

AIX5.3+HACMP+Oracle9I+Weblogic8.1 安 装实施报告

作者:肖艳华(CNOUG ID:brotherxiao)

郭岳(CNOUG ID:David.Guo)

Blog: <http://www.oracledba.com.cn>

原文来自www.oracledba.com.cn,如需转载,请勿更改文档任何部分

第一章 软硬件清单

开始安装前请仔细检查以下内容是否齐全

硬件设备

IBM P510 两台

DS4300 阵列柜一台, 含 4 块 300G 光纤硬盘

盘柜相关附件(串口线 2 根, 光纤线 2 根)

HP DL380 两台

HP KVM 模块及液晶显示器

IBM, HP 标准 4 2 U 机柜各一个, IBM 机柜含 PDU 模块

其他随机附件

软件

AIX5.3 系统安装光盘

AIX5.3 update 补丁光盘

AIX5.3 HACMP 安装光盘

Oracle9i for AIX 安装光盘

Weblogic for AIX5L 安装光盘

Unzip 解压工具包

Windows 2003 server

杀毒软件

工具

一字螺丝刀一把

十字螺丝刀一把

剪刀等其他

第二章 硬件安装

1. 机柜安装

旋转 I B M 机柜底部四个支脚螺丝升起 I B M 机柜，将机柜移动至合适位置。确定好机柜位置后（请考虑电源和网线介入方便）

2. 按照组装图拼装 H P 机柜。

3. 将所有机器安装到机柜上，并接好 UPS 电源。

第三章 盘柜配置

1. 接上盘柜电源，打开电源开关

2. 盘柜启动完成后，接通网线到交换机

3. 在本机上安装 ibm 存储管理软件 storage manager 9，并确定能 ping 192.168.128.101
(默认配置 IP: 192.168.128.101/102.255.255.0)

4. 使用 storage manager 9 连接到盘柜进行配置

5. 划分逻辑驱动器大小为 50G 每个，共 11 个

第四章 操作系统安装

基本操作系统安装

- 打开主机电源。
- 连接好系统终端，把第一张安装介质（磁带、光碟）插入驱动器。
- 在开机后按 <F5>（图形终端）进入系统安装画面。
- 当终端显示如下信息时：

☆☆☆☆☆☆Please define the system console☆☆☆☆☆☆

Type a 1 and press enter to use this terminal as the system console.

Type een 1 en druk op enter om deze terminal als de systeemconsole te gebruiken.

Skrive tallet 1 og trykk paa enter for aa bruke denne terminalen som systemkonsoll.

Pour definir ce terminal comme console systeme, appuyez sur 1 puis sur entree.
Taste 1 and anschliessend die eingabetaste druecken, um
diese datenstation als systemkonsole zu verwenden.
Premier 11 tasto 1 ed invio per usare questo terminal como consolo.
Escriba 1 y pulse intro para utilizar esta terminal como consola del sistema.
Tryck paa 1 och sedan paa enter om du vill att haer terminalen ska vara systemkonsol

键入“1”并回车（注意：键入的“1”不回显）选择主控台

- 屏幕上将不断显示一些信息，几分钟后出现：
- >>> 1 Type 1 and press enter to have English during install.
- 2 Type 2 en druk op enter om tijdens het installeren het Nederlands te krijgen.
- 3 Entrez 3 pour effectuer 1 installation en francais.
- 4 Fr Installation in deutscher sprache 4 eingeben und die eingabetaste drcken.
- 5 Immettere 5 e premere invio per 1 installazione italiana.
- 6 Escriba 6 y pulse intro para usar el idioma español durante la instalación.
- 7 Skriv 7 och tryck ned enter=svenska vid installationen.

88 Help?

>>>Choice 【1】：

键入“1”后回车，选择语言环境为 English

- 此后屏幕出现：

```
Welcome to Base Operating System
Installation and Maintenance

Type the number of your choice and press enter .Choice is indicated by>>>
>>>1 Start install now with default settings
    2 Change/show installation settings and install
    3 Start Maintenance Mode for system Recovery

88 Help?
99 Previous Menu
>>>Choice (1) :
```

这是系统安装和维护的主菜单。

- 安装 BOS 基本操作系统

键入“2”并回车，屏幕出现“Install and Setting”画面：

Installation and a Settings

Either type 0 and press enter to install with current settings, or type the number of the setting you want to change and press enter.

- 1 system setting:
 - Method of installation.....Preserve install
 - Disk when you want to installhdisk0
- 2 Primary Language Environment settings (AFTER Install):
 - Cultural Convention..... English(United States)
 - Language..... English(United States)
 - Keyboard..... English(United States)
 - Keyboard Type..... Default
- 3 Install Trusted Computing Base....No

>>>0 Install **AIX** with the current settings listed above.

- 88 Help?
- 99 Previous Menu

>>>Choice (1):

这是系统安装的默认设置，用户可以根据需要进行修改。
与此同时，在屏幕的右下角会出现如下的警告信息：

WARNING:Base Operating system installation will destroy or impair recovery of ALL data on the destination disk hdisk0.

以上是提醒用户安装系统的后果将破坏用户原有的硬盘上的数据。
安装时可以选择保护安装(Preserve install) 或完全覆盖安装(New and complete overwrite)。

我们这里选择安装类型为全新安装，磁盘 **hdisk0**，语言 **English(United States)**—选项 48，其他默认。

- 设置完毕选择 0 安装即开始，屏幕将不断显示安装的进行情况，中间过程中按照提示换盘。当 BOS 安装完毕系统将自动重启，否则在 BOS 安装完毕时屏幕会再次提示按钥匙并处于等待状态。
- 系统重新启动后，屏幕会提示输入终端类型，键入主控台的类型(vt100 或 ibm3151)，此后将出现系统设置屏，用上下键移动亮条进行基本系统参数的设置：

★set date and time 设置 time zone 选择 cur+8

★set root password 分两次输入相同口令，即可设置成功，移动光标至 Task complete-Exit to AIX LOGIN 退出。

★其他暂不设置

注意：不能用 F3 或 F10 键退出，而应选菜单以通知操作系统设置任务已完成；否则，在下次启机时仍会自动进入系统设置画面。同时如果终端类型输入错误，屏幕显示将不可预料，此时可以用 ctrl-c 退出，重新输入。

至此，BOS安装完毕，以后启动机器时将不会有配置信息，直接出现LOGIN提示符。

扩展软件包的安装

操作系统安装完成后，我们下面将进行操作系统扩展软件包的安装，其具体安装步骤如下：

1、以 root 用户登录系统。

2、在“workspace”中打开一个终端并装载光驱

```
mount -v cdrfs -o ro /dev/cd0 /mnt
```

3、在命令提示符下执行以下命令安装扩展包

```
# smitty install
```

F4 选择安装文件集：

```
bos. data, bos. compat, bos. perf, bos. clvm, bos. adt
```

```
x11. dt, x11. compat, rsct. basic, rsct. core, rsct. compat
```

4、安装过程中请根据提示更换光盘

操作系统参数调整

完成扩展包安装后，调整系统相关参数

1、修改paging size

```
Smitty→system storage management→logical volume manager→paging size→change/show
```

```
characteristics of a paging size
```

增加28 个LP（默认为ps为512M, lp为128M）

```
lsps -a 可查看大小为4G
```

2、修改floating license

```
Smitty→system environment→change /show number of Licensed Users→floating licensing
```

(off→on)

3、增加tty设备

```
Smitty tty→add a tty
```

类型选择: tty rs232 Asynchronous Terminal, port number为0

4、设置异步IO

smitty→devices→ Asynchronous I/O→Posix Asynchronous I/O→Change / Show Characteristics of

Asynchronous I/O

```
MINIMUM number of servers [30]
#
MAXIMUM number of servers per cpu [60]
#
Maximum number of REQUESTS [4096]
#
Server PRIORITY [39]
#
STATE to be configured at system restart available
State of fast path enable
```

5、修改high, lower water mark

smitty→system environment→change/show characteristics of operating system

```
System ID 0X8000170E9F700000
Partition ID 0X8000170E9F700001
Maximum number of PROCESSES allowed per user [1024]
+#
Maximum number of pages in block I/O BUFFER CACHE [80]
+#
Maximum Kbytes of real memory allowed for MBUFS [0]
+#
Automatically REBOOT system after a crash true
+
Continuously maintain DISK I/O history false
+
HIGH water mark for pending write I/Os per file [33]
+#
```

```

LOW water mark for pending write I/Os per file [24]
+#
Amount of usable physical memory in Kbytes 1900544
State of system keylock at boot time normal
Enable full CORE dump false
+
Use pre-430 style CORE dump false
+
Pre-520 tuning compatibility mode disable
+
Maximum login name length at boot time [9]
+#
Stack Execution Disable (SED) Mode select
+
ARG/ENV list size in 4K byte blocks [6]
+#
CPU Guard enable
+
Processor capacity increment 1.00
Partition is capped true
Partition is dedicated true
Entitled processor capacity 1.00
Minimum potential processor capacity 1.00
Maximum potential processor capacity 1.00
Variable processor capacity weight 0

```

4、rootvg镜像

1. 添加新硬盘到 rootvg

```
#extendvg rootvg hdisk1
```

2. 镜像 rootvg

```
#mirrorvg -c 2 rootvg hdisk1
```

3. 重新生成 boot image

```
#bosboot -ad /dev/hdisk0
```

4. 更新 bootlist

```
#bootlist -m normal hdisk0 hdisk1 cd0
```

5. 重起系统

```
#shutdown -Fr
```

基本网络配置

Smitty tcpip → minimum configuration&startup

1 号主机

En0:	en1
IP:172.19.74.197	192.10.10.1
255.255.255.0	255.255.255.0

2 号主机

En0:	en1
IP:172.19.4.198	192.10.10.2
255.255.255.0	255.255.255.0

第五章 安装 HACMP 软件包

开始安装

- 1、 插入 hacmp 光盘
- 2、 `mount -v cdrfs -o ro /dev/cd0 /mnt`
- 3、 `smitty install`
- 4、 按 F4 选择安装的文件集：除了 `cluster.hativoli`, `cluster.haview`, `rsct.exp` 的包以外，所有的 hacmp 的包都要安装。

重启机器

```
Shutdown -Fr
```

第六章 配置卷组

盘柜配置完成后可开始创建 datavg

(如果在开机后接入光纤线, 请运行 `cfgmgr`, 正确情况下 `lsdev -Cc disk` 应该能看到 `hdisk2,hdisk3...hdisk12`)

1. `smitty mkvg`

Add a Volume Group

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

```

                                     [Entry Fields]
VOLUME GROUP name                    [datavg]
Physical partition SIZE in megabytes  256          +
* PHYSICAL VOLUME names                [hdisk2,hdisk3...hdisk11]
+
Force the creation of a volume group?  no          +
Activate volume group AUTOMATICALLY   no          +
    at system restart?
Volume Group MAJOR NUMBER              []          +#
Create VG Concurrent Capable?         no          +
Create a big VG format Volume Group?  no          +
LTG Size in kbytes                     128        +
```

用同样的方法创建 `webvg` (PV:hdisk12)

将建好的 `datavg`, `webvg` import 到另外一台服务器上并安装 Oracle。

在 1 号机 `varyoffvg datavg`, 然后在 2 号机 `import vg`。

2、创建逻辑卷

将脚本 `mkgrv40lv.sh` 上传到任一服务器执行 (注意要先 `varyonvg`)

```
mklv -y system      -w n -s n -r n -t jfs  datavg  8
```

```
mklv -y user        -w n -s n -r n -t jfs  datavg  8
```

```
mklv -y rbs01       -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y rbs02       -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y rbs03       -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y rbs04       -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y rbs05       -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y temp01      -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y temp02      -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y temp03      -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y temp04      -w n -s n -r n -t jfs  datavg  16
```

```
mklv -y temp05      -w n -s n -r n -t jfs  datavg  16
```



```
mklv -y xcgl_txjx08 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_txjx09 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_txjx10 -w n -s n -r n -t jfs datavg 16

mklv -y xcgl_idx01 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx02 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx03 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx04 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx05 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx06 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx07 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx08 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx09 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_idx10 -w n -s n -r n -t jfs datavg 16

mklv -y xcgl_ycxx01 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx02 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx03 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx04 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx05 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx06 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx07 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx08 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx09 -w n -s n -r n -t jfs datavg 16
mklv -y xcgl_ycxx10 -w n -s n -r n -t jfs datavg 16
```

```
chown oracle:dba /dev/rsystem
```

```
chown oracle:dba /dev/ruser
```

```
chown oracle:dba /dev/rrbs01
```

```
chown oracle:dba /dev/rrbs02
```

```
chown oracle:dba /dev/rrbs03
```

```
chown oracle:dba /dev/rrbs04
```

```
chown oracle:dba /dev/rrbs05
```

```
chown oracle:dba /dev/rtemp01
```

```
chown oracle:dba /dev/rtemp02
```

```
chown oracle:dba /dev/rtemp03
```

```
chown oracle:dba /dev/rtemp04
```

```
chown oracle:dba /dev/rtemp05
```

```
chown oracle:dba /dev/rredo01
```

```
chown oracle:dba /dev/rredo02
chown oracle:dba /dev/rredo03
chown oracle:dba /dev/rredo04
chown oracle:dba /dev/rredo05
chown oracle:dba /dev/rredo06
```

```
chown oracle:dba /dev/rgd01
```

```
chown oracle:dba /dev/rxcgl_daxx01
```

```
chown oracle:dba /dev/rxcgl_rwsj01
chown oracle:dba /dev/rxcgl_rwsj02
chown oracle:dba /dev/rxcgl_rwsj03
chown oracle:dba /dev/rxcgl_rwsj04
chown oracle:dba /dev/rxcgl_rwsj05
chown oracle:dba /dev/rxcgl_rwsj06
chown oracle:dba /dev/rxcgl_rwsj07
chown oracle:dba /dev/rxcgl_rwsj08
chown oracle:dba /dev/rxcgl_rwsj09
chown oracle:dba /dev/rxcgl_rwsj10
```

```
chown oracle:dba /dev/rxcgl_gdsj01
chown oracle:dba /dev/rxcgl_gdsj02
chown oracle:dba /dev/rxcgl_gdsj03
chown oracle:dba /dev/rxcgl_gdsj04
chown oracle:dba /dev/rxcgl_gdsj05
chown oracle:dba /dev/rxcgl_gdsj06
chown oracle:dba /dev/rxcgl_gdsj07
chown oracle:dba /dev/rxcgl_gdsj08
chown oracle:dba /dev/rxcgl_gdsj09
chown oracle:dba /dev/rxcgl_gdsj10
```

```
chown oracle:dba /dev/rxcgl_ycxx01
chown oracle:dba /dev/rxcgl_ycxx02
chown oracle:dba /dev/rxcgl_ycxx03
chown oracle:dba /dev/rxcgl_ycxx04
chown oracle:dba /dev/rxcgl_ycxx05
chown oracle:dba /dev/rxcgl_ycxx06
chown oracle:dba /dev/rxcgl_ycxx07
chown oracle:dba /dev/rxcgl_ycxx08
chown oracle:dba /dev/rxcgl_ycxx09
chown oracle:dba /dev/rxcgl_ycxx10
```

```
chown oracle:dba /dev/rxcgl_txjx01
chown oracle:dba /dev/rxcgl_txjx02
chown oracle:dba /dev/rxcgl_txjx03
chown oracle:dba /dev/rxcgl_txjx04
chown oracle:dba /dev/rxcgl_txjx05
chown oracle:dba /dev/rxcgl_txjx06
chown oracle:dba /dev/rxcgl_txjx07
chown oracle:dba /dev/rxcgl_txjx08
chown oracle:dba /dev/rxcgl_txjx09
chown oracle:dba /dev/rxcgl_txjx10
```

```
chown oracle:dba /dev/rxcgl_idx01
chown oracle:dba /dev/rxcgl_idx02
chown oracle:dba /dev/rxcgl_idx03
chown oracle:dba /dev/rxcgl_idx04
chown oracle:dba /dev/rxcgl_idx05
chown oracle:dba /dev/rxcgl_idx06
chown oracle:dba /dev/rxcgl_idx07
chown oracle:dba /dev/rxcgl_idx08
chown oracle:dba /dev/rxcgl_idx09
chown oracle:dba /dev/rxcgl_idx10
```

```
chown oracle:dba /dev/system
```

```
chown oracle:dba /dev/user
```

```
chown oracle:dba /dev/rbs01
chown oracle:dba /dev/rbs02
chown oracle:dba /dev/rbs03
chown oracle:dba /dev/rbs04
chown oracle:dba /dev/rbs05
```

```
chown oracle:dba /dev/temp01
chown oracle:dba /dev/temp02
chown oracle:dba /dev/temp03
chown oracle:dba /dev/temp04
chown oracle:dba /dev/temp05
```

```
chown oracle:dba /dev/redo01
chown oracle:dba /dev/redo02
chown oracle:dba /dev/redo03
```

```
chown oracle:dba /dev/redo04
chown oracle:dba /dev/redo05
chown oracle:dba /dev/redo06
```

```
chown oracle:dba /dev/gd01
```

```
chown oracle:dba /dev/xcgl_daxx01
```

```
chown oracle:dba /dev/xcgl_rwsj01
chown oracle:dba /dev/xcgl_rwsj02
chown oracle:dba /dev/xcgl_rwsj03
chown oracle:dba /dev/xcgl_rwsj04
chown oracle:dba /dev/xcgl_rwsj05
chown oracle:dba /dev/xcgl_rwsj06
chown oracle:dba /dev/xcgl_rwsj07
chown oracle:dba /dev/xcgl_rwsj08
chown oracle:dba /dev/xcgl_rwsj09
chown oracle:dba /dev/xcgl_rwsj10
```

```
chown oracle:dba /dev/xcgl_gdsj01
chown oracle:dba /dev/xcgl_gdsj02
chown oracle:dba /dev/xcgl_gdsj03
chown oracle:dba /dev/xcgl_gdsj04
chown oracle:dba /dev/xcgl_gdsj05
chown oracle:dba /dev/xcgl_gdsj06
chown oracle:dba /dev/xcgl_gdsj07
chown oracle:dba /dev/xcgl_gdsj08
chown oracle:dba /dev/xcgl_gdsj09
chown oracle:dba /dev/xcgl_gdsj10
```

```
chown oracle:dba /dev/xcgl_ycxx01
chown oracle:dba /dev/xcgl_ycxx02
chown oracle:dba /dev/xcgl_ycxx03
chown oracle:dba /dev/xcgl_ycxx04
chown oracle:dba /dev/xcgl_ycxx05
chown oracle:dba /dev/xcgl_ycxx06
chown oracle:dba /dev/xcgl_ycxx07
chown oracle:dba /dev/xcgl_ycxx08
chown oracle:dba /dev/xcgl_ycxx09
chown oracle:dba /dev/xcgl_ycxx10
```

```
chown oracle:dba /dev/xcgl_txjx01
chown oracle:dba /dev/xcgl_txjx02
chown oracle:dba /dev/xcgl_txjx03
chown oracle:dba /dev/xcgl_txjx04
chown oracle:dba /dev/xcgl_txjx05
chown oracle:dba /dev/xcgl_txjx06
chown oracle:dba /dev/xcgl_txjx07
chown oracle:dba /dev/xcgl_txjx08
chown oracle:dba /dev/xcgl_txjx09
chown oracle:dba /dev/xcgl_txjx10
```

```
chown oracle:dba /dev/xcgl_idx01
chown oracle:dba /dev/xcgl_idx02
chown oracle:dba /dev/xcgl_idx03
chown oracle:dba /dev/xcgl_idx04
chown oracle:dba /dev/xcgl_idx05
chown oracle:dba /dev/xcgl_idx06
chown oracle:dba /dev/xcgl_idx07
chown oracle:dba /dev/xcgl_idx08
chown oracle:dba /dev/xcgl_idx09
chown oracle:dba /dev/xcgl_idx10
```

```
chmod 777 /dev/system
chmod 777 /dev/user
chmod 777 /dev/rbs01
chmod 777 /dev/rbs02
chmod 777 /dev/rbs03
chmod 777 /dev/rbs04
chmod 777 /dev/rbs05
chmod 777 /dev/temp01
chmod 777 /dev/temp02
chmod 777 /dev/temp03
chmod 777 /dev/temp04
chmod 777 /dev/temp05
chmod 777 /dev/redo01
chmod 777 /dev/redo02
chmod 777 /dev/redo03
chmod 777 /dev/redo04
chmod 777 /dev/redo05
chmod 777 /dev/redo06
```

```
chmod 777 /dev/gd01
```

```
chmod 777 /dev/xcgl_daxx01
```

```
chmod 777 /dev/xcgl_rwsj01
```

```
chmod 777 /dev/xcgl_rwsj02
```

```
chmod 777 /dev/xcgl_rwsj03
```

```
chmod 777 /dev/xcgl_rwsj04
```

```
chmod 777 /dev/xcgl_rwsj05
```

```
chmod 777 /dev/xcgl_rwsj06
```

```
chmod 777 /dev/xcgl_rwsj07
```

```
chmod 777 /dev/xcgl_rwsj08
```

```
chmod 777 /dev/xcgl_rwsj09
```

```
chmod 777 /dev/xcgl_rwsj10
```

```
chmod 777 /dev/xcgl_gdsj01
```

```
chmod 777 /dev/xcgl_gdsj02
```

```
chmod 777 /dev/xcgl_gdsj03
```

```
chmod 777 /dev/xcgl_gdsj04
```

```
chmod 777 /dev/xcgl_gdsj05
```

```
chmod 777 /dev/xcgl_gdsj06
```

```
chmod 777 /dev/xcgl_gdsj07
```

```
chmod 777 /dev/xcgl_gdsj08
```

```
chmod 777 /dev/xcgl_gdsj09
```

```
chmod 777 /dev/xcgl_gdsj10
```

```
chmod 777 /dev/xcgl_ycxx01
```

```
chmod 777 /dev/xcgl_ycxx02
```

```
chmod 777 /dev/xcgl_ycxx03
```

```
chmod 777 /dev/xcgl_ycxx04
```

```
chmod 777 /dev/xcgl_ycxx05
```

```
chmod 777 /dev/xcgl_ycxx06
```

```
chmod 777 /dev/xcgl_ycxx07
```

```
chmod 777 /dev/xcgl_ycxx08
```

```
chmod 777 /dev/xcgl_ycxx09
```

```
chmod 777 /dev/xcgl_ycxx10
```

```
chmod 777 /dev/xcgl_txjx01
```

```
chmod 777 /dev/xcgl_txjx02
```

```
chmod 777 /dev/xcgl_txjx03
```

```
chmod 777 /dev/xcgl_txjx04
```

```
chmod 777 /dev/xcgl_txjx05
```

```
chmod 777 /dev/xcgl_txjx06
```

```
chmod 777 /dev/xcgl_txjx07
chmod 777 /dev/xcgl_txjx08
chmod 777 /dev/xcgl_txjx09
chmod 777 /dev/xcgl_txjx10
```

```
chmod 777 /dev/xcgl_idx01
chmod 777 /dev/xcgl_idx02
chmod 777 /dev/xcgl_idx03
chmod 777 /dev/xcgl_idx04
chmod 777 /dev/xcgl_idx05
chmod 777 /dev/xcgl_idx06
chmod 777 /dev/xcgl_idx07
chmod 777 /dev/xcgl_idx08
chmod 777 /dev/xcgl_idx09
chmod 777 /dev/xcgl_idx10
```

```
chmod 777 /dev/rssystem
```

```
chmod 777 /dev/ruser
```

```
chmod 777 /dev/rrbs01
chmod 777 /dev/rrbs02
chmod 777 /dev/rrbs03
chmod 777 /dev/rrbs04
chmod 777 /dev/rrbs05
```

```
chmod 777 /dev/rtemp01
chmod 777 /dev/rtemp02
chmod 777 /dev/rtemp03
chmod 777 /dev/rtemp04
chmod 777 /dev/rtemp05
```

```
chmod 777 /dev/rredo01
chmod 777 /dev/rredo02
chmod 777 /dev/rredo03
chmod 777 /dev/rredo04
chmod 777 /dev/rredo05
chmod 777 /dev/rredo06
```

```
chmod 777 /dev/rgd01
```

```
chmod 777 /dev/rxcgl_daxx01
```

```
chmod 777 /dev/rxcgl_rwsj01
```

```
chmod 777 /dev/rxcgl_rwsj02
chmod 777 /dev/rxcgl_rwsj03
chmod 777 /dev/rxcgl_rwsj04
chmod 777 /dev/rxcgl_rwsj05
chmod 777 /dev/rxcgl_rwsj06
chmod 777 /dev/rxcgl_rwsj07
chmod 777 /dev/rxcgl_rwsj08
chmod 777 /dev/rxcgl_rwsj09
chmod 777 /dev/rxcgl_rwsj10
```

```
chmod 777 /dev/rxcgl_gdsj01
chmod 777 /dev/rxcgl_gdsj02
chmod 777 /dev/rxcgl_gdsj03
chmod 777 /dev/rxcgl_gdsj04
chmod 777 /dev/rxcgl_gdsj05
chmod 777 /dev/rxcgl_gdsj06
chmod 777 /dev/rxcgl_gdsj07
chmod 777 /dev/rxcgl_gdsj08
chmod 777 /dev/rxcgl_gdsj09
chmod 777 /dev/rxcgl_gdsj10
```

```
chmod 777 /dev/rxcgl_ycxx01
chmod 777 /dev/rxcgl_ycxx02
chmod 777 /dev/rxcgl_ycxx03
chmod 777 /dev/rxcgl_ycxx04
chmod 777 /dev/rxcgl_ycxx05
chmod 777 /dev/rxcgl_ycxx06
chmod 777 /dev/rxcgl_ycxx07
chmod 777 /dev/rxcgl_ycxx08
chmod 777 /dev/rxcgl_ycxx09
chmod 777 /dev/rxcgl_ycxx10
```

```
chmod 777 /dev/rxcgl_txjx01
chmod 777 /dev/rxcgl_txjx02
chmod 777 /dev/rxcgl_txjx03
chmod 777 /dev/rxcgl_txjx04
chmod 777 /dev/rxcgl_txjx05
chmod 777 /dev/rxcgl_txjx06
chmod 777 /dev/rxcgl_txjx07
chmod 777 /dev/rxcgl_txjx08
chmod 777 /dev/rxcgl_txjx09
chmod 777 /dev/rxcgl_txjx10
```

```
chmod 777 /dev/rxcgl_idx01
chmod 777 /dev/rxcgl_idx02
chmod 777 /dev/rxcgl_idx03
chmod 777 /dev/rxcgl_idx04
chmod 777 /dev/rxcgl_idx05
chmod 777 /dev/rxcgl_idx06
chmod 777 /dev/rxcgl_idx07
chmod 777 /dev/rxcgl_idx08
chmod 777 /dev/rxcgl_idx09
chmod 777 /dev/rxcgl_idx10
```

完成后同步 vg。

第七章 oracle 安装

Oracle软件安装

1. 在rootvg上建立临时文件系统/oracle_install, 大小10G用于存放安装文件
建立文件系统/oracle用户作为oracle安装目录, 大小10G
设置tmp目录, 确保tmp目录足够(2G), 若tmp满后导致tmp目录不能扩展
2. smitty group→add a group创建dba组(ID:1000)和oinstall组(ID:2000)
3. smitty user创建oracle用户(ID:2000), primary group为dba, group set为system, adm, dba, oinstall
home:/home/oracle,
4. 拷贝(或以二进制模式ftp)安装介质到临时目录/oracle_install(三个.zip文件及一个9207补丁包)
在/oracle_install目录下建立1, 2, 3, 4四个目录用于存放安装文件
su - oracle
cd /oracle_install
mkdir 1
mkdir 2
mkdir 3
mkdir 4
分别插入oracle安装盘1, 2, 3, 4, 将安装文件拷贝到/oracle_install的1, 2, 3, 4目录下
mount -v cdrfs -o ro /dev/cd0 /mnt

```
cp -r /mnt/* /oracle_install/1
```

```
umount /mnt -弹出光盘
```

5. 安装unzip命令解压安装文件(正版安装光盘不需解压)

将unzip解压包unz550x-aix5L.tar.Z上传到/oracle_install目录, 使用uncompress命令解压

```
uncompress unz550x-aix5L.tar.Z --得到 unz550x-aix5L.tar
```

使用tar命令解压 tar -vxf unz550x-aix5L.tar --在解压出来的unzip-5.50目录下的unzip

```
使用unzip解压补丁包文件 ./unzip -d /oracle_install/9207 /oracle_install/9207.zip
```

如果安装文件是zip文件, 则用相同的方法进行解压

6. 用smitty完成创建DBA(1000)组和oracle(2000)用户

oracle用户基本组为dba, 参数 soft file size修改为-1, soft data segment 修改为-1, soft stack size 修改为-1, soft core file size修改为-1

创建oinstall组(1001)

7. 编辑.profile文件如下

```
# Oracle Environment
```

```
ORACLE_BASE=/oracle; export ORACLE_BASE
```

```
ORACLE_HOME=$ORACLE_BASE/product/9.2.0; export ORACLE_HOME
```

```
ORACLE_SID=oradb; export ORACLE_SID
```

```
#ORACLE_TERM=vt100; export ORACLE_TERM
```

```
TNS_ADMIN=$ORACLE_HOME/network/admin;export TNS_ADMIN
```

```
NLS_LANG=american_america.zhs16cgb; export NLS_LANG
```

```
ORA_NLS33=$ORACLE_HOME/ocommon/nls/admin/data; export ORA_NLS33
```

```
LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib; export LD_LIBRARY_PATH
```

```
LIBPATH=$ORACLE_HOME/lib:$ORACLE_HOME/ctx/lib ;export LIBPATH
```

```
ORA_DB=$ORACLE_HOME/dbs;export ORA_DB
```

```
CLASSPATH=$ORACLE_HOME/JRE/lib:$ORACLE_HOME/JRE/lib/rt.jar:$ORACLE_HOME/jlib:
```

```
$ORACLE_HOME/rdbms/jlib; export CLASSPATH
```

```
PATH=$ORACLE_HOME/bin:$PATH; export PATH
```

手工设置DISPLAY到本地:

以root用户执行xhost +

以oracle用户执行export DISPLAY=172.19.74.197:0.0

(如不能启动图形界面可能需要重启服务器)

8. 开始安装

```
su - oracle
```

```
cd /oracle_install/1
```

```
./runInstaller
```

9. 安装过程中请根据提示以root用户执行相关脚本

补丁包安装(9207)

1、将补丁安装文件拷贝到/oracle_install/9207 目录下

2、修改目录权限

```
chown -R oracle:dba /oracle_install/9207
```

```
chmod -R 777 /oracle_install/9207
```

3、执行安装文件./runInstaller，键入 y 并回车，系统将弹出图形化安装界面

4、安装过程根据提示以 root 用户执行相关脚本

创建数据库

以oracle用户运行dbca，创建数据库。

若图形界面出不来，请确定DISPLAY是否设置正确

以root用户执行xhost +

以oracle用户执行export DISPLAY=172.19.74.197:0.0

在图形界面中设置相关参数，最后保存成建库脚本执行。

(因为这里用的是裸设备所以用脚本执行，而不是直接建库)

--9207不能启动dbca/dbua问题

There's a known problem on AIX, if database was patched omitting patchset 9.2.0.4 :

Problem is: library libnjni9.so was not included into later patchsets, but is needed by DBCA/DBUA .

Try the following (before doing this check if libnjni9.so is really missing and we don't have another problem)

1) Download the 9.2.0.4.0 patchset <Bug: 3095277> from Metalink and stage on the server.

2) \$ cd <9204 staging area>/Disk1/stage/Patches/oracle.rsfc.hybrid/9.2.0.3.0/1/DataFiles

3) \$ cp group.1.1.jar /tmp/.

4) \$ cd /tmp

- 5) \$ jar xvf group.1.1.jar libjni9.so --用root用户执行
- 6) \$ mv libjni9.so \$ORACLE_HOME/lib32/.
- 7) Retry DBUA and DBCA

第八章 配置 HA

配置hosts文件

在 CXGRDB1 和 CXGRDB2 主机上的/etc/hosts 文件中用 vi 编辑，内容为：

```
127.0.0.1          loopback localhost      # loopback (lo0) name/address
10.121.11.100 CXGRDB1_svc
10.121.11.102 CXGRDB1_pre
172.19.74.197 CXGRDB1_boot CXGRDB1
192.10.10.1 CXGRDB1_standby

10.121.11.101 CXGRDB2_svc
10.121.11.103 CXGRDB2_pre
172.19.74.198 CXGRDB2_boot CXGRDB2
192.10.10.2      CXGRDB2_standby CXGRDB2
```

配置rhosts文件

CXGRDB1	CXGRDB2
172.19.74.197	172.19.74.198
192.10.10.1	192.10.10.2
172.19.74.198	172.19.74.197
192.10.10.2	192.10.10.1

添加cluster 和node

- 1、smitty hacmp

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tool

Add Nodes to an HACMP Cluster

Configure Resources to Make Highly Available

Configure HACMP Resource Groups
Verify and Synchronize HACMP Configuration
Display HACMP Configuration

* Cluster Name [cxgrdb]
New Nodes (via selected communication paths) [CXGRDB1 CXGRDB2]

Currently Configured Node(s)

F4 选择列表中选择 (ESC + 7) 每个节点的 boot 地址 作为 communication path 。当回车以后，系统会自己 discover hacmp 的资源。

完成后可通过 Display HACMP Configuration 查看结果，显示如下：

```
Cluster Name: cxgrdb
Cluster Connection Authentication Mode: Standard
Cluster Message Authentication Mode: None
Cluster Message Encryption: None
Use Persistent Labels for Communication: No
There are 2 node(s) and 1 network(s) defined
```

NODE CXGRDB1:

```
Network net_ether_01
      CXGRDB2_svc      10.121.11.101
      CXGRDB1_svc      10.121.11.100
      CXGRDB1_boot     172.19.74.197
      CXGRDB1_standby 192.10.10.1
```

NODE CXGRDB2:

```
Network net_ether_01
      CXGRDB2_svc      10.121.11.101
      CXGRDB1_svc      10.121.11.100
      CXGRDB2_boot     172.19.74.198
      CXGRDB2_standby 192.10.10.2
```

.....

添加高可用资源

1. 添加 service ip

Add Nodes to an HACMP Cluster
Configure Resources to Make Highly Available
Configure HACMP Resource Groups
Verify and Synchronize HACMP Configuration
Display HACMP Configuration

Configure Service IP Labels/Addresses

Configure Application Servers

Configure Volume Groups, Logical Volumes and Filesystems

Configure Concurrent Volume Groups and Logical Volumes

Add a Service IP Label/Address

Change/Show a Service IP Label/Address

Remove Service IP Label(s)/Address(es)

添加 CXGRDB1 主机的 Service IP (给 oracle 服务使用)

* IP Label/Address [CXGRDB1_svc]

Network Name [net_ether_01]

2. 添加 application server

Configure Service IP Labels/Addresses

Configure Application Servers

Configure Volume Groups, Logical Volumes and Filesystems

Configure Concurrent Volume Groups and Logical Volumes

Add an Application Server

Change/Show an Application Server

Remove an Application Server

* Server Name [oracle]

* Start Script [/oracle/startoracle.sh]

* Stop Script [/oracle/stoporacle.sh]

3. 创建资源组

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tools

Add Nodes to an HACMP Cluster

Configure Resources to Make Highly Available

Configure HACMP Resource Groups

Verify and Synchronize HACMP Configuration

Display HACMP Configuration

Add a Resource Group

Change/Show a Resource Group

Remove a Resource Group

Change/Show Resources for a Resource Group (standard)

选择资源组类型

Cascading

Rotating

Concurrent

Custom

添加 oracle 资源

* Resource Group Name [oracle]

* Participating Node Names / Default Node Priority [CXGRDB1 CXGGRDB2]

4. 配置资源

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tools

Add Nodes to an HACMP Cluster

Configure Resources to Make Highly Available

Configure HACMP Resource Groups

Verify and Synchronize HACMP Configuration

Display HACMP Configuration

Add a Resource Group

Change/Show a Resource Group

Remove a Resource Group

Change/Show Resources for a Resource Group (standard)

选择 oracle

Resource Group Name oracle

Participating Node Names (Default Node Priority) CXGRDB1 CXGRDB2

* Service IP Labels/Addresses [CXGRDB1_svc]

Volume Groups [datavg]

Filesystems (empty is ALL for VGs specified) []

Application Servers [oracle]

检查和同步HACMP配置

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tools

Add Nodes to an HACMP Cluster
Configure Resources to Make Highly Available
Configure HACMP Resource Groups
Verify and Synchronize HACMP Configuration
Display HACMP Configuration

Cluster Description of Cluster: ahdcc_cluster
Cluster Security Level: Standard
There are 2 node(s) and 2 network(s) defined
NODE jmgrdb1:

```
Network net_ether_01
  jmgrdb1_svc           192.168.109.15
  jmgrdb1_boot         192.168.168.13
  jmgrdb1_standby      172.16.10.1
```

NODE jmgrdb2:

```
Network net_ether_01
  Jmgrdb1_svc          192.168.109.15
  Jmgrdb2_boot         192.168.168.14
  jmgrdb2_standby     172.16.10.2
```

Resource Group jmgrdb1_res

```
Behavior                cascading
Participating Nodes     jmgrdb1 jmgrdb2
Service IP Label        jmgrdb1_svc
```

验证同步群集

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tools

Add Nodes to an HACMP Cluster
Configure Resources to Make Highly Available
Configure HACMP Resource Groups
Verify and Synchronize HACMP Configuration
Display HACMP Configuration
(选择自动修正错误)

测试群集

- (1) 首先启动 CXGRDB1 节点 hacmp, 然后启动 CXGRDB2 节点 hacmp。

```
smitty hacmp
```

```
Initialization and Standard Configuration
```

```
Extended Configuration
```

```
System Management (C-SPOC)
```

```
Problem Determination Tools
```

```
Manage HACMP Services
```

```
HACMP Communication Interface Management
```

```
HACMP Resource Group and Application Management
```

```
HACMP Log Viewing and Management
```

```
HACMP File Collection Management
```

```
HACMP Security and Users Management
```

```
HACMP Logical Volume Management
```

```
HACMP Concurrent Logical Volume Management
```

```
HACMP Physical Volume Management
```

```
Configure GPFS
```

```
Start Cluster Services
```

```
Stop Cluster Services
```

```
Show Cluster Services
```

```
Open a SMIT Session on a Node
```

- (2) smitty clstop 停止 CXGRDB1 上的 hacmp, 以 take over 方式释放资源。我们可以看到 CXGRDB2 节点将 CXGRDB1 的网络资源和 VG 资源接管。
- (3) 重新启动 CXGRDB1 节点的 cluster, 资源应自动切回到 CXGRDB1

添加tty心跳网络

- (1) 建立 tty 设备

```
smitty tty→add a tty, 选择 tty 类型 tty rs232 及 adaper
```

- (2) 添加心跳网络

```
smitty hacmp
```

```
Initialization and Standard Configuration
```

```
Extended Configuration
```

```
System Management (C-SPOC)
```

```
Problem Determination Tool
```

```
Extended Configuration
```

```
Extended Topology Configuration
```

```
Configure HACMP Networks
```

Add a Network to the HACMP Cluster

Rs232

* Network Name [jmgrdb1_tty]
* Network Type rs232

Extended Configuration

Extended Topology Configuration

Configure HACMP Communication Interfaces/Devices

Add Discovered Communication Interfaces and Devices

在下拉列表中选择

jmgrdb1_tty /dev/tty0

jmgrdb2_tty /dev/tty0

加入到串行网络资源中

同步资源

Initialization and Standard Configuration

Verify and Synchronize HACMP Configuration

测试心跳

在1号机上 cat /etc/hosts > /dev/tty0

在2号机上 cat </dev/tty0, 应该能够看到1号机的 hosts 内容

添加永久IP

smitty hacmp

Initialization and Standard Configuration

Extended Configuration

System Management (C-SPOC)

Problem Determination Tools

Discover HACMP-related Information from Configured Nodes

Extended Topology Configuration

Extended Resource Configuration

Extended Cluster Service Settings

Extended Event Configuration

Extended Performance Tuning Parameters Configuration

Security and Users Configuration

Snapshot Configuration

Export Definition File for Online Planning Worksheets

Configure an HACMP Cluster

Configure HACMP Nodes

Configure HACMP Sites

Configure HACMP Networks

Configure HACMP Communication Interfaces/Devices
Configure HACMP Persistent Node IP Label/Addresses
Configure HACMP Global Networks
Configure HACMP Network Modules
Configure Topology Services and Group Services
Show HACMP Topology

分别为两台机器加入永久 IP 地址。

第九章 测试 HACMP

节点接管测试

1) 启动主机及备机的 HACMP 软件

注意：先启动主机上的 HA，确认主机完全启动完毕后，再启动备机上的 HA。以避免资源争用现象。

2) 在主机：

- 用指令“netstat -i”检查 IP address, 确认主网卡使用 service IP address, 备用网卡使用 standby IP address.

用指令“lsvg -o”检查 vary on 的 VG.

```
# lsvg -o
```

```
rootvg
```

```
datavg （若该分区使用共享卷组 datavg）
```

- 确认应用正常启动。（测试 HA 阶段暂不进行此项测试）

3) 在备机：

- 用指令“netstat -i”检查 IP address, 确认主网卡使用 service IP address, 备用网卡使用 standby IP address.

用指令“lsvg -o”检查 vary on 的 VG.

```
# lsvg -o
```

```
rootvg
```

4) 在主机：

- 用指令“smitty clstop”，选择 Takeover 方式停止主机的 HACMP 并使备机接管。（另一种方式是直接按主机的 reset 键使其 reboot.）

5) 在备机：

- 等待接管过程完全结束.

用指令“netstat -i”检查 IP address, 确认主机 的 service IP address 已被备用网卡接管.

用指令“lsvg -o”检查 vary on 的 VG.

```
# lsvg -o
```

rootvg

datavg (若该分区使用共享卷组 datavg)

- 确认应用程序正常启动 (同前) .

6) 在主机:

- 启动机器及 HACMP
- 观察主机恢复正常.

网卡接管测试

1) 启动主机及备机的 HACMP 软件

2) 在主机:

- 用指令 "netstat -i" 检查 IP address, 确认主网卡使用 service IP address, 备用网卡使用 standby IP address.
- 用指令 "lsvg -o" 检查 vary on 的 VG.
- 确认属于主机的应用正常.

3) 在备机:

- 用指令 "netstat -i" 检查 IP address, 确认主网卡使用 service IP address, 备用网卡使用 standby IP address.
- 用指令 "lsvg -o" 检查 vary on 的 VG.
- 确认属于备机的应用正常 (如果存在).

4) 在主机:

- 拔掉 service 网卡接线.
- 等待网卡接管完毕.
- 用指令 "netstat -i" 检查 IP address, 确认主网卡 service IP address 被 standby 网卡接管.
- 确认属于主机的应用正常.
- 恢复 service 网卡接线, 这时 service 地址仍在原来的 Standby 网卡上.

修改网关

一、删除 odm 库中的老的网关 (172.9.66.1)

1、odmget CuAt | grep 172.19.66.1

2、odmget -q "value =net, -hopcount, 0, , 0, 172.19.66.1" CuAt

3、odmdelete -q "value =net, -hopcount, 0, , 0, 172.19.66.1" -o CuAt

二、smitty tcpip

选择最小化配置, 加入或修改默认网关

--尚需加入 weblogic 安装及双机配置, GRV4.0 系统的部署, 数据库表空间的划分

第十章 Weblogic 安装配置

1、系统配置

Aix5.3-03, Weblogic813

2、安装准备

建立 bea 组

Smitty group→add a group

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

```
[Entry]
Fields]
Group NAME [bea]
Group ID [600]
#
ADMINISTRATIVE group? true
+
USER list []
+
ADMINISTRATOR list []
+
Projects []
+
```

建立 weblogic 用户

Smitty user →add a user

```
User NAME weblogic
User ID [600]
#
ADMINISTRATIVE USER? true
+
Primary GROUP [bea]
+
Group SET [bea,system]
+
ADMINISTRATIVE GROUPS []
+
ROLES []
+
```

```

Another user can SU TO USER? true
+
SU GROUPS [ALL]
+
HOME directory
[/home/weblogic]
Initial PROGRAM [usr/bin/ksh]
User INFORMATION []
EXPIRATION date (MMDDhhmmyy) [0]
Is this user ACCOUNT LOCKED? false
+
User can LOGIN? true
+
User can LOGIN REMOTELY(rsh,tn,rlogin)? true
+
Allowed LOGIN TIMES []
Number of FAILED LOGINS before [0]
#
user account is locked
Login AUTHENTICATION GRAMMAR
[compat]
Valid TTYs [ALL]
Days to WARN USER before password expires [0]
#
Password CHECK METHODS []
Password DICTIONARY FILES []
NUMBER OF PASSWORDS before reuse [0]
#
WEEKS before password reuse [0]
#
Weeks between password EXPIRATION and LOCKOUT [-1]
Password MAX. AGE [0]
#
Password MIN. AGE [0]
#
Password MIN. LENGTH [0]
#
Password MIN. ALPHA characters [0]
#
Password MIN. OTHER characters [0]
#
Password MAX. REPEATED characters [8]
#
Password MIN. DIFFERENT characters [0]

```

```

#
Password REGISTRY [files]
Soft FILE size [-1]
#
Soft CPU time [-1]
Soft DATA segment [-1]
#
Soft STACK size [-1]
#
Soft CORE file size [-1]
#
Hard FILE size []
#
Hard CPU time []
Hard DATA segment []
#
Hard STACK size []
#
Hard CORE file size []
#
File creation UMASK [22]
AUDIT classes []
+
TRUSTED PATH? nosak
+
PRIMARY authentication method [SYSTEM]
SECONDARY authentication method [NONE]
Projects []

```

建立文件系统 weblogic_install (2G)用于存放安装文件 smitty fs
建立文件系统 weblogic 作为 weblogic 安装目录

编辑.profile 文件，加入环境变量

PATH=/usr/java14/bin:\$PATH; export PATH

CLASSPATH=/oracle/product/9.2.0/jdbc/lib;export CLASSPATH

安装前注意/weblogic,/weblogic_install 的权限。

插入 weblogic 安装光盘

```
mount -v cdrfs -o ro /dev/cd0 /mnt
```

```
su -weblogic
```

```
cd /mnt
```

```
java -pj_server813_generic.jar
```

开始安装 weblogic

选择 home 目录为/weblogic，完全安装

安装完成后配置 domain

```
su - weblogic
```

```
cd /weblogic/weblogic81/common/bin
```

```
./config.sh
```

第十一章 **Weblogic** 加入 HA 配置