# 一、Linux下搭建11gR2 RAC的系统环境与软件版本：

## 1.1、硬件

VMware vsphere 5 虚拟机两台 ， ISCSI-SAN共享磁盘

## 1.2、操作系统

CentOS 5.9- x64 ，kernel版本 ：Linux 2.6.18-348.el5（默认）

## 1.3、Oracle软件

Oracle RAC：11gR2 for Linux x64版本。

需要安装的文件如下：

* linux.x64\_11gR2\_grid.zip（RAC）
* linux.x64\_11gR2\_database\_1of2.zip；linux.x64\_11gR2\_database\_2of2.zip（database）
* oracleasm-2.6.18-348.el5-2.0.5-1.el5.x86\_64.rpm；oracleasmlib-2.0.4-1.el5.x86\_64.rpm；oracleasm-support-2.1.7-1.el5.x86\_64.rpm（ASM相关RPM软件包，注意RPM包版本需和操作系统内核版本一致，故目前11gR2不支持CentOS 6.x版本）
* linux.x64\_11gR2\_examples.zip（可选）

# 二、搭建过程

搭建过程基本参考<http://wenku.baidu.com/view/7362ebd784254b35eefd3477.html>。

## 2.1、虚拟机和RDM磁盘设定

### 2.1.1、虚拟机硬件设定

1、省略设定vsphere平台过程。

2、通过vsphere client端设定虚拟机。至少要求2台虚拟机，每台配置2颗CPU，4GB内存，80G存储，2块E1000网卡，

### 2.1.2、安装操作系统设定

CentOS5.9 -x 64，注意选择安装包时，选择Desktop-Gnome。



为方便配置请在安装好的虚拟机上安装VMware-tools。

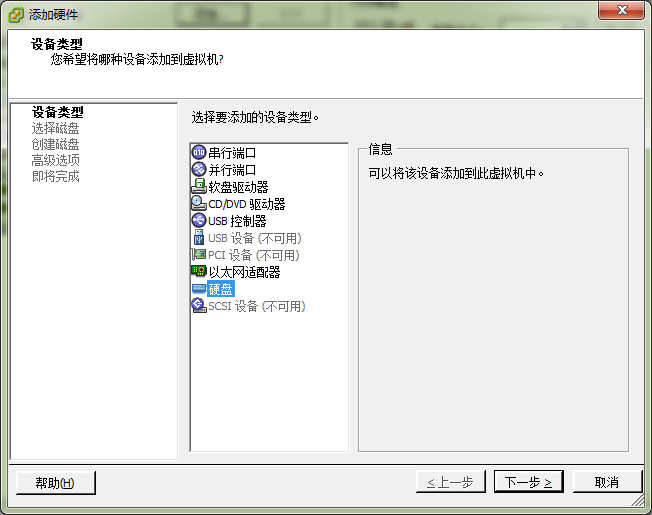
### 2.2.3、RDM磁盘设定

以下分别称两台虚拟机为node1和node2

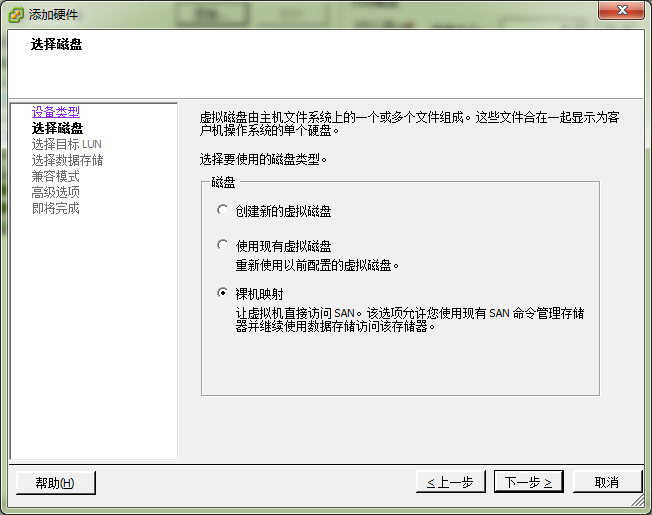
1. 省略在ISCSI-SAN设备上设定磁盘过程。
2. 挂载RDM磁盘，此处使用的是ISCSI协议下的共享磁盘。

#### 编辑虚拟机设置：node1

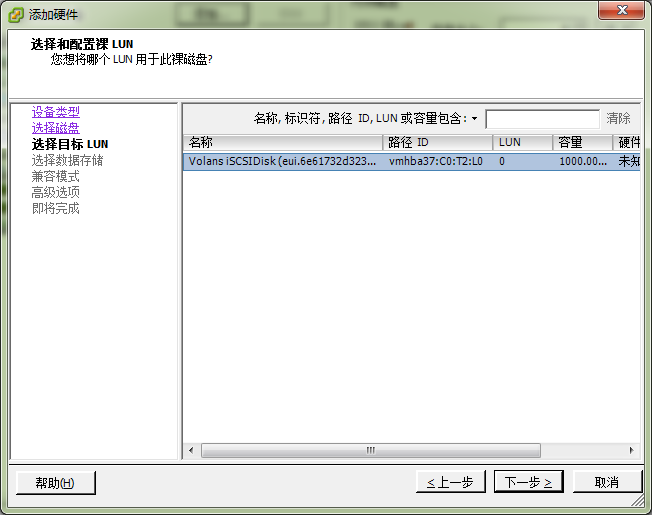
添加硬件，选择硬盘



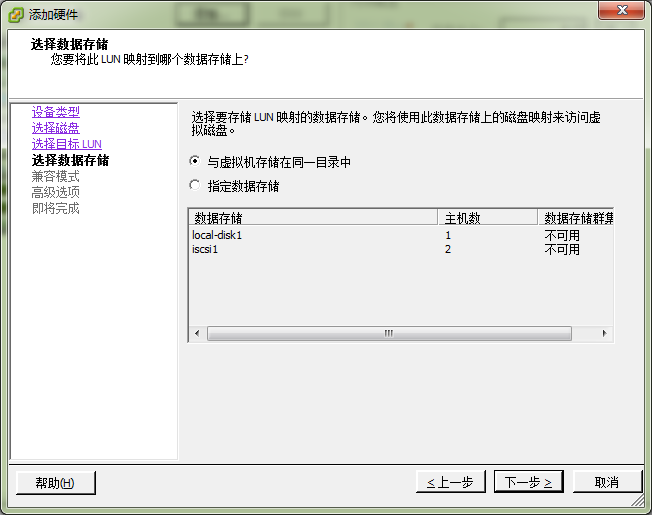
选择磁盘，裸机映射



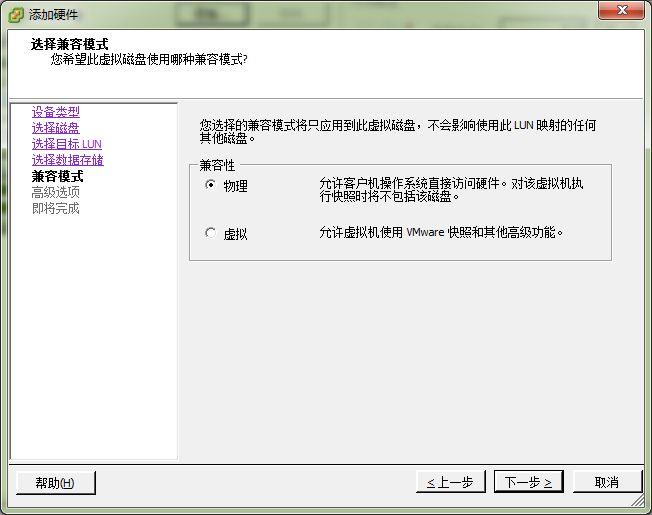
选择和配置裸LUN：选择之前规划的ISCSI硬盘(加载ISCSI硬盘的过程省略)。目前的Vsphere平台貌似有BUG，只能看到2个ISCSI设备。



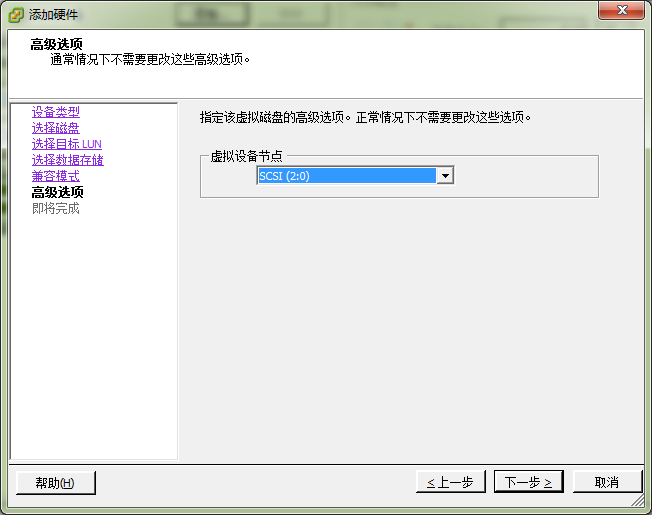
选择数据存储：与虚拟机存储在同一目录中



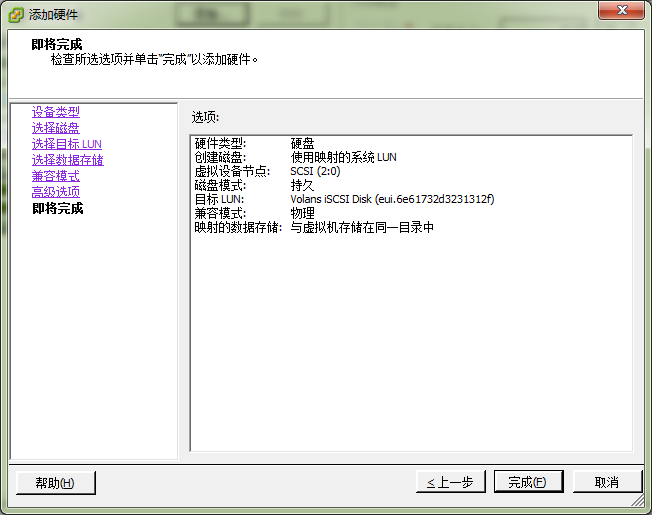
选择兼容模式：物理



高级选项：虚拟设备节点SCSI(2:0)



完成

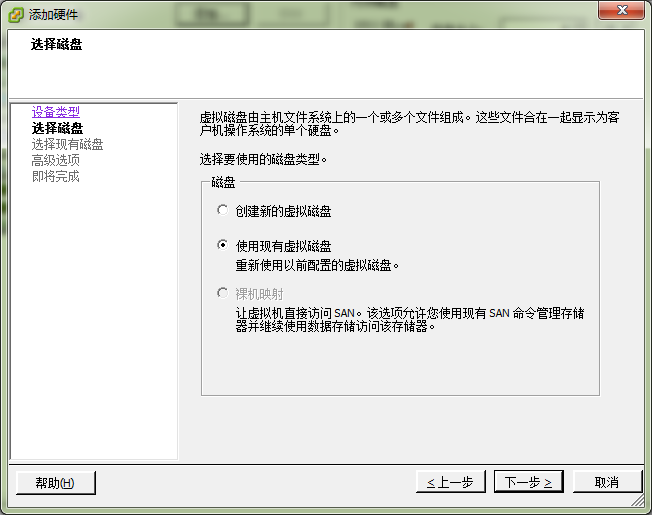


添加完成磁盘后，修改SCSI控制器的总线共享为物理。

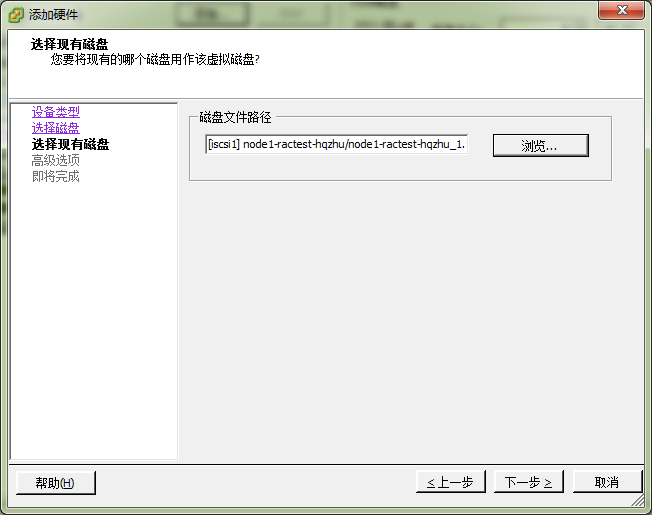


#### 编辑虚拟机设置：node2

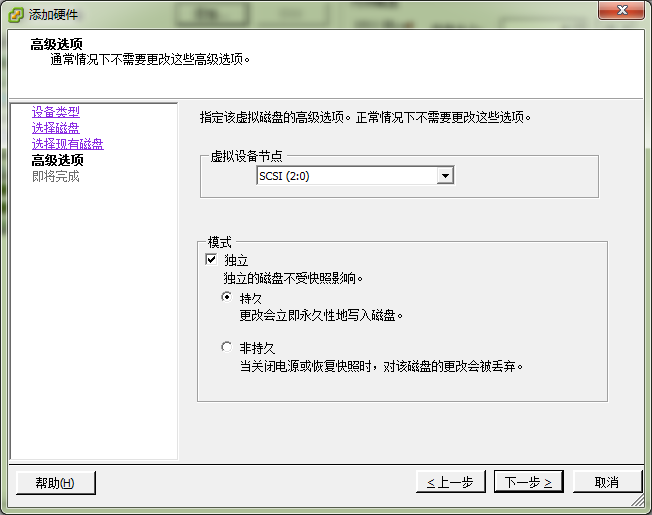
添加硬盘，选择磁盘：使用现有虚拟磁盘



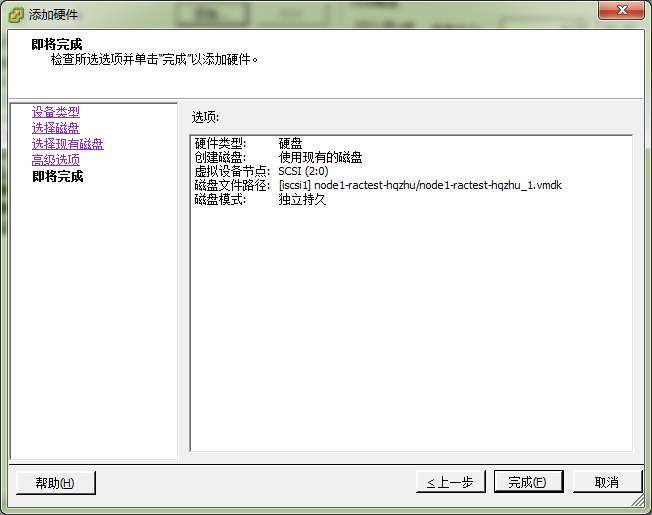
选择现有磁盘，找到node1所在存储上的目录中的RDM磁盘



高级选项，虚拟设备节点选择SCSI(2:0)，模式选择独立持久



完成



修改SCSI控制器总线共享类型为物理。



## 2.2、安装前准备工作

### 2.2.1、网络设定

需要特别说明一下，配置RAC的每台机器需要2块网卡，但需要至少3个IP地址，具体IP地址的含义如下：

Public IP：用于对外提供服务器的IP，一般使用共有IP（对内部提供服务的话，私有IP也是可以的）。需要手动配置在eth0上。

Private IP：用于服务器间通信的IP，一般使用私有IP。需要手动配置在eth1上。（真实环境中eth1和eth0应连接两个不同交换机。）

Virtual IP：用于对Public IP进行备份（故障），轮询（负载均衡）的IP。一般和Public IP在同一网段下。（不需要手动配置，但需要写入/etc/hosts，RAC安装时会自动配置。）

本文采用/etc/hosts文件的方式，而不采用name server的方式进行RAC的配置。所以，配置网络的工作如下。

#### Node1配置/etc/sysconfig/network

# more /etc/sysconfig/network

NETWORKING=yes

NETWORKING\_IPV6=no

HOSTNAME=node1

#### Node1配置/etc/sysconfig/network-scripts/ifcfg-eth0

# more /etc/sysconfig/network-scripts/ifcfg-eth0

# Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)

DEVICE=eth0

BOOTPROTO=static

IPADDR=166.111.141.31

NETMASK=255.255.255.192

GATEWAY=166.111.141.1

HWADDR=00:50:56:BC:1F:65

ONBOOT=yes

#### Node1配置/etc/sysconfig/network-scripts/ifcfg-eth1

# more /etc/sysconfig/network-scripts/ifcfg-eth1

# Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)

DEVICE=eth1

BOOTPROTO=static

IPADDR=192.168.1.31

NETMASK=255.255.255.0

HWADDR=00:50:56:BC:1F:66

ONBOOT=yes

HOTPLUG=no

#### Node1配置/etc/hosts

# more /etc/hosts

# Do not remove the following line, or various programs

# that require network functionality will fail.

127.0.0.1 localhost.localdomain localhost

166.111.141.31 node1.inetboss.com node1

166.111.141.32 node2.inetboss.com node2

192.168.1.31 node1-priv.inetboss.com node1-priv

192.168.1.32 node2-priv.inetboss.com node2-priv

166.111.141.41 node1-vip.inetboss.com node1-vip

166.111.141.42 node2-vip.inetboss.com node2-vip

166.111.141.43 node-cluster-scan.inetboss.com node-cluster-scan

#### Node2配置/etc/sysconfig/network

# more /etc/sysconfig/network

NETWORKING=yes

NETWORKING\_IPV6=no

HOSTNAME=node2

#### Node2配置/etc/sysconfig/network-scripts/ifcfg-eth0

# more /etc/sysconfig/network-scripts/ifcfg-eth0

# Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)

DEVICE=eth0

BOOTPROTO=static

IPADDR=166.111.141.41

NETMASK=255.255.255.192

GATEWAY=166.111.141.1

ONBOOT=yes

#### Node2配置/etc/sysconfig/network-scripts/ifcfg-eth1

# more /etc/sysconfig/network-scripts/ifcfg-eth1

# Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)

DEVICE=eth1

BOOTPROTO=static

IPADDR=192.168.1.41

NETMASK=255.255.255.0

ONBOOT=yes

HOTPLUG=no

#### Node2配置/etc/hosts

# more /etc/hosts

# Do not remove the following line, or various programs

# that require network functionality will fail.

127.0.0.1 localhost.localdomain localhost

166.111.141.31 node1.inetboss.com node1

166.111.141.32 node2.inetboss.com node2

192.168.1.31 node1-priv.inetboss.com node1-priv

192.168.1.32 node2-priv.inetboss.com node2-priv

166.111.141.41 node1-vip.inetboss.com node1-vip

166.111.141.42 node2-vip.inetboss.com node2-vip

166.111.141.43 node-cluster-scan.inetboss.com node-cluster-scan

### 2.2.2、建立oracle用户和组

#### Node1和node2运行以下脚本

# more addname.sh

#!/bin/bash

/usr/sbin/groupadd -g 1000 oinstall

/usr/sbin/groupadd -g 1200 asmadmin

/usr/sbin/groupadd -g 1201 asmdba

/usr/sbin/groupadd -g 1202 asmoper

/usr/sbin/groupadd -g 1300 dba

/usr/sbin/groupadd -g 1301 oper

/usr/sbin/useradd -m -u 1100 -g oinstall -G asmadmin,asmdba,asmoper,oper,dba -d /home/grid -s /bin/bash -c "Grid Infrastructure Owner" grid

/usr/sbin/useradd -m -u 1101 -g oinstall -G dba,oper,asmdba -d /home/oracle -s /bin/bash -c "Oracle Software Owner" oracle

#### Node1和node2分别执行以下命令

#passwd oracle

#passwd grid

### 2.2.3、配置信任关系

#### 配置grid用户环境，分别在node1和node2上执行下列命令

Node1

#su – grid

$ssh-keygen -t rsa

$ssh-keygen -t dsa

$cd .ssh

$cat \*.pub>authorized\_keys

Node2

#su – grid

$ssh-keygen -t rsa

$ssh-keygen -t dsa

$cd .ssh

$cat \*.pub>authorized\_keys

Node1

$scp authorized\_keys [grid@node2:/home/grid/.ssh/key\_dbs](mailto:grid@node2:/home/grid/.ssh/key_dbs)

Node2

$cat key\_dbs >>authorized\_keys

$scp authorized\_keys grid@node1:/home/grid/.ssh/

#### 测试信任关系

Node1,node2

$ssh node1

$ssh node2

$ssh node1-priv

$ssh node2-priv

#### 配置oracle用户环境，分别在node1和node2上执行下列命令

Node1

#su – oracle

$ssh-keygen -t rsa

$ssh-keygen -t dsa

$cd .ssh

$cat \*.pub>authorized\_keys

Node2

#su – oracle

$ssh-keygen -t rsa

$ssh-keygen -t dsa

$cd .ssh

$cat \*.pub>authorized\_keys

Node1

$scp authorized\_keys [oracle@node2:/home/oracle/.ssh/key\_dbs](mailto:oracle@node2:/home/oracle/.ssh/key_dbs)

Node2

$cat key\_dbs >>authorized\_keys

$scp authorized\_keys oracle@node1:/home/oracle/.ssh/

#### 测试信任关系

Node1,node2

$ssh node1

$ssh node2

$ssh node1-priv

$ssh node2-priv

### 2.2.4、设置时间同步ntpd

Node1、node2更新/etc/sysconfig/ntpd

#more /etc/sysconfig/ntpd

OPTIONS="-x -u ntp:ntp -p /var/run/ntpd.pid"

#service ntpd restart

#chkconfig –level 35 ntpd on

### 2.2.5、配置hangcheck-timer

此module用来重启服务器。

### 2.2.6、修改系统版本

Node1、node2

# more /etc/redhat-release

Red Hat Enterprise Linux Server release 4.3 (Tikanga)

### 2.2.7、检查必要软件

Node1、node2

yum -y install gcc compat-libstdc++-33 elfutils-libelf-devel glibc-devel glibc-headers gcc-c++ libaio-devel libstdc++-devel sysstat unixODBC unixODBC-devel pdksh

### 2.2.8、修改内核参数

Node1、node2

#cat /etc/sysctl.conf

net.ipv4.ip\_forward = 0

net.ipv4.conf.default.rp\_filter = 1

net.ipv4.conf.default.accept\_source\_route = 0

kernel.sysrq = 0

kernel.core\_uses\_pid = 1

net.ipv4.tcp\_syncookies = 1

kernel.msgmnb = 65536

kernel.msgmax = 65536

fs.aio-max-nr = 1048576

fs.file-max = 6815744

kernel.shmall = 4294967296

kernel.shmmax = 68719476736

kernel.shmmni = 4096

kernel.sem = 250 32000 100 128

net.ipv4.ip\_local\_port\_range = 9000 65500

net.core.rmem\_default = 262144

net.core.rmem\_max = 4194304

net.core.wmem\_default = 262144

net.core.wmem\_max = 1048576

#sysctl -p

### 2.2.9、修改shell限制

Node1、node2(在/etc/pam.d/login中添加下面一句话)

#cat /etc/security/limits.conf

grid soft nproc 2047

grid hard nproc 16384

grid soft nofile 1024

grid hard nofile 65536

oracle soft nproc 2047

oracle hard nproc 16384

oracle soft nofile 1024

oracle hard nofile 65536

#cat /etc/pam.d/login

…

session required pam\_limit.so

### 2.2.10、建立oracle安装目录

Node1、node2中执行以下脚本：

# cat adddir.sh

#!/bin/bash

mkdir -p /oracle/app/oraInventory

mkdir -p /oracle/app/grid

mkdir -p /oracle/app/11g/grid

chown -R grid.oinstall /oracle/

mkdir -p /oracle/app/oracle

chown -R oracle.oinstall /oracle/app/oracle

### 2.2.11、设定环境变量

#### 设定node1的grid用户环境

#su – grid

$vim .bash\_profile

if [ -f ~/.bashrc ]; then

. ~/.bashrc

Fi

PATH=$PATH:$HOME/bin

export PATH

TMP=/tmp; export TMP

TMPDIR=$TMP; export TMPDIR

ORACLE\_SID=+ASM1; export ORACLE\_SID

ORACLE\_BASE=/oracle/app/grid; export ORACLE\_BASE

ORACLE\_HOME=/oracle/app/11g/grid; export ORACLE\_HOME

NLS\_DATE\_FORMAT="yyyy-mm-dd HH24:MI:SS"; export NLS\_DATE\_FORMAT

THREADS\_FLAG=native; export THREADS\_FLAG

PATH=$ORACLE\_HOME/bin:$PATH; export PATH

if [ $USER = "oracle" ] || [ $USER = "grid" ]; then

if [ $USER = "/bin/ksh" ]; then

ulimit -p 16384

ulimit -n 65546

else

ulimit -u 16384 -n 65536

fi

umask 022

fi

$source .bash\_profile

#### 设定node2的grid用户环境

#su – grid

$vim .bash\_profile

if [ -f ~/.bashrc ]; then

. ~/.bashrc

Fi

PATH=$PATH:$HOME/bin

export PATH

TMP=/tmp; export TMP

TMPDIR=$TMP; export TMPDIR

ORACLE\_SID=+ASM2; export ORACLE\_SID

ORACLE\_BASE=/oracle/app/grid; export ORACLE\_BASE

ORACLE\_HOME=/oracle/app/11g/grid; export ORACLE\_HOME

NLS\_DATE\_FORMAT="yyyy-mm-dd HH24:MI:SS"; export NLS\_DATE\_FORMAT

THREADS\_FLAG=native; export THREADS\_FLAG

PATH=$ORACLE\_HOME/bin:$PATH; export PATH

if [ $USER = "oracle" ] || [ $USER = "grid" ]; then

if [ $USER = "/bin/ksh" ]; then

ulimit -p 16384

ulimit -n 65546

else

ulimit -u 16384 -n 65536

fi

umask 022

fi

$source .bash\_profile

#### 设定node1的oracle用户环境

#su - oracle

$vim .bash\_profile

if [ -f ~/.bashrc ]; then

. ~/.bashrc

fi

PATH=$PATH:$HOME/bin

export PATH

TMP=/tmp; export TMP

TMPDIR=$TMP; export TMPDIR

ORACLE\_BASE=/oracle/app/oracle; export ORACLE\_BASE

ORACLE\_HOME=/oracle/app/oracle/product/11.2.0/dbhome\_1; export ORACLE\_HOME

ORACLE\_SID=racdb1; export ORACLE\_SID

ORACLE\_TERM=xterm; export ORACLE\_TERM

PATH=/usr/sbin:$PATH; export PATH

PATH=$ORACLE\_HOME/bin:$PATH; export PATH

LD\_LIBRARY\_PATH=$ORACLE\_HOME/lib:/lib:/usr/lib; export LD\_LIBRARY\_PATH

CLASSPATH=$ORACLE\_HOME/JRE:$ORACLE\_HOME/jlib:$ORACLE\_HOME/rdbms/jlib; export CLASSPATH

NLS\_DATE\_FORMAT="yyyy-mm-dd HH24:MI:SS"; export NLS\_DATE\_FORMAT

NLS\_LANG=AMERICAN\_AMERICA.ZHS16GBK; export NLS\_LANG

if [ $USER = "oracle" ] || [ $USER = "grid" ]; then

if [ $SHELL = "/bin/ksh" ]; then

ulimit -p 16384

ulimit -n 65546

else

ulimit -u 16384 -n 65536

fi

umask 022

fi

$source .bash\_profile

#### 设定node2的oracle用户环境

#su - oracle

$vim .bash\_profile

if [ -f ~/.bashrc ]; then

. ~/.bashrc

fi

PATH=$PATH:$HOME/bin

export PATH

TMP=/tmp; export TMP

TMPDIR=$TMP; export TMPDIR

ORACLE\_BASE=/oracle/app/oracle; export ORACLE\_BASE

ORACLE\_HOME=/oracle/app/oracle/product/11.2.0/dbhome\_1; export ORACLE\_HOME

ORACLE\_SID=racdb2; export ORACLE\_SID

ORACLE\_TERM=xterm; export ORACLE\_TERM

PATH=/usr/sbin:$PATH; export PATH

PATH=$ORACLE\_HOME/bin:$PATH; export PATH

LD\_LIBRARY\_PATH=$ORACLE\_HOME/lib:/lib:/usr/lib; export LD\_LIBRARY\_PATH

CLASSPATH=$ORACLE\_HOME/JRE:$ORACLE\_HOME/jlib:$ORACLE\_HOME/rdbms/jlib; export CLASSPATH

NLS\_DATE\_FORMAT="yyyy-mm-dd HH24:MI:SS"; export NLS\_DATE\_FORMAT

NLS\_LANG=AMERICAN\_AMERICA.ZHS16GBK; export NLS\_LANG

if [ $USER = "oracle" ] || [ $USER = "grid" ]; then

if [ $SHELL = "/bin/ksh" ]; then

ulimit -p 16384

ulimit -n 65546

else

ulimit -u 16384 -n 65536

fi

umask 022

fi

$source .bash\_profile

### 2.2.12、配置RDM磁盘

Node1，node2（在node1中执行即可，node2可以看到分区）

#fdisk /dev/sdb

Disk /dev/sdb: 1073.7 GB, 1073741824000 bytes

255 heads, 63 sectors/track, 130541 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes

Device Boot Start End Blocks Id System

/dev/sdb1 1 2049 16458561 83 Linux

/dev/sdb2 2050 4098 16458592+ 83 Linux

/dev/sdb3 4099 24579 164513632+ 83 Linux

/dev/sdb4 24580 130541 851139765 5 Extended

/dev/sdb5 24580 45060 164513601 83 Linux

/dev/sdb6 45061 49157 32909121 83 Linux

/dev/sdb7 49158 53254 32909121 83 Linux

### 2.2.13、创建ASM磁盘

#### Node1上执行以下命令

# rpm -Uvh oracleasm-2.6.18-348.el5-2.0.5-1.el5.x86\_64.rpm oracleasmlib-2.0.4-1.el5.x86\_64.rpm oracleasm-support-2.1.7-1.el5.x86\_64.rpm

# /etc/init.d/oracleasm configure

Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM library

driver. The following questions will determine whether the driver is

loaded on boot and what permissions it will have. The current values

will be shown in brackets ('[]'). Hitting <ENTER> without typing an

answer will keep that current value. Ctrl-C will abort.

Default user to own the driver interface []: grid

Default group to own the driver interface []: asmadmin

Start Oracle ASM library driver on boot (y/n) [n]: y

Scan for Oracle ASM disks on boot (y/n) [y]: y

Writing Oracle ASM library driver configuration: done

Initializing the Oracle ASMLib driver: [ OK ]

Scanning the system for Oracle ASMLib disks: [ OK ]

# /etc/init.d/oracleasm createdisk CRS1 /dev/sdb1

Marking disk "CRS1" as an ASM disk: [ OK ]

# /etc/init.d/oracleasm createdisk CRS2 /dev/sdb2

Marking disk "CRS2" as an ASM disk: [ OK ]

# /etc/init.d/oracleasm createdisk DATA1 /dev/sdb3

Marking disk "DATA1" as an ASM disk: [ OK ]

# /etc/init.d/oracleasm createdisk DATA2 /dev/sdb5

Marking disk "DATA2" as an ASM disk: [ OK ]

# /etc/init.d/oracleasm createdisk REC1 /dev/sdb6

Marking disk "REC1" as an ASM disk: [ OK ]

# /etc/init.d/oracleasm createdisk REC2 /dev/sdb7

Marking disk "REC2" as an ASM disk: [ OK ]

#### 重新启动node1与node2

#### 在node2上执行以下命令

# rpm -Uvh oracleasm-2.6.18-348.el5-2.0.5-1.el5.x86\_64.rpm oracleasmlib-2.0.4-1.el5.x86\_64.rpm oracleasm-support-2.1.7-1.el5.x86\_64.rpm

/etc/init.d/oracleasm configure

Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM library

driver. The following questions will determine whether the driver is

loaded on boot and what permissions it will have. The current values

will be shown in brackets ('[]'). Hitting <ENTER> without typing an

answer will keep that current value. Ctrl-C will abort.

Default user to own the driver interface []: grid

Default group to own the driver interface []: asmadmin

Start Oracle ASM library driver on boot (y/n) [n]: y

Scan for Oracle ASM disks on boot (y/n) [y]: y

Writing Oracle ASM library driver configuration: done

Initializing the Oracle ASMLib driver: [ OK ]

Scanning the system for Oracle ASMLib disks: [ OK ]

[root@node2 ~]# /etc/init.d/oracleasm listdisks

CRS1

CRS2

DATA1

DATA2

REC1

REC2

## 2.3、安装阶段

### 2.3.1、安装grid

#### Node1上用grid用户执行下列命令

需在console上执行

$unzip linux.x64\_11gR2\_grid.zip

$cd grid

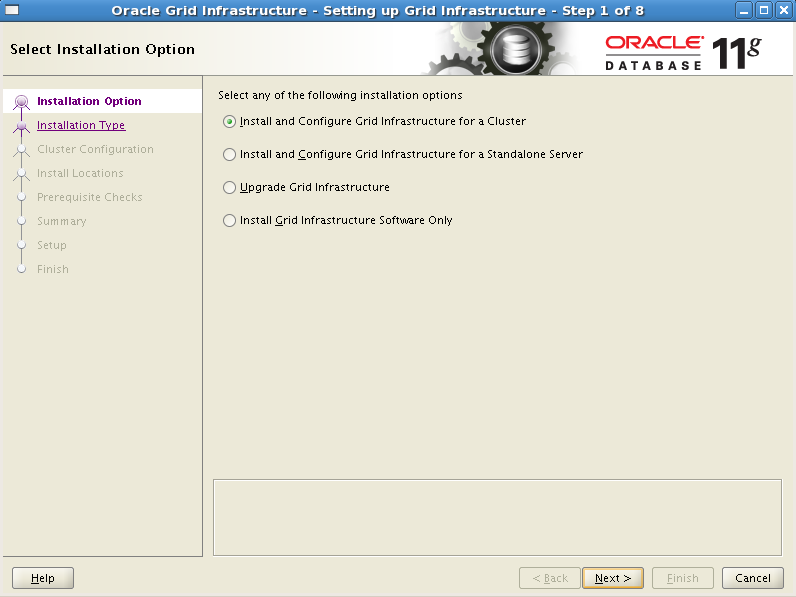
$./runcluvfy.sh stage -pre crsinst -n node1,node2 –fixup

…

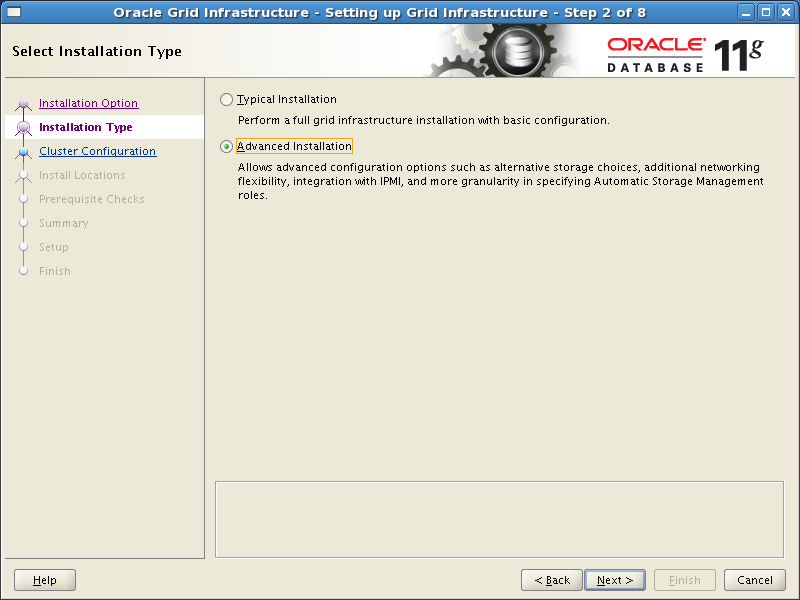
Pre-check for cluster services setup was successful.

$./runInstaller

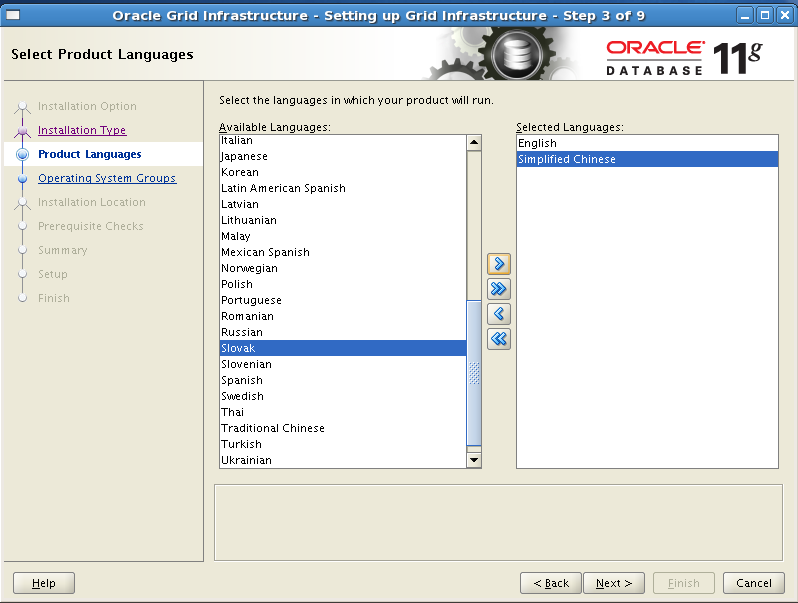
#### 选择安装配置集群的网格基础结构



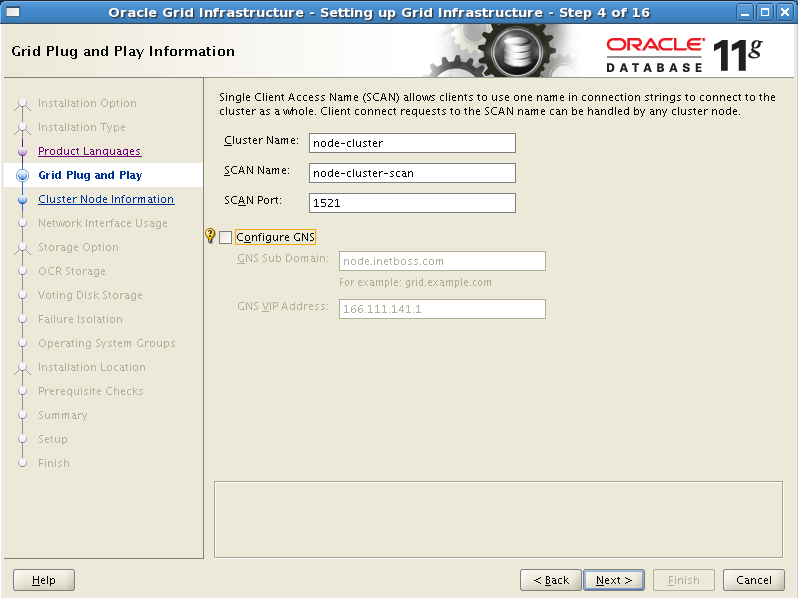
#### 高级安装



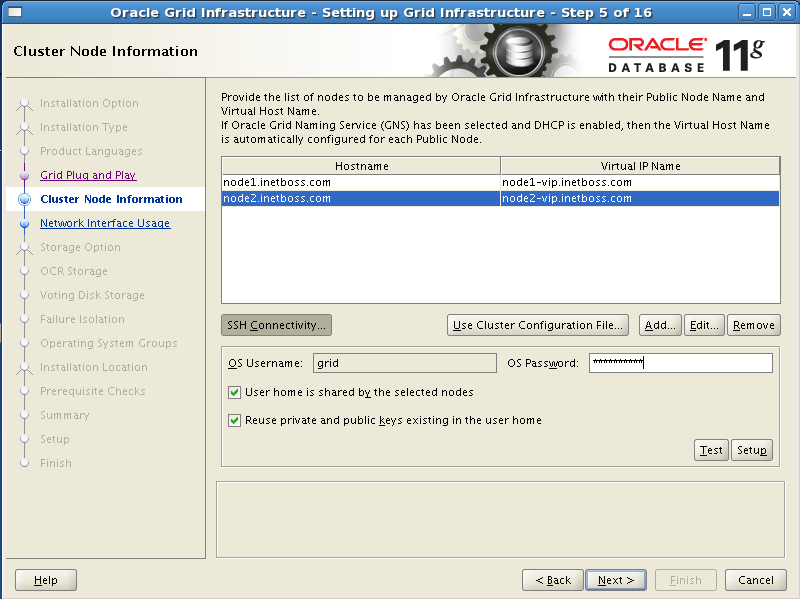
#### 选择语言



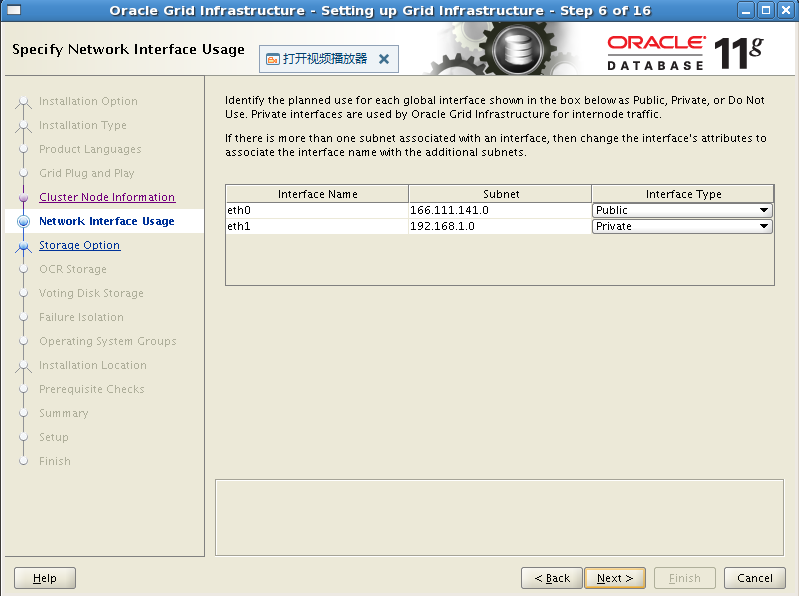
#### 取消配置GNS



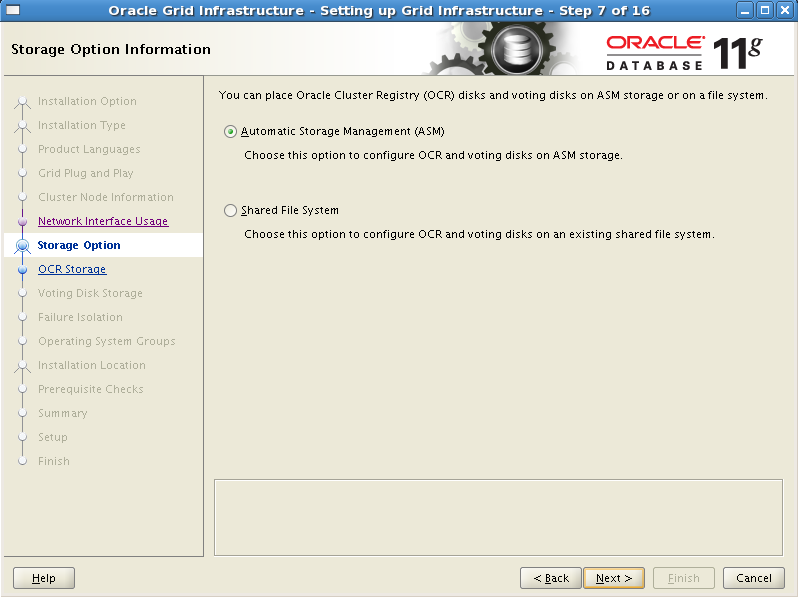
#### 添加节点



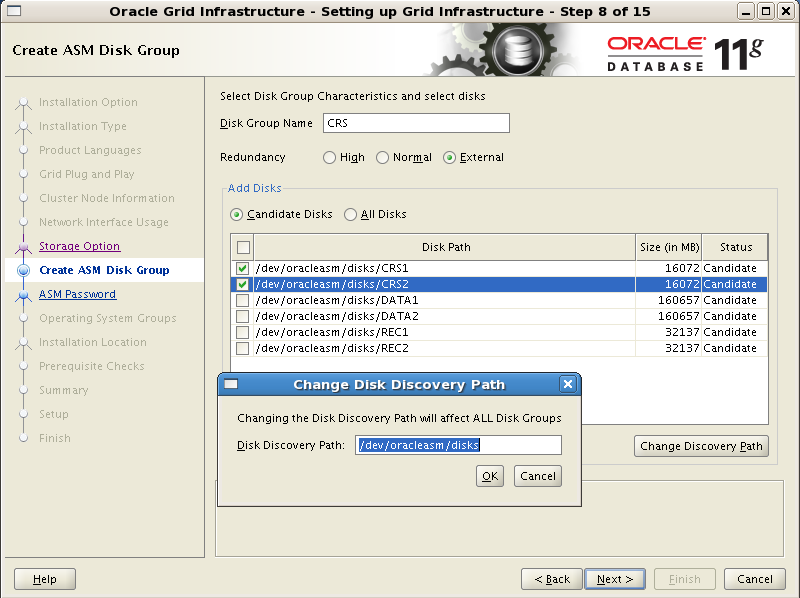
#### 网络信息



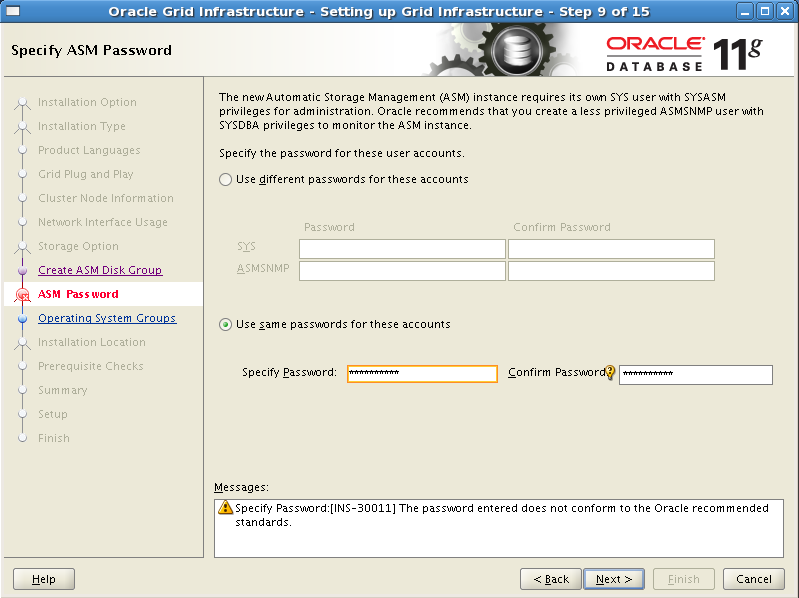
#### 存储选项：自动存储管理（ASM）



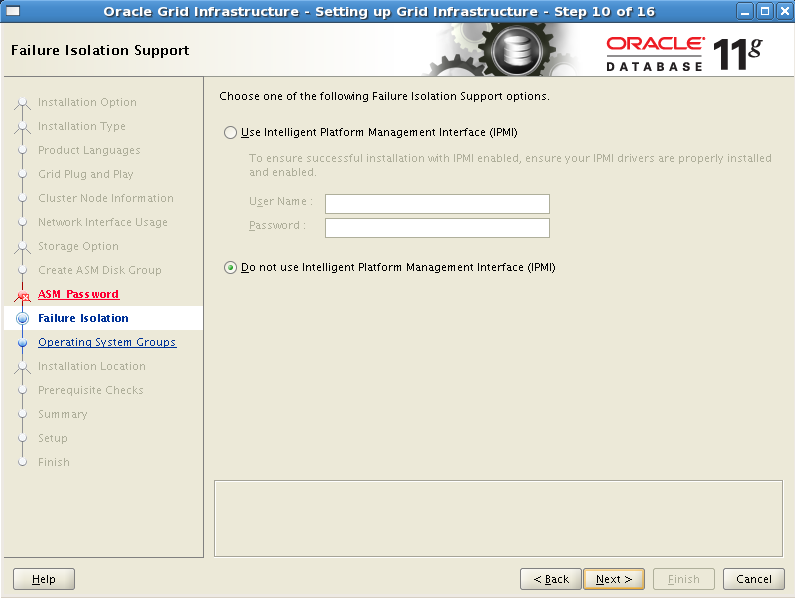
#### 创建磁盘组



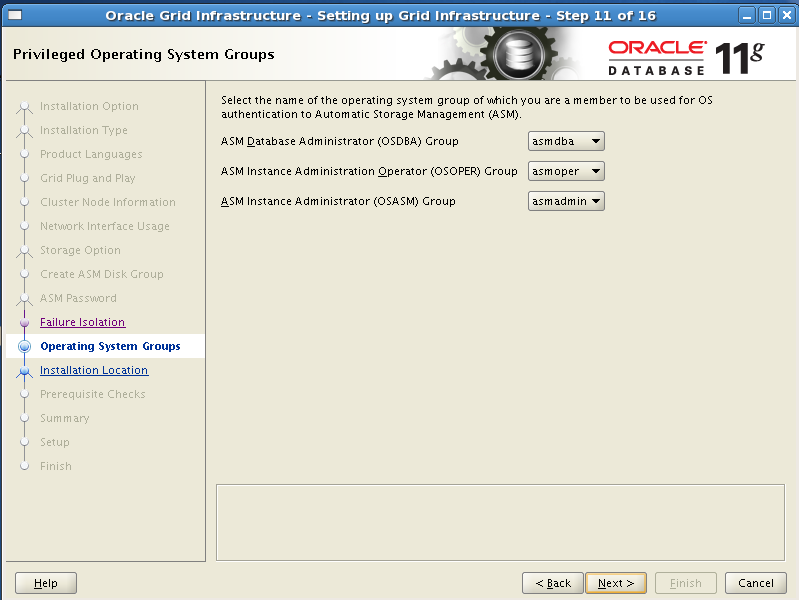
#### 选择口令方式：统一的口令



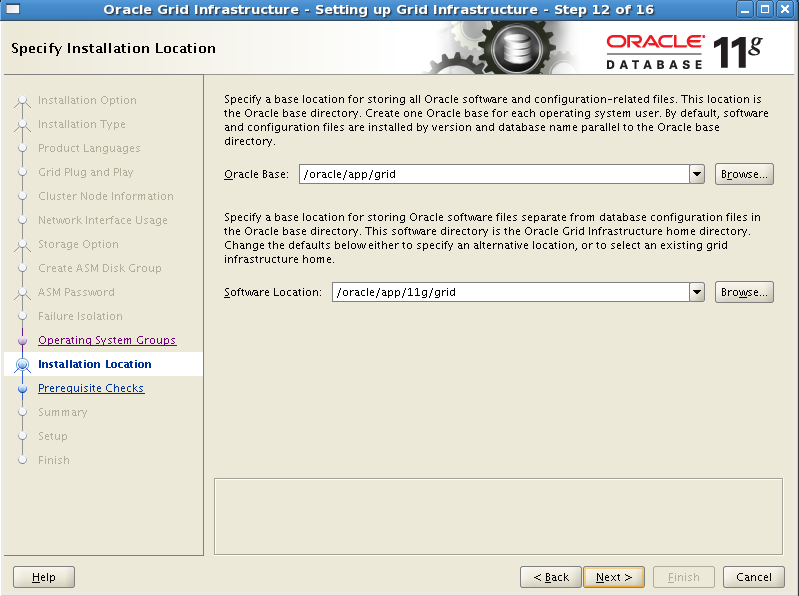
#### 智能平台管理界面：不选择



#### 操作系统用户组：默认



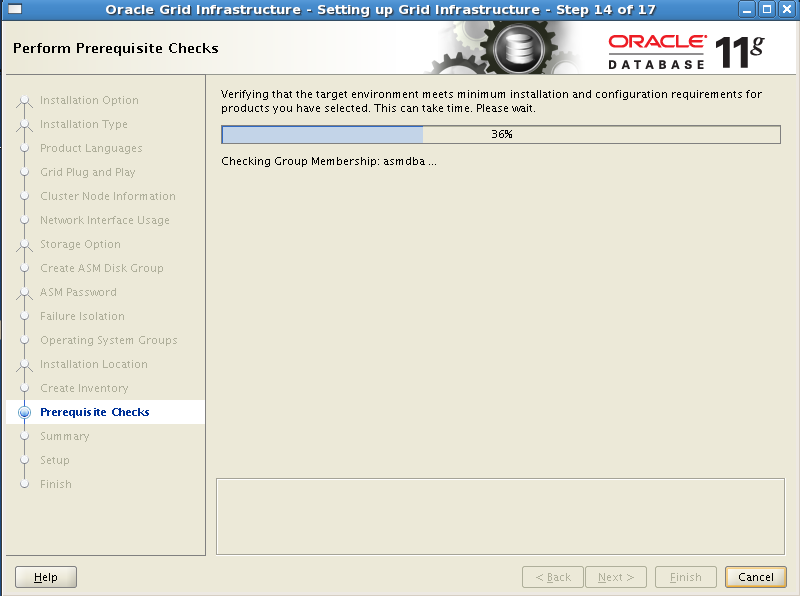
#### 设置安装路径：默认



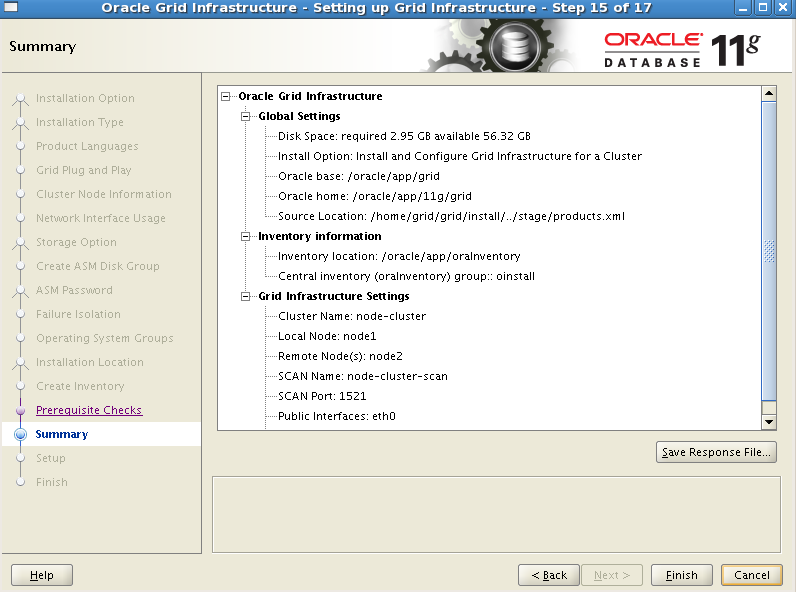
#### 创建产品清单：默认



#### 安装就绪检查：通过

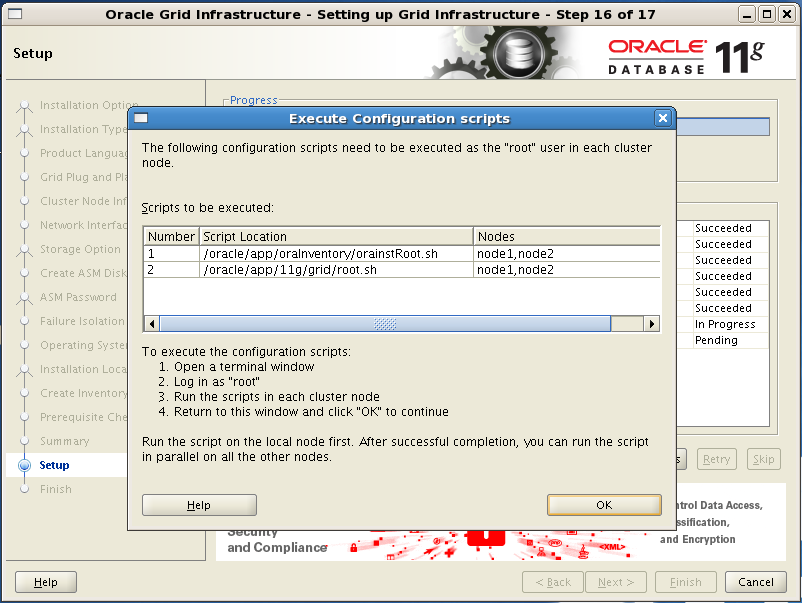


#### 安装前确认



#### 安装完成执行脚本

需特别注意执行顺序，node1执行完成后再执行node2。



脚本执行成功将返回

#/oracle/app/11g/grid/root.sh

…

'UpdateNodeList' was successful.

执行成功后，检查状态

[grid@node1 grid]$ crsctl check crs

CRS-4638: Oracle High Availability Services is online

CRS-4537: Cluster Ready Services is online

CRS-4529: Cluster Synchronization Services is online

CRS-4533: Event Manager is online

[grid@node1 grid]$ crsctl stat res -t

--------------------------------------------------------------------------------

NAME TARGET STATE SERVER STATE\_DETAILS

--------------------------------------------------------------------------------

Local Resources

--------------------------------------------------------------------------------

ora.CRS.dg

ONLINE ONLINE node1

ONLINE ONLINE node2

ora.asm

ONLINE ONLINE node1 Started

ONLINE ONLINE node2 Started

ora.eons

ONLINE ONLINE node1

ONLINE ONLINE node2

ora.gsd

OFFLINE OFFLINE node1

OFFLINE OFFLINE node2

ora.net1.network

ONLINE ONLINE node1

ONLINE ONLINE node2

ora.ons

ONLINE ONLINE node1

ONLINE ONLINE node2

--------------------------------------------------------------------------------

Cluster Resources

--------------------------------------------------------------------------------

ora.LISTENER\_SCAN1.lsnr

1 ONLINE ONLINE node1

ora.node1.vip

1 ONLINE ONLINE node1

ora.node2.vip

1 ONLINE ONLINE node2

ora.oc4j

1 OFFLINE OFFLINE

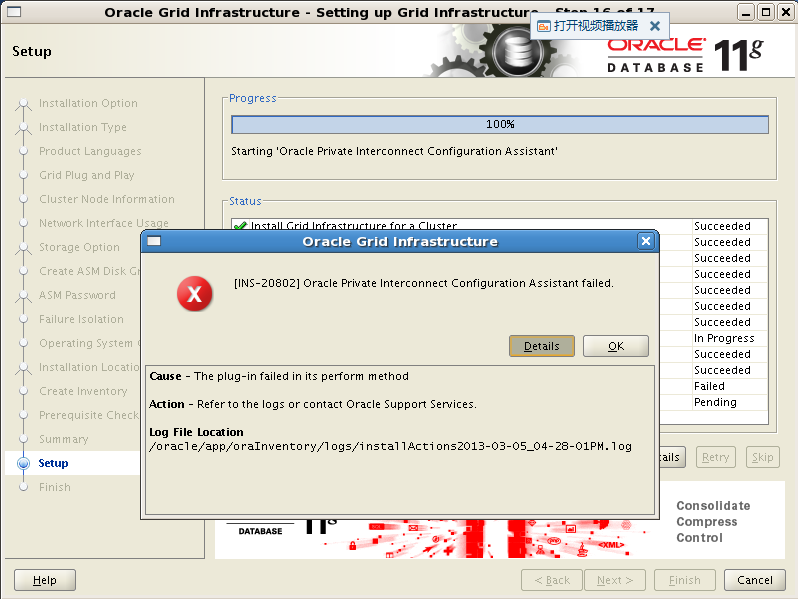
ora.scan1.vip

1 ONLINE ONLINE node1

其中gsd和oc4j服务OFFLINE是11gR2默认选项。

参照<http://space.itpub.net/?uid-7199859-action-viewspace-itemid-628050>可以打开gsd和oc4j服务

#### 完成安装



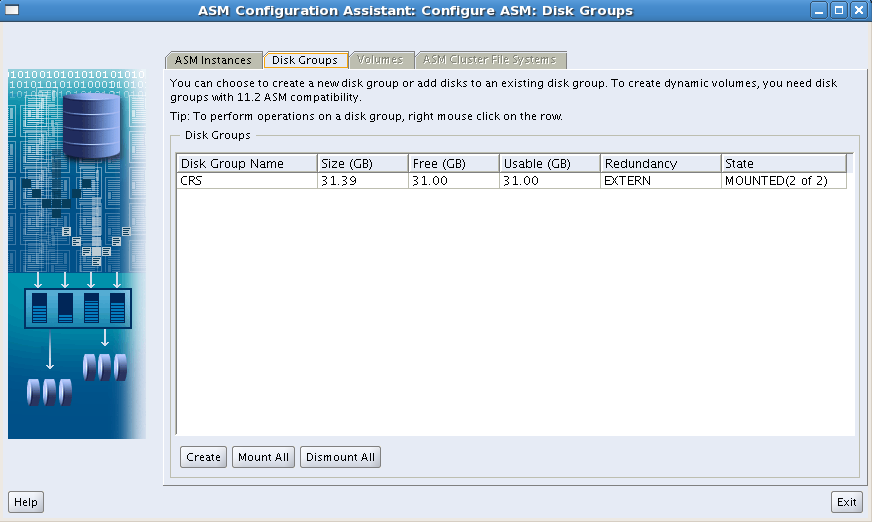
这个错误忽略即可。

### 2.3.2、为数据和快速恢复区创建ASM磁盘组

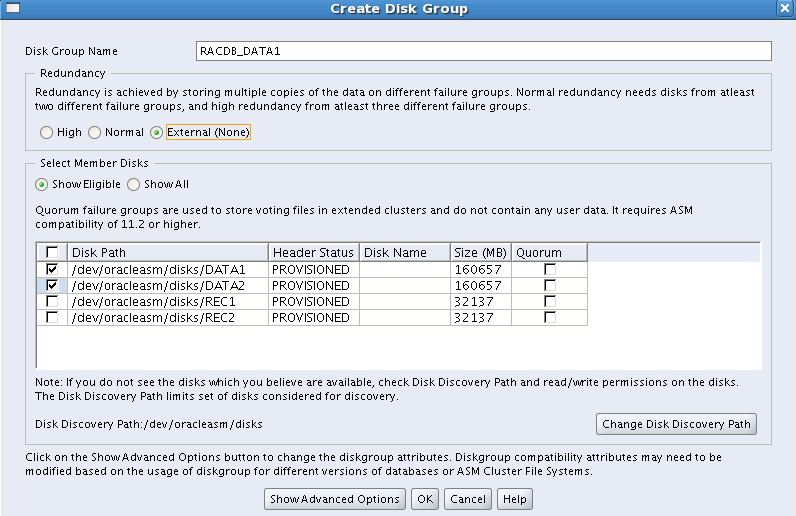
#### 在node1上由用户grid执行下列操作开始创建

[grid@node1 grid]$ asmca &

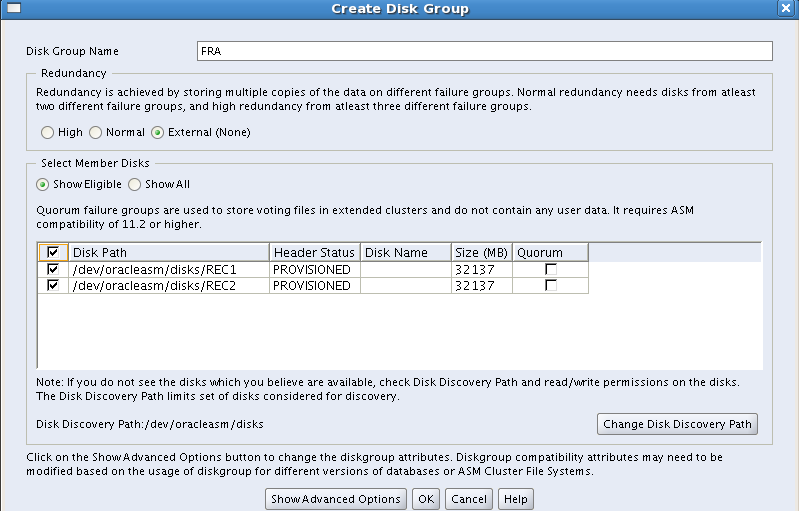
#### 选择磁盘组，点击创建



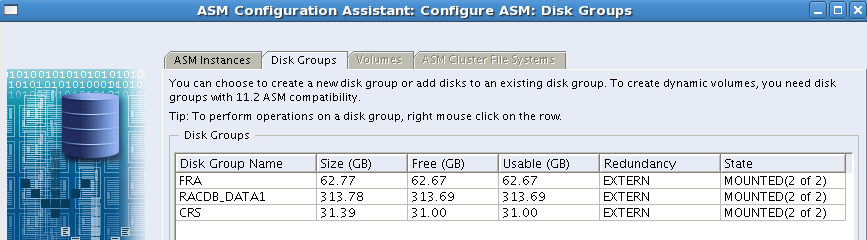
#### 创建RACDB\_DATA1磁盘组



#### 创建FRA磁盘组



#### 创建完成



### 2.3.3、安装oracle数据库

#### Node1上用oracle用户执行下列命令

需在console上执行

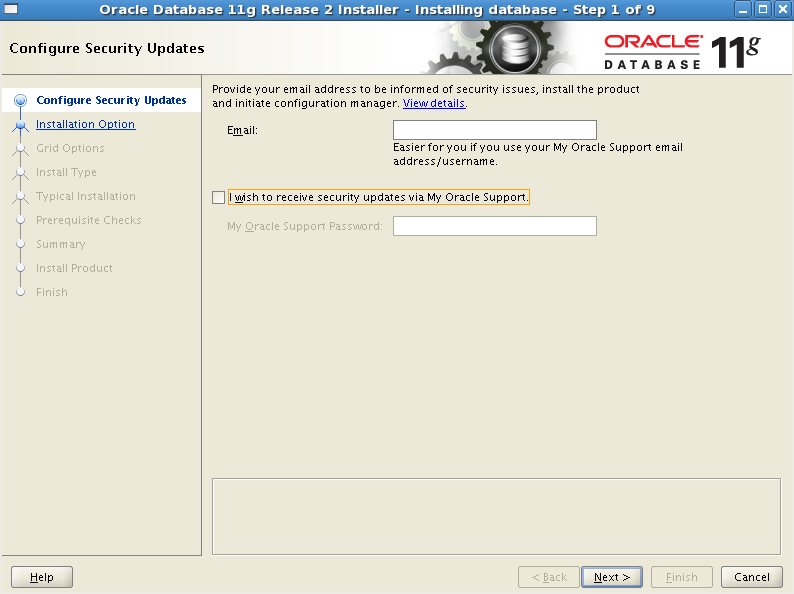
$unzip linux.x64\_11gR2\_database\_1of2.zip

$unzip linux.x64\_11gR2\_database\_2of2.zip

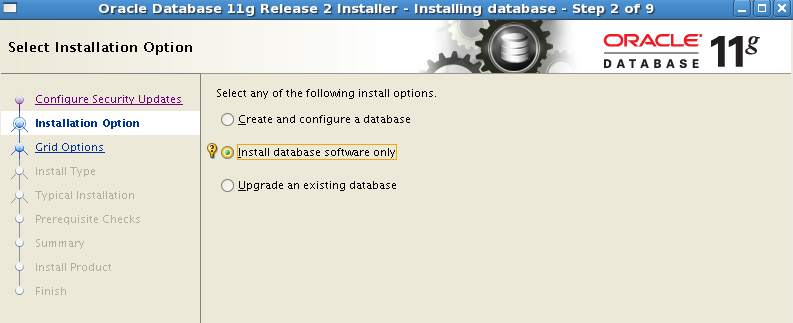
$cd database

$./runInstaller

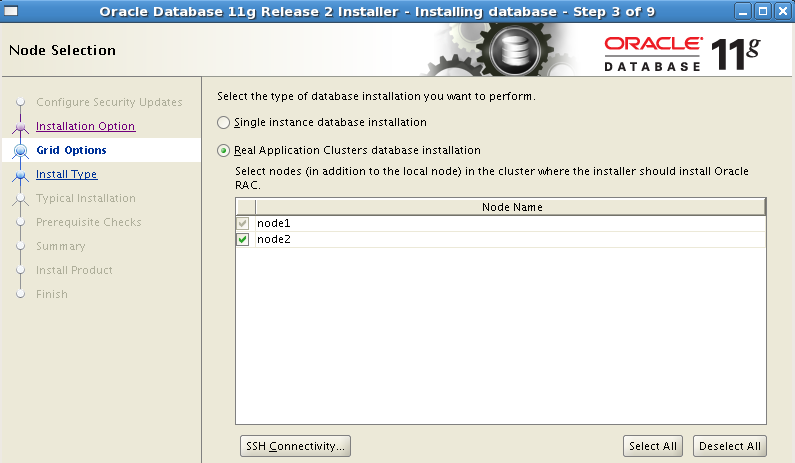
#### 安全更新



#### 安装选项



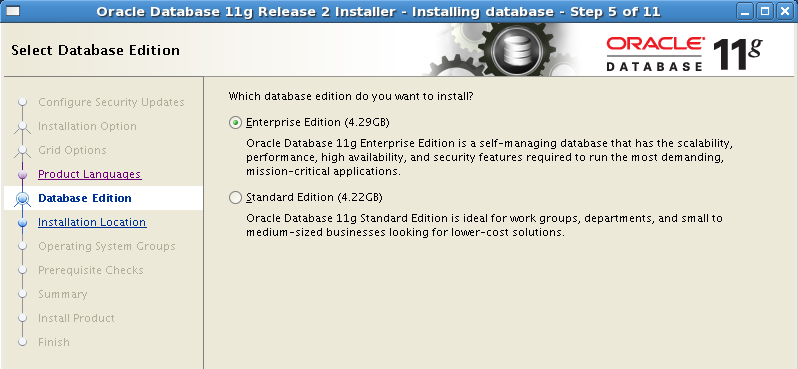
#### 选择安装节点



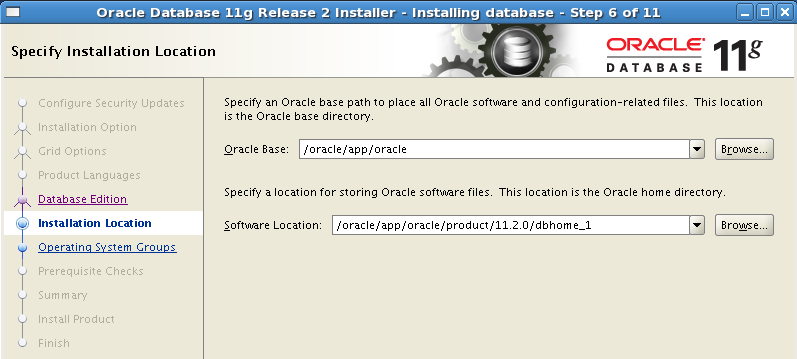
#### 选择安装语言



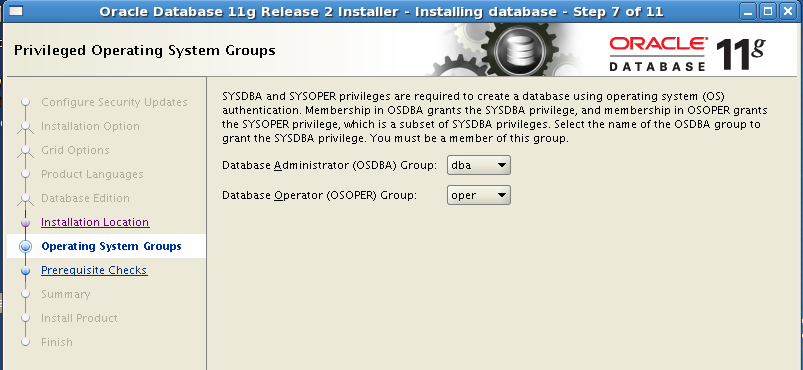
#### 选择企业版安装



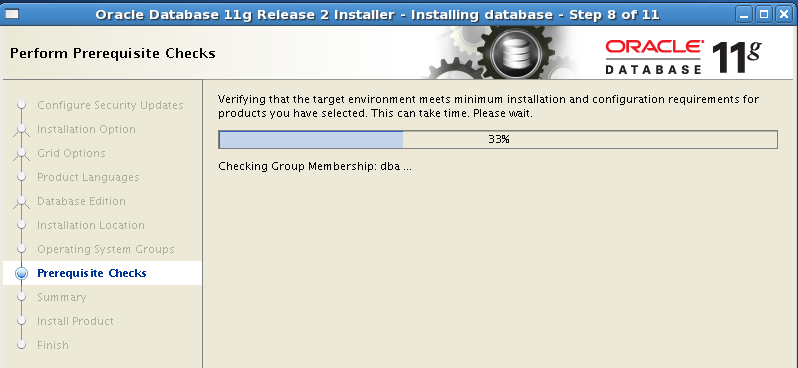
#### 安装路径



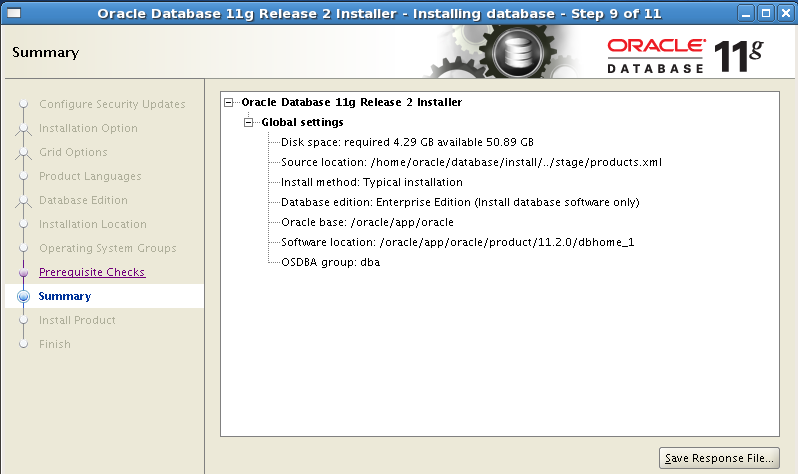
#### 系统用户组



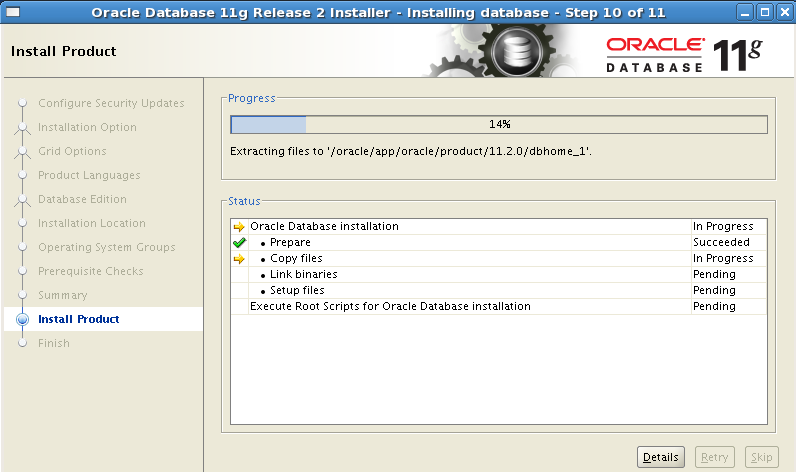
#### 安装前检查



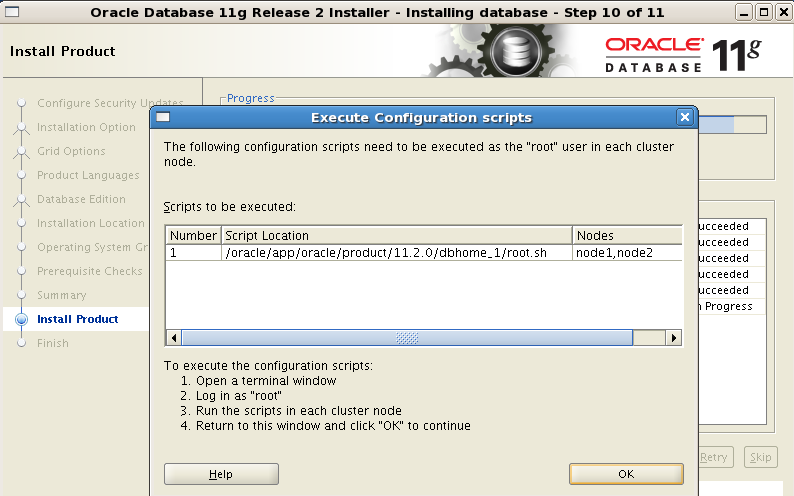
#### 安装确认



#### 开始安装过程



#### 执行安装后脚本



安装结束

# 常用命令：

crsctl status resource -t查看集群系统资源组状态

Oracle RAC：Oracle Real Application Server是Oracle的并行集群。那些位于不同[服务器](http://q.sohu.com/forum/5/topic/5632352)系统的Oracle实例可以同时访问同一个Oracle数据库，节点之间通过私有[网络](http://q.sohu.com/forum/5/topic/5632352)进行[通信](http://q.sohu.com/forum/5/topic/5632352)；所有的控制文件、联机日志和数据文件存放在共享的设备上，能够被集群中的所有节点同时读写。

RDM: Raw Device Mapping ,原始设备映射。VMware中用来表示直接挂载到虚拟机中（不通过虚拟主机Host）的磁盘。

ASM：