
第一章 引 论

1.1 本指南的目的

《AT&T UNIX[®] System V Release 4.0 Version 1.0 系统管理指南》用来帮助管理运行 UNIX System V Release 4.0 的 386 系统,本指南假定读者熟悉其它的 UNIX 系统。系统管理的任务是通过 shell 命令和 **sysadm** 菜单显示出来的,这些菜单在操作、管理和维护 (OA&M) 软件包中。OA&M 不是基本系统的一部分,必须安装它才能使用该界面。

本指南应在依照《AT&T UNIX System V Release 4.0 Version 1.0 安装指南》中的指令安装了软件后才使用。安装和初始化外加应用软件的指令包含在附属于外加产品的文档中。

1.1.1 什么是 386 上的系统管理?

386 上的系统管理包括在计算机系统上所普遍执行的任务:文件的后援和回存、增添和删除用户、对网络的监控、增添和删除硬件设备等等。

386 上提供了两种界面来完成这些任务:shell 命令和 **sysadm** 菜单,前者是针对 UNIX 环境下的管理功能的直接用户界面,而后者是通过 **sysadm** 命令提供一个完成相同任务的菜单界面。有经验的系统管理员应熟悉这两种界面。

1.1.2 操作、管理和维护 (OA&M)

如果安装了操作、管理和维护 (OA&M) 软件包,那么 **sysadm** 便可使用表格、菜单和语言界面 (FMLI) 以窗口的形式来调用菜单。

注意: 如果打算 FACE 和 OA&M 都安装,那么建议在安装框架存取命令环境 (FACE) 之前安装 OA&M。

假如先安装 FACE,那么在装入第一片软盘时,会得到出错消息,其安装过程将以 OA&M 软件包部分出错而告终。

当安装 OA&M 时,将给你一个基本的或是扩充的 OA&M 的选择,其默认的选择是扩充的 OA&M,它将给你一个较周全的后援和回存服务 (详见第三章),加上一些其它的功能,如用户组、软件管理、文件管理等。

1.1.2.1 安装基本的和扩充的 OA&M

在初始的安装过程中,可以安装基本的 OA&M 或者基本的和扩充的合在一起的

OA&M。如果已经安装了基本的和扩充的 OA&M, 并且想删除外加的软件包以节省空间, 那么必须删除基本的和扩充的 OA&M, 并重新装入基本的 OA&M。在 OA&M 菜单文件中的受影响的菜单行上以 #oam# 来标识菜单的扩充。

注意: 通过在“add-ons”目录结构中外加的软件包, 可以在“`.menu`”目标生成文件中加入 FML 源文件 (表格、菜单或正文文件)。当删除外加的软件包时 (模块化), 这些“加入”的文件是可被删除的 (遵循外加软件包开发者指南中所描述的规则)。如果外加软件包原型包括一些定义为 OAMmif 类的“`.mi`”文件, 则自动实现菜单的更新。OA&M 所提供的 OAMmif 类动作脚本能更新主菜单, 主菜单是由作为外加特定扩展的、外加软件包所提供的“`.mi`”文件声明的。这使得重新定向到外加目录特定的位置, 后继 FML 文件必须装在这个位置上。

例如, 小型计算机系统接口 (SCSI) 软件包加入已存在的“`buses`”和“`devices`”菜单选择项上, `buses.menu` 和 `device.menu` 中的 SCSI 特定选择项 (行) 在第 4 字段用 #SCSI# 定义。#SCSI# 由 `object_gen` 解释并翻译成 `/usr/sadm/sysadm/add-ons/scsi` (与 `/usr/sadm/sysadm/menu/*` 相反) —— SCSI 特定扩展的位置。以后其它的外加软件包可加在这些相同的“主”菜单项上 (并且将包含一个适当的 #add-on_name# 重定位)。

1.2 指南的有关说明

系统管理指南分成下述的十三章和三个附录。

第一章 引言。介绍系统管理指南的目的和内容, 说明本指南中所使用的记法约定并且列出了 UNIX SVR4.0 V1.0 家族中的其它资料。

第二章 使用文件系统。说明如何建立、安装和拆卸文件系统。

第三章 系统特性的设置。包含重新设置系统日期、时间和节点名的过程。

第四章 端口管理。描述如何在系统上增添、修改和删除端口及终端。

第五章 网络配置 (uucp) 的设置。描述如何为 uucp 建立网络文件和参数。

第六章 自动任务的调度。讨论如何设置参数而使得系统自动地执行一些任务。

第七章 使用打印机。说明增添和删除打印机、定义打印机类和过滤程序、配置打印机网络和指定队列优先级的过程。

第八章 增添和改变用户信息。说明增添、改变和删除用户组 and 用户注册名的过程。

第九章 管理存储设备。说明如何对存储设备及其上的内容进行格式化、复制、抹掉和删除, 它还描述了如何起设备的别名、如何管理设备组和设置第二个硬盘。

第十章 文件的后援和回存。包含基本和扩展系统上的文件的后援和回存过程。

第十一章 安全性。说明如何使用数据加密, 作为系统上的一种安全措施。

第十二章 硬盘管理。讨论硬盘的布局, 如何处理坏块及硬盘恢复的过程。

第十三章 可调参数。描述系统的可调参数以及如何修改它们来剪裁系统以满足特殊

的需要。

附录 A 记帐。包含记录系统各种功能的过程。

附录 B mtune 文件布局。列出了可调参数以及它们的默认、最小和最大值。

附录 C OA&M 菜单映象。列出 OA&M 菜单中所提供的菜单选择项和任务。

1.2.1 记法约定

下面是本指南中所使用的一组约定：

- 命令和命令任选项以黑体印刷。
- 文件名、目录名、程序和引用其它手册和章节的部分以斜体印刷。
- 菜单选项、屏幕提示和屏幕显示按等宽体字样印刷，在屏幕的布局图中只显示一个框架，仅使用部分屏幕并且不能精确地复制你的主控终端屏幕上所显示的东西。
- 键盘的引用以如下键的图形显示：**Enter**。两个键如 **Ctrl** **d** 表示你要同时按下控制键和一个指定的字符或其它键。

1.2.2 相关的资料

本资料是下列 UNIX SVR4.0 Version 1.0 系列资料中的一本：

《AT&T UNIX[®] System V Release 4.0 Version 1.0 产品简介》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 用户指南》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 FACE 用户指南》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 安装指南》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 用户参考手册》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 系统管理员指南》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 系统管理员参考手册》

《AT&T UNIX[®] System V Release 4.0 Version 1.0 集成软件开发指南》

第二章 使用文件系统

2.1 概述

本章描述如何创建、安装和拆卸 s5 及 ufs 的文件系统类型, 这些操作要求首先格式化软盘或硬盘。如果所使用的磁盘没有格式化, 则参见第九章“管理存储设备”中的“存储设备的格式化”节。

选择 UNIX System V Release 4.0 Version 1.0 系统管理菜单中的 `file_systems` 选项来执行这些任务。

2.2 创建文件系统

创建一个工作文件系统包括几个步骤, 一般来说, 这些步骤是:

1. 软盘格式化。
2. 用 OA&M 菜单或 `mkfs` 命令创建文件系统。
3. 安装文件系统。
4. 当不再需要存取文件系统时拆卸它。

2.2.1 使用 OA&M 菜单来创建文件系统

执行下列菜单来创建文件系统:

1. 从 `Manage File Systems` 菜单中选择 `make`, 系统显示 `Create a File System (make)` 屏幕:

```
3 Create a File System (make)
Device that will contain the file system: diskette1
File system type: s5
Label for the file system:
Once created, should the new file system be mounted? yes
File system name when mounted: /install
```

2. 输入文件系统的信息并按 **SAVE**, 系统显示下列屏幕之一:

4

Create An s5 File System (make)

Name of prototype file:
Block Size in Bytes: 1024
Number of Blocks:
Number of Files:

4

Create A File System (make)

Number of Blocks in the File System:
Block Size in Bytes: 8192
Fragment Size in bytes: 1024

- 3. 输入信息并按 **SAVE** ,系统显示往驱动器插入软盘的指示。
- 4. 在往驱动器里插入软盘之后按 **SAVE** ,系统创建文件系统并试图安装它。

2.2.2 使用 mkfs 创建文件系统

本节首先讨论 **mkfs** 命令的一般格式,然后讨论创建 s5 或 ufs 文件系统的特定应用。
mkfs 命令的格式是:

mkfs [-F *filetype*] [-V] [-m] [*current_options*] [-o *specific_options*] *special_operands*

其中的任选项是:

<i>filetype</i>	文件系统的类型,s5或ufs。
-V	回应完整的命令行,其中包括 <i>vfstab</i> 文件信息。
-m	返回用来创建已存在文件系统的命令行,这个任选项允许用户查看构造文件系统的属性。
<i>current_options</i>	s5支持的任选项。
<i>specific_options</i>	必须写到 <i>/etc/vfstab</i> 文件中的文件系统属性, <i>special</i> 是包含此特定文件系统属性的 <i>vfstab</i> 中的项名。
<i>specific_operands</i>	要创建的文件系统类型的特定操作对象。

2.2.2.1 创建 s5 文件系统

输入下列命令来创建一个新的 s5 文件系统或将一个老的文件系统转换为具有新的逻辑块大小的文件系统:

- 1. 如果要在已存在老的文件系统的磁盘分区上创建一个新的文件系统,则应对老系统做个后援(见第十章),你可使用 **cpio(1)**命令对一个或多个磁盘上的系统作后援。

2. 如果在老的文件系统上创建新的文件系统,可运行**labelit**命令,它报告已安装好的文件系统名和老文件系统的物理卷名。当建立新文件系统时,这些标号被破坏。使用**labelit**时,必须指定文件系统类型,例如,如果设备是f0q15d,文件系统是memo,卷名为memo2.0,将键入:

labelit -F s5 /dev/dsk/f0q15d memo memo2.0

3. 如果在一个老的文件系统上创建新的文件系统,并且新的文件系统有较大的逻辑块,那么,由于碎片的原因,为了存储数据,新的文件系统将要比老文件系统多分配一些磁盘块。使用**fsba (1M)**命令来找出具有新块大小的老文件系统对空间的需求。也可使用这些信息来确定用于新文件系统的磁盘扇区是足够大的。使用**prtvtoc (1M)**命令可找出当前磁盘扇区的大小。
4. 使用下列命令之一:

mkfs [-F s5] [-b blocksize] special blocks[: inodes] [gap blocks/cyl]

或者

mkfs [-F s5] [-b blocksize] special prototype [gap blocks/cyl]

其中的任选项:

<i>blocksize</i>	这是文件系统的逻辑块大小,默认值为1024个字节,s5也支持512个字节和2048字节的逻辑块。
<i>special</i>	<i>vfstab</i> 文件中的入口项,包含文件系统的属性。
<i>blocks</i>	文件系统将占有的块数,每块为512个字节, <i>inodes</i> 的默认值是每四个逻辑块就有一个i节点。
<i>gap blocks/cyl</i>	这些是记录间的间隔和每一柱面上的块数,这些数字依赖于块的大小和硬盘的大小。
<i>prototype</i>	一个文件名,它包括:文件系统所需求的块数,文件系统的目录和文件结构以及将已存在的文件内容读到文件系统中的指令。

注意上述的两种**mkfs**命令格式中并没有给出文件系统名称,文件系统是由它所在的特殊设备文件的文件名标识的,特殊设备文件通常放在/dev目录下,用来标识物理设备的控制器和设备号(分别为主次设备号)。

在第一种命令格式中,**mkfs**命令行上必须提供的其它信息是文件系统要占用的块数,每一块为512个字节。第二种格式允许你包含原型文件中的信息,也可定义新文件系统的目录和文件结构,甚至允许从已存在的文件系统中读入文件内容。

两种格式允许指定记录间的间隔和每个柱面上的块数等信息,如果命令行上未给出这些信息,就使用其默认值。

这里的建议依赖于文件系统的逻辑块的大小,**mkfs**的**-b**任选项允许你定义文件系统所使用的逻辑块大小,默认规定文件系统具有1024字节的逻辑块。通过**-b**任选项,可指定逻辑块大小为1024字节或2048字节,建议的值与命令所使用的默认

值不同,其最佳值依赖于应用的类型(例如,读操作多的应用与写操作多的应用有不同的特性)、所使用的磁盘控制器和磁盘的交叉存取。

在第一种**mkfs**格式中,即使要求文件系统中的块数,i节点数也可以省略。如果省略了i节点数,该命令使用其默认值,即每四个逻辑块一个i节点。

如果使用第一种**mkfs**格式,则建立的文件系统具有单个目录。如果使用了原型文件,如上所述,则它可包含一些信息,使得这条命令为文件系统建立并初始化目录和文件结构。

5. 运行 **labelit** 命令来回存文件系统名和卷名。
6. 填充新文件系统——例如,从后援文件系统中回存,或者如果你的系统有两个硬盘,则在已安装的文件系统上执行 **cpio(1M)**、**(volcopy (1M))**和 **dd(1M)**命令用于拷贝文件系统的映象,它们不能转换逻辑块大小。)

2.2.2.2 创建 ufs 文件系统

当 **mkfs** 命令用来创建 ufs 文件系统时,它建立一个带有 *root* 目录和 *lost+found* 目录的文件系统,其i节点数是作为文件系统大小的函数来计算的。

输入下述命令来创建一个新的 ufs 文件系统或将老的文件系统转换为具有新的逻辑块大小的文件系统:

1. 如果要在已有的老的文件系统的磁盘分区上创建一个新的文件系统,则对老的文件系统做个后援。
2. 如果在老的文件系统上创建新的文件系统,可运行**labelit**命令,它报告老文件系统的文件系统安装名和物理卷名,当建立新的文件系统时,这些标号被破坏。
使用**labelit**命令时,必须指定文件系统类型,例如,如果设备是f0q15d,文件系统是*memo*,卷名是*memo2.0*,将键入:

```
labelit -F ufs /dev/dsk/f0q15d memo memo2.0
```

3. 使用下列命令之一:

```
mkfs -F ufs [-o] [arguments] special size
```

或者

```
mkfs -F ufs [-o] [arguments] special prototype
```

其中的任选项是:

special *vfstab*文件中包含文件系统属性的项。

size 文件系统中的扇区数目。

arguments 任选的实参,是一个由逗号分隔的参数表,它允许对文件系统进行精调,其中比较重要的参数如下:

- **nsect** — 磁盘的每个磁道上的扇区数,其默认值为18。如果对原始磁盘设备执行 **prtvtoc -p** 命令,则扇区数输出为“# sectors”。

- **ntrack** - 磁盘每个柱面上的磁道数,其默认值为9。**prtvtoc -p** 命令的打印输出为“# heads”。
- **bsize** - 文件系统中的文件的基本块大小,它必须是2的幂,其现在的可选值为 4096(默认值)或 8192。
- **fragsize** - 分配给一个文件的最小磁盘空间,它必须是2的幂,其现在的选择范围为 512 至 8192,默认值是 1024。
- **cgsiz**e - 每个柱面组的磁盘柱面数,这个数的范围必须在1至 32 之间,其默认值为 16。
- **free** - 允许的磁盘空闲空间的最小百分比,一旦文件系统容量达到这个限度时,则只有超级用户才能分配磁盘空间。它的默认值为 10。

如果在实参表前有**-o**,则仅需说明想要给出的那些实参,但各实参必须明确地加标号。否则,从左到右对实参进行检查,第一个实参被认为是 **nsect**,第二个被认为是 **ntrack**,以此类推。

作为对这两种格式的说明,下面的两个命令在功能上是一致的:

```
mkfs -F ufs -o bsize=4096, nsect=18, ntrack=9 /dev/rdisk/1s2 35340
```

```
mkfs -F ufs /dev/rdisk/1s2 35340 32 16 4096
```

prototype 一个文件名,它可包含:文件系统所需求的块数、文件系统的目录和文件结构、以及将已存在的文件内容读入到文件系统指令。

4. 运行 **labelit** 命令来回存文件系统名和卷名。
5. 填充新文件系统——例如,从文件系统后援中回存,或者,如果你的系统有两个硬盘,则在已安装的文件系统上执行 **cpio (1M)** (**volcopy (1M)**) 和 **dd (1M)** 命令拷贝文件系统的映象,它们不能转换逻辑块的大小)。

2.2.2.3 选择逻辑块的大小

逻辑块大小是核心用来读写文件的块的大小,逻辑块大小一般不同于物理块的大小,物理块的大小是磁盘控制器所能读写的最小的块的大小,一般为 1024 字节。

使用 **mkfs** 命令来建立文件系统的系统管理员可以指定文件系统的逻辑块大小,逻辑块大小的默认值在 **s5** 文件系统中为 1024 字节 (1K),在 **ufs** 文件系统中为 4096 字节 (4K)。**root** 和 **usr** 文件系统是以 1K 文件系统的方式来发行的。除了 1K 文件系统外,**s5** 文件系统也支持 2048 字节 (2K) 的文件系统。

为系统选择合理的逻辑块大小时,必须考虑性能和空间。对于多数 **ufs** 系统,具有 1K 分块大小的 8K 文件系统产生最好的性能;而对于多数 **s5** 系统,1K 文件系统能产生最好的性能。对于运行在 **s5**(例如 **s5** 文件服务器)下的特殊应用(它们要使用大量的可执行文件或数据文件),2K 文件系统可能是更好的选择。

2.3 安装文件系统

当创建文件系统时,命令行上的唯一的名字(不是原型文件名,假若你使用这个任选项的话)是特殊设备文件名。因为 UNIX System V 的文件系统一般是以其结构中的最高层目录名来引用的,所以创建了文件系统后的第二步必须是将文件系统名连接到它的目录名上。

完成这一工作的基本方法是“安装”文件系统,可使用 OA&M 菜单或 **mount(1M)** 命令来实现。如果文件系统要提供给用户,这一步骤是必须的。

mount 命令的执行使得被安装的磁盘设备与安装目录相对应,它也为 UNIX 系统提供了诸如文件系统类型、安装过程中所使用的任选项和安装的完成时间等等信息,这些信息存放在 `/etc/mnttab` 文件中。

例如,命令

```
mount -F s5 /dev/dsk/1s2 /usr
```

告诉系统将 `/dev/dsk/1s2` 作为 s5 文件系统来安装,并且开始目录为 `/usr`。

如果在 **mount** 命令执行之前,你试图将目录(**cd**)改变到 `usr` 文件系统中的一个目录下,则 **cd** 命令将失败。在 **mount** 命令完成之前,系统不知道 `usr` 文件系统中的任何目录。

可以在一张软盘上完整地定义文件系统,既可把它们作为存储空间也可作为直接存取来使用。然而更普遍的是用户将文件系统拷贝到硬盘的一个目录中,为了完成这一工作,必须首先安装文件系统。

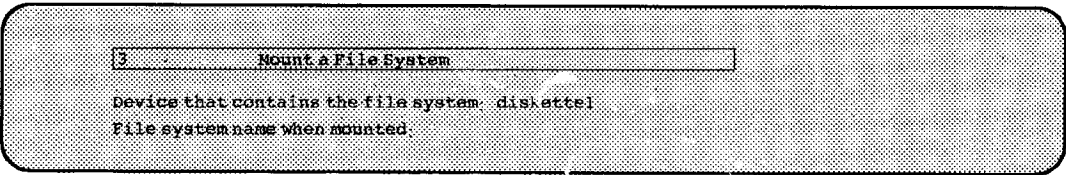
labelit 命令也可用来帮助完成设备特殊文件与文件系统的安装名之间的连接,它将文件系统的最高层次的目录(即它的名字)写到专用块的一个字段中。

注意: 当把 **labelit** 命令用于可删除的文件系统时,如软盘上的文件系统,命令行的一个实参可以是卷标识号。虽然一般的方法仅是将卷名写在贴到包含文件系统的软盘或磁带的标签上,但此卷号也存放在专用块的一个字段中。

2.3.1 使用 OA&M 菜单安装文件系统

在 **System Administration** 菜单中选择 **file_system** 任选项来完成这任务,选择下述的菜单来安装文件系统:

1. 从 **Manage File Systems** 菜单中选择 **mount**, 系统显示 **Mount a File System** 屏幕:



2. 按 **CHOICES** 来选择合法的任选项, 然后按 **SAVE**。

2.3.2 使用 mount(1M) 安装文件系统

mount 命令的一般形式是: **mount**, 例如, 命令

```
mount -F s5 /dev/dsk/ls4 /home
```

告诉系统将 */dev/dsk/ls4* 作为 s5 文件系统来安装, 并且开始目录为 */home*。

一般来说, 用户认为从硬盘上存取文件更方便, 为做到这一点, 要将经常使用的文件从磁带或软盘上拷到硬盘上。

输入下列的命令来安装文件系统并且将其内容拷贝到硬盘上:

1. 在硬盘上创建两个目录: 一个作为软盘和硬盘之间的连接(安装点), 另一个作为被安装的文件系统的根目录。

例如, 下列的命令使用 */mnt* 作为安装点目录, */myfs* 作为根目录。

2. 安装软盘, 例如:

```
mount -F s5 -oro /dev/diskette /mnt    (-oro 意味只读)
```

3. 把当前目录改变到安装目录上; 在本例中为:

```
cd /mnt
```

4. 使用下列命令将文件系统的内容拷贝到 */myfs* 目录中:

```
find . -print | cpio -pdm /myfs
```

有关任选项的说明参见 **find(1)** 和 **cpio(1)**。

2.4 拆卸文件系统

拆卸文件系统逻辑上是解除文件系统与磁盘设备的连接, 用户不能再存取这个文件系统上的目录和文件。

拆卸经常是使用其它命令来操作文件系统的第一步, 例如, 检查和修复文件系统的 **fsck(1M)** 命令要对一个已经拆卸的文件系统进行操作; 拆卸也是关机的一个重要步骤。

在开始拆卸过程之前, 必须关闭要拆卸的文件系统中的所有文件, 并且必须把当前目录改变到不属于该文件系统的一个目录上。例如, 如果你的当前目录 'pwd' 在要拆卸的文件系统中, 则在开始拆卸之前, 必须将当前目录转移出去, 否则, 将得到下列信息:

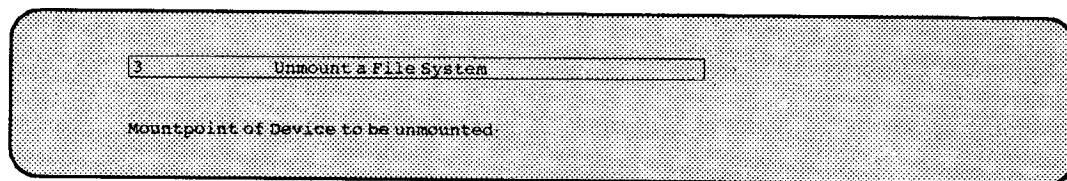
```
/etc/umount: device busy
```

2.4.1 通过菜单来拆卸文件系统

在 **System Administration** 菜单中选择 **file_systems** 任选项来执行这些任务。选择下列的菜单来拆卸文件系统:

1. 在 **Managing File Systems** 菜单中选择 **umount**, 系统将显示 **Unmount a File**

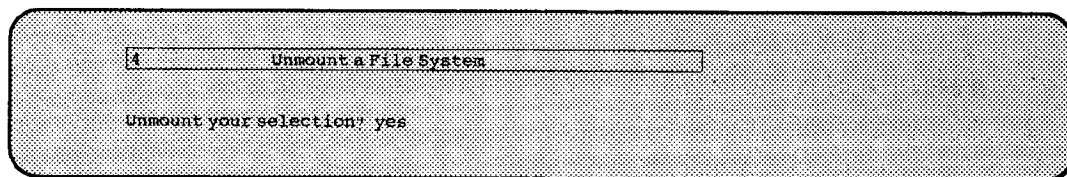
System 屏幕:



3 Unmount a File System

Mountpoint of Device to be unmounted:

2. 按 **CHOICES** 来显示一组可提供的文件系统。使用箭头键将光标移到合适的文件系统上,按 **SAVE** 来选择它。
3. 按 **SAVE** 开始拆卸过程,系统将显示验证屏幕:



4 Unmount a File System

Unmount your selection? yes

4. 按 **SAVE** 继续下去。
拆卸文件系统后,系统显示验证屏幕。

2.4.2 使用 `umount` 命令拆卸文件系统

拆卸文件系统的命令仅要求特殊设备名或安装点。

为拆卸文件系统,在超级用户提示符下输入下列命令:

```
umount device_name
```

例如,如果从软盘上安装了一个文件系统,命令

```
umount /dev/diskette
```

将释放软盘驱动器。

2.5 检查和修复文件系统

本节描述文件系统是如何被破坏的,如何维护文件系统的可靠性以及 UNIX 系统本身是如何对文件系统进行检查的。

2.5.1 文件系统的破坏

文件系统被破坏的原因有许多,下面是三个最常见的原因:

- 非正常的系统关机和启动
- 在拆卸文件系统之前卸出媒质
- 硬件故障

可遵循下述的规则来维护文件系统的可靠性:

- 在关掉计算机之前,总是使用**shutdown**过程,该过程将拆卸所有的文件系统。
- 在抽出被安装的 UNIX 系统软盘之前,总是先拆卸它。
- 当软盘驱动器的红灯亮着的时候,决不要抽出软盘。

2.5.2 文件系统的完整性

计算机在其内部建立了几种可靠性特性,下面是这些特性的一个概括:

- 当向硬盘写文件时,文件的i节点和数据块按能确保最大可靠性的顺序写入,称之为 *ordered writes*。
- 周期性地将系统缓冲区写入到硬盘,以保持文件的内容是最新的,称之为 *automatic update*。
- 如果文件系统已被破坏,则在安装它以前系统要求你运行**fsck**程序来清理文件系统。这就保证了安装到计算机上的所有文件系统的可靠性。

2.5.3 fsck 程序

文件系统检查 (**fsck**) 程序是文件系统的交互式检查和修复程序。**fsck** 使用文件系统本身所带有的信息来完成一致性检查。如果检查到不一致性,则将显示一条说明不一致性的消息。建议你对 **fsck** 选择-y 任选项,使得当 **fsck** 检查到不一致性时自动地修正它们。

在引导 UNIX 系统时,计算机对 *root* 文件系统的状态进行一致性检查,如果存在潜在的问题,则自动地运行 **fsck** 程序来修复 *root* 文件系统。

要手动地执行 **fsck** 程序,你必须首先拆卸文件系统(如果你要检查 *root* 文件系统,必须使得根文件系统是被安装的)。

下面是 **fsck** 命令的一般格式:

```
fsck [-F FSType] [-V] [current_options] [-m] [-o specific_options] [special...]
```

其任选项是:

- F** 指定要操作的文件系统类型, *FSType* 必须在这里指定或必须能通过 */etc/vfstab* 文件中匹配 *special* 而确定文件系统类型。
- V** 回应完整的命令行,但不执行该命令。
- current_options* **fsck** 的 s5 模式所支持的任选项。
- m** 检查但不修复,这个任选项检查文件系统是否适合于安装。
- o *specific_options*** 指定 s5 或 ufs 文件系统类型所特有的子任选项。

如果发现文件系统具有一致性,则报告文件数目,已使用的块数和空闲块数。如果文件系统不一致,则在各种修正之前,向用户提出报告。

2.5.4 使用 fsck 检查 s5 文件系统

下面是 s5 文件系统上使用 **fsck** 命令的格式:

fsck [-F s5] [*generic_options*] [-y] [-n] [-p] [-sX] [-SX] [-t*file*] [-l] [-q] [-D] [-f] [-b] [*special*]

有关这些任选项的完整信息,请参见《系统管理员参考手册》的 **fsck (1M)** 手册页。建议使用 **-y** 任选项,这个任选项对 **fsck** 所提出的所有问题回答 **yes**,并且不要求你的干预。另一个建议的任选项是 **-s**,它按最佳的顺序重新构造空闲表。在使用过程中,空闲表就变得杂乱,空闲表的重新构造能提高以后建立的文件的性能。*special* 指定与文件系统相联系的特别设备的文件名,如果没有指定设备名,则 **fsck** 检查在 */etc/vfstab* 中具有数值的 **fsck pass** 字段的所有文件系统。

下面的屏幕显示了运行 **fsck** 来检查 *usr* 文件系统,其中没有指定任何任选项。系统的回答意味着没有查到不一致性的地方。该命令分多个阶段运行,其中一些阶段仅在要求或对应到命令行任选项时才被执行。每个阶段完成后就显示出一条消息,程序结束时打印出一个表明文件(i节点)数目、块数和空闲块数的总结消息。

```
# fsck -F s5 /dev/dsk/ls2

/dev/dsk/ls2
File System:  usr      Volume:  usr

**phase 1 ~ Check Blocks and Sizes
**phase 2 ~ Check Pathnames
**Phase 3 ~ Check Connectivity
**Phase 4 ~ Check Reference Counts
**phase 5 ~ Check Free List
411 files 4394 blocks 8880 free
#
```

2.5.4.1 s5 中的 fsck 阶段

fsck 程序分多个阶段运行,每个阶段都报告它所检查到的错误。如果错误是 **fsck** 能修复的,则询问用户是否要修复,如果你定义了 **-y** 任选项,则认为对所有的提问都回答“yes”,并且不再向你提问。本节的其余部分描述各个阶段所产生的消息、可能的回答以及相应的出错条件:图 2-1 列出了在 **fsck** 出错消息中所使用的缩写。

图 2-1: fsck 输出中的出错消息缩写

出现在出错消息中的下列缩写具有其后正文所表示的含义:

BLK	块数
DUP	重复块数
DIR	目录名

MTIME	文件最后修改的时间
UNREF	未引用过的
CG	柱面组

当屏幕上出现消息时,下列的单个字母缩写可由其相对的正文替代:

B	块数
F	文件(或目录)名
I	i 节点数
M	文件存取方式
O	文件属主的用户 id 号
S	文件大小
T	文件最后修改的时间
X	连接数
	或 BAD、DUP 或 MISSING 的块数
	或 文件数(依赖于上下文)
Y	修正的连接数
	或 文件系统块数(依赖于上下文)
Z	空闲块数

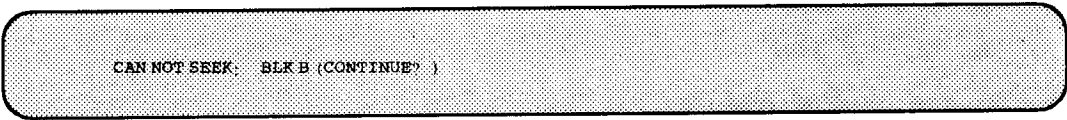
2.5.4.2 初始阶段

检查命令行的语法。在检查文件系统之前,fsck 建立一些表格并打开一些文件,在初始阶段,若 fsck 碰到错误则终止执行。

2.5.4.3 一般错误

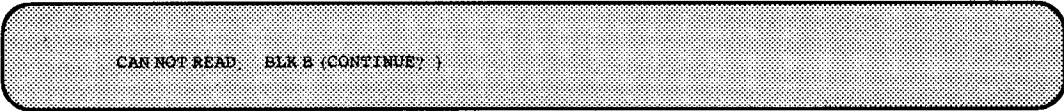
下面的三条出错消息可出现在初始阶段之后的任何阶段中,虽然它们提供了继续执行的任选项,但一般最好是将它们看成是致命的,并终止执行,尽力找出是什么引起这样的问题。

消息



在文件系统中定位到一个指定的块 B 上的请求失败,这个消息指出了一个严重的问题,可能是一个硬件故障。

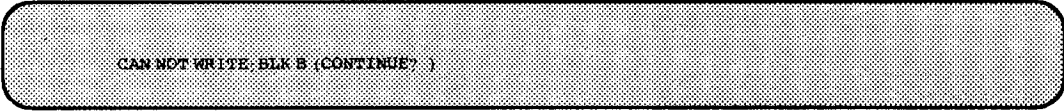
消息



CAN NOT READ: BLK B (CONTINUE?)

在文件系统中读出一个指定的块 B 的请求失败,这个消息指出一个严重问题,可能是一个硬件故障。

消息



CAN NOT WRITE: BLK B (CONTINUE?)

在文件系统中写一个指定的块 B 的请求失败,磁盘可能具有写保护。

2.5.4.4 Yes/No 回答的含义

对 CONTINUE? 提示回答 n(no)的含义是:

终止程序的执行。

(这也是建议的回答。)

对 CONTINUE? 提示回答 y(yes)的含义是:

试图继续执行文件系统的检查。

注意该问题会常常重复出现,这种出错条件阻止了对文件系统的完整的检查,应该第二次运行 **fsck** 来重新检查文件系统。

2.5.4.5 phase 1: 检查块和大小

这个阶段检查 i 节点表,当做下述检查时报告所遇到的出错条件:

- 检查 i 节点的类型
- 建立连接数为零的表
- 检查 i 节点块数中的坏块或重复块
- 检查 i 节点的大小
- 检查 i 节点格式

phase 1 中的出错消息类型

第一阶段产生四种类型的出错消息:

1. 信息性的消息
2. 带 CONTINUE? 提示的消息

3. 带 CLEAR? 提示的消息
4. 带 RECOVER? 提示的消息

在一些信息性的消息和带 CONTINUE? 提示的消息之间存在着一种联系, CONTINUE? 提示通常表示达到了某种极限。

phase 1 中的 yes/no 回答的含义

对 CONTING? 提示回答 n(no)的含义是:

终止程序的执行。

在第一阶段中,对 CONTINUE? 提示回答 y(yes)的含义是:

继续程序的运行。

当出现这个错误时不可能对文件系统进行完整的检查,应第二次执行fsck来重新检查文件系统。

对 RECOVER? 提示回答 y(yes)的含义是:

恢复i节点所指向的所有块。

只有当用户试图删除多余的块时,no回答才是适合的。

对 CLEAR? 提示回答 n(no)的含义是:

忽略出错条件。

no 回答仅在用户试图采取其它步骤来修复错误的情况下才是适合的。

对 CLEAR? 提示回答 y(yes)的含义是:

通过清除i节点的内容来释放i节点I。

这可能在第二阶段中对指向该i节点的各目录产生一个UNALLOCATED出错条件。

第一阶段的出错消息

消息

```
UNKNOWN FILE TYPE i= I (CLEAR? )
```

i 节点 I 的方式字表明该 i 节点不是管道、特别字符类型的 i 节点,也不是一般文件和目录类型的 i 节点。如果定义了-p 任选项,则该 i 节点将被清除。

消息

```
LINK COUNT TABLE OVERFLOW (CONTINUE? )
```

fsck 用来存放连接数为零的已分配的 i 节点的内部表没有空间了。如果定义了-p 任