

创建 Oracle10g ASM 数据库

Step by Step

Author: Kamus

Mail: kamus@itpub.net

Date: 2004 年 7 月

概述	2
硬件配置	2
软件配置	2
安装 VMware	2
在 VMware 中安装 RHEL3	3
安装 Oracle ASMLib	7
安装 Oracle10g	9
创建 ASM 实例	9
创建 ASM 数据库	11
启动 ASM 数据库的步骤	13
ASM 数据库的简单测试	13
总结	17

概述

在真正的海量数据库环境中，DBA 可能会花费很多的时间来作磁盘管理，比如一个表空间将占满整个磁盘，DBA 就需要再添加一块磁盘到操作系统中，然后再在新的磁盘上创建新的数据文件，如果是单个磁盘这倒不是很繁琐，问题是如果原先我们使用的是 RAID 或者说是 LVM，那么现在大量的数据仍然是分布在以前的那些磁盘上，如果我们想让这些数据均匀地分布在以前的磁盘和新增加的磁盘上，我们可能就要耗费一天甚至几天的时间来作原先数据的导出导入。那么如果有一种方法，能实现我们就把一块磁盘加到系统里，然后告诉 Oracle 我们要用这块盘了，剩下的工作全部由 Oracle 来完成，该是多好的一件事情！幸运的是，Oracle10g 已经提供了这个功能，这就是 ASM (Automatic Storage Management)。我们称为“自动存储管理”。Oracle10g 的 ASM 不但帮助 DBA 从繁琐的磁盘空间管理中解脱出来，而且更值得关注的是 ASM 同时提供了条带和镜像的功能，而这些功能原先需要通过单独地配置 RAID 来实现。

本文通过 vmware 安装 RHEL3 + Oracle10g，来体验 ASM 数据库的强大功能，为什么要使用 vmware？因为在 vmware 中我们可以随意地添加或者删除虚拟磁盘 (Virtual Disk)，这样可以更方便地测试 ASM 数据库，我想不是谁都可以顺手抓出五六块硬盘来作这样的测试的☺

硬件配置

IBM R50 5GC

迅驰 1.5G，1G 内存，40G 硬盘

软件配置

WindowsXP Pro SP1，VMware GSX Server 2.5.1，RHEL3，Oracle10g for Linux

安装 VMware

这一步就不多说了，我安装的是 VMware GSX Server，没有测试过 VMware Workstation，所以不保证可以实现本文中的步骤。

在 VMware 中安装 RHEL3

1. 安装 RHEL3

这个过程也不多说了，跟实际环境中安装 RHEL3 一样。值得注意的是，由于通过 VMware 所以可能在安装时无法正常启动 Xwindow，那么在字符界面下安装就可以了。安装完毕重新启动系统，Xwindow 会自动启动的。记住安装所有的开发包即可，这将会尽量避免以后可能出现的 link 错误。我的测试环境中给了 RHEL3 总共 384M 的内存。

2. 添加 4 块虚拟磁盘

按照如下操作，在虚拟机的属性中选择添加硬件，然后添加一个 Virtual Disk



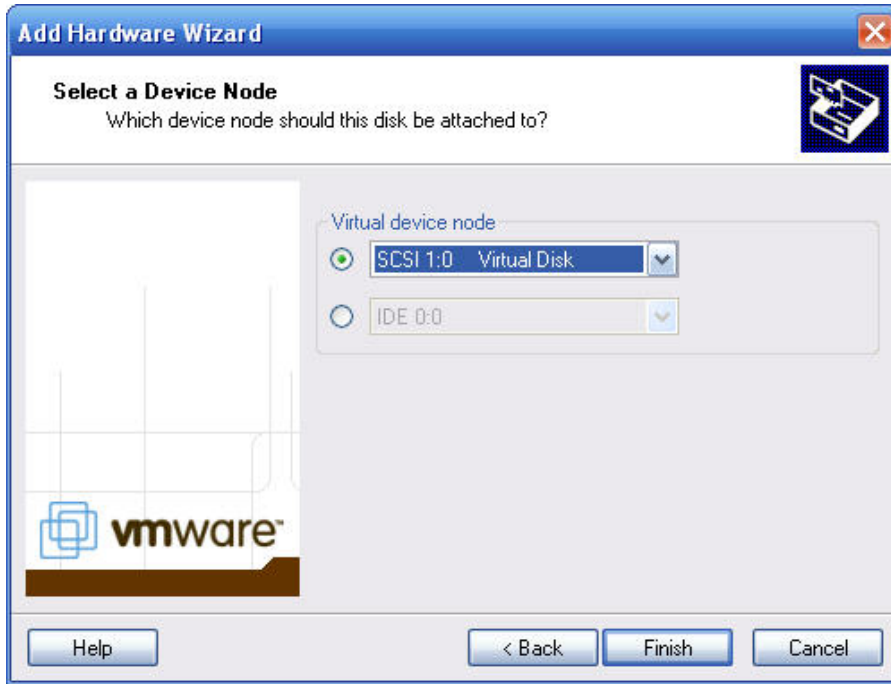
下一步，由于我们创建的虚拟磁盘只是用来放置 Oracle 的数据文件，而且我们将使用 ASM 中的 failgroup 的条带功能，所以设置最大 size 是 1G 就足够了。



下一步，指定虚拟磁盘文件的名称和存放位置，默认存放在虚拟机的相同目录下。这里我们指定文件名为 `physical1`



点击 `Advanced` 按钮，设置虚拟磁盘的物理挂载点，通常我们选择和虚拟机已有的系统盘在不同的 `SCSI` 端口上，比如我们虚拟机的系统盘是 `SCSI0:0`，那么就设置这个磁盘为 `SCSI1:0`。即将添加的另外三块磁盘依次为 `SCSI1:1`、`SCSI1:2` 和 `SCSI1:3`



由于 Virtual Disk 的特点，在新创建这个磁盘文件的时候，并不会用到最大的 size，所以即使整个分区的空闲容量没有 4G（我们要创建 4 个最大 size 是 1G 的虚拟磁盘），也不会有问题，虚拟磁盘文件的大小是随着需要而持续增加的，最大增加到指定的 max size。当然这种特性也就导致了使用 Virtual Disk 比使用 Plain Disk（创建时即占用 max size 大小的磁盘空间）的 I/O 效率要低，所以如果有足够的磁盘空间，那么建议使用 Plain Disk，创建 Plain Disk 的方法是在上面的第二步时，在“Allocate all disk space now”的复选框前面打勾，这样将会创建一个以 pln 和 dat 为扩展名的虚拟磁盘文件。Plain Disk 的好处还在于可以模拟 RAC 环境中的共享磁盘，这是 Virtual Disk 无法作到的。模拟 RAC 不在本文叙述的范围内。如果有兴趣，可以到 www.itpub.net 的数据库管理版本精华区查找相应的发帖。

3. 更新内核

```
>uname -a
```

RHEL3 默认安装的 linux 内核版本是 2.4.21-4，为了能够正确地安装 oracleasm lib，我们需要将内核升级到 2.4.21-15.EL，否则将无法正常使用 oracleasm。

原因是我们安装的 oracleasm lib 是已经编译好的 rpm 包，而我们当前操作系统的内核必须和编译时候的系统内核版本一样，否则即使看上去安装正常，但是在启动 oracleasm 时候会在 /var/log/messages 文件中发现类似如下的报错信息：

```
kernel-module version mismatch
```

```
^I/lib/modules/2.4.21-EL-ABI/oracleasm/oracleasm.o was compiled for kernel version 2.4.21-15.EL ^Iwhile this kernel is version 2.4.21-4.EL.
```

从这个报错信息中我们也可以知道应该安装的内核版本，为了更确切地知道下载的 oracleasm 包需要哪个版本的内核，我们也可以先下载 oracleasm，安装，启动然后查看这个报错信息，之后再升级内核到相应的版本。

注意：本文中提到的 2.4.21-15.EL 内核是本文编写时 `oracleasm` 编译使用的内核，随着 `oracle` 不断地推出 `asmlib` 的新版本，需要的内核版本可能也在不断变化。

下载新内核：

kernel-2.4.21-15.EL.i686:

<http://public.planetmirror.com/pub/pmstuff/patches/kernel-2.4.21-15.EL.i686.rpm>

安装新内核：

```
# rpm -ivh --nodeps kernel-2.4.21-15.EL.i686.rpm
```

安装完毕以后，修改 `/boot/grub/grub.conf`，rpm 安装完毕以后，应该已经自动在 `grub.conf` 文件中添加了新版 kernel 启动的段落，我们需要将 `default=1` 修改为 `default=0`，这样保证每次启动都用新的内核启动，当测试没有问题之后，我们也可以直接删除 `grub.conf` 文件中的 `title Red Hat Enterprise Linux AS (2.4.21-4.EL)` 部分，防止因错误操作而使用了原有的系统内核启动系统。

```
# cat /boot/grub/grub.conf
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You do not have a /boot partition.  This means that
#           all kernel and initrd paths are relative to /, eg.
#           root (hd0,0)
#           kernel /boot/vmlinuz-version ro root=/dev/sda1
#           initrd /boot/initrd-version.img
#boot=/dev/sda
default=0
timeout=10
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
title Red Hat Enterprise Linux AS (2.4.21-15.EL)
    root (hd0,0)
    kernel /boot/vmlinuz-2.4.21-15.EL ro root=LABEL=/
    initrd /boot/initrd-2.4.21-15.EL.img
title Red Hat Enterprise Linux AS (2.4.21-4.EL)
    root (hd0,0)
    kernel /boot/vmlinuz-2.4.21-4.EL ro root=LABEL=/
    initrd /boot/initrd-2.4.21-4.EL.img
```

修改完毕以后，重新启动系统，保证使用了新内核

```
# uname -a
```

```
Linux LinuxNode1 2.4.21-15.EL #1 Thu Apr 22 00:27:41 EDT 2004 i686 i686
i386 GNU/Linux
```

安装 Oracle ASMLib

下载 ASMLib 需要的三个文件:

asmlib:<http://otn.oracle.com/software/tech/linux/asmlib/files/RPMS/rhel3/x86/oracleasmlib-1.0.0-1.i386.rpm>

asm-support:<http://otn.oracle.com/software/tech/linux/asmlib/files/RPMS/rhel3/x86/oracleasm-support-1.0.2-1.i386.rpm>

asm-kernel:<http://otn.oracle.com/software/tech/linux/asmlib/files/RPMS/rhel3/x86/oracleasm-2.4.21-EL-1.0.0-1.i686.rpm>

如果是多处理器的版本，那么可以下载下面这个文件替换上面的第三个文件:

asm-smp-kernel:<http://otn.oracle.com/software/tech/linux/asmlib/files/RPMS/rhel3/x86/oracleasm-2.4.21-EL-smp-1.0.0-1.i686.rpm>

安装:

```
$su -
# rpm -Uvh oracleasm-support-1.0.2-1.i386.rpm \
    oracleasm-lib-1.0.0-1.i386.rpm \
    oracleasm-2.4.21-EL-1.0.0-1.i686.rpm
```

配置 ASMLib:

```
# /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 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
Fix permissions of Oracle ASM disks on boot (y/n) [y]: y
Writing Oracle ASM library driver configuration          [ OK ]
Creating /dev/oracleasm mount point                     [ OK ]
Loading module "oracleasm"                             [ OK ]
Mounting ASMLib driver filesystem                      [ OK ]
Scanning system for ASM disks                          [ OK ]
```

以上操作将加载 oracleasm.o 驱动，并且 mount 上 ASM 文件系统，同时我们可以通过以下命令来手工的卸载和加载 ASMLib

```
# /etc/init.d/oracleasm disable
```

```
Writing Oracle ASM library driver configuration      [ OK ]
Unmounting ASMLib driver filesystem                 [ OK ]
Unloading module "oracleasm"                       [ OK ]

# /etc/init.d/oracleasm enable
Writing Oracle ASM library driver configuration      [ OK ]
Loading module "oracleasm"                         [ OK ]
Mounting ASMLib driver filesystem                   [ OK ]
Scanning system for ASM disks                       [ OK ]
```

如果当前操作系统的内核和编译 ASMLib 的内核不同，那么上面的操作将会出错，请检查 /var/log/messages 文件，确认需要更新的系统内核版本。

添加 **init** 文件使系统启动时自动加载 **ASMLib**

```
$ su -
# cd /etc/rc3.d
# ln -s ../init.d/oracleasm S99oracleasm
# ln -s ../init.d/oracleasm K01oracleasm
```

注意：如果系统启动是自动进入图形界面，也就是系统的 **init level** 是 5，那么上面生成链接的命令应该在 **/etc/rc5.d** 目录中操作。

重新启动系统，确认 **ASMLib** 已经可以自动加载

```
#lsmod | grep oracleasm
oracleasm          14224    1

#dmesg | grep oracleasm
ASM: oracleasmfs mounted with options: <defaults>
```

创建 **ASM** 磁盘

```
# /etc/init.d/oracleasm createdisk VOL1 /dev/sdb
Marking disk "/dev/sdb" as an ASM disk [ OK ]
# /etc/init.d/oracleasm createdisk VOL2 /dev/sdc
Marking disk "/dev/sdc" as an ASM disk [ OK ]
# /etc/init.d/oracleasm createdisk VOL3 /dev/sdd
Marking disk "/dev/sdd" as an ASM disk [ OK ]
# /etc/init.d/oracleasm createdisk VOL4 /dev/sde
Marking disk "/dev/sde" as an ASM disk [ OK ]

# /etc/init.d/oracleasm listdisks
VOL1
VOL2
VOL3
VOL4
```

如果要删除 ASM 磁盘通过以下命令：

```
# /etc/init.d/oracleasm deletedisk VOL4  
Removing ASM disk "VOL4" [ OK ]
```

注意：如果是在 RAC 环境中的某一个节点中添加了 ASM 磁盘，那么需要在其它的节点上运行 scandisk 来获取这种变化。

```
# /etc/init.d/oracleasm scandisks  
Scanning system for ASM disks [ OK ]
```

详细的安装文档可以通过以下连接查看：

<http://otn.oracle.com/tech/linux/asmlib/install.html>

现在我们已经完成了创建 ASM 实例的物理基础，下面开始安装数据库。

安装 Oracle10g

在 RHEL3 上安装 Oracle10g 已经是非常简单的操作了，只要系统内核参数设置正确，几乎不会碰到任何问题。

从以下连接可以得到在 RHEL3 上安装 Oracle10g 的详细步骤，本文不再赘述。

<http://www.puschitz.com/InstallingOracle10g.shtml>

在安装 Oracle10g 的时候，我们选择不创建数据库，只安装软件的方式，在软件安装结束以后再通过 dbca 来创建 ASM 实例和数据库。

创建 ASM 实例

1. 创建初始化文件

```
$ su - oracle
```

```
$ cd $ORACLE_HOME/dbs
```

```
$ vi init+ASM.ora
```

创建 asm 实例的初始化脚本，脚本内容如下所示：

```
# *****Below is init script for asm instance*****  
*.asm_diskstring='ORCL:VOL*'  
*.background_dump_dest='/oracle/admin/+ASM/bdump'  
*.core_dump_dest='/oracle/admin/+ASM/cdump'  
*.instance_type='asm'  
*.large_pool_size=12M  
*.remote_login_passwordfile='SHARED'
```

```
*.user_dump_dest='/oracle/admin/+ASM/udump'  
# *****Above is init script for asm instance*****
```

说明:

1. `dump` 目录指向 `$ORACLE_BASE/<sid>`, 本文中 ASM 实例的 SID 是 `+ASM`。最好不要修改这个实例名, 否则在下一步用 `dbca` 创建数据库的时候, `dbca` 将找不到可用的磁盘组, 我想这应该是 `dbca` 的一个 bug, 手工创建 ASM 类型的数据库就应该没有这个限制了。
2. `asm_diskstring` 用于实例启动的时候检查可用的 ASM 磁盘, 由于我们前面创建 ASM 磁盘的名称是 `VOL1` 到 `VOL4`, 所以我们此处使用 *通配符, 前面的 `'ORCL:'` 不可以省略, 否则 ASM 实例将无法检查到磁盘。

2. 创建密码文件

```
$ su - oracle  
$ cd $ORACLE_HOME/dbs  
$ orapwd file=orapw+ASM password=dba
```

3. 创建目录结构

```
$ su - oracle  
$ cd $ORACLE_HOME/dbs  
$ mkdir -p $ORACLE_BASE/admin/+ASM/udump  
$ mkdir -p $ORACLE_BASE/admin/+ASM/bdump  
$ mkdir -p $ORACLE_BASE/admin/+ASM/cdump
```

4. 启动实例

```
$ export ORACLE_SID=+ASM  
$ sqlplus / as sysdba  
SQL> startup  
ASM instance started
```

```
Total System Global Area 100663296 bytes  
Fixed Size 777616 bytes  
Variable Size 99885680 bytes  
Database Buffers 0 bytes  
Redo Buffers 0 bytes  
ORA-15110: no diskgroups mounted
```

因为我们是首次启动 asm 实例, 还没有创建 diskgroup, 所以显示 15110 错误是正常的。

如果启动实例的时候碰到如下报错:

```
ORA-29701: unable to connect to Cluster Manager
```

那么请检查 `/etc/inittab` 文件, 看看是否有下面这行

```
h1:35:respawn:/etc/init.d/init.cssd run >/dev/null 2>&1 </dev/null
```

如果没有请添加, 如果被注释了请取消注释。

5. 创建 spfile, 重新启动

```
SQL> create spfile from pfile;
```

```
SQL> shutdown immediate;
```

```
SQL> startup;
```

6. 创建 diskgroup

```
SQL> create diskgroup dgroup1 normal redundancy
```

```
  2 failgroup fgroup1 disk 'ORCL:VOL1','ORCL:VOL2'
```

```
  3 failgroup fgroup2 disk 'ORCL:VOL3','ORCL:VOL4';
```

Diskgroup created.

```
SQL> col name for a10
```

```
SQL> select name,state from v$asm_diskgroup;
```

NAME	STATE
-----	-----
DGROUP1	MOUNTED

如果上述命令返回的 state 是 DISMOUNTED, 那么我们需要手工将挂载 diskgroup

```
SQL> ALTER DISKGROUP dgroup1 MOUNT;
```

Diskgroup altered.

此时检查数据库初始化参数, 发现 asm_diskgroups 参数值已经自动设置为 DGROUP1, 这表明在下次启动 asm 实例的时候, 这个 diskgroup 会被自动挂载。

```
SQL> col name for a10
```

```
SQL> col type for a10
```

```
SQL> col value for a20
```

```
SQL> show parameter asm_diskgroups
```

NAME	TYPE	VALUE
-----	-----	-----
asm_diskgroups	string	DGROUP1

如果要卸载 diskgroup, 使用下面的命令:

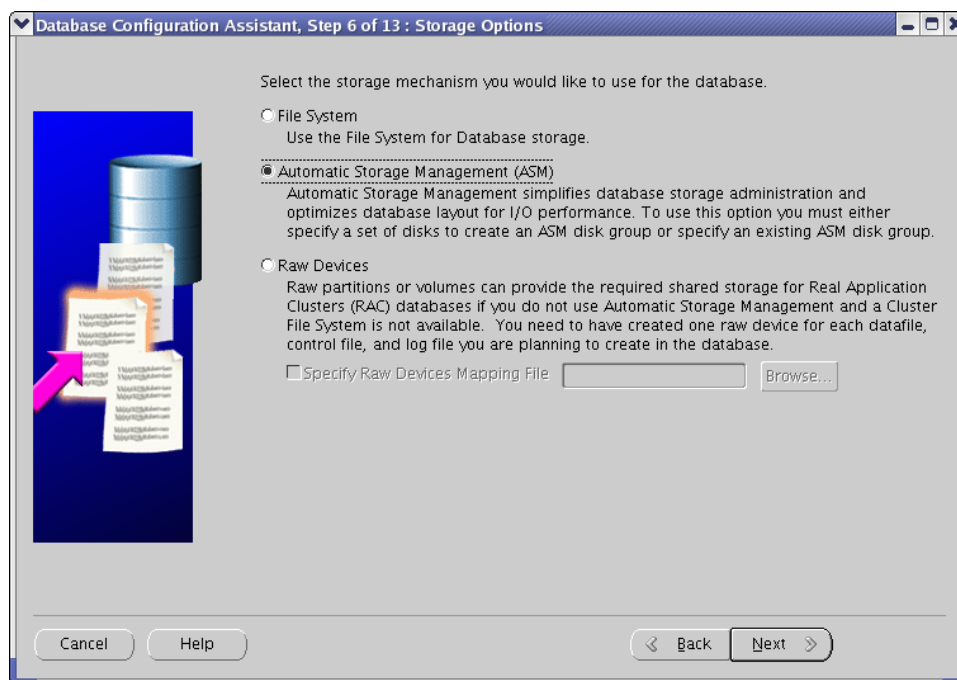
```
ALTER DISKGROUP ALL DISMOUNT;
```

创建 ASM 数据库

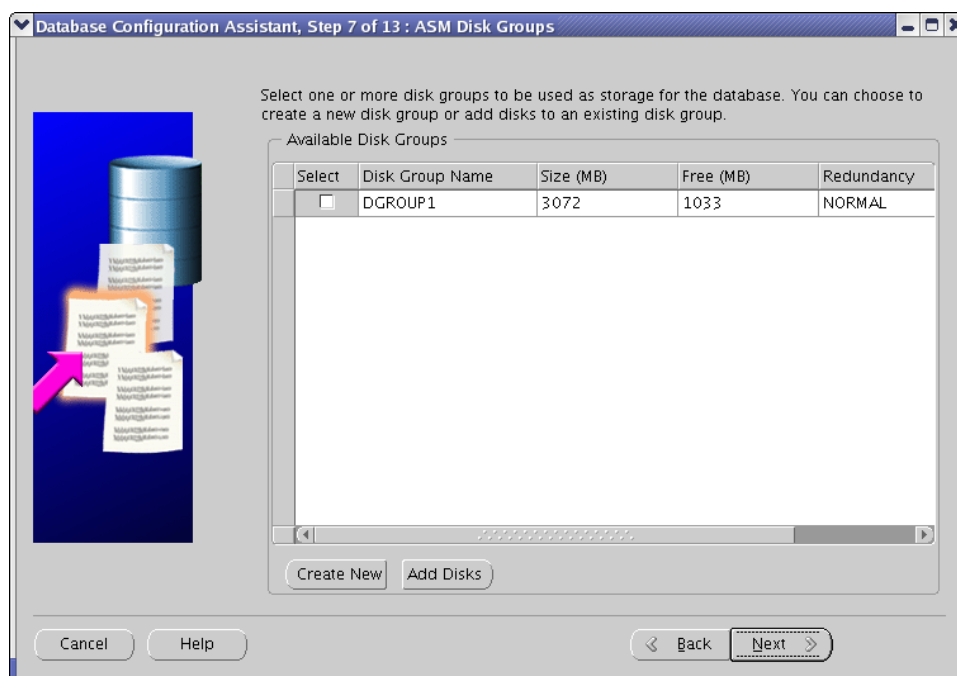
用 dbca 创建数据库, 一直到选择数据类型的画面以前都跟创建普通数据库没有区别。

下面是 dbca 创建数据库的第 6 步画面, 普通的数据库我们通常选择 File System, 而 ASM

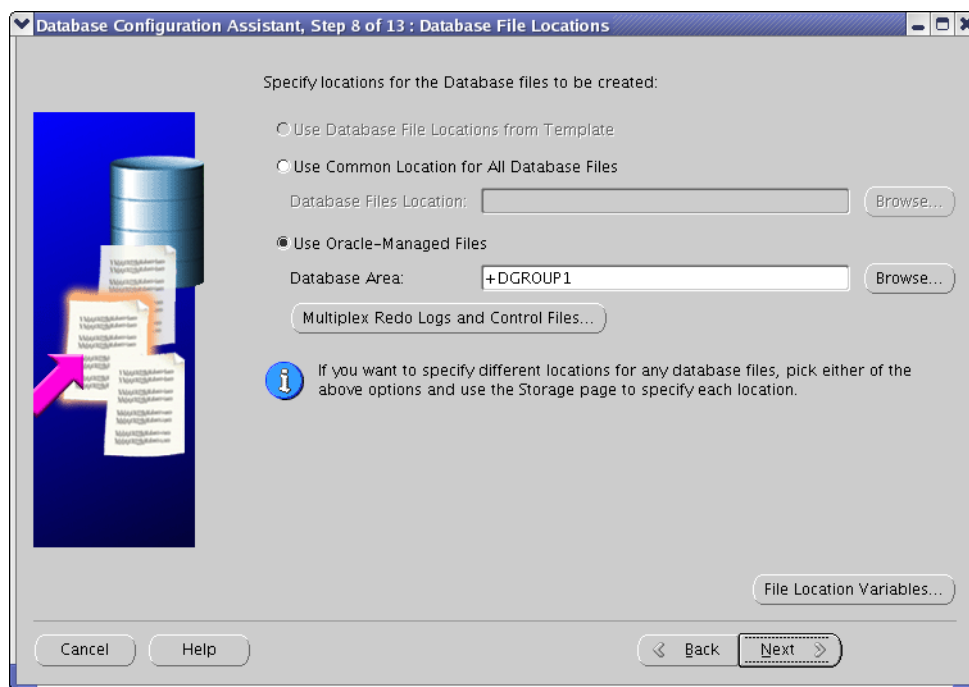
类型的数据库我们则要选择“Automatic Storage Management”。



下一步，oracle 将自动检查出已经加载的 Diskgroup，在这个界面我们可以通过 Create New 按钮创建新的 Diskgroup，也可以通过 Add Disks 按钮向当前的磁盘组中添加额外的磁盘，当然这些磁盘必须是先用 oracleasm createdisk 过的（在第 4 章节的**创建 ASM 磁盘**部分有所叙述）。



在数据文件的位置管理设定界面，通常应该使用 OMF，由 oracle 自己管理数据文件的命名，这样才算是进一步解放了 DBA 对于物理存储方面的繁琐工作。



继续往下的步骤，跟创建普通数据库已经没有什么区别了。不再赘述。

启动 ASM 数据库的步骤

由于启动 ASM 数据库必须要先启动 ASM 实例，所以基本上启动步骤如下。

1. 以 oracle 用户进入操作系统
2. `$ export ORACLE_SID=+ASM`
3. `SQL> sqlplus / as sysdba`
4. `SQL> startup`
5. `SQL> exit`
6. `$ export ORACLE_SID=<your_real_db_sid>`
7. `SQL> sqlplus / as sysdba`
8. `SQL> startup`

ASM 数据库的简单测试

测试 ASM 数据库的 automatic rebalance 功能

添加两个虚拟硬盘，分别是 sdf 和 sdg，大小 500M
启动系统，启动 ASM 实例，启动数据库实例

创建 ASM 磁盘

```

#/etc/init.d/oracleasm createdisk VOL5 /dev/sdf
#/etc/init.d/oracleasm createdisk VOL6 /dev/sdg

```

登入 ASM 实例检查 v\$asm_disk 视图，发现这两个新加的硬盘已经找到了
将硬盘添加到磁盘组中

```

SQL> ALTER DISKGROUP DGROUP1 ADD FAILGROUP FGROUP1 DISK
'ORCL:VOL5' FAILGROUP FGROUP2 DISK 'ORCL:VOL6';

```

Diskgroup altered.

检查数据库的 auto rebalance 工作进度，数据库正在将原来分布在两个磁盘上的数据平均分配到三个磁盘上

```

SQL> SELECT group_number, operation, state, est_work, sofar, est_rate,
est_minutes FROM v$asm_operation;

```

GROUP_NUMBER	OPERATION	STATE	EST_WORK	SO FAR
EST_RATE	EST_MINUTES			
1	REBAL	RUN	416	0
0				0

```
SQL> /
```

GROUP_NUMBER	OPERATION	STATE	EST_WORK	SO FAR
EST_RATE	EST_MINUTES			
1	REBAL	RUN	275	39
1				148

```
SQL> /
```

以上表示数据重新分配的工作已经成功完成

```

SQL> select name,allocation_unit_size,total_mb from v$asm_diskgroup;

```

NAME	ALLOCATION_UNIT_SIZE	TOTAL_MB
DGROUP1	1048576	5120

现在磁盘组空间总共是 5120M

登入数据库实例

创建一个自己的表空间，创建一个用户，创建一个表

```
SQL> create tablespace ts_test datafile '+DGROU1' size 200M;
```

Tablespace created.

```
SQL> create user kamus identified by pass default tablespace ts_test;
```

User created.

```
SQL> grant dba to kamus;
```

Grant succeeded.

```
SQL> conn kamus/pass
```

Connected.

```
SQL> create table t_test as select * from dba_objects;
```

Table created.

```
SQL> select count(*) from t_test;
```

```

COUNT(*)
-----
      10319

```

登入 ASM 实例，将磁盘从磁盘组中删除

```
SQL> alter diskgroup dgroup1 drop disk VOL4;
```

Diskgroup altered.

```
SQL> SELECT group_number, operation, state, est_work, sofar, est_rate,
est_minutes FROM v$asm_operation;
```

GROUP_NUMBER	OPERATION	STATE	EST_WORK	SOFAR
1	REBAL	RUN	566	46
3				151

```
SQL> select name,allocation_unit_size,total_mb from v$asm_diskgroup;
```

NAME	ALLOCATION_UNIT_SIZE	TOTAL_MB
-----	-----	-----

```
DGROUP1                                1048576      4096
```

Executed in 0.13 seconds

```
SQL> select label,failgroup from v$asm_disk;
```

LABEL	FAILGROUP
VOL1	FGROUP1
VOL2	FGROUP1
VOL3	FGROUP2
VOL5	FGROUP1
VOL6	FGROUP2

可以看到磁盘组的总容量已经减少了，磁盘数目也减少了

有些文档上说删除了磁盘之后要执行：**ALTER DISKGROUP REBALANCE**

但是发现不需要手动执行，整个 **REBALANCE** 的工作确实是自动的，当删除磁盘以后，数据库自动开始作 **REBALANCE** 了

为了两个 **FAILGROUP** 的大小相同，再删除一个磁盘

```
SQL> alter diskgroup dgroup1 drop disk VOL2;
```

Diskgroup altered.

我们可以从下面的 SQL 知道在 **ASM** 实例中可以查询到所有可以用的磁盘，而在数据库实例中只可以查询到让自己使用的磁盘

ASM 实例：

```
SQL> SELECT disk_number, label FROM V$asm_disk;
```

DISK_NUMBER	LABEL
0	VOL4
1	VOL2
0	VOL1
2	VOL3
4	VOL5
5	VOL6

数据库实例：

```
SQL> SELECT disk_number, label FROM V$asm_disk;
```

DISK_NUMBER	LABEL
0	VOL1

2 VOL3
4 VOL5
5 VOL6

总结

通过上面简单的测试，我们已经可以基本领略到 ASM 数据库在数据文件存储方面的强大功能，欢迎大家对于 ASM 数据库进行更深入的讨论。

如果有任何问题，可以到 www.itpub.net 的 Oracle10g 和新技术板块进行讨论，也可以给我发邮件。我的邮件地址见本文开始处。