# Oracle的安装

## 创建用户和组

|  |
| --- |
| [root@jimmy ~]# groupadd oinstall  [root@jimmy ~]# groupadd dba  [root@jimmy ~]# id oracle  id: oracle: No such user  [root@jimmy ~]# useradd oracle -g oinstall -G dba  [root@jimmy ~]# id oracle  id: oracle: No such user  [root@jimmy ~]# useradd oracle -g oinstall -G dba |

## Identifying Required Software Directories

|  |
| --- |
| [root@jimmy ~]# mkdir /u01/app/oracle -p  [root@jimmy ~]# **chown** oracle.oinstall /u01 -R |

是chown，不是chmod

## 创建oracle使用到的环境变量

整个安装过程是使用新创建的oracle用户来安装的，所以先用oracle用户登录系统，或者在SSH客户端中新建一个session

修改.bash\_profile文件

|  |
| --- |
| [oracle@jimmy ~]$ vim .bash\_profile |

在文件的最后追加以下内容

|  |
| --- |
| export PATH  export EDITOR=vim  export ORACLE\_BASE=/u01/app/oracle  export ORACLE\_HOME=$ORACLE\_BASE/product/10.2.0/db\_1  export PATH=$ORACLE\_HOME/bin:$PATH |

如果我们现在执行”./runInstaller”的话，会报出如下的错误：

|  |
| --- |
| [oracle@jimmy database]$ ./runInstaller  Starting Oracle Universal Installer...  Checking installer requirements...  Checking operating system version: must be redhat-3, SuSE-9, redhat-4, UnitedLinux-1.0, asianux-1 or asianux-2  Passed  All installer requirements met.  Preparing to launch Oracle Universal Installer from /tmp/OraInstall2013-11-10\_04-19-18PM. Please wait ...[oracle@jimmy database]$ Exception in thread "main" java.lang.UnsatisfiedLinkError: /tmp/OraInstall2013-11-10\_04-19-18PM/jre/1.4.2/lib/i386/libawt.so: libXp.so.6: cannot open shared object file: No such file or directory  at java.lang.ClassLoader$NativeLibrary.load(Native Method)  at java.lang.ClassLoader.loadLibrary0(Unknown Source)  at java.lang.ClassLoader.loadLibrary(Unknown Source)  at java.lang.Runtime.loadLibrary0(Unknown Source)  at java.lang.System.loadLibrary(Unknown Source)  at sun.security.action.LoadLibraryAction.run(Unknown Source)  at java.security.AccessController.doPrivileged(Native Method)  at sun.awt.NativeLibLoader.loadLibraries(Unknown Source)  at sun.awt.DebugHelper.<clinit>(Unknown Source)  at java.awt.Component.<clinit>(Unknown Source) |

## 安装依赖软件

要解决上面的问题我们需要安装下面的软件

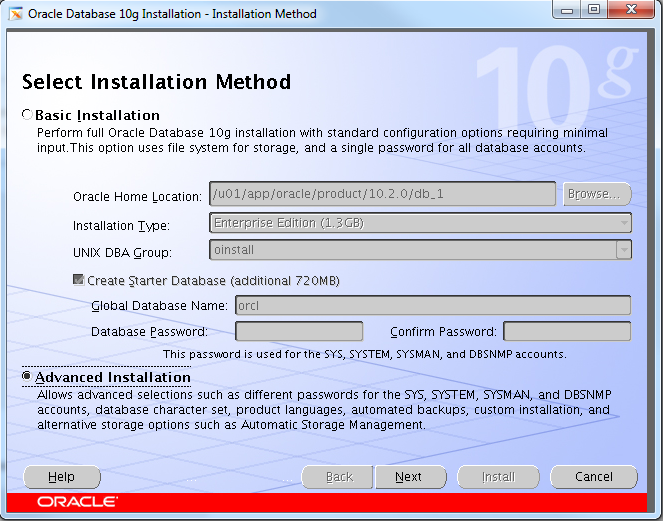
挂载Linux安装光盘后

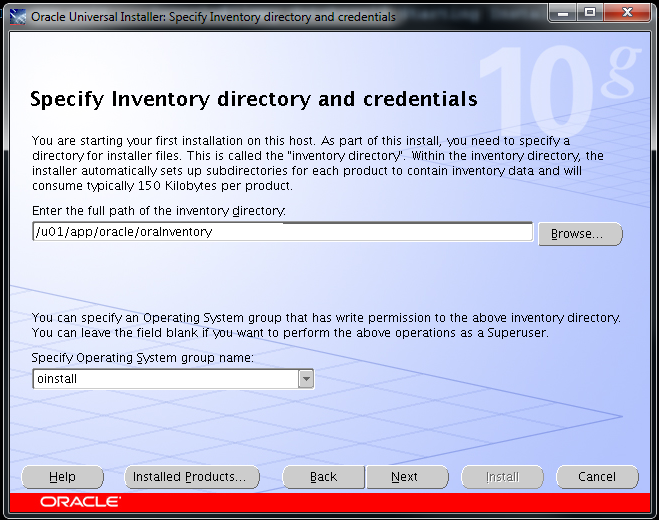
|  |
| --- |
| [root@jimmy Server]# pwd  /media/RHEL\_5.4 i386 DVD/Server  [root@jimmy Server]# ls \*libXp\*  libXp-1.0.0-8.1.el5.i386.rpm libXp-devel-1.0.0-8.1.el5.i386.rpm libXpm-3.5.5-3.i386.rpm libXpm-devel-3.5.5-3.i386.rpm  [root@jimmy Server]# **rpm -ivh** libXp-1.0.0-8.1.el5.i386.rpm  warning: libXp-1.0.0-8.1.el5.i386.rpm: Header V3 DSA signature: NOKEY, key ID 37017186  Preparing... ########################################### [100%]  1:libXp ########################################### [100%]  [root@jimmy Server]# |

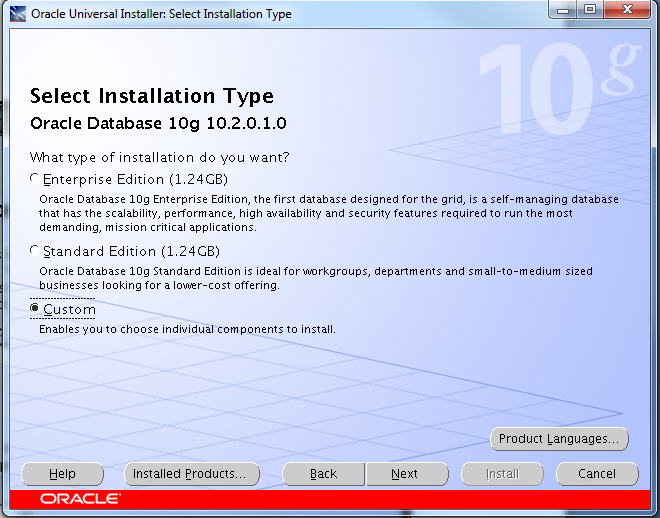
rpm的ivh参数是为了使安装时显示的信息更加的美观

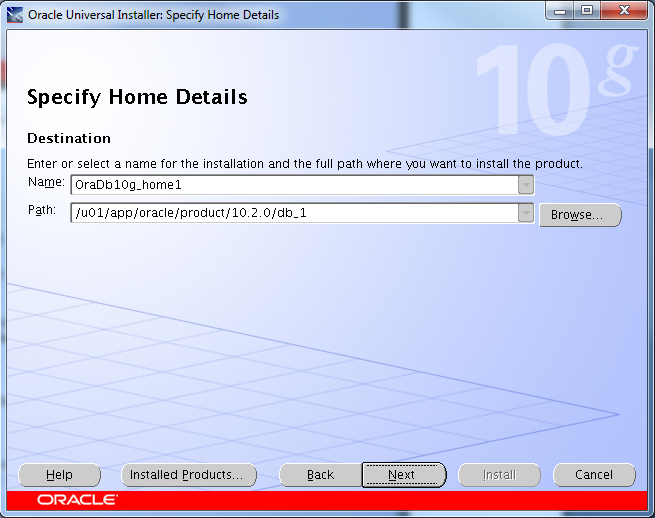
## ./runInstaller

切换到oracle用户，到解压目录执行./runInstaller

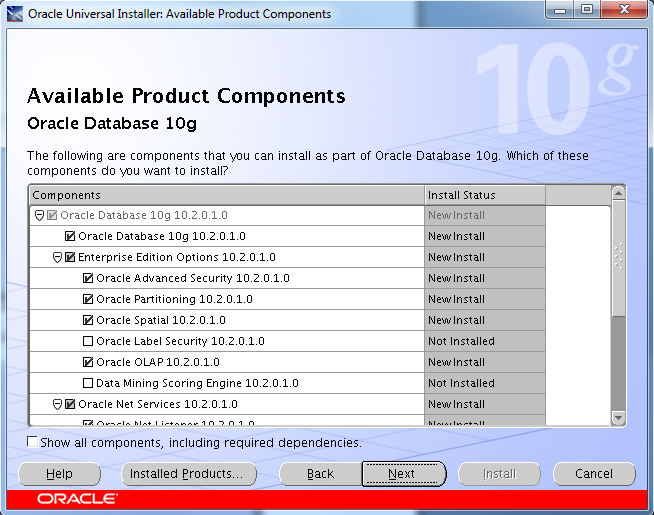




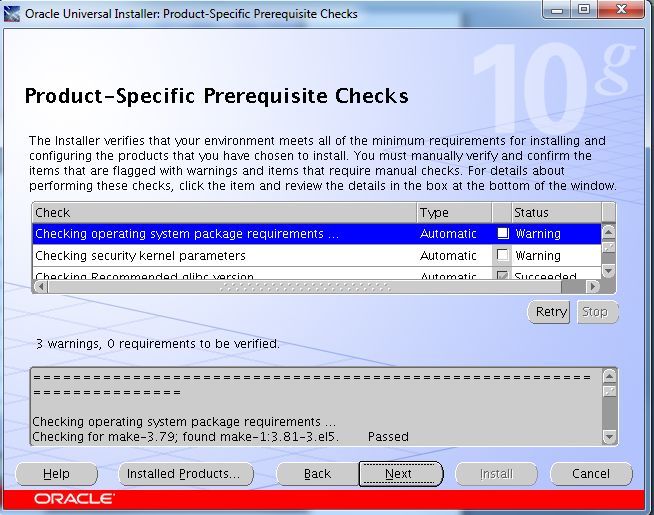




这里由于我们设置了oracle的一些环境变量，所以会自动找到我们指定的目录



直接下一步

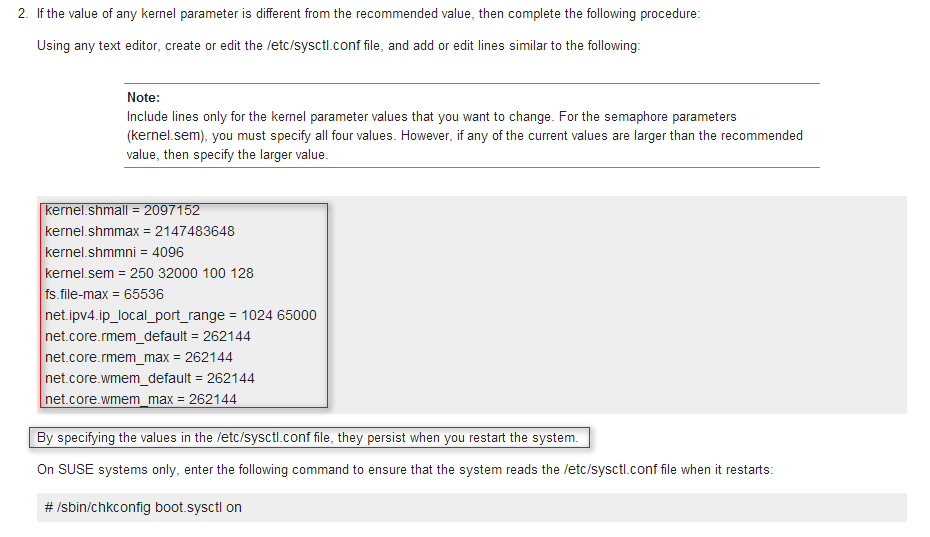


测试安装条件时有3个警告，这是由于我们当初没有进行内核参数的修改导成的，下面进行这些修改

## 使用root用户修改内核参数

参照官方安装文档修改内核参数

|  |
| --- |
| [root@jimmy Server]# vim /etc/sysctl.conf |



将上面的内容复制过去后，要将里面已经存在的重复的变量删除

就像上面的文档提示的一样，你需要重启Linux，使其配置起作用（也可以使用sysctl –p）

重启后还有两个错误，一个一个解决

### Checking for gcc-3.2; found Not found. Failed <<<<

进入安装光盘的目录，安装gcc

|  |
| --- |
| [root@jimmy Server]# pwd  /media/RHEL\_5.4 i386 DVD/Server  [root@jimmy Server]# ls gcc\*  gcc-4.1.2-46.el5.i386.rpm gcc44-c++-4.4.0-6.el5.i386.rpm gcc-c++-4.1.2-46.el5.i386.rpm gcc-gnat-4.1.2-46.el5.i386.rpm gcc-objc-4.1.2-46.el5.i386.rpm  gcc44-4.4.0-6.el5.i386.rpm gcc44-gfortran-4.4.0-6.el5.i386.rpm gcc-gfortran-4.1.2-46.el5.i386.rpm gcc-java-4.1.2-46.el5.i386.rpm gcc-objc++-4.1.2-46.el5.i386.rpm |

由于不知道这些安装文件的依赖关系，所以使用yam来安装，编辑/etc/yum.repos.d/iso.repo（其实是新建）

|  |
| --- |
| [rhel-Server]  name=5u5\_Server  baseurl=file:///media/RHEL\_5.4 i386 DVD/Server  enable=1  gpgcheck=0  gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release |

保存退出后安装：

|  |
| --- |
| [root@jimmy Server]# yum clean all  Loaded plugins: rhnplugin, security  Options Error: Error parsing 'file:///media/RHEL\_5.4 i386 DVD/Server': URL must be http, ftp, file or https not ""  [root@jimmy Server]# |

出现报错，原因是我们写的路径里有空格（注意这里面的提示信息不是很友善）

为了方便，我们将光盘里的server目录下的内容复制到一个目录中

|  |
| --- |
| cp /media/RHEL\_5.4\ i386\ DVD/Server/ -r /RHEL5U5 |

再修改下面的文件为：

|  |
| --- |
| [rhel-Server]  name=5u5\_Server  baseurl=**file:///RHEL5U5**  enable=1  gpgcheck=0  gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release |

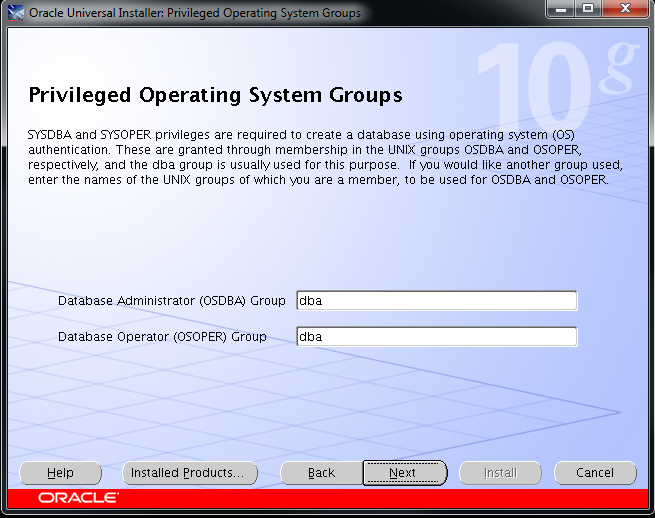
执行安装：

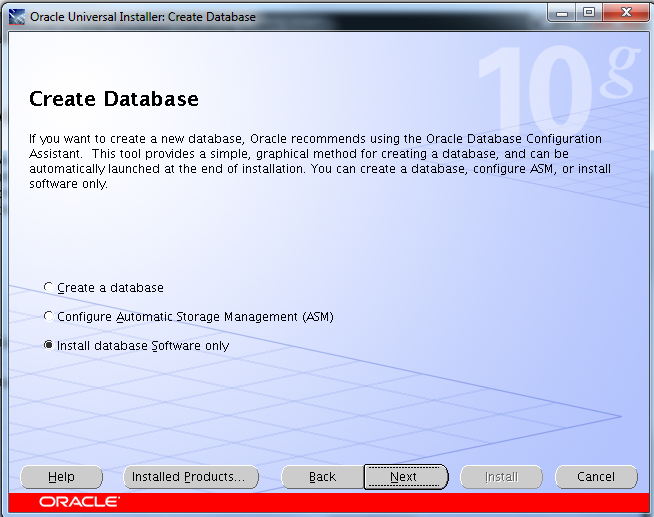
|  |
| --- |
| [root@jimmy Server]# yum clean all  Loaded plugins: rhnplugin, security  Cleaning up Everything  [root@jimmy Server]# yum install gcc |

### 内存空间与期待值不够

关闭虚拟机，将内存改为1G就可以了

所有的检查通过后，next





选择第三个，不要使用默认的。我们要手动创建数据库

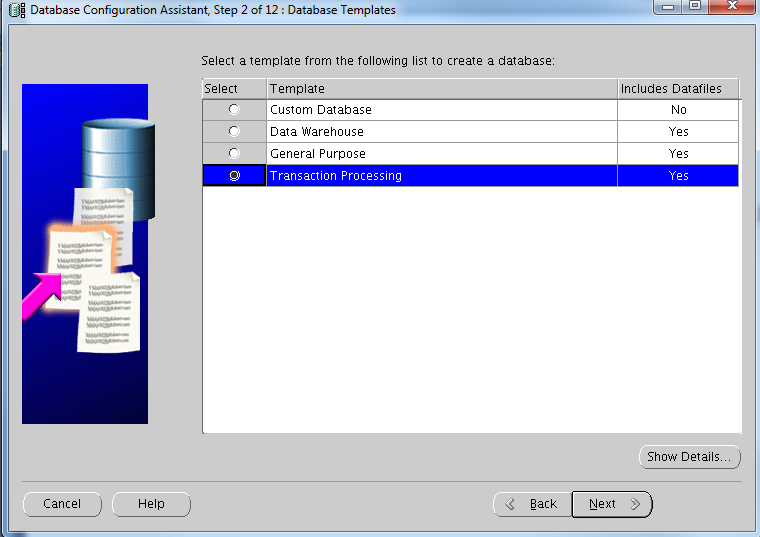
## 按提示执行脚本

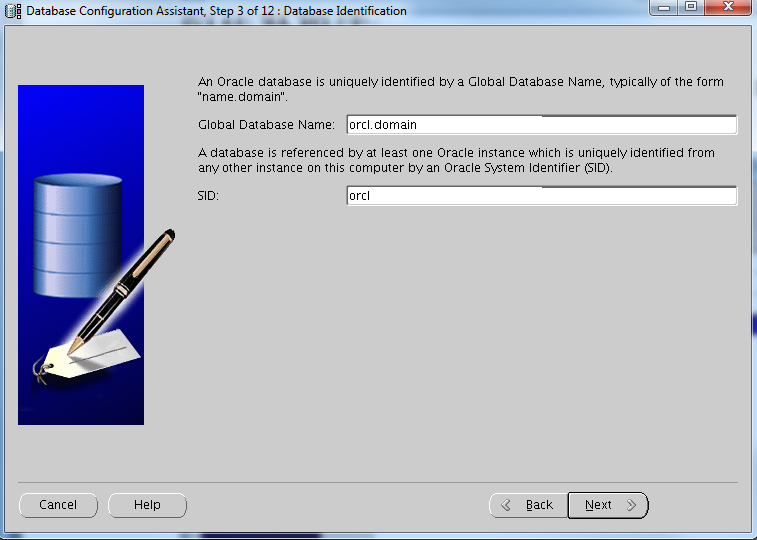
最后一个脚本会提示输入路径，使用默认值就可以了

## 创建数据库

使用oracle来登录，输入dbca（因为我们只在oracle用户的bash\_profile里引用了oracle目录下的bin目录，所以只有这个用户可以来创建数据库）

### 选择Transaction Processing





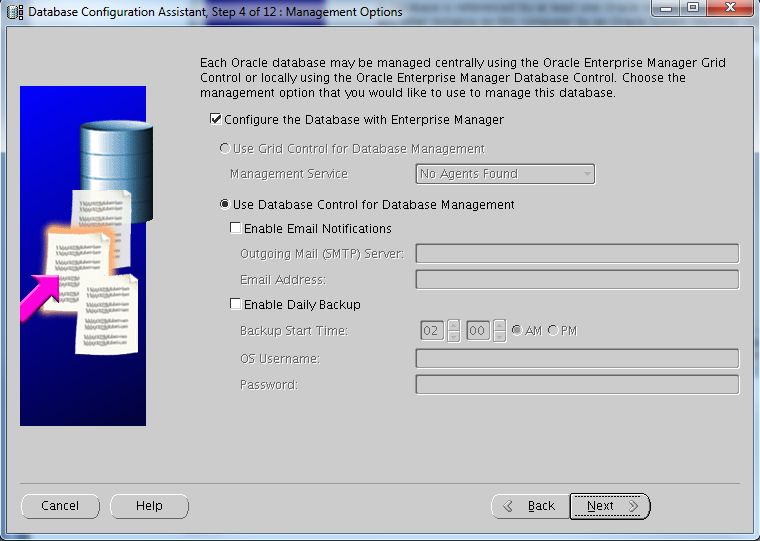
SID ==（等价于） ORACLE\_SID == instance name（三者等价）

global\_db\_name == db\_name+db\_domain

instance == SGA + 后台进程

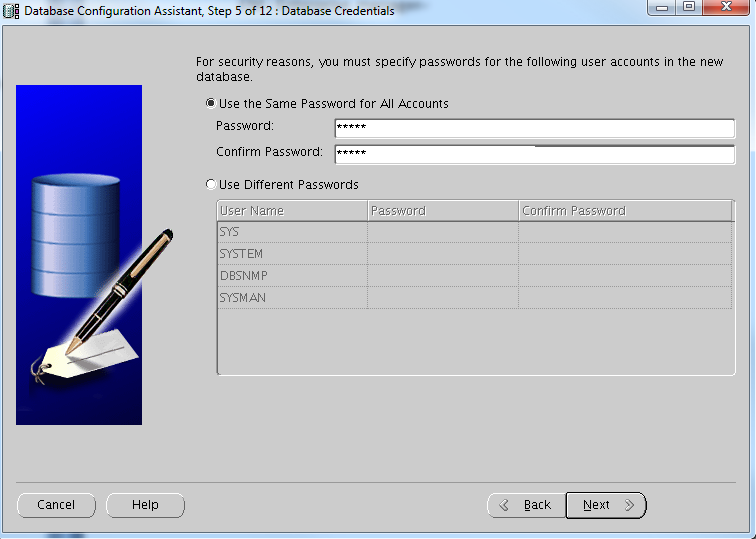
oracleServer == instance + 物理存储

### 安装EM

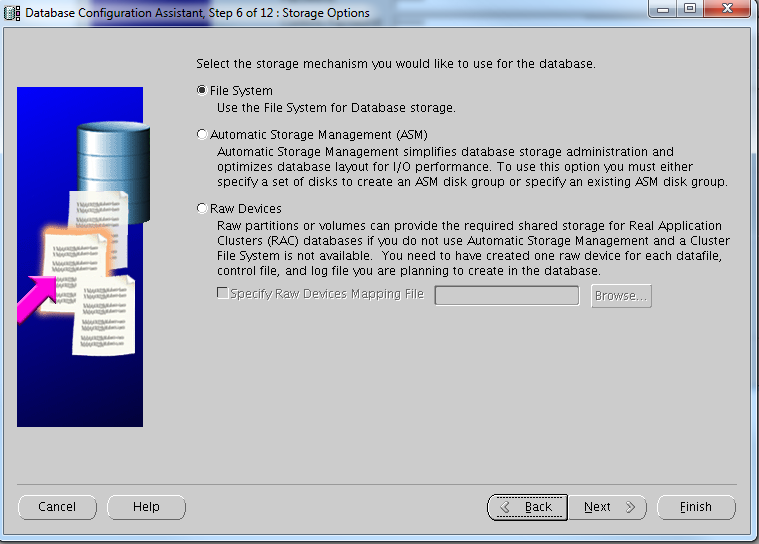


EM : Enterprise manager

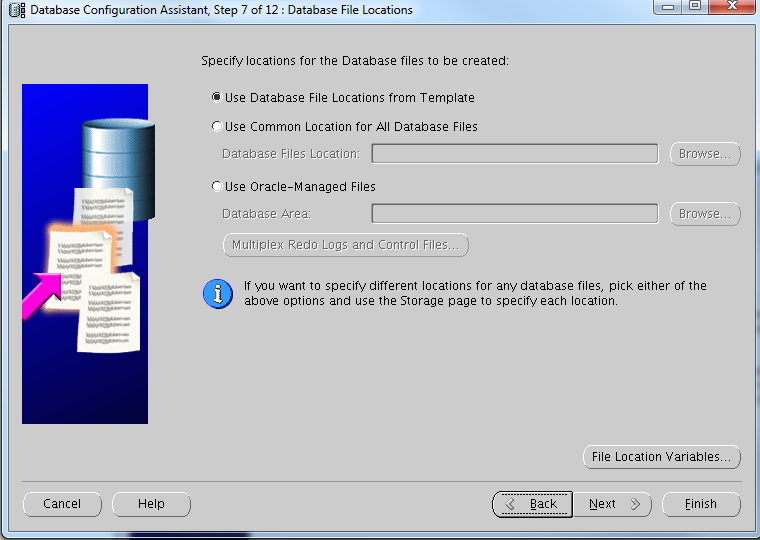
### 指定密码给所有的用户：



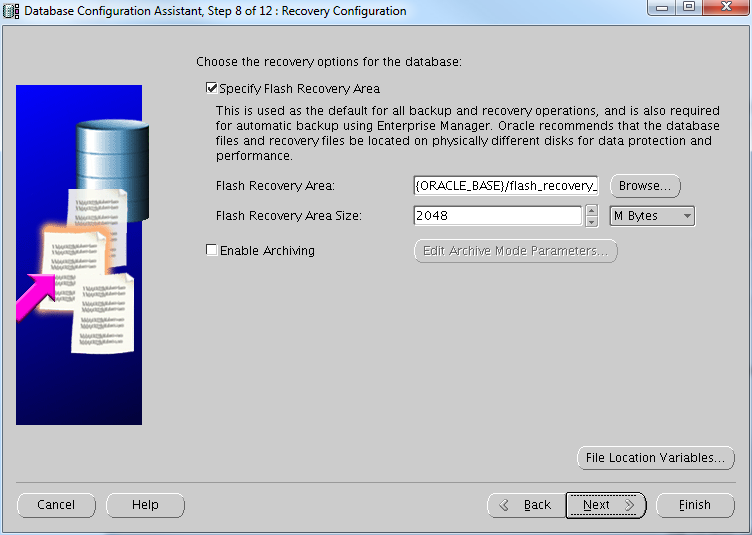
### 选择文件系统类型



### 指定日志，控制文件的位置

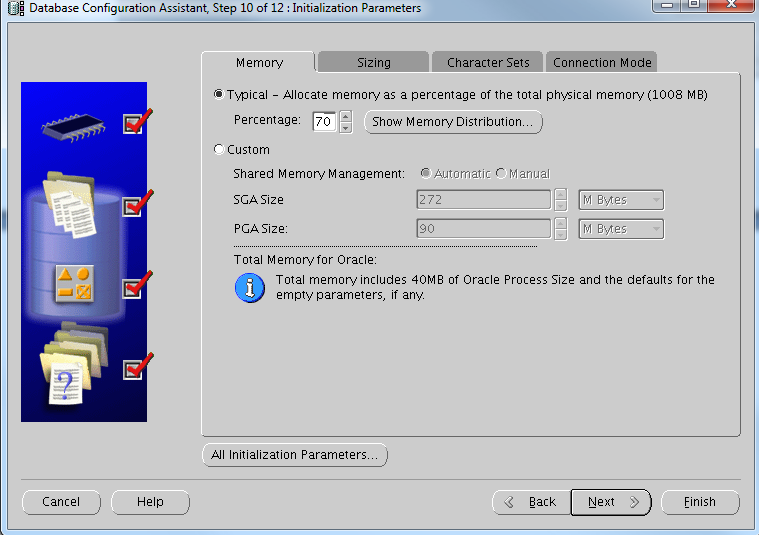


### 配置闪回区

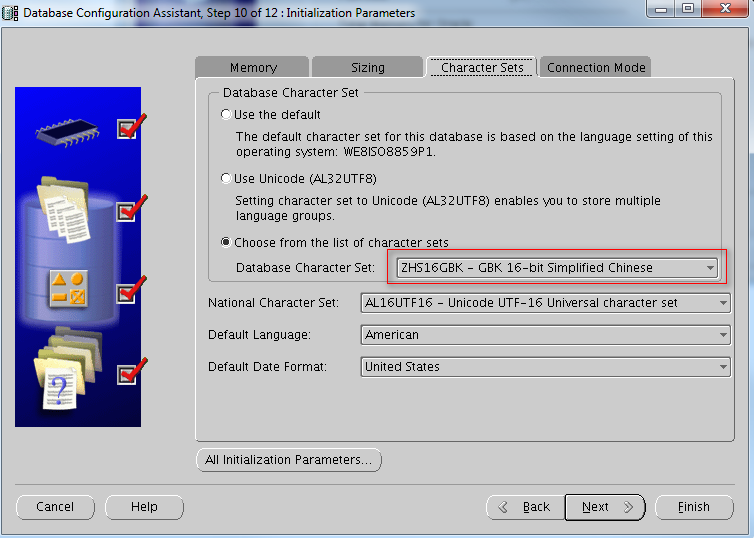


## 初始化参数

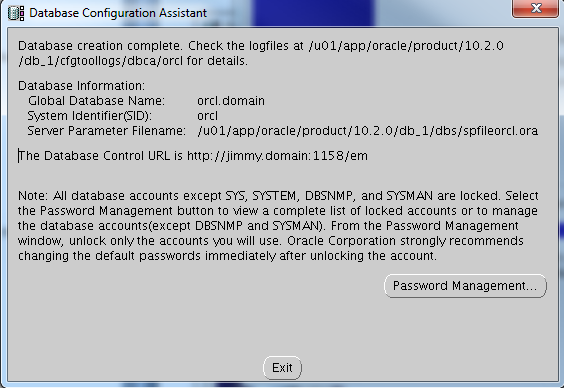
### 指定数据库占用系统的内存改为70



### 字符集—使用中文字符集

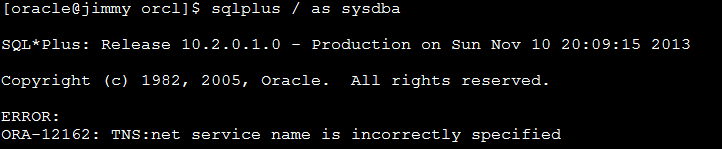


### 无须修改密码—直接OK

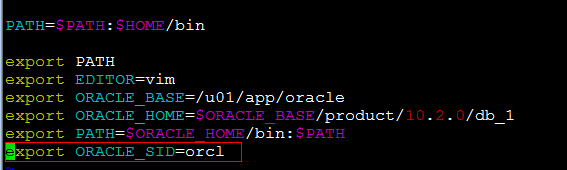


## 登陆时出现的问题

安装完成后，登陆时出现如下错误



原因是我们没有指定oracle的SID，修改~/.bash\_profile文件增加如下内容

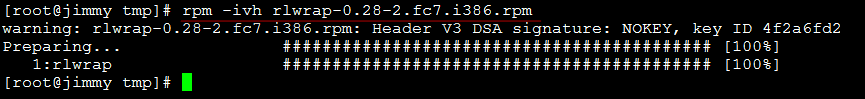


这样就可以登陆成功了。

## Sqlplus的一些问题

### 使用删除键，或者方向键时出现乱码

安装如下软件



为sqlplus起别名



### 使用vim作为sqlplus的edit全集的默认编辑器

在~/.bash\_profile中加入如下命令（貌似默认已有）



## 其它安装时常碰到的问题

### 安装时出现ora-12547 TNS: lost contact

缺少libaio-0.3.93 i386.rpm包

### Cannot restore segment prot after reloc permission denied

vim /etc/selinux/config

将SELINUX从enable改为disable，并重启

或者下载selinux-policy-target-xxxx noarch.rpm并安装它就可以了

### Areasqueries

安装路径上有中文或者其它的特殊字符

## 卸载数据库

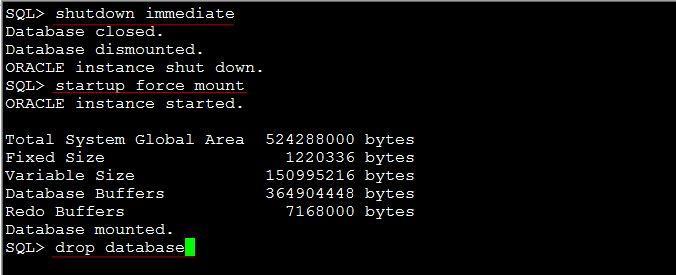
### 手工方式

#### 先登录oracle

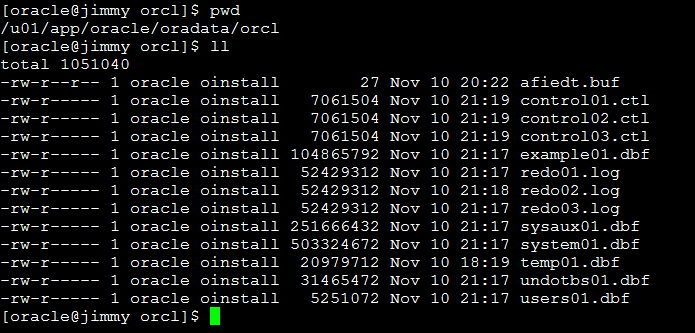
sqlplus / as sysdba

#### 将oracle置为mount状态

##### 如果oracle已经在打开状态，先关闭数据库

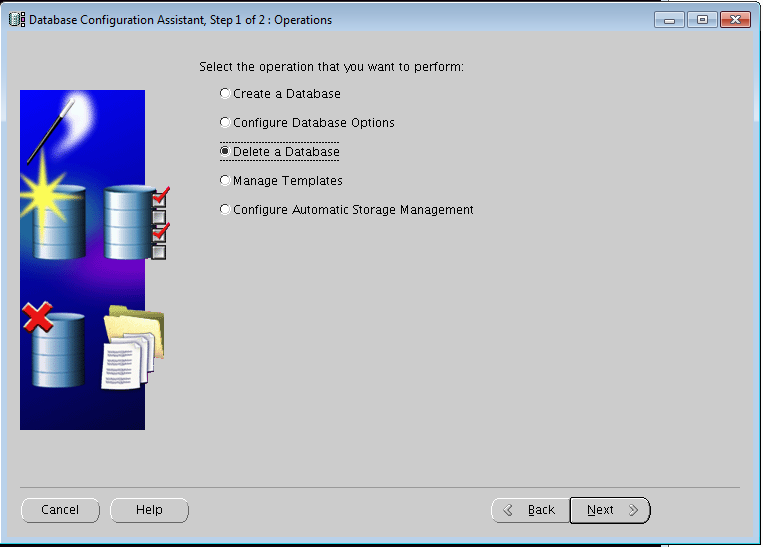


#### 删除遗留文件



手动删除上述文件

## 使用dbca删除数据库



# SQLPLUS的基本使用

## 登录

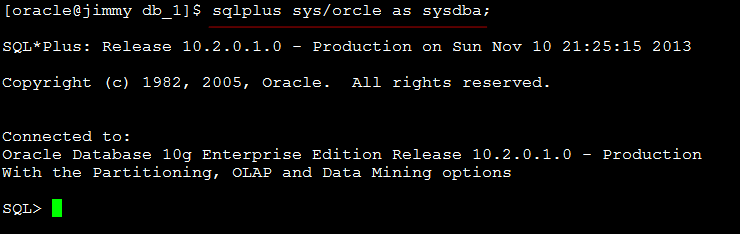
Sqlplus username/password[@connect\_id] [as sysdba]

我们一般使用如下方式登录：



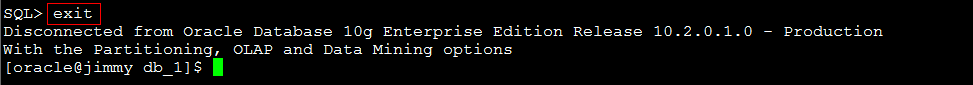
这是表示使用当前用户(oracle)来登录数据库

这个命令与下面的命令是等价的

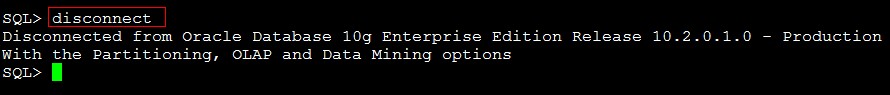


## 退出登录

### 方式1



### 方式2

第二种方式退出后的提示符依旧是sql>

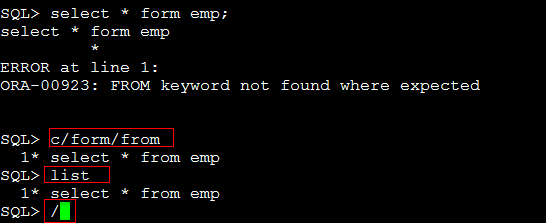
可以通过conn或者connect来重新连接

## SQLPLUS的SQL缓冲区

### 常用操作

在SQLPLUS中，输入”/”可以执行上一次执行的命令，因为这条命令是记录在SQL缓冲区中的，这个缓冲区不会存储SQLPLUS内部的命令，只会存储SQL语句

同时可以对缓冲区的数据进行修改等：



List命令只是打印缓冲区中的语句，并不执行

但最常用的还是edit命令。

输入edit命令可以调用指定的EDITOR（我们前面配了它为vim）来编辑缓冲区中的SQL

### 保留缓冲区内的内容

save ‘/tmp/test.sql’

## sqlplus的命令帮助

|  |
| --- |
| Sql> ? set |

这样可以查看set命令的帮助文档

## 远程连接ORACL实例

由于Oracle是不支持网络监听功能的，所以远程连接ORACLE是通过Listener服务建立的。Client会通过Listener来对远程数据库发起请求，Listener根据提供的IP，端口和实例名等信息发起请求得到响应后，Client就直接与远程数据库建立起了连接，此时Listener就与建立起的连接没有任何联系了

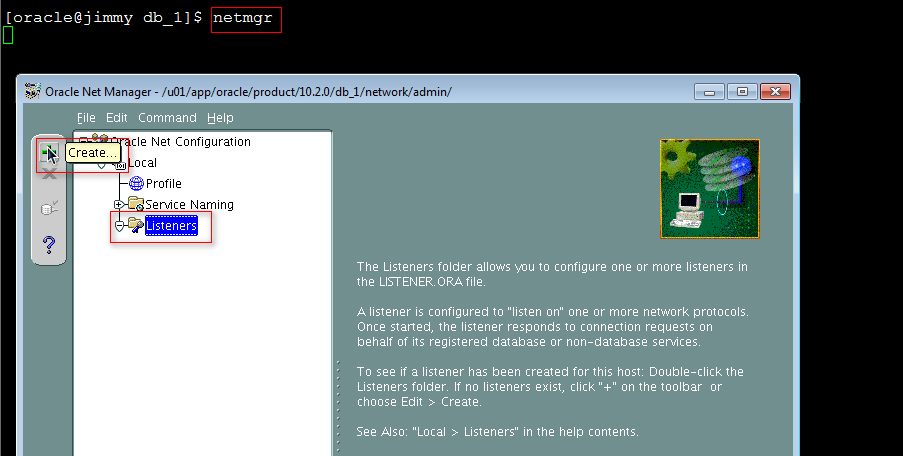
### 在服务器中建立Listener

#### 使用图形化工具

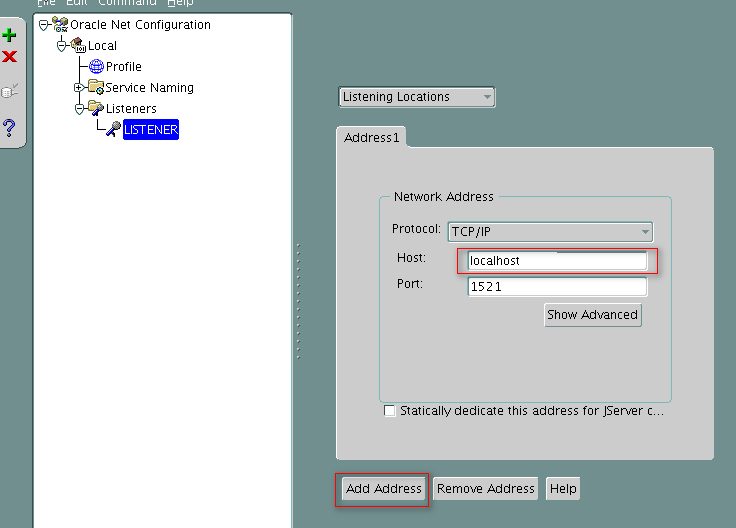
共有两种方式都可以创建

##### Netca

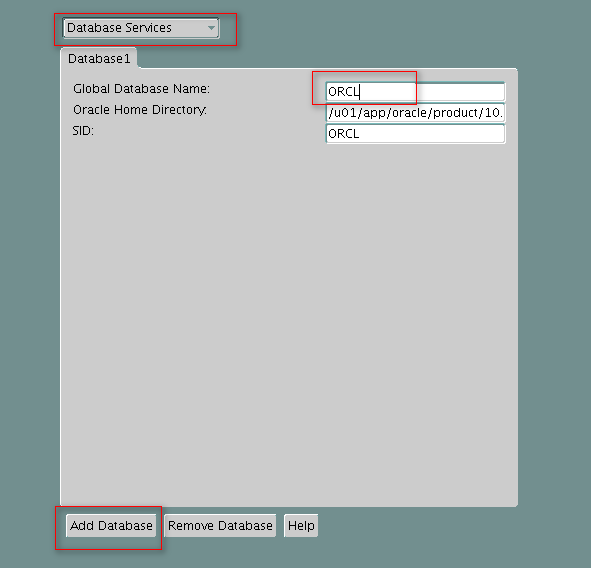
##### Netmgr



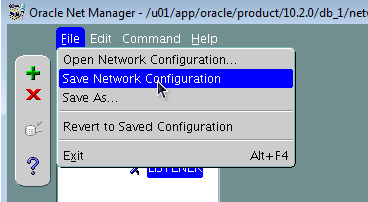
###### 添加地址：使用localhost或者直接输入IP



###### Change global database name to “ORCL”



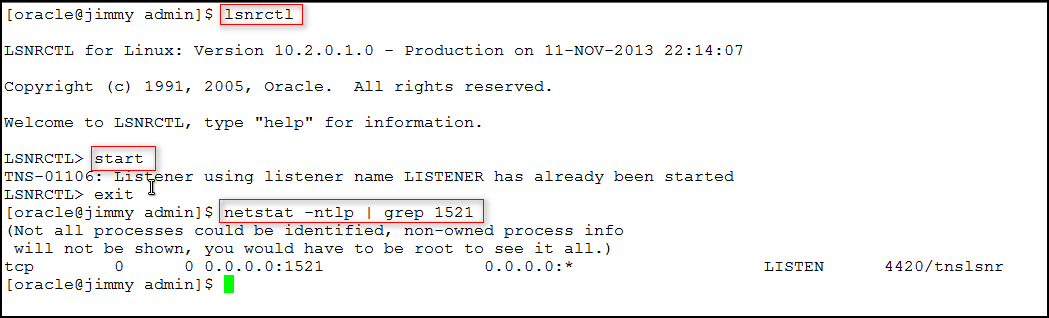
###### Save netwok configuration



##### 使用netca命令创建Listener

与上面的步骤大同小异

##### 启动Listener



使用lsnrctl命令进入Listener的管理命令窗口，这时可以使用的命令有start, stop, service和status命令

也可以直接在lsnrctl后面跟上这些命令： lsnrctl start或者lsnrctl service

同时可以使用netstat –ntlp | grep 1521来验证是否真的启动了

##### 远程连接oracle

这时就可以使用下面的格式的命令来远程连接服务器上的oracle了

Sqlplus username/password@ip:port/serviceName

如：

C:\>sqlplus [scott/tiger@192.168.1.200:1521/ORCL](mailto:scott/tiger@192.168.1.200:1521/ORCL)

上面的操作也可以进行一些简化：

1. 在本地的oracle客户端中打开Oracle Net Configuration Assistant(也就是netca)
2. 根据提示进行配置，在最后的一步提供一个配置描述符如aaa
3. 这时在本地就可以使用c:\>sqlplus scott/tiger@aaa来登录了

上面的所谓简化就是在修改本地客户端中的c:\oracle\product\10.2.0\db\_1\NETWORK\ADMIN\tnsnames.ora配置文件，其中配置描述符是不区分大小写的

##### 配置可以远程连接ORACLE的IP地址

在本地的c:\oracle\product\10.2.0\db\_1\NETWORK\ADMIN\目录下除了tnsnames.ora外还有一个叫做sqlnet.ora的文件。但这个文件一般在本地不进行任何修改，只修改服务器上对应的文件，这是用来定义oracle的网络配置的信息的，如可以定义是否进行IP的checking，那些IP可以访问，那些不可以等

在Linux上的目录为：/u01/app/oracle/product/10.2.0/db\_1/network/admin/

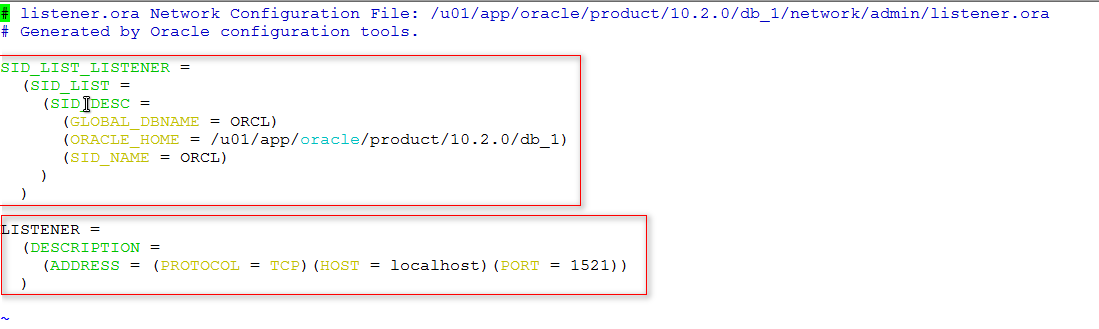
这个sqlnet.ora的文件可能不存在，这时只需要新建它就行了，无须理会。

|  |
| --- |
| tcp.validnode\_checking=yes  tcp.invited\_nodes=(ip1,ip2,ip3...)  tcp.exclude\_nodes=(ip1,ip2,ip3...) |

有点像Linux中的/etc/hosts.deny和/etc/host.allow文件

###### 小结

其实上面的配置只是为了在/u01/app/oracle/product/10.2.0/db\_1/network/admin/listener.ora文件中生成上面的配置信息



最上面那部分暂时不用管，将它删除也是可以使用的

第二部分才是定义出来的Listener信息

# 技巧与花絮

## 解锁用户

|  |
| --- |
| Alter user scott **account** unlock; |

不要忘了这个account