Logipedia: towards a Wikipedia of formal proofs

Gilles Dowek

The universality of mathematical truth

The property we cherish the most

Without it, nothing to share (even in the digital age of science)

But jeopardized several times (e.g. non Euclidean geometries)
Always a remedy:

- in Predicate logic $\mathcal{T} \vdash A$ absolute
- different geometries: different sets of axioms $\mathcal{T}$

Computer checked proofs: a major step on the endless road to rigor

But a new jeopardy: A Coq proof of..., a HOL Light proof of...

Philosophical and concrete problem: interoperability, sustainability, rechecking

A remedy

- Define in ~~predicate logic~~ some logical framework the theories implemented in Abella, Agda, Atelier B, Coq, FoCaLiZe, HOL Light, HOL4, Isabelle/HOL, Lean, Matita, Minlog, Mizar, PVS, Rodin, TLA+...
- Analyze which proof uses which "axiom" (similar to reverse mathematics)
- Translate to other systems the proofs that can be translated

Why extending predicate logic?

- No bound variables ($\lambda x\ x$)
- No syntax for proofs
- No notion of computation
- No good notion of cut
- Classical and not constructive

New logical frameworks

- No bound variables ($\lambda x\ x$): λ-Prolog, Isabelle, $\lambda\Pi$-calculus
- No syntax for proofs: $\lambda\Pi$-calculus
- No notion of computation: Deduction modulo theory
- No good notion of cut: Deduction modulo theory
- Classical and not constructive: Ecumenical logic

The $\lambda\Pi$-calculus modulo theory generalizes them all

Dedukti: an implementation of it

Defining a theory in Dedukti

No universal method
But several paradigmatic examples

- Any (finite) theory expressed in Predicate logic
- Axiom schemes
- Simple type theory (without and with polymorphism)
- Pure type systems (CoC...)
- Inductive types
- Universes
- new: Proof irrelevance
- new: Universe polymorphism

Ongoing: predicate subtyping

Simple type theory and the Calculus of constructions

Express in $D[STT]$ the proofs developed in HOL Light
Express in $D[CoC]$ the proofs developed in Matita

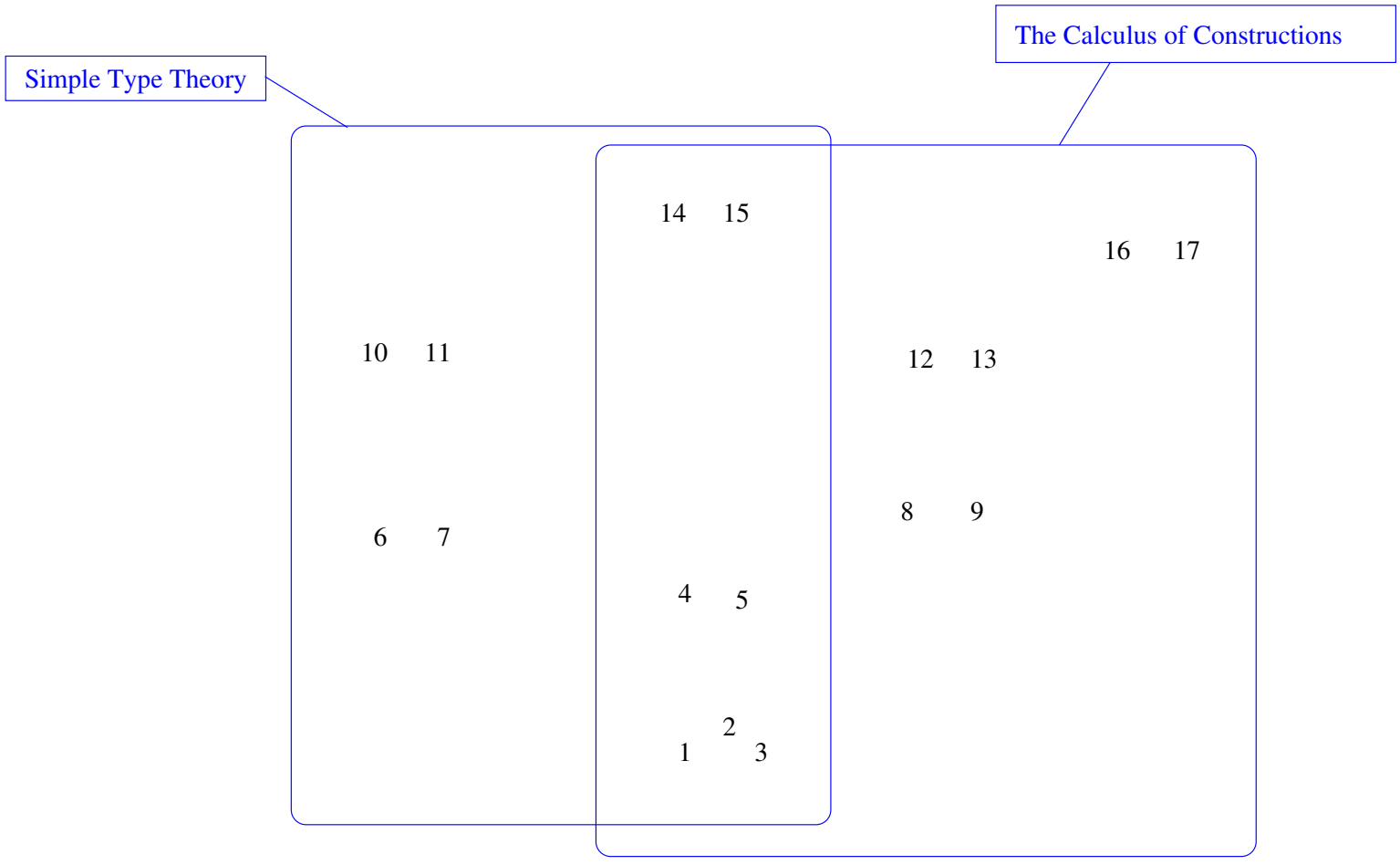
In practice a little bit more difficult: HOL Light a little bit more than *STT*, Matita a little bit more than *CoC*

Very similar theories
11 "axioms" for STT, 13 for CoC
And 7 in common
$11 + 13 - 7 = 17$, 17 "axioms" altogether

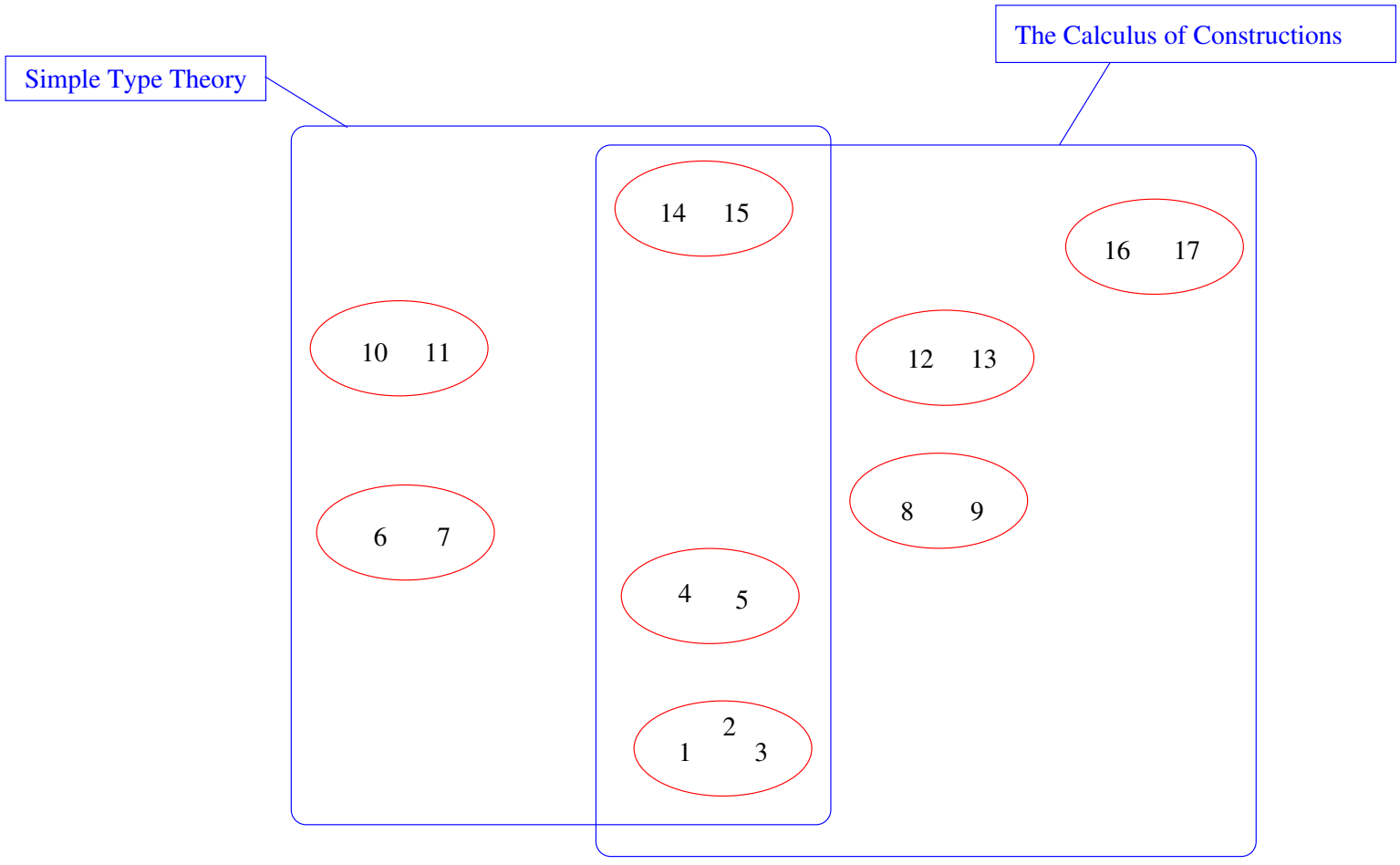
17 “axioms”

$$
\begin{array}{rcll}
\mathit{set} & : & \mathit{TYPE} & (1)\\
\eta & : & \mathit{set} \to \mathit{TYPE} & (2)\\
\mathit{nat} & : & \mathit{set} & (3)\\
\mathit{prop} & : & \mathit{set} & (4)\\
\varepsilon & : & (\eta\ \mathit{prop}) \to \mathit{TYPE} & (5)\\
\mathit{arrow} & : & \mathit{set} \to \mathit{set} \to \mathit{set} & (6)\\
(\eta\ (\mathit{arrow}\ x\ y)) & \rhd & (\eta\ x) \to (\eta\ y) & (7)\\
\mathit{arrow}_d & : & \Pi x : \mathit{set}\ (((\eta\ x) \to \mathit{set}) \to \mathit{set}) & (8)\\
(\eta\ (\mathit{arrow}_d\ x\ y)) & \rhd & \Pi z : (\eta\ x)\ (\eta\ (y\ z)) & (9)\\
\Rightarrow & : & (\eta\ \mathit{prop}) \to (\eta\ \mathit{prop}) \to (\eta\ \mathit{prop}) & (10)\\
(\varepsilon\ (\Rightarrow\ x\ y)) & \rhd & (\varepsilon\ x) \to (\varepsilon\ y) & (11)\\
\Rightarrow_d & : & \Pi p : (\eta\ \mathit{prop})\ (((\varepsilon\ p) \to (\eta\ \mathit{prop})) \to (\eta\ \mathit{prop})) & (12)\\
(\varepsilon\ (\Rightarrow_d\ x\ y)) & \rhd & \Pi z : (\varepsilon\ x)\ (\varepsilon\ (y\ z)) & (13)\\
\forall & : & \Pi a : \mathit{set}\ (((\eta\ a) \to (\eta\ \mathit{prop})) \to (\eta\ \mathit{prop})) & (14)\\
(\varepsilon\ (\forall\ x\ y)) & \rhd & \Pi z : (\eta\ x)\ (\varepsilon\ (y\ z)) & (15)\\
\pi & : & \Pi p : (\eta\ \mathit{prop})\ (((\varepsilon\ p) \to \mathit{set}) \to \mathit{set}) & (16)\\
(\eta\ (\pi\ x\ y)) & \rhd & \Pi z : (\varepsilon\ x)\ (\eta\ (y\ z)) & (17)
\end{array}
$$

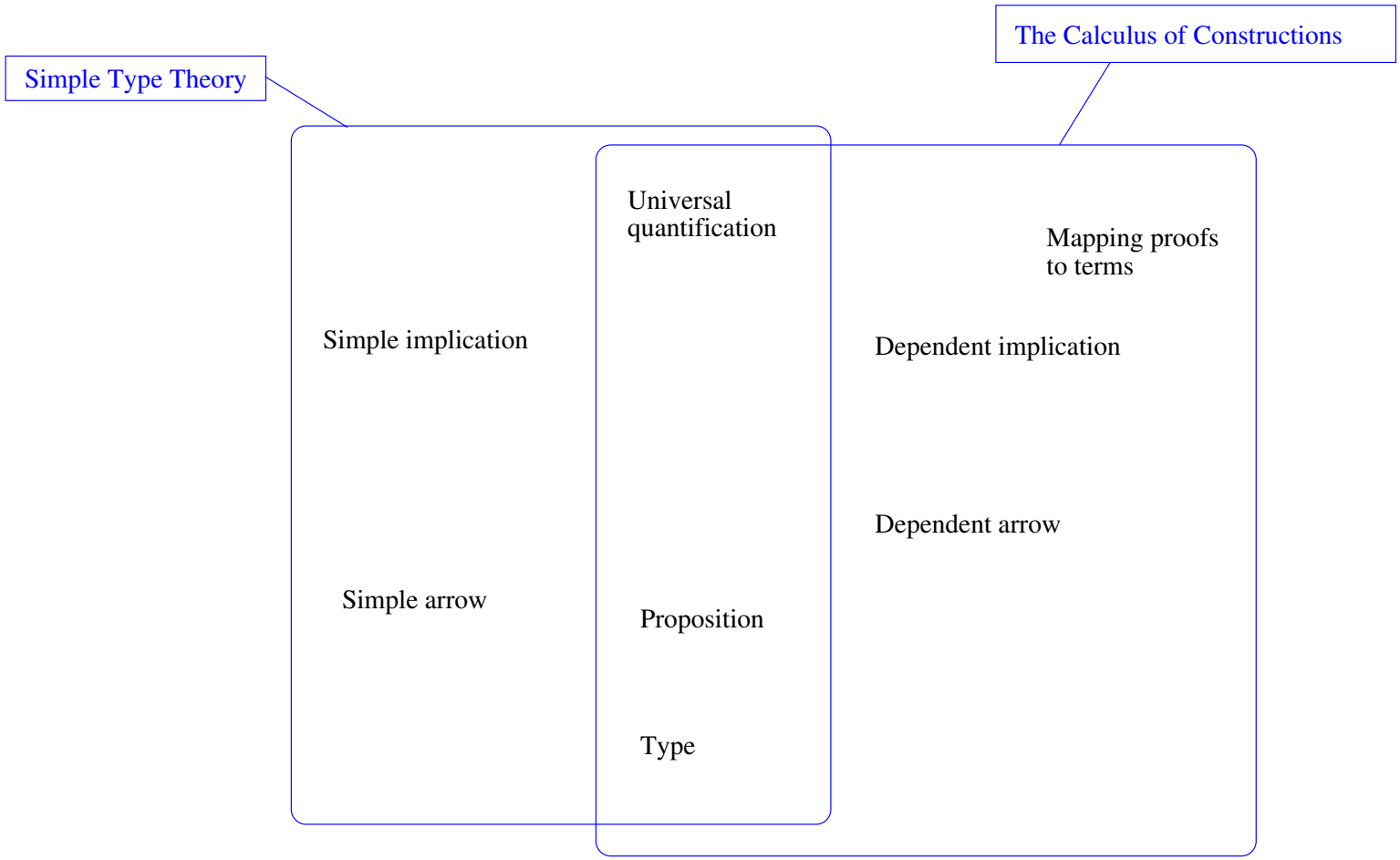
Two fragments



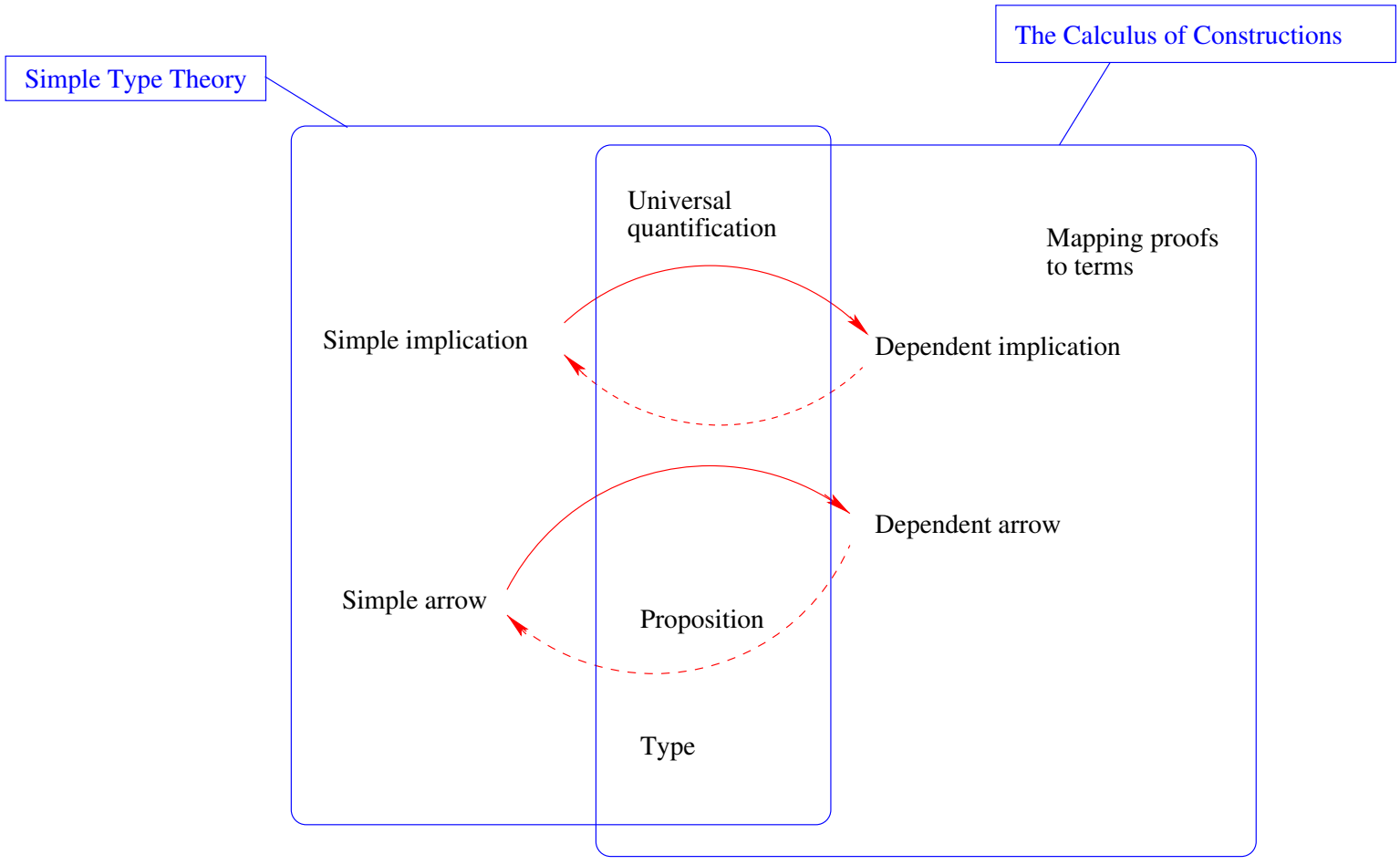
Eight features



Eight features



Translations



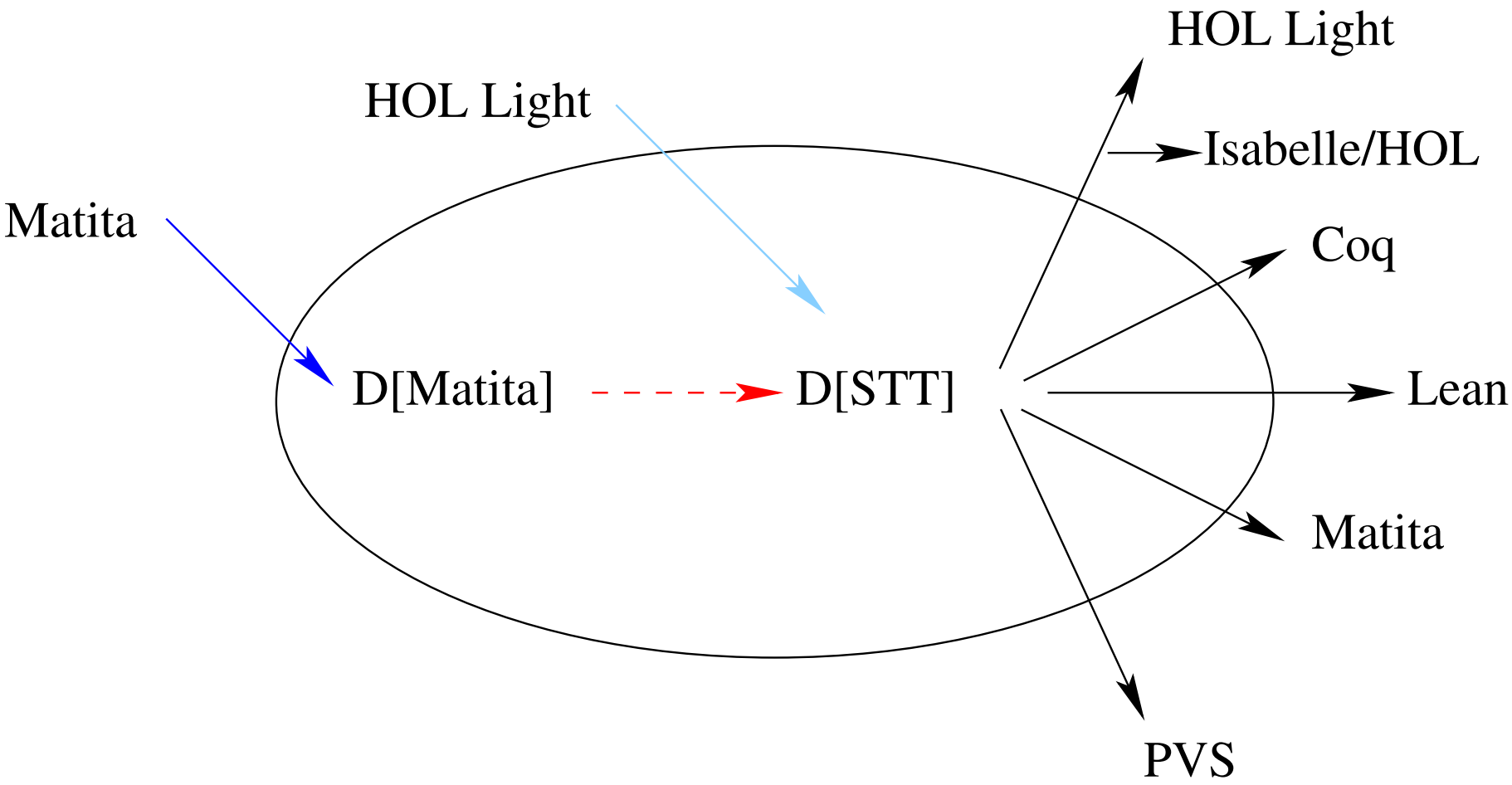
Another feature: polymorphism (four more "axioms") STT$\forall$

Ongoing work: predicate subtyping

Reverse mathematics in Dedukti

- All proofs in Simple type theory can be translated to the Calculus of constructions
- Some proofs in the Calculus of constructions can be translated to Simple type theory (the others: genuine Calculus of constructions proofs)

For example: all the proofs of the arithmetic library of Matita


HOL Light
HOL Light
Isabelle/HOL
Matita
Coq
D[Matita]
D[STT]
Lean
Matita
PVS

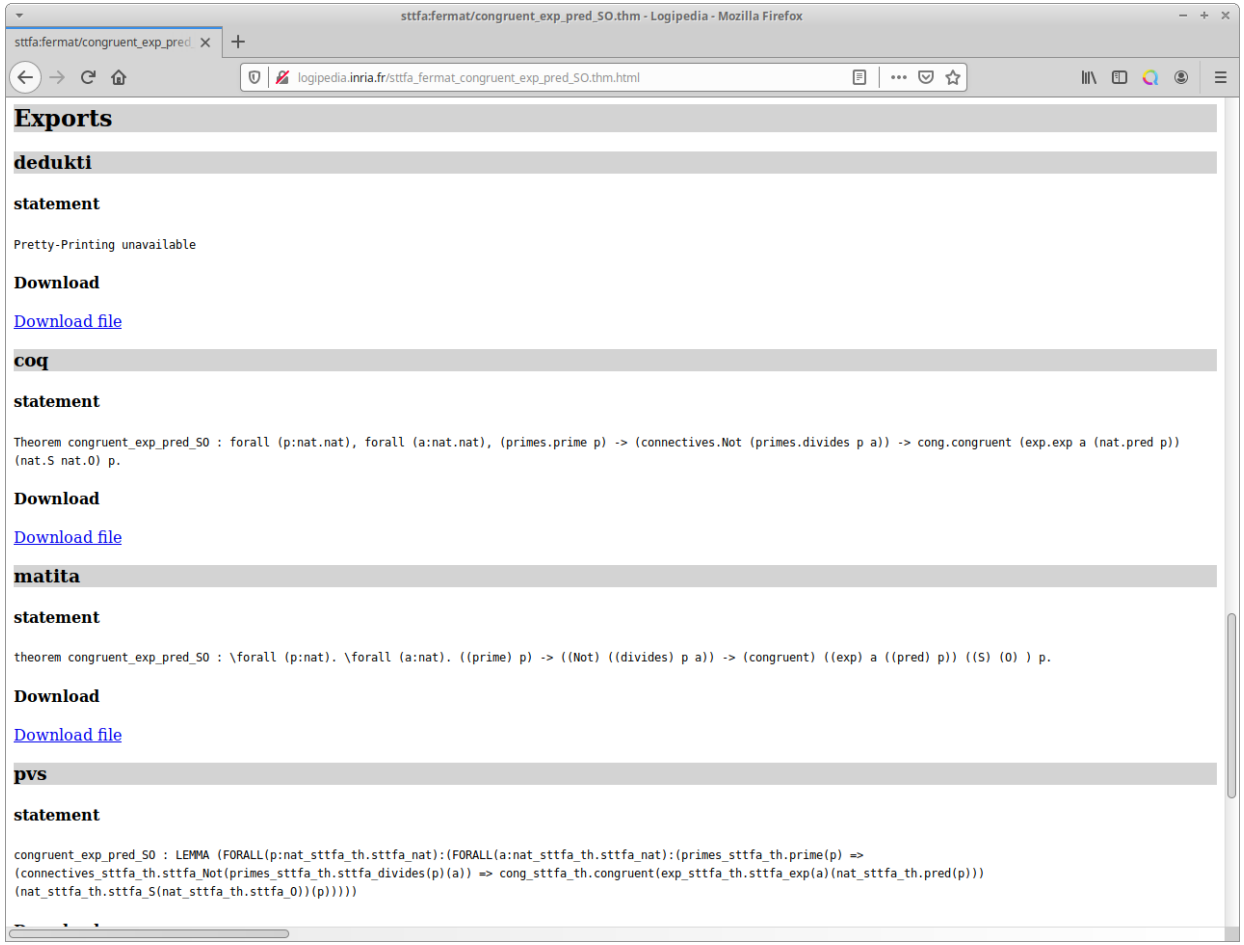
Why does it work so well?

Because proof systems implement very expressive theories and use only a tiny part of it

Early empirical evidences

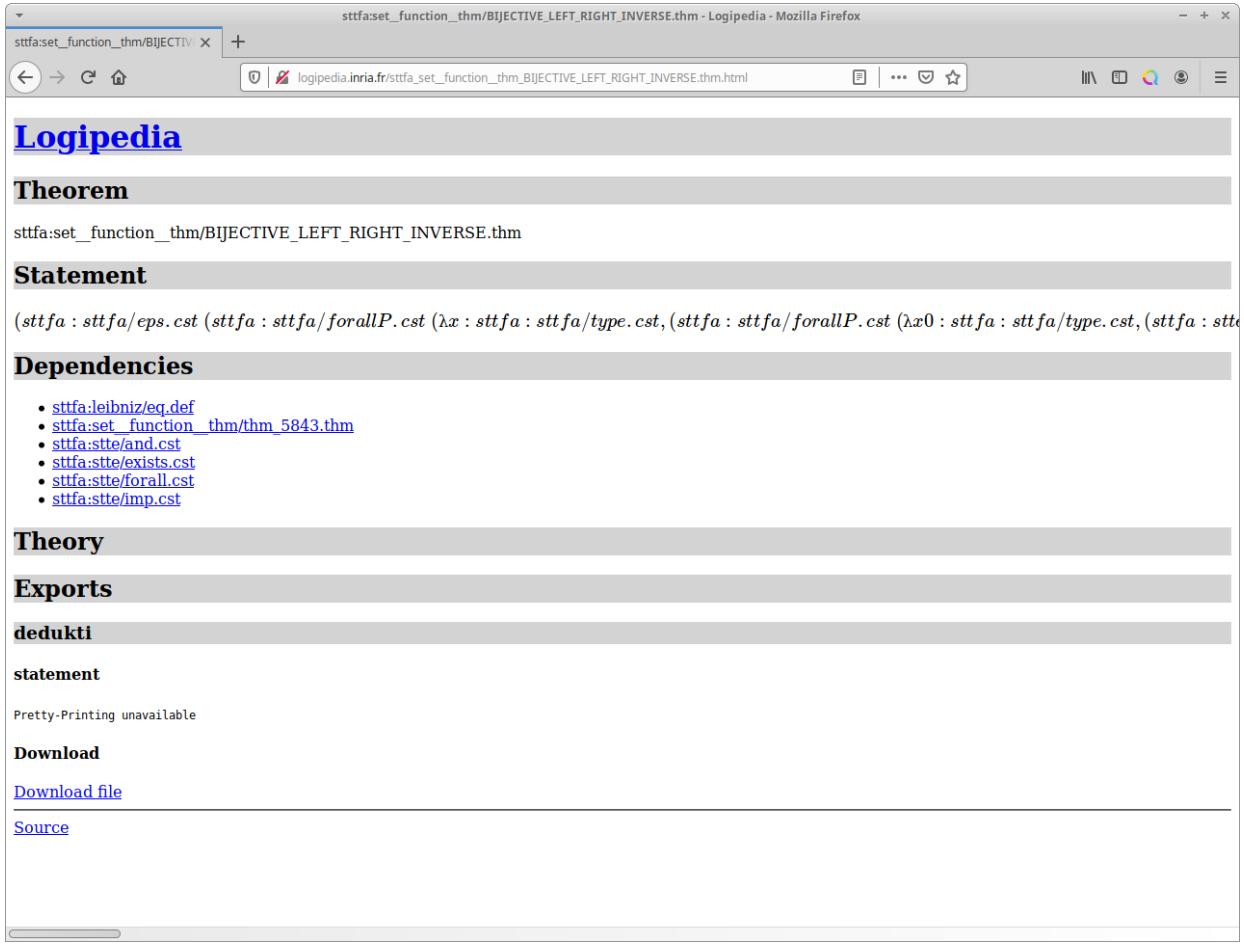
- Proof systems: very different theories, but very similar libraries
- Mathematicians do not bother much about the actual logical system they work in: any theory seems to fit

Collecting all DEDUKTI proofs in a single data base



http://logipedia.science

Collecting all DEDUKTI proofs in a single data base



Coming soon: COQ, ISABELLE/HOL

Steps

- Define the theory of system X in Dedukti (PVS, Mizar)
- Translate the library of system X in $D[X]$ (Isabelle/HOL, Coq)
- Analyze and translate these proofs (Matita)
- Align concepts with other concepts of the library (making formal the saying: Cauchy sequences or Dedekind cuts immaterial)

Already concrete results

While QED (1993) did not go very far

- Better understanding of the theories implemented in the various proof systems
- A new logical framework to express these theories
- Analyzing the proofs (reverse mathematics) before we share them (partial translations)

Universality of mathematical truth strikes back