＜SCDL：SOTIF拡張＞
SCDLを利用したSOTIF対応についての検討

今井　美紗子
ＤＮＶビジネス・アシュアランス・ジャパン（株）
S&S事業部　機能安全部
テクニカルエキスパート

Agenda

1. SCDLをSOTIFで利用するモチベーション
2. SOTIF拡張の検討
3. SOTIF拡張のケーススタディ紹介
4. SCDL拡張利用のためのポイント整理
5. まとめと課題

ASAM SCDL 1.6.0

- **ASAM Standard SCDL 1.6.0 :2021は、ISO 26262機能安全の安全コンセプトを表記するための標準としてリリースされた**
- **SCDLは、機能安全規格の適用開発で利用される**

1. SCDLをSOTIFで利用するモチベーション

■ SOTIF規格（ISO 21448）の国際規格化

SOTIF：Safety Of The Intended Functionality

意図した機能の 仕様の不十分性，性能限界 又は 合理的に予見可能なミスユースから生じる不合理なリスクがないこと（意図した機能の安全性）

ISO 26262（機能安全）

2018年12月第2版発行
危険の原因：故障
・ハードウェア部品の故障
・仕様記載ミス、設計ミスなどのヒューマンエラーによる機能不全

ISO 21448（SOTIF）

2022年7月発行
危険の原因：正常機能の不十分、非故障
・性能限界や外部環境の影響
・ミスユース（誤用）

Application to SOTIF area

1. SCDLをSOTIFで利用するモチベーション

■ 背景

安全設計を進めるには、ISO 21448（SOTIF）とISO 26262（機能安全）の活動を連携させる必要があるが、ASAM SCDL1.6は機能安全のために標準化されたものでSOTIFに対応していない

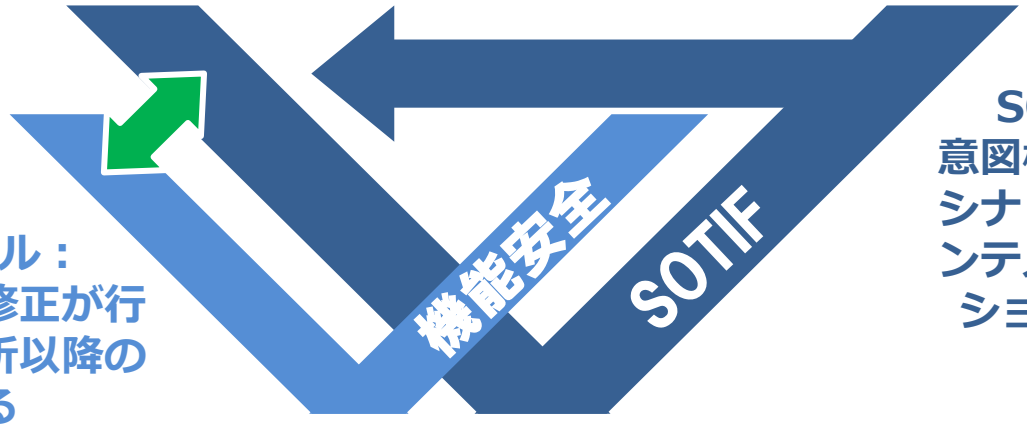
■ 検討方針

SOTIF意図機能のアーキテクチャをSCDLで示し設計検証活動に活用することで、SOTIFと機能安全の効果的・効率的なインタラクションが可能となると仮定し、ケーススタディで検討

- ➡ 2022年4月サブワーキンググループ設立
- ➡ SCDLでSOTIF意図機能を作成し、機能安全とのインタラクション等を研究

SOTIF/機能安全 communication:
安全ベースの効果的・効率的なインタラクションがあるとよい

機能安全ライフサイクル：
意図機能アーキテクチャ修正が行われた場合には，影響分析以降の再活動が必要になる



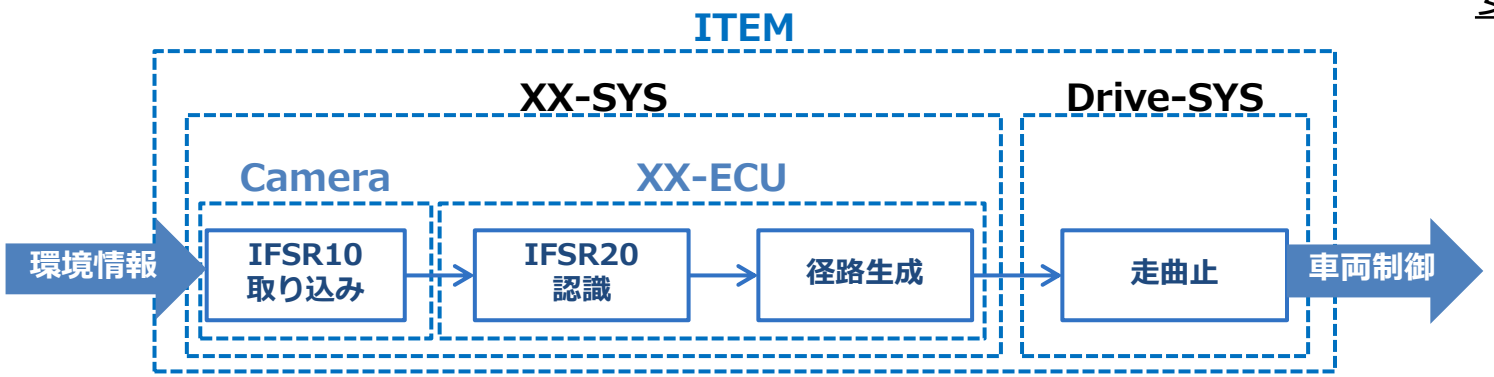
SOTIFライフサイクル：
意図機能の作りこみにおいて、シナリオベースのバリデーションテストやフィールドオペレーションによる大きなフィードバックループが発生

2. SOTIF拡張の検討

■ 検討概要（ケーススタディ）

- 意図機能の分析と対処の段階におけるSCDL適用のアーキテクチャを作成
- SOTIF対応で更新したアーキテクチャを機能安全のアイテム定義とみなして対応
- 機能要求の性能要件、意図機能の作りこみにおけるプロセス要件記述などに関して分析表現含めたSCDL拡張の課題を検討する

■ 検討アイテム



仮想の自動運転システム　“XX-SYS”

安全目標の定義

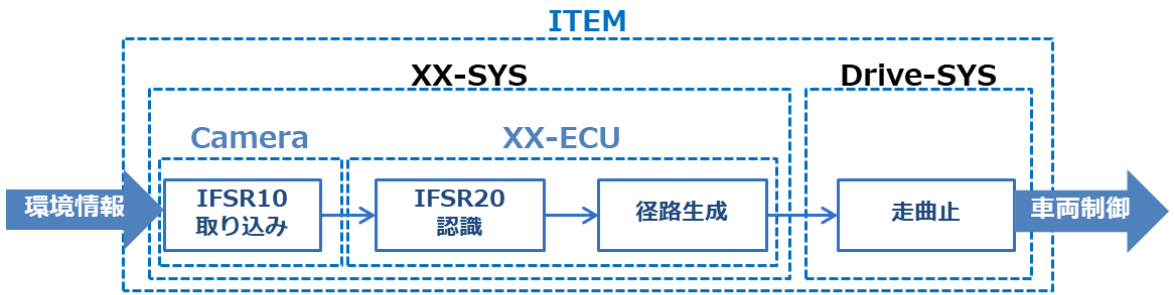
U-SG： Universal - Safety Goal
ITEMのハザードは機能安全／SOTIF共通として扱うことにし、その和集合をU-SGと定義

※本ケーススタディのU-SGは『自動運転を失敗しない』と設定した

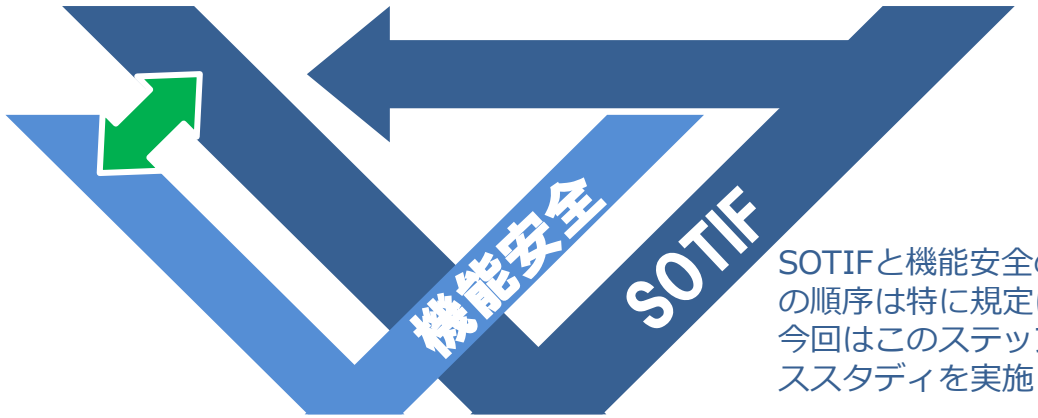
3. SOTIF拡張のケーススタディの紹介

ケーススタディを以下のステップで行い、議論ポイントを抽出する

1. **仮想の自動運転システム“XX-SYS”を定義する**
2. **意図機能安全アーキテクチャ構築を安全要求ベースで行う（SOTIF）**
3. **これに基づく安全コンセプトを作成する（機能安全）**
4. **バリデーションからのフィードバックによりアーキテクチャの更新をする（SOTIF）**
5. **安全コンセプトの見直しを行う（機能安全）**



仮想の自動運転システム “XX-SYS”



SOTIFと機能安全のプロセスの順序は特に規定はないが、今回はこのステップでケーススタディを実施した

3. SOTIF拡張のケーススタディの紹介

1. 仮想の自動運転システム“XX-SYS”を定義する

2. 意図機能安全アーキテクチャ構築を安全要求ベースで行う（SOTIF）

- 意図機能アーキテクチャのハザード分析を実施する
- 分析結果よりリスク対策を検討し、意図機能のアーキテクチャを完成させる

SOTIFの安全要求侵害の原因として以下が挙がったと仮定し、対策を検討した
（１）原因：カメラからの映像の偽陽性・偽陰性　→対策：SM追加
（２）原因：フュージョンの機能の不十分性　→対策：仕様変更等

※1：安全分析表は省略
※2：「IFSR20’認識＋」はIFSR11追加に伴う変更とプロセス要求の追加を表現

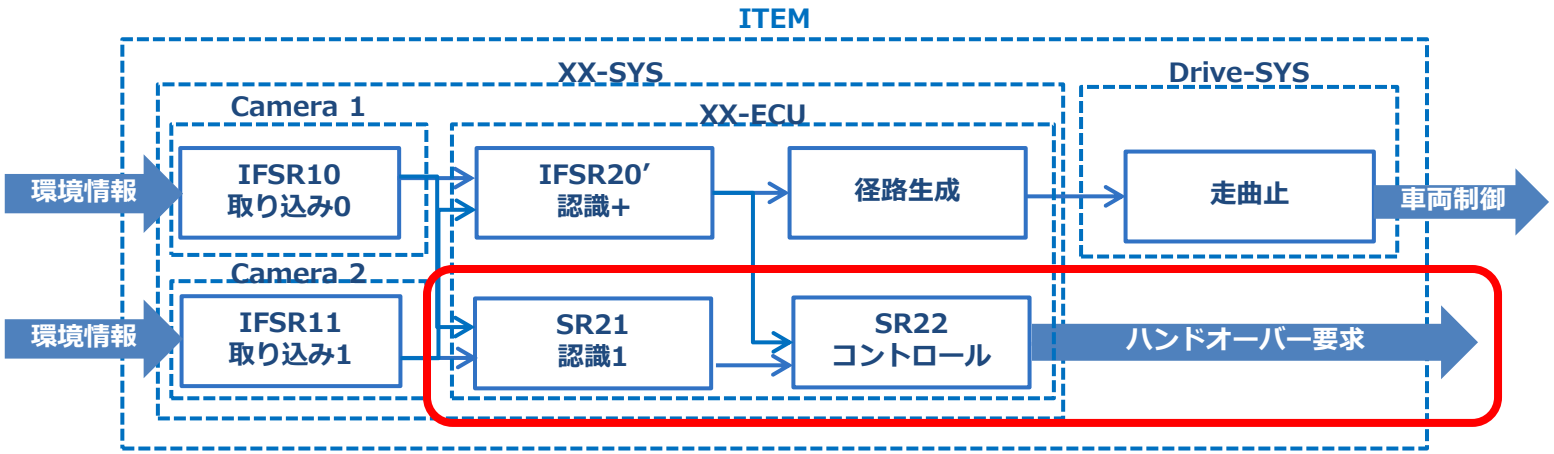
機能的不十分性又はトリガー条件を検知する機能の表現について、SCDL上で誤解を与えないような表現ができるかの議論
【SCDLにおける性能要求の取り扱い方法】

IFSR20’の中のMLの対策など、プロセス的安全方策(学習補強、性能向上を含む)をアーキテクチャ上で表現できる方法があるか
【SCDLにおけるプロセス要求の取り扱い方法】

3. SOTIF拡張のケーススタディの紹介

3. 意図機能アーキテクチャに基づく安全コンセプトを作成する（機能安全）

- 機能安全観点での安全分析を行う
  - カメラの故障はSOTIFで追加した安全機構で対応が可能と考える
  - 認識機能はソフトウェア演算などの機能不全原因の対応が必要となる
- 安全分析結果に基づきアーキテクチャを更新する
  - ECU内の演算は、認識に相違があった場合に運転手に必要な操作を促す等のハンドオーバーを行う



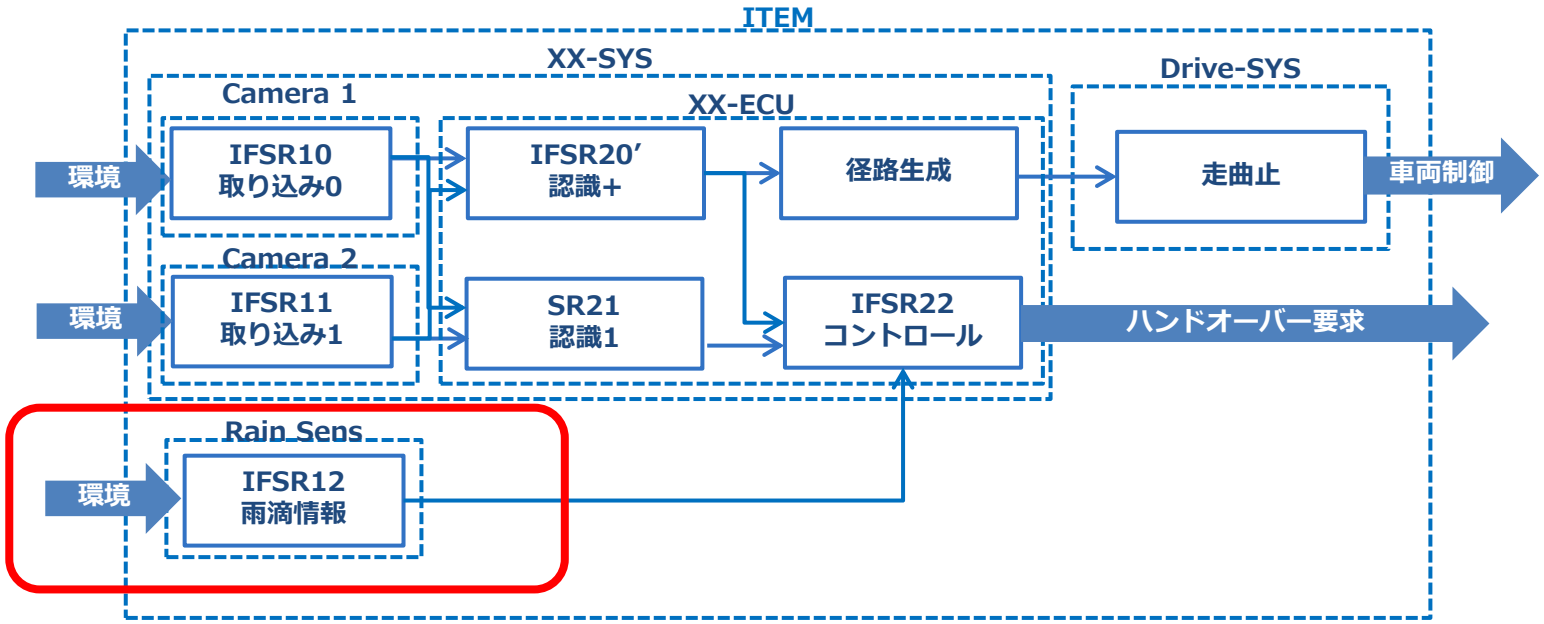
※3：ASIL、独立要求などの表現は省略

3. SOTIF拡張のケーススタディの紹介

4. バリデーションからのフィードバックでアーキテクチャの更新をする（SOTIF）

- 構築されたコンセプトに基づき設計開発が行われ、バリデーションを実施する
- バリデーション結果より、雨天走行時に雨滴によるカメラのセンシング性能低下の指摘あり

→安全方策を検討する：意図機能要求としてIFSR12を追加し雨滴情報をIFSR22へ送信する



＊本ケーススタディ題材においてはレインセンサ情報のセンサフュージョン等は使用しておらず、雨滴情報は雨滴の一定量超過の判断とその後のコントロールとハンドオーバーの為にのみ使用している。

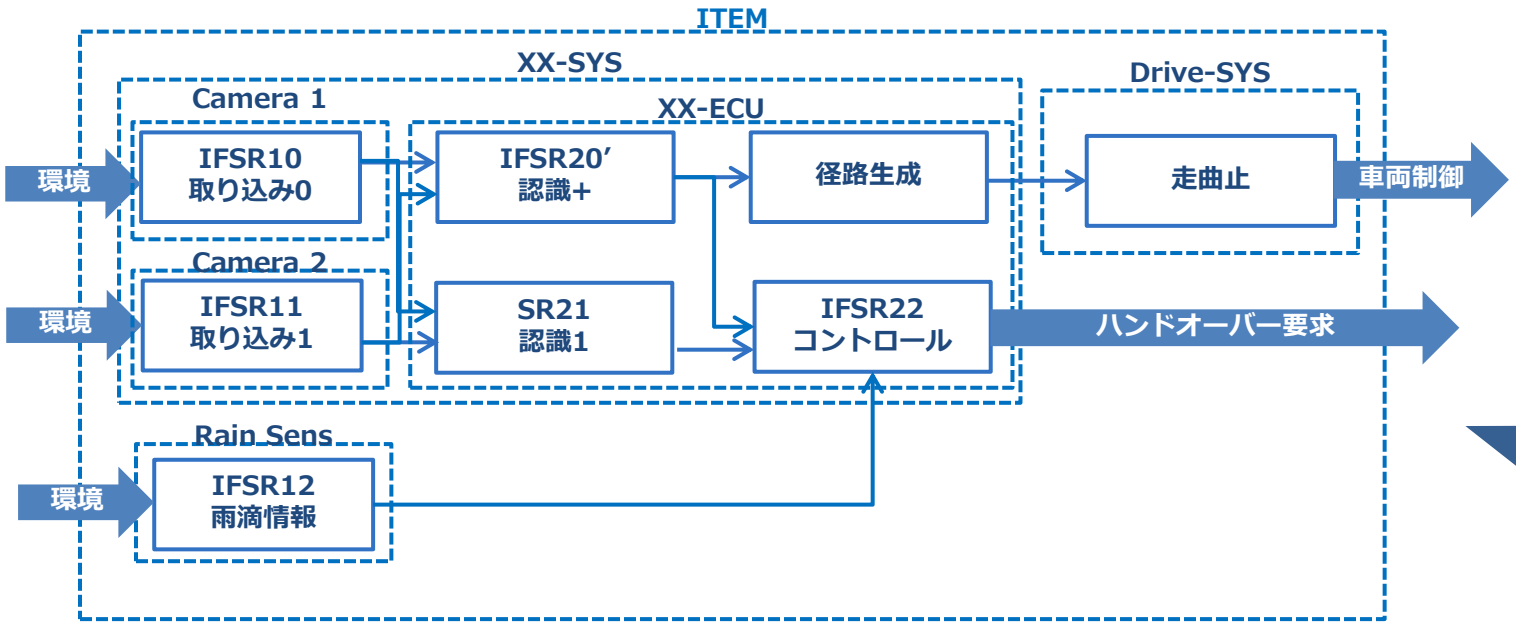
※4：「SR22」は意図機能要求の対応を追加し機能の再定義のため「IFSR22」へとIDを変更

3. SOTIF拡張のケーススタディの紹介

5. 安全コンセプトの見直しを行う（機能安全）

- SOTIF対策の更新が行われた結果を元に機能安全コンセプトの見直しを実施する
- 機能安全の安全分析の更新
  - 意図機能安全要求のIFSR12, IFSR22はSOTIF対応で追加されたが、

いずれも機能不全を起こすと新たなハザードの候補となる為、機能安全でのHARA再実施が必要



SOTIFの安全機構(IFSR12の故障)の故障した場合の対策を表現方法が特別に必要か
【SOTIFの安全機構の故障対応の表現方法】

※5：安全分析表は省略

4. SCDL拡張利用のためのポイント整理

ケーススタディで議論ポイントとしてHLR（High Level Requirement）を抽出

① SCDLにおける性能要求の取り扱い方法

- 機能的不十分性又はトリガー条件を検知するなどのSOTIF意図機能の表現について、SCDL上で誤解を与えないような表現ができるか検討する
- 性能の不十分性などを分析するためSOTIF性能要求をSCDL上で表現する方法を検討する

② SCDLにおけるプロセス要求の取り扱い方法

- SOTIF安全論証に必要なプロセス的安全方策(学習補強、性能向上を含む)の表現について(構造図、表など表現方法は未定)検討する

③ SOTIFの安全機構の故障対応の表現方法

- SOTIFの安全機構の故障(latent faultに相当)の場合の対処の表現について検討する

4. SCDL拡張利用のためのポイント整理

HLRの議論結果より得られた知見

① SCDLにおける性能要求の取り扱い方法

➡SOTIF意図機能の表現は非機能要求を含むため、全てをSCDLで表現できない

➡ただし、意図機能アーキテクチャとしてSCDLで表現することは可能

② SCDLにおけるプロセス要求の取り扱い方法

➡プロセス的安全方策も非機能的安全要求となるため、アーキテクチャでの表現は難しい

➡コンストレインツ標記は可能

③ SOTIFの安全機構の故障対応の表現方法

➡機能安全の対応として対処できれば問題ないため、SOTIF特有の課題ではない

5. まとめと課題

ケーススタディで取り上げた事例：

- SOTIFの安全機構が機能安全の安全機構を兼ねる事例
- SOTIFアーキテクチャに機能安全の安全機構を追加する事例
- 意図機能アーキテクチャが更新された時のSOTIFと機能安全のインタラクションや機能安全のイテレーションの事例

まとめ：

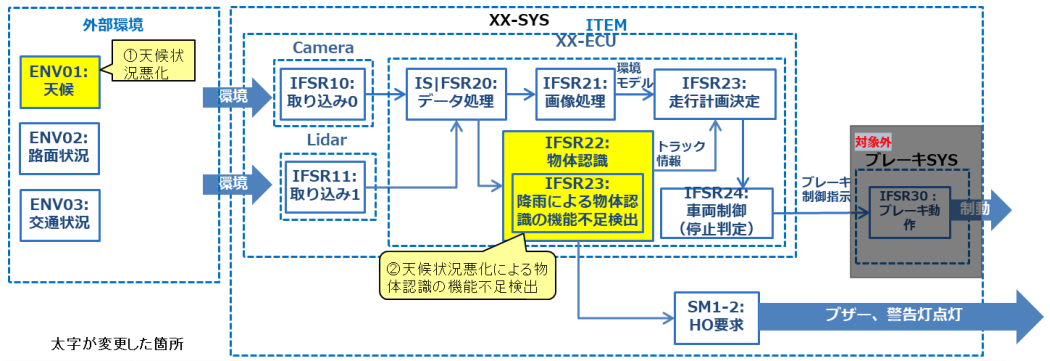
- SOTIFの意図機能アーキテクチャはSCDLで提示でき，機能安全活動との効果的で効率的な連携が可能である
- SOTIF対策は非機能要求が多く，SCDLでの表現は難しい

5. まとめと課題

今後の課題：

- 別の視点での要素を追加したケーススタディで新しい提案の可能性を継続検討する
- 分析結果の非機能要求とアーキテクチャとの効果的・効率的な連携について継続検討する
- これらによってSOTIFとしての活用事例を整理する

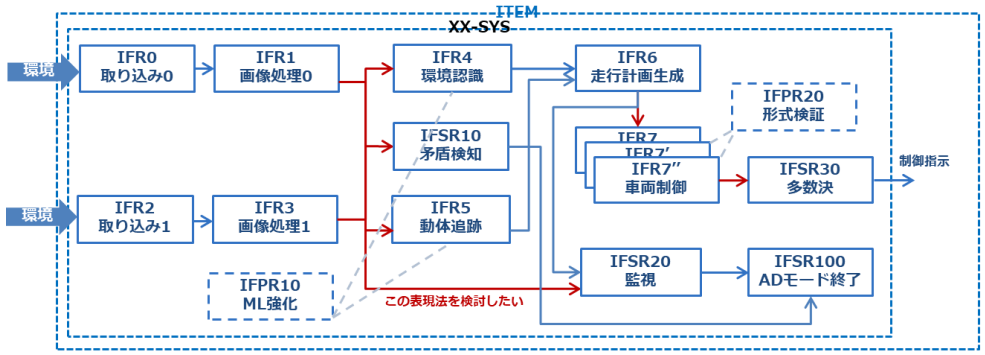
例１：外部環境を追加表現する事例



太字が変更した箇所

SR	SRV mode	誘発条件例（SOTIF）	SMr / SM	
IFSR21: 周辺環境モデルを生成する	停止標識を誤って環境モデルとして生成する	ML学習不足により看板と標識を誤って認識してしまう	-	SMr1：MLの再学習による性能向上
IFSR20：周囲の物体を認識する(認識処理する)	周囲の物体を誤って認識する	**天候状況悪化により物体を誤って認識する**	**SM1：天候状況悪化を検出した際にハンドオーバーを要求する**	~~SMr2：MLの再学習による性能向上~~

例2：機能要求に着目し他のSWGで検討している手法と連携する事例



IFR(意図機能要求)	SGVmode（SG0V）	安全方策	SOTIF-SR
IFR0 取り込み0	存在しない物標を取り込む	・IFR0,1(2,3)の一定レベルの機能性能低下についてはIFR2,3(0,1)とIFR4,5の縮化で性能維持する ・IFR0,1(2,3)の喪失が明らかであればIFR2,3(0,1)のみでIFR4,5でディグレードする. ・(喪失は不明だが)IFR0,1とIFR2,3が明らかに不一致であればこれを検出してADモード終了処理する	IFPR10 ML強化
IFR1 画像処理0	存在しない物標を生成する		IFSR10矛盾検知
IFR2 取り込み1	存在しない物標を取り込む		IFSR100 ADモード終了
IFR3 画像処理1	存在しない物標を生成する		
IFR4 環境認識	存在しない環境物標を認識する	・IF4,5,6をカバーする簡素なエンベロープ生成アルゴリズム（これもML）で監視し，一定以上の乖離が認められた場合はADモード終了処理する	IFSR20 監視
IFR5 動体追跡	存在しない動的物標を認識する		IFSR100 ADモード終了
IFR6 走行計画生成	誤った回避走行計画を生成する		
IFR7 車両制御	誤った回避制御指示を生成する	・2oo3構成で出力する(アルゴリズム・コード自体の確からしさは形式検証する)	IFR7′車両制御 IFR7″車両制御 IFSR30 多数決 IFPR20 形式検証

Thank you for your attention