19c ORACLE Database

Oracle ACFS

NAS Maximum Availability Extensions

ORACLE WHITE PAPER | FEBRUARY 2019

Table of Contents

Introduction

Oracle ACFS is a clustered file system, provided as part of the Oracle Clusterware technology set, which provides for access to files on any node of the Oracle Grid Cluster. This technology is available to both Oracle Linux Support licensees, or as part of the Oracle RAC license.

Building on top of the Oracle ACFS file system, Oracle ACFS NAS Maximum Availability Extensions (ACFS NAS|MAX) is a set of technologies that utilizes Oracle ACFS and Oracle Clusterware Resources to support running certain OS protocols (such as NFS or SMB) over ACFS. In doing so, Oracle ACFS NAS|MAX provides for Highly Available Service Architecture, to the limit of the RAC cluster availability. Oracle ACFS NAS|MAX is built on top of the Highly Available Technology provided by the Oracle Grid Infrastructure components and the OS protocol technology.

Beginning with 12.2.0.1, Oracle ACFS NAS|MAX provides highly available front ends for:

- NFS v3, v4, and v4 w\Locking (HA-NFS)
- SMB protocol for Windows and Samba (HA-SMB)

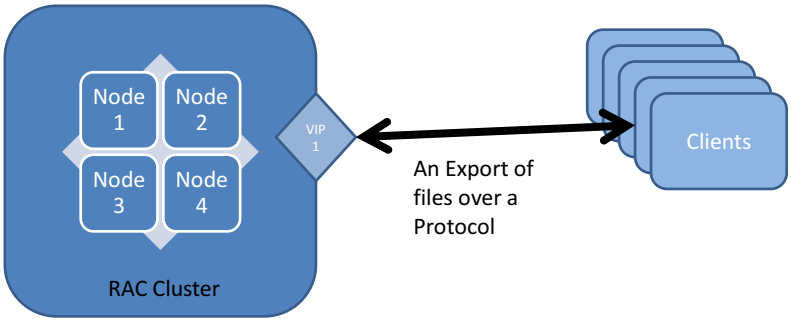
With this technology, customers will be able to create and maintain a set of NFS or SMB exports that survive the loss of any one particular node in the Oracle Grid Cluster, ensuring that the exports are always available to clients. This is a key part of many a Maximum Availability Architecture.

Within this white paper, the reader will learn about the current features of the technology, basic and more advanced configuration, limitations and system requirements.

High Level Design

At a high level, Oracle ACFS NAS|MAX provides an architecture where clients connect to a virtual IP (VIP). This VIP is controlled by CRS resources and will move around the cluster, should the services not be available on the current node. In most cases, clients are unaware that the server has been replaced by another server, and continue to carry on transactions with the new server that the VIP is hosted on.

Conceptual Design



As long as a Node in the RAC Cluster is available, the Export will remain logically available over the VIP. Clients will be unaware of which RAC node is providing access to the VIP.

Benefits of Oracle ACFS NAS|MAX

- Oracle ACFS NAS|MAX is built on top of ACFS, a cluster file system, allowing NFS and SMB shares to leverage the management interfaces and advantages of ACFS.
  - Because ACFS is a cluster file system, this means that data is available from any node in the RAC cluster.
  - ACFS tagging allows files to be tagged with attributes for easy filtering later, such as for replication or backup. (Tag information is only available on the Hosts, not on the Clients.)
  - ACFS replication allows for the entire file system to be replicated across clusters, providing for high availability across data centers.

- ACFS snapshots allow for admins to create copies of data for quick recovery, controlled changes and backups. These snapshots can be exported via ACFS NAS|MAX.

- ACFS Security and Encryption allows for flexible security rules to define access rights to the files on the file system, as well as industry standard disk encryption.

- Ability to automatically resize a file system while the file system is mounted and available.

- ACFS Compression can reduce the size of the information stored on the ACFS File System.

- For more information on these and other Oracle ACFS Features, please see the Oracle ASM Administrators Guide.

- ACFS NAS|MAX provides for SMB and NFS integration into the Grid Infrastructure stack startup and shutdown, providing an alternative to traditional SMB and NFS node start init script configuration. Since ASM is not available until later in the boot process, the traditional method of placing exported file systems into the exports file or samba configuration file for export at boot time is not available to file systems hosted on ADVM volumes. Using Oracle ACFS NAS|MAX to provide the exports for your server allows an admin to use ASM features such as failure groups, redundancy and management interface, while still providing for normal NFS and SMB operation.

  - Scalability – Performance of the cluster throughput can be scaled by adding new nodes to the Oracle ACFS NAS|MAX cluster, to the limits of the backend network and storage fabric.

Supported Platforms

Oracle ACFS NAS|MAX is supported on most platforms that ACFS is supported on. **For the most updated content, please refer to Knowledge Base Article #1369107.1 on My Oracle Support.**

Note that there are specific requirements for certain OS variants, listed below:

- AIX HA-SMB: Samba version 3.6.24 or later
- Solaris HA-SMB: Samba version 3.6.24 or later
- Linux HA-NFS: nfs-utils-1.0.9-60 or later
- Linux HA-SMB: Samba version 4.2 or later
- Windows – SMB only
  - SMB is built into Windows. Newer servers support newer SMB versions, but the client can force an older version. Running the newest Windows OS on the server is recommended.

Oracle ACFS NAS|MAX and underlying OS Services

Oracle ACFS NAS|MAX relies on the underlying technology stack from the OS. While Oracle ACFS NAS|MAX adds additional functionality layered overtop of the underlying OS protocols, it does not replace the underlying OS protocols.

Before attempting to configure ACFS NAS|MAX, ensure that the underlying OS is able to export and service the protocol you are attempting to utilize (SMB\NFS). Ensuring that basic operation works will help to rule out other errors when ACFS NAS|MAX is brought into play.

For multi-homed systems, ensure that routes are available for each network you will be using to communicate between the server and the client.

Ensure that the firewalls and routers are configured to pass the appropriate ports and packets.

SMB Support

On Windows, SMB has several protocols. The protocol negotiation is driven by 2 factors:

1. The Server OS that you are running – for instance, Windows 2012R2 supports a much newer SMB protocol than 2008R2.

2. The Client OS that you are running – the server will negotiate the agreed upon version based upon which SMB version the Client and Server can both handle.

For more information, please see this link.

For error recovery and resume features (Highly Available), it is strongly suggested that both Server and Client use at least SMB 3.0 versions. This would require Windows 2012 and later OS versions.

Other versions of SMB will work, but may:

a) Require that the client remount the share after a network failure

b) Exhibit slower transfer speeds

The Powershell cmdlet **Get-SmbConnection** can be used to check which version of SMB the Server or Client is using.

On Linux, Solaris, and AIX, the Samba packages must be installed. These packages are available from www.samba.org. Some OS vendors may provide an alternative SMB implementation, but HA-SMB requires Samba.org. It is recommended that the latest versions of Samba be used.

NFS Support

On AIX, Solaris, and Linux, HA-NFS supports v3 and v4 with No Locking.

Additionally, on Linux only, HA-NFS supports NFSv4 with Locking.

NFS is generally a part of the OS kernel - it is recommended that the latest available kernel for your OS be used. Please see MOS Note 2171587.1 for more information on UEK and RH kernels and NFSv4 support.

Description

In addition to ACFS, ADVM and ASM, Oracle ACFS NAS|MAX also relies on new 12.1 CRS resources, the HAVIP and the ExportFS. HA-NFS with Locking relies on a new 12.2 resource the NetStorageService Resource.

The HAVIP resource is a special class of the standard Oracle node VIP resource. This resource is responsible for providing a single IP address in the cluster, on one and only one node. It will move around the cluster as necessary to provide the client facing interface for the export file system. The HAVIP requires one or more configured exports in order to successfully run on a node. When other HAVIPs are configured in a cluster, they will attempt to run on separate nodes of the cluster. From 12.2.0.1 and later, the HAVIP can be given a preferred node to run on, using the –homenode option. This will cause the HAVIP to attempt to move back to its preferred node when that node is available.

The ExportFS resource is responsible for ensuring that the OS exports the file system over NFS. It needs to run over an ACFS file system that has been configured to be available on every node. If the file system is unavailable on any one node, it is responsible for moving to another node, exporting the

file system and continuing to provide the export services there. An ExportFS resource is attached to an HAVIP resource, allowing the HAVIP to provide that export to clients. The placement of the HAVIP dictates which RAC node the Export is serviced from.

The new NetStorageService resource is responsible for managing the OS NFS server. This resource is only available when HA-NFS w\Locking is enabled. Instead of running the NFS service on every node of the RAC cluster, the HA-NFS w\Locking mode runs a single NFS server in the cluster. This NFS server moves around the cluster with the HAVIP resources (even if more than one is configured). This difference in operation is intentional, and is due to the way that locking and the lock recovery grace period is handled in the OS NFS servers.

Client Usage of ACFS NAS|MAX

When using HA-SMB and HA-NFS, the HAVIP and ExportFS resources are treated as a single group by CRS. When not utilizing HA-NFS w\Locking, a simple rule of thumb can help when setting up a highly available export: The HAVIP will run on the node with the most file systems that it depends on (ie, the ones that it is intended to export) available, and the least number of other HAVIPs. This ensures that 2 things are true:

1. The HAVIP needs to ensure that the maximum number of file systems that it is supposed to export (as configured by attached ExportFS resources) are available. It will attempt to run on the node that has the most file systems available.

2. The HAVIP will attempt to distribute itself throughout the cluster from other HAVIP services. This ensures that a minimal level of load-balancing is done, and prevents HAVIPs from clumping on a single node. Note that a preferred home node will override this balancing, and may cause multiple HAVIPs to run on a single RAC node.

With these 2 simple rules in mind, we can generate a series of guidelines for ensuring that HANFS provides the maximum scalability and availability:

1. Exports should be grouped into 2 categories: those that require maximum throughput, and those that require maximum availability.

2. HAVIPs should be configured so that the estimated throughput on all attached ExportFS resources is roughly similar to the other HAVIPs.

3. Exports that require maximum availability should be configured to have their own HAVIP.

Note that when HA-NFS w\Locking is enabled, all NFS shares will run on the same OS Node, regardless of how many HAVIPs are available.

SRVCTL Command Explanation

The full text of the various commands can be found in the Oracle Clusterware Administration and Deployment Guide, Appendix G, SRVCTL Command Reference. Alternatively, running the srvctl command and using the help option will also list the current usage.

The command *'srvctl add HAVIP'* is used to add a new HAVIP to the system. This HAVIP can be used for both HA-NFS and HA-SMB. Some important options are:

-id	This is the unique id being used to identify this HAVIP. It will become the name of the resource, as well as be used later when querying the status of this resource.
-address	*This is the IP Address or hostname that will be used by the HAVIP. Note that the Address must be a non-DHCP, non-round robin DNS address.*
-homenode	This is the preferred node for this HAVIP. If the node is up and available, this HAVIP will run on the preferred node. This resource movement may cause other HAVIPs to reconfigure themselves as they move off that node, or place more than 1 HAVIP on a single node in the cluster.

To manage the ExportFS resources, the *'srvctl * exportfs'* family of commands is used. When creating an ExportFS, use *'srvctl add exportfs'*.

-name	This is a unique identifier for the ExportFS. It will show up in later status commands.
-path	The path that the export will export. Certain operating systems interpret the NFS spec differently. Thus, valid paths may be OS dependent. For instance – Solaris will not allow the export of a sub-directory of an already exported directory. In this case, the start of the ExportFS will display an error message to the user. Additional considerations on Solaris include exporting an ACFS snapshot - the parent of all snapshots will be exported (This will be handled by the resources themselves.)
-id	This is the HAVIP ID that the ExportFS will be attached to.
-options	This is the options that will be passed through to the NFS or SMB Server. For instance, setting RW or RO, various security settings, file system ID, and other OS specific attributes can be placed here. If no option is provided, 'RO' is used. Please see the "Options" section for more information.
-type	This type (NFS or SMB) determines what the type of the export is. Both types can be used on the same HAVIP.
-clients	(Linux Only) On Linux, various client specifiers can be placed here: subnet, IP, hostname, etc. No validation is done of these clients. Also valid is '*', meaning all clients.

Server-Side Export Options

The Options argument for a particular export is treated differently for various OS and Protocol types.

» NFS
  - » For NFS, any valid NFS option can be specified here.
  - » Linux only – clients should not be specified here – they are specified in the –clients option.
  - » For Solaris and AIX, please use the standard NFS options for these platforms to specify clients and security.
  - » When options are not specified, the default options for your NFS server configuration are used. 'ro' is the default option when no options are specified.

» SMB
  - » For SMB, clients should be specified here.
  - » Any supported options are valid here, and must be passed as a comma separated key-value pairs. For instance: "share=share1,group=accessgroup1,description="RW,HR,Accessgroup1""
  - » Windows – any net share command options are supported here. If no options are specified, /GRANT:everyone,READ will be used.
  - » Linux, Solaris, AIX – any smb.conf share configuration parameters that are supported by your Samba version are applicable here. If not options are specified, "read only = yes, browsable = yes" will be used.

Client-Side Mount Options

» **HA-NFS**

When mounting the Oracle ACFS HA-NFS export on the client, the following options are recommended:

» **hard** – this tells the NFS client to continue retrying the operation by attempting to contact the server. This differs from soft mounting – a soft mount will return an error as soon as the file system is unavailable. A hard mount will wait until the file system is available again before making determinations on the state of the file system.

» **intr** – when used in conjunction with hard, this allows NFS operations on the client to be interrupted (such as by ^C or other SIGINT). This allows the user to terminate operations that appear to be hanging.

» **nolock** – HANFS supports NFSv3. This version of NFS does not appropriately handle locks and lock recovery on all platforms and in all scenarios. Thus, for safety sake, it is better to disallow lock operations rather than having an application think locks are working correctly. This is specific to the NFS Server. (Note – when using HANFS v4 w\Locking, this is not necessary.)

» **Retrans=10000** – This tells the NFS client to retry the operation several times in the event of a failure. Since operations are usually quite fast, this can smooth out network glitches and HAVIP failovers.

» **Vers** – When utilizing NFSv4 Locking, ensure that the client mounts using vers=4 (or set this as the system default).

» **Other Options** – Other options, such as wsize and rsize may greatly affect the characteristics of NFS performance. Please consult with your system administrator and application documentation as to recommended NFS options for your environment and usage. For Oracle Applications, please see MOS Note 359515.1 and MOS Note 384248.1, as well as any specific documentation for your application.

» **HA-SMB**

When mounting the Oracle ACFS HA-SMB export on the client, there are no options specified. Options are configured on the server, and as with Oracle ACFS HA-NFS, can greatly affect performance and reliability. Please consult with your administrator and application documentation for required options

Basic NFS and Server Setup

In this simple scenario, we will setup both HA-NFS and HA-SMB on a single server. As always, consult with your system administrator and vendor OS documentation for the configuration steps for your particular platform and installation.

This walkthrough will take place on Oracle Linux 7. Let's start by making sure that the daemons that we require for this OS version are running. We can check this on each node by using:

- ***# /bin/systemctl status rpcbind***

rpcbind.service - RPC bind service
Loaded: loaded (/usr/lib/systemd/system/rpcbind.service; static)
Active: active (running) since Thu 2018-10-25 18:30:12 EDT; 1 day 21h ago
Main PID: 813 (rpcbind)
CGroup: /system.slice/rpcbind.service
??813 /sbin/rpcbind –w
Oct 25 18:30:12 host1 systemd[1]: Started RPC bind service.bash-3.2

- ***# /bin/systemctl status nfs***

nfs-server.service - NFS server and services
Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled)
Active: active (exited) since Thu 2018-10-25 18:30:12 EDT; 1 day 21h ago
Main PID: 1389 (code=exited, status=0/SUCCESS)
CGroup: /system.slice/nfs-server.service

- ***[root@host1 bin]# systemctl status smb***

```
smb.service - Samba SMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/smb.service; disabled)
   Active: active (running) since Thu 2018-10-25 18:30:12 EDT; 1 day 21h ago
 Main PID: 5206 (smbd)
   Status: "smbd: ready to serve connections..."
   CGroup: /system.slice/smb.service
           ??5206 /usr/sbin/smbd
           ??5215 /usr/sbin/smbd

Oct 27 16:21:22 host1.us.oracle.com systemd[1]: Starting Samba SMB Daemon...
Oct 27 16:21:22 host1.us.oracle.com smbd[5198]: [2018/10/27 13:21:22.8987...
Oct 27 16:21:22 host1.us.oracle.com smbd[5198]: standard input is not a s...
Oct 27 16:21:22 host1.us.oracle.com systemd[1]: smb.service: Supervising ...
Oct 27 16:21:23 host1.us.oracle.com smbd[5206]: [2018/10/27 13:21:23.4076...
Oct 27 16:21:23 host1.us.oracle.com systemd[1]: Started Samba SMB Daemon.
Hint: Some lines were ellipsized, use -l to show in full.
```

If one of these had not been running, we could have started it by using:

```
/sbin/systemctl start <service>
```

The ‘systemctl’ command can be used to ensure that these services are started at boot time:

- *bash-3.2# /bin/systemctl enable nfs*
- *bash-3.2# /bin/systemctl enable rpcbind*
- *bash-3.2# /bin/systemctl enable smb*

If SELinux is configured, ensure that any SELinux setup on the system itself is correctly configured for NFS and SMB file systems – you should be running in a mode that allows NFS and SMB network access to be allowed. Generally, this is via 'enforcing – targeted' or 'permissive'.

Creating an initial file system to export

Now that we have our initial setup out of the way, we can configure our file systems. Remember that Oracle ACFS NAS|MAX requires an ACFS file system that is configured to be mounted on all nodes via a single file system resource. Please refer to the Oracle ASM Administrator Guide for more detailed information on these commands. There are several ways to achieve this:

» Using ASMCA:

  » Click the ‘Volumes’ tree entry.
    - Click ‘Create’ button at the bottom of the pane.
    - Follow prompts

- Click the ‘ACFS File Systems’ tree entry.
  - Click ‘Create’ button at the bottom of the pane.
  - Ensure that “Cluster File System” is specified as the “Type of ACFS”
  - Ensure that “Auto Mount” is selected.
  - Click “OK” and run the script, or let ASMCA run the commands for you.

» Command line:

- Create a volume device using asmcmd
- Format the volume device using 'mkfs'
- Use 'srvctl add filesystem -device <device> -path <mount path>' to register the file system with crs
- Use 'srvctl start filesystem -device <device>' to mount the path

Example File System Creation

```
bash-3.2# /sbin/mkfs -t acfs /dev/asm/test1-194
mkfs.acfs: version                   = 19.1.0.0.0
mkfs.acfs: on-disk version           = 39.0
mkfs.acfs: volume                    = /dev/asm/test1-194
mkfs.acfs: volume size               = 5368709120
mkfs.acfs: Format complete.
bash-3.2# mkdir /hr1
bash-3.2# /scratch/crs_home/bin/srvctl add filesystem -path /hr1 -device /dev/asm/test1-194
bash-3.2# /scratch/crs_home/bin/srvctl start filesystem  -device /dev/asm/test1-194
bash-3.2# mount -t acfs
/dev/asm/test1-194 on /hr1 type acfs (rw)
bash-3.2# /scratch/crs_home/bin/srvctl status filesystem  -device /dev/asm/hr1-194
ACFS file system /hr1 is mounted on nodes host1,host2
```

Configuring a simple Oracle ACFS HA-NFS Scenario (No Locking)

For this scenario, let's assume a simple setup of a 2 node RAC cluster. The following exports will be necessary from this cluster:

» /hr1 – expected average throughput around 500KB/s, mostly read.
» /hr2 – expected average throughput around 500KB/s, mostly read.

Each file system is hosted in the same ASM disk group, with no failover and external redundancy. The admin does not expect any availability issues for these file systems, and since they are in the same disk group, it is likely that a storage outage will affect both of them equally. The combined throughput is low enough that there won't likely be any network bandwidth issues if a single node hosted both exports.

Thus, for simplicity, the admin has chosen to create only a single HAVIP and to attach both export file systems to this VIP. This gives all clients a single HAVIP address to access both mount points over. This has the downside that adding new nodes to the Oracle RAC HANFS cluster will not automatically scale the throughput of the entire cluster.

Registering the HAVIP:

```
bash-3.2# /scratch/crs_home/bin/srvctl add HAVIP -id HR1 -address HAVIP1.us.oracle.com -netnum 1 -description
"HR specific exports for the Omega Project"
bash-3.2# /scratch/crs_home/bin/srvctl status HAVIP -id HR1
HAVIP ora.hr1.HAVIP is enabled
HAVIP ora.hr1.HAVIP is not running
bash-3.2# /scratch/crs_home/bin/srvctl start HAVIP -id HR1
PRCR-1079 : Failed to start resource ora.hr1.HAVIP
CRS-2805: Unable to start 'ora.hr1.HAVIP' because it has a 'hard' dependency on resource type
'ora.HR1.export.type' and no resource of that type can satisfy the dependency
```

Why the failure to start? Recall earlier that we mentioned that an HAVIP requires 1 or more ExportFS configured. Without an ExportFS, the HAVIP will not start. If a client had mounted the ExportFS and the HAVIP started without the ExportFS available, the client would receive an ESTALE error. This prevents the resumption of NFS services on the client.

Registering the Export FS

```
bash-3.2# /scratch/crs_home/bin/srvctl add exportfs -path /hr1 -id HR1 -name HR1 -options "rw,no_root_squash" -
clients agraves-vm5,agraves-vm6
bash-3.2# /scratch/crs_home/bin/srvctl status exportfs -name HR1
export file system hr1 is enabled
export file system hr1 is not exported
bash-3.2# /scratch/crs_home/bin/srvctl add exportfs -path /hr2 -id HR1 -name HR2 -options "ro" –clients
10.149.236.0/22
```

At this point, starting either the ExportFS or the HAVIP will start all configured ExportFS on that HAVIP, or will start the associated HAVIP.

We've chosen to export the second ExportFS, HR2 to only the sub-net of the network resource:

```
bash-3.2# /scratch/crs_home/bin/srvctl config exportfs -name HR2
export file system hr2 is configured
Exported path: /hr2
Export Options: ro
Configured Clients: 10.149.236.0/22
bash-3.2# /scratch/crs_home/bin/srvctl config exportfs -name HR1
export file system hr1 is configured
Exported path: /hr1
Export Options: rw,no_root_squash
Configured Clients: host1,host2
```

Compare that with HR1, which is only available to 2 clients: host1 and host2

We can see the configured dependencies from the HAVIP to the other resources:

```
START_DEPENDENCIES=hard(ora.net1.network,uniform:type:ora.HR1.export.type)
attraction(ora.data.hr1.acfs,ora.data.hr2.acfs) dispersion:active(type:ora.HAVIP.type) pullup(ora.net1.network)
pullup:always(type:ora.HR1.export.type)
STOP_DEPENDENCIES=hard(intermediate:ora.net1.network,uniform:intermediate:type:ora.HR1.export.type)
```

These dependencies ensure that the HAVIP is started after the ExportFS and ACFS resources, and is stopped before them.

There are several ways that we could start the exports:

» srvctl start exportfs -id <ID> - will start all exports attached to the HAVIP with the id <ID>

» srvctl start HAVIP -id <ID> - will start all exports attached to the HAVIP with the id <ID>

» srvctl start exportfs -name <NAME> - will start just the <NAME> ExportFS and its HAVIP

```
bash-3.2# /scratch/crs_home/bin/srvctl start exportfs -id HR1
bash-3.2# /scratch/crs_home/bin/srvctl status exportfs
export file system hr1 is enabled
export file system hr1 is exported on node host2
export file system hr2 is enabled
export file system hr2 is exported on node host2
bash-3.2# /usr/sbin/exportfs -v
/hr1
host1.us.oracle.com(rw,wdelay,no_root_squash,no_subtree_check,fsid=128850576,anonuid=65534,anongid=655
34)
/hr1
host2.us.oracle.com(rw,wdelay,no_root_squash,no_subtree_check,fsid=128850576,anonuid=65534,anongid=655
34)
/hr2
10.149.236.0/22(ro,wdelay,root_squash,no_subtree_check,fsid=1573414370,anonuid=65534,anongid=65534)
```

Here we can clearly see that the exports are exported to the proper clients with the proper options. Linux allows for multiple exports of the same file system or directories under the file system. Other OS NFS implementations have different implementations, and may restrict the export of parent and child directories. In Solaris, when exporting a snapshot, the parent (root) of the file system mount point will be exported.

Adding an HA-SMB Export

Now, let's add an HA-SMB export to this. First, we'll add a new mount on the same file system

[root@host1 bin]# ./srvctl add exportfs -path /hr1/smb -id HR1 -name smbexport -type SMB

Note that we have added it to the same HAVIP with our HA-NFS exports. Both exports will move around the cluster.

```
 [root@host1 bin]# ./srvctl status exportfs -id HR1
export file system hr1 is enabled
export file system hr1 is exported on node host2
export file system smbexport is enabled
export file system smbexport is not exported
```

Since the HAVIP already exists, we will need to explicitly start our new export. In the future, the export will start with the HA-NFS exports when the HAVIP starts up.

```
[root@host1 bin]# ./srvctl start exportfs -name smbexport
[root@n bin]# ./srvctl status exportfs -id HR1
export file system hr1 is enabled
export file system hr1 is exported on node host2
export file system hr2 is enabled
export file system hr2 is exported on node host2
export file system smbexport is enabled
export file system smbexport is exported on node host2
```

HA-SMB Samba Configuration

Once we have the HA-SMB export running, we can view the backend structure of the Samba configuration.

```
[root@host2 ~]# testparm -s
Load smb config files from /etc/samba/smb.conf
rlimit_max: increasing rlimit_max (8192) to minimum Windows limit (16384)
Processing section "[smbexport]"
Processing section "[homes]"
Processing section "[printers]"
Loaded services file OK.
Server role: ROLE_STANDALONE
[global]
        workgroup = MYGROUP
        server string = Samba Server Version %v
        log file = /var/log/samba/log.%m
        max log size = 50
        idmap config * : backend = tdb
        cups options = raw

[smbexport]
        path = /hr1/smb

[homes]
        comment = Home Directories
        read only = No
        browseable = No

[printers]
        comment = All Printers
        path = /var/spool/samba
        printable = Yes
        print ok = Yes
        browseable = No
```

Where is this information coming from? Each new export will generate a separate include file, which lives in /etc/samba/acfs/. It is included in a single new include file, acfsinc.conf, which is linked off the new smb.conf for the system.

Node Relocation

Now, let's say we need to do a node-relocation. The command for this is 'srvctl relocate HAVIP':

```
bash-3.2# /scratch/crs_home/bin/srvctl relocate HAVIP -id HR1 -node host1
bash-3.2# /scratch/crs_home/bin/srvctl status HAVIP -id HR1
HAVIP ora.hr1.HAVIP is enabled
HAVIP ora.hr1.HAVIP is running on node host1
```

Using any of the commands available to determine resource state (crsctl, srvctl), we can see that they are now running on the new node:

```
bash-3.2# /scratch/crs_home/bin/crsctl stat res -w "TYPE = ora.HR1.export.type"  -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.hr1.export
      1        ONLINE  ONLINE       host1                    STABLE
ora.hr2.export
      1        ONLINE  ONLINE       host1                    STABLE
--------------------------------------------------------------------------------
```

The same principle would hold true in the case of an unplanned outage – such as network failure, storage failure, or complete node failure.

Setup of HA-NFS V4 with Locking

Let's move forward and discuss the HA-NFS v4 w\Locking support. Introduced in 12.2.0.1, this is a new mode that operates slightly differently from the standard HA-NFS and HA-SMB modes. When this mode is in operation, all Oracle ACFS NAS|MAX exports will run on the same cluster node. This is due to the treatment of Locking by the underlying NFS server. In order to provide for failover and recovery semantics for these locks, the NFS server must implement a grace period for lock recovery time. Lock state is stored in a persistent location, and during startup of the server, this state is queried. During this time, new locks and operations are not granted (this is called the grace period and can be adjusted using OS parameters). When the state is read into memory and clients are given a chance to respond with their lock state, normal operation resumes.

On client nodes, mount the file system specifying NFS v4 as the NFS version. This prevents the server from defaulting to NFS v3, and enables support for the NFS v4 locking functionality. In order to enable Oracle ACFS HA-NFS with Locking, take the following steps:

1) Create a new volume for use with HANFS persistent storage.
   Restrictions on the Oracle ADVM volume include:
   1. No previously existing Oracle ACFS resource should exist for this new Oracle ADVM volume.
   2. No Oracle ACFS file system should exist on this Oracle ADVM volume.
   3. This Oracle ADVM volume should not be in use anywhere in the cluster.
2) Run the addnode operation:

```
bash-4.2# ./acfshanfs addnode -nfsv4lock -volume /dev/asm/hanfs-48
ACFS-9603: The script will do the following actions:
ACFS-9604:  - Update the operating system startup procedure so that NFS does not automatically
start.
ACFS-9605:    Management of the NFS daemons will be moved to Oracle Clusterware.
ACFS-9606:  - Format the volume: /dev/asm/hanfs-48.
ACFS-9607:  - Create an ACFS resource for the file system.
ACFS-9608:  - Mount the ACFS file system on '/var/lib/nfs'.
ACFS-9609:  Continue the installation? [1=yes,2=no]:


ACFS-9612: Stopping NFS Service.
Redirecting to /bin/systemctl stop  nfs.service
ACFS-9179: Command executed: '/sbin/service nfs stop', output = '0'
ACFS-9179: Command executed: '/install/usm161118/bin/srvctl start filesystem -device
/dev/asm/hanfs-48 -n host1.us.oracle.com', output = '2'
ACFS-9611: Starting NFS Service.
Redirecting to /bin/systemctl start  nfs.service
ACFS-9179: Command executed: '/sbin/service nfs start', output = '0'
ACFS-9203: true
```

3) Check the newly mounted NFS file system. It is mounted on the OS NFS directory so that all nodes in the cluster will see this information. Do not modify this new Oracle ACFS resource, or NFS locking functionality will cease to function.

```
bash-4.2# mount -t acfs
/dev/asm/hanfs-48 on /var/lib/nfs type acfs (rw,relatime,device,rootsuid,ordered)
```

4) Repeat for each node. Nodes added after the setup will have this step done automatically.

Once this is complete, we can take a look at the new resource – ora.netstorageservice. This resource is what manages the OS services for NFS. As the NFS exports move around the cluster, each machine will start and stop the NFS server. During start, the grace period will be triggered, allowing the OS NFS server to properly handle failover and lock recovery.

Since the OS NFS server is moving around the cluster and being restarted, it is not recommended to mix Oracle ACFS HANFS exports and non-ACFS exports. As the OS NFS server is restarted, access will be lost to the non-ACFS exports.

Our new netstorageservice resource is online:

```
NAME=ora.netstorageservice
TYPE=ora.netstorageservice.type
TARGET=ONLINE
STATE=ONLINE on host1
```

Now, let's add a new export. Here, the process is the same as before, using srvctl.

```
bash-4.2# ./srvctl add HAVIP -address host1-HAVIP1 -id HAVIP1
bash-4.2# ./srvctl add exportfs -name export1 -id HAVIP1 -path /mnt/oracle/tfa
bash-4.2# ./srvctl status exportfs
export file system export1 is enabled
export file system export1 is not exported
bash-4.2# ./srvctl start HAVIP -id HAVIP1
bash-4.2# ./srvctl status exportfs
export file system export1 is enabled
export file system export1 is exported on node host1
```

With this new Export, we can take a look at the dependencies. During conversion of the Oracle ACFS HA-NFS operation to enable locking, existing HA-NFS resources will be modified as well.

```
START_DEPENDENCIES=hard(ora.mgmthost1.tfarepos.acfs,ora.netstorageservice)
attraction(ora.HAVIP1.HAVIP) dispersion(type:ora.export.type)
pullup(ora.mgmthost1.tfarepos.acfs,ora.netstorageservice)
```

Note the existence of the dependency chain – this is what makes sure that the resources are running on the same node as the NFS server.

A more complex scenario

Now, imagine a more complex scenario. The admin has chosen 6 mount points for HANFS:

- /hr1 – 500Kb/s
- /hr – 10Mb/s – must be always available
- /source – 100Kb/s – must be always available
- /PDF – 10Mb/s
- /Games – 100Mb/s
- /Media – 1Mb/s

Due to high availability requirements, the best configuration for this would be:

- /hr1, /PDF and /Media configured on 1 HAVIP address
- /Games on 1 HAVIP address

» /source on 1 HAVIP address

» /hr on 1 HAVIP address

Rationale:

» Placing /Games on its own HAVIP address isolates its intense throughput from other HAVIPs, allowing CRS to potentially place this HAVIP and its associated ExportFS on its own server, away from the other HAVIPs. (Assuming you have enough servers in your cluster.)

» Placing /source on its own HAVIP address allows CRS to move it to a cluster member that can serve the file system, should there be a storage connection issue. Since there is only 1 ExportFS on this HAVIP, CRS needs only find a node where the ACFS file system is available, with no policy decision or trade-off necessary.

» Placing /hr on its own HAVIP allows for the same logic to apply to /hr as applies to /source.

» Placing the rest on their own HAVIP simplifies the number of IP addresses necessary. In the unlikely event that a different file system is available on each node of our cluster, CRS will place the HAVIP on the node that it determines is the best. This could cause 1 file system to be unavailable.

» The same considerations exist for both HA-SMB and HA-NFS exports.

CRS Policy Illustration – Choosing the Best Node

Consider the following cluster:

Node 1 – available file systems: /fs1 /fs2 /fs3
Node 2 – available file systems: /fs1 /fs3
Node 3 – available file systems: /fs2 /fs3 /fs4

If we consider a single HAVIP, exporting all 4 file systems, CRS will make a policy decision as to the best place to export the file systems from. No node truly satisfies the intent - all 4 file systems are available. So, CRS will determine that either Node 1 or Node 3 is the best place for our HAVIP and associated ExportFS. Either of these choices will result in 1 file system being unavailable due to storage connection issues.

Under what situations will the HAVIP and Exports move to other nodes ?

1. Server cluster membership change events (such as a node leaving or joining the cluster) will force the HAVIP to reevaluate their distribution, potentially moving the HAVIP and ExportFS resources around nodes. During this time, the client will see a momentary pause in service, but as soon as the export is reestablished (usually under 3s), the client will continue to operate as if there was no interruption in services.

2. In the rare event of a storage failure, resulting in a file system that is no longer accessible on a particular node of the server, the HAVIP will evaluate if a particular node is able to better provide for all its required file systems. If so, the HAVIP will move to that node, thus ensuring that the

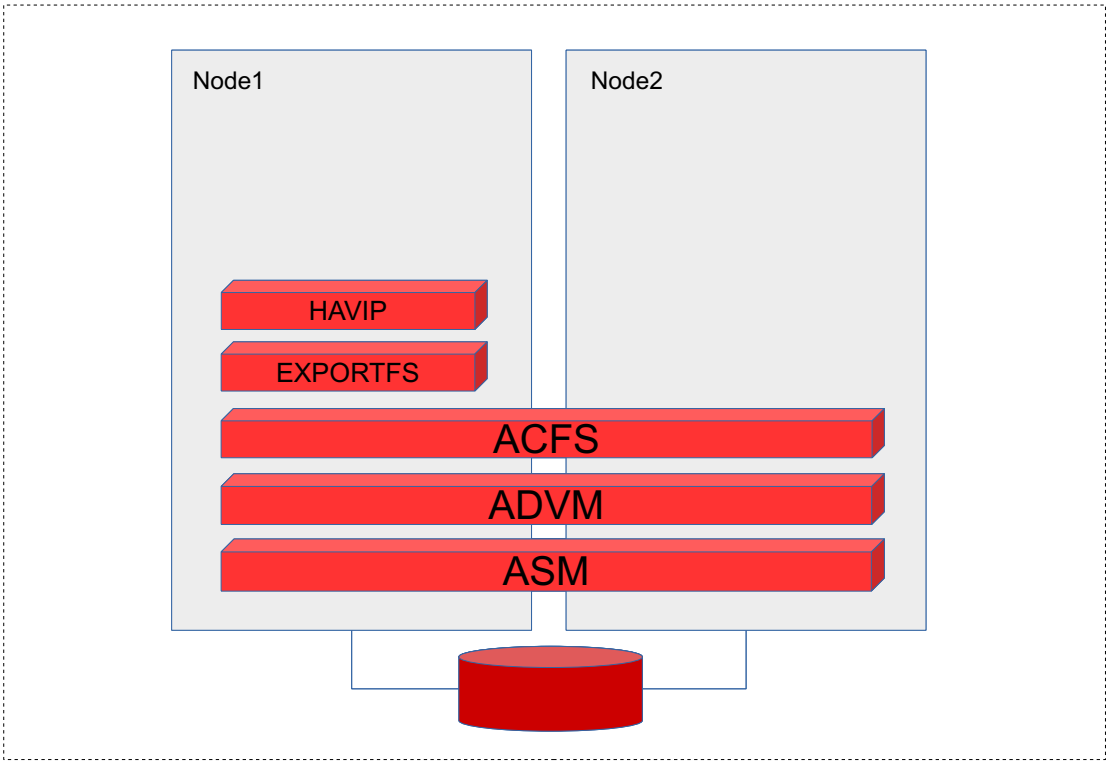
node with the maximum available file systems is where it is located. This ensures that the most clients will have access to the most file systems at any time.

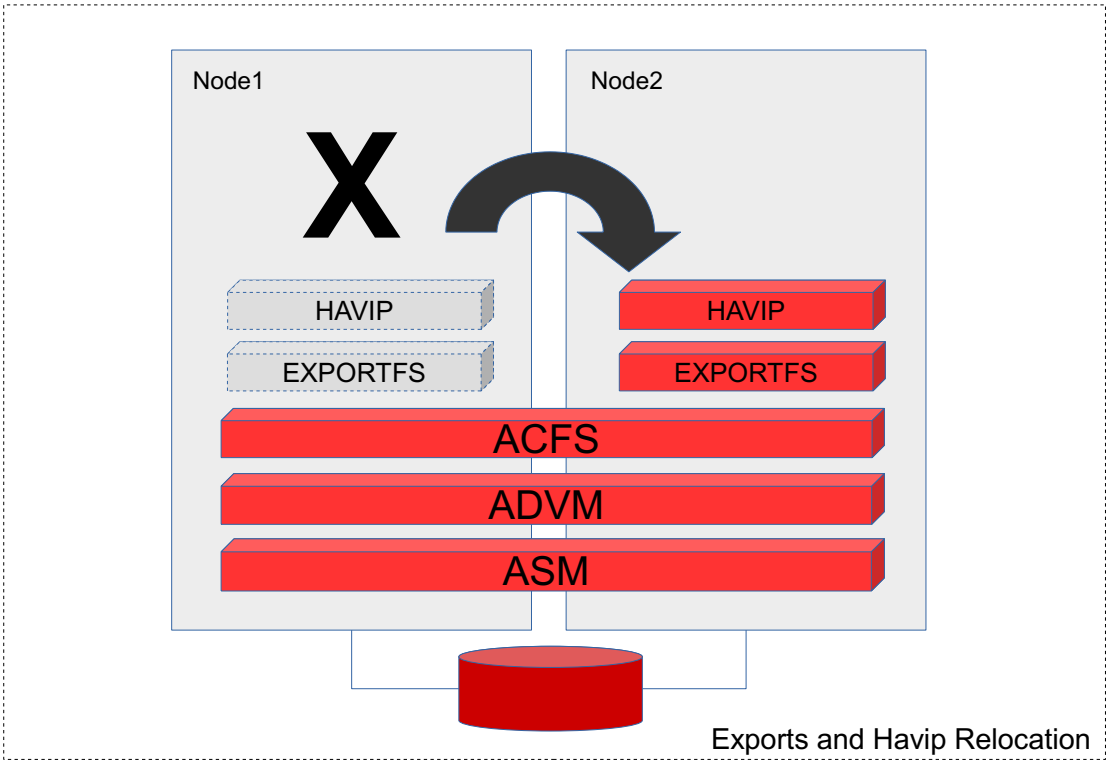
3. When the admin requests a move. Using the new 12.1 commands, an admin can do a planned relocation using the srvctl command, forcing the HAVIP and its associated ExportFS resources to move to another node of the cluster. This can be useful for planned node outages.

4. In the case of cluster member specific network connectivity issues, the cluster member will be removed from the cluster namespace, and the HAVIP with its associated ExportFS resources will move to a connected node.

5. If an HAVIP has a preferred node set, then the HAVIP will move back to that node when the node rejoins the cluster.

6. When using HA-NFS w\Locking, if the NFS server fails on the currently hosting node, the service will be restarted on another node. This can cause a longer pause than normal. In addition to the time taken to restart the resource elsewhere, the NFS server itself has a lock recovery timeout – usually 90s. This can be lowered (or raised) using the OS defaults for this, but care must be taken when doing this.





Exports and Havip Relocation

HAVIP Failover times

There are 2 cases to consider when considering node failover:

- Planned relocation
- Node failure

Planned Relocation

When the admin forcibly moves the HAVIP and associated ExportFS resources from node to node, the following steps are taken:

1) HAVIP on first node is shutdown.
2) Associated ExportFS resources on first node are shutdown.
3) Associated ExportFS resources are started on second node.
4) HAVIP is started on first node.

When an HAVIP is configured with a large number of associated ExportFS resources, the time taken for this planned failover may become large. Each individual ExportFS should take no more than 1 or 2 seconds to shutdown on Node 1. If there is sufficient processing power that CRS can stop all of them in parallel, then this may happen quickly. However, if CRS must sequentially stop each ExportFS, due to CPU processing limitations, the worst case scenario is 2s * <Number of ExportFS resources>.

The same scenario applies in reverse for startup time. Thus, worst case scenario TotalTimeToRelocate= (2s * <Number of ExportFS resources>) * 2 + 5s (where 5s is the time to stop and start the havip).

The exported file systems are unavailable until the HAVIP is started, which would happen at the end of this process.

So, while maintaining a single HAVIP may make management easier, it may increase the time for file systems to again begin processing and be available for clients.

Note that different versions of SMB may or may not be able to support HA Failover. The client may need to remount the file system.

Node Failure

Node failure is similar – although easier. From the time that CRS notices the node is gone, the only steps that will be taken are the HAVIP startup, which will happen after all associated ExportFS resources are started. Thus, time taken is ½ the relocation time.

NFS Performance Considerations

Since HA-NFS relies on NFS, standard NFS configuration and performance tuning is applicable to the Oracle ACFS HA-NFS product. Client options such as rsize and wsize may have a dramatic difference in the speed of data access. Additionally, the location of your file servers to your clients will affect NFS performance – best performance is usually gained by collocating NFS servers with your clients.

SMB Performance Considerations

Since HA-SMB relies on Samba (or SMB on Windows), standard configuration and performance tuning is applicable.

Please see https://www.samba.org/samba/docs/man/Samba-HOWTO-Collection/speed.html for Samba tuning.

Oracle ACFS NAS|MAX Scalability – Non-Locking NFS and SMB

Let's discuss for a second the performance of the Oracle RAC HANFS Cluster. Assume that the backed network fabric supports 1 Gb/s throughput. If each cluster member is connected to this network, a theoretical maximum throughput for a single machine is equal to 1 Gb/s * # of nodes in Oracle RAC HANFS Cluster. However, remember that CRS will move the HAVIP around, so if multiple HAVIPs are hosted on a single node, the maximum throughput is equal to (1 Gb/s * # of nodes in Oracle RAC HANFS Cluster) / (# of HAVIP on a single node).

We can quickly see some performance benefits to keeping the number of HAVIPs in a single cluster = # of nodes in the Oracle RAC HANFS Cluster, assuming that this meets our high availability needs (as discussed earlier). If each node hosts 2 HAVIPs, then we could double our Oracle RAC HANFS Cluster's combined throughput by simple doubling the number of nodes in the cluster.

Consider also the case where a single node is hosting 2 HAVIPs – and the performance of those HAVIPs exported file systems is 50% expected. Assuming that the backed storage and network fabric are appropriate for the expected throughput, we can remedy this situation by simply adding a new

node to the Oracle RAC HANFS cluster. This causes CRS to move the HAVIP in such a manner that each node now only hosts 1 HAVIP, instead of having 2 on one node.

Export FS Resource Behavior

- Alternative methods of starting the resource:
  - It is possible to start an export resource via Transparent High Availability, like many other resources. In this case, the ExportFS resource monitors whether or not the system reports that the file system is exported. Thus, it is possible to use the 'exportfs' (Linux) command to bring an export online. When this is done, the ExportFS resources may end up running on different nodes. This state is allowable, but not recommended. As soon as the associated HAVIP is started, the ExportFS resources will move to the node hosting that resource.

- Modifying Export Options using system tools:
  - When the admin uses a system tool to modify the export options (such as changing rw to ro, or modifying the samba config file), the ExportFS resource will reflect this via a 'PARTIAL' state. This tells the admin that the ExportFS is running, but the options that it was configured with are different than the currently exported options on that server.
- Stopping the ExportFS using THA:
  - If the admin manually removes an export via a command line tool such as exportfs, the associated ExportFS resource will also go offline.

- Removal of exported directory:
  - If the directory that is configured to be exported is removed via 'rm', the associated ExportFS will go offline.

Controlling the location of Exports in the Cluster

For the admin that prefers more control over the location of exports, an HAVIP can be configured to only run on certain nodes by use of the 'disable' and 'enable' commands.

For instance, assume that the admin wanted to only have HR2 and HR1 (exported over HAVIP HR1) running on 1 node of a 2 node cluster. After adding the resource, the admin could run the following to limit the resource:

```
bash-3.2# /scratch/crs_home/bin/srvctl disable havip -node host2 -id HR1
```

Further Thoughts

It will be left to the reader to apply other Oracle ACFS and Oracle ASM technology to the Oracle RAC HANFS product. One could imagine:

- Utilizing replication on an ACFS file system exported via Oracle ACFS NAS|MAX. This file system would be replicated to another data center. ACFS replication would provide for a consistent file system on the other side, allowing the address of the HAVIP to be changed to the address of an HAVIP on the other cluster. This would be a very fault-tolerant setup.

- Taking an ACFS snapshot of an ACFS file system exported via Oracle ACFS NAS|MAX. This would allow for a backup to be made.

- Exporting an Oracle Home over Oracle ACFS NAS|MAX, and ensuring that it was always available.

- Configuring an Oracle ACFS NAS|MAX ExportFS with Oracle ACFS Realms so that it was read-only during certain time periods to prevent unauthorized access.

- Using Oracle ACFS NAS|MAX with ASM disk groups configured with more than the default of 1 failure group to ensure that the underlying storage would be extremely fault tolerant. This would effectively remove the possibility of storage failure for a single disk group, while Oracle RAC HANFS would allow the export itself to be always available, creating a single extremely highly available file server.

Troubleshooting

As noted multiple times in this doc, ACFS NAS|MAX is highly dependent on the underlying configuration of the OS server. Most errors can be fixed by working with your system administrator and replicating using the basic OS functionality.

Common Issues:

- Server Not Responding

  Often, you will find that the server is missing the correct RPC daemons. This is encountered on the client with the message “RPC – Program not registered”. This is an OS dependent message, which may change from OS to OS.

  To fix this, ensure that the RPC daemons are running on the client.

- Common Debugging Tools

  - Rpcinfo – shows information on the rpc daemon on a particular server.

  - Showmount – shows the NFS exports from a particular server.

- Tcpdump – tcpdump will show the exact flow of traffic from node to node. This is useful for determining where things are going wrong – for instance, is the network dropping packets?

- Traceroute – ensure that a route is available to the server from the client.

- Testparm – used to verify Samba configuration files.

» Multiple NICs – The NFS server is a bit undirected – it will push packets over any interface that seems to have a route to the client\server. Thus, it is very important to ensure that the netstat routing table (netstat –rn) shows a route to the server\client over the interface that is being used.

- Additionally, the route –rn and route command can be used to ensure that the correct interface is being used to send traffic. NFS will use the default adapter for each address range. If one is not defined for the client server IP relationship, than an incorrect network adapter may be used, resulting in permission denied and other issues. 'ip route add' can be used to add a new route. As always, commands like this should be used with care.

» Permission Denied – perhaps the most common of issues. Information may be found in system logs (ie - /var/log/messages). Firewalls may be dropping packets, Security Configuration (like SELinux) may be limiting access, Permissions on directories may be incorrect, user equivalence mapping (such as root_squash) may be incorrect, clients may be requesting access over a network interface that does not have permission to access the share.

» The Powershell cmdlet **Get-SmbConnection** can be used to check which version of SMB the Server or Client is using. See https://blogs.technet.microsoft.com/josebda/2012/06/06/windows-server-2012-which-version-of-the-smb-protocol-smb-1-0-smb-2-0-smb-2-1-or-smb-3-0-are-you-using-on-your-file-server/

» In general, Google has a lot of results for NFS troubleshooting. One very good resource is: http://h41361.www4.hpe.com/tipnfs.html

Conclusion

This paper has journeyed through the configuration of Oracle ACFS NAS|MAX, beginning with a basic understanding of the technology involved – Oracle ACFS and Oracle Clusterware. Both of these components provide for a Maximum Availability Architecture that is useful in setting up highly available OS Protocols. Though these are just building blocks, Oracle ACFS NAS|MAX uses the Highly Available nature of these building blocks to wrap the underlying OS protocols and ensure their availability.

When these technologies are utilized with NFS and SMB, the system administrator is enabled to create a series of exports which can provide files to clients over the life of a single Oracle Grid Cluster even if one or more machines in the cluster are rebooted or unavailable. This setup can provide a low cost alternative to more expensive NAS devices that a system administrator might traditionally employ and with possibly greater availability.

As shown in the configuration section, support for these protocols is highly dependent on the OS itself running the latest and greatest software is always recommended, and specific packages or updates are often required of the system installation. Oracle ACFS NAS|MAX cannot guarantee the working of the system if the underlying protocols are incorrectly configured. In addition, when configuring an Oracle ACFS NAS|MAX installation, it is necessary to keep in mind whether certain features of NFS or SMB are necessary and to configure the system accordingly.

In addition to the Availability of Oracle ACFS NAS|MAX, additional features of Oracle ACFS are also available, which extend the workings of Oracle ACFS NAS|MAX. Additional information on these features is available in the Oracle ASM Administrators guide and in separate white papers. The combination of these features and the high availability makes for a truly integrated, fully featured solution.

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle

facebook.com/oracle

twitter.com/oracle

oracle.com

Integrated Cloud Applications & Platform Services

Oracle ACFS – NAS MAXIMUM AVAILABILITY EXTENSIONS
February 2019
Author: Allan Graves – Oracle ACFS Development Team