

# Reusable Components for Formal Semantics

Neil Sculthorpe

PLanCompS Project  
Theory Group  
Department of Computer Science  
Swansea University  
N.A.Sculthorpe@swansea.ac.uk

Swansea, Wales  
29th August 2014

# Formal Semantics for Programming Languages

## Pros:

- Precise and unambiguous
- Benefits programmers and implementers

## Cons:

- Time-consuming to specify
- Get outdated as a language evolves
- No reuse between specs of similar languages

# Funcons: Reusable Semantic Components

- We're creating a repository of **fundamental constructs** (funcons)
- Funcons are **reusable** components for defining semantics
- Each funcon defines a programming concept, e.g.
  - while loop
  - variable assignment
  - function application
- Funcons have **formally specified** semantics

# Case Studies

- Caml Light
- C#
- Java