

The XEM \TeX Project

Marie-Louise Chaix

EDP Sciences, Paris

mlchaix@edpsciences.com

<http://www.fptex.org/xemtex/>

Fabrice Popineau

Supélec, Metz

fabrice.popineau@supélec.fr

<http://www.fptex.org/xemtex/>

Abstract

We are presenting the XEM \TeX project which is funded by the French government under the RNTL program (National Network on Software Technologies). The purpose of the project is to provide a high-quality, user-friendly, free integrated typesetting platform to typeset scientific or technical documents.

Résumé

Nous présentons ici le projet XEM \TeX qui a été retenu pour être financé par le gouvernement français dans le cadre du Réseau National des Technologies Logicielles. Ce projet a pour objectif de fournir une plateforme typographique intégrée, libre, facile d'accès et de haute qualité pour la mise en page de documents scientifiques et techniques.

Why XEM \TeX ?

\TeX has been available and used by many people for 25 years by now, so it must have some strength and it must outperform its competitors. But there are not so many \TeX vendors. It is not mainstream among desktop publishing software or text processing software for several reasons:

- \TeX is a programming language, which makes the learning curve much steeper,
- \TeX is useful for complex typesetting jobs, with lots of mathematics, or complex typography, but many people have jobs that do not require the complexity of \TeX ,
- \TeX source code is free since it has been given to the community by its author, D.E. Knuth. There are not so many commercial, supported versions of \TeX ; it is essentially used in the academic community, in its free form.

\TeX distributions have been floating around for free for a very long time. They used to be difficult to install and to maintain. Much progress has been made in the recent years.

One of the very first complete and well-designed distributions of \TeX was named em \TeX (from its author, Eberhard Mattes) and targeted MS-DOS. This distribution of \TeX had the advantage of fitting on only a few

floppy disks. It was solid, well integrated with the environment, and had a lot of success.

Then te \TeX brought Unix users a complete and sound set of programs and style files, much more extensive than the set provided with em \TeX . Since 1994, Thomas Esser has done a great job in smoothing over the installation of \TeX , especially if we consider that in the tradition of Unix programs, you get the sources and it is up to you to compile and install everything.¹ Thomas Esser's distribution is based on the Web2C sources of \TeX maintained by Karl Berry and now by Olaf Weber, and it is completed by several scripts and programs helping a lot with use, configuration and maintenance of \TeX .

Windows users have different requirements than Unix users. The level of technical knowledge is not generally the same. For example, Windows users do not expect to interact with the computer through the command line — don't even think of making them compile their programs! Windows and MacOS advertise their graphical user interface. Unlike MacOS (up to version 9), Windows has a builtin command interpreter, even quite a fancy one under the NT/2K/XP flavours of Windows, but although it exists, only a few users know how to use it. \TeX being a compiler which is invoked through the command line, Windows and MacOS users expect it to

1. Things have changed only recently with the advent of Linux distributions and the availability of precompiled packages.

be wrapped up in a GUI application. Dedicated text editors have been devised to this effect, other ones have been adapted to it.

MiKTeX and fpTeX are the main free versions of TeX for Windows. MiKTeX is an entirely new port of TeX to Windows, whereas fpTeX is based on the very same sources as teTeX and Web2C. These distributions are now much larger than emtex: they do not fit on floppies, but on CD-ROMs! In the meantime, the TeX community has developed lots of new tools and macro packages, which explains the inflation.

During all these years, another project was born to deliver a full, ready-to-run version of TeX on CD-ROM: the TeX Live project. This project aimed at being the most complete set of TeX programs, macro packages and style files, or related tools. But the novelty was that you could run the tools from the CD-ROM without installing anything. Moreover, almost all frequently used architectures were available: several Unices, Windows, MS-DOS and Amiga up to some point, Mac OS X nowadays.

But all these distributions target only the TeX programs and related tools. They do not target the full environment needed to actually typeset documents. To our knowledge, only the 4AllTeX project had this ambition in the past. The 4AllTeX CD-ROMs brought you a complete environment under Windows to type in, compile and print your documents. Everything was included, from the editor to the printer drivers, and including tools to handle images or to draw graphs, a spell checker, etc. The wonderful idea was to write a set of *wizards* that assisted you in your most common tasks. Everything was configurable, so that you could choose your text editor or your TeX engine and so on. The result was a nice environment, really easy to work with. Unfortunately, this project stopped circa 1999.

Having worked on the TeX Live project for many years, trying to bring Windows users with a comprehensive, up-to-date and sound TeX distribution, we wondered how to turn this distribution into something closer to a Windows application than has been the case till now. Windows users like Microsoft Word (or any other DTP program) because when it is installed, there is only one icon on the desktop to click, and all functionality is available from there. The application is compact and the interaction is clear. Obviously, TeX being much older and free software, it cannot compete with such a facility of use. However, it seemed that it should be possible to reduce the gap between both kinds of programs.

There are free text editors that are both well integrated with their environment and with cross-platform availability. The most well-known freely available text editor is probably GNU Emacs. But we would not qualify it as well integrated with its environment. Using GNU Emacs under Windows may seem a bit strange to

Windows users, mostly because it does not take advantage of native controls (native toolbars or other common controls). On the other hand, there is extensive support for TeX documents inside Emacs. Fortunately, there is an alternative to GNU Emacs: XEmacs is a forked project from GNU Emacs 18 series. It started as Lucid Emacs, and then became XEmacs. The difference between the emacsen is mostly in the internals and the external aspects, but they share many features and a lot of the “feel” part of the look-and-feel. The look part is much more in keeping with the environment under XEmacs, especially if we consider the Windows native version and the GTK version.

So the big idea came that XEmacs could act as an integrated environment for writing, compiling and printing TeX documents. Well, this is not so big an idea for experienced computer users, but the fundamental concept is that this environment could be distributed to less experienced computer users. Let’s reserve some of the details for the moment. The main point is that the user could access his typesetting environment just by clicking on an icon on the desktop, as is the case for most DTP programs, and that’s a big selling point in our opinion. TeX and all the machinery would remain hidden behind the scene.

Free software and funding

Not everyone is interested in TeX, and TeX is not the right answer for everyone. However, for certain types of jobs or types of documents, TeX has no competitors. Education is a target for TeX, for several reasons:

- obviously for mathematics and science teachers, because of the high quality result in typesetting maths,
- stability and long lasting documents: TeX has been available for 25 years and the input language does not change with each version like most commercial products,
- it is free and available on all platforms.

For a few years, there has been a growing demand from teachers — especially from high-schools — towards GUTenberg for hints on how to start with TeX. Part of this demand is due to the growing popularity of free operating systems like Linux. GUTenberg had a project to build a dedicated CD-ROM and distribute it through the national education system in every high-school in France. However, this was a free project, and like many free projects, it was to be developed in the free time of people who were already busy, so it came to nothing.

As we wanted to ground this project a bit more than the average free software project, the question arose about the viability of the project and how to fund it. At the same time, the French government brought us an

opportunity though the so-called National Network on Software Technologies. In short, projects eligible to be funded must involve at least one company and one academic institution. Projects can fall in one of several categories:

1. pre-competitive projects, which should lead to final products at the end of the project,
2. research projects to develop new technologies or new algorithms.

The XEM \TeX project did not fall into either of these categories in any obvious way. However, the RNTL allowed for a third kind of projects to be presented: free software projects. They even released an economic study of the various business models for free software distribution with the clear intention to support some free software projects.

So this was the opportunity we were looking for, and partners to develop and submit the project were quickly found:

- Fabrice Popineau teaches at Supelec (<http://www.supelec.fr>), an academic institution in the field of electrical engineering. He has been working on the \TeX Live project for several years,
- Marie-Louise Chaix is Project Leader at EDP Sciences (<http://www.edpsciences.org>), a French publisher specializing in scientific journals, books and electronic publishing. In her company, many people are potential professional users of the resulting product and will bring their desires and expertise into its design.

We submitted the XEM \TeX project with the following arguments.

First of all, there are very few commercial DTP and text processing programs that are very widely spread, the most common being MSWord. So any new product has little chance to gain an audience. But in the particular domain of scientific and technical documents, those widespread commercial programs have poor performance. If you want high quality typesetting, then you must resort to \TeX . It will provide the required quality in whatever notation system you need: maths, physics, languages from all around the world, music, etc. Being programmable, \TeX has been able to adapt to new technologies like PDF and HTML.

When it comes to text editors, you can't avoid speaking about the Emacs family. The emacsen stability and versatility is well-known. As they are programmable, they can be tailored to anybody's use. Even if the standard way to interact with an emacs is difficult to learn, and maybe not that intuitive today (key sequences), it is possible to reprogram this and turn it into something more suited to our needs.

So all the pieces exist, free of proprietary rights, to assemble a high quality typesetting platform that would outperform many commercial programs for our kind of documents — and that is the first goal of the project.

The second goal of the project is to widen the \TeX audience by providing people with an integrated platform, easy to use even for novice users. As a matter of fact, for many people, \TeX is more difficult to install and maintain than it is to use. Typing in some \LaTeX document can be explained quite easily and it does not require any system administrator skill. It is not the same when it comes to actually putting the \TeX software or any of the related tools needed on the machine. Any \TeX distribution is made up of thousands of files, and conflicts can arise easily. Many get frustrated when they fail to install it at first try, and it gets even worse if they actually spend time trying to fix problems, unless they manage to do it. By using XEmacs as an integrator, we can isolate our \TeX system from the rest of the software installed on the machine, and the user will more likely get a working program. So we claim that if \TeX was as easy to install as other programs are, \TeX would have more users.

The project will rely on software components already available from the community and any new developments will be made available to the community free of rights. In the end, the XEM \TeX project will be integrated to the \TeX Live project. The XEM \TeX CD-ROM resulting from the project will be sent to all high-school teachers (mathematics and science) by the GUTenberg association.

The road to XEM \TeX

Framework All the needed tools to build a complete environment are freely available. \TeX Live under Windows used to try to help people with installing the most frequently needed tools:

- a text editor to select among half a dozen (WinShell, WinEdt, etc.)
- the Ghostscript PostScript interpreter, either in its free or non-free version,
- image files tools like NetPBM or ImageMagick,
- the Ispell spell-checker with dictionaries,
- Perl because many scripts use it and it is not available by default under Windows.

But this list of supplementary programs was difficult to maintain, mainly because these products were not repackaged and their installation procedure kept changing. Also, they were not mandatory, and their installation was not delegated to the \TeX Live system: they installed as standalone products, at the risk of conflicting with other versions.

Anyway, the starting point of the project is a subset of the \TeX Live 7 CD-ROM, augmented with XEmacs

and the packages relevant to \TeX typesetting, and the supplementary tools cited above. The goal of the project is by no means to write lots of new programs, but rather to put glue between existing programs and package the result so that it is easy to use for anybody. Relying on several large, complex products like \TeX , XEmacs, ..., we are forced to play safe with updates to these products. That means we can't patch them heavily to suit our needs, except if we can make sure the patches will find their way in the main distribution. That is why it is better to use the glue strategy than the patch strategy.

For its \TeX part, XEM \TeX will be a proper subset of \TeX Live: only the most commonly used packages, as few binaries as possible, nothing related to bitmap PK fonts. The other parts have already been named: XEmacs for integrating everything, Ghostscript, ImageMagick and NetPBM.

The project has been split into several tasks which are described below. The goal is to get progressively closer to the final XEM \TeX product.

Installer Lots of work has already been done in the \TeX Live project regarding installation, especially about specification of packages and writing so-called TPM files. XEM \TeX should be very simple to install, hence should offer as few options as possible to the user, contrary to \TeX Live, which offers lots of options to the user.

\TeX Live installers for Windows and for Unix are very different: it is an application with a graphical user interface under Windows and a shell script running in text mode under Unix. It has been unclear for a long time if a portable installer was feasible and desirable.

Given that the XEM \TeX installer should not offer many options, we may think of something much simpler for both cases. However, it appears that those platforms are really different in several system aspects — file associations, icons, menus and so on — so that it seems difficult to have only one installer. More precisely, each platform offers services to install applications, and these services are different. So the answer is that a native installer should be built on each platform. Given our targets, that means a Debian package will be built under Debian Linux, and a MSI (Microsoft Installer) package will be built under Windows.

Editing text The starting point here is XEmacs and the AUC- \TeX and Preview- \LaTeX packages. As powerful as XEmacs may be, some aspects of the user interaction may seem a bit strange or uneasy to novice users — think about dialog through the minibuffer, or complex key sequences for example. This is partly due to the fact that emacsen have been used mainly by programmers that understand this way of thinking, and partly by the fact emacsen can run in console mode, and no effort has been made to build a simplified user interaction mode. Emacs-

sen are complex tools for complex jobs. But it would be a shame if the most powerful text editor could not be tailored to suit the needs of less experienced people. So this part of the project will tackle the problem of ergonomics and definition of menus, toolbars and keymaps suitable for our goal. Following the general philosophy, we won't rewrite big parts of existing stuff, but rather tailor and wrap up the existing tools.

As an exception to this rule and as a specific subpart, an equation editor has been considered. Given our targeted audience, it could be a fancy and useful tool to provide people with. A first sketch has been written in wxPython to be portable. If it appears that the result is stable enough under both platforms, it will be integrated into the XEM \TeX product.

Apart from typing in \TeX documents, the user will have to handle other kinds of files: images and graphics. The related tasks will also be identified, and the common ones will be offered through menus and toolbars. For example, MetaPost is a nice tool to draw graphics and support should be provided for it inside XEmacs.

Viewing documents \TeX is not a WYSIWYG tool, which is a bit disturbing for people who don't know it yet. Specific attention will be paid to coupling the compiler and the viewer. Up to now, DVI and PDF have been the output formats considered. The DVI format has the advantage to be fast to display on screen, but the drawback not to be self-contained: fonts, images are stored externally. For the PDF format, it is the opposite.

The status of viewers available on \TeX Live is not the same under Windows and under Unix. The XDvi viewer for Unix is able to display Type 1 fonts and to use source specials to map a location in the viewer to the same place in the source file. Windvi available for Windows does not yet have those features.² However, we can wonder if the DVI format should be advertised as the format of choice for new users, especially if it were possible to view with the PDF format and the same source specials mechanism feature. Heiko Oberdiek has implemented such a mechanism in his vpe package. What needs to be investigated is: can it be reasonably used? Is it fast, is it convenient? If yes, no doubt the PDF format should be advertised over the DVI format.

What we want to avoid at any price is the use of bitmap font files. The mechanism to build these fonts is too complex, too error prone. Nowadays, with scalable fonts, we can completely avoid using bitmap fonts, and thus avoid installing METAFONT and all the programs calling it, like mktexpk. In any case, the Windvi viewer will eventually be brought up to the same level of features as the XDvi viewer.

². The source special mechanism is available, but not documented.

Documentation Documentation is the weak point about TeX: there is too much of it, and it is difficult for a new user to get started. So a specific effort will be devoted to write a clear set of documentation to bring people in the game. We won't rewrite a TeX or L^ATeX primer, but namely:

1. an installation guide which will describe precisely the installation, the removal and the maintenance of the product on each platform;
2. a quick starting guide which will describe how to typeset a document as quickly as possible;
3. a user manual which will describe in details what is possible at each step of the creation of a document, and the technical aspects of XEmTeX.

Current status of the project

The project has officially started November 2002 for 18 months. We did not require a lot of manpower and it is a small project among the RNTL funded projects. To make progress, we are adopting an iterative process: rather than tackle the whole problem at once at the risk of ending nowhere, we prefer to finalise small parts and release often.

Hopefully, the first 0.1 release of XEmTeX will be available for the 2003 EuroTeX conference!³ Once the initial framework is set up, we will build much more easily on top of it.

Conclusion

We think that TeX deserves to get a much broader audience than is the case now. The price to pay to get high quality typesetting is not that expensive. It may even be cheaper than the price to pay for using some commercial products which are less reliable. The national funding we have got enables us to make a further step in simplifying the access to TeX for many people through the use of free software. We hope that the TeX community will stand together and require bigger funding to revive the TeX ideas into much more modern programs that could find their way in the mainstream of DTP programs.

³. Updated since; you can download the current version from <http://www.fptex.org/xemtex/>.