

MathML/XML series update

Some approaches to mathematical blogging

Peter James Rowlett
School of Science and Technology
Nottingham Trent University
peter.rowlett@ntu.ac.uk

The blog created by this work is Accessing Mathematics, based on the experiences of Emma Rowlett as a student with a visual impairment studying university-level mathematics for the first time. Read it at <http://maths.accessingmaterials.org.uk/> and read Emma's article on the same subject on page 27. The blogging system adapted for this purpose is part of pCMS by Rhodes & Rowlett.



For a recent project, it was necessary to place a blog on a website into which can be placed mathematical content. The blog author is familiar with LaTeX (or at least is already having to learn) so a solution based on that language would be most appropriate. A suitable output format for online mathematics is MathML.

Wordpress

The Wordpress blogging software [1] supports LaTeX in posts [2], delimited by dollar symbols, i.e. $LaTeX\ code$. However, this only provides conversion to images of the mathematical content, which is not regarded as acceptable. This is a real shame, since the existence of such a feature as standard in this blogging engine clearly shows a need for mathematics-enabled blogging (as a quick look at the comments at the original announcement of this feature [3] will attest) and a MathML output format would be ideal.

itex2MML

A mathematical blogging solution is provided by Distler [4] through the itex2MML plugin for the Moveable Type blogging software [5]. (In fact, itex2MML is a binary and the download also includes Ruby bindings, so this could be integrated into other solutions also).

Basically, Movable Type is a fully-featured blogging engine and the itex2MML plugin allows content to be entered with itex code between $...$ being translated into MathML in the final output.

itex is a LaTeX-like language designed to be closer to the structure of MathML than LaTeX itself [6]. Movable Type is free for personal, non-commercial use and is relatively easy to install provided you have a server that meets the minimum requirements. The itex2MML plugin to Movable Type relies on the itex2MML binary to operate. Distler [7] has a comprehensive piece on adapting Movable Type to ensure MathML posts can be viewed by the full range of MathML enabled browsers.

Unfortunately, access was not available to configure the server on which the blog was required. While Movable Type was successfully installed, the server could not be convinced to execute the itex2MML binary in its current configuration. Still, this looks to be a very usable solution for anyone with less restrictive access to their server.

Other server-side LaTeX rendering

There are several LaTeX-to-web renderers which could be integrated into a blogging solution. Unfortunately most of these require the server to have LaTeX installed and most return an image rather than MathML. MathTran [8] looks interesting as a LaTeX to MathML converter but again requires a TeX-enabled server, which is not available in this case.

For those without LaTeX-enabled servers, generous individuals offer publicly available servers to which a LaTeX expression can be passed and an image containing that expression is returned; some are detailed by Mayer [9]. Of course you are relying on another server but this may be a suitable solution in some cases, particularly if a server can be found which parses LaTeX and returns a MathML snippet.

Client-side LaTeX to MathML rendering

LaTeXMathML [10] is a JavaScript LaTeX to MathML converter based on ASCIIMathML. (ASCIIMathML was featured in a previous instalment of the MathML / XML Series [11]). This is remarkably easy to set up and basically converts any content on the page found between $...$ from LaTeX to MathML. This is not a perfect solution for displaying mathematics online as it requires that users are viewing the page from a JavaScript enabled user agent but may be a good solution for some purposes. Woodall [10] outlines some differences between standard LaTeX and the notation acceptable to LaTeXMathML.

In this case, a system was developed which took dollar delimited LaTeX content in a textarea (see Fig 1) and then displayed this as a “preview” mode back to the user using LaTeXMathML (see Fig 2). JavaScript was then used to copy the processed content (i.e., including the MathML) to a hidden form variable in the page, so when the “preview” was accepted the server received the MathML content through the form’s POST data. Then this code could be stored as the blog content (see Fig 3). Since the system setup of the blog author is a known variable, it is acceptable to insist on JavaScript for that user; those viewing the blog will simply see the MathML content.

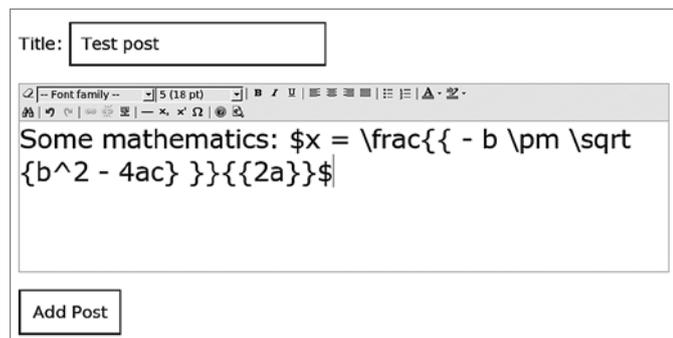


Fig 1 – LaTeX content is entered between dollar symbols in the TinyMCE Rich Text Editor

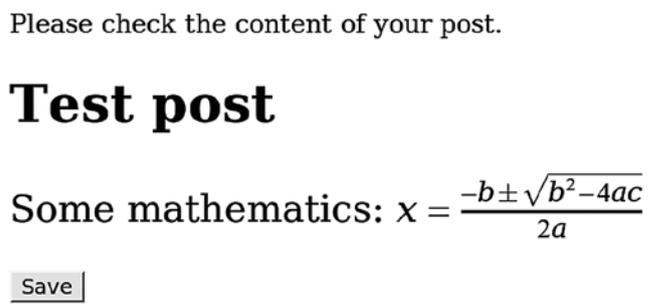


Fig 2 – In Preview mode, the LaTeX content is converted to MathML using LaTeXMathML

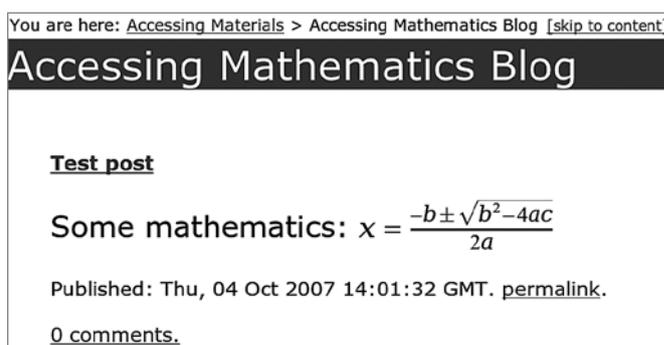


Fig 3 – The output from LaTeXMathML is stored as the blog post, so blog readers see only normal MathML

This process was not quite as straightforward as it sounds. In LaTeXMathML, Internet Explorer (IE) users are presented with different code to those with other browsers, in order to accommodate MathPlayer. The code that is desired to be stored is that served to non-IE browsers, since this is more generic MathML content and if sent in the page content (rather than generated after page loading through JavaScript) would activate the MathPlayer plugin. In order to make the preview mode work with IE, it was necessary to parse two separate regions of the page – the first for display in a manner suitable for which browser was being used, and the second as a hidden div area (“display:none” with CSS) rendered always as if the browser were not IE. It is then this second div area which is used to pass to the database. These involved relatively simple adaptations to the LaTeXMathML code. The author is happy to be contacted for more details.

This solution integrated well as a preview step into a primitive blog engine created in PHP as part of another project which incorporated TinyMCE [12], a JavaScript Rich Text Editor. LaTeX code was embedded in the content authored in TinyMCE.

This method is not perfect but would seem to be suitable for anyone wanting to have a blog which allows mathematical content who does not have access to the server configuration. It is relatively simple and versatile.

Non-LaTeX, non-MathML methods

Of course, there are many other solutions for creating MathML content, many of which will be more appropriate than LaTeX to MathML conversion. LaTeX and MathML have some differences in approach which are difficult

to consolidate, a problem itex tries to overcome. All of the solutions here output Presentation MathML, due to the limitations of LaTeX, where Content MathML may be preferred. There are also other ways to present mathematical content than MathML and these might be more suitable for some situations.

In this case, where the blog required few, occasional mathematical snippets from an author already interested in learning LaTeX, hosted on a server whose configuration could not be altered greatly, this solution seemed most appropriate.

Other than blogging

Of course, the method outlined above basically converts some content posted through a HTML form to change dollar delimited LaTeX code to MathML, could be applied to many situations other than just blogging. Any situation in which content is entered into a web interface could be adapted to use this method. The same method could be applied to a Content Management System for editing MathML content on a website, for instance.

References

1. Wordpress: <http://wordpress.com/> [Accessed 11 Oct 2007].
2. Wordpress. *Can I put Math or Equations in my Posts?* [online] (2007). Available via: <http://faq.wordpress.com/2007/02/18/can-i-put-math-or-equations-in-my-posts/> [Accessed 4 Oct 2007].
3. Wordpress. *Math for the Masses* [online] (2007). Available via: <http://wordpress.com/blog/2007/02/17/math-for-the-masses/> [Accessed 4 Oct 2007].
4. Distler, J. *itex2MML* [online] (2007). Available via: <http://golem.ph.utexas.edu/~distler/blog/itex2MML.html> [Accessed 4 Oct 2007].
5. Movable Type: <http://www.movabletype.org/> [Accessed 11 Oct 2007].
6. Gartside, P. *An introduction to Itex* [online] (2001). Available via: <http://pear.math.pitt.edu/mathzilla/itex2mmltex.html> [Accessed 4 Oct 2007].
7. Distler, J. *Serve it Up!* [online] (2003), Musings. Available via: <http://golem.ph.utexas.edu/~distler/blog/archives/000167.html> [Accessed 4 Oct 2007].
8. MathTran: <http://www.mathtran.org/> [Accessed 11 Oct 2007].
9. Mayer, S. *Online LaTeX* [online] (2007). Available via: <http://www.sixthform.info/steve/wordpress/?p=59> [Accessed 4 Oct 2007].
10. Woodall, D.R. *LaTeXMathML: translating LaTeX math notation dynamically to Presentation MathML* [online]. Available via: <http://www.maths.nottingham.ac.uk/personal/drw/lm.html> [Accessed 11 Oct 2007].
11. Gray, J. 'ASCIIMathML: now everyone can type MathML' (2007), MSOR Connections Aug 2007, Vol 7 No 3. Available via: http://mathstore.ac.uk/newsletter/aug2007/pdf/26_gray_j_asciimathml.pdf [Accessed 11 Oct 2007].
12. TinyMCE: <http://tinymce.moxiecode.com/> [Accessed 11 Oct 2007].