



ISE&PPOOA

ISE&PPOOA Profile for Cameo System Modeler Tutorial for Installation and Usage

José L. Fernández · Alfonso García

ppooa.com.es

October 2026

Contents

1. Introduction
2. ISE&PPOOA Concept Model
3. Resource Installation
4. Project Setup
5. Functional Architecting Process:
 1. Define Functional Hierarchy
 2. Define Functional Interfaces
 3. Validate Functional Consistency
 4. Iterate on Child Functions
 5. Define Functional Flows

1. Introduction

Why a New SysML V1 Profile?

Functional architecture is a representation of the system behavior (what the system does) in the form of functions considered as **transformations of inputs into outputs**.

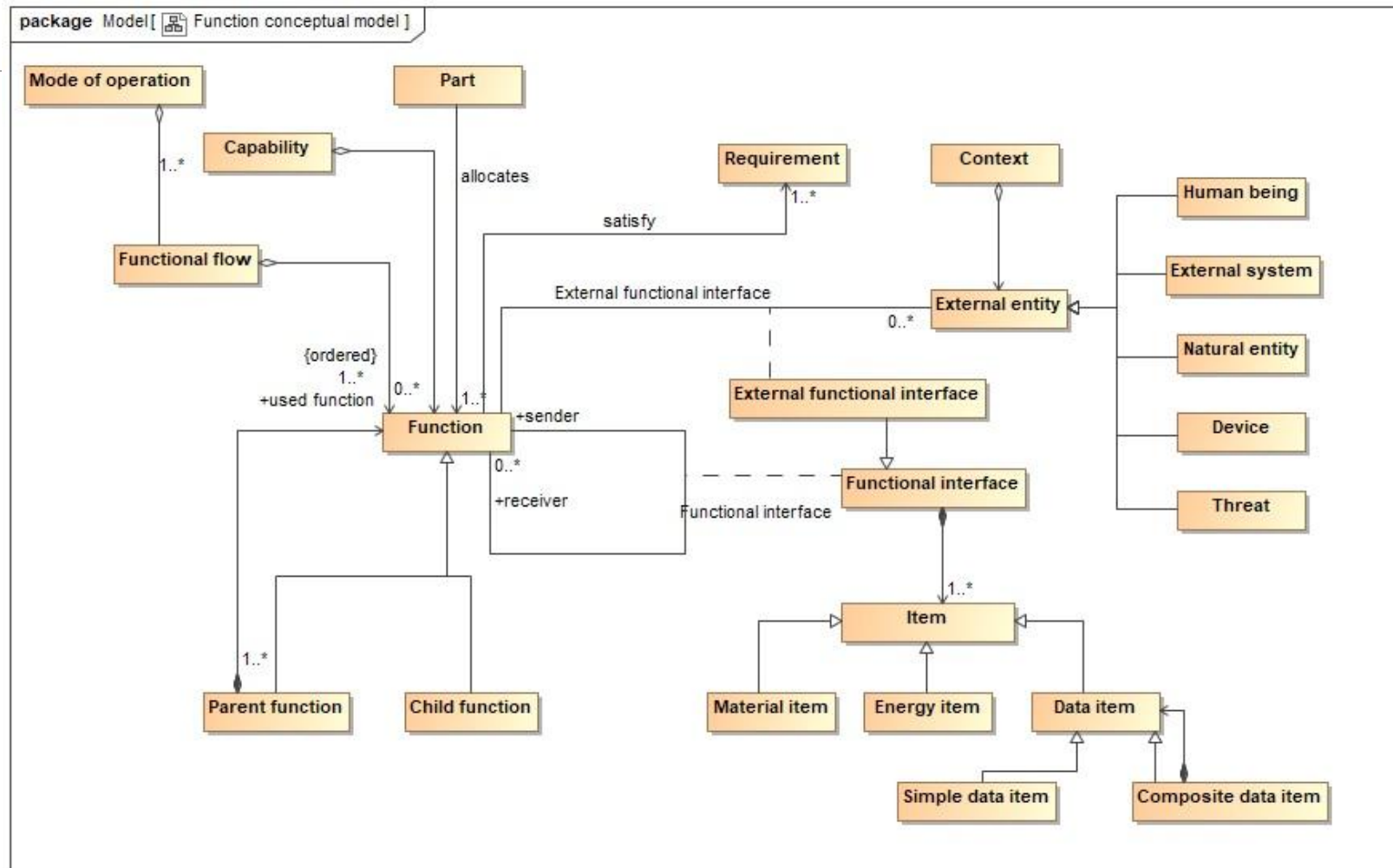
Traditionally, the **functional analysis approach** for developing functional architectures has been applied to the development of many systems since the 1950s.

We consider a correct functional architecture to be the one that follows the basic guidelines of traditional functional analysis, where there are two fundamental types of relationships: **the parent-child relationship between a parent function and its child functions, and the out->in dependency relationship between functions that exchange matter, energy, or data**.

This new ISE&PPOOA SysML profile & plugin for the Cameo System Modeler MBSE tool is the best way to facilitate the creation and validation of functional architectures by MBSE engineers. That is, functional architectures that meet parent-child relations and out-in dependencies between functions.

2. ISE&PPOOA Concept Model

ISE&PPOOA Conceptual Model Diagram



ISE&PPOOA Conceptual Model for Functional Architectures

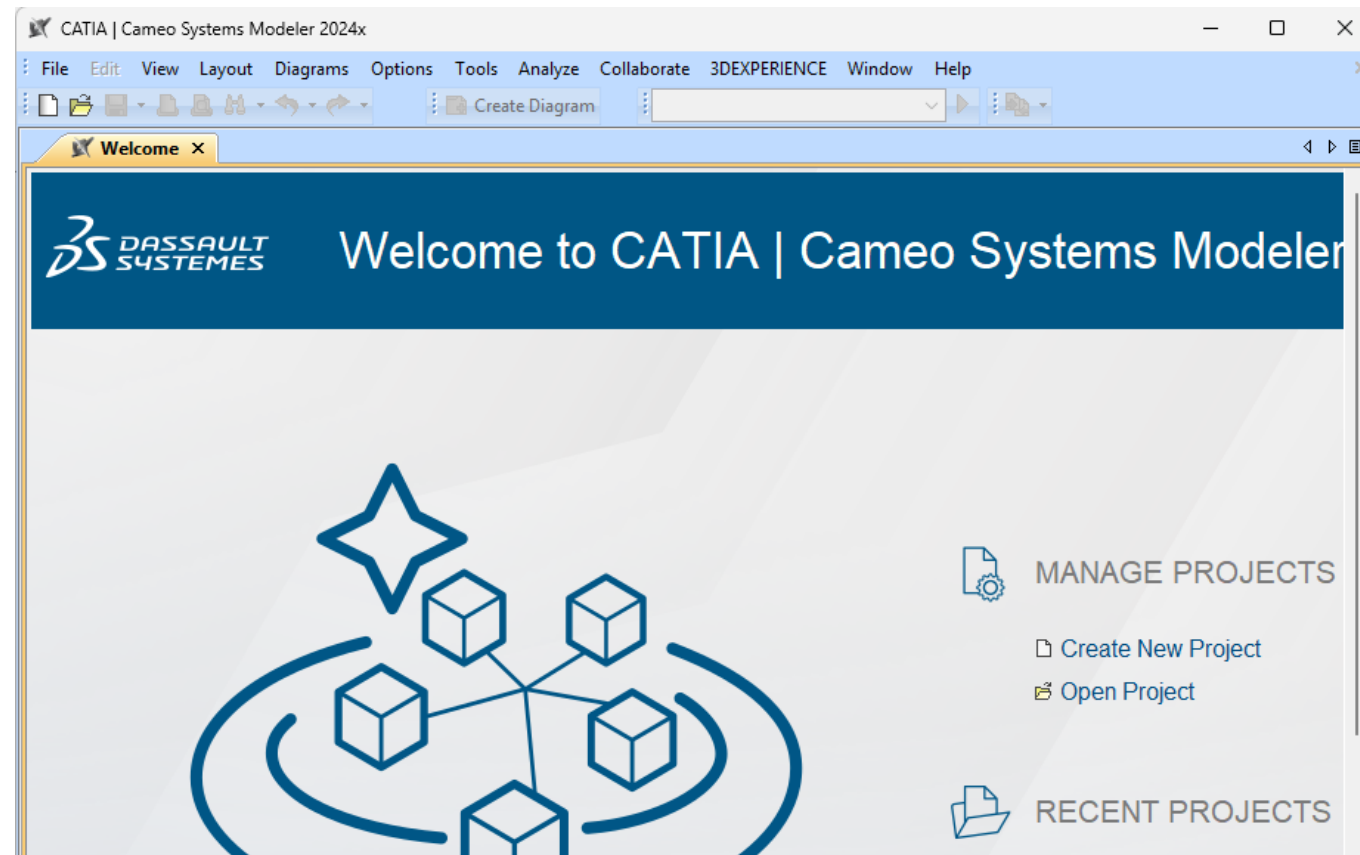
- In the **ISE&PPOOA conceptual model**, a **function** is a **transformation** that may play the role of sender or receiver of items.
- An **item** may be either a **material item**, **energy item** or **data item**.
- When a function is a **parent function** it has **one to many children**.
- A **functional flow** is an ordered aggregation of functions that have the role of **used functions**.
- So, the **parent-child relation** is established when the functions are defined. **The order** of functions is established when the functions are used in a functional flow. Between each pair of functions, we model **functional interfaces**. Functional interfaces are defined as well between some functions and external entities interacting with our system of interest

3. Resource Installation

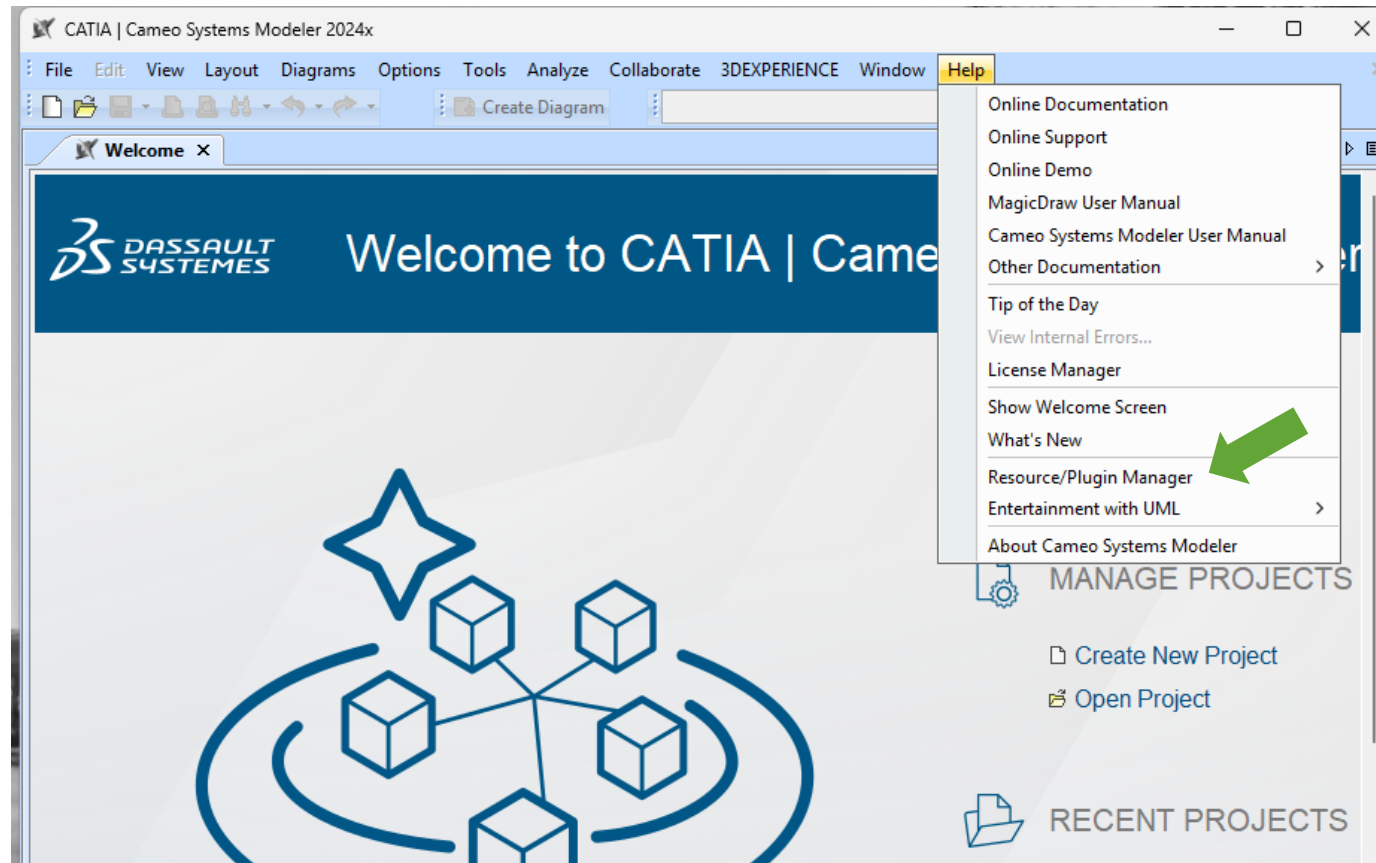
Profile Installation Process

1. Start Cameo 2024x as Administrator
2. Open Help → Resource/Plugin Manager
3. Click “Import”
4. Browse to the downloaded “ISEPPOOA.zip” and click Open
5. Wait until Cameo completes the installation
6. Close the Resource/Plugin Manager window
7. Restart Cameo (Required, as the plugin will not load until Cameo is restarted)
8. Confirm ISE&PPOOA plugin is shown in the Resource/Plugin Manager Window

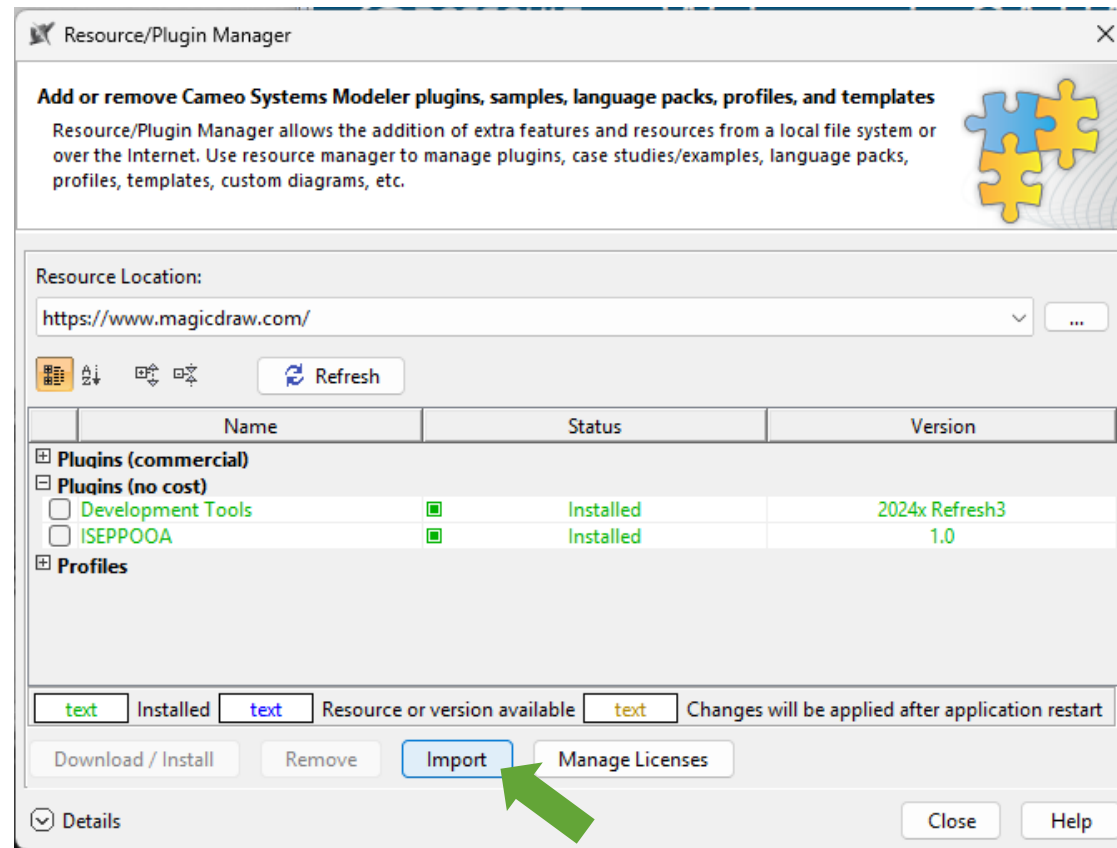
Start Cameo 2024x as Administrator



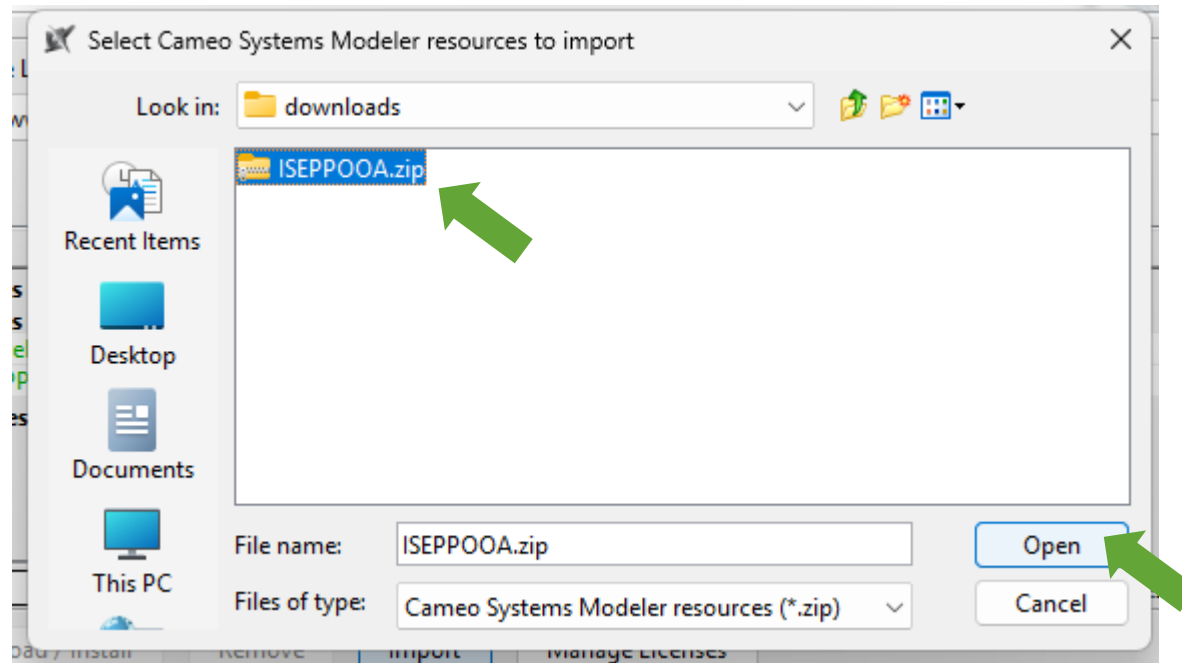
Open Help → Resource/Plugin Manager



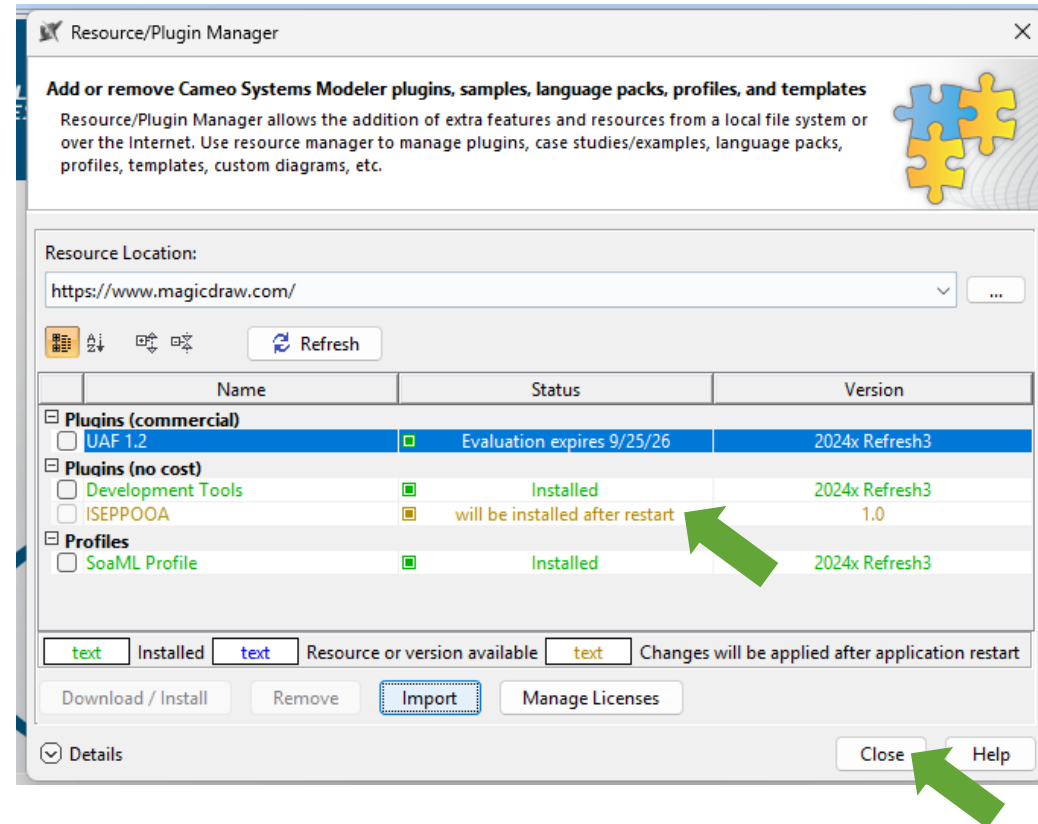
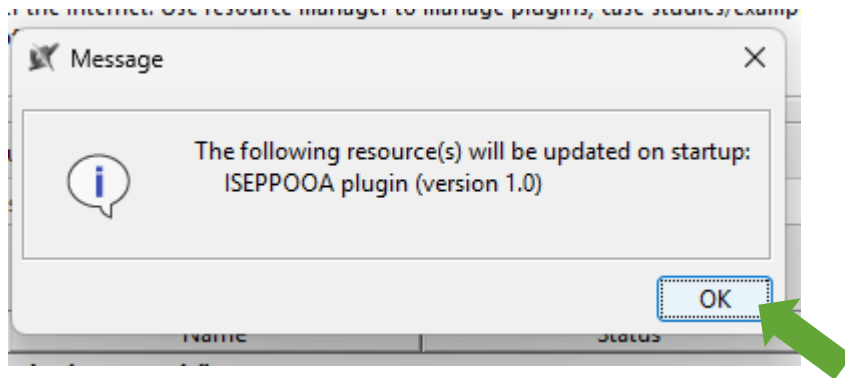
Click “Import”



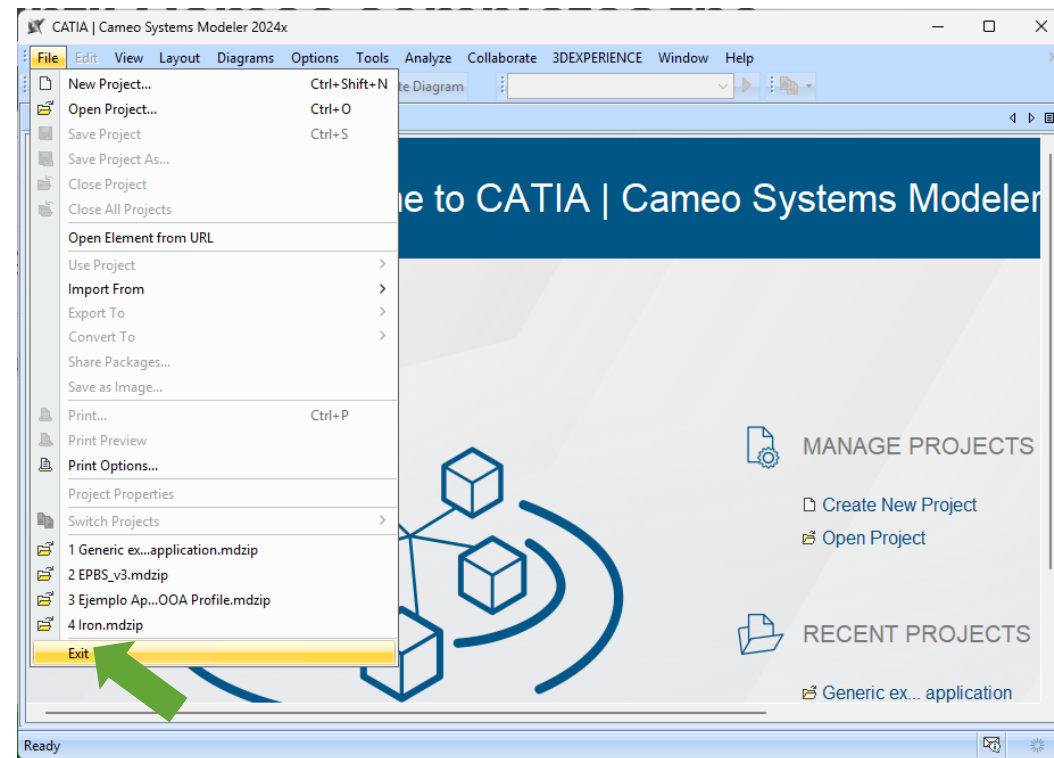
Select ISEPPOOA.zip and Click Open



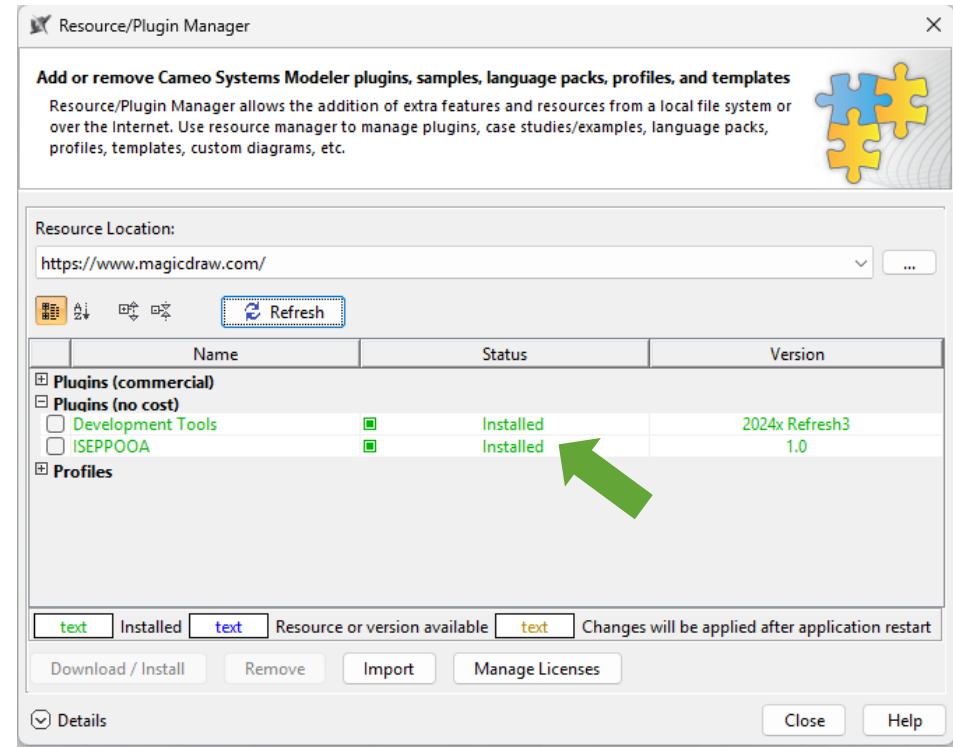
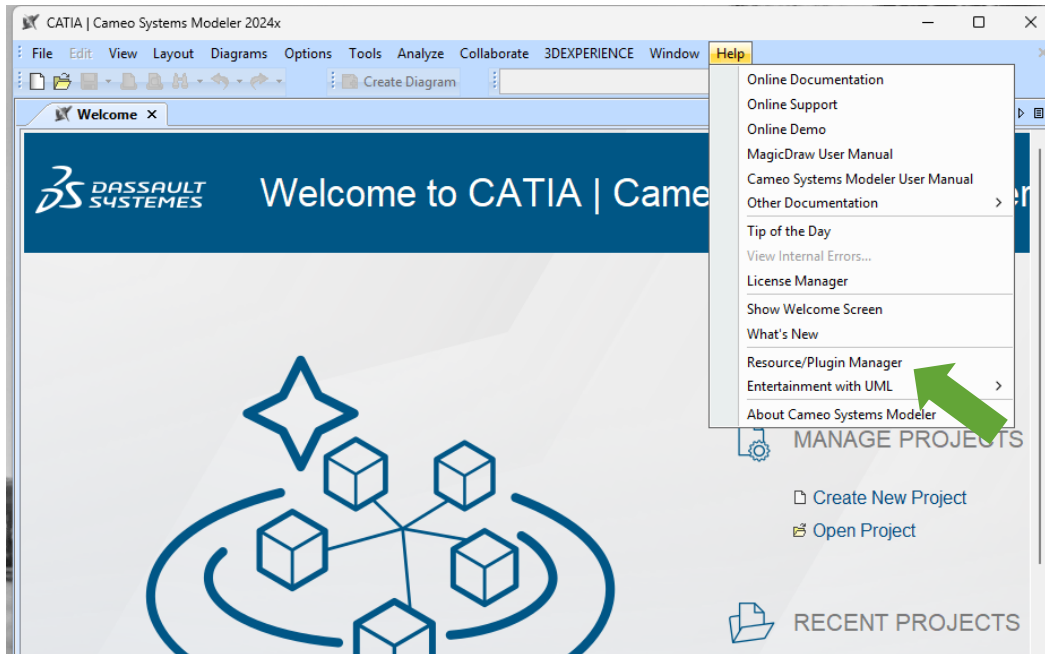
Wait Until Cameo Completes the Installation & Close Resource/Plugin Manager



Exit and Restart Cameo as Administrator



Confirm ISEPPOOA Plugin is Installed

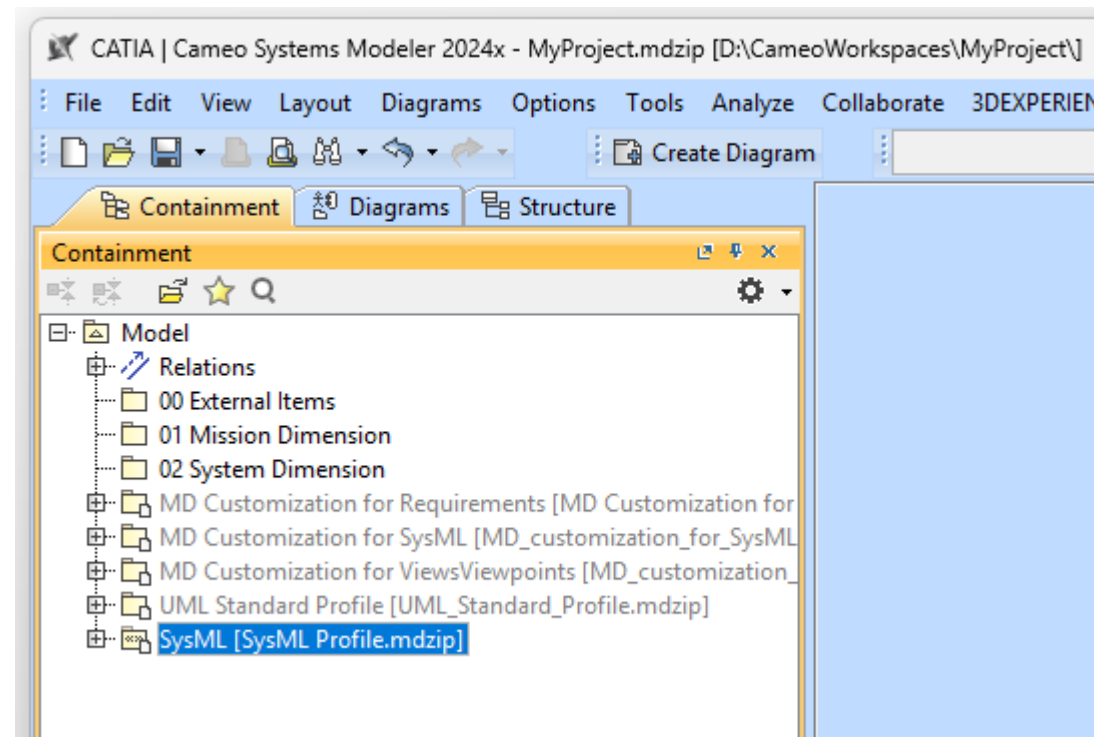


4. Project Setup

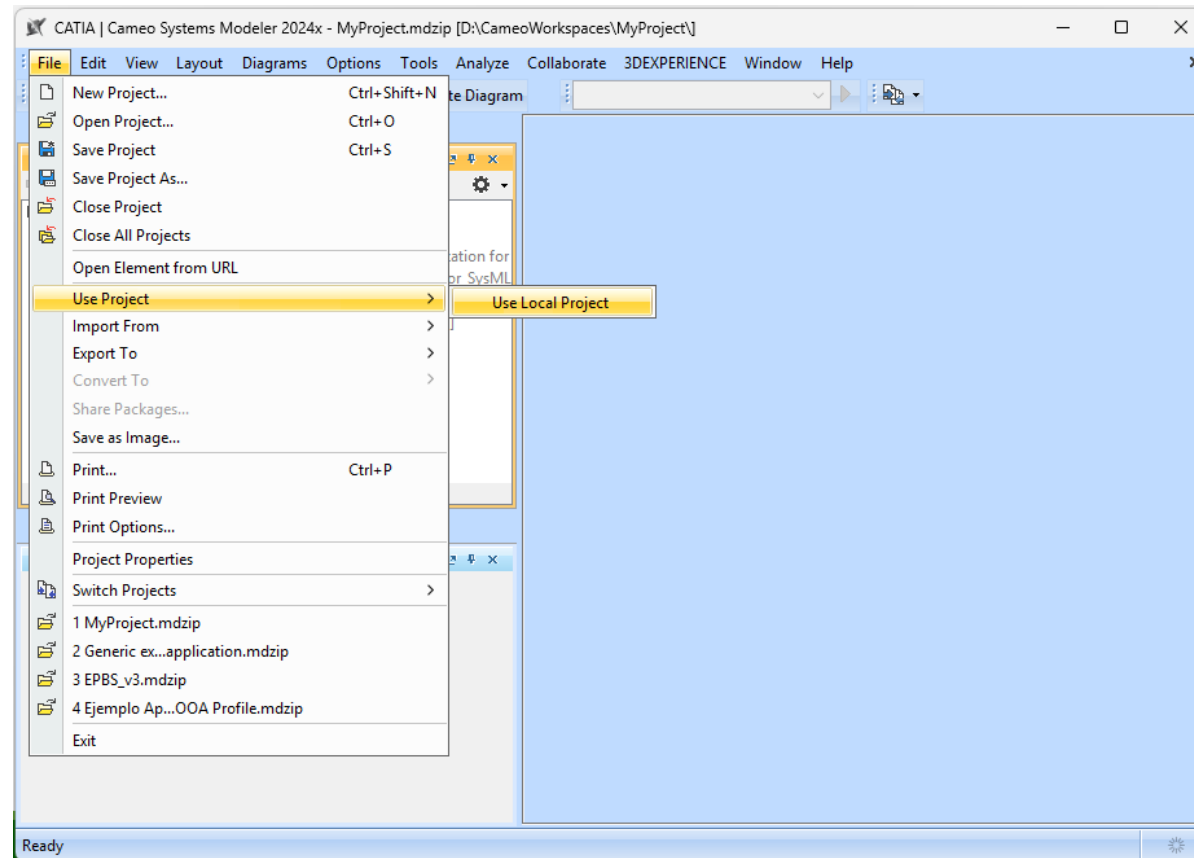
Project Set Up Process

1. Open an existing Project in Cameo which uses the SysML profile
2. Open File → Use Project → Use Local Project
3. Select the “ISEPPOOA Profile.mdzip” in <install.root>\profiles
4. Ensure the profile is always loaded when the project is opened
5. Confirm ISE&PPOOA Profile is used in the project

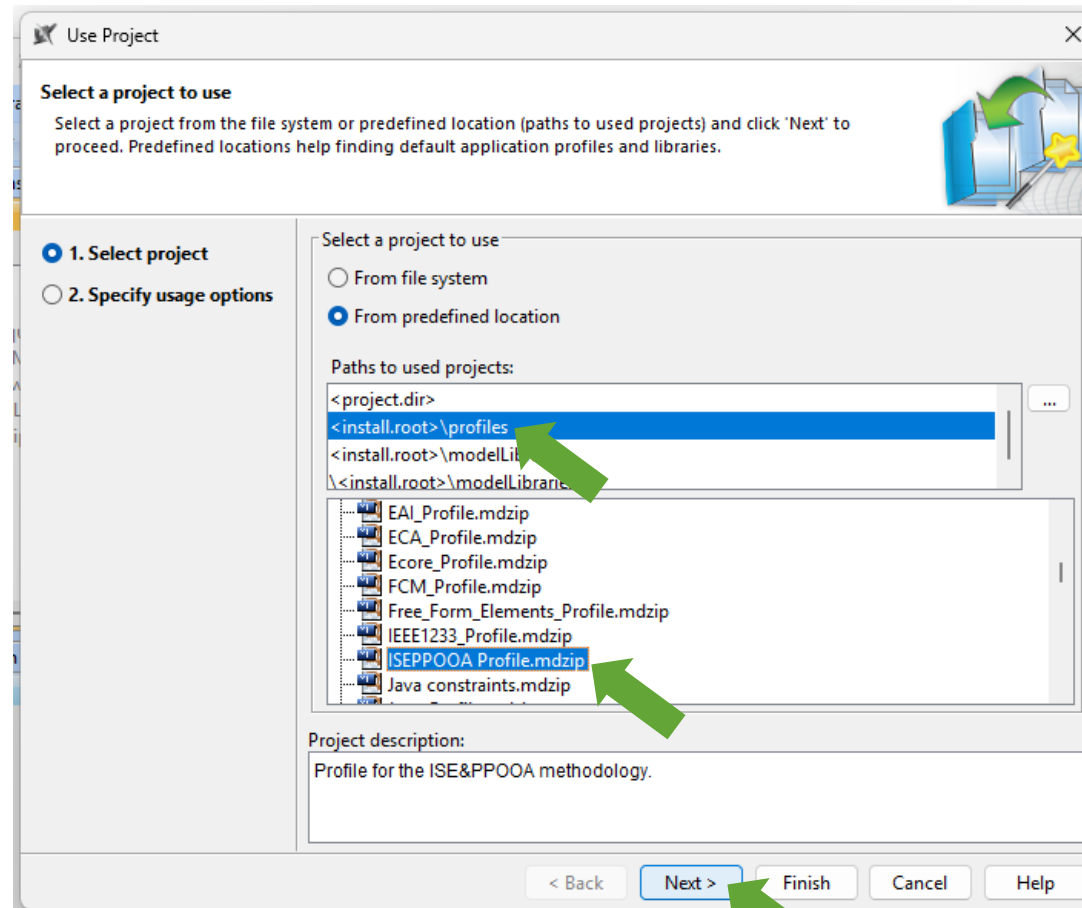
Open a Project



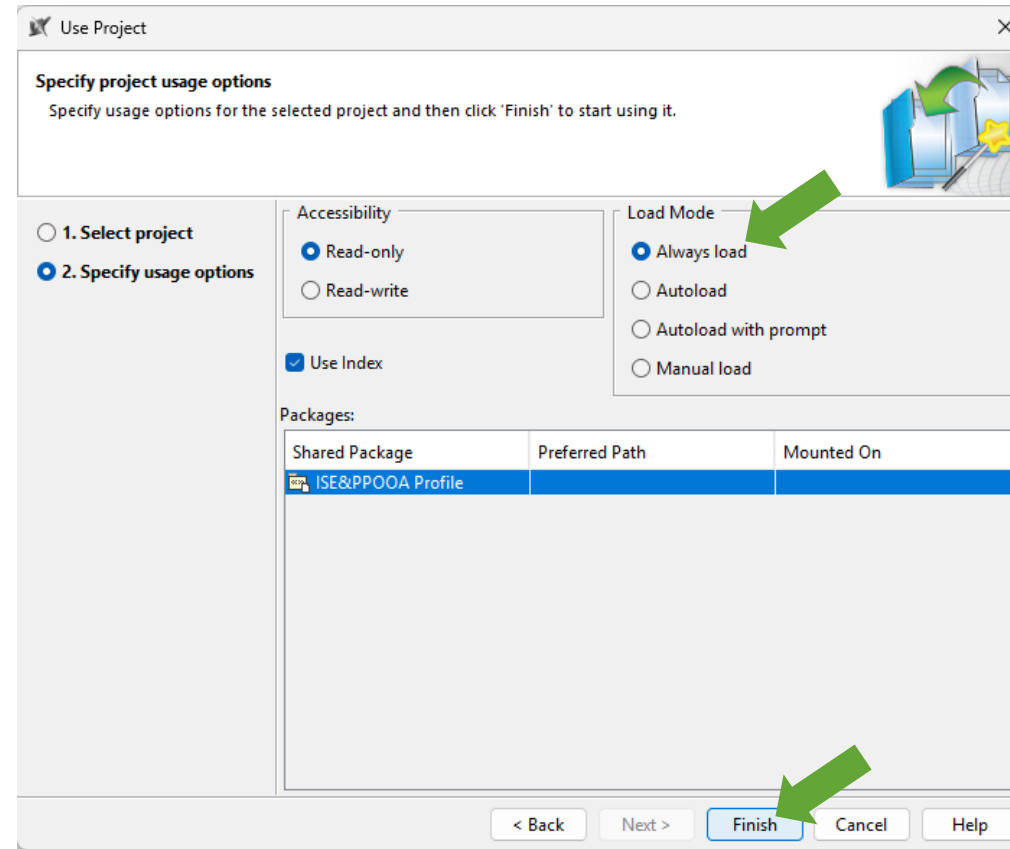
Open File → Use Project → Use Local Project



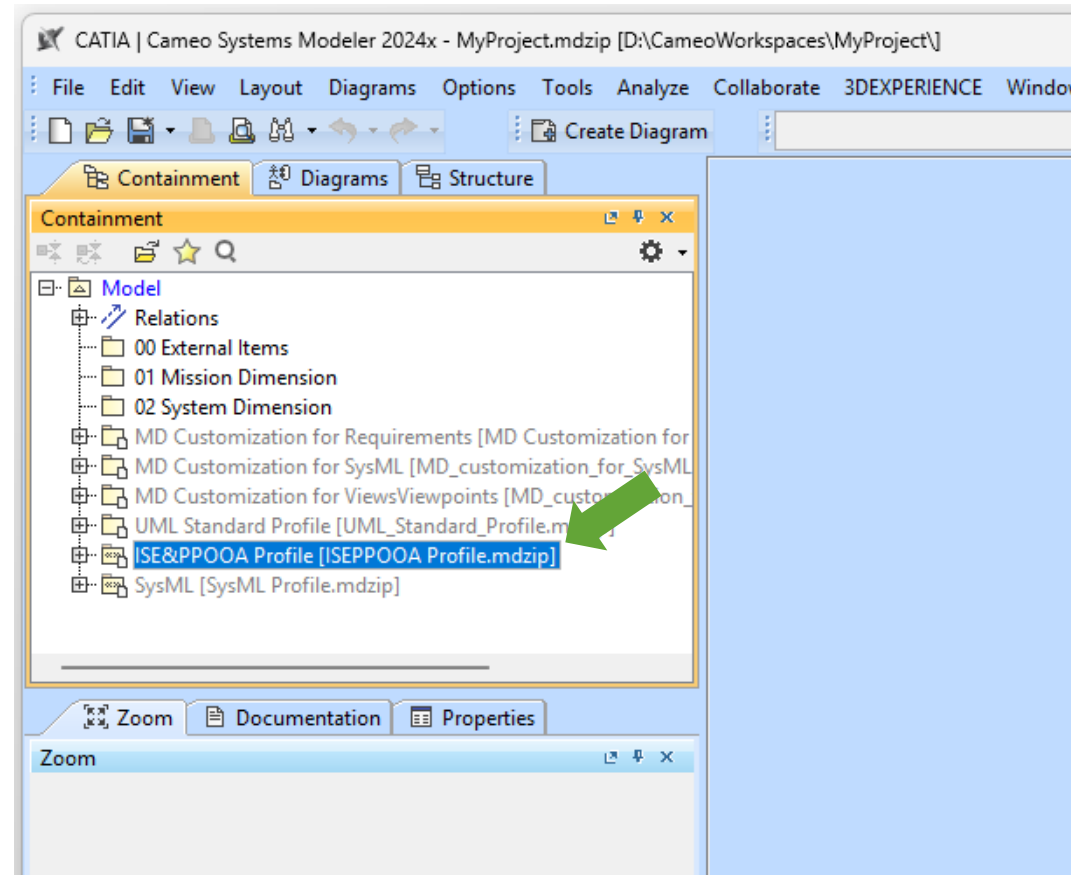
Select the ISE&PPOOA Profile



Ensure the ISE&PPOOA Profile Is Always Loaded



Confirm the ISE&PPOOA Profile is Used

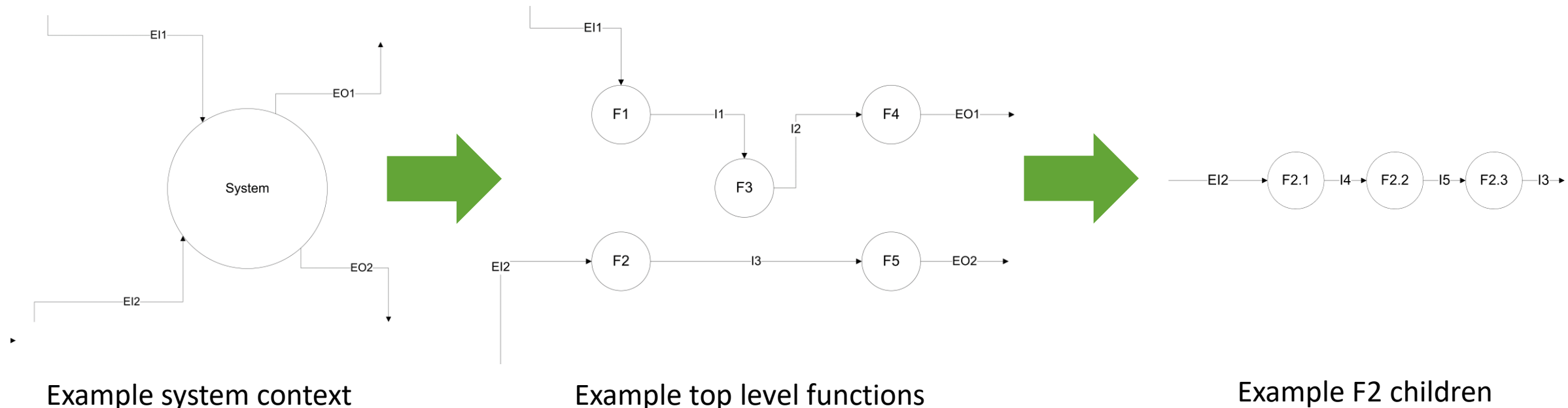


5. Functional Architecting Process

Goal of the ISE&PPOOA Functional Architecting Process

The goal of the ISE&PPOOA functional architecting process is to build a **correct functional architecture that emulates the traditional functional decomposition process** used with DFDs by using SysML notation and N-squared charts instead.

Example of a traditional functional analysis process using Dataflow Diagrams (DFDs)



Functional Architecting Process Overview

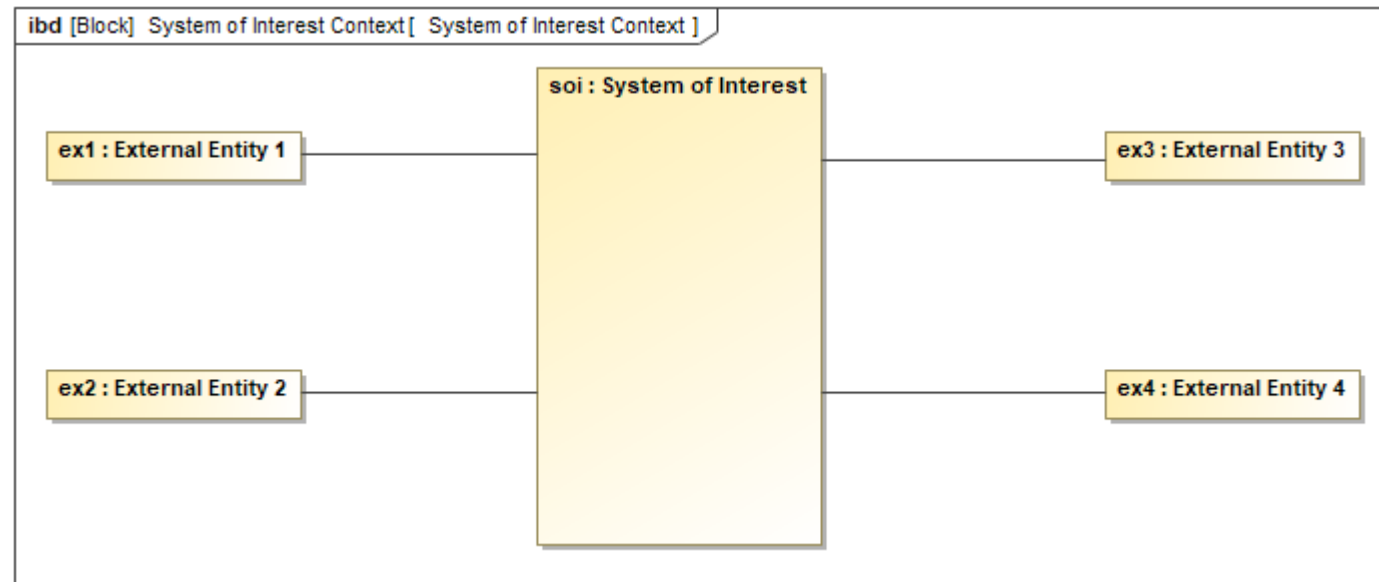
- Pre-condition: the system context is already modeled (defined in the Mission Dimension in the ISE&PPOOA methodology)
- Functional Architecting Process Steps:
 1. Define Functional Hierarchy
 2. Define Functional Interfaces
 3. Validate Functional Consistency
 4. Iterate on Child Functions
 5. Define Functional Flows

Precondition: System Context

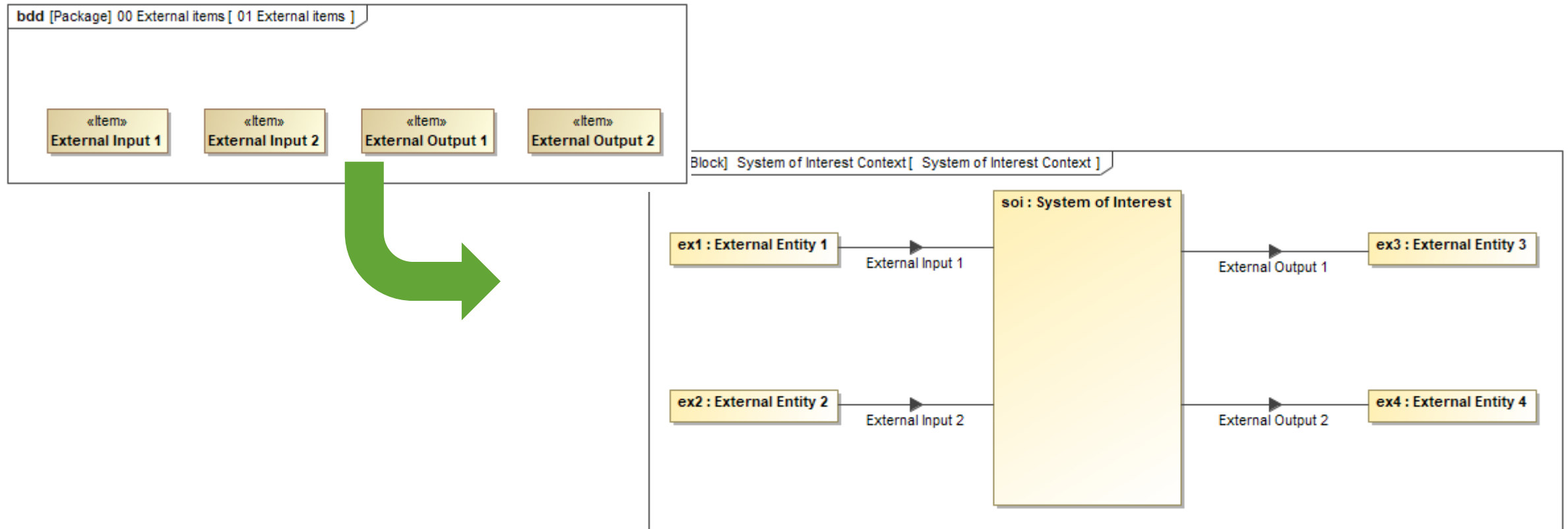
During the definition of the Mission Dimension, the System Context is created. The steps to create it are:

1. Identify external entities interacting with the system of interest
2. Identify external items send and/or received by our system of interest from external entities

Identify External Entities



Identify External Exchanged Items and Add them to Context



5.1 Define Functional Hierarchy

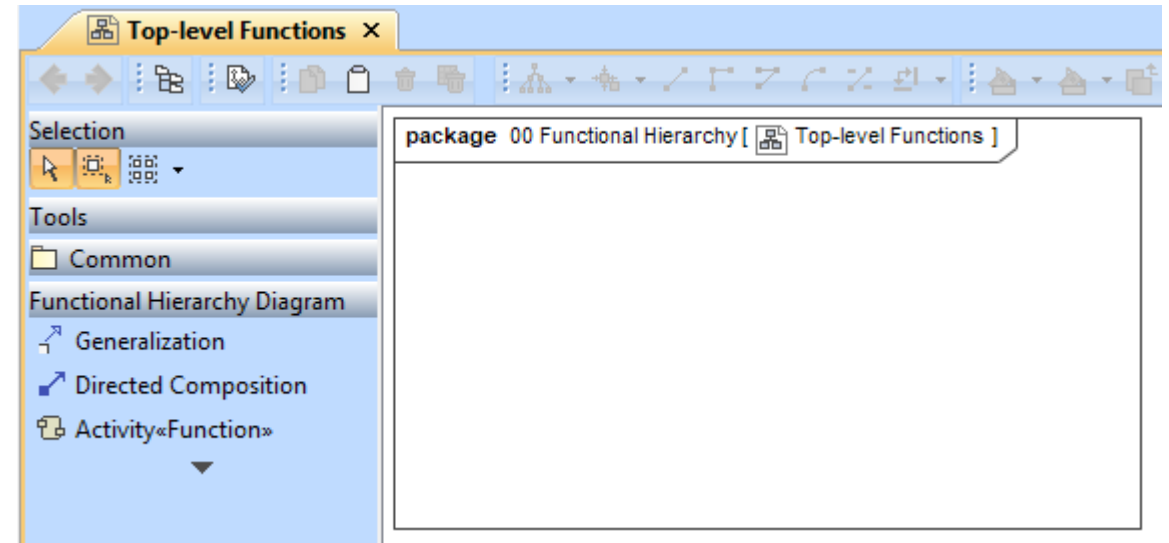
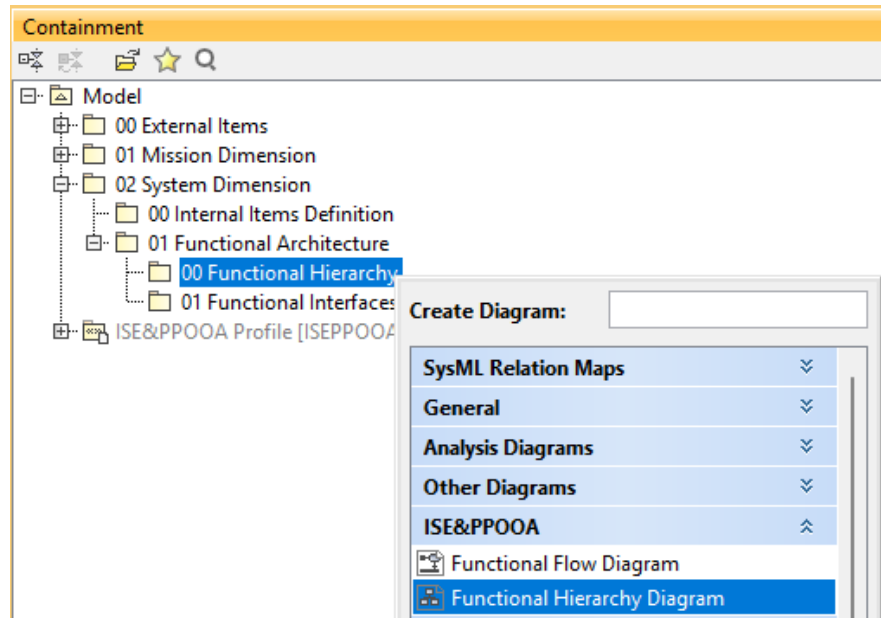
Define the Functional Hierarchy

In a Package within the Functional Architecture:

1. Create a Functional Hierarchy Diagram
2. Create the top-level functions of the system

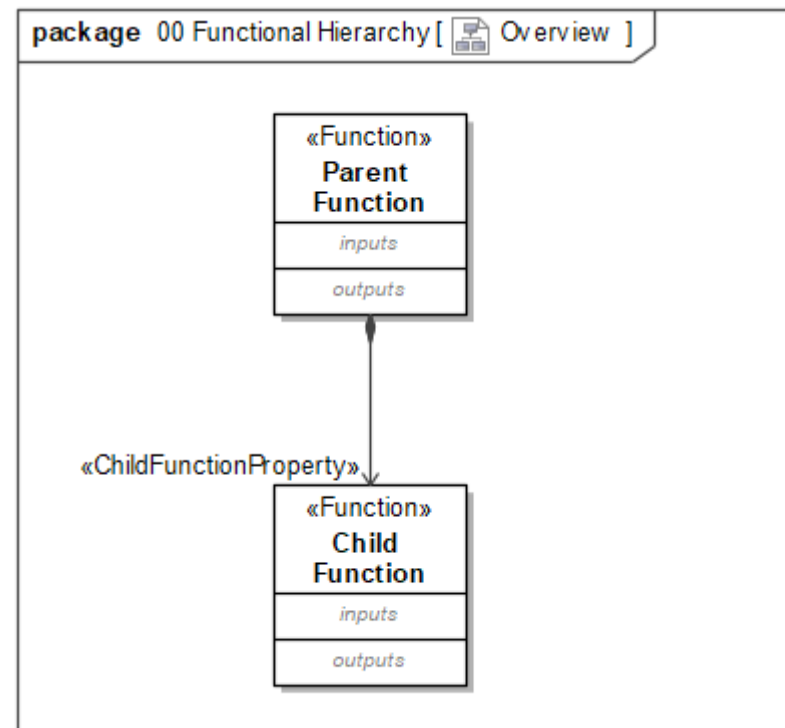
Create a Functional Hierarchy Diagram

The ISE&PPOOA Profile provides a **special type of SysML block definition diagram** to help users create functional hierarchies

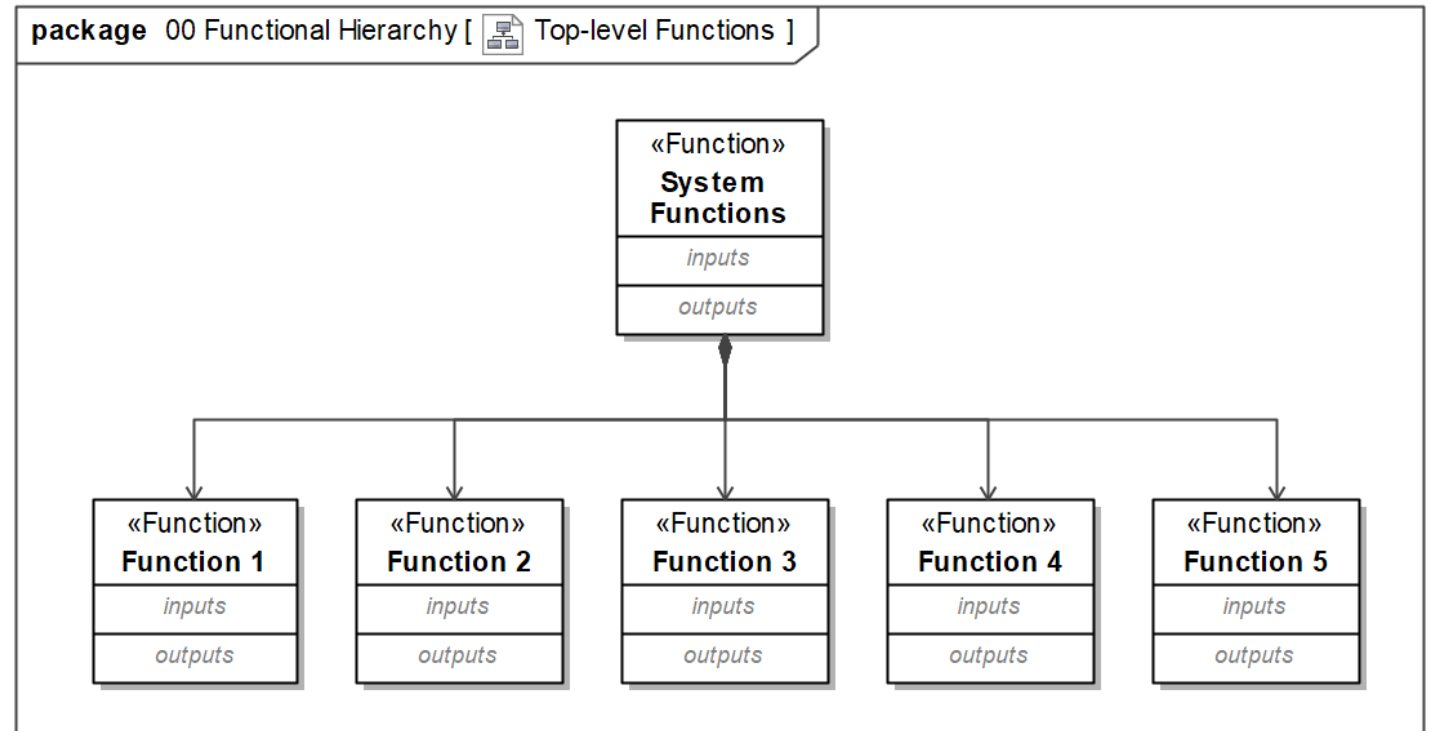
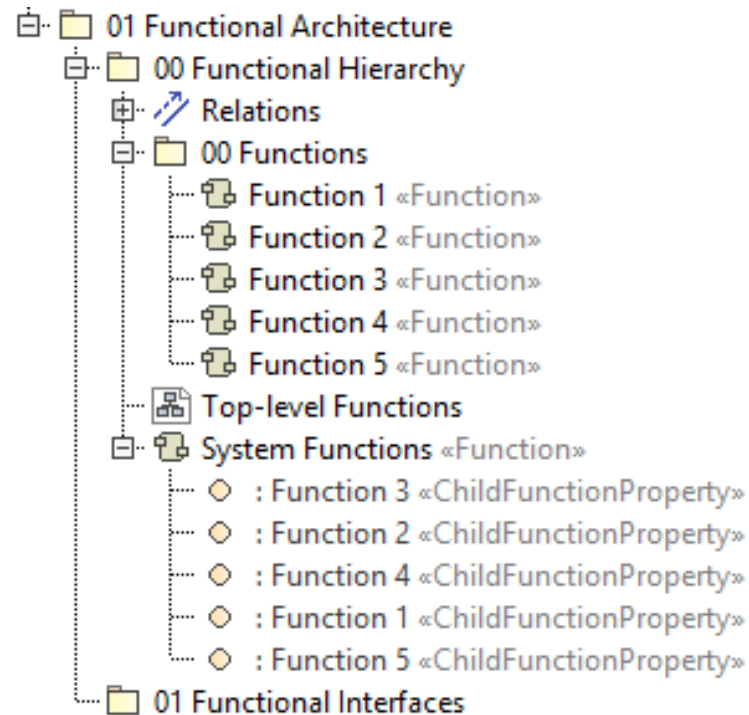


Create the Top-level Functions of the System (1/2)

Use the ISE&PPOOA Profile «**Function**» to represent the functions of the system, and the directed-composition association to represent the parent-child relationships:



Create the Top-level Functions of the System (2/2)



5.2 Define Functional Interfaces

Define Functional Interfaces

In a Package within the Functional Architecture :

1. **Select a group of functions that are related among them**
 - It is recommended that these functions are at the same level of decomposition in the functional hierarchy
2. **Create the N-squared diagram for the selected functions**
 - Here the N squared diagram is the equivalent to a DFD and **the core** of the functional architecting process
3. **Add each item external to the group** to the N-squared diagram to define which function receives or provides the item
4. **Create and add the internal items between functions** to define their needed exchanges to transform the received external items into the system outputs
5. **Reorder the functions** if needed

Create N-squared Diagram for Selected Functions

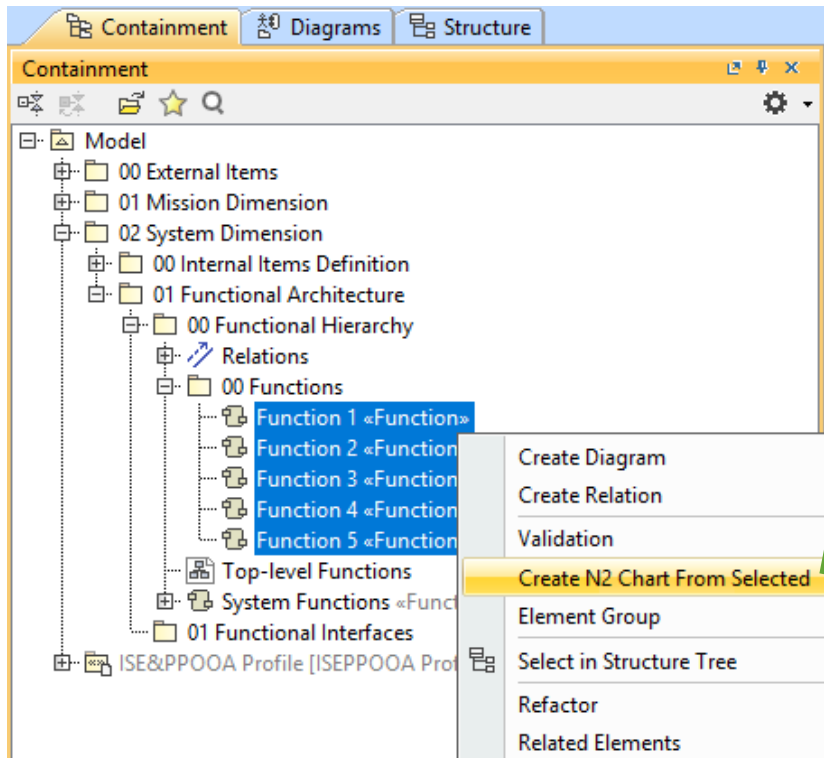
The ISE&PPOOA Profile & Plugin allows users to create and modify N-squared diagrams easily.

To create an N-squared diagram:

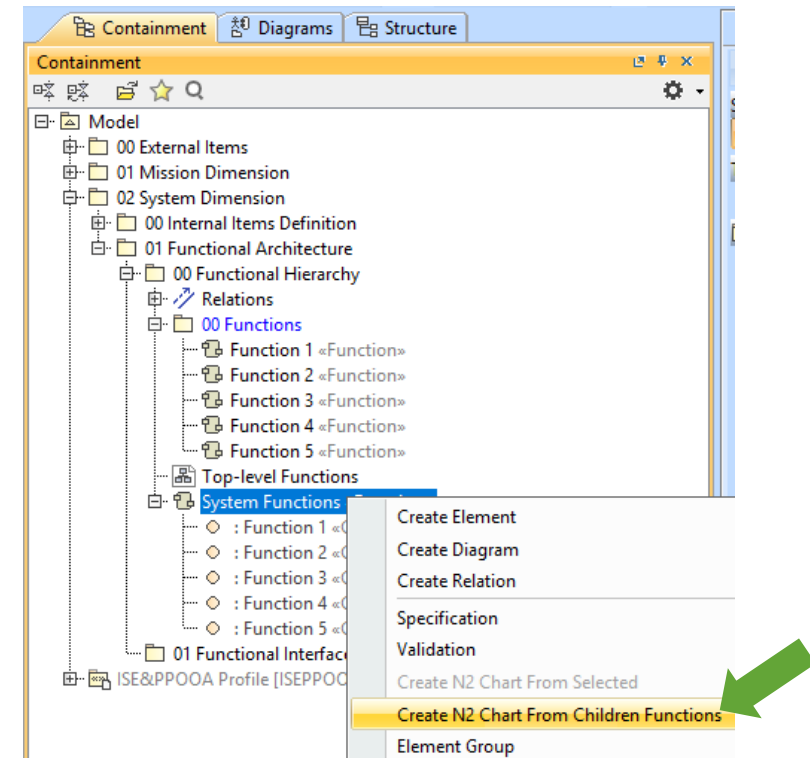
1. **Select the functions** in the containment tree that you need to create the N-squared diagram
2. Right-click to open the context menu and select „**Create N2 Chart From Selected**“
3. **The ISE&PPOOA Plugin creates the N-squared diagram** and all the needed infrastructure elements in a package named N2 Package accordingly
4. **Move the created package to the Functional Interfaces package** and rename it accordingly

Create N-squared Diagram for Selected Functions

Create the N-squared diagram from the selected functions. Alternatively, you can do the same N-squared diagram **from the root** of the functional hierarchy.



OR



Create N-squared Diagram for Selected Functions

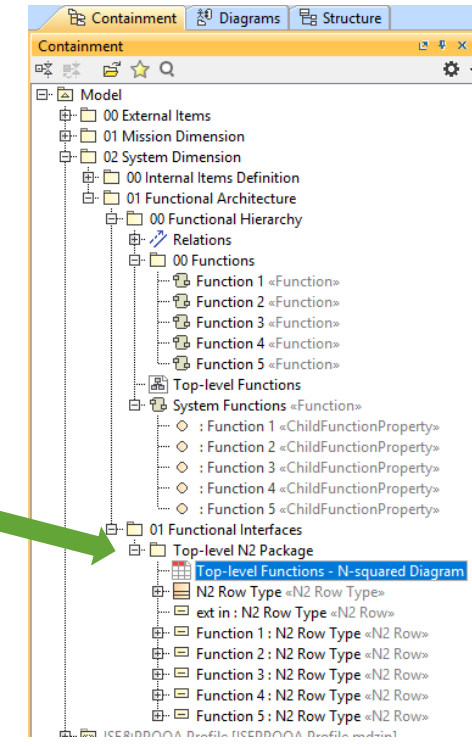
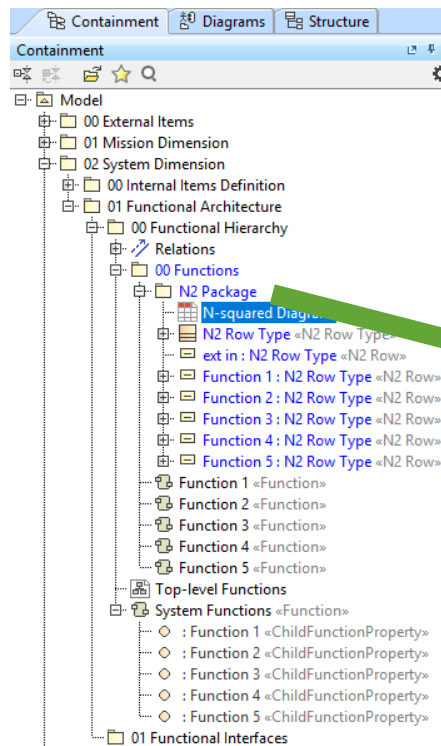
The ISE&PPOOA Plugin creates the related N-squared diagram accordingly

The screenshot displays the ISE&PPOOA Plugin interface. On the left, the 'Containment' tree shows a hierarchical structure of model elements, including '00 Functions' and 'N2 Package'. The 'N-squared Diagram' tool is selected. On the right, the 'N-squared Diagram' window shows a table with 8 columns: '#', 'Name', 'Function 1', 'Function 2', 'Function 3', 'Function 4', 'Function 5', and 'ext out'. The table contains 6 rows of data, representing the N-squared diagram for the selected functions.

#	Name	Function 1	Function 2	Function 3	Function 4	Function 5	ext out
1	ext in						
2	Function 1	Function 1					
3	Function 2		Function 2				
4	Function 3			Function 3			
5	Function 4				Function 4		
6	Function 5					Function 5	

Create N-squared Diagram for Selected Functions

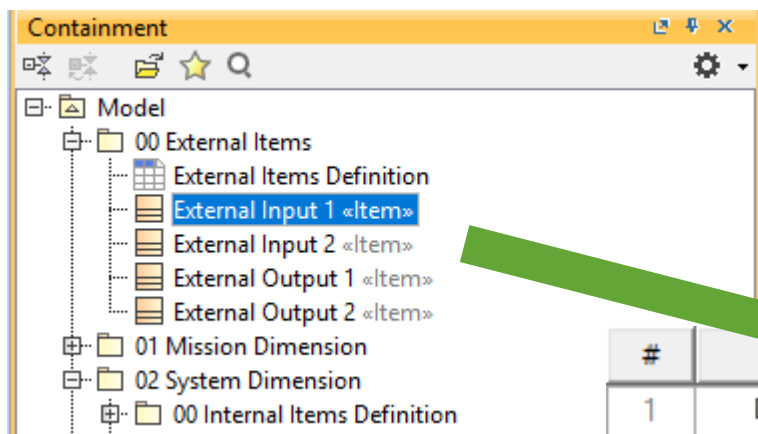
Move the package containing the N-squared diagram to the **Functional Interfaces** package and rename the package and diagram accordingly.



Add the External Items To the N-squared Diagram (1/2)

Drag and drop the items external to the group of selected functions to the n-squared diagram.

Note: in case the group is the top-level functions, then the external items are the external items to the system identified during the context definition step.

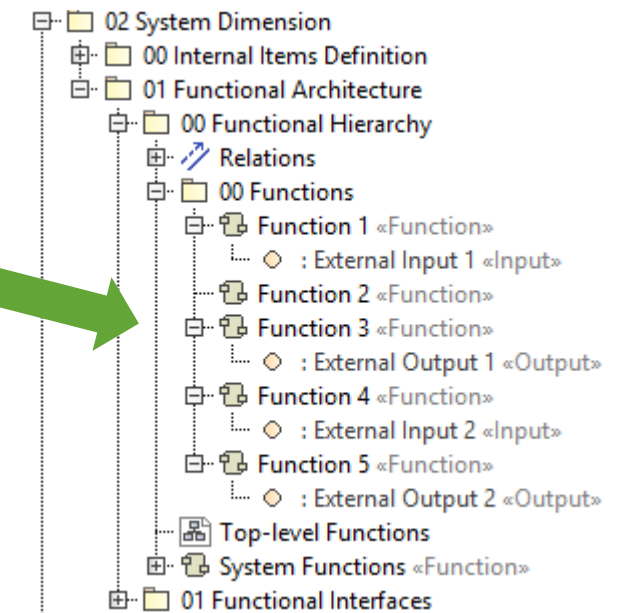


#	Name	Function 1	Function 2	Function 3	Function 4	Function 5	ext out
1	ext in	External Input 1					
2	Function 1	Function 1					
3	Function 2		Function 2				
4	Function 3			Function 3			
5	Function 4				Function 4		
6	Function 5					Function 5	

Add the External Items To The N-squared Diagram (2/2)

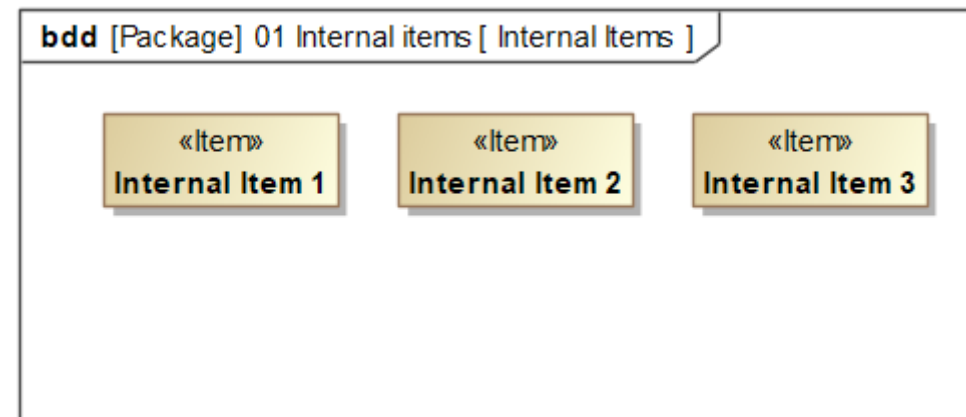
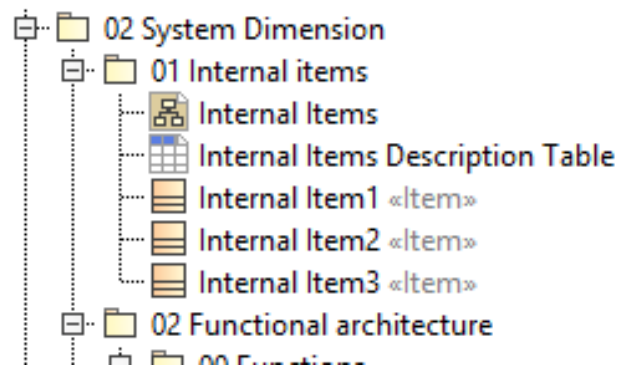
Note that as you add items to the N-squared diagram, the ISE&PPOOA Plugin automatically updates the function definitions to add the related inputs and outputs accordingly, maintaining consistency.

#	Name	Function 1	Function 2	Function 3	Function 4	Function 5	ext out
1	ext in	External Input 1			External Input 2		
2	Function 1	Function 1					
3	Function 2		Function 2				
4	Function 3			Function 3			External Output 1
5	Function 4				Function 4		
6	Function 5					Function 5	External Output 2



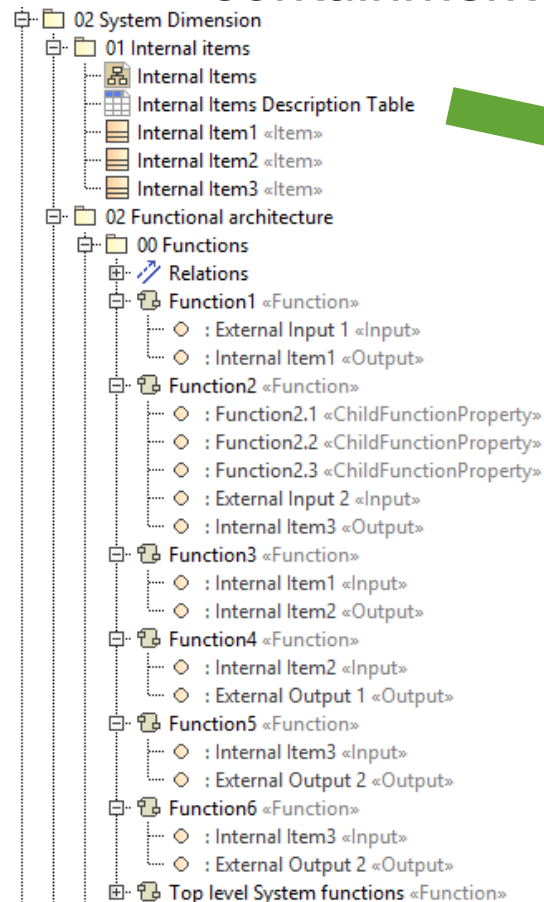
Identify the Internal Items Exchanged between Functions

In the Internal Items package, define the internal items to be exchanged between the functions displayed in the N-squared diagram.



Add the Internal Items between Functions

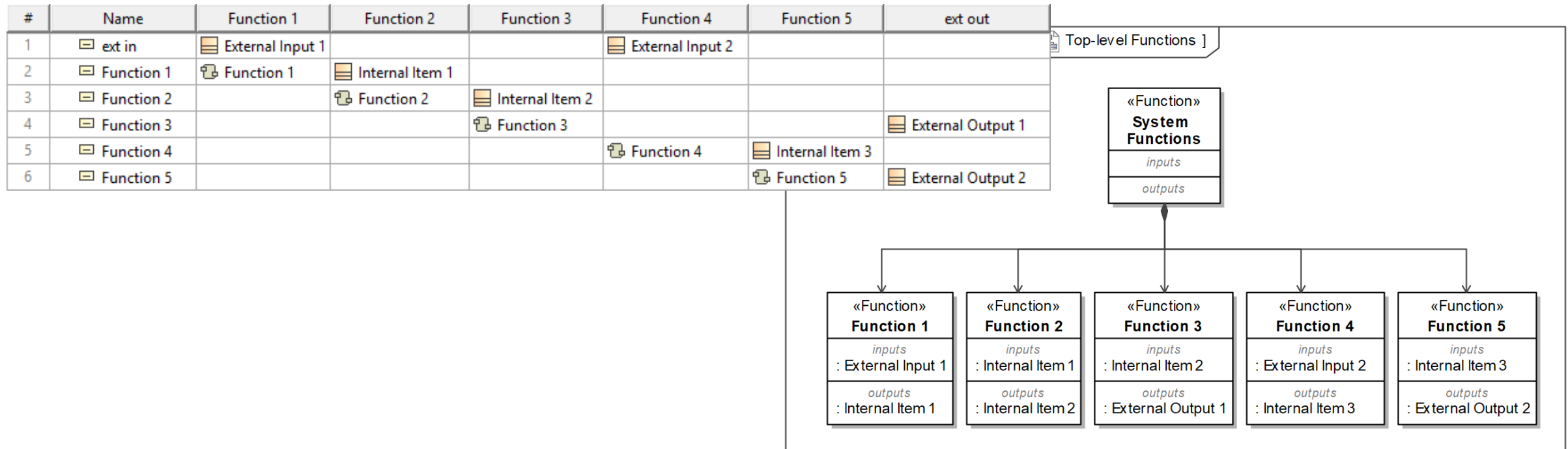
Add the internal items to the N-squared diagram by drag & drop from the containment tree.



#	Name	Function 1	Function 2	Function 3	Function 4	Function 5	ext out
1	ext in	External Input 1			External Input 2		
2	Function 1	Function 1	Internal Item 1				
3	Function 2		Function 2	Internal Item 2			
4	Function 3			Function 3			External Output 1
5	Function 4				Function 4	Internal Item 3	
6	Function 5					Function 5	External Output 2

Add the Internal Items between Functions

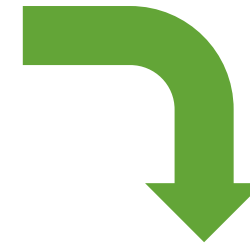
The updated inputs and outputs resulting from the creation of the N-squared diagram can be displayed in the functional hierarchy diagrams.



(Optional) Re-order Functions Manually

Drag a function in the diagram and drop it on another function in the diagonal of the N-squared diagram to exchange their positions,

#	Name	Function1	Function2	Function4	Function3	Function5	ext out
1	ext in	External Input 1	External Input 2				
2	Function1	Function1			Internal Item1		
3	Function2		Function2			Internal Item3	
4	Function4			Function4			External Output 1
5	Function3			Internal Item2	Function3		
6	Function5					Function5	External Output 2



#	Name	Function1	Function3	Function4	Function2	Function5	ext out
1	ext in	External Input 1			External Input 2		
2	Function1	Function1	Internal Item1				
3	Function3		Function3	Internal Item2			
4	Function4			Function4			External Output 1
5	Function2				Function2	Internal Item3	
6	Function5					Function5	External Output 2

5.3 Validate Functional Consistency

Validate Functional Consistency

The ISE&PPOOA Profile & Plugin allow users to evaluate:

1. (Manual) Consistency between defined inputs and outputs and between
 - sibling functions at the same level of the functional hierarchy
 - parent functions and their child functions
2. (Automatic) Consistency between defined inputs and outputs and the related functional interfaces in N-squared diagrams.

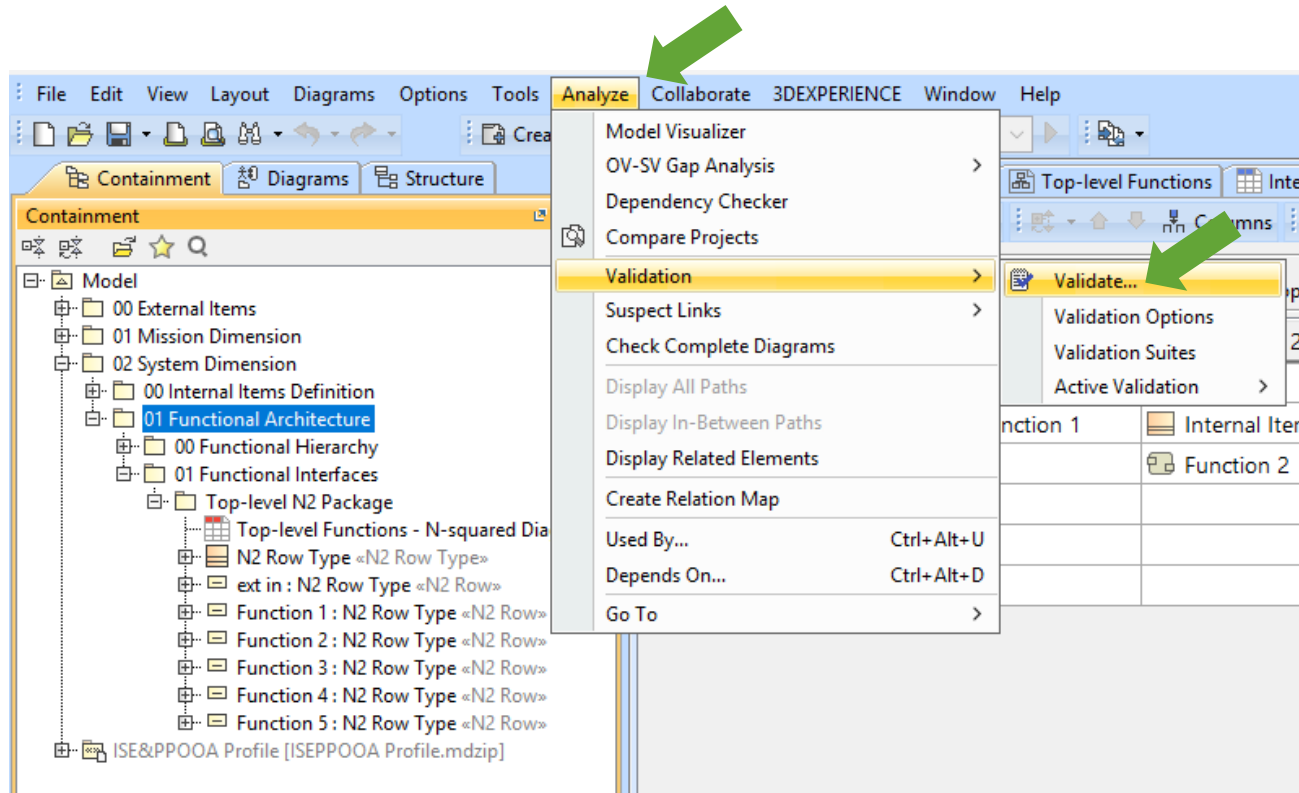
Validate Functional Consistency

To validate the consistency of functional interfaces,

1. Open the Validation analysis tool
2. Execute the ISE&PPOOA Validation Suite on the model
3. Fix all the findings in the Validation Results
4. Confirm all findings are resolved

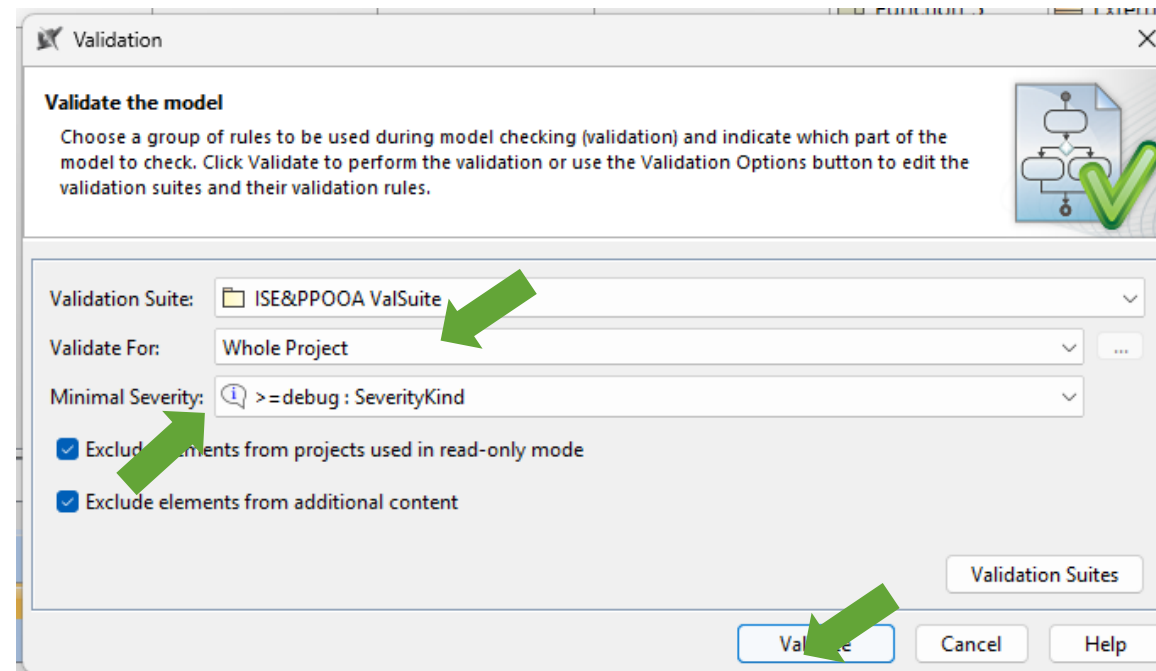
Open the Validation Tool

Open Analyze → Validation → Validate...



Execute the ISE&PPOOA Validation Suite

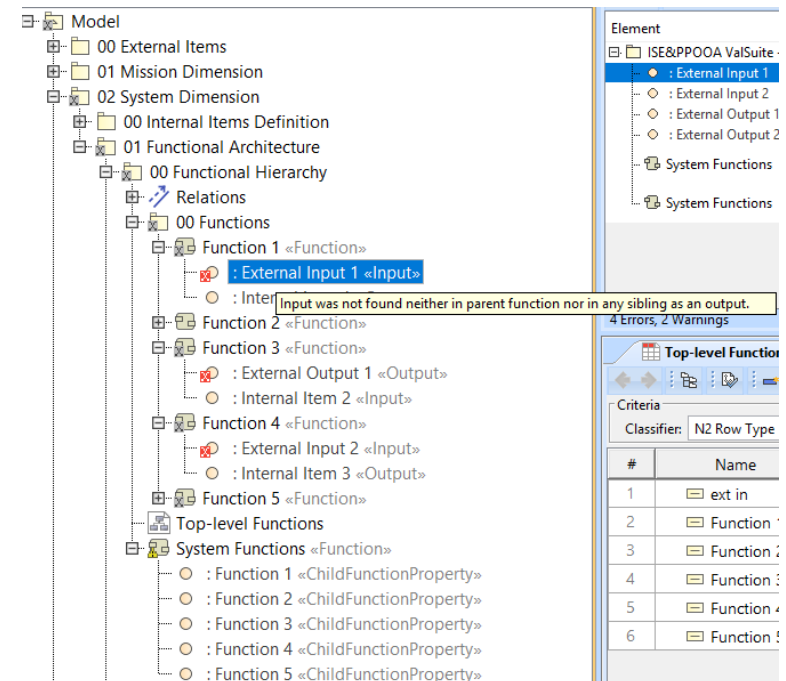
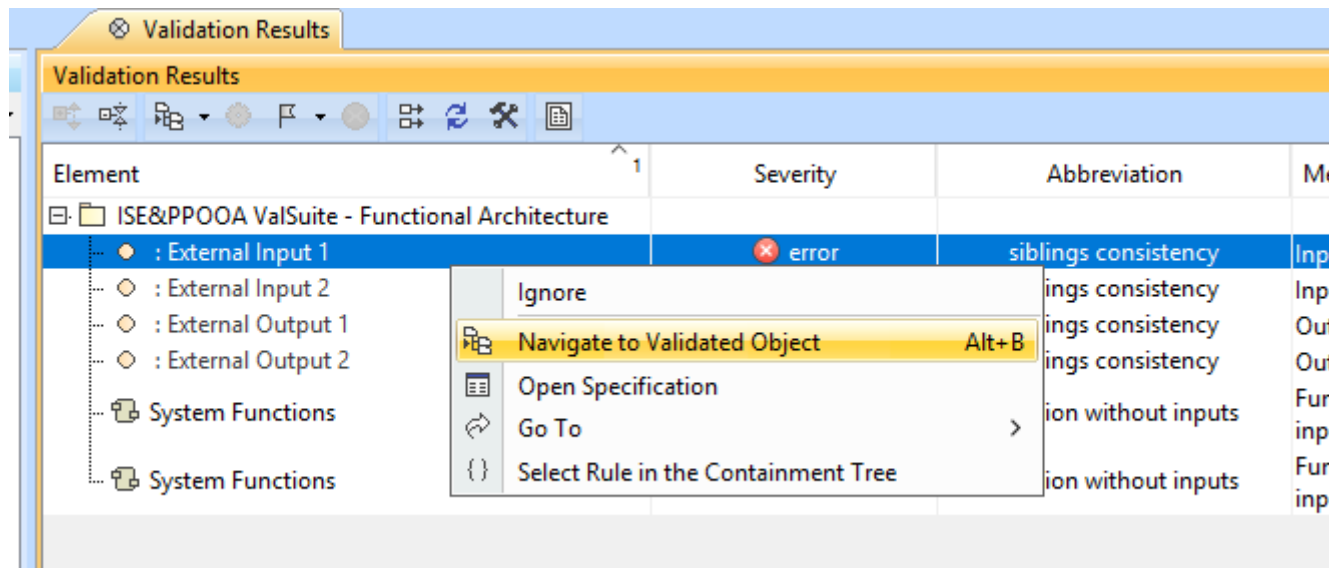
1. Select “ISE&PPOOA ValSuite” as the “Validation Suite”.
2. Select “Whole Project” as the “Validate For”
3. Click on the “Validate” button



Fix All the Findings in the Validation Results

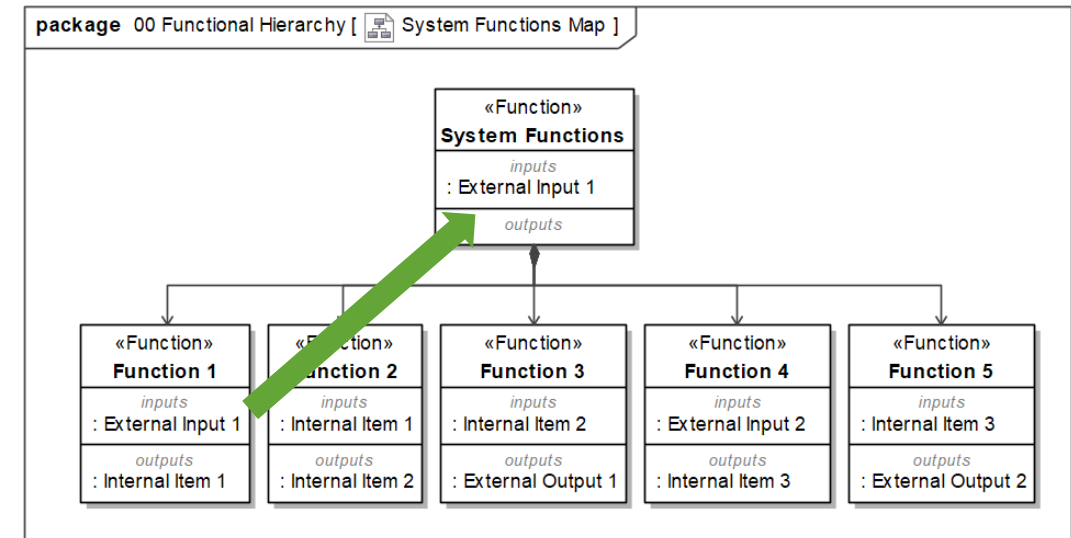
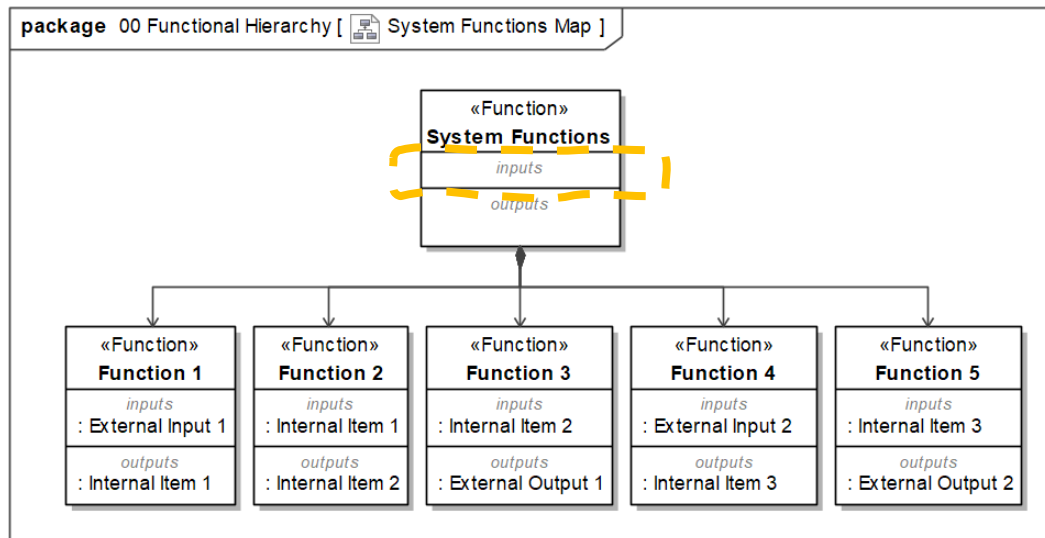
Update the model by adding/deleting inputs/outputs to fix each inconsistency found by the ISE&PPOOA Validation Suite.

For example, **External Input 1** (which types an input of **Function 1**) was not found to type any input of its parent function (i.e. **System Functions**) or any output of its sibling functions (i.e. Function 2,...,Function 5).



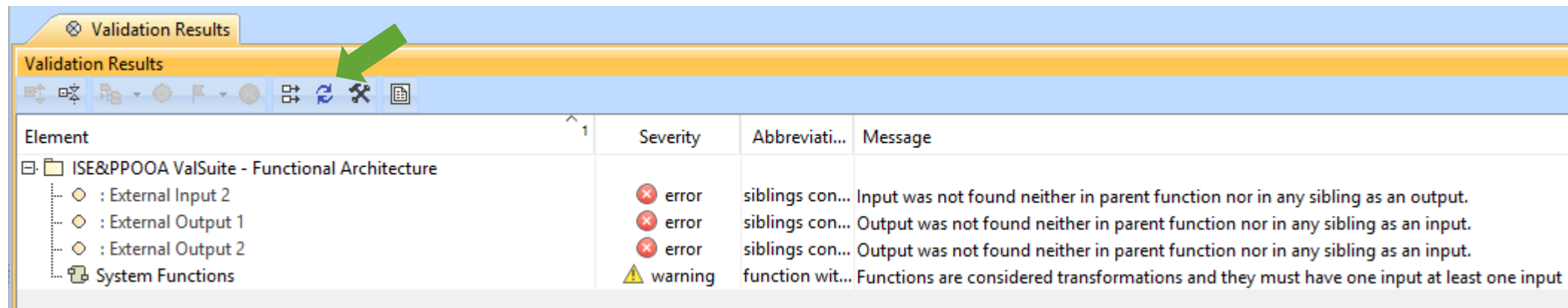
Fix All the Findings in the Validation Results

External Input 1 comes from outside the system boundary, so it must be an input for **System Functions**. To resolve this issue, we add the missing input to the parent function of **Function 1**.



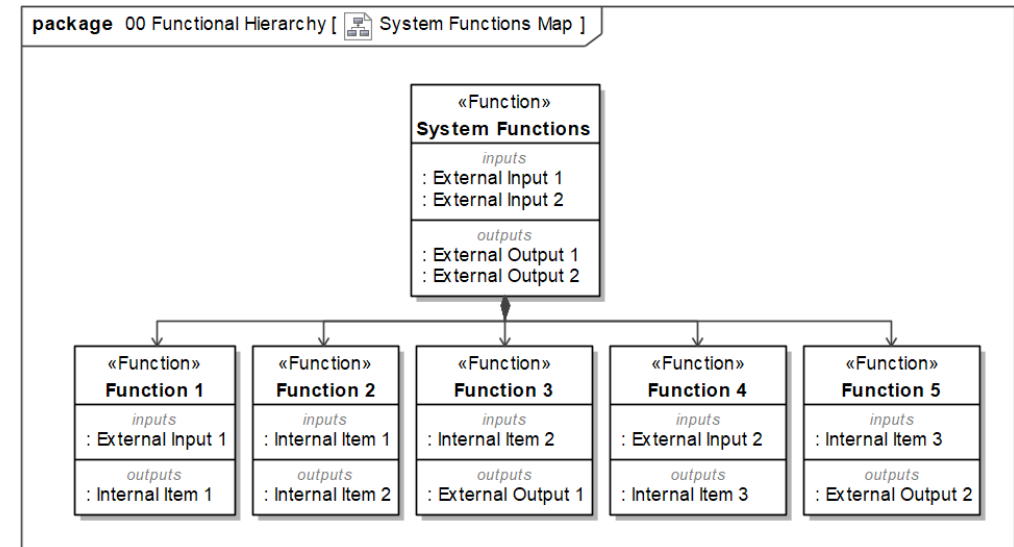
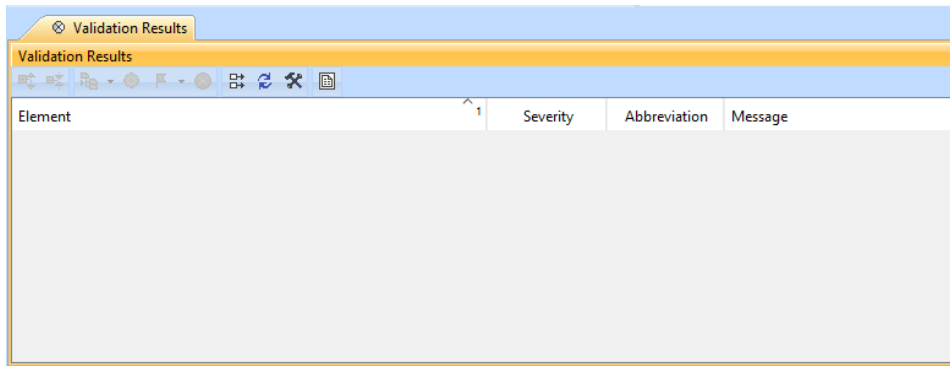
Confirm All Findings Are Resolved (1/2)

After resolving each finding, re-run the validation suite to confirm it is resolved, repeating the process until no issues remain.



Validation Results				
Validation Results				
Element	Severity	Abbreviati...	Message	
ISE&PPOOA ValSuite - Functional Architecture				
: External Input 2	error	siblings con...	Input was not found neither in parent function nor in any sibling as an output.	
: External Output 1	error	siblings con...	Output was not found neither in parent function nor in any sibling as an input.	
: External Output 2	error	siblings con...	Output was not found neither in parent function nor in any sibling as an input.	
System Functions	warning	function wit...	Functions are considered transformations and they must have one input at least one input	

Confirm All Findings Are Resolved (2/2)



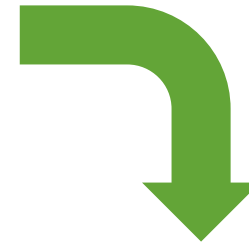
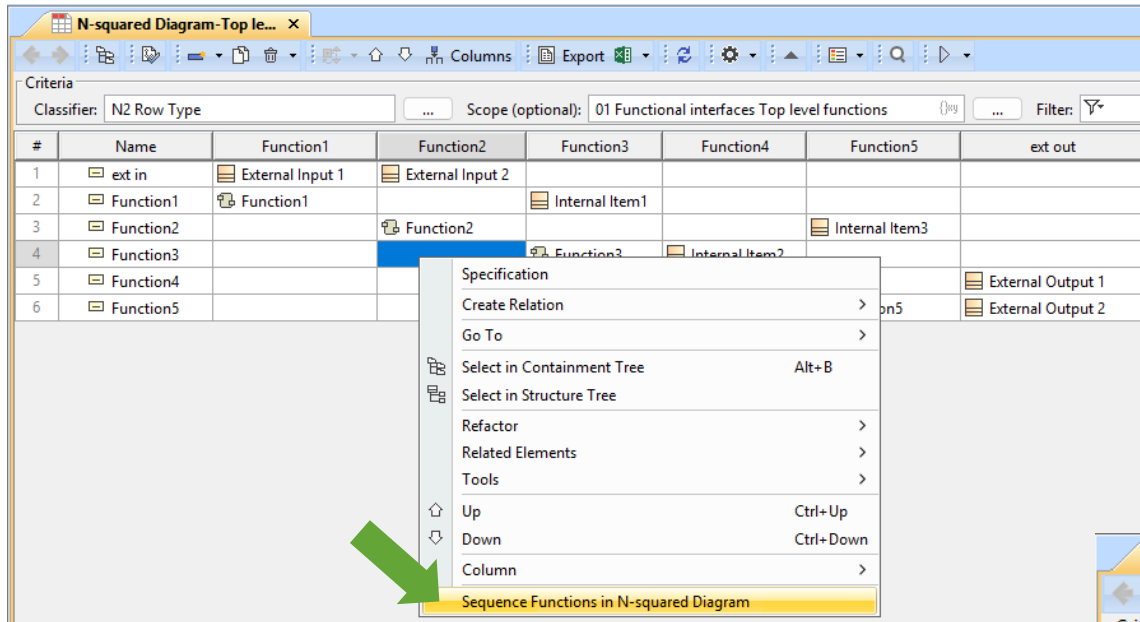
#	Name	Function 1	Function 2	Function 3	Function 4	Function 5	ext out
1	ext in	External Input 1			External Input 2		
2	Function 1	Function 1	Internal Item 1				
3	Function 2		Function 2	Internal Item 2			
4	Function 3			Function 3			External Output 1
5	Function 4				Function 4	Internal Item 3	
6	Function 5					Function 5	External Output 2

Sequence the N-squared Diagram

The ISE&PPOOA Profile allows users to sequence N-squared Diagrams easily. To sequence a N-squared Diagram,

1. Open the N-squared Diagram
2. Right-click anywhere within the diagram
3. Select “Sequence Functions in N-squared Diagram”
4. (Optional) Re-order the functions manually

Select “Sequence Functions In N-squared Diagram”



The screenshot shows the 'N-squared Diagram-Top le...' window after the 'Sequence Functions in N-squared Diagram' action. The table below shows the resulting sequence of functions.

#	Name	Function1	Function3	Function4	Function2	Function5	ext out
1	ext in	External Input 1			External Input 2		
2	Function1	Function1	Internal Item1				
3	Function3		Function3	Internal Item2			
4	Function4			Function4			External Output 1
5	Function2				Function2	Internal Item3	
6	Function5					Function5	External Output 2

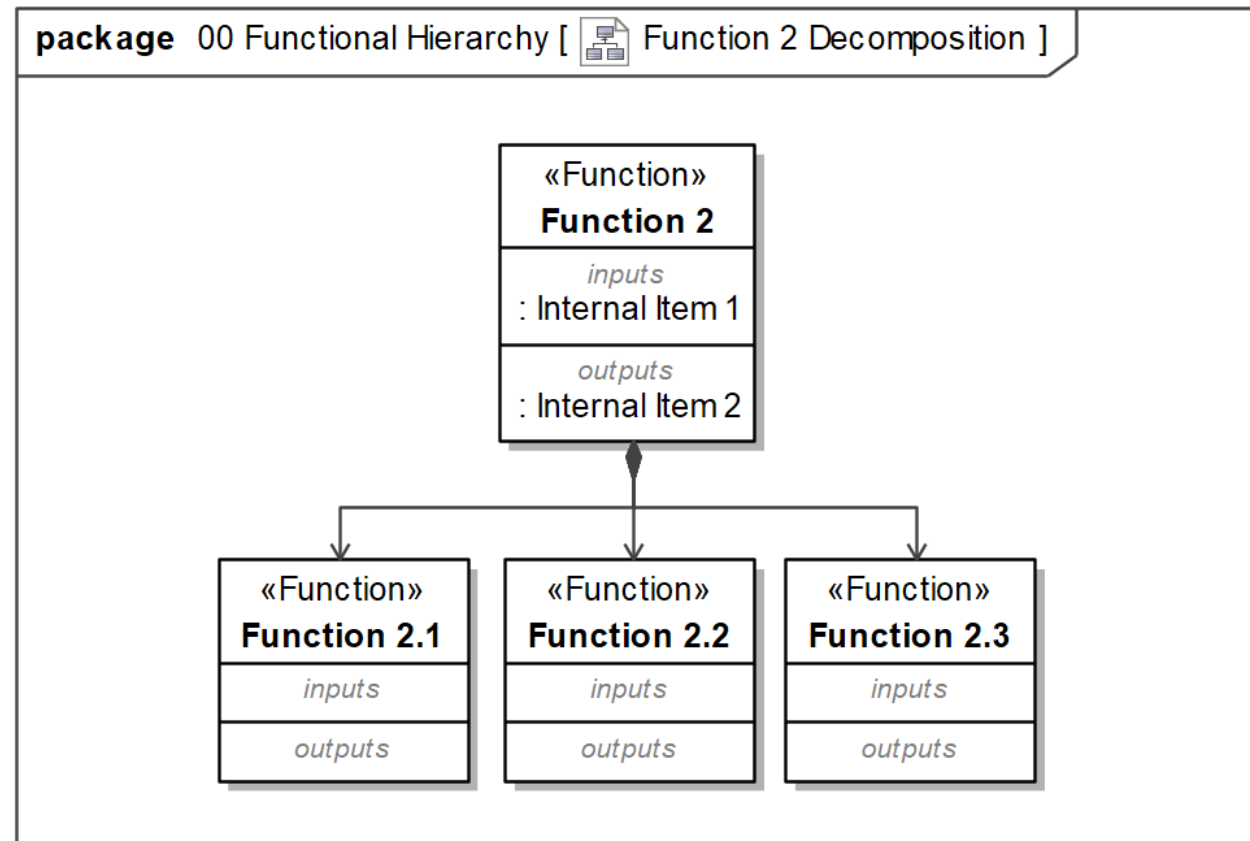
5.4 Iterate on Child Functions

Iterate on Child Functions

Select a child function and repeat the whole process to expand the functional analysis

1. Define Functional Hierarchy for the selected function
2. Define Functional Interfaces for the child functions of the selected function
3. Validate Functional Consistency of the child functions

Define Functional Hierarchy for Selected Function



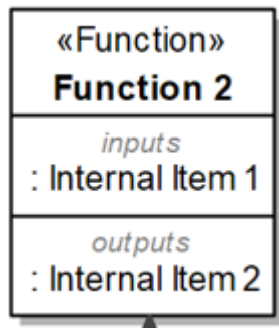
Define Functional Interfaces for its Children

Create the N squared diagram for the child functions of the selected function

#	Name	Function 2.1	Function 2.2	Function 2.3	ext out
1	 ext in				
2	 Function 2.1	 Function 2.1			
3	 Function 2.2		 Function 2.2		
4	 Function 2.3			 Function 2.3	

Define Functional Interfaces for its Children

1. Add the inputs of the selected function to its child functions which transform them
2. Add the outputs of the selected function to its child functions which produce them



#	Name	Function 2.1	Function 2.2	Function 2.3	ext out
1	 ext in	 Internal Item 1			
2	 Function 2.1	 Function 2.1			
3	 Function 2.2		 Function 2.2		
4	 Function 2.3			 Function 2.3	 Internal Item 2

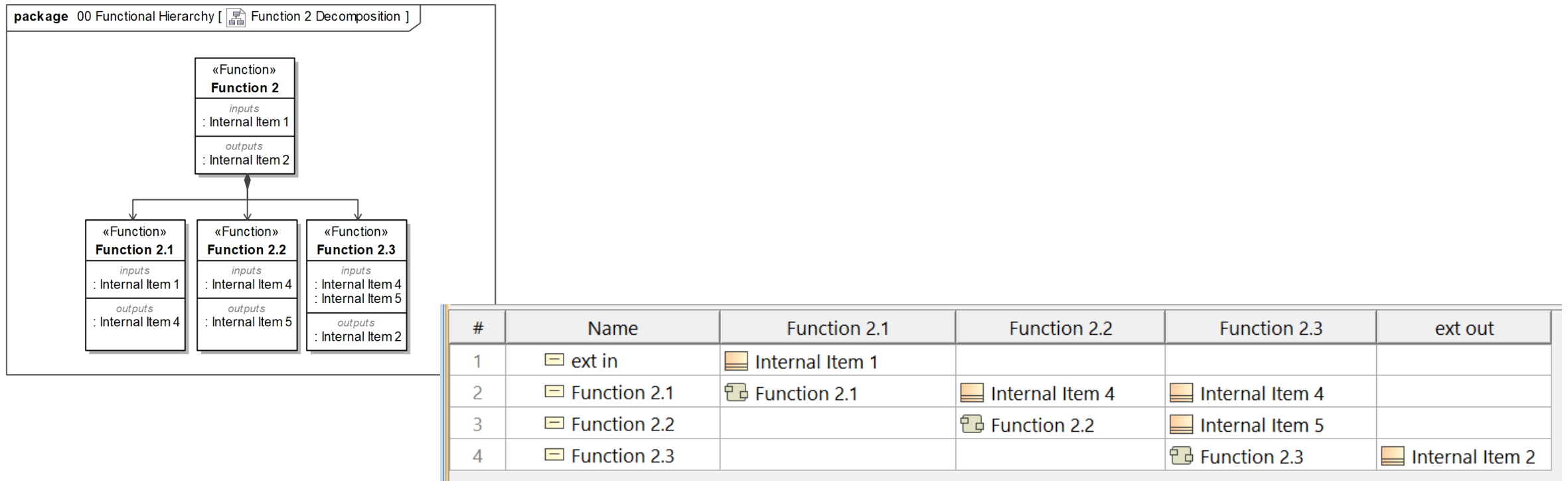
Define Functional Interfaces for F2 Children

Define the internal items which are exchanged between the child functions and add them in the N-squared diagram accordingly.

#	Name	Function 2.1	Function 2.2	Function 2.3	ext out
1	 ext in	 Internal Item 1			
2	 Function 2.1	 Function 2.1	 Internal Item 4	 Internal Item 4	
3	 Function 2.2		 Function 2.2	 Internal Item 5	
4	 Function 2.3			 Function 2.3	 Internal Item 2

Validate Functional Consistency

Validate and fix all issues for the updated functions



Continue...

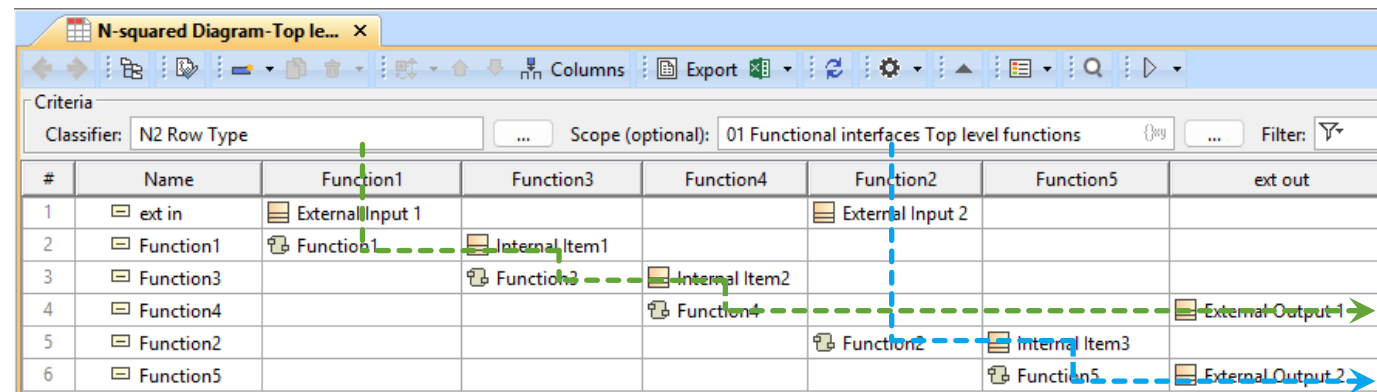
Continue until all children functions are defined accordingly.

5.5 Define Functional Flows

Overview

Once the functional hierarchies are modeled and the out-in dependencies are identified from the N-squared diagram, it is quite easy for the systems engineer to build up the **functional flows** to depict how different functions are used and activated to transform the inputs to the system into the required outputs.

The **out-in dependencies identified in the N-squared charts** are critical to represent functional sequences.



Create Functional Flows

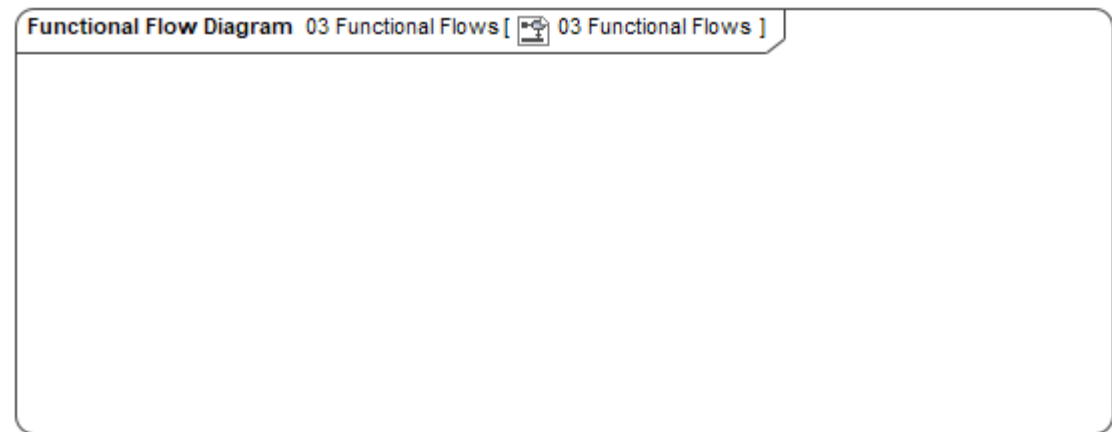
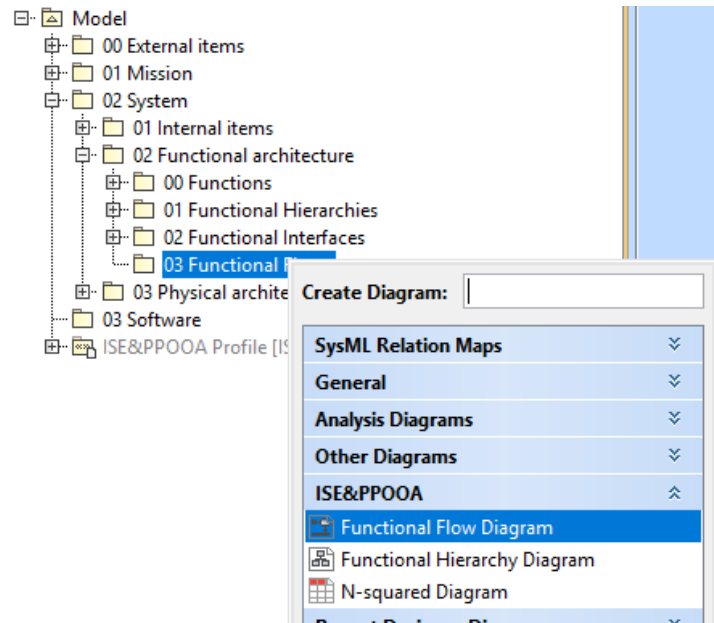
In a Package within the Functional Architecture:

1. **Create a Functional Flow Diagram**
2. **Add used functions** to the functional flow
3. Represent control flows to show the activation of functions

Repeat the process until all functional flows are defined

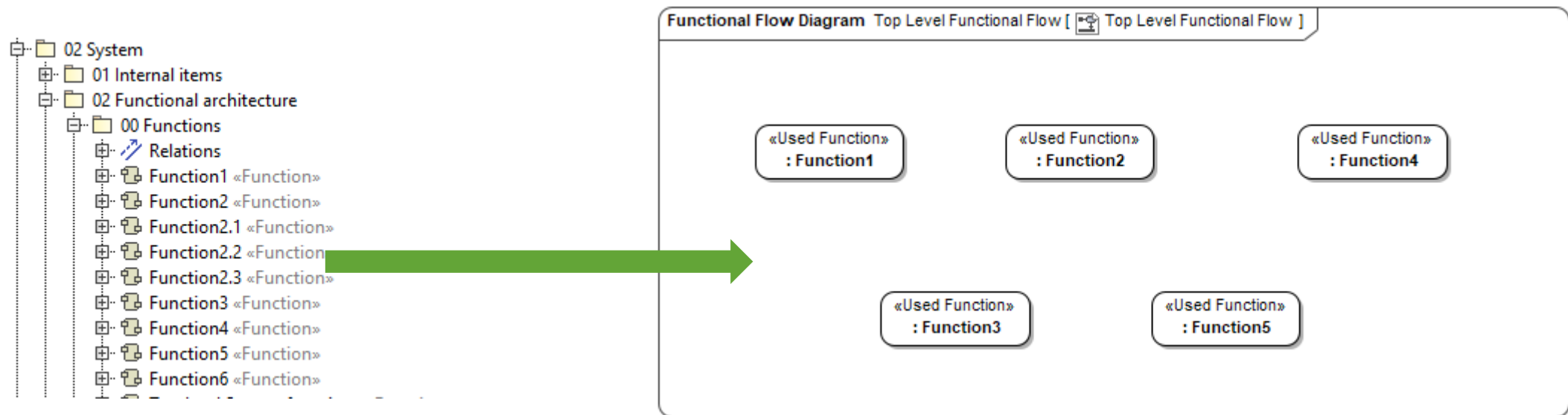
Create a Functional Flow Diagram

The ISE&PPOOA Profile provides a **special type of SysML activity diagram** to help users create **functional flows** easily.



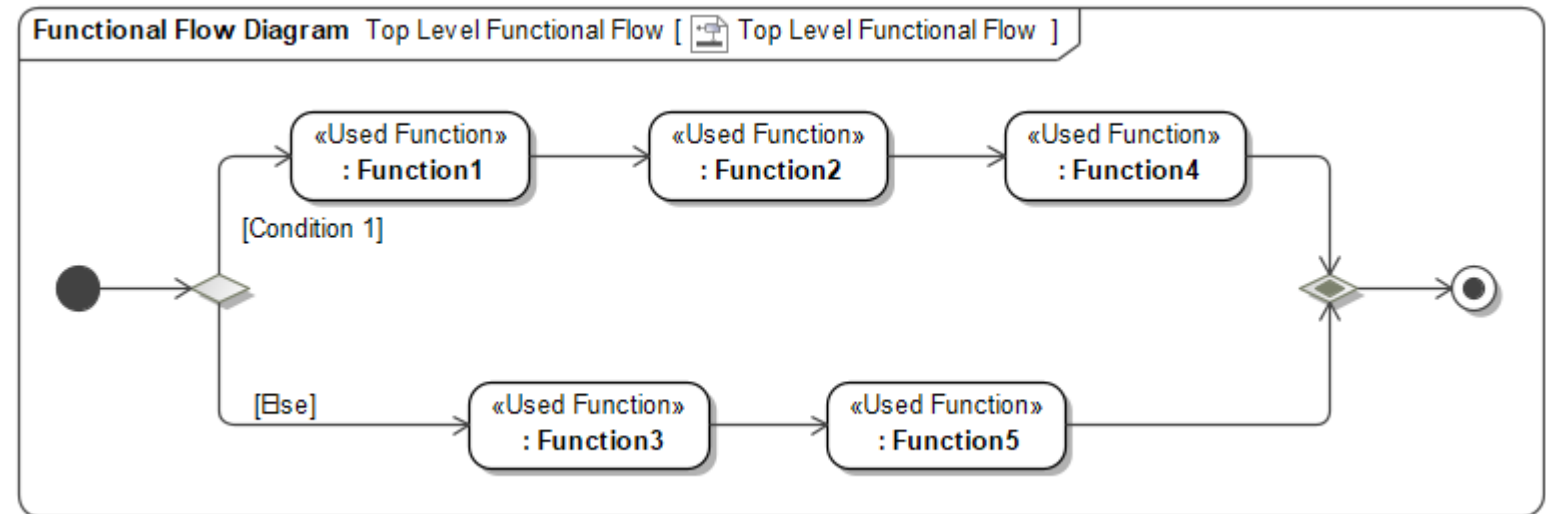
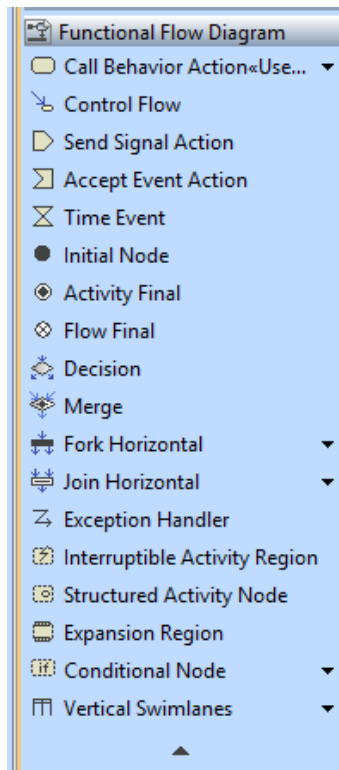
Add Used Functions to The Functional Flow

Drag functions from the containment tree onto the Functional Flow diagram to add used functions:



Add Control Flows to the Functional Flow (1/2)

Use the tools bar to add the needed control flows and nodes as it is done for SysML activity diagrams:

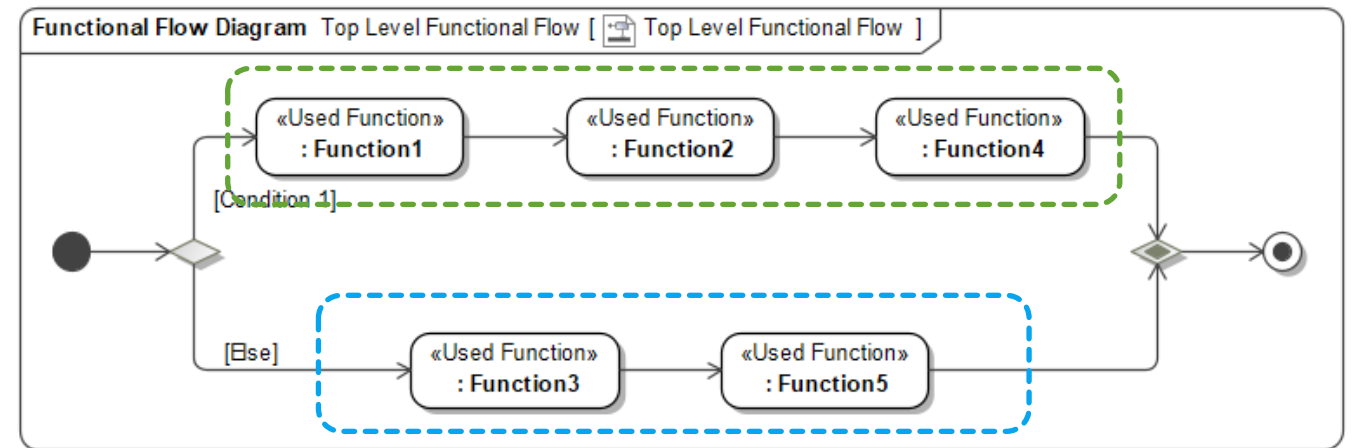


Add Control Flows to the Functional Flow (2/2)

N-squared Diagram-Top le... X

Criteria
Classifier: N2 Row Type ... Scope (optional): 01 Functional interfaces Top level functions ... Filter: Y

#	Name	Function1	Function3	Function4	Function2	Function5	ext out
1	ext in	External Input 1			External Input 2		
2	Function1	Function1	Internal Item1				
3	Function3		Function3	Internal Item2			
4	Function4			Function4			External Output 1
5	Function2				Function2	Internal Item3	
6	Function5					Function5	External Output 2



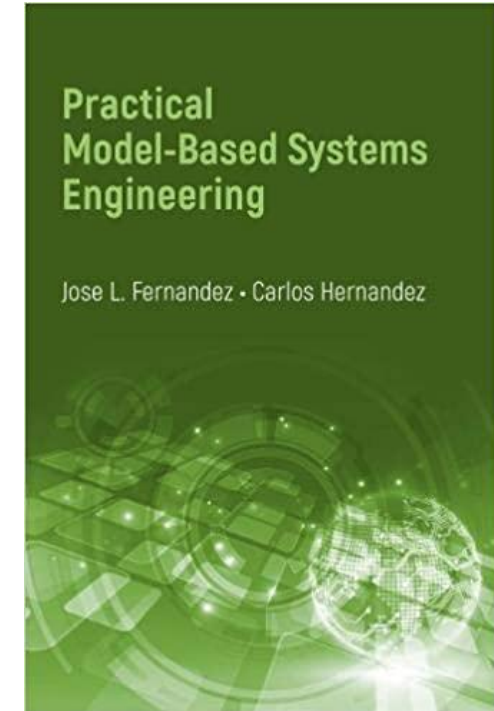
Do You Want To Know More?

Book “Practical Model-Based Systems Engineering”

Authors: Jose L. Fernandez and Carlos Hernández

Publisher: Artech House, 2019

ISBN-13:978-1-63081-579-0



ISE&PPOOA MBSE Methodology Training Course

Other resources

ISE&PPOOA MBSE methodology in OMG wiki:

<https://www.omgwiki.org/MBSE/doku.php?id=mbse:ppooa>

ISE&PPOOA official website: <http://www.ppooa.com.es/>

Cameo ISE&PPOOA Profile & Plugin Available For Free

Get the ISE&PPOOA Profile & Plugin for Cameo Systems Modeler 2024x by contacting us via email ise&ppooa@ppooa.com.es with your contact details (name, institution or company, location, and professional email address).