Towards ARIA Standards for Mathematical Markup

Volker Sorge

MathJax Consortium

(University of Birmigham, UK; Progressive Accessibility Solutions, Ltd.)

joint work with Davide Cervone and Peter Krautzberger

This work was partially supported by the Alfred P. Sloan Foundation.

3rd International Workshop on Digitization and E-Inclusion In Mathematics and Science, 2016, Shonan, 4 February 2016

Motivation

Lessons learnt from a project supported by the Alfred P. Sloan Foundation and the AMS

- "Semantic enrichment of mathematics for accessibility and display on the web"
- Enhance MathJax for working with
  - Implement Responsive Behaviour of Equations
  - Provide an Assistive Technology Tool
- Tangible outcome:
  - A MathJax extension that does the above
- Not so tangible results:
  - A wishlist/discussion points for future standards

The State of Mathematics on the Web

- 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- MathML spec is seriously outdated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

Accessibility of Mathematics on the Web

- MathPlayer
  - Only until IE 9 on Windows (MathML and MathJax)
- ChromeVox in Chrome (works on MathML, and LaTeX and ASCIIMath via MathJax)
- VoiceOver on Safari (MathML only)
- Orca on Firefox (MathML only)
- NVDA, JAWS via MathPlayer (MathML only)
- Alt text via Maths to Image conversion tools

What is MathJax?

- MathJax is a polyfill bridging between Maths an its (lack of) web implementation
- JavaScript library for rendering Mathematics in all browsers
- Can take LaTeX, AsciiMath, and MathML as input
- Generates browser output, e.g. HTML/CSS, SVG
- Standard Maths rendering solution for:
  - stackexchange, wordpress blogs, mediawiki, etc.
  - Elsevier, IEEE, Springer, IOP, Wiley 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)

MathJax and Accessibility

- MathML 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 Technolgy 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, ...

```
<span class="math" id="MathJax-Span-7" role="math" style="width: 8.246em; display: inlin
  <span style="display: inline-block; position: relative; width: 7.115em; height: 0px; f
    <span style="position: absolute; clip: rect(1.457em 1000em 2.751em -999.997em); top:
      <span class="mrow" id="MathJax-Span-8">
        <span class="mi" id="MathJax-Span-9" style="font-family: STIXGeneral; font-style
        <span class="msubsup" id="MathJax-Span-10">
          <span style="display: inline-block; position: relative; width: 0.919em; height
            <span style="position: absolute; clip: rect(3.397em 1000em 4.151em -999.997e
              <span class="mi" id="MathJax-Span-11" style="font-family: STIXGeneral; fon
                <span style="display: inline-block; overflow: hidden; height: 1px; width
              </span>
              <span style="display: inline-block; width: 0px; height: 3.99em;"></span>
            </span>
            <span style="position: absolute; top: -4.415em; left: 0.488em;">
              <span class="mn" id="MathJax-Span-12" style="font-size: 70.7%; font-family
              <span style="display: inline-block; width: 0px; height: 3.99em;"></span>
            </span>
          </span>
        </span>

  ...
</span>
```

Trivialty of Presentation MathML

- 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:

```
<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

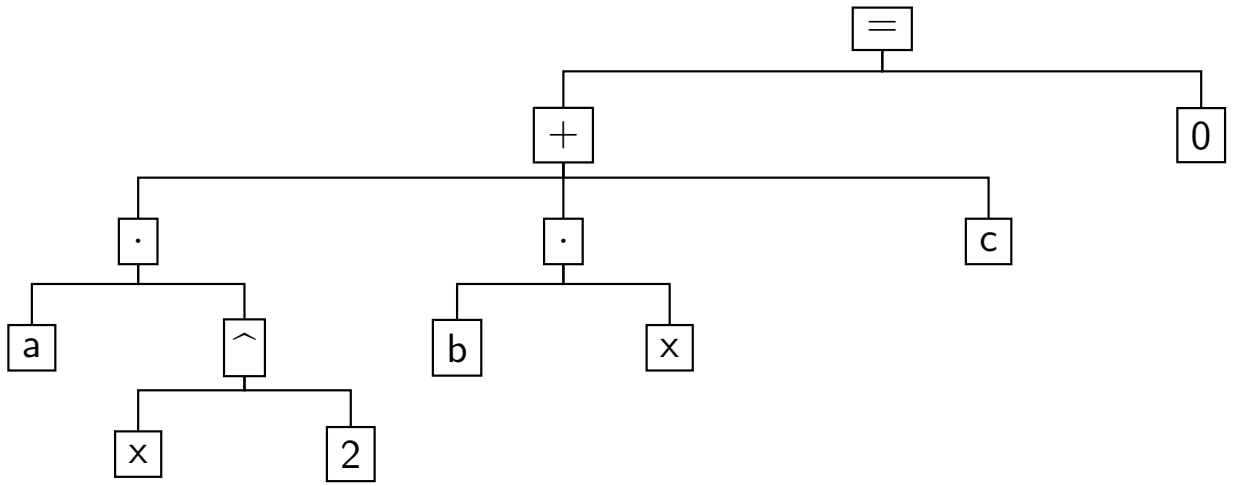
- Based on some work done in ChromeVox at Google and later extended in Benetech's MathMLCloud project
- Impose "light" semantic interpretation on MathML expression
- Rewrite syntax tree into a term tree using heuristics:
  - Combine operator and relation sequences,
  - Determine potential function applications,
  - break up symbol sequences into elided products,
  - combine bracketed expressions as much as possible,
  - recognise scope of big operators (e.g., sums, integrals),
- Embed the semantic interpretation directly into MathJax's internal MathML using HTML5 data attributes
- Data attributes are retained in the rendered expression regardless of the particular MathJax renderer used.

Semantic Tree Example

$$ax^2 + bx + c = 0$$

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>
  <mo>=</mo>
  <mn>0</mn>
</math>
```



Assistive Technology Extension

- Responsive Equations and Abstraction
  - Automatic reflow, Intelligent linebreaking, chunking and collapsing
- Highlighting
  - Dyslexia support via highlighting
  - Customisation of fore- and background colours for high contrast
- Interactive Exploration
  - "Walkers" allow to interactively dive into mathematical expression
- Speech Generation
  - Currently uses the MathSpeak rules and special summarisations

UX the same regardless of the renderer.

Universal Accessibility

Enabled for a multitude of 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

Full support matrix at `https://github.com/mathjax/MathJax-RespEq/wiki/Support-Matrix-a11y-tool`

Lessons Learned

- Better and more standardised interfaces in the Assistive Technology Ecosystem: screen readers, magnifiers, braille displays, etc.
- Semantics has to be provided regardless of the underlying implementation
  - WCAG 2.0 is not sufficient
  - WAI-ARIA needs to be expanded and become more flexible
  - Semantics for STEM should start in ARIA not HTML!
- Great example: current work of W3C SVG-4-A11Y task force

MathML is dead, but long live Maths (on the Web)

- Web rendering has three major compontens:
  - Content/Base (HTML), Styling (CSS), Semantics (ARIA)
- Two original arguments for a special Maths web standard:
  - Implementations do not support sufficient styling for Maths
    - No longer valid
  - styling influences semantics, e.g., bold letters carry meaning
    - conveying meaning via styling is a flawed approach in mathematics just as it is in any other subject domain
    - style information can still be accessed and exploited
    - the Web offers new stylistic possibilities and should not restricted to what we always had in print only.
- Having (monolithic) MathML in the HTML5 standard stops us from moving forward!

Towards WAI-ARIA for Maths: A Basis for Discussion

- Should strive for a minimial set of what is necessary
- Rather work with descriptions and pointers similar to ARIA `role` and `labelled-by`, `described-by`, etc.
- Proposed minimal set:
  - `math` (exists) for the enclosing element
  - `identifier`, `number`, `operator` (possibly with pre/in/post-fix),
  - `operand` pointer for the operands of an `operator`.
  - `divided` for fractions and fraction-like content.
  - `separator` (exists) for `mfrac` and `mstack` lines.
  - `enclosed-by` for `menclose` notation,. `mroot`, `msqrt`, `mfenced` and fences etc.
  - `scripts` (pre-/-post sub/sup)

Conclusion & Calls

- MathJax AT Extension: packaged and now mostly stable
  - Please start using and give feedback
  - Please help in localising (Unicode would be a start)!
  - Difficult to maintain due to reliance of third parties (browsers, screen readers, etc.), so community testing is essential!
- We need a minimal ARIA standard for Math.
  - This needs to be a new, MathML independent innitiative!
  - Other STEM subjects might follow.
  - Call for interest and participation.

Web References

- Demo:
  - http://mathjax.github.io/MathJax-RespEq/Semantics-Lab/Struik.html
  - http://mathjax.github.io/MathJax-RespEq/Semantics-Lab/Semantics-Lab-TeX.html
  - http://mathjax.github.io/MathJax-RespEq/Semantics-Lab/Semantics-Lab-TeX-linebreaking.html
- 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/