Sharing Formal Mathematics and Programming

Sharing Formal Mathematics and Programming

Explosion in formal math

- **Applications**

Problems & Goals

- **Problems**

  Knowledge is formalized in different logics

  Reasoning/tactics are logic specific

Prover architecture has not kept pace

- **Flat name spaces**

Results

- **Logical foundation for formal modules**
  - Dependent records; very dependent types
- **Design of Nuprl-Light library component**
  - object-oriented capabilities based on very-dependent types
  - protection mechanism
  - abstraction
  - initial integration of tactic and logical state

Implementation of Nuprl-light•

  - multiple type theories
  - Ensemble integration

Scalability

Module systems

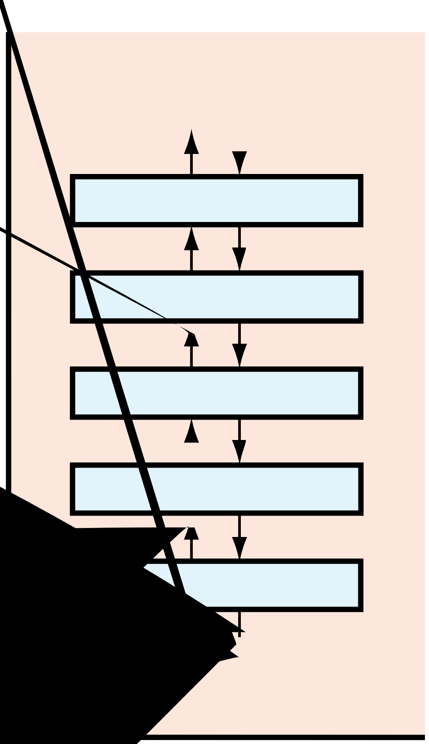
- **Java modules provide abstraction and securit**

Hierarchy
Abstraction
Security
Naming

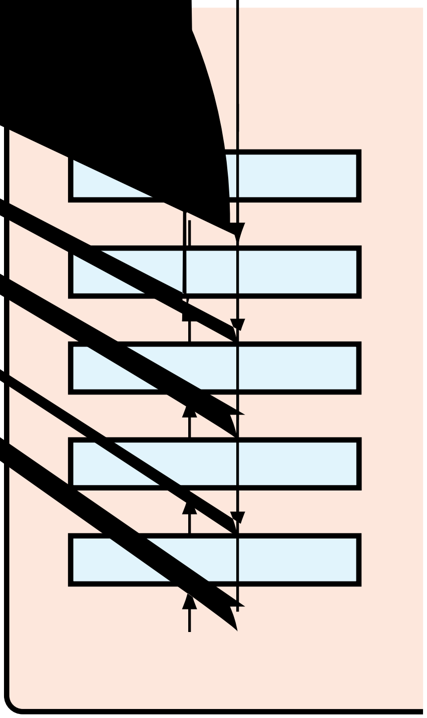
- **ML provides second order modules**
- **Functors/sharing constraints for object oriented programming**

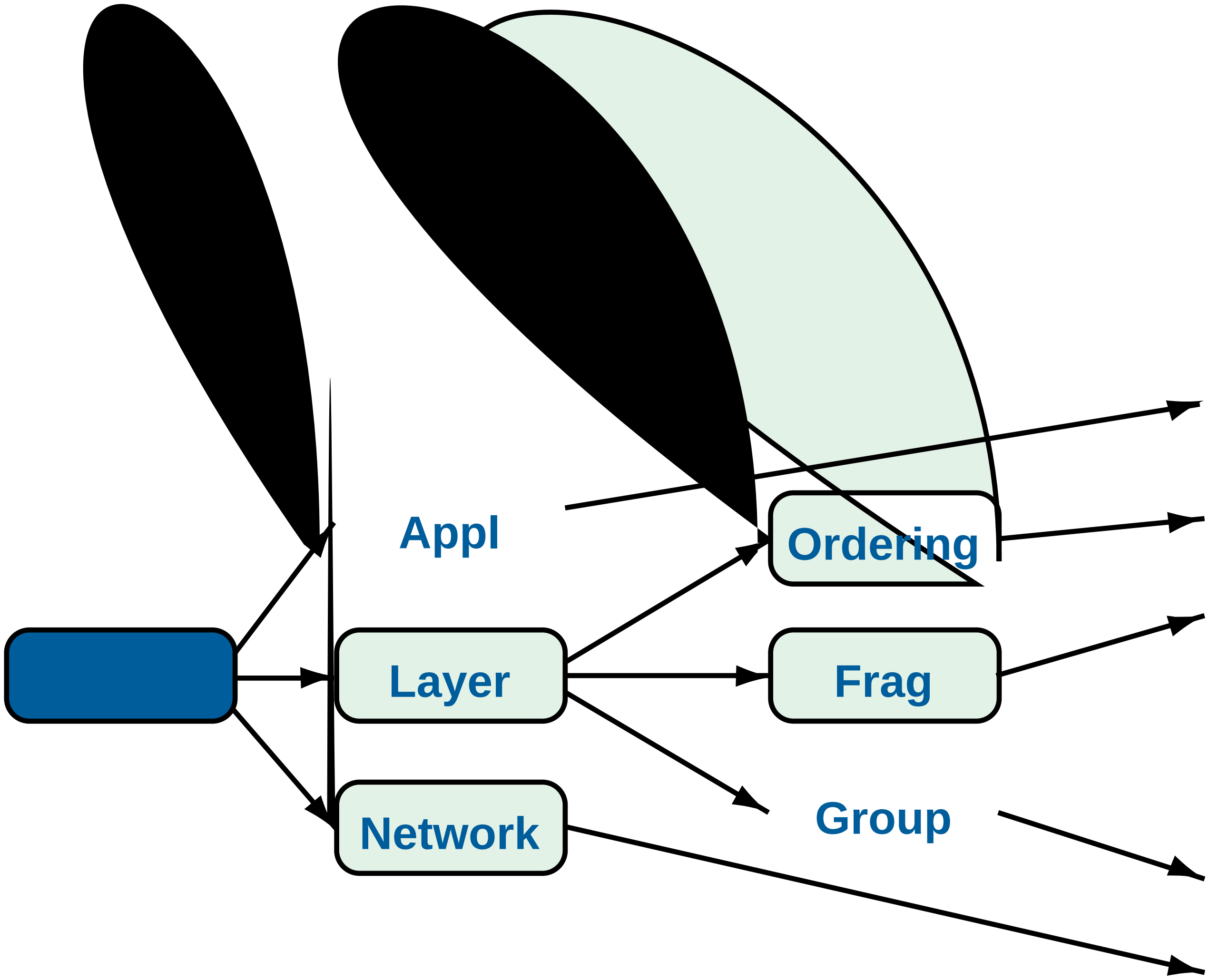
Software models

Hierarchical, refinement ordering



- Hierarchical,





Software Architecture

Mathematical models

Arbitrary

graph of

A Programming Logic

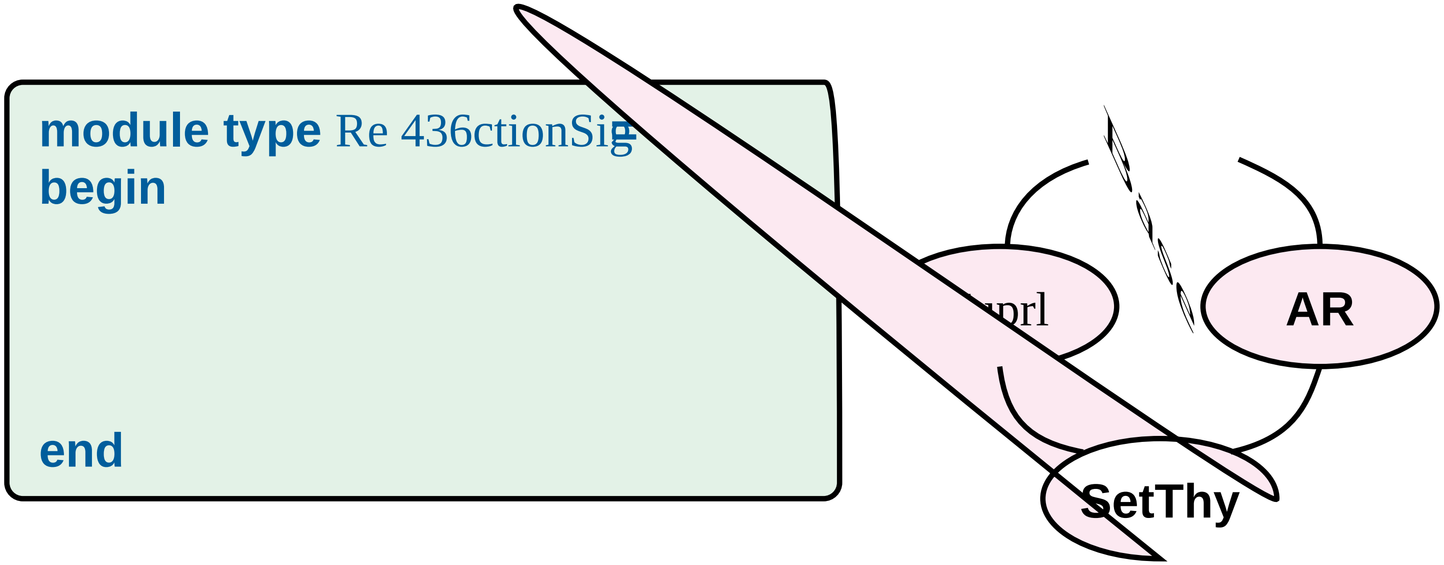
module type *A_LogicSig* **=**
begin

axiom inv_cond: $\Gamma \vdash \{I \quad P\}\ p_1\ \{I\} \qquad \Gamma \vdash \{I$
$\Gamma \vdash \{I\} \quad P$ **then** p_1 **else** $p_2)\ \{I\}$

end

Derivations

- **Derive Intuitionistic Set theory from type theory**
- **Multiple inheritance**



```
module type Re 436ctionSig
begin



end
```

Status

Nuprl-Light alpha release

- **modular Nuprl type theory**
- **LF type theory**
- **automatic generation of formal module types**
- **modular tactic librar**

Conclusion

Newmodeloftheoremproving

- searchable mathematics

- formal programming

- assistance of knowledge exchange

- theorem proving *in the large*