TEX Development Fund 2014–2019 report

TEX Development Fund committee

These projects (listed by reference number) have been supported in recent years by the TEX Development Fund. For application and donation information, the complete list of supported projects, and more, please see: https://tug.org/tc/devfund

32. HarfTEX engine based on LuaTEX

Applicant: Khaled Hosny, Egypt.
https://github.com/khaledhosny/harftex
Amount: US$1000; acceptance date: 1 May 2019; completed 9 September 2019.

HarfTEX is a TEX engine based on LuaTEX, extending it with HarfBuzz, ICU and possibly other libraries for Unicode text layout and modern fonts support.

The engine will be extended to offer more libraries, and to fix some of the limitations faced during the previous stage of development.

This project has been adapted into the LuaHBTEX engine, now maintained as part of the LuaTEX sources by Luigi Scarso.

31. HarfBuzz access from LuaTEX

Applicant: Khaled Hosny, Egypt.
Amount: US$2000; acceptance date: 1 November 2018; completed 17 April 2019.

Provide a set of Lua modules that bridge LuaTEX and HarfBuzz (possibly/eventually also ICU, FreeType, and FontConfig), and bundle these modules with LuaTEX to extend its functionality with the ability to typeset more world languages and scripts. Also provide Lua code that integrate these modules with LuaTEX callbacks.

30. Documentation in Persian of the core TEX system

Applicant: Vafa Khalighi, Australia.
Amount: US$1000; acceptance date: 10 October 2018.

Comprehensive documentation in Persian of TEX, Metafont, and Computer Modern, including WEB programming, Pascal programming (as used in *.web), and other material.

29. Light (LA)TEX Make

Applicant: Takuto Asakura, Japan.
https://github.com/github.com/wtsnjp/llmk
Amount: US$1000; acceptance date: 25 July 2018; completed 30 August 2019.

The procedure of building is essential for LATEX documents to get the correct output, as the creator intends. The procedure of building includes the information like which TEX engine to use, which outer programs to use, what options and arguments and/or which configuration files should be given for each program, and the processing order of the programs.

The proposed features of the llmk program are as follows (all subject to change with more experience):

- Working solely with Lua on LuaTEX (texlua). Neither external programs nor any Lua libraries from third parties are required.
- Using TOML to declare the settings. TOML (https://github.com/toml-lang/toml) is a minimal configuration format similar to JSON and YAML. I selected TOML because it is human readable and its specification is relatively simple, making it easy to write a parser from scratch.
- No complicated nesting of configuration. llmk allows users to write configuration (in TOML format) in the source of LATEX documents or the special file (named llmk.toml). Since the procedure of building is essential and independent for each project, llmk does not load the configuration files recursively.
- Modern default settings. llmk will try to make a modern de facto standard. For instance, if an user doesn't specify any TEX engine to use, llmk will use LuaTEX to compile the source.

Current status and project goals:

- The most basic functionality of llmk is already implemented (see the README at the project page for the details). In the year, I'm going to enhance the program including:
- full support for TOML specification (currently, only partially supported)
- supporting some types of magic comments (e.g., those of TEXworks)
- full documentation (currently, only README)
- other convenient features (e.g., quiet mode and an option to support specifying an output directory)

28. MetaPost updates

Applicant: Luigi Scarso, Italy.
Amount: US$1000; acceptance date: 28 December 2015; completed 24 May 2017.

Bug fixes and maintenance for MetaPost path resolution, binary, decimal, and double number systems. Also, more efficient integration with mplib by excluding the libraries needed only in LuaTEX. A report from May 2017 is available at https://tug.org/devfund/documents/2017-05-scarso.pdf.

27. Libertine OpenType math fonts

Applicant: Khaled Hosny, Egypt.
https://github.com/khaledhosny/libertinus
Amount: US$2000; acceptance date: 24 December 2014; completed 1 February 2016.

Building an OpenType math companion for Linux Libertine and Linux Biolinum fonts (ultimately named Libertinus), and fixing bugs in those fonts. Also coordinating with the authors of Linux Libertine (LA)TEX support files to adapt the new and fixed fonts (ultimately they did not respond, so upstream Linux Libertine is unmaintained as far as we can tell). The package has been released to CTAN: https://ctan.org/pkg/libertinus-otf. ◇