McMaster University

Inspiring Innovation and Discovery

Functional Safety of Autonomous Vehicles Through Model Based Design & Assurance

Prof. Mark Lawford, Ph.D., P.Eng.
Director of McSCert

Thursday May 3, 2018

at

IOT Trends

IWDK

McSCert

McMaster Centre for Software Certification

McMaster University
Inspiring Innovation and Discovery

McSCert

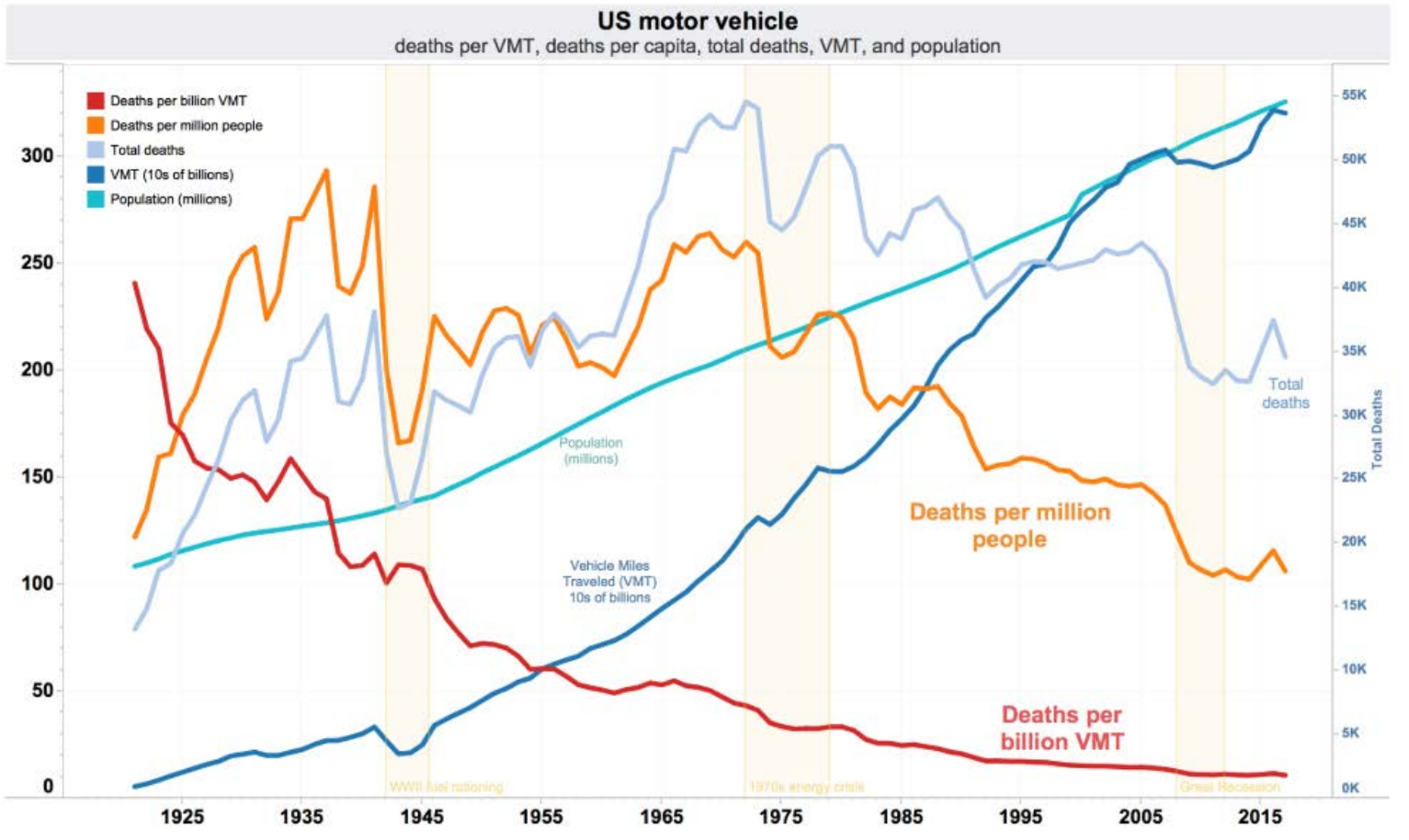
Outline

- Motivation
- Modeling the System & The Safety Assurance
- Determining impact of design changes to facilitate assurance reuse (& predict unsafe emergent behaviours)
  - Example: Power Sliding Door
- Developing better safety assurance architectures for ADAS
- Standardizing arguments through assurance case templates

McMaster University
Inspiring Innovation and Discovery

McSCert

Disclaimer: Vehicles are getting safer



Source: Wikipedia

- That doesn't mean we can't do better!

McMaster University
Inspiring Innovation and Discovery

McSCert

Motivation

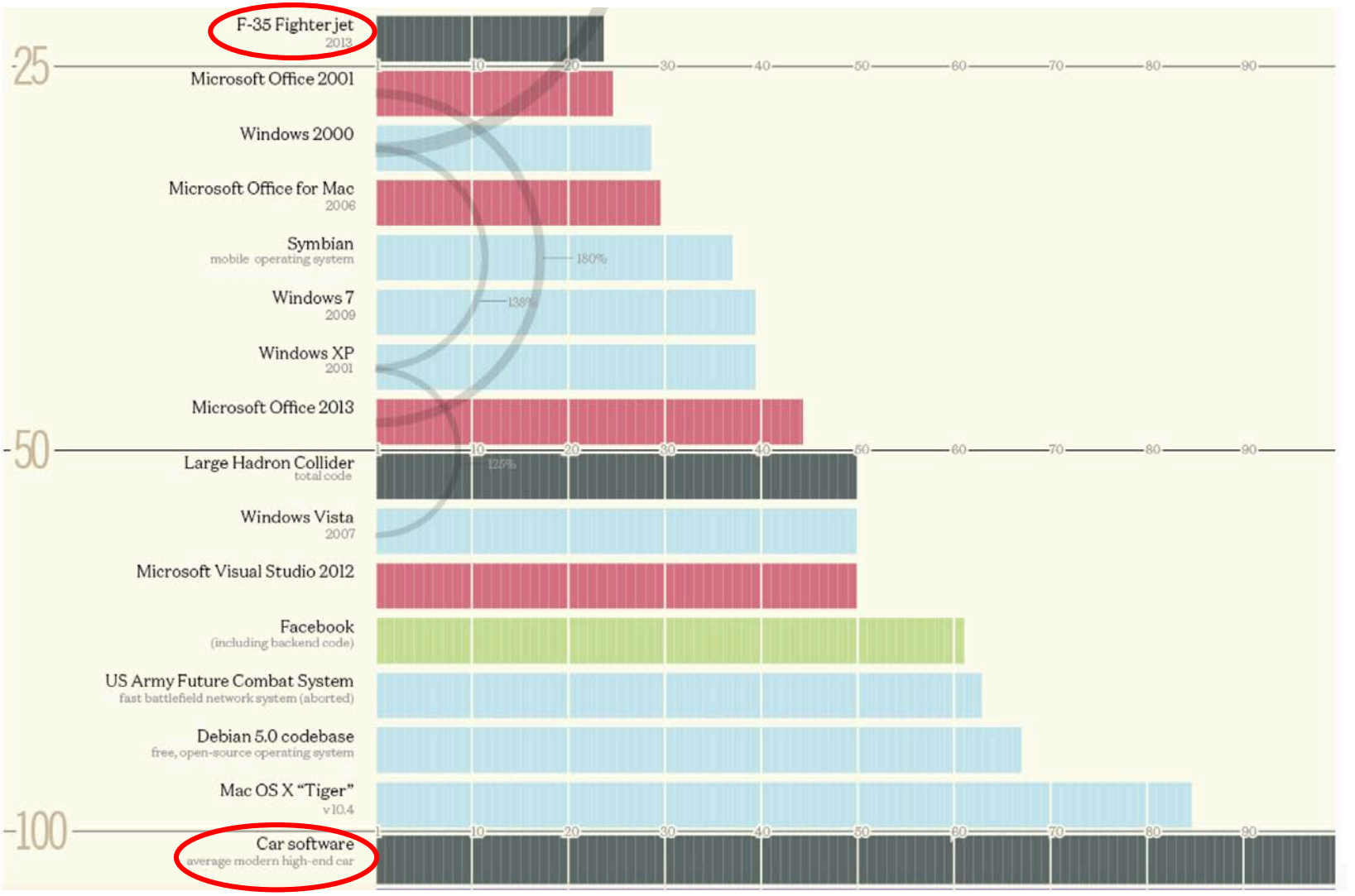
Uber Self-Driving Car Fatality Reveals the Technology's Blind Spots

The ride-sharing company has halted its autonomous vehicle testing while it investigates the accident in Arizona

McMaster University
Inspiring Innovation and Discovery

McSCert

Motivation: Automotive vs Other Industries



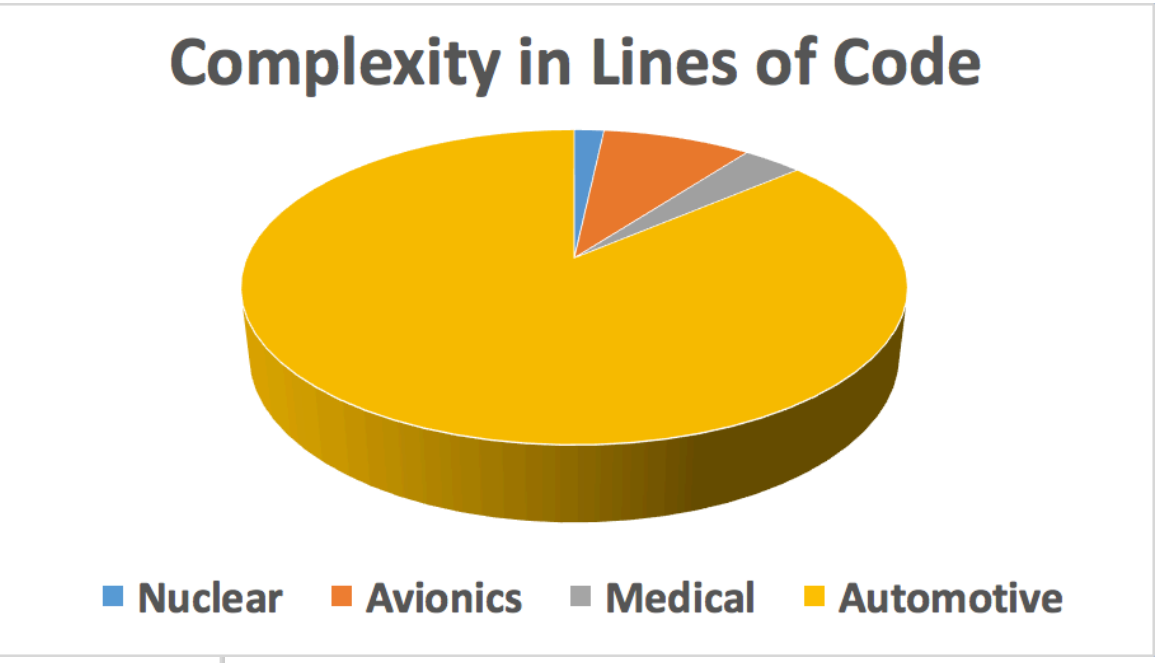
Source: Information is beautiful MLOC by product

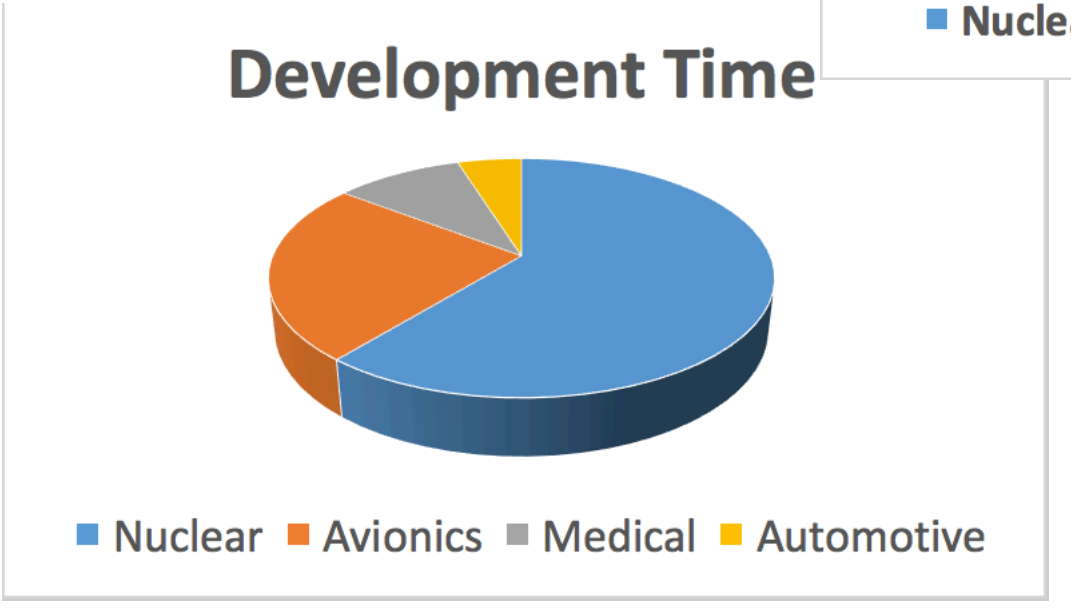
Motivation: Automotive Complexity and Time

McMaster University
Inspiring Innovation and Discovery

McSCert

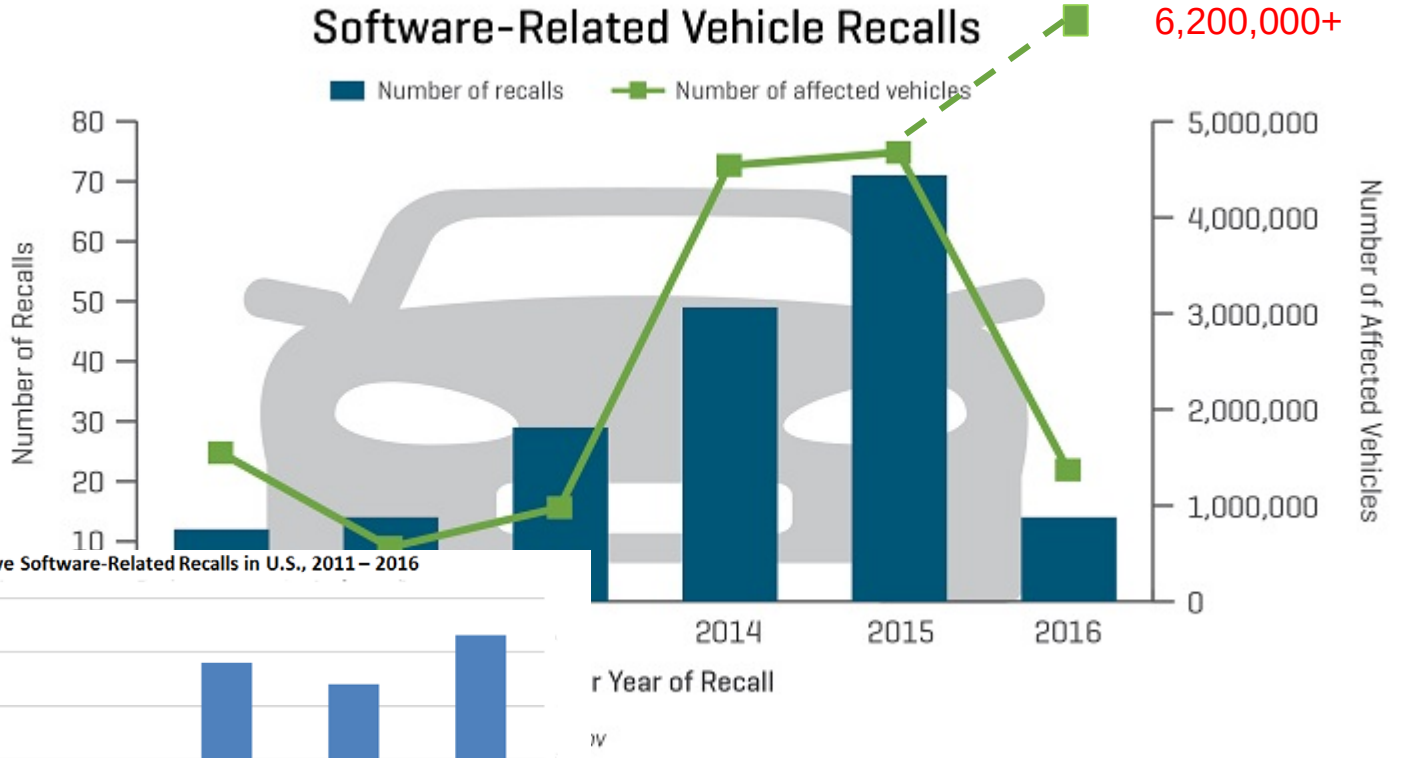
Clearly automotive has to do a lot more . . .

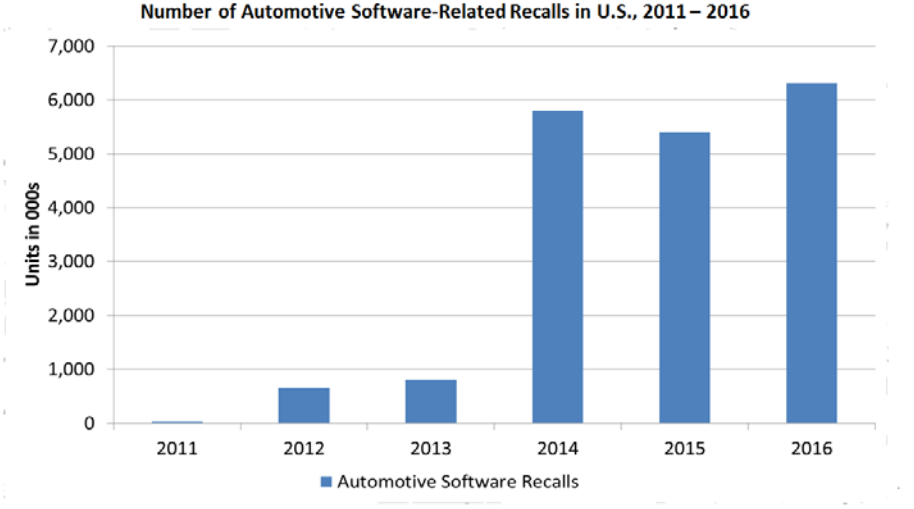




… in a lot less time!

Something needs to change





Year	SW related recalls
2011	5%
2015	15%

Source: *Software Is Eating the Auto Industry* by Roger Lanctot | Aug 25, 2017

Incremental development: The answer to the Complexity/Time Crunch?

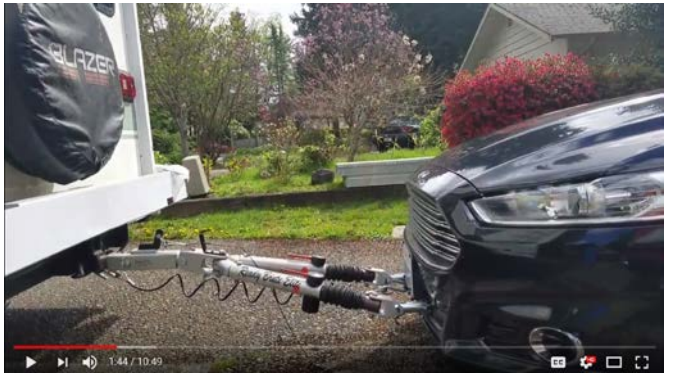
- Idea: Reuse existing safety assurance arguments for minor design changes (e.g. towing features)
- Sounds promising! What are the results?

A jury has awarded $3 million in the case of a fatal crash involving this Toyota Camry.

<u>Toyota Unintended Acceleration (UA)</u>

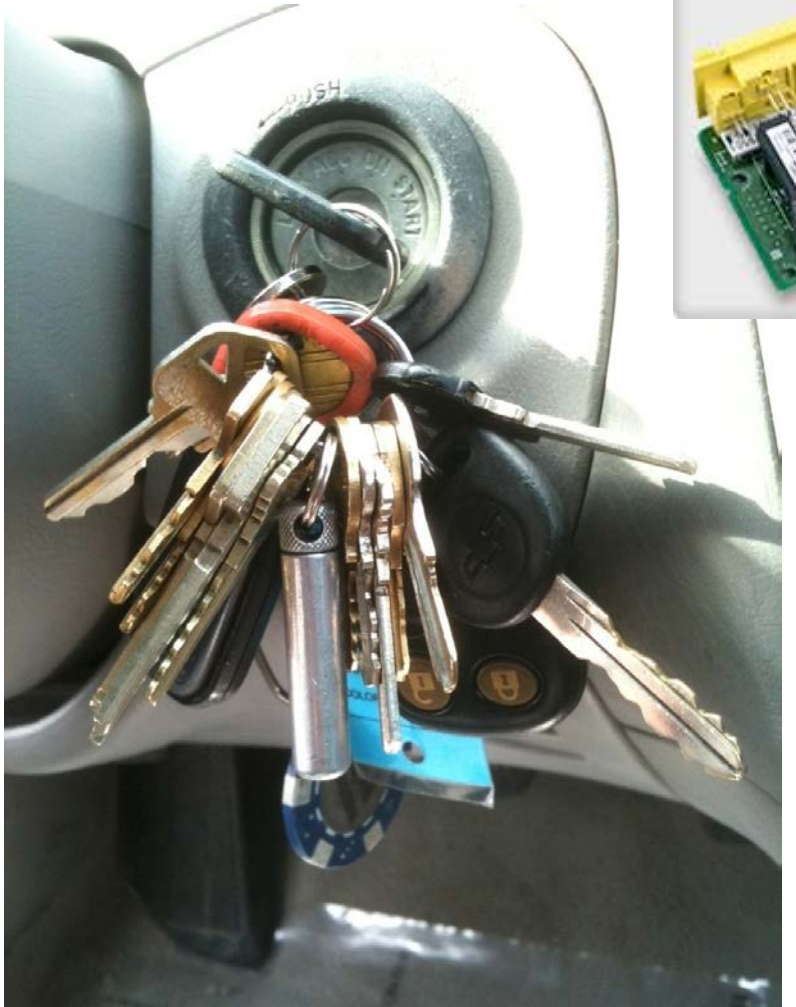
- Brake Echo Check failsafe system for UA only received “brake transistions”
- If your foot is already on the brake
  - And then a UA event occurs
  - You may have to completely take foot off the brake & reapply to trigger failsafe system!

<u>2015 Ford Fusion vehicles</u> equipped with a mechanical key and dual screen cluster, 30 minutes after the ignition is turned off, the Body Control Module (BCM) allows the key to be removed when the transmission is not in Park. Part 573 Safety Recall Report 14V-736



Even your keychain might be part of the problem!

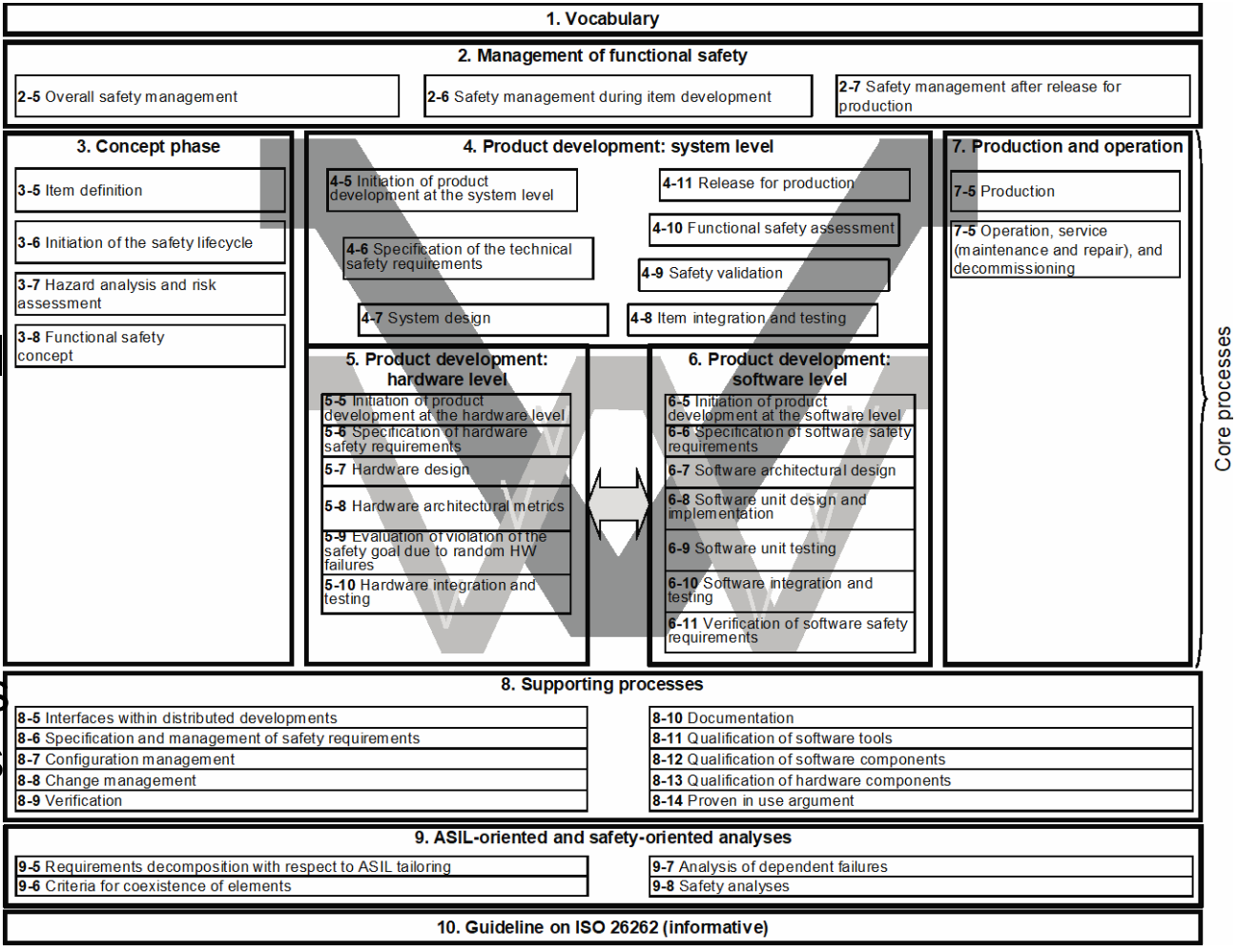
- A 1.6 mm difference in a $0.57 part!



ISO 26262

ISO 26262 (Road vehicles- Functional safety):

- is the de facto standard for functional safety of automotive vehicles
- It deals with the electrical and electronic components of automotive vehicles - including software
- It consists of 10 parts



- It covers planning, development, maintenance, operation & decommissioning of software & electronics in automotive vehicles

McMaster University

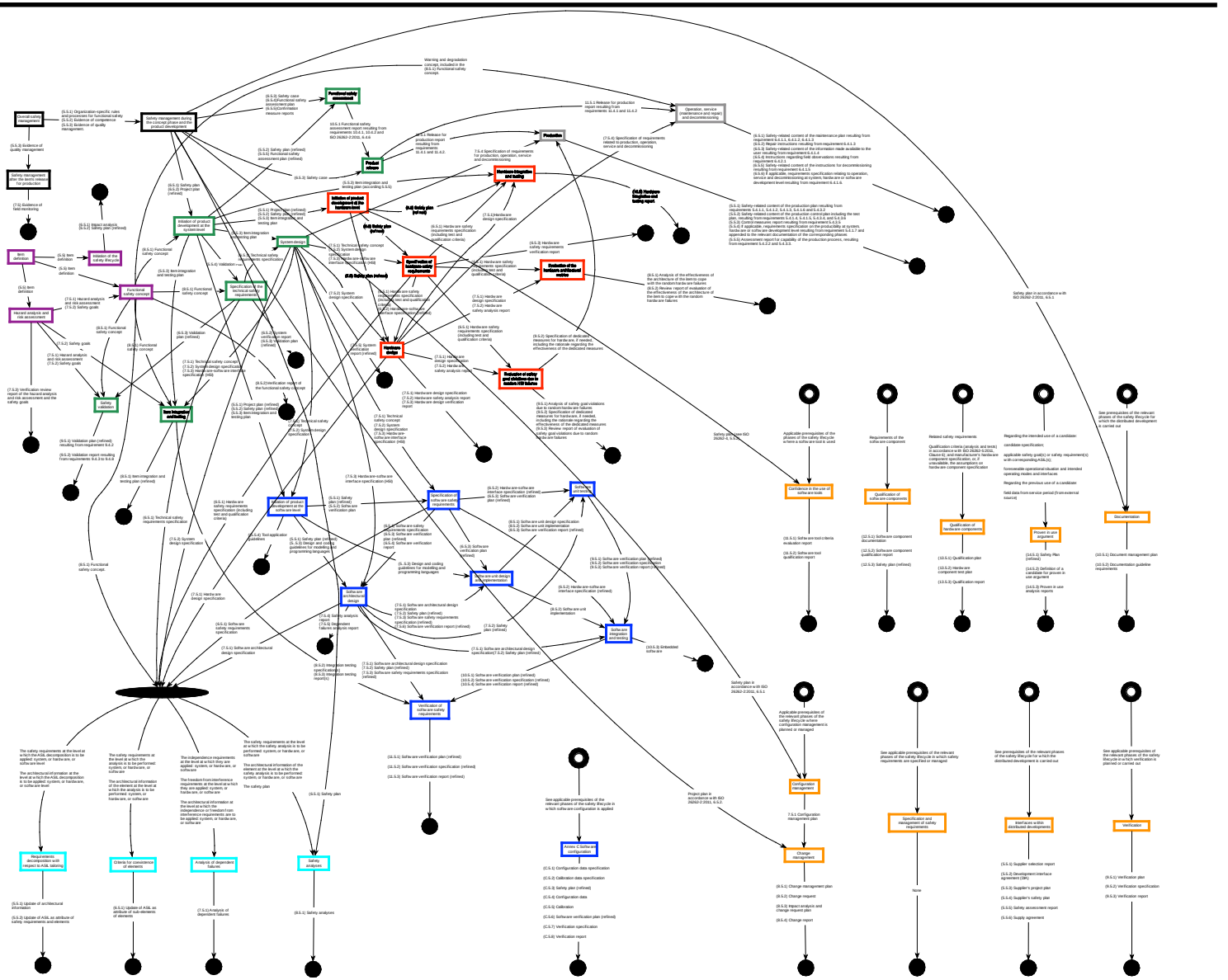
Inspiring Innovation and Discovery

McSCert

ISO 26262 is Complex

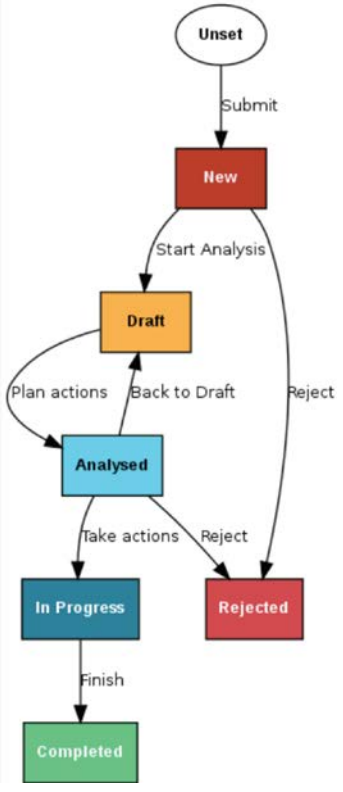
This represents the workflow for ISO 26262

Existing commercial tools help companies comply with the standard



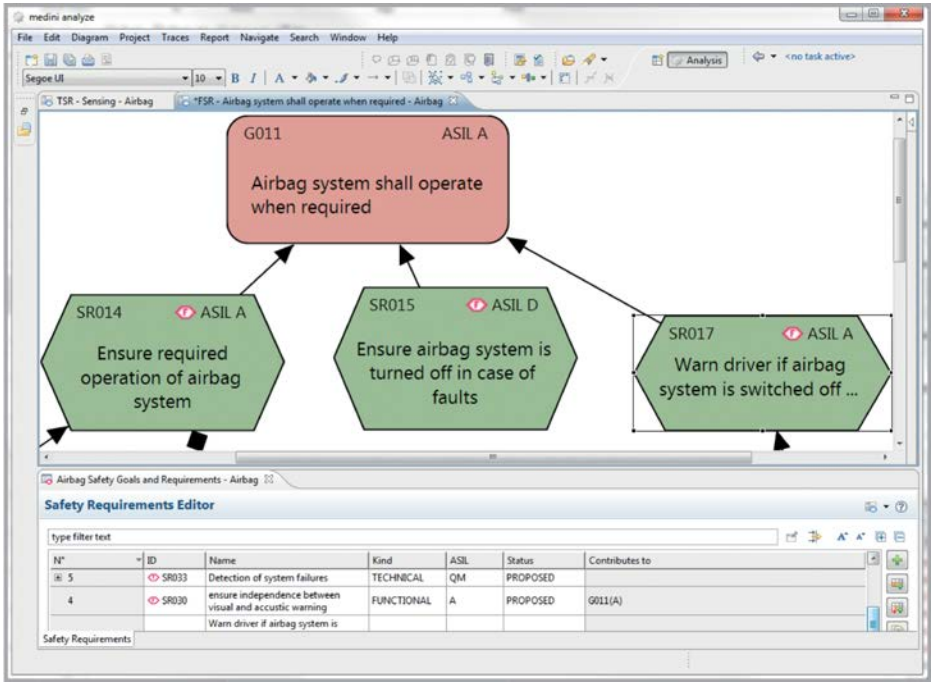
McMaster University
Inspiring Innovation and Discovery

Current State of Practice



codeBeamer
ISO 26262
template
workflow

- Existing Application Lifecycle Management (ALM) and compliance tools such as Intland's codeBeamer and *medini Analyze*
- Process and document driven system and safety assurance
- Help ensure process workflows followed and provide traceability
- Provide some automation but significant manual rework required using Excel spreadsheets and Word documents
- *Little support for incremental safety analysis*
- *No explicit model of the product safety case*



medini Analyze
ISO 26262 ASIL
assignment

McMaster University
Inspiring Innovation and Discovery

SACM 2.0 Standard

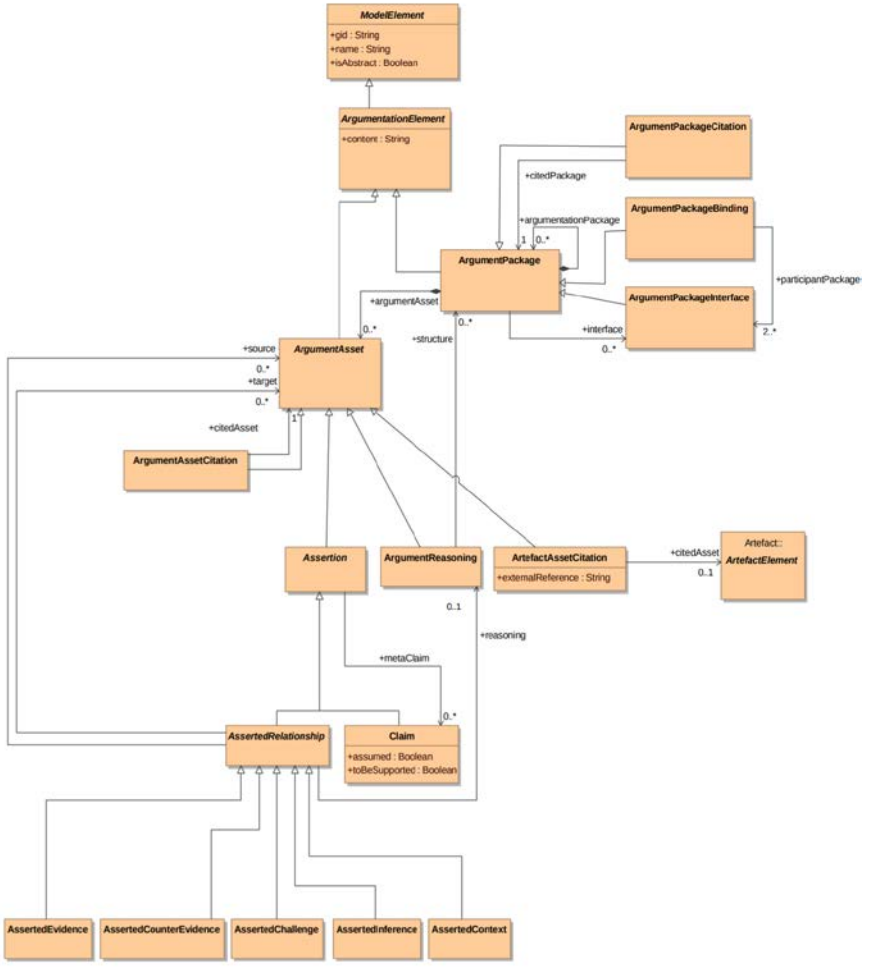


Figure 11.1 – Argumentation Class Diagram

- OMG's Structured Assurance Case Metamodel (SACM) provides a representation of general Assurance Case (AC) and "templates" that are argument patterns
- Will use it to create ACs
- SACM does not provide ISO 26262 specific AC template
- SACM does not provide way of performing incremental assurance using MM or cover product lines

McMaster University
Inspiring Innovation and Discovery

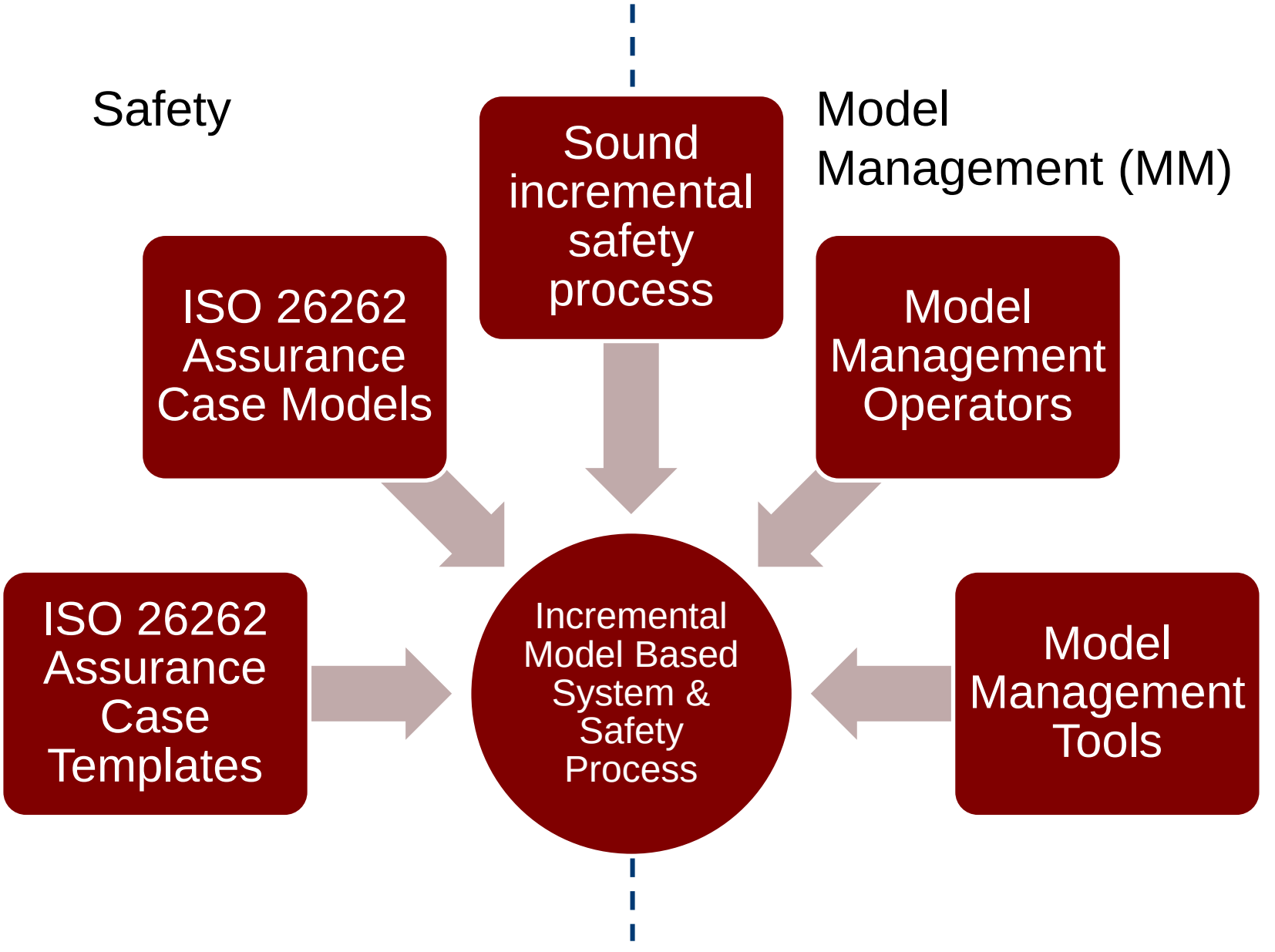
McSCert

High Level Objectives

Support System and Safety Evolution

- … correctly
- … quickly (via scalable automated tool support)
- … while facilitating product-level and product-line reuse

Connecting 2 Research Areas to Address the Problem



McMaster University
Inspiring Innovation and Discovery

McSCert

Objective: Create explicit AC template for ISO 26262

- Current status
  - Principles for the Systematic Development of an Assurance Case Template from ISO 26262
    - ISSRE 2017 paper

McMaster University
Inspiring Innovation and Discovery

McSCert

Current Research: Standards & Assurance Cases

- Researchers have examined the implicit/inherent assurance case embodied in standards
  - 2005 – Ankrum, Kromholz – CC, DO-178B, ISO 14971 [6]
  - 2011 – Bender – CSA N290.14
  - 2013, 2015 – Holloway – DO 178C [7], [8]
  - 2013 – Birch, et al – ISO 26262 [9]
  - 2014 – Hocking, Knight, Aiello, Shiraishi – ISO 26262 [10]
- Recent research related to ISO 26262
  - Gallina, et al – 2013,14 (17) [11], [12]
  - Chowdhry, Wassyng et al – 2017
- Purpose of making the implicit assurance case explicit is to
  - Better understand the standard
  - Identify gaps in the standard

McMaster University
Inspiring Innovation and Discovery

McSCert

Assurance Cases

General introduction and discussion on assurance cases can be found in references [1], [2], [3]

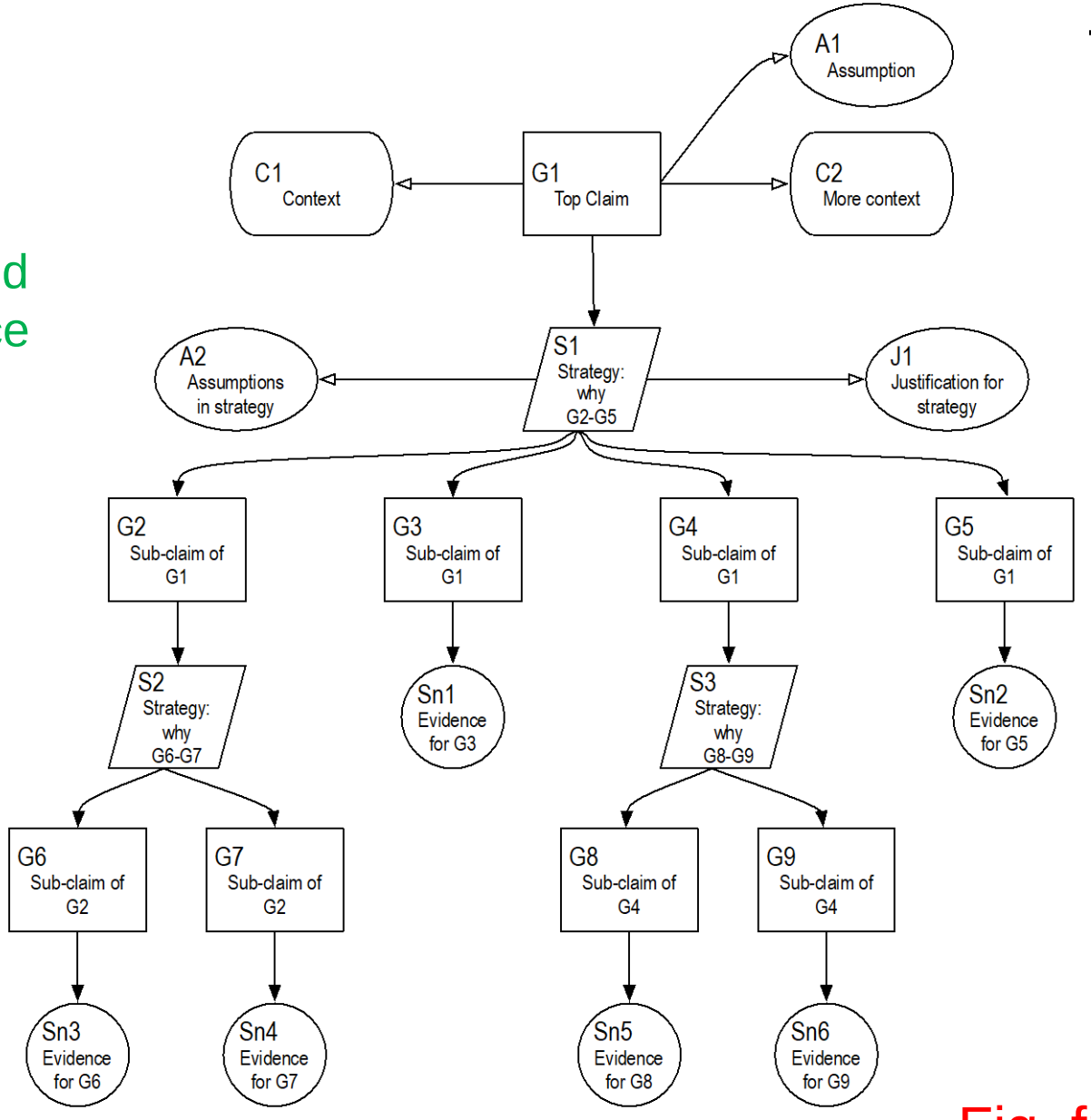
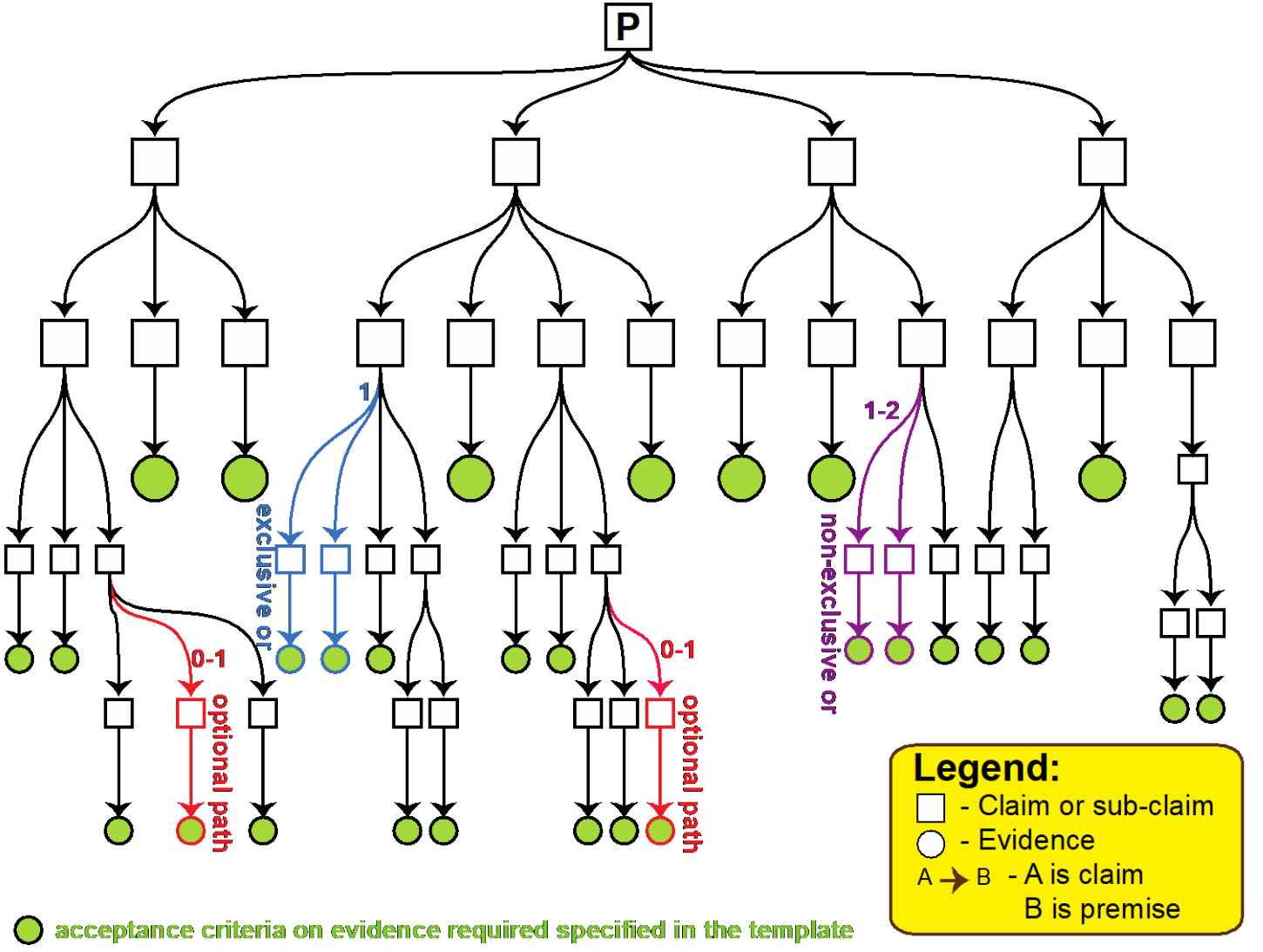


Fig. from [5]

Assurance Case Templates [5]

- Complete assurance case for a product line



McSCert

- Complete assurance case produced before development starts
- Optional argument paths
- Evidence nodes specify type of evidence required and acceptance criteria on that evidence
- Requires explicit reasoning (not shown here)
- Try to make it robust with respect to change
- Assume it will be developed by a community in the same way that standards are
- Could replace traditional standards

McMaster University
Inspiring Innovation and Discovery

McSCert

Modeling ISO 26262

To understand *ISO 26262* in sufficient detail, we will construct:

- a model of consolidated work products
- a data flow model of *ISO 26262* processes
- a conceptual model

Consolidated Work Products – ISO 26262

2-5.5.1 Organization-specific rules and processes for functional safety, resulting from 5.4.2 and 5.4.5.

2-5.5.2 Evidence of competence, resulting from 5.4.3.

2-5.5.3 Evidence of quality management, resulting from 5.4.4.

2-6.5.3 Safety case, resulting from 6.4.6.

2-6.5.5 Confirmation measure reports, resulting from 6.4.7 to 6.4.9.

2-7.5.1 Evidence of field monitoring, resulting from 7.4.2.4.

3-5.5 Item definition resulting from the requirements of 5.4.

3-6.5.1 Impact analysis resulting from the requirements of 6.4.2.1 to 6.4.2.4.

3-7.5.1 Hazard analysis and risk assessment resulting from the requirements of 7.4.1.1 to 7.4.4.2

3-7.5.2 Safety goals resulting from the requirements of 7.4.4.3 to 7.4.4.6

3-7.5.3 Verification review report of the hazard analysis and risk assessment and the safety goals resulting from the requirement of 7.4.5.

3-8.5.1 Functional safety concept resulting from the requirements of 8.4.1 to 8.4.4.

3-8.5.2 Verification report of the functional safety concept resulting from the requirements of 8.4.5.

2-6.5.2 Project plan (refined), resulting from 6.4.3.4. (Original is "external?")
4-5.5.1 Project plan (refined) resulting from requirement 5.4.4.

2-6.5.4 Functional safety assessment plan, resulting from 6.4.9.
4-5.5.5 Functional safety assessment plan (refined) resulting from requirement 5.4.3.

4-6.5.1 Technical safety requirements specification resulting from requirements 6.4.1 to 6.4.5.

4-7.5.1 Technical safety concept resulting from requirements 7.4.1 and 7.4.5.

4-7.5.2 System design specification resulting from requirements 7.4.1 to 7.4.5.

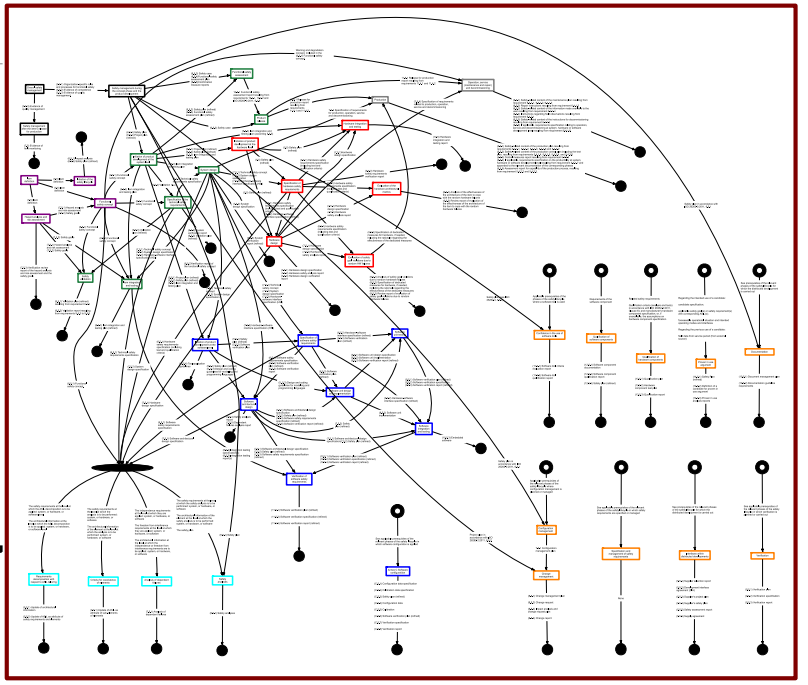
4-7.5.4 Specification of requirements for production, operation, service and decommissioning resulting from requirements 7.4.7.

4-6.5.2 System verification report resulting from requirement 6.4.6.
4-7.5.5 System verification report (refined) resulting from requirement 7.4.8.

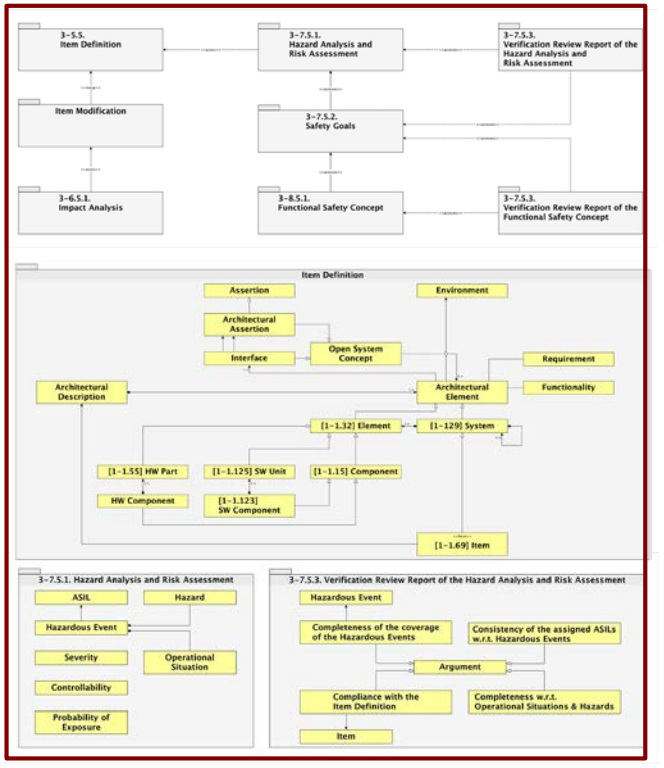
4-7.5.6 Safety analysis reports resulting from requirement 7.4.3.

4-8.5.3 Item integration and testing plan resulting from requirement 5.4.1.
4-8.5.1 Item integration and testing plan (refined) resulting from requirement 8.4.1.

4-8.5.2 Integration testing specification(s) resulting from requirements 8.4.1.

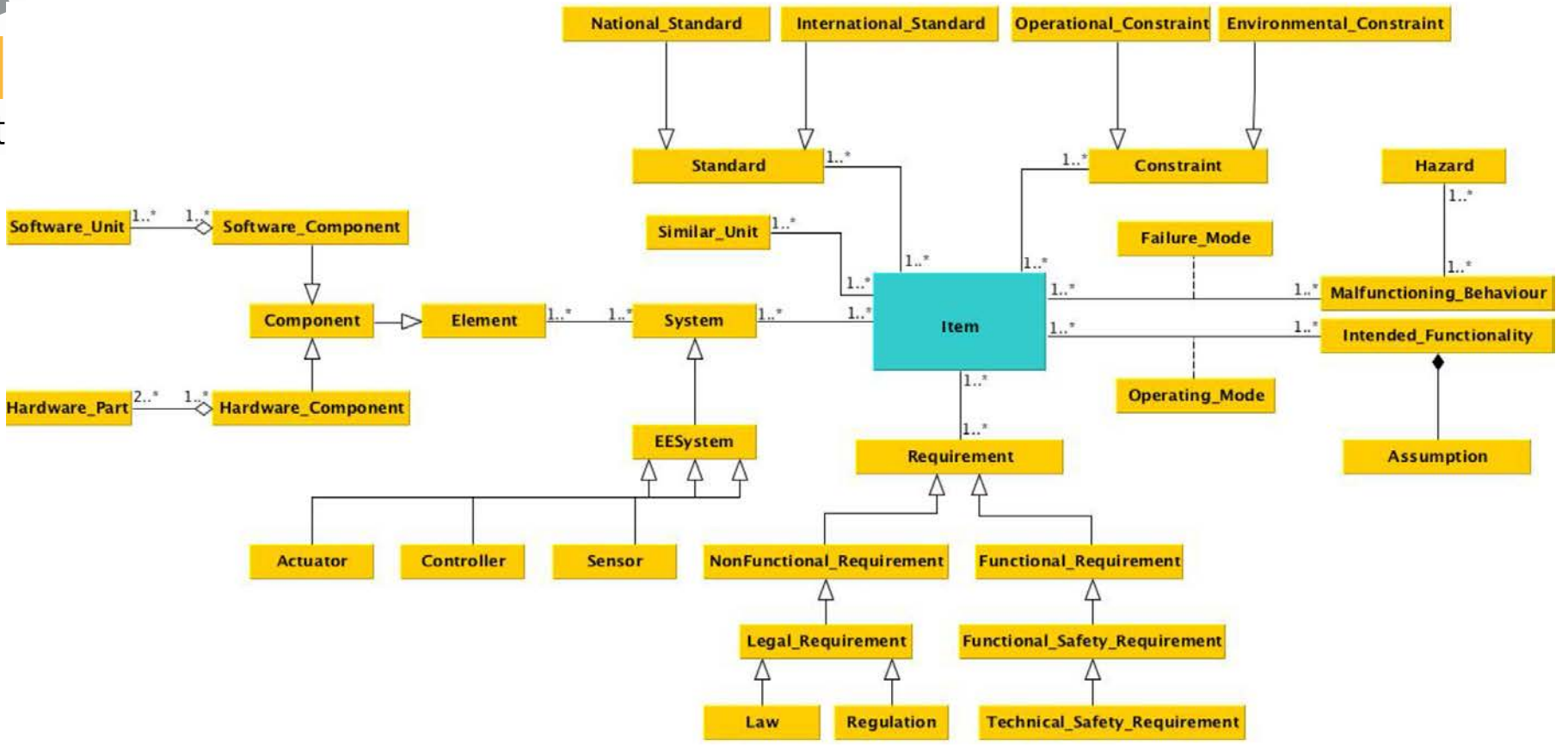


Data Flow



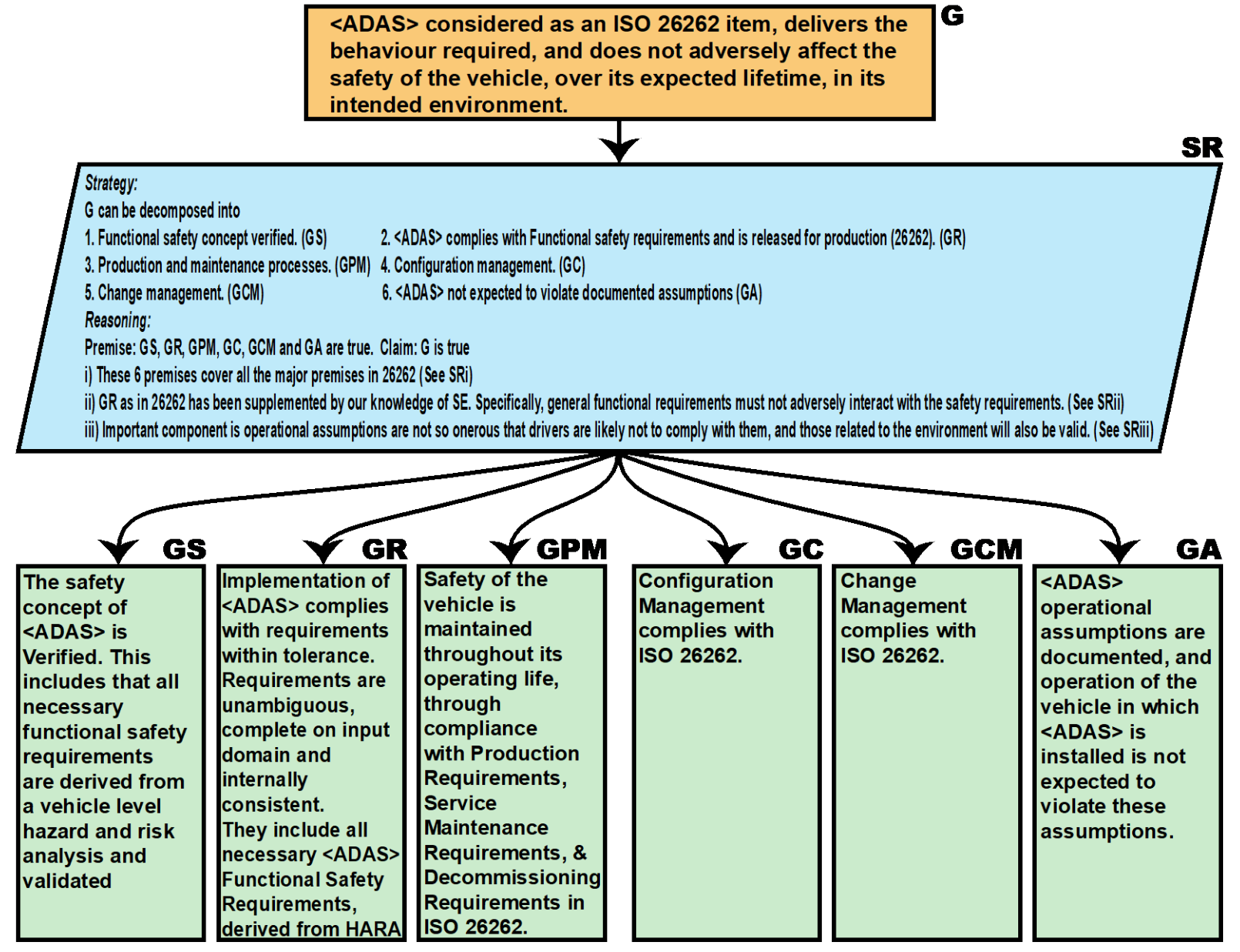
Conceptual Model

ISO 26262 Conceptual Model



- Conceptual model of an *Item* can automate checking of completeness and consistency [16]
- Use OCL to specify additional requirements from ISO 26262 such as Automotive Safety Integrity Level (ASIL) decomposition rules

ISO 26262 Assurance Case Template for ADAS [15]



McMaster University
Inspiring Innovation and Discovery

McSCert

Objective: Use template as a basis to create explicit AC models for products

- Current status
  - Created partial ISO 26262 Template for ADAS
  - Instantiated template for ACC variants
  - Can reuse most of the template for any other automotive system

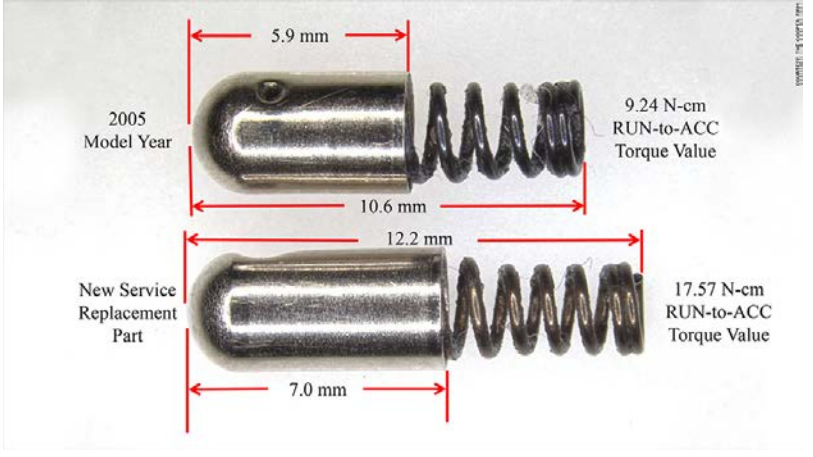
McMaster University
Inspiring Innovation and Discovery

McSCert

Challenges (research and technical)

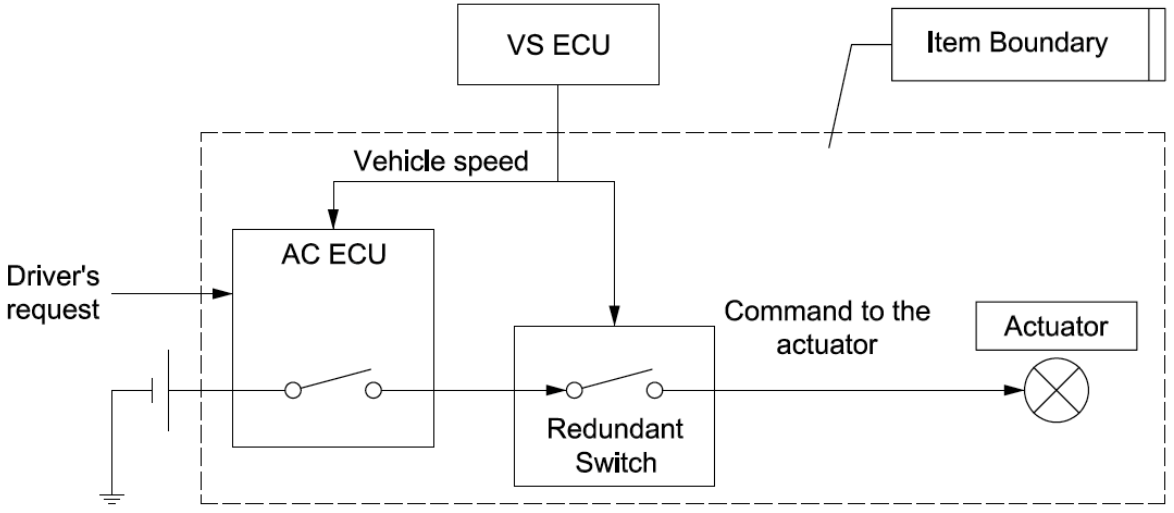
- Correctly specifying the complete ISO 26262 implicit assurance case
- Creating a complete correct conceptual model for ISO 26262 & validating model
- Creating a complete correct conceptual model for an OEMs safety standard
- Checking compliance of OEM's standard model with ISO 26262 model
- Creating explicit assurance cases models from OEM's existing assurance documents

Ignition Switch (keychain) Recall Revisited



- Explicit Safety Assurance Case helps system understanding
- Provides basis for evaluating changes

Example: Power Sliding Door System

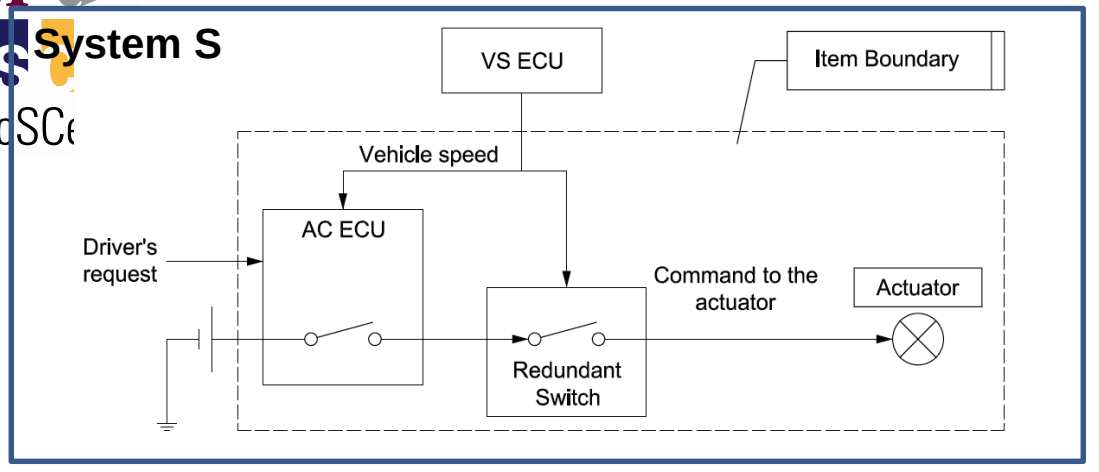


Safety goal SG1
Avoid activating the actuator when vehicle speed > 15 kph

McMaster University
Inspiring Innovation and Discovery

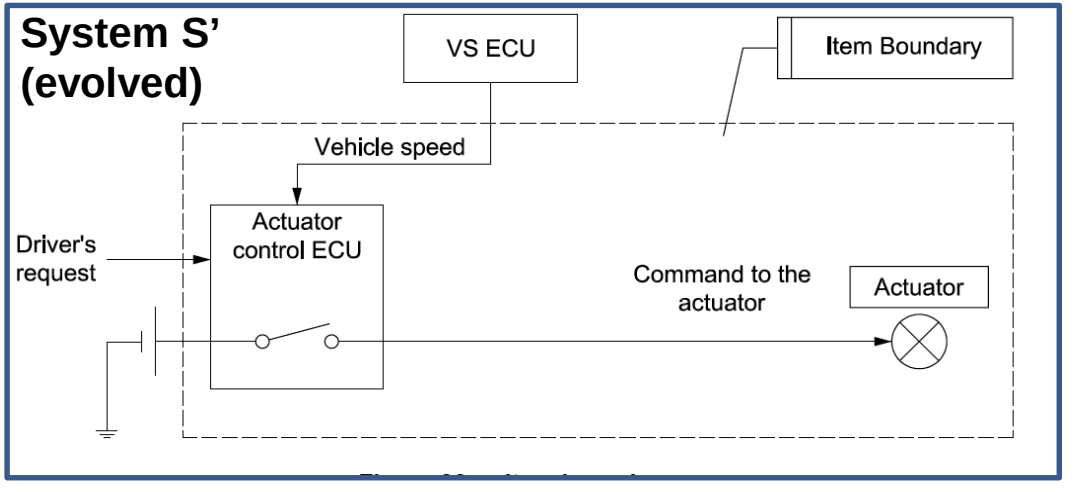
Power Sliding Door Safety Case

System change: Removing Redundant Switch in Power Sliding Door (PSD)



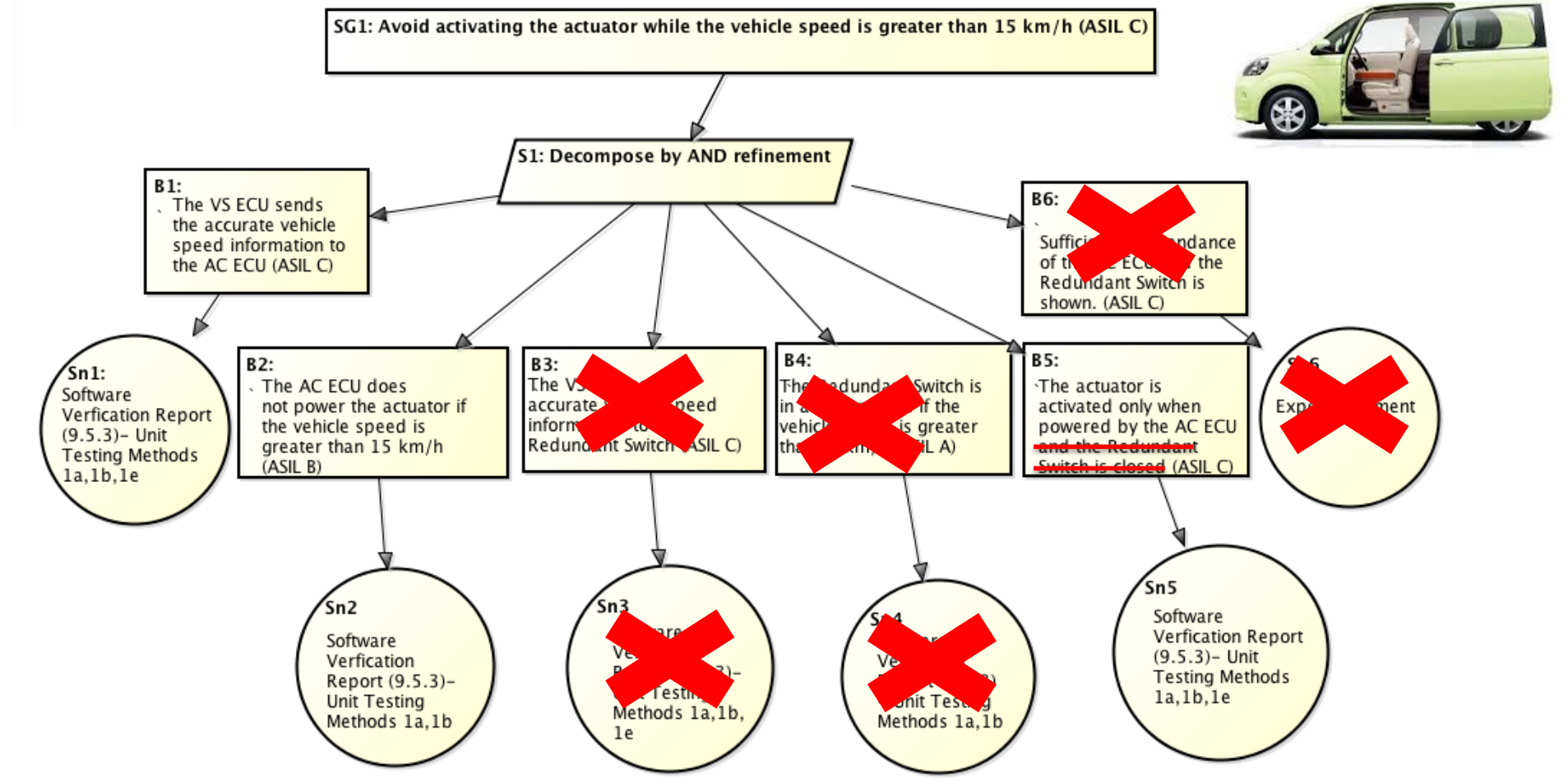
Safety Goal SG1:
Avoid activating the actuator when vehicle speed > 15 kph

Δ: removal of redundant switch



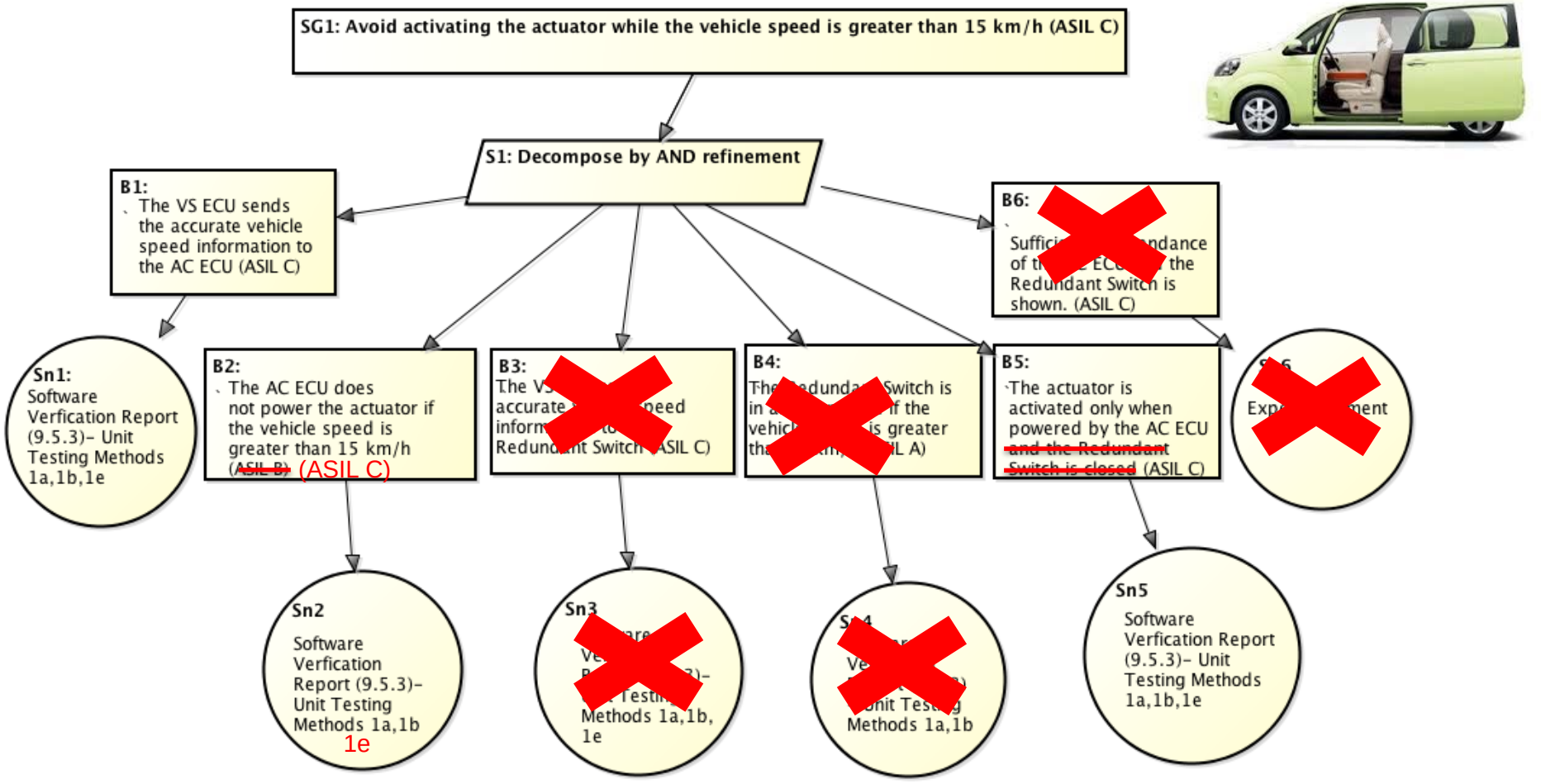
Safety goal remains the same.
How you achieve it & why you believe it changes.

Naïve Evolution of Safety Case



Naïve evolution approach: Delete everything related to switch

Safety informed evolution of safety case (after review and refinement by engineers)



Removing redundant switch changes B2 from ASIL B to ASIL C

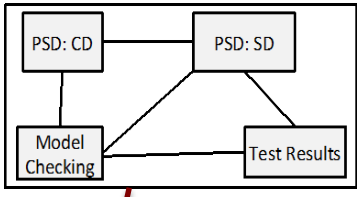
… so acceptance criteria for evidence Sn2 changes to add 1e

Solution: Model Based Impact Assessment

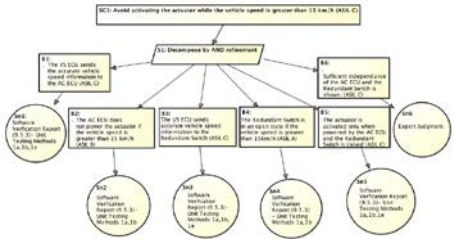
McMaster University
Inspiring Innovation and Discovery

McSCert

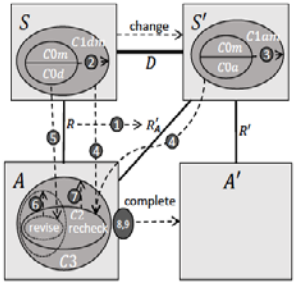
System Megamodel



Traceability Relation

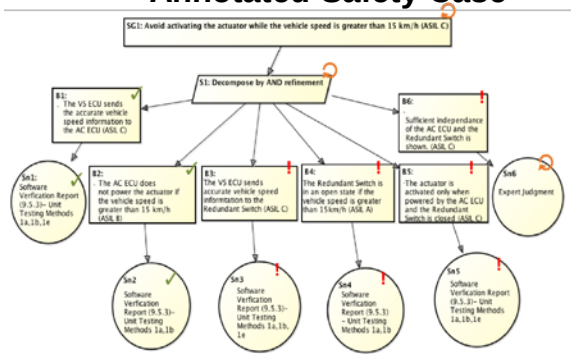


Model-Based Impact Assessment Algorithm



Human-in-the-loop refinement

Annotated Safety Case



✓ reuse ↻ recheck ! revise

Delta (change)

Model and Safety Case Slicers

[MODELS'16]: original approach

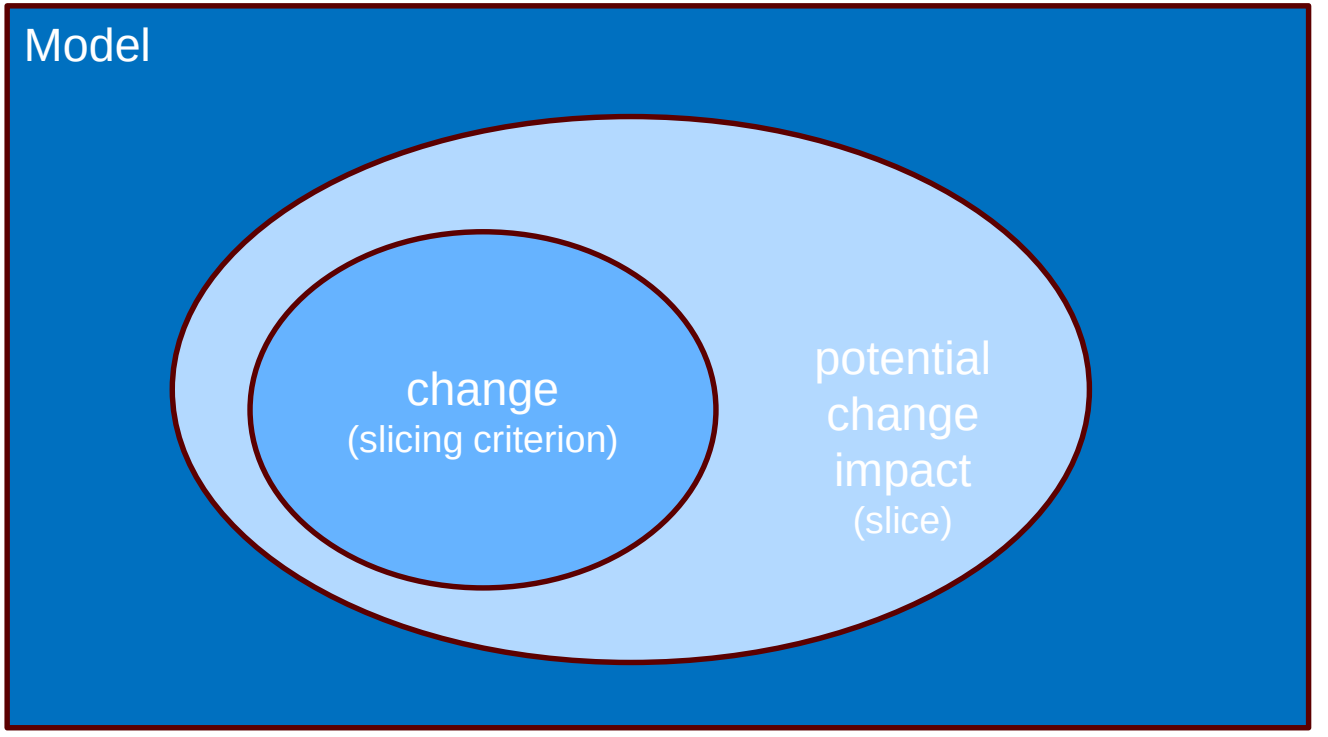
[SafeComp'17]: improved approach, safety case slicer, cost-savings analysis [17]

McMaster University
Inspiring Innovation and Discovery

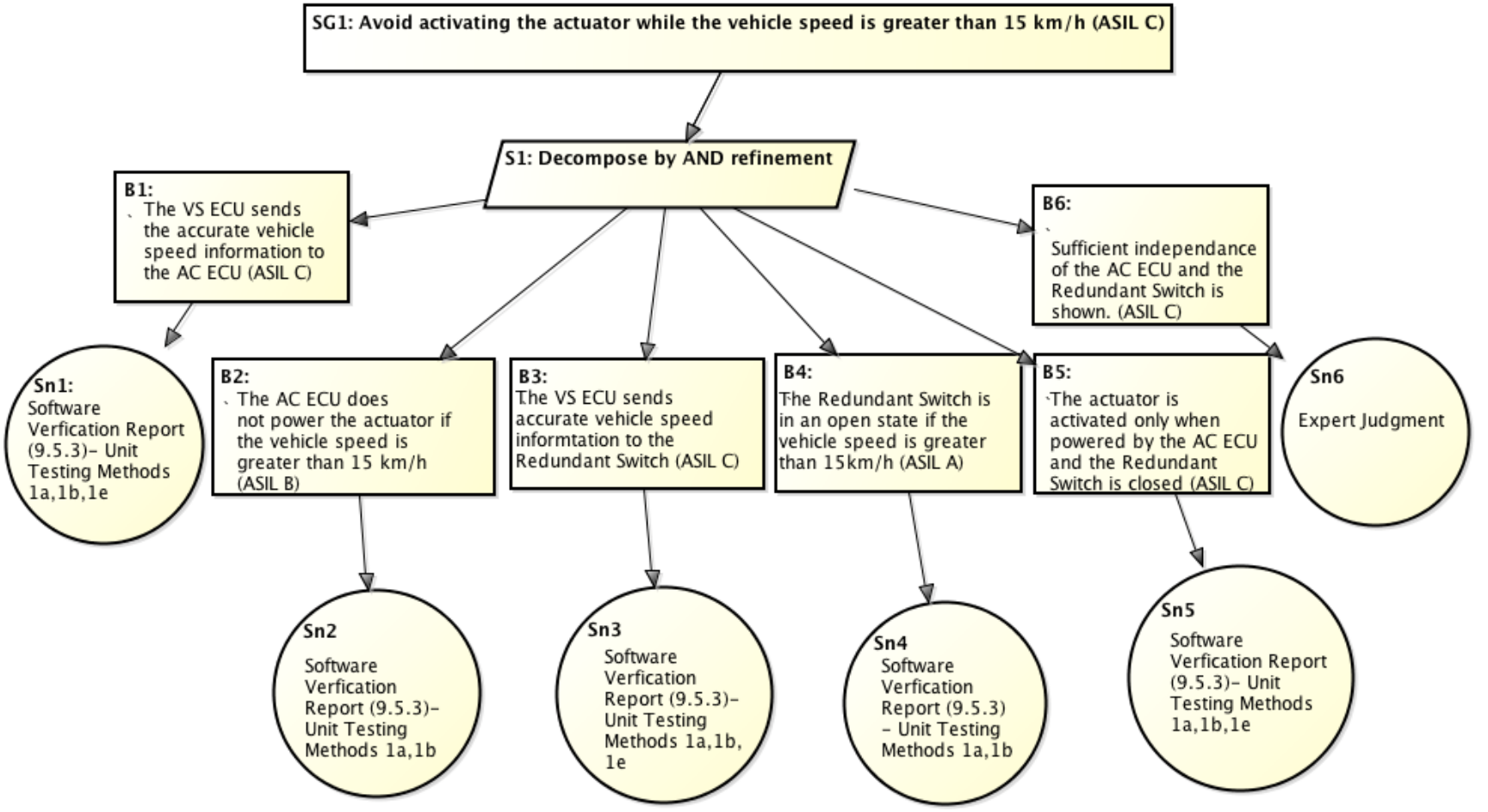
McSCert

Model Slicing

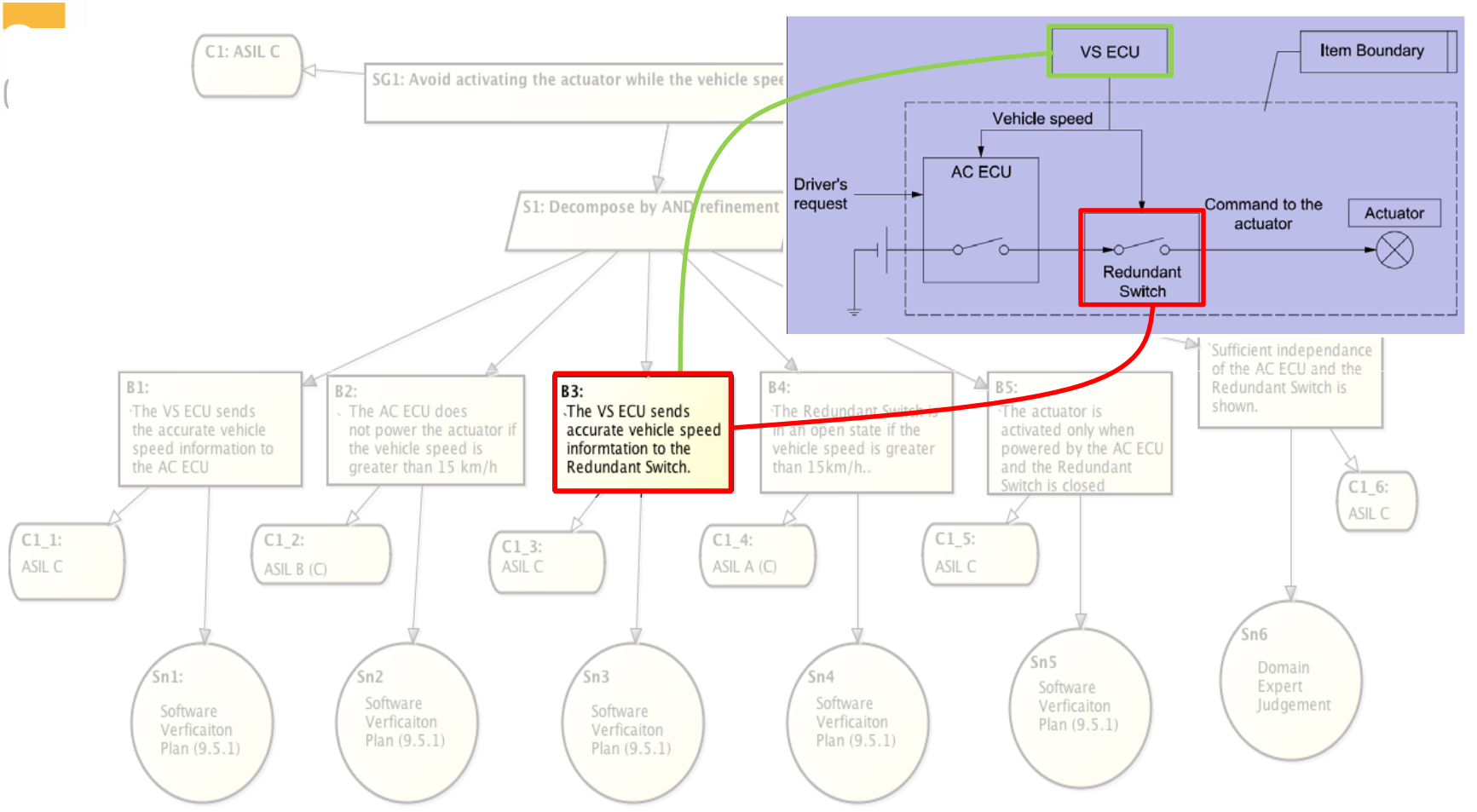
- Model slicing to identify change impact is a key technique for supporting model evolution



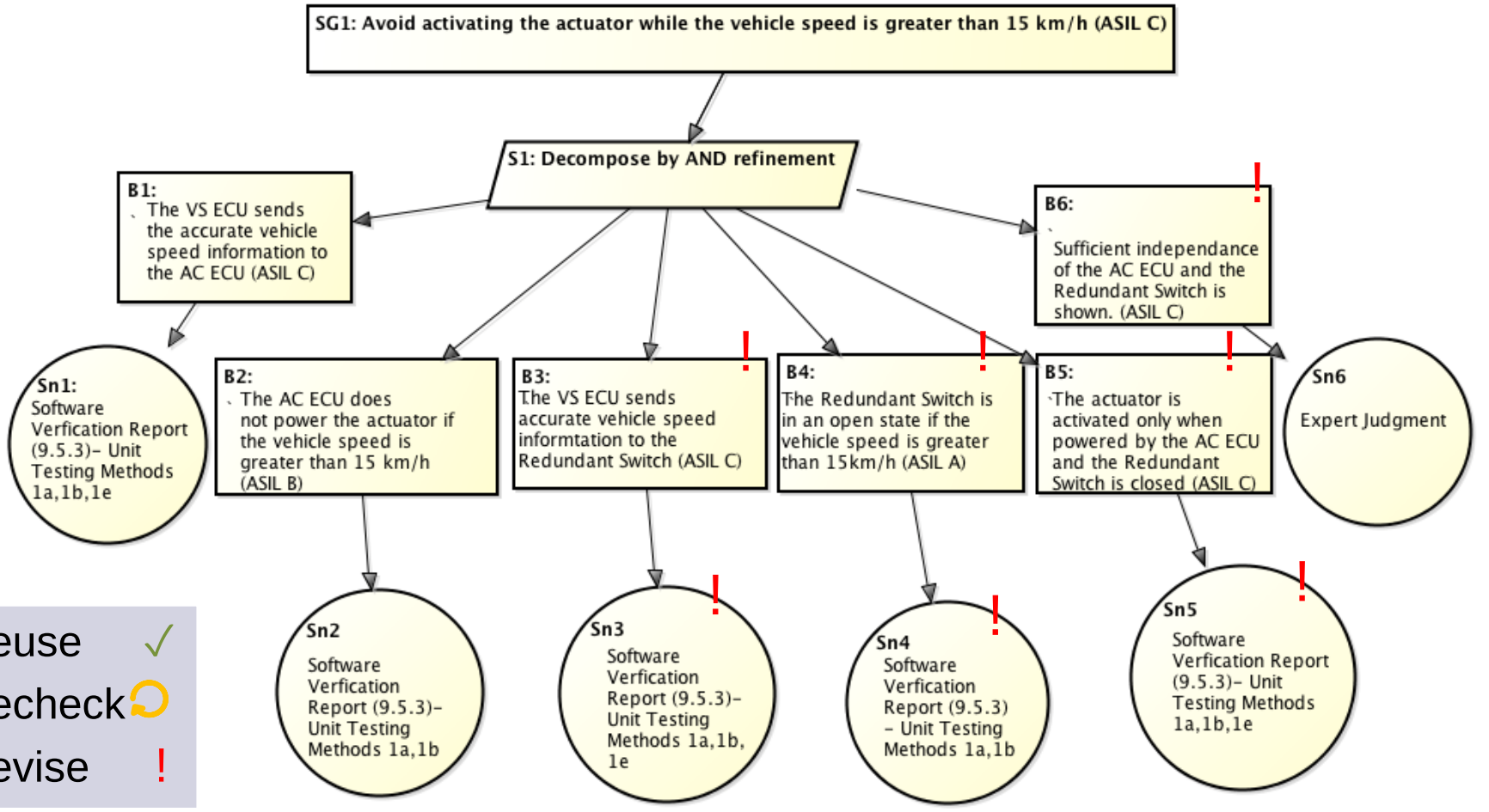
Begin with original safety case



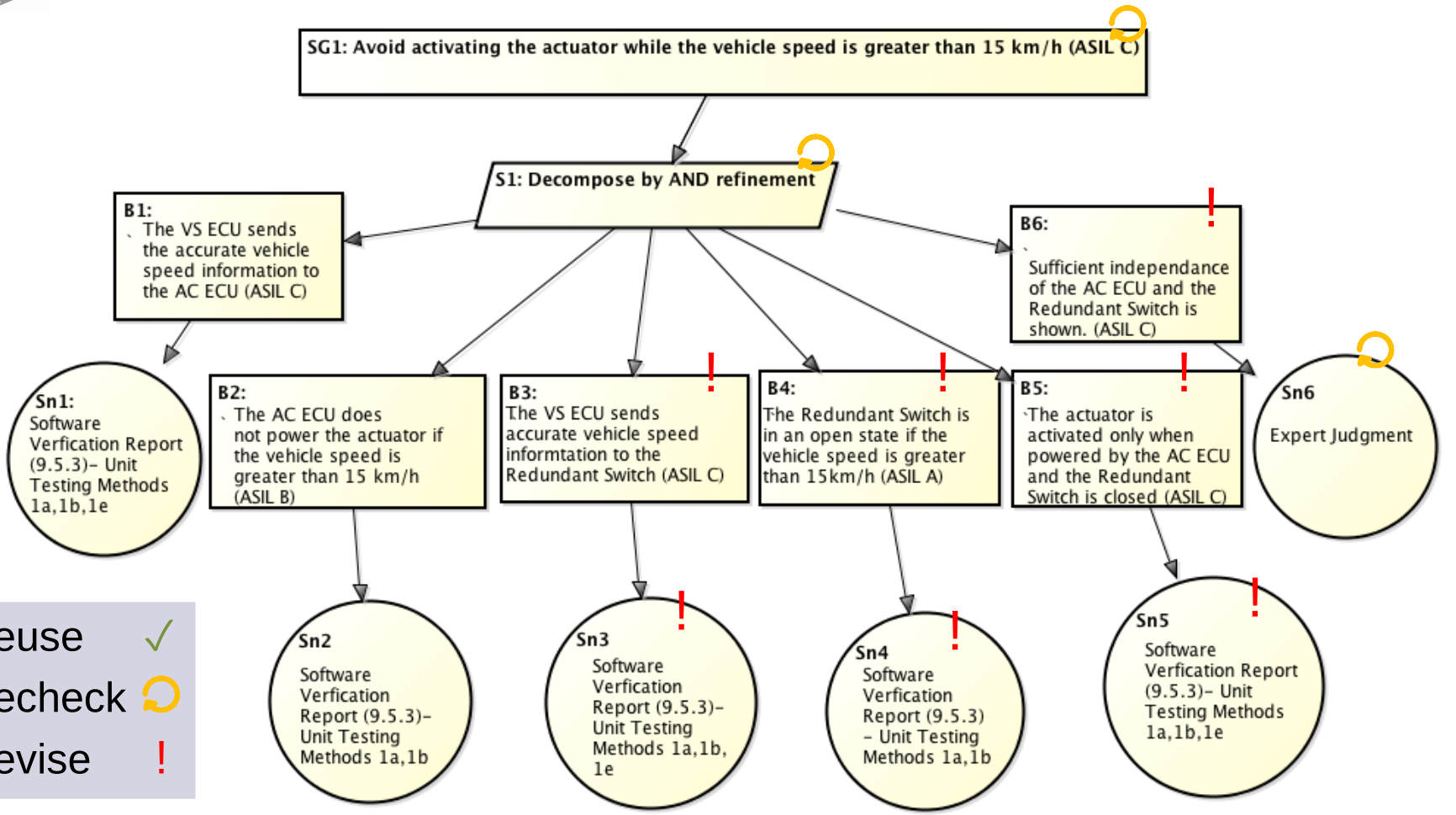
Based on traceability between system and safety case…



Mark elements directly related to "redundant switch" for revision



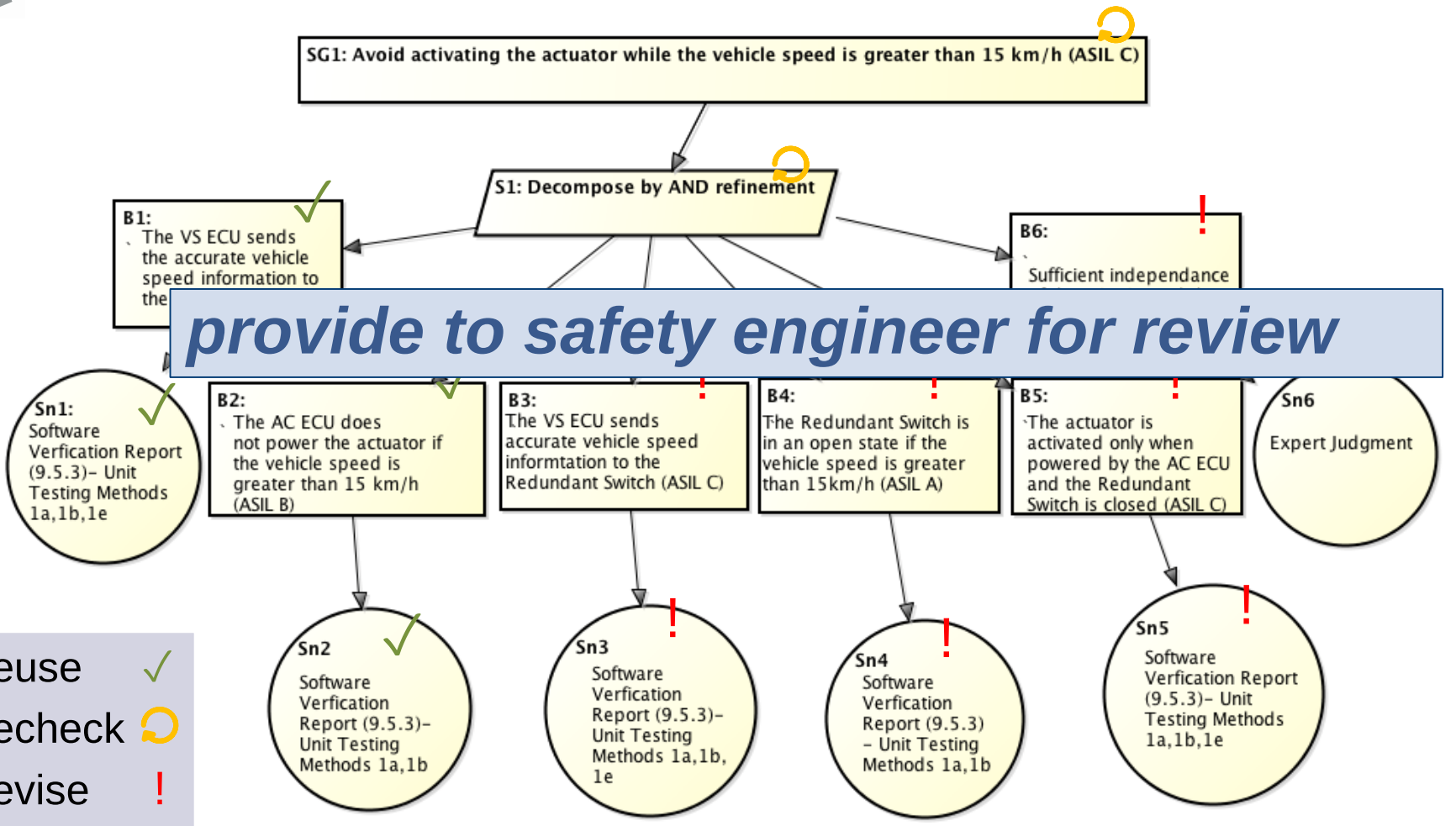
Use GSN slicers used to propagate annotations



McMaster University
Inspiring Innovation and Discovery

McSCer

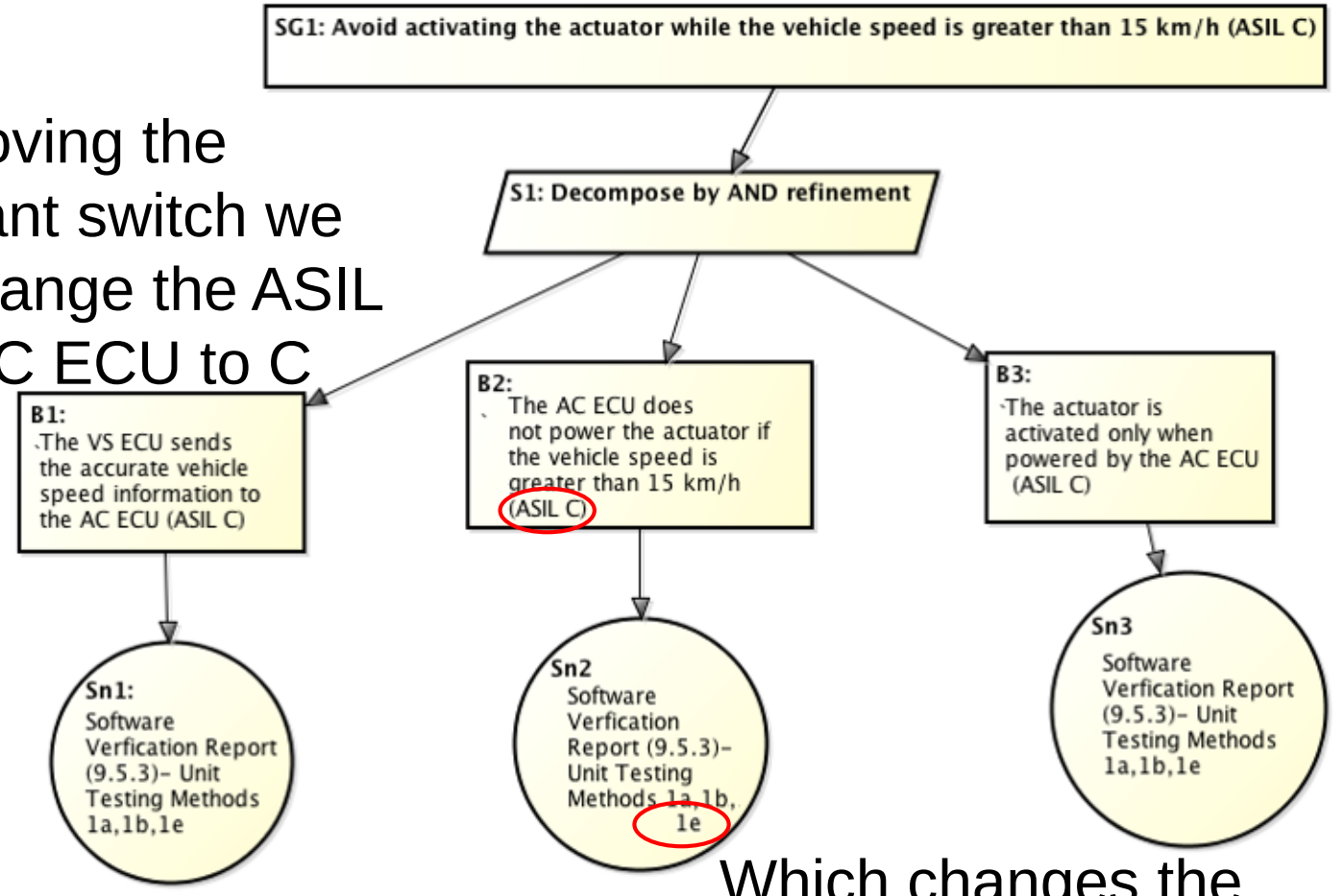
Mark everything left for reuse



Reuse ✓
Recheck ↻
Revise !

Safety informed evolution of safety case (after review and refinement by engineers)

By removing the redundant switch we have change the ASIL of the AC ECU to C



Which changes the acceptance criteria for the evidence

McMaster University
Inspiring Innovation and Discovery

McSCert

Impact Assessment of Simulink Models

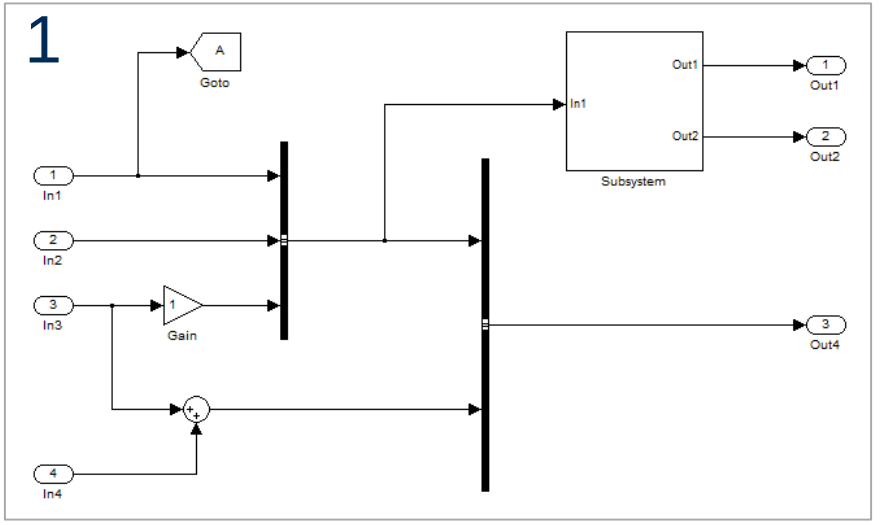
- Automotive controls software is overwhelmingly developed in Matlab/Simulink
- In order to understand impact of design changes on system safety, must be able to understand data and control flow of Simulink models
- Eventually might want to transform Simulink models to help meet new incremental safety goals
- We have been successfully developing tools to do these things for 6+ years
- Tools are currently used by a major OEM

McMaster University
Inspiring Innovation and Discovery

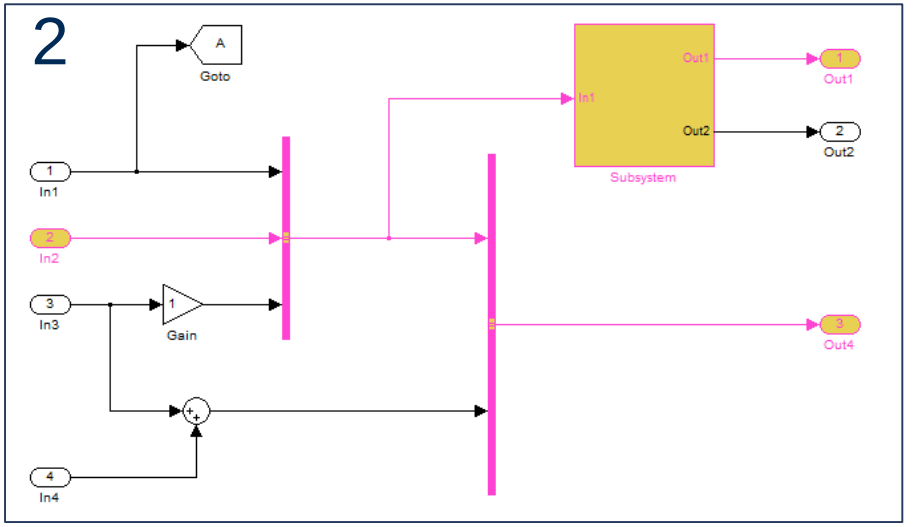
McSCert

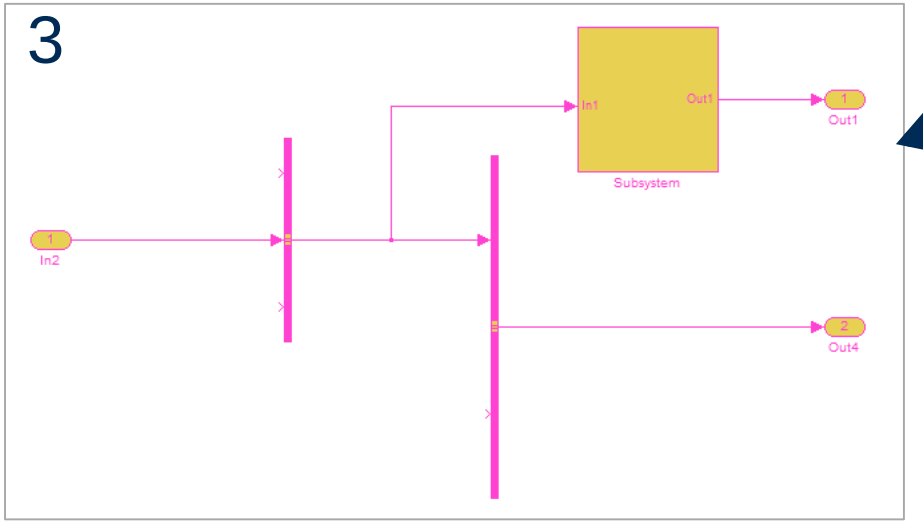
Reach/Coreach Tool [18]

BEFORE



AFTER





Trace data/control flow and slice model

McMaster University
Inspiring Innovation and Discovery

McSCert

Research Questions

- How do we effectively express the implicit assurance case behind a standard so we can automate compliance checking?
- How do we know when we can reuse evidence and when we can't?
- How do we express the reasoning behind the assurance case and evolve it correctly?
- How do we automate the evolution process in a safe and effective manner?

McMaster University
Inspiring Innovation and Discovery

McSCert

Challenges (research and technical)

- Automating & integrating the production of required assurance artefacts as part of the forward system/software development process
  - E.g. PSD example changes ASIL from B ->C then automatically generate addition test cases for unit test coverage for Method

    1e Back-to-back comparison test between model and code
- Automating identification of Simulink changes that could impact safety goals
- Checking conformance of an assurance case with development processes

Current Research: Example Tools from NASA

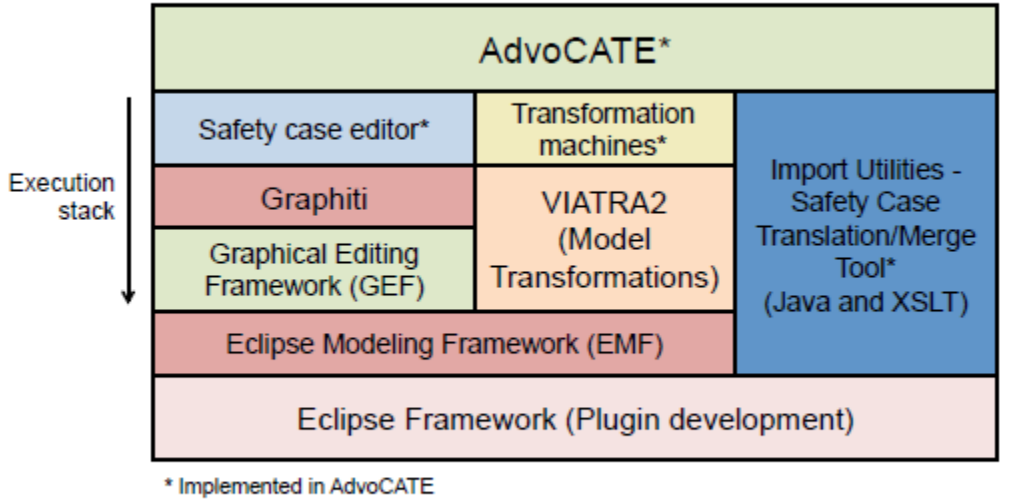
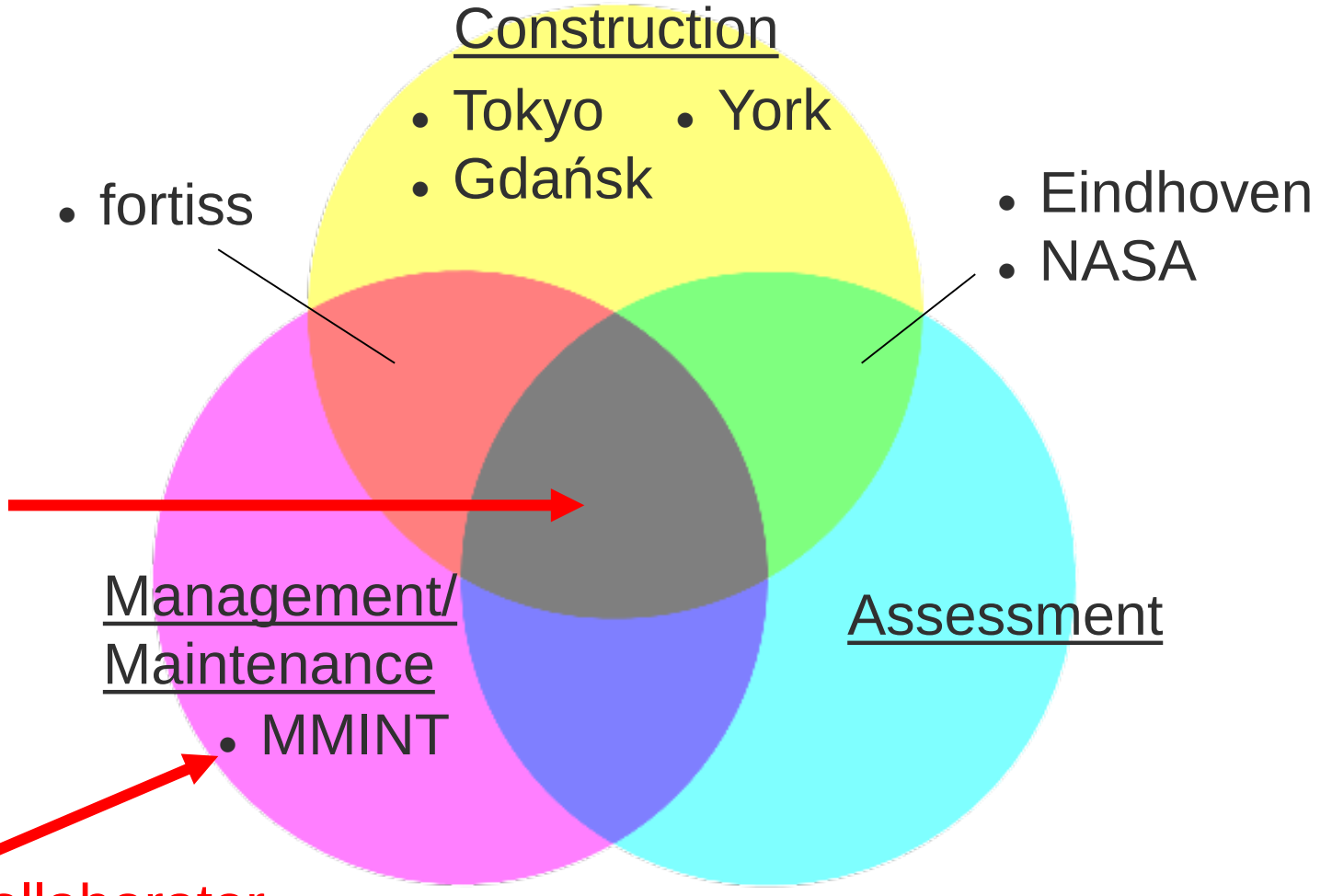


Fig. 2. Frameworks in the AdvoCATE tool chain architecture.

- AutoCert [13]
  - Generation of fragments of explicit safety cases for formal verification
  - Done by using domain specific information that can be compiled into "annotation patterns"
- ADVocate [14]
  - Edit and combine partial safety cases from AutoCert
  - Transform models to generate different views and artefacts to support safety case development & evaluation
  - Supports metrics for safety case evaluation

Assurance Case Tools (product view)



Eventually we want to be here!

Collaborator Chechik@UToronto is here!

McMaster University
Inspiring Innovation and Discovery

McSCert

Fallacies in existing (implicit) Assurance Cases for ADAS

- The driver is going to catch the Machine Learning (ML) failures
- https://giphy.com/embed/3o85xswHIaaX3R19de

McMaster University
Inspiring Innovation and Discovery

McSCert

Getting too (artificially) intelligent with safety

- Object identification is very useful
- Can help predict and plan in addition to help partially meet some safety goals
- Pedestrian detection is a good example of how ML fails badly with the key safety requirement: “Don’t hit things!”

AI/ML Version:

“I don’t know what it is so its not there”

vs

Safety Version:

“I don’t know what it BUT ITS THERE!”

Getting too (artificially) intelligent with safety

- Uber
  - Switched off Volvo's standard Aptiva/Intel Mobile Eye collision avoidance/mitigation system
  - It would have detected pedestrian 1 second before impact and started braking
  - Why?
    - To reduce interference/false positives?
    - Think of trying to making a right turn @Younge & Dundas in TO
  - Maybe because they weren't required to have it on by an industry standard assurance case
  - What's a poor autonomous vehicle to do?
    - Maybe Boat Towing Requirement can help answer this.

McMaster University
Inspiring Innovation and Discovery

McSCert

A Tesla Model X Crash Cause?

McMaster University
Inspiring Innovation and Discovery

McSCert

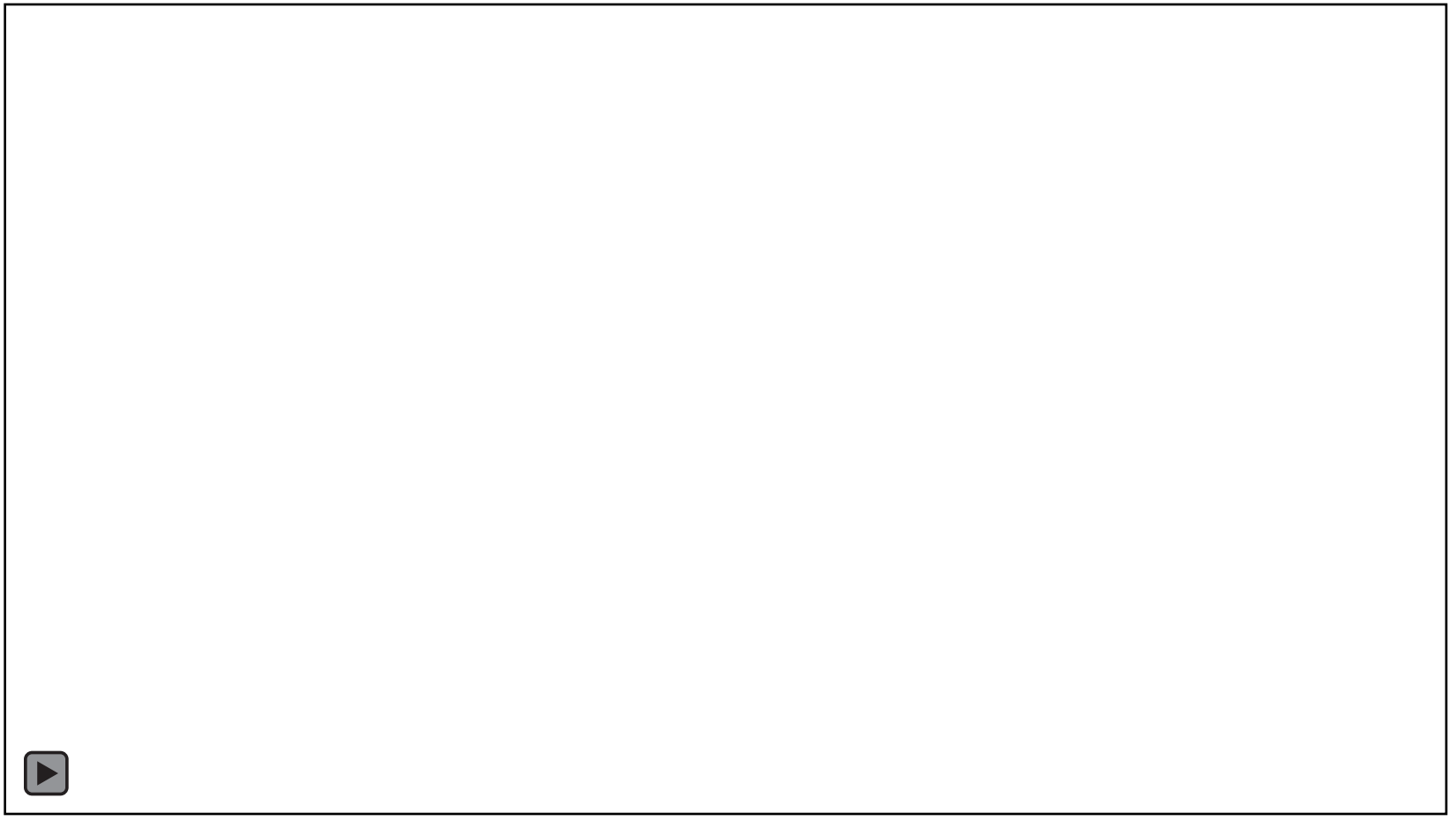
The trouble with AI in safety critical situations

- Using ML to deal with cross walks:
  - AI does a good job with this but not …

McMaster University
Inspiring Innovation and Discovery

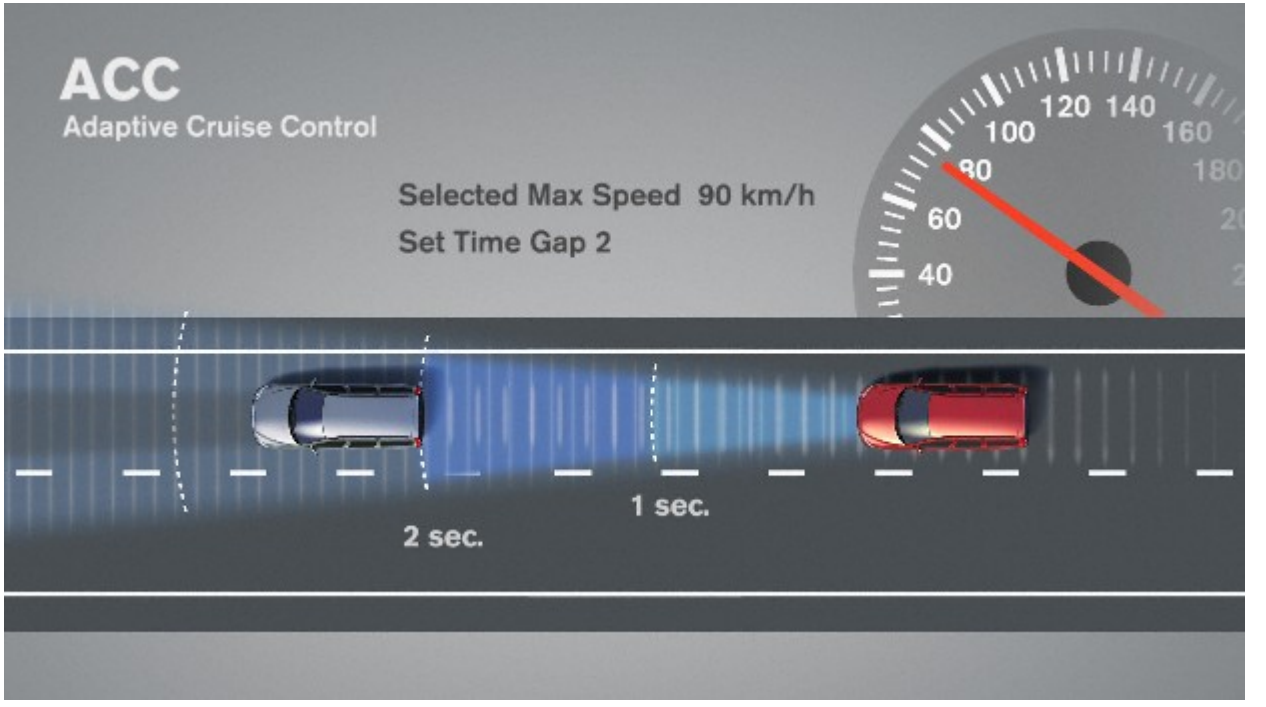
McSCert

If ML Doesn't Recognize It, It's Not There



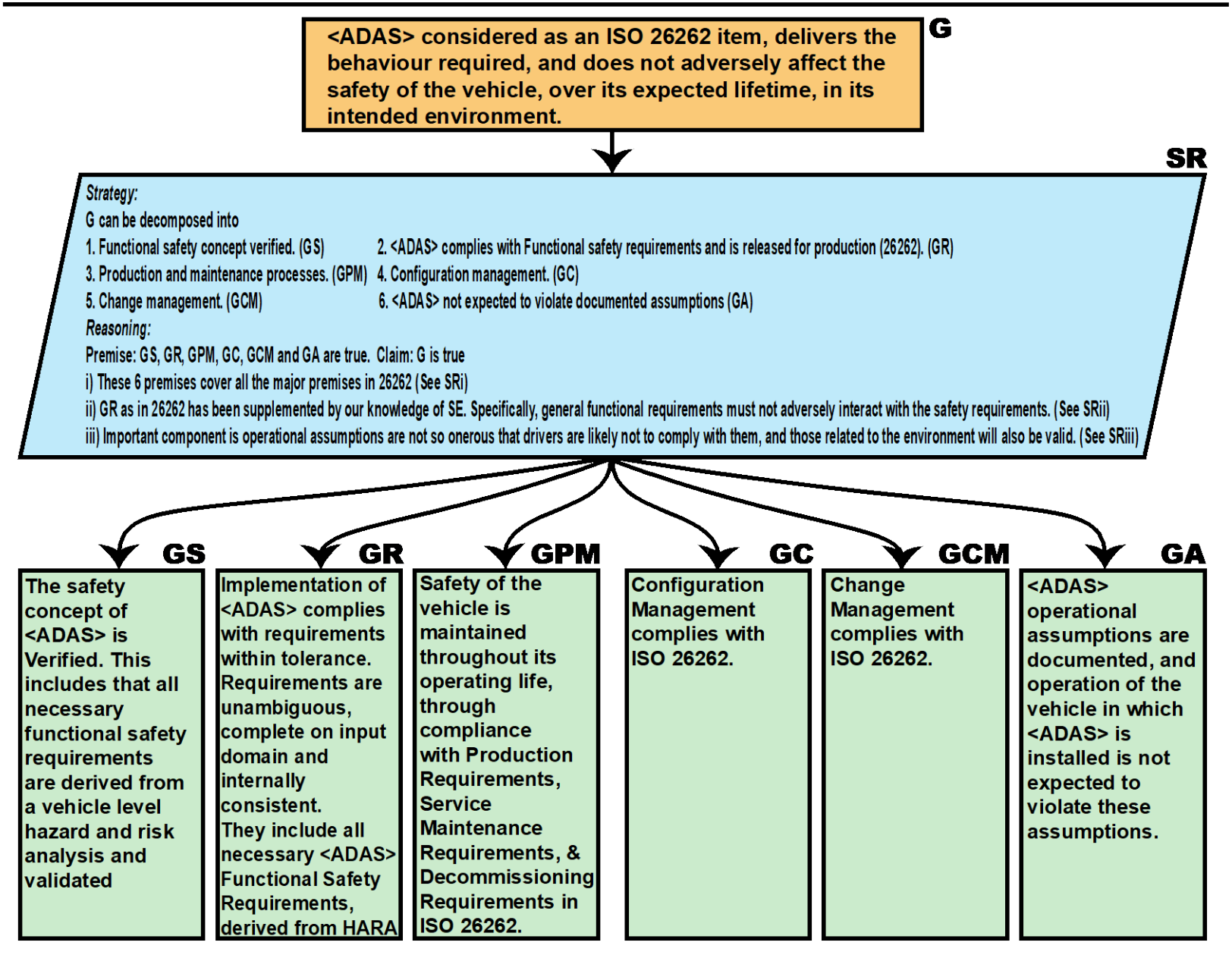
Instantiating Assurance Case Template for ADAS for ACC

- Adaptive Cruise Control is increasingly common on vehicle
- Many possible variants for sensors – camera, Doppler radar, etc
- Created a feature diagram



- These variants are part of a product line
- We have instantiated the ADAS Template for several variants

Assurance Case Template for ADAS



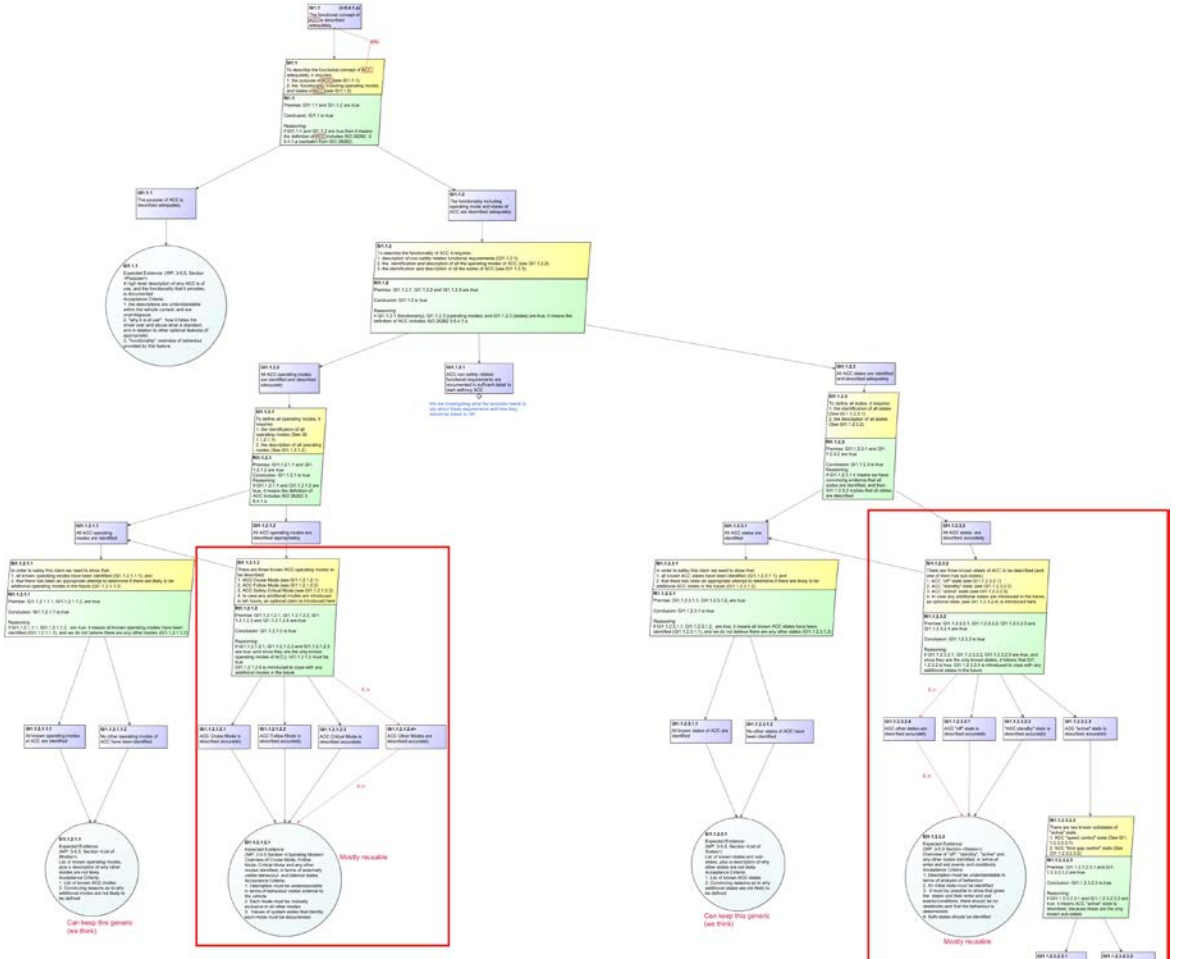
Isolating Differences in Product-Line

McMaster University
Inspiring Innovation and Discovery

McSCert

In ADAS template:

- product differences are kept to lower levels of the assurance case template
- Makes the template robust with respect to change

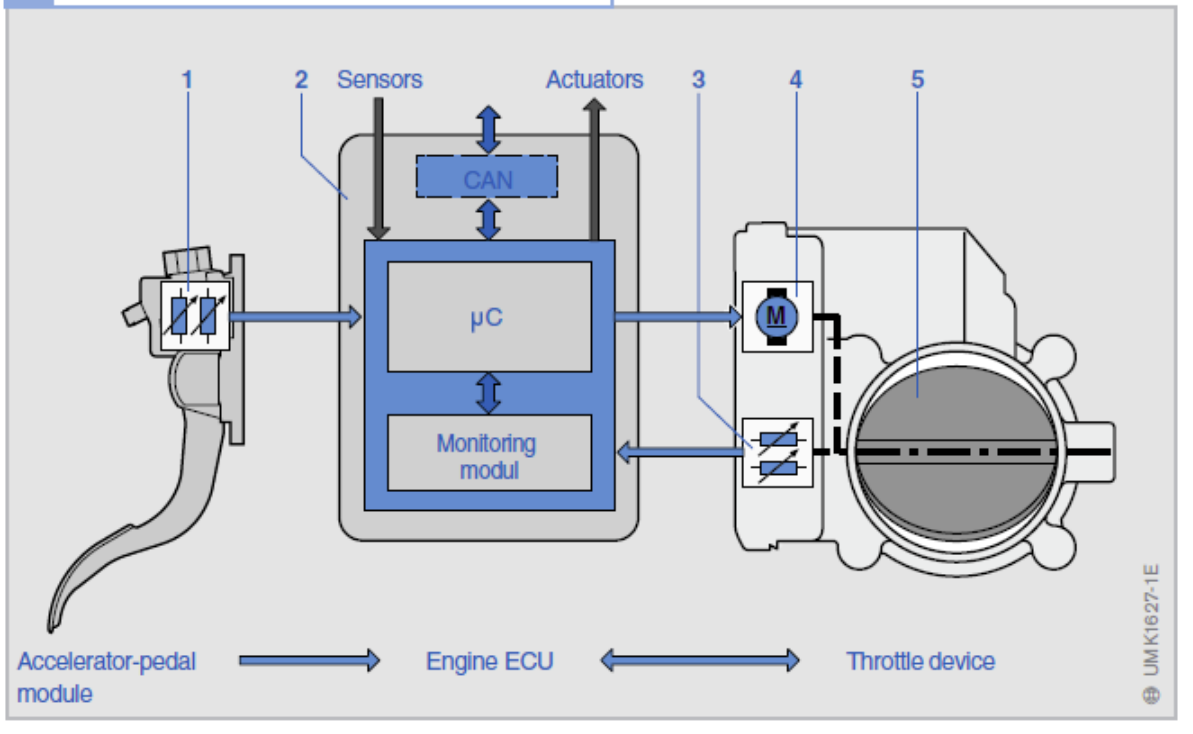


Specific to <ACC>

Specific to <ACC>

Electronic Throttle Control (EGAS) [19]

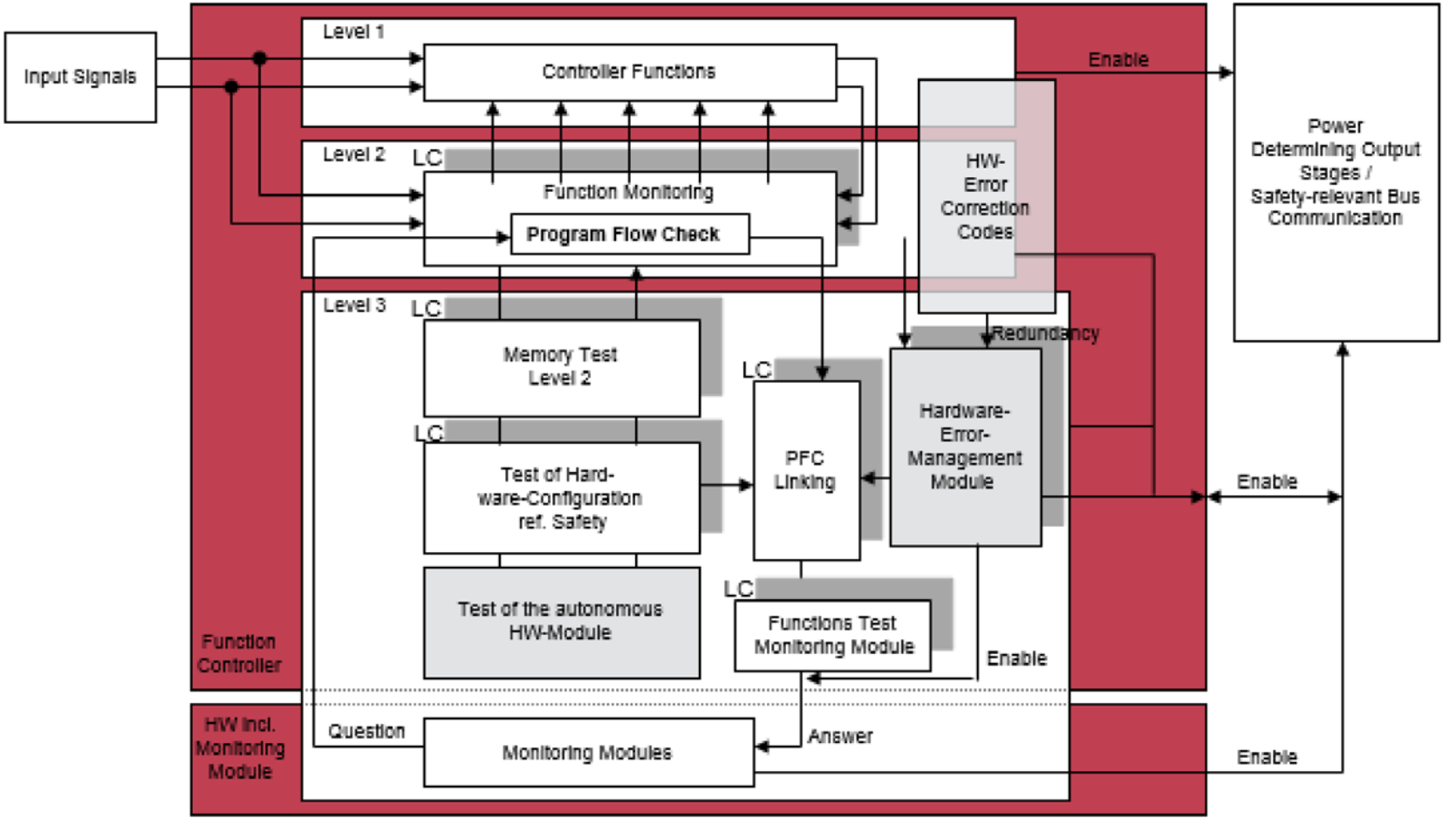
The ETC system (Electronic Throttle Control or EGAS)



1 Accelerator-pedal sensor
2 Engine ECU
3 Throttle-valve-angle sensor
4 Throttle-valve drive (DC motor)
5 Throttle valve

McMaster University
Inspiring Innovation and Discovery

3 Level Safety for EGAS [19]



McMaster University
Inspiring Innovation and Discovery

McSCert

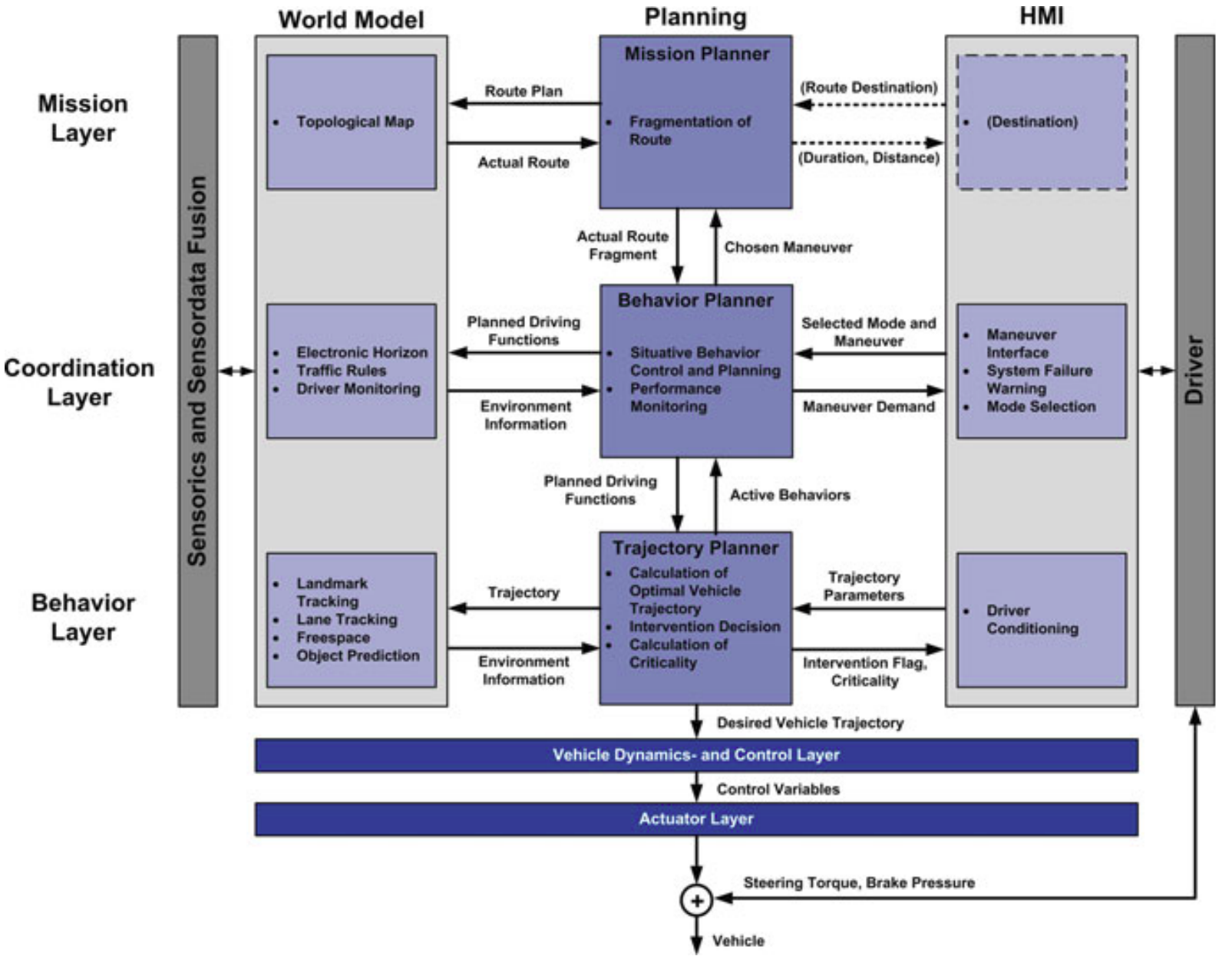
Can We Use 3 Level Safety for Autonomous Driving?

- Provide a reference architecture
- Provide a reference Safety Assurance Case Template
  - That includes explicit standardized sub-argument, supported by the architecture for “Don’t hit stuff” that mitigates ML failures

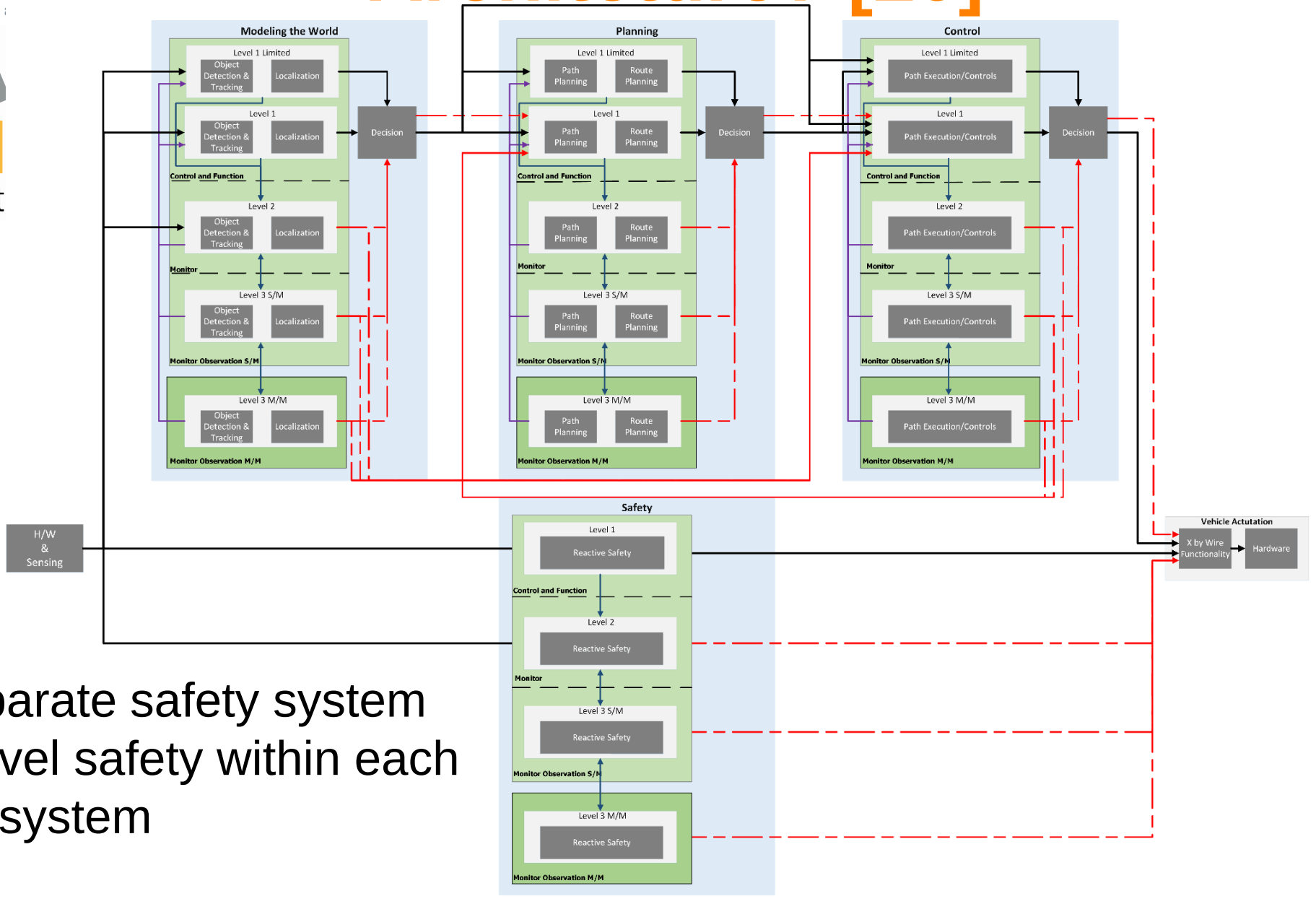
McMaster University
Inspiring Innovation and Discovery

McSCert

Autonomous Driving Architecture



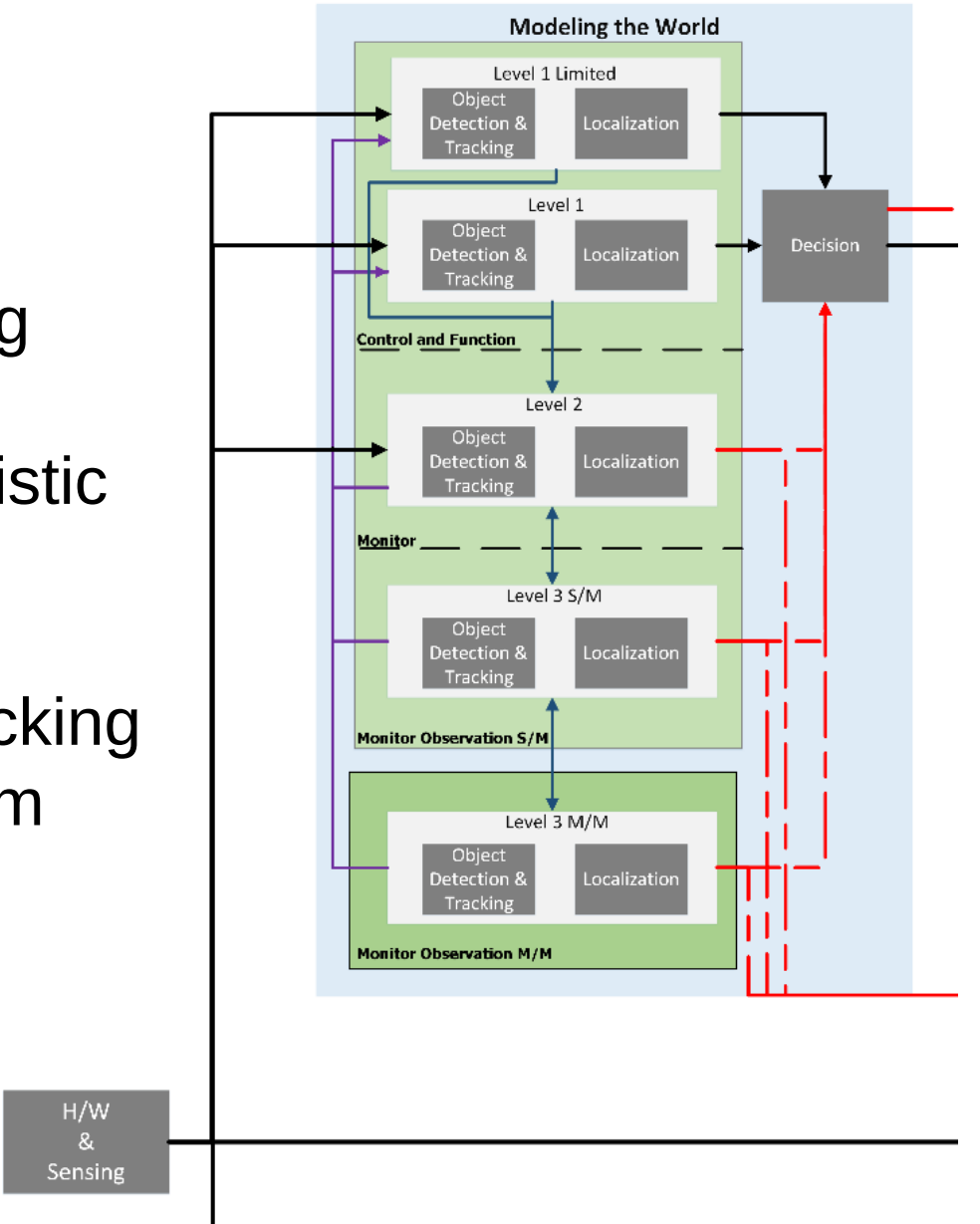
3 Level ADAS Reference Architeture? [20]



- Separate safety system
- 3 level safety within each subsystem

3 Level Safety for Object Detection/Localization

- Level 1: Machine Learning
- Level 2: Simple Deterministic Monitoring algorithm
- Level 3: Can provide checking inputs to Level 2 to confirm



Conclusion

Tesla Model S Autopilot fatality
Source: Reuters

"The primary goal of this research project is to improve the safety, security and dependability of advanced automotive software systems, by developing methods and tools for creating and managing effective and practical assurance cases in the automotive industry."

McMaster Team

- Prof. Mark Lawford, PI
- Prof. Alan Wassyng, Safety Lead
- Prof. Tom Maibaum, MM & Safety
- Dr. Zinovy Diskin, RE
- Paul Nguyen, RE→MASc student
- Sahar Kokaly, PhD student
- Thomas Chowdhry, PhD student
- Alison Bayzat, MASc student
- Paul Aoanan, MASc student

Dr. Lynda Bruce
Operations Manager

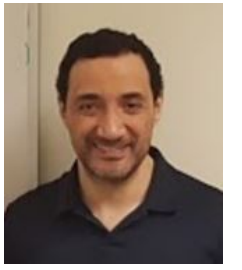
Toronto Team (MM)

Prof. Marsha Chechik (Toronto, MM lead)

Sahar Kokaly, Ph.D. student (McMaster/Toronto/GM)

Dr. Rick Salay, PDF (Toronto)

Alessio Di Sandro, research programmer (Toronto)



Ramy Shahin, Ph.D. student (Toronto)

Nick Fung, Masters student (Toronto)

Mike Maksimov, Masters student (Toronto)

McMaster University
Inspiring Innovation and Discovery

McSCert

Anticipated Outcomes

- Scientific contribution
  - Model management for impact analysis of assurance cases and standards compliance
  - Sound theory of incremental assurance
  - Product line extensions of assurance cases
- Benefits to industry
  - Reduced development time
  - Increased confidence in assurance results
- General benefits
  - Improved vehicle safety for next gen vehicles

McMaster University
Inspiring Innovation and Discovery

McSCert

Q & A

Questions?

McMaster University
Inspiring Innovation and Discovery

McSCert

References

1. ISO, “26262: Road vehicles—Functional safety,” International Standard ISO/FDIS, vol. 26262, 2011.
2. J. Rushby, X. Xu, M. Rangarajan, and T. L. Weaver, “Understanding and evaluating assurance cases,” SRI International, Tech. Rep., 2015.
3. R. Bloomfield, P. Bishop, C. Jones, and P. Froome, “ASCAD, Adelard Safety Case Development Manual,” Adelard, 1998. ISBN 0-9533771-0, vol. 5, 1998.
4. T. P. Kelly, “Arguing Safety—A Systematic Approach to Safety Case Management,” Department of Computer Science, The University of York, 1998.
5. A. Wassyng, P. Joannou, M. Lawford, T. S. Maibaum, and N. K. Singh, “Chapter 13 New Standards for Trustworthy Cyber-Physical Systems,” in Trustworthy Cyber-Physical Systems Engineering. CRC Press, 2016, pp. 337–368.
6. T. Ankrum and A. Kromholz, “Structured assurance cases: three common standards,” in HASE 2005: 9th IEEE International Symposium on High-Assurance Systems Engineering, 2005, pp. 99–108.
7. C. M. Holloway, “Making the implicit explicit: Towards an assurance case for DO-178C,” 2013.
8. C. M. Holloway, “Explicate’78: Uncovering the Implicit Assurance Case in DO-178C,” 2015.
9. J. Birch, R. Rivett, I. Habli, B. Bradshaw, J. Botham, D. Higham, P. Jesty, H. Monkhouse, and R. Palin, “Safety cases and their role in ISO 26262 functional safety assessment,” Computer Safety, Reliability, and Security. Springer, 2013, 154–165.
10. A. B. Hocking, J. Knight, M. A. Aiello, and S. Shiraishi, “Arguing software compliance with ISO 26262,” in Software Reliability Engineering Workshops, 2014, 226–231.
11. B. Gallina, et al, “VROOM & cC: a Method to Build Safety Cases for ISO 26262-compliant Product Lines”, *SAFECOMP 2013 - Workshop SASSUR,* Toulouse, France. 2013.
12. B. Gallina, “A Model-Driven Safety Certification Method for Process Compliance”, In: Proc. of ISSRE'14 Workshops. pp. 204-209. IEEE, 2014.

McMaster University
Inspiring Innovation and Discovery

McSCert

References, continued

13. E. Denney, G. Pai, Evidence arguments for using formal methods in software certification}, Software Reliability Engineering Workshops (ISSREW), 2013 IEEE International Symposium on, pp. 375-380, 2013.
14. E. Denney, G. Pai, J. Pohl, “AdvoCATE: An assurance case automation toolset”, Computer Safety, Reliability, and Security. Springer, pp. 8-21, 2012.
15. T. Chowdhury, C-W. Lin, B. Kim, M. Lawford, S. Shiraishi, A. Wassyng, "Principles for Systematic Development of an Assurance Case Template from ISO 26262", IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), IEEE, 2017, 69-72.
16. S.Z. Naqvi, Compliance checking with ISO 26262 using Conceptual Modeling, M.A.Sc. Thesis, Department of Computing and Software, McMaster University, 2018.
17. S. Kokaly, R. Salay, M. Chechik, M. Lawford, T. Maibaum, "Safety Case Impact Assessment in Automotive Software Systems: An Improved Model-Based Approach", SAFECOMP 2017, LNCS 10488, Springer International Publishing, 2017, 69–85.
18. V. Pantelic, S, Postma, M. Lawford, M. Jaskola, B. Mackenzie, A. Korobkine, M. Bender, J. Ong, G. Marks, A. Wassyng, "Software engineering practices and Simulink: bridging the gap", International Journal on Software Tools for Technology Transfer 20 (1), 2018, 95-117.
19. EGAS Workgroup, Standardized E-Gas Monitoring Concept for Gasoline and Diesel Engine Control Units, 2015.
20. S.A. Shah, Safety Architectures for Autonomous Driving, M.A.Sc. Thesis, Department of Computing and Software, McMaster University, (In preparation).