Oracle® Database Appliance

Administration and Reference Guide

Release 12.1.2.2.0 for Linux x86-64

E55580-06

February 2015

ORACLE®

Oracle Database Appliance Administration and Reference Guide, Release 12.1.2.2.0 for Linux x86-64

E55580-06

Primary Authors: David Austin, Douglas Williams, Virginia Beecher

Contributors: Madhusudhan Achar, Aiman Al-Khammash, Mark Bauer, Michael Bechler, Tammy Bednar, Ruggero Citton, Jonathan Creighton, Kevin Deihl, Yingwei Hu, Winston Huang, Sarat Kakarla, Gabriele Knauth, Erich Kreisler, Bill Manry, Mary Martyak, Anil Nair, Josh Ort, Ram Pandarapattathil, Angelo Pruscino, Fenando Betanzo Sanchez, Mark Scardina, Ravi Sharma, Richard Shinkle, Sandeep Singla, Janel Stern, Srinivasan Subramaniam, Su Tang, Krishnadev Telikicherla, Bob Thome, Paul Tsien, Ralph Woodley, Yusuke Yamamoto

Contents

4 Validating and Troubleshooting Oracle Database Appliance

5 Oracle Appliance Manager Command-Line Interface

Preface

Oracle Database Appliance is an optimized, prebuilt and ready-to-use clustered database system that is easy to deploy, operate, and manage. By integrating hardware and software, Oracle Database Appliance eliminates the complexities of nonintegrated, manually assembled solutions. Oracle Database Appliance reduces the installation and software deployment times from weeks or months to just a few hours while preventing configuration and setup errors that often result in suboptimal, hard-to-manage database environments.

Audience

This guide is intended for anyone who configures, maintains, or uses Oracle Database Appliance. This includes:

- system administrators
- network administrators
- database administrators
- application administrators and users

This book does not include information about Oracle Database architecture, tools, management, or application development that is covered in the main body of Oracle Documentation unless there are features specific to Oracle Database Appliance. Users of Oracle Database Appliance software are expected to have the same skills as users of any other Linux-based Oracle Database and Oracle Real Application Clusters installations.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

The following documents are published in the Oracle Database Appliance online documentation library, available at http://www.oracle.com/goto/oda/docs:

- *Oracle Database Appliance Setup Poster* (a full-size printed copy ships with Oracle Database Appliance)
- *Oracle Database Appliance Getting Started Guide*
- *Oracle Database Appliance Release Notes*
- *Oracle Database Appliance Owner's Guide*
- *Oracle Database Appliance Service Manual*
- *Oracle Database Appliance Safety and Compliance Guide*
- *Oracle Database Appliance Licensing Information*
- *Oracle Database Appliance Security Guide*

For more information about using Oracle Database, see the following documents:

- *Oracle Database Concepts*
- *Oracle Database Administrator's Guide*
- *Oracle Database SQL Language Quick Reference*
- *Oracle Database Reference*
- *Oracle Database 2 Day + Real Application Clusters Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Clusterware Administration and Deployment Guide*

For more details about other Oracle products that are mentioned in Oracle Database Appliance documentation, such as Oracle VM, Oracle Integrated Lights Out Manager, and so on, see the Oracle Documentation home page at the following address:

http://docs.oracle.com

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action or terms defined in the text.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.
#	The pound (#) prompt precedes a Linux command.

1

Updating Oracle Database Appliance Release

To keep Oracle Database Appliance running with the latest release of the software, check for and apply patches when they are released. As well as offering new features, patches may improve the functionality of existing features.

These three sections explain the patch process as well as how to find and apply patches:

- About Oracle Database Appliance Server Patches
- Checking for Patches on My Oracle Support
- Patching Oracle Database Appliance Software with Oracle Appliance Manager

About Oracle Database Appliance Server Patches

All patching of Oracle Database Appliance is done with the regular Oracle Database Appliance Patch Bundle. The Patch Bundle provides all relevant patches for the entire system, including:

- BIOS
- Hardware drivers
- Oracle Linux
- Oracle ILOM
- Oracle Database clone binaries
- Oracle Grid Infrastructure clone binaries

When the regular Oracle Database Appliance Patch Bundle is available, log on to My Oracle Support as the registered Oracle Database Appliance software owner. Follow the instructions in My Oracle Support note 888888.1 as well as the instructions in the patch README file for information about patching the system.

Infrastructure patches (OS, Firmware, Oracle ILOM, and so on) will require a short downtime of Oracle Database Appliance while the patch is being applied. Oracle Appliance Manager verifies that a patch meets all prerequisite requirements to prevent you from installing patches in the wrong order. For example, you cannot just patch the GI without first updating the infrastructure.

Caution: You must use an Oracle Database Appliance patch bundle or SAP patch bundle (downloaded from the SAP Service Marketplace) to patch Oracle Database Appliance. Do not use individual patches for Oracle Grid Infrastructure, Oracle Database patches, or Linux. If you use patches that are not intended for Oracle Database Appliance, or if you use Opatch or a similar patching tool, then Oracle Database Appliance inventory is not updated, and you cannot complete future patch updates.

Checking for Patches on My Oracle Support

Oracle Database Appliance Patch Bundles are released on a quarterly schedule. My Oracle Support note 888888.1 provides information about the latest Oracle Database Appliance patch bundle.

To check for patches that apply to your system:

1. Log on to My Oracle Support at:

   `https://support.oracle.com`

2. Search for note 888888.1.
3. If patches are available for your software, follow the directions in note 888888.1 to download and apply them.

Patching Oracle Database Appliance Software with Oracle Appliance Manager

The following steps and example outline the general procedure to patch the software on Oracle Database Appliance. Read the specific patch Readme and any help information for details on how to apply each particular patch.

1. Log in to My Oracle Support using the Support Identifier of the registered software owner of Oracle Database Appliance.

   `https://support.oracle.com`

2. Check the information in My Oracle Support note 888888.1 to identify, download, and apply the patch on your system. The patching procedure for a given patch bundle might differ from the standard procedure.

   Under Patch Search, select **Oracle Database Appliance** from the Product list and the patch release number from the Release list. Click **Search**. Then select the patch or patches and click **Download**.

3. Log in as `root`.
4. Move the patch to a temporary directory (for example, to `/tmp`) on each node of Oracle Database Appliance.
5. To prepare the patch for installation, unpack the patch files on each node with the `oakcli unpack -package` command. Use the following command syntax, where *`path`* is the absolute path to the patch file:

   ```
   # oakcli unpack -package path
   ```

   For example, to prepare the patch files for Oracle Database Appliance release 12.1.2.2.0, after copying the patch files (`p20340774_121220_Linux-x86-64_`

1of2.zip and p20340774_121220_Linux-x86-64_2of2.zip) into the /tmp, directory on each node, run the following oakcli unpack -package commands:

On Node 0:

```
# oakcli unpack -package /tmp/p20340774_121220_Linux-x86-64_1of2.zip
# oakcli unpack -package /tmp/p20340774_121220_Linux-x86-64_2of2.zip
```

On Node 1:

```
# oakcli unpack -package /tmp/p20340774_121220_Linux-x86-64_1of2.zip
# oakcli unpack -package /tmp/p20340774_121220_Linux-x86-64_2of2.zip
```

6. Before installing the patch, Oracle recommends that you run the oakcli validate -c ospatch -ver *patch_version* command on Node 0. For example, you would use the following command to check for possible problems with the 12.1.2.2.0 patch:

```
# oakcli validate -c ospatch -ver 12.1.2.2.0
```

The output will inform you whether the patch will succeed. If the output indicates a possible problem, for example, an unpacked version of the patch is not available on the system, you may wish to defer applying the patch until you have downtime to fix the expected problem.

7. To install the patch, run the oakcli update -patch command on Node 0. Use the following command syntax, where *version* is the patch update version number:

```
# oakcli update -patch version
```

For example, to update to patch 12.1.2.2.0, run the oakcli update -patch command on Node 0 as follows:

```
# oakcli update -patch 12.1.2.2.0
```

Most new patches, particularly Oracle Database and Oracle Grid Infrastructure patches, will install themselves on both nodes automatically. The output displayed by the patch process advises you if the patch is being installed on one of both nodes. For older patches that only install on a single node, repeat the oakcli update -patch command on the second node.

2

Managing Oracle Database on Oracle Database Appliance

This chapter describes how to perform management tasks for Oracle Databases on Oracle Database Appliance. The chapter covers the following tasks:

- Managing and Maintaining Oracle Database
- Creating and Converting Databases
- Managing Multiple Databases on Oracle Database Appliance
- Using Oracle Database Appliance SSDs
- Upgrading Oracle Database on Oracle Database Appliance

Managing and Maintaining Oracle Database

Many tasks related to managing Oracle Databases, described in the Oracle Database documents on the Oracle Help Center page at `http://docs.oracle.com`, are also required with databases on Oracle Database Appliance. However, Oracle Database Appliance provides its own command line tool, Oracle Appliance Kit Command Line Interface, or OAKCLI, to manage all components on the system. This includes commands to create, upgrade, and patch databases; create and upgrade Oracle homes; create and modify database creation parameter files; and so on.

Additionally, because Oracle Database Appliance combines the capabilities of the database administrator role into its `root` user, database creation and related tasks are simplified and should always be performed using OAKCLI. See Chapter 5 for detailed syntax and usage information of all OAKCLI commands.

Oracle Enterprise Manager Database Control

The primary product for managing your database is Oracle Enterprise Manager Database Control (Database Control), a web interface. After you have installed Oracle Database, created or upgraded a database, and configured your network, use Database Control to manage your database. Database Control also provides an interface for performance advisors and for Oracle Database utilities such as SQL*Loader and Oracle Recovery Manager (RMAN).

> **See Also:** *Oracle Database 2 Day DBA* for an introduction to Database Control

Data Migration and Management

If you are loading data or migrating data from an existing database to Oracle Database Appliance, then you can use tools such as SQL*Loader, Oracle Data Pump,

transportable tablespaces, and RMAN. You can also use the RMAN utility to back up and recover databases on Oracle Database Appliance.

> **See Also:** *Oracle Database Backup and Recovery User's Guide* and *Oracle Database Backup and Recovery Reference* for more information about Oracle Database backup and recovery capabilities, *Oracle Database Utilities* for information about data loading, and "Performing Oracle ASM Data Migration with RMAN" in *Oracle Automatic Storage Management Administrator's Guide* for an overview of how to migrate data to Oracle Automatic Storage Management (Oracle ASM).

Oracle Clusterware

Oracle Clusterware provides the cluster technology required for Oracle Real Application Clusters (Oracle RAC). In addition, Oracle Clusterware manages applications and processes as resources that you register with Oracle Clusterware. The number of resources that you register with Oracle Clusterware to manage an application depends on the application. Applications that consist of only one process are usually represented by only one resource. More complex applications, that were built on multiple processes or components, might require multiple resources.

> **See Also:** *Oracle Clusterware Administration and Deployment Guide* for information about making applications highly available with Oracle Clusterware

Oracle RAC One Node

Oracle RAC One Node is a single instance of an Oracle RAC database that runs on one node in a cluster. Instead of stopping and starting instances, you can use the Oracle RAC One Node online database relocation feature to relocate an Oracle RAC One Node instance to another server.

Oracle RAC One Node databases are administered slightly differently from Oracle RAC or single-instance Oracle Databases. For Oracle RAC One Node databases, one node is the primary node, and the other node is a candidate node, which is available to accommodate services if the primary node fails or is shut down for maintenance. The nodes, databases, and database services reside in the generic server pool.

> **See Also:** *Oracle Real Application Clusters Administration and Deployment Guide* for more information about administering Oracle RAC One Node

Oracle Real Application Clusters

Oracle Real Application Clusters (Oracle RAC) provides technology that links two or more individual computers so that they function as one system. Oracle RAC deployed on Oracle Database Appliance enables each node to share access to a database. If one node fails or is taken offline, then the other node continues operating and the entire Oracle RAC database remains available. To applications, each node appears as a single computer.

> **See Also:** *Oracle Real Application Clusters Administration and Deployment Guide* for information about administering Oracle RAC

Oracle Database Appliance currently supports only administrator-managed databases, where the database administrator allocates each instance of the database to a specific node in the cluster. Policy-managed databases, where the database administrator

defines the number of database instances required, but not the nodes where they will run, are not available on Oracle Database Appliance.

When you review the database resource for an administrator-managed database, you see a server pool defined with the same name as the Oracle database. This server pool is part of a special Oracle-defined server pool called Generic. Oracle RAC manages the Generic server pool to support administrator-managed databases. When you add or remove an administrator-managed database using either the Server Control utility (SRVCTL) or Oracle Database Configuration Assistant (DBCA), Oracle RAC creates or removes the server pools that are members of Generic. You cannot use SRVCTL or Oracle Clusterware Control (CRSCTL) utility commands to modify the Generic server pool.

> **See Also:** *Oracle Clusterware Administration and Deployment Guide* and *Oracle Real Application Clusters Administration and Deployment Guide* for more information about Oracle Clusterware and Oracle Real Application Clusters

Administrative Groups and Users

During configuration, two administrative accounts are created for Oracle Database Appliance: the user `grid`, with a user ID (UID) of 1000, and the user `oracle`, with a UID of 1001. The user `grid` is the Oracle Grid Infrastructure installation owner and the user `oracle` is the Oracle Database installation owner and the owner of all Oracle Database homes. By default, these users belong to groups shown in the following table.

Table 2–1 grid and oracle User Group Memberships

Group Name	Group ID (GID)	*grid* is member	*oracle* is a Member
`oinstall`	1001	yes (primary group)	yes (primary group
`dba`	1002	no	yes
`racoper`	1003	yes	yes
`asmdba`	1004	yes	yes
`asmoper`	1005	yes	no
`asmadmin`	1006	yes	no

If you create an initial database during deployment, then the password for the `SYS` and `SYSTEM` users is `welcome1`. You should change this password for both users as soon as possible to prevent unauthorized access to your database using these privileged accounts.

> **See Also:** *Oracle Grid Infrastructure for Linux* for information about operating system privileges groups and *Oracle Automatic Storage Management Administrator's Guide* for information about Oracle ASM system privileges

Creating and Converting Databases

This section describes the following tasks:

- Creating Databases with Oracle Appliance Manager Commands

- Creating Database Configuration Files with Oracle Appliance Manager Commands
- Creating Snapshot Databases
- Converting Single-Instance Databases to Oracle RAC or Oracle RAC One Node

You should use Oracle Appliance Manager commands, described in Chapter 5, "Oracle Appliance Manager Command-Line Interface," to create and manage databases on Oracle Database Appliance. This will help you avoid using commands and parameters that could reduce the functionality of your databases, such as changing parameters associated with database file locations, including `control_files`, `db_create_file_dest`, `db_recovery_file_dest`, and so on.

Creating Databases with Oracle Appliance Manager Commands

Use the `oakcli create database` command to create additional databases on Oracle Database Appliance.

When you run this command, respond to each prompt by entering the number that corresponds with the option you want to apply to your database. When a default is supplied and is the value you want to use, typically shown as option 1, then accept that value by pressing the Enter key. When there are many options, then you might need to press 0 to reveal all of the options if the value you want is not displayed.

See "oakcli create database" on page 5-15 for more information.

Creating Database Configuration Files with Oracle Appliance Manager Commands

Use the `oakcli create db_config_params -conf` *`filename`* command to create a configuration file for configuring multiple databases on Oracle Database Appliance, where *`filename`* is the name of the configuration file that you generate.

When you run this command, respond to each prompt by entering the number that corresponds with the option you want to apply to your database. When a default is supplied and is the value you want to use, typically shown as option 1, then accept that value by pressing the Enter key. When there are many options, then you might need to press 0 to reveal all of the options if the value you want is not displayed.

To see your existing database configuration files, use the `oakcli show db_config_params` command. For example:

```
# oakcli show db_config_params
Available DB configuration files are:
default
largedb
extralargedb
mytest.params
```

Note that only non-default extensions are included in the output, the default extension, `.dbconf`, is not shown.

To use a database configuration file to create a database, or many databases with identical profiles, use the `oakcli create database` command and include the `-params -conf` *`file_name`* option where file_name is the name of the required configuration file.

Remove unwanted database configuration files with the `oakcli delete db_config_params -conf` command, providing your configuration file name as the `-conf` parameter value. As with other Oracle Appliance Manager commands related to

database configuration files, you do not need to include the extension if your file has the default extension value, which is `.dbconf`.

> **See Also:** "oakcli delete db_config_params" on page 5-25 for more information about the command

Creating Snapshot Databases

An Oracle snapshot database is created by taking a snapshot of the ACFS file system where the source datafiles reside. The source database can be single instance, Oracle RAC, or Oracle RAC One Node. Compared to other methods of creating copies of databases, snapshot databases require less time and storage space and involve no downtime of the source database. Additionally, you can create any database type and class from any other type and class. For example, you can create an Oracle RAC database from an Oracle RAC One Node database. Similarly, you can create a database that is different in size than the source database.

On Oracle Database Appliance, you can create snapshot databases from any Oracle Database stored on Oracle ASM Cluster File Systems (ACFS). Beginning with Oracle Database Appliance Release 12.1.2.0.0, this includes any Oracle Database Release 11.2.0.4 or later database created or upgraded on the system. Additional requirements for a database to be used as the source for a snapshot database include:

- must not be a standby or container database
- must not be running in read-only mode, or in restricted mode, or in online backup mode
- must be in ARCHIVELOG mode
- must have all defined datafiles available and online

Also, ensure that the system clocks on the two Oracle Database Appliance nodes are synchronized before creating a snapshot database. To create a snapshot database, use the `oakcli create snapshotdb` command, as shown in the following example, which creates a snapshot database named `snapprod` from the database named `prod`.

```
oakcli create snapshotdb -db snapprod -from prod
```

Before issuing the command, ensure that the system clocks on the two Oracle Database Appliance nodes are synchronized. If the clocks are sufficiently different, the command may fail.

> **Caution:** Oracle Database Appliance does not support centralized wallets with Transparent Data Encryption. Recovery of encrypted data may fail in the snapshot database if the source database relies on an external, centralized wallet.

Converting Single-Instance Databases to Oracle RAC or Oracle RAC One Node

Use the `rconfig` command line utility to convert a single-instance database to either Oracle RAC or Oracle RAC One Node. The contents of a `ConvertToRAC_AdminManaged.xml` file determine the type and other characteristics of the converted database.

See Also: Appendix C, "Converting to Oracle RAC and Oracle RAC One Node from Single-Instance Oracle Databases" in *Oracle Real Application Clusters Installation and Configuration Guide* for instructions to convert a single-instance database.

Managing Multiple Databases on Oracle Database Appliance

An Oracle home is the directory in which you install Oracle Database binaries, and from which Oracle Database runs. Use Oracle Appliance Manager (through `oakcli` commands) to create and manage multiple Oracle homes and databases on Oracle Database Appliance. Oracle Appliance Manager automatically creates an Oracle Database Oracle home that is compliant with the Optimal Flexible Architecture (OFA) standards.

Oracle Database Appliance supports multiple Oracle Homes including different versions for Oracle Database 11g Release 2, 11gR2 and Oracle Database Release 12c Release 1. The exact releases differ from version to version. Check the related Read Me files or Release Notes for specific versions.

See Also: My Oracle Support note 888888.1, at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=888888.1`, for more details about the available database versions.

When you use `oakcli` commands to create multiple homes on Oracle Database Appliance, the commands start the cloning process used by Oracle Grid Infrastructure. In the current release, the user `oracle` owns all of the Oracle homes.

Note: If you are not upgrading from an earlier release, then download the Oracle Database Appliance End-User Bundle for the Oracle Database version that you want to install. See Note 888888.1 for more details:

`https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=888888.1`

Use `oakcli` commands to create, manage, patch, and upgrade multiple databases on Oracle Database Appliance using the multiple Oracle home feature. The `oakcli create database` command enables you to create a database with minimal user input. When used without any additional options, the command creates a new database home. Alternatively, create a database in an existing home by using the `-oh` option. Note that Oracle does not recommend applying RDBMS patches directly, you should only use Oracle Database Appliance Patch Bundles, which are tested to work across the whole software stack.

Note: Use `oakcli` commands to create new databases in either existing Oracle homes or in new Oracle homes.

Managing Multiple Database Instances Using Instance Caging

Oracle Database provides a method for managing CPU allocations on a multi-CPU server that runs multiple database instances. This method is called instance caging. Instance caging and Oracle Database Resource Manager (the Resource Manager) collaborate to support your desired service levels across multiple instances. Consolidation can minimize idle resources, maximize efficiency, and lower costs.

Oracle Database Appliance templates are already tuned for the size of each database instance workload. They are designed to run on a specific number of cores. Instance caging assures that each database workload is restricted to the set of cores allocated by the template, enabling multiple databases to run concurrently with no performance degradation, up to the capacity of Oracle Database Appliance. You could select database template sizes larger than your current needs to provide for planned growth.

Note: Oracle strongly recommends that you use the Oracle Database Appliance templates, because they implement best practices and are configured specifically for Oracle Database Appliance.

Tip: Oracle Appliance Manager configurator refers to the database sizing templates as *classes* of databases.

By default, instance caging is not enabled on Oracle Database Appliance. To enable instance caging, set the initialization parameter, `RESOURCE_MANAGER_PLAN`, for each database on Oracle Database Appliance. The parameter specifies the plan to be used by Resource Manager for the current instance. Setting this parameter will direct Resource Manager to allocate core resources among databases. If no plan is specified with this parameter, then the Resource Manager is not enabled and instance caging will not be enabled.

Instance caging allocation of core resources is enabled in accordance with the Oracle Database Appliance database template size that you select for each database. The `CPU_COUNT` initialization parameter is set in the template. Use the `CPU_COUNT` setting that matches the size of each database to consolidate, and follow the standard instructions for configuring instance caging.

See Also: *Oracle Database Administrator's Guide* for more information about enabling and configuring instance caging and the Resource Manager.

Using Oracle Database Appliance SSDs

Oracle Database Appliance includes SSDs to enhance the performance of certain operations. SSDs are used for:

- Accelerating Redo Log Writes
- Caching Database Data
- Improving I/O Performance for Database Files

Accelerating Redo Log Writes

Oracle Database Appliance contains four dedicated SSDs in slots 20-23 specifically for database redo logs. An ASM diskgroup named +REDO with High Redundancy is provisioned during the deployment process to accelerate database redo log write operations and improve latency. Databases automatically utilize these SSDs, and no other files can be hosted on them.

Caching Database Data

Oracle Database Appliance X5-2 introduces four additional 400 GB SSDs in slot numbers 16-19 that can be used to host database files, or they can be used as a database flash cache in addition to the buffer cache.

An ASM diskgroup named +FLASH with Normal Redundancy is provisioned on these SSDs. All of the storage in the +FLASH diskgroup is allocated to an ASM Dynamic Volume (flashdata), and formatted as an ACFS file system. Storage in this flashdata file system is then made available as an ACFS file system and is used to create database flash cache files that accelerate read operations. The file that contains the flash cache is automatically created for each database and is specified using the database init.ora parameter `db_flash_cache_file`. By default, `flash_cache_file_size` is set to 3 times the size of SGA, up to 196 GB, unless there is not enough space, in which case the size parameter is set to 0. Changing the `flash_cache_file_size` parameter requires restarting the database in order to use the newly sized flash cache.

See Also: *Oracle Database Administrator's Guide* for information about Configuring Database Smart Flash Cache

Improving I/O Performance for Database Files

Oracle Appliance Manager configurator and the `oakcli create database` command provide the option to store entire databases in flash using the flashdata ACFS file system on the 400 GB SSDs (also used for the flash cache).

If there is not enough space available in flash, the `oakcli create database` command will not prompt you with an option to store databases in flash, and the databases files will automatically be created in the +DATA diskgroup. You can also store database data files on both flash and HDDs, but this must be manually managed. It requires a thorough understanding of database usage patterns and is only recommended for advanced administrators.

The limitations of this strategy are:

- Oracle Database versions must be 11.2.0.4 or later
- Database type must be OLTP
- Only non-CDB databases can be completely stored in flash
- There must be 160 GB of available space in the +FLASH diskgroup

See Also:

- "Storage on Oracle Database Appliance" on page A-3 for information about ASM Cluster File System (ACFS) Storage Architecture on Oracle Database Appliance
- "oakcli create database" on page 5-15

Upgrading Oracle Database on Oracle Database Appliance

To upgrade Oracle Database, use the appropriate Oracle Database Appliance Patch Bundle. Typically, you would perform infrastructure patching, then Grid Infrastructure patching, and then the Oracle Database patching. This section contains the instructions for applying each of the available database upgrade patches. These patches perform rolling upgrades that automatically patch Node 1 after patching Node 0. To upgrade Oracle Database, follow the steps in the sections that cover your specific upgrade requirements:

- Upgrading to Oracle Database 11.2.0.3.12 from Previous Oracle Database 11.2.0.3.x Releases
- Upgrading to Oracle Database 11.2.0.3.12 from Oracle Database 11.2.0.2.x Releases

Upgrading to Oracle Database 11.2.0.3.12 from Previous Oracle Database 11.2.0.3.x Releases

1. Run the `oakcli show databases` command, on Node 0 only, to confirm that you have a database with the appropriate release number for this upgrade. The command and output should look similar to the following example.

```
# oakcli show databases
Database Name     Database Type     Database HomeName
    Database HomeLocation                    Database Version
tpcc              RAC               dbhome11203
    /u01/app/oracle/product/11.2.0.3/dbhome_1  11.2.0.3.2(13696216,13696251)
EE                RACOneNode        dbhome11203
    /u01/app/oracle/product/11.2.0.3/dbhome_1  11.2.0.3.2(13696216,13696251)
```

2. Run the `oakcli update -patch 12.1.2.2.0 --database` command on Node 0 only, the command automatically patches both nodes. If you have more than one database home that could be patched, then the software provides a select list from which you pick one, some, or all of the database homes to upgrade.

> **Note:** You cannot perform individual database upgrades from Release 11.2.0.3.x to 11.2.0.3.12. All databases running the homes that you upgrade, such as the two databases listed in the example in Step 1, will be patched to Oracle Database 11.2.0.3.12.

3. After the command completes on both nodes, check your database version with the `oakcli show databases` command on Node 0. The command and output should now look similar to the following example.

```
# oakcli show databases
Database Name     Database Type     Database HomeName
    Database HomeLocation                    Database Version
tpcc              RAC               dbhome11203
    /u01/app/oracle/product/11.2.0.3/dbhome_1  11.2.0.3.12(18522512,17592127)
EE                RACOneNode        dbhome11203
    /u01/app/oracle/product/11.2.0.3/dbhome_1  11.2.0.3.12(18522512,17592127)
```

Upgrading to Oracle Database 11.2.0.3.12 from Oracle Database 11.2.0.2.x Releases

1. Ensure that you have an 11.2.0.3.12 database home by completing the following actions on Node 0:

   - Run the command `oakcli show dbhome`. The output appears similar to the result in the following example.

```
Oracle Home Name       Oracle Home version
                             Home Location
dbhome11202           11.2.0.2.5(13343424,13343447)
                             /u01/app/oracle/product/11.2.0.2/dbhome_1
OraDb112011_home1      11.2.0.3.12(18522512,17592127)
                             /u01/app/oracle/product/11.2.0.11/dbhome_1
```

     If the output from the command includes an 11.2.0.3.12 home, as in the preceding example:, then continue with Step 2. However, your output may appear similar to the following, which has no 11.2.0.3.12 home listed:

```
Oracle Home Name       Oracle Home version
                             Home Location
```

```
dbhome11202            11.2.0.2.5(13343424,13343447)
                             /u01/app/oracle/product/11.2.0.2/dbhome_1
```

If you do not have the required 11.2.0.3.12 home, then complete the following actions before proceeding to Step 2:

- Download the 11.2.0.3.12 RDBMS Clone Patch 14777276 (file name p14777276_121200_Linux-x86-64.zip) from My Oracle Support.
- Create an 11.2.0.3.12 database home on Node 0 with the following command:

  ```
  # opt/oracle/oak/bin/oakcli create dbhome -version 11.2.0.3.12
  ```

- Return to the start of Step 1 and repeat the actions, beginning with the `oakcli show dbhome` command, until you have an 11.2.0.3.12 database home.

2. Run the `oakcli show databases` command, on Node 0 only, to confirm that you have a database with the appropriate release number for this upgrade. The command and output should look similar to the following example, which shows three databases that are candidates to be upgraded.

```
# oakcli show databases
Database Name      Database Type     Database HomeName
    Database HomeLocation                       Database Version
tpcc               RAC               dbhome11202
    /u01/app/oracle/product/11.2.0.2/dbhome_1   11.2.0.2.5(13343424,13343447)
RACOne             RACOneNode        dbhome11202
      /u01/app/oracle/product/11.2.0.2/dbhome_1 11.2.0.2.5(13343424,13343447)
EE                 SINGLE            dbhome11202
   /u01/app/oracle/product/11.2.0.2/dbhome_1    11.2.0.2.5(13343424,13343447)
```

3. Upgrade a database by running the `oakcli upgrade database` command, on Node 0 only, providing the name of the database to upgrade and the 11.2.0.3.12 database home name from Step 1. The following example shows how to upgrade the tpcc database, listed in Step 2, using the OraDb11203_home1 identified in the example shown in Step 1:

```
# oakcli upgrade database -db tpcc -to OraDb11203_home1
```

4. After the command completes, verify your database version with the `oakcli show databases` command on Node 0 again. The command and output should now look similar to the following example:

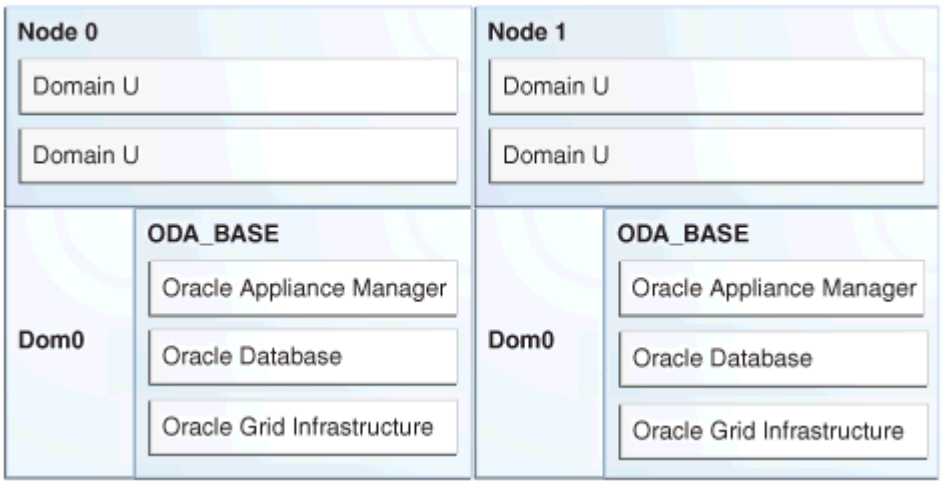
```
# oakcli show databases
Database Name      Database Type     Database HomeName
    Database HomeLocation                       Database Version
tpcc               RAC               OraDb11203_home1
    /u01/app/oracle/product/11.2.0.3/dbhome_1   11.2.0.3.12(18522512,17592127)
RACOne             RACOneNode        dbhome11202
    /u01/app/oracle/product/11.2.0.2/dbhome_1   11.2.0.2.5(13343424,13343447)
EE                 SINGLE            dbhome11202
    /u01/app/oracle/product/11.2.0.2/dbhome_1   11.2.0.2.5(13343424,13343447)
```

3

Managing Oracle Database Appliance Virtualized Platform

After you deploy Oracle Database Appliance Virtualized Platform, your system will have two domains on each server node, Dom0 and ODA_BASE (also known as Dom1). You can use the CPU cores not assigned to Oracle Database (ODA_BASE) for virtual machines, each of which is referred to as user domain or a Domain (Dom) U. The following figure shows the virtualized structure as a block diagram.

Figure 3–1 Oracle Database Appliance Virtualized Platform



Oracle recommends that you use shared repositories with Oracle Database Appliance Virtualized Platform for high availability. The virtual machine files can be stored on shared disks, providing shared storage for the database as well as the application virtual machines. Additionally, CPU pools and a resizeable Oracle Database domain (ODA_BASE) ensure that the virtual machines do not consume cycles from each other or from your assigned database CPU cores. Find details about the uses and configuration of these features, as well as information about resizing your Oracle Database domain (ODA_BASE), in the following sections of this chapter:

- Overview of Guest Virtual Machine Deployments
- About Shared Repositories and Virtual Machines on Oracle Database Appliance Virtualized Platform
- Managing Shared Repositories and Virtual Disks on Oracle Database Appliance Virtualized Platform

- About Virtual Machine Templates and Assemblies on Oracle Database Appliance Virtualized Platform
- Managing Virtual Machine Templates and Assemblies on Oracle Database Appliance Virtualized Platform
- About Virtual Machines on Oracle Database Appliance Virtualized Platform
- Managing Virtual Machines High Availability on Oracle Database Appliance Virtualized Platform
- Managing Virtual Machines on Oracle Database Appliance Virtualized Platform
- About CPU Pools on Oracle Database Appliance Virtualized Platform
- Managing CPU Pools on Oracle Database Appliance Virtualized Platform
- About Network Infrastructure and Virtual Local Area Networks on Oracle Database Appliance Virtualized Platform
- Resizing ODA_BASE

Many of the sections that cover management of a feature include a set of related examples.

Overview of Guest Virtual Machine Deployments

The main steps to deploy a guest virtual machine are as follows:

1. Create a new repository if necessary.
2. Import the required template or assembly.
3. Configure the template or assembly.
4. Clone the template or assembly.
5. Update the virtual machine as required.
6. Start the virtual machine.
7. Access the virtual machine using the virtual machine console.

For additional details and options available to complete each of the preceding steps, see the remaining sections in this chapter.

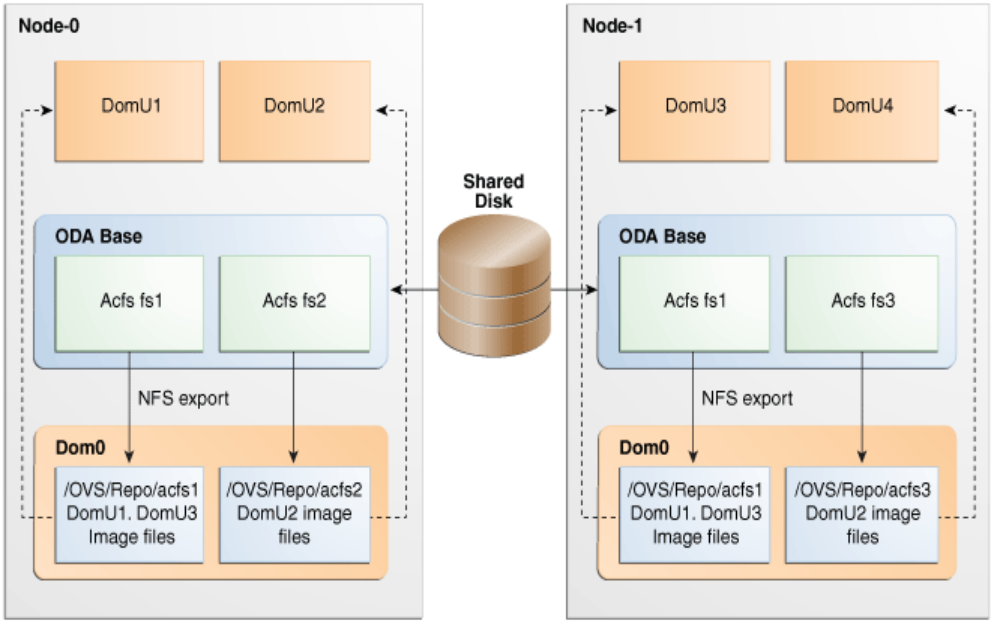
About Shared Repositories and Virtual Machines on Oracle Database Appliance Virtualized Platform

Oracle Database Appliance Virtualized Platform enables you to create one or more shared repositories for the storage of virtual machine (VM) files. A VM shared repository provides high availability support. A VM can be configured to fail over from one node to another node in case of node failure, and a VM can auto-restart on the failover node if the preferred node is not available.

Additionally, you can create virtual disks in shared repositories. Virtual disks provide additional storage options for virtual machines, by allowing you to attach additional block storage to your VMs. Similarly, you can detach the storage if you no longer need the additional space. You can use virtual disks to expand existing file system storage inside the VM by extending the storage volume onto a virtual disk or creating a new file system on a virtual disk. Your virtual disks can also be shared by multiple VMs running on same shared repository.

The following figure shows a typical architecture of Oracle Database Appliance Virtualized Platform with a shared storage system.

Figure 3–2 Architecture Overview of Oracle Database Appliance Virtualized Platform Shared Repositories



The preceding figure shows the shared disks on Oracle Database Appliance Virtualized Platform connected directly to ODA_BASE. ODA_BASE contains three shared repositories named fs1, fs2, and fs3. Each shared repository is an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) in ODA_BASE created on top of the ASM disk group (DATA or RECO) chosen for the repository. The process that creates a repository also performs an NFS export of the repository to the respective Dom0 via the private network. The export enables shared storage for the virtual machine files.

With the configuration shown in the figure, you can create multiple repositories. Mount these repositories either on the nodes where the virtual machine needs to run (such as fs1 and fs3 in the figure), or on both the nodes (such as fs2 in the illustration). Create one or more virtual machines or virtual machine templates on these shared repositories.

Create and manage shared repositories, virtual disks, and their virtual machines, including all of the underlying architecture shown in the figure, with Oracle Appliance Manager commands.

Managing Shared Repositories and Virtual Disks on Oracle Database Appliance Virtualized Platform

To create a shared repository, use the `oakcli create repo` command to identify the repository name, the disk group to use for its storage (DATA or RECO), and its size (in Gigabytes). Once you have created a shared repository, start the repository with the `oakcli start repo` command to make the storage available before assigning a virtual machine to the repository. Note that this command will also start any virtual machines assigned to the repository that are defined to be automatically started. The `oakcli`

`start repo` command has a required parameter for shared repositories to identify the node where the repository needs to be started.

Other Oracle Appliance Manager shared repository commands, including commands to show and to stop (dismount) existing repositories, are similar to to those used for non-shared repositories, but require a parameter to identify the node. Unlike the default repositories, which are permanent, you may delete a shared repository that has no active (mounted) virtual machines.

> **Note:** Do not issue an oakcli stop repo command while virtual machines are still active in the repository for the selected node.

The following commands have additional options for managing virtual machines on shared repositories:

- `oakcli configure vm`

  Include a `-prefnode` clause, to identify the node where you want the virtual machine to run by default, and a `-failover` clause, to indicate if you want the virtual machine to use the other node when the preferred node is not available (either at startup or while already active).

  > **Note:** If your virtual machine is assigned to a specific CPU pool and is allowed to failover, then the virtual machine will try to use the same CPU pool on the secondary node. If the CPU pool exists but is a different size, then the performance of your virtual machine might be impacted when running on the secondary node. If the assigned CPU pool does not exist on the secondary node, then the virtual machine will not fail over.

- `oakcli clone vm`

  Use the name of the shared repository in the `-repo` clause and include a `-node` clause to identify the node on which you want to perform the cloning process.

- `oakcli configure repo`

  Use the name of the shared repository in the `repo` clause and include the `-incsize` parameter to increase the size of the repository. The number you provide in the -incsize clause defaults to gigabytes but you can changed the unit to megabytes by appending `M` to the size.

- `oakcli import vmtemplate`

  Use the name of the shared repository in the `-repo` clause and include a `-node` clause to identify the node where you want to import a template or an assembly.

To create a virtual disk within a shared repository, use an `oakcli create vdisk` command to define the size of the virtual disk, the shared repository in which it will reside, a unique name within that shared repository, and whether or not the virtual disk can be shared by multiple virtual machines.

After you create one or more virtual disks, you can see information about them all using an `oakcli show vdisk` command. To see details about an individual virtual disk, use the same command but include the virtual disk name and a `-repo` clause to name the shared repository where you created the virtual disk. You may also clone virtual disks using an `oakcli clone vdisk` *`clone_vdisk`* `-repo` *`repo_name`* `-src` *`src_vdisk`* command, where *`clone_vdisk`* is the name to assign to the cloned virtual disk,

repo_name is the repository where the original virtual disk resides, and *src_vdisk* is the name of the virtual disk that you are cloning.

You can also delete a virtual disk with an `oakcli delete vdisk` command that includes the virtual disk name and a `-repo` clause to name shared repository.

The section "Examples of Oracle Appliance Manager Commands for Shared Repositories, Virtual Disks, and Virtual Machines" contains examples of commands used to manage shared repositories and virtual disks.

Examples of Oracle Appliance Manager Commands for Shared Repositories, Virtual Disks, and Virtual Machines

Example 1 Create a Shared Repository

This command creates a shared repository named *repo1* in the ASM DATA disk group with 30 gigabytes of available storage:

```
oakcli create repo repo1 -dg data -size 30
```

Example 2 Show the Status of All Shared Repositories

This command displays information about all existing repositories, which includes the default, local repositories as well as the shared repositories:

```
oakcli show repo

      NAME                    TYPE          NODENUM  FREE SPACE     STATE

      odarepo1                local          0             N/A     N/A
      odarepo2                local          1             N/A     N/A
      repo1                   shared         0          68.25%     ONLINE
      repo1                   shared         1          68.25%     ONLINE
      repo2                   shared         0             N/A     OFFLINE
      repo2                   shared         1          89.83%     ONLINE
```

Example 3 Start a Shared Repository

This command starts the shared repository named repo1 on Node 1:

```
oakcli start repo repo1 -node 1
```

Example 4 Stop a Shared Repository

This command stops the shared repository named repo1 on Node 0:

```
oakcli stop repo repo1 -node 0
```

Example 5 Show the Status of Named Shared Repository

This command displays information from Node 1 about the shared repository named repo1:

```
oakcli show repo repo1 -node 1

Resource: repo1_1
        AutoStart       :       restore
        DG              :       DATA
        Device          :       /dev/asm/repo1-286
        ExpectedState   :       Online
```

```
FreeSpace       :       87.703125M
MountPoint      :       /u01/app/repo1
Name            :       repo1_0
Node            :       all
RepoType        :       shared
Size            :       30720
State           :       Online
Version         :       2
```

Example 6 Delete a Shared Repository

This command deletes the shared repository named repo1 if the repository is offline (stopped) on both nodes:

```
oakcli delete repo repo1
```

Example 7 Import Virtual Machine Templates from an External Repository Assembly into a Shared Repository

This command imports virtual machine templates contained in an external repository template assembly file. Note the single quotation marks that enclose the URL. Assuming that the assembly contains three different templates, they are assigned the names *myol6u_15gb1*, *myol6u_15gb2*, and *myol6u_15gb3*, and they are imported into the shared repository, *repo2*, on Node 1.

```
oakcli import vmtemplate myol6u_15gb -assembly
'http://example.com/assemblies/OEL6/OVM_OL6U1_x86_PVHVM_15GB.ova'
-repo repo2 -node 1
```

Example 8 Create a Virtual Machine from a Template in a Shared Repository

This command creates a virtual machine named *myol6u_test* from the virtual machine template named myol6u_15gb1, which is stored in shared repository named repo2 on Node 0.

```
oakcli clone vm myol6u_test -vmtemplate myol6u_15gb1 -repo repo2 -node 0
```

> **Note:** The `-node` clause identifies the node where the cloning activity is to be run. Also, the node value does not assign the default startup node for the virtual machine, this assignment is set by the `oakcli configure vm` command.

Example 9 Configure a Virtual Machine for Use on a Shared Repository

This command sets values for specific resources in the virtual machine named *myol16u_test*:

- number of CPUs assigned to the virtual machine when started (vcpu)
- CPU access priority (cpuprio)
- maximum percentage of a CPU's capacity that will be assigned to the virtual machine (cpucap)
- amount of memory assigned when the virtual machine starts up (memory)
- the node where the virtual machine would normally start automatically when the shared repository is started or when the virtual machine is started manually (prefnode)
- enable automatic failover if the default node (prefnode) is not available (failover)

These values will override values assigned to these same parameters in the virtual template from which this virtual machine was derived. The virtual machine will use default values for parameters that are not defined in either the parent template or in a configuration command.

```
oakcli configure vm myol16u_test
-vcpu 2 -cpuprio 150 -cpucap 20 -memory 1G
-prefnode 0 -failover true
```

Example 10 Create a Virtual Disk for Use in a Shared Repository

This command creates a virtual disk named *sdisk1* in the *repo2* shared repository with a size of 1 gigabyte and the ability to be shared by virtual machines:

```
oakcli create vdisk sdisk1 -repo repo2 -size 1G -type shared
```

Example 11 Attach a Virtual Disk to a Virtual Machine

This command attaches the virtual disk named *sdisk1*, as created in Example 11, to the virtual machine named *myol6u_test* in the *repo2* shared repository, as created in Example 8:

```
oakcli modify vm myol6u_test -attachvdisk sdisk1
```

About Virtual Machine Templates and Assemblies on Oracle Database Appliance Virtualized Platform

Import and configure virtual machine templates as the source for the virtual machines deployed on Oracle Database Appliance Virtualized Platform. If you have created shared repositories, then import templates into the desired repository, otherwise import templates into the local repository on the desired node.

You might also import assemblies that contain one or more templates. When you import a template or assembly into a shared repository, identify the node that will perform the operation. Avoid overworking a busy node by selecting the node carefully. The repository will be available to both nodes no matter which node performs the import.

Templates imported into local nodes use the repository name supplied in the import command to identify the node that will complete the import and provide the storage. On Node 0, the local repository is named `odarepo1` and on Node 1, the local repository is named `odarepo2`..If you want to import a template to both local repositories, then you must provide a different template name when you import the template into the second node.

> **Note:** You cannot create or clone templates directly on Oracle Database Appliance Virtualized Platform. Find virtual machine templates at http://edelivery.oracle.com/linux.

If you import an assembly that contains more than one template, then the command automatically modifies the template name that you provide so that all template names remain unique. The first template will have the number "1" appended to the name, the second template will have the number "2" appended, and so on.

Once you have imported a virtual machine template, you can customize the template with Oracle Appliance Manager commands. For details about all the commands to manage virtual machines, see the section "Managing Virtual Machine Templates and Assemblies on Oracle Database Appliance Virtualized Platform".

Managing Virtual Machine Templates and Assemblies on Oracle Database Appliance Virtualized Platform

Use Oracle Appliance Manager `import` commands to store and name virtual machine templates for Oracle Database Appliance Virtualized Platform. Customize and manage the templates with additional Oracle Appliance Manager commands.

Examples of the commands described in this section are available in "Examples of Oracle Appliance Manager Virtual Machine Templates and Assembly Management Commands".

Importing Virtual Machine Templates

Use the Oracle Appliance Manager `oakcli import vmtemplate` command to import virtual machine templates and assemblies. You can import virtual machine templates and assemblies directly from a remote repository using a URL to identify the source of the files. Optionally, use a remote copy command to copy files from the remote repository into your Dom0 `/OVS` directory and then import the files using the path and names to identify the downloaded files.

Note: When importing templates or assemblies to a local repository, do not use the `-node` clause. The target node is implicit in the name of the repository.

Displaying and Modify Virtual Machine Template Configurations

Once you have imported a virtual machine template to a storage repository, examine the template configuration parameters with the Oracle Appliance Manager `oakcli show vmtemplate` command. If you need to reconfigure the template for specific requirements, then use the Oracle Appliance Manager `oakcli configure vmtemplate` command. This is useful if you plan to deploy multiple virtual machines with the same characteristics from a single template. If you will be deploying only one virtual machine or many virtual machines but with different characteristics, then set required values in the virtual machines with the `oakcli configure vm` command after you deployed the template.

Listing Stored Virtual Machine Templates

To find all your stored virtual machine templates, use the Oracle Appliance Manager `oakcli show vmtemplate` command with no parameters. If you no longer need a template that you previously stored, then remove the template from the repository with the Oracle Appliance Manager `oakcli delete vmtemplate` command.

Examples of Oracle Appliance Manager Virtual Machine Templates and Assembly Management Commands

Example 1 Import a Virtual Machine Template from Dom0

This command imports a virtual machine template that is defined in the file named `OVM_OL5U5_X86_64_PVM_10GB.tgz`. This file was previously copied from an external

template repository into the /OVS file system on Dom0. The template is assigned the name *myol5u1* and is imported into the repository on Node 0.

```
oakcli import vmtemplate myol5u
-files /OVS/OVM_OL5U5_X86_64_PVM_10GB.tgz -repo odarepo1
```

Example 2 Import a Virtual Machine Template Using an External Repository URL

This command imports a virtual machine template file named OVM_OL5U7_X86_64_PVM_10GB.tgz from an external template repository. Note the single quotation marks that enclose the URL. The templates is assigned the name *myol5u7_10gb* and is imported into the repository on Node 1.

```
oakcli import vmtemplate myol5u7_10gb -files
'http://example.com/vmtmplt/OEL5/OVM_OL5U7_X86_64_PVM_10GB.tgz'
-repo odarepo2
```

Example 3 Import Virtual Machine Templates from an External Repository Assembly

This command imports virtual machine templates contained in an external template repository assembly file. Note the single quotation marks that enclose the URL. Assuming that the assembly contains three different templates, they are assigned the names *myol6u_15gb1*, *myol6u_15gb2*, and *myol6u_15gb3*, and they are imported into the repository on Node 1.

```
oakcli import vmtemplate myol6u_15gb -assembly
'http://example.com/assemblies/OEL6/OVM_OL6U1_x86_PVHVM_15GB.ova'
-repo odarepo2
```

Example 4 Configure a Virtual Machine Template

This command sets values for specific resources in the virtual machine template named *myol5u7_10gb*:

- Number of CPUs assigned when the virtual machine starts up (vcpu)
- Maximum number of CPUs that can be assigned to the virtual machine (maxvcpu)
- Maximum percentage of a CPU's capacity that will be assigned to the virtual machine (cpucap)
- Amount of memory assigned when the virtual machine starts up (memory)
- Maximum amount of memory that can be assigned to the virtual machine (maxmemory)
- Operating system used by the virtual machine (os)

These values will become the default values for any virtual machine cloned from this template, although you can change any or all them later with the Oracle Appliance Manager `oakcli configure vm` command.

```
oakcli configure vmtemplate myol5u7_10gb -vcpu 2 -maxvcpu 4 -cpucap 40
-memory 1536M -maxmemory 2G -os OTHER_LINUX
```

Example 5 Configure Network Information in a Virtual Machine Template

This command sets *net1* as the network used to access a virtual machine cloned from the *myol5u7_10gb* virtual machine template.

```
oakcli modify vmtemplate myol5u7_10gb -addnetwork net1
```

Example 6 List the Existing Virtual Machine Templates

This command displays the name and repository for each virtual machine template as well as the default number of CPUs and default amount of memory that Oracle Database Appliance Virtualized Platform would assign to a virtual machine created from the template.

```
oakcli show vmtemplate
```

Example 7 Show Configured Values for a Virtual Machine Template

This command displays the values for the configurable options in the virtual machine template named *myol5u7_10gb*.

```
oakcli show vmtemplate myol5u7_10gb
```

Example 8 Remove a Virtual Machine Template

This command removes the virtual machine template named myol6u_15gb3 from Oracle Database Appliance Virtualized Platform.

```
oakcli delete vmtemplate my0l6u_15gb3
```

About Virtual Machines on Oracle Database Appliance Virtualized Platform

Deploy virtual machines on Oracle Database Appliance Virtualized Platform to run applications and other software on CPUs that are independent of Oracle Database software running in ODA_BASE. Use Oracle Appliance Manager to create and manage virtual machines, including starting and stopping them, and opening consoles for GUI access.

Note: You can use a VNC session to open a GUI console for a virtual machine.

You create ("clone") virtual machines from imported templates (or assemblies) or else from an existing virtual machine. In the former case, your virtual machine can be a complete clone of the contents of the template or assembly, or it can be a snap clone, which contains only the metadata of the virtual machine definition. In the latter case, all copies of existing virtual machines must be snap clones, either of a complete virtual machine or a snap clone.

You can clone virtual machines that have attached virtual disks. In such cases, a clone of a virtual machine will contain the contents of virtual disks attached locally but not of virtual disks attached in shared mode. If needed, clone the virtual disks required by a cloned virtual machine in a separate step.

Note: Creating snap clones is a very fast operation compared to cloning the entire template or assembly. However, if you update the contents of a snap clone, your system will acquire additional storage for the clone to retain the original content as well as the changed content, unless the changes are on shared virtual disks.

Unless you are using local repositories, you could set high-availability options for your virtual machines. These include identifying the node where the virtual should be started by default and whether the virtual machine should be failed over to the other

node. Failover can occur if the node where the virtual is already running should fail or if the preferred node is not available when the virtual attempts to start.

Although it is possible to reconfigure an active virtual machine, your changes would not take effect until you stopped and restarted the virtual machine. You can also display high level information about all of your virtual machines or detailed information about the configuration of single virtual machines. When you no longer have use for a particular virtual machine, then delete the virtual machine to regain storage.

For details about the commands you use to manage virtual machines, see "Managing Virtual Machines on Oracle Database Appliance Virtualized Platform".

> **See Also:** *Oracle Database Appliance Release Notes* for the version of Oracle VM supported by each release of Oracle Database Appliance. Oracle Database Appliance Virtualized Platformsupports all guest operating systems provided by the supported version of Oracle VM. Find the supported guest operating systems in the related Oracle VM Release Notes.

Managing Virtual Machines High Availability on Oracle Database Appliance Virtualized Platform

A system that is highly available is designed to provide uninterrupted computing services during essential time periods, during most hours of the day, and most days of the week throughout the year. Reliability, recoverability, timely error detection, and continuous operations are primary characteristics of a highly available solution. Two essential parts of high availability are automatic restart and failover.

Automatic Restart

A virtual machine automatically restarts after a hardware or software failure or whenever your database host computer restarts. Using the `oakcli configure vm` command, you can start a virtual machine on a preferred node by setting the `-prefnode` parameter.

Failover

Failover lets the virtual machine be restarted on the other node with no manual intervention in the event of an unplanned node failure. The `-failover` parameter, used with the `oakcli configure vm` command, enables a virtual machine to fail over from one node to another.

> **See Also:** "oakcli configure vm" on page 5-10 for information about using the `-prefnode` and `-failover` parameters

Managing Virtual Machines on Oracle Database Appliance Virtualized Platform

Create and manage user domain virtual machines on Oracle Database Appliance Virtualized Platform using Oracle Appliance Manager commands. Access your active virtual machines using the command line (after configuring the virtual machine with commands similar to those in Example 7 later in this section) or GUI VM consoles opened with Oracle Appliance Manager.

Use Oracle Appliance Manager commands for the following tasks:

- Creating an image for a new virtual machine

  Use the `oakcli clone vm` command to create an image for a new virtual machine on Oracle Database Appliance. By default, the image inherits the content and configuration information from the template or from the virtual machine that you are cloning and resides in the same repository as the source. To create a snap clone, which will only contain the configuration metadata, include the `-snap` clause in the `oakcli clone` command.

  Unless you are using local repositories for your virtual machines, include the `-node` clause in the `oakcli clone` command to identify the node that should run the cloning process. To create a snap clone, include the `-snap` clause in the `oakcli clone` command.

- Displaying and Modifying virtual machine configurations

  To see the current configuration of a virtual machine image, use the Oracle Appliance Manager `oakcli show vm` command. If you need to make changes to the configuration, such as setting high-availability options, then run the Oracle Appliance Manager `oakcli configure vm` command.

  Note: CPU capacity is controlled either by the CPU pool assigned to a virtual machine (by the `-cpupool` parameter), or by the default_unpinned_pool if the virtual machine is configured without a `-cpupool` parameter.

  Note: If you reconfigure a virtual machine that is currently running, then your changes will not be effective until you stop and restart the virtual machine.

- Starting and stopping virtual machines

  When you are ready to start a virtual machine on Oracle Database Appliance Virtualized Platform, run the Oracle Appliance Manager `oakcli start vm` command Similarly, to stop a virtual machine, run the `oakcli stop vm` command.

- Sending messages to active virtual machines

  New Oracle Virtual Machine Templates include a utility, Oracle VM Guest Additions (ovmd) that provides a messaging interface for first-boot installation configuration. To send such messages to a virtual machine on Oracle Database Appliance Virtualized Platform, use the `oakcli modify vm` command with the `-s` parameter, where the argument for the parameter is list of parameters enclosed in single or double quotation marks. The elements in the list are key and value pairs with a colon delimiter to separate the pair of values and a semi-colon to separate value pairs from each other. The following example has two value pairs.

  ```
  oakcli modify vm vmo16u3 -s
  'com.oracle.linux.network.device.0:eth0;com.oracle.linux.network.ipaddr.0:192.1
  .2.18'
  ```

  See Also: The section *Using the Oracle VM Guest Additions* in *Oracle VM Utilities Guide for x86* for more information about ovmd, and the Oracle VM Guest Additions.

- Accessing an active virtual machine

To open a GUI virtual machine console for an active virtual machine, run the Oracle Appliance Manager `oakcli show vmconsole` command. If the console does not display correctly, then close the console, set the DISPLAY environment variable to an appropriate value for your monitor, and then rerun the `oakcli show vmconsole` command.

You can also configure your virtual machine (using `oakcli modify vm` commands with the `-s` parameter) to allow access from the Oracle Database Appliance command line instead of a virtual machine console.

- Listing the virtual machines in your repositories

  Display a list your existing virtual machines, including some basic information about each one, with the Oracle Appliance Manager `oakcli show vm` command.

- Adding a virtual disk to or removing a virtual disk from a virtual machine

  Use an `oakcli modify vm` *`virtual_machine_name`* command with an `-attachvdisk` or a `-detachvdisk` clause that names the virtual disk to add or remove the virtual disk.

- Removing a virtual machine from a repository

  Remove an unwanted virtual machine with the Oracle Appliance Manager `oakcli delete vm` command.

For examples of the commands to manage virtual machines described in this section, see "Examples of Oracle Appliance Manager Virtual Machine Commands".

Examples of Oracle Appliance Manager Virtual Machine Commands

Example 1 Create a Virtual Machine Image from a Template

This command creates a complete virtual machine image named *`myol15u_test`* from the virtual machine template named *`myol15u`* which is stored in the shared repository *`shrepo`* on Node 0.

```
oakcli clone vm myol15u_test -vmtemplate myol5u -repo shrepo -node 1
```

Example 2 Create a Snap Clone from a Template

This command creates a snap clone named *`myol15u_snap`* from the virtual machine template named *`myol15u`*.

```
oakcli clone vm myol15u_snap -vmtemplate myol5u -snap
```

Example 3 Create a Snap Clone of an Existing Virtual Machine

This command creates a snap clone named *`myol15u_test1`* from a virtual machine named *`myol15u_test`*.

```
oakcli clone vm myol15u_test1 -vm myol5u_test1 -snap
```

Example 4 Configure a Virtual Machine

This command sets values for specific resources in the virtual machine named *myol5u_test*:

- Number of CPUs assigned to the virtual machine when started (vcpu)
- CPU access priority (cpuprio)
- Maximum percentage of a CPU's capacity that will be assigned to the virtual machine (cpucap)

- Amount of memory assigned when the virtual machine starts up (memory)
- CPU pool to be assigned to the virtual machine (cpupool)
- Definition of the keyboard type to be used for virtual machine access (keyboard)
- Definition of the mouse type to be used for virtual machine access (mouse)

These values will override values assigned to these same parameters in the virtual template from which this virtual machine was derived. The virtual machine will use default values for parameters that are not defined in either the parent template or in a configuration command.

```
oakcli configure vm myol5u_test
-vcpu 2 -cpuprio 150 -cpucap 20 -memory 1G
-cpupool linpool -keyboard en-us -mouse USB_MOUSE
```

Note: CPU capacity is controlled by the CPU pool assigned to a virtual machine by either the `-cpupool` parameter or by the default_unpinned_pool (if the virtual machine is not configured with a `-cpupool` parameter). Values for `-vcpu` and `-maxcpu` parameters that are larger than the number of CPUs in the assigned CPU pool are ignored.

Example 5 List the Existing Virtual Machine Images

This command displays the name, the repository, and the current state (online or offline) for each virtual machine. The output also contains the default number of CPUs and default amount of memory that Oracle Database Appliance Virtualized Platform will assign to each virtual machine.

```
oakcli show vm
```

Example 6 Show Configured Values for a Virtual Machine

This command displays the definition of a virtual machine named *myol5u_test*. The output contains the current values for all of the configurable parameters along with additional information such as the virtual machine autostart setting.

```
oakcli show vmtemplate myol5u_test
```

Example 7 Start a Virtual Machine

This command starts the virtual machine named *myol5u_test.*

```
oakcli start vm myol5u_test
```

Example 8 Open a VM Console for a Virtual Machine

This command opens a GUI VM console window to an active virtual machine named *myol5u_test.*

```
oakcli show vmconsole myol5u_test
```

Note: The `oakcli show vmconsole` command requires a valid definition for the DISPLAY environment variable to work correctly.

Example 9 Set Up a Virtual Machine for Access from eth0 Using an IP Address

The following set of commands configures the virtual machine vmol6u3 (which has ovmd) with the IP address 192.168.16.51 for the eth0 interface and sets the `root` password to `password123a`.

```
oakcli clone vm vmol6u3 -vmtemplate ol6u3 -repo shrepo -node 0
oakcli modify vm vmol6u3 -addnetwork priv1
oakcli start vm vmol6u3
oakcli modify vm vmol6u3 -s 'com.oracle.linux.network.device.0:eth0'
oakcli modify vm vmol6u3 -s 'com.oracle.linux.network.onboot.0:yes'
oakcli modify vm vmol6u3 -s 'com.oracle.linux.network.bootproto.0:static'
oakcli modify vm vmol6u3 -s 'com.oracle.linux.network.ipaddr.0:192.168.16.51'
oakcli modify vm vmol6u3 -s 'com.oracle.linux.network.netmask.0:255.255.255.0'
oakcli modify vm vmol6u3 -s 'com.oracle.linux.root-password:password123'
```

Example 10 Add a Virtual Disk to a Virtual Machine

This command adds a virtual disk named *sdisk1* to the virtual machine named *myol5u_test*.

```
oakcli modify vm myol5u_test -attachvdisk sdisk1
```

Example 11 Stop a Virtual Machine

This command stops the virtual machine named *myol5u_test*.

```
oakcli stop vm myol5u_test
```

Example 12 Remove a Virtual Machine

This command removes the virtual machine named myol5u_test from Oracle Database Appliance Virtualized Platform.

```
oakcli delete vm my0l5u_test
```

About CPU Pools on Oracle Database Appliance Virtualized Platform

Isolate workloads by creating CPU pools and assigning (pinning) virtual machines to a specific CPU pool. When you pin a virtual machine to a CPU pool, you ensure that the virtual machine will use CPUs in that pool only.

When Oracle Database Appliance Virtualized Platform is configured, a *default-unpinned-pool* is created on each node. The size of this pool depends on the hardware model as follows:

- On Oracle Database Appliance X5-2 Virtualized Platform, the default-unpinned-pool contains 72 CPUs
- On Oracle Database Appliance X4-2 Virtualized Platform, the default-unpinned-pool contains 48 CPUs
- On Oracle Database Appliance X3-2 Virtualized Platform, the default-unpinned-pool contains 32 CPUs
- On Oracle Database Appliance Version 1 Virtualized Platform, the default-unpinned-pool contains 24 CPUs.

When you create the ODA_BASE domain, a new CPU pool, named *odaBaseCpuPool*, is created on both nodes and the required CPUs are removed from the default-unpinned-pool. ODA_BASE is the only domain allowed to use the CPUS in the odaBaseCpuPool. When you start other virtual machines they run on CPUs that were

left in the default-unpinned-pool, effectively ODA_BASE from the work being done by other virtual machines.

You might also cage groups of virtual machines by creating additional CPU pools. These additional pools enable you to pin a virtual machine, or a set of machines, to its own CPU pool. Virtual machines running in a specific CPU pool do not share CPU cycles with virtual machines running in other CPU pools. Define as many CPU pools as you want, up to the number of available CPUs on your system.

If your application requirements change over time, resize, add, or drop CPU pools as needed. Resize ODA_BASE if necessary, although this requires a special command that also updates your Oracle Database Appliance Virtualized Platform license. See the section "Resizing ODA_BASE" for details.

A CPU pool can have a different size on each node (except for the odaBaseCpuPool), as shown by the following `oakcli show cpupool` commands, one for each node:

```
oakcli show cpupool -node 0
                 Pool                          Cpu List
default-unpinned-pool   [14, 15, 16, 17, 18, 19, 20, 2
                                              1, 22, 23]
               twocpu                           [12, 13]
       odaBaseCpuPool   [0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
                                               10, 11]

oakcli show cpupool -node 1
                 Pool                          Cpu List
default-unpinned-pool   [12, 13, 14, 15, 16, 17, 18, 1
                                    9, 20, 21, 22, 23]
       odaBaseCpuPool   [0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
                                              10,11]
```

For commands to manage CPU pools, other than odaBaseCpuPool, see "Managing CPU Pools on Oracle Database Appliance Virtualized Platform".

Over-Subscribing CPU Pools

A CPU can belong to one and only one CPU pool, although you can assign multiple virtual machines to a CPU pool. A CPU pool becomes over-subscribed when the virtual machines that are active in the pool require more CPUs than you configured for the pool. For example, if a CPU pool has four CPUs, then you might start two virtual machines that have been defined to use four CPUs each. In this case, the CPU pool is over-subscribed because each of the four CPUs is supporting two virtual machines. Similarly, if you stop one of those virtual machines but start another one that requires two CPUs, then the CPU pool is still over-subscribed because two of the CPUs are supporting both virtual machines. When over-subscribing a CPU pool, you need to assess the performance of the virtual machines in that pool. You should be prepared to re-assign one or more virtual machines to a different CPU pool if sharing an over-subscribed pool degrades performance to unacceptable levels.

Managing CPU Pools on Oracle Database Appliance Virtualized Platform

Use Oracle Appliance Manager commands to manage CPU pools on each node of Oracle Database Appliance Virtualized Platform.

The actions you could perform on CPU pools include:

- Creating additional CPU pools with the Oracle Appliance Manager `oakcli create cpupool` command.

- Changing the number of CPUs allocated to a CPU pool with the Oracle Appliance Manager `oakcli configure cpupool` command.
- Examining your existing CPU pools with the Oracle Appliance Manager `oakcli show cpupool` command.
- Pinning a virtual machine to a specific CPU pool with the `-cpupool` option of the Oracle Appliance Manager `oakcli configure vm` command. You can pin multiple virtual machines to the same CPU pool.

See examples of the commands discussed in this section in "Examples of Oracle Appliance Manager CPU Pool Management Commands".

Examples of Oracle Appliance Manager CPU Pool Management Commands

Example 1 Create a New CPU Pool on Oracle Database Appliance Virtualized Platform

This command creates CPU pool named *winpool* with 4 CPUs on Node 0.

```
oakcli create cpupool winpool -numcpu 4 -node 0
```

Example 2 Change the Number of CPUs Assigned to a CPU Pool on Oracle Database Appliance Virtualized Platform

This command changes the number CPUs assigned to the CPU pool named *linpool* on Node 1. The new number of CPUs will be six after the command runs.

```
oakcli configure cpupool linpool -numcpu 6 -node 1
```

Example 3 Show the CPU Pools Configured on Oracle Database Appliance Virtualized Platform Nodes

This command displays the CPUs assigned to each defined CPU pool on Node 0. The command also lists the virtual machines, if any, assigned to each CPU pool.

```
oakcli show cpupool -node 0
```

Example 4 Assign a Virtual Machine to a CPU Pool on Oracle Database Appliance Virtualized Platform

This command pins the virtual machine named *wintest* to the CPU pool named *winpool*.

```
oakcli configure vm wintest -cpupool winpool
```

Note: You do not manage odaBaseCpuPool with `oakcli cpupool` commands. Instead, you have to use commands that configure ODA_BASE, as discussed in "Resizing ODA_BASE".

About Network Infrastructure and Virtual Local Area Networks on Oracle Database Appliance Virtualized Platform

To specify which network should access a virtual machine, you employ network infrastructure components of Oracle Database Appliance Virtualized Platform. This section describes these infrastructure components.

Oracle Database Appliance Virtualized Platform manages all the high level network infrastructure components for you by precreating the bonds and bridges for all

networks. The front end point for accessing a virtual machine will be one of the bridges defined for Dom0.

The following tables show the default network interfaces and are categorized by hardware as listed here:

- Table 3–1 and Table 3–2 list the default network interfaces for Oracle Database Appliance Virtualized Platform on systems with a storage shelf.
- Table 3–3 lists the default network interfaces for Oracle Database Appliance Virtualized Platform on systems without a storage shelf.

In all cases, connections to user domains are through the selected interfaces.

Table 3–1 Oracle Database Appliance Virtualized Platform Dual Port 10-GbE Network Interfaces

Interfaces at Dom0	Bond Devices at Dom0	Bridge in Dom0	Interfaces in ODA_BASE Domain
1. eth0	icbond0	priv1	eth0
2. eth1			

Table 3–2 Oracle Database Appliance Virtualized Platform On Board Quad Port 10-GbE Network Interfaces

Interfaces at Dom0	Bond Devices at Dom0	Bridge in Dom0	Interfaces in ODA_BASE Domain
1. eth2	bond0	net1	eth1
2. eth3			
1. eth4	bond1	net2	eth2
2. eth5			

Note: If you define a fiber public network connection, then bond0 is configured on PCIe boards.

Table 3–3 Oracle Database Appliance Version 1 Virtualized Platform Network Interfaces

Type	Interfaces at Dom0	Bond Devices at Dom0	Bridge in Dom0	Interfaces in ODA_BASE Domain
Private	eth0	bond0	priv1	eth0
	eth1			
On Board Public	eth2	bond1	net1	eth1
	eth3			
1st Pair Quad Port	eth4	bond2	net2	eth2
	eth5			
2nd Pair Quad Port	eth6	bond3	net3	eth3
	eth7			
10-GbE Interface	eth8	xbond0	net4	eth4
	eth9			

Note: If you define a fiber public network connection, then bond0 is configured on PCIe boards.

When you configure a virtual machine, you define which network the virtual machine should use by identifying the related bridge. For example, to connect a virtual machine named `myvm1` to the net1 network, you would use the following command:

```
oakcli modify vm myvm1 -addnetwork net1
```

Figure 3–3 Basic Virtual Machine Local Area Network

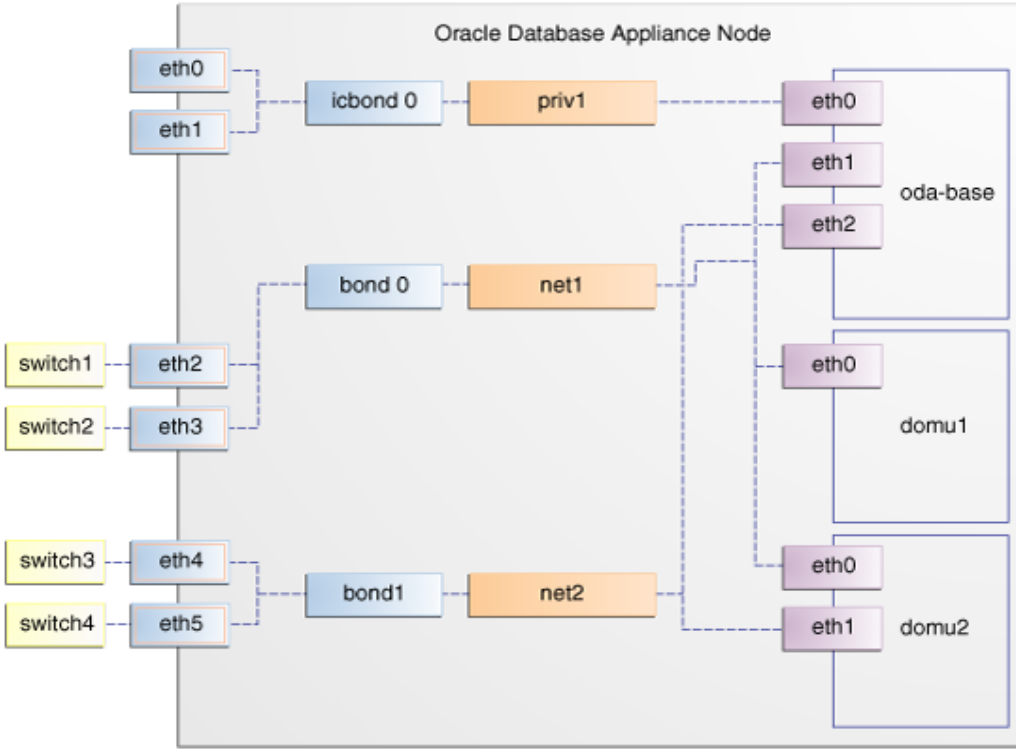
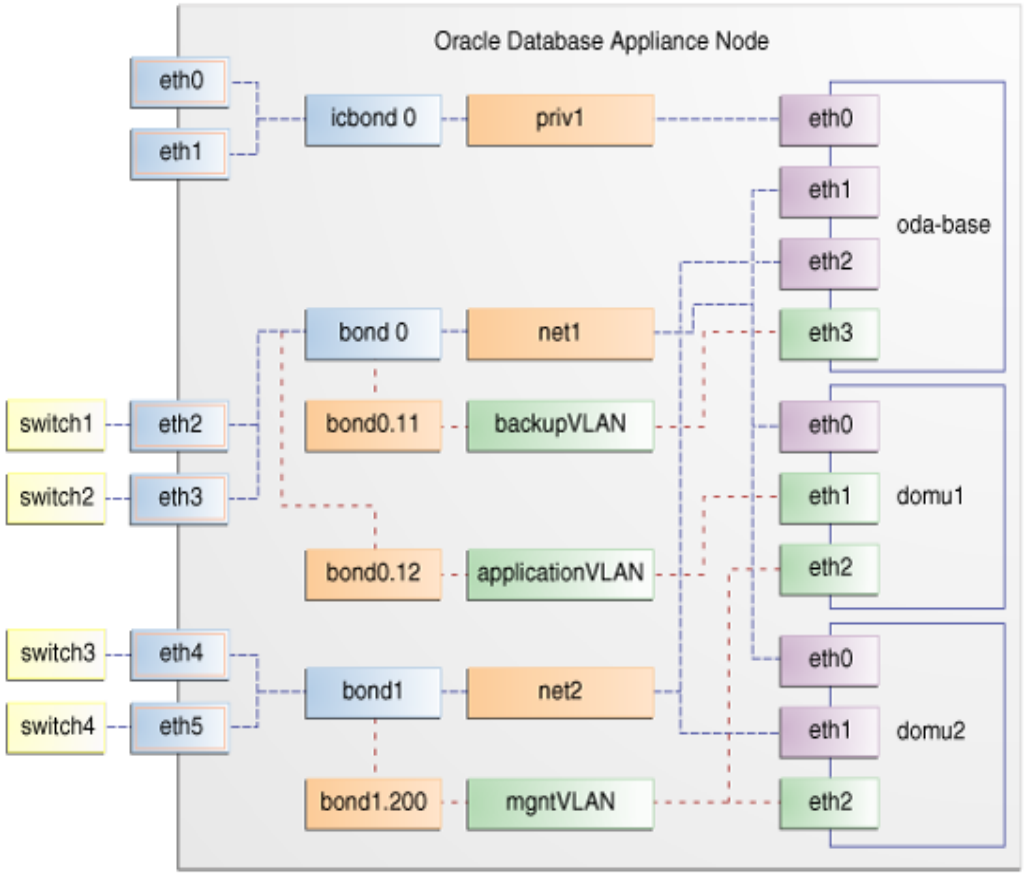


Figure 3–3 shows a typical Oracle Database Appliance configuration based on the preceding information.

During the installation and configuration of Oracle software on Oracle Database Appliance Virtualized Platform, you had an opportunity to assign default Virtual Local Area Networks (VLANs) to ODA_BASE. Figure 3–4, "Oracle Database Appliance Virtualized Platform with Virtual Local Area Networks" shows a typical Oracle Database Appliance configuration using VLANs. The figure shows the same configuration as Figure 3–3 but with three tagged VLANs added for backups (backup), for applications (application), and for management (mgnt).

Figure 3–4 Oracle Database Appliance Virtualized Platform with Virtual Local Area Networks



The following section, "Managing Virtual Local Area Networks on User Domains and on ODA_BASE", describe how to create new or remove existing VLANs in ODA_BASE and in your user domains respectively.

Managing Virtual Local Area Networks on User Domains and on ODA_BASE

Manage VLANs with Oracle Appliance Manager commands that are fully documented in Chapter 5, "Oracle Appliance Manager Command-Line Interface." To manage VLANs for user domains, log into ODA_BASE, and to manage VLANs for ODA_BASE, log into Dom0. The examples in this section use a VLAN named *sample10*.

> **Note:** You can create VLANs from Dom0 only before your deployment of Oracle Database Appliance or Oracle Database Appliance Virtualized Platform.

Creating a Virtual Local Area Network

To create a VLAN, use the `oakcli create vlan` command. You need to provide the following information to create a VLAN:

- VLAN name that is unique on the node where the VLAN is created (but which can be the same as a VLAN name on the other node of Oracle Database Appliance Virtualized Platform)

- VLAN tag number between 2 to 4096 inclusive that is unique on the node where the VLAN is created (but which can be the same as a VLAN tag number on the other node of Oracle Database Appliance Virtualized Platform)
- Name of the interface on which the VLAN is to be created. Find the available interfaces for your hardware listed in the *Bond Devices at Dom0* column in either Table 3–2 or Table 3–3.
- Node on which to create the VLAN

> **Note:** Create the same VLAN on both nodes, if needed, by issuing the `oakcli create vlan` command twice, once for node 0 and once for node 1.

The following example shows one option for creating the sample10 VLAN on node 0:

```
oakcli create vlan sample10 -vlanid 10 -if bond0 -node 0
```

> **WARNING: If you are planning to use a VLAN with a virtual machine created in a shared repository, then you should create that VLAN on both nodes. A virtual machine fails if an assigned network is not available on the node where the virtual machine is trying to run by default or following a failover.**

Assigning and Removing a Virtual Local Area Network for a User Domain

Use the oakcli modify vm command with an `-addnetwork` clause to assign an existing VLAN to a virtual machine and with a `-deletenetwork` clause to remove a VLAN from a virtual machine. The clauses must also contain the name of the VLAN.

The following example shows how to assign the sample10 VLAN to the myol5u_test virtual machine:

```
oakcli modify vm myol5u_test -addnetwork sample10
```

Assigning and Removing a Virtual Local Area Network for ODA_BASE

Use the `oakcli configure oda_base` command to add an existing VLAN to ODA_BASE or to remove a VLAN from ODA_BASE. Note that this command will also let you resize ODA_BASE and domain memory size. If you only want to manage VLANs, enter the number that corresponds to the number of your current CPU cores and current number for your memory.

In the following partial example, the CPU core count and default memory values are left unchanged, while the test01 VLAN is assigned to ODA_BASE. Note that the current CPU core count, 6, corresponds to selection number 3 in Core Licensing Options list of values.

```
# oakcli configure oda_base
Core Licensing Options:
        1. 2 CPU Cores
        2. 4 CPU Cores
        3. 6 CPU Cores
        4. 8 CPU Cores
        5. 10 CPU Cores
        6. 12 CPU Cores
        Current CPU Cores       :6
        Selection[1 : 6](default 12 CPU Cores) : 3
```

```
        ODA base domain memory in GB(min 8, max 88)(Current Memory 48G)[default
64]     : 48
INFO: Using default memory size i.e. 64 GB
Additional vlan networks to be assigned to oda_base? (y/n) [n]: y
Select the network to assign (test00,test01,test02,test03): test01
Additional vlan networks to be assigned to oda_base? (y/n) [n]:
Vlan network to be removed from oda_base (y/n) [n]:
INFO: . . .
```

Viewing and Deleting Virtual Local Area Networks

To see what VLANs currently exist in ODA_BASE, run the `oakcli show vlan` command. In the following example, we show a number of VLANs in addition to the sample10 VLAN used in previous examples:

```
oakcli show vlan

NAME                 ID    INTERFACE   NODENUM

net1                 1     bond0       0
net1                 1     bond0       1
net2                 1     bond1       0
net2                 1     bond1       1
net3                 2     bond1       0
net3                 4     bond0       1
net10                20    bond1       0
net10                20    bond1       1
```

To remove an unwanted VLAN from a node, use an `oakcli delete vlan` command, providing the VLAN name and the node number. The following command would remove the sample10 VLAN from node 0 (where the VLAN was assigned in the earlier `oakcli create vlan` example):

```
oakcli delete vlan sample10 -node 0
```

Note: You cannot delete a native VLAN, that is, a virtual local area network configured during the deployment of Oracle Database Appliance Virtualized Platform software.

Resizing ODA_BASE

Increase the number of CPU cores assigned to the ODA_BASE domain on Oracle Database Appliance Virtualized Platform if you need more computing power or memory for your installed Oracle databases. Alternatively, decrease the CPU cores if you need more CPUs assigned to your virtual machine domains. You must increase or decrease the assigned CPU core count on each node by two or multiples of two.

Oracle recommends that you use templates to size the databases that you deploy in ODA_BASE, following the guidelines in *Appendix A* of *Oracle Database Appliance Getting Started Guide*. Sum the number of CPUs that these databases require, using the tables in that Appendix, and divide the value by two determine the number of CPU cores you will need for ODA_BASE.

Plan to change your ODA_BASE core count when there is no critical activity running on your Oracle databases. This is because the ODA_BASE domain shuts down during the resizing process. When you are ready to proceed, complete these steps:

1. Log onto Dom0 and run the `oakcli configure oda_base` command as shown in this example, which changes the CPU core count from six to eight in ODA_BASE:

```
# oakcli configure oda_base
Core Licensing Options:
        1. 2 CPU Cores
        2. 4 CPU Cores
        3. 6 CPU Cores
        4. 8 CPU Cores
        5. 10 CPU Cores
        6. 12 CPU Cores
        Current CPU Cores       :6
        Selection[1 : 6](default 12 CPU Cores) : 10
        ODA base domain memory in GB(min 8, max 88)(Current Memory 64G)[default
32]     :
INFO: Using default memory size i.e. 32 GB
Additional vlan networks to be assigned to oda_base? (y/n) [n]:
Vlan network to be removed from oda_base (y/n) [n]
INFO: Node 0:Configured oda base pool
INFO: Node 1:Configured oda base pool
INFO: Node 0:ODA Base configured with new memory
INFO: Node 0:ODA Base configured with new vcpus
INFO: Changes will be incorporated after the domain is restarted on Node 0
INFO: Node 1:ODA Base configured with new memory
INFO: Node 1:ODA Base configured with new vcpus
INFO: Changes will be incorporated after the domain is restarted on Node 1
```

2. Perform any actions listed in the output from the command. Not all versions of the software, such as the one shown in the preceding example, require any additional actions before restarting ODA_BASE.
3. Restart the domain to implement the changed configuration for ODA_BASE by running the following restart command on Dom0 of both nodes:

```
oakcli restart oda_base
```

4

Validating and Troubleshooting Oracle Database Appliance

This chapter contains information about how to validate changes and troubleshoot Oracle Database Appliance problems. Various tools that perform one or both of these tasks are described in the following sections:

- Oracle Database Appliance Diagnostics and Validation Tool
- Oracle Database Appliance Configuration Error Messages
- Preparing Log Files for Oracle Support Services
- Additional Troubleshooting Tools and Commands
- Oracle Database Appliance Hardware Monitoring Tool

Oracle Database Appliance Diagnostics and Validation Tool

The Oracle Appliance Manager diagnostics and validation tool is managed with Oracle Appliance Manager `oakcli validate` commands. The tool provides diagnostic and validation functions to resolve support issues. If you experience problems with Oracle Database Appliance, then use the `oakcli validate` command to verify that your environment is properly configured and that best practices are in effect. When placing a service request, also use Oracle Appliance Manager as described in this chapter to prepare the log files to send to Oracle Support Services.

Note: The Oracle Appliance Manager diagnostics and validation tool is not available on hardware prior to Oracle Database Appliance X3-2.

Oracle Database Appliance Validation Tool Overview

Use the command `oakcli validate` to validate the status of Oracle Database Appliance. You must run the `oakcli validate` command as the `root` user.

The command uses the following syntax, where *`checklist`* is a single check or a comma-delimited list of checks, and *`output_file_name`* is the name that you designate for a validation output file:

```
oakcli validate -h
oakcli validate [-V | -l | -h]
oakcli validate [-v] [-f output_file] [-a | -d | -c checklist] [-v patch_version]
```

See the following two tables for a summary of the validation tool options and system checks.

Table 4–1 Oracle Database Appliance Validation Tool Options

Option	Purpose
`-a`	Run all system checks, including `DiskCalibration`. Oracle recommends that you use this command to validate system readiness before deployment. Do not run `oakcli validate` with this option on a busy production system, because the `DiskCalibration` system check can cause performance degradation. See Table 4–2 for details about each check.
`-c` *`checklist`*	Run the validation checks for the items identified in *`checklist`*, a comma-delimited list. Use this parameter to check either a single item or subset of items.
`-d`	Run only the default checks. The default checks are `NetworkComponents`, `OSDiskStorage`, `SharedStorage`, `DiskCalibration`, and `SystemComponents`. See Table 4–2 for details about each check.
`-f` *`output_file`*	Send output to a file with a fully-qualified file name, *`output_file`*, instead of to the screen (`stdout`).
`-h`	Display the online help.
`-l`	List the items that can be checked along with their descriptions.
`-v`	Show verbose output (must be used with a parameter that generates a validation report).
`-V`	Display the version of oakValidation.
-ver *`patch_version`*	Report any reasons for not being able to patch Oracle Database Appliance with the patch named in patch_version.

Table 4–2 Oracle Database Appliance Validation Checks

Check	Purpose
`asr`	Validate Oracle Auto Service Request (Oracle ASR) components based on Oracle ASR configuration file and Oracle Integrated Lights Out Manager (Oracle ILOM) sensor data.
`DiskCalibration`	Preinstallation check for the storage disk performance using `/opt/oracle/oak/orion`. Do not run this check after you have deployed Oracle software on Oracle Database Appliance, because running the `DiskCalibration` command on a deployed system creates performance issues.
`NetworkComponents`	Validate public and private network hardware connections. Note: This option is not valid on hardware prior to Oracle Database Appliance X3-2.
`OSDiskStorage`	Validate the operating system disks, and file system information.
`ospatch`	Validates that the system will be able to complete an upgrade successfully using the named patch
`SharedStorage`	Validate shared storage and multipathing information
`StorageTopology`	Validate the storage shelf connectivity
`SystemComponents`	Validate system components, based on Oracle ILOM sensor data readings.

Examples of Oracle Database Appliance Validation Tool Commands

The following command lists and describes all validation command options:

```
# oakcli validate -l
```

The following command runs all system checks:

```
# oakcli validate -a
```

The following command performs a system check for disk calibration:

```
# oakcli validate -c DiskCalibration
```

The following command runs system checks to validate hardware system components and Oracle Database Appliance network components:

```
# oakcli validate -c SystemComponents,NetworkComponents
```

Note: The `NetworkComponents` option is not available on hardware prior to Oracle Database Appliance X3-2.

The `oakcli validate -c StorageTopology` command performs a check of the cable configuration between the system controllers and the storage shelf, as well as the storage expansion shelf if one is installed. Oracle recommends that you run this command immediately after deploying the system or after adding an expansion storage shelf. The output shown in the following example reports a successful configuration. If the cabling is not correct, you would see errors in your output.

```
# oakcli validate -c storagetopology
 It may take a while. Please wait...
 INFO : ODA Topology Verification
 INFO : Running on Node0
 INFO : Check hardware type
 SUCCESS : Type of hardware found : X4-2
 INFO : Check for Environment(Bare Metal or Virtual Machine)
 SUCCESS : Type of environment found : Virtual Machine(ODA BASE)
 SUCCESS : Number of External LSI SAS controller found : 2
 INFO : Check for Controllers correct PCIe slot address
 SUCCESS : External LSI SAS controller 0 : 00:15.0
 SUCCESS : External LSI SAS controller 1 : 00:16.0
 INFO : Check if JBOD powered on
 SUCCESS : 1JBOD : Powered-on
 INFO : Check for correct number of EBODS(2 or 4)
 SUCCESS : EBOD found : 2
 INFO : Check for External Controller 0
 SUCCESS : Controller connected to correct ebod number
 SUCCESS : Controller port connected to correct ebod port
 SUCCESS : Overall Cable check for controller 0
 INFO : Check for External Controller 1
 SUCCESS : Controller connected to correct ebod number
 SUCCESS : Controller port connected to correct ebod port
 SUCCESS : Overall Cable check for controller 1
 INFO : Check for overall status of cable validation on Node0
 SUCCESS : Overall Cable Validation on Node0
 INFO : Check Node Identification status
 SUCCESS : Node Identification
 SUCCESS : Node name based on cable configuration found : NODE0
 INFO : Check JBOD Nickname
 SUCCESS : JBOD Nickname set correctly : Oracle Database Appliance - E0
 INFO : The details for Storage Topology Validation can also be found in log
file=/opt/oracle/oak/log/<hostname>/storagetopology/StorageTopology-2014-07-03-08:
57:31_7661_15914.log
```

Oracle Database Appliance Configuration Error Messages

If you encounter errors while configuring Oracle Database Appliance, then review the following messages and actions:

Error Encountered in Step 11 Validation VIP appears to be up on the network

Cause: This message is most likely to occur when you attempt to redeploy the End-User Bundle without cleaning up a previous deployment. This error occurs because an existing VIP is configured for the addresses assigned to Oracle Database Appliance.

Action: Run cleanupDeploy.pl on Node 0, and then restart Oracle Appliance Manager.

Error "CRS-4402: The CSS daemon was started in exclusive mode but found an active CSS daemon on node oda2-1, number 1, and is terminating"

Cause: This error occurs when the Oracle Grid Infrastructure CSS daemon attempts to start the node as a standalone cluster node, but during startup discovers that the other cluster node is running, and changes to cluster mode to join the cluster.

Action: Ignore this error

Installation requires partitioning of your hard drive

Cause: This message occurs on a node if one of the two operating system disks is not installed, but you are attempting to reimage the operating system.

Action: Ensure that both operating system disks are installed and are available.

Machine Check Exception ...This is not a software problem

Cause: There is a hardware system error.

Action: Log in to the Oracle ILOM Remote Console to determine the specific hardware error.

No volume control GStreamer plugins and/or devices found

Cause: Operating system plug-ins required for sound cards for the Oracle ILOM remote redirection console are not installed.

Action: Ignore this message. You do not require volume control for the console.

Reboot and Select proper Boot device Or Insert Boot Media in selected Boot device and press a key

Cause: One or both operating system disks are not available. This message occurs if you select "Default hard disk" during reimaging the system, but that disk is not available.

Action: Ensure that both operating system disks are installed and are available.

The AoDB Linux installation tree in that directory does not seem to match your boot media

Cause: This message occurs on a node if both operating disks are installed, and you choose to reimage the operating system disks. If you select "Default (use BIOS settings)" as your imaging option, but one or both of the disks is not available.

Action: Ensure that both operating system disks are available for use.

ERROR: Gateway IP is not pingable

Cause: On Windows platforms, the Oracle Appliance Manager configurator uses the echo service on port 7 to contact the gateway. If the echo service is disabled, possibly for security reasons, the ping fails.

Action: Run the native platform ping command. If the ping is successful, then the configurator validation output can be ignored.

ACFS Resources Failed to Start After Applying 2.2 Infra Patch

Cause: Oracle Database Appliance operating system upgrade includes upgrade of Oracle Enterprise Linux to Oracle Unbreakable Enterprise Kernel (Oracle UEK). Since Oracle Automatic Storage Management Cluster File System (ACFS) is not supported on all versions of Oracle Linux, a successful upgrade of the operating system may effectively disable Oracle ACFS.

Upgrade to Oracle Database Appliance 2.2 has three options: `–infra`, `–gi`, and `–database`. The `–infra` option includes upgrade from Oracle Enterprise Linux to Oracle UEK. Before the `–infra` upgrade to 2.2, the operating system is Oracle Enterprise Linux with 11.2.0.2.*x* Grid Infrastructure. After the `–infra` upgrade, the operating system is Oracle UEK and 11.2.0.2.*x* ACFS, which is not compatible with Oracle UEK.

For example, upgrade to Oracle Linux 2.6.32-300.11.1.el5uek causes `reco.acfsvol.acfs` and `ora.registry.acfs` to temporarily go to an OFFLINE state, because 2.6.32-300.11.1.el5uek does not support Oracle 11.2.0.2.*x* ACFS. However, when Oracle Grid Infrastructure is upgraded to 11.2.0.3.2, these components are online again.

Action: Upgrade to Oracle Database Appliance 2.2 with the `–gi` option. This version of the software includes Oracle Grid Infrastructure 11.2.0.3.2, which includes Oracle ACFS modules that works with Oracle UEK.

For more information, see My Oracle Support note 1369107.1:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1369107.1

Preparing Log Files for Oracle Support Services

If you have a system fault that requires help from Oracle Support Services, you might need to provide log records. Collect log file information by running the `oakcli manage` command. This command consolidates information from log files stored on Oracle Database Appliance into a single log file for use by Oracle Support Services. The location of the file is specified in the command output.

Additional Troubleshooting Tools and Commands

This section describes additional tools and commands to diagnose and troubleshoot problems with Oracle Database Appliance, some of which are specific to Oracle Database Appliance while others are tools for all clustered systems. The section provides information about the following resources:

- Oracle Appliance Manager Tools for Configuration Auditing and Disk Diagnosis
- Trace File Analyzer Collector

Oracle Appliance Manager Tools for Configuration Auditing and Disk Diagnosis

Oracle Appliance Manager provides access to a number of sophisticated monitoring and reporting tools, some of them derived from standalone tools that require their

own syntax and command sets. The following list briefly describes the ORAchk command and the disk diagnostic tool:

- ORAchk

  The ORAchk Configuration Audit Tool audits important configuration settings for Oracle RAC two node deployments in categories such as:

  - Operating system kernel parameters, packages, and so on
  - RDBMDS
  - Database parameters and other database configuration settings
  - CRS/Grid infrastructure
  - ASM

  ORAchk is system-aware and checks for best practices, for example, that are specific to Oracle Database Appliance when run by Oracle Appliance Manager. To explore ORAchk on Oracle Database Appliance use the `oakcli orachk -h` command. Find more details about ORAchk at https://support.oracle.com/epmos/faces/DocContentDisplay?id=1268927.2.

- Disk Diagnostic Tool

  Use the Disk Diagnostic Tool to help identify the cause of disk problems. The tool produces a list of fourteen disk checks for each node. To run the tool, enter the following command:

  ```
  # oakcli stordiag eshelf_pd_unit
  ```

Trace File Analyzer Collector

Trace File Analyzer (TFA) Collector simplifies diagnostic data collection on Oracle Clusterware/Grid Infrastructure and Oracle RAC systems. TFA behaves in a similar manner to the ion utility packaged with Oracle Clusterware. Both tools collect and package diagnostic data. However, TFA is much more powerful than ion because TFA centralizes and automates the collection of diagnostic information.

TFA provides the following key benefits and options:

- Encapsulation of diagnostic data collection for all CRS/GI and Oracle RAC components on all cluster nodes into a single command executed from a single node
- Option to "trim" diagnostic files during data collection to reduce data upload size
- Options to isolate diagnostic data collection to a given time period and to a particular product component, such as ASM, RDBMS, or Clusterware
- Centralization of collected diagnostic output to a single node in Oracle Database Appliance, if desired
- On-Demand Scans of all log and trace files for conditions indicating a problem
- Real-Time Scan Alert Logs for conditions indicating a problem (DB Alert Logs, ASM Alert Logs, Clusterware Alert Logs, etc.)

> **See Also:** My Oracle Support note "TFA Collector- Tool for Enhanced Diagnostic Gathering" at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1513912.1

Oracle Database Appliance Hardware Monitoring Tool

The Oracle Database Appliance Hardware Monitoring Tool, implemented with the Oracle Appliance Manager `show` command, displays the status of different hardware components in Oracle Database Appliance server nodes. Use the tool on bare metal and on virtualized systems.

See the list of monitored components in the output of the `oakcli show -h` command.

> **See Also:** Chapter 5 for detailed information about all Oracle Appliance Manager commands including `oakcli show`

```
oakcli show power

NAME            HEALTH HEALTH DETAILS PART_NO. SERIAL_NO.          LOCATION
INPUT POWER OUTPUT POWER INLET TEMP      EXHAUST TEMP

Power Supply_0  OK     -              7047410   476856F+1242CE0020 PS0
Present     88 watts     31.250 degree C 34.188 degree C
Power Supply_1  OK     -              7047410   476856F+1242CE004J PS1
Present     66 watts     31.250 degree C 34.188 degree C
```

> **Note:** Upon initial startup of ODA_BASE on Oracle Database Appliance Virtualized Platform, the Oracle Database Appliance Server Hardware Monitoring Tool is enabled and collects base statistics for about 5 minutes. During this time, the tool displays a "Gathering Statistics…" message.

The information reported by the Oracle Database Appliance Hardware Monitoring Tool is only for the node on which you run the command. Details in the output depend on the component you select to review. The following example shows the output for the power subsystem on the current node:

5

Oracle Appliance Manager Command-Line Interface

This appendix describes the content and use of the Oracle Appliance Manager command-line interface, or the `oakcli` commands, also known as known as the Oracle Appliance Kit Command Line Interface, or OAKCLI. The current set of commands along with their syntax and usage notes are included, as well as examples of many of the commands. This appendix contains the following sections:

About Oracle Appliance Manager Command-Line Interface

Use Oracle Appliance Manager command-line interface to perform Oracle Database Appliance management tasks such deploying the software, configuring core keys, applying patches, monitoring and troubleshooting, managing virtual machines, and creating Oracle Database homes and databases. Specific tasks that you can complete with Oracle Appliance Manager command-line interface include:

- Applying the core configuration key
- Configuring the network for Oracle Database Appliance deployment
- Copying the deployment configuration file
- Deploying Oracle Database Appliance
- Locating a disk on Oracle Database Appliance
- Managing the Oracle Database Appliance repository
- Managing the Oracle Database Appliance diagnostics collection
- Updating Oracle Database Appliance
- Unpacking packages into the Oracle Appliance Manager command-line interface repository
- Validating Oracle Database Appliance

Depending on your version of Oracle Appliance Manager and your hardware, some of the commands described in this Appendix could be unavailable. To see which Oracle Appliance Manager command-line interface commands are supported on your version of Oracle Appliance Manager and your hardware, enter the following command: `oakcli -h`.

Oracle Appliance Manager Command-Line Interface Operational Notes

Usage Information

Oracle Appliance Manager command-line interface is in the following directory:

```
/opt/oracle/oak/bin/oakcli
```

The `root` user account should have the `oakcli` environment variable defined as this directory's path name.

Oracle Database Appliance maintains logs of `oakcli` command output in the following directory.

```
/opt/oracle/oak/log/hostname/client/oakcli.log
```

Using Oracle Appliance Manager Command-Line Interface Help

Run the following command to see the usage information for all `oakcli` commands:

```
oakcli -h
```

Run the following command to see detailed help about a specific `oakcli` command:

```
oakcli command -h
```

Run the following command to see detailed help about a specific `oakcli` command's objects and its options:

```
oakcli command object -h
```

Privileges and Security

You should typically use Oracle Appliance Manager command-line interface when logged into Oracle Database Appliance as the `root` user. If you are not logged in as `root`, then you will be unable to complete particular tasks. For example, you could view storage information but not modify the storage configuration.

Oracle Appliance Manager Command-Line Interface Command Reference

Oracle Appliance Manager command-line interface commands and parameters are not case sensitive. An `oakcli` command uses the following command syntax:

```
oakcli command object [parameters]
```

In `oakcli` syntax:

- *command* is a verb such as `show`, `locate`, `apply`, and so on
- *object* (also known as a noun) is the target or object on which the `oakcli` command performs the operation, such as a disk or controller. You can also use object abbreviations.
- *parameters* extend the use of a preceding command combination to include additional options for the command. Parameter names are preceded with a dash, for example, `-h`, which is the help parameter available with every command.

The remainder of this chapter contains syntax and other details about the `oakcli` commands available in this release, sorted alphabetically as shown in the following table.

Table 5–1 Oracle Appliance Manager Command-Line Interface Commands

Command	Description
oakcli apply on page 5-3	Reconfigures Oracle Database Appliance core capacity
oakcli clone on page 5-4	Clones virtual components
oakcli configure on page 5-6	Configures Oracle Database Appliance components
oakcli copy on page 5-14	Prepares a copy of the configuration file for use during deployment
oakcli create on page 5-14	Creates Oracle Database Appliance components
oakcli delete on page 5-23	Removes Oracle Database Appliance components
oakcli deploy on page 5-28	Deploys Oracle Database Appliance
oakcli diskwritecache on page 5-28	Manages disk write cache
oakcli import vmtemplate on page 5-29	Imports a virtual machine template
oakcli locate on page 5-30	Locates a disk
oakcli manage diagcollect on page 5-31	Collects diagnostic statistics and information, primarily for use when working with Oracle Support
oakcli modify on page 5-32	Add, updates, or removes a network from a virtual machine or template configuration
oakcli orachk on page 5-33	Audits configuration settings
oakcli resize dbstorage on page 5-34	Resizes the space used for an ACFS storage structure.
oakcli restart oda_base on page 5-35	Restarts ODA_BASE on the local node
oakcli show on page 5-35	Displays information about various Oracle Database Appliance components
oakcli show vmconsole on page 5-50	Opens a GUI VM console for a virtual machine.
oakcli start on page 5-50	Starts a Domain U or ODA_BASE virtual machine
oakcli stop on page 5-52	Stops a Domain U or ODA_BASE virtual machine
oakcli stordiag on page 5-53	Runs tests on a storage shelf or storage expansion shelf device
oakcli test asr on page 5-54	Checks if Oracle Auto Service Request (Oracle ASR) is functioning properly
oakcli unpack on page 5-54	Unpacks the given package to the Oracle Appliance Manager command-line interface repository
oakcli update on page 5-54	Updates Oracle Database Appliance
oakcli upgrade on page 5-55	Upgrades one or more databases to a newer version
oakcli validate on page 5-56	Validates Oracle Database Appliance

oakcli apply

Use the `oakcli apply` command to reconfigure your Oracle Database Appliance core capacity.

Syntax

Use the following syntax where *`core_config_key_file`* is the full path name of a configuration key file generated on My Oracle Support and copied to Oracle Database Appliance:

```
oakcli apply core_configuration_key core_config_key_file [-h]
```

Parameters

Table 5–2 oakcli apply Command Parameters

Parameter	Description
`-h`	Displays help
core_config_key_file	Identifies the full path and name of the configuration key file

Usage Notes

- Run the `oakcli apply` command from the first node in Oracle Database Appliance as `root`.
- After you run the `oakcli apply` command, Oracle Database Appliance reboots both the nodes.

Example

Reconfigure Oracle Database Appliance with a new core count using the configuration key file `/tmp/set8cores.conf`:

```
oakcli apply core_configuration_key /tmp/set8cores.conf
..................done
INFO: Cluster will be rebooted after applying the core_configuration_key
successfully
INFO: ................................

INFO: Applying core_configuration_key on '192.0.2.21'
INFO: ....................................................................
INFO: Running as root: /usr/bin/ssh -1 root 192.0.2.21 /tmp/tmp_lic_exec.pl
INFO: Running as root: /usr/bin/ssh -1 root 192.0.2.21 /opt/oracle/oak/bin/oakcli
enforce core_configuration_key /tmp/.lic_file
INFO: Applying core_configuration_key on '192.0.2.20'

INFO: ....................................................................
INFO: Running as root: /usr/bin/ssh -1 root 192.0.2.20 /tmp/tmp_lic_exec.pl
INFO: Running as root: /usr/bin/ssh -1 root 192.0.2.20 /opt/oracle/oak/bin/oakcli
enforce core_configuration_key /tmp/.lic_file
INFO: Applying core_configuration_key on '192.0.2.20'
```

oakcli clone

Use the `oakcli clone` command to clone virtual machines or virtual disks.

Table 5–3 oakcli clone Command Summary

Command	Description
oakcli clone vdisk on page 5-4	Creates a clone of an existing virtual disk.
oakcli clone vm on page 5-5	Creates a clone or snapshot clone of an existing virtual machine.

oakcli clone vdisk

Use the `oakcli clone vdisk` command to create clones of virtual disks.

Syntax

Use the following syntax for the `oakcli clone vdisk` command to create a clone of virtual disk:

```
oakcli clone vdisk new_vdisk_name -repo repo_name -src source_vdisk_name [-h]
```

where new_vdisk_name is the name given to the clone virtual disk, repo_name is the name of the repository source for the virtual disk being cloned, source_vdisk_name is the name of the virtual disk being cloned, and -h optionally displays the help usage for this command.

Example

Clone a virtual disk named *my_vdisk2* from an existing virtual disk named vdisk1 that is stored in the repository named *vrepo1*:

```
oakcli clone vdisk my_vdisk2 -repo vrepo1 -src vdisk1
```

oakcli clone vm

Use the `oakcli clone vm` command to create clones and snap clones of virtual machines.

Syntax

Use the following syntax for the `oakcli clone vm` command to create a virtual machine from a template:

```
oakcli clone vm vm_name -vmtemplate template_name -repo repo_name [-node 0|1]
```

Use the following syntax for the `oakcli clone vm` command to create a snapshot clone of an existing virtual machine:

```
oakcli clone vm vm_name -vm src_name -snap
```

Use the following syntax for the `oakcli clone vm` command to create a virtual machine snapshot from a template:

```
oakcli clone vm vm_name -vmtemplate template_name -snap
```

Parameters

Table 5–4 oakcli clone vm Command Parameters and Variables

Parameter	Description
vm_name	*vm_name* is the name given to the cloned virtual machine.
-vmtemplate *vmtemplate_name*	*vmtemplate_name* is the name of the template containing the virtual machine that you want to clone.
-repo *repo_name*	*repo_name* is the name of the repository that contains the template being cloned.
-node 0 | 1	Identifies the Oracle Database Appliance node that contains the shared repository from which the virtual machine is to be cloned. The -node parameter must be used when cloning from a shared repository and is invalid for non-shared repositories.
-vm *src_name*	src_name is the name of the virtual machine that is to be cloned.

Table 5–4 (Cont.) oakcli clone vm Command Parameters and Variables

Parameter	Description
-snap	Creates a snapshot of the source virtual machine or virtual machine template.

Example 1

Create a virtual machine image named *myol15u_test* from the virtual machine template named *myol15u* which is stored in the Node 0 repository:

```
oakcli clone vm myol15u_test -vmtemplate myol5u -repo odarepo1
```

Example 2

Create a virtual machine image named *myol6u_test* from the virtual machine template named *myol6u_15gb1*, which is stored in the shared repository named repo2 on Node 0:

```
oakcli clone vm myol6u_test -vmtemplate myol6u_15gb1 -repo repo2 -node 0
```

oakcli configure

Use the `oakcli configure` command to configure components on Oracle Database Appliance.

Table 5–5 oakcli configure Command Summary

Command	Description
oakcli configure asr on page 5-6	Configures Oracle Auto Service Request for Oracle Database Appliance
oakcli configure cpupool on page 5-7	Configures a CPU pool
oakcli configure firstnet on page 5-7	Configures initial network information
oakcli configure network on page 5-8	Configures the network after either replacing a network card or swapping the public network from copper to fiber and vice versa
oakcli configure oda_base on page 5-8	Configures CPU core count, memory allocation, virtual local area networks for ODA_BASE
oakcli configure repo on page 5-9	Configure a shared repository by increasing its size
oakcli configure vm on page 5-10	Configures a virtual machine
oakcli configure vmtemplate on page 5-12	Configures a virtual machine template

oakcli configure asr

Use the `oakcli configure asr` command to configure Oracle Auto Service Request (Oracle ASR) on Oracle Database Appliance.

Syntax

```
oakcli configure asr [-h]
```

where `-h` will display online help for the command.

Usage Notes

- The `oakcli configure asr` command initiates an interactive script that requests the following information to implement Oracle ASR on your Oracle Database Appliance:
  - Action to be performed (internal or external install, deinstall, or reconfigure)
  - PROXY server name, port, user ID, and password
  - ASR user ID and password
  - ASR Manager IP and port

oakcli configure cpupool

Use the `oakcli configure cpupool` command to configure a CPU pool on one Oracle Database Appliance Virtualized Platform node.

Syntax

```
oakcli configure cpupool poolname -numcpu cpu_count -node nodenum [-h]
```

Parameters

Table 5–6 oakcli configure cpupool Command Parameters

Parameter	Description
poolname	Uniquely names the CPU pool.
-numcpu *cpu_count*	Defines the number of CPUs for the CPU pool.
-node *nodenum*	Defines the node where the CPU pool will be created (0 or 1).
`-h`	(Optional) Displays the help usage for this command.

Examples

Example 1 Configuring a Two Core CPU Pool:

Configure a CPU pool with two cores on Node 1 of Oracle Database Appliance Virtualized Platform:

```
oakcli configure cpupool twonode -numcpu 2 -node 1
```

oakcli configure firstnet

Use the `oakcli configure firstnet` command to configure an initial network on Oracle Database Appliance that enables you to download deployment software.

Syntax

```
oakcli configure firstnet
```

Examples

Example 1 Configuring the Initial Network on Bare Metal Oracle Database Appliance

The following command creates an initial network on a new Oracle Database Appliance using the interactive script, as shown in the following:

```
oakcli configure firstnet
```

```
Select the interface to  configure network on [bond0 bond1 bond2 xbond0]:bond0
Configure DHCP on bond0?(yes/no):no
       INFO: Static configuration selected
       Enter the IP address to configure:192.0.2.18
       Enter the netmask address to configure:255.255.252.0
       Enter the gateway address to configure:192.0.2.1
Plumbing the IPs now
Restarting the network
::::::::::::::::::
```

Note: Oracle recommends using the `oakcli configure firstnet` command only one time on Oracle Database Appliance. Subsequent use after configuring the initial network can cause unpredictable changes to your network settings.

oakcli configure network

Use the `oakcli configure network` command to configure the network after either replacing a network card or swapping the public network from copper to fiber and vice versa. The meaning of the command changes depending on which parameter you use.

Syntax

```
oakcli configure network [-changeNetCard|-publicNet]
```

Parameters

Table 5–7 oakcli configure network Command Parameters

Parameter	Description
`-changeNetCard`	Configures the network card after it has been replaced. You must run the `oakcli configure network -changeNetCard` command on each node if the network card has been replaced on each node. This parameter is supported on all Oracle Database Appliance hardware models. This command must be executed from dom0 on virtualized platforms
`-publicNet`	Used to swap the public network from copper to fiber and vice versa. You must run the `oakcli configure network -publicNet` command on each node, and this requires the stack to be down. The stack includes GI and RDBMS. When running in Virtualization mode, the virtual machines need to be down and may need some configuration changes, especially if they are using VLANs. This parameter is only supported on Oracle Database Appliance X4-2 and X5-2 hardware models.
`-h`	(Optional) Displays the help usage for this command.

oakcli configure oda_base

Use the `oakcli configure oda_base` command to change the CPU core count assigned to ODA_BASE or to add or remove virtual local area networks an initial network on Oracle Database Appliance that enables you to download deployment software.

- To change the CPU core count assigned to ODA_BASE
- To change the amount of memory assigned to ODA_BASE
- To add virtual local area networks to ODA_BASE

- To remove virtual local area networks to ODA_BASE

Syntax

```
oakcli configure oda_base
```

Examples

Example 1 Changing the CPU core count in ODA_BASE:

Change the CPU core count from six to eight in ODA_BASE:

```
# oakcli configure oda_base
Core Licensing Options:
        1. 2 CPU Cores
        2. 4 CPU Cores
        3. 6 CPU Cores
        4. 8 CPU Cores
        5. 10 CPU Cores
        6. 12 CPU Cores
        Current CPU Cores       :6
        Selection[1 : 6](default 12 CPU Cores) : 4
        ODA base domain memory in GB(min 8, max 88)(Current Memory 64G)[default
32]     :
INFO: Using default memory size i.e. 32 GB
Additional vlan networks to be assigned to oda_base? (y/n) [n]:
Vlan network to be removed from oda_base (y/n) [n]
INFO: Node 0:Configured oda base pool
INFO: Node 1:Configured oda base pool
INFO: Node 0:ODA Base configured with new memory
INFO: Node 0:ODA Base configured with new vcpus
INFO: Changes will be incorporated after the domain is restarted on Node 0
INFO: Node 1:ODA Base configured with new memory
INFO: Node 1:ODA Base configured with new vcpus
INFO: Changes will be incorporated after the domain is restarted on Node 1
```

oakcli configure repo

Use the `oakcli configure repo` command to increase the size of a shared repository.

Syntax

```
oakcli configure repo reponame -incsize size [M|G]
```

where reponame is the name of the shared repository and size is a number that can be followed by M to define the size as megabytes or by G to define as size as gigabytes.

Examples

Example 1 Increasing the size of a shared repository:

Change the size of the repo1 shared repository by 2 gigabytes:

```
# oakcli configure repo repo1 -incsize 2G

Configured Shared Repo : repo1 with new size 3712.0.
```

oakcli configure vm

Use the `oakcli configure vm` command to configure a virtual machine on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli configure vm name [-vcpu cpucount -maxvcpu maxcpu -cpuprio priority
-cpucap cap -memory memsize -maxmemory max_memsize -os sys -keyboard lang -mouse
mouse_type -domain dom -network netlist -autostart astart -disk disks -bootoption
bootstrap -cpupool pool -prefnode 0|1 -failover true|false]
```

Parameters

Table 5–8 oakcli configure vm Command Parameters

Parameter	Description
name	The name assigned to the virtual machine.
-vcpu *cpucount*	Number of nodes assigned to the virtual machine. This number depends on your Oracle Database Appliance configuration: ■ On Oracle Database Appliance X5-2, the range is from 1 to 72 ■ On Oracle Database Appliance X4-2, the range is from 1 to 48 ■ On Oracle Database Appliance X3-2, the range is from 1 to 32 ■ On Oracle Database Appliance, the range is 1 to 24
-maxvcpu *maxcpu*	Maximum number of CPUs that the virtual machine can consume. This number depends on your Oracle Database Appliance configuration: ■ On Oracle Database Appliance X5-2, the range is from 1 to 72 ■ On Oracle Database Appliance X4-2, the range is from 1 to 48. ■ On Oracle Database Appliance X3-2, the range is from 1 to 32 ■ On Oracle Database Appliance, the range is 1 to 24
-cpuprio *priority*	Priority for CPU usage, where larger values have higher priority (1 - 256).
-cpucap *cap*	Percentage of a CPU the virtual machine can receive (1 - 100).
-memory *memsize*	Amount of memory given to the virtual machine: (1-88)G or (1-90112)M. Default is M.
-maxmemory *max_memsize*	Maximum amount of memory allowed for the virtual machine: (1-88)G or (1-90112)M. Default is M.
-os *sys*	Operating system used by the virtual machine (WIN_2003, WIN_2008, WIN_7, WIN_VISTA, OTHER_WIN, OL_4, OL_5, OL_6, RHL_4, RHL_5, RHL_6, LINUX_RECOVERY, OTHER_LINUX, SOLARIS_10, SOLARIS_11, OTHER_SOLARIS, or NONE).
-keyboard *lang*	Keyboard used by virtual machine (en-us, ar, da, de, de-ch, en-gb, es, et, fi, fo, fr, fr-be, fr-ca, hr, hu, is, it, ja, lt, lv, mk, nl, n--be, no, pl, pt, pt-br, ru, sl, sv, th, or tr).

Table 5–8 (Cont.) oakcli configure vm Command Parameters

Parameter	Description
`-mouse` *mouse_type*	Mouse type used by the virtual machine (OS_DEFAULT, PS2_MOUSE, USB_MOUSE, or USB_TABLET).
`-domain` *dom*	Domain type from the following options: ■ Hardware virtualized guest (XEN_HVM) - The kernel or operating system is not virtualization-aware and can run unmodified. - Device drivers are emulated. ■ Para virtualized guest (XEN_PVM) - The guest is virtualization-aware and is optimized for a virtualized environment. - PV guests use generic, idealized device drivers. ■ Hardware virtualized guest (XEN_HVM_PV_DRIVERS) The PV drivers are hypervisor-aware and significantly reduce the overhead of emulated device input/output.
`-network` *netlist*	MAC address and list of networks used by the virtual machine.
`-autostart` *astart*	Startup option for virtual machine (always, restore, or never).
`-disk` *disks*	List of disks (slot, disktype, and content) used by virtual machine.
`-bootoption` *bootstrap*	Boot option used to bootstrap virtual machine (PXE, DISK, or CDROM).
`-cpupool` *pool*	Assign the named CPU pool to virtual machine.
`-prefnode 0|1`	Define the node, 0 or 1, where the virtual machine should attempt to start. This parameter is only valid for virtual machines created in shared repositories.
`-failover true|false`	Allow (use the keyword "true") or disallow (use the keyword "false") the virtual machine to start or restart on a node other than the node defined by the `-prefnode` parameter. This parameter is only valid for virtual machines created in shared repositories.

Usage Notes

- All of the parameters, except for `name`, are optional.
- You must include at least one optional parameter for the command to work.

> **See Also:** Oracle VM Release 3.1 documentation at `http://docs.oracle.com/cd/E27300_01` for more information about the options in the preceding table. For example, see `http://docs.oracle.com/cd/E27300_01/E27309/html/vmusg-ovm-vms.html` for details about the `-domain` *dom* options

Examples

Example 1 Changing virtual machine count and virtual memory size for a virtual machine

Enter the following command to change the number of virtual CPUs to 3 and the virtual memory size to 4GB in a virtual machine named `sample_odarep01`:

```
oakcli configure vm sample_odarep01 -vcpu 3 -memory 4196
```

oakcli configure vmtemplate

Use the `oakcli configure vmtemplate` command to configure a virtual machine template on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli configure vmtemplate name [-vcpu cpucount -maxvcpu maxcpu -cpuprio
 priority -cpucap cap -memory memsize -maxmemory max_memsize -os sys -keyboard
 lang -mouse mouse_type -domain dom -network netlist -disk disks]
```

Parameters

Table 5–9 oakcli configure vmtemplate Command Parameters

Parameter	Description
name	The name assigned to the virtual machine template.
-vcpu *cpucount*	Number of nodes assigned to virtual machines cloned from the template. ■ On Oracle Database Appliance X5-2, the range is from 1 to 72 ■ On Oracle Database Appliance X4-2, the range is from 1 to 48. ■ On Oracle Database Appliance X3-2, the range is from 1 to 32 ■ On Oracle Database Appliance, the range is 1 to 24
-maxvcpu *maxcpu*	Maximum number of CPUs that virtual machines cloned from the template can consume. ■ On Oracle Database Appliance X5-2, the range is from 1 to 72 ■ On Oracle Database Appliance X4-2, the range is from 1 to 48. ■ On Oracle Database Appliance X3-2, the range is from 1 to 32 ■ On Oracle Database Appliance, the range is 1 to 24
-cpuprio *priority*	Priority for CPU usage, where larger values have higher priority (1 - 256).
-cpucap *cap*	Percentage of a CPU that virtual machines cloned from the template can receive (1 - 100).
-memory *memsize*	Amount of memory given to virtual machines cloned from the template (1G - 88 G or 1M - 90112M).
-maxmemory *max_memsize*	Maximum amount of memory allowed for virtual machines cloned from the template.

Table 5–9 (Cont.) oakcli configure vmtemplate Command Parameters

Parameter	Description
`-os` *sys*	Operating system used by virtual machines cloned from the template (WIN_2003, WIN_2008, WIN_7, WIN_VISTA, OTHER_WIN, OL_4, OL_5, OL_6, RHL_4, RHL_5, RHL_6, LINUX_RECOVERY, OTHER_LINUX, SOLARIS_10, SOLARIS_11, OTHER_SOLARIS, or NONE).
`-keyboard` *lang*	Keyboard used by virtual machines cloned from the template (en-us, ar, da, de, de-ch, en-gb, es, et, fi, fo, fr, fr-be, fr-ca, hr, hu, is, it, ja, lt, lv, mk, nl, n--be, no, pl, pt, pt-br, ru, sl, sv, th, or tr).
`-mouse` *mouse_type*	Mouse type used by virtual machines cloned from the template (OS_DEFAULT, PS2_MOUSE, USB_MOUSE, or USB_TABLET).
`-domain` *dom*	Domain type from the following options: ■ Hardware virtualized guest (XEN_HVM) - The kernel or operating system is not virtualization-aware and can run unmodified. - Device drivers are emulated. ■ Para virtualized guest (XEN_PVM) - The guest is virtualization-aware and is optimized for a virtualized environment. - PV guests use generic, idealized device drivers. ■ Hardware virtualized guest (XEN_HVM_PV_DRIVERS) The PV drivers are hypervisor-aware and significantly reduce the overhead of emulated device input/output.
`-network` *netlist*	MAC address and list of networks used by virtual machines cloned from the template.
`-disk` *disks*	List of disks (slot, disktype, and content) used by virtual machines cloned from the template.

Usage Notes

- All of the parameters, except for `name`, are optional.
- You must include at least one optional parameter for the command to work.

> **See Also:** Oracle VM Release 3.1 documentation at `http://docs.oracle.com/cd/E27300_01` for more information about the options in the preceding table. For example, see `http://docs.oracle.com/cd/E27300_01/E27309/html/vmusg-ovm-vms.html` for details about the `-domain` *dom* options

Examples

Example 1 Configure a Virtual Machine Template

Set values for the following configuration values in the virtual machine template named *myol5u7_10gb*:

- Number of CPUs assigned when the virtual machine starts up (vcpu)
- Maximum number of CPUs that can be assigned to the virtual machine (maxvcpu)

- Maximum percentage of a CPU's capacity that will be assigned to the virtual machine (cpucap)
- Amount of memory assigned when the virtual machine starts up (memory)
- Maximum amount of memory that can be assigned to the virtual machine (maxmemory)
- Network used to access the virtual machine (network)
- Operating system used by the virtual machine (os)

```
oakcli configure vmtemplate myol5u7_10gb
-vcpu 2 -maxvcpu 4 -cpucap 40 -memory 1536M -maxmemory 2G
-network "['type=netfront,bridge=net1']" -os OTHER_LINUX
```

oakcli copy

Use the `oakcli copy` command to prepare a copy of the configuration file for use during the configuration of Oracle Database Appliance.

Syntax

Run the `oakcli copy` command as follows, where *`absolute_conf_file`* is the full path name of an existing configuration file:

```
oakcli copy -conf absolute_conf_file [-h]
```

Parameters

Table 5–10 oakcli copy Command Parameters

Parameter	Description
`-conf` *`absolute_conf_file`*	Specifies the full path name of the configuration file.
`-h`	(Optional) Displays help

Examples

Example 1 Preparing a copy of the configuration file

If you created a configuration file previously and copied this file to Oracle Database Appliance, then prepare the configuration file to be used during the configuration process. For example, if you copied the file `myserver1.conf` to `/tmp`, then enter the following command:

```
oakcli copy -conf /tmp/myserver1.conf
```

oakcli create

Use the `oakcli create` command to create components on Oracle Database Appliance.

Table 5–11 oakcli create Command Summary

Command	Description
oakcli create cpupool on page 5-15	Creates a new CPU pool
oakcli create database on page 5-15	Creates a new database
oakcli create dbhome on page 5-17	Creates a new database home

Table 5–11 (Cont.) oakcli create Command Summary

Command	Description
oakcli create dbstorage on page 5-18	Creates a new ACFS storage structure
oakcli create db_config_params on page 5-19	Creates a database configuration file
oakcli create repo on page 5-20	Creates a virtual local area network on a Oracle Database Appliance Virtualized Platform node
oakcli create snapshotdb on page 5-21	Creates a snapshot database from an existing database
oakcli create vdisk on page 5-22	Creates a virtual disk in a shared repository on Oracle Database Appliance Virtualized Platform
oakcli create vlan on page 5-22	Creates a virtual local area network on a Oracle Database Appliance Virtualized Platform node

oakcli create cpupool

Use the `oakcli create cpupool` command to create a CPU pool on one Oracle Database Appliance Virtualized Platform node.

Syntax

```
oakcli create cpupool poolname -numcpu cpu_count -node nodenum [-h]
```

Parameters

Table 5–12 oakcli configure cpupool Command Parameters

Parameter	Description
poolname	Uniquely names the CPU pool.
-numcpu *cpu_count*	Defines the number of CPUs for the CPU pool.
-node *nodenum*	Defines the node where the CPU pool will be created (0 or 1).
`-h`	(Optional) Displays the help usage for this command.

Examples

Example 1 Creating a Two Core CPU Pool

Create a CPU pool with two CPUs on Node 1 of Oracle Database Appliance Virtualized Platform:

```
oakcli create cpupool twonode -numcpu 2 -node 1
```

oakcli create database

Use the `oakcli create database` command to create additional databases on Oracle Database Appliance.

When you run `oakcli create database`, the command prompts you for further inputs. See "Creating a new database showing prompts" in the Examples section.

Syntax

```
oakcli create database -db db_name [[[-oh home] | [-version version]] [-params
```

params_file] [-cdb]]

Parameters

Table 5–13 oakcli create database Command Parameters

Parameter	Description
-db *db_name*	*db_name* is the name of the database that you want to create.
-oh *home*	(Optional) *home* is the name of an existing Oracle home to use when creating the database. By default, the command creates a new database home.
-version *version*	(Optional) *version* is the version of the database that you want to create. By default, Oracle Database Appliance uses the highest version that you have downloaded.
-params *params_file*	(Optional) *params_file* is the configuration file. By default, Oracle Database Appliance uses the default configuration file.
-cdb	(Optional) create the database as a container database

Usage Notes

- The -oh and the -version parameters are mutually exclusive. Attempting to use both in the same command will generate an error.
- When a database is created without identifying an Oracle Home, a new Oracle Home is created, using a standard naming convention, for example, OraDb11203_home3. The number at the end of the name is incremented by one for each new home created with the same version number.
- When you upgrade the database, infrastructure, and Oracle Grid Infrastructure, you must specify an existing home to create a new database.
- If you try to create a database using the option -version *version* before downloading and unpacking the the specific version DB clone files, then the command will fail.
- You can create configuration files with the oakcli create db_config_params command.
- The prompt "Do you want to keep the data files on FLASH storage: [Y | N] " is only shown if you choose the OLTP database type and if there is some free space available on flash storage. See "Improving I/O Performance for Database Files" on page 2-8 for more information about this option.

Examples

Creating a new database showing prompts

When you run oakcli create database you are prompted for several inputs. The options listed for each input requested depend on the platform you run the command on. For example, Database Class options 9 and 10 are only available on Oracle Database Appliance X5-2.

```
oakcli create database -db mydb -oh OraDb12102_home1

Please enter the 'root'  password :
Please re-enter the 'root' password:

Please enter the 'oracle'  password :
Please re-enter the 'oracle' password:
```

```
Please enter the 'SYSASM'  password : (During deployment we set the
SYSASM password to 'welcome1'):
Please re-enter the 'SYSASM' password:
Please select one of the following for Database type  [1 .. 3]:
1    => OLTP
2    => DSS
3    => In-Memory
1
Selected value is : OLTP
Please select one of the following for Database Deployment  [1 .. 3]:
1    => EE : Enterprise Edition
2    => RACONE
3    => RAC
3
Selected value is : RAC

Do you want to keep the data files on FLASH storage: [ Y | N ]?N

Specify the  Database Class (1. odb-01 '1 core, 8 GB memory'   2.
Others) [1]:2

Please select one of the following for Database Class    [1 .. 10]:
1    => odb-01s  (   1 cores ,     4 GB memory)
2    =>  odb-01  (   1 cores ,     8 GB memory)
3    =>  odb-02  (   2 cores ,    16 GB memory)
4    =>  odb-04  (   4 cores ,    32 GB memory)
5    =>  odb-06  (   6 cores ,    48 GB memory)
6    =>  odb-12  (  12 cores ,    96 GB memory)
7    =>  odb-16  (  16 cores ,   128 GB memory)
8    =>  odb-24  (  24 cores ,   192 GB memory)
9    =>  odb-32  (  32 cores ,   256 GB memory)
10    =>  odb-36  (  36 cores ,   256 GB memory)
```

Creating a new database in an existing Oracle Home

The following command creates a database called `sales1` in `OraDb11203_home2`:

```
oakcli create database -db sales1 -oh OraDb11203_home2
```

Creating a new database from a template

The following command creates a database called `sales2` from the `salesdbtemplated.dbconf` file (by appending the default file extension to the file name provided). The command also creates a new Oracle Home:

```
oakcli create database -db sales2 -params salesdbtemplate
```

Creating a new database as a container database

The following command creates a container database called `sales3`:

```
oakcli create database -db sales3 -version 12.1.0.2 -cdb
```

oakcli create dbhome

Use the `oakcli create dbhome` command to create a new database home on Oracle Database Appliance.

syntax `oakcli create dbhome [-version` *version*`] [-h]`

Parameters

Table 5–14 oakcli create dbhome Command Parameters

Parameter	Description
`-version` *version*	(Optional) *version* is the version that you want to install. If not provided, Oracle Database Appliance uses the latest available version.
`-h`	(Optional) Displays the help usage for this command

Examples

Example 1 Creating a new database home for version 11.2.0.2

The following command creates a database called `sales1` using version 11.2.0.2:

```
oakcli create dbhome -version 11.2.0.2
```

oakcli create dbstorage

Use the `oakcli create dbstorage` command to create a storage structure for migrating databases from ASM to ACFS.

Syntax

```
oakcli create dbstorage -db db_name [-cdb]
```

Parameters

Table 5–15 oakcli create dbstorage Command Parameters

Parameter	Description
`-db` *dbname*	Set up the required ACFS storage structure for the database to be created called *db_name*
-cdb	Must be passed if you are creating a multitenant container database
`-h`	(Optional) Displays the help usage for this command

Examples The oakcli create dbstorage command requests user input to determine the size of the storage structure to create as shown in this example.

```
oakcli create dbstorage -db sales

Please enter the 'root' password :
Please re-enter the 'root' password:

Please enter the 'oracle' password :
Please re-enter the 'oracle' password:

Please enter the 'SYSASM' password : (During deployment we set the SYSASM password
to 'welcome1'):
Please re-enter the 'SYSASM' password:

Specify the Database Class (1. odb-01 '1 core, 8 GB memory' 2. Others) [1]:2

Please select one of the following for Database Class [1 .. 8] :
1 => odb-01s ( 1 cores , 4 GB memory)
2 => odb-01 ( 1 cores , 8 GB memory)
3 => odb-02 ( 2 cores , 16 GB memory)
4 => odb-04 ( 4 cores , 32 GB memory)
```

```
5 => odb-06 ( 6 cores , 48 GB memory)
6 => odb-12 ( 12 cores , 96 GB memory)
7 => odb-16 ( 16 cores , 128 GB memory)
8 => odb-24 ( 24 cores , 192 GB memory)


Selected value is: odb-01s ( 1 cores , 4 GB memory)
...
```

oakcli create db_config_params

Use the `oakcli create db_config_params` command to generate a database configuration file. The configuration file is created in `/opt/oracle/oak/install/dbconf` and is given the default extension `.dbconf`.

Syntax

```
oakcli create db_config_params -conf filename -h
```

Parameters

Table 5–16 oakcli create db_config_params Command Parameters

Parameter	Description
`-conf` *`filename`*	*`filename`* is the name you want to give to the configuration file, without its pathname.
`-h`	(Optional) Displays the help usage for this command

Examples

Example 1 Creating a new database configuration file

The following command creates the database parameter file `/opt/oracle/oak/install/dbconf/newconf.dbconf`:

```
# oakcli create db_config_params -conf newconf

Please select one of the following for Database Block Size  [1 .. 4]:
1 ==> 4096
2 ==> 8192
3 ==> 16384
4 ==> 32768
2
Selected value is: 8192

Specify the  Database Language (1. AMERICAN 2. Others) [1]:
Selected value is: AMERICAN

Specify the  Database Characterset (1. AL32UTF8 2. Others) [1]:2

Please select one of the following for Database Characterset [0 .. 10] :
0 => Others
1 => AL32UTF8
2 => AR8ADOS710
3 => AR8ADOS710T
4 => AR8ADOS720
5 => AR8ADOS720T
6 => AR8APTEC715
7 => AR8APTEC715T
8 => AR8ARABICMACS
```

```
9 => AR8ASMO708PLUS
10 => AR8ASMO8X
1
Selected value is: AL32UTF8

Specify the  Database Territory (1. AMERICA 2. Others) [1]:2

Please select one of the following for Database Territory [0 .. 10] :
0 => Others
1 => ALBANIA
2 => ALGERIA
3 => AMERICA
4 => ARGENTINA
5 => AUSTRALIA
6 => AUSTRIA
7 => AZERBAIJAN
8 => BAHRAIN
9 => BANGLADESH
10 => BELARUS
3
Selected value is: AMERICA

Specify the  Component Language (1. en 2. Others) [1]:2

Please select one of the following for Component Language [0 .. 10] :
0 => Others
1 => en : English
2 => fr : French
3 => ar : Arabic
4 => bn : Bengali
5 => pt_BR : Brazilian Portuguese
6 => bg : Bulgarian
7 => fr_CA : Canadian French
8 => ca : Catalan
9 => hr : Croatian
10 => cs : Czech
1
Selected value is: en
Successfully generated the Database parameter file 'newconf'
```

oakcli create repo

Use the `oakcli create repo` command to create a new shared repository on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli create repo repo_name -size size [M|G] -dg DATA|RECO  -h
```

where *repo_name* is the name assigned to the shared repository and the parameters are described in the following table (Table 5–17).

Parameters

Table 5–17 oakcli create repo Command Parameters

Parameter	Description
`-size` *size* [M|G]	*size* is the amount of storage to be assigned to the shared repository which can be defined as megabytes, with the M option or in gigabytes with the G option.

Table 5–17 (Cont.) oakcli create repo Command Parameters

Parameter	Description
`-dg DATA|RECO`	The ASM disk group in which the shared repository is to be stored, either the DATA+ disk group or the RECO+ disk group, selected by using the DATA or RECO option respectively.
`-h`	(Optional) Displays the help usage for this command.

Usage Notes

- The `-size` parameter requires a whole number for size.
- The minimum value for *`size`* is 500 when M is used or 1 when G is used for the sizing unit.
- The default sizing unit for *`size`* is G (gigabytes).

Examples

Example 1 Creating a new shared repository

The following command creates a 25 gigabyte shared repository named `repoprod1` in the DATA+ disk group.

```
oakcli create repo repoprod1 -dg DATA -size 25
```

oakcli create snapshotdb

Use the `oakcli create snapshotdb` command to create a snapshot database from an existing database.

Syntax

```
oakcli create snapshotdb [-db snap_dbname -from dbname] | [-h]
```

where *`snap_dbname`* is the name of the snapshot database to be created, *`dbname`* is the name of the source database and the parameters are described in the following table (Table 5–18).

Parameters

Table 5–18 oakcli create snapshotdb Command Parameters

Parameter	Description
`-db`	This parameter precedes the name to be given to the new snapshot database.
`-from`	This parameter precedes the name of the database from which the snapshot database is to be built.
`-h`	(Optional) Displays the help usage for this command.

Examples

Example 1 Creating a snapshot database

The following command creates a new snapshot database, name `snapprod`, from the database named `prod`.

```
oakcli create snapshotdb database -db snapprod -from prod
```

oakcli create vdisk

Use the `oakcli create vdisk` command to create a new virtual disk in a shared repository on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli create vdisk vdisk_name -repo repository_name -size size -type shared|local
-h
```

where *vdisk_name* is the name assigned to the virtual disk that is unique within the name repository and the parameters are described in the following table (Table 5–20).

Parameters

Table 5–19 oakcli create vdisk Command Parameters

Parameter	Description
`-repo` *repository_name*	*repository_name* is the name of the shared repository where the virtual disk will be created and from which it will acquire its storage.
`-size` *size*	*size* is the amount of storage to be assigned from the shared repository to the shared disk, where the default unit is G (for gigabytes) and the minimum size is 500M.
`-type shared | local`	Sets the option of allowing the virtual disk to be shared by more than one virtual machine (`shared`) or used by only one virtual machine (`local`).
`-h`	(Optional) Displays the help usage for this command.

Examples

Example 1 Creating a new virtual disk

The following command creates a virtual disk named `t2g` in the shared repository named `repoprod1` for use by only one virtual machine at a time in that repository:

```
oakcli create vdisk t2g -repo repoprod1 -type local -size 2G
```

oakcli create vlan

Use the `oakcli create vlan` command to create a new virtual local area network (VLAN) on an Oracle Database Appliance Virtualized Platform node.

Syntax

```
oakcli create vlan vlan_name -vlanid tag_id -if interface_name -node 0|1 -h
```

where *vlan_name* is the name assigned to the VLAN and the parameters are described in the following table (Table 5–20).

Parameters

Table 5–20 oakcli create vlan Command Parameters

Parameter	Description
`-vlanid` *tag_id*	*tag_id* is a tag number, used for packet routing, from 2 to 4096 inclusive that uniquely identifies the VLAN on a node. The same tag number can be used on both nodes.

Table 5–20 (Cont.) oakcli create vlan Command Parameters

Parameter	Description
`-if interface_name`	*interface_name* is the name of the interface on which the VLAN network is created.
`-node 0 | 1`	The node on which the VLAN is created, either 1 or 2.
`-h`	(Optional) Displays the help usage for this command.

Examples

Example 1 Creating a new virtual local area network

The following command creates a VLAN named `sample10` on Node 1 using the `bond1` interface and a tag with the number `10`:

```
oakcli create vlan sample10 -vlanid 10 -if bond1 -node 1
```

Example 2 Duplicating a virtual local area network on the second node

The following command creates the same VLAN as Example 1 (named `sample10`) on Node 0:

```
oakcli create vlan sample10 -vlanid 10 -if bond1 -node 0
```

oakcli delete

Use the `oakcli delete` command to delete components from Oracle Database Appliance.

Table 5–21 oakcli delete Command Summary

Command	Description
oakcli delete cpupool on page 5-23	Deletes an existing CPU pool
oakcli delete database on page 5-24	Removes an existing database
oakcli delete dbhome on page 5-24	Deletes an existing database home
oakcli delete dbstorage on page 5-25	Deletes an ACFS storage structure
oakcli delete db_config_params on page 5-25	Deletes a database configuration file
oakcli delete repo on page 5-25	Deletes an existing shared repository
oakcli delete vdisk on page 5-26	Deletes a virtual disk from a shared repository
oakcli delete vlan on page 5-26	Deletes an existing virtual machine
oakcli delete vm on page 5-27	Deletes an existing virtual machine
oakcli delete vmtemplate on page 5-27	Deletes an existing VM template

oakcli delete cpupool

Use the `oakcli delete cpupool` command to delete a CPU pool from one Oracle Database Appliance Virtualized Platform node.

Syntax

```
oakcli delete cpupool poolname -node nodenum [-h]
```

Parameters

Table 5–22 oakcli configure cpupool Command Parameters

Parameter	Description
poolname	Names the CPU pool to be deleted.
-node *nodenum*	Defines the node from which the CPU pool will be deleted (0 or 1).
-h	(Optional) Displays the help usage for this command.

Examples

Example 1 Deleting a CPU Pool

Delete the CPU pool named *twonode* from Node 1 of Oracle Database Appliance Virtualized Platform:

```
oakcli delete cpupool twonode -node 1
```

oakcli delete database

Use the `oakcli delete database` command to delete a database from Oracle Database Appliance.

syntax `oakcli delete database -db` *db_name* `[-h]`

where and displays the help usage for this command.

Examples

Example 1 Deleting a database

The following command deletes the database named `sales1`:

```
oakcli delete database -db sales2
```

oakcli delete dbhome

Use the `oakcli delete dbhome` command to delete a database home from Oracle Database Appliance.

syntax `oakcli delete dbhome -oh` *oracle_home* `[-h]`

Parameters

Table 5–23 oakcli delete dbhome Command Parameters

Parameter	Description
oracle_home	*oracle_home* is the database home to be deinstalled.
-h	(Optional) Displays the help usage for this command

Examples

Example 1 Deleting an existing database home

The following command deletes a database home called `ora11_1`:

```
oakcli delete dbhome -oh ora11_1
```

oakcli delete dbstorage

Use the `oakcli delete dbstorage` command to delete a storage structure that was created for the purpose of migrating databases from ASM to ACFS. For example, run this command if you created a storage structure using `create dbstorage` that is no longer required.

Syntax

```
oakcli delete dbstorage -db  db_name [-cdb]
```

Parameters

Table 5–24 oakcli delete dbstorage Command Parameters

Parameter	Description
-db *dbname*	Database structure to be deleted
-cdb	Must be passed if you are deleting a multitenant container database
-h	(Optional) Displays the help usage for this command

Examples Delete a storage structure.

```
oakcli delete dbstorage -db sales
```

oakcli delete db_config_params

Use the `oakcli delete db_config_params` command to delete a database configuration file.

Syntax

```
oakcli delete db_config_params -conf filename -h
```

Parameters

Table 5–25 oakcli delete db_config_params Command Parameters

Parameter	Description
-conf *filename*	*filename* is the name of the configuration file that you want to remove, without its path name.
-h	(Optional) Displays the help usage for this command

oakcli delete repo

Use the `oakcli delete repo` command to remove a shared repository.

Syntax

Use the following syntax for the `oakcli delete repo` command:

```
oakcli delete repo repository_name [-h]
```

where *repository_name* is the name of the shared repository to be deleted.

Examples

Example 1 Deleting a shared repository

The following command deletes the `testrepo01` shared repository:

```
oakcli delete repo testrepo01
```

The command will not succeed if testrepo01 is active one or both nodes.

oakcli delete vdisk

Use the `oakcli delete vdisk` command to remove a virtual disk from a shared repository on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli delete vdisk vdisk_name -repo repository_name -h
```

where *vdisk_name* is the name assigned to the virtual disk, r*epository_name* is the name of the shared repository where the virtual disk was created, and -h shows the help text for the command.

Examples

Example 1 Removing an existing virtual disk

The following command removes a virtual disk named `t2g` from the shared repository named `repoprod1`:

```
oakcli delete vdisk t2g -repo repoprod1
```

oakcli delete vlan

Use the `oakcli delete vlan` command to remove a virtual local area network.

Syntax

Use the following syntax for the `oakcli delete vlan` command:

```
oakcli delete vlan vlan_name -node node_number [-h]
```

Parameters

Table 5–26 oakcli delete vm Command Parameters

Parameter	Description
`vlan` *vlan_name*	*vlan_name* is the name of the virtual local area network to be deleted.
`-node` *node_number*	*node_number* is the Oracle Database Appliance node from which you want to remove the virtual local area network.
`-h`	(Optional) Displays the help text.

Examples

Example 1 Deleting a virtual local area network

The following command deletes the `sample1` virtual local area network from node 1:

```
oakcli delete vlan sample1 -node 1
```

oakcli delete vm

Use the `oakcli delete vm` command to remove a virtual machine.

Syntax

Use the following syntax for the `oakcli delete vm` command:

```
oakcli delete vm vm_name [-server node_number] [-h]
```

Parameters

Table 5–27 oakcli delete vm Command Parameters

Parameter	Description
`vm` *vm_name*	(Optional) *vm_name* is the name of the virtual machine to be deleted.
`-server` *node_number*	*node_number* is the Oracle Database Appliance node from which you want to remove the virtual machine. If this optional parameter is not included, then the virtual machine is removed from both nodes.
`-h`	(Optional) Displays the help text.

Examples

Example 1 Deleting a virtual machine

The following command deletes the `ovu22` virtual machine from node 1:

```
oakcli delete vm ovu22 -server 1
```

oakcli delete vmtemplate

Use the `oakcli delete vmtemplate` command to remove a virtual machine template.

Syntax

Use the following syntax for the `oakcli` delete vmtemplate command:

```
oakcli delete vmtemplate template_name [-server=node_number][-h]
```

Parameters

Table 5–28 oakcli delete vmtemplate Command Parameters

Parameter	Description
`vmtemplate` *template_name*	*template_name* is the name of the virtual machine template to be removed.
`-server` *node_number*	*node_number* is the Oracle Database Appliance node from which you want to remove the virtual machine template. If this optional parameter is not included, then the virtual machine template is removed from both nodes.
`-h`	(Optional) Displays the help text.

Examples

Example 1 Deleting a virtual machine template

The following command deletes the ovu22 virtual machine template from both nodes:

```
oakcli delete vmtemplate ovu22
```

oakcli deploy

Use the oakcli deploy command to deploy Oracle Grid Infrastructure for a cluster on Oracle Database Appliance.

Syntax

Use the following syntax for the oakcli deploy command:

```
oakcli deploy [config] [-conf config_file] [-advance][-h]
```

Parameters

Table 5–29 oakcli deploy Command Parameters

Parameter	Description
config	(Optional) Runs the Oracle Appliance Manager Configurator.
-conf *config_file*	(Optional) Preloads the configuration stored in the named configuration file, *config_file*..
-advance	(Optional) Performs the deployment, or runs the deployment configurator, in advance mode.
-h	Optional) Displays the help text.

Usage Notes

- oakcli deploy deploys the complete Oracle Database Appliance.
- oakcli deploy config runs the Oracle Database Appliance configurator.
- oakcli deploy -conf config_file preloads an existing configuration file.

oakcli diskwritecache

Use the oakcli diskwritecache command to locate disks with write cache enabled and to disable disk write cache for those disks. Enabled write caches should be disabled as soon as downtime for Oracle Database Appliance can be scheduled. During the downtime, use this command with the disable option for each disk in turn that has an enabled write cache.

Syntax

```
oakcli diskwritecache  [disable disk_name | enable disk_name | status ] -h
```

where -h displays the help text for this command.

Examples

Example 1 Identifying disks with cache enabled

The following command lists the write cache status of all disks:

```
oakcli diskwritecache status
```

oakcli import vmtemplate

Use the `oakcli import vmtemplate` command to import virtual machine templates.

Syntax

Use the following syntax for the `oakcli import vmtemplate` command:

```
oakcli import vmtemplate vmtemplatename -files image_files | -assembly assembly_
file -repo repo_name [- node 0 | 1 ]
```

Parameters

Table 5–30 oakcli import vmtemplate Command Parameters

Parameter	Description
vmtemplatename	*vmtemplatename* is the name that you want to assign to the template.
-files	Use the -files option when importing one or more files that comprise a template.
image_files	*image_files* is one of the following: ■ a single template file name ■ a comma-separated list of files that comprise a single template ■ a URL enclosed in single quotes that links to a template file
-assembly	Use the -assembly option when importing an assembly file.
assembly_file	*assembly_file* is an assembly file or a URL enclosed in single quotes that links to an assembly file.
repo_name	*repo_name* is the name of the repository to store the template or templates that you are importing.
-node	Use the -node option when importing into a shared repository with a value of 0 or 1 to identify the node.

Usage Notes

- Include only one of the options, -files or -assembly, each time you run this command. You cannot include both of these options in the same statement.
- If the command imports more than one template from an assembly, then each template will automatically be given a unique name. These template names will include the name given in the vmtemplate *vmtemplatename* clause followed by a sequence number, such as *vmtemplatename1*, *vmtemplatename2*, *vmtemplatename3*, and so on.
- When importing into a shared repository, you must include the -node option with a valid node number, 0 or 1. Using the -node option for imports into non-shared repositories will cause the command to fail.

Examples

Example 1 Importing a VM Template from Dom0

The following command imports the required template (`OVM_OL5U7_X86_64_PVM_10GB.tgz`) from the `/OVS` directory in Dom0 into the `odarepo1` repository:

```
oakcli import vmtemplate OL5U7 -files /OVS/OVM_OL5U7_X86_64_PVM_10GB.tgz -repo
odarepo1
```

Example 2 Importing a VM Template from a Remote Server

The following command imports a template from a remote server using a URL to identify the server and the template file:

```
oakcli import vmtemplate OL5U6 -files 'http://example.com/vm-template/OEL-5/OVM_
OL5U6_X86_64_PVM_10GB.tgz' -repo odarepo2
```

Example 3 Importing VM Templates from an Assembly on a Remote Server

The following command imports the templates contained in the assembly stored on a remote server at the URL provided:

```
oakcli import vmtemplate OL6U1 -assembly 'http://example.com/assemblies/OEL6/OVM_
OL6U1_x86_PVHVM.ova' -repo odarepo1
```

Example 4 Importing VM Templates into a Shared Repository from an Assembly on a Remote Server

The following command imports the templates contained in the assembly stored on a remote server at the URL provided into a shared repository named repo4 on Node 1:

```
oakcli import vmtemplate OL6U1 -assembly 'http://example.com/assemblies/OEL6/OVM_
OL6U1_x86_PVHVM.ova' -repo repo4 -node 1
```

oakcli locate

The `oakcli locate` command helps you locate the physical disk that is associated with a named Oracle Automatic Storage Management disk by turning on (or off) the disk's LED light.

Syntax

Use the following `oakcli locate` command syntax where *`diskname`* is the name of the ASM disk that you want to locate:

```
oakcli locate disk diskname on|off
```

Parameters/Options

Table 5–31 oakcli locate Command Parameters and Options

Parameter	Description
`disk` *`diskname`*	Specifies the name of the Oracle ASM disk to locate
`on`	(Optional) Turns on the LED of the named disk.
`off`	(Optional) Turns off the LED of the named disk.

Examples

Example 1 Turning on the LED of a selected disk

The following command turns on the LED of the ASM disk `disk_pd_23`:

```
oakcli locate disk pd_23 on
```

oakcli manage diagcollect

Use the `oakcli manage diagcollect` command to collect diagnostic information about your Oracle Database Appliance for troubleshooting purposes, and for working with Oracle Support.

Syntax

```
oakcli manage diagcollect [--all | --crs [--crshome crs_home_dir] [--core] |
--install | --chmos [--incidenttime time [--incidentduration time] |
--adr adr_location [--afterdate date] [--aftertime time] [--beforetime time] ]
 [excl comp1,comp2,...] [--clean] [--storage]
```

Parameters

Table 5–32 oakcli manage diagcollect Command Parameters

Parameter	Description
`--all`	To collect all of the diagnostic information excluding Automatic Diagnostic Repository (ADR) and Cluster Health Monitor. This is the default option.
`--crs`	To collect Oracle Clusterware diagnostic information.
`--crshome` *crs_home_dir*	To specify the location of the Oracle Clusterware home directory.
`--core`	To package core files with the Oracle Clusterware diagnostic data.
`--install`	To collect the installation logs when the installation failed before running the script `root.sh`.
`--adr` *adr_location*	To collect diagnostic information for ADR, where *adr_location* specifies the location of the ADR information.
`--afterdate` *date*	To collect archives from the specified date. Specify the date in the `mm/dd/yyyy` format.
`--aftertime` *time*	To collect the archives after the specified time. Enter the time using the format `YYYYMMDDHHMISS24`. Supported only the with the `-adr` parameter.
`--beforetime` *time*	To collect the archives before the specified time. Enter the time using the format: `YYYYMMDDHHMISS24`. Supported only the with the `-adr` parameter.
`--chmos`	To collect Cluster Health Monitor data.
`--incidenttime` *time*	To collect Cluster Health Monitor data from the specified time. Enter the time using the format: `YYYYMMDDHHMISS24`. If you do not use the `--incidenttime` parameter, then the command collects data for the past 24 hours.

Table 5–32 (Cont.) oakcli manage diagcollect Command Parameters

Parameter	Description
`--incidentduration` *time*	To collect Cluster Health Monitor data for the duration after the specified time. Enter the time using the format: HH:MM. If you do not specify a duration, then the commands collects all Cluster Health Monitor data after the specified incident time.
`--excl` [*comp1*,*comp2*]	To exclude the specified component logs. Valid components are: `acfs`, `invt`, `sys`, `ocr`, `crs`, `home`, and `base`.
`--clean`	To remove the diagnosability information gathered by this command.
`--storage`	Collects all of the logs for any storage issues. This can be used when you are experiencing any problems with storage and need support to diagnose the logs.

oakcli modify

The `oakcli modify` command can

- attach virtual disks to or detach virtual disks from virtual machines
- transmit first-boot installation configuration messages to virtual machines
- assign networks to or delete networks from virtual machines and virtual machine templates

Syntax

Use the following `oakcli modify` command syntax to modify a virtual machine or virtual machine template:

```
oakcli modify [vm vmname [-attachvdisk vdisk_name | -detachvdisk vdisk_name | -s
key1:value1;key2:value2;...]] | [[vm vmname | vmtemplate vmtemplatename]
[-addnetwork network | -deletenetwork network]] [-h]
```

where name is the *vmname* of the virtual machine or *vmtemplatename* is the virtual machine template being modified and the objects and parameters are described in the following table, Table 5–33.

Objects and Parameters

Table 5–33 oakcli modify Command Objects and Parameters

Parameter	Description
`-attachvdisk` *vdisk_name*	Attaches the named virtual disk to the named virtual machine.
`-detachvdisk` *vdisk_name*	Detaches the named virtual disk from the named virtual machine.
`-s` *key1*:*value1*;*key2*:*value2*;...	Identifies a message consisting of one or more key/value pairs to send to the ovmd utility, where each key and value is separated by a colon (:) and each key/value pair is separated from the next key/value pair by a semicolon (;).
`-addnetwork` *network*	Identifies a new network to be assigned to the named virtual machine or template.
`-deletenetwork` *network*	Identifies the network to be deleted from the named virtual machine or template.

Table 5–33 (Cont.) oakcli modify Command Objects and Parameters

Parameter	Description
-h	(Optional) Displays the help text.

Usage Notes

- The vmtemplate object is valid with only the -addnetwork or -deletenetwork parameters.
- Do not use the -attachvdisk or the -detachvdisk option with the -addnetwork, -deletenetwork, or -s parameters.
- Include only one of the -addnetwork, -deletenetwork, or -s parameters when you use this command.

Examples

Example 1 Modifying the Network Defined in a Virtual Machine Template

The following command replaces the network assigned to the gc_11g virtual machine template with the net1 network:

```
oakcli modify vmtemplate gc_11g -addnetwork net1
```

Example 2 Sending a Message to a Running Virtual Machine

The following command updates the root user password for the gc_11g virtual machine:

```
oakcli modify vm gc_11g -s "com.oracle.linux.root-password:root123"
```

oakcli orachk

Use the oakcli orachk command to audit configuration settings with the ORAchk utility.

Syntax

oakcli orachk [-abvhpfm[u -o][o]ct] [-clusternodes *list* | -localonly] [-debug] [-dbnames *list* |-dbnone | -dball] [upgrade]

Parameters

Table 5–34 oakcli orachk Command Parameters

Parameter	Description
-a	Perform best practice check and recommended patch check.
-b	Perform best practice check only without the recommended patch check.
-v	Display version.
-h	Display command usage (help).
-p	Perform patch check only.
-f	Run the command offline.
-m	Exclude checks for Maximum Availability scorecards.

Table 5–34 (Cont.) oakcli orachk Command Parameters

Parameter	Description
`-u -o`	Perform check on pre-upgrade best practices (-u -o pre) or on post-upgrade best practices (-u -o post)
`-o`	As an argument to an option, if `-o` is followed by `v`, `V`, `Verbose`, or `VERBOSE`, output will display checks that pass on the display. Without the `-o` option, only failures will display on the screen.
`-c`	Determines granularity of information displayed on the screen. For use only when working with Oracle Support.
`-clusternodes` *list*	*list* is a comma delimited list containing the names of the nodes where the command should run.
`-localonly`	Run the command only on the local node.
--debug	Creates a debug log.
-dbnames list	*list* is a comma delimited list containing the names of the subset of databases on which the command should run.
-dbnone	Skip all database-related checks on all databases without prompting to select which database to skip.
-dball	Run all database-related checks on all databases without prompting to select which databases to check.
-upgrade	Force upgrade of the version of the orachk being run.

Usage Notes

- The command offers multiple options that are generic to the orachk command when run on servers other than Oracle Database Appliance. You can find details about these options by running the `oakcli orachk -h` command. The options are grouped into the following categories, but this document does not list the options for each category:
  - Report Options
  - Auto Restart Options
  - Daemon Options
  - Profile Run Options
- For more information about ORAchk, see the My Oracle Support note 1268927.2, "ORAchk Health Checks for the Oracle Stack" at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1268927.2.

oakcli resize dbstorage

Use the `oakcli resize dbstorage` command to resize the space used for a storage structure that was created for the purpose of migrating databases from ASM to ACFS. You can check the current space usage using `oakcli show fs` and then add or remove space using `oakcli resize dbstorage`.

Syntax

```
oakcli resize dbstorage -data size -reco size -redo size -db  db_name
```

Parameters

Table 5–35 oakcli resize dbstorage Command Parameters

Parameter	Description
`-data` *size*	Extendable size in GB for the DATA volume
`-reco` *size*	Extendable size in GB for the REDO volume
`-redo` *size*	Extendable size in GB for the RECO volume
`-db` *dbname*	Database for which these volumes must be resized
`-h`	(Optional) Displays the help usage for this command

Examples

The following command increases the size of the volume by 10 GB on the Data disk group.

```
oakcli resize dbstorage -data 10G
```

oakcli restart oda_base

Use the `oakcli restart oda_base` command to stop and restart ODA_BASE on the local node. Use this command when you are not concerned about the current status of ODA_BASE because it performs a forced shut down. You must run this command from Dom0 and, typically, if ODA_BASE needs to be restarted, you need to restart it on both nodes.

Syntax

```
oakcli restart oda_base | -h
```

where the ODA_BASE to be restarted is on the same node as the Dom0 from which you run this command.

oakcli show

Use the `oakcli show` command to display the status of Oracle Database Appliance components. The information displayed is for the node where you run the command. Use the help option, `oakcli show -h`, to see the list of components available on the current node.

Table 5–36 oakcli show Command Summary

Command	Description
oakcli show asr on page 5-36	Displays the Oracle Auto Service Request configuration
oakcli show cooling on page 5-37	Displays the status of the cooling units
oakcli show controller on page 5-37	Displays information about the disk controllers
oakcli show core_config_key on page 5-38	Displays information about the core deployment
oakcli show cpupool on page 5-38	Displays information about mappings between cores and virtual machines
oakcli show databases on page 5-39	Displays information about the databases
oakcli show dbhomes on page 5-39	Displays information about the database homes

Table 5–36 (Cont.) oakcli show Command Summary

Command	Description
oakcli show db_config_params on page 5-39	Displays configuration file names and parameters
oakcli show disk on page 5-40	Displays information about shared or local disks
oakcli show diskgroup on page 5-41	Displays information about Oracle ASM disk groups.
oakcli show env_hw on page 5-41	Displays the current server's environment type and hardware version
oakcli show expander on page 5-42	Displays information about the expanders
oakcli show enclosure on page 5-41	Displays information about the storage enclosure
oakcli show memory on page 5-42	Displays information about the memory subsystem
oakcli show network on page 5-43	Displays information about the network subsystem
oakcli show power on page 5-44	Displays the status of the power supply subsystem
oakcli show processor on page 5-44	Displays processor (CPU) information
oakcli show repo on page 5-44	Displays information about virtual machine repositories
oakcli show server on page 5-45	Displays information about the server subsystem
oakcli show storage on page 5-46	Displays information about the storage
oakcli show validation storage on page 5-46	Displays status (enabled or disabled) of validation storage error reporting
oakcli show validation storage errors on page 5-46	Displays hard storage errors
oakcli show validation storage failures on page 5-46	Displays soft storage errors
oakcli show vdisk on page 5-46	Displays information about virtual disks
oakcli show version on page 5-47	Displays version information for the software and firmware
oakcli show vlan on page 5-47	Displays version information about virtual local area networks
oakcli show vm on page 5-48	Displays version information about virtual machines
oakcli show vmtemplate on page 5-49	Displays information about virtual machine templates

Note: Depending on your model and version of Oracle Database Appliance software, the `oakcli show` command options may differ from the ones shown in the preceding table and explained in the following sections. Run the command `oakcli show -h` for an annotated list of components covered on your system.

oakcli show asr

Use the `oakcli show asr` command to display your Oracle Auto Service Request configuration details.

Syntax

```
oakcli show asr [-h]
```

Parameters

Table 5–37 oakcli show asr Command Parameters

Parameter	Description
-h	(Optional) Displays the help usage for this command

oakcli show cooling

Use the `oakcli show cooling` command to show information about the cooling subsystem.

Syntax

```
oakcli show cooling
```

Examples

Example 1 Displaying the cooling unit information

The following command to display the cooling unit information for the node where the command is executed:

```
oakcli show cooling
        NAME   HEALTH HEALTH_DETAILS LOCATION FAN % FAN SPEED
        Fan_0  OK     -              FM0      30 %  6300 RPM
        Fan_1  OK     -              FM0      19 %  3800 RPM
        Fan_10 OK     -              FM2      34 %  6600 RPM
        Fan_11 OK     -              FM2      23 %  4100 RPM
        Fan_12 OK     -              FM3      32 %  6300 RPM
        Fan_13 OK     -              FM3      22 %  3900 RPM
        Fan_14 OK     -              FM3      24 %  4700 RPM
        Fan_15 OK     -              FM3      14 %  2500 RPM
        Fan_2  OK     -              FM0      29 %  6400 RPM
        Fan_3  OK     -              FM0      18 %  3700 RPM
        Fan_4  OK     -              FM1      32 %  6400 RPM
        Fan_5  OK     -              FM1      20 %  3700 RPM
        Fan_6  OK     -              FM1      33 %  6400 RPM
        Fan_7  OK     -              FM1      22 %  3800 RPM
        Fan_8  OK     -              FM2      33 %  6400 RPM
        Fan_9  OK     -              FM2      22 %  3900 RPM
```

oakcli show controller

Use the `oakcli show controller` command to display information about the controllers.

Syntax `oakcli show controller` *`controller_id`* `[-h]`

Parameters

Table 5–38 oakcli show controller Command Variables and Parameters

Parameter	Description
controller_id	Specifies the controller for which the information should be displayed.
-h	(Optional) Displays the help usage for this command.

Examples

Example 1 Displaying information about controller 0 or controller 1

The following command displays details of controller 0:

```
oakcli show controller 0
```

oakcli show core_config_key

Use the `oakcli show core_config_key` command to display information about how to deploy your Oracle Database Appliance cores.

Syntax `oakcli show core_config_key`

Examples

Example 1 Determining whether the Oracle Database Appliance core configuration key has been applied

The following command shows the core count status on a new Oracle Database Appliance that has not been configured:

```
oakcli show core_config_key
Optional core_config_key is not applied on this machine yet!
```

Example 2 Displaying the Oracle Database Appliance core count status on a configured Oracle Database Appliance

The following command shows the core count status on a previously configured Oracle Database Appliance:

```
oakcli show core_config_key
Host's serialnumber = 1132FMW003
Configured Cores = 20
```

oakcli show cpupool

Use the `oakcli show cpupool` command to display core allocations to virtual machine mappings.

Syntax `oakcli show cpupool -node` *nodenum*

where nodenum is the number of the Oracle Database Appliance node that you wish to examine, either 0 or 1.

Examples

Example 1 Displaying core allocations to virtual machine mappings for a node

Display the core mapping information for Node 0:

```
oakcli show cpupool -node 0
         Pool           Cpu List          VM List
default-unpinned-pool   [14, 15, 16, 17,  ['test1_odarepo1','sample5_odarepo1',
                         18, 19, 20, 21,   'vm_very_long_name_sample1_odarepo1',
                         22, 23]           'win_vm1']
         twocpu          [12, 13]         ['vm1_odarepo1']
```

```
odaBaseCpuPool      [0, 1, 2, 3, 10  ['oakDom1']
                     , 11]
```

oakcli show databases

Use the `oakcli show databases` command to display information about each existing database, including database name, database type, database home name and location, and database version.

Syntax

```
oakcli show databases [-h]
```

Parameters

Table 5–39 oakcli show databases Command Parameters

Parameter	Description
`-h`	(Optional) Displays the help usage for this command

oakcli show dbhomes

Use the `oakcli show dbhomes` command to display information about each existing Oracle database home, including home name, home location, and database version.

Syntax

```
oakcli show dbhomes [-detail] [-h]
```

Parameters

Table 5–40 oakcli show dbhomes Command Parameters

Parameter	Description
-detail	(Optional) Includes a list of databases associated with each home
`-h`	(Optional) Displays the help usage for this command

oakcli show db_config_params

Use the `oakcli show db_config_params` command to display information about each existing Oracle database home, including home name, home location, and database version. By default, the command will search for files with the extension `.dbconf` located in the `/opt/oracle/oak/install/dbconf` directory.

Syntax

```
oakcli show db_config_params [-conf filename] [-detail] [-h]
```

Parameters

Table 5–41 oakcli show dbhomes Command Parameters

Parameter	Description
`-conf` *filename*	(Optional) Name of configuration file to be displayed. If not included, then the command displays all configuration files.
`-detail`	(Optional) Displays the parameter values stored in the configuration file or files.

Table 5–41 (Cont.) oakcli show dbhomes Command Parameters

Parameter	Description
-h	(Optional) Displays the help usage for this command.

Examples

Example 1 Displaying the default database configuration parameter values

The following command displays the default configuration parameter values:

```
oakcli show db_config_params -detail
Available DB configuration files are:
          Default
          DATABASE_BLOCK_SIZE       =>    8192
          DATABASE_LANGUAGE         =>    AMERICAN
          DATABASE_CHARACTERSET     =>    AL32UTF8
          DATABASE_TERRITORY        =>    AMERICA
          COMPONENT_LANGUAGES       =>    en
```

oakcli show disk

Use the `oakcli show disk` command to display disk information.

Syntax

```
oakcli show disk [-local | -shared | shared_disk_name [-all]| -h]
```

Parameters

Table 5–42 oakcli show disk Command Parameters

Parameter	Description
-local	(Optional) Specifies that information for all of the local disks should be displayed.
-shared	(Optional) Specifies that information for all of the shared disks should be displayed.
shared_disk_name	(Optional) Specifies that information for only the specific shared disk should be displayed.
-all	(Optional) Specifies that complete details of the selected disk or disks be displayed.
-h	(Optional) Displays the help usage for this command.

Usage Notes

- Running the command with no parameters is identical to running the `oakcli show disk -shared` command.
- The `-all` parameter produces valid output only when used with the *shared_disk_name* parameter. All other parameters are optional and cannot be combined with other parameters.

Examples

Example 1 Displaying information about the local disks

The following command displays information about all the local disks:

```
oakcli show disk -local
```

Example 2 Displaying information about the shared disks

The following command displays information about all the shared disks:

```
oakcli show disk -shared
```

Example 3 Displaying information about a specific shared disk

The following command displays information about the shared disk named `pd_01`

```
oakcli show disk pd_01:
```

oakcli show diskgroup

Use the `oakcli show diskgroup` command to display Oracle ASM disk group information.

Syntax

```
oakcli show diskgroup [disk_group_name]
```

Parameters

Table 5–43 oakcli show disk group Command Parameters

Parameter	Description
disk_group_name	(Optional) The name of an Oracle ASM disk group for which complete details should be displayed. If you do not specify this parameter, then information for all of the Oracle ASM disk groups is displayed.

oakcli show env_hw

Use the `oakcli show env_hw` command to display the environment type and hardware version of the current node.

Syntax

```
oakcli show env_hw [-h]
```

where `-h` will generate the help information for the command.

Examples

Example 1 Showing the environment type and hardware model

The following example shows the output from the `oakcli show env_hw` command when logged onto ODA_BASE on Oracle Database Appliance X3-2 Virtualized Platform:

```
oakcli show env_hw
VM-ODA_BASE ODA X3-2
```

oakcli show enclosure

Use the `oakcli show enclosure` command to display information about the storage enclosure subsystem on the node where the command is executed.

Syntax

```
oakcli show enclosure
```

Examples

Example 1 Displaying storage enclosure subsystem information

Display the storage enclosure subsystem information of the node where the command is executed:

```
oakcli show enclosure

NAME        SUBSYSTEM         STATUS      METRIC

E0_FAN0  Cooling             OK          3000 rpm
E0_FAN1  Cooling             OK          3220 rpm
E0_FAN2  Cooling             OK          3520 rpm
E0_FAN3  Cooling             OK          3070 rpm
E0_IOM0  Encl_Electronics    OK          -
E0_IOM1  Encl_Electronics    OK          -
E0_PSU0  Power_Supply        OK          -
E0_PSU1  Power_Supply        OK          -
E0_TEMP0 Amb_Temp            OK          25 C
E0_TEMP1 Midplane_Temp       OK          32 C
E0_TEMP2 PCM0_Inlet_Temp     OK          34 C
E0_TEMP3 PCM0_Hotspot_Temp   OK          44 C
E0_TEMP4 PCM1_Inlet_Temp     OK          31 C
E0_TEMP5 PCM1_Hotspot_Temp   OK          42 C
E0_TEMP6 IOM0_Temp           OK          42 C
E0_TEMP7 IOM1_Temp           OK          50 C
```

oakcli show expander

Use the `oakcli show expander` command to display information about an expander.

Syntax

```
oakcli show expander expander_id
```

where *expander_id* identifies the specific expander.

oakcli show memory

Use the `oakcli show memory` command to display information about memory modules.

Syntax

```
oakcli show memory
```

Examples

Example 1 Displaying memory module information

Display the memory information of the node where the command is executed:

```
oakcli show memory
        NAME    HEALTH HEALTH_DETAILS PART_NO.    SERIAL_NO.         LOCATION
MANUFACTURER MEMORY_SIZE CURR_CLK_SPEED ECC_Errors

        DIMM_0  OK     -              001-0003-01 00CE02124685D963AC P0/D0
Samsung      16 GB       1600 MHz       0
        DIMM_1  OK     -              001-0003-01 00CE02124634F65B85 P0/D1
```

```
Samsung      16 GB       1600 MHz       0
        DIMM_10 OK     -              001-0003-01 00CE02124685D963B2 P1/D2
Samsung      16 GB       1600 MHz       0
        DIMM_11 OK     -              001-0003-01 00CE02124634F6565B P1/D3
Samsung      16 GB       1600 MHz       0
        DIMM_12 OK     -              001-0003-01 00CE02124634F6598B P1/D4
Samsung      16 GB       1600 MHz       0
        DIMM_13 OK     -              001-0003-01 00CE02124685D963B4 P1/D5
Samsung      16 GB       1600 MHz       0
        DIMM_14 OK     -              001-0003-01 00CE02124634F65956 P1/D6
Samsung      16 GB       1600 MHz       0
        DIMM_15 OK     -              001-0003-01 00CE02124685D96348 P1/D7
Samsung      16 GB       1600 MHz       0
        DIMM_2  OK     -              001-0003-01 00CE02124685D963B1 P0/D2
Samsung      16 GB       1600 MHz       0
        DIMM_3  OK     -              001-0003-01 00CE02124634F65B6E P0/D3
Samsung      16 GB       1600 MHz       0
        DIMM_4  OK     -              001-0003-01 00CE02124634F65C65 P0/D4
Samsung      16 GB       1600 MHz       0
        DIMM_5  OK     -              001-0003-01 00CE02124685D963AB P0/D5
Samsung      16 GB       1600 MHz       0
        DIMM_6  OK     -              001-0003-01 00CE02124634F6598E P0/D6
Samsung      16 GB       1600 MHz       0
        DIMM_7  OK     -              001-0003-01 00CE02124685D9659E P0/D7
Samsung      16 GB       1600 MHz       0
        DIMM_8  OK     -              001-0003-01 00CE02124685D963A3 P1/D0
Samsung      16 GB       1600 MHz       0
        DIMM_9  OK     -              001-0003-01 00CE02124634F656F7 P1/D1
Samsung      16 GB       1600 MHz       0
```

oakcli show network

Use the `oakcli show network` command to display information about the network subsystem.

Syntax

```
oakcli show network
```

Examples

Example 1 Displaying the network information

Display the network information of the node where the command is executed:

```
oakcli show network
        NAME           HEALTH HEALTH_DETAILS LOCATION PART_NO MANUFACTURER MAC_
ADDRESS        LINK_DETECTED DIE_TEMP

        Ethernet_NIC_0 OK     -              NET0     X540    INTEL
00:10:e0:23:f1:08  yes (eth0)    61.000 degree C
        Ethernet_NIC_1 OK     -              NET1     X540    INTEL
00:10:e0:23:f1:09  yes (eth1)    61.000 degree C
        Ethernet_NIC_2 OK     -              NET2     X540    INTEL
00:10:e0:23:f1:0a  yes (eth4)    53.750 degree C
        Ethernet_NIC_3 OK     -              NET3     X540    INTEL
00:10:e0:23:f1:0b  yes (eth5)    53.750 degree C
        Ethernet_NIC_4 -      -              NET4     82599EB INTEL
00:1B:21:B6:09:F1  yes (eth3)    -
        Ethernet_NIC_5 -      -              NET5     82599EB INTEL
00:1B:21:B6:09:F0  yes (eth2)    -
```

oakcli show power

Use the `oakcli show power` command to display information about the power supply subsystem.

Syntax

```
oakcli show power
```

Examples

Example 1 Displaying the power supply information

Display the power supply information of the node where the command is executed:

```
oakcli show power
        NAME            HEALTH HEALTH_DETAILS PART_NO. SERIAL_NO.         LOCATION
INPUT_POWER OUTPUT_POWER INLET_TEMP      EXHAUST_TEMP

        Power_Supply_0  OK     -              7047410  476856F+1242CE0020 PS0
Present     113 watts    33.250 degree C 36.688 degree C
        Power_Supply_1  OK     -              7047410  476856F+1242CE004J PS1
Present     89 watts     37.000 degree C 39.438 degree C
```

oakcli show processor

Use the `oakcli show processor` command to display information about CPU processors.

Syntax

```
oakcli show processor
```

Examples

Example 1 Displaying the CPU processor information

Display the CPU processor information of the node where the command is executed:

```
oakcli show processor
        NAME  HEALTH HEALTH_DETAILS PART_NO. LOCATION
MODEL                         MAX_CLK_SPEED TOTAL_CORES ENABLED_CORES

        CPU_0 OK     -              060D     P0 (CPU 0)
Intel(R) Xeon(R) CPU E5-2690  2.900 GHZ         8           8
        CPU_1 OK     -              060D     P1 (CPU 1)
Intel(R) Xeon(R) CPU E5-2690  2.900 GHZ     8           8
```

oakcli show repo

Use the `oakcli show repo` command to display information about virtual machine repositories. To see all repositories, do not include the repository name and node number. To see a specific shared repository, include the repository name and node.

Syntax

```
oakcli show repo [reponame -node 0|1]
```

where *reponame* identifies a specific repository name.

Examples

Example 1 Displaying the available virtual machine repositories

Display the virtual machined repositories on the two nodes of your Oracle Database Appliance Virtualized Platform:

```
oakcli show repo
        NAME      REPOTYPE  NODENUM
        odarepo1  local     0
        odarepo2  local     1
        repo1     shared    0
        repo1     shared    1
```

Example 2 Displaying details about a specific shared repository

Display information about the repository named repo1 on Node 1:

```
oakcli show repo repo1 -node 1

Resource: repo1_1
        AutoStart       :       restore
        DG              :       DATA
        Device          :       /dev/asm/repo1-286
        ExpectedState   :       Online
        MountPoint      :       /u01/app/repo1
        Name            :       repo1_0
        Node            :       all
        RepoType        :       shared
        Size            :       102400
        State           :       Online
```

oakcli show server

Use the `oakcli show server` command to display information about the server subsystem.

Syntax

```
oakcli show server
```

Examples

Example 1 Displaying the server information

Display the server information of the node where the command is executed:

```
oakcli show server
        Power State              : On
        Open Problems            : 1
        Model                    : SUN FIRE X4170 M3
        Type                     : Rack Mount
        Part Number              : 31324979+1+1
        Serial Number            : 1250FML046
        Primary OS               : Not Available
        ILOM Address             : 10.210.57.26
        ILOM MAC Address         : 00:10:E0:23:F1:0C
        Description              : Oracle Database Appliance X3-2 AK00050333
        Locator Light            : Off
        Actual Power Consumption : 268 watts
        Ambient Temperature      : 22.000 degree C
        Open Problems Report     :
```

```
Open Problem 1
        Problem time        : Tue Apr  2 06:10:37 2013
        Problem subsystem   : System
        Problem location    : /SYS (Host System)
        Problem description : An error has occurred in which no automated
diagnosis is available to identify faulty component. (Probability: 100, UUID:
a72b4e35-140d-c86d-b87b-daf3eb43f5c7, Part Number: 31324979+1+1, Serial Number:
1250FML046, Reference Document: http://www.sun.com/msg/SPX86-8003-JP)
```

oakcli show storage

Use the `oakcli show storage` command to display information about the storage.

Syntax

```
oakcli show storage -errors
```

The `oakcli show storage` command displays information about the controllers, expanders, and disks. If the `-errors` parameter is included, then the command will display detailed information about reported errors.

oakcli show validation storage

Use the `oakcli show validation storage` command to show whether validation storage is enabled or disabled.

Syntax

```
oakcli show validation storage
```

oakcli show validation storage errors

Use the `oakcli show validation storage errors` command to show hard storage errors. Hard errors include having the wrong type of disk inserted into a particular slot, an invalid disk model, or an incorrect disk size.

Syntax

```
oakcli show validation storage errors
```

oakcli show validation storage failures

Use the `oakcli show validation storage failures` command to show soft storage errors. A typical soft disk error would be an invalid version of the disk firmware.

Syntax

```
oakcli show validation storage failures
```

oakcli show vdisk

Use the `oakcli show vdisk` command to display information about virtual disks on Oracle Database Appliance Virtualized Platform.

Syntax

```
oakcli show vdisk [vdisk_name -repo repository_name]
```

where *vdisk_name* is an optional value if you want see information for just one virtual disk and -repo *repository_name* is a required parameter if a vdisk is named in the command.

Examples

Example 1 Display the information for all vdisks

Display information about all vdisks on your Oracle Database Appliance:

```
# oakcli show vdisk

    NAME                          SIZE     TYPE    REPOSITORY
    myvdisk                       10G      local   vdiskrepo
    newv                          1G       local   vdiskrepo
```

Example 2 Display information for a single vdisk

Display information for the vdisk named myvdisk:

```
# oakcli show vdisk myvdisk1
Resource: myvdisk_vdiskrepo
        Name          :     myvdisk_vdiskrepo
        RepoName      :     vdiskrepo
        Size          :     10G
        Type          :     local
        VmAttached    :     0
```

oakcli show version

Use the oakcli show version command to display patch versions for Oracle Database Appliance software and firmware.

Syntax

```
oakcli show version [-detail]
```

Use the -detail option to display detailed version information.

Examples

Example 1 Displaying basic version information

Display the version information for the software and firmware on your Oracle Database Appliance:

```
oakcli show version
```

oakcli show vlan

Use the oakcli show vlan command to display information about virtual local area networks configured on Oracle Database Appliance.

Syntax

```
oakcli show vlan
```

Examples

Example 1 Displaying Virtual Local Area Network Information

Display the names, tag ID numbers, networks, and node assignments for the available local virtual area networks:

```
oakcli show vlan

NAME                 ID    INTERFACE   NODENUM

net1                 1     bond0       0
net1                 1     bond0       1
net2                 1     bond1       0
net2                 1     bond1       1
net3                 2     bond1       0
net3                 4     bond0       1
net10                20    bond1       0
net10                20    bond1       1
```

oakcli show vm

Use the `oakcli show vm` command to display information about virtual machines.

Syntax

```
oakcli show vm [vm_name | -h]
```

Parameters

Table 5–44 oakcli show vm Command Parameters

Parameter	Description
vm_name	(Optional) The name of the virtual machine for which details should be displayed. If you do not specify this parameter, then information for all of the virtual machines is displayed.
-h	(Optional) Displays the help usage for this command.

Examples

Example 1 Displaying details for all virtual machines

Display the virtual machine names, memory and vCPU allocations, status, virtual disks, and repository name for all virtual machines:

```
oakcli show vm

NAME                 MEMORY          VCPU        STATE          REPOSITORY

sample5_odarepo1       2048            2         OFFLINE        odarepo1
sample6_odarepo1       2048            2         OFFLINE        odarepo2
test1_odarepo1         2048            2         OFFLINE        odarepo1
test2_odarepo2         2048            2         OFFLINE        odarepo2
vm1_odarepo1           4096            4         ONLINE         odarepo1
vm2_odarepo2           2048            2         OFFLINE        odarepo2
win_vm1                1500            1         ONLINE         odarepo1
```

Example 2 Displaying information for a single virtual machine

The following command displays information about the `vm1_odarepo1` virtual machine:

```
oakcli show vm vm1_odarepo1
Resource: vm1_odarepo1
        AutoStart       :       restore
        CPUPriority     :       100
        Disks           :       |file:/OVS/Repositories/odarepo1/Vi
                                rtualMachines/vm1_odarepo1/System.i
                                mg,xvda,w||file:/OVS/Repositories/o
                                darepo1/VirtualMachines/vm1_odarepo
                                1/u01.img,xvdb,w|
        Domain          :       XEN_PVM
        ExpectedState   :       online
        FailOver        :       false
        IsSharedRepo    :       false
        Keyboard        :       en-us
        MaxMemory       :       3000
        MaxVcpu         :       4
        Memory          :       4096
        Mouse           :       OS_DEFAULT
        Name            :       vm1_odarepo1
        Networks        :       |mac=00:21:F6:00:00:E4|
        NodeNum         :       0
        NodeNumStart    :
        OS              :       OL_5
        PrivateIP       :       None
        ProcessorCap    :       100
        RepoName        :       odarepo1
        State           :       Online
        TemplateName    :       otml_sample1_odarepo1
        Vcpu            :       4
        cpupool         :       twocpu
        vncport         :       5901
```

oakcli show vmtemplate

Use the `oakcli show vmtemplate` command to display information about virtual machine templates.

Syntax

```
oakcli show vmtemplate [vmtemplate_name | -h]
```

Parameters

Table 5–45 oakcli show vmtemplate Command Parameters

Parameter	Description
vmtemplate_name	(Optional) The name of the virtual template for which details should be displayed. If you do not specify this parameter, then information for all of the virtual templates is displayed.
-h	(Optional) Displays the help usage for this command.

Examples

Example 1 Displaying details for a single virtual template

The following command displays information about the `sample1_odarepo1` virtual template:

```
oakcli show vmtemplate sample_odarepo1
Resource: sample1_odarepo1
        CPUPriority     :       100
        Disks           :       |file:/OVS/Repositories/odarepo1/Te
                                mplates/otml_sample1_odarepo1/Syste
                                m.img,xvda,w||file:/OVS/Repositorie
                                s/odarepo1/Templates/otml_sample1_o
                                darepo1/u01.img,xvdb,w|
        Domain          :       XEN_PVM
        Keyboard        :       en-us
        MaxMemory       :       2048
        MaxVcpu         :       2
        Memory          :       2048
        Mouse           :       OS_DEFAULT
        Name            :       sample1_odarepo1
        Networks        :       |bridge=priv1||bridge=net1||bridge=
                                net2|
        NodeNum         :       0
        OS              :       OL_5
        ProcessorCap    :       100
        RepoName        :       odarepo1
        Vcpu            :       2
```

oakcli show vmconsole

Use the `oakcli show vmconsole` command to open a GUI VM console to manage a specific virtual machine.

Syntax oakcli show vmconsole *vmname* `[-h]`

where *vm_name* is the name of the virtual machine for which you want to open a console and -h displays the help usage for the command.

Examples Example 1 Opening a virtual machine console

The following command opens a console for the virtual machine named `vm1_odarepo1`:

```
oakcli show vmconsole vm1_odarepo1
```

oakcli start

Use the `oakcli start` command to start a virtual machine, to start a shared repository on a node, or to start ODA_BASE on the local node.

Table 5–46 oakcli start Command Summary

Command	Description
oakcli start oda_base on page 5-51	Starts ODA_BASE on the local node
oakcli start repo on page 5-51	Starts a shared repository on the specified node
oakcli start vm on page 5-51	Starts a virtual machine on the specified node

oakcli start oda_base

Use the `oakcli start oda_base` command to start ODA_BASE on the local node.

Syntax

```
oakcli start oda_base [-h]
```

Parameters

Table 5–47 oakcli start oda_base Command Parameters

Parameter	Description
`-h`	(Optional) Displays the online help.

Example

Connect to Dom0 on the desired node and enter the following command to start ODA_BASE on that node:

```
oakcli start oda_base
```

oakcli start repo

Use the `oakcli start repo` command to start a shared repository on a node.

Syntax

```
oakcli start repo repo_name [-node node_number] [-h]
```

Parameters

Table 5–48 oakcli start repo Command Parameters

Parameter	Description
repo_name	*repo_name* is the name of the shared repository to be started.
-node *node_number*	Specifies the node on which to start the shared repository. *node_number* is the number of the node where it is to be started, either 0 or 1. If `-node` is not specified, the shared repository is started on both nodes.
`-h`	(Optional) Displays the online help.

Example

The following command starts the shared repository named `repo1` on Node 0:

```
oakcli start repo repo1 -node 0
```

oakcli start vm

Use the `oakcli start vm` command to start a virtual machine on a node.

Syntax

```
oakcli start vm vm_name [-node node_number] [-d] [-h]
```

Parameters

Table 5–49 oakcli start vm Command Parameters

Parameter	Description
vm_name	*vm_name* is the name of the virtual machine to be started.
-node *node_number*	Specifies the node on which to start the virtual machine. *node_number* is the number of the node where it is to be started, either 0 or 1. If -node is not specified, a virtual machine is started on both nodes.
-d	Provides details about the virtual machine starting procedure
-h	(Optional) Displays the online help.

Example

The following command starts the virtual machine named vm1_odarepo1 on Node 0.

```
oakcli start vm vm_odarepo1 -node 0
```

oakcli stop

Use the oakcli stop command to stop a virtual machine, to stop a shared repository on a node, or to stop ODA_BASE on the local node.

Syntax

```
oakcli stop [vm vm_name [-force] | repo repo_name [-node node_number] | oda_base ]
[-h]
```

Parameters/Options

Table 5–50 oakcli stop Command Parameters

Parameter	Description
-h	(Optional) Displays the online help.
vm *vm_name*	*vm_name* is the name of the virtual machine to be stopped.
-force	(Optional) forces the virtual machine to stop.
repo *repo_name*	*repo_name* is the name of the shared repository to be stopped.
-node *node_number*	*node_number* is the number of the node where the shared repository is to be stopped, either 0 or 1. The -node parameter is only valid when stopping a virtual machine on a shared repository. If -node is not specified, the shared repository is stopped on both nodes.
oda_base	stops ODA_BASE on the local node.

Examples

Example 1 Forcing a virtual machine to stop

Enter the following command to force the virtual machine named vm1_odarepo1 to stop:

```
oakcli stop vm vm_odarepo1 -force
```

Example 2 Stopping a Shared Repository

Enter the following command to stop the shared repository named `repo1` on Node 0:

```
oakcli stop repo repo1 -node 0
```

Example 3 Stopping the local ODA_BASE

Connect to Dom0 on the desired node and enter the following command to stop ODA_BASE on that node:

```
oakcli stop oda_base
```

oakcli stordiag

Use the `oakcli stordiag` command to run diagnostic tests on a disk in the storage shelf or storage expansion shelf.

Syntax

`oakcli stordiag` *`resource_typen`* | `-h`

where *`resource_type`* is a prefix that depends on the configuration, *`n`* is the disk number (starting with 0 and increasing to one less than the number of disks), and `-h` shows the help for this command. Select the value for resource_type based from one of the following options:

- on Oracle Database Appliance Version 1, use `d`
- on Oracle Database Appliance with a single storage shelf, use `pd_`
- on Oracle Database Appliance with an expansion storage shelf, use `e0_pd_` for a disk in the storage shelf and use `e1_pd_` for a disk in the expansion storage shelf

Usage Notes

For Oracle Database Appliance systems that have internal storage, use the format d_[..] to identify the disk to be diagnosed, for Oracle Database Appliance system that have connected a connected shelf (and optional storage expansion shelf), use the format e[0..1]_p[0..23] to identify the disk to be diagnosed.

Examples

Example 1 Run diagnostic tests on a disk in a storage expansion shelf

The following command runs the diagnostic tests on the fourth storage unit in the storage shelf, where there is no expansion storage shelf:

```
# oakcli stordiag e1_pd_3
 Node Name : hr0
 Test : Diagnostic Test Description

   1  : OAK Check
        NAME          PATH          TYPE          STATE         STATE_DETAILS
        pd_03         /dev/sdw      HDD           ONLINE        Good

   2  : ASM Check
. . .
<output truncated>
```

oakcli test asr

Use the `oakcli test asr` command to send a test trap to determine if Oracle Auto Service Request (Oracle ASR) is configured and working correctly. The command returns a success message if Oracle ASR is functioning properly.

Syntax

```
oakcli test asr [-h]
```

oakcli unpack

Use the `oakcli unpack` command to unpack packages into the Oracle Appliance Manager command-line interface repository.

Syntax

```
oakcli unpack -package absolute_package_name
```

where *absolute_package_name* identifies the package to be unpacked using the package's full absolute path and file name.

Examples

Example 1 Unpacking a patch package into the Oracle Appliance Manager command-line interface repository

The following command unpacks the `p13982331_23000_Linux-86-64.zip` package, which was previously copied to `/tmp` on the current node, into the node's Oracle Appliance Manager command-line interface repository:

```
oakcli unpack -package /tmp/p13982331_23000_Linux-86-62.zip
```

oakcli update

Use the `oakcli update` command to apply Oracle Database Appliance patches. For upgrading only the Oracle Database software, see the `oakcli upgrade` command.

Syntax

```
oakcli update -patch version [[--infra] | [[--gi][--database]]] [--noreboot] |
[--clean] | [--verify]
```

Parameters

Table 5–51 oakcli update Command Parameters

Parameter	Description
`-patch` *version*	Specifies the patch update that you want to install
`--infra`	Specifies that the infrastructure is to be patched, including firmware, OS, ASR, HMP, OAK, and so on. This is the default option.
`--gi`	Specifies that the Grid Infrastructure is to be patched.
--database	Specifies that the database homes are to be patched.
--noreboot	Node will not be rebooted after patching.
--clean	Clean up all temporary files on the local node.

Table 5–51 (Cont.) oakcli update Command Parameters

Parameter	Description
--verify	Show the patchable components on the node.

Examples

Example 1 Patching a node

The following command updates the current node with the 2.5.0.0.0 patch:

```
oakcli update -patch 2.5.0.0.0
```

oakcli upgrade

Use the `oakcli upgrade` command to upgrade Oracle Databases on Oracle Database Appliance. For patching Oracle Database Appliance itself, see the `oakcli update` command.

Syntax

```
oakcli upgrade database [-db db_names | -from source_home] -to destination_home
[-h]
```

Parameters

Table 5–52 oakcli upgrade Command Parameters

Parameter	Description
`-db` *db_names*	Specifies the name or names (in a comma-delimited list) of the database or databases you want to upgrade.
`-from` *source_home*	Specifies the current Oracle Database home of the databases you are upgrading.
`-to` *destination_home*	Specifies the Oracle Database home containing the version to which you want to upgrade the databases
--h	(Optional) Displays the online help.

Usage Notes

- You must include either a `-db` parameter or a `-from` parameter.
- Running the command with a `-db` parameter upgrades only the named databases, regardless of their current Oracle Database homes. If you include a `-from` parameter, in addition to a `-db` parameter, then the command ignores the `-from` parameter. That is, the command upgrades named databases from other homes and ignores the databases in the named home if they are not listed in the `-db` parameter .
- Running the command without a `-db` parameter will upgrade all of the databases in the named Oracle Database home.
- You must always provide a `-to` parameter which names an existing Oracle Database home.

Examples

Example 1 Upgrading an Oracle Database

The following command upgrades an 11.2.0.2.5 database named `tpcc` to version 11.2.0.3.1 using the Oracle Database home directory `OraDb11203_home1`:

```
oakcli upgrade database -db tpcc -to OraDb11203_home1
```

oakcli validate

Use the `oakcli validate` command to validate the state of an Oracle Database Appliance or the viability of an operating system patch.

Syntax

Use the following `oakcli validate` syntax to validate an Oracle Database Appliance:

```
oakcli validate [[-V | -l | -h]] | [[-v][-f output_file] [-a | -d | -c
checklist][-ver patch_version]]
```

Parameters/Options

Table 5–53 oakcli validate Command Parameters

Parameter	Description
-V	Display the version of oakValidation.
-l	List the items that can be checked along with their descriptions.
-h	Display the online help.
-v	Show verbose output (must be used with a parameter that generates a validation report).
-f *output_file*	Send output to a file with a fully-qualified file name, *output_file*, instead of to the screen (`stdout`).
-a	Run all system checks, including `DiskCalibration`. Oracle recommends that you use this command to validate system readiness before deployment. Do not run `oakcli validate` with this option on a busy production system, because the `DiskCalibration` system check can cause performance issues.
-d	Run only the default checks.
-c *checklist*	Run the validation checks for the items identified in *checklist*, a comma-delimited list. Use this parameter to check either a single item or subset of items.
-ver *patch_version*	Report any reasons for not being able to patch Oracle Database Appliance with the patch named in patch_version.

Examples

Example 1 Listing all checks and their descriptions

List all of the checks available with `oakcli validate` along with their descriptions:

```
oakcli validate -l

        Checkname -- Description
        =========    ===========
```

```
        *SystemComponents -- Validate system components based on ilom sensor data
        readings
        *OSDiskStorage -- Validate OS disks and filesystem information
        *SharedStorage -- Validate Shared storage and multipathing information
        DiskCalibration -- Check disk performance with orion
        *NetworkComponents -- Validate public and private network components
        *StorageTopology -- Validate external JBOD connectivity
        asr -- Validate asr components based on asr config file and ilom sensor
        data readings

* -- These checks are also performed as part of default checks
```

Note: The `NetworkComponents` validation check is not available on hardware prior to Oracle Database Appliance X3-2.

Example 2 Running all checks

Enter the following syntax to run all checks

```
oakcli validate -a
```

Example 3 Validating storage cable connections

Enter the following syntax to validate the connections to you storage shelf and, if connected, your storage expansion shelf:

```
oakcli validate -c storagetopology
```

Example 4 Validating ASR

Enter the following syntax to validate your ASR configuration:

```
# oakcli validate -c asr
INFO: oak Asr information and Validations
RESULT: /opt/oracle/oak/conf/asr.conf exist
RESULT: ASR Manager ip:10.139.154.17
RESULT: ASR Manager port:1162
SUCCESS: ASR configuration file validation successfully completed
RESULT: /etc/hosts has entry 141.146.156.46 transport.oracle.com
RESULT: ilom alertmgmt level is set to minor
RESULT: ilom alertmgmt type is set to snmptrap
RESULT: alertmgmt snmp_version is set to 2c
RESULT: alertmgmt community_or_username is set to public
RESULT: alertmgmt destination is set to 10.139.154.17
RESULT: alertmgmt destination_port is set to 1162
SUCCESS: Ilom snmp confguration for asr set correctly
RESULT: notification trap configured to ip:10.139.154.17
RESULT: notification trap configured to port:1162
SUCCESS: Asr notification trap set correctly
INFO: IP_ADDRESS HOST_NAME SERIAL_NUMBER ASR PROTOCOL SOURCE PRODUCT_NAME
INFO: --------------- ------------------------------
------------------------------ -------- --------- --------------
------------------------------
10.170.79.98 oda-02-c 1130FMW00D Enabled SNMP ILOM SUN FIRE X4370 M2 SERVER
10.170.79.97 oda-01-c 1130FMW00D Enabled SNMP ILOM SUN FIRE X4370 M2 SERVER
INFO: Please use My Oracle Support 'http://support.oracle.com' to view the
activation status.
SUCCESS: asr log level is already set to Fine.
RESULT: Registered with ASR backend.
RESULT: test connection successfully completed.
```

```
RESULT: submitted test event for asset:10.139.154.17
RESULT: bundle com.sun.svc.asr.sw is in active state
RESULT: bundle com.sun.svc.asr.sw-frag is in resolved state
RESULT: bundle com.sun.svc.asr.sw-rulesdefinitions is in resolved state
RESULT: bundle com.sun.svc.ServiceActivation is in active state
SUCCESS: ASR diag successfully completed
```

Example 5 Checking the Viability of a Patch

Use the following command before attempting to patch Oracle Database Appliance to determine if it will succeed or if changes need to be made before applying the patch. Warning and error labels are highlighted in magenta and red font respectively.

```
# oakcli validate ospatch -ver 12.1.2.2.0
INFO: Validating the OS patch for the version 12.1.2.2.0
WARNING: 2015-02-10 06:30:32: Patching sub directory
/opt/oracle/oak/pkgrepos/orapkgs/OEL/5.10/Patches/5.10.1 is not existing
INFO: 2015-02-10 06:30:32: May need to unpack the Infra patch bundle for the
version: 12.1.2.2.0
ERROR: 2015-02-10 06:30:32: No OS patch directory found in the repository
```

A

Oracle Database Appliance Software Configuration Defaults

The Oracle Database Appliance software configuration defaults are detailed in the following sections:

- Directory Paths for Oracle Database Appliance
- Oracle Groups and Users Configuration for Oracle Database Appliance
- Database Disk Group Sizes for Oracle Database Appliance
- Storage on Oracle Database Appliance
- System Configuration for Oracle Database Appliance

Directory Paths for Oracle Database Appliance

Table A–1 Directory Paths for Oracle Database Appliance

Item	Directory Path
Grid home	/u01/app/*release-specific_name*/grid
Grid base	/u01/app/grid
Oracle home	/u01/app/oracle/product/*release-specific_name*/dbhome_*sequence_number*
Oracle base	/u01/app/oracle
Oracle Inventory	/u01/app/oraInventory

Oracle Groups and Users Configuration for Oracle Database Appliance

Table A–2 Oracle Groups and Users Configuration for Oracle Database Appliance

Groups and Users	Default Value
Oracle Grid Infrastructure installation owner	`grid`, password `welcome1` (change after installation), UID 1000
Oracle Database installation owner	`oracle`, password `welcome1` (change after installation), UID 1001
Oracle Database system administrator	`sys`, password `welcome1` (change after installation)
Oracle Database generic administrator	`system`, password `welcome1` (change after installation)
Oracle Inventory system privileges group	`oinstall`, GID 1001
Oracle ASM Administrators system privileges	`asmadmin`, GID 1006
Oracle ASM Users system privileges	`asmdba`, GID 1004

Table A–2 (Cont.) Oracle Groups and Users Configuration for Oracle Database Appliance

Groups and Users	Default Value
Oracle ASM Operator system privileges	`asmoper`, GID 1005
Oracle Database Administrators system privileges	`dba`, GID 1002
Oracle Database Operator system privileges	`racoper`, GID 1003

Database Disk Group Sizes for Oracle Database Appliance

Table A–3 shows the sizes for DATA, RECO, REDO, and FLASH disk groups on various configurations of Oracle Database Appliance. Each row has values for either normal or for high redundancy levels. The disk capacities shown in the table vary because they are derived by converting disk hardware terabytes (based on 1 kilobyte=1,000 bytes) into data storage terabytes (based on 1 kilobyte=1,024 bytes).

The space calculated with a storage expansion shelf approximately doubles the space mentioned in Table A–3. Note that the storage expansion shelf can only be used in Oracle Database Appliance X3-2, X4-2, and X5-2.

Table A–3 Database Disk Group Sizes for Oracle Database Appliance

Sizing	Version 1 (GB)	X3-2 and X4-2 (GB)	X5-2 (GB)
HDD Size	559	838	3,600
Total HDD	11,180	16,760	57,600
Total SSD (REDO Diskgroup)	272	744	744
Total SSD (FLASH Diskgroup	N/A	N/A	1,492
Total HDD with **High** Redundancy	3,727	5,587	19,200
Total HDD with **Normal** Redundancy	5,590	8,380	28,800
DATA Diskgroup with **High** Redundancy - External Backup	3,205	4,805	16,512
RECO Diskgroup with **High** Redundancy - External Backup	522	782	2,688
DATA Diskgroup with **High** Redundancy - Local Backup	1,603	2,402	8,256
RECO Diskgroup with **High** Redundancy - Local Backup	2,124	3,185	10,944
DATA Diskgroup with **Normal** Redundancy - External Backup	4,807	7,207	24,768
RECO Diskgroup with **Normal** Redundancy - External Backup	783	1,173	4,032
DATA Diskgroup with **Normal** Redundancy - Local Backup	2,404	3,603	12,384
RECO Diskgroup with **Normal** Redundancy - Local Backup	3,186	4,777	16,416
REDO Diskgroup	91	248	248
FLASH Diskgroup	N/A	N/A	746

Note: High Redundancy is triple-mirroring and Normal Redundancy is double-mirroring. REDO Diskgroup is always High Redundancy. FLASH Diskgroup is always Normal Redundancy.

See Also: "Managing Capacity in Disk Groups" in *Oracle Automatic Storage Management Administrator's Guide* for more information about determining the amount of free space in your Oracle Database Appliance ASM diskgroups. Table A–3 only provides the raw disk information.

Storage on Oracle Database Appliance

Oracle Database Appliance uses the ASM Cluster File System (ACFS) for storage of database and virtual machine files. ACFS provides both servers with concurrent access to some or all of the shared storage on Oracle Database Appliance. ACFS supports space-efficient storage snapshots, which provides fast provisioning databases and virtual machines within Oracle Database Appliance.

There are three types of ACFS file systems that are used in Oracle Database Appliance:

- Database
- Shared Repositories
- General Purpose Storage

Database file systems are used exclusively for storing database files, and they include a FLASH file system for storing database data files and flash cache files, a DATA file system for database data files, a RECO file system for storing archive files and backups, and a REDO file system for storing redo log files.

Shared Repositories are file systems created on Oracle Database Appliance Virtualized Platform, and they are used to store virtual machine templates, run-time images, and virtual disks.

A general purpose cluster file system, cloudfs, is also created by default on every Oracle Database Appliance, and cloudfs can be used for general purpose storage that must be shared between the servers, for example, staging for data loads.

All ACFS file systems are created on ASM Dynamic Volumes provisioned from ASM diskgroups created in the shared disk storage pool. In a bare metal deployment, these file systems are mounted directly in the OS hosting the databases. In a virtualized deployment, these file systems are managed and mounted directly in ODA_BASE.

ACFS Mount Points

The various ACFS file systems are mounted in the OS, or in ODA_BASE (Virtualized Platform) in different locations. Table A–4 describes the various mount points and related ASM diskgroups and volume information.

Table A–4 ACFS Mount Points and Related ASM Diskgroups and Volume Information

File System	ASM Diskgroup	ASM Dynamic Volume	Mount Point
DATA (Non-CDB)	+DATA	/dev/asm/datastore-<nnn>	/u02/app/oracle/oradata/datastore
RECO (Non-CDB)	+RECO	/dev/asm/datastore-<nnn>	/u01/app/oracle/fast_recovery_area/datastore
REDO (Non-CDB)	+REDO	/dev/asm/datastore-<nnn>	/u01/app/oracle/oradata/datastore
DATA (per CDB)	+DATA	/dev/asm/dat<dbname>-<nnn>	/u02/app/oracle/oradata/dat<dbname>
RECO (per CDB)	+RECO	/dev/asm/rco<dbname>-<nnn>	/u01/app/oracle/fast_recovery_area/rco<dbname>
REDO (per CDB)	+REDO	/dev/asm/rdo<dbname>-<nnn>	/u01/app/oracle/oradata/rdo<dbname>
FLASH	+FLASH	/dev/asm/flashdata-<nnn>	/u02/app/oracle/oradata/flashdata
Shared Repository <name>	+DATA or +RECO	/dev/asm/<reponame>-<nnn>	/u01/app/sharedrepo/<reponame>
General ACFS Storage	+RECO	/dev/asm/acfsvol-<nnn>	/cloudfs (default)

Space Management

The ACFS file systems are automatically created when you create a shared repository, or when you create a database. However, the file systems do not initially consume all of the storage in the appliance, preserving space for additional repositories, or in some cases, database files stored directly in ASM. In Oracle Database Appliance software releases 12.1.2.2 and earlier, the ACFS file systems do not automatically extend should they run low on space, even if there is still storage space available in the shared storage pool. You can check for available storage space in your file systems by running the OS command `df -h` as shown in the following example.

```
df -h
Filesystem        Size  Used Avail Use% Mounted on
/dev/mapper/VolGroupSys-LogVolRoot
                  30G  8.6G   20G  31% /
/dev/mapper/VolGroupSys-LogVolU01
                  97G   23G   70G  25% /u01
/dev/mapper/VolGroupSys-LogVolOpt
                  59G  6.8G   49G  13% /opt
/dev/sda1         99M   26M   68M  28% /boot
tmpfs            127G  1.2G  125G   1% /dev/shm
/dev/asm/datafsvol-352
                 5.0G   87M  5.0G   2% /odadatafs
/dev/asm/rdocdb1-66
                 5.0G  4.2G  874M  83% /u01/app/oracle/oradata/rdocdb1
/dev/asm/datcdb1-303
                 100G  4.2G  96G    4% /u02/app/oracle/oradata/datcdb1
/dev/asm/flashdata-138
                 558G  1.4G  557G   1% /u02/app/oracle/oradata/flashdata
/dev/asm/rcocdb1-352
                 132G  788M  132G   1% /u01/app/oracle/fast_recovery_area/rcocdb1
/dev/asm/acfsvol-352   50G  178M   50G   1% /cloudfs
/dev/asm/datastore-66
                 59G  4.3G   55G   8% /u01/app/oracle/oradata/datastore
/dev/asm/datastore-303
                 3.6T  2.1G  3.6T   1% /u02/app/oracle/oradata/datastore
/dev/asm/datastore-352
```

```
                  4.8T   10G  4.8T   1% /u01/app/oracle/fast_recovery_
area/datastore
```

If you need to extend the size of one of your file systems to accommodate growth in your data, you must manually expand the pool.

See Also:

- "oakcli resize dbstorage" on page 5-34 to expand space for database files
- "oakcli configure repo" on page 5-9 to expand shared repositories
- My Oracle Support note 1437717.1 "Expanding /cloudfs File System" at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1437717.1

System Configuration for Oracle Database Appliance

Table A–5 System Configuration for Oracle Database Appliance

Item	Value
Oracle Linux with the Red Hat compatible kernel	Oracle Linux 5.11 UEK2

Table A–5 (Cont.) System Configuration for Oracle Database Appliance

Item	Value
Oracle Grid Infrastructure and Oracle Database Version (initial release)	■ Release 2.1: Oracle Database 11*g* Release 2 (11.2.0.2), with PSU 3 and patches 12639177 (ASM), 12914151 (MLR on Grid Infrastructure PSU3), and 12419331 (Database PSU3) ■ Release 2.1.0.3: Oracle Database 11*g* Release 2 (11.2.0.2), with PSU5 ■ Release 2.1.0.3: Oracle Database 11*g* Release 2 (11.2.0.2), with PSU7 ■ Release 2.2: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU2 ■ Release 2.3: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU3 ■ Release 2.4: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU4 ■ Release 2.5: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU5 ■ Release 2.5.5: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU5 ■ Release 2.6: Oracle Database 11*g* Release 2 (11.2.0.3), with PSU6 ■ Release 2.7: Oracle Database 11*g* Release 2 (11.2.0.2.11 and 11.2.0.3.7), with PSU7 ■ Release 2.8: Oracle Database 11*g* Release 2 (11.2.0.2.12 and 11.2.0.3.8 with PSU8, and 11.2.0.4) ■ Release 2.9: Oracle Database 11*g* Release 2 (11.2.0.2.12, 11.2.0.3.9 and 11.2.0.4.1), with DB PSU 1 ■ Release 2.10: Oracle Database 11*g* Release 2 (11.2.0.4.2 with PSU2), optionally Oracle Database 11*g* Release 11.2.0.2 with PSU9, and 11.2.0.2.12 and 11.2.0.3.10 ■ Release 12.1.2: Oracle Database 12*c* Release 1 (12.1.0.2.0) and optionally 11.2.0.2.12, 11.2.0.3.11, and 11.2.0.4.3 ■ Release 12.1.2.1.0: Oracle Database 12*c* Release 1 (12.1.0.2) with PSU1, optionally Oracle Database 11*g* Release 2 (11.2.0.4) with PSU4, and Oracle Database 11*g* Release 2 (11.2.0.3) with PSU12 ■ Release 12.1.2.2.0: Oracle Database 12*c* Release 1 (12.1.0.2) with PSU2, optionally Oracle Database 11*g* Release 2 (11.2.0.4) with PSU5, and Oracle Database 11*g* Release 2 (11.2.0.3) with PSU13
SCAN port number	1521
Oracle Enterprise Edition DB control port	1158 Use: `https://`*`hostname`*`:1158/em`, where *`hostname`* is the name of one of the Oracle Database Appliance server nodes.

B

Database Sizing for Oracle Database Appliance

Use the information in this appendix to select database templates for your planned databases. Oracle strongly recommends that you use the Oracle Database Appliance templates, because they implement best practices and are configured specifically for Oracle Database Appliance.

The templates not only define databases with parameters that have been selected specifically to optimize performance on Oracle Database Appliance but also help you to set up appropriate instance caging, to create ODA_BASE with a matching template on Oracle Database Appliance Virtualized Platform, and to acquire an appropriate license on bare metal installations.

> **Tip:** Oracle Appliance Manager configurator refers to the database sizing templates as *classes* of databases.

Choosing a Database Template

With its multiple CPUs, Oracle Database Appliance enables you to consolidate many databases onto a single system. Consolidation can minimize idle resources, maximize efficiency, and lower costs. By using instance caging, in conjunction with Oracle Database Resource Manager (the Resource Manager) you can provide the desired levels of service across multiple instances on a single Oracle Database Appliance.

Oracle Database Appliance templates are already tuned for the size of each database instance workload. They are designed to run on a specific number of cores. Caging assures that each database workload is restricted to the set of cores allocated by the template, enabling multiple databases to run concurrently with no performance degradation, up to the capacity of Oracle Database Appliance. You could select database template sizes larger than your current needs to provide for planned growth, which you accommodate later by adjusting System Global Area (SGA) and Program Global Area (PGA) sizes as well as the number of cores.

> **Note:** Oracle strongly recommends that you use the Oracle Database Appliance templates, because they implement best practices and are configured specifically for Oracle Database Appliance.

Use the following tables to help select the best templates for your databases. When using these tables remember:

- The information in the tables assume that you are creating disk backups. The information in the tables assume that you are creating local disk backups.

Consider the space requirements for your database and the policy for local disk backups vs external backups. With external backups, there is more space available for the database versus the local backups where backups are stored locally.

- Container databases are created on Oracle ACFS and the following tables show the amount space taken in the ACFS file system. Note that an Oracle ACFS file system can be extended to avoid running out of space.
- I/O per second (IOps) values are derived from an 8 K random read-write response time of 10-12 milliseconds for HDDs, and less than 1 millisecond for Flash, for Oracle Database Appliance X5-2; 5-7 milliseconds for Oracle Database Appliance X4-2 and X3-2; and 5 milliseconds for Oracle Database Appliance Version 1 to service an online transaction processing (OLTP) I/O request. The rates are not based on the number of CPUs and assume that the system is running at capacity.
- Throughput, in MB per second (MBps). is based on a 1 MB sequential read/write for a data warehousing system. As with IOps, the MBps is a measure of throughput when the system is at capacity. With just a single small database running, the MBps would be the maximum available on the system for a large database.
- The log file size assumes four redo log groups for each instance with a log switch every 15 minutes when the system is running at full capacity.

Find the information that is appropriate to your database workload and hardware:

- Use Table B–2 if your database workload is of type OLTP
- Use Table B–3 if your database workload is of type DSS (Data Warehouse)
- Use Table B–4 if your database workload is of type In-Memory
- Use Table B–5 if your database is on Oracle Database Appliance X5-2
- Use Table B–6 if your database is on Oracle Database Appliance X4-2
- Use Table B–7 if your database is on Oracle Database Appliance X3-2
- Use Table B–8 if your database is on original model of Oracle Database Appliance.

Note: Oracle Appliance Manager Release 12.1.2.2.0 provides improved database templates that are specific to the type of database workload (OLTP, DSS, In-Memory) rather than the generic templates that were used in the earlier releases. The database sizing tables provide updated template names and sizing based on the number of CPUs and memory attributes for each type of database workload. Table B–1 provides a reference between the generic database template names and the database template names based on CPU cores.

Table B–1 Oracle Database Appliance Database Template Name Conversions

CPU Core-Based Database Template Names	odb-01s	odb-01	odb-02	odb-04	odb-06	odb-12	odb-16	odb-24
Generic Database Template Names	Very Very Small	Very Small	Small	Medium	Large	Extra Large	Extra Extra Large[1]	Extra Extra Extra Large[1]

[1] Extra Extra Large is applicable to Oracle Database Appliance X3-2 and Oracle Database Appliance X4-2. Extra Extra Extra Large is applicable to Oracle Database Appliance X4-2 only.

Table B–2 Oracle Database Appliance OLTP Database Template Size

Template	CPU Cores	SGA (GB)	PGA (GB)	Flash (GB)	Processes	Redo log file size (GB)	Logbuffer (MB)
odb-01s (All Hardware Versions)	1	2	1	6	200	1	16
odb-01 (All Hardware Versions)	1	4	2	12	200	1	16
odb-02 (All Hardware Versions)	2	8	4	24	400	1	16
odb-04 (All Hardware Versions)	4	16	8	48	800	1	32
odb-06 (All Hardware Versions)	6	24	12	72	1200	2	64
odb-12 (All Hardware Versions)	12	48	24	144	2400	4	64
odb-16 (X5-2, X4-2, X3-2 Only)	16	64	32	192	3200	4	64
odb-24 (X5-2, X4-2 Only)	24	96	48	192	4800	4	64
odb-32 (X5-2 Only)	32	128	64	192	6400	4	64
odb-36 (X5-2 Only)	36	128	64	192	7200	4	64

Note: Flash is applicable to Oracle Database Appliance X5-2 only.

Table B–3 Oracle Database Appliance DSS Database Template Size

Template	CPU Cores	SGA (GB)	PGA (GB)	Processes	Redo log file size (GB)	Logbuffer (MB)
odb-01s (All Hardware Versions)	1	1	2	200	1	16
odb-01 (All Hardware Versions)	1	2	4	200	1	16
odb-02 (All Hardware Versions)	2	4	8	400	1	16
odb-04 (All Hardware Versions)	4	8	16	800	1	32
odb-06 (All Hardware Versions)	6	12	24	1200	2	64
odb-12 (All Hardware Versions)	12	24	48	2400	4	64
odb-16 (X5-2, X4-2, X3-2 Only)	16	32	64	3200	4	64
odb-24 (X5-2, X4-2 Only)	24	48	96	4800	4	64
odb-32 (X5-2 Only)	32	64	128	6400	4	64
odb-36 (X5-2 Only)	36	64	128	7200	4	64

Table B–4 Oracle Database Appliance In-Memory Database Template Size

Template	CPU Cores	SGA (GB)	PGA (GB)	In-Memory (GB)	Processes	Redo log file size (GB)	Logbuffer (MB)
odb-01s (All Hardware Versions)	1	2	1	1	200	1	16
odb-01 (All Hardware Versions)	1	4	2	2	200	1	16
odb-02 (All Hardware Versions)	2	8	4	4	400	1	16
odb-04 (All Hardware Versions)	4	16	8	8	800	1	32
odb-06 (All Hardware Versions)	6	24	12	12	1200	2	64
odb-12 (All Hardware Versions)	12	48	24	24	2400	4	64
odb-16 (X5-2, X4-2, X3-2 Only)	16	64	32	32	3200	4	64
odb-24 (X5-2, X4-2 Only)	24	96	48	48	4800	4	64
odb-32 (X5-2 Only)	32	128	64	64	6400	4	64
odb-36 (X5-2 Only)	36	128	64	64	7200	4	64

Table B–5 Oracle Database Appliance X5-2 Database Template Size Storage Performance

Template	Number of databases using this template that you can deploy	Container Database ACFS Size on DATA Diskgroup (GB)	I/O per second with single storage shelf (HDD/FLASH)	Throughput (MB/second) with single storage shelf (HDD/FLASH)	I/O per second with storage shelf plus storage expansion shelf (HDD/FLASH)	Throughput (MB/second) with storage shelf plus storage expansion shelf (HDD/FLASH)	Log generation (MB/second)
odb-01s	36	100	42/4167	83/83	83/8333	167/167	6.83
odb-01	36	100	42/4167	83/83	83/8333	167/167	6.83
odb-02	18	200	83/8.3K	167/167	167/16.7K	333/333	6.83
odb-04	9	400	167/16.7K	333/333	333/33.3K	667/667	13.65
odb-06	6	800	250/25K	500/500	500/50K	1000/1000	27.3
odb-12	3	1600	500/50K	1000/1000	1000/100K	2000/2000	27.3
odb-16	2	1600	750/75K	1500/1500	1500/150K	3000/3000	27.3
odb-24	1	1600	1500/150K	3000/3000	3000/300K	6000/6000	27.3
odb-32	1	1600	1500/150K	3000/3000	3000/300K	6000/6000	27.3
odb-36	1	1600	1500/150K	3000/3000	3000/300K	6000/6000	27.3

Note: Actual IOps and throughput for a given database will depend on the percentage of I/O operations that are serviced through the FLASH versus the HDDs. An OLTP workload whose working set fits in FLASH will be 80% reads (serviced through the flash cache) and 20% writes (serviced through the HDDs). Thus, the actual IO performance will be weighted appropriately.

Table B–6 Oracle Database Appliance X4-2 Database Template Size Storage Performance

Template	Number of databases using this template that you can deploy	Container Database ACFS Size on DATA Diskgroup (GB)	I/O per second with single storage shelf	Throughput (MB/second) with single storage shelf	I/O per second with storage shelf plus storage expansion shelf	Throughput (MB/second) with storage shelf plus storage expansion shelf	Log generation (MB/second)
odb-01s	24	100	138	146	275	229	6.83
odb-01	24	100	138	146	275	230	6.83
odb-02	12	200	275	292	550	458	6.83
odb-04	6	400	550	583	1100	917	13.65
odb-06	4	800	825	875	1650	1375	27.3
odb-12	2	1600	1650	1750	3300	2750	27.3
odb-16	1	1600	3300	3500	6600	5500	27.3
odb-24	1	1600	3300	3500	6600	5500	27.3

Table B–7 Oracle Database Appliance X3-2 Database Template Size Storage Performance

Template	Number of databases using this template that you can deploy	Container Database ACFS Size on DATA Diskgroup (GB)	I/O per second with single storage shelf	Throughput (MB/second) with single storage shelf	I/O per second with storage shelf plus storage expansion shelf	Throughput (MB/second) with storage shelf plus storage expansion shelf	Log generation (MB/second)
odb-01s	16	100	206	219	413	344	6.83
odb-01	16	100	206	219	413	344	6.83
odb-02	8	200	413	438	825	688	6.83
odb-04	4	400	825	875	1650	1375	13.65
odb-06	2	800	1650	1750	3300	2750	27.3
odb-12	1	1600	3300	3500	6600	5500	27.3
odb-16	1	1600	3300	3500	6600	5500	27.3

Table B–8 Oracle Database Appliance Version 1 Database Template Size Storage Performance

Template	Number of databases using this template that you can deploy	Container Database ACFS Size on DATA Diskgroup (GB)	I/O per second	Throughput (MB/second)	Log generation (MB/second)
odb-01s	12	100	333	250	6.83
odb-01	12	100	333	250	6.83
odb-02	6	200	667	500	6.83
odb-04	3	400	1333	1000	13.65
odb-06	2	800	2000	1500	27.3
odb-12	1	1600	4000	3000	27.3

For a single database, select a template that best fits your expected workload on the hardware you are using in terms of required CPU and IOps. When creating multiple databases, the overall workload will be affected by the CPU and IOps consumed by the existing databases that are already on the system.

Index

P

R

S

T

U

V

W