Technical Report

Best Practices for Oracle Databases on NetApp Storage

Jeffrey Steiner, NetApp
March 2014 | TR-3633

Important

Consult the Interoperability Matrix Tool (IMT) to determine whether the environment, configurations, and versions specified in this report support your environment.

TABLE OF CONTENTS

LIST OF TABLES

1 Introduction

There are very few clear best practices for configuring an Oracle® Database on NetApp® storage due to a wide variety of user requirements, including database sizes, performance requirements, and data protection needs. Known deployments on NetApp storage include everything from a virtualized environment of approximately 6,000 databases running under VMware® ESX® to a single-instance data warehouse currently at 810TB and growing.

This document addresses the small number of true requirements for deploying an Oracle Database on NetApp storage. In addition, it reviews many design considerations that must be addressed by the architect of the Oracle storage solution based on that architect's specific business requirements. In this report, the topics are organized by general considerations for all environments, followed by general requirements specific to Network File System (NFS) and storage area network (SANs). Finally, specific recommendations are included for various operating systems in alphabetical order.

2 Recommendations for General Data ONTAP Configuration

A complete explanation of NetApp Data ONTAP® configuration is beyond the scope of this document. One best practice for an environment with 2,000 virtualized databases might be inappropriate for a configuration of three very large ERP databases; even small changes in the data protection and recovery requirements can significantly affect storage design. Some basic details are reviewed in this section, but for comprehensive assistance with design, contact NetApp or a NetApp reseller for further assistance.

2.1 Backups Based on Snapshot

The most important consideration in file system layout is the plan for leveraging Snapshot™ backups. There are two primary approaches:

- Crash-consistent backups
- Hot backups protected by Snapshot

A crash-consistent backup of a database requires the entire database structure, including data files, redo logs, and control files, to be captured at a single point in time. If the database is stored in a single FlexVol® volume (flexible volume), the process is simple: a Snapshot copy can be created at any time. If a database spans volumes, a consistency group (CG) Snapshot copy must be created. Several options exist for creating CG Snapshot copies, including Snap Creator™ software, SnapManager® for Oracle, SnapDrive® for UNIX®, and user-maintained scripts.

Crash-consistent Snapshot backups are primarily used when point-of-the-backup recovery is sufficient. Archive logs can be applied under some circumstances, but when more granular point-in-time recovery is required, a hot backup is preferable.

The basic procedure for a hot backup based on Snapshot is as follows:

1. Place the database in hot backup mode.
2. Create a Snapshot copy of all volumes hosting data files.
3. Exit hot backup mode.
4. Execute "alter system archive log current" to force log archival.
5. Create Snapshot copies of all volumes hosting archive logs.

This will yield a set of Snapshot copies containing (a) data files in hot backup mode and (b) the critical archive logs generated while in hot backup mode. These are the two requirements for recovering a database. Files such as control files should also be protected for the sake of convenience, but the only absolute requirements are the data files and archive logs.

Although customers might appear to have very different strategies, almost all of these strategies are ultimately based on the principles outlined earlier in this section.

2.2 Recovery Based on Snapshot

When designing volume layouts for Oracle Databases, one particular decision must be made first: whether volume-based SnapRestore® technology will be used.

Volume-based SnapRestore (VBSR) offers the capability to revert the state of a volume to an earlier point in time almost instantly, but it also means that **all** the data on the volume will be reverted. This is not appropriate for many use cases. For example, if an entire database, including data files, redo logs, and archive logs, was stored on a single FlexVol volume and this volume was restored using VBSR, data would be lost because the newer archive logs and redo data would be discarded.

VBSR is not required for restoration, and many databases can be restored using file-based SnapRestore (SFSR) or by simply copying files from the Snapshot copy back into the active file system.

When a database is very large or it must be recovered as quickly as possible, VBSR is preferred. This requires isolation of the data files. In an NFS environment, data files of a given database must be stored in dedicated FlexVol volumes that are uncontaminated by any other type of file. In a SAN environment, data files must be stored in dedicated LUNs on dedicated FlexVol volumes. If a volume manager is used (including ASM), the disk group must also be dedicated to data files.

Isolating data files in this manner allows the state of the data files to be reverted to an earlier state without damaging other file systems.

2.3 Thin Provisioning

Thin provisioning is of limited use in an Oracle environment because Oracle initializes data files to their full size at the time of creation. Care must be taken when thin provisioning an Oracle environment because data change rates can increase unexpectedly. For example, if tables are reindexed, the Snapshot copy space consumption can grow rapidly, or a misplaced RMAN backup can write a large amount of data in a very short time. Finally, it can be difficult to recover an Oracle Database if a file system runs out of free space during data file extension.

Most problems can be avoided by careful configuration of volume autogrow and Snapshot autodelete policies.

NFS

Most customers using Oracle in an NFS environment configure their Oracle Databases to automatically extend data files and use volume autogrow to make sure adequate free space exists in the volume.

SAN

The efficiency of thin provisioning in a file system environment can be lost over time as deleted and erased data occupies more and more unallocated white space in the file system.

Thin provisioning is more effectively used at the logical volume layer. When a logical volume manager such as Veritas™ VxVM or Oracle ASM is used, the underlying LUNs are divided into extents. These extents will only be utilized when needed. For example, if a database begins at 2TB in size but might grow to 10TB over time, it could be placed on 10TB of thin-provisioned LUNs organized in an LVM disk group. It would occupy only 2TB of disk space at the time of creation and would only claim additional space as extents get allocated to accommodate database growth.

3 Recommendations for General Oracle Configuration

The following parameters are generally applicable to all configurations.

3.1 filesystemio_options

This Oracle initialization parameter controls the use of asynchronous and direct I/O. Contrary to common belief, asynchronous and direct I/O are not mutually exclusive. NetApp has observed that this parameter is very frequently misconfigured in customer environments. This misconfiguration is directly responsible for many performance problems.

Asynchronous I/O essentially means that Oracle I/O operations can be parallelized. Prior to the availability of asynchronous I/O on various operating systems, users needed to configure numerous dbwriter processes and change the server process configuration. With asynchronous I/O, the operating system itself performs I/O on behalf of the database software in a highly efficient and parallel manner. This does not place data at risk, and critical operations, such as Oracle redo logging, are still performed synchronously.

Direct I/O means a bypass of the OS buffer cache. I/O on a UNIX system will ordinarily flow through the OS buffer cache. This is of value to applications that do not maintain an internal cache, but Oracle has its own buffer cache within the SGA. In almost all cases, it is better to enable direct I/O and allocate server RAM to the Oracle SGA than to rely on the OS buffer cache. The Oracle SGA will use the memory more efficiently. In addition, when I/O flows through the OS buffer, it is subject to additional processing, which increases latencies. The increased latencies are especially noticeable with heavy write I/O, where low latency is a critical requirement.

The options for `filesystemio_options` can be summarized as follows:

- **Asynchronous I/O.** Oracle should submit I/O requests to the operating system for processing. This permits Oracle to carry on with other work rather than waiting for I/O completion and increases parallelization of I/O.
- **Direct I/O.** Oracle should perform I/O directly against physical files rather than routing I/O through the host operating system cache.
- **None.** Use synchronous and buffered I/O. In these configurations, the choice between shared and dedicated server processes and the number of dbwriters will become more important.
- **Setall.** Use both asynchronous and direct I/O.

In almost all cases, the use of “setall” is the optimum value, but the following considerations should be made:

- Some customers might have encountered async I/O problems in the past, especially with Red Hat Enterprise Linux® (RHEL4) releases. These problems are no longer reported, and asynchronous I/O is stable on all current operating systems.
- If a database has been using buffered I/O, a switch to direct I/O might also warrant a change to the SGA size. Disabling buffered I/O will result in a loss of the performance benefit of the host OS cache to the database. Adding RAM back to the SGA will repair this damage. The net result should be improvement in I/O performance.
- Although it is nearly always better to use OS RAM for the Oracle SGA rather than use the host OS buffer cache, sometimes it is impossible to determine the best value. For example, on a database server with many Oracle instances that are intermittently active, it might be preferable to use buffered I/O with very small SGA sizes. This will allow the remaining free RAM on the OS to be used flexibly by all running database instances. This is a highly unusual situation, but it has been observed at some customer sites.

  Note: The `filesystemio_options` parameter has no effect in DNFS and ASM environments. The use of Direct NFS (DNFS) or Automatic Storage Management (ASM) automatically results in the use of both asynchronous and direct I/O.

3.2 db_file_multiblock_read_count

This parameter controls the maximum number of Oracle Database blocks that Oracle will read as a single operation during sequential I/O.

- It is NOT the number of blocks Oracle reads during any and all read operations. It does not affect random I/Os. Only sequential I/O is affected.
- Oracle recommends that this parameter be left unset by the user, which means that the database software will automatically set the optimum value. This generally means it will be set to a value that yields an I/O size of 1MB. For example, a 1MB read of 8k blocks would require 128 blocks to be read, and the default value for this parameter would therefore be 128.
- Most database performance problems observed by NetApp at customer sites involve this parameter being incorrectly set. There were valid reasons to change this value with Oracle 8 and Oracle 9, and the parameter might be unknowingly present in `init.ora` files as the database was upgraded in place to Oracle 10 and later. A legacy setting of 8 or 16, compared to a default value of 128, will significantly damage sequential I/O performance.

Absent proof that a change to this value has a measurable benefit on performance as seen by the users, this parameter should not be set in the `init.ora` file.

3.3 Redo Block Size

Oracle supports either a 512-byte or 4k-byte redo block size. The default is 512 bytes. The best option is expected to be 512 bytes because this minimizes the amount of data written during redo operations. However, it is possible that the 4k size would offer a performance benefit at very high logging rates. For example, a single database with 50MB/sec of redo logging might be more efficient if the redo block size is larger. A storage system supporting many databases with a large total amount of redo logging might benefit from a 4k redo block size because it would eliminate inefficient partial I/O processing where only part of a 4k block needs to be updated.

The default block size should only be changed upon the specific recommendation of NetApp or Oracle Customer Support and should be based on analysis of actual I/O patterns on a running database.

4 Oracle RAC

This section applies to Oracle 10.2.0.2 and later. For earlier versions of Oracle, consult Oracle document 294430.1 in conjunction with this document to determine optimal settings.

4.1 disktimeout

The primary storage-related Real Application Cluster (RAC) parameter is `disktimeout`. This parameter controls the threshold for voting file I/O to complete. If `disktimeout` is exceeded, the RAC node will be evicted and will reboot.

The default for this parameter is 200. This should be sufficient for standard cluster takeover/giveback procedures. NetApp strongly recommends that RAC configurations be tested thoroughly before being placed into production because many factors affect a takeover/giveback. In addition to the time required for storage failover to complete, additional time is also required for Link Aggregation Control Protocol (LACP) changes to propagate; SAN multipathing software to detect an I/O timeout and retry on an alternate path; and, if a database is extremely active, a large amount of I/O to be queued and retried before voting disk I/O is processed.

If an actual storage takeover/giveback cannot be performed, the effect can be simulated through cable pull tests on the database server.

4.2 misscount

The `misscount` parameter ordinarily affects only the network heartbeat between RAC nodes. The default is generally 30 seconds. If the OS boot disk is not local, this parameter might become important. This includes hosts with boot disks located on an FC SAN, NFS-booted operating systems, and boot disks located on virtualization datastores such as a VMDK file. If access to a boot disk is interrupted by a storage takeover/giveback, it is possible that the entire OS will temporarily hang. The time required for Data ONTAP to complete the storage operation and for the OS to change paths and resume I/O might exceed the `misscount` threshold. The result will be that a node will immediately evict after connectivity to the boot LUN is restored. In most cases, the eviction and subsequent reboot will occur with no logging messages to indicate the reason for the reboot. Not all configurations are affected, so any SAN booting, NFS booting, or datastore-based host in a RAC environment should be tested so that RAC remains stable if communication to the boot disk is interrupted.

In the case of nonlocal boot disks, `misscount` might need to be changed to match `disktimeout`. If this parameter is changed, further testing should be conducted to also understand the effect on RAC behavior such as node failover time.

5 Use of Flash Cache, Flash Pool, and SSD for Oracle Environments

A comprehensive explanation of the use of flash and SSD technologies with Oracle Databases is beyond the scope of this document, but some common questions and common errors must be considered.

All principles explained in this section apply equally to all protocols and file systems, including Oracle ASM.

Flash Cache: flexscale.lopri_blocks

This parameter applies to the use of Flash Cache™ intelligent caching. The default for this option is `off`, which means I/O involving generally low-priority block operations such as random overwrites and sequential I/O should **not** be cached. The reason is simple: Most databases are limited by latency on random read operations. When a random overwrite occurs, an Oracle Database will almost always retain a copy of that block, and it is highly unlikely to be reread soon, so caching overwrites waste valuable space in Flash Cache. When Oracle performs sequential read I/O, it is a very large-block operation that is inherently efficiently processed by a storage array, even if the underlying disk is SATA. This type of I/O does not benefit from Flash Cache, and attempting to cache that I/O will generally place unnecessary load on the CPU and will again waste valuable space in Flash Cache that could be better used for caching random I/O.

Note: This parameter should be changed only following careful consultation with NetApp Customer Support or Professional Services.

Use of SSD Aggregates

Placing redo logs on an SSD aggregate is a frequent error. An SSD drive is valuable for improving logging performance when used with direct connected devices, but NetApp arrays already contain nonvolatile, mirrored NVRAM/NVMEM-based solid-state storage. When an Oracle Database performs a write operation, the write is acknowledged as soon as it is journaled into NVRAM/NVMEM. Write performance is not affected by the type of drives that eventually receive the writes.

At best, use of an SSD aggregate for hosting sequential writes such as redo logging or for temporary data file I/O will have no effect, but in most cases SSD aggregates have far fewer devices than the SAS or SATA aggregates on the system. NetApp has observed severe performance problems caused by moving sequential write heavy workloads to an SSD aggregate with too few devices.

SSD aggregates should be reserved for workloads involving random I/O. Indexes are particularly good candidates for placement on SSD drives. NetApp Professional Services can assist in analyzing Oracle AWR or statspack files for a more detailed analysis.

Flash Pool: Write Caching

The same principles explained earlier regarding Flash Cache and SSD aggregates apply to Flash Pool™ intelligent caching. Using Flash Pool for write caching is more likely to damage performance than help because (a) writes commit to the NVRAM/NVMEM cache first and (b) Oracle is unlikely to write and then reread data quickly, and therefore caching of this low-priority write I/O will displace caching of more important read activity. There are exceptions, especially where the Oracle buffer cache is under pressure and blocks are aging out of cache only to be read again quickly. Testing of the benefit of write caching is encouraged where Oracle block write levels are high.

6 Ethernet Configuration

The TCP/IP settings required for Oracle Database software installation itself are usually sufficient to provide good performance for all NFS or iSCSI storage resources. In some cases, NetApp has seen performance benefits in 10GbE environments from implementing specific recommendations from the manufacturer of the network adapter.

6.1 Ethernet Flow Control

This technology allows a client to request a sender to temporarily stop transmission of data. This is usually done because the receiver is unable to process incoming data quickly enough. At one time, requesting a sender to cease transmission was less disruptive than having a receiver start discarding packets because buffers were full.

Performance problems caused by Ethernet flow control have been increasing in recent years. The reason is that Ethernet flow control operates at the physical layer. If a network configuration permits any database server to send an Ethernet flow control request to a storage system, the result is a pause in I/O for **all** connected clients. As an increasing number of clients are served by a single storage controller, the likelihood of one or more sending flow control requests increases. The problem has been seen frequently at customer sites with extensive use of OS virtualization.

A NIC on a NetApp system should not receive flow control requests. The method to achieve this varies based on the network switch manufacturer. In most cases, flow control can be set to `receive desired or receive on`, which means a flow control request will not be forwarded to the storage controller. In other cases, the network connection on the storage controller might not allow flow control to be disabled. In these cases, the clients must be configured to never send flow control requests, by a change to either the NIC configuration on the database server itself or the switch ports to which the database server is connected.

6.2 Jumbo Frames

The use of jumbo frames has been shown to offer some performance improvement in GbE networks by reducing CPU and network overhead, but the benefit is not usually significant. Even so, NetApp recommends trying to implement jumbo frames where possible, both to achieve the performance benefit that is possible and to future-proof the solution.

The use of jumbo frames in a 10GbE network should be considered nearly mandatory. The reason is that without jumbo frames, most 10GbE implementations will reach a packets/second limit before they reach the 10GbE mark. Using jumbo frames improves efficiency in TCP/IP processing because it allows the database server, NICs, and storage system to process fewer but larger packets. The performance improvement varies from NIC to NIC, but it is significant.

There is a common incorrect belief that implementing jumbo frames requires all connected devices to support jumbo frames. Two network endpoints should negotiate the highest available frame such as when establishing a connection. In a typical environment, a network switch has an MTU size set to 9216, the NetApp controller is set to 9000, and the clients are a mix of 9000 and 1514. The clients that can support

an MTU of 9000 would be using jumbo frames, and the clients that only support 1514 would negotiate a lower value.

Problems are rarely seen in a completely switched environment. Care must be taken in a routed environment so that no intermediate router is forced to fragment jumbo frames.

6.3 TCP Parameters

There are three settings that are frequently misconfigured: TCP timestamps, SACK, and TCP window scaling. Many out-of-date documents continue to exist on the Internet and recommend disabling one or more of these parameters in order to improve performance. There was some merit to this many years ago when CPU capabilities were much lower and there was a benefit to reducing the overhead on TCP processing wherever possible.

With modern operating systems, disabling any of these TCP features usually results in no detectable benefit or damages performance. Performance damage is especially likely in virtualized networking environments without these features because they are required for efficient handling of packet loss and changes in network qualify. The presumption should be that any server hosting Oracle Databases has TCP timestamps, SACK, and TCP windowing scaling all enabled.

7 General NFS Configuration

7.1 Installation and Patching

The presence of the following mount options in ORACLE_HOME causes host caching to be disabled:

```
cio, actimeo=0, noac, forcedirectio.
```

This can have a severe negative impact on the speed of Oracle software installation and patching. Many customers will temporarily remove these mount options during installation or patching of the Oracle binaries. This can be done safely if the user verifies that no other processes are actively using the target ORACLE_HOME during the installation or patching process.

7.2 Clustered Data ONTAP and NFS Flow Control

Under some circumstances, the use of clustered Data ONTAP requires changes in the Oracle or Linux kernel parameter. The reason is related to NFS flow control. This should not be confused with Ethernet flow control. NFS flow control allows an NFS server such as Data ONTAP to limit network communication with an NFS client that is not acknowledging receipt of data. This protects the NFS server in the cases where a malfunctioning NFS client is requesting data at a rate beyond its ability to process the responses. Without protection, the network buffers on the NFS server will fill up with unacknowledged packets.

Under rare circumstances, bursts of I/Os from both Oracle DNFS clients and newer Linux NFS clients can exceed the limits at which the clustered Data ONTAP NFS server can protect itself. The NFS client lags in its processing of inbound data while continuing to send requests for more data. This can lead to performance and stability problems with NFS connectivity.

Although problems are rare, as a best practice NetApp recommends the following protective measures. This applies only to clustered Data ONTAP. These changes should not adversely affect performance.

1. Where Oracle DNFS is used, set the `DNFS_BATCH_SIZE` parameter to `128`. This parameter is available with Oracle 11.2.0.4 and higher. For earlier versions of Oracle, contact Oracle customer support regarding availability of a patch or contact NetApp for other recommendations.
2. When using a newer Linux distribution, make sure that the TCP slot tables are limited. These parameters control the number of outstanding NFS operations that can exist at one time. Run `sysctl –a` and look for the following parameter:

```
sunrpc.tcp_max_slot_table_entries
```

If it exists, set both of the following parameters to `128`:

```
sunrpc.tcp_max_slot_table_entries
sunrpc.tcp_slot_table_entries.
```

The result of these parameter changes will be to limit the number of outstanding I/O operations that exist at any one time.

7.3 NFS Locking

If an Oracle Database server crashes, it might have problems with stale NFS locks upon restart. This problem can be avoided with careful attention to the configuration of name resolution on the server. If this is not possible, the locks will need to be cleared manually on the storage system. The following article explains the options in detail: https://kb.netapp.com/support/index?page=content&id=1010994.

Problems resulting from stale locks can usually be avoided entirely. The NLM lock manager uses `uname -n` to determine the host name, whereas the rpc.statd process uses `gethostbyname()` to determine the host name. These need to match for the OS to properly clear stale locks. For example, the host might be looking for locks owned by "filer5," but the locks were registered by the host as "filer5.mydomain.org." If `gethostbyname()` does not return the same value as `uname -a`, the lock release process will not succeed.

Following is a sample script to verify whether name resolution is fully consistent:

```
#! /usr/bin/perl
$uname=`uname -n`;
chomp($uname);
($name, $aliases, $addrtype, $length, @addrs) = gethostbyname $uname;
print "uname -n yields: $uname\n";
print "print gethostbyname yields: $name\n";
```

If `gethostbyname` does not match `uname`, there is a likelihood of stale locks. For example, this result would show a potential problem:

```
uname -n yields: filer5
print gethostbyname yields: filer5.mydomain.org
```

The solution is usually found by changing the order in which hosts appear in `/etc/hosts`. For example, assume the hosts file included this entry:

```
10.156.110.201  filer5.mydomain.org filer5 loghost
```

The solution would be to change the order in which the FQDN and short host name appear. This would result in `gethostbyname()` returning the short `filer5` host name. This now matches the output of `uname`, and locks will be cleared automatically after a server crash.

8 General SAN Configuration

8.1 LUN Alignment

LUN alignment refers to optimizing I/O with respect to the underlying file system layout. On a NetApp system, the storage is organized in 4k units. An 8k block on an Oracle data file should be aligned to exactly two 4k blocks. If an error in LUN configuration shifted the alignment by 1k in either direction, each 8k Oracle block would exist on three different 4k storage blocks rather than on only two. The end result of this is increased latency and additional I/O performed within the storage system.

LUN alignment is generally only a concern when a logical volume manager is not used, which means that the primary concerns are with the use of Linux and Solaris. If a physical volume within a logical volume group is defined on the whole disk device, meaning no partitions are created, the first 4k block on the LUN will align to the first 4k block on the storage system. This is a correct alignment. Problems arise with partitions because they shift the starting location at which the OS will use the LUN. As long as the offset is shifted in whole units of 4k, the LUN will be aligned. In Linux environments, logical volume groups should be built on the whole disk device. Where a partition is required, alignment can be checked by running "fdisk –u" and verifying that the "Start" of each partition is a multiple of 8.

Solaris environments are more complicated; refer to the appropriate host utilities documentation for further information.

See http://support.netapp.com/documentation/productlibrary/index.html?productID=61343,

Caution
In Solaris x86 environments, additional care must be taken for proper alignment because most configurations have several layers of partitions. Solaris x86 partition slices usually exist on top of a standard master boot record partition table.

8.2 LUN Misalignment Warnings

Oracle redo logging will normally generate unaligned I/O that can cause misleading warnings about misaligned LUNs on Data ONTAP. Oracle redo logging performs a sequential overwrite of the redo log file with writes of varying size. A log write operation that does not align to 4k boundaries will not ordinarily cause performance problems because the next log write operation will complete the block. The end result is that Data ONTAP will be able to process almost all writes as complete 4k blocks, even though the data in some 4k blocks was written in two separate operations.

Alignment can be verified through the use of utilities, such as `sio` or `dd`, that can generate I/O at a defined block size and then view the I/O alignment statistics on the storage system with the `stats` command.

8.3 LUN Count

Oracle Database performance is affected by the capability to perform parallel I/O through the SCSI layer. The result is that two LUNs will offer better performance than a single LUN. The simplest method to increase parallelism is to use a logical volume manager, such as Veritas VxVM, the Linux LVM2, or Oracle Automatic Storage Management (ASM). NetApp customers have generally experienced minimal benefit from increasing the number of LUNs beyond 8, although testing with 100% solid-state drive (SSD) environments with very heavy random I/Os has shown further improvement up to 64 LUNs. The general recommendation is to build a volume group with an extent size that enables I/O to be evenly distributed. For example, a 1TB volume group composed of 10 100GB LUNs and an extent size of 100MB would yield 10,000 extents in total (1,000 extents per LUN). The resulting I/O on a database placed on this 1TB volume group should be evenly distributed across all 10 LUNs.

Striping is not generally necessary. Most databases are limited by random I/O performance, not sequential performance. A data file that exists across a large number of extents enables a large amount of random I/O to be randomized across many extents. This means that all LUNs in the volume group will be evenly used, and no individual LUN will limit performance.

8.4 Data File Block Size

Some operating systems offer a choice of file system block sizes. For file systems supporting data files, the block size should be 4k. There might be cases in which a larger value is warranted, but it must be a multiple of 4k. If a data file is placed on a file system with a 512-byte block, there is a possibility of misaligned files. The LUN and the file system might be properly aligned based on NetApp

recommendations, but the file I/O itself would be misaligned. The result will be severe performance problems.

8.5 Redo Block Size

File systems supporting redo logs must use a block size that is a multiple of the redo block size. This will generally require that both the redo log file system and the redo log itself use a block size of 512 bytes. At very high redo rates, it is possible that 4k block sizes will perform better because this allows I/O to be performed in fewer and more efficient operations. If redo rates are greater than 50MB/sec, consider and test using a 4k block size.

A few customer problems have been identified with databases using redo logs with a 512-byte block size on a file system with a 4k block size and with many very small transactions. The overhead involved in applying multiple 512-byte changes to a single 4k file system block led to performance problems, which were resolved by changing the file system to use a block size of 512 bytes.

9 AIX

9.1 General Notes

The mount option `cio` is especially important in IBM AIX environments. It prevents performance limitations caused by serialization of write I/O and file system locking operations. This applies to both NFS and SAN file systems.

Concurrent I/O

The mount option `cio` enables concurrent I/O. Optimum performance requires the use of concurrent I/O on an AIX system. Without concurrent I/O, performance limitations are likely because AIX will then perform serialized atomic I/O, which incurs significant overhead.

The best method for concurrent I/O is using the `init.ora` parameter `filesystemio_options=setall`. This allows Oracle to open specific files for use with concurrent I/O. Using `cio` as a mount option forces the use of concurrent I/O, which can have negative consequences. For example, forcing concurrent I/O will disable readahead on file systems, which can damage performance for I/O occurring outside the Oracle Database software, such as copying and tape backups. Furthermore, products such as Oracle GoldenGate and SAP® BR*Tools are not compatible with the use of the `cio` mount option with certain versions of Oracle.

For these reasons, NetApp does not recommend using the `cio` mount option at the file system level. Concurrent I/O should be enabled through the use of `filesystemio_options=setall`.

9.2 AIX NFSv3 Mount Options

The following options should be used.

Table 1) Single instance.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,[cio?]
ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,intr

Table 2) Real Application Clusters.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,[cio?],nointr,noac
CRS/Voting	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,cio,nointr,noac
dedicated ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
shared ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr

The primary difference between single-instance and RAC mount options is the addition of `noac` to the mount options. This has the effect of disabling the host OS caching, which enables all instances in the RAC cluster to have a consistent view of the state of the data. Although the use of the `cio` mount option and the init.ora parameter `filesystemio_options=setall` has the same effect of disabling host caching, use of `noac` is still required.

The reason `noac` is required for shared ORACLE_HOME deployments is to facilitate consistency of files such as the Oracle password files and spfiles. If each instance in a RAC cluster has a dedicated ORACLE_HOME, this parameter is not required.

Some customers have reported performance problems resulting from an excessive amount of I/O on data in the ADR_HOME location. Removal of the `noac` mount option will allow host OS caching to occur and reduce storage I/O levels.

Note: This step should be taken only in consultation with NetApp and Oracle Support.

9.3 AIX jfs/jfs2 Mount Options

The following options should be used.

Table 3) Single instance.

File Type	Mount Options
ADR_HOME	defaults
controlfiles, datafiles, redo logs	defaults,[cio?]
ORACLE_HOME	defaults

Before using AIX hdisk devices in any environment, including databases, the parameter queue_depth should be checked. This is not HBA queue depth; it relates to the SCSI queue depth of the individual hdisk device. Depending on how the LUNs were configured, the value for queue_depth might be too low to offer good performance. Testing has shown the optimum value to be 32–64.

10 HP-UX

10.1 HP-UX NFSv3 Mount Options

The following options should be used.

Table 4) Single instance.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,suid
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,forcedirectio,nointr,suid
ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,suid

Table 5) Real Application Clusters.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,noac,suid
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,forcedirectio,suid
CRS/Voting	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,forcedirectio,suid
dedicated ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,suid
shared ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,suid

The primary difference between single-instance and RAC mount options is the addition of `noac` and `forcedirectio` to the mount options. This has the effect of disabling the host OS caching, which enables all instances in the RAC cluster to have a consistent view of the state of the data. Although the use of the init.ora parameter `filesystemio_options=setall` has the same effect of disabling host caching, use of `noac` and `forcedirectio` is still required.

The reason `noac` is required for shared ORACLE_HOME deployments is to facilitate consistency of files such as the Oracle password files and spfiles. If each instance in a RAC cluster has a dedicated ORACLE_HOME, this parameter is not required.

Some customers have reported performance problems resulting from an excessive amount of I/O on data in the ADR_HOME location. Removal of the `noac` mount option allows host OS caching to occur and reduces storage I/O levels.

Note: This step should be taken only in consultation with NetApp and Oracle Support.

10.2 HP-UX VxFS Mount Options

The following mount options should be used for file systems hosting Oracle binaries:

```
delaylog,nodatainlog
```

The following mount options should be used for file systems containing data files, redo logs, archive logs, and control files in which the version of HP-UX does not support concurrent I/O:

```
nodatainlog,mincache=direct,convosync=direct
```

Where concurrent I/O is supported (VxFS 5.0.1 and later, or with the ServiceGuard Storage Management Suite), the following mount options should be used for file systems containing data files, redo logs, archive logs, and control files:

```
delaylog,cio
```

Note: The parameter `db_file_multiblock_read_count` is especially critical in VxFS environments. Oracle recommends that this parameter remain unset in Oracle 10*g*™ R1 and later unless specifically directed otherwise. The default with an Oracle 8k block size is 128. If the value of this parameter is forced to 16 or less, the use of the `convosync=direct mount` option will damage sequential I/O performance and should be removed. This will damage other aspects of performance, and this step should only be taken if it is determined that the value of `db_file_multiblock_read_count` is genuinely required to be changed from the default value.

11 Linux

11.1 General Notes

NFS performance on Linux depends a great deal on a parameter called `tcp_slot_table_entries`. This parameter regulates the number of outstanding NFS operations that are permitted on a Linux operating system.

The default in most 2.6-derived kernels, which includes RH5 and OL5, is 16, and it frequently causes performance problems. An opposite problem occurs on newer kernels in which the `tcp_slot_table_entries` value is uncapped and can cause storage problems by flooding the system with excessive requests.

The solution is to statically set this value. Any Linux operating system using NetApp NFS storage with an Oracle Database should use a value of 128.

RHEL6.2 and Earlier

This is done by placing the following entry in `/etc/sysctl.conf`:

```
sunrpc.tcp_slot_table_entries = 128
```

In addition, there is a bug in most Linux distributions using 2.6 kernels. The startup process reads the contents of /etc/sysctl.conf before the NFS client is loaded. The result is that when the NFS client is eventually loaded, it takes the default value of 16. To avoid this problem, edit `/etc/init.d/netfs` to call `/sbin/sysctl -p` in the first line of the script so that `tcp_slot_table_entries` is set to 128 before NFS mounts any file systems.

RHEL6.3 and Later

The following modification should be applied in the RPC file of the clients using RHEL 6.3 and later:

```
echo "options sunrpc udp_slot_table_entries=64 tcp_slot_table_entries=128
tcp_max_slot_table_entries=128" >> /etc/modprobe.d/sunrpc.conf
```

11.2 Linux NFSv3 Mount Options

The following options should be used.

Table 6) Single instance.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr
ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr

Table 7) Real Application Clusters.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,actimeo=0
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,actimeo=0
CRS/Voting	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,actimeo=0
dedicated ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
shared ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,actimeo=0

The primary difference between single-instance and RAC mount options is the addition of `actimeo=0` to the mount options. This has the effect of disabling the host OS caching, which enables all instances in the RAC cluster to have a consistent view of the state of the data. Although the use of the `init.ora` parameter `filesystemio_options=setall` has the same effect of disabling host caching, use of `actimeo=0` is still required.

The reason `actimeo=0` is required for shared ORACLE_HOME deployments is to facilitate consistency of files such as the Oracle password files and spfiles. If each instance in a RAC cluster has a dedicated ORACLE_HOME, this parameter is not required.

Some customers have reported performance problems resulting from an excessive amount of I/O on data in the ADR_HOME location. Removal of the `actimeo=0` mount option allows host OS caching to occur and reduces storage I/O levels.

Note: This step should be taken only in consultation with NetApp and Oracle Support.

Linux Direct NFS

One additional parameter is required when DNFS is enabled and a source volume is mounted more than once on a single server in a nested mount. This is seen primarily in environments supporting SAP applications. For example, a single volume on a NetApp system could have a directory located at /vol/oracle/base and a second at /vol/oracle/home. If /vol/oracle/base is mounted at /oracle and /vol/oracle/home is mounted at /oracle/home, the result is nested NFS mounts that originate on the same source.

The OS can detect the fact that /oracle and /oracle/home reside on the same volume, which is the same source file system. It will then use the same device handle for accessing the data. This improves the use of OS caching and certain other operations, but it interferes with DNFS. If DNFS needs to access a file on /oracle/home, such as the spfile, it might erroneously attempt to use the wrong path to the data. The result is a failed read or write operation. In these configurations the `nosharecache` mount option should be added to any NFS file system that shares a source FlexVol volume with another NFS file system on that host. This forces the Linux OS to allocate an independent device handle for that file system.

11.3 Linux ext3/ext4 Mount Options

NetApp recommends using the default mount options.

12 Solaris

12.1 Solaris NFSv3 Mount Options

The following options should be used.

Table 8) Single instance.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,llock,suid
ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,suid

The use of llock has proven to dramatically improve performance in customer environments by removing the latency of acquiring and releasing locks on the storage system. This option should be used with care in environments in which numerous servers are configured to mount the same file systems and in which Oracle is configured to mount these databases. This is a highly unusual configuration, but it has been observed. If an instance is accidentally started a second time, data corruption can occur because Oracle would be unable to detect the lock files on the foreign server. NFS locks do not otherwise offer protection; as in version 3, they are advisory only.

Because the `llock` and `forcedirectio` parameters are mutually exclusive, it is important that `filesystemio_options=setall` is present in the init.ora file so that `directio` is used. Without this parameter, host OS buffer caching will be used, and performance can be adversely affected.

Table 9) Real Application Clusters.

File Type	Mount Options
ADR_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,noac
controlfiles, datafiles, redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,force directio
CRS/Voting	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,force directio
dedicated ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,suid
shared ORACLE_HOME	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536,nointr,noac,suid

The primary difference between single-instance and RAC mount options is the addition of `noac and forcedirectio` to the mount options. This has the effect of disabling the host OS caching, which enables all instances in the RAC cluster to have a consistent view of the state of the data. Although the use of the init.ora parameter `filesystemio_options=setall` has the same effect of disabling host caching, use of `noac` and `forcedirectio` is still required.

The reason `actimeo=0` is required for shared ORACLE_HOME deployments is to facilitate consistency of files such as the Oracle password files and spfiles. If each instance in a RAC cluster has a dedicated ORACLE_HOME, this parameter is not required.

Some customers have reported performance problems resulting from an excessive amount of I/O on data in the ADR_HOME location. Removal of the `noac` mount option allows host OS caching to occur and reduces storage I/O levels.

Note: This step should be taken only in consultation with NetApp and Oracle Support.

12.2 Solaris ufs Mount Options

NetApp strongly recommends the use of the “logging” mount option so that data integrity remains in the case of a Solaris host crash or interruption of FC connectivity and so that Snapshot backups are usable.

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

NetApp provides no representations or warranties regarding the accuracy, reliability, or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer’s responsibility and depends on the customer’s ability to evaluate and integrate them into the customer’s operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.

Go further, faster®

TR-3633-0314

NetApp
www.netapp.com