Accessible EPUB : Making EPUB 3 documents universal accessible

Thorsten Schwarz, Sachin Rajgopal, and Rainer Stiefelhagen

Karlsruhe Institute of Technology, Studycentre for the Visually Impaired,
Engesserstr. 4, 76131 Karlsruhe, Germany
thorsten.schwarz@kit.edu,
WWW home page: http://szs.kit.edu

Abstract. Current document standards have the characteristic that they can only serve one group of users under the aspect of vision (sighted, visually impaired or blind reader). Wouldn't it be nice if one document-format could combine all properties in one document? This paper presents a new approach of an universal accessible version of EPUB 3 documents, which will allow sighted, visually impaired and blind readers to use and share the same EPUB 3 document by an "integrated switching mechanism" to change the output format. Furthermore, we will introduce a simple word processing tool that allows to easily create such accessible EPUBs without knowing how an EPUB file is constructed or created.

Keywords: epub, epub3, accessible documents

1   Introduction

In recent years, accessibility has become increasingly important, especially for accessible documents. Several nations have passed regulatory laws that ensure equal treatment of all people and ensure that documents are accessible to all [1]. The currently dominant formats for accessible electronic documents are Microsoft Word and PDF (Portable Document Format) documents, or more precisely PDF/UA (PDF/Universal Accessibility) documents. First of all, both formats have a predefined page size. While this is useful for printed documents, a computer screen can rarely display all contents of the document to the detriment of visually impaired people [2]. Therefore, an electronic document format without a fixed document size containing semantic and structural information and a fixed reading order would be better suited to meet the requirements of accessibility.

Furthermore, different "selectable" forms of presentation would be advantageous, especially for graphics or mathematical formulas. For example, formulas in the LaTeX source code for blind users or high-contrast images for users with limited residual vision. This could be combined with EPUB 3 [2].

This is exactly where we come in with our new EPUB approach, which combines all three versions in one document. But why is EPUB the format of our choice?

EPUB stands for **E**lectronic **PUB**lication and is a format primarily used for books in an electronic format (E-book). The EPUB format was created by the International Digital Publishing Forum (IDPF) and the current version is 3.1 which is a minor update to EPUB 3 [3]. EPUB uses XML based formats like XHTML, and thus also uses the accessibility standards and guidelines already established in many nations like the Web Content Accessibility Guidelines (WCAG)[4]. This was done as reading systems can have different screen sizes and the EPUB content can therefore be reflowable. Font type and size can also be adapted to the individual needs of the users. Visually impaired people could therefore adjust the document to their preferences in font style, size and color. The EPUB 3 specification also contains guidelines for accessibility so these features are built in and not an afterthought [5].

The EPUB working group has also made important changes from EPUB 2 to EPUB 3 which improve the accessibility of documents. For example, mathematical equations can now be displayed in MathML and there is better navigation and more support for Cascading Style Sheets (CSS). However, not all of these changes are yet supported by EPUB readers and devices [6].

Creating EPUBs, however, is not as simple as word documents. There are several editors available, but the issues of them will be presented in chapter 2.1. Accessible EPUB is an attempt to remedy these issues and simplify the process so that accessible documents for STEM subjects can be created without intricate knowledge of EPUBs.

2 Related Work

Of course, there are already works that have tried to generate different accessible formats. Here are a few examples: Takaira et al. [7] created a tool which analyzes a PDF document and outputs different kinds of textbooks by detecting structure of the document properly assigning the tags. Leporini et al. [8] have created the tool "Book4All" to convert PDF-E books into a more accessible format. This was done by analyzing the PDF and converting it into the XML based Intermediate Book Format. This can then be converted into XHTML or Daisy [9]. Coming back to EPUB, Book4All was also developed to create EPUBs and Lenzi and Leporini [10] found out that while there are difficulties using voiceover with EPUBs, this can be improved by using the tags and attributes properly. Bartalesi and Leporini [11] carried out an online survey in their work and asked 25 users to rate "enriched" EPUB in comparison to the original PDF format regarding accessibility and usability. 50% of respondents preferred EPUB over other e-book formats, while 13% said EPUB was equivalent. The sample group also felt that it was easier to access content in EPUBs and use the table of contents than in PDFs. Furthermore, 80% of blind users were unable to read images in PDFs correctly with their screen reader, while the corresponding value for EPUBs was less than 50%. 64% of users found the EPUB's document structure easy to understand.

2.1 Existing EPUB editors

There are now a number of different ways to create EPUB documents [12]. Adobe InDesign[1] is one way and is suitable for publishers, but has no built-in MathML support and is a commercial program that is unaffordable to most user groups. LibreOffice in its latest version 6.0 is able to export Writer documents to EPUB, too. This and perhaps Word (with installed EPUB macros) are maybe the easiest options to create an EPUB document. Nevertheless, these versions have to be edited in special EPUB editors to meet proper accessiblity levels. For example, alternative texts for images or LaTeX source code alternatives for mathematical formulas.

Some EPUB editors are pure WYSIWYG editors (What You See Is What You Get), but important functions such as mathematical equations and semantic information are usually missing. EPUB editors where the program code has to be edited are limited to persons with programming experience. Sigil [13], for example, is such a popular open source WYSIWYG EPUB editor with many features, but important features such as text alternatives for images can only be added manually with coding.

After all, most editors are not able to produce the kind of universal accessible document we want out of the box, nor accessible to visually impaired or blind people.

3 Specifications of our "Accessible EPUB"

3.1 EPUB Creation Process

First of all, what is needed to create an EPUB file and then get it accessible? An EPUB file is nothing more than a ZIP file[2] [14], but renamed from "*.zip" to "*.epub". If you change the file extension and extract the file contents, the individual files, such as XHTML and image files, can be opened and edited. The creation of an EPUB document can therefore be briefly described in three steps:

1. create content documents (texts, images, etc.) in formats such as XHTML and SVG,
2. create the package document "package.opf" and
3. compression of the data including metadata in a ZIP file.

The package document consists of five sections describing how the EPUB document is structured [5].

3.2 EPUB switching mechanism

The introduction shows that although EPUB 3 is a promising format, the lack of easy-to-use editors means that not everyone can create accessible EPUB documents. Therefore, the main objective of this work is to lay the foundations for
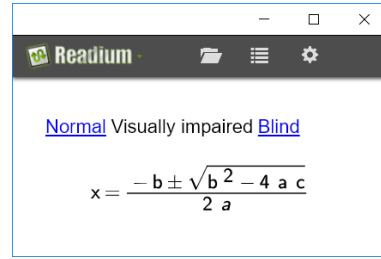

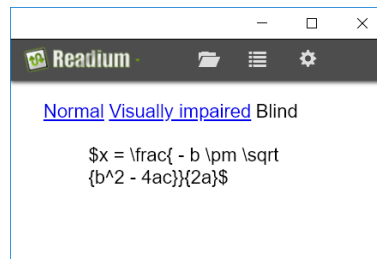
[1] https://www.adobe.com/products/indesign.html
[2] ZIP is an archive file format that supports lossless data compression.
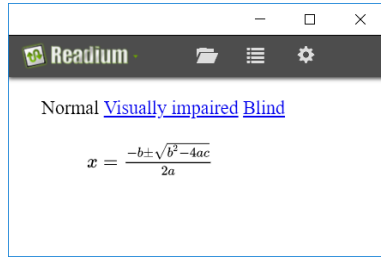
Fig. 1. EPUB in 'Normal' mode,

Fig. 2. in 'Visual impairment' mode,

Fig. 3. and in 'Blind' mode.

such EPUB documents. On the one hand, there is a suitable basic structure that unifies all format versions, and on the other hand, a suitable editor that gets the mentioned problems/restrictions under control for the end user or document creator.

This means that, at the end of the day, it should be easy for teachers and other educators to create an EPUB document that can be used by blind, visually impaired and sighted people. By a simple switch in the document, the CSS can be adapted to the needs of the target group with one click (see figures 1 to 3).

For example, the text alternatives for mathematical equations (e. g. the LaTeX source code) and images are displayed automatically when the blind version is selected [5]. The switching mechanism can be done either with JavaScript or via CSS. Although JavaScript is easier to use, not all EPUB readers or devices support JavaScript and the EPUB specification does not require JavaScript to be supported [5]. The CSS version is much more limited because all content has to be inserted into a single XHTML file and CSS selectors are not fully supported yet, but CSS support is included in the EPUB specification. Ultimately, we need an EPUB framework that can be used by most end devices, as well as a suitable easy-to-use editor.

Unfortunately a unified accessible standard for the EPUB documents could not be found, as EPUB readers do not support all features mentioned in the EPUB 3 specifications. As a result, Accessible EPUB allows users to create documents either with CSS or JavaScript as basis for the switching mechanism.

It is possible that when EPUB readers either have widespread support for CSS 3 or the specification makes JavaScript support mandatory, one of the standards will establish itself. Until then it is better to design both to allow users to have a choice.

3.3 Editor: "Accessible EPUB"

"Accessible EPUB" is a WYSIWYG program written in C# (. NET) that serves to create accessible EPUB documents using the switching mechanism (see figure 4). The process of creating documents should be as simple as possible so that even people with little experience in using special software can use the program. Primary target groups are, for example, teachers or implementation centers that have a need for accessible documents for different user groups.
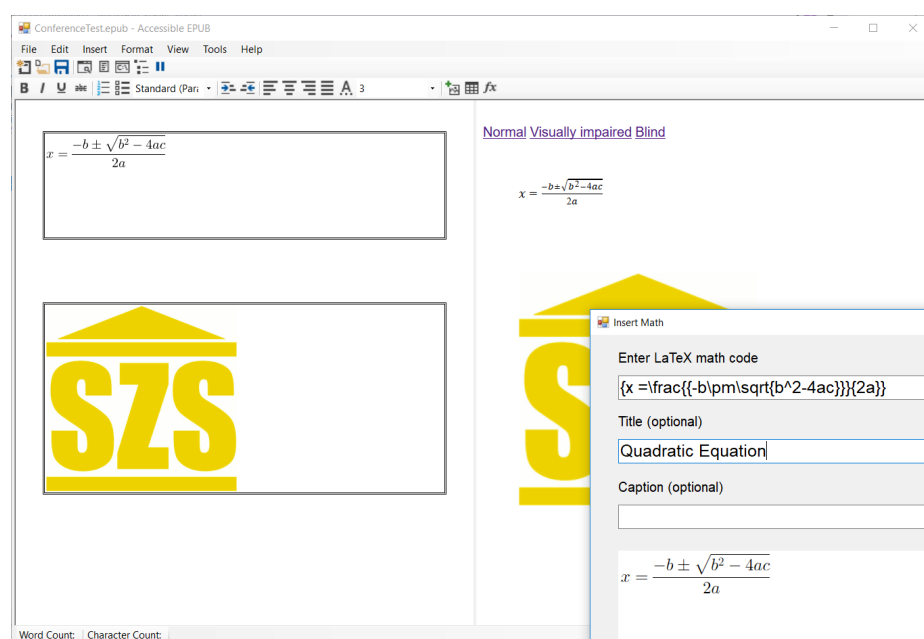
Fig. 4. Accessible EPUB equation editor

A feature is to allow the user to have immediate feedback to their actions. Any text or other content written in the editor of left hand side is shown on the preview browser on the right hand side, shown in figure 4. The user can switch between normal, visually impaired and blind versions of the content on the preview browser. The user does not need to know how the switching is done; the program does this every few seconds automatically.

One of the goals of this project is to create documents for STEM subjects. In other EPUB editors, like Sigil or Adobe InDesign, there is no possibility of

adding MathML without coding it in. Equations have to be inserted as normal text or in image form. As discussed earlier in this paper, MathML is preferable due to is superior presentation and the ability to interact with equations, such as highlighting a part of an equation. Therefore adding a easy way to insert equations was of particular importance in this program. The equation insertion tool can be seen in figure 4. It accepts LaTeX code and the result is shown in a panel in the bottom half of the screen. Errors mean that the panel is highlighted with a red border. This immediate feedback should improve the usability of the program.

Tables and images are inserted similarly. When inserting images, the user has to enter some alternative text to ensure the accessibility of the document. Entering a caption and title is optional. All of these will be put in a figure element, and if the user moves it, the caption and alternative text will move it. Editing commands commonly found in word processors are also present in a toolbar at the top of the program.

4   Evaluation of Document Standard

The document standard was tested on various programs to see if the EPUB can be read properly. The programs tested include Adobe Digital Editions[3], Calibre[4], AZARDI[5] and Readium[6] on Windows 10 and iBooks[7] on a iPad. The test examined if they supported the following: the CSS standard, the JavaScript standard, MathML and screen reader. The results are shown in figure 5. Several other software readers were tested, like the Bluefire Reader[8] and the Icecream Ebook Reader[9], but only the one listed below supported MathML. Adobe Digital Editions is an exception, but is among the better known readers so it was included in the comparison.

	CSS	JavaScript	MathML	Screen reader
Adobe Digital Editions	✗	✗	✗	✓
Calibre	✗	✓	✓	✗
AZARDI	✗	✗	✓	✓
Readium	✓	✗	✓	✓
iBooks	✗	✗	✓	✓

Fig. 5. Features supported by several EPUB readers

---

[3] https://www.adobe.com/solutions/ebook/digital-editions.html

[4] https://calibre-ebook.com/

[5] http://azardi.infogridpacific.com/

[6] http://readium.org/

[7] https://www.apple.com/ibooks/

[8] http://www.bluefirereader.com/

[9] https://icecreamapps.com/Ebook-Reader/

The first problem was the lack of MathML support many programs had. EPUB 3 was approved by the IDPF in 2011, but MathML support is still not common. Even major companies like Adobe do not have MathML support in Adobe Digital Editions. Only the open source program Calibre supports the JavaScript standard, which of course not mandatory to support. It supports session storage, so until the program is closed the EPUB will retain its appearance. Session and local storage as used in web browsers to store settings for websites. Calibre, however, does not have any support for screen readers. iBooks does support JavaScript, but does not allow access to local or session storage across the whole file. While the initial page changes its style, other pages remain in their default state. CSS 3 support in the form of CSS selectors is also very rare. Only Readium supports it and is therefore the only reader capable of displaying the CSS standard. However, sometimes Readium has issues with selectors which the Gecko based Mozilla Firefox does not have, so the support can still be improved. Readium also supports MathML and screen readers so it is the only option for blind users to read documents of this standard. AZARDI is mentioned by the DAISY Consortium as a EPUB reader with screen reader support.

The document standard created by us uses new EPUB 3 features extensively, and this is where the problem lies. EPUB 3 support is slowly improving, but is still not widespread seven years after it was approved. The site, EPUB 3 Support Grid, [10] gives a comprehensive overview of what feature is supported by what reading system.

5 Future Work

In its current form Accessible EPUB allows users to create functional accessible EPUB documents which can be read with either Calibre or Readium. Future development will focus on usability and easing the transition from other programs. Users are already able to import text and HTML documents allowing them to keep their work in this formats. There are still issues with importing HTML documents though. Images and formulas will not be in the format specified by the document standard, so a wizard which works its way through each image and formula and allowing the user to enter alternative text will be added.

One essential feature being working on is the ability to edit equations and images after inserting them. Currently, users would have to remove the figure containing them and insert them again. A way has not been found to access the individual HTML elements in the editor. This is an extremely important feature which is present in word processors, which the users will expect Accessible EPUB to have.

A feature which advanced users might want is a source code editor. A code viewer is already present, but it presently shows the whole file with the inner workings. The user should not be able to see it as editing it increases the possibilities of errors in the document.

---

[10] http://epubtest.org/testsuite/epub3/

The JavaScript document standard supports multiple HTML files as content documents. If the reading system supports local or session storage it remembers which version should be displayed across several HTML files. Accessible EPUB does not support this yet. Adding this feature would allow users to create longer documents and allow them to manage individual sections.

6 Conclusion

A EPUB document standard was developed to enable visually impaired, blind and normal sighted users to use the same document, while satisfying their respective accessibility needs. A editor named "Accessible EPUB" was created so that users with no programming knowledge can create accessible documents. The document standard uses EPUB 3, which is not supported fully by most EPUB reading systems. It is planned to add features which will improve the usability of Accessible EPUB.

References

[1] WebAIM (Web Accessibility in Mind): World Laws: Introduction to Laws Throughout the World, https://webaim.org/articles/laws/world/, visited on 30.01.2018

[2] IDPF: Understanding EPUB 3, http://epubzone.org/epub-3-overview/understanding-epub-3, visited on 29.01.2018

[3] IDPF: EPUB 3 Specification, http://www.idpf.org/epub/301/spec/epub-overview.html, visited on 29.01.2018

[4] W3C: Web Content Accessibility Guidelines (WCAG) 2.0, https://www.w3.org/TR/WCAG20/, visited on 29.01.2018

[5] Garrish, M.; Gylling, M.: EPUB 3 Best Practices, 2013, O'Reilly, ISBN 1449329144, 9781449329143

[6] IDPF: EPUB 3 Changes from EPUB 2.0.1, http://www.idpf.org/epub/30/spec/epub30-changes.html, visited on 29.01.2018

[7] Takaira, T.; Tani, Y.; Fujiyoshi, A.: Development of a Unified Production System for Various Types of Accessible Textbooks, Computers Helping People with Special Needs, 2016, Springer International Publishing, 381-388, ISBN 978-3-319-41264-1

[8] Calabrò, A.; Contini, E.; Leporini, B.: Book4All: A Tool to Make an e-Book More Accessible to Students with Vision/Visual-Impairments, HCI and Usability for e-Inclusion, 2009, Springer Berlin Heidelberg, 236-248, ISBN 978-3-642-10308-7"

[9] DAISY Consortium: Baseline for Accessible EPUB3, http://www.daisy.org/baseline, visited on 29.01.2018

[10] Bartalesi Lenzi, V.; Leporini, B.: Investigating an Accessible and Usable ePub Book via VoiceOver: A Case Study, Human Factors in Computing and Informatics, 2013, Springer Berlin Heidelberg, 272-283, ISBN 978-3-642-39062-3"

[11] Bartalesi, V.; Leporini, B.: An Enriched ePub eBook for Screen Reader Users, Universal Access in Human-Computer Interaction. Access to Today's Technologies, 2015, Springer International Publishing, 375-386, ISBN 978-3-319-20678-3

[12] DAISY Consortium: Tools for creating EPUB 3 files, http://www.daisy.org/daisypedia/tools-creating-epub-3-files, visited on 29.01.2018

[13] Markovi, S.; Schember, J.; Hendricks, K.; Massay, D.: Sigil is a multi-platform EPUB ebook editor, https://sigil-ebook.com/, visited on 29.01.2018

[14] Garrish, M.: What is EPUB 3?, 2011, O'Reilly, ISBN 9781449314545