md:multiple devices

通过md的功能实现raid 是一个内核模块

通过模拟 多个设备模拟成一个设备

dm

device mapper 设备映射

还是将物理设备转换为逻辑设备

这个也是内核模块

dm本身能实现的功能和md有重叠也有不同，比md更强大

dm本身是一个设备映射表

叫dm target：

主要是raid456这几个级别

1就叫mirror 镜像

0 条strip

snapshot 快照

dmsetup也能做raid0和1 不过管理程序不一样

dmsetup过于底层

还有别的功能也用到了dmsetup

LVM 逻辑卷管理器

其实和软raid同样的东西

都被称为metadisk

把底层的组织起来当做一个设备使用

lvm依赖的dm也能实现raid功能

不过现在一般都使用md来实现raid

应用到dm比较有名的项目还应用到了：

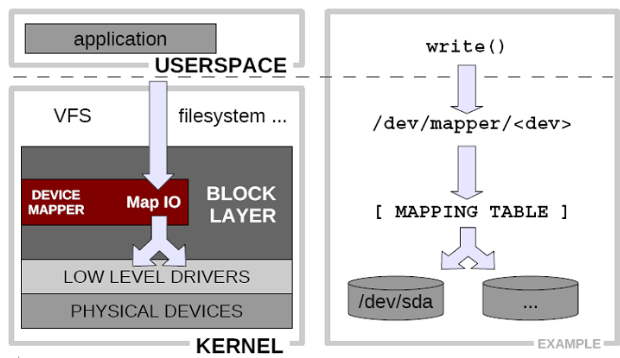
multipath指定访问设备有一个以上的路径

这种多路径实现冗余的功能

这个mulitpath本身依赖的也就是dm

下图的device mapper就是dm

这个dm提供的重要功能就是lvm



lvm实际上是一个软件

是dm的前端组件

dm🡪LVM

md🡪mdadm

lvm用处：

无损扩展缩小

**快照 快照功能实际就是写时复制 COW**

**逻辑卷需要先调整标号：8e**

**然后初始化 初始化使用pv相关命令**

**块设备初始化为物理卷PV**

**多个PV组成为一个VG 卷组volume group**

**这个卷组再分为分区 就是逻辑卷 logical volume**

块设备可以为raid或者分区，只要是个块设备就行

块设备先表示为8e

然后初始化为pv

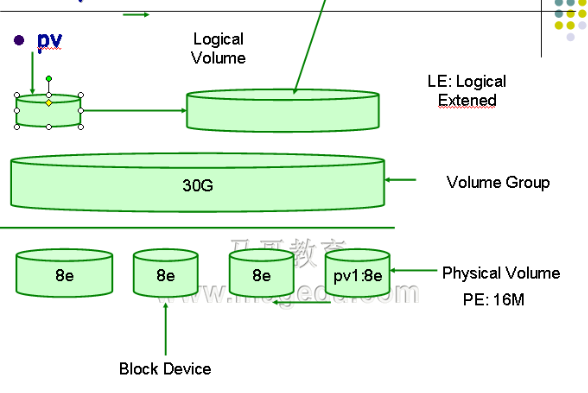
然后组成为vg

但是如果之后缩减的话那么某个pv卸掉的话需要先转移到别的pv

**pe 物理分区单位大小 在创建卷组的时候决定**

逻辑卷就是由一个个pe组成

创建快照必须在这个逻辑卷所在的逻辑卷内创建 快照卷



所有和物理卷相关的命令：

pv开头

pvcreate pvscan pvdisplay(比pvs详细) pvs pvmove pvremove

所有和卷组相关的

vg开头

vgcreate vgscan vgdisplay vgs vgextend vgremove

vgreduce

逻辑卷相关的

lv开头

lvcreat lvscan lvdisplay lvs lvextend lvreduce lvremove

例如创建一个10gVG

2G的lv

那么pv：

2\*5g

1\*10g

简单起见，直接一个10g的pv

新建分区 改标号

pvcreate /dev/sda9

创建物理卷并不难决定pe大小

在卷组创建时决定

vgcreate 默认4m pe大小



lvcreate -L 2G -n mylv1 myvg

-L 指定逻辑卷大小

-n 名字

然后再指定在哪个卷组中建立



使用先格式化



扩展卷组：

先建pv

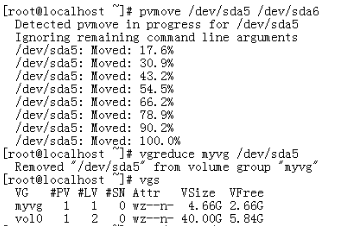


vgextend VGNAME /dev/DEVICE



缩减卷组：

PVmove 把一个上面的移到另一个上面：



然后再vgreduce



然后删除，移除pv



扩展逻辑卷：先扩展物理边界在扩展逻辑边界

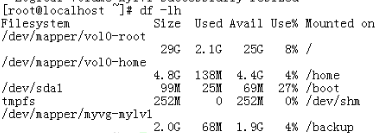
lvextend -L [+]SIZE /dev/VGNAME/LVNAME

大小 指定逻辑卷

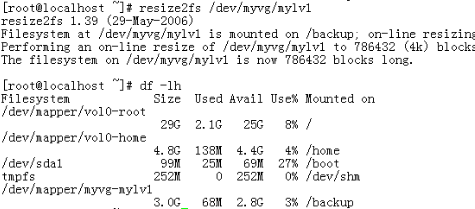
**+1g 扩展1G**

**1g 扩展到1G**





但是发现文件系统没有扩大，因为文件系统边界没有扩大，需要resizee2fs



**缩减逻辑卷，得先卸载卷，而且要强制进行文件系统检测**

**缩减就是先缩减文件系统边界再缩减物理系统边界**

umount

e2fsck

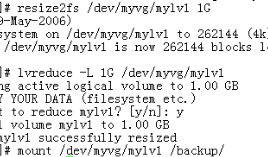
resize2fs 指定大小

lvreduce

mount







注意**缩减1G和缩减到1G的区别**

**创建快照卷**

不需要指定卷组，只需要指定对哪个逻辑卷

-s明确说明是个快照卷

-p r指定只读卷

-n 指定名字

lvcreate -L SIZE -n NAME -s -p r /path/to/LV



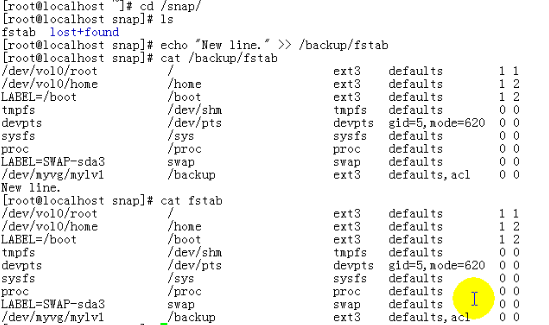
卷大小取决于存活时间



其实就是个链接文件 链接到了mapper下面







备份直接tar



之后就可以删除了

先umount

