Oracle Configurations

Enterprise applications

NetApp
December 17, 2024

Table of Contents

Oracle Configurations

Overview

The use of SnapMirror active sync does not necessarily add to or change any best practices for operating a database.

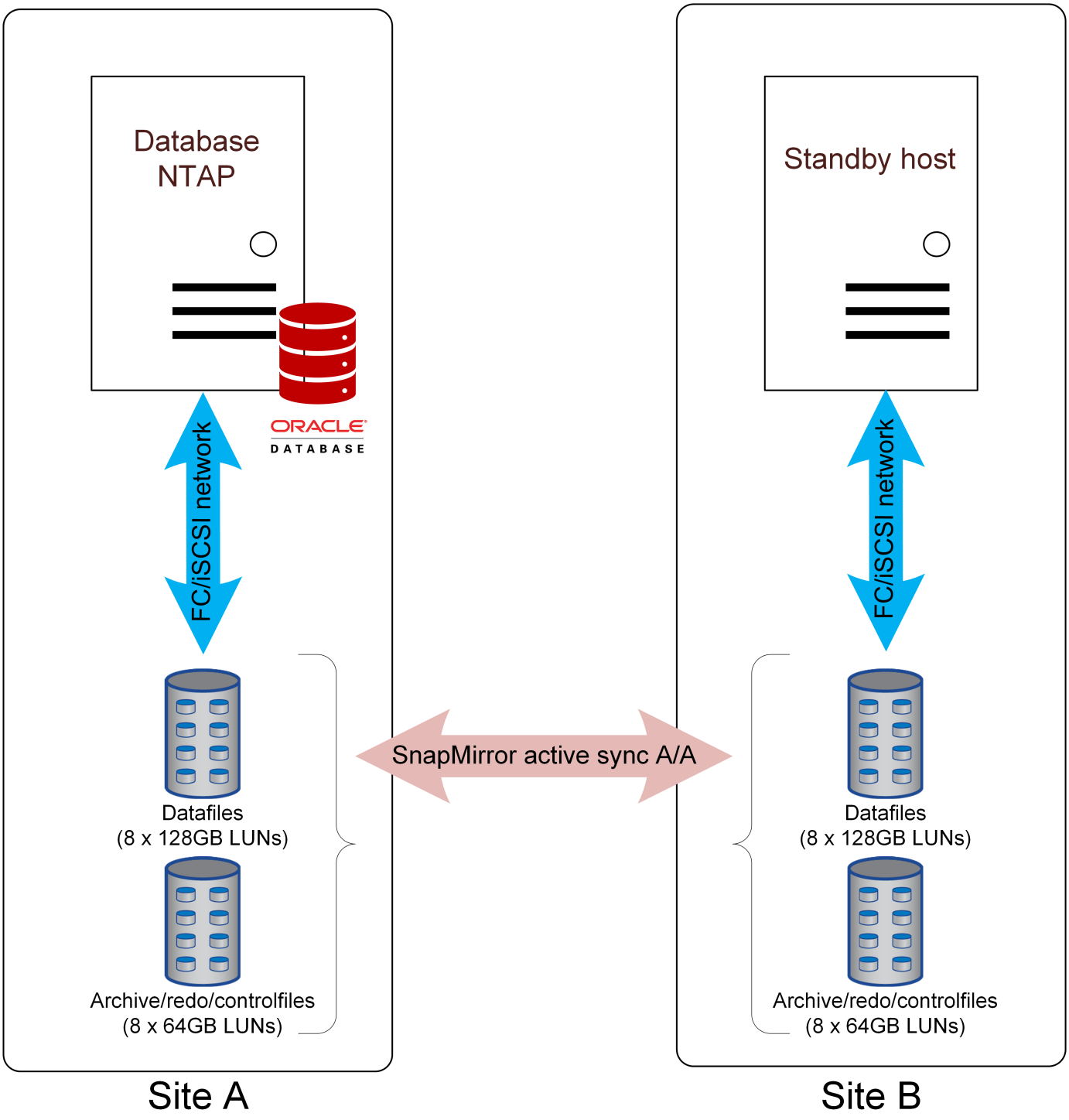
The best architecture depends on the business requirements. For example, if the goal is to have RPO=0 protection against data loss, but the RTO is relaxed, then using Oracle Single Instance databases and replicating the LUNs with SM-as might be sufficient as well as less expensive from an Oracle licensing standpoing. Failure of the remote site would not interrupt operations, and loss of the primary site would result in LUNs at the surviving site that are online and ready to be used.

If the RTO was more strict, basic active-passive automation through scripts or clusterware such as Pacemaker or Ansible would improve failover time. For example, VMware HA could be configured to detect VM failure on the primary site and active the VM on the remote site.

Finally, for extremely rapid failover, Oracle RAC could be deployed across sites. The RTO would essentially be zero because the database would be online and available on both sites at all times.

Oracle Single-Instance

The examples explained below show some of the many options for to deploying Oracle Single Instance databases with SnapMirror active sync replication.



Failover with a preconfigured OS

SnapMirror active sync delivers a synchronous copy of the data at the disaster recovery site, but making that data available requires an operating system and the associated applications. Basic automation can dramatically improve the failover time of the overall environment. Clusterware products such as Pacemaker are often used to create a cluster across the sites, and in many cases the failover process can be driven with simple scripts.

If the primary nodes are lost, the clusterware (or scripts) will bring the databases online at the alternate site. One option is to create standby servers that are preconfigured for the SAN resources that make up the database. If the primary site fails, the clusterware or scripted alternative performs a sequence of actions similar

to the following:

1. Detect failure of primary site
2. Perform discovery of FC or iSCSI LUNs
3. Mounting file systems and/or mounting ASM disk groups
4. Starting the database

The primary requirement of this approach is a running OS in place on the remote site. It must be preconfigured with Oracle binaries, which also means that tasks such as Oracle patching must be performed on the primary and standby site. Alternatively, the Oracle binaries can be mirrored to the remote site and mounted if a disaster is declared.

The actual activation procedure is simple. Commands such as LUN discovery require just a few commands per FC port. File system mounting is nothing more than a `mount` command, and both databases and ASM can be started and stopped at the CLI with a single command.

Failover with a virtualized OS

Failover of database environments can be extended to include the operating system itself. In theory, this failover can be done with boot LUNs, but most often it is done with a virtualized OS. The procedure is similar to the following steps:

1. Detect failure of primary site
2. Mounting the datastores hosting the database server virtual machines
3. Starting the virtual machines
4. Starting databases manually or configuring the virtual machines to automatically start the databases.

For example, an ESX cluster could span sites. In the event of disaster, the virtual machines can be brought online at the disaster recovery site after the switchover.

Storage failure protection

The diagram above shows the use of nonuniform access, where the SAN is not stretched across sites. This may be simpler to configure, and in some cases may be the only option given the current SAN capabilities, but it also means that failure of the primary storage system would cause a database outage until the application was failed over.

For additional resilience, the solution could be deployed with uniform access. This would allow the applications to continue operating using the paths advertized from the opposite site.

Oracle Extended RAC

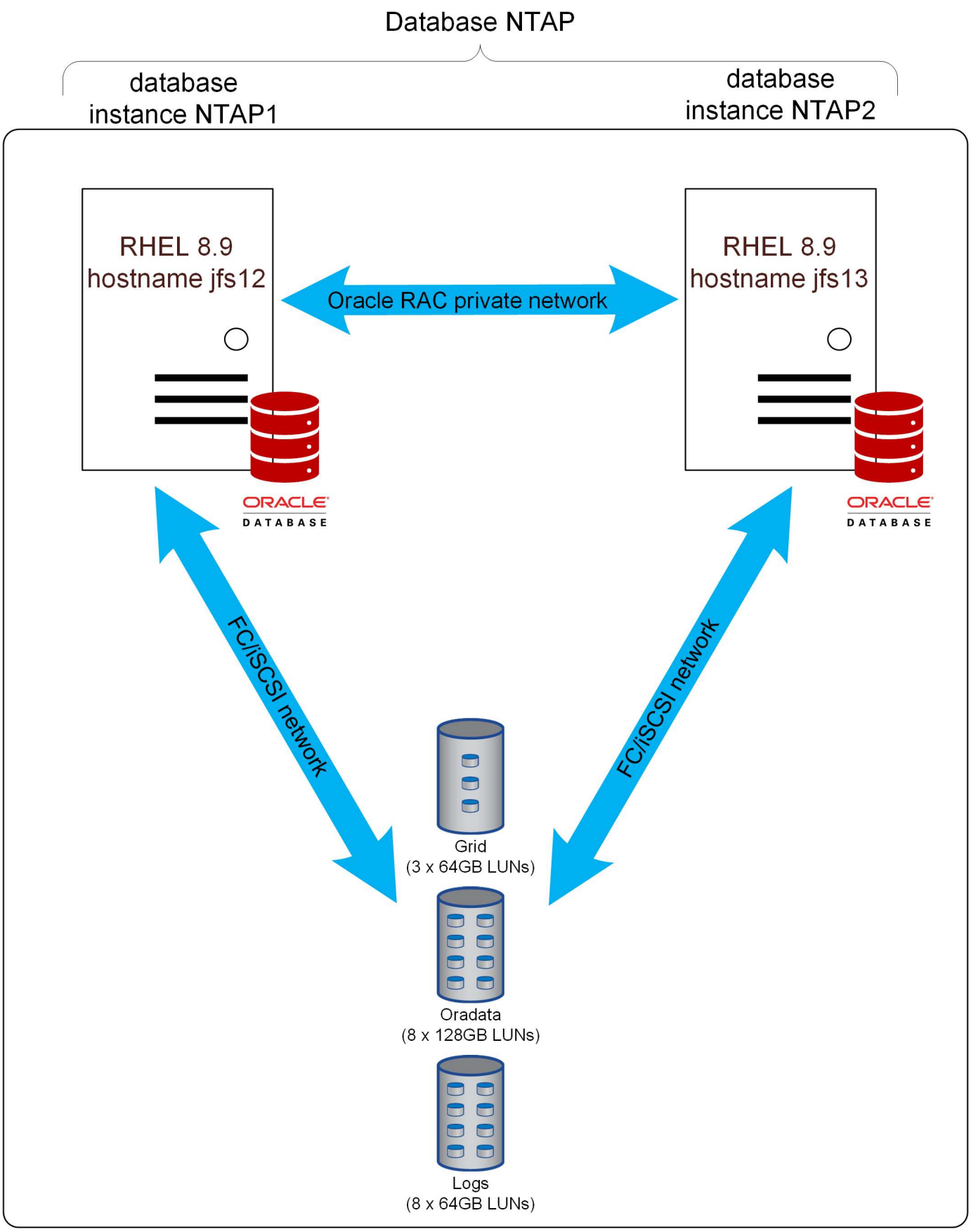
Many customers optimize their RTO by stretching an Oracle RAC cluster across sites, yielding a fully active-active configuration. The overall design becomes more complicated because it must include quorum management of Oracle RAC.

Traditional extended RAC clustered relied on ASM mirroring to provide data protection. This approach works, but it also requires a lot of manual configuration steps and imposes overhead on the network infrastructure. In contrast, allowing SnapMirror active sync to take responsibility for data replication dramatically simplifies the solution. Operations such as synchronization, resynchronization after disruptions, failovers, and quorum

management are easier, plus the SAN does not need to be distributed across sites which simplifies SAN design and management.

Replication

They key to understanding RAC functionality on SnapMirror active sync is to view storage as a single set of LUNs which are hosted on mirrored storage. For example:



There is no primary copy or mirror copy. Logically, there is only a single copy of each LUN, and that LUN is available on SAN paths that are located on two different storage systems. From a host point of view, there are no storage failovers; instead there are path changes. Various failure events might lead to loss of certain paths

to the LUN while other paths remain online. SnapMirror active sync ensures the same data is available across all operational paths.

Storage configuration

In this example configuration, the ASM disks are configured the same as they would be in any single-site RAC configuration on enterprise storage. Since the storage system provides data protection, ASM external redundancy would be used.

Uniform vs nonuninform access

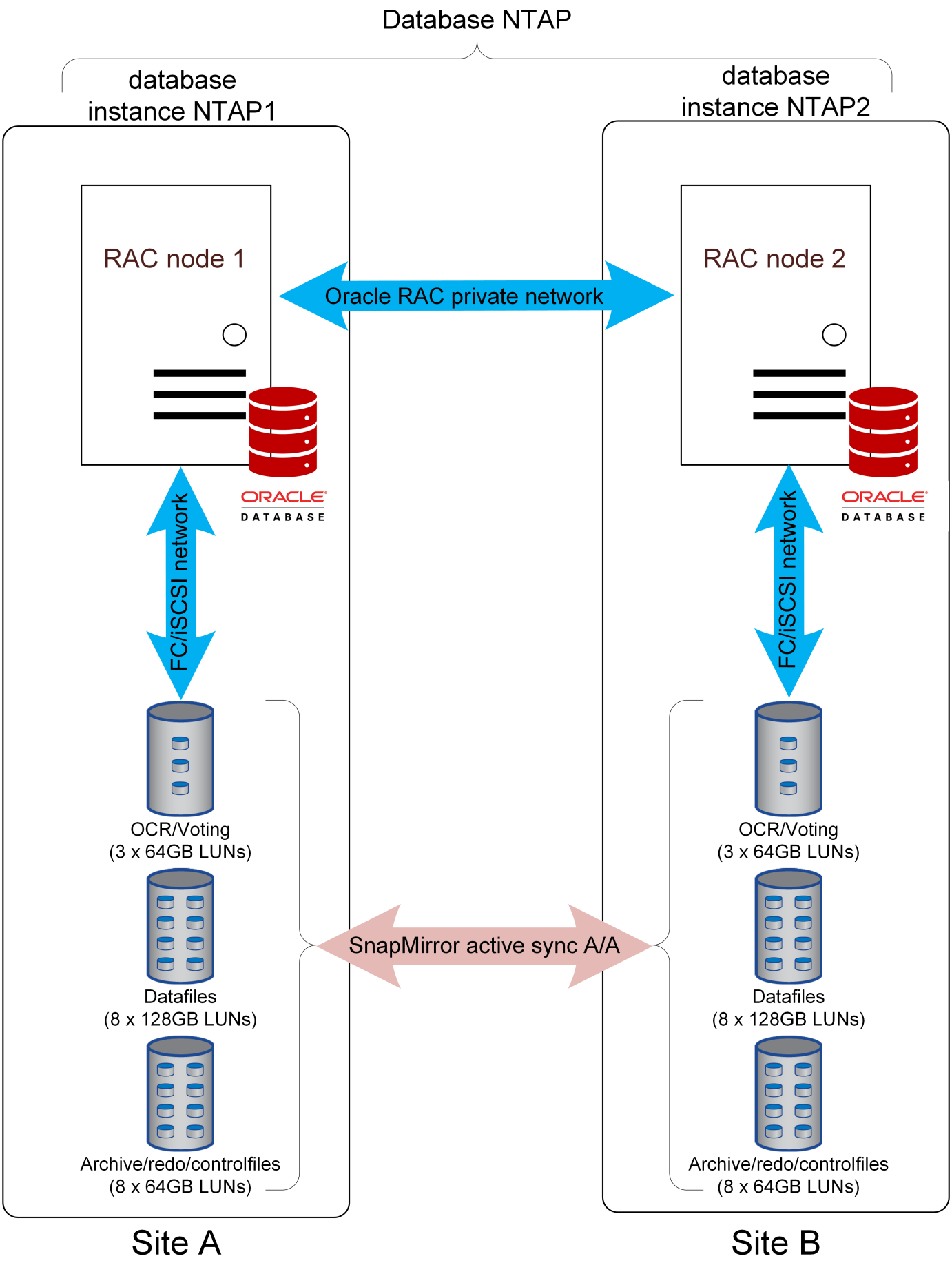
The most important consideration with Oracle RAC on SnapMirror active sync is whether to use uniform or nonuniform access.

Uniform access means each host can see paths on both clusters. Nonuniform access means hosts can only see paths to the local cluster.

Neither option is specifically recommended or discouraged. Some customers have dark fibre readily available to connect sites, others either do not have such connectivity or their SAN infrastructure doesn’t support a long-distance ISL.

Nonuniform access

Nonuniform access is simpler to configure from a SAN perspective.


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
SnapMirror active sync A/A
OCR/Voting (3 x 64GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B

The primary downside of the nonuniform access approach is that loss of site-to-site ONTAP connectivity or loss of a storage system will result in loss of the database instances at one site. This obviously is not desirable, but it may be an acceptable risk in exchange for a simpler SAN configuration.

Uniform access

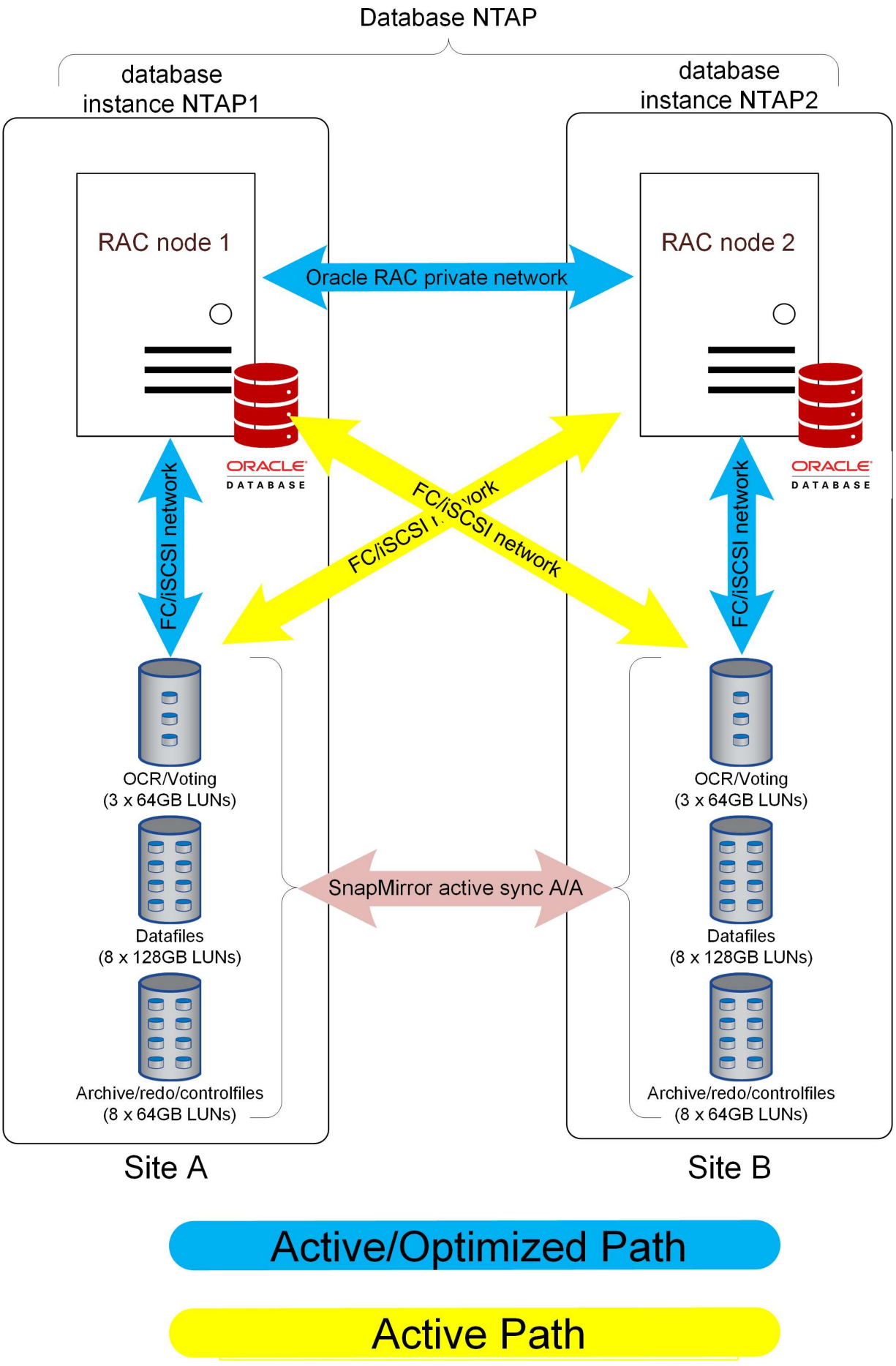
Uniform access requires extending the SAN across sites. The primary benefit is that loss of a storage system will not result in loss of a database instance. Instead, it would result in a multipathing change in which paths are currently in use.

There are several ways to configure nonuniform access.

> In the diagrams below, there are also active but nonoptimized paths present that would be used during simple controller failures, but those paths are not shown in the interest of simplifying the diagrams.

AFF with proximity settings

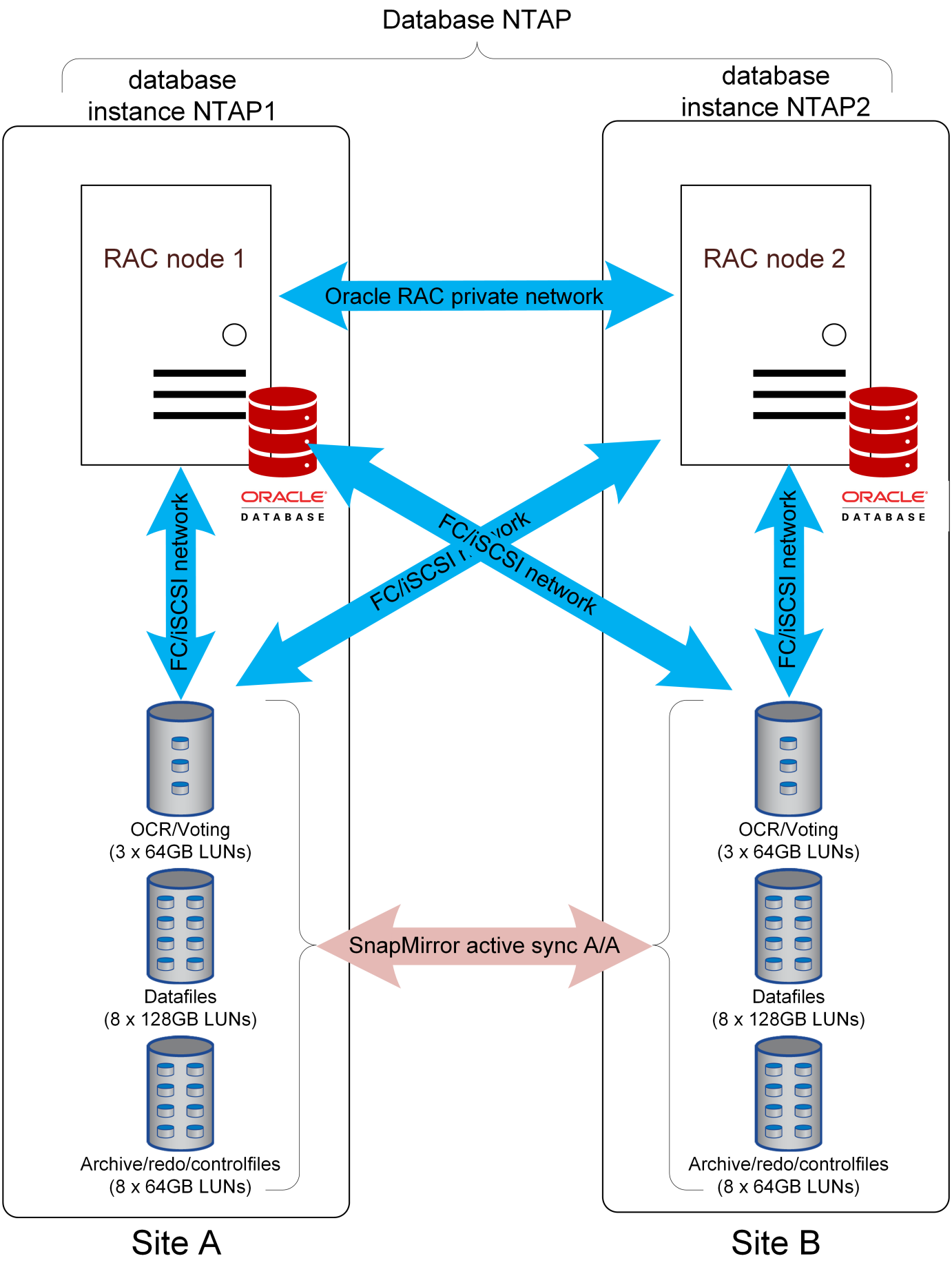
If there is significant latency between sites, then AFF systems can be configured with host proximity settings. This allows each storage system to be aware of which hosts are local and which are remote and assign path priorities appropriately.


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
OCR/Voting (3 x 64GB LUNs)
SnapMirror active sync A/A
Datafiles (8 x 128GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B
Active/Optimized Path
Active Path

In normal operation, each Oracle instance would preferentially use the local active/optimized paths. The result is that all reads would be serviced by the local copy of the blocks. This yields the lowest possible latency. Write IO is similarly sent down paths to the local controller. The IO must still be replicated before being acknowledged and therefor would still incur the additional latency of crossing the site-to-site network, but this cannot be avoided in a synchronous replication solution.

ASA / AFF without proximity settings

If there is no significant latency between sites, then AFF systems can be configured without host proximity settings, or ASA can be used.


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
OCR/Voting (3 x 64GB LUNs)
SnapMirror active sync A/A
Datafiles (8 x 128GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B

Each host will be able to use all operational paths on both storage systems. This potentially improves performance significantly by allowing each host to draw upon the performance potential of two clusters, not just one.

With ASA, not only would all paths to both clusters be considered active and optimized, but the paths on partner controllers would also be active. The result would be all-active SAN paths on the entire cluster, all the time.

ASA systems may also be used in a nonuniform access configuration. Since no cross-site paths exist, there would be no impact on performance resulting from IO crossing the ISL.

RAC tiebreaker

While extended RAC using SnapMirror active sync is a symmetric architecture with respect to IO, there is one exception that is connected to split-brain management.

What happens if the replication link is lost and neither site has quorum? What should happen? This question applies to both the Oracle RAC and the ONTAP behavior. If changes cannot be replicated across sites, and you want to resume operations, one of the sites will have to survive and the other site will have to become unavailable.

The ONTAP Mediator addresses this requirement at the ONTAP layer. There are multiple options for RAC tiebreaking.

Oracle tiebreakers

The best method to manage split-brain Oracle RAC risks is to use an odd number of RAC nodes, preferably by use of a 3rd site tiebreaker. If a 3rd site is unavailable, the tiebreaker instance could be placed on one site of the two sites, effectively designating it a preferred survivor site.

Oracle and css_critical

With an even number of nodes, the default Oracle RAC behavior is that one of the nodes in the cluster will be deemed more important than the other nodes. The site with that higher priority node will survive site isolation while the nodes on the other site will evict. The prioritization is based on multiple factors, but you can also control this behavior using the `css_critical` setting.

In the example architecture, the hostnames for the RAC nodes are jfs12 and jfs13. The current settings for `css_critical` are as follows:

```
[root@jfs12 ~]# /grid/bin/crsctl get server css_critical
CRS-5092: Current value of the server attribute CSS_CRITICAL is no.

[root@jfs13 trace]# /grid/bin/crsctl get server css_critical
CRS-5092: Current value of the server attribute CSS_CRITICAL is no.
```

If you want the site with jfs12 to be the preferred site, change this value to yes on a site A node and restart services.

```
[root@jfs12 ~]# /grid/bin/crsctl set server css_critical yes
CRS-4416: Server attribute 'CSS_CRITICAL' successfully changed. Restart
Oracle High Availability Services for new value to take effect.

[root@jfs12 ~]# /grid/bin/crsctl stop crs
CRS-2791: Starting shutdown of Oracle High Availability Services-managed
resources on 'jfs12'
CRS-2673: Attempting to stop 'ora.crsd' on 'jfs12'
CRS-2790: Starting shutdown of Cluster Ready Services-managed resources on
server 'jfs12'
CRS-2673: Attempting to stop 'ora.ntap.ntappdb1.pdb' on 'jfs12'
…
CRS-2673: Attempting to stop 'ora.gipcd' on 'jfs12'
CRS-2677: Stop of 'ora.gipcd' on 'jfs12' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed resources
on 'jfs12' has completed
CRS-4133: Oracle High Availability Services has been stopped.

[root@jfs12 ~]# /grid/bin/crsctl start crs
CRS-4123: Oracle High Availability Services has been started.
```

Copyright information

Trademark information