BETTY WG3: Languages

March 24, 2013

What WG3 is about

Purpose: filling the gab between abstract models and techniques and concrete programming languages

- *syntax*

- *static semantics* (type checking, verification, . . .)

- *runtime semantics* (implementation of communication mechanisms, monitoring, adaptation, . . .)

What we have proposed to do

1. mainstream programming languages
   - **object-oriented** languages
     (adoption, tool availability, integration with behavioral type theories)
   - **functional** languages
     (playground for new constructs, parallelism)
   - **XML/JSON/YAML**
     (serialization and exchange formats)

2. domain-specific languages

How we have proceeded

- code annotations

- macro expansion of high-level constructs
  (enhanced typing)

- conservative extensions (in toy/real languages)

What's out there

- OO (methods, state)   (**talk 1**)
- Plaid (type-state)
- C
- Java extensions (macros)
- ML & Haskell   (**talk 2**)
- Scribble + Python   (**talk 3**)
- Singularity OS
- APIs
- . . .

Workplan

Year 1
State of the art (what is available, current practice)

Year 2
Non-invasive extensions (comments, macros, attributes, annotated APIs, . . .)

Year 3
Feedback from WG4 $\Rightarrow$ high-level constructs

Year 4
Conservative extensions (adapters/interfaces)