75 Fifth Street NW, Suite 312
Atlanta, GA 30308, USA
voice: +1-404-592-6897
web: www.Intercax.com
email: info@Intercax.com

Dirk Zwemer, Intercax LLC

Technote: Application of MBE to Automotive Engineering

Introduction

Engineering in the automotive industry faces a broad set of challenges:

- Rapidly changing technology encompassing mechanical, electronic, software and other engineering disciplines
- Diverse bodies of requirements: market, regulatory and technical
- High product mix and extensive individual customization and integration
- Extended supply chains across organizational and international boundaries

Because of these challenges, automotive engineers have been leaders in developing and adopting advanced techniques in systems engineering. Their engineering tool needs include

- An architectural framework inclusive of structure, behavior, requirements and analysis
- Cross-vendor and cross-discipline interoperability
- Easy-to-use, non-discipline-specific query and visualization capabilities

Model-Based Engineering (MBE) depends on a single, self-consistent digital model of the system, spread across multiple engineering tools and repositories, as illustrated in Figure 1. It extends the concept of Model-Based Systems Engineering (MBSE) of capturing a system's specification as a model, rather than a series of static disconnected documents. As the MBSE idea developed, it became clear that a single model in a single tool, for example, a SysML system architecture model, was insufficient for the purpose.

In Model-based Engineering, all the disciplines and tools in the engineering process are engaged in an ongoing network.
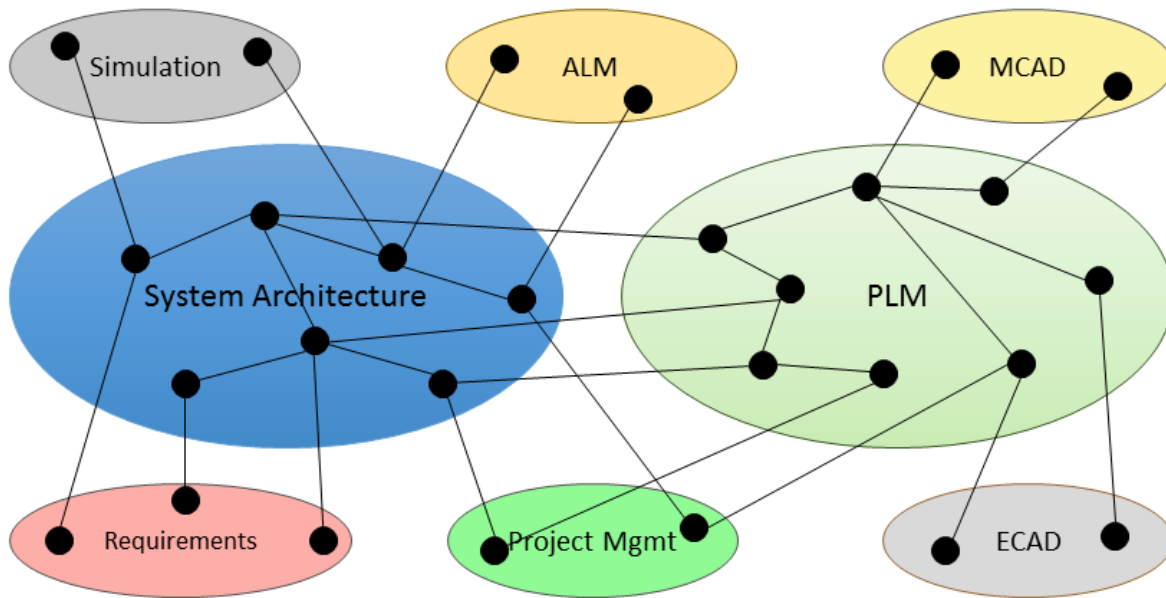

Figure 1 Total System Model as a Network of Connections inside and between Engineering Software Tools and Repositories
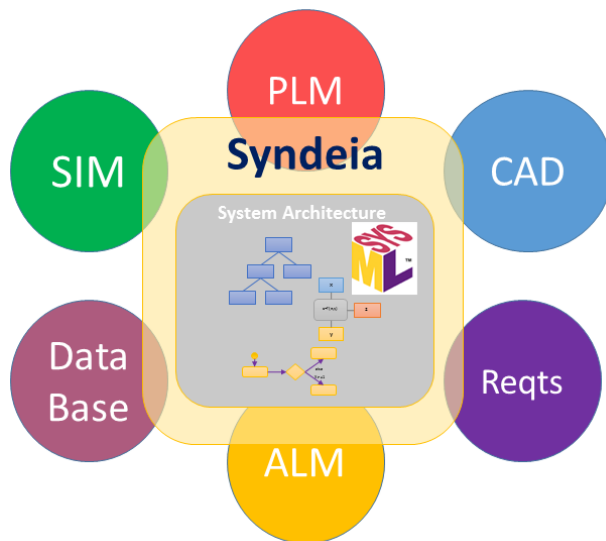
Syndeia

Figure 2 Syndeia architecture

Syndeia™, from Intercax, is a platform for the practice of MBE. As shown in Figure 2, it uses a system architecture model as central hub and connects SysML elements to elements in PLM, CAD, ALM (application lifecycle management), project management, requirements, simulation and other engineering tools.

SysML is an effective medium for building a high-level roadmap of the system because it provides a rich language for connecting structure, behavior, requirements and analysis concepts, concepts that can map to corresponding concepts in more specialized tools. Currently, Syndeia supports two SysML modeling tools, MagicDraw (No Magic Inc.) and IBM Rational Rhapsody, and a variety of other tools, including Teamcenter and NX (Siemens), Windchill and Creo (PTC), MySQL (Oracle), DOORS NG (IBM Rational), Simulink (The Mathworks), JIRA (Atlassian), GitHub (GitHub, Inc.) and Excel (Microsoft). Contact Intercax to learn about support for other tools.

System Architecture Model

We begin by describing a simple SysML model (available for download from the Intercax website) that will act as the central connection point for our total system model. Structural decomposition for our system-of-interest, Vehicle (Figure 3), shows multiple subsystems. The same diagram shows that six different specializations of the Powertrain subsystem have been identified, including several hybrid variants.
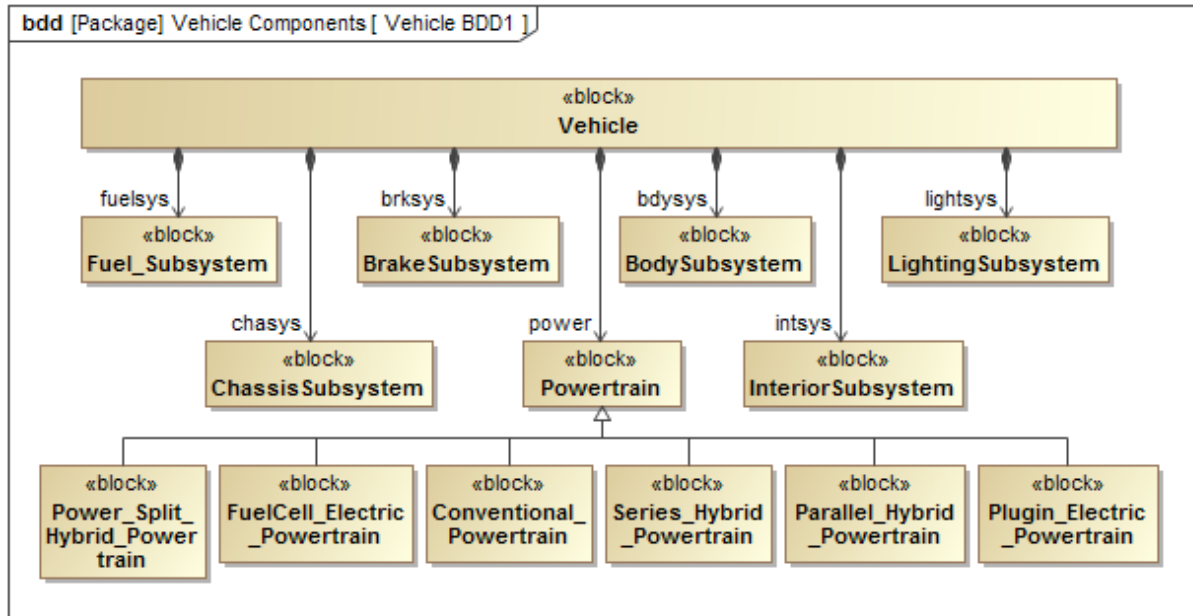

Figure 3 Vehicle Structural Decomposition, including Powertrain variants

All six share many of the same functional elements (e.g. electrical storage system ESS or internal combustion engine ICE), but have them connected in different ways. Connectivity for the Series Hybrid Powertrain variant is shown in Figure 4.
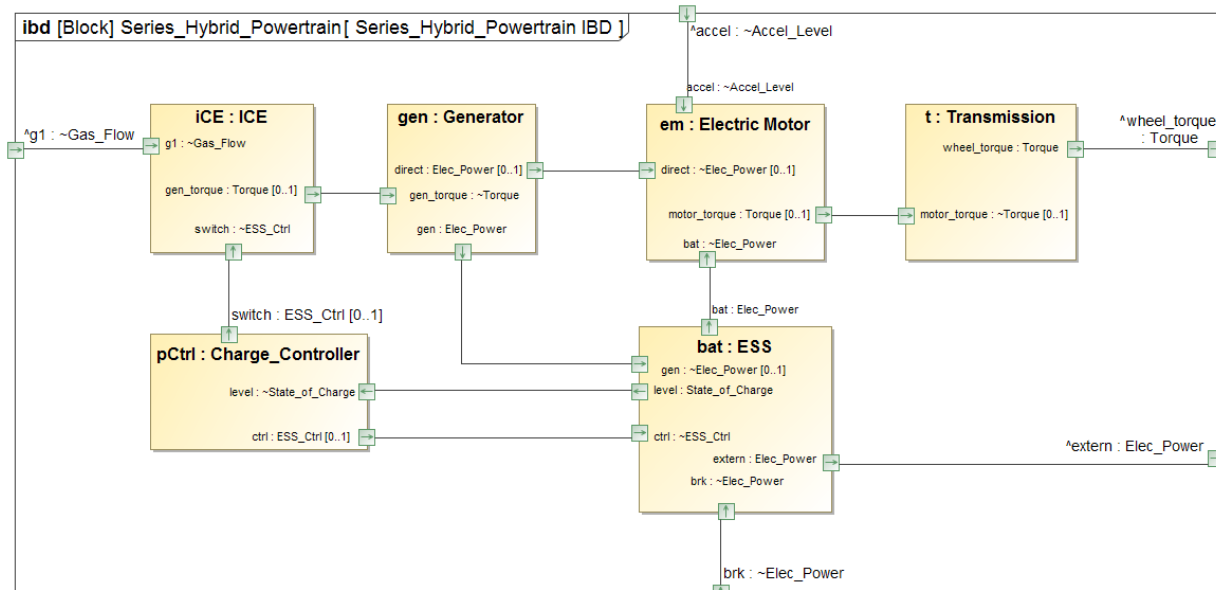

Figure 4 Series Hybrid Powertrain connectivity in SysML internal block diagram

The system architecture model can also include system requirements, especially where separate requirement types are related to each other or to other system elements. Figure 5 shows several examples of such relationships, where technical requirements are shown as derived from market requirements and satisfied by structural elements of the system.
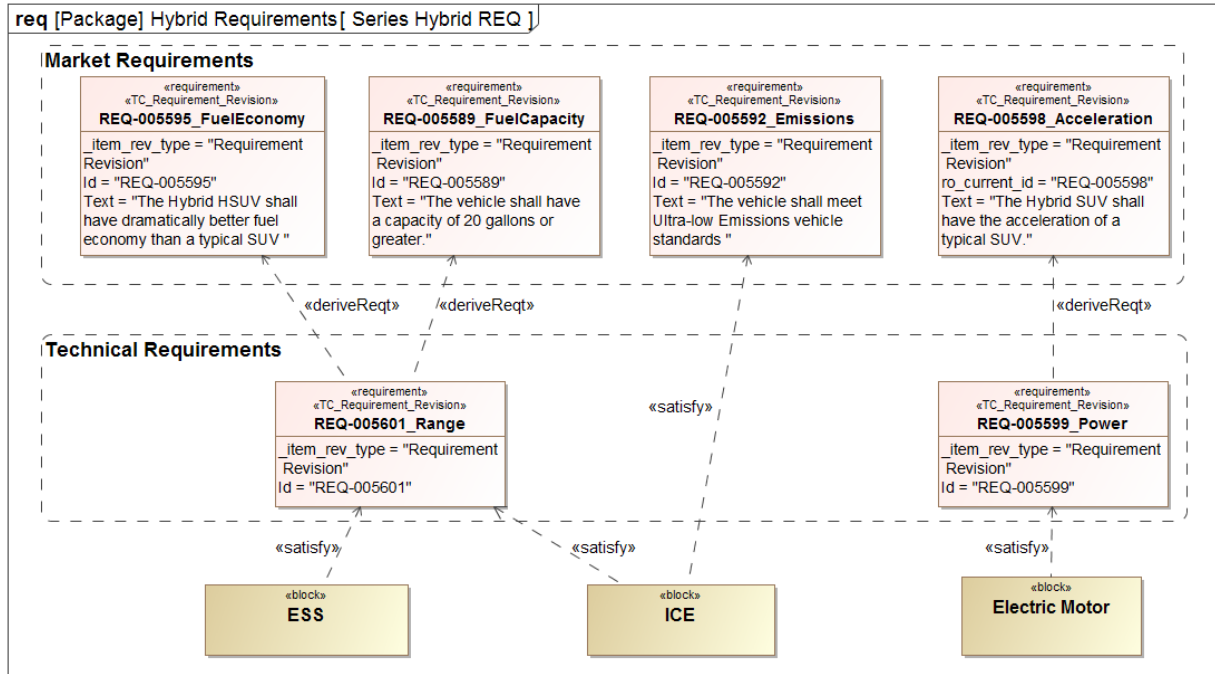

Figure 5 Sample SysML connectivity between market requirements, technical requirements, and structure

The system architecture model also describes the behaviors or functions of the system and its parts. In Figure 6, the overall function of the series hybrid powertrain is decomposed into a series of subfunctions (actions), each of which is allocated to a structural component. These allocation dependencies are indicated by the swimlanes in the activity diagram.
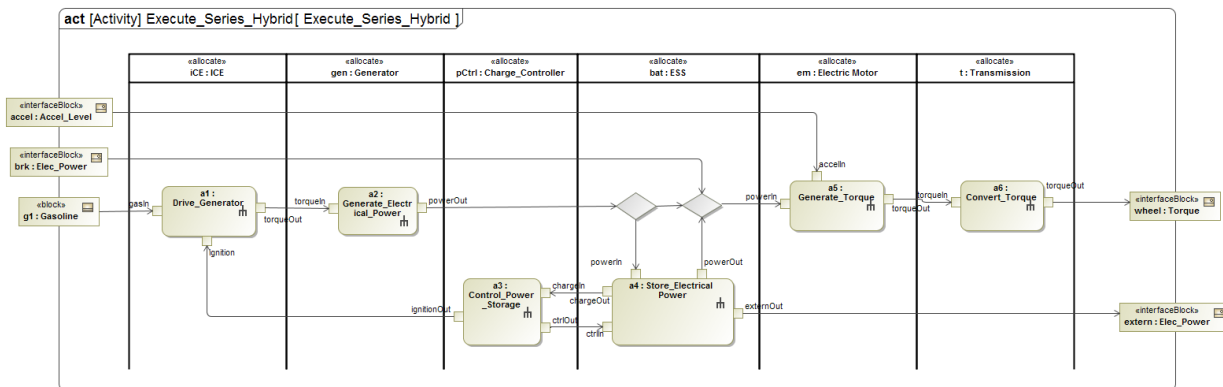

Figure 6 SysML Activity diagram for Series Hybrid Powertrain, with allocations to structural components

All the relationships shown in these figures, representing composition, dependency and flow, are "intra-model" connections, connections created and stored within a single software tool, in this case, the SysML modeling tool. But many of these same SysML elements also relate to elements in other

software tools. These are "inter-model" connections, which we will create, manage and use with Syndeia, as described in the next section.

Inter-model Connections

Inter-model connections are made using the Syndeia dashboard (Figure 7) using a variety of drag-and-drop mechanisms. In our sample model,

- Market and technical requirements were dragged from two different PLM repositories into the SysML model, retaining their hierarchical structures;
- Structural connectivity (e.g. Figure 4) were dragged from SysML into MATLAB, creating a Simulink block structure (blocks, ports and signals, but no internal MATLAB code);
- Single reference links were created from SysML activities to software folders in GitHub.
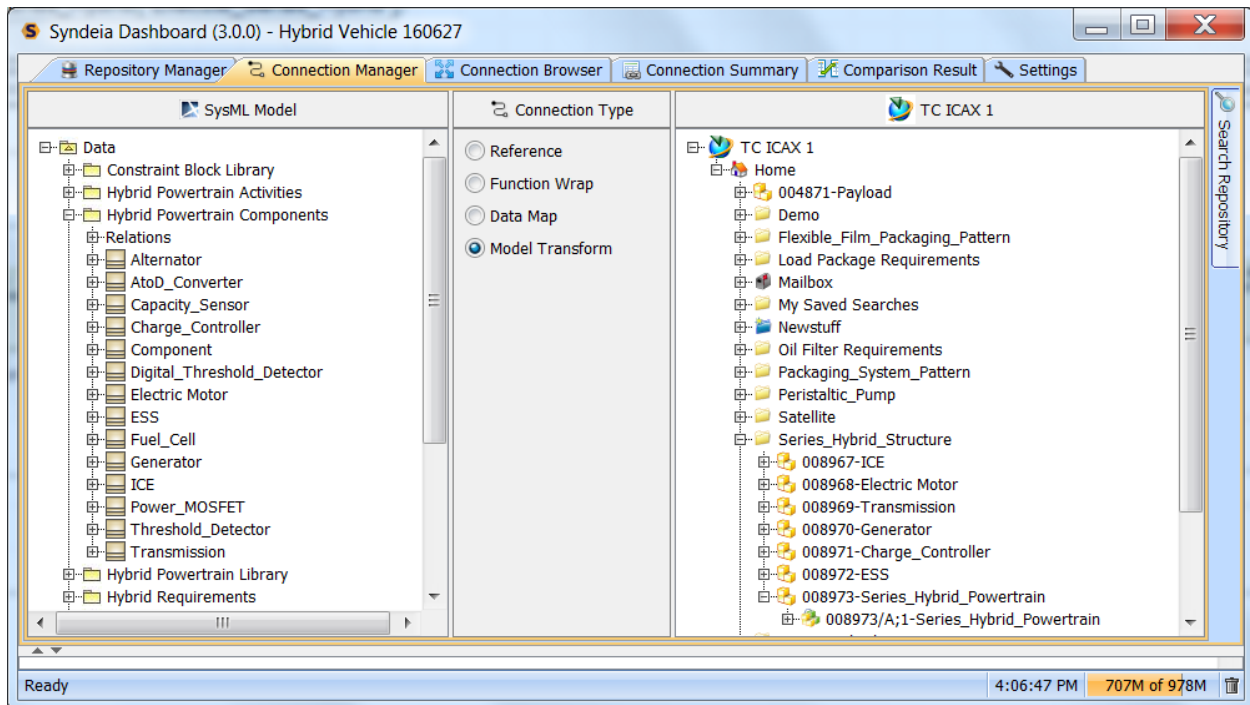

Figure 7 Syndeia dashboard showing SysML and Teamcenter PLM repository contents side by side.

In each case, the elements remain connected, linking the models in different tools into a single system model, the ultimate objective of MBE. Depending on the nature of the connection (and the capability of the tools involved), these connections can support use cases of practical value to engineering, including

- Compare and synchronize SysML block structure with PLM bill-of-materials or Simulink block structure as they evolve over time
- Update SysML requirements from a master requirements repository
- Open and view CAD models or project management issue trackers directly from the system model

Visualizing the Total System Model

As the number of connections grows, our ability to understand the scope and complexity of the total system model and to identify extended linkages between elements diminishes. We need to be able to

view the system model globally and to trace connections locally. Syndeia 3.0 offers both for automotive applications.

Figure 8 shows a global view of the inter-model connections in a chord plot. The upper right box reports that 73 elements are connected by 42 Syndeia-managed connections. The elements on the periphery of the circle represent SysML (orange), Teamcenter (red), Simulink (blue), GitHub (green) and JIRA (beige) elements. The nodes and the lines between them can be interactively highlighted and identified in the light blue box on the upper right, and the entire plot can be searched and filtered. Using this, the system engineer can see at a glance the types of elements being connected and the density of connections in the current state of the model.
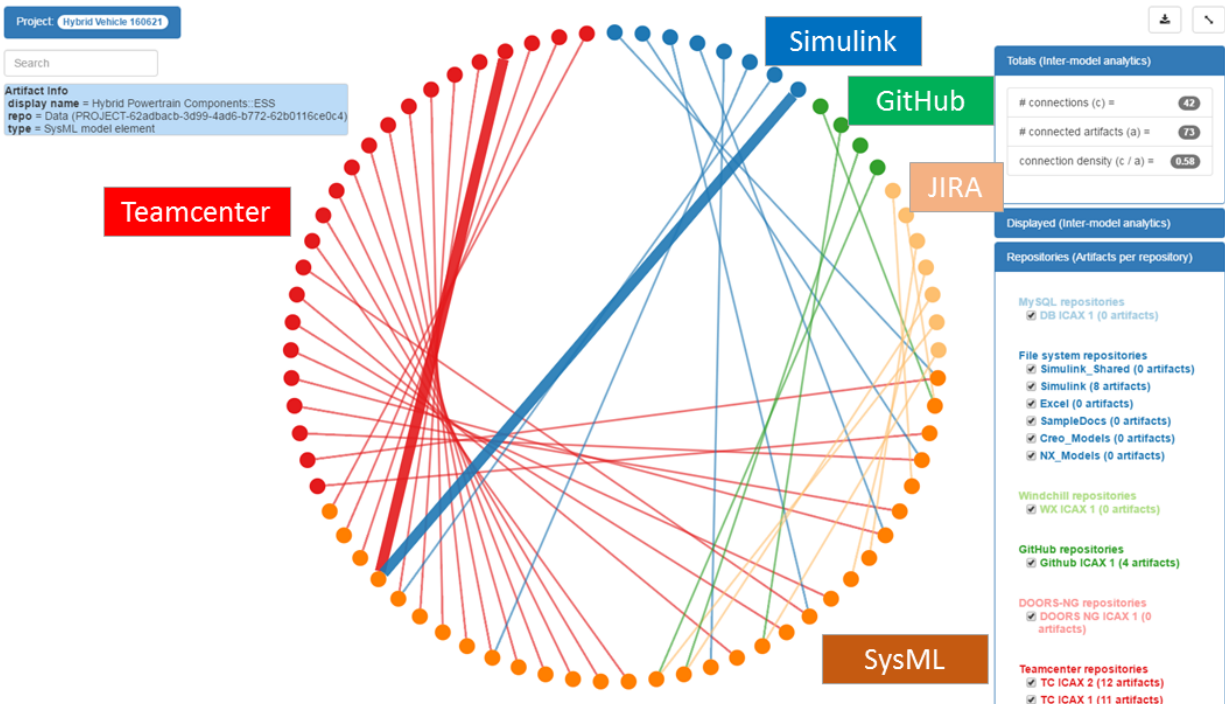

Figure 8 Chord plot of Inter-model Connections – Global Visualization

While the global view in Figure 8 gives the big picture, many use cases involve tracing extended linkages over both inter-model and intra-model chains. Alternate graphing techniques can make this task simpler. One example is shown in Figure 9, a tree plot initiated from a specific model element in the SysML model or the Syndeia dashboard. Launched from a SysML activity, Drive Generator, it immediately displays three nearest neighbors, a JIRA issue and a GitHub software folder connected by inter-model connections (shown in black) and a SysML block, ICE, connected by an intra-model connection (shown in orange). This last node can be further expanded by clicking on it, showing an additional four next nearest neighbors and the process can be continued selectively until the chain is terminated. In the resulting plot, the extended linkage between the Teamcenter requirement at the bottom left, the SysML function it impacts, and the JIRA issue that records that impact, is traced over five connections across three tools. Consider the value of these approaches in expediting design reviews, with explicit connections between system elements and the ability to open those elements in their native tools from a central high-level roadmap of the system.
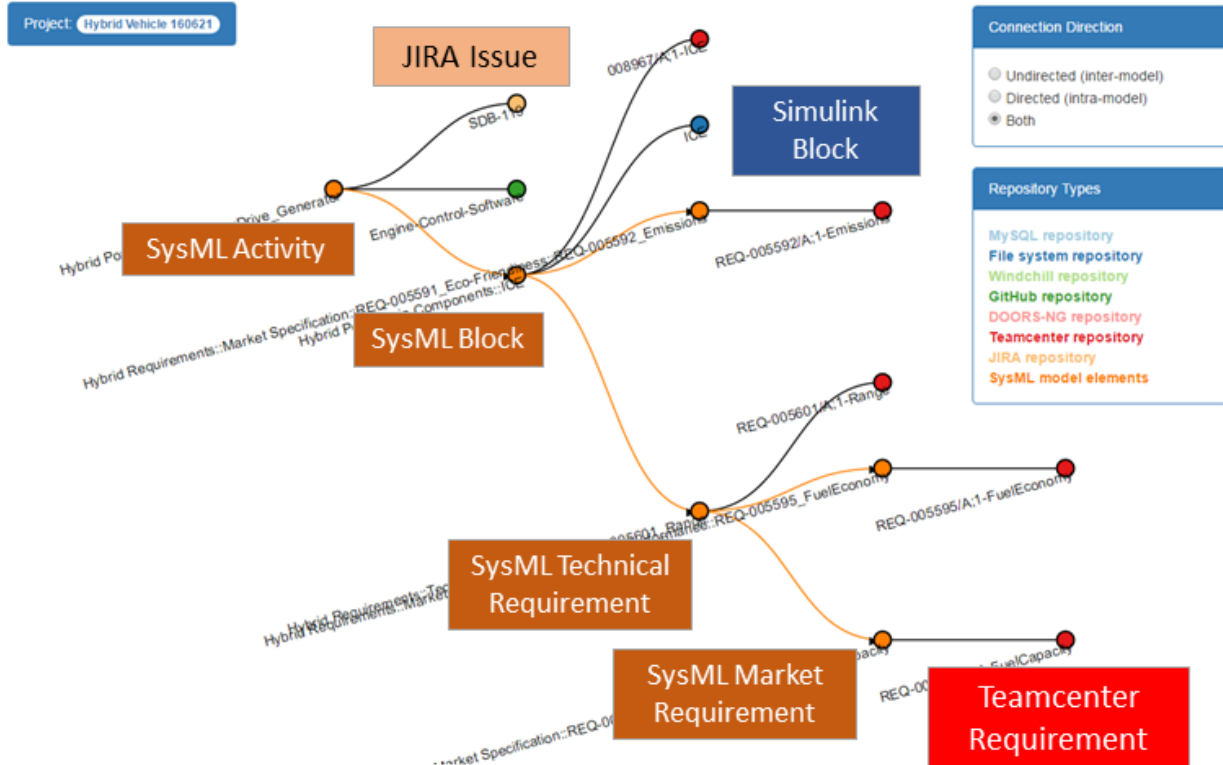
Figure 9 Tree plot of Inter-model and Intra-model Connections demonstrating traceability across graph

The Future of MBE for Automotive and Transportation Applications

The field of Model-Based Engineering is evolving rapidly. Driven by new technologies like the Internet-of-Things and Smart Grid energy systems, engineering software tools are being pushed to the limits. Automotive systems engineers, working on innovations like Autonomous Vehicles, will need to be at the forefront of these developments.

To meet their needs, MBE tools will need to evolve, as well. A SysML-centric architecture, as illustrated by Syndeia, will be no more effective than a PLM-centric or a CAD-centric one. MBE tools will function as web services, providing a complete picture of the Total System Model from any position in the engineering process. Powerful graph databases will provide scaleability and sophisticated query capability for identifying extended connections. Users can interact with the graph through sophisticated modeling tools or simplified web interfaces, using their choice of laptop, tablet or handheld device.

About the Author

Dr. Dirk Zwemer (dirk.zwemer@intercax.com) is President of Intercax LLC (Atlanta, GA), a supplier of MBE engineering software platforms like Syndeia and ParaMagic. He is an active teacher and consultant in the field and holds Level 4 Model Builder-Advanced certification as an OMG System Modeling Professional.

For further information, visit us at www.intercax.com or contact us at info@intercax.com.