# Creating and Reading Accessible Math

Date: October 20, 2021

Full details about this webinar including links to related resources can be found on our website:
<https://daisy.org/news-events/articles/creating-reading-accessible-math-w/>

- [Stacy] Hello, everyone. And a very warm welcome to you today. My name is Stacy Scott, and I am your guest host for today's webinar entitled Creating and Reading Accessible Math. Okay, so let's begin. Today's webinar will see us take a dive into the challenging and complex topic of accessible math. This for me is a subject extremely close to my own heart. As a blind person with a degree in mathematics, I truly know the challenges first-hand of how difficult it is to read, manipulate, and write MathML in a truly accessible way. However, flash forward to today, I would say technology and solutions have come a long way and improved significantly. However, I think we can all agree that accessible math remains a very complex topic, and there is still a long way to go. This is why DAISY has brought together today an incredible panel to share their expertise and to bring us up to speed with the latest developments. In today's webinar, our panel will remove some of that complexity introducing a broad range of solutions. So prepare yourself as we get engaged with accessible equations, my personal favourite. We have lots to cover, so I will hand over to a wonderful panel of experts and ask them to please introduce themselves.

- [Neil] Hi, my name is Neil Soiffer. I work for Talking Cat Software.

- [Joseph] Hi, I'm Joseph Polizzotto. I'm an accessibility technologist at Wake Technical Community College in Raleigh, North Carolina.

- [Homiyar] Hi, this is Homiyar. I look after Benetech operations for the Asia-Pacific zone. And I look after the major programme, which Benetech runs, Bookshare.

- [Richard] Hello, everyone. My name is Richard Orme, and I work at the DAISY Consortium. Okay, so a quick overview of what we'll cover today. We'll start with an introduction to math formats from Neil, and then we've structured the session around a workflow. As Stacy said, this is a complex topic, and there are lots of different tools and techniques. So rather than show you slide after slide after slide of different tools, we'll take you on a journey through the process of creating accessible math. Going from an inaccessible image-based document to an accessible math expression experience on web pages, and we'll travel via Word. We'll demonstrate specific tools that you can grab and use, but we'll also mention alternatives as we go, and we'll be happy to discuss these and others in the Q&A session. So that's creating math. We'll then move to reading math and spend time both in a web browser and back in Microsoft Word with different screen readers and also with Read Aloud. So let's pass over to Neil who will tell us about math formats.

- [Neil] So on any journey that you take you need to be able to be a little bit familiar with the language. So where are you going to encounter documents? Typically, you're going to get an HTML document, just a nice webpage, you see them all the time. You might have run into PDFs, which are not that accessible. They could be, but it's much more difficult to make them accessible. And as a newer format would be EPUB, which basically bundles together a bunch of web pages so that you get more of a book feel out of the web pages. And because we're talking about math, there are three different terms you're gonna hear along the way. One of them that you may know about is called MathML. And I have it a little example of a square root of two as an example. And MathML is, if you've ever looked at a webpage, there's these little angle brackets, might have a angle bracket P for paragraph. In math who have an angle bracket math. And inside of that, in this example, there's an angle bracket MSQRT for square root and so on. If you're not familiar with Tex, most people, when they talk about math, they talk about only the tech part of math. And I'll focus on that. Typically the math in Tex is inside of single dollar signs, or maybe double the dollar signs for a larger math expression. But sometimes it's inside of other bracketing things. And if you look at Tex, you're going to see a lot of backslashes, in this case, backslash SQRT for square root. And you'll see a lot of braces. In this case, you take the argument and you put it inside of braces, so there's a two there, and braces are basically invisible parenths. And that's what Tex or LaTex looks like. LaTex is just some more packages on top of Tex. Another format you might come across is ASCIIMath, and ASCIIMath is sort of like what you might type to a calculator. So in this case, it's just SQRT two, it's harder to get some hard structure into ASCIIMath, but it's simpler to get simple structure in. And lastly, we're going to get at the end, we're gonna talk about editing. And a lot of people do the editing in Word. And there you can put math into Word. A common form that you'll often see in some LMS tools is called MarkDown. And it's kind of a short form for doing webpages. So instead of typing angle bracket this or that, you often might have a hash sign for a heading and so forth. And then often just a plain text editor is where you're going to do editing on math, especially if you're doing the editing on math in ASCIIMath or Tex. So next up we have Joseph.

- [Joseph] Great, thanks, Neil. As Neil mentioned, there are a variety of math notation formats, and having a document that contains one of these formats, LaTex, MathML or ASCIIMath can help tremendously when you're creating an accessible document or accessible content. We're going to be taking you through an example of a workflow today, however, that involves an image-based math document as our starting point. And in this situation, most disability service offices and higher ed can spend a great deal of time offering their own document so that the math can be prepared and published into an accessible format to students using assistive technology, such as a screen reader, a text reader, a magnifier, or a refreshable braille display. In my career, working in accessibility in higher ed, I've been pleased to see a development of more OCR tools that can be used for mathematical content. And by OCR I'm referring to optical character recognition. And there are currently three popular programmes that can be used for math OCR in this space. EquatIO is one, MathPix, and InftyReader. And while there's no doubt that each one of these tools can save you time, the crucial factors that determine which tool you choose will have to do with your institution's approach to converting documents. Some tools may work better for content creators, like faculty or students themselves, whereas other tools might be more appropriate for an alternative media production house that's converting lots of content on behalf of students and academic departments. When you look at these OCR programmes, there are two types of use cases that I want to highlight. One is you can take a screenshot of an individual math expression and then paste the contents of the accessible math into an editor. This screen grabbing or screen snipping type of software works well when you do more ad-hoc conversion of math, perhaps you have a particularly large math object that could take a long time to author in an editor. Another approach is to have the entire page of a document, math plus text surrounding it converted to accessible format. This is obviously a boon when you have lots of math on one page and trying to turn a document around quickly. In this table, we're showing which math OCR tools can be used with each of these approaches. For doing screen grabbing of math content you can currently use MathPix Snip and EquatIO, and with InftyReader version three you can use a screenshot reader of Adobe Reader. For converting entire documents with math content, you can either use MathPix OCR or InftyReader. And keeping with our workflow, which we're going to be going through Word, each of these math OCR programmes can be outputted to a math format in Microsoft Word. When you're screen grabbing a math equation, you can paste the math as MathML directly into Microsoft Word. And when you do that, you'll see that the math appears as Microsoft Word's native math equation format, or Office math equation format. If you have the programme MathType installed, you can also choose to paste the math as a MathType equation. If you're converting an entire document, on the other hand, MathPix OCR and InftyReader both will create a new document from your image file. MathPix will output a docx and InftyReader will create an XML file that you can open up in Microsoft Word, and then save as a docx. When outputting documents to Microsoft Word, MathPix OCR and Infty will convert the math into MathML, which will be displayed in Office math or a native equation format. And we're now going to see an example of how this process works with one of these programmes. Here I have a PDF with images of math. I have extracted the text to a Word document, but the math is not accessible. I have images of math as a placeholder for easier editing. With EquatIO's screenshot reader tool I can draw a box around an image of math and select the copy MathML option. Then I can paste the MathML into my Word document. I now have an Office math equation so I can delete the image of the math. I can proceed in the same way for the other equations I need to convert. First I will draw a box around the math, I will select copy MathML, I will then paste the MathML into the document. Finally, I will delete the image or placeholder for the math.

- [Richard] Thanks, Joseph. Well, thanks to the workflow so far we now have a Word document with math expressions in there, rather than images. If you're originating a document yourself, you may want to enter equations rather than convert them from an image or you may want to edit the equations that have been imported. And to do this, there are some options for using an Equation Editor. And the first I'm going to talk about is the Microsoft Equation Editor, which is built into Office. So that's not only in Word, but also in Excel, Outlook and PowerPoint. Now the Microsoft Equation Editor used to be a poor relation to other options, but it has improved a lot in recent years. And the expressions that you create using the Microsoft Equation Editor will be in that Office math format. So there are various options for entering math expressions, including picking from built in common expressions, using a graphical editor or typing in what Microsoft refer to as the linear format, or even handwriting. So let's look at these with a simple example, which is the area of a circle. So here's the built-in expressions method. I go to insert equation, and I pick, there it is at the top conveniently, area of a circle. And it will place that math expression where the cursor was. Okay, so that was too easy, wasn't it? Let's now look at using the graphical method using the GUI math expression editor. So I'm going to go to insert equation. It places a math area, and I can just start typing A equals, and now I need to do a Pi. Well, I haven't got Pi on my keyboard, so go up to the ribbon and I'm going to the symbols area, and I'm moving from basic math through to Greek letters. So it's got uppercase and lowercase Greek letters. I'm looking for the Pi. There it is, I click on Pi, it's now inserted that. Now the next part of this expression is R squared. This is a structure, and up on the ribbon there's an area with structures in. If I go up there, I can see that there are different ones for fractions, for example, and what I want is a script, and this is just a simple superscript. So I click on that, and it inserts two dotted boxes into my math expression. So I can now just cursor into these and type in the relevant pieces. So an R in the bigger box, move right, type a two into the smaller box, that gives me my superscript, and I'm done. Okay, so next let's look at using the linear format method. This can be fast once you know how to do it. I'm using the keyboard for this, and the keystroke in Microsoft Word to insert formula is alt equals. So I've pressed that and it makes a math area for me, and I can start typing, A equals. And now I'm going to use the expression, I'm currently in Unicode mode. It's kind of similar to LaTex that was mentioned by Neil. And to get a Pi I'm going to use a backslash and P-I, and press space, and it makes a Pi symbol for me. Then I want an R, so I've done the R, and then to get a superscript, I don't actually know the name of this character, it's kind of a pointy up a symbol that's above my keyboard, above the number six. Is it called a caret, and then two. So it's that pointy up and then two, press space, and it turns that into the superscript two for me, and I'm done. So that can be really quite a quick method once you've learned those techniques. And then lastly we have the inking method, so I can use any kind of pointing device. I happen to have a computer with a Stylus, so that's what I'm going to use. So I'm going to go to insert and equation. And one of the options I have over on the left on my ribbon is ink equation. So I'll just click on that. And it puts a math input control up on the screen. And I write the math using kind of handwriting in that space. So I'm drawing a Pi, no, I'm not, it starts with A, doesn't it? A equals Pi R and then squared. And as I'm writing that, it's recognising that handwriting, putting it into the kind of preview box, and when I'm happy with it I can click on the insert button and it places it in there for me. So those are some different techniques I can use using Microsoft's built in Equation Editor. And these will all mean that I have Office math expressions in my document. The other equation editor that I have on this slide is MathType, we've had that referred to already. This is a powerful equation editor that can be used in Word, also Google Docs, D2L, WordPress, it has lots of different integrations. it's available for Mac and Windows. MathType actually uses its own binary format. It's a commercial product, but many people find it affordable. And there are licencing options for educators that bring the cost down further. In the past, people preferred to use MathType over Microsoft's Equation Editor because it was much more powerful and more accessible too. But I think it's fair to say the gap has closed a lot. There'll be some people who love to use MathType, and others for whom the built-in Microsoft Equation Editor is just fine. It's nice, isn't to have options. Okay, so we have a lovely Word document with lots of math expressions in it. Some of them have been converted from images, we've created some ourselves, and we've certainly gone through and checked the ones that have been created using OCR to make sure they'd been accurately imported. And of course, with this Word document, we will have paid attention to other accessibility best practises, such as using styles, having good navigation by the appropriate use of headings, describing images or marking them as decorative, and so on. Now, the workflow we're showing here is to create accessible web pages, so we need to go from Word to the web, or at least to HTML files that could be read in a browser. We have some options, Word has a feature, which is save as webpage or save as filtered web page. This will result in kind of okay HTML, some would say it's rather messy. And for our purposes it isn't useful because the math expressions are exported as images without ALLText. So another option we have is MathType, which has a feature called publish as math page. I think this builds on Microsoft's HTML export, but it replaces those naughty images with MathML. And this feature works well. It's easy to use and screen reader users report that the resulting math pages are a good experience. You still get rather messy HTML, but I've never heard of an end user complain about this. I don't know if any LMS objects to it. But our images, or our math expressions rather are in Office math. So you could convert them to MathType before publishing as webpage, but we're going to use a different tool, which is WordToEPUB. This is a free tool from the DAISY Consortium for Windows, and it converts structured and accessible Word documents to accessible EPUB, and it handles maths as well. Happily for our purposes, it will also convert to HTML. So we'll use that tool to turn Joseph's document into a webpage. So saving as HTML is not enabled by default, so first we need to enable it in the preferences. So here I am in Word, I'm going to go up to WordToEPUB in the ribbon, and I'm going to click on preferences, and in user interface options we have the option offer HTML output format. I've turned that on, we only need to do this once. So back in the tool I can just move to advanced mode, and I'm going to specify the output file name for this, and then click on the HTML tab, create HTML version. And that's it, we've gone from images of math expressions that Joseph started with to Office math, and now to MathML in a webpage. And so now it's over to Joseph.

- [Stacy] Thank you so much to all of our panellists. So we do actually have a lot of questions. So any that we can't answer today we will answer via email. So if we start with questions for each of our panellists, can we start with Homiyar? Bookshare is known for creating accessible books, including STEM materials. How close is your current practises, how close is this to your current practises?

- [Homiyar] Thanks, Stacy, for bringing them in. And, as you all know, Bookshare is the largest library for accessible content. At Bookshare we have collections mainly based as EPUBs. And Neil, as he explained right at the beginning, it is nothing else but a collection of webpages put together in an organised fashion. Now (interference drowns out speaker) panellists were mentioning, most of the EPUBs, which were donated to us by our partners, didn't have math content, but it was all stored as images because the technology wasn't there. And because Bookshare has been gathering contents since the last 20 years, and technology wasn't there since the last 20 years, which was able to make math accessible. But fortunately, in the last couple of years the technology has evolved and all content containing math, which has been reconverted or refitted, or retrofitted by our production houses has math in an accessible format. That means any content, which has been manually reproduced by our vendors will have math in an absolute accessible format just as what Joseph or Richard demonstrated. But for those content, which have been given to us by our publisher partners, Bookshare is now working on an engineering solution, which will mass convert all the math titles that it has in two phases. In phase one, we're going to provide Alt Text to all the images which contain math. And in phase two we are going to actually render all these equations accessible using MathML. So fortunately, we have already completed phase one where we have retrofitted more than 20,000 titles with math and made them accessible using Alt Text. The phase two begins in early next year, so all of our readers should be able to enjoy more and more titles with accessible math from Bookshare. Back to you, Stacy.

- [Stacy] Thank you so much, that was really helpful. I have one for Neil, please. So, Neil, there are many combinations of browsers reading systems and assistive technologies. Do you have any recommendations for the best outcome?

- [Neil] Okay, so I have to start off with a disclaimer, as I worked on MathPlayer, and so I'm very biassed in some sense. So I think MathPlayer with NVDA gives the best reading experience. You heard what the reading was like for a mixed fraction from some of the other systems. So you may prefer them. Firefox with NVDA works very well. Chrome has improved greatly in the last few years to work with it, so both of those are pretty good solutions.

- [Stacy] Great, thank you very much. And I have to say, it was interesting hearing the whole variety of different screen reader voices that I haven't heard before. So I may have to start picking some new ones. Richard, I have a couple more questions for you, please, if you don't mind. Selene asks, does it work equally well with PowerPoint?

- [Richard] So Office math is supported within PowerPoint, and in theory it should work. I have done some experiments using NVDA with Office math using that trick that Neil showed of changing the preferences in the advanced part of NVDA. I actually wasn't successful in reading math in PowerPoint slides, but this could be user error, but I can't report success at this time.

- [Neil] So let me answer it slightly. If you're using the MathType version of math in the slides, and I think both JAWS and NVDA will read the PowerPoint, at least NVDA will read the PowerPoint. I think JAWS also does, but I'm not sure about JAWS, but it has to be the MathType math, but you can convert the Word to MathType.

- [Richard] Thanks for that clarification. As you said in your introduction, Stacy, this is a complex area.

- [Stacy] It definitely is, so there's one more question. So Angie would like to know, does it work also with chemistry?

- [Homiyar] Yes, we have created chemistry using the equation editor. It does come out pretty well in NVDA response to all the arrows and all the notations. But sometimes NVDA gets a little confused with the symbols. Like F, and FE, it will try and read out its original name, FE and F, and sometimes if the editing is not done properly, NVDA goes for a toss. So sometimes it will pronounce, what do you call, the elements properly, and sometimes it will just read out the characters. But in most cases it works properly.

- [Neil] So let me throw in my two cents here. As far as I know, MathPlayer is the only system that's able to read the chemistry at the moment. And the chemistry detection depends upon having enough chemistry there for MathPlayer to realise it's chemistry. So if you have just an FE, it's not going to realise it's chemistry, even if it's got maybe a subscript or superscript. But once you get to a few chemical symbols in there, like the arrows and the brackets for concentration, then it turns on. And I'm going to take this opportunity to mention there's a MathML working group working on the next version of MathML. And one of the things we're working hard on is having a way for authors to express what they really mean. So one possibility then is for them to tell us that this is chemistry, and then there'd be no more guessing about whether it's chemistry or not. And then hopefully the systems will be able to pick up on that and do a proper job of reading the chemistry. So I apologise for a little bit long-winded answer there.

- [Stacy] Wonderful, it's always good to have as much information as we can get in these sorts of things. Just another question we have from Luke. So again, on MathType, MathType uses the image alternative text to host its MathML format, and it's phonetic Read Aloud information. Is there a way to swap this around so screen reader users don't have to listen to the MathML?

- [Richard] If anyone can answer that, I hoping it's Neil.

- [Neil] I don't think there's a way to do that. I could be wrong, I've been a MathML fanatic, I haven't tried to get the alternative text. I'm not sure why that would be good. So one of the things that you may have heard is that there's pausing and there's actually a forced long A sound to it. And if you use alternative text, the speech engines read the math as English text, and they tend not to do a particularly great job with that.

- [Joseph] Just a thought here. If it was the case that the screen reader user would find a better experience navigating Office math equations rather than MathType equations, there are ways to convert as a batch process, like Neil was showing, all the math equations from MathType to Office math. And in that case you would use a utility called GrindEQ MathTypeToWord. So that might, in the end, if that's what the desired outcome is in this case, be something to look into.

- [Stacy] Okay, great, thank you. So a question from Sarah. So Sarah, as more K-12 teachers are moving to Google Docs using the Google Equation Editor, do you know of any ways to make those accessible?

- [Joseph] The accessibility of math within the Google Docs environment in my view is only possible when using like the read and write tools by text help, read and write will allow you to have those Google Doc equations read out loud. I have not seen screen readers being able to read that math content correctly, but I would be happy to be updated on that point.

- [Neil] Google Doc math is its own little world, and they have don't expose any information to anyone. And so the way text help gets it is they have images and they pull information from the images, like Alt Text and stuff. So, yeah, Google Docs are a really bad experience for math, cited or un-cited.

- [Stacy] Okay, thank you. I have a question for Homiyar here, please. Marianne has asked if there is a particular font that you can recommend that is good for displaying formulas.

- [Homiyar] I believe we use a standard Calibri for our text production, and I don't think we change anything even for the math part.

- [Neil] Yeah, I think for the math in Word they use what's called Cambria Math, which was designed specifically for math for Word, but in a web doc, I think MathJax is using either the sticks fonts or some other font, I'm trying to remember what it might also use, but I think it uses the sticks fonts.

- [Joseph] I think that's correct, sticks, yeah.

- [Stacy] Okay, thank you. We have a question from Kaylos, is MathPlayer the best solution for reading math in braille?

- [Neil] Okay, so I'm going to take this opportunity to do a little bit of self aggrandisement maybe is the word. So MathPlayer actually takes advantage of LibLouis, which I believe JAWS also uses that, and there are a number of bugs in LibLouis for generating Nemeth math. And MathPlayer was a commercial product for free from MathType. MathType was bought by Wiris. And Wiris after several years of discussion with me, I was trying to fix some bugs. And MathPlayer has decided that they do not want to support MathPlayer. And so MathPlayer has no further development going on. I've taken this as a challenge and I am developing a new system to replace MathPlayer called MathCat, Math Capable Assistive Technology with a very cute logo. So I know it's a good name because of the cute logo. And as part of that, my last six weeks or so, I've been re-implementing a braille solution that is vastly better for Nemeth, and in fact I've compared it to all the other ones out there except for the Duxbury solution. That's the one I need to try and get a version of. And all the other solutions have lots of bugs. I'm really amazed people have not been screaming about how bad the Nemeth generation is from these other systems. But I'm happy to say MathCat is doing a pretty good job. I have a lot of tests, several hundred of them. And unlike other systems where they fail like well over 50% of the tests, MathCat's got them all. So I hope that when MathCat comes out by the end of the year, you'll have a better solution. It will hook up with and NVDA. And there's been some discussion with JAWS as to whether they might move over to MathCat. They're very slow in making these kinds of decisions. And so if they do, I'm sure it will be years from now. Sorry for tooting my own horn.

- [Stacy] No, that's absolutely fine. So Luke asks, what process would you recommend for making untied PDF documents containing LaTex accessible?

- [Joseph] I might take that one. Well, I would go back to the source LaTex and experiment with one of the command line tools, like Tex for HT and run the make for HT command, and directly export the LaTex to HTML. And if that doesn't work, you might also try, WordToEPUB does have a LaTex converter that allows you to bring the LaTex file directly into Word, and then you could follow the same workflow that Richard outlined where you could output the Word document as HTML or EPUB. So I would try to go out away from exporting to PDF as your target format towards HTML or EPUB.

- [Stacy] So unfortunately I must call time, as all good things must come to an end, including this wonderful webinar. Once again, thank you so much to our wonderful panel, Homiyar, Joseph, Neil, and Richard for sharing their time and expertise with us all. Just before we close, I wanted to share some details of some great webinars by DAISY in the coming weeks. In two weeks, November 3rd, a panel of experts will introduce the incredibly important topic of validating and conformance checking EPUB, and opening up a box of free tools to help guide us through the processes. On November 17th, we dive into creating and editing an EPUB, exploring some of the latest tools for producing quality publications. And on December 1st, we return to the topic of the EU accessibility act, a legislation, which is positioned to change the way material is published, sold and consumed, not just within Europe, but has implications more globally. If you have any ideas for future webinars or would like to suggest a topic, we would love to hear from you. We hope you'll join us again soon. In the meantime, thank you for your time, stay safe and well, and have a wonderful rest of your day. Take care, goodbye.