

## 第 5 章 磁盘管理

系统管理员的一项重要工作就是管理磁盘。磁盘作为存储数据的重要载体，在如今日渐庞大的软件资源面前显得尤为重要。随着硬件成本的逐年下降，磁盘容量越来越大，但同时磁盘管理难度也越来越高。良好的磁盘管理可以进一步节省存储空间、提高系统效能、节约成本。磁盘管理通常涉及磁盘的分区、格式化、空间管理等内容。

### 5.1 常用的磁盘管理工具

磁盘管理工具是系统管理员需要经常使用的软件，是完成磁盘管理的重要手段。常用的磁盘管理工具包括 fdisk、Partition Magic、parted、mkfs、e2fsck 等。

#### 5.1.1 Linux 磁盘分区简介

由于硬盘容量动辄几十 GB，为了方便管理，通常将硬盘分成若干分区。对于普通用户而言，每个分区都可以视为独立的磁盘来使用。硬盘的分区方案被记录到“磁盘分区表”中。通常磁盘分区表由 4 个部分组成，每个部分定义一个分区的信息。因此，原始概念中一个硬盘最多只能建立 4 个分区，称为“主分区”。当系统中存在多个主分区时，必须指定一个主分区为“活动分区”。活动分区上的操作系统在系统引导时将被自动引导。

由于硬盘容量越来越大，仅仅 4 个分区已无法满足用户的需要，为此引入了“扩展分区”的概念。扩展分区由扩展磁盘分区表维护，可以在扩展分区内划分若干更小的称为“逻辑分区”的分区。逻辑分区没有数量的限制，如果删除了扩展分区，其下的所有逻辑分区都将被删除。扩展分区必须建立在主分区上，即扩展分区必须属于 4 个主分区之一。主分区、扩展分区与逻辑分区关系如图 5.1 所示。

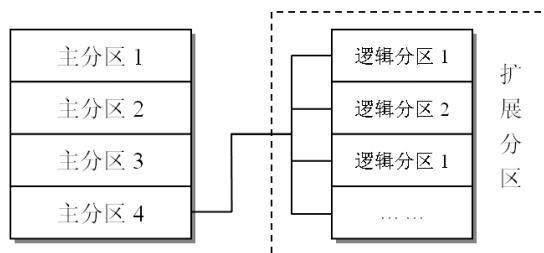


图 5.1 主分区、扩展分区与逻辑分区

在 Red Hat Enterprise Linux 5 安装到硬盘上以前，必须建立硬盘分区。理论上，在硬盘空间足够时，可以建立任意数量的分区(挂载点)，但除非在极为特殊的条件下，否则建议至少建立以下三个分区：

- ❑ **swap 分区：**swap 分区的系统标识为 82。swap 分区实现上是虚拟内存的一部份。所谓虚拟内存(Virtual Memor)技术是指在物理内存无法提供足够的处理空间时，多余的数据就会暂时写入硬盘指定的分区中，待物理内存可处理时再由该分区中将数据移入。虚拟内存技术突破了物理

内存的限制，使程序可以使用大于实际物理内存的空间。一般情况下，swap 分区应该大于或等于物理内存，不应小于 32 MB。建议物理内存在 2G 以下时，swap 分区设为物理内存的 2~2.5 倍。如果物理内存在 2G 以上，swap 分区一般设为与物理内存相同即可。

- ❑ boot 分区：boot 分区中存放着操作系统的内核。在 Red Hat Enterprise Linux 5 中系统内核文件为 /boot/vmlinuz-2.6.18-8.el5xen，建议使用 32 MB 以上的空间。
- ❑ 根分区 (/)：根分区是整个操作系统的根目录，几乎所有的文件都位于此目录之下，因此它的容量越大越好，建议将硬盘中剩余的空间都提供给根分区使用。

Linux 还提供了多种分区创建方案，见表 5-1。

表 5-1 Linux常用的分区方案

分区方案	说明
根分区、swap分区	适用于磁盘空间有限的计算机
根分区、/boot分区、swap分区	较大磁盘空间的典型配置，也是Red Hat Enterprise Linux 默认配置
根分区、/boot分区、/var分区、swap分区	可以避免日志文件大小失控
根分区、/boot分区、/home分区、swap分区	对于一台为许多用户提供服务的计算机，可以帮助控制用户占用的空间量

注意：在安装 Red Hat Enterprise Linux 5 过程中，系统默认把根 (/) 和 /boot 目录安装到独立的分区上。

Linux 支持的分区类型众多，每一种分区类型对应一个系统标识号。系统标识号由 2 位 16 进制数表示，包含了操作系统类型、文件系统类型、分区容量等信息。常用分区类型与系统标识如表 5-2 所示。

表 5-2 常用分区类型与系统标识

系统标识	分区类型	系统标识	分区类型	系统标识	分区类型	系统标识	分区类型
0	Empty	1e	Hidden Win95 FAT16	80	Old Minix	be	Solaris boot
1	FAT12	24	NEC DOS	81	Minix / old Linux	bf	Solaris
2	XENIX root	39	Plan 9	82	Linux swap / So	c1	DRDOS/sec (FAT-12)
3	XENIX usr	3c	PartitionMagic	83	Linux	c4	DRDOS/sec (FAT-16)
4	FAT16 <32M	40	Venix 80286	84	OS/2 hidden C:	c6	DRDOS/sec (FAT-16)
5	Extended	41	PPC PReP Boot	85	Linux extended	c7	Syrinx
6	FAT16	42	SFS	86	NTFS volume set	da	Non-FS data
7	HPFS/NTFS	4d	QNX4.x	87	NTFS volume set	db	CP/M / CTOS / .
8	AIX	4e	QNX4.x 2nd part	88	Linux plaintext	de	Dell Utility
9	AIX bootable	4f	QNX4.x 3rd part	8e	Linux LVM	df	BootIt
a	OS/2 Boot Manager	50	OnTrack DM	93	Amoeba	e1	DOS access
b	Win95 FAT32	51	OnTrack DM6 Aux	94	Amoeba BBT	e3	DOS R/O
c	Win95 FAT32 (LBA)	52	CP/M	9f	BSD/OS	e4	SpeedStor

e	Win95 FAT16 (LBA)	53	OnTrack DM6 Aux	a0	IBM Thinkpad hi	eb	BeOS fs
f	W95 Ext'd (LBA)	54	OnTrackDM6	a5	FreeBSD	ee	EFI GPT
10	OPUS	55	EZ-Drive	a6	OpenBSD	ef	EFI (FAT-12/16/)
11	Hidden FAT12	56	Golden Bow	a7	NeXTSTEP	f0	Linux/PA-RISC b
12	Compaq diagnostics	5c	Priam Edisk	a8	Darwin UFS	f1	SpeedStor
14	Hidden FAT16 <32M	61	SpeedStor	a9	NetBSD	f4	SpeedStor
16	Hidden FAT16	63	GNU HURD or Sys	ab	Darwin boot	f2	DOS secondary
17	Hidden HPFS/NTF	64	Novell Netware	b7	BSDI fs	fd	Linux raid auto
18	AST SmartSleep	65	Novell Netware	b8	BSDI swap	fe	LANstep
1b	Hidden Win95 FAT3	70	DiskSecure Multi-Boot	bb	Boot Wizard hidden	ff	BBT

### 5.1.2 fdisk 磁盘分区工具简介

fdisk 是一种功能强大的磁盘分区工具，不仅适合于 Linux 系统，在 Windows 及 DOS 中也有广泛应用。fdisk 命令格式如下

```
fdisk [-u] [-b sectorsize] [-C cyls] [-H heads] [-S sects] 设备名
fdisk -l [-u] [device ...]
fdisk -s partition ...
fdisk -v
```

其中常用的设备名包括下面几种：

- ❑ /dev/had：第一块 IDE 硬盘
- ❑ /dev/hdb：第二块 IDE 硬盘
- ❑ /dev/sda：第一块 SCSI 硬盘
- ❑ /dev/sdb：第二块 SCSI 硬盘

设备名的命名规则如下：对于 IDE 设备，设备名为/dev/hd[a-h]；对于 SCSI 设备，设备名为/dev/sd[a-p]；对于 ESDI 设备，使用/dev/ed[a-d]；XT 设备使用/dev/xd[ab]。其中的设备是指整个物理磁盘设备。命令 fdisk 常用选项及说明如表 5-3 所示。

表 5-3 命令fdisk常用选项及说明

选项	说明
-b	定义磁盘扇区的大小，有效值包括512,1024和2048。磁盘扇区大小是系统内核已知的参数，设置该选项仅仅是对以前内核版本的支持，以及满足重载内核的需要。
-C	定义磁盘柱面数。
-H	定义磁盘分区的磁头数，该值专指分区中使用的磁头数目。有效值为255或16。
-S	定义磁盘分区中每磁道包含的扇区数目，有效值为63。
-l	列出指定设备的分区表并退出。如果没有指定设备，则使用/proc/partitions文件提供的信息。
-u	在列出分区表时，以扇区为单位显示而不是以柱面为单位。
-s	在标准输出设备中列出指定分区的大小，默认以block为单位。
-v	显示版本信息。

例如，使用选项“-l”显示当前设备的分区表并退出，命令行如下：

```
# fdisk -l
Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
/dev/sda1    *           1           13        104391   83  Linux
/dev/sda2             14          1044       8281507+   8e  Linux LVM
```

使用选项“-s”可以显示指定分区的大小。在上例中分别显示/dev/sda1 和/dev/sda2 分区大小，命令行如下：

```
# fdisk -s /dev/sda1
104391
# fdisk -s /dev/sda2
8281507
```

如果没有指定具体的分区，而仅指定设备名，则显示该设备中所有分区大小的总和，命令行如下：

```
# fdisk -s /dev/sda
8388608
```

使用选项“-v”可以显示 fdisk 的版本号，命令行如下：

```
# fdisk -v
fdisk (util-linux 2.13-pre7)
```

### 5.1.3 进入 fdisk 交互模式

使用命令“fdisk 设备名”，系统会进入分区管理交互模式。用户可以通过输入指令，执行相应的磁盘管理操作，fdisk 操作指令如表 5-4 所示。

表 5-4 fdisk操作指令

操作指令	说明
a	设置或取消可引导标志，硬盘驱动器应至少有一个主分区设为可引导分区。
d	删除一个分区。
l	列举出已知的分区类型，共有100多个类型，其中83是Linux分区，82是Linux swap分区。
m	显示可使用的fdisk命令。
n	创建一个新分区。
p	显示当前的分区表。
q	退出fdisk，但不自动保存修改。
t	更改分区的ID。
v	对当前的分区表进行验证。
w	保存修改结果并退出fdisk。
b	编辑bsd磁盘标签。
c	设置DOS兼容标志。
o	创建一个新的DOS分区表。
s	创建一个空的Sun磁盘标签。
u	改变显示单位。

例如对/dev/sda 设备使用 fdisk 命令，命令行如下：

```
# fdisk /dev/sda
The number of cylinders for this disk is set to 1044.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
```

2) booting and partitioning software from other OSs  
(e.g., DOS FDISK, OS/2 FDISK)

Command (m for help):

可以看到，fdisk 命令进入交互模式后，首先提示当前的磁盘柱面数为 1044。由于当前磁盘的柱面数大于 1024，可能会与一些软件产生冲突，包括在系统启动时运行的软件（例如旧版的 LILO 程序）以及其他操作系统的启动和分区软件（例如 DOS 或 OS/2 下的 fdisk 命令）。

### 1. m 指令

在“Command(m for help):”提示符下键入“m”，fdisk 将显示所有指令列表：

Command (m for help): m

Command action

```
a  toggle a bootable flag
b  edit bsd disklabel
c  toggle the dos compatibility flag
d  delete a partition
l  list known partition types
m  print this menu
n  add a new partition
o  create a new empty DOS partition table
p  print the partition table
q  quit without saving changes
s  create a new empty Sun disklabel
t  change a partition's system id
u  change display/entry units
v  verify the partition table
w  write table to disk and exit
x  extra functionality (experts only)
```

### 2. p 指令

输入命令“p”，列出当前/dev/sda 设备上的分区信息，如下所示：

Command (m for help):p

Disk /dev/sda: 8589 MB, 8589934592 bytes

255 heads, 63 sectors/track, 1044 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	13	104391	83	Linux
/dev/sda2		14	1044	8281507+	8e	Linux LVM

可以看到当前/dev/sda 设备容量为 8589 MB，包括 255 个磁头，每磁道 63 扇区，磁盘柱面总共 1044。列表以柱面为单位进行显示，其中 1 柱面等于 8225280 字节。

### 3. u 指令

如果用户希望以扇区为单位进行显示，可以在执行 fdisk 命令时加入“-u”选项，命令行如下：

```
# fdisk -u /dev/sda
```

The number of cylinders for this disk is set to 1044.

There is nothing wrong with that, but this is larger than 1024,  
and could in certain setups cause problems with:

- 1) software that runs at boot time (e.g., old versions of LILO)
- 2) booting and partitioning software from other OSs  
(e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p //输入 p 指令

Disk /dev/sda: 8589 MB, 8589934592 bytes  
255 heads, 63 sectors/track, 1044 cylinders, total 16777216 sectors  
Units = sectors of 1 \* 512 = 512 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	63	208844	104391	83	Linux
/dev/sda2		208845	16771859	8281507+	8e	Linux LVM

可以看到显示内容已经改为以扇区为单位，其中 1 扇区等于 512 字节。用户也可以在进入 fdisk 交互模式后执行“u”指令完成单位切换，例如：

Command (m for help): u  
Changing display/entry units to cylinders //将显示单位改为柱面

Command (m for help): u  
Changing display/entry units to sectors //将显示单位改为扇区

#### 4. d 指令

使用指令“d”可以删除已经存在的分区，例如：

Command (m for help): p

Disk /dev/sda: 8589 MB, 8589934592 bytes  
255 heads, 63 sectors/track, 1044 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	13	104391	83	Linux
/dev/sda2		14	1044	8281507+	8e	Linux LVM

Command (m for help): d  
Partition number (1-4): 1 //输入要删除的分区号，其中 1 指/dev/sda1 分区，2 指/dev/sda2 分区  
Command (m for help): p

Disk /dev/sda: 8589 MB, 8589934592 bytes  
255 heads, 63 sectors/track, 1044 cylinders, total 16777216 sectors  
Units = sectors of 1 \* 512 = 512 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda2		208845	16771859	8281507+	8e	Linux LVM

//dev/sda1 分区已被删除

注意：如果出现误删除，只需用“q”指令退出 fdisk，则本次所做的所有操作均不会生效。用户只需重新进入 fdisk 并重新进行配置。如果使用“w”指令退出 fdisk，则 fdisk 会在退出前进行写入操作，删除的分区将无法恢复。另外，如果删除了扩展分区，则扩展分区下的逻辑分区都会被删除。

#### 5. n 指令

使用指令“n”可以创建一个新分区，例如：

```

Command (m for help):p                //显示当前磁盘分区

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders, total 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda2            208845      16771859      8281507+   8e  Linux LVM

Command (m for help): n                //新建一个分区
Command action                        //指令“e”表示创建扩展分区，指令“p”键表示创建主分区
  e   extended
  p   primary partition (1-4)
p                                       //创建主分区
Partition number (1-4): 1              //指定分区号为 1
First sector (63-16777215, default 63): //输入扇区开始位置，默认为 63
Using default value 63
Last sector or +size or +sizeM or +sizeK (63-208844, default 208844): //输入扇区结束位置，默认为 208844
Using default value 208844

Command (m for help): p                //显示当前磁盘分区

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders, total 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes

Device      Boot      Start         End      Blocks   Id  System
/dev/sda1            63          208844      104391    83  Linux
/dev/sda2            208845      16771859      8281507+   8e  Linux LVM

```

## 6. t 指令

对于新建的分区，系统默认的分区类型为 83，即 Linux 分区。如果希望设置其他的分区类型，可以使用“-t”指令：

```

Command (m for help): t                //改变系统标识码
Partition number (1-4): 1              //操作的分区
Hex code (type L to list codes): l     //列出所有可用的标识码
0  Empty                1e  Hidden W95 FAT1  80  Old Minix          be  Solaris boot
1  FAT12                24  NEC DOS           81  Minix / old Linux  bf  Solaris
2  XENIX root           39  Plan 9            82  Linux swap / So   c1  DRDOS/sec (FAT-12)
3  XENIX usr            3c  PartitionMagic    83  Linux              c4  DRDOS/sec (FAT-16)
4  FAT16 <32M           40  Venix 80286       84  OS/2 hidden C:    c6  DRDOS/sec (FAT-16)
5  Extended             41  PPC PReP Boot     85  Linux extended    c7  Syrix
6  FAT16                42  SFS               86  NTFS volume set   da  Non-FS data
7  HPFS/NTFS            4d  QNX4.x            87  NTFS volume set   db  CP/M / CTOS / .
8  AIX                  4e  QNX4.x 2nd part   88  Linux plaintext   de  Dell Utility
9  AIX bootable         4f  QNX4.x 3rd part   8e  Linux LVM          df  BootIt
a  OS/2 Boot Manag     50  OnTrack DM        93  Amoeba             e1  DOS access
b  W95 FAT32            51  OnTrack DM6 Aux   94  Amoeba BBT         e3  DOS R/O
c  W95 FAT32 (LBA)      52  CP/M              9f  BSD/OS             e4  SpeedStor
e  W95 FAT16 (LBA)      53  OnTrack DM6 Aux   a0  IBM Thinkpad hi    eb  BeOS fs

```

```

f  W95 Ext'd (LBA)    54  OnTrackDM6      a5  FreeBSD        ee  EFI GPT
10 OPUS                55  EZ-Drive        a6  OpenBSD         ef  EFI (FAT-12/16/
11 Hidden FAT12        56  Golden Bow     a7  NeXTSTEP        f0  Linux/PA-RISC b
12 Compaq diagnost    5c  Priam Edisk     a8  Darwin UFS      f1  SpeedStor
14 Hidden FAT16 <3    61  SpeedStor      a9  NetBSD          f4  SpeedStor
16 Hidden FAT16        63  GNU HURD or Sys ab  Darwin boot     f2  DOS secondary
17 Hidden HPFS/NTF     64  Novell Network b7  BSDI fs         fd  Linux raid auto
18 AST SmartSleep      65  Novell Network b8  BSDI swap       fe  LANstep
1b Hidden W95 FAT3     70  DiskSecure Mult bb  Boot Wizard hid ff  BBT
1c Hidden W95 FAT3     75  PC/IX

Hex code (type L to list codes): 8e      //输入 Linux 分区 16 进制标识码 8e
Command (m for help): p                  //显示当前磁盘分区

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders, total 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes

Device      Boot      Start          End      Blocks      Id  System
/dev/sda1                63       208844      104391      8e  Linux LVM      //系统标识码已改为 8e
/dev/sda2          208845      16771859      8281507+     8e  Linux LVM

```

注意：在创建逻辑分区前必须创建扩展分区。

#### 5.1.4 使用 fdisk 划分磁盘分区实例

例如对当前硬盘（/dev/sda，容量 8.5GB）划分磁盘分区：其中主分区 2 个，一个 100MB，一个 5GB，其余空间分配给扩展分区。扩展分区划分为两个逻辑分区，一个 2GB，一个 1GB。所有分区默认类型均为 83，执行命令过程如下：

（1）显示当前设备的分区表

```

# fdisk /dev/sda
The number of cylinders for this disk is set to 1044.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p      //显示当前设备的分区表

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device Boot      Start          End      Blocks      Id  System
//当前/dev/sda 设备分区表为空

```

（2）创建 100MB 的主分区，可以用来作为 /boot 分区，如下所示：

```

Command (m for help): n      //创建分区
Command action              //选择创建分区类型
e   extended
p   primary partition (1-4)
p                              //创建主分区

```



```

Partition number (1-4): 1           //输入分区号
First cylinder (1-1044, default 1): //输入起始柱面
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1044, default 1044): +100M //此处可以输入柱面号也可以直接输入空间大小, 输入 "+100M"

```

```

Command (m for help): p           //显示创建结果

```

```

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	13	104422	83	Linux

//已创建 100M 主分区, 类型 83

(3) 创建 5GB 的主分区, 可以作为根分区使用, 如下所示:

```

Command (m for help): n

```

```

Command action

```

```

  e   extended

```

```

  p   primary partition (1-4)

```

```

p

```

```

Partition number (1-4): 2

```

```

First cylinder (14-1044, default 14):

```

```

Using default value 14

```

```

Last cylinder or +size or +sizeM or +sizeK (14-1044, default 1044): +5000M //输入该分区大小

```

```

Command (m for help): p

```

```

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	13	104422	83	Linux
/dev/sda2		14	622	4891792+	83	Linux

//已创建 5GB 主分区, 类型 83

(4) 主分区创建完毕, 接下来创建扩展分区, 如下所示:

```

Command (m for help): n

```

```

Command action

```

```

  e   extended

```

```

  p   primary partition (1-4)

```

```

e

```

```

//创建扩展分区

```

```

Partition number (1-4): 3           //输入分区号

```

```

First cylinder (623-1044, default 623):

```

```

Using default value 623

```

```

Last cylinder or +size or +sizeM or +sizeK (623-1044, default 1044): //将默认的所有剩余空间分配给扩展分区

```

```

Using default value 1044

```

```

Command (m for help): p

```

```

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	13	104422	83	Linux
/dev/sda2		14	622	4891792+	83	Linux
/dev/sda3		623	1044	3389715	5	Extended //已创建扩展分区, 类型为 5

(5) 扩展分区创建完成后, 可以在扩展分区上创建逻辑分区, 首先创建 2GB 的逻辑分区, 如下所示:

```
Command (m for help): n
Command action
  l   logical (5 or over)
  p   primary partition (1-4)
l                                     //输入 "l", 创建逻辑分区
First cylinder (623-1044, default 623):
Using default value 623
Last cylinder or +size or +sizeM or +sizeK (623-1044, default 1044): +2000M //创建 2GB 的逻辑分区

Command (m for help): p

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device      Boot      Start         End      Blocks   Id  System
/dev/sda1   Boot          1          13       104422   83  Linux
/dev/sda2   Boot        14          622     4891792+  83  Linux
/dev/sda3   Boot        623         1044     3389715    5  Extended
/dev/sda5   Boot        623          866     1959929+  83  Linux //已创建 2GB 逻辑分区, 类型 83
```

(6) 将剩余空间 (约 1GB) 分配给最后一个逻辑分区, 如下所示:

```
Command (m for help): n
Command action
  l   logical (5 or over)
  p   primary partition (1-4)
l                                     //创建逻辑分区
First cylinder (867-1044, default 867):
Using default value 867
Last cylinder or +size or +sizeM or +sizeK (867-1044, default 1044): //将扩展分区剩余空间分配给最后一个分区
Using default value 1044

Command (m for help): p                                     //显示创建的分区表

Disk /dev/sda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device      Boot      Start         End      Blocks   Id  System
/dev/sda1   Boot          1          13       104422   83  Linux
/dev/sda2   Boot        14          622     4891792+  83  Linux
/dev/sda3   Boot        623         1044     3389715    5  Extended
/dev/sda5   Boot        623          866     1959929+  83  Linux
```

```
/dev/sda6      867      1044      1429784+  83  Linux
```

(7) 分区表编辑完成, 可以按 “w” 键, 执行写入操作并退出指令:

```
Command (m for help): w
```

```
The partition table has been altered!
```

如果放弃本次所做的修改, 按 “q” 键, 直接退出 fdisk 交互模式:

```
Command (m for help): q
```

### 5.1.5 使用 Partition Magic 管理磁盘分区

Partition Magic 是 Windows 下功能强大的磁盘分区工具, 支持 FAT/FAT32/NTFS/LINUX/UNIX 等多种磁盘分区格式。Partition Magic 不仅可以对磁盘进行分区, 还可以在不损坏分区内数据的情况下, 对分区容量进行调整以及实现分区类型的转换。用户利用 Partition Magic, 在 Windows 环境下可以为 Linux 系统创建磁盘分区, 具体操作步骤如下:

#### 1. 使用 Partition Magic 前需做好备份

在进行分区调整过程中, 系统会一直处于风险之中, 一些重要的数据必须事先做好备份。如果在分区调整期间出现断电、程序突然终止等意外情况, 会对分区表造成较为严重的损坏, 甚至造成系统崩溃。为此需要事先把硬盘中的重要数据做好备份, 关闭与 Partition Magic 无关的其他程序, 确保硬盘中有足够的剩余空间, 条件具备时应接上 UPS 电源。

#### 2. 使用 Partition Magic 重新分配 Windows 分区, 并建立 Linux 分区。

例如从一台 Windows XP 主机上的 E 盘剩余空间上划分 6GB 分给 Linux。由于空间有限, 在该 Linux 系统中只建立两个分区, 一个 Linux 根分区 (分区类型 ext3) 约 5.4GB, 一个交换分区 (分区类型 Linux swap) 约 800MB (交换分区容量通常是主机物理内存的 2~2.5 倍, 若主机内存在 2G 以上, 交换分区与物理内存相当即可)。

(1) 启动 Partition Magic, 打开 Partition Magic 主界面, 如图 5.2 所示。可以看到, 系统中存在一个主分区 (IBM\_PRELOAD C:) 约 20GB, NTFS 文件系统。扩展分区由 D 盘 (约 30GB, NTFS 文件系统) 和 E 盘 (约 40GB, NTFS 文件系统) 以及 IBM\_SERVICE 分区 (IBM 专用分区, FAT 文件系统) 组成。

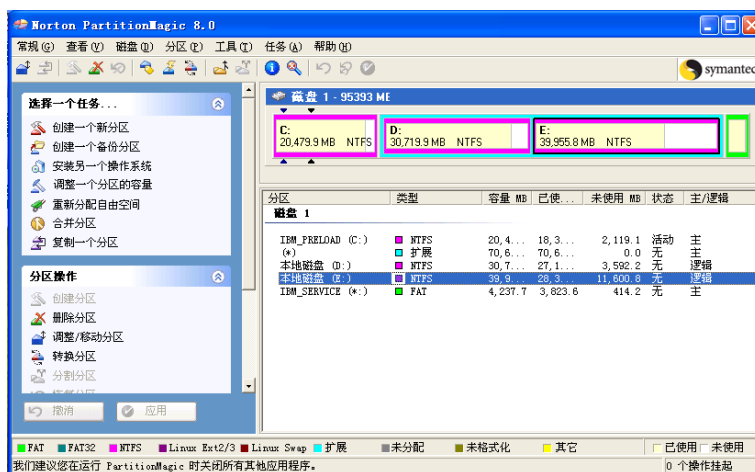


图 5.2 磁盘原始分区状态

(2) 为了安装 Linux, 需从 E 盘中划分一部分 (约 6GB) 的磁盘空间。选择 E 盘后, 单击鼠标右

键，然后在弹出的快捷菜单中选择【调整容量/移动】选项，如图 5.3 所示。

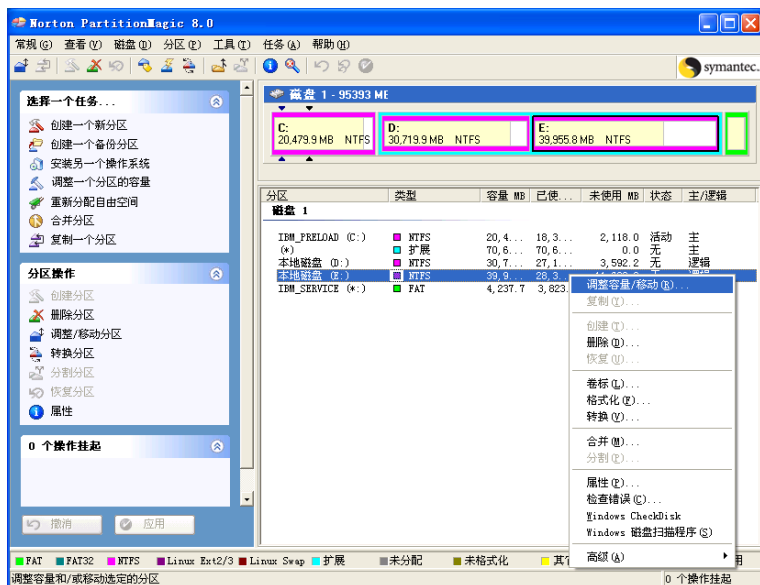


图 5.3 选择需调整容量的磁盘分区

(3) Partition Magic 将弹出如图 5.4 所示的【调整容量/移动分区】对话框，可以看到 E 盘的最大容量为 39955.8MB，其中已经使用了 30476.2MB。

(4) 按【自由空间之后】的微调按钮或拖动容量显示条右侧的箭头，将容量设为 6GB 左右（6231.1MB），如图 5.5 所示。单击【确定】按钮，返回到主窗口。

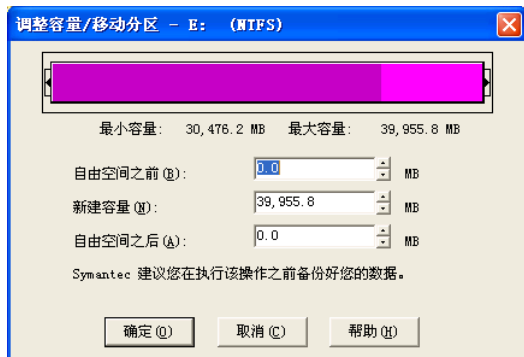


图 5.4 调整前 E 盘的容量

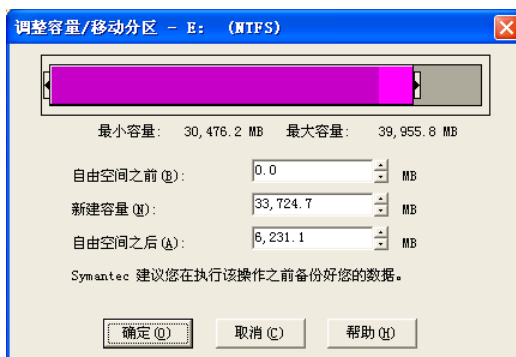


图 5.5 调整 E 盘容量之后

从图 5.6 可以看到，E 盘的容量已由约 39GB 调整为约 33GB，有约 6GB 的磁盘空间已经被划出。被划出的磁盘空间位于 E 盘下面，标识为未分配。

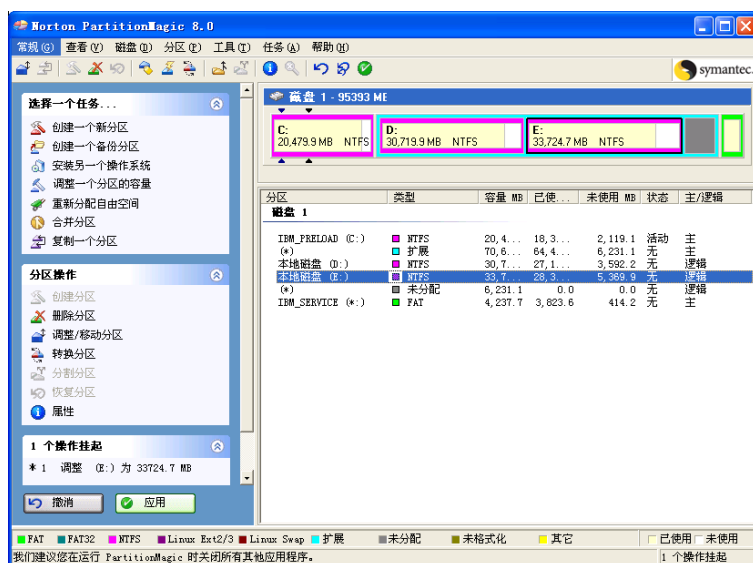


图 5.6 调整空闲空间完毕

(5) 选择标识为【未分配】的磁盘空间，单击鼠标右键，在弹出的快捷菜单中选择【创建】选项，如图 5.7 所示。

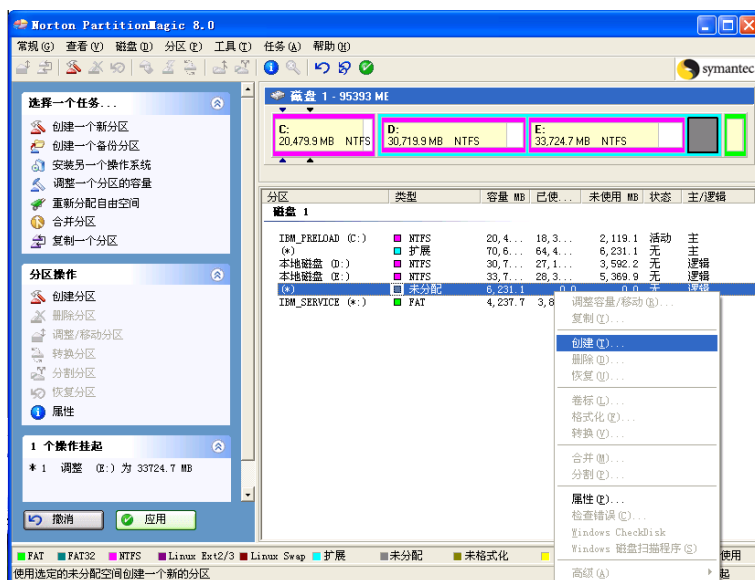


图 5.7 开始创建 Linux 分区

(6) 在弹出的【创建分区】对话框中，选择创建【逻辑分区】，分区类型为“Linux Ext3”，通过微调按键将容量设为约 5.4GB，如图 5.8 所示。该分区将作为 Linux 的根分区，单击【确定】按钮，返回主界面。

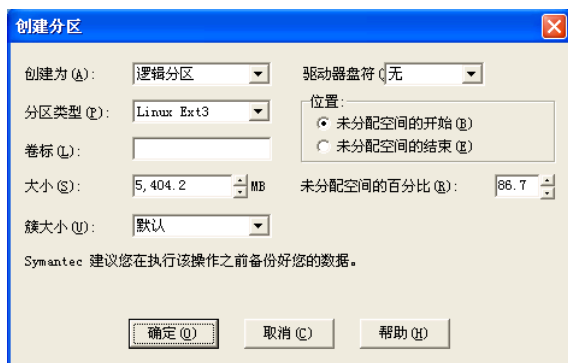


图 5.8 创建 Linux Ext3 分区

(7) 从图 5.9 中可以看到，Partition Magic 已经创建了约 5.4GB 的 Linux Ext3 扩展分区。选择标识为【未分配】的磁盘空间，单击右键，然后选择【创建】选项，继续创建 Linux 交换分区。

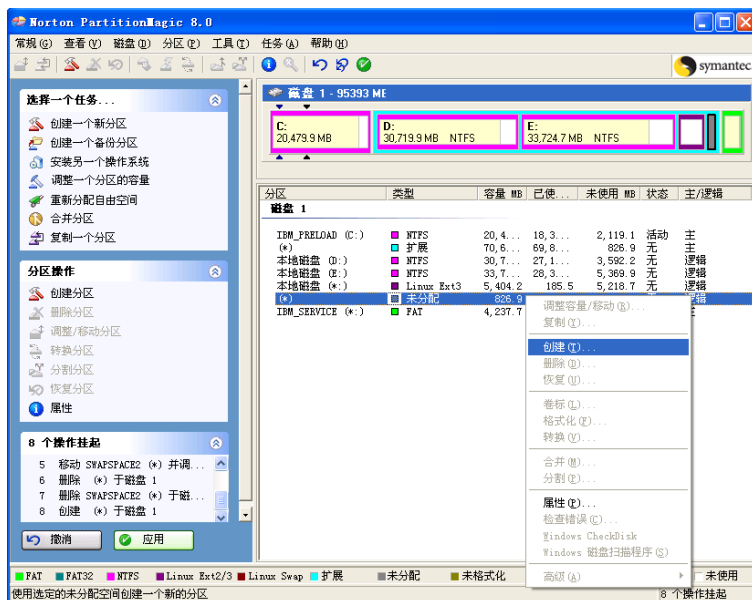


图 5.9 在剩余未分配空间中创建分区

(8) 在弹出的【创建分区】对话框中，选择“逻辑分区”，分区类型为“Linux Swap”，将剩余空间的 826.8MB 空间全部分配给该分区，如图 5.10 所示。单击【确定】按钮，返回主窗口。

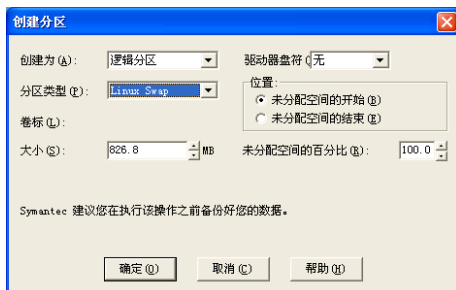


图 5.10 创建 Linux Swap 分区

如图 5.11 所示，系统已经创建了两个 Linux 分区，一个约 5.4GB 的 Linux Ext3 分区，一个约 800MB 的 Linux Swap 分区。

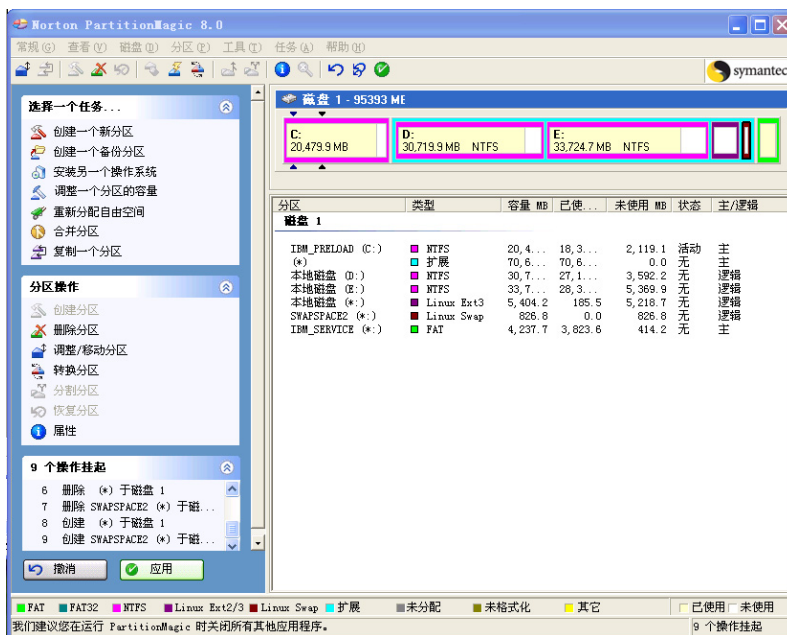


图 5.11 分区划分完毕

(10) 至此分区已重新划分完毕，但并没有实际生效。单击窗口左侧下方的【应用】按钮，弹出如图 5.12 所示的【应用更改】对话框，单击【是】按钮，将弹出【操作过程】对话框。重新启动机器后，系统将使用新的分区设置。

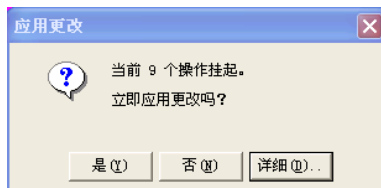


图 5.12 【应用更改】对话框

### 5.1.6 使用 parted 磁盘分区工具简介

parted 是 Linux 下的磁盘分区与分区调整工具，与 Partition Magic 类似，parted 可以对 Linux 下分区进行创建、删除或重新划分大小。使用 parted 命令要求分区所在设备不能正在被使用（即分区不能被挂载，并且交换空间不能被启用），可以使用 umount 命令来卸载分区，使用 swapoff 命令来关闭硬盘驱动器上的交换空间。

```
parted [选项] [设备] [命令 [选项...]]
```

- ☐ -h: 显示帮助信息
- ☐ -i: 交互模式
- ☐ -s: 脚本模式
- ☐ -v: 显示版本信息

例如使用 parted 命令，对/dev/sda 进行分区操作，命令行如下：

```
#parted /dev/sda
GNU Parted 1.8.1
使用 /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
```

`(parted)`

### 5.1.7 进入 parted 交互模式

在启动 `parted` 命令后，系统会进入交互模式，用户可以通过操作指令完成对分区的管理。常用的 `parted` 操作指令如表 5-5 所示：

表 5-5 parted 操作指令及说明

操作指令	说明
<code>check NUMBER</code>	对文件系统进行检测
<code>cp [FROM-DEVICE] FROM-NUMBER TO-NUMBER</code>	把文件系统复制到另一个分区
<code>help [COMMAND]</code>	显示可用的帮助命令或指定命令的帮助信息
<code>mklabel</code>	创建一个新的磁盘标签
<code>mkfs NUMBER FS-TYPE</code>	在分区上创建 FS-TYPE 类型的文件系统
<code>mkpart PART-TYPE [FS-TYPE] START END</code>	创建一个分区
<code>mkpartfs PART-TYPE FS-TYPE START END</code>	创建分区和指定的操作系统
<code>move NUMBER START END</code>	移动分区
<code>name NUMBER NAME</code>	以分区号命名
<code>print [free NUMBER all]</code>	显示分区表
<code>quit</code>	退出
<code>rescue START END</code>	修复丢失的分区
<code>resize NUMBER START END</code>	重新划分分区
<code>rm NUMBER</code>	删除分区
<code>select DEVICE</code>	选择需要配置的设备
<code>set NUMBER FLAG STATE</code>	改变分区的 FLAG 标志
<code>toggle [NUMBER [FLAG]]</code>	切换分区的 FLAG 标志

注意：其中 NUMBER 指 Linux 分区号，主分区号从 1 到 4，逻辑分区号大于等于 5。START 和 END 用于磁盘定位，可以采用例如 4GB 或 10% 方式来表示。如果该值为负数，表示从磁盘的末端开始计算，例如，“-1s”表示从磁盘末端开始的第一个扇区，即磁盘的最后一个扇区。

#### 1. print 指令

例如，启动了 `parted` 后，键入以下命令来查看分区表：

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sda: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
```

Number	Start	End	Size	Type	File system	标志
1	32.3kB	107MB	107MB	主分区	ext3	启动
2	107MB	8587MB	8480MB	主分区		lvm

其中第一行显示了磁盘的类型；第二行显示了磁盘容量；第三行显示逻辑/物理扇区大小，其余各行输出显示磁盘分区表。在分区表中，Number 表示分区号，例如，分区号码为 1 的分区和 `/dev/sda1` 相对应。Start（开始）和 End（结束）值以 MB 为单位。Type（类型）取值为主分区、扩展分区或逻辑分区。Filesystem（文件系统）是文件系统的类型，可以是 `ext2`、`ext3`、`FAT`、`hfs`、`jfs`、`linux-swaps`、`ntfs`、`reiserfs`、`hp-ufs`、`sun-ufs`、`ufs` 等。“标志”列表示分区被设置的标准。可用的标志有：`boot`、`root`、`swap`、`hidden`、`raid`、`lvm` 或 `lba`。

#### 2. select 指令



如果需要选择不同的设备，不需要重新启动 `parted`。只需在 `parted` 交互模式下使用 `select` 命令即可。例如要查看 `/dev/sdb`，只需输入下面命令，然后就可以查看或配置 `/dev/sdb` 的分区表：

```
(parted)select /dev/sdb
使用 /dev/sdb
```

### 3. rm 指令

如果要在 `parted` 下删除分区 `/dev/sdb2`，首先需要查看当前的分区表来判定要删除的分区码：

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sda: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
```

Number	Start	End	Size	Type	File system	标志
1	32.3kB	107MB	107MB	主分区	ext3	启动
2	107MB	8587MB	8480MB	主分区		lvm

使用 `rm` 指令来删除分区号为 2 的磁盘分区：

```
(parted) rm 2
```

与 `fdisk` 不同，`parted` 不需再执行写入操作，只需按 `Enter` 键，改变就会立即发生。因此在执行前请仔细检查。删除了分区后，可以使用 `print` 命令来确认分区在分区表中是否已被删除：

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sda: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
```

Number	Start	End	Size	Type	File system	标志
1	32.3kB	107MB	107MB	主分区	ext3	启动

也可以使用下面的命令来确认是否内核已经知道该分区已被删除：

```
# cat /proc/partitions
major minor #blocks name
8 0 8388608 sda
```

退出 `parted` 程序后，如果需要，还应对 `/etc/fstab` 文件进行更新，删除已删除分区的对应行。

注意：正在被使用的设备上的分区是无法删除的。

### 4. resize 指令

可以使用 `resize` 指令重新划分分区大小。分区类型必须是下列之一：`FS-TYPEs`、`ext3`、`ext2`、`fat32`、`fat16`、`hfsx`、`hfs+`、`hfs` 或 `linux-swaps`。要重划大小的分区上已用的空间必须大于新建的大小。使用 `resize` 指令前可以使用 `print` 指令查看分区号，然后在 `resize` 指令后跟随分区号以及以 `MB` 为单位的起始点和终止点。例如：

```
(parted) resize 2 107 2048
```

分区被重新划分了大小后，使用 `print` 命令来确认分区已被正确地重新划分了大小，并且具备正确的分区类型和文件系统类型。也可以在重新引导系统后，使用 `df` 命令来查看分区是否已被挂载，以及分区的大小、类型。

注意：正在被使用的设备上不允许使用 `parted` 命令重新划分分区的大小

### 5.1.8 使用 mkfs 创建文件系统

磁盘建立分区后，需要为每一个分区创建文件系统，使用命令 `mkfs` 可以完成该任务。`mkfs` 命令格式如下所示：

```
mkfs [-V] [-t fstype] [fs-options] filesys [blocks]
```

- ❑ `-V`：以冗余模式进行输出,该选项通常仅用于测试
- ❑ `-t`：定义创建的文件系统类型，默认为 `ext2`
- ❑ `-c`：在创建文件系统前检查设备坏块。
- ❑ `-l`：从指定的文件读入坏块信息
- ❑ `-v`：以冗余模式进行输出

例如在硬盘 `/dev/hda3` 上创建 `ext3` 文件系统，命令行为：

```
#mkfs -t ext3 /dev/hda3
```

例如使用 `mkfs` 命令在软盘上创建 `ext3` 文件系统，命令行为：

```
#mkfs -t ext3 /dev/fd0
```

在分区上创建文件系统之后，还可以使用命令 `tune2fs` 对文件系统设置进行调整。`tune2fs` 命令可以修改文件系统的卷标、转换文件系统的类型。例如将硬盘 `/dev/hdb2` 上的 `ext2` 文件系统转换为 `ext3` 文件系统，命令行如下：

```
#tune2fs -j /dev/hdb2
```

### 5.1.9 使用 fdformat 格式化磁盘

命令 `fdformat` 可以实现对软盘的格式化，该格式化属于低级格式化。`fdformat` 命令格式如下所示：

```
fdformat [-n] 设备名
```

其中选项 “`-n`” 表示格式化完成后，不进行验证。软盘设备可以是下列之一，其中主设备号均为 2，从设备号如括号中所示：

```
/dev/fd0      (minor=0)
/dev/fd0d360  (minor = 4)
/dev/fd0h1200 (minor = 8)
/dev/fd0D360  (minor = 12)
/dev/fd0H360  (minor = 12)
/dev/fd0D720  (minor = 16)
/dev/fd0H720  (minor = 16)
/dev/fd0h360  (minor = 20)
/dev/fd0h720  (minor = 24)
/dev/fd0H1440 (minor = 28)
/dev/fd1d360  (minor = 5)
/dev/fd1h1200 (minor = 9)
/dev/fd1D360  (minor = 13)
/dev/fd1H360  (minor = 13)
/dev/fd1D720  (minor = 17)
/dev/fd1H720  (minor = 17)
/dev/fd1h360  (minor = 21)
```

除了上面定义的软盘设备文件外，用户还可以根据需要使用 `mknod` 命令自行创建。

### 5.1.10 使用 e2fsck 修复磁盘

e2fsck 命令是 Linux 操作系统中非常有用的工具，其主要功能是检查文件系统的正确性，对受损的文件系统进行修复。由于系统长期运行，很可能会因为网络攻击或非正常关机而导致文件系统受损，造成诸如文件系统的 i 节点表和磁盘的内容不一致等问题。e2fsck 是专门为 Linux 操作系统的 ext2 文件系统设计的修复工具，其命令格式如下：

```
e2fsck [-pacnyrdfkvstDFSV] [-b superblock] [-B blocksize] [-l|-L bad_blocks_file] [-C fd] [-j external-journal]
[-E extended_options] 设备名
```

其主要选项及其说明如表 5-6 所示。

表 5-6 e2fsck 命令选项及说明

选项	说明
-a	不询问便自动进行修复，与 -p 选项作用相同。保留该选项仅为了向后兼容，建议尽可能使用 -p 选项。
-b	使用指定的超级块（superblock），而不使用预设的超级块。
-B	指定搜索的超级块的大小。
-c	调用 badblocks 程序对磁盘做只读性扫描，查找存在的坏块。如果该选项被定义了两篇，则使用非破坏性的读/写方法对磁盘坏块进行扫描。
-C	将扫描过程信息记录在指定的文件中，使整个检测过程受到监控。
-d	显示调试信息。
-D	优化文件系统目录。如果文件系统支持目录索引，则重新建立索引；对于传统的线性目录或较小的目录，使用排序和压缩技术。
-E	设置 e2fsck 扩展选项，选项之间用逗号分隔。
-f	即使文件系统很简单也强制检测。
-F	在开始检测前将文件系统的设备缓冲区清空。
-j	设置外部日志的路径。
-k	与 -c 选项联合使用，已在坏块列表中列出的坏块不做修改，通过使用 badblocks 程序检测出的新坏块将被添加到坏块列表中。
-l	将文件中指定的区块添加到坏块列表中。
-L	与 -l 命令相似，只是在将指定区块添加到坏块列表前，清空坏块列表。
-n	以只读方式打开文件系统，并且假定所有交互的答案均为“no”。
-p	不进行询问，自动修复文件系统。
-r	该选项并无实际意义，只是为了向前兼容。
-s	如果文件系统已经是标准的字节顺序，则交换字节顺序；如果文件系统已是标准的字节顺序，则不进行任何操作。
-S	忽略当前的字节顺序，直接进行交换。
-t	显示 e2fsck 的时间统计。
-v	冗余模式。
-V	显示版本信息。
-y	在交互模式下，假定所有的答案均为“yes”。

在使用 e2fsck 命令修复文件系统时，必须将要被修复的文件系统卸载，否则是极不安全的，甚至会造成文件系统的崩溃。因此在安装 Linux 文件系统时，建议至少创建两个以上的主文件系统，这样在一个文件系统受损时，可以通过另一个文件系统对其进行修复。例如，假定计算机上安装了两个主文件系统：hda1 和 hda2，当 hda1 文件系统出现故障，可以使用 hda2 引导系统，然后卸载 hda1 并利用 e2fsck 修复，命令行如下：

```
#umount /dev/hda1
#e2fsck -av /dev/hda2
```

修复完成后，重新启动计算机，即可使用 hda1 的文件系统重新引导系统。

## 5.2 磁盘配额管理

Linux 是一个多用户、多任务操作系统，支持多客户端、多用户的使用。经验表明，在多用户系统上，加入的磁盘空间越多，用户使用的也越多，磁盘空间的浪费也越多，同时系统的可靠性也会大幅降低。保证系统有足够磁盘空间的最好方法就是有效地限制用户的使用量。在 Linux 中可以使用磁盘配额技术。磁盘配额可以为每一个用户或者用户组，甚至一个文件系统设定磁盘使用额度。通过磁盘配额，当用户使用的磁盘空间过多或分区占用过大时，系统管理员都会得到警告，从而及时采取措施。

在 Red Hat Enterprise Linux 5 中，系统自带了磁盘配额软件，默认情况下系统会自动安装，因此可以直接使用磁盘配额系统。

### 5.2.1 磁盘配额的系统配置

磁盘配额的系统配置主要包括添加参数、重载分区和生成配额文件 3 个步骤：

#### 1. 添加 “usrquota” 和 “grpquota” 参数

首先需要确定要使用磁盘配额的分区。例如，/home 分区由于其目录下包含了所有用户的主目录，且用户利用 FTP 登录主机时也通常使用该目录作为起始目录，因此最好利用磁盘配额对其空间的使用进行限制。

在确定需要使用磁盘配额的分区后，需要在/etc/fstab 文件中对分区进行标注，打开/etc/fstab 文件，在/home 分区记录项添加 “usrquota” 和 “grpquota” 参数，如下所示：

/dev/VolGroup00/LogVol00 /	ext3	defaults	1 1
LABEL=/boot /boot	ext3	defaults, 1 2	
LABEL=/home /home	ext3	defaults,usrquota,grpquota	1 2
devpts /dev/pts	devpts	gid=5,mode=620	0 0
tmpfs /dev/shm	tmpfs	defaults	0 0
proc /proc	proc	defaults	0 0
sysfs /sys	sysfs	defaults	0 0
/dev/VolGroup00/LogVol01 swap	swap	defaults	0 0

其中参数 “usrquota” 表示要在该文件系统中限制用户的使用空间，而参数 “grpquota” 表示要在该文件系统中限制用户组的使用空间。

#### 2. 重新挂载配额分区

为使修改生效需要重新启动计算机，也可以使用 mount 命令对该分区进行重载，命令行如下：

```
#mount -o remount /home
```

重载过程是先卸载分区，然后再重新挂载，由于根分区无法卸载，不能使用该命令。

#### 3. 生成 aquota.group 和 aquota.user 磁盘配额文件

为了使系统能够按着磁盘配额进行工作，必须建立磁盘配额文件 aquota.group 和 aquota.user。使用 quotacheck 命令可以完成配额文件的自动创建。quotacheck 命令还具有检查文件系统、建立硬盘使用率列表，以及检查每个文件系统的空间限额等功能。quotacheck 命令参数如表 5-7 所示。

表 5-7 quotacheck 主要参数及说明

选项	说明
-a	扫描在/etc/mtab文件中所有挂载的非NFS文件系统。
-d	启用调试模式，会产生大量的调试信息，有详细的输出结果但扫描速度比较慢。

-u	计算每个用户占用的目录和文件数目，并创建aquota.user文件。
-g	计算每个用户组占用的目录和文件数目，并创建aquota.group文件。
-c	忽略现存的配额文件，重新扫描并建立新配额文件。
-v	显示命令执行过程。
-b	强制quotacheck在建立新配额文件前对旧配额文件进行备份。

文件 /etc/mtab 与/etc/fstab 内容类似，/etc/fstab 表示系统开机时默认加载的分区，而/etc/mtab 表示目前系统加载中的分区。例如扫描/etc/mtab 文件，建立 aquota.group 和 aquota.user 磁盘配额文件，命令行为：

```
quotacheck -avgu
```

执行 quotacheck 命令后，可以看到在/home 文件系统中新建了 aquota.group 和 aquota.user 文件：

```
#ls -al
drwxr-xr-x 13 root    root    4096 08-31 07:59 .
drwxr-xr-x 24 root    root    4096 08-31 13:42 ..
-rw-----  1 root    root    6144 08-31 13:38 aquota.group
-rw-----  1 root    root    6144 08-31 13:38 aquota.user
drwx-----  2 stud1   users   4096 2007-07-14 stud1
drwx-----  2 stud2   users   4096 2007-07-14 stud2
... ..
```

本例中，在成功建立 aquota.group 和 aquota.user 文件后，就可以在/home 分区下对用户和用户组进行磁盘使用空间的限制。直接运行命令“quota 用户名”，可以显示分给指定用户的磁盘配额：

```
# quota student
Disk quotas for user tanli (uid 605): none
```

注意：对于有些版本的 Linux，在对/etc/fstab 文件设置配额参数后，需要修改/etc/rc.d/rc.local 脚本文件，才能使系统每次启动时能够自动创建配额文件并自动启用磁盘配额程序。例如，修改/etc/rc.d/rc.local 脚本文件如下：

```
... ..
#Check quota and then turn quota on
if [ -x /sbin/quotacheck ]; then
echo "Checking quotas. This may take some time..."
/sbin/quotacheck -avug
echo "Done"
fi
if [ -x /sbin/quotaon ]; then
echo "Enabling disk quota.."
/sbin/quotaon -avug
echo "Done"
fi
... ..
```

其中命令“/sbin/quotacheck -avug”用于扫描/etc/fstab 中设置磁盘配额的分区，以及在该分区下建立配额文件 aquota.group 和 aquota.user。需要注意的是：对于 Red Hat Enterprise Linux 不需要这个步骤，在 Red Hat Enterprise Linux 的/etc/rc.d/rc.sysinit 文件中已经设置启用 quota。

## 5.2.2 对用户设置磁盘配额

在上节中创建了用户和用户组磁盘配额文件 aquota.group 和 aquota.user。由于 aquota.group 和 aquota.user 文件结构复杂，无法直接在编辑器中打开，必须通过 edquota 命令才能正常编辑。edquota 命令常用参数如表 5-8 所示。

表 5-8 edquota命令参数及说明

选项	说明
-g	设置用户组的磁盘配额
-u	设置用户的磁盘配额，默认值
-p	对磁盘配额设置进行复制
-t	对文件系统设置软时间限制

edquota 命令进入的是 vi 编辑状态（vi 的使用方法参见第 9 章第 4 节文本编辑器），可以编辑用户和用户组的磁盘空间限制。图 5.13 列出了 edquota 命令的编辑格式。

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda1	20	0	0	9	0	0
设置 quotat 分区	已有磁盘空间	磁盘空间配额		已用 i 节点数	i 节点配额	

图 5.13 edquota 命令编辑格式

其中，“Filesystem”字段指出目前使用磁盘配额的文件系统为/dev/sda1，“block”字段表示用户已经使用的空间为 20 个 blocks，其中 1 个 block 是 1024 字节。“block”字段后的“soft”和“hard”字段，表示磁盘空间配额的软限额和硬限额，值为“0”表示没有限制。“inodes”表示已使用的 i 节点数。i 节点数就是文件和目录数，其后的“soft”和“hard”字段分别表示 i 节点数的软限额和硬限额，值为“0”表示没有限制。

所谓软限额是指用户达到此限制时，系统会发出警告信息，但是用户仍然可以继续使用，不过一旦达到硬限额时，用户就无法再写入了。

例如为用户 student 设置磁盘限额，其中磁盘空间软限额为 10MB，硬限额是 50MB，i 节点的软限额为 600，硬限额为 800，使用 edquota 命令编辑 student 用户配额如下：

```
#edquota -u student
Disk quotas for user student (uid 605):
Filesystem    blocks    soft    hard    inodes    soft    hard
/dev/sda1      20       10000   50000    8         600     800
```

其中 blocks 为 20 表示用户 student 已经使用了 20 块（共  $20 \times 1\text{KB} = 20\text{KB}$ ）的磁盘空间。当用户使用的磁盘空间达到软限额 10MB 时，系统会提示用户已达到软限额，但用户仍可以断续写入。当达到硬限额 50MB 时，系统将禁止 student 向磁盘写入任何数据。inode 为 8 表示用户 student 已经建立了 8 个文件，当用户建立的文件数达到 600 时，系统会给出警告提示用户已达到软限额，当达到 800 将不再允许建立文件或目录。

例如设置用户 teacher 使用的磁盘软限额为 500MB，硬限额为 800MB，规定建立文件/目录的软限额为 128，硬限额为 512，使用 edquota 命令如下：

```
#edquota -u teacher
Disk quotas for user teacher (uid 601):
Filesystem    blocks    soft    hard    inodes    soft    hard
/dev/sda1      0        500000  800000    0         128     512
```

如果有很多用户需要设置相同的配额，可以使用 edquota 命令的“-p”选项。例如建立用户 teacher1、teacher2、teacher3 与上例中用户 teacher 一样的磁盘配额方案，命令行为：

```
#edquota -p teacher -u teacher1 teacher2 teacher3
```

除了可以限制用户使用磁盘空间大小和 i 节点数目外，还可以设置“grace period”，即宽限期。所谓宽限期是指当用户使用的磁盘空间或 i 节点数达到软限额，但还未达到硬限额情况下，允许用户继续使用的天数。一旦达到硬限额或超出设置的宽限期，用户将无法继续使用，直到将磁盘的使用量或 i

节点数降到软限额以下。使用“edquota -t”命令可以设置宽限期：

```
#edquota -t
Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
Filesystem      Block grace period   Inode grace period
/dev/sda1        7days                7days
```

系统默认的宽限期为 7 天，可以按日、小时、分钟和秒为单位进行修改。在此设置的宽限期对所有用户和用户组有效。

### 5.2.3 对用户组设置磁盘配额

在多用户操作系统中，每个用户都有自己所属的用户组，一个用户组中可以包含若干用户，因此可以针对用户组设置磁盘配额。在设置用户组磁盘配额时，应按着组内用户数及每一用户的磁盘配额计算用户组配额，使用户组配额限制约等于组内各个用户磁盘配额的总和，否则会出现“不足”或“过剩”现象。

例如，设用户组 teacher 包含 Mike 和 Jerry 两个用户。假设 Mike 和 Jerry 各自的磁盘配额上限为 50MB，而用户组 teacher 的磁盘配额上限为 80M，那么当 Mike 使用 50MB 磁盘空间后，Jerry 只能使用  $80\text{MB} - 50\text{MB} = 30\text{MB}$  磁盘空间，而不再是原先设置的 50MB。

例如，设用户组 student 包含 student1~student20 共 20 个用户。设每个 student 用户使用的磁盘软限额为 100MB，硬限额为 150MB，则用户组的软限额可以为  $100\text{MB} \times 20 = 2\text{G}$ ，而硬限额可以为  $150\text{MB} \times 20 = 3\text{G}$ 。由于组内用户较多，同时达到硬限额的可能性很低，为了可省磁盘空间可以尝试将用户组的磁盘硬限额设为 2.5G。

例如，设用户组 staff 包含两个用户 Jack 和 Jane。假设 jack 和 Jane 的各自的磁盘配额上限为 20MB，如果用户组 staff 的磁盘配额上限设为 50 MB，即用户组磁盘配额上限大于用户磁盘配额上限的总和，则当 Jack 和 Jane 均达到各自 20MB 的磁盘配额上限时，即使用户组的磁盘配额上限还未达到，也不能继续使用磁盘。由此很可能造成磁盘空间浪费。

使用“edquota -g”命令可以为用户组设定磁盘配额。例如建立用户组 test 的磁盘配额，设置磁盘空间使用的软限额为 4G，硬限额为 8G，对 i 节点不设限制，配置过程如下：

(1) 编辑/etc/fstab 文件，添加 grpquota 参数，如下所示

/dev/VolGroup00/LogVol00 /	ext3	defaults	1 1
LABEL=/boot	/boot	ext3	defaults, 1 2
LABEL=/home	/home	ext3	defaults,usrquota,grpquota 1 2
devpts	/dev/pts	devpts	gid=5,mode=620 0 0
tmpfs	/dev/shm	tmpfs	defaults 0 0
proc	/proc	proc	defaults 0 0
sysfs	/sys	sysfs	defaults 0 0
/dev/VolGroup00/LogVol01 swap	swap	defaults	0 0

(2) 重新挂载配额分区。分区重新可以通过重新启动计算机，也可以使用 mount 命令对该分区进行重载，命令行如下：

```
#mount -o remount /home
```

(3) 生成 aquota.group 磁盘配额文件，命令行如下：

```
quotacheck -avg
```

(4) 使用 edquota 命令为用户组 test 建立磁盘配额。进入用户组 test 的磁盘配额编辑状态，命令行如下：

```
#edquota -g test
```

```
Disk quotas for group test (gid 106):
```

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda1	3242	0	0	34	0	0

对其中的磁盘空间“soft”和“hard”字段以及i节点的“soft”和“hard”字段进行修改，如下：

```
#edquota -g test
```

```
Disk quotas for group test(gid 106):
```

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda1	3242	4000000	8000000	0	34	0

此外，edquota 使用“-gp”参数可以对磁盘配额进行复制。例如为用户组 review 建立与用户组 test 相同的磁盘配额，命令行如下：

```
#edquota -gp test review
```

#### 5.2.4 启动和终止磁盘配额

在设置好磁盘配额之后，用户可以使用 quotaon 和 quotaoff 命令启动和终止磁盘空间配额的限制。例如，启动/home 磁盘空间配额，命令行如下：

```
# quotaon /home
```

```
/dev/sda1 [/home]: group quotas turned on
```

```
/dev/sda1 [/home]: user quotas turned on
```

关闭/home 磁盘空间配额可以使用命令：

```
# quotaoff /home
```

```
/dev/sda1 [/home]: group quotas turned off
```

```
/dev/sda1 [/home]: user quotas turned off
```

也可以不指定操作的分区，使用“-aguv”参数设定自动搜索，命令如下：

```
# quotaon -aguv
```

```
/dev/sda1 [/home]: group quotas turned on
```

```
/dev/sda1 [/home]: user quotas turned on
```

```
/dev/sda1 [/www]: group quotas turned on
```

```
/dev/sda1 [/www]: user quotas turned on
```

```
# quotaoff -aguv
```

```
/dev/sda1 [/home]: group quotas turned off
```

```
/dev/sda1 [/home]: user quotas turned off
```

```
/dev/sda1 [/www]: group quotas turned off
```

```
/dev/sda1 [/www]: user quotas turned off
```

#### 5.2.5 使用 quota 命令监视磁盘使用

在设定磁盘配额后，系统管理员需要查看某个用户正在使用的磁盘空间情况，可以直接运行 quota 命令，命令格式如下：

```
quota -u 用户名
```

例如查看用户 kitty 使用的磁盘空间情况，命令行如下：

```
#quota -u kitty
```

```
Disk quotas for userkitty (uid 608):
```

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda1	10	0	0	3	0	0



可以看到用户 kitty 使用了 10K 磁盘空间，建立了 3 个文件或目录。同样也可以使用参数“-g”查看用户组使用磁盘空间的情况，命令格式如下：

```
quota -g 用户组名
```

如果系统没有设置磁盘配额，则无法使用 quota 命令来查看磁盘空间的使用情况，系统会提示：

```
#quota -u kitty
```

```
Disk quotas for user kitty (uid 608):none
```

## 5.2.6 使用 du 命令进行磁盘空间统计

除了使用 quota 命令监视磁盘使用情况外，用户还可以尝试使用其他的方法，例如使用命令 du。命令 du 可以报告文件和目录占用的磁盘空间，du 命令格式如下：

```
du [选项] [文件或目录名]
```

如果没有指定文件或目录名，则默认对当前目录进行统计。命令 du 主要命令选项如表 5-9 所示。

表 5-9 du命令选项说明

选项	说明
-s	仅显示文件或目录占用的块数，默认1块等于1024字节。
-a	显示所有文件及其子目录占用的数据块数。
-c	在显示结果最后添加一个总计（系统默认设置）。
-k	以1024字节即1K为单位进行统计（系统默认设置）。
-b	以字节为单位进行统计。
-l	对于硬链接文件，系统会计算多次。
-x	对不同文件系统上的同名目录不进行统计。
-m	以1024K字节即1M为单位进行统计。
-h	以人们习惯的方式显示输出结果。

例如，使用选项“-a”对用户 kitty 主目录进行统计（包括其中的子目录），命令行如下：

```
# du -a ~kitty
4      /home/stud1/test.2
4      /home/stud1/test.4
4      /home/stud1/test.3
4      /home/stud1/pic/hen
4      /home/stud1/pic/cock
16     /home/stud1/pic
4      /home/stud1/homework.1
4      /home/stud1/test.1
44     /home/stud1
```

例如，使用选项“-s”对指定的 kitty 用户进行统计，命令行如下：

```
# du -s ~kitty
44     /home/stud1
```

如果以字节为单位进行统计，命令行如下：

```
# du -b ~kitty
4096   /home/stud1/pic
8192   /home/stud1
```

如果以 1024 字节即 1K 为单位进行统计，命令行如下：

```
# du -k ~kitty
16     /home/stud1/pic
44     /home/stud1
```

如果以 1024K 字节即 1M 为单位进行统计，命令行如下：

```
# du -m ~ kitty
1      /home/stud1/pic
1      /home/stud1
```

如果希望给显示结果加上总计，应使用选项“-c”为，命令行如下：

```
# du -c ~ kitty
16     /home/stud1/pic
44     /home/stud1
44     总计
```

为了使输出结果更加友好（按千字节、兆字节或千兆字节），可以使用“-h”选项，例如：

```
# du -h ~kitty
16k    ./pic
76k    .
```

命令 du 配合命令 sort，可以按占用空间的大小顺序对磁盘空间的使用情况进行统计。例如使用下面的命令对当前目录进行统计，并将统计结果保存到/home/disk\_used 文件中，命令行为：

```
# du -a | sort -n > /home/disk_used &
```

其中“du -a”用于统计当前目录下所有文件和目录占用的磁盘空间，“sort -n”命令对“du -a”的统计结果按从小到大的顺序进行排序，然后存储到/home/disk\_used 文件中。由于这种对磁盘空间使用情况的统计一般会占用相当长的时间，使用“&”将该命令移到后台运行。命令执行完毕，生成的/home/disk\_used 文件内容如下：

```
# cat /home/disk_used
4      ./pic/cock
4      ./pic/hen
4      ./test.3
8      ./homework.1
8      ./test.2
8      ./test.4
16     ./pic
24     ./test.1
76     .
```

## 5.3 磁盘管理常见问题

由于磁盘应用越来越广泛，面向磁盘的各种技术不断创新，作为系统管理员，需要诊断和排除的与磁盘相关的各种问题也越来越复杂。系统管理员不仅需要掌握大量的工具，还需要熟悉各种解决问题的方法。

### 5.3.1 添加新磁盘

由于硬盘空间耗尽而需要添加新硬盘是系统管理员经常遇到的问题。在 Linux 中添加一块新硬盘通常涉及以下几个步骤：

#### （1）硬盘的安装

首先关闭计算机，然后根据硬盘的类型(SATA、EIDE 或 SCSI)，按照硬盘供应商的推荐值，对硬盘以及 BIOS 进行设置，将硬盘安装到系统中。

重新启动计算机，并进一步确认新硬盘是否已被 Linux 系统识别。可以通过使用 `dmesg` 命令进行核实。例如，假设新硬盘是安装到计算机第 2 个 IDE 接口的第 1 块硬盘，则新硬盘的设备名应为 “hdc”，以 root 身份在 shell 下运行下面命令：

```
#dmesg |grep hdc
```

如果输出结果中有 hdc 相关内容的描述，则说明新硬盘已被 Linux 系统识别，否则还需做进一步检查。

### （2）在新硬盘上创建分区

使用 `fdisk` 或 `parted` 命令可以在一块硬盘上创建一个或多个 Linux 分区，并按需要为其分配不同的文件系统类型和不同的挂载点。最简单的创建分区的方法是把整个硬盘作为一个单独的 Linux 分区。例如在上例中使用 `fdisk` 命令，对 hdc 创建 Linux 分区：

```
#fdisk /dev/hdc
```

按第 1 节所讲的方法，使用 `fdisk` 将整个硬盘划分为一个分区/dev/hdc1。

### （3）分区格式化

在 Linux 中的格式化命令是 `mkfs`。例如将/dev/hdc1 格式化为 ext3 文件系统，命令行为：

```
#mkfs -t ext3 /dev/hdc1
```

### （4）挂载文件系统

文件系统创建后，需要建立分区挂载点。例如将/dev/hdc1 挂载到/mnt/newdisk 目录下，需要在/mnt 目录下创建空目录 newdisk，然后使用 `mount` 命令进行挂载，命令行为：

```
#mkdir /mnt/newdisk
```

```
#mount /dev/hdc1 /mnt/newdisk
```

如果希望每次系统启动时自动挂载该硬盘上的文件系统，需要编辑/etc/fstab 文件，添加类似下面的一行：

```
/dev/hdc1 /mnt/newdisk ext3 default 1 1
```

## 5.3.2 修复受损文件系统

非正常关机经常会导致文件系统出现错误。在 Windows 下通常使用 `chkdisk` 命令进行修复，在 Linux 下可以使用 `fsck` 命令（其中 `e2fsck` 是专门针对 ext2 文件系统的修复命令）。文件系统受损后，经常提示：

```
... ..  
Giver root password for maintenance  
(or type Control-D for nomal setup):
```

在提示符下输入 root 密码，然后按 Enter 键进入修复模式，使用 `fsck` 尝试对文件系统进行修复，命令行为：

```
#fsck /dev/hda1  
... ..
```

如果系统最终提示 “passed”，说明检查和修复完成，重新启动后即可正常进入系统。

## 5.3.3 无法引导文件系统

如果系统出现较为严重的错误，导致无法引导文件系统，可以尝试从引导盘启动系统，然后对受损的文件系统进行修复。如果没有定制引导盘，也可以使用 Red Hat Enterprise Linux 5 安装光盘套装中的第 1 张光盘，通过 Linux Rescue 模式，启动 Linux 系统进行修复。

(1) 将 Red Hat Enterprise Linux 5 第 1 张安装盘放入光盘驱动器，然后设置系统从光盘启动。从光盘引导系统后，出现如图 5.14 所示界面。

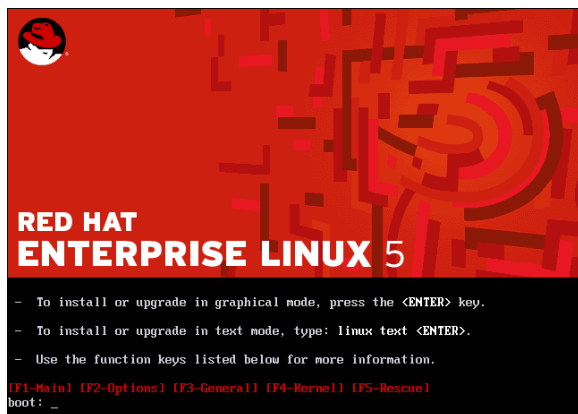


图 5.14 从光盘引导 Linux 文件系统

(2) 按 F5 键进入救援模式 (Linux Rescue)，如图 5.15 所示。在提示符下键入 “linux rescue”，按 Enter 键，启动救援模式。

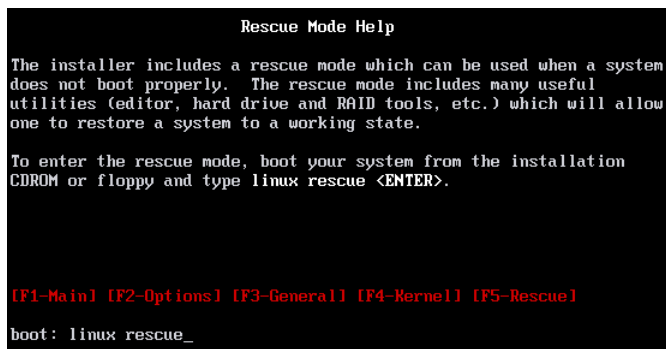


图 5.15 键入 “linux rescue” 进入救援模式

(3) 选择语言、键盘以及是否启用网络后，弹出如图 5.16 所示的【Rescue】选择对话框。



图 5.16 Linux 救援模式选项

其中包括三个选项：

- ☐ Continue 选项：如果选择了 Continue（继续）选项，Red Hat Enterprise Linux 将搜索已安装的硬盘，所有查找到的文件系统均被安装在/mnt/sysimage 目录中。

❑ **Read-Only 选项：**Read-Only（只读）选项与 Continue 选项几乎完全相同，差别仅在于 Read-Only 选项会将查找到的文件系统以只读模式加以挂载。

❑ **Skip 选项：**Skip（跳过）选项直接进入一个命令 Shell。

（4）单击 **【Continue】** 选项，弹出 **【Rescue】** 对话框，如图 5.17 所示。



图 5.17 从 **【Continue】** 选项进入救援模式

从 **【Rescue】** 对话框可以看到下列提示信息：

❑ 硬盘的文件系统被挂载到了 /mnt/sysimage 目录下。如果用户仍然希望使用自己以前系统的根环境，可以在进入 Shell 后使用命令“chroot /mnt/sysimage”。

❑ 使用命令 exit 可以从 Shell 中退出，退出后系统自动重启。

（5）在图 5.17 中，选择 **【OK】**，系统会进入一个命令 Shell，如下所示：

```
Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.
sh-3.1#
```

使用 e2label 命令可以进一步查看分区被重新挂载的位置，例如查看 /sda1 的挂载点，命令行如下所示：

```
#e2label /dev/sda1
/boot
```

可以看到 /dev/sda1 被挂载到了 /boot 目录下。

进入 /mnt/sysimage 目录，查看原来的文件系统：

```
sh-3.1# cd /mnt/sysimage
sh-3.1# ls
dev lib mnt proc selinux sys
var bin exit lost+found opt root share tmp
boot home media password sbin srv usr
```

如果有一个或多个文件系统没有得到安装，可以用 VI 编辑器打开 fstab 配置文件，检查其中的设置和格式本身的完整性，如下所示：

```
sh-3.1# vi /mnt/sysimage/etc/fstab
```

可以使用 fsck 命令修复一个受损的文件系统，例如修复 /dev/sda2，使用下面的命令：

```
sh-3.1# fsck /dev/sda2
```

如果在 Linux Rescue 模式下需要访问 Linux 手册页面，由于顶级根目录已不是原来文件系统分级结构的最顶层，将无法激活 Linux 手册页面。可以运行 chroot 命令，更改顶级根目录，命令行如下：

```
sh-3.1# chroot /mnt/sysimage
sh-3.1# cd /
sh-3.1# ls
```

```
dev lib mnt proc selinux sys
var bin exit lost+found opt root share tmp
boot home media password sbin srv usr
sh-3.1# man
What manual page do you want?
```

(6) 如果在图 5.16 中选择了【Read-Only】选项，系统也将进入救援模式，系统提示：

```
When finished please exit from the shell and your system will reboot.
sh-3.1#
```

【Read-Only】选项表明原文件系统均以只读方式安装在系统中。如果用户有多个文件系统，比较安全的作法应该是以只读方式进行检查，在需要写入操作时，再使用 `mount` 命令重新挂载需要写入操作的文件系统。例如重新挂载 `/dev/sda2` 分区，使其能够执行写入操作：

```
sh-3.1# mount -w -o remount /dev/sda2
```

(7) 如果在图 5.16 中选择了【Skip】选项，系统会直接进入一个 Shell。该选项是将一个最小的根系统和内核装入到一个 RAM 盘中，不安装任何文件系统，仅提供给用户一个 Shell。用户需要完全依靠手动挂载、修复文件系统，需要用户对 `mount`、`mkdir`、`cp`、`move`、`fdisk`、`fsck` 等命令十分熟悉。