# The Essentials in Your Toolbox, T-122 Days – webinar transcript

Full details about this webinar including links to related resources can be found on our website:

<https://daisy.org/news-events/articles/the-essentials-in-your-toolbox-t-122/>

Welcome, everyone. My name is Richard Orme, and I'm delighted to have you join us today for another webinar hosted by the DAISY Consortium. As we count down the days until the European Accessibility Act comes into force. We're bringing you this series of webinars to help you prepare. This week, we explore some of the essential accessibility tools that everyone should consider for their toolbox. Accessible digital publishing can be a challenge trying to follow the latest standards and best practices, often while working with content within the constraints of established workflows. Prescribed source formats and authoring processes. But thankfully, a portfolio of tools are available to aid you on your accessible publishing journey, helping to perform essential tasks. Highlight issues, and providing confidence that your material will deliver the level of accessibility required. So for this session, we're joined by a wonderful panel of experts, each of whom will present a different series of tools, including unique insights and special features you may not be familiar with. Okay, let's get started. First up, we're delighted to be joined by Romain Deltour from the DAISY consortium. Over to you, Romain.

Hey, everyone. Let's talk about epubcheck. You probably know. Or at least I've heard of this tool, especially if you produce epub books on a regular basis. In fact, check is used almost universally by all publishers of distribution platforms. So if you ever consumed an e-book, it is very likely that someone at some point used EPUBCheck on that very same e-book. Quite an essential piece of your toolbox, I would say. In this presentation, I'm going to explain what EPUBCheck is and what it can do, and demo how we can use it for quick one trip editing. So what is it exactly? Well, as its tagline says, it is a conformance checker for EPUB publications. It means EPUBCheck can take an EPUB document, analyze it, and tell if these documents conforms to the EPUB format. Or in other words, what is required by the official specification. Some people call EPUBCheck an EPUB validator, although I personally prefer the term conformance checker. What EPUBCheck is not is an accessibility checker, an EPUB can very well be conforming to the EPUB specification and yet have serious accessibility shortcomings. But conformance is an important first step anyway. It is a foundational ground for a good reading experience and good interoperability across reading systems and assistive technologies. EPUBCheck is open source software available at no cost. It is maintained by the DAISY Consortium on behalf of W3C, where the EPUB specification are made. EPUBCheck is cross-platform and can be run in windows, macOS and Linux. In this presentation, we're going to use EPUBCheck as a command line tool, which is readily available in the official downloadable package, but be reassured if you prefer, EPUBCheck is also usable with graphical user interfaces or integrated in various editors or production tools. So let's dive in and see how to run our first command. The most basic command is to type EPUBCheck. That's the name of the command followed by the path to the file you want to check. Here my test file is called sabotage dot epub. So if I'm in the directory where my file is I can type EPUBCheck space sabotage dot epub It turns out I'm lucky. Or rather the EPUB producer did a good job because if EPUBCheck tells us no errors or warnings were detected. The EPUB is conforming to the specification. Yay! But let's say I want to do a quick edit to my EPUB. To edit or inspect its contents. We first need to access its internals. An input file is an archive, essentially a zip file. You can use any unzip tool to extend this archive and access the contents within. A common trick is to rename the dot EPUB extension to dot zip, and then unzip the file, as you would do with any other zip file. Once you've done that, you can access the EPUB contents as a set of files and directories. For instance, let's say we want to add or tweak the accessibility metadata for the EPUB. We can open the package document that's the file with that extension. Open it in a text editor. Edit the metadata contents from there, and then save the file. Text editors typically have very powerful search and replace capabilities or multiple selection features, and that can be great for repetitive edits or quick fixes. Or as another example, we can open one of the HTML content documents in a browser to preview them, or use the developer tools to inspect the markup, the accessibility tree, and so on. Browser developer tools are usually very advanced, and using them is a great way to inspect the EPUB markup in detail. What is sometimes less known is that EPUBCheck can also check individual documents within an expanded EPUB. Let's say we edited the package document of our EPUB publication. For example, to edit accessibility metadata. We can now ask EPUBCheck to evaluate the package document only. So a quicker and more targeted result. To do that we can invoke the EPUBCheck command with the option dash mode space opf Followed by the path to the package documents dot opf file. If I unpacked my earlier test file to a directory named sabotage, the command goes: EPUBCheck space dash mode space opf space Sabotage / oebps / content dot opf Note, however, that when checking individual files within the EPUB publication, only a subset of checks will be executed. The reason is simply that EPUBCheck needs to be aware of the entire file set to run the full evaluation. For example, you could keep a check can only assess if internal links resolves correctly if it has access to the entire file set context. Evaluating a single file will miss that kind of errors. If you want to check the entire publication, but as an expanded directory, you can use the same dash mode option, but this time with the exp keyword followed by the path to the directory with my test file, the full command becomes epubcheck space dash mode, space exp space sabotage. This is almost exactly the same checks as if you evaluated a zipped EPUB, but since you've unpackaged your EPUB, it is much easier to make edits and check again. Finally, when we are done editing or inspecting our EPUB, we can use EPUBCheck to repackage the expanded publication in an EPUB file. To do that, we need to run EPUBCheck on an expanded EPUB directory and ask to package the directory in a dot EPUB file with the dash save option. If no errors or warnings were issued, the EPUB file will be created in the same directory where the expanded directory is located. For instance, in my working directory where expanded the sabotage EPUB directory, I can run the command EPUBCheck space dash mode, space Exp space dash save space, sabotage and it will create the file sabotage dot EPUB in the working directory alongside the expanded sabotage directory. Just be careful however. Any preexisting EPUB file with the same name in that directory will be overwritten. So let's wrap up our brief introduction to back check. We've seen how to run EPUBCheck as a command line tool to check simple EPUBs. After extracting the EPUB file set to a directory, we were able to use powerful inspection tools, make quick edits, and then repackage the publication as an EPUB file. With EPUBCheck itself EPUBCheck should be your go to a utility to ensure your EPUB contents conforms to the EPUB format, it is free. It is available and integrated in various other tools and workflows. But remember, Epub check is not an accessibility checker. It is a first step only, but an essential one. Thank you.

Thank you, Romain, for that wonderful presentation, including some features of epub track that I hadn't heard of before. Next up, we're joined by Daniel Weck, who will be giving us a deeper dive into Ace.

Hello everyone. My name is Daniel Weck. I am a software engineer. I work with the DAISY consortium and I am a core contributor in the Ace project. In this short presentation, I will provide an overview of Ace by DAISY, the Accessibility Checker for EPUB, A C E. Ace is a software tool that analyzes Epub files to check for accessibility issues. Additionally, the tool extracts useful information from publications such as metadata, navigation, outlines, and image files with their descriptions. Ace is not a certification tool, but it helps evaluate conformance and facilitates content remediation. Ace has a command line interface or CLI. That is to say, an executable program that runs in a terminal. Ace is also an interactive app with a graphical user interface or GUI. The software is free and available on windows, Mac and Linux. The code is open source. It is written in JavaScript for the cross-platform Node.js runtime. Ace implements a specification named Epub Accessibility Version 1.1, which was published by the W3C in October 2024. This describes the requirements that accessible Epub publications must meet IEEE quantum conformance. This also specifies the discoverability metadata. I will come back to this in the next slide. Additionally, practical guidance for EAP creators is available in a W3C working Group note entitled Accessibility Techniques. Version 1.1, published in December 2024. Here are some of the accessibility rules that Ace checks for. Discoverability. Metadata makes qualitative statements about the publication by advertising accessibility features. Access modes, potential hazards, etc.. There is also metadata that expresses conformance claims and provides details about the accessibility, evaluation and certification. Ace checks patient navigation properties such as the print source metadata, page lists, navigation and page breaks in the actual HTML files. Ace also verifies the reading order of synchronized text audio. Crucially, a great number of rules are dictated by the Web Content Accessibility Guidelines, or WCAG. Sometimes referred to as waycag the W in WCAG stands for Web, but this also applies to Epub, which is essentially a website in a box. A publication is a single file zip archive that contains some XML data that describes the EPUB package. The actual e-book spine consists of HTML documents with CSS, JavaScript, multimedia files, etc. in addition to verifying the Epub specific data, Ace essentially orchestrates the analysis of individual web pages under the hood. This is performed by another open source tool called Axe, created and maintained by an organization named Deque Deque Axe is the core engine that checks HTML documents based on an extensive list of accessibility rules. The reported issues are labeled with different impact levels best practices minor, moderate, serious, or critical issues are also categorized on weight. WCAG version 2.0 2.1 2.2 with their level A, double A or triple A. It is worth noting that DAISY Ace customizes Deque Axe in order to meet Epub specific requirements. We therefore maintain an open source fork of Axe that we integrate in Ace. As previously mentioned, Ace is a command line tool. The terminal executable, which is essentially a large JavaScript application, is published at the official node package manager repository NPM. The tool runs in a single pass and generates a report which consists of several files saved to the local file system. The raw data is available in Json format, but Ace also produces an actual page that presents the list of issues in a tabular fashion. In the table, display rows can be filtered by severity and category. For example, moderate WCAG 2.0 level double A column headers can be clicked to sort rows alphabetically, and the rows are paginated to limit the number of items per screen. The generated file set also includes the extracted publication images, which are presented in the HTML report alongside their descriptions. The Ace app offers a graphical user interface that is more user friendly. The application can be installed in usual way for windows, Mac and Linux. No advanced knowledge is required. The interactive report table is presented inside the application window in a familiar way, similar to the previously mentioned report. Ace app includes a bonus feature which isn't present in the command line tool. There is a basic metadata editor, which can be used to quickly and easily add, remove, modify metadata in the output XML package file located inside the publication container. The modified Epub is immediately reevaluated to check whether the remediation worked. We know from user feedback that accessibility issues reported for HTML files are difficult to locate inside the original source markup. Ace currently shows an HTML code snippet and a somewhat cryptic technical pointer. We plan to make this easier by export annotations that will then be displayed inside a reading system at the exact location where the accessibility issues were encountered. Crucially, DAISY ace integrates the DAISY Knowledge Base, or KB, for short. The KB helps interpret the accessibility issues reported by Ace, and it provides guidance on how to solve problems. Ace generates Json and HTML reports with links to the KB website, but ace app contains an offline copy of the knowledge base so that an internet connection is not necessary. To conclude this short presentation. Here are some key takeaway points. Ace by DAISY performs an exhaustive list of checks and consolidates issues into a detailed report. The core specifications that describe the list of rules are Epub Accessibility and WCAG. or W C A G the DAISY Knowledge Base is an essential documentation hub to help interpret and address accessibility issues. The Deque Axe project provides additional information about web specific issues. Ace is a terminal tool, CLI for semi-automated processes, and Ace app. G UI is an application for interactive workflows. And that's it for now. Thank you.

Many thanks, Daniel, for that insightful presentation on Ace. Our final presenter is Laura Brady. Laura describes herself as an accessibility busybody, which is shorthand for somebody who's incredibly knowledgeable and deeply committed to accessibility, and who works with a wide variety of organizations to aid them in their accessibility journey. We're delighted to have you join us. Laura, over to you.

So I want to talk about InDesign as an e-book creation tool today. Is InDesign a useful e-book tool? In my opinion, that's the wrong question to ask. InDesign is page layout software that can do other things, but because it's the primary starting point for 95% of books, many of us have to use it to make ebooks. And the good news is that, it's getting better at making ebooks. In the next few minutes, I'm going to demo some of the ways that I use InDesign to make Epub that prioritizes accessibility. And for the purposes of this video, I'm using InDesign version 20.1 set to the typography workspace. If you're an InDesign user, you might consider keeping your version of InDesign up to date as the changes to how epubs are exported. they are modified just a little bit with every fresh release, so do yourself a favor and keep it up to date. So the first thing I want to talk about is creating a page list. InDesign is now capable of creating a page list out of the box that needs almost no attention or remediation from developers. This is a major upgrade over how you perhaps recreate it from InDesign, even from about six months ago. page list is a literal list of all the print pages in a document, with links to the location of the page, breaks in the navigation document, and when an e-book is created from InDesign, that navigation document is always called the TLC.XHTML The page breaks in the file must be marked with specific HTML and then gathered in the navigation. The main concern for InDesign users is to maintain the page breaks as they are. as developers get a publication ready for export to Epub. This isn't always a straightforward or easy to be frank, especially for content that needs a lot of items like images, sidebars, and marginalia anchored there. The anchoring process is at present a little glitchy and will often cause reflow. So fair warning about that idiosyncrasy. So in the screenshots that you're seeing now, you can see that there's an image that's just sort of placed on the page with some text wrap around it. And then when I anchor it, the text flows behind the image, which is problematic because then where the page breaks is suddenly changed by two full lines. So in order for your, page list to be faithful to the print production, to the print version of the file, you need to pay close attention to what the anchoring process does and modify the text on the page as necessary. Before you export to Epub. Other than that, the thing that you need to do is a lecture page. Page navigation. As you export the epub on the general tab of the Epub export method, the page navigation button will be checked by default. The decision you need to make is whether to add a print ISBN as you export the book. Not including this print corollary identifier will earn you an error in Ace by DAISY. So do consider including it. If the e-book is a digital first or digital only, you can opt for none in that same window or point to another non ISBN to source. So just be aware of all of those things as you create ebooks. Accessibility metadata is the next topic. Another way to level up the accessibility of your e-book is to include accessibility metadata describing the accessibility metrics of that book in detail. And using these data points makes it easier to find for people to find and read e-books that meet their reading needs. Schema.org is the flavor of metadata that's included in the OPF file of the e-book, along with other basic bibliographic metadata like title, author, copyright statement, etc. so use file a, go to file, navigate to file, and then to File info to store that basic bibliographic metadata. for any book. Doing so will mean that that's pulled forward into the ebooks metadata, which various reading systems will use in different ways. Running headers and footers, that sort of thing. And then when exporting to Epub, in the Epub Export Wizard, navigate to the metadata tab and then to the accessibility Subtab. The full set of schema.org values are built in here features hazards, access modes, and an open field for an accessibility summary. You can also include what's known as conformance or evaluation metadata, stating which level of the Epub accessibility spec your e-book hits with a certifying body is, and a link to their credentials, and to certify a report should exist. So full set of really interesting metadata options that you can now include at the Epub export level. So I want to switch gears now and talk about a few InDesign scripts that I use every day. dpub Aria or digital publishing Aria are semantics that, are an important part of accessibility, as this kind of additional information about important structures makes sure that common publishing markup is accessible to assistive technology, and veteran users of InDesign will know that working with text threads, that is, content that is in one long text thread from page one to the end of the book makes it hard to apply semantics and InDesign semantics only work at the object level, and anything applied at the object level goes missing entirely when the text is threaded. Break text read is a script that ships with a design, and this, and is a secret efficiency tool that's been hiding in plain sight for several years. And the utilities panel, which you can access by going to window utilities and then scripts, navigate to the community folder and find the script called BreakTextThread.jsx This handy script can break content between chapters, letting the creator apply semantics to the starts of chapters and other structures in the book so you can break the content before or after the selected frame, or you can ask it to break the content, to break all the threads at a specific paragraph style like chapter title or heading one. And then with the arrow or selection tool activated, select the first frame in that chapter and then navigate to Object Export Options to apply the appropriate Epub type. Now, I know lots of you are going to balk at putting time and attention into this relatively useless version of semantic inflection, but I want to point to another tool. If you use Sigil to edit your EPUBs, there's a free plugin called Access-Aide, and it's access aide with an E at the end, and access aide is hyphenated. This is a plugin that you can add to your version of sigil, that will automate mapping those Epub type semantics to the more meaningful Aria roles. And this can be really handy. This is very useful and will make the whole process of making a fully fleshed out and accessible epub just a little bit easier. Thanks, Laura. And thanks also to Daniel and Roman for being excellent guests in this webinar. That's us for today. But before we go, I just have time to remind you that in the treasure trove that is the DAISY Webinar archive, you'll find more than 30 hours of video articles and links to resources related to accessible publishing. This webinar is part of our EAA Countdown Series, a 12 month program exploring all aspects of accessible publishing and reading, facilitating knowledge sharing and helping all concerned to understand and prepare for the European Accessibility Act. We'll be returning on March the 26th with 94 days to go. We're diving into the topic of accessible math and science in digital publications. Representing math and scientific information is often referred to as complex content and for good reason. These can be challenging materials to make accessible in any format, and that includes within digital publications. This webinar will highlight the latest developments in math and science content authoring and reading, making complex content more manageable and accessible to everyone. On April the 30th, with just 66 days to go, we'll be exploring publishing, accessibility, policy and practice. How do you communicate about accessibility? What do you say internally and what statements do you make in public? Do you make special efforts to facilitate communication with people with print disabilities to get feedback and provide support? This webinar will explore the often behind the scenes efforts that publishers and service providers can make to support their accessibility practices, and to ensure that their intended levels of service are experienced by users. And when things go wrong, how they might be addressed in the most appropriate way. And on May the 28th, with just 31 days before the European Accessibility Act comes into force, we'll be looking at AI and accessible publishing. Well, debate about artificial intelligence is everywhere. With new and improved services being launched, it seems every week and the promise of automating tasks and producing higher quality content. But what might these developments mean for accessible publishing? Do they offer practical solutions? Or do the costs and risks outweigh the benefits? Join us as our panel of experts discuss their experiences of using AI and accessible publishing in a variety of ways, and how the promise of AI might become a reality. Find out more information at DAISY.org/webinars, where you can also sign up to the webinar announcement mailing list to learn about new topics as we add them. If you'd like to suggest a subject, or if you'd like to share your perspectives on the forthcoming European Accessibility Act, then please email us at webinars@DAISY.org. Thank you for coming today. I hope you'll join us again next time. Goodbye.