Data Artifacts for Airworthiness of the Swift UAS

Ewen Denney and Ganesh Pai
SGT / NASA Ames Research Center
{ewen.denney, ganesh.pai}@nasa.gov

UAS in the NAS Annual Meeting, NASA Ames

Regulatory Environment

- FAA
  - FARs, e.g., 14 CFR, part 21 Certification Procedures for Parts and Procedures
  - Advisory Circulars e.g., AC 21 – 12 Airworthiness certification
  - Policy / Guidance documents e.g., Interim guidance 08-01
  - Orders, e.g., 8130.34 (B) – Airworthiness certification
  - …

- NASA aircraft have addition regulatory constraints
  - NPR 7123.1A - NASA Systems Engineering Processes and Requirements
  - NPR 7900.3B - Aircraft Operations Management Manual
  - NASA-STD-8719.13 - NASA Software Safety Standard
  - NASA-STD-8739.8 - Software Assurance Standard
  - NPR 8715.5A - NASA Range Flight Safety Program
  - NPR 8715.3C - NASA General Safety Program Requirements
  - APR 8705.1 - System Safety and Mission Assurance
  - …

Authority to Fly

- Operations
  - Certificate of Authorization / Waiver (COA)
    - Required for flight
      - Demonstrate airworthiness
      - Flight operations safety
        - » Procedures for emergencies, lost link, lost communications
        - » Range safety
      - Aircrew qualification
    - FAA also controls access to airspace
      - Together with ATC for the operational area
  - NASA can self certify
    - Airworthiness and Flight Safety Review Board
    - Flight readiness review, after COA

- COAs mainly apply to public entities
- Private proponents must get a SAC-Exp

COA Process

- Proponent files application → Administrative Review → Feasibility review
  - Check against the existing policy documents and regulations
  - Safety review by UAPO

- Broad data requirements
  - Safety issues
  - Airworthiness
  - Contingency management and Range Safety
  - Reporting and Notification

COA Process: Safety Issues

- COA proponent need not perform and submit a comprehensive hazard analysis
  - Rather,
    - Hazard analysis of the wider system → NAS modification
  - Five hazard clusters identified for UAS operations in Class D airspace (SRMD)
    - Cluster A: Loss of communications / control links
    - Cluster B: Other system issues
    - Cluster C: Internal and external visual limitations
    - Cluster D: Outside interferences and intrusion
    - Cluster E: UAS operations team training
  - Identification of generic safety requirements for operating in class D airspace
    - Automatic flight termination, automatic return capability, etc.

COA Process: Airworthiness

- APR 1740.1 (At NASA Ames)
- JO-3 (Flight operations manual)
- AFSRB looks for due diligence
- Flight readiness review (FRR) after COA

COA Data Requirements

- Proponent description
- Operational Description
- System Description
- Performance Characteristics
- Airworthiness statement
  - MIL-HDBK-516B
  - NPR 7900.3C
  - APR 1740.1

- Procedures for lost link/mission, communication and emergencies
- Onboard avionics and equipment
- Lights/Spectrum Analysis
- ATC communications
- Surveillance / Detection capabilities
- Flight operations areas / plans
- Aircrew qualifications and special circumstances
- Launch / Recovery

Swift UAS Specific Data

- Emergency Procedures
- Aircraft Systems – Communications
  - Two configurations / modes
- Visual observers
- Lost communication procedures

Application Context

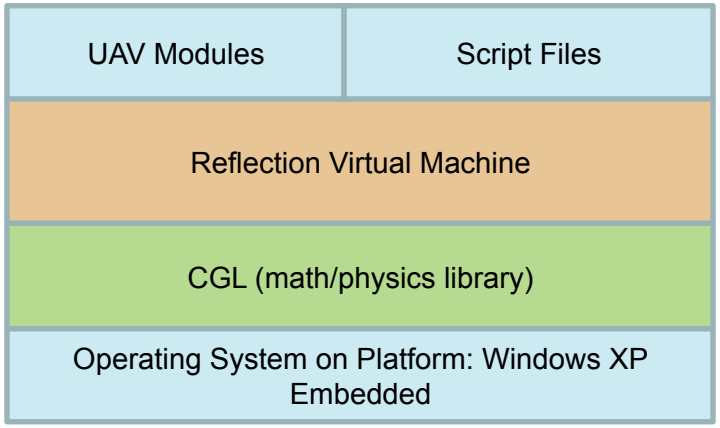
- Target System – Swift Unmanned Aircraft System (UAS)

- Experimental Autonomous Vehicle Program
  - Research vehicle program, Code TI, NASA Ames

- Electric motor, lithium batteries, high glide ratio, all composite wing structure, steel/aluminum fuselage frame

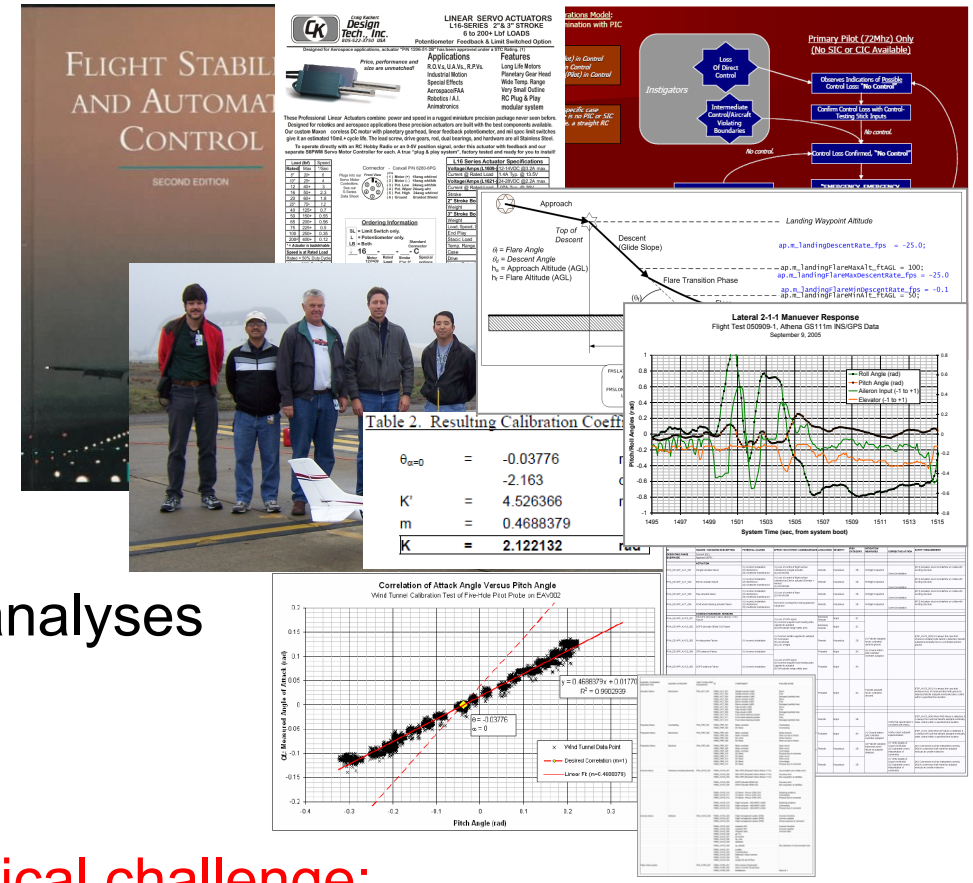
Application Context

- Swift configuration
  - Very low latency, computer controlled with multiple onboard full-power CPUs
  - UAS consists of ground system (GSC), flight system (UAV), and communication infrastructure
- Reflection architecture
  - C and C++ component-based plug-and-play infrastructure
  - Real-time embedded avionics system architecture
- Commands in Reflection script, uploaded from ground system, and interpreted by onboard VM



Data Sources

- A safety argument requires the integration of diverse sources of information
  - Mathematical theory
  - Flight test maneuvers
  - Calibration experiments
  - Manufacturer datasheets
  - Flight operations procedures
  - Software verification
  - Systems and software safety analyses
  - Expert opinion



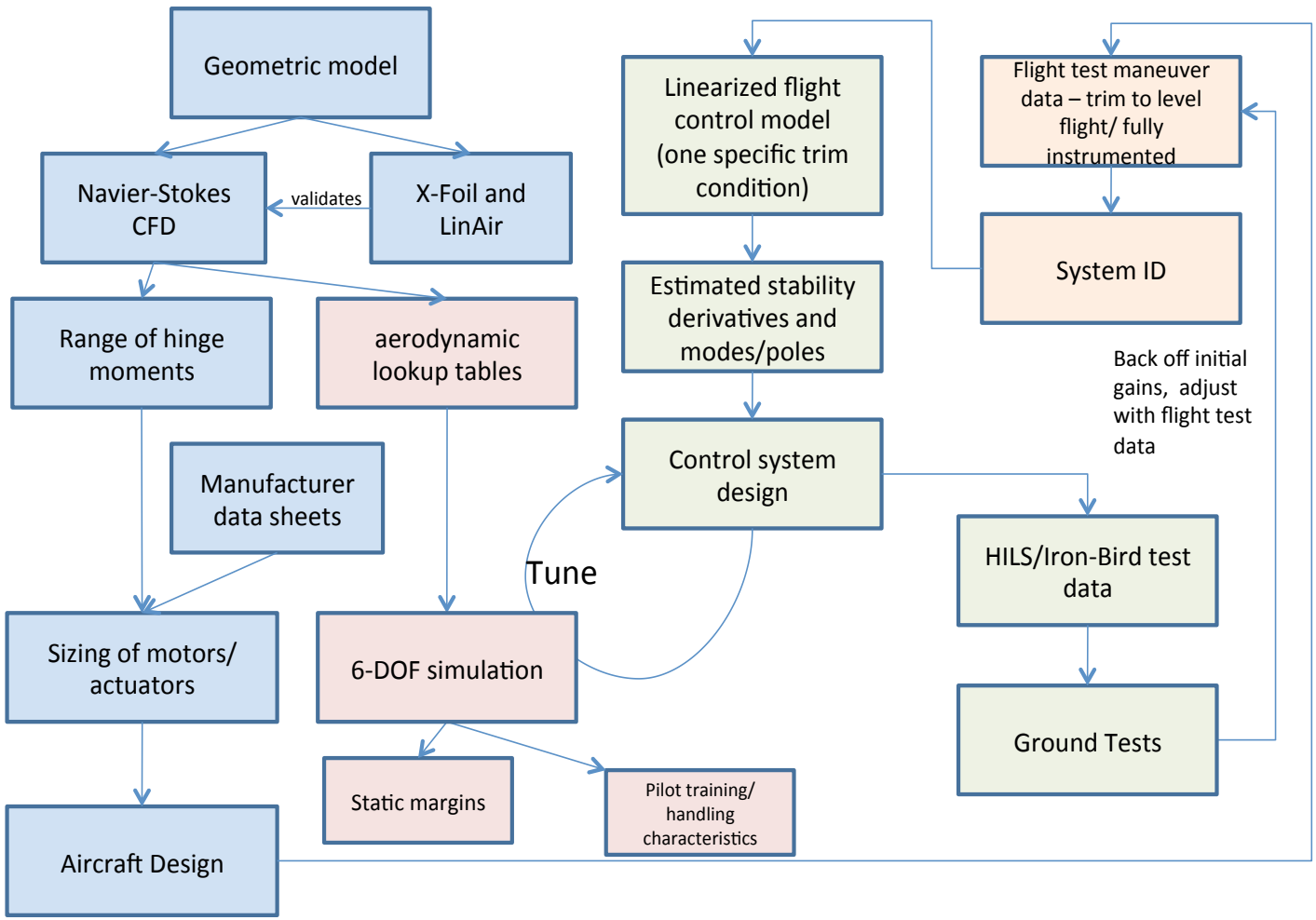
Specific technical challenge:

A methodology and framework that allows integrated reasoning about disparate forms of evidence

Future Plans: Expanding Scope of Safety Analysis

SWIFT UAS Design Flow

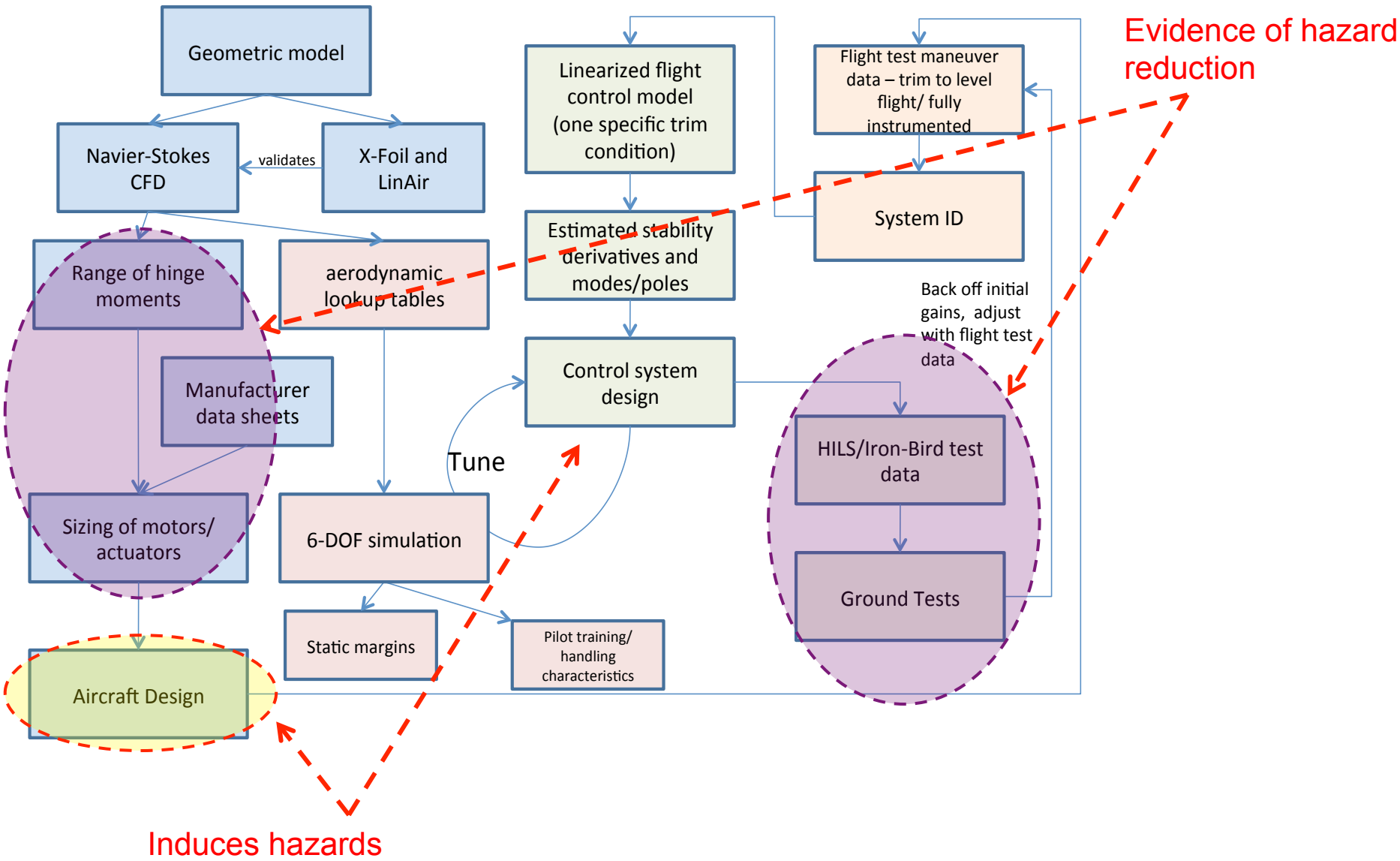
Aircraft Design → Simulation → Controller Design → Flight Test



Future Plans: Expanding Scope of Safety Analysis

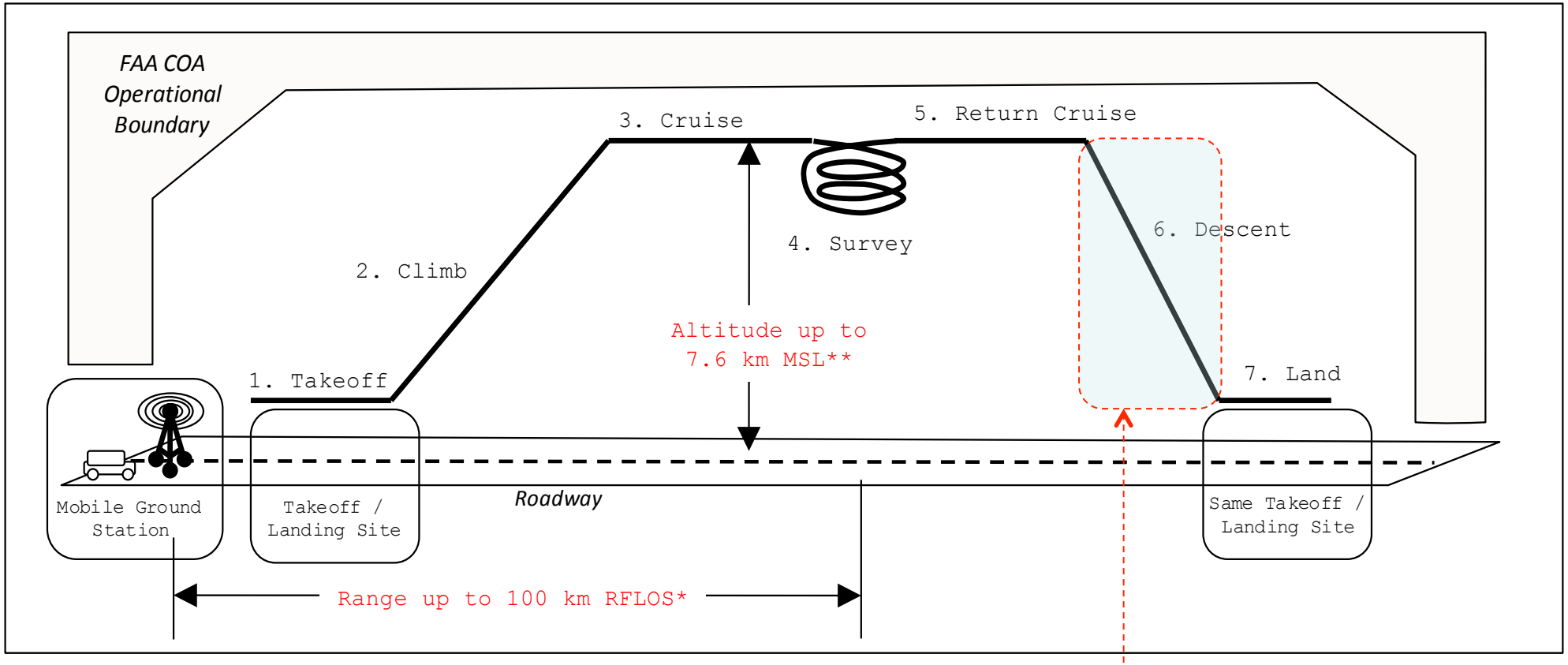
SWIFT UAS Design Flow

Aircraft Design → Simulation → Controller Design → Flight Test



Application Context

Swift UAS Concept of Operations*

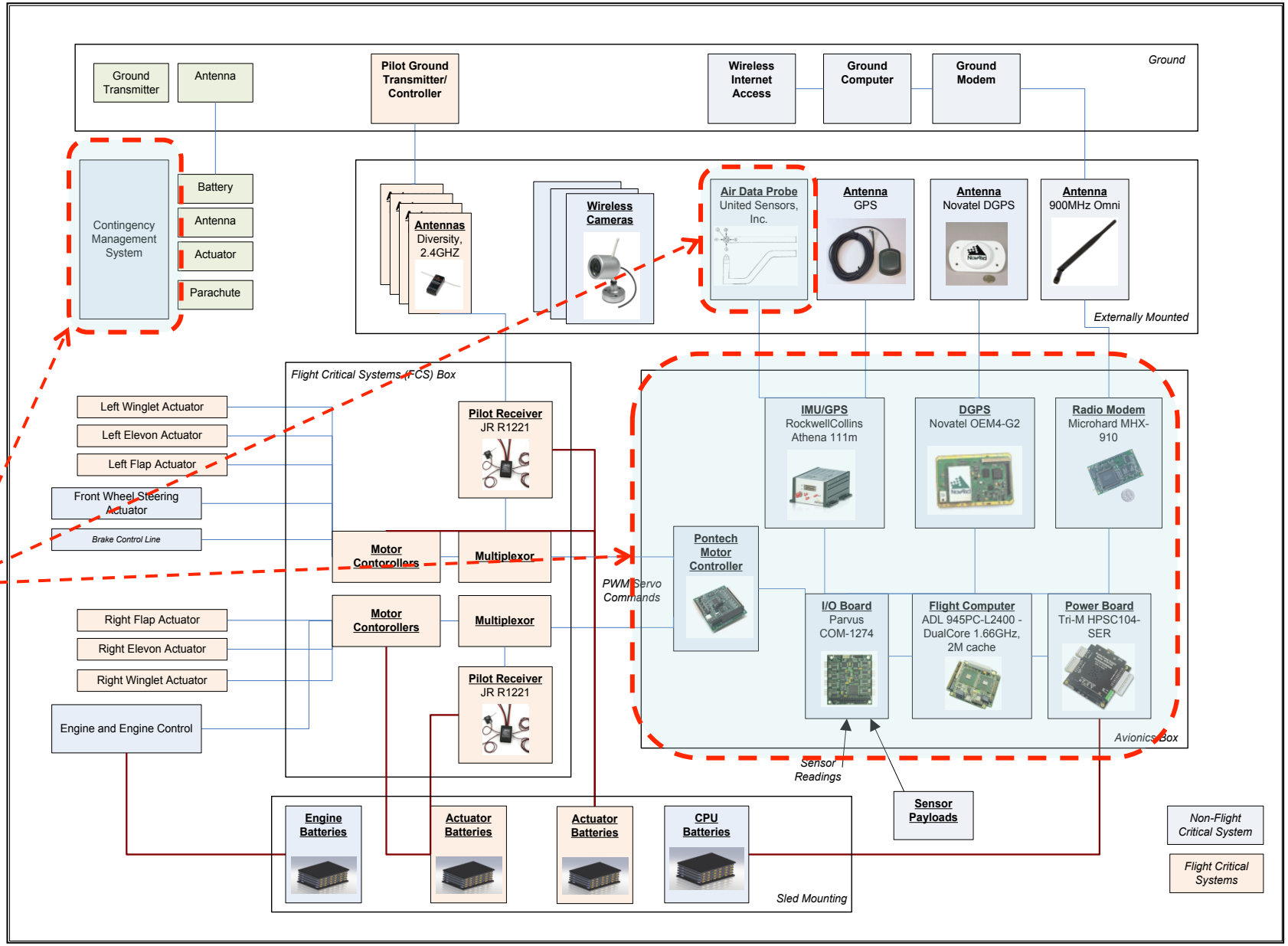


We've mainly looked at the *Descent* phase of operation

* Corey Ippolito, Swift Design Management Plan.

Assurance Scope

Swift UAS Architecture*



We've mainly looked at the Contingency Management System, Air Data Probe, and the Avionics Box

* Corey Ippolito, SWIFT UAS Design Management Plan

Context

- Claiming (Software) Safety
  - Current practice
    - Standards-based
      - Following a process prescribed/ recommended in the standard
      - Processes recommended correspond to reduction of failure probability, for given Safety Integrity Levels* (SIL) and Design Assurance Levels** (DAL)
      - Relationship between SIL / DAL and failure probabilities is not well understood nor sufficiently justifiable
  - State of the art
    - Goal-based safety argumentation AKA Safety Cases
      - Focus on product assurance
      - Link to safety of the system is explicit via arguments and evidence
      - Justify measures taken for hazard elimination / mitigation
      - Representation in various ways (graphical / structured text)

* ISO / IEC 61508
** DO-178B

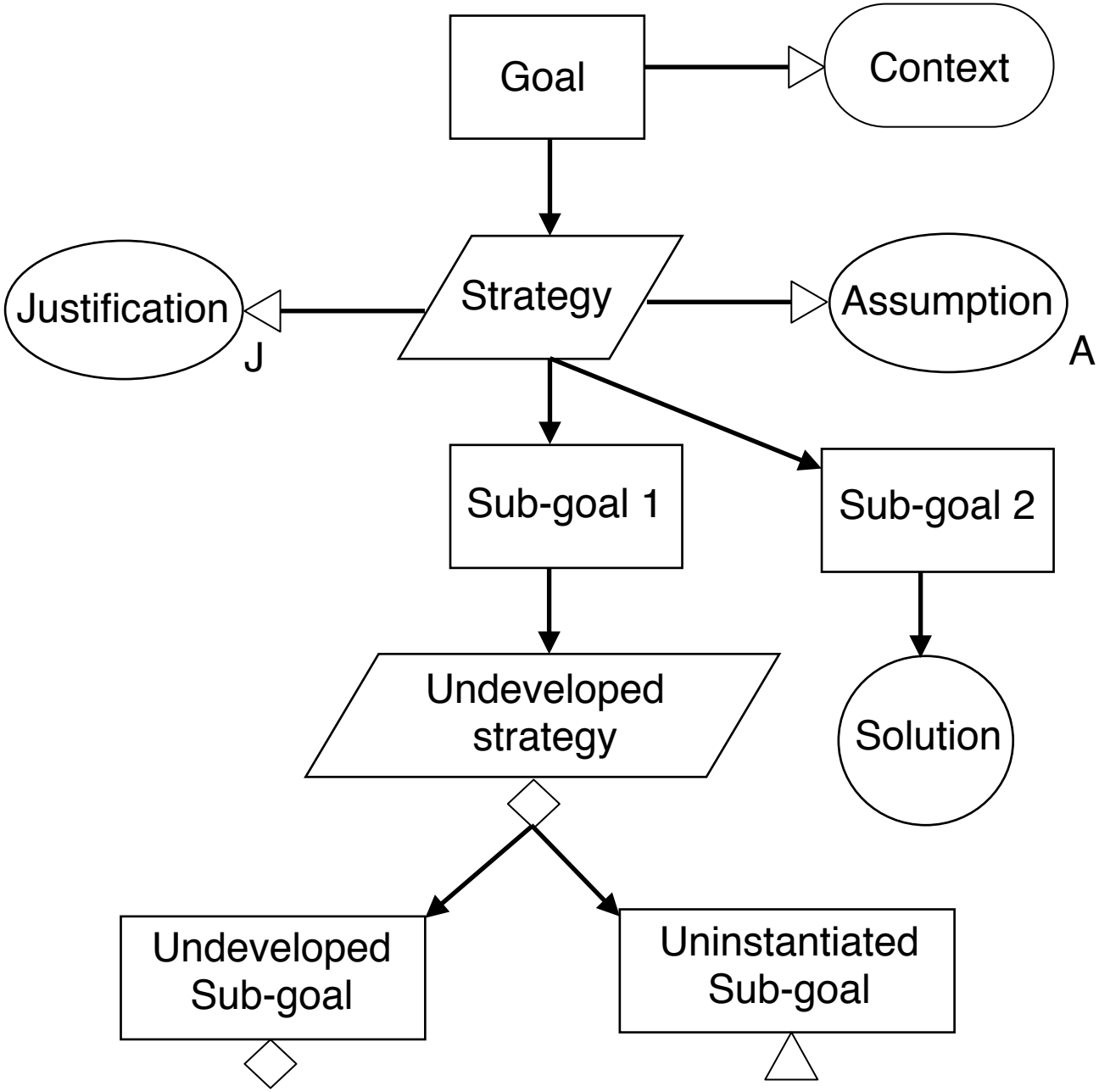
Some Terminology

- Safety[1, 2]
  - Freedom from those hazards considered as presenting unacceptable mishap risk.
  - Interpreted as the reduction of mishap risk to acceptable levels.

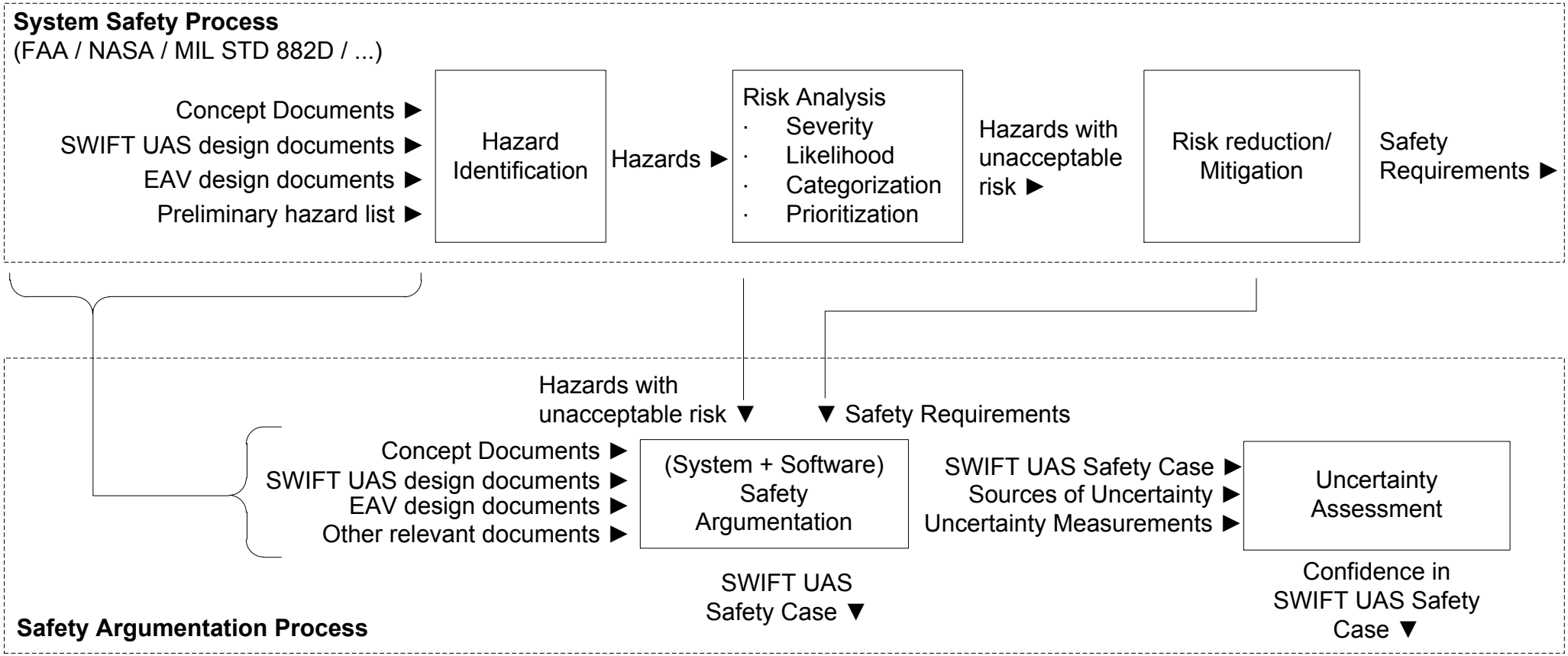
- Safety case[3]
  - A structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment.

[1] NASA Software Safety Standard (NASA-STD-8719.13B) Jul. 2004.
[2] US Dept. of Defense, MIL-STD-882D, Feb. 2000
[3] UK Ministry of Defence, DEF-STD-00-56, Part 1, Issue 4, June 2007

Goal Structuring Notation (GSN)



Safety Methodology



LEGEND:

<Data> ▶ : Data flow

<Data> ▼ : Data flow

▭ : Process / Process step

Philosophy:

Safety argumentation process driven by system safety process

* NASA Software Safety Standard (NASA-STD-8719.13B) Jul. 2004 and NASA Software Safety Guidebook (NASA-GB-8719.13), Mar. 2004

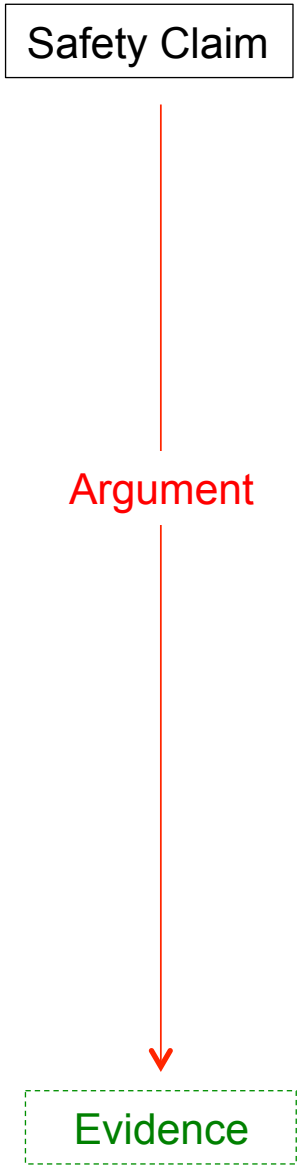
Safety Methodology

- Safety Argumentation Process
  - Applied starting at the level of the system
    - Safety is always considered in the system context
  - Process repeated at the software level
    - Software safety argument created, in part, through a software verification methodology
  - Phased development
    - System development process influences safety argument
    - Safety argumentation process influence system development

  - System + Software Safety Argumentation
    - Iteratively argue that all relevant hazards are eliminated or mitigated
      - Create safety claims
      - Link evidence to claims via a structured argument
      - Use Goal Structuring Notation (GSN)

  - Uncertainty Assessment
    - Identify sources of uncertainty
    - Model and quantify uncertainty (confidence)

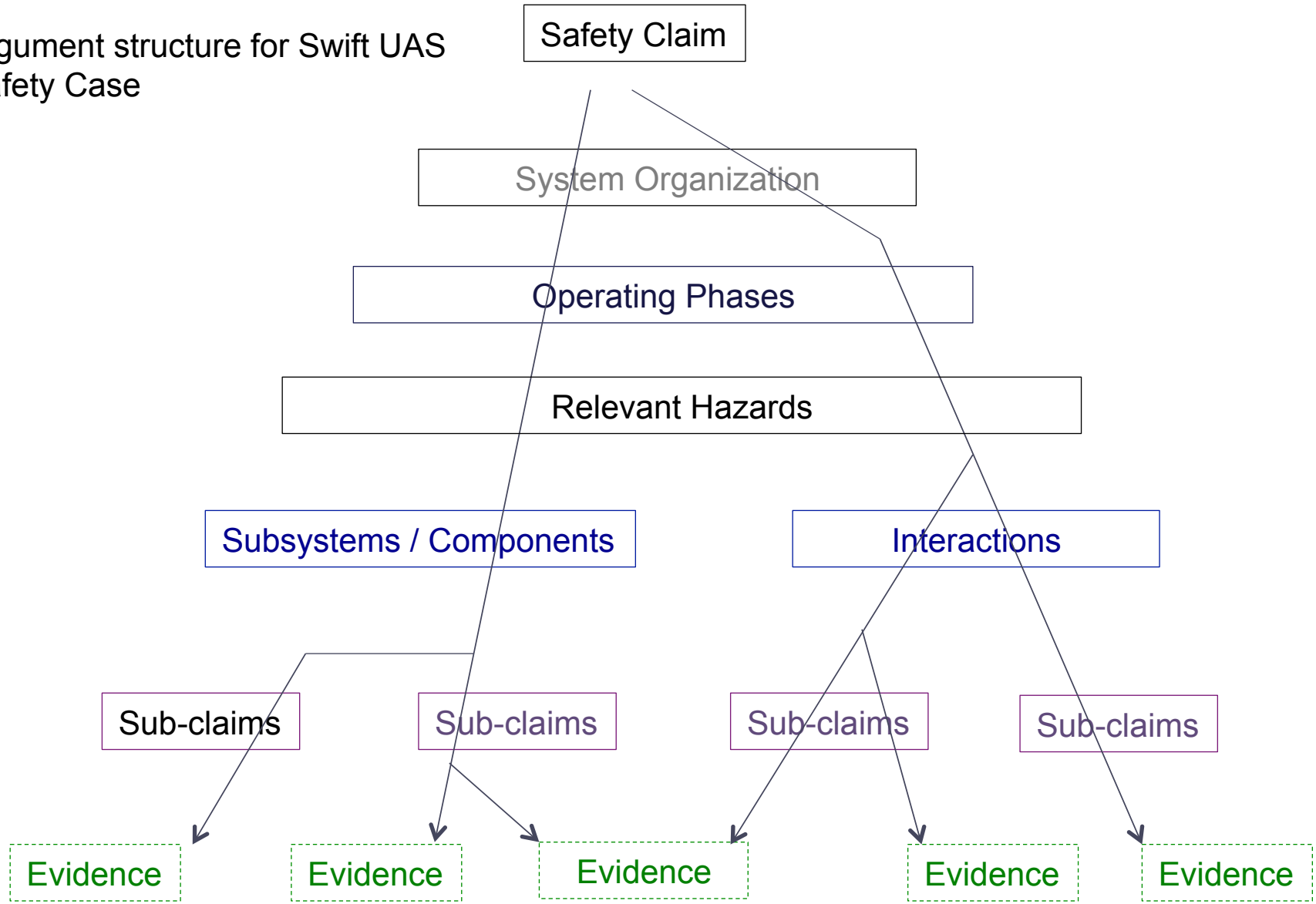
Argument Structures



Generic safety case structure for goals-based safety arguments

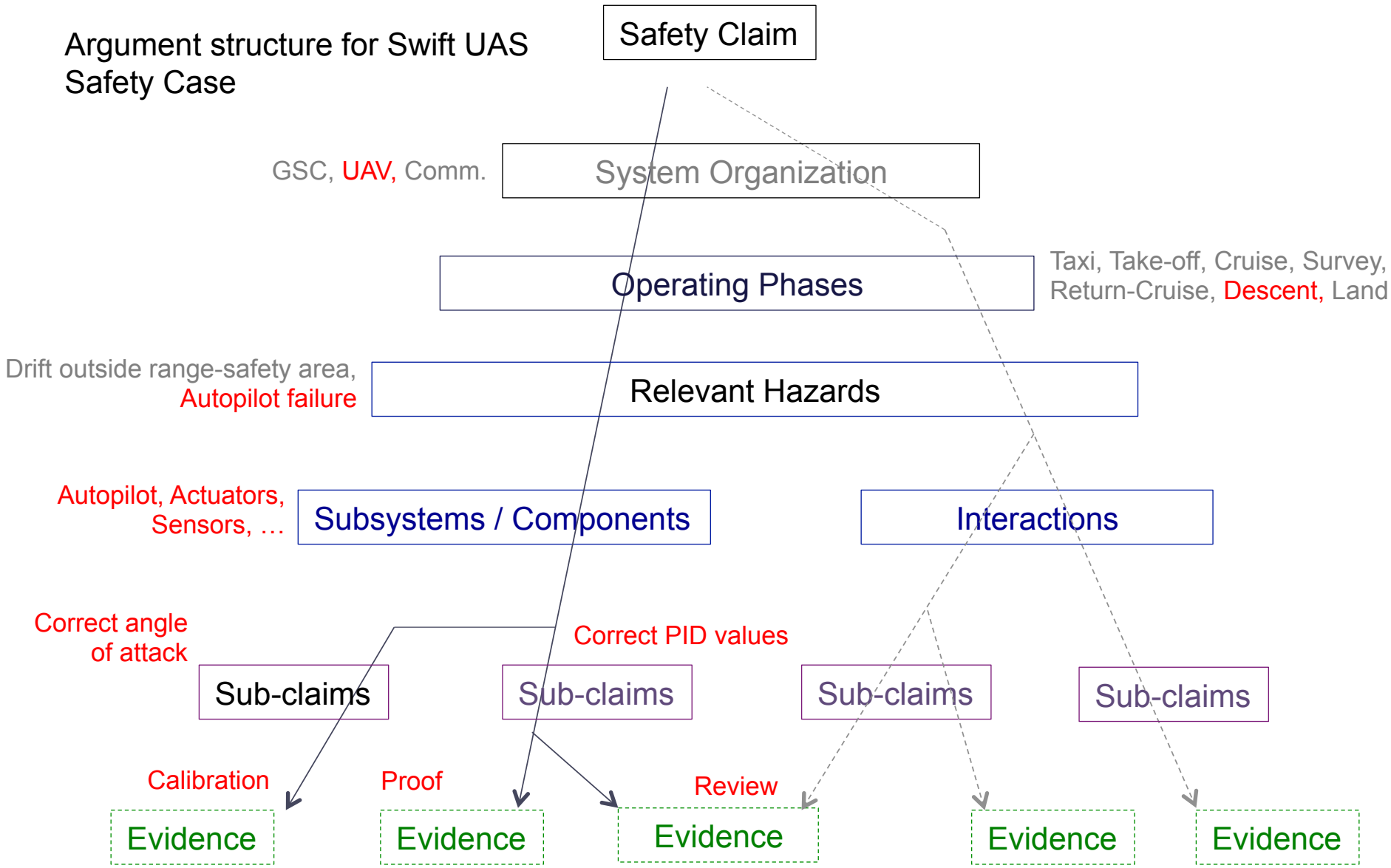
Argument Architecture

Argument structure for Swift UAS Safety Case

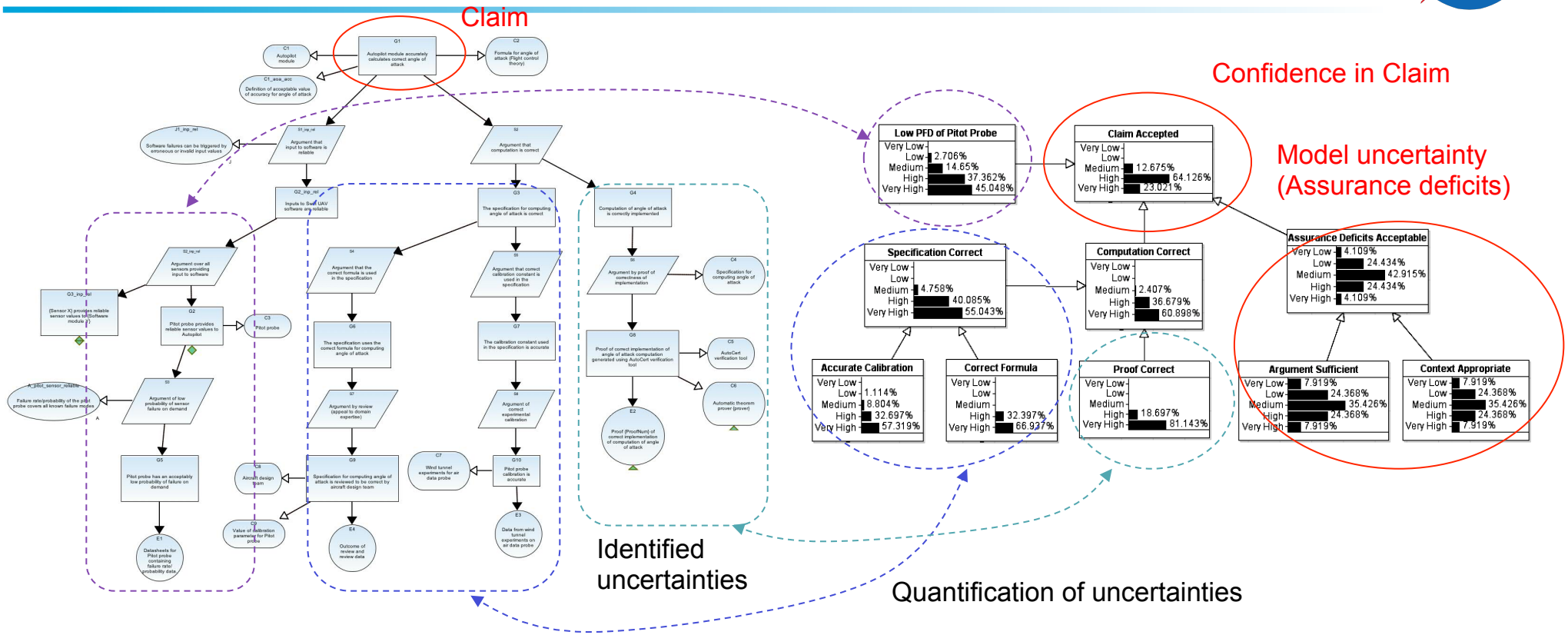


Argument Architecture

Argument structure for Swift UAS Safety Case



Confidence Assessment



Correct computation of Angle of Attack – Argument in GSN

- Uncertainty from *heterogeneity of evidence and reasoning*
  e.g., sensor value correct? Proof correct? Specification valid?
- Uncertainty from *assurance deficits*
  e.g., Evidence sufficient? Context/Justifications valid? Reasoning sound?

Confidence Assessment using Bayesian Networks

- Quantify identified sources of uncertainty
  - Aleatory: From data e.g., uncertainty in experiments
  - Epistemic: Subjective judgment
- Probabilistic modeling
  Confidence in the claim is the joint probability of the constituent uncertainties

Metrics

- Quantitative basis for evaluation
  - Internal measures of “quality” e.g.,
    - To what extent are claims developed – fully? partially?: **Claims coverage**
    - To what extent are high- / low-level safety requirements covered?: **Requirements coverage**
  - External measures of “quality” e.g.,
    - To what extent are hazards covered? – fully? partially?: **Hazard coverage**
  - Integrating confidence into a measure e.g.,
    - How well are the hazards covered?

- Quantitative basis for decision making → Supporting independent assessment
  - Tracking progress of an integrated systems development and safety process e.g.,
    - Coverage of hazards / claims / requirements at a specific milestone
    - Coverage for a specific sub-system / operational mode
  - Resource/Effort allocation e.g.,
    - Low coverage and/or Low confidence = Reallocate effort (contingent on cost-benefit analysis)

AdvoCATE: Assurance Case Automation Toolset

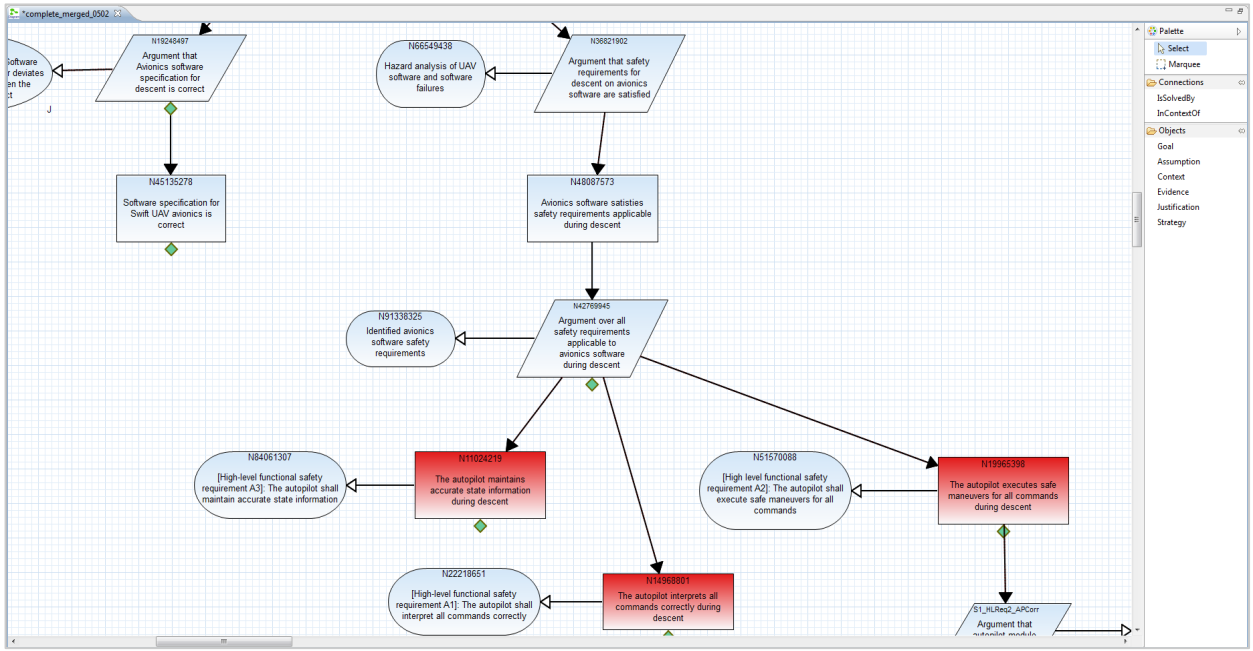
- Platform for research on rigorous safety cases
  - Model-based transformation (both creation and manipulation)
  - Complies with GSN and ARM
  - Integration with formal methods
  - Integration with requirements based processes
  - Generation of useful artifacts
  - Integration with other widely used tools

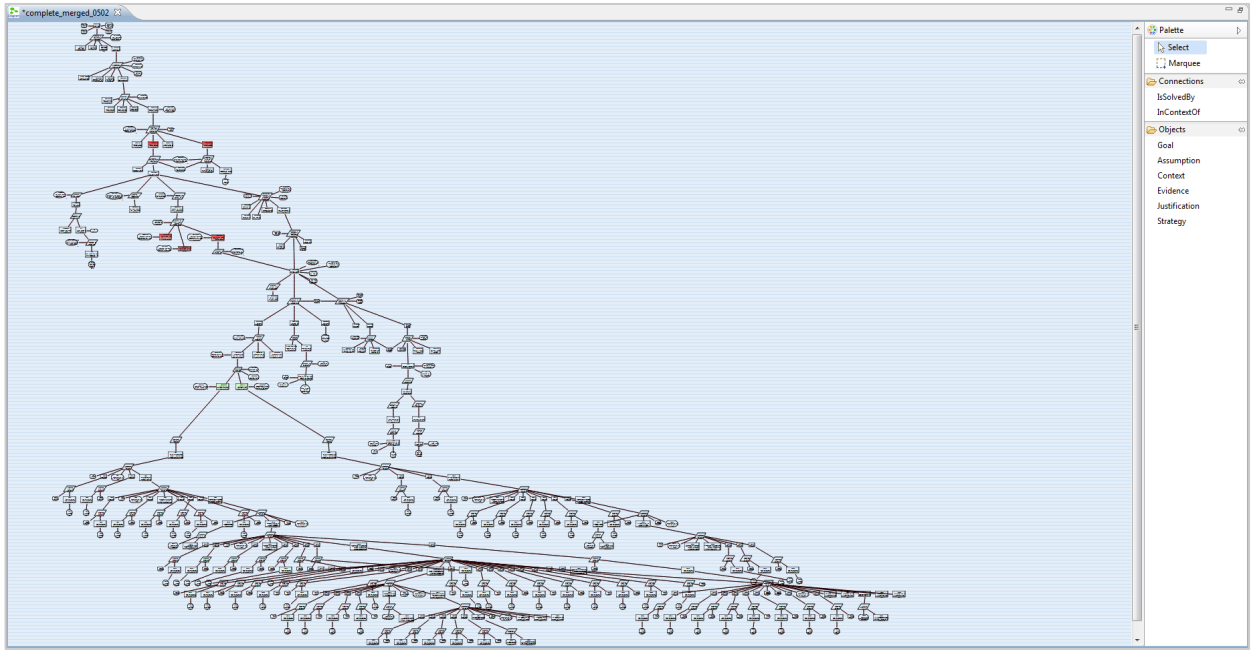
- *Current emphasis: metrics, formal methods, requirements-process integration*

AdvoCATE: Assurance Case Automation Toolset

- Creation of safety case diagram
  - Direct editing of nodes
  - Import from D-Case, ASCE, CertWare
  - Import from formal methods tools
  - Hyperlinks in nodes to allow cross references to documents, data for evidence, context, etc.
  - Metadata on nodes: hazards, high/low requirements, risk (severity, likelihood), provenance (automation, etc.)
- Transformations
  - To tabular form (CSV)
  - To textual form (hierarchical report)
  - Generation of to-do lists
  - Computation of metrics
  - Generation of traceability matrices
- Usability
  - Show / Hide nodes
  - Focusing and centering on nodes

Under Development at NASA Ames





QUESTIONS?