smglom.cls/sty: Semantic Multilingual Glossary for Math

Michael Kohlhase
Jacobs University, Bremen
http://kwarc.info/kohlhase

April 26, 2014

Abstract

The smglom package is part of the sTEX collection, a version of TEX/LATEX that allows to markup TEX/LATEX documents semantically without leaving the document format, essentially turning TEX/LATEX into a document format for mathematical knowledge management (MKM).

This package supplies an infrastructure for writing OMDoc glossary entries.

Contents

1 Introduction

2 The User Interface

2.1 Package and Class Options

`smglom.cls` accepts all options of the `omdoc.cls` and `article.cls` and just passes them on to these.

3 Implementation: The SMGloM Class

3.1 Class Options

To initialize the `smglom` class, we pass on all options to `omdoc.cls`

```
⟨*cls⟩
\DeclareOption*{\PassOptionsToClass{\CurrentOption}{omdoc}}
\ProcessOptions
⟨/cls⟩
⟨*ltxml.cls | ltxml.sty⟩
# -*- CPERL -*-
package LaTeXML::Package::Pool;
use strict;
use LaTeXML::Package;
DeclareOption(undef,sub {PassOptions('article','cls',ToString(Digest(T_CS('\CurrentOption'))));
ProcessOptions();
⟨/ltxml.cls | ltxml.sty⟩
```

We load `omdoc.cls`, and the desired packages. For the LaTeXML bindings, we make sure the right packages are loaded.

```
⟨*cls⟩
\LoadClass{omdoc}
\RequirePackage{smglom}
⟨/cls⟩
⟨*sty⟩
\RequirePackage{amstext}
\RequirePackage{modules}
\RequirePackage{dcm}
\RequirePackage{statements}
\RequirePackage{sproof}
\RequirePackage{cmath}
\RequirePackage{presentation}
\RequirePackage{amsfonts}
⟨/sty⟩
⟨*ltxml.cls⟩
LoadClass('omdoc');
RequirePackage('smglom');
⟨/ltxml.cls⟩
⟨*ltxml.sty⟩
RequirePackage('amstext');
RequirePackage('modules');
RequirePackage('dcm');
RequirePackage('statements');
RequirePackage('cmath');
RequirePackage('presentation');
RequirePackage('amsfonts');
⟨/ltxml.sty⟩
```

3.2 For Module Definitions

`gimport` just a shortcut

```
⟨ltxml.sty⟩RawTeX('
⟨*sty | ltxml.sty⟩
\newcommand\gimport[2][]{\def\@test{#1}%
\edef\mh@@repos{\mh@currentrepos}%
\ifx\@test\@empty\importmhmodule[repos=\mh@@repos,ext=tex,path=#2]{#2}%
\else\importmhmodule[repos=#1,ext=tex,path=#2]{#2}\fi
\mhcurrentrepos\mh@@repos\ignorespaces}
```

`guse` just a shortcut

```
\newcommand\guse[2][]{\def\@test{#1}%
\edef\mh@@repos{\mh@currentrepos}%
\ifx\@test\@empty\usemhmodule[repos=\mh@@repos,ext=tex,path=#2]{#2}%
\else\usemhmodule[repos=#1,ext=tex,path=#2]{#2}\fi
\mhcurrentrepos\mh@@repos\ignorespaces}
```

`gadopt` just a shortcut

```
\newcommand\gadopt[2][]{\def\@test{#1}%
\edef\mh@@repos{\mh@currentrepos}%
\ifx\@test\@empty\adoptmhmodule[repos=\mh@@repos,ext=tex,path=#2]{#2}%
\else\adoptmhmodule[repos=#1,ext=tex,path=#2]{#2}\fi
\mhcurrentrepos\mh@@repos\ignorespaces}
```

`gview` The `gview` environment is just a layer over the `view` environment with the keys suitably adapted.

```
\newenvironment{gview}[3][]{\metasetkeys{mhview}{#1}\def\@test{#1}%
\edef\from@repos{\ifx\mhview@fromrepos\@empty\mh@currentrepos\else\mhview@fromrepos\fi}%
\edef\to@repos{\ifx\mhview@torepos\@empty\mh@currentrepos\else\mhview@torepos\fi}%
\ifx\@test\@empty%
\begin{mhview}[fromrepos=\from@repos,frompath=#2,torepos=\to@repos,topath=#3,ext=tex]{#2}{#3}%
\else%
\begin{mhview}[fromrepos=\from@repos,frompath=#2,torepos=\to@repos,topath=#3,ext=tex]{#2}{#3}%
\fi}
{\end{mhview}}
⟨/sty | ltxml.sty⟩
⟨ltxml.sty⟩');
```

`symbol` has a starred form for primary symbols.

```
⟨*sty⟩
\def\symbol{\@ifstar{\@symbol}{\@symbol@star}}
\def\@symbol#1{\if@importing\else Symbol: \textsf{#1}\fi}
\def\@symbol@star#1{\if@importing\else Primary Symbol: \textsf{#1}\fi}
⟨/sty⟩
⟨*ltxml.sty⟩
DefConstructor('\symbol OptionalMatch:* {}',
     "<omdoc:symbol ?#1(role='primary')(role='secondary') name='#2'/>");
⟨/ltxml.sty⟩
```

*nym

```
77 ⟨*sty⟩
78 \newcommand\hypernym[3][]{\if@importing\else\par\noindent #2 is a hypernym of #3\fi}
79 \newcommand\hyponym[3][]{\if@importing\else\par\noindent #2 is a hyponym of  #3\fi}
80 \newcommand\meronym[3][]{\if@importing\else\par\noindent #2 is a meronym of #3\fi}
81 ⟨/sty⟩
82 ⟨*ltxml.sty⟩
83 DefConstructor('\hypernym [] {}{}',"");
84 DefConstructor('\hyponym [] {}{}',"");
85 DefConstructor('\meronym [] {}{}',"");
86 ⟨/ltxml.sty⟩
```

EdN:1

\MSC to define the Math Subject Classification, [1]

```
87 ⟨*sty⟩
88 \newcommand\MSC[1]{\if@importing\else MSC: #1\fi}
89 ⟨/sty⟩
90 ⟨*ltxml.sty⟩
91 DefConstructor('\MSC{}',"");
92 ⟨/ltxml.sty⟩
```

3.3 For Language Bindings

This functionality must be moved to the smultiling package.

gve The gve environment is just a layer over the mhviewsketch environment with the keys suitably adapted.

```
93 ⟨ltxml.sty⟩RawTeX('
94 ⟨*sty | ltxml.sty⟩
95 \newenvironment{gve}[5][]{\metasetkeys{mhview}{#1}\def\@test{#1}%
96 \edef\from@repos{\ifx\mhview@fromrepos\@empty\mh@currentrepos\else\mhview@fromrepos\fi}%
97 \edef\to@repos{\ifx\mhview@torepos\@empty\mh@currentrepos\else\mhview@torepos\fi}%
98 \ifx\@test\@empty%
99 \begin{mhviewsketch}[id=#2.#3,fromrepos=\from@repos,frompath=#2,torepos=\to@repos,topath=#3,ext
100 \else%
101 \begin{mhviewsketch}[id=#2.#3,fromrepos=\from@repos,frompath=#2,torepos=\to@repos,topath=#3,ext
102 \fi
103 \smg@select@language{#3}}
104 {\end{mhviewsketch}}
105 ⟨/sty | ltxml.sty⟩
106 ⟨ltxml.sty⟩');
```

[1] EdNote: MK: what to do for the LaTeXML side?