MathJax is supported by the American Mathematical Society and many sponsors.

CSUN 2016, San Diego, March 24 2016
Accessibility to Mathematics is essential for inclusive education

TTS of Mathematics is a challenging problem

Particular on the web as mathematics is badly supported

MathJax is already a visual rendering solution

Make it universally accessible

Instead of relying on browsers or screen readers we have created an AT solution in MathJax

Based on some work done in ChromeVox at Google and later extended in Benetech’s MathMLCloud project

Now supported by AMS and Sloan Foundation
MathJax is a JavaScript library for rendering Mathematics in all browsers.

- Can take \( \text{LaTeX} \), AsciiMath, and MathML as input.
- Generates browser output, e.g., HTML/CSS, SVG.
- Standard Maths rendering solution for: stackexchange, wordpress blogs, mediawiki, etc.

- Internal format is (still, something close to) MathML.

MathJax is the de facto rendering solution of (nearly) all Mathematics on the web (35 million unique daily rendering requests via CDN).
MathML is officially part of the HTML5 standard.
Mathematics should be formatted in (presentation) MathML.
Generally this is not the case: Instead it is given as LaTeX or ASCIIMath.

MathML has very limited support from Browser vendors
- Two incomplete implementations: FireFox (Gecko), Safari (WebKit)

MathML spec is seriously outdated
- In particular it does not take modern web technology into account (HTML5, CSS)!

There is no WAI-ARIA spec for Mathematics
MathPlayer
  - Only until IE 9 on Windows (MathML and MathJax)
  - Somewhat in IE11 (not in Edge)

ChromeVox in Chrome (works on MathML, and LaTeX and ASCIIMath via MathJax)

VoiceOver has some support for MathML

NVDA, Jaws via MathPlayer, now via MathJax and MathPlayer library
Maths will never be first class citizen in all browsers

It is also too much to expect Maths solutions from general assistive technology providers

MathJax is the Visual Rendering solution

Turn MathJax also into an Assistive Technology solution

Support users with a wide variety of print impairments

Enable magnification, simplification, highlighting, aural rendering, etc.
MathJax’s Renderers

- MathJax provides a variety of renderers
  - CommonHTML, SVG, HTML/CSS, native MathML, ...

Volker Sorge
New Accessibility Features in MathJax
MathJax uses Presentation MathML as internal format
Mathematical information is rather trivial

Example: Quadratic Equation

\[ax^2 + bx + c = 0\]

is commonly represented in linear form in MathML:

```xml
<math>
  <mi>a</mi>
  <msup>
    <mi>x</mi>
    <mn>2</mn>
  </msup>
  <mo>+</mo>
  <mi>b</mi>
  <mi>x</mi>
  <mo>+</mo>
  <mi>c</mi>
  <mo>=</mo>
  <mn>0</mn>
</math>
```
Semantic Enrichment

- Impose “light” semantic interpretation on MathML expression
- Rewrite syntax tree into a term tree using heuristics:
  - Combine operator and relation sequences,
  - Determine potential function applications,
  - Break up symbol sequences into elided products,
  - Combine bracketed expressions as much as possible,
  - Recognise scope of big operators (e.g., sums, integrals),
  - ...

- Originally developed in ChromeVox for K-12 Mathematics
- Clean interpretation regardless of how horrible the MathML
ax^2 + bx + c = 0

is rewritten from its Presentation MathML representation into its semantic interpretation:

```
<math>
  <mi>a</mi>
  <msup>
    <mi>x</mi>
    <mn>2</mn>
  </msup>
  <mo>+</mo>
  <mi>b</mi>
  <mi>x</mi>
  <mo>+</mo>
  <mi>c</mi>
  <mi>=</mi>
  <mn>0</mn>
</math>
```
Combining Semantic and MathML

- MathML is internal representation in MathJax
- Embed the semantic interpretation directly using HTML5 data attributes
- Alternative view on the MathML element, by providing an orthogonal tree structure
- Data attributes are retained in the rendered expression regardless of the particular MathJax renderer used.
Responsive Equations and Abstraction
Highlighting
Interactive Exploration
Speech Generation

UX the same regardless of the renderer.
Responsive design enhances a core feature of HTML: reflow
Re-arrange, optimise, and transform content:
  - cropping images, abstracting icons, modifying tables
Mathematics combines the properties of text, tables, and graphics into a single problem
Content is usually created with print in mind: manual layout!
Responsive Equations

- Automatic reflow for simplifying layout, adapting to form factor of display and magnification
- Intelligent linebreaking by exploiting semantic enrichment
  - Don’t break in the middle of an expression
- Chunking: Abstracting over large elements
  - collapsing mathematically meaningful sub-expressions
Example

\[ I_\nu(\nu^{-1}, 1) = \frac{\pi^2}{4} \ln \left( \frac{(1 + \nu)^{1+\nu}}{\nu^\nu} \right) - \frac{7\zeta(3)}{8} \nu + 2 \int_1^{\frac{1-\nu}{1+\nu}} \frac{\chi_3(\nu)}{(1 + \nu)^2} \, d\nu \]

Let this be C

\[ = C - \frac{2\chi_3(\nu)}{1 + \nu} \bigg|_1^{\frac{1-\nu}{1+\nu}} + 2 \int_1^{\frac{1-\nu}{1+\nu}} \frac{\chi_2(\nu)}{\nu(1 + \nu)} \, d\nu \]

\[ = C + (1 - \nu)\chi_3 \left( \frac{1 - \nu}{1 + \nu} \right) - \frac{7\zeta(3)}{8} - 2\chi_2(\nu) \ln(1 + \nu) \bigg|_1^{\frac{1-\nu}{1+\nu}} + \int_1^{\frac{1-\nu}{1+\nu}} \ln(1 + \nu) \ln \left( \frac{1 + \nu}{1 - \nu} \right) \, d\nu \]

\[ = C + (1 - \nu)\chi_3 \left( \frac{1 - \nu}{1 + \nu} \right) - \frac{7\zeta(3)}{8} + 2\chi_2 \left( \frac{1 - \nu}{1 + \nu} \right) \ln \left( \frac{1 + \nu}{2} \right) + \frac{\pi^2}{4} \ln 2 \]

\[ + \frac{1}{2} \int_1^{\frac{1-\nu}{1+\nu}} \frac{\ln^2(1 + \nu) - \ln^2(1 - \nu) + \ln^2 \left( \frac{1-\nu}{1+\nu} \right)}{\nu} \, d\nu \]

Example of mathematics “in the wild” taken from math.stackexchange.com.
Exploring Equations

- Collapsed parts are represented by a simple meaningful Unicode construction, ◀X▶.
  E.g., ◀()▶, ◀f()▶, ◀+▶, ◀√▶.
- Dyslexia support via highlighting
- Customisation of fore- and background colours for high contrast
- Highlight mathematically meaningful expressions

**Syntactic highlighting:**

\[ I_{\nu}(\nu^{-1}, 1) = \frac{\pi^2}{4} \ln \left( \frac{(1+\nu)^{1+\nu}}{\nu} \right) - \frac{7\zeta(3)}{8} \nu + 2 \int_{1}^{\frac{1-\nu}{1+\nu}} \frac{\chi_{3}(v)}{(1+v)^2} dv \]

**Semantic highlighting:**

\[ I_{\nu}(\nu^{-1}, 1) = \frac{\pi^2}{4} \ln \left( \frac{(1+\nu)^{1+\nu}}{\nu} \right) - \frac{7\zeta(3)}{8} \nu + 2 \int_{1}^{\frac{1-\nu}{1+\nu}} \frac{\chi_{3}(v)}{(1+v)^2} dv \]
Interactive Exploration

- “Walkers” allow to interactively dive into mathematical expression
- Synchronised highlighting together with aural rendering
- Simple JavaScript extension for MathJax
- Different types of walkers: syntactic, semantic, collapsed
- Simple navigation model using arrow keys
- UX the same regardless of the renderer
Speech strings are computed with Speech Rule Engine initially implemented in the context of ChromeVox and extended for MathMLCloud.

Currently uses the MathSpeak rules: verbose, brief, superbrief

- special summarisations for collapsed parts
- Other rule sets and localisations in the future

WAI-ARIA and CSS to implement interactive exploration

- Speech output by updating ARIA live regions
- Colour/contrast changes by rewriting CSS properties

Speech strings can be precomputed or generated on the fly

Works for all renderers MathJax provides
Tested with combinations of the main browsers, screen readers and platforms

- IE 10-11, Edge, Chrome, Firefox, Web
- NVDA, Jaws, WindowsEye, VoiceOver, ChromeVox, ORCA
- Windows (XP, 7, 8, 8.1, 10), MacOSX, Linux

Responsive Equations
- Feedback from experts and users of Stack Exchange
- AT extension: Collection of user feedback is currently in progress
  - With (blind) experts, users and accessibility support officers
  - Most critique currently focuses on MathSpeak and its verbosity
  - Navigation generally viewed positively
Current implementation is available as MathJax extension at https://github.com/mathjax/MathJax-RespEq/
Should become permanent feature in 3.0 release
Difficult to maintain due to reliance of third parties (browsers, screen readers, etc.)
Proper implementation of standards by all would help
There is a need for a WAI-ARIA specification for mathematics or STEM in general
Web References

- **Demo:**

- **Systems:**
  - https://github.com/mathjax/MathJax/
  - https://github.com/mathjax/MathJax-RespEq/
  - https://github.com/zorkow/speech-rule-engine/
  - https://github.com/mathjax/MathJax-node/