MathLang: experience-driven development of a mathematical language

Fairouz Kamareddine

Manuel Maarek

Joe Wells

{fairouz,mm20,jbw}@macs.hw.ac.uk.

ULTRA Group, MACS, Heriot-Watt University

Situation of Mathematics on Computers

Encoding uses

draft documents

public documents

calculations and proofs

Existing encodings

for printing and rendering

for formalization

for semantical manipulations

Our Aim

One single language which can satisfy each use to be the interface between mathematicians and computers

A framework to make the link with existing systems

MathLang

An example

From chapter 1, § 2 of E. Landau's *Foundations of Analysis* [Lan51].

Theorem 6 (Commutative Law of Addition)

$$x + y = y + x.$$

Proof Fix y, and $\mathfrak{M}$ be the set of all x for which the assertion holds.

I) We have

$$y + 1 = y',$$

and furthermore, by the construction in the proof of Theorem 4,

$$1 + y = y',$$

so that

$$1 + y = y + 1$$

and 1 belongs to $\mathfrak{M}$.

II) If x belongs to $\mathfrak{M}$, then

$$x + y = y + x,$$

Therefore

$$(x+y)' = (y+x)' = y+x'.$$

By the construction in the proof of Theorem 4, we have

$$x' + y = (x + y)',$$

hence

$$x' + y = y + x',$$

so that x' belongs to $\mathfrak{M}$. The assertion therefore holds for all x.

A LaTeX encoding

draft documents	✓
public documents	✓
calculations and proofs	✗

```
\begin{theorem}[Commutative Law of Addition]\label{theorem:6}
  $$x+y=y+x.$$
\end{theorem}
\begin{proof}
  Fix $y$, and $\mathfrak{M}$ be the set of all $x$ for which the
  assertion holds.
  \begin{enumerate}
  \item We have $$y+1=y',$$
    and furthermore, by the construction in
    the proof of Theorem~\ref{theorem:4}, $$1+y=y',$$
    so that
    $$1+y=y+1$$
    and $1$ belongs to $\mathfrak{M}$.
  \item If $x$ belongs to $\mathfrak{M}$, then $$x+y=y+x,$$
    Therefore
    $$(x+y)'=(y+x)'=y+x'.$$
    By the construction in the proof of
    Theorem~\ref{theorem:4}, we have $$x'+y=(x+y)',$$
    hence
    $$x'+y=y+x',$$
    so that $x'$ belongs to $\mathfrak{M}$.
  \end{enumerate}
  The assertion therefore holds for all $x$.
\end{proof}
```

A formal encoding in Coq

draft documents	✗
public documents	✗
calculations and proofs	✓

From Module `Arith.Plus` of Coq standard library

(`http://coq.inria.fr/`).

```
Lemma plus_sym : (n,m:nat)(n+m)=(m+n).
Proof.
Intros n m ; Elim n ; Simpl_rew ; Auto with arith.
Intros y H ; Elim (plus_n_Sm m y) ; Simpl_rew ; Auto with arith.
Qed.
```

A view of a formal encoding

draft documents	✗
public documents	✓
calculations and proofs	✓

Same Module `Arith.Plus` presented by HEΛM (`http://helm.cs.unibo.it/`).

```
DEFINITION plus_sym()
TYPE =
       "n:nat."m:nat.((n+m)=(m+n))
BODY =
       ln:nat
         .lm:nat
           .We prove ((n+m)=(m+n))
           by induction on n
               Case O
                   (plus_n_O .)   Proof of
                    we proved  (m=(m+O))
               Case (S y:nat)
                   By induction hypothesis, we have:
                   (H) ((y+m)=(m+y))
                   (f_equal . . . . . H)
                   we proved ((1+(y+m))=(1+(m+y)))
                   Rewrite (1+(m+y)) with (m+(1+y)) by (plus_n_Sm . .)
                    we proved  ((1+(y+m))=(m+(1+y)))
           we proved ((n+m)=(m+n))   Proof of
        we proved  "n:nat."m:nat.((n+m)=(m+n))
```

An OMDoc/OpenMath encoding

draft documents	✓
public documents	✓
calculations and proofs	✗

OMDoc / OpenMath

http://www.mathweb.org/omdoc/ http://www.openmath.org/

```
<assertion id="th6" type="theorem">
   <commonname> Commutative Law for Addition
   <FMP> x + y = y + x
<proof id="pr-th6" for="th6">
   <CMP> Fix y, and 𝔐 be the set of all x for which
         the assertion holds.
   <derive> <CMP> I) base case
   <derive> <CMP> II) induction hypothesis
   <conclude> <CMP> The assertion therefore holds
                    for all x.
```

MathLang

draft documents	✓
public documents	✓
calculations and proofs	✓

- MathLang describes the grammatical and reasoning structure of mathematical texts
- A *weak type system* checks MathLang documents at a grammatical level
- MathLang eventually should support *all encoding uses*

From MV to WTT to MathLang

N.G. de Bruijn's Mathematical Vernacular

> *The idea to develop MV arose from the wish to have an intermediate stage between* **ordinary mathematical presentation** *on the one hand, and* **fully coded presentation** *in Automath-like systems on the other hand.*
>
> *[dB87]*

- Variables, constants and binders
- *Line-by-line* structure
- Notions of *low-typing* and *high-typing*

From MV to WTT to MathLang

The Weak Type Theory

- WTT refined MV by assigning a unique *atomic weak type* to each text element
- A *meta-theory* describes properties of WTT documents

WTT and its meta-theory have been designed by F. Kamareddine and R. Nederpelt [NK01, KN]

From MV to WTT to MathLang

MathLang

- MathLang extends MV and WTT
- MathLang is closer to a grammatical encoding
- MathLang's development is driven by translation experiences
- MathLang's framework development intends to eventually satisfy mathematicians' needs

MathLang

Grammatical categories

- $\mathbb{T}$ terms
- $\mathbb{S}$ sets
- $\mathbb{N}$ nouns
- $\mathbb{A}$ adjectives
- $\mathbb{P}$ statements

- $\mathtt{D}$ definitions
- $\mathtt{Z}$ declarations
- Γ contexts with flags
- $\mathtt{L}$ lines
- $\mathtt{K}$ blocks
- $\mathtt{B}$ books

MathLang

Weak type checking

$\mathbb{T}$ Terms $\mathbb{S}$ Sets $\mathbb{N}$ Nouns $\mathbb{P}$ Statements Z Declarations Γ Context

Let $\mathfrak{M}$ be a set,
y and x are natural numbers,
if x belongs to $\mathfrak{M}$
then $x + y = y + x$

(idealized view of presentation form, not yet designed)

MathLang

Weak type checking

$\mathbb{T}$ Terms | $\mathbb{S}$ Sets | $\mathbb{N}$ Nouns | $\mathbb{P}$ Statements | Z Declarations | Γ Context

Let $\mathfrak{M}$ be a set,
y and x are natural numbers,
if x belongs to $\mathfrak{M}$
then $x + y$ $\Leftarrow$ **error**

MathLang

blocks flags references

Theorem 6 (Commutative Law of Addition)

$$x + y = y + x.$$

Proof Fix y, and $\mathfrak{M}$ be the set of all x for which the assertion holds.

I) We have

$$y + 1 = y',$$

and furthermore, by the construction in the proof of Theorem 4,

$$1 + y = y',$$

so that

$$1 + y = y + 1$$

and 1 belongs to $\mathfrak{M}$.

II) If x belongs to $\mathfrak{M}$, then

$$x + y = y + x,$$

Therefore

$$(x+y)' = (y+x)' = y+x'.$$

By the construction in the proof of Theorem 4, we have

$$x' + y = (x + y)',$$

hence

$$x' + y = y + x',$$

so that x' belongs to $\mathfrak{M}$. The assertion therefore holds for all x.

MathLang

An internal view of a MathLang document

$x : \mathbb{N}, y : \mathbb{N} \rhd \text{Th6}(x, y) := x + y = y + x$ (97)

Proof Theorem 6 {2.5.4}

Proof Theorem 6 part I {2.5.4.1}

$y : \mathbb{N}$

$\mathfrak{M} : \text{SET}$

$\forall_{x:\mathfrak{M}} \text{Th6}(x, y)$

$(\text{Def } +(38)) \rhd y + 1 = y'$ (98)

$\{2.5.1\} \rhd 1 + y = y'$ (99)

$(98), (99) \rhd 1 + y = y + 1$ (100)

$(100) \rhd \text{Th6}(1, y)$ (101)

$(101) \rhd 1 : \mathfrak{M}$ (102)

Proof Theorem 6 part II {2.5.4.2}

$x : \mathfrak{M}$

$\text{Th6}(x, y) \rhd x + y = y + x$ (103)

$(103) \rhd (x + y)' = (y + x)'$ (104)

$(\text{Def } +(39)) \rhd (y + x)' = y + x'$ (105)

$(104), (105) \rhd (x + y)' = y + x'$ (106)

$\{2.5.2\} \rhd x' + y = (x + y)'$ (107)

$(107), (\text{Def } +(39)) \rhd x' + y = y + x'$ (108)

$(108) \rhd \text{Th6}(x', y)$ (109)

$(109) \rhd x' : \mathfrak{M}$ (110)

$\text{Ax5}(\mathfrak{M}, (102), (110)) \rhd \mathbb{N} \subset \mathfrak{M}$ (111)

$(111) \rhd \forall_{x:\mathbb{N}} \forall_{y:\mathbb{N}} \text{Th6}(x, y)$ (112)

MathLang

Experience-driven development of MathLang

- Language description
  blocks – flags – references
- Translation
  first chapter of *Foundations of Analysis* [Lan51]
- Implementation
  type checker

MathLang

Implementation

- XML syntax

  internal representation, not for users to read/write

- Checker for weak types

  analysing the bindings and the grammatical structure

- Transformation programs

  overview of the content, structure and type information

Future Work

- Extensions of the language
- Translations of *Foundations of Analysis* [Lan51] and *The 13 Books of Euclid's Elements* [Hea56]
- Transparent integration of MathLang in the scientific text editor $\TeX_{\text{MACS}}$ `http://www.texmacs.org/`
- Annotation of OMDoc with MathLang grammatical information

Conclusion

MathLang

- Experience-driven development
- Inspired by the common mathematical language
- Mathematician-oriented framework

References

[dB87] N.G. de Bruijn. The mathematical vernacular, a language for mathematics with typed sets. In *Workshop on Programming Logic*, 1987.

[Hea56] Heath. *The 13 Books of Euclid's Elements*. Dover, 1956.

[KN] Fairouz Kamareddine and Rob Nederpelt. A refinement of de Bruijn's formal language of mathematics. To appear in Journal of Logic, Language and Information.

[Lan51] Edmund Landau. *Foundations of Analysis*. Chelsea, 1951. Translation from German by F. Steinhardt.

[NK01] Rob Nederpelt and Fairouz Kamareddine. An abstract syntax for a formal language of mathematics. In *Fourth International Tbilisi Symposium on Language, Logic and Computation*, September 2001.