WEB BASED MATHML EDITOR
WYSIWYG EQUATION EDITOR

T.Bharathi Priyaa

A dissertation submitted to the

FACULTY OF SCIENCE AND HUMANITIES

in partial fulfillment of the requirements

for the award of the degree of

MASTER OF SCIENCE (INTEGRATED)

in

COMPUTER SCIENCE

ANNA UNIVERSITY CHENNAI

CHENNAI –600025

NOVEMBER 2008

BONAFIDE CERTIFICATE

Certified that this dissertation titled "**Web Based MathML editor-WYSIWYG Equation Editor**" is the bonafide work of **Ms.T.Bharathi Priyaa** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

(Dr. P.VIJAYARAJU)
Professor and Head
Department of Mathematics
Anna University
Chennai – 600 025.

(Dr.R.COKILAVANY)
Lecturer
Department of Mathematics
Anna University
Chennai – 600025.

ABSTRACT

A Web Based MathML editor is developed with an aim to put mathematics in Web. The intended users of the system include teachers, publishers, professors, students etc with a need to put scientific content on web.

The motivation for MathML is a world in which individuals and companies want to move away from proprietary data formats and use XML to express, interchange, and share information.

According to the W3C*, "The goal of MathML is to enable mathematics to be served, received, and processed on the World Wide Web, just as HTML has enabled this functionality for text."*

we propose to develop a web based mathematical editor, which can create and handle equations of various types. It provides a facility for publishers, professors, and students to create or edit the mathematical equations with the symbols provided.

¾¢ð¼ôÀ½¢ ÍÕì¸õ

¸½¢¾ Å¨Ä ÀÂ½£ð¼¡÷¸Ç¢ý ÀÂñÀ¡ðÊü¸¡¸ ¯ÕÅ¡ì¸ôÀð¼ ¦Áý¦À¡Õû ¬Ìõ.¸½¢¾¾ì ÌÈ£ðÎ ¦Á¡Æ¢ «ÊôÀ¨¼Â¢ø À¾¢ôÀ¡ý ¦ÀÕìÌÅ¾üÌ ,¸½¢¾ Å¨ÄÂ¢ø §º÷ôÀ¾üÌ þôÀ¾¢ôÀ¡ý ¯ÕÅ¡ì¸ôÀðÎûÇÐ.

þùÅÈ¢Å¢Âø ¯ûÇ¼ì¸ò¨¾ Å¨ÄÂ¢ø «¨ÁôÀ¨¾ , ¬º¢Ã¢Â÷ , À¾¢ôÀ¡Ç÷¸û ,Å¢Ã¢×¨ÃÂ¡Ç÷¸û ,Á¡½Å÷¸û ÁüÚõ ÀÄ÷ ÀÂýÀÎòÐÅ¡÷¸û.

¸½¢¾ì ÌÈ¢Â£ðÎ ¦Á¡Æ¢Â¢ý ¦¿¡ì¸õ ±ýÉ¦ÅÉ¢ø ¯Ä¸õ ÓØÅÐõ ,¾É¢¿À¦Ã¡ ¸õÀÉ¢¸¦Ç¡ Üð¼¨Áô¦À¡ ¦º¡ó¾Á¡¸ ¾Ã× ¯¨ÃÅÊÅò¾¢Ä¢ÕóÐ ¦ÅÇ¢§ÂÈ¢ , ¯¼ý ¦¾Ç¢×ÀÎò¾ ÀÂýÀÎò¾¢ ¸½¢¾ ¾¸Åø¸¨Ç ÀÃ¢Á¡üÈõ ¦ºöÂ ,¸½¢¾ ¾¸Åø¸¨Ç ÀíÌ¦¸¡ûÇ ¬Ìõ.

þ¾¨É ¿¡í¸û ¾¢ð¼Á¢ðÎ Å¨Ä «ÊôÀ¨¼Â¢ø ¸½¢¾ôÀ¾¢ôÀ¢ ¯ÕÅ¡ì¸¢¦É¡õ.«¨Å ÀÄ Å¨¸ ºÁýÀ¡Î¸¨Ç ÀÂýÀÎò¾×õ ¯ÕÅì¸×õ þôÀ¾¢ôÀ¢¨Â ÀÂýÀÎò¾Ä¡õ.

ACKNOWLEDGEMENT

I would like to convey my gratitude to **Dr. P. Vijayaraju** ,Professor and Head, Department of Mathematics, Anna University for his constant encouragement and inspiration to me during the course.

I am indebted to **Mr. Anjan T,** Front End Engineer, HeyMath! who has guided me with sustained interest and meticulous encouragement without which the completion of this project would not have been possible.

I owe my deep sense of gratitude to **Mr.Anantharaman Mani, Chief Technological Office**, Heymath! who gave me this golden opportunity to work in this project and inspired me always to do the best.

I would like to show my sincere gratitude to my class advisor **Dr.D.ArivudaiNambi,Assistant Professor**, for his constant encouragement.

I also would like to take advantage of this opportunity to express my sincere gratitude to **Dr. R.Cokilavany,** Lecturer, Department of Mathematics, Anna University Chennai for his guidance in completing this project successfully.

And last, but not the least, we wish to thank our families and friends for bearing with us throughout the project period and for having created the opportunity to do this course in such a prestigious institution.

Chennai-25
November 2008 **(T.BHARATHI PRIYAA)**

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 ORGANIZATION PROFILE

Sankhyaa Learning (P) Ltd is an E-Learning Company and operates under the brand name HeyMath! This company was developed with the vision to support the work of teachers and to help students build a strong foundation in Math.

Developed in formal collaboration with the University of Cambridge, HeyMath! provides middle and high schools with technology-enabled and curriculum-oriented animated online resources that help remove the abstraction underlying Math concepts, strengthen student understanding, enhance problem-solving skills and make learning more interesting and interactive.

Teachers save instruction time and achieve differentiated instruction by using HeyMath! lessons in classrooms. HeyMath! also provides schools with access to a growing question-bank (contributed by HeyMath! partner schools), comprising 15,000+ questions, that have been categorised by grade & topic, for ongoing practice, testing and assessment. Implemented at several progressive schools in Singapore, India and the United States, HeyMath! resources reflect a blend of 'best practices' from top schools and are constantly updated based on user feedback.

1.2 MATH-ML EDITOR TOOL OVERVIEW

MathML WYSIWYG Editor is an equation editor component designed using the Flex Framework for all users ranging from students and teachers to the high-end science and technical publishers. MathML .Editor provides a very easy user interface allowing to create every imaginable form of mathematical expressions.The Editor combines the advantageous features of MathML and flash and is designed to be more intuitive and predictive . The Easy-to-use graphical interface enables the user to choose from over 300 mathematical symbols. The editor also includes a Help Assistant which generates suggestions and information messages to the user when he creates the equation.

1.3 MODULES

1.3.1 **WYSIWYG Editor** An easy to use graphical interface provided with an advanced Math Palette which enables user to create complex mathematical equation

1.3.2 **MathML Equation Parser and Renderer** : The renderer parses the MathML and renders the graphical display of equation. This refers to the process of converting the MathML code to a graphical display of Math Formula.

1.3.3 **Intuitive Help Assistant** : The MathML editor is an Intuitive editor which guides the user through his equation creation process. To make it intuitive we need a help assistant which triggers suggestions and information to the user when he creates the Equation.

1.3.4 **Equation Editing** :The MathML code is parsed to render an editable graphical form of Mathematical equation.The equation is then available to the user for “Point and Click Editing”. The Math ML code is parsed as an XML ,and each of the element equation is rendered visually.

1.3.5 **Server Support And Usability Features** : The user creates an Equation for the First time using *WYSIWYG editor*. The user then saves the equation with a particular Equation title in the database.The Equation title along with the generated MathML is stored in the database. The usability features include

- MathML import/export
- Save (and reopen) equations as Jpeg images
- Upload /Download MathML/Images

1.4 ORGANIZATION OF THE REPORT

The First Chapter gives a corporate profile of Heymath, Alwarpet, where the project has been done. Also it gives the outline of the overall project.

The following chapter gives a brief outline of the Mathematical MarkUp language and the existing system of editors which use and manipulate MathML to create and edit mathematical equations.

The System design alternatives and decisions are discussed in Chapter 3.

Chapter 4 discussed about the development of MathML editor in Flex in detail. It also explains the various implementation frameworks of Equation creator and Editor.

The various levels of testing that the system underwent is being discussed in Chapter 5..

Chapter 6 summarizes the whole project and points out the assets of the project.

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

The Planning and Requirements phase of the project includes the following activities

- Requirements Specification .
- Study of MathML .
- History And Need for MathML.
- Intended Users and Benefits of MathML.
- Types of MathML.
- Benefits of MathML.
- Research and Analysis on Existing Mathematical templates..

.

2.2 REQUIREMENTS SPECIFICATION

2.2.1 HARDWARE SPECIFICATION

❖ Processor	:	Intel Pentium IV Processor
❖ RAM	:	128 MB RAM 64 MB RAM
❖ Hard Disk	:	20 GB
❖ Monitor	:	17 inches
❖ Speed	:	1.14 GHz

2.3 SOFTWARE SPECIFICATION

❖ Operating System	:	All Operating systems
❖ Browser	:	IE 6, *Fire fox 1.x,Netsape 7.x,Opera 6* Safari 1.x.
❖ PlugIn	:	Flash Player 9.0 or above

2.4 STUDY OF MATHML

Mathematical Mark up Language is a set of specifications proposed by the World Wide Web Consortium for the display of mathematical equations on Web.

MathML is an XML specification which enables native rendering of mathematical formulae by the browser.It aims at integrating mathematical formulae into World Wide Web documents. It is a recommendation of the W3C math working group.

MathML is intended to be used by everyone from high-school mathematics students to academics and engineers in industry. It also provides an interchange mechanism between applications processing mathematical representation in some form.

2.4.1 GOALS OF MATHML

- The principal goal of MathML is to enable mathematics to be served, received, and processed on the Web, just as HTML has enabled this functionality for text. In more detail, MathML is intended to:
    - encode mathematical material suitable for teaching and scientific communication at all levels.
    - encode both mathematical notation and mathematical meaning.
    - facilitate conversion to and from other math formats, both presentational and semantic.
- allow the passing of information intended for specific renderers and applications.
- support efficient browsing for lengthy expressions.
- provide for extensibility and to be human legible (though it is very verbose), and simple for software to generate and process.

2.5 HISTORY AND NEED FOR MATHML

Until very recently, the mathematics in web pages was frequently encoded as graphics (a .gif, .jpg or .png image). With MathML, the situation is changing.It is now possible to embed mathematical equations into web wages with minimal of storage requirement and No overhead.

The mark up language HTML has a large variety of tags .But there are no explicit tags for the rendering mathematical expressions. As a result ,to embed math equations, authors have resorted to embedding images of equations. Snap shots of equations taken from other packages and saved in GIF format - into technical documents which have a mathematical or scientific content.

Embedding of images in Web Pages are costly in terms of storage space. To overcome this. The MathML specification is put forth which helps browsers understand ,the Math Mark up language

2.6 TYPES OF MATHML

2.6.1 PRESENTATION MATHML

Presentation MathML focuses on the display of the equation rather on the meaning of the Equation. The Components of Presentation math ml constitutes 30 elements and 50 attributes. Attributes mainly control the fine details of the presentation.

Eg,Line Thickness,rspace,lspace,fontstyle.

2.6.1.1 MATHEMATICAL TYPE SETTING

MathML expressions can be regarded as trees, where each node corresponds to a MathML element, the branches under a "parent" node correspond to its "children", and the leaves in the tree correspond to atomic notation or content units such as numbers, characters, etc.

2.6.1.1.1 Expressions /SubExpressions

The <mrow> element is used to denote a row of horizontally aligned material. The material contained between the <mrow> and </mrow> tags is considered to be an argument to the mrow element. Thus the whole expression here is contained in an mrow element. Almost all mathematical expressions decompose into subexpressions. These subexpressions can, in turn, also be contained in an mrow element.

For example, a + b is also contained in an mrow.

2.6.1.1.2 ***Token elements:*** Token elements represent the most basic structures in mathematics.

For example: ***<mi>:*** Identifier, such as a variable, function name, constant

```
 <mi> x </mi>
<mi> D </mi>
<mi> sin </mi>
<mi mathvariant='script'> L </mi>
<mi></mi>
```

Fig 2.6.2.1 Identifer Math element

<mo>: It renders any symbol or notation that should be rendered as an operator.This include fence characters such as braces, parentheses, and "absolute value" bars, separators such as comma and semicolon, and mathematical accents such as a bar or tilde over a symbol.

```
<mo> + </mo>
<mo> &lt; </mo>
<mo> &le; </mo>
<mo> &lt;= </mo>
```

Fig 2.6.2.2 Operator Math Element

<mn>: It renders a numeric literal which can include a sequence of digits(With or without a decimal point.).

```
<mn> 2 </mn>
<mn> 0.123 </mn>
<mn> 1,000,000 </mn>
<mn> 2.1e10 </mn>
<mn> 0xFFEF </mn>
<mn> MCMLXIX </mn>
<mn> twenty one </mn>
```

Fig 2.6.2.3 Numeric Literal Math Element

2.6.1.1.3 *General layout:*

General layout elements describe the nature of the layout.

For example:

<mfrac>: Form a fraction from two sub expressions

The mfrac element is used for fractions. The parent mfrac tag has two children, representing the numerator and denominator respectively. A Fraction can take the following attributes like, *line thickness,numalign,denomalign,beveled.*

$$\frac{\binom{a}{b}\quad\frac{a}{b}}{\frac{c}{d}}$$

<msqrt>: Forms a square root sign

The msqrt element is used for square roots, while the mroot element is used to draw radicals with indices, e.g. a cube root. The syntax for these elements is:

```
<msqrt> base </msqrt>
<mroot> base index </mroot>
```

2.6.1.1.3 Radical MathML Structure

2.6.1.1.4 ***Scripts and limits:*** Scripts and embellishments to symbols are common in mathematical notation. The elements in this section address this requirement. For example:

<msup>: Attach a superscript to a base .

Mathematical notation which is used to denote an exponentiation operation between two operands.(Base and exponent.)

<msub>: Attach a subscript to a base .

MathML specifies an <msup> tag to denote the above. <msup> tag is followed by two children .The First child forms the Base.The Second child is the exponent.

2.6.1.1.5 ***Tables:*** Matrices, arrays, and table-like mathematical notations are also represented using tags in MathML. For example:

<mtable>: Table or matrix

<mtr>: Row in a table or matrix

2.6.1.1.6 *.**Actions*** :The maction element is in a category by itself, and allows coding of various kinds of actions on notation, such as occur in an expression which toggles between two pieces of notation.

<maction>: Binds actions to a subexpression.

2.6.2 CONTENT MATHML

Content MathML focuses on the semantic meaning of the equation, wherein there is an <apply>tag which specifies that a particular operator is applied to the succeeding elements. Content markup facilitates applications other than display, like computer algebra and speech synthesis. The scope of content markup includes arithmetic, algebra, logic, relations, set theory, calculus, sequences, series, functions, statistics, linear algebra, and vector calculus.

For Example,

<apply>Integral

Here Integral is applied to all the elements enclosed.

2.7 Benefits of MathML

The introduction of MathML brings several benefits, including:

- **Computation:**

Given that MathML includes tags for the content of the mathematics, it can be used for computation. In fact, major computer algebra systems, such as Mathematica, have begun to support cut-and-paste of MathML into their documents, enabling their users to transfer mathematics from Web pages and other sources into their documents. After the MathML document has been transferred to a mathematical tool, it can be manipulated for calculation, graphing, searching, and analysis.

- **Public standard*:***

The fact that MathML is a public standard paves the way for the creation of tools and solutions from a number of vendors and open-source organizations. It also means that mathematical data can be easily shared among different organizations and in this way facilitates research and information sharing (both of which are the goals of the World Wide Web).

- **XML advantages:**

MathML is based on XML ,so it brings all the advantages of XML with it. It can be used for representation on any application that handles XML, including Web, paper, and PDF. Also, the representation can be changed using a stylesheet document.

2.7.1 Simple example of MathML

Consider the equation

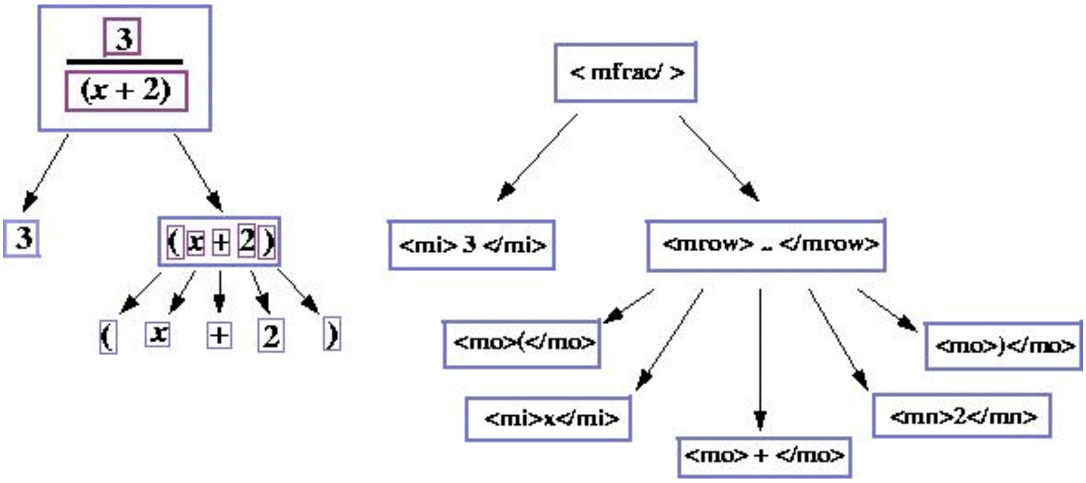
$$\mathbf{x^2 + 4x + 4 =0}$$

Given below are two ways that this can be represented, first using presentational tags, then using semantic tags. The presentational tags generally start with "m" and then use "o" for operator "i" for identifier "n" for number, and so on. The "mrow" tags indicate organization into horizontal groups.

<u>MathML Code:</u>

```
<mrow>
  <mrow>
             <msup>
                    <mi>x</mi>
                   <mn>2</mn>
             </msup>
             <mo>+</mo>
            <mrow>
     <mn>4</mn>
     <mo>&InvisibleTimes;</mo>
     <mi>x</mi>
```

```
</mrow>
<mo>+</mo>
<mn>4</mn>
</mrow>
<mo>=</mo>
<mn>0</mn></mrow>
```



2.9.5 Simple Example :MathML Tree representation

2.8 RESEARCH ON EXISTING MATHEMATICAL TEMPLATES

There are numerous softwares which currently render Mathematics.

2.8.1 Browsers:

- W3C's Amaya browser displays Presentation MathML, and lets a Web page author edit equations directly.
- Firefox 1.0 renders PersentationMathML in HTML pages natively.
- Internet Explorer does not directly support MathML however addons like MathPlayer and TechExplorer extend the functionalities of IE to render mathematics.

2.8.2 Editors:

- Design Science MathType: is an equation editor in MicroSoft Office.
  - Features include,Point and Click Equation Editing,Copy/Paste MathML,Convert Equations to Web Pages.
- sMArTH; It is an online equation editor for MathML and Latex.
  - It is built on Open Web Standards and implements an SVG (Scaleable Vector Graphics)interface.

2.9 FLEX FRAMEWORK:

It provides integrated tools and technology to develop Rich internet application.Flex makes use of Abode flash player thus extending browser capabilities and more responsive client-side applications.Flex applications consist of MXML and Action Script source files.

Mxml: This is xml based language used to present the user layout.

Action script: Similar to javascript .Provides object manipulation that cannot be done with mxml.

2.9.1 BENEFITS OF FLEX FRAMEWORK

- Flex application executes on Flash player which is platform independent.
- Flash player runs consistently in all browsers and hence different client environment.

2.10 Key Features And Benefits:

Based on existing standards	MathML created with our App are valid with the existing W3C recommendations
Platform portability	The MathML app is designed to be Platform Neutral making it easily portable to all the operating systems that currently support Flash.
WYSIWYG-style Equation Editor Features	An Easy-to-use Graphical Interface that covers a wide variety of Math Operators and Symbols(Over 300).
Web Publishing Features	It provides the user with Networking support and Server Support.Docs can be published on the web as it is (with mathematical expressions encoded as MathML).
LateX Export	Feasibility Under Study.

2.11 SUMMARY

The hardware and the software required for the WYSIWYG Editor are analyzed and specified here. A comprehensive report on the various math editors and their user support features also specified here.The next chapter discusses the various modules involved in detail and also explains the various methodologies used in the project.

CHAPTER 3

SYSTEM DESIGN

3.1 WYSIWYG-style Equation Editor

WYSIWYG is an acronym for ***W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et*, used in computing to describe a system in which content displayed during editing appears very similar to the final output, which might be a printed document, web page, slide presentation or even the lighting for a theatrical event.

3.2 Use Case Diagram

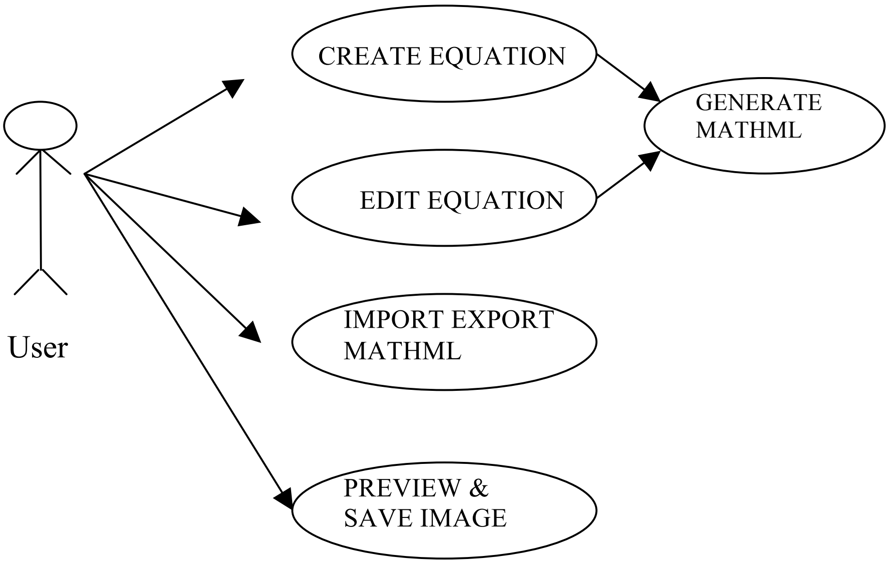


Fig 3.1USE CASE DIAGRAM: Workflow of MathML Editor

3.3 Proposed design:

Our project will be a web based Math Editor which provides the following functionalities:

- An Interactive GUI which contains the menus, toolbars, and a workspace for typing the equation.
- The menu bar contains menu which is similar to all web based editors including an extra menu called as "**Math**" which contains menu items for all the Mathematical operators.
- It also contains toolbar which contains the frequently used symbols.
- The user will be provided with the facility of adding symbols on to the toolbar.
- Each menu item in the menu "Math" contains a list of associated symbols along with its symbols for users to easily interpret the symbols.
- Each menu in turn has a menu item called "**Add Toolbar**" in which the toolbar of that associated menu appears for easily selecting the symbols.
- Our web based editor will be implemented based on MathML standard.
- The user is provided to save the equation as a jpg(jpeg) or other picture format so that the user can import the image file on to his/her application.
- The user is also provided a facility to export the image file to the editor and then can be edited.

CHAPTER 4
SYSTEM IMPLEMENTATION

4.1 EVENT HANDLING IN EQUATION EDITOR

It refers to the process of handling user actions when he goes through the process of creating equations . It also involves guiding the user through the process of creating mathematical expressions. To generate MathML when the user is done with equation creation, the List of user actions needs to be tracked for dynamic MathML generation.

4.1.1 **Equation elements :**

Operators , Operands, Superscripts, Subscripts, Special Math Symbols(Square root,Integral,Limits),UnderOver,Matrix,Fence(brackets.)

4.1.2 **Pre-requisites**:

Action Script GlobalData.as keeps track global elements such as,

- **Current Container:**

  Refers to the current display object which has focus.(i.e. The user has clicked on a particular display object to which the equation is going to be added).The current Container acts as a pre parent to the math Equation that the user might type. Before Equation creation the current container holds the super parent container which is the parent to all math expressions.

- **Counter Id :**

  As and when the user adds math elements into the drawing board, an ID is assigned to the particular math element .Global Data keeps a counter of the entire Id's assigned at any point in the equation creation.

- **Math Table :**

  An array of User defined objects called Math Node.

- **Math Node :**

  An object which uniquely defines a math object in the equation.

Attributes of MathNode

Nodeid : The globally assigned Identifier

ParentId : The identifier of the parent math node which holds this node.

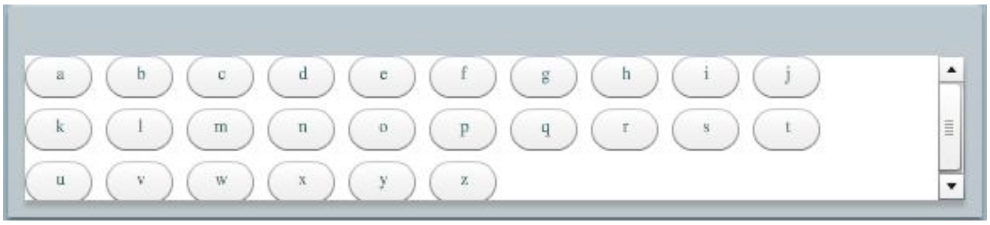
Node Name : The name which specifies what type of math node it is. The name corresponds to the math ml tag which describes .
For e.g.: mo for operators ,
mi for operands,
mn for constants,
msup for superscripts

Node Text : The text of the node. Applicable only for
mn,mo,mi

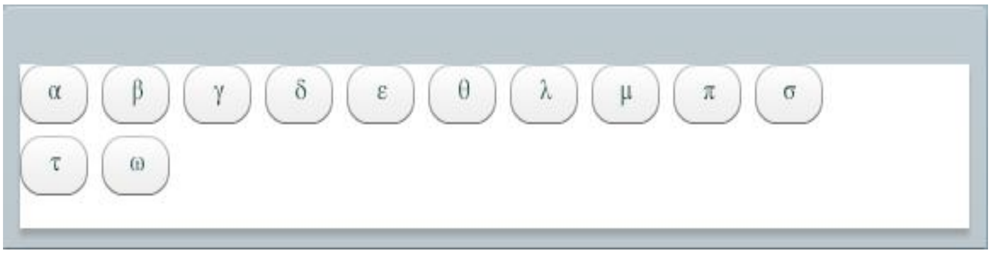
Attributes : Properties of math elements like fontsize,style,thickness etc.

4.2 **Event Handling for Operands**:

When the user clicks on the Operand, And enters the desired operand. (either alphabetical or alphanumerical).Global data assigns an Id to the Operand. The Global data has track of the parent container to which the operand is getting added. From the data from the global data, A corresponding math node is created with the operand's Id,the current container's Id, and the operand name is the nodename.This Math Node is added to the Math Table of Global data.



4.1 Operand Alphabets :WYSIWYG Editor



4.2 Greek Alphabets in WYSIWYG Editor

4.3 **Event Handling for Constants :**

When the user clicks on the Operand, and enters the desired operand. (either alphabetical or alphanumerical).Global data assigns an Id to the Operand. The Global data has track of the parent container to which the operand is getting added. From the data from the global data, A corresponding math node is created with the operand's Id,the current container's Id,And the operand name is the nodename.This Math Node is added to the Math Table of Global data.

4.4 **EVENT HANDLING FOR OPERATOR**

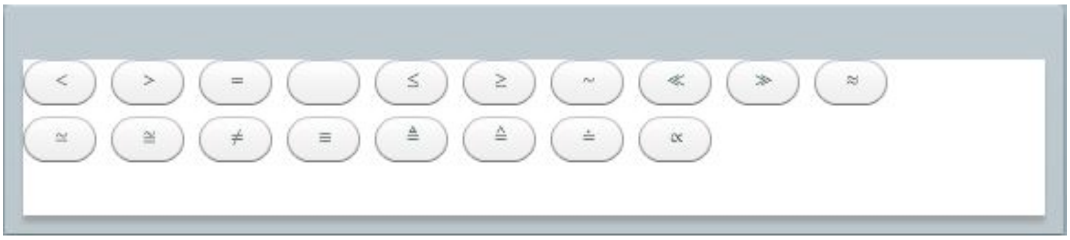
The event for Operator is handled in the Following way.

4.4.1 **Scenario:**

The User clicks on the box wherein he wants to place an operator of his choice. Then He clicks on the Operator:

4.4.2 **Rendering of Operator:**

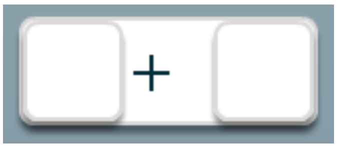
Operators are classified as Unary and Binary Operators.



4.3 Operators in WYSIWYG Editor

Every Binary Operator takes Two operands .Hence Operators are rendered in the following way.

<Operand> <Operator> <Operand>



4.4 Operator Rendering in WYSIWYG Editor

This restriction ensures that the User does not create meaningless or ambiguous mathematical equation.The Global data assigns Id to the First Operand,Operator,and then

Second Operand. The Math Table is populated with details about three of these Math Nodes.

Note:

Since the operator should lie between the two operand ,It is assigned an Id that is the average of the The Two operands.

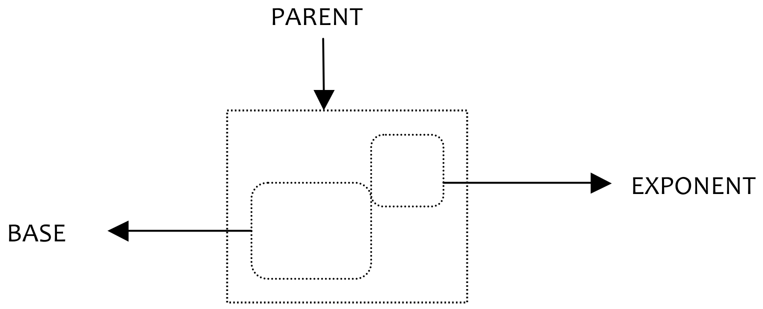
4.5 EVENT HANDLING FOR SUPERSCRIPTS AND SUBSCRIPTS

4.5.1 Description :

Superscript in MathML takes two mathnodes as their children. The base and the exponent. The Global Data assigns Id to the Superscript node which is an imaginary node, Which exists just to hold the two children but has no physical rendering.

4.5.2 MathNode :

<msup><Id><ParentID of Current ontainer><null>



4.5 SuperScript Box Structure :WYSIWYG EDITOR

4.5.3 Rendering of Superscripts :

The base and the exponent have their parent nodes to the NodeId pointed by the imaginery *msup* node.The base and exponent can each hold either a simple operand or a constant or a mathematical expression on their own. Hence the node type is rendered as mrow(The Generic Math tag).

Base <mrow><Id><Msup Id><null>
Exponent <mrow><Id><Msup Id><null>

4.6 SuperScript MathML Tag Structure

4.6 **EVENT HANDLING FOR FRACTION**

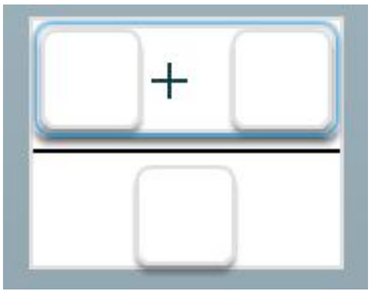
4.6.1 **Description :**

Fraction as a math element has two sub elements named the numerator <mrow> ,denominator <mrow> .

4.6.2 **Rendering of a fraction:**

When rendering a fraction visually ,there is a median which seperates the numerator and denominator.The width of the median is governed by the widths of the numerator and denominator.

Width<median>=Max(width<num>,width<den>)



4.7 Fraction Rendering in WYSIWYG Editor

To handle this width change, a resize event listener is added to both the numerator and denominator. Hence whenever there is a change in the size of the num or den, a function is called. This function (Redraw Median) will compare the widths of the numerator and denominator and will redraw the median.

Global data assigns Id to the imaginary Parent Container, the numerator and the denominator,

<mfrac><Id><Id of Current Container><null>
<mrow><Id><mfrac Id><null>
<mrow><Id><mfrac Id><null>

4.8 Fraction MathML Tag Strucure

4.7 EVENT HANDLING FOR SQUARE ROOT

4.7.1 Description :

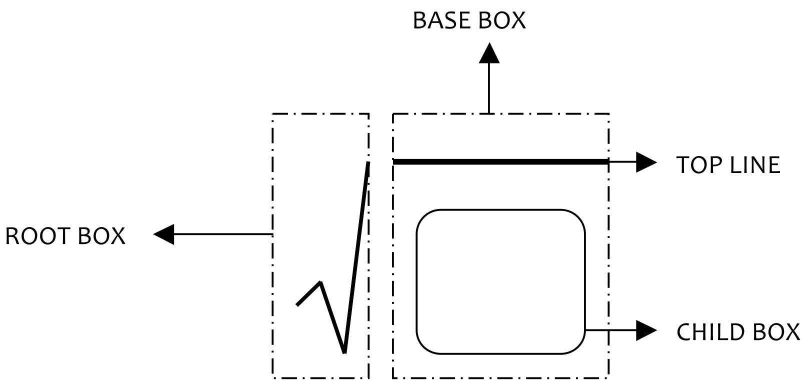
The Math element Square root <msqrt> takes one child which is rendered within the Square Root.

```
<msqrt>
        <mn>2</mn>
</msqrt>
```

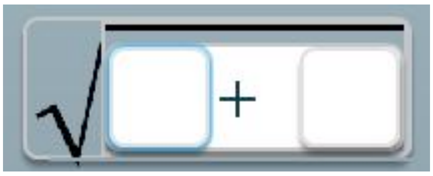
= √2

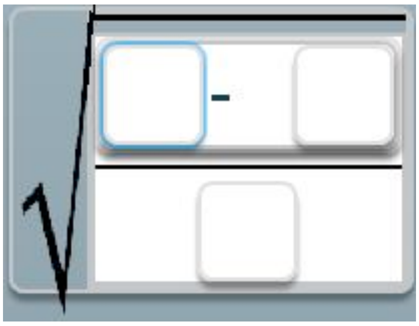
The SqrtBox has five children,

- *RootBox:* The HBox which holds the root.
- *Base Box:* The HBox in turn consists of
  - *Top Line*: The Line which is drawn above the child of the square root.
  - *Child Box*: Holds the content within the Square root.



4.9 Square Root Box in WYSIWYG Editor





Rendering of Square root

The size of the square root and the top line is governed by the size of the Child inside the box.The Square root might hold a simple operand or a constant or a mathematical expression. Depending on that the square root should increase or decrease its height while the width remains a constant. The top line should increase or decrease its width while the height remains constant.

Global data assigns Id to the parent sqrt container, and the child within.

```
<msqrt>, <Id><parent Id of current Container><null>
<mow><id><sqrt Id><null>
```

Fig 4.10 Square root MathML tag structure

4.8 **EVENT HANDLING FOR FENCE <mfence>:**

Fig 4.11 Fences Rendering in WYSIWYG Editor

4.8.1 **Description :**

MathML Fence has two attributes open and close which specifies the opening and closing brackets.

```
<mfence open=’(‘ close=’)’>
        <mn>2</mn>
</mfence>
```

= (2)

4.8.2 **Rendering of Fence :**

When the user clicks on the fence he is provided with both the open and close braces with a content box in between.

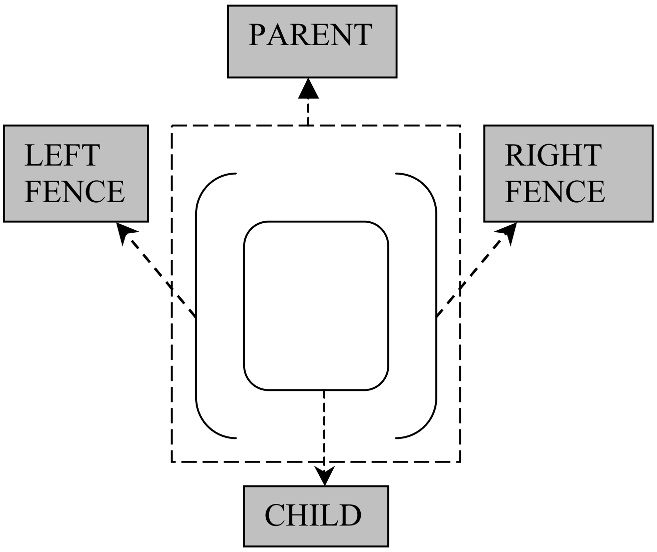


Fig 4.12 Fence Box :Internal Structure

Both are braces are provided to ensure the BODMAS rule. The height of the fence is governed by the height of the mathematical expression inside it. An event listener listens to the resize of the child container and redraws the fence of desired height .Global data assigns Id's to the parent fence and the child container.The math node for fence includes two attributes for open and close.

4.9 **SPECIAL EQUATION ELEMENTS :**

SUMMATION , INTEGRAL ,UNION.
MAPS TO.

4.9.1 **Rendering of Summation ,Integrals:**

Operators like Summations and Integral are governed by the Operand they apply to.MathML editor gives a range of options to the user to handle these operators. Similar to the square root, event listeners are added to the Integral and Summation elements which are redrawn when the size of its child is changed.

4.10 CUSTOM EQUATION ELEMENTS IN FLEX

4.10.1 Custom Operand

The Math Equation Editor places some restrictions on the user .The restricted from creating meaningless or ambiguous equations.

Meaningless Equations :

E.g. : a+*-/b

Ambiguous Equations :

E.g. : **a + b * c**

To prevent the user from doing this, a restriction is placed on the usage of operators and operands .

A custom operand is a normal operand Box which the user adds after or before an operator. The operand Box is provided with two markers on either side which allows the user to add an operator after or before the operand

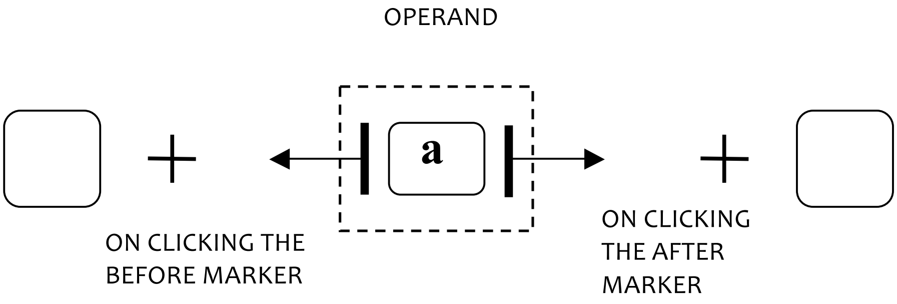


Fig 4.13 Custom Operand in WYSIWYG Editor

4.10.2 Inserting a binary operator

A binary operator can be inserted only between two operands / numbers. While writing an equation, operator can be inserted either before an operand or after an operand. If an operator is inserted before an operand, then the operator appears along with an empty box before to it. (here empty box can get any operand as value).Similarly, if an operator is inserted after an operand, then the operator appears along with an empty box after to it. (here empty box can get any operand as value)

4.10.3 **Creating Custom operand**

Instead of displaying the operands as such, the operands can be displayed using a custom manner. Custom operand is used to display a cursor before and after the operand. The cursor will be displayed only if the mouse pointer moves over the operand. Two vertical bars as label cursor which appears before and after the operand. The user can select any one of the cursor label, which indicates that the binary operator is going to insert at that position.If label before the operand is selected, then before flag is set. When the before flag is set, then operator follows the empty box is added to it. If label after the operand is selected, then after flag is made set. When the after flag is set, then operator is followed by an empty box is added.

4.11 MATH ML GENERATION

The math table holds an array of math nodes , all the nodes which the user has added to the math equation. It is later utilized to create the mathml for the equations. Math ML is basically a XML which carries parent child relationships. The Math Table currently holds data of all the nodes in the Equation.

4.11.1 XML Building in Flex:

Flex provides the XMLDocument class which is used to create the mathml code. The XML class contains methods and properties for working with XML objects.

The MathML is constructed the following way

Step 1: Find the root math node from the node table .This is the node which has parentId as 0; An XML Node is created for the same.

XMLNode(type:uint, value:String)

Creates a new XMLNode object with the name specified in the parameter

create Element (name: String): XMLNode

Creates a new XMLNode object with the name specified in the parameter.

Step 2: On Iterating through the Math Table find all math nodes which have parent id as rootNodeId.

XMLNodes are created for the same and each of the nodes is added to the parent Node.

append Child (node:XMLNode):void

Appends the specified node to the XML object's child list.

Step 3: For each of the child nodes ,repeat steps 2.and 3.

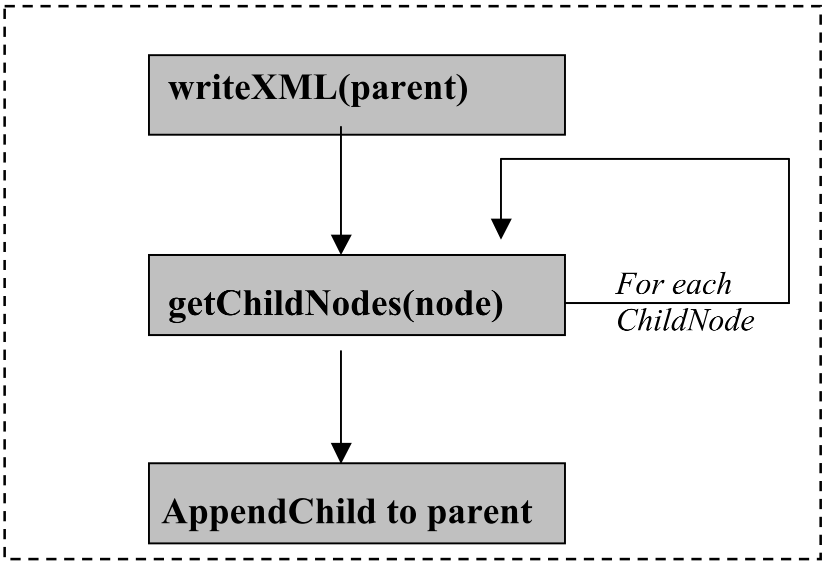


Fig 4.14 MathML Generation

4.12 EQUATION EDITING

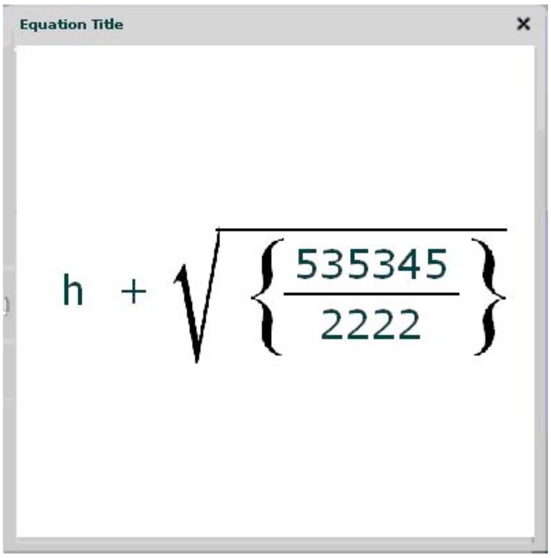
The user creates an Equation for the First time using *WYSIWYG editor*. The user then saves the equation with a particular Equation title in the database. The Equation title along with the generated MathML is stored in the database.Subsequent visits of the user may require to render the MathML as editable equation.

4.12.1 VIEW EQUATION

This module presents a read only mode of equation to the user. The MathML is fed as input to the MathML Parser which parses the xml and draws the equation. The resultant equation is rendered in a Popup window .

4.12.1.1 EQUATION DISPLAY WINDOW

Flex provides the Title Window component which is made as a custom component called Equation Displayer. Title Window has a property “title” which holds the Name of the equation. DrawingBoard is a canvas which holds the equation.PopUpManager creates a PopUp of the Title Window as a modal



$$h + \sqrt{\left\{\frac{535345}{2222}\right\}}$$

Fig 4.15 Equation in View Only Mode

4.13 **POINT AND CLICK EDITING**

The MathML is retrieved from the database and is used to create an editable form of the equation. The editor supports **"Point and Click" Editing** of equations.

Providing the Edit Features requires the following to be taken care of,

- Visual Rendering of Equation in an editable form.
- Reverse Engineer math table to generate updated mathml.
- Provide the Math Palette

The former inherits methods and classes which were used in Equation Creator. The XMLReverter class reverses the MathML into an Array of MathNodes (MathTable).

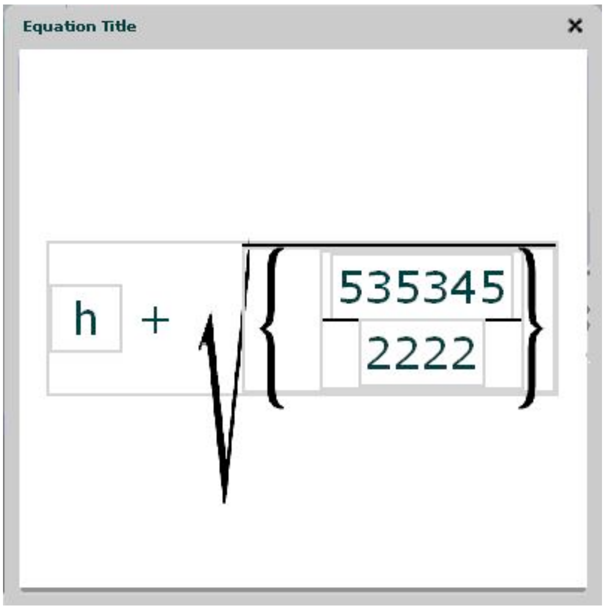


Fig 4.16 Equation in Editable Mode

- To make the equations elements editable, First they have to be rendered as selectable.
- The setBorder() method sets the border to the equation elements.
- The addEffects () method adds event Listeners which listen for Mouse Click events.
- Any Subsequent Changes to the Equation Should be reflected in the Math Table.

  For E.g. : Delete a Square Root or fraction.

  1). Find the CountId of the Deleted Element.

  2) Remove all entries from the Math Table which have the parentId as the deleted element CountId .

  3) Reflect the Changes visually. Initialize a new HBox

4.13.1 **DRAW EDITABLE EQUATION**

The MathML Input is passed to the parser through recursive calls to each of the XML Nodes. For each of the MathML tag, An editable equation element has to be rendered .The Methods and Classes used in Equation Creation are inherited to provide the display of editable equation.

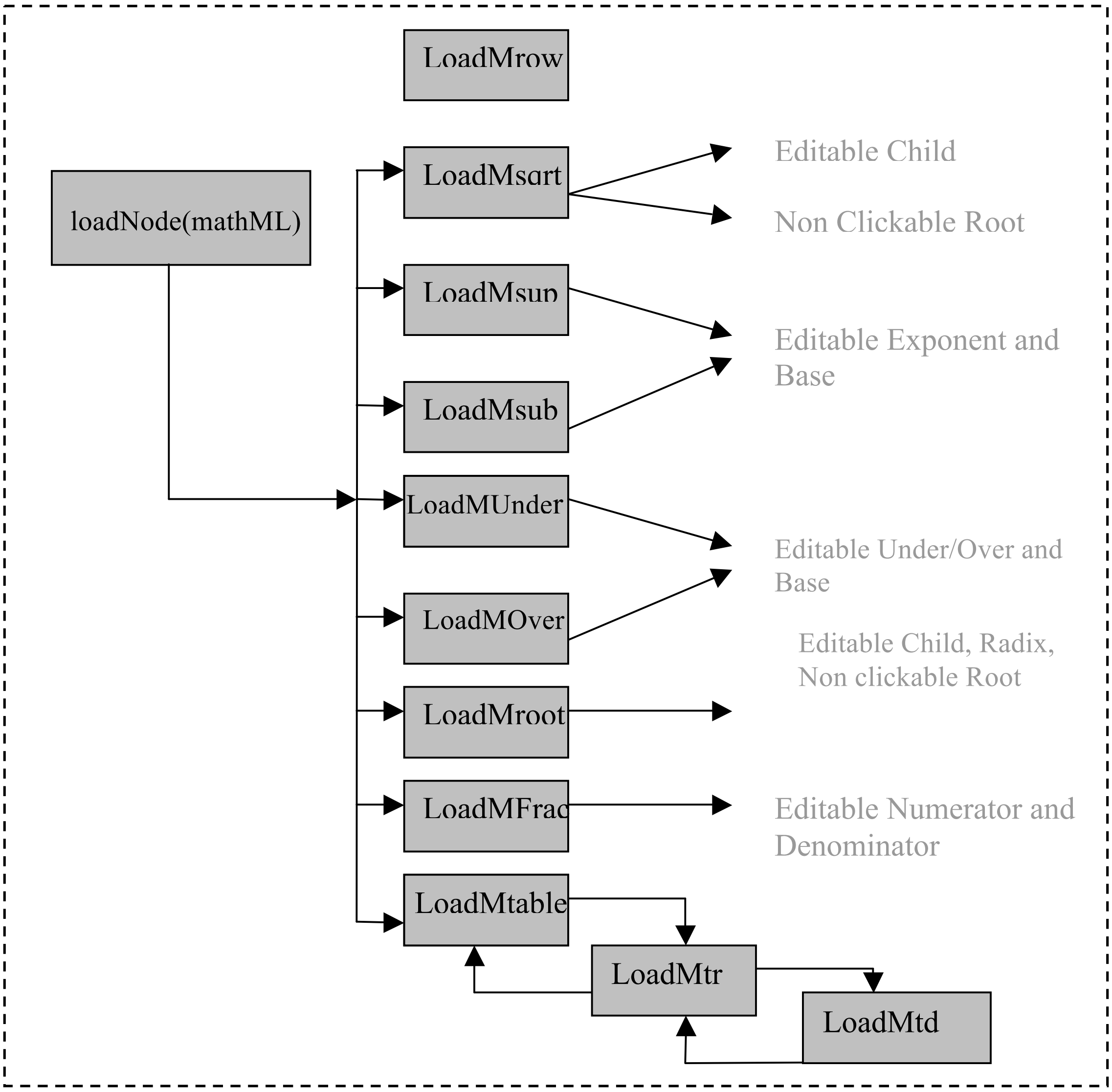


Fig 4.17 Math Equation Edit Parser

4.13.2 **XML Reverter:**

Given the MathML,To display the Math elements in a editable form, First a math table needs to be generated.XML Reverter Class takes the MathML as XML and generates the MathTable.Starting from the Parent ,a math node is created for each of the XMLNode,using information from the XMLNode(Node type and Node text) and User generated NodeId,ParentId,Position

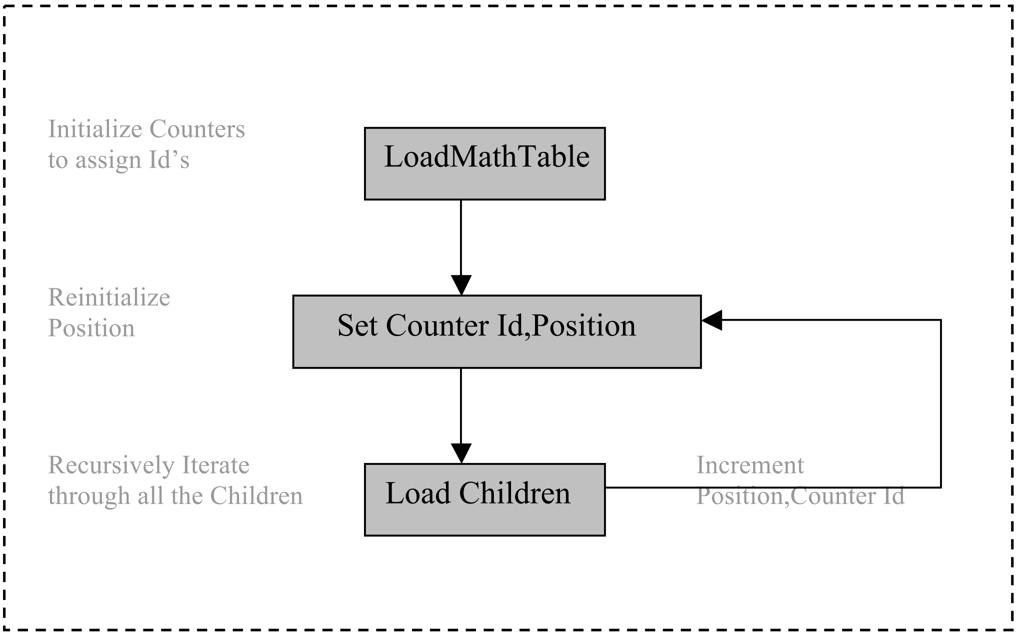


Fig 4.18 XML Reverter

4.14 HELP ASSISTANT IN EQUATION EDITOR

The MathML editor is an Intuitive editor which guides the user through his equation creation process. To make it intuitive we need a help assistant which triggers suggestions and information to the user when he creates the Equation.

4.14.1 Logic of Help Assistant :

4.14.1.1 Com.messageservice.InfoMessageService:

InfoMessageService is an Action Script Class in the com.messageservice package .The methods of this class gets called when the user performs either of the following actions .

1) Clicked on the after or before markers of an Operand or Number.
2) When the user attempts to add meaningless equations. Like Add a fraction after an operand(Only an Operator can be added).
3) When the user creates a matrix but leaves the cells of the matrix empty.
4) Any other attempts to create ambiguous equations is cancelled and the user is notified

InfoMessageService has the following methods.

1) SendAfterOperatorMessage
2) SendBeforeOperatorMessage
3) SendOperandMessage
4) SendErrorMessage

Each of the following methods build an XML "Info" and return the Info XML. Info XML has two children.

Icon :Specifies the location of the ICON image .

Message : A text of the corresponding Info message

```
public var afterIcon:String="icons/Backward.png";
public var afterOperandMessage:String="Insert only an Operator after  the
                                        operand";
```

```
var xmlMess1:XML=<info/>;
xmlMess1.appendChild('<icon>'+afterIcon+'</icon>');
xmlMess1.appendChild('<message>'+afterOperandMessage+'</message>’);
return xmlMess1;
```

4.14.1.2 InfoMessager.MXML

The infoMessager is a custom component in MXML which acts as a message wrapper.It wraps the message text and the corresponding icon.

Flex allows developers to build custom components, based on native Flex containers or controls, that can each contain their methods, or even overwrite native methods. This architecture promotes a modular design, code reuse, and lets multiple developers contribute to the implementation.

InfoMessager is a HBox in Flex. It has two Children.

iIcon : Flex Image Loader which loads the Icon

iMessage : Flex Textbox which holds the Image.

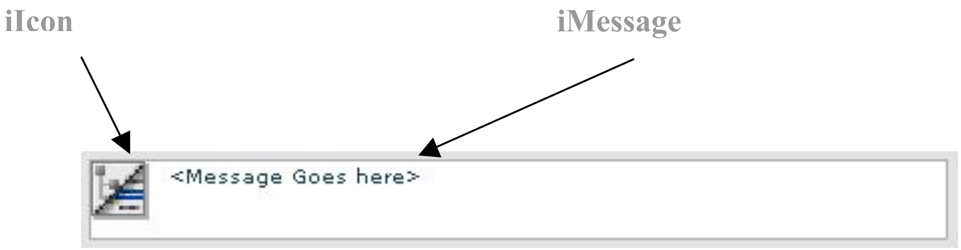
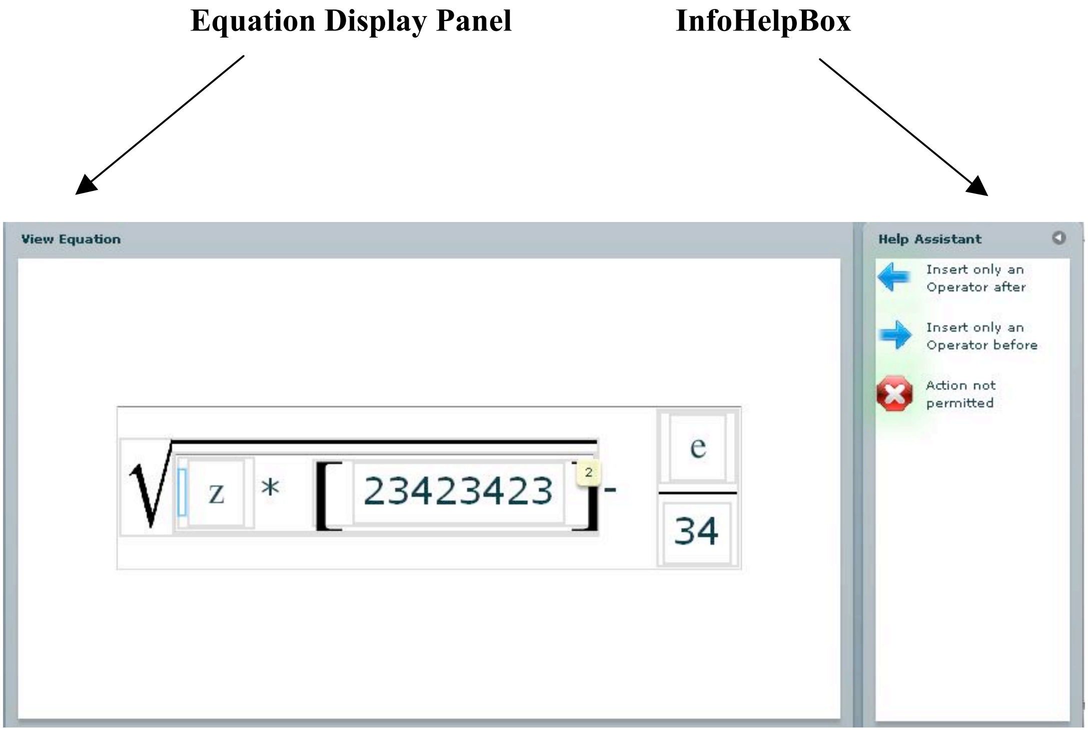


Fig 4.19 InfoMessager.MXML

4.14.1.3 **InfoHelpBox.MXML**

The help Assistant is a Custom Title Window, Which takes the right most position in the Moving Frame.The Title Window has a method addMessage(info:XML)

1) Takes the XML message created by InfoMessageService.
2) Parse the XML for ICON and MESSAGE
3) Creates an Object to InfoMessager.Set the iIcon and iMessage.
4) Adds the InfoMessager Object to the HelpBox.



Equation Display Panel

InfoHelpBox

View Equation

Help Assistant

Insert only an Operator after

Insert only an Operator before

Action not permitted

Fig 4.20 Equation Editor with Help Assistant

CHAPTER 5

SOFTWARE TESTING AND PERFORMANCE TUNING

5.1 SOFTWARE TESTING

5.1.1 INTRODUCTION

Software testing is the process used to identify the correctness, completeness, security, and quality of the developed computer software. Testing is a process of technical investigation, performed on behalf of stakeholders, that is intended to reveal quality-related information about the product with respect to the context in which it is intended to operate. This includes, but is not limited to, the process of executing a program or application with the intent of finding errors. Before the formal testing phase the application is tested at regular intervals at the development phase itself.

5.1.2 TESTING LEVELS

There are many strategies that can be used to test software. Usually software companies follow an incremental approach with various levels of testing which are as follows:

- Unit testing tests the minimal software item that can be tested.
- Component Integration testing exposes defects in the interfaces and interaction between integrated components.
- System testing tests an integrated system to verify that it meets its requirements.

After modifying software, either for a change in functionality or to fix defects, a regression test re-runs previously passing tests on the modified software to ensure that the modifications haven't unintentionally caused a regression of previous functionality.

5.1.2.1 UNIT TESTING

Unit testing or component testing focuses verification effort on the smallest unit of software design – the software component or module. Unit test cases (UTCs) were created to carry out unit testing for each module after the development. A test case is usually a single step, and its expected result, along with various additional pieces of information .i.e. for various inputs the expected results are tabulated in a spreadsheeare compared with the actual results.t When a person tests the module using provided UTCs and finds a bug, they record it in a Bug Tracker which is in the repository so that all developers can access it. It has a column open/closed. When writing the case of bug, it is kept in open state. Once the bug is fixed, the entry can be closed in the bug tracker.

5.1.2.3 SYSTEM TESTING

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that the system elements have been properly integrated and perform allocated functions.

- **Recovery Testing**: In this kind of testing, the status of the operation if the system fails in the middle has to be looked into. In reporting engine, if the connection with the servers could not be established or the connection fails, an error message should be displayed to the user. In case of connection time out with the database server or server crash, an appropriate error message should be displayed to the user.

- **Security Testing:** During security testing, the tester should assume all possible roles and check for illegal access or entry to system. Users should be able to enter only those modules which they are allowed to use.

- **Stress Testing**: This is an important testing that has to be performed for all web applications where many users are likely to access the system simultaneously. For testing the analysis phase in reporting engine, many users were allowed to sign in simultaneously and send mails and replies. The system was tested for maximum loads and watched whether it works without crashing down.

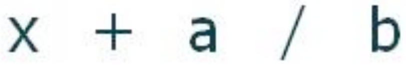
- **Performance Testing:** The run-time performance of software is tested here. This testing might be done at any stage, even in unit testing for checking the performance of an individual module.

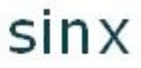
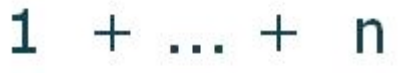
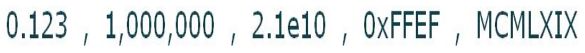
5.3 MathML Test Suite

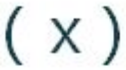
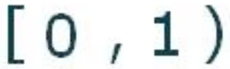
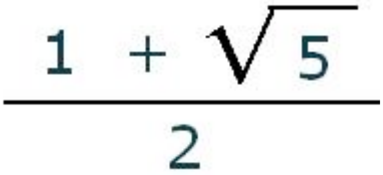
The MathML specification is intended to be used by a wide variety of systems which need to be able to exchange MathMl information. This group of tests is there to encourage interoperability by allowing developers to test the features of MathML markup and to see what features others are already successfully supporting. We all need MathML to be implemented correctly and consistently so as to convey scientific and technical information efficiently.

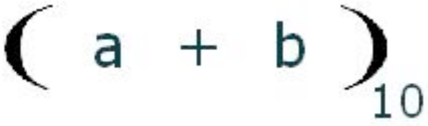
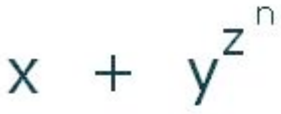
TOKEN ELEMENTS

:

MathML Test Case	**Editor Rendering :**
<math> <mrow> <mi> x </mi> <mo> + </mo> <mrow> <mi> a </mi> <mo> / </mo> <mi> b </mi> </mrow> </mrow></math>	 x + a / b

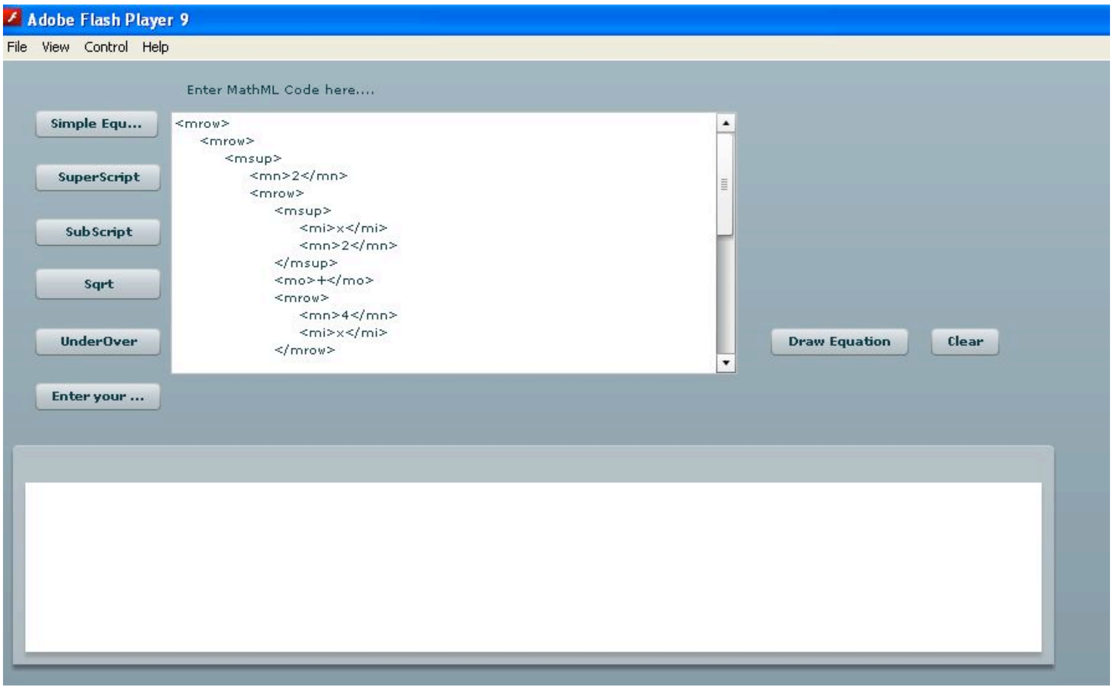
MATHML TEST CASES	EQUATION EDITING
<math> <mrow> <mi> sin </mi> <mo>&ApplyFunction;</mo> <mi> x </mi> </mrow> </math>	 sinx
<math> <mrow> <mn> 1 </mn> <mo> + </mo> <mi> ... </mi> <mo> + </mo> <mi> n </mi> </mrow> </math>	 1 + ... + n
<math> <mrow> <mn> 2 </mn> <mo> , </mo> <mn> 0.123 </mn> <mo> , </mo> <mn> 1,000,000 </mn> <mo> , </mo> <mn> 2.1e10 </mn> <mo> , </mo> <mn> 0xFFEF </mn> <mo> , </mo> <mn> MCMLXIX </mn> <mo> , </mo> <mn> twenty one </mn> </mrow> </math>	 0.123 , 1,000,000 , 2.1e10 , 0xFFEF , MCMLXIX

MathML TEST CASE	EDITOR RENDERING
<math> <mrow> <mo> (</mo> <mi>x</mi> <mo>) </mo> </mrow> </math>	 (x)
<math> <mrow> <mo> [</mo> <mrow> <mn> 0 </mn> <mo> , </mo> <mn> 1 </mn> </mrow> <mo>) </mo> </mrow> </math>	 [0 , 1)
<math> <mfrac> <mrow> <mn> 1 </mn> <mo> + </mo> <msqrt> <mn> 5 </mn> </msqrt> </mrow> <mn> 2 </mn> </mfrac> </math>	 $\frac{1 + \sqrt{5}}{2}$

MATHML TEST CASE	EDITOR RENDERING
<math> <msub> <mfenced open=’(‘ close=’)> <mrow> <mi>a</mi> <mo>+</mo> <mi>b</mi> </mrow> </mfenced> <mn>10</mn> </msub> </math>	 $(a + b)_{10}$
<math> <mrow> <mi>x</mi> <mo>+</mo> <msup> <mi>y</mi> <msup> <mi>z</mi> <mi>n</mi> </msup> </msup> </mrow> </math>	 $x + y^{z^{n}}$

TORTURE TESTS

MathML Equation Renderer Prototype:



```
<mrow>
   <mrow>
      <msup>
         <mn>2</mn>
         <mrow>
            <msup>
               <mi>x</mi>
               <mn>2</mn>
            </msup>
            <mo>+</mo>
            <mrow>
               <mn>4</mn>
               <mi>x</mi>
            </mrow>
```

Fig 5.1 MathML Equation renderer Prototype

Comple Equations

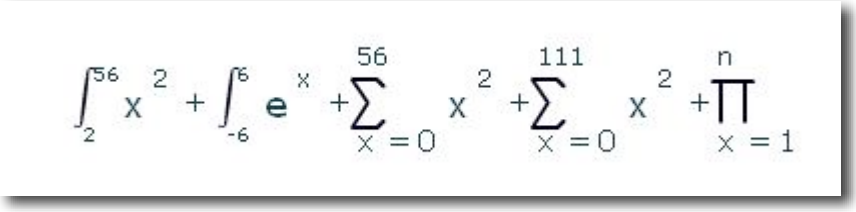
Test Case 1:

```
<mrow>
        <msubsup>
          <mo>&int;</mo>
          <mn>2</mn>
          <mn>56</mn>
        </msubsup>
        <msup>
                <mi>x</mi>
                <mn>2</mn>
        </msup>
        <mtext fontstyle='italic'>dx</mtext>
        <mo>+</mo>
        <msubsup>
                <mo>&int;</mo><mi>-6</mi><mi>6</mi>
        </msubsup>
        <msup>
                <mi>e</mi>
                <mn>x</mn>
```

Contd..

```
            </msup>
            <mtext fontstyle='italic'>dx</mtext>
            <mo>+</mo>
            <munderover>
                  <mo>&sum;</mo>
                  <mrow>
                  <mi>x</mi><mo>=</mo><mn>0</mn>
                  </mrow>
                  <mn>56</mn>
            </munderover>
            <msup>
                  <mi>x</mi>
                  <mn>2</mn>
            </msup>
            <mo>+</mo>
            <munderover>
                  <mo>&sum;</mo>
                  <mrow>
                  <mi>x</mi><mo>=</mo><mn>0</mn>
                  </mrow>
                  <mi>111</mi>
            </munderover>
            <msup>
                  <mi>x</mi><mn>2</mn>
            </msup>
            <mo>+</mo>
            <munderover>
                  <mo>&prod;</mo>
                  <mrow>
                  <mi>x</mi>
                  <mo>=</mo>
                  <mn>1</mn>
                  </mrow>
                  <mi>n</mi>
            </munderover>
      <mrow>
 </mrow>
</mrow>
```

EDITOR RENDERING :



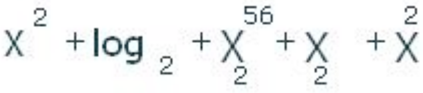
$$\int_{2}^{56} x^{2} + \int_{-6}^{6} e^{x} + \sum_{x=0}^{56} x^{2} + \sum_{x=0}^{111} x^{2} + \prod_{x=1}^{n}$$

Fig 5.2 Complex Equation 1

Complex Equation #2:

```
<mrow>
        <msup>
                <mi>X</mi>
                <mn>2</mn>
        </msup>
<mo>+</mo>
        <msub>
                <mi>log</mi>
                <mn>2</mn>
        </msub>
<mo>+</mo>
        <msubsup>
                <mi>X</mi>
                <mn>2</mn>
                <mn>56</mn>
        </msubsup>
<mo>+</mo>
        <munder>
                <mi>X</mi>
                <mn>2</mn>
        </munder>
<mo>+</mo>
        <mover>
                <mi>X</mi>
                <mn>2</mn>
        </mover>
        </mrow>
```

EDITOR RENDERING :



$$X^{2} + \log_{2} + X_{2}^{56} + \underset{2}{X} + \overset{2}{X}$$

Fig 5.3 Complex Equation 2

COMPLEX EQUATION #3 :

MathML Code:

```
<mrow>
    <mrow>
        <msup>
            <mn>2</mn>
            <mrow>
                <msup>
                    <mi>x</mi>
                    <mn>2</mn>
                </msup>
                <mo>+</mo>
                <mrow>
                    <mn>4</mn>
                    <mi>x</mi>
                </mrow>
                <mo>+</mo>
                <mn>4</mn>
            </mrow>
        </msup>
        <mo>+</mo>
        <mrow>
            <mn>4</mn>
            <mi>x</mi>
        </mrow>
        <mo>+</mo>
        <mn>45</mn>
    </mrow>
    <mo>=</mo>
    <mn>0</mn>
</mrow>
```

EDITOR RENDERING :

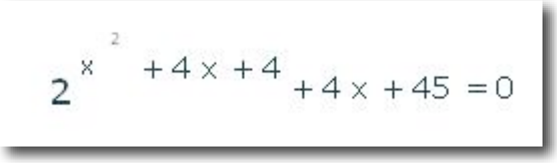


Fig 5.4 Complex Equation 3

CONCLUSION

The advent of MathML has revolusionised the way Mathematics has been rendered in web.Use of MathML promises more efficient and high performance rendering of mathematical equations in the web.

Thus,This MathML editor comes in handy to any scientific publisher or teacher seeking mathematical content.The platform indendence of the Editor makes it compact and portable.

Future enhancements would include addition of more customized mathematical elements to cover wide topics of algebra,trigonometry etc and Expression Evaluation as part of content MathML