Certification and Regulation of Software-Intensive Systems

Nancy G. Leveson

Aeronautics and Astronautics
Engineering Systems
MIT

Slides at http://sunnyday.mit.edu/SCC.ppt
Paper at http://sunnyday.mit.edu/SafetyCases.doc

Software “Certification”

- Software is not safe or unsafe on it’s own
  - Only within a particular system design
  - Examples: Therac-25 and Ariane 5
  - Most spacecraft accidents (and many others) involving software have been reused software.
- Certification of software then only makes sense within a particular system

Safety Regulation Approaches

1. **Prescriptive**

   1a. Product

   - Specific design features (e.g., electrical codes)
   - General design features (e.g., fail-safe, protection system)

   1b. Process: process to be followed for

   - General design and operation of the system (e.g., DO-178B)
   - Safe design and operation (e.g., MIL-STD-882, ARP 4761)

2. **Goal or Performance-Based**

   – e.g.: “The likelihood that the ITP equipment provides undetected erroneous information about accuracy and integrity levels of own data shall be less than 1E-3 per flight hour.” (NextGen ITP)

   – Person seeking certification decides how to accomplish goal and how to show that have accomplished goal

Prescriptive: Product-Based

- Specific Product Features (e.g., electrical codes)
  - Encode past experience and lessons learned from accidents
  - Do we want every project to reinvent these?
    - Cost of doing so
    - Potential incompleteness
  - Software?
    - Examples:
      - Requirements completeness criteria in Safeware
      - Design criteria in Engineering a Safer World
  - Should be part of every certification scheme but technology changes and types of products change, new causes of accidents are created, and so we need more than this

Prescriptive: Product-Based (2)

- Specific Design Features
  - Every industry has different approaches to design for safety
    - Nuclear: defense in depth and protection systems
    - Aviation: fail-safe design
  - Different design approaches appropriate for different types of systems (e.g., is there a safe shutdown state?)
  - Software?
    - Are there design features that we want in all safety critical software, e.g., exception handling, checking for out-of-range inputs?, reducing potential for operator error?
    - Again, see Engineering a Safer World

Prescriptive: Process-Based

- General development process to be followed is specified
  - Process and required artifacts
  - Software?
    - DO-178B is an example

Prescriptive: Process-Based (2)

- Safety process to be followed
  - Process used in safety engineering, not general development process.
  - Examples
    - ARP 4761: Fault Hazard Analysis
    - MIL-STD-882: Hazard Analysis, design precedence, etc.
  - Certification usually on evaluation of quality of artifacts of defined process
  - Software?
    - Included in MIL-STD-882 since 1984
    - New hazard analysis methods (like STPA) include software.

Advantages of These Approaches

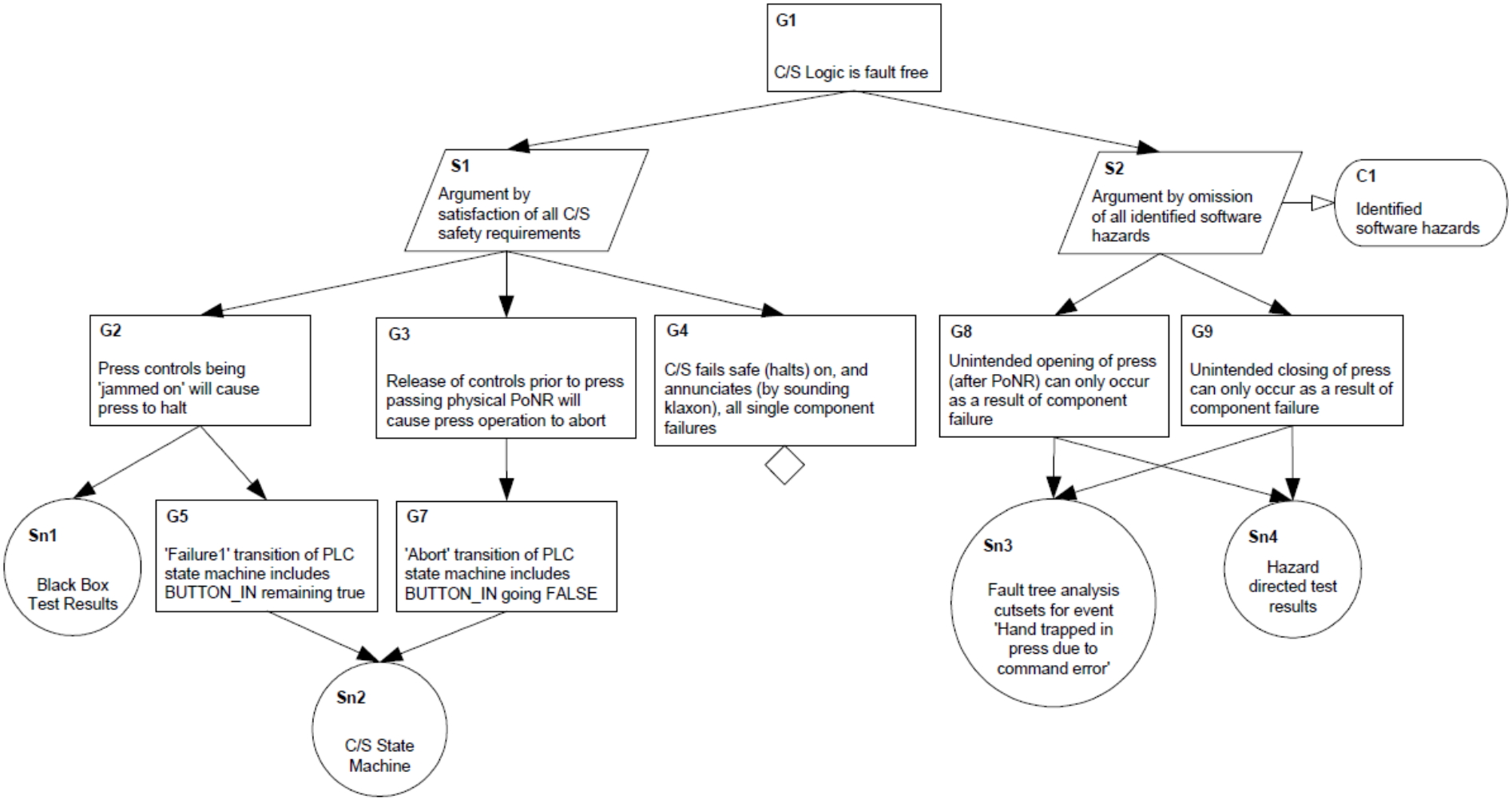
- Relatively straightforward for agencies to implement
  - May license others to evaluate whether product has required features or process followed (e.g., DER, UL ratings). Accountability may be with licensee (P.E.)
  - Uniformity (not subject to whim or capability of a specific regulator)
- Stakeholders can have a say about how the systems they rely on are certified.
  - Users are dependent today on regulatory agencies to assure the safety of their (public) safety. Systems too complex for individuals to assume risk of products they buy or systems they use.
  - Pilot unions, etc., can (and do) participate in standards process and definitions of how systems are certified

Goal or Performance Based

- Focus on desired, measurable outcomes rather than required product features or prescriptive processes.
  - Certification agencies set goal and up to applicant to decide how to accomplish that goal.
  - Examples:
    - An aircraft navigation system must be able to estimate its position to within a circle with a radius of 10 nautical miles with some specified probability
    - “The likelihood that the ITP equipment provides undetected erroneous information about accuracy and integrity levels of own data shall be less than 1E-3 per flight hour”
- How prove that have achieved goal?
  - Safety case: an argument that system will be acceptably safe in a given environment.

Safety Cases: Are They Effective?

- No system is completely safe. So any argument that it is, is by definition wrong.
- What about “acceptably safe”?
  - Acceptable to whom? Those selling it? Those that may be killed?
  - Responsibility of regulatory agencies is to protect the public. Those creating products have a moral responsibility for that but a fiduciary responsibility to their shareholders. A clear conflict of interest exists.
  - An advantage of prescriptive standards is that potential victims (e.g., pilots and ALPA, airline passenger associations, citizens living near nuclear power plants) can participate in the process of establishing standards for evaluating risk.
  - With safety case, company can create any argument and potential victims cannot review this (usually include proprietary information) or have a say in what types of procedures are used in design or in analysis and evaluation because the argument used in determined by the producer of the system.


G1
C/S Logic is fault free
S1
Argument by satisfaction of all C/S safety requirements
S2
Argument by omission of all identified software hazards
C1
Identified software hazards
G2
Press controls being 'jammed on' will cause press to halt
G3
Release of controls prior to press passing physical PoNR will cause press operation to abort
G4
C/S fails safe (halts) on, and annunciates (by sounding klaxon), all single component failures
G8
Unintended opening of press (after PoNR) can only occur as a result of component failure
G9
Unintended closing of press can only occur as a result of component failure
Sn1
Black Box Test Results
G5
'Failure1' transition of PLC state machine includes BUTTON_IN remaining true
G7
'Abort' transition of PLC state machine includes BUTTON_IN going FALSE
Sn2
C/S State Machine
Sn3
Fault tree analysis cutsets for event 'Hand trapped in press due to command error'
Sn4
Hazard directed test results

Argument-Based Safety Cases

What is going on here?

- There is always a way to argue that something is safe, whether it is or not. Always possible to produce evidence that something is safe.
- Major problem is *Confirmation Bias*

Confirmation Bias

- People will focus on and interpret evidence in a way that confirms the goal they have set for themselves
  - If the goal is to prove the system is safe, they will focus on the evidence that shows it is safe and create an argument for safety.
  - If the goal is to show the system is unsafe, the evidence used and the interpretation of available evidence will be quite different.
  - They also tend to interpret ambiguous evidence as supporting their existing position.

Confirmation Bias (2)

- Experiments show people tend to test hypotheses in a one-sided way, by searching for evidence consistent with the hypothesis they hold at a given time.
  - Rather than searching through all the relevant evidence, they ask questions that are phrased so that an affirmative answer supports their hypothesis.
  - A related aspect is the tendency for people to focus on one possibility and ignore alternatives.
  - Fault tree experiment (Fischoff, Slavin, Lichtenstein)

Additional Considerations

- Little to no evaluation has been done
- Have been criticized as a causal factor in accidents
- Value of system safety is doing what engineers do not do. A different viewpoint.
  - Focus on why not safe, not why safe

Safety Cases: Are They Feasible and Practical?

- Certifier must evaluate the argument. Are they qualified to do so? Is anyone?
- Number and qualifications of government employees required may be impractical.
  - Data from oil industry
  - Not only design but operations
- Companies already complain that evaluating to prescriptive processes takes too long. Without prescriptive standards, will take much longer.

Data from Offshore Oil

- In Norway, the PSA has approximately 160 employees, of which approximately 100 perform compliance and audit related tasks regulating 105 offshore installations. Each of these 100 employees has a postgraduate (Masters Degree), or equivalent level of training, in one or more areas of expertise, including drilling, petroleum engineering, structural engineering, and reliability engineering.
- In contrast, the Bureau of Safety and Environmental Enforcement (BSSE) and the U.S Coast Guard share approximately 60 billeted offshore inspectors for over 3,500 offshore installations.

Rena Steinzor: Lessons from the North Sea: Should "Safety Cases" come to America?

- Confidentiality and risk levels tolerated by British system conflict with both spirit and letter of American law.
  - British regulations: plans can be no more protective than preventing 1 in 1,000 worker deaths and require operators to spend no more than $1.5 million per life saved.
  - More lax than comparable American legal requirements.
  - How determine how many lives will be lost? (Pinto case)
- Would require increasing available government agency overseers by orders of magnitude.
- British conditions in North Sea suggest alarming neglect of physical infrastructure that assures safety (undermining claims of efficacy of proponents)

An Alternative Proposal for Certification of Software-Intensive Systems

- Submit plan first (provides some flexibility along with oversight)
  - Probably will include reference to both product standards (specific and/or general design features) and process standards
  - Identify accidents to be considered (perhaps agency identifies these)
  - Identify hazards to be considered and use safety-guided design to eliminate or control them
- Provide hazard analysis and how eliminated or controlled identified hazards. Need to use hazard analysis techniques that include software and human factors as well as consider operations
- Report limitations of what was done
  - Uncertainties
  - Assumptions

Other Issues

- Need to maintain certification during lifetime. Not a one-time thing.
- Need to start certification process early, not an after-the-fact compliance exercise
  - Integrate analysis into development process and system documentation
- Must consider worst-case analysis, not just expected case (design basis accident in nuclear power)
- Needs to be comprehensive and consider all factors, including human factors, management structure and decision-making, etc. (not just factors that can be formulated mathematically)
- Use objective evidence (testible, verifiable), not just probabilistic models of the system unless the probabilities can be verified and tested

Current Needs

- Biggest need is for comparison and effectiveness research on real systems.
  - Stop relying on “proof by vigorous handwaving,” toy problems
- Identify what mean by product and design features for software.
- Create new types of hazard analysis that include software, cognitive human factors (not just hardware and simple human slips) and interactions among components in complex systems.

References

- Wassyng, Maibaum, Lawford, and Bherer. ‘Software Certification: Is There a Case Against Safety Cases?” R. Calinescu and E. Jackson (eds.), Monterey Workshops 2010, LNCS 6662, pp. 206-227, Springer Verlag, 2011.
- Rena Steinzor, “Lessons from the North Sea: Should ‘Safety Cases’ Come to America?” Boston College Symposium on Environmental Affairs Law Review, 2010.