contains an introduction to the system and pointers to interesting directories.

To facilitate periodic updating, FTP command files and directory listings in both alphabetic and reverse chronological order are maintained in each distribution directory. TOPS-20 sites can use the FTP UPDATE command instead to get changes with even less effort.

For the benefit of Unix sites, compressed `tar` files are also maintained.

Source plus documentation amounts to about 1.5Mb, and executable code for each device amounts to 80Kb–150Kb, depending on device and host machine; this is probably too large for EMAIL to Bitnet and Usenet sites.

### Future work

Major pieces of work remaining to be done:

(a) merging in support of PostScript resident fonts;
(b) addition of more \special{} support to the laser printer drivers; currently only DVIALW supports a \special{} command;
(c) addition of support for windowing systems on bitmapped workstation displays; X-windows support is in progress locally.

Volunteer contributions are most welcome!

### DVItoVDU 1.7 and PSPRINT 1.1

Andrew Trevorrow
University of Adelaide Computing Services

### Introduction

DVItoVDU is an interactive page previewer that drives a variety of commonly available terminals. PSPRINT is a PostScript driver for the Apple LaserWriter that can print a DVI file, a raw PostScript program, or an ordinary text file. Both programs run under VAX/VMS and are in the public domain. Copies of the software, including source code and documentation, have been sent to Stanford and Kellerman & Smith for inclusion on their VAX/VMS distribution tapes.

Now that TEX under VAX/VMS seems to have reached its final state, further development of DVItoVDU and PSPRINT is most unlikely. (Another good reason is that I'm leaving my position at the University of Adelaide. If anybody out there needs the services of an experienced

TEXnician, I can be reached at my University address or via electronic mail; my ACSNET address is akt@uacomsci.ua.oz. I'm particularly interested in working overseas on a short-term basis.)

### DVItoVDU 1.7

Version 1.5 was described in detail in TUGboat vol. 7, no. 1. Here is a list of the significant changes that have occurred since that release:

- All font-dependent code has been moved out of the main module and into FontReader. This module replaces PXLReader and can read PXL or PK files. The /font_directory and /dummy_font values determine which format to use, as well as the location and naming convention of all font files. (I was going to handle GF files as well, but finally decided that any site using such files would eventually switch to the PK format. PK files use about a fifth of the disk space required by PXL files, and interpreting the compressed information only adds about 5% to the processing time. See Tomas Rokicki's article in TUGboat vol. 6, no. 3.)
- The /font_directory value is restricted to a single directory rather than a list of directories. The /dummy_font file must represent an unmagnified font.
- The main module's Full display routines have been changed to handle a new method of loading character bitmaps into dynamic memory on a demand basis. DVItoVDU is now slightly faster, but tends to use more memory.
- DVItoVDU can now handle all TEX character codes from 0 to 255.
- There are two new /vdu values. VT220 is a new implementation by John Mann at Monash University that uses downloaded chunky graphics to get a window resolution of 132 by 100. VIS240 is for a ReGIS VDU with a larger window region.
- The page position and first few bytes of any \special commands are now displayed. (Note that PSPRINT allows the use of \special to include a file of PostScript commands.)

## PSPRINT 1.1

"PS" stands for PostScript, the printer language used by the Apple LaserWriter. The decision to replace our Imagen IMPRINT-10 with a LaserWriter was motivated by a desire to support PostScript. Upgrading to a new, high-performance PostScript printer should require only trivial editing changes to a few text files used by the PSPRINT system.

Like DVItoVDU, PSPRINT comes with a comprehensive User Guide explaining how to run the program, and a System Guide containing installation details. The following material has been extracted from these two documents. I'll start with a brief system description.

### System Overview

PSPRINT is *not* a VMS print symbiont; it is actually made up of a number of components. The following sequence of events occurs when a user types 'psprint file':

1. `psprint.exe` is invoked. This small Modula-2 program extracts the file parameter and qualifier values from the command line according to `psprint.cld`. If everything seems okay then `tex_ps:psprint.com` is activated. (`tex_ps` is the logical name for a public-access directory containing all the necessary files that make up the PSPRINT system.)
2. `psprint.com` is a command file that does most of the messy system-dependent work. It can handle three file types: a DVI file, a PostScript program, or an ordinary text file.
3. Given a DVI file, `psprint.com` uses `psdvi.exe` to create a temporary PostScript file. (PSDVI is another Modula-2 program, much of its code borrowed from DVItoVDU. It too can read PXL or PK files.) Typical documents take about 2 CPU seconds per page to translate on a VAX-11/785.
4. Each type of file is bracketed by appropriate header and trailer files before being sent as a single job to a queue called "PS". The standard VMS print symbiont then sends each job to the LaserWriter. Temporary files created in the user's current directory are automatically deleted after printing.

Much of the behaviour of PSPRINT can be changed by simply editing `psprint.com` and/or the various text files used by this command file. Such flexibility is of particular benefit to those unfortunate sites without Modula-2.

The remaining sections have been extracted from the User Guide available to people at our site.

### TEX Output

PSPRINT looks for a DVI file by default. To print `foo.dvi`, simply type:

```
$ psprint foo
```

PSPRINT will translate `foo.dvi` into a temporary PostScript file called `foo.tmp`. The DVI and TEX page numbers are displayed as each page is translated. If no errors are found, the temporary file is automatically sent to the PS queue where it is printed (and then deleted).

PSPRINT translates pages in reverse order so that your document is collated automatically. Various qualifiers allow you to select particular pages for printing, or produce landscaped output, etc. See below for details.

### PostScript Output

PostScript is a complete programming language with a powerful set of operators for manipulating text and graphics. A PostScript program is an ordinary text file. If you give it a name like `foo.ps` then sending it to the LaserWriter is simple:

```
$ psprint foo.ps
```

If you decide to use some other file extension you'll have to add the `/ps` qualifier.

People who want to write PostScript programs will need to get the *PostScript Language Reference Manual* by Adobe Systems Incorporated (published by Addison-Wesley).

### Ordinary Text Output

To print a normal text file, just use any file extension except `.dvi` or `.ps`:

```
$ psprint foo.lis
```

The output is *not* collated. By default you get 66 lines per page, and any lines more than 80 characters long are truncated. The `/landscape` qualifier will rotate the paper so that you can have lines up to 132 characters long (but only 50 lines per page).

Type 'psprint foo.ps /text' if you want to print a PostScript file rather than have it interpreted. FORTRAN carriage-control files and RUNOFF output can also be printed, but overprinting is not handled correctly.

*PSPRINT Qualifiers*

The following three qualifiers override PSPRINT's assumptions about the format of the input file based on the given file specification:

- `/dvi` — for a DVI file.
- `/ps` — for a PostScript program.
- `/text` — for an ordinary text file.

Some qualifiers can only be used with a DVI file:

- `/pages=first:final` — to select a subrange of pages for printing. `first` and `final` can select either DVI page numbers (positive integers) or TEX page numbers (in the same format as used by DVItoVDU). More than one DVI page may match a particular TEX page specification. If `first` is a TEX page then the *lowest* matching DVI page is selected, and if `final` is a TEX page then the *highest* matching page is selected. Note that `first` and `final` are optional; they default to the first and last DVI pages respectively. Omitting ':`final`' entirely sets `final` equal to `first`.
- `/magnification=n` — to override the DVI magnification. `n` is a positive integer 1000 times the desired magnification and should match one of TEX's `\magstep` values.
- `/units=xx` — to set the units of dimensions displayed in some PSPRINT messages (e.g., if the page is off the paper). `xx` can be one of `in`, `cm`, `mm`, `pt`, `pc`, `bp` or `px`.
- `/stats` — to see statistics on the number of rules/fonts/characters used on each page.

The remaining qualifiers apply to all types of input files:

- `/landscape` — to get landscaped output. If given a DVI file, PSPRINT will check each page to make sure it fits within the rotated paper.
- `/delete` — to delete the input file after printing.
- `/notify` — to be notified when the print job on the PS queue has finished.
- `/copies=n` — to specify up to 100 copies. Use of this facility is not encouraged. Note in particular that copies of a multiple-page document will *not* be collated.
- `/note=string` — to include a note on your banner page.

*Error Messages*

If PSPRINT detects one or more errors while translating a DVI file then the temporary PostScript file is automatically deleted and nothing is printed; not even a banner page. The sort of errors you are most likely to see include:

- `Page off paper (paper is ...`
  — the current page contains material beyond one or more paper edges. Use DVItoVDU to see where the problem lies.
- `Couldn't open font file: ...`
  — your DVI file uses a font at a non-existent size. Try a different font magnification in your source file.
- `Couldn't open \special file: ...`
  — PSPRINT could not locate the given `\special` file. Check the file's spelling and/or directory location.

Error messages apply to the most recently displayed DVI/TEX page. If your DVI file uses a large number of fonts or a few very large fonts then you might get an error page with the message "`Font_Memory_Exhausted`". Try printing just a few pages at a time or using fewer/smaller fonts. The LaserWriter has a large but finite amount of memory for storing the character bitmaps downloaded by PSPRINT. This limitation is unlikely to worry most people.

An error in a PostScript program should produce some sort of message on your printed output. If this happens a lot, go back and read the *PostScript Reference Manual* more carefully.

*Merging TEX and PostScript*

Let's assume you have a file called `fig.ps` that contains suitable PostScript commands for generating some sort of figure. To include this figure in a TEX document, just type '`\special{fig.ps}`' after leaving sufficient space:

```
\midinsert
    \vskip 3in
    \special{fig.ps}
\endinsert
```

PSPRINT interprets the `\special` argument as a file name and includes this file in the information sent to the LaserWriter.

It may take a little practice to position the figure correctly. When you print a raw PostScript file (e.g., '`psprint fig.ps`') the default origin is at the bottom left corner of the paper. However, when you include this file in a TEX document the origin is automatically moved to the position of the

`\special` command. You can move the figure about by shifting the `\special` position, or by including a `translate` command in the PostScript file. The latter option is faster because you don't have to run TeX again. For example, '`72 36 translate`' at the start of `fig.ps` would move the figure 1 in right and 0.5 in up (PostScript's default units are "big points" where 72 bp = 1 in).

Alternatively, you may prefer to place your figure at an absolute position on the paper (i.e., at the same position produced by '`psprint fig.ps`'). Simply include an `initgraphics` command at the start of `fig.ps`. The position of the `\special` command then becomes irrelevant; just make sure it's on the right page!

The hardest part of the scheme is creating the PostScript file in the first place. You are faced with two possibilities:

1. Learn enough PostScript to be able to draw your own figures.
2. Use a program to convert an existing graphics file into a corresponding PostScript file. This assumes you are able to produce graphics of some sort (e.g., a Zeta plot file) and have access to a suitable conversion program.

PSPRINT also allows arbitrary PostScript text to appear after the file name in a `\special` command. For example, '`\special{fig.ps 2 1 scale}`' could be used to double the width of the figure in `fig.ps`. At least one space must be used to terminate the file name; further characters are included as a new line at the *start* of the given file. There is a limit of about 200 characters. Note that it is possible to include the same PostScript file more than once, but with a different starting line each time.

## Benson-Varian 9211 Looking for a Home

Until it was disconnected earlier this year, a Benson-Varian model 9211 electrostatic printer was used by the American Mathematical Society as a TeX output device. This machine is offered free to anyone who will pay the shipping costs from Providence to its new location.

This machine has a resolution of 200 dots per inch and uses liquid toner. It uses coated, roll-fed paper, 11″ wide. Physical dimensions are 24.25″ w, 27.25″ d, 40.25″ h; weight, 325 lb. The machine was obtained new in Summer 1979 and has been fully maintained under a service contract with the manufacturer.

It was connected to a DECsystem-20 via a Monolithic MSC 8004 microprocessor, which is also included in this offer. For details, contact

George M. Ogilvie
American Mathematical Society
P. O. Box 6248
Providence, RI 02940
    401-272-9500, ext. 279