SYBASE®

An **SAP** Company

System Administration Guide:
Volume 1

Adaptive Server® Enterprise

15.7

Contents

Overview of System Administration

This chapter introduces the basic topics of Adaptive Server® system administration.

Topic	Page
Adaptive Server administration tasks	1
System tables	7
System procedures	10
System extended stored procedures	12
Logging error messages	13
Connecting to Adaptive Server	14
Security features available in Adaptive Server	18

Adaptive Server administration tasks

Adaptive Server administration tasks include:

- Installing Adaptive Server and Backup Server

- Creating and managing Adaptive Server login accounts

- Granting roles and permissions to Adaptive Server users

- Managing and monitoring the use of disk space, memory, and connections

- Backing up and restoring databases

- Diagnosing system problems

- Configuring Adaptive Server to achieve the best performance

In addition, system administrators may assist with certain database design tasks that overlap with the work of application designers, such as enforcing integrity standards.

Although a system administrator generally concentrates on tasks that are independent of the applications running on Adaptive Server, he or she is likely to have the best overview of all applications. For this reason, a system administrator can advise application designers about the data that already exists on Adaptive Server, make recommendations about standardizing data definitions across applications, and so on.

However, the distinction between what is specific to an application is sometimes unclear. Owners of user databases may consult certain sections of this book. Similarly, system administrators and database owners will use the *Transact-SQL Users Guide* (especially the chapters on data definition, stored procedures, and triggers). Both system administrators and application designers will use the *Performance and Tuning Series*.

Roles required for system administration tasks

Many of the commands and procedures discussed in this manual require the system administrator or system security officer role. Other sections in this manual are relevant to database owners.

Various security-related, administrative, and operational tasks are grouped into the following user roles:

- **system administrator** – by default, the system administrator (sa) is assigned these roles:

    - sa_role

    - sso_role

    - oper_role

    - sybase_ts_role

    The system administrator's tasks include:

    - Managing disk storage

    - Monitoring the Adaptive Server automatic recovery procedure

    - Fine-tuning Adaptive Server by changing configurable system parameters

    - Diagnosing and reporting system problems

    - Backing up and loading databases

    - Modifying and dropping server login accounts

- • Granting and revoking the system administrator role

- • Granting permissions to Adaptive Server users

- • Creating user databases and granting ownership of them

- • Setting up groups, which can be used for granting and revoking permissions

- **System security officer** – performs security-related tasks, such as:

  - • Creating server login accounts, which includes assigning initial passwords

  - • Changing the password of any account

  - • Granting and revoking the system security officer and operator roles

  - • Creating, granting, and revoking user-defined roles

  - • Granting the capability to impersonate another user throughout the server

  - • Setting the password expiration interval

  - • Setting up Adaptive Server to use network-based security services

  - • Managing the audit system

- **Operator** – backs up and loads databases on a server-wide basis. The operator role allows a single user to use the dump database, dump transaction, load database, and load transaction commands to back up and restore all databases on a server without having to be the owner of each one. These operations can be performed for an individual database by the database owner or by a system administrator. However, an operator can perform them for any database.

These roles provide individual accountability for users performing operational and administrative tasks. Their actions can be audited and attributed to them. A system administrator operates outside the discretionary access control (DAC) protection system; that is, when a system administrator accesses objects, Adaptive Server does not check the DAC permissions.

In addition, two kinds of object owners have special status because of the objects they own. These ownership types are:

- • Database owner

- • Database object owner

Database owner

The **database owner** is the creator of a database or someone to whom database ownership has been transferred. A system administrator can use the grant command to grant users the authority to create databases.

A database owner logs in to Adaptive Server using his or her assigned login name and password, and has the "dbo" account. When this user logs in to databases they did not create, this user is known by his or her regular user name.

A database owner can:

- Run the system procedure sp_adduser to allow other Adaptive Server users access to the database

- Use the grant command to give other users permission to create objects and execute commands within the database

Adding users to databases is discussed in Chapter 3, "Managing Adaptive Server Logins and Database Users" in the *Security Administration Guide*. Granting permissions to users is discussed in Chapter 6, "Managing User Permissions."

The database owner does not automatically receive permissions on objects owned by other users. However, a database owner can temporarily assume the permissions of other users in the database at any time by using the setuser command. Using a combination of the setuser and grant commands, the database owner can acquire permissions on any object in the database.

Note Because the database owner role is so powerful, the system administrator should plan carefully who should own databases in the server. The system security officer should consider auditing the database activity of all database owners.

Database object owner

A **database object owner** is a user who creates a database object. **Database objects** include tables, indexes, views, defaults, triggers, rules, constraints, and procedures. Before a user can create a database object, the database owner must grant the user permission to create objects of a particular type. There is no special login name or password for a database object owner.

The database object owner creates an object using the appropriate create statement, and then grants permission to other users.

The creator of a database object is automatically granted all permissions on that object. The system administrator also has all permissions on the object. The owner of an object must explicitly grant permissions to other users before they can access the object. Even the database owner cannot use an object directly unless the object owner grants him or her the appropriate permission. However, the database owner can always use the setuser command to impersonate any other user in the database, including the object owner.

Note When a database object is owned by someone other than the database owner, the user (including a system administrator) must qualify the name of that object with the object owner's name—*ownername.objectname*—to access the object. If an object or a procedure must be accessed by a large number of users, particularly in ad hoc queries, having these objects owned by "dbo" greatly simplifies access.

Using *isql* to perform system administration tasks

This book assumes that you use the commmand line utility isql to perform the system administration tasks described in this guide. This section provides some basic information about using isql. For complete information, see the *Utility Guide*.

You can also use the graphic tool Sybase® Central™ to perform many of the tasks described in this book, as described in "Using Sybase Central for system administration tasks" on page 6.

Starting *isql*

To start isql on most platforms, type this command at an operating system prompt, where *username* is the system administrator:

```
isql -Uusername
```

Adaptive Server prompts you for your password.

Note Do not use the -P option of isql to specify your password; another user might then see your password.

You can use isql in command line mode to enter many of the Transact-SQL examples in this manual.

Entering statements

The statements that you enter in isql can span several lines. isql does not process statements until you type "go" on a separate line. For example:

```
1> select *
2> from sysobjects
3> where type = "TR"
4> go
```

The examples in this manual do not include the go command between statements. If you are typing the examples, you must enter the go command to see the sample output.

Saving and reusing statements

This manual frequently suggests that you save the Transact-SQL statements you use to create or modify user databases and database objects. The easiest way to do this is to create or copy the statements to an ASCII-formatted file. You can then use the file to supply statements to isql to re-create databases or database objects later.

The syntax for using isql with an ASCII-formatted file is the following, where *filename* is the full path and file name of the file that contains Transact-SQL statements:

```
isql -Uusername -ifilename
```

On UNIX and other platforms, use the "less than" symbol (<) to redirect the file.

The Transact-SQL statements in the ASCII file must use valid syntax and the go command.

When reading commands from a file, you must:

- Supply the -P*password* option at the command line, or,

- Include the named user's password on the first line of the input file.

Using Sybase Central for system administration tasks

You can perform many system administration tasks using Sybase Control Center and Sybase Central, a graphic tool that comes with Adaptive Server:

- Initializing database devices

- Setting configuration parameters

- Viewing the amount of free log space in a database

- Generating data definition language (DDL)

- Creating logins

- Adding remote servers

- Creating databases

- Creating stored procedures

- Defining roles

- Adding data caches

- Setting database options

- Backing up and restoring databases

Use the Monitor Viewer feature of Sybase Central to access Adaptive Server Monitor™. Sybase Central includes extensive online help.

Use the Sybase Central DDL-generation feature to record your work to Transact-SQL scripts. The DDL-generation feature lets you save to a script the actions you perform in an entire server or within a specific database.

System tables

The master database contains **system tables** that keep track of information about Adaptive Server. In addition, each database (including the master database) contains system tables that keep track of information specific to that database.

All the Adaptive Server-supplied tables in the master database (the Adaptive Server controlling database) are considered system tables. Each user database is created with a subset of these system tables. The system tables may also be called the **data dictionary** or the system catalogs.

A master database and its tables are automatically created when Adaptive Server is installed. The system tables in a user database are created when the create database command is issued. The names of all system tables start with "sys". You cannot create tables in user databases that have the same names as system tables. See *Reference Manual: Tables* for detailed descriptions of system tables and their columns.

Querying the system tables

Query system tables in the same manner as any other tables. For example, the following statement returns the names of all triggers in the database:

```
select name
from sysobjects
where type = "TR"
```

In addition, Adaptive Server supplies **stored procedure***s* (called **system procedures**), many of which provide shortcuts for querying the system tables.

These system procedures provide information from the system tables:

• sp_commonkey	• sp_helpremotelogin
• sp_configure	• sp_help_resource_limit
• sp_countmedatada	• sp_helprotect
• sp_dboption	• sp_helpsegment
• sp_estspace	• sp_helpserver
• sp_help	• sp_helpsort
• sp_helppartition	• sp_helptext
• sp_helpcache	• sp_helpthreshold
• sp_helpconfig	• sp_helpuser
• sp_helpconstraint	• sp_lock
• sp_helpdb	• sp_monitor
• sp_helpdevice	• sp_monitorconfig
• sp_helpgroup	• sp_showcontrolinfo
• sp_helpindex	• sp_showexeclass
• sp_helpjava	• sp_showplan
• sp_helpjoins	• sp_spaceused
• sp_helpkey	• sp_who
• sp_helplanguage	• sp_help_resource_limit
• sp_helplog	

For complete information about the system procedures, see the *Reference Manual: Procedures*.

Keys in system tables

Primary, foreign, and common keys for system tables are defined in the master and model databases. You can generate a report on defined keys by executing sp_helpkey. For a report on columns in two system tables that are likely join candidates, execute sp_helpjoins.

The *Adaptive Server System Tables Diagram* shows the relationships between columns in the system tables.

Updating system tables

The Adaptive Server system tables contain information that is critical to the operation of your databases. Under ordinary circumstances, you need not perform direct data modifications to system tables.

Update system tables only when you are instructed to do so by Sybase Technical Support, by an instruction in the *Error Messaging and Troubleshooting Guide*, or in this manual.

Before you update system tables, you must issue an sp_configure command that enables system table updates. While this command is in effect, any user with appropriate permission can modify a system table. Other requirements for direct changes to system tables are:

* Modify system tables only inside a transaction. Issue a begin transaction command before you issue the data modification command.

* Verify that only the rows you wanted changed have been affected by the command, and that the data has been changed correctly.

* If the command was incorrect, issue a rollback transaction command. If the command was correct, issue a commit transaction command.

> **Warning!** Some system tables should not be altered by any user under any circumstances. Some system tables are dynamically built by system processes, contain encoded information, or display only a portion of their data when queried. Imprudent, ad hoc updates to certain system tables can prevent Adaptive Server from running, make database objects inaccessible, scramble permissions on objects, or terminate a user session. Moreover, never attempt to alter the definition of the system tables in any way. For example, do not alter system tables to include constraints. Triggers, defaults, and rules are not allowed in system tables. If you create a trigger, bind a rule, or default to a system table, you see an error message.

System procedures

The names of all system procedures begin with "sp_". They are located in the sybsystemprocs database, but you can run many of them in any database by issuing the stored procedure from the database or by qualifying the procedure name with the database name.

Sybase-supplied system procedures (such as sp_who) are created using the *installmaster* installation script. Use sp_version to determine the version of *installmaster* was most recently executed. See the *Reference Manual: System Procedures* for more information about sp_version.

If you execute a system procedure in a database other than sybsystemprocs, the procedure operates on the system tables in the database from which it was executed. For example, if the database owner of pubs2 runs sp_adduser from pubs2 or issues the command pubs2..sp_adduser, the new user is added to pubs2..sysusers. However, this does not apply to system procedures that update only tables in the master database.

Permissions on system procedures are discussed in the *Reference Manual: Procedures*.

Using system procedures

A **parameter** is an argument to a stored or system procedure. If a parameter value for a system procedure contains reserved words, punctuation, or embedded blanks, you must enclose it in single or double quotes. If the parameter is an object name, and the object name is qualified by a database name or owner name, enclode the entire name in single or double quotes.

System procedures can be invoked during a session using either chained or unchained transaction mode. Chained mode implicitly begins a transaction before any data retrieval or modification statement. Unchained mode requires explicit begin transaction statements paired with commit transaction or rollback transaction statements to complete the transaction. See Chapter 21, "Transactions: Maintaining Data Consistency and Recovery," in the *Transact-SQL Users Guide*.

You cannot execute the system procedures that modify data in system tables in the master database from within a transaction, since this may compromise recovery. You cannot run system procedures that create temporary worktables from transactions.

If no transaction is active when you execute a system procedure, Adaptive Server turns off chained mode and sets transaction isolation level 1 for the duration of the procedure. Before returning, the session's chained mode and isolation level are reset to their original settings. See Chapter 21, "Transactions: Maintaining Data Consistency and Recovery," in the *Transact-SQL Users Guide*.

All system procedures report a return status. For example, the following means that the procedure executed successfully:

```
return status = 0
```

If the system procedures do not execute successfully, the return status is a number other than 0.

System procedure tables

The system procedures use several **system procedure tables** in the master and sybsystemdb databases to convert internal system values (for example, status bits) into human-readable format. One of these tables, spt_values, is used by a variety of system procedures, including:

- sp_configure
- sp_dboption
- sp_depends
- sp_help
- sp_helpdb
- sp_helpdevice
- sp_helpindex
- sp_helpkey
- sp_helprotect
- sp_lock

The spt_values table can be updated only by an upgrade; you cannot modify it. To see how it is used, execute sp_helptext and look at the text for one of the system procedures that references it.

The other system procedure tables are spt_monitor, spt_committab, and tables needed by the catalog stored procedures. (The spt_committab table is located in the sybsystemdb database.)

In addition, several system procedures create, and then drop, temporary tables. For example, sp_helpdb creates #spdbdesc, sp_helpdevice creates #spdevtab, and sp_helpindex creates #spindtab.

Creating system procedures

Many system procedures are explained in this manual, in the sections where they are relevant. For detailed reference information, see the *Reference Manual: Procedures*.

System administrators can write system procedures that can be executed in any database. Create a stored procedure in sybsystemprocs and assign it a name that begins with "sp_". The uid of the stored procedure must be 1, the uid of the database owner.

Most system procedures that you create query the system tables. Sybase recommends that you do not create stored procedures that modify the system tables.

To create a stored procedure that modifies system tables, a system security officer must first turn on the allow updates to system tables configuration parameter. Any stored procedure created while this parameter is set on can always update system tables, even when allow updates to system tables is turned off. To create a stored procedure that updates the system tables:

1   Use sp_configure to set allow updates to system tables on.

2   Use create procedure to create the stored procedure.

3   Use sp_configure to set allow updates to system tables off.

Warning! Use caution when you modify system tables. Always test the procedures that modify system tables in development or test databases, rather than in your production database.

System extended stored procedures

An extended stored procedure (ESP) letes you call external language functions from within Adaptive Server. Adaptive Server includes a set or predefined ESPs; users can also create their own. The names of all system extended stored procedures begin with "xp_", and are located in the sybsystemprocs database.

One very useful system ESP is xp_cmdshell, which executes an operating system command on the system that is running Adaptive Server.

Invoke a system ESP just like a system procedure. The difference is that a system ESP executes procedural language code rather than Transact-SQL statements. All ESPs are implemented by an Open Server™ application called XP Server™, which runs on the same machine as Adaptive Server. XP Server starts automatically on the first ESP invocation.

For information about the system ESPs provided with Adaptive Server, see the *Reference Manual: Procedures*.

Creating system ESPs

Use create procedure to create a system ESP in the sybsystemprocs database. System procedures are automatically included in the sybsystemprocs database. The name of the ESP, and its procedural language function, must begin with "xp_". The uid of the stored procedure must be 1, the uid of the database owner.

For general information about creating ESPs see Chapter 17, "Using Extended Stored Procedures," in the *Transact-SQL Users Guide*.

Logging error messages

Each time it starts, Adaptive Server writes start-up information to a local error log file. The installation program automatically sets the error log location when you configure a new Adaptive Server. See the *Configuration Guide* for your platform to learn the default location and file name of the error log.

Many error messages from Adaptive Server are written only to the user's terminal. However, fatal error messages (severity levels 19 and above), kernel error messages, and informational messages from Adaptive Server are recorded in the error log file.

Adaptive Server keeps the error log file open until you stop the server process. Before deleting old messages to reduce the size of the error log, stop the Adaptive Server process.

Note On some platforms, such as Windows, Adaptive Server also records error messages in the operating system event log. See the installation guide and configuration guide for your platform.

Connecting to Adaptive Server

Adaptive Server can communicate with other Adaptive Servers, Open Server applications, and client software on the network. Clients can talk to one or more servers, and servers can communicate with other servers using remote procedure calls. For products to interact with one another, each must know where the others reside on the network. This network service information is stored in the *interfaces* file.

The *interfaces* file

The *interfaces* file is usually named *interfaces*, *interface*, or *sql.ini*, depending on the operating system.

The *interfaces* file lists the name and address of every known server. When you use a client program to connect to a server, the program looks up the server name in the *interfaces* file and then connects to the server using the address, as shown in Figure 1-1.

Figure 1-1: Connecting to Adaptive Server
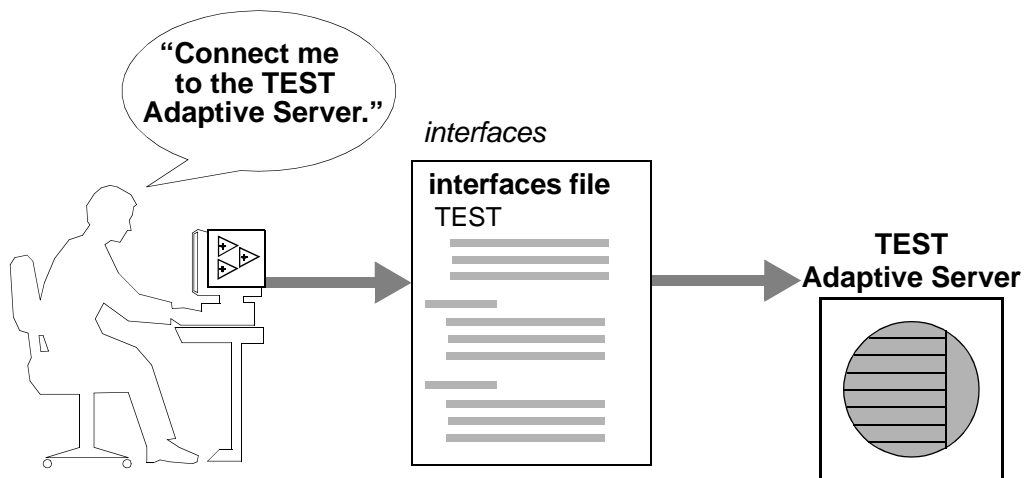
The name, location, and contents of the *interfaces* file differ between operating systems. Also, the format of the Adaptive Server addresses in the *interfaces* file differs between network protocols.

The Adaptive Server installation program creates a simple *interfaces* file that you can use for local connections to Adaptive Server over one or more network protocols. As a system administrator, modify the *interfaces* file and distribute it to users so that they can connect to Adaptive Server over the network. See the *Configuration Guide* for your platform for information about the *interfaces* file.

See Chapter 2, "Networks and Performance" in the *Performance and Tuning Series: Basics* for more information about the interfaces file and network listeners.

Directory services

A directory service manages the creation, modification, and retrieval of network service information. Directory services are provided by platform or third-party vendors and must be purchased and installed separately from Adaptive Server. Two examples of directory services are Registry and Distributed Computing Environment (DCE).

The *$SYBASE/$SYBASE_OCS/config/libtcl.cfg* file is a Sybase-supplied configuration file used by servers and clients to determine:

* Which directory service to use, and

* The location of the specified directory service driver.

If no directory services are installed or listed in the *libtcl.cfg* file, Adaptive Server defaults to the *interfaces* file for obtaining network service information.

The system administrator must modify the *libtcl.cfg* file as appropriate for the operating environment.

Some directory services are specific to a given platform; others can be used on several different platforms. Because of the platform-specific nature of directory services, see the configuration documentation for your platform for detailed information about configuring for directory services.

LDAP as a directory service

Lightweight Directory Access Protocol (LDAP) is an industry standard for accessing directory services. Directory services allow components to look up information by a distinguished name (DN) from an LDAP server that stores and manages server, user, and software information that is used throughout the enterprise or over a network.

The LDAP server can be located on a different platform from the one on which Adaptive Server or the clients are running. LDAP defines the communication protocol and the contents of messages exchanged between clients and servers. Messages are operators, such as client requests for read, write and query, and server responses, including metadata (data about data).

The LDAP server can store and retrieve information about:

- Adaptive Server, such as IP address, port number, and network protocol

- Security mechanisms and filters

- High availability companion server name

- Authentication information for user access to Adaptive Server

  You can authenticate users logging in to Adaptive Server through information stored in the *syslogins* directory or through a centralized LDAP server that enables a single login and password throughout the enterprise. See Chapter 3, "Managing Adaptive Server Logins and Database Users," in the *Security Adiminstration Guide*.

You can configure the LDAP server to use these access restrictions:

- Anonymous authentication – all data is visible to any user.

- User name and password authentication – Adaptive Server uses the default user name and password from the appropriate file:

  - UNIX, 32-bit – *$SYBASE/$SYBASE_OCS/config/libtcl.cfg*

  - UNIX, 64-bit – *$SYBASE/$SYBASE_OCS/config/libtcl64.cfg*

  - Windows – *%SYBASE%\%SYBASE_OCS%\ini\libtcl.cfg*

User name and password authentication properties establish and end a session connection to an LDAP server.

Note The default user name and password stored in *libtcl.cfg* and passed to the LDAP server for authentication purposes are distinct and different from those used to access Adaptive Server. The default user name and password allow access to the LDAP server for administrative tasks.

When an LDAP server is specified in the *libtcl.cfg* or *libtcl64.cfg* file (collectively called *libtcl*.cfg* file), the server information is then accessible only from the LDAP server; Adaptive Server ignores the *interfaces* file.

If multiple directory services are supported in a server, the order in which they are searched is specified in *libtcl*.cfg*. You cannot use the dataserver command line option to specify the search order.

Multiple directory services

Any type of LDAP service, whether it is an actual server or a gateway to other LDAP services, is called an LDAP server.

You can specify multiple directory services for high-availability failover protection in *libtcl*.cfg*. Not every directory service in the list must be an LDAP server.

In the following example, if the connection to *test:389* fails, the connection fails over to the DCE driver with the specified directory information tree (DIT) base. If this also fails, a connection to the LDAP server on *huey:11389* is attempted. Different vendors employ different DIT base formats.

```
[DIRECTORY]
   ldap=libdldap.so ldap://test:389/dc=sybase,dc=com
   dce=libddce.so ditbase=/.:/subsys/sybase/dataservers
   ldap=libdldap.so ldap://huey:11389/dc=sybase,dc=com
```

See the *Open Client Client-Library/C Programmer's Guide* and the *Open Client Client-Library/C Reference Manual*.

LDAP directory services versus the Sybase *interfaces* file

The LDAP driver implements directory services for use with an LDAP server. The LDAP infrastructure provides:

• A network-based alternative to the traditional Sybase *interfaces* file

- A single, hierarchical view of information, including users, software, resources, networks, files, and so on

Table 1-1 highlights the differences between the Sybase *interfaces* file and an LDAP server.

Table 1-1: interfaces file versus LDAP directory services

interfaces file	Directory services
Platform-specific	Platform-independent
Specific to each Sybase installation	Centralized and hierarchical
Contains separate master and query entries	One entry for each server that is accessed by both clients and servers
Cannot store metadata about the server	Stores metadata about the server

Performance

Performance when using an LDAP server may be slower than when using an *interfaces* file because the LDAP server requires time to make a network connection and retrieve data. Since this connection is made when Adaptive Server is started, changes in performance are seen at login time, if at all. During normal system load, the delay should not be noticeable. During high system load with many connections, especially repeated connections with short duration, the overall performance difference of using an LDAP server versus the traditional *interfaces* file might be noticeable.

Security features available in Adaptive Server

The major security features available for Adaptive Server are:

- Identification and authentication controls – Ensures that only authorized users can log in to the system. In addition to password-based login authentication, Adaptive Server supports external authentication using Kerberos, LDAP, or pluggable authentication modules (PAM).

- Discretionary access controls (DAC) – Provides access controls that let object owners restrict access to objects, usually with the grant and revoke commands. This type of control is dependent upon an object owner's discretion. .

- Division of roles – Allows an administrator to grant privileged roles to specified users so only designated users can perform certain tasks. Adaptive Server has predefined roles, called "system roles," such as system administrator and system security officer. In addition, Adaptive Server allows system security officers to define additional roles, called "user-defined roles."

- Accountability – Provides the ability to audit events such as logins, logouts, server start operations, remote procedure calls, accesses to database objects, and all actions performed by a specific user or with a particular role active. Adaptive Server also provides a single option to audit a set of server-wide, security-relevant events.

- Confidentiality of data – Maintains a confidentiality of data using encryption for client/server communication, available with Kerberos or secure sockets layer (SSL). Inactive data is kept confidential with password-protected database backup.

See Chapter 2, "Getting Started with Security Administration in Adaptive Server," in the *Security Administration Guide*.

System and Optional Databases

This chapter describes the system databases that reside on all Adaptive Server systems. It also describes optional Sybase-supplied databases that you can install, and the sybdiag database, which Sybase Technical Support may install for diagnostic purposes.

Overview of system databases

A default installation of Adaptive Server includes these system databases:

- The master database

- The model database

- The system procedure database, sybsystemprocs

- The two-phase commit transaction database, sybsystemdb

- The temporary database, tempdb

Optionally, you can install:

- The auditing database, sybsecurity

- The sample databases, pubs2 and pubs3

- The dbcc database, dbccdb

- The Job Scheduler database, sybmgmtdb

For information about installing the master, model, sybsystemprocs, tempdb, and sybmgmtdb databases, see the installation guide for your platform. For information on installing dbccdb, Chapter 10, "Checking Database Consistency," in the *System Administration Guide: Volume 2.* For information about using Job Scheduler, see the *Job Scheduler Users Guide*.

The master, model, sybsystemdb, and temporary databases reside on the master device which is named during installation. The master database is contained entirely on the master device and cannot be expanded onto any other device. Create all other databases and user objects on other devices.

Warning! Do not store user databases on the master device; doing so makes it difficult to recover both the system databases and any user databases stored on the master device.

Install the sybsecurity and sybmgmtdb databases on their own devices and segment. See the installation documentation for your platform.

Install the sybsystemprocs database on a device of your choice. You may want to modify the installation scripts for pubs2 and pubs3 to share the device you create for sybsystemprocs.

Use the *installjsdb* script (located in *$SYBASE/ASE-15_0/scripts*) to install the sybmgmtdb database. *installjsdb* looks for a device named *sybmgmtdev* on which to create the sybmgmtdb database and its accompanying tables and stored procedures. If the sybmgmtdb database already exists, *installjsdb* creates the Job Scheduler tables and stored procedures in the existing database. If *installjsdb* cannot find either a *sybmgmtdev* device or a sybmgmtdb database, it creates sybmgmtdb on the master device. However, Sybase strongly recommends that you remove the sybmgmtdb database from the master device.

The *installpubs2* and the *installpubs3* scripts do not specify a device in their create database statement, so they are created on the default device. During installation, the master device is the default device. To change this, either edit the scripts or follow the instructions in Chapter 8, "Initializing Database Devices."

master **database**

The master database controls the operation of Adaptive Server and stores information about all user databases and their associated database devices. Table 2-1 describes the information that the master database stores.

Table 2-1: Information the master database stores

Information	System table
User accounts	syslogins
Remote user accounts	sysremotelogins
Remote servers that this server can interact with	sysservers
Ongoing processes	sysprocesses
Configurable environment variables	sysconfigures
System error messages	sysmessages
Databases on Adaptive Server	sysdatabases
Storage space allocated to each database	sysusages
Tapes and disks mounted on the system	sysdevices
Active locks	syslocks
Character sets	syscharsets
Languages	syslanguages
Users who hold server-wide roles	sysloginroles
Server roles	syssrvroles
Adaptive Server engines that are online	sysengines

Because the master database stores information about user databases and devices, you must be in the master database to issue the create database, alter database, disk init, disk refit, disk reinit, and disk mirroring commands.

The minimum size of your master database depends on your server's logical page size. The master database must contain at least 6656 logical pages, so its minimum physical size for each logical page size is:

*   2K page – 13MB

*   4K page – 26MB

*   8K page – 52MB

*   16K page – 104MB

Controlling object creation in *master*

When you install Adaptive Server, only a system administrator can create objects in the master database, because the system administrator implicitly becomes "dbo" of any database he or she uses. Any objects created on the master database should be used only for system administration. Set permissions in master so that most users cannot create objects there.

> **Warning!** Do not place user objects in master. Storing user objects in master causes the transaction log to fill quickly. If the transaction log runs out of space completely, you cannot use dump transaction commands to free space in master.

You may also want to use alter login to change the default database for users (the database to which a user is connected when he or she logs in). See Chapter 3, "Managing Adaptive Server Logins and Database Users," in the *Security Administration Guide*.

Create any system procedures in the sybsystemprocs database rather than in master.

Backing up *master* and keeping copies of system tables

To be prepared for hardware or software failure on Adaptive Server:

- Perform frequent backups of the master database and all user databases. See "Keep up-to-date backups of master" on page 40, and Chapter 13, "Restoring the System Databases," in *System Administration Guide: Volume 2*.

- Keep a copy (preferably offline) of these system tables: sysusages, sysdatabases, sysdevices, sysloginroles, and syslogins. See "Keep offline copies of system tables" on page 40. If you have copies of these scripts, and a hard-disk failure or other disaster makes your database unusable, you can use the recovery procedures described in Chapter 13, "Restoring the System Databases," in *System Administration Guide: Volume 2*. If you do not have current copies of your scripts, it is much more difficult to recover Adaptive Server when the master database is damaged.

model **database**

Adaptive Server includes the model database, which provides a template, or prototype, for new user databases. Each time a user enters the create database command, Adaptive Server makes a copy of the model database and extends the new database to the size specified by the create database command.

Note New databases must be at least as large as the model database.

The model database contains the required system tables for each user database. You can modify model to customize the structure of newly created databases— everything you do to model is reflected in each new database. Some of the changes that system administrators commonly make to model are:

- Adding user-defined datatypes, rules, or defaults.

- Adding users who need access to all databases on Adaptive Server.

- Granting default privileges, particularly for "guest" accounts.

- Setting database options such as select into/bulkcopy/pllsort. These settings are reflected in all new databases. The default settings for these options in model is off. See Chapter 9, "Setting Database Options."

Typically, most users do not have permission to modify the model database. There is not much point in granting read permission either, since Adaptive Server copies its entire contents into each new user database.

The model database cannot be larger than tempdb. By default, the size of model is six allocation units (an allocation unit is 256 logical pages.). You see error message if you increase the size of model without making tempdb at least as large.

Note Keep a backup copy of the model database, and back up model with dump database each time you change it. In case of media failure, restore model as you would a user database.

---

sybsystemprocs **database**

Sybase system procedures are stored in the database sybsystemprocs. When a user in any database executes a system stored procedure (that is, a procedure whose name begins with sp_), Adaptive Server first looks for that procedure in the user's current database. If there is no procedure there with that name, Adaptive Server looks for it in sybsystemprocs. If there is no procedure in sybsystemprocs, Adaptive Server looks for the procedure in master.

If the procedure modifies system tables (for example, sp_adduser modifies the sysusers table), the changes are made in the database from which the procedure was executed.

To change the default permissions on system procedures, modify those permissions in sybsystemprocs.

Note Any time you make changes to sybsystemprocs, back up the database.

tempdb **database**

Adaptive Server has a **temporary database**, tempdb, that provides a storage area for temporary tables and other temporary working storage needs. The space in tempdb is shared among all users of all databases on the server.

The default size of tempdb depends on the logical page size for your server, 2, 4, 8, or 16K. Certain activities may make it necessary for you to increase the size of tempdb:

*   Large temporary tables.

*   A lot of activity on temporary tables, which fills up the tempdb logs.

*   Large or many simultaneous sorts. Subqueries and aggregates with group by also cause some tempdb activity.

Use alter database to increase the size of tempdb. tempdb is initially created on the master device. You can add space to tempdb from the master device or from any other database device.

If you run update index statistics against large tables, the command fails with error number 1105 if tempdb is not large enough.

You can create and manage multiple temporary databases in addition to the system temporary database, tempdb. Multiple temporary databases reduce contention on system catalogs and logs in tempdb.

Creating temporary tables

No special permissions are required to create temporary tables or to execute commands that may require storage space in the temporary database.

Create temporary tables either by preceding the table name in a create table statement with a pound sign (#), or by specifying the name prefix "tempdb..".

Temporary tables created with a pound sign are accessible only by the current Adaptive Server session: users on other sessions cannot access them. These nonsharable, temporary tables are destroyed at the end of each session. The first 13 bytes of the table's name, including the pound sign (#), must be unique. Adaptive Server assigns the names of such tables a 17-byte number suffix. (You can see the suffix by querying tempdb..sysobjects.)

Temporary tables created with the "tempdb.." prefix are stored in tempdb and can be shared among Adaptive Server sessions. Adaptive Server does not change the names of temporary tables created this way. The table exists either until you restart Adaptive Server or until its owner drops it using drop table.

System procedures work on temporary tables, but only if you use them from tempdb.

If a stored procedure creates temporary tables, the tables are dropped when the procedure exits. You can also explicitly drop temporary tables before a session ends.

 Warning! Do not create temporary tables with the "tempdb.." prefix from inside a stored procedure unless you intend to share those tables among other users and sessions.

Each time you restart Adaptive Server, it copies model to tempdb, which clears the database. You cannot recover temporary tables.

sybsecurity database

The sybsecurity database, which contains the auditing system for Adaptive Server, includes :

- The system tables, sysaudits_01, sysaudits_02, ... sysaudits_08, which contain the audit trail

- The sysauditoptions table, which contains rows describing the global audit options

- All other default system tables that are derived from model

See Chapter 7, "Auditing," in the *Security Administration Guide*.

sybsystemdb database

The sybsystemdb database stores information about distributed transactions. Adaptive Server versions 12.0 and later can provide transaction coordination services for transactions that are propagated to remote servers using remote procedure calls (RPCs) or Component Integration System (CIS). Information about remote servers participating in distributed transactions is stored in the syscoordinations table.

The sybsystemdb database also stores information about SYB2PC transactions that use the Sybase two-phase commit protocol. The spt_committab table, which stores information about and tracks the completion status of each two-phase commit transaction, is stored in the sybsystemdb database.

See the *Configuration Guide* for your platform for information about two-phase commit transactions and how to create the sybsystemdb database.

sybmgmtdb database

The sybmgmtdb database stores jobs, schedules, scheduled jobs information, and data the internal Job Scheduler task needs for processing. sybmgmtdb also maintains the output and results from these executed tasks. See the *Job Scheduler Users Guide*.

pubs2 and *pubs3* sample databases

Installing the pubs2 and pubs3 sample databases is optional. These databases are provided as a learning tool for Adaptive Server. The pubs2 sample database is used for most of the examples in the Adaptive Server documentation, except for examples, where noted, that use the pubs3 database.

The sample databases are intended for training purposes only. Do not install them in an Adaptive Server production environment.

For information about installing pubs2 and pubs3, see the installation guide for your platform. For information about the contents of these sample databases, see the *Transact-SQL Users Guide*.

Maintaining the sample databases

The sample databases include a "guest" user login that allows access to the database by any authorized Adaptive Server user. The "guest" login has been given a wide range of privileges in pubs2 and pubs3, including permissions to select, insert, update, and delete user tables. See Chapter 3, "Managing Adaptive Server Logins and Database Users," in the *Security Administration Guide*.

The size of the pubs2 and pubs3 databases are determined by the size of the logical page size for your server; 2, 4, 8, and 16K. If possible, give each new user a clean copy of pubs2 and pubs3 so that she or he is not confused by other users' changes. To place pubs2 or pubs3 on a specific database device, edit the installation script before installing the database.

If space is a problem, instruct users to issue the begin transaction command before updating a sample database. After the user has finished updating one of the sample databases, he or she can issue the rollback transaction command to undo the changes.

pubs2 image data

Adaptive Server includes a script for installing image data in the pubs2 database (pubs3 does not use the image data). The image data consists of six pictures, two each in PICT, TIF, and Sun raster file formats. Sybase does not provide any tools for displaying image data. You must use the appropriate screen graphics tools to display the images after you extract them from the database.

See the installation documentation for your platform for information about
installing the image data in pubs2.

dbccdb database

dbcc checkstorage records configuration information for the **target database**,
operation activity, and the results of the operation in the dbccdb database.
Stored in the database are dbcc stored procedures for creating and maintaining
dbccdb and for generating reports on the results of dbcc checkstorage
operations. See Chapter 10, "Checking Database Consistency," in the *System
Administration Guide: Volume 2*.

sybdiag database

Sybase Technical Support may create the sybdiag database on your system for
debugging purposes. This database holds diagnostic configuration data, and
should not be used by customers.

Determining the version of the installation scripts

sp_version lets you determine the current version of the scripts (*installmaster*,
installdbccdb, and so on) installed on Adaptive Server, whether they ran
successfully or not, and the length of time they took to complete.

The syntax for sp_version is:

    sp_version [*script_file* [, "all"]]

where:

- *script_file* is the name of the installation script (the default value is NULL).

- all reports details about each script, such as the date executed, and the
length of time for execution.

If you run sp_version without any parameters, it reports on all scripts.

This example describes what installation scripts were run, what time they were run, and what time they finished:

```
sp_version null, 'all'
Script          Version
Status
-----------     ------------------------------------------------------------
-----------
installmaster   15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu Sep
23 22:12:12 2004
Complete [Started=Sep 24 2004  3:39PM]-[Completed=Sep 24 2004  3:45PM
```

CHAPTER 3   **System Administration for Beginners**

This chapter:

•   Introduces new system administrators to important topics

•   Helps system administrators find information in the Sybase documentation

Experienced administrators may also find this chapter useful for organizing ongoing maintenance activities.

Topic	Page
Logical page sizes	33
Using "test" servers	34
Considerations when installing Sybase products	35
Allocating physical resources	37
Backup and recovery	39
Ongoing maintenance and troubleshooting	42
Keeping records	43
Additional resources	46

Logical page sizes

Database objects are built with logical pages. A databases and any of its related objects must use the same logical page size. That is, you cannot create a server that uses more than one logical page size. Adaptive Server allows you to create master devices and databases with logical page sizes of 2K, 4K, 8K, or 16K, but a given server installation can use only one of these four logical page sizes.

All databases in a server—and all objects in every database—use the same logical page size. For example, all the pages on a server with a logical page size of 4K must be 4K, even though you may not use some pages beyond the initial 2K.

Select the page size when you create the master device with dataserver -z.

For more information about the dataserver command, which is the command used to create the master device, see the *Utility Guide*. For more information about logical page sizes, see Chapter 3, "Configuring Memory," in *System Administration Guide: Volume 2.*

Using "test" servers

Sybase suggests that you install and use a test or development Adaptive Server, then remove it before you create the production server. Using a test server makes it easier to plan and test different configurations and less stressful to recover from mistakes. It is much easier to learn how to install and administer new features when there is no risk of having to restart a production server or re-create a production database.

If you use a test server, Sybase suggests that you do so from the point of installing or upgrading Adaptive Server through the process of configuring the server. It is in these steps that you make some of the most important decisions about your final production system. The following sections describe how using a test server can help system administrators.

Planning resources

Using a test server helps you plan the final resource requirements for your system and helps you discover resource deficiencies that you might not have anticipated.

In particular, disk resources can have a dramatic effect on the final design of the production system. For example, you may decide that, in the event of a media failure, a particular database requires nonstop recovery. This means you must configure one or more additional database devices to mirror the critical database. Discovering these resource requirements in a test server allows you to change the physical layout of databases and tables without affecting database users.

Use a test server to benchmark both Adaptive Server and your applications using different hardware configurations. This allows you to determine the optimal setup for physical resources at both the Adaptive Server level and the operating system level before bringing the entire system online for general use.

Achieving performance goals

Most performance objectives can be met only by carefully planning a database's design and configuration. For example, you may discover that the insert and I/O performance of a particular table causes a bottleneck. In this case, the best course of action may be to re-create the table on a dedicated segment and partition the table. Changes of this nature are disruptive to a production system; even changing a configuration parameter may require you to restart Adaptive Server.

Considerations when installing Sybase products

The responsibility for installing Adaptive Server and other Sybase products is sometimes placed with the system administrator. If installation is one of your responsibilities, use the following pointers to help you in the process.

Check product compatibility

Before installing new products or upgrading existing products, always read the release bulletin included with the products to understand any compatibility issues that might affect your system. Compatibility problems can occur between hardware and software and between different release levels of the same software. Reading the release bulletin in advance can save the time and guesswork of troubleshooting known compatibility problems. Pay particular attention to the lists of known problems that are included in the release bulletin.

Install or upgrade Adaptive Server

Read through the installation guide for your platform before you begin a new installation or upgrade. You may also want to consult with the operating system administrator to discuss operating system requirements for Adaptive Server. These requirements can include the configuration of memory, raw devices, asynchronous I/O, and other features, depending on the platform you use. Many of these tasks must be completed before you begin the installation.

If you are upgrading a server, back up all data (including the master database, user databases, triggers, and system procedures) offline before you begin. After upgrading, immediately create a separate, full backup of your data, especially if there are incompatibilities between older dump files and the newer versions.

Install additional third-party software

Adaptive Server generally includes support for the network protocols that are common to your hardware platform. If your network supports additional protocols, install the required protocol support.

As an alternative to the Sybase *interfaces* file, you can use a directory service to obtain a server's address and other network information. Directory services are provided by platform or third-party vendors and must be purchased and installed separately from the installation of Adaptive Server. See also "Directory services" on page 15 and the *Configuration Guide* for your platform for a list of the directory services that Adaptive Server currently supports.

Configure and test client connections

A successful client connection depends on the coordination of Adaptive Server, the client software, and network products. If you are using one of the network protocols installed with Adaptive Server, see the *Configuration Guide* for your platform for information about testing network connections. If you are using a different network protocol, follow the instructions that are included with the network product. You can also use "ping" utilities that are included with Sybase connectivity products to test client connections with Adaptive Server. For a general description of how clients connect to Adaptive Server, see "Connecting to Adaptive Server" on page 14. For details about the name and contents of the *interfaces* file, see the *Configuration Guide* for your platform

Allocating physical resources

Allocating physical resources is providing Adaptive Server the memory, disk space, worker processes, and CPU power required to achieve your performance and recovery goals. When installing a new server, every system administrator must make decisions about resource utilization. If you upgrade your platform, or if the design of your database system changes, you must also reallocate Adaptive Server resources by adding new memory, disk controllers, or CPUs. Early benchmarking of Adaptive Server and your applications can help you identify hardware resource deficiencies that create performance bottlenecks.

See Chapter 16, "Overview of Disk Resources" in Volume 2 of the *System Administration Guide: Volume 2* to understand the kinds of disk resources required by Adaptive Server. See also see Chapter 3, "Configuring Memory," in *System Administration Guide: Volume 2* and Chapter 5, Managing Mulitprocessor Servers," in *System Administration Guide: Volume 2* for information about memory and CPU resources.

The following sections provide helpful pointers in determining physical resource requirements.

Dedicated versus shared servers

The first step in planning Adaptive Server resources to understand the resources required by other applications running on the same machine. Generally, system administrators dedicate an entire machine for Adaptive Server use, which means that only the operating system and network software consume resources that might otherwise be reserved for Adaptive Server. On a shared system, other applications, such as Adaptive Server client programs or print servers, run on the same machine as Adaptive Server. It can be difficult to calculate the resources available to Adaptive Server on a shared system, because the types of applications and their pattern of use may change over time.

It is the system administrator's responsibility to take into account the resources used by operating systems, client programs, windowing systems, and so forth when configuring resources for Adaptive Server. Configure Adaptive Server to use only the resources that are available to it. Otherwise, the server may perform poorly or fail to start.

Decision-support and OLTP applications

Adaptive Server contains many features that optimize performance for OLTP, decision-support, and mixed-workload environments. However, to make optimal use of these features, determine in advance the requirements of your system's applications.

For mixed-workload systems, list the individual tables that you anticipate will be most heavily used for each type of application; this can help you achieve maximum performance for applications.

Advance resource planning

It is extremely important that you understand and plan resource usage in advance. In the case of disk resources, for example, after you initialize and allocate a device to Adaptive Server, that device cannot be used for any other purpose (even if Adaptive Server never fills the device with data). Likewise, Adaptive Server automatically reserves the memory for which it is configured, and this memory cannot be used by any other application.

When planning resource usage:

*   For recovery purposes, always place a database's transaction log on a separate physical device from its data. See Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2*.

*   Consider mirroring devices that store mission-critical data. See Chapter 2, "Mirroring Database Devices," in *System Administration Guide: Volume 2*. If your operating system supports these features, consider using disk arrays and disk mirroring for Adaptive Server data.

*   If you are working with a test Adaptive Server, for convenience, you may find it easier to initialize database devices as operating system files, rather than raw devices. Adaptive Server supports either raw partitions or certified file systems for its devices.

*   Changing configuration options can affect the way Adaptive Server consumes physical resources, especially memory. See Chapter 5, "Setting Configuration Parameters," for details about the amount of memory used by individual parameters.

Operating system configuration

Once you have determined the resources that are available to Adaptive Server and the resources you require, configure these physical resources at the operating system level:

- If you are using raw partitions, initialize the raw devices to the sizes required by Adaptive Server. If you initialize a raw device for Adaptive Server, you cannot use that device for any other purpose (for example, to store operating system files). Ask your operating system administrator for assistance in initializing and configuring raw devices to the required sizes.

- Configure the number of network connections. Make sure that the machine on which Adaptive Server runs can actually support the number of connections you configure. See your operating system documentation.

- Additional configuration may be required for your operating system and the applications that you use. Read the installation guide for your platform. Also read your client software documentation or consult with your engineers to understand the operating system requirements for your applications.

Backup and recovery

Making regular backups of your databases is crucial to the integrity of your database system. Although Adaptive Server automatically recovers from system crashes (for example, power outages) or server failures, only *you* can recover from data loss caused by media failure.

The following chapters, from the *System Adminstration Guide: Volume 2*, describe how to develop and implement a backup and recovery plan:

- Chapter 11, "Developing a Backup and Recovery Plan"

- Chapter 12, "Backing Upa and Restoring User Databases"

- Chapter 13, "Restoring the System Databases"

- Chapter 16, "Managing Free Space with Thresholds"

Keep up-to-date backups of master

Backing up the master database is the most crucial element of any backup and recovery plan. The master database contains details about the structure of your entire database system. Its stores information about the Adaptive Server databases, devices, and device fragments that make up those databases. Because Adaptive Server needs this information for recovery, it is crucial that you maintain an up-to-date backup copy of the master database at all times.

To ensure that your backup of master is always up to date, back up the database after each command or procedure that affects disks, storage, databases, or segments, including:

- Creating or deleting databases

- Initializing new database devices

- Adding new dump devices

- Using any device mirroring command

- Creating or dropping system stored procedures, if they are stored in master

- Creating, dropping, or modifying a segment

- Adding new Adaptive Server logins

To back up master to a tape device, start isql and enter the command:

```
dump database master to "tape_device"
```

where *tape_device* is the name of the tape device (for example, */dev/rmt0*).

Keep offline copies of system tables

In addition to backing up master regularly, keep offline copies of these system tables: sysdatabases, sysdevices, sysusages, sysloginroles, and syslogins. Use the bcp utility described in the *Utility Guide* and store a printed copy of the contents of each system table. Create a printed copy by printing the output of:

```
select * from sysusages order by vstart
select * from sysdatabases
select * from sysdevices
select * from sysloginroles
select * from syslogins
```

If you have copies of these tables, and a hard-disk failure or other disaster makes your database unusable, you can use the recovery procedures described in Chapter 13, "Restoring the System Databases," in *System Administration Guide: Volume 2.*

Also keep copies of all data definition language (DDL) scripts for user objects, as described under "Keeping records" on page 43.

Automate backup procedures

Creating an automated backup procedure makes the process easier and quicker to perform. Automating backups can be as simple as using an operating system script or a utility (for example, the UNIX cron utility) to perform the necessary backup commands. Or you can automate the procedure further by using thresholds, which are discussed in Chapter 16, "Managing Free Space with Thresholds," in *System Administration Guide: Volume 2*.

❖ **Creating an automated backup procedure**

Although the commands required to create an automated script vary, depending on the operating system you use, all scripts should accomplish the same basic steps:

1   Start isql and dump the transaction log to a holding area (for example, a temporary file).

2   Rename the dump file to a name that contains the dump date, time, and database name.

3   In a history file, record information about the new backup.

4   In a separate file, record any errors that occurred during the dump.

5   Automatically send mail to the system administrator for any error conditions.

Verify data consistency before backing up a database

Your database backups must be consistent and accurate, especially for master. If you back up a database that contains internal errors, the errors persist in a restored version of the database.

Use the dbcc commands to check a database for errors before backing it up. Always use dbcc commands to verify the integrity of a database before dumping it. If dbcc detects errors, correct them before dumping the database.

Over time, if you discover few or no errors while running dbcc, you may decide that the risk of database corruption is small and that you need to run dbcc only occasionally. If the consequences of losing data are too high, continue to run dbcc commands each time you back up a database.

Note For performance considerations, many sites choose to run dbcc checks outside of peak hours or on separate servers.

See Chapter 10, "Checking Database Consistency," in the *System Administration Guide: Volume 2*.

Monitor the log size

When the transaction log becomes nearly full, it may be impossible to use standard procedures to dump transactions and reclaim space. The system administrator should monitor the log size and perform regular transaction log dumps (in addition to regular database dumps) to avoid this situation. Set up a threshold stored procedure that notifies you (or dumps the log) when the log reaches a certain capacity. See Chapter 16, "Managing Free Space with Thresholds," in *System Administration Guide: Volume 2*. Sybase also suggests that, to shorten the time required to dump and load the database, dump the transaction log immediately prior to performing a full database dump.

You can also monitor the space used in the log segment manually using sp_helpsegment, as described under Chapter 8, "Creating and Using Segments in *System Administration Guide: Volume 2*.

Ongoing maintenance and troubleshooting

In addition to making regularly scheduled backups, the system administrator performs the maintenance activities throughout the life of a server discussed in this section.

Starting and stopping Adaptive Server

Most system administrators automate the procedure for starting Adaptive Server to coincide with the start-up of the server machine. Do this by editing operating system start-up scripts, or by using other operating system procedures. See the configuration documentation for your platform to determine how to start and stop Adaptive Server.

Viewing and pruning the error log

Examine the contents of the error log on a regular basis to determine whether serious errors have occurred. You can also use operating system scripts to scan the error log for particular messages and to automatically notify the system administrator when specific errors occur. Checking the error log regularly may help determine whether there are continuing problems of the same nature, or whether a particular database device is likely to fail. See Chapter 12, "Diagnosing System Problems," for more information about error messages and their severity.

The error log file can grow large over time, since Adaptive Server appends informational and status messages to it each time it starts. You can periodically "prune" the log file by opening the file and deleting old records. Keeping the log file to a manageable size saves disk space and makes it easier to locate current errors.

Keeping records

Keeping records about your Adaptive Server system is an important part of your job as a system administrator. Accurate records of changes and problems that you have encountered can be a valuable reference when you are contacting Sybase Technical Support or recovering databases. They can also provide vital information for administrators who manage the Adaptive Server system in your absence.

Contact information

Maintain a list of contact information for yourself as well as the System Security Officer, Operator, and database owners on your system. Also, record secondary contacts for each role. Make this information available to all Adaptive Server users so that the appropriate contacts receive enhancement requests and problem reports.

Configuration information

Ideally, create databases and database objects, and configure Adaptive Server using script files that you store in a safe place. Storing the script files makes it possible to re-create your entire system in the event of a disaster. You can also use script files to quickly re-create database systems for evaluation purposes on new hardware platforms. If you use a third-party tool to perform system administration, remember to generate equivalent scripts after performing administration tasks.

Consider recording the following kinds of information:

*   Commands used to create databases and database objects (DDL scripts)

*   Commands that add new Adaptive Server logins and database users

*   The current Adaptive Server configuration file, as described in "Using sp_configure with a configuration file" on page 68

*   The names, locations, and sizes of all files and raw devices initialized as database devices

Maintain a dated log of all changes to the Adaptive Server configuration. Mark each change with a brief description of when and why you made the change, as well a summary of the end result.

Maintenance schedules

Keep a calendar of regularly scheduled maintenance activities; list any of the procedures you perform at your site:

*   Using dbcc to check database consistency

*   Backing up user and system databases

*   Monitoring the space left in transaction logs (if this is not done automatically)

- Dumping the transaction log

- Examining the error log contents for Adaptive Server, Backup Server, and Adaptive Server Monitor

- Running the update statistics command (see Chapter 1, "Using the set statistics Commands," in *Performance and Tuning Series: Improving Performance with Statistical Analysis*)

- Examining auditing information, if the auditing option is installed

- Recompiling stored procedures

- Monitoring resource utilization of the server machine

System information

Record information about the hardware and operating system on which you run Adaptive Server, including:

- Copies of operating system configuration files or start-up files

- Copies of network configuration files (for example, the *hosts* and *services* files)

- Names and permissions for the Adaptive Server executable files and database devices

- Names and locations of the tape devices used for backups

- Copies of operating system scripts or programs for automated backups, starting Adaptive Server, or performing other administration activities

Disaster recovery plan

Consolidate the basic backup and recovery procedures, the guidelines in "Backup and recovery" on page 39, and your personal experiences in recovering data into a concise list of recovery steps tailored to your system. This can be useful to both yourself and to other system administrators who may need to recover a production system in the event of an emergency.

Additional resources

The amount of information for system administrators to learn may seem overwhelming. There are several software tools that can help you learn and facilitate basic administration tasks. These include Adaptive Server Monitor, used for monitoring server performance and other activities, and Sybase Central, which simplifies many administration tasks. There are also many third-party software packages available designed to help system administrators manage daily maintenance activities.

CHAPTER 4 **Managing and Monitoring Adaptive Server**

Topic	Page
Sybase Control Center for Adaptive Server	47
Adaptive Server Sybase Central Plug-in	48

This chapter describes the GUI managing and monitoring tools for Adaptive Server: This is meant merely as an overview to introduce you to these tools. See the online help for a complete description.

Sybase Control Center for Adaptive Server

The Sybase Control Center for Adaptive Server® is the preferred tool for managing and monitoring the status and availability of Adaptive Server.

Sybase Control Center is a Web-based tool for monitoring and administration. See the Installation Guide at http://sybooks.sybase.com/nav/summary.do?prod=10680 for information about downloading and installing Sybase Control Center.

You can use a single instance of Sybase Control Center to monitor and manage multiple Adaptive Servers.

Sybase Control Center supports Adaptive Server Enterprise version 15.0.2 and later. See the Sybase Control Center documentation for a list of supported features.

The Sybase Control Center client/server architecture allows multiple clients to monitor all Adaptive Servers in an enterprise through one or more Sybase Control Center servers. Sybase Control Center for Adaptive Server provides availability monitoring, historical performance monitoring, and administration capabilities in a scalable Web application that is integrated with management modules for other Sybase products. It offers shared, consolidated management of heterogeneous resources from any location, e-mail alerts that provide state- and threshold-based notifications about availability and performance in real time, and intelligent tools for spotting performance and usage trends, all via a thin-client, rich Internet application (RIA) delivered through your Web browser.

Use Sybase Control Center for Adaptive Server to gather statistics that over time will give you powerful insight into patterns of use and the behavior of databases, devices, caches, and processes on your servers. You can display collected data as tables or graphs. By plotting results over any period of time you choose, from a minute to a year, you can both see the big picture and focus on the particulars. Detailed knowledge of how your servers have performed in the past helps you ensure that Adaptive Server meets your needs in the future.

See the online help here for more information about using Sybase Control Center at http://sybooks.sybase.com/nav/summary.do?prod=10680.

Adaptive Server Sybase Central Plug-in

Sybase Central is a graphical user interface (GUI) management tool. Sybase Central accepts a variety of "plug-ins" that allow you to manage specific Sybase products. The Adaptive Server plug-in allows you to manage Adaptive Server and helps you perform complex administration tasks without the need to remember the syntax of Transact-SQL commands or system stored procedures. You can use the Adaptive Server plug-in to:

• Manage multiple servers from one console – You can manage all the Adaptive Server installations from the Sybase Central main window.

• Generate database definition language (DDL) – You can generate DDL for the objects in Adaptive Server.

- Visually represent objects – You can see the databases and logins in each Adaptive Server and the objects in each database, and windows expand and contract to display information about databases and logins. The Adaptive Server plug-in expands to display information about many items, including:

    - Databases and tables

    - Disk devices

    - Active processes and locks

    - Logins and users

    - Data caches

    - ASE Replicator, Job Scheduler, and Messaging Services

    - Access to other utilities such as Interactive SQL (for sending queries and displaying query results).

- Navigate between related objects – To get more information about a database object related to the one whose property sheet you are displaying, navigate directly through the displayed object's dialog box to the related object.

- Create a cluster – The Adaptive Server plug-in allows you to create a cluster if you have purchased Adaptive Server Cluster Edition. See the online help and the *User Guide to Clusters*.

Using the Adaptive Server Plug-in

The Adaptive Server plug-in for Sybase Central provides you with an intuitive and easy way to administer Adaptive Server Enterprise. Sybase Central displays the Adaptive Server plug-in in its left-hand pane. Included in this pane is a hierarchical list of folders that represent different objects the plug-in can manage, including:

- Viewing and changing the characteristics of the object

- Creating another object:

- Generating the SQL text for creating an object (which allows you to reverse engineer Adaptive Server objects)

- Deleting an object

- Configuring Adaptive Server

- Managing:

    - Database devices

    - Proxy and temporary databases

    - Indexes

    - Partitions

    - Segments

    - Triggers

    - Logins and roles

    - Views

    - ASE Replicator

- Configuring Adaptive Server jobs with Job Scheduler

- Starting and stopping Adaptive Server

- Executing queries

- Logging SQL statements generated by the plug-in, based on a user's actions.

Starting and stopping Sybase Central

To start Sybase Central:

- (UNIX) Move to the *$SYBASE/shared/sybcentral600* directory and run the *scjview.sh* script.

- (Windows) Choose Programs | Sybase | Sybase Central v6 from the Start menu, or move to the *%SYBASE%\Shared\Sybase Central 6.0.0* directory and run the *scjview.bat* script.

To stop Sybase Central, select File | Exit.

Registering Adaptive Server Plug-in

The Adaptive Server plug-in is registered in Sybase Central as part of the server installation. However, if Adaptive Server plug-in is not correctly registered, you can manually register the Adaptive Server plug-in:

- On Unix, run *$SYBASE/ASEP/bin/registerASEP*.

- On Windows, run *%SYBASE%\ASEP\bin\registerASEP.bat*

- You can register the Adaptive Server plug-in manually by:

    a   Select Register from Tools | Plug-ins. A registration wizard appears.

    b   Select Register

    c   Select "Register a plug-in by specifying a plug-in registration file."

    d   Click Browse.

    e   Navigate to *$SYBASE/ASEP/bin* (*%SYBASE%\ASEP\bin* on Windows) and select *ASEPlugin.jpr*. Follow the wizard to register the Adaptive Server plug-in.

Performing common tasks

The following are some common tasks users perform with the Adaptive Server plug-in.

For more information about all the following tasks, see the Adaptive Server plug-in online help.

Starting and stopping Adaptive Server

If the Unified Agent is monitoring Adaptive Server, you can start, stop, and restart the server by right-clicking on the server and selecting Shutdown, Start, or Restart.

If the Unified Agent is not monitoring Adaptive Server, you can shutdown the server by selecting Shutdown.

Connecting to Adaptive Server

You can connect to an Adaptive Server by any of these methods:

- Select the Connect icon from the tool bar.

- Right click on Adaptive Server Enterprise and select Connect from the menu.

- Right click on any server group and select Connect from the menu.

    The connected server is displayed in the Default server group if the connection is initiated from the Adaptive Server Enterprise folder or the connect icon. The plug-in displays "Connected to server" in the corresponding server group if the connection is initiated from the server group.

You can also specify a server to which you want to connect by any of the following:

- Specifying the server's host name and port number in the Connect dialog box.

- Selecting a pre-defined Adaptive Server from the server name dropdown list. This drop down list is derived from the servers listed in the interfaces file (UNIX) and *sql.ini* files (Windows) and LDAP servers.

- Discover which Adaptive Servers are available by clicking on Find in the Connect dialog. Before you can use this method, you must first define the discovery servers in Server Discovery tab located in the Adaptive Server Enterprise property page.

Creating a database

Before creating a database, make sure enough space is available on the database devices you plan to use.

To create a database:

- Right-click on the Add Database icon in the right-hand panel, or,

1 Select the Databases folder.

2 Choose File | New | Database or click on the Add Database option in the Databases folder. The Create a New Database wizard opens. The Create a New Database wizard asks for the following information:

Table 4-1: Inputs to create a new database wizard

Input	Description
Database name	Enter a name for the database
Database device	Specify the database device or devices on which to allocate the new database
Database device size	Specify a size for each database device
Data or log	Specify whether the database device will store data or the transaction log.
With override	Specify with override if you want to store data and log on the same device.
For load	If you are creating the database so you can restore it from a backup, check the For Load check box. This is the case only if you are recovering from media failure or if you are moving a database from one location to another.
Guest account	Specify whether to create a guest user in the database.

If you do not enter a size, Adaptive Server allocates either the value of the database size configuration variable or the size of the *model* database, whichever is larger.

If you have limited storage *and* must put the transaction log and the data on the same logical device, specifying With Override allows Adaptive Server to maintain the log on separate device fragments from the data.

You cannot remove or change a database device after creating the database unless you first delete the database.

Warning! Deleting a database also deletes all its objects.

Deleting a database

Only the owner of a database can delete it.

To delete a database:

1   Select the database icon.

2   Choose Edit | Delete.

3   Confirm the deletion in the confirmation dialog box.

Note Sybase recommends that you back up the master database after you delete a user database.

Adding a user

Database owners can add and delete users in the databases they own.

To create a user:

1   Expand the databases folder (select the "+" icon) and select the Users folder.

2   Choose File | New | User.

The Add a New User wizard opens and asks for this information:

Table 4-2: Inputs to Add a New User wizard

Input	Description
Name	A name for the user. The name does not have to be the same as the login.
Login name	Login to which this user is assigned.
Group	Optionally, assign a group to the user. Default: public

---

Note A user can be a member of one assigned group or the default "public" group.

You can also select the Users folder. In the right pane, double-click the Add User icon.

Deleting a user

You cannot delete a user who owns objects. Since there is no command to transfer ownership of objects, you must delete objects owned by a user before you can delete the user. Also, you cannot delete a user who has granted permissions to other users without first revoking the permissions with cascade. If appropriate, re-grant the permissions to the other users.

Locking a login is a simple alternative to deleting a user.

To delete a user:

1   Select the user icon.

2   Choose Edit | Delete.

3   Confirm the deletion in the confirmation dialog box.

You can also select the user folder by right-clicking on the user icon and select Delete.

Before you delete a user:

1   Revoke the user's command and object permissions with cascade.

2   Re-grant the permissions to the other users, if appropriate.

3   Delete the user's objects.

Creating a table

Only a database owner or a user with create table permission can create a table.

To create a table:

1   In a database you are working in, select the User Tables folder.

2   Choose File | New | Table or click on the Add Table icon in the User Tables folder.

     The Table Editor opens.

3   In the Name box, enter a name.

4   From the Owner list, choose an owner. The default is "dbo".

You can also select the User Tables folder. In the right pane, double-click the Add Table icon.

Deleting a table

Before you delete a table, be sure that no other objects reference it. If any objects reference it, edit those objects to avoid errors. To find out if other objects reference a table, check its dependencies.

Note When you delete a table, Adaptive Server deletes the indexes and triggers associated with the table and unbinds the rules or defaults that are bound to its columns.

Only table owners can delete tables.

To delete a table:

*   Follow these steps:

    *   Select the table icon.

    *   Choose Edit | Delete.

    *   Confirm the deletion in the confirmation dialog box, or,

*   You can also select the table by right-clicking on the table icon and selecting Delete.

Creating a server group

To create a server group:

1   Select Adaptive Server Enterprise

2   Choose File | New | Server Group

3   Follow the steps provided by the Create New Server Group wizard.

You can also add a server group by double-clicking on the Add Server Group from the right-hand pane.

Getting server status

If the Unified Agent is monitoring Adaptive Server, check the server status by any of the following:

*   Click on the server group to which the server belongs. Check the Status column in the Details pane of the server group.

*   Click on the Adaptive Server Enterprise listed under Sybase Central, and then click on Servers tab on the right hand side panel. The server status is printed in the Status column.

- A green triangle on the lower right-hand side of the server icon indicates that Adaptive Server is running. A red square indicates that Adaptive Server is stopped.

Note By default, the Adaptive Server plug-in does not have Check Server Status enabled. To enable Unified Agent to monitor Adaptive Server:

- Right click on Adaptive Server Enterprise and select Properties.

- Select Preferences and check "Enable Unified Agent (UA) related features."

Getting the server log

If the Unified Agent is monitoring Adaptive Server, retrieve the server log by selecting the server and clicking on the Server Log tab in the right-hand pane.

The server log is retrieved based on how you have configured the filter for the the server log. To configure the server log filtering, right-click on the server and select Server Log Filter. By default, the Adaptive Server plug-in retrieves the last 1000 lines from the server log. You can configure the server filter to retrieve:

- The entire log file.

- The last *n* number of lines.

- The log from the last *n* number of days.

- The lines that match the regular expression

Logging SQL statements

To log all SQL statements executed through the Adaptive Server plug-in:

- Right click on a server and select "Log SQL Statement."

- Select whether you want SQL statements logged directly to a window or to a file.

Executing SQL statements

Execute SQL statements from within the Adaptive Server plug-in by using the Interactive SQL query tool. To start the Interactive SQL tool, you can either:

- Right-click the server on which you want to execute the SQL statements and select Open Interactive SQL from the menu, or

1 Click on Adaptive Server Enterprise.

2 Click the Utilities tab on the right-hand pane and select Interactive SQL

Execute SQL statements simultaneously on a set of servers belonging to a server group:

1 Right -click the server group and choose Execute SQL.

2   Select the servers on which you want to execute the SQL statements

3   Click Execute.

The result set for each server is listed in the Result Set pane of the SQL Execution dialog.

Viewing SQL execution plan and cost information

Use the Adaptive Server plug-in to view a GUI version of the SQL execution plan for individual queries (much like a GUI version of showplan) and execution plans for all queries in a stored procedure. This GUI display includes nodes for each of the operators of the execution plan.

To get the GUI plan:

1   Start Interactive SQL.

2   Execute the query or stored procedure

3   Click on the plan tab in the Results pane of Interactive SQL

4   Select a query from the queries drop down list.

5   Click the Details tab to see the GUI plan of the selected query. Click on an operator node to see the detailed statistics for that node.

6   Click on the XML tab to see an XML representation of the execution plan for the selected query

7   Click on the Text tab to see the execution plan in a text format for the submitted queries

For more information about Interactive SQL, see "Starting Interactive SQL" on page 58.

Viewing and updating object properties

View and modify the configuration of any object represented in the Adaptive Server plug-in using the Property dialog.

To bring up the Property dialog:

1   Click on the object you want to view or modify.

2   Right-click on the object and select Properties.

3   Select the appropriate tab to perform your task.

4   Make any modification in the Property dialog.

5   Click on Apply, OK, or Cancel.

Generate the SQL text for creating an object

Generate the SQL text required for creating an object, which allows you to reverse engineer the object. To generate SQL text, right-click on the object and select "Generate DDL."

Viewing and updating
Adaptive Server
configuration
parameters

View and update the Adaptive Server configuration parameters using the
Server Properties dialog.

1   Right click on the server and select Configuration in the menu

2   Select the functional group from the drop down list in the Show
     Configuration Parameters

3   Find and select the parameter you want to view or update

4   Enter new valuing the value column if update is necessary

5   Click on Apply/OK/Cancel accordingly

Using Interactive SQL

Interactive SQL allows you to execute SQL statements, build scripts, and
display database data to the server. You can use it to:

•   Browse the information in a database.

•   Test SQL statements that you plan to include in an application.

•   Save query results to a file.

•   Edit data in result sets.

•   Load data into a database and carry 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.

Starting Interactive SQL

To start Interactive
SQL from Sybase
Central

To start Interactive SQL, either:

•   Select a database in Sybase Central and select File | Open Interactive SQL.
     Interactive SQL connects to the database. You can also right-click on the
     database and select Open Interactive SQL.

     The menu item Open Interactive SQL opens a connection to a server.
     However, when you select the menu item for a server, Interactive SQL
     opens a connection to the default database for that server. When you select
     a specific database from the Open Interactive SQL menu, Interactive SQL
     opens to the selected database.

•   Select Tools | Adaptive Server Enterprise | Open Interactive SQL to start
     Interactive SQL without a connection to a server. The Connect dialog
     appears.

To start Interactive
SQL from the
command line

How you start Interactive SQL from the command line depends on your
operating system.

If you start Interactive SQL independently, the Connect dialog appears, which
lets you connect to a database just as you would in Sybase Central.

•   For UNIX, change to the *$SYBROOT/DBISQL/bin* directory and enter:

     ```
     dbisql
     ```

     On Windows, change to the *%SYBROOT%\DBISQL\bin* directory and
     enter:

     ```
     dbisql.bat
     ```

•   In the Connection dialog, enter the information to connect to a database in
     the Connect dialog box and click OK.

To open a new Interactive SQL window:

1   Choose Window | New Window. The Connect dialog appears.

2   In the Connect dialog, enter connection options, and click OK to connect.

     The connection information (including the database name, your user ID,
     and the database server) appears on the title bar above the SQL Statements
     pane.

You can also connect to or disconnect from a database with the Connect and
Disconnect commands in the SQL menu, or by executing a connect or
disconnect statement in the SQL Statements pane.

CHAPTER 5   **Setting Configuration Parameters**

This chapter describes the Adaptive Server configuration parameters, which are listed here alphabetically.

A configuration parameter is a user-definable setting that you set with sp_configure. Configuration parameters are used for a wide range of services, from basic to specific server operations, and for performance tuning.

Overview

Configuration parameters are user-definable settings that control various aspects of Adaptive Server behavior. Adaptive Server supplies default values for all configuration parameters. Use configuration parameters to tailor Adaptive Server for an installation's particular needs.

Read this chapter carefully to determine which configuration parameters you should reset to optimize server performance.

Warning! Change configuration parameters with caution. Arbitrary changes in parameter values can adversely affect Adaptive Server performance and other aspects of server operation.

The Adaptive Server configuration file

Adaptive Server stores the values of configuration parameters in a configuration file, which is an ASCII text file. When you install a new Adaptive Server, your parameters are set to the default configuration; the default name of the file is *server_name.cfg*, and the default location of the file is the Sybase Adaptive Server home directory ($SYBASE_ASE). Each time you modify a configuration parameter, Adaptive Server creates a copy of the outdated configuration file, using the naming convention *server_name.001*, *server_name.002*, *server_name.003...server_name.999*. Adaptive Server writes the new values to the file *server_name.cfg* or to a file name you specify at start-up.

Modifying configuration parameters

Set or change configuration parameters in one of the following ways:

- By executing sp_configure with the appropriate parameters and values,

- By editing your configuration file and then invoking sp_configure with the configuration file option, or

- By specifying the name of a configuration file at start-up.

Configuration parameters are either dynamic or static. Dynamic parameters take effect as soon as you execute sp_configure. Static parameters require memory to be reallocated, so they take effect only after you have restarted Adaptive Server. The description of each parameter in this chapter indicates whether it is static or dynamic.

Adaptive Server writes the new value to the system table sysconfigures and to the configuration file when you change the value. The current configuration file and sysconfigures reflect configured values, not run values. The system table syscurconfigs reflects current run values of configuration parameters.

Required roles for modifying configuration parameters

The roles required for using sp_configure:

- Any user can execute sp_configure to display information about parameters and their current values.

- Only a system administrator or a system security officer can execute sp_configure to modify configuration parameters.

- Only a system security officer can execute sp_configure to modify values for:

    - allow procedure grouping

    - allow remote access

    - allow sendmsg

    - allow updates to system tables

    - auditing

    - audit queue size

    - check password for digit

    - current audit table

    - enable ldap user auth

    - enable pam user auth

    - enable ssl

    - log audit logon failure

    - log audit logon success

    - maximum failed logins

    - minimum password length

    - msg confidentiality reqd

    - msg integrity reqd

    - secure default login

    - select on syscomments.text

    - SQL Perfmon Integration

    - syb_sendmsg port number

    - suspended audit when device full

    - systemwide password expiration

    - unified login required

    - use security services

Unit specification using *sp_configure*

sp_configure allows you to specify the value for configuration parameters in unit specifiers. The unit specifiers are p or P for pages, m or M for megabytes, and g or G for gigabytes. If you do not specify a unit, and you are configuring a parameter that controls memory, Adaptive Server uses the logical page size for the basic unit.

Note   When you are configuring memory-related parameters, use only the P (page size) parameter for your unit specification. If you use any other parameter to configure memory related parameters, Adaptive Server may issue an arithmetic overflow error message.

The syntax to indicate a particular unit specification is:

```
sp_configure "parameter name", 0, "p|P|k|K|m|M|g|G"
```

You must include the "0" as a placeholder.

You can use this unit specification to configure any parameter. For example, when setting number of locks to 1024 you can enter:

```
sp_configure "number of locks", 1024
```

or:

```
sp_configure "number of locks", 0, "1K"
```

This functionality does not change the way in which Adaptive Server reports sp_configure output.

Global versus session settings

Some Adaptive Server global settings are similar to session-level settings. For example, both sp_configure and sp_passwordpolicy include a minimum password length parameter.

If a global and session-level parameters configure the same setting, only the session level setting applies to the login (login level settings override global settings).

For example, if you set this global configuration configuration parameter:

```
sp_configure 'minimum password length' 12
```

And you attempt to add user joe to the server with a insufficiently long password:

```
sp_adduser joe, joejoe
```

The command fails:

```
Msg 10317, Level 14, State 1:
Procedure 'sp_password', Line 118:
The specified password is too short. Passwords must be
at least 12 character(s) long.
Msg 17720, Level 16, State 1:
Procedure 'sp_password', Line 128:
Error:  Unable to set the Password.
```

However, if you set sp_passwordpolicy to a shorter length, Adaptive Server requires a shorter password length for this session:

```
sp_passwordpolicy 'set', 'minimum password length', 6
```

The command succeeds:

```
sp_addlogin joe, joejoe

Password correctly set.
Account unlocked.
New login created.
(return status = 0)
```

Adaptive Server tests all configuration limits independently, and the login attempt must pass all the applicable limits to succeed.

Getting help information on configuration parameters

Use either sp_helpconfig or sp_configure to display information on a particular configuration parameter. For example:

```
sp_helpconfig "number of open"

Configuration option is not unique.
option_name                    config_value  run_value
------------------------------ ------------ -----------
number of open databases                 12          12
number of open indexes                  500         500
number of open objects                  500         500


                sp_helpconfig "number of open indexes"

number of open indexes sets the maximum number of indexes that can be open at
one time on SQL Server. The default value is 500.
Minimum Value Maximum Value Default Value Current Value Memory Used
```

```
------------- ------------- ------------- ------------- -----------
          100    2147483647           500           500           208


sp_configure "number of open indexes"

Parameter Name          Default  Memory Used  Config Value  Run Value
----------------------  -------  -----------  ------------  ---------
number of open indexes      500          208           500        500
```

See Chapter 3, "Configuring Memory," in System Administration Guide: Volume 2.

Using *sp_configure*

sp_configure displays and resets configuration parameters. You can restrict the number of parameters that sp_configure shows by using sp_displaylevel to set your display level to one of:

- Basic

- Intermediate

- Comprehensive

For information about display levels, see "User-defined subsets of the parameter hierarchy: display levels" on page 74. For information about sp_displaylevel, see the *Reference Manual: Procedures*.

Table 5-1 describes the syntax for sp_configure. The information in the "Effect" column assumes that your display level is set to "comprehensive."

Table 5-1: sp_configure syntax

Command	Effect
sp_configure	Displays all configuration parameters by group, their current values, their default values, the value to which they have most recently been set, and the amount of memory used by this particular setting.
sp_configure "*parameter*"	Displays current value, default value, most recently changed value, and amount of memory used by the specified parameter.
sp_configure "*parameter*", *value*	Resets *parameter* to *value*.
sp_configure "*parameter*", 0, "default"	Resets the specified parameter to its default value.

Command	Effect
sp_configure "*group_name*"	Displays all configuration parameters in *group_name*, their current values, their default values, the values to which they were recently set, and the amount of memory used by each.
sp_configure "configuration file", 0, "*sub_command*", "*file_name*"	Sets configuration parameters from the configuration file. See "Using sp_configure with a configuration file" on page 68 for descriptions of the parameters.

Syntax elements

The commands in Table 5-1 use the following variables:

- *parameter* – is any valid Adaptive Server configuration parameter or parameter substring.

- *value* – is any integer within the valid range for that parameter. (See the descriptions of the individual parameters for valid range information.) Parameters that work as toggles have only two valid values: 1 (on) and 0 (off).

- *group_name* – is the name of any group in the parameter hierarchy.

Parameter parsing

sp_configure parses each parameter (and parameter name fragment) as "*%parameter%*". A string that does not uniquely identify a particular parameter returns values for all parameters matching the string.

The following example returns values for all configuration parameters that include "lock," such as lock shared memory, number of locks, lock promotion HWM, server clock tick length, print deadlock information, and deadlock retries:

```
sp_configure "lock"
```

Note If you attempt to set a parameter value with a nonunique parameter name fragment, sp_configure returns the current values for all parameters matching the fragment and asks you to specify a unique parameter name.

Using *sp_configure* with a configuration file

Configure Adaptive Server either interactively, by using sp_configure as described above, or noninteractively, by instructing Adaptive Server to read values from an edited or restored version of the configuration file.

By making your changes from the configuration file, you can:

*   Replicate a specific configuration across multiple servers by using the same configuration file.

*   Use a configuration file as a baseline for testing configuration values on your server.

*   Use a configuration file to perform validation checking on parameter values before actually setting the values.

*   Create multiple configuration files and switch between them as your resource needs change.

For information on editing the file, see "Editing the configuration file" on page 70. For information on specifying the name of the configuration file at start-up, see "Starting Adaptive Server with a configuration file" on page 71.

Naming tips for the configuration file

To work with a configuration file that has a name other than the default name, keep the *server_name* part of the file name, and include at least one alphabetic character in the extension (for example *my_server.A001*). Alternatively, you can change the *server_name* part of the file name (for example, *A_my_server.001*). Doing this avoids confusion with the backup configuration files generated by Adaptive Server when you modify a parameter.

Using *sp_configure* to read or write the configuration file

The syntax for using the configuration file option with sp_configure is:

    sp_configure "configuration file", 0, "*subcommand*", "*file_name*"

where:

*   "configuration file" – including quotes, specifies that this command uses the configuration file.

*   0 – required—for backward compatibility—after the configuration file parameter.

*   "*subcommand*" – is one of:

- write – creates a file named *file_name* with the current configuration. If *file_name* already exists, a message is written to the error log; the existing file is renamed using the convention *server_name.001*, *server_name.002*, and so on. If you have changed a static parameter, but you have not restarted your server, write displays the currently running value for that parameter. If you do not specify a directory with *file_name*, the file is written to the directory from which Adaptive Server was started.

- read – performs validation checking on values contained in *file_name* and reads those values that pass validation into the server. If any parameters are missing from *file_name*, the current values for those parameters are used.

    If the value of a static parameter in *file_name* is different from its current running value, read fails and a message is printed. However, validation is still performed on the values in *file_name*.

- verify – performs validation checking on the values in *file_name*. This is useful if you have edited the configuration file, as it prevents you from attempting to configure your server with invalid configuration values.

- restore – creates *file_name* with the most recently configured values. If you have configured static parameters to new values, this subcommand writes the configured, not the currently running, values to the file. This is useful if all copies of the configuration file have been lost and you must generate a new copy. If you do not specify a directory with *file_name*, the file is written to the directory from which Adaptive Server was started.

- *file_name* – specifies the configuration file to use in conjunction with any *subcommand*. If you do not specify a directory as part of the file name, the directory where Adaptive Server was started is used.

Examples   **Example 1**   Performs validation checking on the values in the file *srv.config* and reads the parameters that pass validation into the server. Current run values are substituted for values that do not pass validation checking:

```
sp_configure "configuration file", 0, "read", "srv.config"
```

Example 2   Creates the file *my_server.config* and writes the current configuration values the server is using to that file:

```
sp_configure "configuration file", 0, "write", "my_server.config"
```

Editing the configuration file

The configuration file is an ASCII file that you can edit with any text editor that can save files in ASCII format. The syntax for each parameter is:

parameter_name={*value* | DEFAULT}

where:

- *parameter_name* – is the name of the parameter you want to specify.

- *value* – is the numeric value for set *parameter_name*.

- "DEFAULT" – specifies that you want to use the default value for *parameter_name*.

Examples

Example 1 This example specifies that the transaction can retry its attempt to acquire a lock one time when deadlocking occurs during an index page split or shrink:

```
deadlock retries = 1
```

Example 2 This example specifies that the default value for the parameter cpu accounting flush interval should be used:

```
cpu accounting flush interval=DEFAULT
```

When you edit a configuration file, your edits are not validated until you check the file using the verify option, read the file with the read option, or restart Adaptive Server with that configuration file.

If all your configuration files are lost or corrupted, you can re-create one from a running server by using the restore subcommand and specifying a name for the new file. The parameters in the new file are set to the values with which your server is currently running.

Permissions for configuration files

Configuration files are nonencrypted ASCII text files. By default, they are created with read and write permissions set for the file owner, and read permission set for all other users. If you created the configuration file at the operating system level, you are the file owner; if you created the configuration file from Adaptive Server, using the write or restore parameter, the file owner is the user who started Adaptive Server. Usually, this is the user "sybase." To restrict access to configuration files, use your operating system's file permission command to set read, write, and execute permissions as appropriate.

Note You must set permissions accordingly on *each* configuration file created.

Backing up configuration files

Configuration files are not automatically backed up when you back up the master database. They are operating system files—back them up in the same way you back up your other operating system files.

Checking the name of the configuration file currently in use

Due to space limitations, sp_configure output truncates the name of the configuration file. To see the full name of the configuration file, use:

```
select s1.value2
from syscurconfigs s1, sysconfigures s2
where s1.config = s2.config
and s2.name = "configuration file"
```

Starting Adaptive Server with a configuration file

By default, Adaptive Server reads the configuration file *server_name.cfg* in the start-up directory when it starts. If this file does not exist, it creates a new file and uses Adaptive Server defaults for all values.

You can start Adaptive Server with a specified configuration file. For more information, see the *Utility Guide*.

If the configuration file you specify does not exist, Adaptive Server prints an error message and does not start.

If the command is successful, the file *server_name.bak* is created. This file contains the configuration values stored in sysconfigures prior to the time sysconfigures was updated with the values read in from the configuration file you specified. This file is overwritten with each subsequent start-up.

Configuration file errors

When there are errors in the configuration file, Adaptive Server may not start, or may use default values.

Adaptive Server uses default values if:

- There are illegal values. For example, if a parameter requires a numeric value, and the configuration file contains a character string, Adaptive Server uses the default value.

- Values are below the minimum allowable value.

The parameter hierarchy

Configuration parameters are grouped according to the area of Adaptive Server behavior they affect. This makes it easier to identify all parameters that you might need to tune to improve a particular area of Adaptive Server performance.

Although each parameter has a primary group to which it belongs, many have secondary groups to which they also belong. For example, number of remote connections belongs primarily to the network communication group, but it also belongs secondarily to the memory use group. This reflects the fact that some parameters have implications for a number of areas of Adaptive Server behavior. sp_configure displays parameters in all groups to which they belong.

Table 5-2 lists the configuration parameter groups.

Table 5-2: Configuration groups

Parameter group	Configures Adaptive Server for
Backup/Recovery	Backing up and recovering data
Cache manager	Data and procedure caches
Component Integration Services administration	Component Integration Services
DTM administration	Distributed transaction management (DTM) facilities
Diagnostics	Diagnostic principles
Disk I/O	Disk I/O

Parameter group	Configures Adaptive Server for
Error log	Error log, and the logging of Adaptive Server events to the Windows event log
Extended stored procedures	The behavior of extended stored procedures (ESPs).
General information	Basic system administration
Java services	Memory for Java in Adaptive Server
	See the *Java in Adaptive Server Enterprise* manual for complete information about Java in the database.
	If you use method calls to JDBC, you may need to increase the size of the execution stack available to the user. See "stack size" on page 260.
Languages	Languages, sort orders, and character sets
Lock manager	Locking
Memory use	Memory consumption
Metadata caches	Setting the metadata cache size for frequently used system catalog information. The metadata cache is a reserved area of memory used for tracking information on databases, indexes, or objects. The greater the number of open databases, indexes, or objects, the larger the metadata cache size. For a discussion of metadata caches in a memory-usage context, see Chapter 3, "Configuring Memory," in System Administration Guide: Volume 2.
Monitoring	Collecting monitoring information. By default, Adaptive Server does not collect monitoring information.
	See Chapter 2, "Monitoring Tables," in the *Performance and Tuning Guide: Monitoring and Analyzing*.
Network communication	Communication between Adaptive Server and remote servers, and between Adaptive Server and client programs
O/S resources	Use of operating system resources
Physical memory	Your machine's physical memory resources
Processors	Processors in an SMP environment
Query Tuning	Query optimization
RepAgent thread administration	Replication via Replication Server
Shared Disk Cluster	Cluster Edition.
SQL Server administration	General Adaptive Server administration.
Security related	Security-related features
Unicode	Unicode-related features
User environment	User environments

The syntax for displaying all groups and their associated parameters, and the current values for the parameters, is:

sp_configure

Note The number of parameters returned by sp_configure depends on the value to which you have your display level set. See "User-defined subsets of the parameter hierarchy: display levels" on page 74.

The following is the syntax for displaying a particular group and its associated parameter:

sp_configure "*group_name*"

For example, to display the disk I/O group, enter:

```
sp_configure "Disk I/O"

Group: Disk I/O
Parameter Name         Default Memory Used Config Value Run Value
unit         type
-------------         ------- ----------- ------------ ---------
------         ------------
allow sql server async i/o   1           0           1           1
switch       static
diable disk mirroring        1           0           1           1
switch       static
disk i/o structures        256           0         256         256
number       dynamic
number of devices           10           0          10          10
number       dynamic
number of large I/O buffers  6       12352           6           6
number       dynamic
page utilization percent    95           0          95          95
percent      dynamic
```

Note If the server uses a case-insensitive sort order, sp_configure with no parameters returns a list of all configuration parameters and groups in alphabetical order with no grouping displayed.

User-defined subsets of the parameter hierarchy: display levels

Depending on how you use Adaptive Server, you may need to adjust some parameters more frequently than others. It may be easier to work with a subset of parameters.

The default display level is comprehensive. When you set your display level, the setting persists across multiple sessions. However, you can reset it at any time.

- Basic – shows only the most basic parameters, and is appropriate for general server tuning.

- Intermediate – includes parameters that are somewhat more complex, in addition to the basic parameters.

- Comprehensive – includes all the parameters, including the most complex ones. This level is appropriate for users doing highly detailed server tuning.

The syntax for showing your current display level is:

    sp_displaylevel

To set the display level, use the following, where *user_name* is your Adaptive Server login name:

    sp_displaylevel *user_name*[, basic | intermediate | comprehensive]

The effect of the display level on *sp_configure* output

If your display level is set to either basic or intermediate, sp_configure returns only a subset of the parameters that are returned when your display level is set to comprehensive. For instance, if your display level is set to intermediate, and you want to see the parameters in the languages group, enter:

    sp_configure "Languages"

The output looks like this:

```
sp_configure
Group: Languages

Parameter Name        Default Memory Used Config Value Run Value Unit Type
---------------       ------- ----------- ------------ --------- ---- ----
default character set    1         0           1           1     id   static
default language id      0         0           0           0     id   dyna
. . .
```

This represents only a subset of the parameters in the languages group; some language parameters appear only when your display level is comprehensive.

Performance tuning with *sp_configure* and *sp_sysmon*

sp_sysmon monitors Adaptive Server performance and generates statistical information that describes the behavior of your Adaptive Server system. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

You can run sp_sysmon before and after using sp_configure to adjust configuration parameters. The output gives you a basis for performance tuning and allows you to observe the results of configuration changes.

Using configuration parameters in a clustered environment

For the Cluster Edition, Sybase supports both cluster-wide and instance-specific configuration. Cluster-wide configuration parameters are applied to all instances in the cluster. Local configuration parameters are applied only to a specified instance.

- Local configuration overrides cluster-wide configuration.

- If an instance-specific configuration has not been applied, the cluster-wide configuration applies.

- Some parameters, such as default character set id, cannot be applied to a specific instance. These parameters can only be used over an entire cluster.

The cluster configuration file includes an instance-specific configuration block. Parameter settings in the instance-specific block override cluster-wide settings. For example:

```
max online engines = DEFAULT

[Instance:ase1]
max online engines = 5
[Instance:ase2]
max online engines = 3
```

See the *Clusters Users Guide*.

sp_configure **output**

The sample output below shows the type of information sp_configure prints if your display level is comprehensive, and you execute sp_configure with no parameters. The values it prints vary, depending on your platform and on what values you have already changed.

```
sp_configure
Group: Configuration Options

Group: Backup/Recovery

Parameter Name        Default Memory Used Config Value Run Value Unit Type
--------------        ------- ----------- ------------ --------- ---- ----
allow remote access         1           0            1         1   switch dyn
print recovery info         0           0            0         0   switch dyn
recovery interval in m      5           0            5         5   minutes dyn
...
```

Note All configuration groups and parameters appears in output if your display level is set to "comprehensive."

Where:

- The "Default" column displays the default value. If you do not explicitly reconfigure a parameter, it retains its default value.

- "Memory Used" shows the amount of memory, in kilobytes, used by the parameter at its current value. Some related parameters draw from the same memory pool. For instance, the memory used for stack size and stack guard size is already accounted for in the memory used for number of user connections. If you added the memory used by each of these parameters separately, the sume is more than the amount actually used. Parameters that "share" memory with other parameters are marked with a hash mark ("#").

- "Config Value" displays the most recent value to which the configuration parameter has been set. When you execute sp_configure to modify a dynamic parameter:

  - The configuration and run values are updated.

  - The configuration file is updated.

  - The change takes effect immediately.

  When you modify a static parameter:

- • The configuration value is updated.

- • The configuration file is updated.

- • The change takes effect only when you restart Adaptive Server.

- • "Run Value" displays the value Adaptive Server is currently using. It changes when you modify a dynamic parameter's value and, for static parameters, after you restart Adaptive Server.

- • "Unit" displays the unit value of the configuration parameter. Adaptive Server displays information in the following units:

Name of unit	Unit description
number	Number of items.
clock ticks	Number of clock ticks.
microseconds	Number of microseconds.
milliseconds	Number of millisecond.s
seconds	Number of seconds.
minutes	Number of minutes.
hours	Number of hours.
bytes	Number of bytes.
days	Number of days.
kilobytes	Number of kilobytes.
megabytes	Number of megabytes.
memory pages (2K)	Number of 2K memory pages.
virtual pages (2K)	Number of 2K virtual pages.
logical pages	Number of logical pages. This value depends on the logical page size your server is using: 2, 4, 8, or 16K.
percent	Value of the configured parameter as a percentage.
ratio	Value of the configured parameter as a ratio.
switch	Value of the parameter is either TRUE (the parameter is turned on), or FALSE.
id	ID of the configured parameter you are investigating.
name	Character string name assigned to the run or configure value of the parameter. For example, "binary" appears under the "Run Value or "Config Value" column for the output of sp_configure "lock scheme".
row	Number of rows

- • "Type" displays whether the configuration option is static or dynamic. Changes to static parameters require that you restart Adaptive Server for the changes to take effect. Changes to dynamic parameters take effect immediately without having to restart Adaptive Server.

Named cache configuration parameters

The Named Cache configuration parameter group provides details for named caches:

- cache size - size of the cache. By default Adaptive Server creates 8MB caches. Change this parameter dynamically with sp_cacheconfig, or change the value in the server configuration file to have the change take place after the next server restart.

- cache status – status of the cache. One of `default data cache`, `log only`, `mixed`, or `in-memory storage`. The default data cache must have a cache status of `default data cache`, and cannot be changed. cache status for named caches can be `log only`, `mixed`, or, for in-memory databases, `in-memory storage` (you cannot change the cache status for in-memory databases).

  You cannot dynamically change the cache status in a clustered environment from `log only` on a local cache while other instance use a different cache status.

- cache replacement – describes the cache replacement policy. For named caches and default data caches, the replacement policy is `strict LRU` or `relaxed LRU`. Change this parameter dynamically with sp_cacheconfig, or change the value in the server configuration file to have the change take place after the next server restart. The cache replacement policy must be `none` for in-memory databases because they do not use buffer or page replacement.

- local cache partition number – number of cache partitions. You may partition a named cache into multiple cache partitions. The acceptable values are 0, 2, 4, 8, 16, 32, 64 or 128. You cannot change the number of cache partitions dynamically; you must restart Adaptive Server for the change to take effect.

sysconfigures and *syscurconfigs* tables

The report displayed by sp_configure is constructed mainly from the master..sysconfigures and master..syscurconfigs system tables, with additional information provided from sysattributes, sysdevices, and other system tables.

The value column in the sysconfigures table records the last value set from sp_configure or the configuration file; the value column in syscurconfigs stores the value currently in use. For dynamic parameters, the two values match; for static parameters, which require a restart of the server to take effect, the two values are different if the values have been changed since Adaptive Server was last started. The values may also be different when the default values are used. In this case, sysconfigures stores 0, and syscurconfigs stores the value that Adaptive Server computes and uses.

sp_configure performs a join on sysconfigures and syscurconfigs to display the values reported by sp_configure.

Querying *syscurconfigs* and *sysconfigures*: an example

You might want to query sysconfigures and syscurconfigs to get information organized the way you want. For example, sp_configure without any arguments lists the memory used for configuration parameters, but does not list minimum and maximum values. Use this query to get a complete list of current memory usage, as well as minimum, maximum, and default values:

```
select b.name, memory_used, minimum_value,
maximum_value, defvalue
from master.dbo.sysconfigures b,
master.dbo.syscurconfigs c
where b.config *= c.config and parent != 19
and b.config > 100
```

Configuration parameters

In many cases, the maximum allowable values for configuration parameters are usually limited by available memory, rather than by sp_configure limitations.

Note To find the maximum supported values for your platform and version of Adaptive Server, see "Adaptive Server Specifications" in the *Installation Guide* for your platform.

Alphabetical listing of configuration parameters

The following sections include both summary and detailed information about each configuration parameter.

abstract plan cache

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

abstract plan cache enables caching of abstract plan hash keys. See Chapter 12, "Creating and Using Abstract Plans" in the *Performance and Tuning Series: Query Processing and Abstract Plans*. abstract plan load must be enabled for plan caching to take effect.

abstract plan dump

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

abstract plan dump enables the saving of abstract plans to the ap_stdout abstract plans group. See Chapter 12, "Creating and Using Abstract Plans" in the *Performance and Tuning Series: Query Processing and Abstract Plans*.

abstract plan load

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)

Summary information	
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

abstract plan load enables association of queries with abstract plans in the ap_stdin abstract plans group. See Chapter 12, "Creating and Using Abstract Plans" in the *Performance and Tuning Series: Query Processing and Abstract Plans*.

abstract plan replace

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

abstract plan replace enables plan replacement for abstract plans in the ap_stdout abstract plans group. See Chapter 12, "Creating and Using Abstract Plans" in the *Performance and Tuning Series: Query Processing and Abstract Plans*. abstract plan load must be enabled for replace mode to take effect.

additional network memory

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Network Communication, Physical Memory

additional network memory sets the maximum size of additional memory that can be used for network packets that are larger than the default packet size. Adaptive Server rounds down the value you enter to the nearest 2K value. The default value indicates that no extra space is allocated for large packets.

When a login requests a large packet size, Adaptive Server verifies it has sufficient memory available to satisfy the request. If it does not, Adaptive Server finds the largest available block of memory and tries the appropriate size (which is a multiple of default network packet size) less than the largest memory block. If that fails, Adaptive Server decreases the value of the request by the number of bytes equal to default network packet size, if this is available. Adaptive Server continues for 10 iterations, or until the size equals the value of default network packet size, whichever comes first. On the tenth iteration, Adaptive Server uses the value of the default network packet size for the packet size.

If you increase max network packet size, you must increase additional network memory because all allocated network memory is reserved for users at the default size. Adaptive Server guarantees that every user connection can log in at the default packet size.

If you increase max network packet size but do not increase additional network memory, Adaptive Server does not guarantee that clients who request network packet sizes larger than the default size can login at the requested packet size.

Increasing additional network memory may improve performance for applications that transfer large amounts of data. To determine the value for additional network memory when your applications use larger packet sizes:

1   Estimate the number of simultaneous users who will request the large packet sizes, and the sizes their applications will request,

2   Multiply this sum by three, since each connection needs three buffers,

3   Add two percent for overhead for 32-bit servers, or four percent for 64-bit servers, and

4   Round the value to the next highest multiple of 2048.

For example, if you estimate these simultaneous needs for larger packet sizes:

Application	Packet size	Overhead
bcp	8192	
Client-Library	8192	
Client-Library	4096	
Client-Library	4096	
Total	24576	

Application	Packet size	Overhead
Multiply by 3 buffers/user	* 3=73728	
Compute 2% overhead		* .02=1474
Add overhead	+ 1474	
Additional network memory	75202	
Round up to multiple of 2048	75776	

Set additional network memory to 75,776 bytes.

allocate max shared memory

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Physical Memory

allocate max shared memory determines whether Adaptive Server allocates all the memory specified by max memory at start-up or only the amount of memory the configuration parameter requires.

By setting allocate max shared memory to 0, you ensure that Adaptive Server uses only the amount of shared memory required by the current configuration, and allocates only the amount of memory required by the configuration parameters at start-up, which is a smaller value than max memory.

If you set allocate max shared memory to 1, Adaptive Server allocates all the memory specified by max memory at start-up. If you set allocate max shared memory to 1, and if you increase max memory, Adaptive Server attempts to allocate the memory immediately. If the memory allocation fails, Adaptive Server writes messages to the error log. Check the error log to verify that no errors have occurred.

A successful memory allocation means that Adaptive Server always has the memory required for any memory configuration changes you make and there is no performance degradation while the server readjusts for additional memory. However, if you do not predict memory growth accurately, and max memory is set to a large value, you may waste total physical memory.

allow backward scans

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Query Tuning

allow backward scans controls how the optimizer performs select queries that contain the order by...desc command:

- When the value is set to 1, the optimizer can access the index or table rows by following the page chain in descending index order.

- When the value is set to 0, the optimizer selects the rows into a worktable by following the index page pointers in ascending order, and then sorts the worktable in descending order.

The first method—performing backward scans—can speed access to tables that need results ordered by descending column values. Some applications, however, may experience deadlocks due to backward scans. In particular, look for increased deadlocking if you have delete or update queries that scan forward using the same index. There may also be deadlocks due to page splits in the index.

Use print deadlock information to send messages about deadlocks to the error log. See "print deadlock information" on page 234. Alternatively, you can use sp_sysmon to check for deadlocking. See the *Performance and Tuning Series: Locking and Concurrency Control*.

allow nested triggers

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

allow nested triggers controls the use of nested triggers. When the value is set to 1, data modifications made by triggers can fire other triggers. Set allow nested triggers to 0 to disable nested triggers. A set option, self_recursion, controls whether the modifications made by a trigger can cause that trigger to fire again.

allow procedure grouping

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System security officer
Configuration group	Security Related

allow procedure grouping controls the ability to group stored procedures of the same name so that they can be dropped with a single drop procedure statement.

allow remote access

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration groups	Backup/Recovery, Network Communication

allow remote access controls logins from remote Adaptive Servers. The default value of 1 allows Adaptive Server to communicate with Backup Server.

Note Setting the value to 0 disables server-to-server RPCs. Since Adaptive Server communicates with Backup Server via RPCs, setting this parameter to 0 makes it impossible to back up a database.

Since other system administration actions are required to enable remote servers other than Backup Server to execute RPCs, leaving this option set to 1 does not constitute a security risk.

allow resource limits

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

allow resource limits controls the use of resource limits. When the value is set to 1, the server allocates internal memory for time ranges, resource limits, and internal server alarms. The server also internally assigns applicable ranges and limits to user sessions. The output of showplan and statistics io displays the optimizer's cost estimate for a query. Set allow resource limits to 0 to disable all resource limits.

allow sendmsg

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System security officer
Configuration group	Network Communication

allow sendmsg enables or disables sending messages from Adaptive Server to a User Datagram Protocol (UDP) port. When allow sendmsg is set to 1, any user can send messages using sp_sendmsg or syb_sendmsg. To set the port number used by Adaptive Server, see "syb_sendmsg port number" on page 267.

Note Sending messages to UDP ports is not supported on Windows.

allow sql server async i/o

Summary information	
Default value	1 (on)

Summary information	
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Disk I/O

allow sql server async i/o enables Adaptive Server to run with asynchronous disk I/O. To use asynchronous disk I/O, enable it on both Adaptive Server and your operating system. See your operating system documentation for information on enabling asynchronous I/O at the operating system level.

Disk I/O always runs faster asynchronously than synchronously. This is because when Adaptive Server issues an asynchronous I/O, it does not have to wait for a response before issuing further I/Os.

allow updates to system tables

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

allow updates to system tables enables users with the system administrator role to make changes to the system tables and to create stored procedures that can modify system tables. A database administrator can update system tables in any tables that he or she owns if allow updates to system tables is enabled.

System tables include:

- All Sybase-supplied tables in the master database

- All tables in user databases that begin with "sys" and that have an ID value in the sysobjects table of less than or equal to 100

 Warning! Incorrect alteration of a system table can result in database corruption and loss of data. To protect against errors that might corrupt your databases, always use begin transaction when modifying a system table. Immediately after finishing your modifications, disable allow updates to system tables.

Stored procedures and triggers you create while allow updates to system tables is set on can update the system tables, even after the parameter has been set off. When you set allow updates to system tables to on, you create a "window of vulnerability," a period of time during which users can alter system tables or create a stored procedure with which the system tables can be altered in the future.

Because the system tables are so critical, Sybase suggests that you set this parameter to on only in highly controlled situations. To guarantee that no other users can access Adaptive Server while the system tables can be directly updated, restart Adaptive Server in single-user mode. For details, see startserver and dataserver in the *Utility Guide*.

Note The server-wide configuration option allow updates to system tables takes precedence over the stored procedure settings for allow updates to system tables. If you do not enable allow updates to system tables at the server level, individual stored procedure settings determine whether you can modify system catalogs.

---

average cap size

Summary information	
Default value	200
Range of values	100 – 10000
Status	Static
Display level	
Required role	
Configuration group	Diagnostics

Reserved for future use.

audit queue size

Summary information	
Default value	100
Range of values	1 – 65535
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration groups	Memory Use, Security Related

The in-memory audit queue holds audit records generated by user processes until the records can be processed and written to the audit trail. To change the size of an audit queue, a system security officer can use audit queue size. When you configure the queue suze, there is a trade-off between performance and risk. If the queue is too large, records can remain in it for some time. As long as records are in the queue, they are at risk of being lost if the system fails. However, if the queue is too small, it can repeatedly become full, which affects overall system performance; user processes that generate audit records sleep if the audit queue is full.

Following are some guidelines for determining how big your audit queue should be. You must also take into account the amount of auditing to be performed at your site.

- The memory requirement for a single audit record is 424 bytes; however, a record can be as small as 22 bytes when it is written to a data page.

- The maximum number of audit records that can be lost in a system failure is the size of the audit queue (in records), plus 20. After records leave the audit queue, they remain on a buffer page until they are written to the current audit table on the disk. The pages are flushed to disk every 20 records, less if the audit process is not constantly busy.

- In the system audit tables, the extrainfo field and fields containing names are of variable length, so audit records that contain complete name information are generally larger.

The number of audit records that can fit on a page varies from 4 to as many as 80 or more. The memory requirement for the default audit queue size of 100 is approximately 42K.

auditing

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

auditing enables or disables auditing for Adaptive Server.

automatic cluster takeover

Summary information	
Default value	1
Valid values	1 (enabled), 0 (disabled)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

Setting automatic cluster takeover to 1 allows an instance that is starting to automatically recover from an abrupt total cluster failure. If you set automatic cluster takeover to 0, the cluster may not be able to recover from an abrupt cluster failover unless you include the --cluster_takeover parameter.

The Cluster Edition uses quorum heartbeats and a cluster takeover algorithm to determine when cluster takeover should be performed. This algorithm allows an instance that is starting to distinguish between an inability to join the cluster because the cluster has crashed (in which case takeover is appropriate) and an inability to join the cluster because the instance that is starting does not have network connectivity (in which case takeover is not appropriate).

If you disable automatic cluster takeover (set it to 0), The Cluster Edition writes the results of the algorithm to the error log as an advisory message and then exits.

If you enable auotomatic cluster takeover (set it to 1), the Cluster Edition starts as the cluster coordinator and recovers the databases. This is guaranteed to be a safe operation in environments that have I/O fencing enabled.

In environments without I/O fencing, a malfunction of the algorithm could introduce data corruption, so you can set the configuration parameter to 0 to disable this algorithm. However, environments without I/O fencing have a risk of data corruption, and disabling automatic cluster takeover does not mitigate all of those risks.

builtin date strings

Summary information	
Default value	0
Range of values	0 – 1
Status	Dynamic
Display level	
Required role	
Configuration group	Query tuning

If a string is given as an argument in place of the chronological value, the server interprets it as a datetime value regardless of its apparent precision. This default behavior may be changed by setting the configuration parameter builtin date strings or the set option builtin_date_strings. When these options are set the server will interpret strings given to chronological builtins as bigdatetimes.

caps per ccb

Summary information	
Default value	50
Range of values	5 – 50
Status	Static
Display level	
Required role	
Configuration group	Diagnostics

Reserved for future use.

capture compression statistics

Summary information	
Default value	0 (off)

Summary information

Range of values	1 (on), 0 (off)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	SQL Server Administration

Enables the monTableCompression monitoring table to begin capturing compression statistics.

check password for digit

Summary information

Default value	0 (off)
Range of values	1 (on), 0 (off)
Status	Dynamic
Display level	10
Required role	System security officer
Configuration group	Security Related

The system security officer can tell the server to check for at least one character or digit in a password using the server-wide configuration parameter check password for digit. If set, this parameter does not affect existing passwords.

CIPC large message pool size

Summary information

Default value	512
Valid values	512 – 2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

CIPC large message pool size specifies the number of large message buffers allocated by CIPC at start-up time.

CIPC regular message pool size

Summary information	
Default value	8192
Valid values	2048 – 2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

CIPC regular message pool size specifies the number of regular message buffer allocated by CIPC at start-up time.

cis bulk insert array size

Summary information	
Default value	50
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

When performing a bulk transfer of data from one Adaptive Server to another Adaptive Server, CIS internally buffers rows, and asks the Open Client bulk library to transfer them as a block. The size of the array is controlled by cis bulk insert array size.

cis bulk insert batch size

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis bulk insert batch size determines how many rows from the source tables are to be bulk copied into the target table as a single batch using select into.

If you leave cis bulk insert batch size at 0, all rows are copied as a single batch. Otherwise, after the count of rows specified by this parameter has been copied to the target table, the server issues a bulk commit to the target server, causing the batch to be committed.

If a normal client-generated bulk copy operation (such as that produced by the bcp utility) is received, the client is expected to control the size of the bulk batch, and the server ignores the value of this configuration parameter.

cis connect timeout

Summary information	
Default value	0
Range of values	0–32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis connect timeout determines the wait time, in seconds, for a successful Client-Library connection.

cis cursor rows

Summary information	
Default value	50
Range of values	1 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis cursor rows specifies the cursor row count for cursor open and cursor fetch operations. Increasing this value means more rows are fetched in one operation. This increases speed but requires more memory.

cis idle connection timeout

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis idle connection timeout configures Adaptive Server to check for CIS connections to any remote server that have been unused longer than the specified number of seconds. Adaptive Server deletes the unused connections and reallocates their resources.

Although the number you specify is in seconds, the housekeeper task wakes up, at most , once a minute, so idle connections may be idle for much longer than the configured value. Adaptive Server does not drop idle connections if a transaction is active on the connection, and reestablishes the connection automatically if the user executes any command that accesses the connection.

cis packet size

Summary information	
Default value	512
Range of values	512–32768
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis packet size specifies the size of Tabular Data Stream™ (TDS) packets that are exchanged between the server and a remote server when a connection is initiated.

The default packet size on most systems is 512 bytes, and this may be adequate for most applications. However, larger packet sizes may result in significantly improved query performance, especially when text, unitext, and image or bulk data is involved.

If you specify a packet size larger than the default, then the target server must be configured to allow variable-length packet sizes, using:

- additional netmem

- maximum network packet size

cis rpc handling

Summary information	
Default value	0 (off), default value of 1 for the Cluster Edition
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

cis rpc handling specifies the default method for remote procedural call (RPC) handling. Setting cis rpc handling to 0 sets the Adaptive Server site handler as the default RPC handling mechanism. Setting the parameter to 1 forces RPC handling to use Component Integration Service access methods. See set cis rpc handling in the *Component Integration Services Users Guide*.

cluster heartbeat interval

Summary information	
Default value	10
Valid values	1– 127
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

cluster heartbeat interval controls the interval that cluster instances use to send and check the heartbeat status.

Using a lower value for cluster heartbeat interval reduces the failure detection time but increases the risk of a false failure because of a transient problem (such as an overloaded CPU). Tuning cluster heartbeat interval to a larger value reduces the risk of a false failure but increases the time needed to detect a failure.

cluster heartbeat retries

Summary information	
Default value	1
Valid values	1– 127
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

cluster heartbeat retries controls the number of times an instance retries a failed cluster heartbeat before entering failure mode.

Tuning cluster heartbeat retries to a lower value reduces the time to detect failure but increases the risk of a false failure because of a transient problem (such as an overloaded CPU). Tuning cluster heartbeat retries to a larger value reduces the risk of a false failure but increases the time needed to detect a failure.

cluster redundancy level

Summary information	
Default value	1
Valid values	1 to n – 1, where n is the maximum number of instances specified in *cluster.cfg* or the quorum file.
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

cluster redundancy level (CRL) controls the number of recoverable failed instances in a shared-disk cluster. It is the maximum number of instances that can fail simultaneously while allowing recovery to proceed concurrently with other activity. If the number of failed instances exceeds the maximum number specified by the CRL, the cluster shuts down.

For the cluster to start, the value for cluster redundancy level must be at least one less than the value of maximum number of instances specified in *cluster.cfg* or the quorum file. Thus, the cluster cannot start if you set either of these:

- The value of maximum number of instances to a value that is equal to or less than the value of cluster redundancy level

- The value of cluster redundancy level to a value equal to or greater than the value of maximum number of instances

When the cluster redundancy level value increases, so does the messaging traffic, as there are multiple copies of each lock in the cluster, and an increase in overhead required to maintain this redundancy level.

Other configuration parameters, such as number of locks and cache size need more resources for cluster redundancy level values greater than 1, which means you must increase max memory for the same number of locks value.

Sybase recommends that you set cluster redundancy level to 1 if the failure of multiple instances is expected to be uncommon.

cluster vote timeout

Summary information	
Default value	60
Valid values	1– 127
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

cluster vote timeout controls the maximum amount of time an instance waits for other instances to vote during the voting period. An instance waits only for those instances which it believes are running.

Tuning cluster vote timeout to a lower value can reduce failover time, but increases the risk that an instance that is running is excluded from the new cluster view. Tuning cluster vote timeout to a larger value reduces the risk that an running instance is excluded from the new cluster view, but may increase failover time.

column default cache size

Summary information	
Default value	1024 pages (2MB)
Range of values	128 – 8192 (pages)

Summary information	
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	User Environment

column defult cache size determines the size of the cache that Adaptive Server must keep in memory to provide defaults for nonmaterialized columns.

compression info pool size

Summary information	
Default value	4096
Range of values	0 – 2147483647
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	SQL Server Administration

Determines the size of the memory pool used for compression.

compression memory size

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Physical Memory

Used while loading a compressed dump into an archive database. compression memory size determines the size (in 2KB pages) of the memory pool Adaptive Server uses to decompress a compressed dump. When you set compression memory size to 0, no pool is created and a compressed dump cannot be loaded.

See "Creating a compression memory pool" in Chapter 14, "Archive Database Access," in the *System Administration Guide: Volume 2*.

configuration file

Summary information	
Default value	0 (off)
Range of values	One of: 0, verify, read, write, or restore
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	General Information

configuration file specifies the location of the configuration file currently in use. See "Using sp_configure with a configuration file" on page 68 for a complete description of configuration files.

In sp_configure output, the "Run Value" column displays only 10 characters, so the output may not display the entire path and name of your configuration file.

cost of a logical io

Summary information	
Default value	2
Range of values	0 – 254
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

cost of a logical io specifies the cost of a single logical I/O.

cost of a physical io

Summary information	
Default value	25
Range of values	0 – 254
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

cost of a phsyical io specifies the cost of a single physical I/O.

cost of a cpu unit

Summary information	
Default value	1000
Range of values	1 – 65534
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

cost of a cpu unit specifies the cost of a single CPU operation.

The cost of a serial plan in the optimizer is described by this formula:

```
Cost = PIO X estimated_pio + LIO X estimated_lio + 100 X estimated_cpu / CPU
```

Where the default values are:

• *estimated_pio* = 25

• *estimated_lio* = 2

• *estimated_cpu* = 1000

If your Adaptive Server has sufficient memory, then all tables exist in memory, and a value of 0 for cost of a physical io is appropriate.

If your CPU is fast enough so the value for cost of a cpu unit is not a issue, use this formula to determine the cost of CPU, which combines 2 LIO and 25 PIO (the default values):

```
CPU X 100/configuration_value
```

The default value for *configuration_value* is 1000.

As you increase the value for cost of a cpu unit, this formula reduces the impact of CPU on cost.

cpu accounting flush interval

Summary information	
Default value	200
Range of values	1–2147483647

Summary information

Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

cpu accounting flush interval specifies the amount of time, in machine clock ticks (non-Adaptive Server clock ticks), that Adaptive Server waits before flushing CPU usage statistics for each user from sysprocesses to syslogins, a procedure used in charge-back accounting.

When a user logs in to Adaptive Server, the server begins accumulating figures for CPU usage for that user process in sysprocesses. When a user logs off Adaptive Server, or when the value of cpu accounting flush interval is exceeded, the accumulated CPU usage statistics are flushed from sysprocesses to syslogins. These statistics continue accumulating in syslogins until you clear the totals. Display the current totals from syslogins using sp_reportstats.

The value to which you set cpu accounting flush interval depends on the type of reporting you intend to do. If you run reports on a monthly basis, set cpu accounting flush interval to a relatively high value. With infrequent reporting, it is less critical that the data in syslogins be updated frequently.

However, if you perform periodic ad hoc selects on the totcpu column in syslogins to determine CPU usage by process, set cpu accounting flush interval to a lower value to increase the likelihood of the data in syslogins being up-to-date when you execute your selects.

Setting cpu accounting flush interval to a low value may cause the lock manager to mistakenly identify processes as potential deadlock victims. When the lock manager detects a deadlock, it checks the amount of CPU time accumulated by each competing processes. The process with the lesser amount is chosen as the deadlock victim and is terminated by the lock manager. Additionally, when cpu accounting flush interval is set to a low value, the task handlers that store CPU usage information for processes are initialized more frequently, thus making processes appear as if they have accumulated less CPU time than they actually have. Because of this, the lock manager may select a process as the deadlock victim when, in fact, that process has more accumulated CPU time than the competing process.

If you do not intend to report on CPU usage at all, set cpu accounting flush interval to its maximum value. This reduces the number of times syslogins is updated, and reduces the number of times its pages must be written to disk.

cpu grace time

Summary information	
Default value	500
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

cpu grace time, together with time slice, specifies the maximum amount of time that a user process can run without yielding the CPU before Adaptive Server preempts it and terminates it with a timeslice error. The units for cpu grace time are time ticks, as defined by sql server clock tick length. See "sql server clock tick length" on page 256.

When a process exceeds cpu grace time Adaptive Server "infects" it by removing the process from the internal queues. The process is killed, but Adaptive Server is not affected. This prevents runaway processes from monopolizing the CPU. If any of your user processes become infected, you may be able to temporarily fix the problem by increasing the value of cpu grace time. However, be sure that the problem really is a process that takes more than the current value of cpu grace time to complete, rather than a runaway process.

Temporarily increasing the cpu grace time value is a workaround, not a permanent fix, since it may cause other complications; see "time slice" on page 271. Also, see Chapter 4, "Using Engines and CPUs" in the *Performance and Tuning Series: Basics* for a more detailed discussion of task scheduling.

current audit table

Summary information	
Default value	1
Range of values	0–8
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

current audit table establishes the table where Adaptive Server writes audit rows. A system security officer can change the current audit table, using:

```
sp_configure "current audit table", n
  [, "with truncate"]
```

where n is an integer that determines the new current audit table, as follows:

- 1 means sysaudits_01, 2 means sysaudits_02, and so forth, up to 8.

- 0 tells Adaptive Server to set the current audit table to the next table. For example, if your installation has three audit tables, sysaudits_01, sysaudits_02, and sysaudits_03, Adaptive Server sets the current audit table to:

  - 2 if the current audit table is sysaudits_01

  - 3 if the current audit table is sysaudits_02

  - 1 if the current audit table is sysaudits_03

"with truncate" specifies that Adaptive Server should truncate the new table if it is not already empty. sp_configure fails if this option is not specified and the table is not empty.

Note If Adaptive Server truncates the current audit table, and you have not archived the data, the table's audit records are lost. Be sure that the audit data is archived before using the with truncate option.

To execute sp_configure to change the current audit table, you must have the sso_role active. You can write a threshold procedure to change the current audit table automatically.

deadlock checking period

Summary information	
Default value	500
Range of values	0–2147483
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

deadlock checking period specifies the minimum amount of time (in milliseconds) before Adaptive Server initiates a deadlock check for a process that is waiting on a lock to be released. Deadlock checking is time-consuming overhead for applications that experience no or very few deadlocks, and the overhead grows as the percentage of lock requests that must wait for a lock also increases.

If you set deadlock checking period to a nonzero value (*n*), Adaptive Server initiates a deadlock check after a process waits at least *n* milliseconds. For example, you can make a process wait at least 700 milliseconds for a lock before each deadlock check by entering:

```
sp_configure "deadlock checking period", 700
```

If you set deadlock checking period to 0, Adaptive Server initiates deadlock checking when each process begins to wait for a lock. Any value less than the number of milliseconds in a clock tick is treated as 0. See "sql server clock tick length" on page 256.

Configuring deadlock checking period to a higher value produces longer delays before deadlocks are detected. However, since Adaptive Server grants most lock requests before this time elapses, the deadlock checking overhead is avoided for those lock requests. If your applications deadlock infrequently, set deadlock checking period to a higher value. Otherwise, the default value of 500 should suffice.

Use sp_sysmon to determine the frequency of deadlocks in your system and the best setting for deadlock checking period. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

deadlock pipe active

Summary information	
Default value	0
Range of values	0–1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

deadlock pipe active controls whether Adaptive Server collects deadlock messages. If both deadlock pipe active and deadlock pipe max messages are enabled, Adaptive Server collects the text for each deadlock. Use monDeadLock to retrieve these deadlock messages.

deadlock pipe max messages

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

deadlock pipe max messages determines the number of deadlock messages Adaptive Server stores per engine. The total number of messages in the monSQLText table is the value of sql text pipe max messages times the number of engines running.

deadlock retries

Summary information	
Default value	5
Range of values	0–2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

deadlock retries specifies the number of times a transaction can attempt to acquire a lock when deadlocking occurs during an index page split or shrink.

For example, Figure 5-1 illustrates the following scenario:

- Transaction A locks page 1007 and needs to acquire a lock on page 1009 to update the page pointers for a page split.

- Transaction B is also inserting an index row that causes a page split, holds a lock on page 1009, and needs to acquire a lock on page 1007.

In this situation, rather than immediately choosing a process as a deadlock victim, Adaptive Server relinquishes the index locks for one of the transactions. This often allows the other transaction to complete and release its locks.

For the transaction that surrendered its locking attempt, the index is rescanned from the root page, and the page split operation is attempted again, up to the number of times specified by deadlock retries.

Figure 5-1: Deadlocks during page splitting in a clustered index
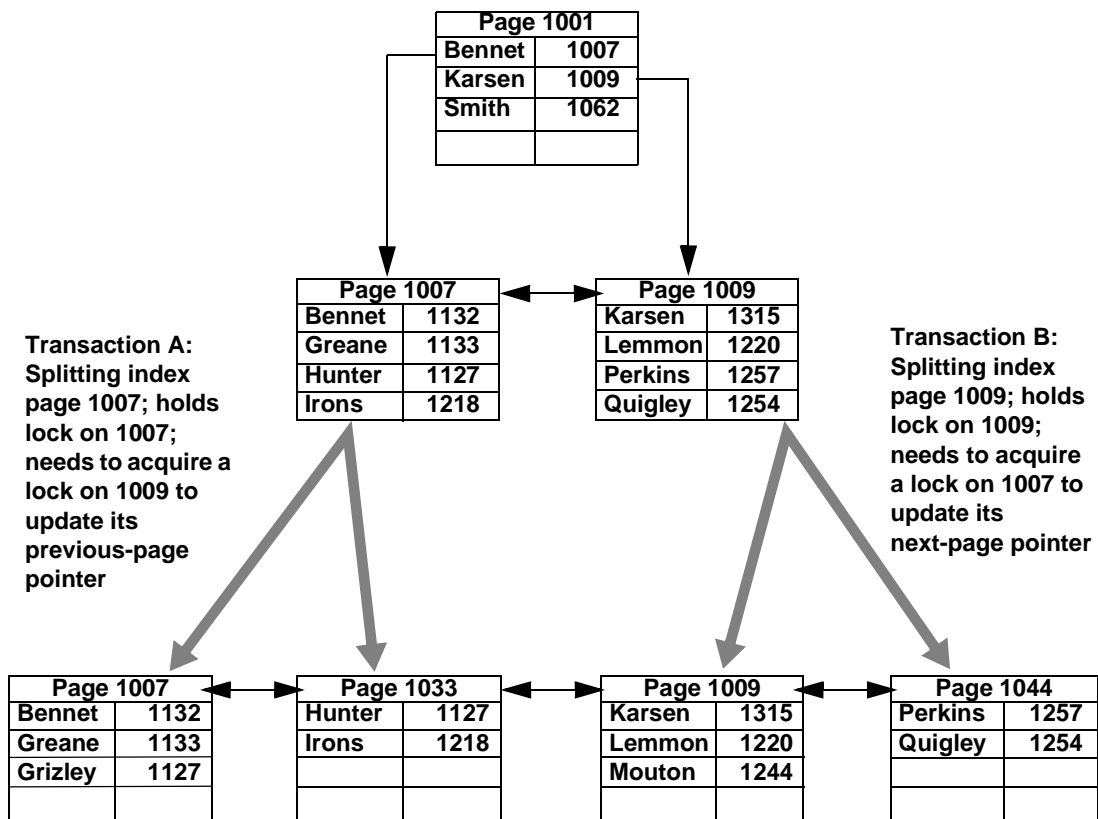
sp_sysmon reports on deadlocks and retries. See the *Performance and Tuning Series: Locking and Concurrency Control*.

default character set id

Summary information	
Default value	1
Range of values	0–255
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	Languages

default character set id specifies the number of the default character set used by the server. The default is set at installation, and can be changed later with the Sybase installation utilities. See Chapter 10, "Configuring Character Sets, Sort Orders, and Languages."

default database size

Summary information	
Default value	3MB
Range of values	2^a –10000 a. Minimum determined by server's logical page size.
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

default database size sets the default number of megabytes allocated to a new user database if create database is issued without any size parameters. A database size given in a create database statement takes precedence over the value set by this configuration parameter.

If most of the new databases on your Adaptive Server require more than one logical page size, you may want to increase the default.

Note If you alter the model database, you must also increase the default database size, because the create database command copies model to create a new user database.

default exp_row_size percent

Summary information	
Default value	5
Range of values	0–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

default exp_row_size percent reserves space for expanding updates in data-only-locked tables, to reduce row forwarding. An "expanding update" is any update to a data row that increases the length of the row. Data rows that allow null values or that have variable-length columns may be subject to expanding updates. In data-only-locked tables, expanding updates can require row forwarding if the data row increases in size so that it no longer fits on the page.

The default value sets aside 5 percent of the available data page size for use by expanding updates. Since 2002 bytes are available for data storage on pages in data-only-locked tables, this leaves 100 bytes for expansion. This value is applied only to pages for tables that have variable-length columns.

Setting default exp_row_size percent to 0 means that all pages are completely filled and no space is left for expanding updates.

default exp_row_size percent is applied to data-only-locked tables with variable-length columns when exp_row_size is not explicitly provided with create table or set with sp_chgattribute. If a value is provided with create table, that value takes precedence over the configuration parameter setting. See the *Performance and Tuning Series: Locking and Concurrency Control*.

default fill factor percent

Summary information	
Default value	0
Range of values	0–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

default fill factor percent determines how full Adaptive Server makes each index page when it is creating a new index on existing data, unless the fill factor is specified in the create index statement. The fillfactor percentage is relevant only when the index is created. As data changes, pages are not maintained at any particular level of fullness.

default fill factor percent affects:

- The amount of storage space used by your data – Adaptive Server redistributes the data as it creates the clustered index.

- Performance – splitting up pages uses Adaptive Server resources.

There is seldom a reason to change default fill factor percent, especially since you can override it in the create index command. See "create index" in the *Reference Manual: Commands*.

default language id

Summary information	
Default value	0
Range of values	0–32767
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Languages

default language id is the number of the language that is used to display system messages unless a user has chosen another language from those available on the server. us_english always has an ID of NULL. Additional languages are assigned unique numbers as they are added.

default network packet size

Summary information	
Default value	2048
Range of values	512– 65024
Status	Static
Display level	Intermediate
Required role	System administrator

Summary information	
Configuration groups	Memory Use, Network Communication, User Environment

default network packet size configures the default packet size for all Adaptive Server users. You can set default network packet size to any multiple of 512 bytes; values that are not even multiples of 512 are rounded down.

Memory for all users who log in with the default packet size is allocated from the Adaptive Server memory pool, as set with total logical memory. This memory is allocated for network packets when Adaptive Server is started.

Each Adaptive Server user connection uses:

• One read buffer

• One buffer for messages

• One write buffer

Each of these buffers requires default network packet size bytes. The total amount of memory allocated for network packets is:

```
(number of user connections + number of worker processes) * 3 * default network
packet size
```

For example, if you set default network packet size to 1024 bytes, and you have 50 user connections and 20 worker processes, the amount of network memory required is:

(50 + 20) * 3 * 1024 = 215040 bytes

If you increase default network packet size, you must also increase max network packet size to at least the same size. If the value of max network packet size is greater than the value of default network packet size, increase the value of additional network memory. See "additional network memory" on page 82.

Use sp_sysmon to see how changing the default network packet size parameter affects network I/O management and task switching. For example, try increasing default network packet size and then checking sp_sysmon output to see how this affects bcp for large batches. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

Requesting a larger packet size at login

Clients automatically use the Adaptive Server default packet size. Use the -A flag to Adaptive Server client programs to request a large packet size. For example:

```
isql -A2048
```

default sortorder id

Summary information	
Default value	50
Range of values	0–255
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Languages

default sortorder id is the number of the sort order that is installed as the default on the server. To change the default sort order, see Chapter 10, "Configuring Character Sets, Sort Orders, and Languages."

default unicode sortorder

Summary information	
Default value	binary
Range of values	Not currently used
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Unicode

default unicode sortorder is a string parameter that uniquely defines the default Unicode sort order installed on the server. To change the Unicode default sort order, see Chapter 10, "Configuring Character Sets, Sort Orders, and Languages."

default XML sortorder

Summary information	
Default value	binary
Range of values	(not currently used)
Status	Static
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration group	Unicode

default XML sortorder is a string parameter that defines the sort order used by the XML engine. A string parameter is used rather than a numeric parameter to guarantee a unique ID. See Chapter 6, "XML Support for I18N" in *XML Services in Adaptive Server Enterprise*.

deferred name resolution

Summary information	
Default value	0 (disabled)
Range of values	0 to 1
Status	dynamic
Required role	System administrator
Configuration group	Query tuning

When deferred name resolution is active (1), deferred name resolution is applied globally to all server connections; all procedures you create in the server are created using deferred name resolution.

Therefore, the stored procedures are created without resolving the objects referenced inside the stored procedure, postponing object resolution processing to the execution time. See Chapter 17, "Using Stored Procedures," in the *Transact-SQL Users Guide*

disable character set conversions

Summary information	
Default value	0 (enabled)
Valid values	0 (enabled), 1 (disabled)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Languages

Changing disable character set conversions to 1 turns off character set conversion for data moving between clients and Adaptive Server. By default, Adaptive Server performs conversion on data moving to and from clients that use character sets that are different than the server's. For example, if some clients use Latin-1 (iso_1) and Adaptive Server uses Roman-8 (roman8) as its default character set, data from the clients is converted to Roman-8 when being loaded into Adaptive Server. For clients using Latin-1, the data is reconverted when it is sent to the client; for clients using the same character set as Adaptive Server, the data is not converted.

By setting disable character set conversions, you can request that no conversion take place. For example, if all clients are using a given character set, and you want Adaptive Server to store all data in that character set, set disable character set conversions to 1, and no conversion takes place.

disable disk mirroring

Summary information	
Default value	1
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Disk I/O

disable disk mirroring enables or disables disk mirroring for Adaptive Server. This is a global variable; Adaptive Server does not perform any disk mirroring after this configuration parameter is set to 1 and Adaptive Server is restarted. Setting disable disk mirroring to 0 enables disk mirroring.

Note You must disable disk mirroring if your Adaptive Server is configured for failover.

disable varbinary truncation

Summary information	
Default value	0 (on)
Range of values	0 (on), 1 (off)
Status	Dynamic

Summary information	
Display level	Comprehensive
Required role	System Administrator
Configuration group	SQL Server Administration

disable varbinary truncation controls whether Adaptive Server includes trailing zeros at the end of varbinary or binary null data.

disk i/o structures

Summary information	
Default value	256
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Disk I/O, Memory Use

disk i/o structures specifies the initial number of disk I/O control blocks Adaptive Server allocates at start-up.

User processes require a disk I/O control block before Adaptive Server can initiate an I/O request for the process. The memory for disk I/O control blocks is preallocated when Adaptive Server starts. To minimize the chance of running out of disk I/O structures, you should configure disk i/o structures to as high a value as your operating system allows. See your operating system documentation for information on concurrent disk I/Os.

Use sp_sysmon to determine whether to allocate more disk I/O structures. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*. You can set the max async i/os per server configuration parameter to the same value as disk i/o structures. See "max async i/os per server" on page 167.

DMA object pool size

Summary information	
Default value	4096
Valid values	2048 – 2147483647
Status	Static

Summary information	
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared disk cluster

DMA object pool size specifies the number of DMA (direct memory access) objects allocated by CIPC at start-up time.

dtm detach timeout period

Summary information	
Default value	0 (minutes)
Valid values	0 – 2147483647 (minutes)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	DTM Administration

dtm detach timeout period sets the amount of time, in minutes, that a distributed transaction branch can remain in the detached state. In some X/Open XA environments, a transaction may become detached from its thread of control (usually to become attached to a different thread of control). Adaptive Server permits transactions to remain in a detached state for the length of time specified by dtm detach timeout period. After this time has passed, Adaptive Server rolls back the detached transaction.

dtm lock timeout period

Summary information	
Default value	300 (seconds)
Valid values	1 – 2147483647 (seconds)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	DTM Administration

dtm lock timeout period sets the maximum amount of time, in seconds, that a distributed transaction branch waits for lock resources to become available. After this time has passed, Adaptive Server considers the transaction to be in a deadlock situation, and rolls back the transaction branch that triggered the deadlock. This ultimately rolls back the entire distributed transaction.

Distributed transactions may potentially deadlock themselves if they propagate a transaction to a remote server, and in turn, the remote server propagates a transaction back to the originating server. This situation is shown in Figure 5-2. The work of distributed transaction "dxact1" is propagated to Adaptive Server 2 via "rpc1." Adaptive Server 2 then propagates the transaction back to the coordinating server via "rpc2." "rpc2" and "dxact1" share the same gtrid but have different branch qualifiers, so they cannot share the same transaction resources. If "rpc2" is awaiting a lock held by "dxact1," a deadlock situation exists.

Figure 5-2: Distributed transaction deadlock
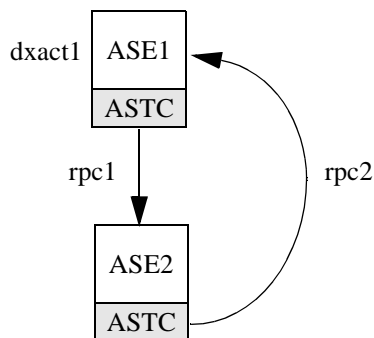
Adaptive Server cannot detect interserver deadlocks. Instead, it relies on dtm lock timeout period. In Figure 5-2, after dtm lock timeout period has expired, the transaction created for "rpc2" is aborted. This causes Adaptive Server 2 to report a failure in its work, and "dxact1" is ultimately aborted as well.

The value of dtm lock timeout period applies only to distributed transactions. Local transactions may use a lock timeout period with the server-wide lock wait period parameter.

Note Adaptive Server does not use dtm lock timeout period to detect deadlocks on system tables.

dump on conditions

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Group Diagnostics

dump on conditions determines whether Adaptive Server generates a dump of data in shared memory when it encounters the conditions specified in maximum dump conditions.

Note The dump on conditions parameter is included for use only by Sybase Technical Support. Do not modify it unless you are instructed to do so by Sybase Technical Support.

dynamic allocation on demand

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Physical Memory

dynamic allocation on demand determines when memory is allocated for changes to dynamic memory configuration parameters.

If you set dynamic allocation on demand to 1, memory is allocated only as it is needed. That is, if you change the configuration for number of user connections from 100 to 200, the memory for each user is added only when the user connects to the server. Adaptive Server continues to add memory until it reaches the new maximum for user connections.

If dynamic allocation on demand is set to 0, all the memory required for any dynamic configuration changes is allocated immediately. That is, when you change the number of user connections from 100 to 200, the memory required for the extra 100 user connections is immediately allocated.

enable backupserver HA

Summary information	
Default value	1
Valid values	1 (enabled), 0 (disabled)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared disk cluster

Setting enable backupserver HA to 1 starts the high availability Backup Server for the cluster. Setting enbale backupserver HA to 0 disables the high availability Backup Server on the cluster.

enable cis

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

enable cis enables or disables Component Integration Service.

enable compression

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive

Summary information

Required role	System administrator
Configuration group	SQL Server Administration

enable cis enables or disables data compression. See the *Compression Users Guide*.

enable console logging

Summary information

Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration groups	SQL Server Administration

enable console logging determines whether Adaptive Server sends messages to the console. By default, enable console logging is disabled, and after startup, Adaptive Server sends no messages to the console (the messages are still sent to the errorlog). Once enabled, enable console logging allows Adaptive Server to send messages to the console and error log at all times.

enable DTM

Summary information

Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Static
Display level	10
Required role	System administrator
Configuration groups	DTM Administration, SQL Server Administration

enable DTM enables or disables the Adaptive Server distributed transaction management (DTM) feature. When DTM is enabled, you can use Adaptive Server as a resource manager in X/Open XA and MSDTC systems. You must restart the server for this parameter to take effect. See the *XA Interface Integration Guide for CICS, Encina, and TUXEDO* for more information about using Adaptive Server in an X/Open XA environment. See *Using Adaptive Server Distributed Transaction Management Features* for information about transactions in MSDTC environments, and for information about Adaptive Server native transaction coordination services.

enable encrypted columns

Summary information	
Default value	0 (off)
Range of values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Security Related

enable encrypted columns enables encrypted columns.

You cannot set enable encrypted columns unless you have purchased, installed, and registered the ASE_ENCRYPTION license on your server. Any attempt to set it without such licensing results in Msg. 10834:

```
Configuration parameter 'enable encrypted columns'
cannot be enabled without license 'ASE_ENCRYPTION'
```

Note Using encrypted columns increases the logical memory used by 8198 kilobytes.

enable enterprise java beans

Summary information	
Default value	0 (off)
Range of values	0 (off), 1(on)
Status	Static
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration group	Java Services

enable enterprise java beans enables and disables EJB Server in the Adaptive
Server database. You cannot use EJB Server until the Adaptive Server is
enabled for EJB Server.

enable file access

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

enable file access enables access through proxy tables to the external file
system. Requires a license for ASE_XFS.

enable full-text search

Summary information	
Default value	1
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Component Integration Services

enable full-text search enables Enhanced Full-Text Search services. Requires a
license for ASE_EFTS.

enable functionality group

Summary information	
Default value	0

Summary information	
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

enable functionality group enables or disables the changes available for these features in Adaptive Server versions 15.7 and later:

- Shareable inline defaults – when enable functionality group is set to 0, Adaptive Server does not create shareable inline defaults, and does not reuse existing shareable inline defaults. The columns that share inline defaults (before you change this configuration parameter) continue to share the defaults until the defaults are removed.

- select for update acquiring exclusive locks

- Quoted identifiers

- Unicode noncharacters

- Monitoring cursor statements

- Reducing query processing latency

- Suppressing Job Scheduler max task messages

The default value for the parameters in this group depends on the value to which enable functionality group is set. A value of DEFAULT for the individual configuration parameters in this group—other than enable functionality group—means they are set to the same value as enable functionality group. That is, if you set enable functionality group to 1, a value of DEFAULT for any other configuration parameter in the group is 1.

Aside from the value for enable functionality group, you can ignore values of DEFAULT in the output from sp_configure and sp_helpconfig for individual configuration parameters in the Application Functionality group.

enable inline default sharing

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic

Summary information	
Display level	Comprehensive
Required role	System administrator
Configuration group	Application Functionality

enable inline default sharing enables Adaptive Server to share inline defaults. That is, once this configuration is enabled, Adaptive Server looks for existing shareable inline defaults having the same value in the database belonging to the same user. If it finds an existing shareable default, Adaptive Server binds this object to the column instead of creating a new default. However, if Adaptive Server does not find an existing shareable inline default, it creates a new default

See "Defining Defaults and Rules for Data" in the *Transact-SQL Users Guide*.

enable HA

Summary information	
Default value	0 (off)
Range of values	0 – 2
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

Set enable HA to 1 to configure Adaptive Server as a companion server in an active-active high availability subsystem. Set enable HA to 2 to configure Adaptive Server as a companion server in an active-passive high availability subsystem.

Adaptive Server uses Sybase Failover to interact with the high availability subsystem. You must set enable HA to 1 before you run the *installhasvss* script (*insthasv* on Windows), which installs the system procedures for Sybase Failover.

Note The license information and the run value for enable HA are independent of each other. Whether or not you have a license for Sybase Failover, the run value and the config value are set to 1 when you restart Adaptive Server. Until you have a license, you cannot run Sybase Failover. If you have not installed a valid license, Adaptive Server logs an error message and does not activate the feature. See the installation guide for your platform for information about installing license keys.

Setting enable HA to 1 or 2 does not mean that Adaptive Server is configured to work in a high availability system. You must perform the steps described in *Using Sybase Failover in a High Availability System* to configure Adaptive Server to be a companion server in a high availability system.

When enable HA is set to 0, you cannot configure for Sybase Failover, and you cannot run *installhasvss* (*insthasv* on Windows).

enable housekeeper GC

Summary information	
Default value	1 (on)
Range of values	0 – 5
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

The housekeeper garbage collection task performs space reclamation on data-only-locked tables. When a user task deletes a row from a data-only-locked table, a task is queued to the housekeeper to check the data and index pages for committed deletes.

The housekeeper garbage collection task is controlled by enable housekeeper GC. See Chapter 3, "Using Engines and CPUs" in the *Performance and Tuning Series: Basics*.

These are valid values for enable housekeeper GC:

- 0 – disables the housekeeper garbage collection task, but enables the delete command's lazy garbage collection. You must use reorg reclaim_space to deallocate empty pages. This is the cheapest option with the lowest performance impact, but it may cause performance problems if many empty pages accumulate. Sybase recommends that you do not use this value.

- 1 – enables lazy garbage collection for the housekeeper garbage collection task and the delete command. If more empty pages accumulate than your application allows, consider options 4 or 5. You can use the optdiag utility to obtain statistics of empty pages.

- 2 – reserved for future.

- 3 – reserved for future.

- 4 – enables aggressive garbage collection for the housekeeper garbage collection task and the delete command. This option is the most effective, but the delete command is expensive. This option is ideal if the deletes on your DOL tables are in a batch.

- 5 – enables aggressive garbage collection for the housekeeper, and lazy garbage collection for the delete command. This option is less expensive for deletes than option 4. This option is suitable when deletes are caused by concurrent transactions

sp_sysmon reports on how often the housekeeper garbage collection task performed space reclamation and how many pages were reclaimed. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

enable hp posix async i/o

Summary information	
Default value	0
Valid values	1 (enabled), 0 (disabled)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Disk I/O

Setting enable hp posix async i/o to 1 enables asynchronous I/O on database devices created on on HP-UX 11.31 and later file systems. You must first enable Adaptive Server to use asynchronous I/O with the allow sql server async i/o configuration parameter.

The combination of these configuration parameters determines whether Adaptive Server uses asynchronous or synchronous I/O for raw partitions and file systems:

enable hp posix async i/o	allow sql server async i/o = 0	allow sql server async i/o = 1
Set to 0	Synchronous I/O for file systems	Synchronous I/O for file systems
	Synchronous I/O for raw devices	Uses /dev/async asynchronous I/O for raw devices
Set to 1	Synchronous I/O for file systems	POSIX asynchronous I/O for file systems
	Synchronous I/O for raw devices	POSIX asynchronous I/O for raw devices

Note enable hp posix async i/o improves performance when you allocate database devices on file systems, but may decrease performance on database devices that are allocated on raw devices.

enable i/o fencing

Summary information	
Default value	0
Valid values	1 (enabled), 0 (disabled)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

Setting enable i/o fencing to 1 enables I/O fencing for each database device that supports the SCSI-3 Persistent Group Reservation (PGR) standard.

enable java

Summary information	
Default value	0 (disabled)
Range of values	0 (disabled), 1 (enabled)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Java Services

enable java enables and disables Java in the Adaptive Server database. You cannot install Java classes or perform any Java operations until the server is enabled for Java.

enable job scheduler

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

enable job scheduler determines whether Job Scheduler starts when Adaptive Server starts.

enable ldap user auth

Summary information	
Default value	0 (off)
Valid values	0 (off) – allows only syslogins authentication.
	1 (on) – allows both LDAP and syslogins authentication.
	2 (on) – allows only LDAP authentication.
Status	Dynamic
Display level	Comprehensive
Required role	System security officer
Configuration group	Security Related

When enable ldap user auth is 1, Adaptive Server searches the LDAP server to authenticate each user. If the LDAP authentication fails, Adaptive Server searches syslogins to authenticate the user. Use level 1 when you are migrating users from Adaptive Server authentication to LDAP authentication.

enable literal autoparam

Summary information	
Default value	0
Range of values	1 (enabled), 0 (disabled)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Query Tuning

enable literal autoparam enables and disables literal server-wide parameterization.

enable logins during recovery

Summary information	
Default value	1
Range of values	0 (enabled), 1 (disabled)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Security Related

enable logins during recovery determines whether non-system administrator logins are allowed during database recovery. A value of 1 indicates that logins are allowed during recovery, and a value of 0 indicates that logins are not allowed during recovery, that is, only the system administrator can log in to Adaptive Server.

enable merge join

Summary information	
Default value	2
Range of values	0 – disables merge joins at the server level.
	1 – enables merge joins at the server level.
	2 – sets merge joins to their default values at the server level.
Status	Dynamic
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration group	Query Tuning

enable merge join enables or disables merge join at the server level.

The default value for merge join depends on current value of the optimization goal configuration parameter:

Value for optimization goal	Default value for merge join
allrows_mix	on
allrows_dss	on
allrows_oltp	off

enable metrics capture

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

enable metrics capture enables Adaptive Server to capture metrics at the server level. Metrics for ad hoc statements are captured in the system catalogs; metrics for statements in a stored procedure are saved in the procedure cache.

enable monitoring

Summary information	
Default value	0 (off)
Range of values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

enable monitoring controls whether Adaptive Server collects the monitoring table data. enable monitoring acts as a master switch that determines whether any Monitoring configuration parameters are enabled.

Note Any data that populates the monitoring tables without enabling enable monitoring should not be considered valid.

enable pam user auth

Summary information	
Default value	0 (off)
Range of values	0 (off) – allows only syslogins authentication.
	1 (on) – allows both PAM and syslogins authentication.
	2 (on) – allows only PAM authentication.
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

enable pam user auth controls the ability to authenticate users using pluggable authentication modules (PAM).

When enable pam user auth is set to 1, Adaptive Server uses the PAM provider to authenticate each user. If the PAM authentication fails, Adaptive Server searches syslogins to authenticate the user. Use level 1 when you are migrating users from Adaptive Server authentication to PAM authentication.

enable pci

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on), 2 (on with operating system override)
Status	Dynamic
Display level	Intermediate
Required role	System Administrator
Configuration group	User Environment

enable pci enables or disables the Java PCI Bridge for Adaptive Server.

Note Do not use setting "2" (on with operating system override) unless instructed to do so by Sybase Technical Support. This setting enables the PCI Bridge on operating system versions that may not fully or correctly support PCI functionality.

---

enable permissive unicode

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Application Functionality

enable inline default sharing allows Adaptive Server to ignore Unicode noncharacters.

See "Allowing Unicode noncharacters" on page 372.

enable query tuning mem limit

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Query Tuning

enable query tuning mem limit enables the query tuning memory limit.

enable query tuning time limit

Summary information	
Default value	1 (on)

Summary information	
Range of values	0 (off), 1 (on)
Status	Intermediate
Display level	Intermediate
Required role	System administrator
Configuration group	Query Tuning

enable query tuning time limit enables the query tuning time limit.

enable real time messaging

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

enable real time messaging enables the real time messaging services.

enable rep agent threads

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Rep Agent Thread Administration

enable rep agent threads enables the RepAgent thread within Adaptive Server.

Other steps are also required to enable replication. For more information, see the Replication Server documentation.

enable row level access control

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System security officer
Configuration group	Security Related

enable row level access control enables row level access control. You must have the security services license key enabled before you can configure enable row level access control.

enable semantic partitioning

Summary information	
Default value	0
Range of values	1 (enabled), 0 (disabled)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

enable semantic partitioning enables partitioning other than round-robin (for example list, hash, and range partitioning) in Adaptive Server. Before you use any of these partitioning schemes, you must first have the appropriate license.

enable sort-merge join and JTC

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

Used when Adaptive Server is in compatibility mode. Once enabled, when Adaptive Server compiles a query in compatibility mode, enable sort-merge join and JTC allows the query processor to select a sort merge or a nested loop join. enable sort-merge join and JTC enables join transitive closure (JTC), which allows the query processor for releases earlier than 15.0 to use additional join clauses.

For more information about compatibility mode, see the *Migration Technology Guide*.

enable sql debugger

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

Enables and disables the Adaptive Server SQL debugger, which allows you to step through your T-SQL code.

enable ssl

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System security officer
Configuration group	Security Related

enable ssl enables or disables Secure Sockets Layer session-based security.

enable stmt cache monitoring

Summary information	
Default value	0 (off)

Summary information	
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administration
Configuration group	Monitoring

enable stmt cache monitoring enables or disables Adaptive Server to collect monitoring information about the statement cache. Once enabled, monStatementCache and monCachedStatement display valid data.

enable surrogate processing

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Unicode

Activates the processing and maintains the integrity of surrogate pairs in Unicode data. If enable surrogate processing is disabled, the server ignores the presence of surrogate pairs in the Unicode data, and all code that maintains the integrity of surrogate pairs is skipped. This enhances performance, but restricts the range of Unicode characters that can appear in the data.

enable unicode conversion

Summary information	
Default value	1
Range of values	0 – uses only the built-in character-set conversion.
	1 – uses the built-in conversion. If it cannot find a built-in conversion, Adaptive Server uses the Unilib character conversion
	2 – uses the appropriate Unilib conversion
Status	Dynamic
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration groups	Languages, Unicode

enable unicode conversion activates character conversion using Unilib for the char, varchar, and text datatypes.

enable unicode normalization

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Unicode

Activates Unilib character normalization. The normalization process modifies the data so there is only a single representation in the database for a given sequence of abstract characters. Often, characters followed by combined diacritics are replaced by precombined forms.

Set enable unicode normalization to 1 to use the built-in process that enforces normalization on all incoming Unicode data. If this parameter is disabled (set to 0), the normalization step is bypassed and the client code is responsible for normalization rather than the server. If normalization is disabled, performance is improved—but only if *all* clients present Unicode data to the server using the same representation.

Note Once disabled, you cannot turn normalization on again. This limitation prevents non-normalized data from entering the database.

enable webservices

Summary information	
Default value	0
Range of values	1 (enabled), 0 (disabled)
Status	Dynamic

Summary information	
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

Enables Webservices.

enable xact coordination

Summary information	
Default value	1 (on)
Valid values	0 (off), 1(on)
Status	Static
Display level	10
Required role	System administrator
Configuration group	DTM Administration

enable xact coordination enables or disables Adaptive Server transaction coordination services. When this parameter is set to 1 (on), coordination services are enabled, and the server can propagate transactions to other Adaptive Servers. This may occur when a transaction executes a remote procedure call (RPC) to update data in another server, or updates data in another server using Component Integration Services (CIS). Transaction coordination services ensure that updates to remote Adaptive Server data commit or roll back with the original transaction.

If this parameter is set to 0 (off), Adaptive Server does not coordinate the work of remote servers. Transactions can still execute RPCs and update data using CIS, but Adaptive Server cannot ensure that remote transactions are rolled back with the original transaction or that remote work is committed along with an original transaction, if remote servers experience a system failure. This corresponds to the behavior of Adaptive Server versions earlier than version 12.x.

enable xml

Summary information	
Default value	0
Range of values	1 (enabled), 0 (disabled)
Status	Dynamic

Summary information	
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

Enables XML services.

engine memory log size

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	
Required role	
Configuration group	Physical Memory

engine memory log size is for diagnostic use only and has no relevance in a production environment. It should be left at the default setting unless otherwise requested by Sybase Tech Support.

errorlog pipe active

Summary information	
Default value	0
Range of values	0–1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

errorlog pipe active controls whether Adaptive Server collects error log messages. If both errorlog pipe active and errorlog pipe max messages are enabled, Adaptive Server collects all the messages sent to the error log. Use monErrorLog to retrieve these error log messages.

errorlog pipe max messages

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

errorlog pipe max messages determines the number of error log messages Adaptive Server stores per engine. The total number of messages in the monSQLText table is the value of sql text pipe max messages times the number of engines running.

esp execution priority

Summary information	
Default value	8
Range of values	0–15
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Extended Stored Procedure

esp execution priority sets the priority of the XP Server thread for ESP execution. Over long periods of time ESPs can be CPU-intensive. Also, since XP Server resides on the same machine as Adaptive Server, XP Server can impact Adaptive Server performance.

See the *Open Server Server-Library/C Reference Manual* for information about scheduling Open Server threads.

esp execution stacksize

Summary information	
Default value	Platform dependent:
	• 196608 – Linux AMD-64, IBM PLinux, HP IA64, Sun x86 64
	• 139264 – SunSparc64
	• 65536 – Sun Sparc32, Windows 32, IBM AIX 64, Linux IA 32, Windows 64
	• 67584 – HP 64
Range of values	Platform default—2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Extended Stored Procedure

esp execution stacksize sets the size of the stack, in bytes, to be allocated for ESP execution.

Use this parameter if you have your own ESP functions that require a larger stack size than the default, 34816.

esp unload dll

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Extended Stored Procedure

esp unload dll specifies whether DLLs that support ESPs should be automatically unloaded from XP Server memory after the ESP call has completed.

If esp unload dll is set to 0, DLLs are not automatically unloaded. If it is set to 1, they are automatically unloaded.

If esp unload dll is set to 0, you can still unload individual DLLs explicitly at runtime, using sp_freedll.

event buffers per engine

Summary information	
Default value	100
Range of values	1 – 2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

event buffers per engine specifies the number of event buffers available for each engine.

Events are used by Adaptive Server Monitor for observing Adaptive Server performance; if you are not using Adaptive Server Monitor, set this parameter to 1.

The value to which you set event buffers per engine depends on the number of engines in your configuration, the level of activity on your Adaptive Server, and the types of applications you are running.

Setting event buffers per engine to a low value may result in the loss of event information. The default value is likely to be too low for most sites. Values of 2000 and greater may be more reasonable for general monitoring. However, experiment to determine the appropriate value for your site.

In general, setting event buffers per engine to a high value may reduce the amount of performance degradation that Adaptive Server Monitor causes Adaptive Server.

Each event buffer uses 100 bytes of memory. To determine the total amount of memory used by a particular value for event buffers per engine, multiply the value by the number of Adaptive Server engines in your configuration.

event log computer name **(Windows only)**

Summary information	
Default value	LocalSystem
Valid values	• Name of an Windows machine on the network configured to record Adaptive Server messages
	• LocalSystem
	• NULL
Status	Dynamic

Summary information	
Display level	Comprehensive
Required role	System administrator
Configuration group	Error Log

event log computer name specifies the name of the Windows PC that logs Adaptive Server messages in its Windows Event Log. This feature is available on Windows servers only.

A value of LocalSystem or NULL specifies the default local system.

You can also use the Server Config utility to set the event log computer name parameter by specifying the Event Log Computer Name under Event Logging.

Setting the event log computer name parameter with sp_configure or specifying the Event Log Computer Name under Event Logging overwrites the effects of the command line -G option, if it was specified. If Adaptive Server was started with the -G option, you can change the destination remote machine by setting event log computer name.

For more information about logging Adaptive Server messages to a remote site, see the *Configuration Guide for Windows*.

event logging **(Windows only)**

Summary information	
Default value	1
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Error Log

event logging enables and disables the logging of Adaptive Server messages in the Windows Event Log.

The default value of 1 enables Adaptive Server message logging in the Windows Event Log; a value of 0 disables it.

Use the Server Config utility to set the event logging parameter by selecting Use Windows Event Logging under Event Logging.

Setting the event logging parameter or selecting Use Windows Event Logging overwrites the effects of the command line -G option, if it was specified.

executable codesize + overhead

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Calculated
Display level	Basic
Required role	System administrator
Configuration group	Memory Use

executable codesize + overhead reports the combined size, in kilobytes, of the Adaptive Server executable and overhead. This a calculated value that is not user-configurable.

extended cache size

Summary information	
Default value	0
Range of values	0 – 31457280
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	Cache Manager

extended cache size specifies the size of the secondary cache.

FIPS login password encryption

Summary information	
Default value	0
Range of values	0 – 1
Status	Static
Display level	Comprehensive
Required role	System security officer
Configuration group	Security related

Enabling FIPS login password encryption requires a Security and Directory Services license. This parameter provides FIPS 140-2 cryptographic module support for encrypting passwords in transmission, in memory, and on disk.

Adaptive Server uses the FIPS 140-2 certified Certicom security provider for login encryption. If this configuration is not enabled, Adaptive Server uses the OpenSSL security provider to perform login password encryption.

global async prefetch limit

Summary information	
Default value	10
Range of values	0–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Cache Manager

global async prefetch limit specifies the percentage of a buffer pool that can hold the pages brought in by asynchronous prefetch that have not yet been read. This parameter sets the limit for all pools in all caches for which the limit has not been set explicitly with sp_poolconfig.

If the limit for a pool is exceeded, asynchronous prefetch is temporarily disabled until the percentage of unread pages falls below the limit. See Chapter 6, "Tuning Asynchronous Prefetch" in the *Performance and Tuning Series: Basics*.

global cache partition number

Summary information	
Default value	1
Range of values	1 – 64, as powers of 2
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	Cache Manager

global cache partition number sets the default number of cache partitions for all data caches. The number of partitions for a particular cache can be set using sp_cacheconfig; the local value takes precedence over the global value.

Use cache partitioning to reduce cache spinlock contention; in general, if spinlock contention exceeds 10 percent, partitioning the cache should improve performance. Doubling the number of partitions cuts spinlock contention by about one-half.

See Chapter 4, "Configuring Data Caches," in the *System Administration Guide: Volume 2* for information on configuring cache partitions. Also see Chapter 6, "Tuning Asynchronous Prefetch" in the *Performance and Tuning Series: Basics*.

heap memory per user

Summary information	
Default value	4K
Valid values	0 – 2147483647 bytes
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Physical Memory

heap memory per user configures the amount of heap memory per user. A heap memory pool is an internal memory created at start-up that tasks use to dynamically allocate memory as needed. This memory pool is important if you are running tasks that use wide columns, which require a lot of memory from the stack. The heap memory allocates a temporary buffer that enables these wide column tasks to finish. The heap memory the task uses is returned to the heap memory pool when the task is finished.

The size of the memory pool depends on the number of user connections. Sybase recommends that you set heap memory per user to three times the size of your logical page.

histogram tuning factor

Summary information	
Default value	20
Range of values	1 – 100
Status	Dynamic
Display level	Intermediate
Required role	System administrator

Summary information	
Configuration group	SQL Server Administration

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. A value of 1 disables the parameter.

Note For Adaptive Server versions 15.0.2 ESD #2 and later, if you set histogram tuning factor to the default value of 20 and a large number of steps are requested for the histogram, the actual step count used for the histogram is limited to the value that reduces the procedure cache usage:

```
min (max (400, requested_steps),
histogram_tuning_factor X requested_steps)
```

In the following example, Adaptive Server generates an intermediate 20-step histogram with 30 values:

```
sp_configure 'histogram tuning factor',20
update statistics tab using 30 values
```

Adaptive Server analyzes the histogram and compresses it into the resulting histogram according to the following parameters:

• The first step is copied unchanged.

• The high-frequency steps are copied unchanged.

• The consecutive range steps are collapsed into the resulting step, so the total weight of the collapsed step is no bigger than one-thirtieth of the value.

The final histogram in sysstatistics:

• Has range steps generated in a way similar for a 30-step update statistics, and high frequency ranges are isolated as if the histogram were created with 600 steps.

• The total number of steps in the resulting histogram may differ between 30 and 600 values.

• For equally distributed data, the value should be very close to 30.

• More "frequent" values in the table means more steps in the histogram.

• If a column has few different values, all those values may appear as high-frequency cells.

Adaptive Server Enterprise

You can achieve the same result by increasing the number of histogram steps to 600, but this uses more resources in the buffer and procedure cache

histogram tuning factor minimizes the resources histograms consume, and increases resource usage only when it is in the best interest for optimization, for example, when there is uneven distribution of data in a column, or highly duplicated values within a column. In this situation, up to 600 histogram steps are used. However, in most cases, histogram tuning factor uses the default value (30 in the example above).

housekeeper free write percent

Summary information	
Default value	1
Range of values	0–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

housekeeper free write percent specifies the maximum percentage by which the housekeeper wash task can increase database writes.

For example, to stop the housekeeper task from working when the frequency of database writes reaches 5 percent above normal, set housekeeper free write percent to 5.

When Adaptive Server has no user tasks to process, the housekeeper wash task automatically begins writing changed pages from cache to disk. These writes result in improved CPU utilization, decreased need for buffer washing during transaction processing, and shorter checkpoints.

In applications that repeatedly update the same database page, the housekeeper wash may initiate some unnecessary database writes. Although these writes occur only during the server's idle cycles, they may be unacceptable on systems with overloaded disks.

The table and index statistics that are used to optimize queries are maintained in memory structures during query processing. When these statistics change, the changes are not written to the systabstats table immediately, to reduce I/O contention and improve performance. Instead, the housekeeper chores task periodically flushes statistics to disk.

The default value allows the housekeeper wash task to increase disk I/O by a maximum of 1 percent. This results in improved performance and recovery speed on most systems.

To disable the housekeeper wash task, set the value of housekeeper free write percent to 0.

Set this value to 0 only if disk contention on your system is high, and it cannot tolerate the extra I/O generated by the housekeeper wash task.

If you disable the housekeeper tasks, keep statistics current. Commands that write statistics to disk are:

- update statistics

- dbcc checkdb (for all tables in a database) or dbcc checktable (for a single table)

- sp_flushstats

Run one of these commands on any tables that have been updated since the last time statistics were written to disk, at the following times:

- Before dumping a database

- Before an orderly shutdown

- After restarting, following a failure or orderly shutdown; in these cases, you cannot use sp_flushstats—you must use update statistics or dbcc commands

- After any significant changes to a table, such as a large bulk copy operation, altering the locking scheme, deleting or inserting large numbers of rows, or performing a truncate table command

To allow the housekeeper wash task to work continuously, regardless of the percentage of additional database writes, set housekeeper free write percent to 100.

Use sp_sysmon to monitor housekeeper performance. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

You might also want to look at the number of free checkpoints initiated by the housekeeper task. The *Performance and Tuning Series: Basics* describes this output.

i/o accounting flush interval

Summary information	
Default value	1000
Range of values	1–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

i/o accounting flush interval specifies the amount of time, in machine clock ticks, that Adaptive Server waits before flushing I/O statistics for each user from sysprocesses to syslogins. This is used for charge-back accounting.

When a user logs in to Adaptive Server, the server begins accumulating I/O statistics for that user process in sysprocesses. When the value of i/o accounting statistics interval is exceeded, or a user logs off Adaptive Server, the accumulated I/O statistics for that user are flushed from sysprocesses to syslogins. These statistics continue accumulating in syslogins until you clear the totals by using sp_clearstats. You can display the current totals from syslogins by using sp_reportstats.

The value to which you set i/o accounting flush interval depends on the type of reporting you intend to do. If you run reports on a monthly basis, set i/o accounting flush interval to a relatively high value. With infrequent reporting, it is less critical that the data in syslogins be updated frequently.

If you perform periodic ad hoc selects on the totio column syslogins to determine I/O volume by process, set i/o accounting flush interval to a lower value. Doing so increases the likelihood of the data in syslogins being current when you execute your selects.

If you do not report on I/O statistics at all, set i/o accounting flush interval to its maximum value. This reduces the number of times syslogins is updated and the number of times its pages must be written to disk.

i/o batch size

Summary information	
Default value	100
Range of values	1–2147483647
Status	Dynamic

Summary information	
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

i/o batch size sets the number of writes issued in a batch before the task goes to sleep. Once this batch is completed, the task is woken up, and the next batch of writes are issued, ensuring that the I/O subsystem is not flooded with many simultaneous writes. Setting i/o batch size to the appropriate value can improve the performance of operations like checkpoint, dump database, select into, and so on.

i/o polling process count

Summary information	
Default value	10
Range of values	1 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

Note i/o polling process count functions only when you configure Adaptive Server for process kernel mode; it is nonfunctional for threaded kernel mode.

i/o polling process count specifies the maximum number of processes that Adaptive Server can run before the scheduler checks for disk and network I/O completions. Tuning i/o polling process count affects both the response time and throughput of Adaptive Server.

Adaptive Server checks for disk or network I/O completions:

- If the number of tasks run since the last time Adaptive Server checked for I/O completions equals the value for i/o polling process count, and

- At every Adaptive Server clock tick.

As a general rule, increasing the value of i/o polling process count increases throughput for applications that generate a lot of disk and network I/O. Conversely, decreasing the value improves process response time in these applications, possibly at the risk of lowering throughput.

If your applications create both I/O and CPU-bound tasks, tuning i/o polling process count to a low value (1 – 2) ensures that I/O-bound tasks get access to CPU cycles.

For OLTP applications (or any I/O-bound application with user connections and short transactions), tuning i/o polling process count to a value in the range of 20 – 30 may increase throughput, but may also increase response time.

When tuning i/o polling process count, consider these other parameters:

- sql server closk tick length, which specifies the duration of the Adaptive Server clock tick in microseconds. See "sql server clock tick length" on page 256.

- time slice, which specifies the number of clock ticks the the Adaptive Server scheduler allows a user process to run. See "time slice" on page 271.

- cpu grace time, which specifies the maximum amount of time, in clock ticks, a user process can run without yielding the CPU before Adaptive Server preempts it and terminates it with a timeslice error. See "cpu grace time" on page 104.

Use sp_sysmon to determine the effect of changing i/o polling process count. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

identity burning set factor

Summary information	
Default value	5000
Range of values	1–9999999
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

IDENTITY columns are of type numeric and scale zero whose values are generated by Adaptive Server. Column values can range from a low of 1 to a high determined by the column precision.

For each table with an IDENTITY column, Adaptive Server divides the set of possible column values into blocks of consecutive numbers, and makes one block at a time available in memory. Each time you insert a row into a table, Adaptive Server assigns the IDENTITY column the next available value from the block. When all the numbers in a block have been used, the next block becomes available.

This method of choosing IDENTITY column values improves server performance. When Adaptive Server assigns a new column value, it reads the current maximum value from memory and adds 1. Disk access becomes necessary only after all values within the block have been used. Because all remaining numbers in a block are discarded in the event of server failure (or shutdown with nowait), this method can lead to gaps in IDENTITY column values.

Use identity burning set factor to change the percentage of potential column values that is made available in each block. This number should be high enough for good performance, but not so high that gaps in column values are unacceptably large. The default value, 5000, releases .05 percent of the potential IDENTITY column values for use at one time.

To get the correct value for sp_configure, express the percentage in decimal form, and then multiply it by 10^7 (10,000,000). For example, to release 15 percent (.15) of the potential IDENTITY column values at a time, specify a value of .15 times 10^7 (or 1,500,000) in sp_configure.

identity grab size

Summary information	
Default value	1
Range of values	1–2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

identity grab size allows each Adaptive Server process to reserve a block of IDENTITY column values for inserts into tables that have an IDENTITY column.

This is useful if you are performing inserts, and you want all the inserted data to have contiguous IDENTITY numbers. For instance, if you are entering payroll data, and you want all records associated with a particular department to be located within the same block of rows, set identity grab size to the number of records for that department.

identity grab size applies to all users on Adaptive Server. Large identity grab size values result in large gaps in the IDENTITY column when many users insert data into tables with IDENTITY columns.

Sybase recommends that you set identity grab size to a value large enough to accommodate the largest group of records you want to insert into contiguous rows.

identity reservation size

Summary information	
Default value	1
Range of values	1–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

identity reservation size sets a limit for the number of identity values.

idle migration timeout

Summary information	
Default value	60
Valid values	0 – 32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

idle migration timeout specifies the amount of time after which an idle connection is closed without invalidating the migration request sent to the client, allowing you to stop an instance after a specified period of time without waiting for idle client connections to migrate.

Setting idle migration timeout to a high value slows down a graceful shutdown because the instance must wait the specified period of time for all idle connections that issued a migration request without the client having initiated migration.

job scheduler interval

Summary information	
Default value	1 (in minutes)
Range of values	1 – 600
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

job scheduler interval sets the interval when the Job Scheduler checks which scheduled jobs are due to be executed.

job scheduler tasks

Summary information	
Default value	4
Range of values	1 – 640
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

job scheduler tasks sets the maximum number of jobs that can run simultaneously through Job Scheduler.

Note If you increase the value of job scheduler tasks to a higher value, you must increase the value for number of user connections by at least twice the value you incremented job scheduler tasks before starting the Job Scheduler.

However, if the Adaptive Server running the scheduled jobs is the same Adaptive Server that is hosting the Job Scheduler, you must increase the value for number of user connections by three times the value you incremented job scheduler tasks before starting the Job Scheduler.

Increasing the number of user connections may require that you increase the value for max memory.

---

js job output width

Summary information	
Default value	80
Range of values	1 – 32768
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

js job output width determines the line width the output uses for jobs stored in the js_output table.

kernel mode

Summary information	
Default value	threaded
Range of values	threaded or process
Status	Static
Display level	Basic
Required role	System administrator
Configuration group	SQL Server Administration

kernel mode determines the mode the Adaptive Server kernel uses, threaded or process. In threaded mode, Adaptive Server uses operating system threads to support Adaptive Server engines and specialized tasks. In this mode, Adaptive Server is a single process running on the operating system. In process mode, Adaptive Server uses individual processes to support Adaptive Server engines. In this mode, each engine is a distinct process running on the operating system. The process mode is the same kernel used by Adaptive Server versions earlier than 15.7.

Note Adaptive Server supports only threaded kernel mode on the Windows platform.

The values for kernel mode are character data, so you must use 0 as a placeholder for the second sp_configure parameter, which must be numeric, and specify threaded or process as the third parameter. See "Unit specification using sp_configure" on page 64.

Sybase assumes you use threaded mode in your production server, and includes process mode for backward compatibilty. Process mode may not support features for Adaptive Server 15.7 and later.

See Chapter 5, "Managing Multiprocessor Servers," in the *System Administration Guide: Volume 2*.

kernel resource memory

Summary information	
Default value	• Solaris – 6184
	• All other platforms – 4096
Range of values	75 – 2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Phsyical Memory

kernel resource memory determines the size, in 2K pages, of the kernel resource memory pool from which all thread pools and other kernel resources are allocated memory. The amount of memory available for kernel resource memory depends on the value of max memory. If max memory has an insufficient amount of memory to allocate to kernel resource memory, Adaptive Server issues an error message that includes the value to which you must raise max memory.

The amount of kernel resource memory required is determined mostly by a combination of max online engines and number of user connections. For configurations of 8 engines or fewer, Sybase recommends that you add one page of kernel resource memory for every two user connections above 100. For configurations of 9 engines or more, add 1 page of kernel resource memory for every user connection.

For example, a server with 6 engines and 200 user connections should add a value of 100 to the default value. A server with 16 engines and 10000 user connections would add a value of 10000 to the default value.

license information

Summary information	
Default value	25
Valid values	$0–2^{31}$
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

license information allows Sybase system administrators to monitor the number of user licenses used in Adaptive Server. Enabling this parameter only monitors the number of licenses issued; it does not enforce the license agreement.

If license information is set to 0, Adaptive Server does not monitor license use. If license information is set to a number greater than 0, the housekeeper chores task monitors the number of licenses used during the idle cycles in Adaptive Server. Set license information to the number of licenses specified in your license agreement.

If the number of licenses used is greater than the number to which license information is set, Adaptive Server writes this message to the error log:

```
WARNING: Exceeded configured number of user licenses
```

At the end of each 24-hour period, the maximum number of licenses used during that time is added to the syblicenseslog table. The 24-hour period restarts if Adaptive Server is restarted.

See Chapter 3, "Managing Adaptive Server Logins and Database Users," in the *Security Administration Guide*.

lock address spinlock ratio

Summary information	
Default value	100
Range of values	1–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

For Adaptive Servers running with multiple engines, the address lock spinlock ratio sets the number of rows in the internal address locks hash table that are protected by one spinlock.

Adaptive Server manages the acquiring and releasing of address locks using an internal hash table with 1031 rows (known as hash buckets). This table can use one or more spinlocks to serialize access between processes running on different engines.

The default value for address lock spinlock ratio defines 11 spinlocks for the address locks hash table. The first 10 spinlocks protect 100 rows each, and the eleventh spinlock protects the remaining 31 rows. If you specify a value of 1031 or greater for address lock spinlock ratio, Adaptive Server uses only 1 spinlock for the entire table.

lock hashtable size

Summary information	
Default value	2048
Range of values	1–2147483647
Status	Static
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration groups	Lock Manager, Memory Use

lock hashtable size specifies the number of hash buckets in the lock hash table. This table manages all row, page, and table locks, and all lock requests. Each time a task acquires a lock, the lock is assigned to a hash bucket, and each lock request for that lock checks the same hash bucket. Setting this value too low results in large numbers of locks in each hash bucket and slows the searches. On Adaptive Servers with multiple engines, setting this value too low can also lead to increased spinlock contention. Do not set the value to less than the default value, 2048.

lock hashtable size must be a power of 2. If the value you specify is not a power of 2, sp_configure rounds the value to the next highest power of 2 and prints an informational message.

The optimal hash table size is a function of the number of distinct objects (pages, tables, and rows) that can be locked concurrently. The optimal hash table size is at least 20 percent of the number of distinct objects that need to be locked concurrently. See the *Performance and Tuning Series: Locking and Concurrency Control*.

However, if you have a large number of users and have had to increase the number of locks parameter to avoid running out of locks, use sp_sysmon to check the average hash chain length at peak periods. If the average length of the hash chains exceeds 4 or 5, consider increasing the value of lock hashtable size from its current setting to the next power of 2.

The hash chain length may be high during large insert batches, such as bulk copy operations. This is expected behavior, and does not require you to reset lock hashtable size.

lock scheme

Summary information	
Default value	allpages
Range of values	allpages, datapages, datarows
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

lock scheme sets the default locking scheme to be used by create table and select into commands when a lock scheme is not specified in the command.

The values for lock scheme are character data, so you must use 0 as a placeholder for the second parameter, which must be numeric, and specify allpages, datapages, or datarows as the third parameter:

```
sp_configure "lock scheme", 0, datapages
```

lock shared memory

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Physical Memory

lock shared memory disallows swapping of Adaptive Server pages to disk and allows the operating system kernel to avoid the server's internal page locking code. This can reduce disk reads, which are expensive.

Not all platforms support shared memory locking. Even if your platform does, lock shared memory may fail due to incorrectly set permissions, insufficient physical memory, or for other reasons. See operating system documentation for your platform for information on shared memory locking.

lock spinlock ratio

Summary information	
Default value	85
Range of values	1–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Lock Manager, Memory Use

Adaptive Server manages the acquiring and releasing of locks using an internal hash table with a configurable number of hash buckets. On SMP systems, this hash table can use one or more spinlocks to serialize access between processes running on different engines. To set the number of hash buckets, use lock hashtable size.

For Adaptive Servers running with multiple engines, lock spinlock ratio sets a ratio that determines the number of lock hash buckets that are protected by one spinlock. If you increase lock hashtable size, the number of spinlocks increases, so the number of hash buckets protected by one spinlock remains the same.

The Adaptive Server default value for lock spinlock ratio is 85. With lock hashtable size set to the default value of 2048, the default spinlock ratio defines 26 spinlocks for the lock hash table. See Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2.

sp_sysmon reports on the average length of the hash chains in the lock hash table. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

lock table spinlock ratio

Summary information	
Default value	20
Range of values	1–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

For Adaptive Servers running with multiple engines, table lock spinlock ratio sets the number of rows in the internal table locks hash table that are protected by one spinlock.

Adaptive Server manages the acquiring and releasing of table locks using an internal hash table with 101 rows (known as hash buckets). This table can use one or more spinlocks to serialize access between processes running on different engines.

The Adaptive Server default value for table lock spinlock ratio is 20, which defines 6 spinlocks for the table locks hash table. The first 5 spinlocks protect 20 rows each; the sixth spinlock protects the last row. If you specify a value of 101 or greater for table lock spinlock ratio, Adaptive Server uses only 1 spinlock for the entire table.

lock timeout pipe active

Summary information	
Default value	1 (off)
Range of values	0 (on), 1 (off)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

lock timeout pipe active controls whether Adaptive Server collects lock timeout messages. If lock timeout pipe active and lock timeout pipe max messages are enabled, Adaptive Server collects the data for each lock timeout that occurs.

Retrieve the lock timeout messages from the monLockTimeout monitor table.

lock timeout pipe max messages

Summary information	
Default value	0
Range of values	0 - 2147483648
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

lock timeout pipe max messages controls the maximum number of rows per engine in the lock timeout pipe, which determines the maximum number of rows that can be returned by the monLockTimeout monitoring table.

lock wait period

Summary information	
Default value	2147483647
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

lock wait period limits the number of seconds that tasks wait to acquire a lock on a table, data page, or data row. If the task does not acquire the lock within the specified time period, Adaptive Server returns error message 12205 to the user and rolls back the transaction.

The lock wait option of the set command sets a session-level number of seconds that a task waits for a lock. It overrides the server-level setting for the session.

lock wait period, used with the session-level setting set lock wait nnn, is applicable only to user-defined tables. These settings have no influence on system tables.

At the default value, all processes wait indefinitely for locks. To restore the default value, reset the value to 2147483647 or enter:

```
sp_configure "lock wait period", 0, "default"
```

log audit logon failure

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Error Log

log audit logon failure specifies whether to log unsuccessful Adaptive Server logins to the Adaptive Server error log and, on Windows servers, to the Windows Event Log, if event logging is enabled.

log audit logon success

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Error Log

log audit logon success specifies whether to log successful Adaptive Server logins to the Adaptive Server error log and, on Windows servers, to the Windows Event Log, if event logging is enabled.

max async i/os per engine

Summary information	
Default value	Platform dependent
Range of values	1– platform-dependent value
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	O/S Resources

max async i/os per engine specifies the maximum number of outstanding asynchronous disk I/O requests for a single engine at one time.

On the Linux platform

On the Linux platform, max async i/os per engine controls the number of asynchronous IOs reserved from the operating system.

In threaded mode, Adaptive Server reserves the value for max async i/os per engine for the engine server. In process mode, Adaptive Server reserves the value for max async i/os per engine for each engine. For example, if you set max async i/os per engine to 4096 and there are 4 engines, in threaded mode Adaptive Server reserves 4096 I/Os, but in process mode Adaptive Server reserves 16384 I/Os.

Your system may benefit from using a number greater than the default value.

You can use sp_sysmon to help tune max async i/os per engine. sp_sysmon's `disk i/o section` contains information about the maximum number of outstanding IOs for each engine during the sample period and the number of I/Os that were delayed because of engine or operating system limits. Generally, any I/Os delayed by engine limits indicate that you should increase the value of max async i/os per engine.

Whether Adaptive Server can perform asynchronous IO on a device depends on whether or not this device support kernel asynchronous I/O (KAIO). The Linux kernel requires that you implement kernel asynchronous I/O support at the file system level. Most major file systems provide support for kernel asynchronous I/O, including ext3, xfs, jfs, and raw devices. The tmpfs file system does not support kernel asynchronous I/O. If the device does not support kernel asynchronous I/O, Adaptive Server cannot perform asyncronous IO on that device, and instead reverts to standard synchronous IO for all reads and writes to that device. Adaptive Server prints a message similar to the following in the error log indicating that the device has switched to synchronous IO:

```
00:00000:00001:2006/12/15 11:47:17.98 kernel  Virtual device
'/dev/shm/tempdb.dat' does not support kernel asynchronous i/o. Synchronous i/o
will be used for this device.
```

max async i/os per server

Summary information	
Default value	Platform dependent
Range of values	1– platform dependent value
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	O/S Resources

max async i/os per server specifies the maximum number of asynchronous disk I/O requests that can be outstanding for Adaptive Server at one time. This limit is not affected by the number of online engines per Adaptive Server. max async i/os per engine limits the number of outstanding I/Os per engine.

Most operating systems limit the number of asynchronous disk I/Os that can be processed at any one time; some operating systems limit the number per operating system process, some limit the number per system, and some do both. If an application exceeds these limits, the operating system returns an error message. Because operating system calls are relatively expensive, it is inefficient for Adaptive Server to attempt to perform asynchronous I/Os that get rejected by the operating system.

To avoid this, Adaptive Server maintains a count of the outstanding asynchronous I/Os per engine and per server; if an engine issues an asynchronous I/O that would exceed either max async i/os per engine or max async i/os per server, Adaptive Server delays the I/O until enough outstanding I/Os have completed to fall below the exceeded limit.

For example, assume an operating system limit of 200 asynchronous I/Os per system and 75 per process and an Adaptive Server with three online engines. The engines currently have a total of 200 asynchronous I/Os pending, distributed according to the following table:

Engine	Number of I/Os pending	Outcome
0	60	Engine 0 delays any further asynchronous I/Os until the total for the server is under the operating system per-system limit and then continues issuing asynchronous I/Os.
1	75	Engine 1 delays any further asynchronous I/Os until the per-engine total is under the operating system per-process limit and then continues issuing asynchronous I/Os.
2	65	Engine 2 delays any further asynchronous I/Os until the total for server is under the operating system per-system limit and then continues issuing asynchronous I/Os.

All I/Os (both asynchronous and synchronous) require a disk I/O structure, so the total number of outstanding disk I/Os is limited by the value of disk i/o structures. It is slightly more efficient for Adaptive Server to delay the I/O because it cannot get a disk I/O structure than because the I/O request exceeds max i/os per server. Set max async i/os per server equal to the value of disk i/o structures. See "disk i/o structures" on page 116.

If the limits for asynchronous I/O can be tuned on your operating system, make sure they are set high enough for Adaptive Server. There is no penalty for setting them as high as needed.

Use sp_sysmon to see if the per server or per engine limits are delaying I/O on your system. If sp_sysmon shows that Adaptive Server exceeded the limit for outstanding requests per engine or per server, raise the value of the corresponding parameter. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

max cis remote connections

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Component Integration Services

max cis remote connections specifies the maximum number of concurrent Client-Library connections that can be made to remote servers by Component Integration Services.

By default, Component Integration Services allows up to four connections per user to be made simultaneously to remote servers. If you set the maximum number of users to 25, as many as 100 simultaneous Client-Library connections are allowed by Component Integration Services.

If this number does not meet the needs of your installation, you can override the setting by specifying exactly how many outgoing Client-Library connections you want the server to be able to make at one time.

max concurrently recovered db

Summary information	
Default value	0
Valid values	1– number of engines at start-up minus 1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Backup/Recovery

max concurrently recovered db determines the degree of parallelism. The minimum value is 1, which uses serial recovery, but you can also use the default value of 0, to use a self-tuning approach. The maximum value is the number of engines at start-up minus 1. max concurrently recovered db is also limited by the value of the configuration parameter number of open databases.

max memory

Summary information	
Default value	Platform-dependent
Range of values	Platform-dependent minimum – 2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Physical Memory

max memory specifies the maximum amount of total physical memory that you can configure Adaptive Server to allocate. max memory must be greater than the total logical memory consumed by the current configuration of Adaptive Server.

There is no performance penalty for configuring Adaptive Server to use the maximum memory available to it on your computer. However, assess the other memory needs on your system, or Adaptive Server may not be able to acquire enough memory to start.

See see Chapter 3, "Configuring Memory," in System Administration Guide: Volume 2.

If Adaptive Server cannot start

When allocate max shared memory is set to 1, Adaptive Server must have the amount of memory available that is specified by max memory. If the memory is not available, Adaptive Server does not start. If this occurs, reduce the memory requirements for Adaptive Server by manually changing the value of max memory in the server's configuration file. You can also change the value of allocate max shared memory to 0 so that not all memory required by max memory is required at start-up.

You may also want to reduce the values for other configuration parameters that require large amounts of memory. Then restart Adaptive Server to use the memory specified by the new values. If Adaptive Server fails to start because the total of other configuration parameter values is higher than the max memory value, see Chapter 3, "Configuring Memory," in System Administration Guide: Volume 2 for information about configuration parameters that use memory.

max native threads per engine

Summary information	
Default value	50
Maximum values	50 – 1000
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	User Environment

Note max native threads per engine is ignored in threaded mode.

max native threads per engine defines 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.

max nesting level

In Adaptive Server 15.0.3 and later, the maximum nesting level has been increased to 100, and the default value to 50.

Summary information	
Default value	50
Range of values	16 – 100
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	User environment

maximum nesting level sets the maximum nesting level for stored procedures and triggers. Each increased nesting level requires about 160 bytes of additional memory. For example, if you increase the nesting level from 16 to 26, Adaptive Server requires an additional 1600 bytes of memory.

max network packet size

Summary information	
Default value	512

Summary information	
Range of values	512–65024
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	Network Communication

max network packet size specifies the maximum network packet size that can be requested by clients communicating with Adaptive Server.

If some of your applications send or receive large amounts of data across the network, these applications can achieve significant performance improvement by using larger packet sizes. Two examples are large bulk copy operations and applications that read or write large text, unitext, and image values.

Generally, you want:

- default network packet size to be small for users who perform short queries, and

- max network packet size to be large enough to allow users who send or receive large volumes of data to request larger packet sizes.

max network packet size must always be as large as, or larger than, the default network packet size. Values that are not even multiples of 512 are rounded down.

For client applications that explicitly request a larger network packet size to receive, you must also configure additional network memory. See "additional network memory" on page 82.

Open Client Server cannot accept a network packet size greater than 64K.

See bcp and isql in the *Utility Guide* for information on using larger packet sizes from these programs. Open Client Client-Library documentation includes information on using variable packet sizes.

Choosing packet sizes

For best performance, choose a server packet size that works efficiently with the underlying packet size on your network. The goals are:

- Reducing the number of server reads and writes to the network

- Reducing unused space in network packets (increasing network throughput)

For example, if your network packet size carries 1500 bytes of data, setting the Adaptive Server packet size to 1024 (512*2) will probably achieve better performance than setting it to 1536 (512*3). Figure 5-3 shows how four different packet size configurations would perform in such a scenario.

Figure 5-3: Factors in determining packet size

Underlying network packets: 1500 bytes after overhead

Packet size 512
Used   1024 bytes
Unused   476 bytes
% Used: 68%
2 server reads

Depending on amount of data, network packets may have 1 or 2 packets

Packet size 1024
Used   1024 bytes
Unused   476 bytes
% Used: 68%
1 server read

Should yield improved performance over default of 512

Packet size 2560
Used   2560 bytes
Unused   440 bytes
% Used   85%
2 server reads
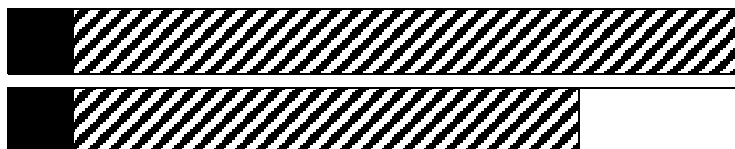
Possibly the best option of illustrated choices

Packet size 1536
Used   1536 bytes
Unused   1464 bytes
% Used   51%
2 server reads
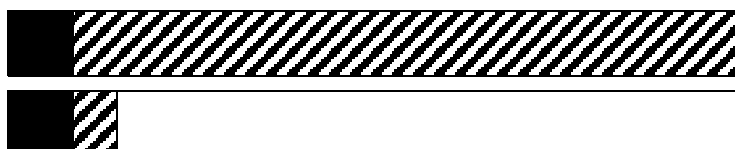
Probably the worst option of illustrated choices

Key:
Overhead   Data   Unused

After you determine the available data space of the underlying packets on your network, perform your own benchmark tests to determine the optimum size for your configuration.

Use sp_sysmon to see how changing max network packet size affects network
I/O management and task switching. For example, try increasing max network
packet size and then checking sp_sysmon output to see how this affects bcp for
large batches. See the *Performance and Tuning Series: Monitoring Adaptive
Server with sp_sysmon*.

max number network listeners

Summary information	
Default value	5
Range of values	0–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Network Communication

max number network listeners specifies the maximum number of network
listeners allowed by Adaptive Server at one time.

Each master port has one network listener. Generally, there is no need to have
multiple master ports, unless your Adaptive Server must communicate over
more than one network type. Some platforms support both socket and TLI
(Transport Layer Interface) network interfaces. See the *Configuration Guide*
for your platform for information on supported network types.

max online engines

Summary information	
Default value	1
Range of values	1 – 1024 (threaded mode)
	1 – 128 (process mode)
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Processors

max online engines places an upper limit of the number of engine threads that can be brought online. It does not take into account the number of CPUs available at start-up, and allows users to add CPUs at a later date. See Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2 for a detailed discussion of how to set this parameter for your SMP environment.

At start-up, Adaptive Server starts with a single engine thread and completes its initialization, including recovery of all databases. Its final task is to allocate additional server engines. Each engine accesses common data structures in shared memory.

When tuning the max engines online parameter:

*   Never have more engine threads online than there are CPUs.

*   max engines online must be large enough to allow Adaptive Server to bring all the engine thread pools online simultaneously.

*   Depending on overall system load (including applications other than Adaptive Server), you may achieve optimal throughput by leaving some CPUs free to run non-Adaptive Server processes.

*   You can achieve better throughput by running fewer engine threads with high CPU use, rather than by running more engine threads with low CPU use.

*   Scalability is application-dependent. Conduct extensive benchmarks on your application to determine the best configuration of online engine threads.

*   In process mode, use sp_engine to take engine threads offline or to bring them online. You can take all engine threads offline except engine zero.

See Chapter 3, "Using Engines and CPUs" in the *Performance and Tuning Series: Basics* .

max online Q engines

Summary information	
Default value	0
Range of values	0 – 127
Status	Static
Display level	Comprehensive
Required role	System administrator

Summary information	
Configuration group	Processors

(Process mode only) max online Q engines specifies the maximum number of Q engines you can have online, and is required for MQ (see the Active Messaging documentation).

The maximum number of Q engines depends on the value of max online engines:

- max online Q engines cannot be greater than max online engines minus number of engines at startup. That is, if the value for max online engines is 57, and the value for number of engines at startup is 28, the value for max online Q engines cannot be greater than 29.

- Setting max online Q engines reserves the high range of max online engines for Q engines. Once you set max online Q engines, Adaptive Server engines can no longer use the engines in the range that is reserved for Q engines. For example, if you set max online engines to 10 and set max online Q engines to 4:

  - Adaptive Server issues an error if you attempt to bind an engine group to engines 6, 7, 8, or 9

  - You cannot change the value of number of engines at startup to 7, 8, or 9

  - You cannot add engines 6, 7, 8, or 9 to an engine group

Because setting max online Q engines may affect existing production environments, Sybase recommends that you increase the value of max online engines by the same amount that you increase max online Q engines. For example, if you increase max online Q engines by 4, also increase max online engines by 4.

max parallel degree

Summary information	
Default value	1
Range of values	1–255
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Query Tuning

max parallel degree specifies the server-wide maximum number of worker processes allowed per query. This is called the "maximum degree of parallelism."

If max parallel degree is too low, the performance gain for a given query may not be as significant as possible; if max parallel degree is too high, the server may compile plans that require more processes than are actually available at execution time, or the system may become saturated, resulting in decreased throughput. To enable parallel partition scans, set this parameter to be equal to or greater than the number of partitions in the table you are querying.

The value of this parameter must be less than or equal to the current value of number of worker processes.

If you set max parallel degree to 1:

- Adaptive Server scans all tables or indexes serially.
- Adaptive Server forces serial query execution and the optimizer may select plans with a higher parallel degree than if it is disabled.

Changing max parallel degree causes all query plans in the procedure cache to be invalidated, and new plans are compiled the next time you execute a stored procedure or trigger.

See Chapter 9, "Parallel Sorting" in the *Performance and Tuning Series: Query Processing and Abstract Plans.*

max pci slots

Summary information	
Default value	0
Range of values	0 – 30
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	User Environment

Sets the maximum number of PCI slots Adaptive Server allows. The values are:

- 0, 1 – default bridge with one PCA.

> **Note** JVM support requires a single slot. Do not increase the number of slots.

- 2 – 30 – allocated for future releases.

For more information about PCI slots, see *Java in Adaptive Server Enterprise*.

max query parallel degree

Summary information	
Default value	1
Range of values	1 – 255
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

Used when Adaptive Server is in compatibility mode. Defines the number of worker processes to use for a given query. This parameter is relevant only if you do not want to enable parallelism globally. The value for number of worker process cannot be less than the value for max query parallel degree.

See Chapter 5, "Parallel Query Processing," in the *Performance and Tuning Series: Query Processing and Abstract Plans*.

For more information about compatibility mode, see the *Migration Technology Guide*.

max repartition degree

Summary information	
Default value	1
Range of values	1 – value of max parallel degree
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

max repartition degree configures the amount of dynamic repartitioning Adaptive Server requires, which enables Adaptive Server to use horizontal parallelism. However, if the number of partitions is too large, the system is flooded with worker processes that compete for resources, which degrades performance. The value for max repartition degree enforces the maximum number of partitions created for these resources. If all of the tables and indexes are unpartitioned, Adaptive Server uses the value for max repartition degree to provide the number of partitions to create as a result of repartitioning the data.

max resource granularity

Summary information	
Default value	10
Range of values	1 – 100
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

max resource granularity indicates the maximum percentage of the system's resources a query can use. This parameter is not enforced at execution time, but is only a guide for the query optimizer, and does not prevent the query processor from running queries in parallel. The query engine can avoid some memory-intensive strategies by using max resource granularity as a guide.

max scan parallel degree

Summary information	
Default value	1
Range of values	1–255
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Query Tuning

max scan parallel degree specifies the server-wide maximum degree of parallelism for hash-based scans which may be used for the following access methods:

• Parallel index scans for partitioned and nonpartitioned tables

• Parallel table scans for nonpartitioned tables

max scan parallel degree applies per table or index; that is, if max scan parallel degree is 3, and one table in a join query is scanned using a hash-based table scan and the second can best be accessed by a hash-based index scan, the query can use 9 worker processes (as long as max scan parallel degree is set to 9 or higher.

The optimizer uses max scan parallel degree as a guideline when it selects the number of processes to use for parallel, nonpartition-based scan operations. It does not apply to parallel sort. Because there is no partitioning to spread the data across devices, parallel processes can be accessing the same device during the scan. This can cause additional disk contention and head movement, which may degrade performance. To prevent multiple disk accesses from becoming a problem, use max scan parallel degree to reduce the maximum number of processes that can access the table in parallel.

If this number is too low, the performance gain for a given query is not as significant as possible; if the number is too large, the server may compile plans that use enough processes to make disk access less efficient. A general rule is to set this parameter to no more than 2 or 3, because it takes only 2 to 3 worker processes to fully utilize the I/O of a given physical device.

Set the value of max scan parallel degree to less than or equal to the current value of max parallel degree. Adaptive Server returns an error if you specify a number larger than the max parallel degree value.

If you set max scan parallel degree to 1, Adaptive Server does not perform hash-based scans.

Changing max scan parallel degree causes all query plans in the procedure cache to be invalidated, and new plans are compiled the next time you execute a stored procedure or trigger.

max SQL text monitored

Summary information	
Default value	0
Range of values	0–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

max SQL text monitored specifies the amount of memory allocated per user connection for saving SQL text to memory shared by Adaptive Server Monitor.

If you do not allocate enough memory for the batch statements, the text you want to view may be truncated. Sybase recommends that you use an initial value of 1024 bytes of memory per user connection.

The total memory allocated from shared memory for the SQL text is the product of max SQL text monitored multiplied by the currently configured number of user connections.

max transfer history

Summary information	
Default value	10
Range of values	1 – 255
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Adaptive Server Administration

max transfer history controls how many transfer history entries Adaptive Server retains in the spt_TableTransfer table in each database. For each table tracked, spt_TableTransfer retains:

(*N* successful entries) + (*N* unsuccessful entries)

Where *N* is the value for max transfer history.

Lowering this parameter does not automatically remove any entries from spt_TableTransfer. Entries are removed for a given transferred table the next time you initiate a transfer for that table. The table's successful transfer entries are cleared if the transfer succeeds. If the transfer is unsuccessful, its failed transfer entries are cleared.

For example, if a table has 12 successful and 9 unsuccessful history entries in spt_configure, and you change max transfer history to 5, the next successful transfer of that table places 5 successful entries in spt_configure, but spt_configure retains the previous 9 failed entries.

maximum dump conditions

Summary information	
Default value	10
Range of values	10–100
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	Group Diagnostics

maximum dump conditions sets the maximum number of conditions you can specify under which Adaptive Server generates a dump of data in shared memory.

Note This parameter is included for use only by Sybase Technical Support. Do not modify it unless you are instructed to do so by Sybase Technical Support.

max buffers per lava operator

Summary information	
Default value	2048
Range of values	500 – 65535
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

maximum buffers per lava opeator sets an upper limit for the number of buffers used by Lava operators that perform sorting or hashing (which are "expensive" in terms of processing). Lava operators use buffers from the session's tempdb data cache pool as a work area for processing rows.

Lava operators often recurse through their input streams. Sorting requires subsequent merge passes until there are enough buffers available to merge all of the remaining runs. Hashing requires subsequent passes to build hash tables on any spilled sets until all of the remaining data can fit into an in-memory hash table. Some queries require less I/O if you increase max buffers per lava operator. This is particularly true for queries that use the HASH DISTINCT, HASH VECTOR AGGREGATE, and HASH UNION operators.

Be careful when you increase the default value of maximum buffers per lava operator for servers with many concurrent users: Adaptive Server may allocate more buffers solely for expensive operators, reducing the number of buffers available for caching user's tables and other session's worktables. Use sp_sysmon to analyze tempdb's data caching effectiveness.

maximum buffers per lava operator works with max resource granularity to limit the number of buffers used. The limit is set to the minimum of:

• The value of maximum buffers per lava operator, or,

• (max resource granularity) X (the number of data buffers in tempdb's pagesize pool)

See "number of sort buffers" on page 216 for information about setting the amount of memory allocated for sort buffers.

maximum failed logins

Summary information	
Default value	0
Range of values	-1 – 32767
Status	Dynamic
Display level	10
Required role	System security officer
Configuration group	Security Related

maximum failed logins allows you to set the server-wide maximum number of failed login attempts for logins and roles.

A value of -1 indicates that the failed login count in the syslogins column logincount is updated whenever an authentication failure occurs, but that the account is not locked. Compare with a 0 (zero) value, which avoids incrementing the column for every failed authentication and avoids locking the account due to authentication failures.

See the *Reference Manual: Procedures* for information about using sp_modifylogin to change the maximum failed logins for a specific role. See the *Reference Manual: Commands* for information about using alter role to change the maxiumum failed logins.

maximum job output

Summary information	
Default value	32768
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

maximum job output sets limit, in bytes, on the maximum output a single job can produce. If a job produces more output than specified in maximum job output, all the data returned above the value you enter is discarded.

memory alignment boundary

Summary information	
Default value	Logical page size
Range of values	2048^a – 16384 a. Minimum determined by server's logical page size
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Cache Manager

memory alignment boundary determines the memory address boundary on which data caches are aligned.

Some machines perform I/O more efficiently when structures are aligned on a particular memory address boundary. To preserve this alignment, values for memory alignment boundary should always be powers of two between the logical page size and 2048K.

Note The memory alignment boundary parameter is included for support of certain hardware platforms. Do not modify it unless you are instructed to do so by Sybase Technical Support.

memory per worker process

Summary information	
Default value	1024
Range of values	1024–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Memory Use

memory per worker process specifies the amount of memory, in bytes, used by worker processes. Each worker process requires memory for messaging during query processing. This memory is allocated from a shared memory pool; the size of this pool is memory per worker process multiplied by number of worker processes. For most query processing, the default size is more than adequate. If you use dbcc checkstorage, and have set number of worker processes to 1, you may need to increase memory per worker process to 1792 bytes.

See Chapter 3, "Configuring Memory," in System Administration Guide: Volume 2.

messaging memory

Summary information	
Default value	400
Range of values	60 – 2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Memory Use, Physical Memory

Configures the amount of memory available for Sybase messaging.

metrics elap max

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic

Summary information	
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

metrics elap max configures maximum elapsed time and thresholds for QP metrics

metrics exec max

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

metrics exec max configures maximum execution time and thresholds for QP metrics.

metrics lio max

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

metrics lio max configures maximum logical I/O and thresholds for QP metrics.

metrics pio max

Summary information	
Default value	0
Range of values	0 – 2147483647

Summary information

Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	

metrics pio max configures maximum physical I/O and thresholds for QP metrics.

min pages for parallel scan

Summary information

Default value	200
Range of values	20 - 2147483647
Status	dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

min pages for parallel scan controls the number of tables and indexes that Adaptive Server can access in parallel. If the number of pages in a table is below the value you set, the table is accessed serially. min pages for parallel scan does not consider page size. If Adaptive Server accesses the indexes and tables, Adaptive Server attempts to repartition the data, if that is appropriate, and to use parallelism above the scans, if that is appropriate.

minimum password length

Summary information

Default value	6
Range of values	0 – 30
Status	Dynamic
Display level	10
Required role	System security officer
Configuration group	Security Related

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

mnc_full_index_filter

Summary information	
Default value	2
Range of values	0 – 2
	• 0 – disable.
	• 1 – enable.
	• 2 – set according to the optimization goal setting.
Status	Dynamic
Display level	Comprehensive
Required roles	System administrator
Configuration group	Query Tuning

mnc_full_index_filter prevents Adaptive Server from considering noncovered indexes that do not have a limiting search argument at the server level, if there is:

• A column in the index

• A predicate that does not have a histogram

You can use mnc_full_index_filter on data-only-locked (DOL) tables in which you have the intelligent index scan, even though the intelligent index scan manufactures search arguments.

Changing the value of mnc_full_index_filter does not increase the amount of memory Adaptive Server uses.

mnc_full_index_filter is not enabled for any specific optional goal; the only way to obtain the behavior is to explicitly enable it.

msg confidentiality reqd

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic

Summary information	
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

msg confidentiality reqd requires that all messages into and out of Adaptive Server be encrypted. The use security services parameter must be 1 for messages to be encrypted.

msg integrity reqd

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

msg integrity reqd requires that all messages be checked for data integrity. use security services must be 1 for message integrity checks to occur. If msg integrity reqd is set to 1, Adaptive Server allows the client connection to succeed unless the client is using one of the following security services: message integrity, replay detection, origin checks, or out-of-seq checks.

net password encryption required

Summary information	
Default value	0
Range of values	0 – 2
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

net password encryption reqd restricts login authentication to use only RSA encryption algorithm or the Sybase proprietary algorithm. Table 5-3 describes valid values for net password encryption reqd.

Table 5-3: Values and descriptions for net password encryption reqd

Value	Description
0	Allows the client to choose the encryption algorithm used for login passwords on the network, including no password encryption.
1	Restricts clients to use either RSA or Sybase proprietary encryption algorithms to encrypt login passwords on the network. This provides an incrementally restrictive setting that allows clients who have previously connect to reconnect with the Sybase proprietary algorithm and new clients to connect with the stronger RSA algorithm. A client that attempts to connect without using password encryption fails.
2	Restricts clients to use only the RSA encryption algorithms to encrypt login passwords on the network. This provides strong RSA encryption of passwords. Clients that attempt to connect without using the RSA encryption fail.

When a connection is refused because network password encryption is required, the client receives:

```
Msg 1640, Level 16, State 2:
Adaptive Server requires encryption of the login
password on the network.
```

number of alarms

Summary information	
Default value	40
Range of values	40 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

number of alarms specifies the number of alarm structures allocated by Adaptive Server.

The Transact-SQL command waitfor defines a specific time, time interval, or event for the execution of a statement block, stored procedure, or transaction. Adaptive Server uses alarms to correctly execute waitfor commands. Other internal processes require alarms.

When Adaptive Server needs more alarms than are currently allocated, this message is written to the error log:

```
uasetalarm: no more alarms available
```

The number of bytes of memory required for each alarm structure is small. If you raise the number of alarms value significantly, adjust max memory accordingly.

number of aux scan descriptors

Summary information	
Default value	200
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

number of aux scan descriptors sets the number of auxiliary scan descriptors available in a pool shared by all users on a server.

Each user connection and each worker process has 48 scan descriptors exclusively allocated to it. Of these, 16 are reserved for user tables, 12 are reserved for worktables, and 20 are reserved for system tables (with 4 of these set aside for rollback conditions). A descriptor is needed for each table referenced, directly or indirectly, by a query. For user tables, a table reference includes:

- All tables referenced in the from clause of the query

- All tables referenced in a view named in the query (the view itself is not counted)

- All tables referenced in a subquery

- All tables that need to be checked for referential integrity (these are used only for inserts, updates, and deletes)

- A table created with select...into

- All worktables created for the query

If a table is referenced more than once (for example, in a self-join, in more than one view, or in more than one subquery) the table is counted each time. If the query includes a union, each select statement in the union query is a separate scan. If a query runs in parallel, the coordinating process and each worker process needs a scan descriptor for each table reference.

When the number of user tables referenced by a query scan exceeds 16, or the number of worktables exceeds 12, scan descriptors from the shared pool are allocated. Data-only-locked tables also require a system table descriptor for each data-only-locked table accessed with a table scan (but not those accessed with an index scan). If more than 16 data-only-locked tables are scanned using table scans in a query, auxiliary scan descriptors are allocated for them.

If a scan needs auxiliary scan descriptors after it has used its allotted number, and there are no descriptors available in the shared pool, Adaptive Server displays an error message and rolls back the user transaction.

If none of your queries need additional scan descriptors, you may still want to leave number of aux scan descriptors set to the default value in case your system requirements grow. Set it to 0 only if you are sure that users on your system will never run queries on more than 16 tables and that your tables will always have few or no referential integrity constraints. See "Monitoring scan descriptor usage" on page 193.

If your queries need more scan descriptors, use one of these methods to remedy the problem:

*   Rewrite the query, or break it into steps using temporary tables. For data-only-locked tables, consider adding indexes if there are many table scans.

*   Redesign the table's schema so that it uses fewer scan descriptors, if it uses a large number of referential integrity constraints. You can find how many scan descriptors a query would use by enabling set showplan, noexec on before running the query.

*   Increase the number of aux scan descriptors setting.

The following sections describe how to use sp_monitorconfig to monitor the current and high-water-mark usage to avoid running out of descriptors, and how to estimate the number of scan descriptors you need.

Monitoring scan descriptor usage

sp_monitorconfig reports the number of unused (free) scan descriptors, the number of auxiliary scan descriptors currently being used, the percentage that is active, and the maximum number of scan descriptors used since the server was last started. Run it periodically, at peak periods, to monitor scan descriptor use.

This example shows scan descriptor use with 500 descriptors configured:

```
sp_monitorconfig "aux scan descriptors"
Usage information at date and time: Apr 22 2002  2:49PM.
```

```
Name                         Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt   Instance_Name
-----------------------      ----------  ----------  -------  -----------
----------  --------------------
number of aux                       260         240    48.00          427
          0                 NULL
```

Only 240 auxiliary scan descriptors are being used, leaving 260 free. However, the maximum number of scan descriptors used at any one time since the last time Adaptive Server was started is 427, leaving about 20 percent for growth in use and exceptionally heavy use periods. "Re-used" does not apply to scan descriptors.

Estimating and configuring auxiliary scan descriptors

To get an estimate of scan descriptor use:

1   Determine the number of table references for any query that references more than 16 user tables, or for those that have a large number of referential constraints, by running the query with set showplan and set noexec enabled. If auxiliary scan descriptors are required, showplan reports the number needed:

```
Auxiliary scan descriptors required: 17
```

The reported number includes all auxiliary scan descriptors that are required for the query, including those for all worker processes. If your queries involve only referential constraints, you can also use sp_helpconstraint, which displays a count of the number of referential constraints per table.

2   For each query that uses auxiliary scan descriptors, estimate the number of users who would run the query simultaneously and multiply. If 10 users are expected to run a query that requires 8 auxiliary descriptors, a total of 80 will be needed at any one time.

3   Add the per-query results to calculate the number of needed auxiliary scan descriptors.

number of backup connections

Summary information	
Default value	0
Range of values	1 – 32768

Summary information	
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	User Environment

number of backup connections sets the maximum number of user connections Backup Server establishes to dump or load in-memory databases. The value of number of backup connections restricts the maximum number of stripes for an archived database because Backup Server requires one user connection per stripe when you run dump or load database, and requires an extra connection to run the dump database command.

number of backup connections is a limit, and does not consume any resources. Setting number of backup connections to 0 means that Backup Server can use the maximum number of user connections.

number of ccbs

Summary information	
Default value	0
Range of values	0 – 100
Status	Static
Display level	
Required role	
Configuration group	Diagnostics

Reserved for future use.

number of checkpoint tasks

Summary information	
Default value	1
Valid values	1– 8
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Backup/Recovery

number of checkpoint tasks configures parallel checkpoints. The value of number of checkpoint tasks must be less than or equal to the value of number of engines at start-up. The maximum value is limited by the value of the configuration parameters number of engines online at startup and number of open databases, with a maximum of 8.

The default value sets serial checkpoints as the default behavior.

number of devices

Summary information	
Default value	10
Range of values	1–2,147,483,647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Disk I/O, Memory Use

number of devices controls the number of database devices Adaptive Server can use. It does not include devices used for database or transaction log dumps.

When you execute disk init, you can also assign the virtual device number (the vdevno), although this value is optional. If you do not assign the vdevno, Adaptive Server assigns the next available virtual device number.

If you do assign the virtual device number, each device number must be unique among the device numbers used by Adaptive Server. The number 0 is reserved for the master device. You can enter any unused device number that falls in the valid range of values.

To determine which numbers are currently in use, enter:

```
select vdevno from master..sysdevices
   where status & 2 = 2
```

Here, "status 2" specifies physical disk.

Setting the number of devices on UNIX

If you are using a large number of devices on UNIX platforms, Sybase recommends that you set the appropriate number of devices and user connections in the configuration file. Attempting to configure a large number of devices dynamically using sp_configure may fail.

The number of network connections and number of devices configuration parameters use operating system file descriptors. You must increase the number of operating system file descriptors to use a large value for either configuration parameter.

What constitutes a large number of devices is platform-specific. These are the upper limits for the number of file descriptors for Adaptive Server:

*   HPIA 64, HPPA 64, AIX 64, Windows – Set by Adaptive Server.

*   IBM RISC 64-bit – 2000

*   IBM PLinux – 2048

*   AMD64 Linux – 10000

*   IA32-bit Linux – 10000

*   IA 64-bit Linux – 1024

Note Changing the number of file descriptors requires you restart Adaptive Server for the change to take affect.

If there are insufficient number of file descriptors, Adaptive Server issues an error message similar to:

```
Msg 5893, Level 16, State 1:Procedure 'sp_configure', Line 1234:
The sum, ( ((number of user connections) + (number of remote sites) + (max
cisremote connections) + (number of java sockets)) / (max online engines)) +
(number of devices ) + (max number network listeners),
must be no greater than '958'.
Msg 5849, Level 16, State 1:Procedure 'sp_configure', Line 1234:
Verification failed for parameter 'number of devices'.
(return status = 1)
```

If you then set the number of file descriptors to 2048 with the command `ulimit -n 2048` and restart the server, Adaptive Server reports this message in the error log when you increase the value for the number of devices:

```
kernel Using 2048 file descriptors.
```

number of disk tasks

Summary information	
Default value	1

Summary information	
Range of values	UNIX: 1– 64
	Windows: 1
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Disk I/O, Processors

number of disk tasks controls the number of tasks dedicated to polling and completing disk I/Os.

number of dtx participants

Summary information	
Default value	500
Valid values	100 – 2147483647
Status	Dynamic
Display level	10
Required role	System administrator
Configuration groups	DTM Administration, Memory Use

number of dtx participants sets the total number of remote transactions that the Adaptive Server transaction coordination service can propagate and coordinate simultaneously. A DTX participant is an internal memory structure that the coordination service uses to manage a remote transaction branch. As transactions are propagated to remote servers, the coordination service must obtain new DTX participants to manage those branches.

Setting number of dtx participants to a number smaller than the default reduces the number of remote transactions that the server can manage. If no DTX participants are available, new distributed transactions cannot start. In-progress distributed transactions may abort if no DTX participants are available to propagate a new remote transaction.

Setting number of dtx participants to a number larger than the default increases the number of remote transaction branches that Adaptive Server can handle, but also consumes more memory.

Optimizing the number of DTX participants for your system

During a peak period, use sp_monitorconfig to examine the use of DTX participants:

```
sp_monitorconfig "number of dtx participants"

Usage information at date and time: Apr 22 2002  2:49PM.
Name                      Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt    Instance_Name
----------------------    ----------  ----------  -------  -----------
----------   --------------------
number of dtx                    80          20     4.00          210
         0                 NULL
```

If the num_free value is zero or very low, new distributed transactions may be unable to start due to a lack of DTX participants. Consider increasing the number of dtx participants value.

A low Max_used value may indicate that unused DTX participants are consuming memory that could be used by other server functions. Consider reducing the value of number of dtx participants.

number of dump threads

Summary information	
Default value	Disabled
Range of values	1 (disabled, no parallelism) – 8 (fully parallel)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Group Diagnostics

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.

When you are determining the number of threads for memory:

• Use a value of 8 if the machine has enough free memory for the file system cache to hold the entire memory dump.

- If you do not know whether the machine has enough free memory, the value for number of dump threads depends on many factors, including the speed of the I/O system, the speed of the disks, the controller's cache, whether the dump file lives in a logical volume manager created on several disks, and so on.

- Disable parallel processing (by assigning a value of 1) if you do not halt the engines when performing memory dumps, described below.

  When Adaptive Server performs a memory dump, the number of files it creates is the sum of the number of memory segments that it has allocated multiplied by the number of threads configured. Adaptive Server uses separate threads to write on separate files. When this job completes, the engines are restarted, and the files are merged into the target dump file. Because of this, the time to dump the shared memory in parallel is greater than doing it serially.

- If you halt the engines during the memory dump, using a value other than 1 may reduce the amount of time the engines spend stopped while dumping the memory.

number of engines at startup

Summary information	
Default value	1
Range of values	1 – number of CPUs on machine
Status	Static
Display level	Basic
Required role	System administrator
Configuration groups	Java Services, Memory Use, Processors

Note When configured for threaded mode, Adaptive Server ignores the number of engines at startup configuration parameter. In threaded mode, Adaptive Server uses the size of the defined thread pools to determine the number of online engines at startup. If the configuration file contains no thread pool configuration information (for example, while you are upgrading from an earlier version), Adaptive Server uses the existing value for number of engines at startup to determine the size the default engine pool.

number of engines at startup is used exclusively during start-up to set the number of engines brought online. It allows great flexibility, subject to the restriction that you cannot set the value of number of engines at startup to a value greater than the number of CPUs on your machine, or to a value greater than the configuration of max online engines. Users who do not intend to bring engines online after start-up should set max online engines and number of engines at startup to the same value. A difference between number of engines at startup and max online engines wastes approximately 1.8 MB of memory per engine.

Adaptive Server allows users to take all engines offline, except engine zero.

number of histogram steps

Summary information	
Default value	20
Range of values	3 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

number of histogram steps specifies the number of steps in a histogram.

number of index trips

Summary information	
Default value	0
Range of values	0–65535
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Cache Manager

number of index trips specifies the number of times an aged index page traverses the most recently used/least recently used (MRU/LRU) chain before it is considered for swapping out. As you increase the value of number of index trips, index pages stay in cache for longer periods of time.

A data cache is implemented as an MRU/LRU chain. As the user threads access data and index pages, these pages are placed on the MRU end of the cache's MRU/LRU chain. In some high transaction environments (and in some benchmarks), you may want to keep index pages in cache, since they will probably be needed again soon. Setting number of index trips higher keeps index pages in cache longer; setting it lower allows index pages to be swapped out of cache sooner.

You need not set the number of index trips for relaxed LRU pages. See Chapter 4, "Configuring Data Caches," in the *System Administration Guide: Volume 2*.

Note If the cache used by an index is relatively small (especially if it shares space with other objects) and you have a high transaction volume, do not set number of index trips too high. The cache can flood with pages that do not age out, and this may lead to the timing out of processes that are waiting for cache space.

Before changing the value of number of index trips to a number other than 0, make sure the application has sufficient cache to store all index, OAM, and data pages. Consult Sybase Technical Support before changing the value of number of index trips.

---

number of java sockets

Summary information	
Default value	0
Valid values	0 – 32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Java Services, Memory Use

number of java sockets enables the Java VM and the java.net classes Sybase supports.

number of large i/o buffers

Summary information	
Default value	6

Summary information	
Valid values	1–256
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Disk I/O, Memory Use, SQL Server Administration

number of large i/o buffers sets the number of allocation unit-sized buffers reserved for performing large I/O for certain Adaptive Server utilities. These large I/O buffers are used primarily by the load database command, which uses one buffer to load the database, regardless of the number of stripes it specifies. load database then uses as many as 32 buffers to clear the pages for the database it is loading. These buffers are not used by load transaction. To perform more than six load database commands concurrently, configure one large I/O buffer for each load database command.

create database and alter database use these buffers for large I/O while clearing database pages. Each instance of create database or load database can use as many as 32 large I/O buffers.

These buffers are also used by disk mirroring and by some dbcc commands.

Note In Adaptive Server version 12.5.0.3 and later, the size of the large I/O buffers is one allocation (256 pages), not one extent (8 pages). The server thus requires more memory allocation for large buffers. For example, a disk buffer that required memory for 8 pages in earlier versions now requires memory for 256 pages.

---

number of locks

Summary information	
Default value	5000
Range of values	1000–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Lock Manager, Memory Use

number of locks sets the total number of available locks for all users on Adaptive Server.

The total number of locks needed by Adaptive Server depends on the number of concurrent and parallel processes, and the types of actions performed by the transactions. To see how many locks are in use at a particular time, use sp_lock.

For serial operation, Sybase suggests that you start by assigning 20 locks for each active, concurrent connection.

Parallel execution requires more locks than serial execution. For example, if you find that queries use an average of five worker processes, try increasing by one-third the number of locks configured for serial operation.

If the system runs out of locks, Adaptive Server displays a server-level error message. If users report lock errors, you may need to increase number of locks; but remember that locks use memory. See Chapter 3, "Configuring Memory," in the *System Administration Guide Volume 2*.

Note Datarows locking may require that you change the value for number of locks. See the *Performance and Tuning Series: Locking and Concurrency Control*.

---

number of mailboxes

Summary information	
Default value	30
Range of values	30–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

number of mailboxes specifies the number of mailbox structures allocated by Adaptive Server. Mailboxes, which are used with messages, are used internally by Adaptive Server for communication and synchronization between kernel service processes. Mailboxes are not used by user processes. Do not modify this parameter unless instructed to do so by Sybase Technical Support.

number of messages

Summary information	
Default value	64

Summary information	
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

number of messages specifies the number of message structures allocated by Adaptive Server. Messages, which are used with mailboxes, are used internally by Adaptive Server for communication and synchronization between kernel service processes. Messages are also used to coordinate between a family of processes in parallel processing. Do not modify this parameter unless instructed to do so by Sybase Technical Support.

number of network tasks

Summary information	
Default value	1
Range of values	UNIX: 1– 64
	Windows: 1
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Disk I/O, Processors

number of network tasks controls the number of tasks dedicated to polling and completing network I/Os.

number of oam trips

Summary information	
Default value	0
Range of values	0–65535
Status	Dynamic
Display level	Comprehensive
Required role	System administrator

number of oam trips specifies the number of times an **object allocation map** (OAM) page traverses the MRU/LRU chain before it is considered for swapping out. The higher the value of number of oam trips, the longer aged OAM pages stay in cache.

Each table, and each index on a table, has an OAM page, which holds information on pages allocated to the table or index and is checked when a new page is needed for the index or table. (See "page utilization percent" on page 227.) A single OAM page can hold allocation mapping for between 2,000 and 63,750 data or index pages.

The OAM pages point to the allocation page for each allocation unit where the object uses space. The allocation pages, in turn, track the information about extent and page usage within the allocation unit.

In some environments and benchmarks that involve significant allocations of space (that is, massive bulk copy operations), keeping OAM pages in cache longer improves performance. Setting number of oam trips to a higher value keeps OAM pages in cache.

Note If the cache is relatively small and used by a large number of objects, do not set number of oam trips too high. This may result in the cache being flooded with OAM pages that do not age out, and user threads may begin to time out.

Before changing the value of number of oam trips to a number other than 0, make sure the application has sufficient cache to store all index, OAM, and data pages. Consult Sybase Technical Support before changing the value of number of oam trips.

---

number of open databases

Summary information	
Default value	12
Range of values	6 –2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Caches, SQL Server Administration

number of open databases sets the maximum number of databases that can be open simultaneously on Adaptive Server.

When you calculate a value, include the system databases master, model, sybsystemprocs, and tempdb. If you have installed auditing, include the sybsecurity database. Also, count the sample databases pubs2 and pubs3, the syntax database sybsyntax, and the dbcc database dbccdb if they are installed.

If you are planning to make a substantial change, such as loading a large database from another server, use sp_helpconfig to calculate an estimated metadata cache size by using sp_helpconfig. sp_helpconfig displays the amount of memory required for a given number of metadata descriptors, as well as the number of descriptors that can be accommodated by a given amount of memory. A database metadata descriptor represents the state of the database while it is in use or cached between uses.

❖ **Optimizing the *number of open databases***

If Adaptive Server displays a message saying that you have exceeded the allowable number of open databases, adjust the value.

1   Use sp_countmetadata to find the total number of database metadata descriptors:

```
sp_countmetadata "open databases"
```

The best time to run sp_countmetadata is when there is little activity on the server. Running sp_countmetadata during a peak time can cause contention with other processes.

Suppose Adaptive Server reports the following information:

```
There are 50 databases, requiring 1719 Kbytes of
memory. The 'open databases' configuration parameter
is currently set to 500.
```

2   Configure number of open databases with the value of 50:

```
sp_configure "number of open databases", 50
```

This new configuration number is only a starting point; base the ideal size on the number of active metadata database cache descriptors, not the total number of databases.

3   During a peak period, find the number of active metadata descriptors:

```
sp_monitorconfig "open databases"

Usage information at date and time: Apr 22 2002  2:49PM.
Name                     Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt    Instance_Name
-----------------------  ----------  ----------  -------  -----------
----------   --------------------
```

```
number of open                    50         20        .00            26
        0                     NULL
```

In this example, 20 metadata database descriptors are active; the maximum number of descriptors that have been active since the server was last started is 26.

See sp_monitorconfig in the *Reference Manual: Procedures* for more information.

4   Configure number of open databases to 26, plus additional space for 10 percent more (about 3), for a total of 29:

```
sp_configure "number of open databases", 29
```

If there is a lot of activity on the server, for example, if databases are being added or dropped, periodically run sp_monitorconfig. Reset the cache size as the number of active descriptors changes.

number of open indexes

Summary information	
Default value	500
Range of values	100–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Caches

number of open indexes sets the maximum number of indexes that can be used simultaneously on Adaptive Server.

If you are planning to make a substantial change, such as loading databases with a large number of indexes from another server, use sp_helpconfig to calculate an estimated metadata cache size. sp_helpconfig displays the amount of memory required for a given number of metadata descriptors, as well as the number of descriptors that can be accommodated by a given amount of memory. An index metadata descriptor represents the state of an index while it is in use or cached between uses.

❖ **Optimizing *number of open indexes***

If the default value of number of open indexes is insufficient, Adaptive Server displays a message after trying to reuse active index descriptors, and you must adjust this value.

1   Use sp_countmetadata to find the total number of index metadata descriptors:

```
sp_countmetadata "open indexes"
```

The best time to run sp_countmetadata is when there is little activity in the server. Running sp_countmetadata during a peak time can cause contention with other processes.

Suppose Adaptive Server reports the following information:

```
There are 698 user indexes in all database(s),
requiring 286.289 Kbytes of memory. The 'open
indexes' configuration parameter is currently set to
500.
```

2   Configure the number of open indexes parameter to 698:

```
sp_configure "number of open indexes", 698
```

This new configuration is only a starting point; base the ideal size on the number of active index metadata cache descriptors, not the total number of indexes.

3   During a peak period, find the number of active index metadata descriptors:

```
sp_monitorconfig "open indexes"

Usage information at date and time: Apr 22 2002  2:49PM.
Name                         Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt   Instance_Name
-----------------------      ----------  ----------  -------  -----------
----------  --------------------
number of open                      182         516    73.92          590
        0                    NULL
```

In this example, 590 is the maximum number of index descriptors that have been used since the server was last started.

See sp_monitorconfig in the *Reference Manual: Procedures*.

4   Configure the number of open indexes configuration parameter to 590, plus additional space for 10 percent more (59), for a total of 649:

```
sp_configure "number of open indexes", 649
```

If there is a lot of activity on the server, for example, if tables are being added or dropped, periodically run sp_monitorconfig. Reset the cache size as the number of active descriptors changes.

number of open objects

Summary information	
Default value	500
Range of values	100–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Caches, SQL Server Administration

number of open objects sets the maximum number of objects that can be open simultaneously on Adaptive Server.

If you are planning to make a substantial change, such as loading databases with a large number of objects from another server, use sp_helpconfig to recalculate an estimated metadata cache size. sp_helpconfig displays the amount of memory required for a given number of metadata descriptors, as well as the number of descriptors that can be accommodated by a given amount of memory. An object metadata descriptor represents the state of an object while it is in use, or cached between uses.

❖ **Optimizing *number of open objects***

If the default number of open objects is insufficient, Adaptive Server displays a message after trying to reuse active object descriptors.

1 Use sp_countmetadata to find the total number of object metadata cache descriptors:

```
sp_countmetadata "open objects"
```

The best time to run sp_countmetadata is when there is little activity in the server. Running sp_countmetadata during a peak time can cause contention with other processes.

Suppose Adaptive Server reports this information:

```
There are 696 user objects in all database(s),
requiring 1166 Kbytes of memory. The 'open objects'
```

```
                              configuration parameter is currently set to 500.
```

2   Configure number of open objects to account for the number of open
    objects:

```
        sp_configure "number of open objects", 1407
```

1407 covers the 1340 user objects, plus 5 percent to accommodate
temporary tables.

This new configuration is only a starting point; base the ideal number on
the active object metadata cache descriptors, not the total number of
objects.

3   During a peak period, find the number of active metadata cache
    descriptors:

```
sp_monitorconfig "open objects"
Usage information at date and time: Aug 20 2007  1:32PM..
Name                     Num_free  Num_active  Pct_act  Max_Used  Reuse_cnt
    Instance_Name
--------------------     --------  ----------  -------  --------  ---------
    -------------------
number of open objects        560         847    71.40      1497          0
    NULL
```

In this example, 1497 is the maximum number of object descriptors that
have been used since the server was last started.

4   Configure the number of open objects to 1397, plus 10 percent (140), for a
    total of 1537:

```
        sp_configure "number of open objects", 1537
```

If there is a lot of activity on the server, for example, if tables are being added
or dropped, periodically run sp_monitorconfig. Reset the cache size as the
number of active descriptors changes. See sp_monitorconfig in the *Reference
Manual: Procedures*.

number of open partitions

Summary information	
Default value	500
Range of values	100 – 2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator

Summary information	
Configuration groups	Memory Use, Meta-Data Caches

Specifies the number of partitions that Adaptive Server can access at one time.

Optimizing the *number of open partitions* parameter for your system

If the default value of number of open partitions is insufficient, Adaptive Server displays a message after trying to reuse active partition descriptors. You must adjust this value.

1 Use sp_countmetadata to find the total number of open partitions. For example:

```
sp_countmetadata "open partitions"
```

The best time to run sp_countmetadata is when there is little activity in the server. Running sp_countmetadata during a peak time can cause contention with other processes.

Suppose Adaptive Server reports the following information:

```
There are 42 user partitions in all database(s),
requiring 109 Kbytes of memory. The 'open
partitions' configuration parameter is currently set
to 110.
```

2 Configure number of open partitions to 110, as reported by sp_countmetadata:

```
sp_configure "number of open partitions", 110
```

3 During a peak period, find the number of active metadata cache descriptors, for example:

```
sp_monitorconfig "open partitions"
Usage information at date and time: Jun 30 2008  3:15PM.

Name                      Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt    Instance_Name
----------------------    ----------  ----------  -------  -----------
----------   -------------------
number of open partitions    27            57       51.8         83
         0                   NULL
```

In this example, 83 is the maximum number of partition descriptors that have been used since the server was last started.

4 Configure the number of open partitions to 83, plus 10 percent (8), for a total of 91:

```
sp_configure "number of open partitions", 91
```

If there is a lot of activity on the server, for example, if tables are being added or dropped, periodically run sp_monitorconfig. Reset the cache size as the number of active descriptors changes. See sp_monitorconfig in the *Reference Manual: Procedures*.

number of pre-allocated extents

Summary information	
Default value	2
Range of values	1–32
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

number of pre-allocated extents specifies the number of extents (eight pages) allocated in a single trip to the page manager. Currently, this parameter is used only by bcp to improve performance when copying in large amounts of data. By default, bcp allocates two extents at a time and writes an allocation record to the log each time.

Setting number of pre-allocated extents means that bcp allocates the specified number of extents each time it requires more space, and writes a single log record for the event.

An object may be allocated more pages than actually needed, so the value of number of pre-allocated extents should be low if you are using bcp for small batches. If you are using bcp for large batches, increase the value of number of pre-allocated extents to reduce the amount of overhead required to allocate pages and to reduce the number of log records.

Using a value of 32 for the number of pre-allocated extents

Using a value of 32 for number of pre-allocated extents has a special significance for configuration and impacts the space allocations Adaptive Server performs internally. If you set number of pre-allocated extents to 32, Adaptive Server reserves an entire allocation unit worth of extents for utility operations like bcp-in and select into, both of which use the large-scale allocation scheme of space reservation. This greatly improves the performance of these utilities, particularly when you run them concurrently on multiple nodes. Consequently, using a value of 32 guarantees that each node of a cluster is able to work independently on its own allocation unit without interference from the other nodes.

In earlier versions of Adaptive Server, the number of pre-allocated extents parameter specified the number of extents reserved in a single allocation call for tables of all sizes.

With this version of Adaptive Server, the value of number of pre-allocated extents is ignored for large tables with 240 or more pages for these commands only:

- alter table *table_name* add *column_name* . . .

- alter table *table_name* modify *column_name* . . .

- alter table *table_name* drop *column_name* . . .

- alter table lock . . .

- reorg rebuild

When you run these command on tables larger than 240 pages, Adaptive Server reserves an entire allocation unit (32 extents), which greatly improves performance, particularly when you run them concurrently on multiple nodes.

The value of number of pre-allocated extents continues to be observed for the above commands for tables with fewer than 240 pages, and for all commands (such as select into, bcp, alter table partition) for tables of all sizes.

number of Q engines at startup

Summary information	
Default value	0
Range of values	0 – 127
Status	Static
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration group	Processors

number of Q engines at startup, which specifies the number of Q engines that are online when the server starts, is required for MQ. You may need to increase max online engines to accommodate the number of max online Q engines.

number of remote connections

Summary information	
Default value	20
Range of values	5–32767
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Network Communication

number of remote connections specifies the number of logical connections that can simultaneously be open to and from an Adaptive Server. Each simultaneous connection to XP Server for ESP execution uses up to one remote connection each. See Chapter 7, "Managing Remote Servers."

number of remote logins

Summary information	
Default value	20
Range of values	0–32767
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Network Communication

number of remote logins controls the number of active user connections from Adaptive Server to remote servers. Each simultaneous connection to XP Server for ESP execution uses up to one remote login each. Set this parameter to the same (or a lower) value as number of remote connections. See Chapter 7, "Managing Remote Servers."

number of remote sites

Summary information	
Default value	10
Range of values	0–32767
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Network Communication

number of remote sites determines the maximum number of remote sites that can simultaneously access Adaptive Server. Each Adaptive Server-to-XP Server connection uses one remote site connection.

Internally, number of remote sites determines the number of site handlers that can be active at any one time; all server accesses from a single site are managed with a single site handler. For example, if you set number of remote sites to 5, and each site initiates three remote procedure calls, sp_who shows 5 site handler processes for the 15 processes. See Chapter 7, "Managing Remote Servers."

number of sort buffers

Summary information	
Default value	500
Range of values	0–32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

number of sort buffers specifies the amount of memory allocated for buffers used to hold pages read from input tables and perform index merges during sorts. number of sort buffers is used only for parallel sorting.

Adaptive Server allocates sort buffers from the cache to which the table is bound. If the cache is not bound to a table, Adaptive Server allocates the sort buffers from the cache to which the database is bound. If the cache is not bound to a table or a database, Adaptive Server allocates sort buffers from the default data cache.

Parallel sorts are used when you:

- Run updates statistics

- Create indexes

See Chapter 10, "Using Statistics to Improve Performance," in the *Performance and Tuning Series: Query Processing and Abstract Plans*.

The value you use for number of sort buffers depends on the page size of the server.

Sybase recommends that you leave this parameter set to the default except when you are creating indexes in parallel.

Setting the value too high can rob nonsorting processes of access to the buffer pool in caches being used to perform sorts.

If you configure a high number of sort buffers, a sort on a large table may require more procedure cache. The effect is more pronounced with tables that have smaller row sizes, because the number of rows per page is higher.

This equation estimates the amount of procedure cache required (in bytes):

(*Number of sort buffers*) X (*rows per page*) X 100

If you do not configure enough procedure cache for the number of sort buffers, the sort may fail with error message 701. If this occurs, reconfigure Adaptive Server with a lower number of sort buffers and retry the sort.

See "max buffers per lava operator" on page 182 for information about setting an upper limit for the number of buffers used by an operator.

number of user connections

Summary information	
Default value	25
Range of values	5–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, User Environment

number of user connections sets the maximum number of user connections that can simultaneously be connected to Adaptive Server. It does not refer to the maximum number of processes; that number depends not only on the value of this parameter but also on other system activity.

| Upper limit to the *maximum number of user connections* | The maximum allowable number of file descriptors per process is operating-system-dependent; see the configuration documentation for your platform. |

The number of file descriptors available for Adaptive Server connections is stored in the global variable @@*max_connections*. You can report the maximum number of file descriptors your system can use with:

```
select @@max_connections
```

The return value represents the maximum number of file descriptors allowed by the system for your processes, minus overhead. Overhead increases with the number of engines. For more information on how multiprocessing affects the number file descriptors available for Adaptive Server connections, see Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2.

In addition, you must reserve a number of connections for the following items, which you also set with configuration parameters:

• The database devices, including mirror devices

• Site handlers

• Network listeners

The number of user connections + (number of devices * max online engines * 2) + number of remote sites + max number network listeners cannot be greater than the value of @@*max_connections*.

| Reserved connections | One connection from the configured number of connections is reserved for temporary administrative tasks to make sure that database administrators can connect to Adaptive Server. A reserved connection has a total login time of 15 minutes, and can be is allocated only to a user who has the sa_role. Adaptive Server terminates the connection after 15 minutes to ensure the availability of the reserved connection at an installation with multiple database administrators. |

Adaptive Server also automatically uses this reserved connection when a client uses the last resource for connecting to Adaptive Server.

If Adaptive Server is using a reserved connection, the following informational message appears when the user logs in to Adaptive Server:

```
There are not enough user connections available; you are being connected
using a temporary administrative connection which will time out after '15'
minutes. Increase the value of th 'number of user connections' parameter
```

Adaptive Server also prints a message similar to the following to the error log when the final connection to Adaptive Server terminates due to a timeout:

```
00:00000:00008:2003/03/14 11:25:31.36 server  Process '16' has been
terminated as it exceeded the maximum login time allowed for such processes.
This process used a connection reserved for system administrators and has a
maximum login period of '15' minutes
```

Optimizing *max number of user connections*

There is no formula to determine how many connections to allow for each user. You must estimate this number, based on the system and user requirements. You must also take into account that on a system with many users, connections needed only occasionally or transiently can generally be shared among users. The following processes require user connections:

• One connection is needed for each user running isql.

• Application developers use one connection for each editing session.

• The number of connections required by users running an application depends on how the application has been programmed. Users executing Open Client programs need one connection for each open DB-Library dbprocess or Client-Library™ cs_connection.

> **Note** Sybase suggests that you estimate the maximum number of connections used by Adaptive Server and update number of user connections as you add physical devices or users to the system. Periodically use sp_who to determine the number of active user connections on your Adaptive Server.

Certain other configuration parameters, including stack size and default network packet size, affect the amount of memory for each user connection.

User connections for shared memory—EJB Server

Adaptive Server uses the value of number of user connections to establish the number of shared-memory connections for EJB Server. Thus, if number of user connections is 30, Adaptive Server establishes 10 shared-memory connections for EJB Server. Shared-memory connections are not a subset of user connections, and are not subtracted from the number of user connections.

To increase the number of user connections for shared memory, you must:

1   Increase number of user connections to a number one-third of which is the number of desired shared-memory connections.

2   Restart Adaptive Server.

Although number of user connections is a dynamic configuration parameter, you must restart the server to change the number of user connections for shared memory. See the *EJB Server Users Guide*.

With Adaptive Server version 12.5.3 ESD #2, no sockets are automatically reserved for EJB. However, you can enable trace flag 1642 to revert to the functionality of earlier version, reserving one-third of the sockets for EJB. Enable traceflag 1642 to set up the EJB server. For this version of Adaptive Server, you can ignore this message, `"hbc_ninit: No sockets available for HBC"`, in the error log if the EJB server is not configured.

In Adaptive Server version 12.5.3 and later, if the EJB server is enabled and HBC sockets are not available, `"hbc_ninit: No sockets available for HBC"` is reported. If traceflag 1642 is not enabled, set the flag, and restart Adaptive Server. If the EJB server is not enabled, then no message is reported and Adaptive Server automatically disables the sockets reserved for EJB server.

number of worker processes

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Query Tuning

number of worker processes specifies the maximum number of worker processes that Adaptive Server can use at any one time for all simultaneously running parallel queries.

Adaptive Server issues a warning message at start-up if there is insufficient memory to create the specified number of worker processes. memory per worker process controls the memory allocated to each worker process.

If you have not configured number of worker processes for a sufficient number of threads from the worker thread pool, Adaptive Server adjusts query plans at runtime to use fewer worker threads. If Adaptive Server cannot adjust the queries at runtime, the queries recompile serially. However, alter table and execute immediate commands are aborted if they do not have sufficient worker threads.

o/s file descriptors

Summary information	
Default value	0
Range of values	Site-specific
Status	Read-only
Display level	Comprehensive
Required role	System administrator
Configuration group	O/S Resources

o/s file descriptors indicates the maximum per-process number of file descriptors configured for your operating system. This parameter is read-only and cannot be configured through Adaptive Server.

Many operating systems allow you to configure the number of file descriptors available per process. See your operating system documentation.

The number of file descriptors available for Adaptive Server connections, which is less than the value of o/s file descriptors, is stored in the variable *@@max_connections*. See "Upper limit to the maximum number of user connections" on page 218.

object lockwait timing

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

object lockwait timing controls whether Adaptive Server collects timing statistics for requests of locks on objects.

open index hash spinlock ratio

Summary information	
Default value	100
Range of values	1–2147483647

Summary information	
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Cache

open index hash spinlock ratio sets the number of index metadata descriptor hash tables that are protected by one **spinlock**. This parameter is used only in multiprocessing systems only.

All the index descriptors belonging to a table are accessible through a hash table. When you run a query on the table, Adaptive Server uses hash tables to look up the necessary index information in its sysindexes rows. A hash table is an internal mechanism used by Adaptive Server to retrieve information quickly.

Usually, you do not need to change this parameter. In rare instances, however, you may need to reset it if Adaptive Server demonstrates contention from hash spinlocks. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

For more information about configuring spinlock ratios, see Chapter 5, "Managing Multiprocessor Servers," in the System Administration Guide: Volume 2.

open index spinlock ratio

Summary information	
Default value	100
Range of values	1–214748364
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Cache

open index spinlock ratio specifies the number of index metadata descriptors that are protected by one **spinlock**.

Adaptive Server uses a spinlock to protect an index descriptor, since more than one process can access the contents of the index descriptor. open index spinlock ratio is used only in multiprocessing systems.

The value specified for this parameter defines the ratio of index descriptors per spinlock.

If one spinlock is shared by too many index descriptors, it can cause spinlock contention. Use sp_sysmon to get a report on spinlock contention. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

If sp_sysmon output indicates an index descriptor spinlock contention of more than 3 percent, try decreasing the value of open index spinlock ratio.

See Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2.

open object spinlock ratio

Summary information	
Default value	100
Range of values	1–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Meta-Data Cache

open object spinlock ratio specifies the number of object descriptors that are protected by one **spinlock**. Adaptive Server uses a spinlock to protect an object descriptor, since more than one process can access the contents of the object descriptor. open object spinlock ratio is used only in multiprocessing systems..

The default value for this parameter is 100; 1 spinlock for each 100 object descriptors configured for your server. If your server is configured with only one engine, Adaptive Server sets only 1 object descriptor spinlock, regardless of the number of object descriptors.

If one spinlock is shared by too many object descriptors, it causes spinlock contention. Use sp_sysmon to get a report on spinlock contention. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

If sp_sysmon output indicates an object descriptor spinlock contention of more than 3 percent, try decreasing the value of the open object spinlock ratio parameter.

See Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2.

optimization goal

Summary information	
Default value	allrows_mix
Range of values	allrows_oltp, allrows_dss
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

Optimization goals are a convenient way to match user query demands with the best optimization techniques, ensuring optimal use of the optimizer's time and resources. Adaptive Server allows users to configure for two optimization goals, which you can specify at three tiers: server level, session level, and query level.

The server-level optimization goal is overridden at the session level, which is overridden at the query level.

These optimization goals allow you to choose an optimization strategy that best fits your query environment:

* allrows_oltp – the most useful goal for purely OLTP queries.

* allrows_dss – the most useful goal for operational DSS queries of medium-to-high complexity.

optimization timeout limit

Summary information	
Default value	10
Range of values	0 – 1000
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

optimization timeout limit specifies the amount of time, as a fraction of the estimated execution time of the query, that Adaptive Server can spend optimizing a query.

A value of 0 indicates there is no optimization timeout.

optimizer level

Summary information	
Default value	ase_default
Range of values	• ase_current – Enables all optimizer changes through the current release
	• ase_default – Disables all optimizer changes since version 1503 ESD #1
	• ase1503esd2– Enables all optimizer changes through version 15.0.3 ESD #2
	• ase1503esd3 – Enables all optimizer changes through version 15.0.3 ESD #3
Status	Dynamic
Display level	
Required role	System administrator
Configuration group	Query Tuning

optimizer level determines the level of optimization the query processor uses. See Chapter 7, Controlling Optimization, in the *Performance and Tuning Series: Query Processing and Abstract Plans* for information about optimization levels.

The values for optimizer level are character data, so you must use 0 as a placeholder for the second parameter, which must be numeric.

page lock promotion HWM

Summary information	
Default value	200
Range of values	2–2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

page lock promotion HWM (high-water mark), with page lock promotion LWM (low-water mark) and page lock promotion PCT (percentage), specifies the number of page locks permitted during a single scan session of a page-locked table or index before Adaptive Server attempts to escalate from page locks to a table lock.

When the number of page locks acquired during a scan session exceeds page lock promotion HWM, Adaptive Server attempts to acquire a table lock. page lock promotion HWM value cannot be higher than number of locks.

For more detailed information on scan sessions and setting up page lock promotion limits, see Chapter 2, "Locking Configuration and Tuning," in the *Performance and Tuning Series: Locking and Concurrency Control*.

The default value for page lock promotion HWM is appropriate for most applications. To avoid table locking, you may want to increase the value. For example, if you know that there are regular updates to 500 pages of an allpages-locked or datapages-locked table containing thousands of pages, increase concurrency for the tables by setting page lock promotion HWM to 500.

You can also configure lock promotion of page-locked tables and views at the object level. See sp_setrowlockpromote in the *Reference Manual: Procedures*.

Use sp_sysmon to see how changing page lock promotion HWM affects the number of lock promotions. sp_sysmon reports the ratio of exclusive page to exclusive table lock promotions and the ratio of shared page to shared table lock promotions. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

page lock promotion LWM

Summary information	
Default value	200
Range of values	2–value of page lock promotion HWM
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

page lock promotion LWM (low-water mark), with page lock promotion HWM (high-water mark) and the page lock promotion PCT, specify the number of page locks permitted during a single scan session of a page locked table or an index before Adaptive Server attempts to promote from page locks to a table lock.

The page lock promotion LWM sets the number of page locks below which Adaptive Server does not attempt to issue a table lock on an object. page lock promotion LWM must be less than or equal to page lock promotion HWM.

The default value for page lock promotion LWM is sufficient for most applications. If Adaptive Server runs out of locks (except for an isolated incident), increase number of locks.

See the *Performance and Tuning Series: Locking and Concurrency Control*.

You can also configure page lock promotion at the object level. See sp_setpglockpromote in the *Reference Manual: Procedures*.

page lock promotion PCT

Summary information	
Default value	100
Range of values	1–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

If the number of locks held on an object is between page lock promotion LWM (low-water mark) and page lock promotion HWM (high-water mark). page lock promotion PCT sets the percentage of page locks (based on the table size) above which Adaptive Server attempts to acquire a table lock.

See Chapter 2, "Locking Configuration and Tuning," in the *Performance and Tuning Series: Locking and Concurrency Control*.

The default value for page lock promotion PCT is appropriate for most applications.

You can also configure lock promotion at the object level for page locked objects. See sp_setpglockpromote in the *Reference Manual: Procedures*.

page utilization percent

Summary information	
Default value	95
Range of values	1–100
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Disk I/O

page utilization percent is used during page allocations to control whether Adaptive Server scans a table's object allocation map (OAM) to find unused pages or simply allocates a new extent to the table. See "number of oam trips" on page 205 for more information on the OAM. The page utilization percent parameter is a performance optimization for servers with very large tables; it reduces the time needed to add new space.

If you set page utilization percent to 100, Adaptive Server scans through all OAM pages to find unused pages allocated to the object before allocating a new extent. When this parameter is set lower than 100, Adaptive Server compares the page utilization percent setting to the ratio of used and unused pages allocated to the table, as follows:

```
100 * used pages/(used pages + unused pages)
```

If page utilization percent is lower than the ratio, Adaptive Server allocates a new extent instead of searching for the unused pages.

For example, when inserting data into a 10GB table that has 120 OAM pages and only 1 unused data page:

- A page utilization percent of 100 tells Adaptive Server to scan through all 120 OAM pages to locate an unused data page.

- A page utilization percent of 95 allows Adaptive Server to allocate a new extent to the object, because 95 is lower than the ratio of used pages to used and unused pages.

A low page utilization percent value results in more unused pages. A high page utilization percent value slows page allocations in very large tables, as Adaptive Server performs an OAM scan to locate each unused page before allocating a new extent. This increases logical and physical I/O.

If page allocations (especially in the case of large inserts) seem to be slow, lower the value of page utilization percent, but reset it after inserting the data. A lower setting affects all tables on the server and results in unused pages in all tables.

Fast bulk copy ignores the page utilization percent setting and always allocates new extents until there are no more extents available in the database.

partition groups

Summary information	
Default value	1024
Range of values	1–2147483647

Summary information	
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Cache

partition groups specifies the maximum number of partition groups that can be allocated by Adaptive Server. Partition groups are internal structures used by Adaptive Server to control access to individual partitions of a table. Partition groups are used while upgrading or during a load database upgrade to unpartition Adaptive Server 12.5.x and earlier partitions.

The default value allows a maximum 1024 open partition groups and a maximum of 2147483647 open partitions. The actual number of partitions may be slightly less, due to the grouping of partitions.

partition spinlock ratio

Summary information	
Default value	10
Range of values	1–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Meta-Data Cache

For Adaptive Servers running with multiple engines, partition spinlock ratio sets the number of rows in the partition descriptors that are protected by one **spinlock**.

Adaptive Server manages access to table partitions using partition descriptors. Each partition descriptor stores information about a partition (for example, the last page of the partition) that processes must use when accessing that partition. Configure partition descriptors using number of open partitions.

The default value of partition spinlock ratio sets 1 spinlock for every 10 partition caches. Decreasing the value of partition spinlock ratio may have little impact on the performance of Adaptive Server. The default setting is correct for most servers.

See Chapter 5, Managing Mulitprocessor Servers," in System Administration Guide: Volume 2.

pci memory size

Summary information	
Default value	64MB
Valid values	0 – 2147483647
Status	Dynamic
Display level	Intermediate
Required role	System Administrator
Configuration group	User Environment

pci memory size sets the size of the pluggable component interface (PCI) memory pool. All pluggable component adapter (PCA) and JVM plug-ins running under the PCI Bridge share a single dedicated PCI memory pool. If you set pci memory size to less than the default, Adaptive Server uses the default size.

This memory pool is fully dedicated to the PCI bridge and any running pluggable component. Like all other memory pools, Adaptive Server controls this memory pool. However, unlike other memory pools, the PCI memory pool is allocated when you initialize the PCI Bridge and does not grow after that time.

per object statistics active

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

per object statistic active controls whether Adaptive Server collects statistics for each object.

percent database for history

Summary information	
Default value	20
Valid values	0 – 100

Summary information

Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

percent database for history specifies the percentage of the total space available in sybmgmtdb that is reserved for the js_history table. Increase percent database for history if there are more jobs running, or to store historical records about executed jobs for future queries.

percent database for output

Summary information

Default value	30
Valid values	0 – 100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

percent database for output specifies the percentage of the total space available in sybmgmtdb that is reserved for job output. Increase the default value if there are more jobs running or jobs that produce lot of output that must be stored for querying.

percent history free

Summary information

Default value	30
Valid values	0 – 100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

percent history free specifies the percentage of reserved space in sybmgmtdb to be kept free For example, if you use the default value, Adaptive Server starts purging the oldest history records to make room for new records when 70 percent of sybmgmtdb is filled.

percent output free

Summary information	
Default value	50
Valid values	0 – 100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

Specifies the percentage of reserved space kept free in sybmgmtdb that is reserved for Job Scheduler output. For example, if you use the default value, Adaptive Server starts purging the oldest history records to make room for new records when 50 percent of sybmgmtdb is filled.

performance monitoring option

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

performance monitoring option enables the license for the BMC DBXray graphical performance monitoring and diagnostic tool .

permission cache entries

Summary information	
Default value	64
Range of values	1–2147483647

Summary information

Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, User Environment

permission cache entries determines the number of cache protectors per task, increasing the amount of memory for each user connection and worker process.

Information about user permissions is held in the permission cache. When Adaptive Server checks permissions, it looks first in the permission cache; if it does not find what it needs, it looks in the sysprotects table. This process is significantly faster if Adaptive Server finds the information it needs in the permission cache and does not have to read sysprotects.

However, Adaptive Server looks in the permission cache only when it is checking user permissions, not when permissions are being granted or revoked. When a permission is granted or revoked, the entire permission cache is flushed. This is because existing permissions have timestamps that become outdated when new permissions are granted or revoked.

If users on your Adaptive Server frequently perform operations that require their permissions to be checked, you may see a small performance gain by increasing the value of permission cache entries. This effect is not likely to be significant enough to warrant extensive tuning.

If users on your Adaptive Server frequently grant or revoke permissions, avoid setting permission cache entries to a large value. The space used for the permission cache would be wasted, since the cache is flushed with each grant and revoke command.

plan text pipe active

Summary information

Default value	0
Range of values	0–1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

plan text pipe active determines whether Adaptive Server collects query plan text. If both plan text pipe active and plan text pipe max messages are enabled, Adaptive Server collects the plan text for each query. You can use monSysPlanText to retrieve the query plan text for all user tasks.

plan text pipe max messages

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

plan text pipe max messages determines the number of query plan text messages Adaptive Server stores per engine. The total number of messages in the monSQLText table is the value of sql text pipe max messages multiplied by the number of engines running.

print deadlock information

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on), 2 (on, print summary)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

print deadlock information prints deadlock information to the error log.

If you are experiencing recurring deadlocks, setting print deadlock information to 1 provides you with detailed information in the error log that can be useful in tracing the cause of the deadlocks. However, setting print deadlock information to 1 can degrade Adaptive Server performance. For this reason, set print deadlock information on only when you are trying to determine the cause of deadlocks.

Use sp_sysmon output to determine whether deadlocks are occurring in your application. If they are, set print deadlock information to 1 to learn more about why they are occurring. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

A value of 2 allows you to print a summary of deadlock information to the error log (as opposed to the detailed information a value of 1 provides). For example:

```
Deadlock Id 34: Process (Familyid 0, Spid 70) was waiting for a 'exclusive page'
lock on page 10858346 of the 'equineline_job' table in database 18 but process
(Familyid 0, Spid 88) already held a 'exclusive page' lock on it.
Deadlock Id 34: Process (Familyid 0, Spid 88) was waiting for a 'exclusive page'
lock on page 11540986 of the 'equineline_job' table in database 18 but process
(Familyid 0, Spid 70) already held a 'update page' lock on it.
```

print recovery information

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Backup/Recovery

print recovery information determines what information Adaptive Server displays on the console during recovery. (Recovery is performed on each database at Adaptive Server start-up and when a database dump is loaded.) The default value means that Adaptive Server displays only the database name and a message saying that recovery is in progress. A value of 1 indicates that Adaptive Server displays information about each individual transaction processed during recovery, including whether it was aborted or committed.

procedure cache size

Summary information	
Default value	7000
Range of values	7000 – 2147483647
Status	Dynamic
Display level	Basic
Required role	System administrator

Summary information	
Configuration groups	Memory Use, SQL Server Administration

Specifies the size of the procedure cache, in 2K pages. Adaptive Server uses the procedure cache while running stored procedures. If the server finds a copy of a procedure already in the cache, it does not need to read it from the disk. Adaptive Server also uses space in the procedure cache to compile queries while creating stored procedures.

Since the optimum value for procedure cache size differs from application to application, resetting it may improve Adaptive Server performance. For example, if you run many different procedures or ad hoc queries, your application uses the procedure cache more heavily, so you may want to increase this value.

Warning! If procedure cache size is too small, Adaptive Server performance degrades.

If you are upgrading, procedure cache size is set to the size of the original procedure cache at the time of upgrade.

procedure deferred compilation

Summary information	
Default value	1(enabled)
Range of values	0 – 1
Status	dynamic
Display level	
Required role	System administrator
Configuration groups	Query tuning

When this parameter is enabled, compiling of statements that reference local variables or temporary tables inside a stored procedure is postponed to execution time, so that the optimization of those statements can use runtime values, instead of estimations or magic numbers.

process wait events

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

process wait events controls whether Adaptive Server collect statistics for each wait event for every task. You can get wait information for a specific task using monProcessWaits.

See Chapter 17, "Using Stored Procedures," in the *Transact-SQL Users Guide*.

prod-consumer overlap factor

Summary information	
Default value	20
Range of values	
Status	dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	

prod-consumer overlap factor affects optimization. Adaptive Server changes the group by algorithm, and you cannot use set statistics I/O with parallel plans.

quorum heartbeat interval

Summary information	
Default value	5
Valid values	1 – 60
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

quorum heartbeat interval specifies the number of seconds between quorum heartbeats. Setting quorum heartbeat interval to a lower number increases the heartbeat overhead but speeds the detection of a lost disk link, resulting in a quicker termination of an instance for which you have set I/O fencing or that has lost its SAN link. Setting quorum heartbeat interval to a high number reduces heartbeat overhead, but delays the detection of a lost disk link.

quorum heartbeat retries

Summary information	
Default value	2
Valid values	0 – 32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

quoted identifier enhancements

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Application Functionality

quoted identifier enhancement enables Adaptive Server to use quoted identifiers for:

- Tables

- Views

- Column names

- Index names

- System procedure parameters

See the *Reference Manual: Blocks*.

read committed with lock

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Lock Manager

read committed with lock determines whether an Adaptive Server using transaction isolation level 1 (read committed) holds shared locks on rows or pages of data-only-locked tables during select queries. For cursors, read committed with lock applies only to read-only cursors declared.

For transaction isolation level 1, select queries on allpages-locked tables continue to hold locks on the page at the current position. Any updatable cursor on a data-only-locked table also holds locks on the current page or row. See the *Performance and Tuning Series: Basics*.

recovery interval in minutes

Summary information	
Default value	5
Range of values	1–32767
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration group	Backup/Recovery

recovery interval in minutes sets the maximum number of minutes per database that Adaptive Server uses to complete its recovery procedures in case of a system failure. The recovery procedure rolls transactions backward or forward, starting from the transaction that the checkpoint process indicates as the oldest active transaction. The recovery process has more or less work to do, depending on the value of recovery interval in minutes.

Adaptive Server estimates that 6000 rows in the transaction log require 1 minute of recovery time. However, different types of log records can take more or less time to recover. If you set recovery interval in minutes to 3, the checkpoint process writes changed pages to disk only when syslogs contains more than 18,000 rows since the last checkpoint.

Note The recovery interval has no effect on long-running, minimally logged transactions (such as create index) that are active when Adaptive Server fails. It may take as much time to reverse these transactions as it took to run them. To avoid lengthy delays, dump each database after index maintenance operations.

Adaptive Server uses the recovery interval in minutes setting and the amount of activity on each database to decide when to checkpoint each database. When Adaptive Server checkpoints a database, it writes all **dirty pages** (data pages in cache that have been modified) to disk. This may create a brief period of high I/O, called a checkpoint spike. The checkpoint also performs other maintenance tasks, including truncating the transaction log for each database for which the truncate log on chkpt option has been set. About once per minute, the sleeping checkpoint process "wakes up," checks the truncate log on chkpt setting, and checks the recovery interval to determine if a checkpoint is needed. Figure 5-4 shows the logic used by Adaptive Server during this process.

Figure 5-4: The checkpoint process
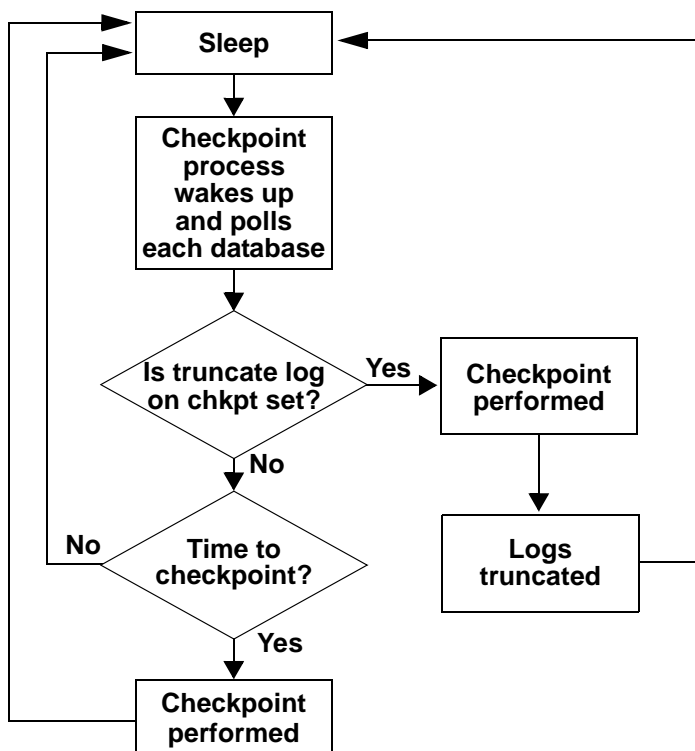
You may want to change the recovery interval if your application and its use change. For example, you may want to shorten the recovery interval when there is an increase in update activity on Adaptive Server. Shortening the recovery interval causes more frequent checkpoints, with smaller, more frequent checkpoint spikes, and slows the system slightly. However, setting the recovery interval too high may cause the recovery time to be unacceptably long. You can reduce the spikes caused by checkpointing by reconfiguring the housekeeper freewrite percent parameter. See "housekeeper free write percent" on page 149. For more information on the performance implications of recovery interval in minutes, see Chapter 5, "Memory Use and Performance," in the *Performance and Tuning Series: Basics*.

Use sp_sysmon to determine how a particular recovery interval affects the system. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

remote server pre-read packets

Summary information	
Default value	3
Range of values	3–255
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Network Communication

remote server pre-read packets determines the number of packets that are "pre-read" by a site handler during connections with remote servers.

To reduce the required number of connections, communication between two servers is managed through a single site handler. The site handler can pre-read and keep track of data packets for each user process before the receiving process is ready to accept them.

The default value for remote server pre-read packets is appropriate for most servers. Increasing the value uses more memory; decreasing the value can slow network traffic between servers. See Chapter 7, "Managing Remote Servers."

restricted decrypt permission

Summary information	
Default value	0
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Basic
Required role	System security officer
Configuration group	Security Related

restricted decrypt permission enables or disables restricted decrypt permission in all databases. You must have the sso_role to set this parameter.

When restricted decrypt permission is set to 0 (off), decrypt permission on encrypted columns acts the same as in versions earlier than 15.0.2:

•   The table owner or the SSO explicitly grants decrypt permission. However, with grant option on decrypt permission is supported.

- Decrypt permission is granted implicitly to table owners and the SSO, as well as to any user through a chain of ownership. For example, if user Fred owns the proc1 stored procedure, which selects data from the encrypted column fred.table1.col1, and if Fred grants exec permission on proc1 to Harry, then Harry has implicit decrypt permission on fred.table1.col1

- Decrypt permission is not needed for alter table decrypt. because the table owner has implicit decrypt permission on encrypted columns.

When restricted decrypt permission is set to 1 (on):

- Decrypt permission is granted implicitly only to the SSO.

- The SSO can grant decrypt permission using the with grant option parameter. This allows the SSO to decide who can grant decrypt permission in the system. For example, if the SSO wants user1 to be able to grant decrypt permission on user3.user3_tab, the SSO issues:

  ```
  grant decrypt on user3.user3_tab to user1
  with grant option
  ```

  If you use a system encryption password, Sybase recommends that, to protect data privacy, you do not grant decrypt permission to the DBO to. Access to keys through user passwords prevents the DBO and other parties from accessing the data unless they have a key's password; however, you may find it convenient for the DBO to decide which users should see the decrypted data. If you are not protecting keys and data with user-specified passwords, the SSO should retain the sole responsibility to grant decrypt permission.

- Table ownership does not give a user implicit decrypt permission. That is, if you create a table with encrypted columns, you do not have decrypt permission on them unless it is explicitly granted to you.

- No user is implicitly granted decrypt permission through an ownership chain. For example, if Fred owns the proc1 stored procedure, which selects data from the encrypted column fred.table1.col1, and if Fred grants exec permission on proc1 to Harry, then Harry must also have explicit decrypt permission on fred.table1.col1 to see the data.

- Aliased users assume the permissions of the user to whom they are aliased. Similarly, a user with sa_role, who is implicitly aliased to the DBO in any database, inherits any decrypt permissions that have been explicitly granted to the DBO.

- Decrypt permission is required for alter table decrypt statement because the table owner does not have implicit decrypt permission on the table.

If you change restricted decrypt permission from 0 to 1, currently executing statements that use implicit decrypt permission finish; however any subsequent statements that use implicit decrypt permission fail with this error until the SSO grants the user decrypt permission on the necessary columns:

```
Msg 10330 "DECRYPT permission denied on object object_name, database
database_name, owner owner_name."
```

If you change restricted decrypt permission from 1 to 0, the rows that reflect explicit grants remain in the sysprotects system table. However, these rows have no effect on implicitly granted decrypt permissions because Adaptive Server does not check sysprotects to make sure decrypt permission can be implicitly granted. sp_helprotect displays misleading information for only those users who were granted or revoked explicit decrypt permission before you reconfigure the system, and who now have implicit decrypt permission.

Sybase recommends that, to keep the system consistent, you revoke any explicit decrypt permissions granted to users before you switch between enabling or disabling restricted decrypt permission to keep the system consistent.

See the *Encrypted Columns Users Guide* for more information about decrypt permissions.

row lock promotion HWM

Summary information	
Default value	200
Range of values	2–2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

row lock promotion HWM (high-water mark), with row lock promotion LWM (low-water mark) and row lock promotion PCT specifies the maximum number of row locks permitted during a single scan session of a table or an index before Adaptive Server attempts to escalate from row locks to a table lock.

When the number of locks acquired during a scan session exceeds row lock promotion HWM, Adaptive Server attempts to acquire a table lock. The lock promotion HWM value cannot be higher than the number of locks value.

See Chapter 2, "Locking Configuration and Tuning," in *Performance and Tuning Series: Locking and Concurrency Control*.

The default value for row lock promotion HWM is appropriate for most applications. To avoid table locking, you may want to increase the value of row lock promotion HWM.. For example, if you know that there are regular updates to 500 rows on a table that has thousands of rows, you can increase concurrency for the tables by setting row lock promotion HWM to around 500.

You can also configure row lock promotion at the object level. See sp_setpglockpromote in the *Reference Manual: Procedures*.

row lock promotion LWM

Summary information	
Default value	200
Range of values	2–value of row lock promotion HWM
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

row lock promotion LWM (low-water mark), with the row lock promotion HWM (high-water mark) and row lock promotion PCT specifies the number of row locks permitted during a single scan session of a table or an index before Adaptive Server attempts to promote from row locks to a table lock.

row lock promotion LWM sets the number of locks below which Adaptive Server does not attempt to acquire a table lock on the object. The row lock promotion LWM must be less than or equal to row lock promotion HWM.

The default value for row lock promotion LWM is sufficient for most applications. If Adaptive Server runs out of locks (except for an isolated incident), increase number of locks.

See the *Performance and Tuning Series: Locking and Concurrency Control*.

You can also configure lock promotion at the object level. See sp_setpglockpromote in the *Reference Manual: Procedures*.

row lock promotion PCT

Summary information	
Default value	100
Range of values	1–100
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Lock Manager, SQL Server Administration

If the number of locks held on an object is between row lock promotion LWM (low-water mark) and row lock promotion HWM (high-water mark), row lock promotion PCT sets the percentage of row locks (based on the number of rows in the table) above which Adaptive Server attempts to acquire a table lock.

The default value for row lock promotion PCT is appropriate for most applications.

For more information on setting up lock promotion limits, see Chapter 2, "Locking Configuration and Tuning," in *Performance and Tuning Series: Locking and Concurrency Control*.

You can also configure row lock promotion at the per-object level. See sp_sterowlockpromote in the *Reference Manual: Procedures*.

rtm thread idle wait period

Summary information	
Default value	600
Range of values	600 – 4026531839
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	SQL Server Administration

rtm thread idle wait period defines the time, in seconds, 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.

runnable process search count

Summary information	
Default value	2000 (default value of 3 for the Cluster Edition)
Range of values	0 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

Note runnable process search count functions only when you configure Adaptive Server for process kernel mode; it is nonfunctional for threaded kernel mode. Use alter thread pool *pool_name* with idle timeout = *n* instead.

runnable process search count specifies the number of times an engine loops while looking for a runnable task before relinquishing the CPU to the operating system.

Adaptive Server engines check the run queue for runnable tasks whenever a task completes or exceeds its allotted time on the engine. At times, there are no tasks in the run queues. An engine can either relinquish the CPU to the operating system or continue to check for a task to run. Setting runnable process search count higher causes the engine to loop more times, thus holding the CPU for a longer time. Setting the runnable process search count lower causes the engine to release the CPU sooner.

If your machine is a uniprocessor that depends on helper threads to perform I/O, you may see some performance benefit from setting runnable process search to perform network I/O, disk I/O, or other operating system tasks. If a client, such as a bulk-copy operation, is running on the same machine as a single CPU server that uses helper threads, you may need to allow both the server and the client access to the CPU.

Note If you are having performance problems, try setting runnable process search count to 3.

For Adaptive Servers running on uniprocessor machines that do not use helper threads, and for multiprocessor machines, the default value should provide good performance.

With a runnable process search count value of 3, the Cluster Edition can better share the system CPU with other processes running on the same machine. However, if you set runnable process search count to 3 and Adaptive Server is running as a standalone process, users may experience delays in server response times. In this case, reset runnable process search count to 2000.

Use sp_sysmon to determine how the runnable process search count parameter affects the Adaptive Server use of CPU cycles, engine yields to the operating system, and blocking network checks. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon.*

runnable process search count versus idle timeout

runnable process search count and alter thread pool ...idle timeout both indicate how Adaptive Server looks for work:

- runnable process search count specifies the number of loops Adaptive Server spends looking for work, and is a server-wide parameter.

- alter thread pool ...idle timeout specifies the period of time Adaptive Server spends looking for work, and is tuned according to individual thread pools. idle timeout is more consistent across processors with varying speeds.

sampling percent

Summary information	
Default value	0
Range of values	0 – 100 percent
Status	Dynamic
Display level	Comprehensive
Required role	System or database administrator
Configuration group	Query Tuning

sampling percent is the numeric value of the sampling percentage, such as 5 for 5%, 10 for 10%, and so on.

To reduce I/O contention and resources, run update statistics using a sampling method, which can reduce the I/O and time when your maintenance window is small and the data set is large. If you are updating a large data set or table that is in constant use, being truncated and repopulated, you may want to perform a statistical sampling to reduce the time and the size of the I/O.

Use caution with sampling, since the results are not fully accurate. Balance changes to histogram values against the savings in I/O.

Although a sampling of the data set may not be completely accurate, usually the histograms and density values are reasonable within an acceptable range.

When you are deciding whether or not to use sampling, consider the size of the data set, the time constraints you are working with, and if the histogram produced is as accurate as needed.

The percentage to use when sampling depends on your needs. Test various percentages until you receive a result that reflects the most accurate information on a particular data set.

Statistics are stored in the system tables systabstats and sysstatistics.

secure default login

Summary information	
Default value	0
Range of values	0 (followed by another parameter naming the default login)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

secure default login specifies a default login for all users who are preauthenticated but who do not have a login in master..syslogins.

Establish the secure default login with:

```
sp_configure "secure default login", 0, default_login_name
```

where:

- secure default login – is the name of the parameter.

- 0 – is a required parameter because the second parameter of sp_configure must be a numeric value.

- *default_login_name* – is the name of the default login for a user who is unknown to Adaptive Server, but who has already been authenticated by a security mechanism. The login name must be a valid login in master..syslogins.

select for update

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Application Functionality

select for update enables Adaptive Server to exclusively lock rows for subsequent updates within the same transaction, and for updatable cursors, preventing other concurrent tasks from updating these rows and from blocking the subsequent update

See "Queries: Selecting Data from a Table" in the *Transact-SQL Users Guide*.

select on syscomments.text

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System security officer
Configuration group	Security Related

select on syscomments.text enables protection of the text of database objects through restriction of the select permission on the text column of the syscomments table. The default value sets select permission to "public." Set the value to 0 to restrict select permission to the object owner and the system administrator.

send doneinproc tokens

Summary information	
Default value	1 (on)
Range of values	0 (off), 1 (on)
Status	Dynamic

Summary information

Display level	Comprehensive
Required role	System administrator
Configuration group	Network Communication

send doneinproc tokens enables or disables Adaptive Server for sending doneinproc packets (TDS messages that are sent after each select statement performed in a stored procedure). send doneinproc tokens replaces dbcc tune 'doneinproc' and trace flag 292. Currently running queries immediately take note of any change in the option.

Setting send doneinproc tokens to 1 is safe in most cases. However some stored procedures are executed using asynchronous commands from CT-Lib, and using a value of 0 may cause state-machine errors in some CT-Lib applications.

session migration timeout

Summary information

Default value	600
Valid values	0 – 32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

session migration timeout specifies the amount of time available for a client to complete a migration by connecting to the target instance. If the client does not migrate to the target instance in the time alloted, Adaptive Server fails the connection.

session tempdb log cache size

Summary information

Default value	The logical page size, in bytes
Range of values	The logical page size up to 2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	User Environment

session tempdb log cache size specifies the size for each session tempdb log cache. Adaptive Server uses the tempdb log cache to buffer the user transaction log records for the session involving tempdb objects. The size for session tempdb log cache is determined by the Adaptive Server page size. Configuring a reasonable size for session tempdb log cache size can help prevent context switches due to a session having to flush the tempdb user log cache.

Adaptive Server includes two log caches for each user connection: the session tempdb log cache, and the user log cache, which is determined by the user log cache size parameter.

shared memory starting address

Summary information	
Default value	0
Range of values	Platform-specific
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	Physical Memory

shared memory starting address determines the virtual address where Adaptive Server starts its shared memory region.

It is unlikely that you will ever have to reconfigure shared memory starting address. Do so only after consulting with Sybase Technical Support.

number of worker processes, max parallel degree, and max scan parallel degree control parallel query processing at the server level. Using the parallel_degree, process_limit_action, and scan_parallel_degree options to the set command can limit parallel optimization at the session level, and using the parallel keyword of the select command can limit parallel optimization of specific queries.

size of auto identity column

Summary information	
Default value	10
Range of values	1–38
Status	Dynamic
Display level	Intermediate
Required role	System administrator

Summary information	
Configuration group	SQL Server Administration

size of auto identity column sets the precision of IDENTITY columns that are automatically created with the sp_dboption auto identity and unique auto_identity index options.

The maximum value that can be inserted into an IDENTITY column is $10^{precision}$ -1. After an IDENTITY column reaches its maximum value, all further insert statements return an error that aborts the current transaction.

If you reach the maximum value of an IDENTITY column, you can increase it with a modify operation in the alter table command. See the *Transact-SQL Users Guide* for examples.

You can also use the create table command to create a table that is identical to the old one, but with a larger precision for the IDENTITY column. After you have created the new table, use the insert command or bcp to copy data from the old table to the new one.

size of global fixed heap

Summary information	
Default values	150 pages (32-bit version)
	300 pages (64-bit version)
Minimum values	10 pages (32-bit version)
	20 pages (64-bit version)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Java Services, Memory Use

size of global fixed heap specifies the memory space for internal data structures and other needs.

If you change size of the global fixed heap, change the total logical memory by the same amount.

size of process object heap

Summary information	
Default values	1500 pages (32-bit version)
	3000 pages (64-bit version)
Minimum values	45 pages (32-bit version)
	90 pages (64-bit version)
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Java Services, Memory Use

size of process object fixed heap specifies the total memory space for all processes using the Java VM.

If you change size of process object fixed heap, change the total logical memory by the same amount.

size of shared class heap

Summary information	
Default values	1536 pages (32-bit version)
	3072 pages (64-bit version)
Minimum values	650 pages (32-bit version)
	1300 pages (64-bit version)
Status	Dynamic
Display level	Basic
Required role	System administrator
Configuration groups	Java Services, Memory Use

size of shared class heap specifies the shared memory space for all Java classes called into the Java VM. Adaptive Server maintains the shared class heap server-wide for both user-defined and system-provided Java classes.

If you change the size of shared class heap, change the total logical memory by the same amount.

size of unilib cache

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Unicode

size of unilib cache specifies the memory used in bytes rounded up to the nearest 1K in addition to the minimum overhead size, which provides enough memory to load a single copy of the largest Unilib conversion table plus the largest Unilib sort table. Asian clients may need to increase size of unilib cache by an extra 100K for every additional character set they want to support via Unicode-based conversion.

sproc optimize timeout limit

Summary information	
Default value	40
Range of values	0 – 4000
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

sproc optimize timeout limit specifies the amount of time Adaptive Server can spend optimizing a stored procedure as a fraction of the estimated execution time.

SQL batch capture

Summary information	
Default value	0 (off)
Range of values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator

Summary information	
Configuration group	Monitoring

SQL batch capture controls whether Adaptive Server collects SQL text. If both SQL batch capture and max SQL text monitored are enabled, Adaptive Server collects the SQL text for each batch for each user task.

SQL Perfmon Integration (Windows only)

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

SQL Perfmon Integration enables and disables the ability to monitor Adaptive Server statistics from the Windows Performance Monitor.

Adaptive Server must be registered as a Windows Service to support monitor integration. This occurs automatically when you start Adaptive Server using the Services Manager in the Sybase for the Windows program group, and is the default configuration when you use the Sybase installer or the syconfig utility to create an Adaptive Server.

The 15.7 version of sybperf exposes a set of Adaptive Server counters that are more useful for monitoring Adaptive Server performance.

sql server clock tick length

Summary information	
Default value	Platform-specific
Range of values	Platform-specific minimum–1000000, in multiples of default value
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

sql server clock tick length specifies the duration of the server's clock tick, in microseconds. Both the default value and the minimum value are platform-specific. Adaptive Server rounds values up to an even multiple of *n*, where *n* is the platform-specific clock-tick default value. Use sp_helpconfig or sp_configure to find the current values for sql server clock tick length.

In mixed-use applications with some CPU-bound tasks, decrease the value of sql server clock tick length to:

*   Help I/O-bound tasks – a value of 20,000 is reasonable for this. Shortening the clock-tick length means that CPU-bound tasks exceed the allotted time on the engine more frequently per unit of time, which allows other tasks greater access to the CPU

*   Marginally increase response times – Adaptive Server runs its service tasks once per clock tick. Decreasing the clock-tick length means that the service tasks are run more frequently per unit of time

Increasing sql server clock tick length favors CPU-bound tasks, because they execute longer between context switches. The maximum value of 1,000,000 may be appropriate for primarily CPU-bound applications. However, any I/O-bound tasks may suffer as a result. You can mitigate this somewhat by tuning cpu grace time (see "cpu grace time" on page 104) and time slice (see "time slice" on page 271).

Note Changing the value of sql server clock tick length can have serious effects on Adaptive Server performance. Consult with Sybase Technical Support before resetting this value.

sql text pipe active

Summary information	
Default value	0
Range of values	0–1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

sql text pipe active controls whether Adaptive Server collects SQL text. If this option is enabled and sql text pipe max messages is set, Adaptive Server collects the SQL text for each query. Use monSysSQLText to retrieve the SQL text for all user tasks.

sql text pipe max messages

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

sql text pipe max messages specifies the number of SQL text messages Adaptive Server stores per engine. The total number of messages in the monSQLText table is the value of sql text pipe max messages multiplied by the number of engines running.

stack guard size

Summary information	
Default value	4096
Range of values	0–2147483647
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, User Environment

stack guard size sets the size, in bytes, of the stack guard area, which is an overflow stack of configurable size at the end of each stack. Adaptive Server allocates one stack for each user connection and worker process when it starts. These stacks are located contiguously in the same area of memory, with a guard area at the end of each stack. At the end of each stack guard area is a guardword, which is a 4-byte structure with a known pattern. Figure 5-5 illustrates how a process can corrupt a stack guardword.

Figure 5-5: Process about to corrupt stack guardword
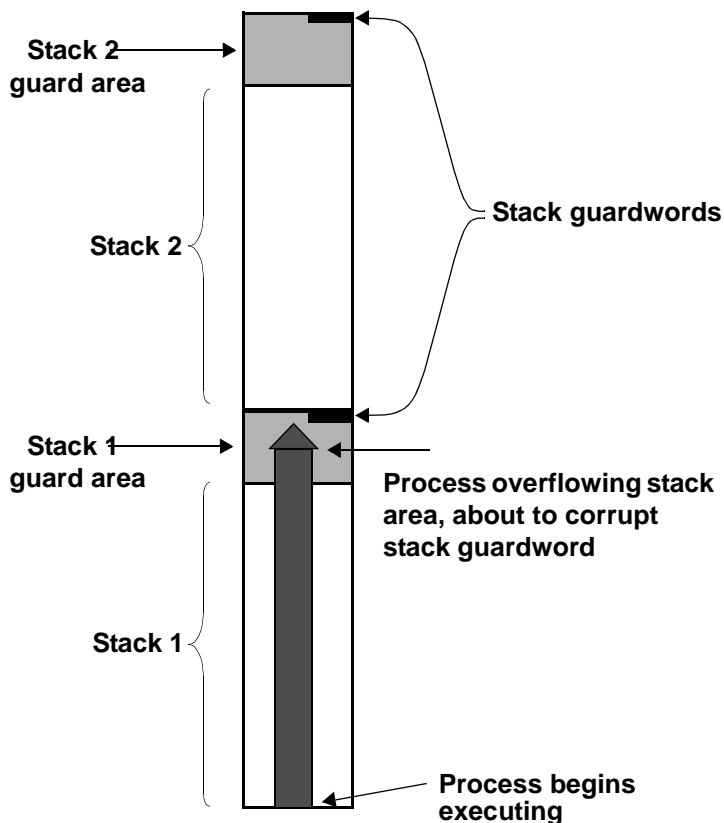
Adaptive Server periodically checks to see whether the stack pointer for a user connection has entered the stack guard area associated with that user connection's stack. If it has, Adaptive Server aborts the transaction, returns control to the application that generated the transaction, and generates error 3626:

```
The transaction was aborted because it used too much
stack space. Either use sp_configure to increase the
stack size, or break the query into smaller pieces.
spid: %d, suid: %d, hostname: %.*s, application name:
%.*s
```

Adaptive Server also periodically checks the guardword pattern to see if it has changed, thus indicating that a process has overflowed the stack boundary. When this occurs, Adaptive Server prints these messages to the error log and shuts down:

```
kernel: *** Stack overflow detected: limit: 0x%lx sp: 0x%lx
kernel: *** Stack Guardword corrupted
kernel: *** Stack corrupted, server aborting
```

In the first message, "limit" is the address of the end of the stack guard area, and "sp" is the current value of the stack pointer.

In addition, Adaptive Server periodically checks the stack pointer to see whether it is completely outside both the stack and the stack guard area for the pointer's process. If it is, Adaptive Server shuts down, even if the guardword is not corrupted. When this happens, Adaptive Server prints the following messages to the error log:

```
kernel: *** Stack overflow detected: limit: 0x%lx sp: 0x%lx
kernel: *** Stack corrupted, server aborting
```

The default value for stack guard size is appropriate for most applications. However, if you experience server shutdown from either stack guardword corruption or stack overflow, increase stack guard size by a 2K increment. Each configured user connection and worker process has a stack guard area; thus, when you increase stack guard size, you use up that amount of memory, multiplied by the number of user connections and worker processes you have configured.

Rather than increasing stack guard size to avoid stack overflow problems, consider increasing stack size (see "stack size" on page 260). The stack guard area is intended as an overflow area, not as an extension to the regular stack.

Adaptive Server allocates stack space for each task by adding the values of the stack size and stack guard size parameters. stack guard size must be configured in multiples of 2K. If the value you specify is not a multiple of 2K, sp_configure verification routines round the value up to the next highest multiple.

stack size

Summary information	
Default value	Platform-specific
Range of values	Platform-specific minimum–2147483647

Summary information	
Status	Static
Display level	Basic
Required role	System administrator
Configuration group	User Environment

stack size specifies the size, in bytes, of the execution stacks used by each user process on Adaptive Server. To find the stack size values for your platform, use sp_helpconfig or sp_configure. stack size must be configured in multiples of 2K. If the value you specify is not a multiple of 2K, sp_configure verification routines round the value up to the next highest multiple.

An execution stack is an area of Adaptive Server memory where user processes keep track of their process context and store local data.

Certain queries can contribute to the probability of a stack overflow. Examples include queries with extremely long where clauses, long select lists, deeply nested stored procedures, and multiple selects and updates that holdlock. When a stack overflow occurs, Adaptive Server prints an error message and rolls back the transaction. See "stack guard size" on page 258, and see the *Troubleshooting and Error Messages Guide* for more information on specific error messages.

The two options for remedying stack overflows are to break the large queries into smaller queries and to increase stack size. Changing stack size affects the amount of memory required for each configured user connection and worker process. See "total logical memory" on page 272.

If you have queries that exceed the size of the execution stack, you may want to rewrite them as a series of smaller queries, especially if there are only a small number of such queries, or if you run them infrequently.

There is no way to determine how much stack space a query requires without actually running the query. Stack space for each user connection and worker process is preallocated at start-up.

Therefore, determining the appropriate value for stack size is an empirical process. Test your largest and most complex queries using the default value for stack size. If they run without generating error messages, the default is probably sufficient. If they generate error messages, begin by increasing stack size by a small amount (2K). Re-run your queries and see if the amount you have added is sufficient. If it is not, continue to increase stack size until queries run without generating error messages.

If you are using CIS, or if Java is enabled in the database and you want to use methods that call JDBC, Sybase recommends that you increase the default by 50 percent. If you are not using JDBC or CIS, the standard default value is usually sufficient.

start mail session (Windows only)

Summary information	
Default value	0 (off)
Valid values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Extended Stored Procedure

start mail session enables and disables the automatic initiation of an Adaptive Server mail session when you start Adaptive Server.

A value of 1 configures Adaptive Server to start a mail session the next time Adaptive Server is started. A value of 0 configures Adaptive Server not to start a mail session at the next restart.

If start mail session is 0, you can start an Adaptive Server mail session explicitly, using the xp_startmail system ESP.

Before setting start mail session, you must prepare your Windows system by creating a mailbox and mail profile for Adaptive Server. Then, create an Adaptive Server account for Sybmail. See the *Configuration Guide for Windows*.

start xp server during boot

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Static
Display level	
Required role	
Configuration group	Extended Stored Procedures

start xp server during boot determines whether XP Server starts when Adaptive Server starts.

When set to 1, XP Server starts when Adaptive Server starts. If you set start xp server during boot to 0, XP Server does not start until you run xp_cmdshell.

startup delay

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Query Tuning

startup delay controls when RepAgent is started during the server start. By default, RepAgent starts at the same time as Adaptive Server. Adaptive Server writes a message to the error log stating the wait time.

statement cache size

Summary information	
Default value	0
Valid values	Size of cache in 2K pages
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, SQL Server Administration

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.

Note If you enable the statement cache, you must configure set chained on/off in its own batch.

Because cached statements are transformed into lightweight stored procedures, statement caching requires additional open object descriptors.

statement pipe active

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

statement pipe active controls whether Adaptive Server collects statement-level statistics. If both statement pipe active and statement pipe max messages are enabled, Adaptive Server collects the statement statistics for each query. Use monSysStatement to retrieve the statistics for all executed statements.

statement pipe max messages

Summary information	
Default value	0
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

statement pipe max messages determines the number of statement statistics messages Adaptive Server stores per engine. The total number of messages in the monSQLText table is the value of sql text pipe max messages multiplied by the number of engines running.

statement statistics active

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Monitoring

statement statistic active controls whether Adaptive Server collects monitoring table statement-level statistics. Use monProcessStatement to get statement statistics for a specific task.

streamlined dynamic SQL

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Application Functionality

quoted identifier enhancement enables the statement cache to store dynamic SQL statements

See "Memory Use and Performance" in the *Performance and Tuning Series: Basics*.

strict dtm enforcement

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Static
Display level	10
Required role	System administrator
Configuration group	DTM Administration

strict dtm enforcement determines whether or not Adaptive Server transaction coordination services strictly enforce the ACID properties of distributed transactions.

In environments where Adaptive Server should propagate and coordinate transactions only to other Adaptive Servers that support transaction coordination, set strict dtm enforcement on. If a transaction attempts to update data in a server that does not support transaction coordination services, Adaptive Server aborts the transaction.

In heterogeneous environments, you may want to make use of servers that do not support transaction coordination. This includes earlier versions of Adaptive Server and non-Sybase database stores configured using CIS. Under these circumstances, set strict dtm enforcement off to allow Adaptive Server to propagate transactions to legacy Adaptive Servers and other data stores. This does not, however, ensure that the remote work of these servers is rolled back or committed with the original transaction.

suppress js max task message

Summary information	
Default value	0 (off)
Valid values	0 (off), 1(on)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	Application Functionality

strict dtm enforcement prevents Adaptive Server from printing the Job Scheduler js maxtask error messages to the errorlog.

suspend audit when device full

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

suspend audit when device full determines what Adaptive Server does when an audit device becomes completely full.

Note If you have two or more audit tables, each on a separate device other than the master device, and you have a threshold procedure for each audit table segment, the audit devices should never become full. Only if a threshold procedure is not functioning properly does the "full" condition occur.

Choose one of these values:

- 0 – truncates the next audit table and starts using it as the current audit table when the current audit table becomes full. If you set suspend audit when device full to 0, you ensure that the audit process is never suspended. However, you incur the risk that older audit records are lost if they have not been archived.

- 1 – suspends the audit process and all user processes that cause an auditable event. To resume normal operation, the system security officer must log in and set up an empty table as the current audit table. During this period, the system security officer is exempt from normal auditing. If the system security officer's actions would generate audit records under normal operation, Adaptive Server sends an error message and information about the event to the error log.

syb_sendmsg port number

Summary information	
Default value	0
Valid values	0, or 1024–65535, or system limit
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Network Communication

syb_sendmsg port number specifies the port number that Adaptive Server uses to send messages to a UDP (User Datagram Protocol) port with sp_sendmsg or syb_sendmsg.

If more than one engine is configured, a port is used for each engine, numbered consecutively from the port number specified. If the port number is set to the default value, 0 Adaptive Server assigns port numbers.

Note Sending messages to UDP ports is not supported on Windows.

A system security officer must set the allow sendmsg configuration parameter to 1 to enable sending messages to UDP ports. To enable UDP messaging, a system administrator must set allow sendmsg to 1. See "allow sendmsg" on page 87. For more information on UDP messaging, see sp_sendmsg in the *Reference Manual: Procedures*.

sysstatistics flush interval

Summary information

Default value	0
Valid values	0 – 32767
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

sysstatiscts flush interval determines the length of the interval, in minutes, between flushes of sysstatistics.

Adaptive Server dynamically maintains the statistics for the number of rows and columns modified in a table as part of any DML statement and flushes them according to the value of sysstatistics flush interval.

Adaptive Server uses these statistics for query optimization since they are more accurate. The datachange function determines the amount of data that is changed at the table, column, or partition level since the last update statistics, and initiates updating statistics on the object.

The in-memory statistics are always flushed to disk during a polite shutdown of the server. You can configure sysstatistics flush interval to flush these in-memory statistics to disk by the house keeper task at regular intervals. Set sysstatistics flush interval to 0 to disable this housekeeper task.

systemwide password expiration

Summary information

Default value	0
Range of values	0–32767
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

systemwide password expiration sets the number of days that passwords remain in effect after they are changed. If systemwide password expiration is set to 0, passwords do not expire.

The password expires when the number of specified days passes. For example, if you create a new login on August 1, 2007 at 10:30 a.m., with a password expiration interval of 30 days, the password expires on August 31, 2007 at 10:30 a.m.

An account's password is considered expired if an interval greater than *number_of_days* has passed since the last time the password for that account was changed.

When the number of days remaining before expiration is less than 25 percent of the value of systemwide password expiration or 7 days, whichever is greater, each time the user logs in, a message displays, giving the number of days remaining before expiration. Users can change their passwords anytime before expiration.

When an account's password has expired, the user can still log in to Adaptive Server but cannot execute any commands until he or she has used sp_password to change his or her password. If the system security officer changes the user's password while the account is in sp_password-only mode, the account returns to normal after the new password is assigned.

This restriction applies only to login sessions established after the password has expired. Users who are logged in when their passwords expire are not affected until the next time they log in.

tape retention in days

Summary information	
Default value	0
Range of values	0–365
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	Backup/Recovery

tape retention in days specifies the number of days you intend to retain each tape after it has been used for either a database or a transaction log dump. This parameter can keep you from accidentally overwriting a dump tape.

For example, if you have set tape retention in days to 7 days, and you attempt to use the tape before 7 days have elapsed since the last time you dumped to that tape, Backup Server issues a warning message.

You can override the warning using the with init option when executing the dump command. Doing this causes the tape to be overwritten and all data on the tape to be lost.

Both the dump database and dump transaction commands provide a retaindays option, which overrides the tape retention in days value for a particular dump. See Chapter 12, "Backing Up and Restoring User Databases," in *System Administration Guide: Volume 2*.

tcp no delay

Summary information	
Default value	1 (on)
Valid values	0 (off), 1 (on)
Status	Static
Display level	Comprehensive
Required role	System administrator
Configuration groups	Network Communication, O/S Resources

tcp no delay controls TCP (Transmission Control Protocol) packet batching. The default value means that TCP packets are not batched.

TCP normally batches small logical packets into single, larger, physical packets, and fills physical network frames with as much data as possible, which improves network throughput in terminal emulation environments where users mostly send keystrokes across the network.

However, applications that use small TDS (Tabular Data Stream) packets may benefit from disabling TCP packet batching.

Note Disabling TCP packet batching means that packets are sent, regardless of size; this increases the volume of network traffic.

text prefetch size

Summary information	
Default value	16
Valid values	0–65535
Status	Dynamic
Display level	Comprehensive

Summary information	
Required role	System administrator
Configuration Group	Network Communications

text prefetch size limits the number of pages of text, unitext, and image data that can be prefetched into an existing buffer pool. Adaptive Server prefetches only text, unitext, and image data that was created with Adaptive Server 12.x or was upgraded using dbcc rebuild_text.

time slice

Summary information	
Default value	100
Range of values	50–1000
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

time slice sets the number of milliseconds that the Adaptive Server scheduler allows a task to run. If time slice is set too low, Adaptive Server may spend too much time switching between tasks, which increases response time. If it is set too high, CPU-intensive tasks may monopolize engines, which also increases response time. The default value allows each task to run for 1/10 of a second before relinquishing the CPU to another task.

See Chapter 3, "Using Engines and CPUs" in the *Performance and Tuning Series: Basics*.

Use sp_sysmon to determine how time slice affects voluntary yields by Adaptive Server engines. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

total data cache size

Summary information	
Default value	0
Range of values	0 – 2147483647
Status	Calculated
Display level	Basic

Summary information	
Required role	System administrator
Configuration groups	Cache Manager, Memory Use

total data cache size reports the amount of memory, in kilobytes, that is currently available for data, index, and log pages. This parameter is a calculated value that is not directly user-configurable.

The amount of memory available for the data cache can be affected by a number of factors, including:

- The amount of physical memory available on your machine

- The values to which the following parameters are set:

  - total logical memory

  - number of user connections

  - total procedure cache percent

  - number of open databases

  - number of open objects

  - number of open indexes

  - number of devices

A number of other parameters also affect the amount of available memory, but to a lesser extent.

For information on how Adaptive Server allocates memory and for information on data caches, see "Configuration parameters" on page 80.

total logical memory

Summary information	
Default value	N/A
Range of values	N/A
Status	Read-only
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, Physical Memory

total logical memory displays the total logical memory for the current configuration of Adaptive Server. The total logical memory is the amount of memory that the Adaptive Server current configuration uses. total logical memory displays the memory that is required to be available, but which may or may not be in use at any given moment. For information about the amount of memory in use at a given moment, see total physical memory. You cannot use total logical memory to set any of the memory configuration parameters.

total physical memory

Summary information	
Default value	N/A
Range of values	N/A
Status	Read-only
Display level	Intermediate
Required role	System administrator
Configuration group	Memory Use

total physical memory is a read-only configuration parameter that displays the total physical memory for the current configuration of Adaptive Server. The total physical memory is the amount of memory that Adaptive Server is using at a given moment in time. Configure Adaptive Server so that the value for max memory is larger than the value for total logical memory, and the value for total logical memory is larger than the value for total physical memory.

transfer utility memory size

Summary information	
Default value	4096
Range of values	0 – 2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration group	SQL Server Administration

Adaptive Server maintains a memory pool for the transfer table command and for tables marked for incremental transfer. This pool provides memory for maintaining state information about current and past transfers, and for memory used to write to and read from transfer files. transfer utility memory size determines the size of this memory pool.

The units for this pool are in memory pages, which are blocks of 2048 bytes. The default size is large enough to accommodate over 100 tables marked for incremental transfer, all transferring simultaneously.

If your installation does not include tables marked for incremental transfer, and does not use the transfer table command, you may set the size of this memory pool to zero to reclaim this memory.

txn to pss ratio

Summary information	
Default value	16
Valid values	1 – 2147483647
Status	Static
Display level	1
Required role	System administrator
Configuration groups	DTM Administration, Memory Use

Adaptive Server manages transactions as configurable server resources. Each time a new transaction begins, Adaptive Server must obtain a free **transaction descriptor** from a global pool that is created when the server is started. Transaction descriptors are internal memory structures that Adaptive Server uses to represent active transactions.

Adaptive Server requires one free transaction descriptor for:

- The outer block of each server transaction. The outer block of a transaction may be created explicitly when a client executes a new begin transaction command. Adaptive Server may also implicitly create an outer transaction block when clients use Transact-SQL to modify data without using begin transaction to define the transaction.

  Note Subsequent, nested transaction blocks, created with additional begin transaction commands, do not require additional transaction descriptors.

- Each database accessed in a **multidatabase transaction**. Adaptive Server must obtain a new transaction descriptor each time a transaction uses or modifies data in a new database.

txn to pss ratio determines the total number of transaction descriptors available to the server. At start-up, this ratio is multiplied by the number of PSS structures to create the transaction descriptor pool:

```
# of transaction descriptors = PSS structures * txn to pss ratio
```

The default value, 16, ensures compatibility with versions of Adaptive Server earlier than 12.x which also allocated 16 transaction descriptors for each user connection. In version 12.x and later, the number of simultaneous transactions is limited only by the number of transaction descriptors available in the server.

Note You can have as many databases in a user transaction as there are in your Adaptive Server installation. For example, if your Adaptive Server has 25 databases, you can include 25 databases in your user transactions.

Optimizing the txn to pss ratio for your system

During a peak period, use sp_monitorconfig to examine the use of transaction descriptors:

```
                sp_monitorconfig "txn to pss ratio"
Usage information at date and time: Apr 22 2002  2:49PM.
Name                      Num_free    Num_active  Pct_act  Max_Used
Reuse_cnt    Instance_Name
------------------------  ----------  ----------  -------  -----------
----------  -------------------
txn to pss ratio               784          80    10.20          523
         0                NULL
```

If the num_used value is zero or very low, transactions may be delayed as Adaptive Server waits for transaction descriptors to become free in the server. In this case, consider increasing the value of txn to pss ratio.

If the Max_Used value is too low, unused transaction descriptors may be consuming memory that can be used by other server functions. Consider reducing the value of txn to pss ratio.

unified login required

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Dynamic
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

unified login required requires that all users who log in to Adaptive Server be authenticated by a security mechanism. The use security services parameter must be 1 to use the unified login security service.

upgrade version

Summary information	
Default value	1100
Range of values	0–2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	SQL Server Administration

upgrade version reports the version of the upgrade utility that upgraded your master device. The upgrade utility checks and modifies this parameter during an upgrade.

Warning! Do not reset upgrade version. Doing so may cause serious problems with Adaptive Server.

You can determine whether an upgrade has been done on your master device by using upgrade version without specifying a value:

```
sp_configure "upgrade version"
```

use security services

Summary information	
Default value	0 (off)
Range of values	0 (off), 1 (on)
Status	Static
Display level	Intermediate
Required role	System security officer
Configuration group	Security Related

use security services specifies that Adaptive Server uses network-based security services. If the parameter is set to 0, none of the network-based security services can be used.

user log cache size

Summary information	
Default value	Logical page size
Range of values	2048^a –2147483647 a. Minimum determined by server's logical page size
Status	Static
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, User Environment

user log cache size specifies the size, in bytes, for each user's log cache. Its size is determined by the server's logical page size. There is one user log cache for each configured user connection and worker process. Adaptive Server uses these caches to buffer the user transaction log records, which reduces the contention at the end of the transaction log.

When a user log cache becomes full or another event occurs (such as when the transaction completes), Adaptive Server "flushes" all log records from the user log cache to the database transaction log. By first consolidating the log records in each user's log cache, rather than immediately adding each record to the database's transaction log, Adaptive Server reduces contention of processes writing to the log, especially for SMP systems that are configured with more than one engine.

Note For transactions using a database with mixed data and log segments, the user log cache is flushed to the transaction log after each log record. No buffering takes place. If your databases do not have dedicated log segments, do not increase the user log cache size.

Do not configure user log cache size to be larger than the maximum amount of log information written by an application's transaction. Since Adaptive Server flushes the user log cache when the transaction completes, any additional memory allocated to the user log cache is wasted. If no transaction in your server generates more than 4000 bytes of transaction log records, set user log cache size no higher than that value. For example:

```
sp_configure "user log cache size", 4000
```

Setting user log cache size too high wastes memory. Setting it too low can cause the user log cache to fill up and flush more than once per transaction, increasing the contention for the transaction log. If the volume of transactions is low, the amount of contention for the transaction log may not be significant.

Use sp_sysmon to understand how this parameter affects cache behavior. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

user log cache spinlock ratio

Summary information	
Default value	20
Range of values	1–2147483647
Status	Dynamic
Display level	Intermediate
Required role	System administrator
Configuration groups	Memory Use, User Environment

For Adaptive Servers running with multiple engines, user log cache spinlock ratio specifies the ratio of user log caches per user log cache **spinlock**. There is one user log cache for each configured user connection.

The default specifies 1 spinlock for each 20 user connections configured for your server.

Use sp_sysmon to understand how this parameter affects cache behavior. See the *Performance and Tuning Series: Monitoring Adaptive Server with sp_sysmon*.

wait event timing

Summary information	
Default value	0
Range of values	0–1
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration groups	Memory Use, Monitoring

wait event timing controls whether Adaptive Server collects statistics for individual wait events. A task may have to wait for a variety of reasons (for example, waiting for a buffer read to complete). The monSysWaits table contains the statistics for each wait event. The monWaitEventInfo table contains a complete list of wait events.

workload manager cache size

Summary information	
Default value	80
Valid values	80 – 2147483647
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Shared Disk Cluster

workload manager cache size specifies the maximum amount of memory, in 2K pages, that the workload manager can use. See Chapter 6, "Managing the Workload," in the *Cluster Users Guide*.

xact coordination interval

Summary information	
Default value	60 (seconds)
Valid values	1 – 2147483647 (seconds)
Status	Dynamic
Display level	10
Required role	System administrator
Configuration group	DTM Administration

xact coordination interval defines the length of time between attempts to resolve transaction branches have been propagated to remote servers.

The coordinating Adaptive Server makes regular attempts to resolve the work of remote servers participating in a distributed transaction. The coordinating server contacts each remote server participating in the distributed transaction in a serial manner, as shown in Figure 5-6. The coordination service may be unable to resolve a transaction branch for a variety of reasons. For example, if the remote server is not reachable due to network problems, the coordinating server reattempts the connection after the time specified by xact coordination level.

Figure 5-6: Resolving remote transaction branches
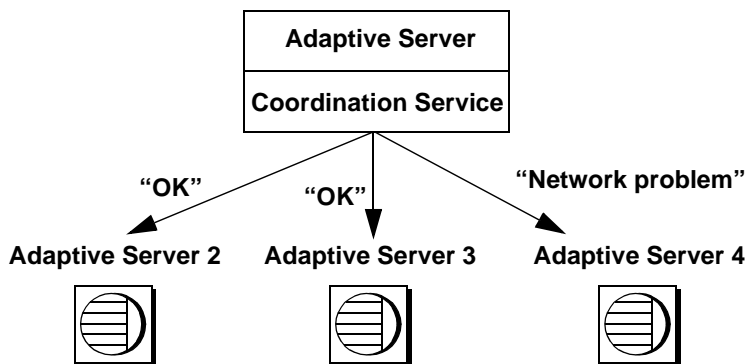
With the default value of xact coordination interval, 60, Adaptive Server attempts to resolve remote transactions once every minute. Decreasing the value may speed the completion of distributed transactions, but only if the transactions are themselves resolved in less than a minute. Under normal circumstances, there is no performance penalty to decreasing the value of xact coordination interval.

Setting xact coordination interval to a higher number can slow the completion of distributed transactions, and cause transaction branches to hold resources longer than they normally would. Under normal circumstances, do not increase the value of xact coordination interval beyond its default.

xp_cmdshell context

Summary information	
Default value	1
Valid values	0, 1, 2
Status	Dynamic
Display level	Comprehensive
Required role	System administrator
Configuration group	Extended Stored Procedure

xp_cmdshell context sets the security context for the operating system command to be executed using the xp_cmdshell system ESP. The values for the context determines under which account the command runs:

• 0 – command runs under XP Server's account.

• 1 – command runs under user's account.

• 2 – command runs under XP Server's account only if the user has administrator privileges.

Setting xp_cmdshell context to 1 restricts the xp_cmdshell security context to users who have accounts at the operating system level. Its behavior is platform-specific. If xp_cmdshell context is set to 1, to use an xp_cmdshell ESP, an operating system user account must exist for the Adaptive Server user name. For example, an Adaptive Server user named "sa" cannot use xp_cmdshell unless he or she has an operating-system-level user account named "sa".

Starting XP Server as root automatically sets xp_cmdshell to 1 so secure access is automatically enabled.

On Windows, when xp_cmdshell context is set to 1, xp_cmdshell succeeds only if the user name of the user logging in to Adaptive Server is a valid Windows user name with Windows system administration privileges on the system on which Adaptive Server is running.

On other platforms, when xp_cmdshell context is set to 1, xp_cmdshell succeeds only if Adaptive Server was started by a user with "superuser" privileges at the operating system level. When Adaptive Server gets a request to execute xp_cmdshell, it checks the uid of the user name of the ESP requestor and runs the operating system command with the permissions of that uid.

If xp_cmdshell context is 0, the permissions of the operating system account under which Adaptive Server is running are the permissions used to execute an operating system command from xp_cmdshell. This allows users to execute operating commands that they would not ordinarily be able to execute under the security context of their own operating system accounts.

Overview of Disk Resource Issues

Topic	Page
Device allocation and object placement	283
Commands for managing disk resources	284
Considerations in storage management decisions	286
Status and defaults at installation time	287
System tables that manage storage	288

Many Adaptive Server defaults are set to reasonable values for aspects of storage management, such as database, table, and index location, and how much space is allocated for each one. Responsibility for storage allocation and management is often centralized, and usually, the system administrator has ultimate control over the allocation of disk resources to Adaptive Server and the physical placement of databases, tables, and indexes on those resources.

Device allocation and object placement

When configuring a new system, the system administrator must consider several issues that have a direct impact on the number and size of disk resources required. These device allocation issues refer to commands and procedures that add disk resources to Adaptive Server.

Table 6-1: Device allocation topics

Task	Chapter
Initialize and allocate a default pool of database devices	Chapter 8, "Initializing Database Devices"
Mirror database devices for recovery	Chapter 2, "Mirroring Database Devices," in *System Administration Guide: Volume 2*

After the initial disk resources have been allocated to Adaptive Server, the system administrator, database owner, and object owners should consider how to place databases and database objects on specific database devices. These object placement issues determine where database objects reside on your system and whether or not the objects share devices. Object placement tasks are discussed throughout this manual, including the chapters shown in Table 6-2.

Table 6-2: Object placement topics

Task	Chapter
Place databases on specific database devices	Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2*
Place tables and indexes on specific database devices	Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2*

Do not consider allocating devices separately from object placement. For example, if you decide that a particular table must reside on a dedicated pair of devices, first allocate those devices to Adaptive Server. The remaining sections in this chapter provide an overview that spans both device allocation and object placement issues, providing pointers to chapters where appropriate.

Commands for managing disk resources

Table 6-3 lists the major commands a system administrator uses to allocate disk resources to Adaptive Server, and provides references to the chapters that discuss those commands.

Table 6-3: Commands for allocating disk resources

Command	Task	See
disk init name = "*dev_name*" physname = "*phys_name*"...	Makes a physical device available to a particular Adaptive Server. Assigns a database device name (*dev_name*) that is used to identify the device in other Adaptive Server commands.	Chapter 8, "Initializing Database Devices"
sp_deviceattr *logicalname*, *optname*, *optvalue*	Changes the *dsync* setting of an existing database device file.	Chapter 8, "Initializing Database Devices"
sp_diskdefault "*dev_name*"...	Adds *dev_name* to the general pool of default database space.	Chapter 8, "Initializing Database Devices"
disk resize name = "*device_name*", size = *additional_space*	Dynamically increases the size of database devices.	Chapter 8, "Initializing Database Devices"

Command	Task	See
disk mirror name = "*dev_name*" mirror = "*phys_name*"...	Mirrors a database device on a specific physical device.	Chapter 2, "Mirroring Database Devices," in *System Administration Guide: Volume 2*

Table 6-4 lists the commands used in object placement. For information about how object placement affects performance, see Chapter 1, "Controlling Physical Data Placement," in the *Performance and Tuning Series: Physical Database Tuning*.

Table 6-4: Commands for placing objects on disk resources

Command	Task	See
create database...on *dev_name* or alter database...on *dev_name*	Makes database devices available to a particular Adaptive Server database. The log on clause to create database places the database's logs on a particular database device.	Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2*
create database... or alter database...	When used without the on *dev_name* clause, these commands allocate space on the default database devices.	Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2*
sp_addsegment *seg_name*, *dbname*, *devname* and sp_extendsegment *seg_name*, *dbname*, *devname*	Creates a segment—a named collection of space—from the devices available to a particular database.	Chapter 8, "Creating and Using Segments in *System Administration Guide: Volume 2*
create table...on *seg_name* or create index...on *seg_name*	Creates database objects, placing them on a specific segment of the database's assigned disk space.	Chapter 8, "Creating and Using Segments in *System Administration Guide: Volume 2*
create table... or create index...	When used without on *seg_name*, tables and indexes occupy the general pool of space allocated to the database (the default devices).	Chapter 8, "Creating and Using Segments in *System Administration Guide: Volume 2*

Considerations in storage management decisions

The system administrator must make many decisions regarding the physical allocation of space to Adaptive Server databases. The major considerations in these choices are:

- Recovery – disk mirroring and maintaining logs on a separate physical device provide two mechanisms for full recovery in the event of physical disk failures.

- Performance – for tables or databases where speed of disk reads and writes is crucial, properly placing database objects on physical devices yields performance improvements. Disk mirroring slows the speed of disk writes.

Recovery

Recovery is the key motivation for using several disk devices. You can mirror database devices to achieve nonstop recovery. You can also ensure full recovery by storing a database's log on a separate physical device.

Keeping logs on a separate device

Unless a database device is mirrored, full recovery requires that a database's transaction log be stored on a different device from the actual data (including indexes) of a database. In the event of a hard disk failure, you can create an up-to-date database by loading a dump of the database and then applying the log records that were safely stored on another device. See Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2* for information about the log on clause of create database.

Mirroring

Nonstop recovery in the event of a hard disk failure is guaranteed by mirroring all Adaptive Server devices to a separate physical disk. See Chapter 2, "Mirroring Database Devices," in *System Administration Guide: Volume 2*.

Performance

You can improve system performance by placing logs and database objects on separate devices:

- Placing a table on one hard disk and nonclustered indexes on another ensures that physical reads and writes are faster, since the work is split between two disk drives.

- Splitting large tables across two disks can improve performance, particularly for multiuser applications.

- When log and data share devices, user log cache buffering of transaction log records is disabled.

- Partitioning provides multiple insertion points for a heap table, adds a degree of parallelism to systems configured to perform parallel query processing, and makes it possible to distribute a table's I/O across multiple database devices.

See Chapter 1, "Controlling Physical Data Placement," in the *Performance and Tuning Series: Physical Database Tuning* for a detailed discussion of how object placement affects performance.

Status and defaults at installation time

The installation program and scripts initialize the master device and set up the master, model, sybsystemprocs, sybsecurity, and temporary databases for you.

When you install Adaptive Server, the system databases, system-defined segments, and database devices are organized as follows:

- The master, model, and tempdb databases are installed on the master device.

- The sybsystemprocs database is installed on a device that you specified.

- Three segments are created in each database: system, default, and logsegment.

- The master device is the default storage device for all user-created databases.

    Note After initializing new devices for default storage, remove the master device from the default storage area with sp_diskdefault. Do not store user databases and objects on the master device. See "Designating default devices" on page 316.

- If you install the audit database, sybsecurity, it is located on its own device.

System tables that manage storage

Two system tables in the master database, sysusages and sysdevices, and three more in each user database (syssegments, sysindexes, and syspartitions) track the placement of databases, tables (including the transaction log table, syslogs), and indexes. The relationship between the tables is illustrated in Figure 6-1.

Figure 6-1: System tables that manage storage
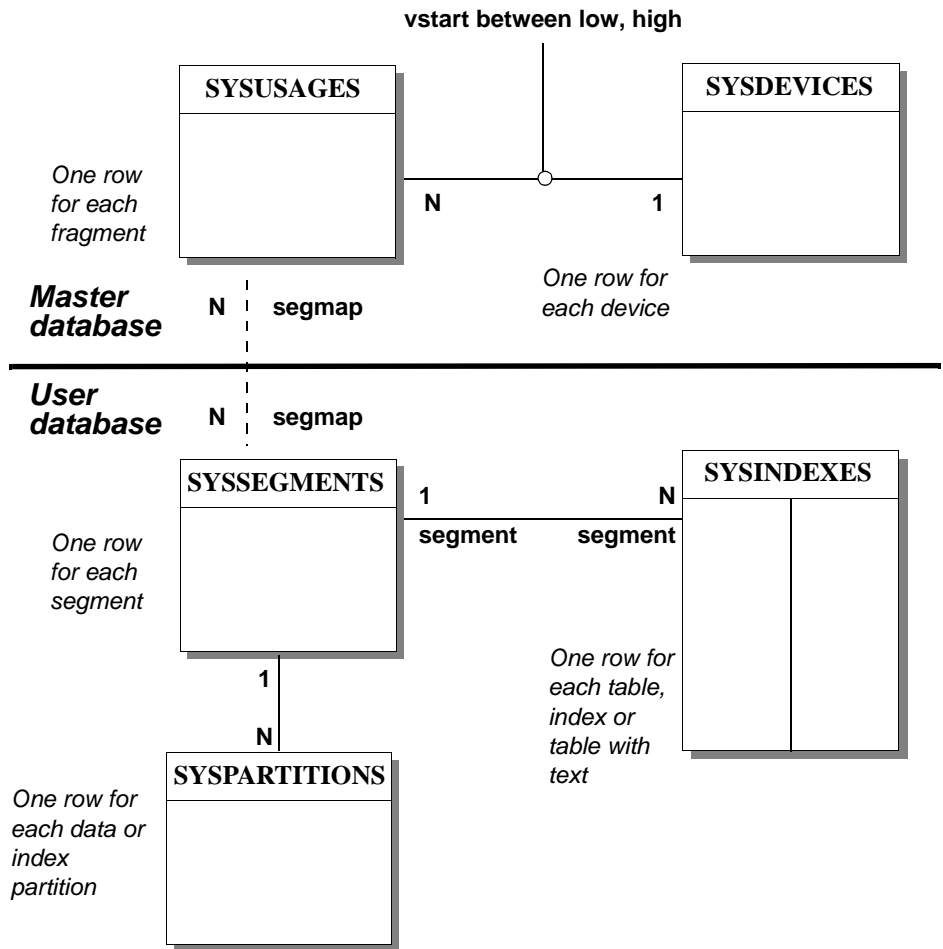
The *sysdevices* table

The sysdevices table in the master database contains one row for each **database device** and may contain a row for each dump device (tape, disk, or operating system file) available to Adaptive Server.

The disk init command adds entries for database devices to master..sysdevices. Dump devices, added using sp_addumpdevice, are discussed in Chapter 11, "Developing a Backup and Recovery Plan," in the *System Administration Guide: Volume 2*

sysdevices stores two names for each device:

• A **logical name** or **device name**, used in all subsequent storage-management commands, is stored in the name column of sysdevices. This is usually a user-friendly name, perhaps indicating the planned use for the device, for example, "logdev" or "userdbdev."

• The **physical name** is the actual operating system name of the device. Use this name only in the disk init command; after that, all Adaptive Server data storage commands use the logical name.

Place a database or transaction log on one or more devices by specifying the logical name of the device in the create database or alter database statement. The log on clause to create database places a database's transaction log on a separate device to ensure full recoverability. The log device must also have an entry in sysdevices before you can use log on.

A database can reside on one or more devices, and a device can store one or more databases. See Chapter 6, "Creating and Managing User Databases," in *System Administration Guide: Volume 2* for information about creating databases on specific database devices.

The *sysusages* table

The sysusages table in the master database keeps track of the space you assign to all Adaptive Server databases.

create database and alter database allocate new space to the database by adding a row to sysusages for each database device or device fragment. When you allocate only a portion of the space on a device with create or alter database, that portion is called a **fragment**.

sp_addsegment, sp_dropsegment, and sp_extendsegment change the segmap column in sysusages for the device that is mapped or unmapped to a segment. See Chapter 8, "Creating and Using Segments in *System Administration Guide: Volume 2*.

The *syssegments* table

The syssegments table, one in each database, lists the segments in a database. A **segment** is a collection of the database devices and fragments available to a particular database. Tables and indexes can be assigned to a particular segment—and therefore to a particular physical device—or can span a set of physical devices.

create database makes default entries in syssegments. sp_addsegment and sp_dropsegment to add and remove entries from syssegments.

The *sysindexes* table

The sysindexes table lists each table and index and the segment where each table, clustered index, nonclustered index, and chain of text pages is stored. It also lists other information such as the max_rows_per_page setting for the table or index.

The create table, create index, and alter table commands create new rows in sysindexes. Partitioning a table changes the function of sysindexes entries for the table.

The syspartitions table

The syspartitions table lists each table and index partition and the segment where the partition is stored. syspartitions maintains key storage management information such as the first page of a data or index page chain, the last page of a heap, the root page of an index partition, and so on.

Use create table, create index and alter table to create new rows in syspartitions.

Managing Remote Servers

This chapter discusses the steps the system administrator and system security officer of each Adaptive Server must execute to enable **remote procedure calls** (RPCs).

Overview

Users on a local Adaptive Server can execute stored procedures on a remote Adaptive Server. Executing an RPC sends the results of the remote process to the calling process, which usually appears on the user's screen.

To enable RPCs, the system administrator and system security officer of each Adaptive Server must execute the following steps:

- On the local server:

    - System security officer – use sp_addserver to list the local server and remote server in the system table master..sysservers.

    - List the remote server in the *interfaces* file or directory service for the local server.

    - Restart the local server so the global variable @@*servername* is set to the name of the local server. If this variable is not set properly, users cannot execute RPCs from the local server on any remote server.

- On the remote server:

- System security officer – use sp_addserver to list the server originating the RPC in the system table master..sysservers.

- To allow the user who is originating the remote procedure access to the server, a system security officer uses create login, and a system administrator uses sp_addremotelogin.

- Add the remote login name as a user of the appropriate database and grant that login permission to execute the procedure. (If execute permission is granted to "public," the user does not need to be granted specific permission.)

Figure 7-1 shows how to set up servers for remote access.

Figure 7-1: Setting up servers to allow remote procedure calls

The user "joe" on ROSE needs to access stored procedures on ZINNIA

ROSE   **ZINNIA**

sp_addserver ROSE, local
sp_addserver ZINNIA

interfaces files must have an entry for ZINNIA

sp_addserver ROSE
create login joe
sp_addremotelogin ROSE, joe

sp_adduser joe (in the appropriate database)
grant execute on *procedure_name* to joe
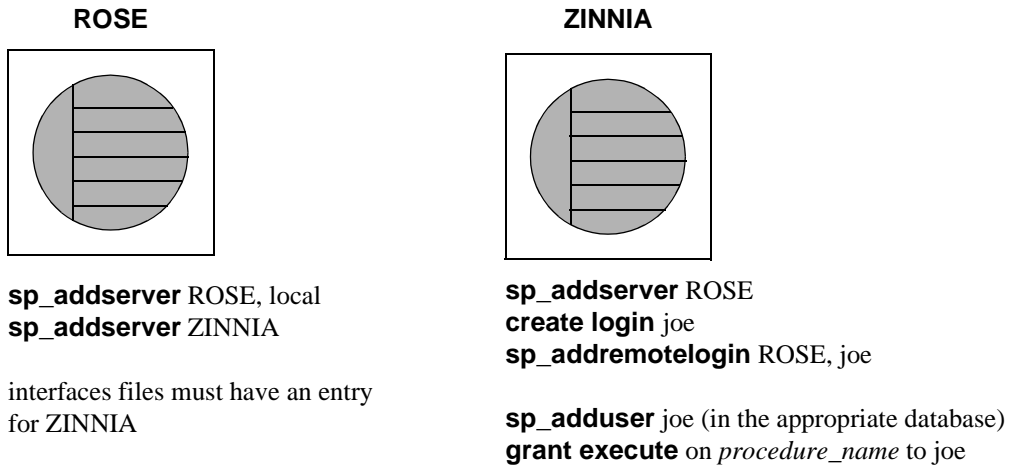
For operating-system-specific information about handling remote servers, see the installation documentation for your platform.

Managing remote servers

Table 7-1 lists the tasks related to managing remote servers, and the system procedures you use to perform the tasks.

Table 7-1: Tasks related to managing remote servers

To	Use	See
Add a remote server	sp_addserver	"Adding a remote server" on page 295
Manage remote server names	sp_addserver	"Managing remote server names" on page 296
Change server connection options	sp_serveroption	"Setting server connection options" on page 297
Display information about servers	sp_helpserver	"Getting information about servers" on page 299
Drop a server	sp_dropserver	"Dropping remote servers" on page 299

Adding a remote server

A system security officer uses sp_addserver to add entries to the sysservers table. On the server originating the call, you must add one entry for the local server, and one for each remote server that your server will call.

When you create entries for a remote server, you can either:

- Refer to them by the name listed in the *interfaces* file, or

- Provide a local name for the remote server. For example, if the name in the *interfaces* file is "MAIN_PRODUCTION," you may want to call it simply "main."

The syntax is:

```
sp_addserver lname [{, local | null}
    [, pname]]
```

where:

- *lname* – provides the local "call name" for the remote server. If this name is not the same as the remote server's name in the *interfaces* file, provide that name as the third parameter, *pname*.

  The remote server must be listed in the *interfaces* file on the local machine. If it is not listed, copy the *interfaces* file entry from the remote server and append it to your existing *interfaces* file. Keep the same port numbers.

- local – identifies the server being added as a local server. The local value is used only after starting up, or after a restart, to identify the local server name so that it can appear in messages printed out by Adaptive Server. null specifies that this server is a remote server.

  Note For users to successfully run RPCs from the local server, add the local server using the local option, and restart it. The restarting is required to set the global variable @@*servername*.

- *pname* – is the remote server listed in the *interfaces* file for the server named *lname*. This optional argument permits you to establish local aliases for any other Adaptive Server, Open Server, or Backup Server that you may need to communicate with. If you do not specify *pname*, to *lname* is the default.

Examples of adding remote servers

This example creates an entry for the local server named DOCS:

```
sp_addserver DOCS, local
```

This example creates an entry for a remote server named GATEWAY:

```
sp_addserver GATEWAY
```

To run a remote procedure such as sp_who on the GATEWAY server, execute either:

```
GATEWAY.sybsytemprocs.dbo.sp_who
```

or:

```
GATEWAY...sp_who
```

This example gives a remote server called MAIN_PRODUCTION the local alias "main:"

```
sp_addserver main, null, MAIN_PRODUCTION
```

The user can then enter:

```
main...sp_who
```

Managing remote server names

The master.dbo.sysservers table has two server name columns:

- srvname is the unique server name that users must supply when executing remote procedure calls.

- srvnetname is the server's network name, which must match the name in the *interfaces* file.

To add or drop servers from your network, use sp_addserver to update the server's network name in srvnetname.

For example, to remove the server MAIN from the network, and move your remote applications to a server named TEMP, use the following statement to change the network name, while keeping the local alias:

```
sp_addserver MAIN, null, TEMP
```

sp_addserver displays a message telling you that it is changing the network name of an existing server entry.

Setting server connection options

sp_serveroption sets the server options timeouts, net password encryption, and the security mechanism, which affect connections with remote servers.

The options you specify for sp_serveroption do not affect communication between Adaptive Server and Backup Server.

The following sections describe timeouts, net password encryption.

Using the *timeouts* option

A system administrator can use the timeouts option to disable and enable the normal timeout code used by the local server.

By default, timeouts is set to true, and the site handler process that manages remote logins times out if there has been no remote user activity for one minute. By setting timeouts to false on both of the servers involved in remote procedure calls, the automatic timeout is disabled. to change timeouts to false use:

```
sp_serveroption GATEWAY, "timeouts", false
```

After you set timeouts to false on both servers, when a user executes an RPC in either direction, the site handler on each machine runs until one of the servers is shut down. When the server is brought up again, the option remains false, and the site handler is reestablished the next time a user executes an RPC. If users execute RPCs frequently, it is probably efficient in terms of system resources to set this option to false, since there is some system overhead involved in setting up the physical connection.

Using the *net password encryption* option

A system security officer can use net password encryption to specify whether connections with a remote server are to be initiated with a client-side password encryption handshake or with the usual unencrypted password handshake sequence. The default is false.

If net password encryption is set to true:

1 The initial login packet is sent without passwords.

2 The client indicates to the remote server that encryption is desired.

3 The remote server returns an encryption key, which the client uses to encrypt its plain text passwords.

4 The client then encrypts its own passwords, and the remote server uses the key to authenticate them when they arrive.

This example sets net password encription to true:

```
sp_serveroption GATEWAY, "net password encryption",
    true
```

Using the *security mechanism* option

The security mechanism option enables Kerberos authentication for connections to the remote server.

Note As of Adaptive Server release 15.7, Kerberos is the security mechanism supported by Adaptive Server. The rpc security model A and rpc security model B are no longer required for enabling security mechanism for remote mechanisms and sp_helpserver will not display the rpc security model. For backward compatibility, databases that use DCE mechanism upgraded to this release will continue to work as earlier.

Getting information about servers

sp_helpserver reports on servers. Without an argument, sp_helpserver provides information about all the servers listed in sysservers. When you include a server name, sp_helpserver provides information about that server only:

    sp_helpserver [*server*]

sp_helpserver checks for both srvname and srvnetname in the master.sysremotelogins table.

For operating-system-specific information about setting up remote servers, see the installation documentation for your platform.

Dropping remote servers

A system security officer can use sp_dropserver to drop servers from sysservers:

    sp_dropserver *server* [, droplogins]

where:

- *server* – is the name of the server you want to drop.

- droplogins – allows you to drop a remote server and all of that server's remote login information. If you do not use droplogins, you cannot drop a server that has remote logins associated with it.

The following statement drops the GATEWAY server and all of the remote logins associated with it:

```
sp_dropserver GATEWAY, droplogins
```

You do not have to use droplogins to drop the local server; that entry does not have remote login information associated with it.

Adding remote logins

The system security officer and system administrator of any Adaptive Server share control over which remote users can access the server, and what identity the remote users assume. The system administrator uses sp_addremotelogin to add remote logins and sp_dropremotelogin to drop remote logins. The system security officer uses sp_remoteoption to control whether password checking is required.

Mapping users' server IDs

Logins from a remote server can be mapped to a local server in three ways:

*   A particular remote login can be mapped to a particular local login name. For example, user "joe" on the remote server might be mapped to "joesmith".

*   All logins from one remote server can be mapped to one local name. For example, all users sending remote procedure calls from the MAIN server might be mapped to "remusers".

*   All logins from one remote server can use their remote names.

The first option can be combined with the other two options, and its specific mapping takes precedence over the other two more general mappings. The second and third options are mutually exclusive; you can use either of them, but not both.

Changing the mapping option

Use sp_dropremotelogin to remove the old mapping.

Use sp_addremotelogin to add remote logins:

    sp_addremotelogin *remoteserver* [, *loginame*
        [, *remotename*]]

If the local names are not listed in master..syslogins, use create login to add them as Adaptive Server logins before you add the remote logins.

Only a system administrator can execute sp_addremotelogin. See the *Reference Manual: Procedures*.

Mapping remote logins to particular local names

The following example maps the login "pogo" from a remote system to the local login "bob". The user logs in to the remote system as "pogo". When "pogo" executes remote procedure calls from GATEWAY, the local system maps the remote login name to "bob".

```
create logn bob with password itsA8secret
sp_addremotelogin GATEWAY, bob, pogo
```

Mapping all remote logins to one local name

The following example creates an entry that maps all remote login names to the local name "albert". All names are mapped to "albert", except those with specific mappings, as described in the previous section. For example, if you mapped "pogo" to "bob", and then the rest of the logins to "albert", "pogo" still maps to "bob".

```
create logn albert with password itsA8secret
sp_addremotelogin GATEWAY, albert
```

If you use sp_addremotelogin to map all users from a remote server to the same local name, use sp_remoteoption to specify the "trusted" option for those users. For example, if all users from server GATEWAY that are mapped to "albert" are to be trusted, specify:

```
sp_remoteoption GATEWAY, albert, NULL, trusted, true
```

If you do not specify logins as trusted, they cannot execute RPCs on the local server unless they specify passwords for the local server when they log in to the remote server. Users can run ct_remote_pwd to specify a password for server-to-server connections when they use Open Client Client-Library. isql and bcp do not permit users to specify a password for RPC connections. See "Password checking for remote users" on page 304 for more information about sp_remoteoption.

Warning! Do not map more than one remote login to a single local login, as it reduces individual accountability on the server. Audited actions can be traced only to the local server login, not to the individual logins on the remote server.

If you are using
network-based
security

If users are logged in to the remote server using unified login, the logins must
be designated as trusted on the local server, or they must specify passwords for
the server when they log in to the remote server.

Warning! Using the trusted mode of sp_remoteoption reduces the security of
your server, as passwords from such "trusted" users are not verified.

Keeping remote login names for local servers

To enable remote users to keep their remote login names while using a local
server:

1   Use create login to create a login for each login from the remote server.

2   Use sp_addremotelogin for the server to create an entry in
    master..sysremotelogins with a null value for the remote login name and a
    value of -1 for the suid. For example:

```
sp_addremotelogin GATEWAY
```

Example of remote user login mapping

This statement displays the local and remote server information recorded in
master..sysservers:

```
select srvid, srvname from sysservers
srvid  srvname
-----  ----------
    0  SALES
    1  CORPORATE
    2  MARKETING
    3  PUBLICATIONS
    4  ENGINEERING
```

The SALES server is local. The other servers are remote.

This statement displays information about the remote servers and users stored
in master..sysremotelogins:

```
select remoteserverid, remoteusername, suid
  from sysremotelogins
remoteserverid   remoteusername   suid
-------------    --------------   ------
```

```
1                     joe                    1
1                     nancy                  2
1                     NULL                   3
3                     NULL                   4
4                     NULL                  -1
```

By matching the value of remoteserverid in this result and the value of srvid in the previous result, you can find the name of the server for which the remoteusername is valid. For example, in the first result, srvid 1 indicates the CORPORATE server; in the second result, remoteserverid 1 indicates that same server. Therefore, the remote user login names "joe" and "nancy" are valid on the CORPORATE server.

The following statement shows the entries in master..syslogins:

```
select suid, name from syslogins
suid    name
------  ------------
    1   sa
    2   vp
    3   admin
    4   writer
```

The results of all three queries together show:

- The remote user name "joe" (suid 1) on the remote CORPORATE server (srvid and remoteserverid 1) is mapped to the "sa" login (suid 1).

- The remote user name "nancy" (suid 2) on the remote CORPORATE server (srvid and remoteserverid 1) is mapped to the "vp" login (suid 2).

- The other logins from the CORPORATE server (remoteusername "NULL") are mapped to the "admin" login (suid 3).

- All logins from the PUBLICATIONS server (srvid and remoteserverid 3) are mapped to the "writer" login (suid 4).

- All logins from the ENGINEERING server (srvid and remoteserverid 4) are looked up in master..syslogins by their remote user names (suid -1).

- There is no remoteserverid entry for the MARKETING server in sysremotelogins. Therefore, users who log in to the MARKETING server cannot run remote procedure calls from that server.

The remote user mapping procedures and the ability to set permissions for individual stored procedures give you control over which remote users can access local procedures. For example, you can allow the "vp" login from the CORPORATE server to execute certain local procedures and all other logins from CORPORATE to execute the procedures for which the "admin" login has permission.

Note Typically, the passwords for users on the remote server must match passwords on the local server.

Password checking for remote users

A system security officer can use sp_remoteoption to determine whether passwords are checked when remote users log in to the local server. By default, passwords are verified (this is the "untrusted" mode). In trusted mode, the local server accepts remote logins from other servers and front-end applications without user-access verification for the particular login.

When sp_remoteoption is used with arguments, it changes the mode for the named user:

```
sp_remoteoption [remoteserver, loginame, remotename,
    optname, {true | false}]
```

For example, to set trusted mode for the user "bob", enter

```
sp_remoteoption GATEWAY, pogo, bob, trusted,
    true
```

Effects of using the untrusted mode

The effects of the untrusted mode depend on the user's client program. isql and some user applications require that logins have the same password on the remote server and the local server. You can write Open Client applications to allow local logins to have different passwords on different servers.

To change your password in "untrusted" mode, you must first change it on all the remote systems you access before you can change it on your local server. If you change your password on the local server first, when you issue the remote procedure call to execute sp_password on the remote server, your passwords no longer match.

The syntax for changing your password on the remote server is:

> *remote_server*...sp_password *caller_passwd*, *new_passwd*

On the local server, the syntax is:

> sp_password *caller_passwd*, *new_passwd*

See Chapter 5, "Managing Adaptive Server Logins and Database Users," in the *Security Administration Guide*.

Getting information about remote logins

sp_helpremotelogin prints information about the remote logins on a server. The following example shows the remote login "pogo" mapped locally to login name "bob", with all other remote logins keeping their remote names:

```
                  sp_helpremotelogin
server      remote_user_name      local_user_name      options
---------   ----------------      -----------------    --------
GATEWAY     **mapped locally**    **use local name**   untrusted
GATEWAY      pogo                  bob                  untrusted
```

Configuration parameters for remote logins

Table 7-2 shows the configuration parameters that affect RPCs. All these configuration parameters are set using sp_configure and do not take effect until Adaptive Server is restarted.

Table 7-2: Configuration parameters that affect RPCs

Configuration parameter	Default
allow remote access	1
number of remote logins	20
number of remote sites	10
number of remote connections	20
remote server pre-read packets	3

See the individual configuration parameter descriptions in Chapter 5, "Setting Configuration Parameters."

CHAPTER 8   **Initializing Database Devices**

Topic	Page
Database devices	307
Using the disk init command	308
disk init syntax	308
Getting information about devices	314
Dropping devices	316
Designating default devices	316
Increasing the size of devices with disk resize	318

Database devices

A database device stores the objects that make up databases. The term **device** does not necessarily refer to a distinct physical device: it can refer to any piece of a disk (such as a disk partition) or a file in the file system that is used to store databases and their objects.

Each database device or file must be prepared and made known to Adaptive Server before it can be used for database storage. This process is called **initialization**.

After a database device has been initialized, it can be:

- Allocated to the default pool of devices for the create and alter database commands

- Assigned to the pool of space available to a user database

- Assigned to a user database and used to store one or more database objects

- Assigned to store a database's transaction logs

Using the *disk init* command

A system administrator initializes new database devices with the disk init command, which:

*   Maps the specified physical disk device or operating system file to a *database device* name

*   Lists the new device in master..sysdevices

*   Prepares the device for database storage

Note Before you run disk init, see the installation documentation for your platform for information about choosing a database device and preparing it for use with Adaptive Server. You may want to repartition the disks on your computer to provide maximum performance for your Sybase databases.

---

disk init divides the database devices into **allocation units**, groups of 256 logical pages. The size of the allocation unit depends on which logical page size your server is configured for (2, 4, 8, or 16K). In each allocation unit, the disk init command initializes the first page as the allocation page, which contains information about the database (if any) that resides on the allocation unit.

Warning! After you run the disk init command, dump the master database. This makes recovery easier and safer in case master is damaged. See Chapter 13, "Restoring the System Databases," in *System Administration Guide: Volume 2*.

---

disk init syntax

See the *Reference Manual: Commands* for the disk init syntax.

Specifying a logical device name

The *device_name* must be a valid identifier. This name is used in the create database and alter database commands, and in the system procedures that manage segments. The logical device name is known only to Adaptive Server, not to the operating system on which the server runs.

Specifying a physical device name

The *physname* of the database device gives the name of a raw disk partition (UNIX), foreign device, or the name of an operating system file. On PC platforms, you typically use operating system file names for *physname*.

Choosing a device number

Adaptive Server accepts, but does not require, the disk init vdevno parameter. If you specify a vdevno, you may choose any currently unused identifier from 1 to 2,147,483,647 (virtual device ID 0 is used by the master device). For example, specifying vdevno = 33 assigns virtual device ID 33 to a device. If you do not specify a vdevno, Adaptive Server chooses a number higher than the highest vdevno currently listed in sysdevices.

The number of database devices you can create is limited by the number of devices configuration parameter. Adaptive Server is initially configured for 10 devices. Use sp_configure to change the value for number of devices.

Your operating system may also limit the number of devices your installation can use concurrently. To the operating system, each Sybase device counts as one open file.

Adaptive Server automatically specifies the next available identifying number for the database device. This is the virtual device number (vdevno). You need not specify this number when you issue the disk init command.

If you manually select the vdevno, it must be unique among the devices used by Adaptive Server. Device number 0 represents the master device. Legal numbers are 1 – 2,147,483,647. You can choose any unused vdevno within that range.

To see the numbers already in use for vdevno, look in the vdevno column of the report from sp_helpdevice, or use the following query to list all the device numbers currently in use:

```
select vdevno from master..sysdevices
   where status & 2=2
```

Here, status & 2=2 specifies physical disk.

Specifying the device size

To indicate the size of the device, use 'k' or 'K' for kilobytes, 'm' or 'M' for megabytes, 'g' or 'G' for gigabytes, and 't' or 'T' for terabytes. Sybase recommends that, to avoid confusion in the actual number of pages allocated, you always include the unit specifier in both the disk init and create database commands. Enclose the unit specifier in single or double quotes or in brackets.

Theoretically, you can create as many as 2,147,483,647 disk devices, each of which can be as large as 2,147,483,648 2K-blocks. The maximum installation size becomes a function of database size, hardware, and operating system limits.

The following guidelines apply to the syntax for disk init:

*   If you do not include a unit specifier for the size argument of disk init or disk reinit, size is measured, by default, in number of virtual pages. Thus, if you enter `size = 15000`, Adaptive Server assumes 15,000 virtual pages. A virtual page is 2048 bytes.

*   You can increase, but not decrease, the size of an existing database device using the disk resize command.

*   If you are planning to use the new device for the creation of a new database, the minimum size depends on the logical page size used by the server, described in Table 8-1:

Table 8-1: Minimum database sizes

Logical page size	Minimum database size
2K	3MB
4K	6MB
8K	12MB
16K	24MB

You cannot have a database smaller than the model database. A model database larger than the minimums listed above, determines the minimum database size.

Adaptive Server allocates and manages database space in allocation units, which are groups of 256 logical pages. The smallest database you can create (using create database) is 1MB; therefore, the size of the smallest usable database device is the larger of 1MB or 256 logical pages (for a 2K or 4K logical page size, this is 1MB for a 8K logical page size, this is 2MB, for a 16K logical page size, this is 4MB.

Keep this grouping of 256 pages in mind when you decide how large to make a device to avoid wasting space. For example, if your installation uses a 16k logical page size, specifying a device as `size = '31M'` leaves 3MB wasted at the end of the device, since an allocation unit is 4MB.

If you are initializing a raw device, determine the size of the device from your operating system, as described in the installation guide for your platform. Use the total size available, up to the maximum for your platform. After you have initialized the disk for use by Adaptive Server, you cannot use any space on that raw device for any other purpose.

disk init uses size to compute the value for the high virtual page number in sysdevices.high. The values for sysdevices.high and sysdevices.low are virtual page numbers in 2K-byte blocks, which is the Adaptive Server unit of physical disk management. This may not be the same as your installation's logical page size

---

Note If the physical device does not contain the number of blocks specified by the size parameter, disk init fails. If you use the optional vstart parameter, the physical device must contain the sum of the blocks specified by both the vstart and size parameters, or the command fails.

Specifying the *dsync* setting (optional)

For devices initialized on UNIX operating system files, the dsync setting controls whether or not writes to those files are buffered. When the dsync setting is on, Adaptive Server opens a database device file using the UNIX dsync flag, which ensures that writes to the device file occur directly to the physical storage media, and that Adaptive Server can recover data on the device in the event of a system failure.

When dsync is off, writes to the device file may be buffered by the UNIX file system, and the recovery of data on the device cannot be ensured. Turn off dsync only when data integrity is not required.

Note The dsync setting is ignored for devices initialized on raw partitions. Instead, writes to the database device take place directly to the physical media.

Performance implications of *dsync*

Using the dsync setting with database device files incurs several performance trade-offs:

* Adaptive Server does not support asynchronous I/O on operating system files for HP-UX.

* If database device files use the dsync option, the Adaptive Server engine that is writing to the device file waits until the write operation completes. This may cause poor performance during update operations.

* When dsync is on, write operations to database device files may be slower compared to earlier versions of Adaptive Server (where dsync is not supported). This is because Adaptive Server must write data to disk instead of simply copying cached data to the UNIX file system buffer.

  In cases where highest write performance is required (but data integrity after a system failure is not required) turning dsync off yields device file performance similar to earlier Adaptive Server versions. For example, you may consider storing tempdb on a dedicated device file with dsync disabled, if performance is not acceptable while using dsync.

* Response time for read operations is generally better for devices stored on UNIX operating system files as compared to devices stored on raw partitions. Data from device files can benefit from the UNIX file system cache as well as the Adaptive Server cache, and more reads may take place without requiring physical disk access.

Limitations and restrictions of *dsync*

The following limitations and restrictions apply to using dsync:

* For the master device, dsync is always set to true and you cannot change the setting.

* If you change a device file's dsync setting using the sp_deviceattr procedure, restart Adaptive Server for the change to take effect.

* When you upgrade from an Adaptive Server earlier than version 12.x, dsync is set to true only for the master device file. Use sp_deviceattr to change the dsync setting for any other device files.

* Adaptive Server ignores the dsync setting for database devices stored on raw partitions. Writes to devices stored on raw partitions are always directly to the physical media.

• The directio and dsync parameters are mutually exclusive. If a device has dsync set to true, you cannot set directio to true for the same device. To enable directio for a device, you must first reset dsync to false.

Using *directio* to bypass operating system buffer

The directio parameter for disk init, disk reinit, and sp_deviceattr allows you to configure Adaptive Server to transfer data directly to disk, bypassing the operating system buffer cache. directio performs I/O in the same manner as raw devices and provides the same performance benefit as raw devices, but has the ease of use and manageability of file system devices. You cannot set directio for the master device. directio is a static parameter; restart Adaptive Server for it to take effect.

Note directio is not available on all platforms. If you issue disk init with the directio parameter on a platform on which it is not supported, Adaptive Server issues the message `No such parameter: 'directio'.`

By default, the directio option is set to false (off) for all platforms.

Note Devices used for databases for which recovery is not important (for example, tempdb), may, by default, have dsync set to false. For these devices, enabling directio may have an adverse performance effect, so carefully review device use before you enable directio.

This example creates a device named "user_disk" that uses directio to write data directly to disk:

```
disk init
name = "user_disk",
physname = "/usr/u/sybase/data/userfile1.dat",
size = 5120, directio = true
```

To initializes 10MB of a disk on a UNIX operating system file, enter:

```
disk reinit
name = "user_disk",
physname = "/usr/u/sybase/data/userfile1.dat",
size = 5120, directio = true
```

By default, directio is disabled for all existing devices; enable it using sp_deviceattr:

sp_deviceattr *device_name*, directio, [true | false]

For example, the following enables directio disk writes for the "user_disk" device:

```
sp_deviceattr user_disk, directio, true
```

Other optional parameters for *disk init*

vstart is the starting virtual address, or the offset, for Adaptive Server to begin using the database device. vstart accepts the following optional unit specifiers: k or K (kilobytes), m or M (megabytes), g or G (gigabytes) and t or T (terabytes). The size of the offset depends on how you enter the value for vstart:

- If you do not specify a unit size, vstart uses 2K pages for its starting address. For example, if you specify vstart = 13, Adaptive Server uses 13 * 2K pages as the offset for the starting address.

- If you specify a unit value, vstart uses the value you enter as the starting address. For example, if you specify vstart = "13M", Adaptive Server sets the starting address offset at 13MB.

The default, and usually preferred value, of vstart is 0. If the specified device does not have the sum of vstart + size blocks available, the disk init command fails.

The optional cntrltype keyword specifies the disk controller. Its default value is 0. Reset it only if instructed to do so by your system administrator.

Note To perform disk initialization, the user who started Adaptive Server must have the appropriate operating system permissions on the device that is being initialized.

Getting information about devices

sp_helpdevice provides information about the devices in the sysdevices table.

When used without a device name, sp_helpdevice lists all the devices available on Adaptive Server. When used with a device name, it lists information about that device. Here, sp_helpdevice is used to report information about the master device:

```
                     sp_helpdevice master
device_name  physical_name  description
-----------  -------------  -----------------------------------------
master        d_master            special, default disk, physical disk, 30 MB


status        cntrltype     vdevno     vpn_low    vpn_high
------        ----------    -------    -------    ---------
3             0             0          0          10239
```

Each row in master..sysdevices describes:

- A dump device (tape, disk, or file) to be used for backing up databases, or

- A database device to be used for database storage.

The initial contents of sysdevices are operating-system-dependent. sysdevices entries usually include:

- One for the master device

- One for the sybsystemprocs database, which you can use to store additional databases such as pubs2 and sybsyntax, or for user databases and logs

- Two for tape dump devices

If you installed auditing, there is a separate device for sybsecurity.

The vpn_low and vpn_high columns represent the page numbers that have been assigned to the device. For dump devices, these columns represent the media capacity of the device.

The status field indicates the type of device, whether a disk device is used as a default storage device when users issue a create or alter database command without specifying a database device, disk mirroring information, and dsync settings.

Table 8-2: Status bits in sysdevices

Bit	Meaning
1	Default disk (may be used by any create or alter database command that does not specify a location)
2	Physical disk
4	Logical disk (not used)
8	Skip header (used with tape dump devices)
16	Dump device
32	Serial writes
64	Device mirrored

Bit	Meaning
128	Reads mirrored
256	Secondary mirror side only
512	Mirror enabled
2048	Used internally; set after disk unmirror, side = retain
4096	Primary device needs to be unmirrored (used internally)
8192	Secondary device needs to be unmirrored (used internally)
16384	UNIX file device uses dsync setting (writes occur directly to physical media)

For more information about dump devices and sp_addumpdevice, See Chapter 11, "Developing a Backup and Recovery Plan in the *System Administration Guide: Volume 2*.

Dropping devices

To drop database and dump devices, use sp_dropdevice:

> sp_dropdevice *logicalname*

You cannot drop a device that is in use by a database. You must drop the database first.

sp_dropdevice removes the device name from sysdevices. sp_dropdevice does not remove an operating system file; it only makes the file inaccessible to Adaptive Server. Use operating system commands to delete a file after using sp_dropdevice.

Designating default devices

To create a pool of default database devices to be used by all Adaptive Server users for creating databases, use sp_diskdefault after the devices are initialized. sp_diskdefault marks these devices in sysdevices as default devices. Whenever users create (or alter) databases without specifying a database device, new disk space is allocated from the pool of default disk space.

The syntax for sp_diskdefault is:

> sp_diskdefault *logicalname*, {defaulton | defaultoff}

After adding user devices, use the defaultoff option to remove the master device from the pool of default space:

```
sp_diskdefault master, defaultoff
```

The following designates sprocdev, the device that holds the sybsystemprocs database, a default device:

```
sp_diskdefault sprocdev, defaulton
```

Adaptive Server can have multiple default devices. They are used in the order in which they appear in the sysdevices table (that is, alphabetical order). When the first default device is filled, the second default device is used, and so on.

Note After initializing a set of database devices, you may want to assign them to specific databases or database objects rather than adding them to the default pool of devices. For example, you may want to make sure a table never grows beyond the size of a particular device.

Choosing default and nondefault devices

sp_diskdefault lets you plan space usage for performance and recovery, while allowing users to create or alter databases.

Do not use these devices as default devices:

- The master device

- The device used for sybsecurity

- Any device intended solely for logs

- Devices where high-performance databases reside

You can use the device that holds sybsystemprocs for other user databases.

Note If you are using disk mirroring or segments, exercise caution in deciding which devices you add to the default list. In most cases, devices that are to be mirrored, or databases that contain objects placed on segments should specifically allocate devices, rather than being made part of default storage.

Increasing the size of devices with *disk resize*

The disk resize command allows you to increase the size of your database devices dynamically, 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.

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

Page size	Allocation unit size	Minimum incremental size
2K	0.5MB	1MB
4K	1MB	1MB
8K	2MB	2MB
16K	4MB	4MB

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.

A user with the sa_role can execute the disk resize command, which:

• Updates the high value in master....sysdevices, and

• Prepares the additional space for database storage.

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.

Resizing a disk requires that:

• You have already initialized the device with disk init.

• *device_name* must refer to a valid logical device name.

• You disable mirroring while the resize operation is in progress. You can reestablish mirroring when the resize operation is complete.

In this example, the configuration of the device testdev is:

```
sp_helpdevice testdev
device_name   physical_name           description
  status   cntrltype   vdevno        vpn_low          vpn_high
```

```
----------   ------------------     -------------
  -------  ---------  -------------   -------------     --------------
testdev     /sybase/dev/testdev.dat   special, dsync on, directio off,
physical disk, 10.00MB
  16386    0             1                 0             5119
```

To increase the size of testdev by 4MB using disk resize, enter:

```
disk resize
name = "test_dev",
size = "4M"
```

testdev.dat is now 14MB:

```
sp_helpdevice testdev
device_name  physical_name          description
  status  cntrltype  vdevno          vpn_low          vpn_high
----------   ------------------     -------------
  -------  ---------  -------------   -------------     --------------
testdev     /sybase/dev/testdev.dat   special, dsync on, directio off,
physical disk, 14.00MB
  16386          0        1           0             7167
```

See the *Reference Manual: Commands* for disk resize syntax.

Insufficient disk space

During the physical initialization of the disk, if an error occurs due to insufficient disk space, disk resize extends the database device to the largest size possible before the error occurs.

For example, on a server that uses 4K logical pages, if you try to increase the size of the device by 40MB, but only 39.5MB is available, the device is extended only by 39.5MB.

You cannot decrease the size of a device with disk resize.

CHAPTER 9 **Setting Database Options**

Database options control:

- Transaction behavior

- Table-colum defaults

- User access restrictions

- Performance of recovery and bcp operations

- Log behavior

The system administrator and the database owner can use database options to configure settings for an entire database. Database options differ from sp_configure parameters, which affect the entire server, and set options, which affect only the current session or stored procedure.

Using the *sp_dboption* procedure

Use sp_dboption to change settings for an entire database. The options remain in effect until they are changed. sp_dboption:

- Displays a complete list of the database options when it is used without a parameter

- Changes a database option when used with parameters

You can change options only for user databases. You cannot change options for the master database. To change a database option in a user database (or to display a list of the database options), execute sp_dboption while using the master database.

The syntax is:

sp_dboption [*dbname*, *optname*, {true | false}]

Note Changes to model's database options do not affect tempdb or current user-defined multiple temporary databases when you restart Adaptive Server. These changes appear only in databases that you create after you change the model database. Restarting Adaptive Server clears objects and data contained in the temporary databases, but does not reset database options.

Database option descriptions

All users with access to the master database can execute sp_dboption with no parameters to display a list of the database options. The report from sp_dboption looks like this:

```
sp_dboption
Settable database options.
-------------------
abort tran on log full
allow nulls by default
allow wide dol rows
async log service
auto identity
dbo use only
ddl in tran
delayed commit
enforce dump tran sequence
full logging for all
full logging for alter table
full logging for reorg rebuild
full logging for select into
identity in nonunique index
no chkpt on recovery
no free space acctg
read only
select into/bulkcopy/pllsort
single user
trunc log on chkpt
trunc. log on chkpt.
unique auto_identity index
```

For a report on which options have been set in a particular database, execute sp_helpdb in that database.

See the *Commands Reference: Procedures* for information about each database option in detail.

Viewing the options on a database

Use sp_helpdb to determine the options that are set for a particular database. sp_helpdb lists each active option in the "status" column of its output.

The following example shows that the read only option is turned on in mydb:

```
            sp_helpdb mydb
name            db_size  owner  dbid  created        status
-----           -------  -----  ----  -----------    -------------------
mydb            20.0 MB    sa     5    Mar 05, 2005   read only

device_fragments      size    usage           created      free kbytes
----------------      ------  -----------     --------     -----------
master                10.0 MB data and log    Mar 05 2005         1792

device                        segment
----------------------------- ------------------------------
master                        default
master                        logsegment
master                        system
```

To display a summary of the options for all databases, use sp_helpdb without specifying a database:

```
                  sp_helpdb
name            db_size   owner dbid  created        status
-------------   --------  ----- ----  -----------    -------------------
master           48.0 MB    sa    1   Apr 12, 2005   mixed log and data
model             8.0 MB    sa    3   Apr 12, 2005   mixed log and data
pubs2            20.0 MB    sa    6   Apr 12, 2005   select into/
    bulkcopy/pllsort, trunc log on chkpt, mixed log and data
sybsystemdb       8.0 MB    sa    5   Apr 12, 2005   mixed log and data
sybsystemprocs  112.0 MB    sa    4   Apr 12, 2005   trunc log on chkpt,
    mixed log and data
tempdb            8.0 MB    sa    2   Apr 12, 2005   select into/
```

```
bulkcopy/pllsort, trunc log on chkpt, mixed log and data
```

Displaying currently set switches with sysoptions

The sysoptions.number column contains the switch ID for currently set switches. sysoptions shows these switches:

- Trace flag set in the runserver file with the -T flag

- Trace flag set with dbcc traceon(flag_number) or set switch serverwide on

- Trace flags and switches set for a specific system process ID (spid) with set switch on

sysoptions shows only the switches that are visible to the user. Users cannot see switches set privately by other spids. The value for number is Null for all option categories other than switches.

sysoptions displays this information for switches:

- spid – spid for the current session.

- name – name of the switch. If an unnamed switch is set, name contains the number of the switch converted to a string.

- category – specifies the string Switch.

- currentsetting – set to 1 if the switch is set, 0 if the switch is not set.

- defaultsetting – contains 0.

- scope – indicates whether the switch is set server-wide or for the session. Values are:

    - 0 – switch is not set.

    - 8 – switch is set server-wide.

    - 16 – switch is set privately to the current spid.

    - 24 – switch is set server-wide and privately.

- number – contains the switch ID as an integer.

This query displays all sysoptions entries for switches:

```
select * from sysoptions
```

```
where category = 'Switch'
```

This query displays the switches set for the current session:

```
select * from sysoptions
where category = 'Switch'
and currentsetting = '1'
```

This shows the sysoptions output after setting trace flag 3604:

```
spid         name                              category          currentsetting
     defaultsetting         scope              number
------       --------------------------        --------------    ---------------
     ---------------        ------------        --------------
   11           print_output_to_client              Switch                 1
                0                       8                              3604
```

Configuring Character Sets, Sort Orders, and Languages

This chapter discusses Adaptive Server internationalization and localization support issues.

Topic	Page
Understanding internationalization and localization	327
Advantages of internationalized systems	328
A sample internationalized system	329
Elements of an internationalized system	331
Selecting the character set for your server	331
Selecting the sort order	341
Selecting a language for system messages	349
Setting up your server: examples	350
Changing the character set, sort order, or message language	352
Installing date strings for unsupported languages	363
Internationalization and localization files	365

Understanding internationalization and localization

Internationalization is the process of enabling an application to support multiple languages and cultural conventions.

An internationalized application uses external files to provide language-specific information at execution time. Because it contains no language-specific code, an internationalized application can be deployed in any native language environment without code changes. A single version of a software product can be adapted to different languages or regions, conforming to local requirements and customs without engineering changes. This approach to software development saves significant time and money over the lifetime of an application.

Localization is the process of adapting an internationalized product to meet the requirements of one particular language or region, for example Spanish, including providing translated system messages; translations for the user interface; and the correct formats for date, time, and currency. One version of a software product may have many localized versions.

Sybase provides both internationalization and localization support. Adaptive Server includes the character set definition files and sort order definition files required for data processing support for the major business languages in Western Europe, Eastern Europe, the Middle East, Latin America, and Asia.

Sybase Language Modules provide translated system messages and formats for Chinese (Simplified), French, German, Japanese, Korean, Brazilian Portuguese, and Spanish. By default, Adaptive Server comes with U.S. English message files.

This chapter describes the available character sets and language modules and summarizes the steps necessary to change the default character set, sort order, or message language for Adaptive Server.

Advantages of internationalized systems

The task of designing an application to work outside its country of origin can seem daunting. Often, programmers think that internationalizing means hard-coding dependencies based on cultural and linguistic conventions for just one country.

A better approach is to write an internationalized application: that is, one that examines the local computing environment to determine what language to use and loads files containing language-specific information at runtime.

When you use an internationalized application, a single application can be deployed in all countries. This has several advantages:

*   You write and maintain one application.

*   The application can be deployed, without change, in new countries as needed. You need only supply the correct localization files.

*   All sites can expect standard features and behavior.

A sample internationalized system

An internationalized system may include internationalized client applications, gateways, and servers running on different platforms in different native language environments.

For example, an international system might include the following components:

- Order processing applications in New York City, Mexico City, and Paris (Client-Library applications)

- An inventory control server in Germany (Adaptive Server)

- An order fulfillment server in France (Adaptive Server)

- A central accounting application in Japan (an Open Server application working with an Adaptive Server)

In this system, the order processing applications:

- Query the inventory control server to determine if requested items are in stock

- Place orders with the order fulfillment server

- Send financial information to the accounting application

The inventory control server and the order fulfillment server respond to queries, and the accounting application collects financial data and generates reports.

The system looks like this:

Figure 10-1: Example of an international system
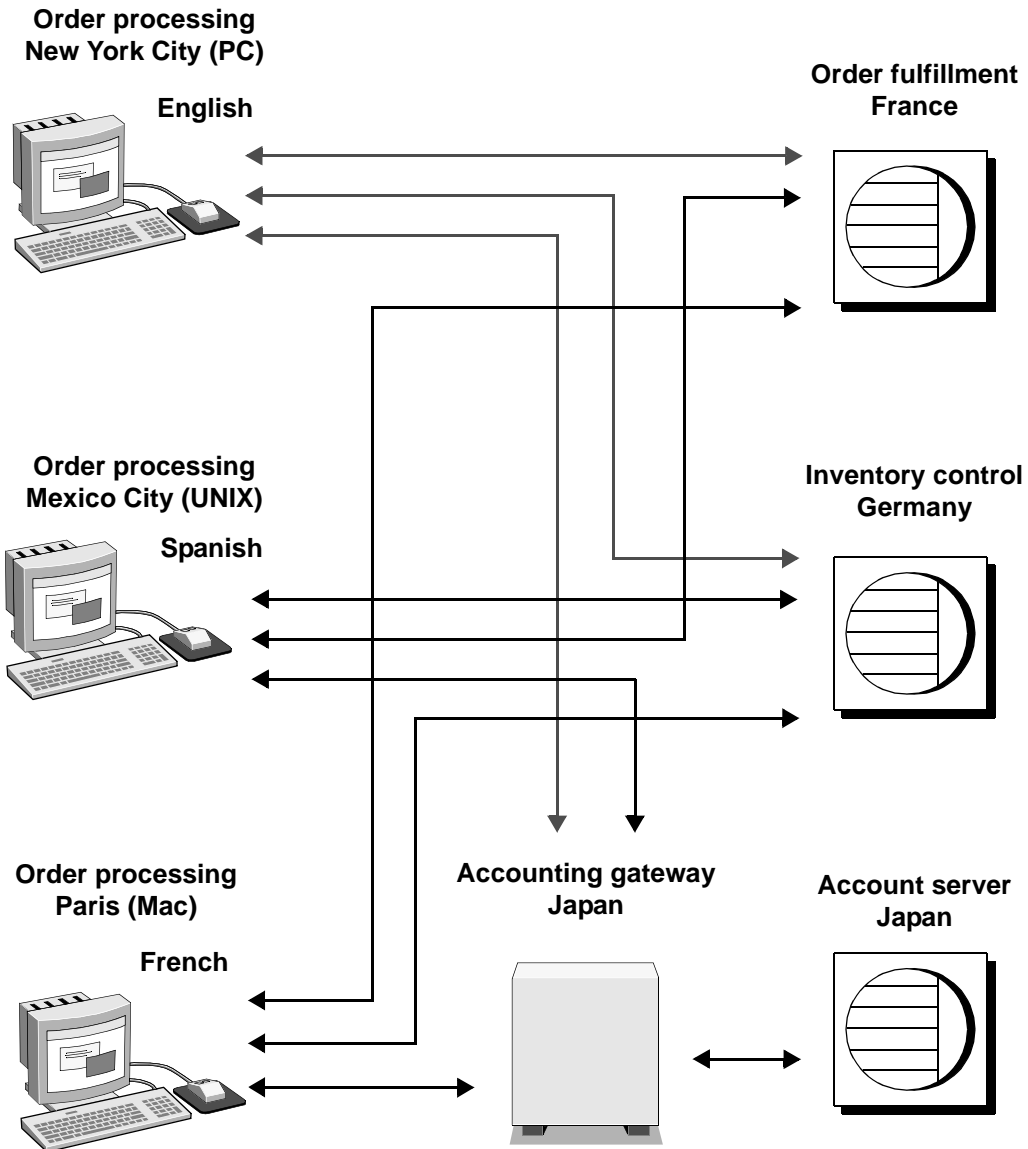
In this example, all applications and servers use local languages and character sets to accept input and output messages.

Elements of an internationalized system

There are three elements that you can manipulate to configure your server language in an internationalized environment. Sybase suggests that you review these three elements and carefully plan the client/server network you want to create.

- Character set – the language in which the server sends and receives data to and from the client servers. Select the character set after carefully planning and analyzing the language needs of all client servers.

- Sort order – sort order options are dependent on the language and character set you select.

- System messages – messages display in one of several languages provided by Sybase. If your server language is not one of the languages provided, your system messages display in English, the default.

The following sections provide details about each of these elements.

Selecting the character set for your server

All data is encoded in your server in a special code. For example, the letter "a" is encoded as "97" in decimal. A **character set** is a specific collection of characters (including alphabetic and numeric characters, symbols, and nonprinting control characters) and their assigned numerical values, or codes. A character set generally contains the characters for an alphabet, for example, the Latin alphabet used in the English language, or a script such as Cyrillic used with languages such as Russian, Serbian, and Bulgarian. Character sets that are platform-specific and support a subset of languages, for example, the Western European languages, are called **native** or **national character sets**. All character sets that come with Adaptive Server, except for Unicode UTF-8, are native character sets.

A **script** is a writing system, a collection of all the elements that characterize the written form of a human language—for example, Latin, Japanese, or Arabic. Depending on the languages supported by an alphabet or script, a character set can support one or more languages. For example, the Latin alphabet supports the languages of Western Europe (see Group 1 in Table 10-1 on page 333). On the other hand, the Japanese script supports only one language, Japanese. Therefore, the Group 1 character sets support multiple languages, while many character sets, such as those in Group 101, support only one language.

The language or languages that are covered by a character set is called a **language group.** A language group can contain many languages or only one language; a native character set is the platform-specific encoding of the characters for the language or languages of a particular language group.

Within a client/server network, you can support data processing in multiple languages if all the languages belong to the same language group (see Table 10-1 on page 333). For example, if data in the server is encoded in a Group 1 character set, you could have French, German, and Italian data and any of the other Group 1 languages in the same database. However, you cannot store data from another language group in the same database. For example, you cannot store Japanese data with French or German data.

Unlike the native character sets just described, **Unicode** is an international character set that supports over 650 of the world's languages, such as Japanese, Chinese, Russian, French, and German. Unicode allows you to mix different languages from different language groups in the same server, no matter what the platform. See "Unicode" on page 334 for more information.

Since all character sets support the Latin script, and therefore English, a character set always supports at least two languages—English and one other language.

Many languages are supported by more than one character set. The character set you install for a language depends on the client's platform and operating system.

Adaptive Server supports the following languages and character sets:

Table 10-1: Supported languages and character sets

Language group	Languages	Character sets
Group 1	**Western European:** Albanian, Catalan, Danish, Dutch, English, Faeroese, Finnish, French, Galician, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Spanish, Swedish	ASCII 8, CP 437, CP 850, CP 860, CP 863, CP 1252[a] , ISO 8859-1, ISO 8859-15, Macintosh Roman, ROMAN8, ROMAN9, ISO-15, CP 858
Group 2	**Eastern European:** Croatian, Czech, Estonian, Hungarian, Latvian, Lithuanian, Polish, Romanian, Slovak, Slovene (and English)	CP 852, CP 1250, ISO 8859-2, Macintosh Central European
Group 4	Baltic (and English)	CP 1257
Group 5	**Cyrillic:** Bulgarian, Byelorussian, Macedonian, Russian, Serbian, Ukrainian (and English)	CP 855, CP 866, CP 1251, ISO 8859-5, Koi8, Macintosh Cyrillic
Group 6	Arabic (and English)	CP 864, CP 1256, ISO 8859-6
Group 7	Greek (and English)	CP 869, CP 1253, GREEK8, ISO 8859-7, Macintosh Greek
Group 8	Hebrew (and English)	CP 1255, ISO 8859-8
Group 9	Turkish (and English)	CP 857, CP 1254, ISO 8859-9, Macintosh Turkish, TURKISH8
Group 101	Japanese (and English)	CP 932 DEC Kanji, EUC-JIS, Shift-JIS
Group 102	Simplified Chinese (PRC) (and English)	CP 936, EUC-GB, GB18030
Group 103	Traditional Chinese (ROC) (and English)	Big 5, CP 950[b] , EUC-CNS, Big 5 HKSCS
Group 104	Korean (and English)	EUC-KSC, cp949
Group 105	Thai (and English)	CP 874, TIS 620
Group 106	Vietnamese (and English)	CP 1258
Unicode	Over 650 languages	UTF-8

a. CP 1252 is identical to ISO 8859-1 except for the 0x80–0x9F code points which are mapped to characters in CP 1252.
b. CP 950 is identical to Big 5.

Note The English language is supported by all character sets because the first 128 (decimal) characters of any character set include the Latin alphabet (defined as "ASCll-7"). The characters beyond the first 128 differ between character sets and are used to support the characters in different native languages. For example, code points 0-127 of CP 932 and CP 874 both support English and the Latin alphabet. However, code points 128-255 support Japanese characters in CP 932 and code points 128-255 support Thai characters in CP 874.

The following character sets support the European currency symbol, the "euro": CP 1252 (Western Europe); CP 1250 (Eastern Europe); CP 1251 (Cyrillic); CP 1256 (Arabic); CP 1253 (Greek); CP 1255 (Hebrew); CP 1254 (Turkish); CP 874 (Thai); iso15, roman9 and CP858. Unicode UTF-8 also supports:

• Traditional Chinese on the Windows and Solaris platforms

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

Note iso_1 and ISO 8859-1 are different names for the same character set.

To mix languages from different language groups you *must* use Unicode. If your server character set is Unicode, you can support more than 650 languages in a single server and mix languages from any language group.

Unicode

Unicode is the first character set that enables all the world's languages to be encoded in the same data set. Prior to the introduction of Unicode, if you wanted to store data in, for example, Chinese, you had to choose a character set appropriate for that language—to the exclusion of most other languages. It was either impossible or impractical to mix character sets, and thus diverse languages, in the same data set.

Sybase supported Unicode in the form of three datatypes: unichar, univarchar, and unitext. These datatypes store data in the UTF-16 encoding of Unicode.

UTF-16 is an encoding wherein Unicode scalar values are represented by a single 16-bit value (or, in rare cases, as a pair of 16-bit values). The three encodings are equivalent insofar as either encoding can be used to represent any Unicode character. The choice of UTF-16 datatypes, rather than a UTF-16 server default character set, promotes easy, step-wise migration for existing database applications.

Adaptive Server supports Unicode literals in SQL queries and a wide range of sort orders for UTF-8.

The character set model used by Adaptive Server is based on a single, configurable, server-wide character set. All data stored in Adaptive Server, using any of the "character" datatypes (char, varchar, nchar, nvarchar, and text), is interpreted as being in this character set. Sort orders are defined using this character set, as are language modules—collections of server messages translated into local languages.

During the connection dialog, a client application declares its native character set and language. If properly configured, the server thereafter attempts to convert any character data between its own character set and that of the client (character data includes any data stored in the database, as well as server messages in the client's native language).This works well as long as the server's and client's character sets are compatible. It does not work well when characters are not defined in the other character set, as is the case for the character sets SJIS, used for Japanese, and KOI8, used for Russian and other Cyrillic languages. Such incompatibilities are the reason for Unicode, which can be thought of as a character superset, including definitions for characters in all other character sets.

The Unicode datatypes unichar, univarchar, and unitext are completely independent of the traditional character set model. Clients send and receive Unicode data independently of whatever other character data they send and receive.

Character set installation

Adaptive Server version 12.5.1 and later supports the 4-byte form of UTF-8. This form is used to represent the same rare Unicode characters that are represented in UTF-16 by pairs of 16-bit values ("surrogate pairs"). Prior to Adaptive Server version 12.5.1, only the 3-byte forms of UTF-8 were supported. If you have installed the UTF-8 character set in an Adaptive Server server earlier than version 12.5.1, you should reinstall it to enable the use of the 4-byte form of UTF-8.

Configuration parameters

The UTF-16 encoding of Unicode includes "surrogate pairs," which are pairs of 16-bit values that represent infrequently used characters. Additional checking is built in to Adaptive Server to ensure the integrity of surrogate pairs. You can switch this checking off by setting the configuration parameter "enable surrogate processing" to 0. This yields slightly higher performance, although the integrity of surrogate pairs is no longer guaranteed.

Unicode also defines "normalization," which is the process by which all possible representations of a single character are transformed into a single representation. Many base characters followed by combining diacritical marks are equivalent to precomposed characters, although their bit patterns are different. For example, the following two sequences are equivalent:

```
0x00E9  -- é (LATIN SMALL LETTER E WITH ACUTE)

0x00650301  -- e (LATIN SMALL LETTER E), ´ (COMBINING ACUTE ACCENT)
```

The enable unicode normalization configuration parameter controls whether or not Adaptive Server normalizes incoming Unicode data.

Significant performance increases are possible when the default Unicode sortorder is set to "binary" and the enable Unicode normalization configuration parameter is set to 1. This combination allows Adaptive Server to make several assumptions about the nature of the Unicode data, and code has been implemented to take advantage of these assumptions.

Functions

All functions taking char parameters have been overloaded to accept unichar as well. Functions with more than one parameter, when called with at least one unichar parameter, results in implicit conversion of any non-unichar parameters to unichar.

To guarantee the integrity of surrogate pairs when enable surrogate processing is set to 1 (the default), the string functions do not allow surrogate pairs to be split. Positions are modified to fall at the beginning of a surrogate pair.

Several functions have been added to round out the unichar support. Included are the functions to_unichar() and uscalar(), which are analogous to char() and ascii(). The functions uhighsurr() and ulowsurr() allow the explicit handling of surrogate pairs in user code.

There are restrictions when using unitext with functions. For information, see the restriction description under the "Usage" section for each function.

Using unichar columns

When using the isql or bcp utilities, Unicode values display in hexadecimal form unless the -Jutf8 flag is used, indicating the client's character set is UTF-8. In this case, the utility converts any Unicode data it receives from the server into UTF-8. For example:

```
% isql -Usa -P -Jiso_1
```

Adaptive Server Enterprise

```
1> select unicode_name from people where unicode_name = 'Jones'
2> go

unicode_name
------------------------------------------------------------------|
0x004a006f006e00650073
(1 row affected)
```

whereas:

```
% isql -Usa -P -Jutf8
1> select unicode_name from people where unicode_name = 'Jones'
2> go

unicode_name
------------------------------------------------------------------
Jones
(1 row affected)
```

This facilitates ad hoc queries. Not all terminal windows are capable of displaying the full repertoire of Unicode characters, but simple tests involving ASCII characters are greatly simplified.

Using unitext

The variable-length unitext datatype 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.

Open Client interoperability

The Open Client libraries support the datatype cs_unichar, which can be bound to user variables declared as an array of short integers. This Open Client datatype interfaces directly with the server's unichar, unitext, and univarchar.

Java interoperability

The internal JDBC driver efficiently transfers unichar data between SQL and Java contexts.

Going from SQL to Java, the class java.sql.ResultSet provides a number of "get" methods to retrieve data from the columns of a result set. Any of these get methods work with columns defined as unichar, unitext, or univarchar. The method getString() is particularly efficient since no conversion needs to be performed.

Use the setString() method of the class java.sql.PreparedStatement to go from Java to SQL. The internal JDBC driver copies Java string data directly into the SQL parameter defined as unichar, unitext, or univarchar.

The external JDBC driver (jConnect) has been modified to support the same seamless interface as the internal driver.

Limitations

Because the earlier releases of Adaptive Server did not include a Unicode-based language parser, a restriction was imposed on the use of the new Unicode datatypes. To use the new datatypes, the server required its default character set to be configured as UTF-8. This restriction has been removed in Adaptive Server release 12.5.1 and later. Unicode datatypes can be used regardless of the server's default character set.

Selecting the server default character set

When you configure your server, you must specify a default character set for the server. The default character set is the character set in which the server stores and manipulates data. Each server can have only one default character set.

By default, the installation tool assumes that the native character set of the platform operating system is the server's default character set. However, you can select any character set supported by Adaptive Server as the default on your server (see Table 10-1 on page 333).

For example, if you are installing the server on IBM RS/6000 running AIX, and you select one of the Western European languages to install, the installation tool assumes the default character set to be ISO 8859-1.

If you are installing a Unicode server, select UTF–8 as your default character set.

For non-Unicode servers, determine what platform most of your client systems use and use the character set for this platform as the default character set on the server.

This has two advantages:

• The number of unmappable characters between character sets is minimized.

Since there is usually not a complete one-to-one mapping between the characters in two character sets, there is a potential for some data loss. This is usually minor because most unconverted characters are special symbols that are not commonly used or are specific to a platform.

• This minimizes the character set conversion that is required.

When the character set on the client system differs from the default character set on the server, data must be converted in order to ensure data integrity. Although the measured performance decrease that results from character set conversion is insignificant, it is good practice to select the default character set that results in the fewest conversions.

For example, if most of your clients use CP 850, specify CP 850 on your server. You can do this even if your server is on an HP-UX system (where its native character set for the Group 1 languages is ROMAN8).

Note Sybase strongly recommends that you decide which character set to use as your default before you create any databases or make any changes to the Sybase-supplied databases.

In the example below (Figure 10-2), 175 clients all access the same Adaptive Server. The clients are on different platforms and use different character sets. The critical factor that allows these clients to function together is that *all* of the character sets in the client/server system belong to the same language group (see Table 10-1 on page 333). The default language for the Adaptive Server is CP 850, which is the character set used by the largest number of clients. This allows the server to operate most efficiently, with the least amount of character set conversion.

Figure 10-2: Clients using different character sets in the same language group
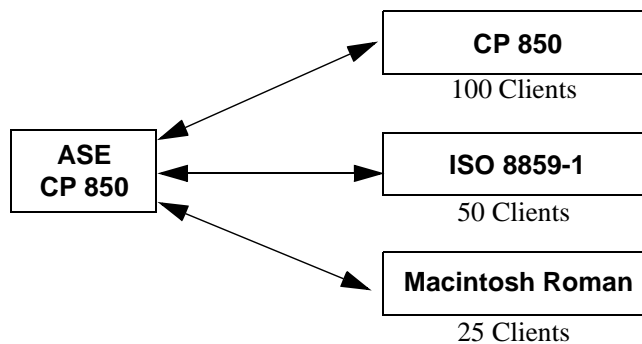
To help you choose the default character set for your server, the following tables list the most commonly used character sets by platform and language.

Table 10-2: Popular Western European client platforms

Platform	Language	Character set
Win 95, 98	U.S. English, Western Europe	CP 1252
Win NT 4.0	U.S. English, Western Europe	CP 1252
Win 2000	U.S. English, Western Europe	CP 1252
Sun Solaris	U.S. English, Western Europe	ISO 8859-1
HP-UX 10,11	U.S. English, Western Europe	ROMAN8
IBM AIX 4.x	U.S. English, Western Europe	ISO 8859-1

Table 10-3: Popular Japanese client platforms

Platform	Language	Character set
Win 95, 98	Japanese	CP 932 for Windows
Win NT 4.0	Japanese	CP 932 for Windows
Win 2000	Japanese	CP 932 for Windows
Sun Solaris	Japanese	EUC-JIS
HP-UX 10,11	Japanese	EUC-JIS
IBM AIX 4.x	Japanese	EUC-JIS

Table 10-4: Popular Chinese client platforms

Platform	Language	Character set
Win 95, 98	Chinese (simplified)	CP 936 for Windows
Win NT 4.0	Chinese (simplified)	CP 936 for Windows
Win 2000	Chinese (simplified)	CP 936 for Windows
Sun Solaris	Chinese (simplified)	EUC-GB
HP-UX 10,11	Chinese (simplified)	EUC-GBS
IBM AIX 4.x	Chinese (simplified)	EUC-GB

Selecting the sort order

Different languages sort the same characters differently. For example, in English, *Cho* would be sorted before *Co*, whereas in Spanish, the opposite is true. In German, β is a single character, however in dictionaries it is treated as the double character *ss* and sorted accordingly. Accented characters are sorted in a particular order so that *aménité* comes before *amène*, whereas if you ignored the accents, the reverse would be true. Therefore, language-specific sort orders are required so that characters are sorted correctly.

Each character set comes with one or more sort orders that Adaptive Server uses to collate data. A sort order is tied to a particular language or set of languages and to a specific character set. The same sort orders can be used for English, French, and German because they sort the same characters identically, for example, *A*, *a*, *B*, *b*, and so on. Or the characters are specific to one of the languages—for example, the accented characters, *é* , *à*, and *á*, are used in French but not in English or German—and therefore, there is no conflict in how those characters are sorted. The same is not true for Spanish however, where the double letters *ch* and *ll* are sorted differently. Therefore, although the same character sets support all four languages, there is one set of sort orders for English, French and German, and a different set of sort orders for Spanish.

In addition, a sort order is tied to a particular character set. Therefore, there is one set of sort orders for English, French, and German in the ISO 8859-1 character set, another set in the CP 850 character set, and so on. The sort orders available for a particular character set are located in sort order definition files (**.srt* files) in the character set directory. For a list of character sets and their available sort orders, see Table 10-5 on page 344.

Using sort orders

Sort orders are used to:

* Create indexes
* Store data into indexed tables
* Specify an order by clause

Different types of sort orders

All character sets are offered with a binary sort order at a minimum, which blindly sorts all data based only on the arithmetic value of the code assigned to represent each letter (the "binary" code) in the character set. Binary sort order works well for the first 128 characters of each character set (ASCII English) and for Asian languages.When a character set supports more than one language (for example, Group 1 or Unicode) the binary sort order most likely give incorrect results, and you should select another sort order.

Character sets may also have one or more of the following dictionary sort orders:

- *Dictionary order, case-sensitive, accent-sensitive* – sorts uppercase and lowercase letters separately. Dictionary order recognizes the various accented forms of a letter and sorts them after the associated unaccented letter.

- *Dictionary order, case-insensitive, accent-sensitive* – sorts data in dictionary order but does not recognize case differences. Uppercase letters are equivalent to their lowercase counterparts and are intermingled in sorting results. Useful for avoiding duplicate entries in tables of names.

- *Dictionary order, case-insensitive, accent-sensitive, order with preference* – does not recognize case difference in determining equivalency of items. A word in uppercase is equivalent to the same word in lowercase. Preference is given to uppercase letters (they appear first) if all other conditions are equal.

  Using case-insensitive with preference may cause poor performance in large tables when the columns specified in an order by clause match the key of the table's clustered index. Do not select case-insensitive order with preference unless your installation requires that uppercase letters be sorted before lowercase letters in otherwise equivalent strings for order by clauses.

- *Dictionary order, case-insensitive, accent-insensitive* – treats accented forms of a letter as equivalent to the associated unaccented letter. It intermingles accented letters in sorting results.

Selecting the default sort order

Sybase servers can support only one default sort order at a time. If your users are using the same language or their languages use the same sort order, then select the desired sort order. For example, if your users are using French data and expect French sorting, then you can pick one of the French dictionary sort orders. Or if your users are using data in multiple languages and the languages use the same sort order, for example English, French, and German, you can pick one sort order and it works for all your users in all languages.

However, if you have users using different languages that require different sort orders, for example French and Spanish, then you must select one of the sort orders as the default. If you pick, for example, a French sort order, your Spanish users will not see the *ch* and *ll* double characters sorted as they would expect. The installation procedure, by default, configures the server with the binary sort order.

You can use the sortkey function to setup customized alternative sort orders for your data—one for each language.These sort orders can be selected dynamically to meet the needs of different users. The sortkey function is separate from the default sort order, but can coexist in the same server. The range and depth of sort orders provided by the sortkey function is better than those provided by the default sort order mechanism. For more information, see sortkey and compare in the *Reference Manual: Building Blocks*.

Table 10-5: Available sort orders

Language or script	Character sets	Sort orders
All languages	UTF-8	Multiple sort orders, see Table 10-7 for list
Cyrillic: Bulgarian, Byelorussian, Macedonian, Russian, Serbian, Ukrainian	CP 855, CP 866, CP 1251, ISO 8859-5, Koi8, Macintosh Cyrillic	Dictionary order, case sensitive, accent sensitive
Eastern European: Czech, Slovak	CP 852, ISO 8859-2, CP 1250	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent sensitive
		Dictionary order, case sensitive, accent sensitive, with preference
		Dictionary order, case insensitive, accent insensitive
English, French, German	ASCII 8, CP 437, CP850, CP 860, CP 863, CP 1252a, ISO 8859-1, ISO 8859-15, Macintosh Roman, ROMAN8, ROMAN9, ISO 15	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent sensitive
		Dictionary order, case sensitive, accent sensitive, with preference
		Dictionary order, case insensitive, accent insensitive
English, French, German	CP 850, CP 858	Alternate dictionary order, case sensitive
		Alternate dictionary order, case sensitive, accent insensitive
		Alternate dictionary order, case sensitive, with preference
Greek	ISO 8859-7	Dictionary order, case sensitive, accent sensitive
Hungarian	ISO 8859-2	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent sensitive
		Dictionary order, case insensitive, accent insensitive
Japanese	EUCJIS, SJIS, DECKANJI	General purpose case-insensitive dictionary ordering
Kazakh	87	50
Russian	CP 866, CP 1251, ISO 8859-5, Koi8, Macintosh Cyrillic	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent sensitive

Language or script	Character sets	Sort orders
Scandinavian	CP 850	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, with preference
Simplified Chinese	EUC-GB, GB-18030, CP936	General purpose case-insensitive dictionary ordering
Spanish	ASCII 8, CP 437, CP850, CP 860, CP 863, CP 1252, ISO 8859-1, ISO 8859-15, Macintosh Roman, ROMAN8	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent sensitive
		Dictionary order, case insensitive, accent insensitive
Thai	CP 874, TIS 620	Dictionary order
Turkish	ISO 8859-9	Dictionary order, case sensitive, accent sensitive
		Dictionary order, case insensitive, accent insensitive
		Dictionary order, case insensitive, accent sensitive
Western European	CP 1252	Dictionary order, case insensitive, case sensitive, with preference, accent insensitive, Spanish dictionary, Spanish case insensitive, Spanish accent insensitive

If your language does not appear here, there is no language-specific sort order for your language. Select a binary sort order and then investigate whether the sortkey function meets your needs. As this table illustrates, many languages have more than one sort order.

Chinese Pinyin sort order

Pinyin, more formally known as "Hanyu Pinyin," uses the Roman alphabet to represent the standard Chinese pronunciation system. Pinyin consists of a system of transliteration to Roman alphabets for reading and writing Mandarin without Chinese characters. Pinyin uses accents to represent the four tones of Mandarin.

Earlier versions of Adaptive Server used the Simplified Chinese (GB) sort orders, gbpinyin and gbpinyinocs, using the Unilib character set, significantly impacting the performance of databases using the GB character sets.

Adaptive Server version 15.0.3 automatically uses the gbpinyin and gbpinyinocs sort orders, eliminating a processing step and significantly improving performance.

In earlier versions, the default size of unilib cache configuration parameter was 268 KB. In version 15.0.3, the default has been increased to 302 KB.

Improved performance occurs in queries that access ASCII and gbpinyin data. However, if the data set has a mixture of other characters, you may not see any performance improvement.

See Chapter 9, "Configuring Character Sets, Sort Orders, and Languages" in the *System Administration Guide* for information about configuring Adaptive Server to use the gbpinyin and gbpinyinocs sort orders.

Selecting case-insensitive sort orders for Chinese and Japanese character sets

Use two stored procedures to select case-insensitive sort orders:

- sp_helpsort

- sp_configure

sp_helpsort   sp_helpsort lists the available case-insensitive sort orders.

```
sp_helpsort
-----------
Name                     ID
-------------------------
nocase_eucgb             52
nocase_cp936             52
nocase_gb18030           52
nocase_eucjis            52
nocase_sjis              52
nocase_deckanji          52
```

sp_configure   To switch to a case-insensitive sort order, enter:

```
sp_configure 'default sortorder id', 52
```

Selecting the default Unicode sort order

The default Unicode sort order is distinctly different from the sort order for the server's default character set. This separate configuration parameter is a static parameter that requires that you restart your server and reindex the unichar data if it is changed. This sort order is identified using a string parameter, rather than a numeric parameter, to guarantee that the sort order is unique.

Table 10-6 lists the available default Unicode sort orders.

Table 10-6: Default Unicode sort orders

Name	ID	Description
defaultml	20	Default Unicode multi-lingual ordering
thaidict	21	Thai dictionary ordering
iso14651	22	Ordering as per ISO14651 standard
utf8bin	24	Ordering for UTF-16 that matches the UTF-8 binary
binary	25	Binary sort
altnoacc	39	Alternate accent-insensitive
altdict	45	Alternate dictionary ordering
altnocsp	46	Alternate case-insensitive with preference
scandict	47	Scandinavian dictionary ordering
scannocp	48	Scandinavian case-insensitive with preference
bin_utf8	50	UTF-8 binary sort order
dict	51	General-purpose dictionary ordering
nocase	52	General-purpose case-insensitive dictionary ordering
nocasep	53	General-purpose case-insensitive with preference
noaccent	54	General-purpose accent-insensitive dictionary ordering
espdict	55	Spanish dictionary ordering
espnocs	56	Spanish case-insensitive dictionary ordering
espnoac	57	Spanish accent-insensitive dictionary ordering
rusnocs	59	Russian case-insensitive dictionary ordering
cyrnocs	64	Cyrillic case-insensitive dictionary ordering
elldict	65	Greek dictionary ordering
hundict	69	Hungarian dictionary ordering
hunnoac	70	Hungarian accent-insensitive dictionary ordering
hunnocs	71	Hungarian case-insensitive dictionary ordering
turknoac	73	Turkish accent-insensitive dictionary ordering

Table 10-7 lists the loadable sort orders.

Table 10-7: Loadable sort orders

Name	ID	Description
cp932bin	129	Ordering that matches the binary ordering of CP932
dynix	130	Chinese phonetic ordering
gb3213bn	137	Ordering that matches the binary ordering of GB2312
cyrdict	140	Common cyrillic dictionary ordering
turdict	155	Turkish Dictionary ordering
euckscbn	161	Ordering that matches the binary ordering of EUCKSC
gbpinyin	163	Chinese phonetic ordering

Name	ID	Description
rusdict	165	Russian dictionary ordering
sjisbin	179	Ordering that matches the binary ordering of SJIS
eucjisbn	192	Ordering that matches the binary ordering of EUCJIS
big5bin	194	Ordering that matches the binary ordering of BIG5

To view this sort order list in Adaptive Server, use sp_helpsort. See the *Reference Manual: Procedures*.

You can add sort orders using external files in the *$SYBASE/collate/Unicode* directory. The names and collation IDs are stored in syscharsets. The names of external Unicode sort orders do not have to be in syscharsets before you can set the default Unicode sort order.

Note External Unicode sort orders are provided by Sybase. Do not attempt to create external Unicode sort orders.

Sort order associated with Unicode data is completely independent of the sort order associated with traditional character data. All relational expressions involving the Unicode datatypes are performed using the Unicode sort order. This includes mixed-mode expressions involving Unicode and non-Unicode data. For example, in the following query the varchar character constant 'Mü' is implicitly cast to unichar and the comparison is performed according to the Unicode sort order:

```
select * from authors where unicode_name > 'Mü'
```

The same holds true for all other comparison operators, as well as the concatenation operator "+", the operator "in", and the operator "between." Once again, the goal is to retain compatibility with existing database applications.

Tables joins based on equality (equijoins) deserve special mention. These are generally optimized by the server to take advantage of indexes that defined on the participating columns. When a unichar column is joined with a char column, the latter requires a conversion, and since the character sort order and the Unicode sort order are distinct, the optimizer will ignore the index on the char column.

In Adaptive Server version 12.5.1 and later, when the server's default character set is configured to UTF-8, you can configure the server's default sort order (for char data) to be any of the above sort orders. Prior to this version, the binary sort order "bin_utf8" (ID=50) was the only well-behaved sort order for UTF-8. Although not required, the sort order for char data in UTF-8 can be selected so that it corresponds with the sort order for unichar.

There is a potential confusion regarding choice of binary sort orders for Unicode. The sort order named "binary" is the most efficient one for unichar data (UTF-16), and is thus the default. This order is based on the Unicode scalar value, meaning that all 32-bit surrogate pairs are placed after all 16-bit Unicode values. The sort order named "utf8bin" is designed to match the order of the default (most efficient) binary order for UTF-8 char data, namely "bin_utf8". The recommended matching combinations are thus "binary" for unichar and "binary" for UTF-8 char, or "utf8bin" for unichar and "bin_utf8" for UTF-8 char. The former favors unichar efficiency, while the latter favors char efficiency. Avoid using "utf8bin" for UTF-8 char, since it is equivalent to "bin_utf8" but less efficient.

Selecting a language for system messages

Any installation of Adaptive Server can use Language Modules containing files of messages in different languages. Adaptive Server provides Language Modules for messages in the following languages: English, Chinese (Simplified), French, German, Japanese, Korean, Brazilian Portuguese, and Spanish. If your client language is *not* one of these languages, you see system messages in English, the default language.

Each client can choose to view messages in their own language at the same time, from the same server; for example, one client views system messages in French, another in Spanish, and another in German. To do this, however, all selected languages must be part of the same language group. For example, French, Spanish and German are all part of language group 1. Japanese, on the other hand, is part of language group 101, which contains no other languages. Therefore, if Japanese is your server language, you can display system messages only in Japanese or English. Remember that all language groups can display messages in English. There is also a server-wide default language, used if the user has not selected a specific language. If you use Unicode, you can view system messages in any of the supported languages.

You can select the language for your system messages in one of two ways:

- Select a language as part of your user profile

- Enter a language in the *locales.dat* file

Table 10-8 displays the supported system message languages and their language groups. Each user can select only one language per session for system messages.

Table 10-8: Supported system messages

Language group	System message languages	Character sets
Group 1	French, German, Spanish, Brazilian Portuguese	ASCII 8, CP 437, CP 850, CP 860, CP 863, CP 1252, ISO 8859-1, ISO 8859-15, Macintosh Roman, ROMAN8
Group 2	Polish	Cp 1250, CP 852, ISO 8859-2
Group 101	Japanese	CP 932, DEC Kanji, EUC-JIS, Shift-JIS
Group 102	Simplified Chinese (PRC)	CP 936, EUC-GB, GB18030
Group 104	Korean	EUC-KSC, CP 949
Group 105	Thai	CP 874, TIS 620
Unicode	French, German, Spanish, Brazilian Portuguese, Japanese, Simplified Chinese, Korean	UTF-8
All Other Language Groups	English	

Install Language Modules for all languages in which clients will receive messages. These Language Modules, located in the *locales* subdirectory of the Adaptive Server installation directory, are part of a group of files called localization files. For information about localization files and the software message directory structure, see "Types of localization files" on page 366.

Setting up your server: examples

This section discusses setup options and the steps necessary to implement them. This is only a sample, and is meant to suggest ideas and methods for your own setup process.

A Spanish-version server

This examples shows how to set up a new server with all clients using the same language. To do this:

1   Select the server language, in this case, Spanish. By reviewing Table 10-1 on page 333, you see that Spanish is part of language group 1. Based on your platform, select a character set from language group 1. Sybase recommends that you select the character set used by the greatest number of clients. Or, if you think your company might someday expand into other countries and languages, you might consider installing Unicode (see "Selecting the character set for your server" on page 331).

2   Install the Spanish Language Module in the server. This allows clients to view system messages in Spanish.

3   Select the default sort order. By referring to Table 10-5 on page 344, you see that Spanish has three possible sort orders, in addition to binary sort order. Select a sort order.

4   Restart the server.

A U.S.-based company in Japan

This example involves clients in Japan, who want to enter data, sort data, and receive system messages in Japanese, while submitting data to a server that is accessed by English-only users:

1   Select the default character set for your server. If you install a character set from language group 101 (Japanese), you can support both Japanese and English data in the same server.

2   Install the Japanese Language Module so that system messages are available in Japanese.

3   Select the sort order. By referring to Table 10-5 on page 344, you can see that a binary sort order is the only sort order available for Japanese. Therefore, both the English and Japanese clients have a default binary sort order. Consider using the sortkey function to provide solutions for both audiences.

4   Make sure that each Japanese user requests Japanese messages by default. Since you are using a character set from language group 101, and you have already installed the Japanese Language Module, your client in Japan sees messages in Japanese, while clients in the U.S. can choose to see messages in either English or Japanese.

A Japan-based company with multinational clients

This company is located in Japan, and has clients in France, Germany, and Spain. You need to mix European and Asian languages in the same server.

1   Select the default server language and character set. Since your company is based in Japan and most of your clients are located in Japan, the default server language should be Japanese. But you also want your clients in France, Germany, and Spain to be able to send and receive data in their native languages. By reviewing Table 10-1 on page 333, you can see that Japanese is part of language group 101, while French, German, and Spanish are part of language group 1. Since the languages you need are not part of the same language group, the only way you can have all of these languages on the same server is to select Unicode as your default character set.

2   Install the Language Modules for Japanese, French, German, and Spanish.

3   Select the binary sort order, since this is the only sort order available for the Unicode character set. (You can, however, consider using the sortkey function inside your application code to supply data sorted according to each user's preference.)

4   Select Japanese as the default language for system messages. Clients in other countries can select their own native language for messages.

Changing the character set, sort order, or message language

Even after you have configured your server, a system administrator can change the default character set, sort order, or message language used by Adaptive Server. Because a sort order is built on a specific character set, changing character sets always involves a change in sort order. However, you can change the sort order without changing character sets, because more than one sort order may be available for a character set.

To display Adaptive Server's default sort order, character set, and a table of its primary sort orders, enter:

```
sp_helpsort
```

Changing the default character set

Adaptive Server can have only one default character set, the character set in which data is stored in its databases. When you install Adaptive Server, you specify a default character set.

Warning! Read the following carefully, and exercise caution when changing the default character set in Adaptive Server. Sybase strongly recommends that you perform backups before you change a default character set.

When you change the default character set in Adaptive Server, you must convert any existing data to the new default character set. Conversion is unnecessary only if:

- There is no user data in the server.
- It is acceptable to destroy user data in the server.
- You are absolutely certain that data in the server uses only ASCll-7. In this case, you can change the default without first copying your data out of the server.

In all other cases, you must convert the existing data as follows:

1   Copy the data out using bcp.

2   Change the default character set.

3   Use bcp with the appropriate flags for data conversion to copy the data back into the server.

See the *Utility Guide* for more information about using bcp to copy data.

Warning! After converting data to a different character set (particularly to UTF-8), the data may be too large for the allocated column size. Re-create the columns affected with a larger size.

Code conversion between the character set of the existing data and the new default character set must be supported. If it is not, conversion errors will occur and the data is not converted correctly. See Chapter 11, "Configuring Client/Server Character Set Conversions," for more information about supported character set conversions.

Even if conversions are supported between the character sets, some errors may occur due to minor differences between the character sets, or because some characters do not have equivalents in other character sets. Rows containing problematic data may not get copied back into the database, or data may contain partial or invalid characters.

Changing the sort order with a resources file

Adaptive Server character sets can be changed using the resource file. The sample resource file *sqlloc.rs* is located in *$SYBASE/ASE-12_5/init/sample_resource_files/*.

The resource file from the Adaptive Server 12.5.1 installation looks similar to the following:

```
sybinit.release_directory: USE_DEFAULT
sqlsrv.server_name: PUT_YOUR_SERVER_NAME_HERE
sqlsrv.sa_login: sa
sqlsrv.sa_password:
sqlsrv.default_language: USE_DEFAULT
sqlsrv.language_install_list: USE_DEFAULT
sqlsrv.language_remove_list: USE_DEFAULT
sqlsrv.default_characterset: USE_DEFAULT
sqlsrv.characterset_install_list: USE_DEFAULT
sqlsrv.characterset_remove_list: USE_DEFAULT
sqlsrv.sort_order: USE_DEFAULT
# An example sqlloc resource file...
# sybinit.release_directory: USE_DEFAULT
# sqlsrv.server_name: PUT_YOUR_SERVER_NAME_HERE
# sqlsrv.sa_login: sa
# sqlsrv.sa_password:
# sqlsrv.default_language: french
# sqlsrv.language_install_list: spanish,german
# sqlsrv.language_remove_list: USE_DEFAULT
# sqlsrv.default_characterset: cp437
# sqlsrv.characterset_install_list: mac,cp850
# sqlsrv.characterset_remove_list: USE_DEFAULT
# sqlsrv.sort_order: dictionary
```

Changing the default sort order

Adaptive Server can have only one default sort order, the collating sequence it uses to order data. When you consider changing the sort order for character data on a particular Adaptive Server, keep this in mind: all of your organization's Adaptive Servers should have the same sort order. A single sort order enforces consistency and makes distributed processing easier to administer.

You may have to rebuild your indexes after changing the default sort order. For more information, see "Reconfiguring the character set, sort order, or message language" on page 355.

Reconfiguring the character set, sort order, or message language

This section summarizes the steps to take before and after changing Adaptive Server's default character set, sort order, or message language. For procedures on how to configure the character set, sort order, or message language for a new server, see the configuration documentation for your platform.

Back up all databases in Adaptive Server before and after you change character sets or sort orders. After you back up your databases, use bcp to copy the data in and out of your databases if:

*   A database contains character data and you want to convert the data to a new character set. Do not load a database dump of the data into a server that uses the new default character set. Adaptive Server assumes the loaded data is in the new character set, and corrupts the data.

*   You are changing the default sort order only and not the default character set. You cannot load a database from a dump performed prior to changing the sort order—if you attempt to, an error message appears, and Adaptive Server aborts the load.

*   You change the default character set, and either the old or the new sort order is not binary. You cannot load a database dump that was made before you changed the character set.

You cannot reload your data from a database dump once you have reconfigured the default character set and sort order (unless both old and new character sets use a binary sort order and no conversion is required between the old and new character sets). See "Changing the default character set" on page 353 for more information,

Unicode examples

In the following example, a fictitious database named xpubs is modified to use univarchar columns.

Schema

Assume a database was created using the following script on a server that has all the installation defaults, namely character set "iso_1" and default sort order ID 50, "binary_iso_1".

```
> create database xpubs
> go
> use xpubs
> go
> create table authors (au_id int, au_lname
varchar(255), au_fname varchar(255))
> go
> create index au_idx on authors(au_lname, au_fname)
> go
```

Then the data was loaded into the server using a series of inserts and updates.

Converting to UTF-8

The first step towards using Unicode is to extract the data and convert it to UTF-8 form.

```
% bcp xpubs..authors out authors.utf8.bcp -c -Jutf8 -Usa -P
```

The next step to install UTF-8 as the default character set in the server:

```
% charset -Usa -P binary.srt utf8
% isql -Usa -P
> sp_configure 'default sortorder id', 50, 'utf8'
> go
> shutdown
> go
```

Restart the server to modify the default character set and re-create indexes on the system tables. Restart the server a second time, then reload the data:

```
% isql -Usa -P
> sp_dboption xpubs, 'select into', true
> go
> use xpubs
> go
> checkpoint
```

Adaptive Server Enterprise

```
> go
> delete from authors
> go
> quit

% bcp xpubs..authors in authors.utf8.bcp -c -Jutf8 -Usa -P
```

Migrating selected columns to unichar

With a working database running with UTF-8 as the default character set, it becomes a simple matter to convert select columns to univarchar:

```
% isql -Usa -P
> use xpubs
> go
> alter table authors modify au_lname univarchar(255),
au_fname univarchar(255)
> go
```

The columns are modified to the new datatypes, the data is converted in place, and the index is re-created.

Migrating to or from unitext

Currently, the alter table modify command does not support text, image, or unitext columns. To migrate from a text to a unitext column, you must first use bcp, create a table with unitext columns, and then use bcp again to place data into the new table. This migration path only works when you invoke bcp with -Jutf8 option.

Preliminary steps

Before you run the installation program to reconfigure Adaptive Server:

1   Dump all user databases and the master database. If you have made changes to model or sybsystemprocs, dump them also.

2   Load the Language Module if it is not already loaded (see the configuration documentation for your platform for complete instructions).

3   If you are changing the Adaptive Server default character set, and your current databases contain non ASCII-7 data, use bcp to copy the existing data out of your databases.

Once you have loaded the Language Module, you can run the Adaptive Server installation program, which allows you to:

- Install or remove message languages and character sets included with Adaptive Server

- Change the default message language or character set

- Select a different sort order

See the configuration documentation for your platform for instructions on using the installation program

---

Note Before you change the character set or sort order, Adaptive Server must have as many open databases as there are databases managed by the server. If Adaptive Server does not have a sufficient number of open databases when it is re-started after a change in sort order, Adaptive Server prints this message to the error log and the server will revert to the former sort order:

```
The configuration parameter 'number of open databases'
must be at least as large as the number of databases,
in order to change the character set or sort order." Re-
start Adaptive Server, use sp_configure to increase
'number of open databases' to at least %d, then re-
configure the character set or sort order
```

To reconfigure the language, character set, or sort order, use the sqlloc utility, described in *Utility Guide*. If you are using Windows, use the Server Config utility, described in Chapter 3, "Default Adaptive Server Configuration," in the *Configuration Guide*.

If you installed additional languages but did not change the Adaptive Server character set or sort order, you have completed the reconfiguration process.

If you changed the Adaptive Server default character set, and your current databases contain non ASCII-7 data, copy your data back into your databases, using bcp with the necessary flags to enable conversion.

If you changed the Adaptive Server default sort order or character set, see "Reconfiguring the character set, sort order, or message language" on page 355.

Setting the user's default language

If you install an additional language, users running client programs can run create login to set that language as their default language, or set the LANG variable on the client machine, with the appropriate entries in locales.dat.

Recovery after reconfiguration

Every time Adaptive Server is stopped and restarted, recovery is performed automatically on each database. Automatic recovery is discussed in detail in Chapter 11, "Developing a Backup and Recovery Plan," in the *System Administration Guide: Volume 2*.

After recovery is complete, the new sort order and character set definitions are loaded.

If you have changed the sort order, Adaptive Server switches to single-user mode to allow the necessary updates to system tables and to prevent other users from using the server. Each table with a character-based index is automatically checked to see if any indexes have been corrupted by the sort order change. Character-based indexes in tables are automatically rebuilt, if necessary, using the new sort order definition.

After the system indexes are rebuilt, character-based user indexes are marked "suspect" in the sysindexes system table. User tables with suspect indexes are marked "read-only" in sysobjects to prevent updates to these tables and use of the "suspect" indexes until they have been checked and, if necessary, rebuilt.

Range-partitioned user tables are checked for character-based partition keys, and are marked "suspect" if the sort order change or character set change might cause partition corruption.

Next, the new sort order information replaces the old information in the area of the disk that holds configuration information. Adaptive Server then shuts down so that it starts for the next session with a complete and accurate set of system information.

Using *sp_indsuspect* to find corrupt indexes

After Adaptive Server shuts down, restart it, and use sp_indsuspect to find the user tables that need to be reindexed.

    sp_indsuspect [*tab_name*]

where *tab_name* is the name of the table you are investigating. If *tab_name* is missing, sp_indsuspect creates a list of all tables in the current database that has indexes marked "suspect" when the sort order changes.

This example shows that running sp_indsuspect in mydb database yields one suspect index:

```
sp_indsuspect

Suspect indexes in database mydb
Own.Tab.Ind (Obj_ID, Ind_ID) =
dbo.holdings.h_name_ix(160048003, 2)
```

Rebuilding indexes after changing the sort order

dbcc reindex checks the integrity of indexes on user tables by running a "fast" version of dbcc checktable. For details, see Chapter 10, "Checking Database Consistency," in the *System Administration Guide: Volume 2*. dbcc reindex drops and rebuilds the indexes where the sort order used is not consistent with the new sort order. When dbcc reindex discovers the first index-related error, it displays a message, then rebuilds the inconsistent indexes. The system administrator or table owner should run dbcc reindex after changing the sort order in Adaptive Server.

dbcc reindex ({*table_name* | *table_id*})

Run this command on all tables listed by sp_indsuspect as containing suspect indexes. For example:

```
dbcc reindex(titles)

One or more indexes are corrupt. They will be rebuilt.
```

In the preceding example, dbcc reindex discovers one or more suspect indexes in the table titles; it drops and re-creates the appropriate indexes.

If the indexes for a table are already correct, or if there are no indexes for the table, dbcc reindex does not rebuild any indexes. It displays a message instead. If a table is suspected of containing corrupt data, the command is aborted. If that happens, an error message instructs the user to run dbcc checktable.

When dbcc reindex finishes successfully, all "suspect" marks on the table's indexes are removed. The "read-only" mark on the table is also removed, and the table can be updated. These marks are removed whether or not any indexes have to be rebuilt.

dbcc reindex does not reindex system tables. System indexes are checked and rebuilt, if necessary, as an automatic part of recovery after Adaptive Server is restarted following a sort order change.

Upgrading *text* data after changing character sets

If you have changed an Adaptive Server character set to a multibyte character set, use dbcc fix_text to upgrade text values.

A text value can be large enough to cover several pages; therefore, Adaptive Server must be able to handle characters that span page boundaries. To do so, Adaptive Server requires additional information on each of the text pages. The system administrator or table owner must run dbcc fix_text on each table that has text data to calculate the new values needed.

To see the names of all tables that contain text data, use:

```
select sysobjects.name
from sysobjects, syscolumns
where syscolumns.type = 35
and sysobjects.id = syscolumns.id
```

The system administrator or table owner must run dbcc fix_text to calculate the new values needed.

The syntax of dbcc fix_text is:

dbcc fix_text (*table_name* | *table_id*)

The table named must be in the current database.

dbcc fix_text opens the specified table, calculates the character statistics required for each text value, and adds the statistics to the appropriate page header fields. This process can take a long time, depending on the number and size of the text values in a table. dbcc fix_text can generate a large number of log records, which may fill up the transaction log. dbcc fix_text performs updates in a series of small transactions so that if a log becomes full, only a small amount of work is lost.

If you run out of log space, clear out your log (see Chapter 12, "Backing Up and Restoring User Databases," in *System Administration Guide: Volume 2*). Then restart dbcc fix_text, using the same table that was being upgraded when the original dbcc fix_text halted. Each multibyte text value contains information that indicates whether it has been upgraded, so dbcc fix_text upgrades only the text values that were not processed in earlier passes.

If your database stores its log on a separate segment, you can use thresholds to manage clearing the log. See Chapter 16, "Managing Free Space with Thresholds," in *System Administration Guide: Volume 2.*

If dbcc fix_text cannot acquire a needed lock on a text page, it reports the problem and continues with the work, like this:

```
Unable to acquire an exclusive lock on text page 408.
This text value has not been recalculated.  In order to
recalculate those TEXT pages you must release the lock
and reissue the dbcc fix_text command.
```

Retrieving *text* values after changing character sets

If you attempt to retrieve text values after changing to a multibyte character set, and you have not run dbcc fix_text, the command fails with this error message:

```
Adaptive Server is now running a multi-byte character
set, and this TEXT column's character counts have not
been recalculated using this character set. Use dbcc
fix_text before running this query again.
```

If you have changed the sort order or character set and errors occurred, see "How to Manually Change Sort Order or Default Character Set" in the *Adaptive Server Enterprise Troubleshooting and Error Messages Guide.*

Handling suspect partitions

Partitions are marked suspect for two reasons:

• A sort order or character set change on a range-partitioned table

• A cross-platform dump and load with a hash-partitioned table

If the table is marked with suspect partitions:

• All updates and cursor activities are suspended on this table.

• No alter table commands, except partition by, are allowed. create index and drop index are not allowed on a table with suspect partitions.

• The select command is allowed on tables containing suspect partitions. However, the optimizer treats such tables as round-robin partitioned tables, to avoid using the possibly corrupt partition condition.

Fixing tables with suspect partitions

- If the partition condition needs fixing after a sort-order change, you can use alter table with the partition by option to repartition a table that has suspect partitions.

- If the partition condition does not need fixing, you can use the reorg rebuild table command to rebuild the table, redistributing only the data rows among the partitions.

- If the indexes as well as the partitions on a table are marked suspect, use partition by or reorg rebuild to fix both the suspect index and suspect partitions.

Handling suspect partitions in cross-platform dump and load operations

- During the first online database command, after you execute load database across two platforms with different endian types, the hash partition is marked suspect.

- Any global clustered index on a round-robin partition, which has an internally generated partition condition with a unichar or varchar partition key, is marked suspect.

- After the database is online, use sp_post_xload to fix the suspect partitions and indexes.

Installing date strings for unsupported languages

Use sp_addlanguage to install names for the days of the week and months of the year for languages that do not have language modules. With sp_addlanguage, lets you define:

- A language name and (optionally) an alias for the name

- A list of the full names of months and a list of abbreviations for the month names

- A list of the full names of the days of the week

- The date format for entering dates (such as month/day/year)

- The number of the first day of the week

For example to add the information for Italian:

```
sp_addlanguage italian, italiano,
"gennaio,febbraio,marzo,aprile,maggio,giugno,luglio,agosto,settembre,ottobre,
novembre,dicembre",
"genn,feb,mar,apr,mag,giu,lug,ago,sett,ott,nov,dic",
"lunedi,martedi,mercoledi,giovedi,venerdi,sabato,domenica",
dmy, 1
```

sp_addlanguage enforces strict data entry rules. The lists of month names, month abbreviations, and days of the week must be comma-separated lists with no spaces or line feeds (returns). Also, they must contain the correct number of elements (12 for month strings, 7 for day-of-the-week strings.)

Valid values for the date formats are: mdy, dmy, ymd, ydm, myd, and dym. The dmy value indicates that the dates are in day/month/year order. This format affects only data entry; to change output format, you must use the convert function.

Server versus client date interpretation

Generally, date values are resolved on the client. When a user selects date values, Adaptive Server sends them to the client in an internal format. The client uses the *common.loc* file and other localization files in the default language subdirectory of the *locales* directory on the client to convert the internal format to character data. For example, if the user's default language is Spanish, Adaptive Server looks for the *common.loc* file in */locales/spanish/char_set*. It uses the information in the file to display, for example, 12 febrero 2007.

Assume that the user's default language is set to Italian, a language for which Adaptive Server does not provide a language module, and that the date values in Italian have been added. When the client connects to the server and looks for the *common.loc* file for Italian, it does not find the file. The client prints an error message and connects to the server. If the user then selects date values, the dates are displayed in U.S. English format.To display the date values added with sp_addlanguage, use the convert function to force the dates to be converted to character data at the server.

The following query generates a result set with the dates in U.S. English format:

```
select pubdate from titles
```

The query below, however, returns the date with the month names in Italian:

```
select convert(char(19),pubdate) from titles
```

Internationalization and localization files

Types of internationalization files

The files that support data processing in a particular language are called internationalization files. Several types of internationalization files come with Adaptive Server.

Table 10-9: Internationalization files

File	Location	Purpose and contents
charset.loc	In each character set subdirectory of the *charsets* directory	Character set definition files that define the lexical properties of each character, such as alphanumeric, punctuation, operand, and uppercase or lowercase. Used by Adaptive Server to correctly process data.
*.srt	In each character set subdirectory of the *charsets* directory	Defines the sort order for alphanumeric and special characters, including ligatures, diacritics, and other language-specific considerations.
*.xlt	In each character set subdirectory of the *charsets* directory	Terminal-specific character translation files for use with utilities such as bcp and isql. For more information about how the *.xlt* files are used, see Chapter 11, "Configuring Client/Server Character Set Conversions," and the *Utility Guide*.

> **Warning!** Do not alter any of the internationalization files. If you need to install a new terminal definition or sort order, contact your local Sybase office or distributor.

Character sets directory structure

Figure 10-3 shows the directory structure for the Western European character sets that come with Adaptive Server. There is a separate subdirectory for each character set in the *charsets* directory. Within the subdirectory for each character set (for example, *cp850*) are the character set and sort order definition files and terminal-specific files.

If you load additional character sets, they also appear in the *charsets* directory:

Figure 10-3: Structure of the charsets directory

```
┌──────────────┐
│   charsets   │
└──────┬───────┘
       │
  ┌────┴────┬─────────┬─────────┬─────────┬──────────────┐
┌─┴───┐  ┌──┴──┐  ┌───┴───┐  ┌──┴──┐  ┌───┴────┐  ┌─────┴──────┐
│cp850│  │iso_1│  │ cp437 │  │ mac │  │ roman8 │  │charset_name│
└─────┘  └─────┘  └───────┘  └─────┘  └────────┘  └────────────┘
```

charset.loc   charset.loc   charset.loc   charset.loc   charset.loc   charset.loc
*.srt   *.srt   *.srt   *.srt   *.srt   *.srt
                *.xlt
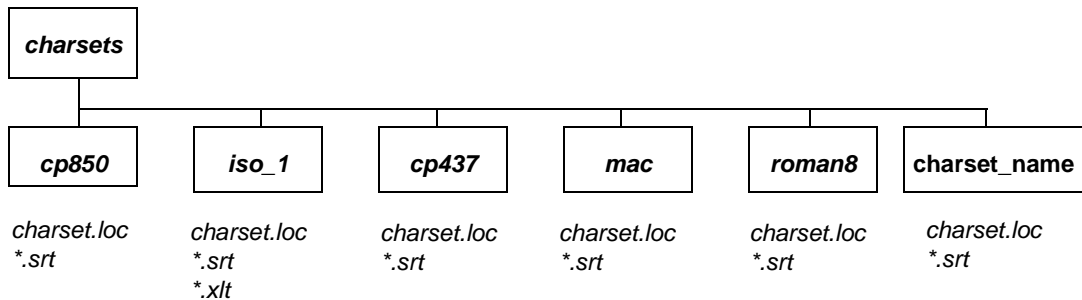
The following global variables contain information about character sets:

- @@*char_convert* – contains 0 if character set conversion is not in effect. Contains 1 if character set conversion is in effect.

- @@*client_csname* – the client's character set name. Set to NULL if client character set has never been initialized; otherwise, contains the name of the character set for the connection.

- @@*client_csid* – the client's character set ID. Set to -1 if client character set has never been initialized; otherwise, contains the client character set ID from syscharsets for the connection.

- @@*client_csexpansion* – returns the expansion factor used when converting from server's character set to client's character set.

- @@*maxcharlen* – the maximum length, in bytes, of a character in the Adaptive Server default character set.

- @@*ncharsize* – the maximum length, in bytes, of a character set in the current server default character set.

- @@*unicharsize* – equals 2.

Types of localization files

Adaptive Server includes several localization files for each Language Module:

- *locales.dat* – in the *locales* directory. Used by client applications to identify the default message language and character set.

- • *server.loc* – in the character set subdirectories under each language subdirectory in the *$SYBASE/$SYBASE_ASE/locales* directory. Software messages translated into the local language. Sybase products have product-specific **.loc* files. If an entry is not translated, the software message or string appears in U.S. English instead of the local language.

- • *common.loc* – in each language and character set directory of the *locales* directory. Contains the local names of the months of the year and their abbreviations, and information about the local date, time, and money formats.

All Adaptive Server-related locales files (used by dataserver, sqlloc, syconfig, and so on) are in *$SYBASE/SYBASE_ASE/locales*. All Open Client/Server-related locales files (ctlib, ctisql, ctbcp, optdiag, installjava, and so on) are located in *$SYBASE/locales*.

Warning! Do not alter any of the localization files. If you need to alter any information in those files, contact your local Sybase office or distributor.

Software messages directory structure

Figure 10-4 shows how localization files are arranged. Within the *locales* directory is a subdirectory for each language installed. There is always a *us_english* subdirectory. (On PC platforms, this directory is called *english*.) During installation, when you are prompted to select the languages you want installed on Adaptive Server, the installation program lists the supported software message languages. If you install language modules for additional languages, you see subdirectories for those languages. Within each language subdirectory are subdirectories for the supported character sets; for example, *cp850* is a supported character set for *us_english*. Software message files for each Sybase product reside in the character set subdirectories.

Figure 10-4: Messages directory structure
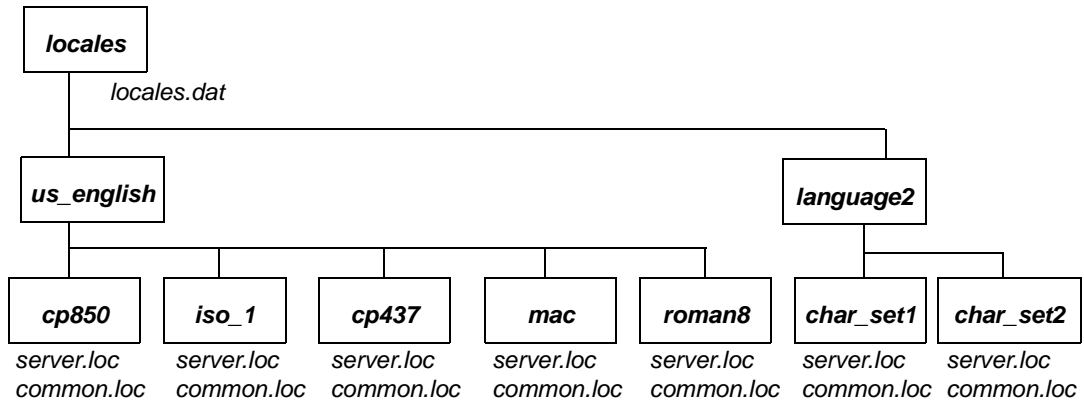
Message languages and global variables

The following global variables contain information about languages:

- @@*langid* – contains the local language ID of the language currently in use (specified in syslanguages.langid)

- @@*language* – contains the name of the language currently in use (specified in syslanguages.name)

Adaptive Server Enterprise

Topic	Page
Character set conversion	369
Supported character set conversions	370
Choosing a conversion type	374
Enabling and disabling character set conversion	376
Error handling in character set conversion	377
Conversions and changes to data lengths	378
Specifying the character set for utility programs	379
Display and file character set command line options	379

Character set conversion

In a heterogeneous environment, Adaptive Server may need to communicate with clients running on different platforms using different character sets. Although different character sets may support the same language group (for example, ISO 8858-1 and CP 850 support the group 1 languages), they may encode the same characters differently. For example, in ISO 8859-1, the character *à* is encoded as *0xE0* in hexadecimal. However, in CP 850 the same character is encoded as *0x85* in hexadecimal.

To maintain data integrity between your clients and servers, data must be converted between the character sets. The goal is to ensure that an "a" remains an "a" even when crossing between machine and character set boundaries. This process is known as **character set conversion**.

Supported character set conversions

Character set conversion occurs between a pair of character sets. The supported conversions in any particular client/server system depend on the character sets used by the server and its clients. One type of character set conversion occurs if the server uses a native character set as the default; a different type of conversion is used if the server default is Unicode UTF-8.

Conversion for native character sets

Adaptive Server supports character set conversion between native character sets belonging to the same language group. If the server has a native character set as its default, the clients' character sets must belong to the same language group. Figure 11-1 is an example of a Western European client/server system. In this example, the clients' character sets and the Adaptive Server default character set all belong to language group 1. Data is correctly converted between the client character sets and the server default character set. Since they all belong to the same language group, the clients can view all data on the server, no matter which client submitted the data.

Figure 11-1: Character set conversion when server and client character sets belong to the same language group
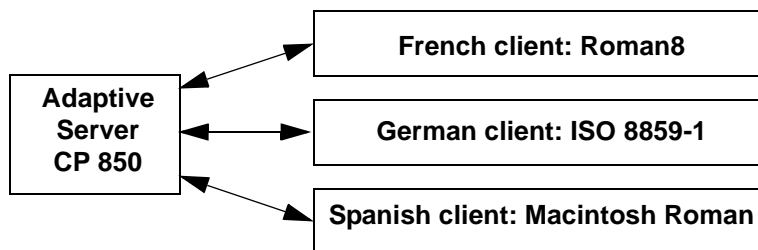
For a list of the language groups and supported character sets, see Table 10-1 on page 333.

Conversion in a Unicode system

Adaptive Server also supports character set conversion between UTF-8 and any native character set that Sybase supports. In a Unicode system, since the server default character set is UTF-8, the client character set may be a native character set from any language group. Therefore, a Japanese client (group 101), a French client (group 1), and an Arabic client (group 6) can all send and receive data from the same server. Data from each client is correctly converted as it passes between each client and the server.

Figure 11-2: Character set conversion in a Unicode system
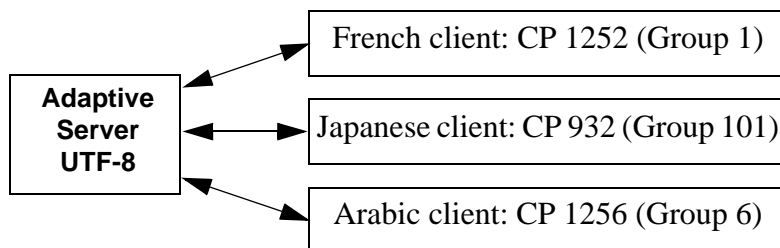
Each client can view data only in the language supported by its character set. Therefore, the Japanese client can view any Japanese data on the server, but it cannot view Arabic or French data. Likewise, the French client can view French or any other Western European language supported by its character set, but not Japanese or Arabic.

Figure 11-3: Viewing Unicode data
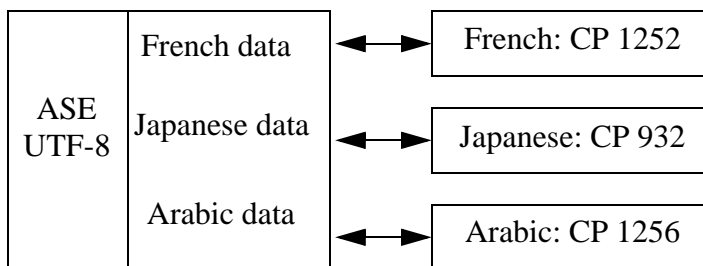
An additional character set, ASCII-7, is a subset of *every* character set, including Unicode, and is therefore compatible with all character sets in all language groups. If either the Adaptive Server or the client's character set is ASCII-7, any 7-bit ASCII character can pass between the client and server unaltered and without conversion.

Sybase recommends that you do not configure a server for ASCII-7. You can achieve the same benefits of compatibility by restricting each client to use only the first 128 characters of each native character set.

Adaptive Server direct conversions

Adaptive Server direct conversions are between two native character sets of the same language group. For example, Adaptive Server supports conversion between CP 437 and CP 850, because both belong to language group 1. Adaptive Server direct conversions exist between many, but not all, native character sets of a language group (see Table 11-1 on page 375).

Unicode conversions

Unicode conversions exists for all native character sets. When converting between two native character sets, Unicode conversion uses Unicode as an intermediate character set. For example, to convert between the server default character set (CP 437), and the client character set (CP 860), CP 437 is first converted to Unicode; Unicode is then converted to CP 860.

Unicode conversions may be used either when the default character set of the server is UTF-8, or a native character set. You must specifically configure your server to use Unicode conversions (unless the server's default character set is UTF-8).

Earlier versions of Adaptive Server used direct conversions, and it is the default method for character set conversions. However, Unicode conversions allow easier and less complex character set conversion. While Adaptive Server direct conversions are still supported, Sybase now also uses Unicode conversions to provide complete conversion support for all character sets and has no plans to add new direct conversions.

Allowing Unicode noncharacters

In versions of Adaptive Server earlier than 15.7, the unichar, univarchar, unitext, char, varchar, and text datatypes under the utf-8 default character set did not accept Unicode noncharacters (code points permanently reserved for internal use).

Adaptive Server version 15.7 allows you to ignore Unicode noncharacters by enabling the enable functionaliry group or the enable permissive unicode configuration parameters.

If you do not enable this feature, Adaptive Server rejects these noncharacters as in earlier versions.

For more information about the Unicode standard, see the Unicode Consortium Web site at http://www.unicode.org/.

When you enable this feature, Unicode noncharacters are not detected in:

- Parameters:

    - Presented as univarchar and unitext (UTF-16) datatypes

    - Presented as varchar and text (UTF-8) datatypes

    - As parameters to dynamic SQL statements

    - As input to parameterized language statements

- String literals when the server's character set is UTF-8

- Escaped string literals (those prefixed with U&), regardless of the server's character set

- Conversion processes between unichar (UTF-16) and varchar (UTF-8) in either direction

In addition, Unicode noncharacters are acceptable in simple expressions such as comparisons, where they sort higher than legal Unicode characters.

You can use Unicode noncharacters as parameters to these functions:

ascii()	lower()	sortkey()
char_length()	ltrim()	soundex()
charindex()	patindex()	str_replace()
compare()	replicate()	stuff()
datalength()	reverse()	substring()
difference()	right()	upper()
left()	rtrim()	uscalar()
len()		

Note This feature does not affect UTF-16 surrogate handling enabled with the enable surrogate handling configuration parameter. See "Setting Configuration Parameters" in the *System Administration Guide*.

Choosing a conversion type

To determine the conversion options that are available for your client/server system, see Table 11-1 on page 375.

Non-Unicode client/server systems

In a non-Unicode system, the character sets of the server and clients are native character sets; therefore, you can use the Adaptive Server direct conversions.

However, there are some character sets for which there is no Adaptive Server direct conversion; in this situation, you must use Unicode conversions.

- If all character sets in your client/server system are column 1 of Table 11-1, use the Adaptive Server direct conversions. The character sets must all belong to the same language group.

- If the character sets in your client/server system are in column 2 of Table 11-1, or some combination of columns 1 and 2, configure your server to use Unicode conversions. Again, the character sets must all belong to the same language group.

For example, assume the server default character set is CP 850 and the clients' character sets are either ISO 8859-1 or ROMAN 8. Table 11-1 shows that direct conversions exist between CP 850 and the client character sets. Now, suppose you add a client using CP 1252. Since there is no direct conversion between CP 1252 and CP 850, (the default server character set), you must use Unicode conversions to convert between CP 1252 and CP 850. When you have a mixture of character sets—some where you can use Adaptive Server direct conversions and others where you must use Unicode conversions—you can specify that a combination of Adaptive Server direct conversion and Unicode conversion be used.

Unicode client/server systems

If your server default is Unicode UTF-8, then all conversions are between UTF-8 and the native character set being used on the client systems. In a Unicode system, Unicode conversions are used exclusively.

Table 11-1: Conversion methods for character sets

Language group	Column 1 – Adaptive Server direct conversions and Unicode conversions	Column 2 – Unicode conversions only
Group 1	CP 437, CP 850, ISO 8859-1, Macintosh Roman	CP 860, CP 1252, ISO 8859-15, CP 863
Group 2	CP 852, CP 1250, CP 8859-1, Macintosh Central European	ISO 8859-2
Group 4	No conversions needed (only one character set supported)	
Group 5	CP 855, CP 866, CP 1251, ISO 8859-5, Koi8, Macintosh Cyrillic	
Group 6		CP 864, CP 1256, ISO 8859-6
Group 7	CP 869, CP 1253, GREEK8, ISO 8859-7, Macintosh Greek	
Group 8		CP 1255, ISO 8859-8
Group 9	CP 857, CP 1254, ISO 8859-9, Macintosh Turkish, TURKISH8	
Group 101	DEC Kanjii, EUC-JIS, Shift-JIS	CP 932
Group 102		CP 936, EUG-GB, GB18303
Group 103		Big 5, CP 950, EUC-CNS
Group 104		EUCKSC, CP 949
Group 105		CP 874, TIS 620
Group 106	No conversions needed (only one character set supported)	
Unicode	No conversions needed (only one character set supported)	

Configuring the server

By default, Adaptive Server uses direct conversions to convert data between different character sets. To use the Unicode conversions, Use sp_configure to set the enable unicode conversions option to either 1 or 2.

- If you set sp_configure "enable unicode conversions" to 1:

  This setting uses Adaptive Server direct conversions or Unicode conversions. Adaptive Server first checks to see if an Adaptive Server direct conversion exists for the server and client character set. If a direct conversion is used; if no direct conversion exists, the Unicode conversion is used.

  Use this setting if the character sets in your client/server system fall into both columns 1 and 2 in Table 11-1.

- If you set sp_configure "enable unicode conversions" to 2:

  This setting uses Unicode conversions only. Adaptive Server uses Unicode conversions, without attempting to find an Adaptive Server direct conversion.

Use this setting if the client/server conversions result in a change in the data length (see "Conversions and changes to data lengths" on page 378)

If all character sets fall into column 2 in Table 11-1, set enable unicode conversions to 2 to always use Unicode conversions.

For Adaptive Server version 15.0 and later, the default value for enable unicode conversions is 1.

If the server default is UTF-8, the server automatically uses Unicode conversions only.

Enabling and disabling character set conversion

A client that is requesting a connection identifies its character set to Adaptive Server. Adaptive Server compares the client character set with its default character set, and if the two names are identical, no conversion is required. If the names differ, Adaptive Server determines whether it supports conversion between its default and the client's character set. If it does not, it sends an error message to the client and continues with the login process. If it does, character set conversion is automatically enabled. If the default character set of the server is UTF-8, Unicode conversions are automatically used. If the default is a native character set, the server uses Adaptive Server direct conversions, unless the user requests Unicode conversions.

You can disable character set conversion at the server level. You may want to do this if:

*   All of your clients are using the same character set as the server default, and therefore, no conversion is required.

*   Conversion between the client character set and the server default is not supported.

*   You want to store data in the server without changing the encoding.

To disable character set conversion at the server level, set the disable character set conversion parameter to 1.

You can control character set conversion at the connection level using the set char_convert command from within a client session. set char_convert off turns conversion off between a particular client and the server. You may want to set char_convert off if the client and the server use the same character set, which makes conversion unnecessary. set char_convert on turns conversion back on.

Characters that cannot be converted

Some characters may not be converted, if:

*   The character exists (is encoded) in the source character set, but does not exist in the target character set. For example, the OE ligature is part of the Macintosh character set (code point 0xCE). This character does not exist in the ISO 8859-1 character set. If the OE ligature exists in data that is being converted from the Macintosh to the ISO 8859-1 character set, it causes a conversion error.

*   The character exists in both the source and the target character set, but in the target character set, the character is represented by a different number of bytes than in the source character set.

    For example, 1-byte accented characters (such as á, è) are 2-byte characters in UTF-8; 2-byte Thai characters are 3-byte characters in UTF-8. Avoid this limitation by configuring the enable unicode conversion option to 1 or 2.

Error handling in character set conversion

The Adaptive Server character set conversion reports errors when a character exists in the client's character set but not in the server's character set, or vice versa. Adaptive Server must guarantee that data successfully converted on input to the server can be successfully converted back to the client's character set when the client retrieves that data. To do this effectively, Adaptive Server must avoid putting suspect data into the database.

When Adaptive Server encounters a conversion error in the data being entered, it generates this message:

```
Msg 2402, Severity 16 (EX_USER):
Error converting client characters into server's
character set. Some character(s) could not be converted.
```

A conversion error prevents query execution on insert and update statements. If this occurs, review your data for problem characters and replace them.

When Adaptive Server encounters a conversion error while sending data to the client, it replaces the bytes of the suspect characters with ASCII question marks (?). The query batch continues to completion. When the statement is complete, Adaptive Server sends the following message:

```
Msg 2403, Severity 16 (EX_INFO):
WARNING! Some character(s) could not be converted into
client's character set. Unconverted bytes were changed
to question marks ('?').
```

Conversions and changes to data lengths

In some cases, converting data between the server's character set and the client's character set results in a change to the length of the data, for example, when the character set on one system uses one byte to represent each character and the character set on the other system requires two bytes per character.

When character set conversion results in a change in data length, there are two possibilities:

- The data length decreases, as in the following examples:

    - Greek or Russian in multibyte UTF-8 to a single-byte Greek or Russian character set

    - Japanese two-byte Hankaku Katakana characters in EUC-JIS to single-byte characters in Shift-JIS

- The data length increases, as in the following examples:

    - Single-byte Thai to multibyte Thai in UTF-8

    - Single-byte Japanese characters in Shift-JIS to two-byte Hankaku Katakana in EUC-JIS

Configuring your system and application

If you are using UTF-8 anywhere in your client/server system, or using a Japanese character set, you are likely to encounter changes in data length as a result of character set conversion. You must configure your server to handle changes in data length. You may also need to set up your client to handle changes in data length.

1   Configure the server to use Unicode conversions. See "Configuring the server" on page 375. If the data length increases between the server and the client, you must also complete steps 2 and 3.

2   The client must be using Open Client 11.1 or later. It must inform the server that it can handle CS_LONGCHAR data at connection time, using the Open Client ct_capability function.

The *capability* parameter must be set to CS_DATA_LCHAR and the *value* parameter must be set to CS_TRUE, where *connection* is a pointer to a CS_CONNECTION structure:

```
CS_INT capval = CS_TRUE
ct_capability(connection,CS_SET,CS_CAP_RESPONS,
    CS_DATA_LCHAR,&capval)
```

3   When conversions result in an increase in data length, char and varchar data are converted to the client's character set and are sent to the client as CS_LONGCHAR data. The client application must be coded to extract the data received as CS_LONGCHAR.

Specifying the character set for utility programs

The Sybase utility programs assume that the default character set of the client platform is the same character set the client is using. However, sometimes the client character set differs from the character set for the platform. For this reason, you may need to specify the client character set at the command line. A command line option for the isql, bcp, and defncopy utilities specifies the client's character set, and temporarily overrides settings of the LANG variable or settings in *locales.dat*.

-J *charset_name* (UNIX and PC) sets the client's character set to the *charset_name*.

If yo omit the client character set's command line flag, the platform's default character set is used. See the *Utility Guide*.

Display and file character set command line options

Although the focus of this chapter is on character set conversion between clients and Adaptive Server, there are two other places where you may need character set conversion:

• Between the client and a terminal

• Between the client and a file system

Figure 11-4 illustrates the paths and command line options that are available in the standalone utilities isql, bcp, and defncopy.

Figure 11-4: Where character set conversion may be needed
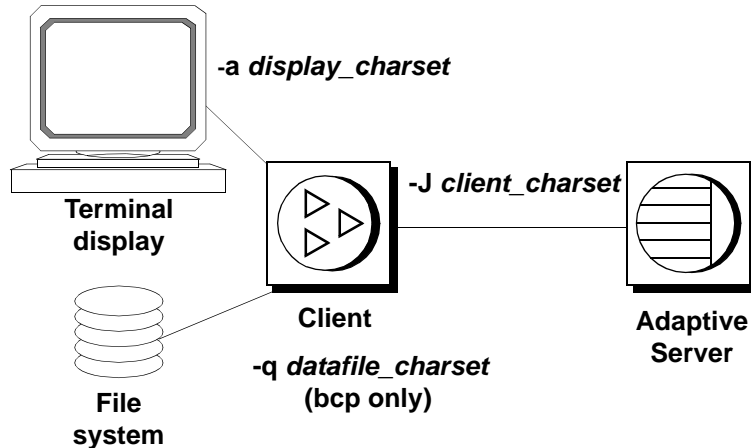
The -J or /clientcharset command line option specifies the character set used by the client when it sends and receives character data to and from Adaptive Server.

Setting the display character set

Use the -a command line option if you are running the client from a terminal with a character set that differs from the client character set. In Figure 11-4, the -a option and the -J option are used together to identify the character set translation file (*.xlt* file) needed for the conversion.

Use -a without -J only if the client character set is the same as the default character set.

Setting the file character set

Use the -q command line option if you are running bcp to copy character data to or from a file system that uses a character set that differs from the client character set. In Figure 11-4, use the -q or /filecharset option and the -J or /clientcharset option together to identify the character set translation file (*.xlt* file) needed for the conversion.

This chapter discusses diagnosing and fixing system problems.

How Adaptive Server uses error messages

When Adaptive Server encounters a problem, it displays an error message that includes:

- A **message number**, which uniquely identifies the error message

- A **severity level number** between 10 and 24, which indicates the type and severity of the problem

- An **error state number**, which allows unique identification of the line of Adaptive Server code at which the error was raised

- An **error message**, which tells you what the problem is, and may suggest how to fix it

See the *Configuration Guide* for your platform for a description of the error log format.

For example, if you try to access a table that does not exist, you see:

```
select * from publisher

Msg 208, Level 16, State 1:
publisher not found. Specify owner.objectname or use
sp_help to check whether the object exists (sp_help
```

```
                        may produce lots of output).
```

There may be more than one error message for a single query. If there is more than one error in a batch or query, Adaptive Server usually reports only the first one. Subsequent errors are reported the next time you execute the batch or query.

Error messages are stored in master..sysmessages, which is updated with each new version of Adaptive Server (and has thousands of rows). Here are the first few rows (from an Adaptive Server that uses us_english as the default language):

```
                        select error, severity, description
                        from sysmessages
                        where error >=101 and error <=106
                        and langid is null
```

```
error severity description
----- -------- -------------------------------------------------
  101        15 Line %d: SQL syntax error.
  102        15 Incorrect syntax near '%.*s'.
  103        15 The %S_MSG that starts with '%.*s' is too long.
                Maximum length is %d.
  104        15 Order-by items must appear in the select-list if
                the statement contains set operators.
  105        15 Unclosed quote before the character string '%.*s'.
  106        16 Too many table names in the query. The maximum
                allowable is %d.
```

```
(6 rows affected)
```

You can query sysmessages to generate a custom list of error messages:

- If your server supports more than one language, sysmessages stores each message in each language. The column langid is NULL for us_english and matches the syslanguages.langid for other languages installed on the server.

- The sqlstate column stores the SQLSTATE value for error conditions and exceptions defined in ANSI SQL92.

- Message numbers 17000 and higher are system procedure error messages and message strings.

Error log format

Adaptive Server uses this format for error messages:

```
instance_ID:thread_ID:family_ID:spid date time {server | kernel} message
```

where:

- *instance_ID* – the instance running in a clustered environment. This value is 00 if Adaptive Server is running in a nonclustered environment.

- *thread_ID* – the thread on which the error occured. Always apears as a four-digit number (for example, 10 is shown as 0010).

> **Note** In process mode, Adaptive Server displays the engine involved for each log entry. The engine number is indicated by a four-digit number. If only one engine is online, it is indicated by 00.

- *family_ID* – family ID of the originating thread:
  - In serial processing, you see 00000.
  - In parallel processing, you see the server process ID number of the parent of the originating thread.
- *spid* – the server process ID of the originating thread:
  - In serial processing, this is the server process ID number of the thread that generated the message. If the thread is a system task, the spid value is 00000.
  - In parallel processing, you see the server process ID number of the originating thread.
- *date* – appeared in the format yyyy/mm/dd, which allows you to sort error messages by date.
- *time* – shown in 24-hour format, which includes seconds and hundredths of a second.
- *server | kernel* – this entry is for use only by Sybase Technical Support.
- *message* – the error message

For example:

```
00:0024:00000:00000:2010/04/27 10:28:07.82 kernel Thread 24 (LWP 24390) of
Threadpool syb_default_pool online as engine 0
```

- *instance_ID* = 0 – the server is not configured for a clustered environment
- *thread_ID* = 0024 – Adaptive Server brought engine 0 online as thread number 24.
- *family_ID* = 00000 – the process is running in serial mode.

- *spid* = 00000 – this is a system task.

- *date* and *time* = 2010/04/27 10:28:07.82.

- *server | kernel* = kernel.

Error messages and message numbers

The combination of message number (error) and language ID (langid) uniquely identifies each error message. Messages that share the same message number but have different language IDs indicate translations.

```
select error, description, langid
from sysmessages
where error = 101

error description                                 langid
----- ------------------------------------------ ------
  101 Line %d: SQL syntax error.                    NULL
  101 Ligne %1!: erreur de syntaxe SQL.               1
  101 Zeile %1!: SQL Syntaxfehler.                    2

(3 rows affected)
```

The error message text describes the problem. The descriptions often include a line number, a reference to a type of database object (a table, column, stored procedure, and so forth), or the name of a particular database object.

In the description field of sysmessages, a percent sign (%) followed by a character or character string serves as a placeholder for these pieces of data, which Adaptive Server supplies when it encounters the problem and generates the error message. "%d" is a placeholder for a number; "%S_MSG" is a placeholder for a type of database object; "%.*s"—all within quotes—is a placeholder for the name of a particular database object. Table 12-1 on page 385 lists placeholders and what they represent.

For example, the description field for message number 103 is:

```
The %S_MSG that starts with '%.*s' is too long. Maximum
length is %d.
```

The actual error message that appears to a user might be:

```
The column that starts with 'title' is too long. Maximum
length is 80.
```

For errors that you report to Technical Support, include the numbers, object types, and object names. (See "Reporting errors" on page 392.)

Variables in error message text

Table 12-1 explains the symbols that appear in the text provided with each error message explanation:

Table 12-1: Error text symbols key

Symbol	Stands for
%d, %D	Decimal number
%x,%X,%.*x,%lx, %04x, %08lx	Hexadecimal number
%s	Null-terminated string
%.*s, %*s, %*.s	String, usually the name of a particular database object
%S_*type*	Adaptive Server-defined structure
%c	Single character
%f	Floating-point number
%ld	Long decimal
%lf	Double floating-point number

Adaptive Server error logging

Error messages from Adaptive Server are sent only to the user's screen.

The stack trace from fatal error messages (severity levels 19 and higher) and error messages from the kernel are sent to an error log file. The name of this file varies; see the configuration documentation for your platform or the *Utility Guide*.

Note The error log file is owned by the user who installed Adaptive Server (or the person who started Adaptive Server after an error log was removed). Permissions or ownership problems with the error log at the operating system level can block successful start-up of Adaptive Server.

Adaptive Server creates an error log for you if one does not already exist. Specify the location of the error log at start-up with the *errorlogfile* parameter in the runserver file or at the command line. The Sybase installer utility configures the runserver file with *$SYBASE/$SYBASE_ASEinstall* as the location of the error log if you do not choose an alternate location. If you do not specify the location in the runserver file or at the command line, the location of the error log is the directory from which you start Adaptive Server. For more information about specifying the location of the error log, see dataserver in the *Utility Guide*.

Note Always start Adaptive Server from the same directory, or with the runserver file or the error log flag, so that you can locate your error log.

Each time you start a server, messages in the error log provide information on the success (or failure) of the start and the recovery of each database on the server. Subsequent fatal error messages and all kernel error messages are appended to the error log file. To reduce the size of the error log by deleting old or unneeded messages, "prune" the log while Adaptive Server is shut down.

Severity levels

The severity level of a message indicates the type and severity of the problem that Adaptive Server has encountered. For maximum integrity, when Adaptive Server responds to error conditions, it displays messages from sysmessages, but takes action according to an internal table. A few corresponding messages differ in severity levels, so you may occasionally notice a difference in expected behavior if you are developing applications or procedures that refer to Adaptive Server messages and severity levels.

Warning! You can create your own error numbers and messages based on Adaptive Server error numbers (for example, by adding 20,000 to the Adaptive Server value). However, you cannot alter the Adaptive Server-supplied system messages in the sysmessages system table.

You can add user-defined error messages to sysusermessages with sp_addmessage. See the *Reference Manual: Procedures*.

Users should inform the system administrator whenever problems that generate severity levels of 17 and higher occur. The system administrator is responsible for resolving them and tracking their frequency.

If the problem has affected an entire database, the system administrator may have to use the database consistency checker (dbcc) to determine the extent of the damage. The dbcc may identify some objects that have to be removed. It can repair some damage, but you may have to reload the database.

For more information, see the following chapters in *System Administration Guide: Volume 2*:

- dbcc is discussed in Chapter 10, "Checking Database Consistency," in the *System Administration Guide: Volume 2*.

- Loading a user database is discussed in Chapter 12, "Backing Up a and Restoring User Databases," in *System Administration Guide: Volume 2*

- Loading system databases is discussed in Chapter 13, "Restoring the System Databases," in *System Administration Guide: Volume 2*.

Severity levels 10 – 18

Error messages with severity levels 10–16 are generated by problems that are caused by user errors. These problems can be corrected by the user. Severity levels 17 and 18 do not terminate the user's session.

Error messages with severity levels 17 and higher should be reported to the system administrator or database owner.

Level 10: Status information

Messages with severity level 10 are not errors at all. They provide additional information after certain commands have been executed and, typically, do not display the message number or severity level. For example, after a create database command, Adaptive Server displays a message telling the user how much of the requested space has been allocated for the new database.

Level 11: Specified database object not found

Messages with severity level 11 indicate that Adaptive Server cannot find an object that is referenced in a command.

This is often because the user has made a mistake in typing the name of a database object, because the user did not specify the object owner's name, or because of confusion about which database is current. Check the spelling of object names, use the owner names if the object is not owned by the user or "dbo," and make sure you are in the correct database.

Level 12: Wrong datatype encountered

Messages with severity level 12 indicate a problem with datatypes. For example, the user may have tried to enter a value of the wrong datatype in a column or to compare columns of different and incompatible datatypes.

To correct comparison problems, use the convert function with select. See the *Reference Manual: Building Blocks* or the *Transact-SQL Users Guide*.

Level 13: User transaction syntax error

Messages with severity level 13 indicate that something is wrong with the current user-defined transaction. For example, the user may have issued a commit transaction command without having issued a begin transaction, or they may have tried to roll back a transaction to a savepoint that has not been defined (sometimes there may be a typing or spelling mistake in the name of the savepoint).

Severity level 13 can also indicate a deadlock, in which case the deadlock victim's process is rolled back. The user must restart his or her command.

Level 14: Insufficient permission to execute command

Messages with severity level 14 mean that the user does not have the necessary permission to execute the command or access the database object. they can ask the owner of the database object, the owner of the database, or the system administrator to grant them permission to use the command or object in question.

Level 15: Syntax error in SQL statement

Messages with severity level 15 indicate that the user has made a mistake in the syntax of the command. The text of these error messages includes the line numbers on which the mistake occurs and the specific word near which it occurs.

Level 16: Miscellaneous user error

Most error messages with severity level 16 reflect that the user has made a nonfatal mistake that does not fall into any of the other categories. Severity level 16 and higher can also indicate software or hardware errors.

For example, the user may have tried to update a view in a way that violates the restrictions. Another error that falls into this category is unqualified column names in a command that includes more than one table with that column name. Adaptive Server has no way to determine which one the user intends. Check the command syntax and working database context.

Messages that ordinarily have severities greater than 16 show severity 16 when they are raised by dbcc checktable or dbcc checkalloc so that checks can continue to the next object. When you are running the dbcc utility, check the *Error Messages and Troubleshooting Guide* for information about error messages between 2500 and 2599 with a severity level of 16.

Note Levels 17 and 18 are usually not reported in the error log. Users should be instructed to notify the system administrator when level 17 and 18 errors occur.

Level 17: Insufficient resources

Error messages with severity level 17 mean that the command has caused Adaptive Server to run out of resources or to exceed some limit set by the system administrator. The user can continue with their work, although they may not be able to execute a particular command.

These system limits include the number of databases that can be open at the same time and the number of connections allowed to Adaptive Server. They are stored in system tables and can be checked with sp_configure. See Chapter 5, "Setting Configuration Parameters," for more information on changing configuration variables.

The database owner can correct the level 17 error messages indicating that the user has run out of space. Other level 17 error messages should be corrected by the system administrator.

Level 18: Nonfatal internal error detected

Error messages with severity level 18 indicate an internal software bug. However, the command runs to completion, and the connection to Adaptive Server is maintained. The user can continue with the work they are doing, although they may not be able to execute a particular command. An example of a situation that generates severity level 18 is Adaptive Server detecting that a decision about the access path for a particular query has been made without a valid reason.

Since problems that generate such messages do not keep users from their work, users tend not to report them. However, users should be instructed to inform the system administrator every time an error message with this severity level (or higher) occurs so that the system administrator can report them.

Severity levels 19 – 26

Fatal problems generate error messages with severity levels 19 and higher. They break the user's connection to Adaptive Server (some of the higher severity levels shut down Adaptive Server). To continue working, the user must restart the client program.

When a fatal error occurs, the process freezes its state before it stops, recording information about what has happened. The process is then killed and disappears.

When the user's connection is broken, he or she may or may not be able to reconnect and resume working. Some problems with severity levels in this range affect only one user and one process. Others affect all the processes in the database. In some cases, the system administrator must restart Adaptive Server. These problems do not necessarily damage a database or its objects, but they can. They may also result from earlier damage to a database or its objects. Other problems are caused by hardware malfunctions.

Error messages from the kernel are directed to the error log file.

Level 19: Adaptive Server fatal error in resource

Error messages with severity level 19 indicate that some nonconfigurable internal limit has been exceeded and that Adaptive Server cannot recover gracefully. You must reconnect to Adaptive Server.

Level 20: Adaptive Server fatal error in current process

Error messages with severity level 20 indicate that Adaptive Server has encountered a bug in a command. The problem has affected only the current process, and the database is unlikely to have been damaged. Run dbcc diagnostics. The user must reconnect to Adaptive Server.

Level 21: Adaptive Server fatal error in database processes

Error messages with severity level 21 indicate that Adaptive Server has encountered a bug that affects all the processes in the current database. However, it is unlikely that the database itself has been damaged. Restart Adaptive Server and run dbcc diagnostics. The user must reconnect to Adaptive Server.

Level 22: Adaptive Server fatal error: Table integrity suspect

Error messages with severity level 22 indicate that the table or index specified in the message has been previously damaged by a software or hardware problem.

The first step is to restart Adaptive Server and run dbcc to determine whether other objects in the database are also damaged. Whatever the report from dbcc may be, The problem may be only in the cache, and not on the disk itself. If so, restarting Adaptive Server fixes the problem.

If restarting does not help, then the problem is on the disk as well. Sometimes, the problem can be solved by dropping the object specified in the error message. For example, if the message tells you that Adaptive Server has found a row with length 0 in a nonclustered index, the table owner can drop the index and re-create it.

Adaptive Server takes any pages or indexes offline that it finds to be suspect during recovery. Use sp_setsuspect_granularity to determine whether recovery marks an entire database or only individual pages as suspect. See sp_setsuspect_granularity in the *Reference Manual: Procedures*.

The user must reconnect to Adaptive Server.

Level 23: Fatal error: Database integrity suspect

Error messages with severity level 23 indicate that the integrity of the entire database is suspect due to previous damage caused by a software or hardware problem. Restart Adaptive Server and run dbcc diagnostics.

Even when a level 23 error indicates that the entire database is suspect, the damage may be confined to the cache, and the disk itself may be fine. If so, restarting Adaptive Server with startserver fixes the problem.

Level 24: Hardware error or system table corruption

Error messages with severity level 24 reflect a media failure or (in rare cases) the corruption of sysusages. The system administrator may have to reload the database. You may need to call your hardware vendor.

Level 25: Adaptive Server internal error

Users do not see level 25 errors; this level is used only for Adaptive Server internal errors.

Level 26: Rule error

Error messages with severity level 26 reflect that an internal locking or synchronization rule has been broken. You must shut down and restart Adaptive Server.

Reporting errors

When you report an error to Sybase Technical Support, include:

- The message number, level number, and state number.

- Any numbers, database object types, or database object names that are included in the error message.

- The context in which the message was generated, that is, the command that was running at the time. You can help by providing a hard copy of the error log.

Backup Server error logging

Like Adaptive Server, Backup Server creates an error log if one does not already exist. Specify the location of the error log at start-up with the *error_log_file* parameter in the runserver file or at the command line. The Sybase installer configures the runserver file with *$SYBASE/install* as the location of the error log if you do not choose an alternate location during installation. If you do not specify the location in the runserver file or at the command line, the location of the error log is the directory from which you start Backup Server. Use the backupserver -V option (bcksvr -V on Windows NT) to limit the messages printed to the error log. See the sections describing Backup Server in the *Utility Guide*.

Backup Server error messages are in this form:

```
MMM DD YYY: Backup Server:N.N.N.N: Message Text
```

Backup Server message numbers consist of four integers separated by periods, in the form N.N.N.N. Messages in the form N.N.N are sent by Open Server.

The four components of a Backup Server error message are *major.minor.severity.state*:

- The *major* component generally indicates the functional area of the Backup Server code where the error occurred:

    - 1 – system errors.

    - 2 – Open Server event errors.

    - 3 – Backup Server remote procedure call errors.

    - 4 – I/O service layer errors.

    - 5 – network data transfer errors.

    - 6 – volume-handling errors.

    - 7 – option-parsing errors.

    Major error categories 1– 6 may result from Backup Server internal errors or a variety of system problems. Major errors in category 7 are almost always due to problems in the options you specified in your dump or load command.

- *minor* numbers are assigned in order within a major category.

- *severity* is:

    - 1 – informational, no user action necessary.

- 2, 3 – an unexpected condition, possibly fatal to the session, has occurred. The error may have occurred with usage, environment, or internal logic, or any combination of these factors.

- 4 – an unexpected condition, fatal to the execution of the Backup Server, has occurred. The Backup Server must exit immediately.

- *state* codes have a one-to-one mapping to instances of the error report within the code. If you need to contact Technical Support about Backup Server errors, the state code helps determine the exact cause of the error.

Killing processes

A process is a unit of execution carried out by Adaptive Server. Each process is assigned a unique process identification number when it starts. This number is called a spid. These numbers are stored, along with other information about each process, in master..sysprocesses. Processes running in a parallel-processes environment create child processes, each of which has its own spids. Several processes create and assign spids: starting Adaptive Server, login tasks, checkpoints, the housekeeper tasks, and so on. You can see most of the information by running sp_who.

Running sp_who on a single-engine server shows the sp_who process running and all other processes that are "runnable" or in one of the sleep states. In multi-engine servers, there can be a process running for each engine.

The kill command gets rid of an ongoing process. The most frequent reason for killing a process is that it interferes with other users, and the person responsible for running it is not available. The process may hold locks that block access to database objects, or there may be many sleeping processes occupying the available user connections. A system administrator can kill most running or "runnable" processes, including those that are waiting for:

- An alarm, such as a waitfor command

- Network sends or receives

- A lock

- Synchronization messages from another process in a family

Adaptive Server allows you to kill processes only if it can cleanly roll back any uncompleted transactions and release all system resources that are used by the process. For processes that are part of a family, killing any of the child processes also kills all other processes in the family. However, it is easiest to kill the parent process. For a family of processes, the kill command is detected more quickly if the status of the child processes is `sync sleep`.

Table 12-2 shows the status values that sp_who reports and when the kill command takes effect.

Table 12-2: Status values reported by sp_who

Status	Indicates	Effects of kill command
`recv sleep`	Waiting on a network read.	Immediate.
`send sleep`	Waiting on a network send.	Immediate.
`alarm sleep`	Waiting on an alarm such as: `waitfor delay "10:00"`	Immediate.
`lock sleep`	Waiting on a lock acquisition.	Immediate.
`sync sleep`	Waiting on a synchronization message from another process in the family.	Immediate. Other processes in the family must also be brought to state in which they can be killed.
`sleeping`	Waiting on a disk I/O, or some other resource. Probably indicates a process that is running, but doing extensive disk I/O	Killed when it "wakes up," usually immediate; a few sleeping processes do not wake up and require a server restart to clear.
`runnable`	In the queue of runnable processes.	Immediate.
`running`	Actively running on one of the server engines.	Immediate.
`infected`	Server has detected serious error condition; extremely rare.	kill command not recommended. Server restart probably required to clear process.
`background`	A process, such as a threshold procedure, run by Adaptive Server rather than by a user process.	Immediate; use kill with extreme care. Recommend a careful check of sysprocesses before killing a background process.
`log suspend`	Processes suspended by reaching the last-chance threshold on the log.	Immediate.

Only system administrators can issue the kill command; permission to use it cannot be transferred.

The syntax is:

    kill *spid*

You can kill only one process at a time, but you can perform a series of kill commands in a batch. For example:

    1> kill 7

```
                          2> kill 8
                          3> kill 9
                          4> go
```

A kill command is irreversible and cannot be included in a user-defined transaction. spid must be a numeric constant; you cannot use a variable. Here is some sample output from sp_who:

```
sp_who
fid  spid  status       loginame  origname  hostname         blk_spid  dbname
        tempdbname    cmd                  block_xloid   threadpool
---  ----  ----------   --------  --------  ---------------  --------  ------
        ----------    ----------------  -----------   ------------------
0      1   recv sleep    howard    howard             svr30eng      0  master
        tempdb        AWAITING COMMAND     0        syb_default_pool
0      2   sleeping       NULL      NULL                           0  master
        tempdb        NETWORK HANDLER      0        syb_default_pool
0      3   sleeping       NULL      NULL                           0  master
        tempdb        DEADLOCK TUNE        0        syb_default_pool
0      4   sleeping       NULL      NULL                           0  master
        tempdb        MIRROR HANDLER       0        syb_default_pool
0      5   sleeping       NULL      NULL                           0  master
        tempdb        CHECKPOINT SLEEP     0        syb_default_pool
0      6   sleeping       NULL      NULL                           0  master
        tempdb        HOUSEKEEPER          0        syb_default_pool
0      7   recv sleep     bill      bill          bigblue         0  master
        tempdb        AWAITING COMMAND     0        syb_default_pool
0      8   recv sleep    wilbur    wilbur           hazel         0  master
        tempdb        AWAITING COMMAND     0        syb_default_pool
0      9   recv sleep     joan      joan        luv2work         0  master
        tempdb        AWAITING COMMAND     0        syb_default_pool
0     10   running        foote     foote      svr47hum         0  master
        tempdb        SELECT               0        syb_default_master
(10 rows affected, return status = 0)
```

In the example above, processes 2–6 cannot be killed: they are system processes. The login name NULL and the lack of a host name identify processes them as system processes. NETWORK HANDLER, MIRROR HANDLER, HOUSEKEEPER, and CHECKPOINT SLEEP (or, rarely, CHECKPOINT) always appear in sp_who output. AUDIT PROCESS appears if auditing is available.

Processes 1, 8, 9, and 10 can be killed, since they have the status values "recv sleep," "send sleep," "alarm sleep," and "lock sleep."

In sp_who output, you cannot tell whether a is "recv sleep" belongs to a user who is using Adaptive Server and may be pausing to examine the results of a command, or whether a user has restarted a PC or other terminal, and left a stranded process. Query the sysprocesses table to learn more about questionable processes. For example, this query shows the host process ID and client software used by process 8:

```
select hostprocess, program_name
    from sysprocesses
where spid = 8

hostprocess program_name
----------- ----------------
3993        isql
```

This query, plus the information about the user and host from the sp_who results, provides additional information for tracking down the process from the operating system level.

Using kill with statusonly

The kill ...statusonly command reports on the progress of a server process ID (spid) in rollback status. It does not terminate the spid. The statusonly report displays the percent of rollback completed and the estimated length of time in seconds before the rollback completes. To track the progress of a rollback, you must run kill...with statusonly multiple times:

kill *spid* with statusonly

Where *spid* is the number of the process you are terminating.

For example, the following reports on the process of the rollback of spid number 13:

```
kill 13 with statusonly
spid: 13 Transaction rollback in progress. Estimated rollback completion: 17%
Estimated time left: 13 seconds
```

If the rollback of the spid has completed when you issue kill...statusonly or if Adaptive Server cannot roll back the specified spid, kill...statusonly returns the following message:

```
Status report cannot be obtained. KILL spid:nn is not
in progress.
```

Using *sp_lock* to examine blocking processes

In addition to sp_who, sp_lock can help identify processes that are blocking other processes. If the blk_spid column in the sp_who report indicates that another process has been blocked while waiting to acquire locks, sp_lock can display information about the blocking process. For example, process 10 in the sp_who output above is blocked by process 7. To see information about process 7, execute:

```
sp_lock 7
```

For more information about locking in Adaptive Server, see the *Performance and Tuning Series: Locking and Concurrency Control*.

Housekeeper functionality

The housekeeper task provides important functionalities:

- The housekeeper consists of three tasks: housekeeper wash, housekeeper garbage collection, and housekeeper chores. sp_who displays all three tasks in its output as HK WASH, HK GC, and HK CHORES:

```
sp_who
fid    spid    status        loginame      origname      hostname
blk_spid  dbname  tempdbname  cmd          block_xloid  threadpool
----  -----  ---------   ----------    ---------   ----------
--------  -------  ---------  -------------  ---------  -------------
   0     8    sleeping         NULL         NULL      luv2work
     0   master    tempdb      HK WASH         0  syb_default_pool
   0     9    sleeping         NULL         NULL         NULL
     0   master    tempdb       HK GC          0  syb_default_pool
   0    10    sleeping         NULL         NULL         NULL
     0   master    tempdb    HK CHORES         0  syb_default_pool
```

- The general automatic restart of housekeeper-related system tasks: you need not restart the server if these system tasks quit unexpectedly.

- A system administrator can change all housekeeper task priorities.

  sp_showpsexe, as well as sp_who, recognizes all three housekeeper names.

For more information about sp_who and sp_showpsexe, see the *Reference Manual: Procedures*.

Housekeeper wash

Washing buffers is an optional task that, if enabled, runs only during 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.

Housekeeper chores

The housekeeper chores task th runs only at idle times, and does not use a common configuration parameter. It manages miscellaneous chores, such as:

- Flushing table statistics.

- Flushing account statistics.

- Handling timeout of detached transactions. You can turn off this task using the configuration parameter dtm detach timeout period.

- Checking licence usage. You can turn off this task using the configuration parameter license information.

Housekeeper garbage collection

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, which is affected by the fragmentation of allocated table space, 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 the deleted transactions are committed.

  Use the enable housekeeper GC configuration parameter to configure the delete command and the housekeeper garbage collection task for aggressive or lazy garbage collection.

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.

Running at user priority

The housekeeper garbage collection task typically operates at the priority level of an ordinary user, competing for CPU time with ordinary user tasks. This prevents the list of empty pages from growing faster than the housekeeper can delete them. However, if Adaptive Server is configured for threaded mode, use the sp_bindexeclass 'sv' object type to change the server-wide priority setting for the housekeeper wash task. Set the priority to the EC1, EC2, EC3 level or define a new user-created execution class. This example sets the housekeeper wash task to the highest, EC1, priority:

```
sp_bindexeclass 'HK WASH', 'sv', NULL, 'EC1'
```

Use sp_setpsexe to set the priority level for the housekeeper task for the session if Adaptive Server is configured for threaded or process mode.

This example sets the priority level for the housekeeper wash task (with a spid of 8) to HIGH for the current session:

```
sp_setpsexe 8, 'priority', 'HIGH'
```

See the *Reference Manual: Procedures*.

Configuring enable housekeeper GC

To configure Adaptive Server for garbage collection task, use:

```
sp_configure "enable housekeeper GC", value
```

The valid values for the enable housekeeper GC configuration parameter are:

• 0 – disables the housekeeper garbage collection task, but enables lazy garbage collection by the delete command. You must use reorg reclaim_space to deallocate empty pages. This is the cheapest option with the lowest performance impact, but it may cause performance problems if many empty pages accumulate. Sybase recommends that you do not use this value.

- 1 – enables lazy garbage collection, by both the housekeeper garbage collection task and the delete command. This is the default value. If more empty pages accumulate than your application allows, consider options 4 or 5. You can use the optdiag utility to obtain statistics of empty pages.

- 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. This option is the most effective, but the delete command is the most expensive. This option is ideal if the deletes on your dataonly locked tables are in a batch.

- 5 – enables aggressive garbage collection for the housekeeper, and lazy garbage collection by delete. This option is less expensive for deletes than option 4. This option is suitable when deletes are caused by concurrent transactions.

For example, enter:

```
sp_configure "enable housekeeper GC", 4
```

Using the reorg command

Garbage collection is most effective when you set enable housekeeper GC to 4 or 5. Sybase recommends that you set the parameter value to 5. However, if performance considerations prevent setting this parameter to 4 or 5, and you have an accumulation of empty pages, run reorg on the affected tables. You can obtain statistics on empty pages through the optdiag utility.

When the server is shut down or crashes, requests to deallocate pages that the housekeeper garbage collection task has not yet serviced are lost. These pages, empty but not deallocated by the housekeeper garbage collection task, remain allocated until you remove them by running reorg.

See Chapter 9, Using the reorg Command," in the *System Administration Guide: Volume 2*.

Shutting down servers

A system administrator can shut down Adaptive Server or Backup Server using:

shutdown [*backup_server_name*] [with {wait|nowait}]

The default for the shutdown command is with wait. That is, shutdown and shutdown with wait do exactly the same thing.

Shutting down Adaptive Server

If you do not provide a server name, shutdown shuts down the Adaptive Server you are using. When you issue a shutdown command, Adaptive Server:

1   Disables logins, except for system administrators

2   Performs a checkpoint in each database, flushing pages that have changed from memory to disk

3   Waits for currently executing SQL statements or procedures to finish

In this way, shutdown minimizes the amount of work that automatic recovery must do when you restart Adaptive Server.

The with nowait option shuts down Adaptive Server immediately. User processes are aborted, and recovery may take longer after a shutdown with nowait. You can help minimize recovery time by issuing a checkpoint command before you issue a shutdown with nowait command.

Shutting down a Backup Server

To shut down a Backup Server, include the Backup Server name:

```
shutdown SYB_BACKUP
```

The default is with wait, so any dumps or loads in progress complete before the Backup Server process halts. After you issue a shutdown command, no new dump or load sessions can be started on the Backup Server.

To see the names of the Backup Servers that are accessible from your Adaptive Server, execute sp_helpserver. Use the value in the name column in the shutdown command. You can shut down a Backup Server only if it is:

•   Listed in sysservers on your Adaptive Server, and

•   Listed in your local *interfaces* file.

Use sp_addserver to add a Backup Server to sysservers.

Checking for active dumps and loads

To see the activity on your Backup Server before executing a shutdown command, run sp_who on the Backup Server:

```
SYB_BACKUP...sp_who
spid    status     loginame   hostname     blk    cmd
-----   --------   --------   ----------   ---    --------------
    1   sleeping       NULL         NULL     0     CONNECT HANDLER
    2   sleeping       NULL         NULL     0     DEFERRED HANDLER
    3   runnable       NULL         NULL     0     SCHEDULER
    4   runnable       NULL         NULL     0     SITE HANDLER
    5   running          sa   heliotrope     0     NULL
```

Using *nowait* on a Backup Server

The shutdown *backup_server* with nowait command shuts down the Backup Server, regardless of current activity. Use it only in severe circumstances. It can leave your dumps or loads in incomplete or inconsistent states.

If you use shutdown with nowait during a log or database dump, check for the message indicating that the dump completed. If you did not receive this message, or if you are not sure whether the dump completed, your next dump should be a dump database, not a transaction dump. This guarantees that you are not relying on possibly inconsistent dumps.

If you use shutdown with nowait during a load of any kind, and you did not receive the message indicating that the load completed, you may not be able to issue further load transaction commands on the database. Run a full database consistency check (dbcc) on the database before you use it. You may have to reissue the full set of load commands, starting with load database.

Learning about known problems

The release bulletin is a valuable resource for learning about known problems or incompatibilities with Adaptive Server and Backup Server. Reading the release bulletin in advance can save you the time and guesswork of troubleshooting known problems.

Adaptive Server Enterprise

Index

Symbols

% (percent sign)
    error message placeholder   384
?? (question marks)
    for suspect characters   377
" " (quotation marks)
    enclosing parameter values   10
@@*client_csexpansion* global variable   366

Numerics

7-bit ASCII character data, character set conversion for
    371

A

abstract plan cache configuration parameter   81
abstract plan dump configuration parameter   81
abstract plan load configuration parameter   81
abstract plan replace configuration parameter   82
adding
    database devices   219, 308–314
    date strings   363
    months of the year   363
    remote logins   300–302
    remote servers   294–299
    users to a database   219
additional network memory configuration parameter
    82
address, server   14
aggressive garbage collection   399
    priority level   400
aggressive housekeeper   399
aliases
    server   296
allocation
    pages   308

    units   308
    units *See also* size; space allocation
allow backward scans configuration parameter   85
allow nested triggers configuration parameter   85
allow procedure grouping configuration parameter
    86
allow remote access configuration parameter   86
allow resource limits configuration parameter   87
allow sendmsg configuration parameter   87
allow sql server async i/o configuration parameter   87
allow updates configuration parameter (now called
    allow updates to system tables)   12
allow updates to system tables configuration parameter
    12, 88
alter database command
    omitting database device and   315, 316
    system tables and   290
alternate languages. *See* languages, alternate
application design   219
Arabic character set support   333
ASCII characters
    character set conversion and   371
asynchronous I/O
    limiting Server requests for   166
asynchronous prefetch
    configuring   146
@@*char_convert* global variable   366
@@*client_csid* global variable   366
@@*client_csname* global variable   366
@@*langid* global variable   368
@@*language* global variable   368
@@*max_connections* global variable   218
@@*maxcharlen* global variable   366
@@*ncharsize* global variable   366
audit queue size configuration parameter   90
audit trail   28
    stacktrace of error messages   385
auditing
    queue, size of   90
    sybsecurity database   28

E

                                        Adaptive Server Enterprise

Adaptive Server Enterprise