



Red Hat Reference Architecture Series

Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6

Deployment Recommendations

Version 1.1
March 2012





Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6.2: Deployment Recommendations

1801 Varsity Drive
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC 27709 USA

Linux is a registered trademark of Linus Torvalds. Red Hat, Red Hat Enterprise Linux and the Red Hat "Shadowman" logo are registered trademarks of Red Hat, Inc. in the United States and other countries.

Fusion-io is a registered trademark of Fusion-io, Inc.
IBM and System x are registered trademarks of IBM Corporation.
Intel, the Intel logo, and Xeon are registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
Oracle is a registered trademark of Oracle.
QLogic is a trademark or registered trademark of QLogic Corp.

All other trademarks referenced herein are the property of their respective owners.

© 2012 by Red Hat, Inc. This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, V1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

The information contained herein is subject to change without notice. Red Hat, Inc. shall not be liable for technical or editorial errors or omissions contained herein.

Distribution of modified versions of this document is prohibited without the explicit permission of Red Hat Inc.

Distribution of this work or derivative of this work in any standard (paper) book form for commercial purposes is prohibited unless prior permission is obtained from Red Hat Inc.

The GPG fingerprint of the security@redhat.com key is:
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Send feedback to refarch-feedback@redhat.com.



Table of Contents

1	Executive Summary	5
2	Introduction	6
3	Audience	7
4	Installing Oracle Database 11g Release 2	8
4.1	Red Hat Enterprise Linux 6.2 Operating System Installation (Summary)	8
4.1.1	SELinux	8
4.1.2	Firewall Configuration	9
4.1.3	Automatic System Tuning for Database Storage	10
4.1.4	Red Hat Enterprise Linux Packages	10
4.1.5	Network Adapter Configuration	11
4.1.6	Huge Pages	12
4.2	System Configuration Required by Oracle Universal Installer for the Database and, optionally, Grid Infrastructure	13
4.2.1	Virtual Memory	13
4.2.2	Shared Memory	13
4.2.3	Semaphores	14
4.2.4	User Accounts and Groups	15
4.2.5	File System Layout	15
4.2.6	Ephemeral Network Ports	16
4.2.7	Optimizing Network Settings (TCP/IP)	16
4.2.8	Miscellaneous System Configuration	16
4.2.9	Security Settings and Recommendations	17
4.3	Preparing Storage LUNs for Oracle Database 11g Release 2 (General Discussion)	17
4.4	Installing Oracle 11g Database Release 2 for a Standalone Server	18
4.4.1	Installing and Configuring Oracle Grid Infrastructure 11g Release 2 for a Standalone Server (optional)	18
4.4.2	Installing Oracle Database 11g Release 2 on a Standalone Server	27
4.4.3	Post-installation Steps	39
5	Database Storage Examples for Standalone Oracle Database 11g Release 2	40
5.1	Scenario 1: SAN Only	40
5.1.1	Connecting to the IBM DS3524 Storage Array	41
5.1.2	Preparing the LUNs for ASM without ASMLib	41
5.1.3	Using Linux udev Instead of Oracle ASMLib	41
5.1.4	Creating ASM Disk Groups	42
5.1.5	Creating the Database	44
5.2	Option 2: SSD Only	44
5.2.1	Connecting the Fusion-io ioDisks	44
5.2.2	Using Linux udev Instead of Oracle ASMLib	44
5.2.3	Creating ASM Disk Groups	45
5.2.4	Creating the Database	45
5.3	Option 3: HDD for Most with SSD for Database Logs	45
5.3.1	Connecting the Storage	45
5.3.2	Using Linux udev	46
5.3.3	Creating ASM Disk Groups	46



5.3.4	Creating the Database	46
5.4	Option 4: Fusion-io directCache Fronting the SAN	46
5.4.1	Installing and Configuring Fusion-io directCache	46
5.4.2	Using ASM Disk Groups and Linux udev	47
5.4.3	Creating the Database	47
6	Terms.....	48
7	References	49
Appendix A.	Revision History.....	50



1 Executive Summary

This document provides recommendations for deploying standalone Oracle Database 11g Release 2 (version 11.2, or 11gR2) on Red Hat Enterprise Linux 6 that go beyond Oracle's installation instructions, and describes the benefits for doing so in certain cases. In particular, it describes the benefits of advanced features found in Red Hat Enterprise Linux 6 to increase performance and provide security; for example, I/O scheduling, transparent and regular huge pages, automatic tuning of TCP/IP parameters, SELinux, and tuned profiles for automatically configuring the system for enterprise storage.

This document provides database deployment instructions in Red Hat Enterprise Linux 6.2 for four reference storage solutions using both traditional spinning devices (hard disk drives, or HDDs) and low-latency spinless devices (solid-state drives, or SSDs). The first two configurations store the entire set of database files and logs on one device. (See Figures 5-1 and 5-2.)

The other two reference storage solutions use storage area network (SAN) HDDs and SSDs simultaneously. Depending on your database application and budget, a hybrid configuration might yield better performance for a reasonable price (currently, SSDs are generally more expensive per gigabyte than HDDs). In the third reference configuration, we assign frequently used database files, logs and some indices, on the smaller, low-latency SSD, and the bulk of the data on the HDDs. (See Figure 5-3.)

The fourth configuration uses software provided by the SSD vendor Fusion-io to accelerate access to HDDs. The SSDs are used as a memory device rather than for long-term storage. They become an *automatic* front-side file cache for the HDDs, which provide the long-term storage. (See Figure 5-4.) This Linux package, ioMemory, caches frequently used data read from the HDD in the SSD so that it can be quickly sent to the CPU. Unlike the previous configuration, users need not think in terms of files as the cache works with smaller chunks of data, or determine which files to assign to which device. For further details on Oracle Database performance with Flash-based SSDs, see the Red Hat Performance Brief, *Optimizing Fusion ioMemory on Red Hat Enterprise Linux 6 for Database Performance Acceleration*.¹



2 Introduction

Deploying Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6 is a straightforward task that allows for a number of storage configuration options. This document provides configuration recommendations that enables you to get the most from your Oracle 11g R2 database installation on Red Hat Enterprise Linux 6.

This document presents five connected but independent topics:

1. Operating system installation and general configuration (Section 4.1)
2. Additional Oracle-specific configuration (Section 4.2)
3. Database LUN preparation, discussed generally (Section 4.3)
4. Oracle software installation and database creation (Section 4.4)
5. In-depth description of storage setup and creation of LUNs, volume groups, and files for the application database (Section 5)

If you need only a quick look at deploying Oracle Database 11g Release 2, or you are uninterested in the storage scenarios considered here, proceed immediately to Section 5. Otherwise, continue on to Section 4 for step-by-step instructions on deploying Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6.2.



3 Audience

While the ideal team for deploying Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6.2 might be the Red Hat Enterprise Linux system administrator, storage administrator, database administrators, and application owner, either a system administrator or a database administrator (the latter having limited administrator access) can install and configure Oracle Database on Red Hat Enterprise Linux by following the recommendations below.



4 Installing Oracle Database 11g Release 2

The first four steps for deployment of Oracle Database 11g Release 2 on Red Hat Enterprise Linux 6.2 take you from system preparation to creating a database instance. Two methods for creating file systems to hold database files are given: standard Linux filesystems and Oracle's disk volume manager, ASM. In order to use ASM, however, you must first install Oracle Grid Infrastructure.

4.1 Red Hat Enterprise Linux 6.2 Operating System Installation (Summary)

This guide assumes an installation of Red Hat Enterprise Linux 6.2 x86_64 from DVD on an IBM® System x®3650 M3 (7945AC1) server with 96 GB of RAM, and two hex-core Intel® Xeon® processor X5670s. This server has two internal hard drives in a RAID 1 configuration for the operating system, and for Oracle home and its configuration files. This guide first describes selecting the base package configuration, and describes adding additional packages to support Oracle Database below.

When installing the operating system, choose the hostname of the database with some care, and do not use underscores (_). Server hostnames are used, by default, in determining the global identifier of a database, and it can be difficult to connect some SQL*Net clients to a database instance with an underscore in its name.

The next five sections provide additional operating system configuration steps that are common to many application-server installations. Root access is needed, and these tasks can be performed using either `sudo` or `su`.

4.1.1 SELinux

SELinux provides mandatory policy-based controls to prevent unexpected (usually unauthorized) access to system resources including files, TCP ports, and more. These controls greatly augment the Unix/Linux model of read/write/execute access by a file owner/group-member/other. Access controls can follow the data.

Red Hat provides robust policies in Enterprise Linux that, in particular, work with Oracle Database, and recommends that the SELinux mode be kept at its default settings: *enforcing* (the security policy) and *targeted* (to protect only specific processes).

Note that if your Oracle Database configuration differs significantly from the customary (e.g., the TNS Listener uses TCP port 5678 instead of 1521), the default SELinux configuration may prevent the application from functioning correctly. In such cases, the SELinux policy can be modified to associate this TCP port with Oracle TNS Listener. See *Red Hat Enterprise Linux 6 Security-Enhanced Linux User Guide*,² for more information regarding SELinux configuration.

For example, to label TCP port 1600 as one used by Oracle, run the following command:

```
# semanage port -a -t oracle_port_t -p tcp 1600
```




To see the security labels attached to running processes, run the command `ps -eZ`. The security contexts attached to files are shown with the command `ls -Z`. For example, running these commands will give results resembling the following traces:

```
# ps -eZ
LABEL                                PID  TTY  TIME  CMD
system_u:system_r:init_t:s0          1    ?    00:00:02  init
system_u:system_r:kernel_t:s0        2    ?    00:00:00  kthreadd
...
system_u:system_r:getty_t:s0         2762  tty6  00:00:00  mingetty
system_u:system_r:sshd_t:s0-s0:c0.c1023 5778  ?    00:00:00  sshd
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 5781 pts/0 00:00:00  bash
system_u:system_r:udev_t:s0-s0:c0.c1023 5814  ?    00:00:00  udevd
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 5887 pts/0 00:00:00  man
```

```
# ls -Z /etc
drwxr-xr-x. root root system_u:object_r:abrt_etc_t:s0  abrt
drwxr-xr-x. root root system_u:object_r:etc_t:s0      acpi
-rw-r--r--. root root system_u:object_r:adjtime_t:s0  adjtime
-rwxr-xr-x. root root unconfined_u:object_r:etc_t:s0    agent.properties
-rw-r--r--. root root system_u:object_r:etc_aliases_t:s0  aliases
-rw-r--r--. root root system_u:object_r:etc_aliases_t:s0  aliases.db
...
```

4.1.2 Firewall Configuration

In general, the iptables firewall should be configured to permit access to the Oracle Database network ports *only* from authorized database or database-management clients. The former makes requests to the TNS Listener, while the latter might use Web-based Oracle Enterprise Manager. Although maintaining the firewall configuration over the lifetime of the application requires a modicum of discipline in following procedures and maintaining documentation, implementing a sensible firewall policy during deployment can often guard against intrusions from unexpected sources.

The iptables firewall does not perform deep packet inspection, and cannot, for example, prevent attacks encoded in SQL transactions or in malformed SQL input.

For this standalone database server, modify the iptables configuration to allow a database client with IP address 192.168.201.17, for example, to communicate with the database server via SQL*Net on its default port, 1521/tcp by adding one firewall rule to the default configuration file, `/etc/sysconfig/iptables`.

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```



```
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
# What: oracle database: permit SQL*Net traffic from client
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
-A INPUT -m state --state NEW -m tcp -p tcp -s 192.168.201.17 --dport 1521 -j ACCEPT
# End of modifications for oracle database
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

4.1.3 Automatic System Tuning for Database Storage

The tuned package in Red Hat Enterprise Linux 6.2 is recommended for automatically tuning the system for common workloads: enterprise storage, high network throughput, and power savings. For Oracle Database, enable the enterprise-storage profile to set the I/O scheduler, adjust the read-ahead buffers for non-system disks, etc.

```
# yum install tuned
# chkconfig tuned on
# tuned-adm profile enterprise-storage
```

4.1.4 Red Hat Enterprise Linux Packages

The Oracle Database 11g Release 2 documentation for Red Hat Enterprise Linux 5.5 (x86_64) notes that the RPMs in Figure 4-1 are needed to run the Oracle Universal Installer, an X Window application. See Section 4.3 in the *Oracle Database Installation Guide 11g Release 2 (11.2) for Linux*. Red Hat Enterprise Linux 6.2 provides updated versions of these packages that work with Oracle Database 11g Release 2.

```
binutils-2.17.50.0.6
compat-libstdc++-33-3.2.3
compat-libstdc++-33-3.2.3 (32 bit)
elfutils-libelf-0.125
elfutils-libelf-devel-0.125
gcc-4.1.2
gcc-c++-4.1.2
glibc-2.5-24
glibc-2.5-24 (32 bit)
glibc-common-2.5
glibc-devel-2.5
glibc-devel-2.5 (32 bit)
glibc-headers-2.5
ksh-20060214
libaio-0.3.106
libaio-0.3.106 (32 bit)
libaio-devel-0.3.106
libaio-devel-0.3.106 (32 bit)
libgcc-4.1.2
libgcc-4.1.2 (32 bit)
libstdc++-4.1.2
libstdc++-4.1.2 (32 bit)
libstdc++-devel-4.1.2
make-3.81
sysstat-7.0.2
```

Figure 4-1: Required RPMs for the Oracle Universal Installer for Red Hat Enterprise Linux 5.5 (x86_64).

To install this required software, but in their updated, 6.2 versions, two one-line shell scripts may be useful. First, create a file, *req-rpm.txt*, with the list of RPMs from Figure 4-1, one to a line, but be sure to correct the typographical error for package “libstdc++-devel 4.1.2”,



replacing it with “libstdc++-devel-4.1.2”. The optional Grid Infrastructure software requires the 64-bit version of compat-libcap1.

After this preparatory work, install the 64-bit packages and their dependencies with this one-line command:

```
# yum install `grep -v '32 bit' req-rpm.txt | sed 's/-[^-]*$//'` compat-libcap1
```

Similarly, install the 32-bit packages and their dependencies with this one-line command:

```
# yum install $(grep '32 bit' req-rpm.txt | sed 's/-[^-]*$/^*86/')
```

Finally, if your database needs an ODBC environment, install it as follows:

```
# yum install unixODBC unixODBC-devel unixODBC.i686
```

4.1.5 Network Adapter Configuration

When possible, use separate networks and network interfaces for server management functions and database functions. For example, assign one network interface, eth0, to access the server remotely for database and system management (on network 10.10.1.0/24), and one, eth5, to connect for the database connections (on network 192.168.201.0/24).

Oracle expects the name assigned to the TNS Listener to be resolvable. Add the IP address and its hostname for the database network to /etc/hosts, for example:

```
# Database Network
192.168.201.31      db001
```

Network interfaces can be logically bonded to improve performance and/or provide higher availability. Red Hat Enterprise Linux provides several bonding modes: round-robin and XOR bonding actively switch traffic from one path to another, while active/passive bonding uses one path, keeping the others in reserve. If the network interfaces to be bonded are part of a switched network, the network switch attached to the bonded ports may also require configuration in order to properly read and write network traffic from the bonded interface.

As an example of an active/passive bond, named bond0 with IP address 192.168.30.10, using the physical interfaces, eth0 and eth1, first modify or create the files ifcfg-bond0, ifcfg-eth0 and ifcfg-eth1 in directory /etc/sysconfig/network-scripts as follows:

```
# ifcfg-bond0
DEVICE=bond0
## active/passive mode; check the active port every 100 ms.
BONDING_OPTS='mode=1 miimon=100 primary=eth0'
NM_CONTROLLED=no
IPADDR=192.168.30.10
NETMASK=255.255.255.0
GATEWAY=192.168.30.1
ONBOOT=yes
```



```
# ifcfg-eth0
DEVICE=eth0
SLAVE=yes
MASTER=bond0
ONBOOT=yes
```

```
# ifcfg-eth1
DEVICE=eth1
SLAVE=yes
MASTER=bond0
ONBOOT=yes
```

Next, create `/etc/modprobe.d/bonding.conf` with contents:

```
alias bond0 bonding
```

Start the new network configuration with:

```
# service network restart
```

4.1.6 Huge Pages

Red Hat Enterprise Linux 6.2 uses transparent huge pages, sometimes called anonymous huge pages, by default. Unlike ordinary huge pages, no additional configuration is needed to use them. Huge pages can boost application performance by increasing the chance a program will have quick access to a memory page. Unlike ordinary huge pages, transparent huge pages can be swapped out (as smaller 4kB pages) when virtual memory cleanup is needed.

Because Oracle uses System V shared memory, it does not use transparent huge pages for interprocess communication. To reap the benefit of huge pages, allocate regular (or static) huge pages for your Oracle database.

Note: Huge pages require special setup, and misconfigured systems could lead to severe performance penalties, in extreme cases resulting in an unresponsive system. The general rule is to allocate just enough static huge pages so that all of Oracle's SGA fits. If the huge pages pool is too small, Oracle opens SGA in regular system pages, leaving the entire huge page section unused.

4.1.6.1 Configuring huge pages for Oracle

In Red Hat Enterprise Linux 6, the size of the huge-pages pool is specified by the desired number of huge pages. To calculate the number of huge pages, first determine the size of one huge page by running the following command:

```
# grep Hugepagesize /proc/meminfo
Hugepagesize: 2048 kB
```



The output shows that their size on this system is 2MB. Thus, to create a huge-page pool of 1GB, for example, 512 pages are needed. Add the following to `/etc/sysctl.conf`:

```
# What: oracle database: allocate 1 GB of static huge pages
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
vm.nr_hugepages=512
```

Because static huge pages are locked in memory and cannot be swapped, the oracle user account must be given permission to allocate and lock this memory. For this example of a huge-page pool of 1GB, add the following lines in the `/etc/security/limits.conf`:

```
oracle hard memlock 1048576
oracle soft memlock 1048576
```

4.2 System Configuration Required by Oracle Universal Installer for the Database and, optionally, Grid Infrastructure

Oracle Database 11g Release 2 requires additional system configuration to function properly. The Oracle installer checks many of these settings and offers the opportunity to correct them, but it is preferable to set them before running the installer.

As mentioned previously, if you wish to use Oracle ASM to manage storage at a low level but hidden behind a high level of abstraction, you must also install Oracle Grid Infrastructure 11g Release 2.

4.2.1 Virtual Memory

The following values are recommended to control the rate at which virtual memory is reclaimed. Add the following to `/etc/sysctl.conf`:

```
# What: oracle database: tune shared memory size
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
vm.swappiness=0
vm.dirty_background_ratio=3
vm.dirty_ratio=15
vm.dirty_expire_centisecs=500
vm.dirty_writeback_centisecs=100
```

The swap should be at least 16GB for 64GB to 256GB of RAM. For swap sizes for other values of RAM, see *Performance Tuning Guide: Optimizing subsystem throughput in Red Hat Enterprise Linux 6*.³

4.2.2 Shared Memory

Like many operating systems, Linux provides shared memory for interprocess communication, such as incoming data requests, the database's internal state, and internal control parameters. Red Hat Enterprise Linux 6.2 provides by default an ample supply of shared memory, but Oracle Database benefits from some adjustments.



Shared-memory tuning depends upon the size of Oracle's system global area (SGA). Because more is usually better, SGA allocation ideally increases with database and system RAM size. Ultimately, the amount of server RAM limits SGA size. To tune shared memory for the SGA, you calculate the maximum amount of shared memory for the kernel to allocate, the maximum size of any one memory segment, and the maximum number of segments.

First, determine the maximum amount of shared memory (the `shmall` kernel parameter), in units of page size for this 64-bit system by running one command and performing one calculation. First, obtain the page size of the system:

```
# getconf PAGE_SIZE ## page size in bytes
4096
```

If the amount of RAM is 96 GB, for example, calculate `shmall` by dividing 96 GB by the page size; namely, $96 \times 1024^3 / 4096$ or 25,165,824; for 16 GB of RAM, `shmall` is proportionally smaller: $16 \times 1024^3 / 4096 = 4,194,304$.

The maximum size of a shared segment (`shmmax`) should be one-half the size of total memory. So for our 96 GB example, this parameter should be set to $96 \times 1024^3 / 2$ or 51,539,607,552; for 16 GB, `shmmax`, is proportionally smaller: $16 \times 1024^3 / 2 = 8,589,934,592$.

Oracle recommends setting the maximum number of shared-memory segments to 4,096.

Add the following to `/etc/sysctl.conf`, remembering to adjust `shmall`, and `shmmax` to your server's RAM:

```
# What: oracle database: tune shared memory size
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
#### NOTE: the values of shmall and shmmax are proportional to the
#### amount of server RAM, and must be adjusted accordingly.
##
## The following values are for 96 GB of RAM
kernel.shmall=25165824
kernel.shmmax=51539607552
# do not scale this parameter with RAM
kernel.shmmni=4096
```

4.2.3 Semaphores

Linux provides semaphores for sharing small amounts of information between processes. Semaphores are numbers—counters or on/off values—that are organized into groups. Oracle Database needs more groups, more room in each group, and more counters than the operating system supplies by default.

Add the following to `/etc/sysctl.conf` to provide Oracle with sufficient resources for semaphores:



```
# What: oracle database: increase no. system-wide semaphore arrays, etc.  
# When: 12/12/2011  
# Who: J. B. Admin, jba@localhost  
kernel.sem = 250 32000 100 128
```

4.2.4 User Accounts and Groups

Oracle Database 11g Release 2 has nicely separated roles for the application's lifecycle; for example, software installer, Oracle application owner(s), database administrator, database operator (backups), and of course, system administrator (root). Even though these roles are often combined or may not be immediately required, it is beneficial to create them now.

As root, create the following user accounts, groups, and group assignments using a consistent UID and GID assignments across your organization:

```
# groupadd -g 1001 oinstall          ## software inventory  
# groupadd -g 1002 dba              ## database  
# groupadd -g 1003 oper             ## database  
# groupadd -g 1004 asmadmin        ## ASM, if needed  
# groupadd -g 1005 asmdba         ## ASM, if needed  
# groupadd -g 1006 asmoper        ## ASM, if needed  
  
# useradd -u 1002 -g oinstall -G dba,oper,asmadmin,asmdba,asmoper oracle  
# passwd oracle
```

During ordinary operation, the *oracle* account starts many processes, opens many files, and requires more resources than an ordinary user. At the same time, hard limits on the account prevent some cases of resource starvation. The default PAM configuration loads the modules that set and enforce these limits.

As root, add the following lines to */etc/security/limits.conf*:

```
# What: oracle database: increase resources available to oracle account  
# When: 12/12/2011  
# Who: J. B. Admin, jba@localhost  
oracle -          nproc          16384  
oracle -          nofile         65536  
oracle soft      stack           10240
```

4.2.5 File System Layout

Following Oracle's long-standing file-system conventions, create a local file system mounted at */u01* for the Oracle inventory and software. The Oracle "base" for application installation is */u01/app/oracle*. As root, create this directory tree and protect it:

```
# mkdir -p /u01/app/oracle  
# chown -R oracle:oinstall /u01  
# chmod -R 775 /u01
```



4.2.6 Ephemeral Network Ports

Add the following line to `/etc/sysctl.conf` to modify the range of IP ports for automatically assigned client source-ports:

```
# What: oracle database: increase autonomous client-side IP port range
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
net.ipv4.ip_local_port_range = 9000 65500
```

4.2.7 Optimizing Network Settings (TCP/IP)

Red Hat Enterprise Linux 6.2 now *automatically* tunes TCP/IP-socket buffer sizes and application-socket buffer sizes. System administrators no longer must adjust them for high performance.

However, Oracle and the Oracle Universal Installer recommend setting static sizes for the default and maximum buffers for application sockets in Database 11g Release 2. Their recommended values form a reasonable starting point. Add the following to `/etc/sysctl.conf` to modify tuning of these buffers:

```
# What: oracle database: increase application socket buffers
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
```

4.2.8 Miscellaneous System Configuration

You may need to increase the following configuration values if they are too low:

- Maximum number of open files: Oracle recommends 6,815,744 as a minimum. Check the default value by running this command, `sysctl fs.file-max`. Add “`fs.file-max = 6815744`” to `/etc/sysctl.conf` if the current value is less than this.
- The maximum number of concurrent asynchronous I/O requests: Oracle recommends raising this value to 1048576. If necessary, add “`fs.aio-max-nr = 1048576`” to `/etc/sysctl.conf`.

Configure the NTP daemon to synchronize system time for the stand-alone database server. The default time sources in `/etc/ntp.conf` are fine, but you may wish to change them to those used by your organization. To continue using the principle of separating network functions, configure the NTP daemon to only use the management network and avoid the database-client’s network. Oracle also recommends using slew time corrections. As root, modify the file `/etc/sysconfig/ntpd` and add “`-i eth5`” and “`-x`” to the argument list, where `eth5` is the management interface:

```
# Drop root to id 'ntp:ntp' by default.
OPTIONS="-u ntp:ntp -p /var/run/ntpd.pid -g -x -l eth5"
```




Finally, enable NTP and run it:

```
# chkconfig ntpd on  
# service ntpd start
```

4.2.9 Security Settings and Recommendations

Best practices for initial set-up for database security are straightforward, but ongoing security—or security as a process—is more difficult to summarize. Briefly, use root powers sparingly. Oracle has separated and grouped root actions; e.g., a small set of pre-installation tasks and a small set of post-install fix-up scripts. Do not use application accounts as your normal account; that is, log onto the database host as a non-generic, named user (jqbach, for example), and when database tasks are needed, use sudo or su to switch to the application (oracle) account.

These extra steps require discipline to follow (because the alternatives require fewer keystrokes), but the benefit lies in preventing accidents that might cause significant amount of time to correct, and in providing a simple log of who has done what and when.

4.3 *Preparing Storage LUNs for Oracle Database 11g Release 2 (General Discussion)*

Before providing configuration recommendations for specific storage devices, the configuration and preparation of general SCSI storage devices will be shown. Aligning LUNs or disk partition to match the underlying disk geometry – especially with the newer disks with 4kB sectors -- can improve performance by preventing unnecessary access. The native disk and file-system utilities, (e.g., parted and mkfs) in Red Hat Enterprise Linux 6.2 align automatically. For the purposes of this general discussion, assume that either hardware or software RAID is protecting the database storage, and that two 125GB storage devices have been allocated to the database: /dev/sdc and /dev/sdd.

Note: When using multipath or other device-mapper objects, be sure to use the mapped device (usually found in /dev/mapper) and not unmapped device (/dev/sdb, e.g.) in the following.

If you plan on using Oracle ASM rather than the OS to manage the storage (disk groups), skip the next five steps.

1. Create one partition spanning the device:

```
# parted /dev/sdc mklabel gpt mkpart primary "1 -1"  
# parted /dev/sdd mklabel gpt mkpart primary "1 -1"
```

2. Create an ext4 files system on each:

```
# mkfs.ext4 /dev/sdc1  
# mkfs.ext4 /dev/sdd1
```



3. Create mount points at /u02 and /u03:

```
# mkdir /u02 /u03
```

4. Edit fstab to mount them automatically. Append the following to /etc/fstab:

```
# What: oracle database: file systems when not using ASM
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost
/dev/sdc1 /u02 ext4 defaults 1 1
/dev/sdd1 /u03 ext4 defaults 1 1
```

5. Mount the file systems:

```
# mount /u02
# mount /u03
```

4.4 Installing Oracle 11g Database Release 2 for a Standalone Server

Note: You will need to change the installation parameters used in this section to match your requirements and environment.

To have access to ASM disk-group objects later, install and configure the Oracle Grid Infrastructure software first (Section 4.4.1), followed by the Oracle Database 11g Release 2 software (Section 4.4.2).

Although the Oracle Universal Installer (OUI) can run from the command line, it is convenient to use the X Window GUI for the initial installation. Log onto the database server either at the console or remotely, assuming you have set the DISPLAY variable, and used either xauth or xhost to permit communication between the server and your display workstation.

The OUI for Oracle Grid Infrastructure and Database 11g Release 2 may report missing RPMs (see, for example, steps 13 and 21 in sections 4.4.1 and 4.4.2, respectively) because it scans for these RPMs in the i386 architecture, whereas Red Hat Enterprise Linux 6.2 now provides them in the i686 architecture. This dependency check may be overridden (as shown below) and the installation will proceed as long as the corresponding i686 RPMs have been installed. See Section 4.1.4 for a list of required packages.

4.4.1 Installing and Configuring Oracle Grid Infrastructure 11g Release 2 for a Standalone Server (optional)

As of Oracle Database 11g Release 2, you must install Oracle Grid Infrastructure to use the Oracle ASM tools, which manage storage at a higher level of abstraction.

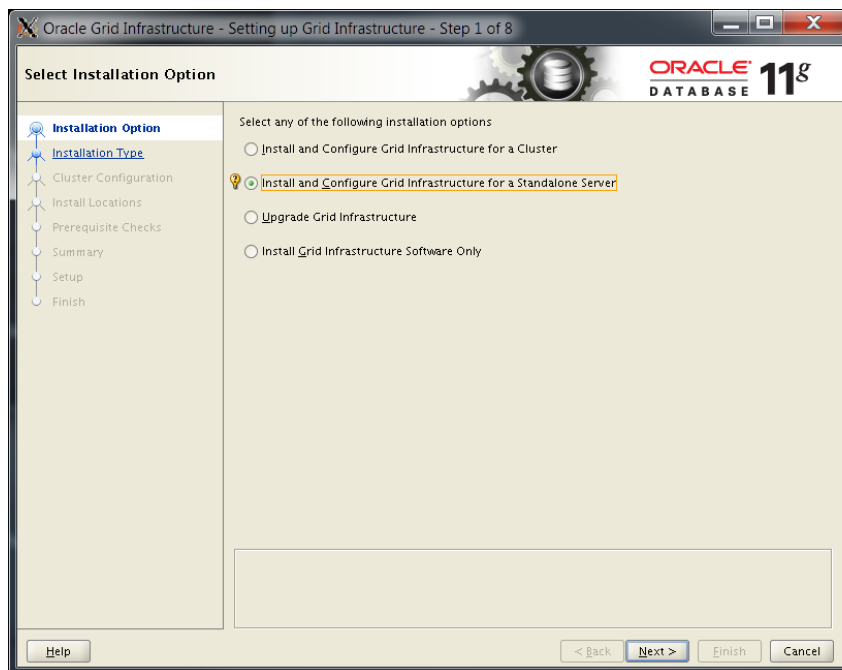
1. Log into the server, and switch to the oracle account.
2. Unzip the Oracle Grid Infrastructure software distribution file into /home/oracle, set the oracle hostname, oracle base, and run the installer. The installer will create a log file in the directory /u01/app/orainventory/logs/:

```
$ cd /home/oracle
```

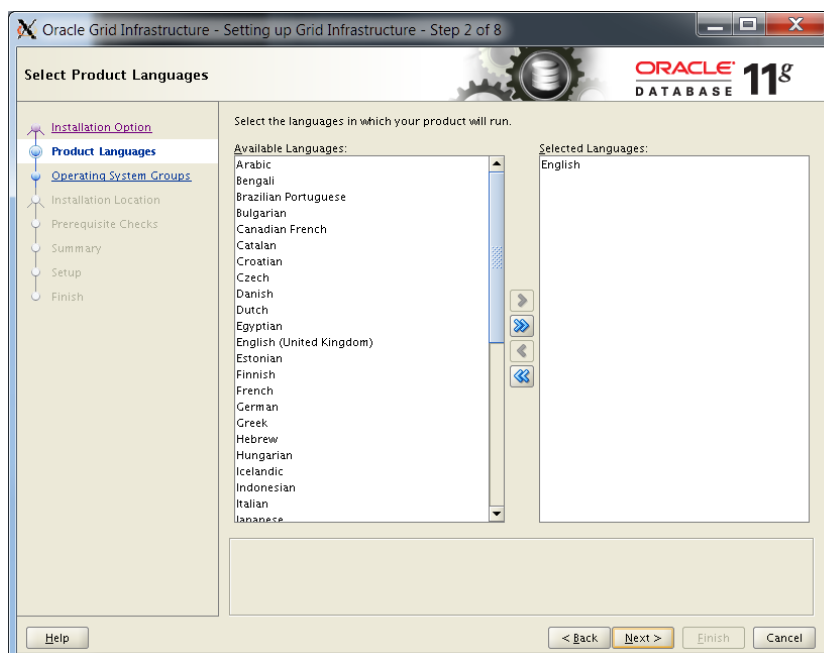


```
$ unzip linux.x64_11gR2_grid.zip
$ cd grid
$ ORACLE_HOSTNAME=db001
$ ORACLE_BASE=/u01/app/oracle
$ ./runInstaller
```

3. Type y and press Enter to bypass the error message regarding `/usr/bin/xdpyinfo` as that package is neither installed nor needed.
4. On the Step 1 screen, select Install and Configure Grid Infrastructure Software for a Standalone Server, and click Next.

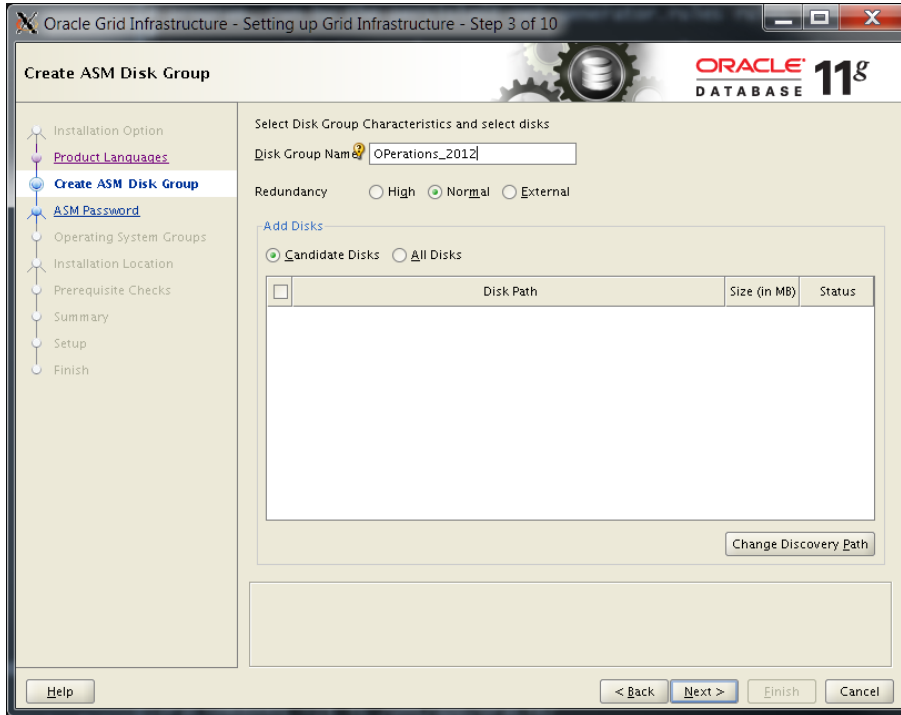


5. On the Step 2 screen, click Next to use English for the application language.





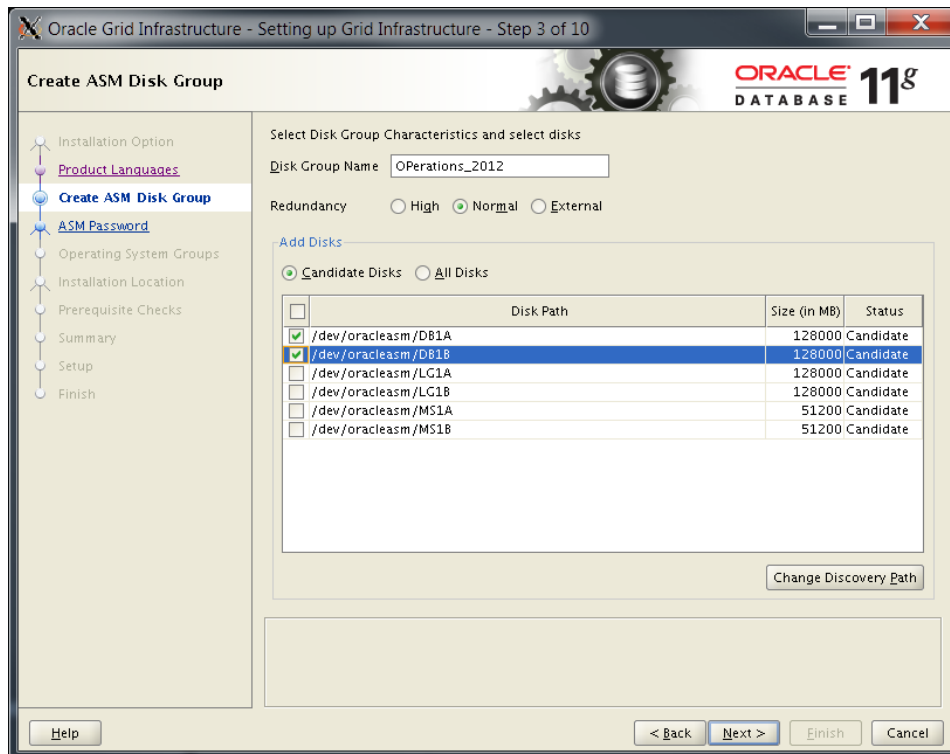
6. On the Step 3 screen, enter `Operations_2012`, for example, as the Disk Group Name, and click `Change Discovery Path`.



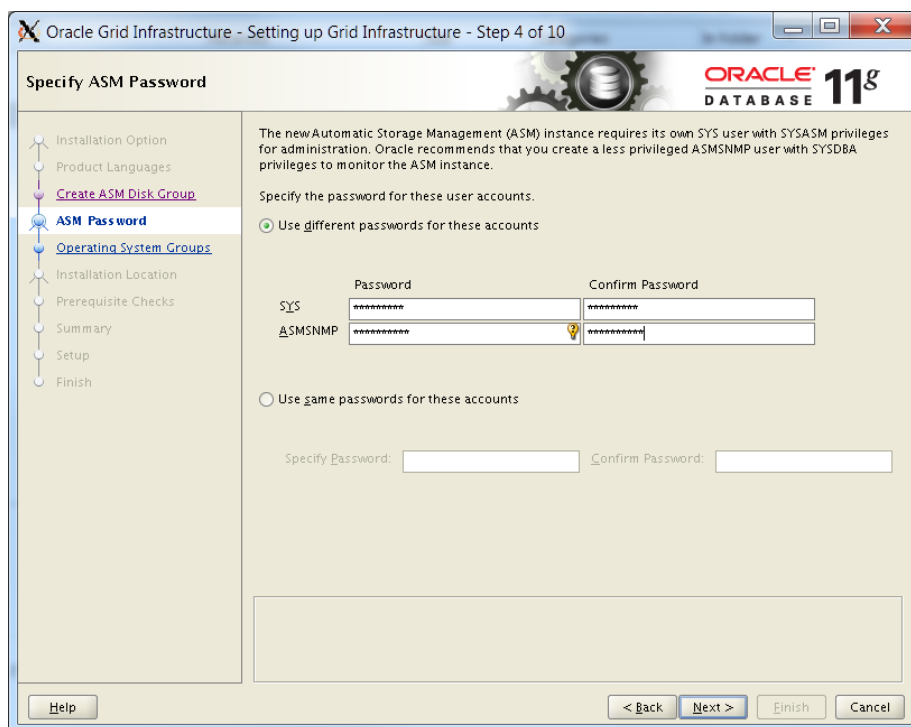
7. On the `Change Discovery Path` screen, enter `/dev/oracleasm`, and click `OK`.



8. The Step 3 screen is now populated with the LUNs prepared in Section 4.3. Select disks DB1A and DB2A, and click Next.

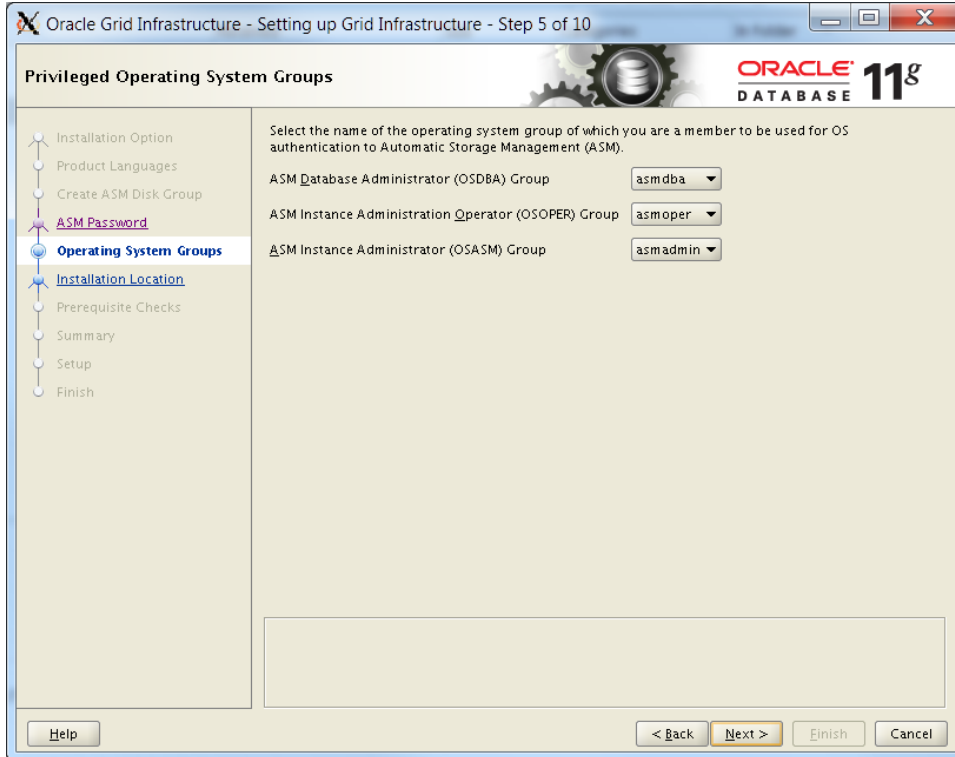


9. On the Step 4 screen, enter passwords for the ASM SYS and SNMP accounts, and click Next.

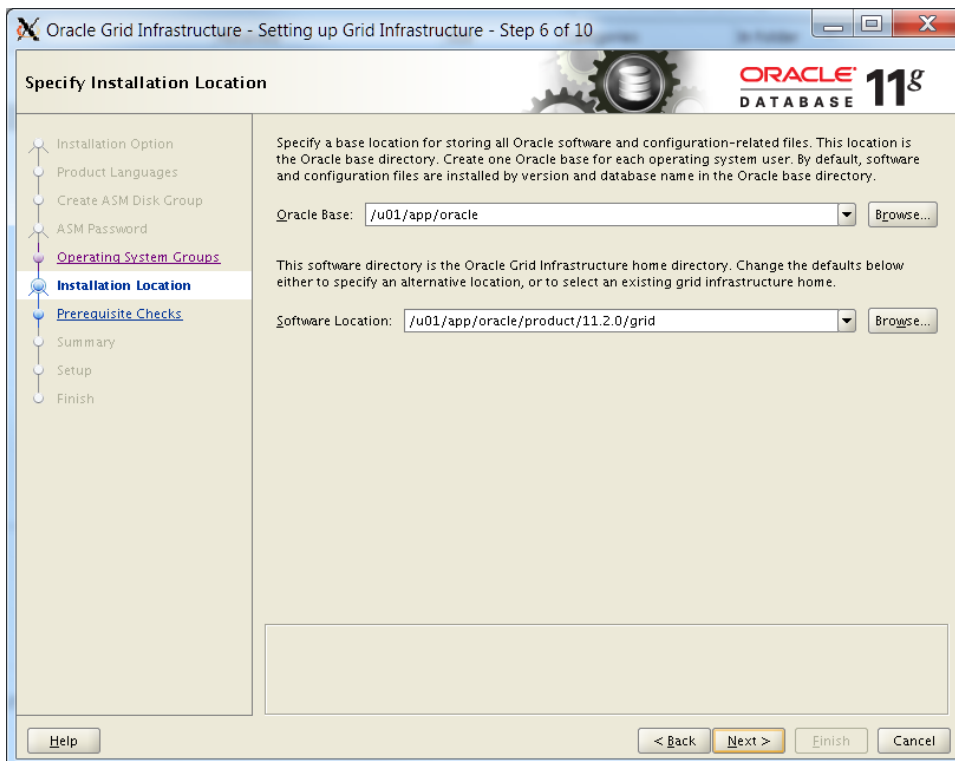




10. On the Step 5 screen, click Next to use the defaults for the three ASM groups.

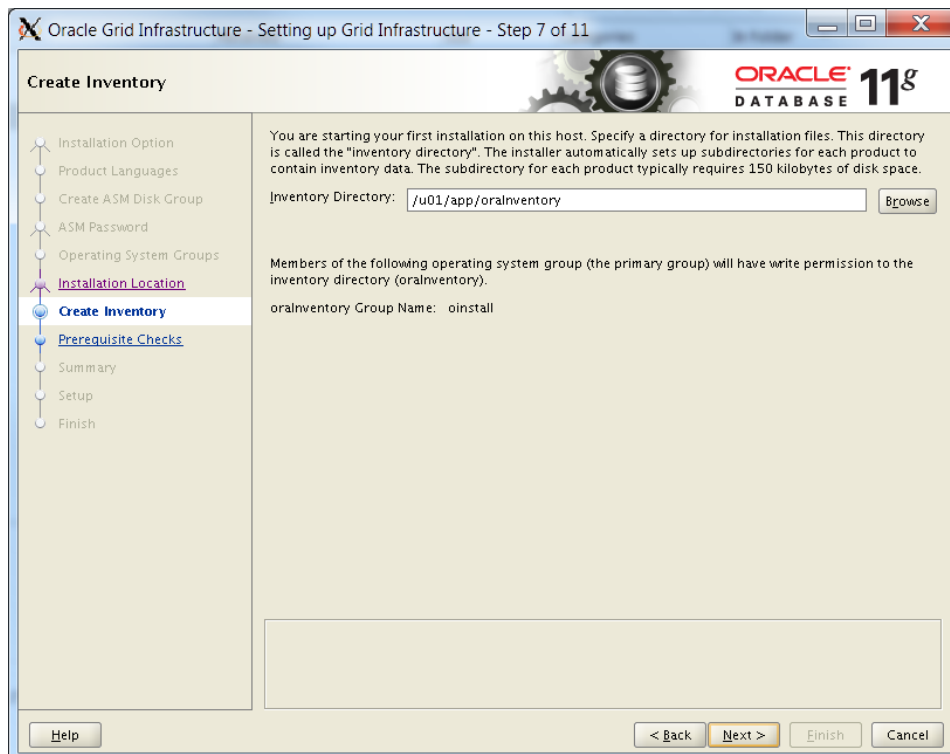


11. On the Step 6 screen, click Next to use the defaults for Oracle Base and Software location, which were computed from the value of ORACLE_BASE.

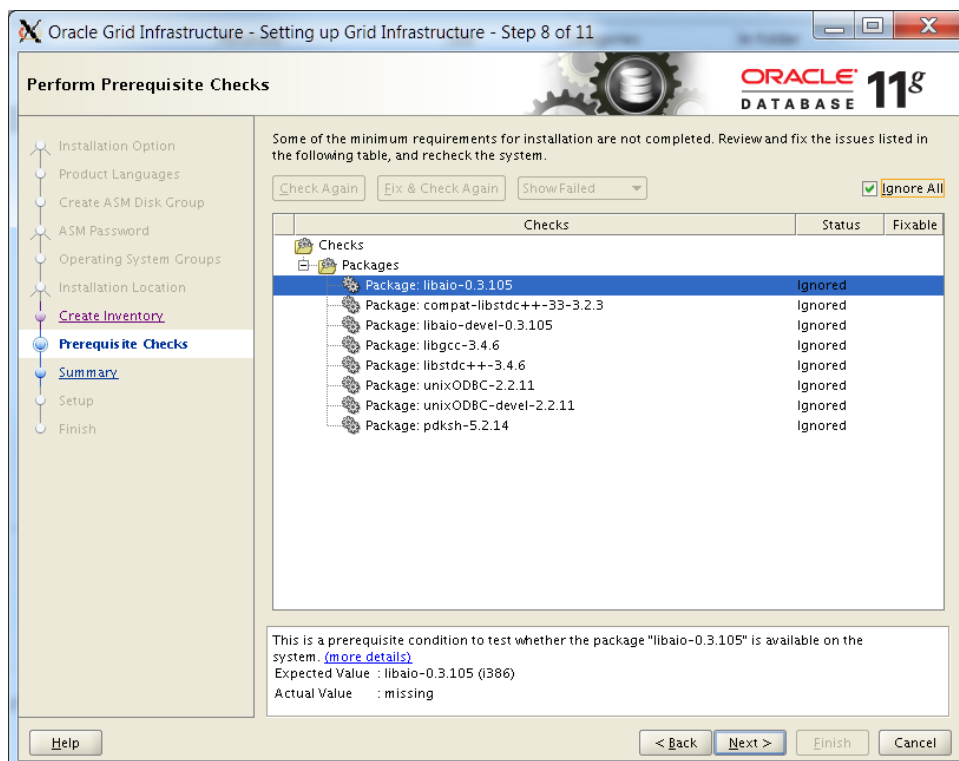




12. On the Step 7 screen, click Next to accept location for the Oracle product inventory.

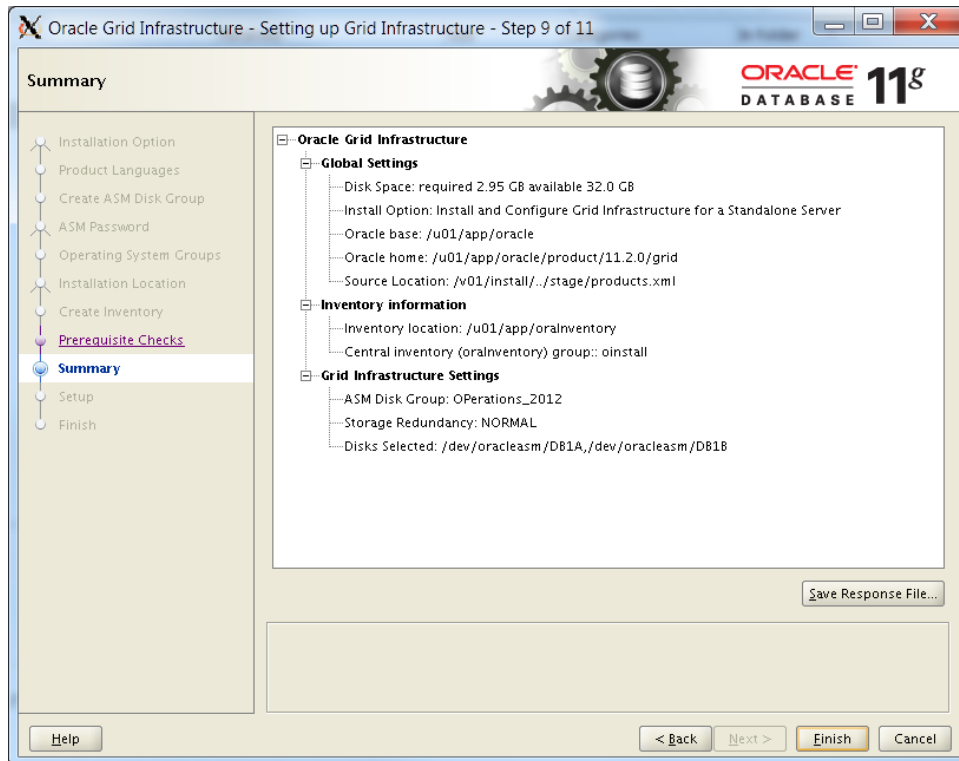


13. On the Step 8 screen, the installer checks the server's configuration against its minimum requirements. You can safely ignore the failed checks for packages if you completed the RPM installation procedure in Section 4.1.3.



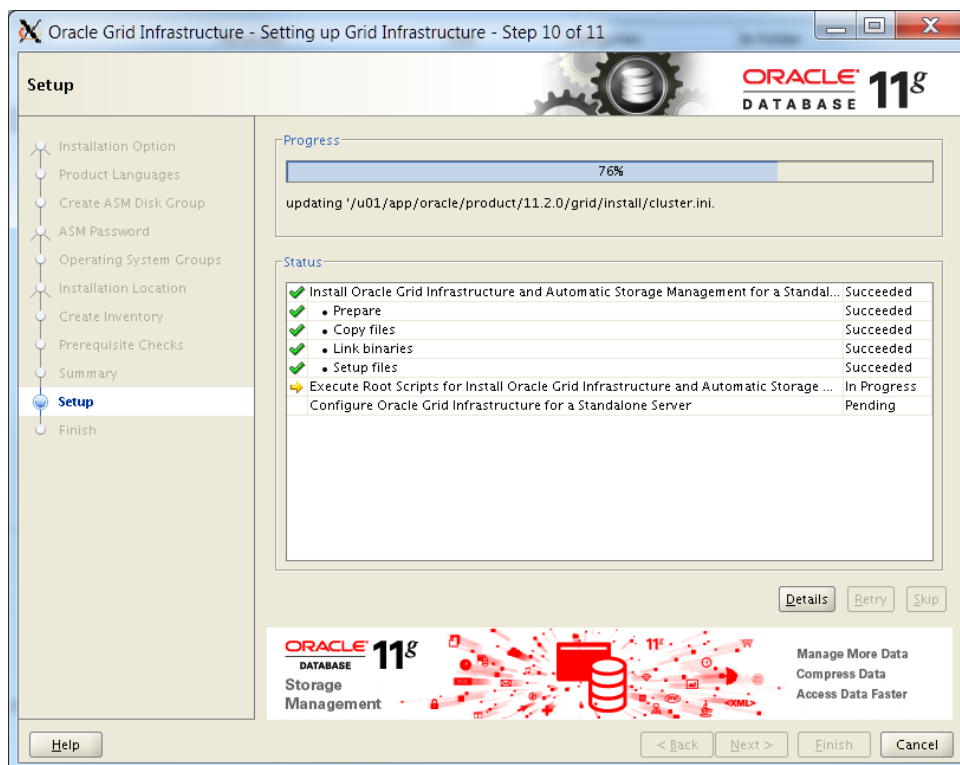


14. If other errors appear, the installer describes the problems and suggests solutions. Correct those errors. If the only remaining errors are the previously mentioned RPMs, select Ignore All, and click Next.
15. On the Step 9 screen, click Finish to start the installation. A session log can be found in directory `/u01/app/orainventory/logs/`.

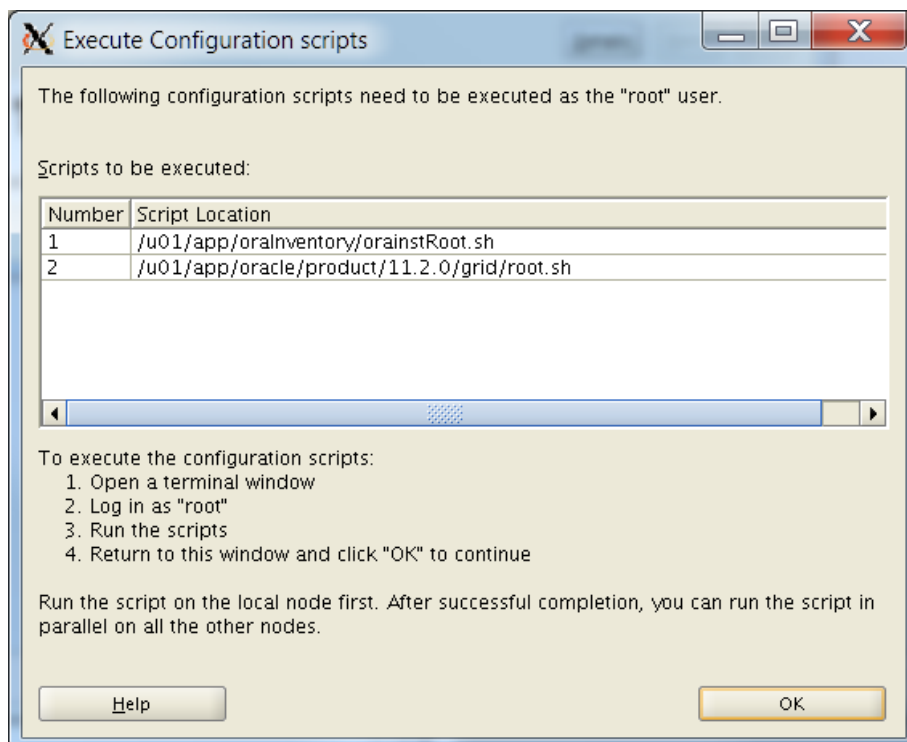




16. The Step 10 screen displays the progress of the installations.



17. At the end of the software installation, the Execute Configuration scripts window appears.





One to two scripts need to be run as root. Before running them (in particular, the `root.sh`) a Grid Infrastructure–specific workaround must be implemented as the Oracle High Availability service, an optional but useful watchdog, does not install properly in the `upstart` system-daemon environment that Red Hat Enterprise Linux 6.2 uses.

- a. As root, create the file `/etc/init/ohasd.conf` with these contents:

```
# What: oracle high-availability watchdog
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost

start on runlevel [35]
stop on runlevel [016]

respawn
exec /etc/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

- b. Run the first script as root:

```
# /u01/app/oralInventory/orainstRoot.sh
```

- c. Run the second script as root

```
# /u01/app/oracle/product/11.2.0/grid/root.sh
```

If there is a warning that the “Oracle High Availability Services startup failed”, start a new root login session, and signal the system to start the `ohasd` daemon from that session:

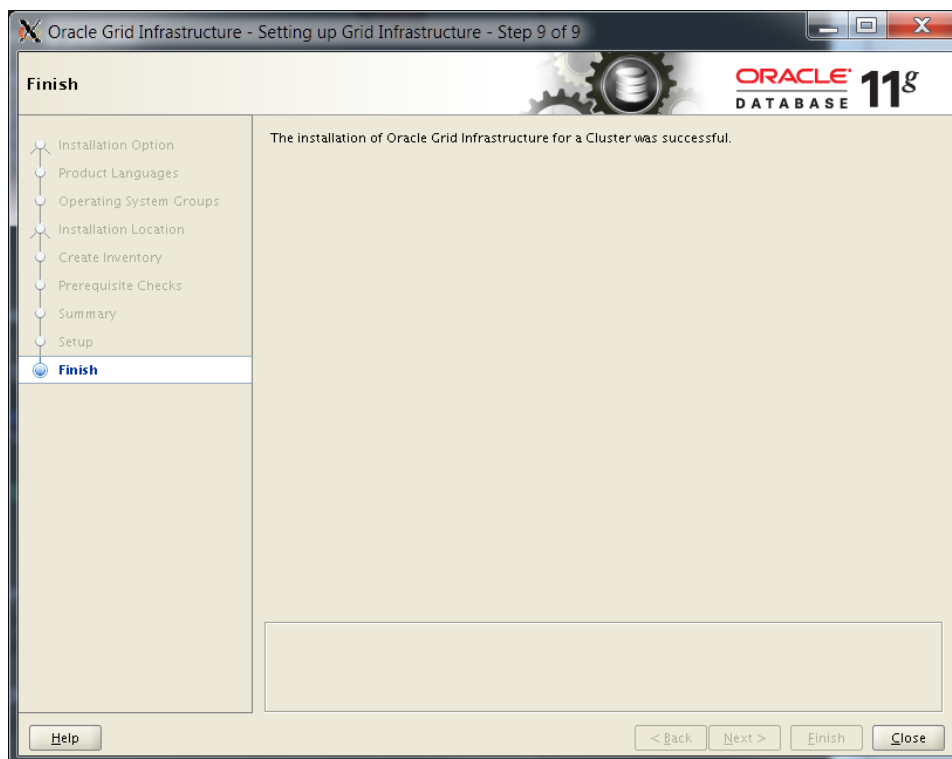
```
# initctl start ohasd
```

The second script should then run to completion.

18. Click OK in the Execute Configuration scripts.



19. On the Step 9 screen, click Close to end the installation.



4.4.2 Installing Oracle Database 11g Release 2 on a Standalone Server

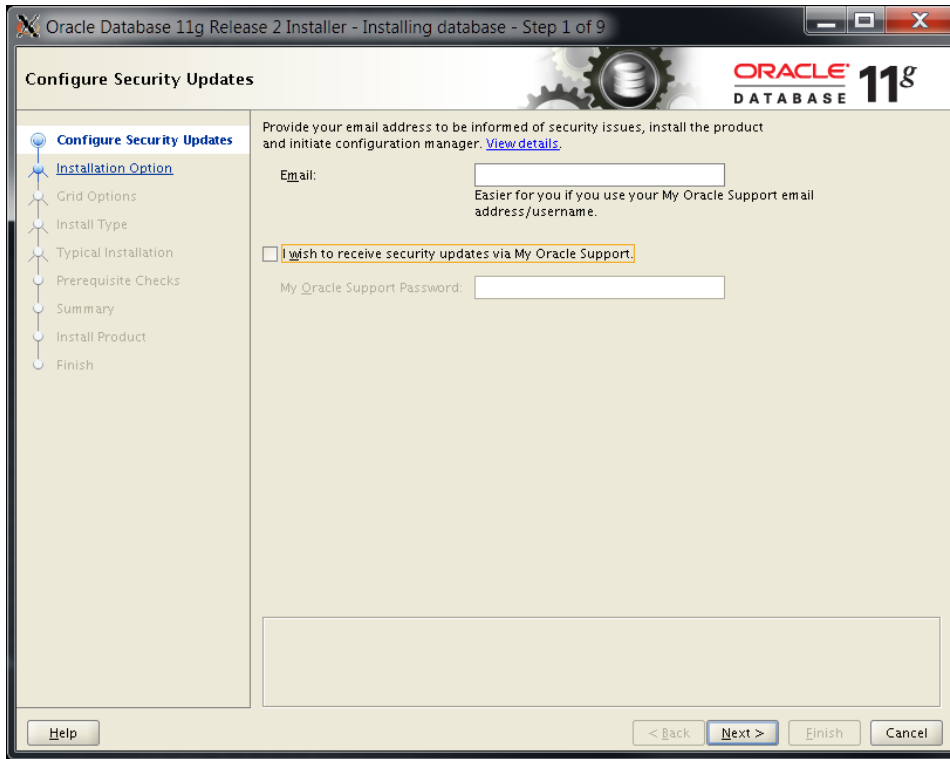
1. Log into the system and switch to the oracle account.
2. Unzip the Oracle Database software distribution files into /home/oracle. If the Oracle variables for hostname and base are the same as before, run the installer. The installer will create a log file in the directory /u01/app/orainventory/logs/.

```
$ cd /home/oracle
$ unzip linux.x64_11gR2_database_1of2.zip
$ unzip linux.x64_11gR2_database_2of2.zip
$ cd database
$ ./runInstaller
```

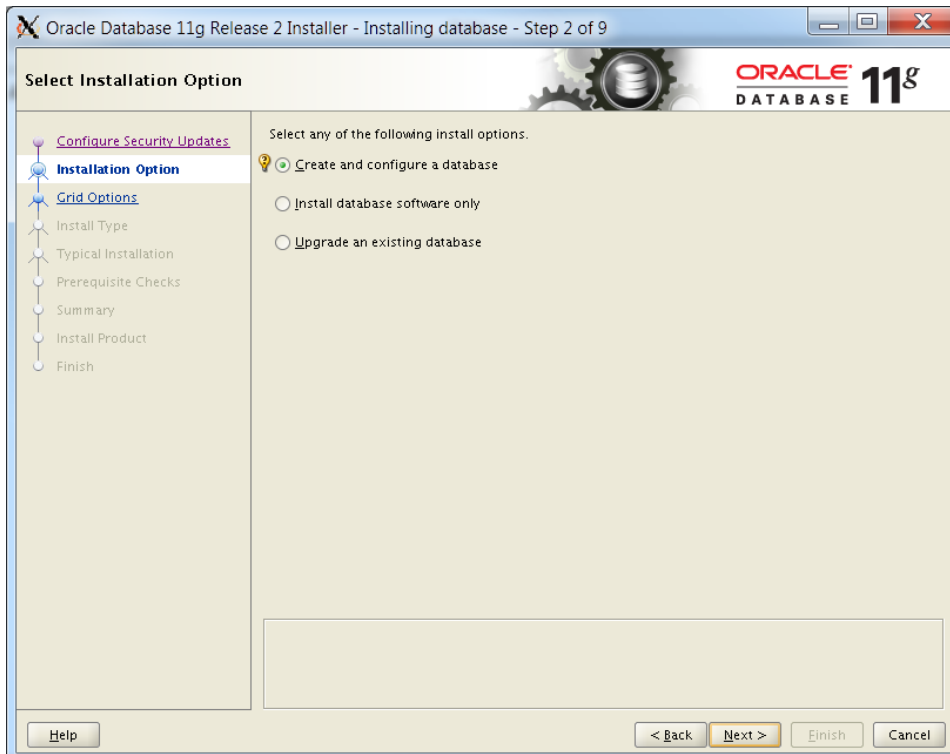
3. Type y and press Enter to bypass the error message regarding /usr/bin/xdpyinfo, as that package is neither installed nor needed.



4. On the Step 1 screen, fill out the contact information if desired, or uncheck the box, and click Next.

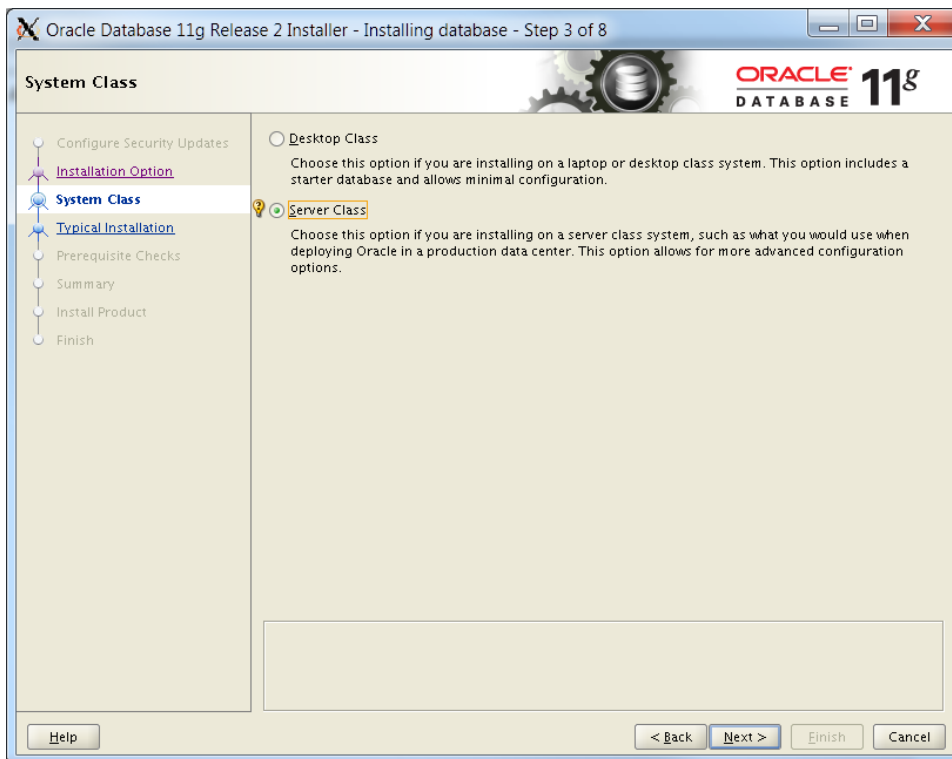


5. On the Step 2 screen, select Create and configure a database, and click Next.

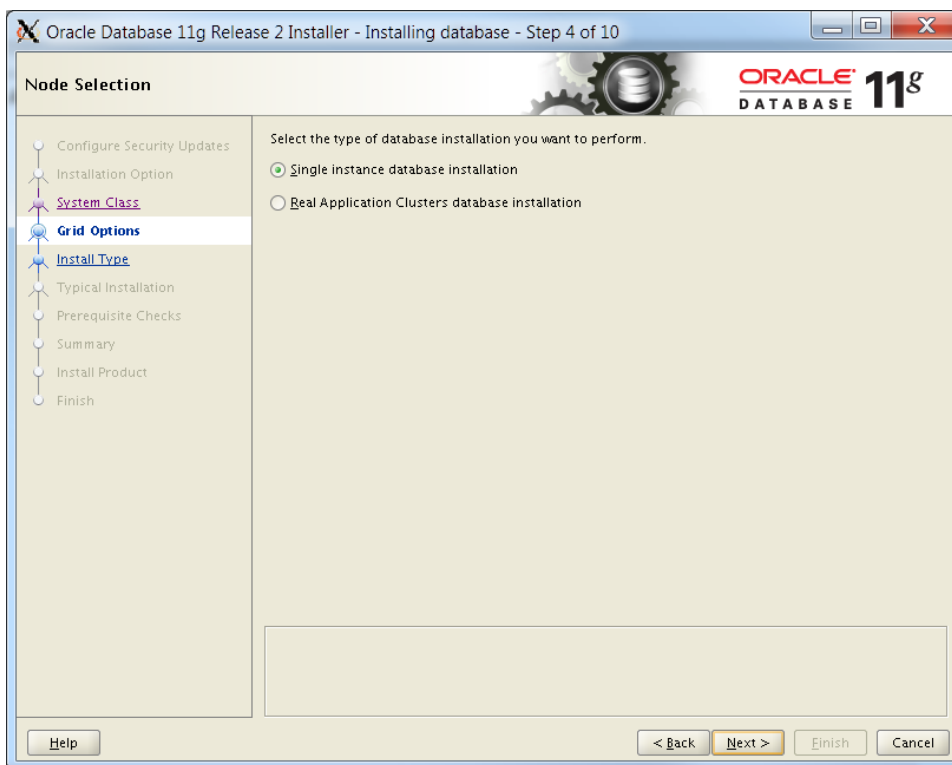




6. On the Step 3 screen, select Server Class, and click Next.

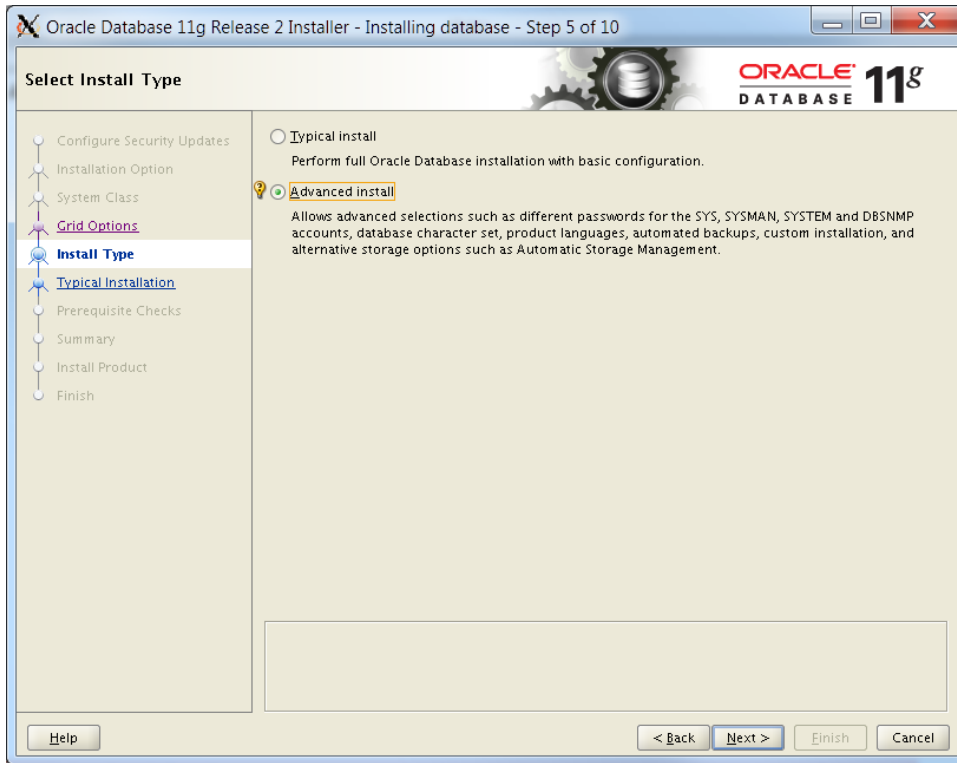


7. On the Step 4 screen, select Single instance database installation, and click Next.

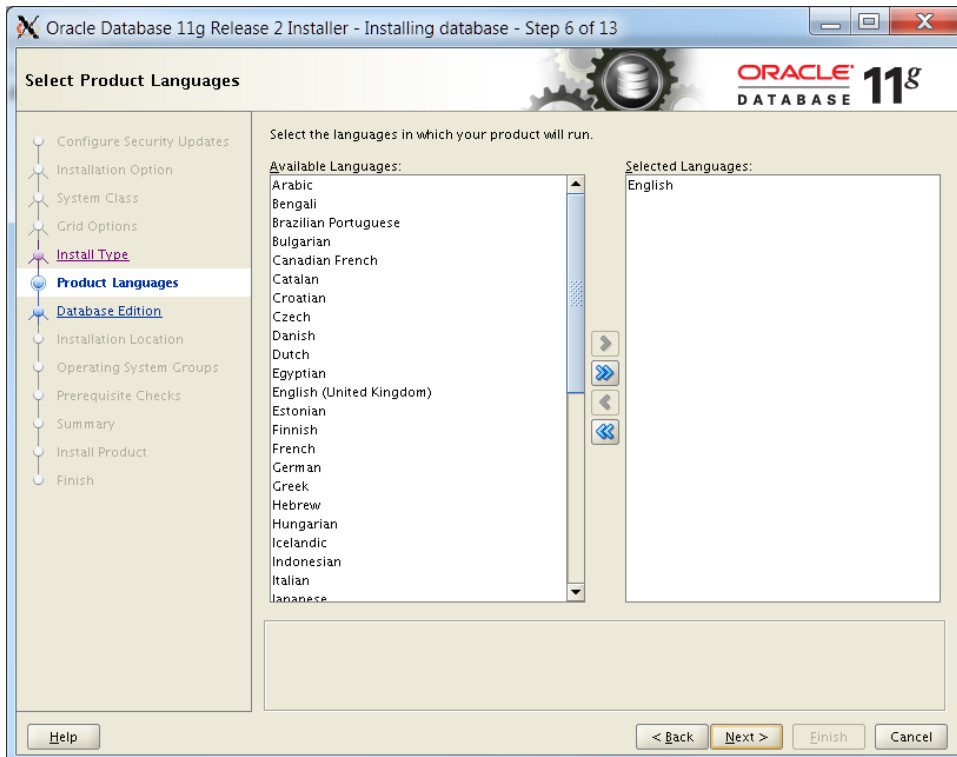




8. On the Step 5 screen, Select Advanced install, and click Next.

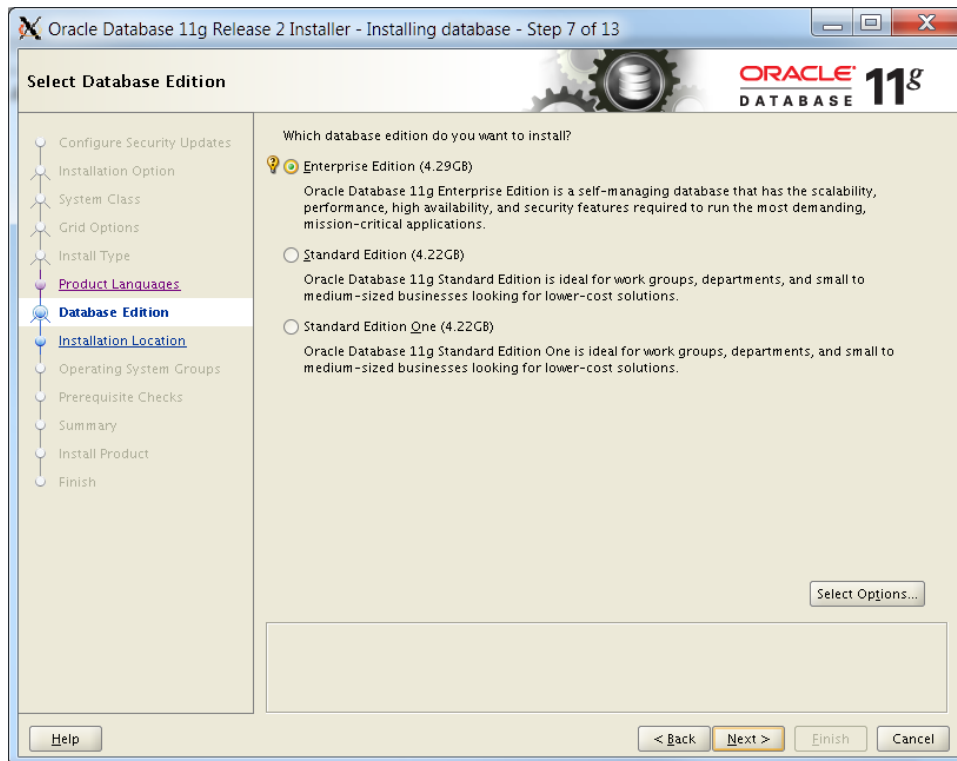


9. On the Step 6 screen, click Next to use English for the software.

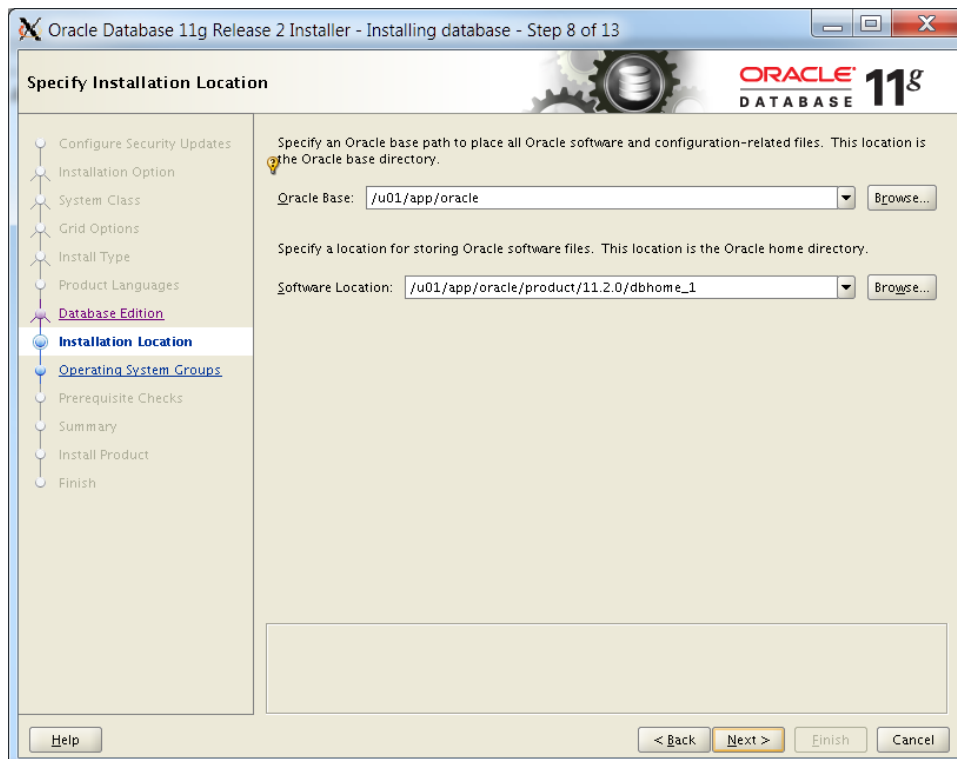




10. On the Step 7 screen, Select Enterprise Edition, and click Next.

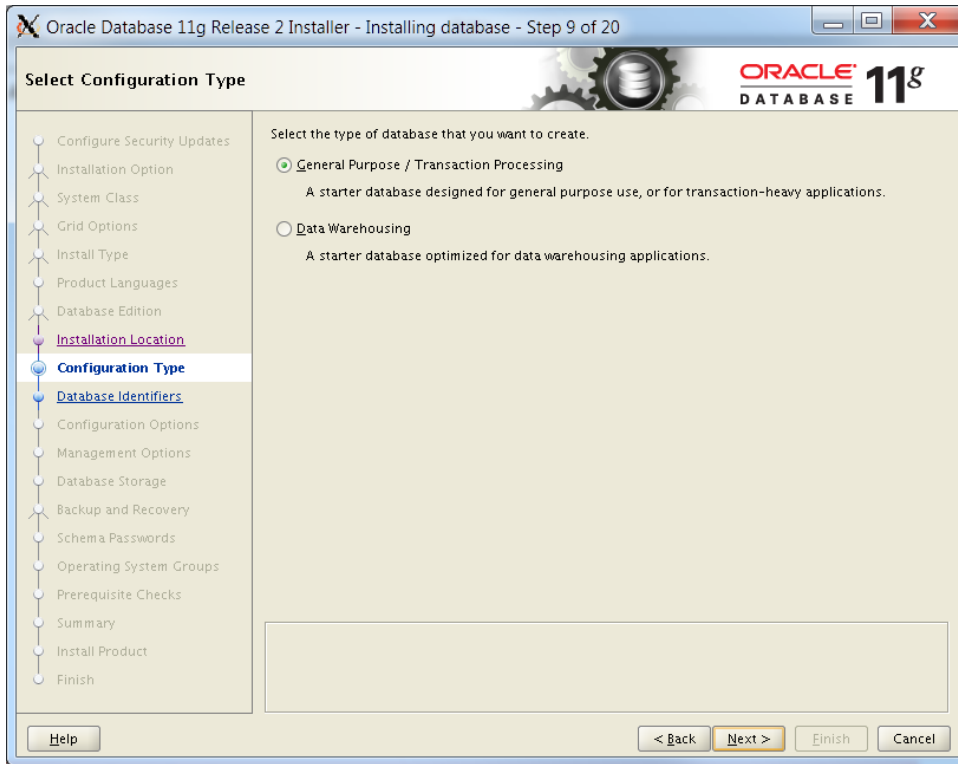


11. On the Step 8 screen, accept the default locations, and click Next.

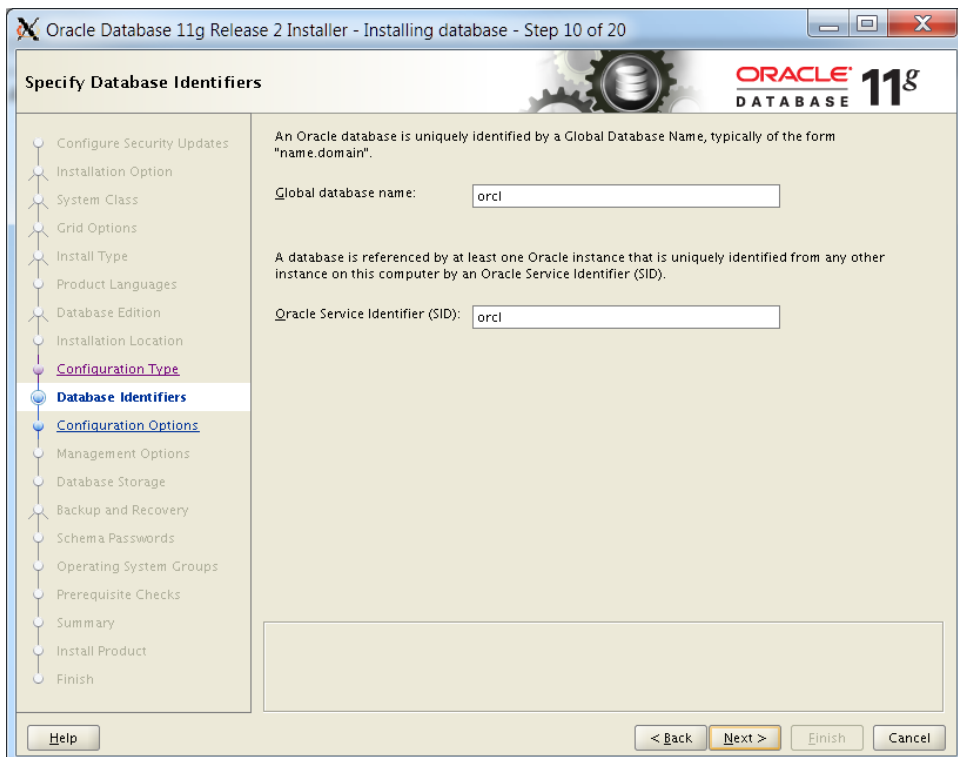




12. On the Step 9 screen, select General Purpose / Transaction Processing, and click Next.

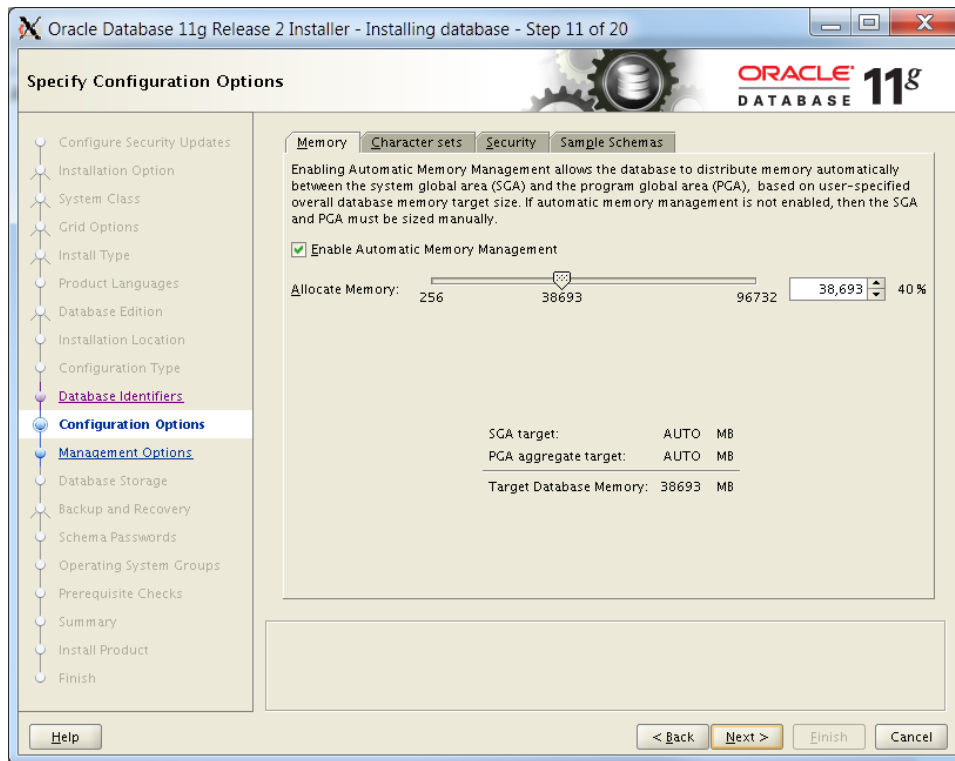


13. On the Step 10 screen, enter the database name and SID, and click Next.

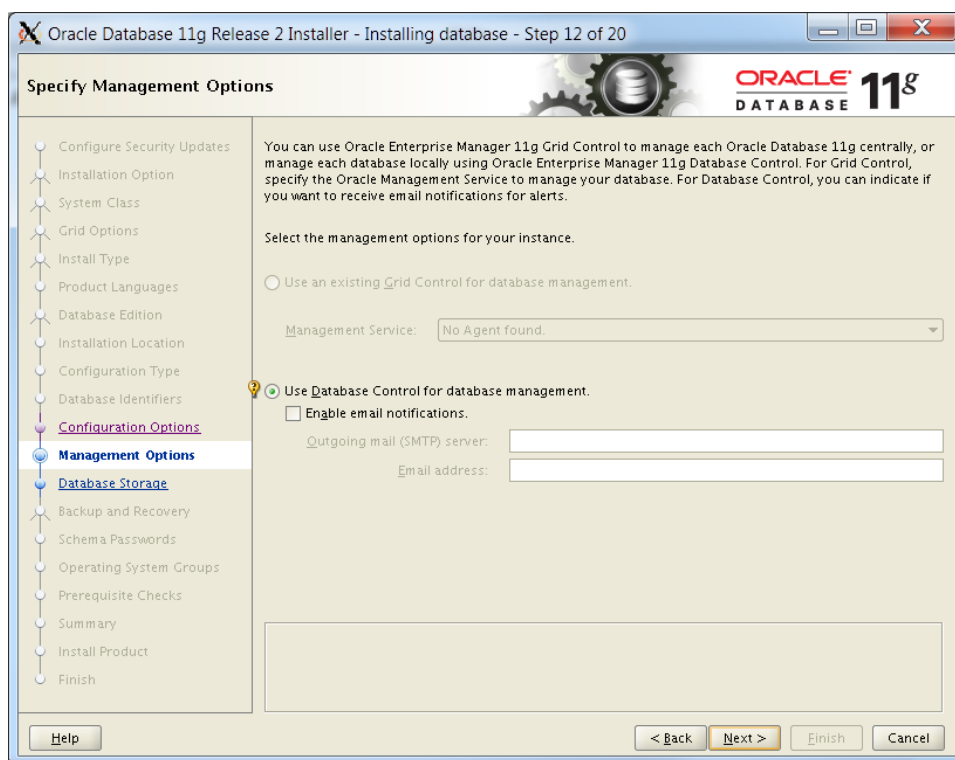




14. On the Step 11 screen, select the defaults for automatic memory management (AMM), and click Next. Note: If you plan to use static huge pages, you must disable AMM.

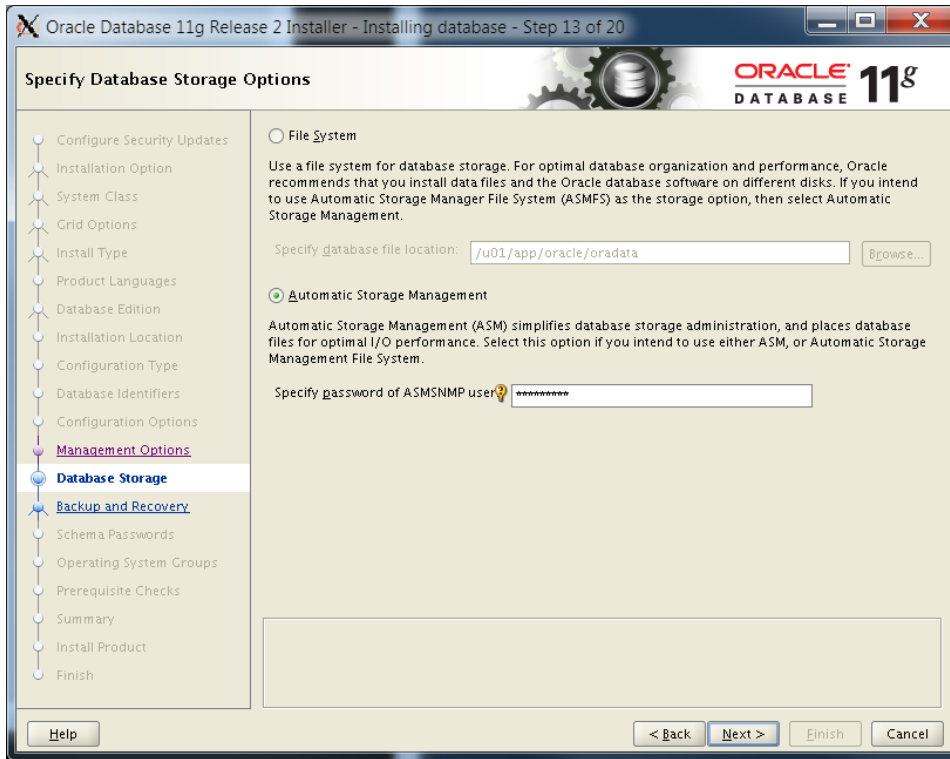


15. On the Step 12 screen, select Use Database Control for database management, and click Next.

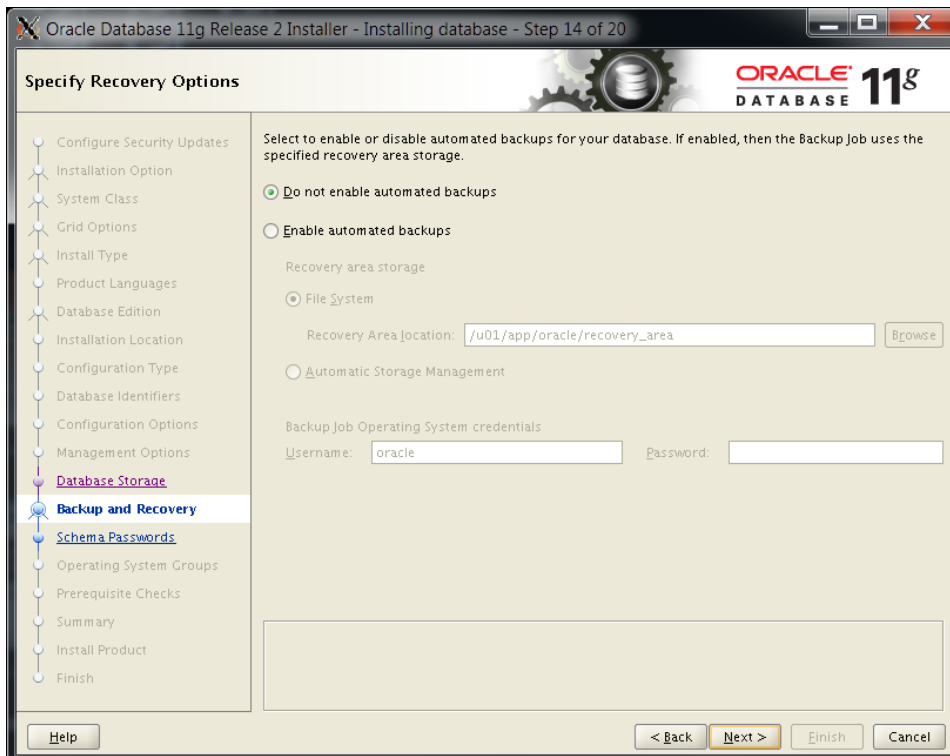




16. On the Step 13 screen, select Automatic Storage Management (for this example) and enter the password for the ASMSNMP user as in Section 4.3, and click Next.

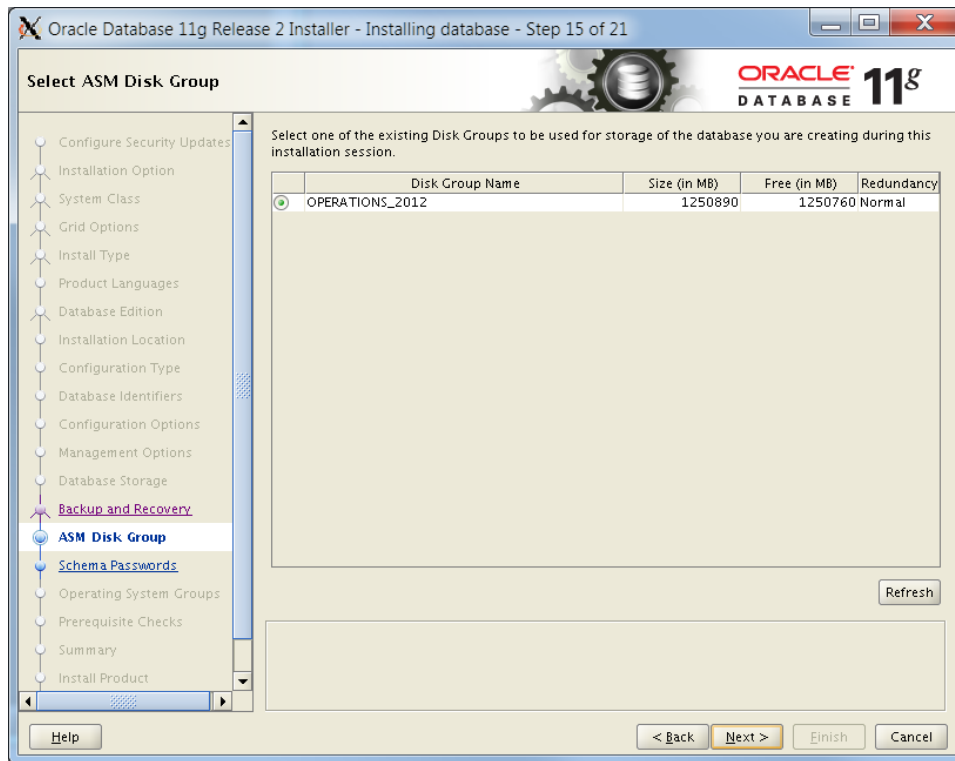


17. On the Step 14 Screen, keep the default, and click Next.

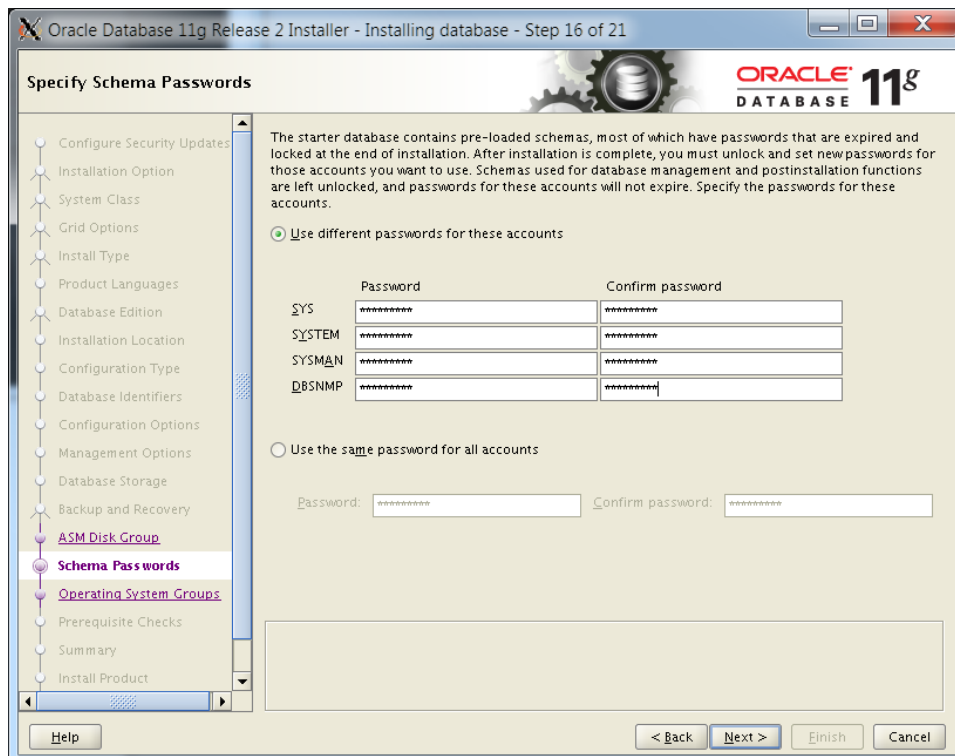




18. On the Step 15 screen, select the ASM disk group created during the installation of Oracle Grid Infrastructure, and click Next.

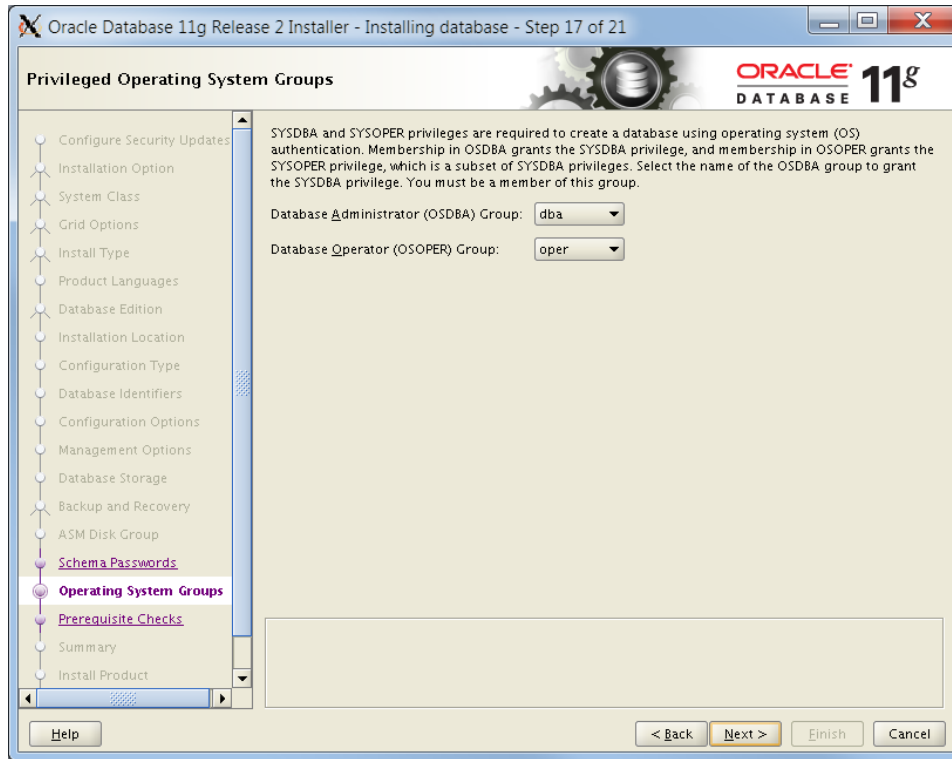


19. On the Step 16 screen, enter the passwords for the database accounts, and click Next.

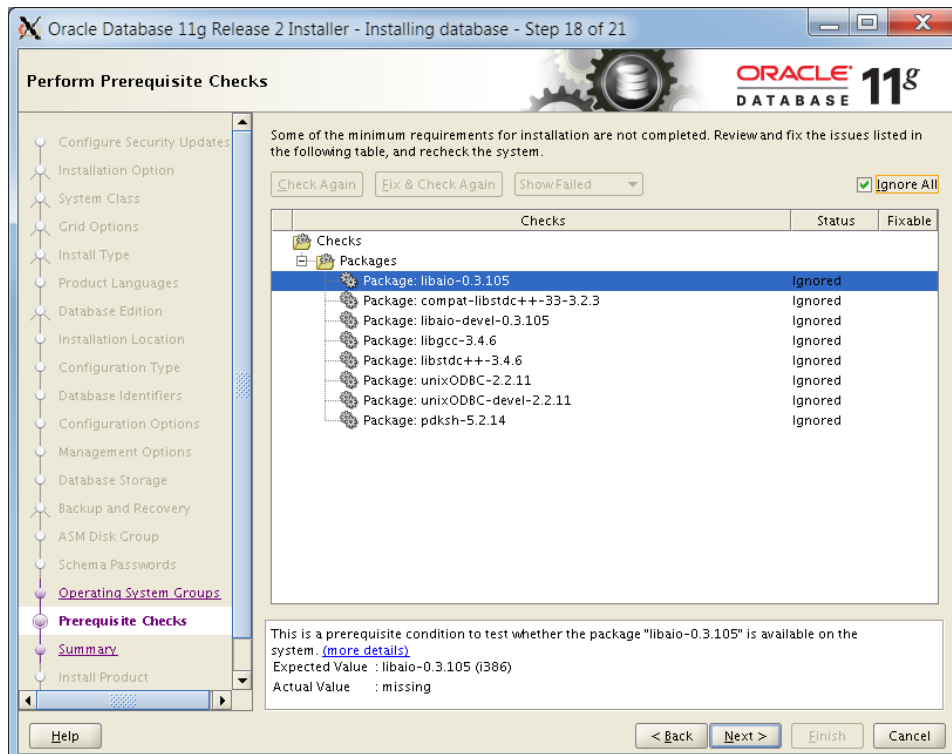




20. On the Step 17 screen, click Next to accept the defaults for the system groups, created earlier in Section 4.2.4.

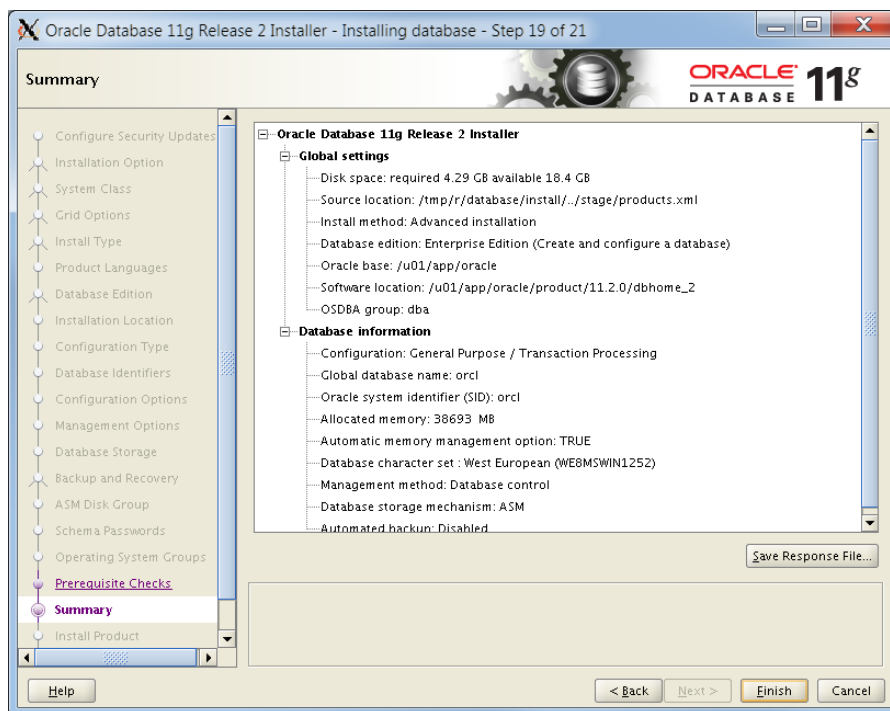


21. On the Step 18 screen, select Ignore All, and click Next.

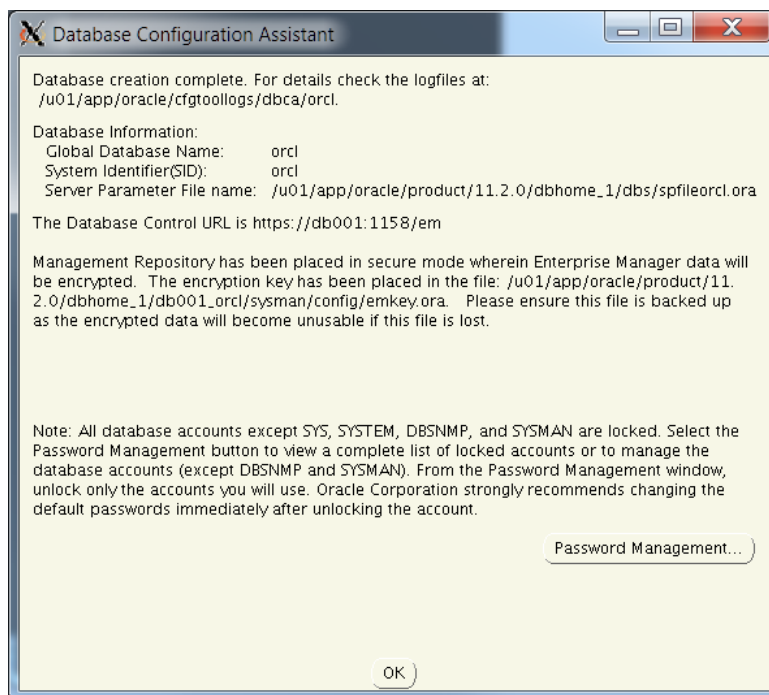




22. On the Step 19 screen, click Finish to begin the installation.

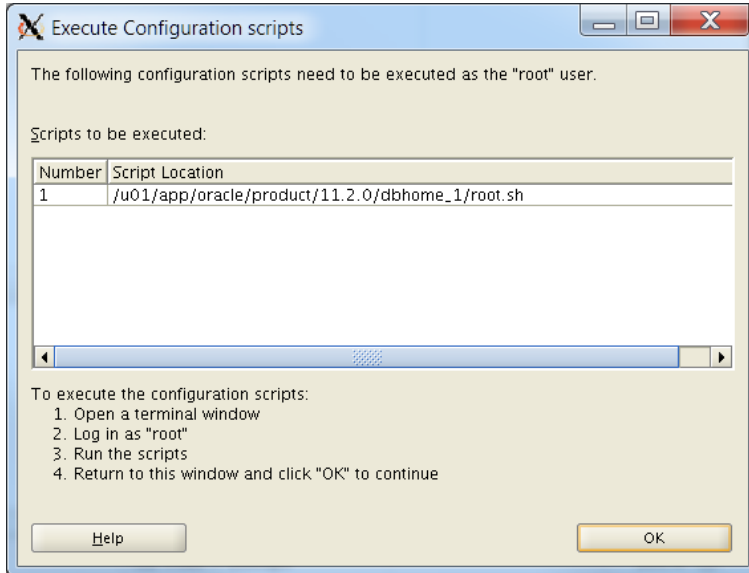


23. During installation, several screens from the Database Configuration Assistant may appear, for example, to unlock database management accounts. Follow their instructions, if desired.



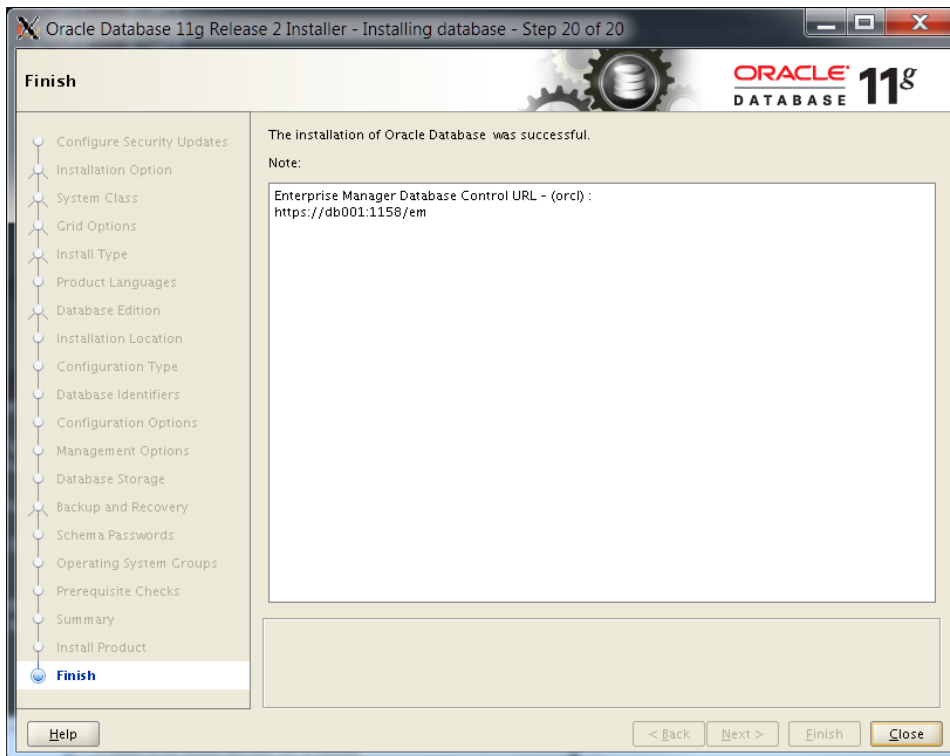


24. The Execute Configuration Scripts screen appears with a list of scripts to run as root. Open a terminal, log in as root, and run the script(s).



25. Click OK after the scripts complete.

26. On the final screen, click Close to end the installation





4.4.3 Post-installation Steps

Besides performing the post-installations steps recommended by Oracle (modifying the oracle account's path and environment variables), you can now delete the unzipped software distributions, if desired, to save space.

```
# cd /home/oracle  
# rm -rf grid database
```

Starting with 11g Release 2, Oracle has deprecated the dbstart and dbshut utilities for starting and stopping a database instance. In particular, the previous (10g) instructions for starting the database at boot time should not be followed. Instead, Oracle recommends configuring the Oracle Restart feature to automatically restart the database, the listener, Oracle ASM (if needed), and other Oracle components after a restart or after a hardware or software failure.



5 Database Storage Examples for Standalone Oracle Database 11g Release 2

To provide concrete examples, consider the following four database storage options. (See Figures 5-1 through 5-4) The scenarios are similar at a high level; viz., data, tablespaces, indices, and log files can be assigned to different physical devices in order to balance load, to effectively use a low-latency device, or to save cost.

Choosing which database entities should go where is best done by collecting database and system performance data to identify bottlenecks or resource starvation. Oracle provides tools to measure the performance of database subsystems, and database instances. Red Hat Enterprise Linux provides tools to measure the performance of the server's subsystems (virtual memory, CPU, RAM, etc.) as well as tools to measure the performance of data transfer to both internal and external devices (network and storage devices).

To be definite, one 100GB database was created and deployed for each storage scenario. This database was configured to support an expected Decision Support Suite business workload of requests and updates. The sections below provide recommended Red Hat Enterprise Linux tunings.

Often, Oracle data files live on ordinary EXT4 file systems, which are created on a partitioned storage device. The instructions for creating these files systems will not be reproduced here. Instead, this section presents instructions for preparing storage for Oracle ASM, as they may be unfamiliar to some. Choose the method that best suits your organization's needs.

5.1 Scenario 1: SAN Only

Perhaps the most typical storage for Oracle databases is a homogeneous set of HDDs, which could be internal to the server, external but directly attached via a SAS link, on an NFS file system, or presented as logical LUNs on a SAN (iSCSI or FC).

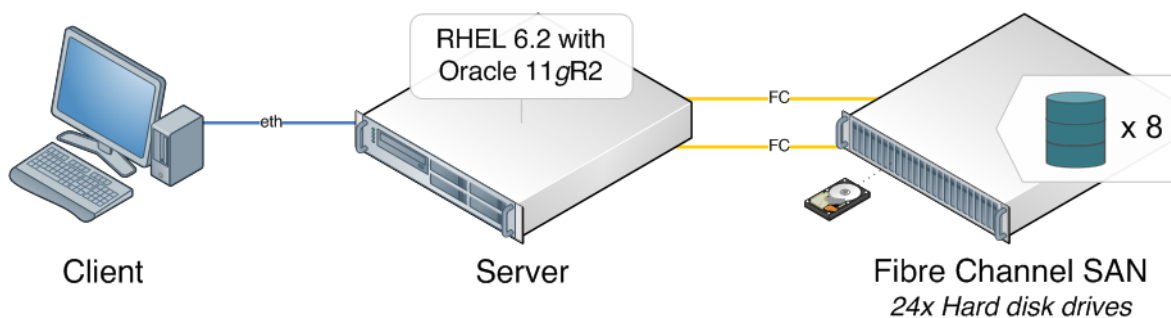


Figure 5-1: The first reference storage solution. All database files and logs are on the same kind of storage device, Fibre Channel SAN.

Oracle's OFA standard has provided guidelines for allocating files across this storage configuration.

In this document, one IBM DS3524 storage array of 24 (74GB 15,000 RPM) SAS drives connected via two Fibre cables to a dual-port 8 Gbps QLogic™ HBA provided the HDD storage. Three 250GB LUNs were created from a RAID 10 pool. IBM's native multipathing



software, rather than the native Linux device-mapper modules, provided two active-passive paths to each LUN and balanced the traffic between the FC paths.

Oracle ASM software was used to create disk groups, but the Linux utilities and the udev subsystem, rather than Oracle ASMLib utilities, were used to create and present the storage LUNs to ASM.

5.1.1 Connecting to the IBM DS3524 Storage Array

After installing the QLogic HBA in the server and configuring it from UEFI, its firmware was upgraded. Its connection to the array was a simple path-redundant, two-cable topology. The DS3524 management software package, SM10.77_Linux_64bit_x86-64_SMIA-10.77.x5.28.tgz, was installed.

The multipath software was configured following the instructions in the *Red Hat Enterprise Linux 6 DM Multipath Configuration and Administration* manual. The three LUNs are accessed through devices `sdb`, `sdc`, and `sdd`.

If database files will be stored on ordinary file systems and ASM will not be used, skip to section 5.1.5.

5.1.2 Preparing the LUNs for ASM without ASMLib

ASM expects each LUN to have a single partition. The `parted` command is ideal, and you can create the partitions on devices `sdc` and `sdd` as follows:

```
# parted /dev/sdc mklabel gpt mkpart primary "1 -1"  
# parted /dev/sdd mklabel gpt mkpart primary "1 -1"
```

5.1.3 Using Linux udev Instead of Oracle ASMLib

Oracle's ASMLib utilities provide one means for creating and manipulating ASM objects, but are not available on all OSs. However, standard Linux commands can be used instead.

Once the LUNs have been created, partitioned, and presented to the system, use the udev subsystem to ease later maintenance by isolating the devices to an Oracle-specific directory, and to automatically allow access by Oracle processes.

We use UDEV to map LUN `/dev/sdb` to `/oracleasm/disks/DSK1`, creating directories as necessary, and to change the LUN's ownership to `oracle:oinstall`. Device `/dev/sdb` is handled similarly.

The device attributes used in the udev rules template are found by running the following command on each device (here as an example, `/dev/sdc`) and substituting the corresponding values in the example below:

```
# scsi_id --page=0x83 --whitelisted --device=/dev/sdc  
3600605b00162840001e285f91316df21
```

With this information, you can isolate the disk devices to be used by ASM to their own directory (`/dev/oracleasm/disks`), and automatically change their ownership and access



permission so that ASM can read them. Create a new file, `/etc/udev/rules.d/99-asm.rules`, with contents similar to the following:

```
# What: oracle database: devices to be used by ASM
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost

## Important: each rule must either be a single line or extend to multiple lines with backslashes.

KERNEL=="sd*", PROGRAM=="scsi_id --page=0x83 --whitelisted --device=/dev/%k", \
    RESULT=="3600605b00162840001e285f91316df21", \
    SYMLINK+="oracleasm/disks/DSK1", OWNER="oracle", GROUP="dba", MODE="0660"

KERNEL=="sd*", PROGRAM=="scsi_id --page=0x83 --whitelisted --device=/dev/%k", \
    RESULT=="360080e50001c2c8600001a254eeb7a3d", \
    SYMLINK+="oracleasm/disks/DSK2", OWNER="oracle", GROUP="dba", MODE="0660"
```

These rules will take effect after the next reboot.

Occasionally, when reusing a LUN, ASM may detect a previous disk header and refuse to reuse the disk. If that is the case, clear the header by writing 25 MB of nulls, and repartition the LUN. For example:

```
# dd if=/dev/zero of=/dev/oracleasm/disks/DSK1 bs=1M count=25
```

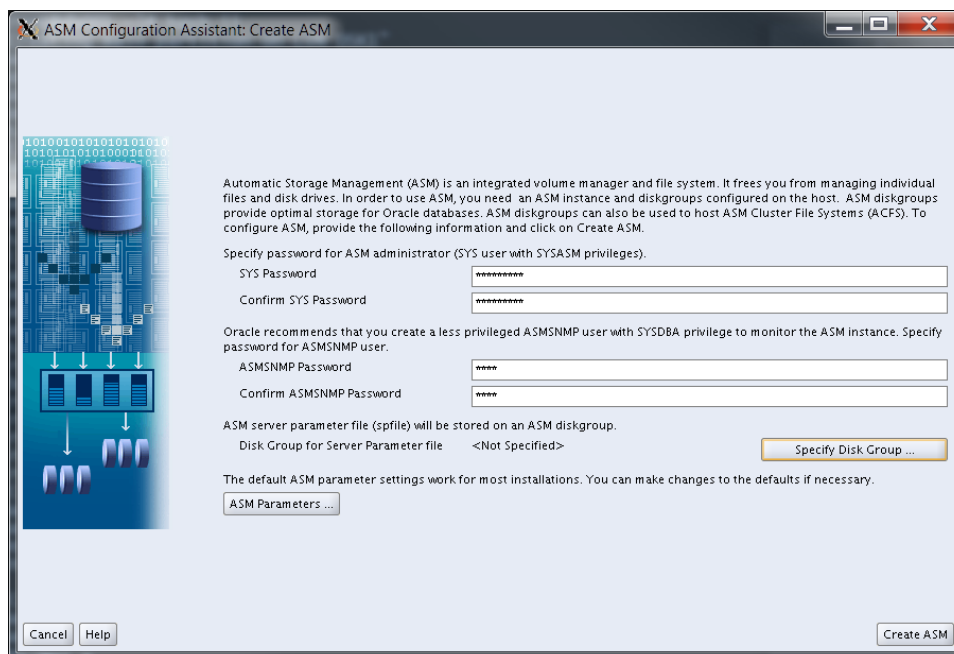
5.1.4 Creating ASM Disk Groups

1. Log into the system as an ordinary user.
2. Switch to the oracle user using `su – oracle`.
3. Start Grid Infrastructure.
4. As in Section 4.3, configure X Window for the GUI utility.
5. Start the ASM configuration assistant.

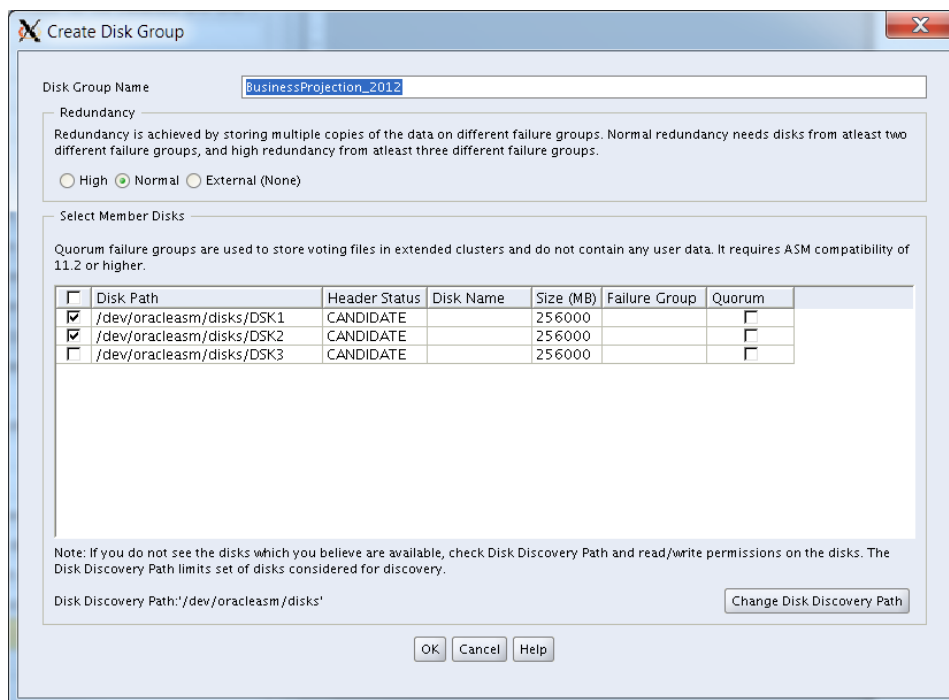
```
# /u01/app/11.2.0/grid2/bin/asmca
```



6. On the first screen, enter passwords for the ASM administrator, and ASM SNMP user.



7. Click Specify Disk Group...
8. On the Create Disk screen, enter the name of the main disk group, and click Change Disk Discovery Path, enter `/dev/oracleasm/disks`, and click OK.
9. The IBM LUNs will be listed as CANDIDATES. Select two, and click OK.





5.1.5 Creating the Database

To create the database, rerun the Oracle database configuration assistant (dbca) and specify the name as TESTDB1.

5.2 Option 2: SSD Only

5.2.1 Connecting the Fusion-io ioDisks

Two ioDisks were installed in two of the x3650 M3 PCIe (Gen2) x4 bays. Fusion-io provides software packages for Red Hat Linux Enterprise 6. These contain the utilities and source code for the drivers.

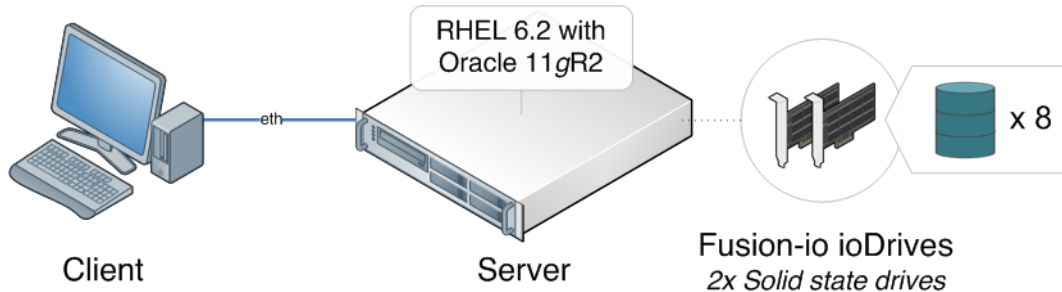


Figure 5-2: The second reference storage solution. All database files and logs are on the same kind of storage device, PCIe-attached internal SSDs.

Red Hat recommends these options for the ioDrive modules to improve CPU efficiency. Add the following to `/etc/modprobe.d/iomemory-vsl.conf`:

```
options iomemory-vsl tintr_hw_wait=50
options iomemory-vsl disable_msi=0
```

If database files will be stored on ordinary file systems and ASM will not be used, skip to Section 5.2.4.

5.2.2 Using Linux udev Instead of Oracle ASMLib

After the `iomemory-vsl` module is loaded, the SSDs are brought online with the following commands:

```
# fio-attach /dev/fct0
# fio-attach /dev/fct1
```

The Fusion-io ioDrives will be named `/dev/fioa` and `/dev/fiob`. Follow the instructions for using the Grid Infrastructure installer found in Section 5.1.3.

Once you have created, partitioned, and presented the LUNs to the system, use the `udev` subsystem to ease later maintenance by isolating the devices to an Oracle-specific directory, and to automatically allow access by Oracle processes.

`udev` will map LUN `/dev/fioa` to `/oracleasm/disks/DSK1`, creating directories as necessary, and change the LUNs ownership to `oracle:oinstall`. Device `/dev/fiob` is handled similarly.

The device attributes for the ioDrive devices are unique and simpler `udev` rules can be used.



Create a new file, `/etc/udev/rules.d/99-asm.rules`, with contents similar to the following:

```
# What: oracle database: ioDrive devices to be used by ASM
# When: 12/12/2011
# Who: J. B. Admin, jba@localhost

## Important: each rule must either be a single line or extend to multiple lines with backslashes.

KERNEL=="fioa", SUBSYSTEM=="block", \
    SYMLINK+="oracleasm/disks/DSK1", OWNER="oracle", GROUP="dba", MODE="0660"

KERNEL=="fiob", SUBSYSTEM=="block", \
    SYMLINK+="oracleasm/disks/DSK2", OWNER="oracle", GROUP="dba", MODE="0660"
```

These rules will take effect after the next reboot.

Occasionally, when reusing a LUN, ASM may detect a previous disk header and refuse to reuse the disk. If that is the case, clear the header by writing 25 MB of nulls, and repartition the LUN. For example:

```
# dd if=/dev/zero of=/dev/oracleasm/disks/DSK1 bs=1M count=25
```

5.2.3 Creating ASM Disk Groups

Follow the instructions for using the Grid Infrastructure installer found in Section 5.1.4.

5.2.4 Creating the Database

To create the database, rerun the Oracle database configuration assistant (dbca) and specify the name as TESTDB2.

5.3 Option 3: HDD for Most with SSD for Database Logs

5.3.1 Connecting the Storage

Follow the instructions in sections 5.1.1 and 5.2.1 to connect both the SAN and SSD storage.

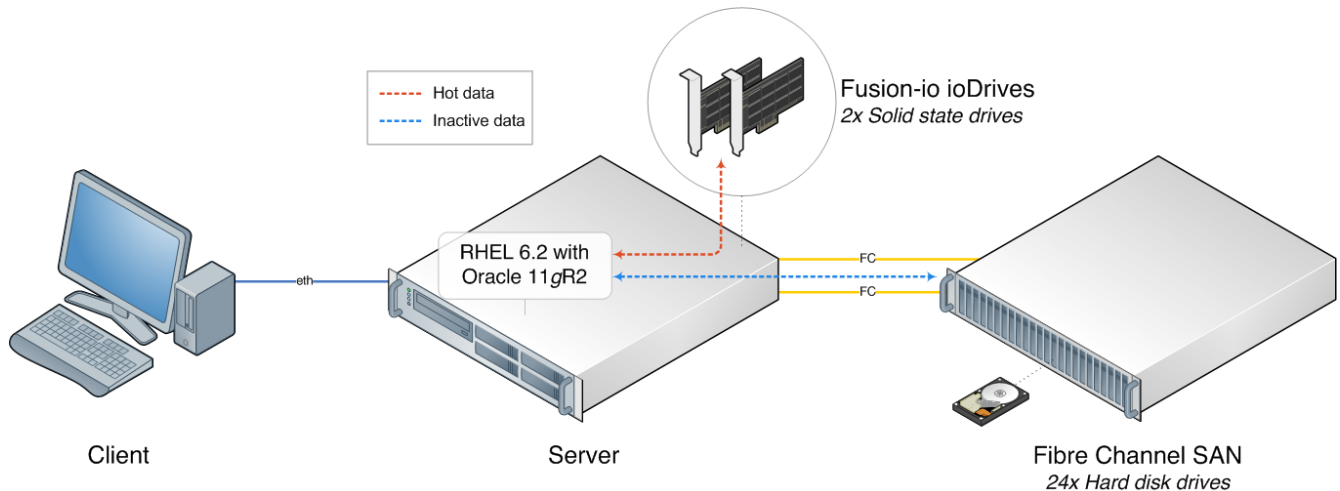


Figure 5-3: The third reference storage solution is also file based; both storage devices store database files. The administrator assigns frequently used (hot) data to the low-latency, smaller storage (SSD), and the bulk of the data to the Fibre Channel SAN.

If database files will be stored on ordinary file systems and ASM will not be used, skip to Section 5.3.4.

5.3.2 Using Linux udev

Follow the instructions for the HDD and SSD devices in sections 5.1.3 and 5.2.2 respectively to prepare the devices for ASM.

5.3.3 Creating ASM Disk Groups

Follow the instructions for using the Grid Infrastructure installer found in Section 5.1.4, but now create a second disk group using the SSD LUNs mapped to the directory `/dev/oraceasm`.

5.3.4 Creating the Database

To create the database, rerun the Oracle database configuration assistant (dbca) and specify the name as TESTDB3.

5.4 Option 4: Fusion-io directCache Fronting the SAN

5.4.1 Installing and Configuring Fusion-io directCache

Install Fusion-io's ioMemory and directCache packages following the vendor's instructions. Bind each Fusion-io disk to its backing store (e.g., `/dev/mapper/mpathcp1`), which will create a new storage device for accessing the storage through the SSD, e.g., `/dev/fdc0a`.

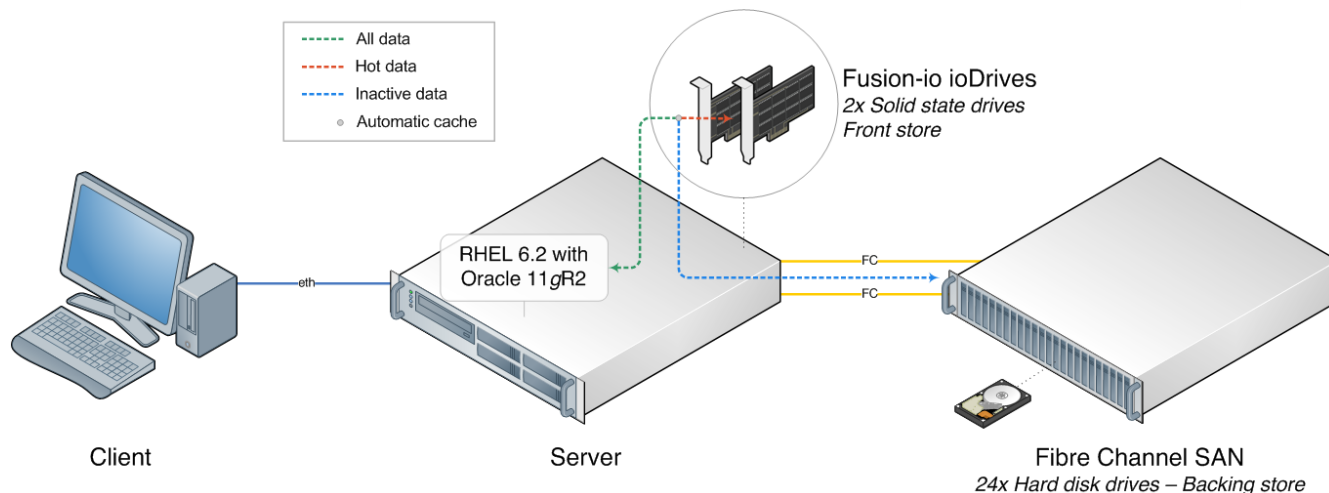


Figure 5-4: The fourth reference storage solution is not completely file-based. Although all database files are stored on the Fibre Channel SAN, the Fusion-io ioMemory software automatically caches frequently used data in the low-latency SSD, where it can be quickly sent to the CPU.

If ASM will not be used, update `/etc/fstab` and mount the front-end storage as in Section 4.3 and skip to Section 5.4.3.

5.4.2 Using ASM Disk Groups and Linux udev

Follow the instructions for the HDD and SSD devices in sections 5.1.3 and 5.2.2 respectively to prepare the devices for ASM. Be sure to replace the name of the SSD disk in the UDEV rules with the name of the front-end storage device, e.g., replace “fioa” with “fdc0a”.

5.4.3 Creating the Database

To create the database, simply rerun the Oracle database configuration assistant (dbca) and specify the name as TESTDB4.



6 Terms

Term	Meaning
ASM	Oracle Automatic Storage Management (ASM) provides one way to manage Oracle files by defining and working with abstracted disk groups rather than physical disks or files. ASM objects can be created and managed with the <code>asmca</code> program, which is bundled in Grid Infrastructure in 11g Release 2.
ASMLib	Oracle ASMLib is an optional management interface to ASM objects. As noted above, such objects can be created, assigned to Oracle resources, and destroyed using the <code>asmca</code> program. Red Hat Enterprise Linux, through its <code>udev</code> interface, supplements <code>asmca</code> by providing convenient management of physical disks.
Decision support system	Decision support systems use a database model that emphasizes <i>ad hoc</i> queries rather than periodic report generation.
DSS	Decision Support System
FC	Fibre Channel
Flash SSD	These SSDs use Flash memory (rather than DRAM, say) to store data.
HDD	Hard disk drives store data on spinning magnetic disks
Huge Pages	Virtual memory management simplifies and can offer performance benefits when memory is allocated and indexed in larger chunks (page sizes) of 2 MB, rather than the normal page size of 4 kB.
Oracle Grid Infrastructure	Oracle's Grid Infrastructure for a single database server provides utilities to manage Oracle disk groups via ASM, and to restart the database automatically.
Oracle RAC	Oracle Real Application Cluster (RAC) provides a multi-server, shared-disk, clustered database with a shared application cache.
RAC	See Oracle RAC
SAN	Storage area network
SSD	Solid-state drives store data in some kind of solid-state memory.
UDEV	Red Hat Enterprise Linux 6 uses <code>udev</code> as its device manager, especially those devices traditionally accessed through <code>/dev</code> .
VM	Virtual machine, also known as guest

Table 6-1: Terms



7 References

1. *Red Hat Performance Brief, Optimizing Fusion ioMemory on Red Hat Enterprise Linux 6 for Database Performance Acceleration*, https://access.redhat.com/sites/default/files/fusion_iodrives_on_rhel_6_for_database_performance_acceleration_0.pdf Sanjay Rao, August 2011.
2. *Red Hat Enterprise Linux 6 Security-Enhanced Linux User Guide*, http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/index.html. Red Hat, 2011.
3. *Performance Tuning Guide: Optimizing subsystem throughput in Red Hat Enterprise Linux 6*, http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security-Enhanced_Linux/index.html , Red Hat.
4. *Security-Enhanced Linux User Guide*, http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security-Enhanced_Linux/index.html, Red Hat.
5. *Oracle Database Installation Guide 11g Release 2 (11.2) for Linux*, http://docs.oracle.com/cd/E11882_01/install.112/e24321/toc.htm, Oracle.
6. *Oracle Grid Infrastructure Installation Guide 11g Release 2 (11.2) for Linux*, http://docs.oracle.com/cd/E11882_01/install.112/e22489/toc.htm, Oracle.
7. *ioDrive Data Sheet*, <http://www.fusionio.com/data-sheets/iodrive-data-sheet/>, Fusion-io.
8. *Fusion-io directCache*, <http://www.fusionio.com/data-sheets/directcache/>, Fusion-io.



Appendix A. Revision History

Revision 1.1 Thursday, 2 August 2012

Changed the resource limits for the oracle account so that the soft and hard limits were equal.

Clarified which steps in Section 4.3 applied to non-ASM LUNs.

Updated the UDEV rules for ASM volumes to use the `scsi_id` command to identify volumes.

Clarified the guidance that UDEV rules should be on a single line or continued with backslashes if on multiple lines.

Revision 1.0 Wednesday, 7 March 2012

Initial Edition