

# Webアプリのテスト自動化エンジニアが、 1年半で車載器の自動テストアーキテクチャを語れるようになった話

2022年 3月11日

株式会社ベリサーブ  
中部・西日本オートモーティブ事業部  
石丸 温

■名前：石丸 温（いしまる あつし）

■経歴

2000年～2008年：電機メーカーにて組み込みソフト開発

2008年～2009年：ITベンチャーにてWebアプリ開発

2009年～2020年：コンサル系SIerにて組み込みソフト開発 + プロセス改善

2020年8月～ 現職ベリサーブ（1年半）

■現在担当している業務

自動車メーカー様向け 車載器品質保証業務

- 組み込み機器ロングランテスト：windowsアプリ + UART + フーム側スタブ
- 組み込みWebサーバーのテスト自動化：Selenium+Robotframework
- ネットワーク機能のシステムテスト自動化：Python+Expect+UART
- 組み込みソフト自動ビルドシステム導入：SCM(Git・Perforce)+Jenkins

近年、車載器開発においては、発売後のOTA(※)ソフト配信への対応など開発のスピードアップが求められており、テスト自動化は欠かせないものとなっています。

本講演では、車載器のテスト自動化の事例を紹介します。

自動車業界外から入ってきたエンジニアの目線で、

Webアプリの技術が活用できるポイントなどをお話しします。

新規に車載器開発に取り組む方向けに必要な知識の紹介なども行います。

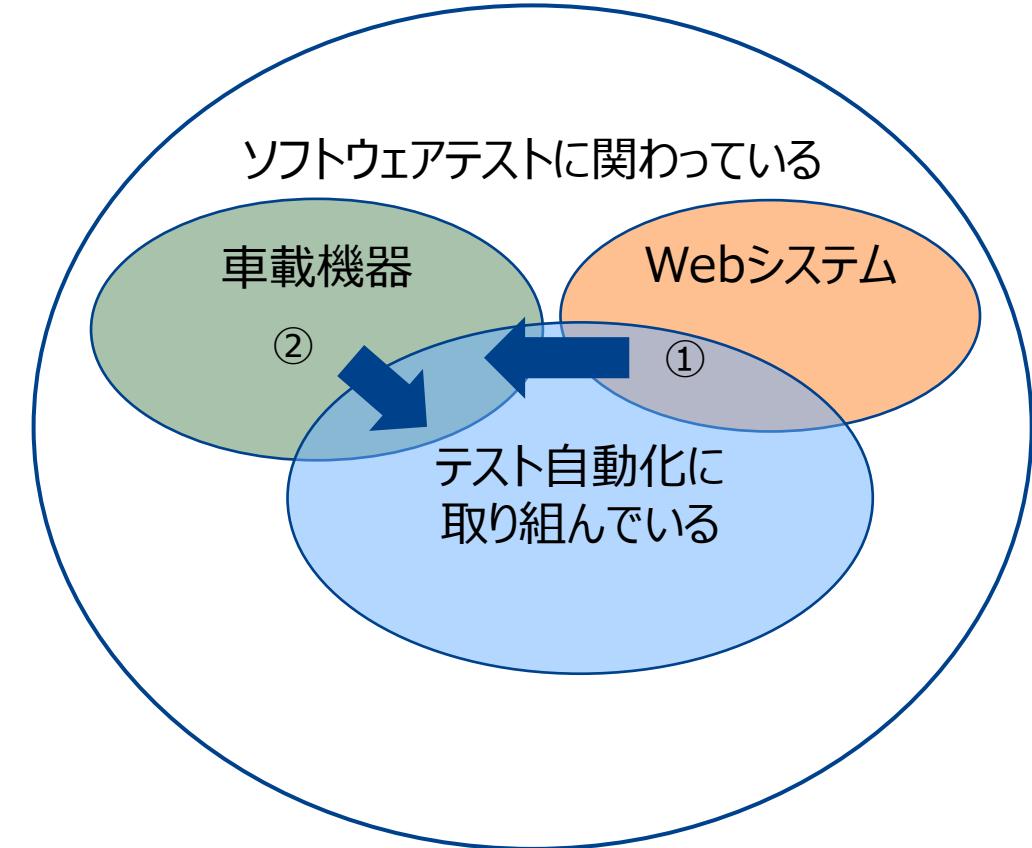
※OTA : Over The Airの略。データ送受信を無線通信で行うこと。

## ①Webアプリなどのテスト自動化に取り組んでいる方

車載器の開発は、外から見ると特殊なことをやっていると思っているのではないでしょうか。実はITシステムと技術的共通点が多く、自分の活躍の場を広げられる可能性があることを事例から知っていただきたい。

## ②車載器開発・テストに携わっている方

事例紹介をヒントに自動化の適用範囲拡大のきっかけにしていただきたい。



1. 車載器開発の変化
2. テスト自動化の概要
3. 車載器テスト自動化事例の紹介
  - 3-1. テスト実行の自動化
  - 3-2. テスト設計の自動化
  - 3-3. テスト管理の自動化
  - 3-4. ツールチェーンの自動化
4. まとめ

# 1. 車載器開発の変化

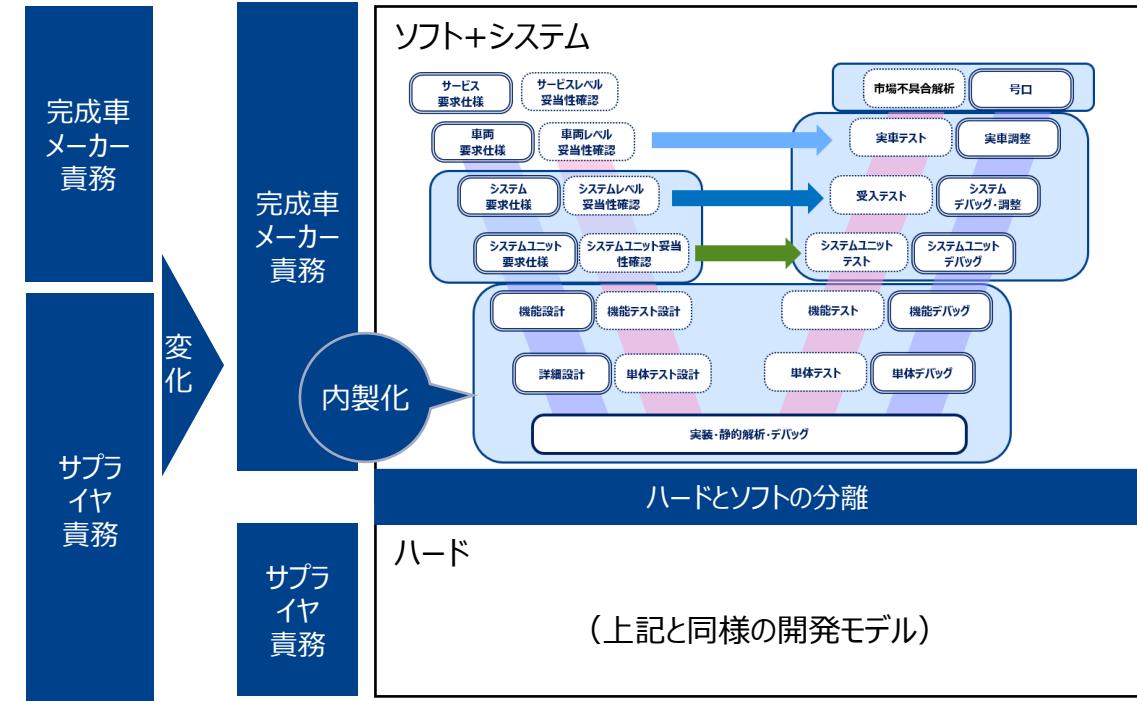
---

## ソフトウェア・ファースト

要求仕様の変化に迅速に対応し、  
OTA(Over The Air) でお客様に  
価値提供

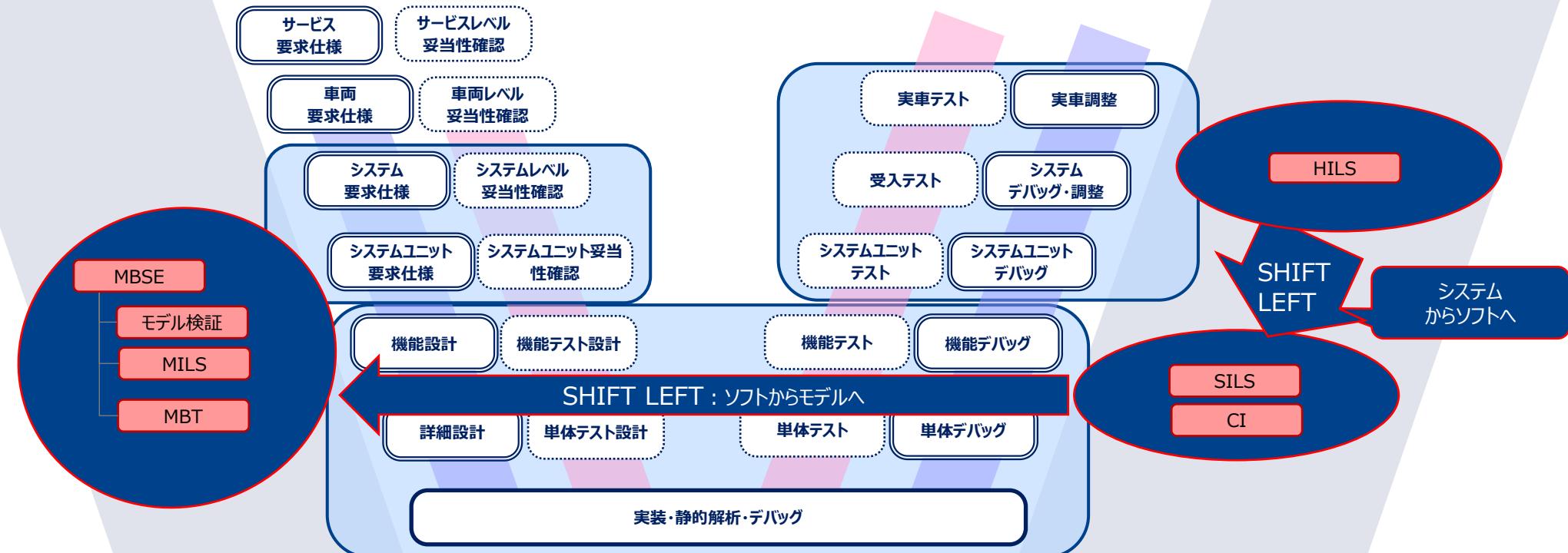
開発プロセスの変化

内製化



# SHIFT LEFT

## 使用技術



※HILS : Hardware In the Loop Simulation、SILS : Software In the Loop Simulation

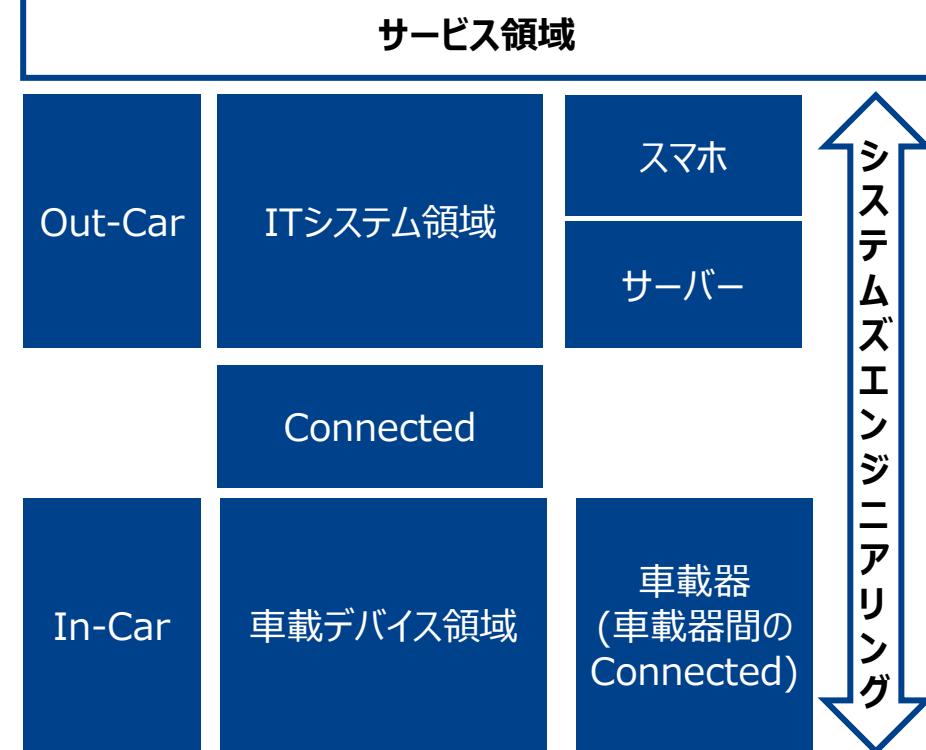
CI : Continuous Integration、MBSE : Model Based System Engineering

MILS : Model In the Loop Simulation

## システムズエンジニアリング

複雑化するConnectedサービス

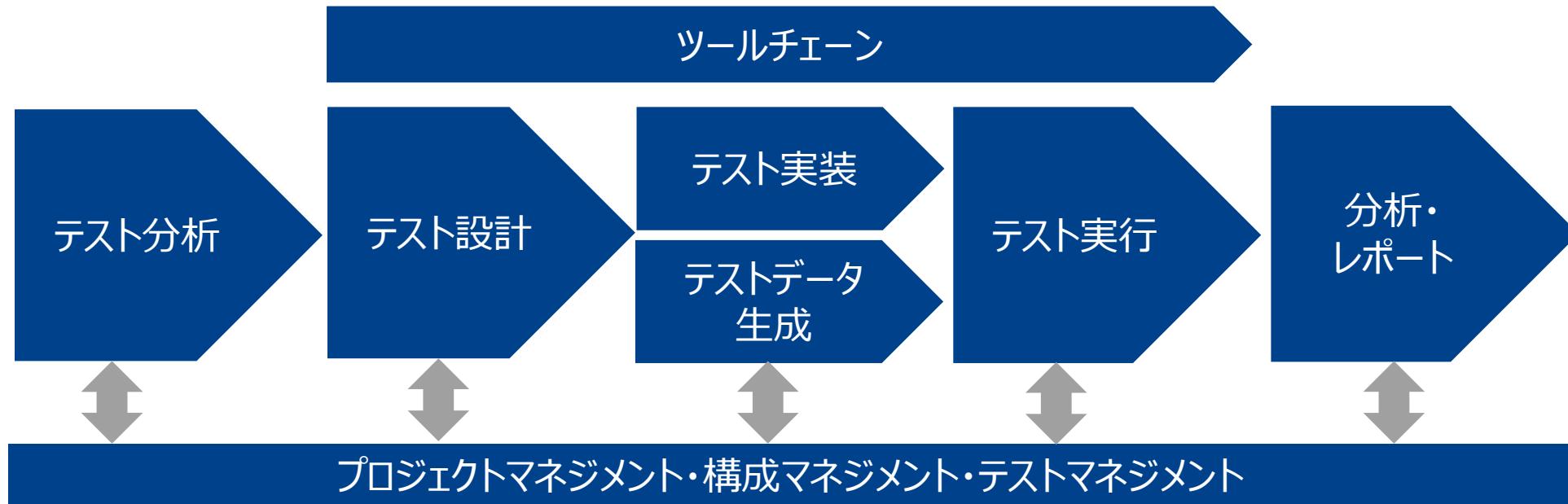
複数システムを俯瞰し、  
系統立てて考える  
システムズ エンジニアリング



## 2. テスト自動化の概要

---

テスト自動化は実行だけではない。全てのプロセスが対象。

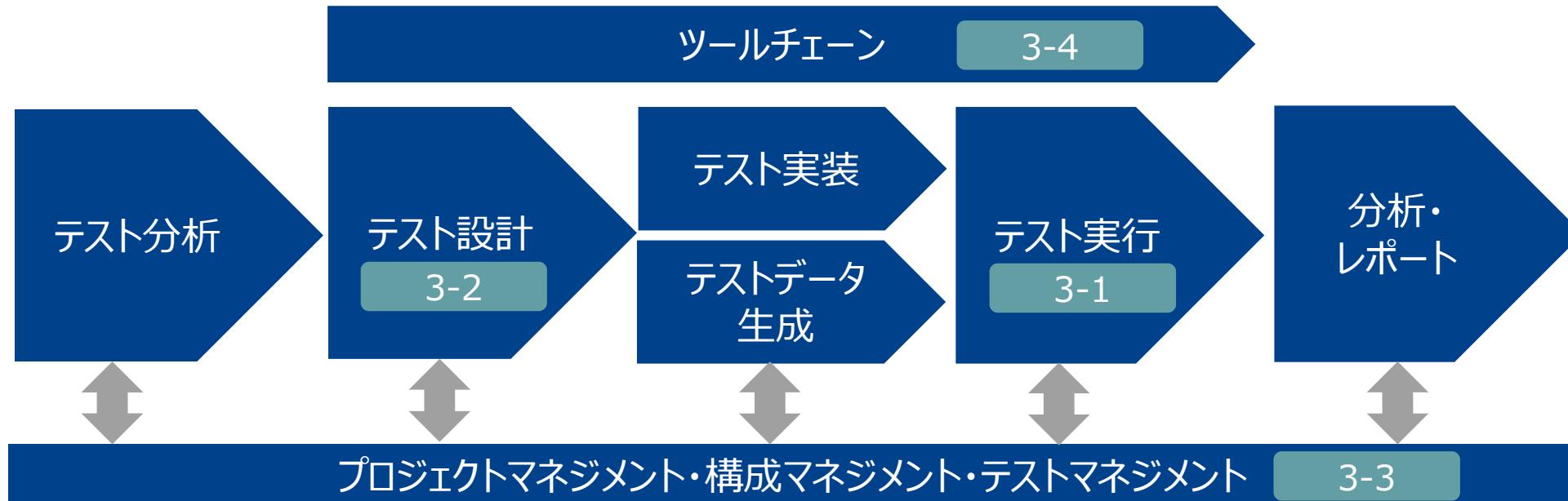


参考：JSTQB Advanced Levelシラバス「テスト自動化エンジニア」：[http://jstqb.jp/dl/JSTQB-Syllabus.Advanced\\_TAE\\_Version2016.J01.pdf](http://jstqb.jp/dl/JSTQB-Syllabus.Advanced_TAE_Version2016.J01.pdf)

### 3. 車載テスト自動化事例の紹介

---

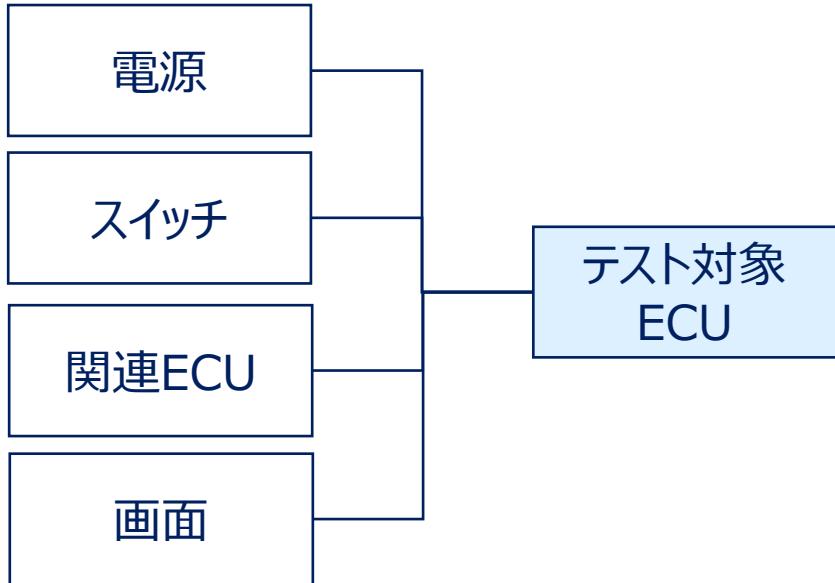
- 3-1. テスト実行の自動化(1) : ECUのHILSテスト
- (2) : 単体テスト
- 3-2. テスト設計の自動化 : Model Based Testing(MBT)
- 3-3. テスト管理の自動化 : ダッシュボード化
- 3-4. ツールチェーン自動化 : CIによるツールチェーン自動化



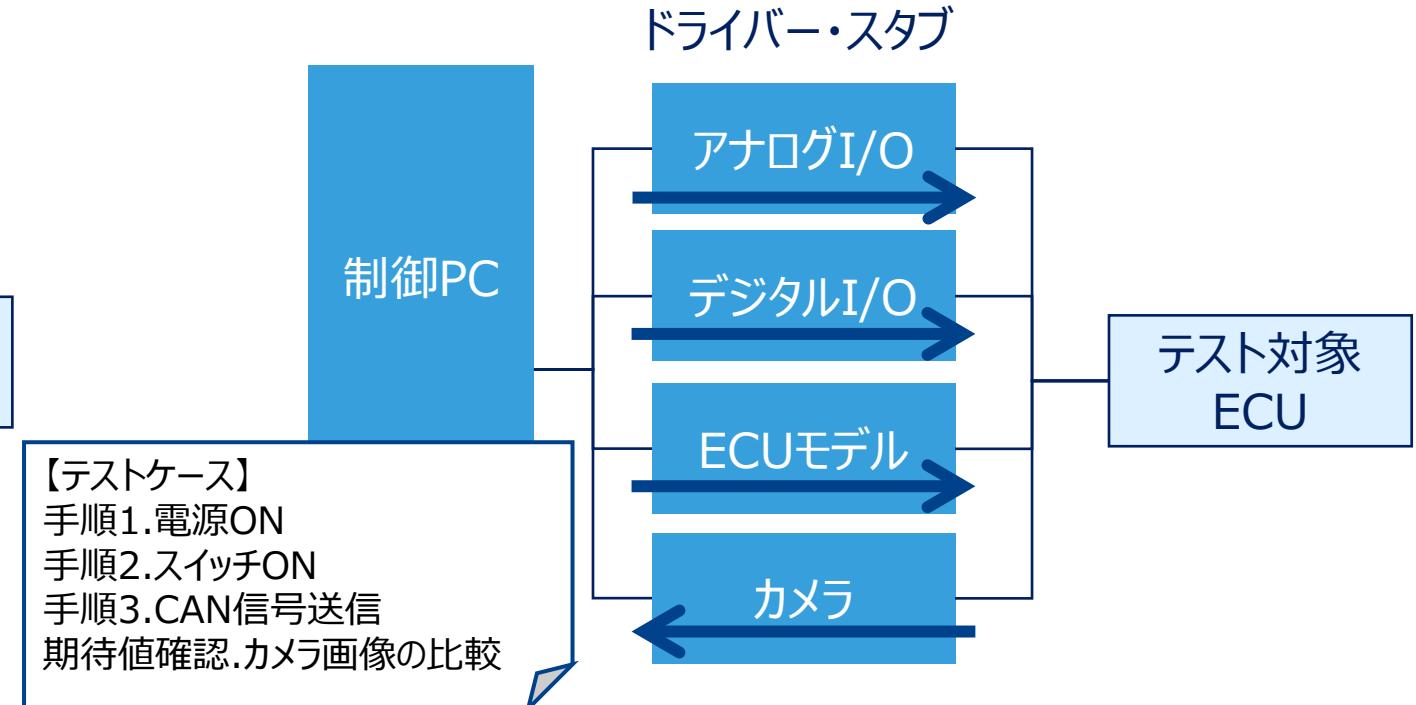
## 3-1. テスト実行の自動化

## インターフェイスに応じた環境を用意する必要がある

### ■ 実機環境



### ■ HILS環境



ポイント：費用対効果等を考慮し、どのインターフェイスのテストまで自動化するかは検討する必要あり。  
→ テスト自動化の8原則①手動テストはなくならない

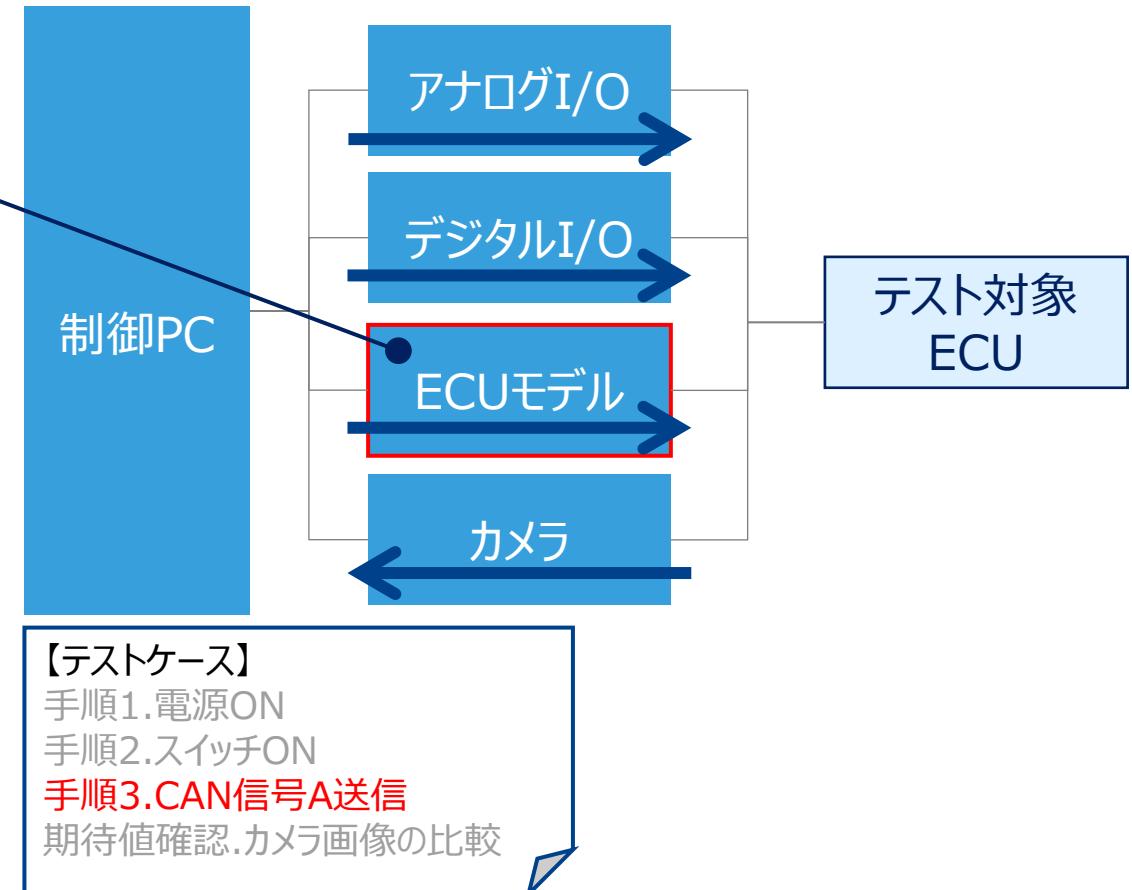
- CAN(-FD)
- LIN
- FlexRay
- 車載Ethernet(100BASE-T1、1000BASE-T1)

OSI参照モデル		車載器	ITシステム
7	アプリケーション層	SOME/IP	
6	プレゼンテーション層		HTTP/MQTT
5	セッション層		
4	トранスポート層	TCP/UDP	TCP/UDP
3	ネットワーク層	IPv4/IPv6	IPv4/IPv6
2	データリンク層	Ethernet MAC+ VLAN	Ethernet
1	物理層	100BASE-T1/ 1000BASE-T1	100BASE-T/ 1000BASE-T

## C言語技術をECUモデル開発に活用

### ■ ECUモデルの例

Vector社CANoeの  
CAPL (Communication Access  
Programming Language)を使用。  
C言語技術を活用できる。



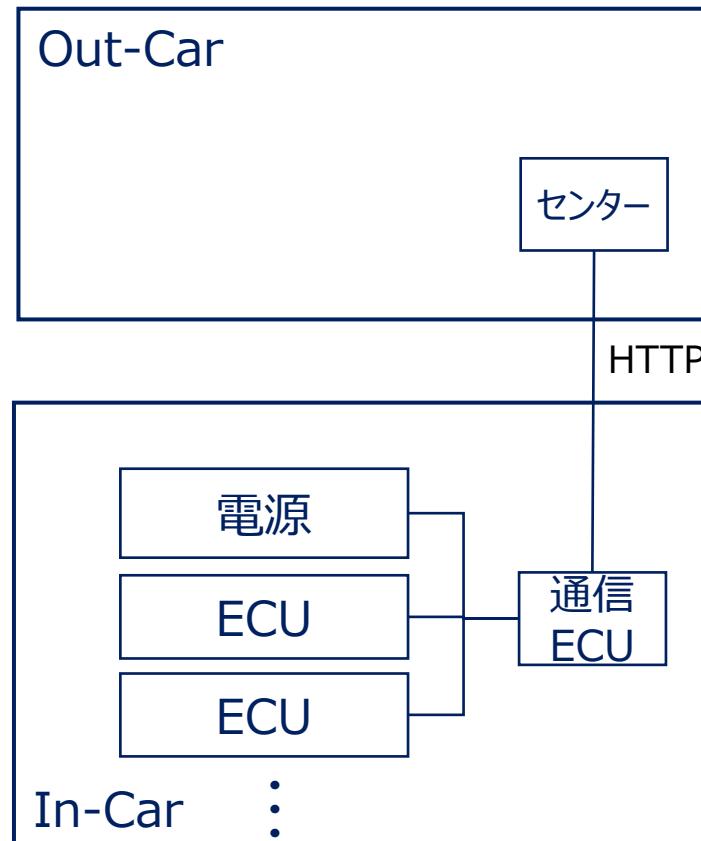
車載器開発の変化：コネクティッドサービスの普及



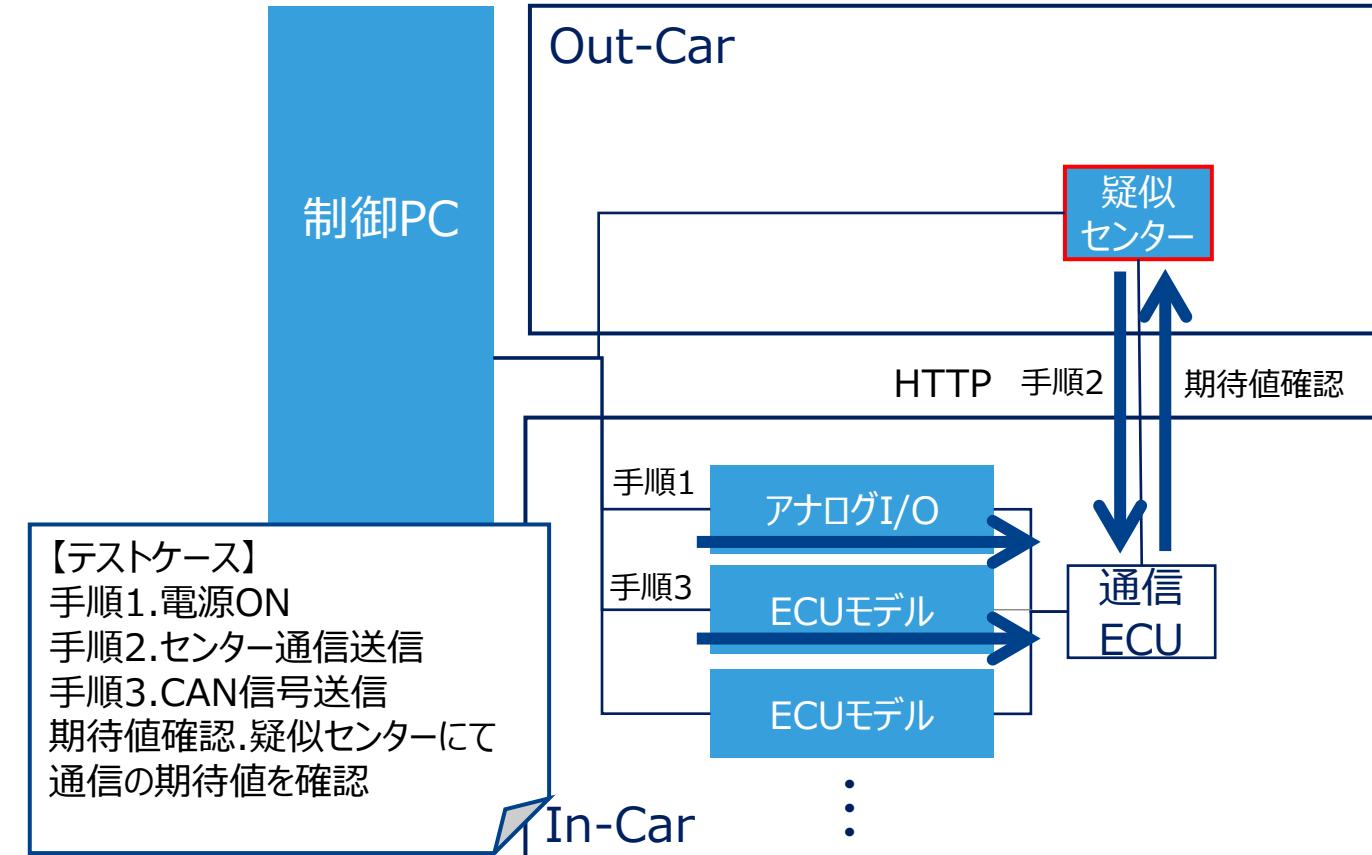
車載器がインターネットに繋がり、一般的なネットワークインターフェイスを使用したテスト技術の活用の場の広がり

## ドライバー・スタブとして疑似センターを用意

### ■ 実機環境



### ■ HILS環境



ITシステムでも使われているインターフェイスの場合、同じ技術が使用できる

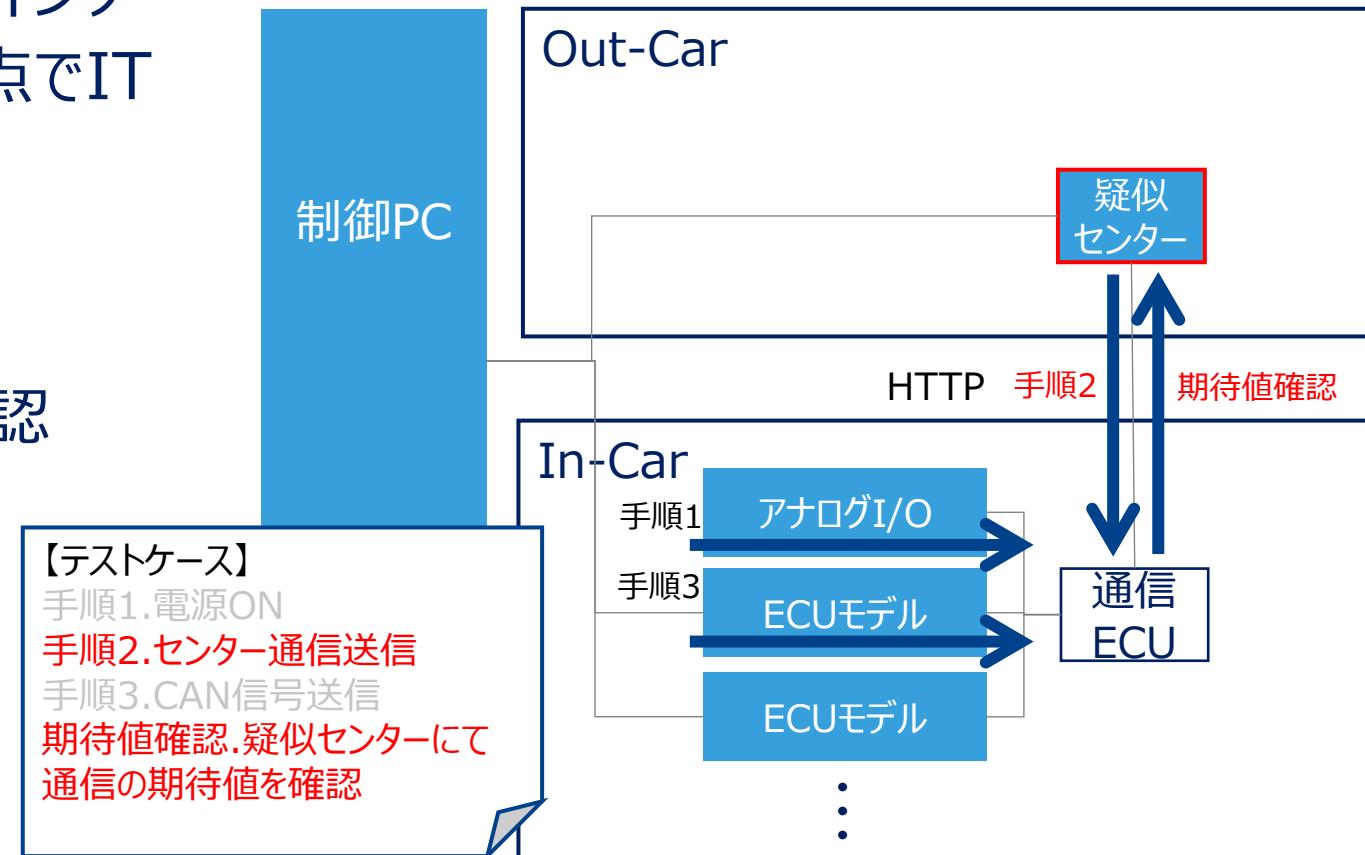
HTTPなどITシステムでも使用するインターフェイスを使用する場合、以下の2点でIT技術が活用できる。

- ・疑似センターの開発

テストドライバーでの信号送信

車載器からの通信の期待値確認

## ■ HILS環境



単体テストはITシステム全般で実施されている。



コードカバレッジをプロジェクトのゲートにしていることが多い。



コードカバレッジをゲートにしている場合は、単体テストツールを使うことが一般的。



車載器のソースコードカバレッジは、MC/DC100%が求められることが多い。

## テスト分析における 機能安全規格の考慮

### 機能安全規格 ISO26262

- ・ ハザード分析とリスクアセスメントによりASIL (Automotive Safety Integrity Level : 自動車安全水準) を導く。
- ・ ASILに応じたテストを行う。

例 : ASIL D(ハザードの程度が最高)の場合

- ・ 強く推奨するテスト手法
- ・ **MC/DC 100%**

## 単体テストツールにより複数のプロセスを自動化できる



- テストケースの自動生成
- ドライバー・スタブの自動生成
- 自動実行
- コードカバレッジの自動計測



コードカバレッジが不足する場合は  
手動でテストケースを追加

ポイント：自動を手動で補完することでコードカバレッジ100%を実現する。  
→ テスト自動化の8原則①手動テストはなくならない

## 3-2. テスト設計の自動化

車載器開発の変化：シムテムズエンジニアリングの重要性の高まり

システムズモデリング言語SysMLによる仕様書の普及

SysMLの活用場面

- モデルでの検証
- モデルからソースコードの自動生成
- モデルからテストケースを自動生成

SysMLモデルからテストケースを自動生成する取り組み  
省略してMBT(Model Based Testing)と呼ぶ。



## MTBの意義

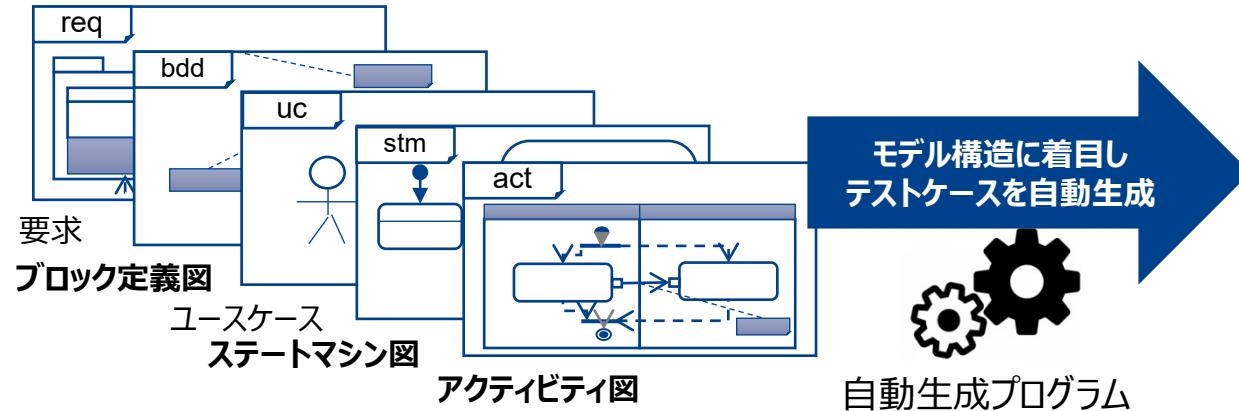
- ・ ステークホルダーがテスト設計仕様を理解しやすくなる。
- ・ テスト自動化がしやすくなる。

## 自動生成の意義

- ・ ツールの誤りがなければ、テスト設計品質の向上が期待できる。
- ・ テスト設計を省力化でき、多数のテストをローコストで作成できる。

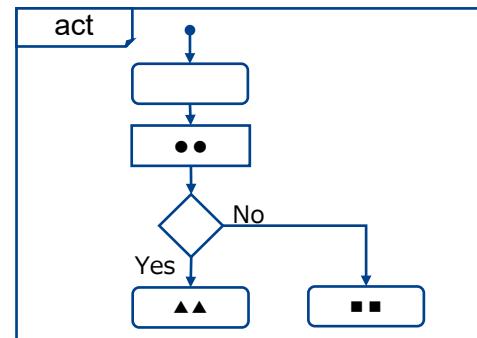
## ➤ モデルからテストケースを自動生成

### 仕様モデル

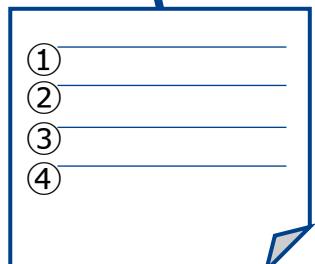


### ■ アクティビティ図の例

- ◆ 分岐条件のオブジェクトがある場合  
にどちらの分岐を通るか

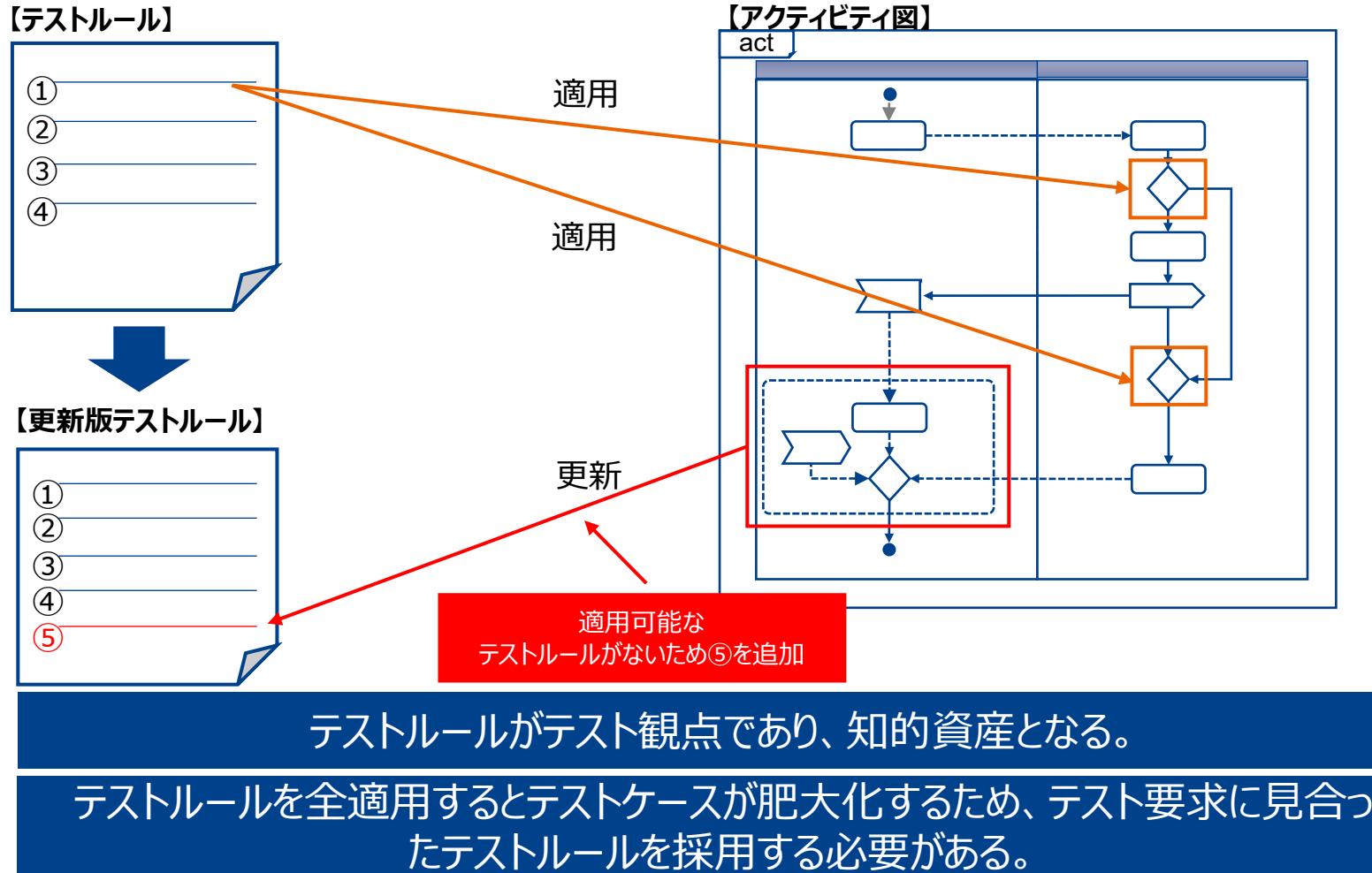


【テストルール】



## ➤ テストルール更新のプロセス

- テストルール…分岐、Loop、停止
- テストルール…イベント（時間、割り込み）



## 使用技術や特徴：

特定のツール（Rhapsody、Enterprise Architect）を常用。

ツール固有のフォーマットからSysMLを拡張した独自フォーマットに変換する必要あり。

## 車載器以外への活用：

SysMLやUMLで仕様書を作成している場合は、車載器以外でも同じ技法が使える。

## 3-3. テスト管理の自動化

開発のスピードアップ



テストからのフィードバックのスピードアップ



テスト状況をリアルタイムで把握する必要性



テスト自動化の拡大

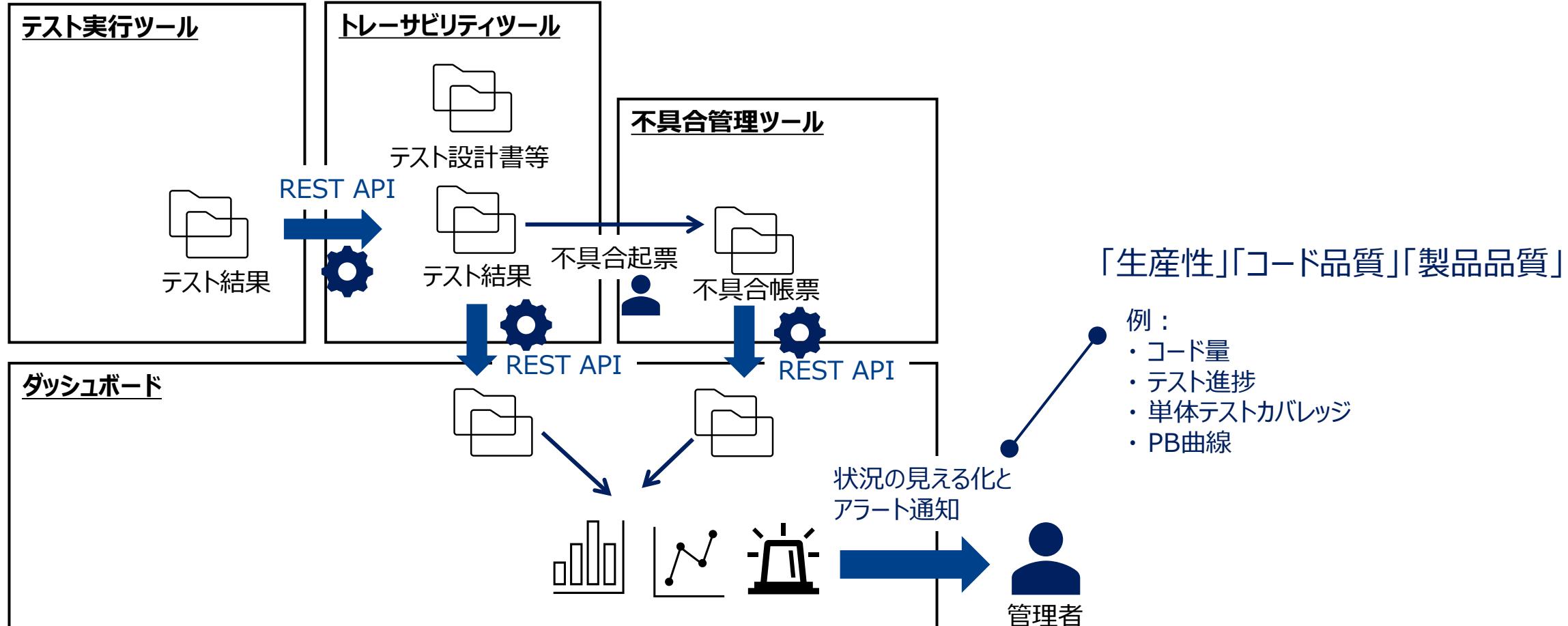


テスト結果など管理対象の増大



管理も自動化しなければ欲しい情報がリアルタイムで取得できない

## REST APIによる情報収集とダッシュボード化



## 活用できるIT技術：

テスト対象が車載器かWebアプリかに関わらず、REST APIによるシステム連携技術は活用できる。

## 使用技術や特徴：

トレーサビリティツール等の、車載器では常用するツールのREST APIに関する実践的な事例情報が少ない。

## 3-4. ツールチェーンの自動化

自動化適用プロセスの拡大  
リグレッションテストの高頻度化



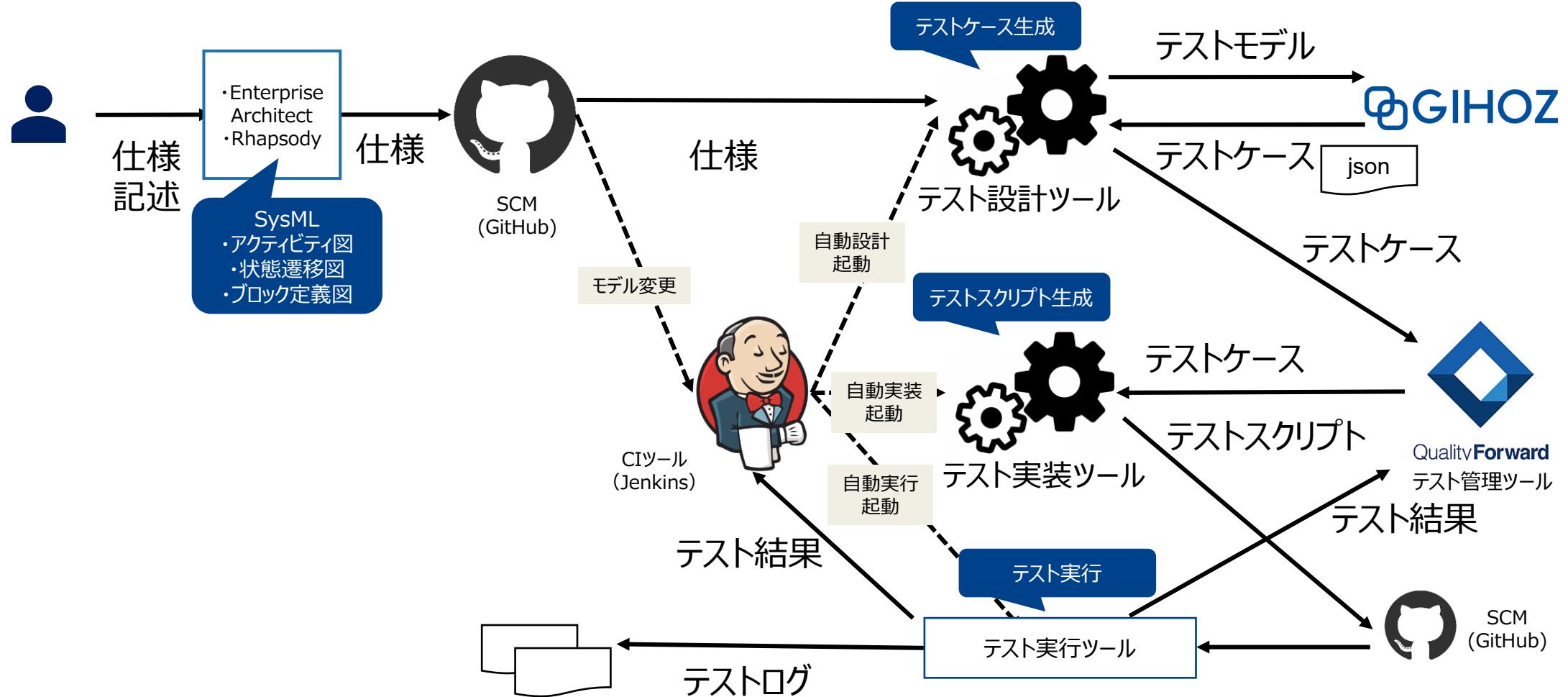
各プロセスを連続して（チェーン）起動することにより  
プロセス単体の自動化より更なる効率化を目指せる。

## ECUを対象としたテスト実行の自動化



ハードボタン押下で書き込みモードに入るシステムだと、書き込み自動化のためにシステムが必要となってしまう。ソフトのみで書き込みモードに入れる仕組みを、プロジェクト開始時に要件化しておくのが良い。  
→ テスト自動化の8原則⑥「自動化検討はプロジェクト初期から」

## MBTのテスト実装・テスト実行への拡大



## 4. まとめ

---

自動化フェーズ	車載器の事例	車載器開発の変化	活用できるIT技術	車載器の特徴
3-2. テスト設計	SysMLからの テストケース自動生成	仕様のモデル化	全般	特定のツールを使用し フォーマット変換が必要
3-1. テスト実行	非通信ECUのHILSテスト	—	C言語技術	多彩なインターフェイス 車載ネットワーク規格
	通信ECUのHILSテスト	コネクティッドサービスの 普及	HTTPなどの 通信規格の使用	同上
	単体テストにおける カバレッジ確保	—	全般	ISO26262に基づく MC/DC100%
3-3. テスト管理	情報収集と ダッシュボード化	テスト状況をリアルタイムで 把握する必要性	REST APIでの システム連携	(特有のシステムとの連携)
3-4. ツールチェーン	CIによる ツールチェーン自動化	スピードアップの必要性	全般	ソフト書き込み

1. 手動テストはなくならない
2. 手動でおこなって効果のないテストを自動化しても無駄である
3. 自動テストは書いたことしかテストしない
4. テスト自動化の効用はコスト削減だけではない
5. 自動テストシステムの開発は継続的におこなうものである

6. 自動化検討はプロジェクト初期から
7. 自動テストで新種のバグが見つかることは稀である
8. テスト結果分析という新たなタスクが生まれる

3-1(1). テスト実行自動化：ECUのHILSテスト  
費用対効果等を考慮し、どのインターフェイスのテストまで自動化するか検討

3-1(2). テスト実行自動化：単体テスト  
自動を手動で補完することでカバレッジ100%を実現する。

3-4.ツールチェーンの自動化：CI  
ソフトのみで書き込みモードに入れる仕組みをプロジェクト開始時に要件化しておくのが良い。

3-3.テスト管理の自動化：ダッシュボード化  
分析の作業にも自動化を導入できる。

引用：テスト自動化研究会「テスト自動化の8原則」[https://sites.google.com/site/testautomationresearch/test\\_automation\\_principle](https://sites.google.com/site/testautomationresearch/test_automation_principle)

事例を用いて、車載器で常用する技術・ポイントに注意しつつ、IT技術を活用できれば車載器のテスト自動化の幅が広がるというお話をさせていただきました。

そのヒントを1つでも本日持ち帰っていただくことができれば幸いです。

品質を創造する

VERISERVE

ご清聴ありがとうございました。