LaTeX3 News

Issue 2, June 2009 (LaTeX release 2009-06-01)

TeX Live and the expl3 code

TeX Live 2009 is almost upon us, and the LaTeX3 team have been readying a new release of the experimental LaTeX3 code for this. Very dramatic changes have occurred since the last public release of the code in TeX Live 2008; no backwards compatibility has been maintained (as warned in the beginning of the documentation) but we believe the changes made are all much for the better. Almost every single part of expl3 has been scrutinized, resulting in a far more coherent code base.

The expl3 code is now considered to be much more stable than it was before; a comprehensive test suite has been written that helps to ensure that we don't make any mistakes as we change things in the future. In the process of writing the test suite, many minor bugs were fixed; we recommend such test suites for all similar developmental projects! Some small underlying changes are still expected in the expl3 code, but major, disruptive, changes aren't planned.

Planned updates

Until now, the last update to CTAN of the expl3 bundle was for TeX Live 2008. Now that work on the code is happening on a semi-steady basis, we plan to keep updates rolling out to CTAN more frequently. This will allow anyone who wishes to experiment with the new code to use the TeX Live or MiKTeX updaters to install a recent version without having to 'check out' the SVN repository and install the packages manually.

New members

We didn't say anything about it in the last status update, but Joseph Wright and Will Robertson are now members of the LaTeX Team. They have been working fairly exclusively on the expl3 code.

It's worth repeating that LaTeX 2$_\varepsilon$ is essentially frozen in order to prevent any backwards compatibility problems. As desirable as it is to benefit from the new features offered by new engines XeTeX and LuaTeX, we cannot risk the stability of production servers running older versions of LaTeX 2$_\varepsilon$ which will inevitably end up processing documents written into the future.

LaTeX3 will not be inheriting the same restraints, so stay tuned.

Some specifics

Morten Høgholm will be presenting the recent changes in much more detail at TUG 2009. Here are some quick specifics for those interested. New code written and broad changes made to the expl3 modules:

More logical function names Many function names that were hold-outs from the TeX naming system have been changed to fit into the more logical scheme of expl3; e.g., `\def:Npn` and `\let:NN` are now `\cs_set:Npn` and `\cs_set_eq:NN`.

Defining functions and conditionals Much thought was put into new ways to define functions and conditionals with a minimum of code. See `\cs_set:Nn` and `\prg_set_conditional:Nnn`.

Smart comparisons Comparisons can be made much more easily now, with familiar notation such as `\intexpr_compare_p:n{ #1+3 != \l_tmpa_int }`.

Data from variables A new function argument specifier `V` has been added for extracting information from variables of different types, without needing to know the underlying variable structure. Some other tidy-ups on the argument specifiers offered, partially as a result of the addition of this new one.

l3msg New module to deal with communication between LaTeX3 code and the user (info messages, warnings, and errors), including message filtering partially inspired by the silence package.

The next six months

Having overhauled the expl3 code, we now plan to perform an analogous process with the foundations of the xpackages. These are the higher-level packages that will provide the basic needs such as control of the page layout and rich document-level interaction with the user. As the groundwork for this layer of the document processing matures, we will be able to start building more packages for a LaTeX3 kernel; these packages will also be usable on top of LaTeX 2$_\varepsilon$ and serve as broadly customisable templates for future document design.

As gaps in the functionality offered by expl3 are found (in some cases, we know that they exist already), the programming layer will be extended to support our needs. In other cases, wrappers around TeX functions that can be more usefully handled at a higher level will be written.

LaTeX3 News, and the LaTeX software, are brought to you by the LaTeX Project Team; Copyright 2009, license LPPL.

In terms of what we're planning to work on next, three xpackages will take the focus of our attention.

xbase 'xbase' is actually two packages: xparse and template. These contain code for, respectively, defining new document commands (such that a user would use; e.g., `\section`, `\makebox`, ...) and for handling keyval lists for user input and document specification. xparse was presented at TUG 1999[1] and Lars Hellström wrote some notes on template in 2000[2]. Functionality coverage for these packages is good but concepts need a good "airing". There are various approaches taken for keyval input, some more recent than the template code, so there are some alternatives to evaluate.

galley2 Sophisticated handling for constructing paragraphs and other document elements. Morten spoke on this at TUG 2008[3]. Design needs to be revisited after some stress testing.

xor This is the LaTeX3 output routine for splitting the galley into page and sub-page sized chunks. Ideas and code need work to move to "production ready" status. Early developments with this package were published by Frank in 2000[4].

Expect to hear again from us at Christmas. If you'd like to discuss any of these ideas, please join us on the LATEX-L mailing list[5].

[1] `http://www.latex-project.org/papers/tug99.pdf`
[2] `http://www.latex-project.org/papers/template-notes.pdf`
[3] `http://river-valley.tv/the-galley-module/`
[4] `http://www.latex-project.org/papers/xo-pfloat.pdf`
[5] `http://www.latex-project.org/code.html`