The Electronic Publishing Toolbox

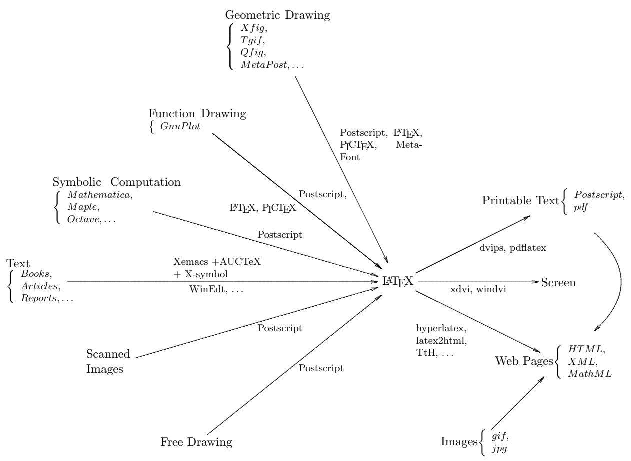
Pedro Quaresma*
CISUC
Departamento de Matemática, Universidade de Coimbra
3001-454 COIMBRA, PORTUGAL

September 15, 2001

Abstract

The Electronic Publishing of Math Texts, from books to web-pages is possible with the use of the *Electronic Publishing Toolbox*.

The TEX System and the Postscript language are the core of the Electronic Publishing Toolbox. Numerous auxiliary tools can be found ranging from Symbolic Computation Systems to Free-hand Drawing programs.



We will describe such a toolbox, its highlights, and also some of its drawbacks.

Keywords. Electronic Publishing, LATEX, Postscript, HTML

*This work was partially supported by the Portuguese Ministry of Science and Technology (MCT), under the programme PRAXIS XXI.

1 Introduction

What do we want of a Electronic Publishing Toolbox (EPT for short)?

- Quality, as much as possible;
- Extensibility, that is, the capability of dealing with many different types of requests;
- Platform independence, to allow an easy cooperation among authors;
- Math-capability, that is, it must be able to deal with math texts.

We think that none of the existing commercial programs fulfil all of this items: many of them are platform dependent; many of them have a low to average capability/quality when dealing with math texts; many of then, if not all, are unable to deal with all the requests of the users.

Having that in mind we think that we must aim to a toolbox solution, that is a set of cooperative platform independent tools, tools of excellence in its own domains, integrated in some form.

The LATEX and the Postscript language are the tools of excellence around which we can construct our EPT. Why?

- excellency, in regular texts, math texts, and graphics.
- platform independence.

We may add to this reasons the fact that the TEX system is freely available in the Internet and/or from the local support groups.

2 The LaTeX System

The TeX system [7] was created by Donald Knuth in the late 1970s as a special-purpose programming language that is the centrepiece of a typesetting system that produces publication quality mathematics (and surrounding text), available to and usable by individuals. The LaTeX subsystem [8, 10] is a special version of TeX that adds to this many commands to give the user a more powerful and friendly environment. The TeX/LaTeX systems are now a "de facto standard" in typesetting mathematics.

One of the TeX drawbacks was/is its inability to deal with graphics, but "the door was left open", that is, the command `special` was left in the language as a door to the outside world, as a door to the graphics world, and through that door the Postscript Language made its entrance [5].

3 The Postscript Language

The Postscript has been established for over a decade as an extremely flexible page-description language and it remains the tool of choice for professionals typesetters. Among the features that make it so attractive are:

- the quantity, quality, and flexibility of Type 1 fonts;
- the device-independence and portability of files;
- the quality of graphics and the quantity of drawing packages that generate it;
- the facilities to manipulating text;
- the mature colour-printing technology;
- the encapsulation conventions that make it easy to embed Postscript graphics;
- the availability of screen-based implementations (Display Postscript and Ghostscript/Ghostview).

4 The Electronic Publishing Toolbox

To each user its toolbox!

The EPT is a set of many tools, each user will build his own toolbox according to his needs. The EPT that we will describe now is biased by the author experience and needs, other authors will include/exclude some tools, the core will remain the same.

The author is, for many years now, a Unix/Linux user so this EPT reflects that, but all the tools (exceptions to that will be explicitly noted) that will be described are available in many platforms.

4.1 The Editor (Xemacs+AUCTeX+X-symbol)

Emacs is a extensible, customisable, self-documenting real-time display editor. XEmacs [13] is a version of Emacs with an extensive graphical user interface support when running under the X-window system.

Xemacs is easily extensible since its editing commands are written in Lisp, AUCTeX and X-symbol are two such extensions especially implemented to deal with TeX documents.

The environment provided by the combination AUCTeX [1] plus X-symbol [14] provides a very friendly environment to write, compile, visualise, and print TeX documents.

This system is present in the Unix/Linux platforms, but it is also present (at least the Emacs + AUCTeX combination) in other platforms, e.g., the Win32 platform.

There are other (platform specific) high quality editors with a embedded TeX capability. All the TeX distributions (commercial/non-commercial) contains, at least one, high-quality TeX-editor.

4.2 Free Drawing & Scanned Images

Many programs are available to deal with images, scanned images or images created by the user. The inclusion of such images in a LaTeX document is possible by way of embedding Postscript graphics. So it is a very simple process:

1. create the image (free drawing or scan);
2. save it in the Postscript format;
3. include it using a `special` command.

The `graphicx` package [5, 8] has many commands to deal with the insertion of Postscript graphics in a LaTeX document, inclusion, scaling, rotation, and others.

4.3 Symbolic Computation & Function Drawing

Many symbolic computation systems are available for the mathematical practitioner, their high-level languages provides a convenient way for solving, among others, linear and nonlinear problems numerically, to draw graphics, to manipulate mathematical formulas symbolically. Again the interface between this type of programs and the TeX system is the Postscript language.

The *Gnuplot* [3] program deserves a special reference. Gnuplot is a command-driven interactive function plotting program, some of its features are:

- Plots any number of functions, built up of C operators, C library functions, and some things C doesn't have like **, sgn(), etc. Also support for plotting data files, to compare actual data to theoretical curves.
- User-defined X and Y ranges (optional autoranging), smart axes scaling, smart tic marks.
- Labelling of X and Y axes.
- User-defined constants and functions.
- Support through a generalised graphics driver for (among others): EEPIC, TeXDraw, EmTeX, LaTeX, PostScript, X11, IBM CGA, EGA, VGA.
- Shell escapes and command line substitution.
- Load and save capability.
- Output redirection.
- All computations performed in the complex domain. Just the real part is plotted by default, but functions like imag() and abs() and arg() are available to override this.

Two of its features are of great interest: its cross-platform capability (Unix/Linux, Win32, Commodore Amiga, among others); its LaTeX interface, the graphics produced by Gnuplot can be embedded directly in a LaTeX document providing in this way a complete integration in the surrounding text.

4.4 Geometric Drawing

The *Xfig* (X11), *tgif* (X11), *Qfig* (win32), *LaTeX-CAD* (win32) are examples of menu-driven tools that allows the user to draw and manipulate objects interactively under a specific Window System.

They all have in common the possibility to export the drawings in Postscript and/or in a LaTeX specific graphics format (e.g. the PICTeX [12] format). As in the Gnuplot case the exporting in a LaTeX specific format allows a complete integration with the surrounding text.

5 The Final Product

We have written about the tools of our EPT, we will now write about the texts produced by the EPT.

5.1 Viewing

The output of a TeX compiler is a file in the format "dvi" (for **De**vice **I**ndependent format), each TeX distribution has a dvi-viewer (at least one), they are capable of visualise the final (pure) LaTeX document in the computer screen. Many of this dvi-viewer are also capable of visualising the LaTeX documents with Postscript specials included, but scaling, rotating, and other Postscript "dirty-tricks" are (usually) not supported, to that purpose we must relay in a postscript-viewer.

The postscript-viewer *GhostView* [4] is a program that we can find in many platforms, with it we can preview the Postscript files, selecting pages, magnifying, printing, and many others commands.

The *dvips* [2] is the program, available in every TeX system, capable of dvi-file to ps-file convertion, it deals with the Postscript specials, doing the inclusion of any Postscript file in the LaTeX dvi-file producing a Postscript file that contains all the information.

5.2 Printing

All TeX distributions have many printer-drivers capables of printing in many different printers, but for simplicity the solution adopted by most (if not all) the TEX distributions, is:

1. produce the dvi-file;

2. convert it to a ps-file (dvips);

3. print the ps-file (ghostview).

In the last few years Postscript has spawned an enterprising child, the PDF (portable Document Format) language used by the Adobe Acrobat. This is another possibility for producing the final text, the *pdftex*, and *pdflatex* are two special versions of TEX, and LATEX capables of producing pdf-files. After that we can use the *Acrobat Reader* for viewing and printing.

5.3 Web-Publishing

The web-publishing is another area that the users of a EPT can try with total confidence. There are two ways (at least) to deal with the problem for a EPT user:

- to use a converter LATEX to *Html*.

- to use a special LATEX package to produce *Hyper*Documents, and a converter to produce the corresponding *HTML* file.

For the first solution there are many candidates, *latex2html*, *TtH*, are two of them. They try to convert the LATEX document in a HTML document, because of the limitations of the HTML format the mathematical text must remain has an outside object, most of the time as an image converted automatically by the converter.

The *HyperLaTeX* [6] package has a different solution for the problem. The user of such package uses a special LATEX style file, this style file provides many new commands wich allows the user to write many hypertext commands directly in the LATEX document, after that a emacs-lisp converter, transform the LATEX file in a HTML file.

The two solutions have each some good points and some bad points. The converters are very powerful, but when they cannot deal with something they left the user with an error-message and nothing more.

With the HyperLaTeX solution the user has much more control over the final result, and in most error situations the user his capable of devising a solution to the problem.

In either cases we must realize that the HTML format is appropriate only for plain text, the inclusion of math text in Web-pages must be deal with:

- the conversion of the formulas to a graphical representation of the formulas;

- use XML, and MathML (in the near future);

- include the text in PDF format.

This last solution is the best solution if we want to include in a web-page a text in its final form. As we already saw the *pdftex* and *pdflatex* are capables of converting any LATEX document into a PDF file.

6 Conclusions

The presentation of the EPT is based in our belief that it is better to have many (excellent) tools glueded together by two excellent tools (the TEX system, and the Postscript language), than to have a (fair) tool trying to do all, even if packed with a nice suit.

References

[1] *The AUC TeX info-page.*

[2] *The Dvips man-page.*

[3] *The Gnuplot man-page.*

[4] *The Gostview man-page.*

[5] Michel Goossens, Sebastian Rahtz, and Frank Mittelbach. *The LATEXGraphics Companion.* Addison-Wesley, Reading, 1997.

[6] *The HyperLaTeX man-page.*

[7] Donald E. Knuth. *The TeXbook.* Addison-Wesley Publishing Company, Reading,Massachutts, 3 1986.

[8] Leslie Lamport. *LATEX: A Document Preparation System.* Addison-Wesley Publishing Company, Reading,Massachusetts, 2nd edition, 1994.

[9] *The latex2html man-page.*

[10] Pedro Quaresma. *Introdução ao LaTeX*. Escolar Editora, 1996.

[11] *TTH: the TEX to HTML translator web-page* http://hutchinson.belmont.ma.us/tth/

[12] Michael Wichura. *The PiCTeX Manual.* M. Pfeffer & Co., New York, 1987.

[13] *The Xemacs man-page, and info-page.*

[14] *The X-Symbol info-page.*