



# 自動運転開発向け連成シミュレーション環境の 合流判断技術への適用

MATLAB EXPO 2023

マツダ株式会社 統合制御システム開発本部  
石崎 大智 山内 優

# 目次

- 会社概要と直近リリース紹介
- AD/ADAS開発における問題、課題、アプローチ
- 構想設計MILS 取り組み紹介
  - ✓道路モデル自動生成
  - ✓技術アイデアの組み込みシミュレーション
- MILS運用の結果及び効果
- まとめ、及びMathWorks様への要望

# はじめに

## • 会社概要

- 社名 マツダ株式会社
- 設立 1920年（大正9年）
- 本社 広島県安芸郡府中町新地3-1
- 代表者 丸本 明 ⇒ 毛籠 勝弘(2023.6～)
- 事業内容 乗用車・トラックの製造、販売など
- 従業員数 単体 23,266名 連結 48,750名
- 販売台数 約111万台（2023年3月期）

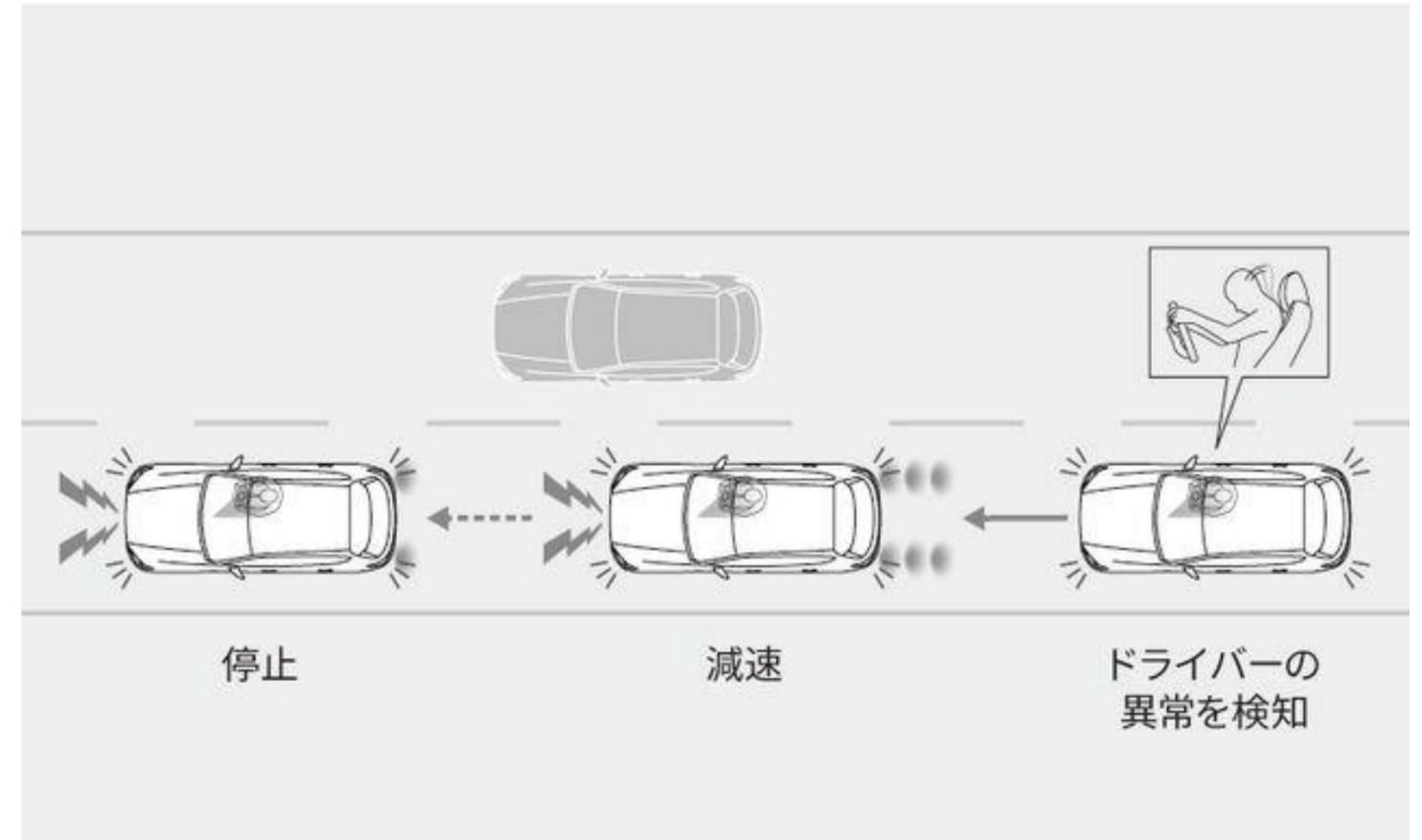
## • 直近の国内リリース

- MAZDA CX-60
  - ✓市村産業賞 功績賞 受賞
  - ✓日本自動車殿堂 カーテクノロジーオブザイヤー 受賞



# 最新ADAS機能紹介：ドライバー異常時対応システム（DEA）

- DEA：Driver Emergency Assist
- 国連協定規則に国内で初めて対応したドライバー異常時対応システム
- 搭載車種：MAZDA CX-60国内向けモデル

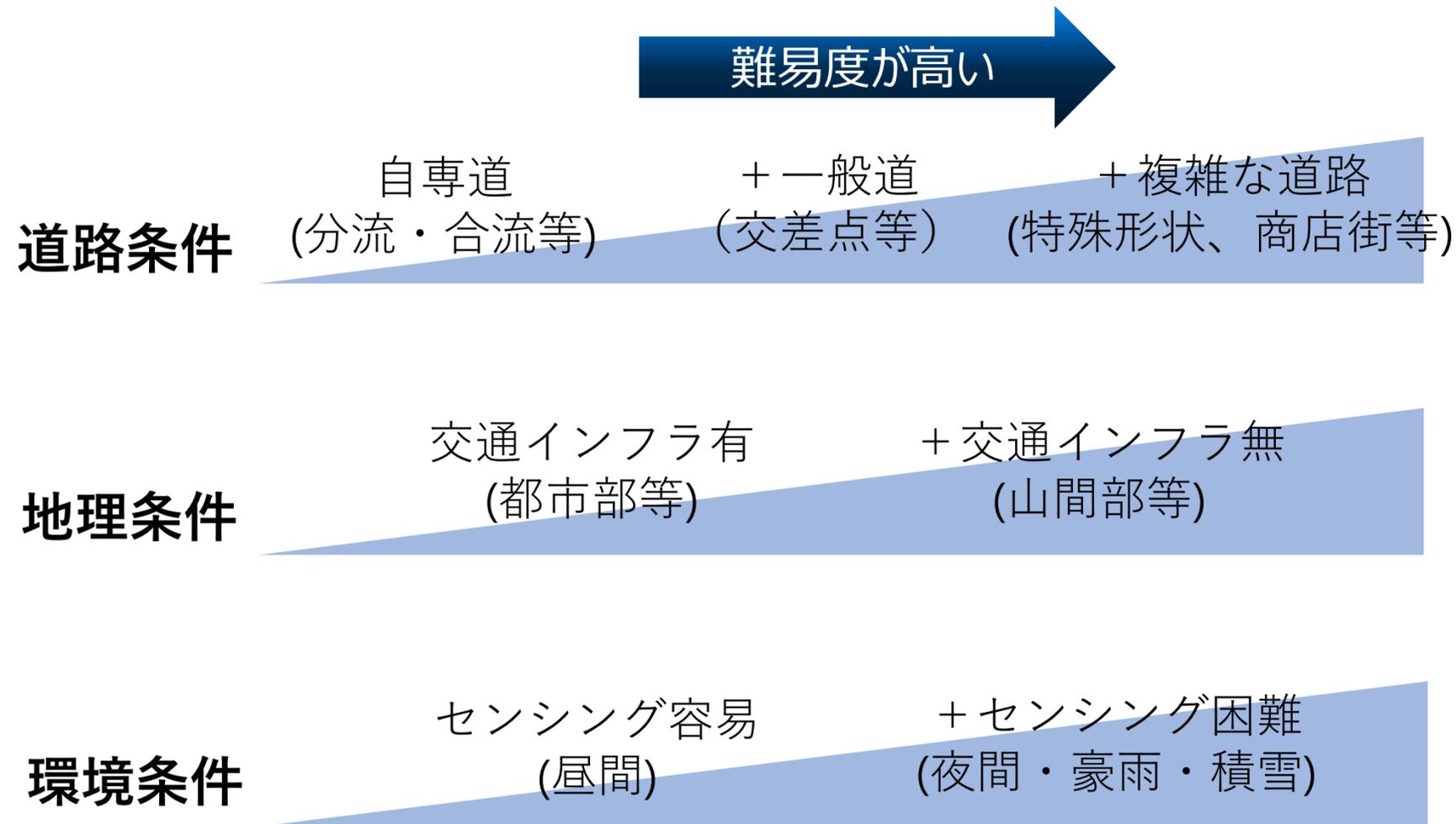


非常点滅表示灯に加え、ブレーキランプの点滅とホーンの吹鳴を開始し、車両を減速・停止

AD/ADAS機能で安心安全を提供⇒カーライフを通じて人生の輝きを人々に提供

# AD/ADAS機能のODDロードマップ

- 安心安全なAD/ADAS機能のあるべき姿
    - 「いつでも」、「どこでも」、「どんなときでも」正しく働く
- ⇒ 広いODD(Operational Design Domain) = 運転設計領域



## ODD拡大が遅れる要因

アルゴリズム複雑化、ソフトウェア規模増大



マツダ車のソフトウェア規模推移

コード行数によるソフトウェア規模比較  
<https://informationisbeautiful.net/visualizations/million-lines-of-code/>

問題：ソフトウェア規模増大によりODD拡大が遅れる

# 問題解決のキーとなる新技術

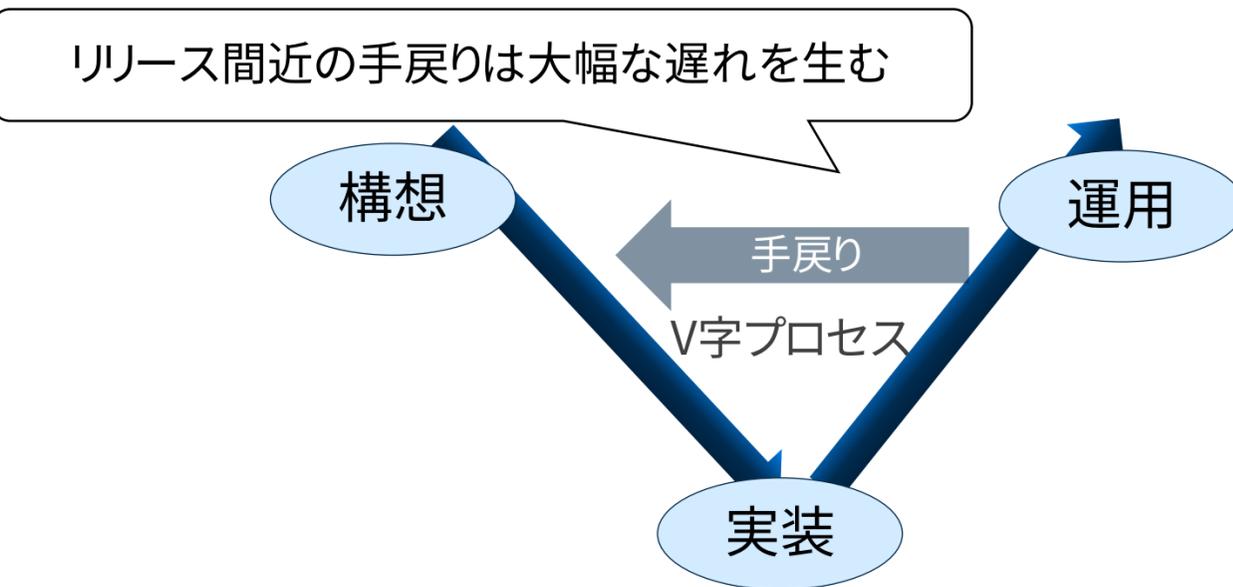
## • 打開策となる新技術

### ➤ AI（人工知能）、IoT、ビッグデータ

✓リスク：挑戦による**多くの手戻り、事故等**

#### ■ 国内での事例

- 2019.8 : 自動運転車両が接触事故
- 2020.3 : 自動運転バスが都内で物損事故
- 2020.8 : 実証実験で接触事案
- 2020.12 : 自動運転バスがガードレールに接触
- 2021.8 : 東京五輪の選手村で自動運転車が接触事故
- 2023.1 : 大津市内で乗客が座席から滑り落ちる事故



#### ■ 海外での事例

- 2016.2 : 自動運転車が路線バスと衝突
- 2016.5 : 部分自動運転モード中に死亡事故
- 2018.3 : 自動運転中に歩行者と死亡事故
- 2018.3 : 自動運転モード中の死亡事故

参考: 自動運転車の事故(2023年最新版) | 自動運転ラボ (jidounten-lab.com)

課題:「新技術」を構想段階で見極め、ODD拡大を加速させる環境構築

# アプローチ：構想設計MILS※

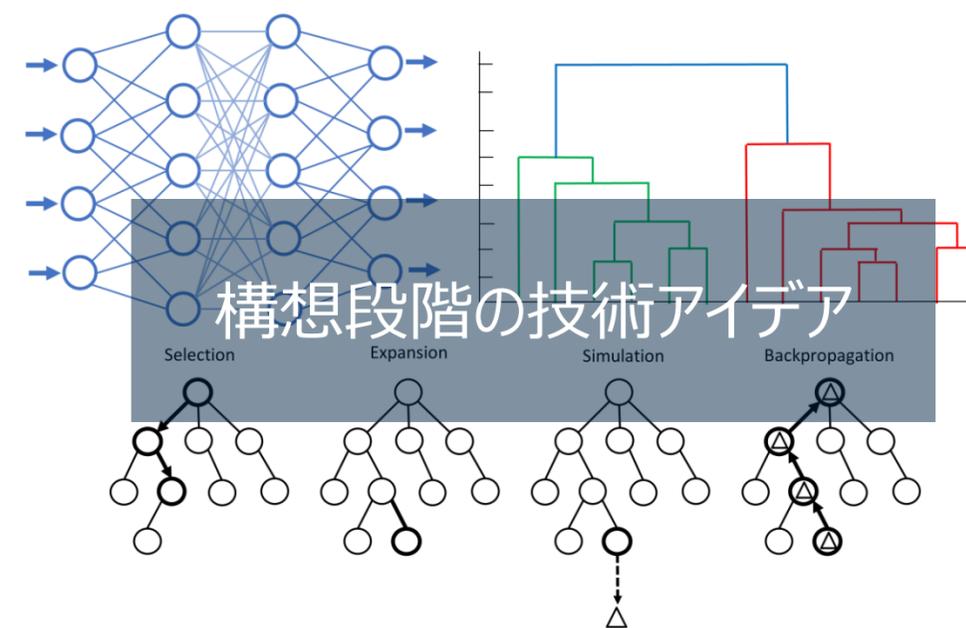
- 要求：実運用シーンで技術アイデアを机上検証できる

※MILS:Model In the Loop Simulationの略で制御器と制御対象をすべて数値モデルとして開発し、結合するシミュレーション手法

## 構想設計MILS



運用シーンに合わせて、車両、他者エージェント、道路、インフラ、人間(ユーザ)等を組み替えることができる

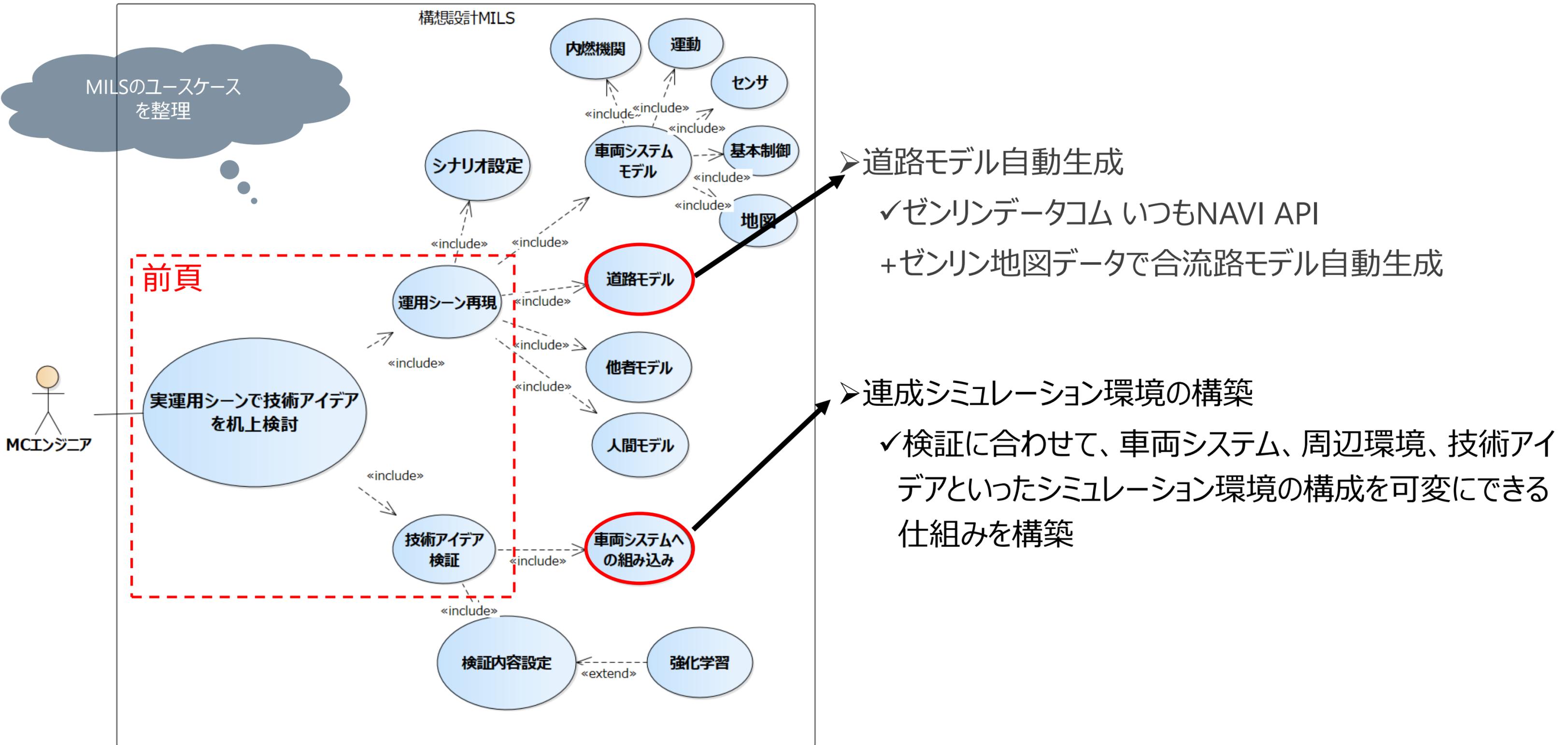


AD/ADAS機能を実現する学習アルゴリズム等設計技術を組み換えながら検証できる

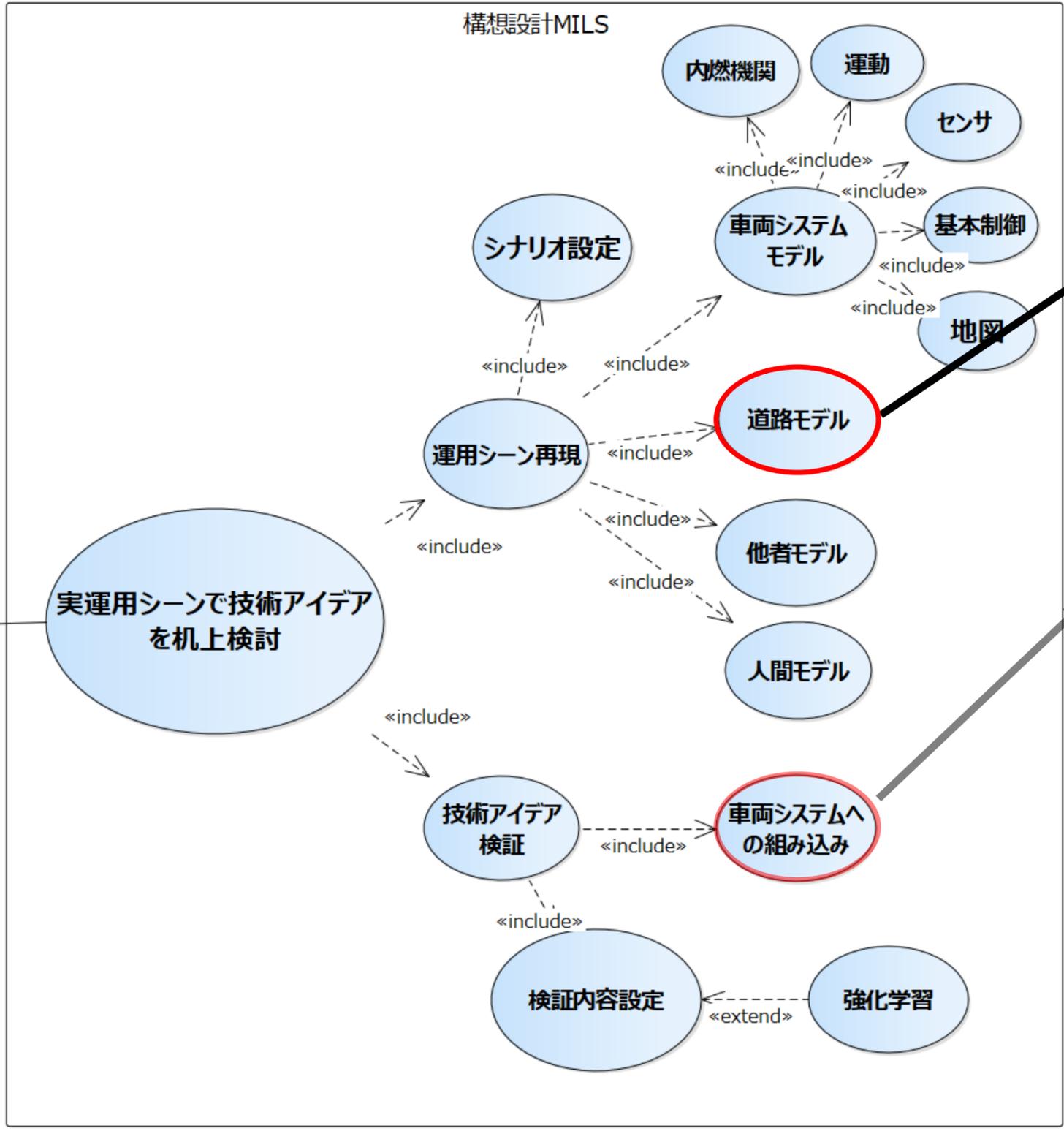
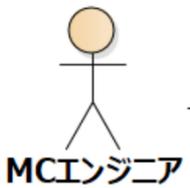
運用シーンの再現

技術アイデアの評価検証

# 構想設計MILS開発 取り組み紹介



# 構想設計MILS開発 取り組み紹介



道路モデル自動生成  
 ✓ゼンリンデータコム いつもNAVI API  
 +ゼンリン地図データで合流路モデル自動生成

連成シミュレーション環境の構築  
 ✓検証に合わせて、車両システム、周辺環境、技術アイデアといったシミュレーション環境の構成を可変にできる仕組みを構築

# 道路モデル取り組み紹介

## • 目標

- 実世界の様々な道路を机上に再現する
  - ✓ 直近要求：日本全国の合流路の再現

## • 問題点

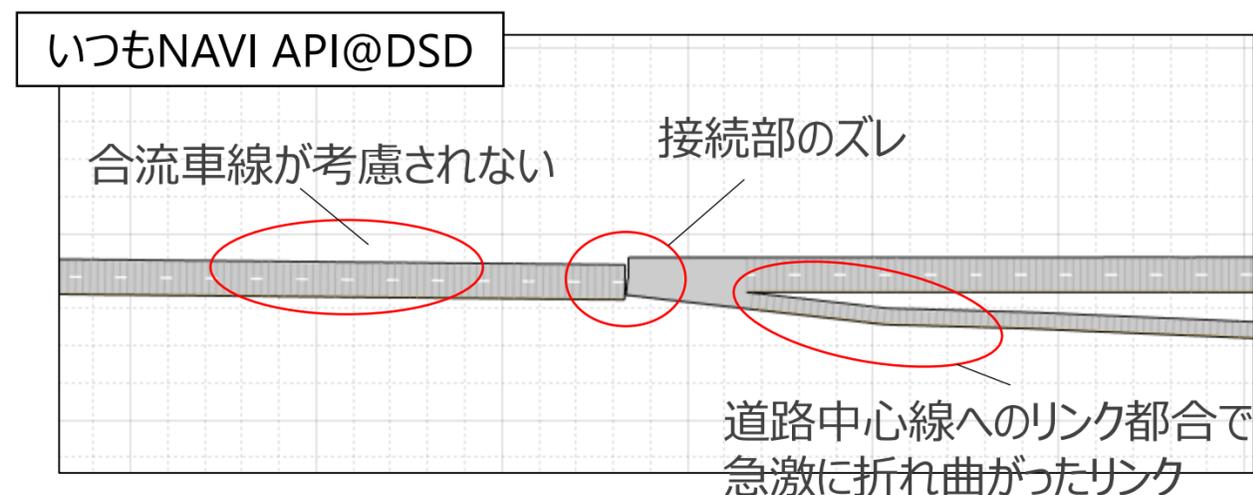
- Automated Driving Toolbox™(ADT)の Driving Scenario Designer(DSD)でいつもNAVI APIからインポートし生成した道路は粒度・精度不十分
- HDマップは高コストかつ対応地域が一般道含めた全国を網羅していない

## • 課題

- 広範囲低コストなSDマップを組み合わせ要求を満たす道路モデルを生成

## • アプローチ

- いつもNAVI API + ゼンリンの地図データ(shape形式)を組み合わせた道路モデル生成ツール開発

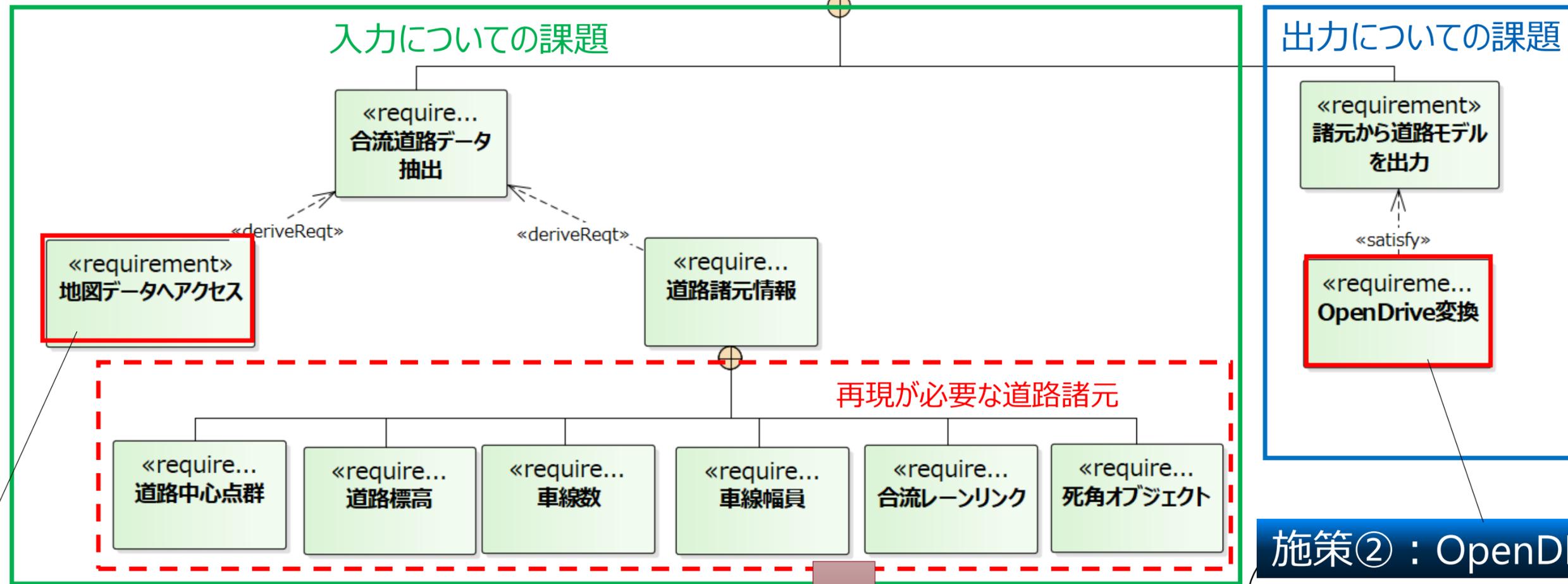


引用元: Google社「Google マップ」

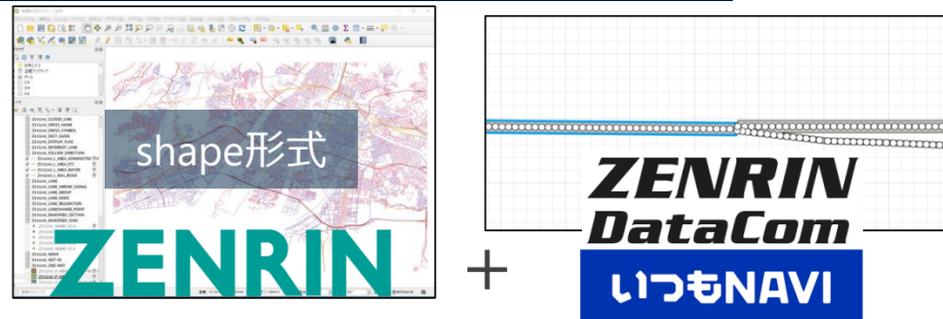
**ZENRIN**

**ZENRIN  
DataCom**

# 道路モデル自動生成ツールの要件



## 施策①：地図データへアクセス

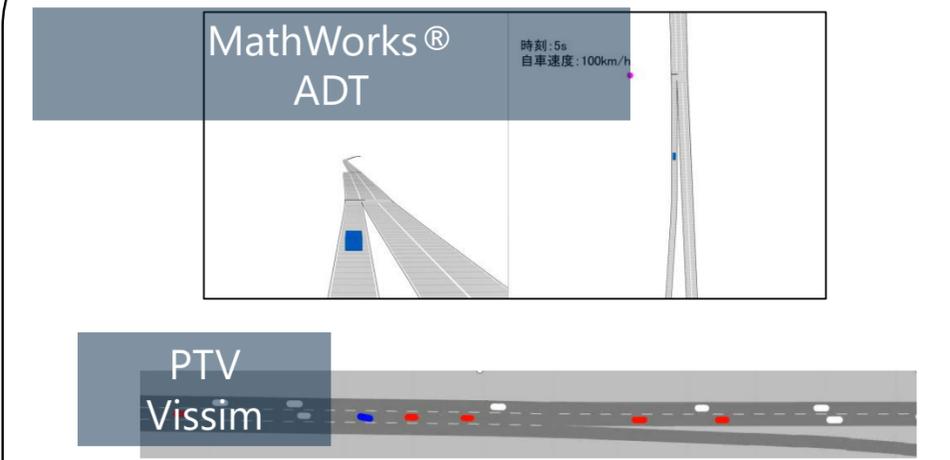


必要粒度の情報を持つナビ地図データか日本全国の道路モデルパラメータを抽出

	ゼンリンデータコム いつもNAVI API	ゼンリン 地図データ
道路中心点群	○	○
道路標高	○	×
車線数	△	○
車線幅員	△	○
合流レーンリンク	×	○
死角オブジェクト	×	×

※死角オブジェクトは別途対応を検討中

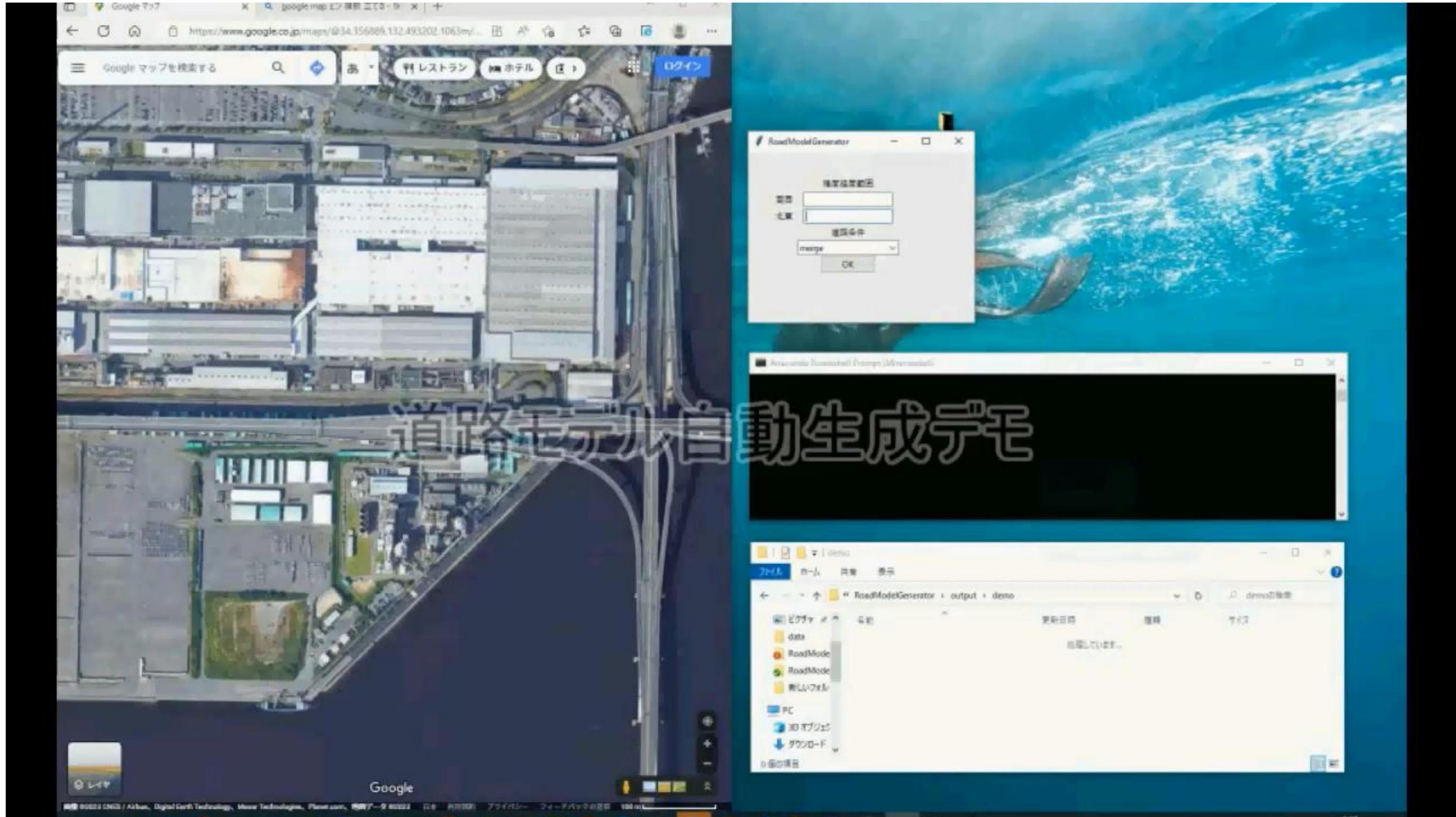
## 施策②：OpenDRIVE変換



学習・検証シミュレーション環境にインポート可能なOpenDrive形式に変換

# 合流路モデル自動生成デモ

引用元: Google社「Google マップ」

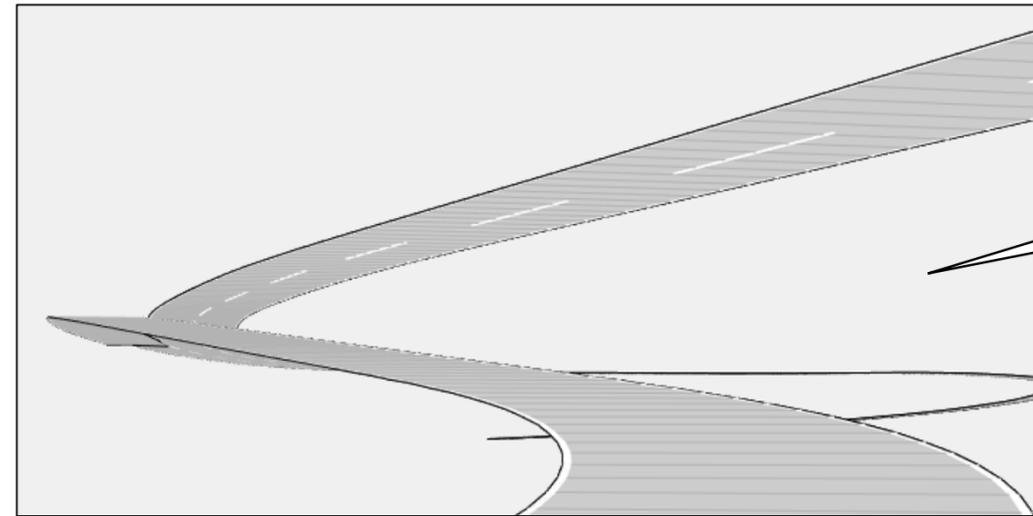


# 評価：道路モデルの再現度

- 出力されたOpenDRIVEファイルを実道路(Googleマップ)と比較評価



引用元: Google社「Google ストリートビュー」

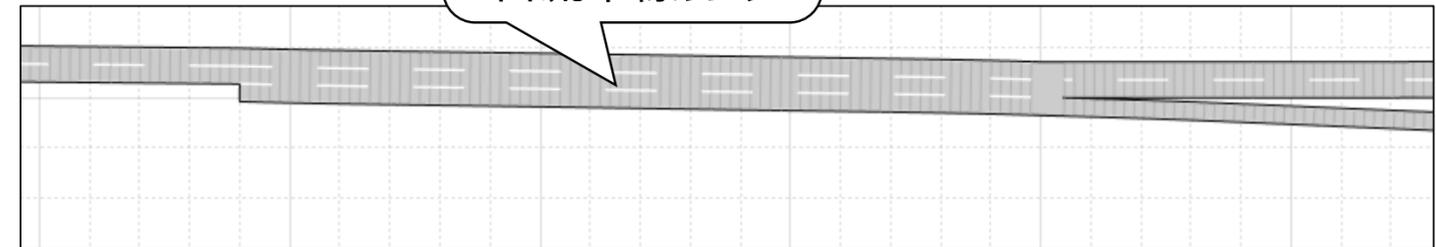


・道路の曲率(道路中心点群)  
・標高

	ゼンリンデータコム いつもNAVI API	ゼンリン 地図データ
道路中心点群	○	○
道路標高	○	×
車線数	△	○
車線幅員	△	○
合流レーンリンク	×	○
死角オブジェクト	×	×



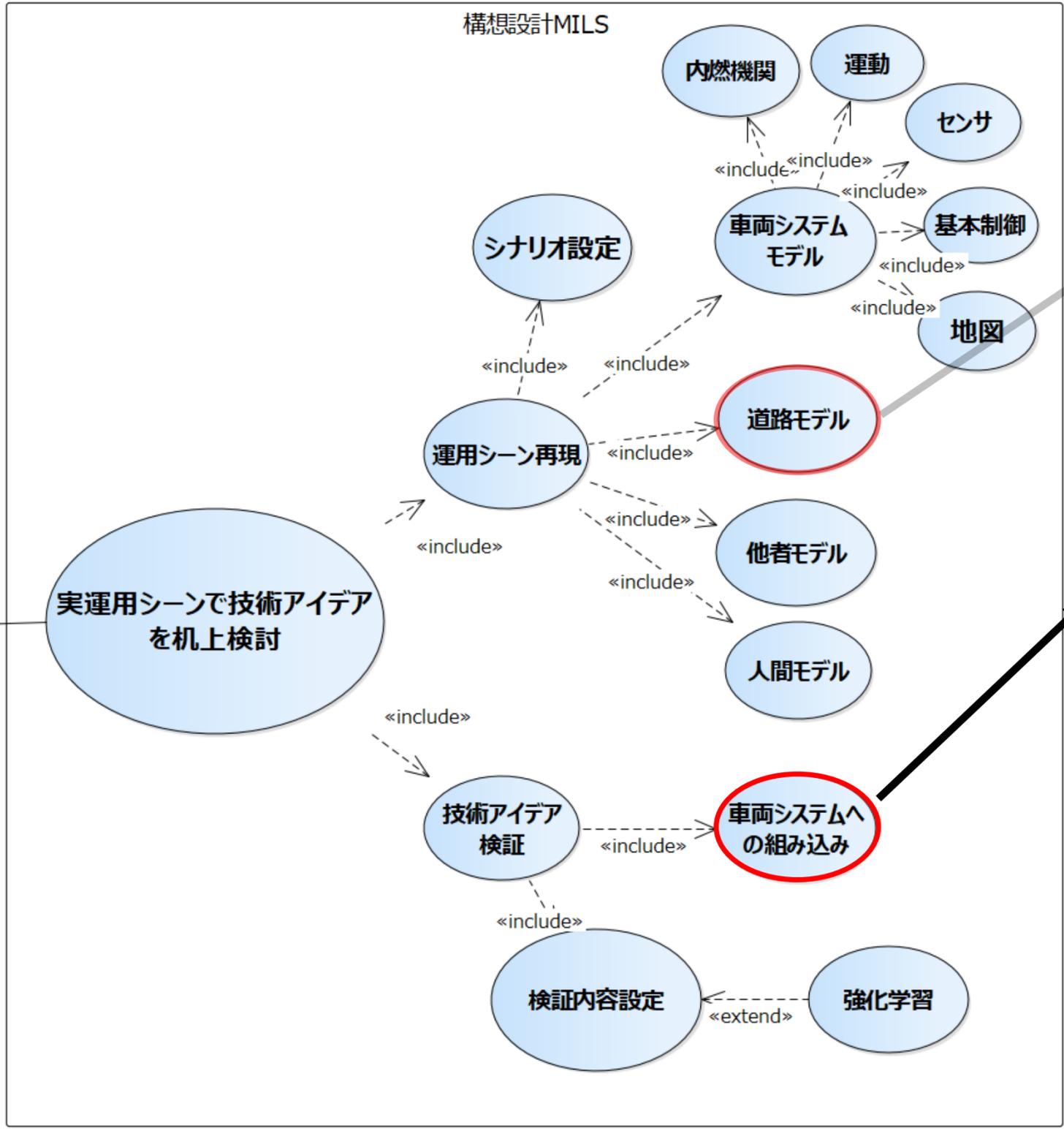
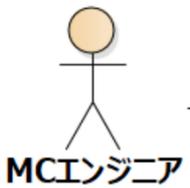
引用元: Google社「Google マップ」



・車線数  
・幅員  
・合流車線リンク

達成：第一段階として、日本全国の合流路について再現が可能となった

# 構想設計MILS開発 取り組み紹介



道路モデル自動生成  
 ✓ゼンリンデータコム いつもNAVI API  
 +ゼンリン地図データで合流路モデル自動生成

連成シミュレーション環境の構築  
 ✓検証に合わせて、車両システム、周辺環境、技術アイデアといったシミュレーション環境の構成を可変にできる仕組みを構築

# 連成シミュレーション環境の構築

## • 目標

- ▶ 実環境を再現したバーチャル環境で技術アイデア(AI等)を検証

## • 問題点

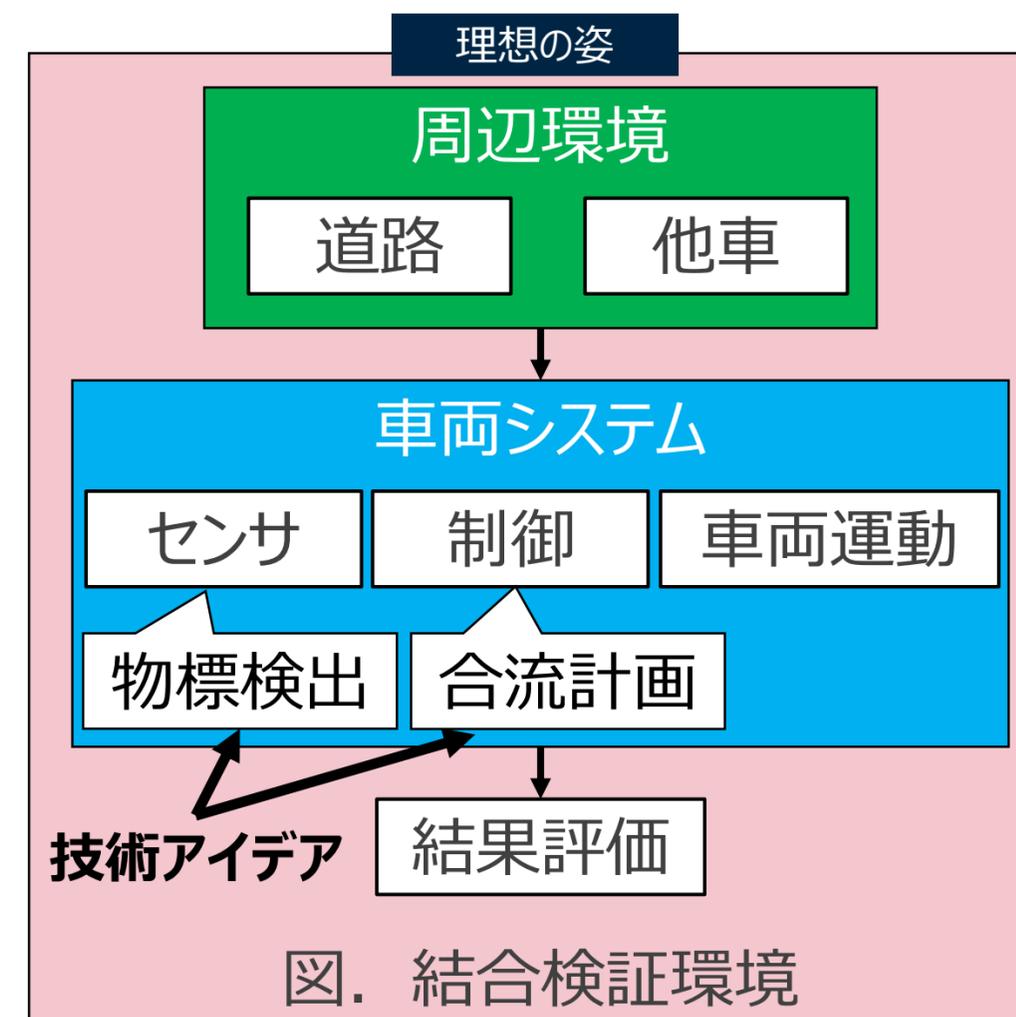
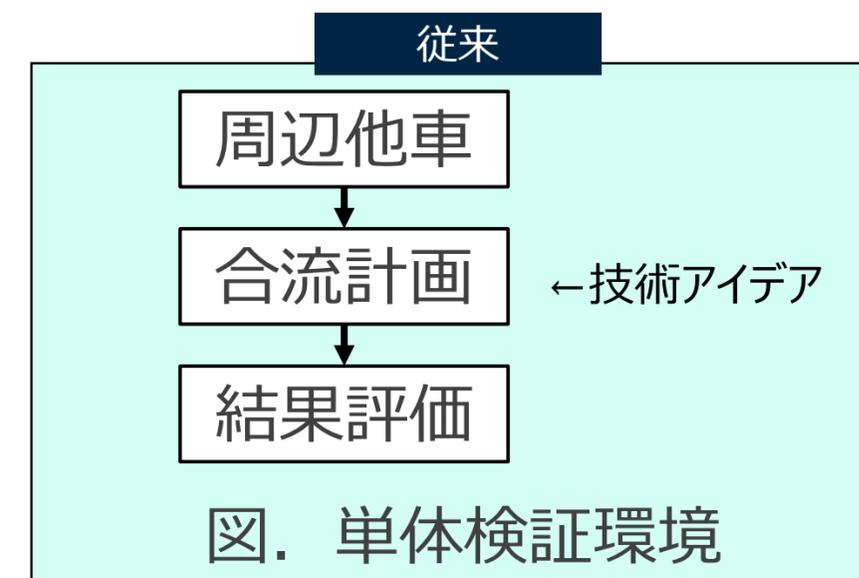
- ▶ 開発対象の技術ごとに独立した単体検証環境を用意
  - ✓ 重複する環境要素があり非効率的
- ▶ 関心領域外であるモデル（例．道路モデル）の再現性不足により、想定できていなかった不具合が実運用シーンで発生

## • 課題

- ▶ 周辺環境モデルと車両1台分のシステムを含む結合検証環境
  - ✓ 技術アイデアのみ実装して検証可能⇒環境構築に割く工数を最小化
  - ✓ 複数の技術アイデアを結合して検証可能⇒構想段階から技術課題の早期発見が可能

## • アプローチ

- ▶ 「MATLAB®/Simulink® + Automated Driving Toolbox™ (ADT)」の活用
- ▶ ドメイン特化（道路、他車など）ツールとの連成環境構築



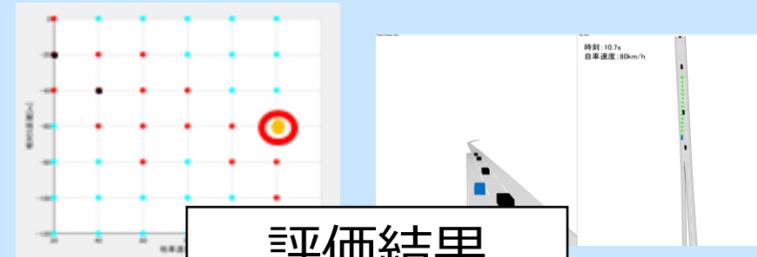
# MATLAB/Simulink+ADTを活用したシミュレーション環境の全体構成

## ユーザインタフェース

① 検証内容を設定できる  
⇒ MATLAB + ADT



検証内容設定



評価結果

② 周辺環境を組み替える  
⇒ ADTの連成用  
ブロックセット等

ゼンリン地図データ

道路モデル

MC内製  
(@DSD)

or

PTV Vissim

他者行動モデル

周辺環境モデル

③ 車両システムを組み替える  
⇒ Simulink+ADT

センサ・制御  
運動モデル

④ 技術アイデアの組み込み  
⇒ Deep Learning Toolbox™

行動計画モデル

技術アイデア

車両システムモデル

# MATLAB/Simulink+ADTを活用したシミュレーション環境の全体構成

## ユーザインタフェース

①検証内容を設定できる  
⇒MATLAB + ADT

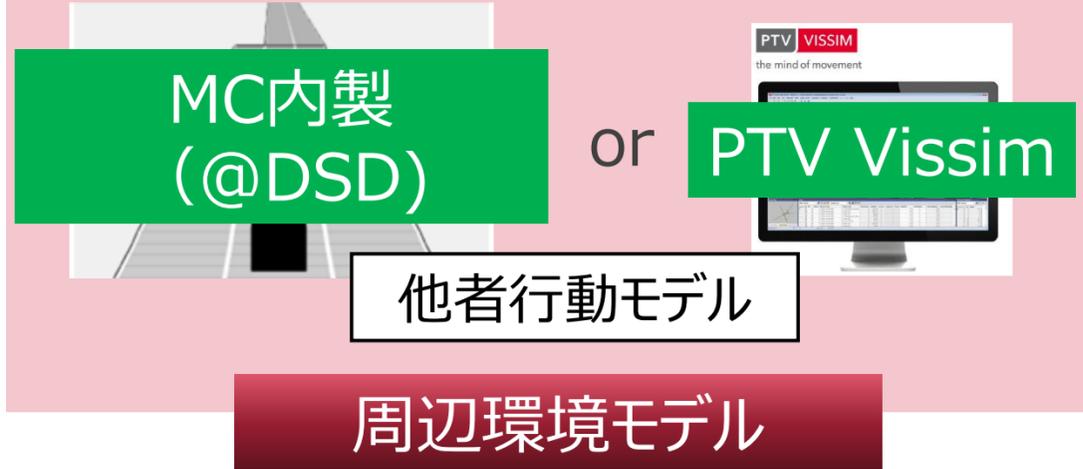


検証内容設定



評価結果

②周辺環境を組み替える  
⇒ADTの連成用  
ブロックセット等



③車両システムを組み替える  
⇒Simulink+ADT



④技術アイデアの組み込み  
⇒Deep Learning Toolbox™

# ① 検証内容を設定できるMATLAB + ADT

- ユーザが扱いやすいように各種パラメータは「App Designer」を活用してGUIで設定

```

12 % YAW_CONTROL_SWITCH = 1; %1:理想制御, 2:PID制御, 3:Later
13
14 VEHICLE_DYNAMICS_SWITCH = 2; %車両運動モデルの切り替え (1:理想
15 SPEED_CONTROL_SWITCH = 3; %1:理想制御, 2:PID制御, 3:Longi
16 YAW_CONTROL_SWITCH = 3; %1:理想制御, 2:PID制御, 3:Lateral
17
18 INI_EGO_VEL = 60/3.6; %自車初期速度
19 TARGET_VELOCITY_MAP = 80/3.6*ones(1,6); %自車の目標速度マップ
20 MAX_ACCEL = 9.8*0.5; %最大加速度[m/s^2] (理想モデルの時に有効)
21 MAX_BRAKE = -9.8*1; %最大減速度[m/s^2] (理想モデルの時に有効)
22 SPEED_CONTROL_KP = 1e+1; %速度制御をPID制御で行う場合の比例ゲ
23 SPEED_CONTROL_KI = 1e-1; %速度制御をPID制御で行う場合の積分ゲ
24 SPEED_CONTROL_KD = 0; %速度制御をPID制御で行う場合の微分ゲイ
25 YAW_CONTROL_KP = 1e+0; %ヨー制御をPID制御で行う場合の比例ゲイ
26 YAW_CONTROL_KI = 0e-3; %ヨー制御をPID制御で行う場合の積分ゲイ
27 YAW_CONTROL_KD = 0; %ヨー制御をPID制御で行う場合の微分ゲイン
28 MAX_YAW_RATE = 1; %1sで変えられる最大ヨー角 (理想車体のみは)
29 REQUEST_ROUTE = [1 2]; %目標経路の道路番号と車線番号
30
  
```



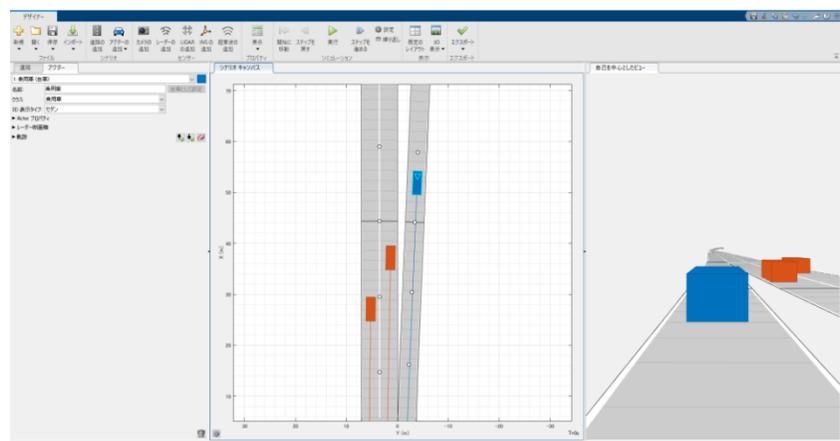
mスクリプトで記述する方法からGUIで設定できる方法へ



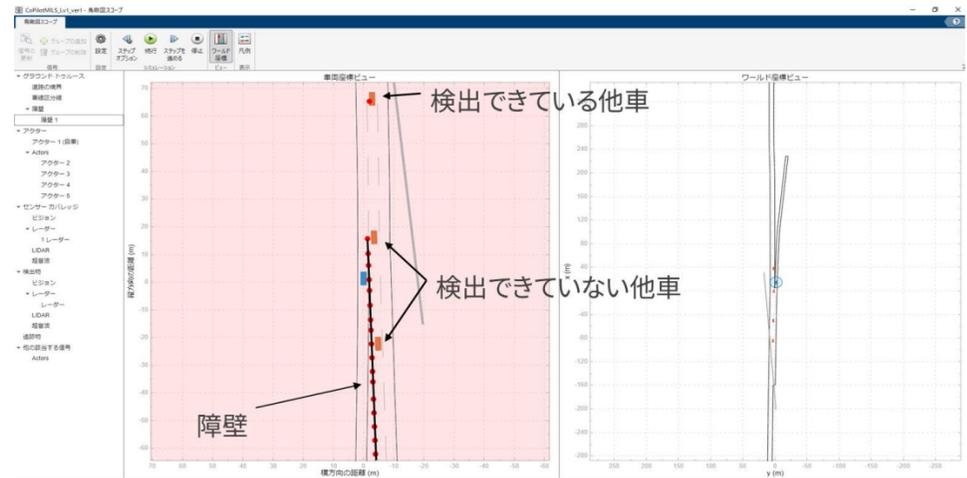
- ADTのDSDで道路や他者を含むシナリオを直感的な操作で作成可能。

➤ 作成したシナリオは専用の変数型で格納でき、シミュレーションの実行中や実行後の可視化関数の引数として利用可能

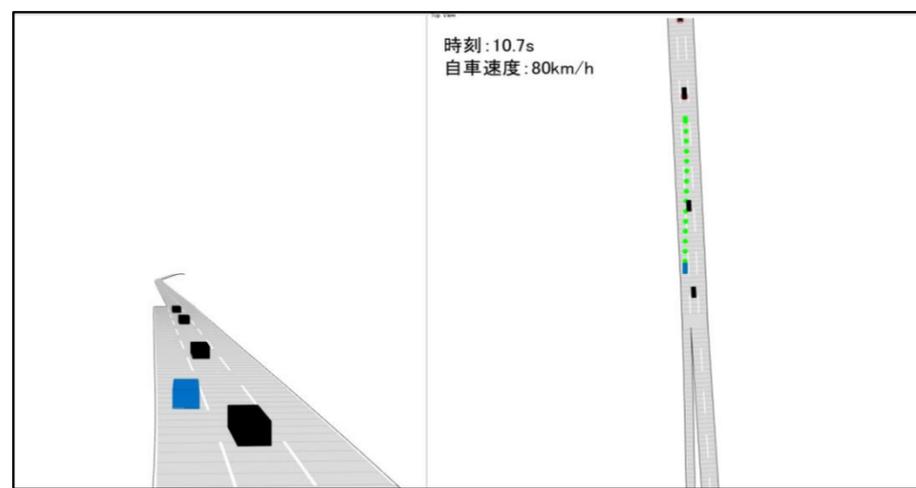
シミュレーション前



シミュレーション中



シミュレーション後



図：ドライビングシナリオデザイナーアプリ

図：シナリオ変数を活用した可視化機能

達成：様々な検証を想定したシナリオ作成含むパラメータ設定のGUI構築

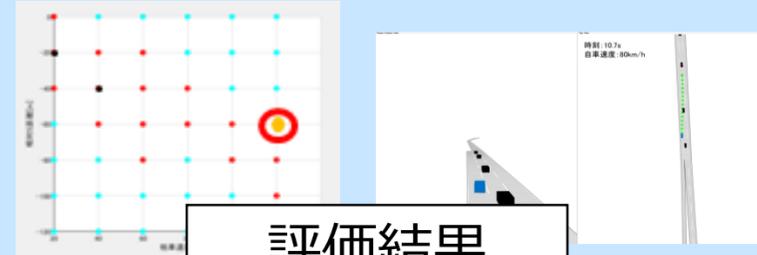
# MATLAB/Simulink+ADTを活用したシミュレーション環境の全体構成

## ユーザインタフェース

① 検証内容を設定できる  
⇒ MATLAB + ADT

パラメータ	値	単位	説明
1. 車	0.00	m	シミュレーション開始位置
2. 車長	5.0	m	車長の長さ
3. 車幅	1.8	m	車幅の長さ
4. 車高	1.5	m	車高の長さ
5. 車重	1500	kg	車重の長さ
6. 車速	10.0	m/s	車速の長さ
7. 車加速度	0.1	m/s <sup>2</sup>	車加速の長さ
8. 車減速度	0.1	m/s <sup>2</sup>	車減速の長さ
9. 車ブレーキ	0.1	m/s <sup>2</sup>	車ブレーキの長さ
10. 車エンジン	0.1	m/s <sup>2</sup>	車エンジンの長さ
11. 車タイヤ	0.1	m/s <sup>2</sup>	車タイヤの長さ
12. 車ホイール	0.1	m/s <sup>2</sup>	車ホイールの長さ
13. 車サスペンション	0.1	m/s <sup>2</sup>	車サスペンションの長さ
14. 車シフト	0.1	m/s <sup>2</sup>	車シフトの長さ
15. 車クラッチ	0.1	m/s <sup>2</sup>	車クラッチの長さ
16. 車ブレーキペダル	0.1	m/s <sup>2</sup>	車ブレーキペダルの長さ
17. 車アクセルペダル	0.1	m/s <sup>2</sup>	車アクセルペダルの長さ
18. 車ステアリング	0.1	m/s <sup>2</sup>	車ステアリングの長さ
19. 車エンジンブレーキ	0.1	m/s <sup>2</sup>	車エンジンブレーキの長さ
20. 車ブレーキペダル	0.1	m/s <sup>2</sup>	車ブレーキペダルの長さ
21. 車アクセルペダル	0.1	m/s <sup>2</sup>	車アクセルペダルの長さ
22. 車ステアリング	0.1	m/s <sup>2</sup>	車ステアリングの長さ
23. 車エンジンブレーキ	0.1	m/s <sup>2</sup>	車エンジンブレーキの長さ
24. 車ブレーキペダル	0.1	m/s <sup>2</sup>	車ブレーキペダルの長さ
25. 車アクセルペダル	0.1	m/s <sup>2</sup>	車アクセルペダルの長さ
26. 車ステアリング	0.1	m/s <sup>2</sup>	車ステアリングの長さ
27. 車エンジンブレーキ	0.1	m/s <sup>2</sup>	車エンジンブレーキの長さ
28. 車ブレーキペダル	0.1	m/s <sup>2</sup>	車ブレーキペダルの長さ
29. 車アクセルペダル	0.1	m/s <sup>2</sup>	車アクセルペダルの長さ
30. 車ステアリング	0.1	m/s <sup>2</sup>	車ステアリングの長さ

検証内容設定



評価結果

② 周辺環境を組み替える  
⇒ ADTの連成用  
ブロックセット等

ゼンリン地図データ

道路モデル

MC内製  
(@DSD)

or

PTV Vissim

他者行動モデル

周辺環境モデル

センサ・制御  
運動モデル

③ 車両システムを組み替える  
⇒ Simulink+ADT

行動計画モデル

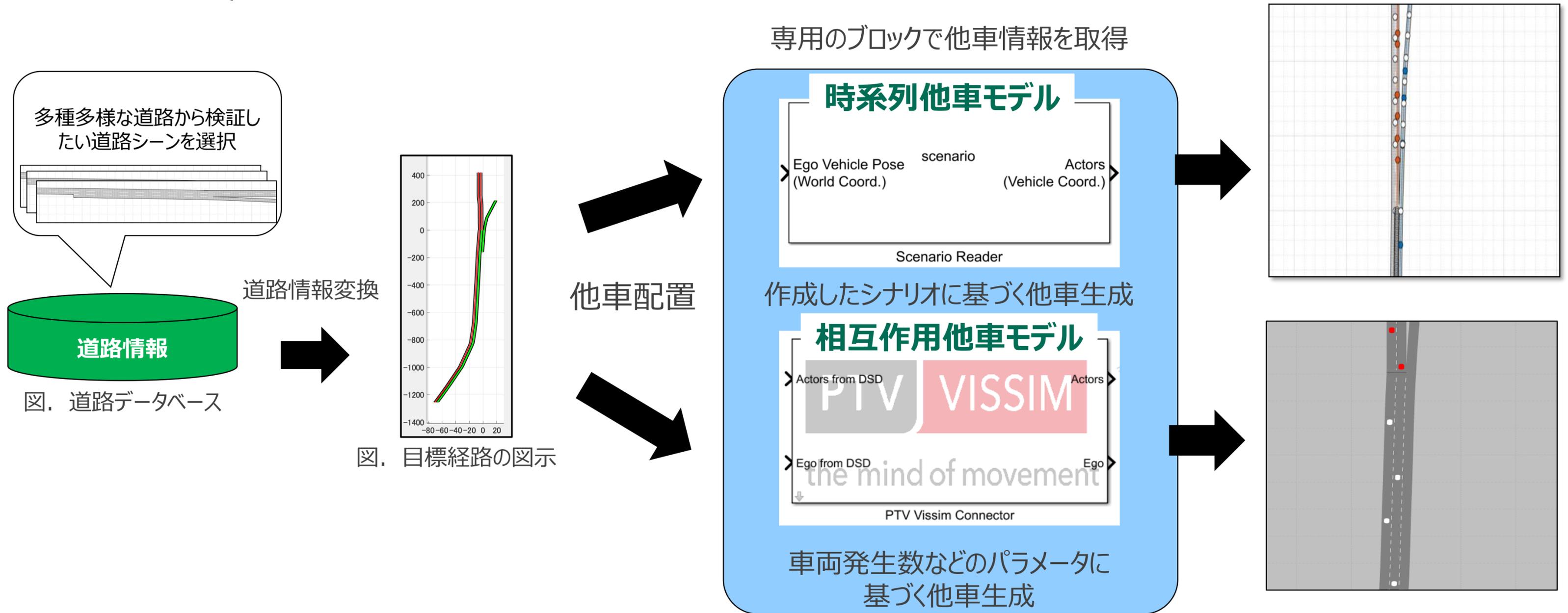
④ 技術アイデアの組み込み  
⇒ Deep Learning Toolbox™

技術アイデア

車両システムモデル

## ②周辺環境を組み替える⇒ADTの連成用ブロックセット等

- Vissim連成&OpenDRIVEインポートによる周辺環境の組み換え



達成：検証用途に合わせて、多様なシミュレーションツールと連成する仕組み

# PTV Vissim

- 交通流の再現
  - 道路追従、前方車追従、車線変更、信号認識等の機能が存在
- 複雑な道路形状への対応
  - 交差点に始まり、自転車レーン、公共交通路線、路上駐車等をモデル化
- シームレスな外部連携機能
  - 汎用的なCOM（起動時）インターフェース及び外部シミュレーターとの連携に特化したDriving Simulator Interfaceを活用することで、様々な外部シミュレータと容易に接続可能

PTV Vissim  
交通シミュレーション

マルチモーダル交通シミュレーション:

- 1つのソフトウェアであらゆる交通モードをシミュレーション
- 公共交通機関のモデリング
- 交通環境全体の詳細な表現
- 2Dおよび3D表示での視覚化
- 複雑な交通制御のシミュレーション
- エミッションモデリング

PTV GROUP [www.ptvgroup.com](http://www.ptvgroup.com)

(株)PTVグループジャパン様より提供

# MATLAB/Simulink+ADTを活用したシミュレーション環境の全体構成

## ユーザインタフェース

①検証内容を設定できる  
⇒MATLAB + ADT

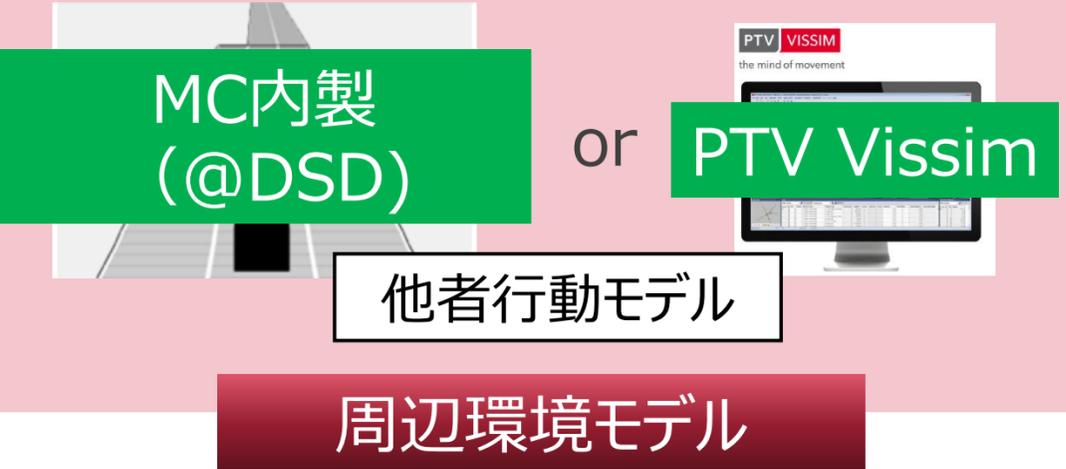


検証内容設定



評価結果

②周辺環境を組み替える  
⇒ADTの連成用  
ブロックセット等



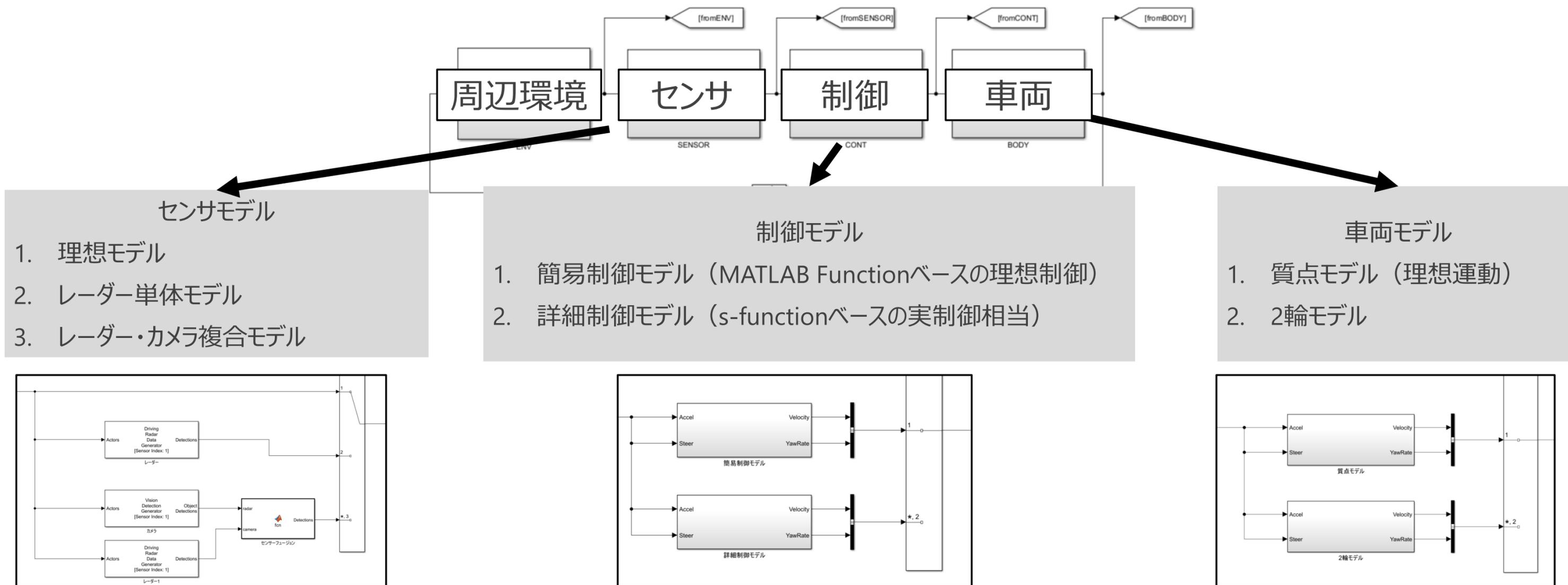
③車両システムを組み替える  
⇒Simulink+ADT



④技術アイデアの組み込み  
⇒Deep Learning Toolbox™

### ③ 車両システムを組み替える⇒Simulink+ADT

- Simulinkを活用し、各コンポーネントモデルへの関係や役割を明確化



達成：検証用途に合わせて、システムの組み替えが容易にできる仕組み

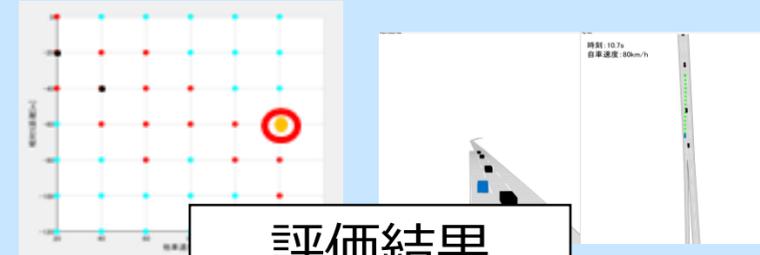
# MATLAB/Simulink+ADTを活用したシミュレーション環境の全体構成

## ユーザインタフェース

① 検証内容を設定できる  
⇒ MATLAB + ADT



検証内容設定



評価結果

② 周辺環境を組み替える  
⇒ ADTの連成用  
ブロックセット等

ゼンリン地図データ

道路モデル

MC内製  
(@DSD)

or

PTV Vissim

他者行動モデル

周辺環境モデル

③ 車両システムを組み替える  
⇒ Simulink+ADT

センサ・制御  
運動モデル

④ 技術アイデアの組み込み  
⇒ Deep Learning Toolbox™

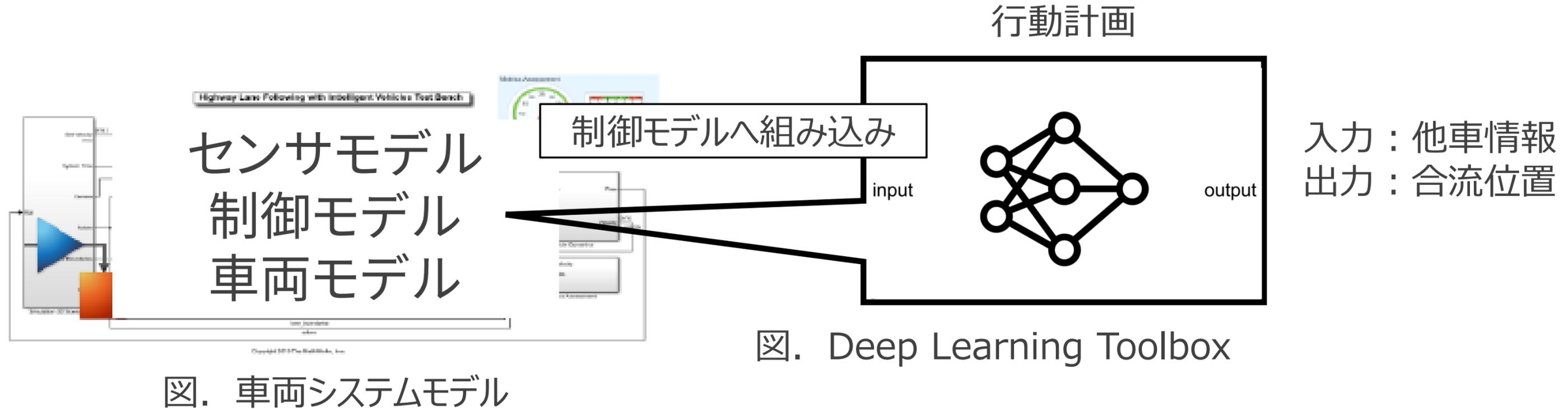
行動計画モデル

技術アイデア

車両システムモデル

## ④技術アイデアの組み込み⇒Deep Learning Toolboxの活用

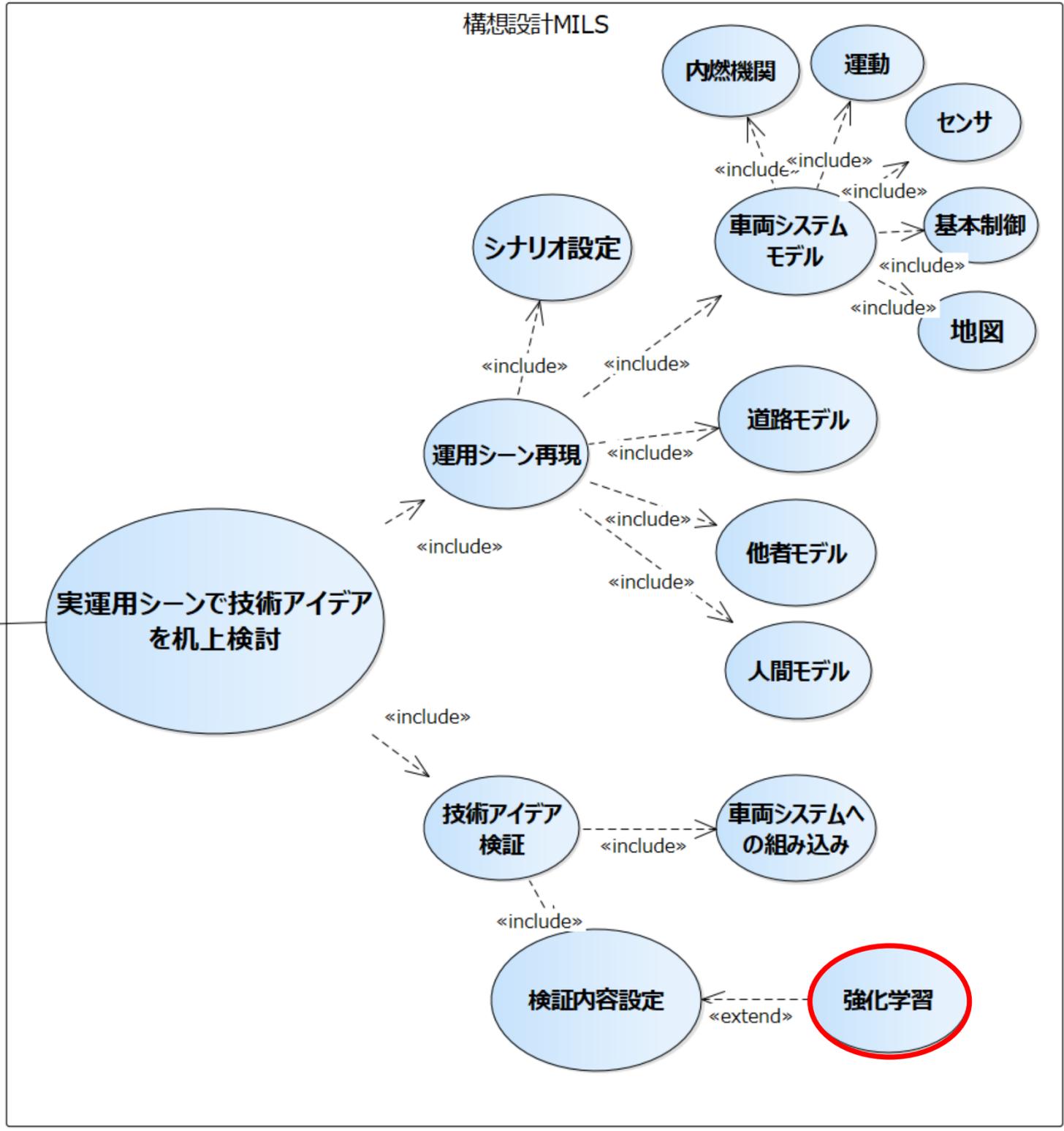
- 課題：技術アイデアの育成環境としての機能（主に強化学習の学習機能）を有しておらず、学習と検証が独立している
  - 検証環境で技術課題を発見した場合は別環境の学習環境へ手戻りしており、非効率的な開発になる
- 解決方針：Deep Learning Toolboxを活用し、学習（技術育成）も同時に行えるシミュレーション環境への育成



達成：Deep Learning Toolboxを活用し、学習環境としての活用も可能に

# シミュレーション実行の様子





# MILS運用の結果及び効果

## 合流位置判断ロジックの強化学習 + 検証

# 合流位置判断ロジックの強化学習 + 検証

## • ロジック概要

- 入力：他車の位置・速度
- 出力：目標合流位置
  - ✓ 制御モデルは目標合流位置に対して、加減速を行って車線変更する

## • 学習条件

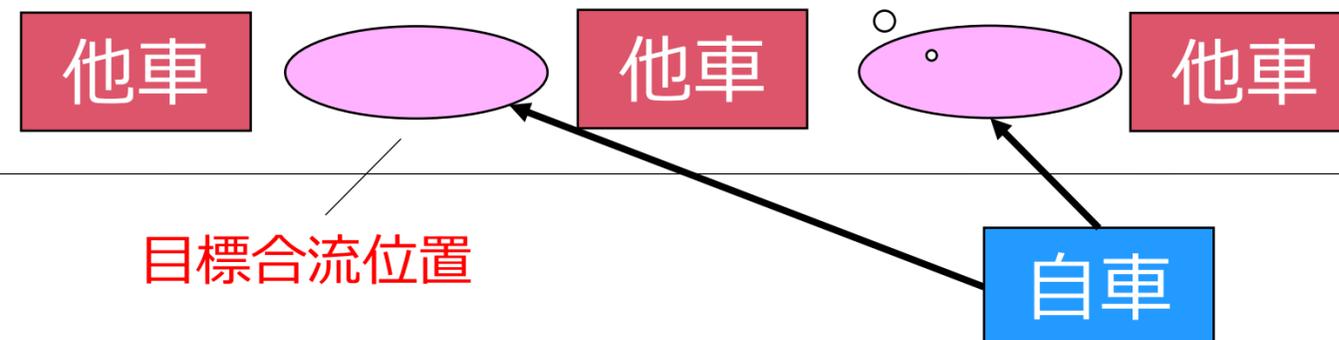
### ➤ 学習データ

- ✓ 本線他車両の位置等を変更した200シナリオを用意。  
1シナリオ実行（データ取得）⇒1000回学習（オフライン学習）のセットを繰り返して学習
- ✓ 報酬：合流成功（合流車線終了位置までに合流）、  
後方車両の減速度、後方車両との車間距離

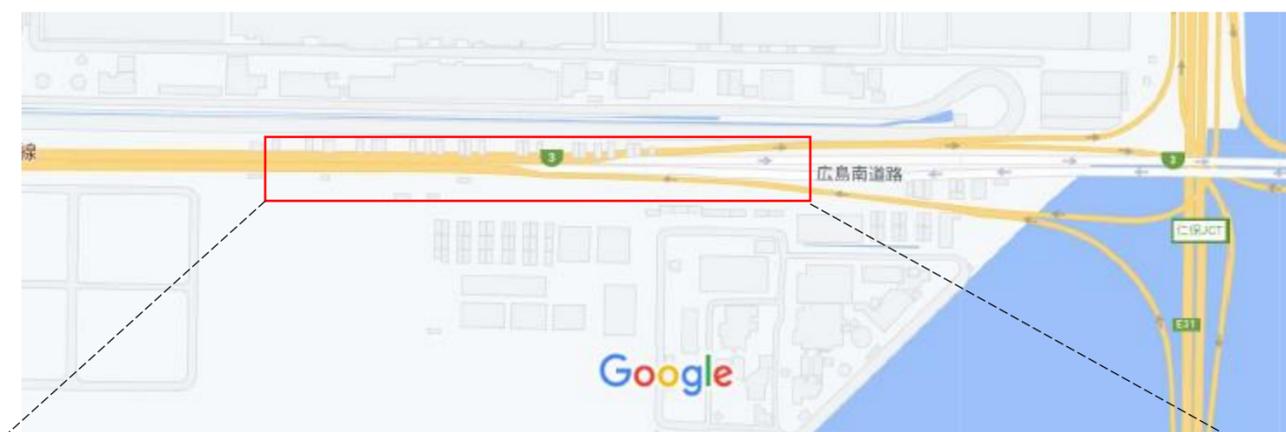
## • 検証条件

- 他車：本線他車両をランダム生成
  - ✓ 条件を分散させてPTV Vissimで自動生成
- 道路：広島高速3号線の仁保JCTの合流路

どの車間を目標にすれば安全に車線変更できるか



技術イメージ図



引用元：Google社「Google ストリートビュー」

# 検証結果

表：学習回数毎の合流成功確率

	10000回学習時点	30000回学習時点	40000回学習時点
合流成功確率 (ランダムな40シナリオをもとに算出)	0%(0/40)	37.5%(15/40)	90%(36/40)

- 目標合流位置がハンチングして位置が定まらず、合流できない。

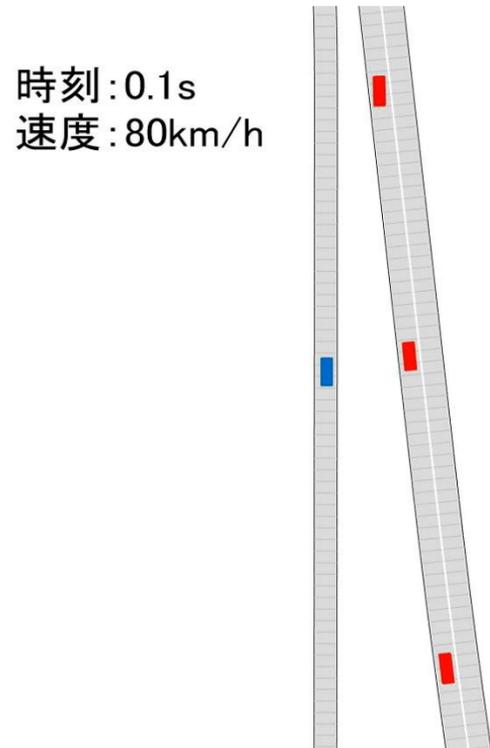


図. 10000回学習時点

- 目標合流位置が安定化してきたが、選定する位置が適切でなく、合流できていない。

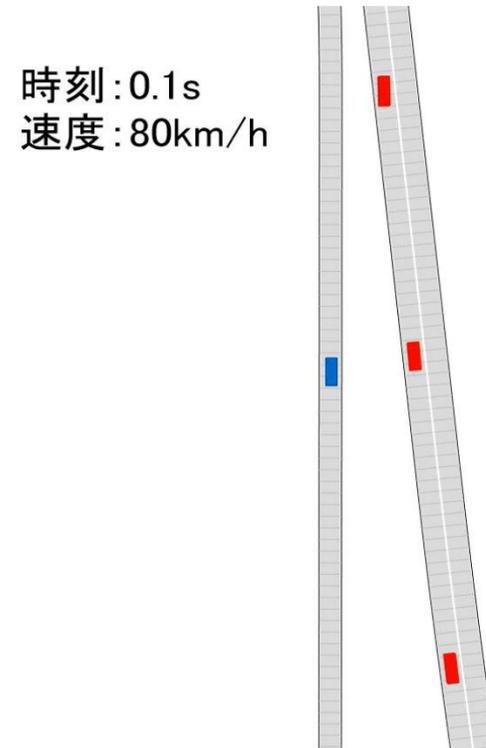


図. 30000回学習時点

- 目標合流位置が安定化しつつ、適切な位置を選択できており、合流ができています。

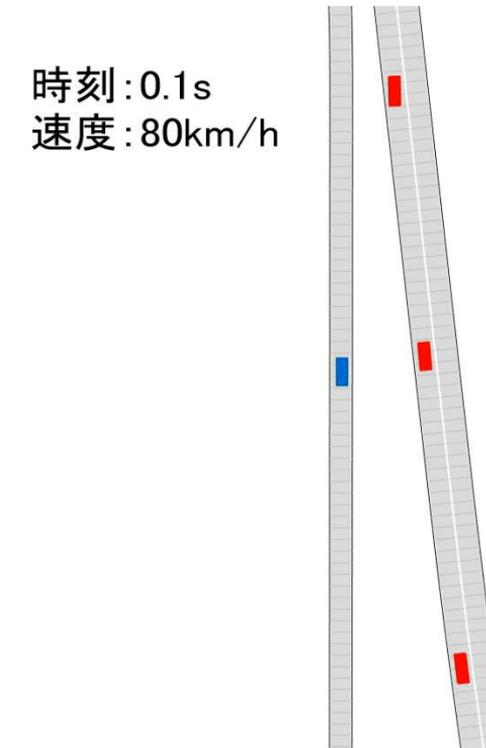


図. 40000回学習時点

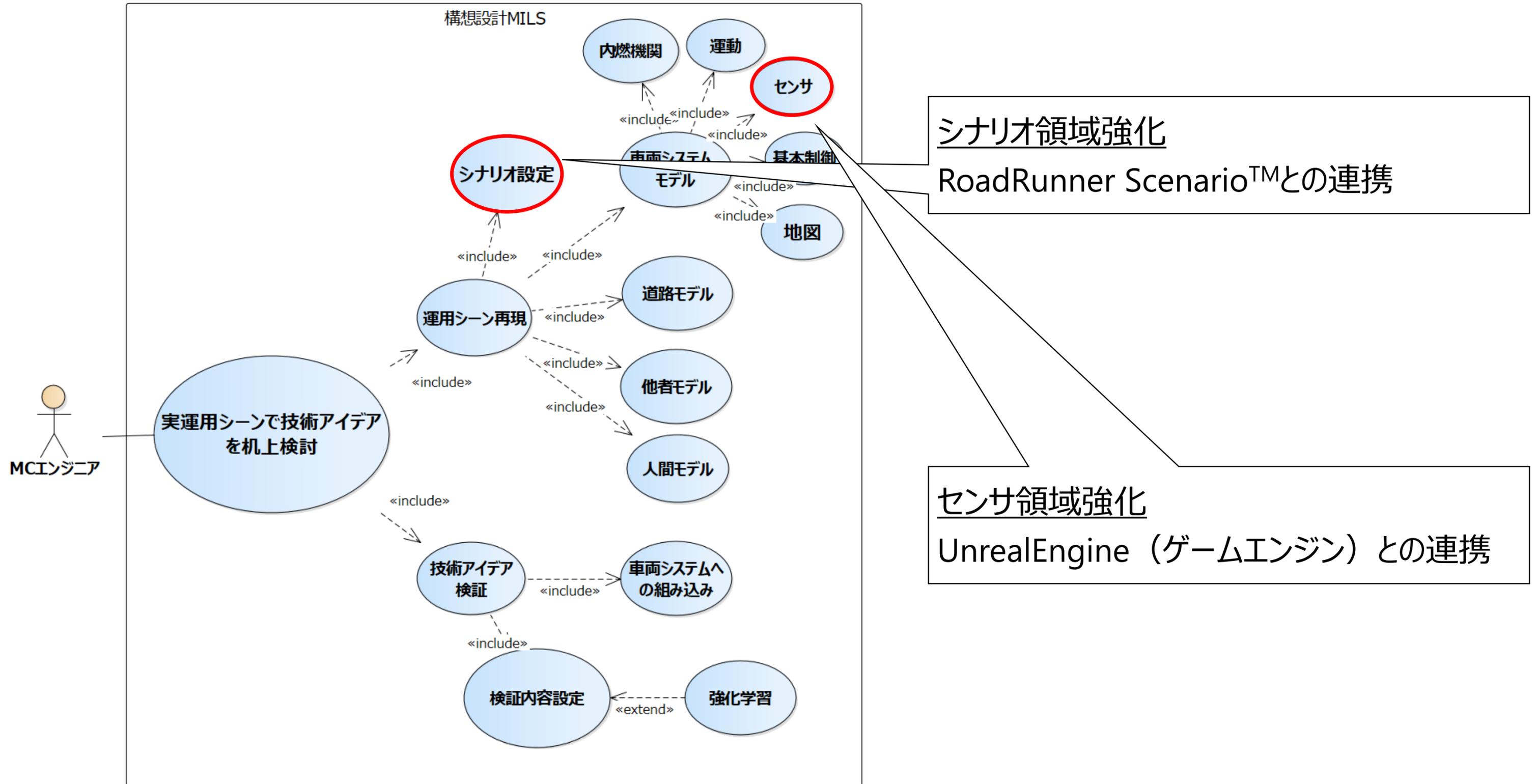
達成：構想段階で技術アイデアの有効性を確認

## まとめ

- 道路モデル自動生成
  - ゼンリンデータコム、ゼンリンの地図データを利用し「合流路モデル」の自動生成機能を構築
    - ⇒検証の幅を拡大
- 連成シミュレーションの構築
  - 検証に合わせて、車両システム（センサ・運動モデル）、周辺環境、技術アイデアといったシミュレーション環境の構成を可変にできる仕組みを構築
    - ⇒検証用途に合わせて、環境の構成を変更して効率的な開発を行える仕組みづくり
- MILS運用の結果及び効果の確認
  - 合流位置判断ロジックの学習と検証を実施
    - ⇒学習モデルの判断精度が向上していく過程を確認

達成：技術の構想段階における検証を実施するシミュレーション環境の構築・運用

# 今後の展望紹介



# シナリオ領域強化

＜課題＞ ADTに用意された「ドライビングシナリオデザイナーアプリ」では、他者の挙動を時系列で設定する。

⇒道路が変更された場合などは再設定の必要あり。

＜解決方針＞ RoadRunner Scenarioを活用すると経路が自動で車線の中心に沿って設定される。また、イベントベースの挙動も設定でき、経路設定の簡略化や目標のシナリオ作成に対して効果を期待できる。

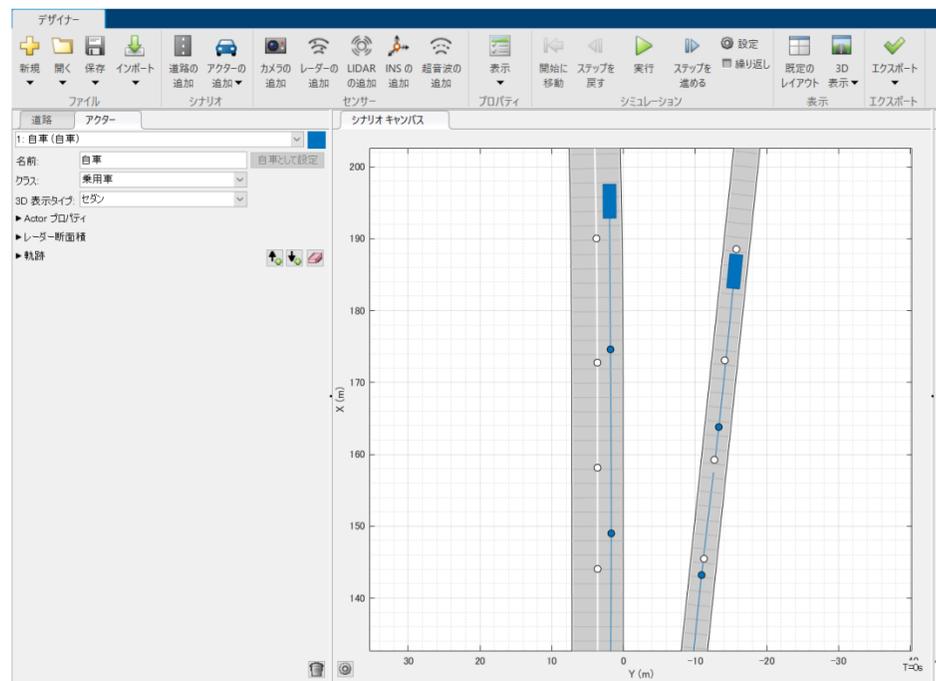


図. ドライビングシナリオデザイナーアプリ

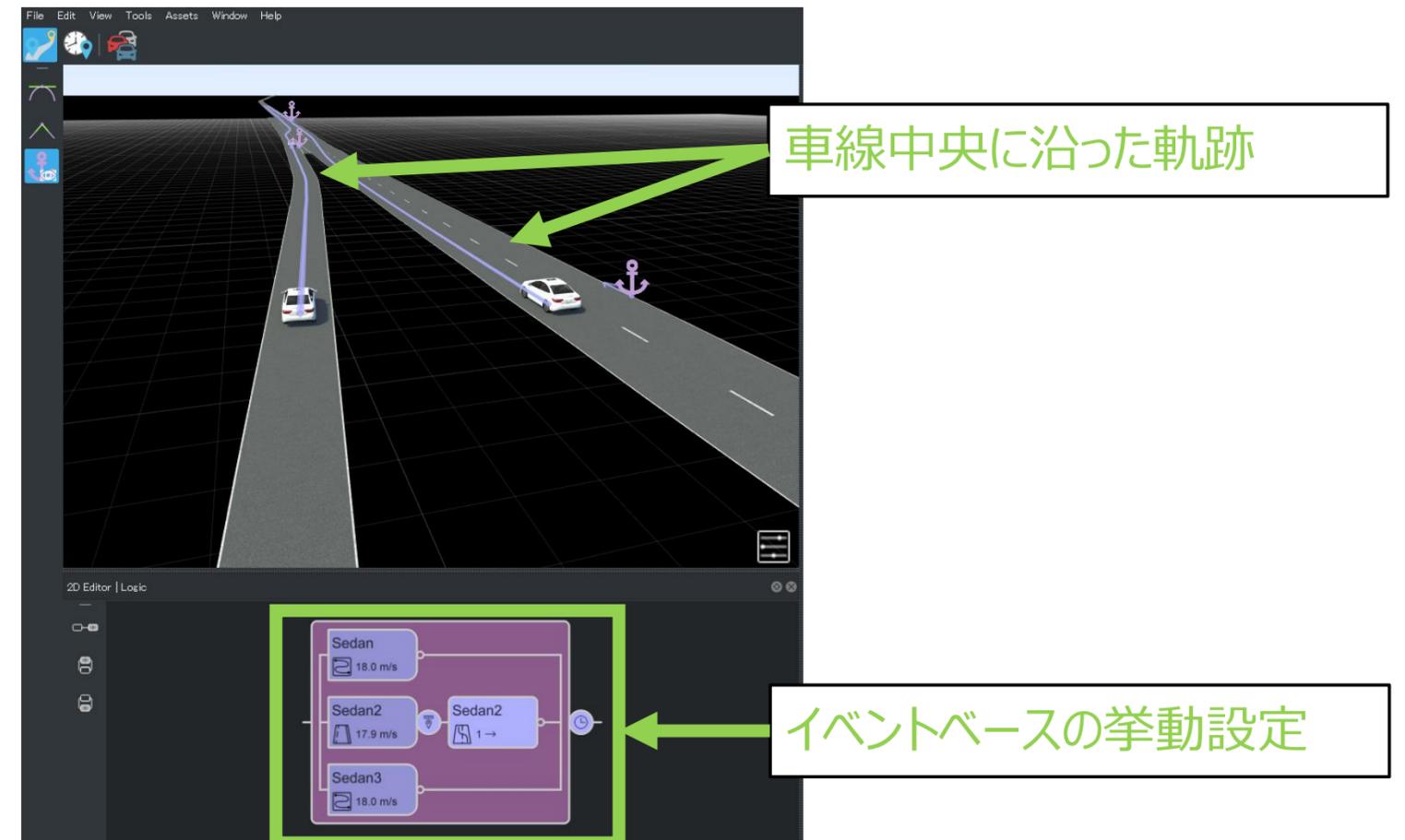
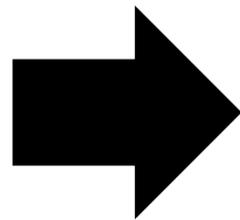


図. RoadRunner Scenario

# センサ領域強化

<課題> 直方体ベースではフォトリアリスティックな環境でのセンサシミュレーションができない。

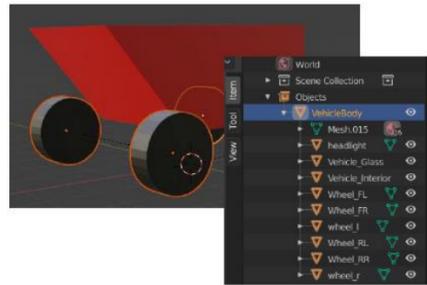
⇒カメラモデルを実装した、センサシミュレーションが困難

<解決方針> ADTが用意しているゲームエンジンのUnrealEngineとSimulinkとの連携機能を採用

⇒フォトリアリスティックな環境で検証することが可能になり、センサシミュレーションを精度良く実施できる。

## Unreal連携シミュレーション – 新機能

### カスタムメッシュのサポート



任意の車両形状を設定可能

[Prepare Custom Vehicle Mesh for the Unreal Editor](#)  
Automated Driving Toolbox™

### 車両のライト・方向指示器制御



3D Vehicleブロックでヘッドライト等の色・強度や向き等が設定可能

[Simulation 3D Vehicle with Ground Following](#)  
Automated Driving Toolbox™

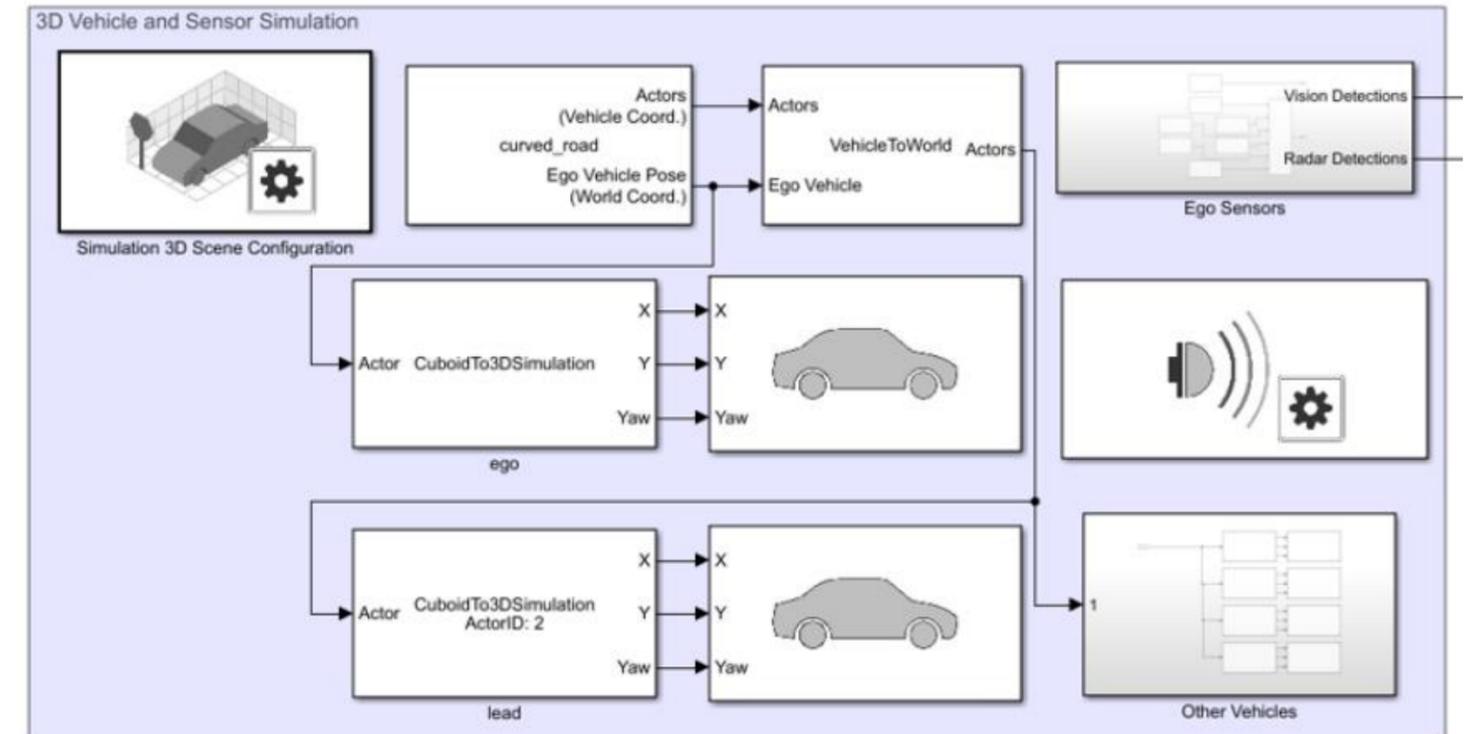
### 天候や太陽の位置等制御



3D Scene Configurationブロックで太陽の照度や位置、雨・霧の強弱が指定可能

[Simulation 3D Scene Configuration](#)  
Automated Driving Toolbox™

図. UnrealEngineを連携させた機能



Copyright 2019-2020 The MathWorks, Inc.

図. UnrealEngine連携用Simulinkブロック

# Mathworks様への要望

- 自動運転向けシミュレーション環境における各種モデルのライブラリ化
  - 車両領域外の道路や標識等のモデル、車両領域のセンサ・制御・運動モデルをライブラリして用意し、GUIにて各種モデルを選択すると、Simulinkモデルが自動で構築され、簡単に実行ができる仕組み
    - ⇒ 技術者の関心領域のみ（例：行動計画）モデルの組み替えを行い、素早くシミュレーションが行える

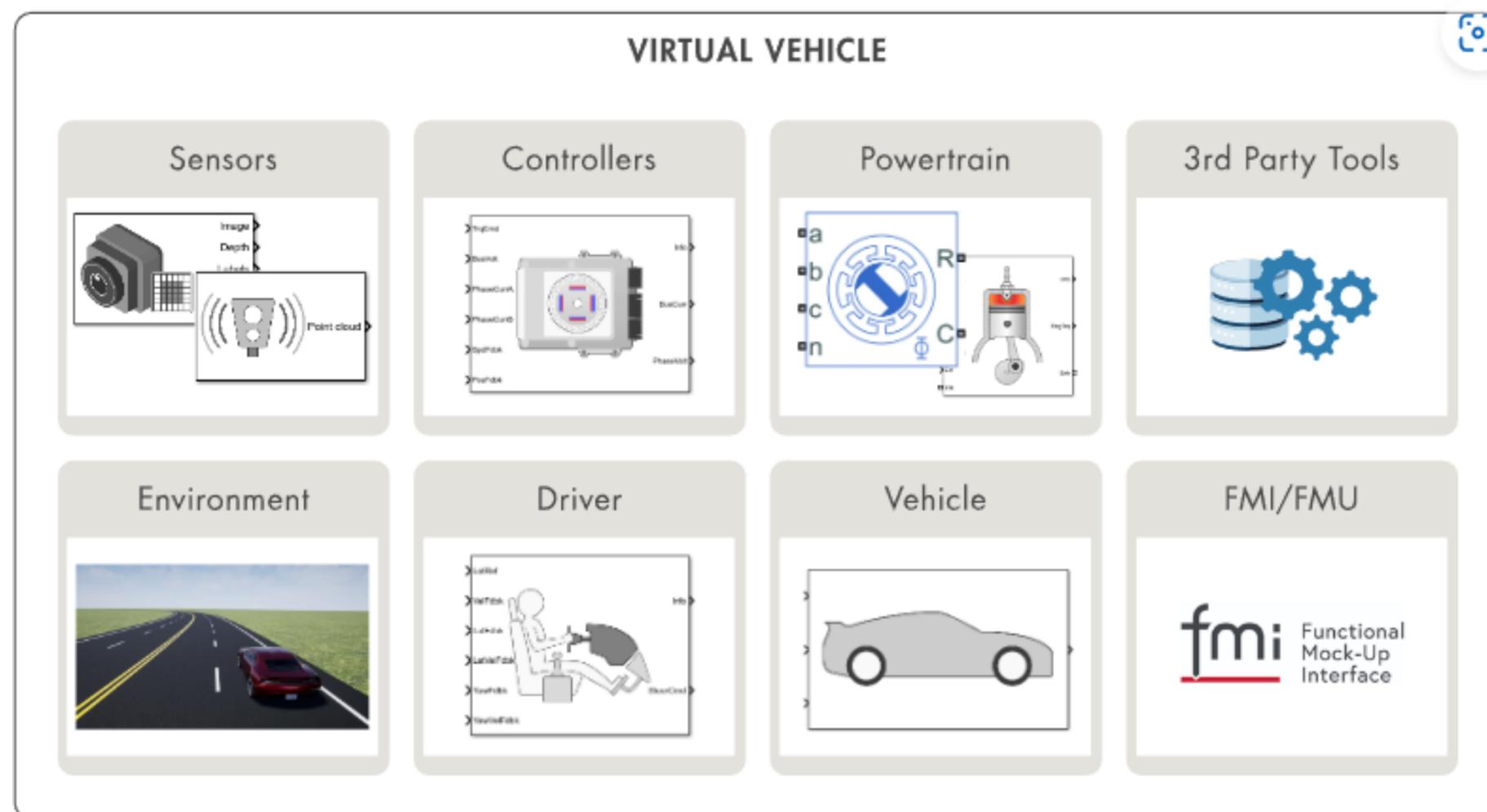


図. Virtual Vehicle Composer



ありがとうございました



**mazda**