



Modeling Tool Integration Plugins (MTIP)

January 31, 2022

Trent Severson

The Aerospace Corporation

Karina Martinez

The Aerospace Corporation

Approved for public release. OTR 2022-00282.

Project MTIP Team Introduction



Karina Martinez

- Software Tools and Assurance Department
- SparxEA tool developer



Trent Severson

- MBSE Office
- Cameo tool developer



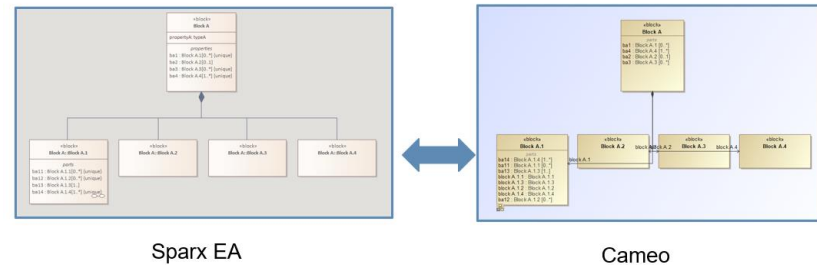
Joel Thomas

- MSD
- Cameo tool developer



Agenda

- Definitions
- Problem Overview
- Approach to MBSE Tool Integration
- Examples
- Code Base and Usages
- Demo
- What's Next?



CAMEO
SYSTEMS MODELER™



Definitions: Ontology, Metamodel, SysML, UML



- Ontology
 - *Defines entities and relationships*
 - *Helps introduce a sharable and reusable knowledge but can also add new knowledge about the domain.*
- Metamodel
 - *A set of rules that define valid construction of a model*
 - *Allows a model to be checked for syntactic correctness using rules and constraints defined by the metamodel*
- Systems Modeling Language (SysML) Metamodel
 - *A general-purpose graphical modeling language for specifying, analyzing, designing and verifying complex systems*
 - *Provides graphical representation with a semantic foundation for modeling system requirements, behavior, and structure.*
- Unified Modeling Language (UML)
 - *A standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing and documenting the artifacts of systems.*

Definitions: Cameo, Sparx EA, XML, XMI



- Cameo Systems Modeler
 - *A collaborative Model-Based Systems Engineering (MBSE) environment, which provides robust, and intuitive tools to define, track, and visualize all aspects of systems in the most standard-compliant SysML models and diagrams.*
- Sparx Enterprise Architect (EA)
 - *A collaborative modeling design and management platform based on UML and related standards*
- eXtensible Markup Language (XML)
 - *A markup language designed to store and transport data.*
- XML Metadata Interchange (XMI)
 - *An XML-based integration framework for the exchange of models, and, more generally, any kind of XML data.*
 - *Used in the integration of tools, repositories, applications, and data warehouses.*
- Huddle Unified Data Schema (HUDS)
 - *Allows data to be easily shared between different plugins while still adhering to a custom set of semantics.*
 - *More human readable than XMI*

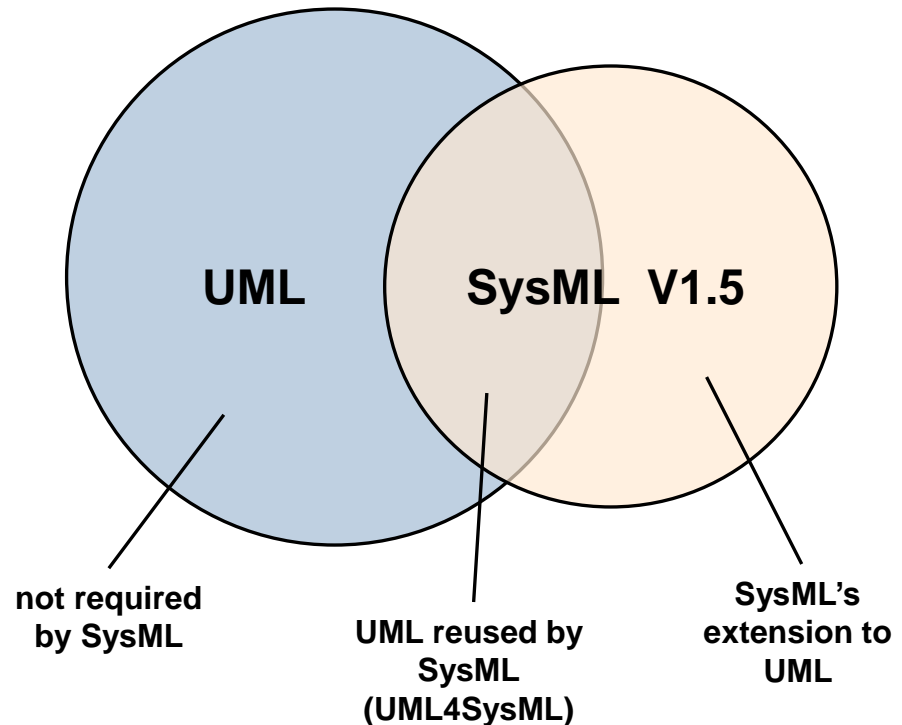


Integration of MBSE Tools

Problem Overview

Cameo and Sparx EA Interoperability

- Commercial tools are not sufficient to work with models that do not fall within UML standards
- XMI import/export is the current standard interchange format between MBSE tools.
- XMI format adheres to UML standard and does not account for additional architectures.
- SysML specific data is not transferred between tools
 - *Only UML4SysML elements are transferred correctly*



Relationship Between SysML and UML[1]

Current XMI standard insufficient for transferring SysML models



Sparx EA \leftrightarrow Cameo

Approach to Tool Integration for SysML Metamodel

- **Project Goal**
 - Create software plugins and exchange standards to enable 1-to-1 translation from Sparx EA to Cameo and back
 - Create a way for SysML models to be exchanged freely between tools
- **Implementation Approach**
 - Metamodel Mapping
 - Map SysML metamodel in Sparx EA and Cameo to a common schema
 - Data imported/exported via plugins using Huddle Unified Data Schema (HUDS) V2 XML format
- **Current Capabilities**
 - Transfers SysML models from Sparx EA into Cameo and back
 - Includes all elements, relationships and diagrams

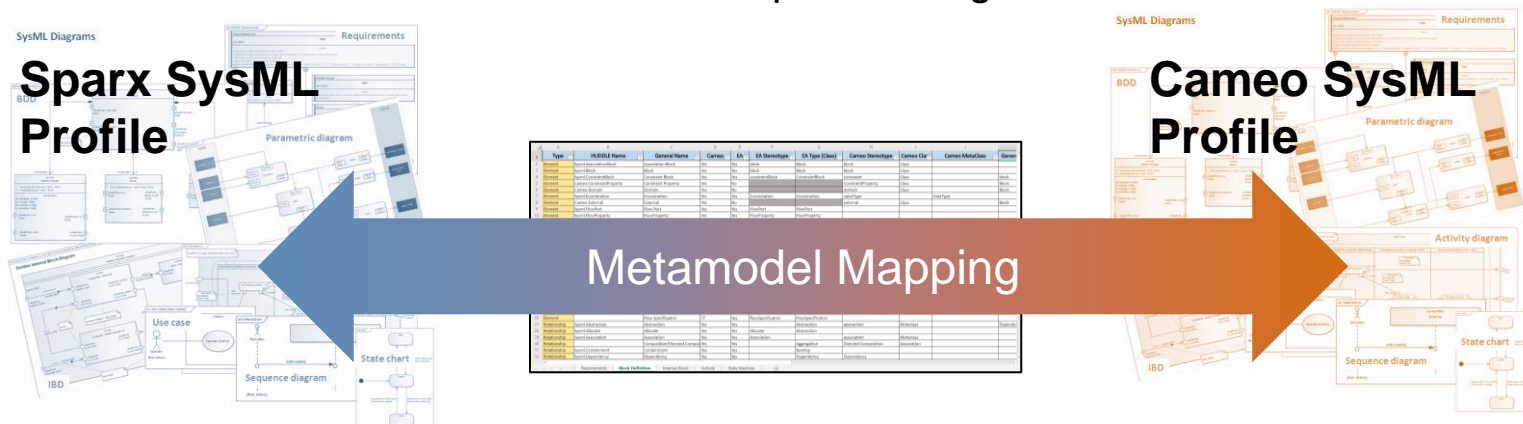
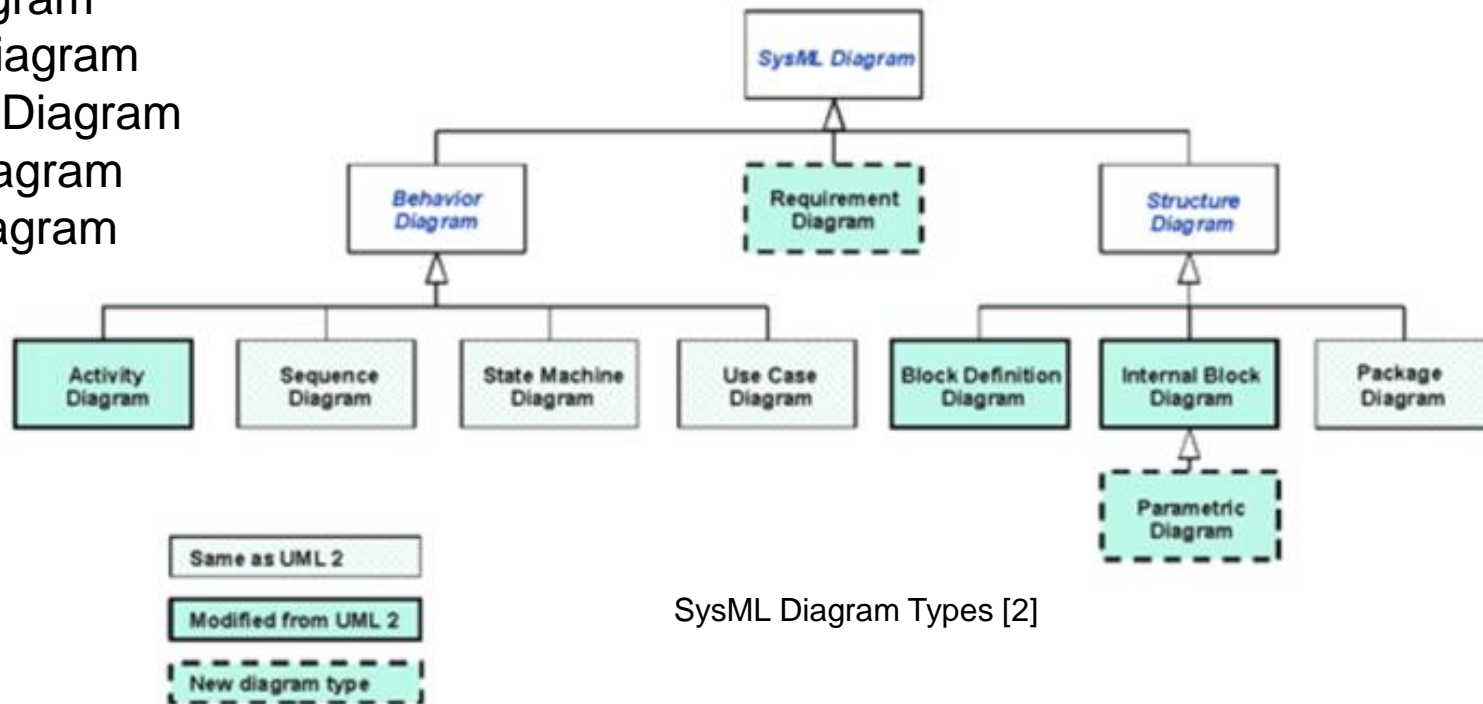


Diagram Types

Organization of the SysML Metamodel

- Block Definition Diagram
- Internal Block Diagram
- Activity Diagram
- State Machine Diagram
- Package Diagram
- Parametric Diagram
- Requirement Diagram
- Sequence Diagram
- Use Case Diagram



SysML Diagram Types [2]

Graphic nodes and paths for diagram types listed above are supported in MTIP plugin translation

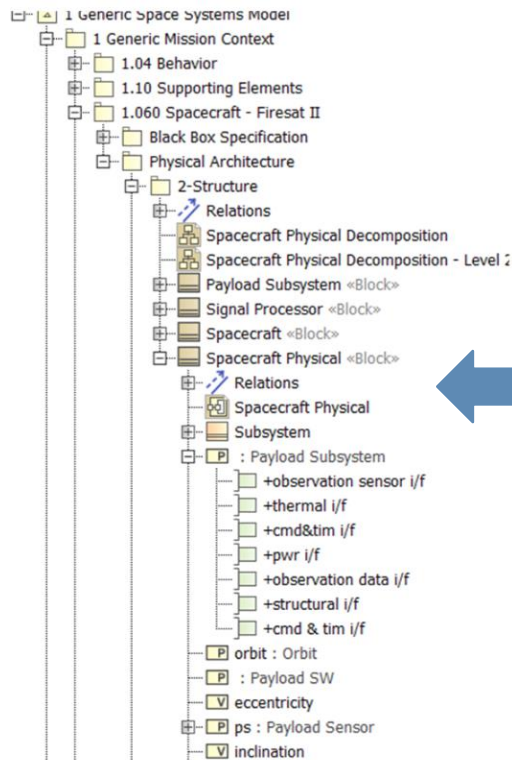
HUDS V2 XML Format

Cameo \leftrightarrow HUDS V2 XML \leftrightarrow Sparx EA

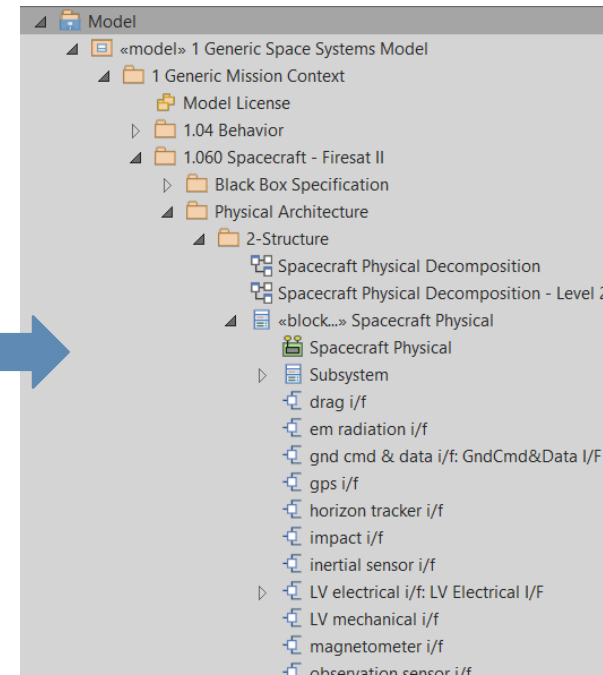
Cameo

HUDS XML

Sparx EA



```
<data>
  <relationships _dtype="dict">
    <hasParent _dtype="dict">
      <type _dtype="str">sysml.Package</type>
      <id _dtype="str">_19_0_4_96401ee_1638823247540_971842_42624</id>
      <relationship_metadata _dtype="dict"/>
    </hasParent>
  </relationships>
  <id _dtype="dict">
    <cameo _dtype="str">_19_0_4_96401ee_1638823247540_313539_42625</cameo>
  </id>
  <attributes _dtype="dict">
    <attribute _dtype="dict" key="name">
      <attribute _dtype="str" key="value">Spacecraft Physical</attribute>
    </attribute>
    <attribute _dtype="dict" key="stereotype">
      <attribute _dtype="dict" key="stereotypeName">
        <attribute _dtype="str" key="value">Block</attribute>
      </attribute>
      <attribute _dtype="dict" key="stereotypeId">
        <attribute _dtype="str" key="value">_11_5EAPbeta_be00301_1147424</attribute>
      </attribute>
      <attribute _dtype="dict" key="profileName">
        <attribute _dtype="str" key="value">SysML</attribute>
      </attribute>
      <attribute _dtype="dict" key="profileId">
        <attribute _dtype="str" key="value">_11_5EAPbeta_be00301_1147434</attribute>
      </attribute>
    </attribute>
  </attributes>
  <type _dtype="str">sysml.Block</type>
</data>
```



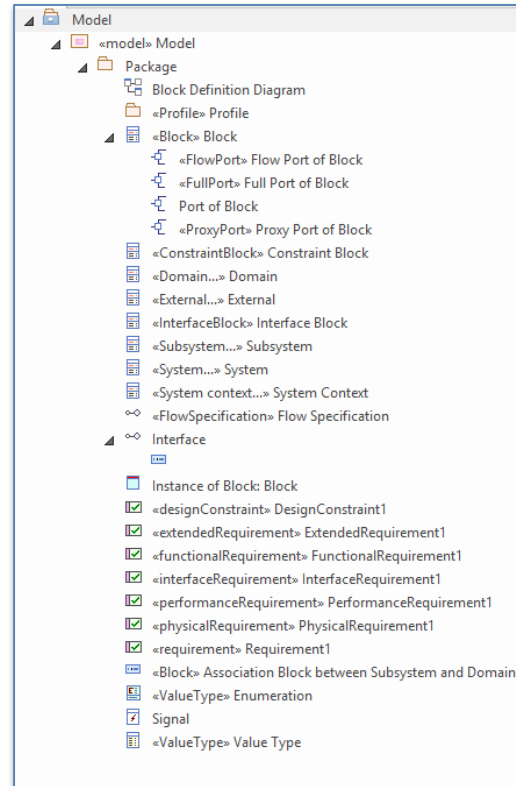
Each element has its own data block. Each data block consists of type, id, attributes, and relationships required during MTIP plugin translation

Block Definition Diagram Package Structure

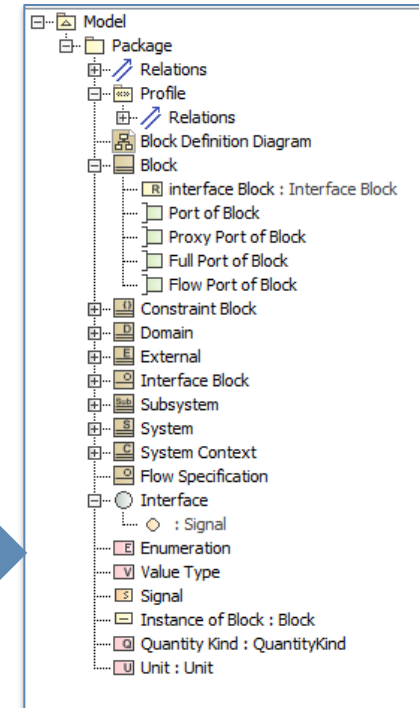


- Diagrams
 - *Block Definition Diagram*
- Elements
 - *Port*
 - *Full Port*
 - *Proxy Port*
 - *Flow Port*
 - *Interface Block*
 - *Flow Specification*
 - *Constraint Block*
 - *Domain*
 - *Subsystem*
 - *External*
 - *System*
 - *System Context*
 - *Value Type*
 - *Enumeration*
 - *Signal*
 - *Instance Specification*
 - *Interface*
 - *Property*
- Relationships
 - *Interface Realization*
 - *Generalization*
 - *Association Block*
 - *Directed Association*
 - *Directed Aggregation*
 - *Directed Composition*
 - *Association*
 - *Item Flow*
 - *Usage*

Sparx EA



Cameo

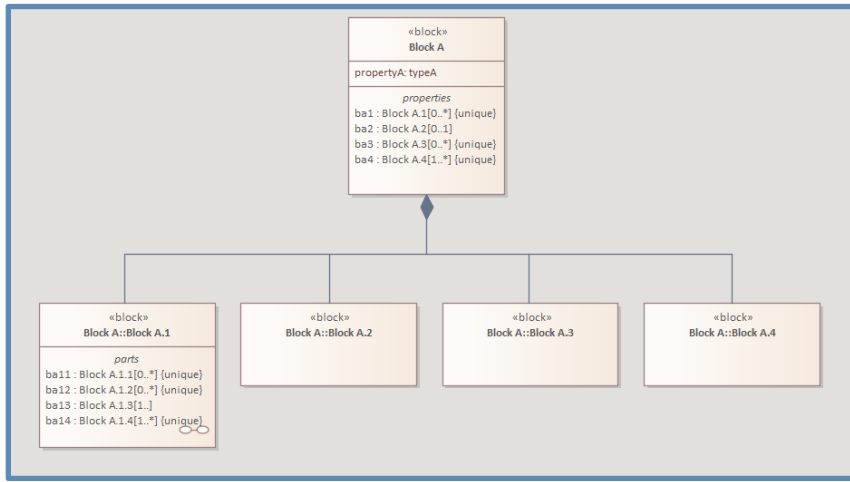


Side by side comparison of project browser/containment tree of a translated model in EA (left) and Cameo (right)

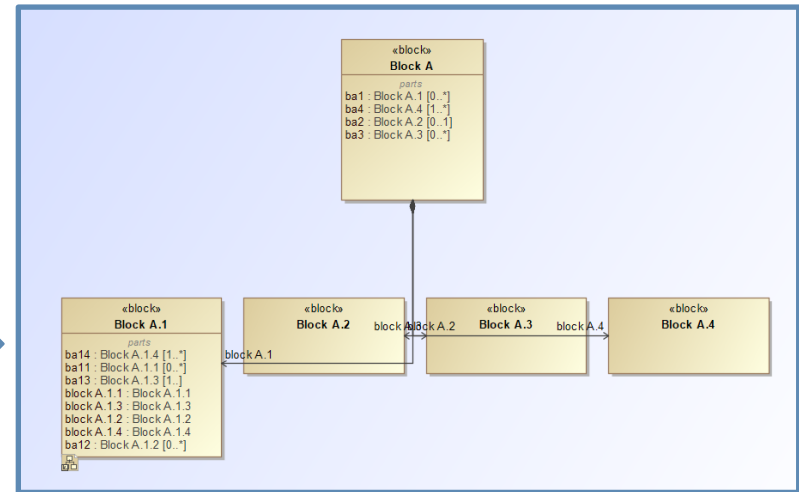
Block Definition Diagram

Sparx EA \leftrightarrow Cameo

- Block Definition Diagram



Sparx EA

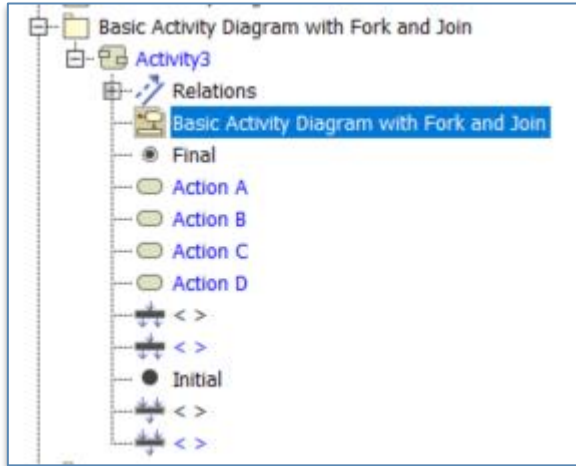


Cameo

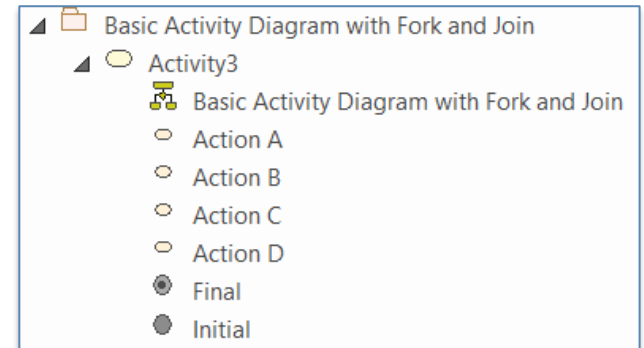
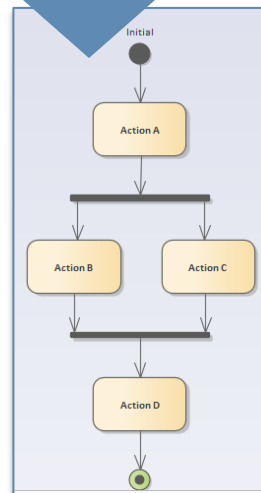
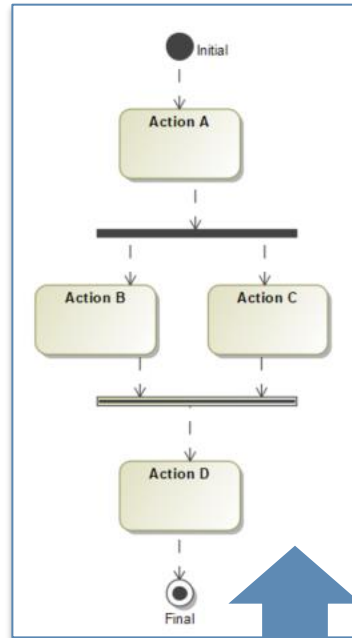
- Diagram information captured by the plugins:
 - Which elements and relationships appear on the diagram
 - Position of the elements on the diagram

Path styling, colors, and other formatting not translated

Activity Diagram

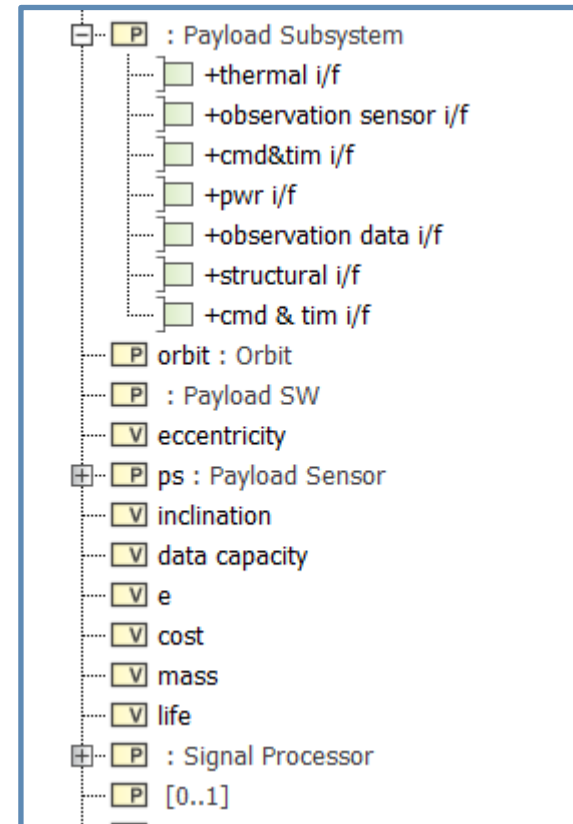
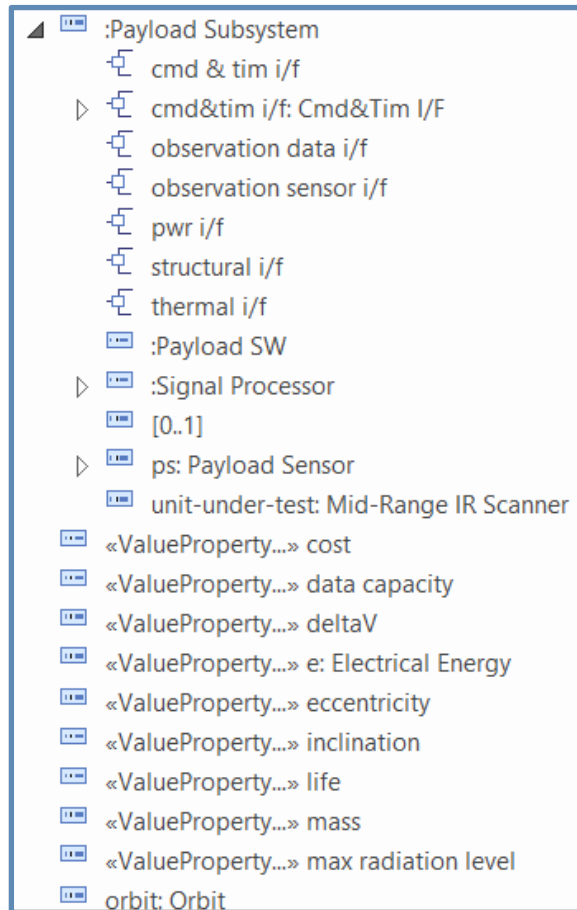


Cameo



Sparx EA

Classifiers and Types



Classifiers captured as seen in part property classified by Payload Subsystem block. Shows inherited ports

Projects Benefiting from MTIP Plugins



- Metamodel support
 - *Establishing a metamodel for corporate-wide use*
 - *Developing tooling to check for consistency and to ease use of a standardized metamodel*
- Exporting behavioral data for simulation
 - *Translating structural and behavioral diagrams based on a mission into an external tool's simulation scenario templates*
- Mapping to reference models
 - *Exporting a Government Reference System Model in SysML from a collection of ASOTs including enterprise model in SysML, requirements, conceptual design, etc.*
- Interfacing with classification tool
 - *Aerospace project that is developing a capability to apply classification markings to a model's elements and relationships based on rules derived from Security Classification Guides*
- Supporting SysML model translations for in-tool simulations
 - *Brought existing simulation ran with Sparx EA's simulation engine into Cameo for team to run using Cameo's Simulation toolkit*

Code Base and Usage

Overview

Cameo Plugin	Sparx EA Plugin
Apache License 2.0	Apache License 2.0
Written in Java	Written in C#
~13,000 SLOC	~10,000 SLOC
~400 classes	~50 classes

Installation and Usage

- Cameo Plugin is packaged in compressed (zipped) folder
 - Imported via the Resource Plugin Manager in Cameo Systems Modeler
 - Accessed from the top menu bar of the application
- Sparx EA plugin can be installed directly into the application using installer executable
 - Accessed in the “Specialized” tab in Sparx Enterprise Architect
- Plugins import HUDS V2 XML file created on export of the model



Demo

Sample Model Translation Demo Video



MTIP_Sample_Model - Enterprise Architect

Start Design Layout Develop Publish Simulate Specialize Construct Execute Configure Find Command...

Search Portals Design Share Run Workspaces Perspectives Preferences Full Screen Visual Style Discussions Review Library Mail Calendar Journal My Kanban My Gantt Help Home Page Libraries Register Help

Model

Browser

- Model
 - «model» 1 Generic Space Systems Model
 - 1 Generic Mission Context
 - Model License
 - 1.04 Behavior
 - Perform Mission
 - Control Trajectory
 - Deploy Mechanisms
 - Launch S/C
 - Maintain Spacecraft Operations
 - Perform Mission
 - Provide Observation Data
 - Separate from L/V
 - 1.060 Spacecraft - Firesat II
 - 1.10 Supporting Elements
 - Custom Stereotypes
 - «stereotype» analysis
 - «stereotype» caused by
 - «stereotype» failure mode
 - «stereotype» general category
 - «stereotype» logical
 - «stereotype» mechanical
 - «stereotype» orbit
 - «stereotype» store
 - «stereotype» test component
 - «stereotype» tpm

Properties

Element

General

Name	
Type	
Stereotype	
Alias	
Keywords	
Status	
Version	

Model (from SysML 1.3)

Package

URI	
Visibility	Public

Project

Features

Attributes Operations Receptions Parts / Properties Interaction Points

Inspector

Properties Toolbox Notes

All Perspectives

+ CAP NUM SCRL CLOUD



19

What's Next?

- Support for additional Metamodels (i.e., UAF, DoDAF, UML)
- Continued adoption and testing through wider userbase
- Joining the OpenMBEE community





Questions?

List of Sources



[1] - Figure 1. Relationship between SysML and UML, Object Management Group (OMG). <https://www.omgsysml.org/what-is-sysml.htm>

[2] - Figure 2. SysML Diagram Types, Object Management Group (OMG). <https://www.omgsysml.org/what-is-sysml.htm>