Introduction To Model-Based System Engineering (MBSE) and SysML

Presented at the Delaware Valley INCOSE Chapter Meeting
July 30, 2015

Laura E. Hart
Lockheed Martin, IS&GS
Laura.E.Hart@lmco.com
610-354-6529

LOCKHEED MARTIN

Topics

- MBE/MBSE Terminology and Overview
- SysML Overview
- Object Oriented SE Methodology (OOSEM)
- Modeling Tools and the Environment

Terminology

- Model:
  - A simplified version of a concept, phenomenon, relationship, structure or system
  - A graphical, mathematical or physical representation
  - An abstraction of reality by eliminating unnecessary components
  - The objectives of a model include;
    - to facilitate understanding
    - to aid in decision making, examine 'what if' scenarios
    - to explain, control, and predict events

> “Model-Based Engineering (MBE): An approach to engineering that *uses models as an integral part of the technical baseline* that includes the requirements, analysis, design, implementation, and verification of a capability, system, and/or product throughout the acquisition life cycle.”
>
> Final Report, Model-Based Engineering Subcommittee, NDIA, Feb. 2011

Terminology

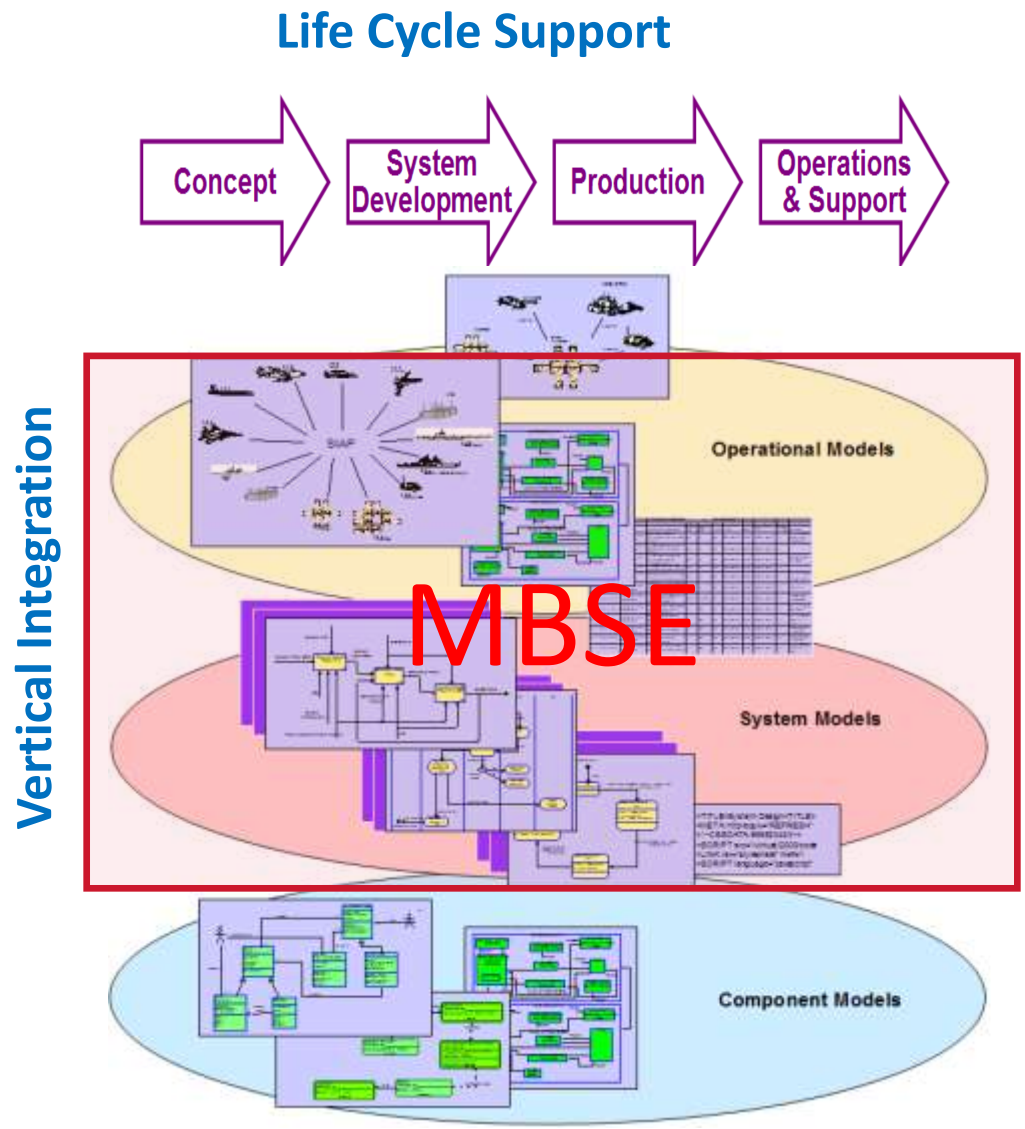
- MBSE: Model Based Systems Engineering
  - Those aspects of MBE specifically associated with SE
  - includes behavioral analysis, system architecture, requirement traceability, performance analysis, simulation, test, etc.

"Model-based systems engineering (MBSE) is the *formalized application of modeling* to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases."

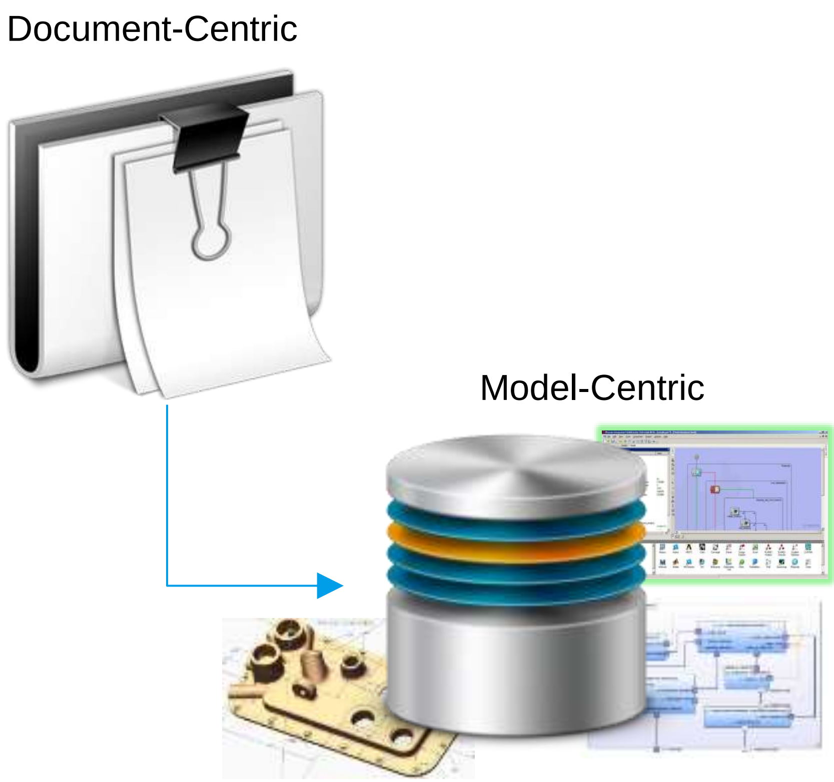
INCOSE SE Vision 2020 (INCOSE-TP-2004-004-02, Sep 2007)

MBSE Focus

- Formalizes the practice of systems development through the use of models
- Broad in scope
  - Includes multiple modeling domains across life cycle from SOS to component
- Results in quality/productivity improvements & lower risk
  - Rigor and precision
  - Communications among development team and customer
  - Management of complexity



Model-Based Engineering: What, Why and How?



- Digital models have been common in engineering since the late 1960s but today's focus on Model-based Engineering goes beyond the use of disparate models

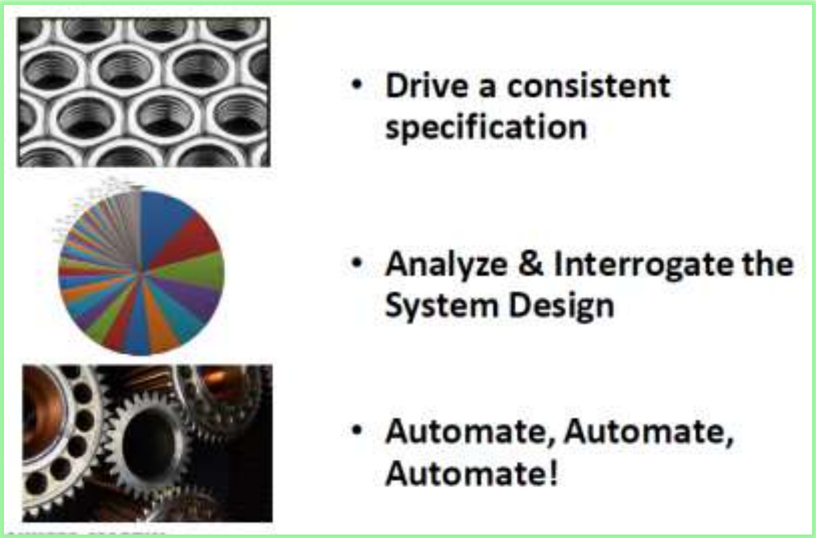
- Model-based Engineering moves the record of authority from documents to digital models including M-CAD, E-CAD, SysML and UML managed in a data rich environment

- Shifting to model-based enables engineering teams to more readily understand design change impacts, communicate design intent and analyze a system design before it is built

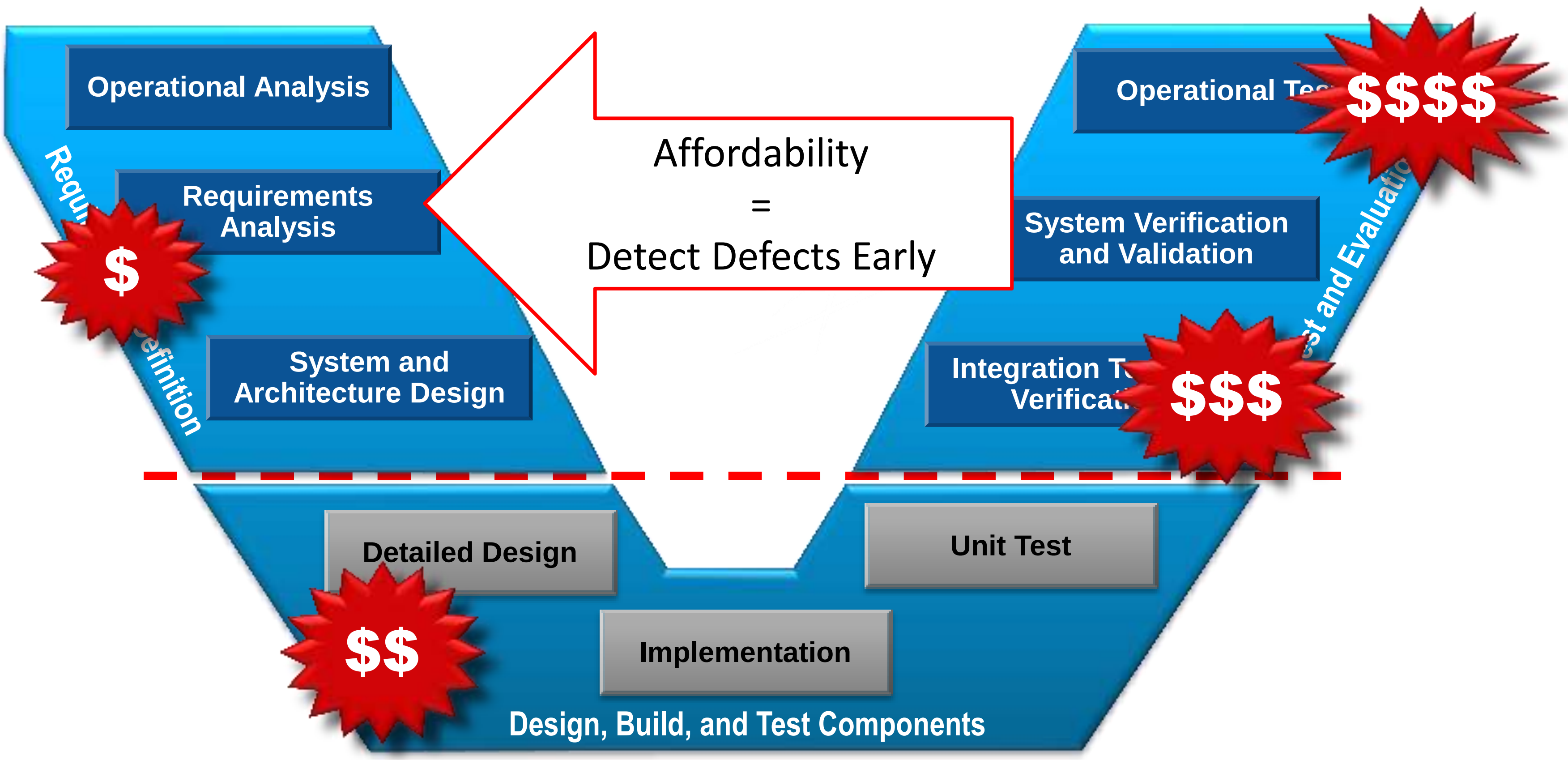
Model-Based Engineering: What, Why and How?

- Model-based Systems Engineering provides a mechanisms for driving more systems engineering depth without increasing costs
- Data-centric specifications enable automation and optimization, allowing SEs to focus on value added tasks and ensure a balanced approach is taken
- Unprecedented levels of systems understanding can be achieved through integrated analytics, tied to a model-centric technical baseline.

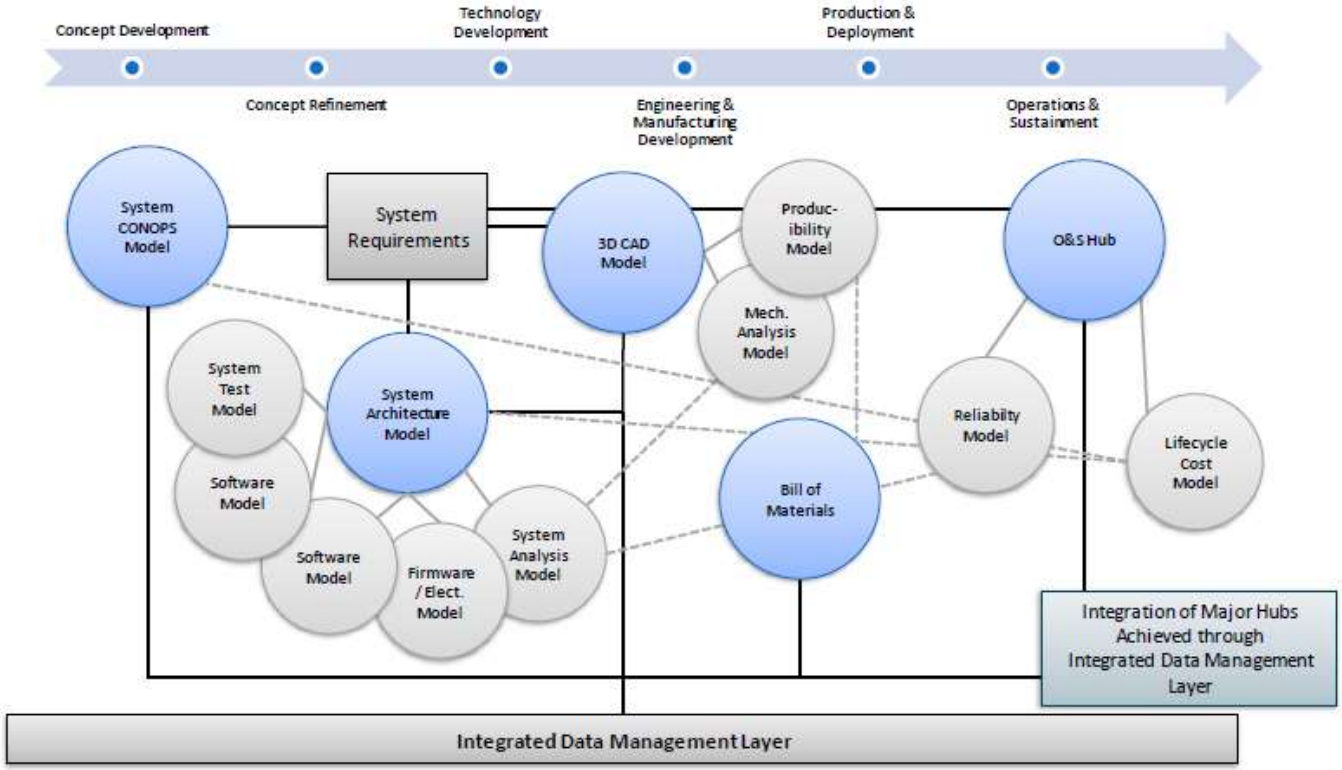
Why Model???



Detect Defects Earlier



Model-Based Engineering: What, Why and How?



The key to a successful model-based approach is scoping the problem!

- What do you want to get out of your models?
- What fidelity do you need to accomplish those goals?
- What are the success criteria for the effort?

Scoping and managing a modeling effort is both an art and a science

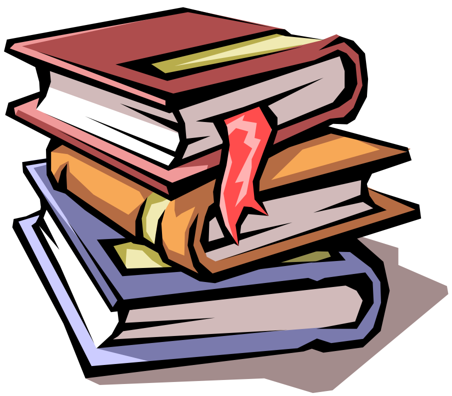
- Driving change in an organization takes time and continuous investment

Model Based Environment Characteristics

- Set of interconnected Models
  - Models are an abstraction of Reality
  - Structure, Behavior and Requirements
- Standard Language
  - Graphical Notation, Syntax, Semantics
  - Visual focus
  - Static and Dynamic
- Shared System Information Base

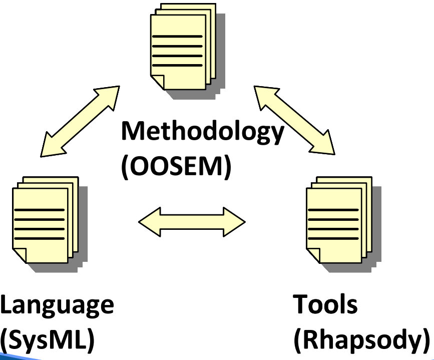
SE Practices for Describing Systems

Past

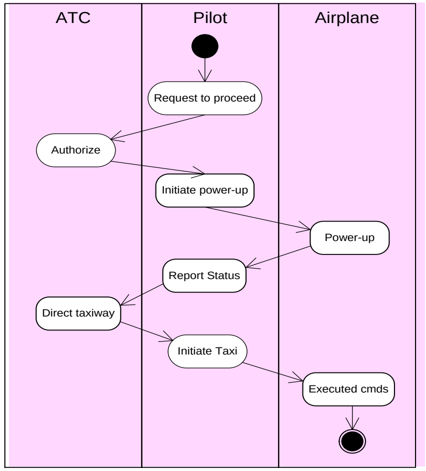


Document
Centric

- Specifications
- Interface requirements
- System design
- Analysis & Trade-off
- Test plans



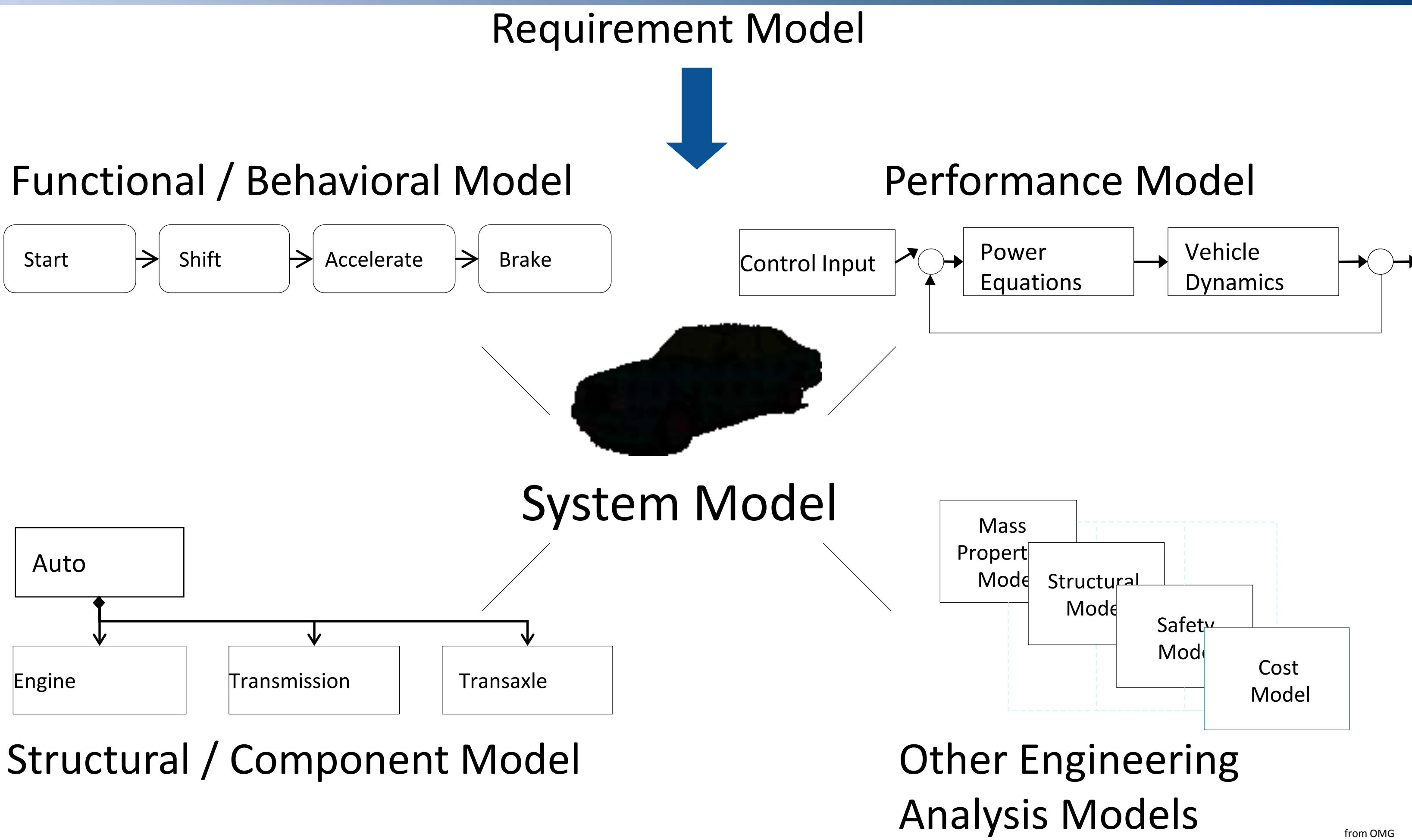
Future



Model
Centric

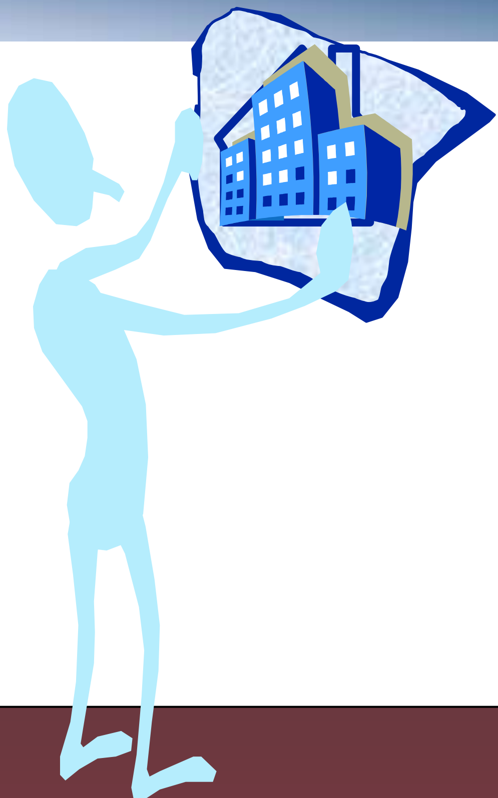
Moving from Document centric to Model centric

System Modeling



Integrated System Model Must Address Multiple Aspects of a System

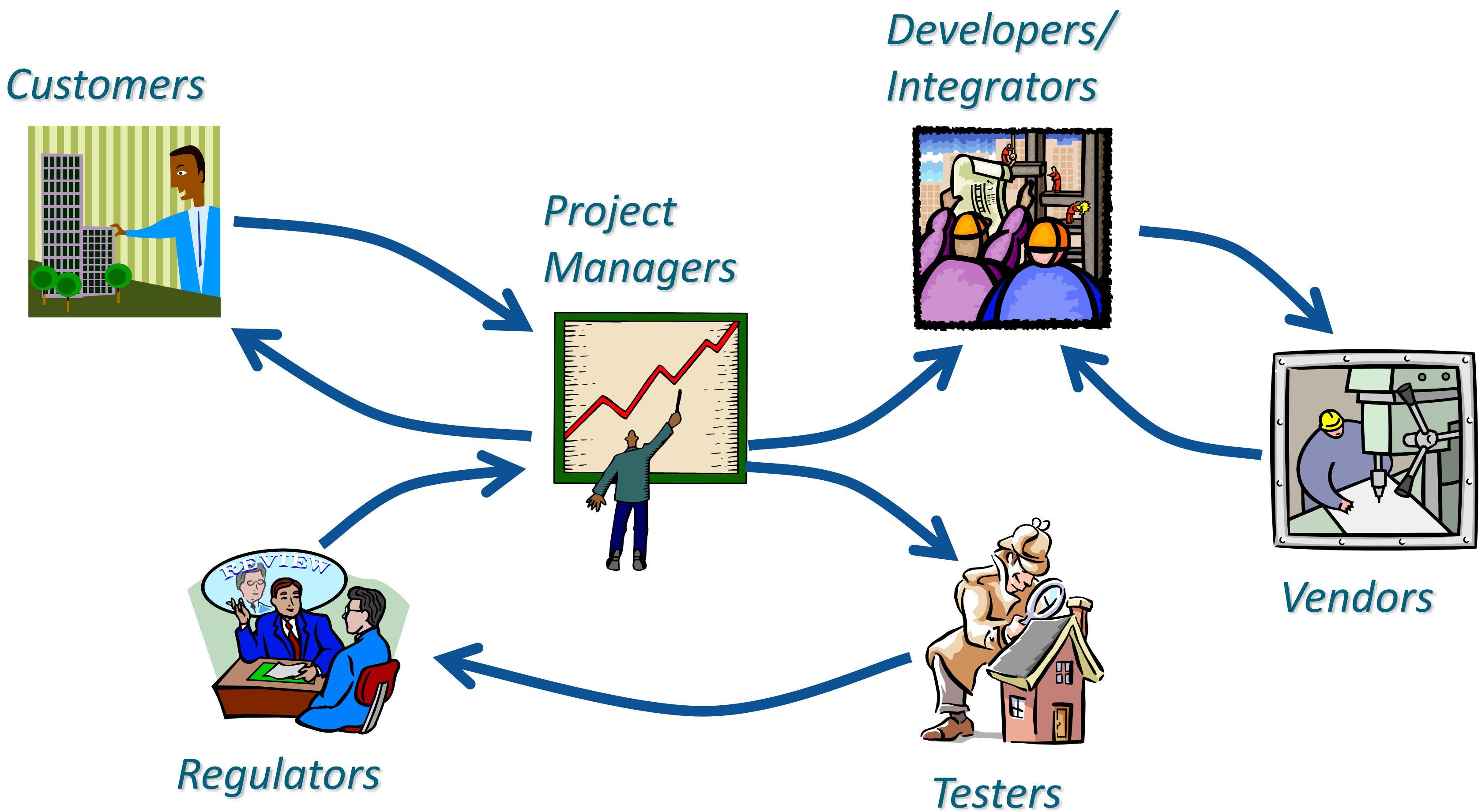
Modeling Helps Produce



- Increased ability to manage System Complexity
  - Model can be viewed from multiple perspectives
    - Supports concurrent and distributive teams
  - Traceability
    - Supports impact/change analysis
  - Complex development lifecycles
    - Incremental, iterative, parallel

- Improved system and software
  - Specification
  - Visualization
  - Architecture
  - Construction
  - Simulation and Test
  - Documentation
  - Validation and Verification
- Improved communications
  - Enhanced knowledge capture and transfer
  - Training Support
- Improved design quality
  - Decreased ambiguity
  - Increased precision
  - Supports evaluation of Consistency, Correctness & Completeness
  - Supports evaluation of trade space

Stakeholders Involved in System Acquisition



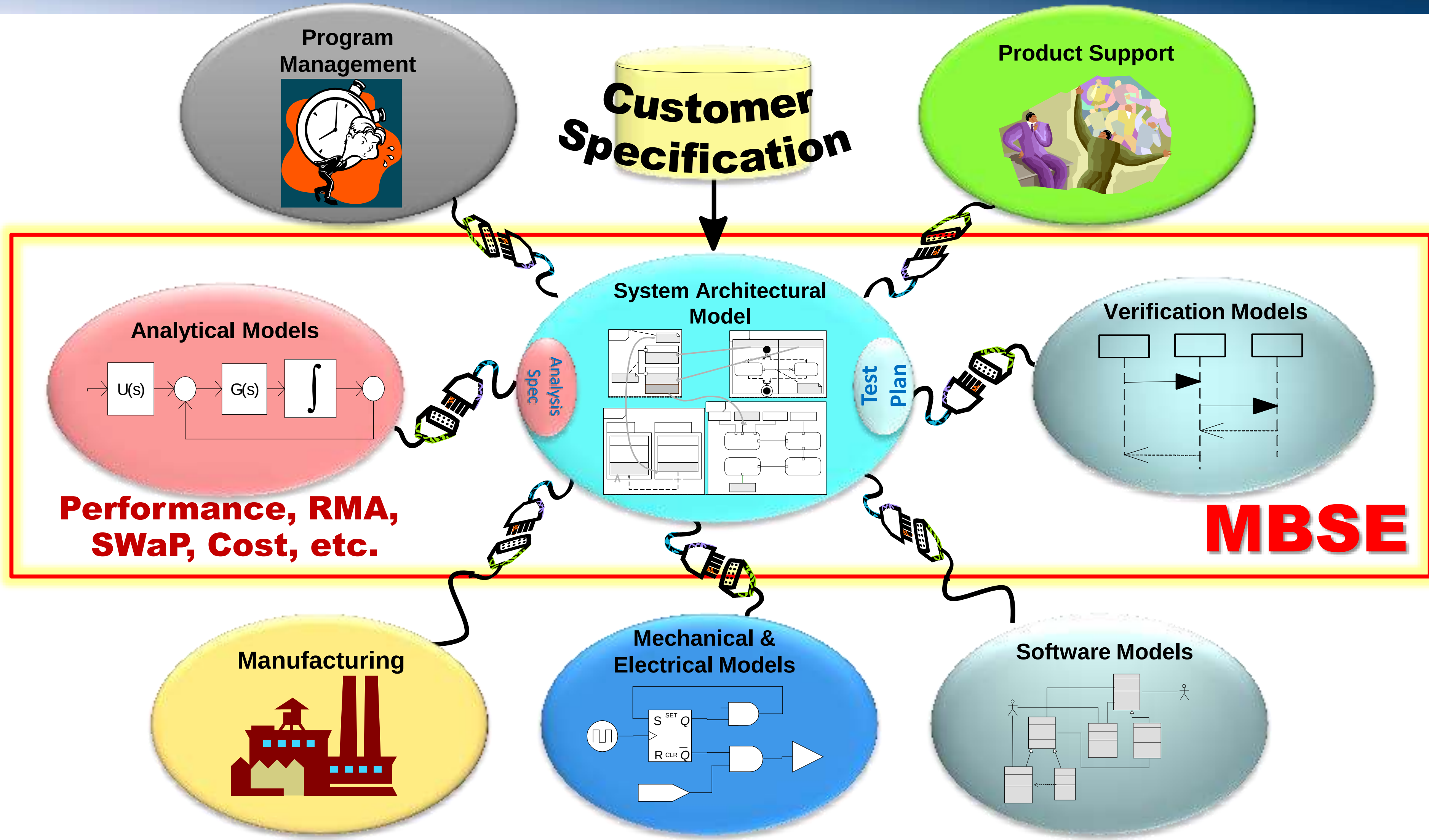
from OMG

Modeling Needed to Improve Communications across all Stakeholders

Information Base Characteristics

	Document Based	MBSE Based
Information	- Mostly Text - Add Hoc Diagrams - Loosely coupled, repeated in multiple documents	- Visual and Textual - Constructs Defined once and re-used - Shared across Domains - Consistent notation in diagrams - Defined relationships
Information Views	- By Document	- Provides Viewpoints - Filters By Domain, Problem Space, etc.
Measuring Change Impact	- Spans across Multiple Documents - Often Text Reqts. Are isolated from Structure and Behavior	- Relationships define traceability paths - Natural part of the modeling process - Programmatically Automated
Measuring Integrity - Completeness, Quality & Accuracy	- By manual inspection	- Programmatically Automated - Animation of Spec

The MBSE Integration Across Domains



System Architecture Models *vs.* Analytical Models

System Architecture Model

- Emphasize how pieces fit together into a ***consistent*** whole
- Repository-based to support capture of inter-relationships
- Used to capture
  - functions, behavior
  - structure, components, objects
  - info flow, interfaces, ports
  - Interactions, scenarios
- A vehicle to "hang" & integrate analysis products

Analytic Model

- Emphasize specific aspects of performance, ***consistent*** with the Architecture Model
- Mathematically-based to computation or simulation
- Reduce risks thru analysis, validation and optimization of:
  - MoM, MOE, MOP, KPP, TPM
  - timing, probability of hit/survival
  - reliability/availability, MTBF
  - cost, total cost of ownership
- A vehicle to solve some problem or

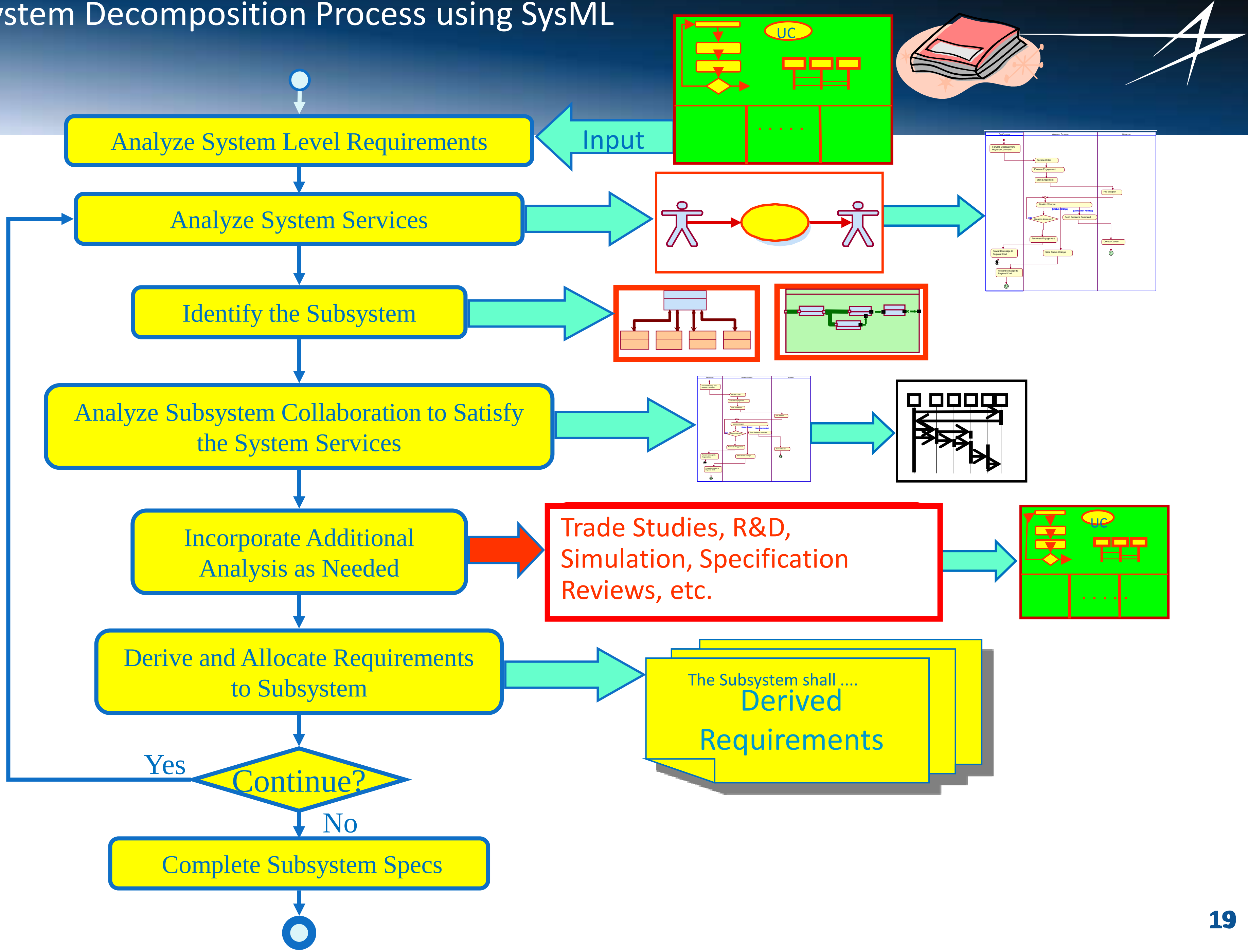
Consistency: Faithful to all known and relevant aspects of the system at the current time

System Model

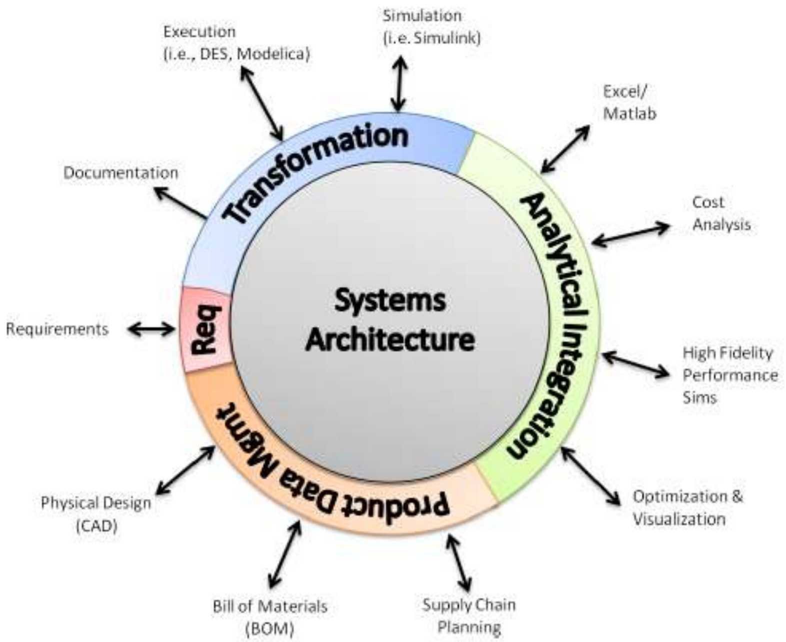
System Model – A structured representation that focuses on the overall system requirements, behavior, structure, properties, & interconnections

- Requirements
  - What are the stakeholder goals, purposes, and success conditions for the system
  - Specification of black box behavior and characteristics
- Behavior
  - What the system has to do to meet the requirements
  - Transformations of inputs to outputs (functional/activity models)
  - State/Mode-based behavioral differences (state models)
  - Responses to incoming requests for services (message models)
- Structure
  - The parts that exhibit the behavior
  - The component hierarchy, elements, and stores
- Properties
  - The performance, physical characteristics and governing rules that constrain the structure and behavior
- Interconnections
  - The way the structural elements arrange and communicate to achieve the required behavior under the given constraints

System Decomposition Process using SysML



Beyond Specifications: Integrated Systems Models

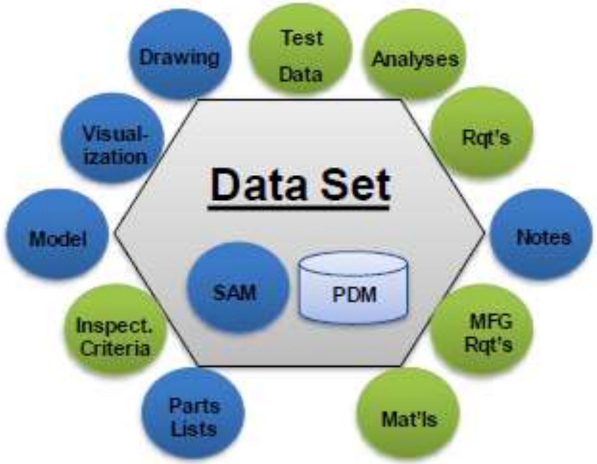


- **Model-based Systems Engineering doesn't end with the creation of specifications and ICDs**
- **A Systems Architecture Model provides a "hub" for data integration and transformation across the product lifecycle**
- **Specifically of note is the ability to link analysis through the systems model to provide insight into architectural and system level decisions**

Model-based Engineering Baseline: An Integrated Data Set

A Product Data Set Contains Artifacts That Define A Product…

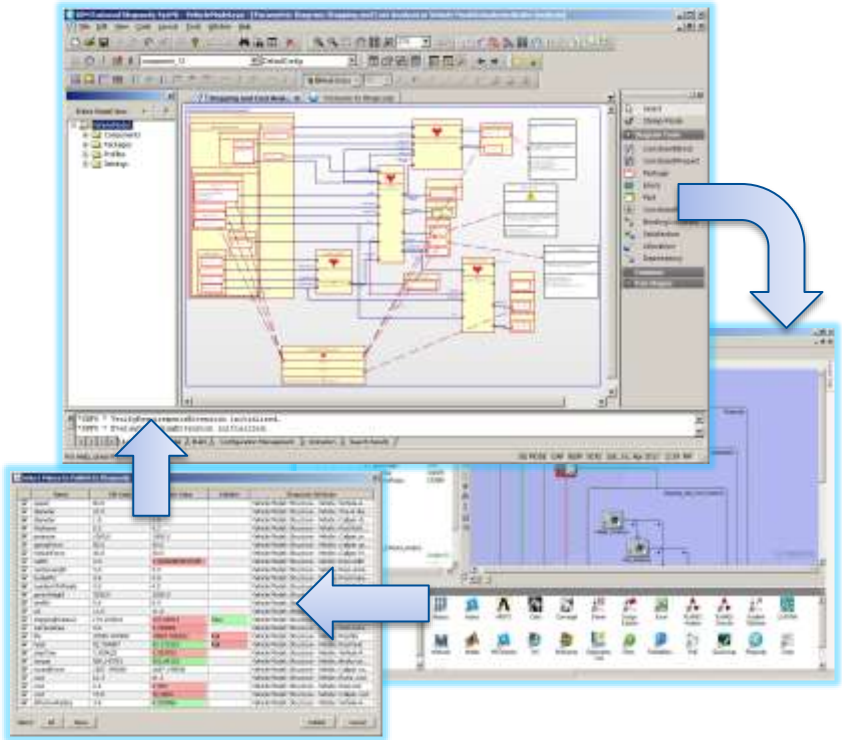
- Systems Architecture Model
- 3D Model
- Specifications
- Notes
- Metadata or Attributes
- Bill Of Material
- Design Requirements
- Other Product Descriptive Info



The Entire Data Set Must Be Managed

Systems Understanding and Analytics through Model-based Architectures

- A critical task in Systems Engineering is the upfront trade and analysis process to ensure the best value system is developed to satisfy the mission need
- As missions become more complex, understanding all items that can impact the system performance becomes harder
- Integrating high fidelity analytics through a consistently defined systems architecture can help provide insight into key system characteristics not evident through traditional analysis alone
- Integrated tools allow engineers to analyze many more system configurations against mission scenarios, helping to identify key driving requirements and lowest cost alternatives for systems design

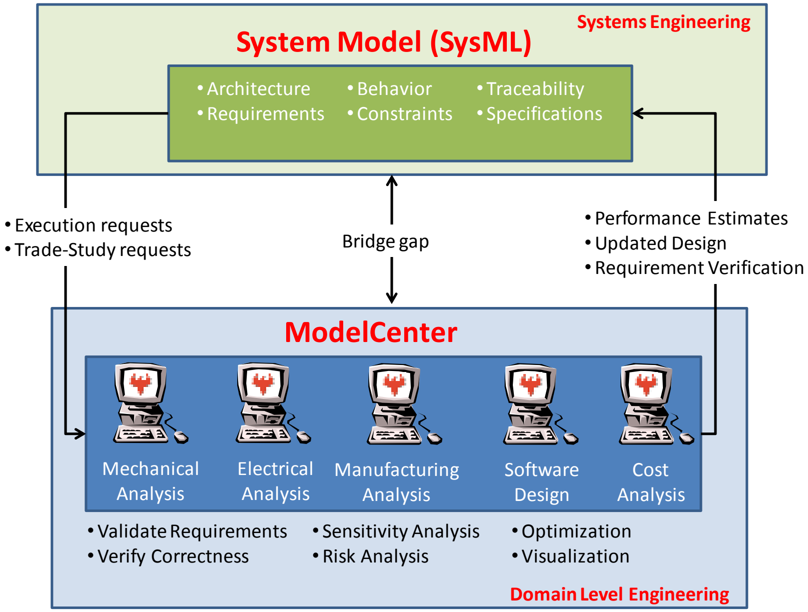


Expanding base of MBSE Tool Capabilities

- The MBSE tools marketplace has been expanding beyond just graphical modeling and requirement tools with a focus on data and model integration

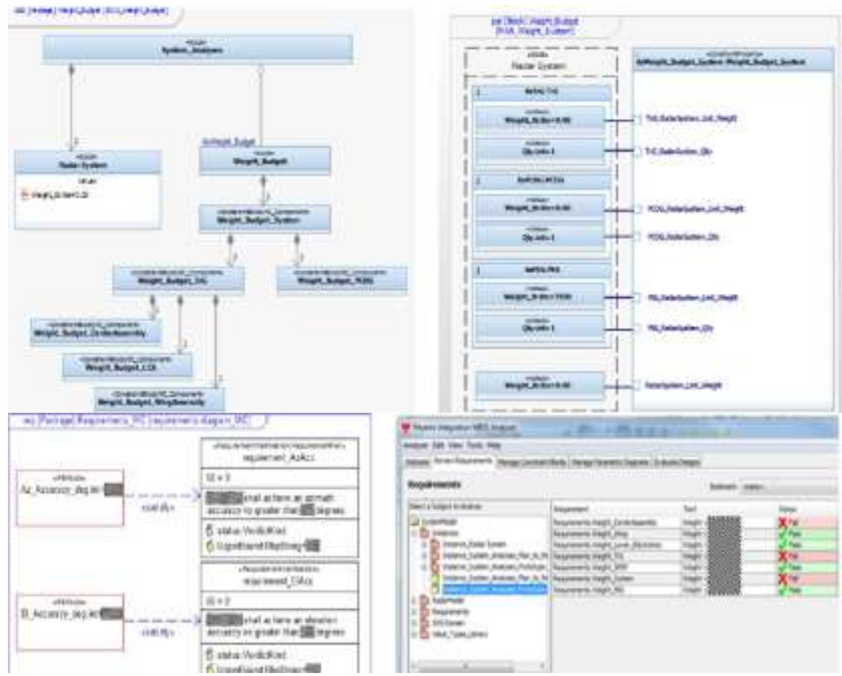
- Phoenix Integration ModelCenter ® has been expanded to help bridge SysML to multidisciplinary analysis through the release of the ModelCenter MBSE Pak

- With MBSEPak, SysML parametric models within Rhapsody® or MagicDraw® can be executed, linking requirements to design to analysis and back



Architecture Centric Analytics

- **A SysML Architecture can serve as a hub for integrated analytics, capturing analysis, analysis context, requirements and key architectural parameters**
- **Analysis context specifies the boundaries of the analysis, parametric views define the analysis to be performed and requirements diagrams can capture design goals, thresholds and driving requirements to bound the tradespace**
- **This model-centric approach provides a consistent, managed framework for analysis which often tends to be ad-hoc**



Integrated Modeling and Analysis Support for Decision Makers

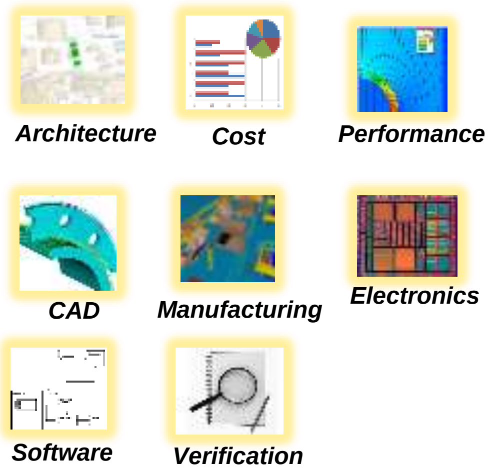
- Decision makers will have more information and options from which to draw conclusions

- Integrated analytics models will both increase the amount of information available to decision makes as well as help decision makers make sense of the information

- Tools to explore, visualize and understand a complex tradespace, rooted in MBSE can provide early insight into the impact of decisions ranging from technical solutions to complex public policies

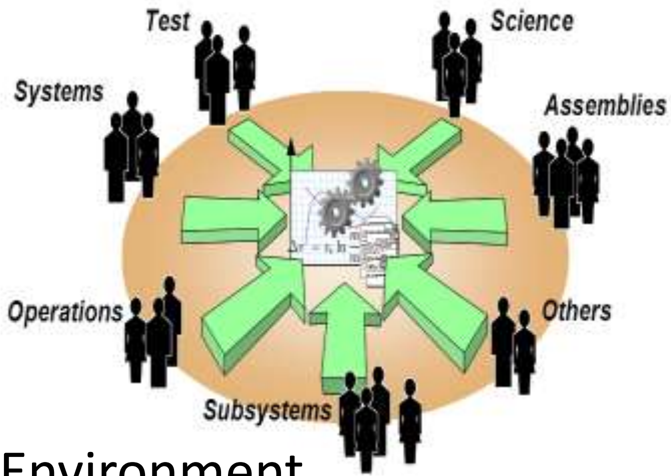
Content Credit INCOSE Systems Engineering Vision 2025

Model-based Engineering: What, Why and How?



- The primary focus of most current industry efforts to move toward a Model-based Engineering approach focus on integrating data through models
- Engineering Is Responsible To Understand All Items That Could Impact A Design And Determine A Resolution For Those Items – an integrated end-to-end modeling environment supports this role
- By bringing together varied but related models into a data rich, architecture centric environment, new levels of systems understanding can be achieved
- Model-based Systems Engineering forms a means to achieve integration

MBE Summary



- What?
  - Extends beyond Engineering
  - Covers the entire Life Cycle
  - Standards Based
  - Integrated Domains/Disciplines Data Environment
  - Doing the same SE Tasks
  - Environment for Automation and Validation
- Why? – Customers want “Cheaper, Better, Faster”
  - Reduce Design Time
  - Improve Quality
  - Make Complex Systems Affordable

Topics

- MBE/MBSE Terminology and Overview
- SysML Overview
- Object Oriented SE Methodology (OOSEM)
- Modeling Tools and the Environment

What is SysML?

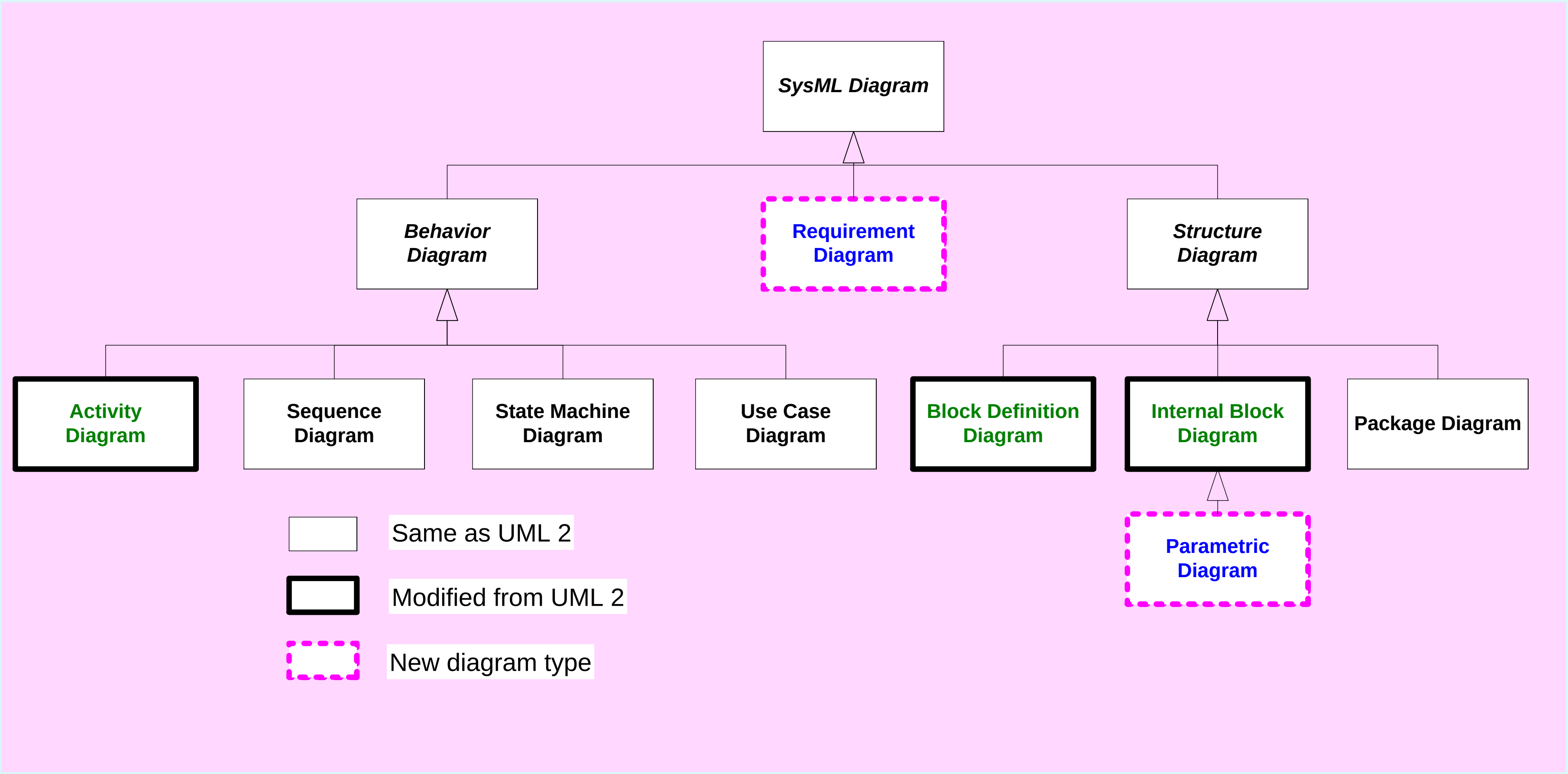
- A graphical modelling language in response to the UML for Systems Engineering RFP developed by the OMG, INCOSE, and AP233
  - a UML Profile that represents a subset of UML 2 with extensions

- Supports the specification, analysis, design, verification, and validation of systems that include hardware, software, data, personnel, procedures, and facilities

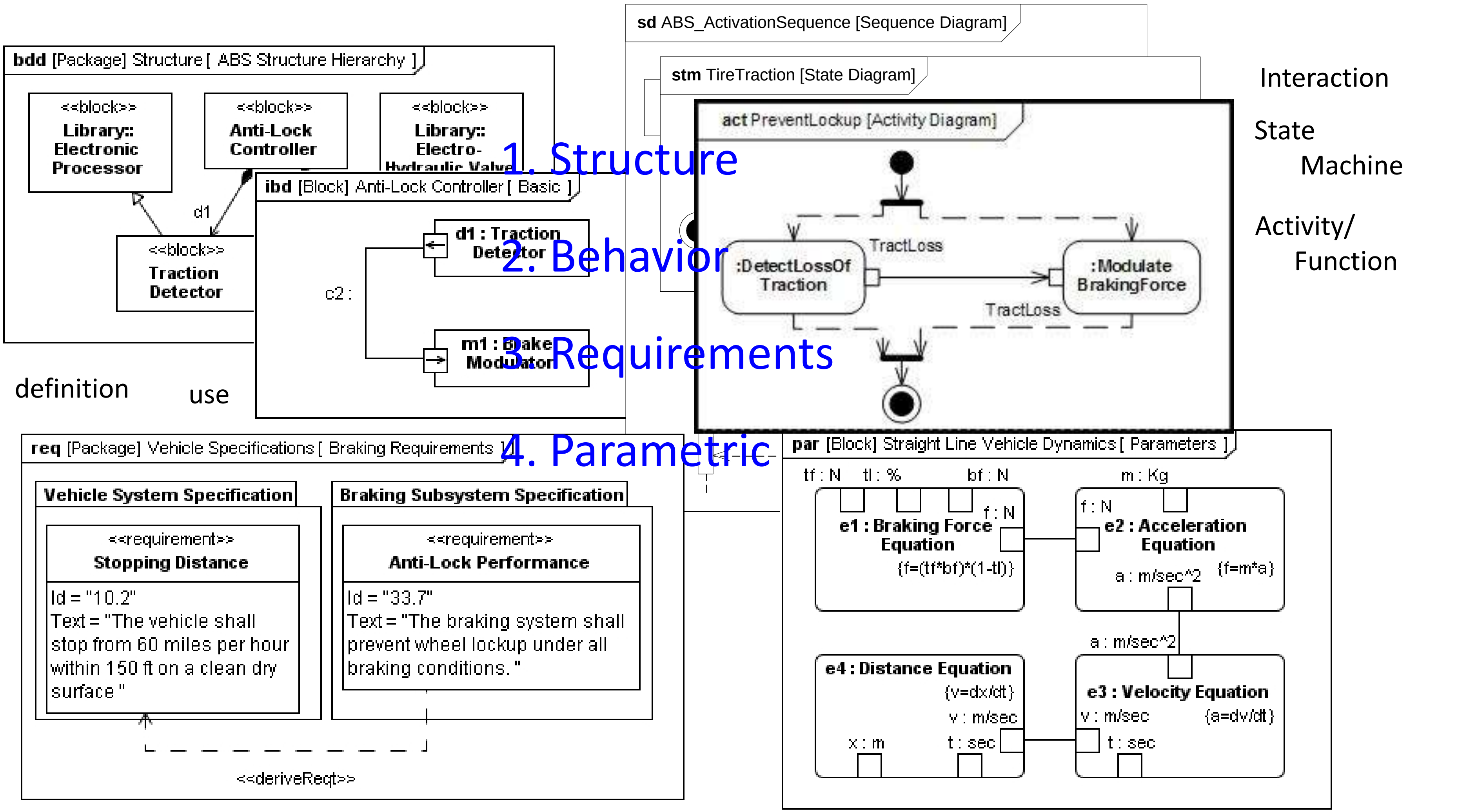
- Supports model and data interchange via XML Metadata Interchange (XMI®) and the evolving AP233 standard (in-process)

SysML is Critical Enabler for Model Driven SE

SysML Diagram Taxonomy

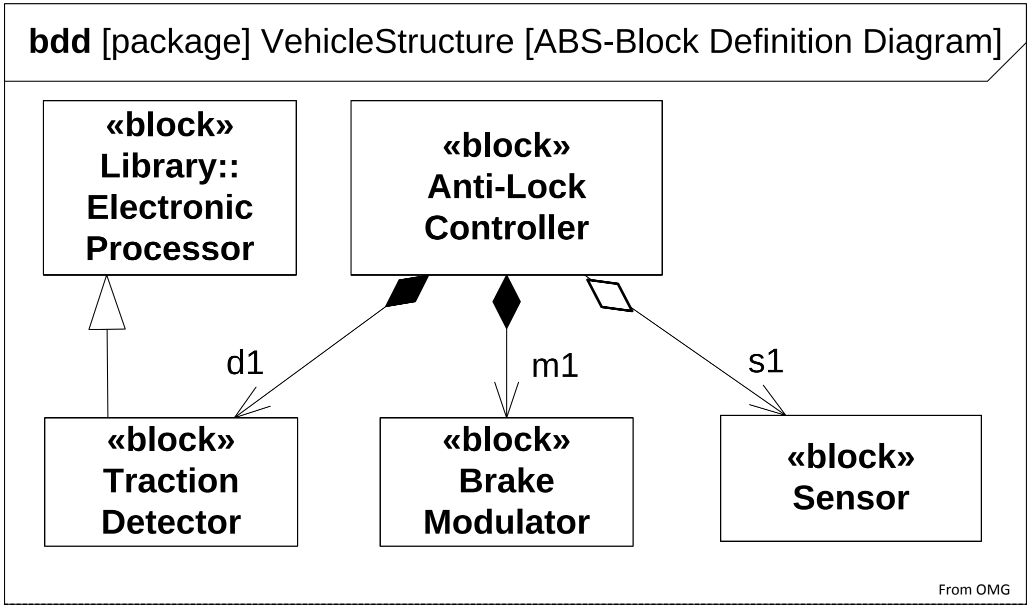


4 Pillars of SysML – ABS Example

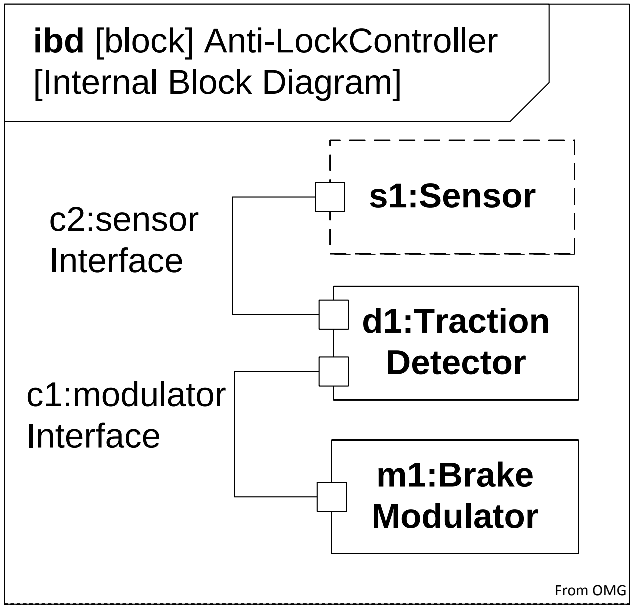


Block Definition and Usage

Block Definition Diagram



Internal Block Diagram



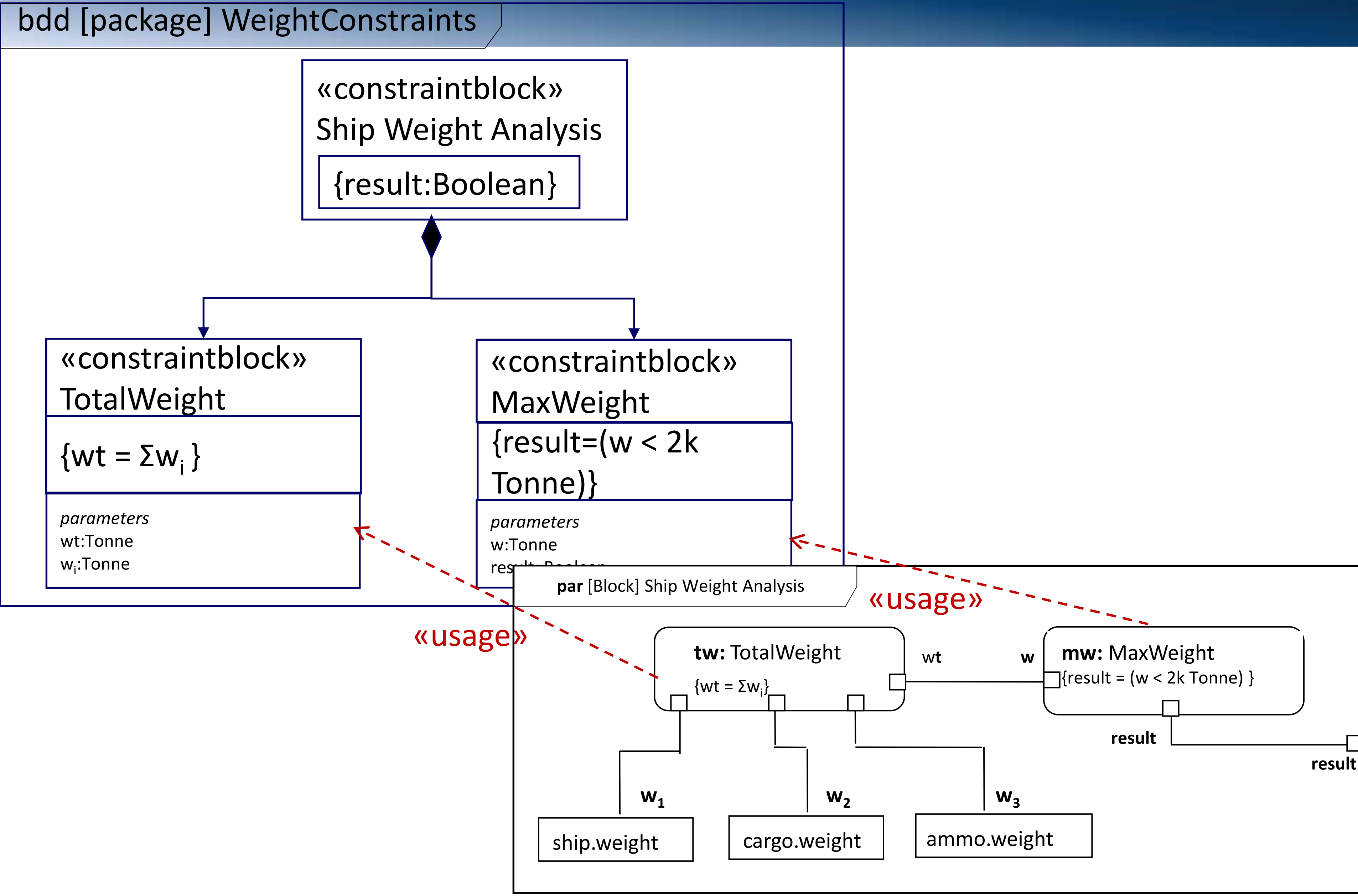
Definition

- Block is a definition/type
- Captures properties, etc.
- Reused in multiple contexts

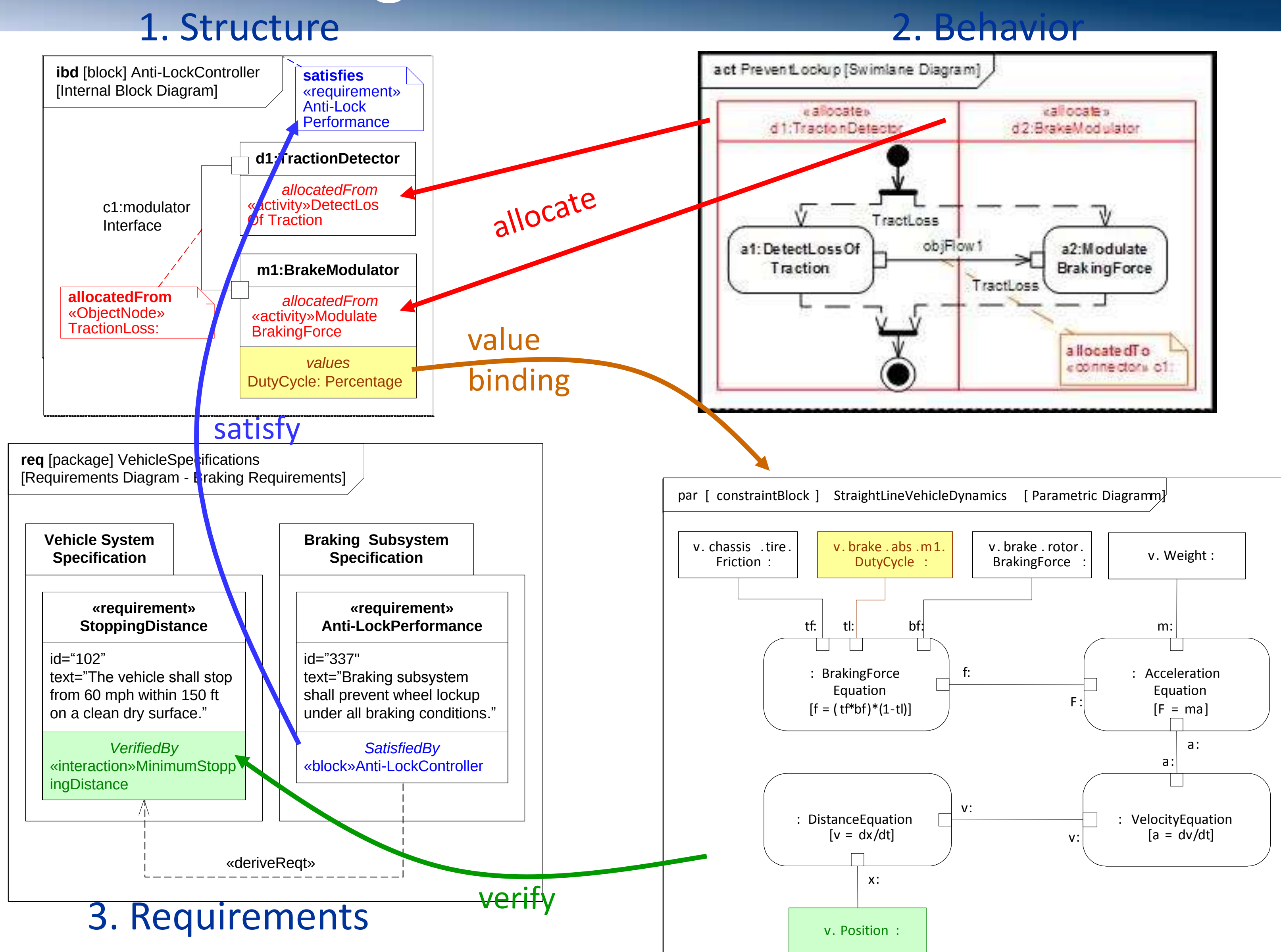
Usage

- Part is the usage in a particular context
- Typed by a block
- Also known as a role

Parametrics -Constraint Definition and Usage



Cross-Connecting Model Elements



Topics

- MBE/MBSE Terminology and Overview
- SysML Overview
- Object Oriented SE Methodology (OOSEM)
- Modeling Tools and the Environment

OOSEM Model-Based SE Methodology

- Traditional top-down Scenario driven SE approach using SysML to facilitate:
  - Capture and analysis of requirements and design information to specify complex systems
  - Integration with OO Software development, hardware development and test processes
  - Flexibility to accommodate changing requirements and design evolution
- SE method that leverages object-oriented (OO) concepts such as:
  - Blocks, value properties and operations
  - Generalizations/Specialization
  - Encapsulation
  - Use cases
- Uses OO concepts however applied differently for SE vs. SW

OOSEM

Typical Systems Req'ts & Design Activities

Major SE Development Activities

1. Setup Project Iteration
- IPT Setup
- Model Setup

2. Analyze Needs
- Causal analysis
- Mission use cases/scenarios
- Enterprise model

3. Define System Requirements
- System use cases/scenarios
- Elaborated context
- Req'ts diagram

4. Define Logical Architecture
- Logical decomposition
- Logical scenarios
- Logical subsystems

5. Synthesize Allocated Architecture
- Node diagram
- HW, SW, Data arch
- System deployment

Common Sub-activities

6. Capture Domain & Assumptions
- Domain Models
- Vocabulary
- Justification Models

7. Optimize & Evaluate Alternatives
- Parametric Diag
- Trade study

8. Reuse Opportunity Analysis
- Reusable Asset Repository
- COTS/GOTS

9. Trace Reqt and Allocations
- Req'ts diagram
- Tables

10. Validate & Verify System
- Test system
- Test cases

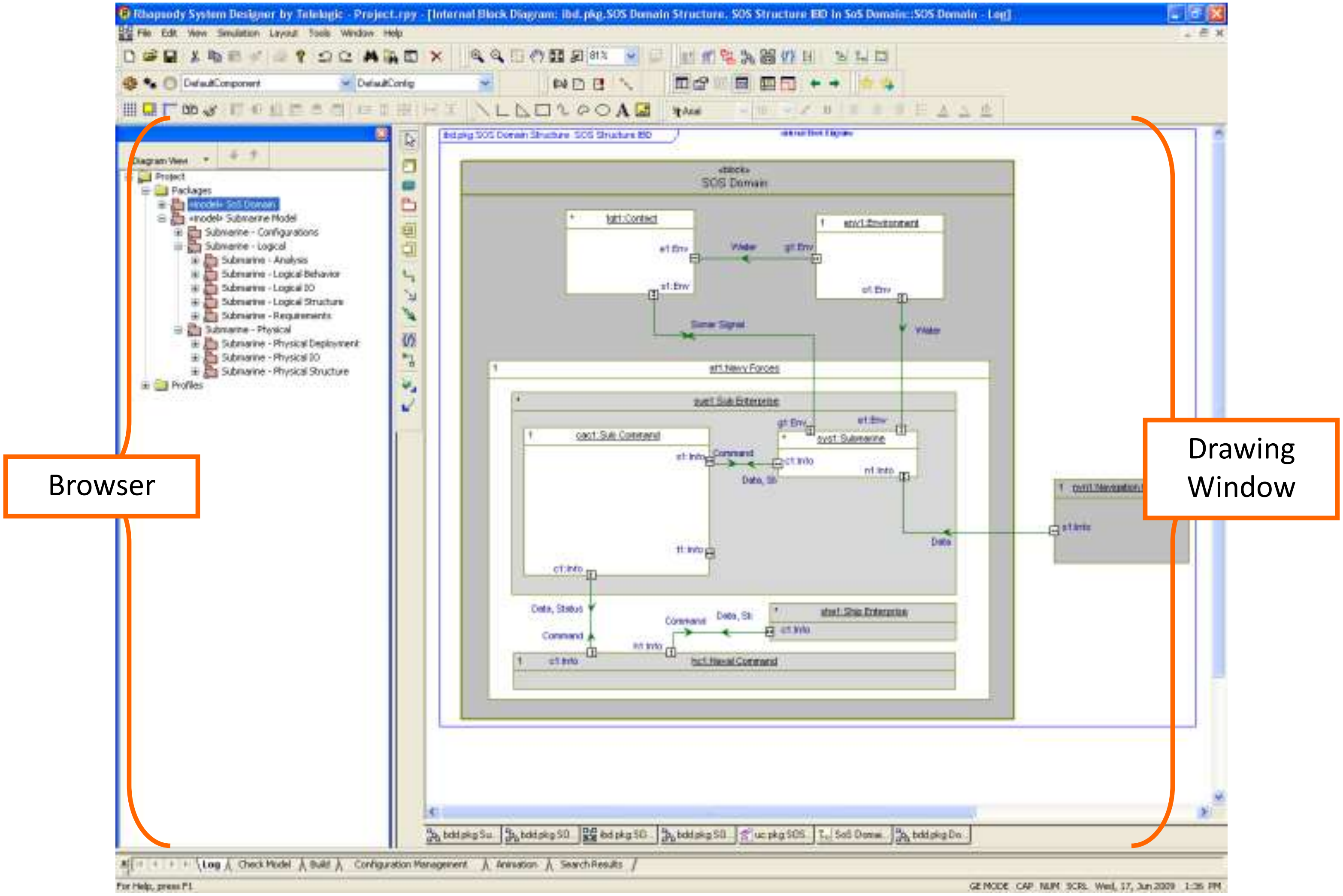
Topics

- MBE/MBSE Terminology and Overview
- SysML Overview
- Object Oriented SE Methodology (OOSEM)
- Modeling Tools and the Environment

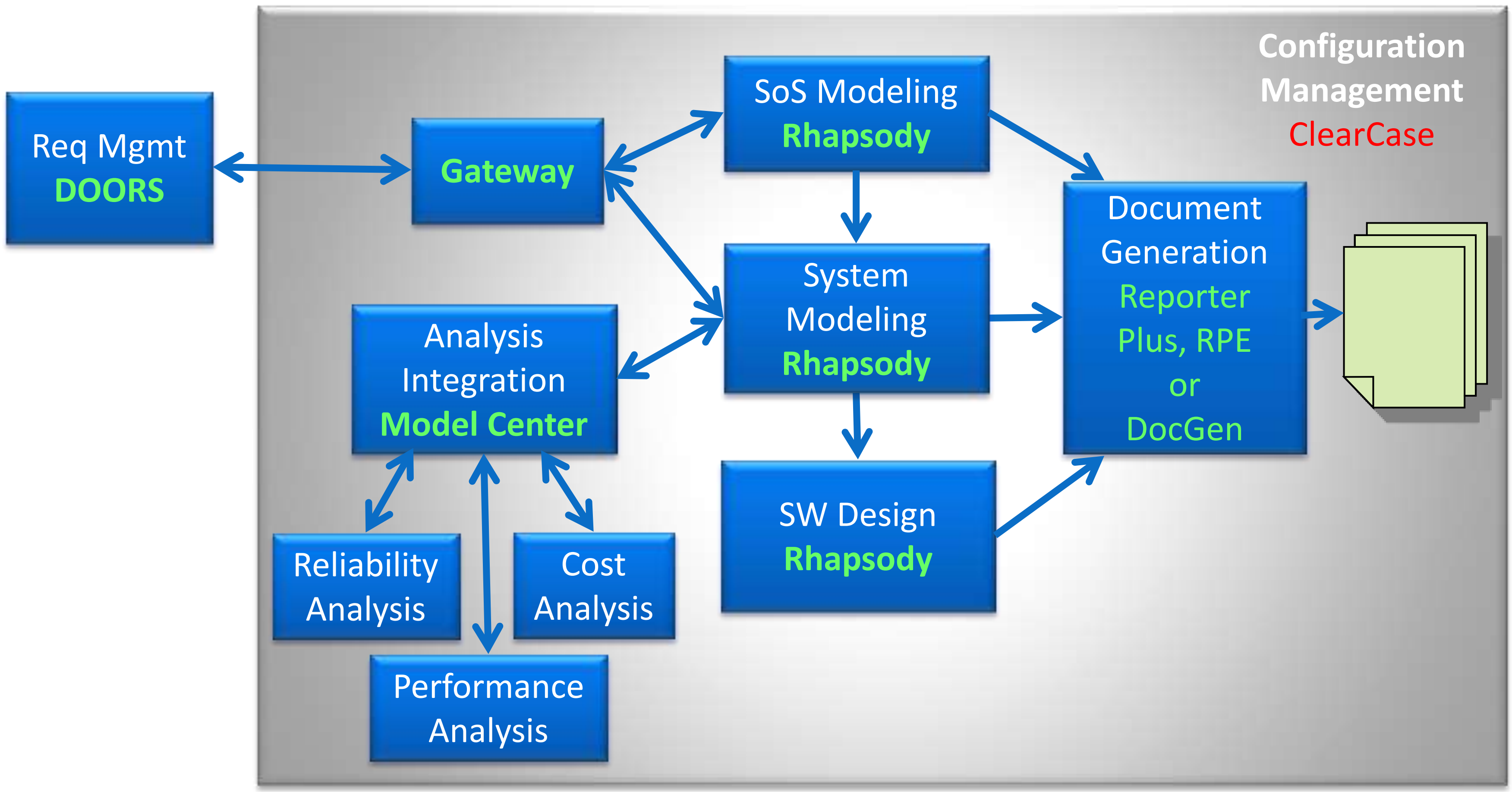
A modeling tool - Rhapsody Example

- The Architecture model in Rhapsody provides dynamic database in which to store system design information
- A given model object, such as a block, maintains it’s characteristics throughout the model, on every diagram in which it appears
- Rhapsody provides a consistent design model that is also tied to requirements
- Rhapsody is one of many UML/SysML tools

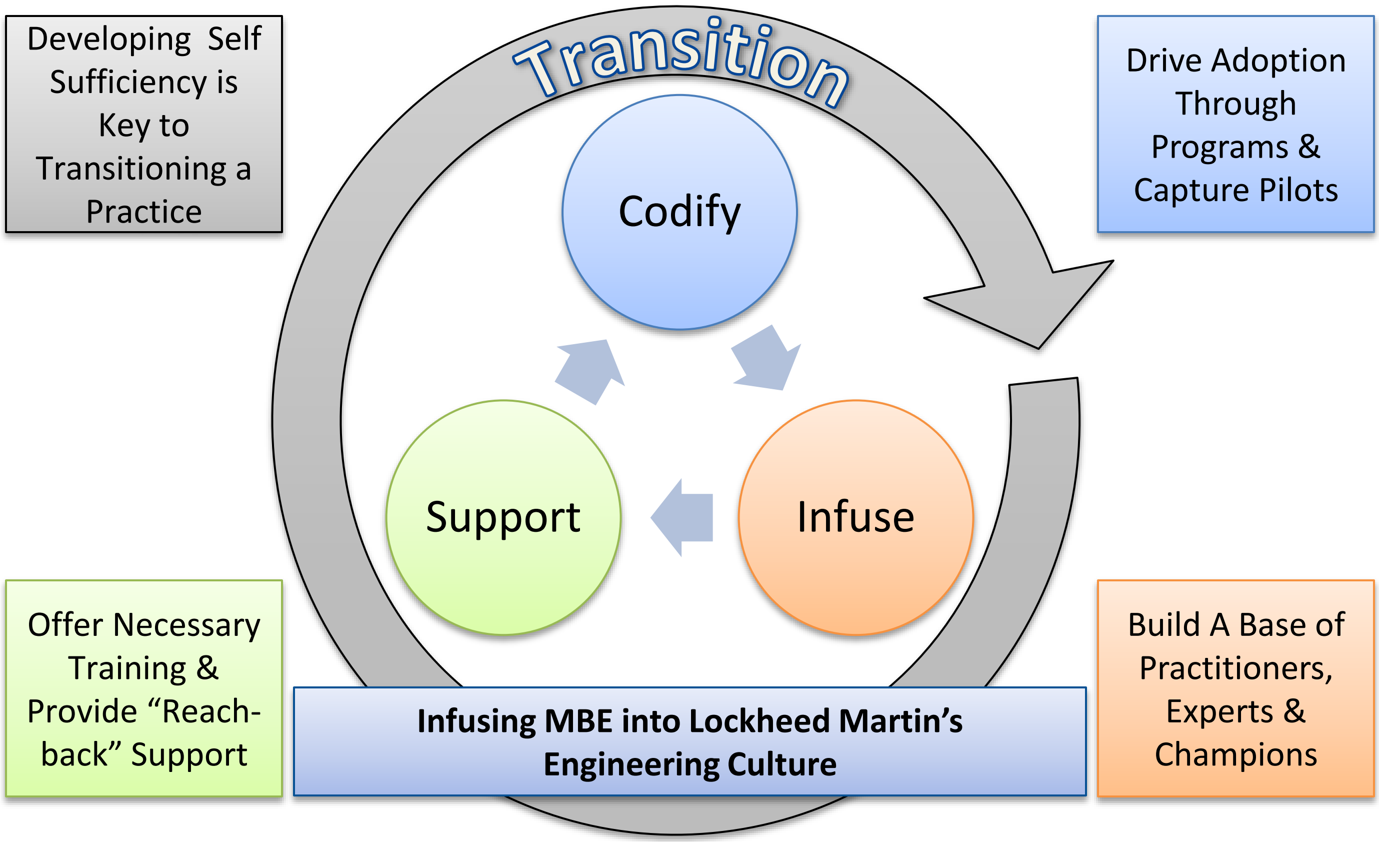
Rhapsody View



Typical Tool Environment



How to Transition & Sustain A Practice



Laura E. Hart
Lockheed Martin, IS&GS
Laura.E.Hart@lmco.com
610-354-6529