improved over the last year, and I think anyone in a position to compare would agree that it has, it is through her careful analysis of the problems that are phoned in, and through her constant review of every detail of the documentation and directory organization on the Unix TEX tapes.

Typesetting on Personal Computers

Recovering from a Hard-Disk Failure

Mitch Pfeffer and Alan Hoenig

I'm sure it's a corollary of Murphy's Law: The most precious part of your computer — the hard disk — is the part most prone to failure. Having gone through more that one Seagate hard disk in the past year, I decided to devise a strategy that would minimize my down time in the event of another hard-disk failure.

I realized the importance of backing up onto two different media (one of which should be removable) when a client lived through the following nightmare: He had been backing up his hard disk to floppies — but what he didn't know was that his floppy-disk drive was drifting out of alignment; immediately after writing a floppy, he could still read it back in on that same drive, so he suspected nothing. When his hard disk failed, and he tried to restore his system from the floppies, he found that his floppy-disk drive had now drifted still further out of alignment: not only couldn't he read his own floppies, but, because they were written with the heads out of alignment, nobody else could read them either. (Incidentally, it was a Priam hard disk that failed. Although Priams are considered to be highly-reliable drives, I've noticed that they fail in dusty environments.)

In addition to backing up to floppies, my solution is to use a pair of inexpensive ($250) hard disks in a system, and to copy just those files that have changed from my working hard disk to the backup hard disk every day, using DOS's `xcopy`. This only takes a few moments, and requires no fiddling with floppies. With this approach, all I need do to get back in operation if my main hard disk blows, is to shift two cables. This gives me an important advantage over a tape backup: If you use a tape backup and your hard disk blows, you can't run your system off the tape backup — you must first replace the drive, and then restore the contents of the tape to the new drive. Besides — at $250, the hard disk is cheaper than a tape drive, as well as faster and more convenient.

(Prices given in this article are dealer prices, which are often identical to mail-order prices.)

The drives I use are the Miniscribe 8438F: These are 30 Mb half-height RLL drives, with a moderately-fast 40 ms access time. (The type of work I do — programming, TEX'ing, and writing — doesn't benefit from a faster access time. TEX turned in the identical performance with this drive as with a $900 28 ms 60 Mb Priam.) I've been installing a pair of these drives in all the systems I've sold over the past several months, and not one has failed.

(I recently came across a different Miniscribe drive that looks even more attractive: the 3675. This is a 63 Mb, $275 drive, which has a 42 ms access time when formatted as two 30 Mb partitions (its normal access time is 61 ms). I hope to test this drive in future systems.)

To get the full 30 Mb out of the 8438F drive, you must use an RLL controller; normally, computers come with MFM controllers. RLL drives transfer information 50% faster than an MFM drive. I'm using Adaptec 2372A controller ($160); it features a 1-to-1 interleave, which means that an entire track can be read during a single rotation of the disk. The controller supports two $5\frac{1}{4}''$ floppy-disk drives, in addition to the two hard disks.

I also tested two other RLL controllers: the DTC-5287, and the Western Digital 1003-RA2.

The DTC controller performed well, but lacks the 1:1 interleave feature of the Adaptec; however, the DTC controller is rated for running in a system with a 12 MHz, 1 wait-state bus, while the Adaptec is rated for an 8 MHz, 1 wait-state bus. (The Adaptec rating is conservative: I've had no trouble running it in a 10 MHz, 0 wait-state system.) By the time you read this, DTC should be shipping their 7287 controller, which does support 1:1 interleaving.

My experience with the Western Digital RLL controller was dismal: The first two Western Digital controllers I received proved defective. When I finally got one that worked, I found that it took three times longer to read in files than the Adaptec.

Hardware Installation: The drives are shipped with their address-jumpers set to the lowest address position; this setting is for XT-class machines. For AT-class systems, you must move the jumper to the next-to-the-lowest position on both drives. The terminating resistor — a thin, colored, multi-pronged strip near the address-jumper — is removed from the drive that will be attached to the plug in the middle of wide cable. This drive will become the 'D:' drive.

Should the 'C:' drive (the drive at the end of the cable) blow, and you need to turn your 'D:' drive into your 'C:' drive, you'll need to insert the terminator resistor into the 'D:' drive, and move the two plugs from the 'C:' drive to 'D:' drive.

(To avoid having my clients insert the terminator into the 'D:' drive when the 'C:' drive blows, I'm considering creating a non-standard hard-disk control cable: it would be twisted going into the middle connector, and twisted again going into the connector at the end of the cable, effectively undoing the effect of the first twist. As always, the drive at the end of the cable is the only drive with the terminator; however, the drive at the end of the cable will now be the 'D:' drive. Should the 'C:' drive (attached to the connector in the middle of the cable) blow, the client need only move the plugs from the 'C:' drive to what was the 'D:' drive — the terminator is already in place.)

When plugging the power connectors into the drives, the connectors on the wires may ride up on the pins when you push the plug down into place. This would cause intermittent problems with the drives. To avoid this, I'd recommend that before you attempt to plug the connector to the drive, you first push each of the four leads down into the plug, and hold onto both the plug and the wires when pushing the plug into place. Support the printed-circuit board by placing your finger below the printed-circuit board when pushing the plug down. Once the plug is firmly in place, take a needle-nose pliers, or a hemostat, and push each of the four leads down onto the pins on the drive.

Software Installation: The Adaptec contains a built-in low-level format program, which is activated by typing 'g=c800:5' to the DOS debug program. As usual, this is followed by running fdisk on each drive. (For some reason, I found I had to do the low-level format twice on the 'D:' drive, before the fdisk would work properly.) Finally, do a 'format c:/s/v' and a 'format d:/s/v', to high-level format each drive; this also makes both drives bootable (you'll want to be able to boot off your 'D:' drive, should the 'C:' drive blow).

Operation: After installing all your software on the 'C:' drive, copy the entire contents of the 'C:' drive to the 'D:' drive by running the DOS 3.3 command 'xcopy c:\. d:\ /s/e/v'.

At the end of each day, you can copy just those files whose archive bit is 'on' (i.e., those files that have not yet been backed up to floppy — see below), by typing 'xcopy c:\. d:\ /a/s/e/v'. Or, to avoid copying unnecessary files, such as TEX's dvi and log files, you can run this batch file:

```
attrib -a c:\*.dvi /s
attrib -a c:\*.log /s
attrib -a c:\park!@#.cor /s
attrib -a c:\*.qex /s
attrib -a c:\*.qeb /s
xcopy c:\. d:\ /a/e/s/v
```

This clears the archive attribute of the superfluous files throughout the hard disk before doing the xcopy; the '/a' option tells xcopy to copy only those files whose archive bit is 'on'. (Microspell creates file with the qex extension after processing a tex file, and a file with a qeb extension results when you check a WEB file; park!@#.cor is the temporary file created by the Cordata driver.)

Note that if you delete a file from your 'C:' drive, its backup copy will still remain on the 'D:' drive; this would eventually lead to the 'D:' drive filling up with copies of deleted files. To solve this problem, I'm planning to toss together a program that deletes those files on 'D:' no longer found on 'C:', and invoke it at the start of the above batch file.

Fifth Generation Systems, makers of the Fastback backup program, just introduced a hardware card that automatically mirrors the two drives: any file that's saved or deleted from your working drive is automatically saved or deleted from the backup drive. If your working drive should fail, this card (call Counterpart) will sound an alarm and instantly switch you over to your back-up disk. This arrangement can be essential in many environments, such as on-line order entry. Fifth Generation supplied me with an evaluation unit, and I hope to have a report on this card in a future issue.

Note that the Miniscribe 8438F and the 3675 both lack an auto head-park feature, so before moving your system, be sure to park the heads. I use the park program included in the PC-Tools utility package. I found that the park program that comes with Disk Manager did not work with

the Adaptec, although it did work with the DTC controller.

Backing up to a removable device

The other component of my backup regime consists of backing up to floppies, using Fastback-Plus. Every month, I do a full backup of my entire hard disk to floppies; in doing so, Fastback turns off the archive bit of all the files. Every other day (or so), I do a differential backup, where Fastback copies just those files that have their archive bits 'on' — that is, just those files that have been changed or added since the last full backup; these are the same files that are copied to the `D:` drive by the batch file given above.

Fastback-Plus does have its problems: version 1.00 was unable to restore two out of the fifteen floppies it had created in backing up my hard disk (I did the backup with Fastback's read-after-writing verification turned on — see below). I suspect that the problem was caused by an imperfection in the way version 1.00 formatted new disks as it backed up. (If there's one program that had better be flawless, it's your backup program.)

Fastback also had trouble restoring files it had placed on a Bernoulli cartridge. If you have a Bernoulli box, I suggest you partition your hard disk into 20 Mb partitions, and backup the hard disk by `xcopy`'ing each partition to individual cartridges. This has the added advantage of not having to de-Fastback the files in order to use them.

I also noticed that if I formatted 360 K diskettes in my 1.2 Mb drive after running Fastback's installation routine, an inordinately high number of bad sectors were unjustly locked out. This problem went away after I reset the computer.

One of the nicer features of Fastback is that it allows you to exclude file and directories from the backup. I've set up my copy so that it excludes: the operating system kernel (`\command.com`, `\ibmbio.com`, and `\ibmdos.com` under PC-DOS); all the DOS programs (kept in `\DOS` on my system); and `*.dvi`, `*.log`, `*.qeb`, `*.qex`, and `park!@#.cor`. (Should my hard disk fail, I'll need to restore DOS and Fastback from their distribution diskettes anyway, in order to run Fastback to restore the other files.)

I run Fastback with 'write-verify' on, 'compression' set to 'save disks', and 'error correction' on. On my 386 system, Fastback takes about a minute per megabyte with these settings. The 'write-verify' option sounds like it offers more security than it really does: Fastback does not try to read back the information it wrote out to the floppy — all it does is compare what it read off your hard disk with the copy of that information it has in RAM.

If you use a hard-disk cache, make sure to turn it off before running a backup program — it will defeat the verification attempted by the program. (Personally, I don't bother when doing a differential backup, but before doing a full backup, I replace my `autoexec.bat` and `config.sys` files with the simplest possible versions and re-boot, to avoid any detrimental interactions that might occur between the backup program and, say, a resident program.)

For greater security, I alternate between two sets of floppies for both the full backups and the differential backups. To keep the differential sets straight, I move a Post-It marked 'Use Next' between the sets. Before doing a full backup, I put the last differential at the back of the box containing the most recent full backup, and then over-write the box containing the oldest full backup. This system not only allows easy recovery, but also allows me to dig up an early copy of a file if I find out that I've accidentally trashed a file, and backed up the trashed file.

The current version of Fastback-Plus is Version 2. It includes a separate verify feature: after the entire backup is complete, you run this option, and re-insert every diskette; Fastback will compare each file on the hard disk to the copy on your floppies (unfortunately, this still doesn't guard against an out-of-alignment floppy-disk drive). It also can be set to automatically delete unwanted history files.

Evaluation of K-Talk

C.G. van der Laan and J.R. Luyten
Rijksuniversiteit Groningen

We would like to announce the availability of a report entitled "Evaluation of K-Talk", RC-report 22, Groningen, 1988. Further information can be obtained from the authors. The Foreword of the report is reproduced below.

At the Rijksuniversiteit Groningen document preparation is done by text processors and document preparation systems 'at the desk', with possibly remote 'execution' and printing.

At the moment WordPerfect and TEX as representation of respectively text processors and document preparation systems enjoy the highest 'support category' — they are standards for the time being.