omtext: Semantic Markup for Mathematical Text Fragments in \LaTeX*

Michael Kohlhase
FAU Erlangen-Nürnberg

\url{http://kwarc.info/kohlhase}

March 20, 2019

Abstract

The \texttt{omtext} package is part of the \texttt{\LaTeX} collection, a version of \LaTeX that allows to markup \LaTeX documents semantically without leaving the document format, essentially turning \LaTeX into a document format for mathematical knowledge management (MKM).

This package supplies an infrastructure for writing OMDoc text fragments in \LaTeX.
# Contents

1 Introduction .............................................. 3

2 The User Interface ...................................... 3
   2.1 Package Options ...................................... 3
   2.2 Mathematical Text .................................. 3
   2.3 Phrase-Level Markup ................................. 3
   2.4 Declarations (under development) .................. 4
   2.5 Block-Level Markup ................................. 5
   2.6 Index Markup ........................................ 5
   2.7 Miscellaneous ....................................... 6

3 Limitations ............................................... 6

4 Implementation .......................................... 7
   4.1 Package Options ...................................... 7
   4.2 Mathematical Text ................................... 7
   4.3 Phrase-level Markup ................................. 8
   4.4 Declarations (under development) ................. 9
   4.5 Block-Level Markup ................................. 9
   4.6 Index Markup ........................................ 10
   4.7 Miscellaneous ...................................... 12
   4.8 Deprecated Functionality ......................... 12
1 Introduction

The omtext package supplies macros and environment that allow to mark up mathematical texts in $\text{\LaTeX}$, a version of $\text{\LaTeX}/\text{\bLaTeX}$ that allows to markup $\text{\LaTeX}/\text{\bLaTeX}$ documents semantically without leaving the document format, essentially turning $\text{\LaTeX}/\text{\bLaTeX}$ into a document format for mathematical knowledge management (MKM). The package supports direct translation to the OMDoc format \cite{Koh06}.

2 The User Interface

2.1 Package Options

\texttt{showmeta} The omtext package takes a single option: \texttt{showmeta}. If this is set, then the metadata keys are shown (see \cite{Koh18a} for details and customization options).

2.2 Mathematical Text

\texttt{omtext} The \texttt{omtext} environment is used for any text fragment that has a contribution to a text that needs to be marked up. It can have a title, which can be specified via the \texttt{title} key. Often it is also helpful to annotate the \texttt{type} key. The standard relations from rhetorical structure theory: \texttt{abstract}, \texttt{introduction}, \texttt{conclusion}, \texttt{thesis}, \texttt{comment}, \texttt{antithesis}, \texttt{elaboration}, \texttt{motivation}, \texttt{evidence}, \texttt{transition}, \texttt{note}, \texttt{annotate} are recommended as values. Note that some of them are unary relations like \texttt{introduction}, which calls for a target. In this case, a target using the \texttt{for} key should be specified. The \texttt{transition} relation is special in that it is binary (a “transition between two statements”), so additionally, a source should be specified using the \texttt{from} key.

Note that the values of the \texttt{title} and \texttt{type} keys are often displayed in the text. This can be turned off by setting the \texttt{display} key to the value \texttt{flow}. Sometimes we want to specify that a text is a continuation of another, this can be done by giving the identifier of this in the \texttt{continues} key.

Finally, there is a set of keys that pertain to the mathematical formulae in the text. The \texttt{functions} key allows to specify a list of identifiers that are to be interpreted as functions in the generate content markup. The \texttt{theory} specifies a module (see \cite{KGA18a}) that is to be pre-loaded in this one\textsuperscript{1} Finally, \texttt{verbalizes} specifies a (more) formal statement (see \cite{Koh18b}) that this text verbalizes or paraphrases.\textsuperscript{2}

2.3 Phrase-Level Markup

\texttt{\phrase} The \texttt{phrase} macro allows to mark up phrases with semantic information. It takes an optional \texttt{KeyVal} argument with the keys \texttt{verbalizes} and \texttt{type} as above and \texttt{style}, \texttt{class}, \texttt{index} that are disregarded in the $\text{\bLaTeX}$, but copied into the gen-

\textsuperscript{1}EDNote: this is not implemented yet.

\textsuperscript{2}EDNote: MK:specify the form of the reference.
We use the \texttt{\lex{phrase}} for marking up phrases that serve as natural language examples and \texttt{\nlex{phrase}} for counter-examples (utterances that are not acceptable for some reason). In natural language examples, we sometimes use “co-reference markers” to specify the resolution of anaphora and the like. We use the \texttt{\coreft{phrase}}{\langle\texttt{mark}\rangle} to mark up the “target” of a co-reference and analogously \texttt{\corefs} for coreference source – e.g. for an anaphoric reference. The usage is the following:

\texttt{\lex{If \coreft{a farmer}\textsuperscript{1} owns \coreft{a donkey}\textsuperscript{2}, \corefs{he}\textsuperscript{2} beats \corefs{it}\textsuperscript{2}.}}

is formatted to

\textit{If a farmer\textsuperscript{1} owns a donkey\textsuperscript{2}, he\textsuperscript{2} beats it\textsuperscript{2}.}

The \texttt{\sinlinequote} macro allows to mark up quotes inline and attribute them. The quote itself is given as the argument, possibly preceded by the a specification of the source in a an optional argument. For instance, we would quote Hamlet with

\texttt{\sinlinequote[Hamlet, \cite{Shak:1603:Hamlet}]{To be or not to be}}

which would appear as “\textit{To be or not to be}” Hamlet, (Shakespeare 1603) in the text. The style in which inline quotations appear in the text can be adapted by specializing the macros \texttt{\@sinlinequote} — for quotations without source and \texttt{\@@sinlinequote} — for quotations with source.

### 2.4 Declarations (under development)

Declarations are special phrases that carry a lot of meaning in mathematics: they introduce and further specify the identifiers available in formulae. They are marked up via the \texttt{\vdec} macro. Inside a declaration we can use the \texttt{\vids} macro to mark up the variable names and \texttt{\vrest} to mark up the restrictions. In the simplest case we have a single variable as in “\ldots for all \texttt{\vdec[\vids[\texttt{$u$}]]{\texttt{$u$}}}”, which we mark up as.

\ldots for all \texttt{\vdec[\vids[\texttt{$u$}]]{\texttt{$u$}}}.

A more complex example has multiple identifiers embedded in a restriction, as in “Let \texttt{$x, y, z \in \mathbb{R}$}, such that \texttt{$x + 2y = z$}, then \ldots”, which we mark up as.

Let \texttt{\vdec[\texttt{$x, y, z$}]{\texttt{\vcond$\minset{x,y,z}\mathbb{Reals}$}, such that \texttt{\vrest{\texttt{$x+2y=z$}}}, then \ldots’’}}

\footnotesize
\begin{itemize}
\item[3] \texttt{EdNote:} explain and make better examples
\item[4] \texttt{EdNote:} talk with Frederic about this see what other examples there are.
\item[5] \texttt{EdNote:} how do we identify the variables in complex restriction patterns. maybe with \texttt{LMXref}, which we should reinstate for this.
\item[6] \texttt{EdNote:} document strucdec and impdec
\end{itemize}

\textsuperscript{4} EdNote: explain and make better examples
\textsuperscript{4} EdNote: talk with Frederic about this see what other examples there are.
\textsuperscript{5} EdNote: how do we identify the variables in complex restriction patterns. maybe with LMXref, which we should reinstate for this.
\textsuperscript{6} EdNote: document strucdec and impdec
2.5 Block-Level Markup

The \sblockquote environment is the big brother of the \sinlinequote macro. It also takes an optional argument to specify the source. Here the four internal macros \begin@sblockquote to \end@sblockquote are used for styling and can be adapted by package integrators. Here a quote of Hamlet would marked up as

```
\begin{sblockquote}{Hamlet, \cite{Shak:1603:Hamlet}}\obeylines
To be, or not to be: that is the question:
Whether 'tis nobler in the mind to suffer
\end{sblockquote}
```

and would render as

```
To be, or not to be: that is the question:
Whether 'tis nobler in the mind to suffer

Hamlet, (Shakespeare 1603)
```

\lec
The \lec macro takes one argument and sets it as a comment at the end of the line, making sure that if the content is too long it is pushed into a new line. We use it internally for placing the source of the \sblockquote environment above. The actual appearance of the line end comment is determined by the \@@lec macro, which can be customized in the document class.

2.6 Index Markup

The \omtext package provides some extensions for the well-known indexing macros of \LaTeX. The main reason for introducing these macros is that index markup in OMDoc wraps the indexed terms rather than just marking the spot for cross-referencing. Furthermore the index commands only indexes words unless the noindex option is set in the \usepackage. The \omtext package and class make the usual \index macro undefined.

\indi
The \indi macro renders a word and marks it for the index. Sometimes, we want to index a slightly different form of the word, e.g. for non-standard plurals: while \indi{word}s works fine, we cannot use this for the word “datum”, which has the plural “data”. For this we have the macro \aindi, which takes another argument for the displayed text, allowing us to use \aindi{data}{datum}, which prints “data” but puts “datum” into the index.

The second set of macros adds an infrastructure for multi-word compounds. Take for instance the compound “OMDoc document”, which we usually want to add into the index under “OMDoc” and “document”. \indii{OMDoc}{document} is a variant of \indi that will do just this. Again, we have a version that prints a variant: This is useful for situations like this the one in Figure 1:

```
Analogously, there are variants for tree/four-word compounds: \indiii, \aindiii, \aindii, and \indiv. For instance for “wonderful OMDoc document”. \aindiv{wonderful}{OMdoc}{document} will make the necessary index entries un-
```

\Indiii
\EdNote: implement this and issue the respective error message
We call group \textit{Abelian}, iff \ldots

and put “Abelian Group” into the index.

\textbf{Example 1:} Index markup

\begin{Verbatim}
\textbf{\texttt{\textbackslash Indi*}} \quad Finally, there are variants \texttt{\textbackslash Indi}, \texttt{\textbackslash Indii}, \texttt{\textbackslash Indiii}, and \texttt{\textbackslash Indiv} that print the capitalized version of the word complex, and \texttt{\textbackslash indis}, \texttt{\textbackslash indiis}, \texttt{\textbackslash indiiis}, and \texttt{\textbackslash indivs} that add a plurals, and ultimately \texttt{\textbackslash Indis}, \texttt{\textbackslash Indiis}, \texttt{\textbackslash Indiiis}, and \texttt{\textbackslash Indivs} that print the capitalized version of the plural.

All index macros take an optional first keyword argument: If the \texttt{loadmodules} key is given, we import the module we are in, otherwise all the currently imported modules. We do not have to require the module files, since the index is a the end of the document. If the \texttt{at} key is given, then we use that for sorting in the index.

\end{Verbatim}

2.7 Miscellaneous

We supply some text-level shortcuts for mathematical formulations, for instance $\equiv$ for “this corresponds to” or $\therefore$ for “therefore”. They are semantic in the sense that they are used as special words – not part of formulae, even though they look like mathematical symbols. The following table gives the full set.

\begin{center}
\begin{tabular}{|l|c|l|}
\hline
\texttt{\textbackslash hateq} & \texttt{$\equiv$} & this corresponds to \\
\texttt{\textbackslash hatequiv} & $\equiv$ & this statement corresponds to \\
\texttt{\textbackslash ergo} & $\therefore$ & therefore \\
\texttt{\textbackslash ogre} & $\because$ & because of \\
\hline
\end{tabular}
\end{center}

3 Limitations

In this section we document known limitations. If you want to help alleviate them, please feel free to contact the package author. Some of them are currently discussed in the \LaTeX\ GitHub repository \TeX.

1. none reported yet
4 Implementation

4.1 Package Options

We declare some switches which will modify the behavior according to the package options. Generally, an option xxx will just set the appropriate switches to true (otherwise they stay false).8

\newif\if@omtext@mh\@omtext@mhfalse
\DeclareOption{mh}{\@omtext@mhtrue}
\PassOptionsToPackage{\CurrentOption}{modules}
\newif\ifindex\indextrue
\DeclareOption{noindex}{\indexfalse}
\DeclareOption*{\PassOptionsToPackage{\CurrentOption}{modules}}
\ProcessOptions
\ifindex\makeindex\fi
\if@omtext@mh\RequirePackage{omtext-mh}\fi
\RequirePackage{xspace}
\RequirePackage{modules}
\RequirePackage{comment}
\RequirePackage{mdframed}
\RequirePackage{latexsym}

4.2 Mathematical Text

We define the actions that are undertaken, when the keys are encountered. The first set just records metadata; this is very simple via the \addmetakey infrastructure \[\text{\textbackslash Kohl18a}\]. Note that we allow math in the title field, so we do not declare it to be \texttt{Semiverbatim} (indeed not at all, which allows it by default).

\srefaddidkey{omtext}
\addmetakey{omtext}{functions}
\addmetakey*{omtext}{display}
\addmetakey{omtext}{for}
\addmetakey{omtext}{from}
\addmetakey{omtext}{type}
\addmetakey*{omtext}{title}
\addmetakey*{omtext}{start}
\addmetakey{omtext}{theory}
\addmetakey{omtext}{continues}
\addmetakey{omtext}{verbalizes}
\addmetakey{omtext}{subject}

The next keys handle module loading (see \[\text{\texttt{KGA18b}}\]).

\% \textbf{\texttt{\textbackslash define\texttt{\textbackslash key}}}\texttt{(omtext)}\% \texttt{\textbackslash require} \texttt{(omtext-\texttt{mh})}\% \texttt{\textbackslash RequirePackage} \texttt{(xspace)}\% \texttt{\textbackslash RequirePackage} \texttt{(modules)}\% \texttt{\textbackslash RequirePackage} \texttt{(comment)}\% \texttt{\textbackslash RequirePackage} \texttt{(mdframed)}\% \texttt{\textbackslash RequirePackage} \texttt{(latexsym)}

\% \textbf{\texttt{\textbackslash define\texttt{\textbackslash key}}}\texttt{(omtext)}\% \texttt{\textbackslash require} \texttt{(omtext)}\% \texttt{\textbackslash RequirePackage} \texttt{(xspace)}\% \texttt{\textbackslash RequirePackage} \texttt{(modules)}\% \texttt{\textbackslash RequirePackage} \texttt{(comment)}\% \texttt{\textbackslash RequirePackage} \texttt{(mdframed)}\% \texttt{\textbackslash RequirePackage} \texttt{(latexsym)}

\% \textbf{\textbackslash define\texttt{\textbackslash key}}}\texttt{(omtext)}\% \texttt{\textbackslash require} \texttt{(omtext)}\% \texttt{\textbackslash RequirePackage} \texttt{(xspace)}\% \texttt{\textbackslash RequirePackage} \texttt{(modules)}\% \texttt{\textbackslash RequirePackage} \texttt{(comment)}\% \texttt{\textbackslash RequirePackage} \texttt{(mdframed)}\% \texttt{\textbackslash RequirePackage} \texttt{(latexsym)}

\% \textbf{\textbackslash define\texttt{\textbackslash key}}}\texttt{(omtext)}\% \texttt{\textbackslash require} \texttt{(omtext)}\% \texttt{\textbackslash RequirePackage} \texttt{(xspace)}\% \texttt{\textbackslash RequirePackage} \texttt{(modules)}\% \texttt{\textbackslash RequirePackage} \texttt{(comment)}\% \texttt{\textbackslash RequirePackage} \texttt{(mdframed)}\% \texttt{\textbackslash RequirePackage} \texttt{(latexsym)}

\% \textbf{\textbackslash define\texttt{\textbackslash key}}}\texttt{(omtext)}\% \texttt{\textbackslash require} \texttt{(omtext)}\% \texttt{\textbackslash RequirePackage} \texttt{(xspace)}\% \texttt{\textbackslash RequirePackage} \texttt{(modules)}\% \texttt{\textbackslash RequirePackage} \texttt{(comment)}\% \texttt{\textbackslash RequirePackage} \texttt{(mdframed)}\% \texttt{\textbackslash RequirePackage} \texttt{(latexsym)}

8\textbf{EdNOTE}: need an implementation for \LaTeX{}XML

7
We define this macro, so that we can test whether the display key has the value flow.

We define a switch that allows us to see whether we are inside an omtext environment or a statement. It will be used to give better error messages for inline statements.

The omtext environment is different, it does not have a keyword that marks it. Instead, it can have a title, which is used in a similar way. We redefine the \lec macro so the trailing \par does not get into the way.

For the moment, we do disregard the most of the keys.

4.3 Phrase-level Markup
4.4 Declarations (under development)

The declaration macros are still under development (i.e. the macros) are still under development and may change at any time. Currently they are completely empty.

4.5 Block-Level Markup

The line end comment macro makes sure that it will not be forced on the next line unless necessary.

9EdNote: document above
10EdNote: document above
The actual appearance of the line end comment is determined by the `\@lec` macro, which can be customized in the document class. The basic one here is provided so that it is not missing.

```latex
\providecommand{\@lec}[1]{(#1)}
\def\@lec#1{\strut\hfil\strut\null\nobreak\hfill\@lec{#1}}
\def\lec#1{\@lec{#1}\par}
```

We set up a special treatment for including graphics to respect the intended OMDoc document structure. The main work is done in the transformation stylesheet though.

```latex
\newcommand\mygraphics[2][]{\includegraphics[#1]{#2}}
\newcommand\mycgraphics[2][]{\begin{center}\mygraphics[#1]{#2}\end{center}}
\newcommand\mybgraphics[2][]{\fbox{\mygraphics[#1]{#2}}}
\newcommand\mycbgraphics[2][]{\begin{center}\fbox{\mygraphics[#1]{#2}}\end{center}}
```

### 4.6 Index Markup

these are the main internal indexing commands – dividing them into four macros is awful, but I did not get list processing running. It makes sure that the modules necessary for interpreting the math in the index entries are loaded. If the `loadmodules` key is given, we import the module we are in otherwise all the currently imported modules. We do not have to require the module files, since the index is a the end of the document. If the `at` key is given, then we use that for sorting in the index.

```latex
\addmetakey{omdoc@index}{at}
\addmetakey[false]{omdoc@index}{loadmodules}[true]
\newcommand{omdoc@indexi}[2][]{\ifindex\metasetkeys{omdoc@index}{#1}\@bsphack\begingroup\@sanitize\protected@write\@indexfile{}{\string\indexentry{\ifx\omdoc@index@at\@empty\else\omdoc@index@at @\fi\ifx\omdoc@index@loadmodules\@true\string\withusedmodules{\@ifundefined{mod@id}\used@modules\mod@id}{#2}\else #2\fi% loadmodules\thepage}}\endgroup\@esphack\fi}%ifindex
\newcommand{omdoc@indexii}[3][]{\ifindex\metasetkeys{omdoc@index}{#1}\@bsphack\begingroup\@sanitize\protected@write\@indexfile{}{\string\indexentry{\ifx\omdoc@index@at\@empty\else\omdoc@index@at @\fi\ifx\omdoc@index@loadmodules\@true\string\withusedmodules{\@ifundefined{mod@id}\used@modules\mod@id}{\@ifundefined{mod@id}\used@modules\mod@id}{#2}\else #2!#3\fi% loadmodules\thepage}}\endgroup\@esphack\fi}%ifindex
\newcommand{omdoc@indexiii}[4][]{\ifindex\@bsphack\begingroup\@sanitize\protected@write\@indexfile{}{\string\indexentry{\ifx\omdoc@index@at\@empty\else\omdoc@index@at @\fi\ifx\omdoc@index@loadmodules\@true\string\withusedmodules{\@ifundefined{mod@id}\used@modules\mod@id}{\@ifundefined{mod@id}\used@modules\mod@id}{\@ifundefined{mod@id}\used@modules\mod@id}{#2}\else #2!#3!#4\fi% loadmodules\thepage}}\endgroup\@esphack\fi}%ifindex
```

```
Now, we make two interface macros that make use of this:

\*indi*
4.7 Miscellaneous

Some shortcuts that use math symbols but are not mathematical at all; in particular, they should not be translated by \LaTeX.

\newcommand\hateq{\ensuremath{\widehat=}\xspace}
\newcommand\hatequiv{\ensuremath{\widehat\equiv}\xspace}
\ifundef{\ergo}{\newcommand\ergo{\ensuremath{\leadsto}\xspace}}{\renewcommand\ergo{\ensuremath{\leadsto}\xspace}}
\newcommand{\reflect@squig}[2]{\reflectbox{$\m@th#1\rightsquigarrow$}}
\newcommand{\ogre}{\ensuremath{\mathrel{\mathpalette{\reflect@squig}{\relax}}}\xspace}
\newcommand{\notergo}{\ensuremath{\not\leadsto}}
\newcommand{\notogre}{\ensuremath{\not\mathrel{\mathpalette{\reflect@squig}{\relax}}}\xspace}

4.8 Deprecated Functionality

In this section we centralize old interfaces that are only partially supported any more.

\*def*
\newcommand{\indextoo}[2]{\indi[#1]{#2}}\PackageWarning{omtext}{\protect\indextoo\space is deprecated, use \protect\indi\space instead}
\newcommand{\indexalt}[2]{\aindi[#1]{#2}}\PackageWarning{omtext}{\protect\indextoo\space is deprecated, use \protect\aindi\space instead}
\newcommand{\twintoo}[3]{\indii[#1]{#2}{#3}}\PackageWarning{omtext}{\protect\twintoo\space is deprecated, use \protect\indii\space instead}
\newcommand{\twinalt}[3]{\aindii[#1]{#2}{#3}}\PackageWarning{omtext}{\protect\twintoo\space is deprecated, use \protect\aindii\space instead}
\newcommand{\atwintoo}[4]{\indiii[#1]{#2}{#3}{#4}}\PackageWarning{omtext}{\protect\atwintoo\space is deprecated, use \protect\indiii\space instead}
\newcommand{\atwinalt}[4]{\aindiii[#1]{#2}{#3}{#4}}\PackageWarning{omtext}{\protect\atwintoo\space is deprecated, use \protect\aindiii\space instead}
Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Abelian group, 6 Abelian, 6
### Change History

**v0.4**  
General: added index markup ... 1  
 augmenting the index macros with optional values ... 1

**v0.6**  
General: removing `ttin emin` and friends, they were almost never used.  
remnanig `myindex` to `omdoc index`, `twin` to `twin`, and `atwin` to `atwin` to make them package-local ... 1

**v0.7**  
General: changing blockquote to `sblockquote` and `inlinequote` similarly ... 1

**v0.9**  
General: separated out from `omdoc.dtx` ... 1

**v1.0**  
General: fixing typos ... 1

**v1.1**  
General: changing `\textleadsto` to `\ergo` and defining inverse `\ogre` ... 1  
moving MathHub support out to separate package ... 1

### References


[sTeX] KWARC/sTeX. URL: [https://github.com/KWARC/sTeX](https://github.com/KWARC/sTeX) (visited on 05/15/2015).