Table des matières

Standalone 3
Règles de base 3
Créer un VG (Volume Group) 3
Créer un filesystem 3
Augmenter un filesystem 4
Supprimer un filesystem 4
Créer un rawdevice Sybase 4
Supprimer un rawdevice Sybase 4
Réduire un FS 5
Cluster 5
Règles de base 5
Utilisation du PVG 5
Création d'un filesystem 5
Augmentation d'un filesystem 6
Suppression d'un filesystem 6
Création d'un rawdevice Sybase 7
Suppression d'un rawdevice Sybase 7
Troubleshooting 8
Perte de fonctionnalités 8
Supprimer le mode cluster d'un VG 8
Tips 8
fsadm: ERROR: V-3-20340: attempt to resize <volume> failed with errno 28 8
/usr/contrib/bin/mapdsk 8
Renommer un VG 10

Standalone

Règles de base

- Le volume groupe vg00 doit être réservé pour le système.
- De préférence, on ne crée pas de volume logique (LV) pour des logs sur un disque où il y a des datas.
- On spécifie toujours le disque sur lequel on étend un LV ainsi que son miroir (s'il y a lieu d'être).

Créer un VG (Volume Group)

On prépare les disques :

```
root@server2835:/$ pvcreate /dev/rdsk/c6t8d0
Physical volume "/dev/rdsk/c6t8d0" has been successfully created.
root@server2835:/$ pvcreate /dev/rdsk/c8t8d0
Physical volume "/dev/rdsk/c8t8d0" has been successfully created.
```

On prépare la création du VG :

```
root@server2835:/dev$ ls -l */group*
crw-------   1 root       sys         64 0x010000 Mar 26  2007 alt_vg00/group
crw-r-----   1 root       sys         64 0x000000 Mar 26  2007 vg00/group

root@server2835:/dev$ mkdir extvg01

root@server2835:/dev$ mknod extvg01/group c 64 0x020000

root@server2835:/dev$ ls -l */group*
crw-------   1 root       sys         64 0x010000 Mar 26  2007 alt_vg00/group
crw-r--r--   1 root       sys         64 0x020000 Jan 22 11:20 extvg01/group
crw-r-----   1 root       sys         64 0x000000 Mar 26  2007 vg00/group
```

On créé le VG :

```
root@server2835:/dev$ vgcreate -l 255 -p 128 -e 4096 -s 64 /dev/extvg01 /dev/dsk/c6t8d0
Volume group "/dev/extvg01" has been successfully created.
Volume Group configuration for /dev/extvg01 has been saved in /etc/lvmconf/extvg01.conf
root@server2835:/dev$ vgextend /dev/extvg01 /dev/dsk/c68t8d0
```

- -l : nombre MAX de LVs
- -p : nombre max de disques
- -e : nombre max de PEs par disques
- -s : taille de la PE (en Mo)

Créer un filesystem

- Initialisation du LV :

```
lvcreate -n lv_name /dev/vg_name
lvcreate -r N -n lv_name /dev/vg_name (avec Symmetrix)
```

- Définition de la taille du LV :

```
lvextend -L "taille en MO" /dev/vg_name/lv_name  /dev/dsk/cxtydz
```

- Création du miroir (si nécessaire)

```
lvextend -m 1 /dev/vg_name/lv_name /dev/dsk/cxtydz_miroir
```

- Création du FS :

```
newfs -F vxfs -o largefiles /dev/vg_name/rlv_name
```

- Création du point du point de montage :

```
mkdir <point de montage>
```

- Ajout de l'entrée dans le fichier /etc/fstab :

```
/dev/vg_name/lv_name <point de montage> vxfs delaylog 0 2
```

- Montage du FS :

```
mount  <point de montage>
```

- Vérification :

```
bdf
```

- Vérification des droits :

```
ls -l <point de montage>
```

Augmenter un filesystem

- Définition de la nouvelle taille du LV :

```
lvextend -L "taille en MO"  /dev/vg_name/lv_name  /dev/dsk/cxtydz /dev/dsk/cxtydz_miroir
avec cxtydz primaire et cxtydz_miroir le disque miroir (si nécessaire)
```

- Augmentation du FS :

```
fsadm -F vxfs -b (taille en MO*1024) <point de montage>
```

- Vérification :

```
bdf
```

Supprimer un filesystem

- Démontage du FS :

```
umount <point de montage>
```

- Suppression du LV :

```
lvremove /dev/vg_name/lv_name
```

- Suppression de la ligne concernée dans le fichier /etc/fstab
- Suppression du point de montage

Créer un rawdevice Sybase

- Initialisation du LV :

```
lvcreate -n lv_name /dev/vg_name
lvcreate -r N -n lv_name /dev/vg_name (avec Symmetrix)
```

- Définition de la taille du LV :

```
lvextend -L "taille en MO" /dev/vg_name/lv_name /dev/dsk/cxtydz
```

- Création du miroir (si nécessaire)

```
lvextend -m 1 /dev/vg_name/lv_name /dev/dsk/cxtydz_miroir
```

- Ajout des droits SYBASE :

```
chown <user sybase>:<group sybase> /dev/vg_name/r*
```

- Ajout du lien symbolique :

```
ln -s /dev/vg_name/rlv_name /sybase_data/<instance sybase>/rawdevice/<sybase name>
```

Supprimer un rawdevice Sybase

- Suppression du LV :

```
lvremove /dev/vg_name/lv_name
```

Réduire un FS

Avec vxfs on peut réduire un FS à chaud (pas spécifique à HP-UX). On réduit **d'abord** le FS avec *fsadm* puis on réduit **ensuite** le LV avec *lvreduce*. Il faut être précis sur les tailles utilisées, ici on passe le FS à de 6 Go à 4 Go par ex. :

- Réduction du FS

```
fsadm -b $((4000*1024)) /toto
```

- Réduction du LV

```
lvreduce -L 4000 /dev/vg_titi/lv_toto
```

Cluster

Règles de base

- Le volume groupe vg00 doit être réservé pour le système
- On ne crée jamais de volume logique pour des logs sur un disque où il y a des datas
- On spécifie toujourss le disque sur lequel on étend un LV ainsi que le miroir
- On doit exporter la map du vg si on crée/supprime un LV
- Lors d'un ajout d'un raw device, ne pas oublier de lui attribuer les bons droits

Utilisation du PVG

Lorsque des LVs sont mirrorés sur des disques distants (du SAN inter sites par ex.) il faut être vigilant lors de la création ou l'augmentation des filesystems pour ne pas endommager le miroir. Si on mirrore mal (c'est-à-dire en spécifiant des mauvais disques) on se retrouve avec un mirroir incohérent et on a donc des LVs qui contiennent des PEs sur des disques situés un même site. En cas de crash d'une baie ou d'un incendie, on perd donc les disques, les données et le miroir. D'où l'intérêt de mirrorer avec soin.

Par ex. SAM[1] ne sait pas déterminer où sont situés les disques physiquement. Lorsqu'on lui demande d'étendre un LV il prend les premiers disques libres. Et la plupart du temps on se retrouve un miroir incorrect car les *n* disques utilisés peuvent potentiellement se situés sur un même site. On peut rétablir la situation à coups de *lvextend -m 0*, *lvextend -m 1* et *pvmove* mais c'est long et moins on a d'espace disque de libre et plus c'est fastidieux à faire.

Si beaucoup de personnes disposent des droits *root* sur une machine on a donc plus de risques d'obtenir un LVM moisi. Tout le monde ne fait pas forcément attention lors des manipulations LVM. Ou bien d'autres personnes ne savent pas faire ou pensent savoir faire. Pour éviter ça le *PVG* permet de réduire les risques.

Ci-dessous un résumé des étapes pour mettre en place le *PVG* :

- Création du fichier */etc/lvmpvg* :

```
VG /dev/vg_data1
PVG PV_siteA
/dev/dsk/c1t6d0
/dev/dsk/c2t6d0
PVG PV_siteB
/dev/dsk/c3t6d0
/dev/dsk/c4t6d0
```

Pour le VG *vg_data1* on a 2 *PVGs* composés des disques c[1,2]t6d0 d'une part et c[3,4]t6d0 d'autre part. C'est à dire que lorsqu'on utilise *lvextend* ou *lvreduce* le système sait qu'il doit travailler avec 1 disque de chaque PVG.

- Modification des paramètres des LVs :

On utilise l'option *PVG-strict* de la commande *lvchange* (c'est le même principe que le *strict / superstrict* sous AIX) :

```
lvchange -s /dev/vg_data1/lv_data1
```

! Attention, il faut passer le LV en PVG-strict avant de lui ajouter un mirroir. Ce qui revient à faire, dans l'ordre, : lvcreate, lvchange puis lvextend.
!! Un *vgexport* met à vide le fichier *etc/lvmpvg*, il faut penser à le backuper avant de faire des export/import de VG.

C'est tout. Maintenant un simple *lvextend* sans spécifier de disque fonctionnera en utilisant un disque de chaque PVG. De cette manière on aura un mirroring propre. Par ailleurs on peut également utiliser la méthode en spécifiant manuellement les disques. Enfin quand on rajoute ou supprime un ou plusieurs disques d'un VG il faut penser à mettre à jour le */etc/lvmpvg* en conséquence.

Création d'un filesystem

- Initialisation du LV :

```
lvcreate -n lv_name /dev/vg_name
lvcreate -r N -n lv_name /dev/vg_name (avec Symmetrix)
```

- Définition de la taille du LV :

```
lvextend -L "taille en MO" /dev/vg_name/lv_name /dev/dsk/cxtydz
```

- Création du miroir (si nécessaire)

```
lvextend -m 1 /dev/vg_name/lv_name /dev/dsk/cxtydz_miroir
```

- Initialisation du filesystem :

```
newfs -F vxfs -o largefiles /dev/vg_name/rlv_name
```

- Mise à jour du fichier control.sh
- Ouvrir avec vi le fichier /etc/cmcluster/package_name/control.sh
- Se rendre à la derniere ligne du type LV[xx]...
- La dupliquer sur la ligne en dessous en l'ajoutant
- Sur la ligne dupliquée ajouter 1 a tous les index et renseigner les champs LV[xx]= et FS[xx]
    - Exemple :

LV[26]="/dev/vg_package2/lv_ubix_data4"; FS[26]="/oracle_data/disk4/UBXP"; FS_MOUNT_OPT[26]=""
LV[27]="/dev/vg_package2bis/lv_gispro"; FS[27]="/appli/package2/GISPROXY"; FS_MOUNT_OPT[27]=""

- Création du point du point de montage :

```
mkdir <point de montage>
```

- Monter le filesystem en copiant/collant la ligne du control.sh

```
mount  <lv_name>  <point de montage>
```

- Appliquer les droits sur le filesystem

```
chown user_name:group_name <point de montage>
```

- Mise à jour de la map du vg

```
vgexport -p -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
```

- Recopie de la map sur les autres nœuds du cluster via ftp (répertoire équivalent)
- Mise à jour du vg sur les autres nœuds :

```
vgexport /dev/vg_name
mkdir /dev/vg_name
chmod 755 /dev/vg_name
mknod /dev/vg_name/group c 64 même_mineur_du_vg_source (ex : 0x010000)
vgimport -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
chown <user sybase>:<group sybase> /dev/vg_name/r*
mkdir <point de montage>
```

- Reproduire les modifications appliquées au fichier control.sh de l'autre noeud :
- Ouvrir avec vi le fichier /etc/cmcluster/package_name/control.sh
- Se rendre à la dernière ligne du type LV[xx]...
- La dupliquer sur la ligne en dessous
- Sur la ligne dupliquée ajouter 1 a tous les index et renseigner les champs LV[xx]= et FS[xx]

Augmentation d'un filesystem

- Définition de la nouvelle taille du LV :

```
lvextend -L "taille en MO"  /dev/vg_name/lv_name  /dev/dsk/cxtydz /dev/dsk/cxtydz_miroir
```

- Augmentation du FS :

```
fsadm -b (taille en MO * 1024) <point de montage>
```

- Vérification :

```
bdf
```

Suppression d'un filesystem

- Démontage du FS :

```
umount <point de montage>
```

- Suppression du point de montage du FS :

```
rmdir <point de montage>
```

- Suppression du LV :

```
lvremove /dev/vg_name/lv_name
```

- Suppression du filesystem dans le fichier control.sh :
- Ouvrir avec vi le fichier /etc/cmcluster/package_name/control.sh
- Rechercher à la ligne du type LV[xx]...contenant les informations concernant le filesystem
- Supprimer la ligne et refaire la numérotation des lignes suivantes (attention 3 champs par ligne)
- Exemple :

LV[26]="/dev/vg_package2/lv_ubix_data4"; FS[26]="/oracle_data/disk4/UBXP"; FS_MOUNT_OPT[26]=""
LV[27]="/dev/vg_package2bis/lv_gispro"; FS[27]="/appli/package2/GISPROXY"; FS_MOUNT_OPT[27]=""

- Mise à jour de la map du vg :

```
vgexport -p -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
```

- Recopie de la map sur les autres nœuds du cluster via ftp (répertoire équivalent)
- Mise à jour du vg sur les autres nœuds
- Reporter les modification du control.sh du premier nœud sur les autres nœuds

- Supprimer le point de montage :

```
rmdir <point de montage>
vgexport /dev/vg_name
mkdir /dev/vg_name
chmod 755 /dev/vg_name
mknod /dev/vg_name/group c 64 même_mineur_du_vg_source (ex : 0x010000)
vgimport -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
chown <user sybase>:<group sybase> /dev/vg_name/r*
```

Création d'un rawdevice Sybase

- Initialisation du LV :

```
lvcreate -n lv_name /dev/vg_name
lvcreate -r N -n lv_name /dev/vg_name (avec Symmetrix)
```

- Définition de la taille du LV :

```
lvextend -L "taille en MO" /dev/vg_name/lv_name /dev/dsk/cxtydz
```

- Création du miroir (si nécessaire) :

```
lvextend -m 1 /dev/vg_name/lv_name /dev/dsk/cxtydz_miroir
```

- Ajout des droits SYBASE :

```
chown <user sybase>:<group sybase> /dev/vg_name/r*
```

- Ajout du lien symbolique pour les DBA :

```
ln -s /dev/vg_name/rlv_name /sybase_data/<instance sybase>/rawdevice/<sybase name>
```

- Mise à jour de la map du vg.

Suppression d'un rawdevice Sybase

- Suppression du LV :

```
lvremove /dev/vg_name/lv_name
```

- Mise à jour de la map du vg :

```
vgexport -p -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
```

- Recopie de la map sur les autres noeuds du cluster via ftp (répertoire équivalent)

- Mise à jour du vg sur les autres noeuds :

```
vgexport /dev/vg_name
mkdir /dev/vg_name
chmod 755 /dev/vg_name
mknod /dev/vg_name/group c 64 même_mineur_du_vg_source (ex : 0x010000)
vgimport -s -m /etc/cmcluster/Packagex/vg_name.map /dev/vg_name
chown <user sybase>:<group sybase> /dev/vg_name/r*
```

Troubleshooting

Perte de fonctionnalités

On peut avoir un souci de montage de FS avec certaines options comme *convosync* ou *mincache*. Ou alors être dans l'impossibilité d'augmenter un FS avec *fsadm*. Il faut vérifier la license VxFS avec *vxlicense -p* :

```
vrts:vxlicense: INFO: Feature name: HP_OnlineJFS [50]
vrts:vxlicense: INFO: Number of licenses: 1 (non-floating)
vrts:vxlicense: INFO: Expiration date: Sun Jun 24 10:00:00 2007 (148.1 days ago)
vrts:vxlicense: INFO: Release Level: 22
vrts:vxlicense: INFO: Machine Class: All
vrts:vxlicense: INFO: Site ID: 0
```

Il faut dans ce cas patcher le **OnLineFS** avec la version la plus à jour.

Supprimer le mode cluster d'un VG

```
vgchange -c n /dev/vg_apps
```

Tips

fsadm: ERROR: V-3-20340: attempt to resize <volume> failed with errno 28

Soit on augmente le FS octet par octet avec *fsadm -b* soit on vérifie la fragmentation du FS :

```
root@server:/ fsadm -E /some/fs
fsadm: /etc/default/fs is used for determining the file system type
  Extent Fragmentation Report
        Total    Average      Average     Total
        Files    File Blks    # Extents   Free Blks
        41818          4           4       57741
    blocks used for indirects: 319
    % Free blocks in extents smaller than 64 blks: 100.00
    % Free blocks in extents smaller than  8 blks: 100.00
    % blks allocated to extents 64 blks or larger: 0.86
    Free Extents By Size
           1:      57741           2:          0           4:          0
           8:          0          16:          0          32:          0
          64:          0         128:          0         256:          0
         512:          0        1024:          0        2048:          0
        4096:          0        8192:          0       16384:          0
       32768:          0       65536:          0      131072:          0
      262144:          0      524288:          0     1048576:          0
     2097152:          0     4194304:          0     8388608:          0
    16777216:          0    33554432:          0    67108864:          0
   134217728:          0   268435456:          0   536870912:          0
  1073741824:          0  2147483648:          0
```

Plus le nombre de blocs contenus dans les extents les plus petits est important, plus le FS est fragmenté, ce qui est le cas ici.

Pour défragmenter, à chaud :

```
fsadm -e -d /some/fs
```

/usr/contrib/bin/mapdsk

Exemple de sortie de la commande :

```
root@server:/ mapdsk
Scanning hardware, please wait...
/dev/rdsk/c0t0d0 is a CDROM device
/dev/dsk/c1t0d0s2: cannot open or is not a character device file
/dev/dsk/c1t0d0s3: cannot open or is not a character device file
/dev/dsk/c1t1d0s2: cannot open or is not a character device file
```

```
/dev/dsk/c1t1d0s3: cannot open or is not a character device file

/dev/rdsk/c2t0d0                UNUSED
/dev/rdsk/c2t4d7                UNUSED
/dev/rdsk/c2t5d0        -       CONFIG in /dev/vg_Package1
/dev/rdsk/c2t5d1                UNUSED
/dev/rdsk/c2t5d2                UNUSED
/dev/rdsk/c2t5d3                UNUSED
/dev/rdsk/c2t5d4                UNUSED
/dev/rdsk/c2t5d5                UNUSED
/dev/rdsk/c2t5d6                UNUSED
/dev/rdsk/c2t5d7                UNUSED
/dev/rdsk/c2t6d0        -       CONFIG in /dev/vg_Package2
/dev/rdsk/c2t6d1                UNUSED
/dev/rdsk/c2t6d2                UNUSED
/dev/rdsk/c2t6d3                UNUSED
/dev/rdsk/c2t6d4                UNUSED
/dev/rdsk/c2t6d5                UNUSED
/dev/rdsk/c2t6d6                UNUSED
/dev/rdsk/c2t6d7                UNUSED
/dev/rdsk/c2t7d0                UNUSED
/dev/rdsk/c2t7d1        -       NOCONF in /dev/vg00
/dev/rdsk/c2t7d2                UNUSED
/dev/rdsk/c2t7d3        -       CONFIG in /dev/vg_tngserveur
/dev/rdsk/c2t7d4        -       CONFIG in /dev/vg_apps
/dev/rdsk/c8t0d0                UNUSED
/dev/rdsk/c8t4d7        -       CONFIG in /dev/vg_tngserveur
/dev/rdsk/c8t5d0        -       CONFIG in /dev/vg_Package1
/dev/rdsk/c8t5d1                UNUSED
/dev/rdsk/c8t5d2                UNUSED
/dev/rdsk/c8t5d3                UNUSED
/dev/rdsk/c8t5d4                UNUSED
/dev/rdsk/c8t5d5                UNUSED
/dev/rdsk/c8t5d6                UNUSED
/dev/rdsk/c8t5d7                UNUSED
/dev/rdsk/c8t6d0        -       CONFIG in /dev/vg_Package2
/dev/rdsk/c8t6d1                UNUSED
/dev/rdsk/c8t6d2                UNUSED
/dev/rdsk/c8t6d3                UNUSED
/dev/rdsk/c8t6d4                UNUSED
/dev/rdsk/c8t6d5                UNUSED
/dev/rdsk/c8t6d6                UNUSED
/dev/rdsk/c8t6d7                UNUSED
/dev/rdsk/c8t7d0                UNUSED
/dev/rdsk/c8t7d1                UNUSED
/dev/rdsk/c8t7d2                UNUSED
/dev/rdsk/c8t7d3                UNUSED
/dev/rdsk/c8t7d4        -       CONFIG in /dev/vg_apps
/dev/rdsk/c4t0d0                UNUSED
/dev/rdsk/c4t4d7                UNUSED
/dev/rdsk/c4t5d0        -       CONFIG in /dev/vg_Package1
/dev/rdsk/c4t5d1                UNUSED
/dev/rdsk/c4t5d2                UNUSED
/dev/rdsk/c4t5d3                UNUSED
/dev/rdsk/c4t5d4                UNUSED
/dev/rdsk/c4t5d5                UNUSED
/dev/rdsk/c4t5d6                UNUSED
/dev/rdsk/c4t5d7                UNUSED
/dev/rdsk/c4t6d0        -       CONFIG in /dev/vg_Package2
/dev/rdsk/c4t6d1                UNUSED
/dev/rdsk/c4t6d2                UNUSED
/dev/rdsk/c4t6d3                UNUSED
/dev/rdsk/c4t6d4                UNUSED
/dev/rdsk/c4t6d5                UNUSED
/dev/rdsk/c4t6d6                UNUSED
/dev/rdsk/c4t6d7                UNUSED
/dev/rdsk/c4t7d0                UNUSED
/dev/rdsk/c4t7d1        -       NOCONF in /dev/vg00
/dev/rdsk/c4t7d2                UNUSED
/dev/rdsk/c4t7d3        -       CONFIG in /dev/vg_tngserveur
/dev/rdsk/c4t7d4        -       CONFIG in /dev/vg_apps
/dev/rdsk/c6t0d0                UNUSED
/dev/rdsk/c6t4d7        -       CONFIG in /dev/vg_tngserveur
/dev/rdsk/c6t5d0        -       CONFIG in /dev/vg_Package1
/dev/rdsk/c6t5d1                UNUSED
/dev/rdsk/c6t5d2                UNUSED
```

```
/dev/rdsk/c6t5d3                UNUSED
/dev/rdsk/c6t5d4                UNUSED
/dev/rdsk/c6t5d5                UNUSED
/dev/rdsk/c6t5d6                UNUSED
/dev/rdsk/c6t5d7                UNUSED
/dev/rdsk/c6t6d0        -       CONFIG in /dev/vg_Package2
/dev/rdsk/c6t6d1                UNUSED
/dev/rdsk/c6t6d2                UNUSED
/dev/rdsk/c6t6d3                UNUSED
/dev/rdsk/c6t6d4                UNUSED
/dev/rdsk/c6t6d5                UNUSED
/dev/rdsk/c6t6d6                UNUSED
/dev/rdsk/c6t6d7                UNUSED
/dev/rdsk/c6t7d0                UNUSED
/dev/rdsk/c6t7d1                UNUSED
/dev/rdsk/c6t7d2                UNUSED
/dev/rdsk/c6t7d3                UNUSED
/dev/rdsk/c6t7d4        -       CONFIG in /dev/vg_apps
```

Au lieu de se contenter de lire les fichiers lvm, elle interroge aussi la pvda des disques, voici ce que ça donne sur un cluster:

(on teste l'ajout d'un disque sur un VG lié à un package)

server1		server2		commande
/dev/rdsk/c6t7d3	UNUSED	/dev/rdsk/c2t7d3	UNUSED	**pvcreate /dev/rdsk/c2t7d3**
/dev/rdsk/c6t7d3	NOCONF in /dev/vg00	/dev/rdsk/c2t7d3	NOCONF in /dev/vg00	**vgextend /dev/vg_Package2 /dev/dsk/c2t7d3**
/dev/rdsk/c6t7d3	NOCONF in /dev/vg_Package2	/dev/rdsk/c2t7d3	CONFIG in /dev/vg_Package2	**export import des maps**
/dev/rdsk/c6t7d3	CONFIG in /dev/vg_Package2	/dev/rdsk/c2t7d3	CONFIG in /dev/vg_Package2	**vgreduce /dev/vg_Package2 /dev/dsk/c2t7d3**
/dev/rdsk/c6t7d3	ERROR in /dev/vg_Package2	/dev/rdsk/c2t7d3	NOCONF in /dev/vg00	**export import des maps**
/dev/rdsk/c6t7d3	NOCONF in /dev/vg00	/dev/rdsk/c2t7d3	NOCONF in /dev/vg00	**pvremove /dev/rdsk/c2t7d3**
/dev/rdsk/c6t7d3	UNUSED	/dev/rdsk/c2t7d3	UNUSED	

Du coup, on en déduit:

UNUSED	On peut utiliser le disque on est sur de sa dispo
NOCONF in /dev/vg00	Sur un noeud du cluster, le pvcreate a été fait, mais le disque n'est affecté à aucun VG
CONFIG in /dev/<vg>	Le Disque est dans un VG, la map est OK
NOCONF in /dev/<vg>	Le disque est dans le VG, la map n'est pas à jour export/import de map à effectuer
ERROR in /dev/<vg>	Le disque n'est plus dans le VG, la map n'est pas à jour export/import de map à effectuer

Lors des tests, on n'a pas regargé le comportement des disques alternés.

Renommer un VG

- On génère la map

```
vgexport -s -p -m /tmp/pkg3.map /dev/vg_Package3
```

- On demonte le vg

```
vgchange -a n /dev/vg_Package3
```

- On supprime le vg

```
vgexport /dev/vg_Package3
```

- On créé l'enveloppe du vg

```
mkdir /dev/vg_Package8
chmod 755 /dev/vg_Package8
mknod /dev/vg_Package8/group c 64 0x070000
```

- On importe la map dans la nouvelle enveloppe

```
vgimport -m /tmp/pkg3.map -s /dev/vg_Package8
```

- On monte le vg pour check (/!\ si le vg est en mode exclusif, penser à le sortir du cluster: vgchange -c n vg_Package8)

```
vgchange -a y vg_Package8
lsvg vg_Package8
```

Si le VG est sur un cluster, refaire 3 4 5 sur le second noeud.

1)
outil de gestion système d'HP-UX

From:
https://unix-bck.ndlp.info/ - **Where there is a shell, there is a way**

Permanent link:
https://unix-bck.ndlp.info/doku.php/informatique:nix:hp:hpux_lvm

Last update: **2012/02/29 15:49**