

IRIS FailSafe™ 2.0 Oracle®
Administrator's Guide

Document Number 007-3933-001

CONTRIBUTORS

Written by Susan Ellis, revised by Anita Manders
Illustrated by Dany Galgani
Edited by Rick Thompson
Production by Heather Hermstad and Amy Swenson
Engineering contributions by Gilberto Arnaiz, Rishi Avtar, and Bill Sparks
Cover design and illustration by Rob Aguilar, Rikk Carey, Dean Hodgkinson,
Erik Lindholm, and Kay Maitz

© 1996, 1999 Silicon Graphics, Inc.— All Rights Reserved

This document or parts thereof may not be reproduced in any form unless permitted by contract or by written permission of Silicon Graphics, Inc.

LIMITED AND RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in the Rights in Data clause at FAR 52.227-14 and/or in similar or successor clauses in the FAR, or in the DOD, DOE or NASA FAR Supplements. Unpublished rights reserved under the Copyright Laws of the United States.

Contractor/manufacture is SGI, 1600 Amphitheatre Pkwy., Mountain View, CA 94043-1351.

Silicon Graphics and IRIS are registered trademarks, and IRIS FailSafe, Origin200, Origin2000, the Silicon Graphics logo, SGI, and XFS are trademarks of Silicon Graphics, Inc.

INFORMIX is a trademark of Informix Software, Inc. Netscape is a trademark of Netscape Communications Corporation. NFS is a trademark of Sun Microsystems, Inc. Oracle is a trademark of Oracle Corporation.

Contents

	List of Figures	v
	List of Tables	vii
	List of Examples	ix
	About This Guide	xi
	Audience	xi
	Structure of This Document	xi
	Related Documentation	xii
	Conventions Used in This Guide	xiii
1.	IRIS FailSafe Oracle Overview	1
	IRIS FailSafe Oracle Database Monitoring	1
	Database Startup and Shutdown Scripts	3
	Overview of Configuring IRIS FailSafe for Oracle	3
2.	Configuring IRIS FailSafe for Oracle	5
	Required Software	5
	Planning Oracle Configuration	6
	Choosing an Oracle Configuration	6
	Example Oracle Resource Configuration	8
	Configuration Parameters for Oracle	9
	Customizing the Oracle Startup and Shutdown Scripts	10
	Adding Oracle Resource Information to the Configuration Database (CDB)	12
	Creating an Oracle Resource Type	12
	Creating the Oracle_DB Resource Type	13
	Creating an Oracle Resource	14
	Creating an Oracle Resource Group	16
	Installing Startup and Shutdown Scripts	16

Testing the Oracle Resource	17
Testing the start Script	19
Testing the stop Script	19
Testing the monitor Script	20
Testing the restart Script	21
Testing the exclusive and probe Scripts	21
Testing Resource Group Failovers	22

List of Figures

Figure 1-1 Message Paths in IRIS FailSafe 2

List of Tables

Table i	IRIS FailSafe Release Notes	xiii
Table 2-1	Oracle Resource Configuration Parameters	9

List of Examples

Example 2-1	Sample Oracle Startup Script	10
Example 2-2	Sample Oracle Shutdown Script	11
Example 2-3	Oracle Installation Process	12
Example 2-4	Creating the Oracle_DB Resource	14

About This Guide

This guide provides information about configuring IRIS FailSafe 2.0 systems with the IRIS FailSafe 2.0 Oracle database option. This option enables Oracle 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 2.0 Administrator's Guide*.

This guide was prepared in conjunction with the initial release of the IRIS FailSafe 2.0 Oracle option and the 1.1 Release of the IRIS FailSafe software product. It describes IRIS FailSafe Oracle software for Oracle 7.3 and Oracle 8.0.3.

Audience

This guide is written for system administrators who are responsible for configuring and administering an IRIS FailSafe system with the optional IRIS FailSafe Oracle software. These system administrators must be able to customize several shell scripts and must be familiar with Oracle configuration and Oracle startup and shutdown procedures.

Structure of This Document

This guide contains the following chapters:

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

Related Documentation

For Oracle installation information, see the *Oracle817 for MIPS ABI Systems Installation and Configuration Guide*.

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

- *IRIS FailSafe 2.0 Administrator's Guide*
- *IRIS FailSafe 2.0 Programmer's Guide*
- *IRIS FailSafe 2.0 INFORMIX Administrator's Guide*
- *Iris Failsafe 2.0 Netscape Server Administrator's Guide*
- *Iris Failsafe 2.0 NFS Administrator's Guide*

The IRIS FailSafe reference pages are as follows:

- failsafe(7M)
- ha_cilog(1M)
- ha_cmsd(1M)
- ha_exec2(1M)
- ha_filelock(1M)
- ha_fileunlock(1M)
- ha_fsadm(1M)
- ha_fsd(1M)
- ha_gcd(1M)
- ha_http_ping2(1M) (IRIS FailSafe Web option)
- ha_ifdadmin(1M)
- ha_ifd(1M)
- ha_ifmx2(1M) (IRIS FailSafe INFORMIX option)
- ha_macconfig2(1M)
- ha_mom(1M)
- ha_srmd(1M)
- ha_statd2(1M)

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

Table i IRIS FailSafe Release Notes

Release Note	Product
failsafe2	IRIS FailSafe 2.0
failsafe2_nfs	IRIS FailSafe 2.0 NFS
failsafe2_web	IRIS FailSafe 2.0 Netscape
failsafe2_informix	IRIS FailSafe 2.0 INFORMIX
failsafe2_oracle	IRIS FailSafe 2.0 Oracle

Conventions Used in This Guide

These type conventions and symbols are used in this guide:

Italics New terms, manual/book titles, variable command line arguments, and variables to be supplied by the user in examples, code, and syntax statements

Fixed-width type Command names and command line arguments, file names, and examples of command output and code

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 super user (`root`)

IRIS FailSafe Oracle Overview

This chapter provides information about the components that are added to the base IRIS FailSafe 2.0 product by the IRIS FailSafe 2.0 Oracle option. It assumes that you are familiar with the basic components of IRIS FailSafe described in Chapter 1 of the *IRIS FailSafe 2.0 Administrator's Guide*. This chapter also lists the steps you must take to add Oracle databases to the highly available services that are failed over on an IRIS FailSafe cluster.

The major sections in this chapter are as follows:

- “IRIS FailSafe Oracle Database Monitoring” on page 1
- “Database Startup and Shutdown Scripts” on page 3
- “Overview of Configuring IRIS FailSafe for Oracle” on page 3

IRIS FailSafe Oracle Database Monitoring

The IRIS FailSafe Oracle option adds database support to the IRIS FailSafe base software. The database monitoring scripts monitor the Oracle database. If the monitor script detects a failure, it will report a local monitoring failure to FailSafe, and FailSafe will take the appropriate action. Figure 1-1 shows an example of the message paths in IRIS FailSafe.

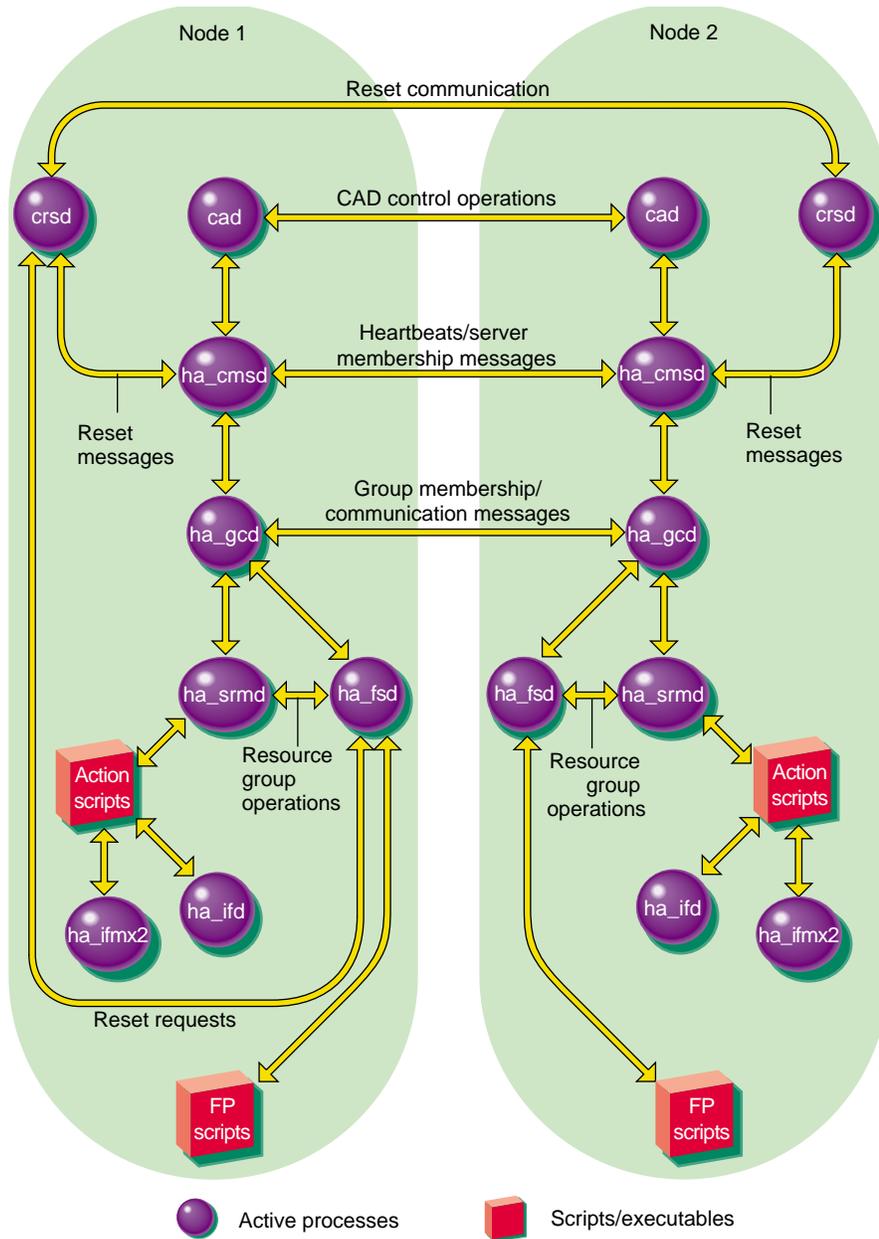


Figure 1-1 Message Paths in IRIS FailSafe

Database Startup and Shutdown Scripts

The standard Oracle startup and shutdown scripts, `$ORACLE_HOME/bin/dbstart` and `$ORACLE_HOME/bin/dbstop`, cannot be used with IRIS FailSafe. You must use scripts that you develop by starting with the following two sample scripts:

```
/var/cluster/ha/resource_types/Oracle_DB/startup_oracle.sample  
/var/cluster/ha/resource_types/Oracle_DB/shutdown_oracle.sample
```

A key difference between the standard scripts and IRIS FailSafe scripts is that the standard scripts start up and shut down all Oracle databases. IRIS FailSafe scripts start up or shut down just one Oracle instance (one Oracle server accessing one Oracle database), the one identified by `$ORACLE_SID`.

For each type of script (startup and shutdown), you can use any combination of the following strategies:

- Write one script that takes the database instance as an argument (which IRIS FailSafe passes as an environment variable). Use it to start up (or shut down) each database instance.
- Use different startup (or shutdown) scripts for different database instances.
- Use scripts with the same pathnames on each node, but different contents. For example, the startup script for an Origin2000 that is a primary node might be different from the startup script on a backup node that is an Origin200.

Overview of Configuring IRIS FailSafe for Oracle

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

1. Install, configure, and test the base IRIS FailSafe software as described in the *IRIS FailSafe 2.0 Administrator's Guide*.
2. Install additional software if required. See the section "Required Software" in Chapter 2.
3. Choose how to configure Oracle software and databases on the disks. See the section "Planning Oracle Configuration" in Chapter 2.
4. Prepare the Oracle startup and shutdown scripts. See the section "Customizing the Oracle Startup and Shutdown Scripts" in Chapter 2.

5. Add the Oracle resource type, resource attribute, and resource information to the configuration database (CDB) file. See the section “Adding Oracle Resource Information to the Configuration Database (CDB)” in Chapter 2.
6. Install the Oracle startup and shutdown scripts on each node. See the section “Installing Startup and Shutdown Scripts” in Chapter 2.
7. Test Oracle failover. See the section “Testing the Oracle Resource” in Chapter 2.

Configuring IRIS FailSafe for Oracle

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

- “Required Software” on page 5
- “Planning Oracle Configuration” on page 6
- “Customizing the Oracle Startup and Shutdown Scripts” on page 10
- “Adding Oracle Resource Information to the Configuration Database (CDB)” on page 12
- “Installing Startup and Shutdown Scripts” on page 16
- “Testing the Oracle Resource” on page 17

Required Software

The required software for Oracle failover is as follows:

- Oracle database software
See the guide *Oracle817 for MIPS ABI Systems Installation and Configuration Guide*.
- Base IRIS FailSafe software
See the section “Installing the IRIS FailSafe Software” in Chapter 2 of the *IRIS FailSafe 2.0 Administrator’s Guide* for a complete list of required base software.

- IRIS FailSafe Oracle software, as follows:
 - The *IRIS FailSafe 2.0 Oracle Administrator's Guide* subsystem is `failsafe2_oracle.books.book_AG`.
 - The IRIS FailSafe 2.0 Oracle release notes subsystem is `failsafe2_oracle.man.relnotes`.
 - The base software subsystem is `failsafe2_oracle.sw.base`.

Planning Oracle Configuration

The following subsections discuss choosing the location of key Oracle files and Oracle databases and the need for customizing certain Oracle files. They also describe an example configuration of Oracle files and the configuration parameters for this configuration.

Choosing an Oracle Configuration

In configuring Oracle on an IRIS FailSafe cluster, you need to consider the locations of these components:

- Oracle software (`$ORACLE_HOME`)
- The Oracle parameter file in its default location (`$ORACLE_HOME/dbs/init${ORACLE_SID}.ora`) or a location you specify
- IRIS FailSafe versions of the Oracle startup and shutdown scripts
- Oracle databases (including all data, log, and control files)

All of the preceding items are specified for each Oracle instance. Therefore, you have the flexibility to use different copies of each component, for example Oracle software, for each database instance (the Oracle software used by one instance must be identical on both nodes). In addition, the Oracle parameter file, startup script, and shutdown script for a database instance need not be identical on each node. For example, if the two nodes have different amounts of memory, you may want different Oracle parameter files on each node, so that the size of the System Global Area is different on each node. The two parameter files must have identical pathnames, but the contents can be different.

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

- Do all database instances use the same Oracle software?
- Is the Oracle parameter file identical on each node for each database instance?
- Are the Oracle startup and shutdown scripts identical in location and content for each database instance?
- For each database instance, is the content of its Oracle startup script identical on each node?
- For each database instance, is the content of its Oracle shutdown script identical on each node?

After answering all of the preceding questions, follow these guidelines for each database instance to decide whether to use shared or non-shared disks for each of the Oracle components for that instance:

- Oracle databases must be on one or more XFS file systems or XLV logical volumes that are on shared disks.
- If the Oracle parameter file is identical on each node, it can be on a shared disk or replicated on non-shared disks.
- If the Oracle parameter file is not identical on each node, it must be on non-shared disks on each node.
- If the Oracle startup and shutdown scripts are identical on each node, they can be on a shared disk or replicated on non-shared disks.
- If the Oracle startup and shutdown scripts are not identical on each node, they must be on a non-shared disk on each node.
- Oracle software can be on a shared disk or replicated on non-shared disks (it must be identical on each disk).
- If the Oracle parameter file is not identical on each node, it must be on non-shared disks and linked to the directory in the Oracle software where Oracle expects to find it.

Once you have decided whether to store the Oracle startup and shutdown scripts on shared or non-shared disks, you can choose the pathnames of the files. Some possibilities are as follows:

- The scripts can be installed in a directory in the Oracle software, for example, `/usr/oracle/scripts/dbstart`. This is a good choice when using Oracle software installed on a shared disk.
- The scripts can be installed in the directory `/var/cluster/ha`, for example, `/var/cluster/ha/resource_types/Oracle_DB`.

Example Oracle Resource Configuration

An example configuration for a simple Oracle configuration with one database instance called `Oracle1` is as follows:

Oracle software

On a shared disk with `$ORACLE_HOME` set to `/usr/people/oracle`

Oracle parameter file

Identical on both nodes and stored in `$ORACLE_HOME/dbs/initfs.ora`, which is on the same shared disk as the Oracle software

Oracle startup script

Identical on both nodes and stored in `$ORACLE_HOME/scripts/dbstart`, which is on the same shared disk as the Oracle software

Oracle shutdown script

Identical on both nodes and stored in `$ORACLE_HOME/scripts/dbstop`, which is on the same shared disk as the Oracle software

Configuration Parameters for Oracle

Table 2-1 lists the label and configuration parameters for the Oracle database resource in the configuration described in the previous section, “Example Oracle Resource Configuration.”

Table 2-1 Oracle Resource Configuration Parameters

Label or Parameter	Example Value	Comments
<i>resource-name</i>	oracle	This value is the value of \$ORACLE_SID. It is used to create the pathname of the Oracle parameter file unless a different parameter file is specified in the Oracle startup script.
release-dir	/usr/people/oracle	This value is the value of \$ORACLE_HOME.
startup-script	/usr/people/oracle /scripts/dbstart	The value is the location of the Oracle startup script for this instance (copy from /usr/cluster/ha/resource_types/Oracle_DB).
shutdown-script	/usr/people/oracle /scripts/dbstop	The value is the location of the Oracle shutdown script for this instance (copy from /usr/cluster/ha/resource_types/Oracle_DB).
master-db-fs	ofs2fs	The value is the label for one (any one) of the file systems that is used by the database. The server-node of the file system’s volume must be the primary node of the database.
oracle-user	oracle	The UNIX user of Oracle.

The Oracle_DB resource also contains a resource dependency on the `filesystem` and `IP_address` resources. These resources must be specified when defining the resource group (see Chapter 2, “Creating an Oracle Resource Group”).

Customizing the Oracle Startup and Shutdown Scripts

Example 2-1 is a copy of the sample Oracle startup script provided in `/var/cluster/ha/resource_types/Oracle_DB/startup_oracle.sample`. Most comments have been removed and line numbers have been added.

Example 2-1 Sample Oracle Startup Script

```
1  #! /sbin/ksh
2
3  PATH=$ORACLE_HOME/bin:$PATH
4  # By default ${ORACLE_HOME}/dbs/init${ORACLE_SID}.ora will be
5  # used as the parameter file
6  svrmgrl <<EOF
7  connect internal
8  startup
9  EOF
10 if test $? -eq 0 ; then
11     logger "Database \"${ORACLE_SID}\" started"
12     exit 0;
13 else
14     logger "Database \"${ORACLE_SID}\" not started"
15     exit 1;
16 fi
```

As written, the script assumes that `$ORACLE_HOME` is set to the value of the parameter `release-dir` and `$ORACLE_SID` is set to the value of `resource-name`. These environment variables are set by the script `/var/cluster/ha/resource_types/Oracle_DB /`, which calls this startup script.

As mentioned in the comments on lines 4 and 5, the file `${ORACLE_HOME}/dbs/init${ORACLE_SID}.ora` is used as the Oracle parameter file by default. To use a different file, change line 8 to the following:

```
startup pfile=parameterfile
```

where *parameterfile* is the full pathname of the parameter file.

Lines 6 through 9 start an Oracle instance.

Line 10 checks the exit status of the `svrmgrl` command to determine whether the database has been successfully started. The script exits with a status of 0 for a successful start or 1 for an unsuccessful start.

Example 2-2 is a copy of the sample Oracle shutdown script provided in `/var/cluster/ha/resource_types/Oracle_DB/shutdown_oracle.sample`. Again, comments have been removed and line numbers have been added.

Example 2-2 Sample Oracle Shutdown Script

```
1  #! /sbin/ksh
2
3  PATH=$ORACLE_HOME/bin:$PATH
4  svrmgrl <<EOF
5  connect internal
6  shutdown normal
7  EOF
8  if test $? -eq 0 ; then
9      logger "Database \"${ORACLE_SID}\" shut down."
10     exit 0;
11 else
12     logger "Database \"${ORACLE_SID}\" not shut down."
13     exit 1;
14 fi
```

Lines 4 through 7 shut down an Oracle instance.

Line 8 checks the exit status of the `svrmgrl` command to determine whether the database has been successfully shut down. The script exits with a status of 0 for a successful shutdown or 1 for an unsuccessful shutdown.

When customizing the Oracle startup and shutdown scripts, remember these points:

- The script cannot be interactive. It must be able to run as a background process.
- The script will be called once for each Oracle block in the configuration file that gives the pathname of this script as the value of `startup-script` or `shutdown-script`.
- The script is automatically run with the environment variable `ORACLE_HOME` set to the value of the configuration parameter `release-dir` and the environment variable `ORACLE_SID` set to the value of the configuration parameter `resource-name`.
- If you have changed the location of the Oracle parameter file, be sure that the startup script has the correct pathname.

Adding Oracle Resource Information to the Configuration Database (CDB)

This section describes procedures that show you how to create an Oracle resource type, resource, resource group, and configuration database that includes Oracle configuration information. These procedures assume that a CDB that does not include Oracle has already been created, installed, and tested as described in the *IRIS FailSafe 2.0 Administrator's Guide*.

Creating an Oracle Resource Type

To create an `Oracle_DB` resource type, this subsection assumes that you are already familiar with the concepts of resource types. The `Oracle_DB` resource type defines the following resource attributes. See Table 2-1 for the definitions of these resource attributes.

- `resource-name`
- `release-dir`
- `startup-script`
- `shutdown-script`
- `master-db-fs`
- `oracle-user`

The `Oracle_DB` resource type is created at cluster creation time. If this automatic resource creation fails, the administrator must create the resource type before an Oracle resource is created. The `Oracle_DB` resource type must be installed if you want to add an NFS resource to a cluster that was created before the NFS software was installed.

Example 2-3 Oracle Installation Process

```
Inst> 1 *
View: distribution
Status: N=new, U=upgrade, S=same, D=downgrade
Stream: maint
Selection: i=install, r=remove, k=keep

Subsystem Types [bdrpc]: b=reBoot needed, d=Default, r=Required,
p=patch, c=Client only
```

```
i U failsafe2_oracle.books.book_AG [d] 0 FailSafe ORACLE Plugin 2.0
Administrator's Guide
i U failsafe2_oracle.man.relnotes 0 FailSafe ORACLE Plugin 2.0 release
notes
i U failsafe2_oracle.sw.base [d] 4+ FailSafe ORACLE Plugin 2.0 base
software
```

```
Disk space summary (Kbytes): / /nfs_cml /apps
```

```
Current free space 2277016 4422412 5656968
- Selections net change 4+ 0 0
- Temporary inst overhead 1296+ 0 0
= Minimum free during install 2275716 4422412 5656968
```

```
Final projected free space 2277012 4422412 5656968
```

```
Inst> g
Pre-installation check .. 8%
Checking space requirements .. 16%
Installing/removing files .. 16%
Upgrading selected failsafe2_oracle.books subsystems
Installing/removing files .. 65%
Upgrading selected failsafe2_oracle.man subsystems
Installing/removing files .. 72%
Upgrading selected failsafe2_oracle.sw subsystems
Installing/removing files .. 94%
Running exit-commands .. 99%
Checking dependencies .. 100% Done.
Installations and removals were successful.
You may continue with installations or quit now.
```

```
Inst> quit
```

Creating the Oracle_DB Resource Type

You can use one of the following methods to create the Oracle_DB resource type:

- Run cluster manager (cmgr) and manually create the resource type. For more information, see the *IRIS FailSafe 2.0 Administrator's Guide*.

- Run cluster manager (cmgr) and install the resource type, as follows:

```
cmgr> show resource_types installed
```

```
NFS
template
Netscape_web
statd
Oracle_DB
MAC_address
IP_address
INFORMIX_DB
filesystem
volume
```

```
cmgr> install resource_type Oracle_DB in cluster eagan
```

```
cmgr>
```

- Use the template scripts supplied with IRIS FailSafe 2.0 located in /var/cluster/cmgr-template/cmgr-create-resource_type.
- Execute /var/cluster/ha/resource_type/Oracle_DB/create_resource_type and include the path of the CDB argument and the cluster name.
- Use the **Load Resource Type** GUI task to load the resource type.

Creating an Oracle Resource

After you have defined the resource type, the administrator must define the Oracle_DB resources based on the resource type. Each resource requires a unique resource name (for example, the Oracle_DB resource type is the Oracle instance name). Then, the administrator must supply the resource parameters. To create the resource, either use the cluster manager (cmgr), the cmgr template scripts, or the GUI.

Example 2-4 Creating the Oracle_DB Resource

```
cmgr> create resource oracle8 of resource_type Oracle_DB in cluster eagan
```

Enter commands, when finished enter either "done" or "cancel"

Type specific attributes to create with set command:

```
Type Specific Attributes - 1: shutdown-script
Type Specific Attributes - 2: db-master-fs
Type Specific Attributes - 3: release-dir
Type Specific Attributes - 4: oracle-user
Type Specific Attributes - 5: startup-script
```

Resource type dependencies to add:

```
Resource Dependency Type - 1: IP_address
Resource Dependency Type - 2: filesystem
```

```
resource oracle8 ? set release-dir to /disk1/oracle
resource oracle8 ? set startup-script to /disk1/oracle/dbstart
resource oracle8 ? set shutdown-script to /disk1/oracle/dbstop
resource oracle8 ? set db-master-fs to /disk1
resource oracle8 ? set oracle-user to oracle
resource oracle8 ? done
Successfully created resource oracle8
```

```
cmgr> modify resource oracle8 of resource_type Oracle_DB in cluster eagan
```

Enter commands, when finished enter either "done" or "cancel"

Type specific attributes to modify with set command:

```
Type Specific Attribute - 1: shutdown-script
Type Specific Attribute - 2: db-master-fs
Type Specific Attribute - 3: release-dir
Type Specific Attribute - 4: oracle-user
Type Specific Attribute - 5: startup-script
```

No resource type dependencies to add

```
resource oracle8 ? add dependency /disk1 of type filesystem
resource oracle8 ? add dependency 128.162.101.20 of type IP_address
resource oracle8 ? done
Successfully modified resource oracle8
```

```
cmgr>
```

Creating an Oracle Resource Group

To create a resource group, you must first become familiar with the terms and concepts of FailSafe. A resource group can be created either by the GUI or the cluster manager (cmgr).

To define an effective resource group, you must include all of the resources that the Oracle_DB resource is dependent on, such as file systems, volumes, and IP addresses. The following example shows the creation of a typical resource group:

```
cmgr> create resource_group oracle in cluster eagan
Enter commands, when finished enter either "done" or "cancel"

resource_group oracle ? set failover_policy to ordered-in-order
resource_group oracle ? add resource oracle8 of resource_type Oracle_DB
resource_group oracle ? add resource 128.162.101.20 of resource_type
IP_address
resource_group oracle ? done
Successfully created resource group oracle

cmgr> show resource_group oracle in cluster eagan
Resource Group: oracle
Cluster: eagan
Failover Policy: ordered-in-order
Resources:
oracle8 (type: Oracle_DB)
128.162.101.20 (type: IP_Addresses)
```

Installing Startup and Shutdown Scripts

To install Oracle startup and shutdown scripts, follow these guidelines:

- If they are to be installed on a shared disk, install them from the disk's primary node.
- If they are to be installed on non-shared disks, install them on each node.
- Install the scripts in the locations specified by the startup script and shutdown script for the Oracle database instances that use them.

- If you are using different scripts on each node, verify that the correct script is installed on each node.
- The mode of the script should be 755.

Testing the Oracle Resource

To ensure that the `Oracle_DB` resource has been correctly configured, you can test individual actions by executing the scripts. Each script, located at `/var/cluster/ha/resource_types/Oracle_DB`, requires two arguments, an input file and an output file. The contents of these files are the resource names. The scripts will display either a 0 if they are successfully executed or a positive integer that indicates the error type. For more information on error codes, see the *IRIS FailSafe 2.0 Programmer's Guide*.

In the following example, you can test the `start` script by starting the Oracle resource with the resource name `oracle8`.

```
$ cd /var/cluster/ha/resource_types/Oracle_DB
$ echo "oracle8" > /tmp/ipfile
$ ./start /tmp/ipfile /tmp/opfile
```

This should start the `oracle8` instance, named by the Oracle resource `oracle8`.

To view the individual script actions, you must edit the script, and add `"set -x"` to the action function.

In the following example, to see what the `start_oracle()` function does, edit the start script and add `"set -x"` in the start function.

```
#
# Start the resource on the local machine.
# Return HA_SUCCESS if the resource has been successfully started on
# the local machine and HA_CMD_FAILED otherwise. The resource name is #
# the oracle instance id.
#
start_oracle()
{
set -x
```

```
# for all oracle resource configured
...
```

You are now ready to run the script. In this example, /dev/null is the name of the output file.

```
# ./start /tmp/ipfile /dev/null
+ get_oracle_info oracle8
+ [ 0 -ne 0 ]
+ ha_get_field shutdown-script /disk1/oracle/dbstop
db-master-fs /disk1
release-dir /disk1/oracle
oracle-user oracle
startup-script /disk1/oracle/dbstart release-dir
+ [ 0 -ne 0 ]
+ export ORACLE_HOME=/disk1/oracle
+ export ORACLE_SID=oracle8
+ oracle_probe oracle8
+ [ 1 -eq 1 ]
+ ha_get_field shutdown-script /disk1/oracle/dbstop
db-master-fs /disk1
release-dir /disk1/oracle
oracle-user oracle
startup-script /disk1/oracle/dbstart startup-script
+ [ 0 -ne 0 ]
+ STARTUP_SCRIPT=/disk1/oracle/dbstart
+ ha_get_field shutdown-script /disk1/oracle/dbstop
db-master-fs /disk1
release-dir /disk1/oracle
oracle-user oracle
startup-script /disk1/oracle/dbstart oracle-user
+ [ 0 -ne 0 ]
+ UNIX_USER=oracle
```

```
+ [ -x /disk1/oracle/dbstart ]
+ ha_write_status_for_resource oracle8 4
```

Testing the start Script

You can use the following procedure to test the start script.

1. To test the start script, you must shut down the Oracle database on the node (if it is already running), as in the following example:

```
# export ORACLE_HOME release-dir
# export ORACLE_SID resource-name
# su oracle-user -c shutdown-script
```

The *release-dir*, *resource-name*, *oracle-user*, and *shutdown-script* configuration parameters are from the `Oracle_DB` resource definition. This procedure should stop the Oracle instance.

2. To start the Oracle instance, enter the following command:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/start /tmp/ipfile
/dev/null
```

3. You can then check to see if the database is running and if you can access the database. You can check for the Oracle database processes by executing the following command:

```
# ps -ef | grep oracle
```

Testing the stop Script

You can use the following procedure to test the stop script.

1. To test the stop script, you must start the Oracle database on the node (if it is not already running), as in the following example:

```
# export ORACLE_HOME release-dir
# export ORACLE_SID resource-name
# su oracle-user -c startup-script
```

The `release-dir`, `resource-name`, `oracle-user`, and `startup-script` configuration parameters are from the `Oracle_DB` resource definition. This procedure should start the Oracle instance.

2. To stop the Oracle instance, enter the following command:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/stop /tmp/ipfile
/dev/null
```

Check to ensure that the database has shut down.

Testing the monitor Script

You can use the following procedure to test the monitor script.

1. To test the monitor script, enter the following command:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/monitor /tmp/ipfile
/dev/null
```

2. As with all scripts, the output is logged to the `/var/cluster/ha/log/script_hostname` files. To check that the script has worked, use the `tail` command and execute the script as in the following example:

```
tail /var/cluster/ha/log/script_hebl
```

The following example shows the testing of the monitor script.

```
# echo "oracle8" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/monitor /tmp/ipfile
/dev/null ; echo $?
+ oracle_probe oracle8
+ HA_CMD=/usr/cluster/bin/ha_exec2 -s 0 -t ora_pmon_oracle8
+ ha_execute_cmd Checking for oracle process (ora_pmon_oracle8)
+ [ 101 -eq 0 ]
+ ha_log Failed to find process ora_pmon_oracle8
+ return 1
+ [ 1 -eq 0 ]
```

```
+ ha_write_status_for_resource oracle8 2
0
# tail /var/cluster/ha/log/script_cml
Wed Oct  7 16:37:01.078 <ha_cilog script 22618:0>
/var/cluster/ha/resource_types/Oracle_DB/monitor[24196]:
/var/cluster/ha/resource_types/Oracle_DB/monitor called with
/tmp/ipfile and /dev/null
Wed Oct  7 16:37:01.103 <ha_cilog script 23725:0>
/var/cluster/ha/resource_types/Oracle_DB/monitor[24196]: Checking for
oracle process (ora_pmon_oracle8)
Wed Oct  7 16:37:01.147 <ha_cilog script 23077:0>
/var/cluster/ha/resource_types/Oracle_DB/monitor[24196]:
Wed Oct  7 16:37:01.169 <ha_cilog script 23280:0>
/var/cluster/ha/resource_types/Oracle_DB/monitor[24196]:
/usr/cluster/bin/ha_exec2 -s 0 -t ora_pmon_oracle8 exited with status
101
Wed Oct  7 16:37:01.195 <ha_cilog script 23956:0>
/var/cluster/ha/resource_types/Oracle_DB/monitor[24196]: Failed to find
process ora_pmon_oracle8
```

Testing the restart Script

You can use the following procedure to test the restart script.

1. To test the restart script, enter the following command:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/restart
/tmp/ipfile /dev/null
```

Testing the exclusive and probe Scripts

You can use the following procedure to test the exclusive and probe scripts.

1. To test the exclusive and probe scripts, enter one of the following sets of commands:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/exclusive
```

```
/tmp/ipfile /dev/null
```

For the probe script, enter the following command:

```
# echo "resource-name" > /tmp/ipfile
# /var/cluster/ha/resource_types/Oracle_DB/probe
/tmp/ipfile /dev/null
```

2. As with all scripts, the output is logged to the `/var/cluster/ha/log/script_hostname` files. To check that the script has worked, use the `tail` command and execute the script as in the following example:

```
tail /var/cluster/ha/log/script_hebl
```

The following example shows the testing of the probe script:

```
/var/cluster/ha/resource_types/Oracle_DB/probe /tmp/ipfile /dev/null
echo $?
0
tail /var/cluster/ha/log/script_cm1
Wed Oct 7 16:44:58.701 <ha_cilog script 20709:0>
/var/cluster/ha/resource_types/Oracle_DB/probe[24983]:
/var/cluster/ha/resource_types/Oracle_DB/probe called with /tmp/ipfile
and /dev/null
```

Testing Resource Group Failovers

You can test the failover policy by using either `cmgr` or the GUI to move the resource group to another node in the cluster. To ensure that the resource group correctly failed over, use `cmgr` or the GUI to display the resource group states. The following example uses `cmgr` to test the failover policy:

```
cmgr> admin offline resource_group oracle in cluster eagan
Resource group is changing state -- please wait
```

```
cmgr> admin move resource_group oracle in cluster eagan to node cm2
Resource group is changing state -- please wait
```

```
cmgr> admin online resource_group oracle in cluster eagan  
Resource group is changing state -- please wait
```

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 007-3933-001.

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: 650-932-0801
- To send your comments by **traditional mail**, use this address:

Technical Publications
Silicon Graphics, Inc.
1600 Amphitheatre Parkway, M/S 535
Mountain View, California 94043-1351