

A Proposal for DefMPs

James Davenport

(as interpreted by Mike Dewar)

Observation: Four Kinds of OMS

1. Mathematically primitive

- e.g. `<OMS name="set" cd="set1"/>`
- FMPs *describe* the symbol

2. Primitive from an OpenMath point of view

- e.g. `<OMS name="exp" cd="transc1"/>`
- FMPs *describe* the symbol

3. Defined in terms of other OMS (i.e. logically redundant)

- e.g. `<OMS name="sin" cd="transc1"/>`
- FMPs *define* the OMS in terms of other OMS
- e.g. $\sin(x) = (\exp(ix) - \exp(-ix))/2i$

4. Defined recursively

- e.g. factorial
- FMPs *define* the OMS in terms of other OMS and itself

Basic Proposal

- Distinguish
 - *descriptive* FMPs
 - exp
 - *defining* FMPs
 - sin
 - *evaluating* FMPs
 - factorial
- e.g. by an optional attribute on the FMP element

Defining FMPs

- An FMP which can be used as the definition of a symbol
- An instance of the LHS can always be replaced by the RHS
- An OMS can have at most one
- The replacement value must not, either directly or indirectly via a chain of defining or evaluating FMPs, involve the OMS being defined

Evaluating FMPs

- Provides an algorithm for generating the value of a concrete instance of an OMS given concrete instances of its input arguments
- An OMS can have at most one
- The replacement value must, after a finite number of applications of this, and any other evaluating or defining FMPs, lead to an expression free of the symbol being defined, whenever the symbol is applied to concrete instances of the correct type(s).

Comments and Questions

- Why restrict to only one defining/evaluating FMP?
 - avoids ambiguity/questions of consistency
- Should an OMS be allowed to have both a defining and an evaluating FMP?