6. Analysis and Discussion

The CourseEditor web authoring tool was successfully implemented and tested for creating online course materials. At the beginning of the research several sub problems were identified and it was able to address them all during the design and implementation process.

6.1. Cross Browser Support for MathML

The generated web pages with MathML content were tested on the following browsers and those browsers were able to successfully render the mathematical content.

- Mozilla FireFox 2.0 and above in Windows environment
- Mozilla FireFox 1.5 on Linux environment
- IE 6.0 and above in Windows environment with MathPlayer 2.1 plug-in
- IE 7.0 in Windows environment with MathPlayer 2.1 Plug-in
- Netscape 7.0 in Windows environment

Although Mozilla has the native support for MathML, it will not render certain mathematical symbols correctly (e.g. square root symbol). It requires additional math fonts [13] to be installed in the client machine for proper display of mathematical content. IE 6.0 and 7.0 render the MathML with the use of MathPlayer plug-in. MathPlayer has few additional features other than rendering MathML. It enables Math zoom and have the accessibility feature which enables to speak equations through the computer's sound system.

The generated web pages properly render both the inline mathematical contents and the mathematical paragraphs (See Appendix B-Figure B-1). Inline MathML does not disturb the line spacing and the alignment of the paragraph. MathML Tester by Mozilla [14] is used for verifying the generated MathML content.
The main disadvantage of using a third party extension to render MathML within a web page is that it requires specific markup to specify the rendering extension (such as `<applet>` for Java applets, `<embed>` for plug-in and `<object>` for Microsoft Behavior extensions). The use of such markup ties the document to one particular platform, whereas the ideal of publishing information on the web is that it should be accessible to all using a range of tools. CourseEditor follows the W3C proposed mechanism for using the XSLT transformation language to avoid these drawbacks and to allow XML files that conform to the XHTML and MathML Recommendations to be rendered on a range of browsers, with no browser specific markup in the document.

By default, IE does not allow XSLT style sheets to be applied to a document if it is not located on the same server as the document. Instead it is necessary to copy the style sheets to the same server as the document. Alternatively it is possible to change the security options in the browser which is not advisable.

6.2. Web Content Creation using CourseEditor

There are quite a number of interactive HTML authoring tools existing today to develop custom material for the web starting from scratch. CourseEditor is not intended to be a replacement for these tools. Instead, it complements HTML editors by offering rapid, fully automatic conversion from MS Word documents to HTML format. The content generated from this tool is far different from the web page created by MS Word's "Save as HTML" feature. This option in MS Word produces very dense Microsoft-specific HTML code that often needs to be cleaned up. Proposed CourseEditor bypass this completely and generates simple, clean XML and HTML code that is validated against W3C standards (Appendix B – Figure B-2).

MS Word’s “Save as HTML” feature converts the MathML content into images. Those images in those documents will not be viewable in browsers other than IE. Even the IE will not display these images properly with the alignments specified in
the MS Word document. All the inline mathematical content gets right aligned in
the web page.

CourseEditor will automatically break a long document into separate pages based
on the headings in the document. This feature becomes handy for document
management purposes, unlike converting multiple MS Word documents for each
web page in the web site. The CourseEditor will link the web pages together with
graphical or text-based navigation links so the user can navigate through the pages.
It will also create an online table of contents. This tool enables the users without
knowledge of HTML, or without the desire to work with dedicated HTML editors,
to easily produce sophisticated web pages from their MS Word documents.

6.3. System Design Considerations

CourseEditor system is designed in a way that it’s possible to plug new modules
easily. New modules like self test questions, flash cards are intended to include in
the system in future. Custom event handlers are used to segregate the business logic
from the presentation logic of this tool making room for easy extendibility.

Macro programs along with the Word Automation techniques can be used to
extract the content from the MS Word 2007 documents. The main disadvantage of
applying this approach is the security concerns over macros. CourseEditor uses
.NET Framework 3.0 redistribution package to extract the content from Word 2007
documents.

CourseEditor allows a high degree of configurability for the generated web content
through the CSS files. CSS is used to help authors of course content to define
colors, fonts, layout, and other aspects of web page presentation. CSS allows the
separation of document content (written in MS Word) from web presentation.
CourseEditor is developed basically using .NET framework 3.0 and C#. Unlike .NET 2.0, .NET 3.0 has support for Open Packaging Conventions (package handling and navigating the relationships) used by MS Office 2007 applications. MS Office 2007 file is essentially a ZIP archive containing many other files. Office-specific data is stored in multiple XML files inside that archive. This is in direct contrast with old WordML formats (used in Word 2003 files) which were single, non-compressed XML files. Although more complex, new approach offers more benefits. CourseEditor can be deployed on the MS Windows versions including Windows 2000, Windows 2003, Windows XP and Vista which has .NET framework 3.0.

6.4. Testing MathML and SCOs

MathML tester page by Mozilla [14] used to test the MathML fragments generated by CourseEditor. It renders the MathML fragments on the fly complying with the W3C MathML standards insuring uniformity and interoperability between MathML implementations.

Since performance is not a critical concern, a performance test was not carried out. The standard design implementation best practices have been followed and it does not contain any bottlenecks which might cause performance degrades. But the generated MathML content and the SCO(s) are successfully tested against the relevant conformance suites as described in section 5.5. The SCO test suite includes a step-by-step set of detailed instructions as well as a log of each test action and the outcome. This test suite runs within a Web browser on one local computer. All of the software is loaded on the machine and requires no HTTP Web server to run.