

# Linux 磁盘管理

## 一、磁盘分区

磁盘分区可以将硬盘驱动器划分为多个逻辑存储单元，这些单元称为分区。

磁盘分区告诉操作系统分区可访问的柱面范围（扇区范围），记录在第一个扇区的分区表中，最多记录 4 个分区信息。



主分区：也称为主磁盘分区，主分区中不能再划分其他类型的分区，因此每个主分区都相当于一个逻辑磁盘。

扩展分区：扩展分区与逻辑分区是为了突破分区表中只能保存 4 个分区的限制而出现的，扩展分区不能直接使用，需要在扩展分区内划分一个或多个逻辑分

区后才能使用。在扩展分区上面，可以创建多个逻辑分区，逻辑分区是盘上一块连续的区域，它是扩展分区的组成部分。

### 磁盘分区命名规则

在 Linux 中，没有盘符这个概念，通过设备名来访问设备，设备名存放在 /dev 目录中。

命名规则： /dev/xxN

xx： 代表设备类型，通常是 hd（IDE 磁盘），sd（SCSI 磁盘），fd（软驱），vd（virtio 磁盘）

y： 代表分区所在的设备，例如 /dev/hda（第一个 IDE 磁盘）或 /dev/sdb（第二个 SCSI 磁盘）

N： 代表分区，前 4 个分区（主分区或扩展分区）用数字 1 到 4，逻辑分区从 5 开始，例如 /dev/hda3 是第一个 IDE 磁盘上第三个主分区或扩展分区；/dev/sdb6 是第二个 SCSI 硬盘上的第二个逻辑分区。

## 二、磁盘管理命令

### 1. df 命令（disk free）

格式：df [选项] [文件名]

功能：显示文件系统上的磁盘空间使用情况，默认为 KB。

选项：

- a 显示所有系统文件
- B <块大小> 指定显示的块大小
- h 以容易阅读的方式显示
- H 以 1000 字节为换算单位来显示
- i 显示索引字节信息
- k 指定块大小为 1KB
- l 只显示本地文件系统
- t <文件系统类型> 只显示指定类型的文件系统
- T 输出时显示文件系统类型

### 2. lsblk 命令（list block）

格式：lsblk [选项] [设备名]

功能：列出所有可用块设备的信息。

选项:

-a 显示所有设备

-f 输出文件系统信息

### 3. blkid 命令

格式: blkid

功能: 输出块设备上属性, 包括所使用的文件系统类型、LABEL、UUID 等信息。

### 4. fdisk 命令

格式: fdisk [选项] <设备>    fdisk [选项] -l [设备]

功能: 管理磁盘分区。

选项:

-b 指定扇区大小

-l 输出磁盘分区表信息

子菜单:

m : 显示菜单和帮助信息

a : 活动分区标记/引导分区

d : 删除分区

l : 显示分区类型

n : 新建分区

p : 显示分区信息

q : 退出不保存

t : 设置分区号

v : 进行分区检查

w : 保存修改

x : 扩展应用, 高级功能

## 三、Linux 文件系统

Linux 文件系统常用类型: ex3/ex4、xfs、swap、vfat、NFS;

### 1. mount 命令

格式: mount [-t vfstype] [-o options] 设备文件 挂载点

功能: 挂载文件系统。

选项:

-r: 只读挂载

-w: 读写挂载

-n: 不更新/etc/mtab

-a: 自动挂载所有在/etc/fstab 文件中定义的支持自动挂载的设备

-o options 选项: 主要用来描述设备或档案的挂接方式。

使用举例:

挂载硬盘分区:

```
mkdir /usr/music
```

```
mount /dev/sdc5 /usr/music
```

挂载光驱:

```
mkdir /mnt/cdrom
```

```
mount /dev/cdrom /mnt/cdrom
```

挂载 U 盘:

```
mkdir /mnt/usb
```

```
mount /dev/sdd1 /mnt/usb
```

挂载 Windows 下的 C 盘 (FAT32 格式):

```
mkdir /mnt/dosc
```

```
mount -t vfat /dev/sda1 /mnt/dos
```

注意: C 盘必须已经被格式化为 FAT32 格式。

2. umount 命令

格式: umount [-ahnrV] [-t vfstype] 设备名|挂载点

功能: 卸载文件系统。

选项:

-t 指定创建的文件系统类型

3. fsck 命令 (file system check)

格式: fsck [选项] [文件系统]

功能: 检查和修复 Linux 文件系统, 文件系统可以为设备名或者挂载点。

选项:

- t 指定创建的文件系统类型
- a : 如果检查有错则自动修复
- r : 如果检查有错则由使用者回答是否修复

#### 4. du 命令 (disk usage)

格式: du [选项] [文件]

功能: 显示指定的目录或文件所占用的磁盘空间, 默认为 KB。

选项:

- a 显示目录中所有文件大小
- k 以 KB 为单位显示文件大小
- m 以 MB 为单位显示文件大小
- g 以 GB 为单位显示文件大小
- h 以易读方式显示文件大小
- s 仅显示总计

### 四、逻辑卷管理

#### 1. 逻辑卷概念

LVM-逻辑卷管理是 Linux 环境中对磁盘分区进行管理的一种机制, 是建立在硬盘和分区之上、文件系统之下的一个逻辑层, 可提高磁盘分区管理的灵活性。当分区的空间不够用时无法扩展其大小, 只能通过添加硬盘、创建新的分区来扩充空间。



LVM(Logical volume Manager) 逻辑卷管理: 通过将底层物理硬盘抽象地封装起来, 以逻辑卷的形式表现给上层系统。逻辑卷的大小可以动态调整, 而且不会丢失现有的数据。新加入的硬盘也不会改变现有上层的逻辑卷。

可伸缩的“分区卷”, 其分区大小可以动态调节, 而不受制于原始的固定分区大小。

## 2. 逻辑卷管理术语

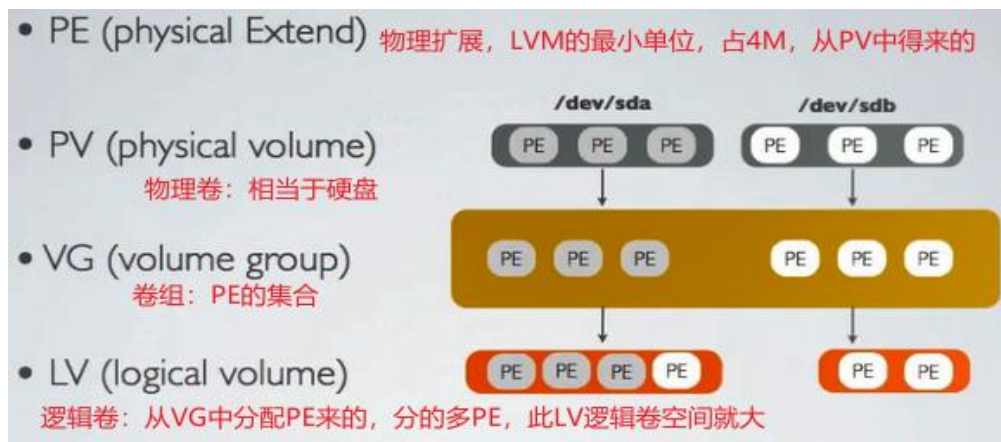
**物理存储介质** (Physical Storage Media): 指系统的物理存储设备: 磁盘, 如: /dev/hda、/dev/sda 等, 是存储系统最底层的存储单元。

**物理卷** (Physical Volume, PV): 指磁盘分区或从逻辑上与磁盘分区具有同样功能的设备 (如 RAID), 是 LVM 的基本存储逻辑块, 但和基本的物理存储介质 (如分区、磁盘等) 比较, 却包含有与 LVM 相关的管理参数。

**物理块** (Physical Extent, PE): PE 是物理卷 PV 的基本划分单元, 具有唯一编号的 PE 是可以被 LVM 寻址的最小单元。PE 的大小是可配置的, 默认为 4MB。物理卷 (PV) 由大小等同的基本单元 PE 组成。

**卷组** (Volume Group, VG): 类似于非 LVM 系统中的物理磁盘, 其由一个或多个物理卷 PV 组成。可以在卷组上创建一个或多个 LV (逻辑卷)。

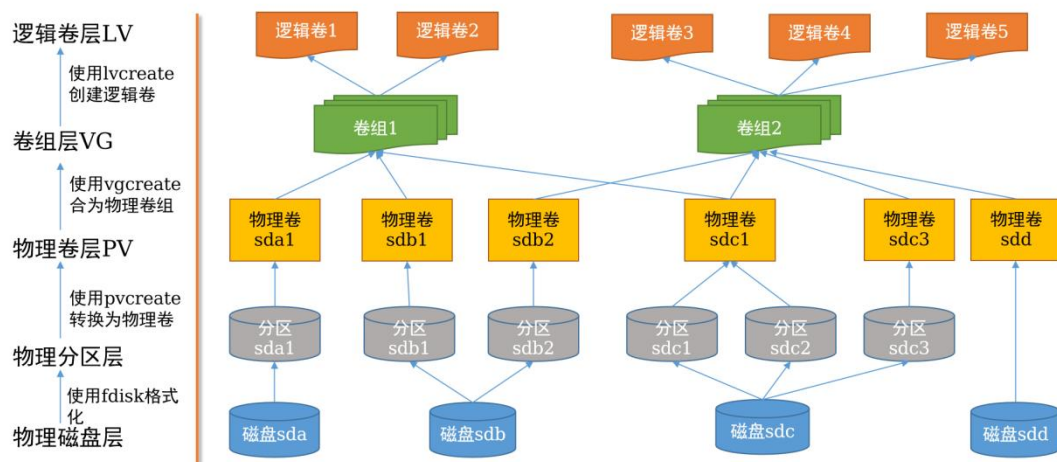
**逻辑卷** (Logical Volume, LV): 类似于非 LVM 系统中的磁盘分区, 逻辑卷建立在卷组 VG 之上。在逻辑卷 LV 之上可以建立文件系统 (比如 /home 或者 /usr 等)。



## 3.3 逻辑卷管理

逻辑卷创建流程:

物理磁盘被格式化为 PV (physical volume), 空间被分为一个个 PE (physical extent) (大小为 4 MB, 为逻辑卷的最小单位)。不同的 PV 加入同一个 VG (volume group), 不同 PV 的 PE 全部进入 VG 的 PE 池内。LV (logical volume) 基于 PE 创建, 大小为 PE 的整数倍, 组成 LV 的 PE 可能来自于不同的物理磁盘。LV 挂载后使用。



## 4 逻辑卷使用

### (1) 安装 lvm (Ubuntu)

```
sudo apt-get install lvm2
sudo systemctl enable lvm2-lvmetad.service
sudo systemctl enable lvm2-lvmetad.socket
sudo systemctl start lvm2-lvmetad.socket
sudo systemctl start lvm2-lvmetad.service
```

### (2) LV 创建步骤

#### ① 将物理磁盘设备初始化为物理卷

```
pvcreeate /dev/sdb /dev/sdc
```

查看物理卷信息: `pvdipaly` 或者 `pvs`

#### ② 创建卷组, 并将上面两个 PV 加入卷组 myvg 中

```
vgcreate myvg /dev/sdb /dev/sdc
```

查看卷组信息: `vgdisplay` 或者 `vgs`

#### ③ 基于卷组 myvg 创建 2G 大小空间的逻辑卷

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

查看逻辑卷信息: `lvdisplay` 或者 `lvs`

#### ④ 为创建好的逻辑卷格式化创建文件系统

```
mkfs.ext4 /dev/myvg/mylv
```

#### ⑤ 将格式化好的逻辑卷挂载使用

```
mount /dev/myvg/mylv /mnt 将逻辑卷挂载到 /mnt 中
```

### (3) LV 删除步骤

卸载已经挂载的 LV

```
umount /dev/myvg/mylv
```

删除 LV

```
lvremove /dev/myvg/mylv
```

删除 VG

```
vgremove myvg
```

删除物理卷 PV

```
pvremove /dev/sdb
```

### (4) LV 拉伸步骤

检查 vg 卷组剩余空间

```
vgdisplay
```

从 vg 卷组扩容 lv

```
lvextend -L +100M /dev/myvg/mylv
```

```
lvresize -L 300M /dev/myvg/mylv
```

查看扩充后 LV 大小

```
lvdisplay
```

更新文件系统（必须的，否则 df -h 还是显示原来的空间大小）

```
resize2fs /dev/myvg/mylv
```

查看更新后文件系统

```
df -h
```