

吴文东 刘林生 张伟华 编著

科学技术文献出版社

友元 审



NORTON 6.0 磁盘管理

使 用 指 南

NORTON 6.0 磁盘管理 使用指南

吴文东 刘林生 张伟华 编著

友元 审

科学技术文献出版社

(京)新登字 130 号

内 容 提 要

NORTON 磁盘管理软件主要用于磁盘管理,NORTON 6.0 为该软件的最新版本。NORTON 6.0 操作界面友好,能直接对软盘和硬盘进行物理读写,并能自动对磁盘进行诊断,此外,还可以直接对目录树进行整理。因此,NORTON 磁盘管理软件是用户执行磁盘管理、磁盘和文件恢复、软件加密和解密的有效工具。本书共分三大部分:其中包括 NORTON 磁盘探测、NORTON 6.0 操作指南及 NDOS 使用指南。全书语言流畅、内容完整新颖,是广大微机操作人员难得的参考书。

NORTON 6.0 磁盘管理使用指南

吴文东 刘林生 张伟华 编著

友元 审

责任编辑 洪 宇

科学技术文献出版社出版

(北京复兴路15号 邮政编码 100038)

天津市静一胶印厂印刷

新华书店北京发行所发行 各地新华书店经售

*

787×1092 毫米 16 开本 17.5 印张 640 千字

1994 年 8 月第 1 版 1994 年 8 月第 1 次印刷

印数:1—1500 册

科技新书目:310—096

ISBN 7-5023-2174-8/TP·115

定 价:25.00 元

目 录

第一篇 NORTON 磁盘探测	(1)	3.6 网络要求	(73)
第一章 磁盘构造入门	(1)	3.7 兼容性	(73)
1.1 简介	(1)	3.8 命令行用法	(73)
1.2 硬件结构	(1)	第四章 目录排序工具(DS)	(75)
1.3 软件结构	(6)	4.1 概述	(75)
1.4 磁盘中的一些问题	(12)	4.2 功能	(75)
第二章 盘编辑器	(14)	4.3 何时使用	(75)
2.1 Disk Editor 的功能	(14)	4.4 怎样使用	(75)
2.2 何时使用	(14)	4.5 求助信息	(76)
2.3 如何使用	(15)	4.6 交互方式选择项	(76)
2.4 菜单一览表	(27)	4.7 实例	(78)
2.5 操作注意事项	(30)	4.8 命令行使用	(79)
2.6 命令行使用	(30)	第五章 磁盘编辑器(DISKEDIT)	(80)
第三章 故障修复	(31)	5.1 功能	(80)
3.1 引言	(31)	5.2 何时使用	(80)
3.2 数据恢复	(31)	5.3 命令行格式	(81)
3.3 目录修复	(36)	第六章 磁盘监视器(DISKMON)	(82)
3.4 系统区修复	(41)	6.1 功能	(82)
3.5 绝对扇区修复	(43)	6.2 何时使用	(82)
第二篇 NORTON 6.0 操作指南	(47)	6.3 如何使用	(82)
第一章 综述	(47)	6.4 求助信息	(84)
1.1 Norton Utilities 简介	(47)	6.5 网络要求	(85)
1.2 6.0 版中新增加的内容	(48)	6.6 命令行用法	(85)
1.3 咨询信息	(48)	第七章 磁盘实用工具(DISKTOOL)	(86)
1.4 键盘和鼠标的使用方法	(50)	7.1 简介	(86)
1.5 运行 Norton Utilities	(52)	7.2 制做引导盘	(86)
1.6 结语	(60)	7.3 利用 DOS 的 Recover 恢复	(86)
第二章 高效批处理工具(BE)	(61)	7.4 恢复损伤磁盘	(87)
2.1 功能	(61)	7.5 标记簇	(87)
2.2 何时使用	(61)	7.6 建立补救盘	(88)
2.3 如何使用	(61)	7.7 利用补救盘恢复	(88)
2.4 求助信息	(61)	第八章 文件加密器(DISKREET)	(89)
2.5 正本文件	(62)	8.1 概述	(89)
2.6 Batch Enhancer 子命令	(62)	8.2 功能	(89)
第三章 加速及检测磁盘读取速度及 可靠性工具(CALIBRATE)	(70)	8.3 背景知识	(89)
3.1 功能	(70)	8.4 何时使用	(89)
3.2 背景知识	(70)	8.5 关于口令的使用	(90)
3.3 何时使用	(70)	8.6 如何使用	(90)
3.4 如何使用	(70)	8.7 求助信息	(101)
3.5 求助信息	(73)	8.8 与其它工具的关系	(102)
		8.9 内存使用	(102)
		8.10 网络要求	(102)
		8.11 菜单参考	(103)

8.12 命令行用法	(104)	14.6 选项	(120)
第九章 删除保护器(EP)	(105)	14.7 命令行用法	(121)
9.1 概述	(105)	第十五章 文件空间占用检查工具(FS)	(122)
9.2 功能	(105)	15.1 概述	(122)
9.3 何时使用	(105)	15.2 功能	(122)
9.4 怎样使用	(105)	15.3 何时使用	(122)
9.5 求助信息	(106)	15.4 如何使用	(122)
9.6 网格注释	(106)	15.5 求助信息	(123)
9.7 命令行用法	(107)	15.6 实例	(123)
第十章 文件属性设定工具(FA)	(108)	15.7 命令行使用	(123)
10.1 概述	(108)	第十六章 统系区数据映象器(IMAGE)	(124)
10.2 功能	(108)	16.1 功能	(124)
10.3 何时使用	(108)	16.2 何时使用	(124)
10.4 怎样使用	(109)	16.3 怎样使用	(124)
10.5 求助信息	(109)	16.4 求助信息	(124)
10.6 选项	(109)	16.5 网络要求	(124)
10.7 实例	(109)	16.6 命令行用法	(124)
10.8 命令行用法	(110)	第十七章 行式打印工具(LP)	(125)
第十一章 文件日期设定工具(FD)	(111)	17.1 概述	(125)
11.1 概述	(111)	17.2 功能	(125)
11.2 功能	(111)	17.3 何时使用	(125)
11.3 何时使用	(111)	17.4 怎样使用	(125)
11.4 怎样使用	(111)	17.5 求助信息	(125)
11.5 求助信息	(111)	17.6 选项	(125)
11.6 命令行用法	(111)	17.7 命令行用法	(127)
第十二章 文件搜索工具(FILEFIND)	(112)	第十八章 缓存加速器(NCACHE)	(128)
12.1 概述	(112)	18.1 概述	(128)
12.2 功能	(112)	18.2 功能	(128)
12.3 何时使用	(112)	18.3 背景	(128)
12.4 怎样使用	(112)	18.4 何时使用	(128)
12.5 求助信息	(113)	18.5 怎样使用	(129)
12.6 选项	(113)	18.6 选项	(129)
12.7 网络要求	(114)	18.7 装入开关项	(129)
12.8 命令行用法	(114)	18.8 配置开关	(133)
第十三章 文件修复工具(FILEFIX)	(116)	18.9 其它配置问题	(136)
13.1 功能	(116)	18.10 兼容性	(136)
13.2 何时使用	(116)	18.11 实例	(137)
13.3 怎样使用	(116)	18.12 错误信息	(138)
13.4 修补 Spreadsheet 文件	(116)	18.13 命令行用法	(138)
13.5 求助信息	(119)	第十九章 目录管理工具(NCD)	(140)
13.6 兼容性	(119)	19.1 功能	(140)
13.7 命令行用法	(119)	19.2 何时使用	(140)
第十四章 文件定位工具(FL)	(120)	19.3 求助信息	(140)
14.1 概述	(120)	19.4 怎样使用	(140)
14.2 功能	(120)	19.5 快速查找	(141)
14.3 何时使用	(120)	19.6 NCD 中的功能键	(141)
14.4 怎样使用	(120)	19.7 菜单参考	(141)
14.5 求助信息	(120)	19.8 限制	(143)

19.9 网络要求	(143)	24.3 何时使用	(165)
19.10 命令行格式	(143)	24.4 如何使用	(165)
19.11 示例	(144)	24.5 运行 Speed Disk	(166)
第二十章 硬件设置工具(NCC)	(145)	24.6 在批处理模式下运行 Speed Disk	(168)
20.1 功能	(145)	24.7 求助信息	(168)
20.2 怎样使用	(145)	24.8 Speed Disk 菜单	(168)
20.3 求助信息	(148)	24.9 Speed Disk and CHKDSK	(171)
20.4 命令行用法	(148)	24.10 Speed Disk 与其它 Norton 程序	(171)
第二十一章 磁盘医生(NDD)	(150)	24.11 网络要求	(171)
21.1 功能	(150)	24.12 命令行用法	(171)
21.2 何时使用	(150)	第二十五章 系统信息检测工具(SYSINFO)	(172)
21.3 怎样使用	(150)	25.1 功能	(172)
21.4 NDD I 测试	(151)	25.2 何时使用	(172)
21.5 求助信息	(152)	25.3 如何使用	(172)
21.6 网络要求	(152)	25.4 限制	(173)
21.7 限制	(152)	25.5 求助信息	(173)
21.8 兼容性	(153)	25.6 菜单参考	(174)
21.9 命令行用法	(153)	25.7 命令行用法	(176)
第二十二章 Norton 配置设定工具 (NUCONFIG)	(154)	第二十六章 正文搜索工具(TS)	(177)
22.1 功能	(154)	26.1 功能	(177)
22.2 何时使用	(154)	26.2 何时使用	(177)
22.3 如何使用	(154)	26.3 怎样使用	(177)
22.4 求助信息	(155)	26.4 求助信息	(177)
22.5 菜单条目	(155)	26.5 选项	(177)
22.6 口令	(155)	26.6 实例	(178)
22.7 菜单编辑	(156)	26.7 命令行用法	(179)
22.8 视频与鼠标器	(156)	第二十七章 恢复删除文件工具(UNERASE)	(180)
22.9 Norton Cache	(157)	27.1 功能	(180)
22.10 Config.sys	(158)	27.2 背景知识	(180)
22.11 Autoexec.bat	(159)	27.3 何时使用	(180)
22.12 替换名	(159)	27.4 怎样使用	(180)
22.13 展开程序	(159)	27.5 求助信息	(183)
22.14 退出	(166)	27.6 MENU 参考	(183)
第二十三章 安全格式化工具(SFORMAT)	(161)	27.8 与其它实用工具的关系	(186)
23.1 功能	(161)	27.9 命令行用法	(186)
23.2 背景知识	(161)	第二十八章 格式化恢复工具(UNFORMAT)	(187)
23.3 何时使用	(161)	28.1 功能	(187)
23.4 怎样使用	(162)	28.2 背景知识	(187)
23.5 求助信息	(162)	28.3 何时使用	(187)
23.6 CONFIGURE 菜单	(162)	28.4 怎样使用	(187)
23.7 Safe Format 选项	(163)	28.5 求助信息	(189)
23.8 举例	(164)	28.6 与其它工具的关系	(189)
23.9 命令行用法	(164)	28.7 网络要求	(190)
第二十四章 硬盘优化工具(SPEEDISK)	(165)	28.8 命令行格式	(190)
24.1 功能	(165)		
24.2 背景知识	(165)		

第二十九章 信息擦除器(WIPEINFO)	(191)	5.5 多项命令	(206)
29.1 功能	(191)	5.6 条件命令	(206)
29.2 背景	(191)	5.7 换码	(207)
29.3 何时使用	(192)	5.8 命令显示	(207)
29.4 怎样使用	(192)	5.9 参数引用	(207)
29.5 配置设置	(192)	5.10 批文件和内存批文件	(208)
29.6 交换驱动器	(193)	5.11 批文件参数	(208)
29.7 交换文件	(193)	5.12 别名	(208)
29.8 求助信息	(194)	5.13 当这个别名这样执行	(209)
29.9 与其它工具的关系	(195)	5.14 可执行扩展名	(209)
29.10 网络要求	(195)	5.15 KEYSTACK 命令	(210)
29.11 命令行使用	(195)	5.16 环境变量	(210)
29.12 示例	(195)	5.17 变量函数	(211)
第三篇 NDOS 使用指南	(196)	5.18 I/O 改向	(212)
内容简介	(196)	5.19 命令解析	(213)
第一章 导论	(197)	第六章 NDOS 配置选择	(214)
1.1 NDOS 特点	(197)	6.1 NDOS 文件	(214)
1.2 新增命令和增强型命令	(198)	6.2 NDOS 使用内存情况	(214)
1.3 增强型批处理器	(199)	6.3 建立 CONFIG.SYS 配置文件	(214)
第二章 NDOS 试运行	(201)	6.4 NDOS 启动选择项	(215)
2.1 NDOS 文件	(201)	6.5 环境变量 COMSPEC 和次级外壳程序	(217)
2.2 NDOS 帮助系统	(201)	6.6 NDSHELL 环境变量	(217)
第三章 DOS 命令处理器	(202)	6.7 请求在线帮助	(217)
3.1 初级命令处理器	(202)	第七章 NDOS 和 DOS 命令差别	(218)
3.2 次级命令处理器	(202)	7.1 DOS 中的 APPEND 命令	(218)
第四章 NDOS 基础知识	(203)	7.2 NDOS 设置属性命令	(218)
4.1 HELP 和快速 HELP	(203)	7.3 DOS 4.X 格式化命令	(219)
4.2 显示惯例	(203)	7.4 NDOS 4.X 选择命令	(219)
4.3 RETRY、IGNORE、FAIL、ABORT	(203)	第八章 NDOS 命令使用指导	(220)
4.4 NSTART 和 AUTOEXEC.BAT	(204)	8.1 NDOS 命令一览	(220)
第五章 NDOS 使用方法	(205)	8.2 用法约定	(220)
5.1 命令行编辑	(205)	8.3 NDOS 命令使用指导	(221)
5.2 命令历史清单和调用	(205)	附录 A 提示和故障维修	(264)
5.3 命令完整化	(206)	附录 B NDOS 和 DOS 错误信息	(269)
5.4 文件完整化	(206)	附录 C NDOS 编程	(273)
		附录 D 键值代码表	(274)

第一篇 NORTON 磁盘探测

本篇内容包括三个组成部分。第一章“磁盘构造入门”，解释了很多基本概念；第二章“磁盘编辑”，讲解了如何进行磁盘编辑；最后一章，“故障恢复”(Troubleshooter)，逐步讲解了最普通的数据和磁盘的修复过程。

第二章磁盘编辑(Editor)软件是一个强有力的工具，它能够存取和修改磁盘最敏感的区域：文件区、文件分配表(FATS)，以及控制着数据存贮及恢复的目录区。像其他所有实用工具一样，诺顿磁盘编辑软件的使用非常灵活。不过，请不要轻易编辑修改磁盘(尤其是硬盘)的任何地方，除非对要做的事情了解得非常清楚。因为这样做可能使DOS或其他程序(不包括磁盘编辑软件)不能正确存取磁盘上的数据。

要灵活而有效地使用诺顿磁盘编辑软件，必须阅读本部分三章的全部内容。第一章介绍磁盘工作情况，如果是首次接触“簇”、“FAT”、“分区表”(partitionable)等名词，那么必须阅读这一章。

当磁盘编辑软件工作在只读状态时(这是一种假设工作方式)，就能够安全地检索磁盘而不改变其中的数据。一旦学会用磁盘编辑软件查寻磁盘数据，就可以边读“磁盘构造入门”边检索硬盘，并实际地看一看该章所描述的内容，这样，本软件的使用会变得更加方便。

本篇的最后部分，“故障恢复”(Trouble shooter)，将逐步讲解最普通的数据和磁盘的修复过程。对于大多数磁盘故障，诺顿磁盘医生Ⅰ可以自动加以恢复。但是，有些用户更喜欢全部采用手工方式进行数据恢复和磁盘修补工作，“故障检修程序”将提供这一方便。

第一章 磁盘构造入门

1.1 简介

本章的第二节简要地介绍了磁盘的内部工作过程。首先，从硬件的角度考察磁盘和磁盘驱动器：它们是如何构成的，怎样进行工作，各个部分的名称是什么。其次，还要介绍各基本组成部分(如磁盘片、磁头、磁道、磁柱面)的构造、工作过程，以及如何由它们组合成一个磁盘驱动器。

第三节将从软件的观点加以讨论，了解物理和逻辑格式化的不同；详细了解簇的概念以及文件的基本组成部分；考察文件分配表(FAT)；分析一些对磁盘操作有影响的软件实体；最后，以磁盘目录的讨论和一些UnErasing之后的概念作为结束。

在第三节中，偶尔也提及一些第二节中讨论过的内容。使其有一定的独立性。因此，如果时间紧迫，用户可对第二节稍作浏览而立即进入第三节。

磁盘是微机中最有趣的部件之一，它包含的内容较多，所以，本指南只挑选出用户在日常使用中最有用的内容加以介绍。还有一些有关问题，例如磁盘管理，也是非常有趣而且很有介绍价值的。但是，如果本指南全部加以介绍，就会过于臃肿而不精练。因此，也就不再画蛇添足了。

1.2 硬件结构

磁盘特别是硬盘一直以复杂闻名，令人惊奇的是，有关的基本概念却非常简单。

尽管有关知识你可能早已知道，还是有必要重复一遍：磁盘(无论是硬盘还是软盘)存贮数据的原理与录音带和录像带的原理一样，都是通过磁头磁化媒介表面微粒子的方法来实现的，运动的微粒子通过磁头，就被磁化。

录音带(计算机磁带也一样)是由很长的软质带上镀上一层金属粒子构成的(顺便说一下，磁带上最普通的附着物质的成分是氧化铁，氧化铁的另一状态是铁锈。软盘外表之所以呈褐色，是因为上面覆盖着一层“锈”)，这种

长而薄的带子并没有什么魔术,只要拿着附有可磁化媒介的东西贴着磁头作机械运动,就能够使用附有媒介的任何形状的东西录上磁信号。

1.2.1 磁道

想象一下用食指在铅笔头上抹一下,然后放到正在唱片机中旋转着的唱片上方。如果用沾着墨水的指头轻轻接触唱片,唱片上将留下一指宽墨迹形成的圆圈。现在把唱片想象成软盘,指尖想象成磁性读写头,则指头在磁盘上留下的轮迹有一个专业名称:磁道。

在读写头上加上数据字信号,就能够在软盘的磁道上录下信息。录音带上录制信号也是采用同样的方法。不同的是,录音带上录制的是随声波变化而变化的连续信号;而软盘上录下的或者是峰值信号或者没有信号。换句话说,录音带上录制的是模拟信号,软盘上录制的是数字信号。

再来看一看磁盘上的圆环形磁道,因为物理磁头的要求,磁道特别窄。这意味着磁盘上的单个数据磁道只占整个盘的极小部分。但是,正像能够录制两道、四道或更多道信号到一根磁带上一样,一张磁盘上也可以复合录制很多磁道。所采用的方法也和录音带录制方法类似。方法有两种:一种是增加磁头,录制下一个磁道的磁头排在前一个磁头的内侧(其磁道半径比前一个稍小)。另一种方法是单个磁头在磁盘上来回移动。第一种方法花费高但速度快,因为多个磁道可以同时录制信号。第二种方法(控制单个磁头在磁盘上移动)要慢一些,但是却经济得多,因而大多数的制造商都是采用这种方法(并不是全部,有些高性能的磁盘驱动器每个磁盘表面就采用了多个磁头,借以使速度大大提高)。

1.2.2 双面磁盘

至此,已经讨论了只有一个面的软盘。可是,软盘也和唱片一样,有上顶面和下底面。一般磁盘两面都涂上磁性物质,以便使用者能在两面都录上信息(在这方面,磁盘和磁带有所不同。磁带只有一面涂层且只能在一面上录制,软盘的情况与之相反;如果只在一面涂上磁性物质,则软盘就会向另一面弯曲而变形)。

采用双面磁盘将产生很大的经济效益。磁盘驱动器进行单面读写时必须有下列部件:一个使磁盘旋转的马达,一个把磁头伸向磁盘的支杆,另一个控制磁头在磁盘上来回移动的马达,以及组装这些部件的框架(底座)等。如果要在另一面读写,就不需要再增加框架、马达、或任何其他过多的东西。只要加大控制磁头部件的拉伸力(现在这个部件将弯成类似于钳子的东西),再对应地在原磁头对面像钳子一样增加另一个磁头就行了。

1.2.3 磁柱面

在磁盘两面录制信息还有一点好处:可以写上两磁道的数据而不需移动磁头。第一磁道的数据写在磁盘上顶面的磁道上,然后,不移动磁头把另一部分数据写到底面的磁道上。这样不移动磁头就能读写的上下对应的双磁道合在一起,称为磁柱面。

为了使用方便,磁柱面和磁道都被编号。软盘最外面的磁道是0磁道;上顶面的称0面0磁道,下底面的称1面0磁道。或者也可以把两个磁道合在一起称为0柱面。PC机的360K软盘有40个柱面,编号从0柱面(最边缘)到39柱面(离中心轴最近)。磁盘中的编号一般从0开始。有一例外是扇区号,是从1开始的。簇号是另一例外,而且很不寻常是从2开始的。后面我们会介绍簇。

现在继续完成磁盘驱动器的讨论。在此以前,我们描述的主要是双面、双磁头软盘驱动器的优点。下面将全面考虑。对于软盘,磁头是直接压在磁盘的磁涂层上的。这样,磁头就有可能读写到最强的信号而出现故障(磁头和磁涂层之间的距离增加,信号的强度就会下降)。此外,摩擦限制了磁盘的转速。还有一个棘手的问题是如何保持磁头和软质磁盘长时间地接触,而与其同样麻烦的问题还有怎样设计磁头才能不成为一把凿刀。因为转速直接影响到读数速度,软盘速度慢是在所难免的(读二进制位的速度不能超过磁头移动的速度)。

可供选择的解决途径之一(硬盘采用的方法)是把磁头稍稍抬高盘面,以消除因磁盘表面细微不平造成的摩擦,同时也解决了磁头对磁盘表面的磨损问题。但是,这一途径也引起了一系列其他问题。因为读写信号的强度随磁头与盘之间的距离增加而成指数下降,所以让磁头尽可能地贴近磁盘是极其重要的。同时,由于各部件温度上升而膨胀和磁盘表面的缺陷,又必须保证磁头距盘面有足够的距离,以免相互接触。

硬盘制造商在这种要求下采取了一个巧妙的方案:把磁头设计成像一架微型飞机或微型滑翔机。

优质硬盘系统的标志就是飞速转动的硬盘激起一层气流,而磁头则漂浮在上面,你可以想象一下这层极薄的空气,它大约只有10微寸(十万分之一英寸),就连人的头发都不能从磁头和磁盘之间滑过。

尽管以上的优点是主要的。硬盘系统中要解决的问题也是存在的。由于必须保证磁盘尽可能接近而又不触及

磁盘,使用像软盘一样的软质材料做磁盘就根本不行了。原因很简单:每分钟 3600 转的高速下,软盘会发生波动和晃动,无法在它上面装上一个距离只有几个微寸的磁头。

还有一个关键问题就是保持磁头和磁盘之间的滑动。

尽管以上的优点是主要的。硬盘系统中要解决的问题也是存在的。由于必须保证磁头尽可能接近而又不触及磁盘,使用像软盘一样的软质材料做磁盘就根本不行了。原因很简单:每分钟 3600 转的高速下,软盘会发生波动和晃动,无法在它上面装上一个距离只有几个微寸的磁头。

还有一个关键问题就是保持磁头在硬质磁盘几微寸的高度上滑翔。因为磁头离盘极近,防止灰尘微粒落在盘面造成堵塞就尤为重要。滑翔着的磁头碰上微粒,就会碰及磁盘,破坏数据。因此,必须密封整个磁头装置(HDA, Head-Disk Assembly)并且过滤进入密封箱的空气。为了防尘,硬盘部分的装配一定要在洁净度很高的房间里进行。这也是硬盘造价昂贵的原因之一。

由于需要洁净度很高的装配房间,硬盘绝大部分的花费来自 HDA,因而必须尽可能地压缩其成本。在软盘驱动器中,增加第二个磁头,只增加很少一点制造成本。与此相同,在 HDA 中增加一个或两个额外的盘片,也只提高一点制造成本。标准的 20 兆字节硬盘,就是两张盘装在同一轴上。并且四个磁头(每张盘每面一个)安装在同一操作臂上成纵列运动。

多盘片硬盘的编号办法很简单,把双面磁盘驱动器的编码办法加以扩展即可,第一张盘的顶面是 0 面,底面是 1 面;第二张盘的顶面是 2 面,底面是 3 面。和双面磁盘驱动器划分柱面的方法类似,这里把两张盘的 0 磁道组合在一起作为 0 柱面。

在硬盘中,一个磁道的数据容量取决于所使用的数据记录方式。对于现行的记录方式(各种方式中主要的一种,叫 MFM,即 Modified Frequency Modulation)每一磁道能够正确读写的数据容量是 8 至 12KB。

尽管一次从盘上读写 8~12KB 的数据是完全可能的,但以过去的观点看来,这样数目的数据对实际操作来说还是太大了。因此,无论是硬盘还是软盘,实际上都设计为每次只读写一个磁道的一部分数据。这部分数据的字节数(更一般的说法是扇区的字节数)取决于磁盘的控制器硬件和操作系统。厂商设计的磁盘控制器支持几种不同的扇区尺寸,而操作系统则在其中选取可用的一种。扇区的典型尺寸有:128、256、512 和 1024 字节。PC-DOS 从 1.0 版起,提供给硬盘和软盘的扇区大小是 512 字节。

在软盘中,每一个磁道现在可以很安全地容纳 18 个扇区。不知为什么,IBM 公司却很保守地设计为每磁道仅使用九个扇区(在 DOS1.1 版只使用八个扇区)。每磁道九扇区乘以每面 40 磁道,再加上每张盘有两面,这样就构成了大家非常熟悉的 368640 字节的软盘。

由于更高的转速、硬质的盘面、以及远远超过软盘的严格的制造精度,硬盘每面能容纳更多的磁道,每一磁道能容纳更多的扇区。最惊人的改进还在于硬盘每面的磁道数大大超过软盘可能的磁道数。一个标准的硬盘,每英寸可划分的磁道数在 600 以上(可以想象,磁头要在这些极狭窄的磁道上方精确定位,需要怎样的精密程度。一次移动六个磁头横过三个磁盘并不比一次移动一个磁头横过一个磁盘花费多很多,这就是组合式硬盘如此普遍的原因)。

硬盘能够在给定的磁道上支持更多的扇区,虽然这种改善并不惊人。标准硬盘每磁道可以有 17 个扇区,当采用 RLL(Run-Length Limited Coding)记录模式时可以有 25 或更多的扇区。有些高容量磁盘使用加强型 RLL 记录方式(Advanced Run-Length Limited Coding),它允许每磁道有 33 个或更多个扇区。

高容量是硬盘超过软盘的两个主要长处之一,另一个是速度。

磁道划分成扇区改善了某些问题(需要缓存的数据减少了),另一方面又增加了新问题。现在,查找一批数据,不但要指定盘面和磁道,而且还要给出指定磁道的扇区号。

如果看看广告或读读硬盘的规格说明表,就会碰到一些用来评测磁盘性能的描述。特别地,像寻址时间、存取时间、等待时间和数据传输率等参数指标会出现在任何有关磁盘性能的论述中。检测程序也会测量出这些特性参数。

1.2.4 寻址时间

寻址时间是磁头从初始磁道移到读数据磁道的时间。它依下述条件而定:磁头在哪儿,距离目的磁道有多远。显然,每次读盘的寻址时间都各不相同。所以,在两种寻址时间中,最重要的还是道→道寻址时间(道间寻址时间)。

1.2.5 道间寻址时间(Track-to-Track Seek Time)

道间寻址时间是从一个磁道到相邻的另一磁道的寻址时间。AT 级磁盘驱动器的道间寻址时间处在 8 到 10 毫秒之间。软盘的寻址时间是硬盘的很多倍。

1.2.6 存取时间

存取时间是磁头找到需要存取数据的磁道所花费的时间。进一步说,如果磁头刚刚在 3 磁道读完数据,现在要到 4 磁道读数据,此时的存取时间就等于道间寻址时间。可是,能有如此幸运总是读相邻的磁道吗?(实际上,这种情况比想象要经常得多,对磁盘的老用户来说尤其是这样)。

1.2.7 平均存取时间

平均存取时间是从当前位置到读数磁道的平均时间的一种量度。初看起来,平均存取时间似乎等于寻址横穿过磁盘一半磁道所需要的时间。可是,这仅仅在磁头总是停在磁盘边缘的情况下才正确。而实际上,磁盘操作过后,磁头可能停留在盘面上的任何地方。最佳的情况应该是磁头停在磁盘中间位置。如果这样,磁头将寻址横穿过的磁道数平均起来是整个磁道数一半的一半,即全部磁道的四分之一。

如果认为平均存取时间介于四分之一和二分之一磁道寻址时间之间,就更为准确。实际上,查寻任意磁道的平均时间等于查寻全部磁道的三分之一所花费的时间(注意:这里的平均存取时间独立于高效的操作系统采用的最佳方法,即把顺序读取的数据按顺序的磁道存放。在实际使用的这种方法里,大多数时间的寻址只需移动一个磁道)。

在磁头读取数据之前,两个条件必须满足:第一,查寻到所需磁道;第二,所读磁道的一个特殊的扇区(即要在读的多个扇区中最先读的扇区)应该转到磁头下面。已讲述过,磁头找到目标磁道所需时间的平均值就是平均寻址时间。找到目标磁道后,磁头查寻到正确扇区所需时间叫做等待时间。

1.2.8 平均等待时间

平均等待时间就是磁盘旋转半周所需的时间。硬盘的转速是 3600 转/分,即每 16.67 毫秒转一周。所以等待时间大约为 8.3 毫秒。所有的硬盘转速相同,这一指标(等待时间)不会因不同的硬盘而有所不同,因此,等待时间不能作为比较硬盘的参数。软盘的转速是 300 或 360 转/分,其平均等待时间在 100 毫秒以上,十倍于硬盘的平均等待时间。

1.2.9 数据传输率

数据传输率是以磁盘上读取二进制位的速率,也就是二进制位从磁盘传输到主机的速率。数据传输率取决于磁盘转速的快慢(找到数据才能读取),以及二进制位在磁道上存放的稠密程度。硬盘的传输率显著地超过软盘,因为硬盘转速高(3600 转/分比 300 转/分),而且二进制位存储密度也大。

现在绝大多数硬盘的数据传输率为每秒 5MB。由于逐渐采用新的硬盘界面标准,如 ESDI(Enhanced Small Device Interface),数据传输率已提高到每秒 10MB 甚至 15MB。

1.2.10 扇区寻址

前面已经概要地说明了硬盘中的等待时间,即在找到所需磁道后,等候需要的数据旋转到磁头下所花的时间。我们真正等候的是一个特殊的扇区,希望它飞速到来。问题是:当查寻到这个扇区时,如何来识别它?

早期的软盘采用物理方法进行扇区鉴别。绕磁盘一周按有规律的间隔穿上小孔。每一个孔都被机器识别,并标志着一个扇区的开始。这种方法称为硬分区。

除了很不精确以外,硬分区还没有适用于高性能的驱动器。所以,一种更先进的扇区识别方法得到了发展,这就是对每一个扇区进行编址并存入扇区数据中(扇区绕磁道顺序编号,因此扇区地址也绕磁道顺序编排)。这种方法称为软分区。

迄今为止,已经详细叙述了磁盘驱动器的物理特性:磁道和面的编号,每一扇区的大小,存取时间等等。所有这些特性都由硬件决定(即由驱动器装置本身及其控制器来确定)。

正如上面所述,在读一个扇区之前,必须设置某种标记信号到扇区上。对于硬分区,区的地址实际上已经写入本扇区中(写地址的工作在格式化时进行)。扇区地址将作为扇区数据固定的首部。首部中,与扇区地址一起,还写

着一些特殊的同步字节。这些字节出现在每一个首部的开始部分,它们按照唯一的顺序排列,以便让磁盘控制器知道将开始读扇区地址。为叙述完整,扇区中的间隔字节也必须介绍一下。间隔字节也叫填充字节,它被放置在扇区之间,用来作为每一个扇区的同步检测标记。

如果扇区首部被破坏或很难读取,“Sector not Found”的错误信息将被显示出来,扇区中的数据也会丢失。标准程序能够修复这种扇区首部,防止这类数据丢失。

1.2.11 物理格式化

把扇区地址、同步字节、间隔字节、及少量其他各种各样的数据块写入扇区首部的工作称为硬、物理或低级格式化,因为这些工作只能由磁盘控制器中的硬件来完成。在硬格式化期间,主机中的软件操纵磁盘控制器格式化一个磁道,选取一个有效的扇区尺寸,并且提供少量其他参数;在这之后,进行格式化就是控制器自己的事了。

在任何软分区磁盘被用来存贮数据之前,物理格式化是必需的。第二步,另一称之为逻辑格式化(马上将要介绍)的过程,也必须在磁盘准备存贮数据之前完成。

下列情况可能混淆物理格式化和逻辑格式化过程:在执行 DOS 格式化或诺顿格式化命令时,对软盘将完成两个过程的工作,而对硬盘只进行逻辑格式化。当格式化软盘时,格式化命令首先完成物理格式化,接着再进行逻辑格式化;当格式化硬盘时,物理格式化将被跳过。

下面就来看看物理格式化硬盘时的情形。

1.2.12 交叉因子

硬格式化的主要工作是把每一扇区地址写入扇区数据首部。最先设想的是扇区绕磁道顺序编号(这似乎是不言而喻的),在这一点上,实际情况也基本如此。但是,这并不是法定的,不按顺序编号也有充分的理由。

假设有一台计算机,其性能对硬盘来说极其完美,当数据从磁盘控制器流出时,计算机总是设法立即将数据存入内存,并且恰好抢在下一批数据到来时返回与控制器接通。

这是一种理想状况。实际生活中,事情极少如此顺利。经常的是,一旦磁盘找到了所需磁道,其数据输出速度就要超过计算机接收速度。这样,就会出现以下情况:控制器读出一个扇区的数据,计算机将它存好。接着,计算机再返回控制器,准备接收下面的数据。但是,这时下一个顺序扇区已过去,第三个顺序扇区却刚好从磁头下掠过。现在,计算机只好转为等待状态,一直到磁盘旋转近一周,第二扇区再次转过来为止。

这是一种很坏的配合状况,因为磁盘每旋转一周只能提供一个扇区的数据。由于扇区地址只是简单的数据(尽管是一组特殊数据),是可以被改变的,所以也可以环绕磁道按顺序每隔一个扇区标一号码。

开始环绕磁道时,扇区标号为1;然后跳过一个扇区,再标出2扇区;再跳过一个扇区,标3扇区,等等。当环绕又回到开始的地方时,10扇区就在1扇区和2扇区之间;11扇区在2和3之间;这样一直到17扇区,它在9扇区和1扇区之间(假定是17扇区磁盘)。把扇区相隔数从1改为2,再按标号顺序读扇区时,就有两倍的时间来处理一个扇区,因而能够在下一个顺序扇区到来前完成存贮工作。

对于以上交叉因子为2的情况,不再需要转17周才能读取全部磁道数据,而只要2周即可。这当然没有一周读取全部数据好,但是比每周只读一个扇区好多了。

改变交叉因子是协调硬盘和主机性能的一种方法。但是请注意:就像刚才所看到的那样(即最初没有间隔的例子),如果计算机(处理速度)不落后于磁盘的话,降低交叉因子将使性能急剧下跌。

标准(Calibrate)程序将为用户计算机系统计算好最佳交叉因子,并且不需对磁盘备份就可直接调整成最佳交叉因子。

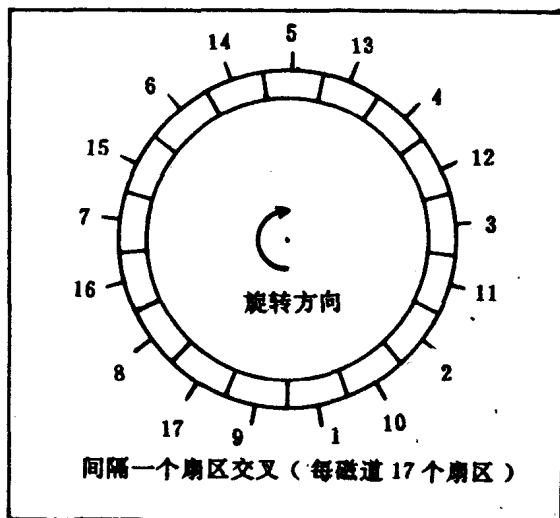


图 1-1 交叉因子

1.3 软件结构

1.3.1 逻辑格式化

前面,我们已处理了磁盘硬件的物理特性:每片盘的磁道或柱面的数目,盘片数,扇区的字节数,扇区的数目,相隔因子,等等。这些特性或者是不可改变的(如盘片数目),或者是由硬件按照操作系统的指令设置的(设置之后就很少改动)。还有另一组磁盘特性,它与操作系统在磁盘上编排查寻数据有关。

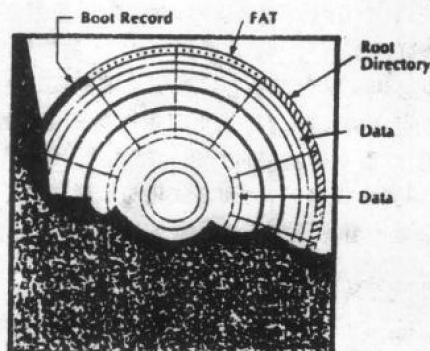
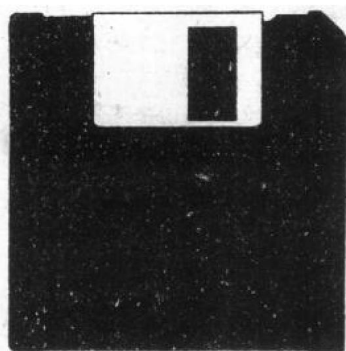


图 1-2 磁盘结构

在存取 20MB、40MB 或更多的字节数据时,为了速率和效益,操作系统建立目录和索引显然是必需的。目录和索引将告诉操作系统:什么数据在哪儿,磁盘的哪一部分是被说明占用的,哪一部分是自由的,甚至哪一部分因物理损坏而永远不能使用。把这些信息编排到磁盘上的方法叫磁盘的逻辑格式化,填写各种各样支撑这一编排的目录和索引的过程称为逻辑格式化磁盘。如同很早所描述的,当使用 DOS 格式化命令(或我们的安全格式化程序)格式化一个硬盘时,实际上所做的全部工作就是逻辑格式化;而物理格式化早已完成。对于一张软盘,DOS 格式化命令首先进行物理格式化,然后自动进行逻辑格式化;而安全格式化则不同,如果需要,它可以只进行物理格式化。

无论使用的哪种类型的磁盘,即硬盘或软盘,固定或可更换式,360K 或 60MB 磁盘,DOS 总是使用同一种逻辑格式化,它把磁盘分成四个主要的区域:引导区、文件分配表(FAT)、根目录区和数据区(硬盘能按不同操作系统进行分区,所以有第三个区域叫分区表。我们会作简单介绍)。

1.3.2 引导区

DOS 的引导区总是占据着第一面的第一磁道的第一个扇区,即 0 面 0 磁道 1 扇区(实际上,严格地说仅仅对软盘才正确,因为硬盘的第一扇区是存放分区表的)。引导区所做的正是其名字所表示的:首先,读取一段非常短的程序(即引导代码),然后执行这段程序,依次读取操作系统的其它部分,从而使计算机自动启动(从空脑子的白痴变成高效率的管理者)。

除了很小的引导程序以外,引导区还包含着操作系统其他必需的数据信息:

磁盘关键参数表。表中的参数有每扇区的字节数、磁盘扇区总数、每一磁道上的扇区数、以及磁头的数目。由于这个表的重要性,所有的磁盘都将在逻辑格式化期间建立引导区,那些不包含三个系统文件(这些文件是引导启动所必需的)或系统的软盘也不例外。

DOS 在逻辑格式化期间建立的第二和第三个区域是 FAT 和根目录,它们都用于记录第一个文件存放的磁

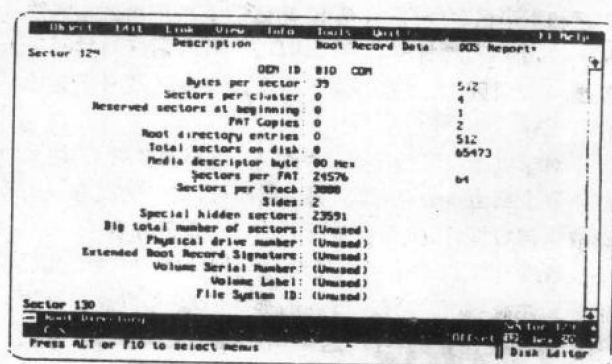


图 1-3 引导记录区

道。

1.3.3 簇

磁盘格式化后,磁盘控制器一次能物理地读写的最小数据块是一个扇区,对于目前的 DOS 磁盘,就是 512 字节。DOS 能够(1.2MB 高密软盘中就是这么做的)记载每一个单个扇区的状态,不管这个扇区是已占用的、自由的还是因损坏而不能使用的。但是,在大容量磁盘中,采用一个一个扇区登录的方法记载磁盘,占用的空间太大。所以,DOS 使用了扇区群,这种扇区群称为簇。

在进一步了解簇和 DOS 如何把文件记录到磁盘上之前,我们先看一看 DOS 在建立、扩张、压缩和删除文件时遇到的一些问题。

1.3.4 文件变动

管理计算机文件的基本问题是文件变动(自然,能轻易地改动计算机文件也是很有吸引力的)。请考虑一下以下的例子:假设有一个 2250 字节的文件,我们要把它写到一张新的未用过的 1.2MB 软盘上。在磁盘上读写的最小字节数是一个扇区,这也确实是 DOS 将在 1.2MB 盘上读写的最小字节数目。那么,存贮这个 2250 字节的文件就需要 5 个扇区。前四个扇区存放 2048 字节,最后这个扇区仅存放 202 字节。既然不能写一个扇区的局部,DOS 将把第五个扇区全部给予该文件,最后扇区的剩余空间将浪费。这个新文件将占据软盘最前面的五个扇区。

现在再写第二个文件,其长度为七个扇区。它将占据数据区接下来的七个扇区。现在设想:要在第一个文件后增加 460 字节的数据。回想一下,最后的 512 字节扇区仅仅占用了 202 字节,因此还有 310 字节的空闲。但是,留下的 150 字节(460 减去加到最后扇区尾部的 310 字节)将放到哪儿呢?

我们不能简单地就把它们放到文件的五个扇区后面的第一个扇区里,因为那一扇区已经用于下一个文件(七扇区文件)。可以把下一个文件往后移一个扇区,但是要花费时间。不仅如此,如果后面一兆字节(1MB)的所有扇区都被文件占用,这种方法则极其低效:写小小的 150 字节要移动 1MB 的数据。这不是一个好的解决办法。

1.3.5 文件分配表

DOS 采用的一个非常好的解决办法是使用一个存放所有扇区状态的表。这个重要的表被称之为文件分配表,即 FAT,它将指明一个给定的扇区是自由的还是已被某个文件占用的。这样,当要扩展一个文件时,只需查看 FAT,寻找下一个自由扇区并将它分配给文件就可以了。在上面的例子中,前面 5+7 个扇区已被使用,因而第一个自由扇区就是数据区的第 13 扇区。这样,第一个文件就占据了数据区前面五个扇区,再加上数据区第十三扇区。文件被分割了。开始部分存放在磁盘前面几个相邻的扇区里,而结尾部分则存放在另一地方。这并不成为问题,只要能想出办法把结尾部分和开始部分链接起来。以便 DOS 任意查寻就行了。

为清楚起见,我们说 DOS 保存着一个表,称为 FAT,它记载着磁盘上所有扇区的状态。这并不确切。对于一个 32MB 磁盘,登录每个扇区状态的表会有 65535 个表目(32MB 除以每扇区 512 字节)。DOS 为了存取迅速,使用磁盘时将把 FAT 存入内存中,而大磁盘的 FAT 会很快占完内存空间;除此之外,搜寻一个 128K 的 FAT(64K 表目(entry)乘上每个表目 2 字节)也将花费一段时间。由于这些理由,DOS 把一些相邻的扇区组合在一起,作为一个单位“簇”来处理。DOS 在文件分配表中登录的实际上是簇而不是扇区。

因为簇是 DOS 的创造物,DOS 就能按自己的需要来确定一个簇由多少扇区组成。与其固定一个簇的尺寸,DOS 更愿意采用不同尺寸来适应磁盘媒介;任何一个磁盘的簇尺寸将在逻辑格式化期间确定,并且不再改变(除非再次格式化)。AT 机 30MB 硬盘簇尺寸为 4 扇区,即 2048 字节。1.2MB 软盘格式化的簇尺寸是一个扇区:512 字节。小的簇尺寸反映了速度与可用空间的最佳使用之间的折中。对于小一些的簇尺寸,处理磁盘的时间要长一些。因为有更多要处理的 FAT 表目,而且需要更多的读写操作。在另一方面,簇尺寸越小,每一个文件尾部浪费的空间就越少。

前面,我们讨论了一个 2250 字节长,在高密盘上占据五个扇区的文件。因为高密软盘每一个簇由单个扇区组成,在文件尾的最后扇区(或簇)中,仅仅浪费了大约 200 字节。但是,假设把同一文件移到 30MB 硬盘上,簇由格式化定为四个扇区,又会怎样呢?DOS 按一个一个簇用 FAT 来登记存贮空间;由于 32MB 盘的簇尺寸可能是四扇区(在 DOS3.X 版中),在这个盘上,DOS 能够分配的最小空间是 2048 字节。2250 字节的文件将占有两个簇,因为现在每一个簇是 2048 字节,文件尾上最后簇浪费的字节数从 200 左右跳到 1800 以上。

那些簇尾不能使用的空间称为碎块。一个文件从硬盘移到软盘时会缩小,从软盘移到硬盘时会扩大,碎块就是原因。对于大一些的簇尺寸,必然会带来更大的碎块,作为补偿,得到大尺寸的同时也能得到更好的性能。

1.3.6 FAT 补充

再回到 FAT。前面已经讲过, DOS 在每一磁盘的开始部分存放着一个表, 称为文件分配表, 即 FAT。我们知道: FAT 用于记载哪些簇是自由的, 哪些簇被占用以及被什么文件占用。我们也知道: 簇只不过是一个简单的单扇区或扇区群, 它们的产生减少了记载硬盘空间使用情况的记录量。迄今为止, 仍然不清楚的是 DOS 如何把要记录的信息写到 FAT 中去。

FAT 本身设计得很简单。对磁盘上的每一个簇都分配两个字节(DOS3.0 及以上版本都如此)。每一个四字节的顺序编号与数据区中同样编号的簇直接对应。这样又出现了新的端口号: DOS 是如何对簇编号的。

1.3.7 三种计数方式

还在我们观察磁盘的物理划分时, 就知道磁盘控制器采用了一种特殊的三坐标系统地址: 盘面数(对于软盘是 0 或 1), 磁道数(双面双密软盘是 0 到 39), 以及扇区数(360K 软盘), 因而 ROM-BIOS 也通过盘面、磁道和扇区来确定磁盘内地址。但是, 对操作系统来说, 这个三坐标系统则很不方便, 因为各个磁盘的磁道、扇区、盘面的数目都不一样。DOS 采用一维扇区编号方案来确定数据地址: DOS 简单地按顺序对所有的扇区进行编号, 从 0 面 0 道的 1 扇区开始, 编完 0 道的全部扇区后再编 1 面 0 道。一个柱面的所有扇区处理完后再处理下一柱面; 这样, 顺序读取扇区时, 磁头的移动最小, 寻址时间也最短。DOS 使用的连续扇区编号方案称为逻辑扇区编号; 它不像物理扇区编号始于 1, 而是从 0 开始。

不要因 DOS 顺序扇区编号的叙述而引起混乱, 这里没有物理格式化的继续, 也没有地址实际写到磁盘上。所发生的仅仅是 DOS 提供的扇区方式与 BIOS 及磁盘控制器提供的方式之间的转换, 就像把英文中的“Sector One”变为法文中的“Premiere Secteur”一样。

确定了 DOS 的扇区编号方案后, 簇的编号方案很快就能得到了。首先, 必须跳过下列区域所占用的扇区: 引导记录(一个扇区), 文件分配表 FAT(随磁盘容量的不同而变化)和根目录区(还没有讨论过)。这三个区域总是在逻辑格式化时建立, 并且被认为是系统区域部分而排除在簇编号方案之外。

除上述系统区域之外的所有扇区(开始于根目录区之后的第一个扇区)被认为是数据区域部分。这部分包括在簇编号方案之中。在一个四扇区为一簇的磁盘上, 数据区域最前面的四个扇区称为 2 簇(簇的编号从 2 开始, 不是 0 也不是 1), 接下来的四个扇区是 3 簇, 再下面四个扇区是 4 簇, 这样一直到磁盘的最后扇区。一个 30MB 的磁盘, 以四扇区为一簇, 将有 15230 个簇(数据区域); 最后一个簇的编号为 15231, 它比簇的数目还大——是因为簇是从 2 开始编号的。

1.3.8 再论 FAT

作了以上讨论后, 我们再一次回到 FAT。FAT 包含有磁盘每一个簇的状态信息, 并且磁盘上每一个簇都有一个 FAT 表目(entry)相对应。最前面的两个 FAT 表目用来放特殊信息。第三个 FAT 表目存放第一个簇的有关信息(这个簇编号为 2), 下一表目存放第 2 个簇有关信息(3 簇), 如此直到末尾, 最后的 FAT 表目存放磁盘最后一簇的状态。(严格地说, 我们现在论述的某些细节除了适用于单分区的硬盘外, 也适用于软盘。这一节的最后我们将介绍硬盘分区, 无论有没有分区, 上述磁盘的基本格式都是一样的)。

下面来看看 FAT 表目需要提供哪些信息。首先, 由于即使是最好的磁盘也可能有坏扇区, 所以只要簇中有坏扇区, 其 FAT 表目就应该告诉 DOS; 而 DOS 则知道永远不把这个簇分配给文件。实际情况的确如此, 这是 DOS 在逻辑格式化期间建立 FAT 时要记录的信息之一。在完成格式化磁盘之后(写引导记录、创建原始的 FAT 和根目录), DOS 会试读磁盘的每一个扇区, 那些包含有不能正确读取的扇区的簇将作为坏簇标记在 FAT 中。这一过程称为坏扇区映象(mapping), 这对于确保数据完整是极其重要的。

顺便提一下, 你一定觉得奇怪, 从一张刚格式化并被认为全空的软盘上, DOS 能读出什么数据来。这是一个很好的问题, 答案是这种“空”盘并不是空的, 仅仅是没有有用的数据而已。当 DOS 或 Safe Format 格式化软盘时, 用把一个特殊的字节(十六进制数 F6)填满全部数据区域。在这一格式化过程之后, 再从每一个扇区中试读这个字节。

可能有些扇区不能被磁盘控制器读出; 在这种情况下, 控制器返回一错误信息给 DOS, DOS 则标记上这个簇含有坏扇区。也可能有一些扇区可以被控制器读出, 但读出时有循环冗余校验字节错, 此时将标出读数据错。

1.3.9 循环冗余校验

循环冗余校验又称为 CRC。当写入一个扇区时,将计算一个特殊的校验和值(称为循环冗余校验),这个值取决于写入扇区的所有字节的值。这个值(即 CRC)紧接在扇区数据之后写在磁盘的一个特殊位置上。当从磁盘上读出扇区时,它的校验和(或 CRC)字节段也将读出。同样的计算也将进行,不同的是这次取决于刚刚读出的数据。刚读出的数据的 CRC 和以前计算出并写在磁盘上的 CRC 进行比较。如果二者不相配,就意味着读出的数据与写入的数据不完全相同,即出现了读数据错。诺顿磁盘诊断 II(The Norton Disk Doctor II)程序将把包含有这种扇区的簇标记为坏簇,以便不再使用这一区域。

再说一下,当用户要求 DOS 进行磁盘拷贝验证时(或者使用带/V 参数的拷贝命令,或者使用命令 SETVERIFYON),DOS 并不是把源盘和目标盘上的数据一个字节一个字节地比较,而是简单地重读目标盘上的文件,确认没有 CRC 错或任何其他类型的读数据错。如果 CRC 检验通过,拷贝的数据就是正确的。

言归正传,下面继续完成 FAT 的讨论。到目前为止,我们已经看到,对于每一个簇都必须在其对应的 FAT 表中保存两个事实:簇是不是坏的,簇是自由的还是已被其它文件占用的。如果文件不被随机地增长、缩短或删除,则上面所说的就是要保存在 FAT 中的全部信息。可是实际情况并非如此,回想一下前面讲过的例子:在磁盘最前面写入一个很短的五个簇的文件,接着再写入第二个与之无关的文件。现在需要在第一个文件后附加一些数据,显然在第一个文件后取用下一个顺序簇是不行的,因为它已经被第二个文件占用。我们只能在再往后更远的某个地方选用一个簇。这样就会出现一个问题:当 DOS 读取第一个文件时,它如何知道在哪里能找到远离前一部分的最后那个簇?

答案很简单。我们把文件前五个簇的数据写到磁盘前面五个相邻的簇内,然后跳过一些簇,写入文件的第六个簇:为什么能在相邻的最后一个簇(第五个簇)的 FAT 表目中存入一个指针绕过那些跳过的簇而直接指向文件的第六个簇呢?如果文件被存入 2 簇到 6 簇,然后再加上 14 簇,我们就可以把 14 簇写入 6 簇的 FAT 表目中。它的意思就是:“可以在磁盘 14 簇的位置找到文件的下一部分”。为统一起见,可以在每一个用于文件的 FAT 表目中建立指向下一个 FAT 表目的指针,这样, FAT 表目就形成一条链,每一个表目都指向它的下一个。因为每一个 FAT 表目的序号与其对应的簇的编号一样,要确定文件每一簇的位置,就可以沿 FAT 中的链进行跟踪。下面就是一个 DOS 跟踪实例。

如果你已经读到这里,对于 DOS 的磁盘文件的基本结构你会有很好的理解。下面再补充完成一个简单的细节。那就是:现在我们已经知道如何通过跟踪 FAT 中的链来找到文件的所有簇,那么,怎样才能确定这条链的起始位置呢?在这个实例中,六个簇的文件第一个簇也就是磁盘数据区域的起始簇,但是这只是因为我们指定它是这样而已。到了明天,怎样才能找到它呢?那时还能记得这个实例中第二个文件的起始位置吗?

很明显,我们需要在每个文件中的 FAT 表目启动处列一个表。如果我们知道第一个 FAT 表在什么地方,我们就能确定文件第一个簇的位置(请记住,每一个 FAT 表目对簇进行相类似的标号),因为在 FAT 链中,每一个 FAT 表目都指出了下一个表目的位置,所以我们能够找到所有有关这个文件的簇。

除了列出起始簇编号的表以外(这个表可以使我们找到每一个文件的第一个簇),我们还需要确定每一个文件的最后一个簇的位置。可以把最后簇的信息放在存放起始簇信息的同一地方,采用列出最后簇的编号或文件的大小尺寸的办法。要不然,也可以利用 FAT 本身,把一个特殊的字节放入对应于每一个文件最后簇的 FAT 表目中。

DOS 采用了以上两种办法:文件目录中记录了每个文件的长度和起始簇号码,每一文件的最后 FAT 表目中设置了一个特殊的标记。

关于极其重要的 FAT 的讨论已进入尾声。FAT 是一个两字节表目表(对 DOS3.XX 而言),一条表目对应于磁盘上的一簇(早一些 DOS 版本每条 FAT 表目使用 1.5 字节。这节的末尾将讲述其重要性)。每一表目将显示

File Name	First	Link	Size	1000	Date	Time
16364	16973	20297	822	79211	1984	0
0	0	0	24576	4882	2	27591
16365	17485	21327	8224	21771	1984	0
0	0	0	24576	7562	14	26632
20545	21328	8224	8224	417	4829	0
0	0	0	21282	124	20	0
20292	8275	8224	8274	8274	4128	0
0	0	0	21208	218	7	0
20362	21586	20047	8274	8274	412	0
0	0	0	3412	4	1	0
20361	21333	4261	8274	4128	0	0
0	0	0	25229	4882	19	0
21675	20308	22597	17271	11266	6476	0
0	0	0	38322	1447	3968	862
16375	2239	21671	8274	8274	412	0
0	0	0	21208	218	7	0
14387	19266	22593	8224	4224	4728	0
0	0	0	24186	5744	4728	0
16717	19529	8224	8224	8274	4128	0
0	0	0	29345	1142	25	0

图 1-4 文件分配表中的簇链

出其对应的簇是可用的,有缺陷的还是已经由文件占用的。如果簇是被占用的,其表目或者指出文件的下一个簇 FAT 表目,或者标识出簇是文件的最后簇。因此,FAT 表目中包含四种数值中的一种:

值(十六进制)	含意
0000	簇是可用的
0002-FFFF	文件占用的簇(数值指示文件的下一簇)
FFF0-FFF6	保留,不用
FFF7	坏簇;不能使用
FFF8-FFFF	文件的最后簇

指示每个文件的起始 FAT 表目的信息在文件目录中。

1.3.10 根目录

目录(根目录或主目录)是任何由 DOS 格式化的磁盘中系统区域的第三也是最后一部分。如果使用磁盘编辑器(Disk Editor)查看目录,可以发现,每一项(entry)存放了比起始簇号码和文件大小尺寸更多的信息。显然,除了尺寸和起始簇号外,每一目录项必须列出与起始簇号对应的文件名。

从另一方面来看,磁盘上的每一个文件都对应一目录项;文件的目录包含有指针指出 FAT 中链的起始表目,即文件的第一个簇。除此之外,每一个 32 字节的目录项包括有时间日期域,一个字节的文件属性域,以及由 DOS 保留的十个字节。每一目录项的格式如上图。

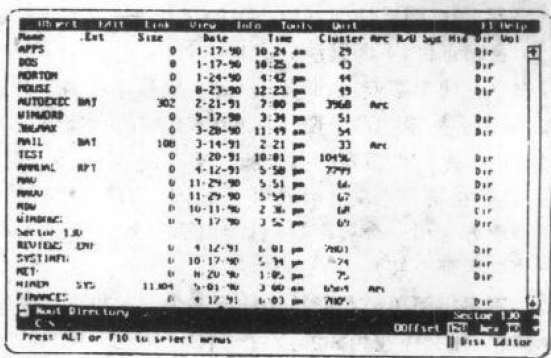


图 1-5 目录项

说明	长度(字节)	形式
文件名	8	ASCII 字符
扩展名	3	ASCII 字符
属性	1	每一二进制位代表一项属性
(保留)	10	没用
时间	2	字,编码(word,coded)
日期	2	字,编码
起始 FAT 表目	2	字
文件(长度)	4	长整数(Longinteger)

文件属性代表了各文件的一些特殊的性质,如只读、隐藏、系统、卷标、子目录、档案。文件属性按位标识,若对应位为 1,则这一属性被设置。

卷标属性代表着一个不常用的情况。任何一个实际存在的文件都不会有这一属性。一个磁盘中只有一个目录项可能有卷标属性,并且这个目录项一定在根目录中。标号占据目录项的文件名和扩展名域,这种情况下它们是一个整体。

我们首先来看一看根目录与子目录之间的差别。到目前为止,我们所讲述的根目录似乎是磁盘中仅有的目录,的确,根目录是目录中最特殊的。首先,根目录是唯一不在磁盘数据区 7 的目录;它是系统区的第三元素,并且总是在紧接 FAT 之后的位置上。根目录的大小尺寸和位置都是固定的,它们创建于逻辑格式化期间而且以后不再改变。其次,其固定的尺寸也是根目录不同于子目录间的地方,子目录可根据需要创建、增长、缩短以及删除。

根目录的尺寸随着磁盘的字节数不同而变化。对于 360K 软盘,格式化建立的根目录占有的空间是 112 个目录项;对于大多数的硬盘,格式化建立的根目录有 512 个目录项。这意味着,如果想要在 32MB 的硬盘上存放超过 512 个文件,就必须创建子目录。

根目录中的任何一个目录项都可提供给一个文件或一个子目录。子目录是一个混和物。它就像文件一样也被分配空间。它也能够增长、扩张和被删除。但是,它不是用来存放数据,而是用来存放文件名。可以这样认为,子目录形式上像任何其他文件,功能上却与根目录完全一样。因为子目录存储贮在数据区域,所以可以按它的需要给它分配空间。