Understanding Evidence: Lessons from the GPCA Case Study

Insup Lee, Oleg Sokolsky
PRECISE Center
University of Pennsylvania

SCC Workshop
January 2013

Penn Engineering

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

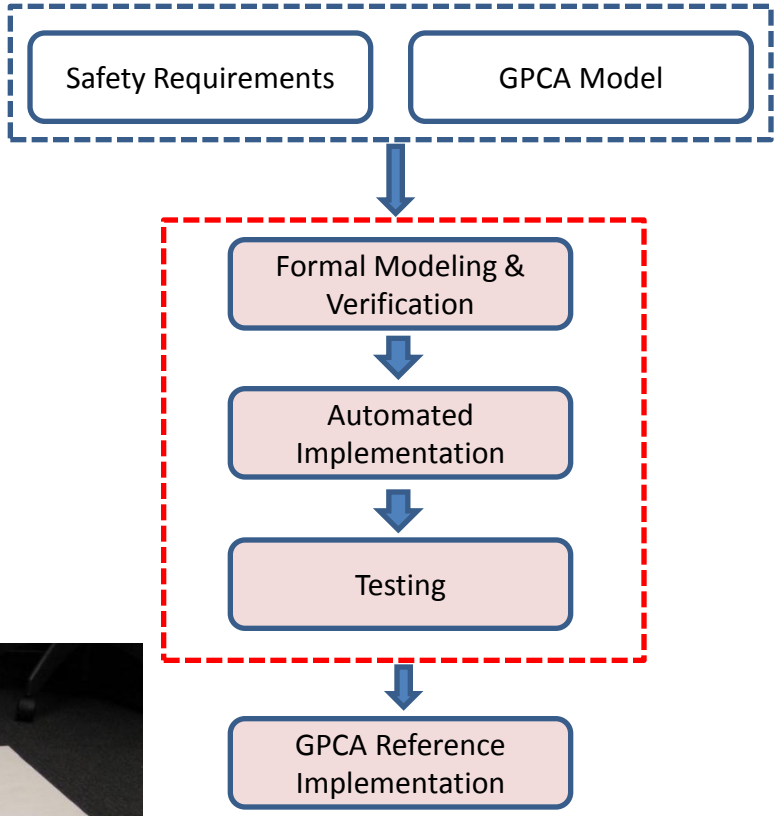
Outline

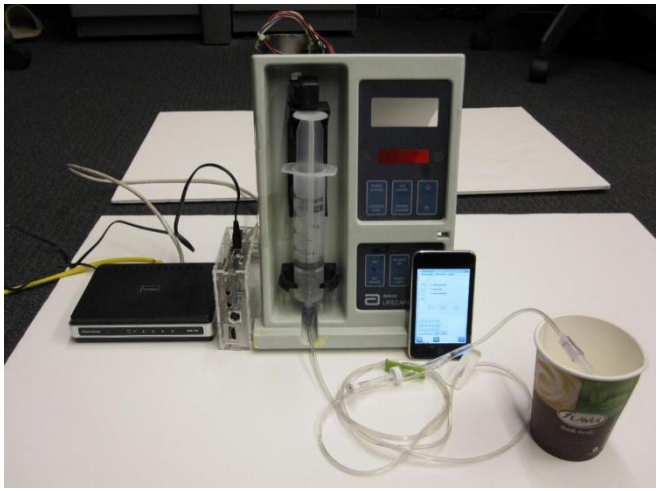
- GPCA Case Study
  - Prototype implementation
  - Development approach
  - Safety argument
- Lessons
  - Evidence for the safety argument
    - Confidence in the evidence
  - Evidence from formalization

GPCA Case Study

Goals:

- Study generative techniques in assurance-based development
- Reason about the achieved level of assurance

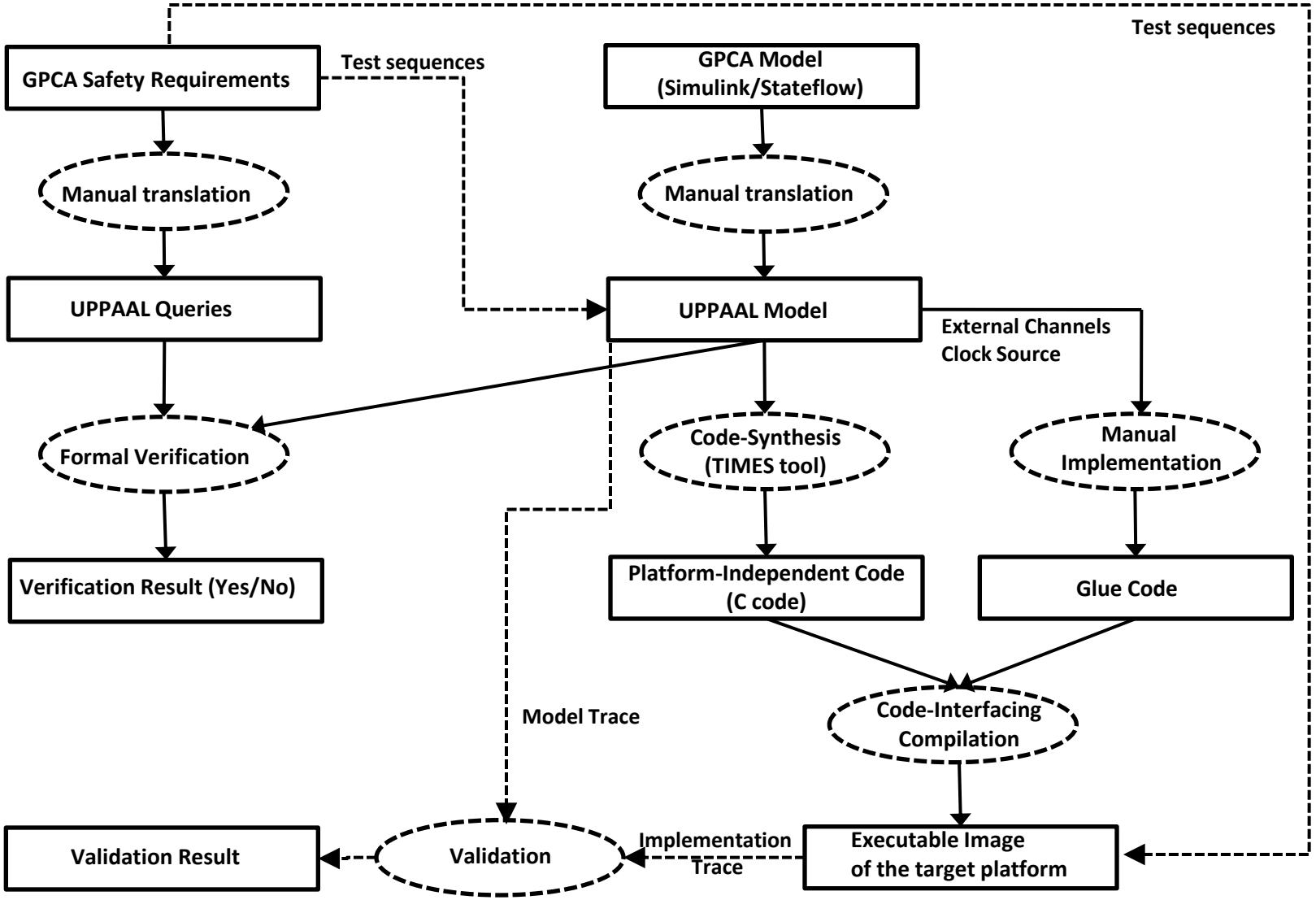




Starting Points

- Hazard analysis
  - Basis for safety requirements derivation
- Safety requirements
  - Determines properties in formal verification
- Design specification
  - Input to the code generation process
    - Via a separate formalization step

Model-based GPCA Implementation



Outcomes of the GPCA Case Study

- Set of artifacts
  - Prototype implementation
  - Formal models and formalized properties
- Development process
  - Still under construction
    - Dealing with platform-dependent code
- Safety argument
  - Generalized to a pattern for model-based development

Evaluation of Starting Points

- How good are the safety requirements?
  - Derived from hazard analysis (mitigation strategies)
    - Are there other sources?
  - Completeness and adequacy
    - Evidence of completeness is traceability
    - What is the evidence of adequate mitigation?
  - Level of abstraction
- In progress
  - In collaboration with Mats Heimdahl

Categorization of Properties

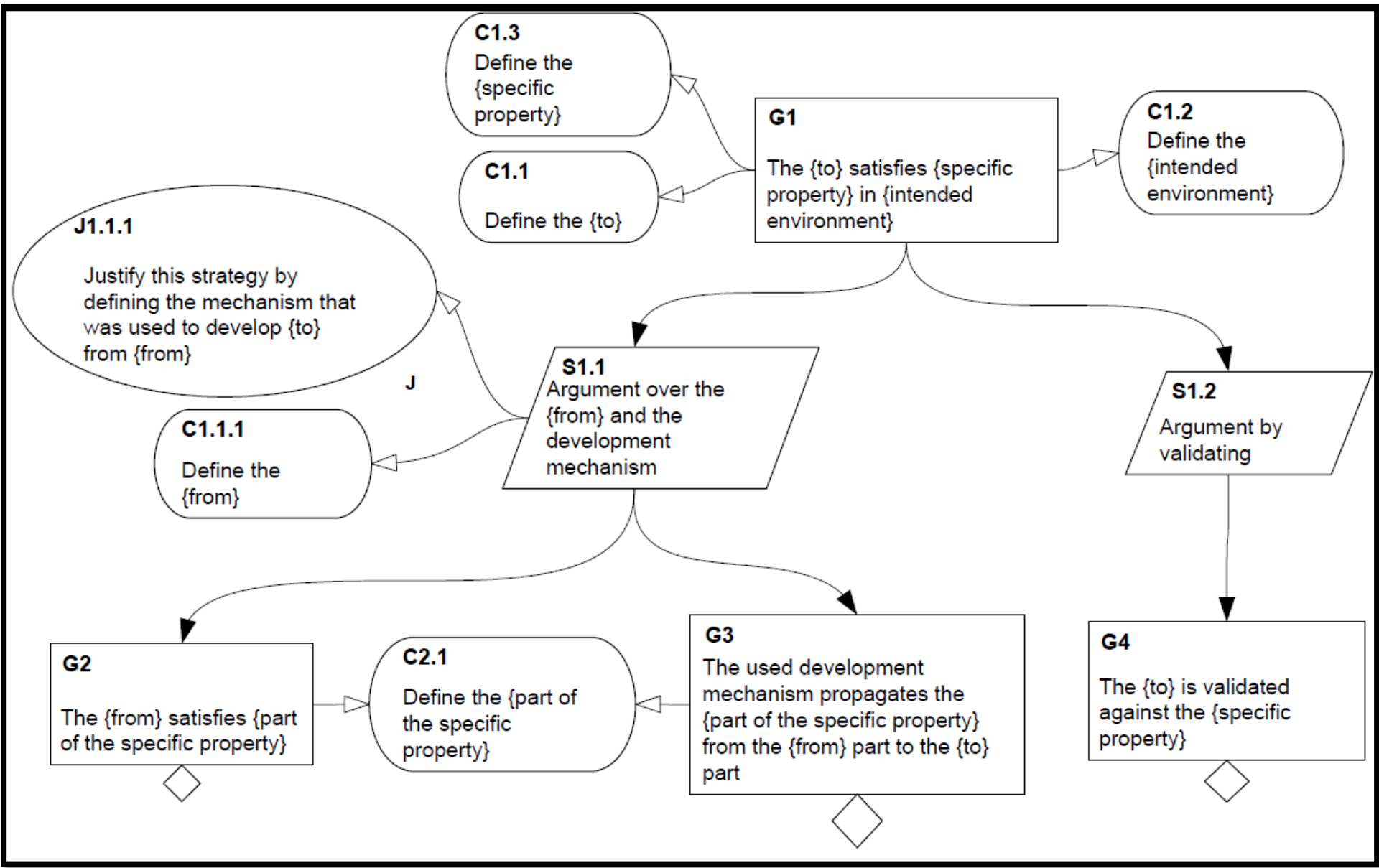
- Category 1: Properties that can be formalized and verified
- Category 2: Properties that are at a different level of abstraction than the model
  - Amount remaining shall be recalculated…
- Category 3: Properties that cannot be formalized but can be informally validated
  - Flow rate shall be programmable
- Category 4: Properties that need clarification
  - A clear indication shall be displayed…

From-To Pattern

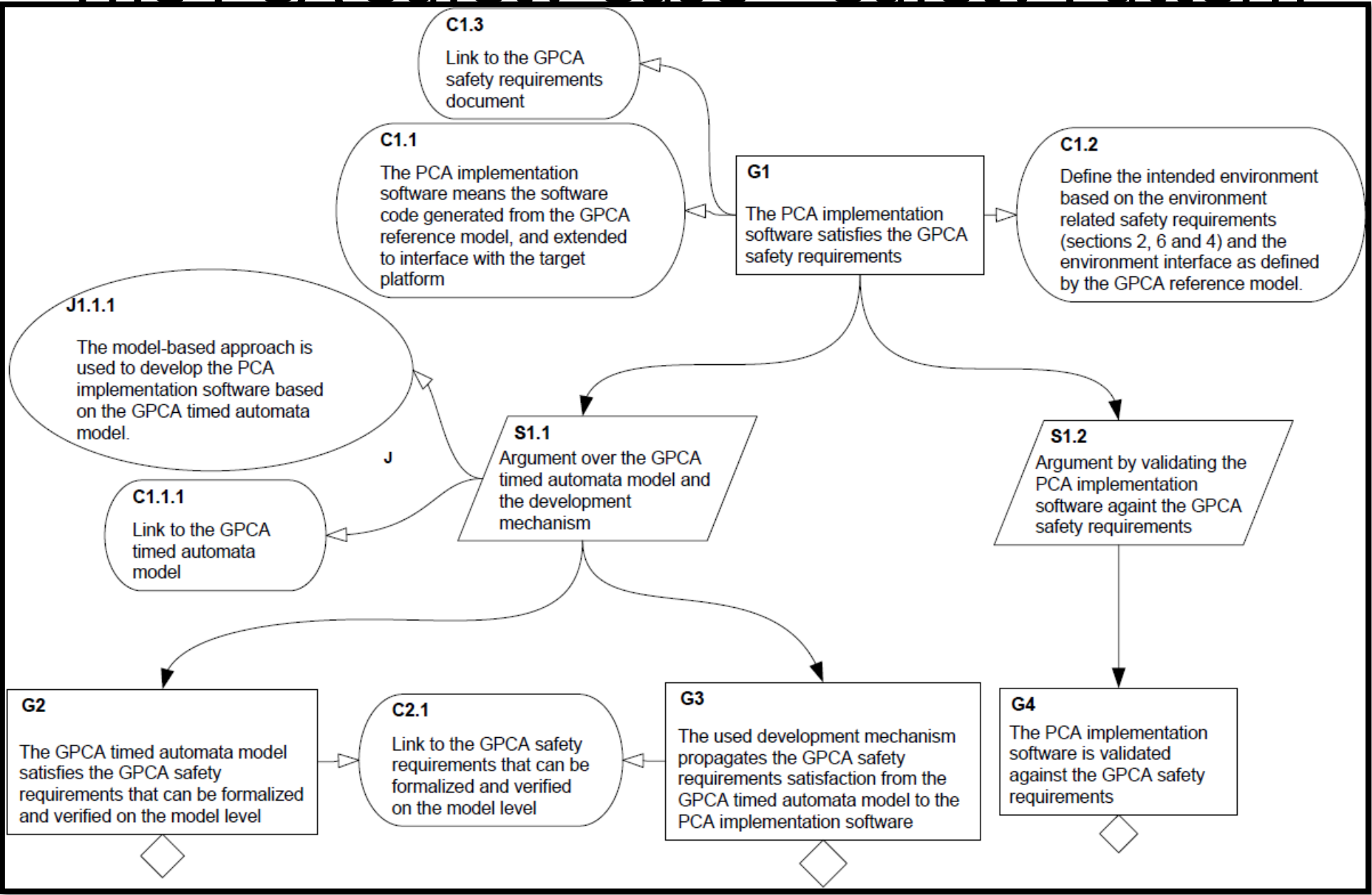
- Similarities in model-based development processes lead to similarities in safety arguments
  - From-To pattern captures these similarities

- Assurance through
  - Verification of properties in models
  - Preservation of properties through transformation

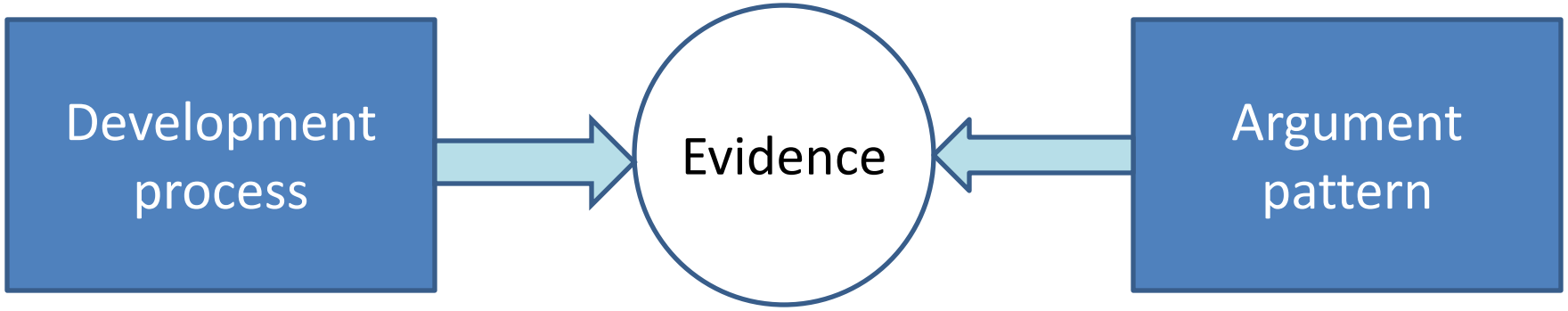
The PCA Safety Case – Safety Pattern



The PCA Safety Case – Safety Pattern



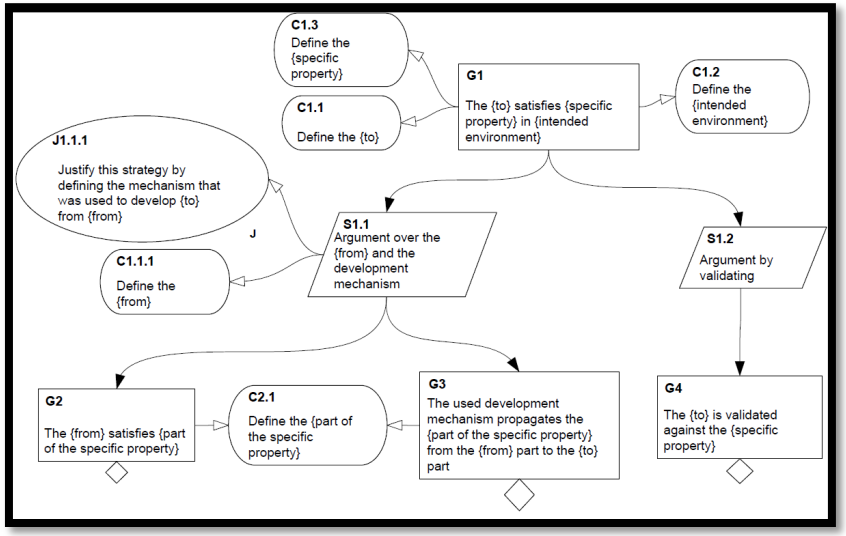
Lesson 1: What Evidence Is Needed?



- Structure of the safety argument determines kinds of evidence needed for assessing safety
  - An argument pattern implies the kinds of evidence needed in argument following this pattern
- Development process determines kinds of evidence that can be obtained

Evidence for From-To Pattern

- Model analysis results
  - Verification
  - Simulation
- Property preservation by the transformation
  - Correctness proofs
  - Tool qualification
- Validation
  - Evaluation of the outcome
  - Reasoning about modeling assumptions



Confidence in Evidence

- Required kind of evidence may be supplied by different evidence items
  - Different evidence items may vary in conclusiveness
  - E.g., test suites with different code coverage offer the same kind of evidence, but different confidence in the outcome
- Separation of safety argument from confidence argument

Lesson 2: Evidence Via Formalization

- Our approach relies on formal modeling and verification
  - Formalization of requirements is part of the process
- Formalization results may be (negative) evidence
  - Category 2: different levels of abstraction
    - Evidence of problems with the process or choice of formalism
  - Category 4: requirements too vague to formalize
    - Evidence of problems with requirements elicitation