Imperial College London

Towards verifiable trust management for software execution

Michael Huth and Jim Kuo

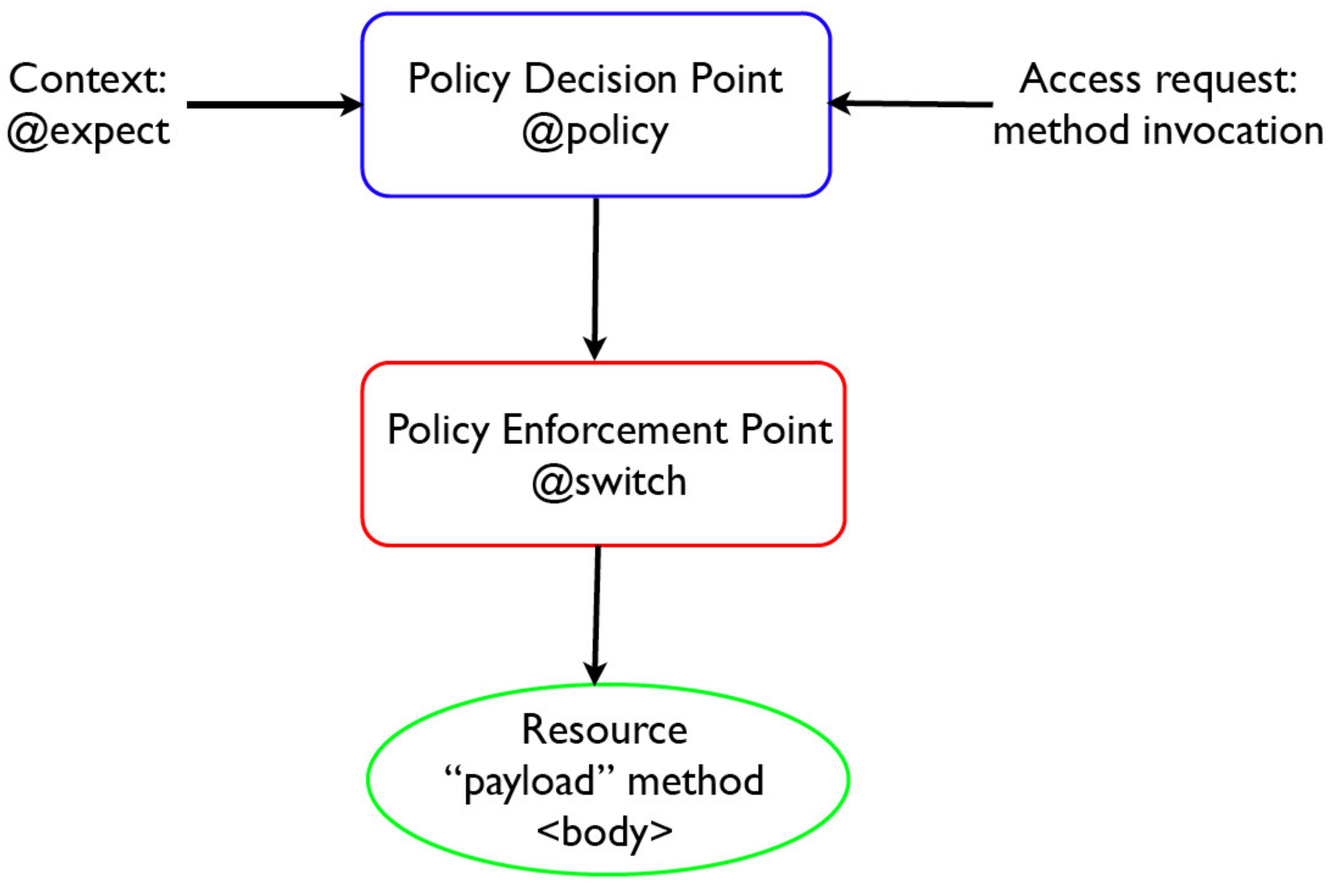
Research Goals

- Bring quantitative trust management into software development.
- Guard-rail software execution by policies that reflect risk posture
- Develop proof of concept prototype for method executions in Scala

Architectural View of Method Guard-Rails

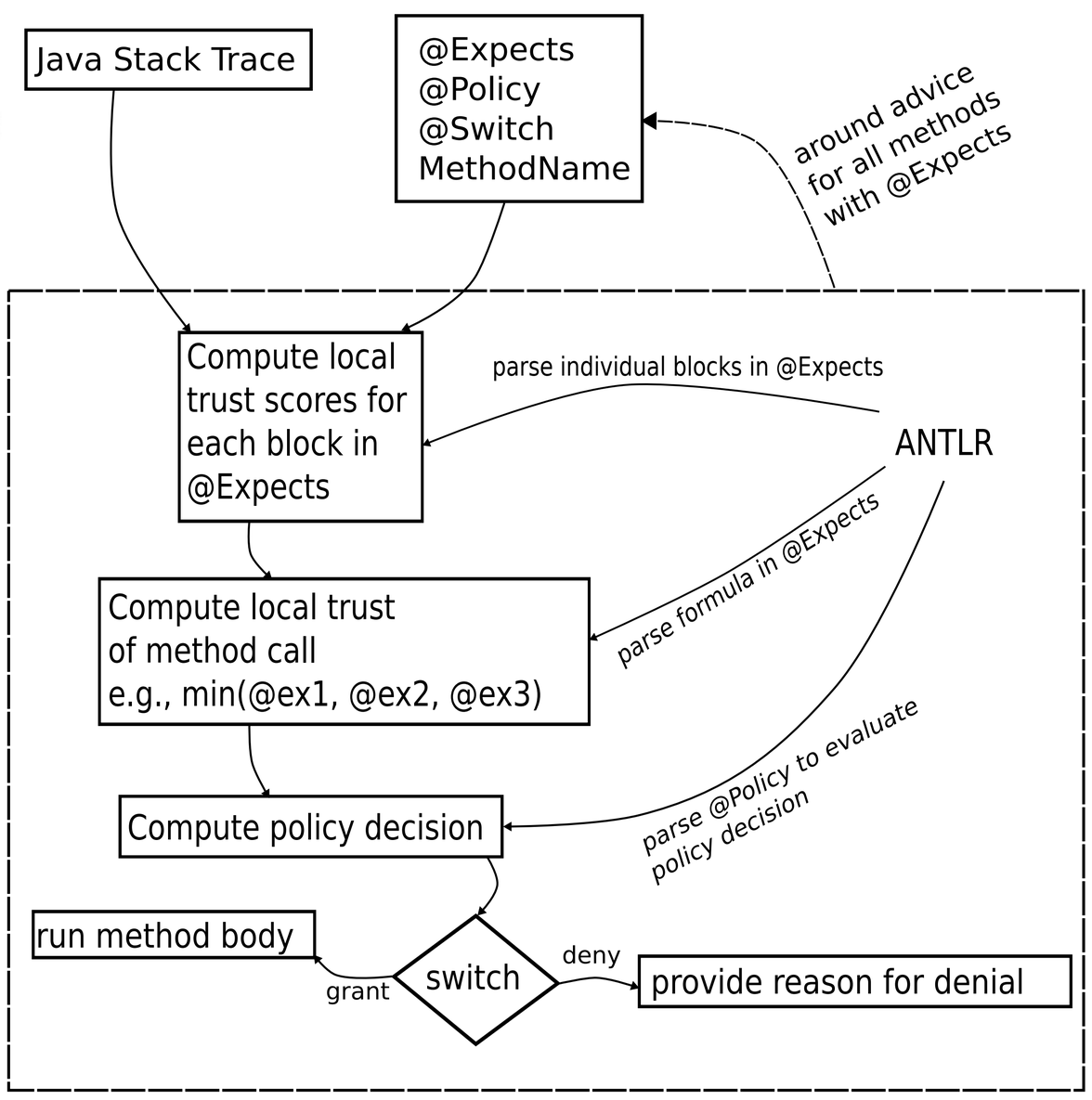
Three annotation layers provide staged guard-rails:

- @Expects - captures behavioural expectations
- @Policy - access decision informed by @Expects
- @Switch - policy enforcement: allow/deny execution



Implementation Details

- Use AspectJ to inject behaviours into program byte code
- Obtain input parameters from AspectJ JoinPoints
- Obtain caller name from Java Stack Trace
- Injected behaviour uses ANTLR to parse input policies
- Parsed strings evaluated to “local trust” value for each @Expects block, and to policy decision for @Policy
- Allow or deny execution of annotated method body according to policy decision and rules specified in @Switch



Annotation Example in Scala

```
@Expects(value = Array(
  new Expect(defaultTrust = 0, rules = Array(“called by foo hasTE 0.2", "[0] > 4 hasTE 0.1", "called by foo1 hasTE 0.5", "[1] > 11 hasTE 0.1"), composition = "+"),
  new Expect(defaultTrust = -999, rules = Array("map [0] from [0,10] to [0,101]", "map [1] from [0,10] to [1,2]"), composition = "max"),
  new Expect(rules = Array("map [2].p1 from [0,20] to [0,1]", "map [2].p2 from [0,20] to [0,1]"), composition = "min")
), localTrust = "max(@0, @1 * @2)")
@Policy(rules = Array("allow if trust >= 50"), composition = "join")
@Switch(na = "deny", deny = "deny", allow = "return", conflict = "return")
def bar(input0: Double, input1: Double, input2: Object): Any = {
  //performing some execution...
  0
}
```

intel
Sponsors of Tomorrow.™

Conclusion

- Future software systems more resilient if risk models are incorporated as first-class citizens
- Trust evidence needs to combine qualitative and quantitative evidence reliably
- Trust evidence may have different information sources and be computed by different techniques
- We demonstrate how trust evidence may be aggregated and enforced in the Java Virtual Machine