SCN-SG Safety Concept Notation Study Group

SCDL Safety Concept Description Language

Concept project proposal for SCDL Next Gen.

Expand SCDL application to SOTIF area

Tomoyoshi Murata
SCN-SG SOTIF-Extension Sub-Working Group

Sep. 19th, 2023

ASAM Association for Standardization of Automation and Measuring Systems

Agenda

1	Motivations for the SCDL Evolution of SOTIF
2	Overview of the SOTIF-Extension Sub-Working Discussions
3	Introduction to the Case Studies
4	Discussion points for the SOTIF-Extensions
5	Summary

About the SOTIF-Extension Sub-Working Group in SCN-SG

- Established in April/2022
- Members :

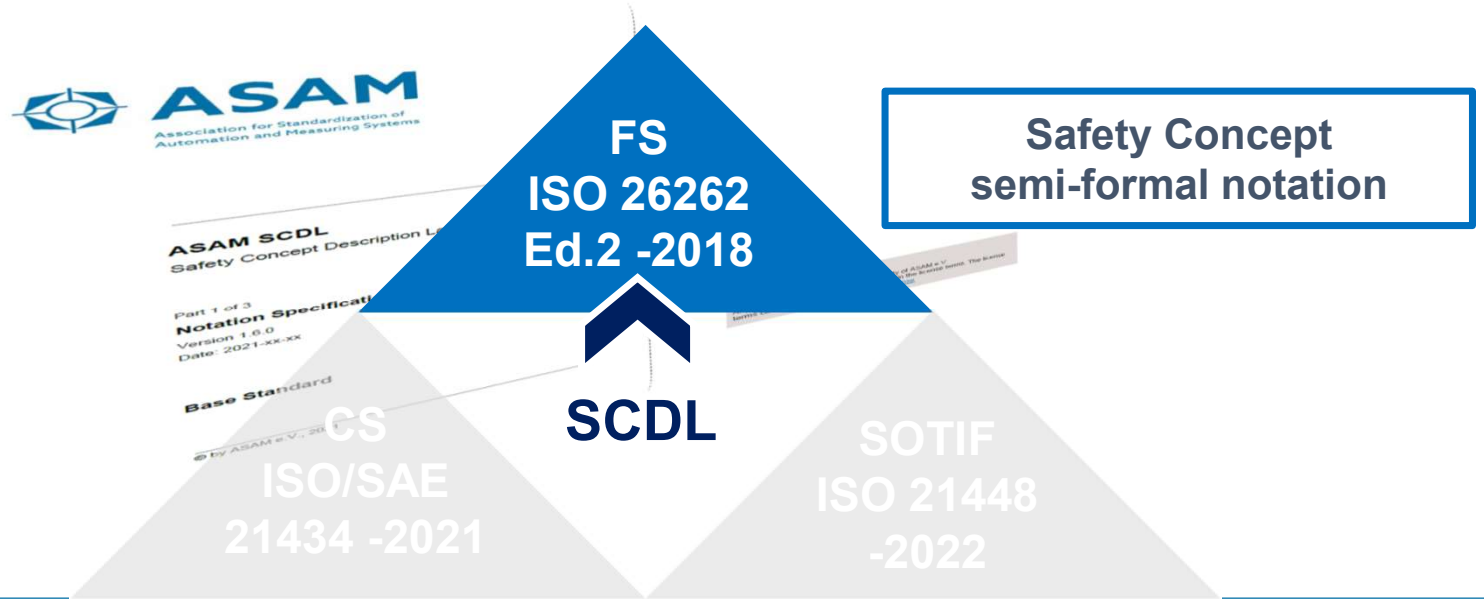
Toyokazu Ogasawara(ODEP & Gunma University)
Sou Kitajima (JARI)
Tsutomu Koshiyama (NISSAN)
Yoshikazu Sasaki (MARELLI)
Kodai Seki (TOYOTA)
Kenji Taguchi (UL Japan)
Nobuyuki Tanaka (OTSL)
Tetsuya Todo (DENSO)
Hideaki Nishihara (AIST)
Tomoyoshi Murata (JARI)
Shuuhei Yamashita (DNV)
Tomoki Yoshida (HINO)
Sandra Watanabe (JARI)

Deputy chair : Akira Takada (DNV)
Chair : Misako Imai (DNV)

ASAM SCDL 1.6.0

SCDL
Safety Concept
Description Language

- ASAM Standard SCDL 1.6.0: 2021, is dedicated to the notation of the safety concept which is previously published ISO 26262
- SCDL is frequently used in the Functional Safety development field



ASAM

1. Motivations for the SCDL Evolution of SOTIF

What is 'SOTIF' ?

Safety **O**f **T**he **I**ntended **F**unctionality

Absence of unreasonable risk from insufficient specifications, and performance limitations or reasonably foreseeable misuse of the intended functionality.

- **ISO26262 (Functional Safety) was initially published as a safety-related standard in response to failures, while ISO21448 (SOTIF) is a standard that specifically addresses the safety of intended functions not adequately addressed by the former.**

ISO 26262（Functional Safety）	ISO 21448（SOTIF）
Published December 2018(2nd) **Hazard Cause: Malfunctions** ・Failure of hardware parts ・Malfunctions due to human errors such as specification errors, design errors, etc.	Published July 2022 **Hazard Cause: Insufficient of normal functions, non-malfunctions** ・Performance limitations and influences from external environment ・Misuses

1. Motivations for the SCDL Evolution of SOTIF

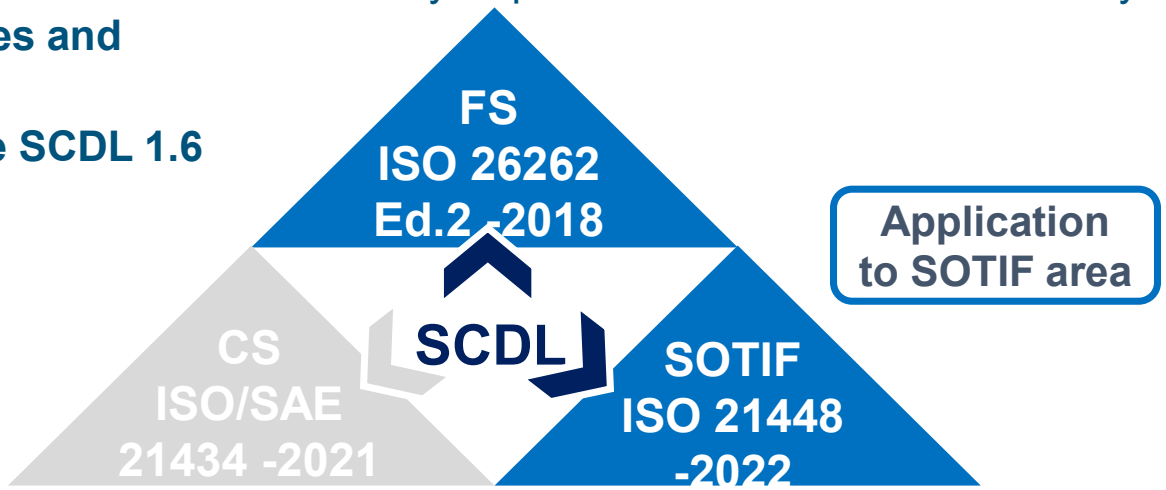
- **Background**

  To implement the safety design, the ISO 21448 (SOTIF) and ISO 26262 (Functional Safety) activities must be coordinated; while ASAM SCDL1.6 was standardized for Functional Safety, SOTIF was not included.

- **Activity content**

  To hypothesize and verify possible effective and efficient interactions by discussing the architecture of the intentional functions referenced in the safety requirements of the functional safety.

  ➡ **Study case-studies and consider additional requirements for the SCDL 1.6**



2. Overview of the Sub-Working Discussions

- The case study of SCDL application aims to assess its functional insufficiencies and ongoing efforts to enhance its Intended Functionals. This result serves as item definition for Functional Safety, allowing for an example of safety analysis and the specification of safety mechanisms to be discussed concurrently.
- Conduct studies to explore possibilities for extending the SCDL. This includes examining functional requirements for functional performance, process requirements for intended functionals development, and various types of analyses, all of which are currently under discussion.

SOTIF/Functional Safety communication:
Work together in SR-based interactions

Functional Safety-Lifecycle：
Impact analysis and redevelopment are necessary after changing the Intended Functional architecture

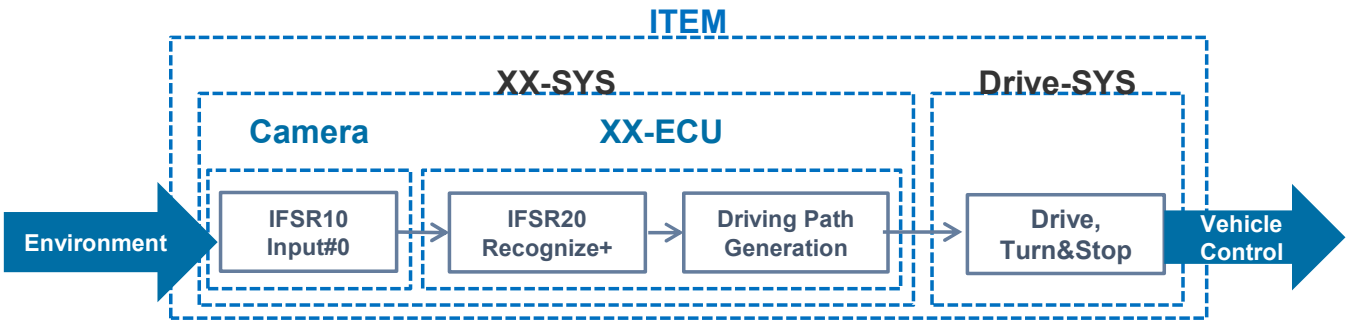
SOTIF-Lifecycle：
Scenario-based validation tests and field operations create large feedback loops

SCDL Safety Concept Description Language

3. Introduction to the Case Studies

A case study is conducted by following the steps outlined below

1. Definement of the virtual AD system “XX-SYS”
2. Construction of the Intended function architecture in accordance with the safety requirements（SOTIF）
3. Creation of a functional safety concept based on the architecture（Functional Safety）
4. Update ~~of~~ the architecture with feedback from V&V results（SOTIF）
5. Update the functional safety concept（Functional Safety）



Virtual AD system “XX-SYS”

Definition of Safety Goal

U-SG： Universal - Safety Goal

*Hazardous events common to Functional Safety and SOTIF are collectively defined as ‘U-SG’.

*In the case of U-SG, the AD system will not fail

ASAM

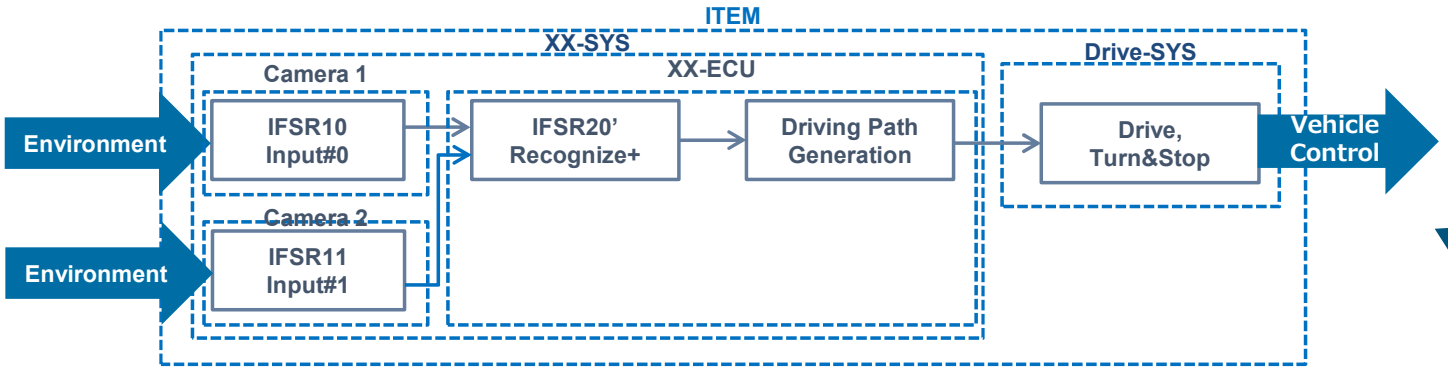
3. Introduction to the Case Studies

1. **Definement of a virtual AD system “XX-SYS”**
2. **Construction of the Intended function architecture according to the safety requirements（SOTIF）**
   - Analyze the architectural hazards associated with the Intended functions
   - Incorporate safety measures into the architecture and finalize the architecture

Design a safety measure considering the following causes of safety requirements violations:

（1）False positives/false negatives camera image information → Implement camera multiplexing

（2）Insufficiency of AI machine learning in Fusion → Implement a relearning ML algorithms

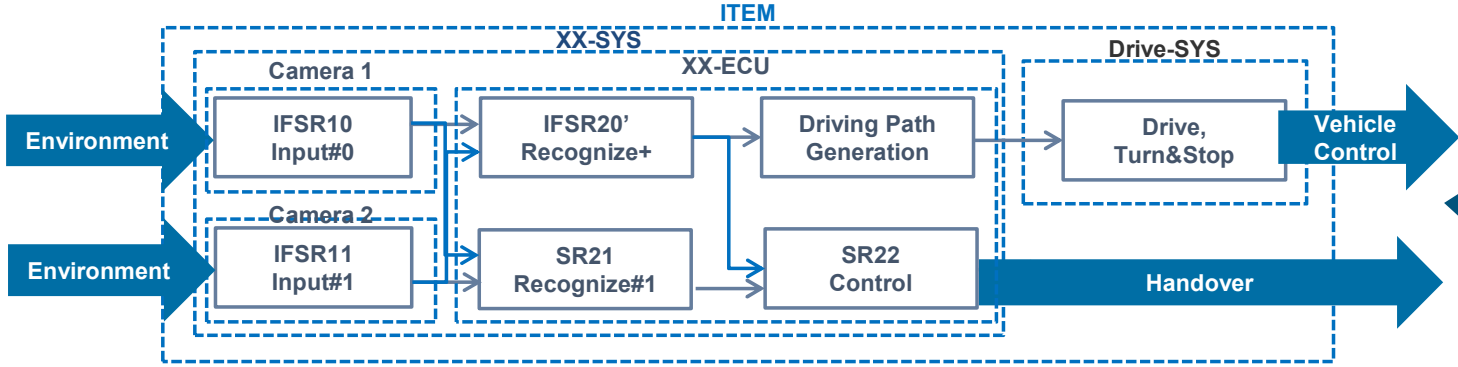


Discuss how to clearly express the ability to detect ‘functional insufficiency’ or ‘triggering conditions’ during SOTIF Hazard analysis.
【How to handle performance requirements in SCDL】

3. Introduction to the Case Studies

3. Creation of a functional safety concept based on the architecture (Functional Safety)

- Safety analysis from Functional Safety point of view
  - Camera failure can be mitigated through SOTIF's safety mechanisms
  - The recognition function necessitates a single-fail handling approach, such as a SW based calculations, to ensure functional safety
- Updated architecture based on safety analysis results
  - Prompts the driver to operate if there is a discrepancy in the calculation results in the ECU



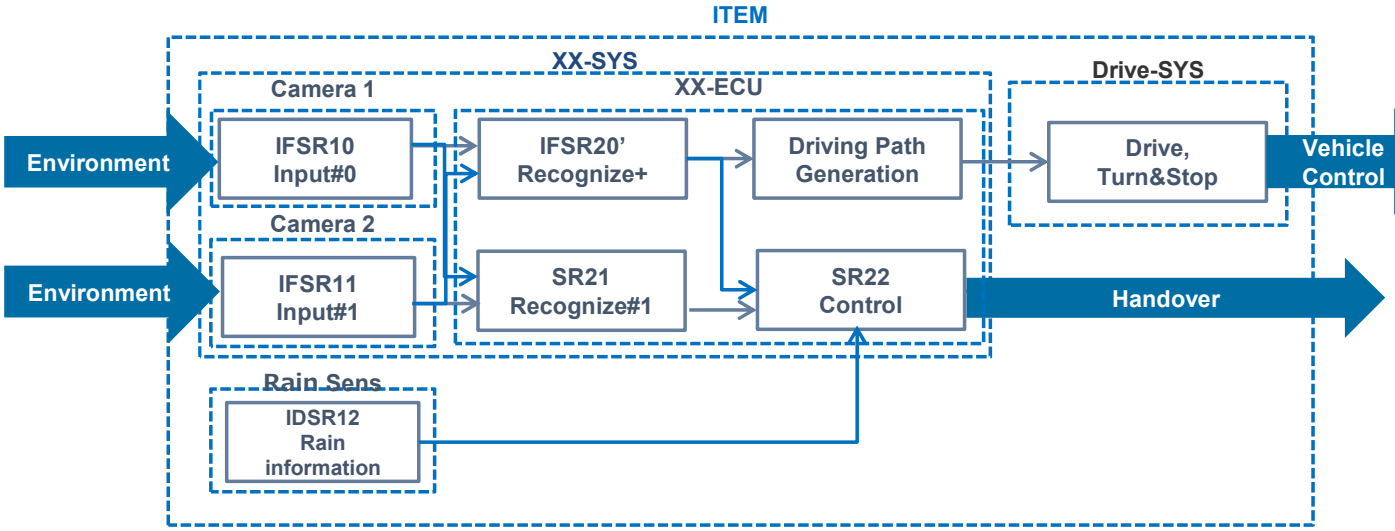
Discuss how to express process safety measures (Includes learning reinforcement and performance enhancement) on the architecture, such as measures for ML in IFSR20'.
【How to handle process requests in SCDL】

3. Introduction to the Case Studies

4. Update the architecture with feedback from the V&V results（SOTIF）

- **Complete the design and development based on the constructed concept and implement V&V**
- **V&V detected a camera sensing performance deterioration due to raindrops while driving in the rain**

→Considered safety measures: Add IFSR12 and send raindrop information to IFSR22



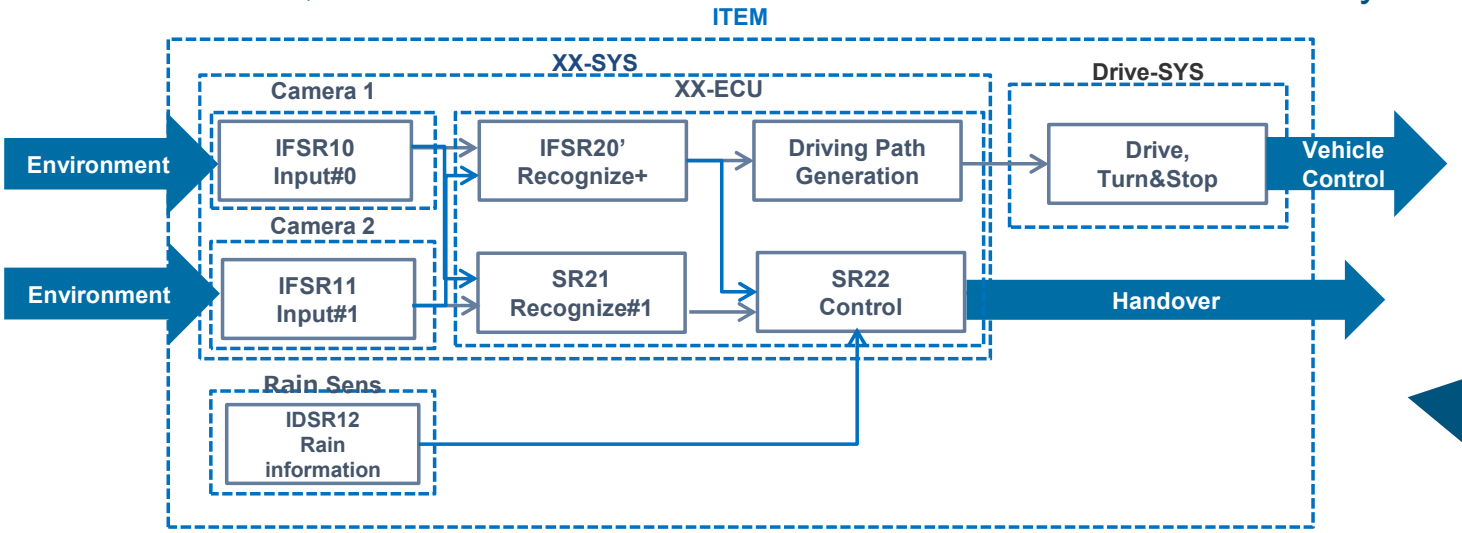
＊SR22 has changed the ID from SR22 to IFSR22 in order to add support for intentional function requirements and redefine the function.

＊In this case study material, sensor fusion of rain sensor information is not used, and raindrop information is used only for judgment of raindrop exceeding a certain amount and subsequent control and handover.

3. Introduction to the Case Studies

5. Update the functional safety concept（Functional Safety）

- Update ‘Safety Concept’ based on the following SOTIF updates
- Safety analysis of the functional safety：
  - IFSR12, IFSR22 are Intended Functions added for SOTIF（analysis targets）



Discuss about special expressions are necessary for countermeasures against the SOTIF safety mechanism (i.g. IFSR12).
【How to express failure response of SOTIF safety mechanism】

4. Discussion Points for the SOTIF-Extensions

Extracted and discussed ‘HLR’ (High Level Requirement) as a discussion point from the case study

① How to handle performance requirements in SCDL

➡Regarding the expression of SOTIF intended functionals, such as detecting functional insufficiency or triggering conditions, consider if unambiguous expressions can be incorporated into SCDL.

➡Consider how to express ‘performance requirements’ etc. on SCDL to analyze performance inadequacies.

② How to handle the process requests in SCDL

➡Consider the explanation of ‘process safety measures’ including learning reinforcement and performance improvement, required for the SOTIF safety argument using formats such as structural diagrams, tables, etc.

③ How to express failure response of SOTIF safety mechanism

➡Regarding the expression of countermeasures for SOTIF safety mechanisms.

4. Discussion Points for the SOTIF-Extensions

Knowledge gained from HLR discussion results

① How to handle performance requirements in SCDL

➡Expressing SOTIF-like intended functions such as ‘functional insufficiency’ or ‘triggering conditions’ detection is difficult in SCDL because it is not designed for modelling funtions.

➡Intended-functional architecture can be expressed in SCDL, but Non-functional requirements cannot be expressed

② How to handle process requests in SCDL

➡Process safety measures are also non-functional safety requirements, so it is difficult to express them in architecture.
（Can be expressed using constraints, Process safety measures are also being considered by other sub-working groups)

③ How to express failure response of SOTIF safety mechanism

➡Respond as a functional safety measure

5. Summary

This case study covered the following cases:

- SOTIF-SM coverage of Functional Safety's-SM
- Addition of Functional Safety's-SM to the SOTIF-architecture
- Exploring interactions between SOTIF and Functional Safety when updating the intended function's architecture

Current Results of the Case study:

- The intentional functional architecture of SOTIF can be represented in SCDL. Additionally, it can facilitate effective and efficient communication with functional safety activities.
- SOTIF countermeasures are non-functional requirements, and the results of the countermeasures cannot be expressed in architecture. Relatedly, the addition of new symbols and grammar as SCDL is not currently visible.
- Continue to explore effective and efficient communication between the analysis results of non-functional requirements and the architecture.

Direction:

- Organize specific proposals for SCDL Next Generation, including case studies!
- Collaborate on common issues with other sub-working groups !

Thank you for your attention