

ABSTRACTS

1 Monday, August 13, 2001

A T_EX Odyssey



1.1 OFFICIAL OPENING

1.2 T_EX 2001: Where will the odyssey bring us?

HANS HAGEN

Pragma ADE, GH Hasselt, The Netherlands
pragma@wxs.nl

Abstract: If there is one thing that the 2001 Space Odyssey movie has demonstrated, it probably is that even on a relatively short timespan predictions are hard to make. In spite of much research, some predictions have turned out the opposite while other developments were not foreseen. When Knuth wrote T_EX, one of his assumptions was that in 100 years from their incarnation, people should still be able to reproduce his books from the sources. Can we look that far ahead when it comes to typesetting? What will happen in the

(near) future with publishing? Is there a pattern in today's developments? Will all typographic problems be solved? In this talk I will try to reflect on these issues from the perspective of today's T_EX based publishing.

1.3 The Future of Publishing, Part 2

MIMI JETT
IBM Research
jett@us.ibm.com
<http://www.software.ibm.com/techexplorer>

Abstract: At the beginning of the past decade, (TUG'90 in T_EXas), we explored the future of publishing with an eye on the role T_EX and the T_EX community might play in the evolution of digital publishing. The idea of open software was fostering open communication between all players along the supply chain, from author through editorial and production. Did this dream materialize? Beyond the imagination, but the real benefits are yet to come. Now we see the collaboration between disparate groups and individuals from academia, business, and government resulting in powerful open standards that will shape not only digital publishing, but interaction and communication worldwide.

1.4 Integrating T_EX into a Document Imaging System

WILLIAM M. RICHTER
Texas Life Insurance Company, Waco, TX
hcswmr@texlife.com

Abstract: Modern computer hardware and software has made possible the construction of “document-imaging” systems. These systems maintain large repositories of documents in electronic form. In production environments of many large companies and in particular the life insurance industry, a significant percentage of printed documents are produced electronically in an automated fashion usually by merging variable data into an existing document with some fixed structure. Storing scanned images of these electronically produced print documents wastes time, computing resources, and disk storage space. It is useful to address the problem of document storage along with the related problem of electronically formatting and producing printed documents. Then the choice of document formatter can be made such that the formatting engine used to produce printed documents may be reused to display those same documents in a document imaging environment.

This paper examines how T_EX has been used as an important component in building a document production and imaging system at Texas Life Insurance Company. T_EX's macro facilities, conditional typesetting, text-based source files, a robust page formatting mechanism, and pre-compiled format files allow it to play a central role in the system. T_EXmerge, a C-language API, was developed to allow variable data to be merged, under program control, with static T_EX source documents containing special merge tags to produce a final output document. This API is used to prepare policy contracts, produce automated client correspondence, as well as in interactive document preparation and in application-specific document production. Documents produced via the T_EXmerge API are filed in the imaging system using a minimalist approach. T_EX form files are stored once and separately from all document instances. Variable data along with a pointer to its associated T_EX form file is all that comprise a stored document instance. When the document is displayed a “just-in-time” compile technique is used to reconstruct the document's .dvi file which is converted to PostScript for display purposes.

T_EX has additional attributes that make it an excellent choice as document formatting engine. The ability to convert raster bitmaps to T_EX fonts allow complex letter-head/footer macros to be developed

and easily used in a fashion that lends itself to effective revision management. Incorporation of a Code 2-of-5 scalable barcode font has enabled printed forms and documents that are returned to the company from external individuals to be recognized by the document imaging scanner and automatically filed in the imaging system.

1.5 Running a T_EX typesetting operation over the internet

C. V. RAJAGOPAL, KAVEH BAZARGAN, AND C. V. RADHAKRISHAN
Focal Image Inc., London, UK
kaveh@focalimage.com

Abstract: Focal Image runs a typesetting business, primarily for academic journals. We have two offices, one in the UK, and one in India. In addition, there are numerous freelancers who work for Focal Image, mainly copy editors. Although the majority of the production is undertaken in India, we have decided that the UK office is essential for the business, and should continue to have full production capability, although on a small scale. In order to allow this, we have developed a web-based workflow system that allows production work to be done in any part of the world. Moreover, the system allows for faster turn around of work, by cutting out administration, and making the work available immediately for the next stage, after one stage is completed. We will describe and give a live demonstration of the system.

1.6 Using T_EX for high end typesetting

HANS HAGEN
Pragma ADE, GH Hasselt, The Netherlands
pragma@wxs.nl
<http://www.pragma-ade.com>

Abstract: The fact that T_EX is already quite old, does not change the fact that it can still pretty well compete with high end desk top publishing programs. The fact that XML has simplified coding, if only by posing limitations, gives T_EX an even stronger position in todays typesetting arena. In this presentation I will discuss the challenges that come to us, present a couple of tools, and demonstrate what we may expect from T_EX today. I will also demonstrate that a modern T_EX like pdfT_EX can give today's T_EX users a pretty strong position.

1.7 T_EX — a mass-market product? Or just an image in need of a makeover?

PETER FLYNN
peter@silmaril.ie

Abstract: If T_EX systems are so good at typography, why does the documentation look so Despite widespread acceptance in the scientific field and with some publishers, and recent advances in the Humanities, T_EX systems have largely been publicised by word of mouth, and no-one can tell how many users

there are. The commercial versions advertise in the relevant places but generic publicity for TeX is not common.

Users can be TeX's best advocates, but formal training is rare. Users learn mostly from colleagues — themselves often ill-taught — and acquire bad habits which are hard to overcome. The results are often responsible for the poor image TeX has had among most printers and publishers. Although TUG runs courses, it is hard to cover a geographically dispersed user population.

Support for TeX via the Internet is excellent, usually far in advance of any commercial system, but there is always a need for more introductory documentation aimed at the non-scientific user. Some installation help is also still needed, especially for the first-timer: the assumption that everyone is already a skilled computer user no longer holds.

This paper argues that the biggest need is for distributable publicity targeted at identifiable markets backed up by readable and presentable documentation. More of the power of TeX systems should be made use of in creating these documents.

1.8 L^ATeX for Windows – A User's Perspective

DAVID M. TULETT

University of Newfoundland, St. John's, Canada
dtulett@mun.ca

Abstract: L^ATeX is a technically brilliant package for typesetting, but in the Windows world Microsoft Word continues to be widely used. From the perspective of an end-user of these products, the relative strengths and weaknesses of L^ATeX and Word are examined.

1.9 Instant Preview and the TeX daemon

JONATHAN FINE (presented by ART OGAWA)

jfine@active-tex.demon.co.uk

Abstract: Instant Preview is a new package, for use with Emacs and xdvi, that allows the user to preview instantly the file being edited. At normal typing speed, and on a 200MHz machine, it refreshes the preview screen with every keystroke.

Instant Preview uses a new program, dvichop, that allows TeX to process small files over 20 times quicker than usual. It avoids the overhead of starting TeX. This combination of TeX and dvichop is the TeX daemon.

One instance of the TeX daemon can serve many programs. It can make TeX available as a callable function. It can be used as the formatting engine of a WYSIWYG editor.

This talk will demonstrate Instant Preview, describe its implementation, discuss its use with L^ATeX, sketch the architecture of a WYSIWYG TeX, and call for volunteers to take the project forward.

Instant Preview at present is known to run only under GNU/Linux, and is released under the GPL. It is available at: <http://www.activetex.org>

1.10 REVTeX version 4.0, an authoring package by the American Physical Society

ART OGAWA
TeX Consultants
Three Rivers CA 93271, USA
ogawa@teleport.com

Abstract: The American Physical Society has just released a new version of their REVTeX authoring package, REVTeX version 4.0. The revtex4 document class for L^AT_EX 2_ε is completely new code, not a rewrite of REVTeX version 3.1. Those preparing electronic submissions to APS journals, like Physical Review, to American Institute of Physics journals, or to Optical Society of America journals will use revtex4, available via <http://publish.aps.org/revtex4> and on CTAN.

The revtex4 document class allows the user to prepare a LaTeX document that will be suitable for electronic submission to any of the subscribing journals, without further alteration. The formatting details connected with a particular target journal are entirely taken care of by a single document class option. For example, a submission to Physical Review Letters would contain the simple statement:

```
\documentclass[prl]{revtex4}
```

If targetting the Reviews of Modern Physics, the “prl” is changed to “rmp”. Within the document itself, the syntax is unaltered.

Version 4 contains most of the features of the considerably outdated REVTeX 3.1, with considerably more powerful formatting. It is capable of all of the features of LaTeX’s twocolumn style, but in addition allows switching to or from a one-column page layout anywhere on the page. A number of bugs in LaTeX have been fixed, among them the infamous “eqnarray” spacing problem. Title page information, in the past restricted to, e.g., \author, \title and \thanks, now accommodate all of the required features of academic journals.

The document class has been expressly designed for compatibility with the longtable package, and fixes a number of that package’s limitations, such as its inability to work in a multicolumn page layout. REVTeX is also compatible with such popular L^AT_EX extensions as hyperref, url, and array. Patrick Daly’s natbib package is always loaded by revtex4, and his custom-bib package has been used to create BibTeX styles for APS journals.

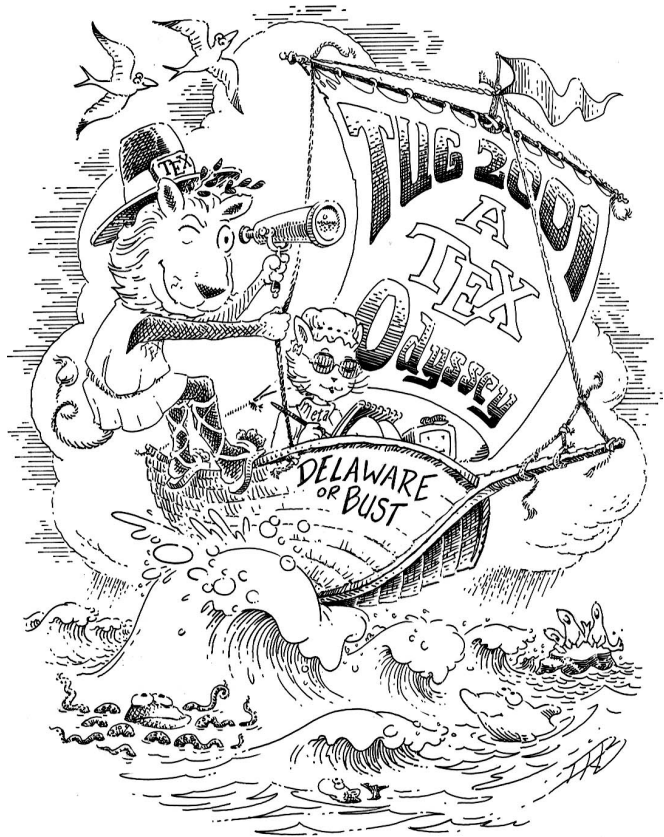
The revtex4 class incorporates two new L^AT_EX packages, ltxutil and ltxgrid, which are available separately for public use under the L^AT_EX Project Public License (LPPL). The latter package, based on code developed by William E. Baxter, provides a completely rewritten output routine for L^AT_EX, one that has far fewer limitations, and which fixes many of L^AT_EX’s bugs.

The revtex4 class is adaptable far beyond APS and OSA journals; users in our beta testing program have employed it for submissions to other journals, for the production of monographs, conference proceedings, and more. As well, the revtex4 document class provides an extensible architecture for other societies and journals to use: all of its APS-specific features are collected into a “sub-package”, called aps.rtx. Those wishing to customize revtex4 to their own journals are encouraged to write their own .rtx file.

The revtex4 document class is the responsibility of the American Physical Society’s Mark Doyle; the first draft was written by David P. Carlisle.

2 Tuesday August 14, 2001

PDF and TeX



2.1 Margin kerning and font expansion with pdfTeX

HÀN THẾ THÀNH
Masaryk University, Brno, Czech Republic
thanh@informatics.muni.cz

Abstract: This paper describes the two typographic extensions that have been implemented into pdfTeX: margin kerning and composing with font expansion.

Margin kerning is the adjustments of the characters at the margins of a typeset text to make the margins appear smooth. Only mechanical justification of the margins often causes them look rather ragged, as certain characters can make a line appear shorter than others. Shifting such characters by an appropriate amount into the margins would greatly improve the appearance of a typeset text.

Composing with font expansion is the method to use a wider or narrower variant of a font to make interword spacing more even. A concept to stretch or shrink a font was introduced and integrated to the box/glue/penalty model of tex line breaking. The potentiality to adjust a line width by font expansion can be taken into consideration while a paragraph is being broken into lines, in order to choose better breakpoints.

These extensions turned out to noticeably improve the appearance of a typeset text. A number of ‘real-world’ documents have been typeset using these typographic extensions.

2.2 PDF presentations using the *Marslide* package

ROSS MOORE

Macquarie University, Sydney, NSW, Australia
ross@maths.mq.edu.au

Abstract: The *Marslide* package is useful for creating high quality PDF presentation slides, especially when mathematics is required. It works equally well with both pdf- \LaTeX and \LaTeX +dvips+distill methods for generating documents in PDF format.

The package does not of itself provide a lot of new functionality; rather it combines the use of existing packages in a consistent way, solving problems of compatibility and loading-order. With such problems solved, the full power of packages such as hyperref, texpower, geometry and everyshi can be exploited fully, to produce presentation slides that rival, indeed surpass, what can be achieved with other tools. Example styles, some using elegant Adobe or TrueType fonts, will be shown.

Some new sub-packages are included to make it easy to use alternative fonts (in particular Lucida for mathematics), and to place background images on every page. An extended option-loading mechanism allows for arbitrary extension of the package, and customised document styles.

2.3 Using \TeX to enhance your presentations

HANS HAGEN

Pragma ADE, GH Hasselt, The Netherlands
pragma@wxs.nl
<http://www.pragma-ade.com>

Abstract: The quality of a presentation can be improved by using a decent visualization of the content. For this purpose Con \TeX t comes with a bunch of (currently there are some 25) of presentation styles. In this talk I will demonstrate how you can use \TeX to structure your presentation in such a way that it suits the topic and way you think. I will show that the combination of structure and high end typesetting will give your presentations the finishing touch they deserve. Although making presentations can be done with any \TeX , I will discuss the power of couple Con \TeX t and pdf \TeX .

2.4 Presentations with pdf \TeX using PDFscreen

C. V. RADHAKRISHAN

[PRESENTED BY MARTIN SCHRÖDER]
cvr@focalimage.com

Abstract: Making presentations or on-line documents that involve heavy mathematics and graphics is a non-trivial job using the commonly available software packages. However, we can make use of pdf \TeX , a variant of \TeX that will directly write to a cross platform format, namely PDF, owing to its remarkable math handling and typesetting abilities. In this article, we describe the generations of presentation and on-line documents using the pdfscreen.sty package. As a corollary, we also examine its ability to include pstricks code in a pdf \TeX document (which is not normally possible). A few examples of technical presentations in Malayalam, a language spoken by 30 million people in the southern state of Kerala, India, is also given, in order to demonstrate the versatility of PDFscreen in handling Indic scripts.

2.5 Techniques of Introducing Document-level JavaScript into a PDF file from a \LaTeX Source

DONALD P. STORY
University of Akron, OH, USA
dpstory@uakron.edu

Abstract: The method of introducing document-level JavaScript (DLJS) into a PDF depends on the application used: pdf \TeX or dvipdfm. Until recently, users of Acrobat’s distiller (e.g., users of dvips or dvipsone) did not have the ability to automatically introduce such JavaScript into the document from a \LaTeX source. For user’s of Acrobat 5.0, this situation is, at last, rectified.

This talk will concern the various techniques of inserting DLJS and focus on a new feature of Acrobat 5.0 that can be exploited to produce a convenient method of introducing DLJS into the PDF file from a \LaTeX source.

The latest extensions to the Exerquiz Package, which uses these techniques, will also be discussed.

This talk will be of interest to a general audience, and to package writers who want to exploit the power of Acrobat JavaScript.

2.6 Online self-marking Quizzes, using pdf \TeX

FRANCES GRIFFIN AND ROSS MOORE
Macquarie University, Sydney, NSW, Australia
ross@maths.mq.edu.au

Abstract: In this talk we describe a system for generating self-marking mathematical quizzes, built using pdf \TeX and extensions to Donald Story’s *exerquiz* package. JavaScript embedded within the PDF documents provides not only the self-marking aspects to these multiple-choice quizzes, but also allows the results obtained by a student to be sent to a central server, for recording a log of each student’s attempts.

Solutions are included with the PDF document, but these are not accessible until after the quiz has been attempted. Students request a quiz through a web interface, but a random element ensures that no two quizzes are identical. With the help of a symbolic algebra package, such as *Mathematica*, each quiz has solutions tailored to the specific correct answer, and there is a high degree of plausibility to each incorrect option.

New quizzes can be easily constructed from a database of question types. This allows quizzes to be developed which test either just a single concept repeatedly, or several different concepts within the same quiz. The purpose in developing this system was to provide students with a way to revise basic skills in mathematics, before embarking on new courses at a higher level. Other applications are certainly possible.

2.7 Using pdf \TeX in a PDF-based imposition tool

MARTIN SCHRÖDER
Germany
ms@artcom-gmbh.de

Abstract: pdf \TeX has been used successfully to build an industrial-strength PDF-based imposition tool. This paper/talk describes the pitfalls we encountered and the lessons learned.

2.8 PANEL DISCUSSION: pdf_TE_X

HANS HAGEN (CHAIR)
pragma@wxs.nl

3 Wednesday August 15, 2001

Graphics, XML, and MathML



3.1 TUG BUSINESS MEETING:

TEX USERS GROUP
office@tug.org

3.2 Adobe plug-in for warmreader

TOM RUARK, WENDY MCKAY, AND ROSS MOORE
Macquarie University, Sydney, NSW, Australia
ross@maths.mq.edu.au

Abstract: The WARMreader method of using Xy-pic for placing labels over imported graphics was presented at TUG'99 in Vancouver. Central to this method is the use of a .bb file, which contains mark-up information concerning “marked points” within the graphic, as well as Bounding-Box information.

A new plug-in module has been developed, for use with Adobe Illustrator (vers. 9, and later), which makes it easy to specify the desired marked points, and store the corresponding information within a .bb file. This information is valuable markup which could be used also for other purposes, with other software packages.

In this talk we will demonstrate some possible work-flows for using the new plug-in tool. Also, we will describe work done to convince the Illustrator Development Team, at Adobe Systems Inc., that such a tool would be a simple and useful addition to their software.

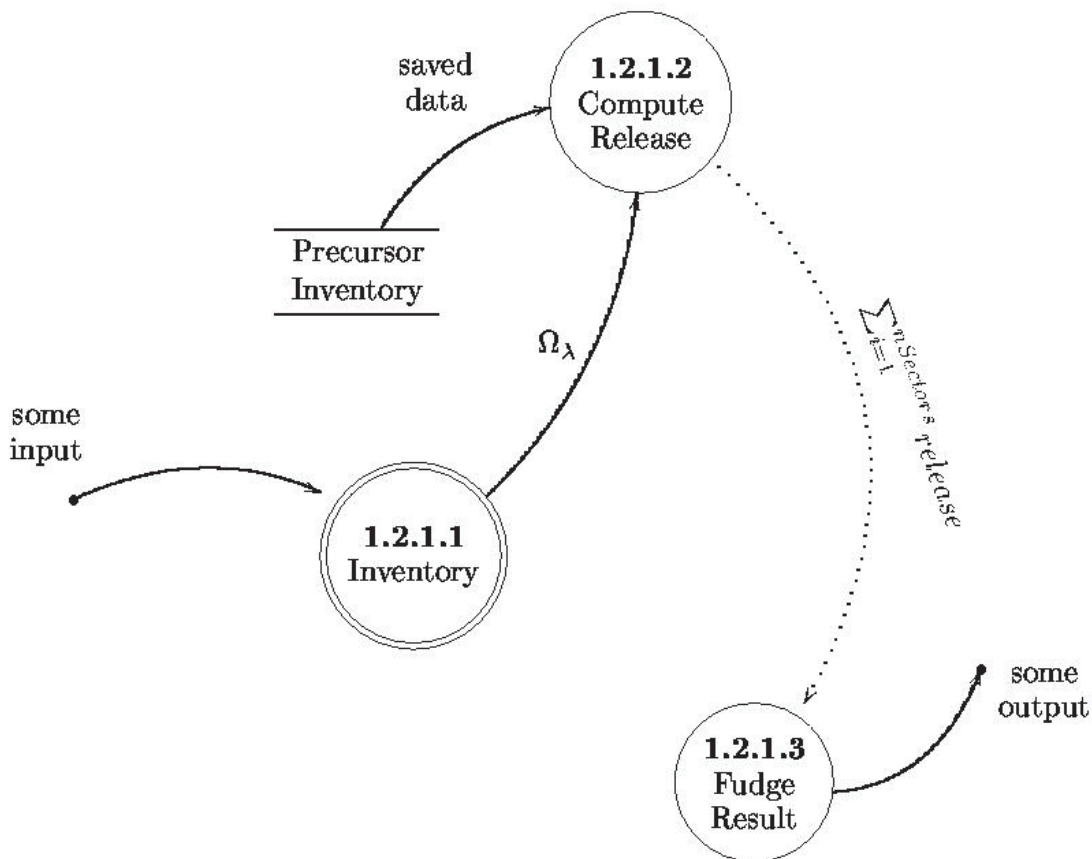
3.3 The TeXspec Tool for Computer Aided Software Engineering

STEPHEN OLIVER
University of Manitoba, Canada
livers@mb.sympatico.ca

Abstract: Commercial tools for Computer Aided Software Engineering (CASE) are common, and implement many development methodologies. Unfortunately, these tools suffer deficiencies when developing software which includes mathematical notation. Also, software development standards such as CSA N286.7 now demand quality assurance and documentation beyond what is commonly supported by the commercial products. The TeXspec tool is being developed for free distribution, and seeks to address some of these problems.

TeXspec supports a single development methodology in the initial release. Derived from the Yourdon/DeMarco method, it produces “Data Flow Diagrams” and “Mini-specs” to document software requirements. Software design is captured in “Structure Charts” and “Module Specifications”. Requirements and Design Data Dictionaries are separate, but design information can be inherited from requirements if they happen to be equivalent. All products can feature mathematical notation, with consistency between products verified automatically. Although the Module Specification can use free form LaTeX, support is included for Noweb syntax, permitting literate programming practice.

The presentation begins with a quick overview of software development using a CASE tool. This is followed by a discussion of some of the problems with the commercial tools, and the way that TeXspec addresses them. Strengths of the commercial tools over TeXspec will also be examined. Examples of TeXspec diagrams and documentation products will be presented, and user input requirements will be discussed. The current state of TeXspec development will be addressed.



3.4 GELLMU: A Bridge for Authors from \LaTeX to XML

WILLIAM F. HAMMOND
 University of Albany, NY, USA
 hammond@csc.albany.edu

Abstract: GELLMU, which stands for “Generalized Extensible \LaTeX -Like Markup”, is a system for using \LaTeX -Like Markup, though not \LaTeX itself, to write consciously for a markup language in the SGML category or in its popular XML subcategory.

- The “basic” level of GELLMU offers a way to use \LaTeX -Like notation together with a \LaTeX -Like “newcommand” (with arguments) macro facility to write web pages.
- The “regular” level of GELLMU enables one additionally to incorporate certain \LaTeX -Like features, such as the use of a blank line for a new paragraph, in writing for an SGML language.
- The didactic GELLMU production system provides an “article” XML language, with some resemblance to \LaTeX itself, that is a rigorous domain for translation to other formats.

For more information see <http://www.albany.edu/~hammond/gellmu/> on the World Wide Web.

3.5 WORKSHOP: Creating Math Web Documents

BOB CAVINESS
University of Delaware, USA
caviness@udel.edu

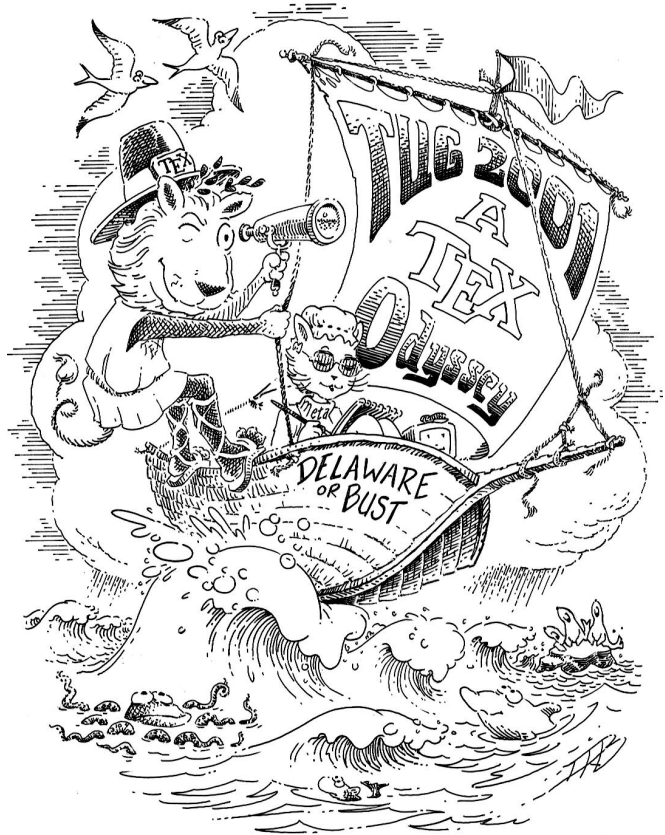
3.6 Online DEMO: Processing mathml with \TeX

HANS HAGEN
Pragma ADE, GH Hasselt, The Netherlands
pragma@wxs.nl
<http://www.pragma-ade.com>

Abstract: Although mathematicians differ in their enthusiasm for coding in a different language other than \TeX , it is to be expected that MATHML will play an important role in the world dominated by angle brackets. In this talk, I will discuss the differences between coding in \TeX and coding in presentational and content MATHML. I will also discuss the importance of the need for control over the visual output when MATHML is used for producing print. Finally, I will demonstrate an online tool for playing with MATHML based on a MATHML interpreter and typesetting engine written in \TeX .

4 Thursday August 16, 2001

Fonts and Tools



4.1 Typesetting Hebrew with T_EX

ALAN HOENIG
City University of New York
ajhjj@cunyvm.cuny.edu

Abstract: I'd like to discuss a new system for typesetting Hebrew with T_EX, in the context not so much of Hebrew but of how you get T_EX to cope with the intricacies of a non-Latin alphabet. From a T_EXnical point of view, Hebrew is a language in which some letters have special forms for word-endings only, which has certain sounds which don't appear in English, and in which vowels which appear as diacritical marks. The challenge with these vowels is that they must be centered not with respect to the geometric center of the character, but with respect to a visual center which varies from character to character (and from font to font). T_EX can be persuaded to deal with all these issues, and the hope is that these methods can be adapted to other non-Latin alphabets.

4.2 Modernizing Computer Modern

ALAN HOENIG
City University of New York
ajhjj@cunyvm.cuny.edu

Abstract: The Computer Modern fonts are based on a pre-existing font called Modern 8a. “Based on” does not mean “identical to,” and in my opinion, the originals do an even better job of drawing a reader into a document than do the CM fonts. I plan on discussing methods to make it possible to use the Monotype Modern fonts, which seem to be the sole extant Modern type available in digital format, more usable to T_EX and L^AT_EX authors.

4.3 PANEL DISCUSSION: Fonts

NELSON BEEBE (CHAIR)
beebe@math.utah.edu

4.4 Managing Multiple TDS Trees

MICHAEL J. DOWNES
American Mathematical Society
mjd@ams.org

Abstract: Did you ever take on the task of editing a file named `texmf.cnf`? Did you find it as easy as falling off a log? Or did you find it about as easy as a log falling on your head—in the case of multiple TDS trees, more than one log? In the latter case, you might like to hear about my own quest to make sense out of the labyrinth of TDS/teTeX/kpathsea configuration.

The TDS specification provides a good structural framework for organizing a single TEXMF tree such as the one at the heart of Thomas Esser’s teTeX distribution. In practice, however, because of the normal processes of local changes and ongoing upgrades it is often desirable to set up a more elaborate system of multiple TDS trees, one that organizes the pieces that have been added above and beyond the original base tree in a way that makes it easier to cope with continuing change.

4.5 Installing a CTAN mirror on your desktop

MICHAEL DOOB
University of Manitoba, Winnipeg, Manitoba, Canada
mdoob@umanitoba.ca

4.6 Installing T_EXshop

RICHARD KOCH

University of Oregon, Eugene, OR, USA

koch@math.uoregon.edu

Abstract: T_EXShop is a T_EX previewer for Mac OS X. According to the TUG program, this session is titled “Installing T_EXShop” (45 minutes). If it takes 45 minutes to install T_EXShop, then Unix has truly come to the Macintosh.

I’ll give a brief demonstration of T_EXShop, explain how to add fonts and files to the t_EX distribution used by T_EXShop, and speak about future directions. Participants should bring lists of features they’d like to see added to the program, and complaints about the current version.

4.7 Font Installation: Agfa Eaglefeather to Linotype Zapfino

WILLIAM ADAMS

ATLIS Graphics & Design, Camp Hill, PA, USA wadams@atlis.com

Abstract: Font Installation (Using David Siegel and Carol Toriumi-Lawrence’s digital version of Frank Lloyd Wright’s lettering style for his Olive Hill project which was named Eaglefeather as well as Prof. Hermann Zapf’s Zapfino which is bundled with Mac OS X)

- Acquiring fonts, purchasing, or downloading: reputable sources for good quality fonts with acceptable imprimatur, guidelines for ethical selection of fonts—do you know who made that font and why?
- Converting fonts (if necessary): Mac → PC/Unix and elsewhere, tools to use (or not use) for this
- Macros and Tools (FontInst and vfont): automatic creation of Virtual Property Lists (vpl) and associated files
- Tweaking: adjusting a vpl to add additional ligatures and other font features
- Installation: including batch processing of (for example) `vp12vf &c` and where to put files
- Testing: a test suite of TeX macros and files to put a font through its paces, and look for errors, or aspects of it which may be improved upon through further tweaking
- Font Samples: an expansion upon Stephen Moye’s nifty TeX sample macros on CTAN

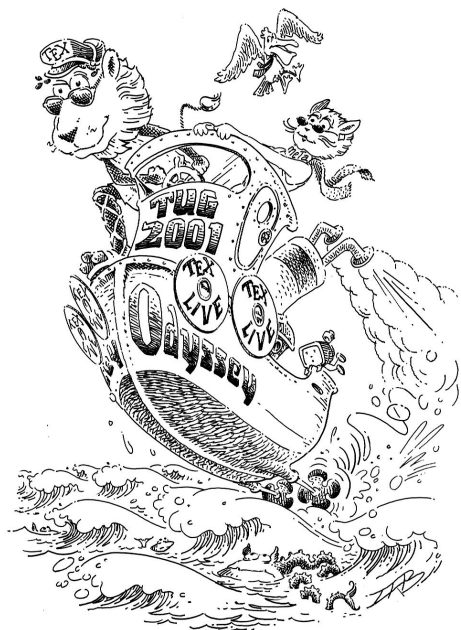
4.8 CLOSING & WRAP-UP: TUG2002—Kerala, India

KAVEH BAZARGAN (TUG 2002 COMMITTEE, CHAIR)

PRESENTED BY ROSS MOORE

kaveh@focalimage.com

ORGANIZERS



CONFERENCE: *Sue DeMeritt, Patricia Monohon*
PROGRAM: *Anita Hoover* (co-chair), *Stephanie Hogue* (co-chair)
PUBLICITY: *Wendy McKay, Heidi Rhodes Sestrich*
PROCEEDINGS: *Patrica Monohon* (editor), *Anita Hoover* (technical)
L^AT_EX COURSE: *Sue DeMeritt* (Organizer), *Cheryl Ponchin* (Instructor)
E-MAIL: tug2001@tug.org
URL: <http://www.tug.org/tug2001/>

Acknowledgements

ILLUSTRATOR-CARTOONIST, T_EX LION ART: *Duane Bibby*
DBibby@aol.com
TUG SYSTEM ADMINISTRATORS: *Kaja Christiansen and Karl Berry*
<http://www.tug.org/>
TUG OFFICE MANAGER: *Robin Lasko*
Tel +1 503-223-9994; Fax +1 503-223-3960;
E-mail office@tug.org



TUG 2001 SUMMARY

THE 22ND ANNUAL MEETING OF THE T_EX USERS GROUP

PROGRAM

August 12–16, 2001

University of Delaware · Newark · DE · USA

SUNDAY AUGUST 12	MONDAY AUGUST 13	TUESDAY AUGUST 14	WEDNESDAY AUGUST 15	THURSDAY AUGUST 16
REGISTRATION	OFFICIAL OPENING	HÀN THẾ THÀNH	TUG BUSINESS MEETING	ALAN HOENIG
WELCOME RECEPTION	HANS HAGEN	ROSS MOORE	BREAK	ALAN HOENIG
	MIMI JETT	HANS HAGEN	ROSS MOORE	BREAK
	BREAK	C. V. RADHAKRISH- NAN	STEPHEN OLIVER	PANEL: FONTS NELSON BEEBE (chair)
	WILLIAM RICHTER	LUNCH	WILLIAM HAM- MOND	MICHAEL DOWNES
	KAVEH BAZARGAN	DON STORY	LUNCH	LUNCH
	LUNCH	ROSS MOORE	BOB CAVINESS	MICHAEL DOOB
	HANS HAGEN	BREAK	BREAK	RICHARD KOCH
	PETER FLYNN	MARTIN SCHRÖDER	HANS HAGEN	WILLIAM ADAMS
	BREAK	PANEL: pdfT _E X HANS HAGEN (chair)	BANQUET	BREAK
	DAVID TULETT			CLOSING AND WRAP-UP: TUG 2002, Kerala
	JONATHAN FINE			
	ART OGAWA			

PRE-CONFERENCE COURSES

AUGUST 6–10, 2001

Intermediate/Advanced L^AT_EX