Assurance Cases for Model-based Development of Medical Devices

Anaheed Ayoub, BaekGyu Kim,
Insup Lee, Oleg Sokolsky

Penn Engineering

PRECISE

Outline

- Introduction
  - State of the art in regulatory activities
  - Evidence-based certification
- Research goals
- Case study
  - Learn by doing
- Methodology
  - Generalize lessons from the case study

State of the Art

- Regulatory approval with respect to a set of safety standards
  - E.g., ISO 9001 (quality management) and ISO 14971 (risk management)
  - Assurance in terms of process
- Evidence in the form of
  - Process checklists
  - Variety of artifacts for regulators to look at

Concerns about State of the Art

- Does not directly evaluate product
  - Good process is necessary but not sufficient
  - Evidence-based certification is the vision
- Perceived high cost
  - Some regulation might be overkill
  - Much activity not directly related to development
- Tight process standards hinder innovation
  - Conservative designs due to regulatory risk (?)
  - Slow adoption of new development practices
  - Growing system complexity threatens to overwhelm existing processes

Challenges of novelty

- Evidence-based certification
  - To gain adoption, we need to understand what works and what does not
    - How do we organize and evaluate evidence
- Model-driven development (MDD)
  - Detects problems early
  - Enables generative techniques
  - New regulatory challenges:
    - Abstractions and assumptions – new validation needs
    - Increased reliance on tools

Claim, Evidence, and Argument

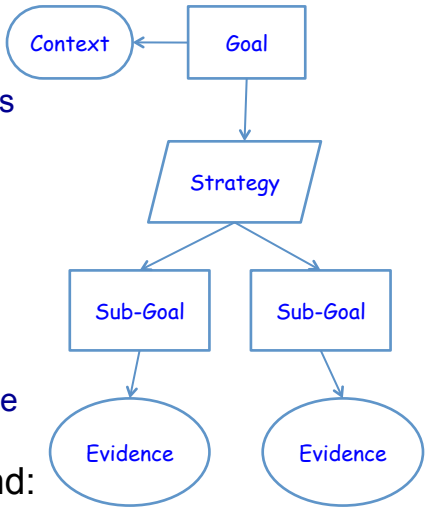
- Explicit Claims
  - State explicitly what properties (safety, security, reliability, performance, etc.) the system must possess and under which assumptions
- Supporting Evidence
  - Results of observing, analysing, testing, simulating and estimating the properties of a system that provide the fundamental information from which safety can be inferred
- High Level Arguments
  - Explanation of how the available evidence can be reasonably interpreted as indicating acceptable dependability

  - Argument without Evidence is unfounded
  - Evidence without Argument is unexplained

- Tim Kelly, 2008

Assurance Cases

- To construct an assurance case we need to:
  - make an explicit set of claims about the system
  - produce the supporting evidence
  - provide a set of arguments that link the claims to the evidence
  - make clear the assumptions and judgments underlying the arguments
- Safety case is a special kind:
  - Claims are limited to safety



Potential Pitfalls

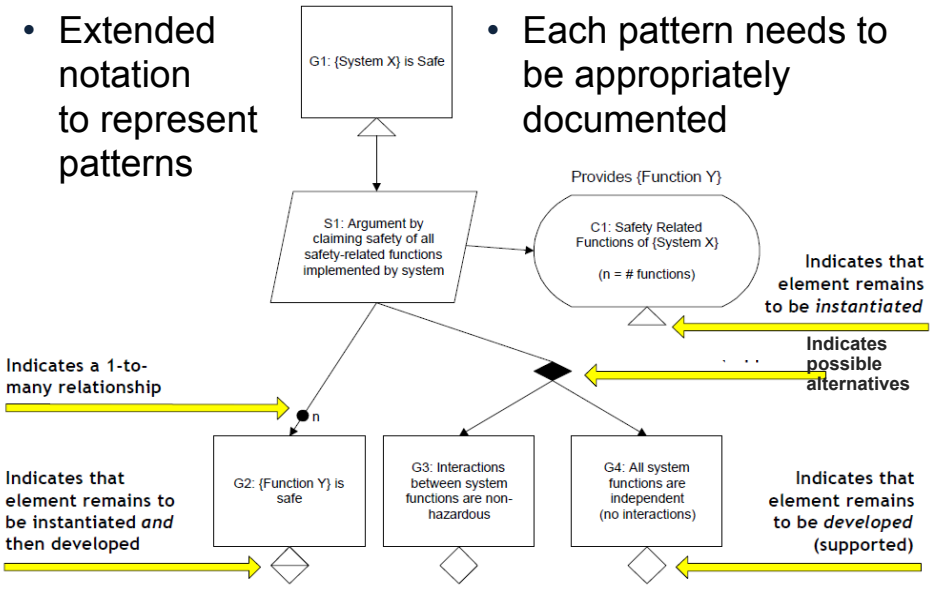
- A poorly constructed assurance case gives false confidence
  - Manufacturers lack experience in creating ones
- A poorly evaluated assurance case lets errors slip
  - "Spagetti argument" is hard to evaluate
- How to avoid pitfalls?
  - Keep the argument simple
  - Reuse successful arguments

Keeping It Simple

- Separate safety argument from confidence argument
- Safety argument
  - Reasoning about safety of the device
    - E.g.: formal verification + code generation implies requirement satisfaction
- Confidence argument
  - Reasoning about confidence in the safety argument, assumptions, evidence
    - E.g.: Tools are trustworthy and were appropriately applied

Assurance Case Patterns

- Extended notation to represent patterns
- Each pattern needs to be appropriately documented



Research Goals

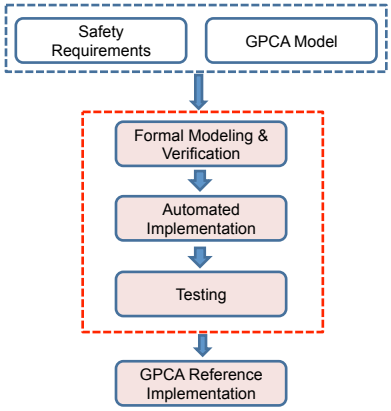
- Determine appropriate patterns for assurance cases
  - Incorporate guarantees of formal methods and code generation into the argument
- Develop techniques for identifying gaps in the argument
  - Aim to create a methodology that system developers can follow to create good assurance cases

Case Study: Infusion Pump

- Deceptively simple
  - 2005—2009: 56, 000 adverse event reports; 87 recalls
    - 1% deaths, 34% serious injuries
- Case study goals
  - Show how to do it right
    - Develop good requirements
    - Apply rigorous development
    - Explore assurance case constructio..
  - Provide guidance to manufacturers

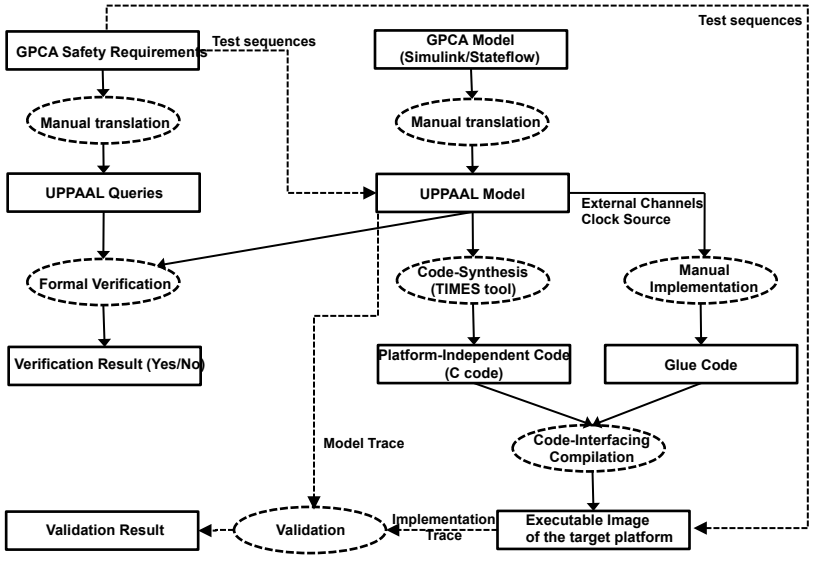
GPCA reference implementation

- FDA initiated
  - GPCA Safety Requirements
  - GPCA Model (Simulink/ Stateflow)
- Develop a GPCA reference implementation
- Provide evidence that the implementation satisfies the safety requirements
  - Safety argument
  - Confidence cases



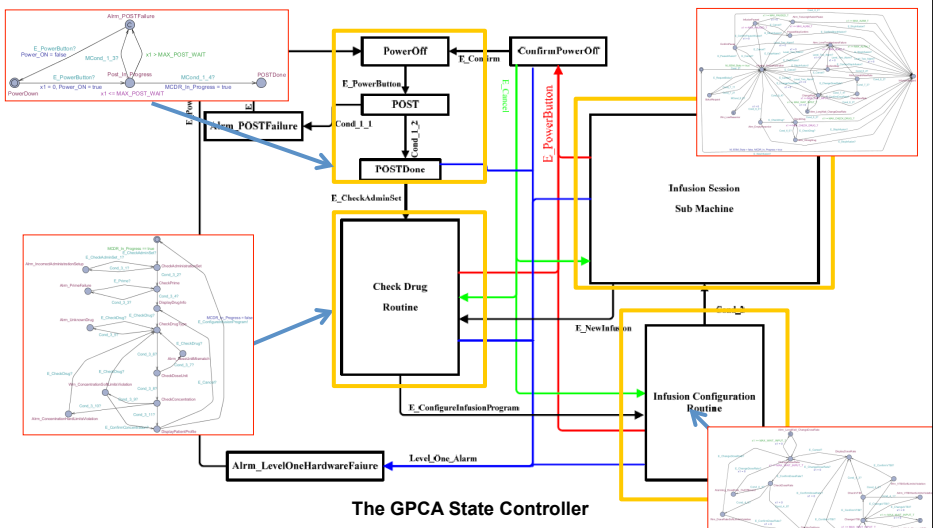
Model-Based Development of GPCA Reference Implementation

Model-based GPCA Implementation



[Kim et al, EMSOFT 2011]

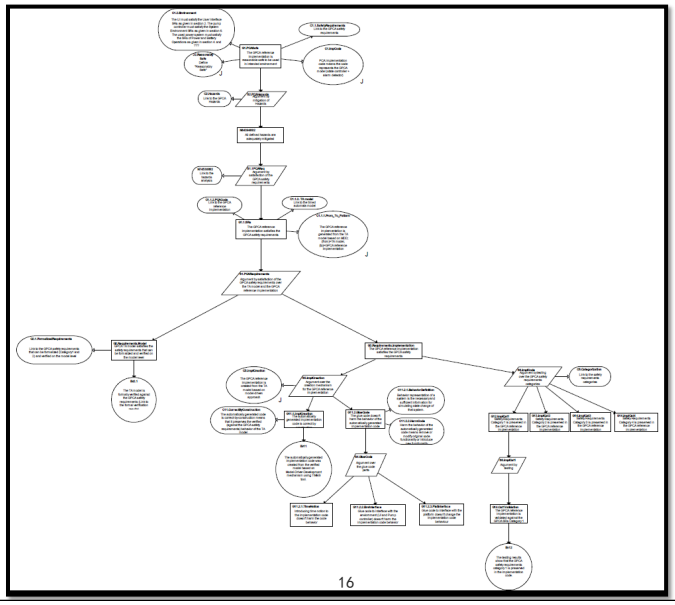
Formalization of the GPCA model

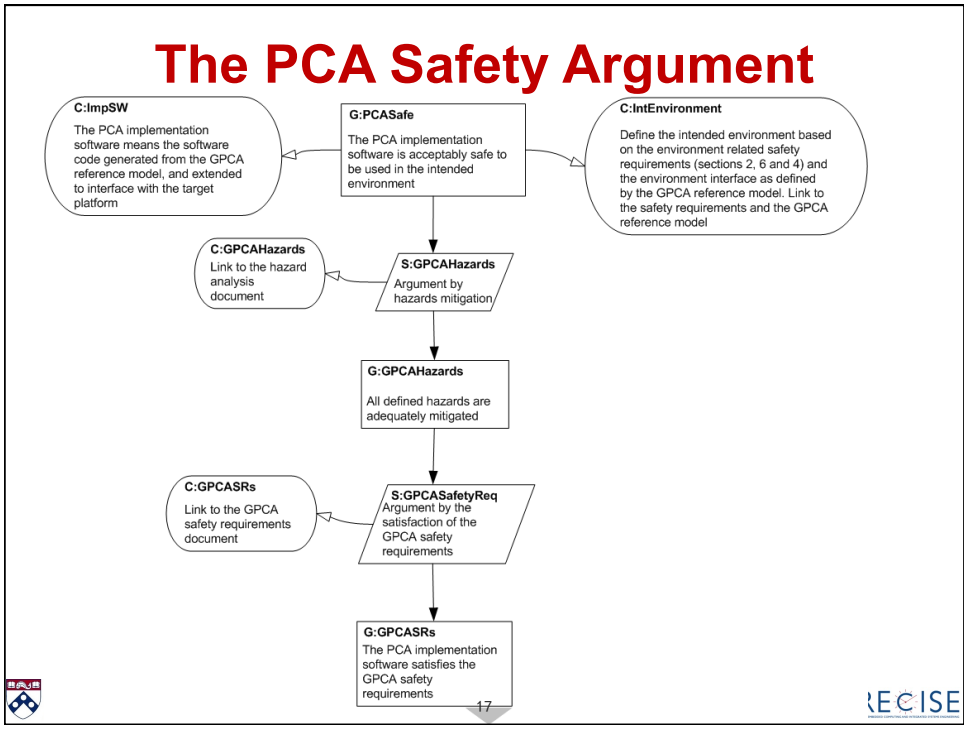


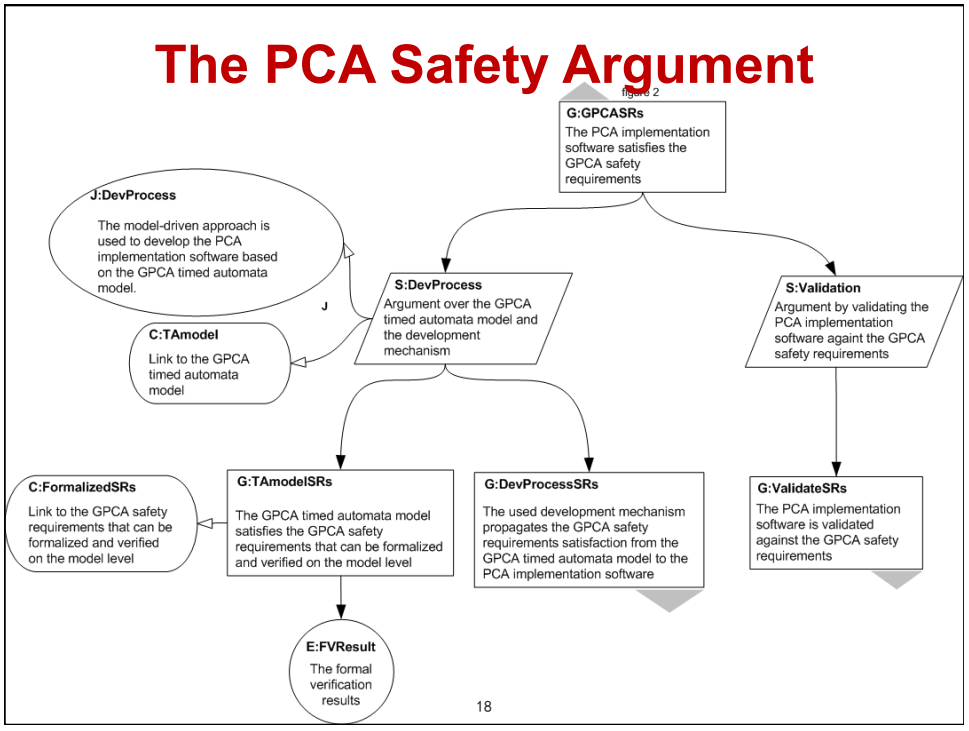
The GPCA State Controller

- Rigorous translation procedure ensures full traceability

The PCA Safety Argument



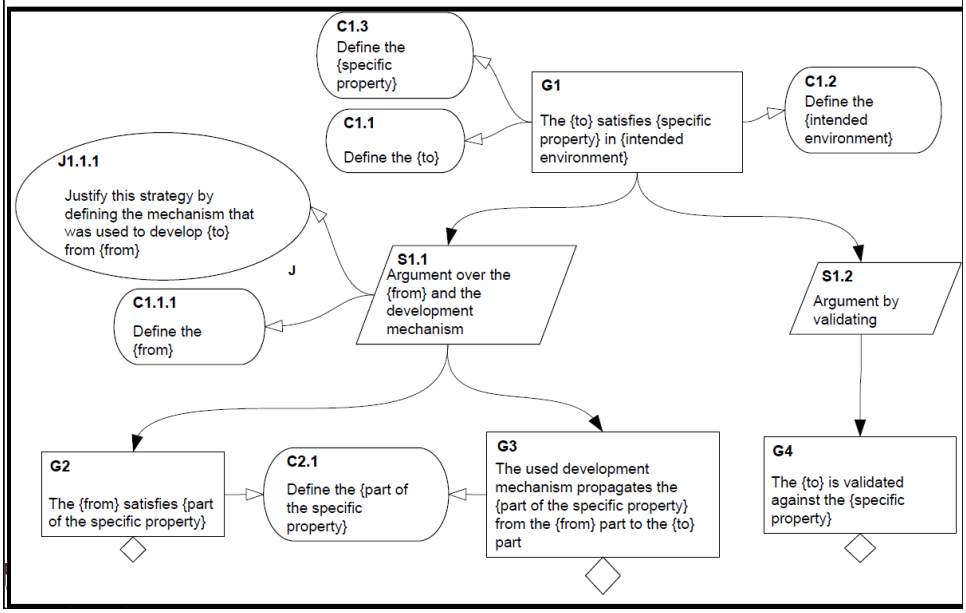

The PCA Safety Argument
C:ImpSW
The PCA implementation software means the software code generated from the GPCA reference model, and extended to interface with the target platform
G:PCASafe
The PCA implementation software is acceptably safe to be used in the intended environment
C:IntEnvironment
Define the intended environment based on the environment related safety requirements (sections 2, 6 and 4) and the environment interface as defined by the GPCA reference model. Link to the safety requirements and the GPCA reference model
C:GPCAHazards
Link to the hazard analysis document
S:GPCAHazards
Argument by hazards mitigation
G:GPCAHazards
All defined hazards are adequately mitigated
C:GPCASRs
Link to the GPCA safety requirements document
S:GPCASafetyReq
Argument by the satisfaction of the GPCA safety requirements
G:GPCASRs
The PCA implementation software satisfies the GPCA safety requirements
17
RECISE


The PCA Safety Argument
G:GPCASRs
The PCA implementation software satisfies the GPCA safety requirements
J:DevProcess
The model-driven approach is used to develop the PCA implementation software based on the GPCA timed automata model.
J
C:TAmodel
Link to the GPCA timed automata model
S:DevProcess
Argument over the GPCA timed automata model and the development mechanism
S:Validation
Argument by validating the PCA implementation software againt the GPCA safety requirements
C:FormalizedSRs
Link to the GPCA safety requirements that can be formalized and verified on the model level
G:TAmodelSRs
The GPCA timed automata model satisfies the GPCA safety requirements that can be formalized and verified on the model level
G:DevProcessSRs
The used development mechanism propagates the GPCA safety requirements satisfaction from the GPCA timed automata model to the PCA implementation software
G:ValidateSRs
The PCA implementation software is validated against the GPCA safety requirements
E:FVResult
The formal verification results
18

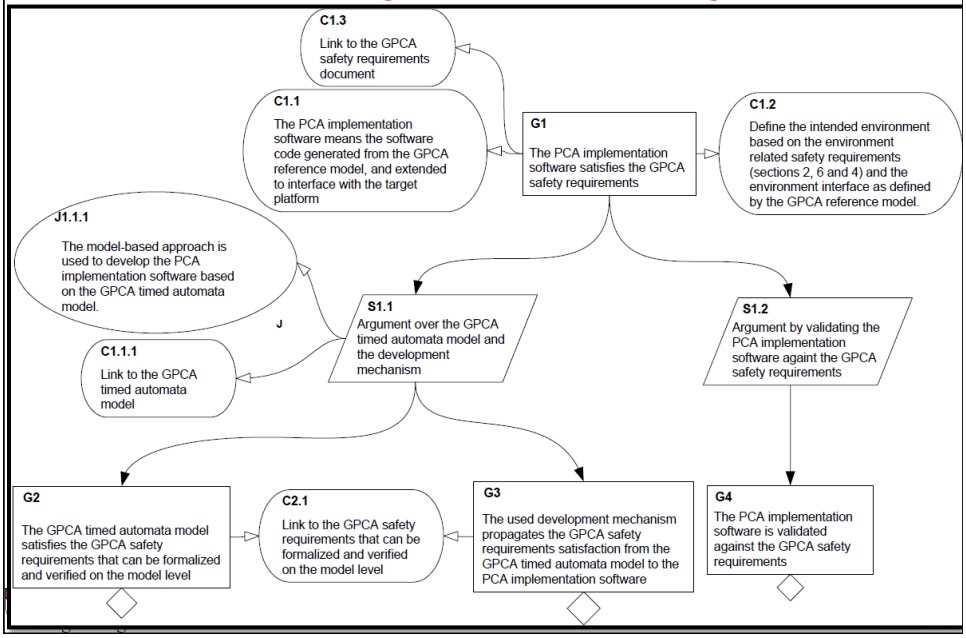
Beyond the Case Study

- Reusing the argument
  - Safety case pattern for model-based development
  - Capture the assurance of formal verification and generative development techniques
- Evaluating the safety argument
  - Where are the gaps?

The PCA Safety Case – Safety Pattern



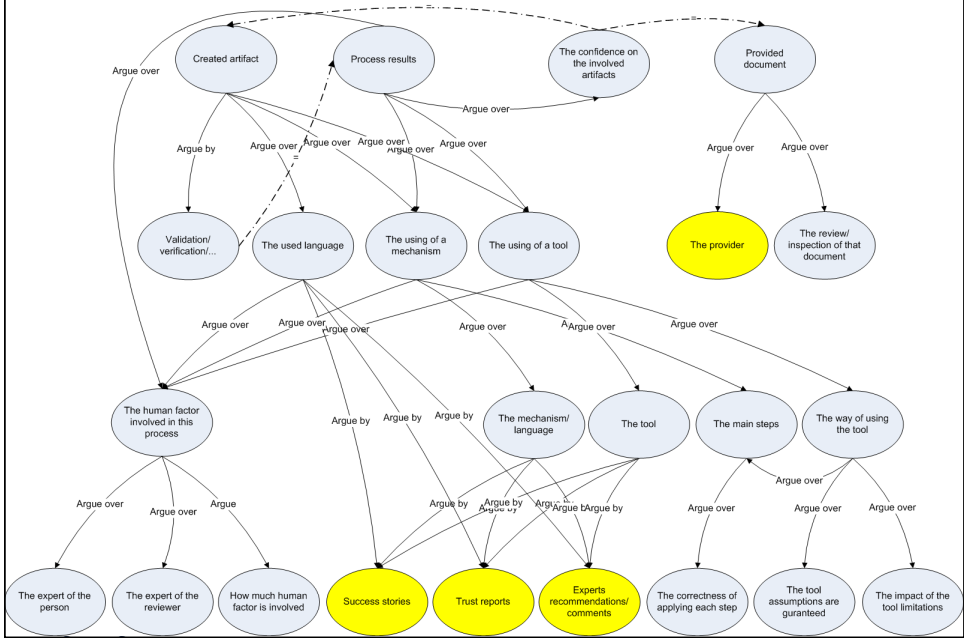
The PCA Safety Case – Safety Pattern



Confidence Case Construction

- We need a mechanism to
  - Systematically construct confidence arguments
  - Identify safety gaps (assurance deficits)
- Generalize experience from GPCA case study
  - Identify common characteristics of concepts that require confidence argument
  - Summarize relationship between the concepts in a map
    - We target trustworthiness; appropriateness is similar

Common Characteristics Map



Map Instantiation

C:TAmodel

Link to the GPCA timed automata model

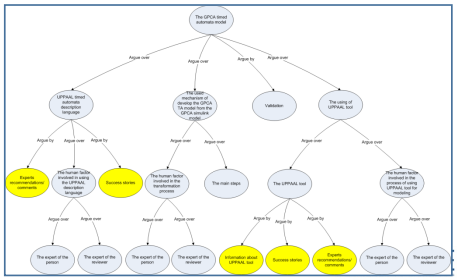
- Given a GSN element, pick the corresponding map node
- Unroll the map
- Find affected GSN elements, repeat

The common characteristics map

The Safety Argument Context or Evidence

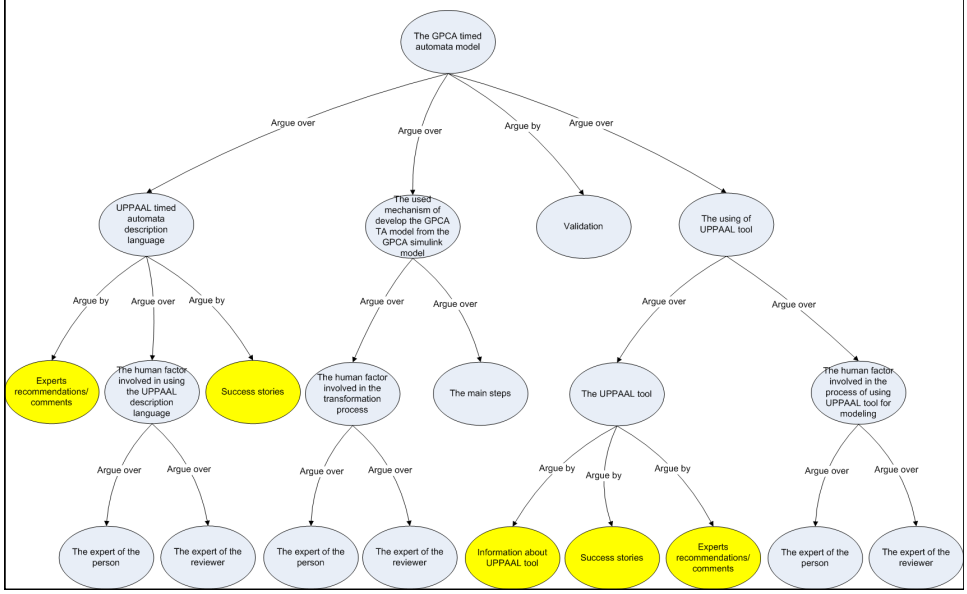
AND

Instantiate the common characteristics map



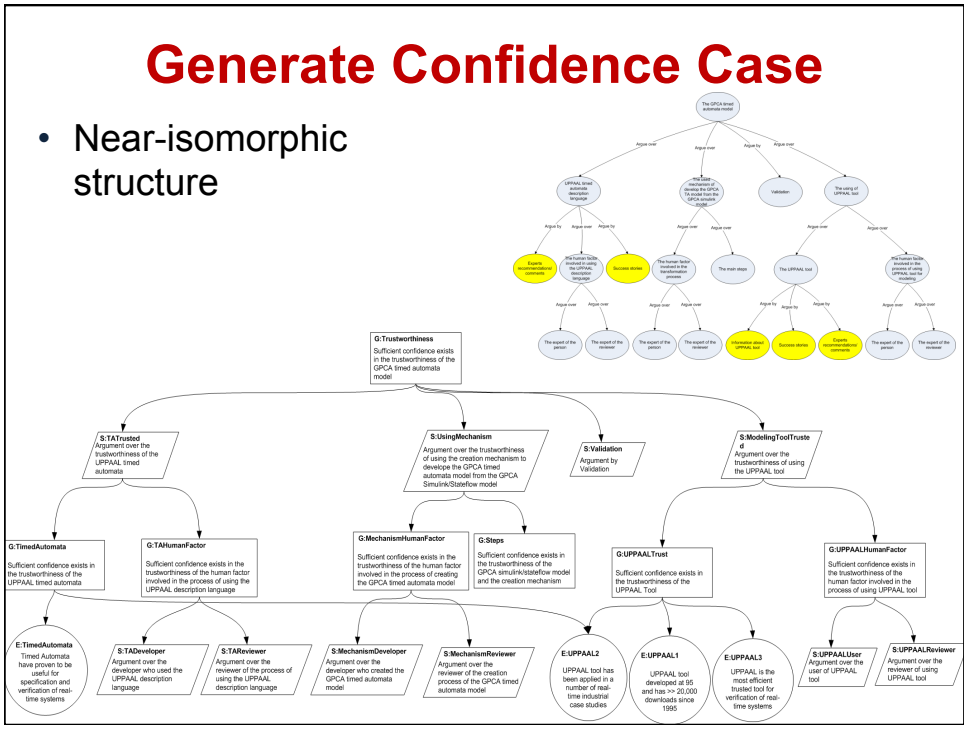
Penn Engineering

Instantiated Map



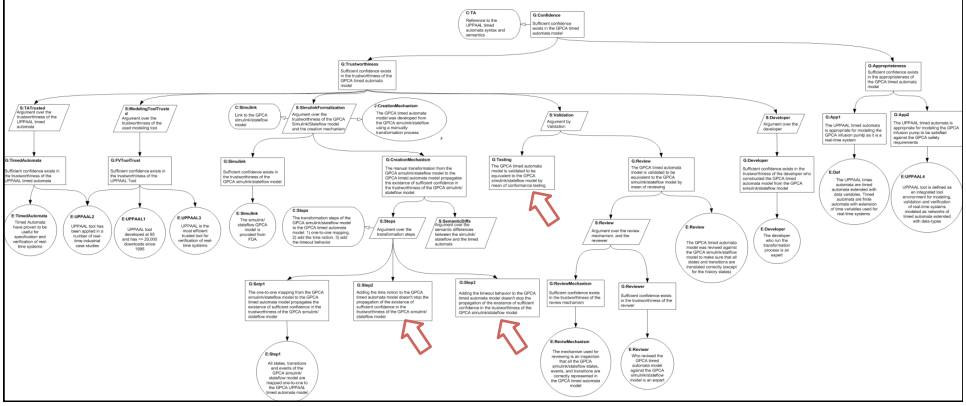
Generate Confidence Case

- Near-isomorphic structure



Identify safety gaps

- Look for branches that do not end with evidence nodes



Conclusions

- In order for assurance cases to work in practice, we need to
  - Determine effective ways to construct them
  - Systematically tie in all the relevant evidence (and no other)
- Case studies are useful:
  - A way to gain experience
  - A source of examples for the community
- Generalization of experience is the next step
  - Under way

Outreach / Technology Transfer

- Collaboration with FDA
  - Frequent visits to compare visions and coordinate plans
- Guidance for manufacturers
  - GPCA case study
    - All artifacts will be freely available to the community, including safety case
  - Some GPCA aspects are already used by manufacturers in preparing 510(k) submissions