72540-01-1102-01

Release Bulletin
Sybase® SQL Server™ Release 11.0.2
for Digital OpenVMS Alpha

The following table lists section titles and page numbers for this *Release Bulletin*:

1. Product Summary

Enclosed is Sybase SQL Server release 11.0.2 and Language Modules, which are compatible with the following platform and operating system configurations:

- Digital Alpha OpenVMS version 6.2

The Language Modules enable French, German, and Spanish language support for SQL Server and Backup Server.

1.1. Network Protocols Supported

This release supports the following network protocols:

1. DECnet
2. TCP
   - UCX version 3.1
   - TGV’s Multinet tcp
   - Wollongong’s Pathway TCP

➤ ***Note***

Wollongong TCP users are limited to port numbers between 5000 and 65535. Port numbers between 1024 and 4999 are reserved by Wollongong for network connections that do not specify an explicit port number. See “Improper Release of ASTs” on page 22 for information about a known problem with Wollongong’s Pathway TCP.

1.1.1. Configuration Changes for UCX Protocol

If you want to use UCX, you must load the INET device emulator provided with the UCX kit before you start SQL Server. SQL Server does not work if the INET driver is not loaded. Use the SYS$MANAGER:UCX$LOAD_INETDRIVER.COM command file to load the INET emulator.

You may need to adjust the BYTLM of clients as well.

For information about the size of BYTLM process quotas, see Chapter 8 or the *Installation and Configuration Guide.*

2. Special Installation Instructions

For installation instructions, please see the *Installation and Configuration Guide for Digital OpenVMS Alpha*. The following sections include special installation instructions.

2.1. OpenVMS Supports TCP NODELAY

VMS now supports TCP NODELAY. Use the **tcp no delay** option with **sp_configure** or edit your server's configuration file to add this option.

You must reboot SQL Server for this option to take effect.

2.2. OpenVMS Supports Host Names for TCP

OpenVMS now supports host names for TCP / IP. IP addresses are no longer required in the interfaces file.

2.3. DECnet Objects

You can now use object numbers for DECnet entries in the interfaces file for Backup Server.

2.4. Installing SQL Server and OmniCONNECT

If you are installing both SQL Server and OmniCONNECT™, make sure that you install OmniSQL Server™ in a separate release directory. If OmniCONNECT is installed in the same release directory as SQL Server, it will overwrite important files, including **buildmaster** and localization files.

2.5. SQL Server Installation Directory (*SYBASE_SYSTEM:[SYBASE}*)

You can install SQL Server release 11.0.2 and compatible Language Modules in the same directory.

Do not install any release 11.0.2 software in the same directory as earlier Sybase releases. If you have earlier releases of Sybase products installed on your system, create a new installation directory for release 11.0.2.

If Sybase SQL Toolset™ is installed in the same directory as SQL Server and Open Client™, you must reinstall SQL Toolset in a separate directory prior installing or upgrading to release 11.0.x.

Open Client release 10.x overwrites existing library files that APT Workbench™ mixed-mode applications require for relinking. By reinstalling SQL Toolset in a separate directory, you preserve those library files necessary for a successful link of your application.

See the *Release Bulletin* for the latest release of SQL Toolset for more information.

2.6. CD-ROM Installations

All installations of Sybase software from CD-ROM media must be local. The **VMSINSTAL** command that you enter for CD-ROM installations does not allow remote installations. If you attempt a remote installation from CD-ROM, you receive an error stating that the image file you have specified does not exist.

2.7. Configuring a New Backup Server

When you use **sybinit** to configure a new 11.0.2 Backup Server, **sybinit** may be unable to connect with the Backup Server to verify that it is running. If you receive connectivity errors during configuration of a new Backup Server, follow these steps to verify that the Backup Server is running:

1. Use **isql** to log into your new 11.0.2 SQL Server.
2. Enter the following command:

   ```
   SYB_BACKUP...sp_who
   ```

   If Backup Server is running, a list of processes is returned.

3. Special Upgrade Instructions

Chapter 5 of the *Installation and Configuration Guide* provides instructions for upgrading your current SQL Server to release 11.0.2.

3.1. Upgrading Sequence

Before you upgrade to SQL Server release 11.0.2, first upgrade other Sybase products you have running.

3.2. Default Stack Size

The following is the stack size configuration for SQL Server release 11.0.2 on Digital OpenVMS Alpha:

Default stack size = 65,536
Minimum stack size = 34,816

During the upgrade, the stack size is automatically changed to the default value if:

- The current value is less than the default value. This means you do not need to increase your SQL Server stack size before beginning the upgrade to SQL Server release 11.0.2.
- The current value is set to 0 (used to indicate the default value in System 10™).

3.3. Default Total Memory Size

The default (and minimum) “total memory” size is 7500 2K pages (15MB). This value is adjusted, if needed, during upgrade.

3.4. Upgrades from Pre-10.0 SQL Server

You can upgrade to SQL Server release 11.0.2 only if you currently use SQL Server release 4.9.2 or later. If you use a version prior to release 4.9.2, upgrade to release 4.9.2 before performing an upgrade to 11.0.2. See Chapter 5 of the *Installation and Configuration Guide* for instructions.

4. Product Compatibilities

4.1. Compatibility with Sybase Products

SQL Server 11.0.2 has been extensively tested against other Sybase products. The following Sybase products have been verified to work with SQL Server 11.0.2.

Table 1: Verified compatibilities

Product	SQL Server 11.0.2
DB-Library™ 4.2.5	•
DB-Library 4.6.2	•
DB-Library 10.0.2	•
DB-Library 10.0.3	•
Client-Library™ 10.0.2	•
Client-Library 10.0.3	•

Table 1: Verified compatibilities (continued)

Product	SQL Server 11.0.2
Open Server™ 10.0.2	•
Open Server 10.0.3	•
Embedded SQL™/C	•
Embedded SQL/Cobol	•
Replication Server® 11.0	•
Replication Server 10.0.3	•
Replication Server 10.1	•
SQL Server Manager 11.0	•
SQL Server Monitor 11.0	•

(1) Please consult the Replication Server Release Bulletin for your platform and Replication Server release level.

The following sections discuss in more detail compatibility between SQL Server and other products.

4.2. Compatibility with Backup Server

SQL Server release 11.0.2 is not compatible with Backup Server release 10.0. If you attempt to use SQL Server release 11.0.2 with Backup Server release 10.0, an error message like the following results:

```
Open Server Session Fatal Error: 16227.15.0: Unknown token TDS
stream received by spid 11
```

Dumps generated by Backup Servers release 10.0 through release 11.0 are compatible with Backup Server 11.0.2.

See Chapter 6 of the *SQL Server Installation and Configuration Guide* for instructions on installing Backup Server.

4.2.1. Heterogeneous Backups

Backup Server supports remote dumps in a heterogeneous (cross-platform) environment. You can execute a dump from a local SQL Server or Backup Server to a tape drive or other dump or load device through a remote Backup Server on a different platform. You must load from a device on the same platform as the device where the dump was directed, and you must load to the same platform from which you dumped.

For example, you could dump to an HP tape drive from SQL Server on a Sun platform, but that dump would have to be read back from an HP tape; it could not be read from a Sun tape drive.

You may encounter an error when you execute a multi-striped dump to remote heterogeneous platforms if one of the remote stripes is on OpenVMS Alpha or OpenVMS VAX and the dump is executed from a different platform. In this situation, the dump succeeds and an error occurs during the load.

See the *System Administration Guide* for detailed information on planning and executing backups.

4.3. Compatibility with SQL Server Monitor

SQL Server release 11.0.2 is not compatible with previous releases of SQL Server Monitor. SQL Server Monitor 11.0.1 is compatible with both SQL Server release 11.0 and SQL Server release 11.0.2.

SQL Server Monitor Server, one of the components of the SQL Server Monitor product, is tightly integrated with the SQL Servers it monitors. For this reason, the Monitor Server executable has a different version for each of the SQL Server releases (10.0.x, and 11.0 or later) that can be monitored by SQL Server Monitor 11.0.1. The following table describes which versions of the Monitor Server executable are compatible with different releases of SQL Server:

Table 2: Compatibility between Monitor Server and SQL Server

For This Release of SQL Server	Use This Release of Monitor Server
10.0.x	10.3
11.0 or later	11.0

If you attempt to use SQL Server release 11.0.2 with a SQL Server Monitor release that is earlier than 11.0, one of two problems can occur: you will receive error messages similar to the following in the *monserver.log* file:

```
2.03/OPT/Fri Jan 28 09:56:01 PST 1995
January 19 10:55:53 1995: msgid:20112 -- kbattach: attached to
wrong Kernel region
January 19 10:55:53 1995: Unable to attach with shared memory.
January 19 10:55:53 1995: Fatal error -1 detected
```

or the SQL Server Monitor will boot successfully, but it will produce erroneous data.

See your SQL Server Monitor documentation for information on installing SQL Server Monitor.

4.4. Compatibility with Replication Server

SQL Server release 11.0.2 is compatible with Replication Server releases 10.0.1 and 11.0.1

4.5. Compatibility with Sybase Open Client/Server Products

The following sections describe compatibility between SQL Server release 11.0 and the Open Client / Server products Open Client / C, Open Server, and Embedded SQL.

4.5.1. General Compatibility with Open Client/Server Products

SQL Server release 11.0.2 for your platform works with the Open Client / Server products shown in Table 3:

Table 3: Interoperability with Open Client/Server products

Platform/Operating System	DB-Library		Client-Library		Open Server	
	10.0.2	10.0.3	10.0.2	10.0.3	10.0.2	10.0.3

Open Client / Server applications that use SQL Server release 11.0.2 must be built with 10.0.3 header files and linked with 10.0.3 libraries.

Open Client / Server applications built with 10.0 precompilers are only compatible with SQL Server release 10.0.

➤ ***Note***

After you have upgraded to SQL Server release 11.0.2, you must use release 10.0.2 precompilers to rebuild applications you used with SQL Server release 10.0. You do not need to rebuild applications you used with SQL Server release 10.0.1 or 10.0.2.

4.5.2. Compatibility with Open Client

SQL Server release 11.0.2 is compatible with Open Client releases 4.x and 10.0.x, although it is not compatible with Open Client release 10.0. Some SQL Server release 10.0.1 and 10.0.2 features are not available to some earlier Open Client releases.

4.5.3. Compatibility with Open Server

SQL Server release 11.0.2 is compatible with Open Server releases 10.0.1 and 10.0.2, but not with Open Server release 10.0.

4.5.4. Compatibility with Embedded SQL

Use Embedded SQL release 10.0.1 or 10.0.2 for applications that need to access SQL Server release 11.0.2. Embedded SQL release 10.0 generates applications that are incompatible with SQL Server release 11.0.2.

4.6. Compatibility with SQL Toolset

Some areas of incompatibility exist between SQL Toolset releases 4.x and 5.x and SQL Server release 11.0.2.

See the *Release Bulletin* for the latest release of SQL Toolset for more information.

4.6.1. Cursors

SQL Toolset does not currently support cursors.

4.6.2. Datatypes

The following issues relate to SQL Toolset’s support of SQL Server datatypes:

- SQL Toolset does not support the *numeric*, *real*, *double precision*, and *decimal* datatypes. Conversion of the values for these datatypes may result in loss of precision.
- You may encounter problems with APT Workbench and Data Workbench® when you attempt to insert a numeric value into a field whose datatype is *numeric*.

- Several problems arise when you use the **Copy Table** functionality of Data Workbench with tables that have *numeric* columns.
- Use of the **Copy Out** functionality of Data Workbench with lower boundary input using *real* datatypes produces incorrect output.

4.6.3. Other SQL Toolset Incompatibilities

The following issues relate to table and column constraints:

- APT-Build™ cannot be used on a table with an IDENTITY column.
- If you leave all fields in a table blank when using the **Data Entry** functionality of Data Workbench, the insert fails.
- APT Workbench cannot compile rules from SQL Server with the **DB Objects Compile All Rules** option if your application has columns that contain a check constraint.
- If you attempt to use the **Copy In** or **Copy Out** options of **Copy Table**, and the host datatype is changed to *float* for the IDENTITY column, an access violation occurs.

4.7. Interoperability with Microsoft Office Products

The following Microsoft Office products have been tested for interoperability with SQL Server release 11.0:

- MS Access 2.0
- MS Visual Basic 3.0
- MS Excel 5.0
- MS Word 6.0

These Microsoft clients require one of the following drivers to communicate with SQL Server 11.0:

- Intersolv ODBC 2.00.0000 for Windows 3.1
- Intersolv ODBC 2.00.0001 for Windows NT

◆ ***WARNING!***

You may experience problems if you attempt to use Intersolv ODBC 2.0 in programming mode. These known problems do not occur in interactive mode.

◆ ***WARNING!***

Microsoft Word 6.0 was not tested with the Intersolv ODBC driver for Windows NT.

4.7.1. Microsoft Access 2.0 Basic and *create table* command

Microsoft Access 2.0 Basic causes a GPF (general protection failure) when trying to execute **create table** statements. This behavior is documented in Bug #62963.

4.7.2. Microsoft Access 2.0 and Exported Tables

Microsoft Access 2.0 attaches tables to a specified database. When the table is exported back to the database, the characteristics of the database table (such as indexes, primary or secondary keys, columns that are not null, and so on) are lost. This behavior is documented in Bug # 63190.

4.8. Advertised Server Name (ASN) Not Supported

ASN is not supported on SQL Server release 11.0.2 running on UNIX platforms.

4.9. SPX/IPX Not Supported from VMS Client Machines

VMS clients (for example, DB-Library, Client-Library, and **isql**) do not understand an SPX/IPX entry in the interfaces file. Sybase does not support the SPX/IPX network protocol from VMS client machines.

4.10. Existing Report Workbench Reports

To use pre-10.0 Report Workbench™ reports with release 10.0.2 and later SQL Server, update the system table that stores the report definitions to the current table version before upgrading your SQL Server to release 10.0.2 or later. The system report table has a column named *level*, which is a reserved word for 10.0 SQL Server and must be changed.

To make reports created with releases 4.1 and 5.0 of Report Workbench compatible with release 10.0.2 or later SQL Server, run the **convpre100rw** script. This script is available with release 5.2.2 or 5.3 SQL Toolset. Use the following syntax to run the script:

```
$ isql :== $sybase_system:[sybase.bin]isql
$ isql /user="sa" /password="password" -
      /input=sybase_system:[sybase.scripts]convpre100rw
```

If you have pre-release 4.1 Report Writer reports, you must first run the **convrw** binary to make them 5.0 compatible before running **convpre100rw**. For more information on **convrw**, see the *SQL Toolset Administration Guide Supplement* for your platform.

5. Release Contents

5.1. Changes to Cache System Procedures

The following system procedures have changed:

- **sp_bindcache**
- **sp_cachestrategy**
- **sp_unbindcache**

5.1.1. *sp_bindcache*

Function: binds a database, table, index, text or image object to a data cache.

Syntax:

```
sp_bindcache cachename, dbname[, [owner_name.]tablename
[,indexname | "text only" ]]
```

Parameters:

cachename – is the name of an existing data cache.

dbname – is the name of the database to bind to the cache, or the database containing the table, index, text or image object to be bound to the cache.

owner_name – is the name of the table's owner. If the table is owned by "dbo", the owner name is optional.

tablename – is the name of the table to bind to the cache, or the name of a table whose index, text or image object is to be bound to a cache.

indexname – is the name of an index to bind to a cache.

text only – binds text or image objects to a cache. Enclose this parameter name in quotes. When this parameter is used, you cannot give an index name at the same time.

5.1.2. *sp_cachestrategy*

Function: enables or disables prefetching (large I / O) and MRU cache replacement strategy for a table, index, text or image object.

Syntax:

```
sp_cachestrategy dbname , [owner_name.]tablename [, indexname |
"text only" | "table only" [, { prefetch | mru }, { "on" |
"off"}]]
```

Parameters:

dbname – is the name of the database where the object is stored.

owner_name – is the name of the table's owner. If the table is owned by "dbo", the owner name is optional.

tablename – is the name of the table.

indexname – is the name of the index on the table.

text only – is specified to change the cache strategy for a text or image object. Enclose this parameter in quotes.

table only – is specified to change the cache strategy for a table. Enclose this parameter in quotes.

prefetch | **mru** – specifies which setting to change.

on | **off** – specifies the setting: "on" or "off". Enclose this parameter in quotes.

5.1.3. *sp_unbindcache*

Function: unbinds a database, table, index, text or image object from a data cache.

Syntax:

```
sp_unbindcache dbname [,[owner.]tablename [, indexname | "text
only" ]]
```

Parameters:

dbname – is the name of database to unbind, or the name of the database containing the objects to be unbound.

owner – is the name of the table's owner. If the table is owned by the "dbo", the owner name is optional.

tablename – is the name of the table to unbind from a cache, or the name of a table whose index, text or image object is to be unbound from a cache.

indexname – is the name of an index to unbind from a cache.

text only – unbinds text or image objects from a cache. Enclose this parameter in quotes.

5.2. Automatic Shutdown After *disk refit*

The **disk refit** command automatically shuts down SQL Server after rebuilding the *sysusages* and *sysdatabases* tables. Reboot the server to use the new system tables.

5.3. Maximum Tables in a Constraint

Constraints cannot reference more than 16 tables and 12 worktables (internally generated tables).

5.4. Configuration File

SQL Server release 11.0.x uses the configuration file to store all SQL Server parameters. This file gets stored on the master device. When you install a new SQL Server or upgrade your current SQL Server, your parameters are set to the default configuration for the new release.

You can edit your configuration file manually using **sp_configure** so that the parameter values are more appropriate for your site.

If your master device becomes corrupted during an upgrade from SQL Server release 11.0, you may lose your configuration file. If you lose your configuration file, SQL Server creates a new file with default settings.

See *What's New in Sybase SQL Server Release 11.0* and the *System Administration Guide* for further information on using, modifying, or recreating the configuration file.

5.4.1. *buildmaster /alter* qualifier No Longer Supported

As of SQL Server release 11.0, the **buildmaster /alter** qualifier is no longer supported. Prior to release 11.0, this qualifier was used to reset your SQL Server configuration parameters.

If your SQL Server configuration parameters are so high that SQL Server does not start, you can remedy the problem in one of two ways:

Method One: Changing the Value Field

Manually edit your configuration file to change the "value" field of the offending parameter to "default":

```
user connections=default
```

When you boot SQL Server using this configuration file, the built-in default value for that parameter is used.

You can also manually edit the "value" fields in your configuration file and set them to any other value within the valid range for that parameter.

Method Two: Renaming the Configuration File

The following method causes SQL Server to generate a configuration file with all the parameters set to SQL Server's built-in default values:

1. If you have configuration files in the *SYBASE* directory, either move them to a different directory or rename them to something other than *SERVER_NAME.CFG*.
2. Start SQL Server with the **startserver** command. This boots SQL Server and creates the configuration file *SYBASE:SERVER_NAME.CFG* using SQL Server's built-in default configuration values.

5.5. *dataserver /config* Command Qualifier

To start SQL Server using a configuration file:

1. Edit your *runserver* file to use the **/config** qualifier with the **dataserver** command:

   ```
   dataserver /config= configuration_file_pathname
   ```

   where *configuration_file_pathname* is the full path and name of the configuration file you want to use.

2. Use the **startserver** command to start SQL Server.

5.6. Backup Server

For releases 10.0 and later, all dump and load operations use Backup Server. See Chapter 6 of the *Installation and Configuration Guide* for information on configuring Backup Server.

5.6.1. Heterogeneous Backups

Backup Server supports remote dumps in a heterogeneous (cross-platform) environment. You can execute a dump from a local SQL Server or Backup Server to a tape drive or other dump or load device through a remote Backup Server on a different platform. You must load from a device on the same platform as the device where the dump was directed, and you must load to the same platform from which you dumped.

Remote dumps are supported between the following platforms:

- Digital Alpha OpenVMS
- NCR System 3000
- Digital Alpha OSF
- DEC VAX OpenVMS
- HP9000 HP-UX
- IBM RISC System / 6000 AIX
- SunOS 4.x (BSD)
- SunOS 5.x (SVR4)

You may encounter an error when you execute a multi-striped dump to remote heterogeneous platforms if one of the remote stripes is on Alpha OpenVMS or VAX OpenVMS and the dump is executed from a different platform. In this situation, the dump succeeds and an error occurs during the load.

See the *System Administration Guide* for detailed information on planning and executing backups.

5.6.2. Supported Media for Backups

Sybase supports the following types of media for dumping from SQL Servers:

- 4-mm / SCSI
- 8-mm / SCSI
- Disk

5.6.3. *sp_addumpdevice* Parameters

In releases 10.0.1 and later, the parameters for **sp_addumpdevice** have been changed. To accommodate these parameter changes, you may need to change the dump scripts you used for 10.0 or pre-10.0 SQL Server.

You no longer need to specify a value for the *cntrltype* parameter when you execute **sp_addumpdevice**. Tape devices are assigned a *cntrltype* of 3; disk devices, a *cntrltype* of 2. In addition, the **skip|noskip** option no longer applies to **sp_addumpdevice**.

See Volume 2 of the *SQL Server Reference Manual* for details on the changed **sp_addumpdevice** parameters. See *What's New in SYBASE SQL Server Release 11.0?* for information on changes to dump scripts and on the logic used in tape label checking.

6. Documentation Updates and Clarifications

The following documentation updates and clarifications apply to Sybase SQL Server release 11.0.2 and Language Modules.

6.1. Volume Change Requests

In most circumstances, when dumping to or loading from tape, floppy disk, or other removable media, you must use the OpenVMS **REPLY** command to change media.

On OpenVMS systems, the operating system, not the Backup Server, requests a volume change when it detects the end of a volume or when the specified drive is off-line. You should use the **REPLY** command to respond to any volume change requests from OpenVMS.

Use **sp_volchanged** only to respond to volume change requests from Backup Server. These requests occur only in limited circumstances. You can recognize Backup Server messages because they always begin with "Backup Server *server_name*:", where *server_name* is the name of the Backup Server executing the dump or the load.

See the *System Administration Guide* for detailed information on executing dumps and loads.

6.1.1. *deadlock retries* Configuration Parameter

The description of the SQL Server configuration parameter **deadlock retries** incorrectly states that this parameter pertains to deadlocks on data pages. Instead, it pertains to deadlocks on index pages.

When a task detects that it has become a deadlock victim during a page split, it releases its locks and traverses the index again, expecting that the task with which it was deadlocked will have finished executing by the time it gets to the split point again.

The **deadlock retries** parameter specifies the number of times SQL Server retries the command before it gives up and returns the application to deadlock status.

6.1.2. *remote access* Variable Permission

The **remote access** configuration variable, which is documented to require the user to be System Security Officer (SSO), returns error 567 when the SSO issues **sp_configure "remote access"**. The System Administrator should execute **sp_configure "remote access"** to turn remote access off or on.

6.2. *Performance and Tuning Guide* Updates

6.2.1. Using *dbcc tune* (*cleanup*) for Performance Improvements

SQL Server performs redundant memory clean-up checking as a final integrity check after processing each task. In very high throughput environments, a slight performance improvement may be realized by skipping this clean-up error check. To turn off the error checking, enter the following command in **isql**:

```
1> dbcc tune(cleanup,1)
2> go
```

This frees up any memory that a task might hold. If you turn the error checking off, but get memory errors, you can re-enable the checking as follows:

```
1> dbcc tune(cleanup,0)
2> go
```

Reissue this command each time the server is started.

For more information about **dbcc tune (cleanup)**, refer to the *Performance and Tuning Guide*.

6.3. *SQL Server Reference Manual* Updates

6.3.1. Date Functions

The Date Functions section does not provide a complete list of the date parts accepted as parameters of the **datepart** function. The following date parts are also accepted by this function:

- **calweekofyear**, which can be abbreviated as **cwk**, returns the ordinal position of the week within the year.
- **calyearofweek**, which can be abbreviated as **cyr**, returns the year in which the week begins.
- **caldayofweek**, which can abbreviated as **cdw**, returns the ordinal position of the day within the week.

6.4. *SQL Server Reference Supplement* Updates

6.4.1. Incorrect *status* Bit Information for *sysdevices*

The *status* control bit information for the *sysdevices* table is incorrect. The following table contains corrected and additional information about the *status* control bits.

Table 4: status control bits in the sysdevices table

Decimal	Hex	Status
1	0x01	Default disk
2	0x02	Physical disk
4	0x04	Logical disk (not used)
16	0x10	Dump device
32	0x20	Serial writes
64	0x40	Device mirrored
128	0x80	Reads mirrored
256	0x100	Secondary mirror side only
512	0x200	Mirror enabled
1024	0x400	Master device is mirrored
2048	0x800	Mirror disabled (used internally)

Table 4: status control bits in the sysdevices table (continued)

Decimal	Hex	Status
4096	0x1000	Primary device needs to be unmirrored (used internally)
8192	0x2000	Secondary device needs to be unmirrored (used internally)

6.4.2. Incomplete *status2* Bit Information for *sysdatabases*

The *Reference Supplement* contains incomplete information about bit representations of the *status2* control bit. The table below describes additional bit representations for the *status2* column.

Table 5: status2 control bits in the sysdatabases table

Decimal	Hex	Status
1	0x0001	**abort tran on log full**; settable
2	0x0002	**no free space acctg**; settable
4	0x0004	**auto identity**; settable
8	0x0008	**identity in nonunique index**; settable
16	0x0010	Database is offline
32	0x0020	Database is offline until recovery completes
64	0x0040	Database is being recovered (internal use)
32768	0x8000	Database does not have a dedicated log device

6.4.3. *sensitivity* Column Incorrectly Included in System Table Descriptions

The descriptions of the *sysloginroles*, *sysroles*, *syssrvroles*, and the *sysusermessages* system tables incorrectly refer to the *sensitivity* column. This column is present only in Secure SQL Server™ and should not appear in SQL Server 11.0.2 documentation.

Specific corrections are:

- *sysloginroles* – The clustered index is on *suid* (not on *suid* and *sensitivity*).
- *sysroles* – The unique, clustered index is on *lrid* (not on *lrid* and *sensitivity*).
- *syssrvroles* – The unique, clustered index is on *srid* (not on *srid* and *sensitivity*).
- *sysusermessages* – The *sensitivity* column should be deleted from the table that lists the columns in *sysusermessages*.

6.4.4. *syscurconfigs* System Table

The information in the table shown below supplements the information given for the **comment** and **memory_used** rows of the *syscurconfigs* table in the “System Tables” chapter of the *SQL Server Reference Supplement*.

Table 6: Columns in the syscurconfigs table

Column	Datatype	Description
comment	*varchar(255)*	Amount of memory used by each configuration parameter, represented in a string format. Values marked with a hash mark (#) share memory with other parameters.
memory_used	*int*	Integer value for the amount of memory used by each configuration parameter.

6.5. *What’s New in Sybase SQL Server Release 11.0?* Updates

6.5.1. Transact-SQL Keywords

Chapter 3 of *What’s New in Sybase SQL Server Release 11.0?* contains a list of new Transact-SQL keywords. In addition to **max_rows_per_page** and **syb_terminate**, the list should include **errordata**.

7. Highlighted Known Problems

This section describes known problems that apply to Sybase SQL Server release 11.0.2 and Language Modules.

7.1. Problem Reports

Listings of all System Problem Reports (SPRs) and Closed Problem Reports (CPRs) for release 11.0.2 SQL Server are located in *SYBASE_SYSTEM:[SYBASE.INSTALL.SPR]*.

The problem reports are organized by functional areas of the product. For example, the file *cpr_bus* contains a listing of closed (fixed) problem reports pertaining to the Backup Server and the file *spr_bus* contains a listing of currently open problem reports for the Backup Server.

7.2. OpenVMS and Multiple Network Engines

OpenVMS does not support multiple network engines.

7.3. Improper Release of ASTs

The Pathway TCP product from Wollongong does not release ASTs properly, causing the SQL Server process to use up its ASTLM quota after some amount of time. The time is dependent upon the number of network connections made and dropped. To delay this problem, increase the ASTLM of the SQL Server through the use of a quota file for SQL Server startup. See Chapter 8 of the *Installation and Configuration Guide* for information on using a quota file.

7.4. Limitations on Packet Size

The **/tdspacketsize** option specifies the network packet size to be used for an **isql** or **bcp** session. The maximum packet size currently supported is 32,256 bytes. Specification of a packet size of 32,768 bytes or greater will result in errors similar to the following:

In the SQL Server error log:

```
kernel nast_readpost: Invalid tdslength (-32768), socket 1,
listener 0
```

In the client error log:

```
SYSTEM-F-LINKDOXCON, network partner disconnected logical link
DB-LIBRARY error: Write to SQL Server failed.
SYBASE-E-BCPFAIL, BCP has failed.
```

7.5. Floating Point Precision

Floating point numbers lose precision because their binary IEEE representation is an approximation of the actual value. For example, 3.14 may appear as 3.13999999.

Because *real* uses fewer bytes, its representation is especially prone to loss of precision.

7.6. Thresholds and Remote Procedure Calls

You cannot execute a remote procedure call (RPC) to another SQL Server from a threshold-activated procedure. RPCs from one SQL Server to another require a user’s name and password to be passed between servers. A threshold procedure does not have a login or password associated with it because it runs as an internal process.

For a workaround, you can execute RPCs to an Open Server from a threshold-activated procedure, and that Open Server can execute an RPC to the second SQL Server. To accept the RPC from SQL Server, the Open Server cannot require any login protocol.

7.7. Buffer Error for Open Databases

If your SQL Server configuration for open databases is less than the number of active databases used in common queries, level 834 errors can occur. To avoid this problem, configure the number of open databases to be equal to or greater than the number of open databases used in common queries. This behavior is described in Bug #86943.

7.8. *drop database* Command

The **drop database** command occasionally does not work if another process has the database locked. The workaround in this situation is to ask the System Administrator to kill the process that’s interfering with dropping the database. For further information, refer to Bug #83148.

7.9. Loss or Corruption of Ownership of a Stored Procedure

If your application has frequent calls to a stored procedure (20 to 30 times every five minutes), the ownership of the stored procedure may be lost or corrupted, resulting in the following error message:

```
Error: 2812, Level 16, State 5: Server ‘MPSPRD’,
Line 1: Stored procedure ‘<stored_procedure>’ not
found. Specify owner.objectname or use sp_help to
check whether the object exists (sp_help may
produce lots of output).
```

As a workaround, include a “with recompile as” statement at the end of your stored procedure. For example:

```
create procedure sp_procedure
( <input/output parameters> )
with recompile as
```

This problem is tracked in Bug #80450.

7.10. View Definitions Using *group by* With No Aggregate Function Reference

View definitions that use **group by** with no reference to an aggregate function in the query return incorrect results. There are two workarounds:

- Use **select distinct** instead of **group by** in the view definition.
- Use an aggregate function in the view definition, perhaps in the select list.

For further information, refer to Bug #85667.

7.11. Setting *memory alignment boundary* to Non-default Values

Setting the SQL Server configuration parameter **memory alignment boundary** to anything other than its default value (2048) causes SQL Server not to boot on restart.

If you encounter this problem, edit your SQL Server configuration file to reset **memory alignment boundary** to its default value. You can do this in either of two ways:

```
memory alignment boundary=2048
```

or

```
memory alignment boundary=default
```

The error message that is generated if you attempt to set **memory alignment boundary** to an out-of-range value is incorrect. The message should read:

```
Bad alignment value encountered. Valid memory
alignment boundary values are multiples of 2048
between 2048 and 16384 inclusive. Units of this
value are bytes.
```

For further information, refer to Bug #80577.

See the *System Administration Guide* for further information on **memory alignment boundary** and editing SQL Server configuration files.

7.12. *sp_unbindcache_all* Failure

sp_unbindcache_all may fail when a database and tables or indexes in the database are bound to a cache in a specific sequence. The error message incorrectly states that the database is in use. This behavior is described in Bug # 79772.

To avoid this failure, bind your cache in the following sequence:

1. Bind the database.
2. Bind the objects to the cache.

Workarounds for the **sp_unbindcache_all** failure are as follows:

- Unbind each object individually.
- Unbind the database and then use **sp_unbindcache_all** to unbind all of the objects.

7.13. *default fill factor percent* and *max_rows_per_page*

If a table is created using the **max_rows_per_page** feature, and then an index is created on that table with the **default fill factor percent** parameter, the **default fill factor percent** parameter will have no effect. For further information, refer to Bug #72732.

7.14. Configuring More User Connections Than Locks

If you configure more user connections than locks, SQL Server fails to boot. The message:

```
process (nnnn)infected with 11
```

appears in the error log, and SQL Server dumps core. For further information, refer to Bug #81209.

7.15. Microsoft Access 2.0 Basic and General Protection Fault

Microsoft Access 2.0 Basic causes a general protection fault (GPF) when it tries to execute a **create table** statement. This is described in Bug #62967.

7.16. Microsoft Access 2.0 Drops Attributes

When Microsoft Access 2.0 exports attached tables back to the database, its attributes (for example, indexes, keys, columns) are lost. This is described in Bug #63190.

7.17. Remote Procedure Calls with Microsoft SQL Server

SQL Server release 11.0.2 currently does not support RPCs to or from ODS-based Microsoft SQL Server. Sybase is investigating the incompatibilities and attempting to enable these RPCs.

7.18. SNMP Subagent and SQL Server

The SNMP subagent release 11.0 is not officially certified to be compatible with SQL Server release 11.0.2. If you use it with SQL Server release 11.0.2 you may encounter the following problem.

The SNMP subagent automatically runs a Transact-SQL query that causes it to disconnect from SQL Server and causes the system table *syscurconfigs* to be deleted. When this happens, a stack trace is written to SQL Server's error log, and SQL Server continues running. However, any subsequent queries TCP/IPueries that read from *syscurconfigs* will fail. To remedy this, shut down and restart SQL Server.

For further information on this problem, refer to Bug #88576.

7.19. SQL Server Arithmetic Operations

Arithmetic operations involving values greater than E+14 or less than E-14 cause a SQL overflow error to occur.

7.20. Highlighted Known Problems with *sybinit*

7.20.1. Interfaces File With a NULL Using TCP/IP

If you create an interfaces file entry using TCP/IP and then enter a NULL value for Name Alias (for example, by entering a Return at the prompt), **sybinit** displays a series of errors and then exits with an **accvio**.

This error is described in Bug# 121439.

7.20.2. Aliases in Interfaces Files

When adding or editing an interfaces file service with the **sybinit** utility, if you enter a new name alias or modify an existing one, and then exit the screen without accepting the alias, you will be unable to use that alias.

If you then re-edit the interfaces file and try to reuse the same name alias, **sybinit** issues the following error message:

```
Interfaces file entry 'name' already exists: duplicate insertion
disallowed. Press <return> to continue.
```

where “name” is the alias you used previously. You must choose another alias name.

If you accept the interfaces alias entry and exit the screen during the initial session, the above error can occur during future **sybinit** sessions using the same interfaces file.

7.20.3. Remap Query Trees

The Remap Query Trees option in the SQL Server Upgrade Menu in **sybinit** only appears after you have completed a successful upgrade. It appears when automatic remapping during the upgrade has failed. If the option appears, you can use **sybinit** to perform remapping. If your upgrade fails during query tree remapping, you may have to remap the query trees manually.

- Run **sp_remap**
- Run **dbcc remap**

If neither of the above suggestions works, drop and re-create the stored procedures.

7.20.4. *sybinit* and Server Connections

On heavily loaded systems, the server may not respond quickly enough to connect to **sybinit**, and the following message appears:

```
Cannot determine the status of server 'server_name': process is
running but unable to log in.
```

where *server_name* is the name of the server **sybinit** is trying to connect to. If you receive this message, try the connection again.

7.20.5. The */log* Command Line Qualifier

If you attempt to use the **/log** command line qualifier at the same time as you use the **/language** qualifier to run **sybinit** in French or German, an access violation occurs. Do not attempt to use the **/log** qualifier for a French or German **sybinit** session. See Appendix B, "sybinit resource Files," of the *SQL Server Installation and Configuration Guide* for more information on **sybinit** command line qualifiers.

7.20.6. Reconfiguring an Existing SQL Server

When you start a **sybinit** session to reconfigure SQL Server, the SQL Server Configuration menu lists *sybsystemprocs* Database Configuration as "Changed" before anything has been changed. This behavior does not prevent you from selecting this menu option and altering SQL Server's *sybsystemprocs* configuration.

When you modify an existing SQL Server interfaces file entry in a **sybinit** session, the SQL Server Configuration menu may not list the Configure Server's Interfaces File Entry as "Changed."

7.21. Backup Server Functionality

The following known problems relate to Backup Server functionality:

7.21.1. Configuring Character Set for Backup Server

If you select ascii_8 as the character set for Backup Server, you may encounter problems with Backup Server messages, especially if Backup Server's language is not U.S. English. Backup Server uses character set information for messages only.

7.21.2. Reconfiguring Backup Server with *sybinit*

If you use **sybinit** to reconfigure Backup Server, **sybinit** may be unable to connect with Backup Server and may return connectivity errors after you have made the configuration changes. If these errors occur, follow these steps:

1. Use **isql** to log into the SQL Server.
2. Use the **shutdown SYB_BACKUP** command to shut down Backup Server.

3. Use **startserver** to restart Backup Server.

7.21.3. Using 4-mm and 8-mm Tapes for Dumps and Loads

Only use data grade tapes for dumps and loads. Do not use 4-mm tapes sold as "digital audio" tapes or 8-mm video tapes for multi-volume dumps or loads. If the dump is too large to fit on one tape, you must use "digital data" 4-mm tapes. Attempts to detect end-of-tape on DAT tapes may cause your hardware to lock up and prevent tape extraction from the drive. If this problem occurs, turn the machine off and on, and then retry extracting the tape.

7.21.4. Case Sensitivity of Device and File Names

Although device and file names on OpenVMS are case-insensitive, the Backup Server records the case used when the names are specified. Various failures can result if the same case is not used each time the name is typed.

7.22. Problems with Upgrade

The following known problems relate to upgrades from previous versions of SQL Server.

7.22.1. Error Message 3444

When SQL Server boots for the first time after upgrading to SQL Server release 11.0.2, SQL Server may display error message 3444 in the error log file:

```
00:95/11/02 13:40:09.36 server  Error: 3444, Severity: 10, State: 1
```

It is safe to ignore this message.

SQL Server issues error message 3444 because all the error messages have not yet been installed. After the upgrade is complete and all the new system messages are installed, SQL Server is able to issue the correct message:

```
00:95/11/06 08:50:09.33 server  Database 'master' is now online.
```

For further information on this problem, refer to Bug #73595.

7.23. Creating a New Device with *alter database*

If the log and data for *sybsystemprocs* share the same device, and you use the **alter database** command to increase space of *sybsystemprocs* onto a new device, the new space will be used for data only, not for both mixed data and log.

If the current device that holds *sybsystemprocs* has sufficient space to accommodate the increased size, the **alter database** command creates additional space for both log and data.

There are two methods for increasing the size of *sybsystemprocs* on a separate device. The first method creates *sybsystemprocs* on a new device; the second method retains your current device, and adds another device for the additional space required for *sybsystemprocs*.

Method 1

To drop and re-create *sybsystemprocs* on a new device:

1. Use **defncopy** to make a copy of any stored procedures that you created or modified.
2. Make sure the device on which you intend to keep *sybsystemprocs* has enough space. *sybsystemprocs* requires a minimum of 21MB if you use this method. If necessary, use the **disk init** command to create a new device.
3. Drop the *sybsystemprocs* database:

   ```
   drop database sybsystemprocs
   ```

4. Re-create the *sybsystemprocs* database on the new device (do not use the **alter database** command in this step):

```
create database sybsystemprocs on device_name = 21
```

5. Run the **installmaster** script to repopulate the stored procedures in *sybsystemprocs*:

```
/user=”sa” /password /input=SYBASE_SYSTEM:[SYBASE.SCRIPTS]INSTALLMASTER
```

6. If you no longer want your old device, run **sp_dropdevice**:

   ```
   sp_dropdevice device_name
   ```

7. Re-create the stored procedures from the copy you made in step 1.

Method 2

sybsystemprocs requires a minimum of 21MB. The space for this database can be split between two devices. For example, if your current device is 10MB, and you do not want to move *sybsystemprocs* to another device, you can create a separate device to hold the additional 11MB required by *sybsytemprocs*.

1. Use **defncopy** to make a copy of any stored procedures that you created or modified.
2. Use the **disk init** command to create a new device. Or find a device that has sufficient room to hold the additional space required by *sybsystemprocs*.
3. Drop the *sybsystemprocs* database:

   ```
   drop database sybsystemprocs
   ```

4. Re-create the *sybsystemprocs* database on the two devices (do not use the **alter database** command in this step):

   ```
   create db sybsystemprocs on dev1 = 10, dev2 = 11
   ```

5. Run the **installmaster** script to repopulate the stored procedures in *sybsystemprocs*:

```
/user=”sa” /password /input=SYBASE_SYSTEM:[SYBASE.SCRIPTS]INSTALLMASTER
```

6. Re-create the stored procedures from the copy you made in step 1.

For further information, see Bug #82062 and #79755.

7.24. Problems with Language Modules and Character Sets

7.24.1. Installing Multiple Languages

If you are installing multiple languages, you should not install them all in a single **sybinit** session. Instead, install each one during a separate **sybinit** session.

For further information on this problem, refer to Bug #88586.

7.24.2. Configuring a Character Set for Backup Server

If you select ascii_8 as the character set for Backup Server, you may encounter problems with Backup Server messages, especially if Backup Server’s language is not U.S. English. Backup Server uses character set information for messages only.

7.25. SQL Server Monitor Functionality

The following known problems pertain to SQL Server Monitor functionality.

7.25.1. Starting Monitor Server

You can boot Monitor Server only after you have booted SQL Server. SQL Server creates the shared memory files Monitor Server uses.

7.25.2. Changing *shared memory starting address* for More Client Connections

In its default configuration, Monitor Server can support approximately five connections. This means that if your Monitor Client is the only one connecting to SQL Server, you can open about five windows.

If multiple clients access the same server, the default configuration may not be able to support the required number of connections. In this case, Monitor Server may refuse connections or display anomalous errors. To remedy this problem, reconfigure the monitored SQL Server to start its shared memory region at a higher virtual address.

➤ ***Note***

Changing the SQL Server **shared memory starting address** configuration parameter causes no overhead or performance penalty for either SQL Server or Monitor Server.

Use the system procedure **sp_configure** to change the value of **shared memory starting address**:

```
sp_configure “shared memory starting address”, value
```

where *value* is the virtual address at which you want SQL Server to start its shared memory region.

8. Language Module and Globalization Issues

This section describes language module and character set functionalities for this release. See “Problems with Language Modules and Character Sets” on page 31 for known globalization and language problems.

8.1. Sybase Character Sets

SQL Server currently supports character sets for sorting, data entry, and data viewing for North and South America, Western Europe, China and Japan. Sybase Character Sets is a new product that provides support for additional regions' character sets, including Eastern Europe, Turkey, and Greece. Sybase has also added support for Arabic, Hebrew, and Korean. Sybase Character Sets can be run on SQL Serve release 10.0 and later.

Conversion between character sets for the same region is fully supported, thus providing interoperability between client and server, even if they do not use the same character set.

Sybase Character Sets can be ordered from your local Sybase sales representative or obtained from the Sybase User Group Forum on CompuServe. For more information, refer to the *Release Bulletin* for Sybase Character Sets.

8.1.1. Character Set Support

Open Client releases 4.6.1, 10.0, 10.0.1, and 10.0.2 and SQL Server releases 10.0, 10.0.1, 10.0.2, 10.1, and 11.0 support supports 7-bit ASCII, 8 bit character sets, and multi-byte character sets (of two or more bytes). This feature may result in a client application reporting that its default character set is not supported in SQL Server.

See the *System Administration Guide* for more information on character sets.

8.2. *.xlt* Files Not Available for Non-iso_1 Character Sets

The *.xlt* files, which are used for character set translation with the **bcp**, **isql**, and **defncopy** when a terminal's character set differs from that of SQL Server, are available only for SQL Server installations that use iso_1 as the default character set. See the *System Administration Guide* for more information about character set conversion. See *SQL Server Utility Programs for OpenVMS* for more information about the **bcp**, **isql**, and **defncopy** utilities.

9. Technical Support

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase

Technical Support. If you have any questions about this installation or need assistance during the installation process, please have a designated person contact Sybase Technical Support or the Sybase subsidiary in your area.

Please have a designated person contact the distributor from which your software was purchased if you do not know how to contact Sybase Technical Support.

10. Auditing Changes in Future Releases

➤ ***Note***

These auditing changes do not affect System 11 auditing. They are described here only to alert you to changes in future releases that may affect your installation.

In the next major releases of SQL Server, the auditing system will provide these enhancements:

- Support for multiple audit tables. This capability enables installations to archive and process audit data with no manual intervention and with no loss of audit records.
- The capability to audit additional events, including the creation of objects; the binding and unbinding of rules, defaults, and messages; all actions by individual users; and all actions performed with a particular role active.
- The addition of a single flag for auditing a set of server-wide security-relevant events.
- A new, streamlined user interface that will allow the SSO to control auditing. In the new interface, which will replace the current auditing interface, a single system procedure provides the capability to set all auditing options.

The next major releases of SQL Server will **not** support **sp_auditoption**, **sp_auditlogin**, **sp_auditdatabase**, **sp_auditobject**, and **sp_auditsproc**. Functions provided by these system procedures will be provided by the new system procedure **sp_audit**. The new auditing system will continue to allow users to add their own audit records with **sp_addauditrecord**.