



**Hewlett Packard**  
Enterprise

Technical white paper

# **HPE Nimble Storage Deployment Considerations for Oracle Database**



# Contents

Executive summary.....	3
General Oracle considerations.....	3
Parameter db_file_multiblock_read_count.....	3
Parameter filesystemio_options.....	3
Redo block size.....	4
Storage layout considerations.....	4
Logical volume managers.....	4
Partitions.....	4
Number of disks or LUNs.....	5
Production versus nonproduction databases.....	5
Performance policy block size and caching.....	5
Separating datafiles from log files and FRA files.....	6
Volume provisioning.....	6
Volume collections.....	6
Multiple array deployment.....	6
Operating system tunings.....	6
Multipathing.....	6
Parameters user_friendly_names and aliases.....	7
Queue depth.....	7
iSCSI sessions.....	7
Clustering.....	8
Oracle Real Application clustering.....	8
Host-based clustering.....	8
Virtualization.....	8
Oracle on VMware.....	8
Backing up an Oracle database.....	9
HPE Nimble Storage snapshots.....	9
RMAN.....	10
Cloning an Oracle database.....	10
Restoring an Oracle database.....	11
Compression.....	11
Deduplication.....	12
HPE Nimble Storage Oracle Data Manager (noradatamgr).....	12



## Executive summary

There are many ways to deploy Oracle databases on HPE Nimble Storage arrays. This document reviews some of the design considerations, parameters, and best practices that Oracle DBAs, storage architects, and administrators should evaluate.

### Target audience

This guide is for Oracle database administrators (DBAs), solution architects, storage engineers, system administrators, and IT managers who analyze, design, and maintain Oracle database environments on HPE Nimble Storage arrays from Hewlett Packard Enterprise. Readers are assumed to have a working knowledge of Oracle databases, the operating system (OS) on which they are deploying their solution, and basic HPE Nimble Storage array operations.

### Document purpose

This deployment considerations guide describes best practices for deploying and using Oracle databases on HPE Nimble Storage arrays. It offers considerations for database instance tuning and storage layout, and it provides information about OS tuning, clustering, virtualization, copy management, and data protection.

## General Oracle considerations

Some Oracle initialization parameters can affect performance when Oracle databases are deployed on HPE Nimble Storage arrays. For maximum storage performance, follow the recommendations in this section to configure your Oracle databases. These recommendations are valid for both standalone and Oracle Real Application Clusters (RAC) databases.

### Parameter `db_file_multiblock_read_count`

The parameter `db_file_multiblock_read_count` controls the number of Oracle blocks that are read during a single Oracle sequential I/O request. This parameter does not affect random I/O requests.

Oracle recommends that you avoid setting this parameter in the `init.ora` parameter file. Leaving this parameter unset enables Oracle to optimize the number of sequential blocks that are retrieved.

During Oracle upgrades from legacy database versions, it is typical for this parameter to remain set at lower values. If the value is set too low, performance problems might occur.

### HPE recommendation

Do not set or remove the Oracle parameter `db_file_multiblock_read_count` from the `init.ora` file.

### Parameter `filesystemio_options`

The Oracle initialization parameter `filesystemio_options` controls both direct and asynchronous I/O operations from an Oracle perspective.

### Options available

The following options are available for this parameter:

- `setall`: Enables Oracle to use both asynchronous and direct I/O.
- `directio`: Enables direct I/O but not asynchronous I/O. This option allows I/O to bypass the OS cache and maximizes the use of the Oracle buffer cache.
- `async`: Enables asynchronous I/O but not direct I/O. This option helps to increase the amount of parallelization in the Oracle I/O and allows Oracle to perform other work rather than waiting for I/O completion.
- `none`: Enables Oracle to perform I/O synchronously and to use the OS cache first for all I/O.

When direct I/O is enabled on an Oracle instance, the Oracle I/O bypasses the OS cache and flows directly to the Oracle buffer cache that is defined in the Oracle system global area (SGA). In most cases, it is best to enable direct I/O and allocate the OS cache to other resources such as the SGA. Avoiding the OS cache with direct I/O enables the buffer cache to handle I/Os more efficiently, which ultimately helps to reduce the amount of latency associated with the I/O. If direct I/O is enabled, the Oracle buffer cache must be sized properly to accommodate I/O requirements.

Asynchronous I/O enables Oracle I/O to be parallelized. It also allows Oracle to perform other work instead of waiting for I/Os to complete. Before asynchronous I/O, Oracle DBAs adjusted the number of dbwriters to enable more I/O parallelism. Asynchronous I/O is typically enabled, which allows the database to perform operations in a more efficient and parallel manner. Performing asynchronous I/O creates no additional data risk, and other operations that are typically synchronous still maintain the same I/O pattern.



**HPE recommendation**

Set this parameter to `filesystemio_options = setall`.

**Redo block size**

The redo log file is a critical part of an Oracle database because all transactions are ultimately committed to an Oracle redo log. Redo logs can be set up in different ways, but HPE Nimble Storage arrays can handle even the most complex deployment. Redo log records are variable in length, so the underlying storage must be able to handle writes of any length and must perform extremely well with them. On the HPE Nimble Storage system, all writes are initially acknowledged and committed into a nonvolatile dual inline memory module (NVDIMM), which immediately allows Oracle to move on to the next outstanding I/O operation.

The typical size of Oracle redo logs is either 512 bytes or 4 KB. The default size is 512 bytes.

**HPE recommendation**

HPE Nimble Storage arrays fully support both redo block sizes. Use the option that is preferred by your Oracle database administrator.

**Storage layout considerations**

There are many ways to create and deploy an Oracle database on an HPE Nimble Storage array. This section describes methods for laying out disk groups and provides recommendations to help you choose the appropriate layout. Before deploying an Oracle database on an HPE Nimble Storage array, consider the following questions:

- How many disk groups are involved?
- How many disks or LUNs will be required?
- Will this database be used in a production environment, or will it be used for development or testing?
- What are the requirements for performance policy block size and caching?
- Will the datafiles be separated from the log and flash recovery area (FRA) files?
- Will the database be spanned across multiple storage controllers?

In general, there are no right or wrong answers when you deploy an Oracle database on an HPE Nimble Storage array. The following implementation suggestions are based on the type of database being deployed.

**Logical volume managers**

If more than two LUNs are involved, HPE recommends that you use a logical volume manager (LVM). Typical LVMs are Veritas VxVM, Linux® LVM2, and Oracle Automatic Storage Management (ASM). Using more than two LUNs increases performance, but there are diminishing returns when more than eight LUNs are used for a data disk group. Be sure to use an LVM policy that distributes data across all LUNs in a parallel manner.

If you create Oracle ASM disk groups or LVM volume groups, HPE recommends that you create a minimum of two or three disk groups for the Oracle database. Create a disk group for datafiles and a disk group for log files. If you are using multiplexed redo logs, or if you have a flash recovery area, you can create a third disk group as well.

The main reason for separating datafiles from log files is to enable easy restore and recovery as well as to help clone the database. For certain types of smaller, mostly unmanaged database deployments, you can use a single disk group. In this case, all files (data and log) are stored in the same disk group.

**HPE recommendation**

Create at least two disk groups: one for datafiles and one for log files, with potentially a third disk group for FRA files and multiplexed redo files.

**Partitions**

HPE highly recommends that you avoid using partitions when you create a device, and use the whole disk instead.

Although disk partitions are supported, they provide no advantage over whole disks, and if they are not set up properly, they can cause problems with I/O alignment. In addition, the HPE Nimble Storage Oracle Application Data Manager tool does not support partitions.

**HPE recommendation**

When creating a device, use the whole disk and do not use partitions.



## Number of disks or LUNs

When deploying an Oracle database on an HPE Nimble Storage array, the DBA must ask, "What is the best number of disks or LUNs to use?" This is a hard question, and its answer depends on the answers to questions such as the following:

- Is this a production database?
- Is this a development or test database?
- How many disk groups will be used?
- How many in-flight I/Os are generated at a given time?

There are two keys to determining how many LUNs are required for a given Oracle database deployment on an HPE Nimble Storage array:

- Factor in the throughput requirements.
- Keep in mind the management headache that a high number of LUNs might cause.

In general, an Oracle database can be deployed on an HPE Nimble Storage array with as few or as many LUNs as needed. It is typical to have more LUNs per disk group for a production database than for a development or test database. An important consideration for throughput requirements is to have enough LUNs to handle the number of in-flight I/Os for the workload.

## Production versus nonproduction databases

The deployment considerations for production databases differ from those for nonproduction databases. The following tables show layouts for production and nonproduction databases that are deployed on HPE Nimble Storage arrays.

**Table 1.** Production database example.

Disk group name	Number of LUNs/disks	Types of files
DATA	4–8	Data files only
LOG	2–4	Redo logs, control file
FRA	2–4	Multiplexed redo logs, control file, archive logs

**Table 2.** Nonproduction database example.

Disk group name	Number of LUNs/disks	Types of files
DATA	1–4	Data files only
LOG	1–2	Redo logs, control file, archive logs (FRA is optional.)
FRA (optional)	1–2	Multiplexed redo logs, control file, archive logs

## Note

For the nonproduction example, the FRA disk group is optional. If the FRA is not used, then HPE recommends that you put all redo and archive logs and all control files in the LOG disk group.

## Performance policy block size and caching

Oracle databases are deployed with varying Oracle block sizes. For online transaction processing (OLTP) databases, the Oracle block size is typically 8 KB. For a decision support system (DSS) or data warehouse, the Oracle block size varies between 16 KB and 32 KB.

Every HPE Nimble Storage volume is associated with a performance policy, which has a block size associated with it. HPE recommends that when you create the performance policy, you match the Oracle block size that is associated with the database.



HPE recommends that all volumes associated with an Oracle database enable caching, regardless of the types of files that are stored in those volumes. Even though log files are typically not cached, it is important to enable caching when both log files and control files are placed in the same location.

**HPE recommendation**

Associate the HPE Nimble Storage volume with a performance policy that has the same block size as the Oracle database. Be sure to enable caching for all volumes associated with an Oracle database.

**Separating datafiles from log files and FRA files**

HPE highly recommends that you separate Oracle datafiles from log files and FRA files and place them into distinct disk groups. Separation makes the restore and recovery of Oracle databases much easier.

If it is not possible to separate datafiles from log files, there should be no noticeable performance impact. A lack of separation, however, limits the types of restore and recovery and the types of database cloning options that are available.

**HPE recommendation**

Use different disk groups for Oracle datafiles and for log and FRA files.

**Volume provisioning**

When you create HPE Nimble Storage array volumes for Oracle database deployments, HPE recommends that you choose thin provisioning. Thin provisioning provides a way to optimize the use of available storage. It relies on on-demand allocation of blocks of data instead of allocating all blocks up front as thick provisioning does.

**HPE recommendation**

Use thin provisioning when you create volumes for Oracle databases.

**Volume collections**

Volume collections enable you to coordinate snapshots for multiple volumes that are grouped together. From a database perspective, HPE recommends that you assign volumes from the same single-instance or RAC database to a single volume collection. Although HPE Nimble Storage technology supports putting multiple databases into the same volume collection, HPE does not recommend doing so because it makes recovery of a single database more difficult.

**HPE recommendation**

Create a volume collection for each individual single-instance or RAC database. Do not place multiple databases into the same volume collection.

**Multiple array deployment**

Consider carefully before spanning multiple arrays. Spanning a database across multiple arrays can help improve performance and increase available capacity. However, one concern that you must evaluate is the possibility of array failure. If a database is spanned between arrays and one of the arrays fails, then that database becomes unavailable.

ASM and other LVMs offer you the ability to span a disk group across multiple arrays to help improve performance. By creating a disk group with LUNs from separate arrays, you can maximize resources from each of the arrays to potentially provide better performance. In addition, spanning a disk group across arrays can increase capacity if the original disk group is full.

**HPE recommendation**

Although it is feasible to span databases across arrays, avoid doing so unless spanning is required to achieve capacity or performance goals.

**Operating system tunings**

The vendor documentation provides the most definitive information about tuning any OS. HPE also provides a Connectivity Guide or Deployment Considerations document for each supported OS. For other considerations, see those documents, which are located on [HPE InfoSight](#).

**Multipathing**

Multipathing is a technique that allows data to be transferred through more than one physical path from a host to a storage device. Multipathing is discussed in all of the OS Connectivity Guides and Deployment Considerations documents that are available on [HPE InfoSight](#), as well as in the documentation from each OS vendor.



Proper multipathing configuration is extremely important for the successful deployment of any Oracle database. See the guides for how to configure multipathing for your OS.

### Parameters `user_friendly_names` and `aliases`

When you include `user_friendly_names` or `aliases` in the `multipath.conf` file, multipathing devices can be recognized and identified more easily. HPE and Oracle support both `user_friendly_names` and `aliases`; however, do not use both or configure them at the same time in the `multipath.conf` file. There have been some problems related to SAN booting with `user_friendly_names`, so give careful consideration to enabling `user_friendly_names`.

### Queue depth

Queue depth represents the number of I/O requests that can be queued to a storage controller from a given host. Operating systems set queue depth values at the level of the individual SCSI device and also at the level of the host bus adapter (HBA). In general, it is easiest to set the HBA queue depth to a larger value and keep the LUN queue depth at its default setting. In this case, ensure that the HBA queue depth is not overrun by a large number of LUNs.

For the most relevant information on setting specific queue depth values, review the OS vendor's documentation on queue depth. That documentation should also provide more information about the following considerations for OLTP databases and data warehouse databases.

#### OLTP databases

Because OLTP databases generally result in many small I/O requests, HPE recommends that you set the queue depth to a higher number (such as 64 or 128) on the HBA. You should always perform testing to ensure that any changes to the queue depth setting have the desired effect of increasing performance.

#### Data warehouse databases

For data warehouse or DSS databases, it is typical to set the HBA queue depth level to a value of 32 or 64. The smaller numbers are appropriate because a data warehouse workload has fewer but larger block requests.

---

### Note

As with all recommendations, it is important to test all changes that fall outside of the normal default values set by the OS and HBA vendor.

---

### iSCSI sessions

In most Linux distributions, the parameter `node.session.nr_sessions` is located in the `/etc/iscsi/iscsid.conf` file. This parameter determines the total number of iSCSI sessions that are created while the database is connecting to the storage devices. This setting is completely configurable based on the performance and connectivity requirements of the database and the requirements of other applications on the same host.

For certain versions of Linux, the HPE Nimble Storage Connection Manager (NCM) tool is available. This tool automatically sets the `node.session.nr_sessions` parameter to the appropriate value based on your configuration. HPE highly recommends that you use this tool if it is available. If it is not available, HPE recommends that you use the following formula to determine a starting point:

#### $(I \times A)$

Where:

- **I** is the number of interfaces configured for iSCSI on a host.
- **A** is the number of arrays in a volume pool.

The value should be set to a minimum of 2. For increased database performance requirements in larger deployments, HPE recommends that you test with higher values.

#### HPE recommendation

If the NCM tool is available, use it to set the value of `node.session.nr_sessions`. If the tool is not available, use the formula provided, with the minimum value set to 2.



## Clustering

You can use Oracle RAC to configure clustering, or you can use an OS-specific cluster solution. The following sections provide HPE recommendations for clustering.

### Oracle Real Application clustering

Oracle RAC requires you to consider two parameters: `disktimeout` and `misscount`.

#### Parameter `disktimeout`

The `disktimeout` parameter controls the threshold at which I/O to the Oracle RAC voting disk must be acknowledged. If the threshold is surpassed, the RAC node is evicted from the cluster and rebooted. The default value for this parameter is 200. The default value should be sufficient to cover all HPE Nimble Storage array operations, even if a storage failover happens to occur.

#### HPE recommendation

Use the default value for `disktimeout`, keeping in mind that this value might need to be adjusted. It is very important to evaluate how a deployed RAC instance will operate with the default settings.

#### Parameter `misscount`

The `misscount` Oracle RAC parameter relates to the network heartbeat between RAC nodes. The default is set at 30 seconds. The following situations might require adjusting this parameter:

- You have boot disks on Fibre Channel (FC) LUNs or on virtualized datastores such as a Virtual Machine Disk (VMDK) file.
- Oracle Grid binaries are located on a LUN.

During storage failover, the I/O to boot LUNs might be affected, which might cause the entire OS or Oracle Grid binaries to hang. If the time surpasses 30 seconds, the default `misscount` value might be exceeded, which would cause a node to be evicted. HPE highly recommends that you test to verify that the default value for `misscount` works with any SAN-booted configuration.

#### HPE recommendation

Use the default value for `misscount`, keeping in mind that this value might need to be increased after storage failover is tested.

---

### Note

When deploying Oracle RAC, it is extremely important to test storage failover with the settings that are being used for `disktimeout` and `misscount`.

---

## Host-based clustering

HPE supports many different host-based clustering products, including Sun Clusters and Veritas Storage Foundation Suite. When you deploy an Oracle database with these solutions, it is important to verify that the solution is supported by both Oracle and HPE. To verify that a configuration is supported by HPE, consult the support matrix on [HPE InfoSight](#). HPE recommends that you follow the individual vendor's documentation when you set up a cluster solution.

## Virtualization

Virtualization with VMware ESXi™, Oracle Virtual Machine (OVM), Kernel-based Virtual Machine (KVM), and Microsoft® Hyper-V has become a widely available option for deploying Oracle and Oracle RAC systems. Each of these solutions provides its own benefits; however, you should carefully consider the following key issues before selecting a virtualization platform for your environment:

- The use of paravirtualized drivers
- The importance of not overcommitting virtual RAM

### Oracle on VMware

To deploy an Oracle database in a VMware® environment running on an HPE Nimble Storage device, you must consider many different options. In general, HPE supports the use of VMware ESXi for Oracle databases. However, each deployment option should be evaluated to determine the best option for the given database.





**Paravirtualized SCSI adapters**

HPE highly recommends that you use a paravirtualized SCSI adapter whenever possible. This type of adapter optimizes the guest OS for performance as compared to an emulated adapter, which would spend more CPU cycles emulating the behavior of physical hardware.

**HPE recommendation**

Use a paravirtualized SCSI adapter.

**Virtual RAM**

As a rule, HPE does not recommend that you overcommit virtual RAM when you create a guest OS for an Oracle database. In some cases, if there is too much overcommitted RAM, the underlying blocks of the Oracle SGA might be swapped out to disk, which could affect performance.

**HPE recommendation**

Do not overcommit virtual RAM to a point at which the Oracle SGA might be swapped out to disk.

**Direct-to-guest**

In a direct-to-guest configuration, the storage LUNs for the Oracle database are connected directly from the guest OS. They are mapped to each individual virtual machine (VM) through a software iSCSI or FC connection. The direct-to-guest configuration provides the most efficient way to complete a backup with HPE Nimble Storage snapshots.

The process for snapshot backups is very similar to the process for physical hardware and OS backup, which is described in “Backing up an Oracle database.” In addition, you can use the Oracle Recovery Manager (RMAN) to back up a database on a direct-to-guest VM.

**Virtual machine disk**

Virtual machine disks (VMDKs) provide an easy way to deploy Oracle databases, especially for single instances. On a particular VM, a VMDK is presented as a local SCSI device. It can then be formatted with a file system, and the database binaries and files can be stored on that newly created file system. The main difficulty with using VMDKs as the primary way to deploy an Oracle database is the inability to easily coordinate a hot backup of the database.

**HPE recommendation**

For VMDKs, it is ideal to have the database in hot-backup mode and the VMDK in a quiesced state before backing up the database with an HPE Nimble Storage snapshot.

**Raw device mapping**

Raw device mappings (RDMs) are a viable way to deploy Oracle databases, and HPE Nimble Storage arrays support RDMs. However, RDMs are not widely used in Oracle deployments.

**Backing up an Oracle database**

HPE Nimble Storage snapshots provide a reliable way for DBAs to back up their Oracle databases. HPE Nimble Storage technology also supports the use of Oracle RMAN for backups.

**HPE Nimble Storage snapshots**

HPE Nimble Storage snapshots provide the ability to create a nearly instant point-in-time image of the active file system. A snapshot freezes the current active blocks of the storage volume and keeps track of those frozen blocks as the data changes. When a snapshot is deleted, the pointers to the frozen blocks are deleted. An HPE Nimble Storage snapshot is nearly instantaneous, and it requires minimal array overhead.

HPE supports two different types of backups for Oracle databases:

- Application-consistent backups
- Crash-consistent backups

For application-consistent backups, the Oracle database must be placed into hot-backup mode before the snapshot of the volume collection is taken. When the database is in hot-backup mode, it is quiesced and ready to have a snapshot taken. It is important that datafiles, control files, redo logs, and archive files have snapshots taken at the same time. In addition, the archive logs should be maintained in a separate disk group from the datafiles.

For a crash-consistent backup, the database does not need to be in hot-backup mode. A volume collection snapshot can be taken at any point without the database being quiesced. In this case, it is important that all datafiles, control files, and redo logs are captured. This capture



can be accomplished by using a volume collection snapshot that contains all of those files. Snapshots can be used to maintain recovery point objectives (RPO) and recovery time objectives (RTO).

Because HPE Nimble Storage snapshots require a minimal amount of space, they can be taken multiple times a day. This frequency can help decrease the data risk in RTO, and it can help provide additional restore and recovery points that become important in RPO. Depending on the backup requirements for the database, snapshots can be taken as frequently as every 5 minutes, or as infrequently as required.

#### **HPE recommendation**

Use HPE Nimble Storage snapshots for primary backups or as secondary backups from RMAN. Take full backups as often as necessary.

### **RMAN**

Oracle Recovery Manager (RMAN) is a backup utility that most DBAs use to back up their databases. Oracle databases that are deployed on an HPE Nimble Storage array can be backed up by RMAN without problems. In addition to block-level recovery, RMAN has the capability to perform point-in-time recovery, which provides granularity for DBAs to restore and recover their databases.

In addition to using RMAN, HPE suggests that you implement HPE Nimble Storage snapshots as an extra layer for backups. HPE Nimble Storage snapshots are a perfect way to create a backup of a database with minimal extra work:

- They are fast, efficient, and easy to use.
- They require no additional licenses or cost.
- They are a reliable way to back up an Oracle database.

In some cases, it might be easier to use an HPE Nimble Storage snapshot than to use RMAN to perform the database restore or recovery.

#### **HPE recommendation**

Create HPE Nimble Storage snapshots frequently throughout the day for fast and efficient backups to enable quick and easy recovery. Use RMAN backups daily for longer-term retention.

### **Cloning an Oracle database**

Cloning an Oracle database can be quite tedious for an Oracle DBA. HPE Nimble Storage snapshots enable the DBA to efficiently create a copy of the database in a matter of minutes. Using the HPE Nimble Storage zero-copy cloning technology with an Oracle database snapshot is a simple, efficient, and accurate way of creating an exact copy of the database. Each snapshot of Oracle datafiles, control files, and redo logs can act as a base for the new database clone.

Zero-copy clones of the Oracle database require very little initial additional space because the blocks being copied are the same as the original ones. Space is allocated and changes are tracked as new updates and inserts are added to the cloned database. Because only a small amount of space is required for any subsequent clone, it is possible to take as many clones as needed for uses such as development and test databases.

To create an application-consistent clone of an Oracle database, follow these high-level steps:

1. Place the database in hot-backup mode.
2. Take a snapshot of the database by using a volume collection snapshot of the database's datafiles, control files, and redo logs.
3. Take the database out of hot-backup mode.
4. On the HPE Nimble Storage array, create a clone of the volumes for the datafiles, control files, and redo logs by using the snapshot that you just created.
5. Update the cloned volumes host initiator to point to the second host.
6. On the second host, discover the cloned volumes and then mount or map them.
7. Verify that the new cloned volumes or LUNs are available and that they are recognized on the new host.
8. Start the database.

---

#### **Note**

To create a valid clone, verify the version of the Oracle software that is installed on the second host and use the same version of the operating system on the second host. These steps do not cover cloning to the same host.

---



**HPE recommendation**

Use HPE Nimble Storage zero-copy clones as a simple, reliable, and efficient way to create clones of an Oracle database.

**Restoring an Oracle database**

The ability to restore an Oracle database is critical in recovering from a disaster. HPE Nimble Storage snapshots provide a way to easily restore the database back to a particular point in time.

You can use two methods to restore an Oracle database from an HPE Nimble Storage snapshot:

- The first method is to perform a full restore of all files that are associated with the database and then apply a small number of redo logs to bring the database into a consistent state.
- The second method provides a more granular restore or recovery capability. In this method, the database is cloned from a snapshot that contains the correct data; afterward, using different methods (such as export/import), the data is copied back into the original database.

To perform a full restore of a database that is using Oracle ASM, follow these high-level steps:

1. Shut down the original database.
2. Stop the associated ASM disk groups for the database.
3. Perform an HPE Nimble Storage volume restore of the volumes associated with the database by using the appropriate application or crash-consistent snapshot.
4. Start the associated ASM disk groups and verify that they are online.
5. Start the original database.

---

**Note**

A full restore overwrites any data in the current volumes. Use it only if a full restore is required.

---

**Compression**

HPE Nimble Storage arrays provide inline compression for all of the data that is stored on the array. Compression is enabled by default, but it can be disabled on a per-volume basis.

HPE recommends that you always leave compression turned on for Oracle datafiles. Compression on Oracle datafiles provides the greatest compression rate as compared to other applications. On Oracle redo and archive log files, compression does not result in significant space savings. For these files, you can turn compression off, but only if datafiles or control files are not in the same volume. Turning off compression will not make a significant difference in performance.

---

**Important**

Before you turn off compression on Oracle redo and archive log files, always verify that no datafiles or control files are contained in the same volume.

---

Oracle itself provides different levels of compression:

- Basic
- OLTP
- Hybrid Columnar Compression (HCC)

Some of these compression capabilities require additional licensing fees from Oracle. Although an HPE Nimble Storage array operates well with these compression options, HPE recommends that you disable them when deploying Oracle for the following reasons:

- HPE Nimble Storage compression is inline.
- It is easily enabled and disabled.
- It does not require additional license fees.



In addition, operating competing compression algorithms at the same time causes wasted overhead and provides minimal improvement to compression rates.

**HPE recommendation**

Enable HPE Nimble Storage compression on Oracle datafiles and control files and disable any Oracle compression algorithms.

**Deduplication**

As a rule, Oracle files do not deduplicate well. Each Oracle block contains a header that is globally unique to the database as well as a unique trailer block. This uniqueness limits the effectiveness of deduplication.

Because deduplication uses storage CPU cycles to perform the processing, HPE recommends that you disable deduplication on any volumes that are associated with Oracle. The only exception to this recommendation is if full copies of an Oracle database are stored on any volumes on the HPE Nimble Storage array. This happens, for example, when RMAN is used to back up an existing database. In this case, deduplication should be enabled to maximize space savings.

**HPE recommendation**

Disable deduplication on Oracle-related volumes unless full copies of an Oracle database also exist on any volume on the storage device.

**HPE Nimble Storage Oracle Data Manager (noradatamgr)**

The HPE Nimble Storage Oracle Data Manager (noradatamgr) is a service included with the HPE Nimble Storage Linux Toolkit (NLT). It simplifies the implementation of HPE Nimble Storage data copy management features with Oracle databases. With noradatamgr, complex workflows such as database instance cloning can be executed with as few one or two commands. The following workflows are supported:

- Create, list, and delete snapshot backups of an Oracle instance.
- Describe a database instance:
  - ASM disks and diskgroups used by the instance.
  - ASM disk device names with HPE Nimble Storage volume details.
- List the pfile and control file of an instance from a snapshot.
- Create, delete, and refresh Oracle database clones.
- Enable synchronous replication of an instance clone.
- Replicate a snapshot.
- Catalog cloned ASM diskgroups in RMAN.

The NLT is a free download that is installed directly on the database server. The following components are required:

- Linux OS
- HPE Nimble Storage arrays
- Oracle ASM for data storage (with or without Oracle ASMLIB)

Noradatamgr commands are run directly on the Linux Oracle database server, so no additional servers or VMs are needed.

You can find the NLT installer and the integration guide on HPE InfoSight at [HPE Nimble Storage Linux Toolkit \(NLT\)](#).



## Resources

HPE Nimble Storage Linux Toolkit (NLT) download

[infosight.hpe.com/org/urn%3Animble%3A0018000000ovsdmAAA/resources/nimble/software/Integration%20Kits/HPE%20Nimble%20Storage%20Linux%20Toolkit%20\(NLT\)](https://infosight.hpe.com/org/urn%3Animble%3A0018000000ovsdmAAA/resources/nimble/software/Integration%20Kits/HPE%20Nimble%20Storage%20Linux%20Toolkit%20(NLT))

Validated Configuration Matrix on HPE InfoSight

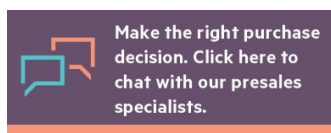
[infosight.hpe.com/app/login?logout#validated-configuration-matrix](https://infosight.hpe.com/app/login?logout#validated-configuration-matrix)

HPE Nimble Storage Documentation on HPE InfoSight


[infosight.hpe.com/org/urn%3Animble%3A0013400001QfUyiAAF/resources/nimble/docs](https://infosight.hpe.com/org/urn%3Animble%3A0013400001QfUyiAAF/resources/nimble/docs)

Learn more at

[hpe.com/storage](https://hpe.com/storage)



 **Share now**

 **Get updates**

---

© Copyright 2019 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. Oracle is a registered trademark of Oracle and/or its affiliates. VMware and VMware ESXi are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. All other third-party marks are property of their respective owners.

a00072441ENW, May 2019