



Introduction to Requirements Engineering for Safety-Critical Systems

Camilo C. Almendra

cca3@cin.ufpe.br

LER - Laboratório de Engenharia de Requisitos

Seminar

March 16, 2018 – Recife, Brazil



Summary

- Introduction to safety-critical systems
- Safety engineering
- Requirements and safety
- Challenges and research directions



INTRODUCTION



Is this safe?



How could these two trains end up in this situation?

What will happen?

If they collide, which are the consequences?

Is there anything that could have been done to avoid these situation?

Is there anything that can be done now to avoid collision?



Is this safe?



Safety is **freedom from accidents**.

Accident is an **unplanned event that results in a loss** (human life or injury, property or environmental loss, ...).

Hazard is **a system state** that together **with worst-case environment** conditions will lead to an accident.



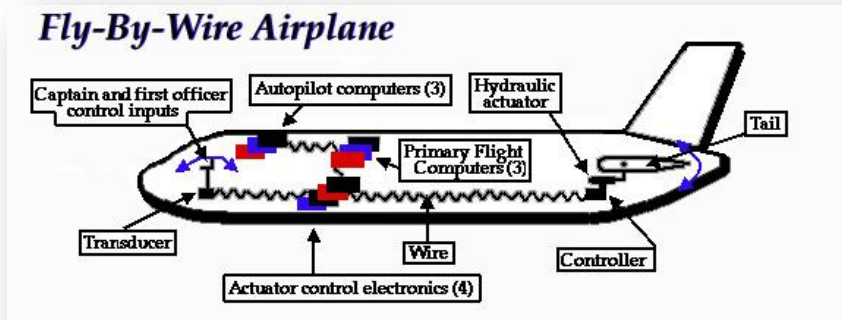
Safety-Critical Systems

- What is a safety-critical system?

A safety-related system comprises everything (**hardware, software and human elements**) necessary to carry out one or more safety functions, where failure of the safety function would give rise to a significant increase in the risk to the safety of persons and/or the environment.



Safety-Critical Systems



Therac-25 Medical Accelerator
1985-1987

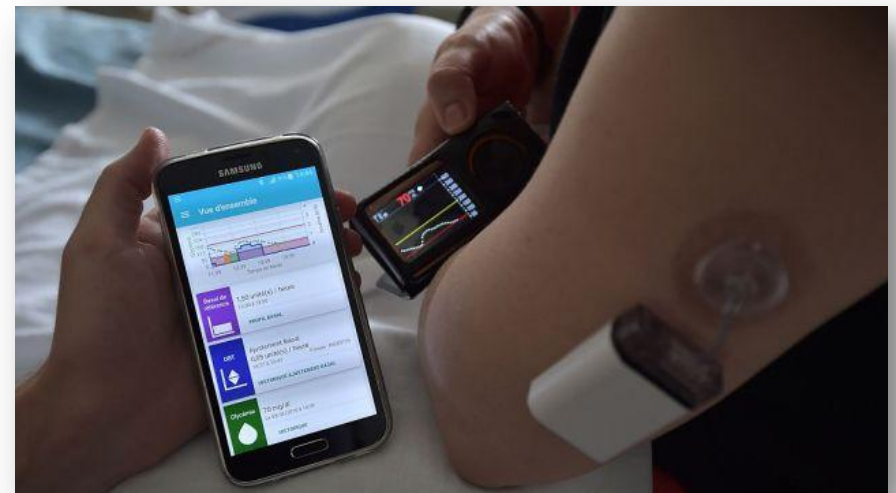
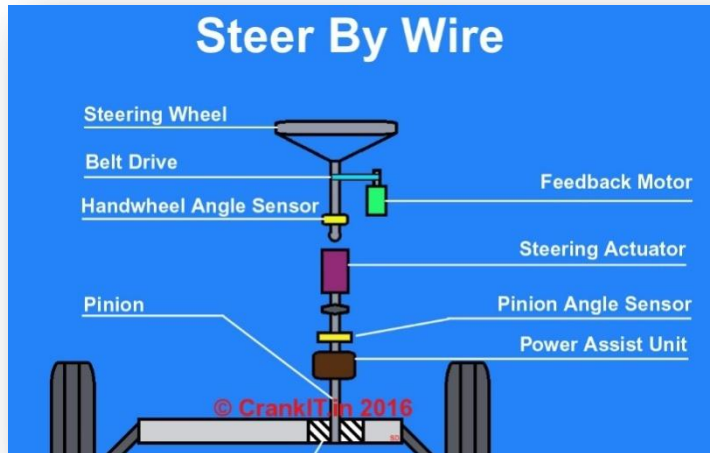
Radiation therapy device malfunctions, delivers lethal doses at several facilities

The 25 was an improved version of an older model

It could deliver beta-particles (electron beam) or x-rays

A photograph of the Therac-25 Medical Accelerator, a large, white, L-shaped machine used for radiation therapy. The machine is positioned in a room with a patient bed and other medical equipment.

Critical behavior controlled by software

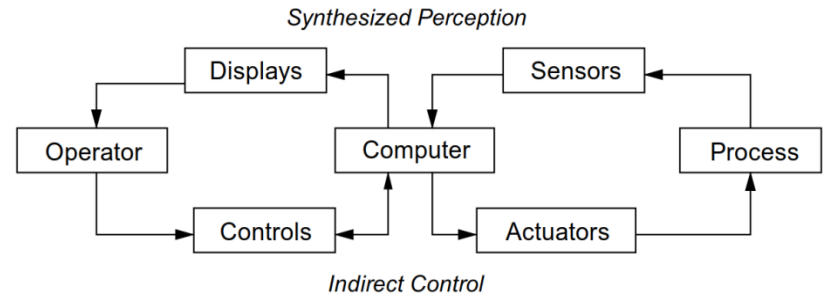




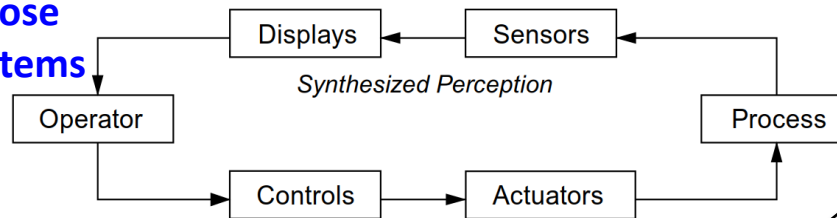
Safety-Critical Systems

Why is **Software Engineering** **important** for safety-critical systems development?

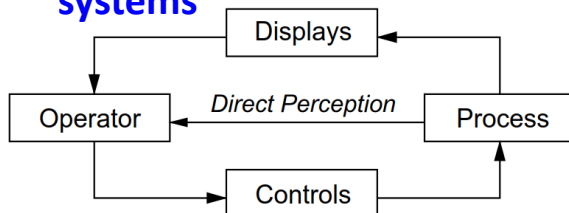
**Computer
in the loop**



**Special-purpose
electronic systems**



**Electric/Mechanical
systems**



**Rise in complexity
&
Increase use of software**



Safety-Critical Systems

Critical function unintendedly controlled by third-party software

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT



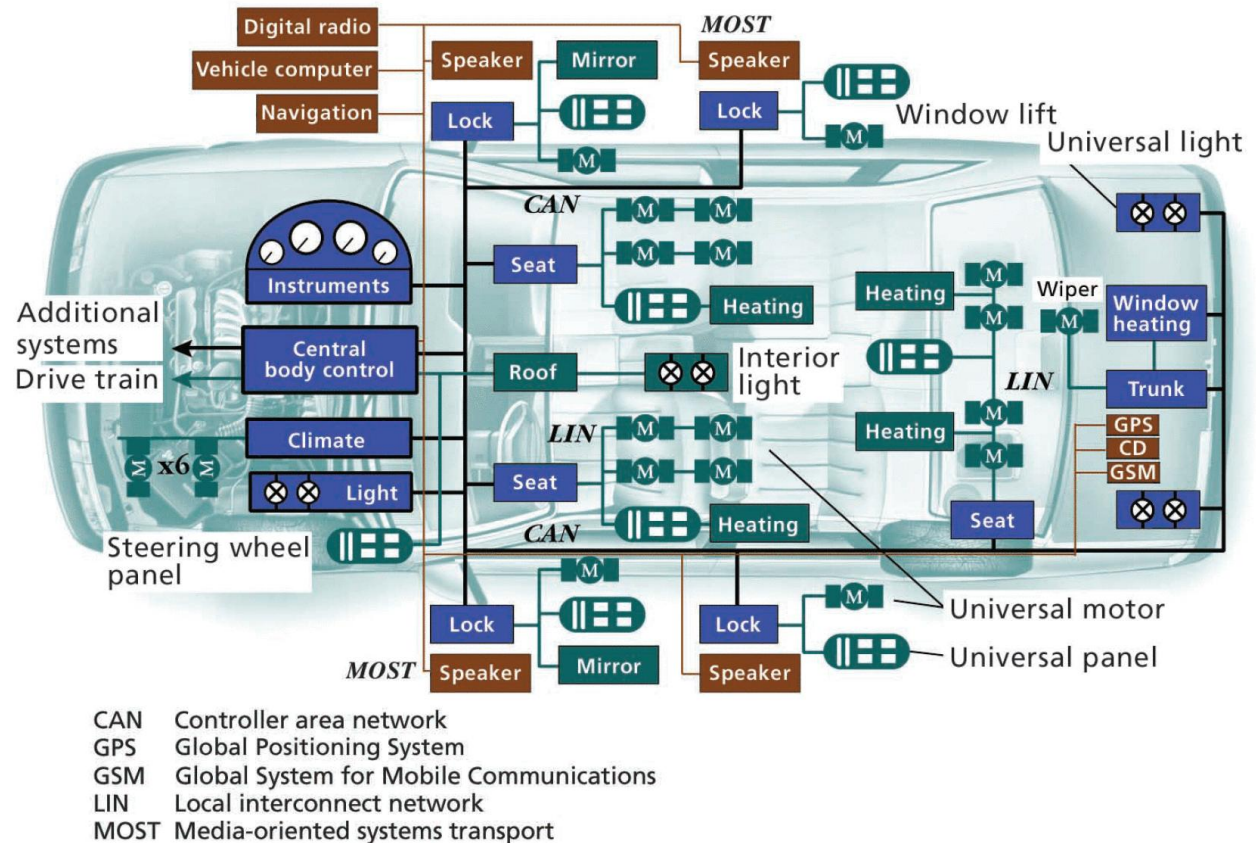
*“As the two hackers remotely toyed with the air-conditioning, radio, and windshield wipers, I mentally congratulated myself on my courage under pressure. **That's when they cut the transmission.**”*

*“Their code is an automaker's nightmare: software that lets hackers send commands **through the Jeep's entertainment system to its dashboard functions, steering, brakes, and transmission**, all from a laptop that may be across the country.”*



Safety-Critical Systems

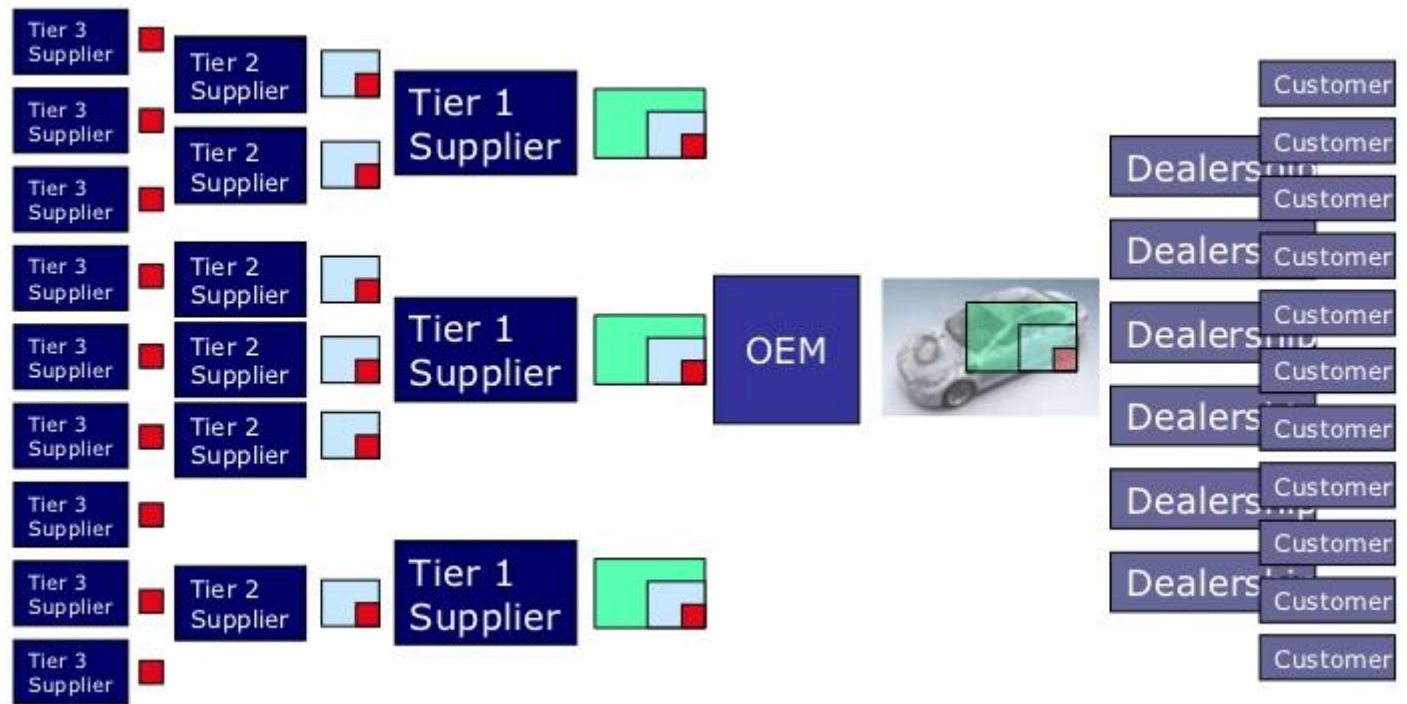
It's not anymore a matter of guaranteeing the safety of a **single component**





Safety-Critical Systems

Also, it's not anymore a matter of dealing only with
your own developed components



Source: <https://www.slideshare.net/blackducksoftware/is-auto-the-new-android>



SAFETY ENGINEERING



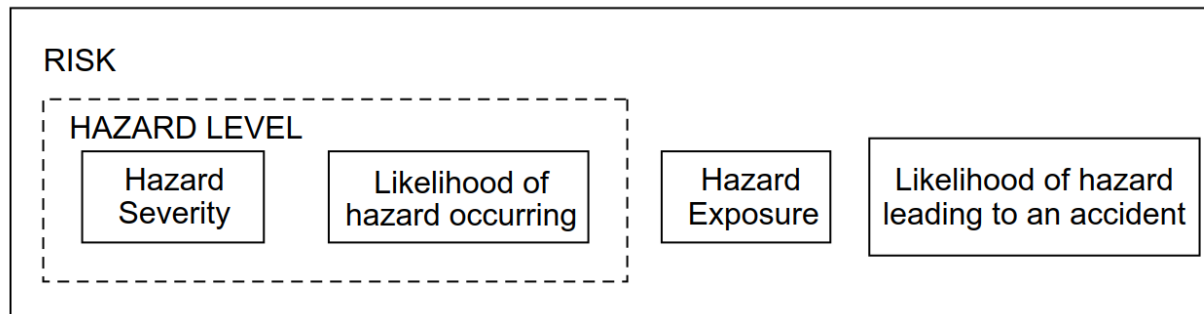
Safety Engineering

- System safety
 - The application of engineering and management principles, criteria, and techniques to **achieve acceptable risk** within the constraints of operational effectiveness and suitability, time, and cost throughout all phases of the system life-cycle
- System safety engineering
 - An engineering discipline that employs specialized knowledge and skills in applying scientific and engineering principles, criteria, and techniques to **identify hazards and then to eliminate the hazards or reduce the associated risks** when the hazards cannot be eliminated.



Safety Engineering

- Hazard/Risk Analysis
 - The process of **identifying hazards/risks** and their potential causal factors.
- Hazard Level
 - A function of the hazard **severity** (worst case damage) and the **likelihood** (quantitative or qualitative) of its occurrence.
- Risk Factors
 - **Factors leading** to an accident, including hazard, states and environment associations

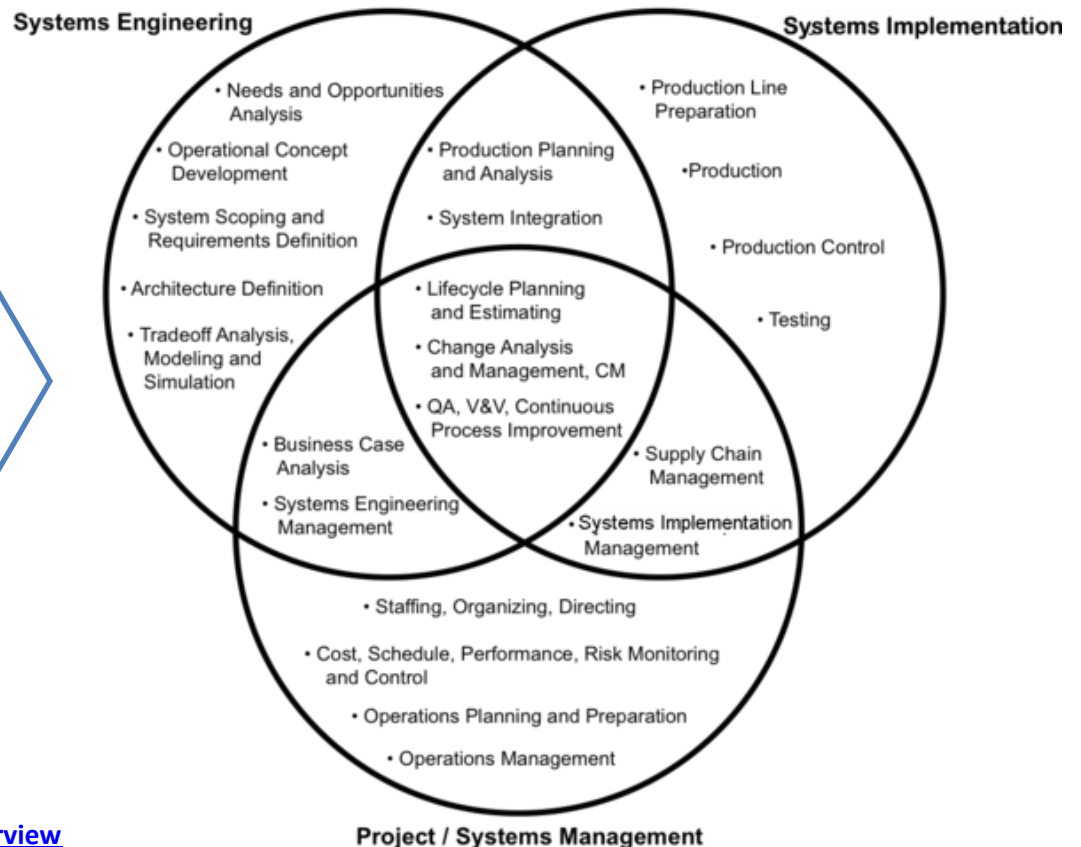
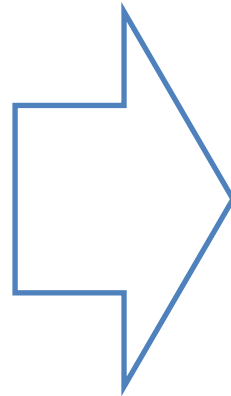




Safety Engineering

- Safety engineering is basically system engineering

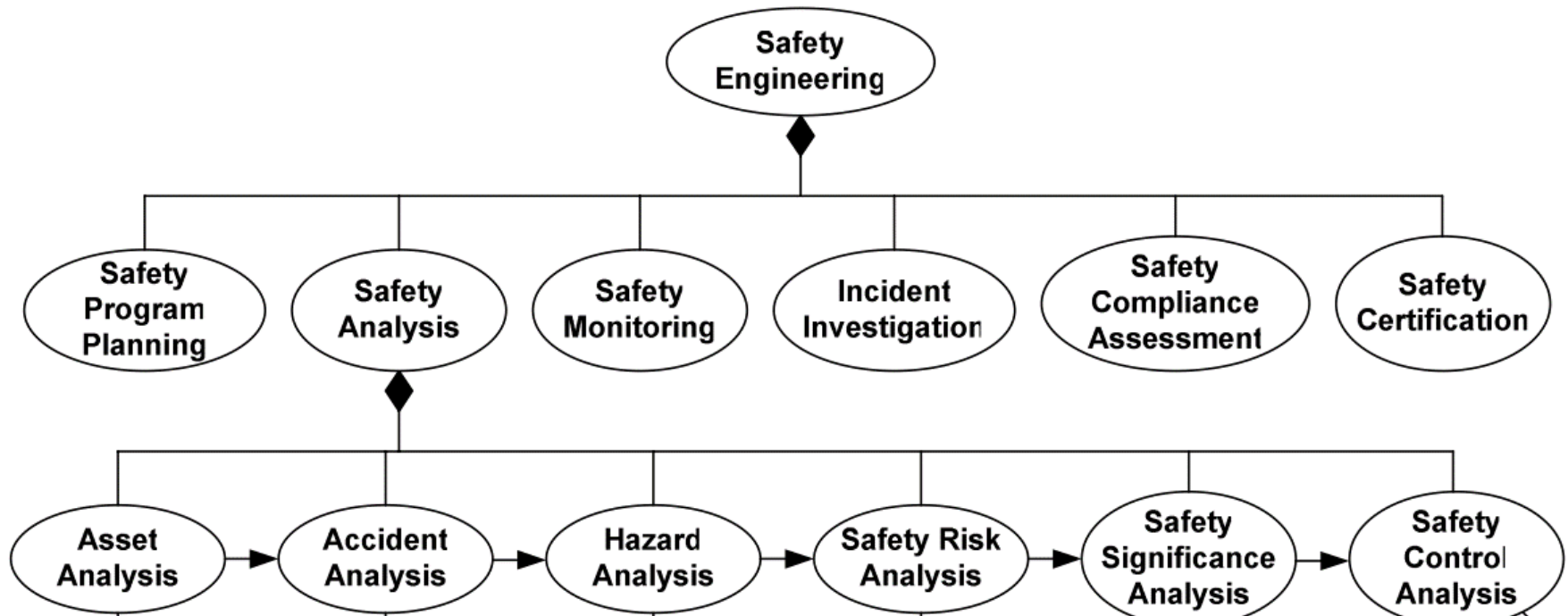
Safety is a **property** of the system, influencing it's entire lifecycle





Safety Engineering

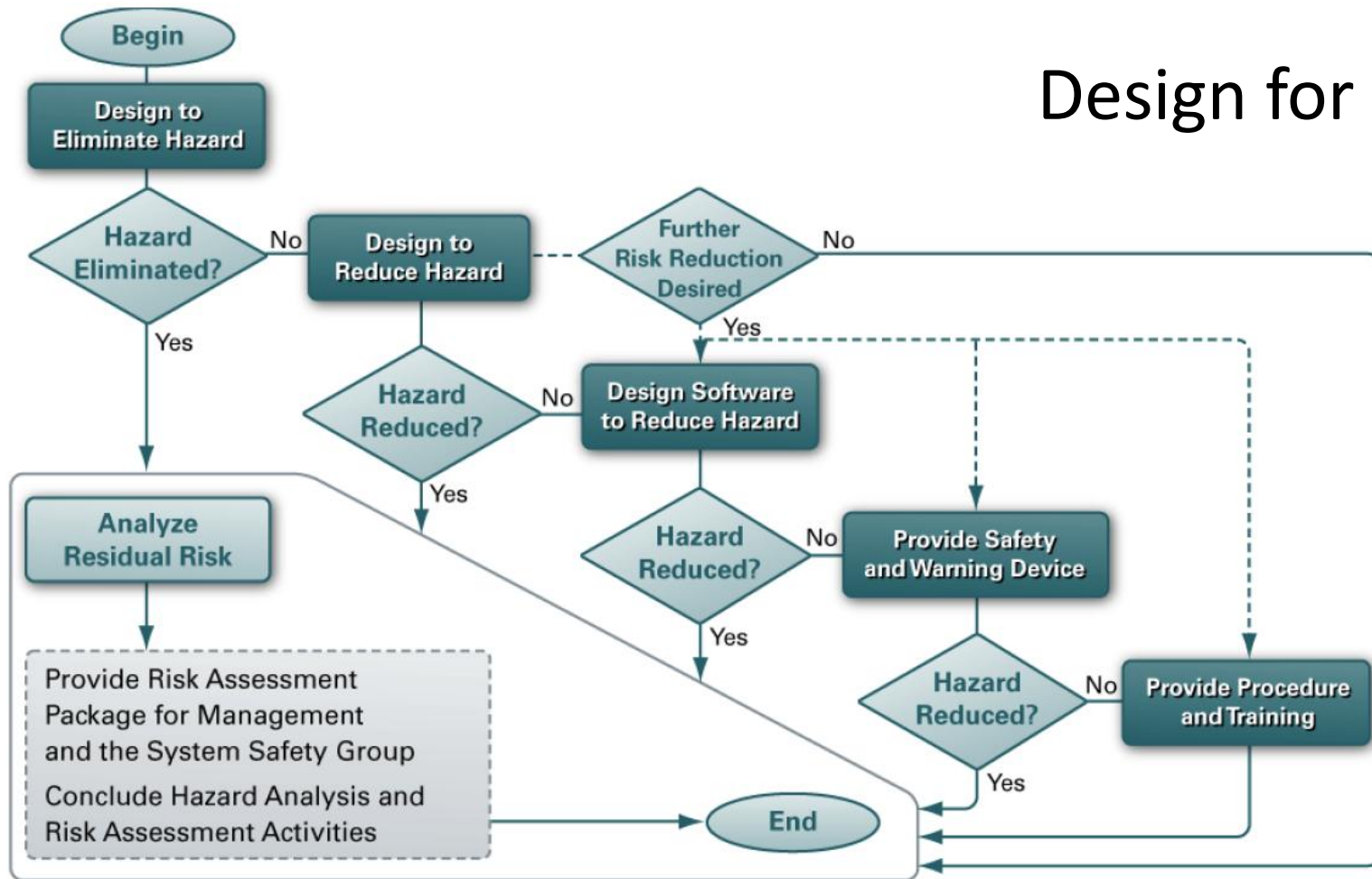
Practices and Techniques





Safety Engineering

Design for Safety



DoD_SSH_017a



REQUIREMENTS ENGINEERING AND SAFETY



RE & Safety

- Developing SCS involves some additional activities not common in non-critical contexts
 - Define high-level safety requirements
 - Identify hazard and risks
 - Investigate if the combined behavior of system's components may lead to hazards
 - Refine safety requirements towards system's design
 - Assure system's safety
- Hazard elimination or mitigation impacts system's requirements:
 - It may derive new functional requirements
 - It may derive new non-functional requirements
 - It may derive new architecture decisions
 - Or it may impact existing ones

**New
Activities/
Processes**

**New
Artifacts/
Information**



RE & Safety

- How to classify these requirements?

“Fire sprinklers shall be activated in 200.0 milliseconds after temperature raises above 120.0 degrees Celsius.” **PERFORMANCE or SAFETY?**

*“External third-party devices shall **not** be able to issue commands to core vehicle control systems.”* **SECURITY or SAFETY?**

“The robot shall have a physical stop button.” **USABILITY or SAFETY?**

“Multimedia system devices must synchronize with vehicle’s dashboard through CAN (Controller area network).” **Is this related to safety somehow?**



RE & Safety

- Classifying safety-related requirements

Category	Description	Typical Sources
Safety Requirements	“Pure” high-level requirements, associated to quality factors	Standards, Regulations, Environment, Domain
Safety-Significant Requirements	System requirements associated with hazards	Hazard analysis output
Safety System Requirements	Requirements added to ensure safety of system/componentes	Hazard elimination or mitigation decisions
Safety Constraints	Constraints intended to ensure safety	Hazard elimination or mitigation decisions



RE & Safety

- Implications of safety for requirements engineering
 - Initial safety analysis may generate new or modified software requirements
 - Requirements that compose the overall system functionality may be themselves a source of hazards
 - Design constraints may also be source of hazards
 - Off-the-shelf components may bring new and unexpected hazards scenarios
 - New mitigation decisions need to be analysed regarding safety (restarting the cycle)



CHALLENGES AND RESEARCH DIRECTION



Research Problem

- **Gaps or inconsistencies in software specifications** may cause accidents or irreversible damages [Favarò et al. 2013]
- Reports from SCS-related accidents brought conclusions **tracing failures back to errors in requirements engineering activities** [Leveson 2011, Hatcliff et al. 2014]
- **Flawed interaction** among components is the most likely origin of software-related accidents [Leveson 2011]
- Need for **integrated approaches** for safety analysis (SA) and requirements engineering (RE) [Heimdahl 2007, Sikora et al. 2012].



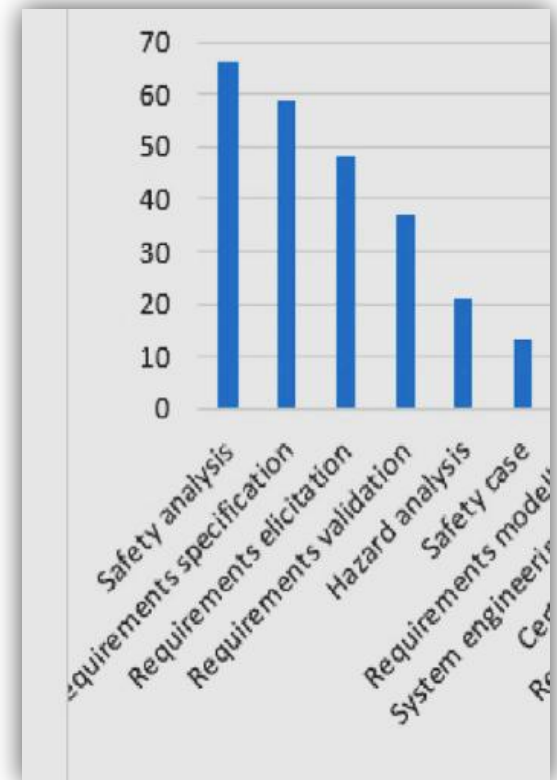
Research Problem

- How such collaboration could be fostered?
 - Better education for both computer scientists and engineers [Lutz 2000, Heimdah 2007]
 - Move towards model-driven methods [Heimdahl 2007, Hatcliff 2014]
 - Explicit incorporation of safety analysis into RE activities [Sikora et al 2012]
 - Better traceability between RE and SA artefacts [Sikora et al 2012]



Research Problem

- Eduardo Martins (UNIFESP, Brazil) and Tony Gorschek (BTH, Sweden) published a literature review on Requirements Engineering for Safety-critical systems (IST journal, 2016)
- They found a number of proposals (165 studies) to support **requirements specification** (39%) and **safety analysis** (43%):
- A common topic was **capturing and definition of safety requirements** during requirements or safety analysis activities
- However, **few works focus** on a structured or **cooperative way of work** between disciplines (7%)





Conclusions

- Software engineering is a relevant topic for Safety-Critical systems development
- Requirements engineering and Safety engineering need to be integrated
 - Both long term, consolidated disciplines
- Industry and academy recognizes a number of challenges to be addressed



Main References

- Favarò, F. M., Jackson, D. W., Saleh, J. H., and Mavris, D. N. (2013). **Software contributions to aircraft adverse events: Case studies and analyses of recurrent accident patterns and failure mechanisms**. Reliability Engineering & System Safety.
- Firesmith, D. G. (2005). **A taxonomy of safety-related requirements**. In International Workshop on High Assurance Systems (RHAS'05).
- Hatcliff, J., Wassyng, A., Kelly, T., Comar, C., and Jones, P. (2014). **Certifiably safe software-dependent systems: challenges and directions**. In Proceedings of the on Future of Software Engineering, ACM.
- Heimdahl, M. P. (2007). **Safety and software intensive systems: Challenges old and new**. In 2007 Future of Software Engineering, pages 137–152. IEEE Computer Society.
- Leveson, N. (2011). **Engineering a safer world: Systems thinking applied to safety**. Mit Press.
- Lutz, R. R. (2000). **Software engineering for safety: a roadmap**. In Proceedings of the Conference on The Future of Software Engineering, ACM.
- Martins, L. E. G. and Gorschek, T. (2016). **Requirements engineering for safety-critical systems: A systematic literature review**. Information and Software Technology, 75:71–89.
- Sikora, E., Tenbergen, B., and Pohl, K. (2012). **Industry needs and research directions in requirements engineering for embedded systems**. Requirements Engineering, 17(1):57–78.



Upcoming Seminars

- **Integration between Requirements Engineering and Safety Analysis** *by Jéssyka Vilela on May 4th*
- **Specifying Safety Requirements using GORE languages** *by Moniky Ribeiro on June 1st*
- **Safety certification** *by Camilo Almendra (TBD)*

Thank you. Contact: cca3@cin.ufpe.br