

数据恢复的 原理与方法

胡向东◎著

SHUJU HUIFU
DE YUANLI YU
FANGFA

胡向东◎著

数据恢复的 原理与方法

 广东高等教育出版社
Guangdong Higher Education Press

· 广州 ·

图书在版编目 (CIP) 数据

数据恢复的原理与方法/胡向东著. —广州: 广东高等教育出版社, 2011. 8

ISBN 978 - 7 - 5361 - 4122 - 3

I. ①数… II. ①胡… III. ①数据管理 IV. ①TP309.3

中国版本图书馆 CIP 数据核字 (2011) 第 160316 号

出版发行	广东高等教育出版社
	社址: 广州市天河区林和西横路
	邮编: 510500 营销电话: (020) 87553335
	网址: www.gdgjs.com.cn
印 刷	湛江日报社印刷厂印刷
开 本	787毫米×1 092毫米 1/16
印 张	22.75
字 数	420 千
版 次	2011 年 8 月第 1 版
印 次	2011 年 8 月第 1 次印刷
印 数	1~3 000 册
定 价	80.00 元

(版权所有, 翻印必究)



前 言

数据恢复与电脑应用的关系，就如同疾病防治与人的健康一样。所谓数据恢复，就是指通过技术手段把各种原因导致的常态下不能正常读出的保留在介质中的数据重新恢复到能够正常读出的状态的过程。数据之所以能够被读出，既有物理的因素，也有逻辑的因素，因此只要存储介质没有严重受损或数据没有被完全覆盖，数据就有可能完全或大部分恢复过来。

随着社会的进步和科技的发展，电脑及数码产品越来越普及，人们对电脑及数码产品的依赖程度越来越高，存储在电脑及数码产品中的数据越来越重要，而因数据丢失造成的风险也越来越大。因此，使更多的人了解数据恢复的原理和方法，有利于预防及自保，甚至可以造福他人，造福社会。这是出版这本书的目的之一。

笔者读过大学，也当过大学老师，现在也有很多高校的朋友，深感我们国家的高等教育与实际有些脱节，计算机专业的本科生、研究生对数据恢复的原理和方法知之甚少，与用人单位的需求相去甚远。因此，希望这本书对这些学生或希望增加这方面知识的人能有所帮助。这是出版这本书的目的之二。

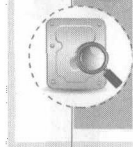
时代在不断发展，新陈代谢是不可抗拒的历史规律，长江后浪推前浪，一代新人超旧人。笔者从1983年9月从事计算机应用研究以来，特别是1992年1月下海创建数据修复公司以来，一直重视把经历过的重要的数据恢复案例记录并加以总结，经过多年的拼搏，现在也该退下来用另一种方式发挥余热了，希望笔者的经验得到传承和发展。这是出版这本书的目的之三。

1985年1月，笔者在郑州举行的“中国统筹法、优选法与经济数学研究会”上受到著名数学家华罗庚先生的接见，并且有幸在广州以应用数学学会的名义两次参与接待过华老，华罗庚先生的事迹和教诲，时时警醒和激励着我们努力努力再努力。数据恢复的核心就是数学，作为华罗庚先生的崇拜者，笔者希望能够以这本书向中国应用数学的先驱华罗庚先生表示敬意。这是出版这本书的目的之四。

在“6.2 固件问题数据恢复”的写作中，得到了广州正大电脑数据修复有限公司胡志德、谢浩然、莫家明、张贤智先生的大力帮助，在此表示深深的谢意。

由于计算机存储技术的飞速发展，加上笔者水平有限、本书篇幅有限等原因，本书错漏之处敬请读者原谅并指正。书中提及的观点、例子或自编软件等如有不尽完善之处，可通过邮件进行交流。笔者邮箱为：huxiangdong1108@163.com。

胡向东
2011年1月



目 录

第 1 章 FAT 数据恢复	(1)
1.1 FAT 文件系统概述	(1)
1.1.1 FAT12、FAT16、FAT32 文件系统的共同点	(1)
1.1.2 FAT12、FAT16、FAT32 文件系统的不同点	(1)
1.2 主分区表修复	(2)
1.2.1 主分区表	(2)
1.2.2 修复主分区表的方法	(4)
1.3 逻辑分区参数表修复	(11)
1.3.1 FAT 逻辑分区参数表	(11)
1.3.2 修复 FAT 逻辑分区参数表的方法	(13)
1.4 FAT 表修复	(38)
1.4.1 FAT 表概况	(38)
1.4.2 注意 FAT 表首区标志	(39)
1.4.3 利用 FAT 副本修复 FAT 正本	(40)
1.4.4 利用 FAT 正副本的比较进行互补修复	(43)
1.4.5 仿造若干扇区的 FAT 表	(46)
1.5 目录项修复	(51)
1.5.1 FAT 分区短文件名目录结构	(51)
1.5.2 FAT 分区长文件名目录结构	(53)
1.5.3 仿造根目录	(56)
1.5.4 对付目录变文件的病毒	(63)
1.6 误删除的修复	(65)
1.6.1 用数据恢复软件扫描	(68)
1.6.2 利用 TMP 文件来恢复	(70)
1.6.3 利用被删除文件中文字之间的内在联系来恢复	(71)
1.6.4 根据被删文件的数据结构来恢复	(73)
1.7 FAT 格式化后的修复	(76)
1.7.1 FAT 格式化成 FAT	(76)

1.7.2	FAT 格式化成 NTFS	(88)
1.8	FAT32 转换成 NTFS	(93)
1.8.1	FAT32 转换成 Windows XP 及以上版本的 NTFS	(93)
1.8.2	FAT32 转换成 Windows 2000 及以下版本的 NTFS	(97)
1.9	误克隆	(99)
1.9.1	盘对盘克隆, 多分区克隆成一个分区	(99)
1.9.2	盘对盘克隆, 分区起点有差异	(102)
1.9.3	分区对分区克隆	(104)
1.10	扇区移位	(107)
第 2 章	NTFS 数据恢复	(112)
2.1	主分区表修复	(112)
2.2	逻辑分区参数表修复	(114)
2.2.1	NTFS 逻辑分区参数表	(114)
2.2.2	修复 NTFS 逻辑分区参数表方法	(116)
2.2.3	NTFS 的主文件表 MFT	(131)
2.2.4	修复主控文件 \$MFT	(179)
2.2.5	修复 MFT 位图	(182)
2.2.6	NTFS 格式化后的修复	(184)
2.2.7	NTFS 转换成 FAT32 出错	(201)
2.2.8	误克隆	(206)
第 3 章	NOVELL 数据恢复	(210)
3.1	NOVELL 分区表	(210)
3.2	HOTFIX 指针区	(212)
3.2.1	HOTFIX 指针区位置	(212)
3.2.2	HOTFIX 指针区大小	(212)
3.2.3	HOTFIX 指针区副本	(212)
3.2.4	HOTFIX 指针区标志	(212)
3.2.5	HOTFIX 指针区的 3 个重要指针	(212)
3.3	卷表指针区	(220)
3.3.1	卷表指针区位置	(220)
3.3.2	卷表指针区大小	(220)
3.3.3	卷表指针区副本	(220)
3.3.4	卷表指针区标志	(220)
3.3.5	卷表指针区第 1 个扇区解析	(220)

3.3.6 一卷跨多盘的卷表指针区	(222)
3.4 卷名	(231)
3.5 文件分配表 (FAT)	(232)
3.5.1 文件分配表 (FAT) 的作用	(232)
3.5.2 文件分配表 (FAT) 的安全性	(232)
3.5.3 对文件分配表 (FAT) 的处理	(232)
3.5.4 文件分配表 (FAT) 构成	(233)
3.5.5 文件分配表 (FAT) 首区特征	(234)
3.6 余数扇区队列	(236)
3.6.1 余数扇区队列的作用	(236)
3.6.2 余数扇区队列的构成	(237)
3.6.3 余数扇区队列的代码	(237)
3.6.4 余数扇区队列的起始位置	(237)
3.6.5 余数扇区队列的存储	(241)
3.6.6 余数扇区队列指针	(241)
3.7 目录项	(242)
3.7.1 目录项的作用	(242)
3.7.2 根目录首区标志	(242)
3.7.3 目录项的长度	(242)
3.7.4 目录项的分类	(242)
3.7.5 文件的目录项数据格式	(242)
3.7.6 目录的目录项数据格式	(247)
3.8 块	(252)
3.8.1 块的作用	(252)
3.8.2 0 块的起点	(252)
3.8.3 块大小的选择	(252)
3.8.4 每块扇区数的判定	(252)
3.9 卷表重建	(254)
3.10 回收站	(279)
3.11 对 NOVELL 的修复工具	(280)
3.11.1 在 DOS 下对 NOVELL 的修复工具	(280)
3.11.2 在 Windows 下对 NOVELL 的修复工具 CAPTAIN	(282)
第 4 章 卷集数据恢复	(284)
4.1 卷集定义	(284)
4.2 卷集数据恢复方法	(284)



- 4.2.1 对同一存储体上连续的卷集，可用更改分区表恢复卷集数据 (284)
- 4.2.2 用无缝驳接的方法，把分布在各存储区域的卷集数据首尾相接拷贝到另一个大硬盘中 (285)
- 4.2.3 使用同款硬盘做替身，重造卷集配置文件 (286)
- 4.2.4 做好备份的前提下，在原盘上设卷集，重造卷集配置文件 (286)

第5章 阵列数据恢复 (288)

- 5.1 阵列类型 (288)
 - 5.1.1 RAID0 阵列 (288)
 - 5.1.2 RAID1 阵列 (290)
 - 5.1.3 RAID5 阵列 (294)
 - 5.1.4 RAID5 + 1 阵列 (296)
 - 5.1.5 RAID5EE 阵列 (296)
- 5.2 掌握阵列数据恢复的七大因素 (297)
 - 5.2.1 服务器中阵列个数 (297)
 - 5.2.2 阵列中硬盘个数 (299)
 - 5.2.3 阵列类型 (301)
 - 5.2.4 标头数 (302)
 - 5.2.5 阵列每块扇区数 (303)
 - 5.2.6 阵列硬盘排列顺序 (310)
 - 5.2.7 阵列硬盘离线先后 (333)
- 5.3 阵列硬盘里保存的阵列卡信息 (337)

第6章 其他问题数据恢复 (340)

- 6.1 坏区问题数据恢复 (340)
 - 6.1.1 有个别坏区的数据恢复 (340)
 - 6.1.2 坏区较多的数据恢复 (345)
- 6.2 固件问题数据恢复 (347)
 - 6.2.1 硬盘固件的定义 (347)
 - 6.2.2 固件存放的位置 (347)
 - 6.2.3 硬盘固件中主要模块（以 MAXTOR, HITACHI/IBM, WD 为例） (348)
 - 6.2.4 修复硬盘固件的工具——PC3000 (351)
 - 6.2.5 修复硬盘固件恢复数据的思路 (351)
 - 6.2.6 实例 (352)



第1章 FAT 数据恢复

1.1 FAT 文件系统概述

FAT 文件系统是微机问世后历史最长的一个文件管理系统，它经历了 FAT12、FAT16、FAT32 三个阶段，目前的 Windows 系统不断地更新版本，但仍然兼容 FAT 系统，可见 FAT 文件系统的巨大影响。

FAT 是文件分配表 (File Allocation Table) 的英文缩写，FAT 文件系统就是以簇来划分数据区，每簇扇区数为 2 的非负整数次幂 (1、2、4、8、16、32、64 或 128)，每一簇对应一个 FAT 项，文件分配表 FAT 就是用来表示某个簇的分配及链接情况。FAT 文件系统由此而得名。

FAT 文件系统的两个重要的数据结构是 FAT 表及目录项。

1.1.1 FAT12、FAT16、FAT32 文件系统的共同点

① 使用 FAT 表管理数据区的簇空间，且都有物理上相连的 FAT 表正副本。

② 使用逻辑扇区号管理系统区。

③ 所有的簇空间都从 2 开始编号 (NTFS 的簇空间从 0 开始编号，见本书第 2 章)。

④ 所有的目录关系都纳入树状结构，每个子目录开头都有子目录父子关系的说明。

⑤ 所有的参数都列在逻辑 0 区偏移量为 0BH (H 表示十六进制) 的 I/O 参数表中。

1.1.2 FAT12、FAT16、FAT32 文件系统的不同点

① FAT12 的每个 FAT 项占 1.5 字节 (12 位)，FAT16 的每个 FAT 项占 2 字节 (16 位)，FAT32 的每个 FAT 项占 4 字节 (32 位)，这是最大的不同。

② FAT12 管理空间不超过 32 M，FAT16 管理空间不超过 2 G，FAT32 管理空间可超过 2 G。



③ FAT12、FAT16 的系统区包括保留区（内含逻辑 0 区），FAT 正副本，根目录区，但 FAT32 的根目录区不包括在系统区中，FAT32 的系统区只包括保留区（内含逻辑 0 区）和 FAT 正副本。

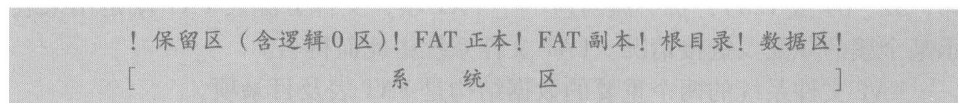
④ FAT12、FAT16 的根目录由于包含在系统区中，因此根目录项数受到限制（一般硬盘根目录只占 32 个扇区，根目录项数最多不超过 512 项）；而 FAT32 的根目录被当作文件处理，因此，根目录项数除了受分区容量限制外，一般不受到限制。

⑤ FAT12、FAT16 的逻辑 0 区只有一份；FAT32 的逻辑 0 区另有一份副本，放在逻辑 0 区正本之后逻辑 6 区处。

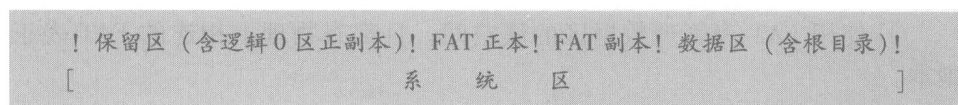
⑥ FAT12、FAT16 的保留区一般为 1（只含逻辑 0 区）；FAT32 的保留区则长很多（多为 32 扇区以上），设计上有利于 FAT 表的保护。

⑦ FAT32 由于在根目录和逻辑 0 区备份等问题上大大改进，因此在逻辑 0 区偏移量为 0BH 的 I/O 参数表中，FAT32 与 FAT12、FAT16 在结构安排上有很大不同。

FAT12 与 FAT16 分区结构：



FAT32 分区结构：



1.2 主分区表修复

1.2.1 主分区表

一个硬盘通常被划分成若干个分区，每个分区的起始位置及分区大小是由硬盘的主分区表所描述。硬盘的第一个扇区（LBA 的 0 扇区，即 0 道 0 头 1 区）偏移量为 01BEH 开始的 64 个字节，就是硬盘的主分区表。

一个分区表可以有 4 个分区表项，每个分区表项有 16 个字节。分区表在 1FEH 处有“55AA”标志。

每个分区表项各字节的含义如下表：

偏移 (H)	字节	含 义	以 10 M 硬盘为例
0	1	80 为活动分区, 可做启动盘, 00 则不可启动	80
1	1	本分区开始磁头号	01
2	1	低 6 位是分区开始扇区号, 高 2 位是分区开始磁道号的高 2 位	01
3	1	分区开始磁道号的低 8 位	00
4	1	分区类型标示: 01: FAT12, 每个 FAT 指针占 1.5 个字节, 分区不大于 32 M 04: 分区不大于 32 M 的 FAT16, 每个 FAT 指针占 2 个字节 06: 分区不大于 2 G 的 FAT16, 每个 FAT 指针占 2 个字节 0B: 分区可大于 2 G 的 FAT32, 每个 FAT 指针占 4 个字节 0C: 分区可大于 2 G 不同版本的 FAT32, 每个 FAT 指针 4 个字节 05: 不大于 8 G 的扩展分区标志 0F: 大于 8 G 的扩展分区标志	01
5	1	本分区结束磁头号	03
6	1	低 6 位是分区结束扇区号, 高 2 位是分区结束磁道号的高 2 位	51
7	1	分区结束磁道号的低 8 位	30
08—0B	4	本分区前使用过的扇区总数	11 00 00 00
0C—0F	4	本分区的总扇区数	F3 50 00 00

上述定义只是硬盘容量不大于 8 G 时的传统描述方法。例中是早期 10 M 硬盘 (305 × 4 × 17) 的 FAT12 主分区表, 但随着硬盘容量的不断增大, 特别是阵列的使用越来越广泛, 目前, 大多数分区表除了要求分区类型标志、本分区前使用过的扇区总数、本分区的总扇区数这三项必须准确外, 对于分区开始或结束的磁道、磁头、扇区数, 描述要求不那么严格, 要么用 1024 作模, 采用余数来描述磁道号, 要么用模糊描述的方法, 在分区表项第二第三字节写 “C1 FF” 或 “C0 FF”, 第六第七字节写 “FF FF”。

注: “×” 只是一个分隔符, 常见作用有两个: 一是在表示硬盘参数时, 如 305 × 4 × 17 表示该硬盘参数为 305 柱面 (磁道), 4 磁头, 每道 17 扇区; 二是表示扇区位置,

如 $1275 \times 0 \times 1$ 表示该扇区位置在 1275 柱面（磁道），0 磁头，1 扇区处。以下同。

例

80 G 硬盘 ($9964 \times 255 \times 63$) 只分 1 个 FAT32 分区，它的主分区表有以下两种写法。

写法一（同余法）：

80 01 01 00 0B FE BF EB
3F 00 00 00 AD 7F 8A 09

写法二（模糊法）：

80 01 01 00 0C FE FF FF
3F 00 00 00 AD 7F 8A 09

由于主分区表含有硬盘各分区起始指针和分区大小的参数，因此主分区表的研究在数据恢复中占有重要位置。

1.2.2 修复主分区表的方法

1.2.2.1 注意分区标志“55AA”

例

4.3 G 硬盘分 2 个 FAT32 分区，是 Windows 98 系统，突然死机后不能启动。把这个硬盘做从盘，用另一个 Windows 98 系统启动硬盘来带，也看不到该硬盘资料。

检查：该硬盘 $0 \times 0 \times 1$ 的主分区表在分区描述上是正确的，但该扇区偏移量为 01FEH 处的“55AA”标志被改为“D5AA”。这是硬盘主分区表不被识别的原因。

解决方法：把该硬盘 $0 \times 0 \times 1$ 扇区偏移量为 01FEH 处的“D5AA”改为分区标志“55AA”，并存盘，则该硬盘可正常工作。

1.2.2.2 利用备份信息修复主分区表

例

1 G 硬盘 ($1023 \times 32 \times 63$) 分 3 个 FAT16 分区，是 Windows 98 系统，电脑公司在维护时，格式化 C 盘并重装系统后，原来装有重要数据的 D 盘和 E 盘不见了，连盘符都不能进去。

**检查:**

① 该硬盘 $0 \times 0 \times 1$ 扇区的主分区表如下所示:

```
80 01 01 00    06 0F 3F CC
3F 00 00 00    F1 26 03 00
00 00 01 CD    05 0F FF FF
30 27 03 00    E0 94 0C 00
```

从分区表参数可看出, 电脑公司在维护时, 电脑主板对硬盘的 CMOS 设置不是采取通常的 LBA 模式, 而是采取 NORMAL 模式, 即只认 16 个磁头, 每道 63 扇区。

② 硬盘 $0 \times 0 \times 6$ 扇区中, 有引导区病毒故意隐藏的原硬盘 $0 \times 0 \times 1$ 扇区的主分区表, 如下所示:

```
80 01 01 00    06 1F 3F 65
3F 00 00 00    01 23 03 00
00 00 01 66    05 1F FF FE
40 23 03 00    E0 54 1C 00
```

从原分区表参数可看出, 用户原来使用时, 电脑主板对硬盘的 CMOS 设置是采取通常的 LBA 模式, 即认 32 个磁头, 每道 63 扇区。D 盘扩展分区起始于 102 (即 66H) $\times 0 \times 1$ 扇区处。

③ 硬盘 $0 \times 1 \times 1$ 扇区的逻辑 0 区, 有新格式化的 C 盘 I/O 参数表, 磁头数为 16, 每道扇区数为 63, 分区大小为 0326F1H。

解决方法:

① 把电脑主板对硬盘的 CMOS 设置由 NORMAL 模式改为 LBA 模式, 即把硬盘的 CMOS 设置改为 1023 磁道, 16 磁头, 每道 63 扇区。

② 用 WinHex 等磁盘工具, 把硬盘 $0 \times 0 \times 6$ 扇区拷贝覆盖到 $0 \times 0 \times 1$ 扇区。

③ 在硬盘 $0 \times 1 \times 1$ 扇区的逻辑 0 区的 C 盘 I/O 参数表, 把磁头数改为 32, 每道扇区数为 63, 分区大小改为 032301H。

④ 核对 $102 \times 0 \times 1$ 扇区, 可看到 D 盘的扩展分区表, 如下所示:

```
00 01 01 66    06 1F 7F FC
3F 00 00 00    E1 84 0C 00
00 00 41 FD    05 1F FF FE
00 85 0C 00    CD CF 0F 00
```

该 D 盘的扩展分区表与硬盘 $0 \times 0 \times 6$ 扇区保留的硬盘主分区表相匹配, 更证明了我们的判断。

⑤ 重启该硬盘, 原来进不去的 D 盘和 E 盘资料又出来了, 资料完整无损。

1.2.2.3 利用逻辑0区I/O参数表仿造主分区表

例1

40 G 硬盘 (4865 × 255 × 63) 分 1 个 FAT32 分区, 是 Windows 98 系统, 突然死机后不能启动。把这个硬盘做从盘, 用另一个 Windows 98 系统启动硬盘来带, 也看不到该硬盘资料。

检查: 该硬盘 0 × 0 × 1 的主分区表被病毒破坏, 而其他重要部位的扇区信息未受破坏。

解决方法:

① 该硬盘参数为 4865 × 255 × 63, 只分 1 个 FAT32 分区, C 盘逻辑 0 区在 0 × 1 × 1 分区。由于是 1 个 FAT32 分区, 根据 C 盘逻辑 0 区在 0 × 1 × 1 扇区的 I/O 参数表的描述, C 盘分区大小为 04A89182H。因此, 可确定硬盘 0 × 0 × 1 扇区的主分区表应为:

```
80 01 01 00    0C FE FF FF
3F 00 00 00    82 91 A8 04
```

② 用 WinHex 等软件工具, 把另一个没有病毒的 Windows 98 硬盘 0 × 0 × 1 扇区拷贝覆盖到该硬盘 0 × 0 × 1 扇区上。

③ 把上面确定的硬盘主分区表写入到该硬盘 0 × 0 × 1 扇区偏移量为 01FEH 处, 替换硬盘主分区表, 改后存盘。

④ 重启该硬盘, 则原 Windows 98 系统正常工作, 数据保存完整。

例2

200 M 硬盘 (987 × 12 × 35), 分 2 个 FAT16 分区, 是 DOS 系统, 突然死机后不能启动。用 DOS 软盘启动, 看不到 C 盘和 D 盘, 需要恢复数据。

检查:

① 该硬盘 0 × 0 × 1 的主分区表被病毒破坏。

② 硬盘 0 × 1 × 1 的逻辑 0 区 I/O 参数表完整, 从该参数表中可看出, 硬盘磁头数为 0CH (即 12), 每道扇区数为 23H (即 35), C 盘分区大小为 F5F5H (即 62965), 而 $(62965 + 35) \div (12 \times 35) = 150$, 说明 C 盘占磁道为 0 至 149 磁道, 后面的分区从 150 × 0 × 1 开始。

③ 检查硬盘 150 × 0 × 1, 有 D 盘的扩展分区表, 如下所示:

```
00 01 01 96    06 0B E3 D9
23 00 00 00    6D 5B 05 00
```

因此扩展分区表分区大小为 351120 (055B6DH + 23H = 055B90H = 351120), $351120 \div (12 \times 35) = 836$, D 盘扩展分区占磁道为 150 至 985 磁道。

**解决方法:**

① 把任意一个 FAT16 分区的硬盘 $0 \times 0 \times 1$ 的主引导扇区拷贝覆盖到本硬盘的 $0 \times 0 \times 1$ 扇区中。

② 按照以上对 C 盘和 D 盘参数的分析, 仿造并修改硬盘 $0 \times 0 \times 1$ 扇区偏移量为 01BE 处的主分区表, 如下所示:

```
80 01 01 00    06 0B 23 95
23 00 00 00    F5 F5 00 00
00 00 01 96    05 0B E3 D9
18 F6 00 00    90 5B 05 00
```

注意, C 盘隐含区数 23H 加上 C 盘大小 F5F5H, 等于 D 盘扩展分区隐含区数 F618H。

③ 重启该硬盘, 硬盘正常启动, C 盘和 D 盘数据完整无损。

例3

40 G 硬盘 ($5005 \times 255 \times 63$) 分 3 个 FAT32 分区, 是 Windows 98 系统, 重要数据放在 D 盘和 E 盘。因系统不稳定, 请电脑公司重装系统, 由于经办人粗心大意, 把硬盘原有分区都删除后再重建 3 个分区并格式化, 然后再把 C 盘装上系统, 但 C 盘 Windows 98 系统启动后, 不见了原来 D 盘和 E 盘的所有数据。

检查:

① 该硬盘 $0 \times 0 \times 1$ 扇区的主分区表参数是新划分的, 如下所示:

```
80 01 01 00    0C FE FF FF
3F 00 00 00    00 C5 FA 00
00 00 C1 FF    0F FE FF FF
3F C5 FA 00    4D DF CF 03
```

从该分区表看出, 该硬盘磁头数 255, 每道扇区 63, D 盘扩展分区定位在 FAC53FH (即 16434495), $16434495 \div (255 \times 63) = 1023$, 即 D 盘扩展分区定位在 $1023 \times 0 \times 1$ 扇区。

② $1023 \times 0 \times 1$ 扇区有 D 盘扩展分区表, 如下所示:

```
00 01 C1 FF    0C FE FF FF
3F 00 00 00    BB AC 57 01
00 00 C1 FF    05 FE FF FF
FA AC 57 01    53 32 78 02
```

从该扩展分区表看出, E 盘扩展分区离 D 盘扩展分区距离 0157ACFAH (即 22523130), $22523130 \div (255 \times 63) = 1402$, 即 D 盘占 1402 磁道, $1023 +$



1402 = 2425, 即 E 盘扩展分区表在 2425 × 0 × 1 扇区。

③ 1023 × 1 × 1 扇区有完整的 D 盘逻辑 0 区, 但 FAT 表为空, 是新建分区。

④ 2425 × 0 × 1 扇区有 E 盘扩展分区表, 如下所示:

```
00 01 C1 FF    0C FE FF FF
3F 00 00 00    14 32 78 00
```

⑤ 2425 × 1 × 1 扇区, 有完整的 E 盘逻辑 0 区, 但 FAT 表为空, 是新建分区。

⑥ 在 40 G 硬盘中, 从 1 × 0 × 1 扇区开始查找分区标志“55AA”, 每次跳跃 16065 扇区, 只找到 1023 × 0 × 1 和 2425 × 0 × 1 扇区。

⑦ 在 40 G 硬盘中, 从 1 × 1 × 1 扇区开始查找分区标志“55AA”, 每次跳跃 16065 扇区, 只找到 1023 × 1 × 1 和 2425 × 1 × 1 扇区。

⑧ 在 40 G 硬盘中, 从 1 × 0 × 1 扇区开始查找分区标志“55AA”, 每次跳跃 1 扇区, 另外找到了 1254 × 151 × 1 和 3058 × 211 × 1 扇区有逻辑分区标志。

⑨ 1254 × 151 × 1 扇区逻辑 0 区的 I/O 参数表表明, 原来硬盘 D 盘磁头数为 FOH (即 240), 每道扇区数为 3FH (即 63), 隐含区数为 01333A8FH, 大小为 01BA469AH。

⑩ 1254 × 151 × 33 扇区开始的 FAT 链表表明 D 盘有大量数据。

⑪ 3058 × 211 × 1 扇区逻辑 0 区的 I/O 参数表表明, 原来硬盘 E 盘磁头数为 FOH (即 240), 每道扇区数为 3FH (即 63), 隐含区数为 02EDD15FH, 大小为 01DD1D01H。

⑫ 3058 × 211 × 1 扇区开始的 FAT 链表表明 E 盘有大量数据。

⑬ 原来 D 盘和 E 盘的 I/O 参数表表明, 40 G 硬盘在原来主板 CMOS 设置不是用 LBA 模式 (5005 × 255 × 63), 而是采用 LARGE 模式 (5318 × 240 × 63)。

解决方法:

① 重设置主板 CMOS 对 40 G 硬盘的参数, 由 LBA 模式 (5005 × 255 × 63), 改为 LARGE 模式 (5318 × 240 × 63)。

② 在 LARGE 模式下, LBA 下的 1254 × 151 × 1 扇区变为 LARGE 的 1331 × 1 × 1 扇区, LBA 下的 3058 × 211 × 1 扇区变为 LARGE 的 3250 × 1 × 1 扇区。

③ 由于用户只需要 D 盘和 E 盘的数据, 故可根据 D 盘和 E 盘逻辑 0 区的 I/O 参数表提供的信息, 在主分区表中仿造两个独立分区出来, 并修改硬盘 0 × 0 × 1 扇区偏移量为 01BEH 的主分区表参数为如下所示:

```
00 01 C1 FF    0C EF FF FF
8F 3A 33 01    91 46 BA 01
```