IRIS FailSafe™ Sybase® Administrator's Guide

Document Number 007-3306-002

CONTRIBUTORS

Written by Susan Ellis
Illustrated by Dany Galgani
Production by Allen Clardy
Engineering contributions by Padmanabhan Sreenivasan and Pablo Sanchez
St Peter's Basilica image courtesy of ENEL SpA and InfoByte SpA. Disk Thrower image courtesy of Xavier Berenguer, Animatica.

IRIS FailSafe™ Sybase® Administrator's Guide
Document Number 007-3306-002

Contents

List of Examples v

List of Figures vii

List of Tables ix

About This Guide xi
Audience xi
Structure of This Document xi
Related Documentation xii
Conventions Used in This Guide xiii

1. **IRIS FailSafe Sybase Overview** 1
Sybase Failover 1
IRIS FailSafe Sybase SQL Server Monitoring 1
    Sybase Database Agent 2
    Monitoring Stored Procedure 3
Overview of Configuring IRIS FailSafe for Sybase 4

2. **Configuring IRIS FailSafe for Sybase** 5
Required Software 5
Planning Sybase Configuration 6
    Choosing a Sybase Configuration 6
    Example Sybase Configuration 7
    Configuration Parameters for Sybase 7
    Monitoring Sybase Servers 8
Customizing the Monitoring Stored Procedure 9
Adding Sybase Information to the Configuration File 9
Installing the Configuration File 10
Testing Sybase Failover 10

3. Configuration File Blocks for Sybase 15

Sybase Application-Class Block 15

Sybase Blocks 16

Sybase Action and Action-Timer Blocks 20

A. Messages About Configuration File Errors 23

List of Examples

Example 3-1 Sybase Application-Class Block 15
Example 3-2 Sybase Block 16
Example 3-3 Sybase Action and Action-Timer Blocks 20

List of Figures

Figure 1-1 Sybase Database Agent Component in IRIS FailSafe 2

Figure 3-1 Monitoring Frequency Configuration Parameters in Sybase Blocks 19

Figure 3-2 Monitoring Configuration Parameters in the Action-Timer Block for Sybase 21

List of Tables

Table 2-1 Sybase Configuration Parameters 7

Table 3-1 Values for Monitoring Frequency Configuration Parameter in Sybase Blocks 19

Table 3-2 Values for Monitoring Frequency Configuration Parameter in Sybase Action-Timer Blocks 21

About This Guide

This guide provides information about configuring IRIS FailSafe™ systems with the IRIS FailSafe Sybase® database option. This option enables Sybase database resources to be failed over from one node to another if a component fails. This guide is intended as a supplement to the information about configuring IRIS FailSafe included in the *IRIS FailSafe Administrator's Guide*.

This guide was prepared in conjunction with release 1.2 of the IRIS FailSafe Sybase option. It describes IRIS FailSafe Sybase software for Sybase 10 and Sybase 11. Sybase EBF 6990 for the SQL Server and EBF 7469 for the Backup Server (emergency bug fixes) were used to certify IRIS FailSafe on IRIX 6.4 systems.

Audience

This guide is written for system administrators who are responsible for configuring and administering an IRIS FailSafe system with the optional IRIS FailSafe Sybase software. These administrators must be familiar with Sybase configuration and stored procedures.

Structure of This Document

This guide contains three chapters and an appendix:

- Chapter 1, "IRIS FailSafe Sybase Overview," describes the IRIS FailSafe components that are included in the IRIS FailSafe Sybase option. It also provides an overview of the steps required to add Sybase databases as highly available services to an IRIS FailSafe cluster.
- Chapter 2, "Configuring IRIS FailSafe for Sybase," explains the procedures for adding Sybase databases as highly available services.

- Chapter 3, “Configuration File Blocks for Sybase,” contains reference information for the Sybase blocks in IRIS FailSafe configuration files.
- Appendix A, “Messages About Configuration File Errors,” explains the error messages and warnings from the *ha_cfgverify* command that are related to Sybase.

Related Documentation

For Sybase installation information, see the *Sybase SQL Server™ Installation Guide for Silicon Graphics IRIX.*

Besides this guide, other documentation for the IRIS FailSafe system includes

- *IRIS FailSafe Administrator's Guide*
- *IRIS FailSafe Programmer's Guide*
- *IRIS FailSafe INFORMIX Administrator's Guide* (IRIS FailSafe INFORMIX® option)
- *IRIS FailSafe Oracle Administrator's Guide* (IRIS FailSafe Oracle® option)

The IRIS FailSafe reference pages are:

- ha_admin(1M)
- ha_appmon(1M)
- ha_cfgchksum(1M)
- ha_cfginfo(1M)
- ha_cfgverify(1M)
- ha_exec(1M)
- ha_hbeat(1M)
- ha_ifa(1M)
- ha_ifmx(1M) (IRIS FailSafe INFORMIX option)
- ha_killd(1M)
- ha_nc(1M)
- ha_orcl(1M) (IRIS FailSafe Oracle option)
- ha_spng(1M)

- ha_sybs(1M) (IRIS FailSafe Sybase option)
- http_ping(1M) (IRIS FailSafe Web option)
- macconfig(1M)
- ha.conf(4)
- failsafe(7M)

Release notes are included with each IRIS FailSafe product. The names of the release notes are as follows:

ha_base	release notes for IRIS FailSafe
ha_ sybs	release notes for IRIS FailSafe Sybase

Conventions Used in This Guide

These type conventions and symbols are used in this guide:

Bold	Literal command-line arguments and literal parameter values
Italics	Command names, filenames, new terms, the names of *inst* subsystems, manual/book titles, variable command-line arguments, and variables to be supplied by the user in examples, code, and syntax statements
`Fixed-width type`	Examples of command output that is displayed in windows on your monitor and of the contents of files
`Bold fixed-width type`	Commands and text that you are to type literally in response to shell and command prompts
#	IRIX™ shell prompt for the superuser (*root*)

Chapter 1

IRIS FailSafe Sybase Overview

This chapter provides information about the components that are added to the base IRIS FailSafe product by the IRIS FailSafe Sybase option, and assumes that you are familiar with the basic components of IRIS FailSafe described in Chapter 1 of the *IRIS FailSafe Administrator's Guide*. This chapter also lists the steps you take to add Sybase SQL Servers to the highly available services that are failed over on an IRIS FailSafe cluster. The major sections in this chapter are as follows:

- "Sybase Failover" on page 1
- "IRIS FailSafe Sybase SQL Server Monitoring" on page 1
- "Overview of Configuring IRIS FailSafe for Sybase" on page 4

Sybase Failover

IRIS FailSafe provides high availability for Sybase SQL Servers by monitoring them and quickly failing them over to the other node in the cluster if a failure is detected and IRIS FailSafe has been configured to fail over when a failure is detected. The clients connected to the Sybase SQL Servers that have been failed over experience a disconnection. It is the client's responsibility to handle the disconnection by trying to reconnect until the SQL Servers (now on the other node) respond. Refer to Chapter 1 of the *IRIS FailSafe Administrator's Guide* for more information about the interruption of services for clients.

IRIS FailSafe Sybase SQL Server Monitoring

The IRIS FailSafe Sybase option includes three components that perform monitoring of Sybase SQL Servers: the monitoring script */var/ha/actions/ha_sybs_lmon*, the database agent */usr/etc/ha_sybs*, and a Sybase stored procedure called *sp_failsafe_mon*. IRIS FailSafe Sybase does not monitor Sybase Backup Servers, but it can be used to start and shutdown Sybase Backup Servers. (This is controlled by the optional backup-server parameters in sybase blocks in the configuration file */var/ha/ha.conf*. See the section "Sybase Blocks" in Chapter 3 for more information.)

Sybase Database Agent

The IRIS FailSafe Sybase option includes a database agent, */usr/etc/ha_sybs*, which monitors Sybase SQL Servers to determine their liveliness. It interacts with other IRIS FailSafe components as shown in Figure 1-1.

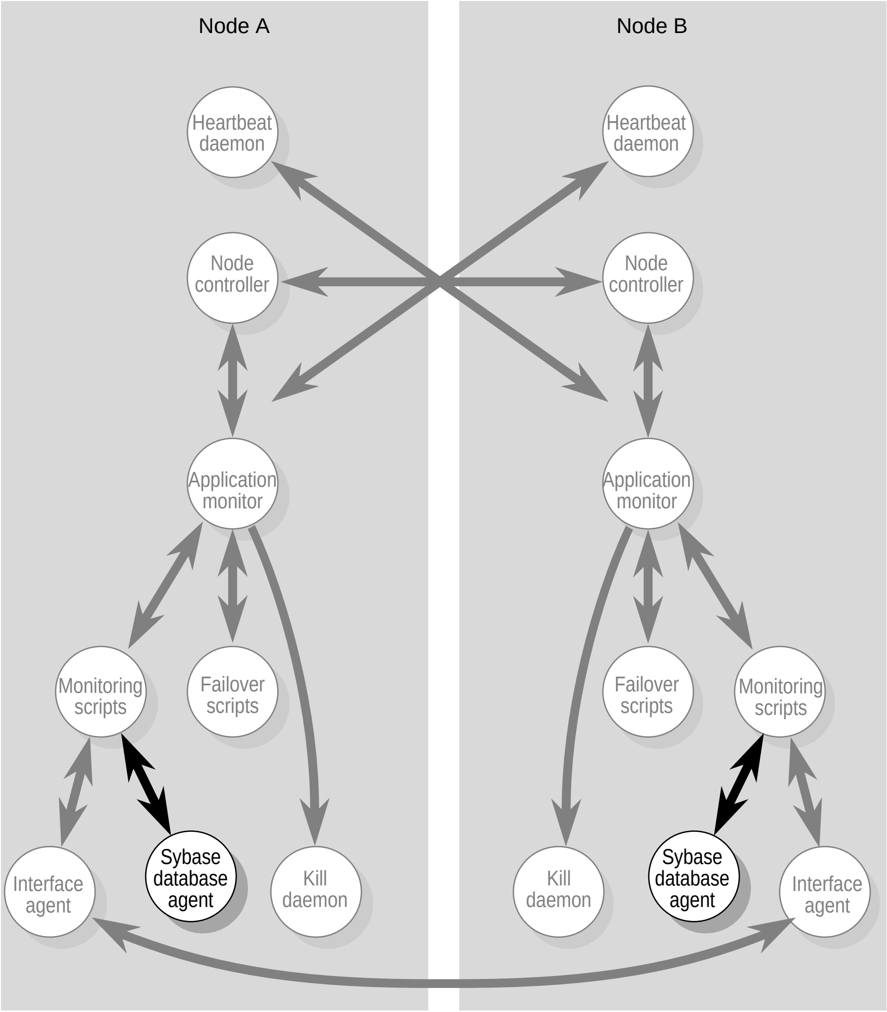

Figure 1-1 Sybase Database Agent Component in IRIS FailSafe

For each SQL Server listed in the */var/ha/ha.conf* configuration file, *ha_sybs* opens a connection to the SQL Server and periodically executes a monitoring stored procedure called *sp_failsafe_mon* to determine if the SQL Server is running (see the next section, "Monitoring Stored Procedure," for more information about the monitoring stored procedure). Depending on the value returned by *sp_failsafe_mon*, *ha_sybs* determines the health of the SQL Server being monitored.

If *ha_sybs* detects a failure, it takes one of two actions, based on the value of the db-avail configuration parameter:

- If db-avail is **low**, *ha_sybs* reports the failure with a message in */var/adm/SYSLOG*.
- If db-avail is **high**, *ha_sybs* exits. The application monitor detects that *ha_sybs* isn't running and reports a local monitoring failure to the node controller. Based on the states of the nodes in the cluster, the node controller determines the actions to be taken, which may include failing over the services on the node to the other node.

Because *ha_sybs* is a multi-threaded processes, you may see several instances of it running on a system simultaneously when you look at the output of the *ps* command.

Monitoring Stored Procedure

When the Sybase database agent *ha_sybs* starts monitoring, it looks for the stored procedure *sp_failsafe_mon*. If it is not found, *ha_sybs* installs a default stored procedure *sp_failsafe_mon* in the sybsystemprocs database. This default stored procedure is very simple and not very intrusive to the Sybase SQL Server. It executes a query on an object in the sybsystemprocs database. If the query succeeds it returns 0. If it fails returns 1. The SQL code of the default *sp_failsafe_mon* is as follows:

```
if exists (select name from sysobjects where
                type = "S" and name = "sysobjects")
        return 0
else
        return 1
```

You can replace this default stored procedure with a different stored procedure. See the section "Customizing the Monitoring Stored Procedure" in Chapter 2 for more information.

Overview of Configuring IRIS FailSafe for Sybase

To configure an IRIS FailSafe cluster for failover of Sybase databases, follow these steps:

1. Install, configure, and test the base IRIS FailSafe software as described in the *IRIS FailSafe Administrator's Guide*.
2. Install additional software if required. See the section "Required Software" in Chapter 2.
3. Choose how to configure Sybase software and databases on the disks. See the section "Planning Sybase Configuration" in Chapter 2.
4. If desired, replace the default monitoring stored procedure with a customized stored procedure. See the section "Customizing the Monitoring Stored Procedure" in Chapter 2.
5. Add Sybase information to the configuration file, */var/ha/ha.conf*. See the section "Adding Sybase Information to the Configuration File" in Chapter 2.
6. Install the new configuration file. See the section "Installing the Configuration File" in Chapter 2.
7. Test Sybase failover. See the section "Testing Sybase Failover" in Chapter 2.

Chapter 2

Configuring IRIS FailSafe for Sybase

This chapter provides information about configuring the IRIS FailSafe Sybase database option for use on IRIS FailSafe systems. The major sections in this chapter are as follows:

- "Required Software" on page 5
- "Planning Sybase Configuration" on page 6
- "Customizing the Monitoring Stored Procedure" on page 9
- "Adding Sybase Information to the Configuration File" on page 9
- "Installing the Configuration File" on page 10
- "Testing Sybase Failover" on page 10

Required Software

The required software Sybase failover is as follows:

- Sybase database software

  See the guide *SYBASE SQL Server™ Installation Guide for Silicon Graphics IRIX* (Sybase document ID 34080-01-1002-01) for information.

  Note: For correct operation of IRIS FailSafe Sybase, the file *ctlib.loc* must exist in the *locales* directory for your language. For example, the location of the English file is *$SYBASE/locales/us_english/iso_1/ctlib.loc*. After installing Sybase and building the SQL Server, ensure that the file for your chosen locale is present; its absense is fatal and difficult to troubleshoot.

- base IRIS FailSafe software

  See the section "Installing the IRIS FailSafe Software" in Chapter 2 of the *IRIS FailSafe Administrator's Guide* for a complete list of required base software.

- IRIS FailSafe Sybase software

  The software subsystem is *ha_sybs.sw.base*.

Planning Sybase Configuration

The subsections below talk about choosing the location of key Sybase files and Sybase databases and the need for customizing certain Sybase files. They also describe a sample configuration of Sybase files and the configuration parameters for this configuration.

Choosing a Sybase Configuration

In configuring Sybase on an IRIS FailSafe cluster, consider location of these components:

- Sybase software ($SYBASE)
- Sybase runserver files in their default location, *$SYBASE/install/RUN_sqlServerName* where the variable *sqlServerName* is the name of an SQL Server
- Sybase databases (including all data files)

All the items above are specified for each Sybase SQL Server. Therefore, you have the flexibility to use different copies of each component, for example Sybase software for each SQL Server (Sybase software for one SQL Server must be identical on both nodes).

To plan an IRIS FailSafe configuration for Sybase, you must answer these questions:

- What is the primary node of each SQL Server?
- Do all SQL Servers use the same Sybase software or not?

After answering all of the questions above, follow these guidelines for each SQL Server to decide whether to use shared or non-shared disks for each of the Sybase components for that SQL Server:

- Sybase software can be on a shared disk or replicated on non-shared disks (it must be identical on each disk).
- Sybase runserver files can be on a shared disk or replicated on non-shared disks (the files must be identical on each disk).
- Sybase databases must be on one or more XFS filesystems or XLV logical volumes that are on shared disks.

  Note: Typically databases are not built on filesystems due to data integrity issues. It is recommended that XLV logical volumes be used as raw devices for your database physical devices.

When configuring highly available SQL Servers, the Sybase runserver files must be located in the *$SYBASE/install* directory. Their filenames must be of the form `RUN_`*sqlServerName*. This is the default name and location used by Sybase—do not customize this pathname.

Example Sybase Configuration

For a simple Sybase configuration with one SQL Server called SYBASE_11 whose primary node is xfs-ha2, an example configuration is as follows:

Sybase software
: On a shared disk with $SYBASE set to */usr/sybase*.

Sybase runserver file
: Stored in *$SYBASE/install/RUN_SYBASE_11* on the shared disk.

Sybase Backup Server
: Called SYB_BACKUP with runserver file *$SYBASE/install/RUN_SYB_BACKUP* on the shared disk.

Sybase database files (master, sybsystemprocs, data)
: On a filesystem on a shared disk and mounted at */shsybs* on xfs-ha2.

Configuration Parameters for Sybase

Table 2-1 lists the label and configuration parameters for the Sybase database in the configuration described in the previous section, "Example Sybase Configuration."

Table 2-1 Sybase Configuration Parameters

Label or Parameter	Example Value	Comments
database label	SYBASE_11	The value is the name of the Sybase SQL Server.
release-dir	/usr/sybase	The value is the value of $SYBASE.

Table 2-1 (continued) Sybase Configuration Parameters

Label or Parameter	Example Value	Comments
master-db-fs	shsybs11	The value is the label for one (any one) of the filesystems that is used by the database. (See the section "Sybase Blocks" in Chapter 3 for more information.) The server-node of the filesystem's volume must be the primary node of the database.
backup-server	SYB_BACKUP	The value is the name of the Backup Server.

Monitoring Sybase Servers

FailSafe/Sybase provides two ways of monitoring Sybase servers. Users specify which monitoring method they prefer in the sybase block of the FailSafe configuration file.

If the user specifies monitoring level 1, FailSafe does Sybase server process monitoring. FailSafe/Sybase checks if the specific Sybase server process is running or not. If this method of monitoring is used in conjunction with Sybase shutdown option 3, users are not required to the specify their Sybase user password (*sa-passwd*) entry in the FailSafe configuration file.

For details about adding information to the FailSafe configuration block, and about the Sybase user password entry and shutdown options, see "Sybase Blocks" on page 16.

To specify monitoring level 1, users have to change the Sybase server startup script, *$SYBASE/install/RUN_sqlServerName*. In the example below, assume this script is */usr/sybase/install/RUN_SYBASE_11*. It contains the *dataserver* command and its options. The **-s** option to the dataserver command must be specified before the **-d** option. Also, the **-s** option argument (Sybase server name) must be in the first 80 characters of the command line. This change is also required if the Sybase shutdown option is set to 3 in the sybase block in the FailSafe configuration file. For example, this command

```
/usr/sybase/bin/dataserver -d/shsybs -sSYBASE_11 \
-e/usr/sybase/install/errorlog_SYBASE_11 -i/usr/sybase
```

must be changed to read:

```
/usr/sybase/bin/dataserver -sSYBASE_11 -d/shsybs \
-e/usr/sybase/install/errorlog_SYBASE_11 -i/usr/sybase
```

Customizing the Monitoring Stored Procedure

If you need more complex monitoring than is provided by the default monitoring stored procedure (see the section "Monitoring Stored Procedure" in Chapter 1) you can create your own monitoring stored procedure. A customized monitoring stored procedure must meet the following requirements:

- It must be called *sp_failsafe_mon*.
- It must be installed in the sybsystemprocs database.
- It must return a value of 0 if it executed successfully and a value of 1 if it failed.
- It must meet a maximum execution time requirement, which is specified by the db-timeout parameter in the sybase block of *ha.conf*.

For more information on Sybase stored procedures, see the *SQL Server Transact-SQL User's Guide* (Sybase document ID 32300-01-1000-03).

Adding Sybase Information to the Configuration File

The procedure for creating a configuration file that includes Sybase configuration information is described below. The procedure assumes that a configuration file that doesn't include Sybase has been created, installed, and tested as described in the *IRIS FailSafe Administrator's Guide*. Using the procedure, you add information to a copy of the configuration file. Installing the configuration file as */var/ha/ha.conf* on both nodes is described in the section "Installing the Configuration File" in this chapter.

1. Copy the file */var/ha/ha.conf* to any directory on one of the nodes in the cluster.
2. Add all of the volumes that will be used for Sybase to the copy of *ha.conf* if they are not already specified in *ha.conf*. Be sure to specify the devname-owner, devname-group, and devname-mode parameters for volumes used as raw devices for Sybase database data. See the sections "Logical Volume Configuration" in Chapter 2 and "Volume Blocks" in Chapter 4 of the *IRIS FailSafe Administrator's Guide* for more information.
3. Add all of the filesystems that will be used for Sybase to the copy of *ha.conf* if they are not already specified in *ha.conf*. See the section "Filesystem Blocks" in Chapter 4 of the *IRIS FailSafe Administrator's Guide* for more information.
4. Append a copy of */var/ha/templates/ha.conf.sybase* to the end of the copy of the configuration file.

5. Complete the block called “application-class sybase.” See the section “Sybase Application-Class Block” in Chapter 3 and the comments in the file for information.
6. Replicate the “sybase” block so that there is one sybase block for each SQL Server.
7. Using the configuration parameter choices you made in the section “Planning Sybase Configuration” in this chapter and the information in the section “Sybase Blocks” in Chapter 3, prepare each of the sybase blocks.
8. Using information in the section “Sybase Blocks” in Chapter 3, prepare the “action sybase” and “action-timer sybase” blocks.
9. Check the configuration file with the *ha_cfgverify* command:

   ```
   # /usr/etc/ha_cfgverify copy_filename
   ```

See the section “Verifying the Configuration File” in Chapter 4 of the *IRIS FailSafe Administrator's Guide* for information about checking configuration with *ha_cfgverify*.

Installing the Configuration File

To install the new version of the configuration file */var/ha/ha.conf* on each node, follow the steps described by “Procedure C” in Chapter 7 of the *IRIS FailSafe Administrator's Guide*.

Testing Sybase Failover

Follow this procedure to test Sybase configuration and failover:

1. Comment out all of the Sybase related blocks in */var/ha/ha.conf* on one node. Do not comment out the blocks for the filesystems and volumes used for Sybase.
2. Copy the copy of *ha.conf* from step 1 to the other node.
3. Test the failover of the volumes and filesystems that are used by Sybase and are on shared disks using the procedures in “Testing Volumes” and “Testing Filesystems” in Chapter 5 of the *IRIS FailSafe Administrator's Guide*.
4. Bring up both nodes in the cluster so that they are running IRIS FailSafe and are in normal state.

5. Bring up each SQL Server on its primary node using these commands (or the equivalent for your shell):

   ```
   # su - unix-user
   # setenv SYBASE release-dir
   # release-dir/install/startserver -frelease-dir/install/RUN_sqlServerName
   ```

   Variables *unix-user* and *release-dir* are the values of the parameters in *ha.conf* with these names for the SQL Server you are starting up; *sqlServerName* is the label for the sybase block in *ha.conf* for this SQL Server.

   (You can identify the primary node of an SQL Server by looking at its master-db-vol or master-db-fs. The volume block for master-db-vol or the volume on which master-db-fs resides lists the server-node for the volume. This is the primary node for the SQL Server.)

6. Bring up each Backup Server (if any) on its primary node using these commands (or the equivalent for your shell):

   ```
   # release-dir/install/startserver -frelease-dir/install/RUN_backup-server
   ```

   backup-server is the values of the parameter in *ha.conf* with this name for the Backup Server you are starting up.

7. Check that you can successfully access each of the SQL Servers by entering the *isql* command for each SQL Server:

   ```
   # release-dir/bin/isql -U sa-user -P sa-passwd -S sqlServerName
   ```

   sa-user and *sa-passwd* are the values of the parameters in *ha.conf* with these names for the SQL Server you are starting up. *sqlServerName* is the name of the SQL Server you are starting up and is also the label for the sybase block in *ha.conf* for this SQL Server. If you get the *isql* prompt, the SQL Server is accessible.

8. If you created a custom stored procedure for monitoring (*sp_failsafe_mon*), test it from *isql*. At the *isql* prompt enter:

   ```
   > use sybsystemprocs
   > go
   > sp_failsafe_mon
   > go
   (return status = 0)
   ```

   If the return status is not 0, check whether all the conditions you want to monitor are met. You may need to debug the custom stored procedure and make sure that it returns the appropriate values.

9. Exit *isql* by entering:

   ```
   > exit
   ```

10. Shut down each SQL Server. See the *SYBASE SQL Server System Administration Guide* (Sybase document ID 32500-01-1000-03) for information on shutting down SQL Servers.

11. Enter this command on one node to shut down IRIS FailSafe on that node:

    ```
    # /etc/init.d/failsafe stop
    ```

    Wait for this command to complete before proceeding.

12. Enter this command on the other node to shut down IRIS FailSafe on that node:

    ```
    # /etc/init.d/failsafe stop
    ```

13. Uncomment all of the Sybase blocks in */var/ha/ha.conf* on one node.

14. Copy the copy of *ha.conf* from step 13 to the other node.

15. Enter this command on each node to start up IRIS FailSafe:

    ```
    # /etc/init.d/failsafe start
    ```

16. Verify that each node is in normal state by giving this command on each node:

    ```
    # /usr/etc/ha_admin -i
    ha_admin: Node controller state normal
    ```

    If either node is not in normal state, wait 30 seconds and try the command again.

17. Check that each of the SQL Server started up by accessing each of the SQL Servers using the *isql* command you used in step 7.

18. For the first SQL Server listed in *ha.conf* (the first sybase block) that has the db-avail parameter set to **high**, shut down this SQL Server.

19. Check that IRIS FailSafe fails over all SQL Servers and all other highly available services on the node on which you shut down the SQL Server to the other node. For example, to check for Sybase processes, enter this command:

    ```
    # ps -ef | egrep sybase
     sybase 27963     1  0   Jul 12 ?        0:00 /bin/sh
    /usr/sybase/install/RUN_SYB_BACKUP
     sybase 27965 27963  0   Jul 12 ?        0:00
    /usr/sybase/bin/backupserver -SSYB_BACKUP
    -e/usr/subase/install/backuplog
     sybase 27955 27942  0   Jul 12 ?        0:17
    /usr/sybase/bin/dataserver -d/usr/sybase/master.dat -sSYBASE_11
    -e/usr/subase/install/errorlog
    ```

```
 sybase 27957 27942  0   Jul 12 ?        0:15
/usr/sybase/bin/dataserver -d/usr/sybase/master.dat -sSYBASE_11
-e/usr/subase/install/errorlog
 sybase 27959 27942  0   Jul 12 ?        0:19
/usr/sybase/bin/dataserver -d/usr/sybase/master.dat -sSYBASE_11
-e/usr/subase/install/errorlog
 sybase 27942     1  0   Jul 12 ?        0:00 /bin/sh
/usr/sybase/install/RUN_SYBASE_11
```

The example shows the processes for the SQL Server SYBASE_11 and the Backup Server SYB_BACKUP.

Note: The parent "dataserver" process spawns children to perform asynchronous I/O for the RDBMS instance. These children are also named "dataserver." Please contact Sybase Technical Support to tune the number of asynchronous I/O threads.

20. Confirm that the node on which you shut down the SQL Server is in standby state:

```
# /usr/etc/ha_admin -i
ha_admin: Node controller state standby
```

21. On the node that is now in standby state, enter this command to return it to normal state:

```
# /usr/etc/ha_admin -rf
```

22. Check that the highly available services that failed over in step 19 are stopped on the backup node and restarted on their original node.

23. For each of the remaining SQL Servers listed in *ha.conf* whose db-avail parameter is set to **high**, repeat steps 18 through 22 to verify that they fail over properly when shut down.

Chapter 3

Configuration File Blocks for Sybase

Sybase configuration parameters must be specified in the */var/ha/ha.conf* configuration file. Sections in this chapter describe each Sybase specific block that must be added and the configuration parameters within each of those blocks. The sections are as follows:

- "Sybase Application-Class Block" on page 15
- "Sybase Blocks" on page 16
- "Sybase Action and Action-Timer Blocks" on page 20

The examples in this chapter show the Sybase configuration file blocks for the example discussed in the sections "Example Sybase Configuration" and "Configuration Parameters for Sybase" in Chapter 2.

Sybase Application-Class Block

Example 3-1 shows the application-class block in a Sybase configuration.

Example 3-1 Sybase Application-Class Block

```
application-class sybase
{
        server-node = xfs-ha1
        agent = /usr/etc/ha_sybs
}
```

The application-class sybase block contains these configuration parameters:

server-node Lists a node that is a primary server for an SQL Server. In this example, just one node is listed because just one node is a primary node for an SQL Server. If both nodes are primary nodes for SQL Servers, server-node is listed twice.

agent The full pathname of the database agent for Sybase. You should not change this parameter.

Sybase Blocks

IRIS FailSafe configuration files contain one sybase block for each SQL Server that is a highly available service.

Example 3-2 shows an example sybase block.

Example 3-2 Sybase Block

```
sybase SYBASE_11
{
    master-db-fs = shsybs11
    backup-server-network-name = SYB_BACKUP
    backup-server-name = SYB_BACKUP
    unix-user = sybase
    sa-user = sa
    sa-passwd = sa_p
    release-dir = /usr/sybase
    shutdown-options = 2
    monitoring-level = 2
    db-avail = high
    db-probe-time = 120
    db-timeout = 120
    start-monitor-time = 300
    db-retry-count = 1
}
```

The label for the sybase block, SYBASE_11 in this example, is the name of the SQL Server. The other parts of this guide refer to this as the *sqlServerName*. Configuration parameters used in sybase blocks are as follows:

master-db-vol

The value must match the label of a volume block. See the discussion at the end of this section about master-db-vol and master-db-fs for information about which volume block label to use. This parameter or master-db-fs must be specified.

master-db-fs

The value must match the label of a filesystem block. See the discussion at the end of this section about master-db-vol and master-db-fs for information about which filesystem block label to use. This parameter or master-db-vol must be specified.

unix-user

The IRIX login name that is the owner of the Sybase release directory.

backup-server-network-name

If the SQL server has a sybase backup server, this field must be specified. The backup server network name is the network name field in the SQL server's "sysservers" system table. It is same as the name of the server in the *$SYBASE/install/RUN_sqlServerName* backup server startup script. This field is used to start the backup server. This field is optional.

backup-server-name

The backup server's logical name as specified in the SQL server's "sysservers" system table. This value can be obtained by executing the stored "sp_helpserver" procedure. This field is used to shutdown the backup server, and is optional.

release-dir

The Sybase release directory specified in the Sybase configuration. This value is also the value of the environment variable SYBASE. This parameter is required.

sa-user

The Sybase login name of the Sybase database system administrator. This parameter is required.

sa-passwd

The unencrypted password of the Sybase database system administrator (sa-user). This parameter should be omitted if there is no password.

shutdown-options

Defines how the SQL Server is shut down on a failover. It is required. The possible values are these:

0—Normal shutdown

1—Shutdown with nowait and no checkpointing

2—Shutdown with nowait with checkpointing

3—Kill all the dataserver processes

monitoring-level

This field determined the type of monitoring that FailSafe/Sybase uses to monitor the servers. This field is required. The possible values are:

1—FailSafe checks if the server processes are running

2—The FailSafe/Sybase agent executes the stored procedure "sp_failsafe_mon" to determine if the sybase server is running

db-avail
: The possible values are **high** and **low**. If the value is **high**, an SQL Server failure forces a failover. If the value is **low**, a failure of the SQL Server does not force a failover, but the failure is reported by sending mail to the address indicated by the "mail-dest-addr" parameter. The default value is **high**.

start-monitor-time
: The amount of time that the Sybase database agent waits after receiving a message to start monitoring before it performs its first probe of the SQL Server. This wait ensures that the SQL Server has had time to start up.

db-probe-time
: Defines the length of time (in seconds) between the completion of one probe of the SQL Server by the Sybase database agent and the beginning of the next probe. This parameter is required.

db-timeout
: Defines the time the Sybase database agent waits for a response to its probe from the SQL Server in seconds. This parameter is required.

db-retry-count
: The number of monitoring retries by the database agent before a failure is declared.

The Sybase database files could reside in multiple filesystems and on multiple logical volumes. You must choose one of these filesystems or volumes to be the "master" and specify it as the value of master-db-fs or master-db-vol. Typically it should be the volume or filesystem where *master.dat* resides. The sybase blocks do not list all filesystems and volumes that are part of the SQL Server. However, the configuration file must list all volumes and filesystems used by each SQL Server in volume and filesystem blocks (see the sections "Volume Blocks" and "Filesystem Blocks" in Chapter 2 of the *IRIS FailSafe Administrator's Guide* for information on these blocks). The IRIS FailSafe system uses the volume and filesystem blocks as a specification of the shared storage that must be online prior to bringing up the SQL Server.

Figure 3-1 shows the monitoring frequency parameters in each sybase block. They control the monitoring of an SQL Server by the Sybase database agent.

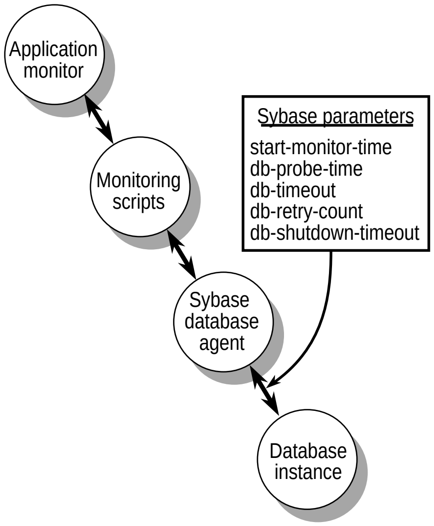

Figure 3-1 Monitoring Frequency Configuration Parameters in Sybase Blocks

Table 3-1 lists each of these monitoring frequency parameters and their suggested values. It also describes relationships between different parameters and the rounding that occurs for some parameter values.

Table 3-1 Values for Monitoring Frequency Configuration Parameter in Sybase Blocks

Parameter	Suggested Value	Comments
start-monitor-time (sybase block)	300	Rounded to the nearest ten seconds. The minimum value is the value of long-timeout.
db-probe-time	120	Rounded to the nearest ten seconds.
db-timeout	120	Rounded to the nearest ten seconds.
db-retry-count	1	The minimum value is 1.

Sybase Action and Action-Timer Blocks

Example 3-3 shows the action and action-timer blocks for Sybase. The action block specifies the pathnames of the local monitoring script and the action-timer block specifies monitoring timing and timeout values for the monitoring of the Sybase database agent by the application monitor.

Example 3-3 Sybase Action and Action-Timer Blocks

```
action sybase
{
        local-monitor = /var/ha/actions/ha_sybs_lmon
}

action-timer sybase
{
        start-monitor-time = 300
        lmon-probe-time = 60
        lmon-timeout = 30
        retry-count = 2
}
```

The parameters used in action and action-timer blocks for Sybase are as follows:

local-monitor
The pathname of the local monitoring script for Sybase. Do not change this value.

start-monitor-time
Specifies the amount of time that the application monitor waits before it starts using the local monitoring script to monitor the Sybase database agent. The value of this parameter should be greater than or equal to the largest value specified for start-monitor parameters in any of the sybase blocks and it should be greater than or equal to the value of long-timeout.

lmon-probe-time
Local monitoring of the Sybase agent is done this often (in seconds).

lmon-timeout
Local monitoring of the Sybase agent times out in this many seconds if no response is received.

retry-count
Specifies the number of probes to the Sybase database agent done by the monitoring script. This value doesn't affect the lmon-timeout value.

Figure 3-2 shows the monitoring parameters in the "action-timer sybase" block. They control the monitoring of the Sybase database agent by the local monitoring script.

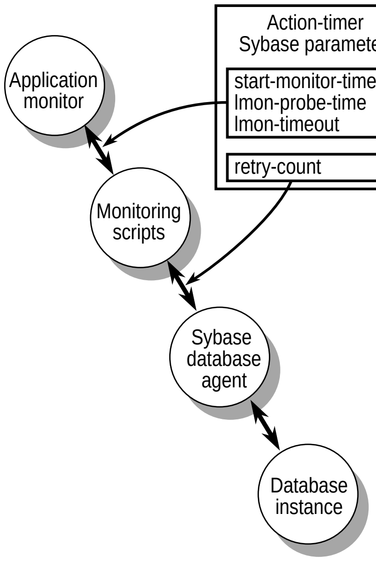

Figure 3-2 Monitoring Configuration Parameters in the Action-Timer Block for Sybase

Table 3-2 lists each of these monitoring parameters and their suggested values. It also describes relationships between different parameters and the rounding that occurs for some parameter values.

Table 3-2 Values for Monitoring Frequency Configuration Parameter in Sybase Action-Timer Blocks

Parameter	Suggested Value	Comments
start-monitor-time (action-timer block)	300	The minimum value is the value of long-timeout. Should be greater than or equal to the largest of the values specified for start-monitor-time in any of the sybase blocks.
lmon-probe-time	60	
lmon-timeout	30	
retry-count	2	This value does not affect the lmon-timeout value. The minimum value is 1.

The length of time it takes the database agent to access SQL Servers is variable, so you have to be very careful in choosing the monitoring time values so that you do not get false failovers. The values suggested as defaults are just a start point if using the default stored procedure.

While tuning the monitoring time parameters, it is a good idea to define db-avail as **low**, so that if there is a monitoring failure IRIS FailSafe does not failover. A message is logged in the */var/adm/SYSLOG* file, so you know that a failover would have occurred if db-avail had been set to **high**.

Appendix A

Messages About Configuration File Errors

This appendix lists the error messages and warnings that can appear as output of the *ha_cfgverify* command that are related to Sybase configuration. Other messages are described in Appendix A of the *IRIS FailSafe Administrator's Guide*.

Note: *ha_cfgverify* generates warnings when a possible error exists, but it is unable to determine for sure. You must check each warning manually. A warning could indicate invalid IRIS FailSafe configuration.

```
ha_cfgverify: No sybase section present in the file
```

There is a Sybase application-class entry but there is no sybase block.

```
ha_cfgverify: sybase <SQL_Server> section: sa-passwd entry is missing.
Assuming no sa-passwd
```

This is not an error. Because the optional sa-passwd entry is missing, IRIS FailSafe assumes that there is no password for the Sybase database system administrator.

```
ha_cfgverify: sybase <SQL_Server> section must have db-avail entry
ha_cfgverify: sybase <SQL_Server> section must have db-probe-time entry
ha_cfgverify: sybase <SQL_Server> section must have db-retry-count entry
ha_cfgverify: sybase <SQL_Server> section must have db-timeout entry
ha_cfgverify: sybase <SQL_Server> section must have either master-db-vol
 or master-db-fs entry
ha_cfgverify: sybase <SQL_Server> section must have release-dir entry
ha_cfgverify: sybase <SQL_Server> section must have sa-user entry
ha_cfgverify: sybase <SQL_Server> section must have shutdown-options
 entry
ha_cfgverify: sybase <SQL_Server> section must have start-monitor-time
 entry
ha_cfgverify: sybase <SQL_Server> section must have unix-user entry
```

A mandatory parameter is missing from the sybase block.

```
ha_cfgverify: sybase <SQL_Server> section: db-avail entry <value> must be
either “high” or “low”
```

You must set the db-avail parameter to **high** (an SQL Server failure causes a failover) or **low** (an SQL Server failure doesn’t cause a failover).

```
ha_cfgverify: sybase <SQL_Server> section: shutdown-options <value> must
be either 0, 1, 2, or 3
```

The values for the shutdown-options parameters are **0** (normal shutdown), **1** (shutdown with nowait and no checkpointing), **2** (shutdown with nowait with checkpointing), and **3** (kill the server processes).

```
ha_cfgverify: sybase <SQL_Server> section: unix-user entry <user name> is
invalid
```

The system failed to get a password file entry for the user specified as unix-user.

```
ha_cfgverify: sybase <SQL_Server>: the master-db-fs entry <filesystem> is
invalid
```

The master-db-fs entry must match the label of a filesystem block.

```
ha_cfgverify: sybase <SQL_Server>: the master-db-vol entry <volume> is
invalid
```

The master-db-vol entry must match the label of a volume block.

```
ha_cfgverify: Warning: sybase <SQL_Server> section: release-dir entry
directory> is invalid
```

The specified directory is not accessible or is not a directory or the underlying filesystem may not be mounted.

```
ha_cfgverify: sybase <SQL_Server>: monitoring-level must be either 1 or 2
```

The valid values for the monitoring level field are 1 and 2.

Tell Us About This Manual

As a user of Silicon Graphics products, you can help us to better understand your needs and to improve the quality of our documentation.

Any information that you provide will be useful. Here is a list of suggested topics:

- General impression of the document
- Omission of material that you expected to find
- Technical errors
- Relevance of the material to the job you had to do
- Quality of the printing and binding

Please send the title and part number of the document with your comments. The part number for this document is 007-3306-002.

Thank you!

Three Ways to Reach Us

- To send your comments by **electronic mail**, use either of these addresses:
  - On the Internet: techpubs@sgi.com
  - For UUCP mail (through any backbone site): *[your_site]*!sgi!techpubs
- To **fax** your comments (or annotated copies of manual pages), use this fax number: 650-965-0964
- To send your comments by **traditional mail**, use this address:

  Technical Publications
  Silicon Graphics, Inc.
  2011 North Shoreline Boulevard, M/S 535
  Mountain View, California 94043-1389