Recommended general database layouts and storage configurations

SnapManager Oracle

NetApp
August 30, 2024

Table of Contents

Recommended general database layouts and storage configurations

Knowing the recommended general database layouts and storage configurations can help you avoid issues related to disk groups, file types, and tablespaces.

- Do not include files from more than one type of SAN file system or volume manager in your database.

  All files making up a database must reside on the same type of file system.

- SnapManager requires a multiple of 4K block size.
- Include the database system identifier in the oratab file.

  Include an entry in the oratab file for each database to be managed. SnapManager relies on the oratab file to determine which Oracle home to use.

- If you want to register SnapManager backups with Oracle Recovery Manager (RMAN), you must create RMAN-enabled profiles.

If you want to leverage the new volume-based restore or full disk group restore, consider the following guidelines related to file systems and disk groups:

- Multiple databases cannot share the same Automatic Storage Management (ASM) disk group.
- A disk group containing data files cannot contain other types of files.
- The logical unit number (LUN) for the data file disk group must be the only object in the storage volume.

The following are some guidelines for volume separation:

- Data files for only one database must be in the volume.
- You must use separate volumes for each of the following file classifications: database binaries, data files, online redo log files, archived redo log files, and control files.
- You do not need to create a separate volume for temporary database files because SnapManager does not back up temporary database files.

Defining the database home with the oratab file

SnapManager uses the oratab file during operations to determine the Oracle database home directory. An entry for your Oracle database must be in the oratab file for SnapManager to work correctly. The oratab file is created during the Oracle software installation.

The oratab file resides in different locations based on the host operating system as shown in the following table:

Host operating system	File location
Linux	/etc/oratab

Host operating system	File location
Solaris	/var/opt/oracle/oratab
IBM AIX	/etc/oratab

The sample oratab file contains the following information:

```
+ASM1:/u01/app/11.2.0/grid:N   # line added by Agent
oelpro:/u01/app/11.2.0/oracle:N         # line added by Agent
# SnapManager generated entry          (DO NOT REMOVE THIS LINE)
smoclone:/u01/app/11.2.0/oracle:N
```

After Oracle is installed, you must ensure that the oratab file resides in the location specified in the previous table. If the oratab file does not reside in the correct location per your operating system, you must contact technical support for assistance.

Requirements for using RAC databases with SnapManager

You must know the recommendations for using Real Application Clusters (RAC) databases with SnapManager. The recommendations include port numbers, passwords, and authentication mode.

- In database authentication mode, the listener on each node that interacts with an instance of the RAC database must be configured to use the same port number.

  The listener that interacts with the primary database instance must be started prior to initiating a backup.

- In operating system authentication mode or an Automatic Storage Management (ASM) environment, the SnapManager server must be installed and running on each node in the RAC environment.
- The database user password (for example, for a system administrator or a user with the sysdba privilege) must be same for all the Oracle database instances in a RAC environment.

Requirements for using ASM databases with SnapManager

You must know the requirements for using Automatic Storage Management (ASM) databases with SnapManager. Knowing these requirements can help you avoid issues with the ASMLib, partitions, and clone specifications, among other things.

- SnapManager (3.0.3 or later) uses the new sysasm privilege available with Oracle 11gR2 instead of the sysdba privilege to administer an Oracle ASM instance.

  If you use the sysdba privilege to run administrative commands on the ASM instance, an error message is displayed. The database uses the sysdba privilege to access disk groups. If you connect to the ASM instance using the sysasm privilege, you have complete access to all the available Oracle ASM disk groups and management functions.

> If you are using Oracle 10gR2 and 11gR1, you must continue to use the sysdba privilege.

- SnapManager (3.0.3 or later) supports backing up databases that are stored directly on ASM disk groups when the disk group also contains an Automatic Cluster File System (ACFS) volume.

  These files are indirectly protected by SnapManager and might be restored with the remaining contents of an ASM diskgroup, but SnapManager (3.0.3 or later) does not support ACFS.

  > ACFS is a multiplatform, scalable file-system storage management technology available with Oracle 11gR2. ACFS extends ASM functionality to support customer files maintained outside the Oracle database.

- SnapManager (3.0.3 or later) supports the backup of files that are stored on ASM disk groups when the disk group also contains Oracle Cluster Registry (OCR) files or voting disk files; however, restore operations require slower, host-based or partial-file snap restore (PFSR) method.

  It is best to have OCR and voting disks on disk groups that do not contain database files.

- Each disk used for ASM must contain only one partition.
- The partition hosting the ASM data must be properly aligned to avoid severe performance problems.

  This implies that the LUN must be of the correct type and the partition must have an offset that is a multiple of 4K bytes.

  > For details about how to create partitions that are aligned to 4K, see the Knowledge Base article 1010717.

- ASM configuration is not specified as part of the clone specification.

  You must manually remove the ASM configuration information in clone specifications that were created using SnapManager 2.1 before upgrading the host to SnapManager (2.2 or later).

- SnapManager 3.1, 3.1p1, and 3.2 or later support ASMLib 2.1.4.
- SnapManager 3.1p4 or later support ASMLib 2.1.4, 2.1.7, and 2.1.8.

Supported partition devices

You must know the different partition devices that are supported in SnapManager.

The following table provides partition information and how it can be enabled for different operating systems:

Operating system	Single partition	Multiple partition	Non-partition devices	File system or RAW devices
Red Hat Enterprise Linux 5x or Oracle Enterprise Linux 5x	Yes	No	No	ext3*

Operating system	Single partition	Multiple partition	Non-partition devices	File system or RAW devices
Red Hat Enterprise Linux 6xor Oracle Enterprise Linux 6x	Yes	No	No	ext3 or ext4*
SUSE Linux Enterprise Server 11	Yes	No	No	ext3*
SUSE Linux Enterprise Server 10	No	No	Yes	ext3***
Red Hat Enterprise Linux 5x or lateror Oracle Enterprise Linux 5x or later	Yes	No	Yes	ASM with ASMlib**
SUSE Linux Enterprise Server 10 SP4or SUSE Linux Enterprise Server 11	Yes	No	Yes	ASM with ASMlib**
SUSE Linux Enterprise Server 10 SP4 or lateror SUSE Linux Enterprise Server 11	Yes	No	No	ASM without ASMlib**

For more information on the operating system versions supported, refer to the Interoperability Matrix.

Support for ASMLib

SnapManager supports different versions of ASMLib, although there are several factors you must consider when using SnapManager with ASMLib.

SnapManager supports ASMLib 2.1.4, 2.1.7, and 2.1.8. All SnapManager operations can be performed with ASMLib 2.1.4, 2.1.7, and 2.1.8.

If you have upgraded from ASMLib 2.1.4 to ASM 2.1.7, you can use the same profiles and backups created with ASMLib 2.1.4 to restore the backups and create the clones.

You must consider the following when using SnapManager with ASMLib:

- SnapManager 3.1 does not support ASMLib 2.1.7.

SnapManager 3.1p4 or later support ASMLib 2.1.4, 2.1.7, and 2.1.8.

- After performing a rolling upgrade from SnapManager 3.1 to 3.2, the backups created by using ASMLib 2.1.7 work only if the repository is rolled back to SnapManager 3.1 and ASMLib 2.1.7 is downgraded to ASMLib 2.1.4.
- After performing a rolling upgrade from SnapManager 3.1 to 3.2, backups created by using ASMLib 2.1.7 do not work if the repository is rolled back to SnapManager 3.1 with ASMLib 2.1.7.

  The rollback succeeds, but the profiles and backups cannot be used.

Support for ASM databases without ASMLib

SnapManager supports ASM without ASMLib, by default. The basic requirement is that the devices that are used for ASM disk groups must be partitioned.

When ASMLib is not installed, the device permissions related to ASM disk groups are changed to root:disk when you perform the following operations:

- Restart the host
- Restore a database from the primary storage by using volume-based SnapRestore (VBSR)
- Restore a database from the secondary storage

You can set the proper device permissions by assigning true to the oracleasm.support.without.asmlib configuration variable in smo.conf. The devices related to the ASM disk groups are added or removed from the initasmdisks file whenever new devices are added or removed from the host. The initasmdisks file is located at /etc/initasmdisks.

For example, if you set oracleasm.support.without.asmlib=true and then perform a backup mount, new devices are added to initasmdisks. When the host is restarted, the device permissions and ownership are maintained by the startup scripts.

The default value for oracleasm.support.without.asmlib is false.

Related information

Supported partition devices

Supported scripts

The asmmain.sh and asmquerydisk.sh scripts allow you to change the grid user, group, and the user, all of which are used to query the ASM disks. The scripts must always be executed from the root.

The asmmain.sh is the main script file called from any operation that adds or deletes devices. The asmmain.sh script calls another script internally, which needs to be executed from the root that has oracle grid credentials. This script queries the ASM disk group's devices, then adds those entries in the initasmdisk file with the permission and the ownership of the devices. You can change the permissions and ownership of this file based on your environment and the regex pattern that is used for matching only the /dev/mapper/*p1.

The asmquerydisk.sh script is used to query the disk list, which is used to create the ASM disk group. You must assign values to ORACLE_BASE, ORACLE_HOME, and ORACLE_SID, depending on your

configuration.

The scripts are located at /opt/NetApp/smo/plugins/examples/noasmlib. However, these scripts must be moved to /opt/NetApp/smo/plugins/noasmlib before starting the SnapManager for Oracle server on the host.

Limitations of using scripts to support an ASM database without ASMLib

You must be aware of certain limitations to using scripts to support an ASM database without ASMLib.

- The scripts provide an alternative solution for any kernel version, but only if ASMLib is not installed.
- The permissions for the scripts must be set in such a way that the scripts can be accessed by root, grid, oracle, or equivalent users.
- The scripts do not support restoration from a secondary location.

Deploying and running the scripts

You can deploy and run the asmmain.sh and asmquerydisk.sh scripts to support ASM databases without ASMLib.

These scripts do not follow the pre-scripts or post-scripts syntax and workflow is called when intitasmdisks is enabled. You can change anything related to your configuration settings in the scripts. It is recommended to verify if everything in the scripts are working as expected by performing a quick dry run.

> These scripts do not harm your system on failures nor will they impact your system. These scripts are executed to update the ASM-related disks to have proper permissions and ownership, so that the disks will always be under ASM instance control.

1. Create the ASM disk groups with the partitioned disks.
2. Create the Oracle database on the DISK GROUPS.
3. Stop the SnapManager for Oracle server.

   > In an RAC environment, you need perform this step on all the RAC nodes.

4. Modify the smo.conf to include the following parameters:

   a. oracleasm.support.without.asmlib = true
   b. oracleasm.support.without.asmlib.ownership = true
   c. oracleasm.support.without.asmlib.username = user name of your ASM instance environment
   d. oracleasm.support.without.asmlib.groupname = group name of your ASM instance environment

   These modifications set the permissions for the absolute path only, which means instead of partition device, permissions will be set only for dm-* device.

5. Modify the plugins scripts available in /opt/NetApp/smo/plugins/examples/noasmlib to include your configuration settings in the scripts.
6. Copy the scripts to /opt/NetApp/smo/plugins/noasmlib before starting the SnapManager for Oracle server on the host.
7. Navigate to the /opt/NetApp/smo directory and perform a dry run by running the following script: sh

plugins/noasmlib/asmmain.sh

The etc/initasmdisks file is created, which is the main file that is used.

You can confirm that the etc/initasmdisks file contains all the devices related to configured the ASM database, such as:

```
chown -R grid:oinstall /dev/mapper/360a98000316b61396c3f394645776863p1
         chmod 777 /dev/mapper/360a98000316b61396c3f394645776863p1
         chown -R grid:oinstall
/dev/mapper/360a980003754322f7a2b433469714239p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714239p1
         chown -R grid:oinstall
/dev/mapper/360a980003754322f7a2b433469714241p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714241p1
         chown -R grid:oinstall
/dev/mapper/360a980003754322f7a2b433469714243p1
         chmod 777 /dev/mapper/360a980003754322f7a2b433469714243p1
```

8. Start the SnapManager for Oracle server.
9. Configure SnapDrive for UNIX by adding the following to snapdrive.conf file.disconnect-luns-before-vbsr=on
10. Restart the SnapDrive for UNIX server.

> In an RAC environment, you need perform the Step 3 through Step 10 for all the RAC nodes.

The /etc/initasmdisks file created, must be executed from either one of the startup scripts or from a script that is newly defined in the rc3.d. The /etc/initasmdisks file should always be executed before the oracleha service starts.

Example

```
# ls -ltr *ohasd*
          lrwxrwxrwx 1 root root 17 Aug  7 02:34 S96ohasd ->
/etc/init.d/ohasd
          lrwxrwxrwx 1 root root 17 Aug  7 02:34 K15ohasd ->
/etc/init.d/ohasd
```

In the following example, sh -x/etc/initasmdisks will not be available by default, and you need to append it as the first line in the function `start_stack()` in an ohasd script:

```
start_stack()
{
sh -x /etc/initasmdisks
# see init.ohasd.sbs for a full rationale case $PLATFORM in Linux
}
```

Support for Oracle RAC ASM databases without ASMLib

If you are using Oracle RAC databases, the RAC nodes must be updated with the initasmdisks file whenever an operation is performed in the master RAC node.

If no authentication is required to log in into the RAC nodes from the master node, the asmmain.sh performs a secure copy (SCP) of initasmdisks to all the RAC nodes. The master node's initasmdisks file will be called whenever restore happens, and the asmmain.sh script can be updated to invoke the same script in all the RAC nodes.

The /etc/initasmdisks file created that must be executed from either one of the startup scripts or from a newly defined script in the rc3.d. The /etc/initasmdisks file should always be executed before the oracleha service starts.

Support for Oracle 10g ASM databases without ASMLib

If you are using Oracle 10g, the asmcmd command is not available for listing disks. You can use the sql query to obtain the disks list.

The disk_list.sql script is included in the existing scripts provided in the examples directory to support sql queries. When you execute theasmquerydisk.sh script, the disk_list.sql script must be executed manually. The example script lines are added with comments in the asmquerydisk.sh file. This file can either be placed in the /home/grid location or another location of your choice.

Sample scripts to support ASM databases without ASMLib

The sample scripts are available in the plugins/examples/noasmlib directory of the SnapManager for Oracle installation directory.

asmmain.sh

```
#!/bin/bash
griduser=grid
gridgroup=oinstall

# Run the script which takes the disklist from the asmcmd
# use appropriate user , here grid user is being used to run
# asmcmd command.
su -c "plugins/noasmlib/asmdiskquery.sh" -s /bin/sh grid
cat /home/grid/disklist

# Construct the final file as .bak file with propre inputs
```

```
awk -v guser=$griduser -v ggroup=$gridgroup '/^\/dev\/mapper/ { print
"chown -R "guser":"ggroup" "$1; print "chmod 777 " $1; }'
/home/grid/disklist > /etc/initasmdisks.bak

# move the bak file to the actual file.
mv /etc/initasmdisks.bak /etc/initasmdisks

# Set full full permission for this file to be called while rebooting and
restore
chmod 777 /etc/initasmdisks

# If the /etc/initasmdisks needs to be updated in all the RAC nodes
# or /etc/initasmdisks script has to be executed in the RAC nodes, then
the following
# section needs to be uncommented and used.
#
# Note: To do scp or running scripts in remote RAC node via ssh, it needs
password less login
# for root user with ssh keys shared between the two nodes.
#
# The following 2 lines are used for updating the file in the RAC nodes:
# scp /etc/initasmdisks root@racnode1:/etc/initasmdisks
# scp /etc/initasmdisks root@racnode2:/etc/initasmdisks
#
# In order to execute the /etc/initasmdisks in other RAC nodes
# The following must be added to the master RAC node /etc/initasmdisks
file
# from the asmmain.sh script itself. The above scp transfer will make sure
# the permissions and mode for the disk list contents are transferred to
the other RAC nodes
# so now appending any command in the /etc/initasmsdisks will be retained
only in the master RAC node.
# The following lines will add entries to the /etc/initasmsdisks file in
master RAC node only. When this script is executed
#  master RAC node, /etc/initasmdisks in all the RAC nodes will be
executed.
# echo 'ssh racnode1 /etc/initasmdisks' >> /etc/initasmdisks
# echo 'ssh racnode2 /etc/initasmdisks' >> /etc/initasmdisks
```

asmquerydisk.sh

```
#!/bin/bash
export  ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=/u01/app/grid/product/11.2.0.3/grid
export ORACLE_SID=+ASM
export  PATH=$ORACLE_HOME/bin:$PATH

# Get the Disk List and save this in a file called dglist.
asmcmd lsdsk  > /home/grid/disklist

# In oracle 10g the above used command 'asmcmd' is not available so use
SQL
# query can be used to take the disk list. Need to uncomment the following
# line and comment the above incase oracle 10g is being in use.
# The disk_list.sql script is availbe in this noasmlib examples folder
itself
# which can be modified as per customer needs.
# sqlplus "/as sysdba" @/home/grid/disk_list.sql > /home/grid/disklist
```

disk_list.sql

```
# su - oracle
-bash-4.1$ cat disk_list.sql
select path from v$asm_disk;
exit
-bash-4.1$
```

Requirements for using databases with NFS and SnapManager

You must know the requirements for using databases with Network File System (NFS) and SnapManager. The recommendations include running as root, attribute caching, and symbolic links.

- You must run SnapManager as root; SnapManager must be able to access the file systems that contain data files, control files, online redo logs, archive logs, and the database home.

  Set either of the following NFS export options to ensure that root can access the file systems:

  - root=host name
  - rw=host name, anon=0
- You must disable attribute caching for all the volumes that contain database data files, control files, redo and archive logs, and the database home.

  Export the volumes by using the noac (for Solaris and AIX) or actimeo=0 (for Linux) options.

- You must link the database data files from local storage to NFS to support symbolic links at the mount point-level only.

Sample database volume layouts

You can refer to sample database volume layouts for help in configuring your database.

Single-instance databases

File types	Volume names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_host name	Yes	On
Data files	oradata_sid	Yes	Off
Temporary data files	oratemp_sid	Yes	Off
Control files	oracntrl01_sid (Multiplexed) oracntrl02_sid (Multiplexed)	Yes	Off
Redo logs	oralog01_sid (Multiplexed) oralog02_sid (Multiplexed)	Yes	Off
Archive logs	oraarch_sid	Yes	Off

Real Application Clusters (RAC) databases

File types	Volume names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_host name	Yes	On
Data files	oradata_dbname	Yes	Off
Temporary data files	oratemp_dbname	Yes	Off
Control files	oracntrl01_dbname (Multiplexed) oracntrl02_dbname (Multiplexed)	Yes	Off

File types	Volume names	Dedicated volume for file types	Automatic Snapshot copies
Redo logs	oralog01_dbname (Multiplexed) oralog02_dbname (Multiplexed)	Yes	Off
Archive logs	oraarch_dbname	Yes	Off
Cluster files	oracrs_clustername	Yes	On

Single instance of an Automatic Storage Management (ASM) database

File types	Volume names	LUN names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_host name	orabin_host namelun	Yes	On
Data files	oradata_sid	oradata_sidlun	Yes	Off
Temporary data files	oratemp_sid	Oratemp_sidlun	Yes	Off
Control files	oracntrl01_sid (Multiplexed) oracntrl02_sid (Multiplexed)	oracntrl01_sidlun (Multiplexed) oracntrl02_sidlun (Multiplexed)	Yes	Off
Redo logs	oralog01_dbname (Multiplexed) oralog02_dbname (Multiplexed)	oralog01_dbnamelun (Multiplexed) oralog02_dbnamelun (Multiplexed)	Yes	Off
Archive logs	oraarch_sid	Oraarch_sidlun	Yes	Off

ASM RAC databases

File types	Volume names	LUN names	Dedicated volume for file types	Automatic Snapshot copies
Oracle binaries	orabin_host name	orabin_host namelun	Yes	On
Data files	oradata_sid	oradata_sidlun	Yes	Off

File types	Volume names	LUN names	Dedicated volume for file types	Automatic Snapshot copies
Temporary data files	oratemp_sid	Oratemp_sidlun	Yes	Off
Control files	oracntrl01_sid (Multiplexed) oracntrl02_sid (Multiplexed)	oracntrl01_sidlun (Multiplexed) oracntrl02_sidlun (Multiplexed)	Yes	Off
Redo logs	oralog01_dbname (Multiplexed) oralog02_dbname (Multiplexed)	oralog01_dbnamelun (Multiplexed) oralog02_dbnamelun (Multiplexed)	Yes	Off
Archive logs	oraarch_sid	Oraarch_sidlun	Yes	Off
Cluster files	oracrs_clustername	oracrs_clusternamelun	Yes	On