End March we released LuaTeX 0.95, the first official ‘stable’ release, a prelude to 1.00 later this year.

The interface will not change significantly so that one can write stable packages.

We have used LuaTeX ourselves for over a decade in order to figure things out.
Around 2005, after we talked a bit about it, Hartmut added the Lua scripting language to pdfTEX as an experiment.

This add-on was inspired by the Lua extension to the Scite editor that I (still) use.
One could query dimensions, counter registers and box dimensions and print strings to the \TeX{} input buffer.

The Oriental \TeX{} 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.
We spend more than a year intensively discussing, testing and implementing the interface between \TeX{} 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 of less satisfied.
The core is still traditional \TeX, but extended with pdf\TeX protrusion and expansion (reworked) and directional features from Aleph (cleaned up).
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.
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.
The project is driven by ConTeXt users and development.

Right from the start ConTeXt supported LuaTeX.

This means that most mechanisms have been tested in production.

Raw performance is less that 8 bit pdfTeX but in practice and on modern machines LuaTeX behaves well.
We will continue development, but functionality will stay stable within versions.

The code will be further streamlined and documented.

The manual will be improved over time.
Hans Hagen
Hartmut Henkel
Taco Hoekwater
Luigi Scarso
In just over a decade of development and extensive usage we added a couple of new primitives.

Also over time many helpers were introduced at the Lua end that access TeX properties and manipulate lists.

The next pages only mention recent additions and changes.
Forms and images are now stored in rule nodes and not longer extension whatsits:

\saveboxresource
\saveimageresource
\useboxresource
\useimageresource
\lastsavedboxresourceindex
\lastsavedimageresourceindex
\lastsavedimageresourcepages
Direction nodes are now core nodes and we only have a few directions left:

TLT
TRT
LTL
RTT
Some pdfTeX commands have been promoted:

\pagewidth \pageheight
\adjustspacing \protrudechars
\ignoreligaturesinfont
\expandglyphsinfont \copyfont
\normaldeviate \uniformdeviate
\setrandomseed \randomseed
\ifabsnum \ifabsdim \ifprimitive
\primitive \savepos \lastxpos
\lastypos \outputmode
\draftmode \pxdimen \inserttht
As a side effect of better separation the backend commands are collapsed:

\texttt{pdfextension} keyword

[some follow up]

\texttt{pdfvariable} keyword

\texttt{pdffeedback} keyword
Redundant or experimental commands and functionality has been removed.
Internally rule nodes are used for rules, reusable content, images.

In addition we have empty rules: `\nolerule` and `\nohrule`.

At the Lua end we can have user rules that trigger a callback in the backend.
We added a few new low level helpers:

\nospaces for preventing spaces
(1 = no spaces, 2 = zero skips)

\[e\]toksapp \[e\]tokspre
for extending token registers

\letcharcode just to be nice

\begincsname
\lastnamedcs for less code

\hpack \vpack \tpack
to avoid callbacks
There are a few error suppressors:

\suppresslongerror
\suppressmathparerror
\suppressoutererror
\suppressifcsnameerror
\textbackslash{}Uleft \textbackslash{}Uright \textbackslash{}Umiddle

[height <dimension>]
[depth <dimension>]
[axis]
<delimiter>
\uvextensible

[height \textit{<dimension>}]  
[depth \textit{<dimension>}]  
[\textit{axis}]  
[\textit{exact}]  
\textit{<delimiter>}

\texttt{luatex}  
0.95 - 2016
\texttt{\textbackslash Uhextensible}
\texttt{\textbackslash U[over|under]delimiter}
\texttt{\textbackslash Udelimiter[over|under]}

[height \texttt{<dimension>}]  
[depth \texttt{<dimension>}]  
[left|middle|right]  
\texttt{<family>}
\texttt{<slot>}

\texttt{luatex}
\texttt{0.95 - 2016}
\textbackslash Umathaccent
[top|bottom|overlay]
[fixed]
[fraction <number>]
<delimiter>
{content}
\Uskewed / [exact|noaxis]
{2} }

\Uskewedwithdelims / () [exact|noaxis]
{2} }
A \texttt{matheqnogapstep} factor that determines the gap between formula and equation number.

A \texttt{mathdisplayskipmode} directive that controls display skips: 1 = always, 2 = only when not zero, 3 = never.
a \textsc{mathscriptsmode} parameter that determines how to apply

\texttt{\textbackslash Umathsubshiftdown (d)}
\texttt{\textbackslash Umathsupshiftup (u)}
\texttt{\textbackslash Umathsubsupshiftdown (s)}

<table>
<thead>
<tr>
<th></th>
<th>dynamic</th>
<th>dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>dynamic</td>
<td>dynamic</td>
</tr>
<tr>
<td>1</td>
<td>d</td>
<td>u</td>
</tr>
<tr>
<td>2</td>
<td>s</td>
<td>u</td>
</tr>
<tr>
<td>3</td>
<td>s</td>
<td>u + s - d</td>
</tr>
<tr>
<td>4</td>
<td>d + (s - d)/2</td>
<td>u + (s - d)/2</td>
</tr>
<tr>
<td>5</td>
<td>d</td>
<td>u + s - d</td>
</tr>
</tbody>
</table>