# SYSML 2.0 UPDATE

Hedley Apperly
VP Solution Management

May 2015

PTC®

# INTRODUCTION

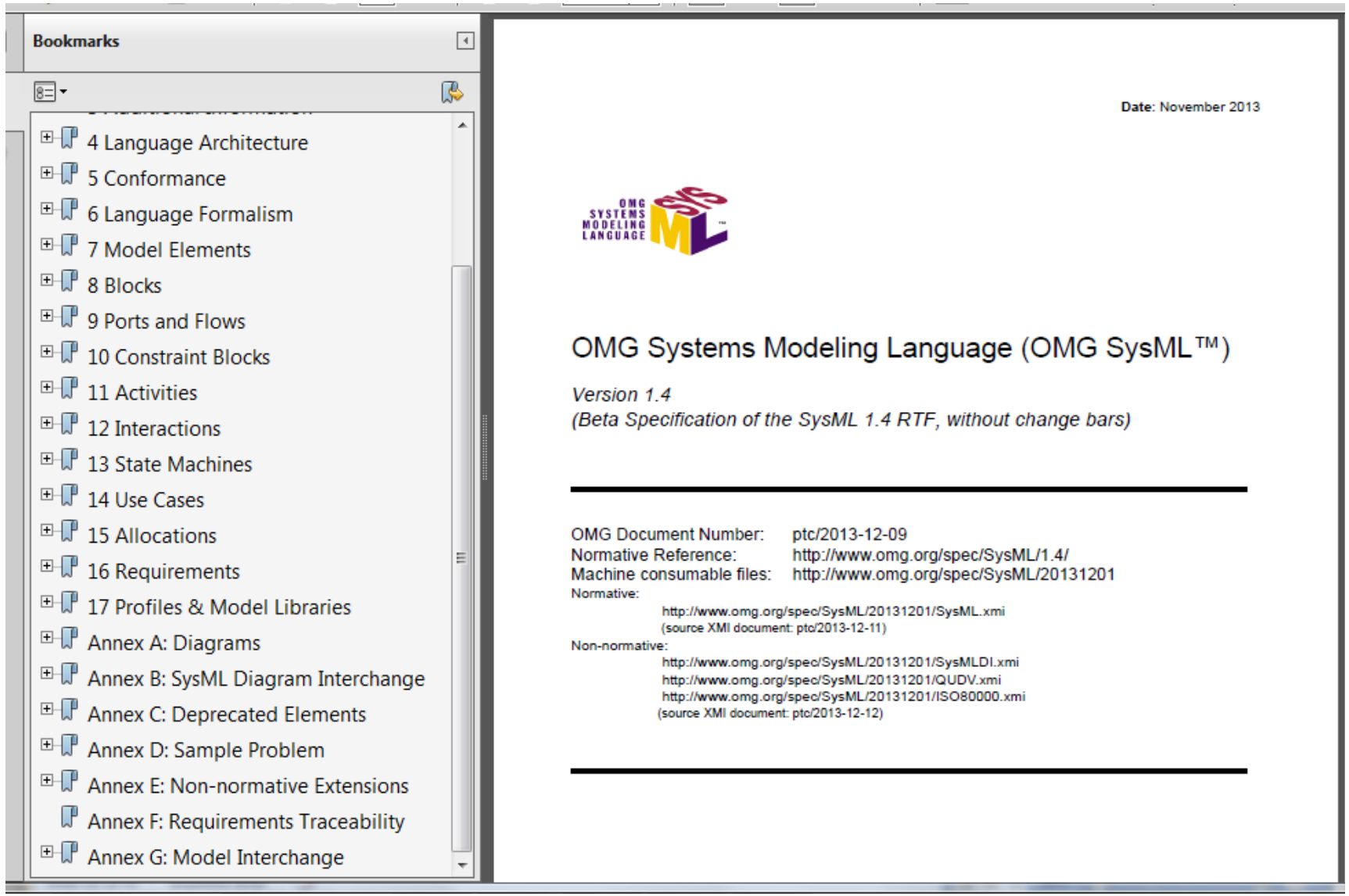
# WHAT IS MBSE

"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), Sept 2007

**MBSE is SE**

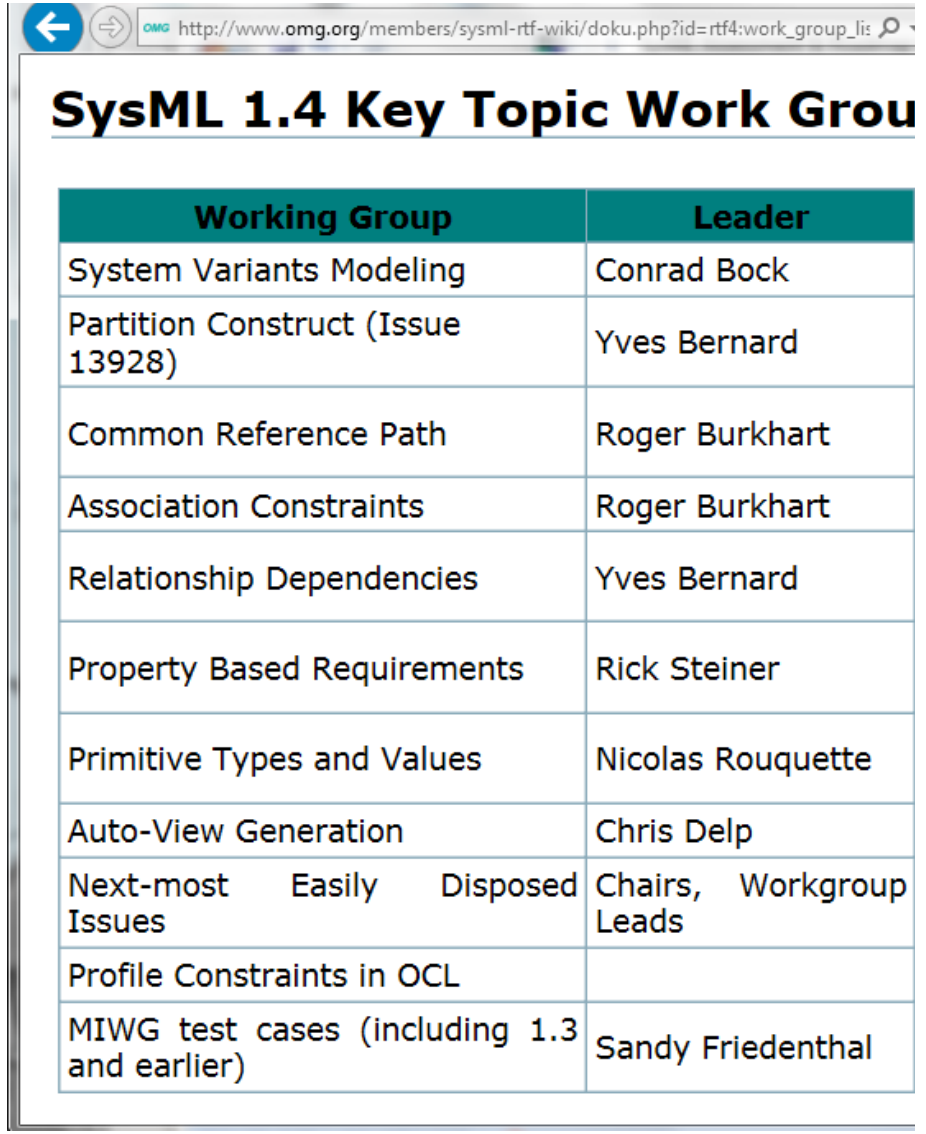
# STARTING POINT – SYSML 1.4



# WHY DO WE NEED SYSML 2.0

- SysML 1.x Problems
- Evolving and Improving MBSE Processes
- Identified Needs
  - Rich diagrammatic syntax with standard symbol libraries for domain specific applications (e.g. Visio libraries)
  - Extensive viewing capability to query the model and present the results. (e.g., similar to building architecture layers)
  - Extensive modeling checking and analysis capability to reason about the system model and confirm its integrity
  - Extensive reuse libraries
  - and more…

OMG SE DSIG 2011




| Working Group | Leader |
|---|---|
| System Variants Modeling | Conrad Bock |
| Partition Construct (Issue 13928) | Yves Bernard |
| Common Reference Path | Roger Burkhart |
| Association Constraints | Roger Burkhart |
| Relationship Dependencies | Yves Bernard |
| Property Based Requirements | Rick Steiner |
| Primitive Types and Values | Nicolas Rouquette |
| Auto-View Generation | Chris Delp |
| Next-most Easily Disposed Issues | Chairs, Workgroup Leads |
| Profile Constraints in OCL | |
| MIWG test cases (including 1.3 and earlier) | Sandy Friedenthal |

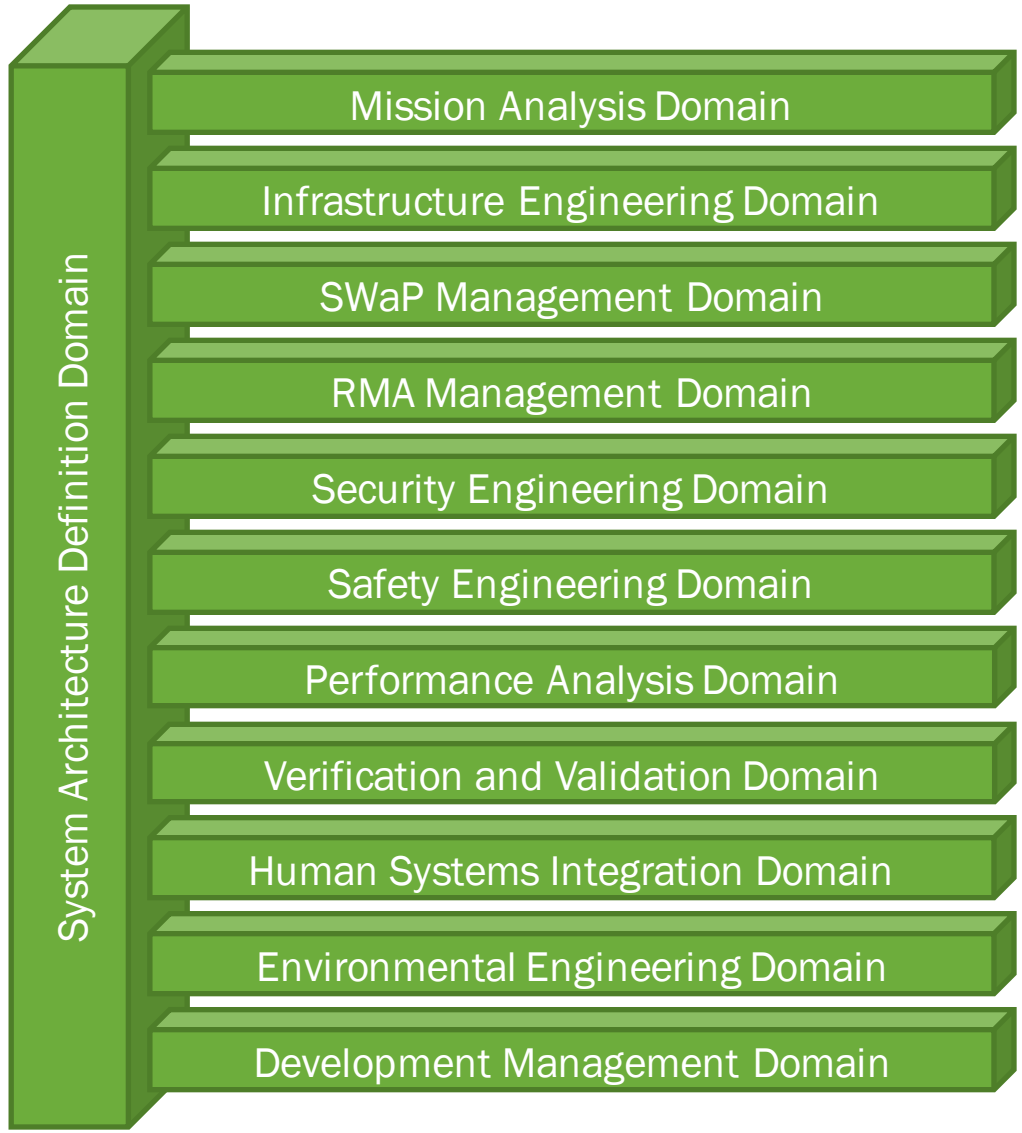
# MBSE ADOPTION ISSUES

- More focus on mechanical engineering
- Provide more examples/guidance
- Availability of libraries of reusable models
- Availability of patterns
- Language stability
- Increased analysis capabilities
- A clear value assessment from using SysML
- Model consistency
- Domain specific icons
- Support for continuum of models that support early concepts and more detailed formal models

OMG 2016

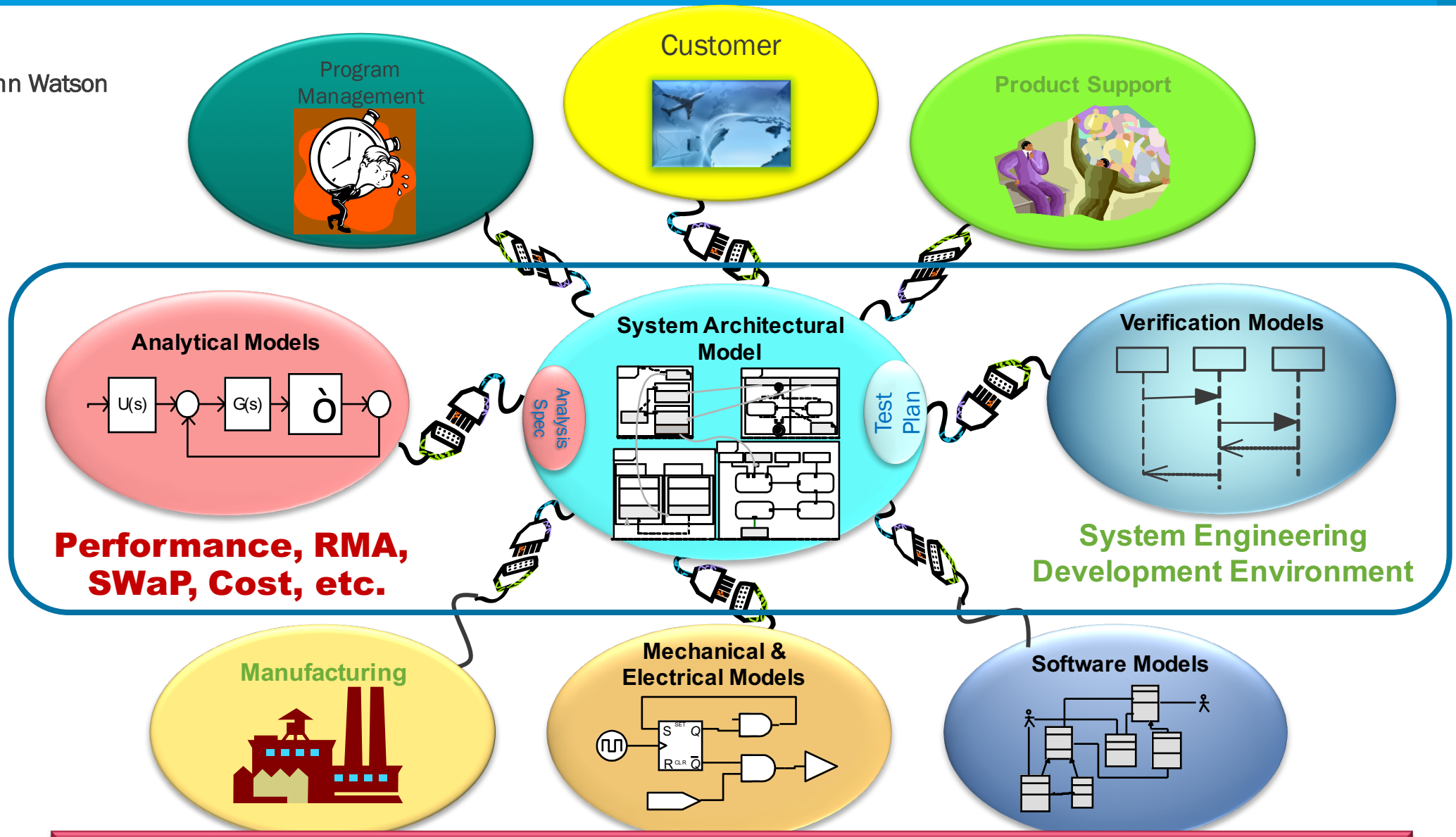
# NEW INPUTS

- **e.g. INCOSE SE BoK**
  - Each SE Domain;
    - Contributes domain specific information
    - Has Responsibility for their information content
    - Information content can include content from other domains
    - Iterates solution with other Domains
    - Has one or more views to information content
    - Defines and manages Requirements
    - Measures impact of their changes in their domain and across other domains
    - Conducts Reviews
    - Produces Deliverables



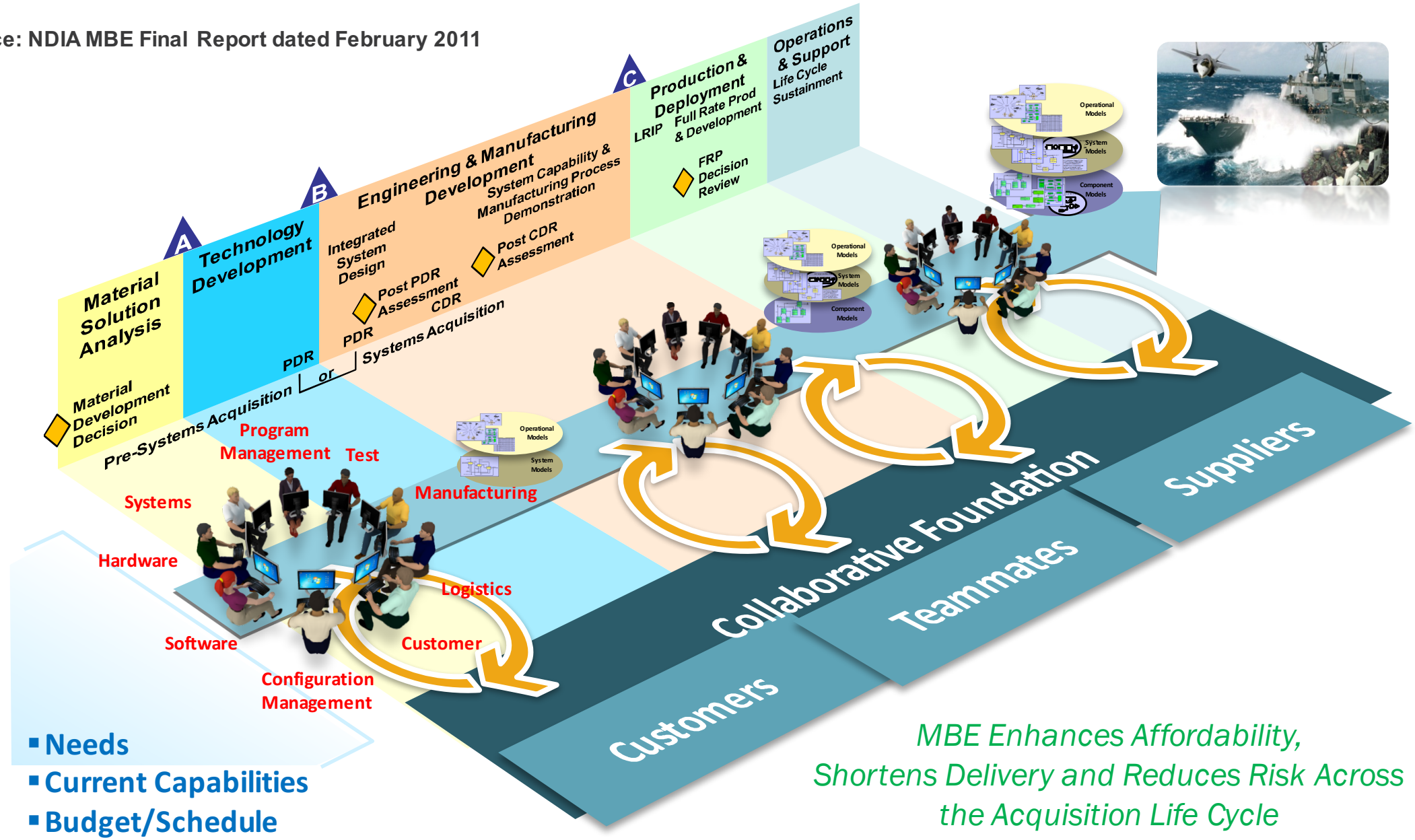
# EVOLVING MBSE USE CASES

Source; John Watson



# MBE TO BE STATE

Source: NDIA MBE Final Report dated February 2011




*MBE Enhances Affordability, Shortens Delivery and Reduces Risk Across the Acquisition Life Cycle*

# SE VISION 2020

- Domain-specific modeling languages and visualization that enable the systems engineer to focus on modeling of the user domain
- Modeling standards based on a firm mathematical foundation that support high fidelity simulation and real-world representations
- Extensive reuse of model libraries, taxonomies and design patterns
- Standards that support integration and management across a distributed model repository
- Highly reliable and secure data exchange via published interfaces

**INCOSE-TP-2004-004-02 September, 2007**

# OMG SYSTEM MODELING ASSESSMENT AND ROADMAP WORKING GROUP

# OMG SYSML 2.0 WORKING GROUP

- **System Modeling Assessment and Roadmap Working Group**
  - Lead by;
    - Sandy Friedenthal with
    - OMG members from INCOSE, industry, vendors and academia;

      Eldad Palachi, Yves Bernard, Manas Bajaj, Laura Hart, John Watson, Chris Delp, Rick Steiner, Roger Burkhart, Hedley Apperly, Uwe Kauffmann, Nerijus Jankevicius, Ron Williamson, Chris Schreiber, Josh Feingold, Marc Sarrel, et al
  - Objectives;
    - Assess effectiveness of system modeling with SysML in support of MBSE Adoption and Use
    - Develop a preliminary System Modeling Roadmap to improve effectiveness
    - Use the Roadmap to influence the SysML specification, tool vendor implementations, related standards efforts, and industry collaborations
    - Develop the concept for the next generation System Modeling Environment (SME) and derive the requirements for SysML v2
    - Current Status
    - Preparing Request for Proposal (RFP)
      - See following slides for OMG process description

# OMG PROCESS (1 - SIMPLIFIED)

- **RFI (Request for Information)**
  - This is Optional and in this case we will not issue an RFI
  - The Working Group will scope the Requirements for the RFP
    - Preparations have been made over the last 18 months, with input from INCOSE and others

- **RFP (Request for Proposal)**
  - Working Group (or Task Force (TF)) will issue an RFP (planned for H2 2016)
  - Before vote to accept and publish RFP at least one member company must submit an LOI (Letter of Intent) to Submit Proposals
  - Internal OMG votes on the RFP and sets Submission Deadline
  - Member companies submit Proposal(s)
  - Interested OMG members read the submissions, and comment on them
  - Revised submission deadline may be set and re-revised
  - OMG members read and evaluate the submissions and revised submissions
  - If most members consider the submission worthy, a series of votes begins

# OMG PROCESS (2 - SIMPLIFIED)

- **Voting to Adopt**
  - The Task Force votes to recommend adoption to its parent Technical Committee (TC)
  - The Architecture Board (AB) votes approval
  - The parent TC votes to recommend to OMG's Board of Directors (BOD)
  - The BOD Business Subcommittee (BSC) reports to the BOD on Business Committee Questionnaire responses from the submitters
  - If at least one satisfactory response was received, the BOD votes to adopt the specification. At this point, the submission becomes an official OMG Adopted Specification, but does not receive a release number

# OMG PROCESS (3 - SIMPLIFIED)

- **Finalization**
  - The TC charters a Finalization Task Force (FTF)
  - The FTF performs the first maintenance revision on the specification
    - Resolving issues submitted to OMG
  - The FTF-revised version of the specification is adopted as official OMG technology, through the same series of votes as the original submission (TF, AB, TC, and BOD).
  - This time it receives a release number, and is designated an available specification
  - The document is edited into a formal OMG specification (planned for 2018)
  - Typically, products reach the market around this time too

- **Revision**
  - A recurring maintenance cycle starts here. The TC charters a Revision Task Force ( RTF) and sets deadlines for its report and specification revision
  - The RTF collects and acts on issues submitted to OMG, producing a revised specification
  - The revised specification is adopted through the series of four votes
  - A new RTF is chartered, and the process repeats

# SYSML 2.0 SCOPE & REQUIREMENTS

# ‘TAKING A FRESH LOOK’

# SCOPE OF THE SYSTEMS MODELING ENVIRONMENT (SME)

- SysML modeling language and tools
- Modeling languages and tools that support use of SysML (e.g. constraint language, transformations)
- Reuse libraries (e.g., models, practices, ..)
- Integrations with other engineering models and tools
- Extension and customization facilities

# MBSE USE CASES

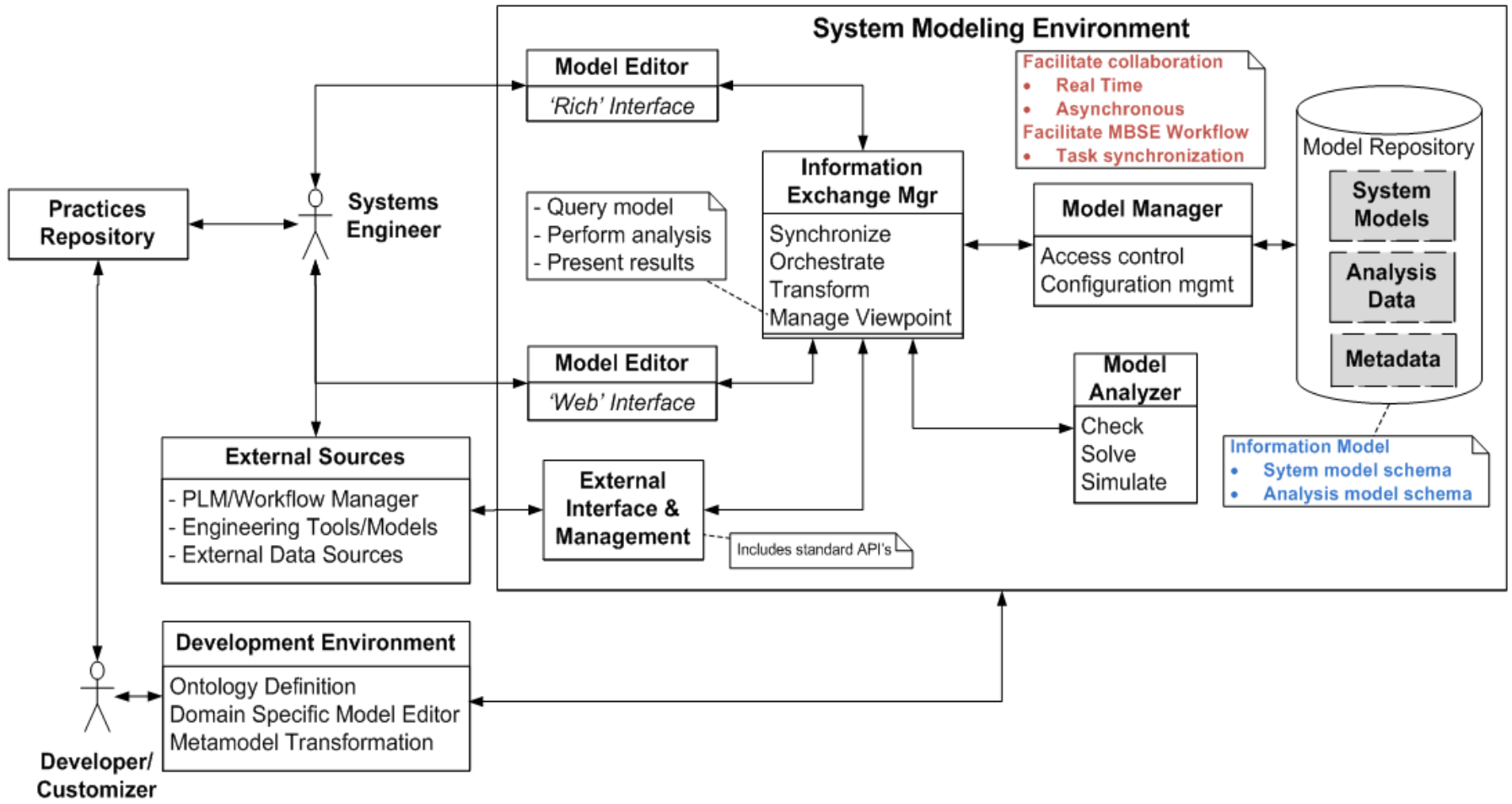
- **SysML RTF RFI Survey – 2009**
  - Open to any practitioner and well publicized
  - 2 parts; SysML and how well SysML supports MBSE
  - Example Result; Block Definition Diagrams (BDD's) and Internal Block Diagrams (IBD's): used the most, valued the most, hardest for stakeholders to understand. Confusion on the use of ports and interfaces
- **Working Group is incrementally developing MBSE use cases based on accepted SE Processes**
  - refer to SEBoK and ISO 15288 as top level framework
- **Then assess how well SysML supports each use case**

# CAPABILITIES OF THE SYSTEMS MODELING ENVIRONMENT (SME)

- Model construction
- Model visualization
- Model analysis
- Model management
- Model exchange and integration
- Support for MBSE collaboration and workflow

# SYSML 2.0 SME

- Defines Logical Tool Architecture
- No longer just a Modeling Language
- Separates Model Visualization from Model Entry & Editing
- Includes Practices & Practice Repository
- Distinct Model Repository & Meta-model



# KEY EFFECTIVENESS MEASURES

- Expressive: Ability to express the system concepts
- Precise: Representation is unambiguous and concise
- Presentation/communication: Ability to effectively communicate with diverse stakeholders
- Model construction: Ability to efficiently and intuitively construct models
- Interoperable: Ability to exchange and transform data with other models and structured data
- Manageable: Ability to efficiently manage change to models
- Usable: Ability for stakeholders to efficiently and intuitively create, maintain, and use the model
- Adaptable/Customizable: Ability to extend models to support domain-specific concepts and terminology

# KEY REQUIREMENTS (1)

1. Must express the core systems engineering concepts
2. Must include precise semantics that avoid ambiguity and enable a concise representation of the concepts
3. Must provide flexible and rich visualization and reporting capabilities to support a broad range of model users
4. Must enable much more intuitive and efficient model construction
5. Must support MBSE in the broader context of Model-Based Engineering (MBE), where the models and tools are fully integrated across discipline-specific engineering tools, including hardware and software design, analysis and simulation, and verification
6. Must provide a standard application programming interface (API) to provide dynamic access to the model, while providing appropriate access controls
7. Must be capable of being managed in a heterogeneous and distributed modeling environment. The ability to integrate with Product Lifecycle Management (PLM) environments, which enable versioning, configuration, and variant management, is a fundamental SME requirement
8. Must enable efficient and intuitive use by a broad range of users with diverse skills

# KEY REQUIREMENTS (2)

9. Must be highly adaptable and customizable to multiple application domains
10. M must support the migration of existing models with minimum information loss protect investments made by organizations. Models must also be capable of being stored in neutral formats that can be retained for future access.
11. Must enable evolution of the above capabilities to take advantage of on-going advances in technologies, concepts, methods, and theories. Its architecture must be modular and extensible and support the definition of conformance levels that satisfy requirements of different applications.

# 10 SUB - WORKING GROUPS

1. Systems Engineering Workflow Use Cases Working Group
2. Systems Engineering Concept Model Workgroup
3. System Analysis Workgroup
4. SysML v2 Model Formalism Working Group
5. Systems Engineering Model Construction Focus Area
6. Model Visualization Working Group
7. Model Lifecycle Management Working Group
8. Systems Engineering Interoperability Working Group
9. MBSE Workflow and Collaboration Working Group
10. Requirements Modeling Working Group

# 1 SYSTEMS ENGINEERING WORKFLOW USE CASES WORKING GROUP

- **John Watson et al**
- Objective; Capture Systems Engineering use cases to identify the capabilities and infrastructure required by Systems Engineering Roles to develop systems across the product lifecycle
- Foundation Material;

1. INCOSE-TP-2003-002-03.2.2, ***INCOSE Systems Engineering Handbook*** v. 3.2.2, October 2011
2. Pyster, A. and D.H. Olwell (eds). 2013. ***The Guide to the Systems Engineering Body of Knowledge (SEBoK),*** v. 1.2. Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed DATE. www.sebokwiki.org/
3. International Standard – **ISO/IEC 15288 and IEEE 15288** – 2008, Second Edition 2008-02-01, Systems and software engineering - System life cycle processes

Found 31 elements

| Use Case Category | Use Case Name |
|---|---|
| Exploratory and Concept Stage | Analyze Stakeholders Needs |
| Exploratory and Concept Stage | Derive System Requirements |
| Exploratory and Concept Stage | Analyze System Life-cycle Costs |
| Exploratory and Concept Stage | Response to a Customer Request |
| Management Workflow Use Cases | Manage SE Development Progress |
| Management Workflow Use Cases | Manage SE Development Environment |
| Management Workflow Use Cases | Create a Baseline |
| Management Workflow Use Cases | Plan a Development Cycle |
| Product and Service Life Management Stage | Support System Modernization Plan |
| Product and Service Life Management Stage | Evaluate Change Request |
| Product and Service Life Management Stage | Architect Sustainability System |
| Product and Service Life Management Stage | Support Initial Installation |
| Product and Service Life Management Stage | Support System Disposal and Retirement |
| Production Stage | Support Produceability Engineering |
| SE Domain Workflow Use Cases | Perform a Trade Study |
| SE Domain Workflow Use Cases | Analyze System Performance |
| SE Domain Workflow Use Cases | Manage Product Lines |
| SE Domain Workflow Use Cases | Integrate Human Domain Constraints |
| SE Domain Workflow Use Cases | Perform Environmental Engineering |
| SE Domain Workflow Use Cases | Collaborate with Implementation Domain Team |
| SE Domain Workflow Use Cases | Perform EMI Engineering |
| SE Domain Workflow Use Cases | Derive Product Architecture |
| SE Domain Workflow Use Cases | Analyze Behavior Correctness |
| SE Domain Workflow Use Cases | Evaluate System Safety |
| SE Domain Workflow Use Cases | Perform System RMA Engineering |
| SE Domain Workflow Use Cases | Allocate and Manage SWaP |
| SE Domain Workflow Use Cases | Apply System Security Engineering |
| Validation and Verification Workflow Use Cases | Execute a Verification Test Procedure |
| Validation and Verification Workflow Use Cases | Develop Verification Plan and Procedures |
| Validation and Verification Workflow Use Cases | Develop a System Integration Plan |
| Validation and Verification Workflow Use Cases | Provide V&V Status |

# 2 SYSTEMS CONCEPT MODEL (SECM) WORKING GROUP

PTC

- **John Watson et al**
- Objective; capture Systems Engineering domain concepts based on well established industry references
- Systems engineers and other discipline engineers contribute to the development and maintenance of the system model throughout the lifecycle to support system specification, design, analysis, and verification activities

| # | Name | Documentation |
|---|---|---|
| 1 | Action | A non-interruptible function. Note: An action represents an atomic unit of processing or work. Actions may be continuous or discrete. Discrete actions may or may not be assumed to execute in zero time. |
| 2 | Activation/deactivation event | An event that occurs when a function is activated or deactivated. |
| 3 | Activation/deactivation requirement | The activation or deactivation that one or more functions must satsify when specified events and conditions occur. |
| 4 | Activation/deactivation rules | The logic which determines when one or more functions are activated and deactivated. |
| 5 | Activation time | The interval of time that a function or state is active. |
| 6 | Activity | One or more related actions. |
| 7 | Analysis | The process of evaluating elements, properties and associated relationships. |
| 8 | Analysis model | A model used to analyze the structure, behavior, and/or properties of systems and environments. |
| 9 | AP-233 | ISO STEP Application Protocol for Systems Engineering Data Interchange Standard |
| 10 | Arc | An association between two or more nodes. |
| 11 | Association | Refer to UML specification |
| 12 | Behavior | The activation/deactivation of one or more functions. Note: This describes how a system interacts with its environment. Reactive behavior includes the stimulus and response. |
| 13 | Behavior allocation | The allocation of functions and/or states to systems, and the allocation of inputs and outputs to system ports. |
| 14 | Boolean | Refer to UML specification |
| 15 | Category | A partitioning of elements based on a classification. |
| 16 | Complex number | A number which can includes a real and imaginary part. |
| 17 | Component | A constituent part of an element or system that contributes to the properties and behaviors of the whole (emergent). Note: A leaf component does not have constituent parts. |
| 18 | Composite function | A function which is decomposed into lower level functions. |
| 19 | Composite state | A state which includes nested states. |
| 20 | Concurrent state | A state which is active at the same time as another state that is part of the same composite state. |
| 21 | Condition | An expression with a discrete output, which is true as long as the expression evaluates true, and is false otherwise. |
| 22 | Connecting component | A specialized component or system, whose primary function is to connect the outputs from one system to the inputs of another system via its ports. Note: This may be a wire, network, or mechanical coupler that has properties and behaviors, which may transform the inputs and outputs. |
| 23 | Connection | Identification of which ports connect to one another. |
| 24 | Connection path | Multiple connections that may represent a single logical connection. |
| 25 | Continuous time model | A model which is based on properities that vary continuously with time. |
| 26 | Control input | An input that activates or deactivates a function. |
| 27 | Control operator | A specialized function that provides logic to transform input events and conditions to discrete values that are supplied as control inputs to functions. |
| 28 | Data | A component of information. |
| 29 | Data type | Refe to UML specification |
| 30 | Decomposition | A description of a whole in terms of its component parts. |
| 31 | Dependency | A relationship where a change to one entity results in a change to the other. |
| 32 | Deployment | A allocation of one component to another that is often associated with the utilization of resources across the distributed nodes of a system. |
| 33 | Design | The process of transforming requirements to an implementation. |
| 34 | Design constraint | A requirement that one or more components of a system must satisfy. Note: This term is sometimes used to refer to a |

# 3 SYSTEM ANALYSIS WORKING GROUP

- **Manas Bajaj et al**
- Objective; Seamless and integrated analyses of complex systems across their lifecycle
- Goals;

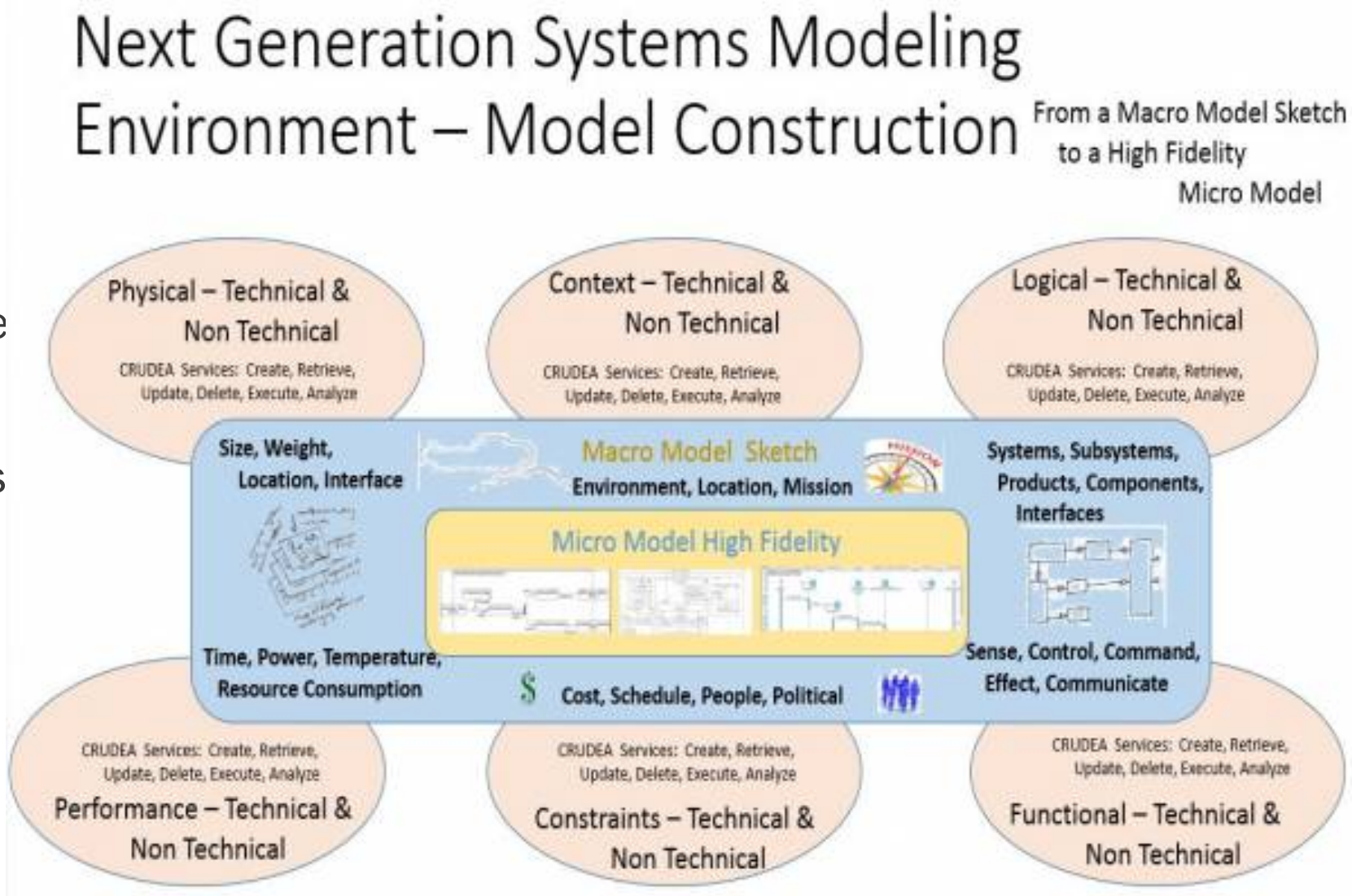
1. Support various types of system analyses and execution tools (e.g. solving, simulation, budgeting, etc.)
2. Manage analysis models and relate results to decisions
3. Improved user interaction to define/generate, execute, archive analysis models (analysis lifecycle)

# 4 SYSML V2 MODEL FORMALISM WORKING GROUP

- **Yves Bernard et al**
- Objective;  Identify requirements related to the logical formalism for SysML v2
- The next-generation modeling language must include precise semantics that avoid ambiguity and enable a concise representation of the concepts
- The language must derive from a well-specified logical formalism that can leverage the model for a broad range of analysis and model checking
- This includes the ability to validate that the model is logically consistent, and the ability to answer questions such as the impact of a requirement or design change, or assess how a failure could propagate through a system
- The language and tools must also integrate with a diverse range of equation solvers and execution environments that enable the capture of quantitative data

# 5 SYSTEMS ENGINEERING MODEL CONSTRUCTION WORKING GROUP

- **Ron Williamson et al**
- **Objectives;** Elaborate concepts, requirements, and metrics for effective model **construction** that support the next generation system modeling language (SysML v2). Identify a set of services that support model **construction.**



# 6 MODEL VISUALIZATION WORKING GROUP

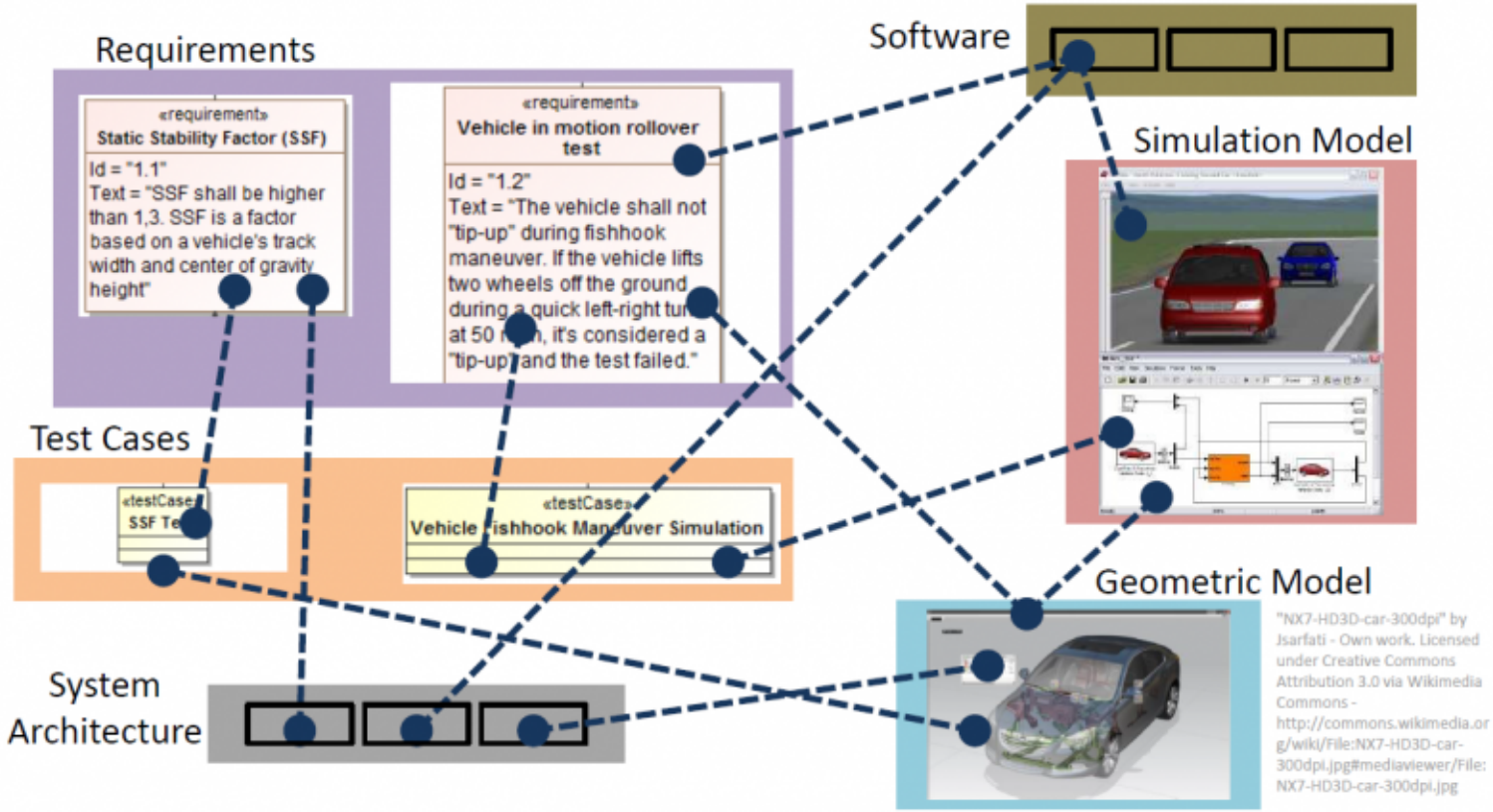
- **Chris Schreiber et al**
- Elaborate concepts, requirements, and metrics for effective model **visualization** that support the next generation system modeling language (SysML v2). Identify a set of services that support model **visualization**.
- Easily navigate to/from model data to other visualizations from other Engineering (and Program Disciplines)
- Must include non-diagrammatic views (tables, matrices, etc.)
- Must support both static and dynamic representations
- Must support SE Use Cases
- Should be initially simple
- Should provide easy access to associated data (meta data)

# 7 MODEL LIFECYCLE MANAGEMENT WORKING GROUP

- **Laura E Heart et al**
- **Objective;** dentify the capabilities (as services), and infrastructure needed to support a key aspect of MBSE Systems Modeling Environment (SME) namely, managing models and their data.
- The next-generation modeling language must be capable of management in a heterogeneous and distributed modeling environment.
- The ability to manage change to the model, where multiple users are collaborating on a single model, is challenging enough. This basic capability requires extensive branch and merge capability that includes effective means for evaluating and integrating changes from multiple users, while maintaining a history of all changes. These challenges increase when multiple models and tools are all part of the collaboration. The ability to integrate with Product Lifecycle Management (PLM) environments, which enable versioning, configuration, and variant management, is a fundamental SME requirement

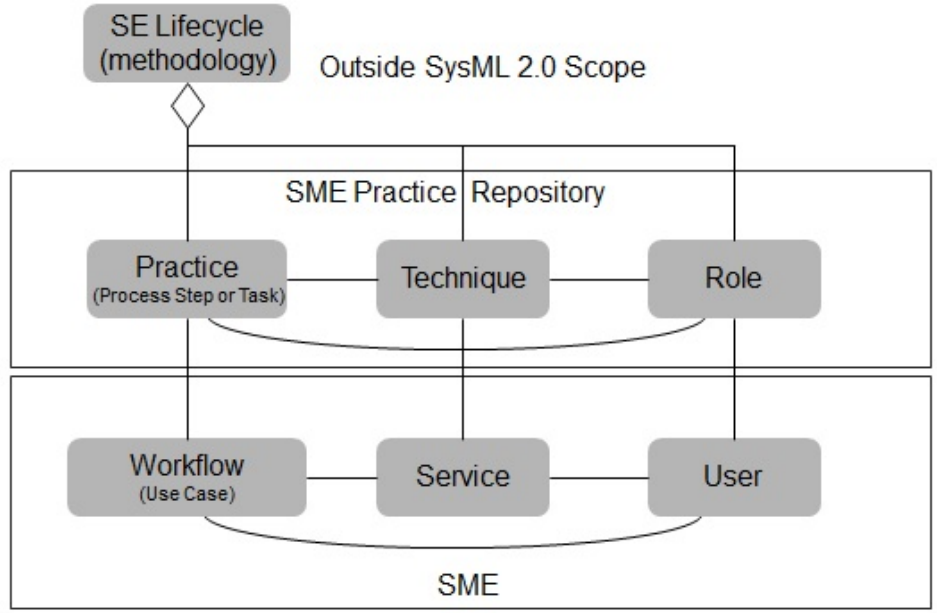
# 8 SYSTEMS ENGINEERING INTEROPERABILITY WORKING GROUP

- **Axel Reichwein et al**
- Objective; Identify requirements related to interoperability such that SysML can support standards for data interoperability, traceability, analytics in a multi-disciplinary engineering context
- The next-generation modeling language must provide a standard (i.e. OSLC) application programming interface (API) to provide dynamic access to the model, while providing appropriate access controls



# 9 MBSE WORKFLOW AND COLLABORATION WORKING GROUP

- **Hedley Apperly et al**
- Objective; Define the concepts, requirements, and metrics for an effective systems engineering workflow, its practices and collaboration for the next generation system modeling language
- The next-generation modeling language must be accompanied by a set of recommended, preferred processes
- The next-generation practice repository must provide services for modifying, creating and implementing a preferred processes for model-based systems engineering
- Rather than only standardizing a modeling language and logical systems modeling environment (SME) process advice (based on industry norms) must also be provided



# 10 REQUIREMENTS MODELING WORKING GROUP

- No details published on wiki to date
- Lots of reference material being reviewed – everyone has a book on this

# SUMMARY

# NEXT STEPS

- Working Group to draft RFP
- Project broken down into 10 sub-working groups (Concepts)
  - Systems Engineering Workflow Use Cases Working Group
  - Systems Engineering Concept Model Workgroup
  - System Analysis Workgroup
  - SysML v2 Model Formalism Working Group
  - Systems Engineering Model Construction Focus Area
  - Model Visualization Working Group
  - Model Lifecycle Management Working Group
  - Systems Engineering Interoperability Working Group
  - MBSE Workflow and Collaboration Working Group
  - Requirements Modeling Working Group

# TO FIND OUT MORE & TRACK PROGRESS

- Go To http://www.omgwiki.org/OMGSysML/doku.php?id=sysml-roadmap:sysml_assessment_and_roadmap_working_group#system_modeling_assessment_and_roadmap_working_group

PTC®

# PRODUCT & SERVICE ADVANTAGE®