

**Roel Wuyts**  
**Université Libre de Bruxelles**

**Benevol, May 26th 2005 (Eindhoven, The Netherlands)**

# Type Reconstruction

- [ Context: program understanding in dynamically typed languages

- e.g. extraction of class diagrams

- [ Type Reconstruction

- input: program without types

- output: program with types

# Trade-offs

- [ Precision vs. efficiency

- We chose efficiency for usage in a development browser

- Use Heuristics as basis for the reconstruction

- instead of full reconstruction

# Heuristics

— [ Direct sends to instance variable

— [ Indirect sends to instance variables (getter methods)

— [ Direct assignment expressions

— [ Indirect assignment expressions (setter methods)

— [ (Type snooping)

# Implementations

- [ Using LiCoR (Library for Code Reasoning) in SOUL
  - on the parse tree
  - average: 500 milliseconds / instance variable
  - more elaborate and easier to extend
- [ Using partial evaluation on the byte code
  - average: 30 milliseconds / class

# Demo

The screenshot shows the Smalltalk IDE interface for the `Point` class. The window title is "Point". The menu bar includes "Browser", "Edit", "Find", "View", "Package", "Class", "Protocol", "Method", "Tools", and "Help". The toolbar contains various icons for navigation and editing. The "Find:" field contains the text "Point".

The interface is divided into several panes:

- Package Hierarchy:** Shows a tree view of packages, with "Graphics-Geometry" selected.
- Class List:** Lists classes in the selected package: "Geometric", "LineSegment", "Point" (selected), "Polyline", and "Rectangle".
- Instance Class Shared Variable:** Lists instance variables: "accessing", "arithmetic", "coercing", "comparing", and "converting".
- Source+Types:** Shows the class definition code and its instance variables.

The class definition code is:

```
Smalltalk.Core defineClass: #Point
  superclass: #{Core.ArithmeticValue}
  indexedType: #none
  private: false
  instanceVariableNames: 'x y'
  classInstanceVariableNames: ''
  imports: ''
  category: 'Graphics-Geometry'
```

The instance variables are:

Variable Name	Type
y	Number
x	Number

At the bottom of the window, there are three status boxes:

- Class: Core.Point
- Parcel: none
- Package: Graphics-Geometry

# Conclusions & Future Work

## Works

- *About 80%* of correctness on built-in libraries
- Better on domain-specific code

## Future Work

- *Fix About*
- Will do this on (untyped) Java code and compare results