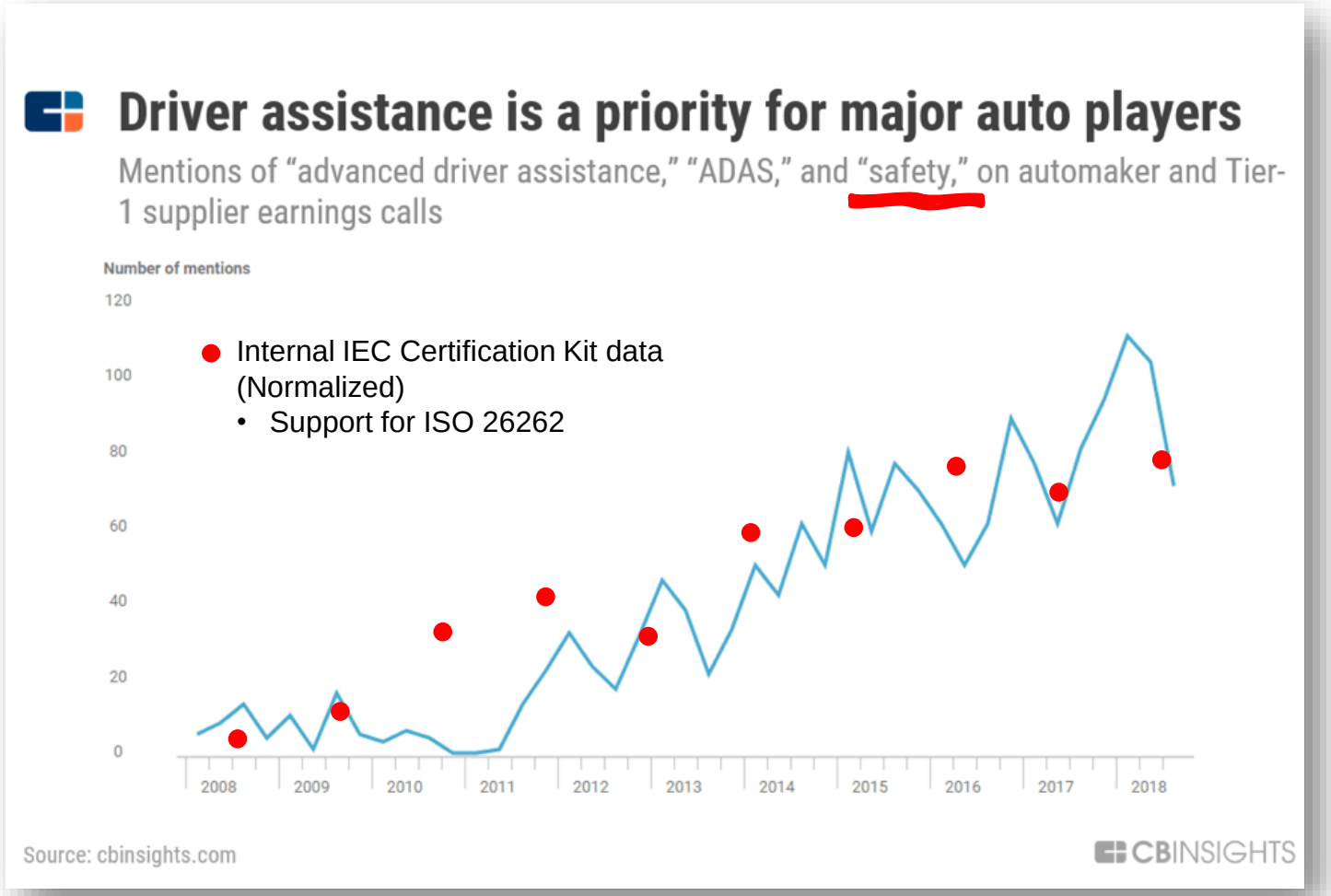
Process Pitfalls in ISO 26262 Compliance

MathWorks Consulting
Jason Moore

Functional safety industry trend

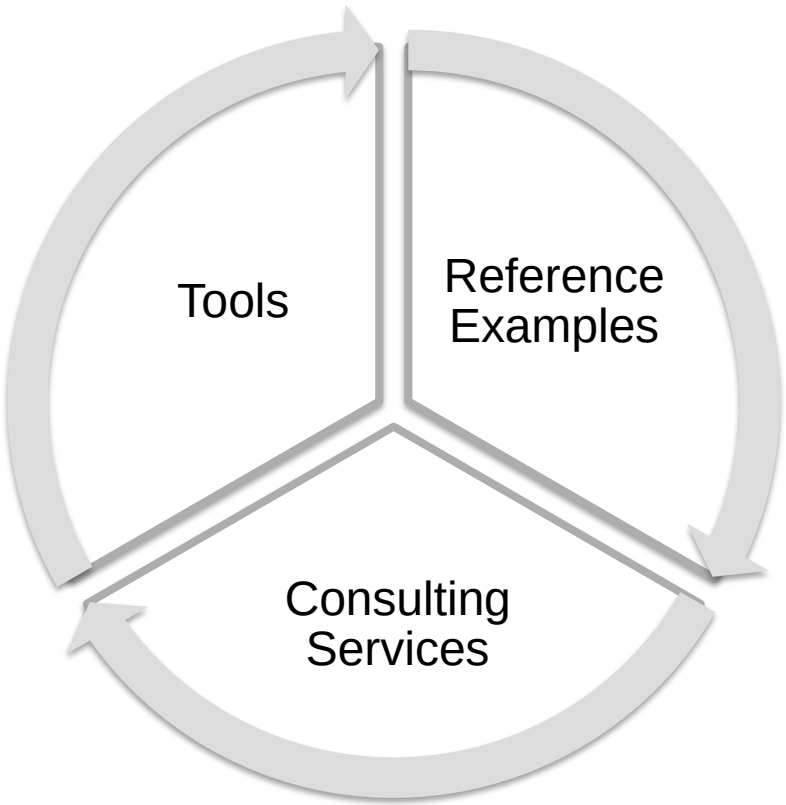


Growing interest in **safety**

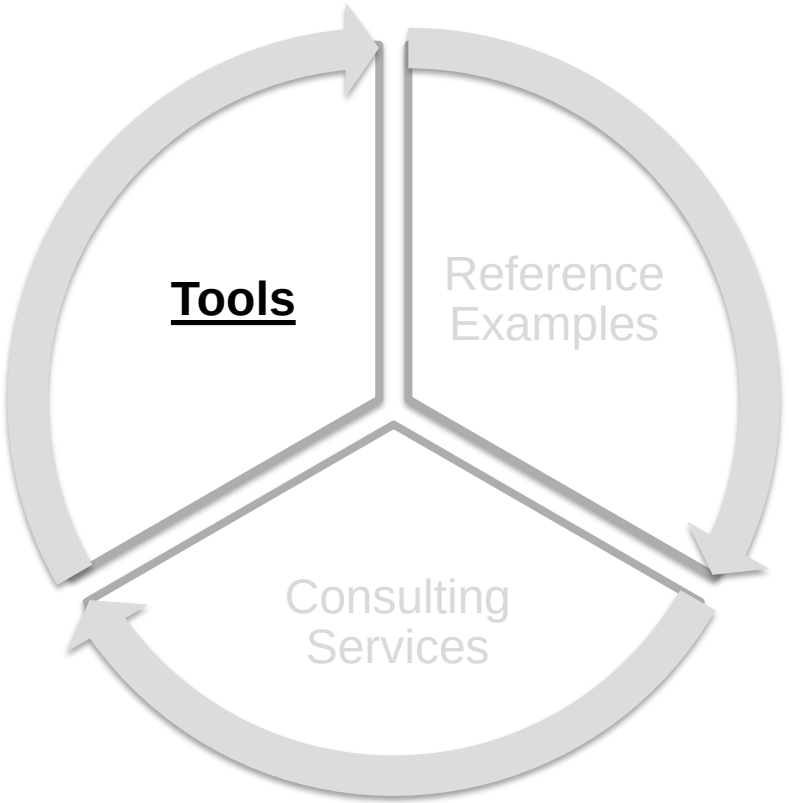
ISO 26262

Is your organization **ready** for ISO?

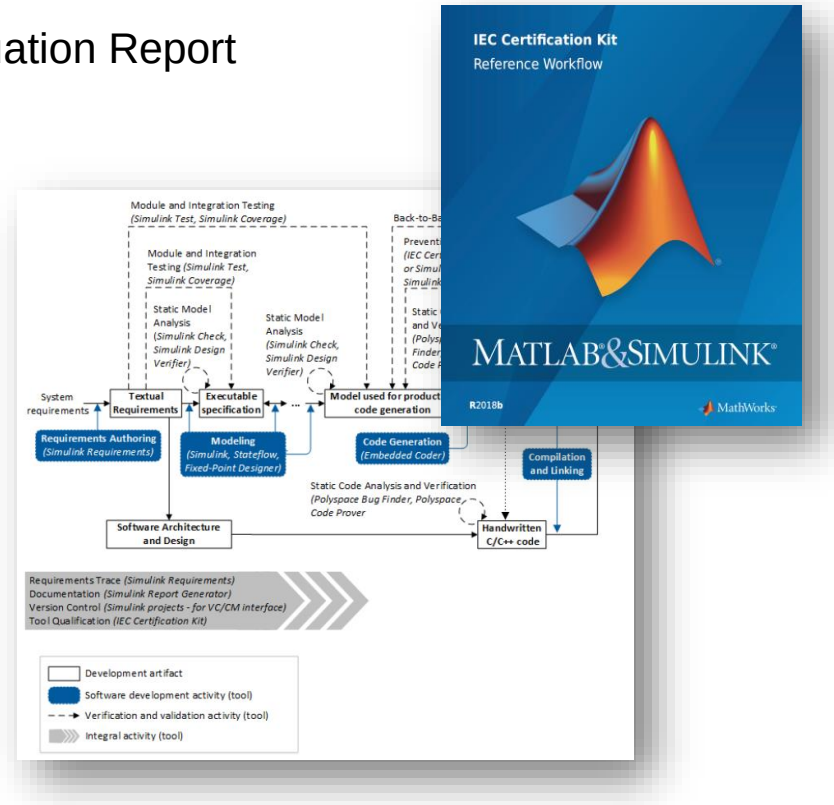
Multifaceted support for ISO 26262



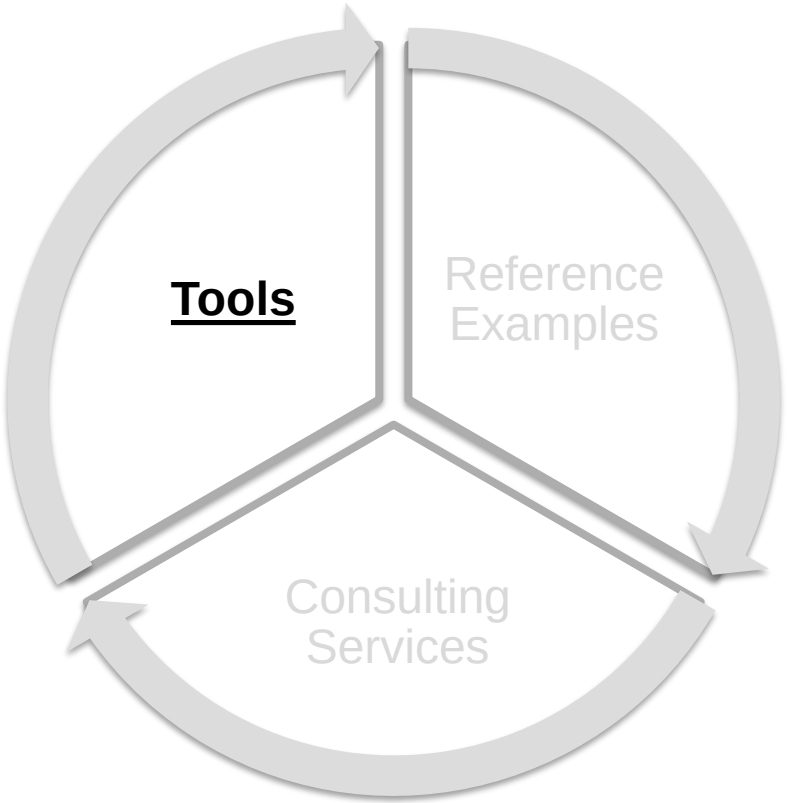
Multifaceted support for ISO 26262



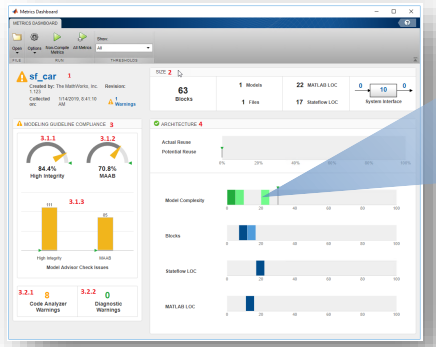
- IEC Certification Kit
  - Model-Based Design Reference Workflow
  - Tool Qualification Package
    - Software Tool Criteria Evaluation Report
    - Software Tool Qualification
    - Tool Validation Suite
    - …etc.



Multifaceted support for ISO 26262

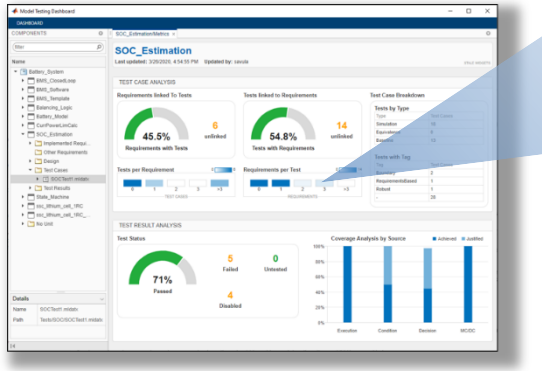


- Targeted Features
  - Model Metrics Dashboard



R2017b

  - Model Testing Dashboard



R2020b

Design Compliance

Table 1 — Topics to be covered by modelling and coding guidelines

Topics		ASIL A	ASIL B	ASIL C	ASIL D
1a	Enforcement of low complexity[a]	++	++	++	++
1b	Use of language subsets[b]	++	++	++	++
1c	Enforcement of strong typing[c]	++	++	++	++
1d	Use of defensive implementation techniques[d]	+	+	++	++
1e	Use of well-trusted design principles[e]	+	+	++	++
1f	Use of unambiguous graphical representation	+	++	++	++
1g	Use of style guides	+	++	++	++
1h	Use of naming conventions	++	++	++	++
1i	Concurrency aspects[f]	+	+	+	+

Verification Compliance

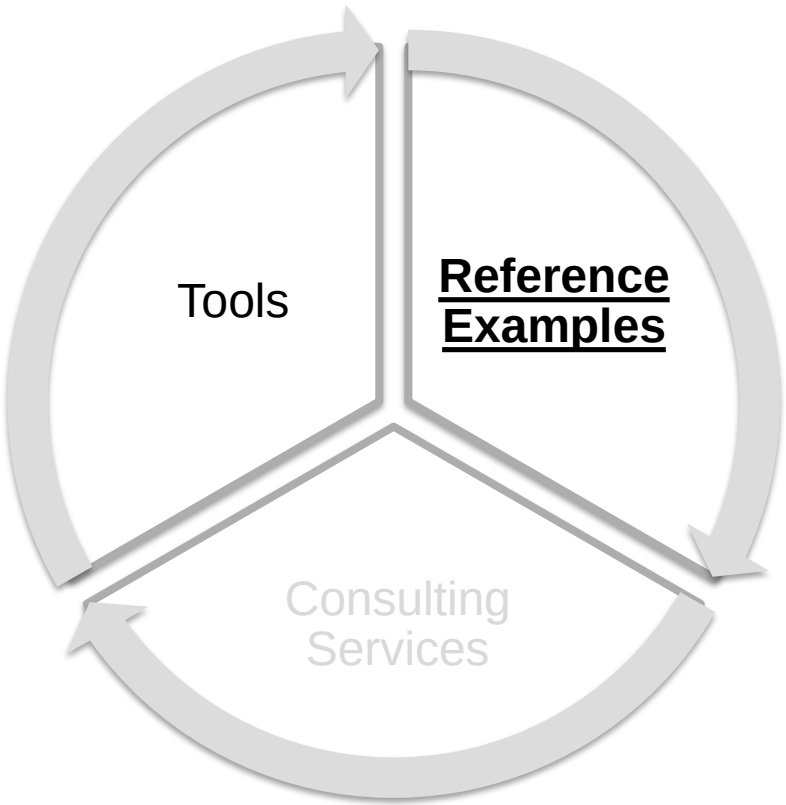
Table 8 — Methods for deriving test cases for software unit testing

Methods		ASIL A	ASIL B	ASIL C	ASIL D
1a	Analysis of requirements	++	++	++	++
1b	Generation and analysis of equivalence classes[a]	+	++	++	++
1c	Analysis of boundary values[b]	+	++	++	++
1d	Error guessing based on knowledge or experience[c]	+	+	+	+

Table 9 — Structural coverage metrics at the software unit level

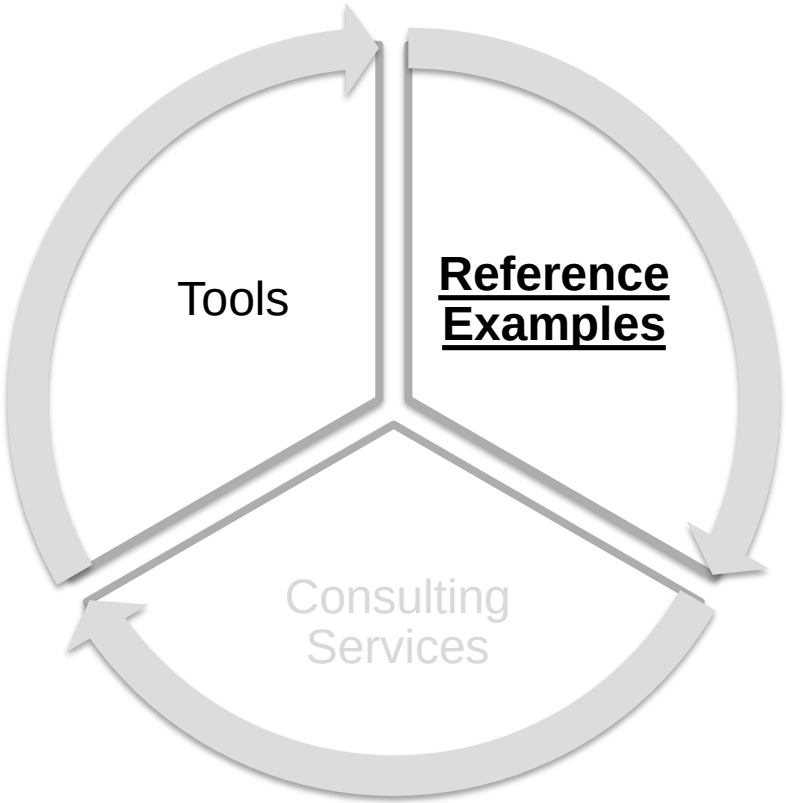
Methods		ASIL A	ASIL B	ASIL C	ASIL D
1a	Statement coverage	++	++	+	+
1b	Branch coverage	+	++	++	++
1c	MC/DC (Modified Condition/Decision Coverage)	+	+	+	++

Multifaceted support for ISO 26262



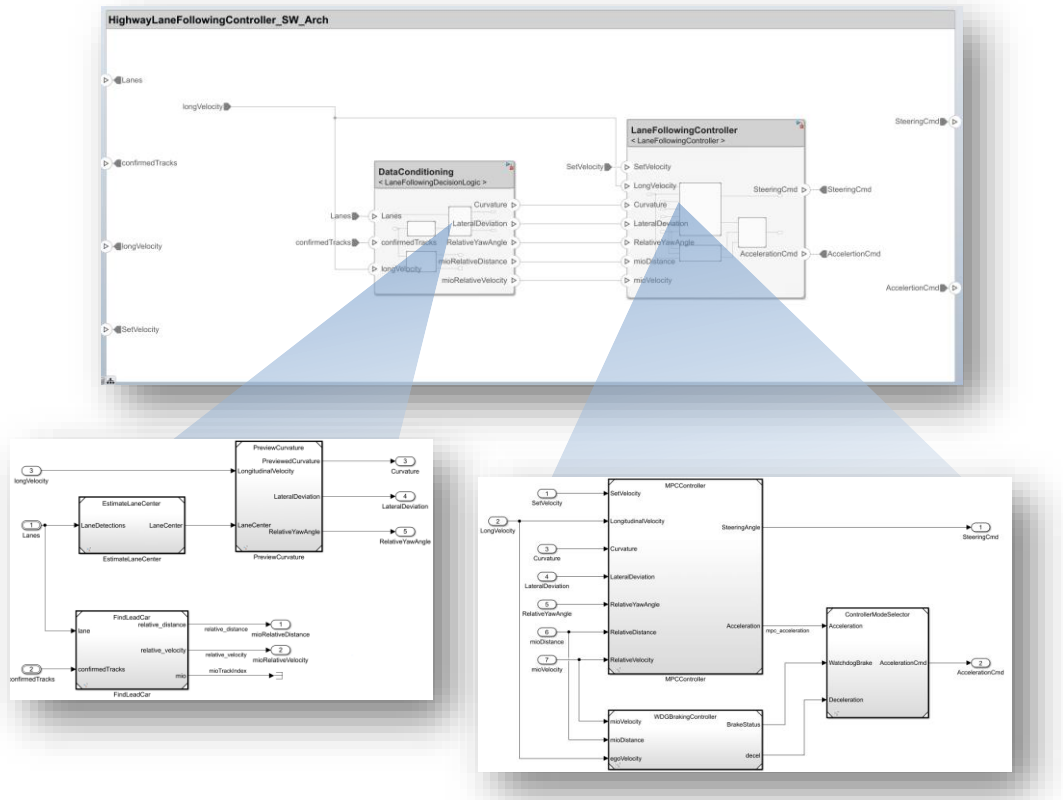
- Best Practice Paper
  - (2018) Model Quality Objectives
    - Recommended model metric and threshold
  - (2019) 11 Best Practices for Developing ISO 26262 Applications with Simulink
    - How to achieve Freedom from Interference?
  - (2020) An ISO 26262 Workflow for Automated Driving Applications Using MATLAB: Guidelines and Best Practices
    - Use of MATLAB as part of ISO 26262 workflow

Multifaceted support for ISO 26262



- Reference Application

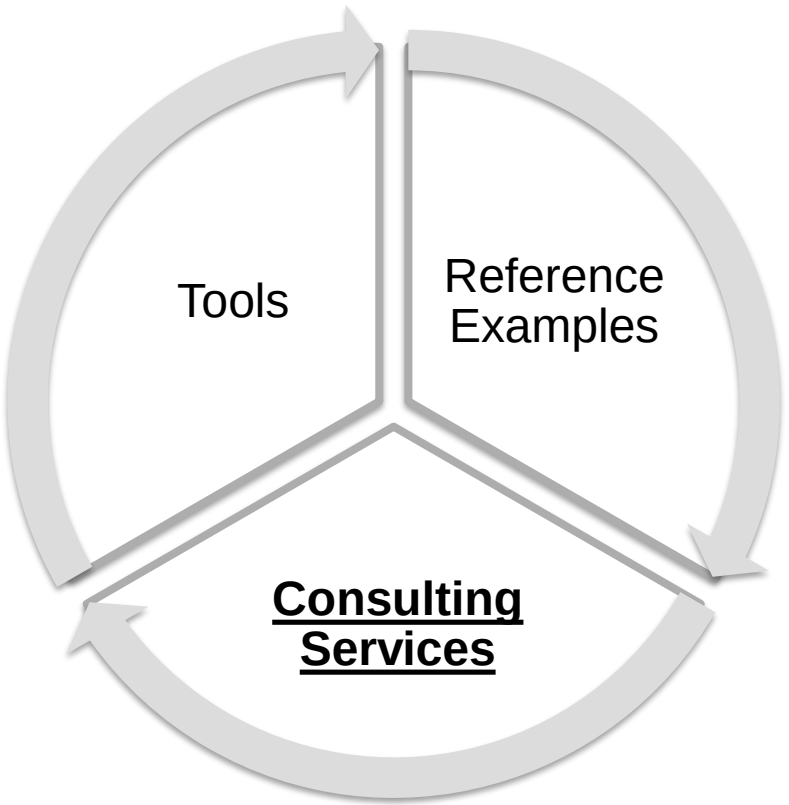
Architecture Design with System Composer



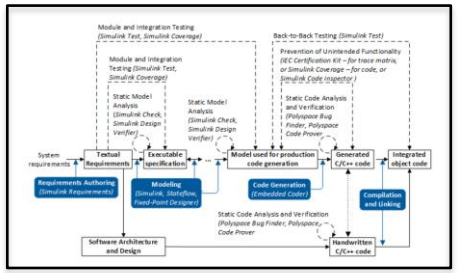
Component/Unit Design with Simulink

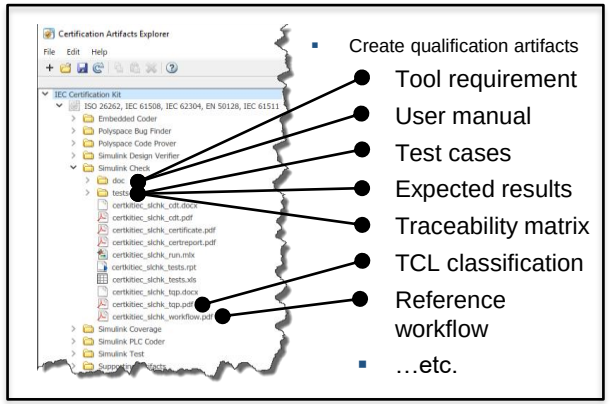
ISO 26262-6
Workflow Example

Multifaceted support for ISO 26262



jumpstart





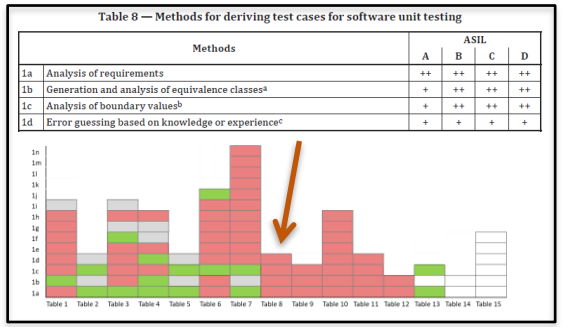
ISO Jumpstart

Process Establishment

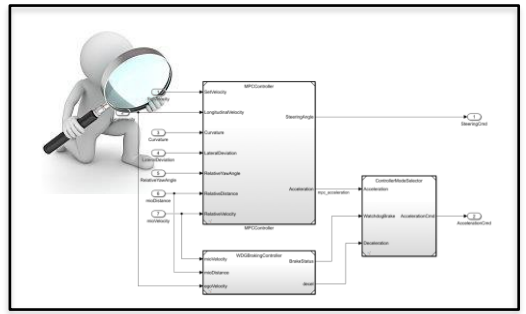
Tool Qualification Support

Range of Consulting Services

Process Gap Analysis



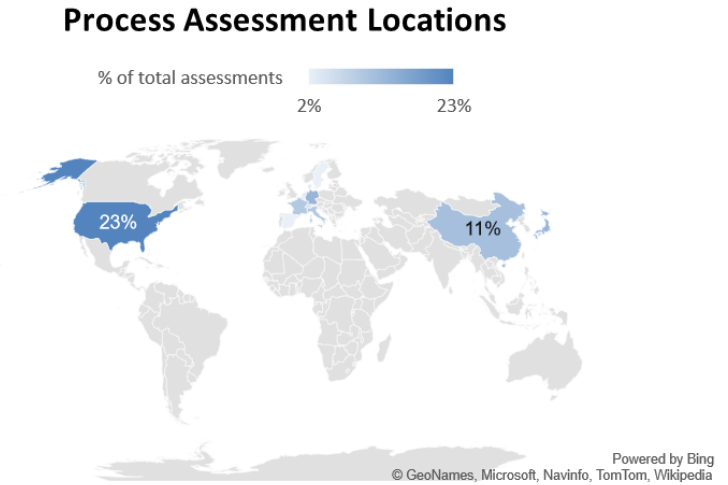
Model Review

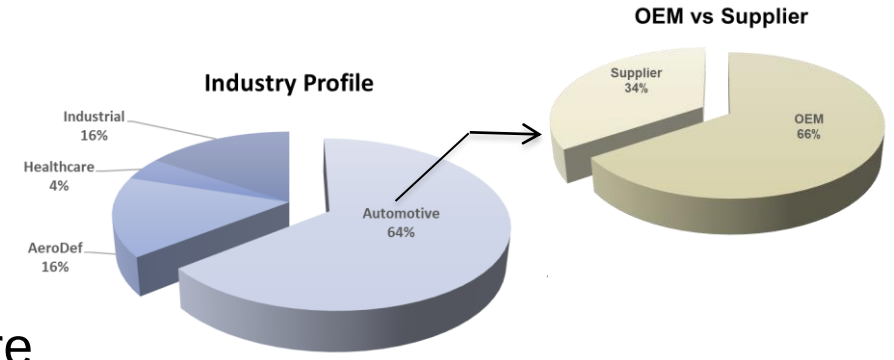


Observations based on our work with industry

Common pitfalls

- Unaware of ISO requirements
- Legacy components developed outside of ISO
- No clear mapping of ISO requirement to workflow
- Lack of tool implementation methods against ISO requirement
- No architecture consideration
- HIL-centric verification workflow
- No justification on method selections
- No clear definition of required work product
- Lack of consistency in work product
- No upfront consideration to tool qualification
- Lack of coordination between functional safety and software development
- Underestimate the effort (cost and timing) required for ISO project
- …etc.

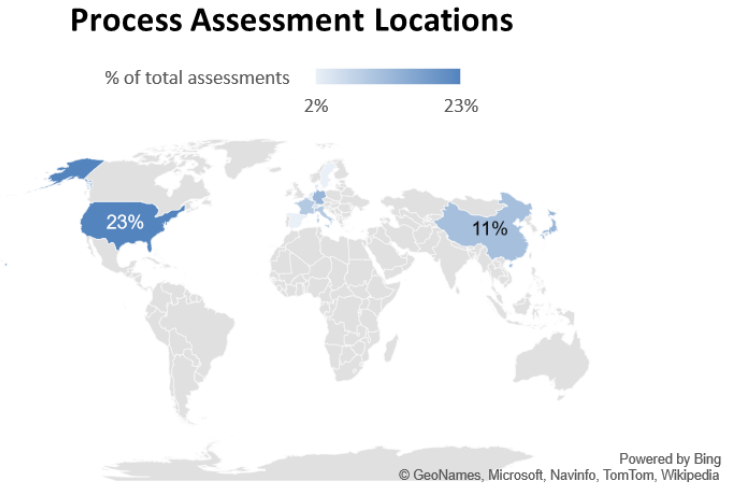


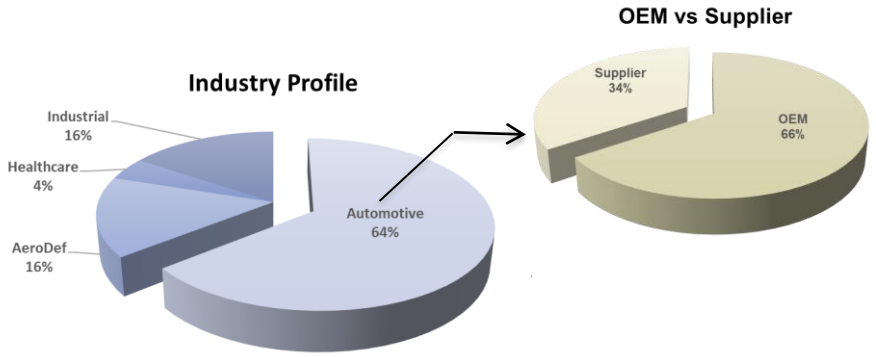


Observations based on our work with industry

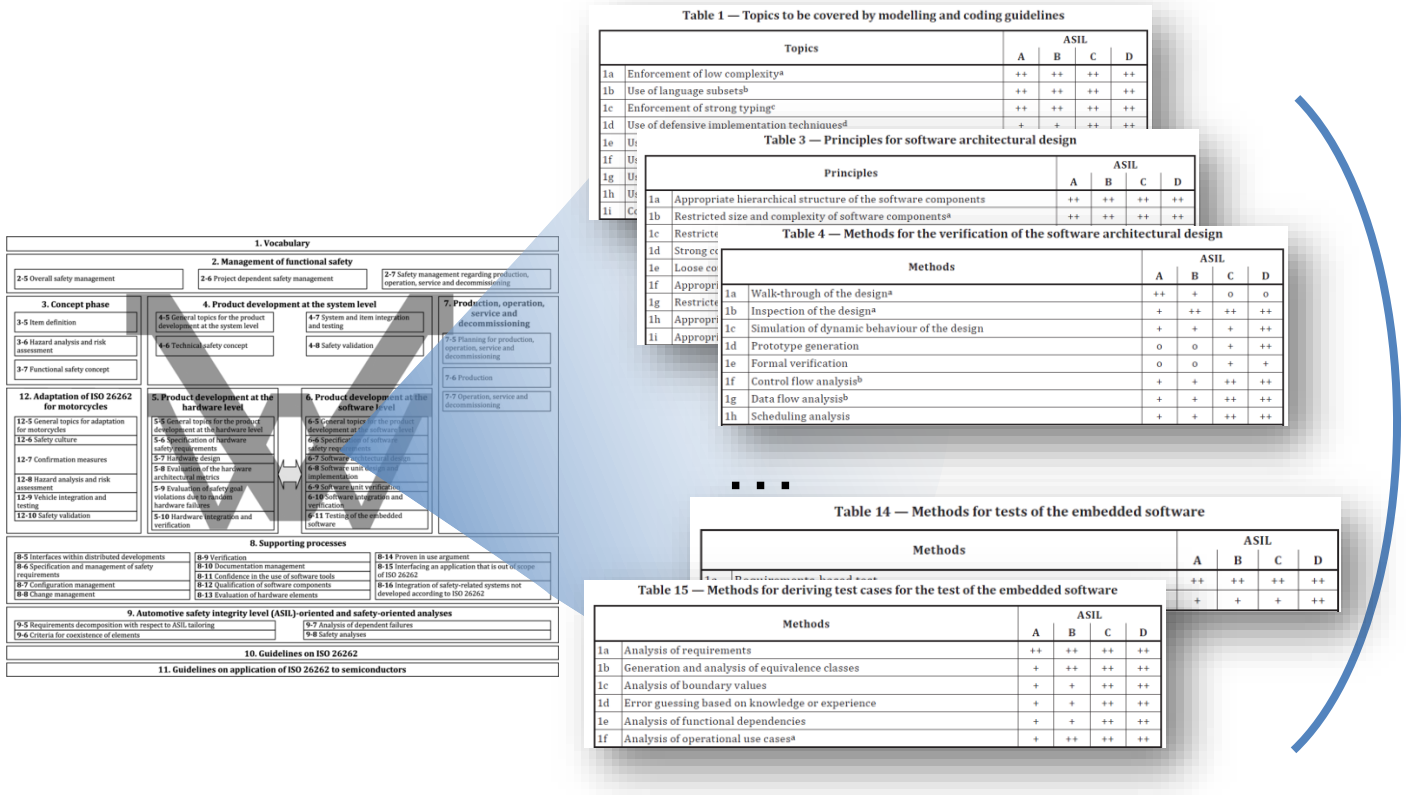
Common Themes

- Process not clearly defined or documented
- Lack top-down architectural design approach
- Poor tool qualification awareness





Process not clearly defined or documented



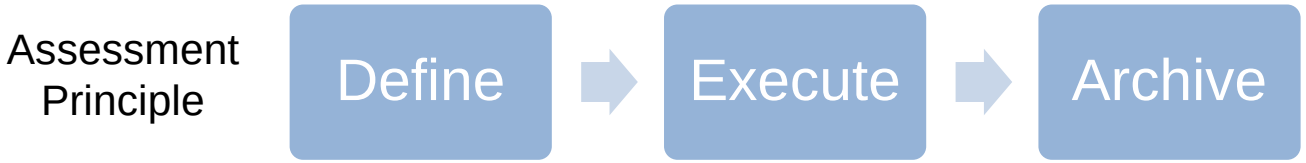
ISO 26262-6

- 15 Tables
- 90 Topics/Methods/Principles

- Which **topics/method/principles** were chosen?
- What **justification** were used?
- What **evidence** were captured?
- What are the **implementation** steps?
- …etc.?

Process not clearly defined or documented

Define process: from ISO Requirement down to Detail Work Instructions



Process not clearly defined or documented

Define process: from ISO Requirement down to Detail Work Instructions

Assessment Principle

Define → **Execute** → **Archive**

ISO recommendations

Decision to follow recommendation

Mapping to Engineering Task

Table 7 – Methods for software unit verification			
Methods		ASIL D	Applicable
1a	Walk-through	0	No*
1b	Pair-programming	+	No*
1c	Inspection	++	Yes
1d	Semi-formal verification	++	Yes
1e	Formal verification	+	Yes
1f	Control flow analysis	++	Yes
1g	Data flow analysis	++	Yes
1h	Static code analysis	++	Yes
1i	Static analysis based on abstract interpretation	+	No
1j	Requirement-based test	++	Yes
1k	Interface test	++	Yes
1l	Fault injection test	++	Yes
1m	Resource usage evaluation	++	Yes
1n	Back-to-back comparison between model and code, if applicable	++	Yes

Table 8 – Methods for deriving test cases for software unit testing			
Methods		ASIL D	Applicable
1a	Analysis of requirements	++	Yes
1b	Generation and analysis of equivalence classes	++	Yes
1c	Analysis of boundary values	++	Yes
1d	Error guessing based on knowledge or experience	++	No*

Table 9 – Structural coverage metrics at the software unit level			
Methods		ASIL D	Applicable
1a	Statement coverage	++	Yes
1b	Branch coverage	++	Yes
1c	MC/DC (Modified Condition / Decision Coverage)	++	Yes

Task Name	ISO Requirement	Work Product
Model Review	ISO 26262-6:2018 Table 3-1b ISO 26262-6:2018 Table 3-1c ISO 26262-6:2018 Clause 5 ISO 26262-6:2018 Table 7-1c ISO 26262-8:2018 6.4.2.3 ISO 26262-8:2018 6.4.3.2	Model Review Checklist
Model Testing	ISO 26262-6:2018 Table 7-1d ISO 26262-6:2018 Table 7-1j ISO 26262-6:2018 Table 7-1k ISO 26262-6:2018 Table 9-1a ISO 26262-6:2018 Table 9-1b ISO 26262-6:2018 Table 9-1c	Software Unit Test Verification Specification Model Unit Test Tool (Excel Input File) Model Coverage Report Model Coverage Filter Software Unit Test Verification Report
Software-in-the-Loop (SIL) Testing	ISO 26262-6:2018 Table 7-1n ISO 26262-6:2018 Table 9-1a ISO 26262-6:2018 Table 9-1b ISO 26262-6:2018 Table 9-1c	Software Unit Test Verification Specification Model Unit Test Tool (Excel Input File) Code Coverage Report Software unit test verification report
Processor-in-the-Loop (PIL) Testing	ISO 26262-6:2018 Table 7-1l ISO 26262-6:2018 Table 7-1m ISO 26262-6:2018 Table 7-1n	Software Unit Test Verification Specification Model Unit Test Tool (Excel Input File) Software Unit Test Verification Report
Code Analysis	ISO 26262-6:2018 Table 6 ISO 26262-6:2018 Table 7-1f ISO 26262-6:2018 Table 7-1g ISO 26262-6:2018 Table 7-1h	Polyspace Bug Finder Report Polyspace Code Prover Report Software Unit Test Verification Specification Software Unit Test Verification Report
Regression Testing	All the above	All the above
Software Unit Verification Checklist Review	ISO 26262-6:2018 Clause 9	Software Unit Verification Checklist

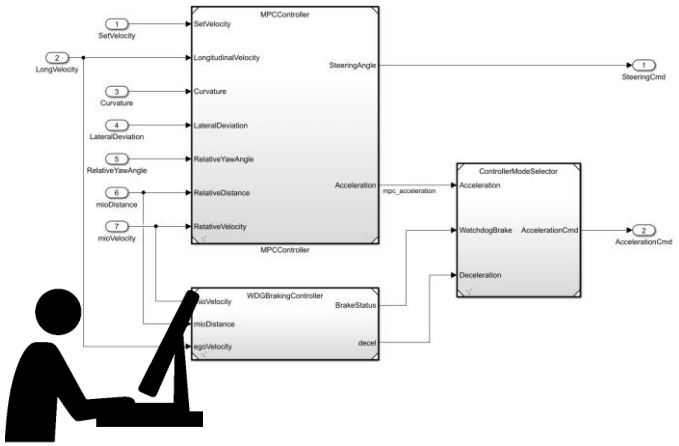
Derive Consistent Work Instructions

Process not clearly defined or documented

Define process: from ISO Requirement down to Detail Work Instructions

Assessment Principle

Define → Execute → Archive



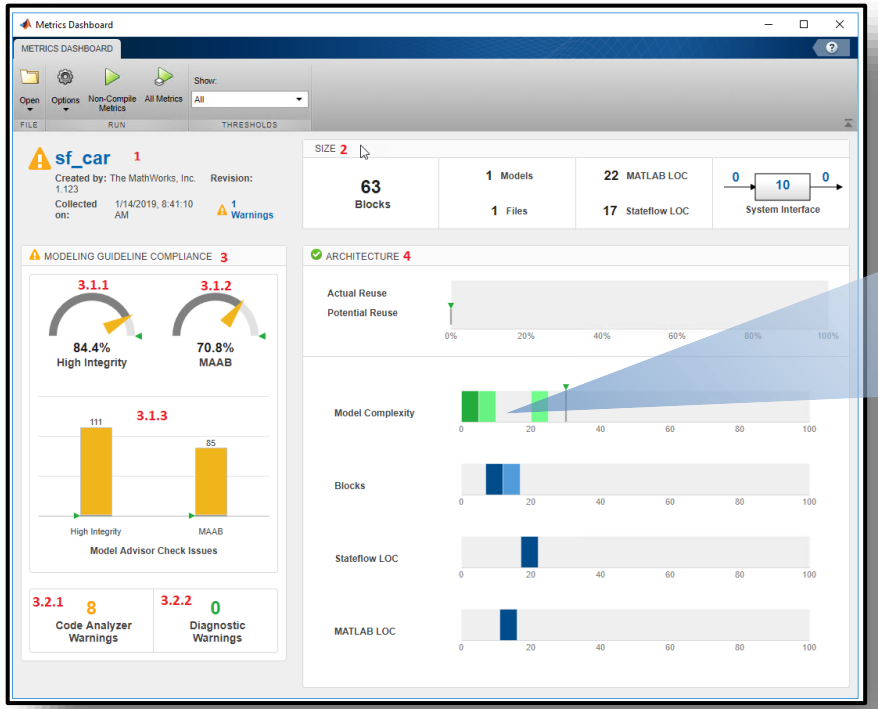


Table 1 — Topics to be covered by modelling and coding guidelines

	Topics	ASIL			
		A	B	C	D
1a	Enforcement of low complexity[a]	++	++	++	++
1b	Use of language subsets[b]	++	++	++	++
1c	Enforcement of strong typing[c]	++	++	++	++
1d	Use of defensive implementation techniques[d]	+	+	++	++
1e	Use of well-trusted design principles[e]	+	+	++	++
1f	Use of unambiguous graphical representation	+	++	++	++
1g	Use of style guides	+	++	++	++
1h	Use of naming conventions	++	++	++	++
1i	Concurrency aspects[f]	+	+	+	+

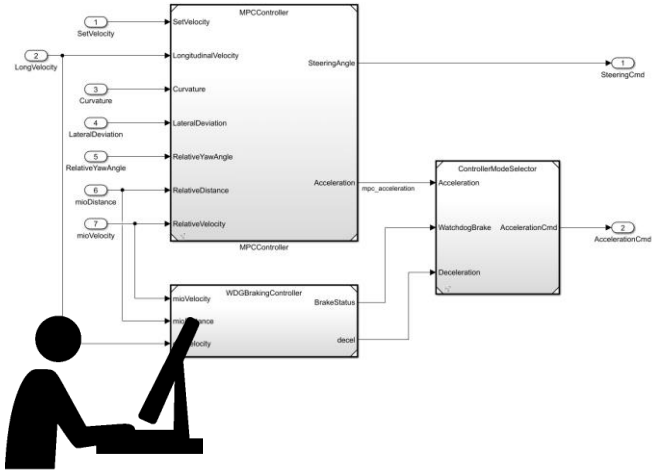
Model Metric Dashboard

Process not clearly defined or documented

Define process: from ISO Requirement down to Detail Work Instructions

Assessment Principle

Define → Execute → Archive



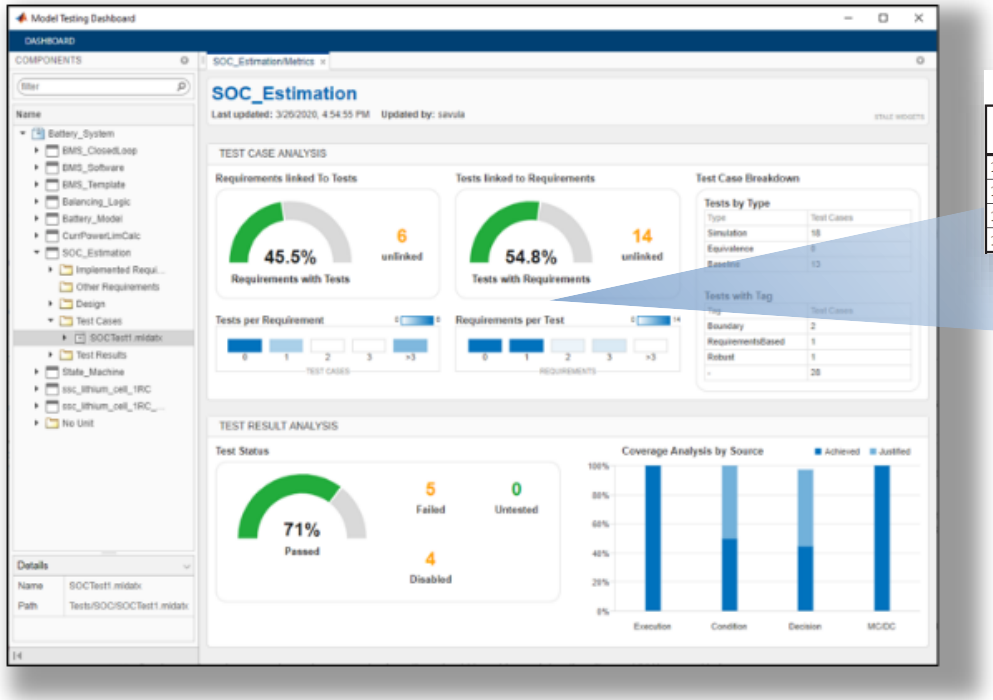


Table 8 — Methods for deriving test cases for software unit testing

	Methods	ASIL A	ASIL B	ASIL C	ASIL D
1a	Analysis of requirements	++	++	++	++
1b	Generation and analysis of equivalence classes[a]	+	++	++	++
1c	Ana…				
1d	Err…				

Table 9 — Structural coverage metrics at the software unit level

	Methods	ASIL A	ASIL B	ASIL C	ASIL D
1a	Statement coverage	++	++	+	+
1b	Branch coverage	+	++	++	++
1c	MC/DC (Modified Condition/Decision Coverage)	+	+	+	++

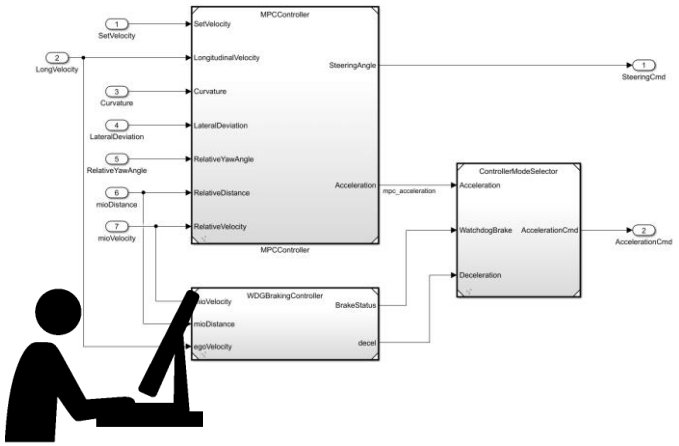
Model Testing Dashboard

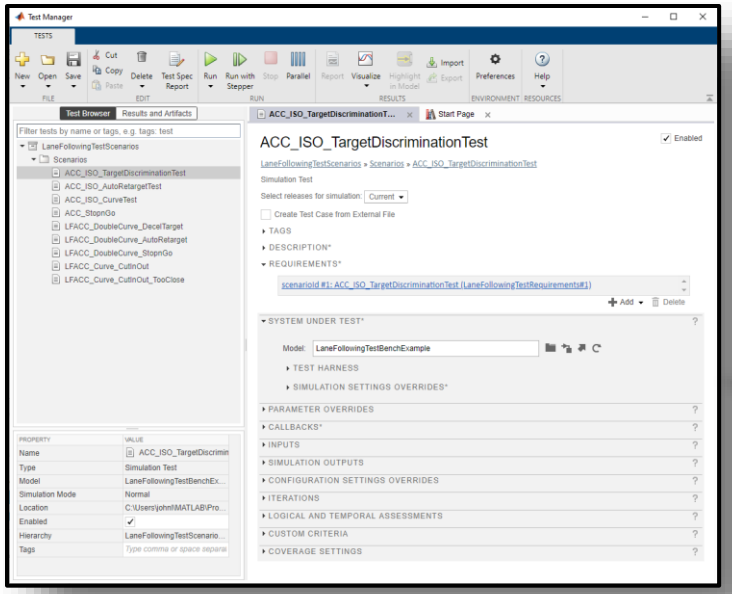
Process not clearly defined or documented

Define process: from ISO Requirement down to Detail Work Instructions

Assessment Principle

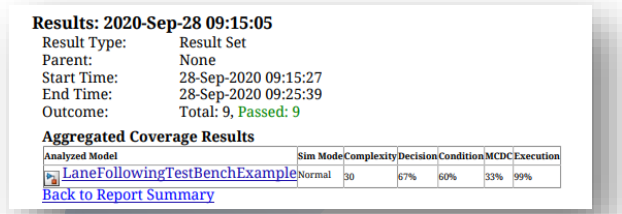
Define → Execute → Archive





Simulink Test (Test Manager)

Coverage Results



Results: 2020-Sep-28 09:15:05

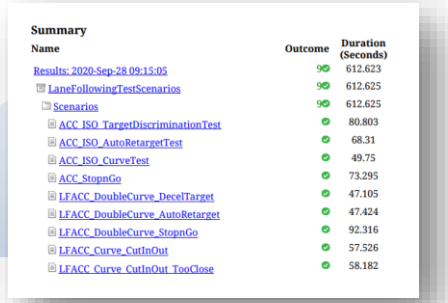
Result Type: Result Set
Parent: None
Start Time: 28-Sep-2020 09:15:27
End Time: 28-Sep-2020 09:25:39
Outcome: Total: 9, Passed: 9

Aggregated Coverage Results

Analyzed Model	Sim Mode	Complexity	Decision	Condition	MCDC	Execution
LaneFollowingTestBenchExample	Normal	30	67%	60%	33%	99%

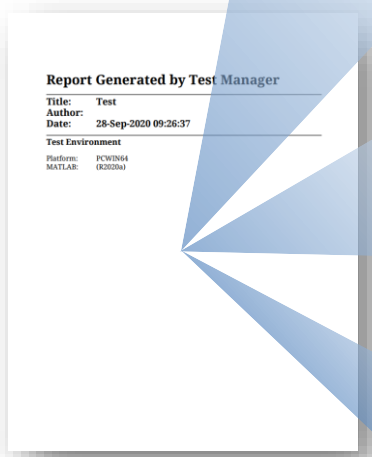
Back to Report Summary

Test Results



Summary

Name	Outcome	Duration (Seconds)
Results: 2020-Sep-28 09:15:05	9	612.623
LaneFollowingTestScenarios	9	612.625
Scenarios	9	612.625
ACC_ISO_TargetDiscriminationTest	✓	80.803
ACC_ISO_AutoRetargetTest	✓	68.31
ACC_ISO_CurveTest	✓	49.75
ACC_StopnGo	✓	73.295
LFACC_DoubleCurve_DecelTarget	✓	47.105
LFACC_DoubleCurve_AutoRetarget	✓	47.424
LFACC_DoubleCurve_StopnGo	✓	92.316
LFACC_Curve_CutInOut	✓	57.526
LFACC_Curve_CutInOut_TooClose	✓	58.182



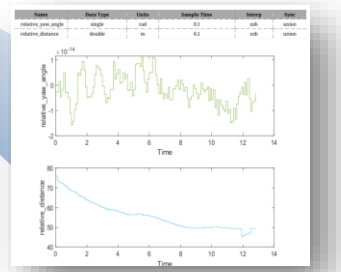
Report Generated by Test Manager

Title: Test
Author:
Date: 28-Sep-2020 09:26:37

Test Environment

Platform: PCWIN64
MATLAB: (R2020a)

Verification Report

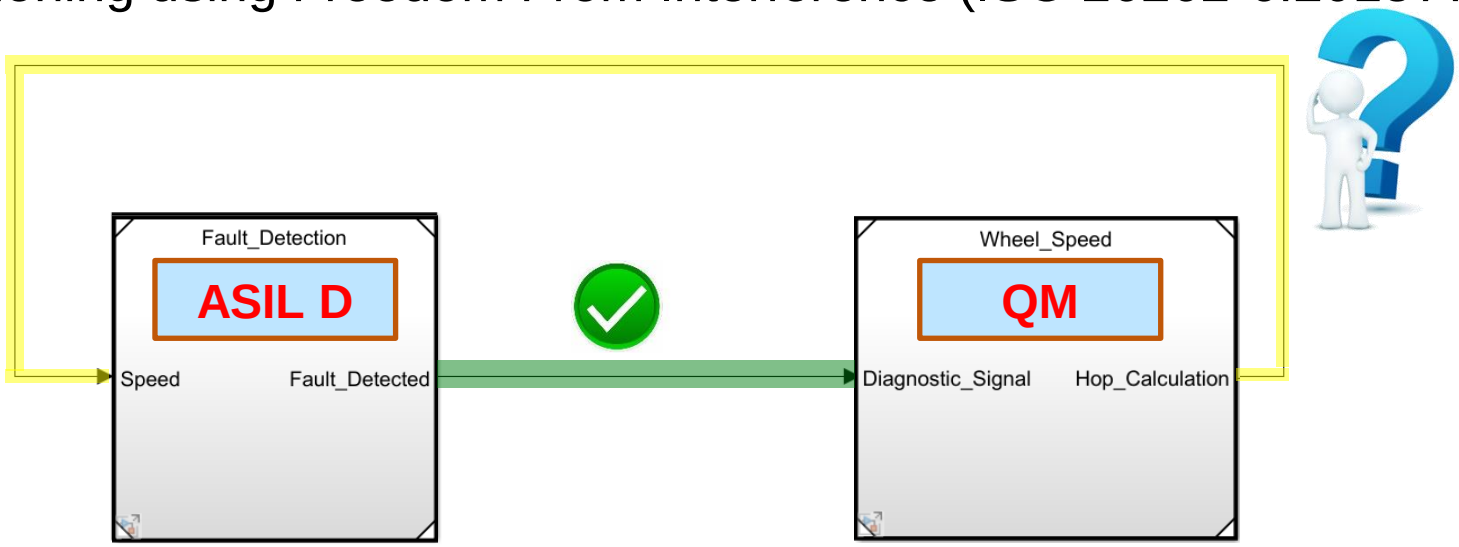


Verification Plots

Lack top-down architectural design approach

- “Bottom up” (legacy) vs “top down” (functional safety) approach

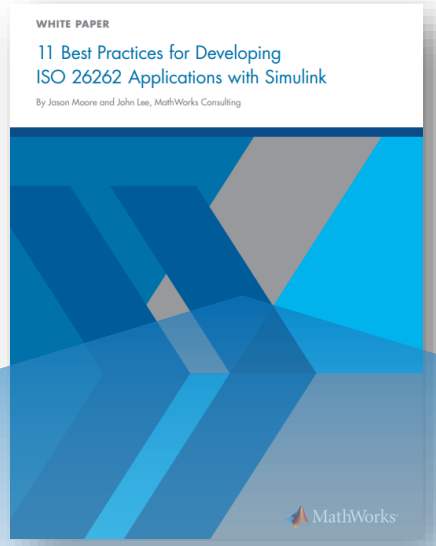
- Concepts:
  - Static and Dynamic architecture description (ISO 26262-6:2018 Clause 7.4.5)
  - Criteria for coexistence of elements (ISO 26262-9: 2018 Clause 6)
  - Safety-Oriented analysis (ISO 26262-9:2018 Clause 8)
  - Analysis of dependent failures (ISO 26262-9:2018 Clause 7)
  - Software partitioning using Freedom From Interference (ISO 26262-6:2018 Annex D)



Lack top-down architectural design approach

Perform architectural review – Freedom From Interference

- ISO 26262-6 (Annex D)
  - Timing and execution
  - Memory
  - Exchange of information



Model architecture

- Use model reference for unit-level models
- Pick a strategy for grouping units into features
- Split ASIL and QM levels at the top level of the model
- Eliminate algorithm content at the integration level
- Use model metrics to monitor unit complexity

Signal routing and definition

- Group bus signals by ASIL, feature, and rate
- Pass only necessary signals to units
- Optimize placement of signal and parameter objects
- Protect data exchanged between ASILs

Code generation configuration

- Determine a code placement strategy
- Use different name tokens for shared utilities

Poor tool qualification awareness

- How do you qualify a tool?

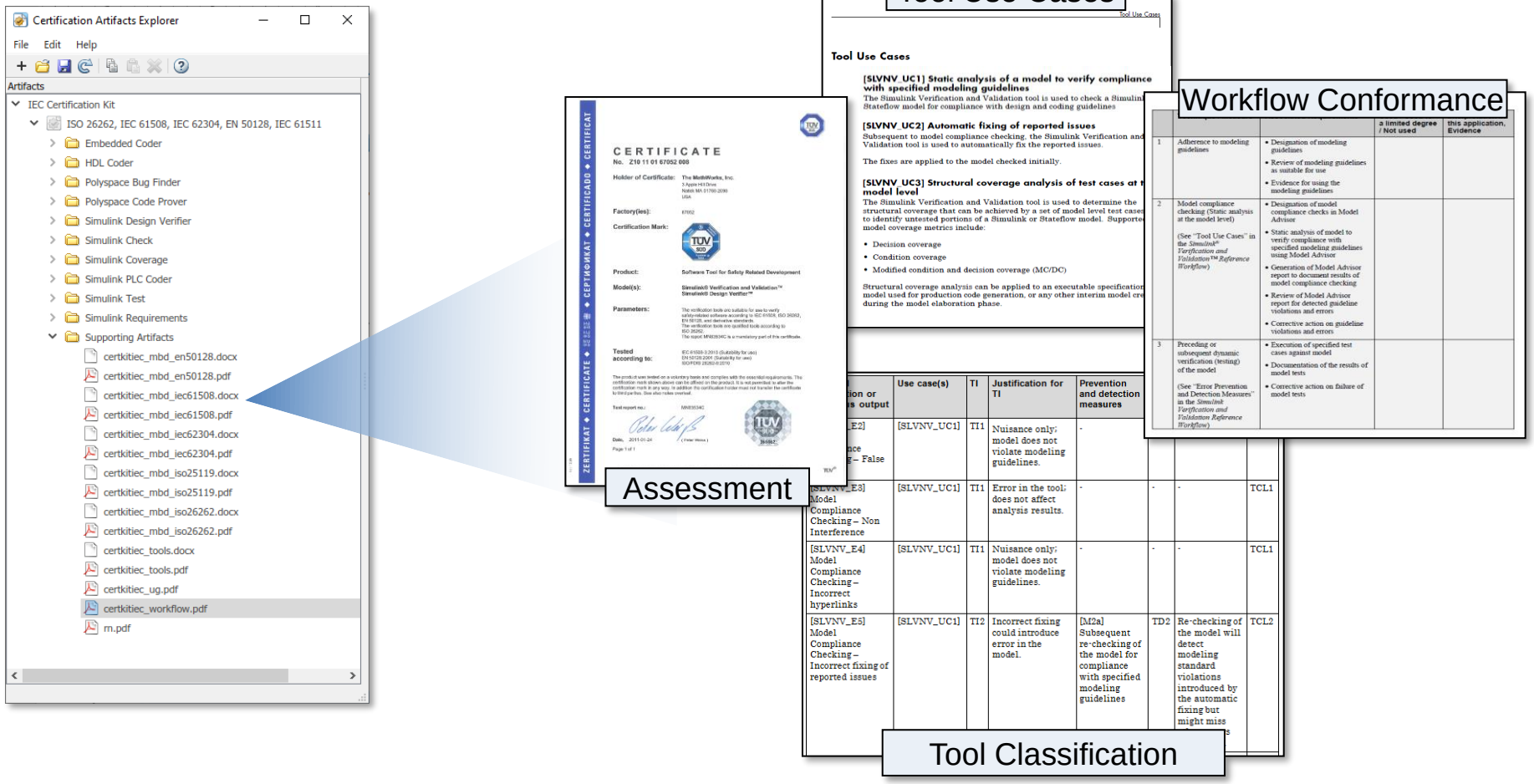
Method	TCL 2				TCL 3			
	ASIL A	ASIL B	ASIL C	ASIL D	ASIL A	ASIL B	ASIL C	ASIL D
1a Increased confidence from use	++	++	++	+	++	++	+	+
1b Evaluation of the tool development process	++	++	++	+	++	++	+	+
1c Validation of the software tool	+	+	+	++	+	+	++	++
1d Development in compliance with a safety standard	+	+	+	++	+	+	++	++

Source: ISO 26262:2018 Clause 11.4.6.1, Tables 4&5

+ … Recommended ++ … Highly recommended

Poor of tool qualification awareness

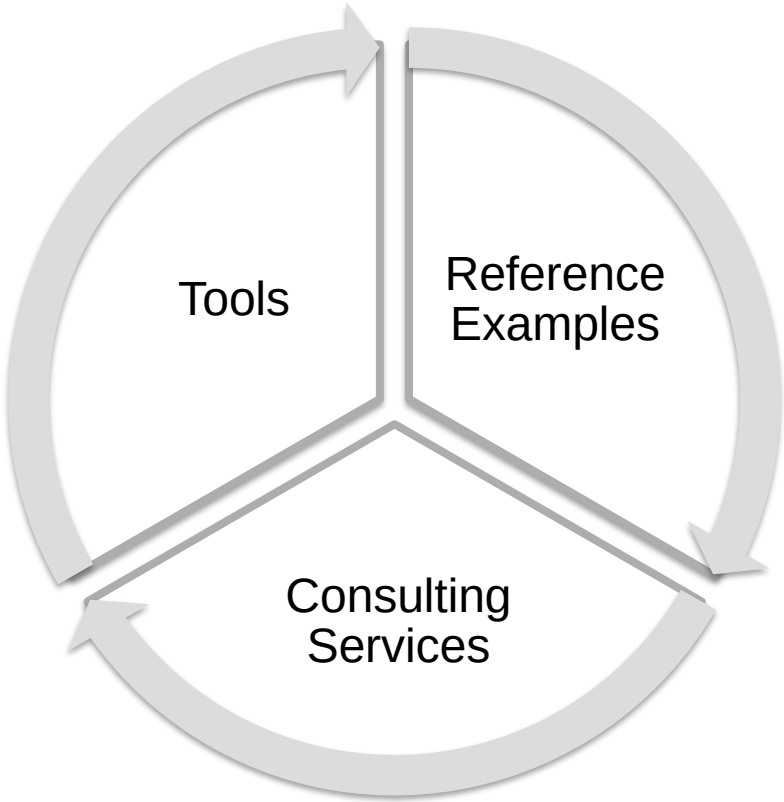
Leverage tool qualification kit from supplier



- Utilize vendor provided tool qualification content as much as possible
- Have a plan to qualify any custom tools or use cases not covered by the tool vendor

Summary

- Process not clearly defined or documented
  - Document process: from ISO Requirement down to Detail Work Instructions
- Lack top-down architectural design approach
  - Review architecture – Implementation of Freedom From Interference
- Poor tool qualification awareness
  - Leverage tool qualification content from tool vendor



Presenter contact info and poll questions

Please contact me at jasonm@mathworks.com with questions

- Poll question : How would you rate your organizations activity on ISO 26262
  a. No interest
  b. Some interest but no activity
  c. Currently implementing an ISO 26262 compliant process
  d. Struggling to implement an ISO 26262 compliant process
  e. Already fully ISO 26262 compliant

- If you would like to an individual follow-up, please let us know in the WebEx poll area.