

# Automating PDF production from SWP

Philip A. Viton

November 29, 2004

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Updates</b>	<b>4</b>
<b>3</b>	<b>Get the Software</b>	<b>6</b>
3.1	GhostScript . . . . .	6
3.2	SWP files . . . . .	7
3.3	The TeXConverter . . . . .	7
<b>4</b>	<b>Install the Software</b>	<b>7</b>
4.1	GhostScript . . . . .	7
4.2	SWP files . . . . .	7
4.3	The TeXConverter . . . . .	7
<b>5</b>	<b>Set up the System</b>	<b>8</b>
5.1	Preliminary note . . . . .	8
5.2	Install a logical PS printer . . . . .	8
5.3	Configure the PostScript printer . . . . .	9
5.4	Get your format name . . . . .	10
5.5	Edit the GhostScript batch file . . . . .	10
5.5.1	GhostScript 6.50 or later . . . . .	10
5.5.2	Pre 6.50 GhostScript . . . . .	11
5.6	Command-line setup under Windows NT4/2000/XP . . . . .	11
5.7	Command-line setup under Windows 95/98/ME . . . . .	12
5.8	Set up the TeXConverter . . . . .	13
<b>6</b>	<b>Test the system</b>	<b>13</b>
6.1	Command-line system . . . . .	14
6.2	TeXConverter . . . . .	14
<b>7</b>	<b>General Usage</b>	<b>15</b>
<b>8</b>	<b>Troubleshooting</b>	<b>15</b>
8.1	Environment size under Windows 95/98/ME . . . . .	16

<b>9 Drivers</b>	<b>17</b>
9.1 Windows 95 . . . . .	17
9.1.1 Working drivers . . . . .	17
9.2 Windows 98 . . . . .	18
9.2.1 Working drivers . . . . .	18
9.2.2 Problematic drivers . . . . .	18
9.3 Windows ME . . . . .	18
9.3.1 Working drivers . . . . .	18
9.4 Window NT4 . . . . .	18
9.4.1 Working drivers . . . . .	18
9.5 Windows 2000 . . . . .	18
9.5.1 Working drivers . . . . .	18
<b>10 Upgrading</b>	<b>19</b>
<b>11 Conclusion</b>	<b>19</b>

## 1 Introduction

You've made the next breakthrough in your field, and written it up with SWP. Now it only remains to tell the world about it, and today, that means the Internet. What are your options?

**TeX/DVI** The advantage of these is that they're small and quickly retrieved, and when processed will preserve your typographic intentions. But each requires specialized software to be viewable (in the case of the TeX file, up to 100 MB of software), not to mention any specialized packages or fonts that you used in your manuscript.

**Scientific Notebook** SN also uses the `.tex` source; math looks like math, SN has good hypertext facilities, and the SN-viewer download is small. But the viewer is available for Win32 systems only; and does not preserve typographic fidelity.

**HTML** This is viewable by just about everyone, text-only documents are small, and its hypertext features are excellent. But HTML does not preserve typographic fidelity, and though it's possible to represent math adequately, the quality leaves a lot to be desired, especially when precise positioning is important.

**PostScript** It's easy to produce PostScript from the TrueTeX Previewer. Like DVI, it will reproduce the layout of your document. But TrueTeX produces *very* large PostScript files, and at least on Windows platforms, PostScript viewers are comparatively rare. The only one I've tried (`gsview`) is a bit blocky on screen, though print quality is excellent.

**PDF** Adobe's Portable Document Format (PDF) preserves typographic fidelity (like DVI and PostScript), has excellent on-screen and print quality, and like HTML is almost universally viewable via free PDF Readers. PDF documents are smaller than TrueTeX PostScript but substantially larger than either TeX/DVI or HTML; however, in these days of fast Internet access, that's not the impediment it once was.

So PDF is an attractive solution. But how to produce PDF from your SWP output? The most attractive possibility is to use or upgrade to SWP 5.0 or later: these include a version of pdfTeX, which will automatically produce PDF documents from within your SWP workspace. A major advantage of this is that you can support some features — like rotation of tables and figures — which are not directly supported by TrueTeX. It's also faster and more convenient than the procedures outlined below.

Otherwise, you could use Adobe's Acrobat Distiller (or Acrobat Exchange); but these are not inexpensive programs. Finally, you could try GhostScript, a free PostScript interpreter which can also produce PDF. You may already have considered this, but been dissuaded since the on-screen appearance of GhostScript PDF was somewhat blocky and a bit hard to read — though the printed output was excellent. But with GhostScript 6.0+ the situation has changed, and now the screen output is — in my estimation — indistinguishable from that produced by the commercial Acrobat products. So GhostScript is a free and viable tool for the production of high-quality PDF from your SWP document.

And it's easy, if a bit cumbersome, to produce PDF this way from SWP. What you do is:

- Typeset-compile your document, using the TrueTeX formatter.
- Preview it using the TrueTeX Previewer, and then, print-to-file using a PostScript printer. Note that you do not need to have a physical PostScript printer installed to do this. The result is a (large) PostScript file on your disk.
- Run the PostScript file through one of GhostScript's conversion batch files.

Bingo! Perfect PDF. An additional consideration is that because you first produce a PostScript file, you don't have to worry about the usability of TrueType fonts on other operating systems: the PostScript printer driver converts them to Type1 fonts. The upshot is that your PDF will be readable on any platform supporting the Adobe PDF standard (version 3+).

The only problem with all this is that you need to run through the sequence of individual steps every time you want to produce PDF. The remainder of this note explains how to set up a command-line system to do it automatically. When it is set up, a single command will produce a PDF version of your `.tex` document in your source directory. And on a fast computer — say 400 Mhz or better — processing a 30-page math- and graphics-rich document takes only

about 30 seconds, even with three LaTeX compilations. Throughout this note I'll use "SWP" to mean either *Scientific WorkPlace* or *Scientific Word*, versions 3.0+ (though I believe that it will also work with SWP 2.5, as long as you're running on a Win32 platform). **A copy of this note in .html format is included in the SWP distribution, see below.** You can also glance at the ?? , produced by the system.

What's the catch, you ask? The only one I've noticed is you cannot produce live hyperlinks. In other words, the PDF you will be producing is no more (and no less) than the typeset output produced by SWP.

Don't like command-line utilities? You're in luck. Steve Mayer has recently updated his GUI TeXConverter to support this setup for producing PDF — see section 3.3.

## 2 Updates

- **November 23, 2004:** I've just noticed that the PDF produced this way may contain your user name (login name) in its "Author" field. This is usually no problem, *except if you may be producing something like a referee report and you want to be totally anonymous.* Whether this happens or not seems to depend on a combination of operating system and printer driver: for my printer, it does happen under Win XP but not under NT4; you will need to check. If you want to be more anonymous, you can first produce your PostScript file in the usual way, by printing-to-disk via a PostScript Printer. Now open the PostScript file in a text editor. Near the top you'll see a line beginning `%%For:` followed by your user name. Replace the user name by something like `xxxx`. Finally, invoke `ps2pdf` by hand — ie not through the batch file (which would start by remaking the PostScript). It's possible — and fairly easy — to automate this too; if anyone needs it, let me know.
- **January 30, 2003:** Added a test file `pdfgs3.tex` which does not contain the `\msihyperref` construct used in SWP4+. If you're using SWP 3.5 or earlier, you should test your setup on this file, and not on `pdfgs.tex`.
- **June 21, 2002:** There is a bug in current issues of Ghostscript which prevents them from correctly producing PDF files which can be loaded into Acrobat Reader 3 for Windows: in GhostScript's `issues.htm`:  
"pdfwrite + TT font => Acrobat 3.x for Windows gives error. Running `ps2pdf12` on the file `test1.ps` produces a PDF on which Acrobat 3.x for Windows complains about not being able to find or create a particular TrueType font that is embedded in the PDF file. However, Acrobat 3.x for other platforms, and Acrobat 4.x for all platforms, accepts the file. SourceForge bug #201955, February 14, 2000.  
Since Acrobat 3 is superseded by Acrobat 4 which is available at no charge, and the file produced by Ghostscript meets published PDF specification, this will most likely be left as is."

If it is important that your PDF be readable by Acrobat 3, then I think you will need to install GS 6.50, instead of the current version. It is still available from the GS site.

- **June 14, 2001:** We may have solved at least some remaining problems with ligatures etc on Win95/98 machines. Thanks to Steve Mayer for extensive help on this. See below, in the installation instructions. Also, miscellaneous editorial changes, mostly upgrading the instructions to reflect the default locations for GhostScript 7.00.
- **May 13, 2001:** Started a new section on PS drivers, with information on what does and does not appear to work. *This section needs your input!* See 9.
- **May 1, 2001:** Ghostscript 7.00 is out, and it appears to solve the Unix-line-endings problem noted on January 17.
- **January 23, 2001:** Added a sub-section 5.3 on configuring the PostScript printer: a bad choice here could increase the size of the generated PostScript file by a factor of about 100 (!); and unfortunately some printer drivers come pre-set with bad choices.
- **January 22, 2001:** The latest version of Steve Mayer's TeXConverter (dated January 22, 2001 or later) can simplify some of the setup, and I've revised the instructions to suit. I now recommend that you obtain the TeXConverter even if you're a die-hard command-line user. See section 3.3 for the location.
- **January 19, 2001:** Major bug fix. The sequence: change to GhostScript's directory, run `ps2pdf12.bat` and then switch back to the source directory, fails if the two are on different drives. (Thanks to Jacek S. Stecki for figuring this out). We now set separate environment variables for the drive and the directory. A revised `pdfgs.exe` handles this for Win 95/98/ME users; and there is a revised version of `pdfgsmakexx.bat`. Note that there are no changes in the setup of the PostScript printer, or in the batch file `pdfmakegs.bat`. So to update:
  - Get the new zip file — see section 3.2
  - Replace `pdfgs.exe` in your path
  - Go to section 5.6 for WinNT4/Win2k, or to section 5.7 for Win 95/98/ME users to configure the new batch files.
  - If you use the TeXConverter, there is also a new `pdfmakegstc.bat`. Go to section 5.8 to for customization instructions. *You must also upgrade the TeXConverter itself* to a version dated January 19, 2001 or later. The new TeXConverter will warn you the batch file it uses to do the conversion is old, and will remind you to update.

- **January 17, 2001:** The current distribution of GhostScript 6.50 apparently comes with batch files with Unix line-endings and this can cause them all to fail on Win 95/98/ME systems. (WinNT4 and Win2k do not have this problem). I can't confirm this, since I no longer have access to those operating systems; and if the problem does exist, I anticipate that it will be fixed promptly. Nonetheless, if the GhostScript stage fails, you can get a utility, `unix2win` from ?? which should fix the problem. You need to process the following batch files in GhostScript's `\lib` directory: `ps2pdf12.bat`, `ps2pdf13.bat`, `ps2pdf14.bat` and `ps2pdfxx.bat`. (Suggestion: copy them each to `.bau` files, then run, eg `unix2win ps2pdfxx.bau ps2pdfxx.bat`).

Alternatively, you may wish to get the older GhostScript 6.01, which works fine. Go to

??

click on the folder icon labelled `GS601` and then download `gs601w32.exe`.

- **January 12, 2000:** GhostScript 6.50 is out, and the structure of the critical batch file has changed: see section 5.5 for details. The default location for GhostScript 6.50 is now `c:\gs`, but except when referring to the new batch file, I've not changed the text to reflect the new GS version. I've also added a Windows ME caveat to the note on increasing the environment size.

As distributed, the SWP support makes PDF via the GhostScript batch file `ps2pdf12.bat`, which produces PDF compatible with the Acrobat Reader version 3+. GhostScript also comes with `ps2pdf13.bat` and `ps2pdf14.bat` which produce PDF compatible with Acrobat 4+ and Acrobat 5+, respectively. If you want to limit the readability of your PDF to either of these Acrobat versions, I see no reason why you can't use these files. See below.

- **September 11, 2000:** Miscellaneous typos fixed; added a Troubleshooting note (see section 8.1) on fixing the `Out of environment space` problem under Win95/98: I'd hoped that because the batch files set only a couple of variables, this problem wouldn't arise; but it has.

## 3 Get the Software

Fortunately, you already have most of what you need.

### 3.1 GhostScript

If you have Aladdin GhostScript 6.0+ you don't need to upgrade; otherwise, you can get the latest version from

??

Click on the link to “Obtaining Aladdin GhostScript”, then find the section for MS-Windows. You should download the setup file, currently `gs814w32.exe`.

### 3.2 SWP files

You will also need a small `.zip` archive containing a few SWP-specific files. You can download it from ??

### 3.3 The TeXConverter

Steve Mayer’s TeXConverter is a GUI interface to various TeX/LaTeX converters: with it you can avoid any inconveniences associated with running a command-line program. Steve has updated the TeXConverter to provide support for producing PDF as described here; and if you get it, you’ll find that setting up PDF-conversion is somewhat simpler. You can obtain the TeXConverter at:

??

The files themselves are under “Download” on that page.

## 4 Install the Software

### 4.1 GhostScript

GhostScript is supplied in a self-extracting executable file. To install it

- Start the Windows Explorer.
- Find the GhostScript distribution you downloaded. Double-click on it; this starts the extraction process.
- I recommend that you install to the default location, which will put the executables in `c:\gs\gs7.00\bin`; and the supplementary (library) files in `c:\gs\gs7.00\lib`. If you elect to install to a different place, make a note of it.
- You have the option of not installing the GhostScript fonts. Though they are not needed for PDF production from SWP, I recommend that you install them anyway.

### 4.2 SWP files

Unzip the contents of `pdfgs.zip` to a temporary directory.

### 4.3 The TeXConverter

- Choose a directory for your files, and unzip the TeXConverter’s files to that directory.

## 5 Set up the System

### 5.1 Preliminary note

The setup is slightly different for Windows 95/98 and Windows NT4/2000 (though the Win 95/98 setup will work under NT4/2000). The reason is that we need to take the name of your source (`.tex`) file and obtain the full path to its location. In Windows NT4/2000 this can be done via those systems' extended batch language, but this is not available under Windows 95/98.

To cope with this, the SWP distribution provides a utility (`pdfgs.exe`) which extracts the directory name and puts it into an environment variable for later use. As far as I can tell, the only way to have the new environment variable available when the conversion programs run is to have `pdfgs.exe` *itself* call a batch file which does all the work. So we have somewhat convoluted series of program calls: an initial batch file (`pdfmakegs.bat`) calls `pdfgs.exe` to create the environment variable. `pdfgs.exe` calls a second batch file (`pdfmakegsxx.bat`) to do the work, and this second batch file ends up calling GhostScript's `ps2pdf12.bat` and then `ps2pdfxx.bat`. It's complicated, but it does work, and of course it's all invisible to you anyway. Under Windows NT4/2000 or Steve Mayer's TeXConverter, everything up to `ps2pdf12` can be done with a single batch file, and that's what we'll be setting up.

OK, now to work.

### 5.2 Install a logical PS printer

In this step we install a PostScript printer, which we'll use to create the PostScript disk-file. If you already use a PostScript printer, you must install a second one unless it's already configured to print to a file. This second one can be the same type (brand and model) as the first, though you will need to give it a different name. The installation is via the standard Windows printer-installation routine: you may need your Windows CD.

It appears that the choice of a PostScript printer can affect the quality of the PDF you produce. I've had good results with the HP LaserJet 4/4M Plus PS 600; but it's easy to experiment with other printers. Also note that some PostScript-capable printers are set up to automatically emulate PCL printers, rather than PostScript; you must make sure that the printer you install in this step is configured to produce PostScript automatically. (For example, under Windows NT4, the Lexmark Optra S 1250 actually installs two drivers, which appear as two separate printer choices: one is for PCL and one for PostScript. You must be sure to choose the PostScript one.)

See also the discussion in section 9 for information on what works and what doesn't.

The following just walks you through the Add Printer procedure.

- Click **Start** -> **Programs** -> **Settings** -> **Printers**. The Printers folder opens.
- Double-click on **Add Printer**. This starts a Wizard. Click **Next**



- Chose Local Printer . Click Next.
- In the port selection dialog, check FILE. Click Next.
- In the next dialog, choose the Manufacturer (I recommend HP) and, under Printers, the model you wish to install (I recommend the HP LaserJet 4/4M Plus PS 600).
- Click Next. At this point you may be asked to insert your Windows CD, or, if the driver is already present, you will be given a choice of keeping the existing driver or getting a new one. I recommend keeping the existing driver. Click Next.
- The next dialog asks you to select a printer name. *It will make your life easier if you choose a short, descriptive name **without any embedded blanks**.* I use SWP-PDF. Make a note of what you choose: you'll need it in the next step. At the bottom of the dialog, you do *not* want this to be the default printer. Click Next.
- In the next dialog, you do *not* want the printer to be shared. Click Next.
- In the final dialog, you do *not* want to print a test page. Click Finish.

At this point you've installed your print-to-file printer. There's one last thing you may need to do. If you insisted on giving your printer a name including embedded spaces, you should now write down the name of the printer, converting each space to an exclamation mark. Example: you chose My PDF Printer. You now write down My!PDF!Printer. This is your **file\_printer\_name**. If you wisely chose a name without spaces, that name is your **file\_printer\_name**.

### 5.3 Configure the PostScript printer

A bad choice of printer parameters can increase the size of the PostScript file by a factor of about 100 (!). Depending on your choice of printer you may not have to change anything here, and some of the options may not even be available, but it is a good idea to check.

- Open the Printers folder; select your PostScript-to-File printer, right-click, select Properties.
- Click the Printing Preferences button, and in the resulting screen, the Advanced button.
- Under Graphic, the Print Quality should be 600 dpi.
- Under Document Options -> Postscript Options: the Truetype Font Download Option should be outline

If you are running on Win95/98 it appears essential to do the following:

- With the Printers folder open, select your PostScript-to-File printer, right-click, select Properties
- Click on the Fonts tab
  - Select Always use TrueType Fonts
  - Click Send Fonts As
    - \* At the top of the dialog, under TrueType Fonts, make sure that Outlines is selected
    - \* Under this, set the Threshold to 1 (it's probably now 100).
    - \* Check Favor TrueType Fonts
    - \* Click OK. You're back at the Fonts tab.
  - Click OK to complete the setup.

## 5.4 Get your format name

Start SWP, click Typeset -> Expert Settings and click on the Format Settings tab. If it says TrueTeX MultiLingual your **format** is `latex_ml`; if it says TrueTeX it is `latex`. If it says something else, you're on your own. Make a note of your **format**.

## 5.5 Edit the GhostScript batch file

If you have the TeXConverter dated January 22, 2001 or later you can skip this step and go to section 5.6 if you're running WinNT4 or Win2K, or to section 5.7 if you're running Win 95/98/ME: the TeXConverter can do the following editing tasks for you.

### 5.5.1 GhostScript 6.50 or later

- Find GhostScript's `lib` directory, typically `c:\gs\gs7.00\lib` (the 7.00 will change if the GhostScript version changes).
- Open `ps2pdfxx.bat` in a text editor.
- Find the line `%GSC% @_.at @_.at2`
- Change the `%GSC%` to refer to the full path to `gswin32c`, which is located in GhostScript's `\bin` directory. For version 7.00 installed to the default location you'd therefore change this line to read:

```
c:\gs\gs7.00\bin\gswin32c @_.at @_.at2
```

- Save the file.

### 5.5.2 Pre 6.50 GhostScript

- Find GhostScript's `lib` directory, typically `c:\gs\gs7.00\lib` (the 7.00 will change if the GhostScript version changes).
- Open `ps2pdfxx.bat` in a text editor.
- In this file there are two lines saying `set PS2PDFGS=gswin32c`. Change *both of them* to add the full path to `gswin32c`, which is located in GhostScript's `\bin` directory. For version 7.00 installed to the default location you'd therefore change this to

```
set PS2PDFGS=c:\gs\gs7.00\bin\gswin32c
```

(Remember to do this twice).

- Save the file.

## 5.6 Command-line setup under Windows NT4/2000/XP

We turn now to the SWP-specific files, beginning with Windows NT4/2000.

- Delete the files `pdfmakegs.bat` and `pdfgs.exe`.
- Rename `pdfmakegsxx.bat` to `pdfmakegs.bat`
- Open the renamed file in a text editor
  - Locate the line `set gsdride=c:` and change it if necessary to point to the drive on which you have intalled GhostScript. Not that there should be *no* trailing `\`.
  - Locate the line `set gsdire=\gs\gs7.00\` and change it if necessary to point to the top-level GhostScript directory. If you're unsure what this should say, look for the GhostScript file `uninstall.txt`. Note that this should both begin and end with a `\` character. The idea is that catenating `gsdrive` and `gsdir` provides a correct-syntax path to GhostScript.
  - Locate the line `set swppath=c:\swp35\tcitex\truetex` and make sure it points to SWP's TrueTeX directory. If you're unsure what this should say, look for the file `initex32.exe`. Note that this should *not* end with a `\` character.
  - Locate the line `::set currdrive=%~d1` and remove the two colons.
  - Locate the line `::set currdire=%~dp1` and remove the two colons.
  - Find the line `start /wait %swppath%\initex32 ^&latex_ml %1`. If you found that your **format** was `latex`, erase the `"_ml"`. *Do the same thing in the two other instances, immediately below.*
  - Find the line beginning

```
start /wait %swppath%\dvigdi32.exe -w!SWP-PDF,winspool,...
```

The text between `-w!` and the following comma (here, `SWP-PDF`) should be your **file\_printer\_name**, including any embedded `!` marks.

- If you want to produce PDF compatible only with Acrobat versions 4+ or 5+, change the line

```
call ps2pdf12 %1.ps %1.pdf
```

and change `ps2pdf12` to `ps2pdf13` (Acrobat 4+) or `ps2pdf14` (Acrobat 5+).

- Save the file, and move it to some directory in your path.

The system is now set up and ready to test.

## 5.7 Command-line setup under Windows 95/98/ME

Here's what you need to do on a Win 95/98/ME system.

- Open the file `pdfmakegsxx.bat` in a text editor.
  - Locate the line `set gsdrive=c:` and change it if necessary to point to the drive on which you have installed GhostScript. Note that there should be *no* trailing `\`.
  - Locate the line `set gsdire=\gs\gs7.00\` and change it if necessary to point to the top-level GhostScript directory. If you're unsure what this should say, look for the GhostScript file `uninstall.txt`. Note that this should both begin and end with a `\` character. The idea is that concatenating `gsdrive` and `gsdir` provides a correct-syntax path to GhostScript.
  - Locate the line `set swppath=c:\swp35\textex\truetex` and make sure it points to SWP's TrueTeX directory. If you're unsure what this should say, look for the file `initex32.exe`. Note that this should *not* end with a `\` character.
  - Find the line `start /wait %swppath%\initex32 ^&latex_ml %1`. Remove the caret (`^`) before `&latex`. If your **format** was `latex`, erase the `_ml`. *Do the same in the two other instances, immediately below.*
  - Find the line beginning

```
start /wait %swppath%\dvigdi32.exe -w!SWP-PDF,winspool,...
```

The text between `-w!` and the following comma (here, `SWP-PDF`) should be your **file\_printer\_name**, including any embedded `!`'s.

- If you want to produce PDF compatible only with Acrobat versions 4+ or 5+, change the line

```
call ps2pdf12 %1.ps %1.pdf
```

and change `ps2pdf12` to `ps2pdf13` (Acrobat 4+) or `ps2pdf14` (Acrobat 5+).

– Save `pdfmakegsxx.bat`

- Move the three files `pdfmakegsxx.bat`, `pdfmakegs.bat` and `pdfgs.exe` to some directory in your path.

The command-line system is now set up and ready to test.

## 5.8 Set up the TeXConverter

To set up the TeXConverter to produce PDF using SWP tools:

- Copy `pdfmakegstc.bat` to the TeXConverter's directory.
- Start the TeXConverter. If you have not previously used it, click on the **Help** tab, then on the **Program Locations** button. Make sure that you provide the correct path to GhostScript's `\lib` directory (bottom-right entry).
- Now click on the **PDF** tab and check **Use SWP+GhostScript**. Click **Setup batch files**. Make sure that the entries on the left are correct: note that you may need to change a digit or two for the **SWP TrueTeX Directory**, but the basic structure of this path is correct. Select the printer you have set up for PostScript-to-disk output.
- Check that the entries on the right are correct; typically they will be.
- Click **OK**. The TeXConverter will tell you if it finds a problem with any of your entries; otherwise you'll be told that the batch files have been set up successfully. You're now ready to use it to make PDF.

## 6 Test the system

The SWP distribution contains two `.tex` versions of this document (`pdfgs.tex` and `pdfgs3.tex`), which you can use to test the system: of course, you can use any `.tex` file you have available. To make sure that all paths work correctly, copy `pdfgs.tex` and `pavhpref.sty` to some directory *other* than the one you unzipped the SWP distribution to, and preferably not in your path. If you are using SWP 3.5 or earlier, you should test your setup using `pdfgs3.tex` (not `pdfgs.tex`), since `pdfgs.tex` contains references to a macro, `\msihyperref` which will be unknown to your system.

## 6.1 Command-line system

To test the command-line system:

- Change to the directory containing the source and issue the command `pdfmakegs pdfgs` (`pdfmakegs pdfgs3` for SWP 3.5 or earlier).
- You should see the TrueTeX formatter start, process your file, and close down three times.
- Next, the TrueTeX previewer will open, open the DVI file, print it, and close down.
- After a short pause, the batch file will report that the PDF file has been made.

You can now open `pdfgs.pdf` (`pdfgs3.pdf` for SWP 3.5 or earlier) in the Acrobat Reader, and see how it looks.

## 6.2 TeXConverter

To run the test using Steve Mayer's TeXConverter:

- Start the TeXConverter, click on the PDF tab. You will see a check-box marked Use SWP+Ghostscript, which uses the `pdfmakegstc.bat` you just created: if that file is not found in the TeXConverter's directory, there will be a message to this effect below the check box. (Trying to Convert a document will produce another warning message). If you see this message, copy `pdfmakegstc.bat` to the directory and re-start the TeXConverter.
- Check Use SWP+Ghostscript
- In the left-hand pane, locate the source file. Remember that if you are using SWP 3.5 or earlier, then you want to process `pdfgs3.tex` not `pdfgs.tex`.
- Decide whether you want to view the PDF as soon as it is created (check Run PDF file), and in the right-hand pane, if you want a non-default number of LaTeX passes: the default is 3, which is needed here.
- Click Convert

The TeXConverter runs the batch file (you should see a notification of the SWP version you are using), and you will see the TrueTeX formatter and Previewer start, run, and close as with the command-line system. If you checked Run PDF file, the Acrobat Reader should start, and you should see your document.

## 7 General Usage

The general form of the command-line system for making PDF is

```
pdfmakegs source_file latex_passes [debug [no_run]]
```

where

`source_file` is the name of your `.tex` file, *without extension*.

`latex_passes` (between 0 and 3) is the number of times you want the TrueTeX formatter to run. The default is 3 which is always safe, and if you have a fast computer, the savings with 1 or 2 are rather small. If you have an up-to-date DVI file, you can ask for 0 passes; otherwise, specify the number of passes that SWP itself uses to typeset your document. Note that we do *not* regenerate either the bibliography (`.bbl`) or index (`.ind`) files.

`debug` (optional): any non-empty value will cause `pdfgs.exe` to print some additional information about what's it's doing. Note that the standard Windows NT4/2000 setup does not use `pdfgs.exe`, so those operating systems ignore this parameter.

`no_run` (optional): any non-empty value will have `pdfgs.exe` print some additional information about what it *would* do, but it will not run the batch file `pdfmakegsxx.bat`. This is sometimes useful to make sure that you've not introduced a typo when you edited these files. Note that this parameter requires a non-empty setting for `debug`. Since the standard Windows NT4/2000 setup does not use `pdfgs.exe`, those operating systems ignore this parameter.

## 8 Troubleshooting

**Out of Environment** If you are running Windows 95 or 98 and you see a DOS-session message `Out of environment space`, see section 8.1.

**Formatter Errors** If the TrueTeX formatter stops with an error on a file you can properly typeset in SWP itself and you are using Windows 95/98, you may have forgotten to remove the caret (^) preceding `&latex` (or `&latex_ml`) in `pdfmakegsxx.bat`. Remember that you need to do this three times.

### Previewer Errors

- If the TrueTeX previewer starts and posts a message saying that it Cannot Access Printer, then there is probably a typo in the specification of the `file_printer_name`. Check this carefully; if you opted for a name containing spaces, this may be a good time to reconsider.

- If the Previewer starts and posts a message that it cannot find the DVI file, first check that it does really exist in your directory. Then (Windows 95/98) try again, but this time run

```
pdfmakegs pdfgs 1 a b
```

This turns on the display of debug information and does not actually try to make PDF; you can see what the program thinks it should be doing. For Windows NT4/2000, edit `pdfmakegs.bat` and after the `set` command add `echo %currdrive% %1 %currdir%%1`, and on the next line `goto end`. In either case, let me know what you see.

**Bad PostScript** If the GhostScript step generates error messages (or any output at all):

- If you're using GhostScript 6.50 on Win95/98/ME read the update note dated January 17, 2001 at the top of this file.
- Check the Previewer Errors, above.
- Are you sure your printer is actually producing PostScript? Try using your PostScript printer to print a 1-line file to disk from within SWP. Then open the disk file in a text editor. Near the top you should see a few comment lines beginning `%%Title:...` and `%%Creator: ...`. If you don't see these, then there is probably something wrong with your printer setup.

**Seeing what's happening** You can temporarily change `@echo off` at the beginning of the batch files to `@echo on` and add a `pause` towards the end (or at any other strategic location). This should allow you to diagnose just where the problem is occurring.

## 8.1 Environment size under Windows 95/98/ME

Under Windows 95/98 you can run into a problem connected with the fixed size that the operating systems reserve for strings, such as your `PATH` variable (this will never happen under NT). If you see a message to the effect that it is "Out of Environment Space" you will need to run the PDF generation setup in a DOS session with a larger environment. You can do this as follows:

- Start a *windowed* DOS session
- Click the icon at the top-left of the session window, and select **Properties**.
- Under **Memory**, change the entry under **Environment Size** from **Auto** to (say) **1024**. If it is already set to something other than **Auto**, select the next larger setting. Click **Apply**.
- Close the window.



When you next open it, the Environment size will be larger, and your problems should be gone. If they're not, repeat the procedure, choosing a larger value.

Note that this is specific to the icon you have just invoked. If you start a DOS session using a different icon, the larger environment size will not be present.

## 9 Drivers

One persistent question is, which printer drivers work for which setups? This section is an attempt to summarize what we know. *It is totally dependent on users telling me what works and what doesn't*, so if you have something to contribute, please let me know. I need to know (a) your operating system; (b) the name of the printer, and if possible, the driver version number; (c) where you got the driver from, especially if it isn't a standard driver supplied on the Windows CD and (d) whether it works or not, and if not, what problems you've noticed.

Under NT4 at least you can get version information by opening the Printers folder (Start -> Settings -> Printers) selecting the printer and then right-clicking and selecting Properties. In the resulting dialog, select the Device Settings tab. Finally, highlight the name of the printer (typically the topmost item displayed) and an About button becomes available. Click it to get information.

In the Windows 95 and 98 listings, "mini test" results are based on a test I performed on a 2-line document to check (a) that ligatures were created correctly; (b) that Greek letters in math rendered properly; and (c) that a few elementary math constructs like fractions, roots and integrals rendered correctly. I did not check color even for color printers. In all cases, the print driver was from the Windows CD.

### 9.1 Windows 95

#### 9.1.1 Working drivers

- Apple LaserWriter [mini test]
- Apple LaserWriter Personal NT [mini test]
- Hewlett Packard LaserJet 4/4M Plus PS [mini test]
- Hewlett Packard LaserJet 4/4M PostScript [mini test]
- Hewlett Packard LaserJet 4P/4MP PostScript [mini test]
- Kodak ColorEase PostScript Printer [mini test]

## 9.2 Windows 98

### 9.2.1 Working drivers

- DPM2000. Driver available from ?? and requires PPD file from ?? . Supports color.
- Hewlett-Packard Laserjet 4/4M Plus PS version 4.00. Standard Win 98 driver; may not even need the CD for installation. B&W. Uses HP-PLUS4.PPD and PSCRIPT.DRV (version 4.10.1998).
- Apple Color LaserWriter 12/600 [mini test]
- Hewlett Packard LaserJet 4/4M Plus PS [mini test]
- Hewlett Packard LaserJet 4/4M PostScript [mini test]
- Hewlett Packard LaserJet 4P/4MP PostScript [mini test]
- Hewlett Packard LaserJet 6P/6MP PostScript [mini test]
- Hewlett Packard Color LaserJet 5/5M PS [mini test]

### 9.2.2 Problematic drivers

- Hewlett Packard Color LaserJet 5/5M. Driver version unknown; downloaded from HP website. Color. This is reported to produce bitmaps for ligatures, and hence blocky output at low resolutions. But see above for my mini-test - I wonder if this is because it was incorrectly configured?

## 9.3 Windows ME

### 9.3.1 Working drivers

## 9.4 Window NT4

### 9.4.1 Working drivers

- Hewlett-Packard LaserJet 4M/4M Plus PS 600. Driver version: 4.00. Standard driver supplied with NT4. B&W.
- Hewlett Packard Color LaserJet 5/5M. Driver version unknown; downloaded from HP website. Color.

## 9.5 Windows 2000

### 9.5.1 Working drivers

- Hewlett-Packard LaserJet 4M/4M Plus PS 600. Driver version: 4.00. Standard driver supplied with NT4. B&W.

## 10 Upgrading

Here is what you will need to change if you upgrade any of the components of this setup.

**GhostScript** Here you must:

- Edit `ps2pdfxx.bat` in GhostScript's `\lib` directory: see section 5.5, where the instructions differ as between versions 6.50 and earlier versions. Alternatively, if you have the TeXConverter, you can re-do the **Setup batch files** procedure, as described in section 5.8.
- Edit `pdfmakegsxx.bat` (Windows 95/98/ME) or `pdfmakegs.bat` (Windows NT4/2000) and `pdfmakegstc.bat` (if you are using the TeXConverter) and change the line beginning `set gsdir=` to point to the new top-level location of GhostScript.

**SWP** Edit `pdfmakegsxx.bat` (Windows 95/98/ME) or `pdfmakegs.bat` (Windows NT4/2000) and change the line beginning `set swppath=` to point to the new TrueTeX directory. If you are using the TeXConverter, re-run the **Setup batch files** procedure, as described in section 5.8.

**OS** If you change from Windows 95/98/ME to Windows NT4/2000 you must:

- Re-install a print-to-file PostScript printer
- Edit `pdfmakegsxx.bat` (Windows 95/98/ME) or `pdfmakegs.bat` (Windows NT4/2000) and add a caret (^) before `&latex` or `&latex_ml`. Remember to do this three times. If you are using the TeXConverter, re-install it, re-configure the **Help -> Program Locations** setup, and then re-run the **Setup batch files** procedure under the PDF tab, as described in section 5.8.

If you change from Windows 95 to Windows 98 or to Windows ME you must re-install a print-to-file PostScript printer.

## 11 Conclusion

Good luck! Let me know at [viton.1@osu.edu](mailto:viton.1@osu.edu) if this is helpful. If you use the TeXConverter, I'm sure that Steve Mayer, at [mayer@dial.pipex.com](mailto:mayer@dial.pipex.com) would appreciate hearing from you, too.