

LVM(动态磁盘管理)

- PE (Physical Extend)
- PV (Physical Volume)
- VG (Volume Group)
- LV (Logical Volume)

PV是逻辑卷管理的最基本单位。
VG相当于一个空间池，装一个个PV的。

工作原理

1. 第一步把硬盘格式化成PV物理卷，把硬盘空间划成一个一个的PE
2. 不同的PV加入同一个VG，不同的PV的PE全部进入VG的池内
3. LV基于PE创建，大小为PE的整数倍一个PE最小为4M，组成LV的PE可能来自不同的物理磁盘
4. 至此LV就可以直接格式化后挂载使用了

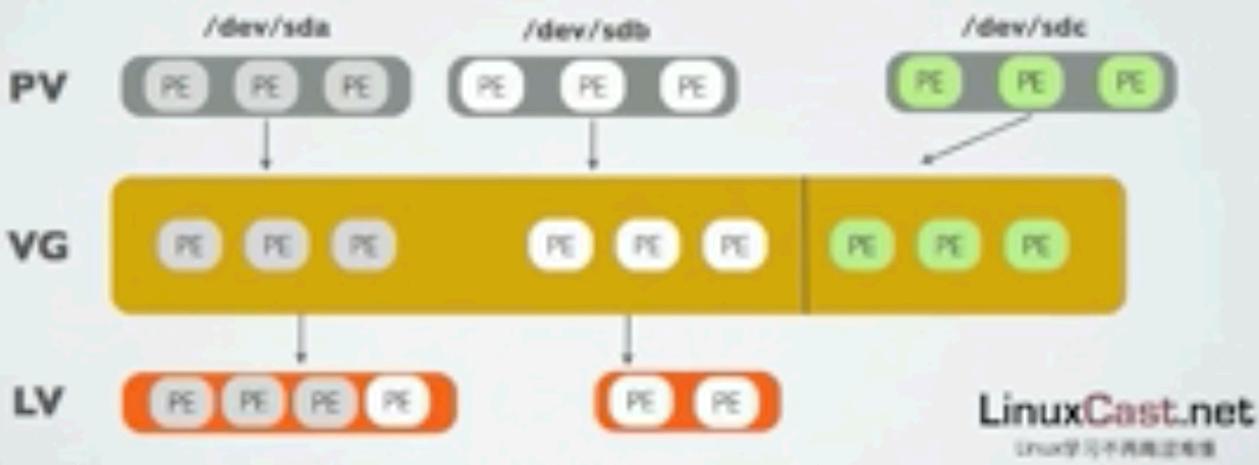
LV拉伸原理

LV的扩充或缩减实际上就是增加或减少组成LV的PE的数量，其过程不丢失原始数据
解释为：

扩充或减少逻辑全LV其实就是从VG空间池中往LV里面加PE或者从LV里减少PE就行了，VG空间不够用的时候就往卷组里加硬盘，从而增加了VG空间池里的PE

LVM

- (1)物理磁盘被格式化为PV，空间被分为一个个PE
- (2)不同的PV加入同一个VG，不同PV的PE全部进入VG的PE池内
- (3)LV基于PE创建，大小为PE的整数倍，组成LV的PE可能来自不同物理磁盘
- (4)LV现在就直接可以格式化后挂载使用了
- (5)LV的扩充缩减实际上就是增加或减少组成该LV的PE的数量。其过程不丢失原始数据



LinuxCast.net
Linux学习不再迷茫地带

创建LVM

1. 将物理磁盘初始化为物理卷

```
pvcreate /dev/sdb /dev/sdc (空格加多个)
```

2. 创建卷组，并将PV加入卷组中

```
vgcreate vg1 /dev/sdb /dev/sdc (vg1是卷组的名字，两个格式化完的物理卷都加入了卷组中)
```

3. 基于卷组创建逻辑卷

```
lvcreate -n lv1 -L 10G vg1 (-n指定逻辑卷名字lv1是逻辑卷名字，-L指定逻辑卷大小，vg1是从这个卷组中拿空间)
```

4. 为创建好的逻辑卷创建文件系统 格式化逻辑卷

```
mkfs.ext4 /dev/vg1/lv1 (文件系统可以是ext4也可以是xfs，默认逻辑卷的路径在dev下的卷组名下面)
```

5. 将格式化好的逻辑卷挂载并使用

```
mount /dev/vg1/lv1 /home (home是创建的目录)
```

查看LVM命令

- * 查看物理卷信息：
`pvdisplay` (详细)
`pvs`
- * 查看卷组信息：
`vgdisplay` (详细)
`vgs`
- * 查看逻辑卷信息：
`lvdisplay` (详细)
`lvs`

创建LVM

1. 将物理磁盘设备初始化为物理卷

```
pvcreate /dev/sdb /dev/sdc
```

2. 创建卷组，并将PV加入卷组中

```
vgcreate linuxcast /dev/sdb /dev/sdc
```

3. 基于卷组创建逻辑卷

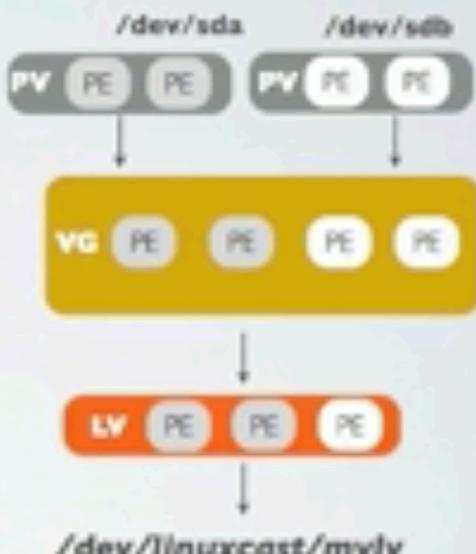
```
lvcreate -n mylv -L 2G linuxcast
```

4. 为创建好的逻辑卷创建文件系统

```
mkfs.ext4 /dev/linuxcast/mylv
```

5. 将格式化好的逻辑卷挂载使用

```
mount /dev/linuxcast/mylv /mnt
```



LinuxCast.net

Linux学习不再遥遥无期

删除LVM

首先umount挂载点

按顺序删除

1. 先删除逻辑卷LV

```
lvremove /dev/vg1/lv1
```

2. 删除卷组VG

```
vgremove vg1
```

3. 删除物理卷PV

```
pvremove /dev/sdb /dev/sdc
```

LVM拉伸一个逻辑卷

1. 保证VG有足够的空闲空间

```
vgdisplay
```

2. 扩充逻辑卷

```
lvextend -L +1G /dev/vg1/lv1
```

3. 查看扩容后的LV大小

```
lvdisplay
```

4. 更新文件系统

```
REHL6: resize2fs /dev/vg1/lv1
```

```
REHL7: xfs_growfs /dev/vg1/lv1
```

5. 查看更新后的文件系统

```
df -hT
```

拉伸一个逻辑卷

逻辑卷的拉伸操作可以在线执行，不需要卸载逻辑卷

1. 保证VG中有足够的空闲空间

```
vgdisplay
```

2. 扩充逻辑卷

```
lvextend -L +1G /dev/linuxcast/mylv
```

3. 查看扩充后LV大小

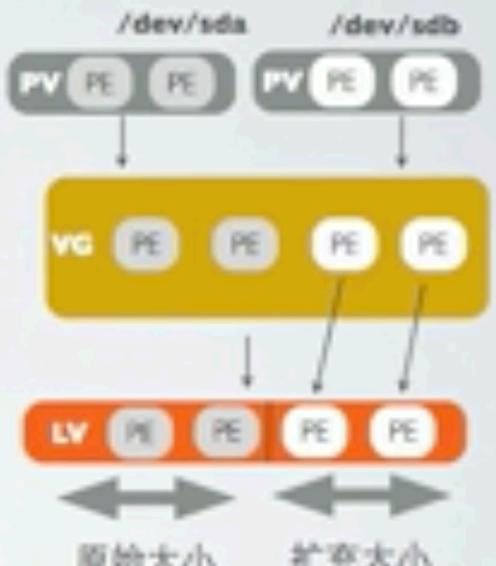
```
lvdisplay
```

4. 更新文件系统

```
resize2fs /dev/linuxcast/mylv
```

5. 查看更新后文件系统

```
df -h
```



LinuxCast.net

LVM拉伸一个卷组

1. 将要添加到VG的硬盘格式化为PV

```
pvcreate /dev/sdd
```

2. 将新的PV添加到指定卷组中

```
vgextend vg1 /dev/sdd
```

3. 查看扩容后VG的大小

```
vgdisplay
```

缩小一个逻辑卷

缩小逻辑卷必须是离线执行的，要先卸载逻辑卷

按顺序操作否则可能造成逻辑卷损坏

1. 卸载已挂载的逻辑卷

```
umount /dev/vg1/lv1
```

2. 缩小文件系统

(会先提示要运行e2fsck检查文件系统)

```
REHL6: resize2fs /dev/vg1/lv1 10G (10G意思是把文件系统缩小为10个G)
```

```
REHL7: xfs_growfs /dev/vg1/lv1 10G
```

3. 缩小LV `lvreduce -L -9G /dev/vg1/lv1` (通常是lvreduce的大小9G要小于resize2fs的大小10G 留下1个G的缓冲 如果大于10G可能造成逻辑卷损坏) 4. 查看缩小后的LV `lvdisplay` 5. 挂载 `mount /dev/vg1/lv1/ /home`

缩小一个逻辑卷

逻辑卷的缩小操作必须离线执行，要卸载逻辑卷

1. 卸载已经挂载的逻辑卷

```
umount /dev/linuxcast/mylv
```

2. 缩小文件系统

(会提示需要运行fsck检查文件系统)

```
resize2fs /dev/linuxcast/mylv 1G
```

3. 缩小LV

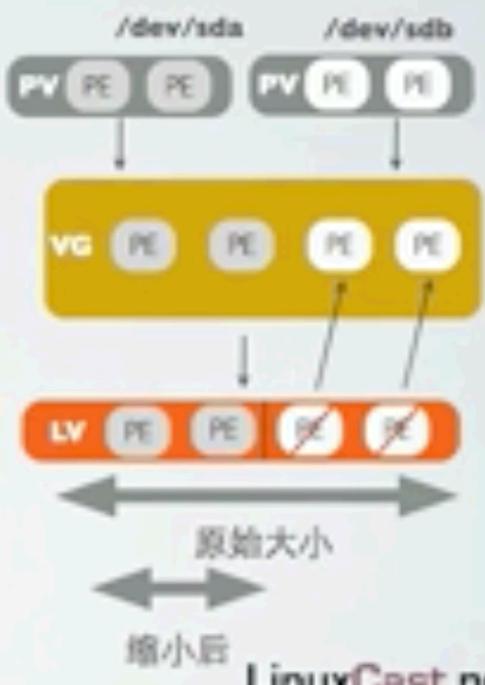
```
lvreduce -L -1G /dev/linuxcast/mylv
```

4. 查看缩小后的LV

```
lvdisplay
```

5. 挂载

```
mount /dev/linuxcast/mylv /mnt
```



缩小一个卷组

缩小之前要保证VG的剩余空间要比拿走的物理卷PV(硬盘)的空间大

1. 将一个PV从指定卷组中移除

```
vgreduce vg1 /dev/sdb
```

2. 查看缩小后的卷组大小

```
vgdisplay
```

3. 查看剩余的PV

```
pvdisplay
```

4. 删除多余的没有卷组的PV

```
pvremove /dev/sdb
```

至此sdb硬盘就从1vm中拿出来了

缩小卷组

1. 将一个PV从制定卷组中移除

```
vgreduce linuxcast /dev/sdd
```

2. 查看缩小后的卷组大小

```
vgdisplay
```

