Compliance Task Group

Allen Baum (new) Chair

Progress: just starting to get (re)organized

- Email about charter outline thoughts sent out
- Monthly meeting time(s) established
  - Week1: Thursday 8am PT *(except first meeting..☺)*
  - Week3: Wednesday 9am PT
- Zoom meeting set up:
  - https://zoom.us/j/6213886723
- Group email/website:
  - workgroup_mailer@workspace.riscv.org
  - https://workspace.riscv.org/higherlogic/ws/groups/Compliance_TG/

What is (and isn't) Compliance

- Why do we care?
  - Customers of Risc-V products want to know that what they will pay for will run their (possibly shrink wrapped) application.
- There is no "Risc-V" compliance!
  - There is only compliance to a particular Execution Environment (EE)
  - The minimal EE is compliant to the user ISA,
    - useful mainly for educational simulators
    - But even there, a memory map must be defined
  - A more useful example is Linux platform profile

Existing Work

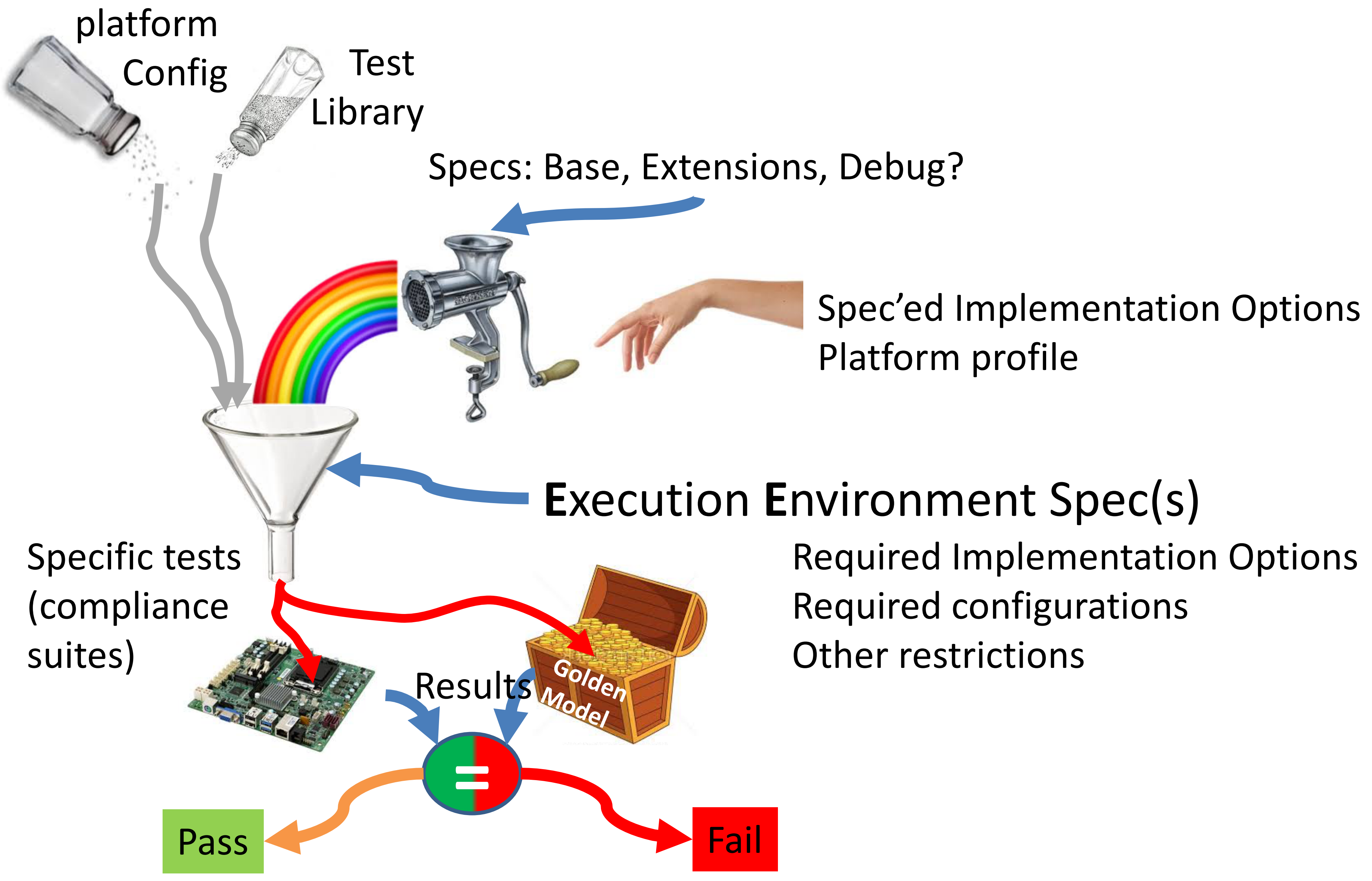
- There is a framework for tests
  - Includes a need for a reference (golden) model and the notion of module and profile specific tests
  - Test specifications
    - Goal(s)
    - Pointer(s) to relevant ISA spec
    - Test description
    - Actual code (format?)
- Tests consist of
  - Initialization definition
  - Instruction by instruction tests
  - Reference model results
  - Test structure and naming convention,
- Some tests, in two categories:
  - Testing that (some of) the tools used to generate tests actually generate what they say they are generating
  - Actual user ISA tests.
  - Reference simulator macros for test development (e.g. Halt, data initialization)

Compliance TG Charter: (for discussion)

The Compliance Task Group will

- Develop a suite of tests for Risc-V, taking into account approved specifications for:
  - Architectural versions (e.g. RV32I, RV32E, RV64I, RV128I)
  - Standard Extensions (M,A,F,D,Q,L,C,B,J,T,P,V,N)
  - All spec'ed implementation options (incl. MHSU modes)
- Develop a method for selecting appropriate tests to an Risc-V implementation, taking into account:
  - Platform profile
  - Execution Environment (EE)
  - Implemented architecture, extensions, and options
- Develop a method to apply the appropriate tests to an implementation and verify that it meets the standard
  - Already decided: test result signature store in memory will be compared to a golden model result signature

Compliance Framework



Charter issues: Work to be discussed

- Selection of Golden Model (from Haskell formal model?) mem model being added – handles constraints?
- Describing Implementation Options
  - A list of all possible options need to be extracted from the spec(s)
  - A standard format for describing them must be established
- Test domain definition (==Execution Environment)
  - Base ISA, privilege level ISA, standard extensions
    - ? How does this relate to non-ISA specs. e.g. debug?
  - Platform profiles (requirements and restrictions) must be defined
  - a standard format for describing them must be defined (at least an example)
- Defining test coverage (especially privilege level)
  - How do we measure how comprehensive our compliance tests are are?
    - E.g. coverage of reference model code
- Acquiring Test source(s)
  - How do we decide to add tests to our repository? Can/should we also pay for them?
  - Should the foundation build and pay for a team to do this?
- Defining revision control methodology (compliance must include version #)
  - both fixing/updating existing tests and adding new ones
- Defining a methodology to specify to test SW which tests an implementation should run
  - Heavily depends on architectural options an implementation has & how communicated
    - Are there dependencies on the config string to be properly defined to communicate this?
    - Options are difficult for compliance testing; interacting options exponentially so.
  - This must include Execution Environment & Platform compliance, not just ISA compliance
- Defining the methodology of compliance test -> applying RiscV compliance trademark
  - e.g. procedures to validate that products have actually passed compliance tests.
  - Note that tests pass based on results store to memory matching golden model results