**TUG 2012 abstracts**

**Bill Cheswick**  
*An iTeX update*  
An update on the iTeX project for ebook publishing with TeX, described in TUGboat 32:2.

**Federico Garcia**  
*Documentation in TExnicolor*  
My package colordoc builds on Frank Mittelbach’s docstrip system of documentation, adding some utilities to use color in the code: matching delimiters ({} and []) are colored the same, just as matching \if–\fi pairs. Commands are made red, bold, and italics, when they are being \defined, just as variables when they are being declared (\newcount, \newif, etc.). These tools have saved me a lot of time and trouble when editing or trying to understand a code. In the presentation I also describe the interesting general lines of the workings of both doc and colordoc.

**Troy Henderson**  
*User-friendly web utilities for generating BiTEx output and MetaPost graphics*  
The full article was printed in TUGboat 33:1. The online previewers are available at:  
http://www.thlhiv.org/ltxpreview BiTEx  
http://www.thlhiv.org/mppreview MetaPost  
http://www.thlhiv.org/mpgraph Function Grapher

**Sherif Mansour & Hossam Fahmy**  
*Experience with Arabic and LuaTeX*  
This is an experience report of an attempt to include the AlQalam font for Arabic script within LuaTeX. We describe the problems we faced trying to figure out how to use a new right-to-left font within LuaTeX. We also describe how to call the many different shapes that are defined via parameters in the original font. We also present some ideas to modify the line breaking algorithm of TeX to allow the use of different shapes for the same character in order to justify the line. This is still work in progress.

**Frank Mittelbach**  
*E-TEx: Guidelines for future TEx extensions, revisited*  
Shortly after Don Knuth announced TeX 3.0 I gave a paper analyzing TeX’s abilities as a typesetting engine. The abstract back then said:  
Now it is time, after ten years’ experience, to step back and consider whether or not TeX 3.0 is an adequate answer to the typesetting requirements of the nineties.  
Output produced by TeX has higher standards than output generated automatically by most other typesetting systems. Therefore, in this paper we will focus on the quality standards set by typographers for hand-typeset documents and ask to what extent they are achieved by TeX. Limitations of TeX’s algorithms are analyzed; and missing features as well as new concepts are outlined.  
Now — two decades later — it is time to take another look and see what has been achieved since then, and perhaps more importantly, what can be achieved now with computer power having multiplied by a huge factor and last not least by the arrival of a number of successors to TeX which have lifted some of the limitations identified back then. [Slides available at www.latex-project.org/papers.]

**Steve Peter**  
*Metafont as a design tool*  
Well-written Metafont sources provide a font designer with a nearly unparalleled tool to explore variations on a typographic theme. Paired with TeX in an advanced environment, the designer can explore serif structure, bracketing, weight variations and more in the context in which the font will be used: real textual matter. I’m going to ignore the production problems inherent to Metafont (not to mention the various possible solutions) to concentrate on the design aspects of this amazing tool.

**Norbert Preining**  
*Typesetting with Kanji — Japanese typography*  
Japanese typography is very particular and demanding in several respects: four writing systems mixed together (Kanji, Hiragana, Katakana, Roman letters); vertical and horizontal typesetting; traditional grid layout versus a mixture of writing systems. This all led to a spin-off TeX implementation called “Publishing TEx” (pTeX) that can deal with these specifics.  
Until 2011 there was an independent distribution of TeX for Japanese users, first based on ptetex, later on TeX Live (ptetex, ptextlive). TeX Live 2011 and 2012 introduced all of the necessary tools and features and we hope that with TeX Live 2012 the need for a special setup for Japanese users is past.  
In this talk we give an overview of the specialties of Japanese typography, presenting the difficulties met in modern texts. Continuing, we present the solutions provided by TeX Live to some of these problems, and discuss further development.

**Norbert Preining**  
*TeX Live 2012: Recent developments*  
TeX Live will be released in early summer 2012 and brings a couple changes that have been in the works for a long time: a “multi-updmap” that reads several updmap.cfg files, and multi-repository support for the TeX Live Manager tlmgr.  
The updmap program generates the necessary
configuration files for dvips, dvipdfm(x), pdftex, and pdxvi to display PostScript Type1 fonts. It reads a configuration file that lists several map files, and combines all the font definitions from these map files. Until now local font maps had to be integrated into this updmap.cfg file, and so could easily be overwritten or otherwise be lost.

The new implementation has a long history. The original Perl version was written by Fabrice Popineau for Windows, later extended by Reinhard Kotucha and Karl Berry and used, starting last year, on all platforms supported by \TeX{} Live. The code has now been extended to deal with multiple configuration files in a transparent way.

This allows a clear separation of updmap.cfg file parts. One updmap.cfg file now can (but does not have to) provide information about only the \texttt{texmf} tree it resides in. In other words, fonts installed into, for example, the \texttt{TEXMFLOCAL} tree can be activated by an entry in the updmap.cfg file \textit{in this tree}.

We will discuss this new functionality and provide usage examples and advise on transition from the old system.

The other big change in \TeX{} Live this year is the extension of the \TeX{} Live Manager with the capacity of reading multiple repositories. In recent years, a few alternative \TeX{} Live repositories have come into existence with a wide range of usage patterns: distribution of local packages (Japanese \TeX{} related packages in \texttt{tlptexlive}, Korean \TeX{} User Group repository), \TeX{} Live infrastructure testing (in \texttt{tlcritical}), provision of development and non-free packages (in \texttt{tlcontrib}), etc.

Until now a user had to go through all desired repositories one by one passing the necessary parameters for each in turn. The new \texttt{timgr} supports use of several sources at the same time. The selection of packages appearing in multiple repositories is done by “pinning” packages to a repository.

We will present this new functionality, give usage examples, and advise on transition from the old system.

We will close with an overview of other changes in \TeX{} Live 2012.

**Will Robertson & Frank Mittelbach**

\textit{\LaTeX{}3: From local to global — a brief history and recent developments}

The original source code for \LaTeX{}3 dates to the early 1990s. Key aspects of its development occurred during that decade, but it was not until the late 2000s that the project began delivering code that was widely used by mainstream \LaTeX{} users. What happened in this time? This talk will discuss how \LaTeX{}3 development evolved over the decades and how it reached a state of being used to produce real users’ documents whether or not they are actually aware of it. \LaTeX{}3 can be thought to consist of separate ‘layers’, and the programming layer known as expl3 is starting to be used to solve problems in and write packages for \LaTeX{} 2ε. Our plans are not restricted to such ‘under-the-hood’ measures, however, and we have discussed layers of \LaTeX{}3 that will have more visibility at the user interface.

Our talk will discuss these separate layers and where our plans lead in the future, and will conclude with a demonstration of what’s new in the current code.

[Slides available at \url{www.latex-project.org/papers}.]

**Will Robertson**

\textit{Lineage and progeny of fontspec and unicode-math}

My first \LaTeX{} package, fontspec, was written in 2004 before I knew how to program in \LaTeX{} and in truth before I knew how to program at all. This trial-by-fire introduced me to the lovely world of \TeX{} programming and after some time I ended up writing a smattering of other works. (All the while actually starting to learn what this whole ‘programming’ thing was all about, including how to please and displease people who were just trying to get work done, thank you very much.) Some time later I foolishly tried ‘planning’ an ambitious new package, unicode-math, that took significantly longer to release. In the course of writing that package I learned really just how little I actually knew, and as a side-effect somehow ended up helping to write code for the \LaTeX{}3 project. In this talk I will talk about the motivation for writing these two packages, discuss recent developments with them, and finally touch on how \LaTeX{}3 influenced their development.

**Herbert Schulz**

\textit{Workshop: Introduction to TeXShop}

A workshop introducing some of the more obscure and less used features of TeXShop for users who wish to become more proficient in its use to produce \LaTeX{} documents.

**Christina Thiele**

\textit{Almost 30 years of using \TeX{}}

It’s not just \TeX{} that’s gotten older and more seasoned . . . Reflections on changes in \TeX{} and friends as used in a small typesetting company: software and hardware, of course, but also procedures and skills, resources that went from zero to virtually infinite, all of it interwoven with life and personal change. It’s not earth-shaking news, but we’ve come far enough that looking back yields some interesting comparisons.