

## CHAPTER 4

# Behavior Changes

About this Chapter      This chapter describes behavior changes since the previous release of the software.

*🔗* Behavior changes and new features overlap. Some behavior changes are described in the chapter "New Features in Adaptive Server Anywhere 6.0" on page 21.

### Contents

<b>Topic</b>	<b>Page</b>
Platform and protocol support	52
Database server changes	53
Administration utility changes	54
Transact-SQL outer join changes	56
Other changes	57

## Platform and protocol support

As new operating systems and network protocols are introduced, and as older ones become obsolete, the range of platforms supported by Adaptive Server Anywhere changes.

- ◆ **Operating systems** There are several changes to the list of supported operating systems.
  - ◆ Several UNIX operating systems are supported for the first time.
  - ◆ DOS and Macintosh operating systems are no longer supported for client or server.
  - ◆ No network database server is supplied for Windows 3.x (no network server). The personal database server now requires the Win 32s subsystem, version 1.30.

Not all operating systems may be made available at the same time.

- ◆ **Network protocols** NetBIOS datagrams (NetDG) is no longer supported as a communications protocol. NetBIOS sessions are still fully supported
- ◆ **Programming interfaces** The HLI and DDE programming interfaces are not supported in Version 6. If you have existing HLI or DDE applications that you wish to run against a Version 6 database engine, you can do so using the Version 5 HLI and DDE interfaces in conjunction with the Embedded SQL interface compatibility library included in Version 6.

*ℳ* For information about the Embedded SQL interface compatibility library, see "Using the compatibility library" on page 76.

JDBC is provided as an additional programming interface.

- ◆ **TCP/IP change** UDP is a transport layer protocol that sits on top of IP. Adaptive Server Anywhere uses UDP on top of IP to do initial server name resolution and TCP for connection and communication after that, while SA used UDP for everything.

## Database server changes

This section describes changes in the behavior of the database server not described elsewhere.

- ◆ **Name changes** In the documentation, the standalone engine is now referred to as the **personal database server**. The network server continues to be known as the **network database server**.
- ◆ **Server interface** The network server interface is now removed to a separate utility named *dbconsol*. The network server itself now displays the same window as the personal server.
- ◆ **Minimized server icon** When the database server for Windows 95 and Windows NT is minimized, it now appears as an icon in the system tray, rather than on the task bar.

The system tray is at the other end of the task bar from the Start button.

- ◆ **Database starting permission** When a database server is started, you can use the `-gd` command-line option to control who can start databases. In previous releases, the default setting has been that all users can start databases. Starting with Version 6.0, the default setting for the network database server is that only users with DBA authority can start a database. For the personal server, the default is still to allow all users.
- ◆ **Cursors closed on ROLLBACK TO SAVEPOINT** In previous versions of the software, cursors were not closed on a ROLLBACK TO SAVEPOINT. Now, cursors opened after the savepoint are closed on a rollback to that savepoint.
- ◆ **Minimum packet size** The minimum packet size for client/server communications has changed from 200 bytes to 280. This setting is controlled by the `-p` database server command-line switch and by the **CommBufferSize** connection parameter

## Administration utility changes

In addition to the support for SQL statements that carry out database administration tasks (see "Database file manipulation statements" on page 43), there are several changes to the behavior of the administration utilities.

The new behavior is documented in "Database Administration Utilities" on page 63 of the book *Adaptive Server Anywhere Reference Manual*. This section describes behavior changes from previous versions of the software.

- ◆ **ISQL name change** The ISQL utility file name has been renamed to *dbisql.exe*. This removes the name conflict with the Adaptive Server Enterprise *isql* utility. Throughout the manual, ISQL is now referred to by its full name of Interactive SQL to avoid confusion with the Adaptive Server Enterprise utility.
- ◆ **DBWATCH replaced** The *dbwatch* server monitoring facility has been replaced by a Console utility named *dbconsol*. The Console utility can monitor the personal server as well as the network server.
- ◆ **dbinfo options** The *dbinfo* utility uses a connection string to locate the database, rather than a database file name.
- ◆ **dbo user in utilities** The *-g* command-line switch for the *dbinit*, *dbupgrad*, *dbunload*, and *dbxtract* utilities has been dropped. This switch was used to specify a different user name for the Adaptive Server Enterprise database owner user ID **dbo**.
- ◆ **Interactive SQL DBTOOL statement** The Interactive SQL DBTOOL statement has been dropped. This statement was provided to allow database administration tasks to be executed from Interactive SQL.

Sybase Central now provides the database administration interface for interactive use, and the addition of SQL statements for administration purposes, such as CREATE DATABASE, allows scripting from other applications.

If you have existing Interactive SQL scripts that you wish to run, you can do so by running your scripts using your Version 5 ISQL utility. Alternatively, you can edit your scripts to use the new statements.

☞ For information on using Version 5 ISQL with Version 6 software, see "Using the version 5 utilities with Adaptive Server Anywhere 6.0" on page 69.

- ◆ **Interactive SQL database tools window** The Interactive SQL database tools window has been dropped. This window was provided to allow database administration tasks to be executed from Interactive SQL.

Sybase Central now provides the database administration interface for interactive use, and the addition of SQL statements for administration purposes, such as CREATE DATABASE, allows scripting from other applications.

- ◆ **dbunload changes** To supply a list of tables to unload for the dbunload utility, you now use the `-t` command-line switch.
- ◆ **Custom collation files** Previously, the value of `-z` could be a collation name (for an internal collation) or a file name (for a custom collation). Now, only a collation name can be specified. The corresponding collation must be defined in either *collseqs.sql* or *custom.sql* in the *scripts* subdirectory.

The procedure for building a database with a custom collation file is now as follows:

- ◆ Create a custom collation file.
- ◆ Edit the custom collation file.
- ◆ Convert the file to a SQL script using the *dbcollat* command-line utility. This step was not required in previous releases.
- ◆ Add the custom collation to the *custom.sql* script. This step was not required in previous releases.
- ◆ Create the new database using the Initialization utility, specifying the name of the custom collation. In previous releases, you would have specified the collation file name.

☞ For a more detailed description, see "Creating databases with custom collations" on page 309 of the book *Adaptive Server Anywhere User's Guide*.

- ◆ **RTSQL dropped** The runtime SQL processor is no longer provided. You can use Interactive SQL with `-q` to perform any tasks previously performed by RTSQL. From the command line, the syntax is as follows:

```
dbisql -q isql-command
```

## Transact-SQL outer join changes

Transact-SQL outer joins use join operators (`*=` and `=*`) in the WHERE clause, while SQL/92 outer joins use an ON keyword in the FROM clause to express the join.

In Version 6 there are some changes to the behavior of Transact-SQL outer joins: some joins allowed in previous versions of the software now give syntax errors.

Transact-SQL outer joins were implemented for compatibility with Sybase SQL Server (now Adaptive Server Enterprise). The changes made reflect changes made in recent versions of Adaptive Server Enterprise. These changes give errors for outer joins that have potentially ambiguous syntax. Increasing the precision required in queries improves the reliability of query results.

The changes are as follows:

- ◆ You cannot mix SQL/92 syntax and Transact-SQL outer join syntax in a single query. This applies to views used by a query also: if a view is defined using one dialect for an outer join, the same dialect must be used for any outer-join queries on that view.
- ◆ A table cannot participate in both a Transact-SQL outer join clause and a regular join clause. For example, the following WHERE clause is not allowed:

```
WHERE R.x *= S.x
AND S.y = T.y
```

When you cannot rewrite your query to avoid using a table in both an outer join and a regular join clause, you must divide your statement into two separate queries.

- ◆ You cannot use a subquery that contains the null-supplying table of an outer join. For example, the following WHERE clause is not allowed:

```
WHERE R.x *= S.y
AND EXISTS ( SELECT *
             FROM T
             WHERE T.x = S.x )
```

## Other changes

This section gathers together other behavior changes, which do not belong in previous sections.

- ◆ **File name changes** Many files have changed names, including executable files.
- ◆ **% comments** %-style comments are now a deprecated feature.

The new **Percent\_as\_comment** option controls the meaning of %. The default setting is ON (that is, % is treated as a comment indicator) for compatibility.

Procedures, triggers and views created with %-style comments are converted to double-dash comments when stored in the catalog.

**Existing procedures must be recreated before changing option**

Any existing procedures containing %-style comments must be recreated before changing the option setting; otherwise, the procedures will fail to load. An unload/reload of your database accomplishes this.

- ◆ **Ignored options** Some database options duplicated the functionality of command-line switches. These options are now ignored, and only the command-line switches are acted on. The options include:
  - ◆ THREAD\_COUNT
  - ◆ THREAD\_STACK
- ◆ **Default NEAREST\_CENTURY option** The default setting for the "NEAREST\_CENTURY option" on page 163 of the book *Adaptive Server Anywhere Reference Manual* has been changed from 0 to 50. This change does not affect existing databases: it takes effect only on databases created with Version 6.
- ◆ **LOAD TABLE and UNLOAD TABLE** The FORMAT clause of these statements now takes a keyword (ASCII) as opposed to a string. The string 'ascii' is accepted for compatibility.
- ◆ **Agent connection parameter** The value of **engine** for this connection parameter is now ignored.

- ◆ **sqlpp -l option** SQLPP -l userid, password allowed you to provide a userid and password to be used for checking permissions on static embedded SQL statements. This feature has been removed in 6.0. Static SQL statements will use the userid and password provided on the CONNECT.
- ◆ **date format changes** The DATE\_FORMAT option takes a string argument built up from a set of predefined characters (MM for months, and so on). The f argument providing French month names is now deprecated. The month and day names are now provided according to the language version of the software.
- ◆ **Temporary table changes** Global temp tables cannot be altered or dropped unless all users except the one making the change that have referenced the temporary table have disconnected.