Creating a VPAT statement for TEX Live

Boris Veytsman, Keiran Harcombe

Abstract

Governments around the world are enforcing accessibility standards. Vendors of software used by government agencies are required to file formal statements of accessibility for their products. This presents a special challenge for open source products, if they are not sponsored by a corporation.

In this paper we discuss our experience in creation of such a formal statement for TEX Live. While command-line tools are usually more accessible than GUI interfaces, the work turned out to be more difficult than we thought in the beginning.

1 Introduction

If we want to prove that the current scientific and technical progress is accompanied by moral progress, then the changing consensus about the people with disabilities may provide a solid argument. We now more and more think that the full participation of disadvantaged people in the life of society is not a generous gift from the society to them, but rather their right. Governments now often act to protect this right. In particular, according to current regulations, software vendors must make reasonable efforts to make their products accessible, and to formally report these efforts. In many countries, including the US and EU, these reports are becoming a condition for the software to be used by government agencies and contractors.

This presents a challenge for free software. Commercial vendors have resources to hire lawyers and experts to create formal compliance reports. Free software like TEX is often created by volunteers across the world. Volunteers are motivated to do "interesting" things rather than wade through pages of regulations and templates. On the other hand, the absence of compliance reports can undermine the usage of free software by the public sector and anybody dealing with the government agencies — which, in modern society, means everybody. Organizations such as TUG might be useful in this situation, since "uninteresting" work necessary for free software to thrive is clearly in their remit.

In this paper we discuss a pilot project in this vein: a VPAT statement for TEX Live.

2 Rules, regulations, templates

A template for a compliance report was developed by the Information Technology Industry Council (ITI).[1] It is available as the Voluntary Product Accessibility Template (VPAT)[2] and is free to use. The template is based on several sources:

1. Web Content Accessibility Guidelines developed by the W3 Consortium (WCAG).[3]
2. Revised Section 508 Standards by the US government.[4]
3. EN 301 549 Accessibility requirements suitable for public procurement of ICT products and services in Europe.[5]

WCAG is not, strictly speaking, a government standard. However, both US and European standards require compliance with WCAG, or, in lawyers' lingo, incorporate it.

ITI's VPAT template has several variants, or "editions". Since TEX is used throughout the world, we chose the largest one, the International edition, that reports compliance with both US and EU standards.

The standards consider several categories of software, as follows:

Web documents are documents (in any format) published on, as you might surmise, the World Wide Web. This is what WCAG specifies.

Electronic documents are documents published in any other manner. Even if a product is not itself a document, the government standards say, in essence, that its documentation must be accessible and thus comply with WCAG standards, even when it is not published on the web.

Software is a generic term for software.

Closed system does not mean what we in the Free software community mean by these words: the standards define closed systems as those which do not allow an easy interaction with assistive technologies.

Authoring tool is software for the creation of documents.

The compliance report may discuss compliance with any or all of these categories, with one exception: documentation is usually a part of any category.

3 Our decisions and the lessons learned

Initially we thought that our task would be easy. TEX and friends are command line tools. We know

[1] https://www.itic.org/

[2] https://www.itic.org/policy/accessibility/vpat

[3] http://www.w3.org/TR/2008/REC-WCAG20-20081211 and https://www.w3.org/TR/WCAG21

[4] https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/text-of-the-standards-and-guidelines

[5] https://www.etsi.org/deliver/etsi_en/301500_301599/301549/03.01.01_60/en_301549v030101p.pdf

doi.org/10.47397/tb/42-2/tb131veytsman-vpat

that command line tools are very accessible: people with low/no hearing or low/no vision can use command line tools with the existing assistive technology. However, when we got better acquainted with the standards and the requirements, we understood this was not as easy as seemed.

First, the compliance report cannot be generic. We cannot say that any implementation of TeX and its ecosystem is accessible. Therefore we decided to choose at first a concrete implementation, which could be installed and operated with assistive technologies. We chose TeX Live 2021: our flagship implementation of TeX and friends. We plan to add other implementations to the list in the future.

Second, we must report whether the documentation for TeX is compliant. This immediately causes a question, what is TeX documentation? At the time we counted, TeX Live 2021 had 36 769 files in the `doc` tree in various formats: mostly text, PostScript, PDF and HTML. Obviously many of them are not accessible. Fortunately, it is not needed to read them *all* to successfully use TeX Live. Also fortunately, in practice the situation is ameliorated by the long-standing policy of TeX Live to require source code for all its documentation: text source code can be read by voice software.

We decided to discuss only one piece of documentation: the TeX Live manual in HTML format at `https://tug.org/texlive/doc/texlive-en`. The WCAG guidelines define three levels of compliance, denoted as A, AA, and AAA. Only the first two are required by the US and EU standards. The third one contains rather difficult requirements such as automatic explanation of abbreviations and special terminology.

However, we found problems even at the first two levels. Namely, the guidelines require alternative text for all images. The TeX Live manual has images for GUI installation mode. One can argue that these images are not especially relevant for low vision users, who are probably going to use text mode installation anyway. However, this is a weak argument, and we need to make the manual fully accessible.

Further, the category *closed systems* is evidently not applicable for TeX Live: it is open not only as open source software, but also as software which allows the user to integrate it with the assistive technologies.

We hit a snag, however, when looking into the requirements for *authoring tools*. Simply saying, these requirements say that the software is not just accessible by itself, but also the documents it creates are accessible. Now the TeX community and TUG have and continue to put considerable efforts into creating the tools for accessible output, notably tagged PDF. However, at present the creation of accessible documents with TeX, while possible, still requires significant effort. Thus we made a decision to not yet make a statement about TeX Live as an authoring tool. That is, we state that, for example, a blind person can efficiently learn and use TeX to create a document. However we do *not* state that this document will be accessible even to its author. This is a sad situation, but hopefully it will be ameliorated soon.

The resulting draft statement is available at `https://github.com/TeXUsersGroup/TeX-VPAT`.

4 Next steps

Our work showed that there is much to do with TeX compliance documentation.

First, we must make the TeX Live manual fully compliant by adding alt text to all images.

Second, we need to make the statement itself fully accessible, either in PDF or HTML format, or both.

Third, we need to add separate statements for other distributions: MacTeX, MikTeX, proTeXt... We would like to invite volunteers best acquainted with these distributions to help with them; we hope it might be easy enough to follow our example.

Fourth, it is important to add the statement about TeX as an authoring tool. Perhaps we may put in statements about ConTeXt, which currently has good capabilities for creation of tagged PDF, or the usage of TeX4ht for creation of accessible HTML files. This requires further research and decisions.

The task of making all TeX Live documentation accessible seems to be very difficult, but we might start with the guidelines for package authors.

Last but not least, we plan to release the statement as a CTAN package.

Making TeX accessible is an ongoing process. We are glad to be a part of it.

◇ Boris Veytsman
George Mason University (US),
TeX Users Group
borisv (at) lk (dot) net
http://borisv.lk.net

◇ Keiran Harcombe
Open University (UK)
kjh (at) harcombe (dot) net
https://harcombe.net/~keiran/contact