Session 2 summary

(Reconfigured session with titles from sessions "*Safety management and design*" and "*Autonomous vehicles*")

Presentations:

Marcus Völp (University of Luxembourg): "Towards sustainable safety and security in autonomous vehicles"

Francesco Brancati (ResilTech) “A methodology to ensure safety (certification) of complex software in safety critical automotive systems"

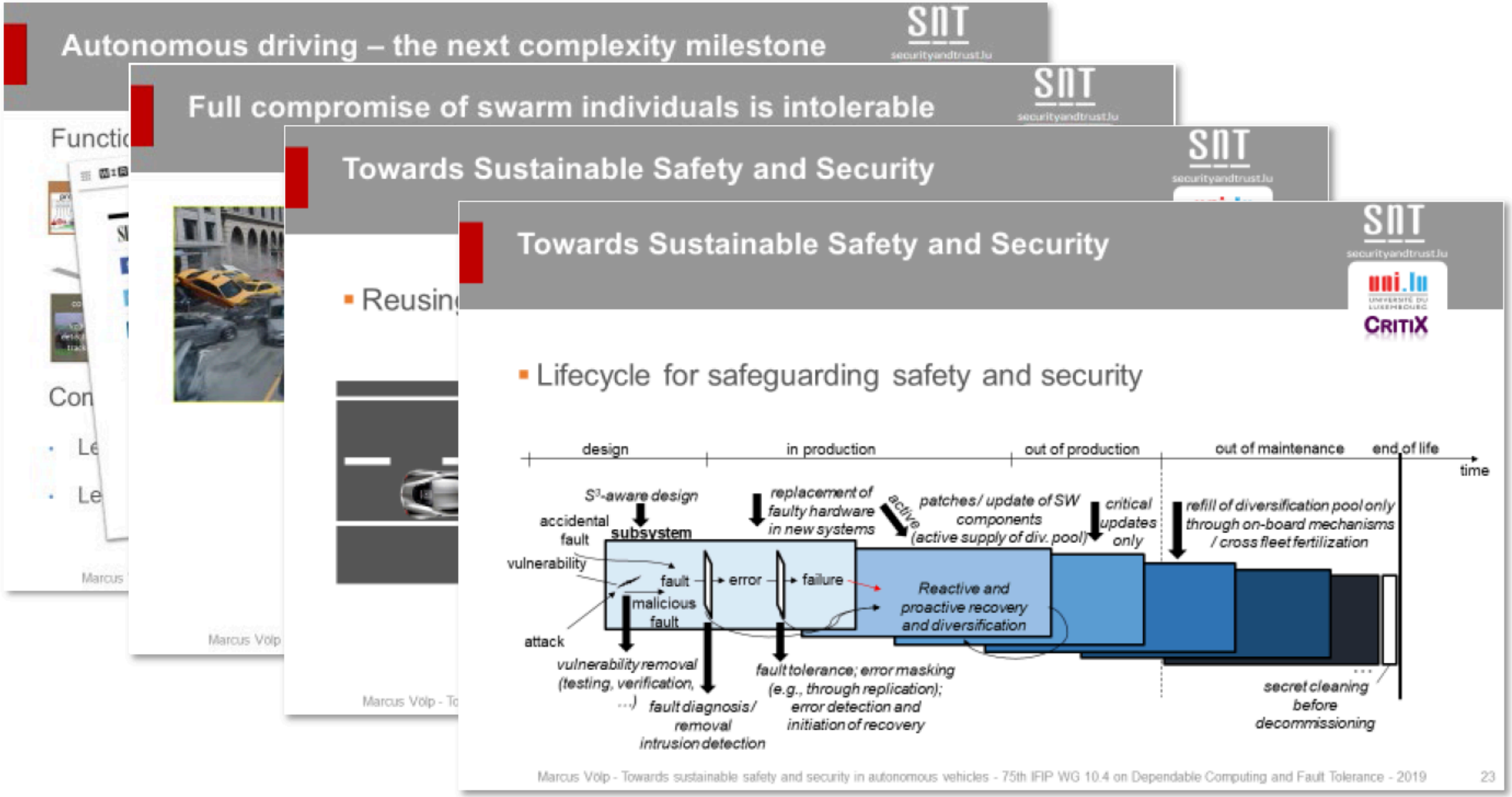
Towards sustainable safety and security in autonomous vehicles

- motivations/background:
  - "Autonomous driving –the next complexity milestone"
    + extrapolating trend of complexity in car automation
  - added complexity/challenges: complexity of envisaged designs (cars and 'ecosystems', dictated by required functions for level 5 autonomy; challenges of perception, complexity of ethical decisions; reputation issues for vendors/promoters; keeping safe *from attackers*

Towards sustainable...

- proposed directions
  - fault/intrusion tolerance, incl. diversity; proactive recovery; maintaining "diversity pool"
    - special attention to need to avoid complete compromise of a 'swarm' member
  - at application level, "plan B" for manoeuvres
  - a complete lifecycle to sustain safety and security
    - high quality development, V&V: F/IT architecture
    - while in use: patches and maintenance of diversity pool
    - with safety/security oriented management, with dwindling effort, until manged complete decommissioning

Towards sustainable...



Autonomous driving – the next complexity milestone

Functi

Con

- Le
- Le

Marcus

Full compromise of swarm individuals is intolerable

Marcus Völp

Towards Sustainable Safety and Security

- Reusin

Marcus Völp - To

Towards Sustainable Safety and Security

- Lifecycle for safeguarding safety and security

design
in production
out of production
out of maintenance
end of life
time
S^3-aware design
accidental fault
subsystem
vulnerability
attack
fault
malicious fault
error
failure
replacement of faulty hardware in new systems
active
patches/ update of SW components (active supply of div. pool)
critical updates only
refill of diversification pool only through on-board mechanisms / cross fleet fertilization
Reactive and proactive recovery and diversification
vulnerability removal (testing, verification, ...)
fault diagnosis/ removal intrusion detection
fault tolerance; error masking (e.g., through replication); error detection and initiation of recovery
...
secret cleaning before decommissioning

Marcus Völp - Towards sustainable safety and security in autonomous vehicles - 75th IFIP WG 10.4 on Dependable Computing and Fault Tolerance - 2019 23

A methodology to ensure safety (certification) of complex software in safety critical automotive systems

Motivation:

- growth in critical functions and their implementation with computer solutions coming from less critical application
- increasing need for built-in error detection
- in the framework of standard-driven practices
- need to support companies in appropriate application of standards (viz ISO 26262

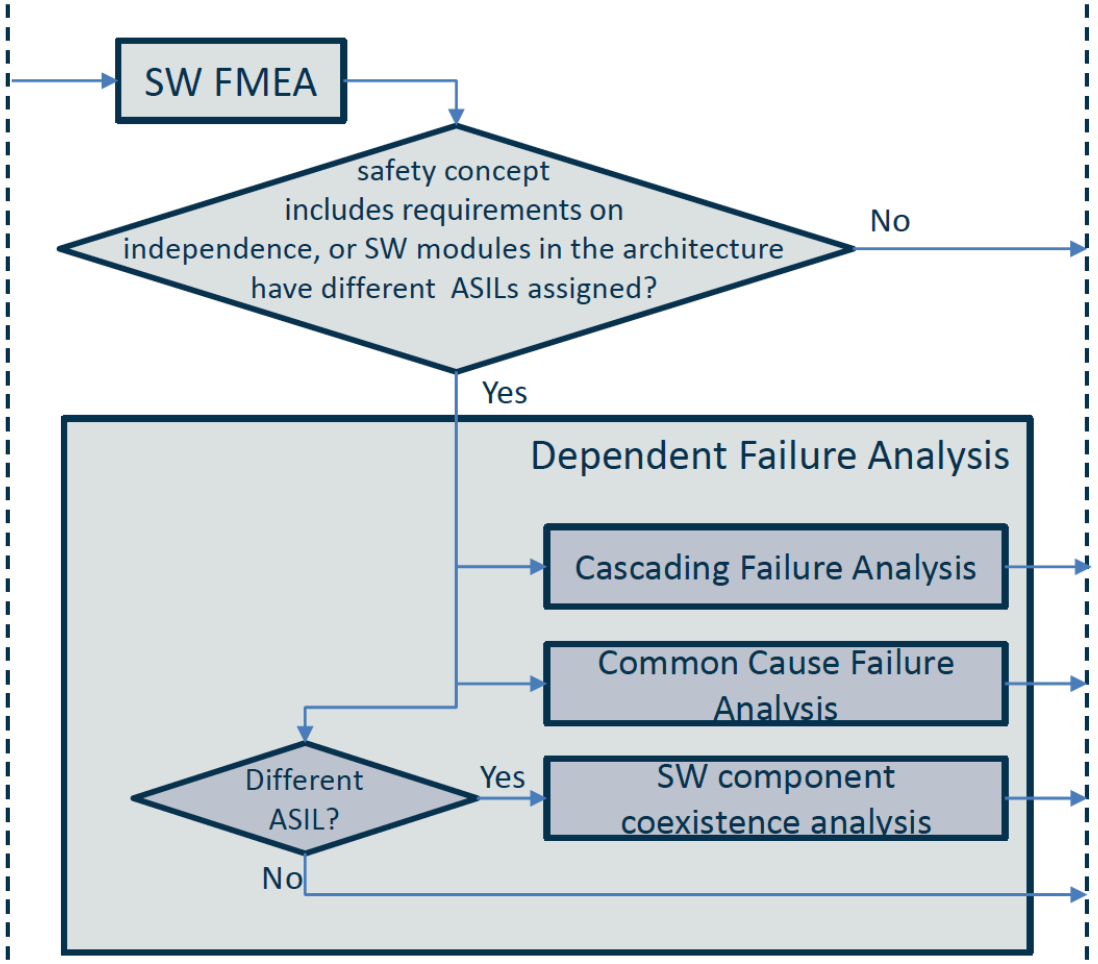
main goals:

- assist verification of safety concept
- verify the coexistence criteria among the software components
- support the specification of safety mechanisms at software architecture level
- observation: can try to evolve trade-offs between architectural changes versus fault-removal techniques.

Main techniques: SW -FMEA FMEA and DFA

methodology to ensure safety (certification)...

a systematic process for verifying an architecture design and recommending acceptance/ SIL changes / architecture changes

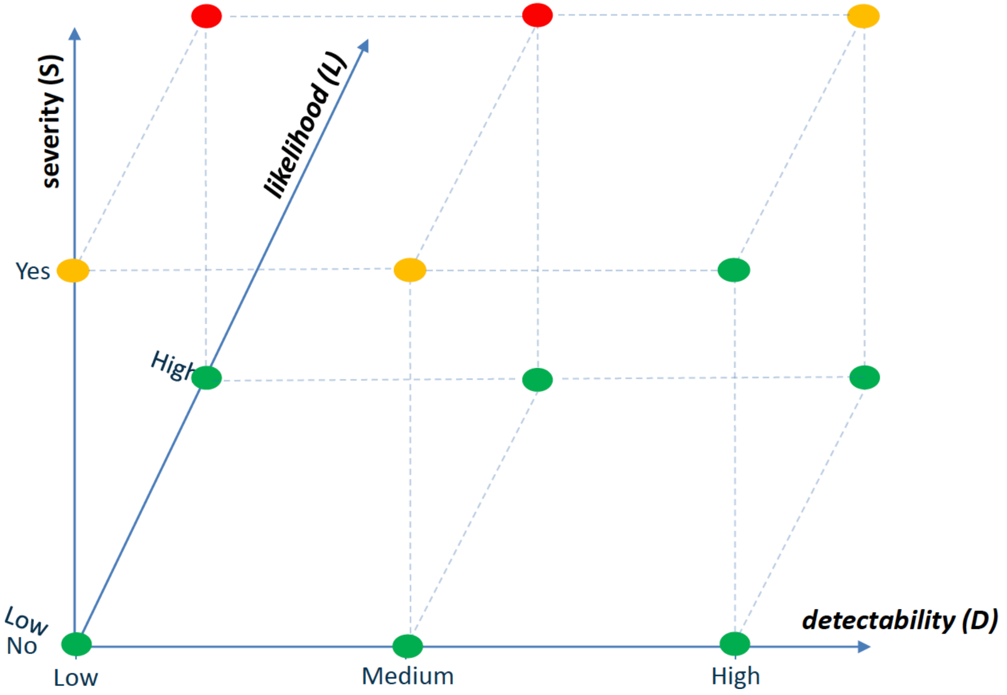

SW FMEA
safety concept includes requirements on independence, or SW modules in the architecture have different ASILs assigned?
No
Yes
Dependent Failure Analysis
Cascading Failure Analysis
Common Cause Failure Analysis
Different ASIL?
Yes
SW component coexistence analysis
No

methodology to ensure safety (certification)...

safety concept /architecture definition including systematic reference to ISO 26262 prescriptions/recommendation for error detection/handling

- FMEA as guided by ISO 26262 guidewords
- likelihood of events estimated (at least on coarse ordinal scale) with the help of product history, design analysis, complexity metrics, ...
- severity as two-level scale (severe or not)
- detectability : low, medium, high

- leading to 3-D matrix
- where state may be fine, "acceptable", to be improved



methodology to ensure safety (certification)...

- also in presentation: similar systematic, standard-driven method for software Dependent Failure Analysis (Cascading failures, Common Cause failures, Software coexistence analysis)

- preliminary feedback from use
  - having a "methodology" is useful
  - guidance in applying runtime safety mechanisms a plus
  - liked by customers' quality departments
  - problems with architecture design not always available at analysis time
  - some difficulty guiding users to preferred solution

  - Plans for future work to support integration with fault injection, partially automated analysis

some themes

- common to most of the workshop:
  - need for F/I Tolerance
  - size of challenges
  - no safety without security
  - economics, how much we are willing to spend

- especially highlighted in this session
  - systematic lifecycle for whole lifetime
  - standard-driven processes [might not be enough for sufficient confidence but]
    - + require assistance/ advice for reasonable application
    - + involve dealing with multiple uncertainties