# 1 A DVI to ATARI ST Previewer, ST SLM and HP Laser Print Driver

## 1.1 Preface

The following text refers to the original **dvist** program. The sections following describe differences between the respective versions. For the most part, the print drivers use an similar interface to the original **dvist** program with minor differences.

## 1.2 Hardware Requirements

The **dvist** program will print a DVI file typically generated by TEX onto an Atari ST monochrome monitor. The program is known to work with the following minimum hardware:

- ATARI ST (1Mbyte although it may work with 512K systems and TOS in ROM).
- Monochrome Monitor
- Floppy disk drive

Included in the package are:

- dvist.ttp
- server.c
- serve (a unix shell script)
- server.doc (a brief description of the server protocol)
- dvist.tex (this file)
- dvist.dvi (latex'ed dvist.tex)
- font files required to print the dvist.dvi file
- atari.mf (Metafont definition used to create the font files).

as an arc'd file.

## 1.3 Installation

### 1.3.1 The Previewer

To demonstrate this program it is necessary to first extract the arc files contents and create a font support directory called \fonts on your disk. This directory contains further directories each reflecting a different font style. The font files (in a packed format) are placed into these directories. The names of these files correspond to the magnification of the font (that's another story in itself). As arc does not have the ability to store and extract the file tree structure, we must rely on you to organize the directories properly. In the archive, you will find the font files named as XXXXXX.YYY. A directory in \fonts should be created with the name XXXXXX. The font file should be renamed to YYYpk and placed into directory XXXXXX. e.g. cmbx10.96 should be placed into \fonts\cmbx10 as 96pk. This should be done for all of the other font files as well.

### 1.3.2 Server

The server program need only be compiled with your C compiler. Three compile time options are available. They are:

- -DSTDIN=0. If your system cannot read from STDOUT.
- -DDEBUG. This forces a fixed checksum to ASCII ".". Acknowledge is set to "A" and NOT Acknowledge is set to "N".
- -DDEBUG1. Implements DEBUG and also prints debugging information to STDERR.

The server when compiled with DEBUG active will NOT work with the preview program as the checksum will be incorrect for requests. It is provided so that you can type client commands directly to the server in order to observe its behavior. A brief description of the protocol is found in the file server.doc.

A small shell script is also provided called serve. This places a UNIX system into raw mode and then calls on the server program proper.

## 1.4 Use

To run the program with this file as input use the command:

dvist dvist

After loading the appropriate fonts, the screen should clear and this text appear on the screen. The keyboard cursor keys are used to scroll the text as well as more forward and back pages. The ESC key exits the program. The command "dvist -h" generates a simple help message.

The program can also access its fonts and dvi files from a remote UNIX or OS9 system. Remote files have a drive specifier "r:" prefixed to their path names. Thus the command:

dvist r:dvist

would load the fonts locally and the dvist.dvi file from the remote server. This negates the need to download dvi files prior to previewing them with this program.

A number of options are available for this program. Briefly they are:

- -f path-list: Locate the base font directory in one of the alternatives e.g. dvist -f a:\fonts,r:/usr/tex/lib/fonts r:thesis.
  This would load the font files from the directory a:\fonts. If a font file is not found there, it will be loaded from the host directory /usr/tex/lib/fonts. The dvi file would also be loaded from the server.
- -s page-number: Start previewing at page page-number.
- -m magnification: Magnify the preview. Default magnification is 1000. The font files corresponding to this magnification must be present.
- -v resolution: Use resolution dots per inch (Default 96dpi). Again, font files must be available for this size.
- -e file-name: Redirect errors to file-name. You should NOT use the remote specification here as fprintf has not been converted to run with a remote file specification.
- -h: Print a help message.

## 1.5 Misc.

A server program in C has also included. It works for reads but has not been thoroughly tested for writes (i.e. ATARI to Host). It has been run on a Cadmus 9000, ICM-3216, Sequent Symmetry, and a SUN 3/160 running SysV and 4.2. It has also been tested on a QT running OS/9. We tried to get it up on VMS but, well . . . you know ;-). (B.T.W. If anyone can tell us how to solve this problem we'd appreciate it. )

We hope to be able to eventually distribute source to this program as well. However at this point, internal problems have prevented us from doing so.

### 1.5.1 Obtaining the Complete Font File Set

It was impossible to send the font files over the net. The complete set (as we define complete) is close to 4Mbytes of data. A working subset for a large document is about 500K and that could be pruned down. Our working set includes 80 and 96 dpi at magsteps of -1, -.5, 0, .5, 1, 2, 3, 4, 5, 6 for latex and cm fonts. We will be generating slitex fonts in the near future. For those that wish to generate there own fonts expect to use about 48 cpu hours on a SUN 3/160. We can forward our local.mf definition file out to you if you wish. In the meantime, we might be able to copy these disk for you PROVIDED that:

- You advise first (though e-mail) so that we can gear up.
- When you receive an acknowledgement, send us the atari disks and a return addressed mailer.
- Provide a few bucks for postage.

You can grab us at:


```
Phone: (416)736-2100 x7765                          Tyler Ivanco
BITNET: fs300022@yusol  UUCP: ...mnetor!yunexus!stpl!tyler
                                ...utzoo!/
tyler@stpl.ists.ca
or
```

Phone: (416)736-5359 Avy Moise
BITNET: fs300013@yusol UUCP: ...mnetor!yunexus!stpl!avy
..utzoo!/
avy@stpl.ists.ca

Our Post address is:

CRESS
York University,
4700 Keele Street,
Downsview, Ontario,
CANADA
M3J 1P3

## 1.6 Changes In Version 3.2 of DVIST

Perhaps the biggest change in 3.2 is the addition of source to the distribution. We have not embellished the program with windows and the like. We just didn't have the inclination since the program works to our satisfaction and we do hope that someone out there will do the work for us (synonym LAZY or just Tyler to my wife).

Operationally, we have modified the program to improve it's perceived speed when previewing long documents. This perception becomes reality when previewing small portions of long documents through the used of demand loading of fonts. We have two versions of demand loading. The first form demand loads the entire character set of a given font when it is first encountered in the document. The second form loads only those characters used on the page. The former is ideal for local font loading as the Atari has rather slow seek functions. The latter seems better suited to those using the fonts that reside on a server.

We've also modified the search pattern for fonts. It currently searches for those fonts that are in a range of $\pm 10$. In addition the font list search path may be defined in the "dvist.rc" file (located in the current working directory).

Outside of a bit of bug fixing (that is now it has more bugs ...) and reorganization that's about it.

Finally we would like to thank Chris Strunk for his helpful suggestions and critisms.

## 1.7 Changes In Version 3.3 of bf DVIST

Well, it turned out that no-one put together a windowed version of the program for us so...well we did (blush). We did it primarily out of curiosity... (so that is how you do windows!) and because we could not afford personal SUN work stations with their sophisticated previewers.

Of course, we are sure that something is done improperly and would be happy to receive any constructive suggestions (subject to our fragile egos).

The windowed version currently supports up to 6 windows into the SAME file subject to memory limitations. It is not possible to re-open files or modify the parameters after preview starts.

To setup the system create directory `RSC` at the ROOT level and place the resource file "`dvist.rsc`" into it. We've started a practise of editing our binaries so that they always look for resource files at this point. Thus it isn't necessary to keep multiple copies of these files cluttering up the drive just because we wish to start the program from some other directory.

All other setup parameters are the same as found in previous versions.

For the most part, the operation of the program and window management should be self-explanatory. The Horizontal arrows move forward/backward one page, the vertical arrows move approximately one line within the page. Sliders work as they should (?) as well as the paging regions about the sliders. Resizing of windows is also operational.

## 1.8 A DVISLM Print Driver

Just to show that the SLM804 isn't totally useless, we've quite literally thrown together a TeX print driver for this device. This driver was put together about a year ago when we were able to borrow a printer for a couple of days. Recently we've actually acquired one of these beasts and so felt that it would be nice to fix the program up a bit and distribute it.

Program execution is just as in the version 3.2 of the previewer with the exception of user interaction. A couple of options are present to control page access and count. The option "`-n <number>`" set the maximum page count and "`-s <number>`" specifies the starting section.

**One important note.** The current driver is configured to find the printer at phyiscal ACSI address 7. It does not scan the devices on the system. If your printer is set at any other address such as 2 or 5, it will be

necessary to ether change the printer address (as discussed in the printer documentation) or edit the binary image with a program such as dlii. Look for a Hex string similar to "`ea 00 8a`". The upper three bits of "`ea`" represent the unit number. Change this bit field to the appropriate value; e.g. physical unit 2 would be "`4a`". After saving these changes, test the program with your hard disk etc. disconnected; after all we wouldn't want this preview to treat your disk drive to a sample of its wares. Once satisfied, then try to connect your hard disk to the system and away you go.

## 1.9 The Atari HP LaserJet Print Driver

Also included in the distribution is the HP LaserJet II (and perhaps PLUS) DVI print driver. It operates by generating a `*.ljp` file that can be transferred either through the serial or parallel port of the ATARI ST as a binary file. Alternatively, it is possible to redirect the output to the PRN: or AUX: ports using the `-o` option. This driver should be able to print at about 8 ppm using this redirect function through a parallel port.

## 1.10 Misc. Notes

A number of new options have been added to the programs. These include:

- `-l` Set Left margin (in 1/1000 of an inch).
- `-c` Produce n copies of each page.
- `-t` Set Top margin (in 1/1000 of an inch).
- `-o` Set output to a specific file or standard out (`-`). This option is available only in dvihp.
- `-a` Start at absolute page n.
- `-e` Set standard error to a specifig file

All three programs will read a default file, `dvist.rc` in order to set the standard font search path. Each item within the search path is separated by a comma (`,`). In addition, the print drivers will use the environment

variable `TEXFONTS` if it cannot find the file `dvist.rc` or it hasn't been set on the command line.

We have not distributed any other fonts with these programs. 300 dpi fonts can be found just about anywhere a TEXlaser printer is found and with the recent Metafont distribution, it just isn't necessary to clutter up the network with megabytes of files.

Source hasn't been included in this distribution. Why, well we are too embarrased to show it around just yet. However it isn't because this is a big secret; we will provide others with the source so long as they sign an "non-embarrassment" agreement. You see, Avy and I are ex-FORTRAN programmers that have moved up (or is it down) to C. Personally I feel that binary micro-code is far more productive and that documentation and mnemonics are simply an annoying facit of some under-productive programmer trying to flesh out his/her code. This is evidenced by the TEXbook where it says virtually nothing in a straightforward mannar and again by this paragraph.