Topics

- MBSE Motivation and Scope
- System Modeling with SysML
- System Model as an Integration Framework
- Integrating SysML with Analytical Models
- SysML with UPDM
- INCOSE MBSE Initiative
- Summary
MBSE Motivation and Scope
SE Practices for Describing Systems

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

**Future**

Moving from Document centric to Model centric
Model-based Systems Engineering (MBSE)

- Formalizes the practice of systems development through use of models
- Broad in scope
  - Integrates with multiple modeling domains across life cycle from system of systems to component
- Results in quality/productivity improvements & lower risk
  - Rigor and precision
  - Communications among system/project stakeholders
  - Management of complexity
System Description

• **Document-Based System Engineering:**

Where is truth?

- Document 1
  - A<B
- Document 2
  - A=B
- Document 3
  - A>B

Inconsistencies within and among documents

• **Model-Based System Engineering:**

Model enforces consistency

Model Repository

© Copyright Lockheed Martin Corporation All Rights Reserved
System Modeling Using SysML
System Modeling

Requirements

Functional/Behavioral Model
- Start
- Shift
- Accelerate
- Brake

Performance Model
- Control Input
- Power Equations
- Vehicle Dynamics

System Model

Structural/Component Model
- Engine
- Transmission
- Transaxle

Other Engineering Analysis Models
- Mass Properties
- Structural
- Safety
- Cost Model

Integrated System Model Must Address Multiple Aspects of a System
What is SysML?

- A graphical modeling 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)
4 Pillars of SysML – ABS Example

1. Structure

- **bdd [Package] Structure [ABS Structure Hierarchy]**
  - **<<block>> Library:** Electronic Processor
  - **<<block>>** Traction Detector
  - **<<block>>** Anti-Lock Controller [Basic]
  - **idd [Block] Anti-Lock Controller [Basic]
  - **d1: Traction Detector**
  - **m1: Brake Modulator**

2. Behavior

- **sd ABS_ActivationSequence [Sequence Diagram]**
  - **d1:Traction Detector**
  - **m1:Brake Modulator**
  - **detTrkLos()**
  - **modBrkFrc()**
  - **sendSignal()**
  - **modBrkFrc(traction_signal:boolean)**
  - **sendAck()**

3. Requirements

- **req [Package] Vehicle Specifications [Braking Requirements]**
  - **Vehicle System Specification**
    - **<<requirement>> Stopping Distance**
      - Id = "10.2" Text = "The vehicle shall stop from 60 miles per hour within 153 ft on a clean dry surface."
  - **Braking Subsystem Specification**
    - **<<requirement>> Anti-Lock Performance**
      - Id = "33.7" Text = "The braking system shall prevent wheel lockup under all braking conditions."

- **<<deriveReq>>**

4. Parametrics

- **par [Block] Straight Line Vehicle Dynamics [Parameters]**
  - **e1: Braking Force Equation**
    - \( f = (f + 5f) \) \((1-d)\)
  - **e2: Acceleration Equation**
    - \( a : m/sec^2 \)
    - \( f : N \)
  - **e3: Velocity Equation**
    - \( a = dv/dt \)
    - \( v : m/sec \)
    - \( t : sec \)
  - **e4: Distance Equation**
    - \( v : m/sec \)
    - \( t : sec \)
System Model as an Integration Framework
MBSE Must Integrate across Modeling Domains

- Ops/Mission Analysis
- Logistics Support
- Manufacturing
- Integration & Test
- Performance Simulation
- Engineering Analysis
- System Design
- Algorithm Development
- Software Design
- Hardware Design
- Human System Integration

© Copyright Lockheed Martin Corporation All Rights Reserved
Using System Architecture Model as an Integration Framework
Using the System Architecture Model to Flowdown Requirements

**System-of-System Level**
- 1st Level of Decompositions
- How our system contributes to the overall mission

**System Level**
- Derives Subsystems
- Allocates Requirements to Subsystems

**Element Level**
- Derives Hardware and Software Components
- Allocates Requirements to Components

**Component Design & Implementation Level**

(From John Watson/LMC SysML Info Days presentation)

**Trade Studies, Simulation, Specification Reviews, etc.**
System Decomposition Process using SysML

1. Analyze System Level Requirements
2. Analyze System Services
3. Identify the Subsystem
4. Analyze Subsystem Collaboration to Satisfy the System Services
5. Incorporate Additional Analysis as Needed
6. Derive and Allocate Requirements to Subsystem

Decision: Continue?
- Yes
  - Continue
- No
  - Complete Subsystem Specs

Trade Studies, R&D, Simulation, Specification Reviews, etc.

The Subsystem shall ....
Derived Requirements

(from John Watson/LMC
SysML Info Days presentation)
Integrating SysML with Analytical Models
System Architecture Model to Support Tradeoff Analysis

### Analysis Results

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Alternative1</th>
<th>Alternative2</th>
<th>Alternative3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor</td>
<td>Sensor1</td>
<td>Sensor2</td>
<td>Sensor3</td>
</tr>
<tr>
<td>Processor</td>
<td>Processor1</td>
<td>Processor2</td>
<td>Processor3</td>
</tr>
<tr>
<td>Control</td>
<td>Control1</td>
<td>Control2</td>
<td>Control3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>Alt 1</th>
<th>Alt 2</th>
<th>Alt 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>0.5</td>
<td>7</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Reliability</td>
<td>0.2</td>
<td>4</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>Cost</td>
<td>0.3</td>
<td>3</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Effectiveness</td>
<td></td>
<td>5.2</td>
<td>4.2</td>
<td>5.9</td>
</tr>
</tbody>
</table>

### Optimization

\[ E = \text{Sum} \left[w_1 u_1(P) + w_2 u_2(P) + w_3 u_3(C)\right] \]

© Copyright Lockheed Martin Corporation All Rights Reserved
SysML-Modelica Transformation
Context & Objective

- Two complementary languages for Systems Engineering
  - Descriptive modeling in SysML
  - Formal equation-based modeling for analyses and trade studies in Modelica

- Objective:
  - Leverage the strengths of both SysML and Modelica by integrating them to create a more expressive and formal MBSE language
  - Define a formal Transformation Specification
    - a SysML4Modelica profile
    - a Modelica abstract syntax metamodel
    - a mapping between Modelica and the profile

Presentation by Chris Paredis for the INCOSE Symposium 2010 Chicago, IL
SysML-Modelica Robot Example: SysML Model

Presentation by Chris Paredis for the INCOSE Symposium 2010 Chicago, IL
SysML-Modelica Robot Example: Modelica model with simulation results

Presentation by Chris Paredis for the INCOSE Symposium 2010 Chicago, IL
SysML with UPDM
UPDM Summary

- UPDM 1.0 is a standardized way of expressing DoDAF 1.5 and MODAF 1.2 artefacts using UML and SysML
  - UPDM is **NOT** a new Architectural Framework
  - UPDM is not a methodology or a process
  - UPDM 2.0 is scheduled to address DoDAF 2.0, MODAF 1.2, NAF 3.x, and DNDAF 1.7

- UPDM 1.0 was developed by members of the OMG with help from industry and government domain experts

- UPDM 1.0 has been implemented by multiple tool vendors

Presentation by M. Hause to the SE DSIG at the OMG Minneapolis meeting in June 2010
UPDM RFC - Domain Meta Model Summary
(Packages)

Presentation by M. Hause to the SE DSIG at the OMG Minneapolis meeting in June 2010

- Package structure organizes stereotypes by viewpoint
- Multiple viewpoints manage model complexity
SysML Extensions with UPDM level 1

- Facilitates integration of DoDAF and MODAF models for system of systems modeling with SysML models for systems modeling

- Enables UPDM to fully leverage SysML features
  - SysML blocks enable structural modeling with flow ports, item flows, and value properties with units and distributions
  - SysML activities support continuous flow modeling, activity hierarchies, and enhanced functional flow block diagrams
  - SysML parametrics enable the integration of engineering analysis with the architecture models
  - SysML allocations support various types of mappings such as an SV-5
  - SysML requirements enable text based requirements to be captured and traced
  - SysML view and viewpoint provide for multiple perspectives of the model
OV-5 Activity Diagram

Presentation by M. Hause to the SE DSIG at the OMG Minneapolis meeting in June 2010
INCOSE MBSE Initiative
Automatic Cruise Control

Thermal/Heat Dissipation: 780°

Ergonomic/Pedal Feedback: 34 ERGS

Hydraulic Pressure: 350 PSI

Sensor MTBF: 3000 hrs

Power Rating: 18 Amps

Hydraulic Fluid: SAE 1340 not-compliant

Minimum Turn Radius: 24 ft.

Dry Pavement Braking Distance at 60 MPH: 110 ft. 90 ft

Provided by Mark Sampson
Extending Maturity and Capability

- Defined MBSE theory, ontology, and formalisms
- Architecture model integrated with Simulation, Analysis, and Visualization
- Matured MBSE methods and metrics, Integrated System/ HW/ SW models
- Emerging MBSE standards
- Distributed & secure model repositories crossing multiple domains

Refer to activities in the following areas:

- Planning & Support
- Research
- Standards Development
- Processes, Practices, & Methods
- Tools & Technology Enhancements
- Outreach, Training & Education

INCOSE MBSE Roadmap

MBSE Capability

- Reduced cycle times
- System of systems interoperability
- Design optimization across broad trade space
- Cross domain effects based analysis

Maturity

- Well Defined MBSE
- Institutionalized MBSE across Academia/ Industry
- Ad Hoc MBSE
- Document Centric
INCOSE MBSE Organization

- **Leadership**
  - Chair - Mark Sampson
  - CoChair - Sanford Friedenthal
  - Communication Lead - Ray Jorgensen

- **Activity Leads**
  - Methodology and Metrics – Jeff Estefan
  - MBSE Usability - Scott Workinger
  - Modeling Standards - Roger Burkhart
  - Model Management - Mark Sampson
  - Ontology - Henson Graves
  - SoS/Enterprise Modeling – Ron Williamson

- **Challenge Team Leads**
  - GEOSS Modeling – Larry McGovern
  - Modeling & Simulation Interoperability – Russell Peak
  - Space Systems Modeling – Chris Delp
  - Telescope Modeling – Robert Karban
MBSE Current State and Future Directions

- Current State of Practice
  - Early adoption and emerging practice
  - Similar to early stage of MCAD/ECAD

- Future Needs that MBSE Must Support
  - Applications in emerging areas (e.g. healthcare, energy, environment, systems biology)
  - Systems that are more complex, adaptive, and chaotic
  - Ability to more rapidly explore broad trade space and perform multi-variable and stochastic design optimization

- MBSE Directions
  - Extend theory and formalisms (e.g., Wymore)
  - Reusable model libraries, taxonomies and patterns
  - Integration across modeling domains in a highly distributed environment
  - Supporting languages, methods, and tools based on standards
Summary

- MBSE is a key practice to advance complex systems development
- Standards such as SysML, UPDM, and Modelica are critical enablers of MBSE
- Multiple tool vendors implementing SysML
- System architecture model and standards based approach facilitate integration across modeling domains
- Growing interest and application of MBSE
- INCOSE MBSE Initiative Providing a Forum for Advancing the Practice