

# Modelling and Verifying Algorithms in Coq

*an introduction*

A CEA-EDF-INRIA summer school



Welcome!

A big thank you to

*Chantal Girodon*

for the help with the organisation



Software and cathedrals  
are much the same.





Software and cathedrals  
are much the same.

First we build them,  
then we pray.



# The Coq proof assistant

---

Coq can be seen as a combination of two things:

1. A simple and slightly idiosyncratic,  
but extremely expressive, *programming language*.
2. A set of tools for stating *logical assertions*  
(including assertions about the behaviour of programs)  
and marshalling evidence of their truth.







Started in 1984 by Thierry Coquand and Gérard Huet.



Extended in 1991 with inductive types by Christine Paulin.

*About 40 people contributed to Coq over more than 20 years.*

A disclaimer

circa 2006, situation has improved since.



# How many months can it take to verify 30 assembly instructions?

After years doing programming language theory without going near a proof assistant, I was finally convinced by the POPLmark ‘buzz’ and conversations at ICFP’05 that it was time to try one.

I can confirm that I have rarely felt as stupid and frustrated as I did during my first few weeks using Coq.

Scripts are unreadable by themselves, as one has no idea what the tactics are doing to the proof state, and the documentation for them is incomprehensible to the novice.

I had many similar difficulties, but then started to make progress.

Coq is worth the bother and it, or something like it, is the future, if only we could make the initial learning experience a few thousand times less painful.

The answer is about four



Nick Benton  
circa 2006

# How many months can it take to verify 30 assembly instructions?

After years doing programming language theory without going near a proof assistant, I was finally convinced by the POPLmark 'buzz' and conversations at ICEP'05 that it was time to try one

In this school  
we have some fantastic Coq-experts handy...

Ask questions whenever you are stuck!

idea what  
mentation

I  
did

I had many similar difficulties, but then started to make progress.

Coq is worth the bother and it, or something like it, is the future, if only we could make the initial learning experience a few thousand times less painful.

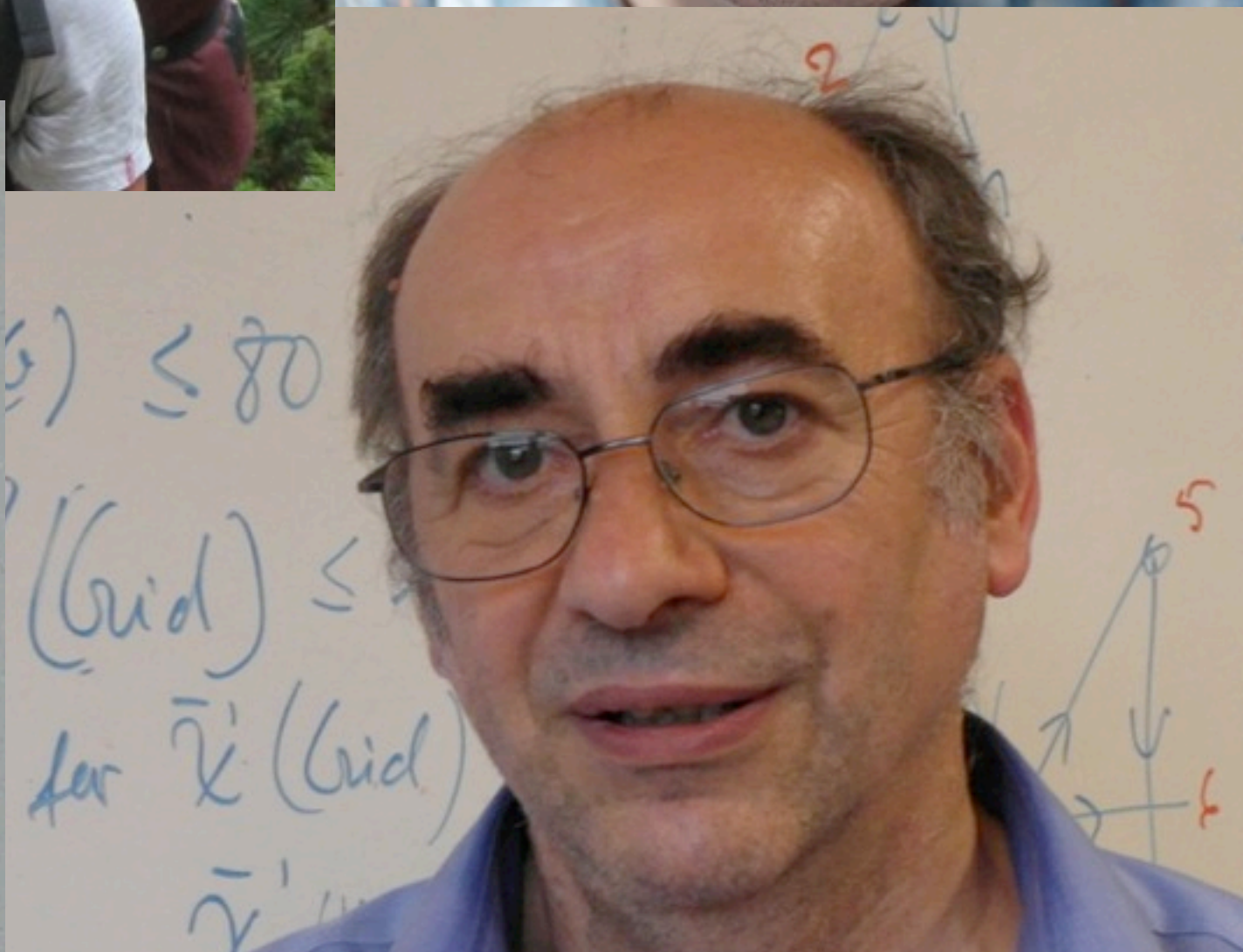
The answer is about four



Nick Benton  
circa 2006

Our speakers







# Plan of the week

---

Programming on natural numbers and lists in Coq  
Propositions and Predicates

*monday*  
*salle bleue*

Making proofs in Coq  
Proofs about programs

*tuesday*  
*salle algorithmme*

Inductive data types  
Inductive Predicates 1

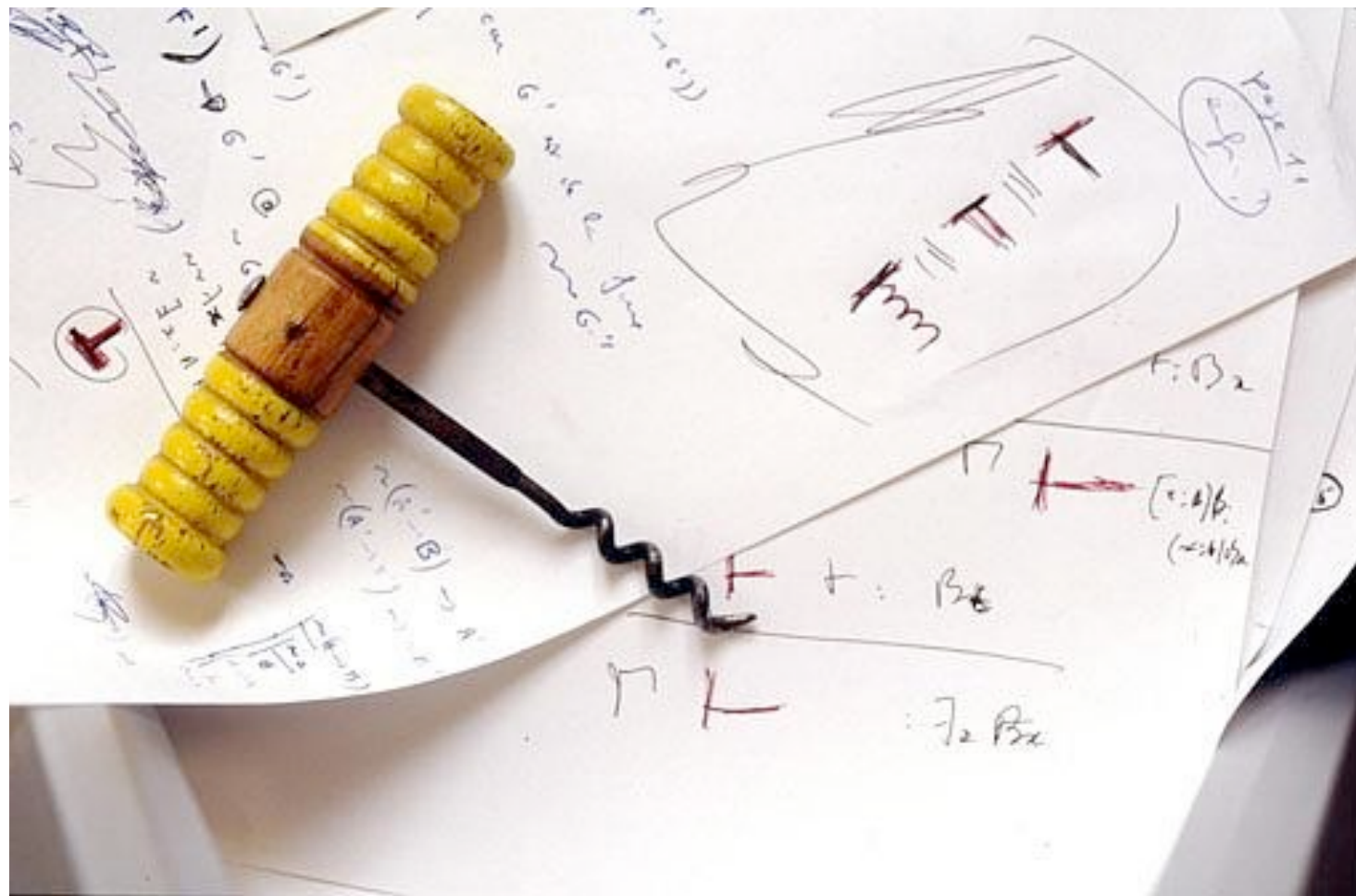
*wednesday*  
*salle orange*

Dependently typed programs with propositions  
Inductive Predicates 2

*thursday*  
*salle verte*

Recursive functions in Coq  
Light introduction to Classes and Setoids

*friday*  
*salle verte*



|             |              |
|-------------|--------------|
| 9h-10h30    | lecture      |
| 10h30 - 11h | <i>pause</i> |
| 11h-12h30   | lecture      |
| 12h30-14h   | <i>lunch</i> |
| 14h-17h     | exercices    |



Hope you will enjoy  
the school!

And do not hesitate  
to ask questions.