



JSAE 2023年秋季大会
システムズエンジニアリングシンポジウム

SysML v2へ: v1との違いと 新標準から見るMBSEの進化



3DEXPERIENCE

2023/10/12
ダッソー・システムズ(株)
Thomas BRUN
河村 憲

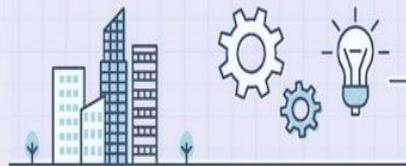
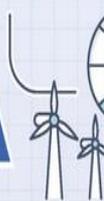


システムズ:

エンジニアリング:

シンポジウム

~ 複雑化する開発に組織として対応するために ~



概要

モデルベースシステムズエンジニアリングの実践を支援するシステムズモデリング言語SysMLは、現在大きく進化しようとしています。

2023年7月、新しい標準仕様SysMLv2のベータ版が、標準化団体OMGより発行されました。

SysMLv2では、利用性、表現力、拡張性等を高める言語体系とし、エンジニアリング活動に必要な情報との相互運用性を確保するAPI仕様を標準化しています。

SysMLの成り立ち、SysMLv1との具体的な差異を含めたSysMLv2の内容、そしてv2への移行の必要性やその狙いについて解説します。

アジェンダ



DS BRUN Thomas (ブランチマ) より

PART1: SysML v2への歩みとDSの関わり (約15分)

- システムズエンジニアリングに対するSysMLの位置づけ
- SysMLの歴史やSysML v2の必要性と狙い
- SysML v2の進行状況とダッソー・システムズの貢献



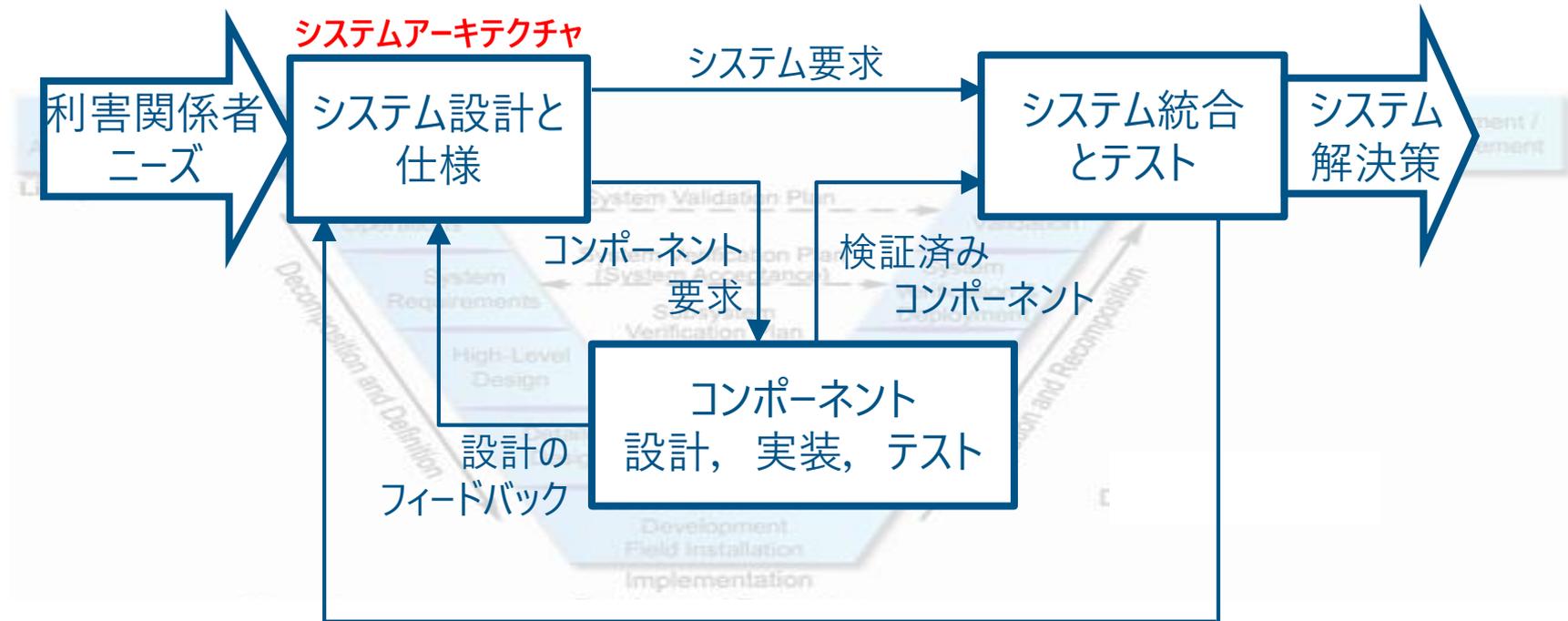
DS 河村 憲 (かわむらけん) より

PART2: SysML v2の解説 (約15分)

- SysML v2のキーポイントとv1との違い (メタモデル・記述方法・API)
- v1モデル資産のv2への移行パス
- ダッソー・システムズの技術ロードマップとまとめ

システムズエンジニアリングとは？（概要・簡易版）

ISO15288で定義されている「システムズエンジニアリング」はスコープが広く、下記の技術プロセス全体に加えて、マネージメントや組織的プロジェクト実現プロセスなども含む



慶應義塾大学SDM西村先生
のご講演資料をもとに加筆

統合とテストからのフィードバック

SEに対するMBSEやSysMLの「位置づけ」

製品開発

のために

「より良い製品」
を開発するための
「より良い設計」
を実現したい。
解決すべき
具体的な
エンジニアリング
課題がある

SE

のために

ドキュメント管理
を脱却して、
システムズ
エンジニアリング
を効率的に
幅広く
取り入れたい

MBSE

のために

国際標準の
言語に基づいて、
社外にも通じる
アーキテクチャ
を書きたい

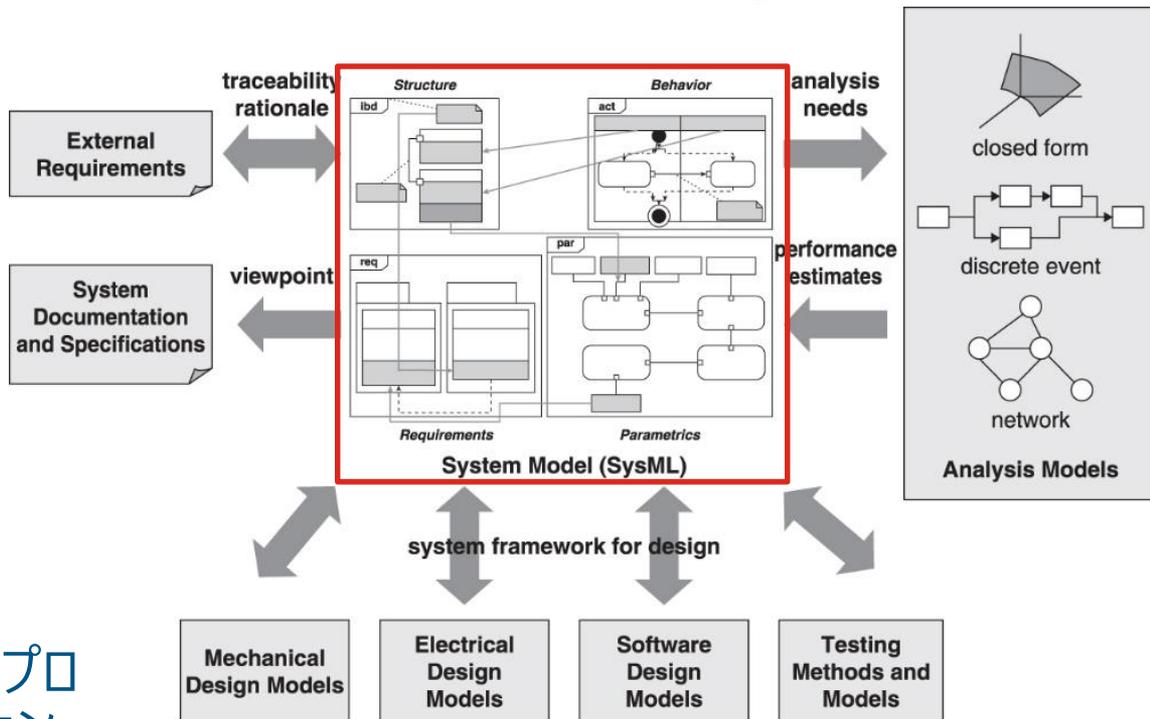
SysML

原点でも
目標でもなく、
手段です！

SysMLとは？（概要・簡易版）

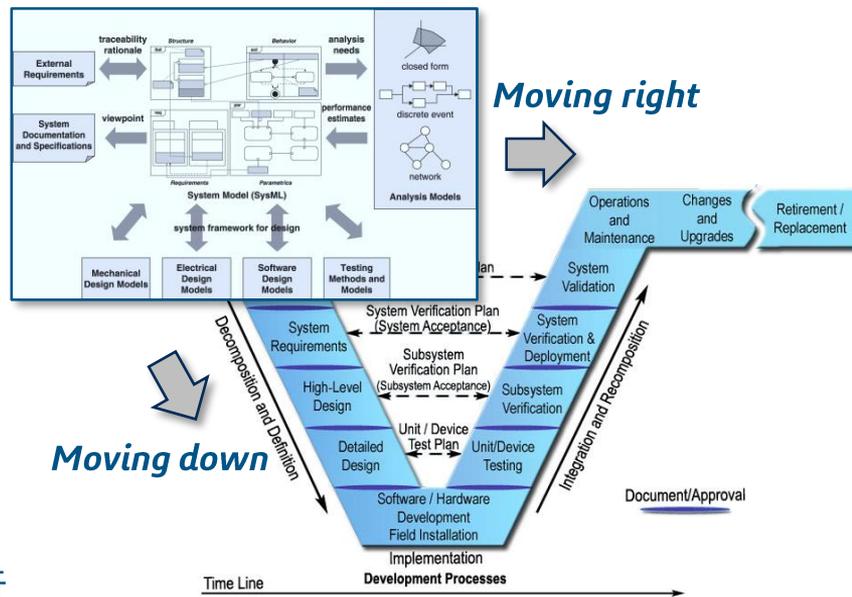


- ▶ 世界で最も使われているシステムズモデリング言語
- ▶ 標準化団体OMGによってUML言語を元に策定
- ▶ 主に4種類の情報をダイアグラムで表現する：
 - ▷ 要求（Requirements）
 - ▷ 構造（Structure）
 - ▷ 振舞い（Behavior）
 - ▷ パラメータ（Parametrics）
- ▶ システムズアーキテクチャ以外のプロセス（分野設計、シミュレーションなど）と連携することを目標とする



SysMLを用いたMBSEのトレンドと課題

- ▶ 航空宇宙・防衛業界を中心に使われていたSE・MBSEが他業界に広がりつつある（特に自動車業界）
- ▶ マネージメント層にも必要性の理解が進み、組織・プロセスの改革に投資をする企業が増加
- ▶ 分野設計との連携（**Move down**）やシミュレーションとの連携（**Move right**）が急ピッチに進化中
- ▶ 「サイロ化を解消するための手段」ですが、MBSEが「もう一つのサイロ」になってしまうケースも
- ▶ 多くの企業はドキュメントベースからモデルベースへの移行にまだ苦戦中（文化的課題が大きい？）
- ▶ 現場展開やトレーニングのベストプラクティスが不十分



SysML v2 の狙い (概要)

- ▶ フォーマルセマンティックに基づく、**UMLに制限されない新メタモデル**
- ▶ 表現力と正確さの向上で幅広いユーザーにより使いやすく
- ▶ テキスト表記も兼用できるより豊富なビューポイント
- ▶ 標準API実装によるPLMや設計ツールとより密接なインテグレーションへ
- ▶ 初期段階からv1からv2への移行パスを計画

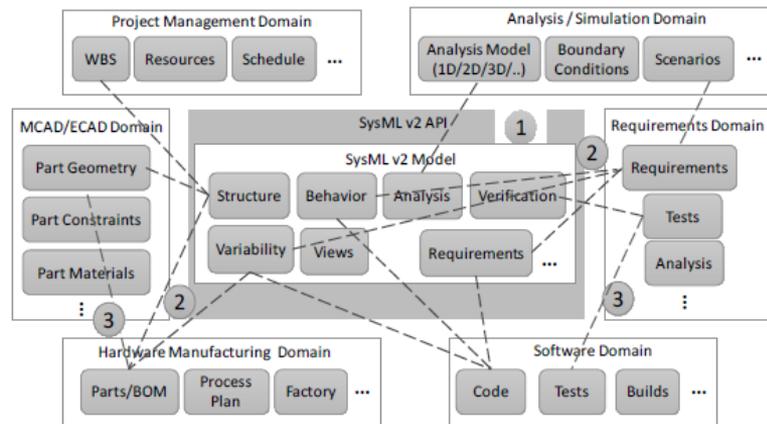
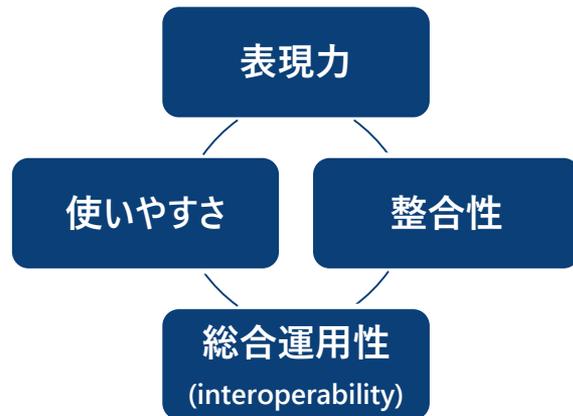


Figure 2 – Scope of SysML v2 API and Services (showing inter-model relationships)

History of SysML: SysML誕生から現状とその先



SysML v1.x
リリース



2017/05
ISO/IEC 19514:2017 (SysML1.4)
ISO標準としてSysML公開

Today

v1.7は
v1最後の
リリース
とされる

SysML v2
動向



Today

レビュー

「早くても」
2024 Q1
...or 2025?

次世代システムズモデリング言語
のワーキンググループ検討スタート

SysML v2 提案仕様は、
SysML v2 Submission Team (SST)より
OMGに提出



SysML v2 Submission Team (SST) とは？

- ▶ エンドユーザ、ベンダー、アカデミック、行政機関などからなる幅広いチーム
 - ▷ 60以上の組織を代表する100人以上のメンバーが参加
- ▶ 2つのRFP(SysML v2 RFP, v2 API & Services RFP)への提出物を作成
- ▶ RFP要求とユーザニーズに基づく



SST Participating Organizations

Academia/Research End User | Tool Vendors Government Rep | INCOSE rep * SST

<ul style="list-style-type: none">Aerospace CorpAirbusANSYS mediniArasArmy Aviation & Missile CenterBAEBigLever SoftwareBoeingCCDC Armaments CenterCEAContact SoftwareDraper LabElbit Systems of AmericaESTACAFordFraunhofer FOKUSGeneral MotorsGeorge Mason UniversityGfSEGeorgia Tech/GTRIIBMIdaho National Laboratory	<ul style="list-style-type: none">IncQuery LabsIntercaxItemisJet Propulsion LabJohn DeereKenntnisLieberLieberLightstreet ConsultingLockheed MartinLSSTMaplesoftMgnite IncMITREModelAlchemy ConsultingModel Driven SolutionsModel FoundryNISTNo Magic/Dassault SystemsOAKObeoOOSEOstfold University College	<ul style="list-style-type: none">Phoenix IntegrationPTCQualtech Systems, Inc (QSI)RaytheonRolls RoyceSAF Consulting *SAICSiemensSierra Nevada CorporationSimulaSystem Strategy *Tata Consultancy ServicesThalesThematrixTom SawyerUFRPEUniversity of CantabriaUniversity of Alabama in HuntsvilleUniversity of Detroit MercyUniversity of Kaiserslautern / VPEVitech88solutions
---	--	---

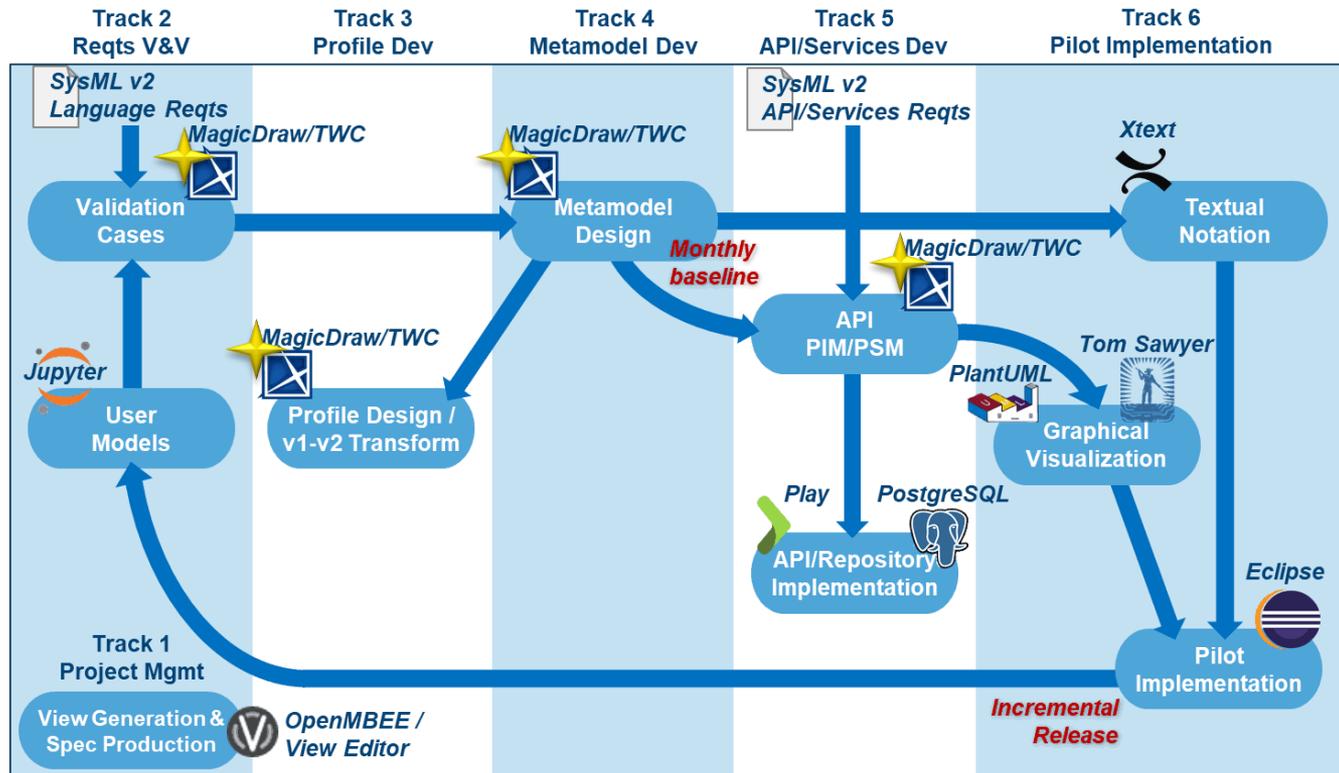
05 November 2019

出典: SysML v2 Submission Team (SST) - A Look Ahead at SysML v2, INCOSE 2020 IW, 26 Jan 2020

SSTのアジャイル開発プロセスとモデルベース環境

▶ 弊社Dassault Systemesはメンバーとしてだけでなく、SST自体の活動や実験的実装環境を支える技術も提供

- ▷ 検証ケース定義
- ▷ プロファイル定義
- ▷ v1-v2変換
- ▷ メタモデル設計
- ▷ APIのPIM・PSM
- ▷ ...

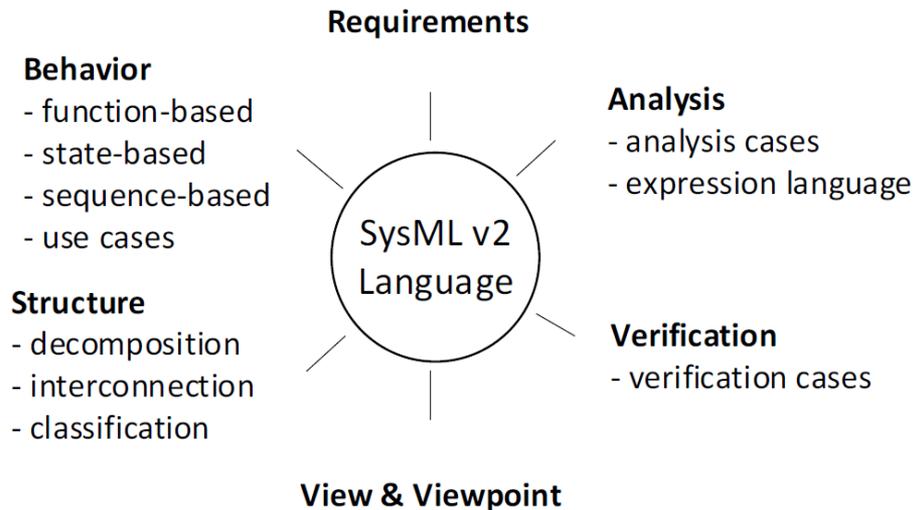


出典: <https://www.ppi-int.com/systems-engineering-newsjournal/ppi-syen-95/>

SysML v2 のキーポイント

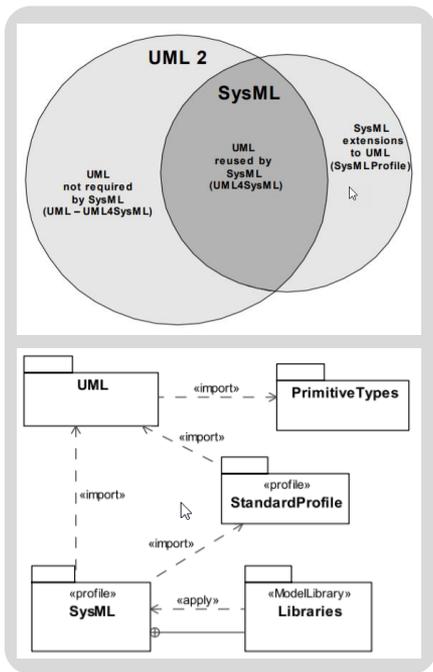
- ▶ UMLに制約されない新しいメタモデル
 - ▷ システムズ・モデリングに焦点を当て、ほとんどのUMLモデリング機能を保持
 - ▷ 形式的なセマンティクスに基づく
- ▶ 柔軟なビュー&ビューポイント仕様に
基づくロバストなビジュアルライゼーション
 - ▷ グラフィカル、表形式、テクスチュアル
- ▶ モデルにアクセスするための
標準化されたAPI

言語のケーパビリティ



SysML v2 言語アーキテクチャ

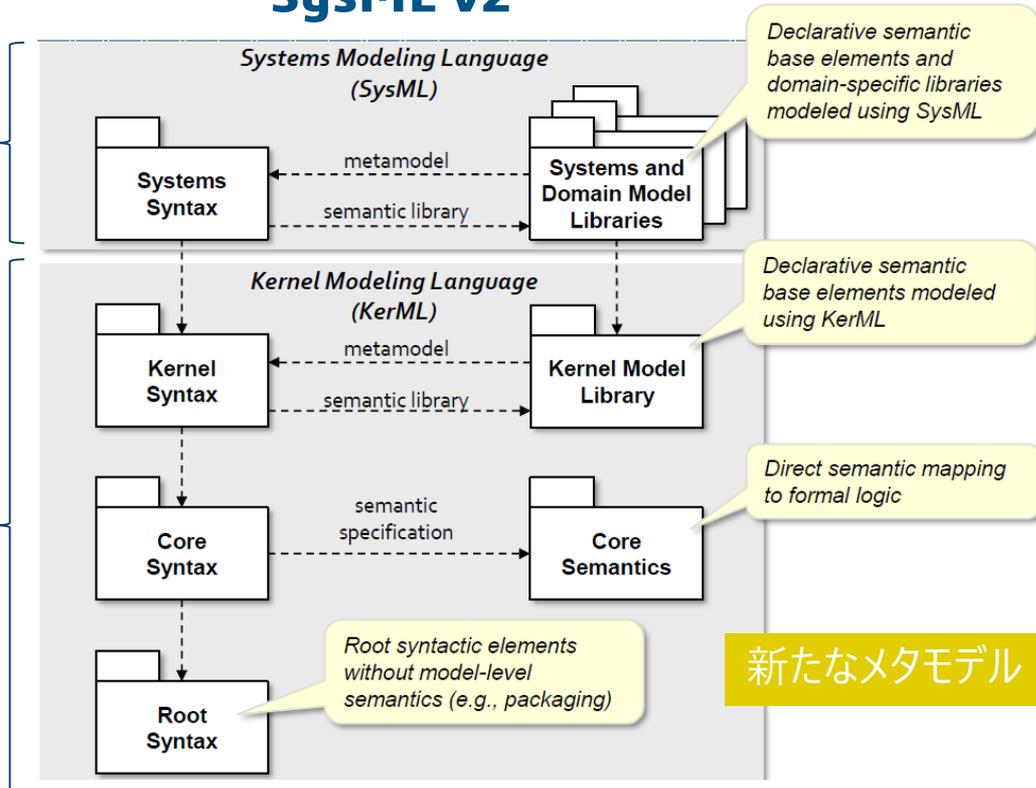
SysML v1.x



SysMLでの 利用部分

基礎部分 (KerML)

SysML v2



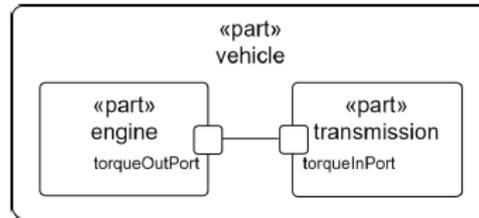
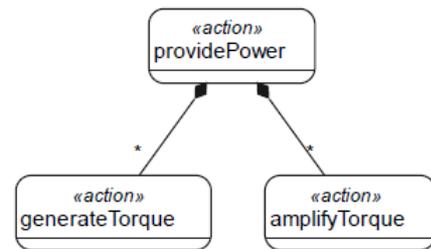
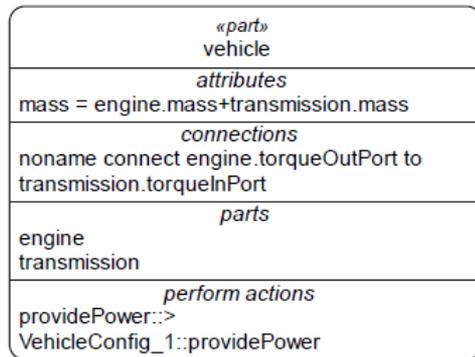
テクスチュアルとグラフィカルの記述例

簡単な自動車モデル

テクスチュアル・ノーテーション

```
part vehicle{
  attribute mass = engine.mass+transmission.mass;
  perform providePower;
  part engine{
    attribute mass;
    port torqueOutPort;
    perform providePower.generateTorque;
  }
  part transmission{
    attribute mass;
    port torqueInPort;
    perform providePower.amplifyTorque;
  }
  connect engine.torqueOutPort to transmission.torqueInPort;
}
action providePower{
  action generateTorque;
  action amplifyTorque;
}
```

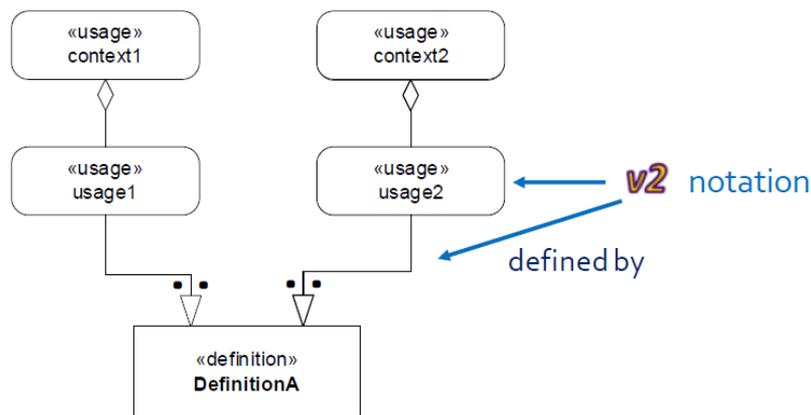
グラフィカル・ノーテーション



SysML v2でキーとなる考え方：DefinitionとUsage

SysML v1からあったDefinition・Usageの概念をより明確に定義し、使いやすく

- ▶ Usageにフォーカスしたモデリングアプローチを可能に
- ▶ より正確に直感的に利用できるようにするためのパラダイムシフト



v1とv2の用語マッピング
(v2では一貫したパターンを適用)

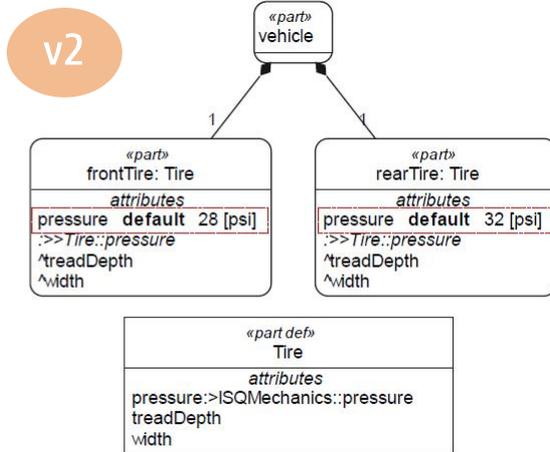
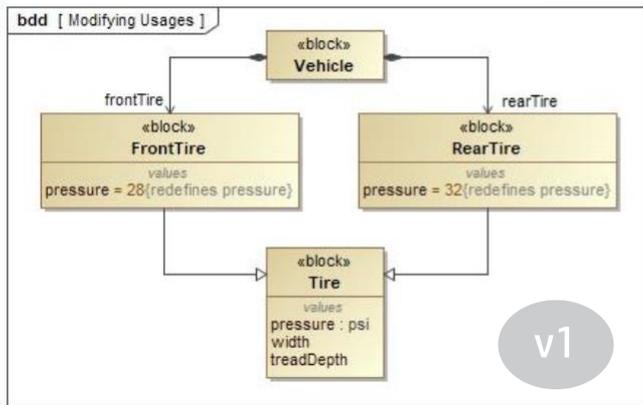
SysML v2	SysML v1
part / part def	part property / block
attribute / attribute def	value property / value type
port / port def	proxy port / interface block
action / action def	action / activity
state / state def	state / state machine
constraint / constraint def	constraint property / constraint block
requirement / requirement def	requirement
connection / connection def	connector / association block
view / view def	view

Usage / Definition

Usage / Definition

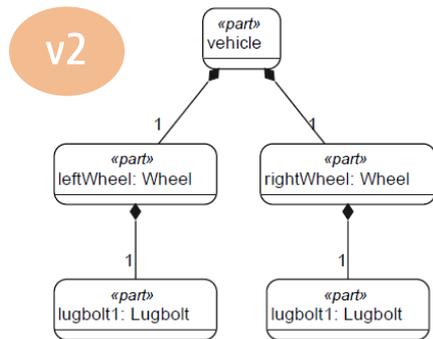
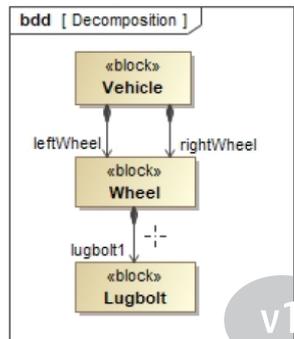
SysML v1とv2の記述の違い（例）

<<part>> は Usage要素



パートの分解(Decomposition)・編集

- ✓ SysML v2はパートの分解(Decomposition)をサポート
- ✓ パートは直接編集可



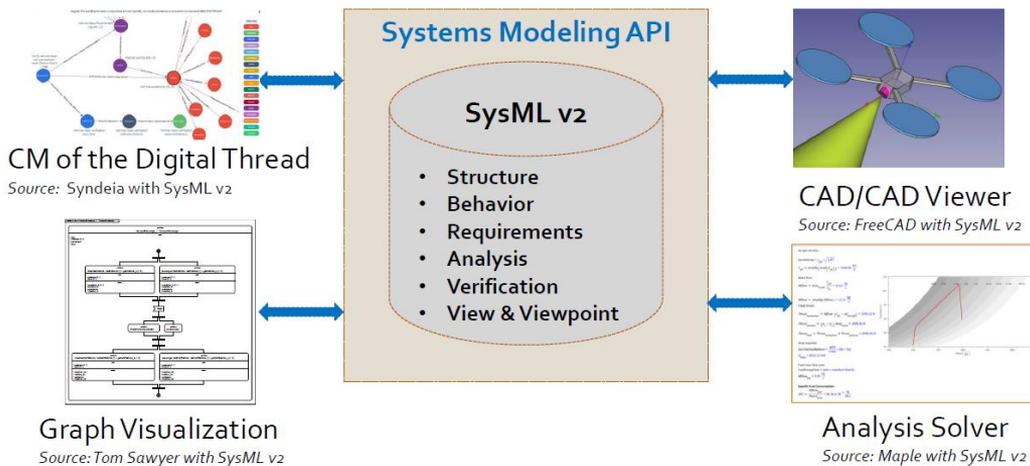
分解(Decomposition)における曖昧さの低減

- ✓ SysML v1 : leftWheelのlugbolt1とrightWheelのlugbolt1は区別がつかない
- ✓ SysML v2 : leftWheelとrightWheelはそれぞれ独自のlugboltを持つ（異なるトルクを定義可）

SysML v2 API & サービス

- ▶ 他のツールやアプリケーションが標準的な方法でSysMLモデルにアクセスできるようにする
- ▶ 次のようなサービスを提供：
 - ▷ 要素の作成、更新、削除
 - ▷ モデルのクエリとナビゲート
 - ▷ その他、モデル管理、分析、変換、ファイルエクスポートなど
- ▶ 共通パターンをサポート
 - ▷ 構成ツリーのナビゲーション
 - ▷ ブランチ作成
 - ▷ 複数の制約でのクエリ
- ▶ REST/HTTP、Java、OSLCなどの様々な実装技術の使用を容易にする

標準APIを通じてSysML v2モデルと接続



SysML v1と比較したv2のまとめ

✓ 学習と使用のためによりシンプルに

- ▷ アドオンではなく、システムズエンジニアリングの概念をメタモデルの中に設計
- ▷ 一貫したDefinitionとUsageのパターンの適用
- ▷ より一貫した用語定義
- ▷ パーツ、アクション、...を分解できる機能
- ▷ パッケージフィルタでより柔軟なモデル構成

✓ より正確に

- ▷ テキスチュアル構文・表現言語
- ▷ 形式的なセマンティックの根拠
- ▷ 制約としての要求

✓ より表現力豊かに

- ▷ バリエーションモデリング
- ▷ 分析ケース、トレードオフ分析
- ▷ Individuals、スナップショット、タイムスライス
- ▷ よりロバストな定量的特性（例：ベクターなど）
- ▷ 簡単なジオメトリ
- ▷ クエリ/フィルター表現
- ▷ メタデータ

✓ より拡張性高く

- ▷ よりシンプルな言語拡張機能
 - ▶ モデルライブラリに基づく

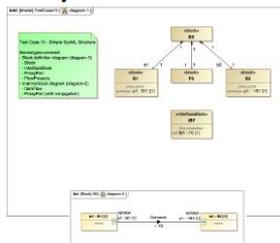
✓ より相互運用性高く

- ▷ 標準化されたAPI

SysML v1 から v2 への移行パス

- 影響評価のためのパイロットプログラムを開始
- ロードマップに関するツールベンダーとの協議を開始
- 更新のための段階的計画の準備
 - MBSEプラクティス
 - ツールインフラ
 - トレーニング
 - メトリクス
 - 参照モデルと再利用リポジトリ
 - 実践に関するMBSEコミュニティ
 - プロジェクト開発の基準

SysML v1 Model



SysML v2 XML



SysML v2 Textual Notation

```
package eVehicleLibrary {
  attribute def ElectricEnergy :> ScalarValues::Integer;
  attribute def BatteryCapacity :> ScalarValues::Integer;
  attribute def Speed :> ScalarValues::Integer;
  port def PowerOutPort { out energy : ElectricEnergy;
}
interface def PowerInterface {
  end supplierPort : PowerOutPort;
  end consumerPort : ~PowerOutPort;
}
package eVehicleDefinitions {
  import eVehicleLibrary::*;
  part def Wheel {
    value size : ScalarValues::Integer;
  }
  part def Battery {
    value capacity : BatteryCapacity;
  }
  part def Engine;
}
.....
```

Source: SST Track 3 Presentation
Yves Bernard, Tim Weikiens
o8 February 2022

*Transition Guidance being developed
by DoD office of DE, Modeling & Simulation*

弊社の SysML v2 への貢献と技術ロードマップ

SysML v2

3DSが

SysML v2ドラフト版

SysML v2 最終リリース

RFP 発行

SST実装WG参加

Q1 2023

Q1 2024 以降

December, 2017

Q4, 2021



SST参加開始

レビュー、技術サポート、プロトタイプ作成

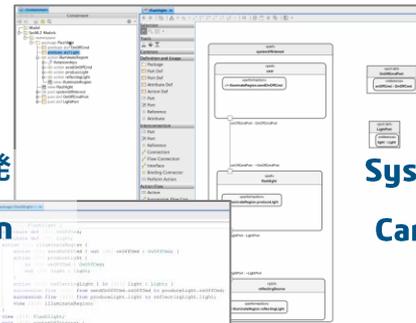
ツールのα・β版開発

SysML v2 準拠ツール提供

- ✓ メタモデル
- ✓ グラフィカルノーテーション/レプレゼンテーション
- ✓ 振る舞い実行セマンティクス
- ✓ ジオメトリセマンティクス
- ✓ v1からv2への変換
- ✓ モデル交換
- ✓ 言語拡張

Alpha Evaluation program

Cameo, CATIA Magic & 3DEXPERIENCE



3DS CATIA | Magic



将来的に複数のツールが
SysML v2対応:

- Cameo, CATIA Magic, Teamwork Cloud
- 3DEXPERIENCE Platform

MBSE導入の「針を動かす」ために

SysML v2 移行のための重要な成功要因：

実務者にとって

- 創造的に – より少ないリソースでより多くのことを行う
- 統合を可能にし、MODSIM (モデリング&シミュレーション) を普及させる
- 使いやすさ、経験と学習の継続性

組織にとって

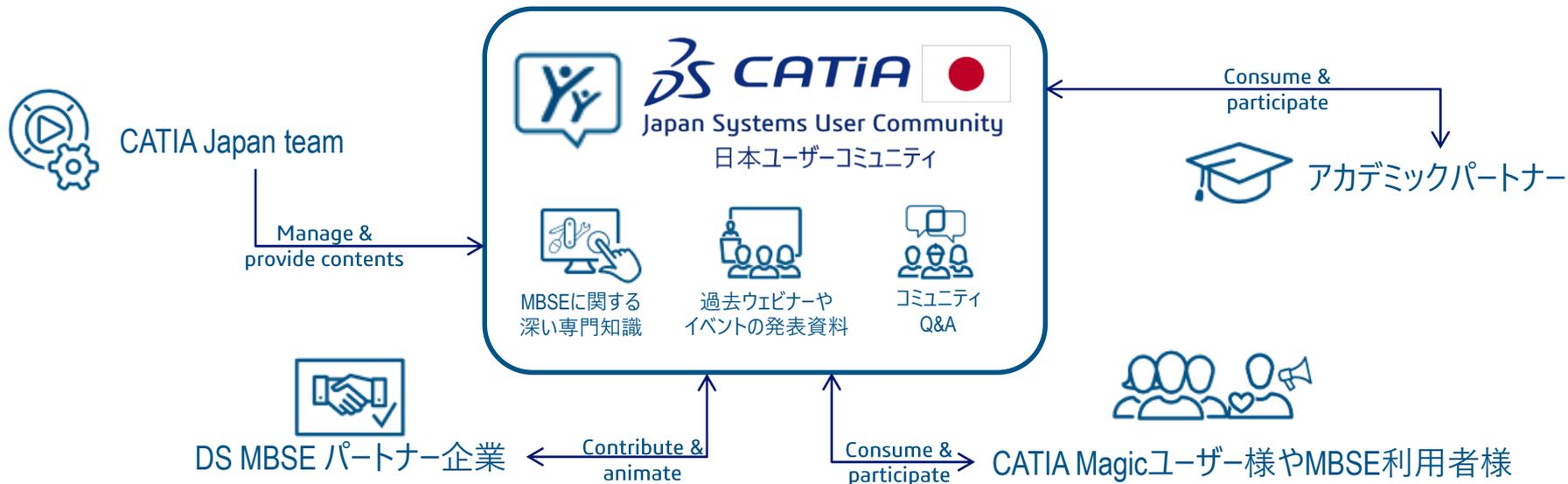
- v1への投資を維持し、中断を最小限に抑える
- より良い統合や多様なステークホルダへの幅広いアピール
- アジリティ、生産性と効率性、定量化可能なROI

学術にとって

- SEプログラムのためのカタリスト
- 知名度/評判

弊社MBSEユーザーコミュニティへのご招待

日本のMBSEコミュニティの活性化に貢献するために、「日本語で」情報を交換する場を設立



参加無料。CATIA Magicユーザーでなくても参加可能
全世界ユーザーコミュニティ（英語）にもアクセス可能

[参加申し込みリンクはこちら](#)

(弊社3DXPFで無料アカウント登録後、参加承認に少々お時間いただきます)

DISCOVER MORE
ON 3DS.COM



参考資料

- OMG SysML v2 beta Specification: <https://www.omg.org/spec/SysML/2.0/Beta1/About-SysML>
- SysML v2 Submission Team (SST): <https://github.com/Systems-Modeling>
 - SysML v2 Release / doc : <https://github.com/Systems-Modeling/SysML-v2-Release/tree/master/doc>
- YouTube – Object Management Group channel
 - Introducing SysML v2: <https://www.youtube.com/watch?v=4iCZVyo2Ce0>
 - Systems Modeling Language™ v2 (SysML® v2) Overview: <https://www.youtube.com/watch?v=FXBlwmw5dEQ>