



Support for Oracle 10gR2 and 11gR1 ASM with Serviceguard for Linux (SG/LX)

1	Executive Summary .....	3 -
2	Technical Introduction .....	3 -
3	Terms and Definitions .....	4 -
4	ASM Background .....	5 -
4.1	What is Automatic Storage Management (ASM)? .....	5 -
5	Serviceguard for Linux (SG/LX) ASM support .....	7 -
5.1	Overview .....	7 -
5.2	ASM over LVM vs. ASM over Raw devices .....	7 -
5.3	Configuring LVM Volume Groups for ASM Disk Groups .....	8 -
5.4	Sample Command Sequence for Configuring LVM Volume Groups .....	10 -
5.5	Configuring Raw Devices for ASM Disk Groups .....	11 -
5.6	Sample sequence for configuring raw devices for ASM Disk Groups .....	11 -
6	Toolkit changes for supporting ASM in SG/LX .....	13 -
6.1	Framework for ASM support with Serviceguard .....	13 -
6.2	How the Toolkit Starts/Stops/Monitors the ASM instance .....	13 -
6.3	How the Toolkit Starts/Stops/Monitors the Database instance .....	14 -
7	Serviceguard Toolkit Internal File Structure .....	17 -
8	ASM File Descriptor Release .....	19 -
9	Installation, Configuration and Troubleshooting .....	20 -
10	Related Documentation .....	21 -

# 1 Executive Summary

This document discusses the use of the Oracle database server feature called Automatic Storage Management (ASM) in configurations of Serviceguard for Linux (SG/LX) for single database instance failover.

The purpose of this document is to describe how the Serviceguard for Linux Oracle Toolkit now supports deployment over ASM. For Oracle single instance failover, Serviceguard support is available for configurations with, ASM over LVM and ASM over raw devices.

The framework for ASM integration with Serviceguard makes use of a Multi-Node Package (MNP) to encapsulate the ASM instance and to have the ASM instance running on all nodes, with one or more Oracle single instance failover packages dependent on this MNP. Customers wishing to use the proposed framework for ASM integration with Serviceguard should use the bundle of Linux Oracle Toolkit scripts, provided by HP. This whitepaper helps in configuring the ASM framework in a Serviceguard environment.

## 2 Technical Introduction

For Oracle single instance failover, Serviceguard support is available for configurations with, ASM over LVM and ASM over raw devices. In ASM over LVM the members of the ASM disk groups are raw logical volumes managed by LVM. LVM provides the necessary I/O fencing mechanism for a failover package via a feature in the volume manager - exclusive activation of volume groups based on LVM2 hosttags. The corresponding storage driver or other multipathing solutions like device mapper provide the multipathing capability. In ASM over raw devices, the Serviceguard A.11.19 for Linux feature "Persistent Reservation over raw devices", supported only in modular package configurations provides the necessary I/O fencing mechanism.

Oracle has released two patches<sup>1</sup> to address the ASM file descriptor issue. Oracle 10.2.0.4 and Oracle 11.0.6 ASM instances, kept file descriptors open on ASM disk group members even after the disk groups had been dismounted. To address this issue, Oracle has released two patches. Hence, without the Oracle patches, it was not possible for Serviceguard to support Oracle single instance failover when ASM is used.

Prior to Serviceguard A.11.19 for Linux, Serviceguard did not support persistent reservation on raw devices. Starting from Serviceguard A.11.19 on Linux release, modular package configurations can make persistent reservation on raw devices on Linux. This feature is made available only on Linux.

---

<sup>1</sup> Oracle 10.2.0.4 and 11.1.0.6 patches 7330611 and 7225720 need to be installed on all nodes for successful database instance failover.

### 3 Terms and Definitions

Term	Definition
ASM	<i>Automatic Storage Management</i>
DM	<i>Device Mapper</i>
LV	<i>Logical Volume</i>
LVM	<i>Logical Volume Manager</i>
MNP	<i>Multi-Node Package</i>
OEM	<i>Oracle Enterprise Manager</i>
PR	<i>Persistent Reservation</i>
PV	<i>Physical Volume</i>

## 4 ASM Background

### 4.1 What is Automatic Storage Management (ASM)?

Automatic Storage Management is a feature provided in Oracle 10g and 11g to simplify the database files management. ASM provides file system and volume management capabilities directly inside the Oracle database kernel, allowing volumes and disk management with familiar SQL statements in Oracle. This is an alternative to platform file systems and volume managers for the management of most file types used to store the Oracle database, including Oracle datafiles, control files, and online and archived redo log files. File types not supported by ASM include Oracle database server binaries, trace files, audit files, alert logs, backup files, export files, tar files, core files. Storage for application binaries and data cannot be managed by ASM.

ASM uses disk groups to store datafiles; an ASM disk group is a collection of disks that ASM manages as a unit. Within a disk group, ASM exposes a file system interface for Oracle database files. The contents of files that are stored in a disk group are evenly distributed, or striped to eliminate hot spots and to provide uniform performance across the disks.

Figure 1. Oracle database storage hierarchy without and with ASM

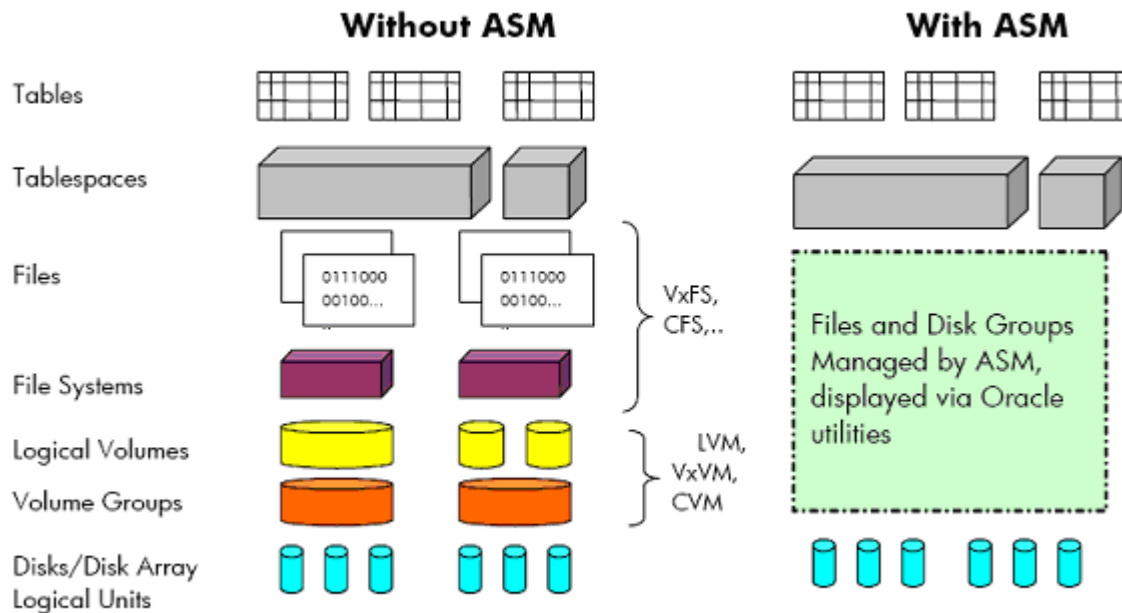


Figure 1 contrasts the Oracle storage hierarchy as it appears when platform or 3rd party volume managers and file systems are used for Oracle data files, compared to when ASM is used. The layers corresponding to file systems and volume managers are absorbed into ASM. The files and directories in the storage hierarchy are not visible using standard operating system commands; to display them the customer must use Oracle utilities.

The major advantage of ASM is the ease of management it provides for database files

- The system administrator has only to specify the set of raw devices to be used in an ASM disk group; the tasks of configuring and administering volume/disk groups and file systems are eliminated. Oracle ASM makes use of the Oracle feature called Oracle-Managed Files and performs the tasks of creating, deleting and extending files on behalf of database instances; additionally, it manages their mirroring and striping.
- If a device is added to, or deleted from, an ASM disk group, ASM automatically rebalances database file striping based on the new disk layout.

## 5 Serviceguard for Linux (SG/LX) ASM support

### 5.1 Overview

ASM in Oracle Database versions 10.2.0.4 and 11.1.0.6 is supported for Serviceguard for Linux (SG/LX) versions A.11.18 and A.11.19 on the x86\_64 architecture only. Please note that for database failover, each database should store its data in its own disk group. Refer to Oracle documentation, mentioned in related documentation, for limitation imposed by ASM on the number of disk groups.

ASM over raw devices requires the Serviceguard feature "Persistent Reservation over raw devices" to ensure I/O fencing. However, Serviceguard provides this feature only in the modular package configurations on the Linux platform, starting from Serviceguard A.11.19.

For databases managed using Serviceguard A.11.18 and Serviceguard A.11.19 legacy package configurations, the disk group members must be raw logical volumes managed by Logical Volume Manager (LVM). For databases managed using Serviceguard A.11.19 modular package configurations, the disk group members can either be raw devices or raw logical volumes managed by LVM.

### 5.2 ASM over LVM vs. ASM over Raw devices

In Serviceguard A.11.18 and A.11.19 legacy style package configurations, ASM disk group members are required to be raw logical volumes managed by LVM. In Serviceguard A.11.19 modular style package configurations, ASM disk group members can be raw devices or raw logical volumes managed by LVM. Existing capabilities of the storage drivers or other multipathing solutions like device mapper are leveraged to provide multipathing for LVM logical volumes.

It is the responsibility of Serviceguard to provide I/O fencing when it fails over for a single instance Oracle database instance. For ASM over raw logical volumes, Serviceguard ensures I/O fencing for a failover package via a feature in the volume manager - exclusive activation of volume groups based on LVM2 hosttags.

Note: Starting from release A.11.16.07, Serviceguard for Linux provides functionality \*similar to\* HP-UX exclusive activation. This feature is based on LVM2 hosttags, and is available only for Linux distributions that officially support LVM2.

For ASM over raw devices, Serviceguard A.11.19 for Linux feature (only in modular package configurations) "persistent reservation over raw devices" ensures I/O fencing for a failover package.

The advantages of the "ASM-over-LVM" configuration are as follows:

- ASM-over-LVM ensures that the devices used for disk group members will have the same names (the names of logical volumes in LVM volume groups) on all nodes, easing ASM configuration.

- ASM-over-LVM protects ASM data against inadvertent overwrites from nodes inside/outside the cluster.

The disadvantages of the ASM-over-LVM configuration are as follows:

- Additional configuration and management tasks are imposed by the extra layer of volume management (administration of volume groups, logical volumes, and physical volumes).
- There is a small performance impact from the extra layer of volume management.

Advantages of ASM over Raw devices configuration are as follows:

- No additional storage configuration and management tasks are there for the user. The user can use the raw devices as they are.
- Database performance is faster in ASM over raw devices than in ASM over raw logical volumes.

Disadvantages of ASM over Raw devices configuration are as follows:

- ASM over raw device requires the user to configure raw devices so that the raw devices have the same names on all the nodes in the cluster. The user can do one of the following to ensure this
  - Create udev devices
  - Create device mapper devices or aliases and use udev along with device mapper
- Raw devices do not have any protection against inadvertent overwrites from nodes inside/outside the cluster. This requires the use PR (Persistent Reservation) on raw devices, supported by Serviceguard A.11.19 modular style package configurations on Linux.

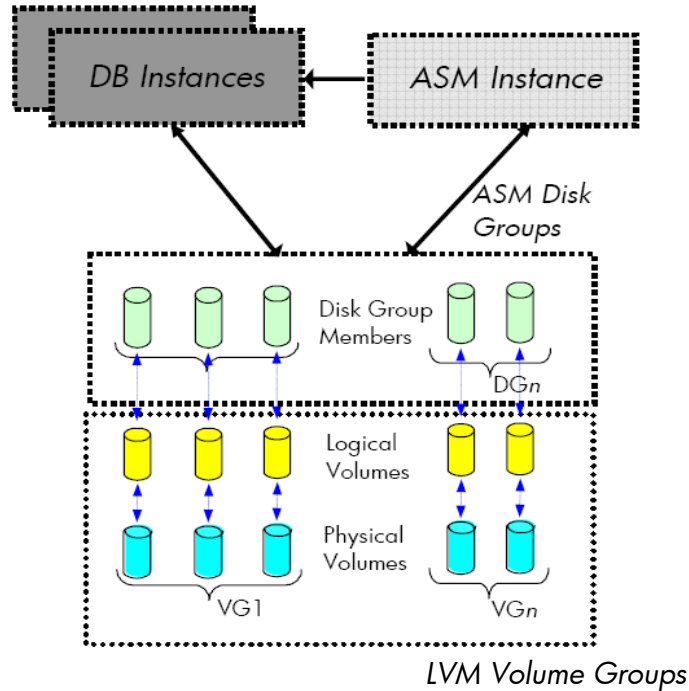
## 5.3 Configuring LVM Volume Groups for ASM Disk Groups

For Serviceguard A.11.18 configurations and Serviceguard A.11.19 legacy style package configurations, ASM disk group members need to be raw logical volumes managed by LVM. These logical volumes presented to ASM are desired to resemble raw disks, as far as possible. Hence, each LVM logical volume (LV) used as a member of an ASM disk group is required to be laid out to occupy the usable space, in contiguous fashion, of exactly one single physical volume (PV). This implies that the LV

- should be contiguous
- should not be striped or mirrored
- should not span multiple PVs

- should not share a PV with other LVs.

Figure 2. Mapping between LVM logical and physical volumes for ASM configuration



The idea is that ASM provides the mirroring, striping, slicing and dicing functionality as needed. The corresponding storage driver or other multipathing solutions like DM, supplies the multipathing functionality not provided by ASM. Figure 2 indicates the 1-1 mapping between LVM PVs and LVs used as ASM disk group members.

The PVs used in an ASM disk group can be organized into LVM volume groups as desired by the customer. In the example shown in Figure 2, for each ASM disk group, the PVs corresponding to its members are organized into a separate LVM volume group.

The LVM volume groups are marked as exclusive volume groups. As noted above, multiple physical paths to each physical volume should be configured using the LVM PV Links feature or a separate multipathing product.

## 5.4 Sample Command Sequence for Configuring LVM Volume Groups

In this section, examples of a command sequence that can be used to prepare LVM Logical Volumes for use by ASM to meet the requirements specified above are provided. The scenario for the example is that, new volume group named *vgpkgA* and *vgpkgB* with three physical paths is being prepared. The physical paths are */dev/cciss/c0d1p1*, */dev/cciss/c0d2p1* and */dev/cciss/c0d3p1*.

1. Create LVM physical volumes on each LUN.

```
# pvcreate -f /dev/cciss/c0d1p1
# pvcreate -f /dev/cciss/c0d2p1
# pvcreate -f /dev/cciss/c0d3p1
```

Create separate volume groups for each Serviceguard package. In the following example, the LUNs */dev/cciss/c0d1p1* and */dev/cciss/c0d2p1*, which are defined earlier as physical volumes, are added to volume group *vgpkgA* and */dev/cciss/c0d3p1* is added to *vgpkgB*:

```
# vgcreate /dev/vgpkgA /dev/cciss/c0d1p1 /dev/cciss/c0d2p1
# vgcreate /dev/vgpkgB /dev/cciss/c0d3p1
```

2. For each of the two VGs, create a corresponding LV

- Create an LV of zero length
- Mark the LV as contiguous
- Extend each LV to the maximum size possible on that PV (the number of extents available in a PV can be determined via `vgdisplay -v <vgname>`)

The following creates a 500 Megabyte logical volume named */dev/vgpkgA/lvol1* and a one Gigabyte logical volume named */dev/vgpkgA/lvol2* in volume group *vgpkgA*

```
# lvcreate -L 500M vgpkgA
# lvcreate -L 1G vgpkgA
```

3. Ensure VG exclusive activation using hosttags

- Edit */etc/lvm/lvm.conf* and add the following line:  
tags { hosttags = 1 }

- Create the file */etc/lvm/lvm\_\$(uname -n).conf* and add the following line to the file  
activation { volume\_list=["@node"] } where *node* is the value of `uname -n`.

- Run `vgscan`  
# vgscan

The disk partitions will be visible on other nodes as soon as those nodes are rebooted. After distributing the LVM configuration to all the cluster nodes, LVM commands can be used to switch volume groups between nodes. (To avoid data corruption, a given volume group must be active on only one node at a time).

4. Mark the volume group as exclusive, as specified by Serviceguard documentation.
5. Now create the raw bindings to the logical volume. Assign the right set of ownerships and access rights to the raw bindings on each node as required by Oracle. Udev rules can be written to retain the permission across reboots.

The raw devices can now be used as disk group members when configuring ASM disk groups using the Oracle database management utilities. There are a few ways to configure the ASM disk groups, e.g., the *dbca* database creation wizard and *sqlplus*.

The same command sequence can be used for adding new disks to an existing volume group that is being used by ASM to store one or more database instances.

## 5.5 Configuring Raw Devices for ASM Disk Groups

For Serviceguard A.11.19 modular package configurations, ASM disk group members in Serviceguard configurations can be raw logical volumes managed by LVM or raw devices.

Raw devices as such can be used for ASM disk groups. However, the raw device names need to be persistent across reboot. Hence, it is recommended to configure device mapper or udev names for the raw devices. Also, ASM requires the right set of ownership and access rights on the raw devices on each node. Device mapper devices do not retain their permission across reboot and hence device mapper needs to be used in conjunction with udev to set the right ownership and access rights to the raw devices.

To configure device mapper names for raw devices

- Install the device-mapper and device-mapper-multipath rpms on the system.
- Create the device mapper names for raw devices.
- Create the raw device bindings to the device mapper devices.
- Write udev rules to set the ownership, rights for the raw device bindings.
- Specify `ASM_DISKSTRING` to be `"/dev/raw/raw*"` and create ASM disk groups.

## 5.6 Sample sequence for configuring raw devices for ASM Disk Groups

1. Install the device-mapper and device-mapper-multipath rpms on the system. After the installation, enable the multipathd service, so that the service multipathd is run on reboot.
2. Create the device mapper devices.

The user can directly create udev devices. It is suggested to use device mapper devices.

- Create device mapper devices with the WWID of the device

This will create the device mapper devices under /dev/mapper. The device mapper devices will be of the form /dev/mapper/<WWID>

- Create user friendly device mapper names,

Edit the multipath configuration and set the user\_friendly\_name to yes. Edit the bindings file and ensure that the user friendly names refer to the same device mapper device on all the nodes in the cluster.

- Create user friendly aliases for devices

Edit the multipath configuration file to create aliases.

Device mapper creates two paths for each device – “/dev/dm-\*” and “/dev/mapper/mpath\*”. Device mapper device of the form /dev/dm\* are not persistent and hence use the device mapper names “/dev/mapper/mpath\*” in the raw bindings configuration.

3. The raw devices can be used as such for the ASM disk groups. However, ASM requires the right set of ownership and access rights on the raw devices on each node. Hence, it is recommended to configure raw device bindings for the device mapper names and use the udev rules to set their permissions and access rights.
4. Set the ASM\_DISKSTRING to “/dev/raw/raw\*”

## 6 Toolkit changes for supporting ASM in SG/LX

Support for Oracle 10.2.0.4 and 11.1.0.6 ASM will be on Serviceguard A.11.18 and A.11.19 for the x86\_64 architecture only. Please note that for database failover, each database should store its data in its own disk group. Refer to Oracle documentation, mentioned in the related documentation section, for limitation imposed by ASM on the number of disk groups. The disks must either be managed by LVM or the package must use "PR for raw devices on Linux" to ensure I/O fencing.

### 6.1 Framework for ASM support with Serviceguard

The framework for ASM integration with Serviceguard makes use of a Multi-Node Package (MNP) to encapsulate the per-node ASM instances, with one or more Oracle single instance failover packages dependent on this MNP. This configuration enables the database instance to start up in the right order in relation to the ASM instance, and in the event of failover, to relocate to a node where an ASM instance is available. The benefits of this package framework for single instance Oracle database using ASM-based storage are:

- Avoid failing over ASM instance. This reduces the failover time due to ASM instance restart.
- Many database instances can share one ASM instance per node.
- Sequencing between Oracle database instance and ASM instance during startup/shutdown/failover is also taken care of. Otherwise users need to start the ASM instance, mount the disk groups and then start the database instance. Now that the ASM instance is always running, users just need to mount the ASM disk groups and start the database instance.
- Flexibility of the database instances to fail over independently.

HP provides a set of scripts known as the Linux Oracle Toolkit scripts which provides for the integration of ASM with Serviceguard. The operation of the Toolkit is described in the following sections.

### 6.2 How the Toolkit Starts/Stops/Monitors the ASM instance

The Toolkit MNP for the ASM instance provides start and stop functions for the ASM instance and has a service for checking the status of the ASM instance.

The start function executes `su` to the ASM software owner user id. It starts the ASM instance specified by the user using the `sqlplus` commands. ASM disk groups associated with a database instance will be mounted before database instance startup and after volume group activation using `hosttags` during the database package startup.

The stop function executes *su* to the ASM software owner user id. It stops the ASM instance using *sqlplus* commands.

The monitor function checks the health of the ASM instance processes continuously. If the monitor function finds that any process has died, it means that the ASM instance has either failed or been inappropriately shut down, i.e., without using *cmhaltpkg*. The service that invokes the function fails at this point and Serviceguard package manager fails the corresponding ASM MNP instance.

On ASM instance failure, all dependent database instances will be brought down and will be started on the adoptive node.

## 6.3 How the Toolkit Starts/Stops/Monitors the Database instance

The Toolkit failover package for the database instance provides start and stops functions for the database instance and has a service for checking the status of the database instance.

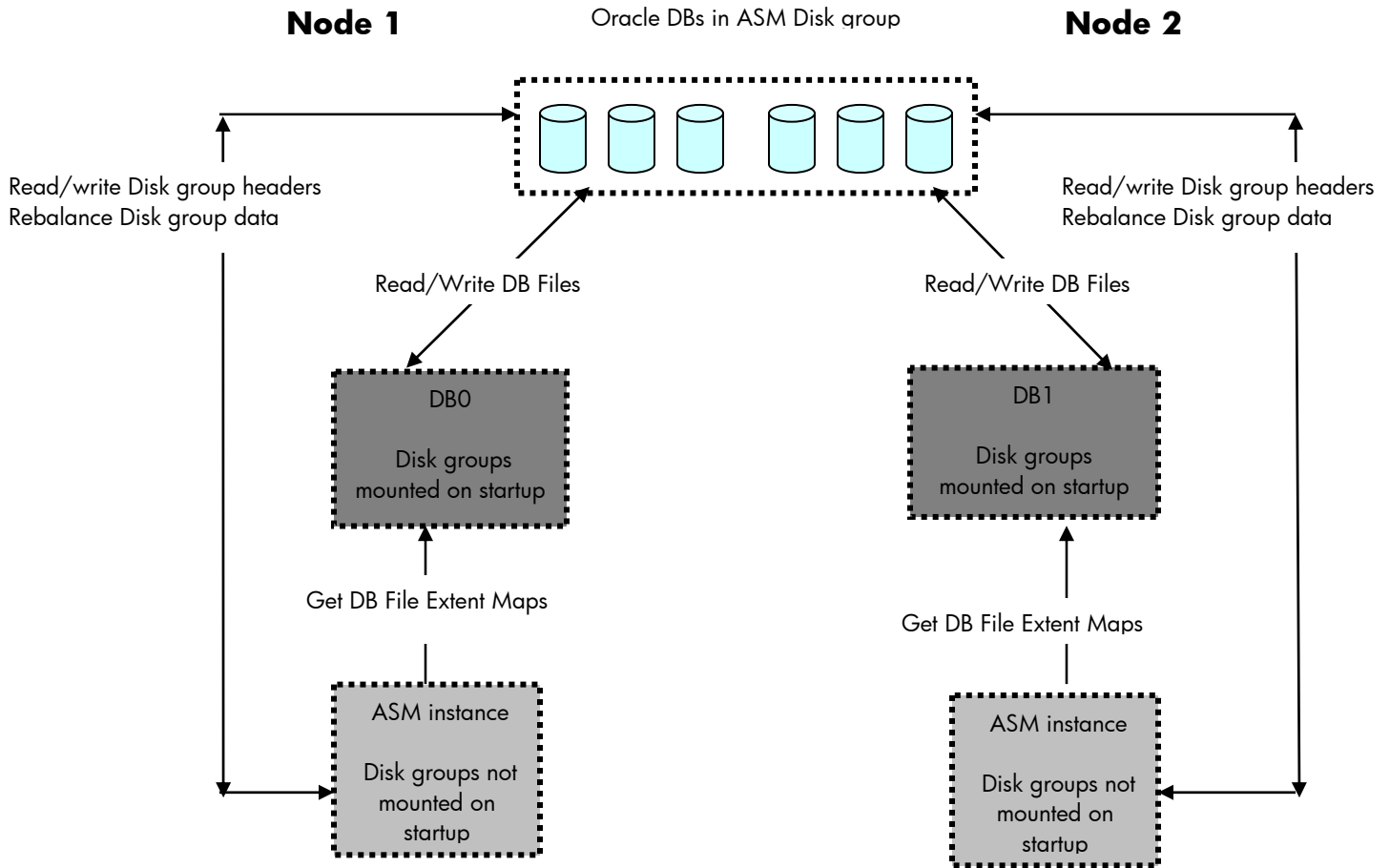
There will be a separate package for each database instance. Each database package has a simple dependency on the ASM package. If the database uses raw logical volumes, the package activates the volume groups in exclusive mode using *hosttags*. If the database uses raw devices, the package makes a persistent reservation on the raw devices. The start function executes *su* to the Oracle software owner user id. This function mounts the ASM disk groups associated with the instance and starts the database instance specified by the user using *sqlplus* commands.

The stop function executes *su* to the Oracle software owner user id. The database instance is shutdown using the *sqlplus* commands and the disk groups associated with the instance are dismounted. After dismounting the disk groups, the logical volumes/raw devices are checked to see if ASM has closed its file descriptors. If file descriptors are still open, necessary steps are taken to ensure that file descriptors are closed. The package then deactivates the volume groups using *hosttags*, if the database uses raw logical volumes.

The monitor function contains a continuous loop to check if the database instance processes specified are healthy. If the monitor function finds that any process has died, it means that the database instance has either failed or been inappropriately shut down, i.e., without using *cmhaltpkg*. The service that invokes the function fails at this point and the Serviceguard package manager fails over the corresponding database package.

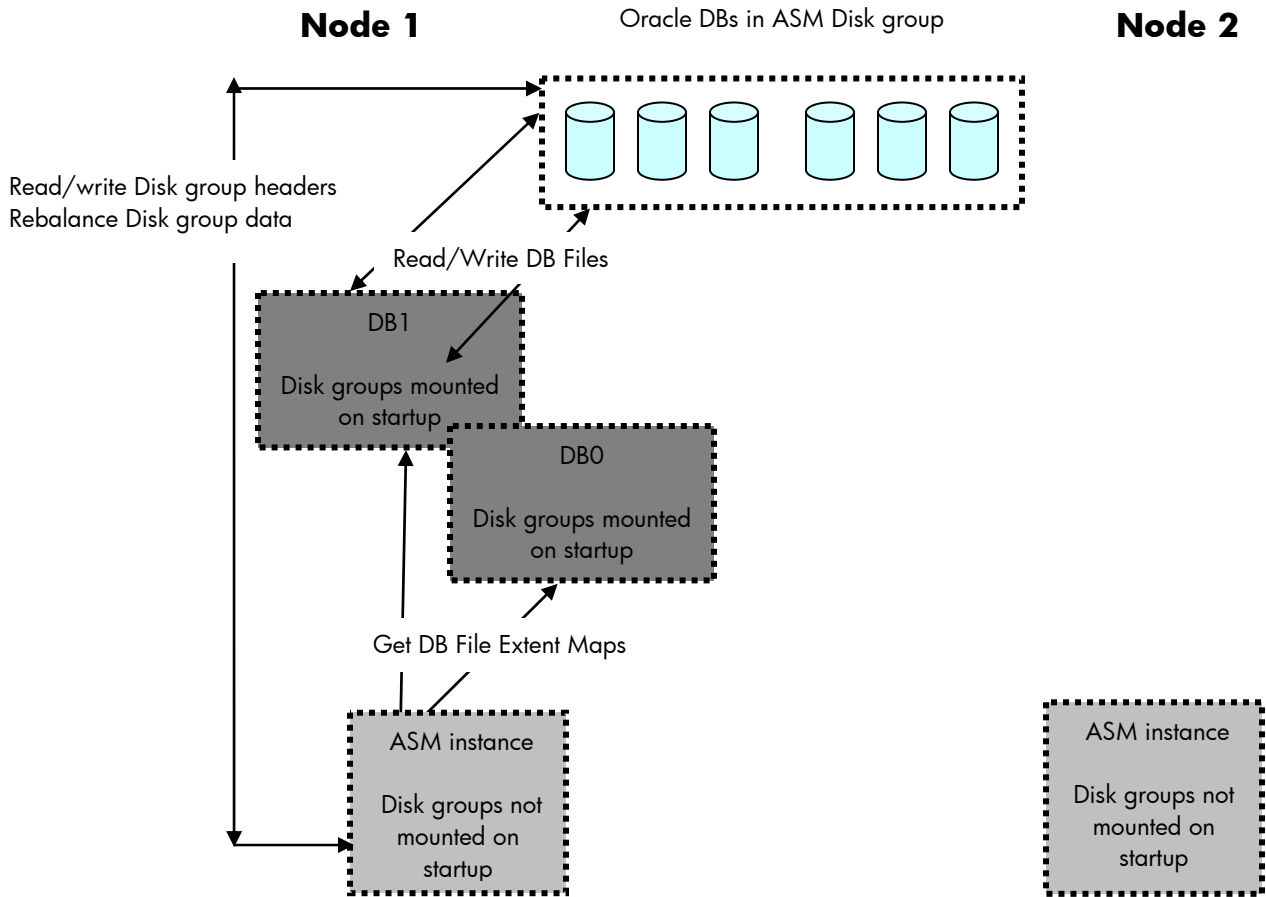
Consider two database instances DB0 and DB1. Each database instance uses its own disk group with the disk group members on its own volume group. Figure 4 depicts the scenario when the ASM instance and the database instances DB0 and DB1 start on nodes 1 and 2 respectively

Figure 4 Serviceguard ASM environments



The diagram below shows the scenario when DB1 fails on node 2.

Figure 5 ASM environments when DB1 fails on node 2

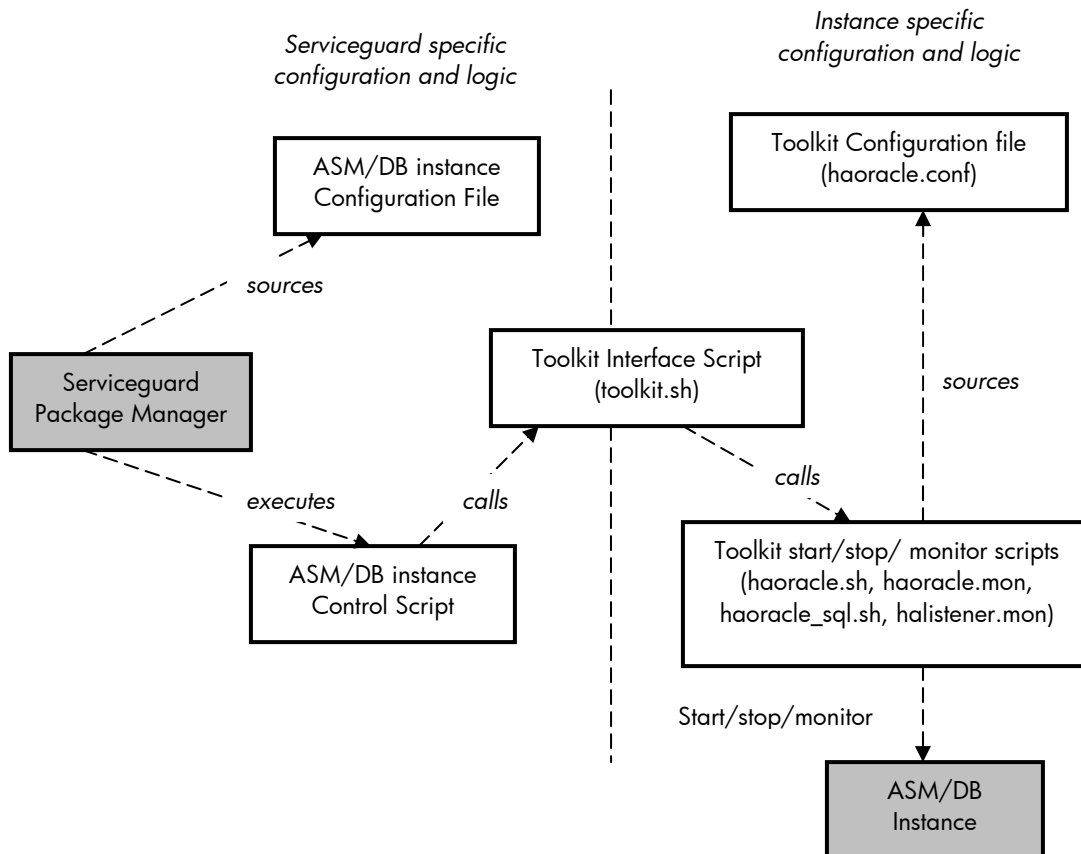


## 7 Serviceguard Toolkit Internal File Structure

HP provides a set of scripts for the framework proposed for ASM integration with Serviceguard. The Linux Oracle toolkit scripts contain the instance specific logic to start/stop/monitor both the ASM and the database instance. These scripts support both legacy and the modular method of packaging.

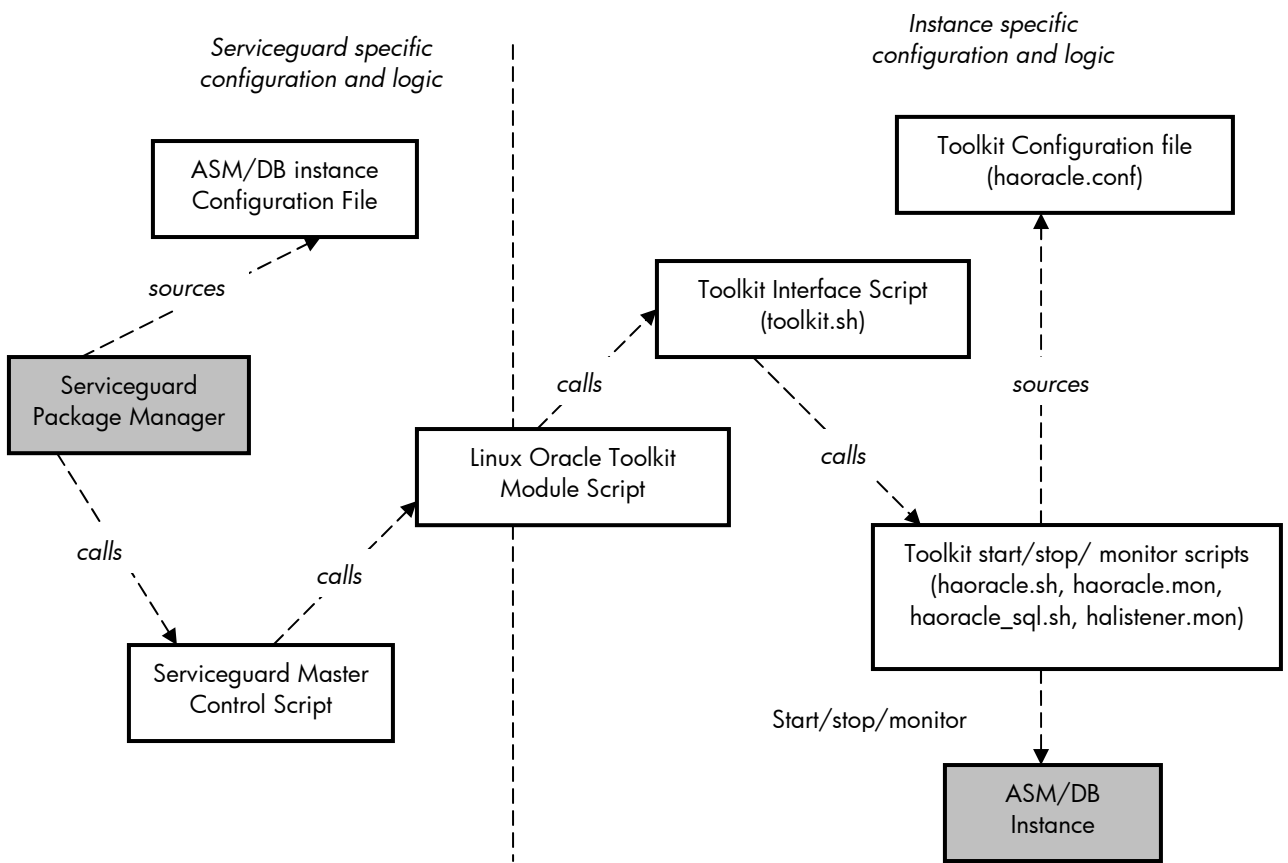
Legacy packages use the package configuration file and the package control script for the ASM or database instance. The package configuration file parameters are stored in the Serviceguard configuration database (CDB) at *cmapplyconf* time, and are used by the package manager in its actions on behalf of this package. The control script invokes the ASM or the database instance specific functions for start/stop/monitor through the toolkit interface script (*toolkit.sh*). On the instance specific side, there is a toolkit configuration file (*haoracle.conf*) which is sourced by the start/stop/monitor (*haoracle.sh*, *haoracle\_sql.sh*, *haoracle.mon* and *halistener.mon*) script files. The toolkit interface script allows the start/stop/monitor calls to remain unaffected by changes in the instance specific scripts. Figure 6 shows the internal file structure for legacy packages.

Figure 6: Internal file structure for legacy packages



Modular packages use the package configuration file for the ASM or database instance on the Serviceguard specific side. The package configuration parameters are stored in the Serviceguard configuration database at *cmaplyconf* time, and are used by the package manager in its actions on behalf of this package. The Serviceguard master control script invokes the ASM or the database instance specific functions for start/stop/monitor through the Oracle toolkit module script. The toolkit module script in turn calls the legacy scripts *toolkit.sh*, *haoracle.sh*, *haoracle\_sql.sh*, *haoracle.mon*, *haoracle.conf* and *halistener.mon* to start/stop/monitor the ASM or database instance

Fig 7: Internal file structure for modular packages



## 8 ASM File Descriptor Release

When an ASM disk group is dismounted on a node in the Serviceguard cluster, it is expected that the ASM instance closes the related descriptors of files opened on the raw volumes underlying the members of that ASM disk group. However, there is a possibility that processes of the ASM instance and client processes to the ASM instance may not close the descriptors.

Consider a configuration in which there are multiple databases using ASM to manage their storage in a Serviceguard cluster. Assume each database stores its data in its own exclusive set of ASM disk groups.

If the database instance is shutdown on a node and then its ASM disk groups are dismounted from that node, some ASM background and client processes may still hold open file descriptors to the underlying raw logical volumes. Hence an attempt to deactivate the corresponding LVM volume group(s) using hosttags on the node may fail.

Oracle has provided interim patches with which the ASM background processes close the file descriptors on the dismounted disk group volumes.

However, any client connections into the ASM instance using sqlplus or Oracle Enterprise Manager (OEM) results in ASM foreground processes opening file descriptors on all the disk group volumes of all the database instances, and these descriptors may not close on dismount of the disk group. Oracle allows for terminating these foreground processes.<sup>2</sup>

A new user configurable parameter `KILL_ASM_FOREGROUNDS` is introduced for ASM integration with Serviceguard. This parameter determines whether the scripts should kill the ASM foregrounds if they have file descriptors open on the dismounted disk group volumes. The ASM PMON process will recover any foreground processes that were killed. This parameter can be set either to yes or no. Default value is "yes".

The scripts will check to see if any ASM foreground processes of the form `oracle<ASM_SID>` have file descriptors open on the dismounted disk group volumes of the database instance. If this parameter is set to "yes", and the ASM foregrounds have file descriptors open after dismount of the disk group, they will be killed using a SIGKILL signal by the scripts. Before killing the ASM foreground processes, the scripts will check every 3 seconds for a maximum of 12 seconds to see if the foreground processes have closed the file descriptors. If the file descriptors are closed within the 12 second timeout, then package halt will continue. If the database uses logical volumes, volume groups will be deactivated using hosttags without waiting for the completion of this timeout. If this parameter is set to "no", the ASM foregrounds will not be killed. However, when this parameter is set to "no", and if ASM foregrounds have file descriptors open, volume group deactivation using hosttags and hence the database instance package halt will fail, if the database uses logical volumes. However, if the database uses only physical volumes, when the parameter is set no and if ASM foregrounds have file descriptors open, the database instance package will halt without any issues.

---

<sup>2</sup> Refer Oracle Metalink Service Request 7000214.993 at <https://metalink.oracle.com> (Oracle Metalink account required)

## 9 Installation, Configuration and Troubleshooting

Oracle ASM is part of the Oracle database server installation. No additional software from HP is required to operate ASM in Serviceguard environment. The Oracle 10.2.0.4 and 11.1.0.6 patches 7330611 and 7225720 need to be installed on all nodes for successful database instance failover.

Oracle ASM and ASM disk groups may be configured at the time of creating a database or as a separate step. Ensure that the LVM volume groups and raw disks have been prepared and activated prior to ASM disk group configuration or reconfiguration, as discussed in this document. Use the names of raw LVs contained in the LVM volume groups when Oracle commands and utilities require the specification of the names of raw devices for storing ASM disk groups. Ensure that each database instance uses different disk groups. In case of raw devices, ensure that the device mapper names in the path `/dev/mapper` is used. Only device mapper names in `/dev/mapper` are persistent and device mapper names of the format `/dev/dm-*` are not persistent.

Assume that the Serviceguard cluster, ASM instance and one or more database instances are already installed and configured.

- Halt the ASM and database instances.
- Configure the ASM MNP using the Linux Oracle toolkit scripts provided by HP following the instructions in the ASM-README file in the scripts bundle.
- Start the ASM instance on all the configured nodes by invoking `cmrunpkg` on the ASM MNP.
- Next, configure the database failover package using the Linux Oracle toolkit scripts provided by HP following the instructions in the README file in the scripts bundle.
- Start the database instance on the node by invoking `cmrunpkg` on the database package.

For troubleshooting, it may be useful to look at subdirectories for database and ASM instances under `$ORACLE_BASE/admin/`, where log and trace files for these instances are deposited. Also look at Linux syslog and Serviceguard package logs.

## 10 Related Documentation

- *Managing Serviceguard Fifteenth Edition, reprinted May 2008* at <http://docs.hp.com/en/ha.html>
- *Serviceguard documentation* at <http://docs.hp.com/en/ha.html#Serviceguard>
- *Oracle Database 10g Release 2 Installation Guide for Linux x86\_64* at <http://www.oracle.com/pls/db102/db102.homepage>
- *Oracle Database 11g Release 1 Installation Guide for Linux x86\_64* at <http://www.oracle.com/pls/db111/homepage>
- Linux Oracle toolkit ASM-README is available when Linux version A.05.00 is installed or downloaded from <http://software.hp.com> under High Availability.
- Configuring raw devices (singlepath) for Oracle Clusterware 10g Release 2 (10.2.0) on RHEL5/OEL5  
[https://metalink2.oracle.com/metalink/plsql/f?p=130:14:8021644706365972990:::p14\\_database\\_id,p14\\_docid,p14\\_show\\_header,p14\\_show\\_help,p14\\_black\\_frame,p14\\_font:NOT,465001.1,1,0,1,Helvetica](https://metalink2.oracle.com/metalink/plsql/f?p=130:14:8021644706365972990:::p14_database_id,p14_docid,p14_show_header,p14_show_help,p14_black_frame,p14_font:NOT,465001.1,1,0,1,Helvetica)
- [ASM using Raw devices](#)
- [Configure UDEV and Device Mapper for Oracle RAC10gr2](#)
- ASM related sections in Oracle manuals
  - *Oracle Database Administrator's Guide 10g R2* at [http://www.oracle.com/pls/db102/portal.portal\\_db?selected=1](http://www.oracle.com/pls/db102/portal.portal_db?selected=1)
  - *Oracle Database Administrator's Guide 11gR1* at [http://www.oracle.com/pls/db111/to\\_pdf?pathname=server.111/b28310.pdf](http://www.oracle.com/pls/db111/to_pdf?pathname=server.111/b28310.pdf)
  - <http://www.oracle.com/technology/documentation/database10gr2.html>
  - <http://www.oracle.com/technology/documentation/database11gr1.html>
  - <http://www.oracle.com/technology/products/database/asm/index.html>

© Copyright 2009 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Linux is a U.S. registered trademark of Linus Torvalds. Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation. UNIX is a registered trademark of The Open Group.

xxxx-xxxxEN, March 2009

