MBSE I:

Tuesday, March 22, 2016
Meeting Room 2

Meeting in the Middle: Integrated Architecture Model Management Using an MBSE Process Application
Philip D. Nerren

How does a single over-seeing entity or organization leverage model based systems engineering (MBSE), to maximize reuse (and insure interoperability) between inter-related missions over the course of a long-range program? A program consisting of a series of multi-billion dollar missions that spans multiple decades and budgets. Model based systems engineering (MBSE) techniques are well suited to support these process needs.

A process was formulated to structure and to manage top down and bottoms up derived architecture design data for the purpose of interrogating inter-related missions architectures over the course of a long-range program. The scope of these interrelated and diverse missions include human, robotic, long and short durations, as well as destinations. These diverse objectives are explored in a multifaceted hierarchical modeling framework to gain insight and visibility to commonality and complexity across the integrated mission objectives. The process facilitates exploring options to maximize reuse (and insure interoperability) between inter-related missions over the course of a long-range program.

Gaining insight into such diversity for purposes of planning requires the framework to manage both systems management and systems design data. The systems management data include requirements, evaluation criteria, risks, technologies, trade study result/decisions, while design analyses, performance values, and designs specification capture the system design. Unified modeling language (UML) diagrams were used to model the systems view, with an operational, functional and physical context from a behavioral and performance perspective but also to provide a framework for mapping system design and systems management parameters to the mission architectures.

The information/data complexity requires a corresponding data model to be established in order to cope with the pertinent information's cross references through the process framework. The data architecture is vital and complementary to the overall model framework. The process also supports understanding relevance in system depth verses breadth throughout the architecture studies. The majority of architecture studies do not allow sufficient time to analyze the necessary fidelity of a system, which supports a better decision rationale.

The model hierarchy also structures the study in such a manner to gain visibility in system behavior (enhance function flow block diagrams (EFFBD) diagrams) across the operational need, which aids in determining where it is appropriate to increase the fidelity of analysis. The additional benefit of such a structured method is to provide a requirement foundation that is efficient and traceable for future implementation.

Engaging ontologies to break MBSE tools boundaries through semantic mediation

Uri Shani, Shmuela Jacobs, Niva Wengrowicz, Dov Dori

We introduce ontologies (as in the Semantic Web ontology language OWL) to serve as formal description of the modeling languages of model-based systems engineering (MBSE) tools. In these tools, systems are designed via abstractions, each tool with its own conceptual modeling languages having different syntaxes and different semantics. This creates barriers in sharing these mathematical models among the tools. Our journey starts with the web, where large amounts of information from any sources on the Web can be linked and combined in many ways. The Semantic Web added meaning through OWL ontologies to the information in web pages so machines could better process information to enhance users' experience.

The field of MBSE offers a great many different tools to the engineer, each representing the system under design or a view of that system albeit mathematically, but differently so that sharing a model from one tool with another is impossible, or darn difficult and error prone.

This problem is often classified as "tools interoperability", but it is primarily a language interoperability problem. The Open Services for Lifecycle Collaboration (OSLC) specifications and initiative brought the Semantic Web technologies into MBSE by introducing Resource Description Framework (RDF) for common model representation, RESTful protocols, as a common communication and data exchange method, and enabling the linking of model elements in the different tools that are used for product lifecycle management (PLM).

The semantic mediation container (SMC) is a platform developed by IBM Research as part of EU projects (SPRINT, DANSE and now PSYMBIOSYS) to support the mediation of models represented in RDF and exchanged over the Internet using RESTful API. SMC extends the benefits of the OSLC approach by adding semantics using the Web Ontology Language (OWL) specifications to define language ontologies. Each tool exchange models with the SMC platform, where models are constructed according to specific ontologies (in OWL) per each of the tools. Models can then be mediated to comply with different ontologies. The rules governing the mediation are also coded as OWL ontologies that bridge two different ontologies and are interpreted by a mediator. Bridging is a form of transformation, or an inference over the statements of the 3 involved ontologies, and those of the input RDF model, all driven by a mediation engine we term mediator. In this paper we introduce the first stages of mediation as applied to two different modeling tools; Rhapsody that implements the OMG standard SysML specification, and OPCAT – the OPM CASE tool implementing the Object-Model Methodology (OPM) which is an emerging ISO 19450 standard.

Using Model Based Systems Engineering in Policy Development: A Thought Paper

Shamsnaz Virani and Tom Rust

Systems engineers often decide the strategy for design and development of complex system. At that point they also have to review all the policies and standards governing a system. Policies or standards are typically documents, using natural language explaining different constraints subjected to the system. As we move towards more model centric approaches in systems development, it therefore is natural to experiment with model based approaches in policy development. Model Based Systems Engineering (MBSE) has found to reduce risk, improve communication, improve quality, and increase productivity. Policy documents are often misinterpreted, not sufficiently tested, and left unchecked result in several massive policy failures. In this thought paper we assume Policy as a System and use MBSE approaches to help policy development process.

A domain-specific language for model composition and verification of multidisciplinary models

Amogh Kulkarni, Daniel Balasubramanian, Gabor Karsai, Anantha Narayanan, Peter Dennoc

Complex, engineered products and manufacturing processes often necessitate integrated analysis that cuts across physical domains and engineering disciplines. When the domain-specific models that contribute to the overall analysis process are available then the problem can be addressed by composing them into an analysis workflow which then can be executed using some execution platform. Such a composition and integrated analysis is essentially a systems engineering approach applied to an engineering process. In this paper we describe a model integration language that allows the rapid composition of models, the verification of the composition and the generation of executable code and other engineering artifacts that are needed for model execution on a software platform. The language is based on OpenMDAO, a widely-used model execution framework, and it improves the engineering process by checking composition constraints that must be satisfied by the integrated model and by automatically generating executable code that facilitates the run-time integration of the models. This paper describes the design of the language, illustrates its use through a running example and outlines future work.