TÜVIT

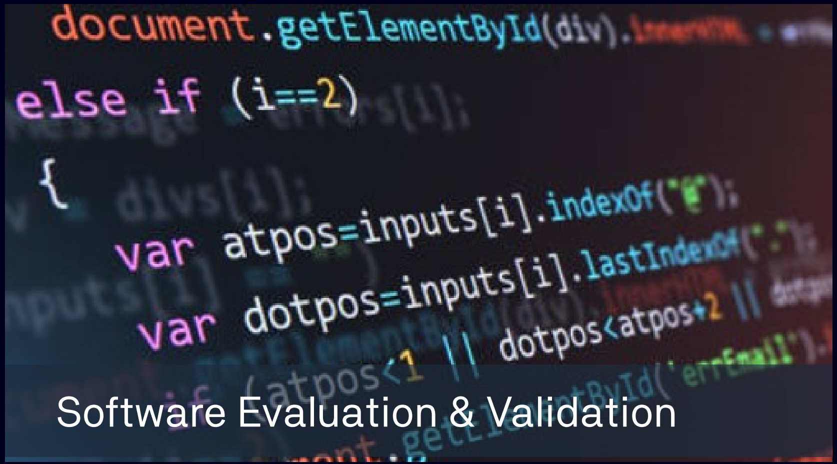
Prüfbarkeit KI-basierter Anwendungen im Kontext IT-Sicherheit

Omnisecure 2024, Berlin

Vasilios Danos | TÜVIT | 23. Januar 2024

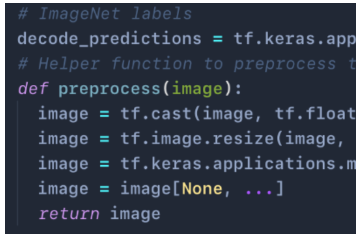
TÜVIT

TÜVNORD



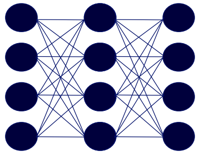
Motivation

Impact of Safety and Security



Common Software

(traceable, clear rules, source code review etc.)

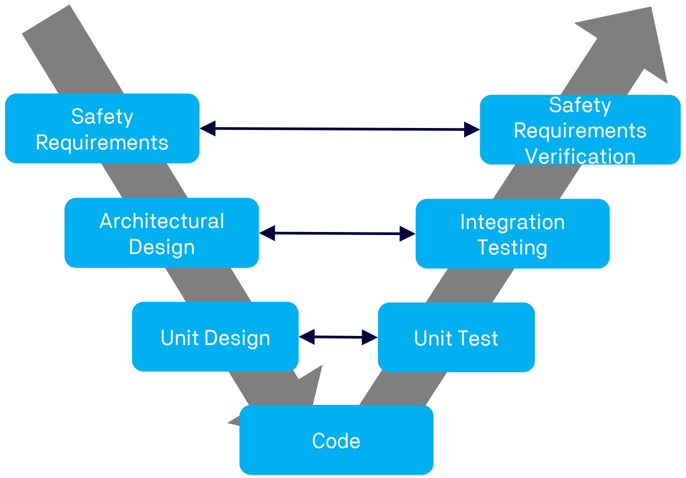


AI (Deep Neural Nets)

(What has been learned? Uncertain robustness, etc.)

Motivation

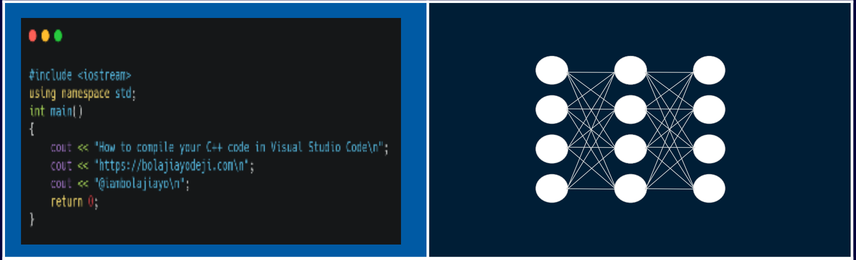
Impact of Safety and Security



Category	Description	Example
ASIL D	Highest safety requirements	Emergency Braking
ASIL C	High safety requirements	Electronic Stability Control
ASIL B	Moderate safety requirements	Cruise Control
ASIL A	Basic safety requirements	Automatic Headlight Activation
QM	Standard quality management	Infotainment System

Embedding of AI-based functionality within common **saftey and security** frameworks

“Common” Software vs. Data driven (ML)



Traceability	“Black Box”
Verifiable	Non-linear, Probabilistic
Testing	Uncertain Significance of Results
“Best Practices”	Lack of Standards

TÜVIT

Approach

BSI Project P538 & P532

Project Partners: BSI, ZF Friedrichshafen, TUVIT

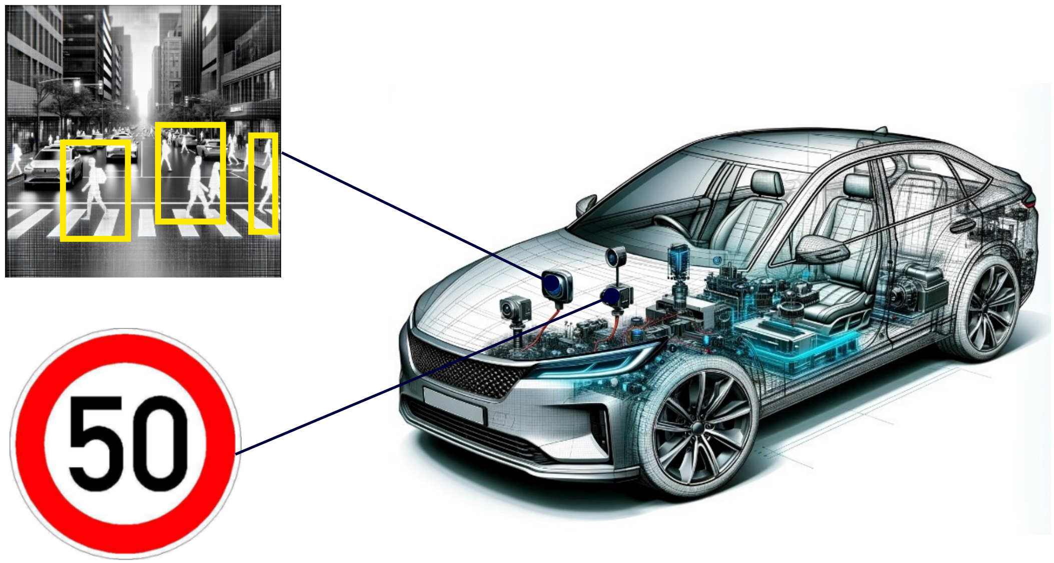
- Project P538 AIAuditMobilityPrep --> finished
- Project P532 AIAuditMobility --> active

Key Facts

- ✓ Development of concepts and methods for **auditing and testing** AI systems in vehicles
- ✓ Focusing on a modular, **use-case-centric** approach
- ✓ Definition of a set of requirements
- ✓ Evaluating the approach on **real-life** scenarios
- ✓ Introducing the results into **standardisation/regulation** bodies

Approach

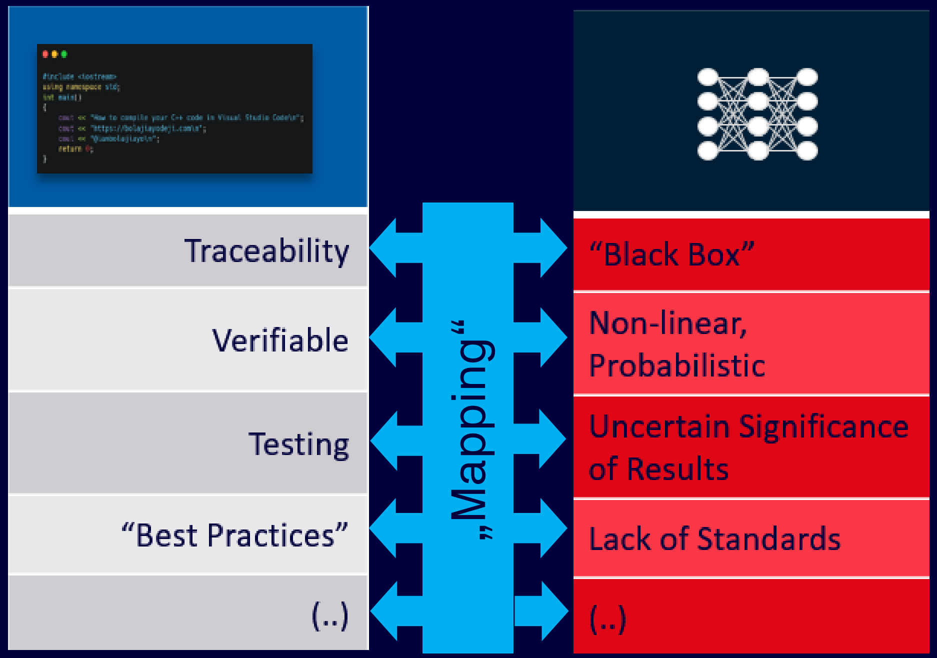
Actual Use-Cases



- ISO/SAE 21434
- UNECE R 155
- UL 4600
- **ISO 26262**

<-- Requirements to comply

How to Map („translate“) Requirements (e.g. ISO 26262) to the AI-Domain?



TÜVIT

Approach

Requirements from ISO 26262

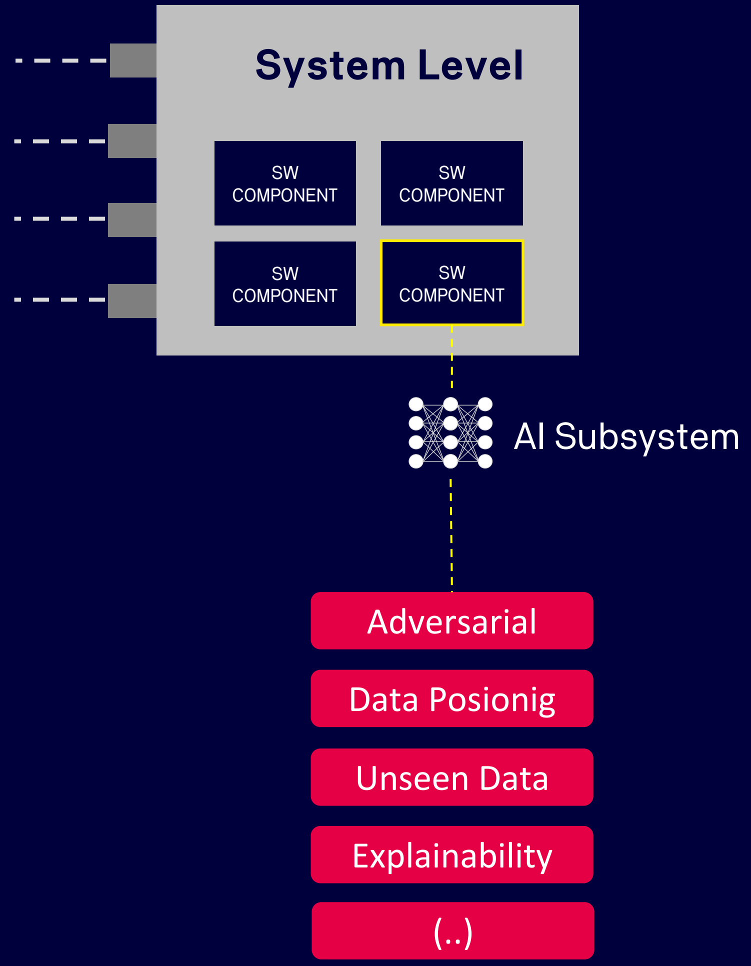
Methods for deriving test cases for integration testing

	Method	ASIL A	ASIL B	ASIL C	ASIL D
DI1	Analysis of requirements	++	++	++	++
DI2	Analysis of external and internal interfaces	+	++	++	++
DI3	Generation and analysis of equivalence classes for hardware-software integration	+	+	++	++
DI4	Analysis of boundary values	+	+	++	++
DI5	Error guessing based knowledge or experience	+	+	++	++
DI6	Analysis of functional dependencies	+	+	++	++
DI7	Analysis of common limit conditions, sequences and sources of dependent failures	+	+	++	++
DI8	Analysis of environmental conditions and operational use cases	+	++	++	++
DI9	Analysis of field experience	+	++	++	++

ASIL recommendations for software testing

	Method	ASIL A	ASIL B	ASIL C	ASIL D
IV1	Requirements-based test	++	++	++	++
IV2	Interface test	++	++	++	++
IV3	Fault injection test	+	+	++	++
IV4	Resource usage evaluation	++	++	++	++
IV5	Back-to-back comparison test between model and code, if applicable	+	+	++	++
IV6	Verification of the control flow and data flow	+	+	++	++
IV7	Static code analysis	++	++	++	++
IV8	Static analyses based on abstract interpretation	+	+	+	+

System / Subsystem



TÜVIT

Requirements

Generic Requirements for AI-specific Functionality

EXAMPLE: System Level Requirements

	Method	ASIL A	ASIL B	ASIL C	ASIL D
DI1	Analysis of requirements	++	++	++	++
DI2	Analysis of external and internal interfaces	+	++	++	++
DI3	Generation and analysis of equivalence classes for hardware-software integration	+	+	++	++
DI4	Analysis of boundary values	+	+	++	++
DI5	Error guessing based knowledge or experience	+	+	++	++
DI6	Analysis of functional dependencies	+	+	++	++
DI7	Analysis of common limit conditions, sequences and sources of dependent failures	+	+	++	++
DI8	Analysis of environmental conditions and operational use cases	+	++	++	++
DI9	Analysis of field experience	+	++	++	++

:

	Method	ASIL A	ASIL B	ASIL C	ASIL D
RS1	Resource usage test	o	+	++	++
RS2	Stress test	o	+	++	++
RS3	Test for interference resistance and robustness under certain environmental conditions	++	++	++	++

Derived Requirements for the AI-System (Excerpt)

Requirement			Risk level			
ID	Description	Type	ASIL A/ Low	ASIL B/ Medium	ASIL C/ High	ASIL D/ Very high
1	The environmental context shall correspond to the operational design domain (ODD).	ASIL	+	++	++	++
2	The communication, interfaces, signals, etc. between different components shall be coordinated.	ASIL	+	++	++	++
3	The sensor setup shall be similar to the development/training setup.	Additional	+	++	++	++
4	The requirements for AI subsystems shall apply to the entire system (if applicable).	Additional	++	++	++	++
5	The adequate performance shall be guaranteed for a certain timeframe after initial deployment.	ASIL	+	+	++	++
6	The performance on key performance indicators (KPIs) shall be as high as possible.	Additional	+	++	++	++
7	The performance shall be compliant to the allowed worst-case error.	ASIL	++	++	++	++
8	The performance shall be reproducible in the real environment for operation.	ASIL	+	++	++	++
9	The feedback of the system shall be tracked while in operation.	ASIL	o	+	++	++
10	The performance shall be corrected when critical errors occur after deployment.	ASIL	+	+	+	++
11	The system state shall be tracked in a reproducible way while in operation.	Additional	+	+	++	++

:

TÜVIT

Requirements

Generic Requirements for AI-specific Functionality

EXAMPLE: ISO 26262 design, development and testing of SW

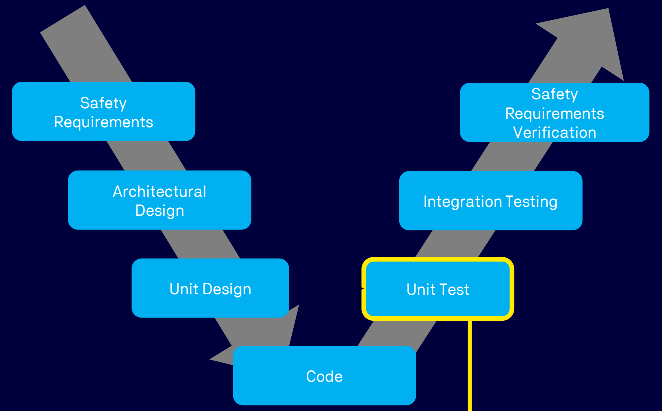
ASIL recommendations for deriving test cases for software unit testing (DU)

	Method	*ASIL A*	*ASIL B*	*ASIL C*	*ASIL D*
DU1	Analysis of requirements	++	++	++	++
DU2	Generation and analysis of equivalence classes	+	++	++	++
DU3	Analysis of boundary values	+	++	++	++
DU4	Error guessing based on knowledge or experience	+	+	+	+

ASIL recommendations for software unit verification (UV) taken from

	Method	*ASIL A*	*ASIL B*	*ASIL C*	*ASIL D*
UV1	Walk-through	++	+	o	o
UV2	Pair-programming	+	+	+	+
UV3	Inspection	+	++	++	++
UV4	Semi-formal verification	+	+	++	++
UV5	Formal verification	o	o	+	+
UV6	Control flow analysis	+	+	++	++
UV7	Data flow analysis	+	+	++	++
UV8	Static code analysis	++	++	++	++
UV9	Static analyses based on abstract interpretation	+	+	+	+
UV10	Requirements-based test	++	++	++	++
UV11	Interface test	++	++	++	++
UV12	Fault injection test	+	+	+	++
UV13	Resource usage evaluation	+	+	+	++
UV14	Back-to-back comparison test between model and code, if applicable	+	+	++	++

Derived Requirements for the AI-System (Excerpt)



	Method	*ASIL recommendation*			
		ASIL A	*ASIL B*	*ASIL C*	*ASIL D*
15	The AI model shall be implemented using mitigation strategies against robustness threats.	+	+	++	++
16	The AI model shall be verified with formal robustness verification techniques.	o	o	+	+
17	The robustness of the AI model shall be verified with empirical robustness estimation techniques.	+	+	++	++
18	The AI model shall be tested against out-of-distribution data.	++	++	++	++
19	Test cases at the boundary values of the input of the AI model shall be derived.	+	++	++	++
20	Test cases based on corner cases of the AI model shall be derived.	+	++	++	++
21	Test cases shall be derived through error guessing based on knowledge and experience of the system.	+	+	+	+

TÜVIT

Requirements

Applicability & Testability

Applicability

Estimates whether an requirement is suitable for the use-case

Requirement		Applicability	Concretization Effort
ID	Description		
1	The environmental context shall correspond to the operational design domain (ODD).	Simple	Minor • Suitable measurement for environmental context
2	The communication, interfaces, signals, etc. between different components shall be coordinated.	Simple	None

:

16	The AI model shall be verified with formal robustness verification techniques.	Unrealistic • How to verify complex systems	Major • Suitable verification techniques
17	The robustness of the AI model shall be verified with empirical robustness estimation techniques.	Simple	Major • Suitable estimation techniques • Suitable coverage of complete robustness
18	The AI model shall be tested against out-of-distribution data.	Simple	Major • Suitable definition of OOD data • Suitable coverage of complete OOD data

Testability

Estimates Test-Effort and the Test-Format (Evidence and/or Metric)

Requirement		Testability	Test	Comments
ID	Description			
1	The environmental context shall correspond to the operational design domain (ODD).	Medium	Evidence-based • Documentation on the environmental domain	
2	The communication, interfaces, signals, etc. between different components shall be coordinated.	Medium	Evidence-based • Interface implementation • Interface documentation	

:

16	The AI model shall be verified with formal robustness verification techniques.	Low	Metric-based • Verification metrics	• Infeasible for complex models (<10^5 neurons and 6 layers); For more information, see (1)
17	The robustness of the AI model shall be verified with empirical robustness estimation techniques.	Medium	Metric-based • Robustness metrics • Assessing suitable metrics • Assessing thresholds	
18	The AI model shall be tested against out-of-distribution data.	High	Metric-based • Performance metric on out-of-distribution data	

Requirements

Examples

Requirement		***Recommendation***
ID	***Description***	***ASIL A/Low***
18	The AI model shall be tested against out-of-distribution data.	++

Requirement		***Recommendation***
ID	***Description***	***ASIL C/High***
7	The performance shall be compliant to the allowed worst-case error.	++

TÜVIT

Thank You

Vasilios Danos
Head of AI Security & Trustworthiness
T.: +49 201 8999-560
E-Mail: v.danos@tuvit.de