

Site Reports

A UK-Based T_EX Mail Archive Server

Peter Abbott
Aston University UK

The computing facilities of the UK academic community are interlinked by JANET (Joint Academic NETWORK), which is a private network based on X.25 protocols and which is administered on behalf of the community by the JNT (Joint Network Team). JANET links Universities, Polytechnics, Further Education centres, major research facilities and other related bodies. Gateways exist at various places on the network to give access to both public and private networks including PSS, EARN, Bitnet, Internet and UUCP, to name but a few.

Systems connected to JANET run the CBS (Colour Book Software), which provides a common set of facilities across a wide range of disparate computer systems. The major components of CBS are File Transfer (Blue Book), Interactive access (Green Book), Electronic mail (Grey Book) and to a lesser extent Job Transfer (Red Book). At Aston we hold an archive of T_EX-related material for the benefit of our community and any other site that can gain access via the gateways. The archive currently (August 1988) contains approximately 200mbytes of data covering mainframes, minis, workstations and PCs with versions of T_EX and L^AT_EX for all these areas. The archive started life in the autumn of 1987, and since that date contributions have been received from wide and far, reflecting its value to the community.

Access from JANET sites to the archive is relatively simple and painless. The FTP facilities that are provided can be illustrated by a simple example. (I shall use the VAX/VMS notation but there are corresponding formats for UNIX, VM/CMS, NOS/VE, etc.). A user called `orinocco` is registered on a system with the name `uk.ac.wimbledon.common`. To extract files from the archive, `orinocco` signs on to his system and types the command `transfer`. The required parameters are input filename, output filename, remote username, remote username password. If we assume `orinocco` wishes to fetch the file `[public]000aston.readme` the sequence is

```
transfer
%_Input filename?
    uk.ac.aston.spock::[public]000aston.readme
%_Output filename? archive.list
%_Remote username? public
%_Remote username password? public
```

Transfer nnnnn has been queued.

Sometime later the file is available on the system at `uk.ac.wimbledon.common`. Failure to find or transfer the file and other error messages are notified via the normal VAX/VMS mail system.

Regrettably the FTP system is not totally machine-independent, and the notorious VAX/VMS file format `stream_lf` creates a number of difficulties for sites which are not running the VAX/VMS operating system. Similarly, the commercial world and many sites beyond the gateways do not have FTP software (and do not wish to implement it either).

Aston does offer a magnetic tape service to compensate for these problems but it is slow and time-consuming for all concerned, and unproductive when one considers the world-wide electronic network which is already in place.

This is not a new problem, and at Rochester a mail server was implemented to provide access to the L^AT_EX Style collection. [Editor's note: The L^AT_EX Style collection has moved to Clarkson, see page 294.] A similar mail service at Aston would open up the UK archive to a much wider community. I am pleased to say that such a mail service has been running on an experimental basis for some weeks now, and although not yet totally bug free does provide a useful service. Credit must be given to both Graham Toal, who put a mail server in place at Edinburgh for a limited period, and to Adrian Clarke, who is still developing the one that now runs at Aston. (Graham no longer recommends his server and refers all queries to the Aston system). Adrian is at the University of Essex, but undertakes the maintenance and development of the mail server (for which I continue to be grateful).

The Aston mail server is a batch job which runs on a VAX 8650 processor under the VAX/VMS operating system; eventually it will not be monitored, so errors will simply be filed in the normal 'black hole'. At this stage of the development cycle, sometimes even genuine mail succumbs, so if no reply is received after a suitable period you are recommended to try again. Log records are kept and common errors will be reported from time to time in UKT_EX (the UK's equivalent to T_EXhax). It is impossible to give estimates of the turnaround time

for any individual user; the server runs once per hour and the mail message are queued for transmission. The mail software makes a maximum of 30 attempts to send a message (10 at 10 minute intervals, 10 at 1 hourly intervals and 10 at 4 hourly intervals). This rather extended period is designed to overcome short-term network failures and for systems which are switched off for short periods of time or overnight. The cluster system at Aston is normally available 24 hours a day, seven days a week, with the occasional booked systems maintenance on a Wednesday morning and twice yearly maintenance checks by DEC.

Instructions on how to extract files from the archive are contained in a help file; this file is available by sending a mail message to

`texserver@uk.ac.aston.spock`

UK addresses on JANET are big-endian format and most users 'on the other side of a gateway' will need to specify it as `texserver@spock.aston.ac.uk`. The subject line in the incoming mail message is ignored, as is any text until a line starting with --- (three minus or hyphen characters in columns 1 to 3); any text on that line is also ignored. The next line is the `name@return address` in UK format and the third line is the word `help` (in UPPER, lower or MiXeD case). For example:

```
--- (any text on this line is ignored)
name@address
help
```

The best rule to observe in quoting `name@address` is to use the format:

- JANET sites
 `name@uk.ac.site.system`
- Sites via earn-relay (Internet, Earn)
 `name%little-endian%big-endian@earn-relay`
- Sites via uk.ac.ukc (UUCP)
 `name%little-endian%big-endian@uk.ac.ukc`

Anyone who has problems getting mail back is welcome to send me (`abbott@uk.ac.aston`) the message that they have tried; I will forward it to

`texserver@uk.ac.aston.spock`

with a copy to the originator showing the `name@address` format that is required. I do not guarantee to be able to solve every query but will do my best.

Atari ST Site Report

Klaus Guntermann
Technische Hochschule Darmstadt

Since our last report several changes have occurred. First we must say that the former distributor Kettler broke down in April 1988 (probably not because of their activities with T_EX for the Atari ST). In the following months we had negotiations with several companies and since August 1988 there is a new distributor.

Furthermore the product has been updated a lot, the documentation has been extended (e.g. a local guide for L^AT_EX is included now), and here we summarize the most important changes:

A new version of T_EX is in distribution (currently 2.92). Now it supports search paths for input files, font files (TFM) and preloaded FMT files. The memory management has been redesigned to use all available memory (up to 65534 mem array elements).

The new preview driver for the monochrome monitor has the following enhancements:

- it is faster,
- it allows a two step reduction and back on the fly without disk access (for 4 : 1, 9 : 1 compression),
- it supports graphics inclusion for bitmap raster files,
 - the format of the files is simple and described in the documentation,
 - a conversion program for DEGAS pictures is included,
- the font search is customizable.

The new laser printer drivers (HP LaserJet+/Series II or Kyocera) support graphics inclusion in both device dependent and "preview compatible" format (i.e. bitmap raster file).

The new integrating menu shell comes with the following features:

- it allows to select work file for all activities,
- it can call an editor, T_EX, INIT_EX, the previewer, a printer driver or BIBT_EX from pull down menu items, function keys, or soft keys on the screen,
- it is customizable
 - to call any editor,
 - to select the FMT file,
 - to specify paths for input files, TFM files and FMT files,
 - to predefine parameters and the initial working directory,

- the customized values may be saved and loaded, e.g. to switch between Plain and L^AT_EX or between different printer drivers.

An installation program for hard disk based versions is included.

New dot matrix drivers that support the “preview compatible” graphics inclusion are in preparation. These will run with less than 700 KB RAM available, even for a resolution of 360 dpi.

The new distributor will handle new requests as well as updates for former Kettler customers. Please direct inquiries for further information about S^T_EX to

T_EXsys
Kranichweg 1, D-6074 Rödermark
Federal Republic of Germany
phone +49 6074 1617

Data General Site Report

Bart Childs

The distribution is stable. Most of the news is that out activities have been in making sure that we have the latest revision of all pieces of the system. We think we do.

My ineptitude with E-mail caused the printer charts to not be correct in the last issue. I hope this time I have successfully gotten the current information to Don Hosek.

The *environment* that gives a menu driven system has been rewritten from CLI macros into a WEB. This is much faster. We are planning on writing change files to make it available for UNIX and VMS. It has already been done for MS-DOS. This will help new users to begin using T_EX quickly because all they have to do is remember `TeXe filename` to start it up, and select a number to edit, T_EX, preview, or print. The menu is somewhat longer, because it also interfaces to several utilities and allows selection of other T_EX files (and attendant macro packages), editor, printer, and switches.

There has been a lot of interest in the 64-bit T_EX and I have been sending out the relevant (non-DG) changes frequently.

IBM VM/CMS Site Report

Dean Guenther
Washington State University

There are several changes and a few newcomers to the IBM VM/CMS distribution tape.

Thanks to Barbara Beeton, the most recent L^AT_EX (April 1988), ~~A~~M^S-T_EX (version 1.1d), and Plain T_EX (version 2.92) are now on the tape. Barbara also sent along the bug fixes for T_EX and METAFONT, so I now have those two updated to versions 2.93 and 1.5 respectively.

Georg Bayer has updated his DVI3279 preview program. It no longer prints out its messages in German. Many thanks Georg!!

Eric Berg sent me the B_IB_TE_X .99 files. After quite a bit of work with Oren Patashnik, I finally got .99 up and running. It and its updated auxiliary files are all included.

Don Hosek has contributed several newcomers for the distribution tape. He has supplied a working version for PXTOPK, GFREAD, PKTYPE, and MFT. Besides those four, Don also modified GFTODVI to create the standard 1K blocked DVI instead of a 2K file; and he changed GFTOPXL so that it conforms to the standard 128 character convention for PXL version 1001 files.

I also updated Weave to version 2.9 (I can't remember who supplied the update) and I changed the default output filetype for GFTOPK from “GF” to “300GF”. Oh yes, I also modified DVI2LIST to quit giving a disconcerting nonzero return code when it was including a page segment directly into IBM's Advanced Function Printing Data Stream (AFP/DS).

Chris Carruthers at the University of Ottawa sent me the Makeindex program. Developed at the University of Berkeley by Peehong Chen, it has been modified by Chris to run on Waterloo C on CMS. Chris has included a module for all those who would like to use Makeindex on CMS, but do not have Waterloo C.

Shashi Sathaye at the University of Kentucky has taken Nelson Beebe's drivers and added the code so that they will compile under Waterloo C. Mike Glendinning from the University of Manchester then took Shashi's mods and was able to get the Beebe drivers to compile under IBM C. I suspect both of these are now available through Nelson. Wayne Podaima, of the National Research Council in Canada, sent the P_IC_TE_X macros, which I've also included on the tape.

MVS Site Report

Craig Platt
University of Manitoba

At the end of July, I sent the latest MVS T_EX tape to Maria Code for distribution. It contains the following enhancements from the June, 1987 tape.

- T_EX version 2.9
- METAFONT version 1.3
- BIBT_EX version 0.99c
- GFtoPK, PKtype and MFT included.
- Dynamic file allocation.

The last item represents the biggest change, and refers to the way T_EX file names are mapped to OS dataset names. MVS T_EX users have always been hampered by the “DD name bottleneck”, whereby the names of all files input or output by a job must be known prior to execution and pre-assigned to an 8-character DD name. This makes packages such as L^AT_EX, that read and write lots of “auxiliary” files, cumbersome to use. The new release uses assembler routines developed by Richard Tilley and others at the University of Manitoba to allow dynamic (run-time) access to OS dataset names. Here is an example of how it works.

Suppose T_EX sees the command `\input story` (or `\input story.tex`). Then several attempts are made to match this to the intended dataset. The first is to find a DD name, in this case, `STORY`. However, if the extension part of the name were other than `.tex`, then a combination name, formed from 5 characters of the “first name” and 3 characters of the extension would be used, as in the previous version of MVS T_EX. For example `\input primes.contents` would be associated with DD name `PRIMECON`.

If no matching DD name is found, then T_EX can search directly for a catalogued dataset with the name `<prefix>.STORY.TEX`, where `<prefix>` is a string supplied by the user in the `PARM` field of the program invocation. It will often be a user’s logon id, but could be any legal OS dataset name prefix.

If this dataset is found, it is used. (Output files will be overwritten.) In case of a non-existent output file, a new file will be created, provided the user also supplies the name of a default output volume to put it on.

For input files, there is also a “library” option, which will look for a DD name of the form `<ext>LIB`, where `<ext>` comes from the first 5 characters of the extension part of the name. If found, this should point to a partitioned dataset which will be searched for a member with the “first name”. In

the case of `story.tex`, this means finding member `STORY` in the dataset allocated to DD name `TEXTLIB`.

Another construction allows specification of a fully qualified OS dataset name in the T_EX source. If a file name begins with a “sharp” character, ‘#’, then the rest of the name is used without modification as a dataset name. For example, `\input #PLATT.STORY.TEX` will read the dataset `PLATT.STORY.TEX` regardless of what DD statements are supplied. This is useful if a user wants access to datasets with a different `<prefix>`, such as system supported macro libraries.

For partitioned datasets, there is yet another syntax. If the file name has the form `aaa:bbb`, then `aaa` specifies a partitioned dataset and `bbb` the member name. For example, to include a macro from a system library, a user might say

```
\input #sys1.macros.tex:today
```

which would correspond to the construction

```
DSN=SYS1.MACROS.TEX(TODAY)
```

in a DD statement.

I’ve used a similar scheme for METAFONT and BIBT_EX, but the other programs in the package (e.g., T_EXware, MFware) use just DD names, since (in most cases) they input or output only a fixed set of files.

So much for the good news. After sending off the tape, I noticed a small problem in the design parameters, which could affect L^AT_EX users (sigh). When a text file is opened, e.g., by an `\openout` command, and no corresponding dataset exists, T_EX creates a file with the default DCB parameters of `RECFM=VB LRECL=256 BLKSIZE=6144`. These values seemed reasonable, but I later discovered that it is quite easy to exceed the 256 character record length in a L^AT_EX auxiliary file. For example, the command `\subsection{Running \TeX, \LaTeX and \AmSTeX.}` expands to a line of 446 characters in the `.aux` file! The result is that the expanded line gets split arbitrarily, usually in the middle of a control sequence, and you get a PASCAL/VS error message: “AMPX059E Text exceeds logical record length in file ...” in the `SYSPRINT` file.

I will try to fix this in the next release (perhaps even “by the time you read this”), by increasing the `LRECL` value to 512. In the meantime, for anyone who has already received the July tape, there are a couple of workarounds. One is to pre-allocate L^AT_EX auxiliary files for each job before running T_EX, using larger DCB values, avoiding the dynamic file creation. It is also possible to keep output lines short by judicious use of `\protect` to prevent

expansion of macros like `\TeX`. Finally, there is a quick fix that an installer can try. The assembler routines that perform the dynamic allocation are compiled separately into a PASCAL/VS segment called `FILPROCS`. There is a copy of the compiled segment in object format included on the tape as file number 36. It is possible to edit this file, changing the embedded string `"LRECL=256"` to `"LRECL=512"`. The resulting object file can then be re-linked into the `TeX` load module with the IBM linkage editor (you need to `REPLACE` the csects `GETMVSPA` and `GETDDN` in the `TeX` module). This is admittedly a sketchy description, but if anyone needs help with it, I can send them detailed instructions.

In addition to enlarging `LRECL`, the next release should contain `TeX 2.93` and `METAFONT 1.5` (still awaiting `trip/trap` testing), but for future versions, I would like to try making the file parameters adjustable by the installer. For sites with PASCAL/VS, the `WEB` source can of course be edited and re-compiled, but this is not a solution for the many sites that don't. One suggestion is to provide a customization module in the form of an assembler subroutine. All MVS sites should then be able to edit and compile it, and re-link it into the `TeX` load module.

UnixTeX Site Report

Pierre A. MacKay

Since the last UnixTeX site report in January, 1988, many of the hopeful promises have been fulfilled. What was there called `TeX-to-C` has been renamed to the more comprehensive `WEB-to-C`, and compilation under this system is now the default. `TeX`, `METAFONT`, `TeXware` and `BIBTeX` are all supplied with `WEB-to-C` change files, and a good start has been made on `MFware`. The two conversion programs `gftopk` and `pktogf` are already done, as well as `mft`, `gftype` and `pktype`.

A few notes are in order on the way in which `MFware` is being approached. The basic `WEBs` are by this time perfectly stable, and since I am doing the translation myself, I have taken the liberty of making the changed programs more Unix-like and less reminiscent of `TOPS20` or `SAIL`. Wherever possible, Unix command-line switches are used in place of the old `dialog` lines, and simple utilities are made to run silently by default. There is only one

file of "extra" routines, `mfwarext.c`, which contains about the same lot of code as that used in `TeXware`. The `test_access` procedure is used to look for an input path in the appropriate environment variable in all cases, and the output file, if its name is automatically generated as in `gftopk`, is always deposited in the current working directory. The rather insignificant lot of output from `gftopk` and `pktogf` can be turned on by means of the command line switch `-v`; otherwise, these two programs run with no output to the screen at all. The `gftype` dialog has been replaced by the two command line switches `-m` for mnemonics, and `-i` for pixel image. The default is to produce neither of these voluminous outputs. The form of the `gftype` command for running the `trap` test is:

```
gftype -m -i trap.72270gf
```

It may be noted that the four programs so far discussed could be converted into true Unix style, by diverting "chatty" output to `stderr` and using `stdin` and `stdout` for the `gf` and `pk` files as appropriate. This has a sort of purist appeal, and would make it possible to run these programs in a pipe. I have been unable, however, to think of any scenario in which that would be useful, and it would eliminate the convenience of having the output file from the conversion programs supplied automatically with the desired filename. It is, of course, possible to force a non-standard output filename, the command line syntax for `gftopk` is:

```
gftopk [-v] gffile [pkfile]
```

The serious omission from all of this is `gftodvi`. If necessary, I will try to supply a change file for this program, but since Donald Knuth has announced his intention of rewriting the `WEB`, I would prefer not to spend much time on it.

In keeping with the attempt to suppress the use of `pxl` format, no attempt has been made, nor will any be made, to adapt `pxl`-related programs to `WEB-to-C`. The effort is better spent on making old `pxl`-reading programs read `gf` or `pk` format instead. Any successes in this line will be gratefully received and incorporated into the distribution. Remember, if you undertake to work on this problem, that `gf` fonts of 255 characters are becoming quite common.

Finally, a rather more serious confession about the change files for `gftopk` and `pktogf`. The `WEBs` for both programs supply a *preamble comment* to replace the dated `METAFONT` comment in the original `gf` file. I have found it very useful to know the date of creation of any font, and am unwilling to lose this information, since I doubt that I am

alone in finding it valuable. I have therefore added to the change file some bits of code which insure that the original METAFONT comment, including the date and time, is passed through unchanged in place of the undated `gftopk` or `pktogf` comment.

The current versions of the principal programs on the distribution are `TEX 2.93` (if you got the earliest copy of this by FTP from `SCORE.STANFORD.EDU`, get it again), `METAFONT 1.5` (same caution) and `BibTEX 0.99c`. At the time of writing, `WEB-to-C` was at version 2.22. Tim Morgan's list of successes in the `README` for version 2.20 was (omitting the notes of detail):

- Sun-3, SunOS 3.2, SunOS 3.4, 3.5, 4.0FCS
- Sun-2, SunOS 3.2-4
- Sun-4, SunOS 3.2-4, and SunOS 4.0FCS
- Sequent Balance, Dynix 2.1.1
- VAXen running 4.2, 4.3BSD, and Ultrix
- Convex
- Amdahl running UTS
- Apollo, SR9.7 and SR10.0 (beta)
- Ridge 32 running ROS 3.5 and C compiler version 2.1B
- UNIXpc (aka 3b1 or PC7300) running System V version 3.51
- MIPS R/1000, compiler version 1.21
- Iris workstation
- Celerity C1260, UNIX version 3.4.78
- Pyramid 98x, running OSx64Q 4.0-870901, C Compiler CCOMP-4.0

To this list may be added `MassComp`, `Tahoe`, `ELXSI`, the Sun 386i and the Cray running `Unicos`. On most of these systems it really is possible to do the minimal editing of a file called `site.h`, and then type `make all`. I even managed, with a certain amount of arbitrary hacking, to get a full compilation on a VAX 11/780 running 4.1BSD.

One of the great advantages of `WEB-to-C` compilation is the ease with which special versions can be made up, both of altered and enhanced versions and of enlarged "gargantuan" `TEX`. A `TEX` with 200,000 "half-words" of box and general storage, with space for 9,500 macro names, and with other limits similarly expanded is available through the use of the `BIGTEX.PATCH` in the distribution. I have even tried a compilation with 1,000,000 "half-words" of general purpose memory, but that produced a 9-Megabyte core image, and was felt to be unneighborly. Big versions of `TEX` are genuine `TEX`, and are so convenient, especially for those who make heavy use of things like `pictex`, that we strongly advise everyone who can to switch to one.

There has been a good deal of mail about new versions of the `undump` program for preloaded `TEX`. I shall not repeat Ken Yap's analysis here, except

to say that on most systems the C compilation of "gargantuan" `TEX` runs marginally faster than small `TEX`, for various reasons, and loads `fmt` files so fast that there is really little reason to use `undump` any longer. Here are the essential lines of a Bourne shell script that we use instead:

```

case ${3+toomany} in toomany )
  echo "Too many arguments!"
  echo "Usage: tex foo[.tex] [my[.fmt]]"
  echo "or latex foo[.tex]"
  echo "or slitex foo[.tex]"
  exit 1;;
esac
case $0 in
  */tex ) virtex '&${2-plain}' $1; exit;;
  tex ) virtex '&${2-plain}' $1; exit;;
  */latex ) virtex '&lplain' $1; exit;;
  latex ) virtex '&lplain' $1; exit;;
  */slitex ) virtex '&splain' $1; exit;;
  slitex ) virtex '&splain' $1; exit;;
esac

```

This script resides somewhere in the path for executable binaries, and is linked to the three names `tex`, `latex` and `slitex`. Notice that this approach gives you the opportunity to load your own private `fmt` file when you invoke the script under the name `tex`.

The modified program `TEX-XET`, for setting in two directions, can be compiled using `cxet.patch` and a first try at Antti Louko's `multihyphen-TEX` (Don Knuth has suggested "MulT_EX" as a name for this) is also there.

There is lots more coming. I have received several new or upgraded graphics packages, which will make up a new `TeXgraphics` directory, and there seem to be new drivers ever other week. One of the most interesting is `crudetype` which answers the need for a readable line-printer output at sites where the use of bitmapping printers is rationed or excessively expensive. I have had some difficulty getting `crudetype` to pick up the `tfm` files it needs to calculate rough spacing, but once that is corrected, `crudetype` will be made a part of the distribution.

Finally, I take this opportunity to express my great debt to Elizabeth Tachikawa, whom many of you have met by telephone when you have called about the Unix `TEX` distribution. Without her management of all the administrative side of the distribution, it would simply be impossible. If the documentation for compilation and installation has

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 T_EX 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 than 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, T_EX'ing, and writing—doesn't benefit from a faster access time. T_EX 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 5¼" 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.