

---

## BachTeX 2014 proceedings

The BachTeX 2014 proceedings was published by GUST, the Polish language TeX user group ([gust.org.pl](http://gust.org.pl)). The web page for the conference program is [gust.org.pl/bachotex/2014/program](http://gust.org.pl/bachotex/2014/program).

JEAN-MICHEL HUFFLEN, What can typography gain from ePub?; pp. 5–12

We show that ePub — a well-known format for electronic books — has integrated many features related to nice typography in comparison with other formats such as HTML5. However, due to their respective designs, ePub does not reach the same quality as (L)TeX, even if some effects are easier to implement. We explain why.

JEAN-MICHEL HUFFLEN, Managing name conflicts and aliasing with MIBIBTeX; pp. 13–16

When several bibliography database (`.bib`) files are used to build a L<sup>A</sup>TeX document's references, BIBTeX signals an error if a bibliographical key is used more than once. A possible solution consists of renaming bibliographical entries, but MIBIBTeX now provides a cleaner way, by means of namespaces associated with `.bib` files. Symmetrically, we can now express that a unique bibliographical item is referred by several keys. These new features are put into action by means of both MIBIBTeX and an additional L<sup>A</sup>TeX 2<sub>ε</sub> package.

JEAN-MICHEL HUFFLEN, Musical symbols in the digital age; pp. 17–24

First, we briefly review the musical symbols available in Unicode and MusiXTeX. Then we show why the definition of such symbols is difficult if we aim to express the whole information included in a musical score.

TACO HOEKWATER, MetaPost development update; pp. 25–26

MetaPost 1.9 enabled the `decimal` arbitrary precision system. Using that as an example, adding the `binary` arbitrary precision system was a simple operation. What this means is that MetaPost 2.0 is finally finished. A momentous occasion for me, but it is also the proper moment to call it quits. I have no plans for further developments to MetaPost, nor does it seem even remotely likely that I would be able to find the time to implement any such even if I did come up with something. This will be my final MetaPost presentation as the active maintainer. The development of MetaPost will continue with Luigi Scarso as the primary maintainer.

BachTeX 2014 proceedings

TACO HOEKWATER, Lua & TeX tokens; p. 27

LuaTeX has had a token Lua library since the early beginnings, but it was more a proof of concept, and has never worked really well at that. This talk presents a new, better interface between Lua code and the TeX language parsing.

PRZEMYSŁAW SCHERWENTKE, Trup w każej szafie (o książce: L<sup>A</sup>TeX dla matematyków) [A skeleton in every closet (a book review: L<sup>A</sup>TeX for Mathematicians)]; pp. 28–30

The book presents the basics of typesetting with L<sup>A</sup>TeX for lay people proudly called mathematicians. Its positive side is that basic packages useful for the daily tasks are presented, particularly those for typesetting tables, pictures and mathematical formulas. Unfortunately over and over again this book makes the impression of being *written* by lay people. Truths are mixed with semi-truths and false statements, and a good part of the presented examples defies the rules of proper typesetting. The article (review) lists the most notable errors along with correction proposals, and compares each to a version from a different manual. There are also a few suggestions as to what should have been given but was omitted from nearly 300 pages of the book.

PIOTR BOLEK, MARIA BOLEK, and MIKOŁAJ TOPICHA-DOLNY, Technika i estetyka książki elektronicznej [The technology and aesthetics of electronic books]; pp. 31–38

What are e-books? Classification of e-books; Formats of e-books; Text formatting in e-books; Devices and software for e-book reading vs. aesthetic and technical aspects; Paging and hyphenation; Typography and graphics; Interaction, animation and multimedia; Practical examples.

PIOTR BOLEK, Używanie fontów systemowych w TeXu w różnych systemach operacyjnych [Using system fonts with TeX in various operating systems]; pp. 39–42

A presentation for beginners and medium advanced users. The aim is to give simple prescriptions on how to use the fonts given with the operating system or other OTF and TTF fonts employing the facilities available with the modern implementations of TeX (X<sub>Y</sub>TeX, LuaTeX). Using alternative glyphs and activating OTF features with ConTeXt and L<sup>A</sup>TeX.

KES VAN DER LAAN, PSlib.eps Catalogue, preliminary and abridged version; pp. 43–95

A selection of PostScript definitions collected in my PSlib.eps library and documented as an e-book

catalogue is presented. Now and then variant pictures have been included from `pic.dat` which comes with `Blue.tex`. Old Metafont code has been included which may be useful for MetaPost programmers. Variants of pictures enriched by postprocessing in Photoshop show other possibilities. Escher's doughnut is a teaser which has to be done in MetaPost. Along with `PSlib.eps` is the file `PDFsfromPSlib`, which contains the pictures in `.pdf` format. The complete `PSlib.eps`, `PDFsfromPSlib` as well as the catalogue as an e-book, will be released on occasion of NTG's 25th lustrum which will be celebrated in the fall of 2014, on [www.ntg.nl](http://www.ntg.nl). A prerelease will be offered to GUST's file server. The (static) library for  $\TeX$  standalone pictures, `pic.dat`, packaged with `Blue.tex`, will be redistributed as well.

HANS HAGEN, Lua in MetaPost; pp. 96–104

For some years I have been wondering how it would be to escape to Lua inside MetaPost, or in practice, in MPLib in  $\text{Lua}\TeX$ . The idea is simple: embed Lua code in a MetaPost file that gets run as soon as it's seen. In case you wonder why using Lua code makes sense, imagine generating graphics using external data. The capabilities in Lua to deal with that are more flexible and advanced than in MetaPost. Of course we could generate a MetaPost definition of a graphic from data but often it makes more sense to do the reverse. I finally found time and reason to look into this and in this article I will describe how it's done.

HANS HAGEN,  $\text{Lua}\TeX$  0.79; pp. 105–108

Around version 0.50 the general picture of  $\text{Lua}\TeX$  became more or less clear. Between versions 0.50 and 0.75 the program reached a level that made it possible to use it for production. Currently we're moving toward version 0.80. This version has some new features and existing features have been improved. The backend code is somewhat better separated due to a partial re-implementation of expansion. We're stepwise making the code base leaner, meaner and cleaner (again as a by-product of a critical edition project). What started as a transition from WEB to readable CWEB (an effort not to be underestimated) hopefully will become a coherent set of files with proper documentation. As there is still a long list of items to do; it will take us a few years to get there, but we're optimistic about the end goals.

In this talk I will discuss the work that has been done in the last year and present some of our plans for future versions.

PAWEŁ ŁUPKOWSKI, Online  $\text{I}\text{A}\text{T}\text{E}\text{X}$  editors and other resources; pp. 109–112

(Reprinted in this issue of *TUGboat*.)

LUGI SCARSO, Experiments with OpenMP and  $\text{Lua}\TeX$ ; pp. 113–138

This paper describes some experimental parallel functions implemented using the OpenMP API. A parallel version of `sort` is shown and discussed, and also some results about performance and efficiency.

KRZYSZTOF PSZCZOŁA, Książka papierowa na rynku zdominowanym przez publikacje elektroniczne: mniej powinno znaczyć lepiej [Paper books on the market dominated by electronic publications: fewer should mean better]; pp. 139–142

I am proposing a different viewpoint on the supposed evolution of the trends in connection with electronic and paper publications. The point I will make is that proliferation of electronic publications will enforce a change the way paper publications will evolve: there will be fewer of them, better edited and visually refined and, perhaps, they will bear less similarity to the books as we know them now. I envisage that, contrary to the prevailing belief that proliferation of electronic publications is a threat for people preparing paper publications (as less books will be printed), this would mean an opportunity for them (because paper publications will be better prepared and so require more work).

I will present possible business models which editors could employ when delivering both paper and electronic versions of the same content. As a brief digression, I will present a short elaboration on the possible physical forms “new books” might have (e.g., folder, leporello (concertina), or poster).

—\* — \* — \*

The following presentations do not appear as articles in the proceedings, but have slides linked from the on-line program. All the slides are located on this page: [www.gust.org.pl/bachotex/2014-pl/presentations/](http://www.gust.org.pl/bachotex/2014-pl/presentations/); however, there are no links from this page, so the name of the PDF file as shown below must be entered as part of the URL.

PATRICK GUNDLACH, Using Lua $\TeX$  the hard way: How to use the internal node structure of Lua $\TeX$  to create a PDF document without using `\backslashes`

With Lua $\TeX$  it is possible to access the internal data structures (so-called “node lists”) that  $\TeX$  creates after parsing the user’s input. You can analyze and modify the data before it gets written to the PDF file. It is even possible to programmatically create your own node lists and render these in the output file. One can also create a node list and instruct  $\TeX$  to break the list into lines, hyphenate it or insert ligatures.

This presentation gives an introduction to how node lists work. The `lua-visual-debug` package for Lua $\TeX$  serves as an example for using the necessary callbacks to analyze the node lists and to manipulate them. A simple node list creator shows how to construct an hbox and use  $\TeX$ ’s line breaking algorithm to get nicely formatted text.

`gundlach-1-b2014.pdf`

PATRICK GUNDLACH, speedata Publisher: Create complex documents from databases

$\TeX$  and  $\LaTeX$  are well suited for many different kinds of documents, not only when formulas are needed. Once you need complex tables and lists, or an index or table of contents or when you need automatically correct cross references and bibliographies, there are few programs better suited for typesetting tasks. But there are remain many cases where  $\LaTeX$  has its difficulties. For example: complex tables that are broken across multiple pages with changing headers and footers and running sums; good looking paragraphs with absolutely no overfull boxes; typesetting on a grid; free (exact) positioning of objects on a page or on a page grid; using arbitrary fonts; automatically adjusting paragraph shape based on image shapes (text flows around images); using text containers with overflow; safe usage of escaping characters/catcodes; automatic selection of master pages; and more.

`gundlach-2-b2014.pdf`

HANS HAGEN, What makes using  $\TeX$  and MetaPost interesting

While working with  $\TeX$  and MetaPost I often run into interesting situations. Sometimes they result in special styles (that probably go unnoticed), they result in additional features (that probably never get used because we forget about them), and they could also result in tracing features (which probably seldom get used). I use this opportunity to discuss a few of them that came up last year: realtime metafonts

(or: a way out of lack of symbols); juggling nodes (or: pseudo-extensions to Lua/ $\TeX$ ); surprising side effects of hashing (or: how LuaJIT $\TeX$  can be slower than Lua $\TeX$ ); generating graphics (or: visualizing data).

`bachotex-2014-metapost.pdf`

BOGUSŁAW JACKOWSKI, PIOTR STRZELCZYK, and PIOTR PIANOWSKI, On the progress of the  $\TeX$  Gyre Math project: TG Schola Math

Three fonts — TG Pagella Math, TG Termes Math, and TG Bonum Math — have been released so far within the frame of the  $\TeX$  Gyre Math Fonts project. We’ll present the next font, i.e.,  $\TeX$  Gyre Schola Math which completes the  $\TeX$  Gyre Math Font project. Of course, the maintenance of the  $\TeX$  Gyre collection will continue.

`tgm-final03web.pdf`

ANDRZEJ TOMASZEWSKI, Cuneiform script — a phenomenon of civilization

I will talk about the evolution of this form of writing from picture to alphabetic forms. About writers, written document types and writing materials used by the people of Sumer, Babylon and Assyria.

`pismo-klinowe.pdf`

ANDRZEJ TOMASZEWSKI, Absolutely non-computer and completely not programmable new book forms

I will talk about the quest for new and unconventional forms of books, created chiefly in designer circles connected to artistic books and the new world trend called bookart and amongst creators of the so-called liberature. It will be an apology of Krzysztof Pszczoła’s prophecies preaching the development of printed book forms.

`formy-ksiazki.pdf`

ULRIK VIETH, An improvised talk about the state of OpenType math fonts

In this talk, we review the state of OpenType math fonts which have been under development in the last few years. We discuss how to evaluate or test the quality of the design and implementation of these fonts. While a lot of progress has been made providing first releases of several new fonts, we suggest areas where additional work may be needed for improving and fine-tuning these fonts to reach production quality.

`conf-talk-ot-math-state.pdf`

[Received from Jerzy Ludwichowski and Tomasz Przechlewski.]