Mathematical Components

Assia Mahboubi

INRIA Microsoft Research Joint Centre
INRIA Saclay – Île-de-France
École Polytechnique, Palaiseau

April 12th, 2011

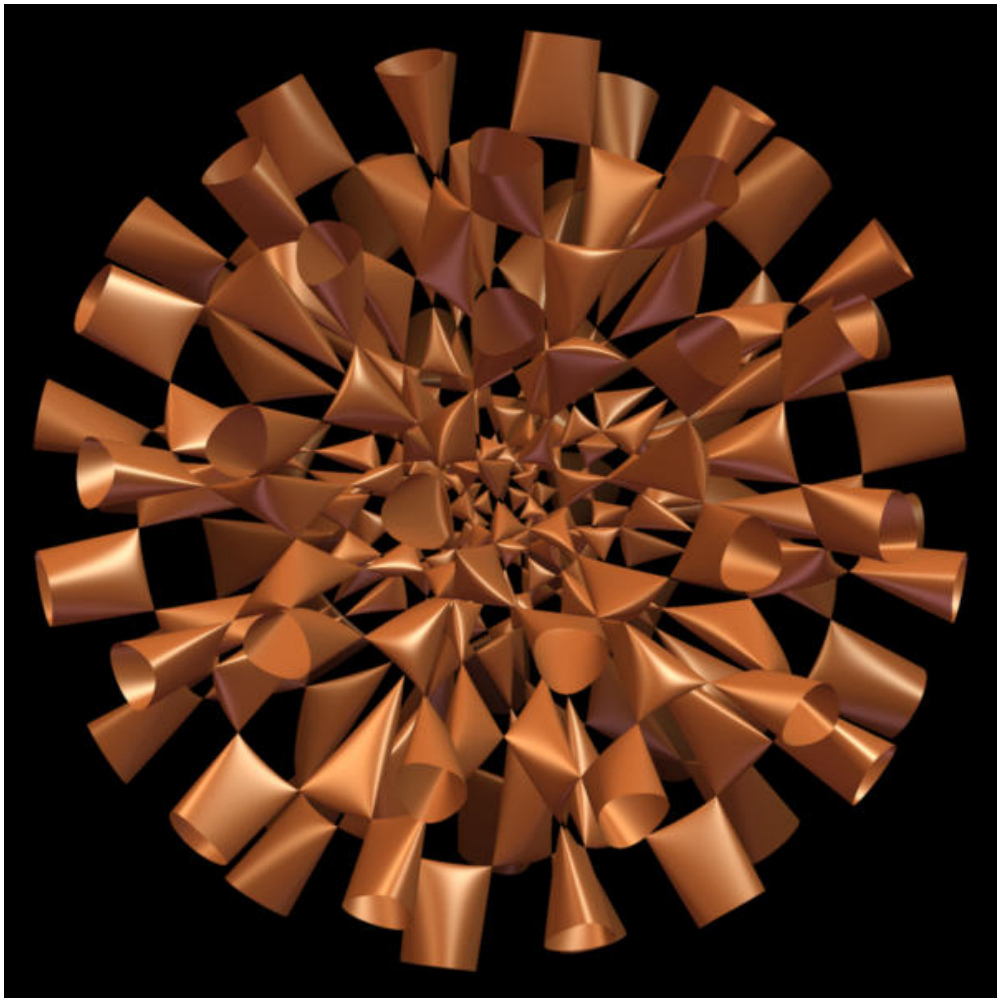
The Mathematical Components team

- Leader: G. Gonthier (Microsoft Research)
- Current members: Y. Bertot, G. Cano, C. Cohen, M. Denès, F. Garillot, A.M., R. O'Connor, L. Rideau, E. Tassi, L. Théry
- Past members: A. Asperti, J. Avigad, S. Le Roux, S. Ould Biha, I. Pasca, G. Melquiond, S. McLaughlin, R. Zumkeller

Use computers to check mathematical proofs.

Why do we write proofs?

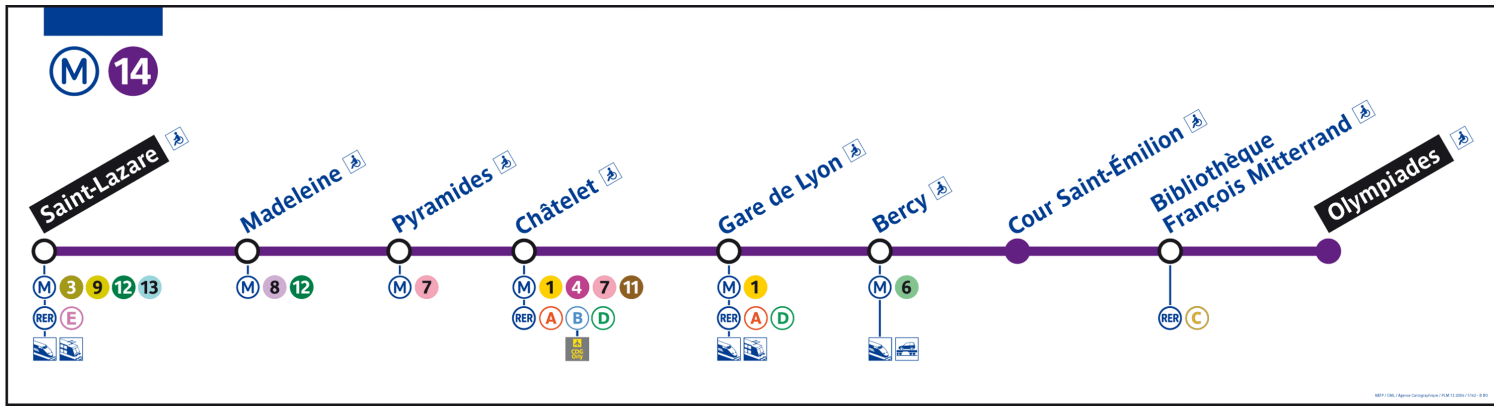
To increase mathematical knowledge.



To understand the behavior of mathematical objects.

Why do we write proofs?

To verify large and technical properties of software, machines,...



That are not scientific breakthroughs but beyond reach of human checking.

Why is a proof convincing?

Proof by intimidation:

Why is a proof convincing?

Proof by semi-formal description:

Since $\mathfrak{A}/\mathfrak{U} = (\mathfrak{K}/\mathfrak{U})(\mathfrak{B}/\mathfrak{U})$, it follows that:

$$\mathfrak{A} = \mathfrak{K}\mathfrak{B} = \mathfrak{D}\mathfrak{U}\mathfrak{B} = \mathfrak{D}\mathfrak{B}$$

Also $\mathfrak{K} \cap \mathfrak{B} = \mathfrak{U}$ and $\mathfrak{U}\mho^p(\mathfrak{K}) \cap \mathfrak{D} = \mho^p(\mathfrak{K})$, so

$$\mathfrak{D} \cap \mathfrak{B} = \mathfrak{D} \cap \mathfrak{K} \cap \mathfrak{B} = \mathfrak{D} \cap \mathfrak{U} = \mathfrak{D} \cap \mathfrak{U}\mho^p(\mathfrak{K}) \cap \mathfrak{U} = \mho^p(\mathfrak{K}) \cap \mathfrak{U} = 1$$

Thus $\mathfrak{A} = \mathfrak{D} \times \mathfrak{B}$. Then $\mathfrak{K} = \mathfrak{D} \times (\mathfrak{B} \times \mathfrak{K}) = \mathfrak{D} \times \mathfrak{U}$, so $\mathfrak{D} \simeq \mathfrak{K}/\mathfrak{U}$ is homocylic and $\mathfrak{D}/\Phi(\mathfrak{D})$ is $\mathfrak{X}$-irreducible. Also $\mathfrak{D} \neq 1$ so the inductive hypothesis may be applied to $\mathfrak{B}$ and the theorem follows at once.

Finite groups, vol 2. VIII.5.9 - Huppert Blackburn (excerpt)

Why is a proof convincing?

Proof by semi-formal description:

Since $\mathfrak{A}/\mathfrak{U} = (\mathfrak{K}/\mathfrak{U})(\mathfrak{B}/\mathfrak{U})$, it follows that:

$$\mathfrak{A} = \mathfrak{K}\mathfrak{B} = \mathfrak{D}\mathfrak{U}\mathfrak{B} = \mathfrak{D}\mathfrak{B}$$

Also $\mathfrak{K} \cap \mathfrak{B} = \mathfrak{U}$ and $\mathfrak{U}\mho^p(\mathfrak{K}) \cap \mathfrak{D} = \mho^p(\mathfrak{K})$, so

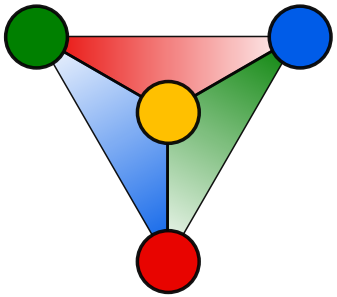
$$\mathfrak{D} \cap \mathfrak{B} = \mathfrak{D} \cap \mathfrak{K} \cap \mathfrak{B} = \mathfrak{D} \cap \mathfrak{U} = \mathfrak{D} \cap \mathfrak{U}\mho^p(\mathfrak{K}) \cap \mathfrak{U} = \mho^p(\mathfrak{K}) \cap \mathfrak{U} = 1$$

Thus $\mathfrak{A} = \mathfrak{D} \times \mathfrak{B}$. Then $\mathfrak{K} = \mathfrak{D} \times (\mathfrak{B} \times \mathfrak{K}) = \mathfrak{D} \times \mathfrak{U}$, so $\mathfrak{D} \simeq \mathfrak{K}/\mathfrak{U}$ is homocylic and $\mathfrak{D}/\Phi(\mathfrak{D})$ is $\mathfrak{X}$-irreducible. Also $\mathfrak{D} \neq 1$ so the inductive hypothesis may be applied to $\mathfrak{B}$ and the theorem follows at once.

Finite groups, vol 2. VIII.5.9 - Huppert Blackburn (excerpt)

Why is a proof convincing?

Computers can help for large computations, and massive data processing but also for our first category of proofs.



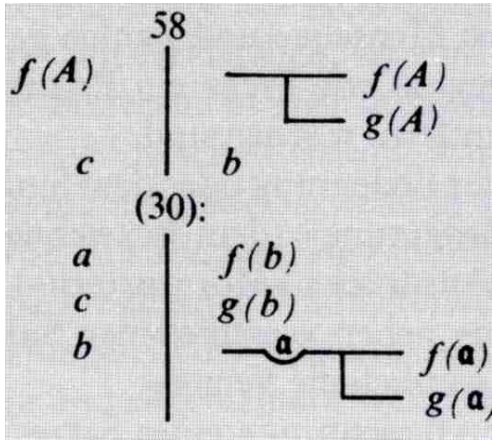
The Four Color Theorem

The Kepler conjecture

Why is a proof convincing?

Computers can help, yet would you always trust it?

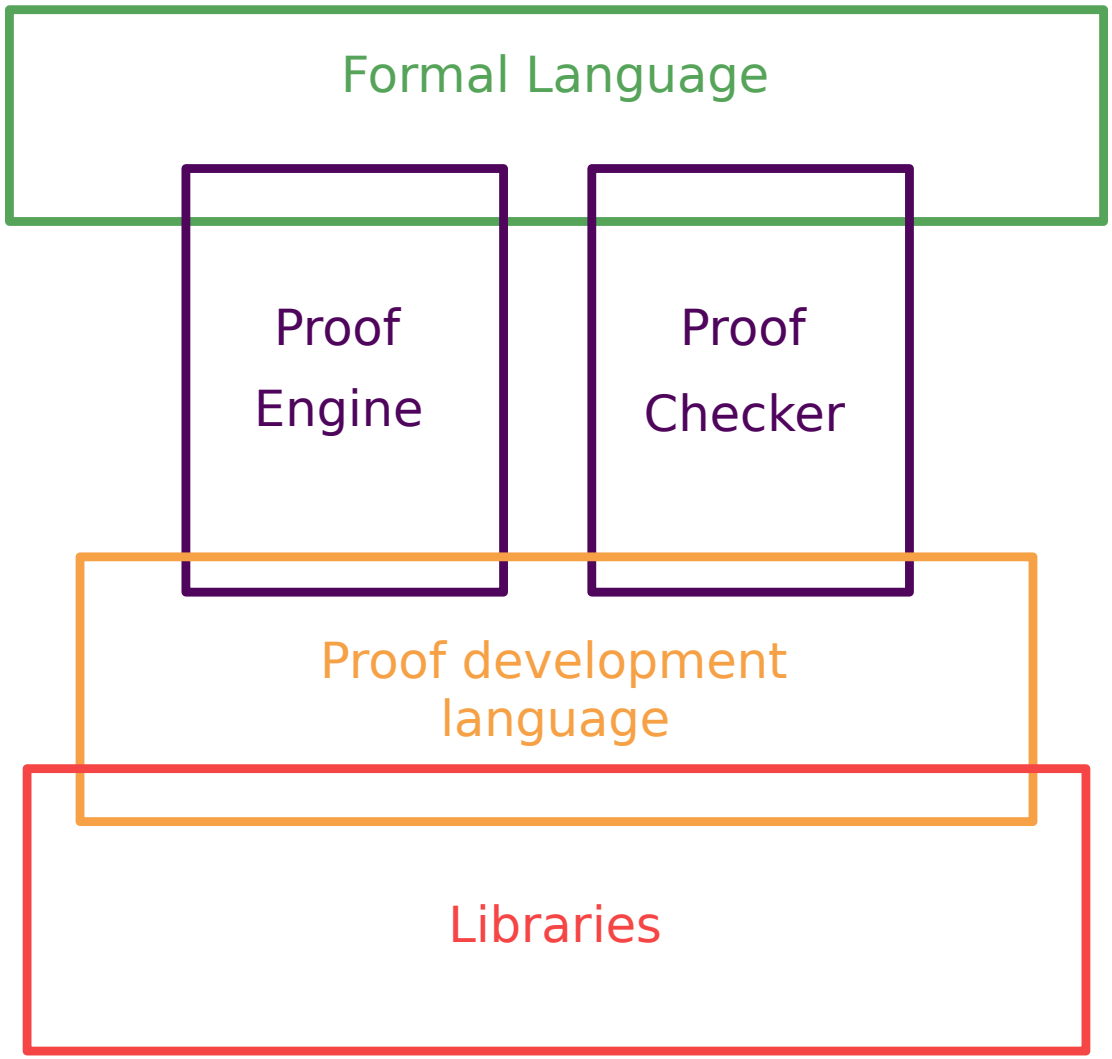
Trust a computer to check your proofs



- Use formal logics as assembly code to describe statements and proofs.
- Use a proof assistant to develop and to check this code.

Why can we trust the machine this time?

Proof assistants



Formal software engineering

What you need:

- A good language
- Appropriate data-strucutures
- Relevant interfaces
- Modular libraries

And a challenging benchmark to test the success of the approach.

Benchmark: The Odd Order Theorem

In 2005, G. Gonthier and B. Werner completed the verification of a proof of the Four Color Theorem Theorem.

Benchmark: The Odd Order Theorem

In 2005, G. Gonthier and B. Werner completed the verification of a proof of the Four Color Theorem Theorem.

So what?

Benchmark: The Odd Order Theorem

In 2005, G. Gonthier and B. Werner completed the verification of a proof of the Four Color Theorem Theorem.

So what? What about truly difficult mathematics?

Benchmark: The Odd Order Theorem

Theorem (Feit - Thompson (1963)) :

Every simple group of odd order is solvable.

otherwise said

Every simple group of odd order is cyclic.

Benchmark: The Odd Order Theorem

Theorem (Feit - Thompson (1963)) :

Every simple group of odd order is solvable.
otherwise said
Every simple group of odd order is cyclic.

- Original published proof: one entire volume of the Pacific Journal of Mathematics

Benchmark: The Odd Order Theorem

Theorem (Feit - Thompson (1963)) :

Every simple group of odd order is solvable.
otherwise said
Every simple group of odd order is cyclic.

- Original published proof: one entire volume of the Pacific Journal of Mathematics
- A collective simplification work: two entire volumes of London Math. Society Lecture Notes.

Benchmark: The Odd Order Theorem

Theorem (Feit - Thompson (1963)) :

Every simple group of odd order is solvable.
otherwise said
Every simple group of odd order is cyclic.

- Original published proof: one entire volume of the Pacific Journal of Mathematics
- A collective simplification work: two entire volumes of London Math. Society Lecture Notes.
- (Wikipedia 12/04/2011) "It takes a professional group theorist about a year of hard work to understand the proof completely."

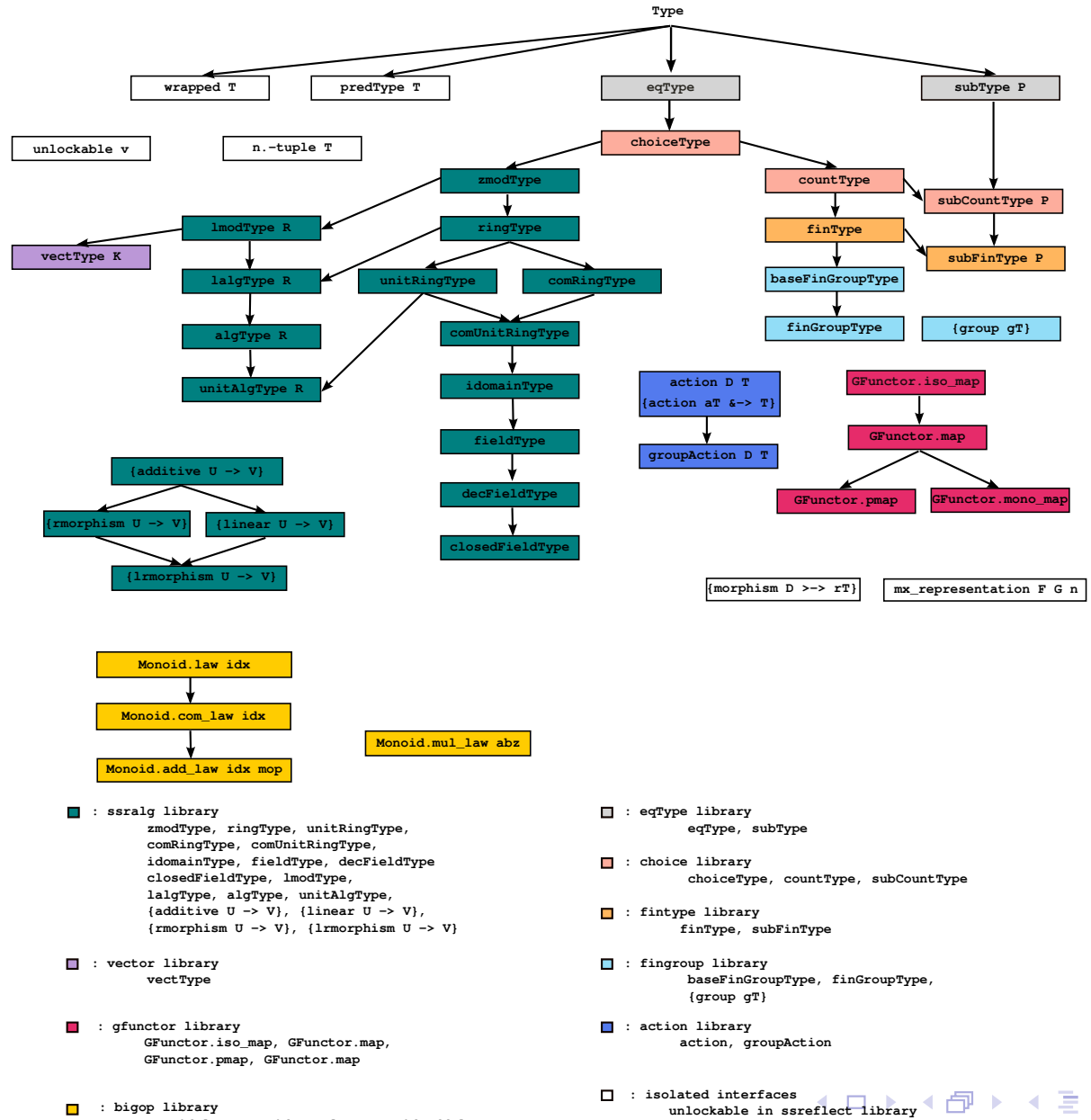
Benchmark: The Odd Order Theorem

- A first step toward the Classification of Finite Groups
  (aka. the Monster theorem)
- A combination of a variety of mathematical theories:
  linear algebra, Galois theory, characters,...
- A large, intricate, well known and controversial proof
- No computational aspect.

Proof assistant and proof development language

- The Coq system:
  a type theory based proof assistant developed at Inria since the 80's.
- The Ssreflect package:
  an extension developed at the Inria Microsoft Joint Centre since 2006.

Libraries: hierarchy of mathematical structures



Libraries: Everyday mathematics syntax, implicit semantics

For any matrix $M \in M_n(F)$,

$$det(M) := \sum_{s \in S_n} (-1)^{\epsilon_s} \prod_i M_{i,s(i)}$$

Libraries: Everyday mathematics syntax, implicit semantics

For any matrix $M \in M_n(F)$,

$$det(M) := \sum_{s \in S_n} (-1)^{\epsilon_s} \prod_i M_{i,s(i)}$$

In LaTeX we write:

```
det(M) :=
  \sum_{s \in S_n}(-1)^{\epsilon_s}\prod_i M_{i, s(i)}
```

Libraries: Everyday mathematics syntax, implicit semantics

For any matrix $M \in M_n(F)$,

$$det(M) := \sum_{s \in S_n} (-1)^{\epsilon_s} \prod_i M_{i,s(i)}$$

In LaTeX we write:

```
det(M) :=
  \sum_{s \in S_n}(-1)^{\epsilon_s}\prod_i M_{i, s(i)}
```

In a proof assistant we would like to write:

```
Definition determinant n (A : 'M[R]_n) : R :=
  \sum_(s : 'S_n) (-1) ^+ s * \prod_i A i (s i).
```

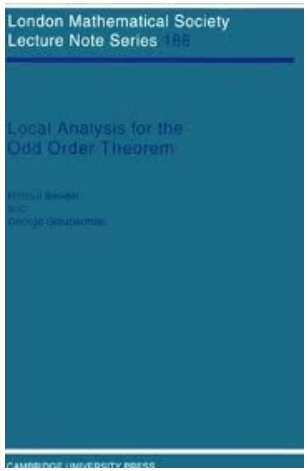
Libraries: Everyday mathematics syntax, implicit semantics

```
Definition determinant n (A : 'M[R]_n) : R :=
  \sum_(s : 'S_n) (-1) ^+ s * \prod_i A i (s i).
```

- This is important to know what you are proving.
- This is not only about syntax but also about property inference.
- This is possible thanks to type inference.

Progress

- 2005: G. Gonthier and B. Werner finish the formal verification of the Four Color Theorem.
- 2006: The Mathematical Component Project starts.
- 2011: First book of the proof of the Odd Order Theorem is completely formally verified.



Ssreflect releases: http://www.msr-inria.inria.fr

- Four Color Theorem Coq sources (G. Gonthier)
  26/04/2006.
- First Inria-MSR release (G. Gonthier, A.M., L. Théry)
  13/03/2008.
  - booleans, lists, natural numbers, finite types
- Latest release (ten people from the MathComp team)
  11/03/2011.
  - basic arithmetics
  - finite functions, finite sets, graphs
  - infrastructure: iterated operators, hierarchy, quotations,...
  - elementary finite group theory
  - matrix, polynomials

Some conclusions

- Our job is not to invent these difficult proofs but to design the infrastructure adapted to their complexity.
- This is computer science for mathematics, and even for computer science.
- Type theory provides the appropriate formal language.
- Computers are now really good at mathematics.