

文件系统是 linux 的一个十分基础的知识，同时也是学习 linux 的必备知识。

本文将站在一个较高的视图来了解 linux 的文件系统，主要包括了 linux 磁盘分区和目录、挂载基本原理、文件存储结构、软链接硬链接、和常见目录的介绍。相信有了这些知识对于深入的学习 linux 会有一定的帮助。文章例子主要是基于 ubuntu 发行版。

如有不对之处请大家多多指出。

## 1.Linux 磁盘分区和目录

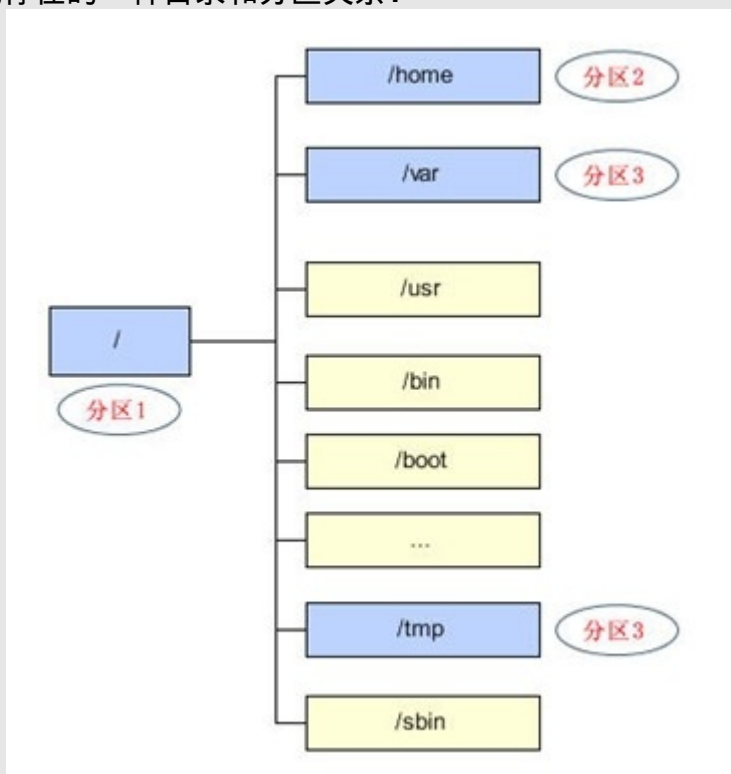
Linux 发行版本之间的差别很少，差别主要表现在系统管理的特色工具以及软件包管理方式的不同。目录结构基本上都是一样的。Windows 的文件结构是多个并列的树状结构，最顶部的是不同的磁盘（分区），如：C，D，E，F 等。

Linux 的文件结构是单个的树状结构。可以用 tree 进行展示。在 Ubuntu 下安装 tree (sudo apt-get install tree), 并可通过命令来查看。

每次安装系统的时候我们都会进行分区，Linux 下磁盘分区和目录的关系如下：

- 任何一个分区都必须挂载到某个目录上。
- 目录是逻辑上的区分。分区是物理上的区分。
- 磁盘 Linux 分区都必须挂载到目录树中的某个具体的目录上才能进行读写操作。
- 根目录是所有 Linux 的文件和目录所在的地方，需要挂载上一个磁盘分区。

以下是我们可能存在的一种目录和分区关系：



Q:如何查看分区和目录及使用情况？

- fdisk 查看硬盘分区表
- df: 查看分区使用情况
- du: 查看文件占用空间情况

Q: 为什么要分区，如何分区？

- 可以把不同资料，分别放入不同分区中管理，降低风险。
- 大硬盘搜索范围大，效率低

- 磁盘配合只能对分区做设定
- /home /var /usr/local 经常是单独分区，因为经常会操作，容易产生碎片

## 2. Mount 挂载和 NFS 简介

挂载的概念：当要使用某个设备时，例如要读取硬盘中的一个格式化好的分区、光盘或软件等设备时，必须先把这些设备对应到某个目录上，而这个目录就称为“挂载点 (mount point)”，这样才可以读取这些设备，而这些对应的动作就是“挂载”。将物理分区细节屏蔽掉。用户只有统一的逻辑概念。所有的东西都是文件。Mount 命令可以实现挂载：

`mount [-fnrsvw] [-t vfstype] [-o options] device dir`

**Q：**所有的磁盘分区都必须被挂载上才能使用，那么我们机器上的硬盘分区是如何被挂载的？

**A：**这主要是它利用了/etc/fstab 文件。每次内核加载它知道从这里开始 mount 文件系统。每次系统启动会根据该文件定义自动挂载。若没有被自动挂载，分区将不能使用。如下是我的/etc/fstab 的定义，主要是根据装机的分区来的：

```
# <file system> <mount point> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0
#/dev/sda1 被自动挂载到 /
UUID=cb1934d0-4b72-4bbf-9fad-885d2a8eeeb1 / ext3
relatime,errors=remount-ro 0 1
# /dev/sda5 被自动挂载到分区/home
UUID=c40f813b-bb0e-463e-aa85-5092a17c9b94 /home ext3
relatime 0 2
#/dev/sda7 被自动挂载到/work
UUID=0f918e7e-721a-41c6-af82-f92352a568af /work ext3
relatime 0 2
#分区 /dev/sda6 被自动挂载到 swap
UUID=2f8bdd05-6f8e-4a6b-b166-12bb52591a1f none swap
sw 0 0
```

**Q：**移动硬盘如何挂载？如何挂载一个新的分区？

移动硬盘有驱动模块会自动挂载，如果有个新硬盘，要先进行分区，并通过 mount 命令挂载到某个文件夹。如果要自动挂载则可以修改/etc/fstab 文件。

**NFS 简介：**NFS 相信在很多地方都有广泛使用，是一个非常好的文件共享方式。我们公司所使用的上传服务就是把文件上传到某台网络服务器上，中间就是通过 NFS 实现。

使用 NFS 客户端可以透明的地访问服务器端的文件。NFS 也是通过 mount 来实现，底层是通过 NFS 通信协议实现。基本原理：

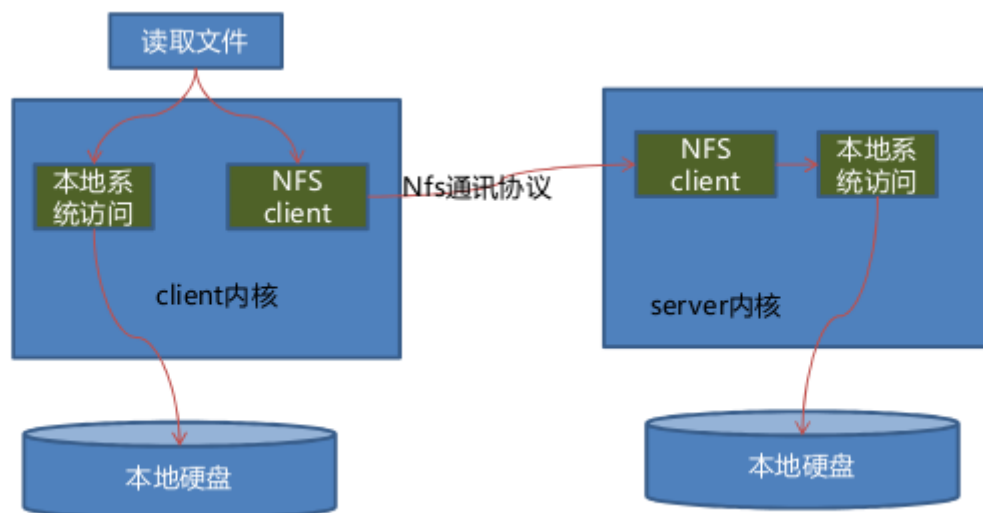


图 2：NFS 基本原理

Ubuntu 下面 Ubuntu 下的例子

服务端：

```
$apt-get install nfs-kernel-server
vi /etc/exports 添加 nfs 目录: /personal/nfs_share
10.1.60.34(rw,sync,no_root_squash)
$sudo exportfs -r
$sudo /etc/init.d/portmap start
$sudo /etc/init.d/nfs-kernel-server start
```

客户端：

```
$sudo apt-get install nfs-common
$sudo mount 10.19.34.76:/personal/nfs_share ~/nfsshare 例子：
```

### 3.文件类型

Linux 下面的文件类型主要有：

- 普通文件：C 语言源代码、SHELL 脚本、二进制的可执行文件等。分为纯文本和二进制。
- 目录文件：目录，存储文件的唯一地方。
- 链接文件：指向同一个文件或目录的文件。
- 特殊文件：与系统外设相关的，通常在/dev 下面。分为块设备和字符设备。

可以通过 ls -l, file, stat 几个命令来查看文件的类型等相关信息。

### 4.文件存储结构

Linux 正统的文件系统(如 ext2、ext3)一个文件由目录项、inode 和数据块组成。

目录项:包括文件名和 inode 节点号。

**Inode**：又称文件索引节点，是文件基本信息的存放地和数据块指针存放地。

数据块：文件的具体内容存放地。

Linux 正统的文件系统(如 ext2、3 等)将硬盘分区时会划分出目录块、inode Table 区块和 data block 数据区域。一个文件由一个目录项、inode 和数据区域块组成。Inode 包含文件的属性(如读写属性、owner 等，以及指向数据块的指针)，数据区域块则是文件内容。当查看某个文件时，会先从 inode table 中查出文件属性及数据存放点，再从数据块中读取数据。

站在 2w 英尺视图,文件存储结构大概如下：

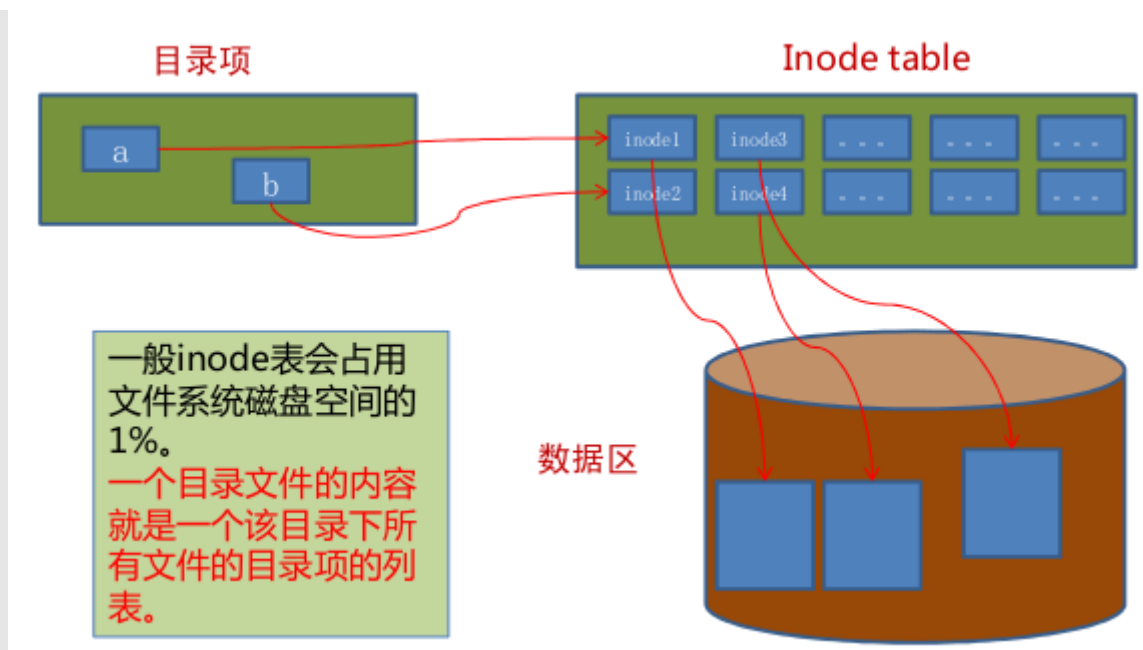


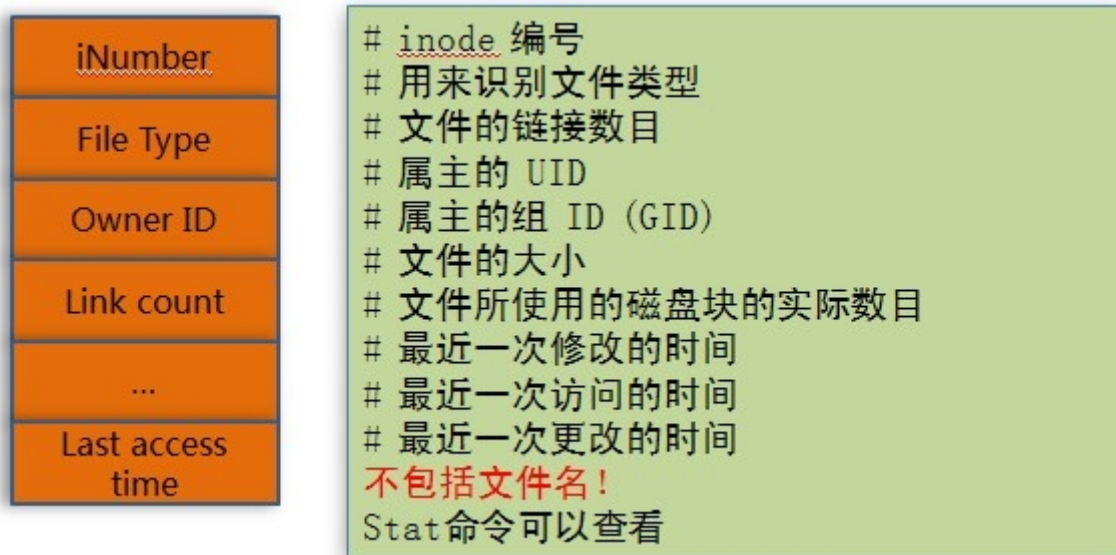
图 3：文件存储结构 2w 英尺视图

其中目录项的结构如下(每个文件的目录项存储在改文件所属目录的文件内容里)：



图 4：目录项结构

其中文件的 inode 结构如下 (inode 里所包含的文件信息可以通过 stat filename 查看得到)：



以上只反映大体的结构，linux 文件系统本身在不断发展。但是以上概念基本是不变的。且如 ext2、ext3、ext4 文件系统也存在很大差别，如果要了解可以查看专门的文件系统介绍。

## 5. 软连接、硬链接

软链接和硬链接是我们常见的两种概念：

**硬连接：**是给文件一个副本，同时建立两者之间的连接关系。修改其中一个，与其连接的文件同时被修改。如果删除其中任意一个其余的文件将不受影响。

**软连接：**也叫符号连接，他只是对源文件在新的位置建立一个“快捷（借用一下 windows 常用词）”，所以，当源文件删除时，符号连接的文件将成为无源之水->仅仅剩下个文件名了，当然删除这个连接，也不会影响到源文件，但对连接文件的使用、引用都是直接调用源文件

的。  
具体关系可以看下图：

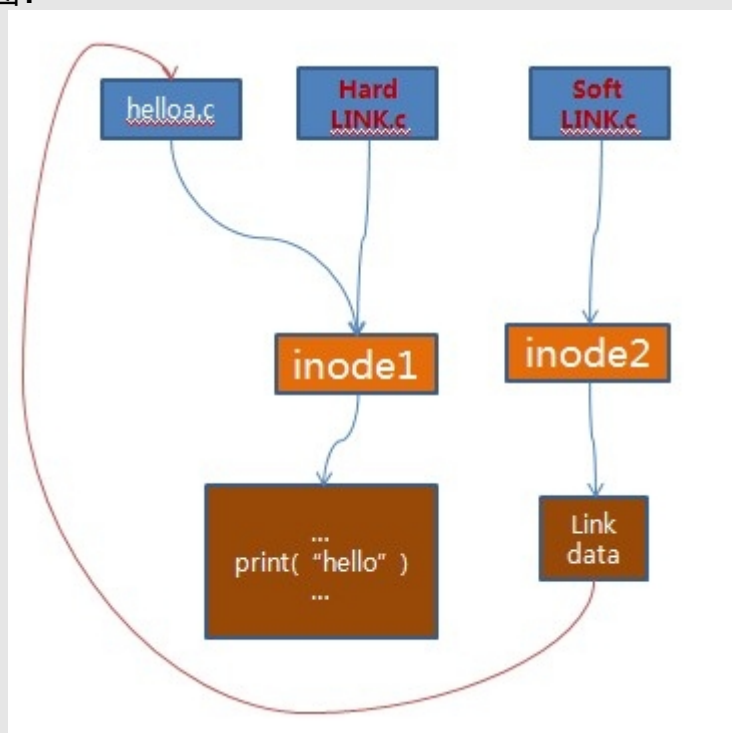


图 5：软链接和硬链接

从图上可以看出硬链接和软链接的区别：

- 1：硬链接原文件和新文件的 inode 编号一致。而软链接不一样。
- 2：对原文件删除，会导致软链接不可用，而硬链接不受影响。
- 3：对原文件的修改，软、硬链接文件内容也一样的修改，因为都是指向同一个文件内容的。

## 6.文件目录管理命令

磁盘和文件空间

fdisk df du

文件目录与管理

cd pwd mkdir rmdir ls cp rm mv

查看文件内容

cat:

cat [file]

查看文件的内容。全程式 concatenate 的意思，将文件内容连续输出到屏幕上。第一行到最后一行显示。

tac:

tac [file]

和 cat 刚好相反 是从最后一行到第一行的方式查看。

cat 有个比较不好的地方是当文件比较大时候没办法看清楚，这个时候可以用 more 或者 Less 命令。

more:

more [file]

如果使用 grep 或者 find 等命令时，可以配合使用 more 一页一页的查看。如果看到一半想退出，则敲入 'q' 即可退出。

less:

less [file]

less 比 more 更有弹性，可以上下翻页。

如果只想读取文件的头几行或者文件的末尾几行，可以用 head 或 tail。

head -n [file]: 读取文件的前 n 行。

tail -n [file]: 读取文件末尾 n 行。

以上命令都是用于查看字符文件，二进制文件出来的都是乱码，要看二进制文件的内容，可以用 od 命令，如查看一个 MP3 文件里面的内容：

od shijiemori.mp3

文件目录与权限

chmod chown chgrp umask

文件查找

which:

which [filename]

该命令用于查询通过 PATH 路径到该路径内查找可执行文件。

如：Which passwd:查找可执行文件 passwd

whereis:

whereis [-bmsu] [keyword]

该命令用于把相关字的文件和目录都列出来。(Linux 会将文件都记录在一个文件数据库里面，该命令式从数据库去查询，所以速度比较快,Linux 每天会更新该数据库)

locate:

locate [filename]

该命令用于把相关字的文件和目录都列出来。查找数据特别快，也是通过数据库方式来查询。但是数据库一周更新一次，所以可能有些存在数据查不到。可以去修改配置文件。

find:

find [path] [参数] [keyword]

该命令用于在指定路径下查找文件。不是通过数据来查询，所以速度会比较慢。

## 7. 常见目录解释

Linux 各种发行版的目录结构基本一致，各个目录简单介绍如下：

目录

描述

/

根目录

/bin

做为基础系统所需要的最基础的命令就是放在这里。比如 ls、cp、mkdir 等命令；功能和/usr/bin 类似，这个目录中的文件都是可执行的，普通用户都可以使用的命令。

/boot

Linux 的内核及引导系统程序所需要的文件，比如 vmlinuz initrd.img 文件都位于这个目录中。在一般情况下，GRUB 或 LILO 系统引导管理器也位于这个目录；启动装载文件存放位置，如 kernels,initrd,grub。一般是一个独立的分区。

/dev

一些必要的设备,声卡、磁盘等。还有如 /dev/null. /dev/console /dev/zero /dev/full 等。

/etc

系统的配置文件存放地. 一些服务器的配置文件也在这里；比如用户帐号及密码配置文件；

/etc/opt: /opt 对应的配置文件

/etc/X11: Xwindows 系统配置文件

/etc/xml: XML 配置文件

.....

/home

用户工作目录，和个人配置文件，如个人环境变量等，所有的账号分配一个工作目录。一般是一个独立的分区。

/lib

库文件存放地。bin 和/sbin 需要的库文件。类似 windows 的 DLL。

/media

可拆卸的媒介挂载点，如 CD-ROMs、移动硬盘、U 盘，系统默认会挂载到这里来。

/mnt

临时挂载文件系统。这个目录一般是用于存放挂载储存设备的挂载目录的，比如有 cdrom 等目录。可以参看/etc/fstab 的定义。

/opt

可选的应用程序包。

/proc

操作系统运行时，进程（正在运行中的程序）信息及内核信息（比如 cpu、硬盘分区、内存信息等）存放在这里。/proc 目录伪装的文件系统 proc 的挂载目录，proc 并不是真正的文件系统，它的定义可以参见 /etc/fstab。

/root

Root 用户的工作目录

/sbin

和 bin 类似，是一些可执行文件，不过不是所有用户都需要的，一般是系统管理所需要使用得到的。

/tmp

系统的临时文件，一般系统重启不会被保存。

/usr

包含了系统用户工具和程序。

/usr/bin: 非必须的普通用户可执行命令

/usr/include: 标准头文件

/usr/lib:/usr/bin/ 和 /usr/sbin/的库文件

/usr/sbin:非必须的可执行文件

/usr/src:内核源码

/usr/X11R6:X Window System, Version 11, Release 6.

/srv

该目录存放一些服务启动之后需要提取的数据