

# What is wrong about your safety case?

Peter Froome  
pkdf@adelard.com



# The talk covers

---

- A review of 8 safety cases
- Guidance developed to address the review recommendations
- Following the theme of today's seminar, I will concentrate on findings and guidance relating to complex safety arguments

# A safety case is ...

---

A documented body of evidence that provides a *convincing* and valid argument that a system is adequately safe for a given application in a given environment, and that safety will be sustained through life

- To be *convincing* it must “tell a (true) story”

# Review aims

---

To examine a selection of safety cases

- To identify common areas of weakness and opportunities for improvement
- Identify best practice
- Assess the clarity and robustness of safety arguments



# Eight safety cases

---

- Safety cases procured along with system
- Safety case procured separately
- Retrospective safety appraisals
  
- Range of systems from fighting platforms to infrastructure
  
- Varying amounts of documentation and detail

# Results

---

- We found shortcomings in a number of areas ...

# Safety argument approach

---

Safety arguments should be explicit  
& appropriate to the system and  
the analysis performed

- Typically approach for “physical safety” may be different from approach to functional safety
  - However this was not seen  
(although sometimes distinction is not clear-cut, and depends on boundaries of safety case)
  - Sometimes functional safety only relevant in wartime but all cases excluded combat or warfare situations

# Safety arguments

---

- We deduced that all are based broadly on same argument:
  - All hazards are identified
  - All hazards sufficiently mitigated
  - Therefore system is ALARP and hence tolerably safe
- This style may be appropriate: but not for all systems

# Hazard identification completeness

---

- Hazard identification is known to be difficult and is rarely complete
- Significant problem when the safety case argument is based on hazard identification and mitigation
- No justification for completeness of hazard identification is generally provided
- Can validate completeness e.g. by review of similar systems

# Other safety case problems

---

- Accidents and hazards not well distinguished
- Abstraction of hazards (top-level vs. failure modes) incorrect
- Overall risk class judgement wrong or missing
- Too many class "C" risks
- Hazard controls applied to accidents
- Not enough mitigation of hazards rather than accidents
- Problem of organisation and management of long lists of hazards
- Poor ALARP justification
- Inconsistent safety criteria
- Risk apportionment
- Use of in-service data

# Guidance

---

- We produced guidance covering these issues
- Will concentrate on safety argument guidance today ...

# Safety case “story”

---

To tell the (true) story we need to:

- Make an explicit set of claims about the system
- Produce the supporting evidence
- Provide a set of safety arguments that link the claims to the evidence
- Make clear the assumptions and prerequisites underlying the arguments
- Allow different viewpoints and levels of detail

# Viewpoints

---

Last point is key. Viewpoints include:

- Safety specialists involved with detail
- Internal and external regulators
- Operators and managers
- Senior staff who accept system as safe to enter service
  - they don't have time to master fine detail !
- And if things go wrong ...  
lawyers



# Explicit safety arguments

---

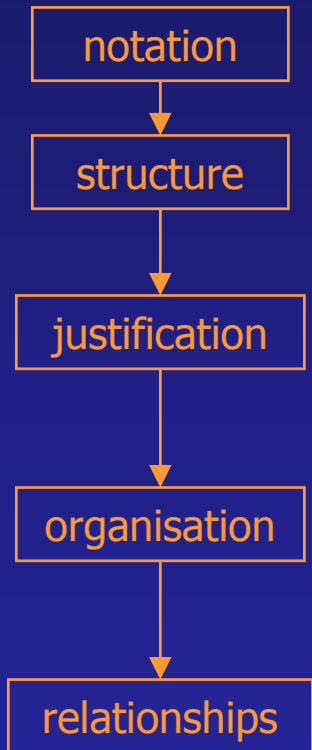
## Advantages of explicit safety arguments:

- The safety case is easier to review
- The safety case should be more convincing
- Negative safety work that does not contribute to the safety argument is less likely to be carried out
- Easier to identify & address potential weaknesses in the argument at an early stage
- Easier to make the safety case a living document

Reduces project risk!

# Process for building safety case

---



- Choose a safety argument notation
- Show the structure of the argument using the chosen notation
- Justify that satisfying the subclaims also satisfies the parent claim
- Organise the safety evidence to support the claims and subclaims chosen
- Relate the claims and subclaims to the safety requirements and hazard log

# Reasoning notation

---

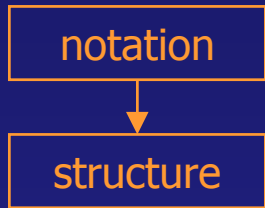
## notation

- Several choices e.g. GSN, Claims-Arguments-Evidence
- Organises the body of evidence supporting the overall safety judgement of the system in a coherent way
- Advantages over purely textual safety cases, which can be cumbersome and difficult to construct and review

The examples in this talk use the "claims-arguments-evidence" notation

# Build structure

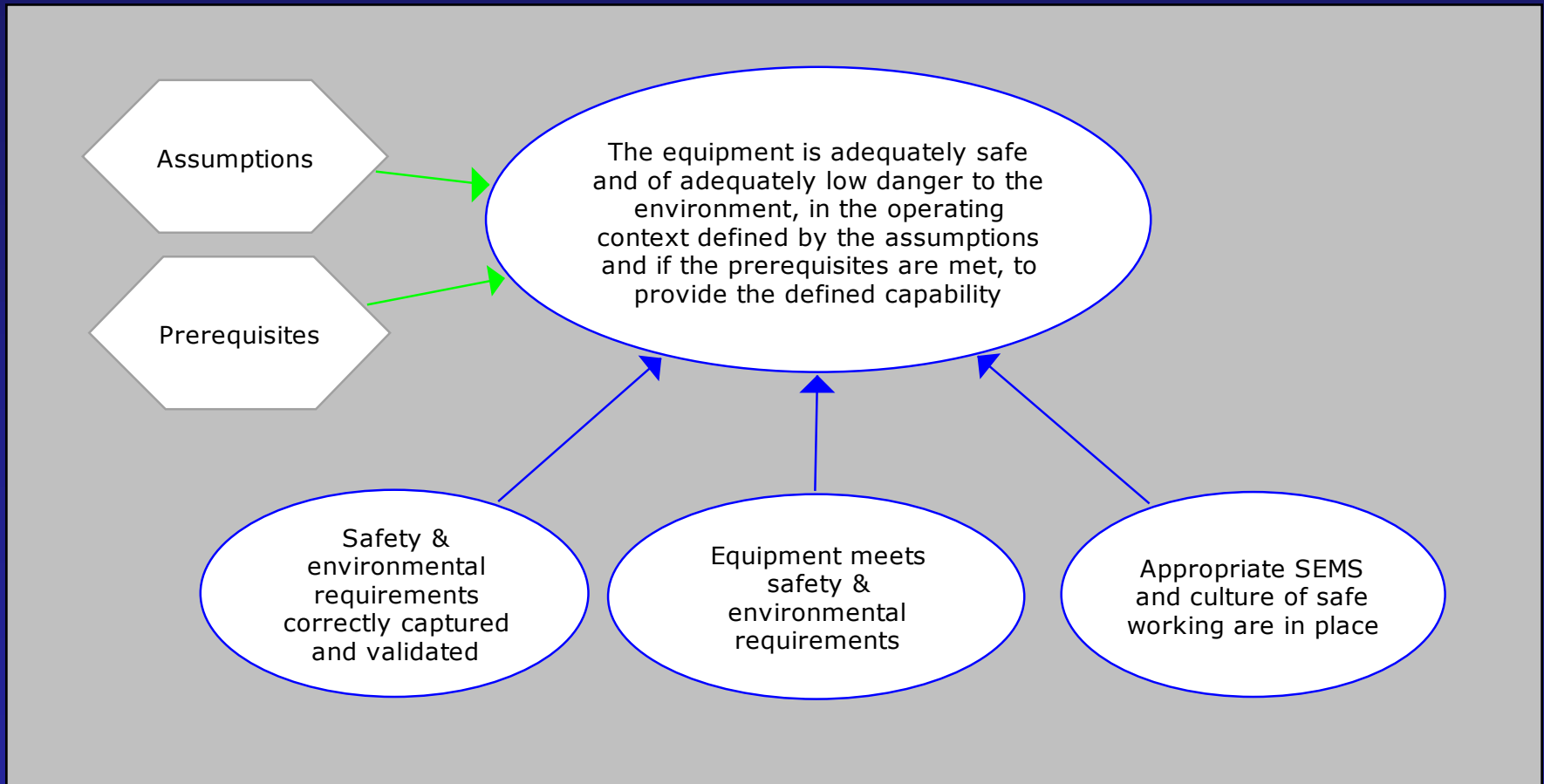
---



- Develop structure using the chosen notation
- Typically, minimum is the overall safety claim and two levels of subclaims

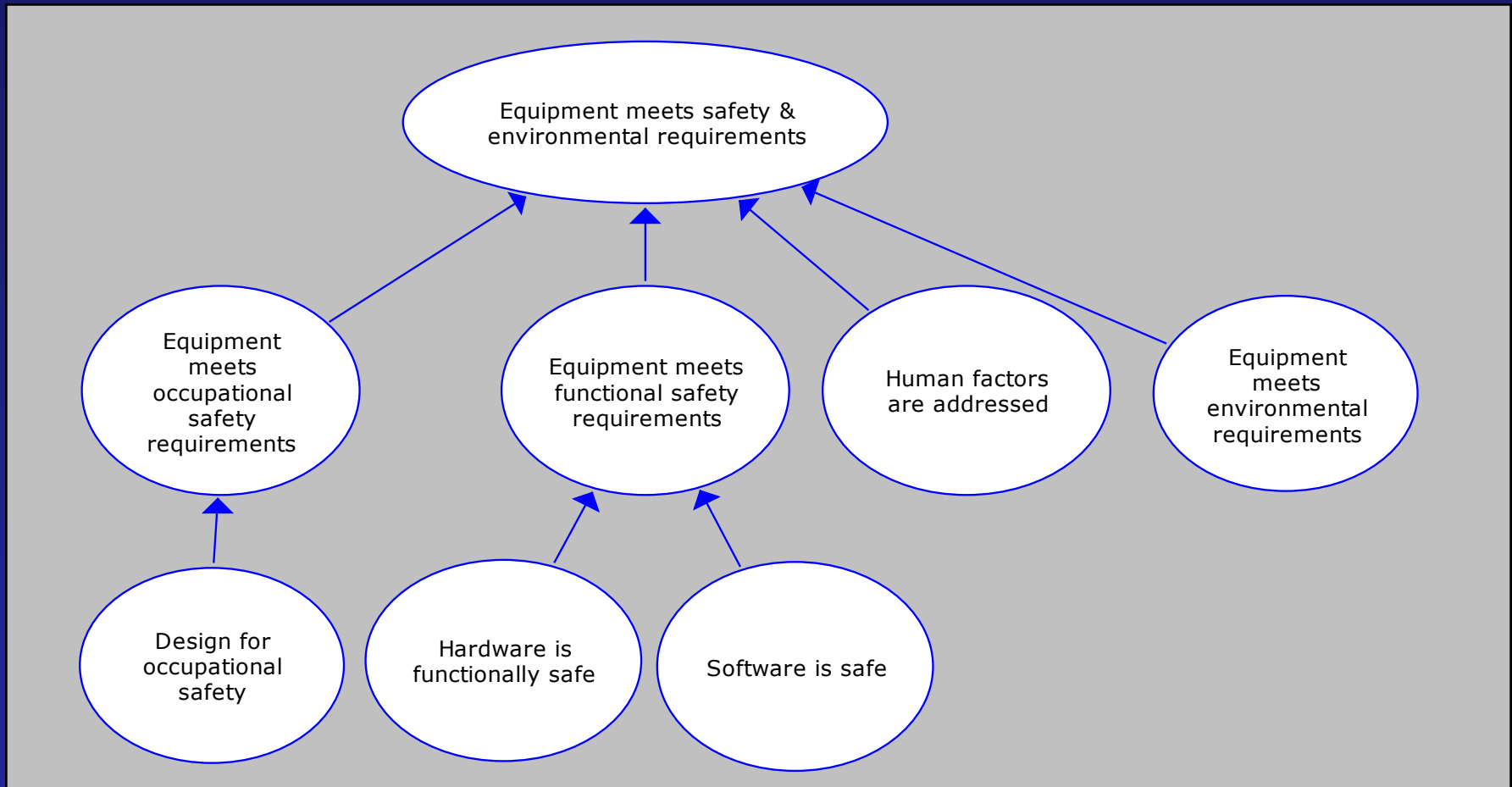
The overall safety claim will typically be of the form:  
"The equipment/system is adequately safe and of adequately low danger to the environment, in the operating context defined by the assumptions and if the prerequisites are met, to provide the defined capability."

# Example top-level claim structure



This is the beginning of the "story"

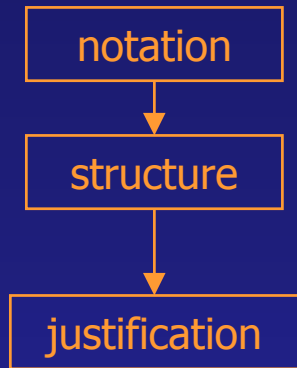
# Example subclaim structure



This is the middle of the "story"

# Justify claim structure

---



- Justify that satisfying the subclaims shows that the parent claim is also satisfied
- Use explicit arguments/strategies

There are four generic arguments/strategies: conformance to standards; analysis; process; and experience (real and simulated)

- Also separation of concerns (physical vs. functional)
- Separation of components (hardware/software)

# Types of claims

---

- Safety case may contain two types of claims

*Direct claims:* Claims about system behaviour supported by sub-claims that demonstrate safety

*Meta-claims:* Claims that support the basis on which direct claims are made, e.g. that sub-claims are sufficient to support a claim (claims about claims)

- Satisfying the meta-claims helps ensure the safety case is valid

# Example meta-claims

---

Meta-claims for structure might include:

- Specified functionality complete
- Specified safety requirements valid
- Integrity targets valid
- Failure rates are realistic

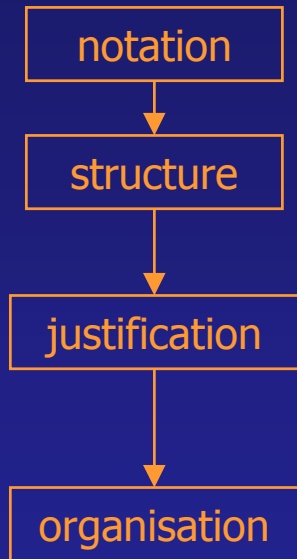
# Risks and robustness

---

- Consider risks to arguments
  - E.g. if argument is based on hazard mitigation, what if hazard identification is incomplete?
- Consider robustness
  - For high-integrity systems, might want several (compensatory) legs of argument

# Organise evidence

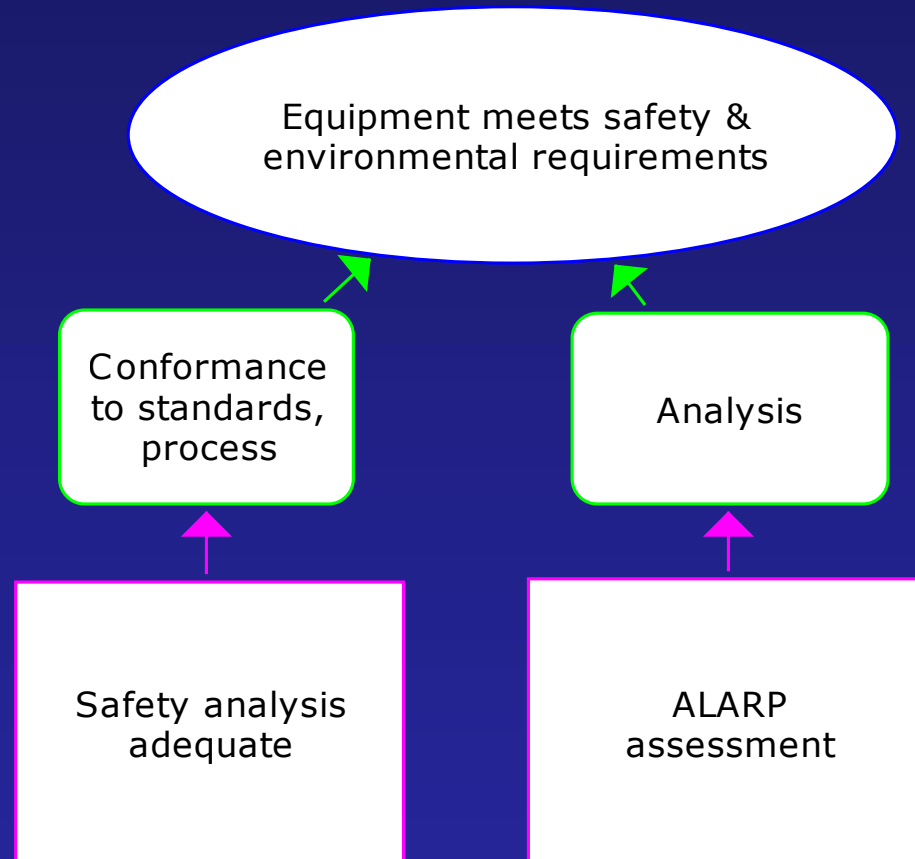
---



- Organise the safety evidence to support the claims and subclaims chosen
- Useful to give summary of evidence and reference to full report
- Useful to use the concept of
  - "direct" evidence (e.g. facts about design, test results, etc.)
  - "indirect" evidence (process, quality standards, competence, etc.)

# Example arguments

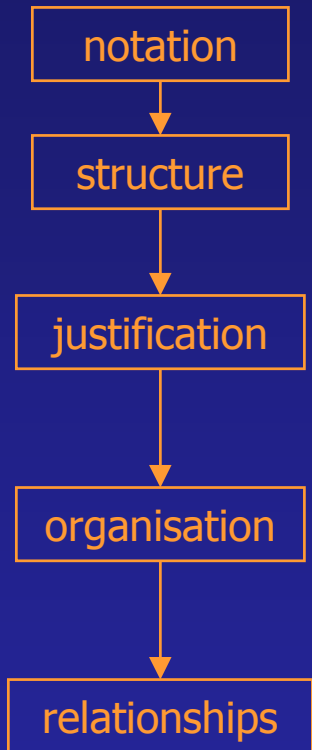
---



This is the end of the “story”

# Relationship to hazard log & reqs

---

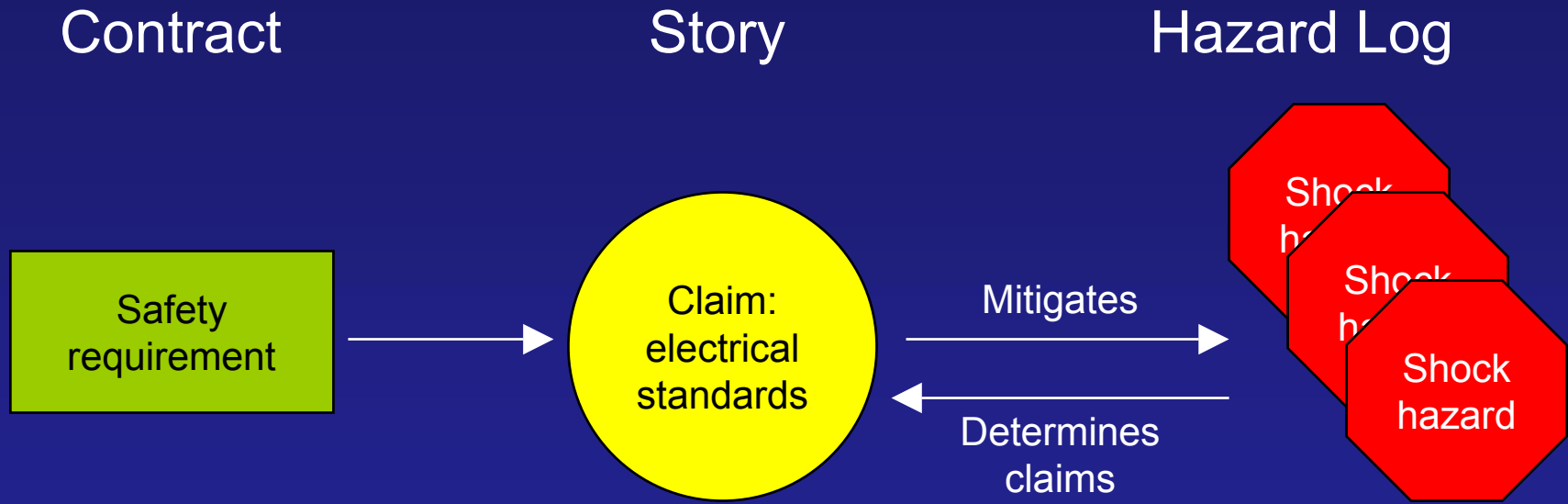


- How does the argument structure relate to:

- The Hazard Log (very detailed)
- Contractual safety requirements
- Derived safety requirements
- ?

# Safety claims & hazards

---



# Conclusions

---

- A number of problems recurred in the safety cases we reviewed
  - Structured approach lacking for safety arguments
  - Inconsistent safety criteria and risk apportionment
  - ALARP and some technical issues badly handled
  - In-service data lacking & little used
- We produced guidance covering these, including
  - Recommendations for argument structure
  - Process for building a well-structured safety case