10 years of \TeX{} Live in Debian
Norbert Preining

Abstract
\TeX{} Live has turned into the most widely used \TeX{} distribution since support ended for \teTeX{}. Debian has carried a packaged version of \TeX{} Live for 10 years now. We review the history of \TeX{} packages in Debian, and in particular the history of \TeX{} Live packaging.

1 Introduction
Getting older, people usually start looking back at things that happened in the past, and I am no different. So I recently realized that this year (2016) there are several anniversaries of my involvement in the \TeX{} world: 14 years ago I started building binaries for \TeX{} Live, 11 years ago I proposed packaging \TeX{} Live for Debian, 10 years ago the \TeX{} Live packages entered Debian. There are other things to celebrate next year (2017), namely the 10 year anniversary of the (no longer new) infrastructure (esp. \texttt{tlmgr}) of \TeX{} Live packaging, but this will come later. In this article I want to concentrate on my involvement with \TeX{} Live in Debian.

2 Debian releases and \TeX{} systems
The \TeX{} system of choice in Debian was for many years \teTeX{} [8], curated by Thomas Esser. Digging through the Debian archive and combining this with ChangeLog entries as well as personal experiences since I joined Debian, the timeline of \TeX{} in Debian to the best of my knowledge can be found in Table 1.

The history of \TeX{} in Debian is thus split more or less into 10 years of \teTeX{}, and 10 years of \TeX{} Live. While I cannot check back to the ultimate origin, my guess is that already in the very first Debian releases (\teTeX{}) was included. The first release I can confirm (via the Debian archive) shipping \teTeX{} is the release Bo (June 1997). Maintainership during the first 10 years showed some fluctuation: The first years/releases, till about 2002, were dominated by Christoph Martin with Adrian Bunk and few others, who did most of the packaging work on \teTeX{} version 1. After this Atsuhito Kohda with help from Hilmar Preusse and others brought \teTeX{} up to version 2, and from 2004 to 2007 Frank Küster, again with the help of Hilmar Preusse and others, took over most of the work on \teTeX{}. Other names commonly appearing throughout the ChangeLog are Julian Gilbey, Ralf Stubner, LaMont Jones, and C.M. Connelly — and there were many more bug reporters and fixers.

Looking at table I have to mention the incredible amount of work that both Atsuhito Kohda and Frank Küster have put into the \teTeX{} packages, and many of their contributions have been carried over into the \TeX{} Live packages. While there weren’t many releases during their maintainership, their work has inspired and supported the packaging of \TeX{} Live to a huge extent.

3 Start of \TeX{} Live
I got involved in \TeX{} Live back in 2002 when I started building binaries for the alpha-linux architecture. I can’t remember when I first had the idea to package \TeX{} Live for Debian, but here is a timeline from my first email to the Debian Developers mailing list concerning \TeX{} Live to the first accepted upload:

2005-01-11: \textit{binaries for different architectures in debian packages} [1]
This is my first email to the Debian community about packaging \TeX{} Live. It is easy to see that I didn’t have much of a clue about Debian packaging at that time, as I proposed to simply reuse the binaries that are included in \TeX{} Live, instead of properly building them for Debian.

After the initial round of feedback (and flames) I proposed a new layout with adaptations, but still continued to try to avoid rebuilding the binaries.

As we were planning to have two distinct (and overlapping) \TeX{} systems in Debian, together with Frank Küster we proposed a package \texttt{tex-base}, later to be named \texttt{tex-common}, as basis for both the \teTeX{} and \TeX{} Live packages, providing common basic infrastructure.

The first official step in packaging a new ‘program’ for Debian is the ITP bug — \textit{Intend to package}.

2005-09-17: Re: Take over of texinfo/info packages [5]
In the course of preparing \TeX{} Live package I needed to put my hands on several other \TeX{}-related packages, the first being \texttt{texinfo}, which was orphaned (without a Debian maintainer) at that time. It was also based on this package that I became a Debian Developer.

When a new package is the first time uploaded to Debian, it cannot enter immediately but has to go through a severe scrutiny by the so-called ‘ftp-masters’. They check for license compliance, Debian
<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Name</th>
<th>t(e)(T)(X)/(t)(e)(X) Live</th>
<th>Maintainers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1993–96</td>
<td>&lt;1</td>
<td>?</td>
<td>?</td>
<td>Christoph Martin</td>
</tr>
<tr>
<td>6/1996</td>
<td>1.1</td>
<td>Buzz</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>12/1996</td>
<td>1.2</td>
<td>Rec</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>6/1997</td>
<td>1.3</td>
<td>Bo</td>
<td>(t)(e)(T)(X) 0.4</td>
<td>?</td>
</tr>
<tr>
<td>7/1998</td>
<td>2.0</td>
<td>Ham</td>
<td>(t)(e)(T)(X) 0.9</td>
<td>?</td>
</tr>
<tr>
<td>3/1999</td>
<td>2.1</td>
<td>Slink</td>
<td>(t)(e)(T)(X) 0.9.9N</td>
<td>?</td>
</tr>
<tr>
<td>8/2000</td>
<td>2.2</td>
<td>Potato</td>
<td>(t)(e)(T)(X) 1.0</td>
<td>?</td>
</tr>
<tr>
<td>7/2002</td>
<td>3.0</td>
<td>Woody</td>
<td>(t)(e)(T)(X) 1.0</td>
<td>?</td>
</tr>
<tr>
<td>6/2005</td>
<td>3.1</td>
<td>Sarge</td>
<td>(t)(e)(T)(X) 2.0</td>
<td>Atsuhito Kohda</td>
</tr>
<tr>
<td>4/2007</td>
<td>4.0</td>
<td>Etch</td>
<td>(t)(e)(T)(X) 3.0,</td>
<td>Frank Küster</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(t)(e)(T)(X) Live 2005 NP</td>
<td></td>
</tr>
<tr>
<td>2/2009</td>
<td>5.0</td>
<td>Lenny</td>
<td>(t)(e)(X) Live 2007</td>
<td>NP</td>
</tr>
<tr>
<td>2/2011</td>
<td>6.0</td>
<td>Squeeze</td>
<td>(t)(e)(X) Live 2009</td>
<td></td>
</tr>
<tr>
<td>5/2013</td>
<td>7.0</td>
<td>Wheezy</td>
<td>(t)(e)(X) Live 2012</td>
<td></td>
</tr>
<tr>
<td>4/2015</td>
<td>8.0</td>
<td>Jessie</td>
<td>(t)(e)(X) Live 2014</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Stretch</td>
<td>(t)(e)(X) Live ≥2015</td>
</tr>
</tbody>
</table>

Table 1: History of \(t\)\(e\)\(X\) systems in Debian

policy compliance, and some say their daily level of comfort, before allowing a new package to enter Debian. After my first upload I got extremely negative feedback, including statements like ‘Why do we need another \(t\)\(e\)\(X\) system.’ Together with Frank Küster we drafted a response, which sparked a long discussion about packaging and helped improve the naming of packages (but not especially the packaging itself).

2006-01-12: Upload of \(t\)\(e\)\(X\) Live 2005-1 to Debian
The first upload that successfully passed the scrutiny of the ftp-masters.

2006-01-22: Accepted texlive-base 2005-1 (source all) [7]
\(t\)\(e\)\(X\) Live packages accepted to Debian/experimental.

One can see from the first emails that at that time I had no idea of correct Debian packaging and proposed to ship the binaries built within the \(t\)\(e\)\(X\) Live system on Debian. What followed was first a long discussion about whether there is any need for “just another” \(t\)\(e\)\(X\) system. The then maintainer Frank Küster took a clear stance in favor of including \(t\)\(e\)\(X\) Live, and after several rounds of proposals, tests, rejections and improvements, the first successful upload of \(t\)\(e\)\(X\) Live packages to Debian/experimental happened on 12 January 2006, so exactly 10 years ago.

4 Packaging

From the beginning, Debian has used a meta-packaging approach. That is, instead of working directly with the \(t\)\(e\)\(X\) Live sources, (Perl) scripts generate Debian source packages from a set of directives. We introduced this extra layer for several reasons:

- The original format of the \(t\)\(e\)\(X\) Live packaging information (\(t\)\(p\)\(m\)) was XML files that Debian parsed with an XML parser (\texttt{libxml}). I surmise (from what I have seen over the years) that only the Debian packages did proper parsing of these \(t\)\(p\)\(m\) files for packaging.
- \(t\)\(e\)\(X\) Live packages were often reshuffled, and Debian package names changed, which would have otherwise caused a certain level of pain during the creation of original tar files and packaging.
- General flexibility in creating additional packages and arbitrary dependencies.

Although I have never been 100% sure that it was the best idea, the scripts nevertheless remain in place to the present day, only adapted to the new packaging paradigm in \(t\)\(e\)\(X\) Live (without XML) and adding new functionality. This allows me to just kick off one script that does all the work, including building .\texttt{orig.tar.gz}, source packages, and binary packages.

For those interested in following the frantic activity during the first few years, there is a file \texttt{CHANGES.packaging} [9] which extensively documents the changes made for the years from 2005 to 2011. I don’t want to count the hours that went into this.

5 Development over the years

\(t\)\(e\)\(X\) Live 2005 was just another \(t\)\(e\)\(X\) system but not the preferred one in Debian Etch and earlier. But

Norbert Preining
in May 2006, Thomas Esser announced the end of
development for te\TeX{}, which cleared the path for\TeX{} Live as the main \TeX{} system in Debian (and the
world!). The next release of Debian, Lenny (1/2009),
already carried only \TeX{} Live. Unfortunately it was
only \TeX{} Live 2007 and not 2008, mostly due to
my having been involved in rewriting the upstream
infrastructure based on plain text package descrip-
tions instead of the notorious XML files. This took
quite a lot of attention and time from Debian away
to upstream development, but this will be discussed
in a different post.

Similarly, the release of \TeX{} Live included in
Debian Squeeze (released 2/2011) was only \TeX{} Live 2009
(instead of 2010), but in the releases since then (Wheezy and Jessie), the versions of \TeX{} Live in
Debian have been the latest releases.

6 Current status
Since about 2013 I am trying to keep a regular sched-
ule of new \TeX{} Live packages every month. These
helps me to keep up with the changes in upstream
packaging and reduces the load of packaging a new
release of \TeX{} Live. It also brings to users of unsta-
able and testing a very up-to-date \TeX{} system, where
packages at most lag one month behind the \TeX{} Live
network updates.

7 Future
As most of the readers here know, besides caring for
\TeX{} (Live) and related packages in Debian, I am also
responsible for the \TeX{} Live Manager (\texttt{tlmgr}) and
most of upstream’s infrastructure including network
distribution. Thus, my (spare, outside work) time
needs to be distributed between all these projects
(among others) which leaves less and less time for
Debian packaging. Fortunately the packaging is in
a state that makes regular updates once a month a
light enough burden to accomplish, since most steps
are automated. What remains a bit of a struggle is
adapting the binary package (\texttt{src:texlive-bin [10]})
to new releases. But also this has become simpler
due to less invasive changes over the years.

All in all, I don’t have many plans for \TeX{} Live in
Debian besides keeping the current system running
as it is. And this is in itself already a good reason
to search for new contributors and maintainers!

8 Search for and advice to future
maintainers and collaborators
I would be more than happy if new collaborators
appear, with fresh ideas and some spare time. Un-
fortunately, my experience over these 10 years with
people showing up and proposing changes (anyone
remember the fellow proposing a complete rewrite in
ML?) has been that nobody wants to invest serious
time and energy, but merely searches for quick solu-
tions. This is not something that will work with a
package like \TeX{} Live, with a size of several gigabytes
(the biggest in the Debian archive), and complicated
inner workings.

I advise everyone interested in helping to pack-
age \TeX{} Live for Debian (or for that matter any other
operating system distribution), to first install normal
\TeX{} Live from TUG, get used to what actions hap-
pen during updates (format rebuilds, hyphenation
patterns, map file updates). One does not need to
have a perfect understanding of what exactly hap-
pens down there in the guts (I didn’t have in the
beginning, either), but if you want to help with pack-
aging but have never heard of format dumps or map
files, this just might be a small obstacle.

9 Conclusion
\TeX{} Live is the only \TeX{} system in wide use across
many hardware architectures and operating systems.
The only comparable system, MiK\TeX{}, is Windows-
specific (although it contains some traces of ports to
Unix). Backed by all the big user groups of \TeX{}, \TeX{}
Live will remain the prime choice for the foreseeable
future, and thus also \TeX{} Live in Debian.

References
[5] https://goo.gl/Hs40KJ.
[8] The \TeX{} Live home page.
http://tug.org/texlive/.
[10] texlive-bin source package on Debian QA.
https://goo.gl/MGmRd3.

Norbert Preining
Japan Advanced Institute of
Science and Technology
Nomi, Ishikawa, Japan
norbert (at) preining dot info
http://www.preining.info