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 Energy Engineering

Abstract

Model-based engineering (MBE), the creation and use of a single system model distributed over multiple tools and repositories, has promising applications in energy systems. In this paper, we present two examples from energy system design and analysis illustrating the advantages of an MBE approach. In the design of an energy generation facility, we show how the integration of multiple software tools, including PLM, CAD, simulation, requirements and project management, facilitates system development. In the analysis of a regional energy system, we show how an object-oriented approach linking modeling and simulation tools simplifies the integration of analysis models and the incorporation of technology advances, e.g. Smart Grid, Smart Home, in the analysis. We use InterCAX's MBE solutions, including Syndeia and ParaMagic, to demonstrate these concepts.

Sample SysML models may be downloaded from the Intercax website for Case Study 1 (for

MagicDraw, Rhapsody, and Enterprise Architect) and Case Study 2 (for MagicDraw).

Introduction

Systems engineering in the energy industry addresses some key industry drivers:

- Multidisciplinary technology encompassing mechanical, chemical, nuclear, electronic, software and other engineering disciplines
- Balancing economic and environmental factors
- High priority placed on energy system security and resilience to natural and man-made disruptions

Because of these challenges, energy system engineers have been early adopters of 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

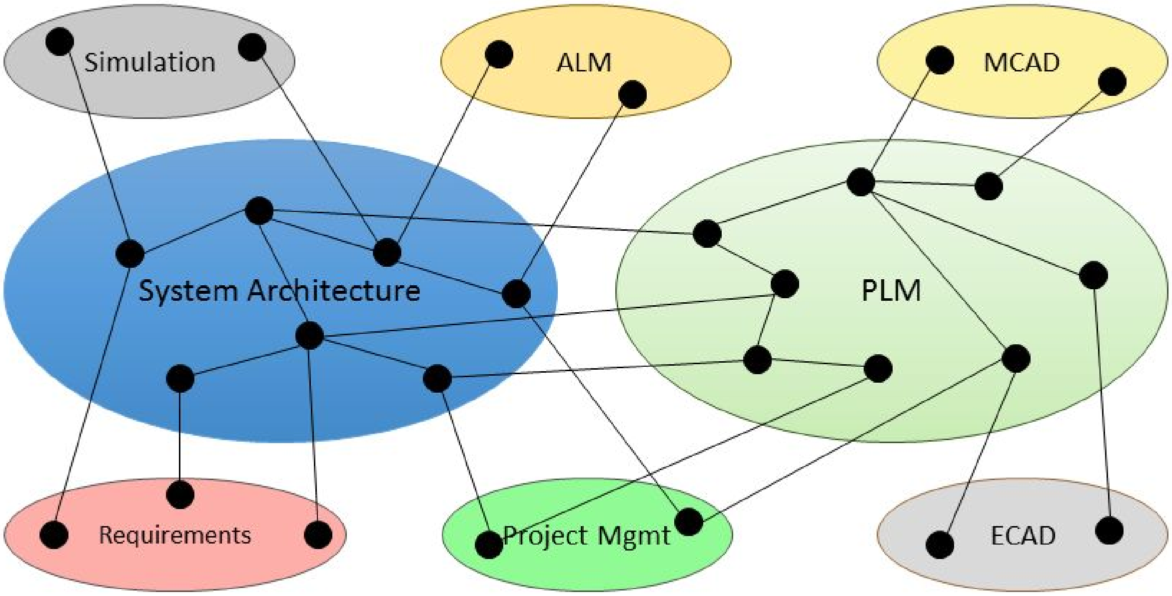


Figure 1 Total System Model as a Network of Connections inside and between Engineering Software Tools and Repositories

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.

Syndeia

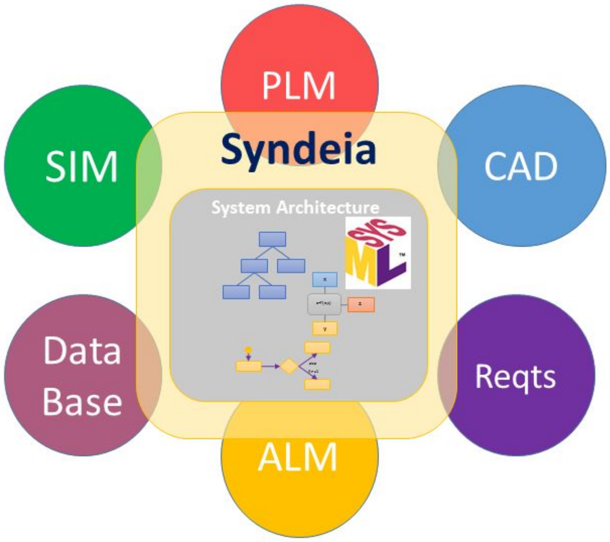


Figure 2 Syndeia architecture

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.

Case Study 1: Energy Generation System Design – Pressurized-Water Nuclear Reactor

Step 1 Setting up the Total System Model

Model-Based Engineering does not define a specific methodology for building the system model. For our example, we assume that it begins with assembling a list of requirement in a requirements management tool such as DOORS NG. Our first tasks are to create a SysML model that will become the hub of our system model, link it to the DOORS NG repository and drag the needed requirements from

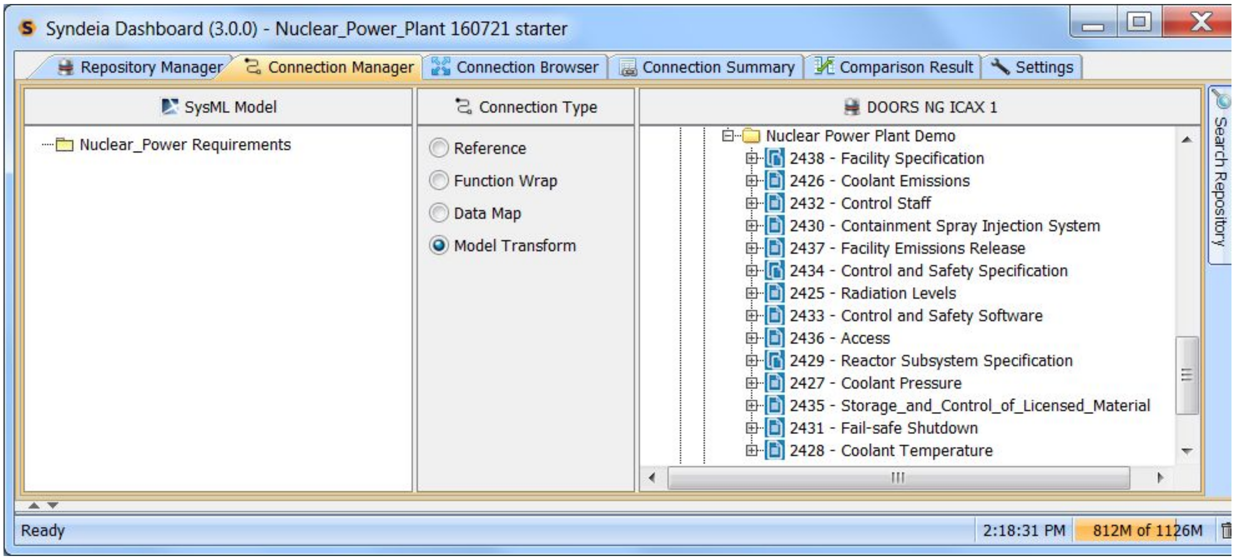


Figure 3 Syndeia dashboard for connecting SysML and DOORS NG models

DOORS into SysML. The Syndeia dashboard to create this model transform connection is shown in Figure 3. The connections created between the DOORS NG and SysML models allow the requirements to be compared and synchronized as the models evolve.

Step 2 Building out the System Architecture Model

Having brought the system requirements into the system architectural model for nuclear power plant, we choose to create a functional model in SysML, using a combination of state machine and activity diagrams. We begin the functional decomposition and generate lower-level activities, for example, Generate_High_Pressure_Steam in Figure 4. We also begin to allocate these activities to structural elements, represented by the swim lanes in the activity diagram.

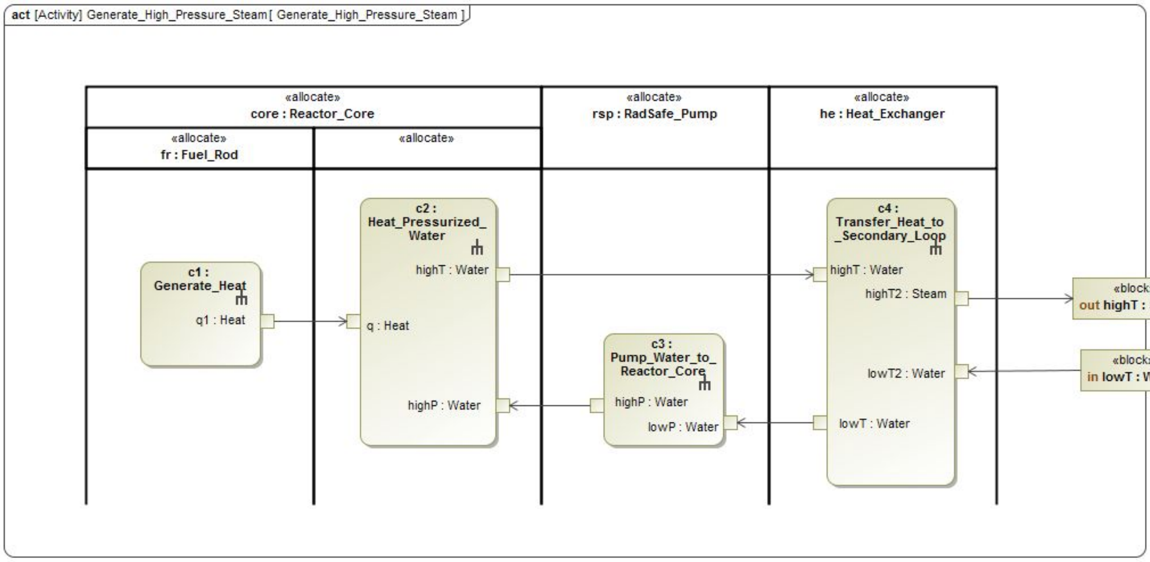


Figure 4 SysML Activity diagram for Generate_High_Pressure_Steam, with allocations to structural components

A high-level view of the structure in SysML, comprising the major subsystems of the generator facility, including reactor, turbine, cooling towers and generator, is shown in Figure 5.

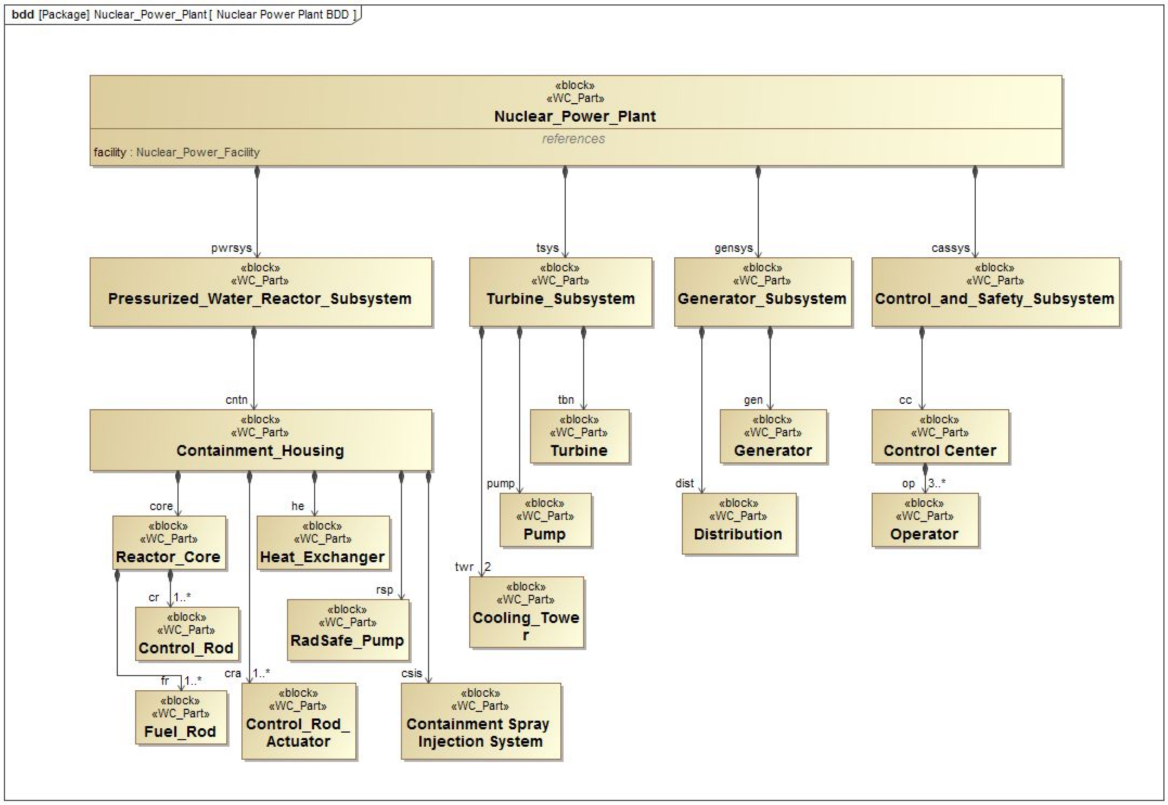


Figure 5 Vehicle Structural Decomposition, including Powertrain variants

A second view of the same SysML model in Figure 7 shows the control software and sensor network of the facility.

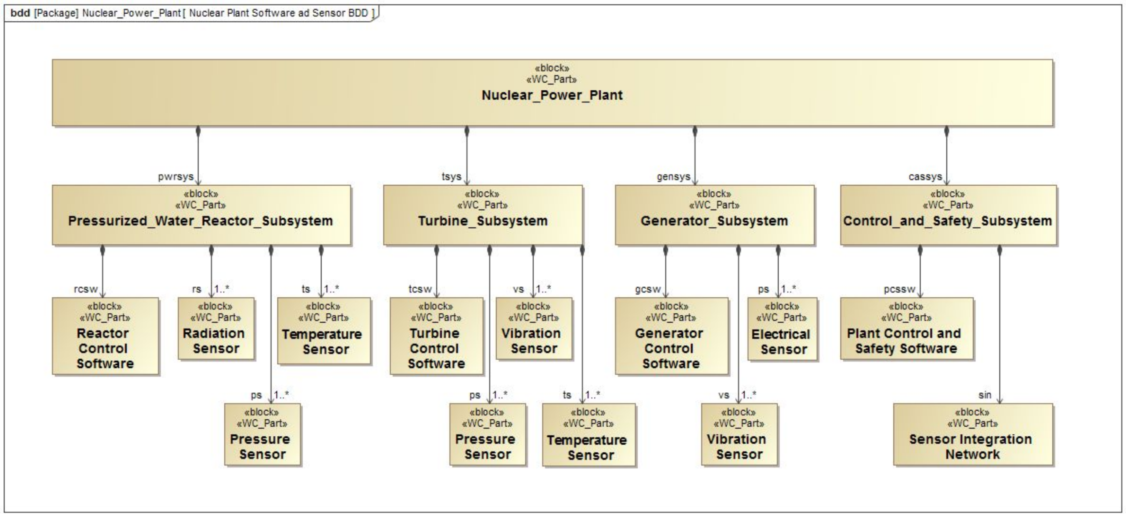


Figure 6 Block definition diagram showing the

As the functional and structural models are developed, we create relationships between those elements and the requirements brought into the system architecture model in the initial step. In Figure 7, we show some of the <<satisfy>> dependencies identified between requirements and structural elements.

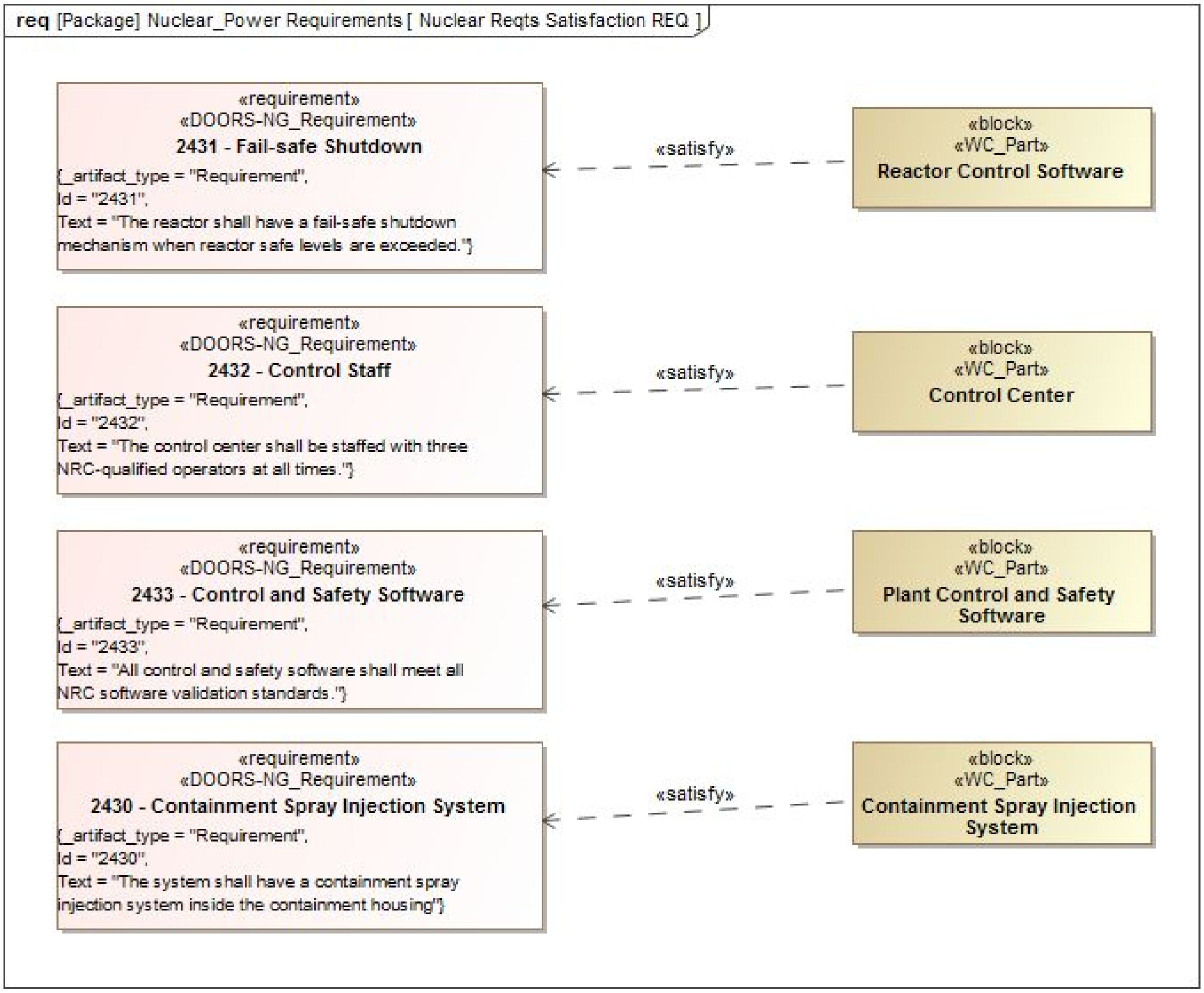


Figure 7 Sample SysML connectivity between requirements and structure

All the relationships shown in the last four 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.

Step 3 Inter-model Connections

As the project engineering progresses, many new software tools are brought into the workflow. Coordinating the efforts of a diverse set of engineering disciplines and tools becomes a significant challenge. MBE seeks to facilitate connection between engineering models so that their efforts integrate efficiently and the single system architecture model provides an up-to-date picture of the entire system for all the engineers and project managers.

Inter-model connections are made using the Syndeia dashboard (Figure 8) using a variety of drag-and-drop mechanisms. In our sample model,

- Structure element block structures were brought into the PLM system (PTC Windchill) as shown in the Syndeia dashboard in Figure 8.

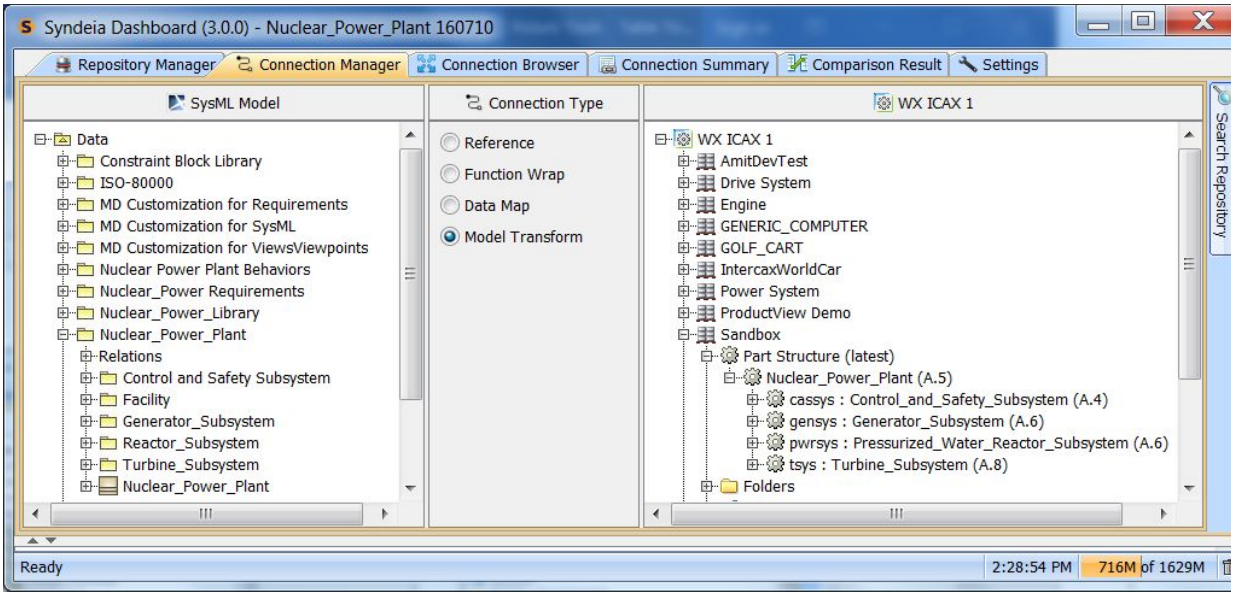


Figure 8 Syndeia dashboard showing SysML and Windchill PLM repository contents side by side.

- Functional (e.g. Figure 4) and structural connectivity were dragged from SysML into MATLAB, creating a Simulink block structure (blocks, ports and signals, but no internal MATLAB code);
- Reference links were created from SysML software elements (e.g. Reactor Control Software in Figure 6 to software folders in GitHub for software version management.
- A Model Transform connection is created to a CAD model (Siemens NX) of the facility (Figure 9)
- The functional activity hierarchy in SysML is used to generate an issue hierarchy in JIRA, a project management system, where status, recent activity, schedule and personnel are tracked.

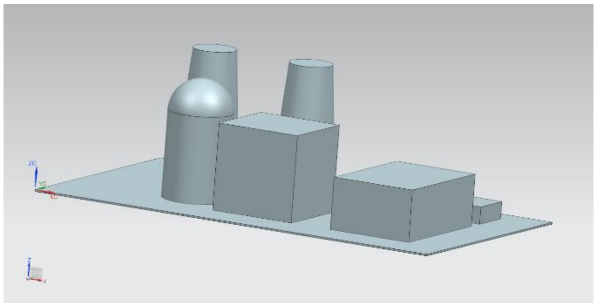


Figure 9 CAD model, power generation facility

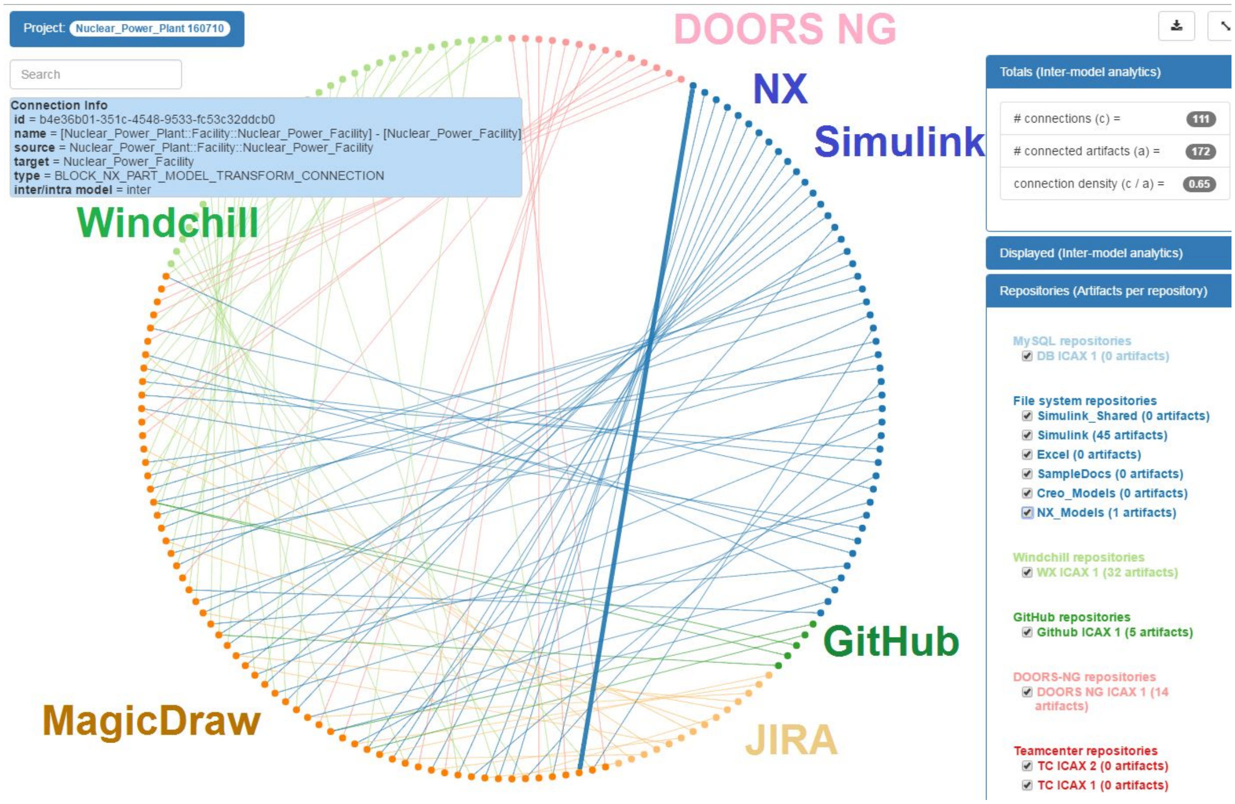
In each case, the elements remain connected, linking the models in different tools into a single system model, the ultimate objective of MBE. Depending on the nature of the connection (and the capability of the tools involved), these connections can support use cases of practical value to engineering, 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
- Import CAD parameters from design files into the system model for requirements verification
- View the latest file versions in system software in GitHub

Step 4 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 the energy industry both.

Figure 10 shows a global view of the inter-model connections in a chord plot. The upper right box reports that 172 elements are connected by 111 Syndeia-managed connections. The elements on the periphery of the circle represent SysML (orange), PTC Windchill (green), Simulink and NX (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.



Connection Info
id = b4e36b01-351c-4548-9533-fc53c32ddcb0
name = [Nuclear_Power_Plant::Facility::Nuclear_Power_Facility] - [Nuclear_Power_Facility]
source = Nuclear_Power_Plant::Facility::Nuclear_Power_Facility
target = Nuclear_Power_Facility
type = BLOCK_NX_PART_MODEL_TRANSFORM_CONNECTION
inter/intra model = inter

connections (c) = 111
connected artifacts (a) = 172
connection density (c / a) = 0.65

MySQL repositories
- ☑ DB ICAX 1 (0 artifacts)

File system repositories
- ☑ Simulink_Shared (0 artifacts)
- ☑ Simulink (45 artifacts)
- ☑ Excel (0 artifacts)
- ☑ SampleDocs (0 artifacts)
- ☑ Creo_Models (0 artifacts)
- ☑ NX_Models (1 artifacts)

Windchill repositories
- ☑ WX ICAX 1 (32 artifacts)

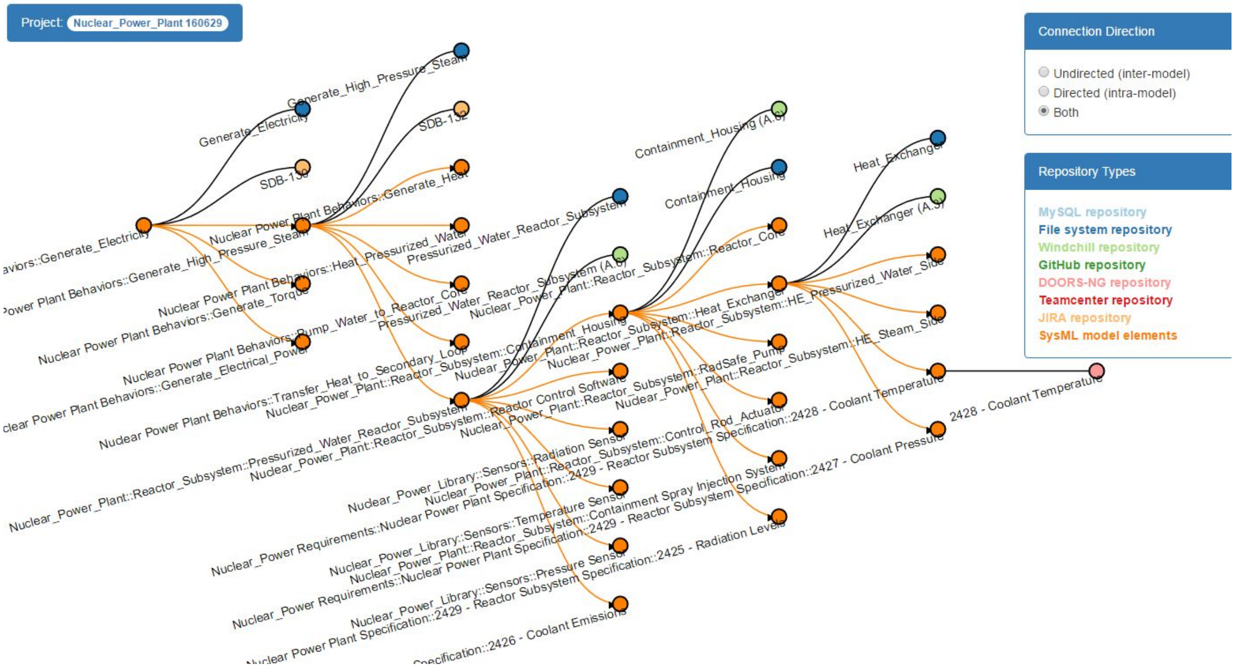
GitHub repositories
- ☑ Github ICAX 1 (5 artifacts)

DOORS-NG repositories
- ☑ DOORS NG ICAX 1 (14 artifacts)

Teamcenter repositories
- ☑ TC ICAX 2 (0 artifacts)
- ☑ TC ICAX 1 (0 artifacts)

Figure 10 Chord plot of Inter-model Connections – Global Visualization

While the global view in Figure 10 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 11, a tree plot initiated from a specific model element in the SysML model or the Syndeia dashboard. Launched from a SysML activity, Generate Electricity, it immediately displays five nearest neighbors, a JIRA issue and a Simulink block connected by inter-model connections (shown in black) and three SysML sub-activities, ICE, connected by an intra-model connection (shown in orange). One of these last nodes can be further expanded by clicking on it, showing an additional seven next nearest neighbors and the process can be continued selectively until the chain is terminated. In the resulting plot, the extended linkage between the DOORS requirement at the far right, the SysML function it impacts, and the JIRA issue that records that impact, is traced over seven 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


Project: Nuclear_Power_Plant 160629

Connection Direction

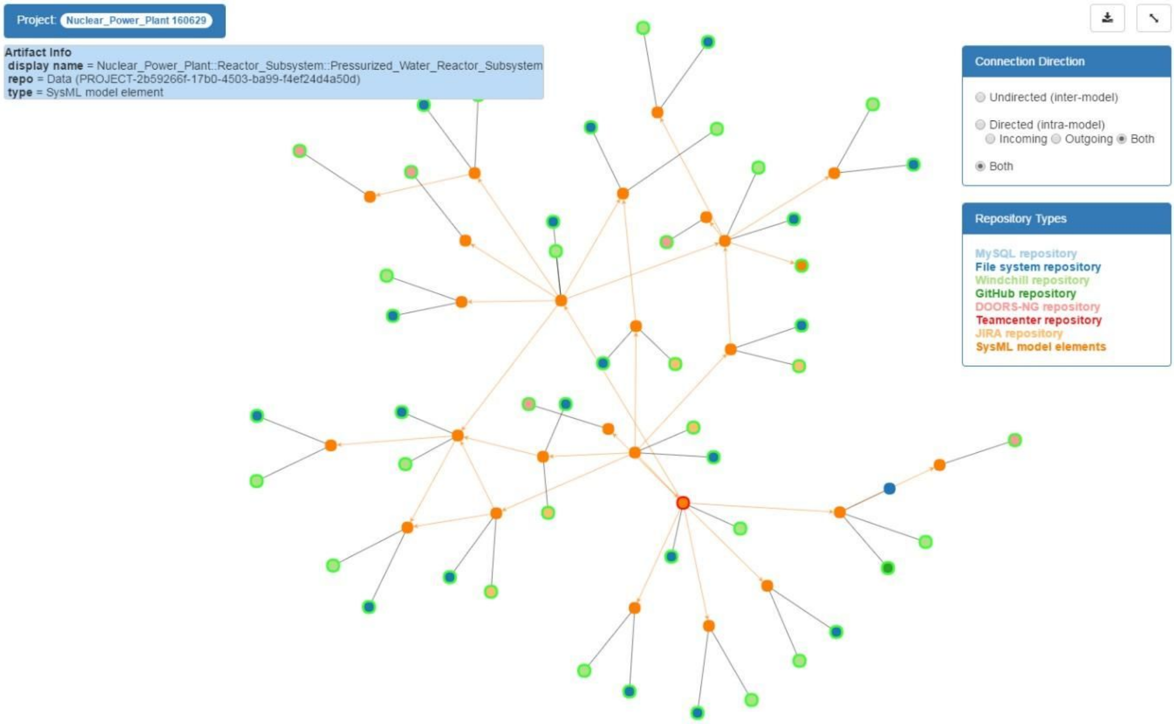
○ Undirected (inter-model)
○ Directed (intra-model)
● Both

Repository Types

MySQL repository
File system repository
Windchill repository
GitHub repository
DOORS-NG repository
Teamcenter repository
JIRA repository
SysML model elements

Figure 11 Tree plot of Inter-model and Intra-model Connections demonstrating traceability across graph

An alternate plot from the same model using an autolayout routine gives an impression of the complexity of connections in even a simple model when fully expanded. Efficient and easy-to-use query features, as described in the last section, are needed to fully utilize the power of MBE.


Project: Nuclear_Power_Plant 160629

Artifact Info
display name = Nuclear_Power_Plant::Reactor_Subsystem::Pressurized_Water_Reactor_Subsystem
repo = Data (PROJECT-2b59266f-17b0-4503-ba99-f4ef24d4a50d)
type = SysML model element

Connection Direction

○ Undirected (inter-model)
○ Directed (intra-model)
○ Incoming ○ Outgoing ● Both
● Both

Repository Types

MySQL repository
File system repository
Windchill repository
GitHub repository
DOORS-NG repository
Teamcenter repository
JIRA repository
SysML model elements

Figure 12 Autolayout plot of inter-model and intra-model connections demonstrating traceability across graph

Case Study 2: Energy System Analysis

The first case study was primarily about design and touched only briefly on analysis, for example, transforming aspects of the system architectural model into the skeleton of a Simulink simulation model. However, analysis is a critical part of MBE and should be incorporated into the system model as early as possible

- To evaluate possible system variants
- To explore system risk
- To verify requirements on an ongoing basis

Generally, complex systems require many different kinds of analytical models and, as such systems are often systems-of-systems, analytical models applied to separate subsystems must be confederated to analyze the total system.

In this case study, we look at how some of these issues play out in modeling a regional energy system. In Figure 13, we consolidated several analytical models into a single structure in SysML.

- The power plant models calculate cost of power generation as a function of capital, fixed operating and variable operating costs, and an environmental cost associated with emissions (solid, liquid, gaseous and capacity build). These blocks could contain models like the one developed in the case study above.

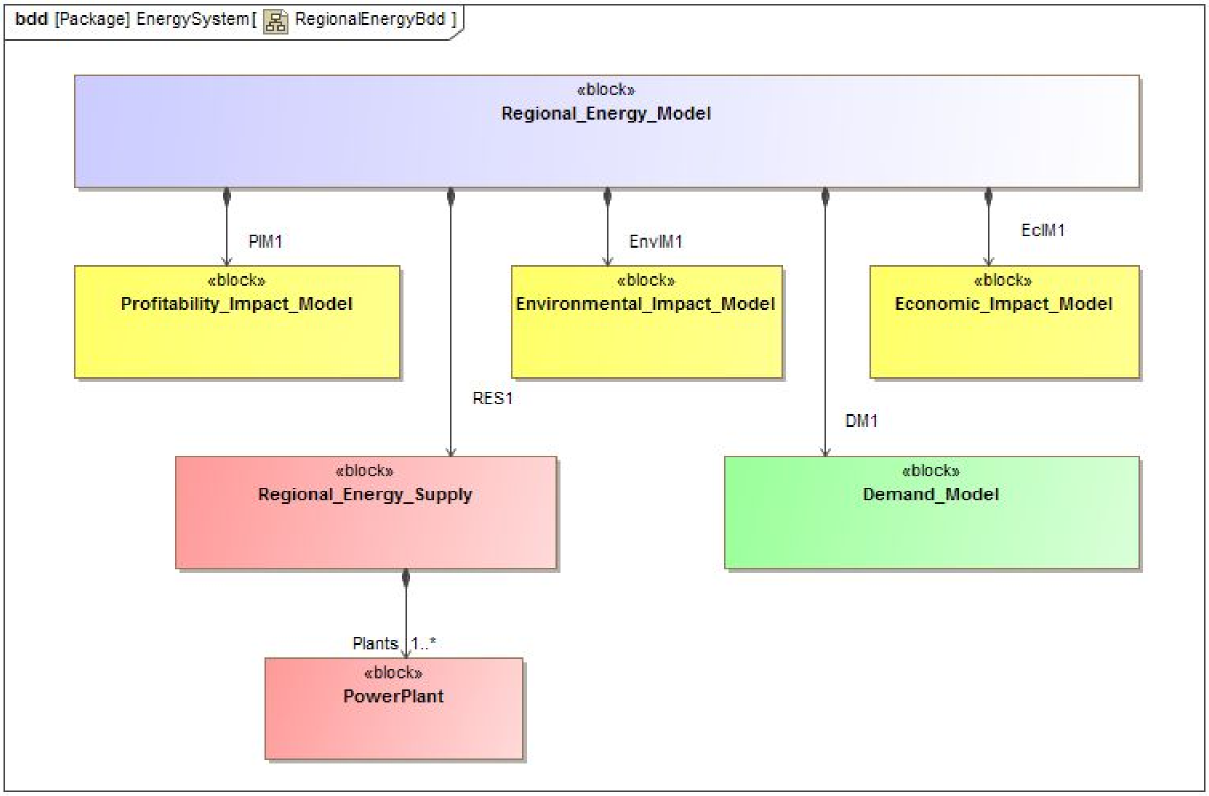


Figure 13 Regional Energy Model block definition diagram, Phase 1

- A Regional Energy Supply model that consolidates the supply models and calculates weighted average costs
- An aggregate Demand model based on standard price elasticity
- Profit, Economic Impact and Environmental Impact models that evaluate the effect of supply and demand values on the utility and the region.

In our study, these models used SysML parametric diagrams to connect parameters and constraints. Some of these models use MATLAB and Mathematica functions. Common parameters such as supply (MW), demand (MW), cost($/kW-hr) and price ($/kW-hr) are exchanged through the top-level Regional_Energy_Model block.

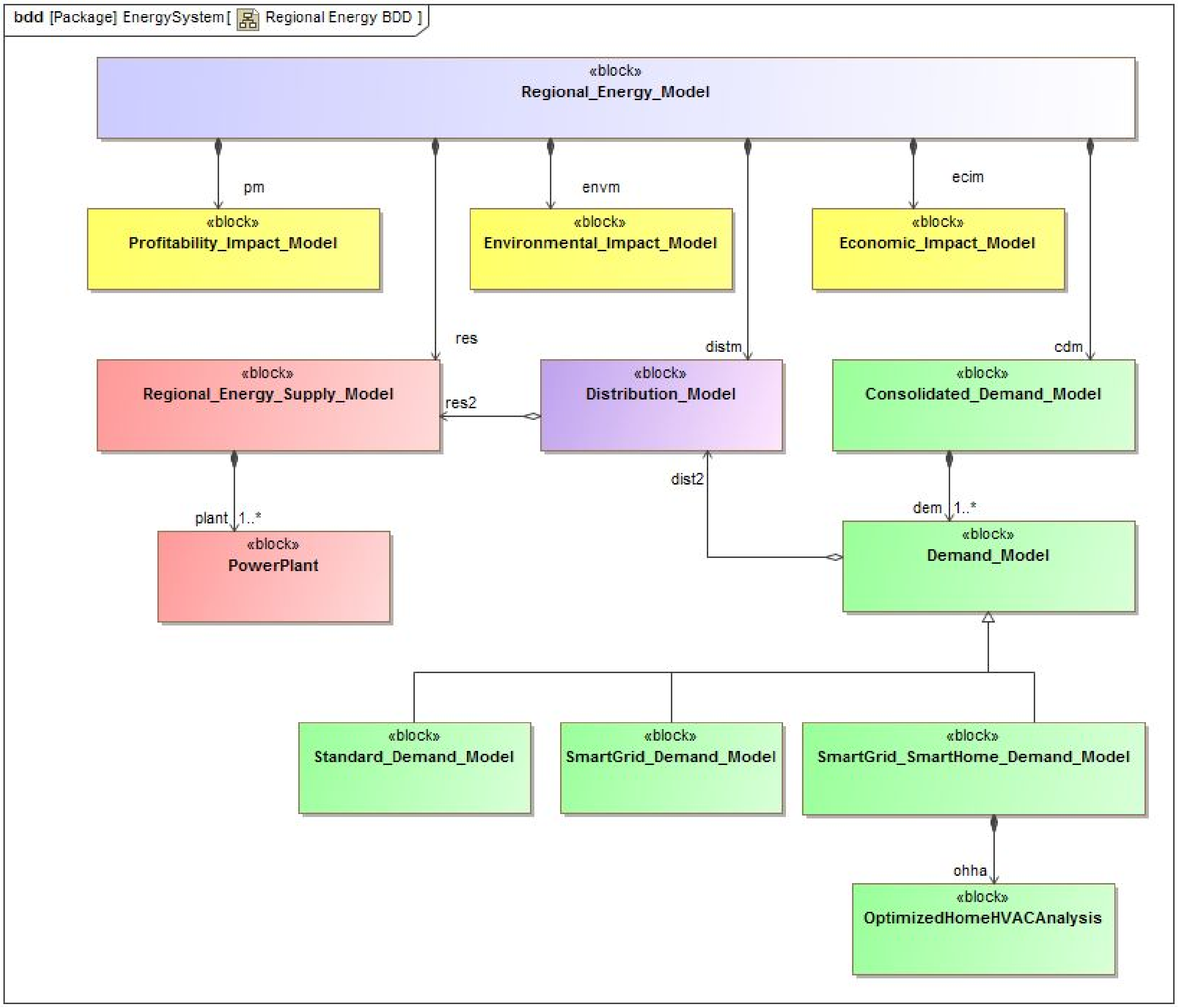


Figure 14 Regional Energy Model block definition diagram, Phase 2

In other projects, Intercax has developed other SysML analysis models evaluating, for example, Smart Grid and Smart Home systems. In the Smart Grid model, the utility distribution system has the built to provide price information to consumers, some of which have the ability to shift demand to low price periods during the day. In the Phase 2 diagram in Figure 14, we add new blocks to the models, including the Distribution and SmartGrid Demand models, from that project.

The SmartHome project explored the use of the “Internet of Things” to reduce demand in a networked residential user, by, for example, reducing HVAC activity when parts of the home were unoccupied. The SysML reference model for the SmartHome, part of which is shown in Figure 15, had its own analysis

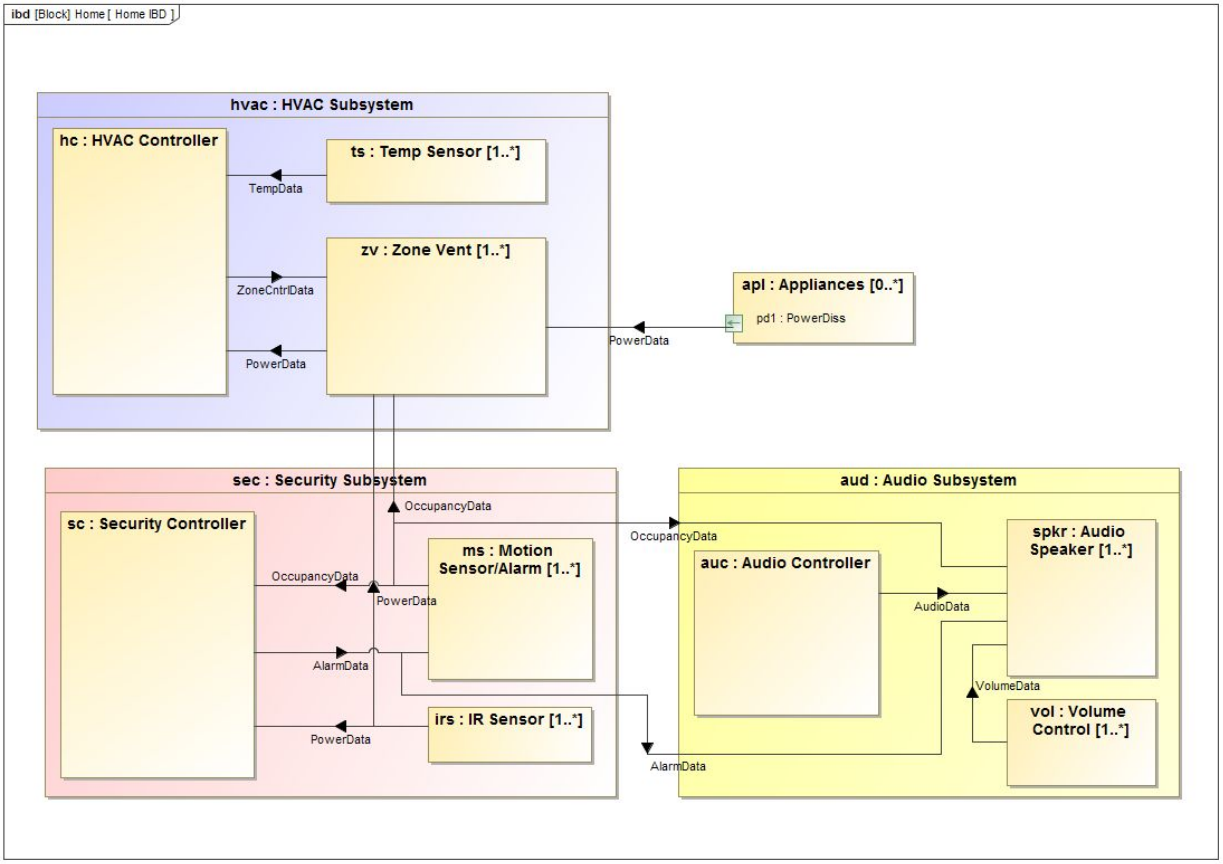


Figure 15 SmartHome interconnection network between HVAC, Security and Audio subsystems in SysML IBD

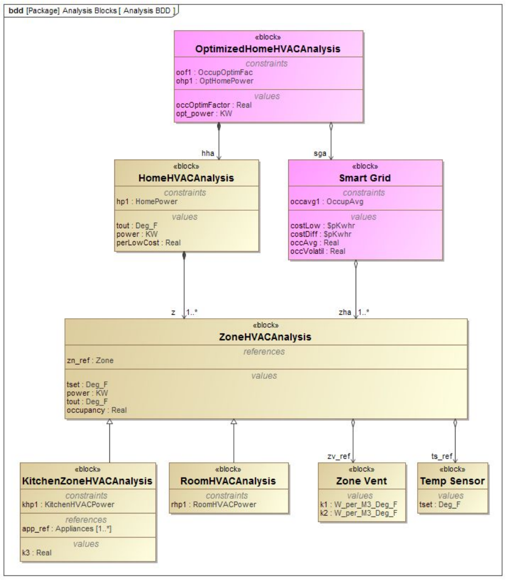


Figure 16 SmartHome analysis models

models, as shown in Figure 16. The top-level analysis block is used directly by the SmartGrid_SmartHome_ Demand model at the bottom right in Figure 14.

When the model in Figure 14 is instantiated, we can readily choose a combination of different sources and consumers. In one example, we chose a combination of three generation facilities (coal-fired, nuclear and solar), each with time-dependent supply and cost values (considered in one-hour intervals over a twenty-four hour cycle). We chose three different combinations of demand models, with a constant base demand schedule underlying different levels of price elasticity, demand shift elasticity and demand reduction, depending on their adoption of SmartGrid and SmartHome technologies.

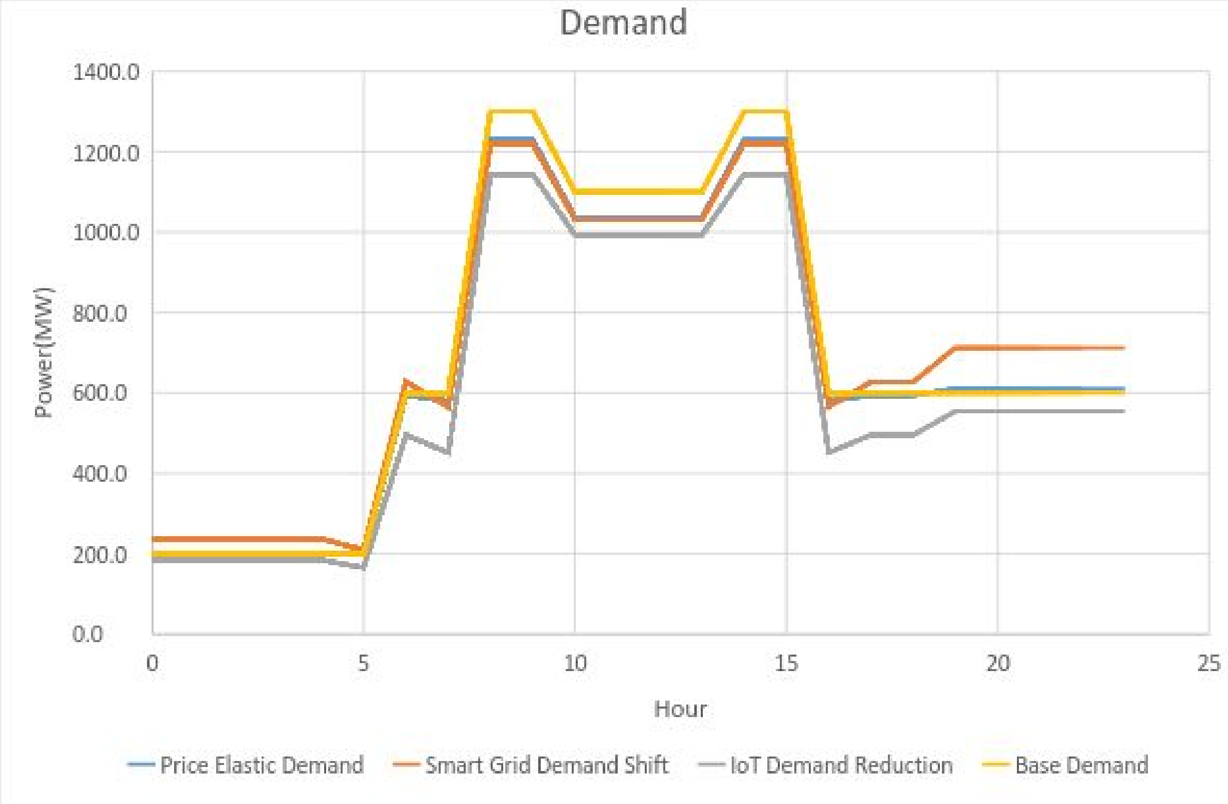


Figure 17 Demand curve shifts reflecting price elasticity, demand shift and demand reduction

Figure 17 shows the results of these calculations by hour of the day for a combination of industrial and residential demand. Price elasticity reduces demand, especially at the most expensive midday rates when consumption is highest and demand shift elasticity has the highest impact for residential customers in the late night and early morning hours, but relatively little for industrial demand which is confined to midday hours in our model.

The objective of this exercise is not to claim that the specific models used here reflect accurate usage predictions, but to show how readily an MBE approach with an object-oriented language like SysML allows federation of diverse multidisciplinary models.

The Future of MBE for Energy 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. To meet their needs, MBE tools will need to evolve as well. A SysML-centric architecture, as illustrated by the current Syndeia, will be no more uniquely 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.