PUBLISHING MATH ON THE WEB – WHAT ARE THE BEST SOLUTIONS?

Bob Mathews
Design Science, Inc. 4028 Broadway, Long Beach CA 90803, bobm@dessci.com

Abstract. "The problem of communicating math on the web is really no different than communicating math via other media. Namely, authoring and displaying mathematical notation is difficult." (Miner and Topping[1]) The degree of this difficulty is decreasing almost daily, but still there is a great deal of planning that must precede the launch of a mathematical web site. As important as this planning phase is, it is equally important that the author employ the right tools and methods. This paper will focus on both the planning phase and the execution phase, while emphasizing the importance of a standards-based approach.

Introduction. When launching a new web site that will contain mathematical content, the web author must consider several questions.

- How much time do I want to invest?
- To what degree do I want to involve myself in the coding of the document?
- How much control do I want (or need) over the final output?
- How important is the appearance of the document when displayed in the browser?
- Should the math "do something" (i.e., dynamic), or just "sit there" (i.e., static)?

The answers to these questions will not only determine the direction to take, but also the most appropriate technology to use.

Dynamic Math. The term "dynamic math" simply means equations that interact with the reader in some way. An author may create a dynamic equation that will display a message in the browser's status bar when the reader points to the equation with the mouse pointer. Other actions include, but are not limited to, "mouseover" highlights, hyperlinks, or equations that expand to reveal subsequent steps when the reader clicks on an ellipsis. Also possible with additional scripting are equations that accept a response from the reader, equations that update when readers change values in a form, or equations that update in real time by other active components.

Many technologies; many solutions. As you plan a new math on the web project, you must choose from a variety of technologies, each with its own unique advantages and disadvantages.

- HTML pages with GIF images for equations
- Adobe Acrobat (PDF)
- IBM techexplorer
- Server-side programming (CGI, Perl, Java, ASP scripts, etc.)
- HTML + Components (Design Science WebEQ, IBM techexplorer)

(See Miner and Topping[2] for a comparison of these technologies, including advantages and disadvantages.)

The Universal Appeal of a Standards-based Approach. None of the technologies listed above is the single, ideal solution to a WebMath project. Since no single approach provides a complete solution, the answer is clearly standards. "MathML is intended to facilitate the use and re-use of mathematical and scientific content on the web, and for other applications such as computer algebra systems, print typesetting, and voice synthesis. MathML can be used to encode both the presentation of mathematical notation for high-quality visual display, and mathematical content, for applications where the semantics plays more of a key role such as scientific software or voice synthesis." (World Wide Web Consortium[3] – Also see this reference for MathML examples and a listing of products supporting MathML.)

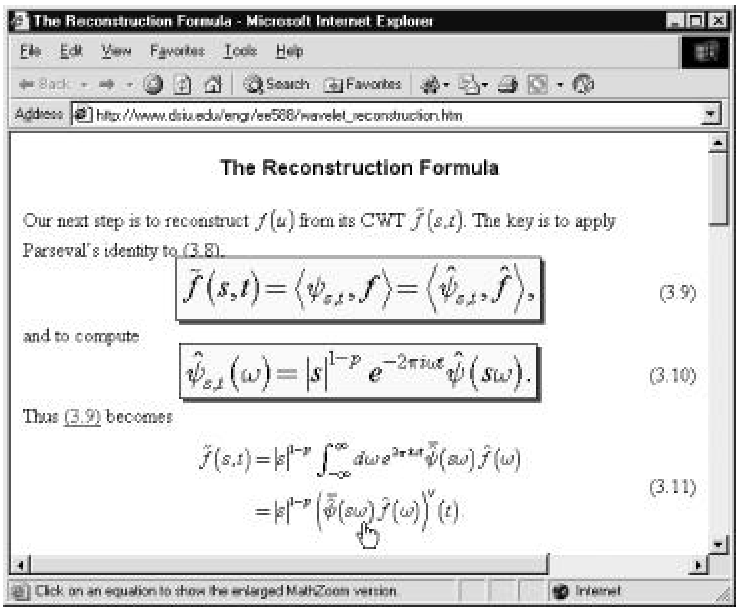
Within the MathML standard are two distinct styles used for encoding expressions – presentation encoding (or presentation markup) and content encoding (or content markup). Which kind of markup is most appropriate for a given task will depend on the situation. Using presentation elements, you can precisely control how an expression will look when displayed in a browser, or printed on paper. Unfortunately, as with any layout-based markup language, it is all too easy to get it to look right, without taking care to get the underlying structure right. In some cases this will not matter, but it is less likely a badly encoded expression could be spoken properly by a voice synthesizer, evaluated in a computer algebra system, or used by other applications which need to know something of the sense of an expression, rather than just its appearance. (Miner and Schaeffer[4])

Content markup is intended for facilitating applications other than display, like computer algebra and speech synthesis. Consequently, when using content markup it is harder to directly control how an expression will be displayed.

MathType 5. MathType 5 for Windows incorporates MathPage technology to produce web pages from Microsoft Word documents with stunningly beautiful results. When using Word's "Save as Web Page" or "Save as HTML" commands, you get a nice-looking web page – unless the page contains math! MathPage corrects all the appearance glitches introduced by Word, and has several impressive features not available elsewhere:

- Inline equations are baselined properly.
- Small details are easy to read.
- Display equations are centered properly.
- MathType equation numbers are properly placed and references become hyperlinks
- Web pages print with word processor-quality, without "jaggies" that normally characterize GIF equations.

Simply by clicking the Export to MathPage icon in Word and reviewing the choices on the ensuing dialog, you can create a web page such as the one in Figure 1. Note that all the items listed above are solved. When the recipient views the page in the browser, simply clicking on any equation zooms the image 150% to allow small details to be read easily. In addition, the page prints directly from the browser with word processor-quality, and the equations can be opened in MathType for use in other documents.


The Reconstruction Formula - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.dsiu.edu/engr/ee588/wavelet_reconstruction.htm
Click on an equation to show the enlarged MathZoom version
Internet

The Reconstruction Formula

Our next step is to reconstruct $f(u)$ from its CWT $\tilde{f}(s,t)$. The key is to apply Parseval's identity to (3.8).

$$\boxed{\tilde{f}(s,t)=\langle\psi_{s,t},f\rangle=\langle\hat{\psi}_{s,t},\hat{f}\rangle,} \qquad (3.9)$$

and to compute

$$\boxed{\hat{\psi}_{s,t}(\omega)=|s|^{1-p}e^{-2\pi i\omega t}\hat{\psi}(s\omega).} \qquad (3.10)$$

Thus (3.9) becomes

$$\begin{aligned}\tilde{f}(s,t)&=|s|^{1-p}\int_{-\infty}^{\infty}d\omega\, e^{2\pi i\omega t}\bar{\hat{\psi}}(s\omega)\hat{f}(\omega)\\&=|s|^{1-p}\left(\bar{\hat{\psi}}(s\omega)\hat{f}(\omega)\right)^{\vee}(t)\end{aligned} \qquad (3.11)$$

Figure 1. MathPage example showing MathZoom and a hyperlinked reference

WebEQ 3.0 Developers Suite. WebEQ Developers Suite is a collection of tools for putting dynamic math on the web. The Developers Suite contains three programs, two applets, templates, sample scripts and extensive documentation. By using the programs and applets in different combinations, it is possible to create a wide variety of math and science documents and interactive web sites.

The individual programs and applets that make up the Developers Suite are:

WebEQ Editor. WebEQ Editor is an equation editor that is used to graphically compose equations. By using templates and symbol palettes from a customizable toolbar, you build up an equation using the mouse and the keyboard in an intuitive way. WebEQ Editor is specifically designed for working with equations encoded in MathML. WebEQ Editor can save equations as MathML markup, as images, or as applet tags for displaying the equations in a browser using the WebEQ Viewer Control.

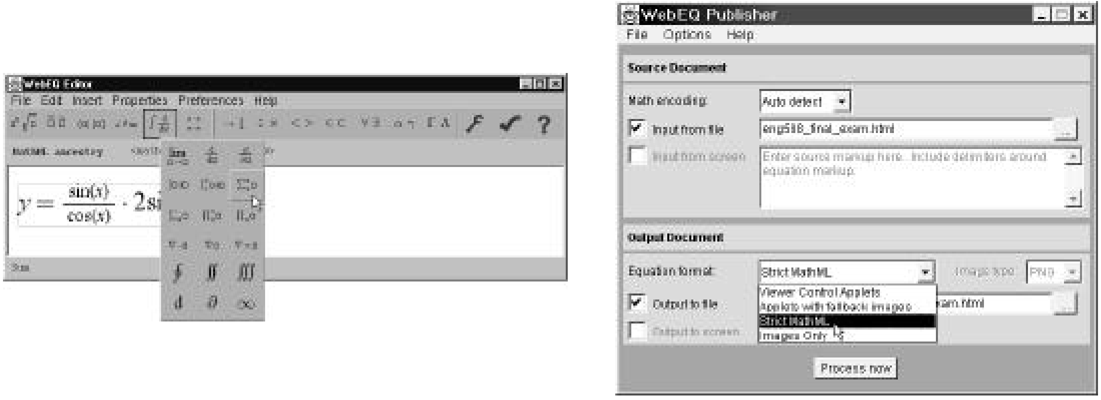
WebEQ Publisher. WebEQ Publisher converts equations and documents into a variety of formats. It is most often used to process an entire HTML page containing many equations, though it can also be used to convert one equation at a time. The Publisher reads in both MathML and WebTeX (a subset of the

L^{A}T$_{E}$X math commands adapted for the web), and it generates the same formats as the Editor – MathML, images, and applet tags.

WebEQ Equation Server. Equation Server is also a program for processing documents and converting between equation formats. It is typically used by developers of interactive math and science web sites to generate images of equations on the fly from scripts. It is very similar to WebEQ Publisher in terms of functionality, but with a command-line user interface, for use from scripting languages.

WebEQ Viewer Control. The WebEQ Viewer Control is a scriptable applet for displaying dynamic math expressions in web browsers.

WebEQ Input Control. The WebEQ Input Control is an equation editor applet that runs in a web page. It gives readers a way to enter mathematical expressions into a web page, just as HTML forms give users a way to enter other kinds of information. The Input Control is not very useful by itself; generally some programming is necessary to do something with the input readers enter.



$y = \frac{\sin(x)}{\cos(x)} \cdot 2\mathrm{si}$

Math encoding: Auto detect

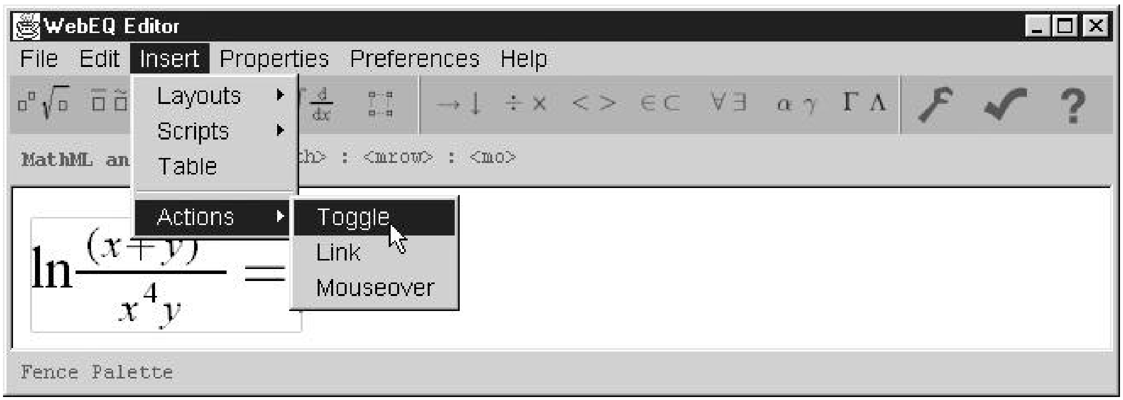
☑ Input from file: eng518_final_exam.html

Equation format: Strict MathML

☑ Output to file: ...am.html

Figure 2. The two visible elements of WebEQ – the Editor and the Publisher

Dynamic math with WebEQ. WebEQ can be scripted to take advantage of many scenarios involving dynamic math. However, no advanced programming skill is required to create any of three dynamic math "actions": a toggle, a hyperlink, or a mouseover. A toggle can be created that will switch between two displays when the reader clicks on part of an equation. A use for a toggle is an equation that expands when the reader clicks on an ellipsis. A hyperlink can take the reader to another page or program, and a mouseover can highlight part of an equation. Messages can be displayed in the browser's status bar with all three actions.



$\ln\frac{(x+y)}{x^4 y} =$

Figure 3. Creating dynamic web math with WebEQ 3.0

Displaying MathML. Much of the preceding discussion has dealt with tools for authoring MathML. As of this writing, neither of the two mainstream browsers (Internet Explorer 6.0 or Netscape 6.2.3) will display MathML natively (i.e., without a display engine or plug-in). There is a great deal of promise on the horizon with [5]Mozilla 1.0 and [6]Netscape 7.0 Preview Release 1, as both of these browsers display MathML presentation markup natively on the Windows platform. So what does one do if Internet

Explorer is the browser of choice? How about if you are a professor and your students may be using either browser? Fortunately, there is an easy answer.

The *Universal MathML Style Sheet*[7] (UMSS) makes it possible to view MathML on a large number of browsers, and using one of several plug-ins or display engines if available. Normally the stylesheet will transform the input document to use a suitable MathML renderer, depending on the browser in use and any browser extensions it detects. It tries a list of possibilities and selects the first one which appears to be installed. If none are detected, the stylesheet will display the math without a renderer. A web page author can easily instruct the UMSS to use a particular renderer (such as Design Science MathPlayer or IBM techexplorer) as the primary display engine. If the specified display engine is present, everything proceeds normally. If it is not present, the style sheet attempts to choose one from its list.

Design Science MathPlayer[8] is a high-performance MathML display engine for Internet Explorer for Windows version 5.5 and later, and available as a free download. It is possible for the author to directly use markup that will cause the browser to render the MathML with MathPlayer. It is also possible to insert a tag that will invoke the UMSS, but will specify MathPlayer to be used if present. Other advantages of MathPlayer are that it allows the user to copy the MathML for pasting into another application, such as MathType or WebEQ. A context menu enables passing the MathML directly to applications such as a computer algebra system or graphing utility.

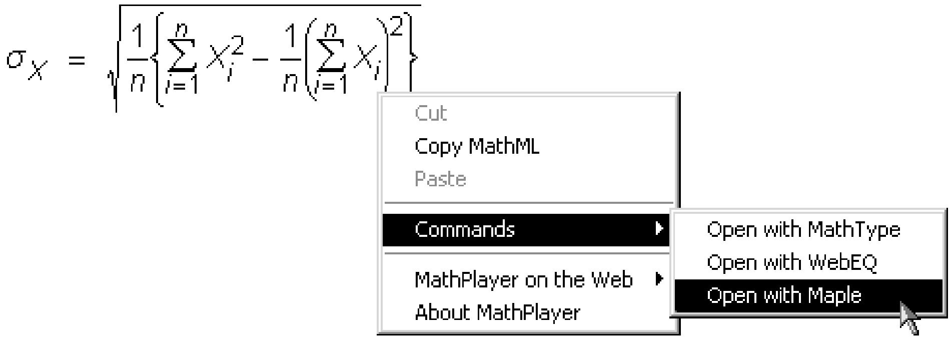

$\sigma_X = \sqrt{\frac{1}{n}\left\{\sum_{i=1}^{n} X_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} X_i\right)^2\right\}}$
Cut
Copy MathML
Paste
Commands
MathPlayer on the Web
About MathPlayer
Open with MathType
Open with WebEQ
Open with Maple

Figure 4. Display MathML in Internet Explorer with Design Science MathPlayer

Summary. When beginning your WebMath project, you need to decide how much time you're willing to spend and how much control you need over the appearance of the page. You need to decide whether static math is satisfactory, or if you need dynamic math. Finally, you need to decide between proprietary or standards-based technology. If you go the standards-based route, you have a choice of authoring and display tools. Consider using the Universal MathML Style Sheet, with Design Science MathPlayer as the primary display engine. If you don't need dynamic math, and print quality and ease of use are important, consider the Microsoft Word/MathType/MathPage combination.

This handout available on the web, with live hyperlinks and larger screenshots & examples, at http://www.dessci.com/sales/training

References.

1. Miner, R., and Topping, P. (July 2001). *Math on the Web: A Status Report – Focus: Distance Learning*. http://www.dessci.com/webmath/status/status_Jul_01.stm
2. Miner, R., and Topping, P. (January 2001). *Math on the Web: A Status Report – Focus: Standards-based Math on the Web*. http://www.dessci.com/webmath/status/status_Jan_01.stm
3. W3C's Math Home Page. http://www.w3.org/Math
4. Miner, R., and Schaeffer, J. (October 2001). *A Gentle Introduction to MathML*. http://www.dessci.com/support/tutorials/mathml/default.stm
5. http://www.mozilla.org
6. http://channels.netscape.com/ns/browsers/7/
7. http://www.w3.org/Math/XSL/
8. http://www.dessci.com/webmath/mathplayer/