases for XQL 86
SQLX
  standard 86
  support 63
statement cache 39
statement cache size configuration parameter 18
statement cache, syntax 39
statistics
  sampling 84
Style files
  for EFTS 42
summarizes 27
Sun Clusters 82
  active-active failover 83
  active-active fallback 83
  active-active setup 82
  active-passive fall back 83
  acttive-passive failover 83
  client connection failover 83
Sybase Central 9
SYBASE_FTS 42
sybmgmtdb database 69
sybmigrate 36
sybmigrate
  Adaptive Server version level 105
  source Adaptive Server 104
  target Adaptive Server 104
sybmigrate utility program 94, 107
synchronization
  moving information between repositories 101
  overview 100
syntax
  dump database 48
  load database 48
syntax conventions xiii
sysattributes system table 97
sysdatabases system table 97
sysstatistics flush interval configuration parameter 18
system procedures, changed 22
system procedures, new 22
system tables, changed 24
system tables, changed status 27
system tables, new 24
system tables, row-locked 3

T

tables
  proxy 103
target Adaptive Server, for migration 104
tempdbs
  and the system *tempdb* 85
  described 84
tempdbs
  described 84
There 4
three housekeeeepers 85
top n functionality 36
truncate table syntax 47

U

Unicode parser 62
Unicode-compliant indexes 102
unitext support 6
unmount Transact-SQL command 73
unsigned int support 7
update statistics 84
  proxy tables 32
update statistics automatic 11
update statistics command 92
update statistics syntax 47
updating statistics
  use sampling 84
updating statistics 84
upgrade, incorporating proxy tables 104
user
  disconnection 35
User-defined web services 10
UTF-8 sort order 62
utility programs, changed 23

V

variables
  global, new 97
Verity style files 42
version for Adaptive Server on **sybmigrate** 105
very lage storage support (VLSS) 10
virtual columns 5

W

Web services 71
  with Adaptive Server 71

X

Xerces parser 103
XML
  converting to SQL data 101
  document heading 103
  document, character encoding 103
  document, extracting SQL base types from 101
  document, querying 102
  for xml 8
  getting results as DOM 101
  HP-UX Itanium 33
  internationalization support 8
  iterating through document 101
  parser, accepting bytes 102
  query mapping 65
  results, sample code 103
  schema support 8
XML enhancements 8
XML processor 64
xmlextract
  built-in query function 66
xmlextract, **xmltest**, for processing XML documents containing non-ASCII data 9
xmlparse
  built-in function 66
xmlparse, storing non-ASCII data 9
xmlrepresentation
  built-in function, determines parsed image column 66
xmltext
  SQL predicate, returns boolean result 66
XQL
  character processing engine 102
  query engine 102