MMT Internals
An Ongoing Tutorial

Florian Rabe, Jonas Betzendahl

February 27, 2019

Contents

1 Overview (2018-11-28/29)

1.1 Structure

Full session on YouTube: Part 1, Part 2.

All MMT content is divided into the *structure level* and the *object level* (compare Figure 1.1). The structure level is a tree of named declarations that all have a path (*D*ocuments have a `DPath`, *M*odules have an `MPath` and Declarations have a `Globalname`). The general idea is that Documents contain lists of modules and modules contain lists of declarations. Both modules and documents can also contain other modules or documents respectively.

Narrative structure (what files are in what directories etc.) does not carry any semantics. For this, modules should be used instead.
The following is a good first overview about what forms MMT terms take (for more details on this, also see Section 2, where the object level is discussed in more detail):

- **OMS**
  (OpenMath Symbol, refers to a constant)
- **OMA**
  (OpenMath Application, takes operator/function and arguments)
- **OMBIND**
  (OpenMath Binder, binds variables)
- **OMV**
  (OpenMath Variable)
- **OMLIT**
  (OpenMath Literals)

1.2 Algorithms

The most interesting and relevant algorithms MMT offers (Simplification, Checking, Parsing) are separated along the same divide of object- and structure-level.
StructureXs take ObjectXs as arguments, to ensure modularity. Any IDE features are built on top of this.

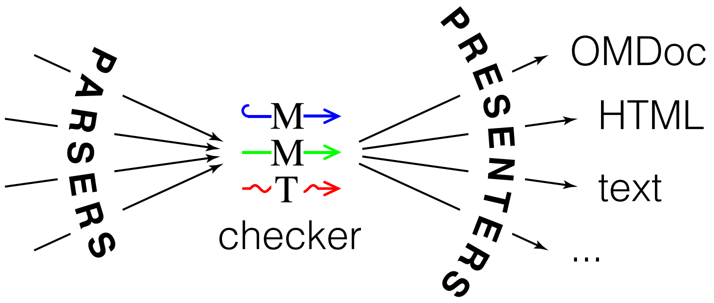

Figure 1: There can be many parsers and many presenters, but they all use the same MMT checker

2 Terms (2019-01-31)

Full session on YouTube: Link.

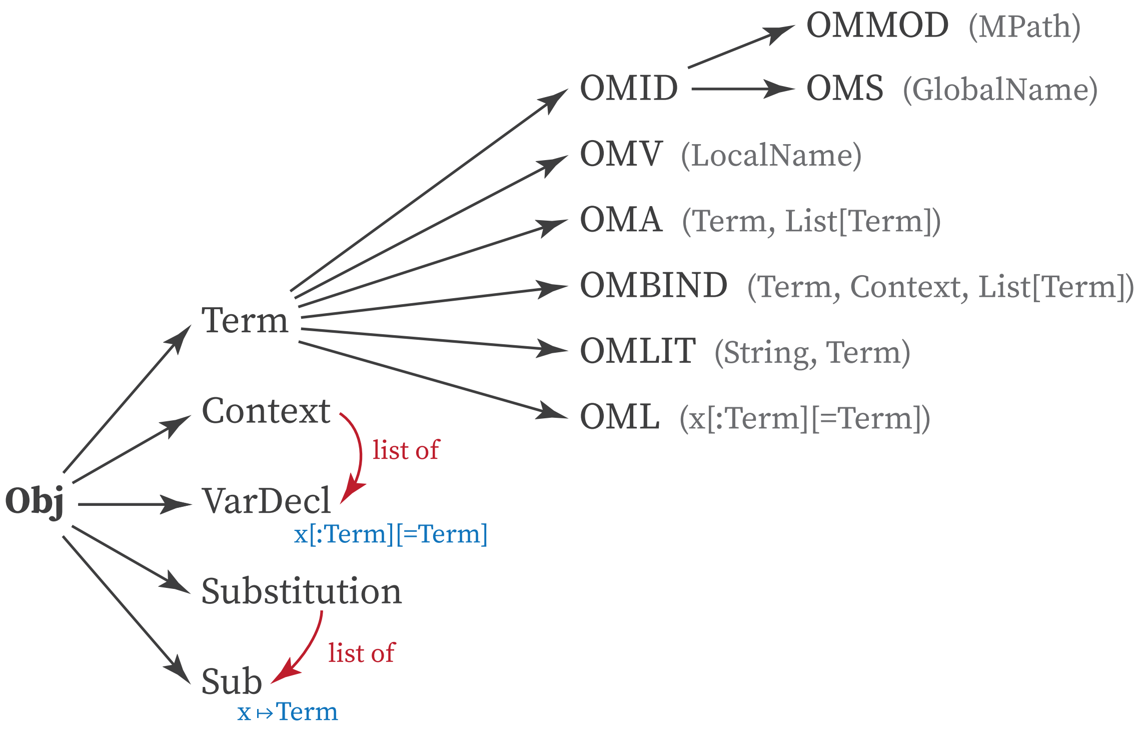

Figure 2: Overview of MMT Terms