2015 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
SYSTEMS ENGINEERING (SE) TECHNICAL SESSION
AUGUST 4-6, 2015 – NOVI, MICHIGAN

THE UNITED FEDERATION OF MODELS: MODEL-BASED SYSTEMS ENGINEERING AT ITS BEST

Michael. J. Vinarcik, ESEP-Acq, FESD
Booz Allen Hamilton
Troy, MI

ABSTRACT

As systems become more complicated, managing them from conceptual design through disposal has become significantly more difficult. Traditional process- and document-based methods often struggle to cope with the realities of designing and fielding modern engineered systems; relevant information can become segmented and out-of-sync, leading to costly errors.

The International Council on Systems Engineering (INCOSE) initiated a Model-Based Systems Engineering (MBSE) initiative nearly fifteen years ago; one of the fruits of this effort is the System Modeling Language (SysML). Modern SysML tools are quite mature and have considerable capabilities to capture, characterize, and connect information.

A number of efforts are currently underway to maximize the value of SysML models, including the development of visualization, co-simulation, and robust model integration. An entire ecosystem of adjacent tools is evolving that will significantly impact the practice of systems engineering within the next five years. This paper presents an integrated look at publically-available information about existing worldwide efforts and their implications for systems engineering practitioners.

SYSTEM MODELING

Engineers have a long history of using models to describe systems; these abstract reality in a useful way to enable the solution of engineering problems. For example, a simple mathematical description of a spring-mass system is a model. As systems have grown in complexity, particularly with the explosion of functions delivered by software, traditional methods of describing, tracking, and managing relevant information have slowly become overwhelmed.

To address this, the International Council on Systems Engineering initiated the development of the Systems Modeling Language (SysML). SysML allows robust modeling of a system's behavior, its structure, and associated relationships (such as parametrics).

A panelist at a recent MBSE conference held at the Jet Propulsion Laboratory opined that "the use of SysML drives crispness in the formulation of issues)." Every node and relationship created in the model has a specific meaning and a clear notation. Careful construction of the model, with subject-matter expert (SME) input, leads to an unambiguous representation of the system of interest.

It should be noted that effective system modeling requires systems engineering expertise, knowledge of the modeling language (SysML for example), and familiarity with the modeling tool. Relevant certifications are now available to provide recognition tied to objective standards of competence. INCOSE has established its Systems Engineering Professional (SEP) certification track for systems engineers; it includes Associate, Certified, and Expert levels. The Object Management Group (OMG) has created the OMG Certified System Modeling Professional Certification (OCSMP), with an associated body of knowledge, to establish a similar set of modeling credentials. OCSMP is a four-level certification, with Model User through Model Builder-Advanced levels.

DERIVED PRODUCTS

Because the primary mechanism for developing SysML is the creation of SysML diagrams, many engineers have the mistaken impression that the SysML model is the drawing; instead, it is the underlying relationships between system elements that are presented (or created) as drawings. Just as a CAD solid model may be used to derive "traditional" two-dimensional views, a SysML model can be used to derive a multitude of other work products. This is of particular importance because many SMEs will rarely, if ever use a SysML modeling tool but will rely upon information resident within the model.

Because the system model, used properly, is the "single repository of truth" about a system under development, it is important to maintain the "freshness" of these derived products to ensure stakeholders are using current, relevant information when conducting analyses or making decisions. Considerable effort is being expended by both tool vendors and customers to leverage model content directly in support of modeling and simulation, document generation, and analytical tools.

INTEGRATION VS. FEDERATION

Merriam-Webster defines *integrate* and *federate* as:

Integrate: to form, coordinate, or blend into a functioning or unified whole

Federate: to join (organizations, states, etc.) in a federation (an organization that is made by loosely joining together smaller organizations)

Integration is often sought as the default desired outcome: direction connection and/or interoperability between software tools. However, it may be more useful to consider integration and federation as a spectrum, with the needs of engineers and other stakeholders, as well as the frequency of updating required for a given product, determining how tight a coupling is required between two tools. Integration is not free; the effort and/or time saved by integrating must be balanced against the development costs (and maintenance costs, which are not trivial).

CURRENT AND EMERGING PRODUCTS

It is not the intent of this paper to endorse any particular product or solutions but to present representative examples of current commercial-off-the-shelf (COTS) and open-source products. These are intended to illustrate the capabilities that can be achieved immediately by appropriate selection and integration/federation, as well as those that will be emerging in the near future.

These tools fall across the spectrum from tightly integrated to loosely federated; each development team seeking to maximize the benefit of system modeling should consider what level of integration is optimal for its needs.

Note: Using a system model as a "hub" for federation reduces the number of interfaces to be managed vs. direct connection (the interfaces grow linearly rather than as the square of interfacing elements).

INTERCHANGE FORMATS

Excel

Microsoft Excel is the lowest common denominator for data exchange and provides many useful capabilities (filtering, lookups, PivotTables, graphing, and other analyses). Comma-separated value (CSV) and native Excel exports are provided by many modeling tools; these exports many not be as "rich" as other formats but are adequate for many useful analyses.

XML Metadata Interchange (XMI) [1]

The Object Management Group (OMG), which is the owner of the SysML specification, has published the XMI standard to support model interchange. It defines the import and export of data using the Extensible Markup Language (XML). Tool vendor support varies but this format tends to support richer data exchange than CSV or Excel formats.

Requirements Interchange Format (ReqIF) [2]

Another OMG-driven standard, ReqIF allows lossless exchange of requirement information via XML. Although this standard is more focused than XMI, it does support interchange for requirements-focused analysis.

Open Services for Lifecycle Collaboration [3]

The Open Services for Lifecycle Collaboration (OSLC) is an effort to define standards to support tool integration. A variety of working groups are currently developing a suite of standards to support requirements management, architecture management, and other topics.

MODEL CONNECTIONS

SysML to CAD

Connection of a system model directly to solid models/CAD provides many benefits; requirements may be viewed in either domain, mass and other properties may be exchanged, and both architectural and physical representations may be kept synchronized. In addition, these connections allow the use highly mature product lifecycle management (PLM) tools for configuration control and version management. These tools tend to be more tightly integrated, with the associated higher costs associated with development and maintenance.

Syndeia [4]

InterCAX (a spin-off of the Georgia Institute of Technology) has developed Syndeia, which connects product lifecycle management (PLM) and database systems (Teamcenter, Windchill, and MySQL), CAD models (NX and Creo), simulation models (Simulink), with SysML modeling tools (No Magic's MagicDraw and IBM's

Rhapsody). Similar capabilities are available or under development from other major software providers (although some are focused on vertical integration within one vendor's toolset).

Open-MBEE [5]

Jet Propulsion Laboratory is leading the development of Open-MBEE, the Open Model-Based Engineering Environment. Open-MBEE (available on GitHub), presents a system model to client applications via a web-based application programming interface (API). This enables two-way communication between the system model, analysis tools, and generated documents. The open source nature of the project is drawing expertise from leading system modelers and the low cost of experimentation make this solution particularly attractive for medium and small enterprises.

MODEL ANALYSIS AND PRESENTATION

Tom Sawyer Software [6]

One of the most common criticisms of SysML drawings is that they are ugly; they are often starkly utilitarian in appearance. Recreation or reformatting of graphics for presentation to management or other stakeholders can be time-consuming with little direct value added to the system model. Tom Sawyer software is a product that addresses this by enabling modelers to import the system model, define rules for deriving more appealing representations, and automatically generating them on demand. A variety of associated network analyses are also possible, including link analysis and network topology assessment.

Gephi [7]

Gephi is an open-source tool for visualizing networks and graphs. Described as a “Photoshop but for data,” Gephi allows users to visualize and analyze network information to conduct exploratory data analysis. SysML model exports (such as requirements or activity relationships) can be imported into Gephi and displayed with its graphing tools. It also generated network metrics from a given data set.

Design Structure Matrices (DSM)

Design Structure Matrices (related to N2 diagrams) are useful to assess the interactions between system elements. Numerous software applications (both open source and commercial) exist, with varying degrees of support for CSV and XMI import. DSM analysis is particularly appealing because of the various partitioning algorithms that may be applied to a system. The results of these clustering algorithms can inform up-front architecture development or allow assessment of an existing system.

SIMULATION AND CO-SIMULATION

Parametric Solvers and Simulation

Parametric relationships, as defined in the system model, provide explicit relationships between values and properties of system elements. Most modeling tools offer integration solutions that connect model constraint relationships with external solvers such as MATLAB/Simulink, Mathematica, and others. Some also provide embedded simulation support, allowing the modeling tool to execute a behavioral or parametric representation.

It should be noted that robust simulation and Monte Carlo analysis can (and should) be used to verify system models, as appropriate. For example, a state machine can be simulated to ensure that the system never enters an unintended (or unsafe) state. This provides a level of rigor that is not easily achievable via traditional, document-centric methods.

Phoenix ModelCenter [8]

Phoenix Integration has created Phoenix ModelCenter, a tool that supports connection of system models with a variety of simulation and analysis tools. It also allows the automation of simulation workflows to support trade study and other analyses. In addition to modeling tools such as MATLAB and Mathematica, it permits the inclusion of legacy FORTRAN and C++ models.

Functional Mockup Interface (FMI) [9]

The open source FMI standard combines XMI files and compiled code to enable tool-independent model exchange and co-simulation. 72 tools [10] currently support it and enable the generation of Functional Mockup Units (FMUs). Compiled FMUs are typically royalty-free; this simplifies co-simulation efforts (eliminating the need to track executions and payments to multiple tool vendors).

In the author's opinion, FMI is likely to support tighter analytical integration within existing SysML modeling tools. Although this integration will be more costly than federation via content export, the benefits to trade studies and other analyses will be substantial.

CONCLUSION

The emergence of SysML as the de facto system modeling language has enabled the growth of an ecosystem of tools that span the spectrum from integration to federation. These allow systems engineers and SMEs to have clear representations of systems that can be used to generate derivative work products to support stakeholder analyses and decision making.

Considerable open source and commercial capability can be harnessed immediately to support acquisition and development projects…with commensurate benefits in clarity and consistency.

REFERENCES

[1] OMG XMI Standard, http://www.omg.org/spec/XMI/.

[2] OMG ReqIF Standard, http://www.omg.org/spec/ReqIF/.

[3] OSLC Standard, http://open-services.net/.

[4] Syndeia, http://intercax.com/products/syndeia/.

[5] For more information about Open-MBEE, see http://www.omgsysml.org/View_Paper-IEEE_2013.pdf or https://github.com/Open-MBEE.

[6] Tom Sawyer Software, http://www.tomsawyer.com.

[7] Gephi, https://gephi.github.io/.

[8] Phoenix Integration, http://www.phoenix-int.com/.

[9] Functional Mockup Interface, https://www.fmi-standard.org/.

[10] Tools supporting FMI, https://www.fmi-standard.org/tools.

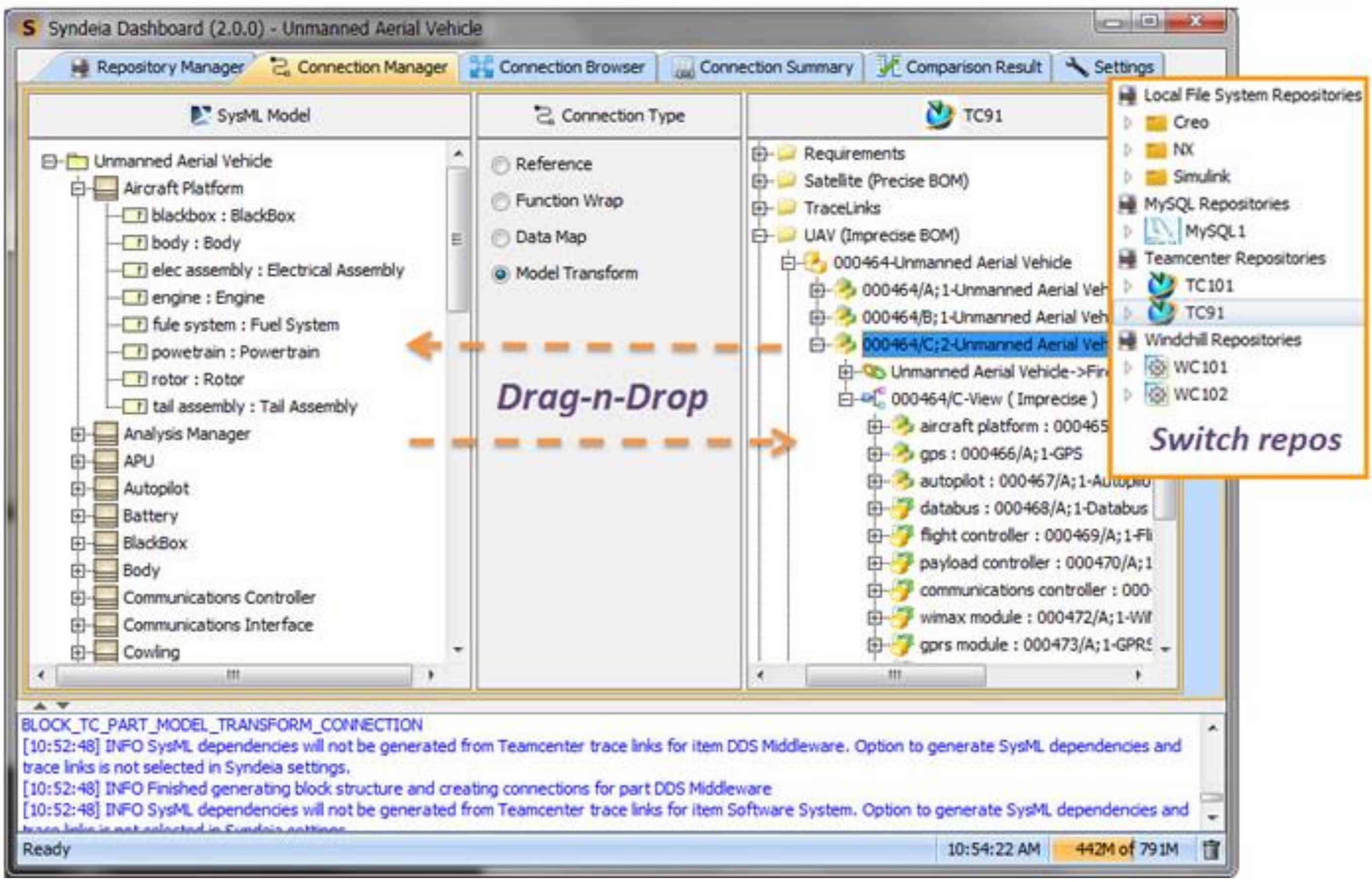

Figure 1: Syndeia (http://intercax.com/products/syndeia/)

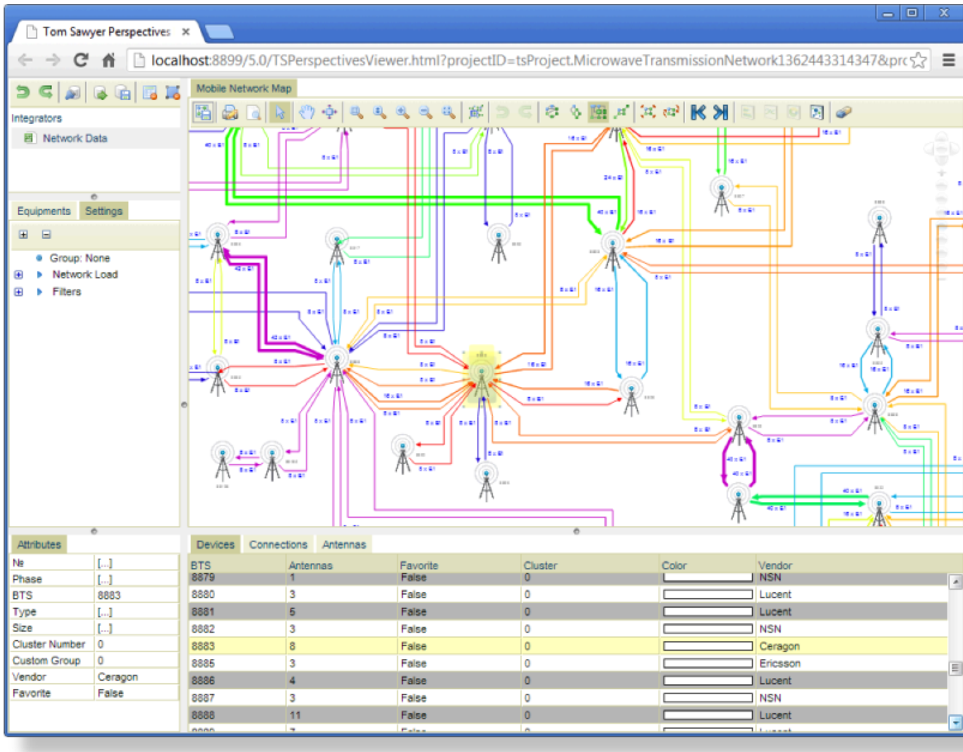

Figure 2: Network Topology Diagram (Courtesy Tom Saywer Software)

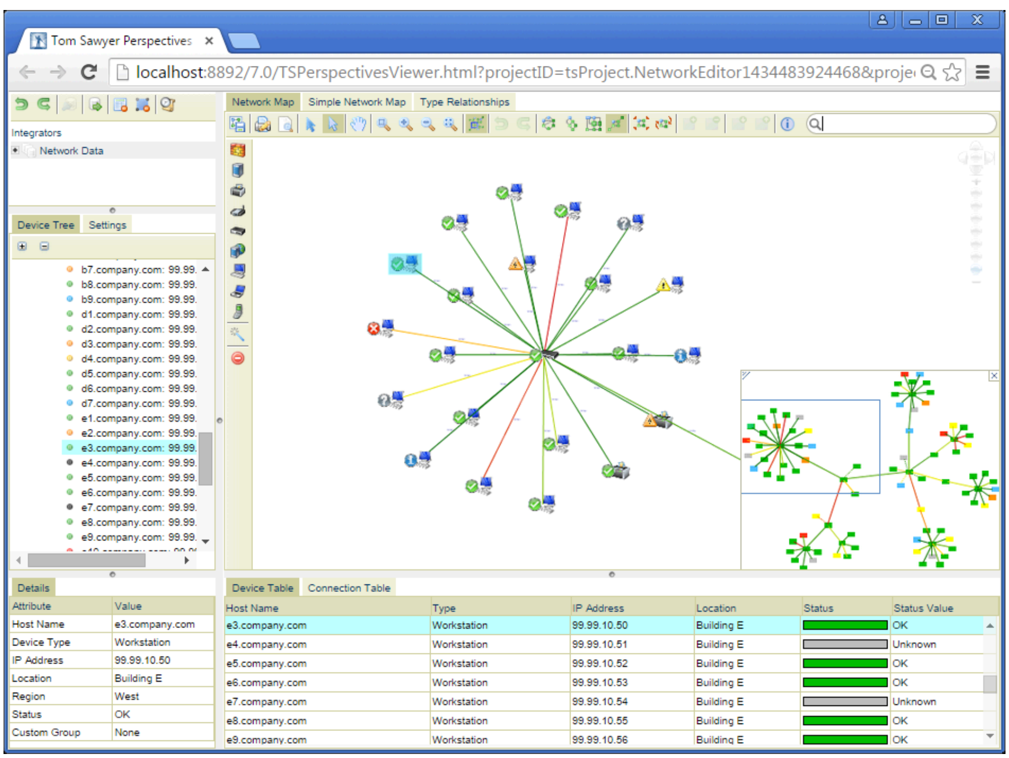

Figure 3: Network Editor (Courtesy Tom Saywer Software)

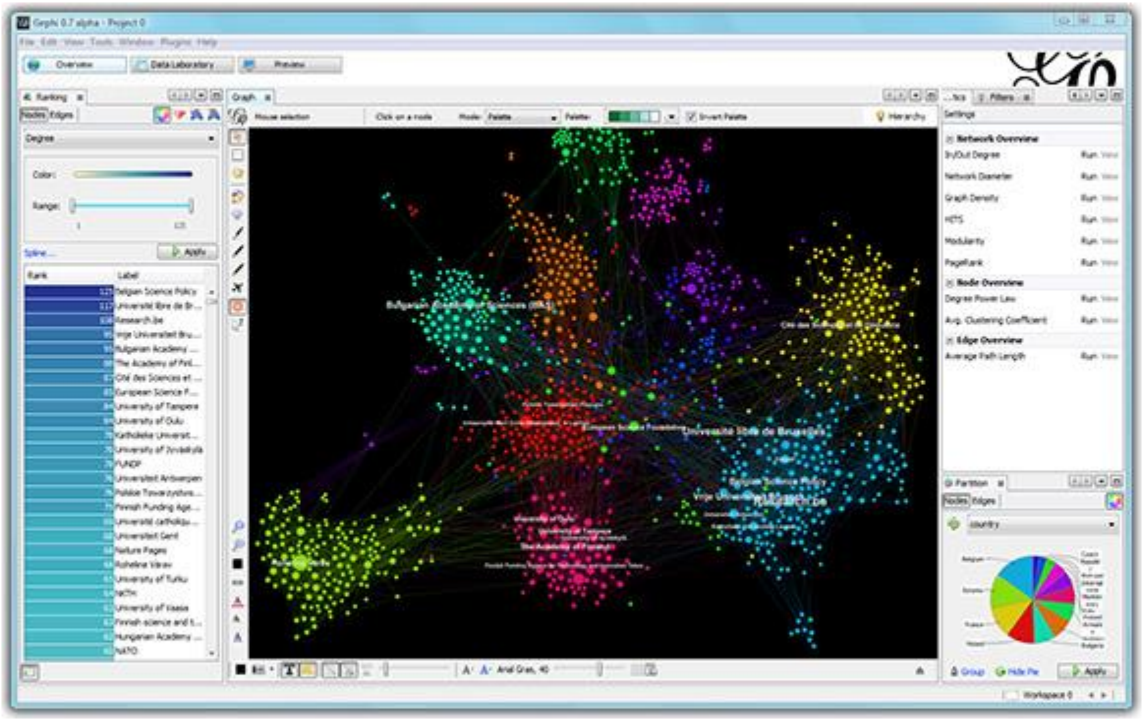

Figure 4: Gephi (https://gephi.github.io/)