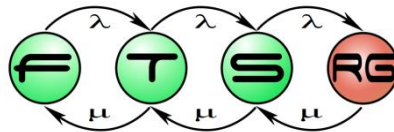


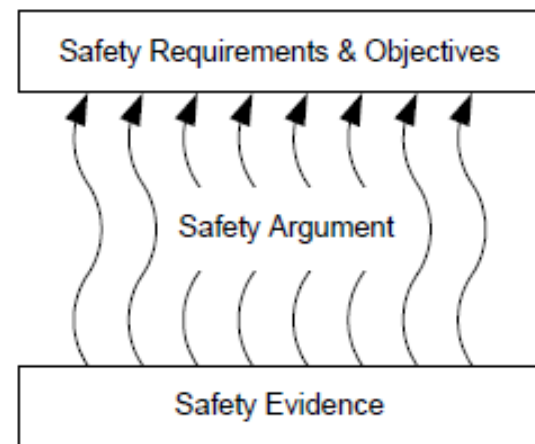
The Safety Case

Structure of Safety Cases
Safety Argument Notation



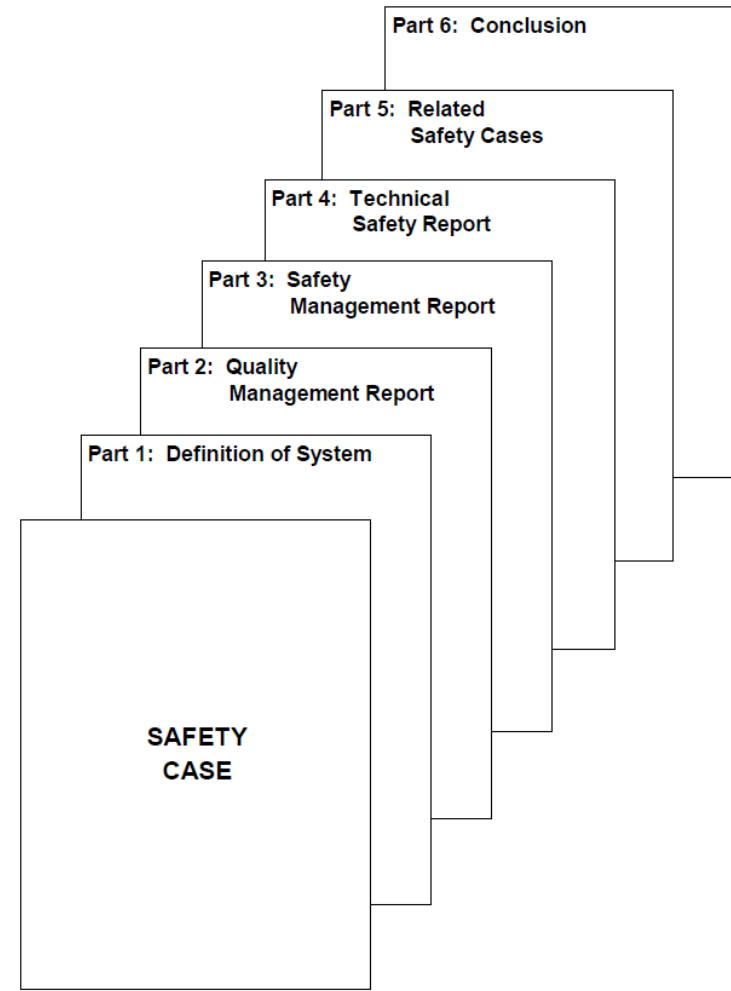
The safety case

- Definition (core): The documented demonstration that the product complies with the **safety requirements**
- Role:
 - A safety case should **communicate** a comprehensive and defensible **argument** that a system is acceptably safe to operate in a particular context
 - Condition for safety **acceptance** and approval
- To be prepared by: Developers and/or operators
- To be accepted by: Safety authority and/or customer
- Principal elements:
 - Safety **requirements** (goals, objectives)
 - **Arguments** (relations)
 - **Evidences**
 - Analysis results (e.g., FTA, FMEA)
 - Formal verification
 - Test results
 - ...



Standard structure of a safety case

- Conditions for safety acceptance
 - Evidence of **quality** management
 - Evidence of **safety** management
 - Evidence of **technical** safety
- Structured presentation of evidence and arguments
- Example: EN50129 (railway)
 - Part 1: Definition of the system
 - Part 2: Quality management report
 - Part 3: Safety management report
 - Part 4: **Technical safety report**
 - Part 5: Related safety cases
 - Part 6: Conclusion



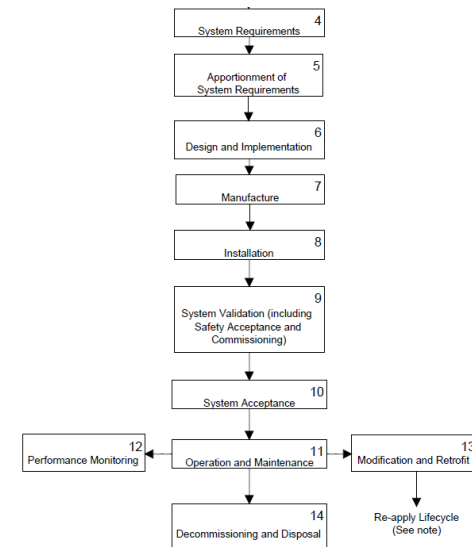
Quality related parts of the safety case

Part 2: Quality management report

- Minimize the incidence of human errors at each stages in the lifecycle:
Reduce the risk of systematic faults

Part 3: Safety management report

1. Safety **lifecycle**: From requirements to validation
2. Safety **organization**: Roles and competence
3. Safety **plan**: Activities and approval milestones + review
4. **Hazard log**: Hazards + risks + risk control
5. Safety requirements
6. System design
7. Safety reviews
8. Safety verification and validation
9. Safety justification
10. System handover (to authority)
11. Operation and maintenance
12. Decommission and disposal



Technical parts of the safety case

Part 4: Technical safety report

1. Introduction:
 - Summary of technical principles and standards
2. Assurance of **correct functional operation**
 - Architecture, interfaces, fulfillment of requirements, assurance of correct hardware and software behavior
3. Effects of **faults**
 - Random hardware faults: Quantified safety target
 - Detection, actions after detection, effects, independence, multiple faults
 - Systematic faults: Risk reduction
4. Operation with **external influences**
 - Demonstration of operability and safety
5. Safety-related **application conditions**
 - Rules, conditions, constraints
6. Safety **qualification tests**
 - Evidence to demonstrate completion

Safety argumentation

Communicating safety arguments

- Typical: Free text
 - Structured form (items, enumerations, references)
 - Complex arguments are difficult to describe
 - Review, management, tracking, coordination is difficult

```
For hazards associated with warnings,
the assumptions of [7] Section 3.4
associated with the requirement to
present a warning when no equipment
failure has occurred are carried
forward. In particular, with respect to
hazard 17 in section 5.7 [4] that for
test operation, operating limits will
need to be introduced to protect against
the hazard, whilst further data is
gathered to determine the extent of the
problem.
```

?

- Graphical notation: Goal Structuring Notation
 - Elements of safety arguments
 - Relationships between the elements

Elements

- **Goal:** Objective, claim about the system
 - Compliance with requirements
 - Sufficient mitigation / avoidance of hazards
 - Without evidence it is unfounded!
- **Strategy:** Decomposition method
 - Derivation of sub-goals
- **Evidence** (solution)
 - Results of observation, analysis, test, simulation, ...
 - Fundamental information from which safety can be inferred
- **Context**
 - Context of demonstrating safety
- **Assumption** or **Justification**
 - Limits, conditions etc.
- **Undeveloped goal**
 - Further development is necessary

System can tolerate single component failures

Argument by elimination of all hazards

Fault Tree for Hazard H1

All Identified System Hazards

A1
Sub-systems are independent



Relations

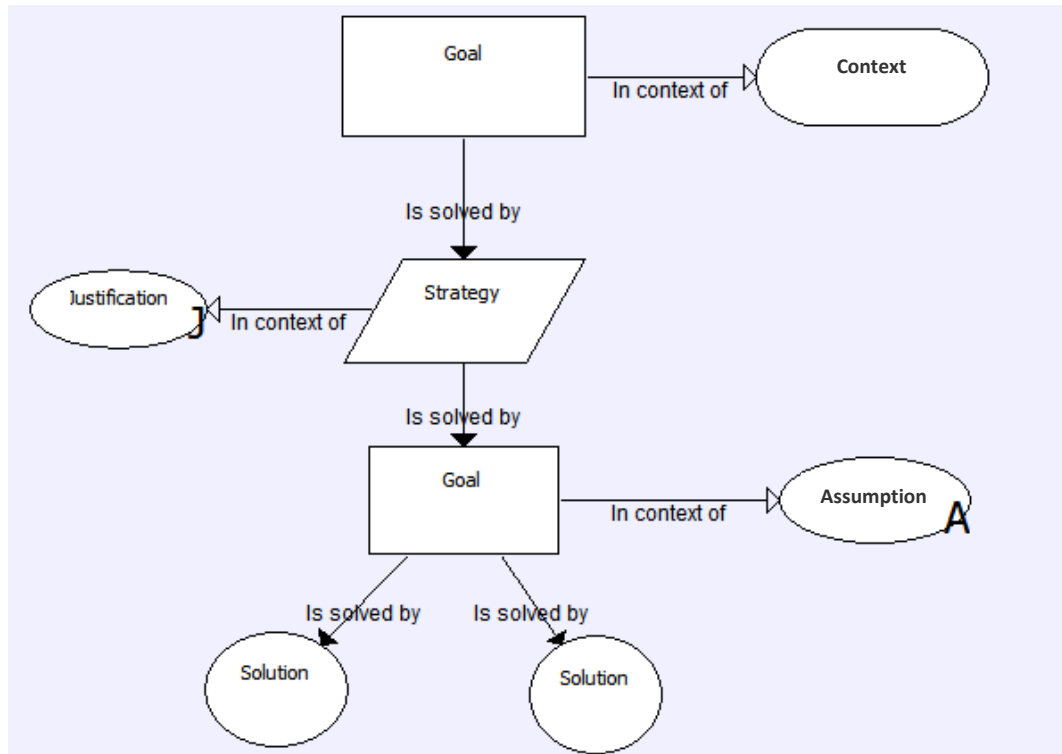
- “Is solved by”
 - Applied between goals, strategies, evidences
- “In context of”
 - Applied between contexts / assumptions / justifications and other elements



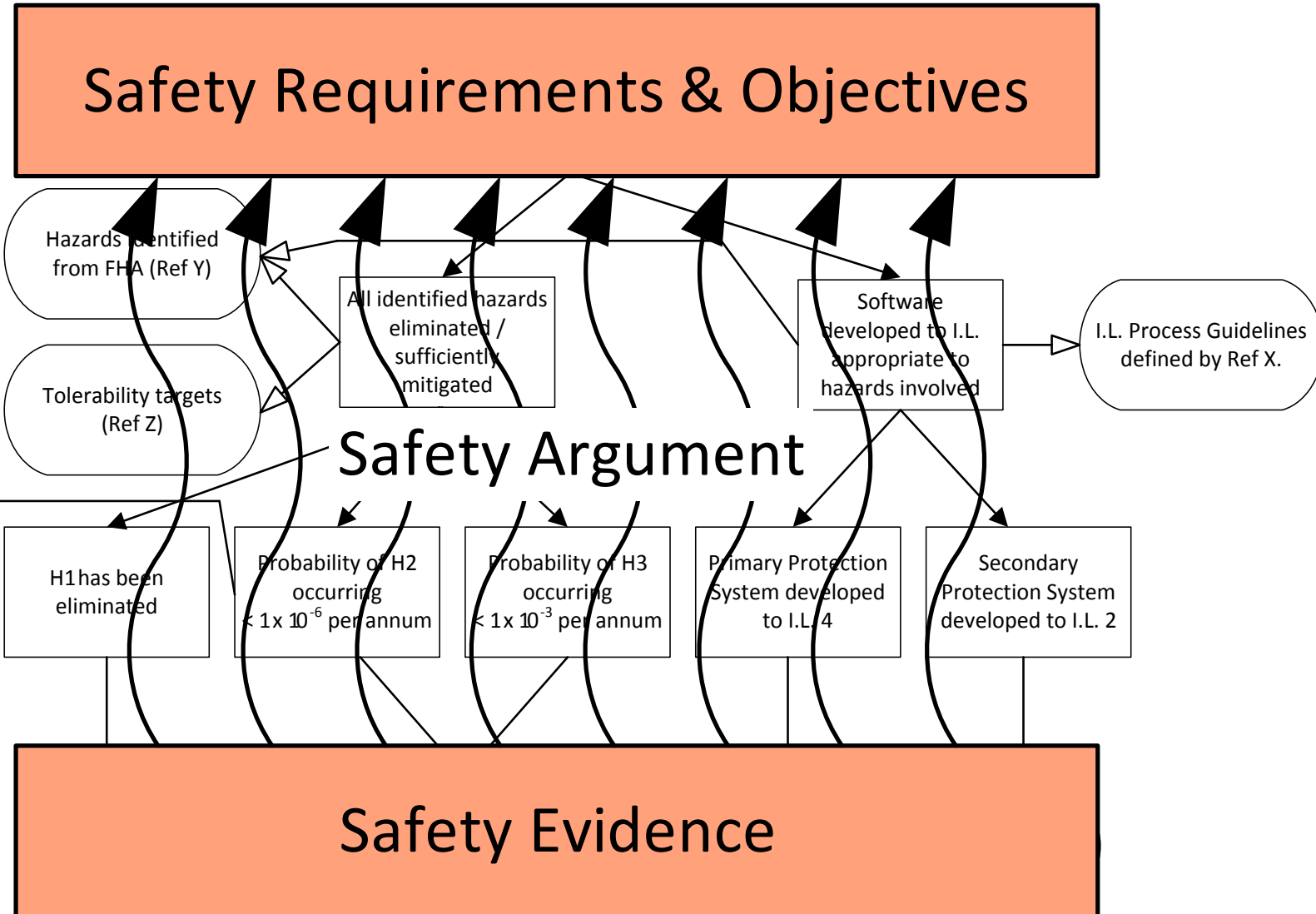
SolvedBy



InContextOf

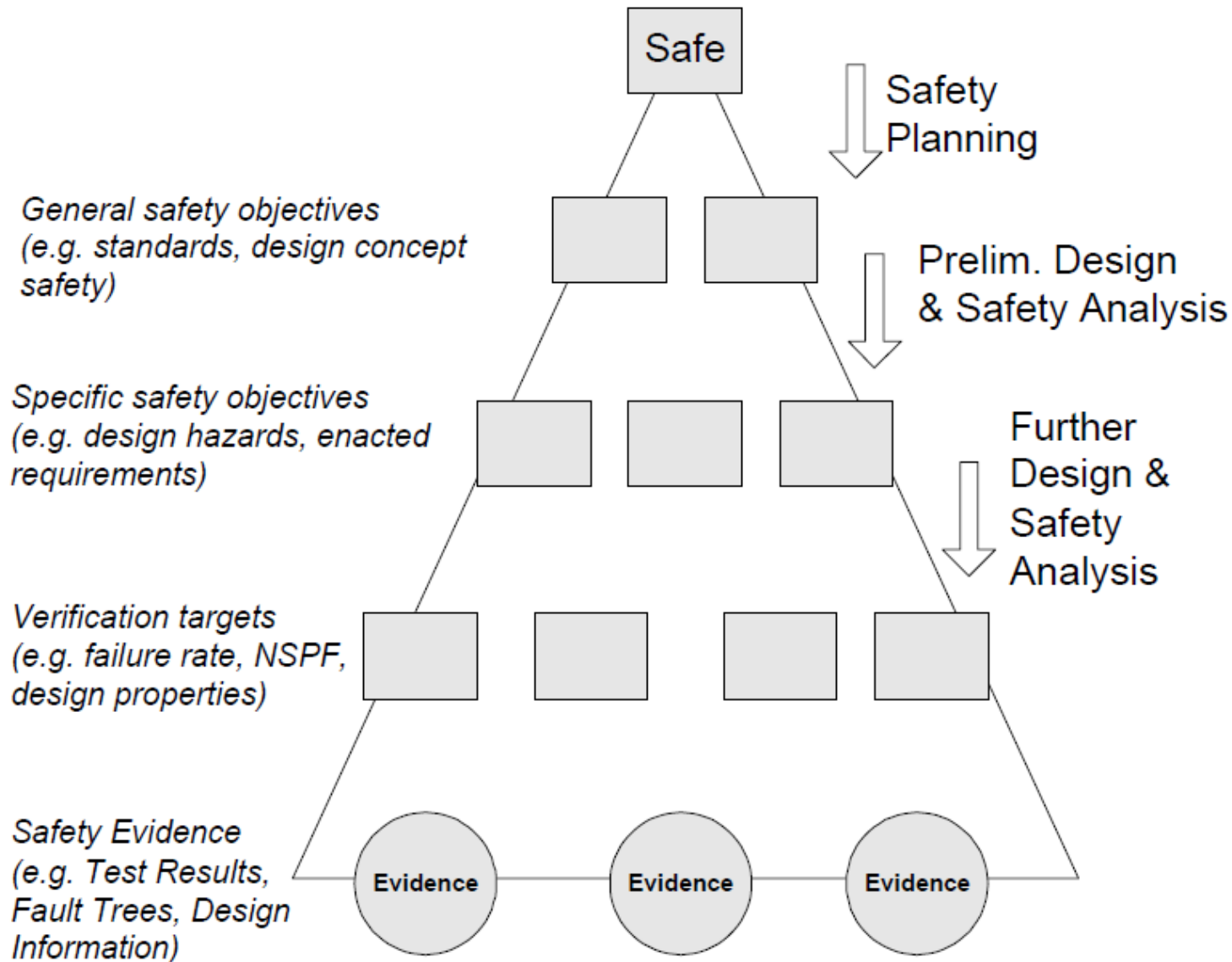


Overview of safety argumentation

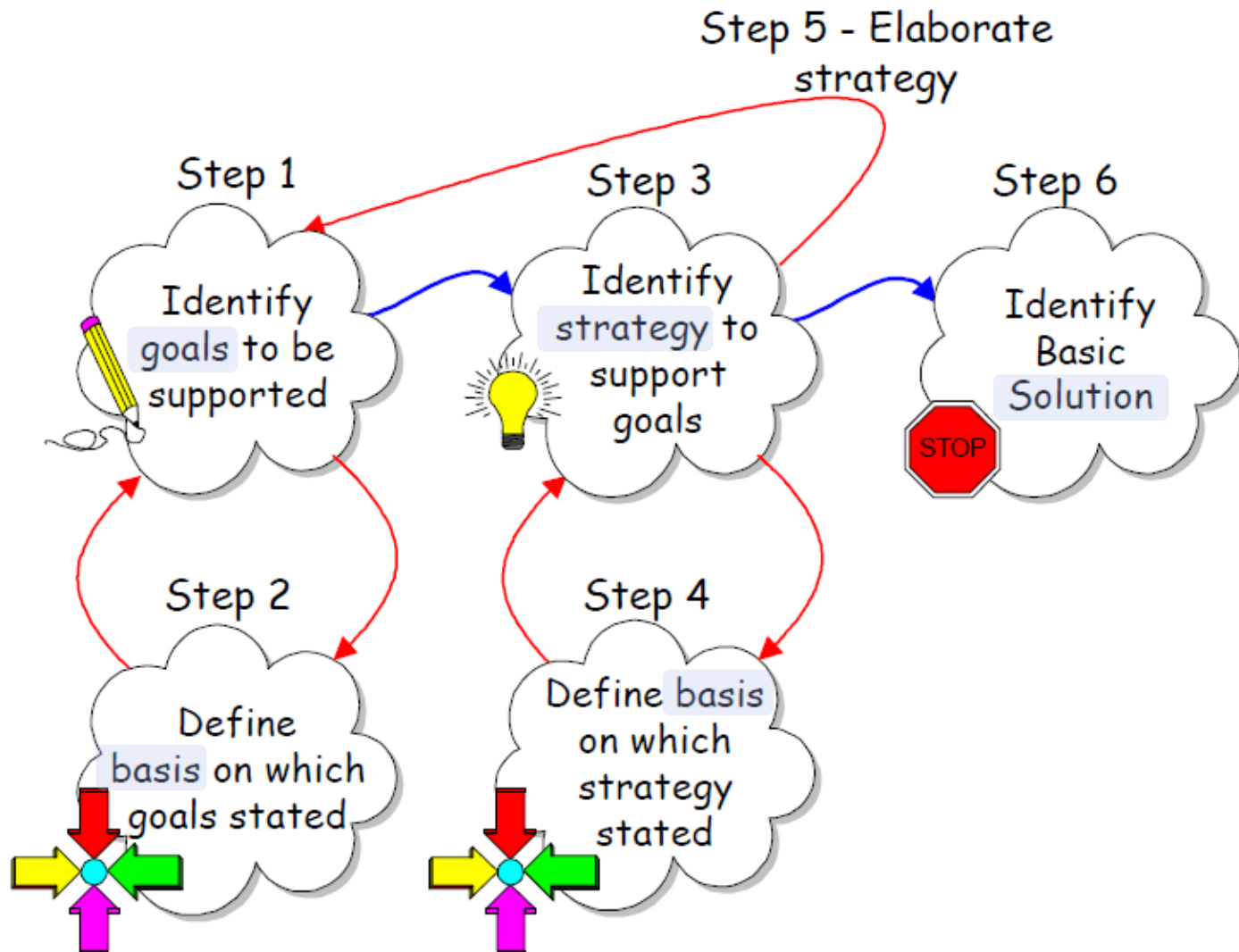


Source: T. Kelly

Evolution of the goal structure

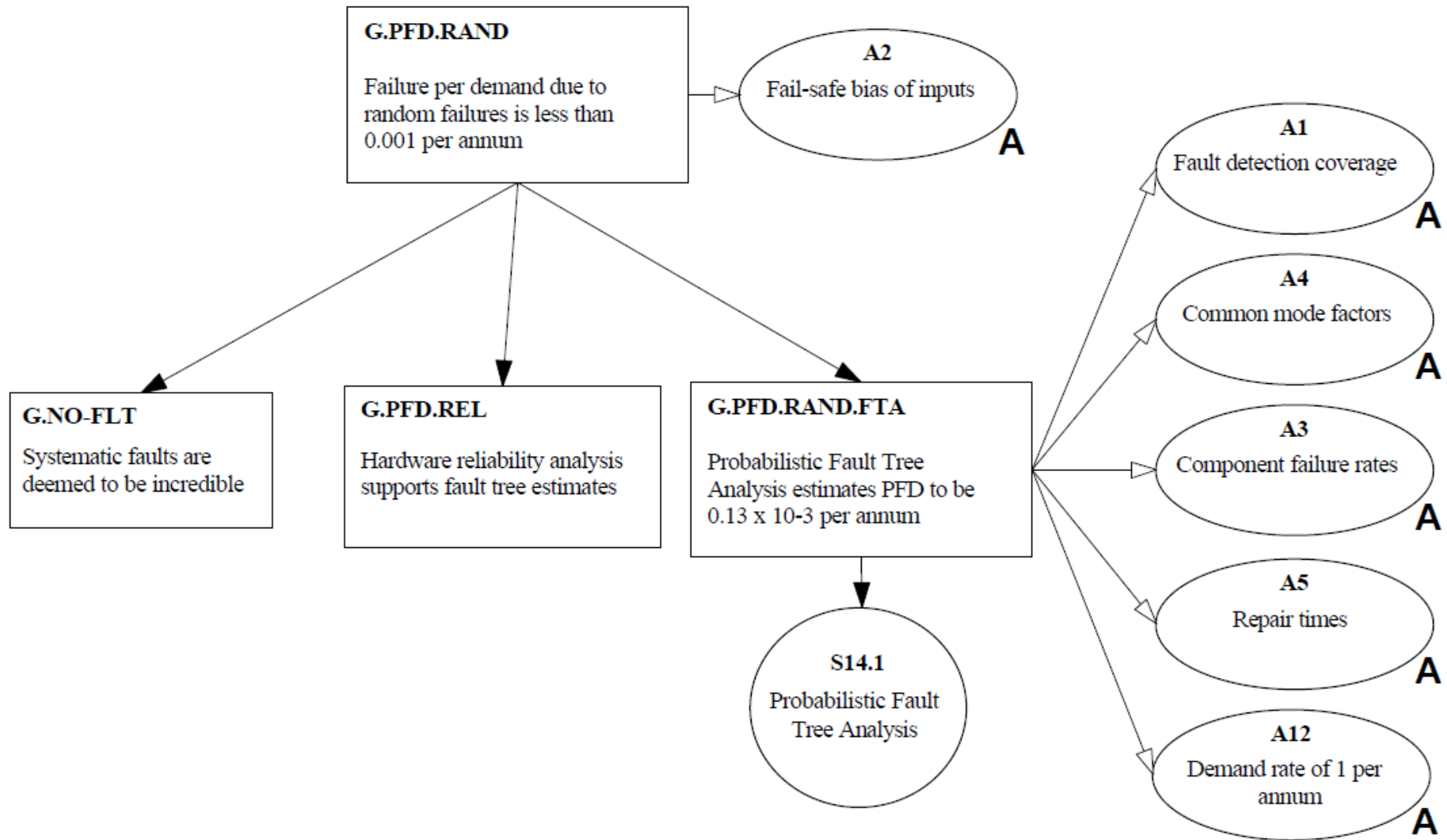


Steps of safety case construction



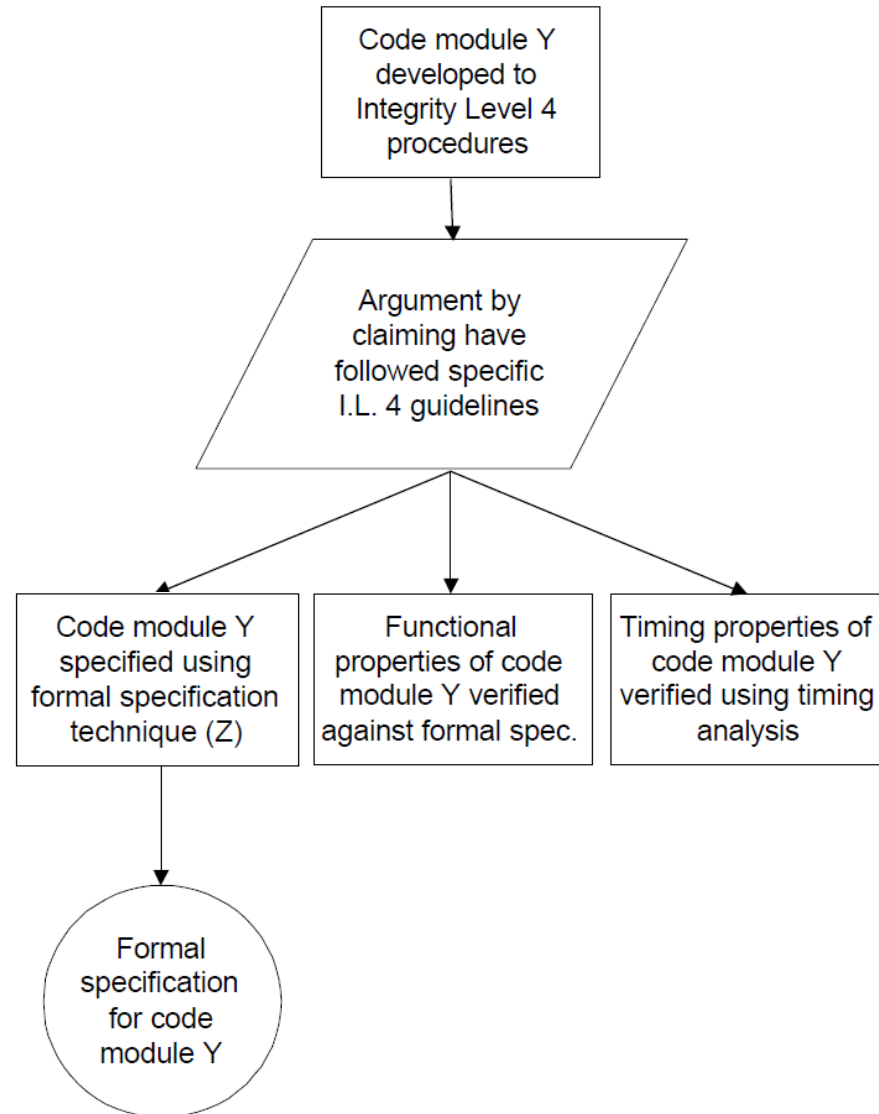
Source: T. Kelly

Safety arguments for hardware

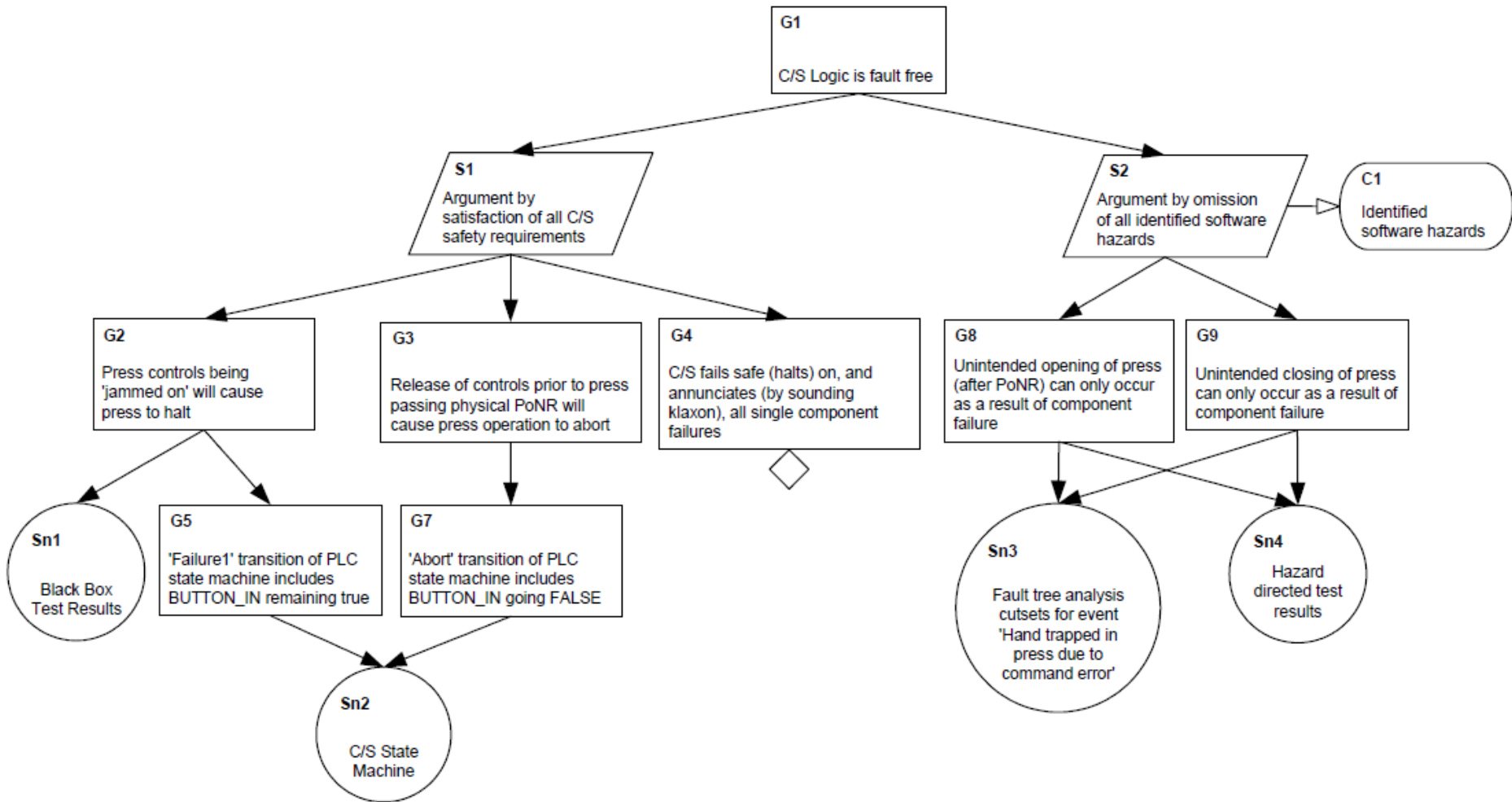


Safety arguments for software

- Software SIL:
Required **techniques**
and **measures**
form arguments and
evidences
- Example: Guidelines
followed for SIL4
 - Formal specification
 - Formal verification of
functionality
 - Formal verification of
timing

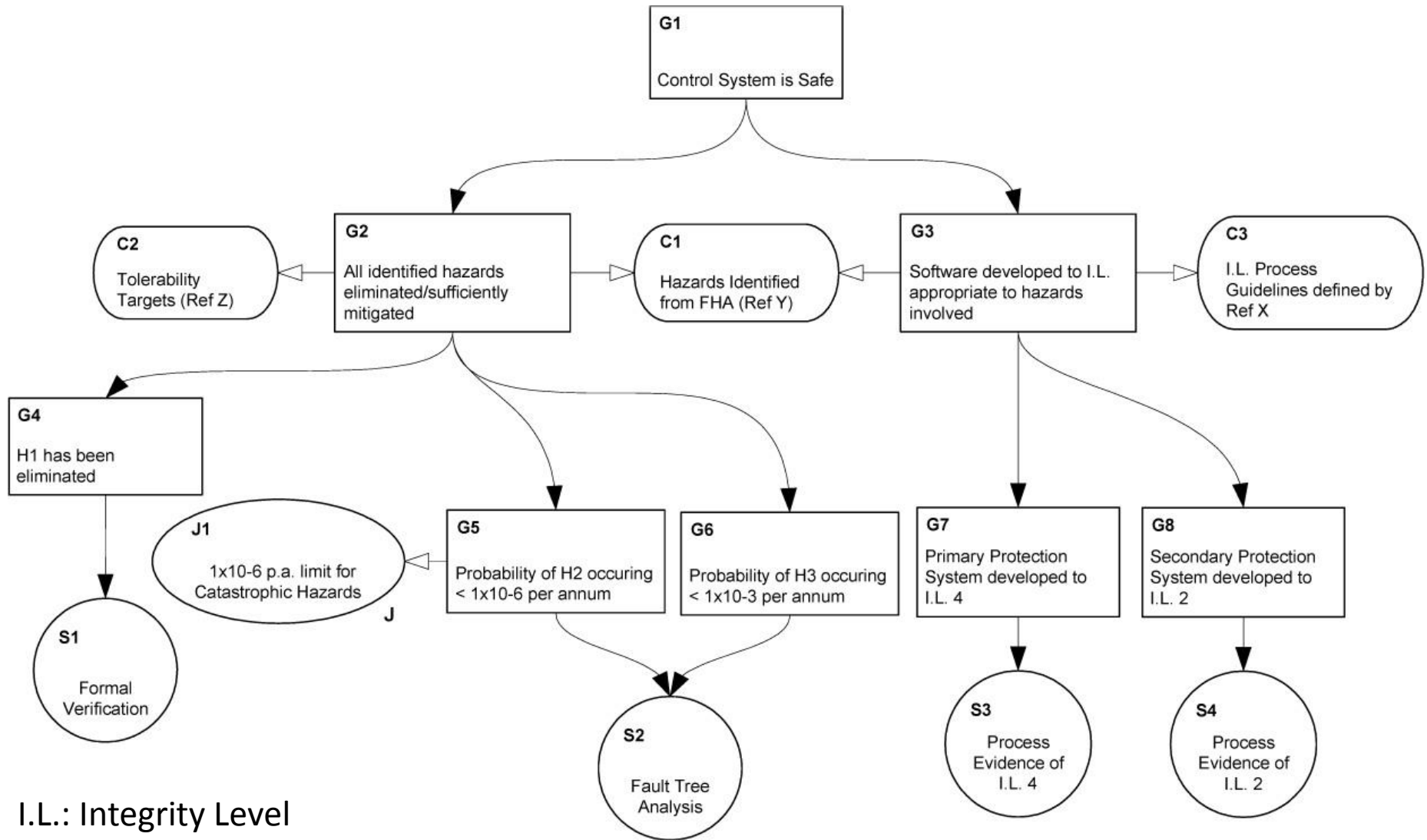


An example goal structure



Evidences: Test results, state machine analysis, fault tree analysis, directed testing

Generic goal structure



I.L.: Integrity Level

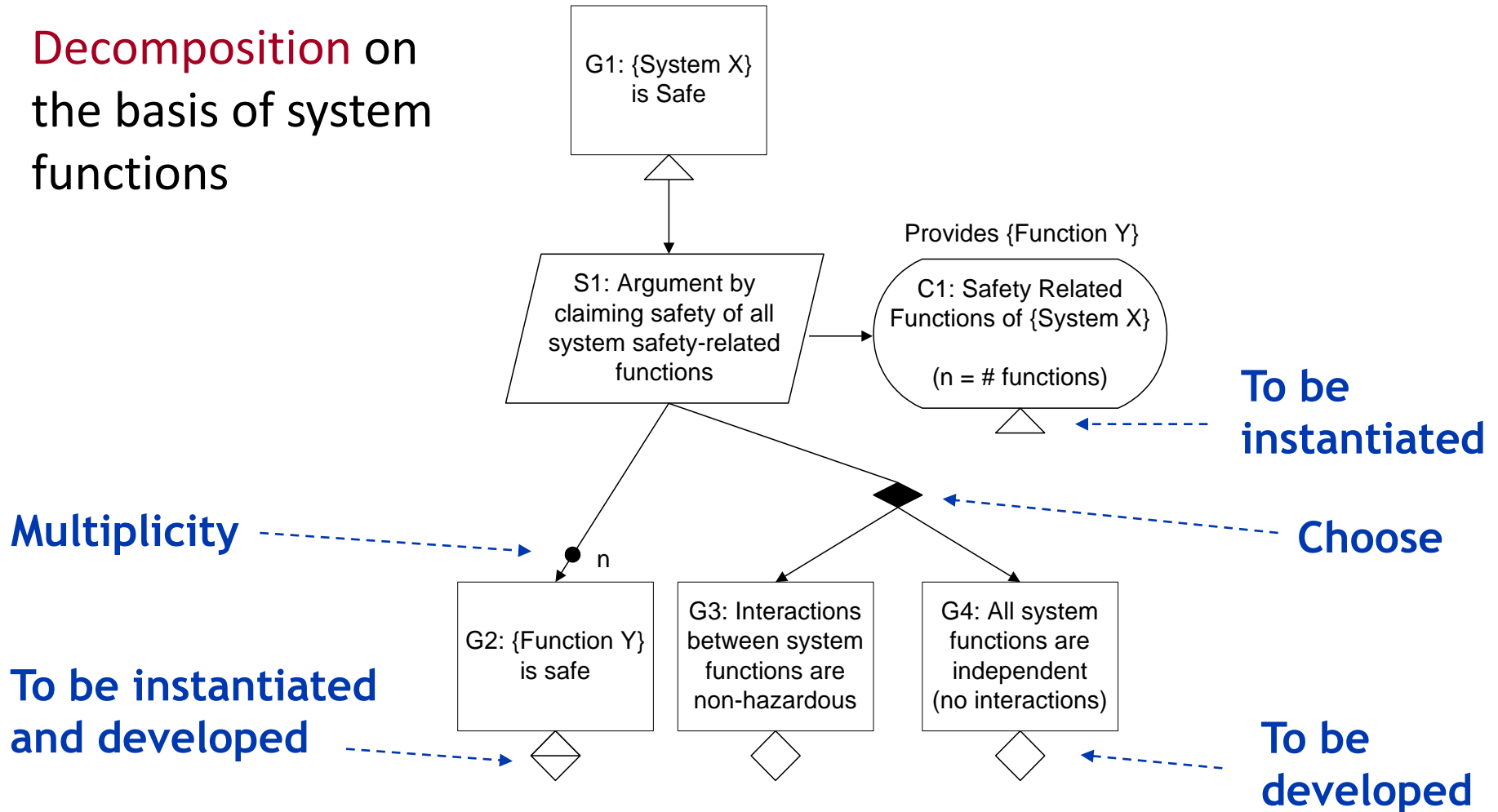
FHA: Functional Hazard Assessment

Safety case patterns

- Combines **argumentation** and **patterning**
 - Supports the **re-use** of successful argument approaches (best practice)
 - Focus on **semantics** rather than the syntax of the safety case
- GSN **extensions** to support capturing patterns
 - Multiplicity
 - Instantiation
 - Develop
 - Instantiation and develop
 - Choice

Example of a GSN pattern

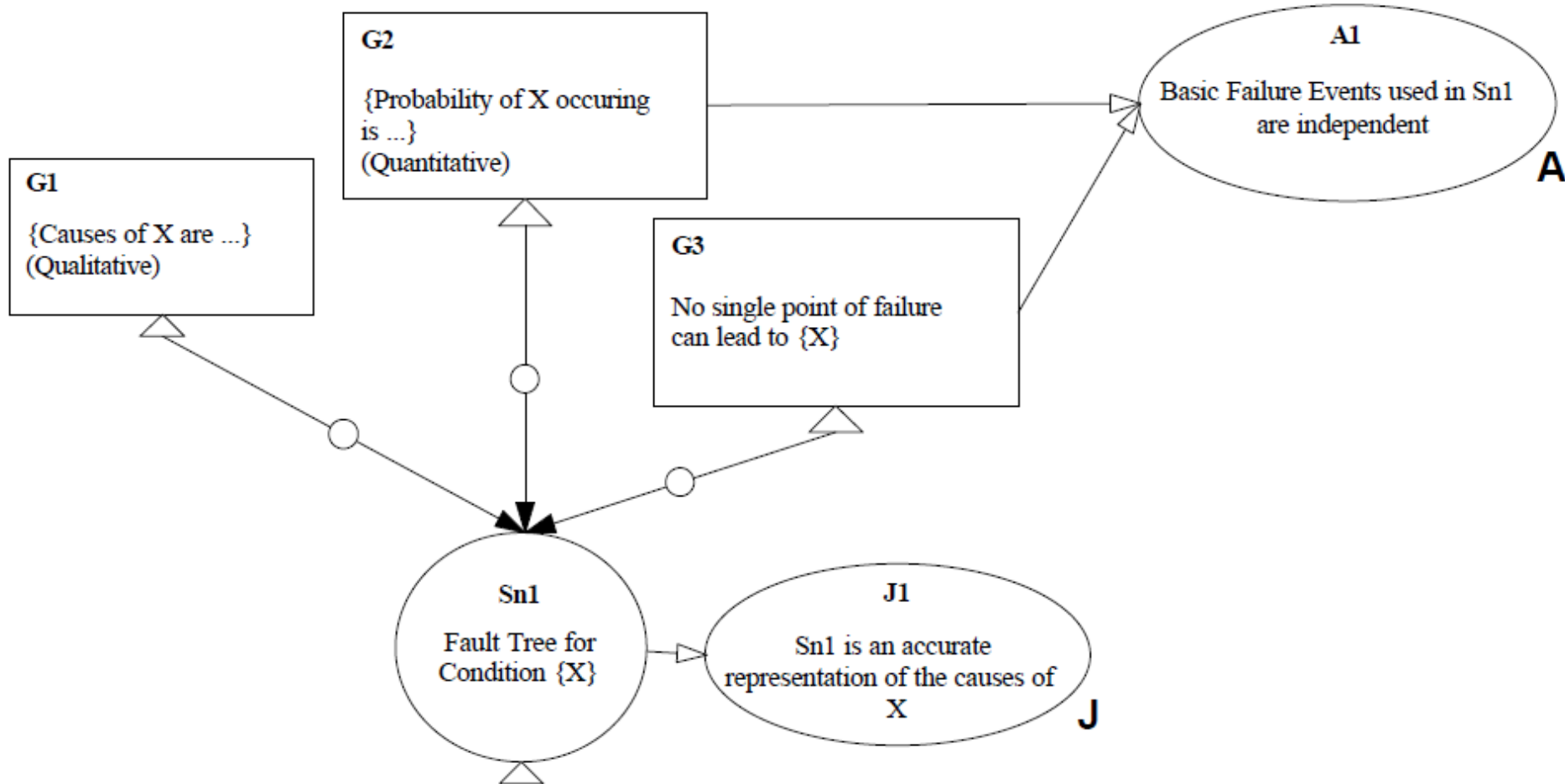
Decomposition on the basis of system functions



The Fault Tree pattern

How a fault tree analysis can be used as evidence

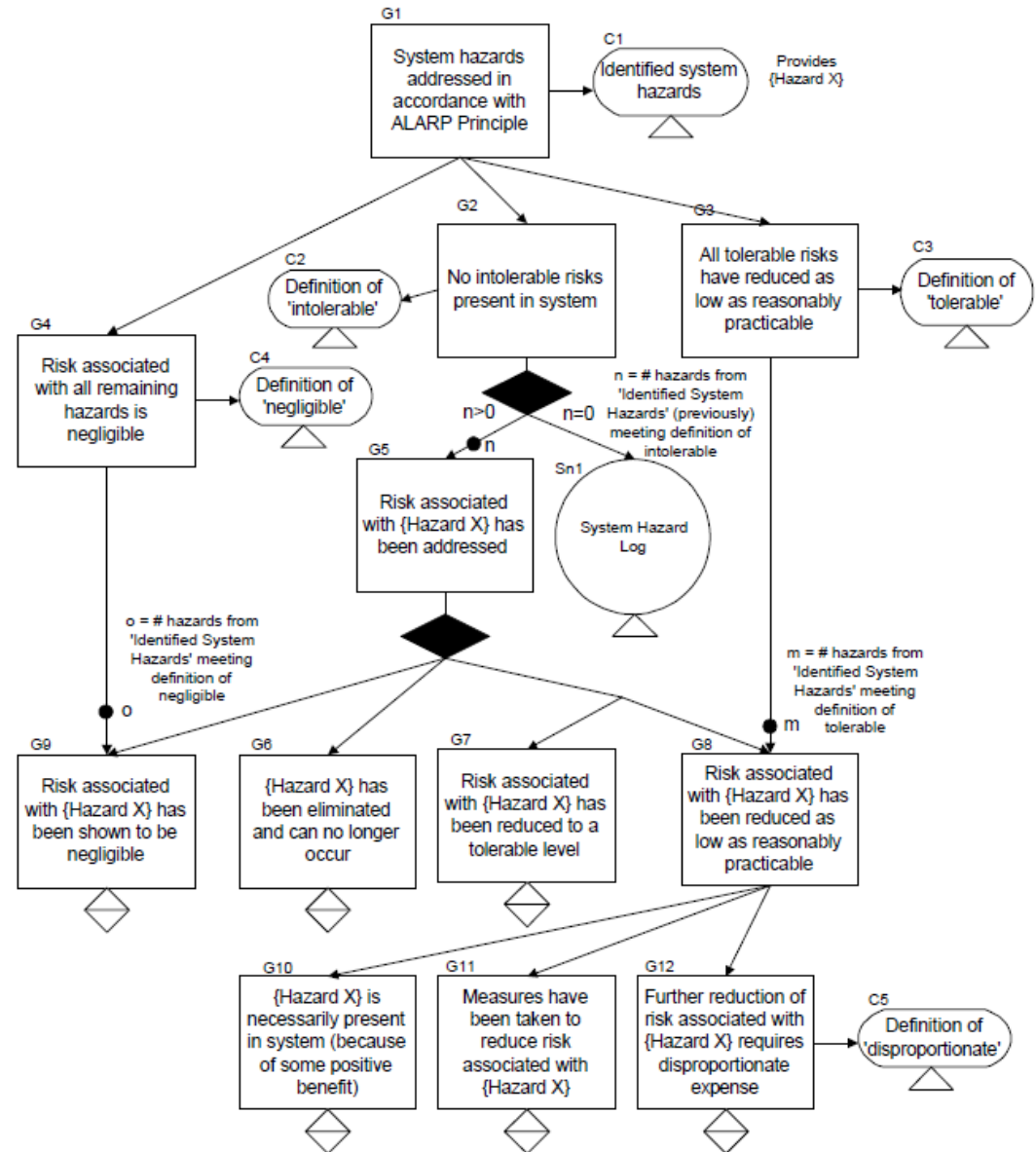
Fault Tree Evidence



The ALARP pattern

ALARP: As Low As Reasonably Practicable

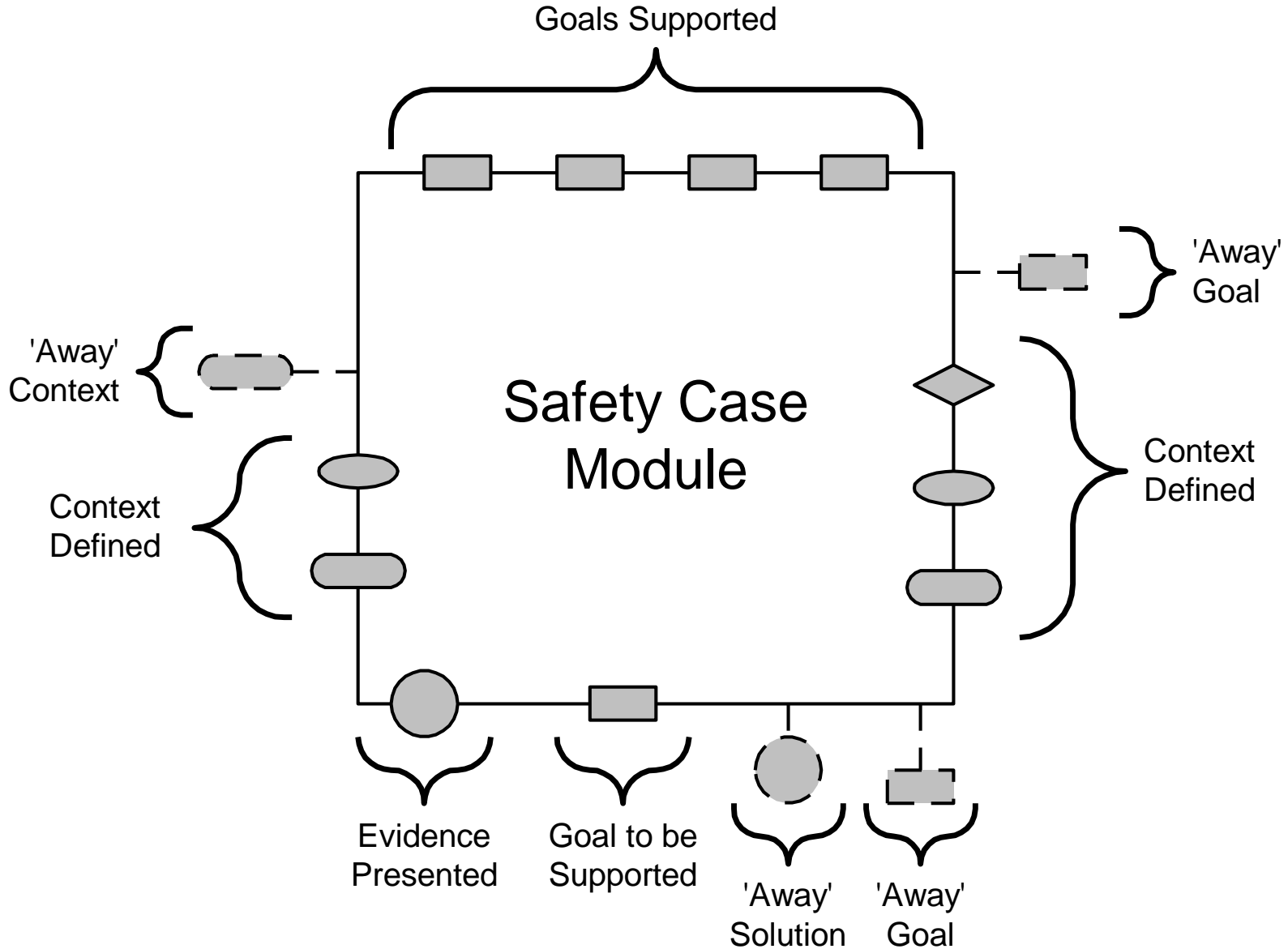
- No intolerable risk
- All tolerable risks have reduced as low as reasonably practicable
- All remaining hazards have negligible risks



Modular safety cases

- Goal: **Modular**, compositional construction of safety cases (corresponding to system structure)
- **Partitioning** of modules
 - **Vertical** (hierarchical) partitioning
 - Claims of one argument are **objectives** of another
 - E.g., case split of system and software safety case
 - **Horizontal** partitioning
 - One argument providing the **assumed context** of another
 - E.g., “All system hazards have been identified” provides assumed context of an argument that “All identified system hazards have been sufficiently mitigated”
- **Module interfaces**
 - Dependency of objectives, evidence, context of other modules

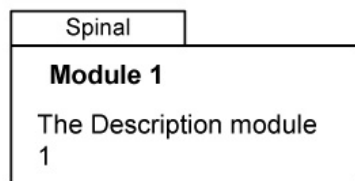
Principle of safety case interface



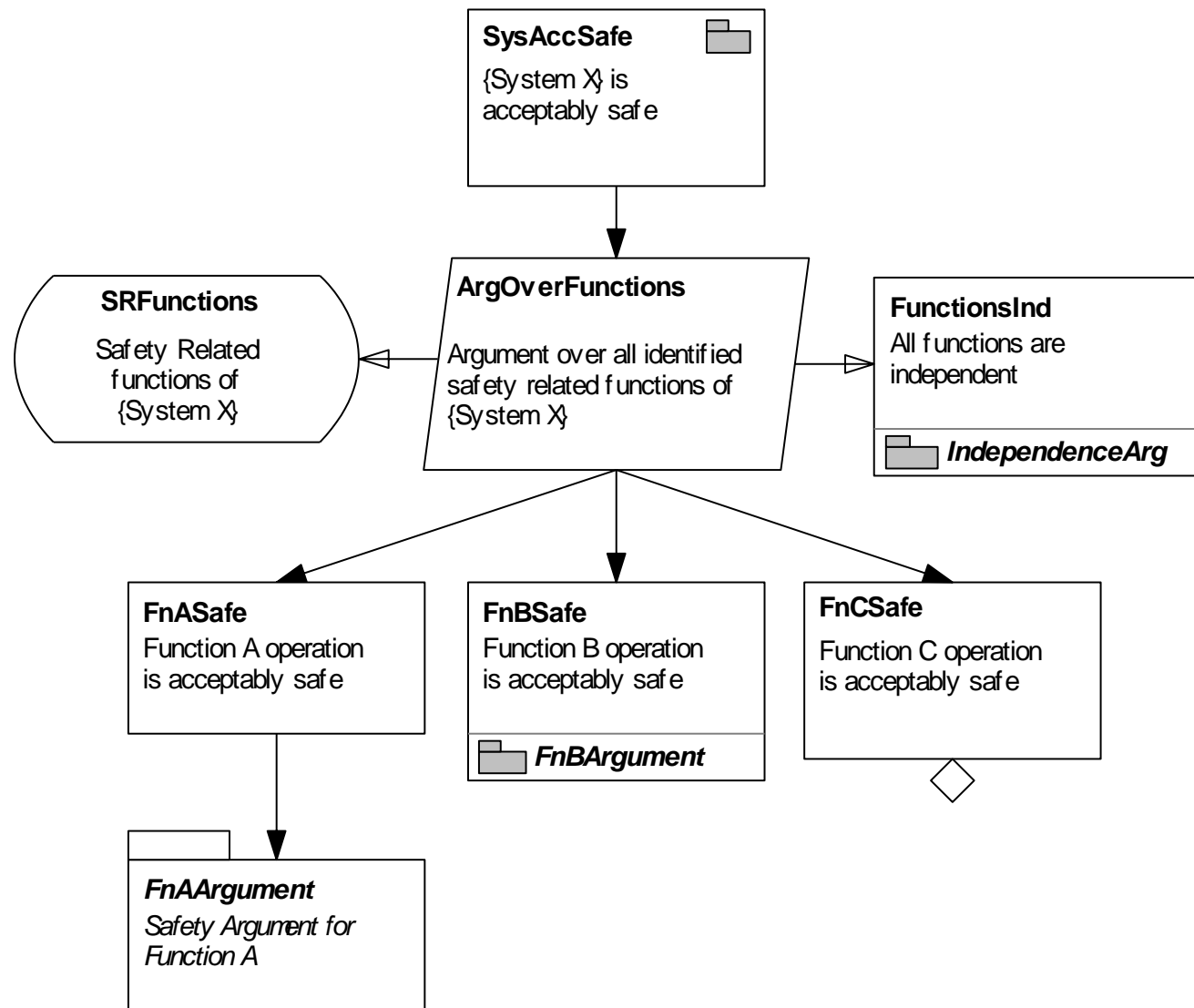
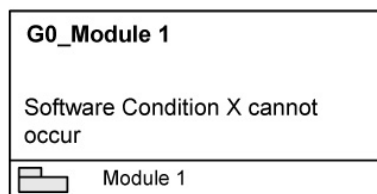
Example of a modular safety case

Elements:

- Safety case modules



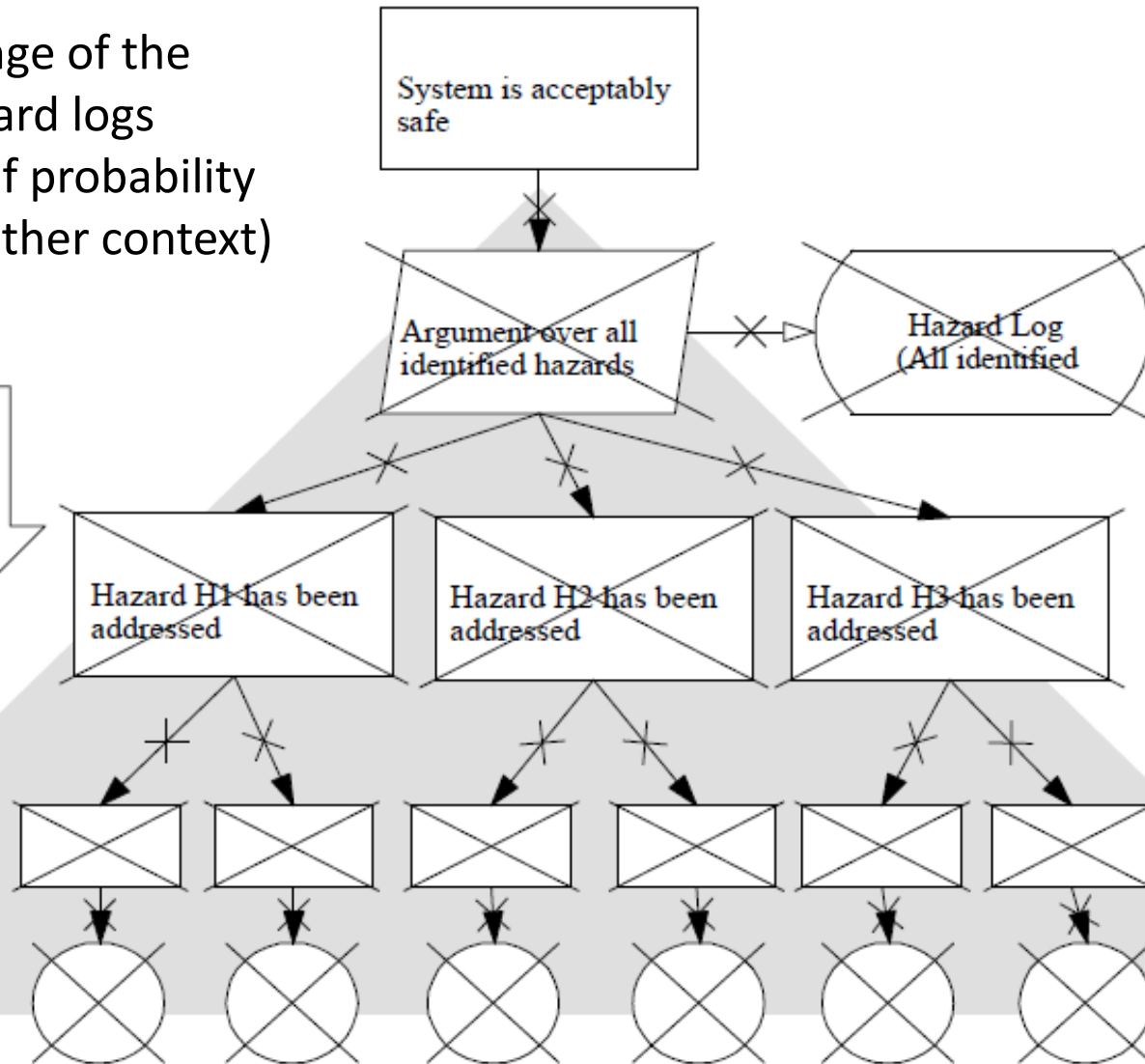
- "Away" goals



Management of safety cases

Example: Change of the context of hazard logs (e.g., change of probability of hazards in other context)

Inherited Change Effect



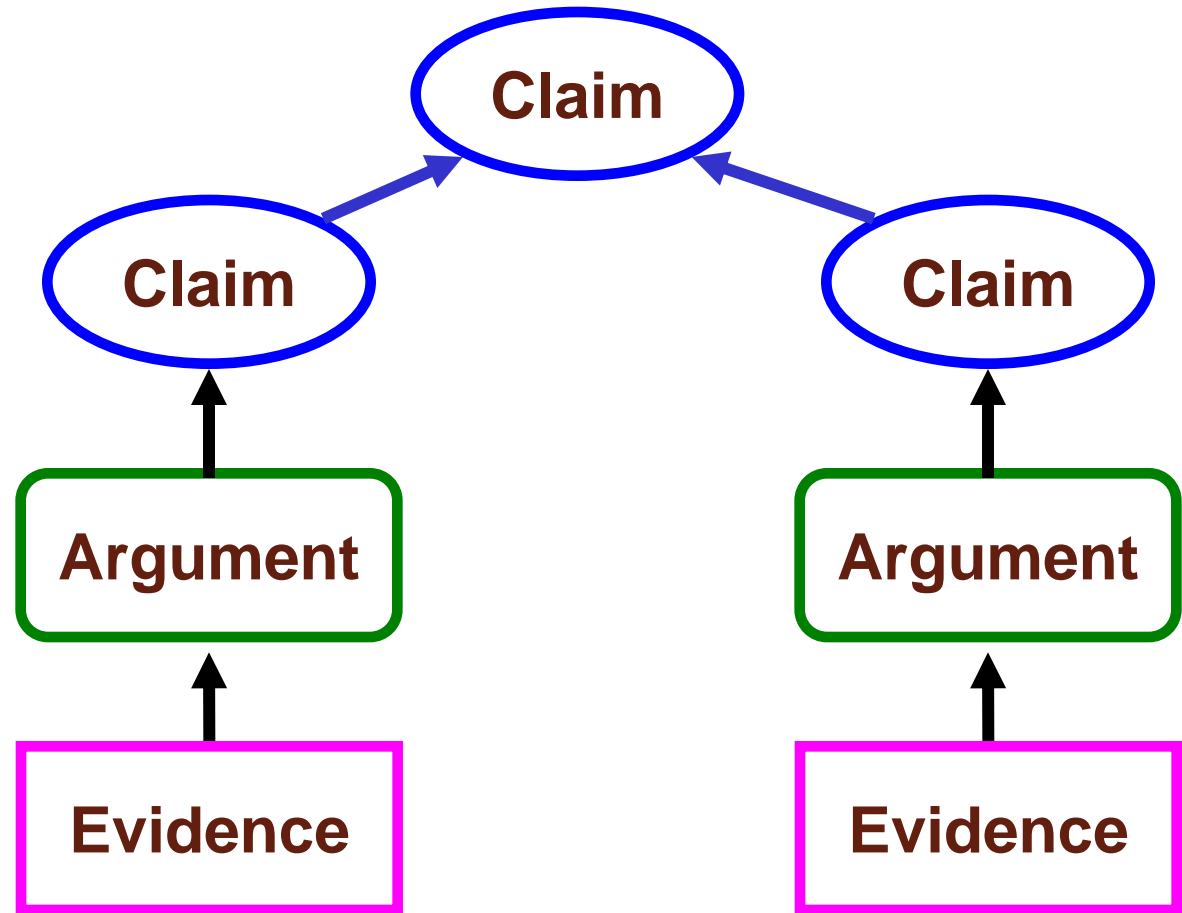
Advantages and disadvantages of GSN

- Advantages:
 - Simple elements
 - Captures the elements most important to safety arguments
 - Structured hierarchical breakdown
 - Method guidance exists
 - Semantics well defined and understood (first order logic)
 - Can be used at various stages of argument development
 - Increasingly being adopted by companies
- Disadvantages:
 - Easy to read, harder to write 😊
 - Doesn't stop you writing bad arguments ☹️

Other approaches

■ ASCAD: Adelard Safety Claims Arguments Data

- **Claim:**
Assertion to be proven
- **Argument:**
How evidence supports claim
- **Evidence:**
Required observation, analysis, test, ...

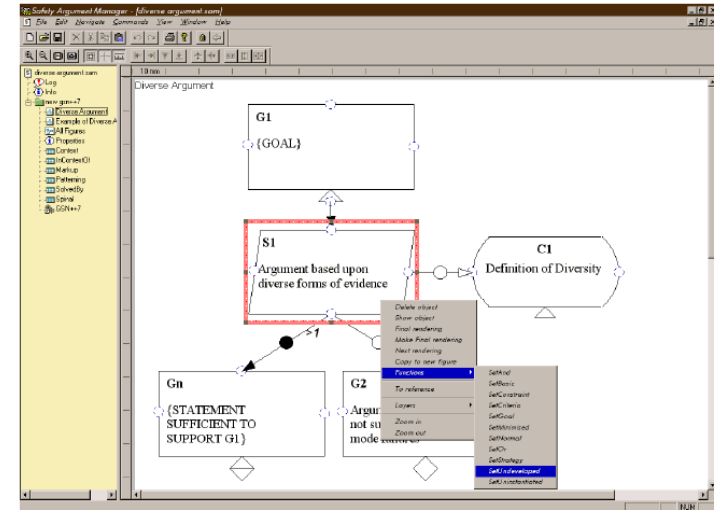


Generalization

- Assurance cases
 - Safety cases
 - Security cases
 - Dependability cases
- Definition
 - A documented **body of evidence** that provides a convincing and valid **argument** that a specified set of **critical claims** regarding a system's properties are adequately justified for a given application in a given environment
- Examples of using assurance cases
 - Security-critical applications: Based on Common Criteria
 - Medical devices: Based on ISO 14971

Supporting tools

- Adelard Safety Case Editor (ASCE)
 - Adelard, www.adelard.co.uk
 - Supports both GSN and ASCAD
- E-Safety Case
 - Praxis HIS, www.esafetycase.com
- GSN CaseMaker
 - ERA Technology, www.era.co.uk
- ISCADE (Integrated Safety Case Development Environment)
 - RCM2, www.iscade.co.uk
- ISIS
 - High Integrity Solutions, www.highintegritysolutions.com
- Freeware Visio Add-on
 - University of York,
<http://www.cs.york.ac.uk/~tpk/gsn/gsnaddoninstaller.zip>



Summary

- Structure of safety cases
 - Evidence of **quality** management
 - Evidence of **safety** management
 - Evidence of **technical** safety
- Safety argumentation – presented using the **Goal Structuring Notation**
 - Elements: Evidence, Strategy, Goal, Context
 - Patterns
 - Modular safety arguments
 - Maintenance of safety arguments
- Generalization: Assurance cases