

三思笔记系列文章 之

手把手教你在 Linux 环境部署 Oracle 11gR2 RAC

2013-01-20 君三思

<http://www.51enet.com/>

1、背景资料扫盲.....	2
1.1 理解 RAC 环境中的兼容性.....	2
1.2 RAC 安装和建库.....	2
1.3 Grid 扩展.....	3
1.4 关于 Grid 基础框架和 RAC.....	4
1.5 关于 Oracle Automatic Storage Management(ASM).....	4
1.6 关于 Oracle Real Application Clusters.....	6
1.7 关于 Oracle RAC One Node.....	7
1.8 关于在不同操作系统安装 RAC.....	7
1.9 安装配置与管理 RAC 的工具.....	7
2、安装前的准备工作.....	8
2.1 软硬件环境准备.....	8
2.2 初始化第一台节点.....	9
2.2.1 设置操作系统环境变量.....	9
2.2.2 配置网卡地址.....	10
2.2.3 配置/etc/hosts 文件.....	11
2.2.4 配置软件目录.....	11
2.2.5 配置用户.....	12
2.3 初始化第二台节点.....	12
3、配置共享磁盘.....	12
3.1 配置安全访问.....	12
3.2 创建共享磁盘.....	13
3.3 配置共享磁盘.....	13
3.4 配置 ASM 磁盘.....	14
4、安装 GRID 基础框架.....	16
5、安装数据库软件.....	29
6、创建数据库.....	36

本文基于 VirtualBox 虚拟化软件维护虚拟机，Oracle Enterprise Linux 6.0+Oracle11gR2 部署 RAC 集群数据库环境。本书出现的所有软件均可在互联网中免费下载获得。

1、背景资料扫盲

安装 Oracle Clusterware 和 Oracle Database，使用 DBCA 创建数据库。做为数据库管理员，安装后需要多三个维度管理 RAC 环境：

- 实例管理；
- 数据库管理；
- 集群管理；

我们下面先简要介绍 RAC 安装的步骤。

更详细的内容可以参考官方文档，与 RAC 特性相关的文档主要有四个：

- [Oracle Real Application Clusters Administration and Deployment Guide](#)
- [Oracle Clusterware Administration and Deployment Guide](#)
- [Oracle Real Application Clusters Installation Guide for Linux and UNIX \(or other operating system\)](#)
- [Oracle Grid Infrastructure Installation Guide for Linux \(or other operating system\)](#)

1.1 理解 RAC 环境中的兼容性

在同一个集群中安装不同的数据库版本，必须安装集群件。举例来说，要在同一个集群中同时安装 9i 和 10g：

- 对于运行 9i 的 RAC 节点，必须要安装 9i 集群：
 - 对于 UNIX 系统，集群件可以是 HACMP/Service Guard/Sun Clusterware/Veritas SF 等；
 - 对于 Windows 或 Linux，集群件可以是 Oracle Cluster Manager；
- 对于 10g 或以上版本的 RAC，必须安装 Oracle Clusterware。对于 11gR2 版本，必须安装 Grid 组件，具体可以参考：
http://docs.oracle.com/cd/E11882_01/install.112/e22489/toc.htm。
- 如果运行 10g 或 11gRac 在同一套集群，则必须运行 Clusterware11g。

1.2 RAC 安装和建库

安装了 Clusterware 及其部件，运行 Universal Installer 安装 Oracle 数据库软件。在安装过程中，可以选择创建数据库，它会自动调用 DBCA 创建数据库。

提示：

如果选择在安装软件时建库，那么必须启动默认监听，否则 DBCA 会报错，并提示运行 NETCA 以创建默认监听。

Oracle RAC 软件是做为数据库安装包的一部分存在的。默认情况下，如果它识别到正在集群环境中安装，数据库软件安装进程也会安装 RAC 选项。安装程序将 RAC 安装到结构类似 ORACLE_HOME 的目录下，以区分系统上运行的不同软件。因为安装程序是集群件，它会将 RAC 软件装到集群定义的所有节点中。

ORACLE 建议用户选择 ASM 做为存储管理。ASM 能够自动所有数据库文件。如果计划基于 ORACLE 标准版创建 RAC 数据库，那么就必须使用 ASM 保存所有的数据库文件。

提示：必须在创建 RAC 数据库前单独配置 ASM。

默认情况下，ORACLE 数据库为 RAC 的安装创建一个服务，这个服务用于数据库。默认数据库服务是按照 db_name.db_domain 来识别。在 RAC 环境中该默认服务对所有实例有效，除非数据库是处于 restricted 模式。

提示：

集群件安装完成后就要避免修改主机名，包括增加或者删除域名中的域标识。修改了主机名的节点，必须先从集群中删除，而后再添加进来才行。

1.3 Grid 扩展

可以在 Grid 环境中通过复制一份镜像到其它节点的方式扩展 ORACLE RAC。克隆到的目标节点应该拥有类似的软硬件环境。通过克隆的方式是最简便的扩展 RAC 环境的方式。

ORACLE 提供了下列方式扩展 RAC 环境：

- 通过克隆脚本，详见：
http://docs.oracle.com/cd/E11882_01/rac.112/e16795/clonerac.htm
- 通过 Oracle EM 克隆；
- 通过 addNoe.sh 脚本及安装管理工具克隆；

提示：

Oracle 克隆不是要替换 EM 中的克隆，而只是提供另一种备用方案。在 EM 克隆过程中，provisioning 进程包括一系列步骤，比如捕获的目录，部署的目录，以及其它各种参数。

对于新的安装，或者只安装一个 RAC 数据库，用户需要用传统自动化交互安装方式，比如 Oracle Universal Installer，或者是 EM 提供的 Provisioning Pack 特性。如果目地是要添加或删除当前集群中的节点，那么可以参考下列步骤：[Chapter 10, "Adding and Deleting Oracle RAC from Nodes on Linux and UNIX Systems"](#)。

克隆过程假定用户已经成功安装了集群件，并且 RAC 环境中至少已经有了一个节点，并且，所有 root 要执行的脚本，在要扩展的节点上已经执行过了。

站在列高层次，Oracle 克隆调用下列的主要任务：

1. 克隆集群件；
2. 克隆 RAC 软件的 HOME 目录；
3. 如果当前还没有数据库，那么执行 DBCA 创建一个；
4. 按照克隆的步骤完成 RAC 环境的扩展；

1.4 关于 Grid 基础框架和 RAC

从 11gR2 版本开始, Clusterware 和 ASM 将安装到相同目录, 现在被称为 Grid 主目录。Grid 框架是一系列复合产品的总和, Clusterware 和 ASM 仍然是独立的产品。

Clusterware 允许将一堆服务器(hosts 或 nodes)看成一个服务器来操作, 这也被称为一个集群(a cluster)。尽管各个服务器物理上仍然是独立的, 每个服务器都拥有额外的进程与其它服务器通讯。这样每个服务器都做为一个独立的服务响应应用和用户。Oracle Clusterware 提供框架易于运行 RAC。多个服务器复合相比单台服务器, 能够提供更高的可用性, 吞吐和伸缩性。

注意:

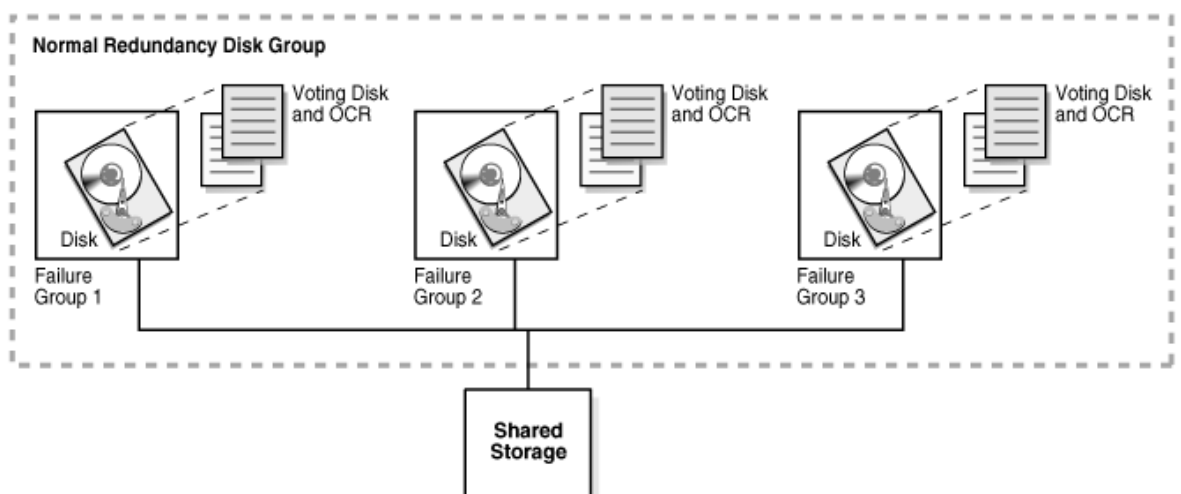
默认情况下, 命名用户可以创建 server pool。要限制操作系统用户拥有这个权限, Oracle 强烈建议用户增加指定用户到 CRS 管理员列表。详见集群件管理与部署文件中的内容。

1.5 关于 Oracle Automatic Storage Management(ASM)

Oracle ASM 是一套完整的高性能的卷管理文件系统。在 11gR2 版本中, ASM 能够支持存储集群件的 OCR 和 voting disk 文件。也被称为 Oracle ACFS(Oracle Automatic Storage Management Cluster File System)

ASM 能够将存储系统中的磁盘分组管理。用户只要管理少量的磁盘组, ASM 能够自动分配磁盘组存储数据文件。OCR 和仲裁文件也能被放在 ASM 磁盘组中。要使用 RAC 特性, 每个实例都必须拥有访问数据文件和恢复文件的权限, ASM 是实现这类需求的最简便方式。

在下图中, ASM 中的普通冗余磁盘组包括三个 failure 组(也可以理解为三个磁盘), 用于保存仲裁磁盘和 OCR 文件。因为仲裁盘中记录哪个节点是集群中的当前成员, 对于双节点的集群, 至少要有三个仲裁盘。如果 ASM 磁盘组中不包含仲裁盘和 OCR 的话, 那么普通冗余磁盘组中可以只建两个 failure 组。



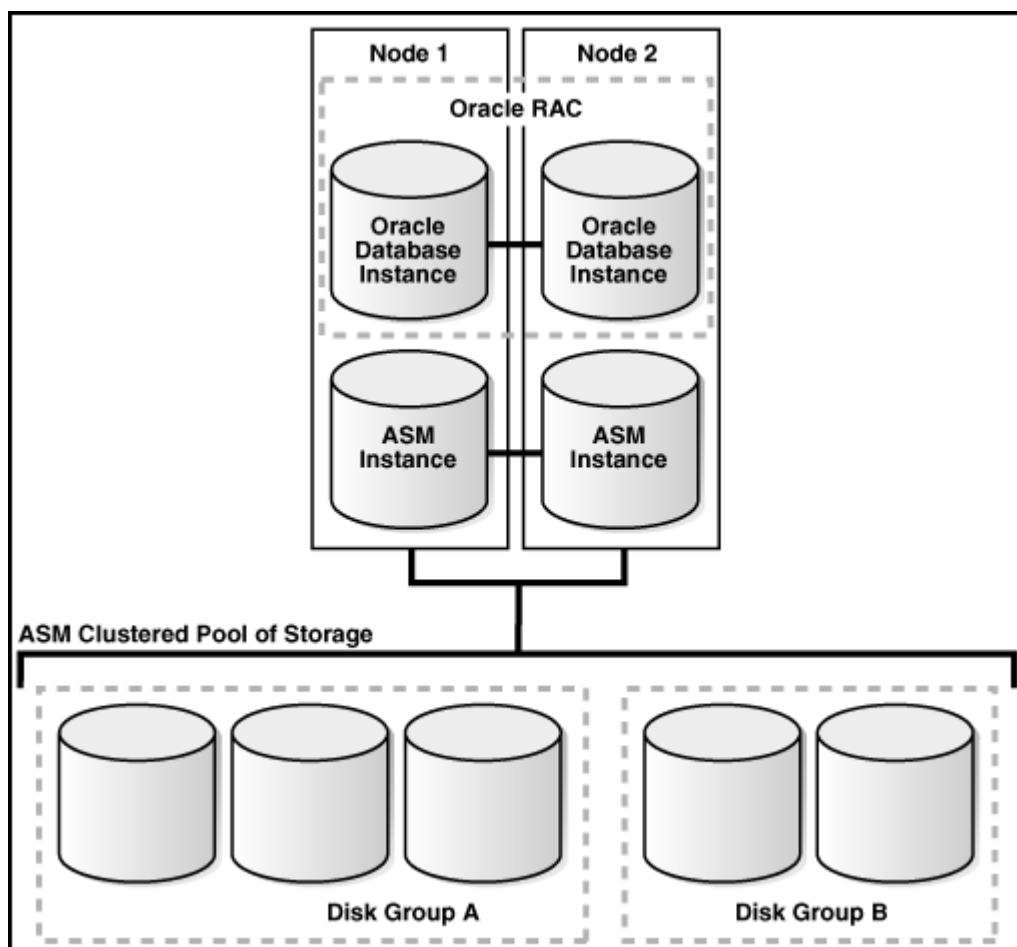
ASM 提供下列特性:

- 条带(Striping): ASM 能够将数据平均分布到各个磁盘,以优化性能并提高利用率;
- 镜像(Mirroring): ASM 通过镜像文件的方式,提升数据可用性。ASM 的镜像是在文件级,不像操作系统的镜像是在磁盘级。镜像也表示保持冗余备份,或镜像复制,来避免磁盘故障导致的文件损坏。每个文件的镜像复制肯定跟其原始文件不在同一块磁盘。这样就算某块磁盘故障,ASM 照样能够从镜像盘中继续读取文件;
- 联机存储重置和动态平衡:使用 ASM 时,甚至可以在数据库操作过程中,仍然增加或删除磁盘。如果增加磁盘到磁盘组,ASM 自动重新分配数据在磁盘组中的存储位置,利用上新增加的磁盘。这种重新给数据分配存储位置的操作过程也被称为 rebalancing。这是项后台任务,对前端完全透明,而且基本不影响性能。
- 管理文件创建和删除:ASM 通过允许数据库管理存储在 ASM 中磁盘组的方式进一步减少管理任务。ASM 会在文件创建时自动命名,并且能够在数据库删除文件时,自动将该文件从存储中删除。

Oracle Automatic Storage Management Cluster File System(简称 Oracle ACFS),是种新的跨平台、易伸缩的文件系统、扩展自 ASM 存储管理技术以支持自定义文件管理。ACFS 即能够支持数据库和应用的各项文件,包括 trace files, alert logs, application reports, BFILEs 以及配置文件。ACFS 提供了动态文件大小调整、条带和镜像。

ASM 可视做一种特殊类型的 Oracle 实例,拥有独立的 SGA 和后台进程。ASM 实例与 Clusterware 和 database 紧密结合。每个使用 ASM 做为存储的数据库服务器上都要有一个 ASM 实例。在 RAC 环境中,每个节点都有一个 ASM 实例,不过也可以多个数据库实例使用一个 ASM 实例。

下图展示双节点 RAC 的存储结构:



Oracle 建议用户使用 ASM 而非操作系统文件系统。不过 ORACLE 数据库能够同时使用 ASM 和非 ASM 的文件。同样也可以在 ACFS 中存储非 ORACLE 的文件。

1.6 关于 Oracle Real Application Clusters

RAC 扩展 ORACLE 数据库，使用户可以更新、存储并且高效地，同时从不同服务器上运行的，多个数据库实例中操作数据。RAC 能够统一管理多个服务器和实例。数据文件必须保存在共享存储上。

传统单实例 Oracle 数据库的数据文件和实例之间是一一对应的关系，不过对于 RAC 数据库，数据库和实例间是一对多的关系，多个实例同时访问同一份数据库文件。

RAC 数据库中的每个实例都拥有独立的内存结构和后台进程。RAC 使用 Cache Fusion 技术同步实例的 buffer cache 中保存的缓存数据。Cache Fusion 将当前数据块(内存中的)在不同数据库实例间移动，而不是将数据写回执行，再由其它实例从磁盘中读取。当某个实例上的缓存块需要被其它实例读取时，Cache Fusion 使用 interconnect 直接将数据传输到目标实例上，使得数据库操作数据时就像是在同一个缓存池中一样。

RAC 同时也是 grid 计算架构中的重要组成部分。多个实例访问同一份数据文件，也能够防止单点故障，如果集群中的某个节点故障，其它存活的节点仍能继续提供数据库服务。

RAC 能够支持各项主流商业应用，包括 OLAP、DSS，以及流行的软件包产品如 SAP、PeopleSoft、Siebel 以及 Oracle E*Business Suite，当然还有客户的应用。能够在单实例中运

行的应用到 RAC 中不需要做修改就可以直接部署。

1.7 关于 Oracle RAC One Node

单节点 RAC 是 11gR2 版本中的新特性，说的是 RAC 数据库在集群中只运行于一个节点上。这个特性允许用户以最小成本合并多个数据库到单个集群，保持其计划内或计划外的停机时间。合并后的数据库提升了高可用性和故障保护，联机的滚动 patch，以及 OS/集群滚动升级。

相比单实例数据库的冷故障切换，单节点 RAC 提供了更高的可用性，怎么理解呢，说的是 ORACLE 提供了一项称为 online database relocation 的技术，能够智能迁移数据库实例，连接到其它集群中的其它节点，以提供高可用和负载均衡。使用 SRVCTL 执行联机数据库迁移。

如果应用部署于单节点 RAC，之后发现单节点已难以承载，那么可以轻易的联机方式升级为 RAC 集群。如果单节点 RAC 超负载，那么可以迁移数据库实例到集群中的其它节点上，而不需要应用停机。

1.8 关于在不同操作系统安装 RAC

如果计划安装和配置 RAC 环境在不同操作系统，这个场景比较特殊，这里先不体现。

1.9 安装配置与管理 RAC 的工具

本小节介绍安装、配置、管理 ORACLE 集群/ASM/RAC 的工具。

- Oracle Universal Installer(OUI): 图形化的安装界面;
- Cluster Verification Utility(CVU): 命令行工具，可用于验证集群范围和 RAC 组织，比如共享存储设备，网络配置，系统需求以及 ORACLE 集群，同样也包括操作系统用户组 and 用户。可以使用 CVU 工具进行安装前/后的集群环境检查。OUI 会在集群安装后运行 CVU 验证环境。
- Oracle Enterprise Manager: B/S 结构管理实例或 RAC 环境，拥有数据库控制和 GUID 控制的图形用户界面。
- SQL*Plus: 命令行界面工具，可用于执行数据库管理操作;
- Server Control(SRVCTL): 命令行界面工具，可用于管理 OCR 中定义的资源。这些资源包括节点应用(nodeapps)，由 Clusterware，ONS(Oracle Notification Service，GSD(Global Services Daemon)，以及 VIP(Virtual IP)组成。其它可被 SRVCTL 管理的资源还包括数据库、实例、监听、服务以及应用程序。使用 SRVCTL 可以启动或停止 nodeapps/databases/instances/listeners/services，增加或移动 instances/services，增加服务，管理配置等;
- Cluster Ready Services Control(CRSCTL): 命令行工具，用于管理 Clusterware 守护

进程。这些进程包括：Cluster Synchronization Services (CSS), Cluster-Ready Services (CRS), and Event Manager (EVM)。可以使用 CRSCCTL 启动或停止 Oracle Clusterware 或检查 Clusterware 的当前状态；

- Database Configuration Assistant(DBCA): 一项专门应用, 可用于创建和配置数据库。
- Oracle Automatic Storage Management Configuration Assistant(ASMCA): 支持安装和配置 ASM 实例、磁盘组、卷和 ACFS, 有 GUI 和非 GUI 两种界面；
- Oracle Automatic Storage Management Command Line utility(ASMCMD): 命令行界面工具, 用于管理 ASM 实例、ASM 磁盘组、磁盘组的文件读写权限、ASM 磁盘组中的文件和目录、磁盘组模板以及 ASM 卷；
- Listener Control(LSNRCTL): 监听控制应用, 命令行界面的工具, 用于管理监听。

如果安装了 Grid 框架, 当管理集群、数据库、实例、ASM、监听时, 要使用正确的主目录下的应用。可以设置 ORACLE_HOME 环境变量到适当目录, 例如:

- 如果使用 ASMCMD、srvctl、sqlplus、lsnrctl 管理 ASM 或监听, 那么其二进制程序要指向 GRID 的主目录, 而不是数据库的主目录；
- 如果使用 srvctl、sqlplus 或 lsnrctl 管理数据库实例或监听, 那么其二进制程序要指向数据库的主目录；

2、安装前的准备工作

接下来就开始执行安装任务。

2.1 软硬件环境准备

硬件环境

虚拟机环境: 两核 4G 内存虚拟机*2

hostname: racnode1,racnode2

软件环境

操作系统: Oracle Enterprise Linux6.0

GRID: linux.x64_11gR2_grid.zip

Database : Oracle database
11gR2(linux.x64_11gR2_database_1of2.zip,linux.x64_11gR2_database_2of2.zip)

网络设定

IP 设定如下:

#PUBLIC IP:

192.168.10.201 racnode1

192.168.10.202 racnode2

#PRIVATE IP:


```
10.0.0.201   racpriv1
10.0.0.202   racpriv2
#VIP IP:
192.168.10.211   racvip1
192.168.10.212   racvip2
#SCAN IP:
192.168.10.206   racscan
```

扫盲：何为 SCAN(Single Client Access Name)

定义的一个网络名及 IP 地址，这样所有客户端均能够通过它访问 RAC 数据库。通过 SCAN，当集群环境发生变化时，就不需要再修改客户端的配置。SCAN 同样允许客户端通过简易连接串的方式，也能够利用到 RAC 的负载平衡及故障切换特性。

不管是否使用 GNS 都需要 SCAN。如果使用 GNS，那么 ORACLE 会自动创建 SCAN。如果不使用 GNS，那就必须在 DNS 中定义 SCAN。

磁盘设定

本地磁盘：各虚拟机节点本地磁盘 100G
共享存储：两个 50G 磁盘文件用于 ASM；

操作系统用户和用户组

```
[root@racnode1 ~]# /usr/sbin/groupadd -g 1000 oinstall
[root@racnode1 ~]# /usr/sbin/groupadd -g 1001 dba
[root@racnode1 ~]# /usr/sbin/useradd -u 1100 -g oinstall -G dba oracle
[root@racnode1 ~]# passwd oracle
.....
[root@racnode1 ~]# id oracle
uid=1100(oracle) gid=1000(oinstall) groups=1000(oinstall),1001(dba)
```

2.2 初始化第一台节点

2.2.1 设置操作系统环境变量

先说一下，也可以不配置，在 OUI 界面中可以由系统自动进行修复。不过这里三思还是决定手动进行配置：

编辑文件：

```
[root@racnode1 ~]# vi /etc/security/limits.conf
```

增加下列内容：

```
oracle soft nproc 2047
```

```
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
```

编译文件:

```
[root@racnode1 ~]# vi /etc/selinux/config
```

修改下列内容(注, 重启后才能生效):

```
SELINUX=disabled
```

编辑文件:

```
[root@racnode1 ~]# vi /etc/pam.d/login
```

增加下列内容:

```
session required pam_limits.so
```

编辑文件:

```
[root@racnode1 ~]# vi /etc/sysctl.conf
```

增加或修改下列内容:

```
fs.aio-max-nr = 1048576
fs.file-max = 6815744
kernel.shmall = 2097152
kernel.shmmax = 4294967296
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 = 1048586
```

注意上述参数中, 需要根据实际情况修改 `kernel.shmmax` 参数的值, 可以直接设置为物理内存大小。

执行命令使之生效:

```
[root@racnode1 ~]# sysctl -p
```

2.2.2 配置网卡地址

不用说了吧, 两个网卡, 分别绑定不同的 IP, 我们这里 `eth0` 绑定 `publicIP`, `eht1` 绑定 `privateIP`。配置好后网络如下:

```
[root@racnode1 ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:FC:F8:BF
          inet addr:192.168.10.201  Bcast:192.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fefc:f8bf/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
RX packets:90517 errors:0 dropped:0 overruns:0 frame:0
TX packets:891 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:14697382 (14.0 MiB) TX bytes:119465 (116.6 KiB)

eth1    Link encap:Ethernet  HWaddr 08:00:27:CA:5A:10
        inet addr:10.0.0.201  Bcast:10.255.255.255  Mask:255.255.255.0
        inet6 addr: fe80::a00:27ff:feca:5a10/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:89571 errors:0 dropped:0 overruns:0 frame:0
        TX packets:52 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:14515911 (13.8 MiB) TX bytes:8581 (8.3 KiB)
```

2.2.3 配置/etc/hosts 文件

编辑 hosts 文件:

```
[root@racnode1 ~]# vi /etc/hosts
```

增加或修改下列内容:

```
#PUBLIC IP:
192.168.10.201  racnode1
192.168.10.202  racnode2
#PRIVATE IP:
10.0.0.201     racpriv1
10.0.0.202     racpriv2
#VIP IP:
192.168.10.211 racvip1
192.168.10.212 racvip2
#SCAN IP:
192.168.10.206 racscan
```

执行 ping 命令确认上述各主机名能被正常 ping 通(VIP 除外, 另外如果 racnode2 还未配置, 则访问有可能不通, 可以等到 racnode2 配置好再来做 ping 测试)。

2.2.4 配置软件目录

创建软件目录并授予相应用户操作权限:

```
[root@racnode1 ~]# mkdir -p /data/app/{oracle,grid}
[root@racnode1 ~]# chown oracle:oinstall /data/app -R
[root@racnode1 ~]# chmod 775 /data/app/ -R
```

2.2.5 配置用户

编辑 oracle 用户属性文件:

```
[root@racnode1 ~]# vi /home/oracle/.bash_profile
```

增加下列内容:

```
export TMP=/tmp
export TMPDIR=$TMP
export ORACLE_BASE=/data/app/oracle
export ORACLE_HOME=$ORACLE_BASE/11.2.0/db_1
export GRID_HOME=/data/app/grid/11.2.0/home_1
export ORACLE_SID=racdb1
export ORACLE_TERM=xterm
export PATH=/usr/sbin:$PATH
export PATH=$ORACLE_HOME/bin:$GRID_HOME/bin:$PATH
export
LD_LIBRARY_PATH=$ORACLE_HOME/bin:/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin/
export
CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib
if [ $USER = "oracle" ]; then
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -p 16384
        ulimit -n 65536
    else
        ulimit -u 16384 -n 65536
    fi
    umask 022
fi
```

注意 ORACLE_SID 环境变量的值哟。

2.3 初始化第二台节点

与第一个节点操作步骤相同, 注意 IP 别配置错了就好。

3、配置共享磁盘

3.1 配置安全访问

这里跳过, 我们选择在 OUI 界面中进行配置。

3.2 创建共享磁盘

VirtualBox 虚拟机上配置共享磁盘。由于 VBOX 虚拟化软件在界面方式创建共享盘步骤稍多，这里为简单起见直接通过命令行方式创建，因此我们需要在虚拟机所在的物理机端，执行 VBoxManage 命令创建多个虚拟机磁盘文件如下：

```
$ VBoxManage createhd --filename /data/vboxdata/share_disk/asm01.vdi --size 50000
--format VDI --variant Fixed
$ VBoxManage createhd --filename /data/vboxdata/share_disk/asm02.vdi --size 50000
--format VDI --variant Fixed
```

将前面创建的文件附加到虚拟机：

```
$ VBoxManage storageattach oel6_11grac1 --storagectl "SATA 控制器" --port 1 --device 0
--type hdd --medium /data/vboxdata/share_disk/asm01.vdi --mtype shareable
$ VBoxManage storageattach oel6_11grac1 --storagectl "SATA 控制器" --port 2 --device 0
--type hdd --medium /data/vboxdata/share_disk/asm02.vdi --mtype shareable
```

注意哟，这一步也需要对**节点 2**(我这里虚拟机名为 oel6_11grac2)执行的哟。

```
$ VBoxManage storageattach oel6_11grac2 --storagectl "SATA 控制器" --port 1 --device 0
--type hdd --medium /data/vboxdata/share_disk/asm01.vdi --mtype shareable
$ VBoxManage storageattach oel6_11grac2 --storagectl "SATA 控制器" --port 2 --device 0
--type hdd --medium /data/vboxdata/share_disk/asm02.vdi --mtype shareable
```

设置文件的类型为共享：

```
$ VBoxManage modifyhd /data/vboxdata/share_disk/asm01.vdi --type shareable
$ VBoxManage modifyhd /data/vboxdata/share_disk/asm02.vdi --type shareable
```

而后两个虚拟机的设备列表中都应该能看到这些磁盘。

3.3 配置共享磁盘

启动虚拟机，在 racnode1 端(也只需在任意一个节点)执行操作，我们目前的环境设置共享存储磁盘：数据存储/dev/sdb, /dev/sdc 各 50G。在 linux 中查看磁盘设备如下：

```
[root@racnode1 ~]# ls /dev/sd*
/dev/sda /dev/sda1 /dev/sda2 /dev/sdb /dev/sdc
```

首先通过 fdisk 对其进行分区，fdisk 命令的用法这里只举一个示例，比如处理/dev/sdb 分区：

```
[root@racnode1 ~]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x2cd79048.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)
```

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to switch off the mode (command 'c') and change display units to sectors (command 'u').

Command (m for help): **n**

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): **1**

First cylinder (1-38, default 1):

Using default value 1

Last cylinder, +cylinders or +size {K,M,G} (1-38, default 38):

Using default value 38

Command (m for help): **w**

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

如果还有不明白的，可以参考三思之前的安装笔记。分区后的设备信息如下：

```
[root@racnode1 ~]# ls /dev/sd*
```

```
/dev/sda /dev/sda1 /dev/sda2 /dev/sdb /dev/sdb1 /dev/sdc /dev/sdc1
```

3.4 配置 ASM 磁盘

对于 *OEL6.0* 及以上版本，不需要再额外安装 *ASMLib* 驱动，它已经被编译到内核中，但是 *oracleasm-support* 包必须安装，该软件包安装光盘上就有。

配置 asm 驱动，会要求指定驱动的属主用户及用户组等信息：

```
[root@racnode1 ~]# oracleasm configure -i
```

```
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 []: oracle
```

```
Default group to own the driver interface []: dba
```

```
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
```

初始化:

```
[root@racnode1 ~]# oracleasm init
Creating /dev/oracleasm mount point: /dev/oracleasm
Loading module "oracleasm": oracleasm
Mounting ASMLib driver filesystem: /dev/oracleasm
```

创建 ASM 磁盘:

```
[root@racnode1 ~]# oracleasm createdisk asmdisk1 /dev/sdb1
Writing disk header: done
Instantiating disk: done
[root@racnode1 ~]# oracleasm createdisk asmdisk2 /dev/sdc1
Writing disk header: done
Instantiating disk: done
```

如果有多个就继续创建。完成后执行 `oracleasm scandisks`:

```
[root@racnode1 ~]# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
```

查看 asm 磁盘:

```
[root@racnode1 ~]# oracleasm listdisks
ASMDISK1
ASMDISK2
```

到第二个节点执行下列操作:

```
[root@racnode2 ~]# oracleasm configure -i
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 []: oracle
Default group to own the driver interface []: dba
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
[root@racnode2 ~]# oracleasm init
Creating /dev/oracleasm mount point: /dev/oracleasm
Loading module "oracleasm": oracleasm
Mounting ASMLib driver filesystem: /dev/oracleasm
```



```
[root@racnode2 ~]# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
Instantiating disk "ASMDISK1"
Instantiating disk "ASMDISK2"
[root@racnode2 ~]# oracleasm listdisks
ASMDISK1
ASMDISK2
```

4、安装 GRID 基础框架

环境初始化好后，下面要进入到安装环境了，前面的操作全都是用 root 用户，接下来的操作如非特别注明，则全都是用 oracle 用户操作的，而且基本告别命令行界面，都是在视窗界面执行的，另外如非特别注明，都是在 racnode1 端操作。

我已经提前下载好了 *grid* 的安装包，放在 */data/software/grid* 目录下。如果是远程操作的话，可以配置 *vmc* 进入到视窗界面。

也可以在安装前先执行 */data/software/grid/runcluvfy.sh stage -pre crsinst -n racnode1,racnode2* 对要安装的节点进行检测

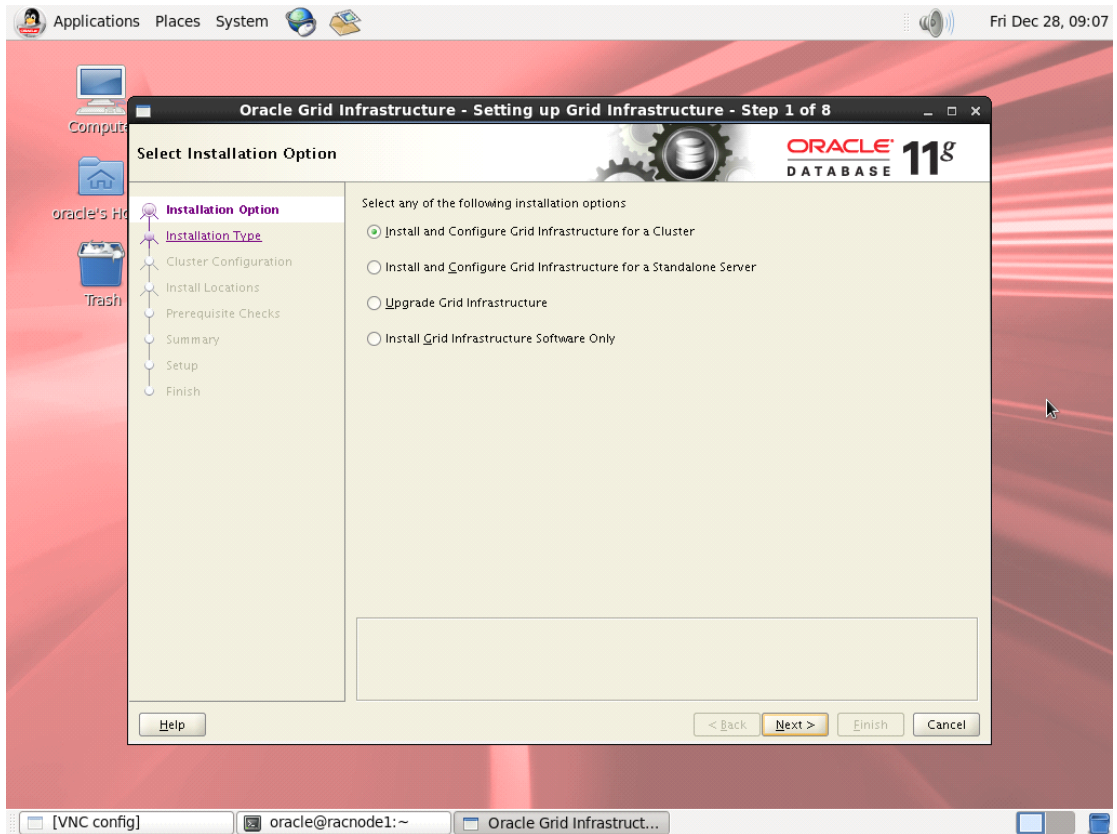
执行 *./runInstaller* 命令进入安装界面：

```
[root@racnode1 ~]# xhost +
[root@racnode1 ~]# su - grid
[oracle@racnode1 ~]$ /data/software/grid/runInstaller
Starting Oracle Universal Installer...

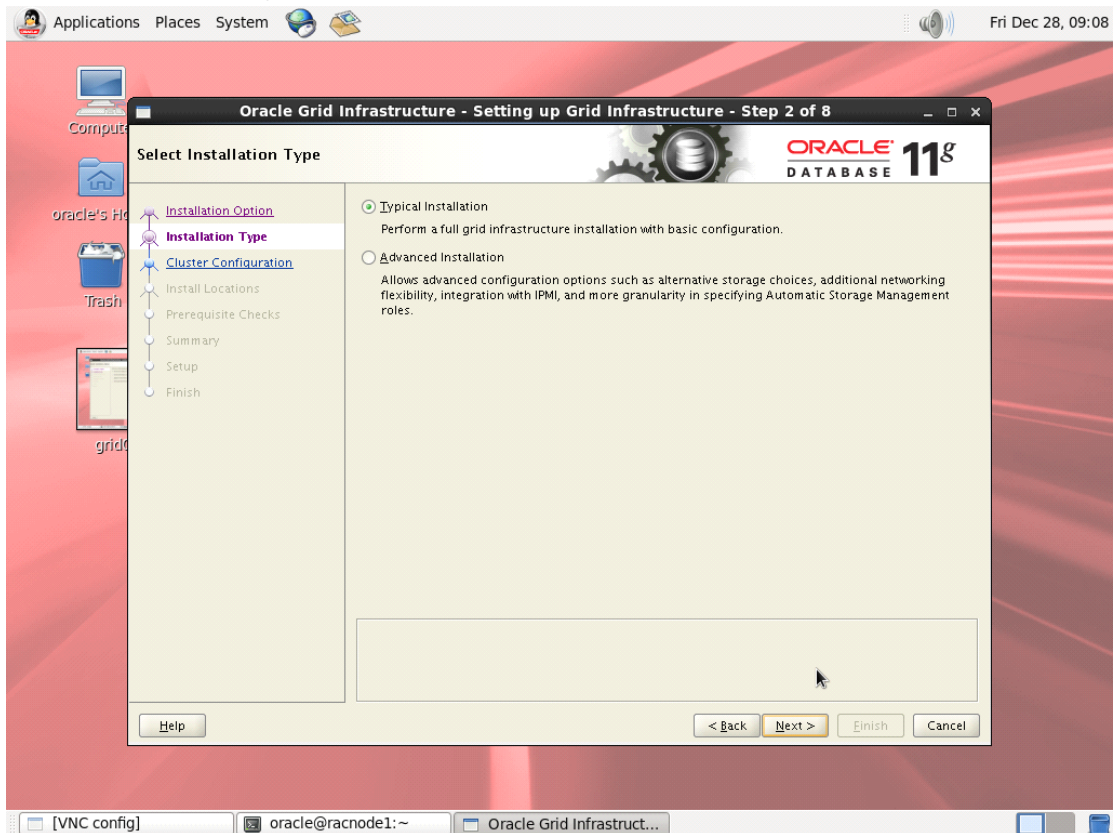
Checking Temp space: must be greater than 120 MB.   Actual 19856 MB   Passed
Checking swap space: must be greater than 150 MB.   Actual 2047 MB   Passed
Checking monitor: must be configured to display at least 256 colors
    >>> Could not execute auto check for display colors using command /usr/bin/xdpyinfo.
Check if the DISPLAY variable is set.   Failed <<<<<

Some requirement checks failed. You must fulfill these requirements before
continuing with the installation,

Continue? (y/n) [n] y
等待一小会儿，安装界面就会出现：
```

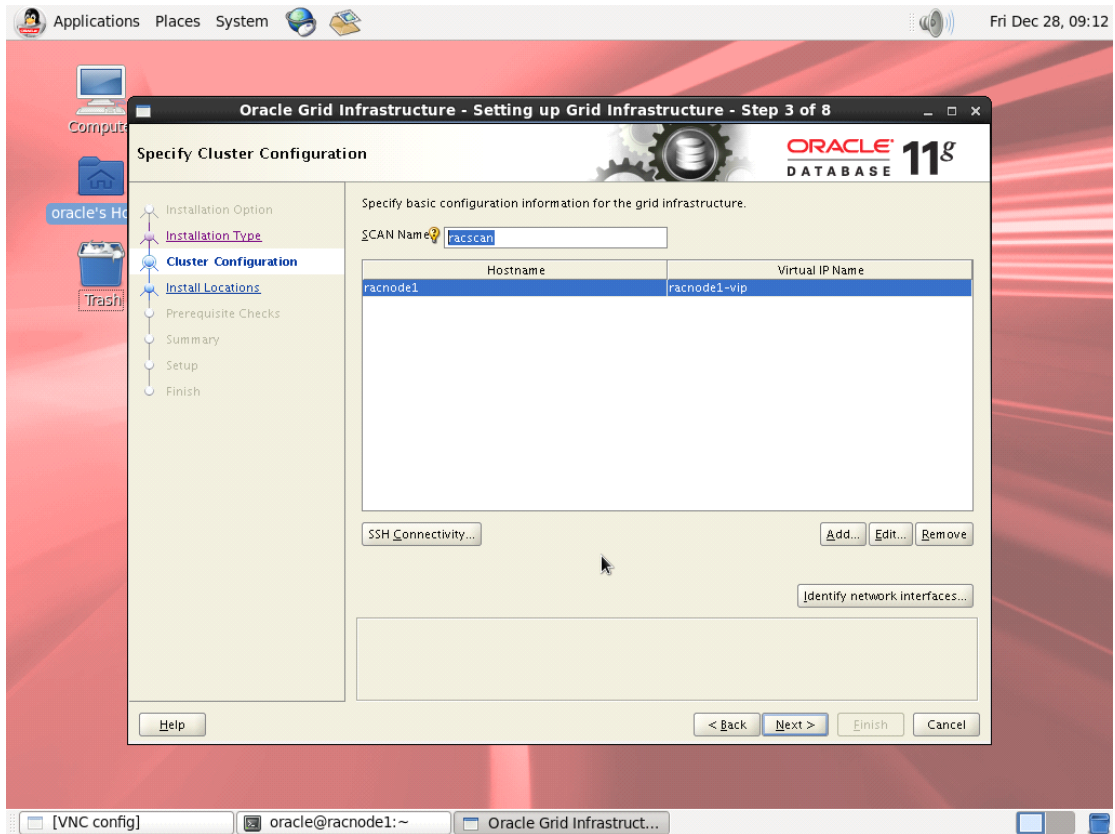


选择"Install and Configure Grid Infrastructure for a Cluster", 点击下一步:

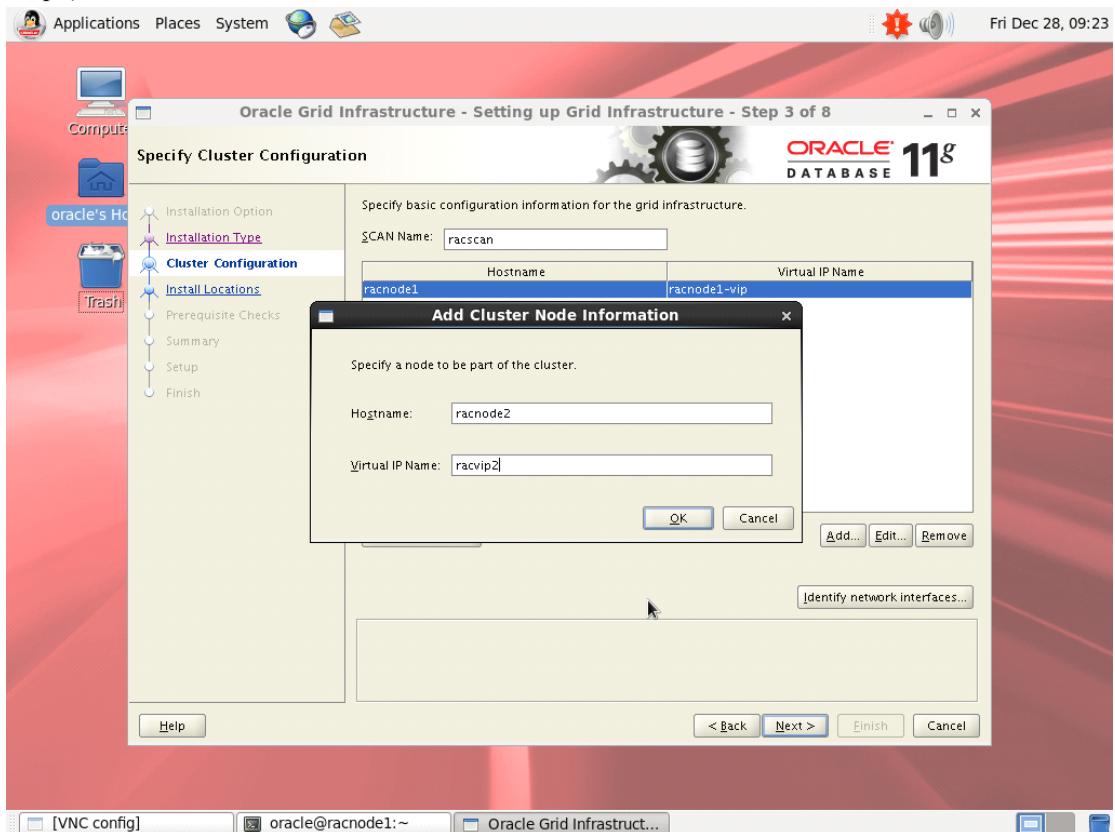


选择"Typical Installation", 点击下一步

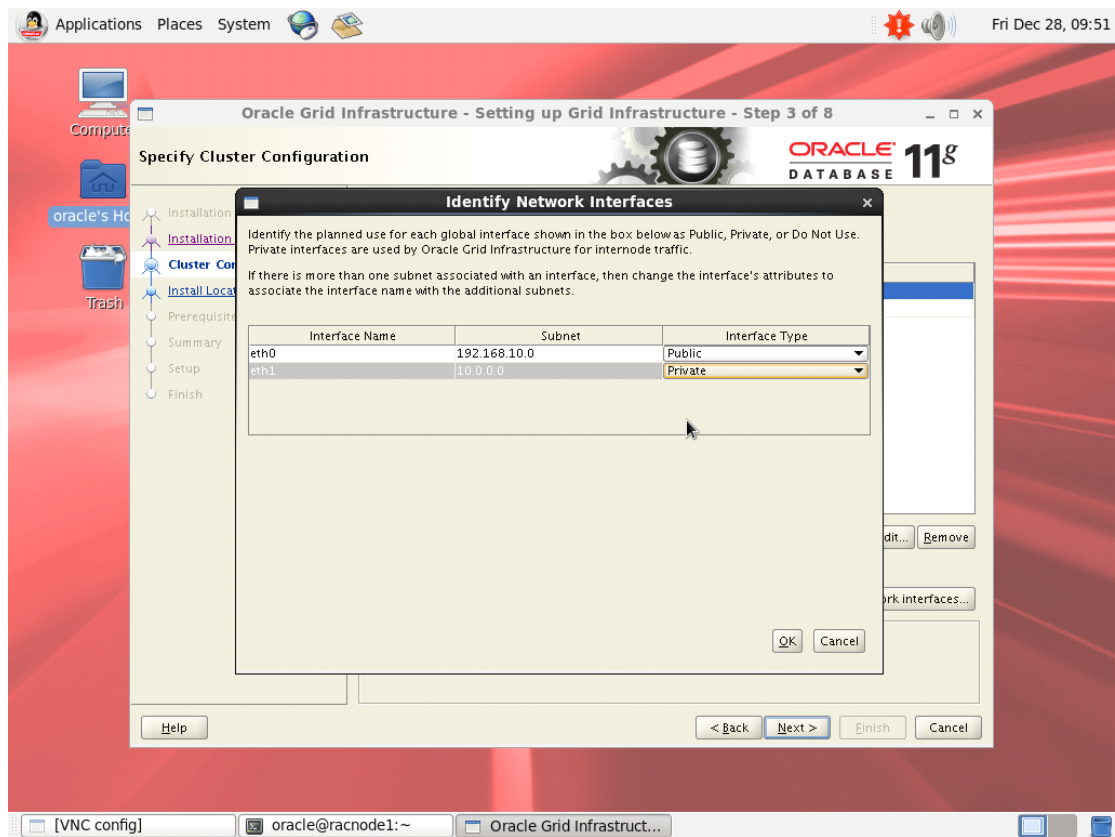
在"SCAN Name"列指定一个集群的唯一名称，可以是主机名，这里指定为前面在 hosts 中配置的 racscan:



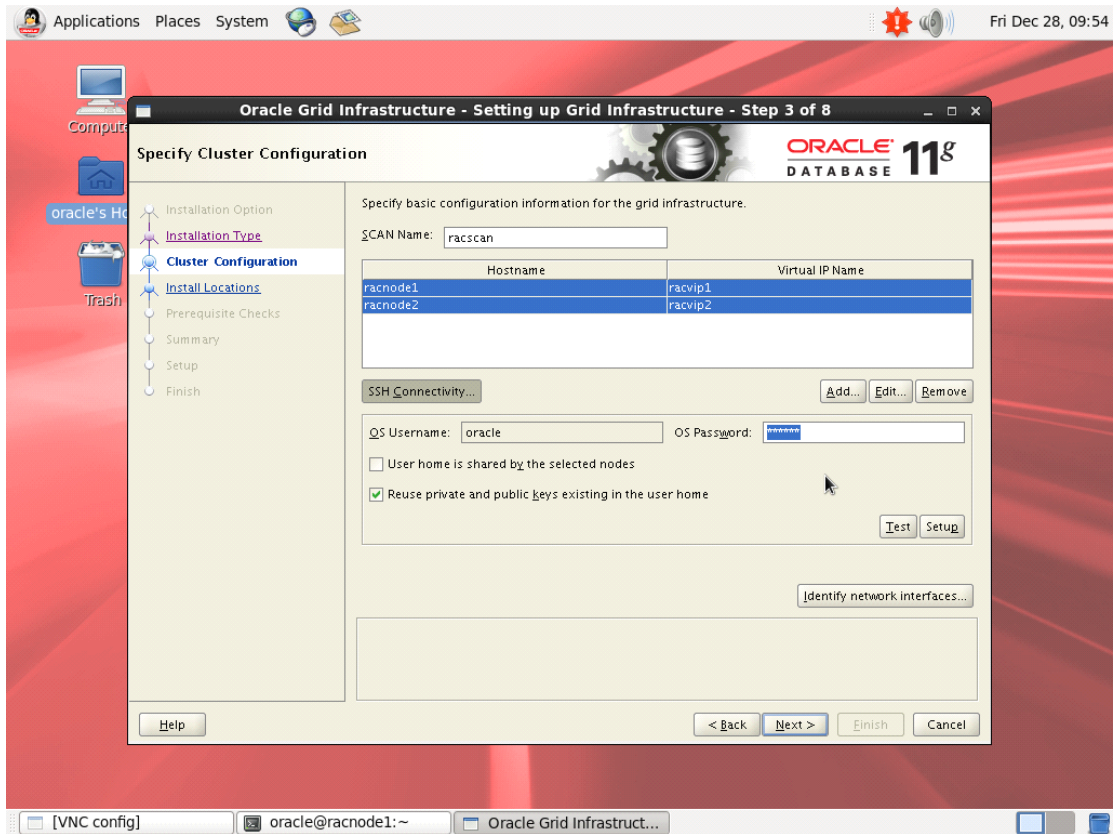
点击右侧中间的"Add"增加第二节点，指定第二个节点的公共名称(racnode2)及虚拟 IP 名(racvip2)，完成后点击 ok:



点击"Identify Network Interfaces"按钮。在窗口中，检查每个网卡是否正确，如果列出的网卡集群中并不使用，那么就将网卡类型修改为 Do Not Use。注意两个节点都要看的哟：

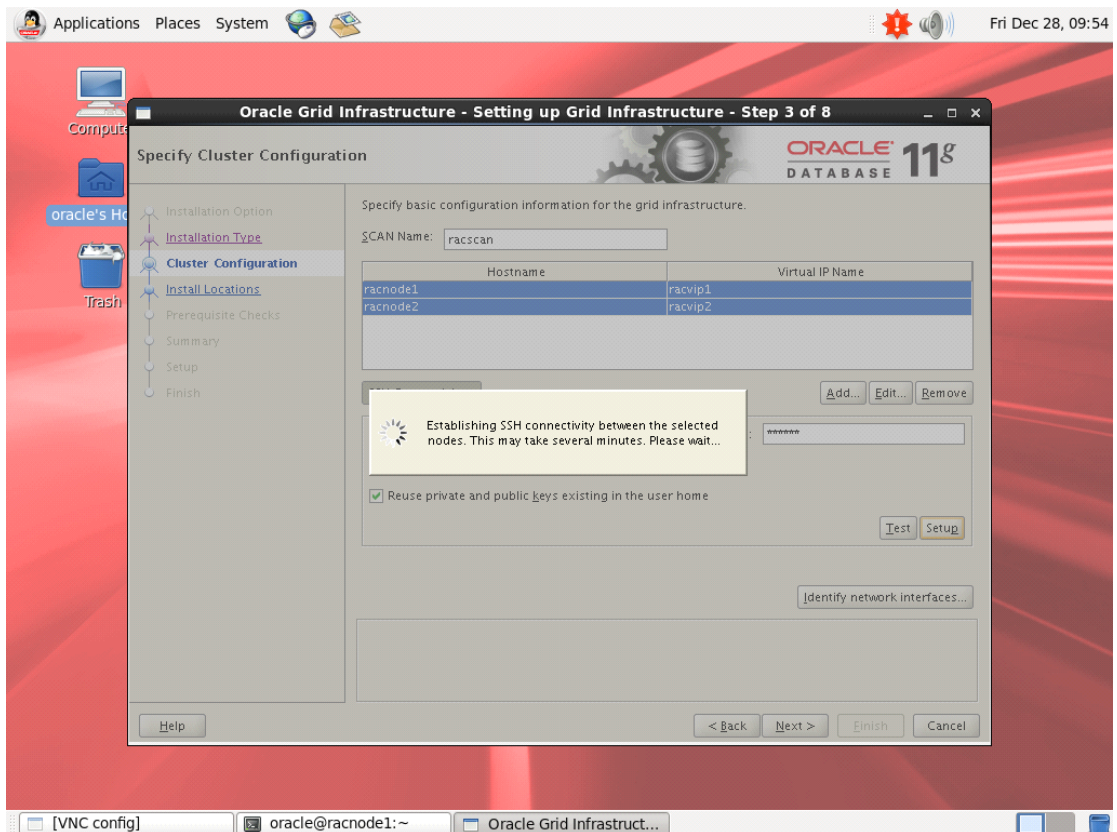


选择两个节点后，点击"SSH Connectivity"，在窗口右侧，会显示 SSH 连接信息。指定 ORACLE 软件属主的用户名(oracle)和密码。选中选项"Reuse private and public keys existing in the user home"，点击"Setup"：



提示：相比 10g 版本此处就简化多了，不必再手动命令行配置 SSH 安全连接，直接在安装界面就一条龙搞定。

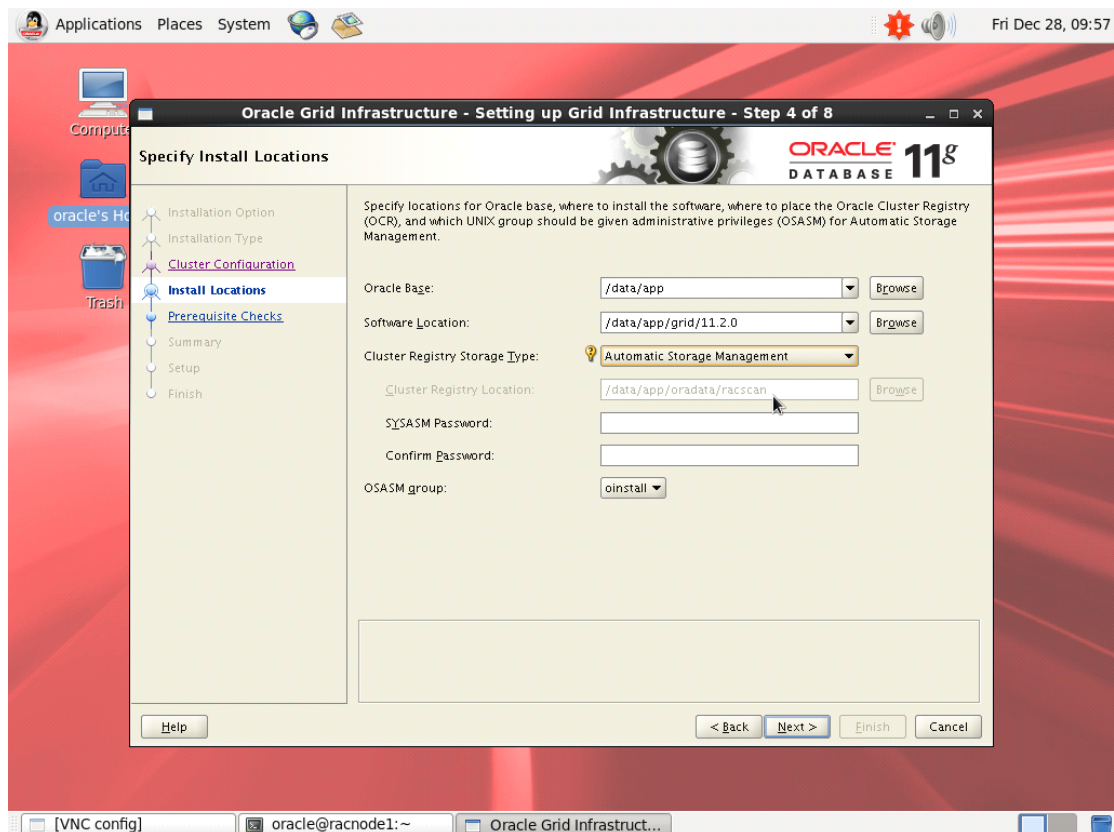
弹出消息窗口，指出可能需要花费数分钟时间配置节点间的 SSH 通讯：



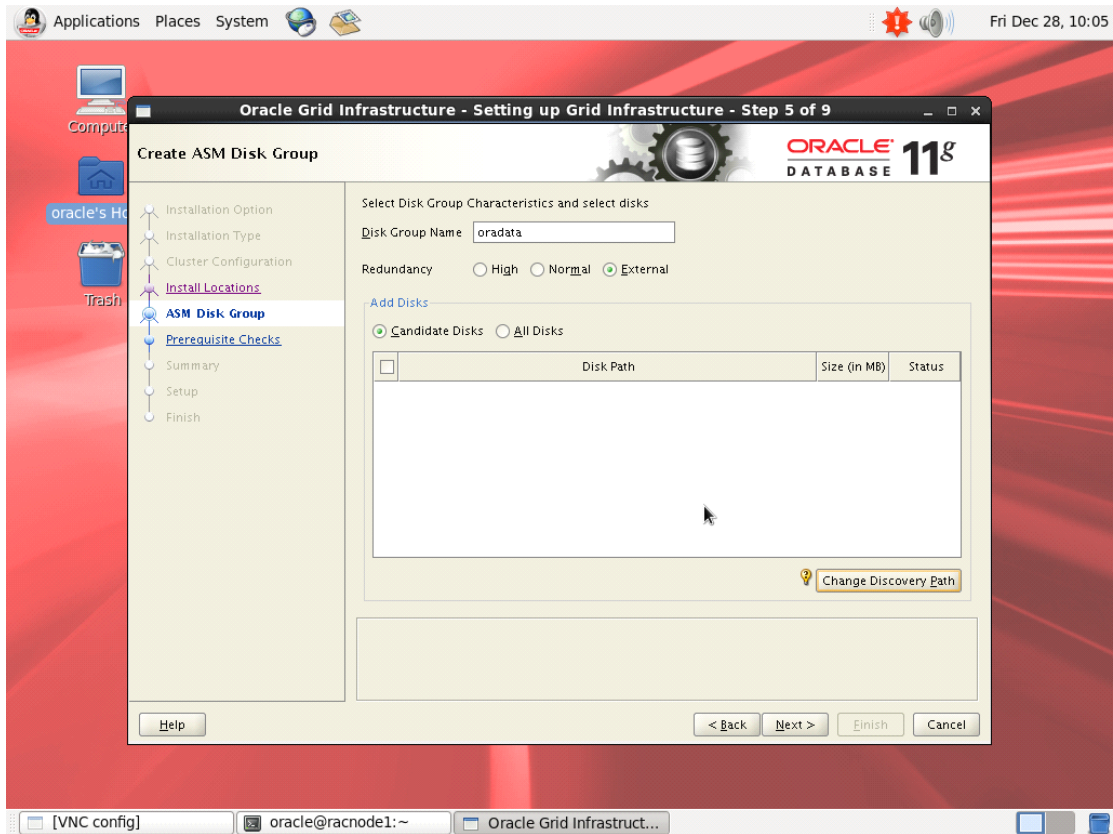
而后另一个信息窗口出来，指出无须密码的 SSH 连接配置好，点击"OK"继续，返回上层界面后可以再点击"Test"测试一下无密码访问是否顺利。都 ok 的话，点击 Next 继续吧。

这一页执行下列操作：

- 在 Oracle base 列，确认 Oracle 主目录路径正确(例如/data/app/grid/)，如果需要修改就点击 Browse。
 - 在 Software Location 列，确认 Grid 主目录路径正确(例如/data/app/grid/11.2.0)，否则也可以修改；
 - 对于 Cluster Registry Storage Type，选择 ASM；
 - 指定 SYSASM 用户的密码，该密码用于管理 ASM；
 - 对于 OSASM 组，使用下拉列表选择对应的管理 ASM 的操作系统用户组，例如 dba；
- 全部完成后点击 Next，创建 ASM 磁盘组窗口出现。

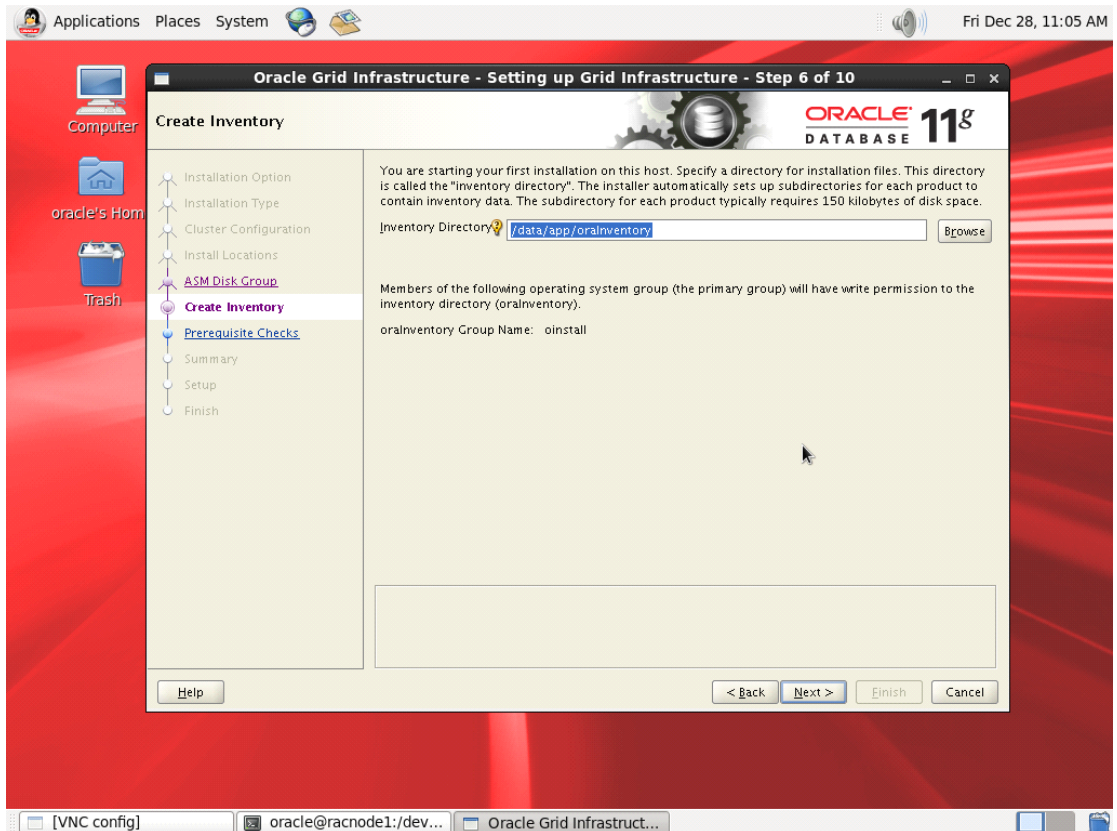


在 Disk Group Name 列，指定一个磁盘组的名称，比如说 ORADATA，选择它的冗余级别，我们这里选择了外部，在添加磁盘环节为它指定磁盘，全部完成后点击 Next

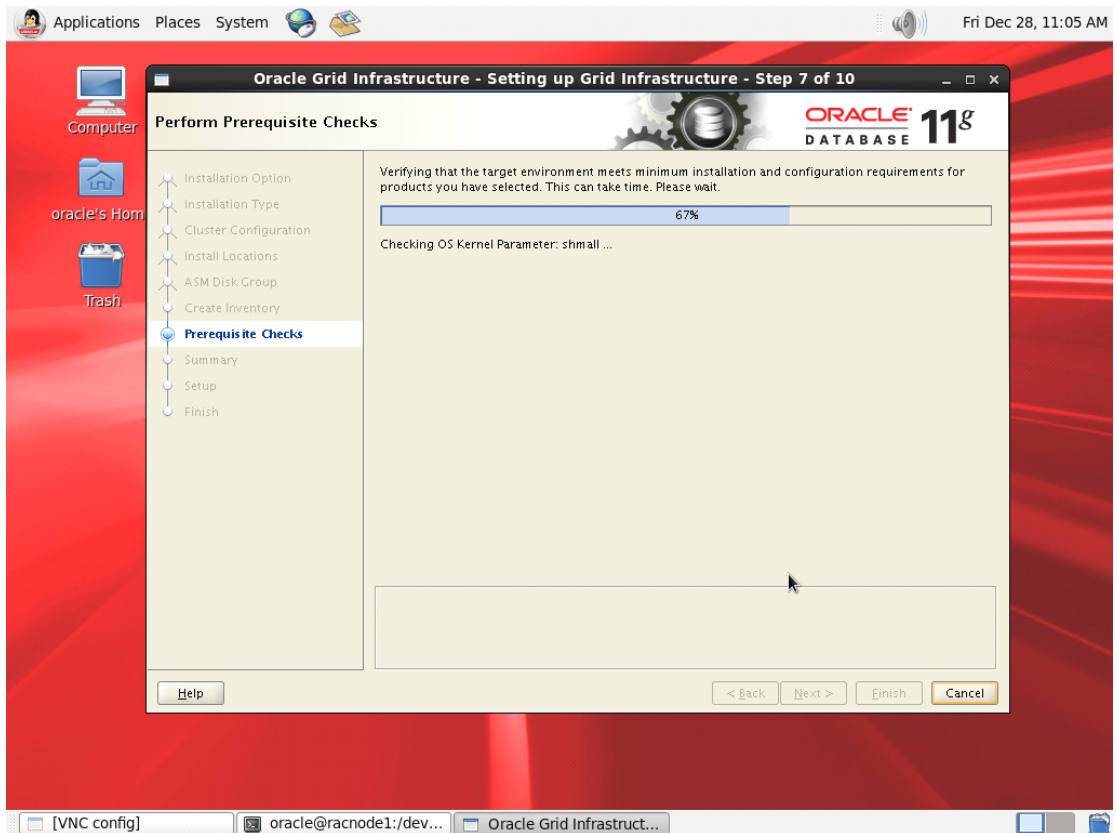


提示，这里如果看不到磁盘，可以点击“Change Discovery Path”，修改查找路径，指定路径到/dev/oracleasm/disks 目录下一般就能看到了。如果仍然找不到，需要检查一下 oracle 用户的权限，比如直接在操作系统层通过 oracle 用户访问上述路径中的磁盘，看看能否顺利访问，只要操作系统层没有问题，这里就一定能够正常访问。如果操作系统层访问时提示无权限，那么可能权限配置有误，导致配置有误的因素较多，比如用户，用户组，oracleasm 配置等等，建议逐项检查。全部检查后均无问题，仍然看不到磁盘的话，最后一招，重启虚拟机试试看。

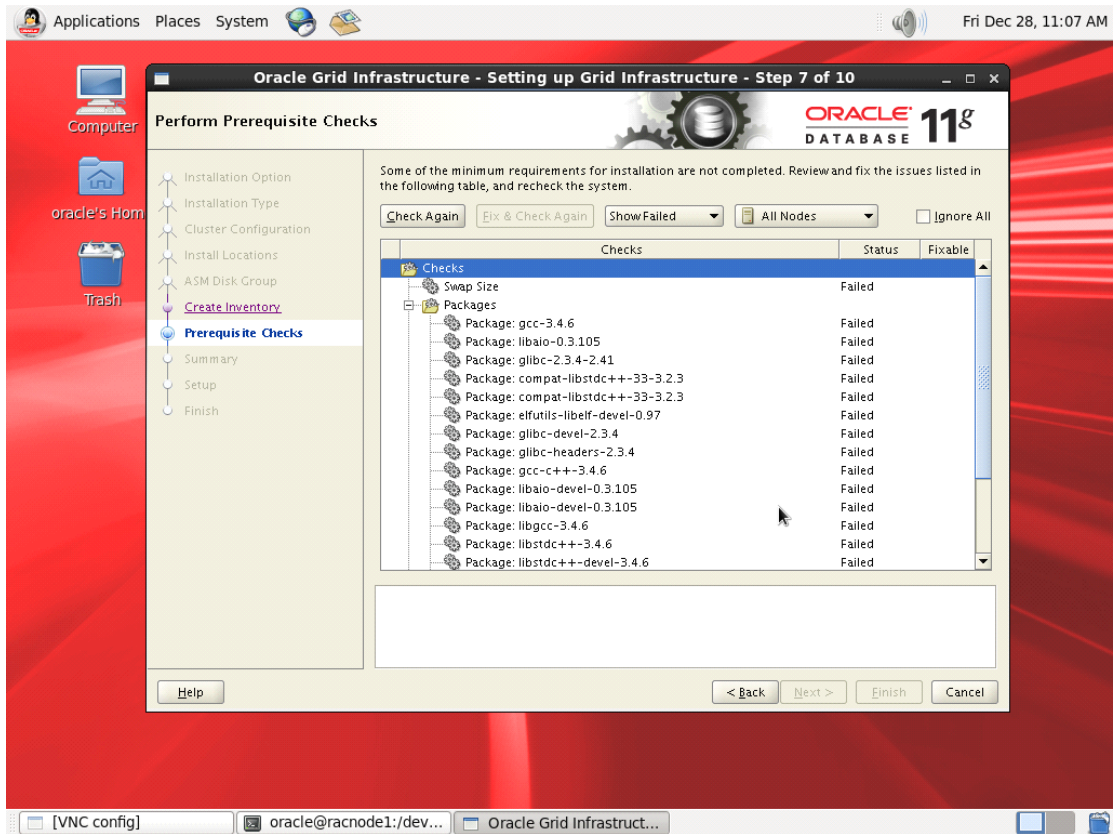
如果需要的话，修改 oraInventory 目录路径，该路径的用户组应该是 oinstall:



执行系统检查(图 12):

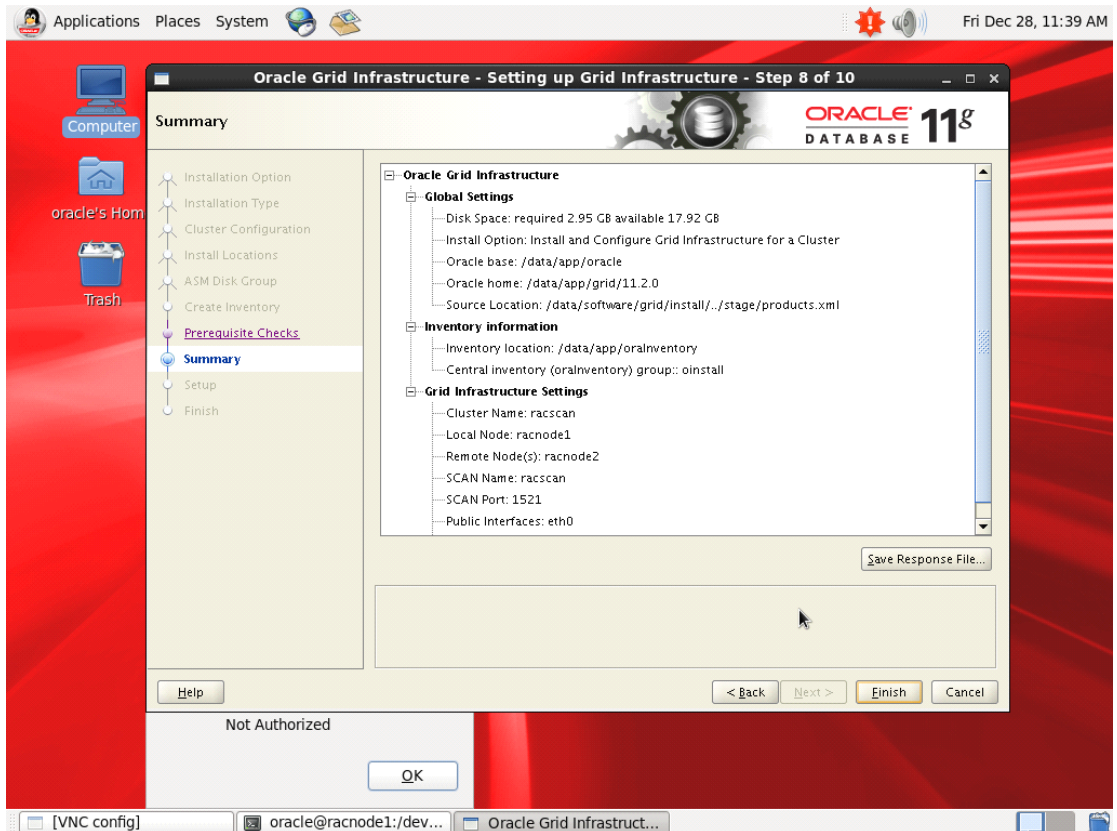


如果检查过程中发现错误，对于内核参数错误，可以点击 Fix & Check Again 按钮，使用 root 用户运行指定的脚本修复，如果是缺包的话，需要手动安装了：

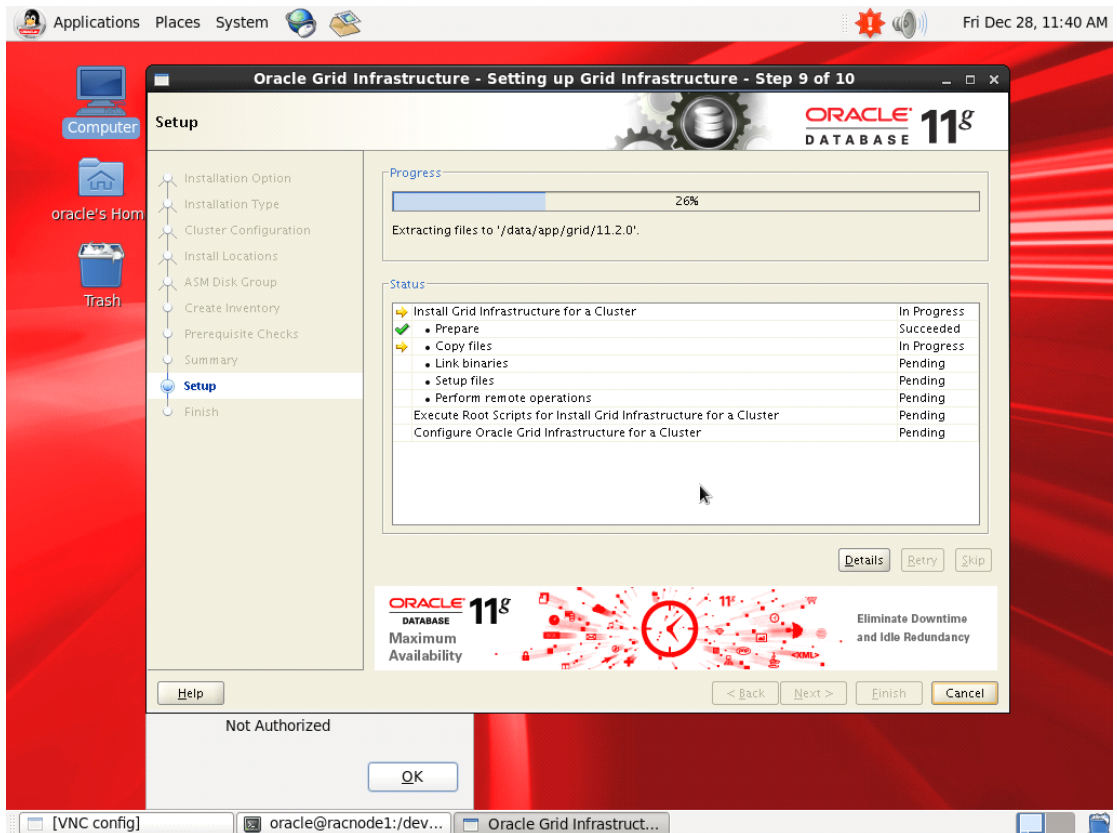


提示：像我这里使用的 OEL6，在安装时总提示 pdksh 包未安装，而且该软件包也确实没有在系统光盘中，不过只要我们安装了 ksh 包，对于 pdksh 包实际上可以忽略。

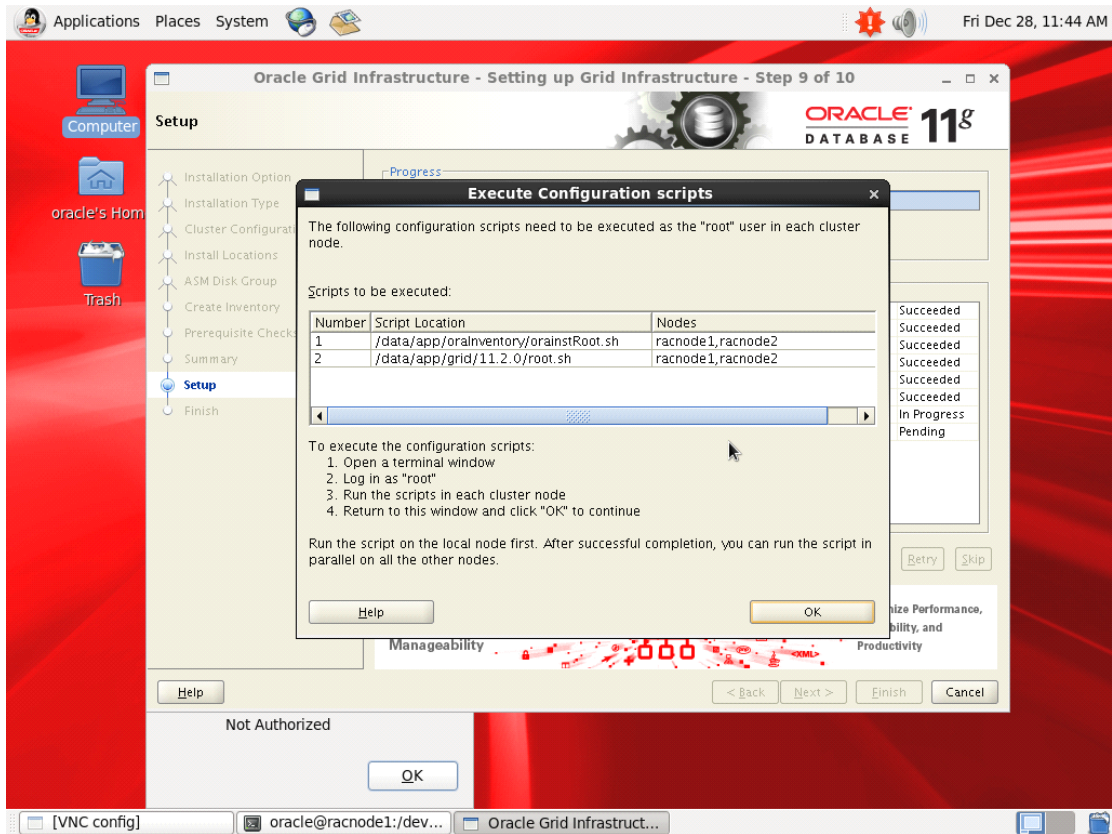
全部修复后点击 check again 验证一下，无误的话进入下一步，查看预览，如无问题点击 Finish：



文件安装和复制中:



做为安装的一部分，用户需要按照提示，以 root 用户身份执行下列脚本：



执行配置脚本窗口显示了配置脚本以及配置文件所在的路径。所有节点都要执行这两个脚本。注意哟，执行有顺序，操作步骤如下：

A. Racnode1 节点，以 root 身份执行下列操作：

```
[root@racnode1 ~]# /data/app/orainventory/orainstRoot.sh
Changing permissions of /data/app/orainventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.
```

```
Changing groupname of /data/app/orainventory to oinstall.
The execution of the script is complete.
```

B. 节点 1 成功后登录到 racnode2，执行相同操作：

```
[root@racnode2 ~]# /data/app/orainventory/orainstRoot.sh
Changing permissions of /data/app/orainventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.
```

```
Changing groupname of /data/app/orainventory to oinstall.
The execution of the script is complete.
```

C. 返回节点 1 执行第二个脚本：

```
[root@racnode1 ~]# /data/app/grid/11.2.0/home_1/root.sh
Running Oracle 11g root.sh script...
```

The following environment variables are set as:


```
ORACLE_OWNER= grid
ORACLE_HOME= /data/app/grid/11.2.0/home_1
```

Enter the full pathname of the local bin directory: [/usr/local/bin]:

Copying dbhome to /usr/local/bin ...

Copying oraenv to /usr/local/bin ...

Copying coraenv to /usr/local/bin ...

Creating /etc/oratab file...

Entries will be added to the /etc/oratab file as needed by

Database Configuration Assistant when a database is created

Finished running generic part of root.sh script.

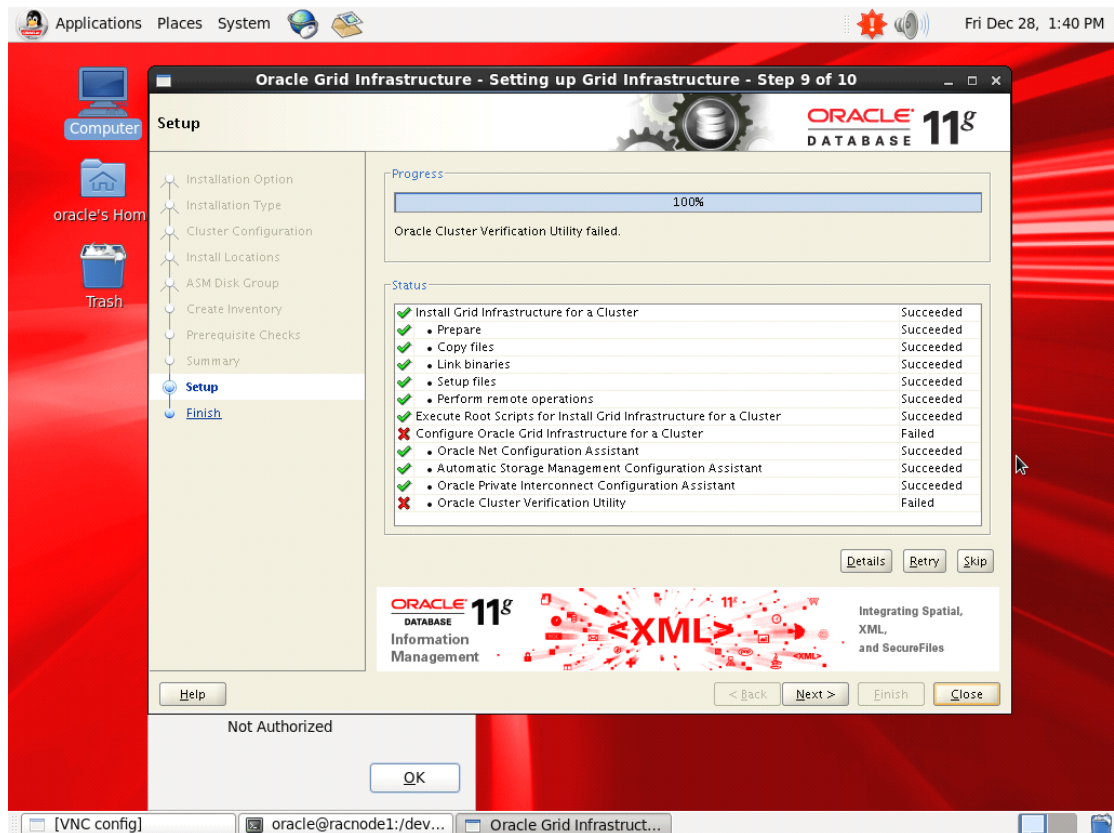
.....

提示：必须先在第一个节点上运行 root.sh，最后一个节点必须等第一个节点执行成功后再运行 root.sh 脚本。

D. 到节点 2 执行第二个脚本：

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

Root.sh 脚本执行成功后，返回 OUI 界面，这里提示遇到一个错误，根据提示的日志分析是由于无法解析到 racscan 导致的，直接忽略，点击 OK。



点击 Close 完成安装进程并退出 OUI 界面。

GRID 基础框架的安装就完成了。安装好集群件后，检查节点上的应用是否启动。
以 oracle 用户身份执行下列命令：

```
[oracle@racnode1 ~]$ /data/app/grid/11.2.0/home_1/bin/crsctl check cluster -all
*****
racnode1:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
racnode2:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
[oracle@racnode2 ~]$ /data/app/grid/11.2.0/bin/crsctl check cluster -all
*****
racnode1:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
racnode2:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
```

而后会在命令输出中看到所有集群件守护进程的信息。
如果集群中某些资源显示脱机或没找到，则说明集群件软件没能正确的安装。

常见问题：

A> 执行# /data/app/grid/11.2.0/root.sh 时提示：

```
/data/app/grid/11.2.0/bin/clscfg.bin: error while loading shared libraries: libcap.so.1: cannot
open shared object file: No such file or directory Failed to create keys in the OLR, rc = 127, 32512
```

这是由于确实不存在 libcap.so.1 文件，针对这种情况，有两种解决方式，一种是我们创建一个软链接：

```
# ln -s /lib64/libcap.so.2.16 /lib64/libcap.so.1
```

或者安装相应的包：

```
# yum install compat-libcap1 -y
```

而后清除 crs 配置：

```
# /data/app/grid/11.2.0/crs/install/rootcrs.pl -deconfig --force
```

并再次执行

```
# /data/app/grid/11.2.0/root.sh
```

B> 执行# /data/app/grid/11.2.0/root.sh 时提示:

```
CRS-4124: Oracle High Availability Services startup failed.  
CRS-4000: Command Start failed, or completed with errors.  
ohasd failed to start: Inappropriate ioctl for device  
ohasd failed to start at /data/app/grid/11.2.0/crs/install/rootcrs.pl line 443.
```

参考: <http://blog.csdn.net/tianlesoftware/article/details/7697366> 处理。

针对这个问题, 需要**强调**的是, oel6 以上版本才会出现, 所以你知道, 版本太新太激进不好了吧。如果你认为只是安装时需要多一个额外的步骤处理一下, 那就大错特错了, 这个问题属于 BUG, 不仅仅出现在安装, 在后面重新启动 CRS 时都需要这样, 否则就会遇到:

```
CRS-4124: Oracle High Availability Services startup failed.  
CRS-4000: Command Start failed, or completed with errors.
```

哎, 无意之中给自己刨了个坑, 不过还好, 这个安装只是用于个人测试, 在正式环境部署时, 操作系统还是使用传统些的 OEL5 吧。

既然说到这儿了, 在这种环境下启动 11gR2 RAC 该怎么做呢(正常情况是 11gR2 版本 RAC 是自动启动, 不需要配置的, 但是我们这个环境中自动一定起不来), 操作步骤如下:

两个节点首先**分别**执行:

```
# /bin/dd if=/var/tmp/.oracle/npohasd of=/dev/null bs=1024 count=1
```

而后再**分别**启动:

```
# /data/app/grid/11.2.0/home_1/bin/crsctl start crs  
CRS-4123: Oracle High Availability Services has been started.
```

而后在**某个节点**启动集群:

```
# /data/app/grid/11.2.0/home_1/bin/crsctl start cluster -all  
CRS-5702: Resource 'ora.evmd' is already running on 'racnode1'  
CRS-5702: Resource 'ora.crsd' is already running on 'racnode1'
```

如果一切顺利, 就等着它启动相应的进程就行了。

5、安装数据库软件

接下来的操作如非特别注明, 则全都是用 oracle 用户操作的, 而且基本告别命令行界面, 都是在视窗界面执行的, 另外如非特别注明, 都是在 racnode1 端操作。

安装 oracle 数据库软件大家应该都熟, RAC 环境中安装跟单实例区别不大, 下面实操一下。下载好文件并解析, 执行 runInstaller 进入到安装界面:

```
[oracle@racnode1 ~]$ /data/software/database/runInstaller  
Starting Oracle Universal Installer...
```

```
Checking Temp space: must be greater than 120 MB.   Actual 12567 MB   Passed  
Checking swap space: must be greater than 150 MB.   Actual 1490 MB   Passed  
Checking monitor: must be configured to display at least 256 colors
```

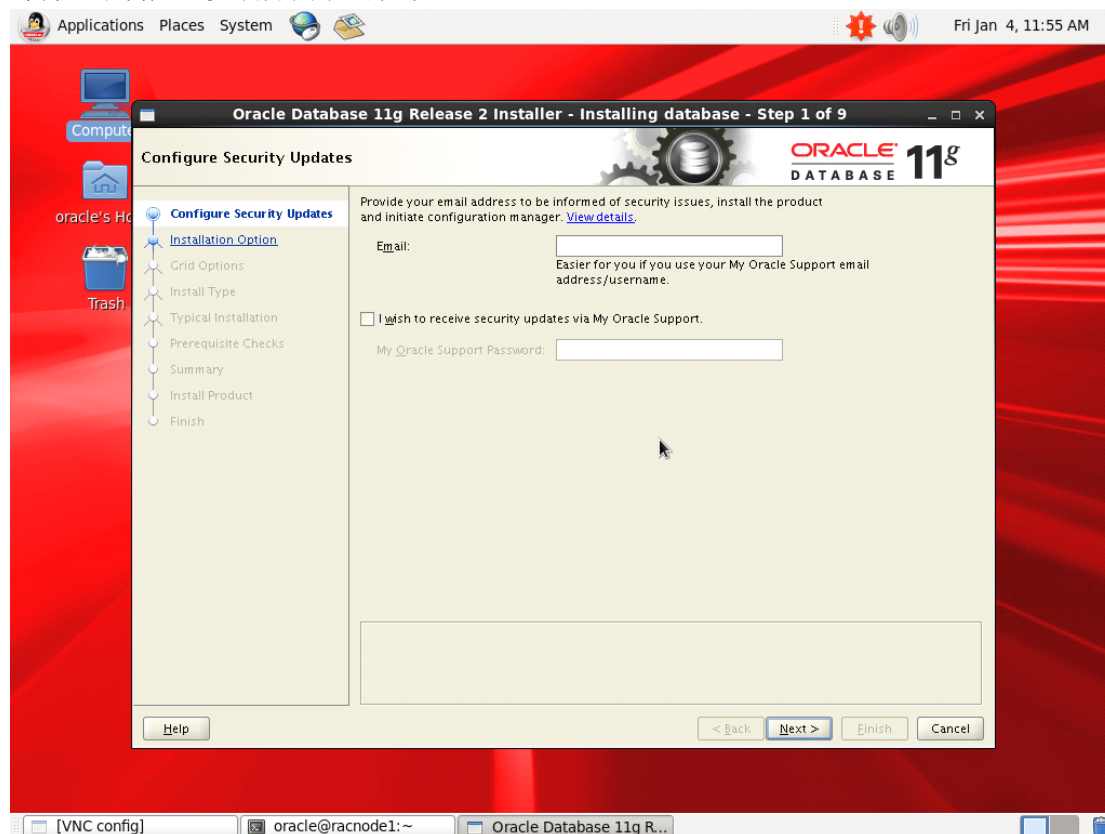

>>> Could not execute auto check for display colors using command /usr/bin/xdpyinfo.
Check if the DISPLAY variable is set. Failed <<<<

Some requirement checks failed. You must fulfill these requirements before

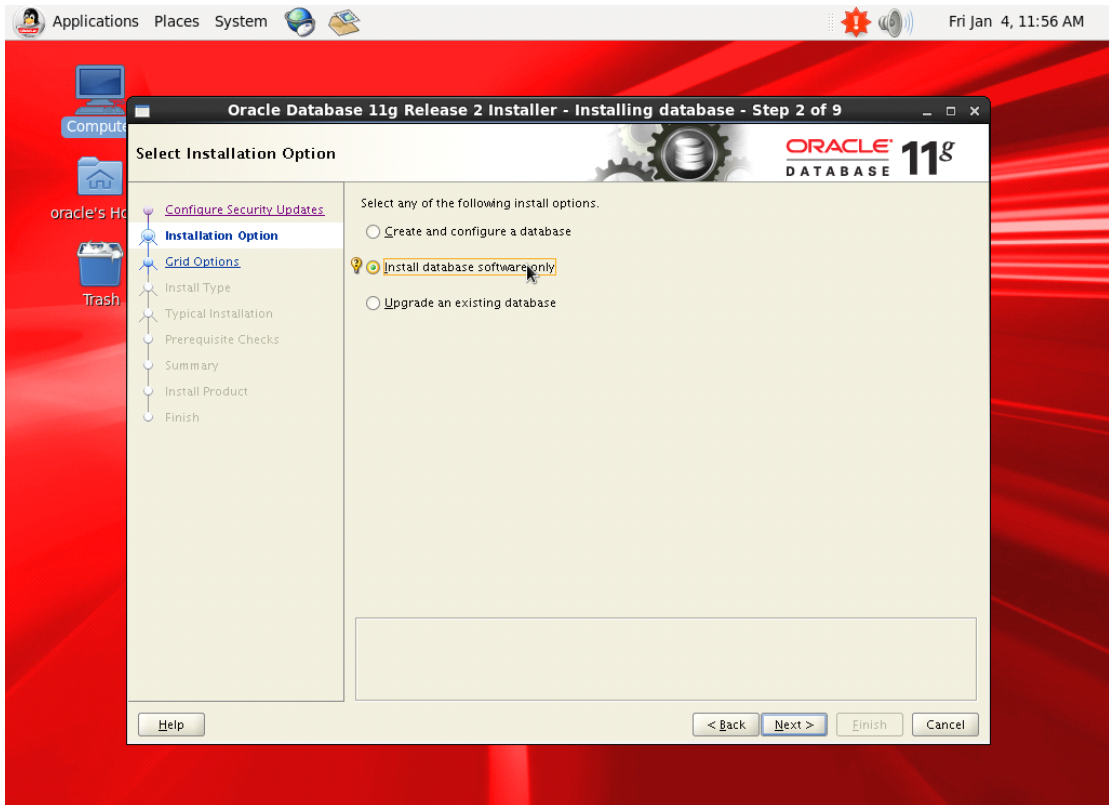
continuing with the installation,

Continue? (y/n) [y]

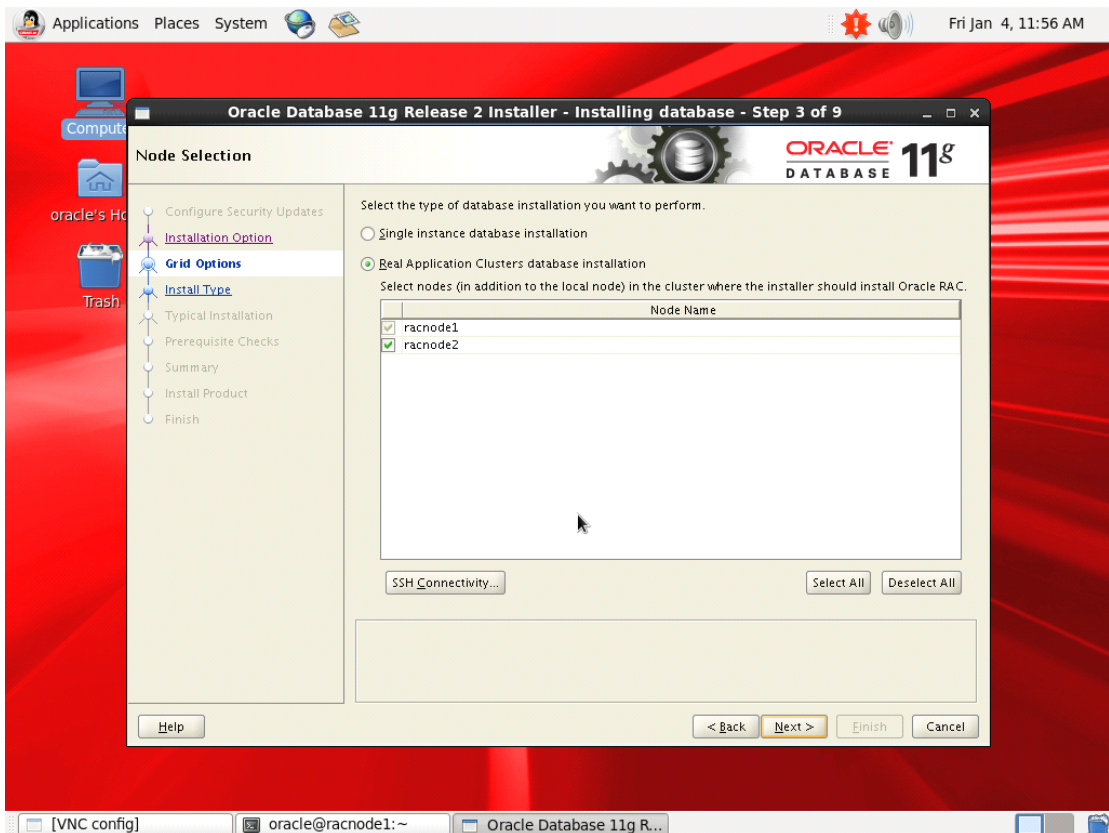
等待一小会儿，安装界面就出现了：



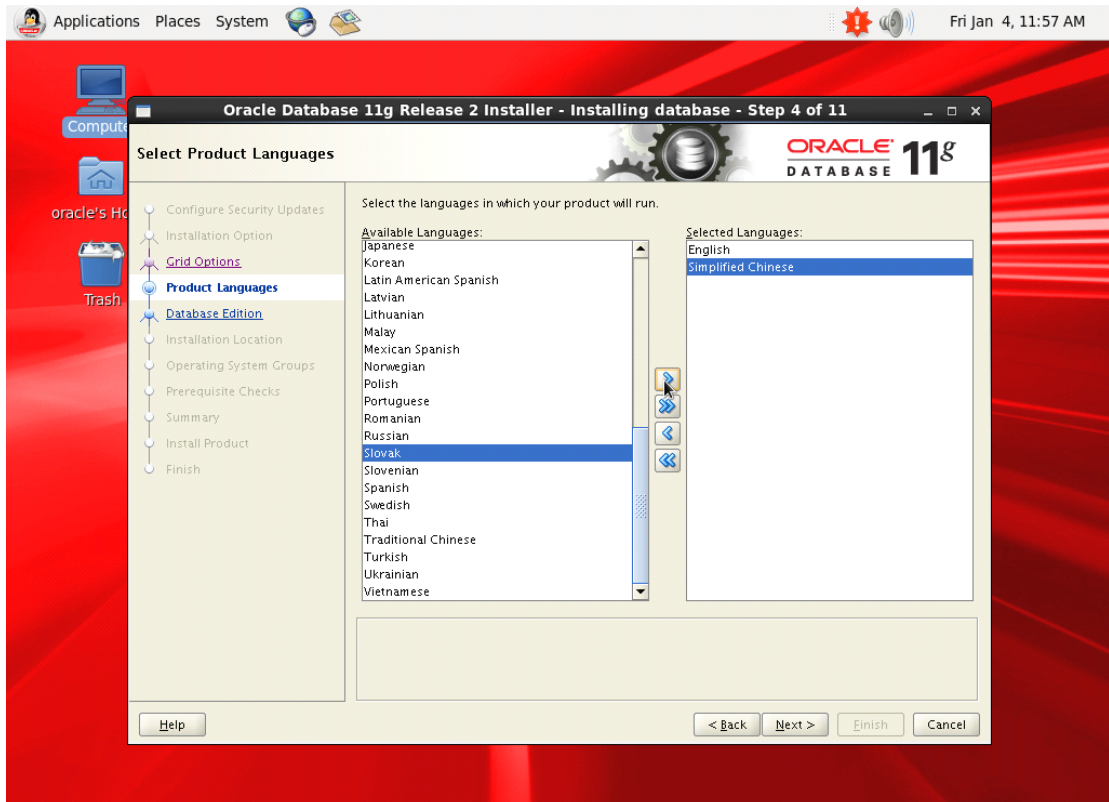
这里我们选择只安装数据库软件：



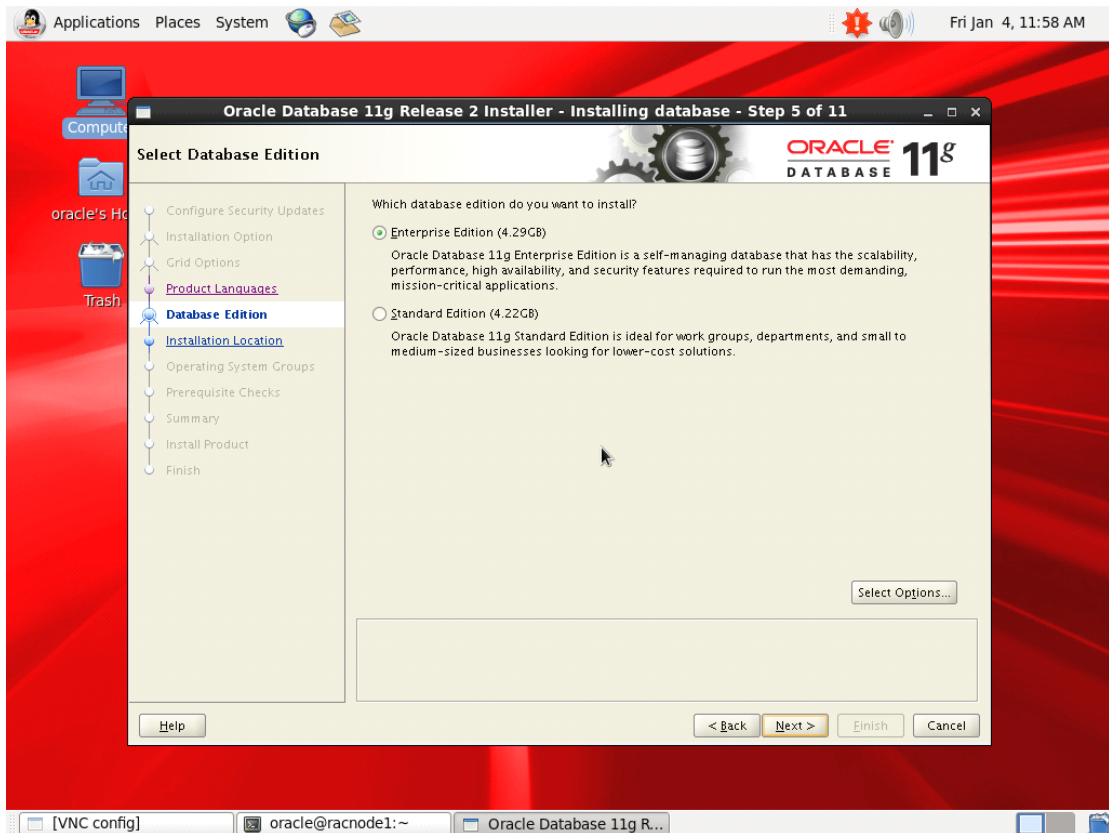
选择 RAC 的数据库安装，注意要把所有的节点都选中。SSH Connectivity 在前面安装 grid 基础框架时已经配置好，这里就不需要重复配置了：



选择语言，默认是英文，我们增加一项简体中文：

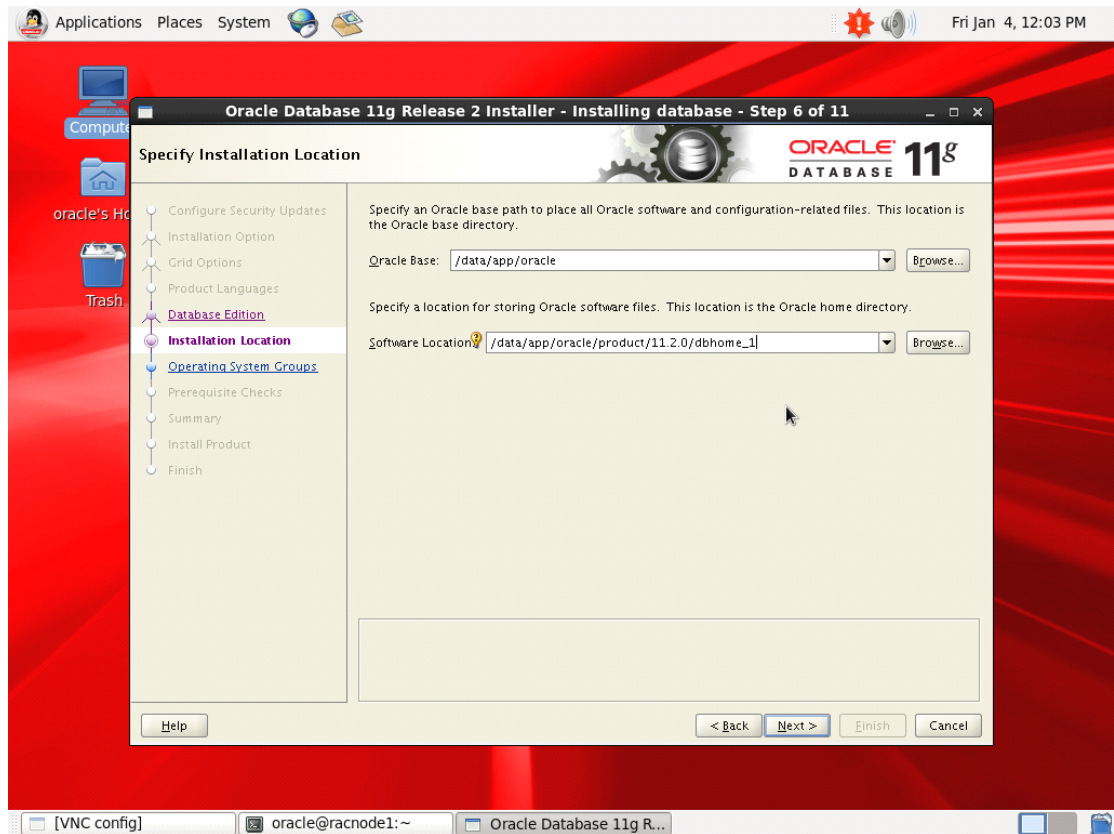


选择数据库版本，标准版还是企业版，企业版那就是必须的呀：



提示：选中企业版的话，options中有些组件供选，一般默认即可，vault和label都属于加强安装审计的特性一般都不需要。

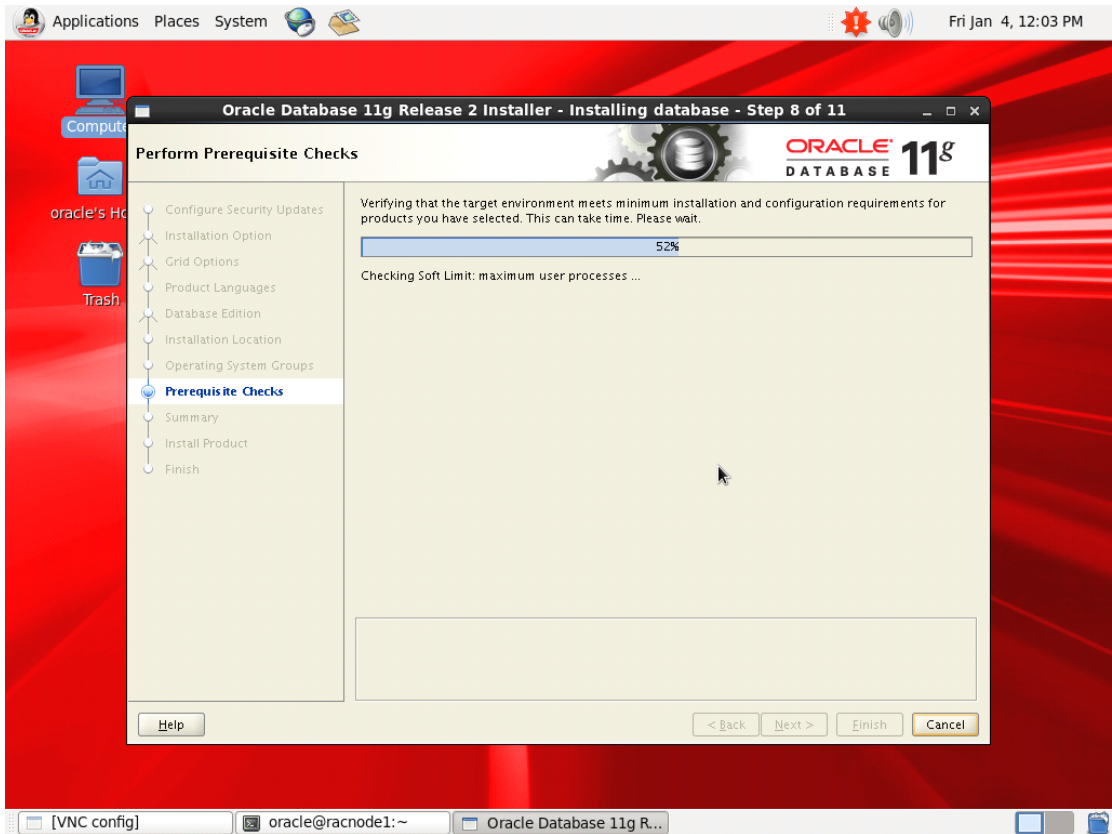
选择软件的安装路径:



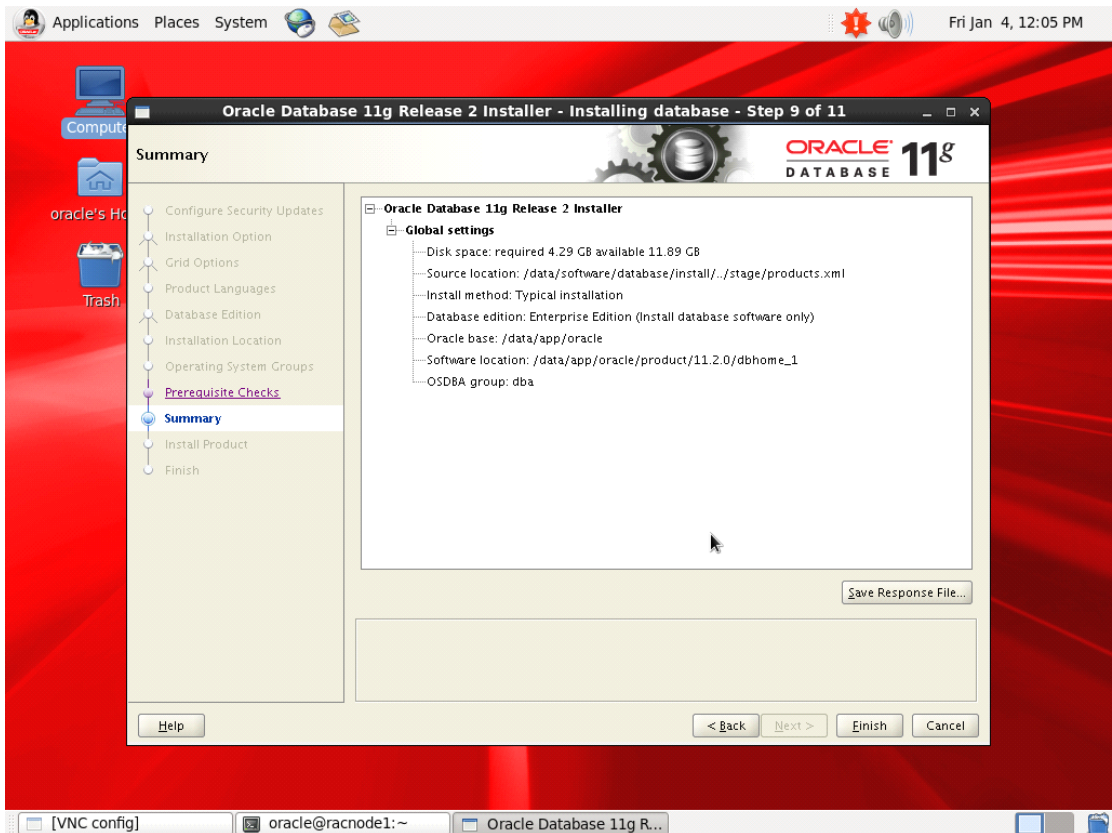
选择用户组，默认即可:



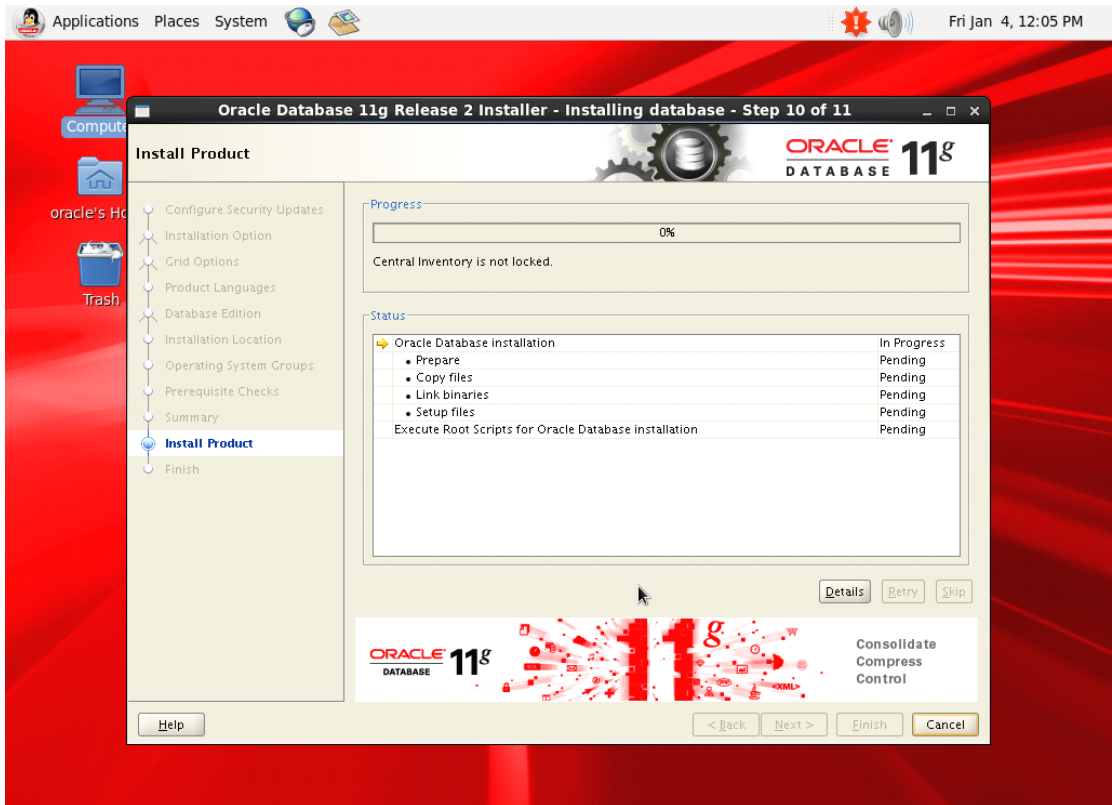
接下来将对系统进行检查，跟前面安装 grid 时差不多:



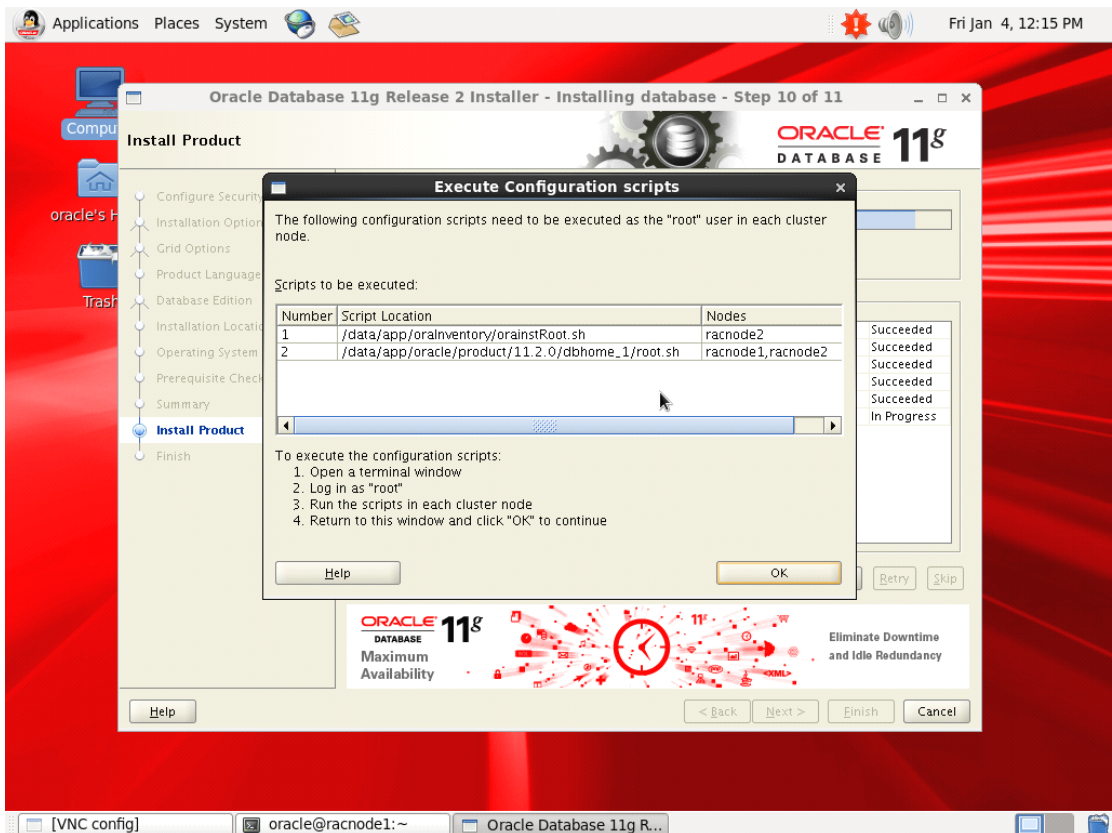
提示汇总预览信息:



执行安装:



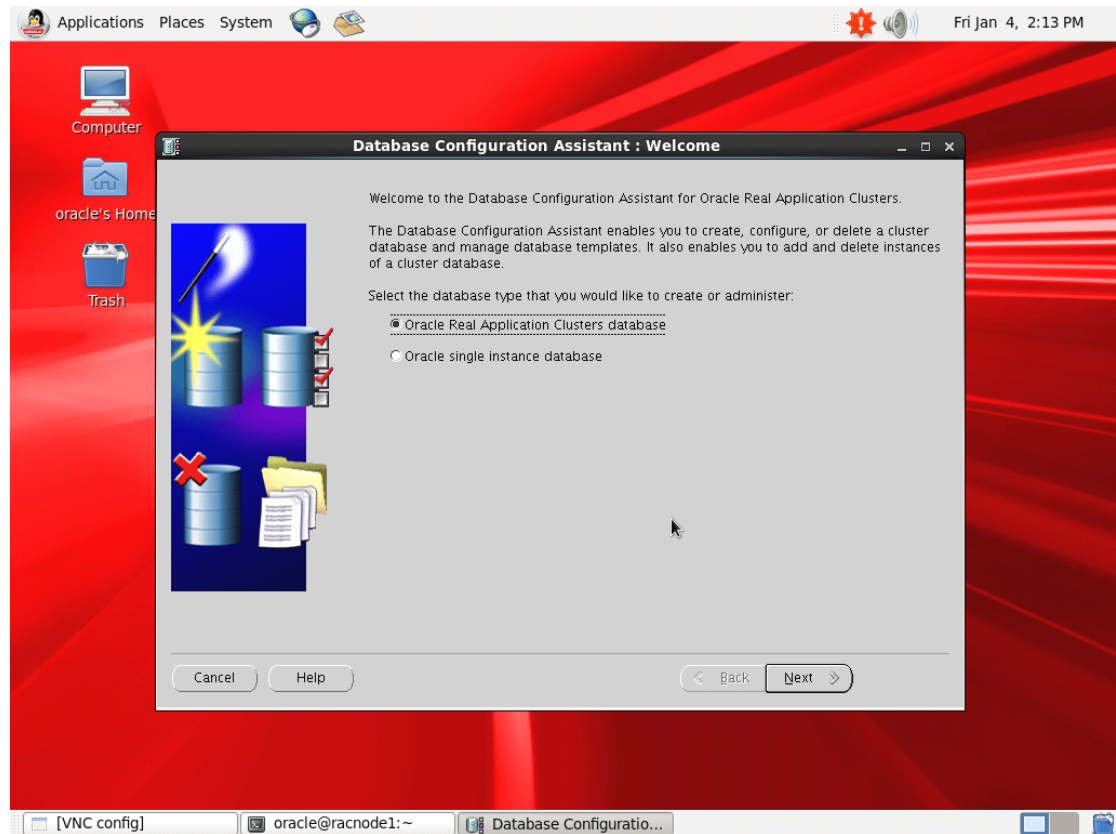
根据提示使用 root 用户执行脚本，执行有顺序，先在节点 1 执行，而后再到节点 2 执行，注意要用 root 用户执行的哟：



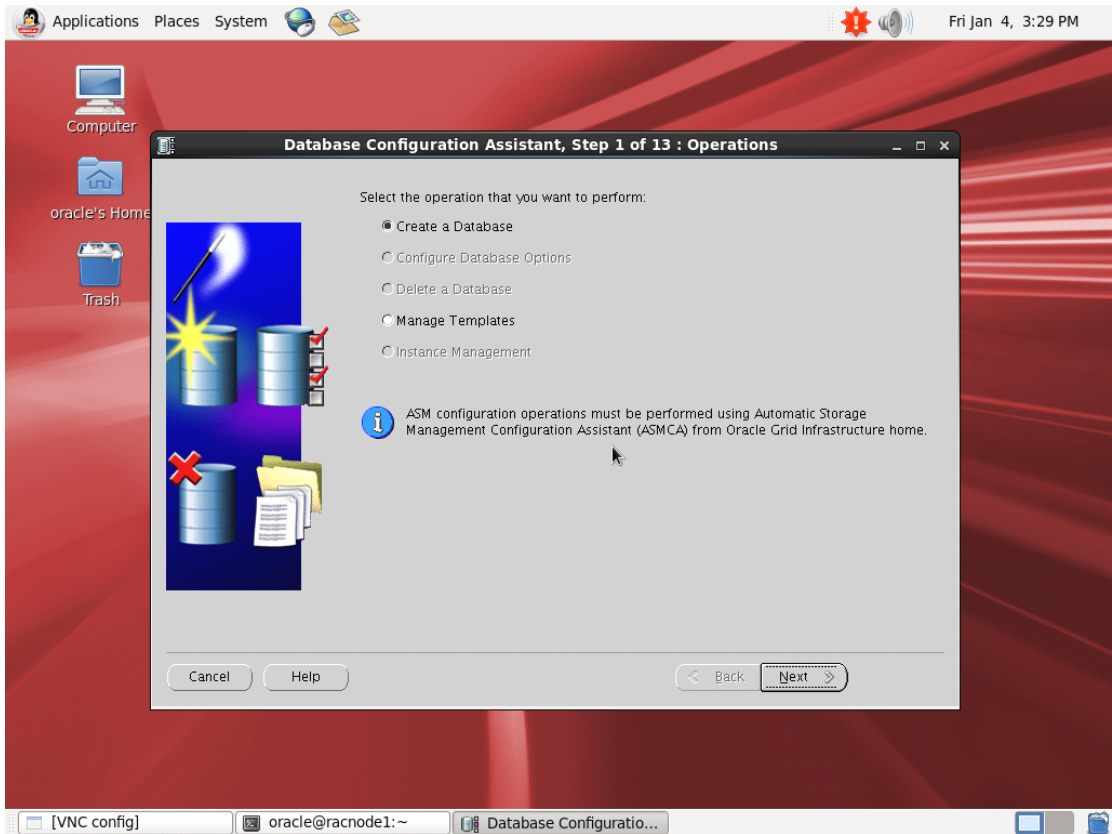
成功后点击 close，然后点击 close 结束，就是这么简单。

6、创建数据库

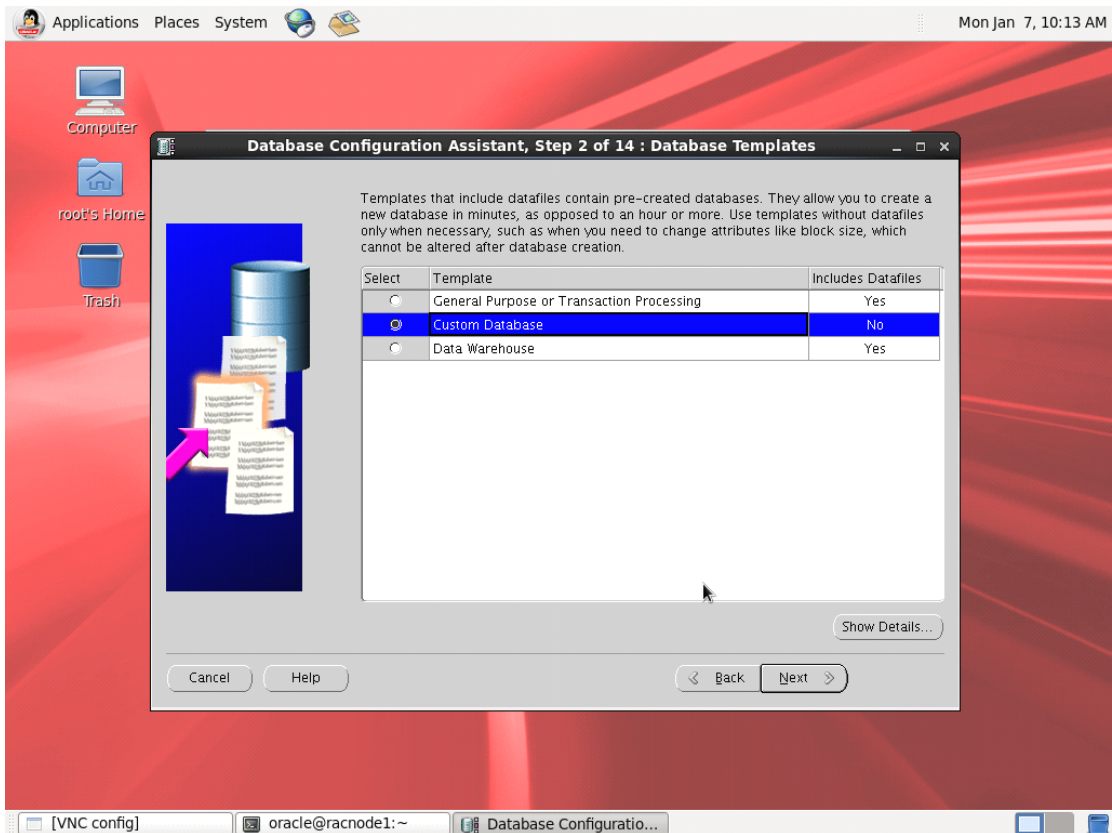
执行 dbca, 进入到安装界面, 这里当然是选择创建 RAC 数据库:



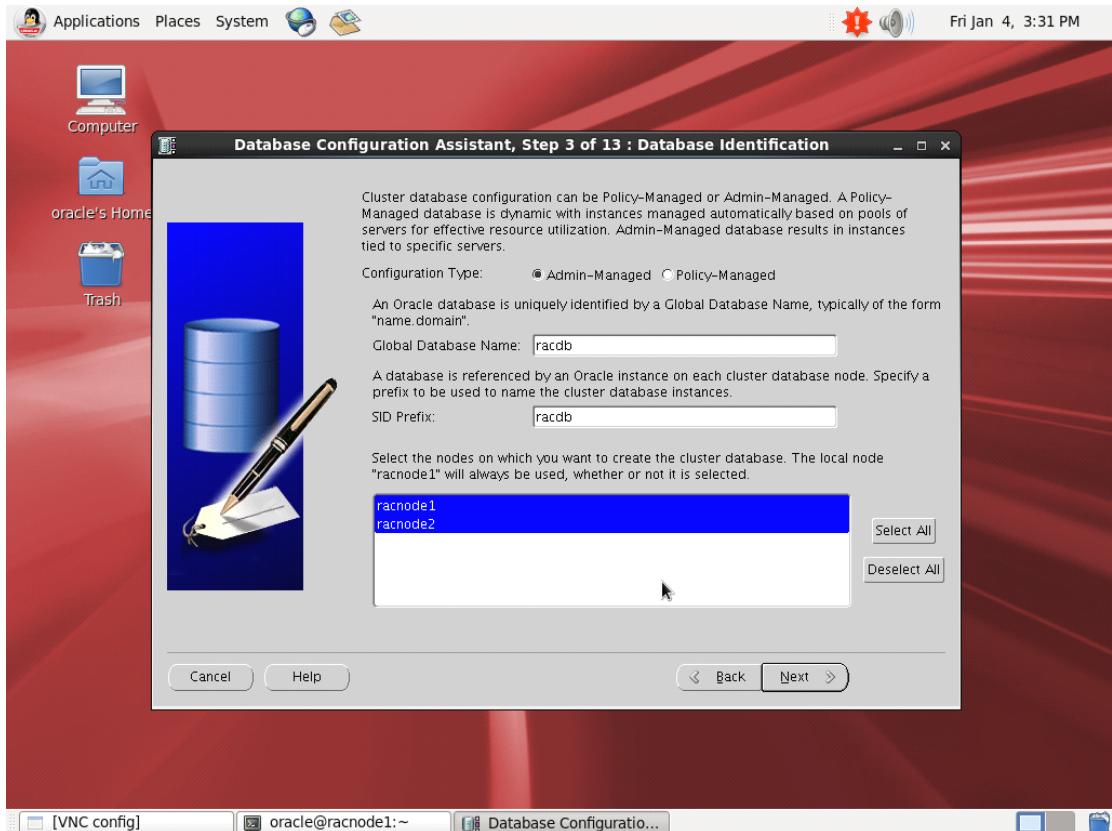
创建数据库:



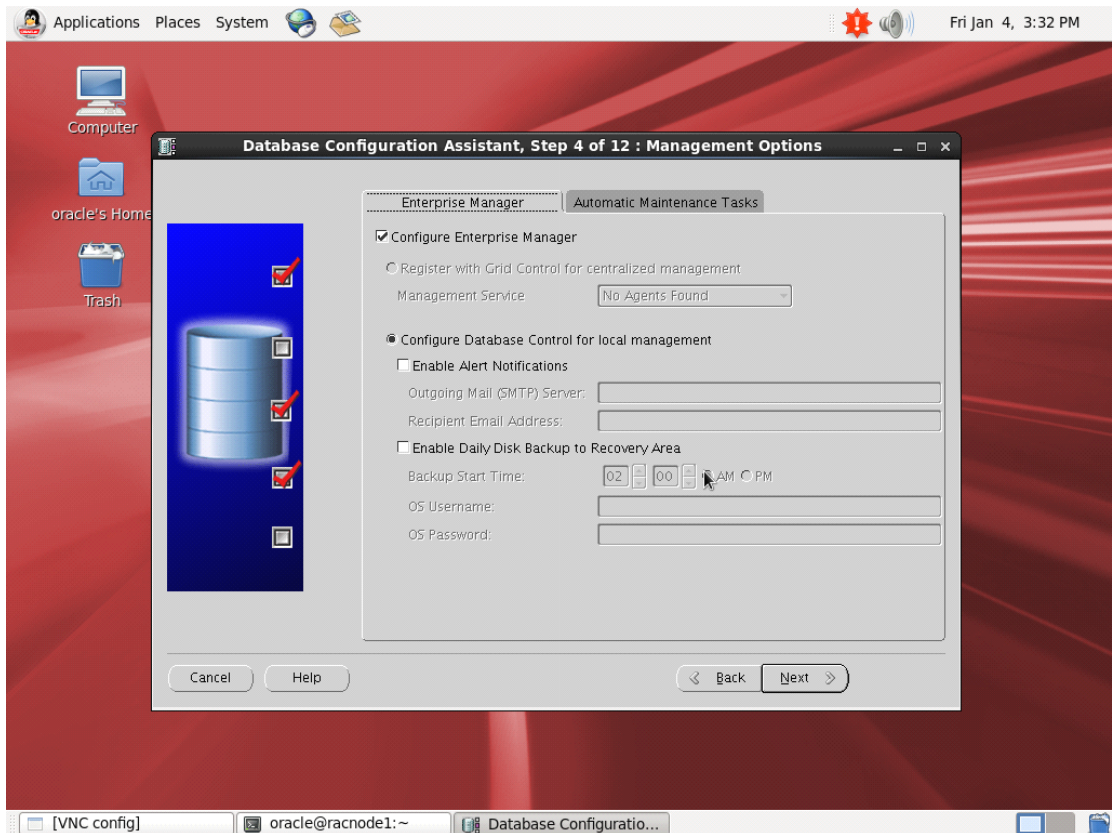
选择自定义数据库：



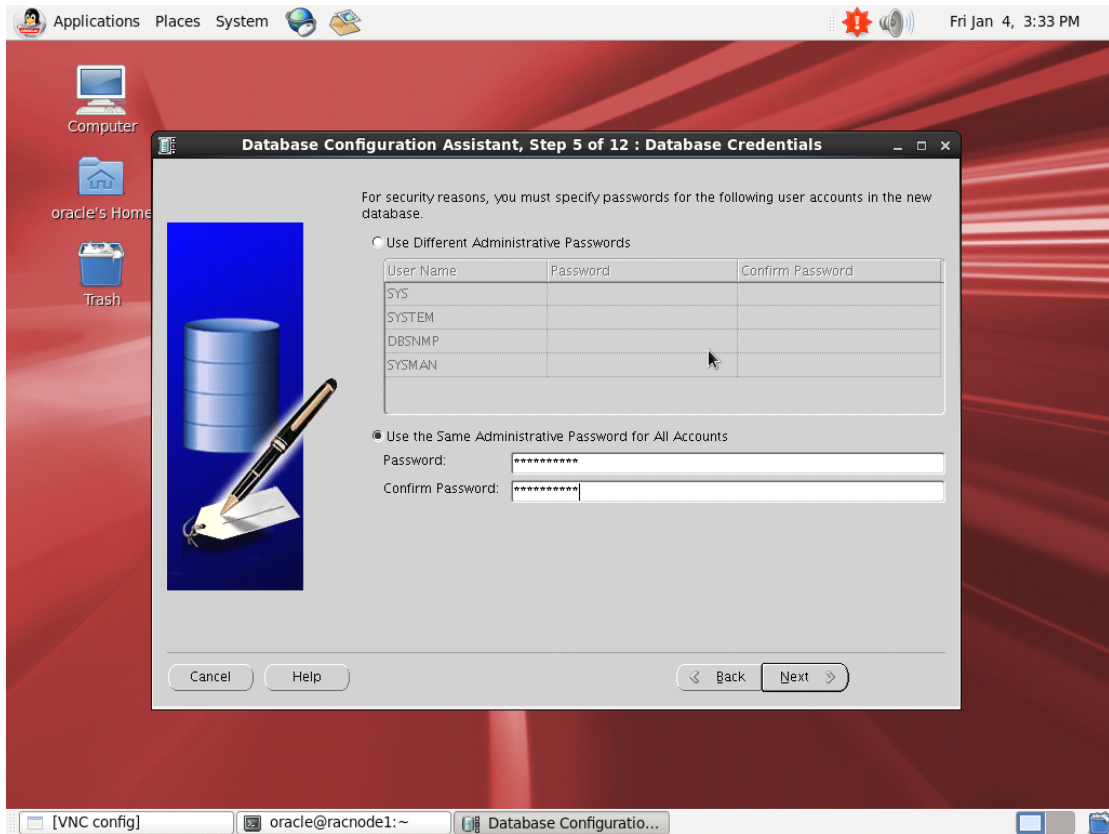
指定数据库名和 sid 的前缀，注意一定要选中所有的节点哟：



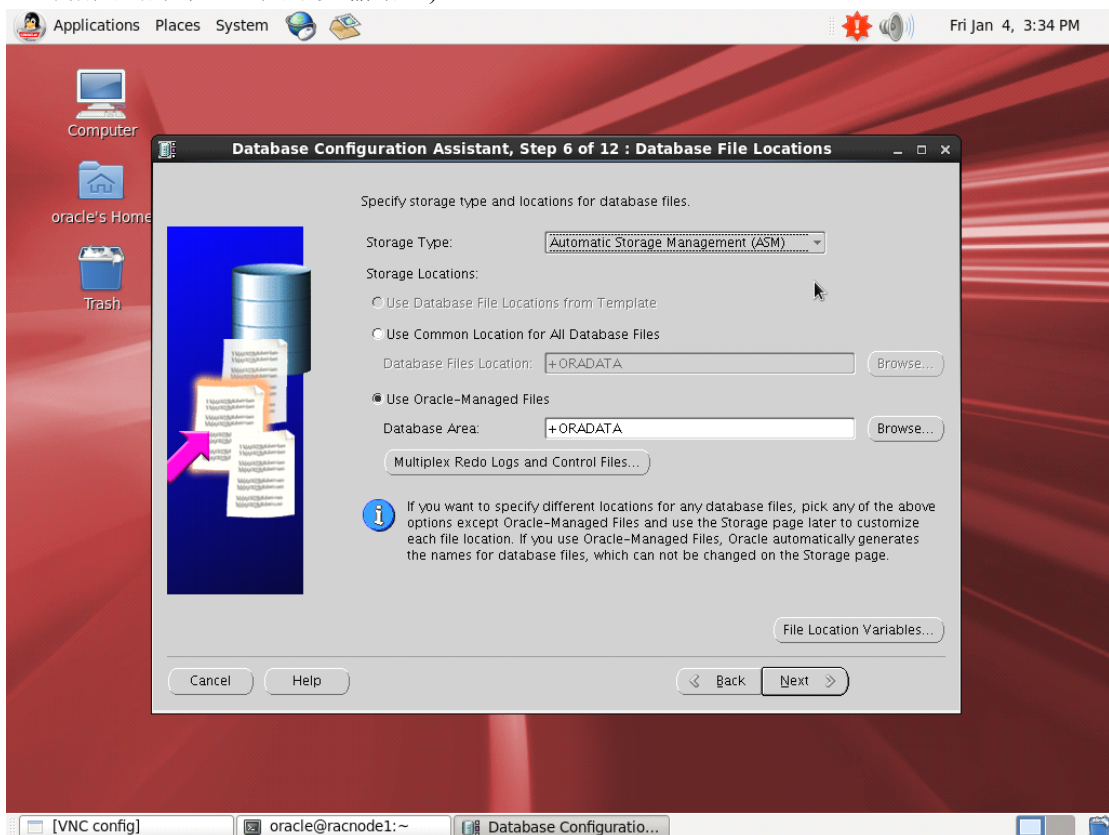
选择是否启用 EM 和自动维护任务:



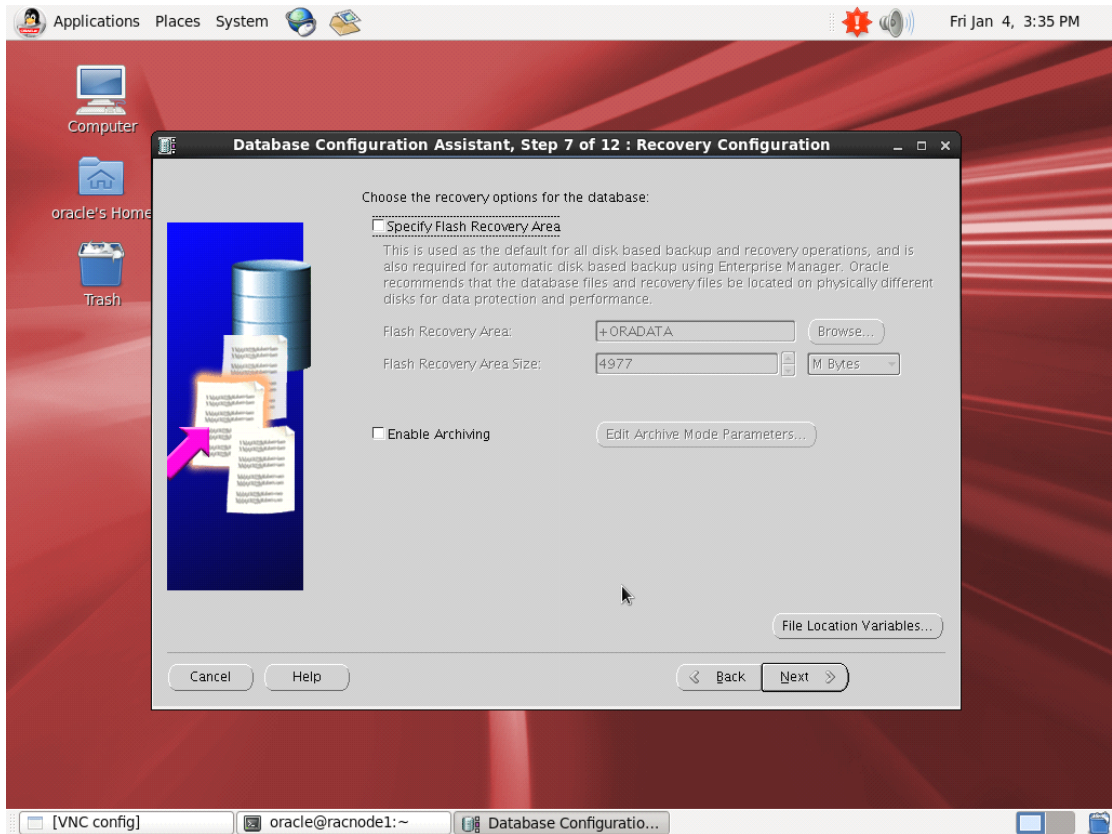
为管理员帐户指定密码,我这里给四个管理员全部指定了相同的口令,这个习惯很不好,大家不要学我:



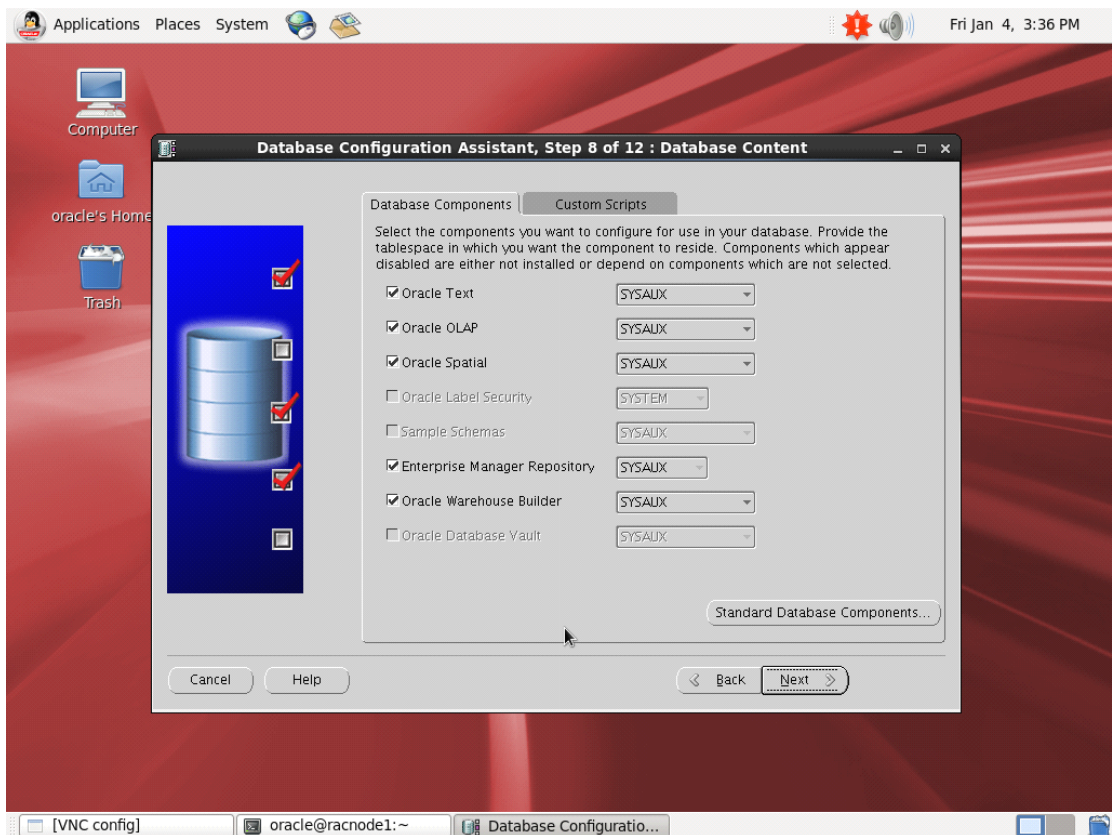
选择存储类型和存储路径，ASM吧(然后它会提示你输出 sysasm 的口令，就是前面配置 grid 时指定的那个，一定不要输错哟):



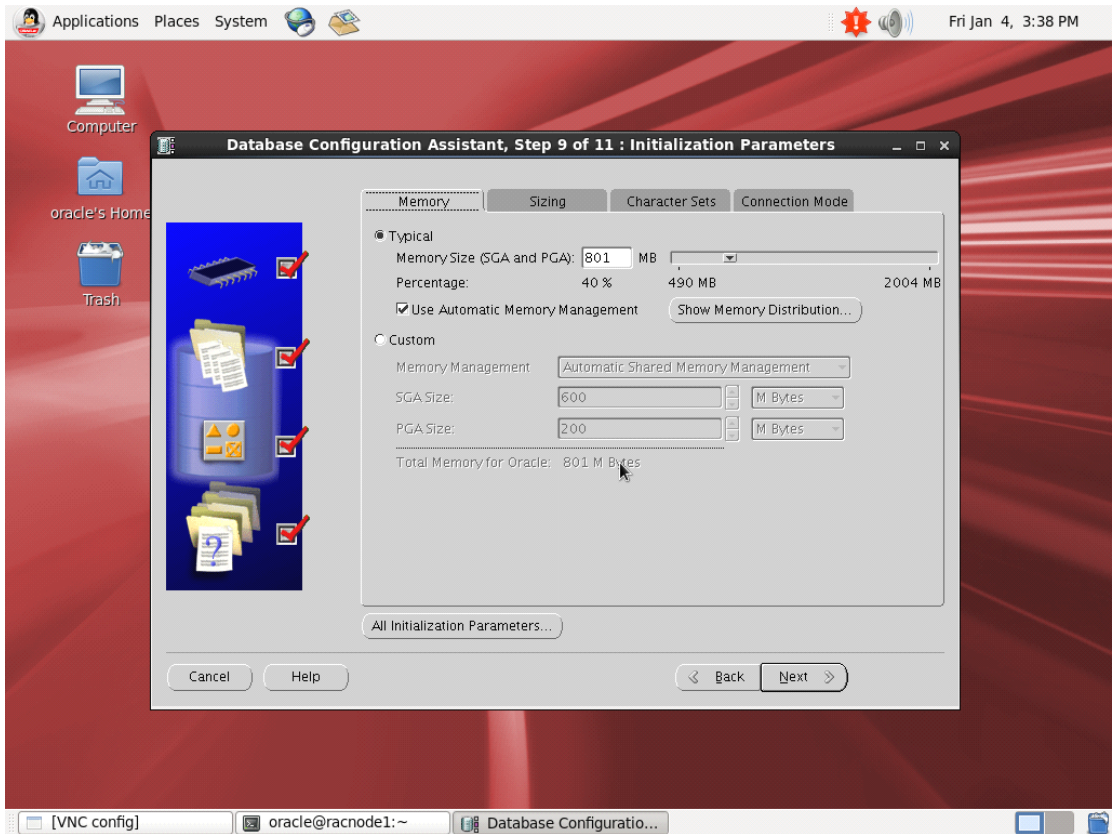
指定是否归档模式和 FRA，测试环境没必要，这里全不选，正式环境的话，起码归档是必须的：



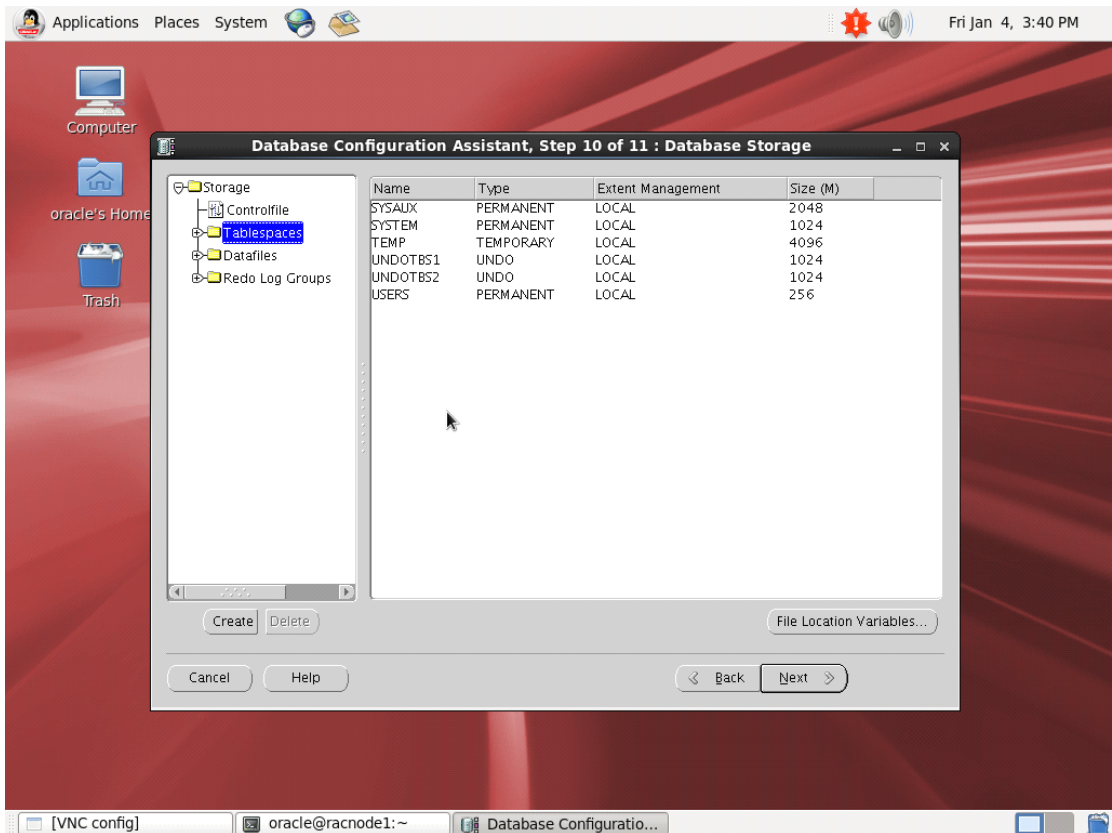
选择要安装的组件:



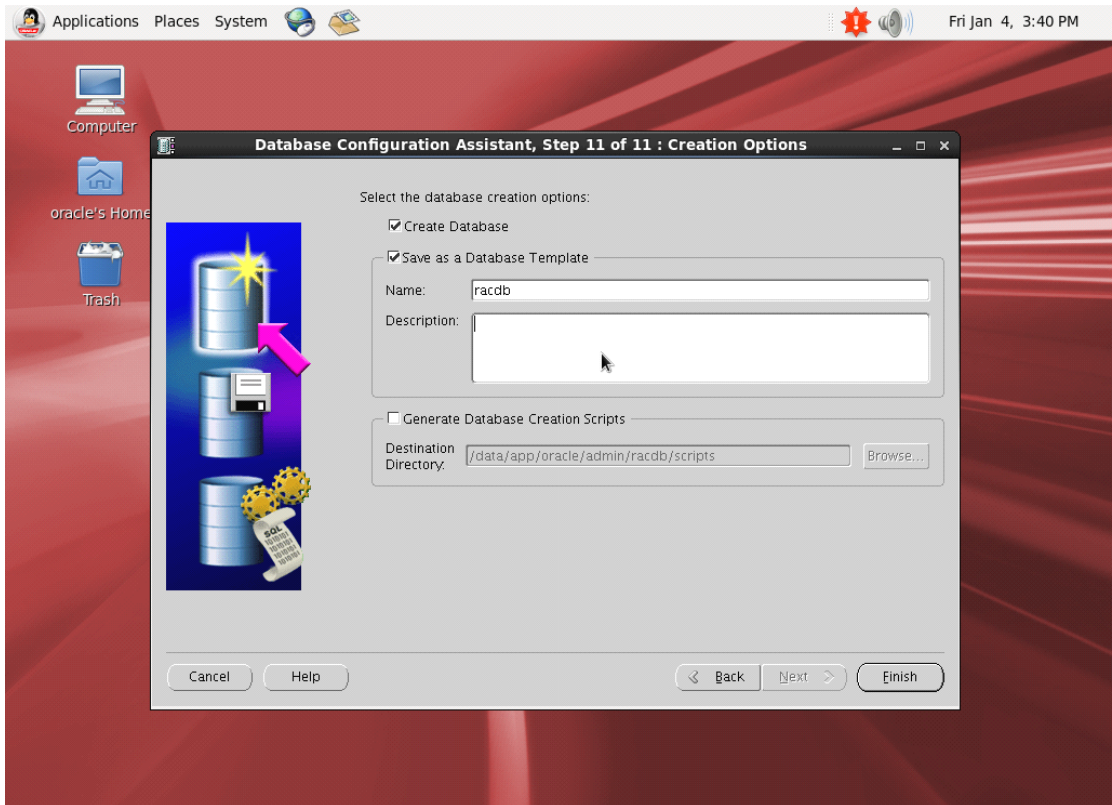
指定数据库的参数，比如内存，块大小，字符集，连接类型等等，跟单实例没有区别，不一一说了：



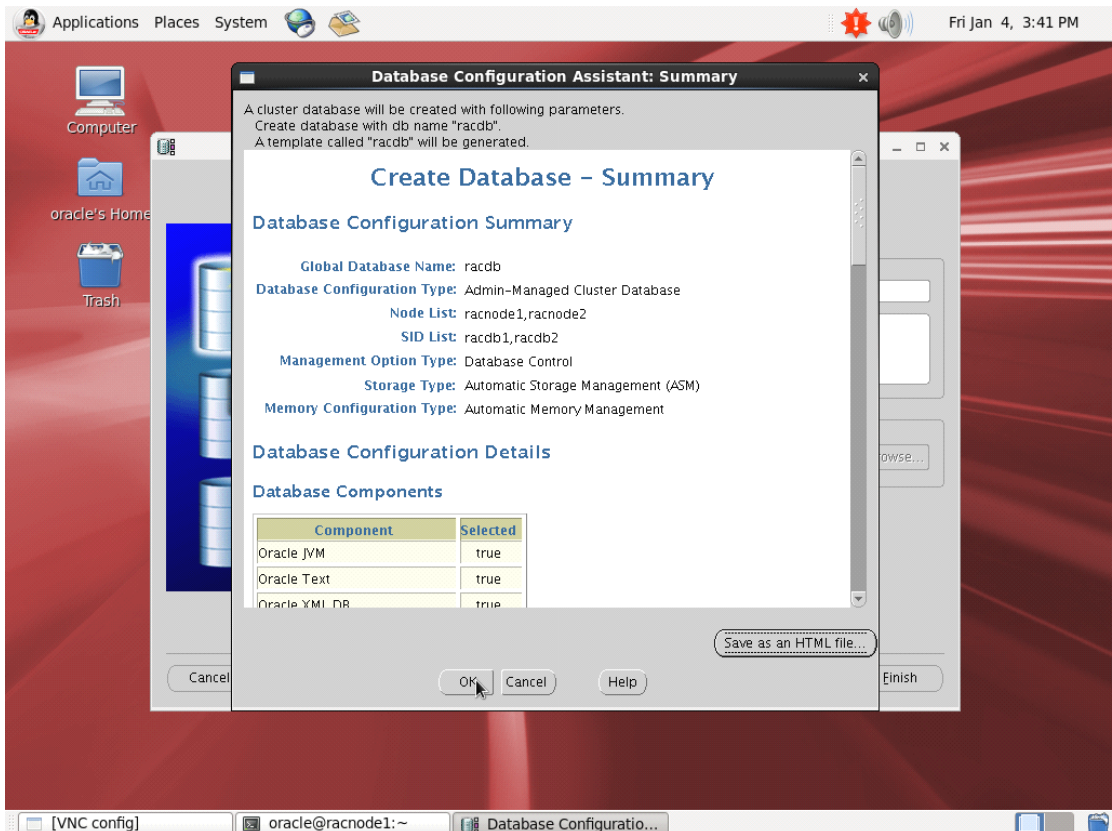
配置数据文件，控制文件等：



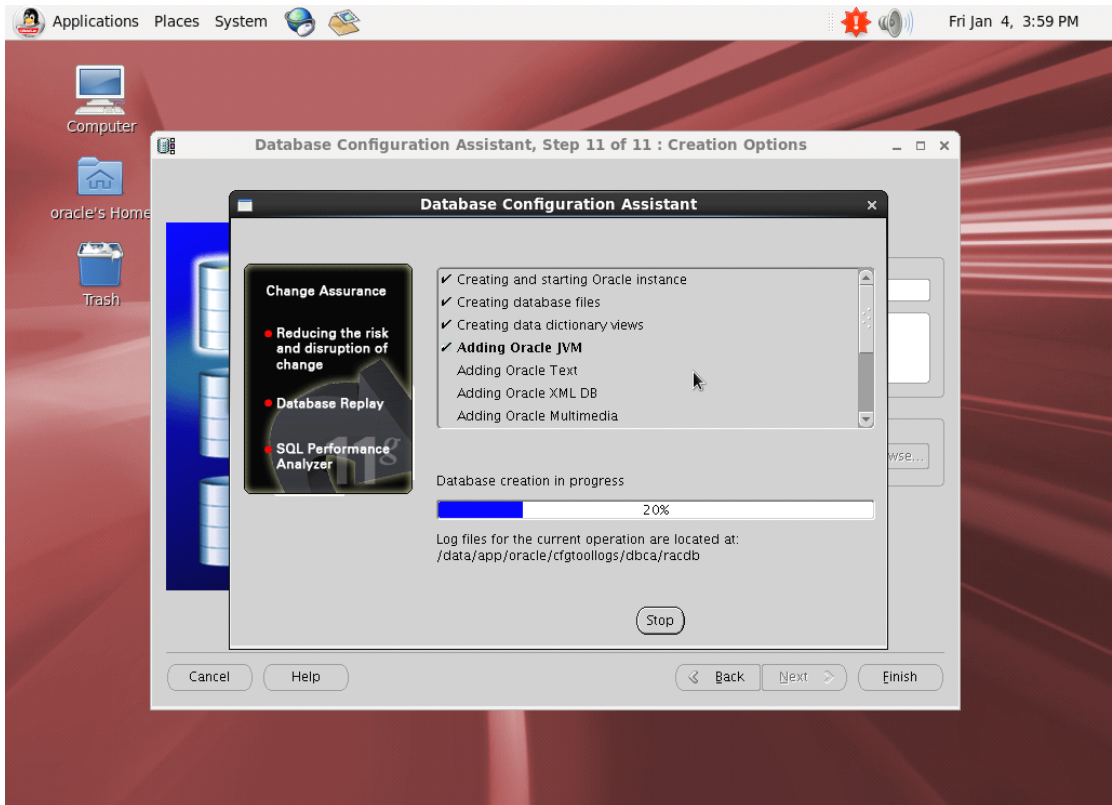
准备开始创建吧：



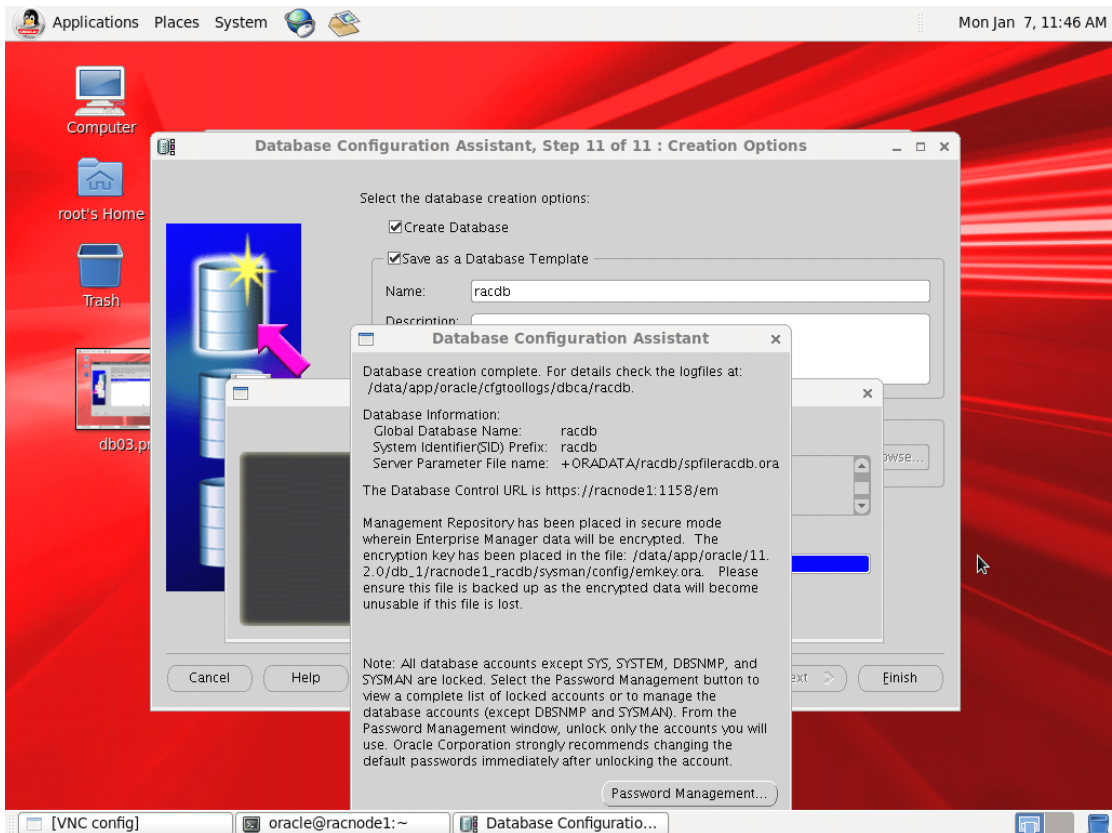
点击 finished 会弹出预览信息，如无问题，点击 ok 开始建库吧：



开始了，这一步要花些时间的哟：



等文件都复制完后，会提示数据库创建成功，点击 OK 就行了。



检查各服务是否正常：

```

[oracle@racnode1 ~]$ 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

[oracle@racnode1 ~]$ crs_stat -t
Name                Type                Target    State    Host
-----
ora....ER.lsnr ora....er.type ONLINE    ONLINE  racnode1
ora....N1.lsnr ora....er.type ONLINE    ONLINE  racnode1
ora.ORADATA.dg ora....up.type ONLINE    ONLINE  racnode1
ora.asm          ora.asm.type ONLINE    ONLINE  racnode1
ora.eons         ora.eons.type ONLINE    ONLINE  racnode1
ora.gsd         ora.gsd.type OFFLINE   OFFLINE
ora....network ora....rk.type ONLINE    ONLINE  racnode1
ora.oc4j        ora.oc4j.type OFFLINE   OFFLINE
ora.ons         ora.ons.type ONLINE    ONLINE  racnode1
ora.racdb.db    ora....se.type ONLINE    ONLINE  racnode1
ora....SM1.asm application ONLINE    ONLINE  racnode1
ora....E1.lsnr application ONLINE    ONLINE  racnode1
ora....de1.gsd application OFFLINE   OFFLINE
ora....de1.ons application ONLINE    ONLINE  racnode1
ora....de1.vip ora....t1.type ONLINE    ONLINE  racnode1
ora....SM2.asm application ONLINE    ONLINE  racnode2
ora....E2.lsnr application ONLINE    ONLINE  racnode2
ora....de2.gsd application OFFLINE   OFFLINE
ora....de2.ons application ONLINE    ONLINE  racnode2
ora....de2.vip ora....t1.type ONLINE    ONLINE  racnode2
ora.scan1.vip  ora....ip.type ONLINE    ONLINE  racnode1

```

一套双节点的 11gR2 RAC 就装好了，至于增删节点，咱有 OUI，操作自然不在话下，朋友们有兴趣可以自己研究一下。