NETAPP SOLUTION DEPLOYMENT GUIDELINES

Thomson Reuters Professional – Oracle on NetApp

Prepared By:
Michael Arndt
arndt@netapp.com
Version 5: January 2013

Table of Contents

1 Executive Summary

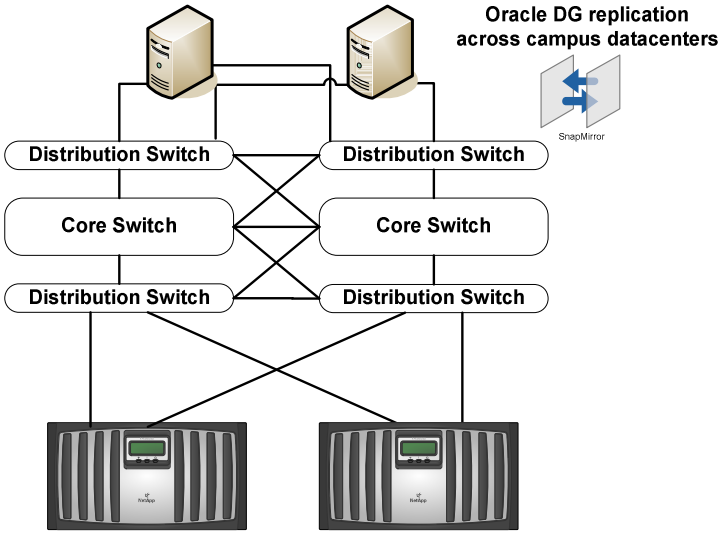
This document outlines the deployment guidelines for the Oracle environment in Thomson Reuters Professional using NetApp storage. Oracle is deployed with NetApp storage to implement an improved backup and recovery architecture using NetApp snapshots and SnapVault. Storage for the Oracle environment will be accessed via the NFS protocol, and will utilize the standard shared NAS environment already in place within each datacenter at Thomson Reuters Professional. While this document covers the majority of the NetApp storage related configurations for the LION standards within Thomson Reuters Professional, it does not cover all platform, database, and network related configuration standards. The high level architecture drawing below outlines the configuration, and the remainder of this document will provide details surrounding each component of the solution as it relates to storage and backup / recovery of the Oracle environment.

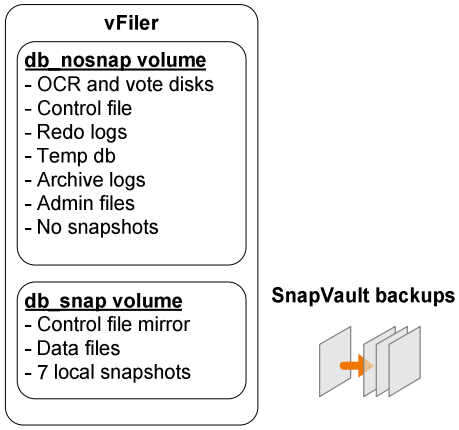
Oracle RAC Servers
- 2 node Oracle RAC cluster
- Optional DataGuard replication
- Storage connected via NFS

Cisco Network Infrastructure
- Servers connected with 1GbE or 10GbE
- Storage connected with 10GbE

NetApp Storage
- Shared storage with 15k RPM drives for high tier
- Shared storage with 7200 RPM drives for low tier
- Dedicated storage for high performance
- Typically one vFiler per database

Storage Backup / Recovery
- 7 local snapshots of "_snap" volume
- 14 snapshots stored on SnapVault secondary
- No snapshots on "_nosnap" volume
- Archive logs backed up to non-NetApp storage
- SnapRestore used for fast database recovery





2 NetApp storage system configuration and provisioning

2.1 General configuration and Data Motion considerations

The typical configuration for Oracle databases using NetApp storage within TRP is to allocate 1 vFiler per Oracle database. When configuring vFilers for use with Oracle databases, the following configuration options and limitations should be taken into consideration:

- The Oracle database servers must be able to access the vFiler via NFSv3 and SSH.
- While the Oracle database servers and the vFiler do not need to reside on the same network subnet, the path should include a minimal number of network hops.
- The following performance related options should be configured on storage controllers used for Oracle:

  nfs.ifc.rcv.high 393216
  nfs.ifc.rcv.low 33170
  nfs.ifc.xmt.high 64
  nfs.ifc.xmt.low 48
- The following performance related options should be configured on vFilers used for Oracle:

  nfs.tcp.recvwindowsize 262144
- Each vFiler will be configured with a local account, named *oracle*, used for communication via SSH between the Oracle Server and the vFiler. The account should be created with the a role that has the following capabilities:

  cli-snapvault,cli-snap*,cli-ndmpcopy*,login-ssh,cli-df**
- There are limitations on the maximum number of volumes allowed per vFiler when doing a Data Motion for vFilers migration. These limits are as follows:
  - FAS3xxx – maximum of 8 flexvols per vFiler.
  - FAS6xxx – maximum of 20 flexvols per vFiler.

2.2 Volume and Qtree configuration

Each Oracle deployment will typically have 3 flexvols configured on a NetApp primary storage system, as show in the following table. The following table gives a high level description of the volumes and qtrees used in a typical LION configuration for TRP.

Storage Volume	Storage Qtree	Description	Snap Reserve	SnapVault?
<bu>_<app>_n01ora1_nosnap	n01oracluster1	RAC OCR/Vote	0%	No
	n01oradata1	Control, Temp, Online and Standby Redo logs		
	n01oraflash1	Flashback Recovery area if in use		
	n01oraadmin1	Admin area, does not require backup - ADR		
	n01oraggs	GoldenGate software and configuration if used		
<bu>_<app>_s01ora1_snap	s01oradata1	Data and control files	20% or 50%	Yes
<bu>_<app>_s01oraadm1_snap	s01oraadmin1	Admin area, requires backup – spfile	20% or 50%	Yes

When provisioning volumes and qtrees for an Oracle configuration, the following should be noted:

- Sizing requirements will be provided by DBA team, based on their requirements and their standard sizing templates for LION configurations.
- All qtrees will have tree quotas enabled, with the tree quota limit as determined by the sizing input given by the DBA team.
- A Snap Reserve of 20% is used for thick provisioned volumes.
- A Snap Reserve of 50% is used for thin provisioned volumes.

The follow configuration settings apply for each "_snap" volume, in order to implement snap autodelete:

- Set the "try_first" volume option to "snap_delete".
- Enable snap autodelete and set the snap autodelete trigger to "snap_reserve".

Note thin provisioning should only be used when provisioning storage on an aggregate for which all other volumes are also thin provisioned. To thin provision a volume, simiply set the volume guarantee to "none".

2.3 Snap Reserve sizing and monitoring

For volumes on which snapshots will be taken, the Snap Reserve configuration listed in section 2.2 is 20% for thick provisioned volumes and 50% for thin provisioned volumes. The reasons for this are as follows:

- Thick provisioned volumes will pre-allocate space from the aggregate for the volumes, including the Snap Reserve area of a volume. A 20% snap reserve has shown to be adequate for most applications, without wasting space. This can be increased in the event that an application is consistently over-utilizing their Snap Reserve area and having too many snapshots autodeleted.
- Using a Snap Reserve of 50% for thin provisioned volumes means that we have an equal amount of space for data and the snapshots of the data. This guarantees that we can always have 1 snapshot of a volume without the possibility of running out of space due to snapshot consumption. The use of thin provisioning means that we don't pre-allocate this additional Snap Reserve space from the aggregate.
- Snap autodelete is used to keep the Snap Reserve area from overflowing into the active filesystem portion of the volume. In the TRP environment, snap autodelete is configured to trigger at 80% of Snap Reserve utilization.

In a worst case scenario, retaining a single snapshot could result in Snap Reserve utilization of above 80% (but less than 100%). In order to manage these scenarios, an Operations Manager alert for the snapshot-full condition should be configured to page the TRP storage team when the Snap Reserve is at 80% utilization for at least 30 minutes (this time delay gives snap autodelete time to bring the utilization back down under 80% if possible). In situations where a single snapshot results in Snap Reserve utilization of over 80%, any newly created Oracle backup snapshots will be immediately removed by snap autodelete. Because of this, the proper response to a page for this condition is to temporarily (or permanently, for repeat offenders) grow the overall size of the volume.

2.4 Snapshot backup configuration on primary storage

Snapshot creation for volumes containing Oracle datafiles on primary storage will be driven completely by scripts running on the Oracle server. These scripts will use SSH to run a "*snapvault snap create*" command while the database is in hot backup mode. In order to support the creation of snapvault snapshots by an external script, the proper configuration of the snapvault schedule on primary storage should be as follows:

vfiler run <Primary-vFiler> snapvault snap sched <volname> <snapname> 7@-

The s01oraadm1_snap for each database volume does not get snapshots taken as part of the backup process, and instead will have snapshots created on primary storage as part of a schedule. For example:

vfiler run <Primary-vFiler> snapvault snap sched <volname> <snapname> 7@Sun-Sat@20

Note that snapvault schedules should only be configured on the "_snap" volumes. All volumes, including the "_snap" volumes, should have their standard snapshot schedule disabled via the following command:

vfiler run <Primary_vFiler> snap sched <volname> 0

2.5 SnapVault backup configuration on secondary storage

The SnapVault relationship for volumes with Oracle databases on them will be initialized in a similar manner as any other SnapVault relationship. Relationships are configured at the volume level, as shown in the following command example:

vfiler run <Secondary_vFiler> snapvault start –S <primary_vFiler>:/vol/<volname> /vol/<volname>/1

Replication schedules are configured on the SnapVault secondary vFiler, typically with a 14 day retention period. SnapVault updates should be staggered on the secondary vFiler, so that all backups are not started at exactly the same time. An example is provided below, in which 14 snapshots are kept, with replication updates starting at 2 am, and run every night of the week (Sun-Sat):

vfiler run <Secondary_vFiler> snapvault snap sched –x <volname> <snapname> 14@Sun-Sat@2

Note that the timing of the SnapVault replication updates is configured on the SnapVault secondary, and therefore SnapVault snapshot creation via Oracle backup scripts should be configured to take place on primary storage prior to the replication update start time configured on the SnapVault secondary.

2.6 Aggregate over-provisioning and capacity alerting

As of the writing of this document, the plan is to over-provision aggregates by a factor of 2 to 1. For example, an aggregate with 10TB of phsyiscal space would have 20TB of thin provisioned volumes allocated on it, with 10TB assigned to the active filesystems for data, and 10TB assigned to volume Snap Reserves for holding snapshot copies of data.

The use of thin provisioned volumes with a 50% Snap Reserve is useful for avoiding out of space conditions on volumes that are using snapshots. This configuration also allows us to over-provision the aggregate so that we can drive up overall storage utilization. This configuration, along with a 2 to 1 over-provisioning scheme, also allows us to guarantee that we will always have enough space in an aggregate to hold all data, even in a worst case scenario of having to delete all volume snapshots in order to accomplish this. Note that we do not expect to ever hit this worst case scenario, as action should be taken before we get to this point.

Since we are using thin provisioned volumes, and managing space at the aggregate level, we must use different capacity alerting metrics than fully provisioned volumes would use. NetApp Operations Manager should be configured to set the Aggregate Full Threshold to 70% and the Aggregate Nearly Full Threshold to 60% for aggregates used with thin provisioned volumes. These settings can be overridden from the global default values by navigating to **Global -> Member Details -> Aggregates -> <aggr name> -> Aggregate Tools Edit Settings** in the Operations Manager 4.0.x web interface or by using the "*dfm aggr set <controller>:<aggr> aggrNearlyFullThreshold=<value>*" and "*dfm aggr set <controller>:<aggr> aggrFullThreshold=<value>*" commands from the Operations Manager 4.0.x and 5.0.x CLI. Operations Manager alarms should then be configured to notify appropriate personnel when these thresholds are breached.

3 Oracle server configurations related to NetApp storage

3.1 Database file layout

Each Oracle database will use a number of NFS mountpoints to store their data. The following table gives a high level description of the standard NFS mountpoint layout and how it maps to the volume layout on the NetApp storage system. A more detailed document on the sizing and configuration for these mountpoints is available from the TRP DBA team.

Mount Description	Mountpoint	Storage Volume	Storage Qtree
RAC OCR/Vote	/n01/oracluster1	<bu>_<app>_n01ora1_nosnap	n01oracluster1
Control, Temp, Online and Standby Redo logs	/n01/oradata1		n01oradata1
Flashback Recovery area if in use	/n01/oraflash1		n01oraflash1
Admin area, does not require backup - ADR	/n01/oraadmin1		n01oraadmin1
GoldenGate software and configuration if used	/n01/oraggs		n01oraggs
Data and control files	/s01/oradata1	<bu>_<app>_s01ora1_snap	s01oradata1
Admin area, requires backup – spfile	/s01/oraadmin1	<bu>_<app>_s01oraadm1_snap	s01oraadmin1

In the event that additional volumes are required when using multiple instances of Oracle on the same Oracle database servers, a second set of mountpoints with a slightly modified naming convention (n02 and s02 instead of n01 and s01) is configured as follows:

Mount Description	Mountpoint	Storage Volume	Storage Qtree
Control, Temp, Online and Standby Redo logs	/n02/oradata1	<bu>_<app>_n02ora1_nosnap	n02oradata1
Flashback Recovery area if in use	/n02/oraflash1		n02oraflash1
Admin area, does not required backup - ADR	/n02/oraadmin1		n02oraadmin1
Data and control files	/s02/oradata1	<bu>_<app>_s02ora1_snap	s02oradata1
Admin area, requires backup – spfile	/s02/oraadmin1	<bu>_<app>_s02oraadm1_snap	s02oraadmin1

3.2 Oracle archive log storage on non-NetApp storage

The database file layout guidelines given above do not address the location of Oracle archive logs. Oracle archive logs are typically written to a NFS mountpoint served by a HNAS NFS server, which allows the logs to be transparently migrated to Sun storage for compression. In the event that the HNAS storage is unavailable due to a planned or unplanned outage, Oracle switches to writing the archive logs to a NetApp NFS mountpoint. The NetApp storage is configured as a single thin provisioned volume per datacenter module named infra_nosnap, with each instance of Oracle having it's own qtree for archive log storage. The following table describes the NFS mountpoints used by the Oracle server for achive log storage:

Mount Description	Mountpoint	Storage System	Storage Volume	Storage Qtree
Primary archive area	/n01/oraarch1	HNAS		
Alternate archive area	/n01/oraarch2	NetApp	infra_nosnap	<bu>_<app>_n01oraarch2

3.3 Oracle archive log storage on a NetApp vFiler with compression

As of Data ONTAP 8.1, NetApp also supports data compression. In environments where there is no HNAS and there are NetApp shared primary storage systems running ONTAP 8.1 or higher, NetApp will be used to store and compress Oracle archive log backups. The NetApp storage is configured as a single thin provisioned volume per datacenter module named infra_nosnap, with each instance of Oracle having it's own qtree for archive log storage. The following table describes the NFS mountpoints used by the Oracle server for achive log storage in this configuration:

Mount Description	Mountpoint	Storage System	Storage Volume	Storage Qtree
Primary archive area	/n01/oraarch1	NetApp	infra_nosnap	<bu>_<app>_n01oraarch1

In these environments, the following standards should be followed:

- There will be a shared vFiler in each datacenter module designated for Oracle archive log file backups.
- The vFiler will be stored on a low tier shared primary storage system that is part of a HA pair.
- Each Oracle instance will get it's own qtree on a shared volume in the vFiler.
- The volume(s) storing the Oracle archive log backups will use the background compression method, not inline compression.
- The qtrees will be NFS mounted directly on the Oracle servers in order to perform log backups.
- The log backups will not be replicated via SnapVault or SnapMirror, and SnapShots will not be used on the volume(s) storing Oracle archive log backups.
- A script on the NetApp Operations Manager (DFM) server in each datacenter module will be run on a daily basis via cron in order to prune old archive log backups.

The following commands demonstrate how to configure background compression on a given volume of a vfiler. In this example, we set compression to run at 1am every night:

vfiler run vfilername sis on /vol/volname
vfiler run vfilername sis config –C true /vol/volname
vfiler run vfilername sis config -s sun-sat@1 /vol/volname
vfiler run vfilername sis config

In order to see the compression savings of a given volume, the command "*df –S volname*" can be used.

3.4 NFS mount options

NetApp and Oracle have well defined and jointly supported NFS mount options for Oracle single instance and Oracle RAC environments. The following KB article on the NetApp support site (login required) documents these best practices:

https://kb.netapp.com/support/index?page=content&id=3010189

Note that for Oracle RAC configurations, the "*actimeo=0*" NFS mount option is listed as being required for all Oracle mountpoints. While this is the published best practice by both NetApp and Oracle, we have seen within the TRP environment that a large amount of NFS GETATTR and LOOKUP traffic is occasionally generated within the ADR location (/n01/oraadmin1). After extensive testing, and the relocation of the spfile to the /s01/oraadmin1 mountpoint, TRP has determined that the *actimeo=0* NFS mount option can be removed from the /n01/oraadmin1 mountpoint. This configuration is only in place for LIONv2 configurations.

This has shown a dramatic reduction in NFS operations on that mountpoint in some instances, and therefore this reduces the load significantly on shared storage systems hosting multiple Oracle databases. Any configuration not following the exact LIONv2 standard for the Oracle file layout should adhere to the recommendations in the KB article and use *actime=0* on all mountpoints.

3.5 Linux kernel TCP tuning on the Oracle server

A number of TCP tunings are used in the Linux kernel of the Oracle server, in order to ensure the best possible performance. The following is a subset of the settings used in /etc/sysctl.conf on the Oracle server, as they are defined for the LIONv2 standard. The TRP DBA and Platform management teams can provide the full list of settings based on a specific implementation, and whenever possible the TRP standard and tested configurations should be used. This section is included in this document only to remind the reader that these types of Linux kernel settings are important in order to ensure the best possible performance when running Oracle over NFS.

```
# 10GigE Standard Network Tunables for TRP
net.core.rmem_default = 16777216
net.core.rmem_max = 16777216
net.core.wmem_default = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_wmem = 16777216 16777216 16777216
net.ipv4.tcp_rmem = 16777216 16777216 16777216
net.core.netdev_max_backlog = 300000

# Increase NFS over TCP parallelism
sunrpc.tcp_slot_table_entries = 128
```

3.6 Oracle init.ora parameters

The Oracle init.ora configuration file contains a number of parameters in it, and these should be configured per the current TRP standards for a given implementation. One parameter in particular is important to have properly configured when running Oracle over NFS, which we will call out in this document, is the *filesystemio _options* setting. For any recent version of Linux (RHEL 5 and above, SLES 11 and above) this should be configured as "*setall*" in order to enable both Direct IO and Async IO. Older versions of Linux that don't support Async IO with NFS should have this option configured simply as "*directio*".

4 Oracle backup and recovery

The Oracle backup and recovery procedures documented here are focused on the use of NetApp snapshots for Oracle datafile backup and recovery. As previously documented, Oracle archive logs are typically kept on non-NetApp storage within the TRP environment, and as such this document does not cover the methods used for maintaining archive logs and making sure they are available when required as part of a restore scenario.

4.1 Oracle backup

Backups are initiated by a shell script, scheduled in cron, from the Oracle server. This shell script, amongst other tasks, will put the Oracle tablespaces in hot backup mode and then run a "*snapvault snap create*" command via SSH on the vFiler hosting it's "snap" volumes. This requires SSH publickey authentication to be properly configured between the Oracle server and the vFiler hosting it's volumes. Each DBA group within TRP manages it's own shell script for doing Oracle backups, but the method by which the snapshots are taken is identical in all scripts.

4.2 Oracle restore scenarios

4.2.1 Restoring a single datafile from primary snapshot backups

There are two methods by which restores of single files can be performed from snapshots on primary storage. The first option is to use a simple "*cp*" command from the Oracle server to copy data from a snapshot directory back to the active filesystem. To see the snapshots available on primary storage, "*cd"* into the directory where the restore is required. Then use "*cd .snapshot*" and "*ls –lu*" to see the snapshots that are available. Once the snapshot path to the file to be restored has been identified, it can be copied back to the fully qualified path to the active filesystem. For example:

cp /s01/oradata1/.snapshot/<dir>/<snapshot_name>/<filename> /s01/oradata1/<dir>/<filename>

The second option is to use the single file snap restore capabilities on the NetApp storage system to perform the restore. You can use the "*snap list*" command to determine which snapshot you want to use for the restore, or you can use the "*ls –lu*" command as described above to view the available snapshots. The single file snap restore and snapshot related commands can be run on the NetApp storage controller via a SSH connection from the Oracle server, and multiple files can be restored in parallel by issuing multiple commands via SSH. For example:

ssh <Primary_vFiler> snap list [<volume_name>]
ssh <Primary_vFiler> snap restore –f –t file –s <snapshot_name> </vol/volname/path/to/filename>

In general, the use of the single file snap restore command should be slightly faster than a simple client based "*cp*" command to restore files from snapshots. That said, the client based copy operation is typically easier to use, as all Oracle administrators are experienced with simple copy operations. If the single file snap restore command is used, the progress can be monitored by periodically checking the size of the file being restored from the Oracle server. When the file size stops changing, the single file snap restore operation is complete.

4.2.2 Restoring all datafiles in a volume from primary snapshot backups

In some use cases, it is necessary to restore all files in a given volume back to a previous point in time. When this is required, NetApp storage controllers have the ability to perform a volume snap restore operation. While this is similar in concept to the single file snaprestore, it has some very important differences:

- Using a volume snap restore can result in snapshot backups being removed from the volume. If a volume snap restore is used to restore to a snapshot that is older than the most recently created snapshot, then all of the snapshots that are newer than the snapshot being used for the restore will be removed. This is because a volume snap restore reverts the entire volume, including snapshots of the volume, back to the state as it existed when the snapshot being used for the restore was created.
- A volume snap restore is much faster than a single file snap restore. While the length of time it takes for a single file snap restore to complete is directly related to the size of the file being restored, a volume snap restore is always very fast, completing in a minute or less. This makes the volume snap restore an ideal candidate for restoring volumes containing large amounts of data, as long as the caveat listed previously is well understood.

An example of performing a volume snap restore is provided here:

ssh <Primary_vFiler> snap restore –f –t vol –s <snapshot_name> <volume_name>

After a volume snap restore operation, the storage support team should be notified, as the SnapVault backup relationship to secondary storage may need to be re-initialized.

4.2.3 Restore from Secondary snapshot backups

In the event that the snapshot backups required for the restore are no longer available on primary storage, the restore must be performed from secondary storage. Again, there are two ways in which this restore can be accomplished.

The first option is that the SnapVault secondary volume can be exported to the Oracle server, and it can be mounted via NFS on the server where the restore is required. This will require that the Oracle DBA make a request to the storage support team for the NFS export of the SnapVault secondary volume to be configured. The Oracle DBA can then use a simple copy operation, similar to what was described in section 4.2.1, to copy data from a snapshot on secondary storage back to the active filesystem on primary storage. The advantage to using this method is that the Oracle DBA has all the secondary snapshots at their disposal for use in the restore process. This can be useful if the DBA is not sure which version of a given file will be required, as they can quickly change to using a different snapshot for their restore process.

The second option is that the Oracle DBA can request that the storage support team directly copy a certain set of files or directories from secondary storage back to primary storage. Since TRP is using volume based SnapVault, this restore operation will typically be performed using the *ndmpcopy* command. Note that performing *ndmpcopy* operations between vFilers in non-default ipspaces is not supported prior to ONTAP version 8.1. The *ndmpcopy* operation can still be performed over the management network from storage controller to storage controller, even for volumes owned by vFilers, until both the source and destination are running 8.1 or higher. Once the source and destination are running 8.1 or higher, the *ndmpcopy* operation can be performed from a vFiler context, and make use of the data networks configured for vFilers.

4.4.4 Partial tablespace restores from snapshot backups

In some cases, only a small portion of a tablespace needs to be restored from a snapshot backup copy. Examples of this use case would be having a single table, or a single row in a table, that needs to be restored. In these cases, a snapshot of the SnapVault secondary volume can be cloned on the secondary controller. This clone is essentially a writable copy of a snapshot, but it only requires storage for blocks that are changed, and therefore it can be created very quickly (typically 10 seconds or less) regardless of the dataset size. This clone can then be exported via NFS to an Oracle restore server that would be used to bring up a version of the database that is used for restore purposes. While this method is not widely used within TRP, the concept is documented here so that it can be considered for future restore scenarios.

Appendix A: Resources and Whitepapers

- NetApp Best Practice Guidelines for Oracle Database 11g:
  http://media.netapp.com/documents/tr-3633.pdf

- NetApp for Oracle Database
  http://media.netapp.com/documents/nva-0002.pdf

- Oracle Archived Logs Management Best Practices
  http://media.netapp.com/documents/tr-3901.pdf

- Oracle Database 10g Release 2 RAC:
  http://media.netapp.com/documents/tr-3555.pdf

- NFS mount options for Oracle databases:
  https://kb.netapp.com/support/index?page=content&id=3010189

- Oracle Clusterware/RAC CSS Timeout Settings with NetApp Clustered Failover:
  https://kb.netapp.com/support/index?page=content&id=3011276