

OzT_EX User Guide

Version 1.4, January 1992

Andrew Trevorrow
c/- Research School of Earth Sciences
Australian National University
GPO Box 4, Canberra 2601, Australia

E-mail: akt150@huxley.anu.edu.au

OzT_EX is a public domain version of T_EX for the Macintosh. This document assumes you know how to use a Macintosh. It also assumes you are familiar with T_EX, the typesetting system developed by Donald Knuth at Stanford University. An understanding of PostScript, the page description language developed by Adobe Systems, is not essential but would be useful.

OzT_EX aims to provide a standard T_EX environment for the Macintosh that can easily be extended or customized to suit your particular needs. If you have had experience with T_EX on some other computer then the way OzT_EX works shouldn't be too surprising.

Comments, bug reports and suggestions are all welcome. I have plans for further development of OzT_EX, but at this stage such work can only be carried out in my spare time. If anybody out there would like to sponsor such work then please get in touch.

Contents

1	Introduction	1
1.1	Changing the default configuration file	2
1.2	Creating other configuration files	3
2	A quick tour of OzTeX	3
2.1	Starting OzTeX	3
2.2	Editing a TeX input file	3
2.3	Running TeX	3
2.4	Viewing a DVI file	4
2.4.1	Paper coordinates and the scale factor	4
2.4.2	Zooming in and out	5
2.4.3	Scrolling around	5
2.4.4	Checking for errors	6
2.4.5	Selecting pages	6
2.5	Printing a DVI file	6
2.5.1	Changing print options	6
2.5.2	The printing process	7
2.5.3	Saving output in a file	7
2.5.4	Positioning pages correctly	7
3	Fonts	7
3.1	TFM files	7
3.2	PK files	8
3.3	PostScript fonts	9
3.4	Macintosh screen fonts	10
3.5	Transferring TFM/PK/DVI files	10
4	OzTeX's menus	11
4.1	The File menu	11
4.2	The Edit menu	12
4.3	The TeX menu	12
4.4	The View menu	13
4.5	The Config menu	15
4.6	The Help menu	16
5	Including graphic files in TeX	16
5.1	Including Macintosh graphic files	17
5.2	Including PostScript files	19
6	Recovering from errors	19
6.1	Errors when running TeX	20
6.2	Errors when viewing a DVI file	20
6.3	Errors when printing a DVI file	21
7	Further information	22

1 Introduction

Assuming you correctly carried out the installation instructions in the **Read-Me** file supplied with OzTeX, your screen should display something like this:

Here is a brief description of the various folders:

- The **Configs** folder contains a number of configuration files.
- The **TeX-fonts** folder contains a standard set of TFM files. These files are required by TeX to carry out typesetting; see section 3.1 for more information about TFM files.
- The **TeX-formats** folder contains some format files used by TeX and the **TeX.pool** file read by INITEX. Section 4.3 describes how to create format files.
- The **TeX-inputs** folder contains a large number of TeX input files. TeX looks in this folder if it can't find a given file in your current folder.
- The **Help-files** folder contains a number of text files.
- The **PS-files** folder contains various PostScript files.
- The **TeX-docs** folder contains examples of TeX input files, including **nasty.tex** which is used in the quick tour of OzTeX.
- The **LaTeX-docs** folder contains the source files for this User Guide.
- The **ΣEdit** folder contains the ΣEdit desk accessory and its documentation.
- The **DVIM72-Mac** folder contains the DVIM72-Mac program and its documentation.
- The **PK-files** folder contains PK files stored in various sub-folders. The name of a sub-folder indicates the “size” of the PK files kept within it; see section 3.2 for how this size is calculated and for more information about PK files.

1.1 Changing the default configuration file

OzTeX reads a configuration file called **Default** when starting up. This file is kept in the **Configs** folder along with other configuration files. (The name and location of the default configuration file are stored as STR resources in the application file, so you can easily change them using a resource editor like ResEdit.)

A configuration file is a simple text file which you can edit and change various parameters that control OzTeX's behaviour. Some of the more important parameters are:

- A list of the configuration files that will appear in the Config menu. This list should only appear in the **Default** file; it will be ignored in every other configuration file.
- A list of the formats that will appear in the TeX menu. Place your preferred format first.
- A list of the help files that will appear in the Help menu. Feel free to add your own help files or remove them entirely (the Help menu won't appear if the list is empty).
- The printer resolution and viewing resolution. The former is used to print a DVI file and the latter for previewing. If you have a set of 72 dpi PK files then you could set the viewing resolution to 72; this value matches the resolution of the standard Mac screen, resulting in a more readable display (it's also faster and uses much less memory).
- The paper dimensions used to detect page-off-paper errors when printing/viewing a DVI file. If the paper width is greater than the height then OzTeX will print/view in landscape mode. The **Default** values specify A4 paper in portrait mode.
- The names of certain special folders: **TeX-formats**, **TeX-inputs**, **TeX-fonts**, **Help-files**, **PS-files** and **PK-files**. The supplied strings are partial path names relative to the location of the OzTeX application, but you might prefer to store certain files elsewhere. For example, you could tell OzTeX to look for PK files on a different volume by changing the path name of the PK folder to "MyDisk:PK-files:".
- The names and locations of special files: **TEXTtoPS.ps**, **DVIttoPS.ps**, **TeX.pool**, etc.
- The colours of various elements seen when viewing a DVI file (paper edges, `\special` markers and missing fonts). The allowed colours are black, red, green, blue, cyan, magenta, yellow and white (white is allowed in case you want to do something like make `\special` markers invisible).
- The values of important TeX parameters that determine how much memory will be allocated for certain arrays. The **Default** values are suitable for a Mac with 1MB of memory. If you have more memory then you may want to increase many of the parameters. If you do then you'll need to increase OzTeX's application memory size for use with MultiFinder or System 7 (see the "Get Info" item in the Finder's File menu). Note also that a change to **mem_top**, **hash_size**, **hash_prime** or **hyf_size** will require the format file(s) you use to be rebuilt (see section 4.3).
- A list of all PostScript TFM file names and their corresponding printer-resident font names and screen font names. Each screen font has a particular encoding scheme (Adobe or Macintosh) and an optional style.

The **Default** configuration file contains many comments describing the purpose of each parameter and how to go about changing them. Before doing so it might be a good idea to save a copy first. In your new **Default** file you can then remove all the comments so that OzTeX starts up a little faster.

1.2 Creating other configuration files

OzTeX's Config menu lets you switch rapidly from one configuration file to another. This is especially useful if you have access to more than one PostScript printer, since each printer might require a separate configuration file (they might have different resolutions or support a different set of resident fonts).

There are other uses for multiple configuration files. See the examples supplied in the Configs folder. It is a simple matter to create your own and add them to the list in the **Default** file. To help keep configuration files short and simple, a couple of special characters can be used:

1. If “?” is the first character of a parameter's value then the parameter is not changed and the rest of the line is ignored.
2. “!” is similar but in addition all remaining parameters are left unchanged (the rest of the file is ignored).

Note that these characters are not recognized as special in the **Default** file since every parameter must be given a value (even if empty).

2 A quick tour of OzTeX

The aim of this section is to get you quickly acquainted with OzTeX's major features.

2.1 Starting OzTeX

Assuming you've installed everything, double-click on the OzTeX application file to get the program started. After a brief pause you should see a window with the title “OzTeX” and a message indicating the name of the default configuration file. (If you made a mistake during installation, especially when editing the **Default** file, then you might also be looking at a dialog box from which you can only quit. The message displayed will hopefully give you some idea of how to fix the problem.) The OzTeX window is always present; note that it has no close box in the upper left corner. This window provides a terminal-like interface for OzTeX.

The production of a TeX document typically involves a number of iterations through the following cycle: edit, typeset, preview and print. (People concerned about our dwindling forests will hopefully preview many more times than they print.) Let's go through this cycle step by step.

2.2 Editing a TeX input file

A TeX input file is a standard Macintosh text file. We won't create an input file from scratch, instead we'll use one of the sample files provided in the **TeX-docs** folder. There is nothing special about this folder, or **LaTeX-docs**. I simply like to keep my TeX and LaTeX input files in separate folders. You can keep your input files anywhere you like.

The file we'll use is called **nasty.tex**. As the name suggests, it is not a typical TeX input file, but it does illustrate most of the things you'll encounter when using OzTeX in the future.

Use any text editor to open **nasty.tex** in the **TeX-docs** folder. The only change we'll make to **nasty.tex** is to add a deliberate error: insert the illegal command `\xxx` at the start of the file. Save this change before switching back to OzTeX.

2.3 Running TeX

Let's now typeset **nasty.tex**. Before starting TeX we need to make sure that the correct format will be used. All the available formats appear at the end of the TeX menu. Check this menu

and see which format item is ticked. It should be “Plain”, unless you changed the format list in the `Default` configuration file. This is the format required by `nasty.tex` so we don’t need to change it. (To change the current format you simply select a different one.)

Choose “ \TeX ...” from the \TeX menu and open `nasty.tex`. The Oz \TeX window will be cleared and brought to the front (if it isn’t already) and after a short pause \TeX will start up, load the Plain format, and begin reading the given input file.

When it sees the unknown command, \TeX will print a suitable error message, beep, and wait for you to type something. Note that a solid block cursor sits next to \TeX ’s “?” prompt. This block cursor always appears when \TeX is waiting for you to type something. The Oz \TeX window should look like this:

```
This is TeX, Version 3.14 (no format preloaded)
**&Plain nasty.tex
(nasty.tex
! Undefined control sequence.
1.5 \xxx

? █
```

Chapter 6 in *The \TeX book* explains what you can do in such a situation. What we will do is simply ignore the error and continue by hitting the Return key. Although `nasty.tex` is not very nice, you shouldn’t see any more \TeX errors. A 19-page DVI file called `nasty.dvi` should be created.

Have a look at the second item in both the File and View menus. Whenever a new DVI file is created its name will automatically appear in these items so you can quickly print or view the DVI file without going through the standard file dialog. The second item in the \TeX menu has also been updated with the name of the most recent input file (`nasty.tex` in this case) so you can easily typeset the same file many times.

2.4 Viewing a DVI file

After generating a DVI file, the next step is to preview it and check for problems that \TeX may have missed, like missing fonts, bad page breaks, spelling mistakes, etc.

Let’s have a look at `nasty.dvi` by choosing the second item in the View menu. Oz \TeX creates a new window, called the “view” window, and sets its title to the selected DVI file. A dialog box has also appeared. It has a scroll bar that allows you to select any page in the DVI file. (The dialog box also allows you to change a couple of viewing options, but we’ll ignore them at this stage; see the description of “View DVI...” in section 4.4 for details.) Play around with the scroll bar if you like but make sure it shows the 2nd DVI page before proceeding. To view this page simply click in the default View button or hit the Return key.

The 2nd page in `nasty.dvi` contains a single rule with some text below it. Dotted lines representing the paper edges should also be visible. Oz \TeX chooses the initial view based on the current size and shape of the view window and the current paper dimensions. It will try to show as much of the paper as possible. The top, left and right paper edges should all be visible (and maybe the bottom edge as well if you have a large screen).

Now is a good time to adjust the size and location of the Oz \TeX and view windows to suit your screen. When you quit, Oz \TeX will remember the current settings and use them the next time it starts up.

2.4.1 Paper coordinates and the scale factor

An understanding of the coordinate system used by Oz \TeX to display (and print) a DVI page would be useful. You may have noticed the “px” units mentioned in the `Default` configuration

file. These are “paper pixels” and are \OzTeX ’s internal units. The resolution parameters in a configuration file define the number of paper pixels per inch; one parameter is only used when printing a DVI file and the other is only used when viewing a DVI file.

\OzTeX uses a coordinate scheme in which the paper pixel at (0,0) is exactly one inch in from the top and left edges of the paper. This position is referred to as the “ \TeX origin” since it is also the origin of the coordinate system used in DVI files. Every rule/character/ \backslash special on a page has a specific location defined by a pair of paper pixels (h,v). Vertical coordinates increase down the paper and horizontal coordinates increase to the right.

If the view window is frontmost then the cursor is changed to a cross whenever it moves over the contents region. The current position of the cross is shown (in paper coordinates) in a box at the lower left corner of the view window. You can click in this box to change units.

A “scale factor” is used to display a DVI page at a particular size. It defines the number of paper pixels in each Macintosh screen pixel (both horizontally and vertically) and always has an integer value greater than or equal to 1.

2.4.2 Zooming in and out

Certain View menu items change the current scale factor:

- “Full View” sets the scale factor to its maximum value and displays the entire page and paper edges in the middle of the view window.
- “Actual Size” sets the scale factor so that the new view will show the page at roughly the right size (it’s only approximate because the scale factor is an integer value).
- “Zoom In” halves the current scale factor.
- “Zoom Out” doubles the current scale factor.

You can also change the scale factor by clicking in the view window in a variety of ways:

- Click-and-drag within the viewing area to zoom in to the selected rectangle. Cancel the operation by making the rectangle very thin (not small, because that may be interpreted as a simple click).
- A simple click will zoom in by halving the scale factor.
- Command-click will zoom in by decrementing the scale factor.
- Option-click will zoom out by incrementing the scale factor.
- Shift-click will zoom out by doubling the scale factor.
- A beep occurs if you can’t zoom in or out any further.

When you zoom in or out by clicking, \OzTeX will try to make the point you clicked the middle of the new view. However, when you choose a menu item, \OzTeX tries to keep the top left page location fixed. In both cases it is possible that the view may unexpectedly shift so that it remains within the scrolling limits set by \OzTeX . Experiment with the various ways of zooming to see which methods you prefer.

The most precise display occurs when the scale factor is 1 since each screen pixel corresponds to exactly one paper pixel. \OzTeX doesn’t allow you to zoom in any further than this.

2.4.3 Scrolling around

The view window has scroll bars that allow you to move over the page in the standard Macintosh manner. Note that the arrow keys can be used instead of clicking in the scroll arrows. \OzTeX won’t let you get too far away from the page/paper boundaries. If you do manage to get lost just choose “Full View”.

2.4.4 Checking for errors

It is a good idea to select the “Page Info” item at least once while viewing a DVI file. This item displays its results in the OzTeX window. The display includes a list of all the fonts used in the DVI file and clearly indicates any that are missing. It also lists any `\special` commands on the current page, showing their locations and arguments.

2.4.5 Selecting pages

There are a number of interesting pages in `nasty.dvi`. Use “Previous Page”, “Next Page” or “Select Page...” to have a look at some of them. Note that the current DVI/TeX page numbers are always displayed in the view window’s title. Here are some points of interest:

- The current scale factor and page location will only change if the new page is off the paper, in which case OzTeX will beep and display a full view. Pages 13 to 15 illustrate this behaviour. When you move from such a page to a normal page, OzTeX reverts back to the default scale and location.
- OzTeX may take a little while to interpret and display a selected page, depending on how complicated it is. Hit Command-C or Command-Dot if you get bored.
- The view window is updated in the following manner. Visible paper edges are drawn first, then `\special` bitmaps and markers, then rules, then characters on a font by font basis, starting with the font that has the least number of characters on the page. Try a full view of page 19.
- Page 3 has examples of `\special` commands. OzTeX is able to preview included files of type PICT, PNTG or EPSF. The location of each `\special` is indicated with a small marker. Note that the size of this marker does not change as you zoom in or out.
- Page 9 uses a couple of PostScript fonts.
- All the fonts used on page 12 are deliberately missing.

When you have finished previewing the DVI file just click in the view window’s close box. The view window is also closed automatically when you run TeX, print a DVI file or read a configuration file.

2.5 Printing a DVI file

To print `nasty.dvi` choose either of the top two items in the File menu. Choosing the second item simply avoids the standard file dialog.

2.5.1 Changing print options

OzTeX will present a dialog box that lets you to change a variety of print options. Most of them should be fairly obvious — details can be found in the description of “Print DVI...” in section 4.1. The option you’ll most often want to change is the page range. (If you find yourself changing a check box option nearly every time then you should probably change its default setting in the `Default` configuration file.) The two scroll bars control the first and final pages. OzTeX prevents you from choosing a first page greater than the final page. Play around with the scroll bars but restore their values so that every page will be printed. Click in the Print button or hit Return.

2.5.2 The printing process

OzTeX will look for the printer most recently selected by the Chooser. If this isn't a PostScript printer then you won't get very far. If the printer is found then a status box will appear and keep you up-to-date on your job's progress and the state of the printer. As each page is translated into PostScript its DVI/TeX page numbers are displayed in the OzTeX window.

You can't actually print every page in `nasty.dvi` because there is a deliberate error that makes this impossible. The first 11 pages should not cause any problems (assuming you're using A4 paper) but the rest will generate nearly every type of error message you're ever likely to encounter. You should eventually get up to page 17 where a deliberate PostScript error in a `\special` file will prevent any further pages being printed. (OzTeX may start translating page 18 before the error is seen.)

2.5.3 Saving output in a file

The print dialog box also lets you send the PostScript output to a file instead of the printer. This file is an ordinary text file with a default name of `Oz.ps`. A warning: the PostScript code generated by OzTeX is resolution-dependent. If you send this code to a PostScript device with a resolution different to that of the current printer resolution parameter then the output will look a little strange! (Use "Show Config" to see the current value of the printer resolution parameter.)

2.5.4 Positioning pages correctly

When printing a DVI file, OzTeX changes the PostScript coordinate system to match the paper coordinate system described in section 2.4.1. For your output to appear in precisely the right location on the paper the TeX origin must be exactly one inch (2.54 cm) in from the top and left paper edges. OzTeX can't always set the TeX origin automatically to the correct position because some PostScript printers don't provide an accurate way of finding out the exact paper dimensions and the relative location of the PostScript origin.

Page 2 in `nasty.dvi` contains a thick rule with its top left corner exactly at the TeX origin. If the printed page shows a significant discrepancy then change the horizontal and vertical offsets in a configuration file to shift the origin to the correct position. If you use more than one printer then you may need a separate configuration file for each.

Choose offset values based on the normal portrait paper orientation. The offsets are adjusted automatically so that the landscape origin should also be correct (if it isn't then you'll have to create a separate configuration file for landscape printing, or if you know a bit of PostScript then alter some numbers in `DVItoPS.ps`; search for the string "TeX origin" in that file).

3 Fonts

OzTeX uses the following sources of font information in the process of typesetting, previewing and printing a document: TFM files (usually stored in `TeX-fonts`), PK files (usually stored in various folders nested within `PK-files`), and Macintosh screen fonts (normally found in the System file).

3.1 TFM files

A TeX Font Metric file contains the crucial typesetting information about a font, such as each character's height, depth and width. The actual character images are not stored in a TFM file (that information is kept in PK files, or, in the case of a PostScript font, inside the printer). TFM files are the only source of font information used when running TeX, since it doesn't need to know anything about character images to be able to create a DVI file. See *The TeXbook* by

Donald Knuth if you want to know more about how \TeX uses fonts, especially Chapter 4 and Appendix F.

\OzTeX also reads TFM files when printing or viewing a DVI file containing PostScript fonts (based on the list of TFM names at the end of your configuration file), or if a requested PK file can't be found (in which case the dummy TFM file specified in your configuration file will be used).

\OzTeX will look for a TFM file in the current folder before looking in the default **TeX-fonts** folder. Most of the TFM file names in **TeX-fonts** begin with “cm”. These letters stand for Computer Modern, a family of fonts created by Donald Knuth using METAFONT. All the PostScript TFM files supplied with \OzTeX have names beginning with “ps-”; this is not required but does make them easy to recognize. If you want to change their names then remember to update your configuration file(s).

3.2 PK files

PK (packed pixel) files store the character images needed to print or view a DVI file containing non-PostScript fonts (those NOT appearing in the list at the end of your configuration file). For each such font there is usually a number of PK files, each one representing the same font but at a different size. This size, which has no relation to the font's design size, is calculated as follows:

$$\text{size} = \text{resolution} \times \text{magnification}$$

where *resolution* is either the printer resolution or the viewing resolution, and *magnification* is the overall font magnification (i.e., the DVI magnification times the individual font scaling). The *size* is then rounded up to the nearest integer and should equal (± 1) one of the folder names in **PK-files** if you decide to keep PK files of the same size in separate folders.

Although \TeX allows you to request a font at virtually any magnification, it is obviously impossible to provide an infinite number of PK files. The compromise solution is to provide a range of sizes for each font. These sizes are in a geometric ratio based on powers of 1.2 and correspond to \TeX 's `\magstep` values. See Chapters 4 and 10 of *The \TeX book* for more details. For example, assuming a *resolution* of 300 and no document magnification:

\TeX input	<i>size</i>	PK file
<code>\font\ra=cmr10</code>	300×1.2^0	<code>:PK-files:300:cmr10</code>
<code>\font\rb=cmr10 scaled\magstephalf</code>	$300 \times 1.2^{0.5}$	<code>:PK-files:329:cmr10</code>
<code>\font\rc=cmr10 scaled\magstep1</code>	300×1.2^1	<code>:PK-files:360:cmr10</code>
<code>\font\rd=cmr10 scaled\magstep2</code>	300×1.2^2	<code>:PK-files:432:cmr10</code>
<code>\font\re=cmr10 scaled\magstep3</code>	300×1.2^3	<code>:PK-files:518:cmr10</code>
<code>\font\rf=cmr10 scaled\magstep4</code>	300×1.2^4	<code>:PK-files:622:cmr10</code>
<code>\font\rg=cmr10 scaled\magstep5</code>	300×1.2^5	<code>:PK-files:746:cmr10</code>

\TeX 's `\magnification` command has a cumulative effect on font scaling:

```

\magnification=\magstep1           % document magnification = 1.2
\font\bigrm=cmr10 scaled\magstep2 % font magnification = 1.44

```

The font size is now $300 \times 1.2 \times 1.44 = 518.4$, so \OzTeX will use `:PK-files:518:cmr10`.

For compatibility with other \TeX systems, \OzTeX actually uses a more complicated search scheme. If `foo` is a PostScript font then \OzTeX looks for `foo.tfm` in the current folder before looking in `TeX-fonts`. If `foo` is NOT a PostScript font then \OzTeX looks for a PK file at the requested size. For example, if the calculated size is 300 then \OzTeX will look for:

1. `foo.300pk` (in the current folder)
2. `:PK-files:foo.300pk` (relative to the location of \OzTeX)
3. `:PK-files:300:foo.300pk` (ditto)
4. `:PK-files:300:foo` (ditto)

In fact, at each step \OzTeX will also add and subtract 1 from 300 because rounding errors can occur in the size calculation. If still not found, \OzTeX gives up and displays a warning.

There is still another possibility to consider: a font name can include an explicit location. For example, your \TeX input file might contain `\font\xxx=:myfonts:foo`. In this case \TeX will look for `:myfonts:foo.tfm` in the current folder; that is, it expects to find a file called `foo.tfm` in a sub-folder called `myfonts`. If not found then it gives up; it does NOT look for `foo.tfm` in the default `TeX-fonts` folder. Similarly, when you print or view the DVI file, \OzTeX uses the same search algorithm described above but does NOT look in the current folder. Note that the use of an explicit font location is not recommended as it decreases the portability of your \TeX input file.

`\magstep` values can also be used with \LaTeX 's `\newfont` command but you won't need to do this very often because \LaTeX provides a large number of predefined font styles and sizes. Note that \LaTeX has no `\magnification` command — use the `11pt` and `12pt` style options to increase the size of all fonts in a \LaTeX document.

All the PK files supplied with \OzTeX have been specifically generated for a 300 dpi, write-black laser printer (like the Apple LaserWriter). If your printer doesn't have the same characteristics then you'll need to get the correct PK files from some other source. If you use \TeX on another computer then you might be familiar with GF or PXL files. PK files contain the same information but in a highly compressed format. Most mainframe \TeX systems provide tools for converting GF or PXL files into PK files.

3.3 PostScript fonts

To be able to use a PostScript font in a \TeX document, \OzTeX needs to know the name of its corresponding TFM file. This information is specified at the end of a configuration file (see the `Default` file).

Each PostScript printer has a certain set of resident fonts. The font you wish to use may not exist in the current printer. To see an alphabetical list of the PostScript fonts available, choose “Send PostScript...” from the File menu, open the `PS-files` folder, and send the file `getfonts.ps`. The list should appear in the \OzTeX window. If you'd prefer to print the list then send `fontlist.ps`.

There are some important differences between PostScript fonts and the Computer Modern fonts created specifically for use with \TeX :

1. A PostScript font can be requested at virtually any size you like since there are no corresponding PK files to worry about. When asking for a PostScript font it is best to use an “at *dimen*” size specification (e.g., `\font\helv=ps-helv at 14pt`) otherwise \TeX will use the design size stored in the TFM file. All the PostScript TFMs supplied with \OzTeX have a design size of 10 pt but this may not be true for other TFMs you might acquire.
2. PostScript fonts usually contain up to 256 characters, with many empty locations. Computer Modern fonts can also contain up to 256 characters but typically use only the first 128 locations (with very few empty).

3. When a PostScript font is scaled to the same design size as a CM font it tends to look darker and larger. You need to be a bit careful about mixing the two font designs.
4. The standard \TeX commands for accents and foreign letters need to be redefined for a PostScript text font. The file `pstext.tex` in the `TeX-inputs` folder contains the required macro definitions. Another file, `psfonts.tex` in `TeX-docs`, inputs these macros and illustrates the use (and abuse) of PostScript fonts in a \TeX document.

If you are a \LaTeX user and would like a document typeset with PostScript text fonts rather than Computer Modern, then try the Times- \LaTeX format supplied with Oz \TeX . With a bit of luck you might not have to make any changes to your input file. Note that CM fonts are still used to typeset mathematics.

3.4 Macintosh screen fonts

When previewing a DVI file containing PostScript fonts or missing fonts, Oz \TeX uses Macintosh screen fonts to simulate the characters. Just which screen fonts are used depends on the information in a configuration file — see the comments at the bottom of the `Default` file.

If you use PostScript fonts a lot then for best previewing results you should get the matching Adobe screen fonts listed in the `Default` configuration file. If you don't have matching screen fonts then there is a good chance that the simulated characters will appear incorrect, especially if they are non-alphabetic. The same problem occurs for simulated characters from missing fonts.

Text in the Oz \TeX window is always displayed using 9 pt Monaco.

3.5 Transferring TFM/PK/DVI files

Oz \TeX reads standard TFM and PK files, and reads and writes standard DVI files. If you have access to \TeX on some other computer system, particularly a VAX/VMS or UNIX mainframe, you should be able to move such files to and fro without any further processing.

For the purposes of data transmission a TFM/PK/DVI file must be treated as a binary file (a stream of arbitrary 8-bit bytes). When using Kermit to transfer binary files from another computer to your Mac, remember to type “`set file type binary`” before using the `send` command. The same goes for receiving such files from your Mac, but with one exception: to transfer a DVI file created by Oz \TeX to a VAX/VMS host you'll need to type “`set file type fixed`” before using the `receive` command.

If you want to print or view a DVI file created by another \TeX system then hold down the Option key while selecting “Print DVI...” or “View DVI...”. All file types will then be available for selection. If the one you choose is not a valid DVI file then a warning message will appear complaining about a “bad id byte”.

Note that a DVI file contains TFM file names; this can be a cause for concern if you plan to transfer DVI files from one \TeX system to another with a different set of TFM files. This incompatibility is most likely to occur if you use PostScript fonts, since there is little agreement on how their corresponding TFM files should be named. If you have a choice, it's always safer to transfer the original \TeX input file so you can edit any incorrect font names.

4 OzTeX's menus

Let's go through each menu and discuss each item in detail, concentrating on the things not mentioned in the quick tour.

4.1 The File menu

Print DVI...

Prints a selected DVI file. After selecting the DVI file you'll be presented with a dialog box that lets you change various printing options:

- The page range. The first and final pages are controlled by separate scroll bars; their current values are displayed in the form "DVI-page/[TeX-page]".
- The paper orientation (landscape or portrait).
- The page order (reverse or normal).
- Printer memory management (conserve VM or not). VM is PostScript's "virtual memory" and the output generated by OzTeX can consume an awful lot of it, especially if your document uses a large number of non-PostScript fonts. If you get a "VMerror" when printing a document then try again with the conserve VM option checked. You should only need to do this for a very unusual document, like a font catalog.
- The paper source (manual feed or normal input tray).
- Whether or not to show statistics about font/rule/\special usage.
- The number of copies of each page.
- The page increment. The primary purpose of this option is to simplify the printing of a document on both sides of the paper. If the first page is 1 and an increment of 2 is chosen then OzTeX will print pages 1, 3, 5, 7, etc. Depending on how your printer stacks the output, you can then put these pages back into the input tray and print the DVI file again using the same increment but starting from the second DVI page (and reversing the page order; this will not effect *which* pages are printed but simply changes the *order* in which they are done).
- The DVI magnification (expressed as an integer 1000 times the desired magnification). You should use the default value unless you know what you're doing. If the DVI file uses non-PostScript fonts then you should probably stick to the numbers corresponding to TeX's \magstep values; i.e., 1000, 1095, 1200, 1440, etc. See Chapters 4 and 10 in *The TeXbook* for details. A document using only PostScript fonts can be printed at almost any magnification (OzTeX allows values from 1 to 10000).

What happens next depends on which dialog button you select. The default Print button will send the PostScript output to the current printer. A status box will appear and indicate what is happening to your job and warn you about any printer problems. Instead, you might prefer to Send the output to a given text file. Any change to a check box option will be remembered the next time you print or view a DVI file, unless you decide to Cancel the dialog.

OzTeX will look for a file called `global.ps` in the current folder and include its contents at the end of the PostScript prolog. For example, `global.ps` in the `PS-files` folder shows how you can get the word "DRAFT" printed in the background on every page.

The DVI/TeX page numbers are displayed as each page is translated into PostScript. Any error messages or statistics apply to the most recently displayed page numbers. (This is not true for PostScript messages sent back by the printer as these occur asynchronously.) Printing can be cancelled by hitting Command-C or Command-Dot.

Print ?.dvi

Prints the indicated DVI file. Exactly the same dialog box described above will appear. This item is initially disabled; its name changes whenever you select a DVI file via “Print DVI...” or “View DVI...”, or create a DVI file by running \TeX . You can print the indicated DVI file even after moving to another folder.

Save Oz \TeX ...

Saves the contents of the Oz \TeX window in a given text file. The default name is `Oz.text`. This item is disabled if the Oz \TeX window is empty.

Print Text...

Prints a selected text file. This file is sent to the current printer after prefixing the `TEXTtoPS.ps` prolog file kept in `PS-files`. If you know a little about PostScript then you might like to modify `TEXTtoPS.ps` to suit your own needs.

Send PostScript...

Sends a selected text file, presumably a PostScript program, to the current printer. The `PS-files` folder contains a number of interesting files that can be sent to almost any PostScript printer. If a file is sent to spooling software rather than directly to the printer then you probably won't see any messages sent back. In such an environment it's a good idea to send `errhandler.ps` first so that any PostScript errors will be printed on paper.

Quit

Quits Oz \TeX .

4.2 The Edit menu

Oz \TeX does not have an integrated editor, so the items in this menu remain disabled until you switch to your editor.

4.3 The \TeX menu

\TeX ...

Typesets a selected text file using the current format (so make sure the correct format is ticked before choosing this item). The Oz \TeX window acts like a terminal during a \TeX session.

One of the nicest things about \TeX is that it behaves the same way on a large range of different computer systems. Virtually everything you read in *The \TeX book* will apply to the version of \TeX run by Oz \TeX (and the same goes for Leslie Lamport's \LaTeX book if you use that format). Only a few special features have been added in Oz \TeX :

- You can interrupt a \TeX session at any time by typing Command-C or Command-Dot. Depending on what it is currently doing, \TeX usually responds immediately with a suitable message and the “?” prompt. If you hit Command-C or Command-Dot at this stage (or whenever the block cursor is visible) then \TeX will immediately abort.
- Most of \TeX 's capacity parameters are set at run-time in a configuration file rather than at compile-time. If you get a “ \TeX capacity exceeded” error then you might be able to overcome the problem by increasing the appropriate parameter.

T_EX ? .tex

Typesets the indicated input file using the current format. This item is initially disabled; its name changes whenever you choose “T_EX...”. You can typeset the indicated file even after moving to another folder.

INITEX

Runs **INITEX**, a special version of T_EX normally used to create format files. This is something you may need to do if you edit a configuration file and change certain T_EX parameters to suit your typesetting needs or your Mac’s memory capacity. Creating a format file is easy. Here are the steps used to create **Plain.fmt**:

1. Run **INITEX**, wait for the “**” prompt, type “**Plain\dump**” and hit the Return key. T_EX will begin reading **plain.tex** from the **TeX-inputs** folder. This takes a little while, so try to remain calm.
2. You’ll eventually get a standard dialog box allowing you to save the format information in the file and folder of your choice. The default file shown will be **Plain.fmt**. The best location for format files is the **TeX-formats** folder.
3. Note that **plain.log** is created in the current folder. It’s not needed, so delete it.

The steps needed to create **LaTeX.fmt** are very similar:

1. Run **INITEX**, wait for the “**” prompt, type “**lplain\dump**” and hit Return. T_EX will begin reading **lplain.tex** from the **TeX-inputs** folder. This takes even longer than above.
2. This time the standard dialog box will show **lplain.fmt**, so simply change the name to **LaTeX.fmt**.
3. **lplain.log** is created in the current folder. Delete it if you want to.

If you decide to rename one of the existing formats, or add a completely new format, then remember to update your configuration file(s).

Different formats

OzT_EX comes supplied with the two most popular formats: Plain and L^AT_EX. There is also a Times-L^AT_EX format for people who prefer to use PostScript text fonts instead of Computer Modern.

The items below **INITEX** are read from the format list specified in the most recently selected configuration file. The current format is indicated by a tick and will be the one used when you typeset a file using either of the first two items in the T_EX menu. To change the current format simply select a different format item. OzT_EX automatically ticks the first format found when reading a configuration file, so place your preferred format first.

4.4 The View menu

View DVI...

Previews a selected DVI file. OzT_EX will open the view window and set its title to the name of the selected DVI file. You’ll then be presented with a dialog box that lets you choose various viewing options:

- The initial page to display. A scroll bar allows you to locate any page in the DVI file; it will be inactive if there is only one page.
- The paper orientation (landscape or portrait). Any change to this option will be remembered the next time you print or view a DVI file, unless you Cancel the dialog.
- The DVI magnification. You should only alter the default value if you plan to print the DVI file at a different magnification. See the description of “Print DVI...” in section 4.1 for more details.

If you Cancel the dialog then OzTeX will close the view window. If you select the View button then the current DVI/TeX page numbers will be appended to the DVI file name in the view window’s title and OzTeX will locate the requested page, interpret it and display it. You might have to wait a few seconds before seeing anything, especially if it is a complicated page.

OzTeX chooses an initial scale factor and page location based on the current size and shape of the view window. Using the paper dimensions in your configuration file (swapped automatically if you chose the landscape option) it will try to show as much of the paper as possible, making sure that the top, left and right edges are visible. If any part of the page is off the paper then OzTeX will beep and show a full view instead (see the “Full View” item below).

View ?.dvi

Previews the indicated DVI file. Exactly the same sequence of events described above will occur. This item is initially disabled; its name changes whenever you select a DVI file via “Print DVI...” or “View DVI...”, or create a DVI file by running TeX. You can preview the indicated file even after moving to another folder.

Show OzTeX/View

Brings either the OzTeX window or the view window (if open) to the front.

Page Info

Displays information about the current DVI page in the OzTeX window, bringing it to the front if necessary. This item is disabled if the view window is closed. The information displayed includes:

- The ten TeX page counters stored with the current page (trailing counters with zero values are not shown).
- The total number of pages, current resolution, DVI magnification and paper dimensions.
- A list of all the fonts used in the entire DVI file. For each font appearing on the current page OzTeX will show the total number of characters used. Note that the order of fonts may change as you move from page to page; OzTeX sorts the list so that all fonts actually used on the current page appear first. A PostScript font is indicated by its corresponding TFM path name and the requested point size. A non-PostScript font is indicated by a PK path name. Missing fonts are flagged by the message “DOES NOT EXIST”.
- The number of rules on the page.
- The location and argument of each \special command on the page. The location is expressed in terms of OzTeX’s paper coordinate system; see section 2.4.1. If the \special command includes a valid PICT/PNTG/EPSF file then the width and height of the bitmap is also displayed (unless it is empty).

Previous Page

Displays the previous DVI page. This item is disabled if the view window is closed or currently displaying the first page.

Next Page

Displays the next DVI page. This item is disabled if the view window is closed or currently displaying the last page.

Select Page...

Brings up a dialog box allowing you to select any DVI page for display. This item is disabled if the view window is closed or if the DVI file only has one page.

Full View

Changes the scale factor and location so that the entire DVI page and paper edges are displayed in the middle of the current view window. The scroll bars are disabled and the scale factor is set to its maximum value. This item is disabled if the view window is closed or already displaying a full view of the page.

Actual Size

Changes the scale factor so that the DVI page is displayed as near as possible to its actual size. The top left corner of the view window will still show the same page location, unless shifting is necessary to keep within the scrolling limits. This item is disabled if the view window is closed or already displaying the page at its actual size.

The scale factor is actually set to the nearest integer equal to the viewing resolution (defined in a configuration file) divided by the Mac screen resolution. A likely calculation is $300/72 = 4.17$, so the scale factor is set to 4. Because of this approximation, a 3 in wide rule in your \TeX input file won't appear exactly 3 in wide in the view window. If this is a serious problem then switch to a configuration file that sets the viewing resolution to some multiple of 72; you'll also need to get a corresponding set of PK files.

Zoom In

Halves the current scale factor. The top left corner of the view window will still show the same page location, unless shifting is necessary to keep within the scrolling limits. This item is disabled if the view window is closed or already displaying the page at minimum scale factor.

Zoom Out

Doubles the current scale factor. The top left corner of the view window will still show the same page location, unless shifting is necessary to keep within the scrolling limits. This item is disabled if the view window is closed or already displaying the page at maximum scale factor.

4.5 The Config menu

Show Config

Shows the current values of all configuration parameters in the $\text{Oz}\text{\TeX}$ window. The output is displayed with the correct syntax for a configuration file; this makes it easy to create a new configuration file by selecting "Save $\text{Oz}\text{\TeX}$..." from the File menu.

Default

The **Default** configuration file is read, resetting all parameters to their default values.

Other configuration files

The remaining items in the **Config** menu are determined by information given in the **Default** configuration file. The items should correspond to the names of text files in the **Configs** folder. Selecting one of these items causes the corresponding configuration file to be read and interpreted. A tick appears next to the most recently selected file.

4.6 The Help menu

The items appearing in this menu are read from the help list in a configuration file and should correspond to the names of text files in the **Help-files** folder. Selecting an item simply causes a copy of the matching file to appear in the **OzTeX** window.

Feel free to add more items to the Help menu. For example, you might like to create a file of **TeX/LaTeX** commands to refer to while editing an input file. If you want to add more help files, or modify the existing ones, then there are a few things to watch out for:

- Don't use tabs.
- Avoid long lines. **OzTeX** uses 9 pt Monaco to display the text; on a Mac Plus screen there is room for up to 80 characters per line.
- If you create a new file then remember to update your configuration file(s).

5 Including graphic files in TeX

Creating figures and illustrations with **TeX** isn't easy. Although it is theoretically possible to place small dots anywhere on a page and build up an arbitrarily complex picture, time and memory limitations make such a scheme impractical. (**LaTeX** provides a **picture** environment, but it is very inefficient and only suitable for small, simple diagrams.)

TeX does however provide a **\special** command which can be used to pass arbitrary information to the DVI interpreter. **OzTeX**'s printing and previewing code interprets a **\special** command using a simple syntax that allows inclusion of a given file.

Note that using **\special** can seriously reduce the portability of your **TeX** documents, since a given DVI-reading program can only understand **\special** commands that obey its own particular syntax. One useful technique to enhance portability is to hide the actual **\special** call inside a macro; then when you move your **TeX** file to another computer all you need to do is change the macro definition (assuming the DVI driver supports similar functionality in its handling of **\special**). For an example of this approach see Larry Siebenmann's **BoxedEPSF.doc** in **TeX-inputs**.

A **\special** command can appear almost anywhere in your input file. It behaves like an invisible box of zero height and width. **TeX** simply stores the given information in the DVI file at the current page position. When previewing a DVI file, **OzTeX** draws a small marker indicating the location of a **\special**. The "Page Info" item will display this location in paper coordinates as well as the text of the **\special** command. Similar information is displayed if you choose the "show statistics" option when printing a DVI file.

5.1 Including Macintosh graphic files

OzTeX allows the inclusion of three common types of Macintosh graphic files:

- PICT files can be generated by almost all drawing/painting programs on the Macintosh. A quick way to get a PICT file if you're using System 7 is to type Shift-Command-3. To include a PICT file use a command of the form:

```
\special{pict: filename}
```

- PNTG files are also very common — they're better known as MacPaint files. Shift-Command-3 on a Mac running System 6.x or earlier will create a screen dump in a file of type PNTG (note that if you have a colour monitor you'll first need to set it to black and white). To include a PNTG file use a command of the form:

```
\special{pntg: filename}
```

- EPSF files are becoming increasingly common. They contain encapsulated PostScript code in the data fork which OzTeX uses when printing and a corresponding PICT image in the resource fork which OzTeX uses when previewing. To include an EPSF file use a command of the form:

```
\special{epsf: filename}
```

The keywords and file names are case-insensitive; “=” can be used instead of “:” to terminate the keyword, and spaces before the file name are ignored. For example, `\special{EPSF=F00}` is the same as `\special{epsf: foo}`.

All three `\special` commands allow you to preview graphics in your DVI file and should become the preferred way of including Mac-generated pictures in OzTeX. If you have a choice, EPSF files are probably the best because:

1. The printed output is determined by the PostScript code and should be of much higher quality than a similar PICT/PNTG image (which are printed as bitmaps scaled to the desired size).
2. A coloured EPSF picture will print correctly (assuming your printer supports colour) even though OzTeX will convert its PICT resource into a black and white bitmap when previewing. (A coloured PICT file is converted to a black and white bitmap for printing AND previewing.)
3. TeX macros can be used to extract the bounding box information in an EPSF file and position the image automatically.
4. Encapsulated PostScript is supported on many other computers.

When creating a PICT/PNTG file it doesn't really matter where you position the picture; OzTeX will convert it to a black and white bitmap, find the bounding box of all black pixels in the bitmap, then place the bounding box so that the bottom left corner coincides with the location of the `\special`

OzTeX does NOT attempt to do this for an EPSF picture. When printing an EPSF picture it temporarily sets the PostScript origin to the location of the `\special` and then includes the encapsulated PostScript code verbatim (disabling any `showpage` command). When previewing, it converts the corresponding PICT resource into a bitmap and places the bottom left corner at the `\special` location.

OzTeX allows optional `keyword=value` pairs after the file name in a PICT/PNTG/EPSF `\special`. (The keyword can also be terminated by a colon or a space.) Here are the currently supported keywords: