## Toward an Object-Oriented Structure for Mathematical Text

#### Fairouz Kamareddine, Manuel Maarek and Joe Wells

ULTRA Group - Heriot-Watt University

July 16, 2005 Mathematical Knowledge Management 2005 International University Bremen, Germany

・ロト ・ 同ト ・ ヨト ・ ヨト

# Computerising mathematical texts

The Common Mathematical Language is

- Meticulous
- Structured
- Coherent
- Is CML reflected in current approaches of computerising Mathematics?

イロト イポト イヨト イヨト

#### Two examples From Euclid to Bourbaki

**Definition 20.** Of trilateral figures, an equilateral triangle is that which has its three sides equal, an isosceles triangle that which has two of its sides alone equal, and a scalene triangle that which has its three sides unequal.

Euclid [The 13 Books of Euclid's Elements, Book I]

**Definition 1.** A set with an associative law of composition, possessing an identity element and under which every elements is invertible, is called a group. [...] A group G is called finite if the underlying set of G is finite [...] A group [with operators] G is called commutative (or Abelian) if its group law is commutative.

N. Bourbaki [Elements of Mathematics - Algebra, volume II, Chapter I, §4]

イロト イポト イヨト イヨト

#### Two examples From Euclid to Bourbaki

**Definition 20.** Of trilateral figures, an equilateral triangle is that which has its three sides equal, an isosceles triangle that which has two of its sides alone equal, and a scalene triangle that which has its three sides unequal.

Euclid [The 13 Books of Euclid's Elements, Book I]

**Definition 1.** A set with an associative law of composition, possessing an identity element and under which every elements is invertible, is called a group. [...] A group G is called finite if the underlying set of G is finite [...] A group [with operators] G is called commutative (or Abelian) if its group law is commutative.

N. Bourbaki [Elements of Mathematics - Algebra, volume II, Chapter I, §4]

・ロト ・ 同ト ・ ヨト ・ ヨト

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### 

#### **ATEX**

#### \begin{definition}

Of trilateral figures, an equilateral triangle is that which has its three sides equal, an isosceles triangle that which has two of its sides alone equal, and a scalene triangle that which has its three sides unequal. \end{definition}

#### \begin{definition}

A set with an associative law of composition, possessing an identity element and under which every elements is invertible, is called a group. [...] A group \$G\$ is called finite if the underlying set of \$G\$ is finite [...] A group [with operators] \$G\$ is called commutative (or Abelian) if its group law is commutative. \end{definition}

Visual representation

Difficult semantic recognition

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### Semantic markup languages MathML, OpenMath, OMDoc

#### OpenMath/OMDoc

- Flexible
- Difficult semantic recognition due to the mixture of natural language and structural and symbolic XML
- Manage embedded formal content

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

# Full formalisation

Theorem provers

## Our goal differs from full formalisation.

We want to provide a control over presentation and phrasing of the semantic structure. Most mathematical texts are unlikely to be formalized, but might well benefit from computerisation.

Procedural style – such as Coq, Isabelle

- Fully formalised
- Requires expertise
- Formalisation that may not reflect the CML text

Declarative style - such as Mizar

- Fully formalised
- Requires expertise and the Mizar Mathematical Library
- Syntax mimics natural language
- Formal Proof Sketch (a lighter version of Mizar)

<ロ> (四) (四) (三) (三) (三)

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

Computerising the mathematical vernacular N.G. de Bruijn's MV – WTT – MathLang-WTT – MathLang

N.G. de Bruijn's Mathematical Vernacular A language with substantives, adjectives and flags

The Weak Type Theory A type system for MV with weak types (TERM, NOUN, ADJ, SET, STAT, LINE and BOOK)

MathLang-WTT A practical evaluation of MV and WTT

- Extends WTT with FLAGS and BLOCKS
- Automates type checking
- Has been used to translate existing CML texts
- Proposes various output-views faithful to CML

MathLang's approach to computerise mathematical texts is to:

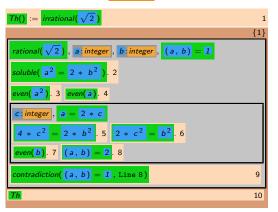
- Capture, in a first layer, the grammatical structure of the text
- Represent, in later gradual layers, the semantic and logic of the text

· □ > · (司 > · (日 > · (日 > · )

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### Computerising the mathematical vernacular MathLang-WTT – output-view (Example from F. Wiedijk's comparison)

T Terms S Sets Ν Nouns A Adjectives P Statements Z Declarations Γ Contexts L Lines F Flags K Blocks B Books



- Symbolic view
- CML view of symbols

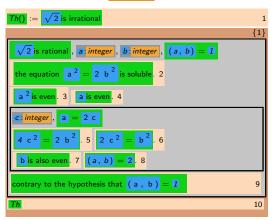
イロト イヨト イヨト イヨト

 CML view of the document

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### Computerising the mathematical vernacular MathLang-WTT – output-view (Example from F. Wiedijk's comparison)

T Terms S Sets Ν Nouns A Adjectives P Statements Z Declarations Γ Contexts L Lines F Flags K Blocks B Books



- Symbolic view
- CML view of symbols

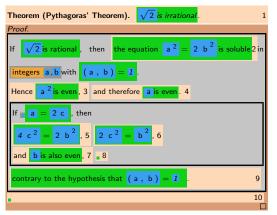
<ロ> (四) (四) (注) (注) (三) (三)

 CML view of the document

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### Computerising the mathematical vernacular MathLang-WTT – output-view (Example from F. Wiedijk's comparison)





- Symbolic view
- CML view of symbols

・ロト ・ 日 ・ ・ ヨ ・ ・ 日 ・ ・

 CML view of the document

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### MathLang-WTT Encodings of Euclid's and Bourbaki's examples?

How to faithfully encode a triangle and its sides, a group and its law in MathLang-WTT?

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

#### MathLang-WTT Encodings of Euclid's and Bourbaki's examples?

How to faithfully encode a triangle and its sides, a group and its law in MathLang-WTT?

triangle and side, group and law as constants of type NOUN.

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

### MathLang-WTT Encodings of Euclid's and Bourbaki's examples?

How to faithfully encode a triangle and its sides, a group and its law in MathLang-WTT?

triangle and side, group and law as constants of type NOUN.

How to encode the intrinsic relation between a triangle and its lines and between a group and its law?

Mathematical word processing Semantic markup languages Full formalisation Computerising the mathematical vernacular

### MathLang-WTT Encodings of Euclid's and Bourbaki's examples?

How to faithfully encode a triangle and its sides, a group and its law in MathLang-WTT?

triangle and side, group and law as constants of type NOUN.

How to encode the intrinsic relation between a triangle and its lines and between a group and its law?

- By parametrising triangle and group with sides and law
   → Constraining & not flexible
- By using a statement "has".

```
has(triangle,line1); has(triangle,line2); has(triangle,line3)
has(group,law)
```

 $\rightarrow$  Verbose & not reliable

# Obviously, this kind of fundamental description of mathematical objects needed improvement

**Object-oriented approach** Examples Multi adjective refinements

# Abstraction with nouns and adjectives

**•** Back to N.G. de Bruijn's informal definitions.

MV's substantives (MathLang-WTT's nouns)

MV's adjectives (MathLang-WTT's adjectives)

MV's names (MathLang-WTT's terms)

(D) (A) (A)

# Abstraction with nouns and adjectives

- Back to N.G. de Bruijn's informal definitions.
- Analogy with Object-oriented programming.

#### MV's substantives (MathLang-WTT's nouns) Classes

MV's adjectives (MathLang-WTT's adjectives) Mixins (functions from classes to classes)

MV's names (MathLang-WTT's terms) **Objects** 

(D) (A) (A)

# Abstraction with nouns and adjectives

- Back to N.G. de Bruijn's informal definitions.
- Analogy with Object-oriented programming.
- ▶ New design of MathLang with object-oriented features.

MV's substantives (MathLang-WTT's nouns)

Classes

Nouns as classes

MV's adjectives (MathLang-WTT's adjectives) Mixins (functions from classes to classes) Adjectives as mixins

MV's names (MathLang-WTT's terms)

Objects

Terms as objects

MKM05

F. Kamareddine, M. Maarek, J. B. Wells Toward an Object-Oriented Structure for Mathematical Text

# Abstraction with nouns and adjectives

- Back to N.G. de Bruijn's informal definitions.
- Analogy with Object-oriented programming.
- New design of MathLang with object-oriented features. Focal

#### MV's substantives (MathLang-WTT's nouns)

Classes

Nouns as classes

# MV's adjectives (MathLang-WTT's adjectives) Mixins (functions from classes to classes)

Adjectives as mixins

MV's names (MathLang-WTT's terms)

Objects

Terms as objects

collection

イロト イヨト イヨト イヨト

species

#### Abstraction with nouns and adjectives Euclid's example

**Definition 20.** Of trilateral figures, an equilateral triangle is that which has its three sides equal, an isosceles triangle that which has two of its sides alone equal, and a scalene triangle that which has its three sides unequal. Euclid [The 13 Books of Euclid's Elements, Book I]

Figure and triangle defined as nouns. Trilateral and equilateral defined as adjectives.

(日) (四) (三) (三) (三)

#### Abstraction with nouns and adjectives Euclid's example

**Definition 20.** Of trilateral figures, an equilateral triangle is that which has its three sides equal, an isosceles triangle that which has two of its sides alone equal, and a scalene triangle that which has its three sides unequal. Euclid [The 13 Books of Euclid's Elements, Book I]

Figure and triangle defined as nouns. Trilateral and equilateral defined as adjectives.

(日) (四) (三) (三) (三)

#### Abstraction with nouns and adjectives Bourbaki's example

MKM05

**Definition 1.** A set with an associative law of composition, possessing an identity element and under which every elements is invertible, is called a group. [...] A group G is called finite if the underlying set of G is finite [...] A group [with operators] G is called commutative (or Abelian) if its group law is commutative. N. Bourbaki [Elements of Mathematics - Algebra, volume II, Chapter I, §4]

Group defined as a noun. Finite and Abelian defined as adjectives.

(日) (周) (王) (王)

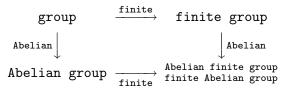
#### Abstraction with nouns and adjectives Bourbaki's example

**Definition 1.** A set with an associative law of composition, possessing an identity element and under which every elements is invertible, is called a group. [...] A group G is called finite if the underlying set of G is finite [...] A group [with operators] G is called commutative (or Abelian) if its group law is commutative. N. Bourbaki [Elements of Mathematics - Algebra, volume II, Chapter I, §4]

Group defined as a noun. Finite and Abelian defined as adjectives.

・ロト ・ 同ト ・ ヨト ・ ヨト

#### Abstraction with nouns and adjectives Multi adjective refinements



- Combine the adjectives *finite* and *Abelian* to obtain either Abelian finite group or finite Abelian group.
- In MathLang both expressions share the same type. Their meaning may differ as the statements introduced by the adjectives may overlap.
- It is possible to define an isosceles equilateral scalene triangle.

Syntax Type system Example

#### Syntax Sets, category expressions and identifiers

ident, i	=	denumerably infinite set of identifiers
label, l	=	denumerably infinite set of labels
cvar, v	=	denumerably infinite set of category variables
category, c	::=	<pre>term(exp)   set(exp)   noun(exp)   adj(exp, exp) stat   dec(category)   cvar</pre>
cident, ci	::=	

MKM05

イロン イヨン イヨン イヨン

-2

#### Syntax Steps

step, s::=phraseBasic unit|label label stepLabelling|step  $\triangleright$  stepLocal scoping| $\{\overline{step}\}$ Block

(an arrow on top of a meta-variable represents a sequence of 0 or more meta-variables)

<ロ> (四) (四) (三) (三) (三)

Syntax Type system Example

#### Syntax Phrases and expressions

phrase, p	::=	exp	
		<i>cident</i> ( <i>ident</i> ) := <i>exp</i>	Definition
	i	$ident(\overrightarrow{exp}) := exp$	Definition by matching case
	Í	ident ≪ cident	Sub-noun and adjective statement
exp, e	::=	cident(ēxp)	Instance
		ident(category) : exp	Elementhood declaration
		ident(category) : category	Declaration
	Í	Noun { <i>step</i> }	Noun
		Adj( <i>exp</i> ) { <i>step</i> }	Adjective
		exp exp	Refinement
		Up <i>exp</i>	Noun lifting
		$\texttt{self} \mid \texttt{super}$	Self and super
		ref <i>label</i>	Referencing

MKM05

▲ロト ▲圖ト ▲屋ト ▲屋ト

-1

Syntax Type system Example

#### Type system Rules for steps

$$\frac{\vdash s_{1} \colon Step \qquad s_{1} \vdash s_{2} \wr Step \qquad \{s_{1}; s_{2}\} \vdash \{\overrightarrow{s}\} \wr Step}{s_{1} \vdash \{s_{2}; \overrightarrow{s}\} \wr Step} \text{ STEP-COMPOSITION}$$

$$\frac{\vdash s \wr Step \qquad s \vdash s' \wr Step \qquad \{s; s'\} \vdash s'' \wr Step}{s \vdash s' \triangleright s'' \wr Step} \text{ LOCAL-SCOPING}$$

$$\frac{\vdash s \wr Step \qquad s \vdash p \wr Stat/Dec(t)/Def(t)}{s \vdash p \wr Step} \text{ ATOMIC-STEP}$$

$$\frac{\vdash \{\} \wr Step}{\vdash \{\} \wr Step} \text{ EMPTY-STEP}$$

MKM05

・ロト ・ 日 ・ ・ ヨ ・ ・ モ ・

Syntax Type system Example

#### Type system Rules for noun and adjective expressions

$$\begin{array}{c} \vdash s \mathrel{\bullet} Step \qquad \{s; \ self : \ Term(T)\} \vdash s' \mathrel{\bullet} Step \\ \hline \forall i \in I(s'), \{s; \ self : \ Term(T); \ s'\} \vdash i \mathrel{\bullet} T(i) \\ \hline \hline s \vdash \operatorname{Noun} \{s'\} \mathrel{\bullet} Noun(T) \end{array} \text{ NOUN}$$

$$\begin{array}{c} \vdash s \mathrel{\bullet} Step \qquad s \vdash e \mathrel{\bullet} Noun(T) \\ T \leq T' \qquad \{s; super : Term(T); self : Term(T')\} \vdash s' \mathrel{\bullet} Step \\ \forall i \in I(s'), \{s; super : Term(T); self : Term(T'); s'\} \vdash i \mathrel{\bullet} T'(i) \\ \hline s \vdash \operatorname{Adj}(e) \{s'\} \mathrel{\bullet} Adj(T, T') \end{array} ADJ$$

$$\begin{array}{c|c} \vdash s & \text{Step} & s \vdash e_1 & \text{Adj}(T_1, T_1') \\ \hline s \vdash e_2 & \text{Noun}(T_2)/\text{Set}(T_2)/\text{Term}(T_2) & T_1 \leq T_2 \\ \hline s \vdash e_1e_2 & \text{Noun}(T_1' \uplus T_2)/\text{Set}(T_1' \uplus T_2)/\text{Term}(T_1' \uplus T_2) \end{array}$$
 REFINEMENT

・ロト ・ 日ト ・ モト・・ モト・

Syntax Type system Example

#### Type system Example of typing – Euclid's example

 Term Terms
 Set Sets
 Noun
 Adj
 Adjectives
 Stat
 Statements

 Def
 Definition
 Step
 Local scopings ▷
 Step
 Blocks { }

Definition 20. Of trilateral figures,

an equilateral triangle is that which has its three sides equal,

an isosceles triangle that which has two of its sides alone equal,

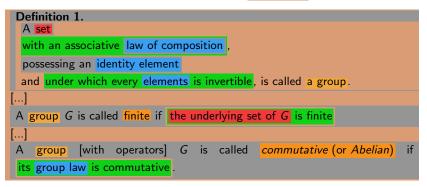
and a scalene triangle that which has its three sides unequal .

<ロ> (四) (四) (三) (三) (三)

Syntax Type system Example

#### Type system Example of typing – Bourbaki's example

Term Terms Set Sets Noun Nouns Adj Adjectives Stat Statements
Def Definition Step Local scopings ▷ Step Blocks { }



イロト イヨト イヨト イヨト

# Future work

- $\blacktriangleright$  Development of a user interface for MathLang based on  $T_{E}\!X_{_{\rm MACS}}$
- Adding semantical and logical annotations (with Krzysztof Retel)
- Continue the translations of Euclid's *Elements* and of E. Landau's *Foundation of Analysis*
- Adapt MathLang's weak typing for OpenMath/OMDoc

< 日 > ( 一) > ( 二) > ( ( 二) > ( ( 二) > ( ( L) > ( ( L)

Future work Conclusion

# Conclusion

We saw how the experience-driven development of MathLang led to

- Turning nouns into classes
- Turning adjectives into mixins

MathLang provides an expressive encoding for computerising the symbolic and natural language parts of mathematical text

< 日 > ( 一) > ( 二) > ( ( 二) > ( ( 二) > ( ( L) > ( ( L)