



LifeKeeper[®] for Linux

Oracle[®] Recovery Kit v7.2 Administration
Guide

February 2011

SteelEye and LifeKeeper are registered trademarks.

Adobe Acrobat is a registered trademark of Adobe Systems Incorporation. Apache is a trademark of The Apache Software Foundation. HP and Compaq are registered trademarks of Hewlett-Packard Company. IBM, POWER, DB2 and Informix are registered trademarks or trademarks of International Business Machines Corporation. Intel, Itanium, Pentium and Xeon are registered trademarks of Intel Corporation. Java is a registered trademark of Sun Microsystems, Inc. Linux is a registered trademark of Linus Torvalds. Microsoft Internet Explorer and Windows are registered trademarks of Microsoft Corporation. MySQL and MaxDB are registered trademarks or trademarks of MySQL AB. Netscape and Netscape Navigator are registered trademarks of Netscape Communications Corporation. NFS is a registered trademark of Sun Microsystems, Inc. Opteron is a trademark of Advanced Micro Devices, Inc. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. PostgreSQL is a trademark of PostgreSQL Global Development Group. Red Flag is a registered trademark of Red Flag Software Co.,Ltd. Red Hat is a registered trademark of Red Hat Software, Inc. SAP is a registered trademark of SAP AG. Sendmail is a registered trademark of Sendmail, Inc. Sun and Solaris are registered trademarks of Sun Microsystems, Inc. SUSE is a registered trademark of SUSE LINUX AG, a Novell business. Other brand and product names used herein are for identification purposes only and may be trademarks of their respective companies.

It is the policy of SIOS Technology Corp. (previously known as SteelEye Technology, Inc.) to improve products as new technology, components, software, and firmware become available. SIOS Technology Corp., therefore, reserves the right to change specifications without prior notice.

To maintain the quality of our publications, we need your comments on the accuracy, clarity, organization, and value of this book.

Address correspondence to:
ip@us.sios.com

Copyright © 2011
By SIOS Technology Corp.
San Mateo, CA U.S.A.
All Rights Reserved

Table of Contents

Introduction	3
Document Contents	3
LifeKeeper Documentation	3
Requirements	4
Kit Hardware and Software Requirements	4
Configuring Oracle with LifeKeeper	5
Specific Configuration Considerations for Oracle	5
Configuring the Oracle Net Listener for LifeKeeper Protection	8
Listener Configuration	8
Configuring Transparent Application Failover with LifeKeeper	10
Configuration Examples	12
Configuration Requirements	13
Active/Standby Configurations	13
Active/Active Configurations	16
LifeKeeper Configuration Tasks	20
Creating an Oracle Resource Hierarchy	20
Deleting a Resource Hierarchy	23
Extending Your Hierarchy	25
Unextending Your Hierarchy	28
Change Username / Password for the Oracle Database Account	29
Testing Your Resource Hierarchy	31
Performing a Manual Switchover from the GUI	31
Recovery Operations	31
Troubleshooting	32
Hierarchy Creation and Extend Errors	32
Hierarchy Restore and Remove Errors	36
Bringing an Oracle Resource In Service (Restore)	36
Taking an Oracle Resource Out of Service (Remove)	36
Maximum number of connections exhausted	36
Troubleshooting Hints and Tips	36
Control File Switchover Failure	36
Truncated Output	36
Flash Recovery Destination Located on a Shared Drive	37
Database Creation Problems	38
Database Startup Problems	38
inqfail error in the LifeKeeper Log	39
Appendix	40
Setting up Oracle to use Raw I/O	40
Adding a Tablespace After Creating Hierarchy	41
Creating a Shared Oracle Listener for Multiple Resources	41
Updating the Listener Protection Level	44
Updating the Listener Recovery Level	44
Updating the Protected Listener(s)	45

Oracle[®] Recovery Kit Administration Guide

Introduction

The LifeKeeper[®] for Linux Oracle[®] Recovery Kit provides fault resilience for Oracle software in a LifeKeeper environment. The Recovery Kit software furnishes a mechanism to tie the data integrity of Oracle databases to the increased availability provided by LifeKeeper.

Document Contents

This guide explains the following topics:

- [LifeKeeper Documentation](#). A list of all the LifeKeeper for Linux documentation and where the information is available.
- [Requirements](#). Before you can install and set up the recovery software, your server must meet certain hardware and software requirements. You should refer to the *LifeKeeper for Linux Planning and Installation Guide* for specific instructions on how to install or remove the LifeKeeper Oracle Recovery Kit.
- [Configuring Your Recovery Kit](#). To ensure that your LifeKeeper configuration provides the protection and flexibility you require, you need to be aware of the configuration rules. To appropriately plan your configuration, you must understand your network configuration, interface selection, user system setup, hierarchy options and the Oracle configuration tasks. In addition to planning your configuration, this section also includes configuration examples and the specific tasks required to configure your Recovery Kit.
- [Troubleshooting](#). This section provides a list of informational and error messages with recommended solutions.

LifeKeeper Documentation

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- *LifeKeeper for Linux Release Notes*
- *LifeKeeper for Linux Online Product Manual* (available from the Help menu within the LifeKeeper GUI)
- *LifeKeeper for Linux Planning and Installation Guide*

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is provided on the SIOS Technology Corp. website at:

<http://us.sios.com/support>

Requirements

Before attempting to install or remove the LifeKeeper Oracle Recovery Kit, you must understand the hardware and software requirements and the installation and removal procedures.

Kit Hardware and Software Requirements

Before installing and configuring the LifeKeeper Oracle Recovery Kit, be sure that your configuration meets the following requirements:

- **Servers.** The Recovery Kit requires two or more *supported* computers configured in accordance with LifeKeeper requirements described in the *LifeKeeper Online Product Manual* and the *LifeKeeper Release Notes*, which are shipped with the product media.
- **Shared Storage.** Oracle databases must reside on shared disk in a LifeKeeper environment.
- **LifeKeeper software.** You must install the same version of LifeKeeper software and any patches on each server. Please refer to the *LifeKeeper Release Notes* and *Online Product Manual* for specific LifeKeeper requirements.
- **LifeKeeper IP Recovery Kit.** This Recovery Kit is required if remote clients will be accessing the Oracle Database. You must have the same version of this Recovery Kit on *each* server.
- **IP network interface.** Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.
Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.
- **TCP/IP software.** Each server also requires the TCP/IP software.
- **Oracle Software.** Each server must have the Oracle software installed and configured before you can configure LifeKeeper and the LifeKeeper Oracle Recovery Kit. The same version should be installed on each server. Consult the *LifeKeeper Release Notes* or *your* sales representative for the latest release compatibility and ordering information.

You should refer to the *LifeKeeper for Linux Planning and Installation Guide* for specific instructions on how to install or remove the LifeKeeper Oracle Recovery Kit.

Configuring Oracle with LifeKeeper

This section contains information you should consider before you start to configure Oracle and examples of typical Oracle configurations.

For instructions on installing Oracle on Linux distributions using the 2.4 kernel, please see your Linux distribution's website.

Also, please refer to your *LifeKeeper Online Product Manual* for instructions on configuring your LifeKeeper Core resource hierarchies.

Specific Configuration Considerations for Oracle

Note: If you plan to use Oracle with Raw I/O, you must install the LifeKeeper Raw I/O Recovery Kit from the LifeKeeper Core CD. You must also properly set up the Raw I/O devices prior to use. See the [Appendix](#) for instructions.

Before configuring the Oracle Recovery Kit, complete the following preparatory steps to ensure that file systems and disk partitions used by Oracle will be accessible from each server.

1. **Remove personal initialization file prompts.** For the Oracle Recovery Kit to work properly, you must remove (or comment out) all prompts in the personal initialization file (i.e., *.profile*, *.bash_profile*) for the Oracle user. This file is specific to the shell that is being used by the Oracle user. The file cannot be interactive.

Note: If “stty” statements are going to be in the personal initialization file, they must be in an “if” statement that verifies that an interactive terminal is being used.

2. **Configure Kernel Parameters.** Please refer to the Oracle documentation for information on how linux kernel parameters such as shared memory and other kernel resources should be configured. An example of how to set these parameters is below. (**Note:** If you are using the Oracle Recovery Kit for LifeKeeper on Novell SLES, installing the *orarun* rpm that is included with the distribution will take care of most of these sorts of configuration points.)

On *each server* in the cluster:

- A. Set the following **ipcs** limits in */etc/sysctl.conf* before configuring LifeKeeper.

```
# changes for Oracle
kernel.shmmax = <value>
kernel.shmmni = <value>
kernel.shmall = <value>
kernel.sem = <value>
```

- B. Run **sysctl -p** to set the above changes in the kernel.
- C. On certain distributions you may need to add `sysctl -p` to the system initialization file (i.e. *boot.local* or *rc.local*) so that these kernel changes are set after each reboot.

3. **\$ORACLE_HOME directory.** When you configure the \$ORACLE_HOME directory and associated files on local disks, be sure that the \$ORACLE_HOME directory and files are identical on all servers. Use the standard Linux utilities to create and copy directories and files to the set of servers.
Note: In certain active/active configurations, the location of \$ORACLE_HOME are different.
4. **Location.** The \$ORACLE_HOME directory can be on shared or non-shared disks. The advantage to having the directory on shared media is that you only need to configure files such as the parameter file *Oracle_HOME/dbs/<initSID.ora or spfileSID.ora>* once, if the same shared disk is used for \$ORACLE_HOME (e.g. in an active/standby configuration). The disadvantage to the shared directory is that direct access to the file system is available to only one server at a time. SCSI reservations permit only one server at a time access to a LifeKeeper protected shared drive. If creating an active-active cluster configuration where two or more Oracle instances (SID) will be protected independently in the cluster, \$ORACLE_HOME must be installed on local, non-shared storage.
5. **User and Group ID.** An oracle user (*oracle*) and group (*dba*) should be created on all servers. The user ID and group ID numbers must be the same on all servers. (**Note:** If you are using the Oracle Recovery Kit for LifeKeeper on Novell SLES, installing the orarun rpm that is included with the distribution will take care of most of these sorts of configuration points.)
6. **Databases, archive files, log files, and control files.** All databases, archive files, log files, and control files *must* be created on shared file systems or disk partitions. These locations are set in the Oracle parameter file *init<SID>.ora or spfileSID.ora*. Please refer to the Oracle documentation for information on editing database parameters. The pathnames must be the same for all servers. Oracle internally keeps this information in its control file; therefore, SYSTEMS database space and paths cannot be changed unless Oracle is running.

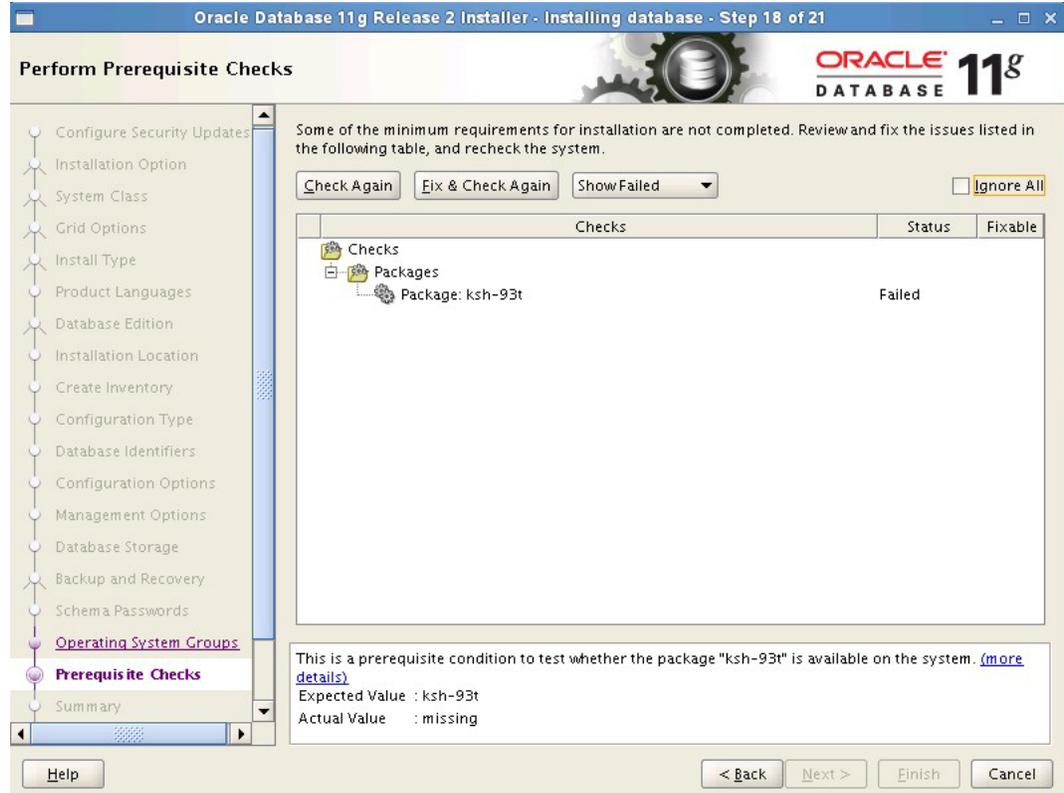
Note that Oracle log archiving is not enabled by default. If it is enabled prior to the creation of the LifeKeeper Oracle hierarchy, LifeKeeper will detect the location of the archive files and create a separate file system hierarchy if necessary. But if log archiving is enabled after the LifeKeeper Oracle hierarchy has been created, you must manually create and extend a file system hierarchy to protect the shared archive location, and create a dependency from the Oracle resource to this new file system hierarchy.

7. On a new installation of Oracle, the final configuration of the database instance is easier if the database installation program is **not allowed** to create a database. When the installer asks if you want to create a database, select **No**. After the installation is complete, run the Oracle Database Creation Assistant (dbca). Dbca provides much better control of where database components get created. When running dbca, specify that the Flash Recovery Area gets created on LifeKeeper protected storage (this applies to Oracle 11g).

Important The Flash Recovery Destination must be located on a shared drive.

If runInstaller is allowed to create a database, the Flash Recovery Area will have to be relocated manually.

8. During the installation of Oracle using the “runInstaller” utility, there will be a point where the installer verifies the packages and configuration of Linux before proceeding with the Oracle database installation. If LifeKeeper 7.2 (or higher version) has already been installed, a message complaining about a missing ksh package will appear.



If this message is displayed, check the box in the upper righthand corner, **Ignore All**. The installation of LifeKeeper has removed the ksh package and replaced it with the Public Domain Korn Shell, pdksh. Oracle should install fine using pdksh.

9. **Tune the database engine.** Refer to Oracle documentation for guidelines on tuning the database engine for data integrity and performance. In particular, the tuning for memory caching and checkpointing frequency is critical to optimizing the application for fault resilience. The checkpointing interval determines the number of database transactions that have not been committed to disk and therefore would be lost during a system failure.
10. **Database entry in oratab file.** The `/etc/oratab` file must contain an entry for the database. The LifeKeeper configuration routines use the contents of this file to relate `$ORACLE_HOME` and `$ORACLE_SID` values. Usually, the Oracle installation program creates the required entry. In a configuration in which the Oracle software is installed to a shared file system, however, you must copy the `oratab` file from the server where the Oracle installation was performed to the `/etc` directory of the other servers so that it is available to all the servers.

Note: The configuration can have only one `oratab` per server. Refer to the *Oracle Product Manual* for information on the file format.

11. **Disable automatic start-ups.** Since LifeKeeper is responsible for starting the databases it controls, be sure to disable any automatic start-up actions. LifeKeeper disables automatic start-up when a hierarchy is created. This is accomplished by modifying the `oratab` file.

12. **The Listener configuration file, listener.ora.** New lines should not be embedded in the entries (e.g., SID_NAME=xx should be on one line).
13. **Oracle Database Username and Password.** LifeKeeper will use the specified Username and Password to control Oracle Database. The Oracle Database user must be able to connect as sysdba authority to the database to be protected and each server's Oracle Database must have the same username and password. If this configuration is skipped during resource creation, then LifeKeeper will not use Username and Password to control the Oracle Database resource. This parameter can be added, changed or removed anytime after creating the resource.

Tips for Creating the Oracle Username and Password.

- a. On the node where the Oracle database is running, log in to Linux with a user that is part of the dba group. (The "oracle" account is most common.) Using the `sqlplus` utility, connect to the database as the administrative user by issuing the following command:

```
$ sqlplus / as sysdba
```

- b. Create a new user for this function:

```
SQL> CREATE USER lkdba IDENTIFIED BY "password";
```

- c. Then grant this user SYSDBA privileges:

```
SQL> GRANT SYSDBA to lkdba;
```

- d. If Oracle has been configured so that each node in the LifeKeeper cluster has a local copy of `$ORACLE_HOME`, execute these commands on each node in the cluster. After creating the LifeKeeper Oracle hierarchy, bring the database in service on the node and then execute the CREATE and GRANT commands (above) to set up the user in Oracle.

CAUTION: Avoid configuring two databases on the same file system. If you must configure two databases on the same file system, exercise great care. In this situation, both databases must be placed under LifeKeeper protection and both hierarchies must have the same primary and backup servers.

Configuring the Oracle Net Listener for LifeKeeper Protection

If your Oracle database will have remote client connections, you will want to protect the Oracle Listener in addition to the Oracle database server. Please refer to the Oracle documentation for information on using Oracle network configuration utilities to create Oracle network configuration files such as `listener.ora` and `tnsnames.ora`.

Note: Refer to the [Creating a Shared Oracle Listener for Multiple Resources](#) section in the appendix in this document for instructions on how to create a shared Oracle Listener for multiple resources.

Listener Configuration

1. You need to choose a switchable IP address for clients to make connections to. You may want to put this address in DNS. (Refer to the *LifeKeeper IP Recovery Kit Administration Guide* for details on creating an IP resource hierarchy. Refer to the *LifeKeeper Online*

Product Manual topic, *Creating a Resource Dependency*, under the GUI Administration Tasks book for details on creating a resource dependency.)

2. In the *listener.ora* file, specify this switchable IP address or the DNS name for the switchable IP address as the **HOST** for the database service name. (See the Oracle documentation for details about the *listener.ora* file.)

Sample format of a *listener.ora*:

```
.
.
.
<listener name>=
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = <switchable IP>) (PORT =
        <port number>))
    )
  )
.
.
.
```

3. Specify the switchable IP address or the DNS name for the switchable IP address as the **HOST** in the *tnsnames.ora* file or Oracle Names. (See the Oracle documentation for details about the *tnsnames.ora* file.)

Sample format of a *tnsnames.ora* file:

```
.
.
.
<SID Name>=
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = <switchable IP>) (PORT =
        <port number>))
    )
    (CONNECT_DATA =
      (SID = <SID Name>)
    )
  )
)
```

These sample files should work with both Oracle 10g and 11g:

listener.ora

LISTENER =

```
(DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = 192.0.2.0) (PORT = 1521))
  )
)

tnsnames.ora

ORA01 =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP) (HOST = 192.0.2.0) (PORT = 1521))
  )
  (CONNECT_DATA =
    (SID = ORA01)
  )
)
```

The normal location of `listener.ora` is in `$ORACLE_HOME/network/admin`. The most common port number is 1521. The global name of the database was defined at creation time. Also keep in mind, if the `$ORACLE_HOME` directory is installed on non-shared storage, a copy of `listener.ora` will need to be on both systems.

Note: Oracle Net provides the option of automatically failing over client connections to another listener if the listener for a service should fail. To take advantage of this feature, set the `FAILOVER` parameter to “ON” in the `tnsnames.ora` file. If the listener for the LifeKeeper-protected Oracle SID should fail, this allows client connections to continue through another listener until LifeKeeper recovers the protected listener.

Configuring Transparent Application Failover with LifeKeeper

When a server failover or an Oracle database failure occurs, users can be severely disrupted. Typically the user’s connections to the database will be lost along with most work in progress. Upon the completion of the failover (or recovery of the Oracle database), clients will have to restart their application and reconnect to the database. With the Transparent Application Failover (TAF) feature of Oracle, this disruption can be reduced or eliminated by masking some types of failures. To configure TAF in a LifeKeeper environment, there are tasks that must be performed on both the LifeKeeper server side and the Oracle client side.

For clients to effectively take advantage of the TAF feature, the client application must use failover-aware API calls from the Oracle Call Interface (OCI). The clients must also configure the appropriate TAF support using the Oracle Net parameters in the `tnsnames.ora` file. TAF mode can be configured by including a `FAILOVER_MODE` parameter under the `CONNECT_DATA` section of the `tnsnames.ora` connect descriptor. The TAF mechanism supports several sub-parameters to control and affect the behavior of a client connection during failover. The LifeKeeper for Linux Oracle Recovery Kit supports the following TAF configuration sub-parameters:

TYPE= (SELECT or SESSION).

This value determines how TAF will handle client connection failover. When the type is set to SELECT, Oracle keeps track of all select statements issued during transition. Upon establishment of a new connection, the select statements are re-executed, and the cursors repositioned so clients can continue to fetch rows. When type is set to SESSION only a new connection is created; work in progress may be lost.

METHOD= (BASIC).

With this method TAF will attempt a reconnect only after the primary connection fails. The alternative method is PRECONNECT, LifeKeeper does not currently support the use of PRECONNECT as a method.

DELAY= (#sec).

This value is the number of seconds that TAF will wait between attempts to connect following a failure. This value should be carefully determined for your client application and environment.

RETRIES= (#number of tries).

This value is the number of times that TAF will attempt to retry a failed connection before giving up. The combination of DELAY and RETRIES must allow enough time for a complete recovery of Oracle in the event of a server failure. This will give TAF enough time to restart after the server failover has completed.

An excerpt from a sample tnsnames.ora file for a client system is included below.

```
LKproDB=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL=TCP) (HOST=<switchableIP>) (PORT=<port number>))
    )
    (CONNECT_DATA=
      (SID=LKroDB)
      (SERVER=DEDICATED)
      (FAILOVER_MODE=
        (TYPE=SELECT)
        (METHOD=BASIC)
        (DELAY=5)
        (RETRIES=30)
      )
    )
  )
)
```

The normal location of tnsnames.ora is in \$ORACLE_HOME/network/admin. The most common port number is 1521. “tnsnames.ora” files can also be located in user’s home directories as well. Also keep in mind, if the \$ORACLE_HOME directory has been installed on non-shared storage, a copy of listener.ora and tnsnames.ora will need to be on both systems.

On the LifeKeeper server protecting the Oracle database, the listener should be configured using a LifeKeeper-protected switchable IP address. Refer to the [Configuring the Oracle Net Listener for LifeKeeper Protection](#) section above for details on configuring Oracle Net and listener support.

Configuration Examples

The following figures illustrate examples of both active/standby and active/active Oracle configurations in a LifeKeeper environment.

The examples in this section show how Oracle database instances can be configured on local and shared disks. Each diagram shows the relationship between the type of configuration and the Oracle parameters. Each configuration also adheres to the configuration rules and requirements described in this administration guide that ensure compatibility between the Oracle configuration and the LifeKeeper software.

This section first describes the configuration requirements and then provides these configuration examples:

- [Active/Standby](#)
- [Active/Active](#)

The examples in this section are only a sample of the configurations you could establish, but understanding these configurations and adhering to the configuration rules helps you define and set up workable solutions for your computing environment.

Configuration Requirements

Each of the examples involves one or two databases: **databaseA** and **databaseB**. By default, LifeKeeper offers a tag name matching the Oracle database system identifier (SID). However, the screen examples in the following pages use tag names consisting of the SID and server name such as *databaseA-on-server1*.

To understand the configuration examples, keep these configuration requirements in mind:

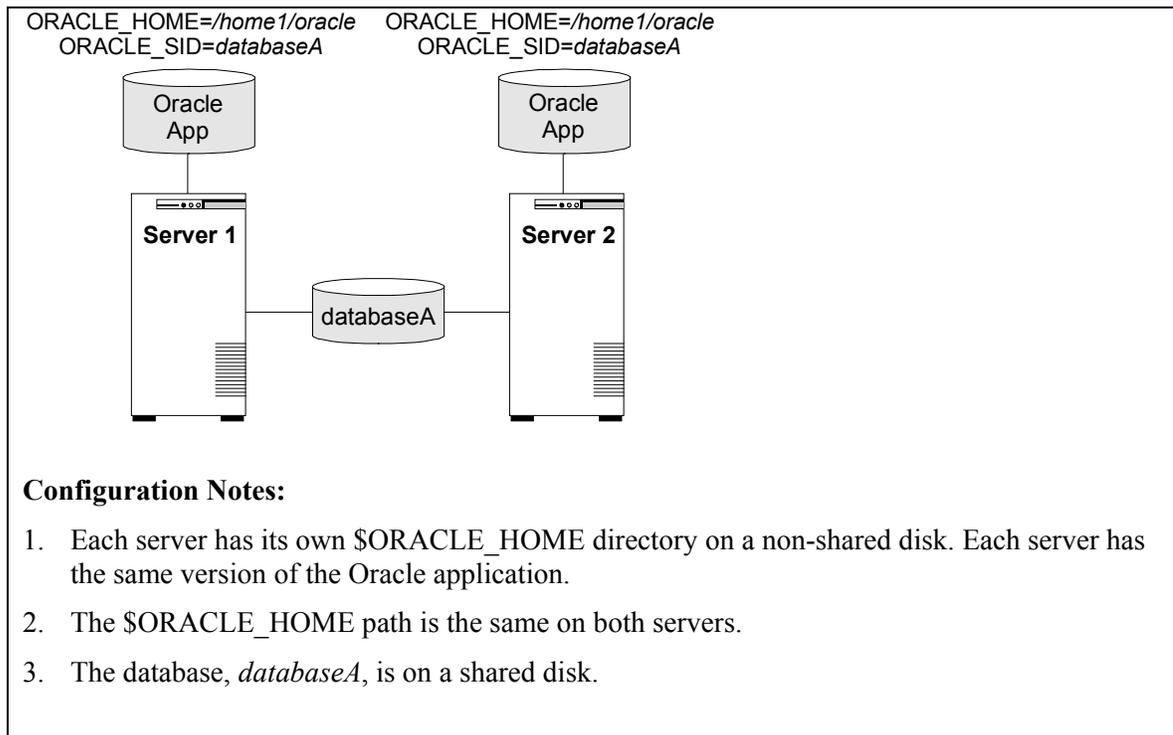
- **LifeKeeper hierarchy.** When performing LifeKeeper administration, the primary server refers to the location where the Oracle instance is currently running. System administration takes place on this server when creating a LifeKeeper hierarchy. For the configuration examples, the primary server is Server 1 and the backup or alternate server is Server 2.
- **Shared disk locked by one server.** When shared storage resources are under LifeKeeper protection, they can only be accessed by one server at a time. If the shared device is a disk array, an entire LUN is protected. If a shared device is a disk, then the entire disk is protected. This prevents inadvertent corruption of the data by other servers in the cluster. When a server fails, the highest priority backup server establishes its own protection, locking out all other servers.
- **Database on shared disk.** In order for the LifeKeeper Oracle Recovery Kit to function properly, the database must always be on a shared device. The database may be on one or more file systems and/or disks.

Note: The tables associated with each configuration illustration provide examples of the appropriate information that would be entered in the Create Resource Hierarchy wizard on the primary server (Server 1) and Extend Resource Hierarchy wizard to the backup server (Server 2). For additional detail on what information to enter into the wizards, refer to the “[LifeKeeper Configuration Tasks](#)” section later in this guide. These tables can be a helpful reference when configuring your Recovery Kit.

Active/Standby Configurations

This section provides two active/standby configuration examples, shown in Figure 1 and Figure 2. In these configurations, Server 1 is considered active because it has exclusive access to the database. Server 2 does other processing. If Server 1 fails, Server 2 gains access to the database and LifeKeeper re-establishes the database operations.

Figure 1. Active/Standby Configuration, Example 1

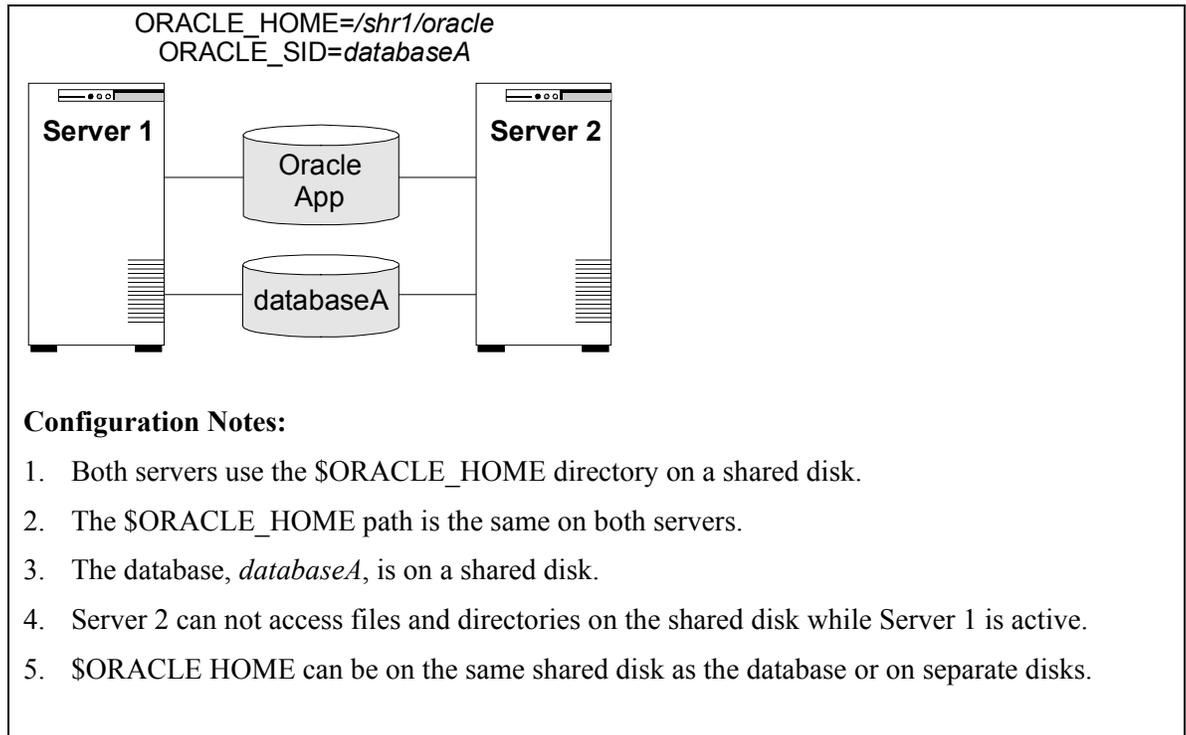


Creating a resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	/home1/oracle
Database Tag:	databaseA-on-server1

Extending the resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend	databaseA-on-server1
Target Server	Server2
Target Priority:	10
ORACLE_HOME for Target Server:	/home1/oracle
Database Tag:	databaseA-on-server2

Figure 2. Active/Standby Configuration, Example 2**Creating a resource hierarchy on Server 1:**

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	/shr1/oracle
Database Tag:	databaseA-on-server1

Extending the resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend	databaseA-on-server1
Target Server	Server2
Target Priority:	10
ORACLE_HOME for Target Server:	/shr1/oracle
Database Tag:	databaseA-on-server2

Active/Active Configurations

An active/active configuration consists of at least two servers, each running a different database instance. The databases **must** be on different shared disks.

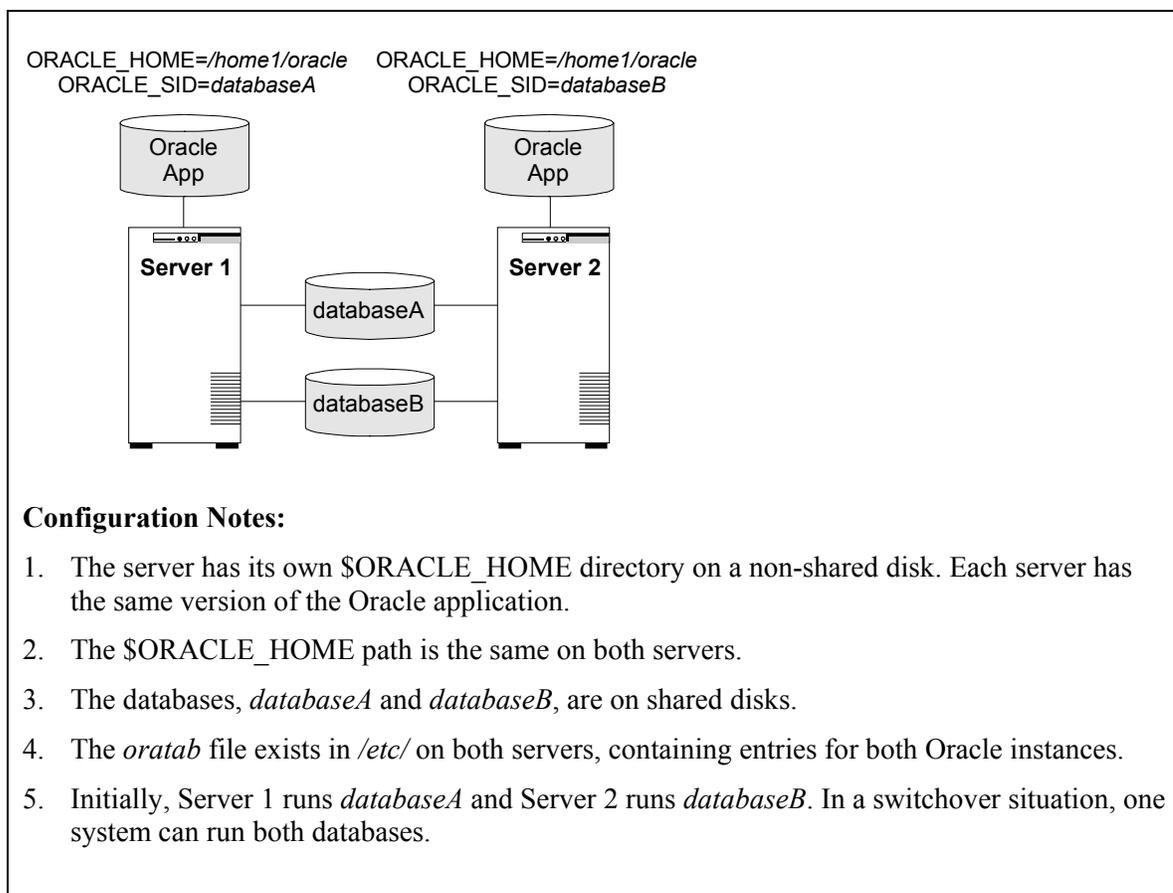
\$ORACLE_HOME can be on non-shared or on shared disks depending upon the configuration requirements. For example, multiple database instances on any of the servers using a common \$ORACLE_HOME require \$ORACLE_HOME to be on non-shared disks. If the \$ORACLE_HOME directories are on shared disk, they must be on separate shared disks.

This section provides two active/active configuration examples, shown in Figure 3 and Figure 4:

- Databases on shared resources and a common \$ORACLE_HOME on non-shared resources.
- Databases on shared resources and the appropriate \$ORACLE_HOME instance on the same shared resource.

Note: Multiple database instances on one server using multiple instances of \$ORACLE_HOME on non-shared resources are not illustrated.

Figure 3. Active/Active Configuration, Example 1



Creating the first resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	<i>/home1/oracle</i>
Database Tag:	databaseA-on-server1

Extending the first resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend	databaseA-on-server1
Target Server	Server2
Target Priority:	10
ORACLE_HOME for Target Server:	<i>/home1/oracle</i>
Database Tag:	databaseA-on-server2

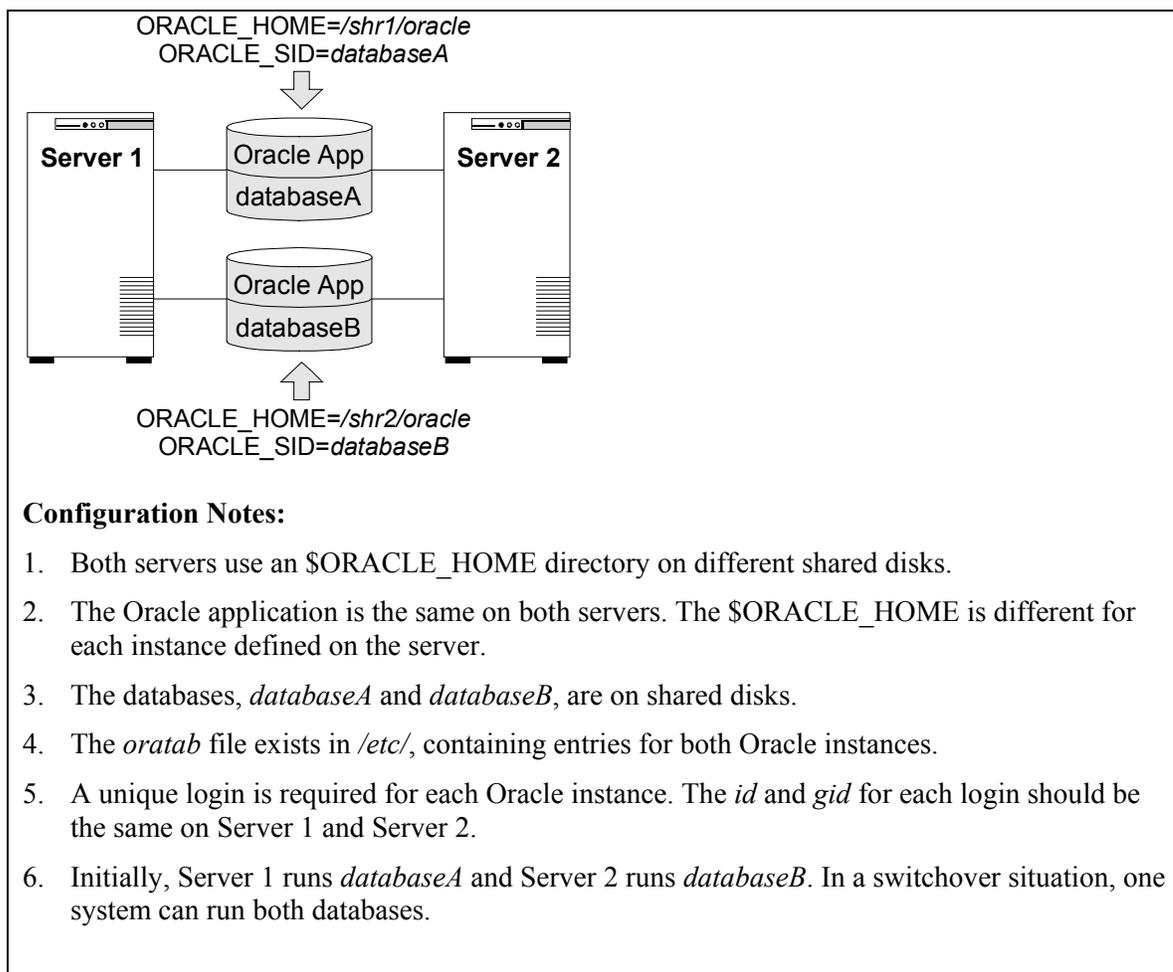
Creating a second resource hierarchy on Server 2:

Server:	Server2
ORACLE_SID for Database:	databaseB
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	<i>/home1/oracle</i>
Database Tag:	databaseB-on-server2

Extending the second resource hierarchy to Server 1:

Template Server:	Server2
Tag to Extend	databaseB-on-server2
Target Server	Server1
Target Priority:	10
ORACLE_HOME for Target Server:	<i>/home1/oracle</i>
Database Tag:	databaseB-on-server1

Figure 4. Active/Active Configuration, Example 2



Creating the first resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	/shr1/oracle
Database Tag:	databaseA-on-server1

Extending the first resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend	databaseA-on-server1
Target Server	Server2
Target Priority:	10
ORACLE_HOME for Target Server:	/shr1/oracle

Database Tag:	databaseA-on-server2
---------------	----------------------

Creating a second resource hierarchy on Server 2:

Server:	Server2
ORACLE_SID for Database:	databaseB
Username for Database	system
Password for Username	*****
ORACLE_HOME for Database:	<i>/shr2/oracle</i>
Database Tag:	databaseB-on-server2

Extending the second resource hierarchy to Server 1:

Template Server:	Server2
Tag to Extend	databaseB-on-server2
Target Server	Server1
Target Priority:	10
ORACLE_HOME for Target Server:	<i>/shr2/oracle</i>
Database Tag:	databaseB-on-server1

LifeKeeper Configuration Tasks

You can perform the following configuration tasks from the LifeKeeper GUI. The following four tasks are described in this guide, as they are unique to an Oracle resource instance and different for each Recovery Kit.

- **[Create a Resource Hierarchy](#)**. Creates an application resource hierarchy in your LifeKeeper cluster.
- **[Delete a Resource Hierarchy](#)**. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- **[Extend a Resource Hierarchy](#)**. Extends a resource hierarchy from the primary server to a backup server.
- **[Unextend a Resource Hierarchy](#)**. Unextends (removes) a resource hierarchy from a single server in the LifeKeeper cluster.
- **[Change Username / Password](#)**. Change the Username and Password to login to protect Oracle Database.

The following tasks are described in the GUI Administration section within the *LifeKeeper Online Product Manual*, because they are common tasks with steps that are identical across all Recovery Kits.

- **Create a Resource Dependency**. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- **Delete a Resource Dependency**. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- **In Service**. Brings a resource hierarchy into service on a specific server.
- **Out of Service**. Takes a resource hierarchy out of service on a specific server.
- **View/Edit Properties**. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, we explain how to configure your Recovery Kit by selecting certain tasks from the **Edit** menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You may also right click on a global resource in the Resource Hierarchy Tree (left-hand pane) of the status display window to display the same drop down menu choices as the **Edit** menu. This, of course, is only an option when a hierarchy already exists.

You can also right click on a resource instance in the Resource Hierarchy Table (right-hand pane) of the status display window to perform all the configuration tasks, except *Creating a Resource Hierarchy*, depending on the state of the server and the particular resource.

Creating an Oracle Resource Hierarchy

Note: In order to take advantage of Oracle Net remote client access, the IP address used for client connectivity must be under LifeKeeper protection as a dependent of the Oracle hierarchy. (Refer to the section [Configuring the Oracle Net Listener for LifeKeeper Protection](#) earlier in the guide for details.)

To create a resource instance from the primary server, you should complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.

Important The Oracle Application must be running when you create the resource.

A dialog box will appear with a drop down list box with all recognized Recovery Kits installed within the cluster. Select **Oracle Database** from the drop down listing. Click **Next** to proceed to the next dialog box.

Note: When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

2. Select the **Switchback Type**. This dictates how the Oracle instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.

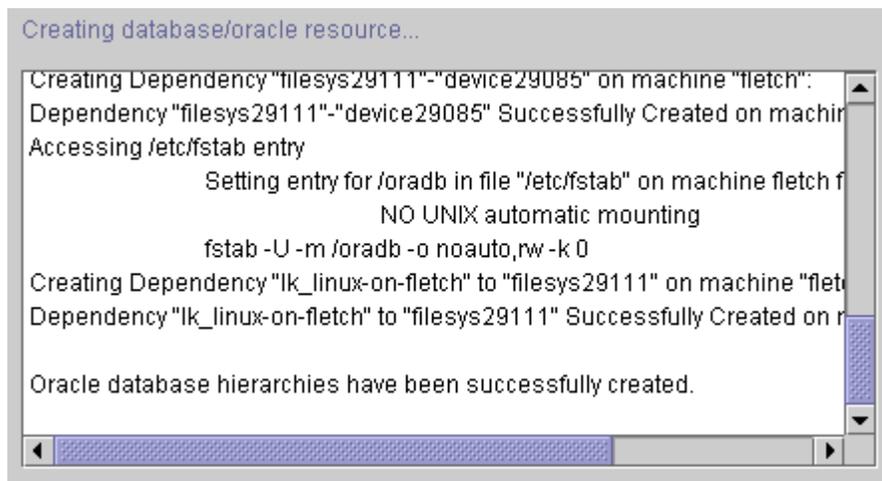
The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. Click **Next** to proceed to the next dialog box.

3. Select the **Server** where you want to place the Oracle Database (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down list box. Click **Next** to proceed to the next dialog box.
4. Select the **ORACLE_SID** for the Database ID. This is the tag name that specifies the Oracle system identifier of the database being configured. An entry for this database must exist in */etc/oratab*. Click **Next** to proceed to the next dialog box.
5. Input the **Username** for ORACLE_SID. This is the Oracle Database Username specified during login to ORACLE_SID. This username must be able to connect as sysdba authority to the database to gain full control. Click **Next** to proceed to the next dialog box. (This field can be left empty. If left empty, LifeKeeper will not use Username and Password to control the Oracle Database resource, and the next step, **Input Password**, will be skipped.)
6. Input **Password**. This is the password specified during login to ORACLE_SID. The password will be saved by LifeKeeper with encrypting. Click **Next** to proceed to the next dialog box.
7. Select or enter the **directory path** of the ORACLE_HOME for the Database SID being protected. This is the directory location where the Oracle application is located on the primary or template server. Click **Next** to proceed to the next dialog box.
8. Select the **tag name** of the Listener to be included as a dependency of the Oracle resource. The list displays all the currently protected Listener resource(s) on the server. Select the Listener resource tag that corresponds to the required listener(s) for the Oracle SID. Select **None**, if no Listener resource exists.

9. Select or enter the **Database Tag**. This is a tag name that LifeKeeper gives to the Oracle hierarchy. You can select the default or enter your own tag name.

When you click **Create**, the Create Resource Wizard will create your Oracle resource.

10. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your Oracle resource hierarchy. If LifeKeeper detects a problem, an **ERROR** will appear in the information box. If the validation is successful, your resource will be created.



Click **Next** to proceed to the next dialog box.

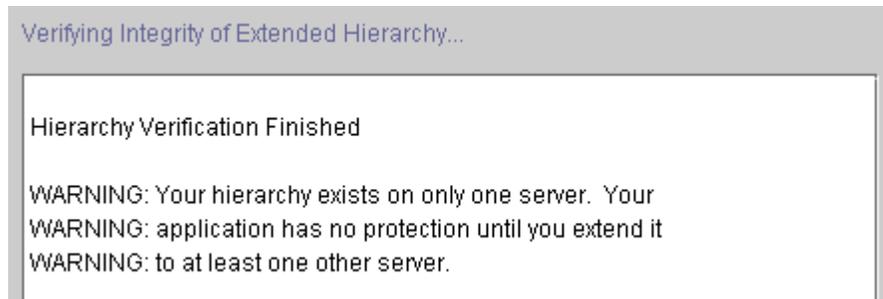
11. Another information box will appear explaining that you have successfully created an Oracle resource hierarchy, and you must **Extend** that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

You have successfully created a resource hierarchy on one server. You may select continue in order to extend this resource hierarchy to another server, or you may cancel at this point.

If you cancel, the resource hierarchy provides no protection for your applications until it is extended to at least one other server in the cluster.

When you click **Continue**, LifeKeeper will launch the Pre- Extend Wizard that is explained later in this section.

If you click **Cancel** now, a dialog box will appear warning you that you will need to come back and extend your Oracle resource hierarchy to another server at some other time to put it under LifeKeeper protection.



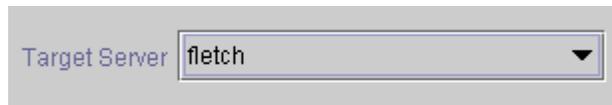
12. Click **Done** to exit the Create Resource Hierarchy menu selection.

Deleting a Resource Hierarchy

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy**.
2. Select the name of the **Target Server** where you will be deleting your Oracle resource hierarchy.

Note: If you selected the Delete Resource task by right clicking from the right pane on an individual resource instance, or from the left pane on a global resource where the resource is on only one server this dialog box will not appear.



Click **Next** to proceed to the next dialog box.

3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it.

Note: If you selected the Delete Resource task by right clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.



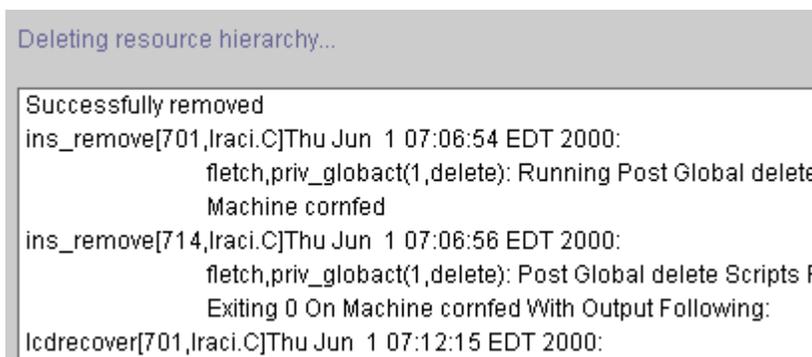
Click **Next** to proceed to the next dialog box.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.



Click **Delete** to delete your resource and proceed to the final dialog box.

5. Another information box appears confirming that the Oracle resource was deleted successfully.



6. Click **Done** to exit out of the Delete Resource Hierarchy menu selection.

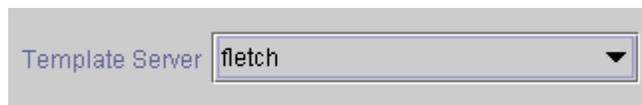
Note: Refer to the [Creating a Shared Oracle Listener for Multiple Resources](#) section in the appendix of this document for instructions on how to create a shared Oracle Listener for multiple resources.

Extending Your Hierarchy

After you have created a hierarchy, you will want to extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server. The first scenario is when you “Continue” from creating the resource into extending that resource to another server. The second scenario is when you enter the Extend Resource Hierarchy task from the edit menu as shown below. The third scenario is when you right click on an unextended hierarchy in either the left or right hand pane. Each scenario takes you through the same dialog boxes (with a few exceptions, which are clearly detailed below).

1. If you are entering the Extend wizard from the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Extend Resource Hierarchy**. This will launch the Extend Resource Hierarchy wizard.
2. The first dialog box to appear will ask you to select the **Template Server** where your Oracle resource hierarchy is currently in service. It is important to remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an *in service* resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you selected. The drop down box in this dialog provides the names of all the servers in your cluster.

Note: If you are entering the Extend Resource Hierarchy task immediately following the creation of an Oracle resource hierarchy, this dialog box will not appear, since the wizard has already identified the template server in the create stage. This is also the case when you right-click on either the Oracle resource icon in the left hand pane or right-click on the Oracle resource box in the right hand pane of the GUI window and choose **Extend Resource Hierarchy**.



It should be noted that if you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

For example, let's say you have created your resource on Server 1 and extended that resource to Server 2. In the middle of extending the same resource to Server 3, you change your mind and click on the **Cancel** button inside one of the dialog boxes. This will cancel only your action to extend the resource to Server 3, not the extension you created to Server 2. If you want to remove Server 2 from this hierarchy, you must unextend the resource from Server 2.

Click **Next** to proceed to the next dialog box.

3. Select the **Tag to Extend**. This is the name of the Oracle instance you wish to extend from the template server to the target server. The wizard will list in the drop down list box all the resources that you have created on the template server, which you selected in the previous dialog box.

Note: Once again, if you are entering the Extend Resource Hierarchy task immediately following the creation of an Oracle resource hierarchy, this dialog box will not appear, since the wizard has already identified the tag name of your Oracle resource in the create stage. This is also the case when you right-click on either the Oracle resource icon in the left hand

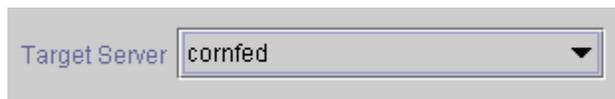
pane or on the Oracle resource box in the right hand pane the of the GUI window and choose **Extend Resource Hierarchy**.



A screenshot of a GUI dialog box. On the left, the text 'Tag to Extend' is displayed in a light blue font. To its right is a dropdown menu with a white background and a thin border. The menu is currently open, showing the selected item 'lk_linux-on-fletch' in black text. A small downward-pointing arrow is visible on the right side of the dropdown box.

Click **Next** to proceed to the next dialog box.

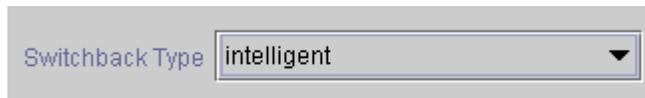
4. Select the **Target Server** where you are extending your Oracle resource hierarchy. The drop down box provides the names of the servers in your cluster that are not already in the selected hierarchy.



A screenshot of a GUI dialog box. On the left, the text 'Target Server' is displayed in a light blue font. To its right is a dropdown menu with a white background and a thin border. The menu is currently open, showing the selected item 'cornfed' in black text. A small downward-pointing arrow is visible on the right side of the dropdown box.

Click **Next** to proceed to the next dialog box.

5. Select the **Switchback Type**. This dictates how the Oracle instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.



A screenshot of a GUI dialog box. On the left, the text 'Switchback Type' is displayed in a light blue font. To its right is a dropdown menu with a white background and a thin border. The menu is currently open, showing the selected item 'intelligent' in black text. A small downward-pointing arrow is visible on the right side of the dropdown box.

The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box.

Click **Next** to proceed to the next dialog box.

6. Select or enter a **Template Priority**. This is the priority for the Oracle hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.
7. Select or enter the **Target Priority**. This is the priority for the new extended Informix hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.



A screenshot of a GUI dialog box. On the left, the text 'Target Priority' is displayed in a light blue font. To its right is a text input field with a white background and a thin border. The field contains the number '10' in black text. A small downward-pointing arrow is visible on the right side of the input field.

Click **Next**.

8. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this Oracle resource have been met. If there were some requirements that had not been met, LifeKeeper would not allow you to select the **Next** button, and the **Back** button would be enabled.

```

Executing the pre-extend script..
Checking existence of extend and canextend scripts
Building independent resource list
Checking extendability for lk_linux-on-fleth
Checking extendability for filesys29111

Pre Extend checks were successful

```

If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box.

If you click **Cancel** now, you will need to come back and extend your Oracle resource hierarchy to another server at some other time to put it under LifeKeeper protection.

When you click **Next**, LifeKeeper will launch you into the Extend Resource Hierarchy configuration task.

9. The next dialog box is an “information only” box displaying the **ORACLE_SID** tag name. You can not change this designation.

ORACLE_SID (information only)

Click **Next**.

10. Select or enter the **ORACLE_HOME for Target Server**. This is the directory location where the Oracle application is located on the backup or target server.

ORACLE_HOME for Targer Server

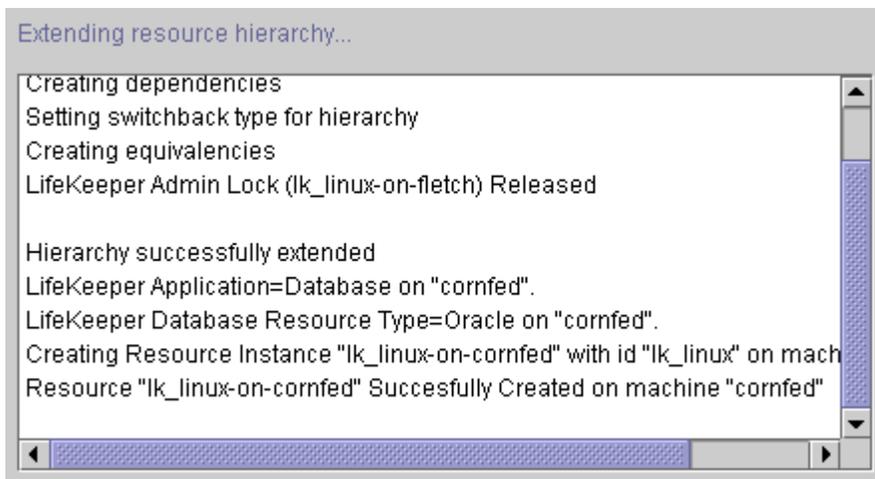
Click **Next**.

11. Select or enter the **Database Tag**. The default tag is SID-on-server.

Database Tag

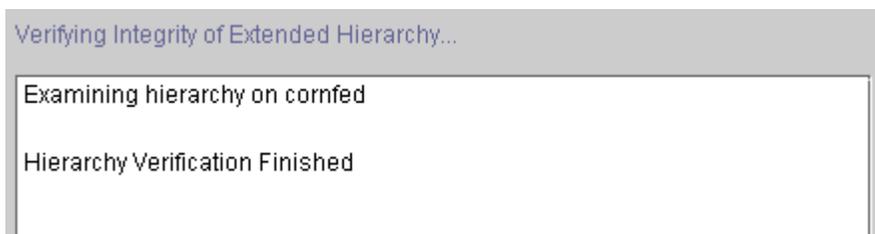
12. Click **Extend**.

13. An information box will appear verifying that the extension is being performed.



Click **Next Server** if you want to extend the same Oracle resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation.

If you click **Finish**, another dialog box will appear confirming LifeKeeper has successfully extended your Oracle resource.



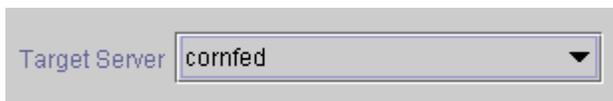
12. Click **Done** to exit from the Extend Resources Hierarchy menu selection.

Note: Be sure to test the functionality of the new instance on *both* servers.

Unextending Your Hierarchy

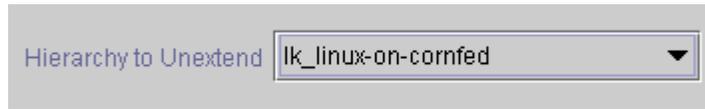
1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the Oracle resource. It cannot be the server where Oracle is currently in service.

Note: If you selected the Unextend task by right clicking from the right pane on an individual resource instance this dialog box will not appear.



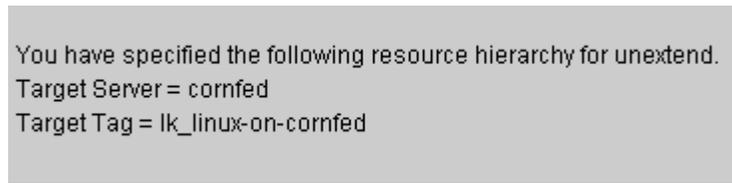
Click **Next** to proceed to the next dialog box.

3. Select the Oracle **Hierarchy to Unextend**. **Note:** If you selected the Unextend task by right clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.



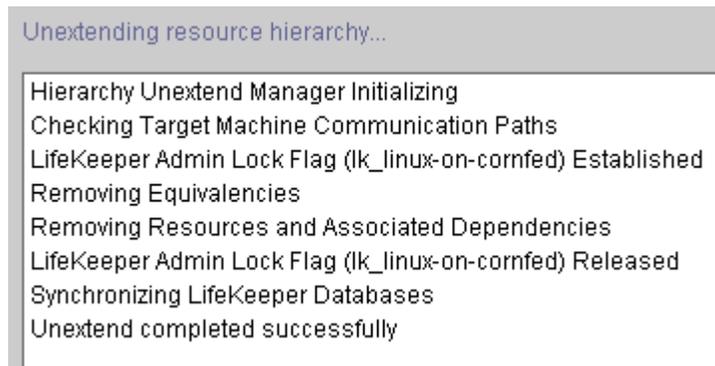
Click **Next** to proceed to the next dialog box.

4. An information box appears confirming the target server and the Oracle resource hierarchy you have chosen to unextend.



Click **Unextend**.

5. Another information box appears confirming that the Oracle resource was unextended successfully.



6. Click **Done** to exit out of the Unextend Resource Hierarchy menu selection.

Change Username / Password for the Oracle Database Account

After a hierarchy has been created, change the Username and Password using one of the following procedures.

If \$ORACLE_HOME is on shared (or replicated) storage (common in active-passive configurations):

1. On the system where the Oracle database resource is in service, edit the LifeKeeper configuration file `/etc/default/LifeKeeper` and add the following line to the file:

```
LK_ORA_NICE=1
```
2. Use `sqlplus` (and the `password` command) to change the Oracle user's password.

```
$ sqlplus lkdba/old-password
SQL> password
```

3. From the LifeKeeper GUI, right-click on the Oracle Database resource hierarchy, then select **Change Username / Password**.
4. Input **New Username** and select **Next**.
5. Input **Password** and select **Next**. Username and Password will be updated after validating.
6. Select **Done**.
7. Once the password change is completed, edit the LifeKeeper configuration file and make the following modification:

```
LK_ORA_NICE=0
```

If \$ORACLE_HOME is on local storage and each node in the cluster has its own copy of \$ORACLE_HOME (common in active-active configurations):

1. On the system where the Oracle database resource is operational, edit the LifeKeeper configuration file /etc/default/LifeKeeper and add the following line to the file:

```
LK_ORA_NICE=1
```

Do exactly the same on each system in the cluster that has the Oracle resource defined.

2. Use sqlplus (and the password command) to change the Oracle user's password on the primary system.

```
$ sqlplus lkdba/old-password
SQL> password
```

3. From the LifeKeeper GUI, right-click on the Oracle Database resource hierarchy, then select **Change Username / Password**.
4. Input **New Username** and select **Next**.
5. Input **Password** and select **Next**. Username and Password will be updated after validating.
6. Select **Done**.
7. Put the Oracle database resource “In Service” on one of the backup systems.
8. Once the database is running on the backup system, use sqlplus to change the Oracle account password.

```
$ sqlplus lkdba/new-password
SQL> password
```

The sqlplus “password” command will prompt you for the “Old Password” and the “New Password.” When making this password change, use the new password that was set in Step 2 as the old password and as the new password. This process resets the security tokens in \$ORACLE_HOME.

9. Put the database “In Service” on each node in the cluster and repeat Step 8.
10. Once the passwords have been changed on all cluster nodes, put the Oracle database back “In Service” on the desired node.
11. Edit the LifeKeeper configuration file on all cluster nodes and make the following change:

```
LK_ORA_NICE=0
```

Testing Your Resource Hierarchy

You can test your Oracle resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, then **Resource**, then finally **In Service** from the drop down menu. For example, an *in service* request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the application is taken out of service without bringing it in service on the other server.

Recovery Operations

When the primary server fails, the Oracle Recovery Kit software performs the following tasks:

- Brings Oracle into service on the backup server by bringing *in-service* the logical interface on one of that server's physical network interfaces. (Note: This occurs only when there is an IP resource instance defined as a dependency of the Oracle hierarchy.)
- Mounts the file system on the shared disk on that server.
- Starts the daemon processes related to Oracle.

Since session context is lost following recovery, after the recovery, Oracle users must reconnect using exactly the same procedures they used to connect originally.

Troubleshooting

This section provides a list of messages that you may encounter during the process of creating and extending a LifeKeeper Oracle resource hierarchy, removing and restoring a resource, and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. Other messages from other LifeKeeper scripts and utilities are also possible. In these cases, please refer to the documentation for the specific script or utility. Messages in this section fall under these topics:

- [Hierarchy Creation and Extend](#)
- [Hierarchy Remove, Restore and Recovery](#)
- [Troubleshooting Hints and Tips](#)

Hierarchy Creation and Extend Errors

“An error occurred while creating Oracle database hierarchies.”

LifeKeeper was unable to create the Oracle hierarchy. See the LifeKeeper logs for additional troubleshooting information.

“*ERROR10* Create Oracle Database dependency failure.”

LifeKeeper was unable to create a dependency between the Oracle hierarchy and one of its dependent resources.

“ERROR: Failed to get oracle login for \$DBPATH.”

Unable to determine the Oracle login. Make sure the login exist, and the permissions for the files and the directories are correct.

“ERROR: Database is not running or open for \$SID.”

The database for the specified SID is not running. The database must be running for hierarchy creation to be successful.

“ERROR: Can not determine Oracle dbspaces and logfiles on \$PRIMACH.”

The locations of the Oracle database files can not be determined. Check your Oracle configuration.

“ERROR: Oracle Database Instance Create on machine \$PRIMACH failed.”

LifeKeeper was unable to create the Oracle resource hierarchy. See the LifeKeeper log file for additional troubleshooting information.

“ERROR: Oracle Database Instance set switchback type on failure.”

LifeKeeper was unable to set the switchback type for the resource. Try to set it manually.

“ERROR(10): Create Oracle Database hierarchy failure.”

LifeKeeper was unable to create the database hierarchy. See the LifeKeeper logs for additional trouble shooting information.

“*ERROR<xx>* Create Oracle Database file system hierarchy failure.” (where xx is the error number)

LifeKeeper was unable to protect the file system that the Oracle database depends on.

“ERROR: No oratab file exists.”

The */etc/oratab* file does not exist. See the Oracle configuration guide.

“An error occurred during creation of LifeKeeper Application=Database on \$MACH.”

LifeKeeper was unable to create the application type ‘Database.’ See the LifeKeeper logs for additional troubleshooting information.

“An error occurred during creation of LifeKeeper Resource Type=Oracle on \$MACH.”

LifeKeeper was unable to create the resource type ‘Oracle.’ See the LifeKeeper logs for additional troubleshooting information.

“ERROR: There are no mounted file systems on shared disks.”

There are no mounted shared disk file systems. See a Linux system administration manual.

“ERROR: Oracle Database file \$DB_file is not on a shared filesystem.”

The Oracle database files are not on a shared disk. The files must be on a shared disk to be protected by LifeKeeper.

“ERROR: Oracle Database file(s) can not be determined.”

LifeKeeper can not determine the location of the Oracle database files. Please check the Oracle database configuration.

“ERROR: Can not determine Oracle dbspaces/chunks (shared) on \$PRIMACH.”

LifeKeeper can not determine the location of the Oracle database files. Please check the Oracle database configuration.

“ERROR: No other LifeKeeper system(s) available for designated (fs/dev) \$FSNAME!”

The file system that Oracle depends on is not shared by another system running LifeKeeper. Make sure the secondary system is running LifeKeeper.

“ERROR: \$ORACLE_SID not found in oratab file on \$SYSTEM.”

The SID specified is not found in the */etc/oratab* file. Please check the Oracle configuration.

“ERROR: LifeKeeper Oracle Database Instance already exists for \$\$SID.”

The SID specified is already protected by LifeKeeper.

“ERROR: STAG already exists on machine \$MACH.”

The specified tag is not unique.

“ERROR: Directory path: ORACLE_HOME does not exist on \$PriMach.”

The ORACLE_HOME directory specified does not exist on the system. Enter the correct ORACLE_HOME path.

“ERROR: “dbs” subdirectory of ORACLE_HOME on \$PriMach does not contain Oracle init\$\$SID.ora.”

There are two possible causes:

- The parameter file does not exist for the SID specified. Check the Oracle configuration.
- The parameter file could not be found in the default location. Please create a link from the dbs directory to the parameter file.

“ERROR: \$COMMAND failed to connect to Oracle Database.”

A connection to the database could not be established. Check the Oracle configuration.

“ERROR: ORACLE_HOME on \$PriMach does not contain Oracle Database commands.”

The ORACLE_HOME entered is invalid. Please enter the correct path.

“ERROR: ORACLE_HOME on \$SecMach does not contain Oracle DBA command.”

The ORACLE_HOME entered is invalid. Please enter the correct path.

“ERROR: “bin” subdirectory of ORACLE_HOME on \$SecMach does not exist.”

The ORACLE_HOME entered is invalid. Please enter the correct path.

“ERROR: Oracle sid = \$\$SID: svrmgrl command not in subdirectory “bin” of ORACLE_HOME.”

The ORACLE_HOME entered is invalid. Please enter the correct path.

“ERROR: Machine \$SECMACH is not on the list of supported systems.”

The system specified is not support. See the LifeKeeper release notes.

“ERROR: Invalid system for \$FSNAME which is designated as shared on \$PRIMACH!”

The shared file system is not shared with the server name entered. Please enter the correct server name.

“ERROR: Invalid system for \$DEV which is designated as shared on \$PRIMACH!”

The shared file system is not shared with the server name entered. Please enter the correct server name.

“ERROR: The \$CHUNKPATH is an Unknown file system or disk sector type for LifeKeeper.”

The database files are on an unsupported file system or disk sector. See the LifeKeeper release notes.

```
“ERROR: Oracle database {SID} startup FAILED
SQL> {SID}078: failure in processing system parameters
LRM-00109: could not open parameter file
'/opt/Oracle11gR2/product/11.2.0/dbhome_1/dbs/init{SID}.ora'
SQL> Disconnected”
```

For example, with an Oracle SID of ORA01, the message might say:

```
ERROR: Oracle database ORA01 startup FAILED
SQL> ORA-01078: failure in processing system parameters
LRM-00109: could not open parameter file
'/opt/Oracle11gR2/product/11.2.0/dbhome_1/dbs/initORA01.ora'
SQL> Disconnected
```

This issue can occur when installing Oracle 11gR2 binary and program files (\$ORACLE_BASE) on non-shared storage.

Copy \$ORACLE_BASE/admin from the primary system where the database instance was created to the backup system (where the hierarchy was extended to) \$ORACLE_BASE/admin. Also change ownership of this directory to oracle:oinstall.

Also copy all *{\$ORACLE_SID}* (ORA01 in this example) files from the primary system in \$ORACLE_BASE/product/11.2.0/dbhome_1/dbs to ORACLE_BASE/product/11.2.0/dbhome_1/dbs on the backup system.

For example, these were the files that were copied from a primary system to the backup, and the ORACLE SID was ORA01.

```
-rw-r----- 1 oracle oinstall 1536 2010-09-08 18:25 orapwORA01
-rw-r----- 1 oracle oinstall  24 2010-09-08 18:25 lkORA01
-rw-r----- 1 oracle oinstall 2560 2010-09-08 18:30 spfileORA01.ora
-rw-r----- 1 oracle oinstall 1544 2010-09-08 18:30 hc_ORA01.dat
```

and a directory

```
peshm_ORA01_0/:
```

Hierarchy Restore and Remove Errors

Bringing an Oracle Resource In Service (Restore)

“LifeKeeper: *ERROR* xxx \$ORACLE_SID **REQUIRES ORACLE ADMINISTRATION” (where xxx is ORA- and some number between 0 and 9, for example ORA-5)**

The Oracle configuration tasks for the SID have not been completed. See the LifeKeeper error log for additional troubleshooting information.

“LifeKeeper: *ERROR* Oracle database \$ORACLE_SID startup FAILED.”

LifeKeeper was unable to start the Oracle instance for the SID. See the LifeKeeper error log for additional troubleshooting information.

Taking an Oracle Resource Out of Service (Remove)

“*ERROR* Oracle database \$ORACLE_SID failed to shutdown.”

Unable to stop the Oracle instance for the SID. See the LifeKeeper error log for additional troubleshooting information.

Maximum number of connections exhausted

Symptom: LifeKeeper quickCheck detects the database instance is down while clients are actively connected.

Cause: All available connections to the Oracle database instance have been exhausted and quickCheck is unable to query the database to determine its state. This results in the quickCheck issuing a local recovery.

Solution: Determine the current setting of “processes” and “sessions” available for Oracle and increase the number of both.

Troubleshooting Hints and Tips

Control File Switchover Failure

If the \$ORACLE_HOME directory does not recover, the database control files may not be set up properly. For automatic switchover, the control files need to be configured on a shared device during the database creation. If you keep the control files on separate servers, you must manually update both servers when you need to implement changes.

Truncated Output

Some versions of Oracle truncate the output when executing the *show parameters control_files* in **sqldba** mode. If your version of Oracle exhibits this behavior, verify the following:

- **controlfile parameter.** Verify that the *controlfile* parameter resides in the *\$Oracle_HOME/dbs/init#SID.ora* file.
- **controlfile devices.** Verify that the *controlfile* devices are on a continuous line, with no new lines, and with each device being separated by a comma.

If an Oracle-related device does not configure properly, then the device can be configured manually using the file system applications available under LifeKeeper Application management.

Flash Recovery Destination Located on a Shared Drive

As noted in the configuration section of this document, it is important that the Flash Recovery destination is located on a shared drive. To see where Oracle believes the Flash Recovery Area is, issue the following query (as SYSDBA):

```
SQL> SELECT substr(Name,1,30) Name,
           (SPACE_LIMIT/1024/1024/1024) Space_Limit_GB,
           SPACE_USED/1024/1024/1024 Space_Used_GB,
           SPACE_RECLAIMABLE, NUMBER_OF_FILES
FROM V$RECOVERY_FILE_DEST;
```

NAME	SPACE_LIMIT_GB	SPACE_USED_GB	SPACE_RECLAIMABLE	NUMBER_OF_FILES
/U01/flash_recovery_area	3.76171875	.156448364		0

Following is an example of how to make a change in *\$ORACLE_HOME/dbs/spfile<sid>* to complete this task:

```
SQL> ALTER SYSTEM SET
DB_RECOVERY_FILE_DEST='/oracledb/oracle/flash_recovery_area' scope=both;

System altered.

SQL> show parameter DB_RECOVERY_FILE_DEST;
```

NAME	TYPE	VALUE
db_recovery_file_dest	string	/oracledb/oracle/flash_recovery_area
db_recovery_file_dest_size	big integer	2G

```
SQL> commit;
```

Database Creation Problems

Problem: During DataBase creation using **dbca**, the following message is received: “**ORA-00439 feature not enabled: string**”

Action: Check the value of the environment variable \$ORACLE_SID. Make sure that is the same as the SID that is being created.

Problem: During Database creation from scripts, the following message is received: “**ORA-01092 ORACLE instance terminated. Disconnection forced**”

Action: See the alert in the *bdump* directory. If you see the message “**ORA-12714 invalid national character set specified**” then check the value of the environment variable \$ORA_NLS33. Make sure that it is set to the correct location.

Problem: If you encounter problems creating the database from the script generated from **dbca**, then do the following:

Action: 1. Be sure to create the following directories if they do not already exist:

```

bdump
cdump
udump
<oracle data base directory>/oradata sid
<oracle data base directory>/dbs
<oracle data base directory>/admin/<SID>

```

If you need to determine the path to your *bdump* and *udump* directories, you can look in the initialization file (*init<SID>.ora*).

2. Make sure the file \$ORACLE_HOME/dbs/orapw exists; if not, use the **orapwd** utility to create it.

Database Startup Problems

Problem: During DataBase start-up using **sqlplus**, the following message is received: “**ORA-03113 end-of-file on communication channel**”

Action: Make sure the initialization file (*init<SID>.ora*) and the password file (*orapw<SID>*) are in the directory \$ORACLE_HOME/dbs.

Problem: During Database startup, the following message is received:
“ORA-01092 ORACLE instance terminated. Disconnection forced”

Action: See the alert in the *bdump* directory. If you see the message
“ORA-12701 CREATE DATABASE character set is not known ” then check the value of the environment variable
 \$ORA_NLS33. Make sure that it is set to the correct location.

inqfail error in the LifeKeeper Log

If a inqfail error similar to the example below appears in your LifeKeeper error log following a failover, you will need to change the filesystemio setting.

Note: The disk id and server name will be different for each configuration.

```
Fri Apr 18 00:10:44 EDT 2008 quickCheck: 001001: Calling sendevent for
resource "ppath8843" on server "satlcallcldb006"
RECOVERY class=disk event=inqfail name=60060160A00D0000359F89C0BF1AD811
```

To resolve this problem, you will need to change the setting filesystemio="SETALL" to
 filesystemio="ASYNCH"

To locate this setting, query the option with the following SQL command:

1. SQL> show parameter filesystemio;
2. Use the following commands to change the settings:


```
SQL> alter system set filesystemio_options=<XXXXXXXX> scope=spfile;
<XXXXXXXX> can be set to
<XXXXXXXX> = {none | setall | direction | asynch}
NONE - no optimization
ASYNCH - enable asynchronous I/O
DIRECTIO – enable direct I/O
SETALL – enables all available features
```

IMPORTANT: Oracle needs to be restarted after resetting the parameter.

Appendix

Setting up Oracle to use Raw I/O

Use the following steps to create an Oracle database that uses shared Raw I/O devices instead of files.

1. Determine the minimum number and sizes of files that you will need to create your database, including control files, tablespaces and redologs. You *can* create a mixed setup with some of those items as files and others on Raw I/O devices. All of the Raw I/O devices must use shared disk partitions.
2. Create a Raw I/O setup with the necessary number of Raw I/O devices.
 - a. Create the raw devices with the same size or larger than you are going to specify for the Oracle database creation.
 - b. Create raw device mappings in the system initialization file(i.e. *boot.local* or *rc.local*) using the **raw** command. You should add meaningful comments to identify which raw device represents which Oracle file. This is done so that the mapping can be re-established in the case of a re-boot of the system. **These mappings should be removed from the file manually once the Raw I/O device is under LifeKeeper protection.**
3. Make the raw devices writable for the Oracle database using the following command:
chown oracle:dba /dev/raw[0-9]*
4. Activate the raw device settings by executing the file that contains the mappings.
5. If you already have a database creation script, go directly to step 6. If not, you may use one of the Oracle Java GUI tools, **dbassist** or **dbca**, to generate your database creation scripts. Using either tool, you must choose to “Save As Script”. Do not choose to create the database.

Notes:

- In **dbca** the “New Database” template must be selected to generate scripts. Change filenames to shared devices and adjust the values for your configuration if necessary.
 - The DB creation process should not be started at this point! The **dbassist** tool checks to see if the file specified for each tablespace already exists, and will not proceed if it does. The **dbca** tool prompts to confirm that it will overwrite the files but fails on raw devices. In either case, you are unable to use raw devices directly from these tools.
6. The database creation scripts (either the existing ones or those created by **dbassist** or **dbca**) must be edited. The desired filename (including the path) must be replaced with the full pathname of the Raw I/O device. The affected files should include (at minimum) the files database creation file (for the **CREATE DATABASE** command) and tablespace creation file (for the **CREATE TABLESPACE** command). Depending on what options you selected in **dbassist** or **dbca** there may more files to be edited. Also, edit the initialization file to change the control files to Raw I/O devices, if desired. The initialization file is located in the directory with the creation script. The result looks like this for the data file:

```
...
CREATE DATABASE "LK"
  maxdatafiles 254
  maxinstances 8
```

```

maxlogfiles 32
character set US7ASCII
national character set US7ASCII
DATAFILE '/dev/raw/raw1' SIZE 260M AUTOEXTEND ON NEXT 10240K
logfile '/ora/LK/redo01.log' SIZE 500K,
        '/ora/LK/redo02.log' SIZE 500K,
        '/ora/LK/redo03.log' SIZE 500K;
...

```

The Raw I/O device must be the minimum size required by Oracle for the data that will be stored.

7. Now create the database by running the script that you created in step #5.
8. Be sure to check the create log for any database or tablespace errors that may have occurred.
9. If you have trouble creating the database with the creation scripts, or you want to add tablespaces on raw devices later, you must create the database with the applicable tool (i.e. **dbassist** or **dbca**). Then, add the Raw I/O device data files by executing a command similar to the following from the **sql** utility:

```

tablespace RAWTS
DATAFILE '/dev/raw/raw217' SIZE 50M REUSE
DEFAULT STORAGE (INITIAL 50K NEXT 50K
MINEXTENTS 1 MAXEXTENTS 4) ONLINE

```

Adding a Tablespace After Creating Hierarchy

If a tablespace is added on a Raw I/O device *after* the Oracle hierarchy has been created in LifeKeeper, you must create a LifeKeeper Raw I/O hierarchy via the GUI and manually create a dependency between the Oracle resource (as parent) and the Raw I/O resource (as child).

Creating a Shared Oracle Listener for Multiple Resources

You may want to create a shared Oracle Listener if any of the following statements are true for your system configuration:

- Multiple Listeners are defined for Multiple Oracle SIDs
- The Oracle Listener is a critical component in your configuration
- A Single Listener is defined for Multiple Oracle SIDs

This process will allow protection of listener(s) within LifeKeeper to accommodate various listener(s) and SIDs combinations.

If you are creating a shared Listener for multiple resources, follow these procedures.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.

Important The Oracle Application must be running when you create the resource.

2. A dialog box will appear with a drop down list box with all recognized Recovery Kits installed within the cluster. Select **Oracle Database Listener** from the drop down listing. Click **Next** to proceed to the next dialog box.
3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information. If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

Field	Tips
Switchback Type	Choose either <i>intelligent</i> or <i>automatic</i> . This dictates how the Listener resource will be switched back to this server when the server comes back up after a failover. If using data replication, choose <i>intelligent</i> as the switchback type. Note: The switchback type must match that of the dependent resources (IP and volume resources) used by the Listener resource, or else the create will fail.
Server	Select the server on which you want to create the hierarchy.
Listener Configuration File Path	Select the full path to the Oracle listener configuration file.
Listener Names(s)	Select the name(s) of the Oracle Listener(s) to provide protection for with this resource instance.
Listener Executable(s)	Select the path to the Oracle listener executable. The listener executable is required to start, stop, monitor and recover the specified Oracle listener(s).
Listener Protection Level	Select one of the following levels: Full Control (Start, Stop, Monitor and Recover) Intermediate Control (Start, Monitor and Recover) Minimal Control (Start and Monitor Only)
Listener Recovery Level	Select the level of recovery for the specified listener: Standard (On) – Enable standard LifeKeeper recovery. If Standard (On) is selected, all listener failures will be tried locally, and if necessary, trigger a failover to an available backup server. Optional (off) – Enable optional LifeKeeper recovery. If Optional (Off) is selected, all listener failures will be tried locally, but will not cause a failover to an available backup server.
IP Address	Select the IP Address resource name that will be protected as

Field	Tips
Name(s)	dependents of this resource hierarchy. IP Address associated with the selected listener(s) are displayed in the choice list. Select None if no IP resources are required for this configuration,
Listener Tag	Enter a unique name for the resource on the server. The valid characters allowed for the name are letters, digits, and the following special characters: - _ . /

- Select the **Create** button to start the hierarchy creation. An information box appears and LifeKeeper will validate that you have provided valid data to create your database listener resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created.
- The Pre-Extend Wizard dialog will appear stating that you have successfully created the resource hierarchy and you will be prompted to select the following information. If you are unfamiliar with the Extend operation, click **Next** after making a selection in each dialog box. If you are familiar with the LifeKeeper **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

Field	Tips
Target Server	Select a Target Server to which the hierarchy will be extended. If you select Cancel before extending the resource hierarchy to at least one other server, LifeKeeper will provide no protection for the applications in the hierarchy.
Switchback Type	This dictates how the Oracle Listener instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. Note: Remember that the switchback strategy must match that of the dependent resources to be used by the Oracle Listener resource.
Template Priority	This field appears only if you did NOT extend directly from the Create function.) Enter a number between 1 and 999 to specify the template server's priority in the cascading failover sequence for this resource. A lower number means a higher priority. LifeKeeper assigns the number "1" to the server on which the hierarchy was created. No two servers can have the same priority for a given resource.
Target Priority	Enter a number between 1 and 999 to specify the target server's priority in the cascading failover sequence for this resource. A lower number means a higher priority. LifeKeeper offers a default of 10 for the first server to which a hierarchy is extended.

- After receiving the message that the pre-extend checks were successful, click **Next** and enter the following information.

Field	Tips
Listener Configuration File Path	Select the full path to the Oracle Listener configuration file.
Listener Executable(s) Path	Select the path to the Oracle Listener executables. The listener executables are required to start, stop, monitor and recover the specified Oracle listener(s).
Listener Tag	This field is automatically populated with a unique name for the new Oracle Listener resource instance on the primary server. The default naming pattern will be displayed for you. You may type in another unique name. The valid characters allowed for the Listener tag are letters, digits and the following special characters - : . /

- Click **Extend**. The Hierarchy Integrity Verification window displays with the following message, Hierarchy Verification Finished. Click **Next Server** or **Finish**.

Updating the Listener Protection Level

- Select a resource and then the  button from the Resource toolbar to update the protection level of the resource.
- Enter the following information.

Field	Tips
Listener Protection Level	Select one of the following: Full Control (Start, Stop, Monitor and Recover) Intermediate Control (Start, Monitor and Recover) Minimal Control (Start and Monitor Only)

- Click **Update** to change the Protection Level from the current state to the new state. Select **Cancel** to leave the value unchanged.

Updating the Listener Recovery Level

- Select a listener and then the  button from the Resource toolbar to update the recovery level of the resource.

2. Enter the following information.

Field	Tips
Listener Recovery Level	<p>Select the level of recovery for the specified listener(s).</p> <p>Standard (On_ – enables a standard LifeKeeper recovery. If Standard, (On) is selected, all listener failures will be tried locally and if necessary trigger a fail over to an available backup server.</p> <p>Optional (Off) – enables optional LifeKeeper recovery. If Optional (Off) is selected, all listener failures will be tried locally, but will not cause a fail over to an available backup server.</p>
Update Confirmation	<p>Select the Update button to change the Recovery Level from the current state to the new state.</p> <p>Select Cancel to leave the current value unchanged.</p>

3. Click **Update** to change the Recovery Level from the current state to the new state. Select **Cancel** to leave the current value unchanged.

Updating the Protected Listener(s)

1. Select a listener and then the  button from the Resource toolbar to update your protected listener(s).
2. Enter the following information.

Field	Tips
Listener Name(s)	Select the name or names of the Oracle Listener(s) to provide protection for with this resource instance.
IP Address Name(s)	<p>Select the IP Address resource name(s) that will be added as a dependent of this resource hierarchy.</p> <p>Select None if no additional IP Resources are required for this configuration.</p>

3. Click **Update** to change the Protected Listener(s) and IP assignment from the current state to the new state. Select **Cancel** to leave the current value unchanged.

If you select update, a dialog displays stating that the Protected Listeners for the specific resource is being updated. Click **Finish**.