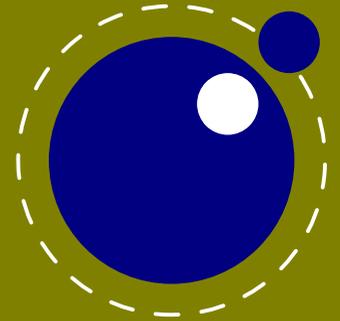


LuaTeX

Version 1.00

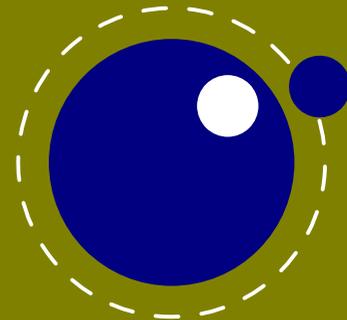
ConTeXt meeting — September 2016



After ten years of stepwise development and experimenting we release version 1.00 of LuaT_EX during the 10th ConT_EXt meeting in the Netherlands, September 2016.

The interface is now rather stable and will not change significantly which means that one can write stable packages.

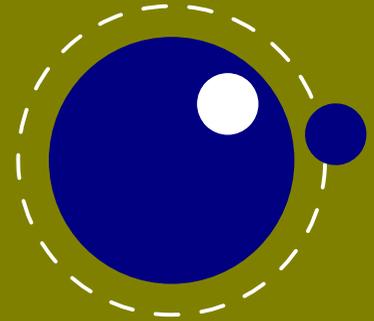
So, it's time for a bit reflection as well as time to tell what we will be doing next.



luatex
1.00 - 2016

Around 2005, after we talked a bit about it, Hartmut added the Lua scripting language to pdfT_EX as an experiment.

This add-on was inspired by the Lua extension to the Scite editor that I (still) use.

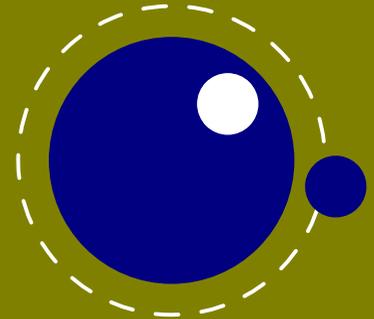


luatex
1.00 - 2016

One could query counter registers and box dimensions and print strings to the T_EX input buffer.

The Oriental T_EX project then made it possible to go forward and come up with a complete interface.

For this, Taco converted the code base from Pascal to C, an impressive effort.

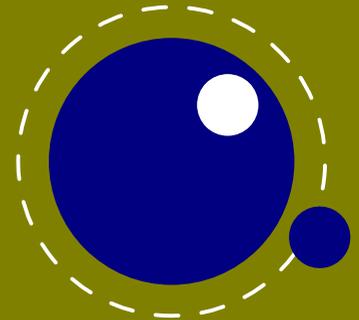


luatex
1.00 - 2016

We spent more than a year intensively discussing, testing and implementing the interface between T_EX and Lua.

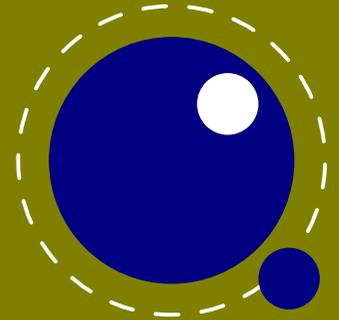
In successive years we polished things and extended bits and pieces.

The last few years we cleaned up, filled in gaps and reached the point where we were more or less satisfied.



luatex
1.00 - 2016

The core is still traditional $\text{T}_{\text{E}}\text{X}$, but extended with pdf $\text{T}_{\text{E}}\text{X}$ protrusion and expansion (reworked) and directional features from Aleph (cleaned up).



luatex
1.00 - 2016

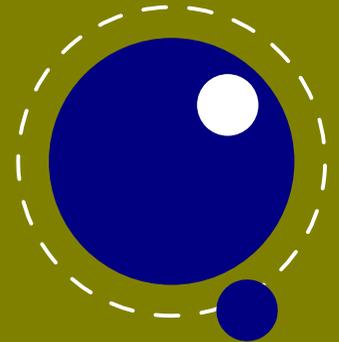
The font subsystem
accept now wide fonts.

The hyphenation machinery
can use runtime loaded
(and extended) patterns.

Hyphenation, ligaturing,
kerning are separated.

Most steps in processing node lists
can be intercepted using callbacks.

The math machinery has
opentype math code paths.



luatex
1.00 - 2016

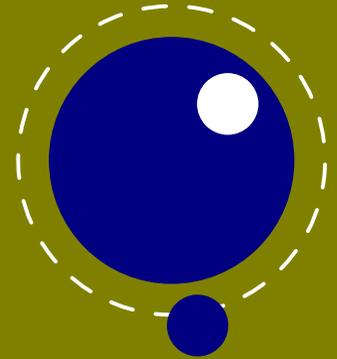
All in- and output can be controlled and intercepted.

The backend code has been separated better.

You can write (simple) parsers.

Nodes can be accessed and manipulated.

Images and reuseable boxes are now native.



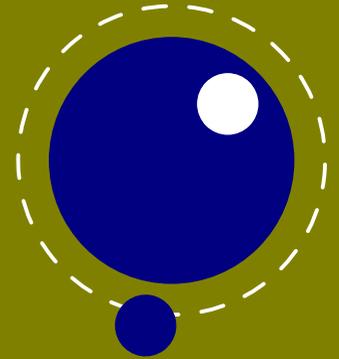
luatex
1.00 - 2016

The project is driven by ConT_EXt users and ConT_EXt development.

Right from the start
ConT_EXt supported LuaT_EX.

This means that most mechanisms
have been tested in production.

Raw performance is less than 8 bit
pdfT_EX but in practice and on modern
machines LuaT_EX behaves well.

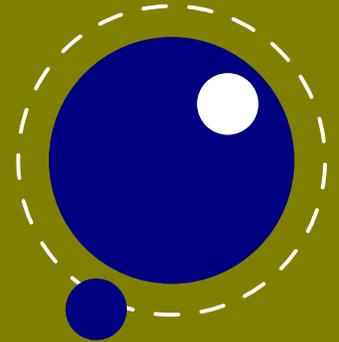


luatex
1.00 - 2016

We will continue development, but functionality will stay stable within versions. Of course bugs will be fixed.

The code will be further streamlined and documented. We deliberately postponed some cleanup till after version 1.00.

Of course the manual will be improved over time.



luatex
1.00 - 2016

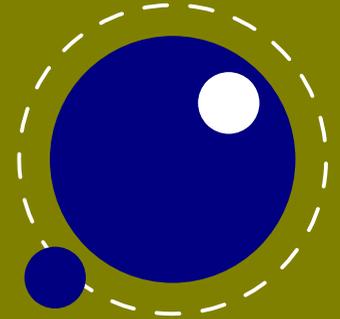
Hans Hagen

Hartmut Henkel

Taco Hoekwater

Luigi Scarso

**many thanks to all the
early adopters**



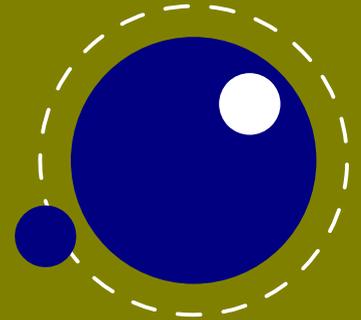
luatex
1.00 - 2016

Some ideas (1)

So far we managed to avoid extensions beyond those needed as part of the opening up.

We stick close to Don Knuths concepts so that existing documentation still conceptually applies. We keep our promise of not adding to the core.

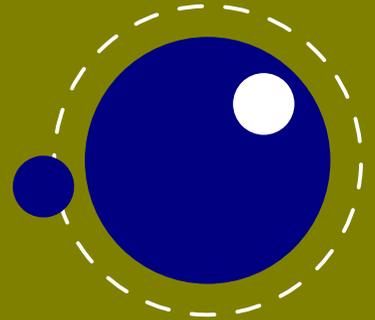
We might open up (make configurable) some of the still hard coded properties.



luatex
1.00 - 2016

Some ideas (2)

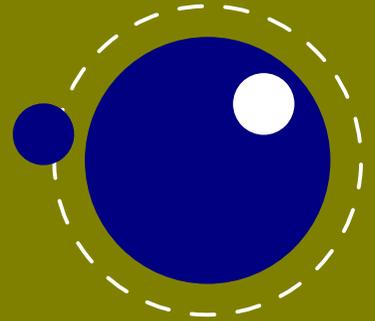
Some node lists can use a bit of (non critical) cleanup, for instance passive nodes, local par nodes, and other left-overs. Maybe we should add missing left/right skips.



luatex
1.00 - 2016

Some ideas (3)

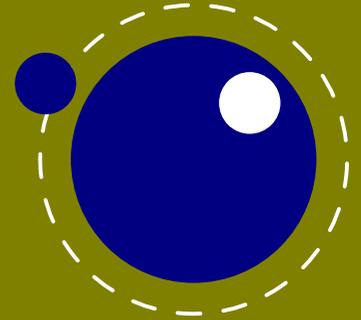
We can optimize some callback resolution (more direct) so that we can gain a little performance.



luatex
1.00 - 2016

Some ideas (4)

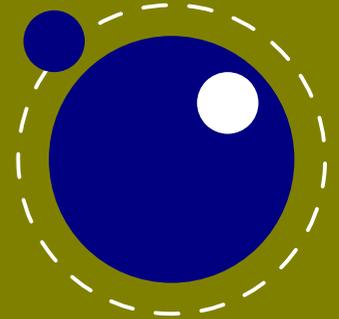
Inheritance of attributes needs checking and maybe we need to permits some more explicit settings.



luatex
1.00 - 2016

Some ideas (5)

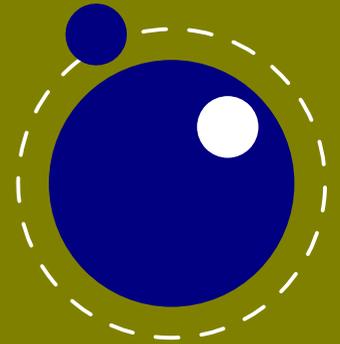
Bring some more code to the api file. Use the global PDF and Lua states consistently. Some macros can probably go away.



luatex
1.00 - 2016

Some ideas (6)

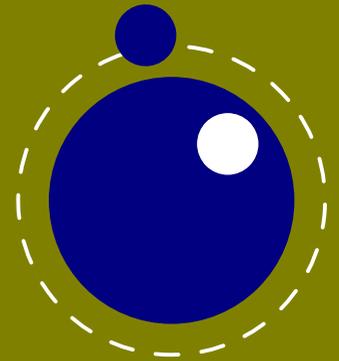
Minimize return values of Lua functions; only return nil when we expect multiple calls in in one line.



luatex
1.00 - 2016

Some ideas (7)

Figure out a way to deal with literals in virtual characters (relates to font switching in the result).

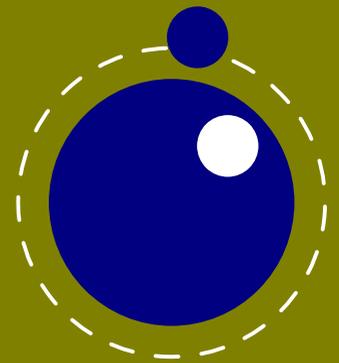


luatex
1.00 - 2016

Some ideas (8)

Maybe reorganize some code so that documentation is easier.

See if we can stick close to what Don Knuth documents.

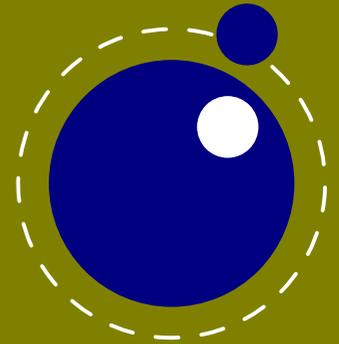


luatex

1.00 - 2016

Some ideas (9)

Cleanup and isolate the backend a bit more. Maybe add a bit more options to delegate to Lua. Get rid of some historic PDF artifacts.



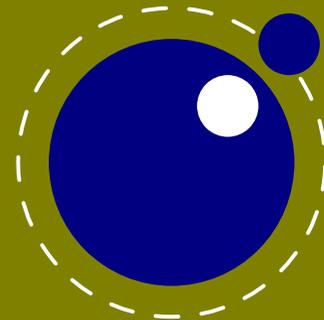
luatex
1.00 - 2016

Some ideas (10)

It is tempting to think of a (lean and mean) LuaT_EX variant for ConT_EXt.

We will not touch stable unless it concerns bug fixes, but we will expose ConT_EXt users to the experimental branch (as we do now).

So ... be prepared.



luatex
1.00 - 2016