IPA Software Engineering Center
Information-technology Promotion Agency, Japan

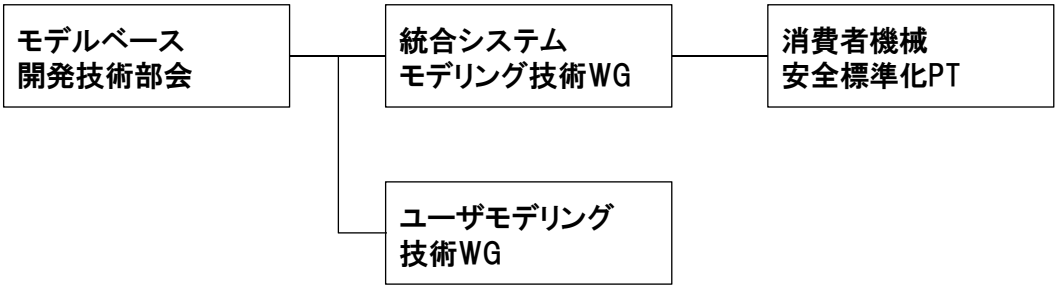
モデルベース開発とモデル検証

独立行政法人情報処理推進機構
技術本部ソフトウェア・エンジニアリング・センター
統合系プロジェクト　研究員　内田功志
2012年11月20日

平成23年度モデルベース開発技術部会

SEC
Software Engineering for Mo・No・Zu・Ku・Ri

- 統合システムモデリング技術WG
  - 障害波及性、システム安定性等今後増大する統合システムにおける主要リスク要因を設計段階で低減するために必要な技術を整理する。
- ユーザモデリング技術WG
  - ユーザ特性をモデル化するための標準的なプロセスを定義するとともに、当該プロセスの有効性確認のためのケーススタディを実施する。



統合システムモデリング技術WG

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ 代表的なモデリング手法を検討
- MDD(Model Driven Development):情報システムで多く使われている
- MBD(Model Based Development):組込みシステムで多く使われている

■ MDDとMBDの統合
- 情報システムと組込みシステムを統合した場合に、「障害の影響が予測できないことがある」という課題に一部対応

⇩

!?「統合システムの上流設計段階で統合システム全体としての信頼性評価を実施」までは難しい

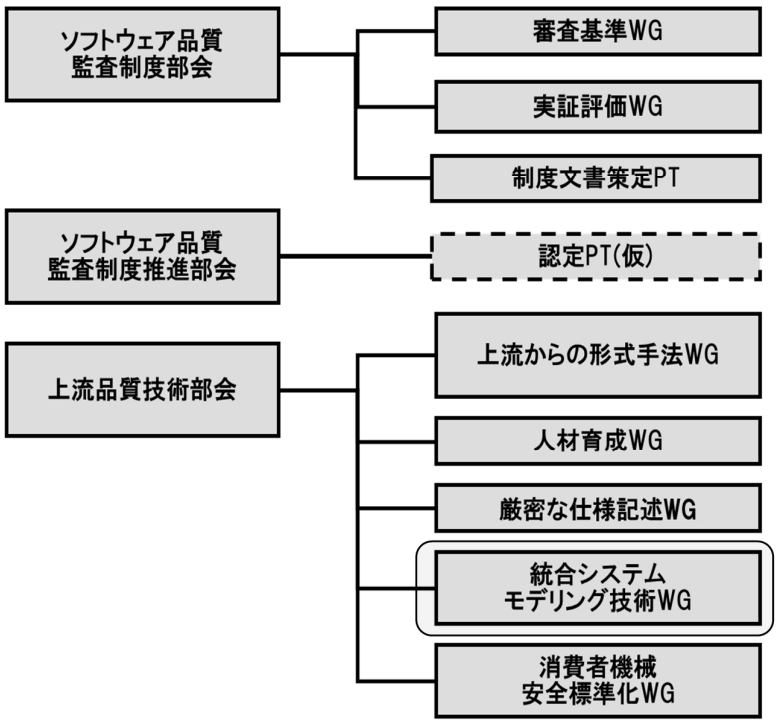
⇩

☆システムズエンジニアリングを検討

平成24年度部会構成

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

H24年度　統合系プロジェクトの部会活動案　12.5.7

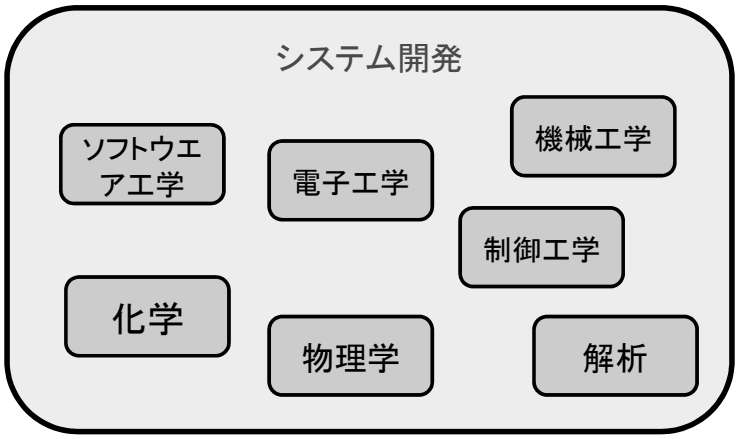


システムズエンジニアリングとは

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■"Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems"

●システムズエンジニアリングとは、システムの実現を成功させることができる複数の専門分野にまたがるアプローチおよび手段
by INCOSE(The International Council on Systems Engineering)



現在のシステム開発の特徴(1)

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

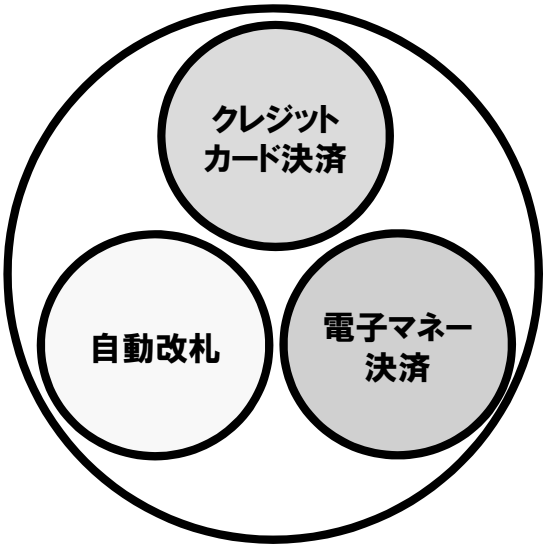
■ 新たなエンジニアリング領域への対応
- 機能安全(ディペンダビリティ)
  - システムの信頼性・安全性に関する領域
  - IEC61508、ISO26262
    - ISOは正式名称を国際標準化機構(International Organization for Standardization)といい、各国の代表的標準化機関から成る国際標準化機関で、電気及び電子技術分野を除く全産業分野(鉱工業、農業、医薬品等)に関する国際規格の作成を行っている
    - 正式名称を国際電気標準会議(International Electrotechnical Commission)といい、各国の代表的標準化機関から成る国際標準化機関であり、電気及び電子技術分野の国際規格の作成を行っている
- 環境工学
  - システムと環境との関係を扱う領域
  - 製造・運用・廃棄に伴なう環境負荷の考慮

■ QCDSE
- QCD＋Safety＋Environment

現在のシステム開発の特徴(2)

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

■ 大規模複雑化の加速
- 多機能化、機能間連携の複雑化
- 統合システム(System-of-Systems)



System-of-Systems
単独でも成立するシステムが組み合わさってさらに高度なサービスを提供するもの

システム開発の課題

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

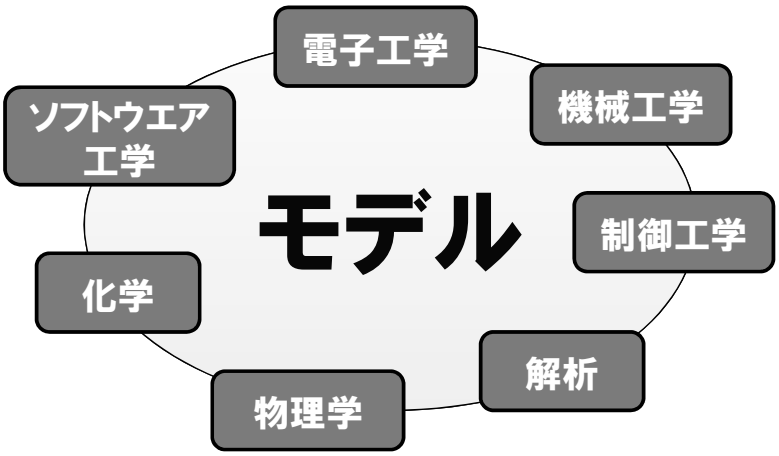
■ エンジニアリング領域の拡大と大規模複雑化
→ 開発関係者の多様化
→ ドキュメントの増加

■ 効率的な記述言語とコミュニケーション方法が必要
→ モデルの導入へ
MBSE(Model Based Systems Engineering)
によるシステム開発が効果的？
→ 日本で使えるように標準化する必要がある！
→ まずは、MBSEを導入のためのマップを作る

MBSEとは

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

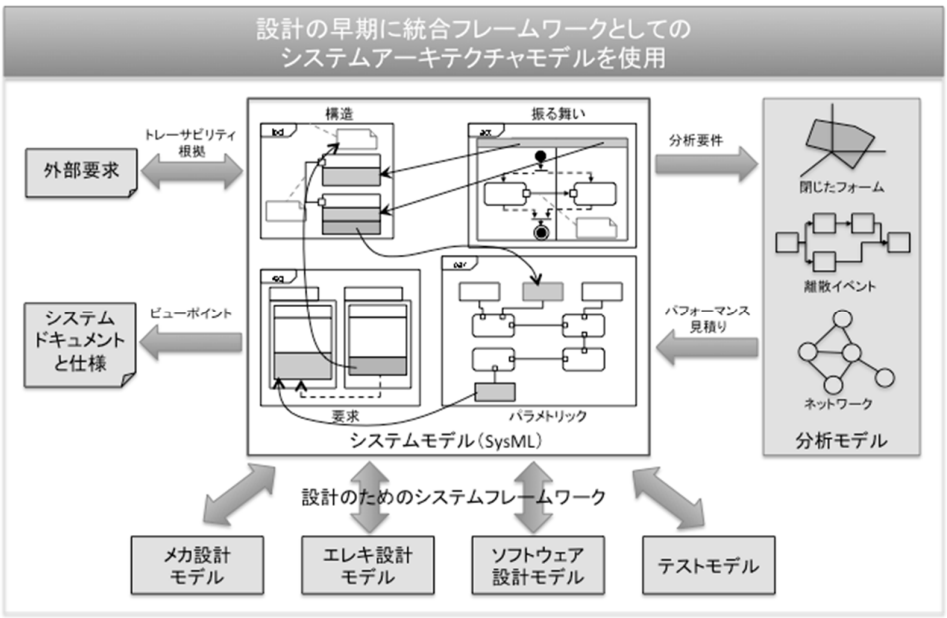
■ モデルを活用したシステムズエンジニアリング
- 様々な分野をモデルで橋渡し
- モデルによる効率的な記述
  - ロ 代表的なモデリング言語がSysML



MBSEの基本構造

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

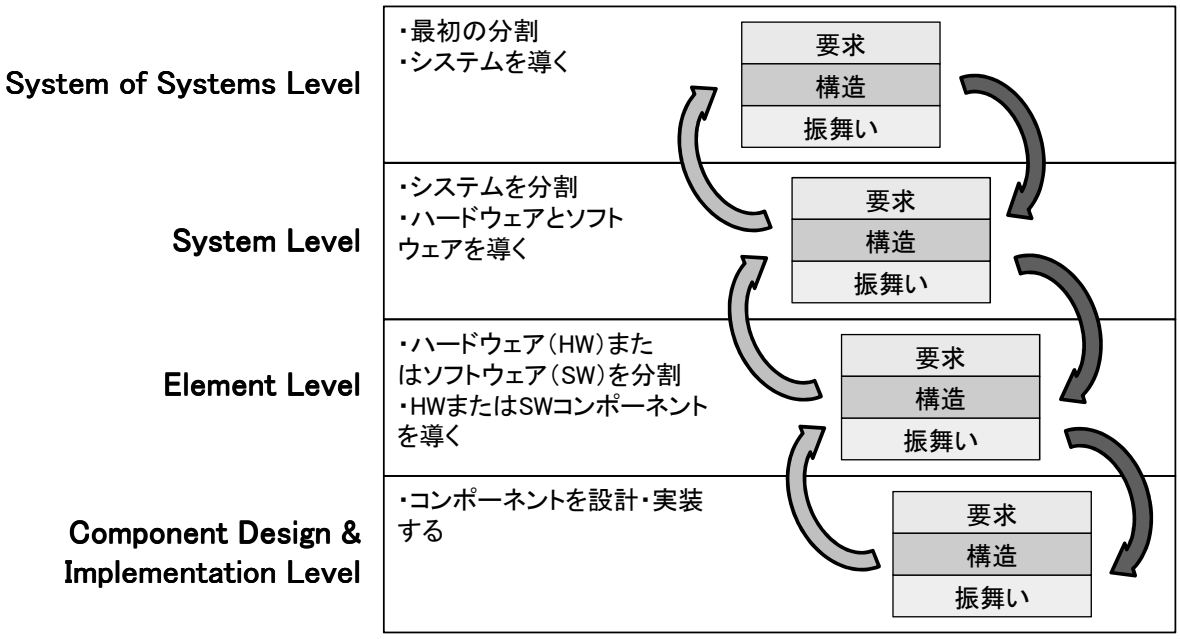
■ 対象のモデルを適切に構成要素に分割(decompose)できるため、QCDSE(品質、コスト、納期、安全性、環境)などのバランスをとることができる



MBSEのスコープ

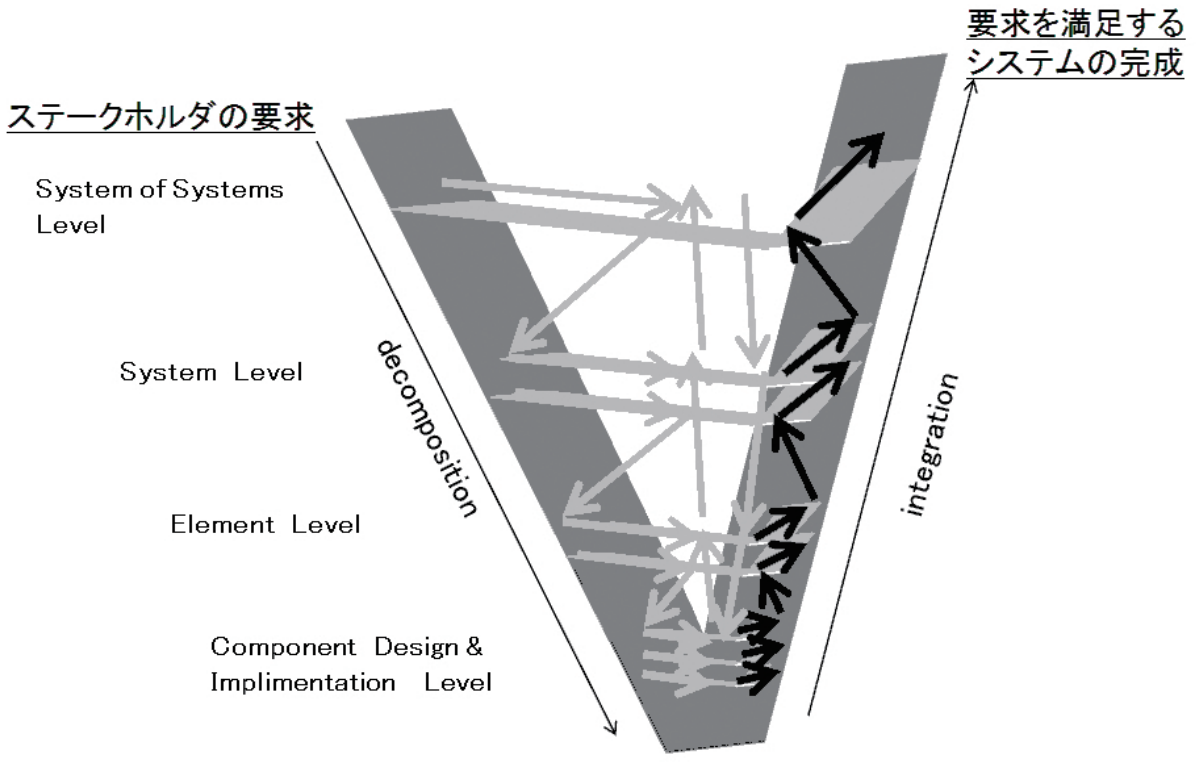
SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ 階層的にシステムを扱うことができる



Dual Vee Modelの流れ

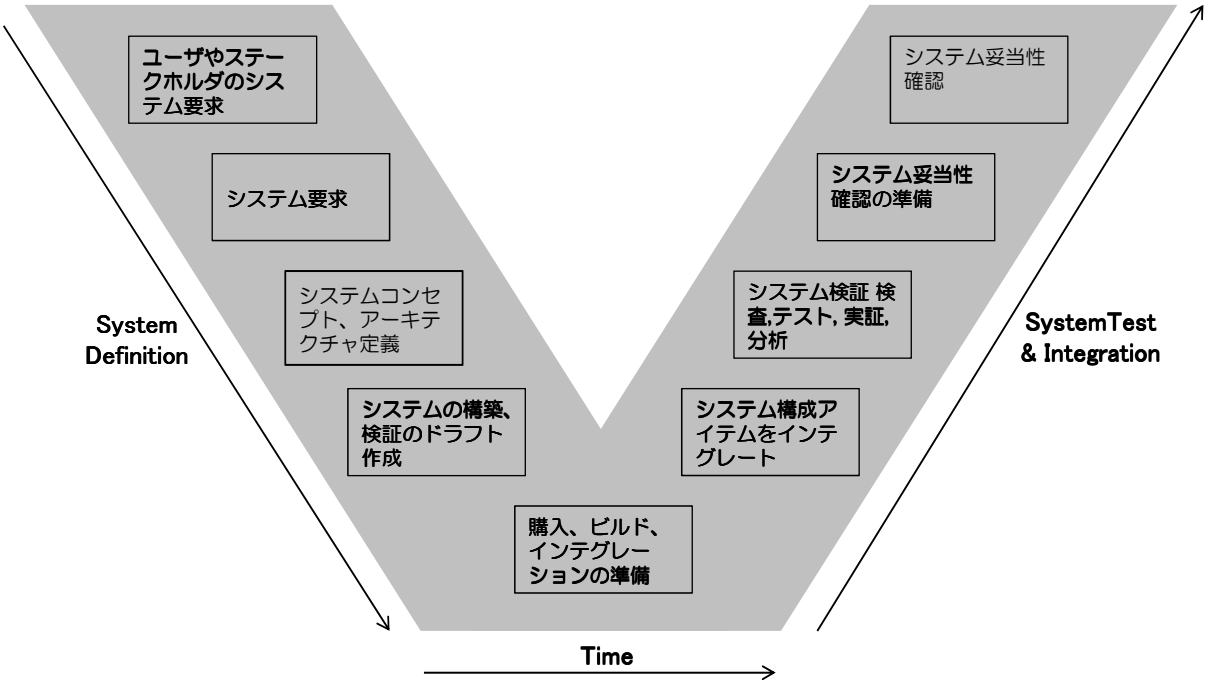
SEC
Software Engineering for Mo·No·Zu·Ku·Ri



Entity Vee (System Level)

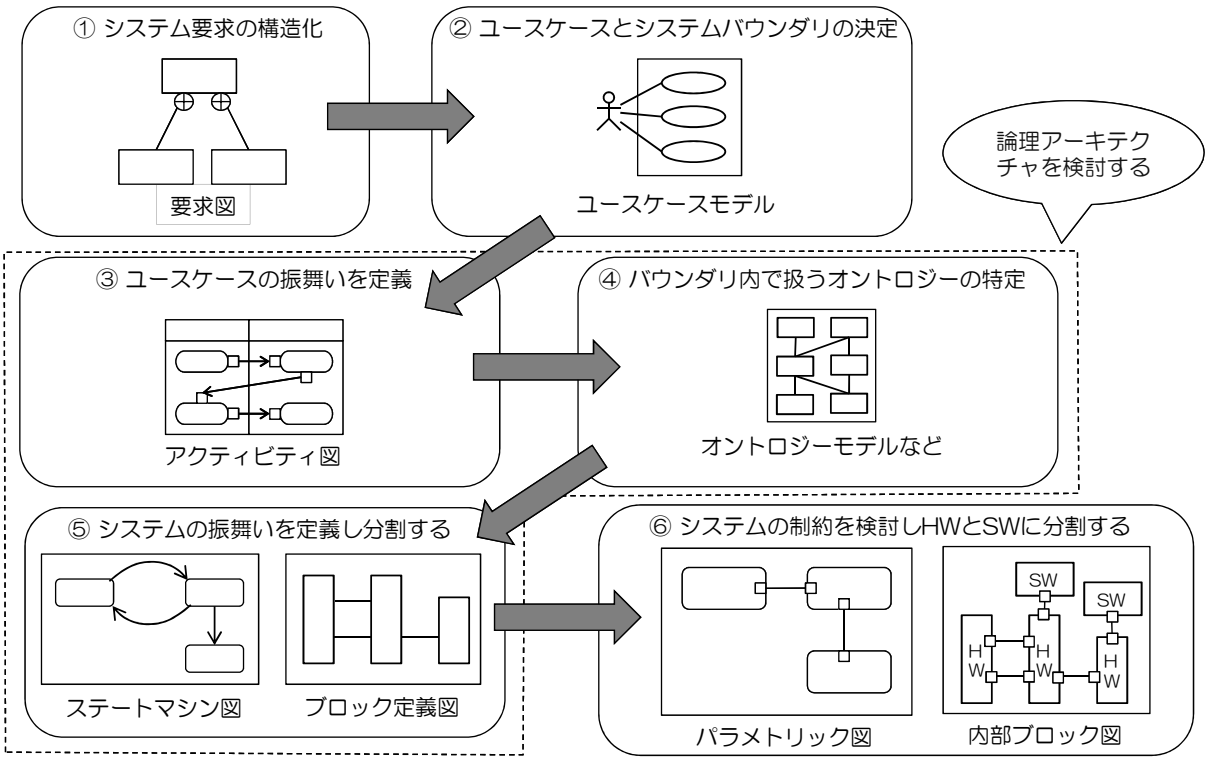
SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ System LevelのEntity Vee



MBSEの基本的な流れ

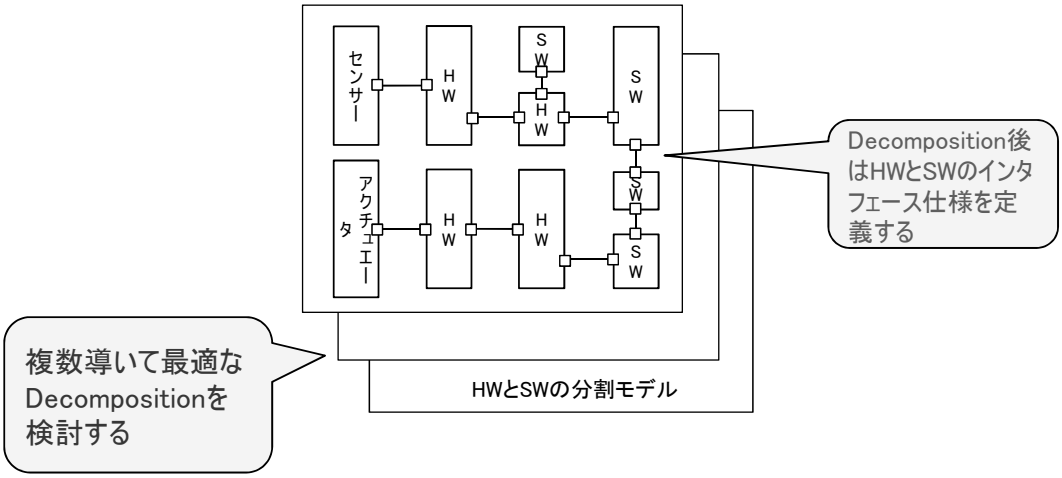
SEC
Software Engineering for Mo·No·Zu·Ku·Ri



MBSEのトレードスタディ

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

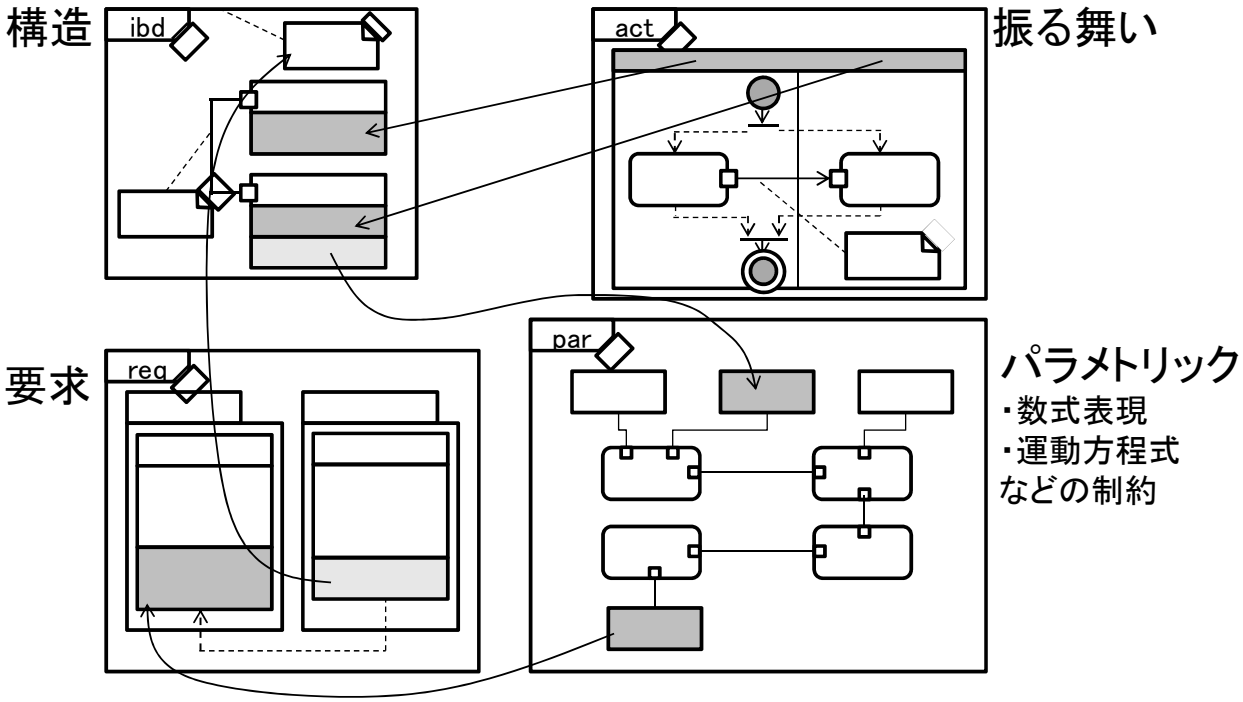
■ トレードスタディにより、ハードウェアとソフトウェアの最適なDecompositionを行う



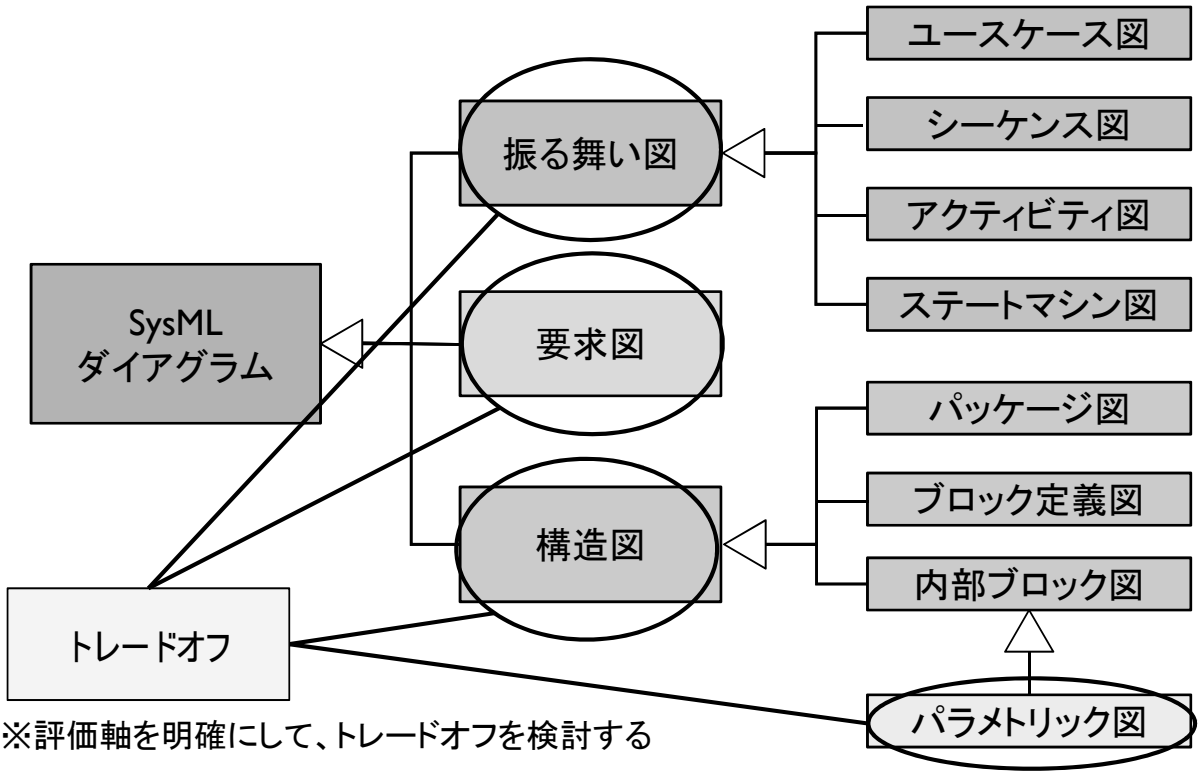
System LevelのSysML4本柱

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

SysMLのダイアグラムは，互いに関連しているので，設計変更があった場合にもその影響を容易に把握できる



SysMLとトレードオフ



※評価軸を明確にして、トレードオフを検討する

SysMLを使用したMBSE

- MBSEが複雑なシステムに有効であることは、米国防システムや航空宇宙関連システムで実証済みである
- 日本においても徐々にではあるが、実績が生まれつつあるが、完全に後手に回ってしまっている
  - このままではアジア圏の中でも遅れを取る可能性がある
- MBSEの導入を加速するためには、導入を推進するための仕掛けが必要である

それこそがMBSE導入マップである

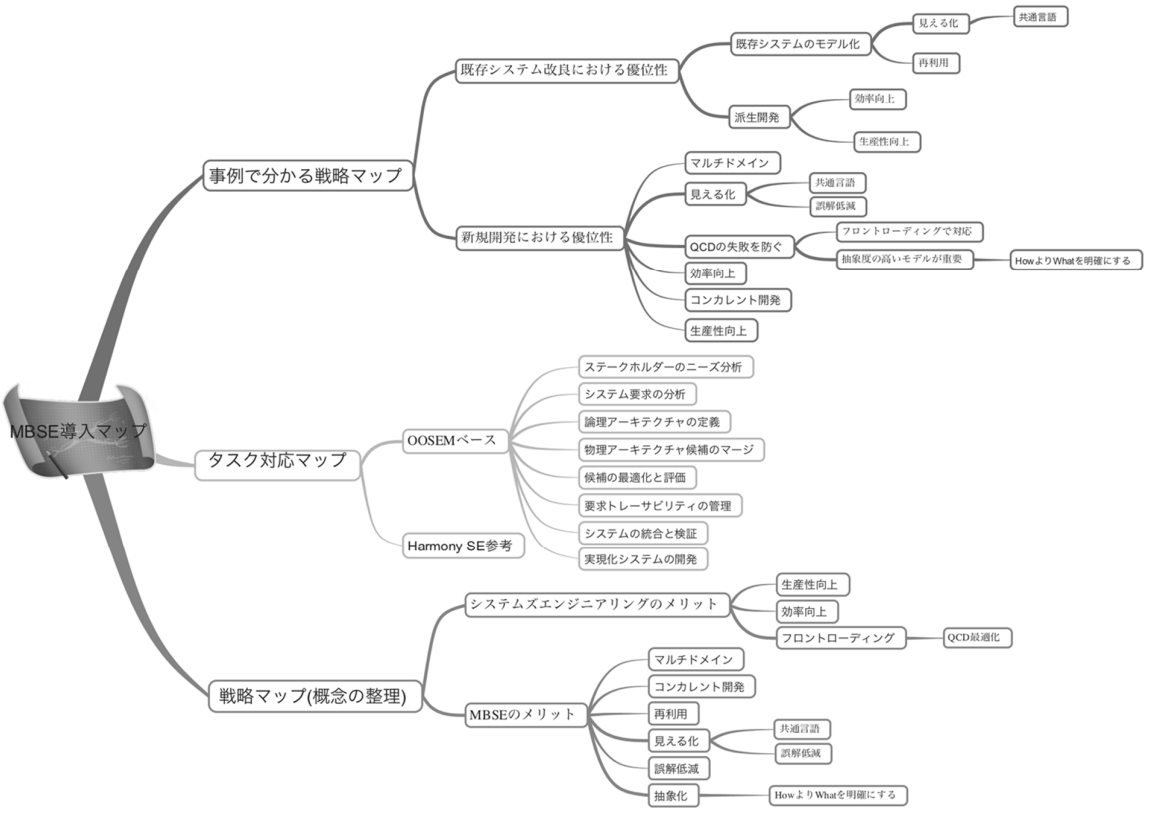
MBSE導入マップとは

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

- MBSEを導入し易くするための仕掛け
- 大きく2つに分けることができる
  - システムエンジニアリング実施における「モデリングと基本タスクの対応モデル」
    - 拡張Veeプロセス中での「各SysMLモデル」を使ったシステムズエンジニアリング「タスク」の対応マップ
  - システムズエンジニアリングを「日本の産業界に導入するための戦略マップ」
    - システムズエンジニアリングタスクとSysMLモデル図の目的・用途に応じた導入優位マップ

MBSE導入マップ

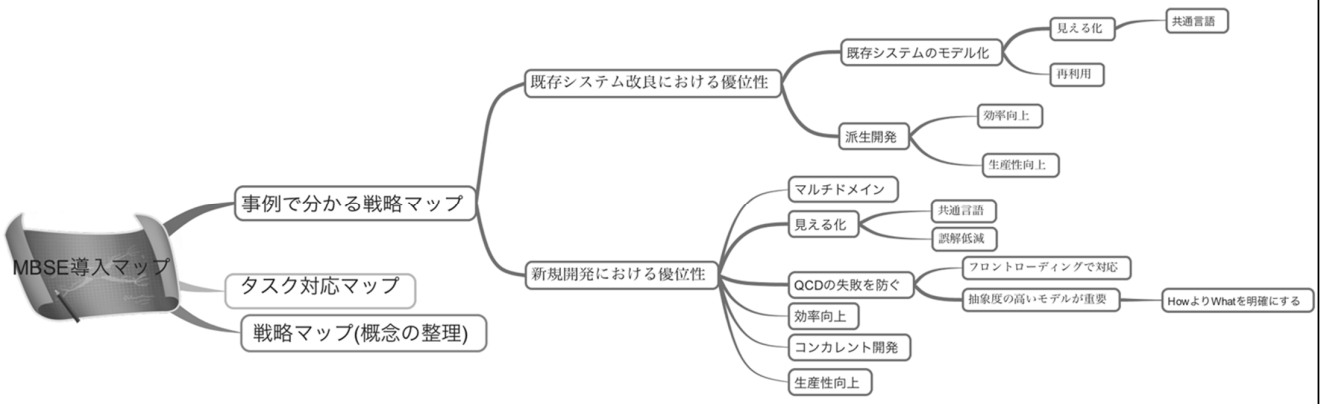
SEC
Software Engineering
for Mo·No·Zu·Ku·Ri



導入マップ：事例

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

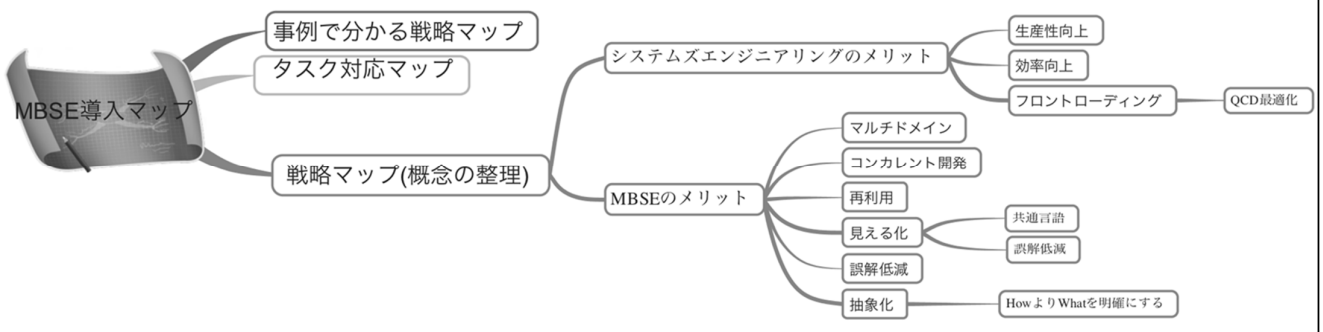
- 既存システムを改良する場合と新規開発でMBSEの優位性を事例を使って説明する
- 既存システムの改良においても、新規開発においても、MBSEが十分優位性があり、導入するメリットが大きいことを示す
- この事例で分かる戦略マップは、MBSEの導入を促進する起爆剤である



導入マップ：戦略

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- 概念を整理するための戦略マップは、事例で分かる戦略マップを補足するもので、付録としての役割を持つ
- MBSE以前に確立されたシステムズエンジニアリングのメリットを明確にし、さらにそれをモデルベースに拡張したMBSEのメリットを明確にし、個々のメリットに関する簡単な説明を付与する



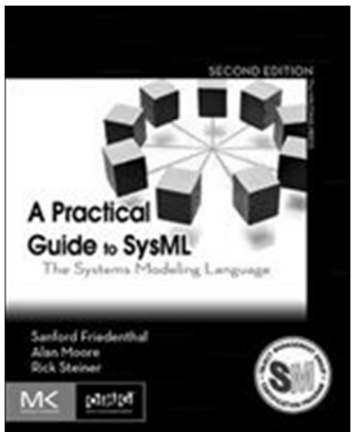
導入マップ：タスク対応

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- タスク対応マップは、INCOSEが提唱しているOOSEM (Object Oriented Systems Engineering Method)をベースにして、IBMの提唱するHarmony SEを参考にして作成していく
- ステークホルダーのニーズ分析～実現化システムの開発という大きな枠組みはOOSEMを流用し、個々のワークフローの中をHarmony SEの要素を参照して必要なものを取り込んでいく

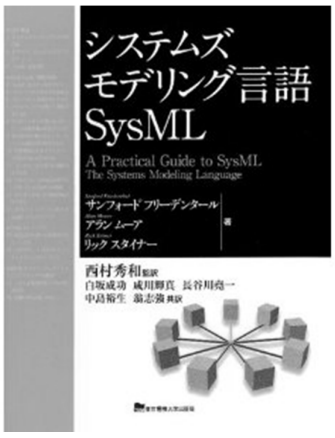
MBSE(OOSEM)の書籍

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



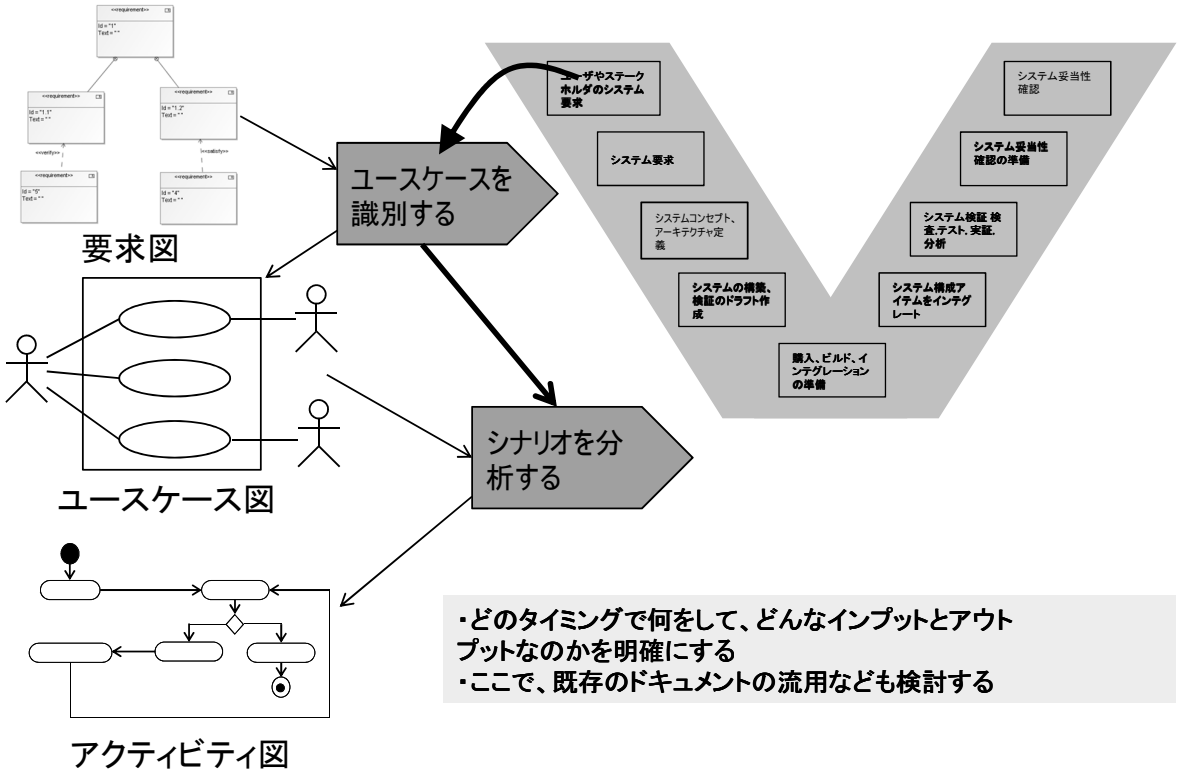
A Practical Guide to SysML, Second Edition: The Systems Modeling Language (The MK/OMG Press)

翻訳



システムズモデリング言語SysML（東京電機大学出版局）

MBSEと基本タスクの対応モデルの例



要求図

ユースケース図

アクティビティ図

・どのタイミングで何をして、どんなインプットとアウトプットなのかを明確にする
・ここで、既存のドキュメントの流用なども検討する

代表的なMBSE手法

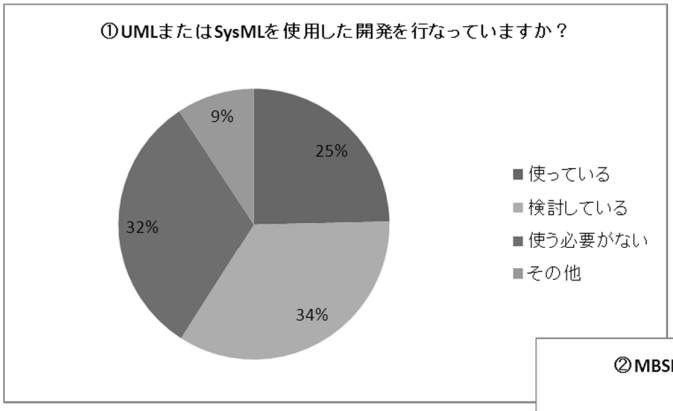
SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

■ 汎用的なもの
- OOSEM
  INCOSE(International Council on Systems Engineering)が提唱するオブジェクト指向によるモデルベース・システムエンジニアリング
- Harmony for SE （MbSE）
  I-Logixの組込み向けソフトウエア開発プロセスのHarmonyをシステムズエンジニアリング向けにしたもの
- RUP for SE
  Rational Unified Processをシステムズエンジニアリング向けにしたもの

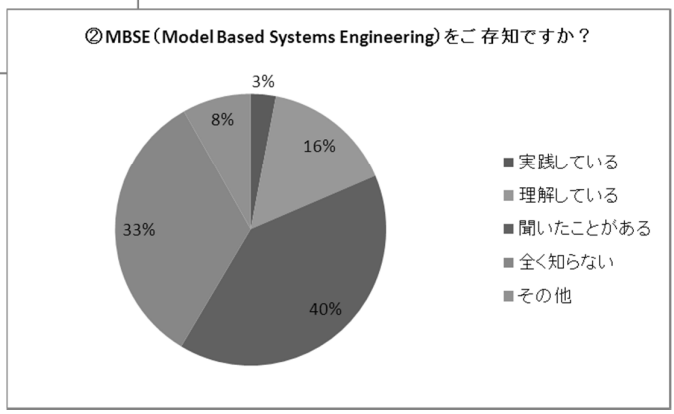
■ ドメインや目的を明確にしたもの
- EASIS　　自動車向け
- ASSERT　　高信頼性システム
- GENESYS　　クロスドメイン

日本の現状

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



ESEC2012のIPAブース
アンケート調査より



2012年度

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- モデルベース・システムズエンジニアリング（MBSE）を検討
- MBSE導入マップの作成
  - 事例による戦略マップを検討
    - 事例：券売機
  - タスク対応マップを検討
    - OOSEM(Object Oriented Systems Engineering Method)をベースにタスク対応マップを検討
  - 戦略マップ一覧を整理

2013年度～（その1）

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- MBSE導入マップの実プロジェクトへの適用検証
  - MBSE導入マップの洗練
- MBSE教育プログラムの実施
  - 教材開発
  - ハンズオンセミナー実施
  - 大学・専門学校への展開
- リスク・ハザード分析の導入
  - リスク・ハザード分析から上位のSafety Goalを導く
  - 上位のSafety GoalをAssurance CaseなどのGSNで分析・管理する
  - MBSEにAssurance Caseを組み入れる

2013年度～（その2）

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- 検証技術の提案
  - SysMLでの上流検証を検討
    - 要求の検証
    - 振舞いの検証
    - 構造の検証
    - パラメトリック検証
- ユーザモデルの提案
  - ユーザモデルによる要求の抽出
  - ユーザモデルによる検証
  - ユーザモデルのMBSEへの適用
- ツール化要件の洗い出し
  - MBSEを実現するためのツールを検討

SVA スマートシステム検証技術協会

一般社団法人スマートシステム検証技術協会

■ 設立目的

スマートシステム全体の信頼性・安全性等の利用者が求める品質を第三者が検証するための、検証手法・検証技術を確立することにより、より安全・安心・快適なスマート社会の実現を目的として設立しました。

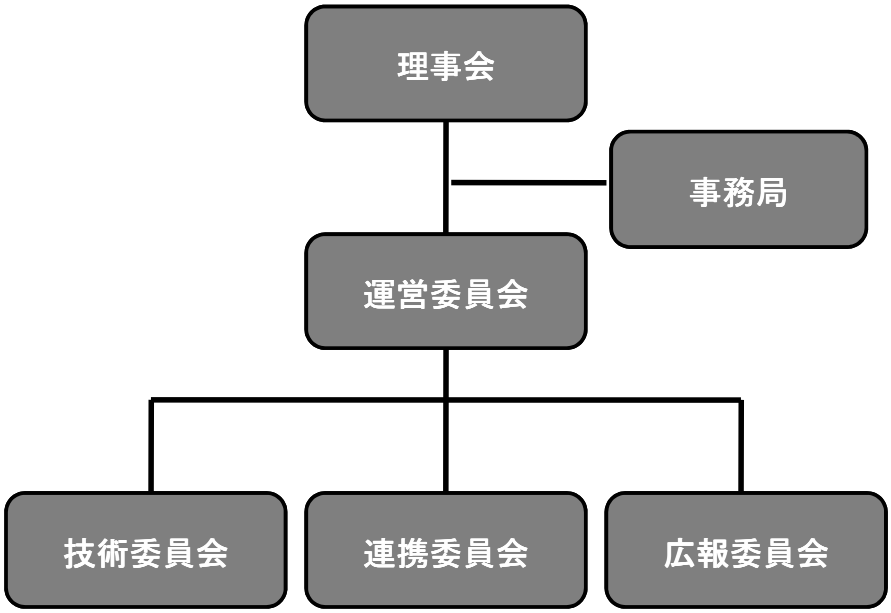
■ 設立：2012年6月1日

■ 実施事業

- スマートシステムの検証手法・検証技術の確立
- スマートシステムの検証手法・検証技術の普及・啓発
- スマートシステムの検証手法・検証技術の確立の応用

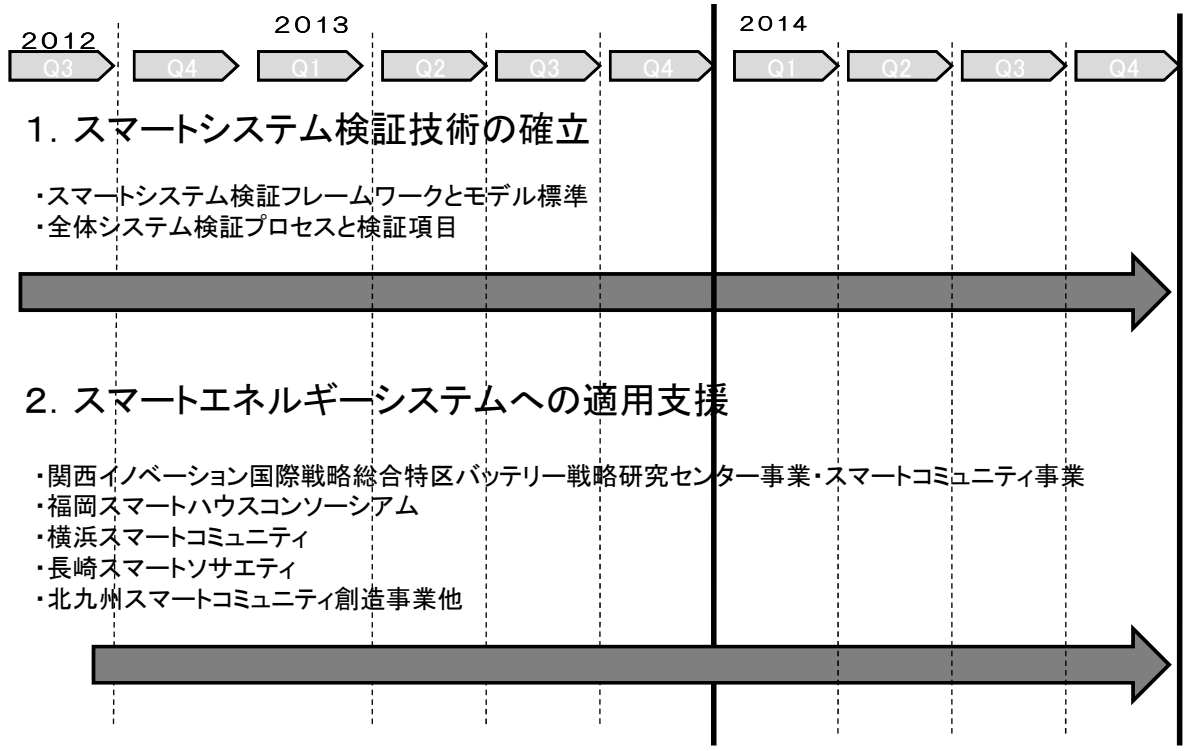
組織図

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



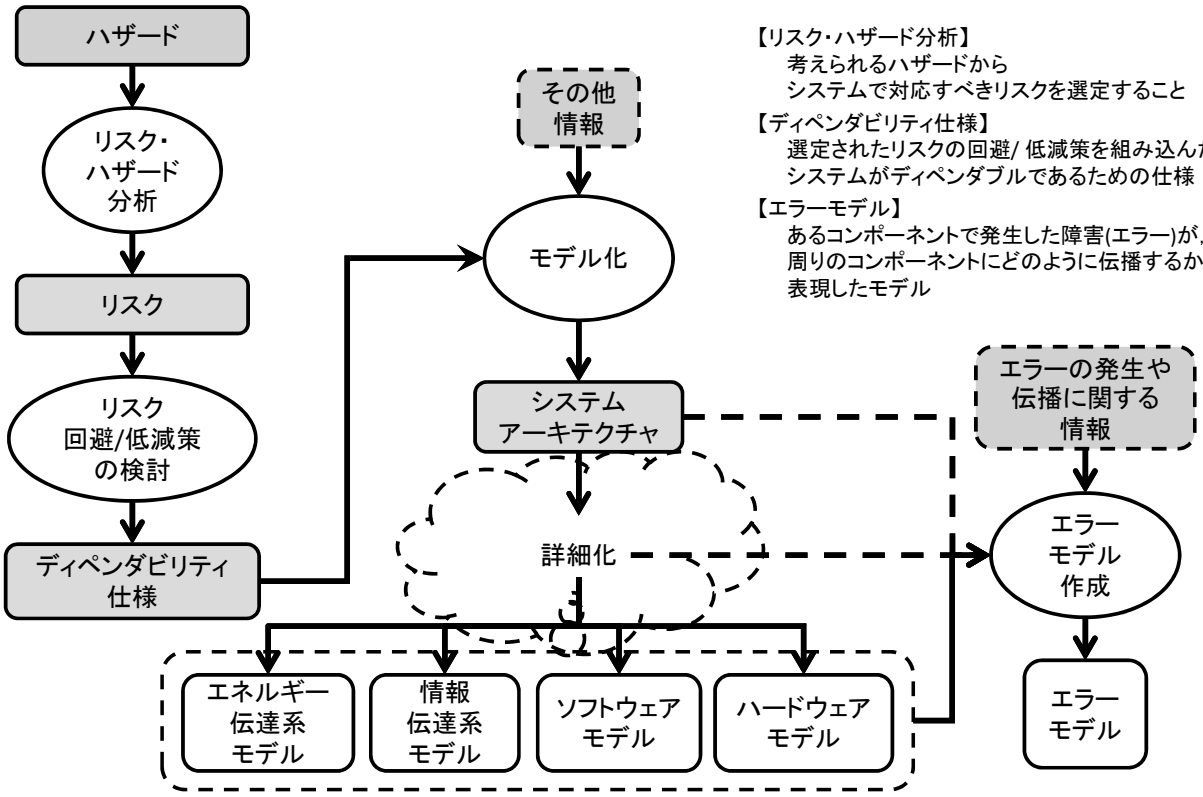
全体スケジュール(中期スパン)

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



リスク・ハザード分析からモデル作成までの流れ

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



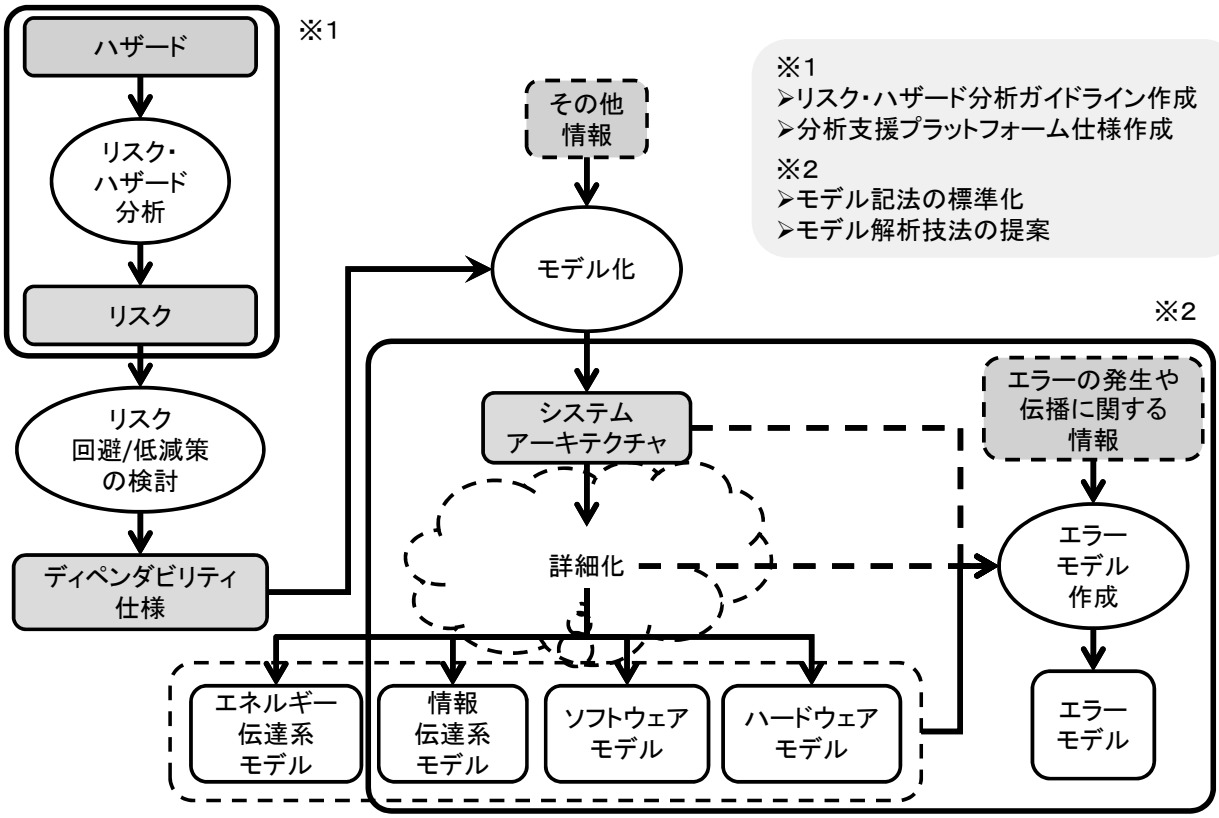
【リスク・ハザード分析】
考えられるハザードから
システムで対応すべきリスクを選定すること

【ディペンダビリティ仕様】
選定されたリスクの回避/ 低減策を組み込んだ,
システムがディペンダブルであるための仕様

【エラーモデル】
あるコンポーネントで発生した障害(エラー)が,
周りのコンポーネントにどのように伝播するかを
表現したモデル

本年度のスコープ

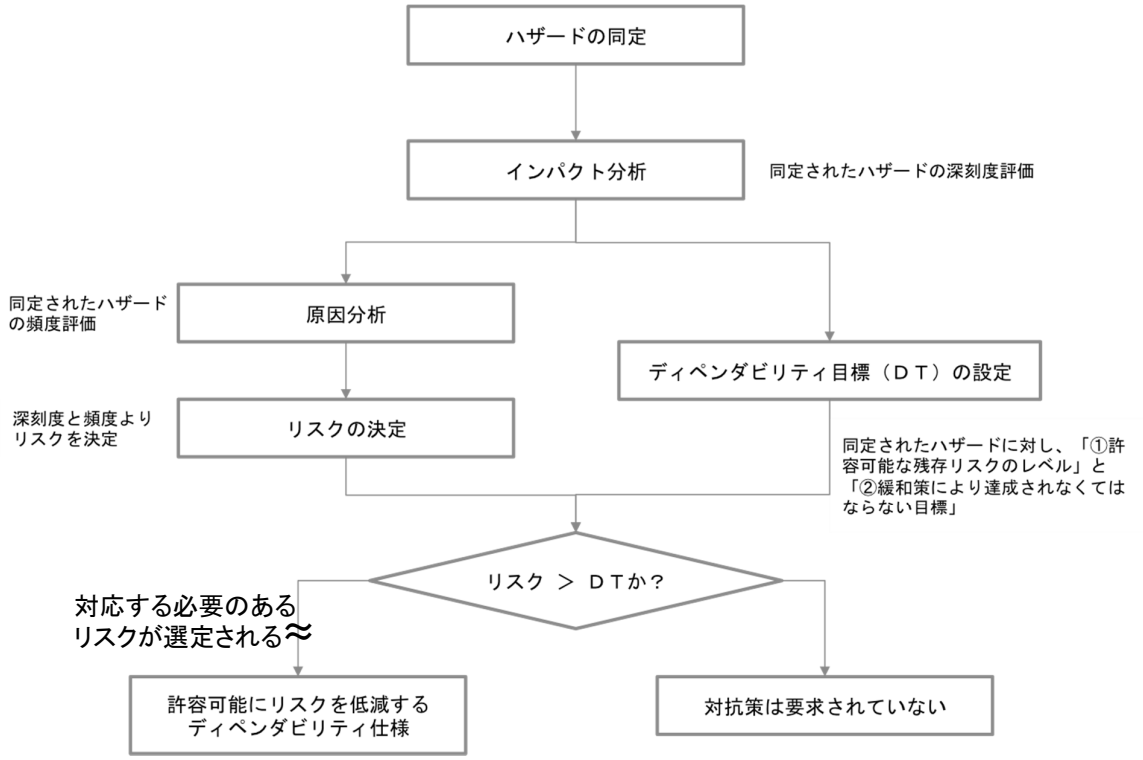
SEC
Software Engineering for Mo·No·Zu·Ku·Ri



※1
- リスク・ハザード分析ガイドライン作成
- 分析支援プラットフォーム仕様作成

※2
- モデル記法の標準化
- モデル解析技法の提案

リスク・ハザード分析プロセスの例



リスク・ハザード分析ガイドラインの目標

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

- 最終的な目標
  - ハザード一覧の作成
  - リスク一覧の作成
  - リスク・ハザード分析プロセスの確立
  - 分析事例集の作成
- 本年度の目標
  - ハザード分類案の提示
  - リスク分類案の提示
  - 既存の分析方法の調査
  - リスク・ハザード分析プロセス案の提示
  - 分析事例の作成
    - 候補：電気自動車とスマートハウスの双方向接続
      - 九州大学福田研究室で行われる予定の実験の成果を引用

リスク・ハザード分析プラットフォームの目標

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ 最終的な目標

- リスク・ハザード分析プラットフォームの実現

■ 本年度の目標

- リスク・ハザード分析プラットフォームの要求仕様作成
  - プラットフォームのユースケースの作成
    - プラットフォームの利用シーンを明らかにする
  - トレーサビリティを確保すべきコンポーネントの決定
    - トレーサビリティを確保すべきコンポーネントの候補
      - ハザード，リスク，ディペンダビリティ仕様，
      - システムアーキテクチャ，システムモデル（の図式）

    （以下は，余裕があれば）
      - 分析に必要となる定量データの種類

モデル記法の標準化の目標

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ 最終的な目標

- 統一的な記法の構築
  - 記法の国際標準化を目指す
- モデル記述ガイドラインの作成
  - モデルの抽象レベルを複数設定し，
    レベル毎に記述すべき標準的な記述内容（特性や項目等）を定める

■ 今年度の目標

- モデル記法案の作成
  - モデルの抽象レベルや何を記述するかに応じて，
    適切な記法を既存のものから選択
    - モデル記述マップ（次頁参照）の洗練
- 記述内容の標準化案の作成
  - モデル記述マップに従い，
    モデルの抽象レベルを幾つか設定し記述例を作成
  - 作成した記述例を基に，レベル毎に記述すべき性質や項目を洗い出す
- モデル記述テンプレートの作成
  - 標準化案をまとめる過程で作成される記述例から，
    汎用的に利用できるクラスライブラリを抽出

モデル記法マップ

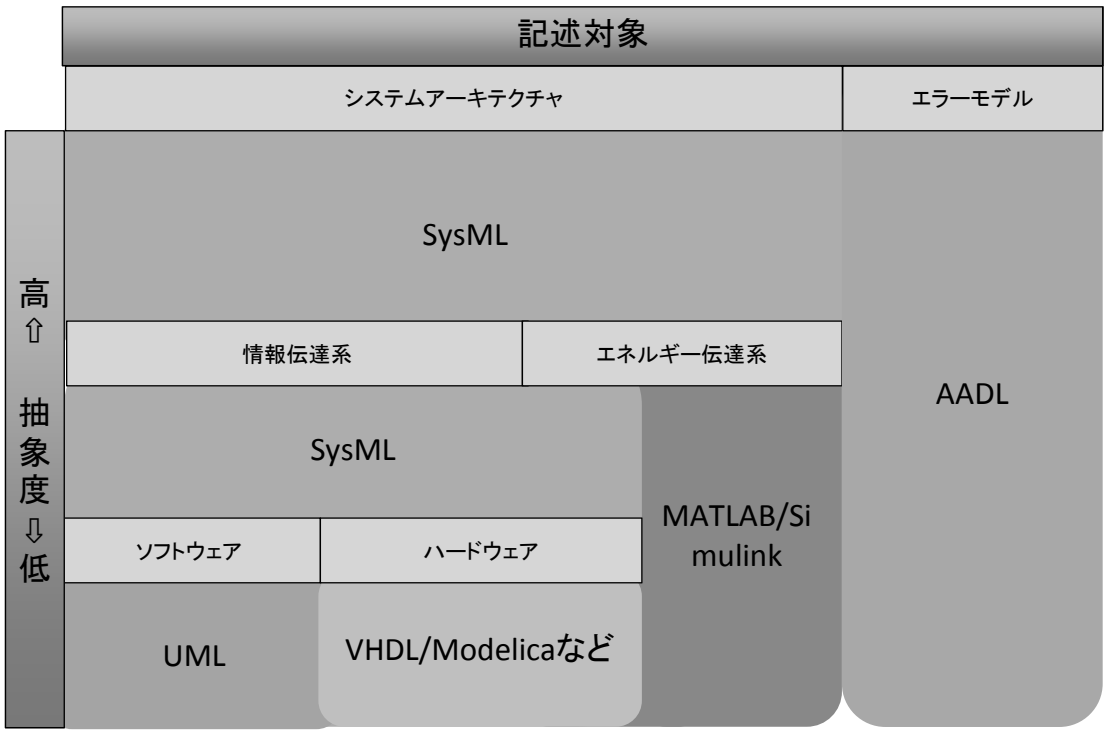
SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ モデル記法マップとは
- スマートシステムのモデル記述に必要なモデル記法について、使用するタイミングや抽象度をマトリックスとして記述したマップ

■ 補足事項
- SysML、MATLAB/Simulink、UML、AADLなどのモデル記述要素を指す
  - SysML：System Modeling Languageの略でシステムモデルを記述する
  - MATLAB/Simulink：Math Works社の製品で、制御モデルを記述するためのツール
  - UML：Unified Modeling Languageの略で、ソフトウェアモデルを記述する
  - AADL：Architecture Analysis &Design Languageの略で、ソフトウェアアーキテクチャを記述する

モデル記法マップ

SEC
Software Engineering for Mo·No·Zu·Ku·Ri



モデル解析技法の目標

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

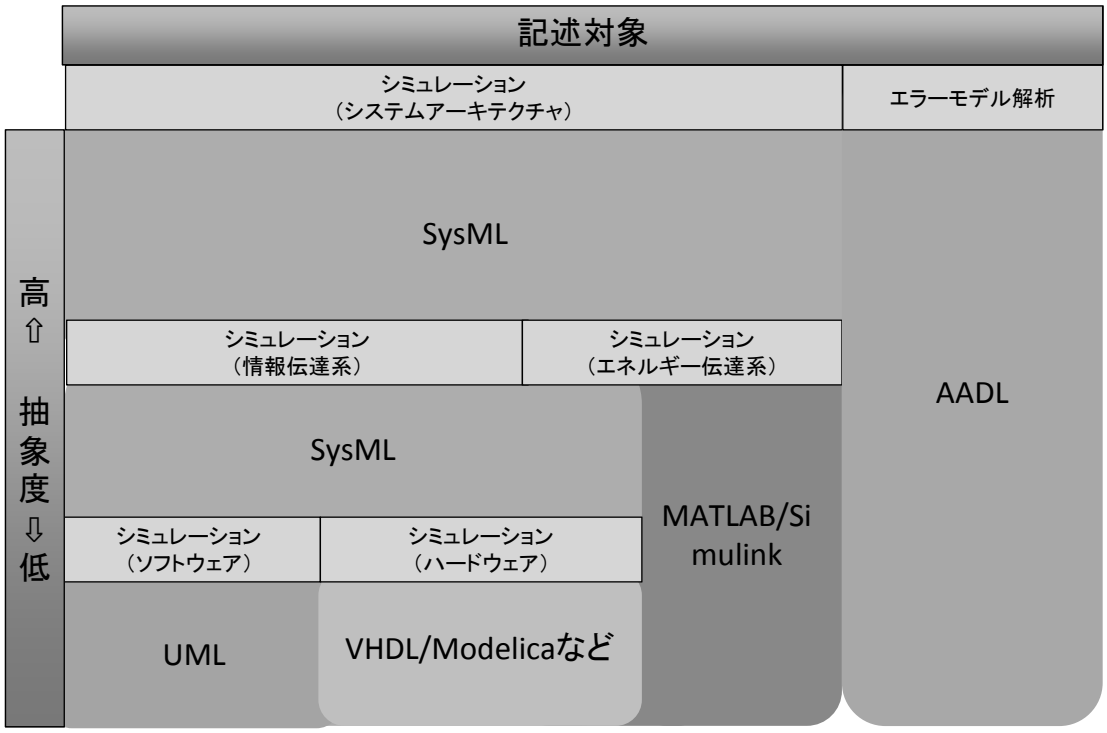
- 最終的な目標
  - 統合解析環境の構築
    - 統一的な記法で記述されたモデルを解析する環境を構築する
      - シミュレーションに基づく解析技法
      - 形式検証に基づく解析技法
- 今年度の目標
  - 記法毎の解析技法の調査
    - モデルの抽象レベル，記述内容，記法に応じて，モデルの解析に用いられる技術を調査・整理する
      - モデル解析技術マップ（次頁参照）の洗練
  - 記法をまたがる解析技法の提示
    - ある記法で記述されたモデルを別の記法のモデルに変換し，異なる記法での解析技術を適用可能にすることで，モデルに従うシステムの信頼性・安全性の精度を上げる
      - エネルギー伝達系：MATLAB/Simulink ↔ SysML
      - エラーモデル：SysML ↔ AADL

モデル解析技法マップについて

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

- モデル解析技法マップとは
  - スマートシステムのモデル解析に必要なモデル解析技法について、使用するタイミングや抽象度をマトリックスとして記述したマップ
- スマートシステムが持つ様々な側面に焦点を当てた解析技法
  - 焦点を当てるスマートシステムの側面
    - 情報伝達系としてのスマートシステム
    - エネルギー伝達系としてのスマートシステム
    - スマートシステムにおけるエラーモデル
  - 解析技法の種類
    - 情報伝達系の解析技法
      - SysML，UML等で記述された離散モデル上でのシミュレーションに基く
    - エネルギー伝達系としての解析技法
      - MATLAB/Simulink等で記述された連続モデル上でのシミュレーションに基く
    - エラーモデル上での解析技法
      - AADL Error Model Annexで記述されたエラーモデル上でシステムの信頼性等を解析

モデル解析技法マップ



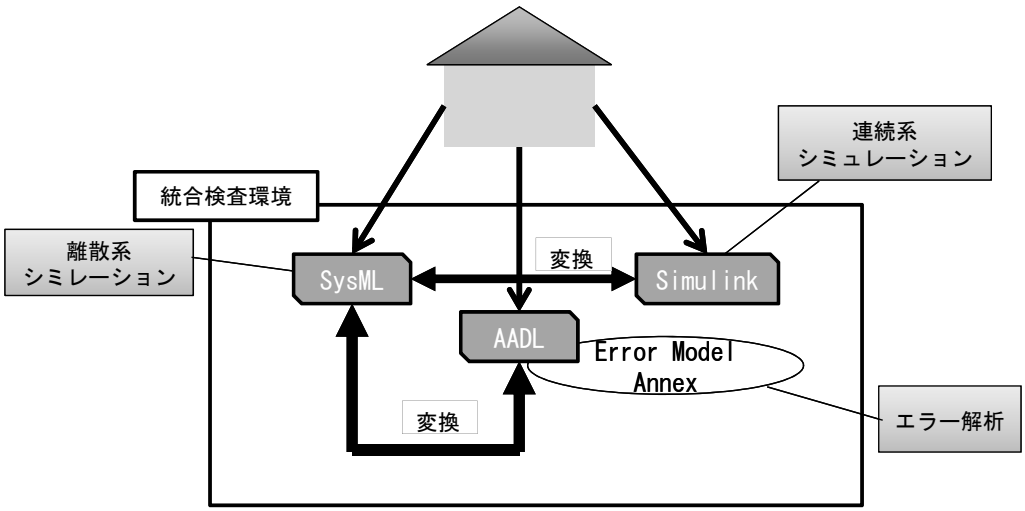
使用する解析技術

- SysML関連の解析技術
  - SysML図式の分類
  - SysMLモデルの解析
- MATLAB/Simulink関連の解析技術
  - スマートシステムのモデルベース開発事例
    MATLAB/Simulinkモデルによるコンポーネント開発とシステム検証
- AADL関連の解析技術
  - AADL Error Model Annex で記述する項目
  - エラーモデル記述例
  - エラーモデル解析技術の例

検査環境

SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ 総合検査環境
- SysMLのパラメトリックにより、離散系(SysML)と連続系(MATLAB/Simulink)を相互変換する
- SysMLとAADL(Error Model Annex)を相互変換する



システムアーキテクチャの解析

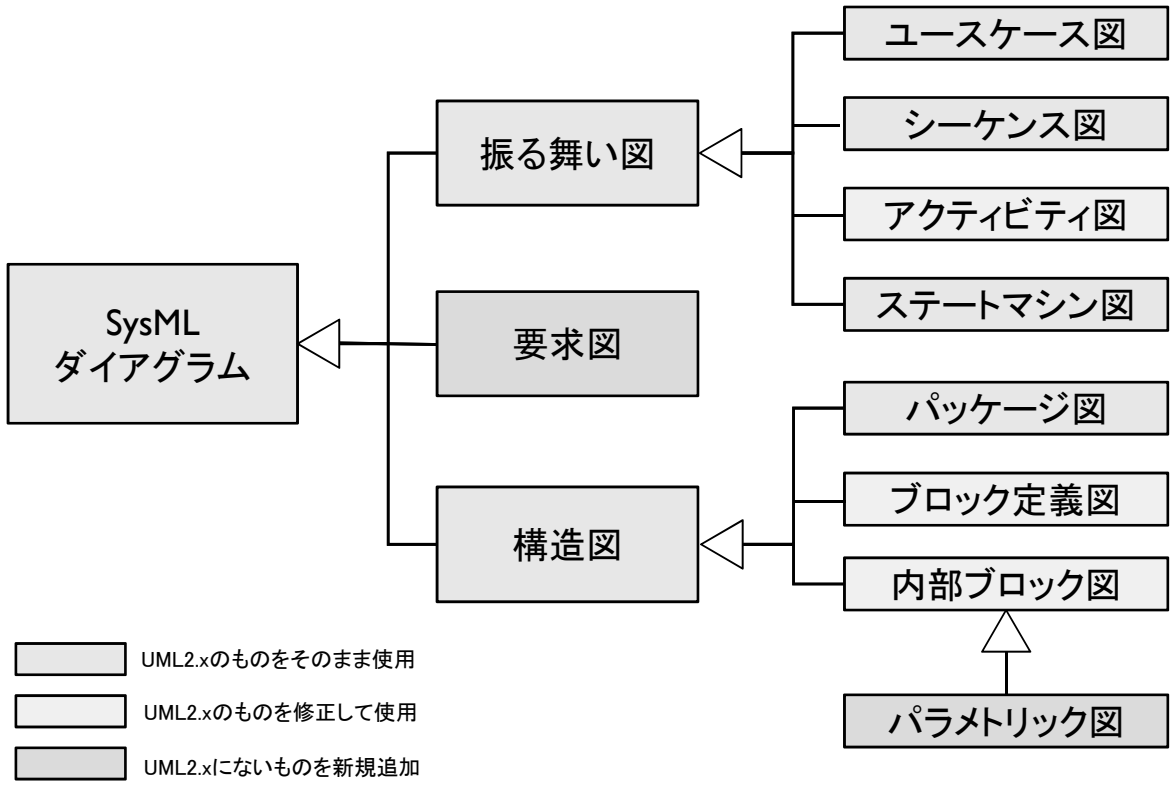
SEC
Software Engineering for Mo·No·Zu·Ku·Ri

■ スコープ
- スマートシステム全体の信頼性・安全性の確認

■ 概要
- SysMLで記述した抽象度の高いアーキテクチャを解析
  - ⇒SysMLモデルの解析＝システム全体の解析
  - ⇒詳細化の過程で発生する手戻りを防止
  - ⇒開発の早い段階での全体の整合性を確認
  - ⇒SysMLモデルや解析結果をエネルギー伝達系や情報伝達系にハンドオフ

SysML図式の分類

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri



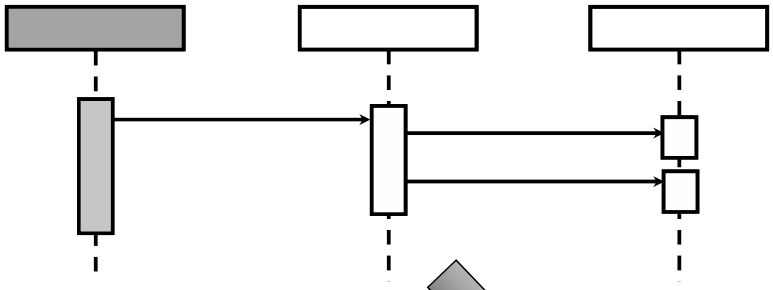
SysMLモデルの解析

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

- SysMLで記述したシステムの振舞い、構造、要求を解析
- 振舞いの解析
  - アクティビティ図での解析
  - シーケンス図での解析
  - ステートマシン図での解析
- 構造の解析
  - パラメトリック図での解析
  - 内部ブロック図での解析
  - ブロック定義図の解析と調整
- 要求の解析
  - ユースケース図の解析と調整
  - 要求図の解析と調整
- 振舞い，構造，要求の間の整合性の確認

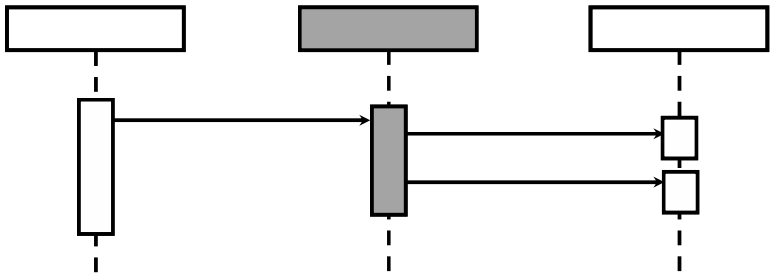
振舞いの検証(例:シーケンス図)

SEC
Software Engineering
for Mo·No·Zu·Ku·Ri



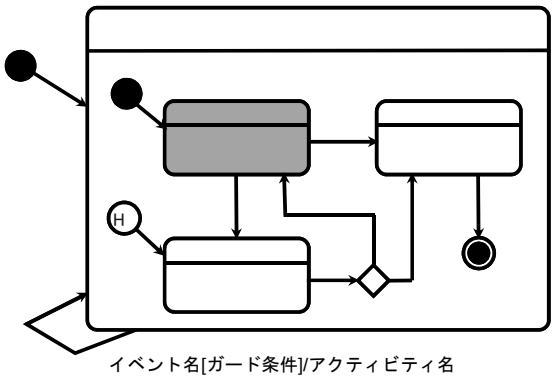
シーケンスが正しいかを
トレースしてコンポーネン
ト間のコラボレーションに
誤りがないか検証する

シーケンスをトレースする



振舞いの検証(例:ステートマシン図)

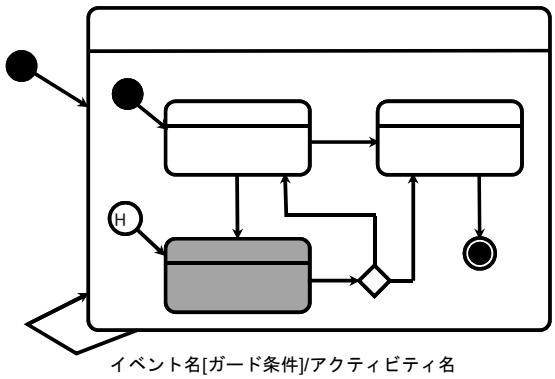
SEC
Software Engineering
for Mo·No·Zu·Ku·Ri



イベント名[ガード条件]/アクティビティ名

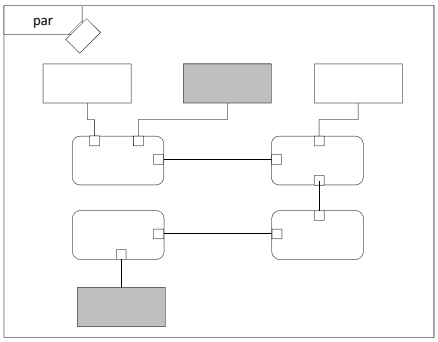
状態遷移をトレースする

状態遷移が正しいかをトレース
して、システムの振舞いに誤り
がないか検証する



イベント名[ガード条件]/アクティビティ名

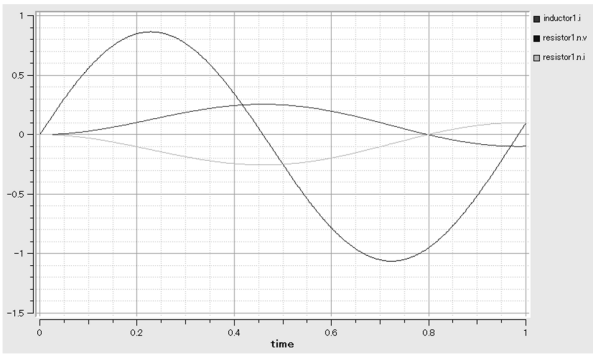
構造の検証(例:パラメトリック図)



SysMLのパラメトリック図

パラメトリックが正しいかをシミュレーションして
制約や制御ロジックに誤りがないか検証する

パラメトリック図をシミュレーションする



エネルギー伝達系の解析

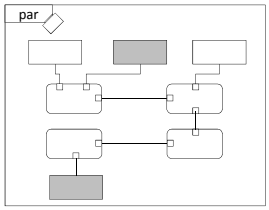
SEC
Software Engineering
for Mo·No·Zu·Ku·Ri

■ スコープ

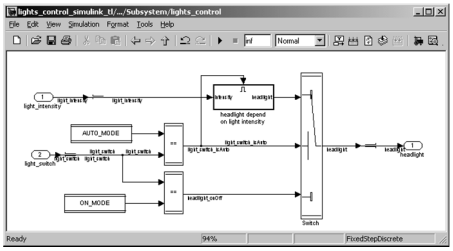
- スマートシステムのエネルギー伝達系の設計の妥当性を確認

■ 概要

- 下記記法で記述したモデルを解析
  - MATLAB/Simulink, Modelica等の連続系モデル記述言語
  - SysML(ブロック図+パラメトリック図)
- 一つのエネルギー伝達系の解析を, MATLAB/SimulinkやSysMLといった異なる記法によるモデル上で行うことで, 解析の制度を高める
  - SysMLのパラメトリック図とMATLAB/Simulinkモデル間の相互変換技術に基づく
  - SysMLのパラメトリック図でもシミュレーションできるが、さらに詳しく検討する場合は、MATLAB/Simulinkに移行してシミュレーションを実施し、その結果をシステムモデルに反映する
- また、制御モデル自体はMATLAB/Simulinkで作成し、シミュレーションを実施する



SysMLのパラメトリック図



ご清聴ありがとうございました。