シミュレーションより厳密な検証手法による フロントローディングでの設計品質向上

（株）構造計画研究所
製造企画マーケティング部

構造計画研究所 KOZO KEIKAKU ENGINEERING Inc.
株式会社　構造計画研究所
〒164-0001 東京都中野区中央4-5-3

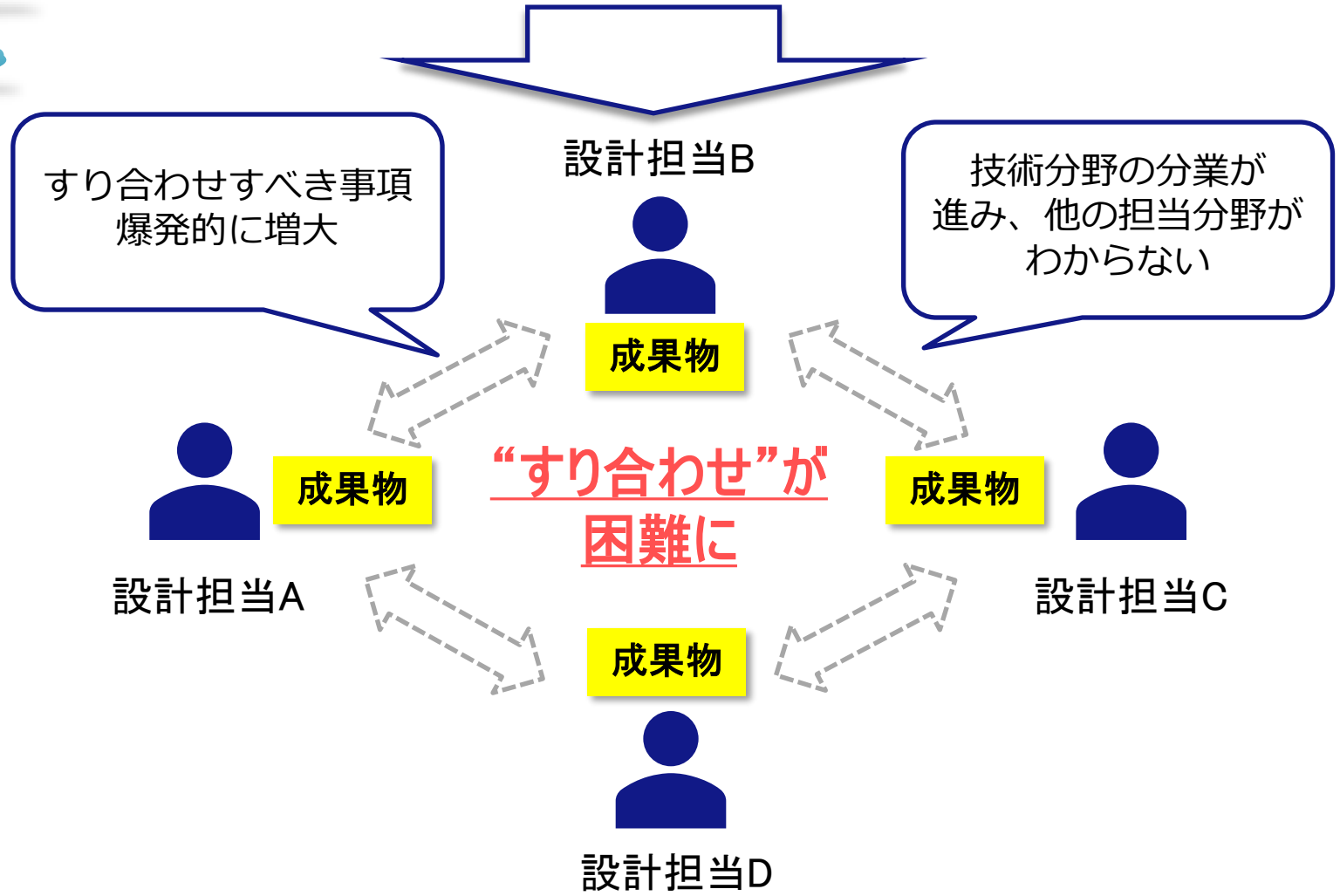
❒Agenda

- ご提案の背景
  - 従来開発とその課題
- システムズエンジニアリング実践の必要性
  - MBSE導入における課題
  - SysMLモデル検証の必要性
  - シミュレーションとモデル検査の違い
  - 設計者自身が検証できるツール環境
- サンプルモデルによるモデル検査適用事例のご紹介
  - クルーズコントローラのモデル化、検証例
- まとめ

❒従来開発とその課題

自動運転
コネクテッド
電動化

システムの大規模化・複雑化・分業化に伴って…



□システムズエンジニアリング実践の必要性

上流工程で
全体を俯瞰した上で
要求分析から設計、開発までを
一貫して行う必要がある

⇩

システムズエンジニアリング（MBSE）の実践

❒MBSE導入(SysMLモデリング)における課題

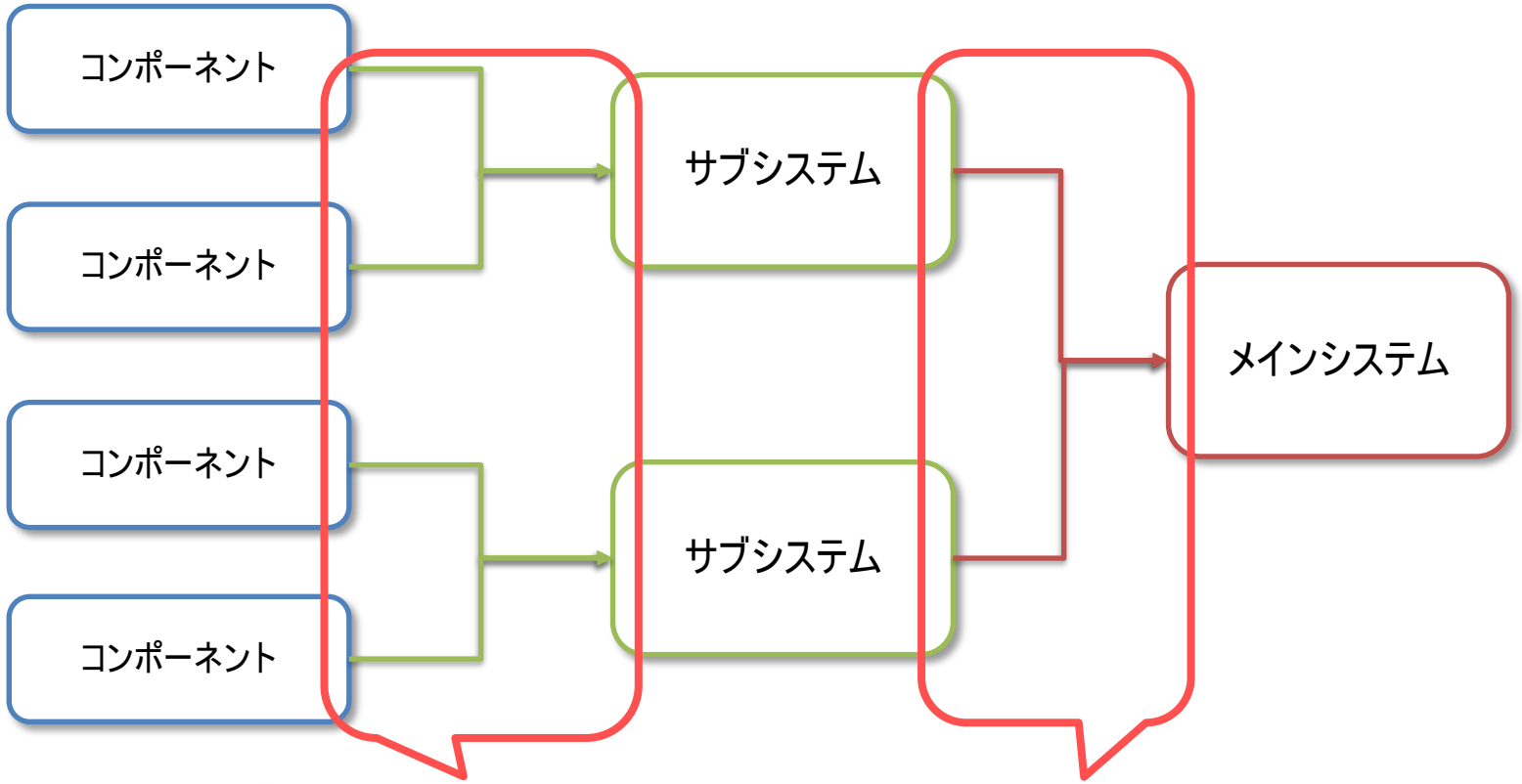
設計が要求仕様を満たしているのかチェックが難しい

SysMLを書くことはできたが、それだけで終わってしまっている

複数の制御の組み合わせで機能を実現している制御のチェックが難しい

□なぜシステムレベルでの設計・検証が重要なのか？

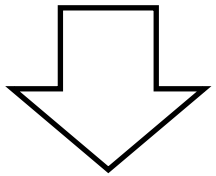
コンポーネント単体が正常に動作する ≠ **統合システムが正常に動作する**



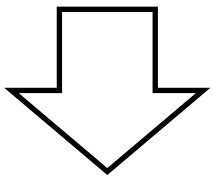
サブシステム・メインシステムに統合する際に
不具合は発生しやすく、これらは人手では発見困難

❒SysMLモデル検証の必要性

SysML等を用いて全体を俯瞰した設計を行い、
人手でチェックするだけでは、発見困難な不具合は見逃してしまう



人手やレビューではチェックが困難なシステム設計を
正しく検証するためにはどうすれば良いか？



**SysMLモデルの検証を属人化せず、
「システマティックに」かつ「厳密に」検証する手法が必要**

❒設計情報の検証について

■SysMLで**モデル化するだけではなく、動かして検証**することが有効。

- ❒モデリングツールの機能等でシミュレーションできる環境が必要
- ❒但し、**シミュレーションだけでは抽出しきれない不具合がある**
  - ➡システム設計段階で不具合を抽出できないと、
    - ☑不具合が混入した設計情報を下流に流してしまい、大きな手戻りが発生する
    - ☑テスト工程でも発見困難な不具合を抽出できず、重大なインシデント発生に繋がる

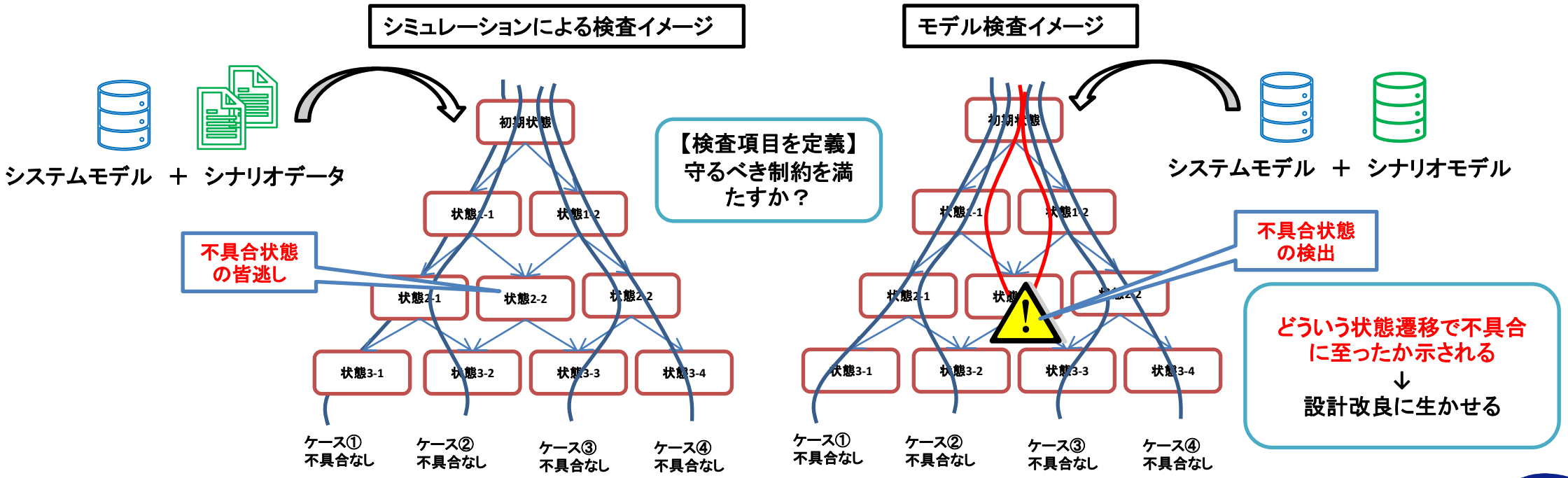
⇩

シミュレーションの限界を補うため、
より厳密な検証を行える「モデル検査」の併用が有効

□シミュレーションとモデル検査の違い

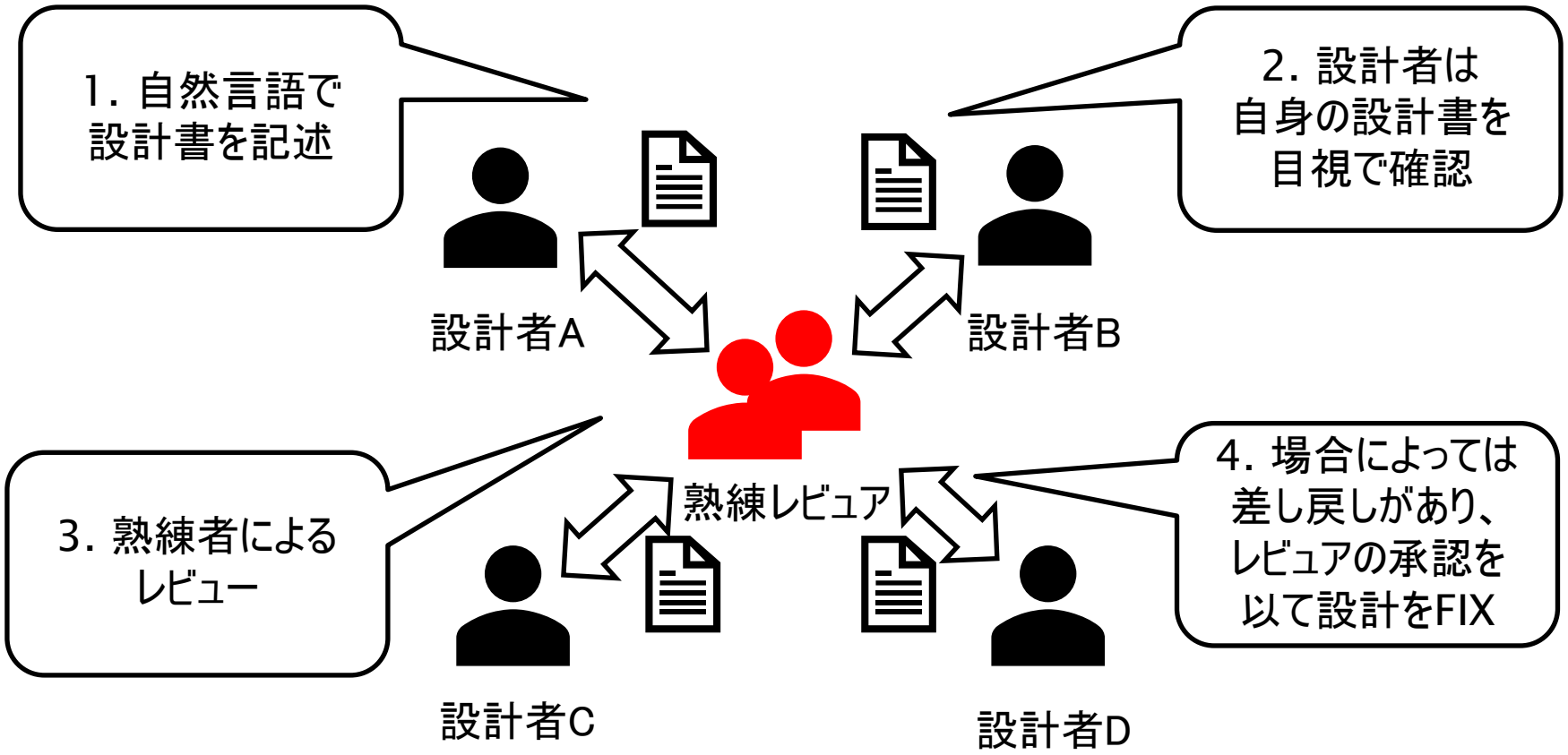
■「シミュレーション」と「モデル検査」

比較項目	シミュレーション	モデル検査
再現するシナリオの準備	人手でシナリオを複数ケース準備する（システムモデルへのINPUT）	外部環境の振る舞いモデルを作成する（システムモデルと並列の外部環境モデルを組み込む）
シナリオ再現に要する実行回数	シナリオ数分の実行が必要	1回の実行で起こりうる振る舞いを網羅的に再現する
起こりうる振る舞いの網羅性	定義したシナリオに依存する	外部環境モデルが取りうる振る舞いは網羅される



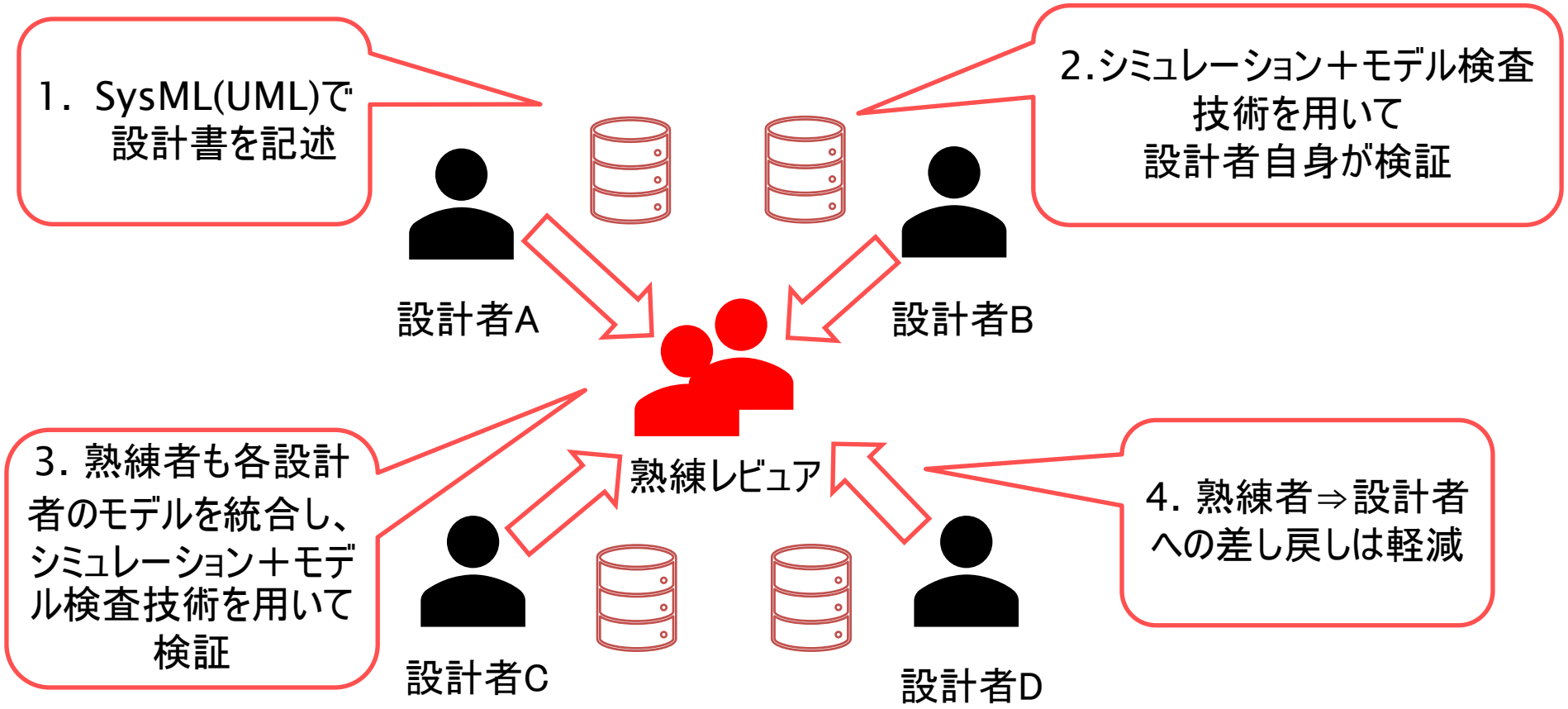
❐本セッションでお伝えしたいこと
～「モデリング＋ツールによる検証」で設計者自身が検証できる環境を！～

従来のシステム設計と検証方法



❐本セッションでお伝えしたいこと
～「モデリング＋ツールによる検証」で設計者自身が検証できる環境を！～

KKEが提案する設計の姿



❐本セッションでお伝えしたいこと
～「モデリング＋ツールによる検証」で設計者自身が検証できる環境を！～

KKE提案の設計・検証手法のメリット

✓ **仕様に関する理解齟齬軽減**
- UML/SysMLモデルを用いることで、仕様に関して曖昧な表現がなくなる。

✓ **熟練レビュアへの依存軽減**
- シミュレーション＋モデル検査を導入することにより、属人性が軽減する。

✓ **設計⇔レビューの手戻り軽減**
- 設計者自身が検証できるので、レビュアの負荷軽減＋手戻りも軽減する。

✓ **発見困難な不具合を早期発見**
- モデル検査技術を用いると、人間系では発見困難な不具合も検出可能になる。

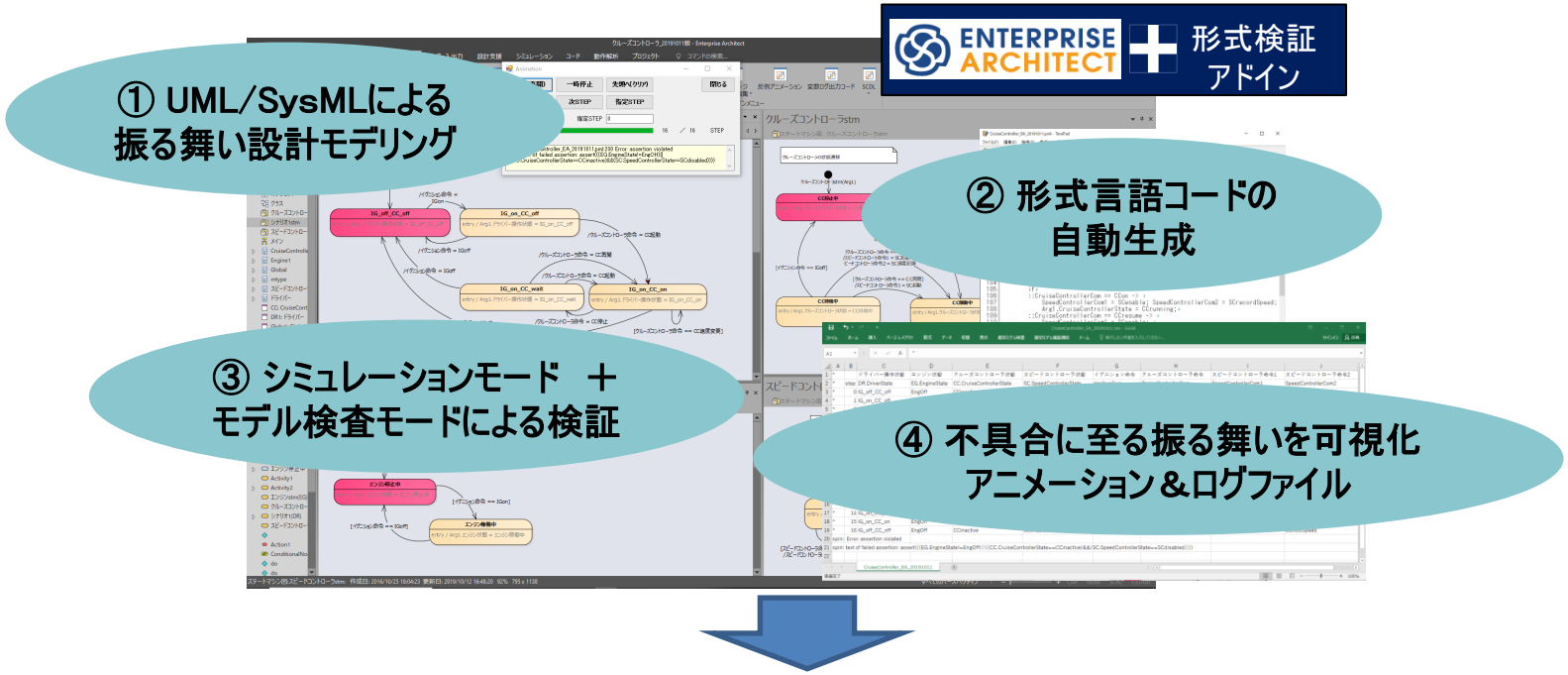
ENTERPRISE ARCHITECT ＋ DynaSpec によって実現

❑本セッションでお伝えしたいこと
～「モデリング＋ツールによる検証」で設計者自身が検証できる環境を！～

モデルベース形式検証ツール「DynaSpec」はSparx Systems社のモデリングツールEnterprise Architectのアドインツールであり、以下の特徴を持ちます。

- EA上の操作でモデリング～シミュレーション＋モデル検査～結果可視化まで可能
- 専用言語習得不要で、形式検証(モデル検査)を実施可能

DynaSpec（Dynamic Specification：動く仕様）



導入ハードルが高かった形式検証を開発現場へ！

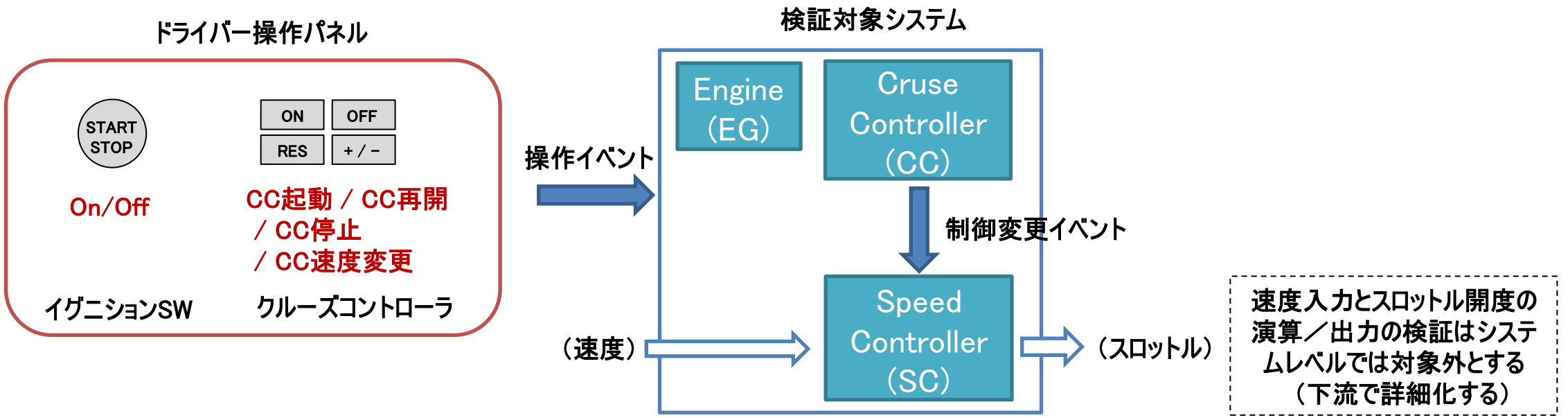
SysMLサンプルモデルによる形式検証適用例のご紹介

- クルーズコントローラシステムのモデル化と検証例

❒ DynaSpec適用対象システム ： クルーズコントローラ

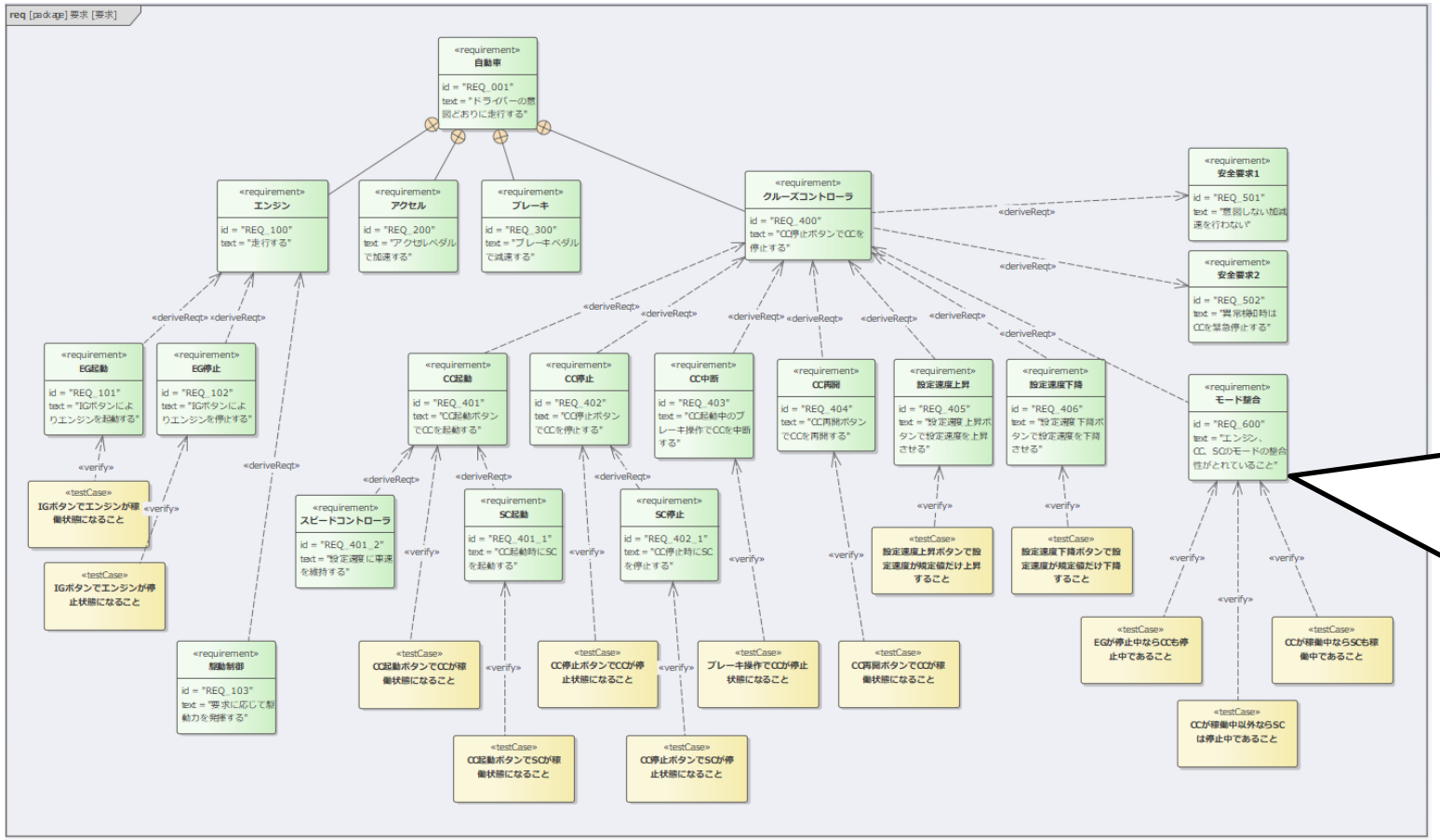
- システム設計
  - サブシステム構成と制御イベントを設計
  - 各コントローラの制御仕様を設計（状態遷移図中心）
- 検証方針
  - ドライバーが想定外の操作を行っても制御が破綻しないかをモデル検査で検証
    - ➡ サブシステム間の**制御モードの不整合**、**デッドロック**が起きないか等

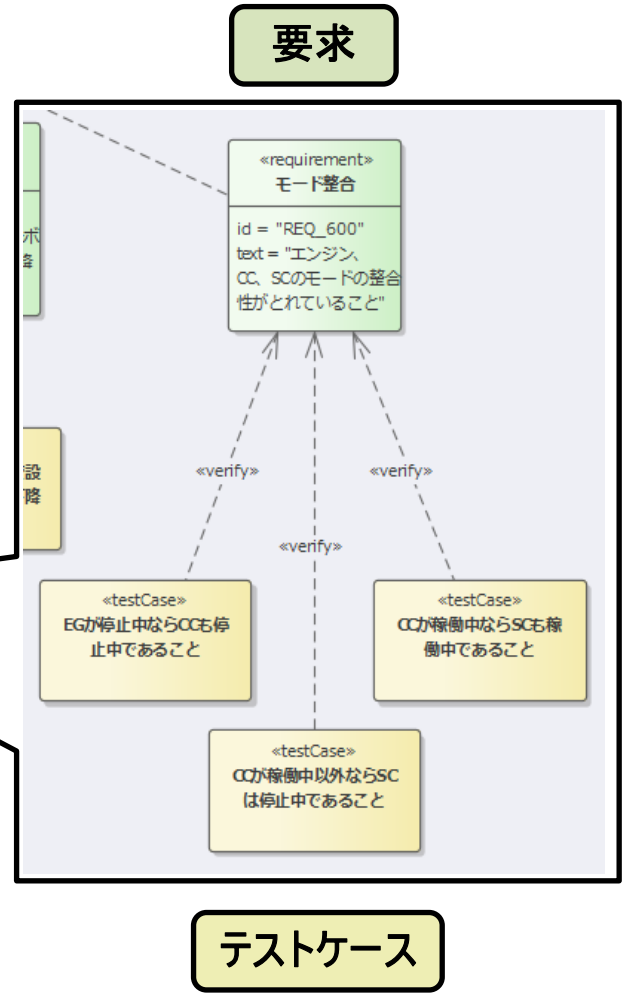
サンプルモデルは、中島震著
「 SPINモデル検査」のクルーズコントローラ
システムを参考にしています。



❒ 要求図

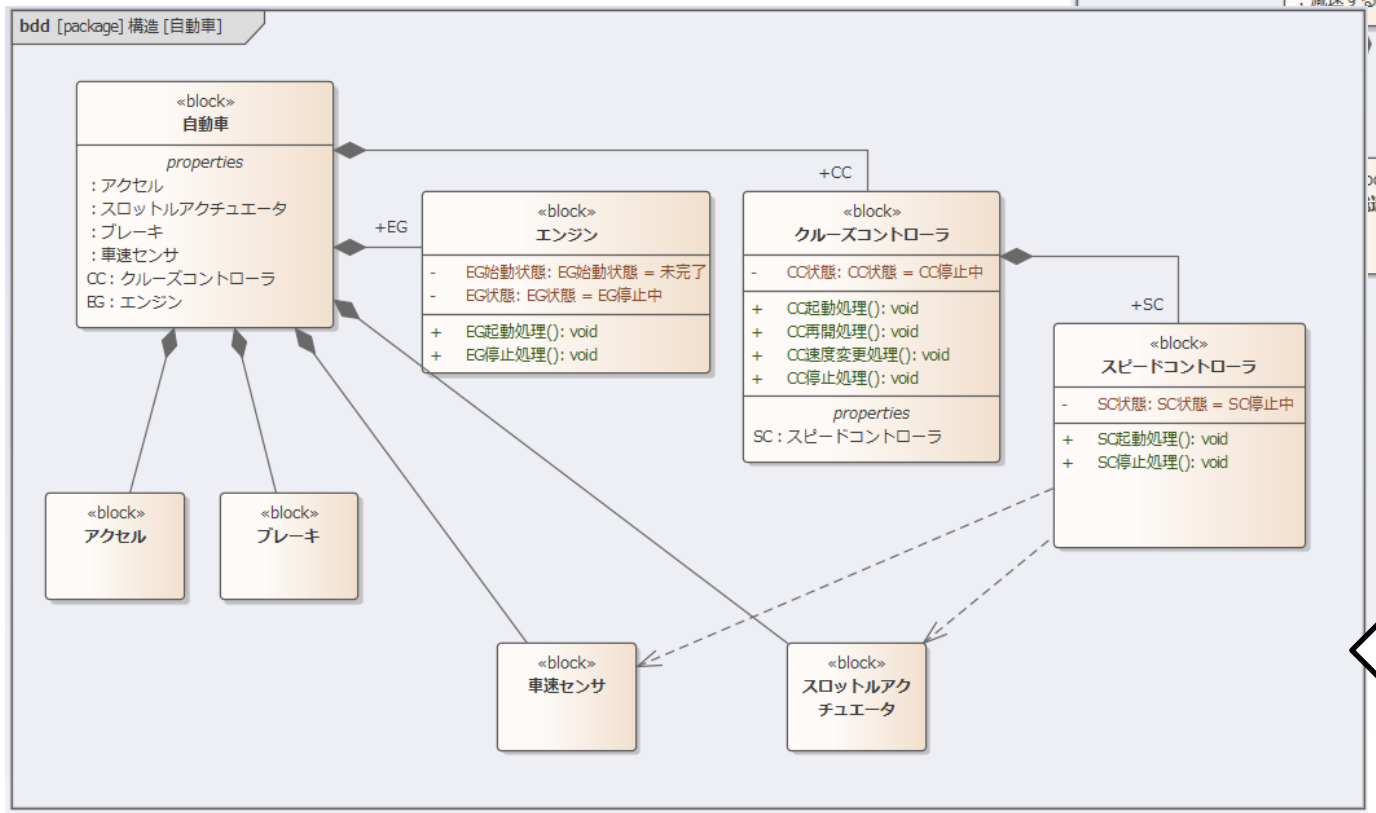
- ユースケースで定義した機能に関する要求に加え、非機能要求、安全要求、制約等も含めて要求図を定義する。
- 要求を満たすかどうかを検証するため、テストケースと関連づけて定義する。
  - テストケースは別の検査用ダイアグラムにおいて検査式として再定義する。

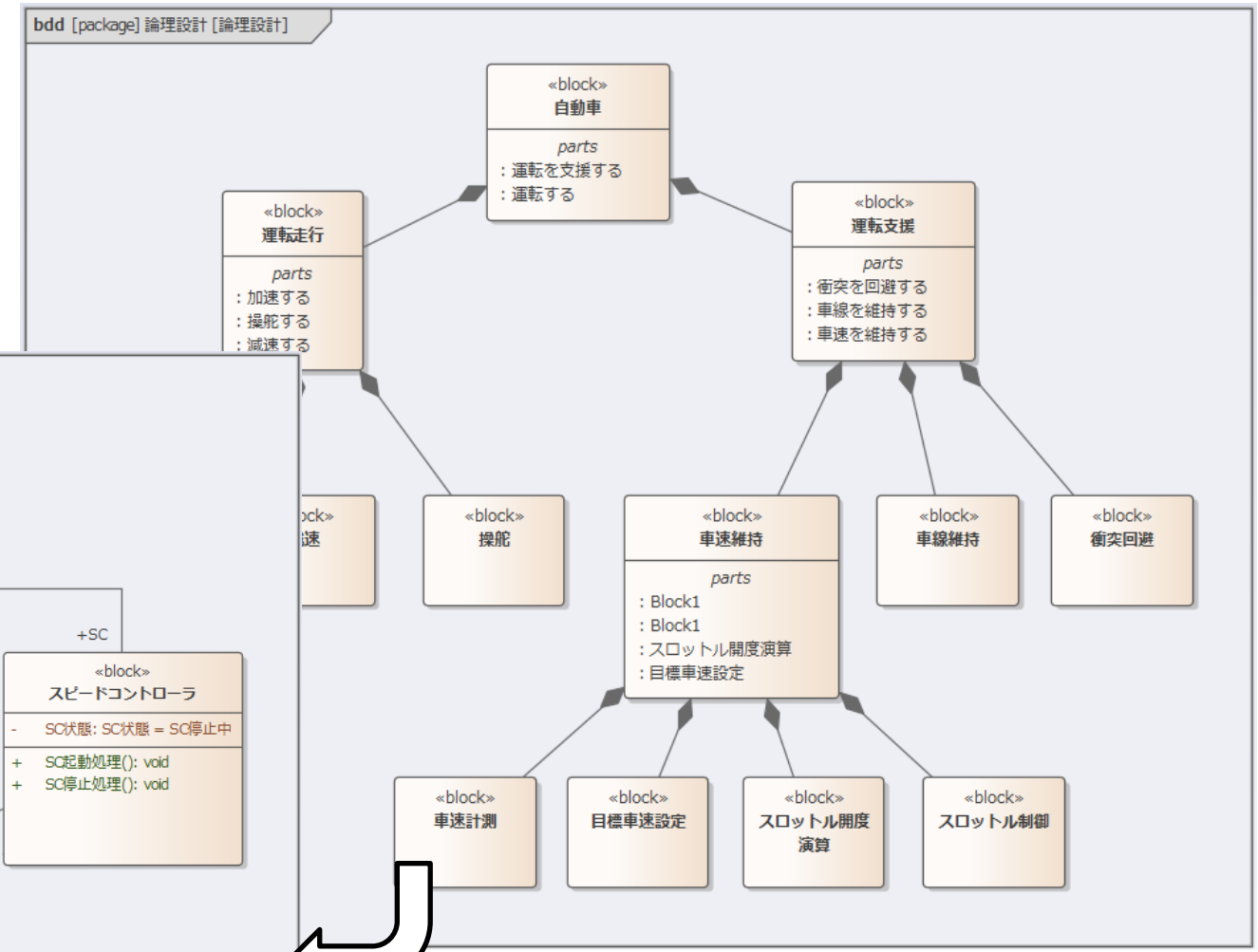




❒ ブロック定義図

- システムの構成要素をブロックで定義する。
  - 論理設計の段階では機能視点でシステムの構成を考える
  - 物理設計では物理的な部品の視点でシステムの構成を考える
    - ➡ 部品への機能の割り当て





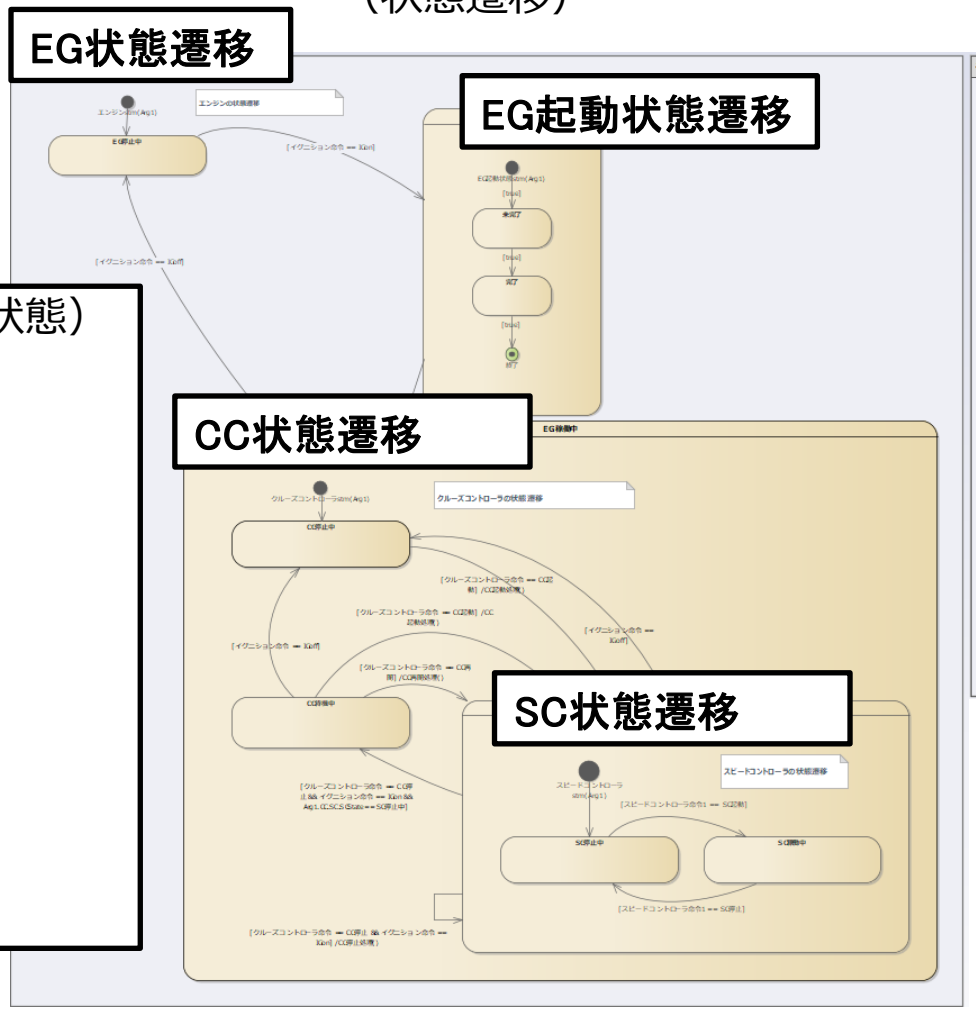
論理設計から物理設計へ

❒ ステートマシン図／アクティビティ図

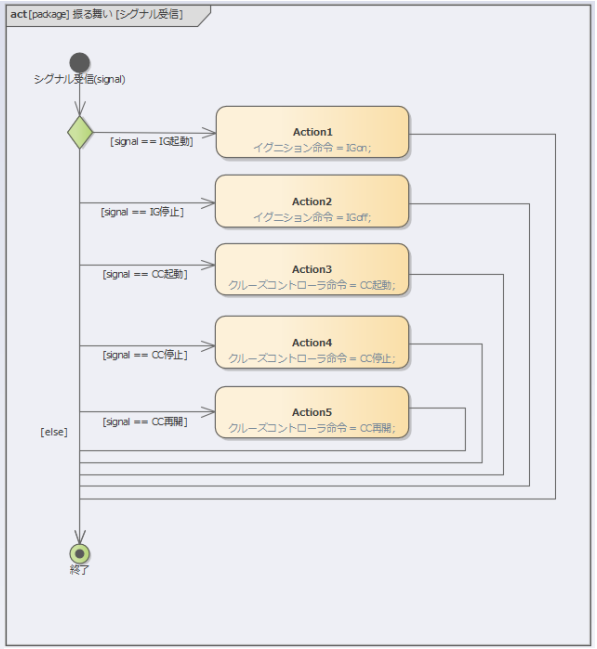
■ システム構成要素の振る舞い図を定義する。

- ■ エンジンコントロールユニット（↓EG状態）
  - ❒ EG停止中
  - ❒ EG起動中（↓EG起動状態）
    - ❒ 未完了
    - ❒ 完了
  - ❒ EG稼働中（↓CC状態）
    - ❒ CC停止中
    - ❒ CC稼働中（↓SC状態）
      - ❒ SC停止中
      - ❒ SC稼働中
    - ❒ CC待機中

◆ ステートマシン図
（状態遷移）



◆ アクティビティ図
（処理ロジック／シーケンス）



❑DynaSpecによりSysMLモデルを形式検証するための実行ステップ

要求定義・分析

要求図

ユースケース図

✓ 要求の定義に加え、要求を満たすかどうかを検証するためテストケースと関連づける。

システム設計

【システム設計モデル】

ブロック定義図

ステートマシン図

アクティビティ図

✓ 要求を実現するためのアーキテクチャを設計する（構造、振る舞い）。

形式検証モデルへ拡張

＋ 【検証条件モデル】

ブロック定義図

ステートマシン図

アクティビティ図

検証対象ダイアグラム

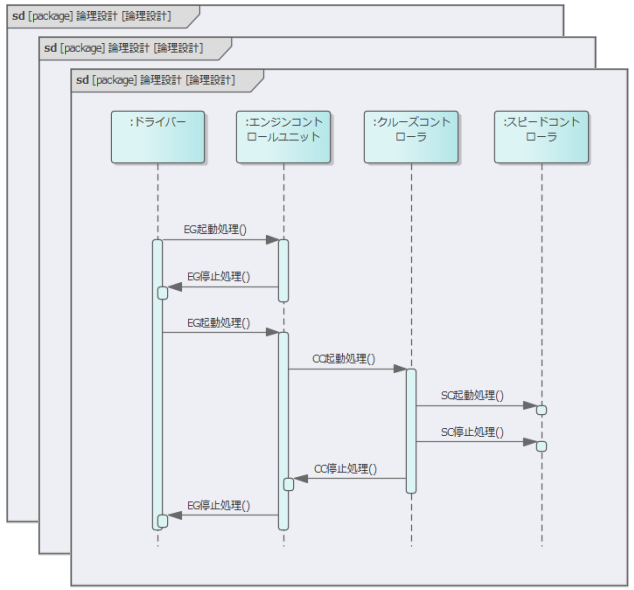
✓ システムが取りうる振る舞いを網羅的に再現させるため、外部環境モデルを新たに作成する。（**検証シナリオに相当するもの**）

✓ 要求定義段階で導出した**テストケースを、検査式として再定義する**。

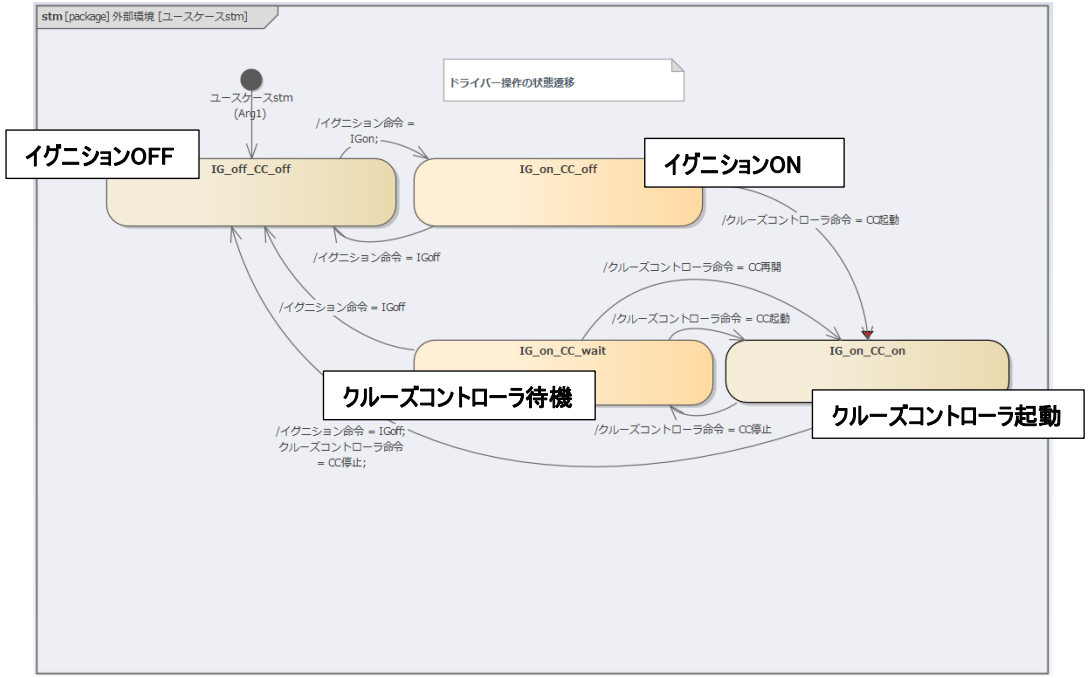
✓ システム全体をどのようなロジック、プロセスで動かすかをメインプロセスとして定義する。

❒検証条件モデル：外部環境モデル

- システムの振る舞い検証を行うため、システムの外部環境（ドライバー）の振る舞いモデルを表現する。
  - 一般的な検証手法では複数の振る舞いパターンを定義する
    - ユースケース図、シーケンス図による検証
    - シナリオベースのシミュレーションによる動的な検証
  - モデル検査ではシナリオ定義の代わりにドライバーの振る舞いをステートマシンで表現する。
    - 遷移条件を非決定とすることで、モデル検査時にあらゆる遷移やタイミングを網羅的に再現できる。



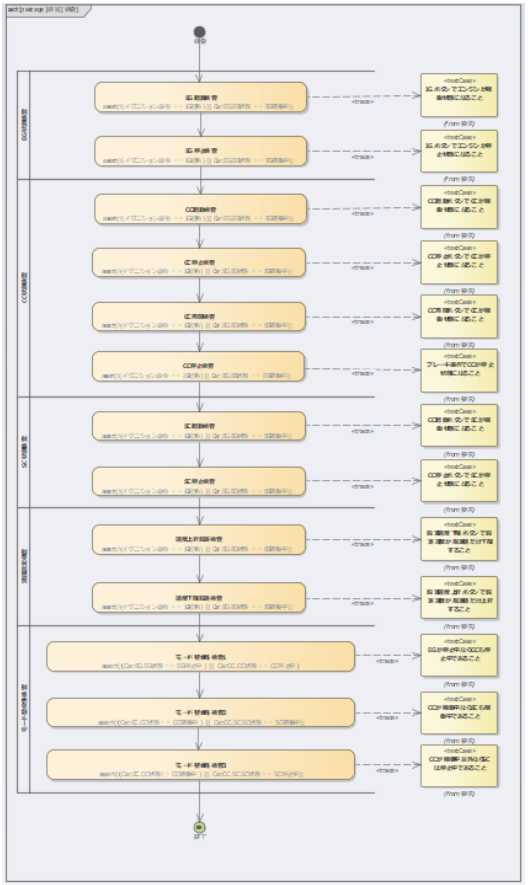
複数のドライバー操作シナリオ

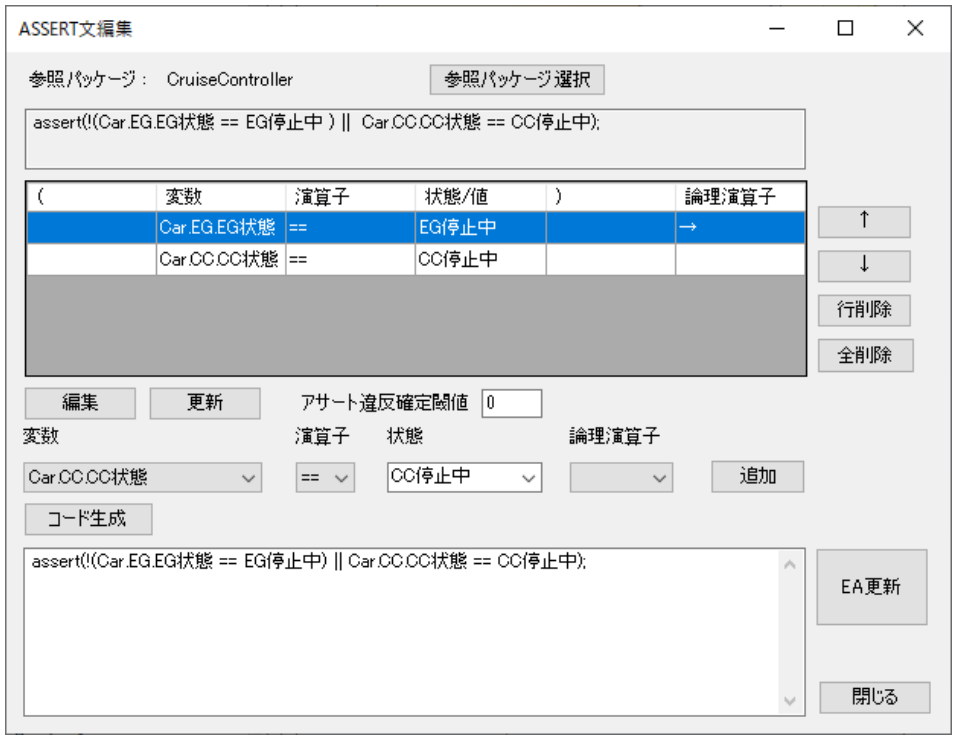


1つのドライバーの操作モデル

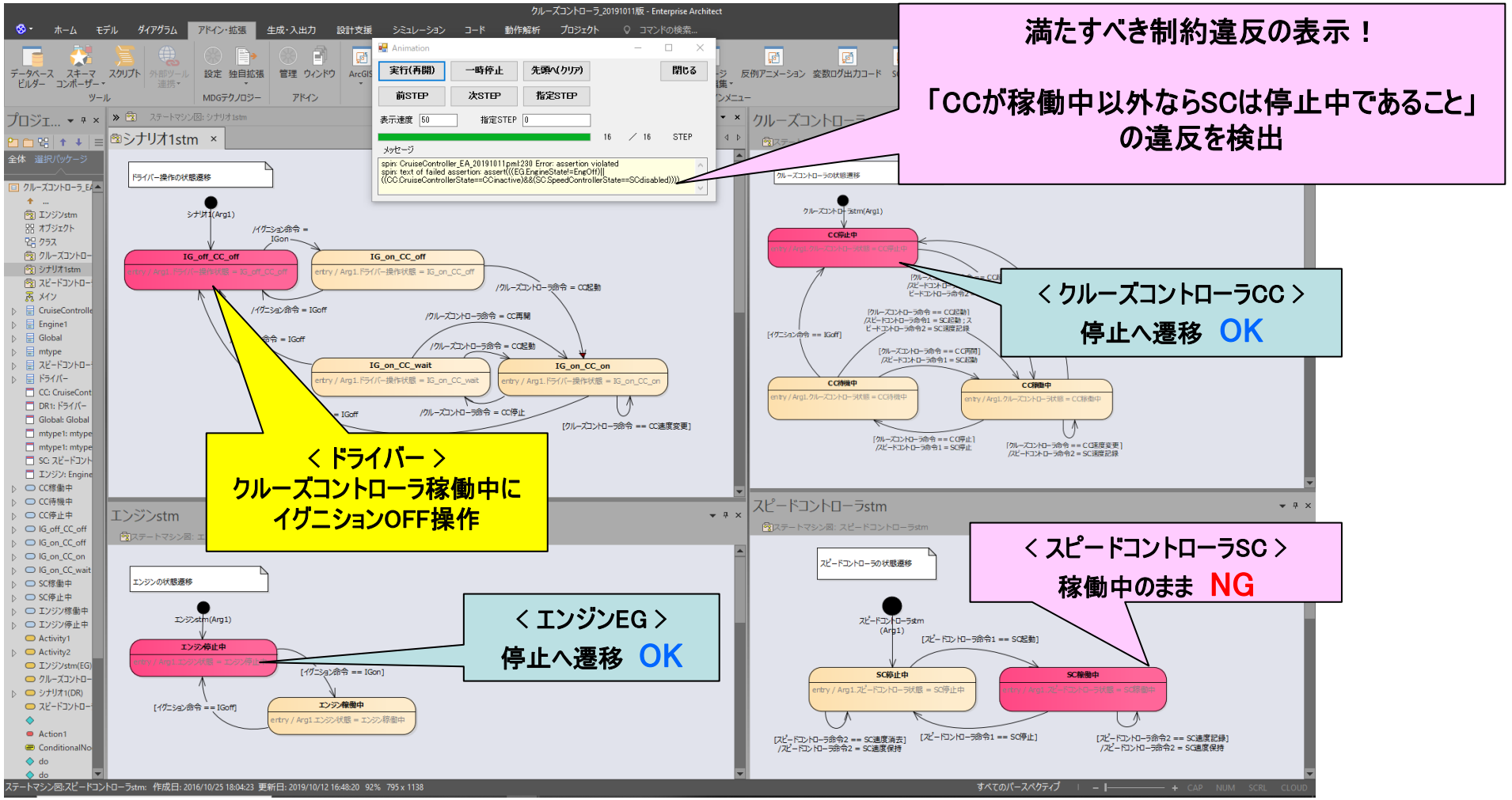
❑検証条件モデル：制約違反の検査式

- 要求・テストケースの条件を満たすかの検査式をASSERTで表現する。
  - 例）EGが停止中ならCCは停止中であること　⇒　Car.EG.EG状態 == EG停止中 → Car.CC.CC状態 == CC停止中
  - 例）CCが稼働中以外ならSCは停止中であること　⇒　Car.CC.CC状態 != CC稼働中 → Car.SC.SC状態 == SC停止中
- 全てのテストケースに対して検査式が定義されているかのトレーサビリティを取る。





モデル検査結果（アニメーション画面）



❒ まとめ

課題

大規模・複雑・分業（ドメイン）により全体のすり合わせ、品質保証が困難

- システム全体を俯瞰したエンジニアリングプロセス（MBSE）が必要
- ただしMBSEプロセスを導入しても、正しく設計できたかのチェックが難しい

KKE提案

属人的検証ではなく、システマティックかつ厳密な検証が必要

- シミュレーションは有効だが限界もあり、モデル検査との併用が有効
- システム設計者がSysMLによる設計と検証を同時並行に行える環境の実現

成果

フロントローディングによる設計品質向上の実現

- モデルを動かして振る舞いを可視化し、技術者間コミュニケーションを促進
- 発見困難な不具合を確実に検出し、上流設計でのヌケモレ、手戻りを低減

ご清聴誠にありがとうございました。

DynaSpec

https://www.kkeformalmethod.com/

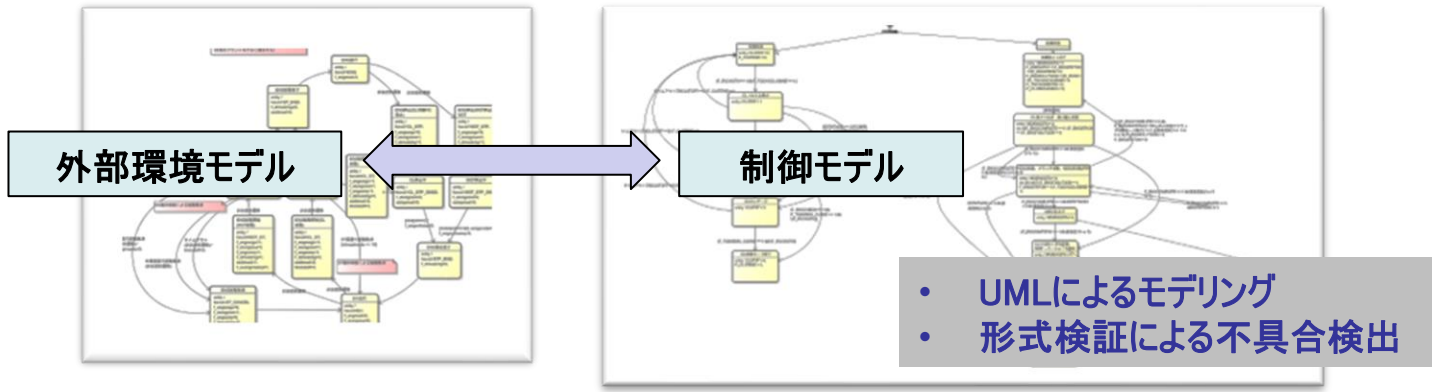
お問い合わせ
株式会社構造計画研究所
製造企画マーケティング部　太田
TEL : 03-5342-1046
Email : aor-bizdev@kke.co.jp

【参考】

□モデル検査手法を用いた検証事例の紹介

❒形式検証の適用対象

- 自動車分野
  - HV の走行モード切替制御（エンジン駆動⇔モーター駆動）の検証
  - EV のシステム起動／停止制御の検証
- 鉄道分野
  - 踏切の警報／遮断制御の検証
- 航空宇宙分野
  - 自律飛行安全システム（飛翔体の位置推定ロジック）の検証
  - 自律飛行安全システム（飛行中断判定ロジック）の検証



❑～ 自動車OEM事例 ～　形式検証導入の背景

■顧客：自動車OEM

■背景／課題：

- コンセプト通りに制御仕様が書けているかのチェックが困難
- 特に複数の制御の組み合わせで機能を実現している制御

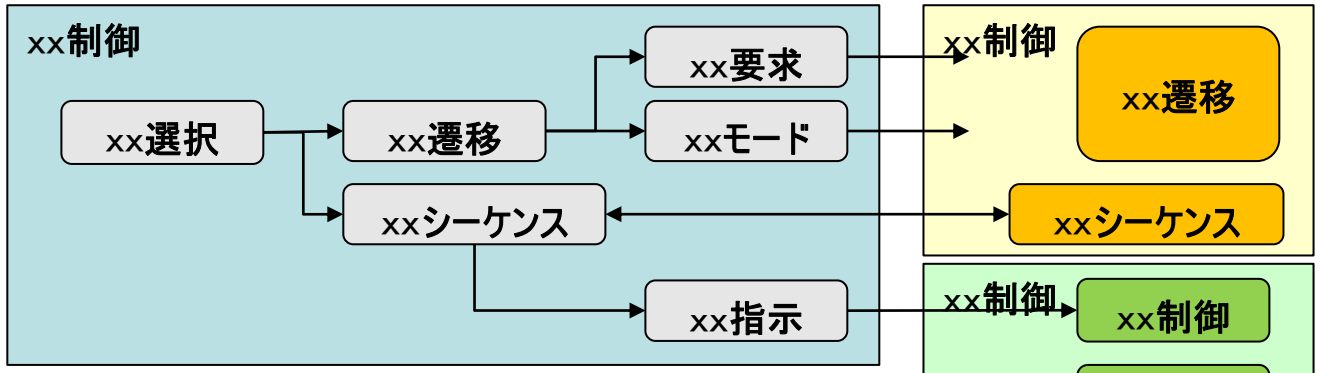
① **HVの走行モード切替制御（エンジン駆動走行⇔モーター駆動走行）**

- 駆動力マネジメント制御、エンジン制御、モーター制御・・・等

② **ＥＶのシステム起動／停止制御**

- エネルギーマネジメント制御、バッテリー制御、ＤＣ／ＤＣ制御・・・等

コンセプト:
制御構造

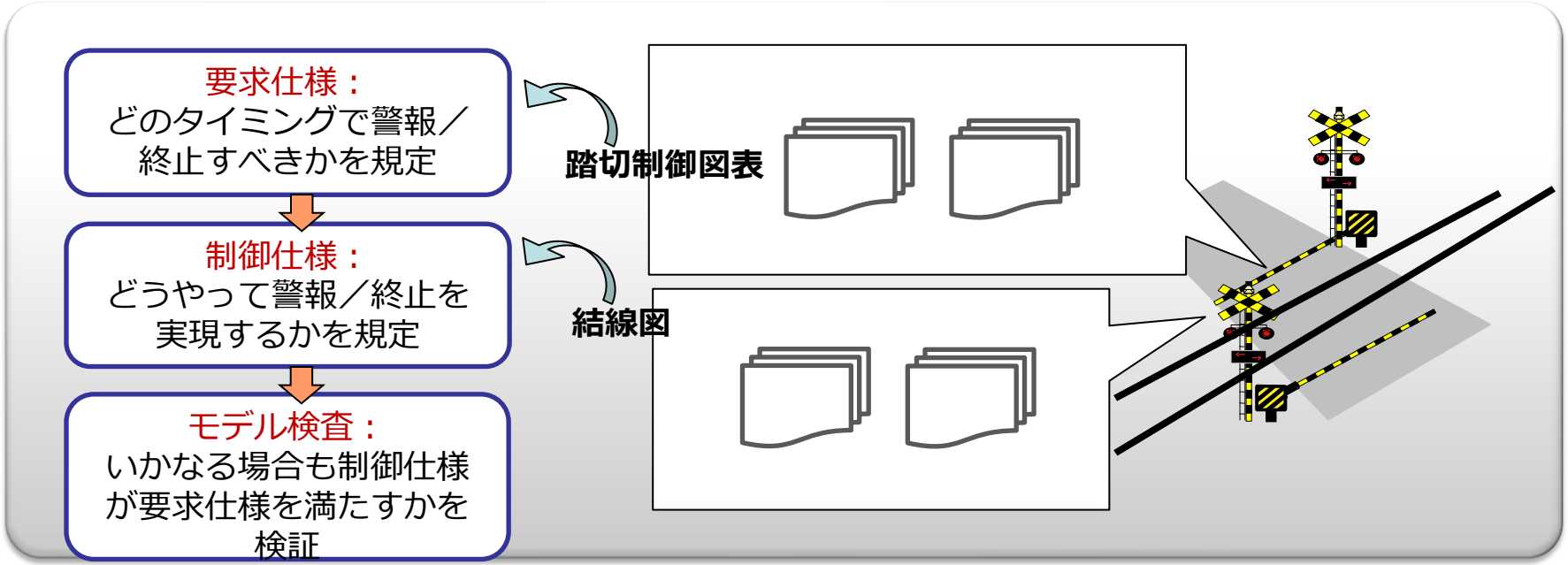


モデル検査による制御設計上の不具合検出を実現

❐～ 鉄道事業者事例 ～　大規模駅構内踏切の警報／遮断制御

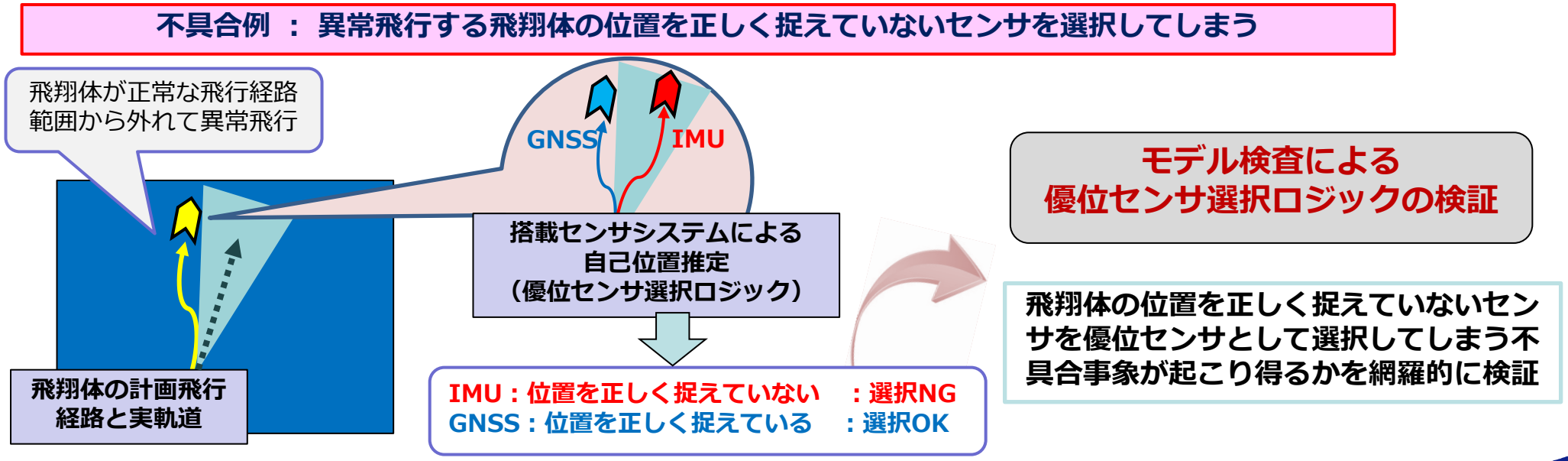
- 顧客：東日本旅客鉄道株式会社
- 背景と目的
  - 踏切の警報等の制御論理は「踏切制御図表」「結線図」によって表現される。
  - 人間系では輸送障害時に発生するような極めてまれな運行状況までを想定した安全性の検証は困難である。

「モデル検査」という論理の変化を網羅的に検査することができる技術を用いて、踏切制御論理の安全性を向上させる。



❑～ 航空宇宙事例 ～　冗長センサシステムによる飛翔体の位置推定

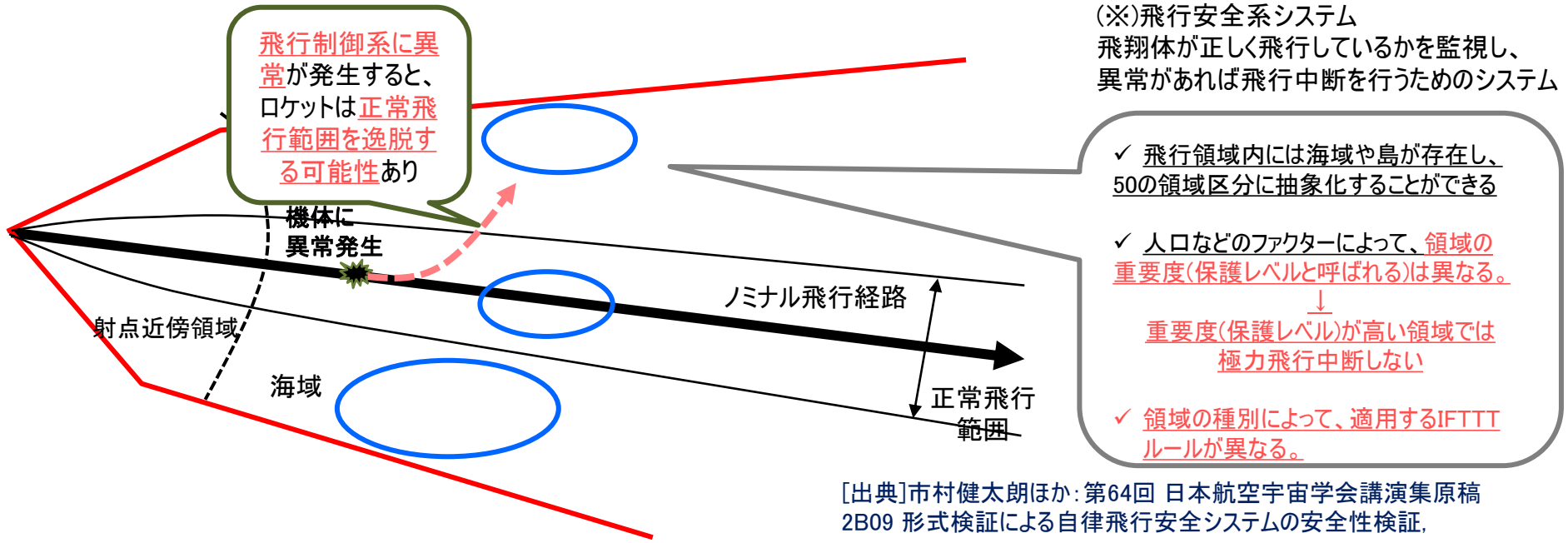
- 顧客：（国研）宇宙航空研究開発機構（JAXA）様、宇宙技術開発（株）（SED）様
  - 経済産業省からの委託事業**「宇宙産業技術情報基盤整備研究開発事業（民生部品等を活用した宇宙機器の軌道上等実証）」**の成果
- 形式検証の適用対象：自律飛行安全システム
  - 自律飛行する飛翔体の位置推定を、 複数のセンサ（GNSS：衛星測位/IMU：慣性航法）を利用した冗長センサシステムにより実現する。
    - ➡ **「優位センサ選択ロジック」**により複数のセンサの中から正しい位置情報を示す優位センサを選択する。
  - 故障・データ異常が起きても安全側に制御する。
    - ➡ 避けたい事象（検証目的）：**飛翔体の位置を正しく捉えているセンサを選択できていない。**



❑～ 航空宇宙事例 ～　自律飛行安全システムの検証

[検証概要]

- 自律飛行安全システムには、ロケットシステムから提供される各種情報に基づき地上の管制者が判断する**10の飛行中断クライテリア(IFTTTルール)**がソフトウェアとして実装されている。
- この自律飛行安全システムに対し、本件では以下の検証を行い、**不具合が存在しないことを明らかにした。**
  - **すべてのシステムの中で2故障の範疇で、意図しない飛行中断、飛行続行が起こらないこと**
  - **飛行安全系システム内(※)で1故障が発生しても、ミッション(ロケットを周回軌道にのせること)が達成できること**
  - **すべてのシステムの中で2故障が発生しても安全性が担保されること**



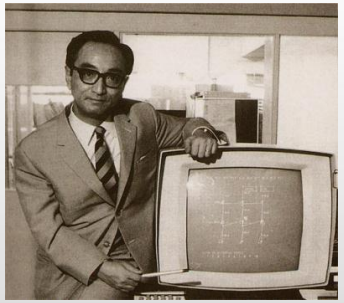
(※)飛行安全系システム
飛翔体が正しく飛行しているかを監視し、異常があれば飛行中断を行うためのシステム

- ✓ 飛行領域内には海域や島が存在し、50の領域区分に抽象化することができる
- ✓ 人口などのファクターによって、領域の重要度(保護レベルと呼ばれる)は異なる。
  ↓
  重要度(保護レベル)が高い領域では極力飛行中断しない
- ✓ 領域の種別によって、適用するIFTTTルールが異なる。

[出典]市村健太朗ほか:第64回 日本航空宇宙学会講演集原稿
2B09 形式検証による自律飛行安全システムの安全性検証,

（株）構造計画研究所のご紹介

構造計画研究所のご紹介　～企業プロフィール



創業者　服部 正　工学博士

■1959年5月6日設立
Arthur D. Littleオフィスを見て
日本に工学を生業とする技術コンサル事務所開設を決意

■1961年
IBM1620を導入（日本に2台）

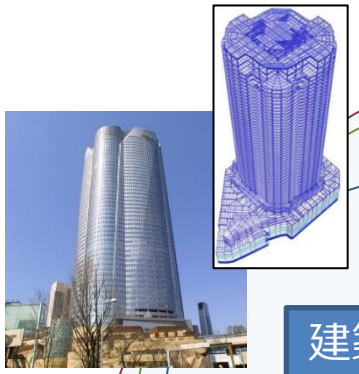
IBM1620

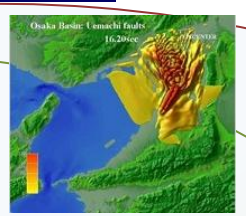
大学・研究機関と実業界とをブリッジする
デザイン＆エンジニアリング企業

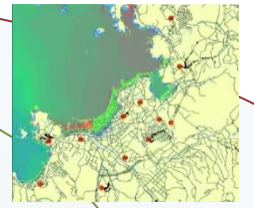
代表執行役会長	服部 正太
代表執行役社長	渡邊 太門
資本金	1,010百万円
従業員	611名(2021.9.8現在)
上場市場	JASDAQ（2000年3月株式公開）

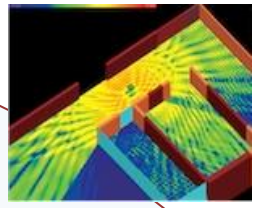
Professional Design＆Engineering Firm

構造計画研究所のご紹介 ～事業の拡大









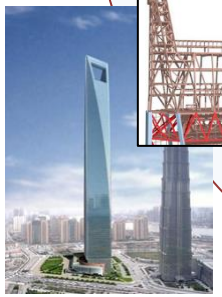
構築物

建築 土木

維持 管理

住宅 住設

-構造設計技術
-耐震・免震・制震技術
-構造解析技術
-画像認識技術
-データ分析技術
-工事管理技術



自然・環境

防災 支援

災害 リスク 評価

環境 評価

-数値シミュレーション技術
-地震動評価技術
-画像認識技術
-データ分析技術
-流体解析技術
-リスク評価技術

社会・企業 コミュニティ

情報 通信

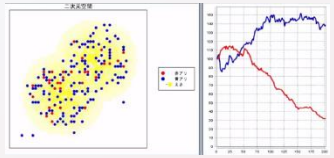
製造 支援

交通 物流

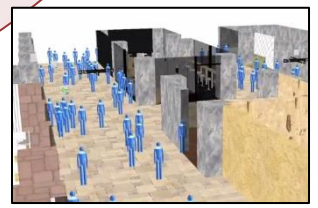
エネル ギー

金融

サー ビス



-システム構築技術
-マルチメディア技術
-数理最適化技術
-意思決定構造分析技術
-ネットワーク構築技術
-ビッグデータ分析技術



構造計画研究所のご紹介 ～Worldwide Collaboration

各分野のすぐれた商品や知識、研究結果を世界の企業・大学研究機関に求め、つねにベストなソリューションを提供できる体制を整えています。