Unmanned Autonomous V&V

Lee Pike and Don Stewart

Galois, Inc.

John Van Enk

DornerWorks

Domain-Specific Languages (DSL)

- Mathematics for engineering domains each have special syntax, functions, etc.

- Programming for the domain should feel like writing the mathematics
  - The boilerplate software should be abstracted out
  - The specification then becomes the program

- Examples:
  - **Yacc**, a parser-generator for compiler front-ends
  - **Cryptol**, a language for specifying cryptographic protocols

Light Weight DSLs (LwDSLs)

- Also known as *embedded* DSLs in the literature.

- Think of LwDSLs as

  domain-specific libraries + domain-specific syntax
  (but a little goes a long ways)

- Why LwDSLs over DSLs?
  - Don't need to write your own compiler
  - Multiple DSLs in the same host language (**compossible** DSLs)
  - Tool and library reuse

But don't take our word for it…

Industrial LwDSLs Today

- (Research for) **Boeing**: a LwDSL for component configuration in real-time embedded systems. Resulted in 30x reduction of spec size & hundreds of errors caught.
- **Eaton**: LwDSL for describing safety-critical behavior of hydraulic hybrid vehicle control.
- **Antiope**: simulation of ultra low power radio chips.
- **Xilinx**: high-level hardware description language.

LwDSL Benefits for V&V

Let's use Haskell (a popular functional language) as a concrete example:

- V&V tools
  - Semantic types.
  - Automated testing (QuickCheck) and coverage analysis.
  - Code coverage
  - Translators into FV tools
- Synthesis tools
  - Efficient compiler (oftentimes comparable to C)
  - Multicore support
  - Profiling support
  - Someone (else!) maintains the compiler
- And you can easily roll-your-own new tools.

LwDSLs for Mixed-Criticality Systems?

- Composable DSLs
  - **Composition** is easy---it’s all hosted in the same language.
  - **Composition** in multiple contexts---compilation, testing, and formal verification.

- Fast **prototyping** of mixed systems for simulation

  Can use as your requirements for a *C* implementation

- Targeted V&V as appropriate for the level required

  Test/verify different functions together

DornerWorks

| galois |