

8th OpenMath Meeting (joint with JEM)

RISC, Schloss Hagenberg, University of Linz
June 25-26th, 2007

Evolving the WebALT Mathematical Grammar Libraries

GLÒRIA CASANELLAS, JORDI SALUDES AND SEBASTIAN XAMBÓ

Facultat de Matemàtiques i Estadística
Universitat Politècnica de Catalunya

Index

- The WebALT TextMathEditor (TME06)
 - The WebALT MGL06
- (Some) Features of the TME06
 - (Some) Features of the MGL06
- MGL08
- TME08
- Conclusion

TME06

Graphical user interface (java-applet) allowing:

Editing of exercise statements consisting of sentences with embedded mathematical formulas (palette-based input)

OpenMath language-independent markup for exercise sentences (text must comply with GF grammar).

Exporting format(s).

Hinting system prompting the user with words/maths objects choices

(TME06)

Calculate the determinant of the transpose of $\begin{pmatrix} 3 & 2 \\ 1 & 1 \end{pmatrix}$.

Select a real number x such that $x^5 = x$ and $x \neq 1$.

Function → Insert math Delete input Delete output

Arithmetic Algebra Analysis Sets Trigonometric Hyperbolic Other Logic Piecewise

$+$	\times	$-$	$\sqrt{\quad}$	$\frac{\quad}{\quad}$	\quad^{-1}	$ \quad $	gcd	\sum_{\quad}^{\quad}	(\quad)
$-$	\div	\quad^{\quad}	$\sqrt{\quad}$	$\frac{1}{\quad}$	$\ \quad\ $	lcm	\prod_{\quad}^{\quad}		

\square

'False'
'True'
0
1
<Function>
<Number>
<Proposition>
<Set>
<Variable>
a
add

add <Numbers> to <Numbers> .
add <Numbers> to <Numbers> . <Sentence> <Sentence>
differentiate <Functions> .
differentiate <Functions> . <Sentence> <Sentence>
differentiate <Functions> at <Numbers> .
differentiate <Functions> at <Numbers> . <Sentence> <Sentence>
divide <Numbers> by <Numbers> .
divide <Numbers> by <Numbers> . <Sentence> <Sentence>

<< Back clear Next >>

Catalan English French Italian Spanish Swedish

(Some) features of the TME06

Inserting a formula: formula icon + formula

Predictive typing: Hinting of how to continue a sentence done in two columns, one for next possible words and another for possible sentence-tail forms

Parsing words: Only when a required word is completed are possible words / sentence-tail forms proposed

Parser and cursor movement are not (suitably) coordinated

(Some) features of the MGL06

Small vocabulary and a small number of structures

(TME06 is only a show-case)

Lack of functoriality

MGL08

GF grammar enhancements

- **In grammar content** (top layer of grammars – Operations)
- **In functoriality** (pattern-matching transformations of the abstract trees which code the exercise – rather than transformations to produce text)

Enhancement in grammar content

- Adding referential elements

- “the last matrix”
- “it”

(Adding productions to the *Operations* layer enabling authors to deal with such kind of clauses)

- Aggregation

(Joining two or more elements to get a similar aggregated one)

- “Compute the determinant of M , N and $M + N$.”
(Example of Noun Phrase aggregation)
- Exercise aggregation

Functoriality

(Abstract tree manipulation for:)

- More natural rendering of text (pronouns)

“Compute the derivative of f . Compute the derivative of the inverse of f ”

→

“Compute the derivative of f and of its inverse”

- Aggregation of simple exercises

“Compute A; Compute B; Compute C”

→

“Compute:

1. A;
2. B;
3. C.”

TME08

- Maths mode activated when typing a formula
- A type-driven module will hint possible sentence-tails
- When typing a word, its possible continuations will be displayed
- Cursor and parser work in a coordinated way. This will decrease time of parsing – no multiple parsing of same sentences
- Improved parser: grammar (vocabulary and structures) can be extended at will
- Spell checker that recognizes
 - spelling mistakes
 - grammatical sentences similar to those produced by the user

Conclusion

Evolution to fit aimed functionality is largely a consequence of repeating over and over the virtuous loop consisting

- variation of objects
- selecting the fittest
- replicating/reusing them

(VSR process)

We propose to evolve TME06 into TME08 by a VSR process focused on GF grammars and their bearing on efficient parsing and on efficient mapping into OpenMath (say), in natural-language-independent form.