Assertion-based Debugging of HW/SW Concurrency Issues in Many-core Systems

Luis Gabriel Murillo. Contact: murillo@ice.rwth-aachen.de

- Debugging is hard with multiple cores
  - Many HW/SW actors working concurrently

- How to deal with concurrency bugs at the system level?
  - Hard to find and understand
  - Root unrelated to visible symptoms
  - Random occurrences
  - Not always reproducible
  - May remain unnoticed

- The trend: Debugging based on Virtual Platforms (VPs)
  - Enable earlier HW/SW integration
  - Full control of HW and SW
  - Non-intrusive inspection
  - Vehicle to reproduce concurrency bugs

- SWAT: Language for System-wide Assertions
  - Easy way to capture user knowledge, covering:
    - SW contexts (thread, process), variables
    - HW devices, signals, registers
    - Concurrency-related events (e.g., OS events)
    - Linear Temporal Logic (LTL)

- SWAT Debugger Framework

  a. Front-end and AST generation
  b. IR-level partitions and transformations
  c. Assertion Checker generation
  d. Source debugger back-end

- Evaluation:
  - EURETILE Platform

- SW stack
  - KPN programming model for user applications (DAL)
  - Threading model and message passing for lower SW layers
  - OS running on each tile

- SWAT being applied for:
  - OS-level debugging and driver ISRs, e.g.: