TUG 2024 abstracts

Editor's note: Links to videos and other information are posted at `tug.org/tug2024`.

– – ∗ – –

Vincent Goulet
A journey through the design of (yet another) journal class
Many scientific journals rely on document classes provided by large publishers. Yet, some journals still prefer to maintain their own class. The *Canadian Journal of Statistics | La revue canadienne de statistique* is one of them. In late 2022, I was commissioned to develop a class and bibliography styles that would update not only the production underpinnings, but also the visuals of the CJS. This presentation will provide a journey through the choices that I had to make along the way, the functionality that I imported from various sources, and the (hopefully) novel solutions I implemented.

Vincent Goulet
You (S)wove? Well, (S)tangle now!
The concept of literate programming, pioneered for TEX by Donald Knuth in the late 1970s, should be familiar to many in the (LA)TEX community. Very briefly stated, it consists of a programming paradigm where a program and its documentation are interspersed in a single file. Source code and documentation are extracted respectively by `tangle` and `weave` procedures. Literate programming plays a central role nowadays in scientific computing for reproducible research purposes: instead of being hard coded into a report, results and graphics are woven in place using the computer code. In the R ecosystem, Sweave and knitr are widely used to build documents this way from R code. The aim of this presentation is to shed some light on the perhaps lesser-known component of literate programming, at least in scientific computing: the `tangle` step. I will describe a use case where a clever combination of weaving and tangling allows for efficiently maintaining a set of exercises and solutions.

Sarah Lang
LaTeX in the Digital Humanities
This talk explores the intersection of LaTeX typesetting and the digital humanities. More specifically, it asks why LaTeX typesetting is used so infrequently in the digital humanities, despite its clear advantages and applications. This talk, on the one hand, aims to understand the reasons why LaTeX isn't more widely used in the digital humanities and, on the other hand, presents three examples of relevant use cases to illustrate its value for the (Digital) Humanities.

Conference proceedings that require full paper submissions, exemplified by the Computational Humanities Conference, are becoming more prevalent in the Digital Humanities and thus, present a good reason to get acquainted with the necessary LaTeX skills. This talk challenges the notion suggested by Quinn Dombrowski that learning LaTeX is too much to ask of humanities scholars. Humanities scholars, even those with minimal technical background, can learn LaTeX at the necessary level to format their submissions effectively, by using it essentially as a markup language.

The second and most important use case for LaTeX in the Digital Humanities is in digital scholarly editing. There is a sizable digital scholarly editing community within the field. One could even say that digital scholarly editing is one of the core fields of work for digital humanists. Digital scholarly editing is a domain where the field initially flourished and continues to thrive, despite recent developments involving deep learning driven approaches and Large Language Model rendering the subfield of the "Computational Humanities" more and more dominant. Digital scholarly editing will continue to be a core task and technology within Digital Humanities; thus, LaTeX will continue to remain relevant to the Digital Humanities.

Despite some early predictions that digital media would make physical books obsolete, we observe in 2024 that this transformation has not yet materialized and seems unlikely to occur in the near future. On the contrary, the value of the book as a material object and a symbol of cultural capital has surged, especially among younger audiences on platforms like BookTok. While this trend might seem irrelevant to academia per se, it underscores an important point: physical books retain their importance to this day and continue to be favored by textual scholars for various tasks, such as in-depth reading. This enduring preference for printed materials highlights the need of ensuring that digital humanities resources, such as digital scholarly editions, can be effectively translated into print when necessary. Transforming digital scholarly editions in TEI-XML format to LaTeX allows us to produce print versions from digital editions, even when the primary intent of the edition is not to publish in print.

Through the transformation of TEI data using XSLT, potentially even employing large language models to generate LaTeX code, scholars can create high-quality printed materials with little effort. This functionality is crucial for the digital humanities community, as it responds to the frequent preference for accessible, printable formats, which are particularly

doi.org/10.47397/tb/45-2/tb140abstracts

useful in teaching settings. While digital editions offer unique benefits, like showcasing multiple witnesses or versions without prioritizing a single one, there remains a substantial demand for printed copies invaluable for detailed study, annotation, or instructional use. The `reledmac` package is a particularly useful tool for Digital Humanists.

The presentation also highlights the utility of LaTeX in managing complex documents within the humanities, such as archaeological catalogs. In fields like archaeology, it is commonplace to compile extensive catalogs of objects or findings as integral components of larger research projects, such as Ph.D. dissertations. Such catalogs often contain numerous images and can become unwieldy when managed with standard word processing software like Microsoft Word due to their size. Although this application is not exclusively within the digital humanities, it underscores the relevance of LaTeX across broader humanities disciplines, as has been discussed previously on the LaTeX Ninja blog.

Frank Mittelbach
Hooks, sockets and plugs
Driven by the need to support tagging, a number of ideas are being introduced into the LaTeX kernel to allow more flexible changes to code paths, design aspects and document command creation. Hooks were introduced a few releases ago, and provide a way to manage the interaction between packages in a flexible and powerful way. More recently, we brought in sockets and plugs: places where exactly one code path is needed, but what that code path is needs to be swappable. In this talk, we'll look at why we need both sockets and hooks, and which to use when.

Wim Obbels, Bart Snapp, Jim Fowler
Ximera interactive math educational resources for all: From LaTeX source code to PDF, HTML and beyond
Ximera is an open-source platform to create online interactive STEM courses using LaTeX as the source code. Initially developed at the Ohio State University (OSU), now around a dozen institutions are using Ximera materials. Backend development of the project has extended from just OSU to now include educators at the University of Florida and KU Leuven.

In this talk we will explain the basic ideas of how Ximera works and demonstrate how the currently available courses easily implement interactive questions, integrate applets from youtube, desmos or geogebra, hyperlinks, etc., and how this, based on TeX4ht, rather smoothly translates between HTML and PDF.

After installing the Ximera package from CTAN, one can immediately generate PDF versions. To publish an online version an extra build environment (`xake`) is needed which enables publishing courses to a public server at `ximera.osu.edu` or optionally to a self-hosted server. Ximera can be integrated with learning management systems, collects useful learning analytics, and can return results to a gradebook.

It is currently under active development, getting extra functionality, more examples and tutorials, options for styling courses both online and in PDF, and integrated gradebook functionality. A docker setup will soon be available for easy deployment, and even a serverless setup with compilation-in-the-browser.

The CI/CD integration currently used at KU Leuven will be explained, which automatically updates the online content and all the corresponding PDFs whenever authorized authors git-push changes.

Interested users can learn more at `github.com/ximeraProject/ximeraFirstSteps`.

Martin Ruckert
The color concept of HiTeX
One of the first feature requests for HiTeX was colored text. Looking at the existing packages supporting colored text, it appears that their design was influenced if not limited significantly by the features provided by the TeX engine, like `\special`, and the features provided by the output format, namely PDF.

Since HiTeX has its own engine, its own output format, and its own viewing applications, it was possible to start a new design from scratch without such limitations, but not without conflicting design objectives. The new design should be simple, flexible, and powerful; it should be easy to understand and easy to implement; and it should be possible to support the existing color packages with only a limited implementation effort for a driver file.

It is planned to include the color support for HiTeX in the 2025 edition of TeX Live. I welcome feedback and suggestions for changes.

samcarter
The moloch beamer theme
The moloch beamer theme (`ctan.org/pkg/moloch`) by Johan Larsson is an updated fork of the well-known `metropolis` theme, which was originally written by Matthias Vogelgesang. In this short talk, I'm going to quickly introduce the theme and then show, from the perspective of a user, how to switch from the metropolis theme to the new moloch theme.

Tyge Tiessen
Rewriting TeX today
TeX82 was written forty years ago. The constrained memory resources as well as portability concerns

and compiler limitations of that time have set strict bounds to the original implementation. As some consequences of these bounds TEX82 features handwritten dynamic memory management, ubiquitous use of global variables, and manual string management.

In this talk, I will discuss the joys and challenges of rewriting TEX82 today without these limitations.

Jan Vaněk, Hân Thế Thành
Exploring Primo: A developer's perspective
In this session, we'll take a practical look at Primo from the viewpoint of developers. We'll examine its technology stack, code structure, and document model, providing insights into its implementation for collaboration.

We'll delve into the front-end development aspects, covering the usage of the VDL JavaScript library for UI components and the related CSS code. Additionally, we'll discuss the deployment process and touch upon the importance of TEX compilation and XML validation in the Primo system.

Didier Verna
A couple of extensions to the Knuth-Plass algorithm
In this talk, we will present our implementation of a couple of extensions to the Knuth-Plass algorithm in ETAP, our experimental typesetting algorithms platform.

Joseph Wright
`siunitx` development continues: 2024
The `siunitx` package was first released in 2008, and has been through three major revisions; v3.0.0 was released in May 2021. Since then, development of new ideas has continued, with new features in numbers, tables and units. Here, I will pick out some highlights from the past three years of work, and look at where we might see additional ideas in the future.

Joseph Wright
Templates: Prototype document elements
Controlling design is something that has been challenging in LATEX to date. While the LATEX Team developed experimental ideas in the mid-1990s for creating flexible design elements, they were not viable for real documents then. These ideas, based around 'templates', have now reached maturity and are included in the Summer 2024 LATEX kernel release.

In this talk, I will look at what a template is, why we'd want to use them and the flexibility and power they will bring to controlling document design.

***Die TEXnische Komödie* 2/2024**

Die TEXnische Komödie is the journal of DANTE e.V., the German-language TEX user group (`dante.de`).

Die TEXnische Komödie 2/2024

[Received from Uwe Ziegenhagen.]