SQL Server Monitor™
Installation Guide
for Digital OpenVMS Alpha

SQL Server Monitor Release 11.0.1
Document ID: 34586-01-1101-01
Last Revised: October 30, 1996

Principal author: Systems Management Products Group

Document ID: 34586-01-1101

This publication pertains to SQL Server Monitor Release 11.0.1 of the Sybase database management software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

Document Orders

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

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor.

Upgrades are provided only at regularly scheduled software release dates.

Sybase Trademarks

Sybase, the Sybase logo, APT-FORMS, Data Workbench, DBA Companion, Deft, GainExposure, Gain *Momentum*, PowerBuilder, Powersoft, Replication Server, S-Designor, SQL Advantage, SQL Debug, SQL SMART, SQL Solutions, SQR, Transact-SQL, and VQL are registered trademarks of Sybase, Inc. Adaptable Windowing Environment, ADA Workbench, AnswerBase, Application Manager, AppModeler, APT-Build, APT-Edit, APT-Execute, APT-Library, APT-Translator, APT Workbench, Backup Server, BayCam, Bit-Wise, Client-Library, Client/Server Architecture for the Online Enterprise, Client/Server for the Real World, Client Services, Column Design, Connection Manager, DataArchitect, Database Analyzer, DataExpress, Data Pipeline, Datawindow, DBA Companion Application Manager, DBA Companion Resource Manager, DB-Library, Deft Analyst, Deft Designer, Deft Educational, Deft Professional, Deft Trial, Designor, Developers Workbench, DirectCONNECT, Easy SQR, Embedded SQL, EMS, Enterprise Builder, Enterprise Client/Server, Enterprise CONNECT, Enterprise Manager, Enterprise SQL Server Manager, Enterprise Work Architecture, Enterprise Work Designer, Enterprise Work Modeler, EWA, Gateway Manager, InfoMaker, InformationCONNECT, Intermedia Server, InternetBuilder, MainframeCONNECT, Maintenance Express, MAP, MDI, MDI Access Server, MDI Database Gateway, media.play, media.splash, MetaWorks, MethodSet, Movedb, Navigation Server Manager, Net-Gateway, Net-Library, New Media Studio, ObjectCONNECT, ObjectCycle, OmniCONNECT, OmniSQL Access Module, OmniSQL Server, OmniSQL Toolkit, Open Client, Open ClientCONNECT, Open

Restricted Rights

Sybase, Inc., 6475 Christie Avenue, Emeryville, CA 94608.

Table of Contents

About This Book

1. Introduction

2. Installing and Running Monitor Server

List of Figures

List of Tables

About This Book

This manual describes how to load, configure, and run the Monitor Server component of SQL Server Monitor™ release 11.0.1 running on Digital OpenVMS Alpha.

Audience

This installation guide is for individuals who are responsible for:

- Preparing the installation environment
- Installing Monitor Server
- Configuring and updating the interfaces files

How to Use This Book

For general information about SQL Server Monitor architecture, hardware and software requirements, and compatibility issues, see Chapter 1, "Introduction."

For instructions on installing Monitor Server and preparing the run-time environment, see Chapter 2, "Installing and Running Monitor Server."

Related Documents

Installing Sybase Products on Digital OpenVMS describes basic installation concepts and requirements common to all Sybase products, such as setting required system logicals, editing interfaces files, and setting up administrator accounts.

Other Sources of Information

The SQL Server Monitor documentation set includes the following manuals:

- *SQL Server Monitor Release Bulletin* contains release notes and platform-specific information about SQL Server Monitor.
- *SQL Server Monitor Server User's Guide* describes how to set up and use Monitor Server.

- *SQL Server Monitor Historical Server User's Guide* describes how to set up and use Historical Server.
- *SQL Server Monitor Client User's Guide* describes how to use Monitor Client. The manual provides step-by-step instructions for viewing and customizing each graphical display of SQL Server data and describes each display in detail.
- *SQL Server Monitor Client Library Programmer's Guide* describes how to access and use Monitor Client Library, which provides programmatic access to the performance data.

In addition, you can obtain performance and tuning information from the following sources:

- *SQL Server Performance and Tuning Guide*
- Sybase offers a SQL Server "Performance and Tuning" class. For details, contact:

  Mail:

  Education Registrars
  Sybase Professional Services
  77 South Bedford Street
  Burlington, MA 01803

  Phone: (800) 8-SYBASE or (617) 564-6970

  Fax: (800) 792-2733 or (617) 564-6960

  E-mail: registrars@sybase.com

Conventions

This book uses the following conventions.

Text Conventions

Commands and parameters appear in a narrow, bold typeface:

monserver
/user=

Bold typeface also indicates emphasis or indicates that a definition accompanies the highlighted term.

Variables, document titles, file names, and directory names appear in italic typeface:

/user=*myname*
SQL Server Monitor Client User's Guide
interfaces file

If You Need Help

You can get help on SQL Server Monitor from the documentation and from the Technical Support Center.

Technical Support

Each Sybase customer site has a person designated to contact the Technical Support Center. If you cannot solve a problem using the manuals or online help, ask the designated person at your site to contact the Technical Support Center for help.

1 Introduction

This manual describes how to load, configure, and run the following components of SQL Server Monitor release 11.0.1 running on Digital OpenVMS Alpha:

- SQL Server Monitor Server release 11.0.1 (Monitor Server)
- SQL Server Monitor Historical Server release 11.0.1 (Historical Server)

This chapter contains the following topics:

- SQL Server Monitor Architecture
- Hardware and Software Requirements
- Compatibility Requirements

SQL Server Monitor Architecture

SQL Server Monitor consists of four components:

- Monitor Server. An Open Server™ application that obtains performance statistics on a Sybase® SQL Server™ and makes those statistics available to its clients. The clients include Monitor Historical Servers, Monitor Clients, and Monitor Client Library applications. Each SQL Server being monitored must have a unique Monitor Server. These SQL Server / Monitor Server pairs must run on the same machine, since they communicate via shared memory.
- Historical Server. An Open Server application that obtains performance statistics for one or many SQL Servers via Monitor Servers. Historical Server stores these statistics in files which can be accessed later for performance problem identification or trend analysis. A Historical Server can gather statistics from one or many SQL Server / Monitor Server pairs.
- Monitor Client. A graphical user interface that presents performance statistics gathered by Monitor Servers and initiates the gathering of performance statistics by Historical Servers.
- Monitor Client Library. A programming interface that provides access to SQL Server performance data gathered by Monitor Servers. It also provides access to a subset of Historical Server functionality.

Figure 1-1 illustrates the interaction of these components.

Only two of the four SQL Server Monitor components run on Digital OpenVMS Alpha: Monitor Server and Historical Server. You can connect to these servers using Monitor Client or applications written with Monitor Client Library running on client-supported platforms, such as Digital UNIX, Sun Solaris, or Microsoft Windows NT.

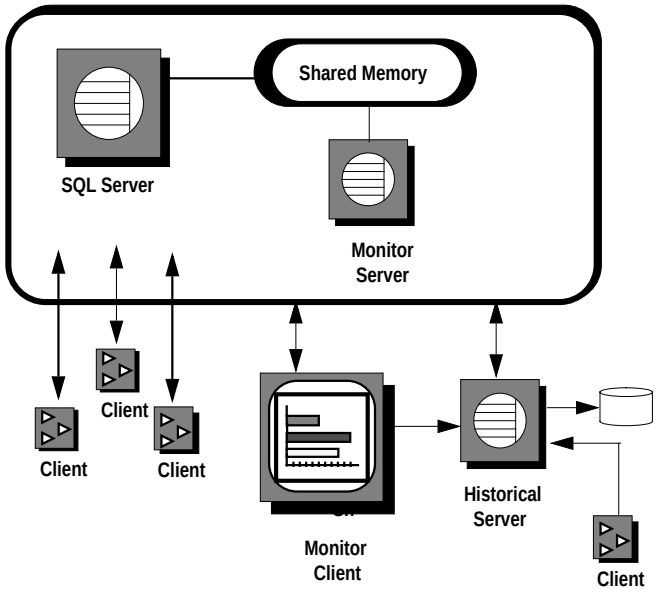

Figure 1-1: SQL Server Monitor architecture

Hardware and Software Requirements

The following hardware and software are necessary for installing and running the components of SQL Server Monitor release 11.0.1 for Digital OpenVMS Alpha.

Hardware

The following table shows the disk space requirements for loading the SQL Server Monitor components.

Component	Disk Space Requirement
Monitor Server	5MB
Historical Server	5MB

In addition, the workstation must have a drive appropriate to the delivery media(tape or CD-ROM disc) on which your Sybase software was received.

Software

Monitor Server release 11.0.1 and Historical Server release 11.0.1 require Digital OpenVMS Alpha version 6.2 or higher.

Compatibility

SQL Server Release Compatibility

Monitor Server release 11.0.1 running on Digital OpenVMS Alpha can monitor SQL Server release 11.0.2 on Digital OpenVMS.

SQL Server Moniitor Component Compatiibility

The components in SQL Server Monitor release 11.0.1 are backward compatible to components in SQL Server Monitor releases 11.0 or later. For example, Monitor Server release 11.0.1 running on Digital OpenVMS Alpha can be accessed by Monitor Client release 11.0 or later from another platform. However, Sybase recommends upgrading all SQL Server Monitor components to release 11.0.1, if possible.

Open Client Release Compatibility

Monitor Server requires Open Server and Open Client™ release 10.0.3. If earlier Open Server releases are used, Monitor Server may shut down. No messages appear on the screen or in the error log.

2 Installing and Running Monitor Server

This chapter describes how to install and run Monitor Server. Topics covered include:

Planning the Installation

This section helps you plan Monitor Server installation and configuration. It contains the following sections:

- "Where to Install Monitor Server"
- "Assumptions"
- "Worksheet for Installing Monitor Server"

Where to Install Monitor Server

Monitor Server must run on the same machine as the SQL Server it monitors. Therefore, you must install Monitor Server on each machine that is the host for a SQL Server you want to monitor.

Assumptions

The instructions in this chapter assume the following:

- Monitor Server software is loaded on your machine using the Sybase installation utility **VMSINSTAL**, as described in this chapter.
- Monitor Server is loaded on the same machine where the SQL Server you want to monitor resides.
- The server to be monitored is SQL Server release 11.0 or later.
- The SQL Server is configured and running.

- You use the SYSTEM user account to load and configure Monitor Server.

Worksheet for Installing Monitor Server

Table 2-1 is a worksheet for planning and performing Monitor Server installation. The worksheet provides an outline of all steps required to complete Monitor Server loading and configuration. The next section provides more information about each item on the worksheet.

Table 2-1: Worksheet for Monitor Server installation

Item	Completed
Back up existing Monitor Server files.	❑ Yes
Verify networking software.	❑ Yes
Log in using the SYSTEM account.	❑ Yes
Load Monitor Server from the delivery media, if it was not loaded with SQL Server. You need the following information to load Monitor Server: 1. SYBASE_SYSTEM: _____ 2. CAS:_____ 3. Is this a secondary installation? ____	❑ Yes
Start SQL Server. SQL Server Name: _____	❑ Yes
Run **installmon** to install Monitor Server stored procedures.	❑ Yes
Determine the SQL Server user account that Monitor Server should use. SQL Server user account: _____	❑ Yes
Create a Monitor Server command procedure. command procedure name: _____	❑ Yes
Add entries to interface file.	❑ Yes

Installation Procedures

This section describes how to load Monitor Server software from the delivery media and how to configure the system in preparation to run Monitor Server.

Back up Existing Monitor Server

If you are upgrading from a previous release of Monitor Server, back up all existing Monitor Server files before proceeding.

Verify Networking Software

Your network configuration must be correct for you to perform the tasks in this guide. Sybase software uses the network even when a client program is connected to a SQL Server on the same machine.

Verify the status of each network you plan to use, as explained in *Installing Sybase Products on Digital OpenVMS Alpha.*

Log On

Log in using the SYSTEM account. See I*nstalling Sybase Products on Digital OpenVMS Alpha* for more information about this account.

Load Monitor Server from Delivery Media

To load Monitor Server, perform the following steps:

1. Place the distribution media in the tape drive or CD-ROM device.
2. (CD-ROM only) Mount the CD-ROM with the following command:

   `$ mount device_name sybase cdrom`

   where *device_name* is the name of the CD-ROM device.

3. Start **VMSINSTAL** with one of the following commands:

Tape	**@sys$update:vmsinstal sybase020 *device_name*** **where *device_name* is the name of the tape drive where you mounted the Sybase tape**
CD-ROM	**@sys$update:vmsinstal sybase020 cdrom:[sybase]**

4. Refer to the information you recorded on the worksheet to answer the following **VMSINSTAL** prompts:
   - **SYBASE_SYSTEM**. Enter the same definition of SYBASE_SYSTEM that was used when SQL Server was installed.
   - **Secondary installation**. Answer "yes" if this is a secondary installation. Otherwise, answer "no."

     A primary installation is intended to be used as the production environment. A secondary installation, on the other hand, is intended to be used as a test environment, or a second primary system. See *Installing Sybase Products on Digital OpenVMS Alpha* for more information about primary and secondary installations.
   - **Customer Authorization String**. Enter the string from the software packaging that allows you to access your products, as recorded on your worksheet.
   - **Sybase products.** The **VMSINSTAL** utility displays a menu listing the available products. Enter the number that corresponds to each product you want to install, pressing Return after each number. When you have entered the number of the last product you want to install on this machine, press Return twice to enter a blank line.
   - **Product confirmation.** The **VMSINSTAL** utility lists the products you have chosen, and prompts for confirmation. Answer:

     "y" or "Y" if the list is correct

     "q" or "Q" to quit

     Any other character to see the menu and choose again
5. Wait while **VMSINSTAL** installs your products**. VMSINSTAL** displays a series of screen messages while it installs your software. Do not

interrupt the installation. When **VMSINSTAL** is finished, it repeats the list of products you selected for installation.

6. (CD-ROM only) When the products have finished loading, dismount the CD-ROM with the following command:

   ```
   $ dismount cdrom
   ```

7. Remove the distribution media from the drive.

Start SQL Server

If you have not done so, start SQL Server.

Install Monitor Server Stored Procedures

Two stored procedures must be installed into the monitored SQL Server. Install these procedures by running the **installmon** batch file. Enter:

```
$ isql /user=username /password=password -
  /server_name=sqlserver -
  /input=SYBASE_SYSTEM:[sybase.scripts]installmon.
```

where *username* and *password* identify a login account with "sa_role" privilege, and *sqlserver* is the name of the SQL Server to be monitored. If *username*, *password*, or *sqlserver* contains lowercase letters, enclose it in quotes. For example:

```
/user="sa"
```

The *installmon* script installs these stored procedures:

- **mon_rpc_attach**. This procedure executes when you start Monitor Server, to attach Monitor Server to SQL Server shared memory. The user identified by the **/user** parameter in the Monitor Server start-up command must have execute permission on **mon_rpc_attach**.
- **mon_rpc_connect**. Monitor Server executes this procedure when a client connects to it to complete the client/server connection. Users must be granted execute permission on **mon_rpc_connect** before they can run a client of Monitor Server.

After successful execution, **isql** exits and returns you to the command line.

Setting User Permissions on Stored Procedures

By default, installmon grants execute permission on **mon_rpc_attach** and **mon_rpc_connect** to the "dbo". Only users who have "sa_role" privilege can use a client of Monitor Server; therefore, any user who is granted execute permission on **mon_rpc_connect** must also have the "sa_role" privilege.

Determine the SQL Server User Account to Use

Monitor Server logs into SQL Server at startup and at shutdown. You specify the user account that Monitor Server should use with the **/user** parameter in the Monitor Server start-up command.

Sybase recommends that you use the "sa" or "sybase" user. However, you can specify any user name that has permission to execute the **master..mon_rpc_attach** procedure installed in the previously described step.

You can create a new user account specifically for Monitor Server. Use **sp_adduser** to create a new user account.

Add a Monitor Server Entry to the *interface*s File

Before SQL Server can be monitored, you must add an entry in the interfaces file for Monitor Server. This entry should contain both master and query lines.

Use the following values in the interfaces file:

- For server name, Sybase recommends using a name that reflects the name of the SQL Server being monitored. For example, if the SQL Server name is SYB1101, you might want to name the Monitor Server MON_SYB1101. (Be sure to limit the name to eleven characters or less.)
- For protocol name, use TCP or DECnet, whichever is appropriate.
- For port, use any valid port number that is not being used by any other server or application. Do **not** use the same port used by SQL Server.

See *Installing Sybase Products on Digital OpenVMS Alpha* for general information about interfaces files and for instructions on using **sybinit** to create new entries in the interfaces file.

Create a Command Procedure for Monitor Server

The most convenient way to start Monitor Server is to save your startup command in a command procedure, and then execute the command procedure.

Create the **RUN_*monitor_server*.COM** command procedure (where *monitor_server* is the name of the Monitor Server.) This command procedure must be placed in the SYBASE_SYSTEM:[sybase.install] directory. To be able to write to this directory, log in as "sybase" or as the account that owns the directory.

The **RUN_*monitor_server*.COM** command procedure sets the DSLISTEN logical to the name of your Monitor Server, defines **monserver** as a foreign command symbol, and executes the **monserver** command.

```
$ define/nolog/process DSLISTEN monitor_server
$ monserver:==$SYBASE_SYSTEM:[sybase.bin]monserver
$ monserver /server=sql_server -
    /monserver=monitor_server -
    /interfaces=interfaces -
    /user=username -
    /password=password -
    /logfile=logfile -
    /connections=max_connections
```

where italics indicates parameter values as described in Table 2-2 on page 2-8.

An example of a **RUN_monServerName.COM** command procedure follows:

```
$ define/nolog/process DSLISTEN MON_SYB1101
$ monserver:==$SYBASE_SYSTEM:[sybase.bin]monserver
$ monserver /server=SYB_1101 -
    /monserver=MON_SYB_1101 -
    /interfaces=sybase_system:[sybase]interfaces. -
    /user="sa" -
    /password="dog" -
    /logfile=MON_SYB1101.log -
    /connections=8
```

➤ ***Note***

Although the above examples include both the DSLISTEN and the **/monserver=** parameter, only one is necessary.

If you do not want to specify the password in the startup command, where it might be visible to others, you can omit the **/password** parameter. In that case, the startup command prompts you for the password.

Monitor Server Start-Up Command Parameters

The Monitor Server startup command is **monserver**. The following table describes all options and arguments to the command.

Table 2-2: Monitor Server start-up command parameters

Parameter and Argument	Description
/connections=*max_connect*	Maximum number of client connections allowed. The default is 5; you can increase the maximum to 20. See "Special Considerations" on page 2-17 for details.
/interfaces=*interfaces*	File specification of the interfaces file to use. The name *SYBASE:interfaces.* is the default interfaces file.
/localconfig=*configfile*	Name of the configuration file. You can use this file to override default configuration values. See "Configurable Parameters" on page 2-14 for details.
/logfile=*logfile*	File specification of the log file in which informational and error messages are logged. The default is *ms.log* in the current directory.
/monserver=*monitor_server*	Name of the Monitor Server to start. This parameter is required if you have not set the DSLISTEN logical.
/password=*password*	Password of the user specified with **/user**. If **/password** is specified without a following argument, the command uses a null password. If **/password** is omitted, you are prompted for the password.
/server=*sql_server*	Name of the SQL Server to monitor. This parameter is required if you have not set the DSQUERY logical.

Table 2-2: Monitor Server start-up command parameters

Parameter and Argument	Description
/trace=*traceflagnum*	0 - If error messages received from SQL Server are not to be printed. 1 - If Monitor Server should not be shut down, even if SQL Server goes down. See "Heartbeat Interval" on page 2-15 for more information.
/user=*username*	User name of a valid login account in the SQL Server being monitored. The *username* must have permission to execute the **master..mon_rpc_attach** stored procedure.
/version	Prints the Monitor Server version string and exits.

Monitor Server Operation

This section describes how to start, verify operation, and stop Monitor Server. It also describes how to connect to Monitor Server to view the data it collects and how to configure run-time parameters.

Setting SYBASE_SYSTEM and SYBASE Logicals

The SYBASE logical must point to a SQL Server / Open Server release 11.0 or later directory. Monitor Server is linked with Open Server System 11 and has dependencies on the SYBASE_SYSTEM:[SYBASE.LOCALES] and SYBASE_SYSTEM:[SYBASE.CHARSETS] directory.

By default, the SYBASE logical is used to determine the location of the interfaces file. If the interfaces file that contains entries for the SQL Server and Monitor Server is not in the directory indicated by the SYBASE logical, use the **/interfaces** parameter to the Monitor Server startup command to indicate the directory and file name of the interfaces file.

SYBASE products assume that the SYBASE logical points to the SYBASE directory under the directory pointed to by the rooted logical SYBASE_SYSTEM. The definition of the SYBASE logical should be SYBASE_SYSTEM:[SYBASE].

For example:

```
$ DEFINE/SYSTEM/EXECUTIVE/TRANSLATION=CONCEALED -
  SYBASE_1101 DUA0:[REL1101.]
```

```
$ DEFINE/SYSTEM/EXECUTIVE SYBASE_SYSTEM -
  SYBASE_1101:
$ DEFINE SYBASE SYBASE_SYSTEM:[SYBASE]
```

Creating Quota Specification File

Create a quota specification file with appropriate VMS resource quotas. Create the file in the SYBASE_SYSTEM:[sybase.install] directory, using a name that reflects the name of your Monitor Server. For example, if the Monitor Server name is MON_SYB1101, you might name the quota specification file QUOTA_MON_SYB1101.DAT.

The SYBASE installation directory includes a sample quota specification file, SYBASE_SYSTEM:[sybase.install]SAMPLE_QUOTA.DAT, which you can copy and use as a starting point for determining appropriate quota values for Monitor Server.

The number of virtual pages of memory required (limited by the page file quota) for the Monitor Server may be up to 50% greater than the virtual pages required by the SQL Server. To find the value used by the SQL Server, enter:

```
$ SHOW PROC/CONTINUOUS/ID=XXXXXXXX
```

where *XXXXXXXX* is the process ID of the SQL Server process. The number of virtual pages used by the SQL Server is listed in the upper right corner of the display.

The PGFLQUOTA value in the sample quota file may be sufficient, but if it is not, increase the value, and also to be sure that the PGFLQUO quota setting for the SYBASE account and the VIRTUALPAGECNT parameter for SYSGEN are both high enough to accommodate the higher setting in the Monitor Server quota file.

Starting Monitor Server

To start Monitor Server, follow these steps:

1. Start the SQL Server you want to monitor.
2. Log in using the same system account that you used to start SQL Server. For example, if you were logged in using the “sybase” account when you started SQL Server, you must also use the “sybase” account to start Monitor Server. It is recommended that

you use the "sybase" account to start both the SQL Server and Monitor Server.

3. Change to the SYBASE_SYSTEM:[sybase.install] directory with the command:

   ```
   $ set default SYBASE_SYSTEM:[sybase.install]
   ```

4. Confirm that there is a foreign command symbol defined for STARTSERVER. It should be pointing at the **startserver** executable in the INSTALL subdirectory where SQL Server is installed.

5. Enter the following command to start Monitor Server:

   ```
   $ startserver /openserver=monitor_server -
     /quota=quota_file
   ```

   where:

   monitor_server is the name of the Monitor Server you want to start. A Monitor Server of this name must be defined in the interfaces file and a **RUN_***monitor_server* **.COM** command procedure must exist in the SYBASE_SYSTEM:[sybase.install] directory. The **startserver** command executes this **RUN_***monitor_server***.COM** command procedure. The **RUN_***monitor_server***.COM** command procedure contains the Monitor Server startup command (**monserver**) and parameters.

   quota_file is the path name of the VMS resource quota file you created for Monitor Server. If you do not provide a full path name, **startserver** looks for the file in the current directory.

The **startserver** command creates an output file named START_*monitor_server*.OUT, where *monitor_server* is the Monitor Server name. This file contains error messages if the Monitor Server fails to start. If startup is successful, the file contains version and copyright information about Monitor Server, followed by configuration information. If start is successful, the final line in the file is:

```
Initialization is over. Ready to accept connections
```

Monitor Server also writes messages to its log file. However, these messages do not appear until Monitor Server terminates.

The default name of the Monitor Server error log is *ms.log*. Its default location is the directory from which the **monserver** command was invoked. The **/logfile** parameter to the **monserver** command overrides the default name and location.

If Monitor Server fails to start, check both the START_monitor_server.OUT file and the Monitor Server log file for

error messages. For assistance in determining the cause of the problem and the action you should take, see “Troubleshooting” on page 2-20.

Verifying That Monitor Server Is Running

To verify that Monitor Server started correctly, run **isql** with the following command:

```
$ isql /user=username /password=password -
  /monserver=monitor_server
```

where *monitor_server* is the name of Monitor Server, *username* is the name used in the **/user** parameter when the server was started, and *password* is the password associated with *username*.

Then, execute the following **isql** statements:

```
1> sms_status server
2> go
```

You should see the following output:

```
Server Name
--------------------------------
sql_server
```

where *sql_server* is the name of the SQL Server being monitored by your Monitor Server.

Stopping Monitor Server

Monitor Server shuts itself down when it detects that SQL Server is down. See “Heartbeat Interval” on page 2-15 for more information.

To manually shut down Monitor Server:

1. Connect to Monitor Server using **isql**:

   ```
   $ isql /user=username /password=password -
       /monserver=monitor_server
   ```

   where *username* is either “sa” or the same user name that was specified with the **/user** parameter when Monitor Server was started; *password* is the password associated with *username*, and *monitor_server* is the name of the Monitor Server you want to stop.

2. When the **isql** prompt appears, issue the **sms_shutdown** command as follows:

```
1> sms_shutdown [no_wait]
2> go
```

The **no_wait** option allows shutdown to occur immediately, even when an outstanding command from another client connection exists against Monitor Server. However, since Monitor Server commands do not take long to complete, the **no_wait** option does not gain you much time.

Connecting to Monitor Server

SQL Server Monitor release 11.0.1 supports all SQL Server release 11.0.x platforms. You can connect to Monitor Server by using any of the following clients:

- Monitor Client release 11.0.x, running on a client supported platform, such as Digital UNIX, Sun Solaris 2.x (SPARC), or Microsoft Windows, Windows 95, or Windows NT.

  See the *SQL Server Monitor Client User's Guide for Microsoft* or *SQL Server Monitor Client User's Guide for UNIX* for information on running Monitor Client.

- Monitor Historical Server release 11.0.x, running on client supported platforms. See the *SQL Server Monitor Historical Server User's Guide* for information on the Historical Server command-line interface. Also, the Record feature in Monitor Client provides an interactive interface to Monitor Historical Server.

- A user-written application linked with Monitor Client Library release 11.0.x. See the *SQL Server Monitor Client Library Programmer's Guide* for information on creating and running your own SQL Server Monitor client applications.

Specifying *Interfaces* File Entries on the Client Machines

For clients to connect to Monitor Server, the interfaces files on the client machines must contain entries for both the Monitor Server and the SQL Server being monitored. Clients to Monitor Server can be Monitor Client, Monitor Historical Server, or user applications linked to the Monitor Client Library.

The client machine may or may not be a Digital OpenVMS machine. If it is a Digital OpenVMS machine, use **sybinit** on the client machine to add appropriate entries to the *interfaces* file. If the client is running on a Microsoft Windows machine, use **sqledit** to add entries to the

sql.ini file. If the client is running on a UNIX-based platform, use **sybinit** to edit the *interfaces* file.

In any case, you need two entries in the interfaces file on the client machines: one entry for SQL Server and one for Monitor Server. Use the following values in the client's interfaces file to describe the servers:

- For the server names, you must use the same names that were used when the servers were configured. For example, if you configured an SQL Server named SYBSQL02, the entry in the client interfaces file must identify the server as SYBSQL02.
- For protocol name, use TCP or DECnet, whichever is appropriate.
- For port numbers, use the values you used when you configured the servers. Remember that SQL Server and Monitor Server each have a unique port number. You must use the correct values for each corresponding entry in the client's interface file.

The installation and configuration documentation for the client component contains more details appropriate to the client's hardware platform.

Configurable Parameters

You can configure the following Monitor Server parameters:

- Event buffer scan interval
- Heartbeat interval
- Internal buffer size
- Maximum number of buffers per summary
- Maximum number of summaries per client connection

These parameters are stored in the configuration file specified by the **/localconfig** parameter in the Monitor Server start-up command.

Event Buffer Scan Interval

Many Monitor Client windows and Monitor Historical Server views cause summaries of monitoring events to be created by the Monitor Server. If one or more clients have enabled event summaries, Monitor Server performs regular scans of the SQL Server engines' event buffers. The frequency of these scans is computed by Monitor Server based on the number of event buffers configured. While it is

very unlikely that these scans will lose any event data if the event buffers are of a reasonable size, you can manually set the **scan_interval** option to change the interval between event buffer scans. Insert or change the following line in the Monitor Server configuration file:

```
scan_interval value
```

where the *value* is specified in milliseconds. The minimum valid scan interval value is 1.

Heartbeat Interval

Monitor Server periodically checks to see whether SQL Server is running by examining appropriate counter values. This is called the heartbeat mechanism. By default, the frequency of checking is once every 120 seconds. You can edit the configuration file to change the **heartbeat_interval**:

```
heartbeat_interval value
```

where *value* is specified in seconds.

➤ ***Note***

To bypass the heartbeat mechanism, use the **/trace** parameter when you invoke Monitor Server. That parameter causes Monitor Server to continue to run even if the SQL Server being monitored is down.

Configuring Monitor Server Heap Space Usage

Monitor Server collects information from two sources in SQL Server shared memory space:

- **Counters** collect information at a high level, such as the number of times a certain activity occurs server-wide or per engine or per database device.
- **Events** are data structures in SQL Server shared memory containing detailed information about activities in SQL Server. Monitor Server collects information from these events and summarizes it. Monitor Server maintains **event summaries** at a level of detail specified by its clients.

Monitor Server allocates the following internal buffers in heap space to hold this information:

- For each client connection, when the connection initially occurs, Monitor Server allocates a small buffer to hold a snapshot of counter values before the values are sent to the client.
- For each event summary request enabled over a client connection, Monitor Server allocates a summary buffer. If a summary buffer fills up, Monitor Server allocates additional buffers dynamically. The size and number of these buffers are controlled by three Monitor Server configuration parameters:
  - **bufsize** controls the size of each summary buffer
  - **max_mem_blks** controls how many buffers may be allocated for each event summary request
  - **max_summaries** controls how many event summaries may be requested by each client connection

  The amount of heap space per summary buffer is typically 32KB, but it can vary with the volume of data encountered. There is a risk that, if the Monitor Server uses too much heap space, its heap will collide with the location at which SQL Server shared memory is attached. In that case, you should reconfigure and reboot SQL Server so that it uses a higher virtual memory starting address, which allows Monitor Server to do the same. This effectively provides more heap space to Monitor Server.

Summary Buffer Size

The **bufsize** parameter controls the size of each buffer allocated for summarized event data. The configuration file entry for this parameter is:

```
bufsize value
```

where *value* is specified in kilobytes. The maximum valid value is 16,384, which equals 16MB. The minimum buffer size is 32K, which is also the default value.

Number of Summary Buffers per Event Summary Request

The Monitor Server allocates buffers dynamically to accumulate summaries of event-related data. Monitor Server acquires a separate set of buffers on behalf of each enabled summary. The **max_mem_blks** parameter controls the maximum number of buffers that may be allocated to any one summary.

The configuration file entry for this parameter is:

```
max_mem_blks value
```

The maximum valid value is 1024. The minimum valid value is 1. By default, up to 32 summary buffers may be acquired for a single summary request.

Maximum Number of Event Summaries per Connection

By default, a maximum of 15 summary requests may be enabled concurrently on a single client connection. You can reset this maximum by specifying the **max_summaries** parameter.

The configuration file entry for this parameter is:

```
max_summaries value
```

where *value* is the maximum number of event summaries that may be active on a client connection. The maximum value is 1024. The minimum value is 1.

Special Considerations

This section describes issues to consider when you run Monitor Server.

Passwords in Runserver File

Although putting a password in a runserver file is not normally recommended or even necessary, it is the easiest method for starting Monitor Server. If this is unacceptable, remove the password specifier from the runserver file. Enter the password when prompted.

Increasing the Maximum Number of Client Connections

In its default configuration, Monitor Server supports up to five connections. Each window opened by a Monitor Client requires a separate connection to the Monitor Server. If yours is the only Monitor Client accessing the server, you can open a maximum of five windows. However, if multiple clients are accessing the same server, the **combined** number of window connections for all clients is still five.

You can increase the allowable Monitor Server client connections to as many as 20 by including the **/connections** parameter in the Monitor Server startup command.

Controlling Event Buffer Overruns

Monitor Server uses several mechanisms to collect data. One source of data is low-level SQL Server monitoring events, which each SQL Server engine writes to its own event buffer in shared memory. Monitor Server scans the event buffers at regular intervals and summarizes the events according to client specifications.

The writing of event records by SQL Server and reading of events by Monitor Server are not directly synchronized in any way. This is essential to avoid impacting the throughput of SQL Server. Therefore, if the size of the event buffers configured is too small, or if the scan interval is too long, it is possible for event buffer overruns to occur and for events to be lost.

The frequency of event buffer scans is normally determined automatically by Monitor Server. The frequency is based on the number of events that can be stored in the event buffers configured in SQL Server. The computed interval between scans varies in direct proportion to the size of the event buffers. You can also override Monitor Server's computed scan interval by specifying the **scan_interval** parameter in the Monitor Server configuration file. In either case, the initial scan interval is dynamically reduced by Monitor Server if it finds that the event buffers are being filled too rapidly.

Event Buffer Sizing for SQL Server Release 11.0.x

The size of the event buffers is controlled by the **"event buffers per engine"** SQL Server configuration parameter, which specifies the maximum number of events to be buffered (per engine). The default event buffer size is 100, a number which most users will want to increase. Trade-offs are involved in selecting an optimal buffer size, since a low number forces more frequent scans by Monitor Server, but a large buffer size reduces the amount of memory available for other uses, such as procedure cache.

The recommended procedure is to first reset the event buffer size to a somewhat larger number. A size of 2000 generally is large enough to avoid event buffer overruns. However, you may want to increase the event buffer size beyond the minimum size that prevents

overruns because larger event buffers decrease the frequency of scans by Monitor Server.

To increase the size of the event buffer in SQL Server memory:

1. Decide on a value for event buffer size.
2. Make sure that SQL Server is running and Monitor Server is not running.
3. Use **isql** to connect to SQL Server.
4. Issue the following command:

   ```
   1> sp_configure "event buffers per engine", size
   2> go
   ```

   where *size* is the desired event buffer size.
5. Shut down SQL Server, restart it, and verify that it starts correctly.
6. Restart Monitor Server, and verify that it starts correctly.

   To verify that Monitor Server started correctly, run **isql** with the following command:

   ```
   $ isql /user=username /password=password -
     /monserver=monitor_server
   ```

   where *username* is the name specified with the **/user** command-line option when starting Monitor Server, *password* is that user's password, and *monitor_server* is the Monitor Server to start.

   Execute the following **isql** statements:

   ```
   1> sms_status server
   2> go
   ```

   You should receive the following output:

   ```
   Server Name
   ---------------------------------
   sql_server
   ```

   where *sql_server* is the name of the SQL Server that your Monitor Server is monitoring.

Verifying Event Buffer Size

To verify that the correct event buffer size is in effect, execute the following **isql** statements:

```
1> sms_status numeventbuf
2> go
```

You receive the following output:

```
Number of Event Buffers
-----------------------
                  size
```

where *size* is the event buffer size you have just configured for the SQL Server being monitored.

Troubleshooting

The following informational and error messages, which are listed alphabetically, may appear in the Monitor Server error log. Possible causes and resolutions follow each error message.

`Allocation failure for <n> bytes: all configured summary buffers in use.`

Action: Modify the Monitor Server configuration file to increase the value of the **bufsize** or the **max_mem_blks** parameter.

`Event buffer wrap: <n> events lost.`

Low-level monitoring information is being written by SQL Server into its event buffers more rapidly than Monitor Server can extract it.

The size of the event buffers is controlled by a SQL Server configuration parameter, which specifies the maximum number of events to be buffered (per engine). The default event buffer size is 100, a number which most users will need to increase. Trade-offs are involved in selecting an optimal buffer size, since a low number forces more frequent scans of the event buffers by Monitor Server, but a large buffer size reduces the amount of memory available for other uses, such as procedure cache.

See “Event Buffer Sizing for SQL Server Release 11.0.x” on page 2-18 for instructions on resetting the event buffer size.

After enlarging event buffer size, it is unlikely, but still possible, that event loss may occur. In that case, messages related to *event buffer wrap* will appear in the Monitor Server log file. Upon detecting event loss, Monitor Server dynamically reduces its scanning interval from its initially computed value to a value at which event loss no longer occurs. You can determine this recomputed scanning interval by using the **sms_status scan_interval** command. Enter the value returned by this command in the configuration file as the value of the

scan_interval parameter. This causes Monitor Server to scan at or below this interval, rather than starting at its calculated interval whenever it is booted.

You may also avoid event buffer overruns by setting the **scan_interval** configuration file parameter to a fixed number of milliseconds, and increase the size of the SQL Server event buffers to a point where event buffer overruns never occur. This technique limits the frequency of event buffer scans, at the expense of absorbing more SQL Server memory.

```
Failed to allocate a chunk of <n> bytes for summarized event
data.
```

The starting address of the shared memory segment that Monitor Server is sharing with SQL Server is too low.

Action: Start SQL Server with a higher virtual memory starting address, which allows Monitor Server to do the same, which allows Monitor Server to use more heap space.

The problem is independent of the amount of memory SQL Server starts with as dictated by the **sp_configure** memory command. It is dependent on the amount of memory available between the size of the Monitor Server binary and the starting address of shared memory.

```
ftok(<$SYBASE/install/sql_servername>.krg) failed: No such file
or directory.
Fatal error -1 detected
```

where $*SYBASE/install/sql_servername* is the path to the SQL Server environment.

Monitor Server is unable to locate the SQL Server *.krg* file. The **memory** parameter value when starting SQL Server must match the **memory** parameter value when starting Monitor Server.

Action: Monitor Server looks for the shared memory file, *servername.krg*, that SQL Server created, where *servername* is the SQL Server name. If the operating system has a limit of 14 characters for any file name, the SQL Server name must consist of ten or fewer characters to allow for the *.krg* suffix in the shared memory file name.

```
Monitor Server has encountered a fatal error and is quitting.
```

Monitor Server detected an unrecoverable error condition and is terminating. A message describing the error condition is written to the Monitor Server log file, if possible. The exit status of Monitor Server may be of use to Sybase Technical Support in diagnosing the problem if the log file does not provide sufficient information.

```
Msg 2812, Level 16, State 4
Stored procedure 'master..mon_rpc_attach' not found. Specify
owner.object
name or use sp_help to check whether the object exists (sp_help
may produce lots of output).
DB-Library error:
    General SQL Server error: Check messages from the SQL Server.
Failed to get shared memory attach information.
Fatal error -1 detected
```

Monitor Server called a stored procedure in SQL Server that does not exist. The stored procedure was not created.

Action: Create the stored procedure by following the steps in "Install Monitor Server Stored Procedures" on page 2-5.

```
msgid:20073 -- os_attach_region: shmget(0xc70e0703): No such
file or directory
msgid:20111 -- kbattach: couldn't attach to Kernel region
Unable to attach with shared memory.
Fatal error -1 detected
```

If a user account starts Monitor Server but does not have read / write permissions on the SQL Server shared memory file (*server_name.krg*), or the user's account that started Monitor Server does not have the same level of permissions as does the user account that started SQL Server, Monitor Server fails to attach to the SQL Server shared memory.

Action: Log in with the same account that started SQL Server and attempt to start Monitor Server again.

```
msgid:20112 -- kbattach: attached to wrong Kernel region
Unable to attach with shared memory.
Fatal error -1 detected
```

A version mismatch between Monitor Server and SQL Server exists. SQL Server Monitor release 11.0.1 can monitor only SQL Server release 11.0 or release 11.0.x.

Action: Be sure that the correct SQL Server Monitor release is attached to a supported SQL Server release.

```
<n> failures to allocalte space for summarized data - no more
will be logged.
```

Action: Check earlier messages in the error log file that describe failed attempts to allocate memory for summarizing event-based monitoring information. Take the actions prescribed for those error messages.

```
No server log file open; Using stderr for log.
Open Server:  Error: 16012/10/0: Can't open log file
'$SYBASE/install/ms.log'
```

```
Openserver srv_init call failed.
Monitor Server has encountered a fatal error and quitting
```

Action: Start Monitor Server as a user that has permission to read and write to the Monitor Server errorlog file.

```
Open Server: Fatal Error: 16029/20/0: Failed to start any
network listeners
```

A user attempted to start Monitor Server while it was already up and running or specified a port number in the interfaces file that is already in use by another application.

Action: Use **isql** to connect to Monitor Server to see if it is already up. If not, use a different port number in the interfaces file and restart Monitor Server.

```
Open Server:  Error: 16104/10/1: Unable to allocate stack, size
34816, for new thread
```

The starting address of the shared memory segment that Monitor Server is sharing with SQL Server is too low. It is not a fatal error for Monitor Server, but Monitor Client cannot open additional windows.

Action: Start SQL Server with a higher virtual memory starting address, which allows Monitor Server to do the same, which allows more Monitor Client connections to Monitor Server.

The problem is independent of the amount of memory SQL Server starts with as dictated by the **sp_configure** memory command. It is dependent on the amount of memory available between the size of the Monitor Server binary and the starting address of shared memory.

```
SQL Server seems to be down. Shutting down the monitor server.
```

If SQL Server is down for approximately 2 minutes, Monitor Server shuts down automatically.

Action: Start SQL Server, and then start Monitor Server.

You can use the heartbeat interval configuration parameter to specify the time that Monitor Server waits before it shuts down after SQL Server appears to be down.

```
Unable to attach with shared memory.
```

SQL Server creates the *servername.krg* file at boot time. The date of this file should match the last SQL Server start-up date.

Action: Verify this by checking the SQL Server errorlog file. If it does not match, Monitor Server was trying to access an old version of this *.krg* file. Do not move the *.krg* file after SQL Server starts.

The value specified for the **/M** parameter when starting SQL Server must match the value for the **/m** parameter when starting Monitor Server.

Action: From the command line or from within the run server files for SQL Server and Monitor Server, restart SQL Server and then Monitor Server using these correctly set parameter.

What's Next?

See "Connecting to Monitor Server" on page 2-13 for various ways to connect to Monitor Server and use the information it gathers.