Oracle® Certification Environment

OAST Installation and User's Guide for Oracle Database

12*c* Release 1 (12.1) for Linux and UNIX Systems

E39367-01

May 2014

Oracle Certification Environment OAST Installation and User's Guide for Oracle Database, 12*c* Release 1 (12.1) for Linux and UNIX Systems

E39367-01

Primary Author: Bharathi Jayathirtha

Contributor: Janelle Simmons, Jason Wu, Jing Liu, Lewis Kaplan, and Yongfeng Zhou

Contents

6 Running OAST Workloads

7 Tuning OAST Workloads to Stress System Resource

8 Configuring OAST Database and Workloads for Oracle Database 12*c* Certification

9 Troubleshooting

A Examples of OAST Installation

B CDB and PDB Utilities

List of Tables

Preface

Use this guide to certify the system for Oracle Database 12*c* Release 1 (12.1) and Oracle Real Application Clusters 12*c* Release 1 (12.1). This guide describes how to install the Oracle product software, the Oracle Certification Environment (OCE) OAST software, and how to certify the system.

Audience

This guide is intended for software engineers who are certifying Oracle Database 12*c* Release 1 (12.1) on UNIX and Linux systems.

This guide assumes that users of Oracle Certification Environment (OCE) kit have no prior experience of certifying Oracle products on UNIX and Linux systems. However, users must have Oracle Database and Oracle RAC experience before starting certification work.

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 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

This guide contains information about installing Oracle Automated Stress/System Testing (OAST) software, which is part of the OCE kit for Oracle Database 12*c* Release 1 and Oracle Real Application Clusters for Oracle Database 12*c* Release 1. It also describes how to run the certification tests.

Installation and administrative tasks for Oracle products are provided in the following guides:

The following documentation and all the relevant additional documentation is available on the Oracle Technology Network (OTN) website:

`http://www.oracle.com/technetwork/indexes/documentation/index.html`

- *Oracle Database Client Installation Guide*

- *Oracle Database Installation Guide*
- *Oracle Database Release Notes*
- *Oracle Grid Infrastructure Installation Guide*
- *Oracle Real Application Clusters Installation Guide*

Ordering Documentation

To order documentation, call the appropriate number:

- In the United States, call Documentation Sales at 1.800.252.0303.
- In the United Kingdom, call Oracle Direct Response at +44.990.332200.
- In other European countries, contact your local Oracle Support office.
- In the Asia-Pacific region, contact your Oracle sales representative.

Shipping Enquiries

For shipping inquiries, product exchanges, or returns, call the appropriate number:

- In the United States, call Client Relations at 1.650.506.1500.
- In the United Kingdom, call Customer Relations at +44.990.622300.
- In other European countries, contact your local Oracle Support office.
- In the Asia-Pacific region, contact your Oracle sales representative.

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 text or the glossary.
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.

1 Introduction to OAST

This chapter provides an overview of Oracle Automated Stress/System Testing (OAST) for certification testing and contains information about the following topics:

- Section 1.1, "Overview of OAST"
- Section 1.2, "What's New in OAST Kit Version 12.1"
- Section 1.3, "OAST Certification Requirements"
- Section 1.4, "OAST Environment Variables"
- Section 1.5, "Checklist for OAST"

1.1 Overview of OAST

OAST is an Oracle partner standard stress test suite intended to test Oracle Database on a particular system by simulating a real world environment.

OAST includes a set of test drivers for stress testing platform specific CPU, I/O, Interconnect, and Memory subsystem when you run Oracle database server software.

OAST workloads can be tuned to maximize the load on all nodes within a cluster or a single node, when running a single instance database.

OAST takes a user input of the system and feeds them into its built-in sizing engine. The sizing engine produces an input parameter (number of warehouses) to the stress OLTP kit generator, which creates a stress OLTP kit. This OLTP kit is then used to prepare the database, create the schema, populate the tables and views, and create indexes. The run script (`nrunoastoltp`*`XXX`*`.sh`) prepares database backup and restore, performs stress test runs, and transaction driver recompilation.

> **Note:** In 12*c* Release 1, OAST run script for CDB-enabled configurations is `oast_main.sh`.

1.2 What's New in OAST Kit Version 12.1

This section provides details about the new features included in the OAST kit. It includes the following topics:

OAST kit version 12.1 is supplied with an updated workload client that has the following features:

1. Oracle Multitenant option: This is a multitenant container database (CDB) that includes one or many customer-created pluggable databases (PDBs). A PDB is a portable collection of schemas, schema objects, and nonschema objects that

appears to an Oracle Net Client as a non-CDB. All Oracle databases before Oracle Database 12*c* Release 1 are non-CDBs.

2. Offline loading: Previous release of OAST used online loading during OAST database population. The current release provides a better way to implement the data loading, which is offline loading. A user can choose to use offline loading, generate all data with `.csv` format, and then use the `sqlldr` option to import data into the database. This feature saves time and the cost of loading. Oracle Database 12*c* Release 1 (12.1) and Oracle Database 11*g* Release 2 (11.2) supports this feature.
3. Hybrid Columnar Compression support: If you enable this option during OAST installation, then you must ensure that the storage used supports this feature.
4. OS Watcher (OSW) integration: OS Watcher Black Box integrated with this version generates a profile containing operating system and network metrics after completing each workload.

In addition, the OAST version numbering has been aligned with the latest Oracle Database release. This is to improve version clarity. However, OAST will continue to be backward compatible with supported Oracle Database releases, such as, Oracle Database 11*g* Release 2 (11.2) and Oracle Database 11*g* Release 1 (11.1).

1.3 OAST Certification Requirements

To achieve OCE standards, OAST must satisfy the standards mentioned in this section.

Hardware and Software Configurations:

The vendor hardware system must be capable of running a balanced database workload with sufficient memory, CPU, network, and storage resources. A separate client system is required to run OAST stress workloads in client/server mode for both Single Instance and Oracle RAC configurations. For Oracle RAC testing, you must run the OAST stress tests and Oracle RAC destructive tests on the same hardware configuration. Refer to the *Destructive and Stress Test Plan for Vendor Compatibility With Oracle RAC 12cR1* document for specific details. You can download this document from the following location:

`http://www.oracle.com/technetwork/database/index.html`

The Oracle database configuration required to run OAST stress workloads depends on the vendor technology stacks that needs to be certified.

Oracle and the vendor will work together to define the minimum requirement for Oracle RAC 12*c* Release 1 configuration. The configuration must be recorded when initiating the certification project using Certification Management System (CMS).

Workload Requirements:

1. Run OAST workloads successfully for 48 hours to stress the system resource to the required level on a Single instance or an Oracle RAC configuration as follows:

   a. The CPU utilization must be greater than or equal to 90 percent.

   b. Network traffic (interconnect and storage usage throughput) must reach approximately 70 percent of their capacities. It is recommended to use Oracle Orion to calibrate the I/O subsystem throughput.

   Multiple OAST stress workloads, such as, OLTP, DSS, and Oracle RAC Interconnect stress workload may be required to run concurrently to achieve the stress level. Refer to Chapter 7, "Tuning OAST Workloads to Stress System

Resource" and Chapter 8, "Configuring OAST Database and Workloads for Oracle Database 12c Certification" for recommendations.

2. For an Oracle RAC configuration, you must run the Oracle RAC elevator test for 12 hours.

3. A minimum 100 GB database size.

1.4 OAST Environment Variables

Table 1–1 provides details about a few environment variables used with OAST. For a detailed list of the environment variables, refer to the `oastinstall.rsp` file located in the `install/stage/responsefile` directory.

Table 1–1 OAST Environment Variables

Variable	Type	Definition	Example
`OAST_HOME`	String	Specifies the directory that contains the necessary OAST configuration files.	`OAST_HOME=/home/oracle/oast/home`
`csFlag`	String	Enables you to set OAST on a Client/Server environment. The value can be `Yes` or `No`.	`csFlag=Yes`
`CLIENTHOME`	String	Specifies the location of the client Oracle home.	`CLIENTHOME=/u01/app/product/12.1.0`
`cnodesStr`	String	Specifies the client host name.	`cnodesStr=`*`racclient`*
`cOAST_USER`	String	Specifies the Oracle Client home directory owner.	`cOAST_USER=`*`oracle`*
`ORACLE_HOME`	String	Specifies the directory containing the Oracle software.	`ORACLE_HOME=/u01/app/oracle/product/12.1.0`
`ORACLE_BASE`	String	Specifies the Oracle base directory structure.	`ORACLE_BASE=/u01/app/oracle`
`OASTDB_NAME`	String	Specifies the OAST database name.	`OASTDB_NAME=oastdb`
`optionFlag`	String	Enables you to set up OAST on a Cluster environment. The values can be `"Cluster"` or `"Non-Clustered"`	`optionFlag="Cluster"`
`cdb_flag`	String	Used to enable or disable the Container Database feature. This applies to only Oracle Database Release 12*c*. The values can be `Yes` or `No`.	`cdb_flag="Yes"`
`hcc_flag`	String	Used to enable or disable the HCC feature for supported storage. The values can be `Yes` or `No`.	`hcc_flag="No"`

1.5 Checklist for OAST

After successfully running the OAST tests, you must complete the OAST section within the certification checklist. You can download the checklist from the following website:

`http://www.oracle.com/technetwork/database/index-089070.html`

The certification checklist helps you to record the successful completion of all the tests. If you have any questions related to the checklist, contact the OCE group at certsup_ww@oracle.com.

2

OAST Preinstallation Requirements

This chapter describes the preinstallation requirements for installing OAST. It includes information about the following topics:

- Section 2.1, "Software Requirements"
- Section 2.2, "OAST Distribution"
- Section 2.3, "Disk Space Requirements"
- Section 2.4, "OAST Workloads"

2.1 Software Requirements

Before installing OAST, you must install the following Oracle products depending on the type of testing you want to perform:

- Oracle Database
- Oracle Database Client

  > **Note:** Oracle Database Client software is required to install OAST in the client/server mode.
  >
  > You must select the Administrator installation type to install Oracle Database Client 12*c* Release 1.

- Oracle Grid Infrastructure

  > **Note:** Oracle Grid Infrastructure is required to install OAST in an Oracle RAC environment.

- The necessary Compiler and Loader

Before you install OAST in client/server mode, the client/server must meet the following requirements:

- The installation must use the same user name that was used when installing Oracle Database on a server or a cluster.
- Passwordless SSH connectivity must be enabled between all the nodes and the client, and from the client to all the nodes.
- The following information is required during OAST installation:
  - Client

Note: The Client information is required for installing OAST in client/server mode.

- Oracle home directory where Oracle Database Client is installed.
- The user name that was used to install the client.
- OAST home directory.
- C Compiler and Linker to build the OAST OCI Driver program.

– Server

- Oracle home directory where Oracle Database is installed.
- Oracle base location where Oracle Database is installed.
- Oracle Database name.
- Oracle Automatic Storage Management disk groups or file system device names. A minimum of 1 disk group or file system device name is required.
- OAST home directory.
- C Compiler and Linker to build the OAST OCI Driver program.

When you run `oastinstall` to install OAST in client/server mode, the driver is distributed to the client during the installation process. The Oracle Database version and the Oracle Database Client version must be identical. If you have any questions related to the installation, then contact the Oracle Certification Environment Group at certsup_ww@oracle.com.

When you run `oastinstall` to install OAST in server/server mode in an Oracle RAC environment, the driver is distributed during the installation process to all the nodes in the cluster.

You can run OAST for all the storage types supported by Oracle Database, such as, NFS, dNFS, Oracle Cluster File System, Oracle Automatic Storage Management, and Oracle Automatic Storage Management Cluster File System (Oracle ACFS). Table 2–1 lists the prerequisites for the storage types. OAST enables Oracle Managed Files.

OAST is restricted to data file creation within a single mount, share, or Oracle Automatic Storage Management disk group. Use the `db_create_file_dest` parameter to create the destination location. If additional mount points or file locations are needed, then refer to Section 9.4.2, "Adding Data Files to Support Multiple Devices."

Table 2–1 Prerequisite for storage types before installing OAST

Storage Type	Prerequisite
Oracle Automatic Storage Management Cluster File System	Create the shared volumes and validate access for each node in the cluster.
Oracle Automatic Storage Management	Create the required disk groups before you install OAST. You can use a single disk group to store both the data files and the redo logs.
Cluster File System	Create the shared volumes and validate access for each node in the cluster.

Table 2–1 (Cont.) Prerequisite for storage types before installing OAST

Storage Type	Prerequisite
NFS	Create the NFS mounts and validate access for each node in the cluster.
dNFS	Refer to *Oracle Database Administrator's Guide* for details about enabling dNFS.

2.2 OAST Distribution

The OAST installation contains the following directories and files:

Directory and Files	Contains
home	The OAST home directory.
install	The installation script and other support directories. OAST is installed in the OAST home directory.
orion	The Oracle Orion scripts that are used to verify the I/O throughput.
`readme.oast`	The readme file.

> **Note:** If you install multiple databases, then each database must reside in a unique OAST home directory. OAST automatically overwrites an existing OAST home location without prompting for confirmation.

2.3 Disk Space Requirements

The following are the disk space requirements for OAST on a 100 GB database, which is equal to 1500 warehouses on an Oracle Database 11*g* Release 2 (11.2) and 500 warehouses on an Oracle Database 12*c* Release 1 (12.1).

Table 2–2 Disk Space Requirements for OAST Installation on a Server

Item	Disk Space Requirement
OAST software	50 MB
OAST home	50 MB
Total Install	100 MB

An example for disk space requirements for OAST installation on a server is listed in Table 2–3. The server details are as follows:

- 1500 Warehouses, which is equal to 100 GB seed
- 4 Nodes
- 2 Log Groups
- 1 Member per Log Group

Table 2–3 An example of Disk Space Requirements for OAST Installation on a Server

Item	Disk Space Requirement
Seed data	100 GB
Logs	2 GB
Free space	30 GB
Total space without archiving	132 GB
Archive Directory	200 GB
Total space with archiving	332 GB

Note: Table 2–2 and Table 2–3 do not include the disk space requirements for the Oracle product software. For more information about disk space requirements, refer to the documents listed in "Related Documents" on page 2-vii.

Table 2–4 An example of the number of warehouses that a 100 GB OAST database contains in Oracle Database 11g Release 2 (11.2) against Oracle Database 12c Release 1 (12.1)

OAST version	Database Name	Number of warehouses
11.2	`oastdb`	1500
12.1	`oastdb`	500

2.4 OAST Workloads

The details of the OAST workloads that are required to run destructive testing are as follows:

- OLTP - workload 1

  The OLTP workload is used to drive IOPS against the storage subsystem that contains the database and the redo files. Using workload 1, OAST will execute the transactions in terms of small random access, which in turn drives IOPS. OAST maintains a very consistent rate of IOPS by using a working set that is larger than the current buffer cache size. OAST achieves this by randomly selecting primary key values before execution of the transaction. This randomization produces a high probability that the block needed for the current transaction is not in the buffer cache, provided the working set is larger than the buffer cache.

- I/O/CPU - workload 2

  The I/O and CPU workload is used to drive both IOPS and CPU. The small random I/O and CPU stress is obtained by utilizing a working set that allows for each thread to have distinct warehouse and district IDs. These IDs are permanent and do not change when a new transaction is generated for a particular client.

- Interconnect - workload 3

Note: This workload is different from workload 6 and workload 7.

This workload does not provide high interconnect traffic, but generates a workload that causes all clients to access the same warehouse, district, and customer. This will force cache buffer exchanges, but will not generate any significant data transfer due to compression and other cache fusion performance enhancements. This workload causes a significant amount of cache fusion locking.

- CPU - workload 4

  The CPU workload is used to drive high CPU consumption by using an in-memory database. A CPU-only workload is achieved by using a read only weighted transactional deck and using a working set that is smaller than the buffer cache. The resulting workload will show initial read activity as the buffer cache is primed with data. After the buffer cache is primed, any requests for blocks will come directly from the buffer cache and not from the disk. The primary keys are consistent and not regenerated on each pass.

- Decision Support System - workload 5

  The Decision Support System (DSS) workload is used to drive MBPS against the storage subsystem containing the database. Using workload 5, OAST executes full table selects that aggregate the individual rows of the tables. With Parallel Query enabled, the DSS workload uses the PGA as the receiving area for the data read. Direct Path reads are executed and the buffer cache is bypassed with multiple parallel query slaves providing additional throughput by allowing concurrent reads from the disk.

- Consistent Read Transfer Single block Workload - workload 6

  The CRSTRESS single block workload is used to drive cache fusion traffic across the interconnect against the storage subsystem containing the database. Using workload 6, each node executes a full table scan from an in-memory table residing on another node. The full table scan causes the node to request a single block data transfer from another node. This action generates significant interconnect traffic.

- Consistent Read Transfer Multiblock Workload - workload 7

  The CRSTRESS multiblock workload is used to drive cache fusion traffic across the interconnect against the storage subsystem containing the database. Using workload 7, each node executes a full table scan from an in-memory table residing on another node. The full table scan causes the node to request a multi-node block data transfer from another node. This action generates significant interconnect traffic.

3 Installing OAST

You can install OAST software in server/server mode or client/server mode. Before you install, ensure that your system meets the preinstallation requirements as described in Chapter 2, "OAST Preinstallation Requirements."

Before you start the installation, ensure that you have performed all the necessary preinstallation tasks. It is important that the system meets all the requirements as described in *Oracle Grid Infrastructure Installation Guide*, *Oracle Database Installation Guide*, and *Oracle Real Application Clusters Installation Guide* for a successful installation. Refer to your platform-specific documentation for more details.

The vendor must validate the hardware and the supporting software.

Note: Running OAST on a system that does not meet the hardware requirements may negatively impact the ability of OAST to generate stress.

This chapter includes the following topics:

- Section 3.1, "Installing OAST Software in Server/Server Mode"
- Section 3.2, "Installing OAST Software in Client/Server Mode"
- Section 3.3, "Installing OAST Software in Silent Mode"

3.1 Installing OAST Software in Server/Server Mode

This section contains the OAST installation procedure in the server/server mode.

See Also: Section 3.2, "Installing OAST Software in Client/Server Mode" for installing OAST in the client/server mode

To install OAST, perform the following tasks on the primary node:

1. Download the OAST archive to a suitable location (for example, `$HOME/oast`) and extract the archive as follows:

```
$ su -oracle_install_username
$ cd $HOME/oast
$ unzip OAST_VERSION_PRODUCTION.zip
```

Note: The value for *VERSION* changes for each release of OAST.

2. Enter the password.

   To obtain a valid password, contact the OCE group at certsup_ww@oracle.com.

3. Enter the following commands to run the OAST installation script:

   ```
   $ cd $HOME/oast/install
   $ ./oastinstall 2>&1 | tee oastinstall.txt
   ```

   > **Note:** Use the tee command to redirect the standard output and errors to a file. Use the resulting file to validate the output and review the errors that occurred during the installation process.

   Refer to Section 1.4, "OAST Environment Variables" for a list of environment variables.

4. Enter the OAST home directory. This location is used to install the generated scripts and workload. The default location is displayed within brackets and is used if you do not provide a value. If you install multiple OAST databases, then select a unique directory.

   > **Note:** If the environment variable OAST_HOME is not set and the directory you entered does not exist, then OAST creates the required directory. OAST overwrites the existing OAST home directory.

5. Client/Server environment?

   Enter 1 for Yes.

   Enter 2 for No.

   > **Note:** If you enter **2**, then the OAST driver is installed on the current server for a single instance or on all the nodes in the cluster for an Oracle RAC environment.

6. Enter the Oracle home directory for Oracle Database. This directory must exist.

   > **Note:** If you are using different Oracle homes for Oracle Automatic Storage Management and Oracle Database, then enter the Oracle home directory for Oracle Database. OAST only validates the existence of that directory.

7. Enter the Oracle base directory. This directory must exist. The Oracle base directory contains the diag and cfgtoollogs subdirectories.

8. Enter the OAST database name. The default database name is oastdb. If you are installing multiple workloads, then enter a unique database name.

9. Would you like to create OAST database as a Container database (CDB)?

   Enter 1 to enable.

   Enter 2 to disable.

   If you enable CDB, then you must enter values for the following questions:

How many pdbs do you want to create in this container db? [1-252]:

Please enter data size for pdb cdb_pdb1 [10 Gigabytes]:

(Enter the number of PDBs if you choose to enable CDB, then enter the data size for each PDB).

10. Would you like to enable Hybrid Columnar Compression?

    Note: The Hybrid Columnar Compression feature is supported only on storage of Exadata, ZFS, and Axiom.

    Enter 1 for Yes.

    Enter 2 for No.

11. Would you like to enable the fast recovery area?

    Enter 1 to enable.

    Enter 2 to disable.

    If you enable fast recovery, then you must enter values for the `db_recovery_file_dest` and `db_recovery_file_dest_size` parameters for the following questions:

    Enter the Flash Recovery Destination: The value entered depends on the storage solution used. No default value is provided.

    Enter the diskgroup for Oracle Automatic Storage Management: Enter the disk group with a + symbol. For example, `+FRA`.

    or

    Enter the File system: Enter the mount, the Cluster File System (CFS), or the local directory location.

    Enter the Flash Recovery Size in MB: Enter only a numeric value to indicate the size in megabytes.

    If you enable the flash recovery area, then OAST uses the location entered for `db_recovery_file_dest` parameter as the location for `log_archive_dest` without prompting. However, if you disable flash recovery area, then OAST will prompt you for the `log_archive_dest` location. This destination directory must exist. OAST does not create or validate this directory.

12. Enable database archive mode?

    Enter 1 to enable.

    Enter 2 to disable.

13. Select Oracle Database options:

    Enter 1 for Cluster (Oracle RAC).

    Enter 2 for Non-Clustered (single instance).

    If you enter **1** (Oracle RAC), then you must have the Oracle Grid Infrastructure running as OAST will query the nodes currently configured using the `olsnodes` binary and will automatically validate the connection to each node. If the grid stack is offline, then OAST will be unable to automate this process and will prompt for node names.

    For Oracle RAC, OAST will create the `oast_home/oast_cluster` directory and several supporting subdirectories on all nodes in the cluster.

14. CPU numbers and Free System Memory (in megabytes) are automatically generated based on the amount of operating system memory on the local server.

    Total System Memory (megabytes) is 14651 MB.

15. Enter the percentage of total memory. Specify the memory that you want to allocate to OAST database. The default value is 33 percent. Enter a number between 10 and 90, do not use the % symbol.

    Note: For multiple databases on the same host, ensure that you have enough shared memory mount area (`/dev/shm`). The percentage of total memory used for database `orcl` (10-90) [33 %].

16. Target OLTP data size (gigabytes) is the OLTP database size that Oracle generates.

    The default database size is 10 GB of seed data. This value represents the amount of OLTP metadata the loader phase will create and does not include free space.

17. Database files storage type: file system or Oracle Automatic Storage Management

    Enter 1 for file system (that include NFS, dNFS, Oracle Cluster File System, local file system for single instance).

    Enter 2 for Oracle Automatic Storage Management (Oracle ASM).

    Option 1: file system

    If you select file system, then the following questions are applicable and you must provide the file system location for data files, the number of redo log groups, the number of redo log members, and the file system location for each redo log member.

    - Enter (Cluster) file system mount point for data files: For example, `/data/oastdb`.
    - Enter number of online redo log groups? [Default is 2 groups, max 255]: Number of Redo log groups for each instance of the database. OAST uses a redo log size of 500 MB.
    - The number of online redo log members in each group? [Default is 1 member, max 5]: If you enable multiple members, a value greater than 1, then you must enter a file system location for each member. Multiple members allow for multiplexing of redo logs and must be located on separate devices. Using the same device name for each member is not recommended.
    - Specify the file system location for log group member 1: For example: `/data/log1`

    Option 2: Oracle Automatic Storage Management

    If you select Oracle Automatic Storage Management (Oracle ASM), then enter the Automatic Storage Management disk group for data files and redo log members. Do not include the + sign when entering the disk group. For example, enter `DATA` if the disk group is `+DATA`.

    - Specify the Oracle Automatic Storage Management disk group name for data (without the leading + symbol).
    - Enter number of online redo log groups? [Default is 2 groups, max 255].
    - The number of online redo log members in each group? [Default is 1 member, max 5].

- If you enable multiple members, a value greater than 1, then you must enter a disk group location for each member. Multiple members allow for multiplexing of redo logs and must be located on separate devices. Using the same device name for each member is not recommended.
- Specify the Oracle Automatic Storage Management disk group name for log group member 1 (without the leading + symbol).

18. Select **Return** to confirm/continue or **e** to exit.

    OAST will print database sizing information for the current installation execution, including the number of warehouses and redo log details.

    To continue the installation, press the **Enter** key.

19. Verify the correctness of the information and select the appropriate number:

    - Continue
    - Modify
    - Exit

    OAST prints information for the current installation execution including total data file size, redo size for all nodes, and other operating system parameter information.

    OAST generates the driver kit and installs the necessary shell and support files across all the nodes in the configuration.

    OAST also executes the `elevator setup` script and copies the required files to all nodes in the cluster.

    If CDB was enabled during installation, then OAST will perform a silent installation for each PDB OAST home, and prints the guideline on the screen. For example:

```
This is a container db, start installation for each pdb.

Installation for PDB cdb_pdb1...
Installation for PDB cdb_pdb1 is done
Change TNS setting ...
Installation for PDB cdb_pdb2...
Installation for PDB cdb_pdb2 is done
Change TNS setting ...
Installation for PDB cdb_pdb3...
Installation for PDB cdb_pdb3 is done
Change TNS setting ...

This is a container db:
Main oast home: /home/oracle/oast/home
Pdb oast homes: /home/oracle/oast/home/cdb_pdb*
Main script: /home/oracle/oast/home/oast_main.sh

Step 1: To create database, on master node bej301301
cd /home/oracle/oast/home
./oast_main.sh -dbca y

Step 2: To compile all binaries for all pdb oast homes, on master node
bej301301
cd /home/oracle/oast/home
./oast_main.sh -kitcompile

Step 3: To build/load oast schema/data for all pdb oast homes, on master node
```

```
bej301301
cd /home/oracle/oast/home
./oast_main.sh -d y

Step 4: To execute the run, please execute the run from each pdb oast home, not
from the main oast home.
cd/home/oracle/oast/home/cdb_pdb*
./nrunoastoltpXXX.sh -n testrun
OR
follow OAST user guide for different workload run.
```

When you install OAST in the server/server mode, the OAST driver is executed on the same servers that contain the Oracle database. OAST automatically executes the driver on each node in the cluster using the driver shell script on node 1.

3.2 Installing OAST Software in Client/Server Mode

This section includes the OAST installation procedure in the client/server mode.

> **See Also:** Section 3.1, "Installing OAST Software in Server/Server Mode" for installing OAST in the server/server mode

To produce the required stress, multiple OAST databases are required. Refer to "OAST Workloads" for more information about the different types of OAST workloads.

The ability to generate stress depends on the working set size. The working set is defined as the number of blocks required by OAST to execute a particular workload. The working set size depends on the following two factors:

- Number of Warehouses

  Affects the range of values used to generate primary keys.

- Workload type

  Affects how often the primary key values change.

  During the install phase, ensure that the number of warehouse exceeds the working set size requirement for stress. The install phase sets the high water mark for the number of warehouses established. During stress testing, the number of warehouses that contribute to the working set can be adjusted down from the maximum warehouse value generated during the install phase, but cannot be adjusted above the maximum number of warehouses without reexecuting the oast install process.

Refer to "OAST Distribution" for more information about configuration on stress workloads.

To install OAST, perform the following tasks on the primary node:

1. Download the OAST archive to a suitable location (for example, `$HOME/oast`) and extract the archive as follows:

   ```
   $ su -oracle_install_username
   $ cd $HOME/oast
   $ unzip OAST_VERSION_PRODUCTION.zip
   ```

2. Enter the password.

   To obtain a valid password, contact the OCE group at certsup_ww@oracle.com.

3. Enter the following commands to run the OAST installation script:

```
$ cd $HOME/oast/install
$ ./oastinstall 2>&1 | tee oastinstall.txt
```

> **Note:** Use the `tee` command to redirect the standard output and error into a file. Use the resulting file to validate the output and review the errors that occurred during the installation.

4. Enter the OAST home directory. OAST uses this location to install the generated scripts and workload. The default location is displayed within brackets and is used if no value is provided. If you install multiple databases, then select a unique directory. This directory is also used as the OAST home directory for the client and will be created on the client as required. The client user name entered in step 8 requires permission to create the directory on the client if necessary.

   > **Note:** If the environment variable `OAST_HOME` is not set and the directory you entered does not exist, then OAST creates the required directory. OAST overwrites the existing OAST home directory.

5. Client/Server environment?

   Enter **1** for yes.

   Enter **2** for no.

   > **Note:** If you enter **1**, then the OAST driver is installed on the current node of the client.

6. Enter the Oracle home directory for Oracle Database Client. This directory must exist on the client and should be a valid Oracle home directory on the client.
7. Enter the host name for the client.
8. Enter the Oracle home owner of the client.

   OAST will attempt to connect to the host name and user name provided in steps 7 and 8 respectively.

9. Enter the Oracle home directory for Oracle Database. This directory must exist.

   > **Note:** If you are using different Oracle homes for Oracle Automatic Storage Management and Oracle Database, then enter the Oracle home directory for Oracle Database. OAST only validates the directory that exists.

10. Enter the Oracle base directory. This directory must exist. The Oracle base directory contains the `diag` and `cfgtoollogs` subdirectories.
11. Enter the OAST database name. The default database name is `oastdb`. If you are installing multiple workloads, then enter a unique database name.
12. Would you like to create OAST database as a Container database (CDB)?

    Enter 1 to enable.

Enter 2 to disable.

If you enable CDB, then you must enter values for the following questions:

How many pdbs do you want to create in this Container db? [1-252] :

Please enter data size for pdb cdb_pdb1 [10 Gigabytes] :

(Enter the number of PDBs if you choose to enable CDB, then enter the data size for each PDB).

13. Would you like to enable the fast recovery area?

    Enter **1** to enable.

    Enter **2** to disable.

    If you enable fast recovery, then you must enter values for the `db_recovery_file_dest` and `db_recovery_file_dest_size` parameters for the following questions:

    - Enter the Flash Recovery Destination: The value entered depends on the storage solution used. No default value is provided.
    - Oracle Automatic Storage Management: Enter the disk group with a + symbol. For example, `+FRA`.
    - File system: Enter the mount, the Cluster File System, or the local directory location.
    - Enter the Flash Recovery Size in MB: Enter only a numeric value to indicate the size in megabytes.

    If you enable the flash recovery area, then OAST uses the location entered for `db_recovery_file_dest` parameter as the location for the `log_archive_dest` parameter without prompting. However, if you disable the flash recovery area, then OAST will prompt you for the `log_archive_dest` location. This destination directory must exist. OAST does not create or validate this directory.

14. Enable database archive mode?

    Enter **1** to enable.

    Enter **2** to disable.

15. Select Oracle Database options:

    Enter **1** for Cluster (Oracle RAC).

    Enter **2** for Non-Clustered (single instance).

    If you enter **1** (Oracle RAC), then ensure that Oracle Grid Infrastructure is running. OAST will query the nodes that are currently configured using the `olsnodes` binary and will automatically validate the connection to each node. If the grid stack is offline, then OAST will be unable to automate this process and will prompt for the node names.

    For Oracle RAC, OAST will create the `oast_home/oast_cluster` directory and several supporting subdirectories on all nodes in the cluster.

16. CPU numbers and Free System Memory (in megabytes) are automatically generated based on the amount of operating system memory on the local server.

    Total System Memory is 14651 MB.

17. Enter the percentage of total memory. Specify the memory that you want to allocate to the OAST database. The default value is 33 percent. Enter a number between 10 and 90, do not enter the % symbol.

Note: For multiple databases on the same host, ensure that you have enough shared memory mount area (`/dev/shm`). The percentage of total memory used for database `orcl` (10-90) is [33 %].

18. Target OLTP data size (gigabytes) is the OLTP database size that Oracle generates. The default database size is 10 GB of seed data. This value represents the amount of OLTP metadata the loader phase will create and does not include free space.

19. Database files storage type: file system or Oracle Automatic Storage Management

    Enter **1** for file system (that include NFS, dNFS, Oracle Cluster File System, local file system for single instance).

    Enter **2** for Oracle Automatic Storage Management (Oracle ASM).

    Option 1: file system

    If you select file system, then the following questions are applicable and you must provide the file system location for data files, the number of redo log groups, the number of redo log members, and the file system location for each redo log member.

    - Enter the (Cluster) file system mount point for data files: For example, `/data/oastdb`.
    - Enter the number of online redo log groups? [Default is 2 groups, max 255]: Number of redo log groups for each instance of the database. OAST uses a redo log size of 500 MB.
    - The number of online redo log members in each group? [Default is 1 member, max 5]: If you enable multiple members, a value greater than 1, then you must enter a file system location for each member. Multiple members allow for multiplexing of redo logs and must be located on separate devices. Using the same device name for each member is not recommended.
    - Specify the file system location for log group member 1: For example: `/data/log1`.

    Option 2: Oracle Automatic Storage Management

    If you select Oracle Automatic Storage Management (Oracle ASM), then enter the Automatic Storage Management disk group for data files and redo log members. Do not include the + sign when entering the disk group. For example, enter `DATA` if the disk group is `+DATA`.

    - Specify the Oracle Automatic Storage Management disk group name for data (without the leading + symbol).
    - Enter number of online redo log groups? [Default is 2 groups and there can be a maximum of 255 groups].
    - The number of online redo log members in each group? [Default is 1 member and there can be a maximum of 5 members].
    - If you enable multiple members, a value greater than 1, you must enter a disk group location for each member. Multiple members allow for multiplexing of redo logs and must be located on separate devices. Using the same device name for each member is not recommended.
    - Specify the Oracle Automatic Storage Management disk group name for log group member 1 (without the leading + symbol).

20. Select **Return** to confirm/continue or **e** to exit.

    OAST will print database sizing information for the current installation execution, including the number of warehouses and redo log details.

    To continue the installation, press the **Enter** key.

21. Verify the correctness of the following info:

    - Continue
    - Modify
    - Exit

    OAST prints information for the current installation execution including total data file size, redo size for all nodes, and other operating system parameter information.

    OAST generates the driver kit and installs the necessary shell and support files across all the nodes in the configuration.

    Client files are copied to the *oast_home*/`oast_clients` location.

    OAST also executes the `elevator setup` script and copies the required files to all nodes in the cluster.

    If CDB was enabled during installation, then OAST will perform a silent installation for each PDB OAST home and prints the guideline on the screen. For example:

    ```
    This is a container db:
    Main oast home: /home/oracle/oast6301/beta2/cs_si
    Pdb oast homes: /home/oracle/oast6301/beta2/cs_si/cdb_pdb*
    Main script: /home/oracle/oast6301/beta2/cs_si/oast_main.sh

    Step 1: To create database, on master node bej301301
    cd /home/oracle/oast6301/beta2/cs_si
    ./oast_main.sh -dbca y

    Step 2: To compile all binaries for all pdb oast homes, on master node
    bej301301
    cd /home/oracle/oast6301/beta2/cs_si
    ./oast_main.sh -kitcompile

    Step 3: To build/load oast schema/data for all pdb oast homes, on master node
    bej301301
    cd /home/oracle/oast6301/beta2/cs_si
    ./oast_main.sh -d y

    It is a client/server mode.
    Step 4: To compile from the client bej301017:
    cd /home/oracle/oast6301/beta2/cs_si/cdb_pdb*/oast_clients
    ./nrunoastoltpXXX.sh -kitcompile

    Step 5: To execute the run from the client bej301017
    cd /home/oracle/oast6301/beta2/cs_si/cdb_pdb*/oast_clients
    ./nrunoastoltpXXX.sh -n testrun
    OR
    follow OAST user guide for different workload run.
    ```

3.3 Installing OAST Software in Silent Mode

OAST supports silent installation. Use the response file template oastinstall.rsp in the install/stage/responsefile directory to perform a silent installation. You can customize the values for each variable in the response file according to your requirements.

To perform a silent installation, navigate to the OAST home directory and run the following commands:

```
$ cd $HOME/oast/install
$ ./oastinstall -responsefile stage/responsefile/oastinstall.rsp 2>&1 |tee silent_
install.out
```

Table 1–1 describes the environment variables used with OAST. The oastinstall.rsp file in the install/stage/responsefile directory contains the description and examples for these variables.

4 Postinstallation Tasks

After installing OAST, you must validate the installation. The following sections describe the procedure to validate the OAST installation:

- Section 4.1, "Validating the Installation"
- Section 4.2, "Compiling the Loader and the Driver Binaries"
- Section 4.3, "Validating the Loader and Driver Binaries"

4.1 Validating the Installation

Use the `tee` command to validate the logs generated as part of the redirection. Review the output for any command errors.

For an Oracle RAC installation, all the nodes in the cluster must contain the `$OAST_HOME/oast_cluster` directory. This directory must contain multiple files and directories.

You must also review the following files and subdirectories in the `OAST_HOME` directory on the primary node. However, this is not a complete list of the generated files and subdirectories.

- `oast_main.sh` (CDB-enabled only)

  Contains the main script if multitenant container database (CDB) is enabled. Refer to Appendix B, "CDB and PDB Utilities" for more information.

- `$CDBNAME_pdb*` directory

  Contains the PDB OAST home if CDB is enabled. Each directory is a valid OAST home. Refer to Appendix B, "CDB and PDB Utilities" for more information.

- `nrunoastoltp`*`num_ware`*`.sh`

  Contains the main drive script, the value for *`num_ware`* depends on the size of the seed data entered in step 18.

- `oastoltp`*`num_ware`* directory

  Contains the scripts to generate tablespaces and tables, and start the loader program.

- `oastoltp`*`num_ware`*`/log`

  Contains the output of the `-d y` option used with the `nrunoastoltp`*`num_ware`*`.sh` script to execute the schema creation and load.

- `dbca_template`

Contains the files required to run DBCA in silent mode. OAST uses these files to build the final DBCA template, which is a part of the installation process.

- oast_cluster (Oracle RAC installation only)

  Contains the directory with multiple subdirectories, supporting scripts, and source code. There are several primary files under this directory structure as follows:

  – oast_cluster/oastoltp*num_ware*

    Contains the rac_elevator.ksh driver script and supporting files. Refer to Section 6.3, "Running OAST Oracle RAC Elevator Test" for more details.

  – oastoltp*num_ware*/oastenv.sh

    Contains various export variables.

  – oastoltp*num_ware*/tnsnames.ora

    Contains the tnsnames.ora file that is used by the drivers. OAST uses the TNS_ADMIN environment variable to control the location of the tnsnames.ora file.

  – oastoltp*num_ware*/stressrun/scripts

    Contains the scripts that support the execution of the driver. During the execution, the output is temporarily stored in this directory, including the tps.out script. After the execution, the logs are copied to the oastoltp*num_ware*/stressrun/results/*testname* directory.

  – oastoltp*num_ware*/stressrun/source/server

    Contains the Oracle Call Interface (OCI) code for the loader and driver. The code in this directory is compiled with the -kitcompile command to generate the driver and the loader binaries.

  – oastoltp*num_ware*/stressrun/bin

    Contains the following loader and driver binaries that are generated after a successful -kitcompile command:

    ```
    oastoltpload.exe
    runoastoltpb.exe
    ```

    Note: Binaries generated on all the platforms will have the .exe extension and it is the expected behavior. In an Oracle RAC configuration, the binaries are compiled once on the primary node using the -kitcompile command and distributed as part of this command to all the nodes in the cluster at the following location:

    $OAST_HOME/oastoltp*num_ware*/stressrun/bin.

  – oastoltp*num_ware*/stressrun/results

    Contains the final results of the execution and completion of a stress run in creating a *testname* subdirectory. For example, execution and completion of the following command:

    ```
    nrunoastoltp1500.sh -n iorun -u 100 -w 1 -t 600
    ```

    where, 1500 is *num_ware* and oast_home is /home/oracle/oast/home will create the results subdirectory as follows:

    ```
    /home/oracle/oast/home/oast_cluster/oastoltp1500/stressrun/results/iorun
    ```

4.2 Compiling the Loader and the Driver Binaries

To compile the loader and the driver binaries, namely, the `oastoltpload.exe` and `runoastoltpb.exe` , you must pass the `-kitcompile` option to the `nrunoastoltp`*`num_ware`*`.sh` script. This script will attempt to locate the compiler and generate files for the current platform. However, if a location cannot be determined, then the script will prompt for the path as necessary. Use the following commands:

```
cd $OAST_HOME
./nrunoastoltp1500.sh -kitcompile 2>&1 | tee kcompile.txt
```

If CDB is enabled, then use the following commands to compile all the binaries for all the PDB OAST homes:

```
cd $OAST_HOME
./oast_main.sh -kitcompile 2>&1 | tee compile_all.out
```

The `oastoltpload.exe` binary is responsible for loading the initial seed metadata for the various tables under the `oastoltp` user name.

The `runoastoltpb.exe` binary is a stress-harnessed OCI-multithread client. This binary supports all workload types, except `crstress`, which are workload 6 and workload 7. Workload 6 and workload 7 do not use the `runoastoltpb.exe` for stress generation, but uses a set of supporting shell scripts and SQL commands to drive interconnect traffic in an Oracle RAC environment.

4.3 Validating the Loader and Driver Binaries

Review the redirected file for warnings or errors issued by the compiler or linker phases. The `-kitcompile` option generates the following two binaries in the `oastoltp`*`num_ware`*`/stressrun/bin` directory:

- `oastoltpload.exe`
- `runoastoltpb.exe`

When you install OAST for an Oracle RAC configuration, the install pushes the binaries to each node. You must verify the binaries that are located in the `oastoltp`*`num_ware`*`/stressrun/bin` directory.

Note: Binaries generated on all platforms will have the `.exe` extension. This is the expected behavior.

OAST generates a transfer message for each node in the cluster. For example:

```
Transferring binaries to system host01:
/scratch/hporcl/oast/home/oast_cluster/oastoltp1500/stressrun/bin ..
```

You must review the output on the screen and in the redirected output file for these messages.

5

Creating a Database

This chapter contains information about creating a database and the OAST schema. It includes the following topics:

- Section 5.1, "Creating a Database"
- Section 5.2, "OAST Schema Creation and Data Load"

5.1 Creating a Database

Although database creation and the `oastoltp` schema creation and data loading can be executed sequentially by combining the `-dbca y -d y|c` options that control these steps, Oracle recommends issuing each command separately and validating each step. This approach saves time and allows for easier isolation of the issue because OAST does not validate creation of the database before executing the schema creation. The schema creation and data loading is a complex step and produces multiple pages of output.

Executing the `-dbca y` option of the `nrunoastoltp`*`num_ware`*`.sh` script deletes an existing database by the same name and then creates the database. OAST uses Oracle Database Configuration Assistant (DBCA) in silent mode to control the deletion and creation steps. Perform the following steps to create a database:

For a non-CDB

1. Navigate to the OAST home directory:

```
cd $OAST_HOME
```

2. Run the following script:

```
./nrunoastoltpnum_ware.sh -dbca y 2>&1 | tee dbcreate.txt
```

For a CDB

1. Navigate to the OAST home directory:

```
cd $OAST_HOME
```

2. Run the following script:

```
./oast_main.sh -dbca y 2>&1 | tee dbca_all.out
```

If you are using Oracle Automatic Storage Management (Oracle ASM), then DBCA prompts for the `ASMSNMP` password. This is the same password that you entered as part of the Oracle Automatic Storage Management creation step during an Oracle Grid Infrastructure installation.

For Oracle RAC, the database name is the value entered in step 11 in Section 3.2, "Installing OAST Software in Client/Server Mode."

Note:

If you do not remember the ASMSNMP password, then you must reset the password. Refer to Section 9.5.2, "How to Reset ASMSNMP Password" for more information.

5.1.1 Validating Database Creation

Validate the database creation by reviewing the generated output. If an error is found, review the Oracle Database Configuration Assistant log files in the following directories:

- $ORACLE_BASE/cfgtoollogs/dbca
- $ORACLE_BASE/cfgtoollogs/dbca/*db_name*

where *oracle_base* and *db_name* are values entered during the installation phase. If Oracle Database Configuration Assistant fails to create the database, then you must review the alert and trace logs in the following directory:

```
$ORACLE_BASE/diag/rdbms/db_name/instance_name/trace
```

Common Issues

The common issues during validating database creation include:

```
Not enough memory defined as shared memory for the SGA.
System configuration settings and undefined or invalid database file locations.
```

When a failure occurs, understanding and resolving the issue is critical. Reexecution of the creation phase works very rarely without additional modifications to the system. If the failure is related or traced back to invalid input entered during the installation phase, then you must start the installation at the oastinstall phase. Otherwise, reexecuting the database creation will be necessary.

5.2 OAST Schema Creation and Data Load

OAST schema creation and data load are initiated by using the -d y option with the nrunoastoltp*num_ware*.sh script. This phase will also execute Oracle's calibrate I/O PLSQL package. This package determines I/O throughput by IOPS and MBPS for the existing file names used within the OAST database.

The freeextent.rpt and extent.rpt reports, which provide usage values are also generated at the end of this phase. Both these reports are generated in the $OAST_HOME/oastoltp*num_ware* directory. Perform the following steps to load the data:

1. Navigate to the OAST home directory:

   ```
   cd $OAST_HOME
   ```

2. Run the following script:

   ```
   ./nrunoastoltpnum_ware.sh -d y 2>&1 | tee schema.txt
   ```

Note: If CDB is enabled, then you must use the following command to load data for all PDB OAST homes:

```
cd $OAST_HOME
./oast_main.sh -d y 2>&1 | tee load_all.out
```

You will be prompted with the following options:

Would you like to use offline loading?

1 Yes

2 No

If you choose yes, then OAST will use offline load tools to generate the .csv data files and then to load the data. You must provide the data directory to store the .csv files. Otherwise, OAST will use online loading, which will not generate the .csv files.

Schema creation and data loading can take several hours to complete depending on the number of warehouses and the processor speed of the nodes. The schema is created under the database user name OASTOLTP. This phase uses the environment variables in oastenv.sh located in the $OAST_HOME/oastoltp*num_ware* directory.

The crstress workload (-w 6 an -w 7) requires to use the -d c options to generate the required schema and tables. The crstress schema creation and load does not use the oastloader for data load. Data load is managed using .sql files.

Note: If CDB is enabled, then oast_main.sh does not support to create crstress workload. You must navigate to a certain PDB OAST home to use the -d c option instead.

5.2.1 Validating OAST Schema Creation and Data Load

As with other tasks, validation starts with reviewing the redirected output. OAST terminates the step if an error occurs. OAST generates schema and loader logs in the $OAST_HOME/oastoltp*num_ware*/log directory. You must review the trace.log file for errors or issues. The trace.log file provides the start and stop entries, with a time stamp for each step in the process. All the tasks in the file must have a done entry. If OAST fails during a step, then a stop file may be generated in the $OAST_HOME/oastoltp*num_ware* directory. This stop file must be removed before you run the Schema Creation - Data Load phase. Since OAST does not clear the $OAST_HOME/ostoltp*num_ware*/log directory when you rerun the schema creation phase, it is recommended that you archive or move the directory to a different location before you rerun the schema creation phase.

Before you continue with the next phase, you must manually validate the tables within the schema by logging in as the oastoltp user.

```
cd oast_home/oastoltpnum_ware
. ./oastenv.sh
sqlplus oastoltp
Enter password: password
SQL>set pages 100
SQL>select table_name from user_tables;
SQL>select count(*) from $table_name;
```

Where table_name can refer to any of the following tables:

- WARE
- DIST
- ITEM
- NORD
- CUST
- ORDR
- HIST
- STOK
- ORDL

These tables must exist under the oastoltp user. The row count of each table depends on the number of warehouses.

The following table lists the results of a SQL query for an OAST database of 1500 warehouses after the data load phase. The database size is approximately 100 GB. The WARE table must have 1500 rows and the DIST table row count must be ten times the WARE table row count, which is 15000 in this case.

TABLE_NAME	NUM_ROWS
WARE	1500
DIST	15000
ITEM	100000
NORD	13500000
CUST	45000000
ORDR	45000000
HIST	45000000
STOK	150000000
ORDL	450000000

5.3 OAST Offline Data Load Tool

OAST includes a separated offline data load tool that can be used to generate OAST data to .csv files and to separately load the oastoltp schema. This tool is a new feature with Oracle Database release 12.1, but it can be used for both Oracle Database release 11.2 and Oracle Database release 12.1.

OAST offline data load generates data to .csv file first, then loads the data to OAST database from .csv file in bulk, which greatly improves the data load speed. Previous version of OAST uses online data load, which generates and loads data record by record.

OAST offline data load tool is located in the $OAST_HOME/oastoltp*XXX*/loader directory. You can use oastdata.sh -g y to generate data and oastdata.sh -l y to load data.

The detailed usage is as follows:

```
[ loader]$ ./oastdata.sh
```

```
Usage:
oastdata.sh [-h] [-g y/n] [-l y/n] [-w WARE] [-d DIR] [-o LOG] [-r ROW] [-R ROW]

Where:
-h          help
-g [y/n]    generic data
-l [y/n]    load data
-w [WARE]   warehouse number
-d [DIR]    data directory
-o {OUT]    log directory
-r [ROW]
-r [ROW]
```

For example,

```
oastdata.sh -g y
```

Generates the data.

```
oastdata.sh -l y
```

Loads the data.

```
oastdata.sh -g y -l y
```

Generates and loads the data.

6 Running OAST Workloads

This chapter contains information and procedures about running OAST workload, running the stress tests and destructive tests, and running Oracle RAC Elevator test. It includes the following topics:

- Section 6.1, "Running OAST Workload for a non-CDB"
- Section 6.2, "Running OAST Workload with CDB Enabled"
- Section 6.3, "Running OAST Oracle RAC Elevator Test"

6.1 Running OAST Workload for a non-CDB

To start an OAST workload for a non-CDB, perform the following steps:

1. Navigate to the OAST home directory.

   For a server/server mode, use the following command:

   ```
   cd $OAST_HOME
   ```

   For a client/server mode, use the following command:

   ```
   cd $OAST_HOME/oast_clients
   ```

2. Run the following command:

   ```
   ./nrunoastoltp1500.sh -n test1 -u 10 -t 300 -w 1 2>&1 | tee test1.out
   where:
   1500 is the number of warehouses
   test1 is the test name
   10 is the number of users
   300 is the time in seconds
   -w 1 is the I/O workload
   ```

After you run the command, the driver/workload results are copied to the following directory:

For a server/server mode:

```
$OAST_HOME/oast_cluster/oastoltpnum_ware/stressrun/results/testname
```

For a client/server mode:

```
$OAST_HOME/oast_clients/oastoltpnum_ware/stressrun/results/testname
```

where *test_name* is the value passed to the `-n` option. During the execution, OAST temporarily generates the results in the `$OAST_HOME/oast_cluster/oastoltp` *num_ware*`/stressrun/scripts` directory.

Use the `tail -f` command to view `tps.out` file during execution. OAST generates Automatic Workload Repository (AWR) report. If you execute the workload for the `oastoltp` user name, then an `invalid user name` or `invalid object` error occurs, which indicates a failure in the Schema Creation and Data Load Phase.

6.2 Running OAST Workload with CDB Enabled

Starting from Oracle Database 12*c* Release 1 (12.1.0.1), OAST supports pluggable database (PDB) and multitenant container database (CDB) features. If you enable CDB option during OAST installation, then you can run OAST workload with CDB. There are two methods to run the OAST workload with CDB:

- Section 6.2.1, "Running Single Workload"
- Section 6.2.2, "Running Multiple OAST Workloads Concurrently"

6.2.1 Running Single Workload

In this method, OAST performs workloads within the same PDB. However, different workloads must be executed separately. This is similar to earlier versions Oracle database where CDB feature was not supported. For example, to start an I/O workload, you must perform the following steps:

1. Navigate to the OAST home directory.

   ```
   cd $OAST_HOME
   ```

2. Run the following command:

   ```
   ./nrunoastoltpXXXX.sh -n test1 -u 10 -t 300 -w 1 2>&1 | tee test1.out
   where:
   XXXX is the number of warehouses
   test1 is the test name
   10 is the number of users
   300 is the time in seconds
   -w 1 is the I/O workload
   ```

6.2.2 Running Multiple OAST Workloads Concurrently

If you want to run multiple workloads, then you must create multiple PDBs for each workload. During installation, specify the PDB number and the size for each PDB. For example, if the target OAST database name is `cdb`, then during OAST installation, you must answer the following questions:

1. Would you like to create OAST db as a CDB?

   You chose: Yes

2. How many PDBs do you want to create in this container db? [1-252]:

   3

3. The PDBs are created as follows:

   ```
   PDB Name: cdb_pdb1 PDB_OAST_HOME: /home/oracle/oast/home/cdb_pdb1

   PDB Name: cdb_pdb2 PDB_OAST_HOME: /home/oracle/oast/home/cdb_pdb2

   PDB Name: cdb_pdb3 PDB_OAST_HOME: /home/oracle/oast/home/cdb_pdb3
   ```

4. Please enter data size for pdb cdb_pdb1 [10 Gigabytes]:

1

Size for pdb cdb_pdb1 is 1 GB.

Please enter data size for pdb cdb_pdb2 [10 Gigabytes]:

2

Size for pdb cdb_pdb2 is 2 GB

Please enter data size for pdb cdb_pdb3 [10 Gigabytes]:

3

Size for pdb cdb_pdb3 is 3 GB

> **Note:** During OAST installation, the following pdb oast homes are installed in the main oast home directory:
>
> - `$OAST_HOME/cdb_pdb1`
> - `$OAST_HOME/cdb_pdb2`
> - `$OAST_HOME/cdb_pdb3`

5. Create the database by running the following commands:

```
cd $OAST_HOME
./oast_main.sh -dbca y
```

> **Note:** Alternatively, you can also run the command, `nrunoastoltp`*`xxxx`*`.sh -dbca y` from each pdb oast home.

6. Compile all the binaries for each pdb oast home. Run the following commands:

```
cd $OAST_HOME
./oast_main.sh -kitcompile
```

> **Note:** Alternatively, you can also run the command, `nrunoastoltp`*`xxxx`*`.sh -kitcompile` from each pdb oast home.

7. Build the oast schema and load the data for each pdb oast home. Run the following commands:

```
cd $OAST_HOME
./oast_main.sh -d y
```

> **Note:** Alternatively, you can also run the command, `nrunoastoltp`*`xxxx`*`.sh -d y` from each pdb oast home.

8. To start multiple workloads, for example 3 workloads together, you must run the following commands from different terminals simultaneously:

OLTP terminal:

```
cd $OAST_HOME/cdb_pdb1
```

```
./nrunoastoltp5.sh -no test1 -w 1 -u 10 -t 900 2>&1 | tee test1.out
```

CPU terminal:

```
cd $OAST_HOME/cdb_pdb2
./nrunoastoltp10.sh -kitcompile
./nrunoastoltp10.sh -no test4 -w 4 -u 10 -t 900 2>&1 | tee test4.out
```

DSS terminal:

```
cd $OAST_HOME/cdb_pdb3
./nrunoastoltp15.sh -kitcompile
./nrunoastoltp15.sh -no test5 -w 5 -u 10 -t 900 2>&1 | tee test5.out
```

Note:

- Do not use the `-n` option. This will restart the database, use `-no` instead.
- The main script, `oast_main.sh` can only execute `-dbca`. The `-kitcompile`, `-d y`, and `no` are executed from the main oast home directory.
- The `crsstress` workload must be executed separately.

6.3 Running OAST Oracle RAC Elevator Test

To run OAST Oracle RAC Elevator test, perform the following steps:

1. If necessary, reload the OAST data using the following commands:

```
$ cd $OAST_HOME
$ nrunoastoltpXXX.sh -d y
```

2. Run the following commands to check the database by starting and stopping the Oracle RAC database:

```
$ cd $OAST_HOME/oast_cluster
$ ./multinodedb.sh node1 node2 nodeX
```

3. Ensure that the listener is correctly running:

Note:

The main driver of the Oracle RAC elevator script will use the `bequeath` protocol to connect. However the Oracle RAC elevator script on the main node will attempt to validate user connect counts by using the `tnsnames` entry `oast1`.

```
cd $OAST_HOME/oast_cluster/oastoltpXXX
. ./oastenv.sh
sqlplus oastoltp/oastoltp@oast1
```

4. Run the elevator test as follows:

```
$ cd $OAST_HOME/oast_cluster/oastoltpXXX/nxhome
$ ./rac_elevator.ksh -n rac-elevator-test -oasthome $OAST_HOME -w XXX -steps
144 -elevator_kind r -shutdown_mode bystep > elevator.out 2>&1
```

5. Check the generated output files `elevator.out` for errors.

7 Tuning OAST Workloads to Stress System Resource

This chapter contains information and procedures about tuning OAST workload, running the stress tests and destructive tests, and the requirements to run the stress tests for Oracle Database 12c. It includes the following topics:

- Section 7.1, "Running the Stress Tests for OAST Certification"
- Section 7.2, "Additional Requirements for Oracle Database 12c"

7.1 Running the Stress Tests for OAST Certification

To run the stress tests for OAST certification, perform the following steps:

1. At least one separate client machine is required and the client machine must be operating system resource capable in terms of cpu, memory, and network to drive the necessary client requests to the database cluster. Since OAST is an open workload, an additional client machine may be required to drive CPU utilization on the cluster. If the cluster database is unable to achieve a significant CPU value, which is above 90 percent, then you must monitor the CPU utilization on the client machine for high system CPU utilization. If the client machine is 100 percent utilized, then adding an additional client machine may be necessary.

   Note: The client machine may be undersized to drive the necessary workload stress on the cluster. You must specifically monitor the CPU on the client to determine if the client is CPU bound. You can copy the OAST client home from client1 to client2 if an additional client is needed, Oracle database must be installed on the second client using the same oracle home location.

2. At least three Oracle ASM diskgroups are required: Each Oracle ASM diskgroup must be mounted on physically separate spindles on the shared storage. An Oracle ASM diskgroup and its underlying disks and volumes must not share the physical disks with any other Oracle ASM diskgroup. The minimum three diskgroups are: `DATADG1`, `DATADG2`, and `REDODG1`. The first diskgroup, `DATADG1` must be used to mount three databases to drive CPU, IOPS/OLTP, and Interconnect/Network databases. The redo logs for each of these databases must reside on the third diskgroup `REDODG1`. The second diskgroup, `DATADG2` must be used to mount the DSS datafiles and its redo logs. Place the redo logs for the DSS on `DATADG2`. The separation of OLTP (IOPS) and DSS (MBPS) on physical spindles will help achieve a controlled and consistent IOPS and MBPS flow by reducing physical disk arm movement for these distinctively different I/O workloads.

3. If the operating system supports `ASMLIB`, then it must be correctly configured to use the multipath name and exclude single port device names.

4. Storage validation: ORION is an Oracle provided standalone binary and can be used to validate and determine the IOPS and MBPS values of the storage. The best practices for using Orion are as follows:

   a. Run IOPS against the LUNs that will be used for `DATADG1` and `REDODG1` sequentially on each node in the cluster. All IOPS results for each node must be within a close range. If the values are different, then you must determine the specific reason and reexecute.

   b. Run MBPS against the LUNs that will be used for `DATADG2` sequentially on each node in the cluster. All MBPS results for each node must be within a close range. If the values are different, then determine the specific reason and reexecute.

   c. After you run IOPS and MBPS and you have consistent values for each on all the nodes, you must run IOPS against the LUNs that will be used for `DATADG1` and `REDODG1` in parallel on each node in the cluster. All IOPS results for each node must be within a close range. If the values are different, then you must determine the specific reason and reexecute.

   d. Run MBPS against the LUNs that will be used for `DATADG2` in parallel on each node in the cluster. All MBPS results for each node must be within a close range. If the values are different, then you must determine the specific reason and reexecute.

   e. Individual values for IOPS must approximately be equal to the sum of the corresponding IOPS that you run in parallel.

      Individual values for MBPS must approximately be equal to the sum of the corresponding MBPS that you run in parallel.

      For example if the MBPS value for a sequential run as mention is step b is 800 for each node, then the MBPS value for a parallel run as mention in step d should sum upto to 800. Each node must execute at 100MBPS.

5. Run oastinstall 4 times, once for each database and provide unique values for the `OAST_HOME` location and `Database_name` for each execution or installation of oastinstall.

6. Redo logs: Use a minimum of 6 redo log groups. The default for OAST is 2, so you must enter a value of 6 during oastinstall. IOPS workload can generate a significant amount of redo logs, if the storage array is very efficient in IOPS and latency. Monitor the `alert.log` file for `checkpoint not complete` messages, which indicate that the logs are switching too fast to complete the checkpoint. You may need to increase the size of the redo logs.

7. If you use multiplexing, ensure that each thread resides on a different REDO diskgroup. Multiplexing writes the same redo into multiple location. Using the same redo diskgroup for each thread will only cause contention.

8. Install in the client/server mode.

9. The client operating system and the servers operating system must be the same.

10. Oracle Database version on the client and the servers must be the same.

11. Client install user name must match the servers.

12. The client must have the ability to use password-less ssh to all nodes in the cluster.

13. Each node must have the ability to use password-less ssh to the client. This means that each node must have the ability to ssh back to all client machines without a password.
14. Execute each workload independently to verify resource and stress utilization. Validate if each workload is running as expected and independent of another workload. This procedure is critical in validating each workload in standalone before executing the workload in parallel.
15. When running test concurrently, disable the OS Watcher on all stressruns except one. `-osw n` option for `nrunoastoltp`*`XXX`*`.sh`.
16. The client's clock must have the same time zone, date, and time as the servers.

7.1.1 Four Stress Workloads

Four databases are typically necessary to drive the resource utilization and increase load. Resource utilization in this document is defined as CPU, Memory, Network interconnect traffic, and I/O to the shared storage subsystem. For I/O, both large random I/O in terms of MBPS and small random I/O in terms of IOPS are targeted.

Each workload is a separate oastinstall scenario with interconnect requiring additional attention during the database and seed load steps. Refer to *New Workload for Interconnect Traffic* section in the `readme.oast` file for specific details.

Verify each workload independently. Execute each workload in the standalone mode to verify if the targeted operating system resource is fully utilized. Once each resource is verified, execute all the four workloads concurrently. When running the test concurrently, disable the OS Watcher (as the default mode is enabled) on all the stress runs except one. Use the `-osw n` option with `nrunoastoltp`*`XXX`*`.sh` script to disable.

The four workloads required are as follows:

> **Note:**
>
> Modification of the source code must occur in the correct location and you must recompile after the initial modification by using the `-kitcompile` command.
>
> The `SCALE` value is defined in the `oastoltpvars.sh` file located on the client at: *`OAST_HOME`*`/oast_clients/oastoltp`*`XXX`*`/stressrun/scripts`. Updating the `SCALE` value does not require a recompile of the code.

OLTP:CPU:Database 1

- Workload 4 (-w 4)
- Datafiles: `DATADG1`
- Redo logs: `REDODG1`
- Database Name: `oastcpu`
- User count: Depends

This workload will drive CPU utilization on all nodes in the cluster by using a read-only transaction set that is fully cached in the SGA. The buffer cache must be large enough to hold the working set. The buffer cache working set (the number of buffers used by the workload) is controlled by an export value called `SCALE`. Specific instructions on setting the `SCALE` value is defined in step 1.

Executing CPU workload requires the following modifications:

1. Modify the SCALE export value in oastoltpvars.sh script on the client, by setting it to 5 * number of nodes. You must note that the oastoltpvars.sh is modified, as each workload will have this file. This will initially be equal to the number of warehouses. Reduce the value to 5 * number of nodes. You must have at least SCALE value for warehouses, otherwise the driver will generate an error.

   For example if the total warehouse value is 15, with 8 nodes, setting the SCALE to a value above 15 will cause an error. The SCALE value is used to limit the number of buffers consumed by OAST by reducing the warehouse ID primary key range. A larger warehouse range and a higher SCALE value will result in more rows being read from the disk.

2. OAST code changes are currently required before you run workload 4 as the read only transactions workload. Contact certsup_ww@oracle.com for more details. A user will be able to specify the transaction read and write ratios as an input parameter value in the next OAST patch set release.

Multiple Client machines

Multiple client machines could be required to support the necessary workload (user threads -u option) to drive CPU on the cluster. The number of client machines depends on the individual server performance and specifications, the number of nodes, and the client's machine specification. Monitor the client machine for CPU utilization at incremental increases in the users threads, use the -u option with nrunoastolpt*XXX*.sh script. If client CPU is over 90 percent, then additional user threads or workers (increasing the -u value) on the same client will have a negative impact and cause a reduction in resource utilization on the cluster. At this point an additional client machine will be necessary to drive additional user threads and achieve a high percentage of CPU on the server.

For example:

- Vendor cluster: 8 nodes
- CPU: 2 clients running each with 110 users. Total users would be 220 from both the clients. User count would be 220/8 users per node

OLTP: IOPS

- Workload 1 (-w 1)
- Datafiles: DATADG1
- Redo logs: REDODG1
- Database Name: oastiop
- User count: 25 * number of nodes

Do not modify the SCALE value initially. The OAST driver will divide and partition the warehouses based on the number of nodes in the cluster. Since this is an IOPS workload, you must have sufficient warehouses to cause a miss for the current primary key. Adjusting the SCALE value reduces the read and write ratio. If all the blocks fit into buffer cache, then the workload will be more write intensive than read. As the LGWR and DBWR will be active, no additional reads are required after the cache is primed.

For example:

- Vendor cluster: 8 nodes
- Total users 200 per node. User count would be 200/8, which is equal to 25 users per node

CRS:Interconnect

- Workload 7 (-w 7)
- Datafiles: DATADG1
- Redo logs: REDODG1
- Database Name: oastcs
- User count: 16

Do not modify the SCALE value. The load option for this workload to build the required tables and seed data is -d c. OAST automatically generates a workload that is capable of handling the number of nodes running in the cluster and no additional steps are required. This workload causes multiblock transfers using full table scan, from and to all nodes in the cluster. Using more than 16 users, the -u option is not recommended. If you do not see high interconnect traffic during the run, then the buffer cache is probably too small. The buffer cache size must be large enough to handle both sending blocks and receiving blocks. If the buffer cache is not large enough, then the workload will be forced to disk and no interconnect traffic will be generated. The -blocks parameter can be used to increase or decrease the number of blocks. The value provided is the total sum of blocks that will be used for sending and receiving data divided by 2. Fifty percent of this value is allocated for sending the blocks and the remaining fifty percent is allocated for receiving the blocks.

For example:

- Vendor cluster: 8 nodes
- Total users 16, users per node is 16, multiblock interconnect traffic.

DSS:MBPS

- Workload 5 (-w 5)
- Datafiles: DATADG2
- Redo logs: REDODG2
- Database Name: oasdss
- User count: 3 * number of nodes

Set the following init.ora parameters:

1. Assign the DOP for the tables that queried for DSS.

   ```
   alter table ordl parallel 4;
   alter table stok parallel 4;
   alter table cust parallel 4;
   alter table hist parallel 4;
   alter table ordr parallel 4;
   ```

2. Set the following common parameters:

   ```
   alter system set parallel_adaptive_multi_user=FALSE scope=both;
   alter system set parallel_min_servers=8 scope=both;
   alter system set parallel_max_servers=64 scope=both;
   alter system set parallel_degree_policy=FALSE scope=both;
   ```

3. The following settings depend on the name of the dss database. This example assumes that the db_name is set to oastdss with an admin database. If you are using a policy-managed database, then the *sid* will have an underscore and the number of nodes will be equal to 8.

Add or subtract depending on the number of nodes in the cluster. Modify the *`sid`* value to the corresponding instance names on your system.

Values for the `instance_groups` and `parallel_instance_group` parameters do not need to be modified, the following values will work on all systems.

Note: For an Oracle RAC non-cross node PQ, add or subtract based on the number of nodes. Isolate PQ to individual nodes.

The following example is for an 8 node cluster:

```
alter system set instance_groups='g1','all' scope=spfile sid='oastdss1';
alter system set instance_groups='g2','all' scope=spfile sid='oastdss2';
alter system set instance_groups='g3','all' scope=spfile sid='oastdss3';
alter system set instance_groups='g4 ,'all' scope=spfile sid='oastdss4';
alter system set instance_groups='g5','all' scope=spfile sid='oastdss5';
alter system set instance_groups='g6','all' scope=spfile sid='oastdss6';
alter system set instance_groups='g7','all' scope=spfile sid='oastdss7';
alter system set instance_groups='g8','all' scope=spfile sid='oastdss8';

alter system set parallel_instance_group='g1' scope=spfile sid='oastdss1';
alter system set parallel_instance_group='g2' scope=spfile sid='oastdss2';
alter system set parallel_instance_group='g3' scope=spfile sid='oastdss3';
alter system set parallel_instance_group='g4' scope=spfile sid='oastdss4';
alter system set parallel_instance_group='g5' scope=spfile sid='oastdss5';
alter system set parallel_instance_group='g6' scope=spfile sid='oastdss6';
alter system set parallel_instance_group='g7' scope=spfile sid='oastdss7';
alter system set parallel_instance_group='g8' scope=spfile sid='oastdss8';
```

Do not modify the `SCALE` value. The OAST driver executes full table scan queries. Due to the fundamental characteristic differences when executing IOPS or MBPS against a physical disk, the requirement is to isolate each workload on separate physical spindles (having separate Oracle ASM diskgroups) on the shared storage to prevent thrashing of the physical disk arm. There have been instances where the use of the same physical disk to handle both larger random sequential reads (MBPS) and small random I/O (OLTP) causes a large queuing of read and write in the servers due to thrashing of the physical disk arm.

This queuing can have a negative impact on IOPS and cause a major bottleneck for the other workloads being executed concurrently.

For example:

- Vendor cluster: 8 nodes
- Total user count 24
- Per node user count at 3
- PQ on: parallel minimum and maximum servers at values indicated earlier DOP of 4 for the tables listed earlier.

Additional Comments:

Note: The specific `SCALE` value and number of users (`-u`) depends on multiple factors on the server and bottlenecks that are introduced.

The physical I/O ratios are not accurate as a large buffer cache causes more logical reads. Very big buffer caches have a high ratio of physical write to physical read in the AWR reports. Due to this, the physical writes must be present on the disk. However, the physical reads can be either logical, which is from the memory or physical, which is from the disk.

7.2 Additional Requirements for Oracle Database 12*c*

This section provides the modified and additional requirements to stress test an Oracle RAC Database environment for 12*c* Release 1. The following information denotes changes in reference to Section 7.1, "Running the Stress Tests for OAST Certification"

Note: Refer to Section 7.1, "Running the Stress Tests for OAST Certification" and Section 7.1.1, "Four Stress Workloads" for generic requirements to perform the stress tests.

- A minimum of 4 disk groups are required. In Oracle Database 12*c*, an additional disk group is required for Oracle ACFS. The remaining 3 diskgroups are as specified in Section 7.1, "Running the Stress Tests for OAST Certification" The Oracle homes must be installed in Oracle ACFS that is contained in the disk group `ACFSDG`. All the disk groups must be formed from separate spindles and not from luns which span the same disks.
- Oracle Cluster Registry (OCR) and voting disk are also required to be in Oracle ASM with normal redundancy. OCR must have at least two failure groups and voting disk must have at least three failure groups. Oracle recommends to have OCR and voting disk in a fifth disk group, however, this is not mandatory.
- The OAST install process is executed four times to create three multitenant container databases (CDBs) and one non-CDB. The first three CDBs contain one PDB each. The `OLTP:CPU CDB` is created with 4k block size and the `DSS:MBPS CDB` is created with 16k block size. Perform the following steps to change the blocksize for the two CDBs:

  1. Run `oastinstall` to setup the home for the CDB.
  2. Edit the *`oast_home`*`/dbca_template/oastdb.dbt` file. Replace `initParam name="db_block_size" value="8" unit="KB"/` with `oastcpu` value as follows:

     ```
     initParam name="db_block_size" value="4" unit="KB"/
     ```

     or with `oastdss` value as follows:

     ```
     initParam name="db_block_size" value="16" unit="KB"/
     ```

- The non-CDB is used for the CRS:Interconnect stress test. The final configuration will be one CDB containing three PDBs and one non-CDB. The PDB from the OLTP:CPU CDB and the PDB from DSS:MBPS CDB will be unplugged from their respective CDBs and plugged into the OLTP:IOPS CDB. Refer to Chapter 8, "Configuring OAST Database and Workloads for Oracle Database 12c Certification" for details regarding the CDB and PDB setup requirement.

- The cluster must be set up with `mgmtdb`.
- The required ASM mode is Flex mode (Oracle Flex ASM). The ASM network will share the private network.
- The required cpu usage on the Oracle RAC nodes must be equal to or greater than 90 percent. Both the desired private network usage and the desired storage network usage must exceed 70 percent.

8

Configuring OAST Database and Workloads for Oracle Database 12*c* Certification

The configuration and tuning procedures described in Chapter 7, "Tuning OAST Workloads to Stress System Resource" must be modified to reflect the use of CDBs. Instead of tuning at the database level, the tuning will be at the PDB level. You must connect to the PDB before making for instance the changes using SQL/Plus listed in the configuration for DSS:MBPS. Apply the steps outlined in Section 7.1, "Running the Stress Tests for OAST Certification" to the PDBs in the 3 CDBs. No change to the CRS:Interconnect configuration as this will be done with the remaining non-CDB.

After achieving the goal of tuning the four configurations and achieving a consistent 90 percent plus cpu usage with all four databases running, the three PDBs will be combined into a single CDB configuration. Save the results of this configuration to use it as a base.

Unplug the pdb 1 `oastcpu_pdb` from the OLTP:CPU CDB database, which is `oastcpu` and the pdb `oastdss_pdb` from the DSS:MBPS CDB database, which is `oastdss` and plug them into the OLTP:IOPS CDB database, which is `oastiop`. Apply all the configurations to each PDB as done previously for the separate CDBs. Add the 4k and 16k caches, set the SGA size to the size of the 3 previous CDBs combined. Set parallel query setting that were set in the DSS:MBPS CDB, the `oastdss` database. The present main methodology for tuning the PDBs together is to use the Oracle Resource Manager. Use the resource manager to tune the combined PDBs toward the base numbers previously obtained in the separate CDBs. Following is an example to move from the CDB containing a single PDB configuration to the CDB containing three PDBs:

1. Unplug the PDBs from their CDBs:

```
$ sqlplus / as sysdba (oracle sid is oastdss1)
SQL> alter pluggable database oastdss_pdb close immediate instances=all;
SQL> alter pluggable database oastdss_pdb unplug into
'/home/oracle/homedss/oastdss_pdb.xml';
SQL> drop pluggable database oastdss_pdb keep datafiles;
SQL> quit
$ sqlplus / as sysdba (oracle sid is oastcpu1)
SQL> alter pluggable database oastcpu_pdb close immediate instances=all;
SQL> alter pluggable database oastcpu_pdb unplug into
'/home/oracle/homedss/oastcpu_pdb.xml';
SQL> drop pluggable database oastcpu_pdb keep datafiles;
SQL> quit
$ srvctl stop database -d oastdss
$ srvctl stop database -d oastcpu
```

2. Modify the `SPFILE` for the `OLTP:IOPS CDB` (`oastiop`) to accept or handle the new PDBs that will be plugged in. Run the following commands:

   ```
   SQL> alter system set db_4k_cache_size=6560M sid='*' scope=spfile
   6560M was the db_cache_size that was found in the AWR report to be the maximum
   allocated to any node during the base run for oastcpu.
   SQL> alter system set db_16k_cache_size=6880M sid='*' scope=spfile
   6880M was the db_cache_size that was found in the AWR report to be the maximum
   allocated to any node during the base run for oastdss
   SQL> alter system set db_cache_size=6496M sid='*' scope=spfile
   6496M was the db_cache_size that was found in the AWR report to be the maximum
   allocated to any node during the base run for oastiop.
   SQL> alter system set memory_target=43488M sid='*' scope=spfile;
   SQL> alter system set memory_max_target=43488M sid='*' scope=spfile
   SQL> alter system set sga_max_size=43488M sid='*' scope=spfile;
   Set memory_target=memory_max_target=sga_max_size =  max(memory_target(oastdss))
   + max(memory_target(oastcpu) + max(memory_target(oastdss)) where max is the
   maximum allocated to any node during the base run as found in AWR report.
   SQL> alter system set sga_target=26240M sid='*' scope=spfile;
   Set sga_target =  max(sga_target(oastdss)) + max(sga_target(oastcpu) + max(sga_
   target(oastdss)) where max is the maximum allocated to any node during the base
   run as found in AWR report.
   SQL> alter system set parallel_adaptive_multi_user=false sid='*' scope=spfile;
   SQL> alter system set parallel_max_servers=64  sid='*' scope=spfile;
   SQL> alter system set parallel_min_servers=8  sid='*' scope=spfile;
   SQL> alter system set instance_groups='g1','all' scope=spfile sid='oastiop1';
   SQL> alter system set instance_groups='g2','all' scope=spfile sid='oastiop2';
   SQL> alter system set instance_groups='g3','all' scope=spfile sid='oastiop2';
   SQL> alter system set instance_groups='g4','all' scope=spfile sid='oastiop4';
   SQL> alter system set parallel_instance_group='g1' scope=spfile sid='oastiop1';
   SQL> alter system set parallel_instance_group='g2' scope=spfile sid=’oastiop2';
   SQL> alter system set parallel_instance_group='g3' scope=spfile sid='oastiop3';
   SQL> alter system set parallel_instance_group='g4' scope=spfile sid='oastiop4';
   ```

   Applying the `oastdss` CDB parallel server settings to `oastiop`.

3. Restart the `oastiop` database to apply the new `spfile`:

   ```
   $ srvctl stop database -d oastiop
   $ srvctl start database -d oastiop
   ```

4. Plug PDBs `oastdss_pdb` and `oastcpu_pdb` into the `oastiop` CDB:

   ```
   $ sqlplus / as sysdba /* ORACLE_SID=oastiop1 */
   SQL> create pluggable database oastdss_pdb using '/home/oracle/homedss/oastdss_
   pdb.xml' nocopy tempfile reuse;
   SQL> create pluggable database oastcpu_pdb using '/home/oracle/homecpu/oastcpu_
   pdb.xml' nocopy tempfile reuse;
   ```

5. Run the CDB and non-CDB together using default resource management for the PDBs in the CDB:

   ```
   $ sqlplus / as sysdba /* ORACLE_SID=oastiop1 */
   SQL> alter pluggable database all open instances = all; /* always make sure
   that all pdbs are up on all instances */
   SQL> quit
   $ ./nrunoastoltp1500.sh -no new_pdb1 -u 100 -t 3600 -w 1 2>&1 | tee new_
   pdb1.out
   $ ./nrunoastoltp150.sh -no new_pdb1 -u 100 -t 3600 -w 4 2>&1 | tee new_pdb1.out
   $ ./nrunoastoltp750.sh -no new_pdb1 -u 12 -t 3600 -w 5 2>&1 | tee new_pdb1.out
   $ ./nrunoastoltp150.sh -no new_pdb1 -u 16 -t 3600 -w 7 2>&1 | tee new_pdb1.out
   ```

6. Validate the results, compare the results to the base, and modify the resource manager as required:

   ```
   exec DBMS_RESOURCE_MANAGER.CREATE_PENDING_AREA();
   BEGIN
      DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN(
        plan => 'cpu_oltp_dss_plan',
        comment => 'CDB Resource plan for OAST stress minus crs');
   END;
   BEGIN
      DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN_DIRECTIVE(
        plan => 'cpu_oltp_dss_plan',
        pluggable_database => 'oastcpu_pdb',
        shares => 2,
        utilization_limit => 60,
        parallel_server_limit => 0);
   END;
   /
   BEGIN
      DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN_DIRECTIVE(
        plan => 'cpu_oltp_dss_plan',
        pluggable_database => 'oastiop_pdb',
        shares => 3,
        utilization_limit => 25,
        parallel_server_limit => 0);
    END;
   /
   BEGIN
      DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN_DIRECTIVE(
        plan => 'cpu_oltp_dss_plan',
        pluggable_database => 'oastdss_pdb',
        shares => 1,
        utilization_limit => 15,
        parallel_server_limit => 100);
    END;
   /
   EXEC DBMS_RESOURCE_MANAGER.VALIDATE_PENDING_AREA();
   EXEC DBMS_RESOURCE_MANAGER.SUBMIT_PENDING_AREA();
   alter system set resource_manager_plan='cpu_oltp_dss_plan' scope=spfile
   sid='*';
   ```

   Based on the parameters set earlier, 50 percent, which corresponds to 3 shares of the resource are allocated to the OLTP:IOPS PDB, 33.3 percent, which corresponds to 2 shares of the resource are allocated to the OLTP:CPU PDB and 16.7 percent, which corresponds to 1 share of the resource is allocated to the DSS:MBPS PDB. OLTP:CPU receives 60 percent of the cpu, OLTP:IOPS receives 25percent of the cpu, and DSS:MBPS receives 15 percent of the cpu. Finally DSS:MBPS receives 100 percent of the parallel servers.

7. Repeat steps 5 and 6. Change value for the `resource_manager_plan` parameter for each iteration to get base results.

Currently there is no method to partition the SGA usage, in particular, the buffer cache allocation for a PDB. There are 3 workloads in the PDB, the `oltp`, `cpu`, and `dss`. The cpu workload should be cached as much as possible, oltp workload must be cached less than the cpu workload and the dss workload must be cached a minimum. The cpu usage (important to give preference to the cpu workload) and parallel servers usage (important for DSS workload preference) can be managed by the Resource Manager. Otherwise the Resource Manager allocates on the gross level - general resource percentage. By using 3 different default cache sizes, the workloads are allocated their own buffer cache area, which is not shared.

9 Troubleshooting

This section provides information about troubleshooting with OAST. It includes the following topics:

- Section 9.1, "Using OAST With Client/Server Mode"
- Section 9.2, "Using OAST With Oracle Automatic Storage Management"
- Section 9.3, "Utility Scripts"
- Section 9.4, "General OAST Troubleshooting"
- Section 9.5, "OAST Troubleshooting for Oracle Database 11g Release 2"
- Section 9.6, "OAST Troubleshooting for Oracle Database 12c Release 1"

9.1 Using OAST With Client/Server Mode

To use OAST in client/server mode, you must ensure that you have performed the following preinstallation tasks:

1. Installed Oracle Database 12*c* software on the client node.
2. Configured ssh between server host and client host for user equivalence.

To enable client/server in OAST, choose **Yes** when prompted for client/server environment.

For example:

```
Client/server environment?
1) Yes
2) NO
#?1
You chose: Yes
Client ORACLE_HOME [/u01/product/12.1.0/db]:
/home/oracle/app/oracle/product/12.1.0/db
CLIENTHOME=/home/oracle/app/oracle/product/12.1.0/db
Enter the host name for client <example: rac1>: msdtrac1
Please enter the owner of your CLIENT ORACLE HOME: oracle
login : oracle
checking connection to system msdtrac1 ...
Client OS : Linux
Client Oracle Version: 32-bit
```

9.2 Using OAST With Oracle Automatic Storage Management

To use this enhancement of OAST, you must ensure that you have performed the following preinstallation tasks:

1. Installed Oracle Grid Infrastructure successfully.
2. Created Oracle Automatic Storage Management disk group and Oracle Automatic Storage Management instances are up and running on each node of the cluster (if you are using Oracle RAC).

> **See Also:** *Oracle Database Administrator's Guide* for more information about installation and Automatic Storage Management disk group creation

To enable Oracle Automatic Storage Management in OAST, choose Oracle Automatic Storage Management as the storage option and specify the Oracle Automatic Storage Management disk group name, which is created ahead of time.

For example:

```
Calculating the size of database...
Do your database files fit in a file system or ASM?
1) file system
2) ASM (Automatic Storage Management)
#? 2
You chose: ASM (Automatic Storage Management)
Please specify the ASM diskgroup name...
ASMDG
We will use the ASM diskgroup ASMDG
filesystem_loc: +ASMDG
loggroupm1_loc: +ASMDG
loggroupm2_loc: +ASMDG
```

As shown in the preceding example, use `ASMDG` as the disk group name.

9.3 Utility Scripts

The scripts discussed in this section are called through `nrunoastoltp`*`num_ware`*`.sh`. These scripts must not be used directly:

- `oastoltp`*`XXX`*`/oastenv.sh`: contains environment variables.
- `oastoltp`*`XXX`*`/addts.sh`: for dropping a tablespace and creating a tablespace.
- `oastoltp`*`XXX`*`/addfile.sh`: for adding a data file to the tablespace.
- `oast_cluster/multinodedb.sh`: for startup and shutdown of all instances.
- `oast_cluster/multinodedb_shut.sh`: for shutdown of all instances.
- `oast_cluster/oastoltp`*`num_ware`*`/oastenv.sh`: contains environment variables for one instance.
- `oast_cluster/listener.ora` and `oast_cluster/tnsnames.ora`: for TNS settings for one instance.
- `$OAST_HOME/utils/freespace`: determines the free space of Oracle files.
- `$OAST_HOME/utils/sysstat.sh`: to get a snapshot of *wait* events.

9.4 General OAST Troubleshooting

Issues that you encounter during installation, database creation, data loading, and execution of OAST workload stress are typically the result of not validating each step, as suggested in this document.

For every task, you must redirect the `standard.out` and `standard.error` to a unique file using the `tee` command.

Note: You must check the file for errors and other issues to prevent additional down-stream problems.

Many issues that occur are traced back to incorrect configuration of system resources. OAST uses Automatic Memory Management. The `MEMORY_TARGET` parameter uses 50 percent of the defined system memory for this value. Review your operating system specific installation guide to determine the correct procedure.

OAST uses a template file, `oastdb.dbt` for Oracle Database Configuration Assistant silent installation. The template file is constructed using the installation values entered during OAST installation. Various parts of the template are shipped with OAST and the final template formed depends on system resources, the specific operating system, and the type of file system being used.

Use `nrunoastoltp`*`numware`*`.sh` to create the OAST database. OAST copies the current `$OAST_HOME/dbca_template/oastdb.dbt` to the *`ORACLE_HOME`*`/assistants/dbca/templates/` directory. Do not modify the version at `assistants/dbca/templates` directory, as OAST overwrites the version.

The different phases where you may encounter problems are as follows:

- OAST Installation

  OAST Installation fails if you use an existing OAST home directory or if you provide an invalid value for the `$OAST_HOME` variable during the installation procedure.

- Oracle Database Configuration Assistant Creation

  OAST leverages DBCA in silent mode to create the database required by OAST. DBCA generates directories and logs in the *`ORACLE_BASE`*`/cfgtoollogs/dbca` directory. The DBCA logs provides additional information for the DBCA error and cause.

  Database creation using DBCA may fail due to insufficient resources, invalid operating system settings, and invalid directories. DBCA may also fail due to Oracle ASM being offline or due to improper cleanup of a previous database with the same name.

  You must validate if the database is correctly generated by DBCA before moving to the next step.

- Schema Creation and Data Load

  OAST uses SQL scripts and the binary executable, `oastoltpload.exe` to create the OAST schema and to load data.

  If the `-d y` phase suspends on the first step of the process, which is `Create User oastoltp`, then it indicates that the database is not up and running and the `-dbca y` step failed. Execution of the script to create the user name will be immediate and should not suspend. Run the following command to find the `createuser` process:

  ```
  ps -ef |grep createuser
  ```

  Run the following command to end the process:

  ```
  kill -9 $process_number
  ```

OAST generates the `trace.log` file in the *OAST_HOME*/`oastoltp`*num_ware*/`log` directory. If an error occurs during schema and data load, then the specific step that failed will be provided in the `trace.log` file.

Verify and confirm the following:

- The `-dbca y` step has completed without errors.
- The binary, `oastoltpload.exe` was generated as part of the `-kitcompile` command. The binary `oastoltpload.exe` is generated in the *OAST_HOME*/`oastoltp`*num_ware*/`stressrun/bin` directory.
- Manual deletion of the `stop` file. OAST will place a zero length file called `stop` in the *OAST_HOME*/`oastoltp`*num_ware* directory if an error occurs. You must remove this file before you rerun the `-d y` option.

9.4.1 Adding Data Files for out of space Condition

OAST uses autoextend for the data files and does not use big files. The maximum size of the data file is operating system dependent. If you reach the maximum size, then adding additional data files will remove the free space issue. Perform the following steps to add data files:

1. Navigate to the OAST home directory:

```
cd $OAST_HOME
```

2. Source the environment variables:

```
../oastoltpnum_ware/oastenv.sh
```

3. Log into SQL *Plus as follows:

```
SQL> CONNECT OASTOLTP
Enter password: password
```

4. Add an Oracle Managed File to an existing tablespace using the device or mount indicated by the `db_create_file_dest`:

```
alter tablespace $tablespace_name add datafile;
```

9.4.2 Adding Data Files to Support Multiple Devices

In some scenarios adding additional data files or mounts to the tablespaces used by OAST may be desirable to achieve a distribution of I/O. This scenario allows for adding additional mount points to the tablespaces before table creation and data load. To achieve the desired effect of I/O distribution across multiple data files, the tablespace must be locally managed and must use a uniform extend. Auto-allocated extend creation, which is the default for tablespaces, does not distribute extends but creates all extends in a single data file before moving to the next. If you add data files after the data seed load, then the existing data will not be redistributed. You must add the data files before the table is created and the seed data is loaded. Adding new mounts or data files to the tablespace before seed load will allow Oracle to use all data files for creating table extends and will produce a more balanced distribution of I/O.

This activity requires using `oastdb.sh` script directly to start specific sections of the schema creation and data load process. OAST provides the ability to supply start and stop points through the file `oastdb.sh`.

Perform the following steps to add additional data files:

1. Modify `addts.sh` to use uniform extends to create tablespace.

   > **Note:** It is recommended to use uniform extends from the start, otherwise Oracle database will not stripe across the mount points for the same tablespace.

2. Run `oastdb.sh` to create the `oastoltp` user and the tablespaces used by `oast`:

   a. Navigate to the `$OAST_HOME` directory:

   ```
   cd $OAST_HOME
   ```

   b. Source the environment variables:

   ```
   ../oastoltpnum_ware/oastenv.sh
   ```

   c. Run `oastdb.sh` as follows:

   ```
   ./oastdb.sh -oastoltp num_ware 5 6
   ```

   where, *`num_ware`* is the number of warehouse.

3. Set the mount point/directory location using the `db_create_file_dest` to identify the location:

   a. Navigate to the `oast_home` directory:

   ```
   cd oast_home
   ```

   b. Source the environment variables:

   ```
   ../oastoltpnum_ware/oastenv.sh
   ```

   c. Log into SQL *Plus as follows:

   ```
   SQL> CONNECT OASTOLTP
   Enter password: password
   ```

   d. Run the alter system set command for the `db_create_file_dest` parameter:

   ```
   alter system set db_create_file_dest=new mount location
   ```

4. Add the Oracle Managed Files data file to the existing tablespaces:

   > **Note:** OAST uses partitioned and nonpartitioned tables. The number of tablespaces created is a factor of the warehouse size. You can query the tablespaces defined using `user_tab_partitions` and `user_tables` or other system tables.

   a. Navigate to the `oast_home` directory:

   ```
   cd oast_home
   ```

   b. Source the environment variables:

   ```
   ../oastoltpnum_ware/oastenv.sh
   ```

   c. Log into SQL *Plus as follows:

   ```
   SQL> CONNECT OASTOLTP
   Enter password: password
   ```

d. Add the Oracle Managed Files to an existing tablespace using the device/mount indicated by the `db_create_file_dest` parameter:

```
alter tablespace $tablespace_name add datafile
```

5. Repeat step 2 and step 3 for each new mount point.
6. Complete the remaining steps in Schema Creation and Data Load Phase:

   a. Navigate to the `oast_home` directory:

   ```
   cd oast_home
   ```

   b. Source the environment variables:

   ```
   ../oastoltpnum_ware/oastenv.sh
   ```

   c. Run `oastdb.sh`

   ```
   ./oastdb.sh -oastoltp num_ware 7 42
   ```

   where *`num_ware`* is the number of warehouse.

   Steps 7 through 42 in the `oastdb.sh` script will complete the Schema Creation and Data Load Phase. Data loading during this process will be load balanced across the devices configured.

9.5 OAST Troubleshooting for Oracle Database 11*g* Release 2

Ensure that the user running OAST tests is a member of the `dba` group. This section contains the following topics:

- Section 9.5.1, "Problem: ORA-27125 Running Startup on Platforms That Use Huge tlbfs Memory"
- Section 9.5.2, "How to Reset ASMSNMP Password"
- Section 9.5.3, "How to Display Oracle Enterprise Manager Database Control Details"
- Section 9.5.4, "Default User Names and Passwords Created As Part of OAST with DBCA"
- Section 9.5.5, "How to Create an Automatic Storage Management Disk Group"

9.5.1 Problem: ORA-27125 Running Startup on Platforms That Use Huge tlbfs Memory

This section describes how to run the startup script on platforms that use huge `tbfs` memory:

```
SQL> STARTUP
ORA-27125: unable to create shared memory segment
Linux-x86_64 Error: 1: Operation not permitted
```

Resolution

```
$ echo 1 > /proc/sys/vm/disable_cap_mlock

SQL> STARTUP
ORA-16032: parameter LOG_ARCHIVE_DEST destination string cannot be translated
ORA-07286: sksagdi: cannot obtain device information.
Linux-x86_64 Error: 2: No such file or directory
```

Ensure Archive logoff destination is created on both nodes, for example:

```
mkdir -p /install/oce/oast/arch

SQL> STARTUP
ORACLE instance started.

Total System Global Area 1426063360 bytes
Fixed Size                  2020768 bytes
Variable Size             520096352 bytes
Database Buffers          889192448 bytes
Redo Buffers               14753792 bytes
Database mounted.
Database opened.
```

9.5.2 How to Reset ASMSNMP Password

Use the following commands to reset the `ASMSNMP` password:

```
srvctl config asm -a displays the grid infrastructure oracle home
export ORACLE_HOME= grid_oracle_home
export ORACLE_SID= ASM_instance_name for current node
asmcmd
ASMCMD> password ASMSNMP
```

9.5.3 How to Display Oracle Enterprise Manager Database Control Details

Run the following commands to display the Oracle Enterprise Manager Database Control details:

```
export ORACLE_UNQNAME=oastdb
emctl status dbconsole
The https address for login will be displayed as part of the output.
```

9.5.4 Default User Names and Passwords Created As Part of OAST with DBCA

OAST with Database Configuration Assistant (DBCA) creates the following user names and passwords:

```
User name Password Usage:
-------- -------- ------
oastoltp oastoltp OAST's OLTP schema
sys manager RDBMS Management
system manager RDBMS Management
dbsnmp manager EM Management
sysman manager EM Management
```

9.5.5 How to Create an Automatic Storage Management Disk Group

The Automatic Storage Management Configuration Assistants program, `asmca`, is located in the `$ORA_CRS_HOME/bin` directory, where `ORA_CRS_HOME` is the Oracle Grid Infrastructure home.

9.6 OAST Troubleshooting for Oracle Database 12*c* Release 1

This section provides information about OAST troubleshooting for Oracle Database 12*c* Release 1 (12.1.0.1). It includes the following topics:

- Section 9.6.1, "createuser Process Fails."

- Section 9.6.2, "Oracle Solaris Compile Error"
- Section 9.6.3, "How to Compile and Build for Separate PDB Homes"
- Section 9.6.4, "Why Does OAST Installation Not Work in EMACS"
- Section 9.6.5, "Using MEMORY_TARGET and SGA_TARGET"
- Section 9.6.6, "How to Differentiate the Resources for Different PDBs"
- Section 9.6.7, "ORA-00845: MEMORY_TARGET Not Supported on this System"
- Section 9.6.8, "How to Add and Modify the Database Parameters"

9.6.1 createuser Process Fails

OAST may stop to respond when the `creat user` process fails. To verify if the `create user` process is running, run the following command:

```
ps -ef | grep createuser
```

If the `create user` process exists, then run the following command to stop the process:

```
kill -9 $process_number
```

Note: Creating a user is the initial step of the `oastdb.sh` script.

After you stop the process, check the error in the log file located in the `$OAST_HOME/oastoltp`*xxxx* directory. If you have enabled CDB, then the log file may contain the following error, which indicates that the PDB you are connecting to is not running:

```
ERROR:
ORA-01033: ORACLE initialization or shutdown in progress
Process ID: 0
Session ID: 0
Serial Number: 0
```

The workaround for this issue is to start the corresponding PDB for the current OAST workload. For more information on starting or closing PDBs, refer to Appendix B, "CDB and PDB Utilities."

The log file may also contain the following error:

```
ERROR:
ORA-12514: TNS:listener does not currently know of service requested in connect
descriptor.
```

This means that the information in the `tnsnames.ora` file is incorrect. The `oast1` entry for the PDB does not exist. Check the following files for the oast1 entry:

```
$OAST_HOME/oastoltpxxxx/oastenv.sh
$TNS_ADMIN/tnsnames.ora
```

Verify if the PDB is the current PDB used. The PDBs should be the same after installation.

9.6.2 Oracle Solaris Compile Error

On Oracle Solaris systems, the `no cc` compiler error is noticed. This is a known issue.

The workaround is to install Sun Studio to get a valid compiler. For example, use the following path for cc:

```
/opt/SunProd/studio12u3/solarisstudio12.3/prod/bin
```

9.6.3 How to Compile and Build for Separate PDB Homes

When CDB is enabled, you can run the compiler and load the data using the oast_main.sh script from the screen output. The oast_main.sh file is used to invoke DBCA and to run commands like compile and load only once. However, you can also run the compiler from a separate pdb oast home. Use the following commands:

```
cd $OAST_HOME/cdb_pdb1
./nrunoastoltp5.sh -kitcompile
./nrunoastoltp5.sh -d y
```

If you have run the oast_main.sh -d y command for all the PDBs, then you will notice that the data is loaded in the same manner. If you choose loading the data offline, then the DATA_DIR is used for .csv files. To use a different method for loading the data for different PDBs or to perform offline loading of data where you can store all the .csv files in different directories, perform the following steps:

1. Load the data using the -d y option from each pdb oast home.
2. Run the oast_main.sh -d y command to load data silently in all the PDBs.

9.6.4 Why Does OAST Installation Not Work in EMACS

OAST installation does not work while running under EMACS. This issue is also noticed when you configure an Oracle RAC installation and depends on how EMACS handles the control characters.

9.6.5 Using MEMORY_TARGET and SGA_TARGET

There are discussions regarding the use of these targets. For example, when DB_CACHE_SIZE is set to zero, changing the use of these targets can cause the buffer cache to fluctuate immensely. This not only leads to thrashing of memory, but also affects the processes that handles these targets. For performance reasons, Oracle recommends to set the SGA_MAX_SIZE and DB_CACHE_SIZE parameters while setting SGA_TARGET and MEMORY_TARGET to zero. This means that you should assign more static values for the SGA size. OAST is a stress test set rather than a performance test. If the database is being stressed, then there should not be too much SGA resizing. However, it is better to set the DB_CACHE_SIZE because this will set a lower bound and the resizing will be low.

9.6.6 How to Differentiate the Resources for Different PDBs

The different stress tests in OAST are meant to stress different resources. With the use of PDBs in the same CDB the resources are shared as the database to each workload can not be tuned. The Resource Manager can turn the cpu and parallel server usage between the PDBs, but not SGA and I/O usage. The initialization parameter OPTIMIZER_INDEX_COST_ADJ can be used to differentiate the preference for using the index access against full table scan for each PDB. We must validate that with the use of the Resource Manager for cpu, the initialization parameter OPTIMIZER_INDEX_COST_ADJ for index preference and data sizing that we can obtain enough differentiation in the configuration of the CPU, OLTP, and DSS tests in a single CDB.

9.6.7 ORA-00845: MEMORY_TARGET Not Supported on this System

Reason: In this release, we enable Automatic Memory Management (AMM) when creating an OAST database, and by default, the value for the MEMORY_TARGET parameter is about 50 percent of the total memory. If you create more databases than the first one, you might see the following error during DBCA:

```
ORA-00845:MEMORY_TARGET not supported on this system.
```

From the error message help page, the following message is displayed:

```
ORA-00845:MEMORY_TARGET not supported on this system.
Cause: Insufficient memory to support SGA and PGA sizes.
Action: Ask your system administrator to increase the shared memory file system.
For example, on Linux systems, increase /dev/shm.
```

And by default, shmall is set to half the size of total memory. For multiple databases you must change the value of shmall to the total memory value. Alternatively, you can also enter a lower value for memory percentage during OAST installation.

9.6.8 How to Add and Modify the Database Parameters

OAST uses a dbca template file to create a database and some parameters like memory_target are set during installation. If you want to add some new parameters or modify the value of existing parameters, then you must modify the template before you run dbca y.

Perform the following steps:

1. Navigate to the $OAST_HOME/dbca_template directory.
2. Open the oastdb.dbt file.
3. Find the lines for which you want to modify the value. For example, to modify the value for db_files parameter, you should search for the following line:

   ```
   <initParam name="db_files" value="500"/>
   ```

4. Change the value from 500 to 1000.
5. Save the oastdb.dbt file.
6. Run -dbca y to create the database with the new parameter values.

A

Examples of OAST Installation

This section contains examples of OAST installation. It includes the following topics:

- Section A.1, "Example of OAST Installation in the Client/Server Mode"
- Section A.2, "Example of OAST Installation in the Server/Server Mode"

A.1 Example of OAST Installation in the Client/Server Mode

This section contains an example of OAST installation in the client/server mode.

`OAST_HOME` is an area that stores necessary OAST configuration files. This directory must not be shared.

1. Enter the OAST home directory for [`/home/oracle/oast/home`]: `/home/oracle/oast/`*`home11`*

   ```
   OAST HOME:/home/oracle/oast/home12
   ```

2. Client/Server Environment?

   ```
   1 Yes
   2 No
   1
   ```

   You chose: Yes

3. Enter the Oracle home directory for Oracle Database Client: `/u01/app/oracle/product/12.1.0/`*`dbhome_1`*

   ```
   CLIENTHOME=/u01/app/oracle/product/12.1.0/dbhome_1
   ```

4. Enter the host name for the client:

   ```
   example:node1:rac3
   ```

5. Enter the Oracle home owner for Oracle Database Client:

   ```
   login:oracle
   checking connection to system rac3 ...
   Client OS:Linux
   Client Oracle Version:32-bit
   Getting Server(s) information...
   Your OS:Linux
   ```

6. Enter the Oracle home directory for Oracle Database:

   ```
   ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
   ```

7. Enter the Oracle base directory:

   ```
   ORACLE_BASE=/u01/app/oracle
   ```

8. Enter the OAST database name:

   ```
   OASTDB_NAME=oltpdb
   ```

9. Would you like to create OAST database as a Container database (CDB)?

   ```
   1 Yes
   2 No
   1
   ```

10. How many pdbs do you want to create in this container db? [1-252]:

    ```
    3
    ```

11. We are going to create the following pdbs in the following directories:

    ```
    PDB Name: oltpdb_pdb1 PDB_OAST_HOME: /home/oracle/oast/home/oltpdb_pdb1
    PDB Name: oltpdb_pdb2 PDB_OAST_HOME: /home/oracle/oast/home/oltpdb_pdb2
    PDB Name: oltpdb_pdb3 PDB_OAST_HOME: /home/oracle/oast/home/oltpdb_pdb3
    ```

    > **Note:** 1 G of seed data = 5 Warehouses = 1.7 G in tablespace size (free and used).

    ```
    Please enter data size for pdb oltpdb_pdb1 [10 Gigabytes]
    10
    Size for pdb oltpdb_pdb1 is 10 GB

    Please enter data size for pdb oltpdb_pdb2 [10 Gigabytes] :
    10
    Size for pdb oltpdb_pdb2 is 10 GB

    Please enter data size for pdb oltpdb_pdb3 [10 Gigabytes] :
    10
    Size for pdb oltpdb_pdb3 is 10 GB
    ```

12. Would you like to enable Hybrid Columnar Compression?

    > **Note:** HCC feature is supported only on storage of Exadata/ZFS/Axiom.

    ```
    1 yes
    2 No
    2
    ```

13. Would you like to enable the fast recovery area?

    ```
    1 enable
    2 disable
    1
    ```

    You chose: enable

14. Enter `db_recovery_file_dest`.

    For example, `+FRA`.

15. Enter `db_recovery_file_dest_size` in MB.

    For example, *204800* for 200 GB: *204800*

16. Would you like to enable database archive mode?

    ```
    1 enable
    2 disable
    1
    ```

    You chose: enable

17. Select Oracle database options:

    ```
    1 Cluster
    2 Non-Clustered
    1
    ```

    You chose: Cluster

18. Total System Memory (Megabytes) is 14651 Megabytes

    > **Note:** For multiple databases on the same host, ensure that you have enough shared memory mount area(`/dev/shm`) percentage of total memory used for database `oltpdb` (10-90) [33%] is 50.

19. rac1 rac2

    The Oracle home server owner:

    ```
    Number of nodes:2
    login:oracle
    checking connection to system rac1 ...
    checking connection to system rac2 ...
    creating oast_cluster directory on system rac1 ...
    creating oast_cluster directory on system rac2 ...
    rac1 rac2
    Number of CPUs=8
    Free System Memory (Megabytes) for Oracle is 16074 Megabytes
    ```

    > **Note:**
    >
    > 1 GB of seed data = 15 Warehouses = 1.3 GB in tablespace size (free and used)
    >
    > Target OLTP SEED data size in GB [10 GB]:100

20. `db_block_size=`*8192*

    ```
    Calculating the size of database...
    ```

21. Are your database files stored on a file system or ASM?

    ```
    1 filesystem
    2 ASM (Automatic Storage Management)
    2
    ```

    You chose: ASM (Oracle Automatic Storage Management)

22. Specify the ASM disk group name for data (without the leading +)... *`DATA`*

23. Enter the number of online redo log groups? [Default is 2 groups, maximum is 255]: *3*

24. Enter the number of online redo log member in each group? [Default is 1 member, maximum is 5]: *1*

25. Specify the ASM disk group name for log group member 1 (without the leading +)... *LOG*

26. OAST uses ASM disk group `+DATA` for data files and `+LOG` for online redo logs:

```
filesystem_loc:+DATA
Disk Storage requirement for Oracle Database:

Number of warehouses:1500
Total Database size (used and free):130500 Megabytes

Number of nodes in cluster:2
Number of log groups:3
Number of members per log group:1
Size of each redo member:500 Megabytes
```

27. Press **Enter** to confirm/continue or **e** to exit.

28. Verify the following information:

```
Input Summary:
Number of nodes:2
Systems in the Cluster:
rac1
rac2

OAST_HOME:/home/oracle/oast/home12
OS:Linux
ORACLE HOME:/u01/app/oracle/product/12.1.0/dbhome_1
ORACLE BASE:/u01/app/oracle
OAST DB NAME:oltpdb
OAST DB INSTANCE NAME:oltpdb
OAST SERVICE NAME:oltpdb
DB BLOCK SIZE:8192
Number of CPUs:8
Memory Size:16074 Megabytes
32-bit Oracle 12.1.0.0.0
Datafiles location:+DATA
Total Datafiles Size:130500 Megabytes
Log group member location:+LOG
Log group member Size:3000 Megabytes

Recommendations:
Number of Warehouses:1500
Number of Users (default):500
Number of datafiles per table (default):5
degree of parallelism (default):16
Number of oastoltp loaders (default):16
Disk storage requirement (Megabytes):133500
SGA (Megabytes):12859
1) Continue
2) Modify
3) Exit
1
```

You chose: Continue

```
Generating kit for 1500 warehouses in /home/oracle/oast/home11 ...
generating p_create.ora ...
generating p_build.ora ...
generating p_run.ora_oastdss ...
generating common.ora ...

generating p_create.ora ...
generating p_build.ora ...
generating p_run.ora_oastdss ...
generating common.ora ...
generating createware.sql...
generating createdist.sql...
generating createhist.sql..
generating createordr.sql...
generating createnord.sql...
generating createitem.sql...
generating createiware.sql...
generating createidist.sql...
generating createiitem.sql...
generating createicust1.sql...
generating createicust2.sql...
generating createistok.sql...
generating createiordr1.sql...
generating createiordr2.sql...
generating createdb.sh...
generating createrollback.sql...
generating createcust.sql...
generating createordl.sql...
generating createstok.sql...
generating createrollsegs.sql...
generating loadware.sh...
generating loaddist.sh...
generating loaditem.sh...
generating loadhist.sh...
generating loadnord.sh...
generating loadordrordl.sh...
generating loadcust.sh...
generating loadstok.sh...
generating createinord.sql...
generating createiordl.sql...
generating offlinerollsegs.sql ...

Transferring files to system rac1: /home/oracle/oast/home11/oast_cluster/ ...
Extracting OS watcher...
Transferring files to system rac2: /home/oracle/oast/home11/oast_cluster/ ...
Extracting OS watcher...
Transferring files to client system rac3:/home/oracle/oast/home11/oast_
clients...
Extracting OS watcher...
This is a Cluster.
Oracle instance in system rac1 is oltpdb1
Oracle instance in system rac2 is oltpdb2
OAST Master Node: rac1
The kit directory: /home/oracle/oast/home11/oastoltp1500
The run script: /home/oracle/oast/home11/nrunoastoltp1500.sh
Database creation and the run scripts are in the /home/oracle/oast/home11
directory.
To create the database and load the seed data, perform the following steps:
Navigate to the /home/oracle/oast/home11 directory.
Run the following command:
    nrunoastoltp1500.sh -dbca y -d y
cd /home/oracle/oast/home11/oast_client
nrunoastoltp1500.sh -n testrun
or
nrunoastoltp1500.sh -help for various options

cd /home/oracle/oast/home11/oast_clients
 ./nrunoastoltp1500.sh -kitcompile
utils/orasim_genjobfile.sh info: /home/oracle/oast/home11/orasim/orasim_
jobfile1500 is created.
Starting RAC Elevator Setup ...
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin is not shared.

The installation is in the client/server mode. Execute the commands from the
client computer:
The compilation for the transaction drivers is required in this release. Perform
the following steps only on the client computer:
Following are the instructions to run Oracle RAC elevator test:

cd /home/oracle/oast/home11/oast_cluster/oastoltp1500/nxhome
      rac_elevator.ksh -n test_name
                       -oasthome oast_home_directory
                       -w number of warehouses
                       -steps number of steps
                       [-elevator_kind test method]
                       [-shutdown_mode shutdown method]
                       [-s service_name]
                       | -h

Usage:
-n                     test name (required)
-oasthome              oast home directory (required)
-w                     number of warehouses (required)
-steps                 number of elevator steps (required)
-elevator_kind         random(r), sequentials(s), maintenance(m)
                       default:random(r)
-shutdown_mode         normal, transactional, immediate, abort, bystep, bynode
                       default:bystep
-s                     service_name is the Net Service Name in the TNS file.
                       -s without service_name, use local connection
                       default:local connection
-h                     print usage only
```

A.2 Example of OAST Installation in the Server/Server Mode

This section contains an example of OAST installation in the client/server mode.

OAST_HOME is an area that stores necessary OAST configuration files. This directory must not be shared.

1. Enter the OAST home directory for [/home/oracle/oast/home]:
   /home/oracle/oast/*home*

   ```
   OAST HOME:/home/oracle/oast/home
   ```

2. Client/Server Environment?

   ```
   1 Yes
   2 No
   2
   ```

You chose: No

```
Getting Server(s) information...
Your OS:Linux
```

3. Enter the Oracle home directory for Oracle Database:

   ```
   ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
   ```

4. Enter the Oracle base directory:

   ```
   ORACLE_BASE=/u01/app/oracle
   ```

5. Enter the OAST database name:

   ```
   OASTDB_NAME=oastdb
   ```

6. Would you like to create OAST database as a Container database (CDB)?

   ```
   1 Yes
   2 No
   1
   ```

7. Would you like to enable the fast recovery area?

   ```
   1 enable
   2 disable
   2
   ```

   You chose: disable

8. Would you like to enable database archive mode?

   ```
   1 enable
   2 disable
   2
   ```

   You chose: disable

9. Select Oracle database options:

   ```
   1 Cluster
   2 Non-Clustered
   1
   ```

   You chose: Cluster

10. rac4 rac5

    The Oracle home server owner:

    ```
    Number of nodes:2
    login:oracle
    checking connection to system rac4 ...
    checking connection to system rac5 ...
    creating oast_cluster directory on system rac4 ...
    creating oast_cluster directory on system rac5 ...
    rac4 rac5
    Number of CPUs=8
    Free System Memory (Megabytes) for Oracle is 16074 Megabytes
    ```

Note:

1 GB of seed data = 15 Warehouses = 1.3 GB in tablespace size (free and used)

Target OLTP SEED data size in GB [10 GB]:100

11. Enter `db_recovery_file_dest_size` in MB.

    For example, *204800* for 200 GB: *204800*

12. `db_block_size=`*8192*

    ```
    Calculating the size of database...
    ```

13. Are your database files stored on a file system or ASM?

    ```
    1 filesystem
    2 ASM (Automatic Storage Management)
    2
    ```

    You chose: ASM (Oracle Automatic Storage Management)

14. Specify the ASM disk group name for data (without the leading +)... *DATA*

15. Enter the number of online redo log groups? [Default is 2 groups, maximum is 255]: *3*

16. Enter the number of online redo log member in each group? [Default is 1 member, maximum is 5]: *1*

17. Specify the ASM disk group name for log group member 1 (without the leading +)... *LOG*

18. OAST uses ASM disk group `+DATA` for data files and `+REDO` for online redo logs:

    ```
    filesystem_loc:+DATA
    Disk Storage requirement for Oracle Database:

    Number of warehouses:1500
    Total Database size (used and free):13050 Megabytes

    Number of nodes in cluster:2
    Number of log groups:2
    Number of members per log group:1
    Size of each redo member:500 Megabytes
    ```

19. Press **Enter** to confirm/continue or **e** to exit.

20. Verify the following information:

    ```
    Input Summary:
    Number of nodes:2
    Systems in the Cluster:
    rac4
    rac5

    OAST_HOME:/home/oracle/oast/home
    OS:Linux
    ORACLE HOME:/u01/app/oracle/product/12.1.0/dbhome_1
    ORACLE BASE:/u01/app/oracle
    OAST DB NAME:oastdb
    OAST DB INSTANCE NAME:oastdb
    ```

```
OAST SERVICE NAME:oastdb
DB BLOCK SIZE:8192
Number of CPUs:8
Memory Size:16074 Megabytes
32-bit Oracle 11.2.0.3.0 (???)
Datafiles location:+DATA
Total Datafiles Size:13050 Megabytes
Log group member location:+REDO
Log group member Size:2000 Megabytes

Recommendations:
Number of Warehouses:1500
Number of Users (default):500
Number of datafiles per table (default):1
degree of parallelism (default):16
Number of oastoltp loaders (default):16
Disk storage requirement (Megabytes):15050
SGA (Megabytes):12859
1) Continue
2) Modify
3) Exit
1
```

You chose: Continue

```
Generating kit for 1500 warehouses in /home/oracle/oast/home ...
\007
generating p_create.ora ...
generating p_build.ora ...
generating p_run.ora_oastdss ...
generating common.ora ...

generating p_create.ora ...
generating p_build.ora ...
generating p_run.ora_oastdss ...
generating common.ora ...
generating createware.sql...
generating createdist.sql...
generating createhist.sql..
generating createordr.sql...
generating createnord.sql...
generating createitem.sql...
generating createiware.sql...
generating createidist.sql...
generating createiitem.sql...
generating createicust1.sql...
generating createicust2.sql...
generating createistok.sql...
generating createiordr1.sql...
generating createiordr2.sql...
generating createdb.sh...
generating createrollback.sql...
generating createcust.sql...
generating createordl.sql...
generating createstok.sql...
generating createrollsegs.sql...
generating loadware.sh...
generating loaddist.sh...
generating loaditem.sh...
generating loadhist.sh...
generating loadnord.sh...
generating loadordrordl.sh...
generating loadcust.sh...
generating loadstok.sh...
generating createinord.sql...
generating createiordl.sql...
generating offlinerollsegs.sql ...

Transferring files to system rac4: /home/oracle/oast/home/oast_cluster/ ...
Extracting OS watcher...
Transferring files to system rac5: /home/oracle/oast/home/oast_cluster/ ...
Extracting OS watcher...
This is a Cluster.
Oracle instance in system rac4 is oastdb1
Oracle instance in system rac5 is oastdb2
OAST Master Node: rac4
The kit directory: /home/oracle/oast/home/oastoltp1500
The run script: /home/oracle/oast/home/nrunoastoltp1500.sh

Database creation and run scripts are in the /home/oracle/oast/home directory
Navigate to the /home/oracle/oast/home directory.
Run the following command:
nrunoastoltp1500.sh -dbca y -d y

To create the database and load the seed data, perform the following steps:
The installation is in the server/server mode. Execute the following commands
from the master node rac4:

cd /home/oracle/oast/home
nrunoastoltp1500.sh -n testrun
or
nrunoastoltp1500.sh -help for various options
The compilation for the transactions drivers is required in this release. Run
the following commands only on the master node rac4:

cd /home/oracle/oast/home
 ./nrunoastoltp1500.sh -kitcompile
utils/orasim_genjobfile.sh info: /home/oracle/oast/home/orasim/orasim_
jobfile1500 is created.
Starting RAC Elevator Setup ...
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin is not shared.
Following are the instructions to run Oracle RAC elevator test:

cd /home/oracle/oast/home/oast_cluster/oastoltp1500/nxhome
      rac_elevator.ksh -n test_name
                       -oasthome oast_home_directory
                       -w number of warehouses
                       -steps number of steps
                       [-elevator_kind test method]
                       [-shutdown_mode shutdown method]
                       [-s service_name]
                       | -h

Usage:
-n                     test name (required)
-oasthome              oast home directory (required)
-w                     number of warehouses (required)
-steps                 number of elevator steps (required)
-elevator_kind         random(r), sequentials(s), maintenance(m)
                       default:random(r)
-shutdown_mode         normal, transactional, immediate, abort, bystep, bynode
```

```
                       default:bystep
-s                     service_name is the Net Service Name in the TNS file.
                       -s without service_name, use local connection
                       default:local connection
-h                     print usage only
```

B

CDB and PDB Utilities

Multitenant container database (CDB) is a new feature in Oracle Database 12*c* Release. To support this feature, OAST provides a set of utilities. You can use these utilities to perform the following:

- Start or stop one or more plugabble databases (PDBs).
- Rename a PDB.
- Clone a PDB.

OAST provides the following CDB and PDB utilities:

oastoltp*xxxx*/cdb/pdb_showstatus.sh

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_showstatus.sh all
```

displays the following results:

CON_ID	INST_ID NAME	OPEN_MODE
2	1 PDB$SEED	READ ONLY
3	1 CDB_PDB1	MOUNTED
4	1 CDB_PDB2	MOUNTED
5	1 CDB_PDB3	MOUNTED

The command, `cdb/pdb_showtatus.sh cdb_pdb1` displays the following result:

CON_ID	INST_ID NAME	OPEN_MODE
3	1 CDB_PDB1	MOUNTED

oastoltp*xxxx*/cdb/pdb_get_allnames.sh

This utility is used to display one or all the existing PDBs and their corresponding status. For example, the command,

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_get_allnames.sh
```

displays the following results:

```
CDB_PDB1
CDB_PDB2
CDB_PDB3
```

oastoltp*xxxx*/cdb/pdb_start_allnodes.sh

This utility starts a single PDB on a single instance when Single Instance is enabled and on all nodes when Oracle RAC is enabled. For example,

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_start_allnodes.sh cdb_pdb1
```

displays the following results:

```
Starting PDB cdb_pdb1 on all nodes
Start PDB cdb_pdb1 open on node bej301301
```

oastoltp*xxxx*/cdb/pdb_close_allnodes.sh

This utility is used to shut down a single PDB on a single instance when Single Instance is enabled and on all nodes when Oracle RAC is enabled. For example,

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_close_allnodes.sh cdb_pdb1
```

displays the following results:

```
Close PDB cdb_pdb1 open on node bej301301
Pluggable database altered.
```

oastoltp*xxxx*/cdb/pdb_rename.sh

This utility renames an existing PDB. For example,

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_rename.sh cdb_pdb1 newpdb
Rename PDB cdb_pdb1 to newpdb
Stopping CDB cdb
Starting CDB cdb
Elapsed:00:00:00.06
Pluggable database altered.
Elapsed:00:00:02.60
Session altered.
Elapsed:00:00:00.10
Pluggable database altered.
Elapsed:00:00:05.88
```

CON_ID	INST_ID NAME	OPEN_MODE
3	1 NEWPDB1	READ WRITE

The PDB `cdb_pdb1` has been renamed to `newpdb` now.

oastoltp*xxxx*/cdb/pdb_clone.sh

This utility clones a new PDB from an existing PDB. For example,

```
cd $OAST_HOME/oastoltpxxxx
cdb/pdb_clone.sh cloned_pdb_cdb_pdb1
```