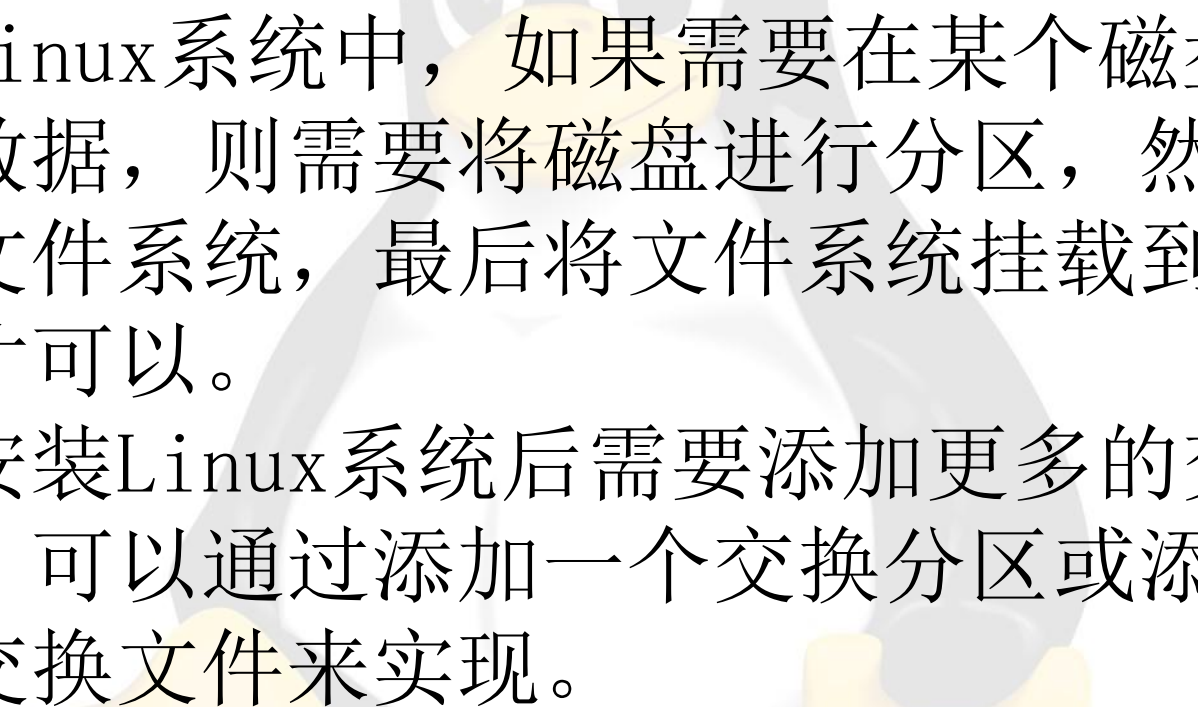


第8章 磁盘分区和文件系统管理



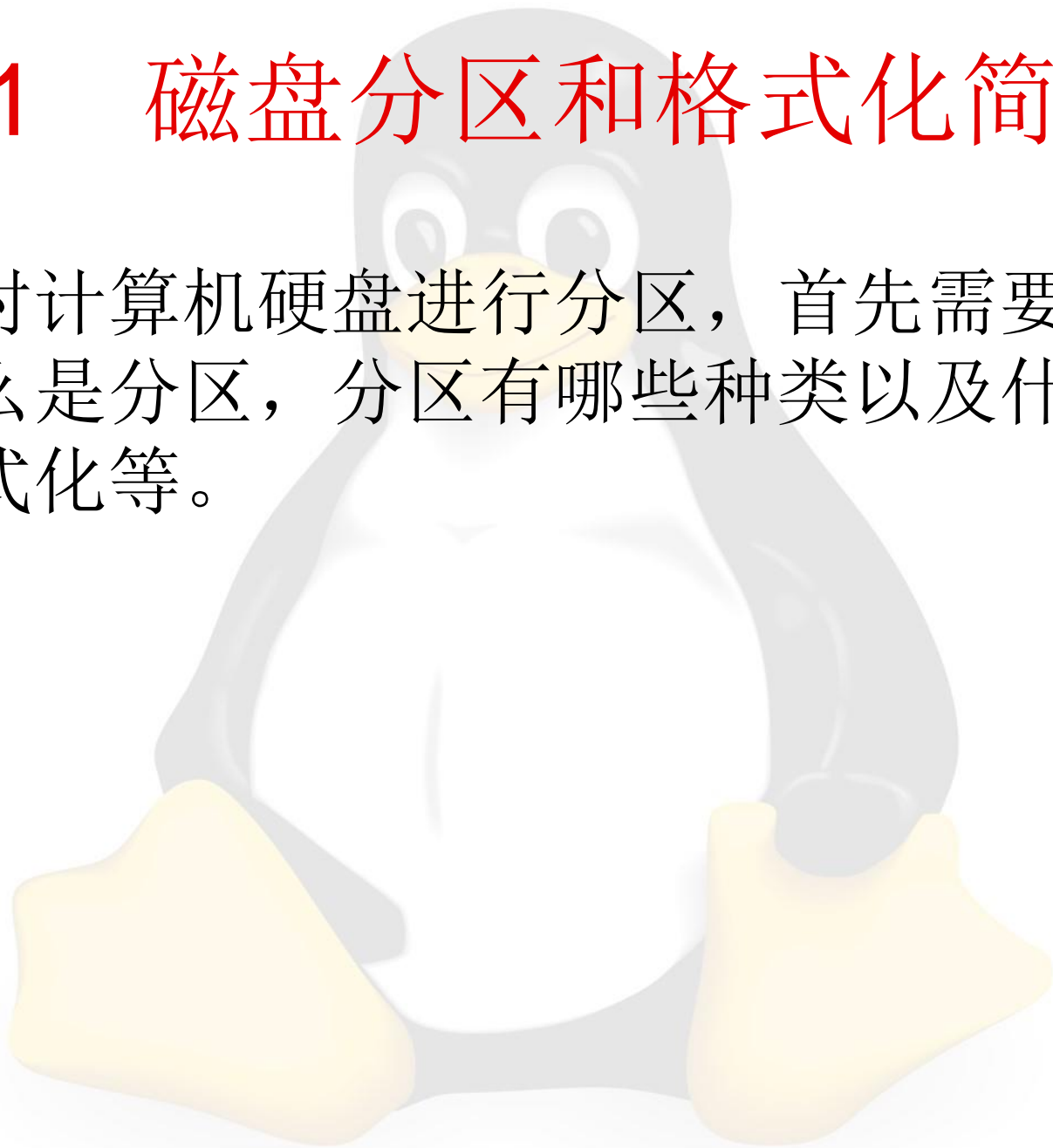
本章内容

- 8.1 磁盘分区和格式化简介
- 8.2 Linux磁盘分区
- 8.3 创建文件系统
- 8.4 挂载和卸载文件系统
- 8.5 开机自动挂载文件系统
- 8.6 使用交换空间

- 
- 在Linux系统中，如果需要在某个磁盘上存储数据，则需要将磁盘进行分区，然后创建文件系统，最后将文件系统挂载到目录下才可以。
 - 在安装Linux系统后需要添加更多的交换空间，可以通过添加一个交换分区或添加一个交换文件来实现。

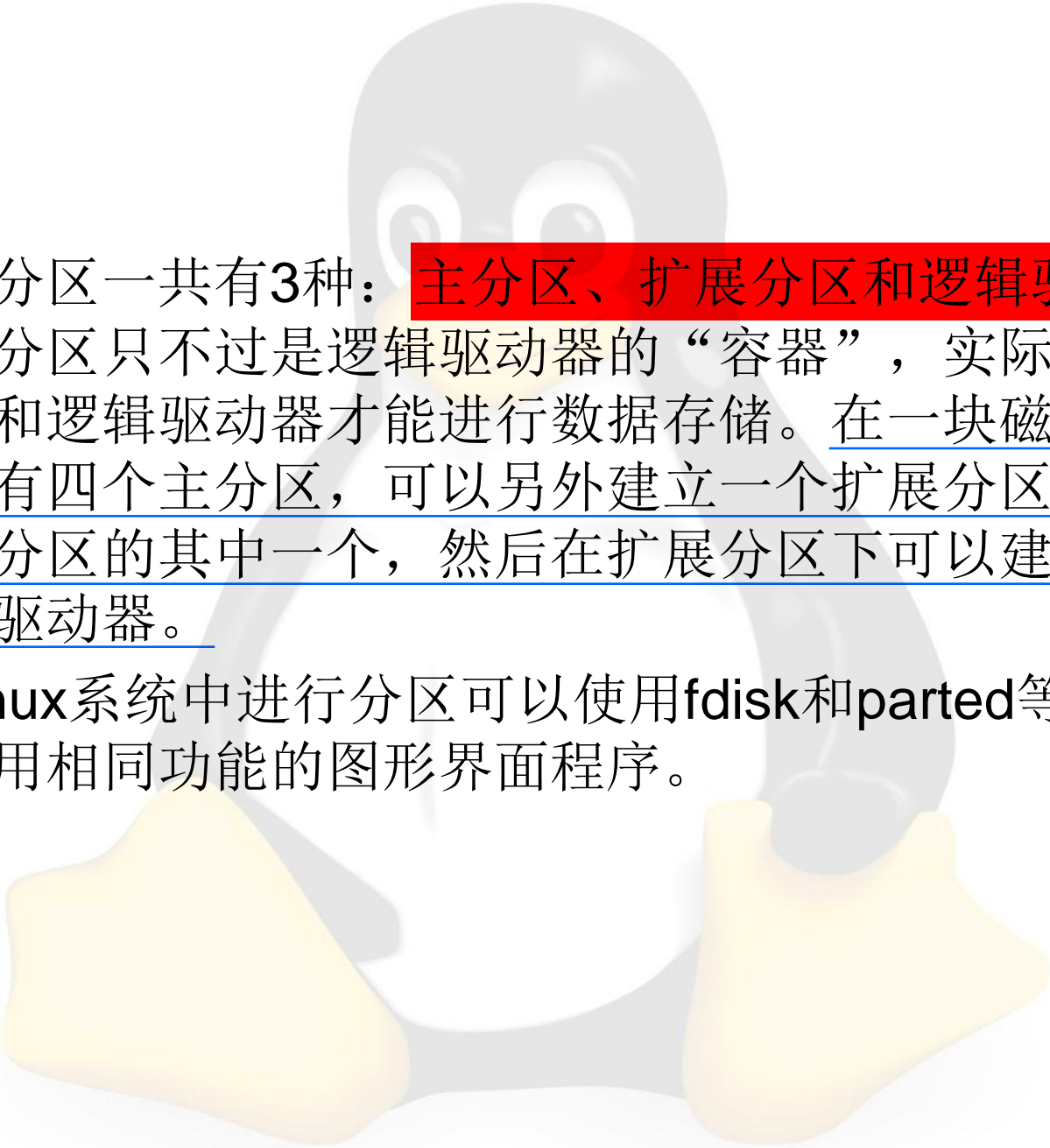
8.1 磁盘分区和格式化简介

- 要对计算机硬盘进行分区，首先需要知道什么是分区，分区有哪些种类以及什么是格式化等。



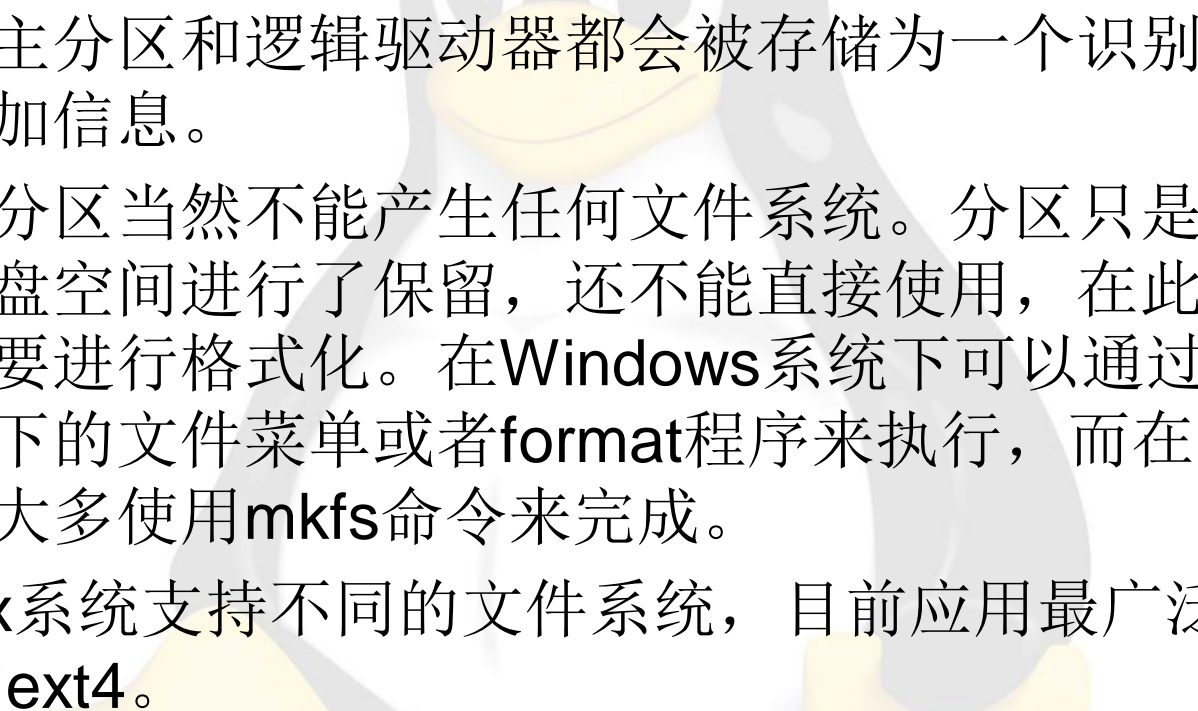
什么是磁盘分区

- 磁盘分区是指对硬盘物理介质的逻辑划分。将磁盘分成多个分区，不仅仅利于对文件的管理，而且不同的分区可以建立不同的文件系统，这样才能在不同的分区上安装不同的操作系统。
- 分区就是磁盘的“段落”，如果用户希望在计算机上安装多个操作系统，将需要更多的分区。假设需要同时安装Windows 10和Windows Server 2012系统，那么至少需要两个分区，原因是不同的操作系统原则上采用不同的文件系统。如果几个操作系统都支持相同的文件系统，通常为了避免在一个分区下有相同的系统目录，也将它们安装在不同的磁盘分区上。在Linux系统中，情况又有所不同，它本身又需要更多的磁盘分区，比如根分区“/”和swap分区。

- 
- 磁盘分区一共有3种：**主分区、扩展分区和逻辑驱动器**。
扩展分区只不过是逻辑驱动器的“容器”，实际上只有主分区和逻辑驱动器才能进行数据存储。在一块磁盘上最多只能有四个主分区，可以另外建立一个扩展分区来代替四个主分区的其中一个，然后在扩展分区下可以建立更多的逻辑驱动器。
 - 在Linux系统中进行分区可以使用fdisk和parted等命令，或者使用相同功能的图形界面程序。

格式化的概念

- 磁盘经过分区之后，下一个步骤就是要对磁盘分区进行格式化的工作（也就是创建文件系统的工作）。格式化是指对磁盘分区进行初始化的一种操作，这种操作通常会导致现有的分区中所有的数据被清除。简单说，就是把一张空白的磁盘划分成一个个小区域并编号，供计算机储存和读取数据使用。
- 格式化是在磁盘中建立磁道和扇区，建立好之后，计算机才可以使用磁盘来储存数据。格式化的动作通常是在磁盘的开端写入启动扇区的数据、在根目录记录磁盘卷标、为文件分配表保留一些空间，以及检查磁盘上是否有损坏的扇区，如果有的话则在文件分配表标上损毁的记号，表示该扇区并不用来储存数据。

- 
- 每个主分区和逻辑驱动器都会被存储为一个识别文件系统的附加信息。
 - 通过分区当然不能产生任何文件系统。分区只是对磁盘上的磁盘空间进行了保留，还不能直接使用，在此之后分区必须要进行格式化。在Windows系统下可以通过资源管理器下的文件菜单或者format程序来执行，而在Linux系统中大多使用mkfs命令来完成。
 - Linux系统支持不同的文件系统，目前应用最广泛的就是xfs和ext4。

8.2 Linux磁盘分区

- 使用**fdisk**命令可以用来对磁盘进行分区，它采用传统的问答式界面，除此之外还可以用来查看磁盘分区的详细信息，也能为每个分区指定分区的类型。

命令语法：

fdisk [选项] [设备]

fdisk交互式操作子命令

子命令	功能
m	显示所有能在fdisk中使用的子命令
p	显示磁盘分区信息
a	设置磁盘启动分区
n	创建新的分区
e	创建扩展分区
p	创建主分区
t	更改分区的系统ID（也就是分区类型ID）
d	删除磁盘分区
q	退出fdisk，不保存磁盘分区设置
l	列出已知的分区类型
v	验证分区表
w	保存磁盘分区设置并退出fdisk

Linux常用分区类型

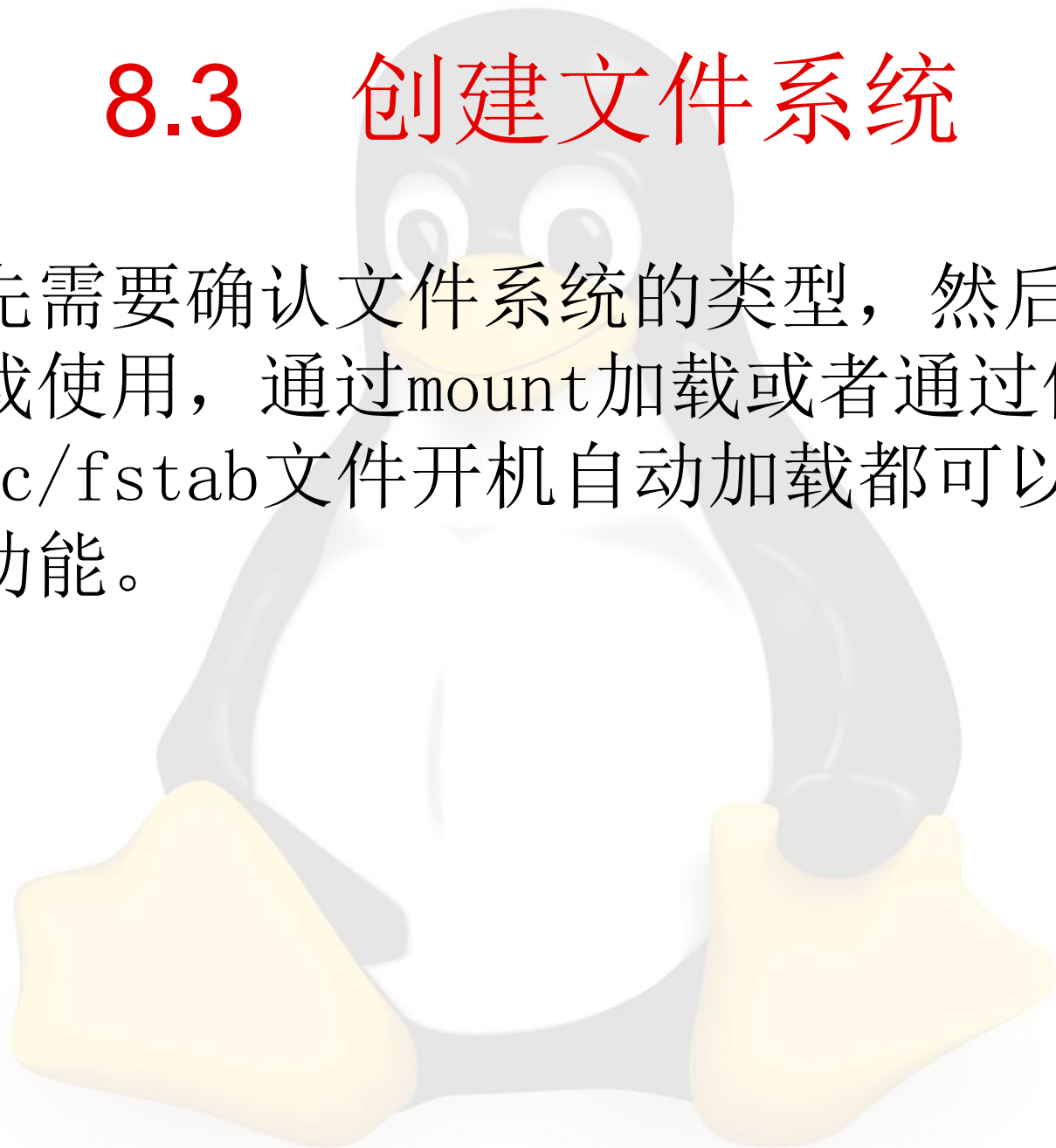
ID	分区类型	描述
83	Linux	Linux普通分区
fd	Linux raid 自动	RAID使用的分区
8e	Linux LVM	LVM使用的分区
82	Linux swap / Solaris	swap分区

Linux系统下硬盘分区举例

1. 进入fdisk界面，显示磁盘分区信息。
2. 创建和删除主分区。
3. 创建扩展分区和逻辑驱动器。
4. 查看并转换分区类型。
5. 保存分区设置信息，并退出fdisk。
6. 在非交互式界面下显示当前硬盘的分区信息。
7. 查看分区情况（使用**partprobe**命令更新分区表，使内核识别分区）。

8.3 创建文件系统

- 首先需要确认文件系统的类型，然后才能挂载使用，通过mount加载或者通过修改/etc/fstab文件开机自动加载都可以实现该功能。



Linux主流文件系统

- 对一个新的磁盘进行分区以后，还要对这些分区进行格式化并创建文件系统。一个分区只有建立了某种文件系统后，这个分区才能使用。建立文件系统的过程，就是用相应的格式化工具格式化分区的过程，这个过程和在Windows系统中格式化某个分区为NTFS分区的过程类似。
- 文件系统是指文件在硬盘上的存储方法和排列顺序。在Linux系统中，每个分区都需要一个文件系统，都有自己的目录层次结构。Linux系统最重要特征之一就是支持多种文件系统，这样它更加灵活，并可以和其他种操作系统共存。
- 随着Linux系统的不断发展，它所支持的文件系统类型也在迅速扩充，其中有XFS、ext4、JFS、ReiserFS、ext2、ext3、ISO9660、MSDOS，VFAT、NFS等。

创建文件系统

- 如果在计算机上新增加了一块硬盘，需要格式化成 **Linux** 文件系统，最好选择 **xfs** 或 **ext4** 文件系统。使用 **mkfs** 命令可以在分区上创建各种文件系统。**mkfs** 命令本身并不执行建立文件系统的工作，而是去调用相关的程序来执行。这里的文件系统是要指定的，比如 **xfs**、**ext4**、**ext3**、**vfat** 或者是 **msdos** 等。

命令语法：

mkfs [选项] [设备]

【例8.1】 使用mkfs命令创建文件系统。

(1) 查看当前磁盘上的分区情况，该磁盘设备是/dev/sda。

```
[root@rhel ~]# fdisk -l /dev/sda
```

```
Disk /dev/sda: 21.5 GB, 21474836480 bytes
```

```
255 heads, 63 sectors/track, 2610 cylinders, total 41943040 sectors
```

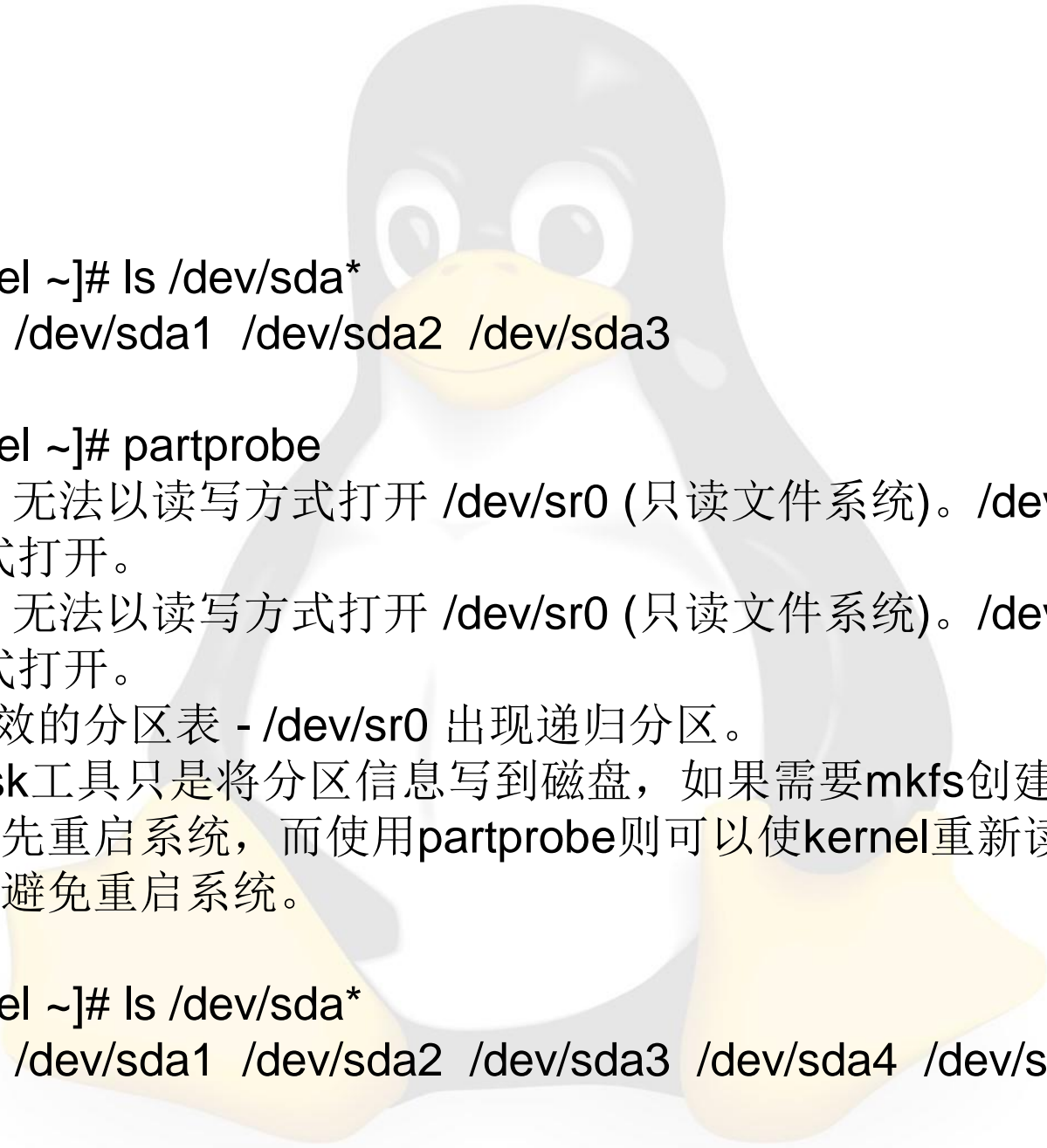
```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk identifier: 0x0001c201
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	206847	102400	83	Linux
/dev/sda2		206848	31664127	15728640	83	Linux
/dev/sda3		31664128	35858431	2097152	82	Linux swap / Solaris
/dev/sda4		35858432	41943039	3042304	5	Extended
/dev/sda5		35860480	35880959	10240	83	Linux
/dev/sda6		35883008	41943039	3030016	c	W95 FAT32 (LBA)



```
[root@rhel ~]# ls /dev/sda*  
/dev/sda /dev/sda1 /dev/sda2 /dev/sda3
```

```
[root@rhel ~]# partprobe
```

Warning: 无法以读写方式打开 /dev/sr0 (只读文件系统)。/dev/sr0 已按照只读方式打开。

Warning: 无法以读写方式打开 /dev/sr0 (只读文件系统)。/dev/sr0 已按照只读方式打开。

Error: 无效的分区表 - /dev/sr0 出现递归分区。

//使用fdisk工具只是将分区信息写到磁盘，如果需要mkfs创建文件系统则需要事先重启系统，而使用partprobe则可以使kernel重新读取分区信息，从而避免重启系统。

```
[root@rhel ~]# ls /dev/sda*  
/dev/sda /dev/sda1 /dev/sda2 /dev/sda3 /dev/sda4 /dev/sda5 /dev/sda6
```



(2) 格式化/dev/sda5分区，创建ext4文件系统。
对计算机硬盘分好分区之后重启计算机，使用如下命令创建文件系统。

```
[root@rhel ~]# mkfs -t ext4 /dev/sda5
```

```
mke2fs 1.42 (29-Nov-2011)
```

```
文件系统标签=
```

```
OS type: Linux
```

```
块大小=1024 (log=0)
```

```
分块大小=1024 (log=0)
```

```
Stride=0 blocks, Stripe width=0 blocks
```

```
2560 inodes, 10240 blocks
```

```
512 blocks (5.00%) reserved for the super user
```

```
第一个数据块=1
```

```
Maximum filesystem blocks=10485760
```

```
2 block groups
```

```
8192 blocks per group, 8192 fragments per group
```

```
1280 inodes per group
```

```
Superblock backups stored on blocks: 8193
```



Allocating group tables: 完成

正在写入inode表: 完成

Creating journal (1024 blocks): 完成

Writing superblocks and filesystem accounting information: 完成

当然也可以把分区格式化其他的文件系统。比如可以把/dev/sda5格式化为ext3、reiserfs、vfat或msdos，命令格式如下：

```
[root@rhel ~]# mkfs -t ext3 /dev/sda5
```

```
[root@rhel ~]# mkfs -t reiserfs /dev/sda5
```

```
[root@rhel ~]# mkfs -t vfat /dev/sda5
```

```
[root@rhel ~]# mkfs -t msdos /dev/sda5
```

8.4 挂载和卸载文件系统

- 如果要挂载一个分区，首先需要确认文件系统的类型，然后才能挂载使用，通过使用mount和umount命令可以实现文件系统的挂载和卸载。

挂载文件系统

- 使用mount命令可以将指定分区、光盘、U盘或者是移动硬盘挂载到Linux系统的目录下。

命令语法：

mount [选项] [设备] [挂载目录]

mount命令挂载选项

挂载选项	描述
defaults	相当于rw, suid, dev, exec, auto, nouser, async, relatime挂载选项
ro	以只读方式挂载
rw	以读写方式挂载
nouser	禁止普通用户（即非root）挂载文件系统
user	允许普通用户挂载文件系统。
users	允许每一位用户挂载和卸载文件系统
remount	尝试重新挂载一个已经挂载的文件系统
owner	如果用户是设备的所有者，允许一个普通的用户挂载该文件系统
exec	在挂载的文件系统上允许直接执行二进制文件
noexec	在挂载的文件系统上不允许直接执行任何二进制文件
atime	在文件系统上更新inode访问时间
noatime	在文件系统上不更新inode访问时间
owner	如果用户是设备所有者，允许普通用户挂载文件系统
group	如果用户的其中一个组群匹配设备的组群，则允许普通的用户挂载文件系统
auto	能够使用-a选项挂载
noauto	只能显式挂载（使用-a选项将不会导致文件系统被挂载）
suid	允许设置用户标识或设置组标识符位才能生效
nosuid	不允许设置用户标识或设置组标识符位才能生效

【例8.5】 挂载分区/dev/sda5到/mnt/kk目录中。

```
[root@rhel ~]# mkdir /mnt/kk
```

```
[root@rhel ~]# mount /dev/sda5 /mnt/kk
```

//创建需要放置文件系统的目录/mnt/kk，然后将/dev/sda5挂载到该目录中

```
[root@rhel ~]# ls /mnt/kk
```

lost+found

//查看目录/mnt/kk，其中没有任何的文件

```
[root@rhel ~]# touch /mnt/kk/abc
```

//在目录/mnt/kk中创建空文件abc

```
[root@rhel ~]# ls /mnt/kk
```

abc lost+found

//再次查看目录/mnt/kk的内容，可以看到刚才创建的文件，其实它是存在于/dev/sda5分区上

```
[root@rhel ~]# df
```

文件系统	1K-块	已用	可用	已用%	挂载点
rootfs	15674684	9442696	5445556	64%	/
devtmpfs	504952	0	504952	0%	/dev
tmpfs	513924	84	513840	1%	/dev/shm
tmpfs	513924	2136	511788	1%	/run
/dev/sda2	15674684	9442696	5445556	64%	/
tmpfs	513924	0	513924	0%	/sys/fs/cgroup
tmpfs	513924	0	513924	0%	/media
/dev/sda1	100604	43743	51741	46%	/boot
/dev/sda5	10084	1290	8282	14%	/mnt/kk



【例8.4】 以只读方式挂载/dev/sda5分区到/mnt/kk目录中。

```
[root@rhel ~]# mount -o ro /dev/sda5 /mnt/kk
```

//以只读方式挂载/dev/sda5分区

```
[root@rhel ~]# mkdir /mnt/kk/a
```

Mkdir: 无法创建目录“/mnt/kk/a”: 只读文件系统

//在目录/mnt/kk中创建目录a，无法创建，因为它是只读的

卸载文件系统

- 使用 **umount** 命令可以将指定分区、光盘、U盘或者是移动硬盘进行卸载。umount 可以卸载目前挂载在 Linux 目录中的文件系统，除了直接指定文件系统外，也可以使用设备名称或挂载目录来表示文件系统。

命令语法：

umount [选项] [设备|挂载目录]

【例8.7】 卸载分区/dev/sda5文件系统。

```
[root@rhel ~]# umount /dev/sda5
```

```
[root@rhel ~]# df
```

文件系统	1K-块	已用	可用	已用%	挂载点
rootfs	15674684	9442696	5445556	64%	/
devtmpfs	504952	0	504952	0%	/dev
tmpfs	513924	84	513840	1%	/dev/shm
tmpfs	513924	2136	511788	1%	/run
/dev/sda2	15674684	9442696	5445556	64%	/
tmpfs	513924	0	513924	0%	/sys/fs/cgroup
tmpfs	513924	0	513924	0%	/media
/dev/sda1	100604	43743	51741	46%	/boot

//使用df命令可以看到分区/dev/sda5已经卸载

查看磁盘分区挂载情况

- 要查看Linux系统上的磁盘分区挂载情况，可以使用df命令来获取信息。使用df命令可以显示每个文件所在的文件系统的信息，默认是显示所有文件系统。检查文件系统的磁盘空间使用情况，利用该命令获取硬盘使用了多少空间、目前还剩下多少空间等相关信息。

命令语法：

df [选项] [文件]

【例8.8】 显示磁盘空间的使用情况。

[root@rhel ~]# df

文件系统	1K-块	已用	可用	已用%	挂载点
rootfs	15674684	9442696	5445556	64%	/
devtmpfs	504952	0	504952	0%	/dev
tmpfs	513924	84	513840	1%	/dev/shm
tmpfs	513924	2136	511788	1%	/run
/dev/sda2	15674684	9442696	5445556	64%	/
tmpfs	513924	0	513924	0%	/sys/fs/cgroup
tmpfs	513924	0	513924	0%	/media
/dev/sda1	100604	43743	51741	46%	/boot
/dev/sda5	10084	1290	8282	14%	/mnt/kk

8.5 开机自动挂载文件系统

- 只有将某个分区或是设备挂载以后才能使用，但是当计算机重新启动以后，又需要重新挂载，这个时候可以通过修改 `/etc/fstab` 文件实现开机自动挂载文件系统。

/etc/fstab文件简介

- **/etc/fstab**文件包含了所有磁盘分区以及存储设备的信息。其中包含了磁盘分区和存储设备如何挂载，以及挂载在什么目录上的信息。**/etc/fstab**文件是一个简单的文本文件，必须要以**root**用户登录才可以编辑该文件。
- 如果在**Linux**系统中不能访问**Windows**的分区，或者做为一名普通用户，不能挂载光驱和向软盘中写入数据，或者在管理**CD-RW**的过程中遇到了问题，就有可能是错误地配置了**/etc/fstab**文件，通常可以通过编辑**/etc/fstab**文件来解决前面提到的问题。

/etc/fstab文件简介

- /etc/fstab文件是一个简单的文本文件，可以用任何文本编辑器去编辑它，但是必须要以root用户登录才可以编辑该文件。
- 由于每一台计算机系统的磁盘分区和设备属性不同，所以/etc/fstab文件也不一样，但是基本的结构总是相似的。每一行都包含着一个设备或磁盘分区的信息，每一行又有多个列的信息。

```
/dev/sda5  /mnt/kk  ext4  defaults  0  0
```

/etc/fstab文件构成

1. 设备

使用设备名称和UUID号表示都可以。如果要查看某设备的UUID号，可以用下面命令：

```
ls -l /dev/disk/by-uuid 或 blkid
```

2. 挂载目录

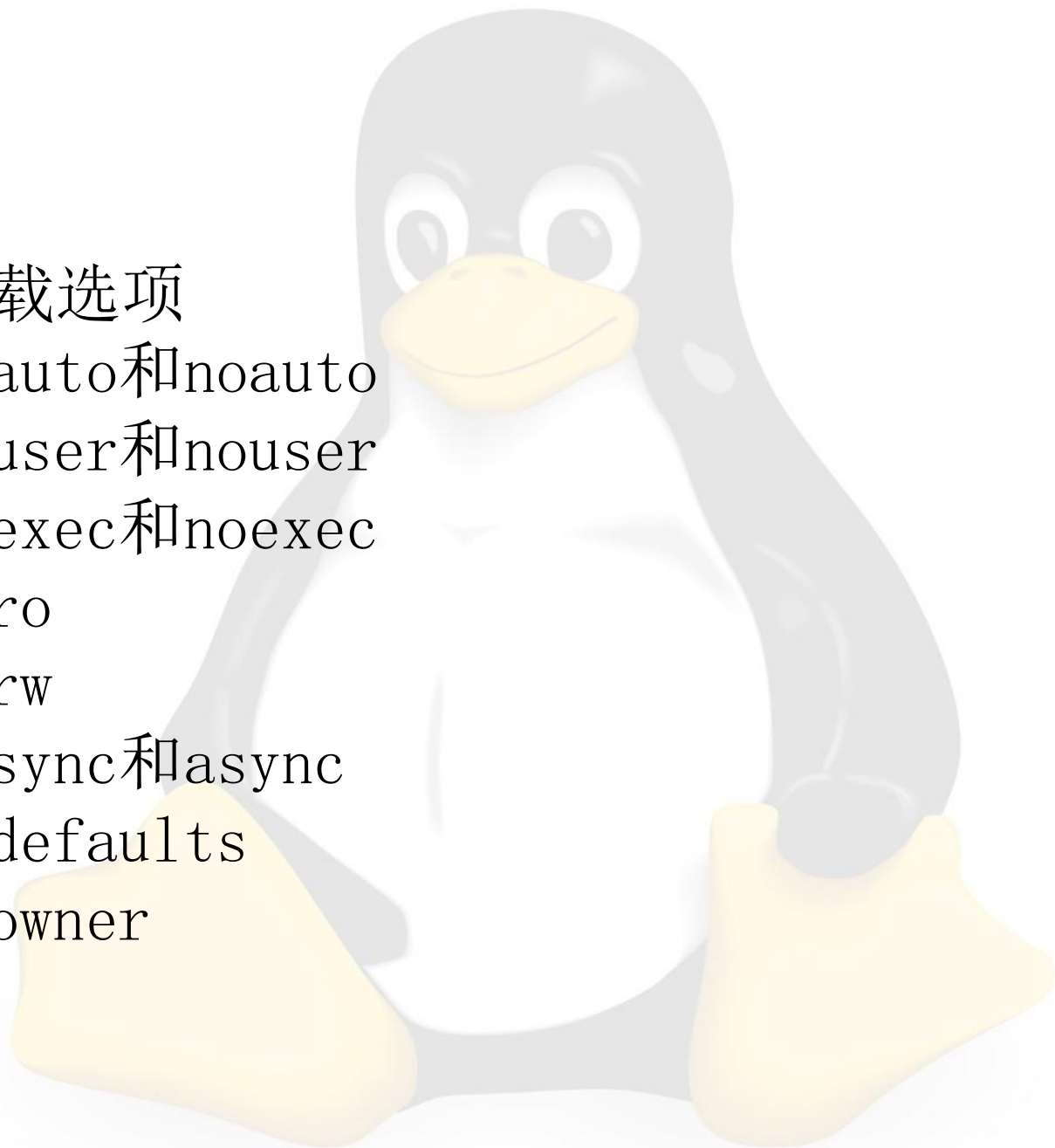
Linux系统为每个设备或分区设定了挂载目录。

3. 文件系统类型

Linux系统为每个设备或分区指定了文件系统类型。

4. 挂载选项

- (1) auto和noauto
- (2) user和nouser
- (3) exec和noexec
- (4) ro
- (5) rw
- (6) sync和async
- (7) defaults
- (8) owner



5. 转储选项

dump选项检查文件系统并用一个数字来决定该文件系统是否需要备份。如果它是0，dump将会忽略该文件系统，不做备份。

6. 文件系统检查选项

fsck选项通过检验第6项中的数字来决定以何种顺序检查文件系统，如果它是0，fsck将不检查该文件系统。根文件系统（“/”）的默认值为1，其他文件系统可以为2-9。

设置开机自动挂载文件系统

- 要实现开机自动挂载文件系统，需要在 **/etc/fstab** 文件中添加该磁盘分区的相关信息，可以通过提供设备名称、**UUID** 和卷标实现，设置完成重启计算机系统以后，文件系统将会自动挂载。

1. 使用设备名

- 编辑/etc/fstab文件，在该文件末尾添加下列内容。

```
/dev/sda5  /mnt/www  xfs  defaults  1  2
```

2. 使用UUID

- 先使用以下命令查看磁盘分区/dev/sda5的UUID信息。

```
[root@rhel ~]# ls -l /dev/disk/by-uuid
```

总用量 0

```
lrwxrwxrwx. 1 root root 10 12月 26 14:56 500f998a-a4c9-43d8-b468-109a42787ec8
```

```
-> ../../sda2
```

```
lrwxrwxrwx. 1 root root 10 12月 26 14:56 b6d87c93-6c54-48f3-9295-ec0f7b92c9f3
```

```
-> ../../sda1
```

```
lrwxrwxrwx. 1 root root 10 12月 26 14:56 d8636446-9878-4437-b43c-50d71af8beba
```

```
-> ../../sda3
```

```
lrwxrwxrwx. 1 root root 10 12月 26 15:02 f084b8a6-c9ab-425b-99da-b64cb35e4fc2
```

```
-> ../../sda5
```

//磁盘分区/dev/sda5的UUID是f084b8a6-c9ab-425b-99da-b64cb35e4fc2

然后编辑/etc/fstab文件，在该文件末尾添加下列内容。

```
UUID= f084b8a6-c9ab-425b-99da-b64cb35e4fc2
```

```
/mnt/www    xfs  defaults    1 2
```

3. 使用卷标

- 先使用以下命令查看磁盘分区/dev/sda5的卷标信息。

```
[root@rhel ~]#xfs_admin -l /dev/sda5
```

```
label = "www"
```

//磁盘分区/dev/sda5的卷标是www

然后编辑/etc/fstab文件，在该文件末尾添加下列内容。

```
LABEL=www /mnt/www xfs defaults 1 2
```

注意：为xfs文件系统/dev/sda5设置卷标使用命令
xfs_admin -L www /dev/sda5。为ext4文件系统/dev/sda6
设置卷标使用命令**e2label /dev/sda6 ftp**。

8.6 使用交换空间

- Linux系统中的交换空间在物理内存被用完时使用。如果系统需要更多的内存资源，而物理内存已经用完，内存中不活跃的页就会被转移到交换空间中。虽然交换空间可以为带有少量内存的计算机提供帮助，但是这种方法不应该被当做是对内存的取代。
- 用户有时需要在安装Linux系统后添加更多的交换空间，可以通过添加一个交换分区（推荐优先使用）或添加一个交换文件来实现。交换空间的总大小一般为计算机物理内存的1~2倍左右，计算机物理内存越大，倍数越小。

使用交换分区



1.添加交换分区：

添加一个交换分区（/dev/sda5）的具体步骤如下：

（1）创建磁盘分区

已经使用fdisk命令创建好了/dev/sda5分区，使用以下命令查看/dev/sda5分区信息。

```
[root@rhel ~]# fdisk -l /dev/sda
```


(2) 创建交换分区

假设将分区/dev/sda5创建为交换分区，在Shell提示下以root用户身份输入以下命令。

```
[root@rhel ~]# mkswap /dev/sda5
```

```
Setting up swapspace version 1, size = 10236 KiB
```

```
no label, UUID=8406e55c-ca76-44a0-b531-62836e208cfb
```

```
//将/dev/sda5分区创建为交换分区
```

```
[root@rhel ~]# free
```

	total	used	free	shared	buffers	cached
Mem:	1027848	941104	86744		0	73424
-/+ buffers/cache:		311472	716376			
Swap:	2097148	588	2096560			

```
//因为当前还没有启动交换分区，所以使用free命令无法看到
```

(3) 启用交换分区

输入以下命令启用交换分区/dev/sda5。


```
[root@rhel ~]# swapon /dev/sda5
```

//启动交换分区/dev/sda5

```
[root@rhel ~]# free
```

	total	used	free	shared	buffers	cached
Mem:	1027848	941104	86744		0	73436
-/+ buffers/cache:		311448	716400			
Swap:	2107384	588	2106796			

//因为当前已经启动了交换分区/dev/sda5所以总的交换分区容量已经增加了



(4) 确认已经启用交换分区
创建并启用了交换分区之后，使用如下命令查看交换分区是否已启用。

```
[root@rhel ~]# cat /proc/swaps
```

Filename	Type	Size	Used	Priority
/dev/sda3	partition	2097148	588	0
/dev/sda5	partition	10236	0	-1

//可以看到当前计算机的SWAP分区由/dev/sda3和/dev/sda5这两个分区构成

(5) 编辑/etc/fstab文件

如果要在系统引导时启用交换分区，编辑/etc/fstab文件添加如下内容。然后在系统下次引导时，就会启用新建的交换分区。

```
/dev/sda5    swap    swap    defaults    0    0
```

2.删除交换分区

（1）禁用交换分区

在Shell提示下以root用户身份输入以下命令禁用交换分区（这里的/dev/sdb5是交换分区）。

```
[root@rhel ~]# swapoff /dev/sda5
```

```
[root@rhel ~]# free
```

	total	used	free	shared	buffers	cached	
Mem:	1027848	873644	154204		0	73572	488988
-/+ buffers/cache:		311084	716764				
Swap:	2097148	604	2096544				

（2）编辑/etc/fstab文件

如果要在系统引导时不启用交换分区，编辑/etc/fstab文件删除如下内容。然后在系统下次引导时，就不会启用交换分区。

/dev/sda5	swap	swap	defaults	0	0
-----------	------	------	----------	---	---

使用交换文件

1.添加交换文件

(1) 创建/swapfile文件

将大小乘以1024来判定块的大小。例如，大小为68MB的交换文件的块大小为65536。

在Shell提示下以root用户身份输入以下命令，其中的count等于想要输入的块大小。

```
[root@rhel ~]# dd if=/dev/zero of=/swapfile bs=1024 count=66536
```

记录了66536+0 的读入

记录了66536+0 的写出

68132864字节(68 MB)已复制，0.447235 秒，152 MB/秒

```
[root@rhel ~]# ls /
```

```
bin  dev  home  lost+found  mnt  proc  run  srv      sys  usr
boot  etc  lib   media      opt  root  sbin  swapfile  tmp  var
```

//在“/”目录下可以看到文件swapfile

（2）创建交换文件

使用以下命令创建交换文件。

```
[root@rhel ~]# mkswap /swapfile
```

Setting up swapspace version 1, size = 66532 KiB

no label, UUID=1587ac78-d291-4fac-ab1e-ac89a7d9c658

（3）启用交换文件

使用如下命令启用交换文件。

```
[root@rhel ~]# swapon /swapfile
```

(4) 查看交换文件是否启用

```
[root@rhel ~]# free
```

	total	used	free	shared	buffers	cached
Mem:	1027848	943240	84608		0	73552
-/+ buffers/cache:		314164	713684			
Swap:	2163680	604	2163076			

```
[root@rhel ~]# cat /proc/swaps
```

Filename	Type	Size	Used	Priority
/dev/sda3	partition	2097148	604	0
/swapfile	file	66532	0	-1

(5) 编辑/etc/fstab文件

如果要在系统引导时启用交换文件，编辑/etc/fstab文件添加如下内容。
然后在系统下次引导时，就会启用新建的交换文件。

/swapfile	swap	swap	defaults	0	0
-----------	------	------	----------	---	---

2.删除交换文件

（1）禁用交换文件

在Shell提示下以root用户身份执行以下命令来禁用交换文件（这里的 /swapfile 是交换文件）。

```
[root@rhel ~]# swapoff /swapfile
```

```
[root@rhel ~]# free
```

	total	used	free	shared	buffers	cached
Mem:	1027848	943268	84580		0	73552
-/+ buffers/cache:		314192	713656			
Swap:	2097148	604	2096544			

（2）删除/swapfile文件

输入以下命令删除/swapfile文件。

```
[root@rhel ~]# rm -rf /swapfile
```

（3）编辑/etc/fstab文件

如果要在系统引导时不启用交换文件，编辑/etc/fstab文件删除如下内容。然后在系统下次引导时，就不会启用交换文件。

/swapfile	swap	swap	defaults	0	0
-----------	------	------	----------	---	---