S*Metamodel Mapping

for

MagicDraw/Cameo Systems Modeler

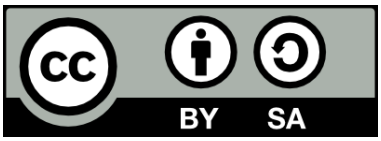
Version 19

Version 1.11.12
2/28/2024

By: S* Patterns Community

Revision History

Date	Version	Description	Author
November 01. 2013	1.0	Initial Content	William Schindel, ICTT
January 19, 2015	1.1	Updated detail mapping	Jason Sherey, ICTT
December 9, 2016	1.2	Add S* Icon Table	Stephen Lewis, ICTT
January 23, 2017	1.3	Table and Scope Edits	Stephen Lewis, ICTT
January 27, 2017	1.4	Table Implementation Statuses	Stephen Lewis, ICTT
March 6-8, 2017	1.5	Updates to Implementation Statuses, SysML Elements Mapping	Stephen Lewis, ICTT
March 9, 2017	1.6	Profile Installation Section	Stephen Lewis, ICTT
June 9, 2017	1.7	Implementation Status Updates	Stephen Lewis, ICTT
June 13, 2017	1.8	Implementation Status Column Edits	Stephen Lewis, ICTT
April 26, 2019	1.9.1	Updated formats and mappings	Jason Sherey, ICTT
November 22, 2019	1.9.1a	Create Commons License nomenclature	Jason Sherey, ICTT
June 24, 2021	1.10.1	Updated mappings	Stephen Lewis, ICTT
January 12, 2022	1.11.1	Changed “Physical System” language to “Design Component”, Changed Architectural Relationship and Architectural Relationship Role Mappings, Added Interface Element Relationship to S* Metaclass Mapping Details, Added Relates AR, Relates FI, Relates IO, Relates Sys to S* Metarelationship Mapping Details. Edited Mapping Strategy section.	Stephen Lewis, ICTT
October 12, 2022	1.11.2	Updated Reference Details	Stephen Lewis, ICTT
February 10-13, 2023	1.11.3-4	Risk Analysis Mappings	Stephen Lewis, ICTT
February 14-15, 2024	1.11.5	Allowed Value, Can Have Value, Requirement Transfer Function, Stakeholder Requirement	Stephen Lewis, ICTT
February 16-28	1.11.6-12	Section 4.3 creation, Sections 4.1-2 layout	Stephen Lewis, ICTT



Corporate Officer
ICTT System Sciences
378 South Airport Street
Terre Haute, IN 47803
812-232-2208

Table of Contents

1 Document Overview

1.1 Document Scope

This document describes the mapping between OMG SysML® and the S*Metamodel of the Systematica® methodology and its models and patterns. This document is further configured for Dassault Systemes MagicDraw/Cameo Systems Modeler Version 19. This mapping is implemented in version 05082023 of the Systematica Profile in MagicDraw/Cameo Systems Modeler.

1.2 Document Organization

This document is organized in the following fashion:

1. **Document Overview:** Reviews the scope, organization and references for this document.
2. **Framework Overview:** Reviews basic OMG SysML schema structures and concepts.
3. **Mapping Overview:** Provides a high-level overview of how the schema configuration uses the structures and concepts described in Section 2 and their mapping to S*Metaclasses and S*Metarelationships.
4. **Detailed Mapping:** Lists each S*Metaclass and S*Relationship and describes how each is mapped into MagicDraw.

1.3 Applicable Documents

The following are referenced or otherwise applicable to this document:

1. Friedenthal, S., et al.. *A Practical Guide to SysML*. Boston: Morgan Kaufmann OMG Press, 2012.
2. ICTT System Sciences. “Systematica Metamodel, Version 8.0.57.” Feb 2024.
3. ICTT System Sciences. “S*Metamodel to SysML Map.” *Visualizing and Understanding Systems Processes and Systems Pathologies, Volume I: Modeler’s Guide, V1.6.2*. INCOSE System Sciences Working Group, SP Modeling Team, December, 2013.
4. Object Modeling Group (OMG). “What Is SysML.” *OMG SysML*, 2018, omgsysml.org/what-is-sysml.htm. Accessed 18 Oct. 2018.

2 OMG SysML Overview

"The OMG Systems Modeling Language (OMG SysML™) is a general-purpose graphical modeling language for specifying, analyzing, designing, and verifying complex systems" (OMG). Being a graphical modeling language, SysML uses elements on a variety of diagrams to model a system's requirements and designs.

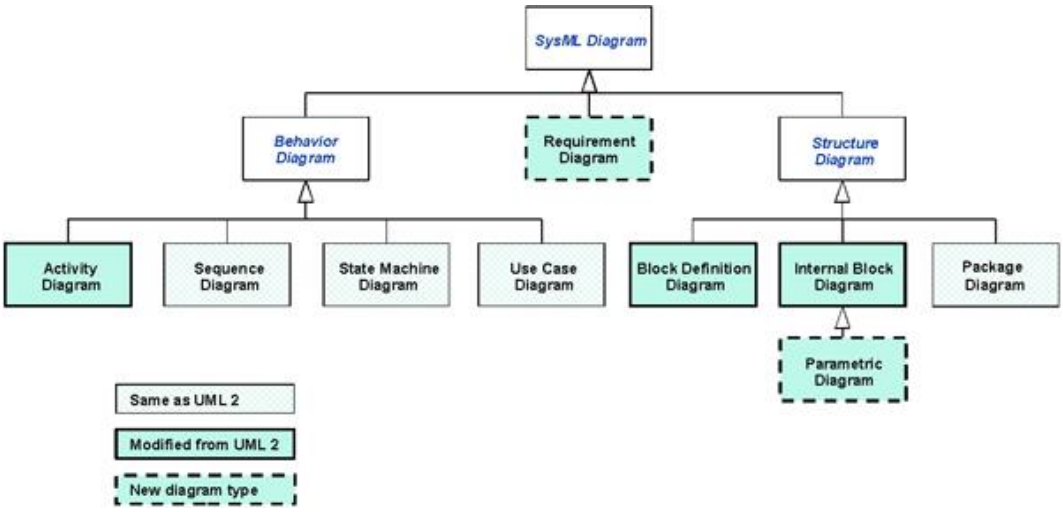


Figure 1: SysML Diagram Types (www.omgsysml.org)

"The «block» is the basic unit of structure in SysML and can be used to represent hardware, software, facilities, personnel, or any other system element. The system structure is represented by block definition diagrams and internal block diagrams. A block definition diagram describes the system hierarchy and system/component classifications. The internal block diagram describes the internal structure of a system in terms of its parts, ports, and connectors. The package diagram is used to organize the model" (OMG).

"The behavior diagrams include the use case diagram, activity diagram, sequence diagram, and state machine diagram. A use-case diagram provides a high-level description of functionality that is achieved through interaction among systems or system parts. The activity diagram represents the flow of data and control between activities. A sequence diagram represents the interaction between collaborating parts of a system. The state machine diagram describes the state transitions and actions that a system or its parts perform in response to events" (OMG).

"SysML includes a graphical construct to represent text-based requirements and relate them to other model elements. The requirements diagram captures requirements hierarchies and requirements derivation, and the satisfy and verify relationships allow a modeler to relate a requirement to a model element that satisfies or verifies the requirements. The requirement diagram provides a bridge between the typical requirements management tools and the system models" (OMG).

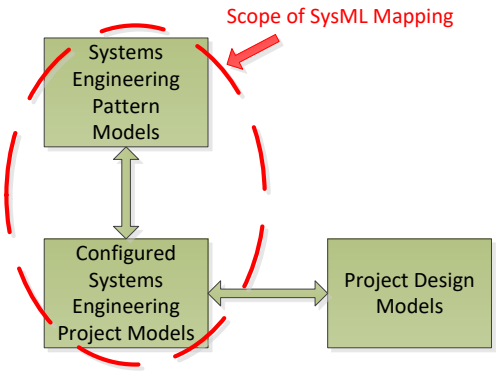
“The parametric diagram represents constraints on system property values such as performance, reliability, and mass properties, and serves as a means to integrate the specification and design models with engineering analysis models” (OMG).

“SysML also includes an allocation relationship to represent various types of allocation, including allocation of functions to components, logical to physical components, and software to hardware” (OMG).

3 Schema Configuration Overview

The mapping described in this document covers only systems engineering pattern and configured model spaces. Mappings for detail design models and any relationships between them and the configured systems engineering models are not contained in this document.

Figure 2: Mapping Model Scope



3.1 Mapping Summaries

The mapping for using OMG SysML with Systematica can be summarized by the following mapping between the Summary S*Metamodel Diagram and the SysML Mapping Overview Diagram.

Figure 3: Systematica Summary Meta-Model Diagram

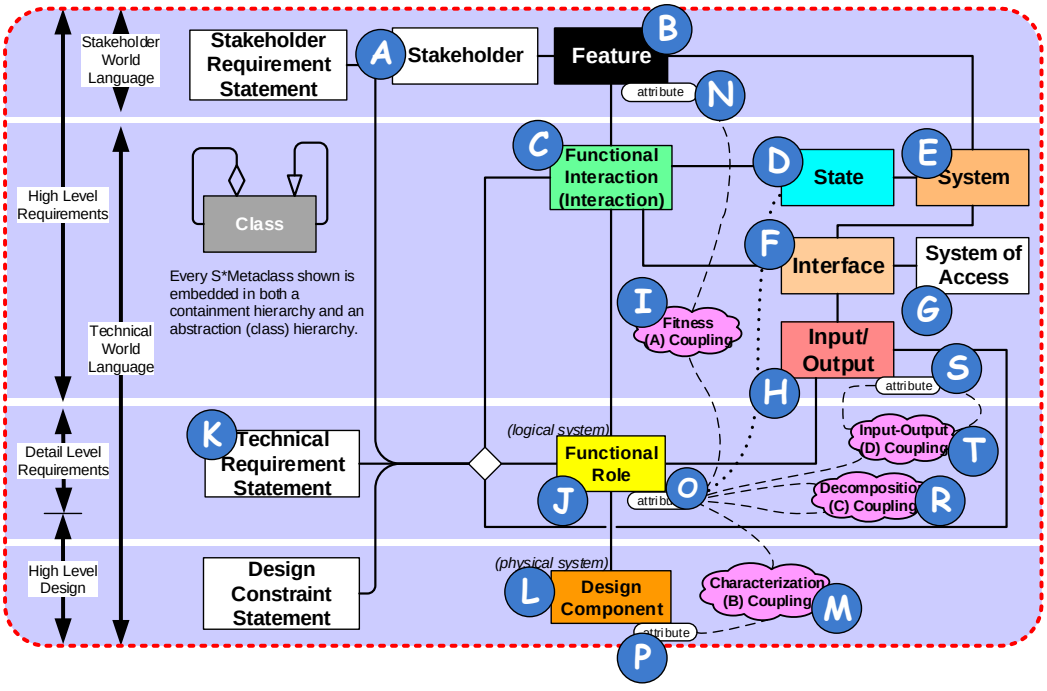
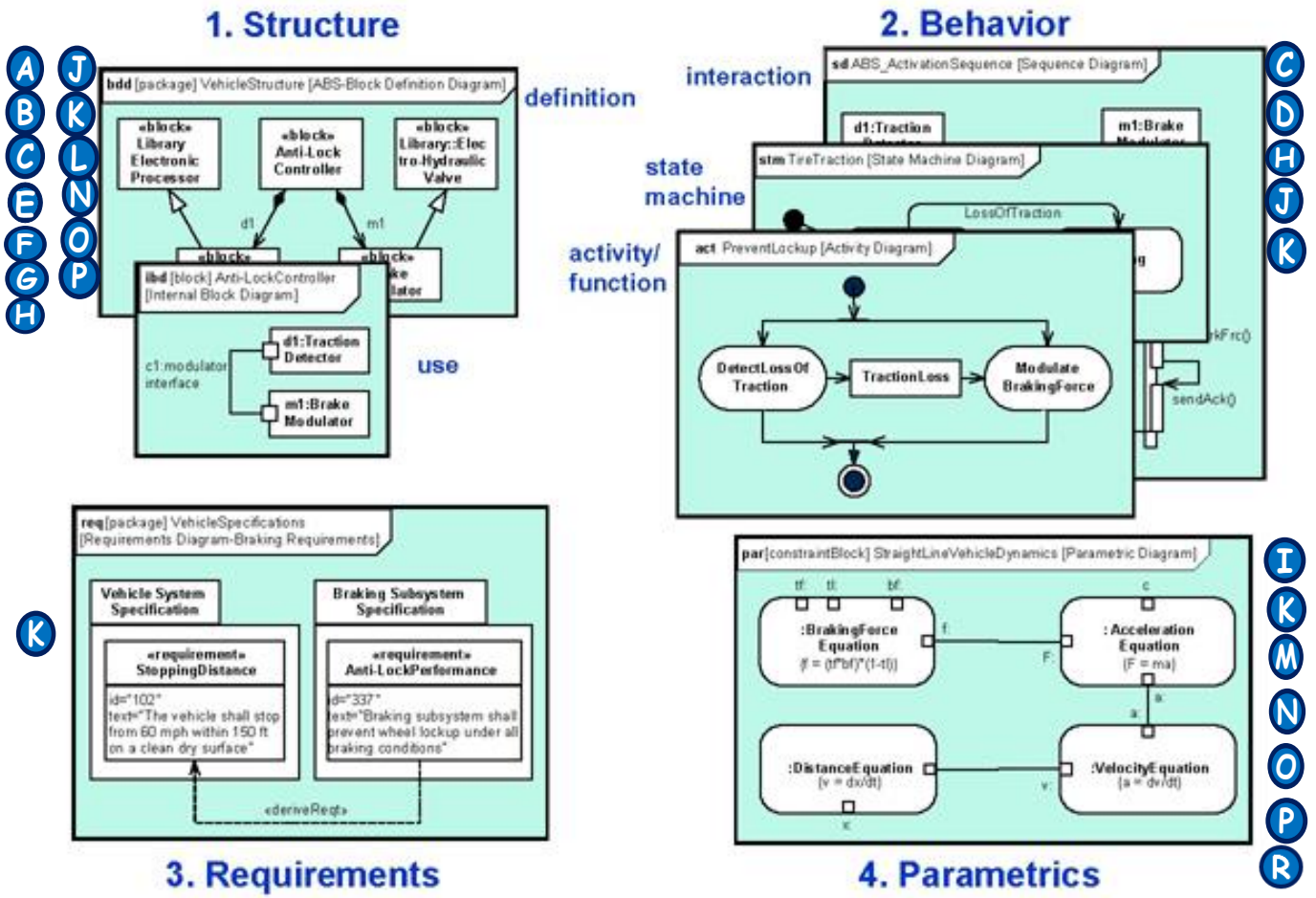


Figure 4: Systematica OMG SysML Mapping Overview Diagram



Note that the Package and Use Case diagrams are not shown in this example, but are respectively part of the structure and behavior pillars

A summary Systematica Metamodel mapping to OMG SysML is described by the following table including a legend for the letters shown in Figure 4 above:

Mapping Letter	Systematica Metamodel Element	SysML Element Type
A	Stakeholder	Block with «Stakeholder» stereotype
B	Feature	Block with «Feature» stereotype
C	Functional Interaction (Interaction)	Activity with «Functional Interaction» stereotype
D	State	State with «State» stereotype
E	System	Block with «Logical System» stereotype
F	Interface	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype
G	System of Access	Block with «System of Access» stereotype
H	Input/Output	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
I	Fitness (A) Couplings	Constraint Block with «Fitness Coupling» stereotype
J	Functional Role	Block with «Logical System» stereotype being referenced by an Interaction via a shared aggregation relationship with «Has Role» stereotype.
K	Requirement Statement	Requirement with «Requirement Statement» stereotype
L	Design Component	Block with «Design Component» stereotype
M	Characterization (B) Couplings	Constraint Block with «Characterization Coupling» stereotype
N	Feature Attribute	Attribute with «Feature Primary Key» or «Feature Attribute» stereotypes
O	Role Attribute	Attribute with «Logical System Attribute» stereotype
P	Design Component Attribute	Attribute with «Design Component Attribute» stereotype
R	Decomposition (C) Matrix Couplings	Constraint Block with «Decomposition Coupling» stereotype
S	IO Attribute	Attribute with «IO Attribute» stereotype
T	Input-Output (D) Coupling	Constraint Block with «Input Output Coupling» stereotype

3.2 High Level Mapping Strategy

The mapping was constrained to only map the classes and relationships of the S*Metamodel. The Systematica Process Views and Transactions are not mapped in this document.

The main mapping choices consisted of:

- Mapping most S* classes and relationships to standard SysML elements staying as close to the intended use of each SysML element as possible,
- Systematica Configured Model classes are mapped to SysML classes whose generalizations are the Systematica Pattern classes,
- Where no standard SysML element matches a Systematica class, a SysML “block” is used with a specialized stereotype,
- Where no standard SysML relationship matches a Systematica relationship, a SysML “dependency” is used with a specialized stereotype,
- Required, Capability, Baseline, Best In Class, or other value types of each Modeled Attribute are mapped to tags of those Modeled Attributes,

- To allow for possible use of SysML Activity Diagrams, SysML Activities are used to model parts of S* Interaction-Role-Requirement relationships and classes even if no diagram is actually drawn,
- S* Architectural Relationships have been bifurcated into simple and reified cases for mapping. In the simple case, it is mapped to the Association relationship between systems and given an Architectural Relationship stereotype. This is a natural fit for the simple case because SysML associations are restricted to only 2 roles. In the reified case, which involves more than 2 roles, it is mapped to a block with Architectural Relationship Definition. That block is then able to participate in dependency relationships with the N number of Architectural Relationship Roles.
- Using SysML classes instead of parts or properties to model components and referenced classes. The components are modeled under the container's namespace and also participate in a composition with the container. This results in minimal use of the parts that occur from the association because Systematica emphasizes that a class is defined by its relationships across all of its uses as opposed to a class being defined as a library entity with independent contextual references.

4 Detail Mapping

*4.1 S*Metaclass Mapping Details*

The Systematica Class mappings to SysML is detailed by the following table:

Systematica Modeled Class Name	SysML Element
Allowed Value	Block with «Allowed Value» stereotype
Architectural Relationship	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype
Architectural Relationship Role	Simple: Association Target and Source Roles; Reified: Dependency with «Has AR Role» stereotype
Attribute Coupling	Constraint Block with «Attribute Coupling» stereotype
Attribute Coupling Map	Constraint of a Constraint Block, Artifact referenced or linked to by Constraint
Attribute Role	Parameter of Constraint Blocks
Class	This is a superclass of other Metaclasses and is not mapped.
Counter Requirement	Requirement with «Counter Requirement» stereotype
Design Component	Block with «Design Component» stereotype
Design Component Attribute Role	Parameter of Constraint Blocks
Design Constraint	Requirement with «Design Constraint» stereotype
Design Constraint Statement	Requirement with «Design Constraint» stereotype
Design Coupling	Constraint Block with «Characterization Coupling» stereotype.
Design Coupling Map	Constraint within Constraint Block, Artifact referenced or linked to by Constraint
Domain	Block with «Domain» stereotype
Domain System	Block with «Logical System» stereotype aggregated within a Domain
Event	Trigger with «Event» stereotype
Failure Impact	Block with «Failure Impact» stereotype
Failure Mode	State with «Failure Mode» stereotype
Failure Mode Context Element	Block with «Failure Mode Context Element» stereotype
Feature	Block with «Feature» stereotype
Feature Attribute Role	Parameters of Constraint Blocks
Functional Interaction	Activity Block with «Functional Interaction» stereotype.
Functional Role	Block with «Logical System» stereotype and in Aggregation relationship with «Has Role» stereotype with Interaction Activity Block

Systematica Modeled Class Name	SysML Element
Information Input-Output	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Input-Output	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Input-Output Attribute Role	Parameter of Constraint Blocks
Input-Output Coupling	Constraint Block with «IO Coupling» stereotype.
Input-Output Coupling Map	Constraint within Constraint Block, Artifact referenced or linked to by Constraint
Input Role	Action Pin on an Activity modeling a Requirement Relationship
Interface	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype
Interface Element	Block with «Interface Element» stereotype.
Logical System	Block with «Logical System» stereotype
Logical System (Advocate)	Logical System Block with «Advocate» stereotype that only participates in advocate-type relationships
Logical System (Stakeholder)	Logical System Block with «Stakeholder» stereotype that only participates in stakeholder-type relationships
Modeled Attribute (Feature Attribute)	Attribute with «Feature Attribute» or «Feature Primary Key» stereotypes
Modeled Attribute (IO Attribute)	Attribute with «IO Attribute» stereotype
Modeled Attribute (Physical System Attribute)	Attribute with «Physical System Attribute» stereotype
Modeled Attribute (Role Attribute)	Attribute with «Logical System Attribute» stereotype
Modeled Relationship	This is a superclass of other meta-classes and is not mapped.
Modeled Relationship Role	This is a superclass of other meta-classes and is not mapped.
Modeled Statement	This is a superclass of other meta-classes and is not mapped.
Output Role	Action Pin on an Activity modeling a Requirement Relationship
Physical Input-Output	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Port	Flow Property with «Input Output» stereotype.
Rationale	«allocate» dependency
Requirements Coupling	Constraint Block with «Fitness Coupling», «Decomposition Coupling», or «Input Output Coupling» stereotype
Requirements Coupling Map	Constraint within Constraint Block, Artifact referenced or linked to by Constraint
Requirement Transfer Function	Activity with «Requirement Transfer Function» stereotype
Requirement Statement	Requirement «Requirement Statement» stereotype

Systematica Modeled Class Name	SysML Element
Role Attribute Role	Parameter of Constraint Blocks
Stakeholder Requirement	Requirement with «Need» stereotype
State	State with «State» stereotype
System	Block with «Logical System» stereotype.
System of Access (SOA)	Logical System block with «System of Access» stereotype.
Transition	SysML Transition with «Systematica Transition» stereotype.
Value	Required, Capability, Baseline, Best In Class, or other value types of each Modeled Attribute are mapped to tags of those Modeled Attributes

*4.2 S*Metarelationship Mapping Details*

The S*Metarelationship mappings to SysML can be summarized by the following table:

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Advocates	Dependency with «Advocated_By (N-ADV)» stereotype	Logical System Block with «Advocate» stereotype that only participates in advocate-type relationships	Requirement with «Need» stereotype
Abnormal State Of	Dependency with «Abnormal State Of» stereotype	State with «Failure Mode» stereotype	Block with «Design Component» stereotype
Allocated To (Design Component-Functional Role)	Association with «Functional Role Allocation» stereotype for LS-DC	Block with «Logical System» stereotype	Block with «Design Component» stereotype
Allocated To (Modeled Attribute-Attribute Role)	Binding Connection for Attribute-Constraint Property or connection between Flow -Action Pin	Parameter of Constraint Blocks	Attribute with «Feature Attribute» or «Feature Primary Key» or «IO Attribute» or «Logical System Attribute» or «Physical System Attribute» stereotypes

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Allocated To (Architectural Relationship Role-Functional Role)	The trace between a System and the end of a Connector that conveys «Architectural Relationship Definition» items, sometimes indirectly through an Interface port	Block with «Logical System» stereotype	Simple: Association Target and Source Roles; Reified: Dependency with «Has AR Role» stereotype
Appears In	Aggregation between Domain and other classes represented as Blocks.	Any mapped S* class	Block with «Domain» stereotype
Benefits	Dependency with «Benefits (FTR-STK)» stereotype	Logical System Block with «Stakeholder» stereotype that only participates in stakeholder-type relationships	Block with «Feature» stereotype
Can Have Value	Dependency with «Can Have Value» stereotype	Attribute with «Feature Attribute» or «Feature Primary Key» or «IO Attribute» or «Logical System Attribute» or «Physical System Attribute» stereotypes	Block with «Allowed Value» stereotype
Causes Behavior	Dependency with «Causes Behavior» stereotype	State with «Failure Mode» stereotype	Requirement with «Counter Requirement» stereotype
Causes Impact	Dependency with «Causes Impact» stereotype	Requirement with «Counter Requirement» stereotype	Block with «Failure Impact» stereotype
Contains	Composition	Any mapped S* class	Any mapped S* class
Derived From	«trace» dependency	Any mapped S* class	Any mapped S* class

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Emerges From	Dependency with «Resolves (IO-AR)» stereotype	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Exemplifies	Dependency with «Exemplified By (AR-IO)» stereotype	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype
Groups	«Input Output» flow property owned by «Interface Definition» interface block	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype	Flow Property with «Input Output» stereotype.
Has Advocate	Dependency with «Represents (ADV-STK)» stereotype	Logical System Block with «Stakeholder» stereotype that only participates in stakeholder-type relationships	Logical System Block with «Advocate» stereotype that only participates in advocate-type relationships
Has Attribute	Block-Block Attribute SysML relationship	Block with «Feature» stereotype	Attribute with «Feature Attribute» or «Feature Primary Key» stereotypes

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Has Attribute	Block-Block Attribute SysML relationship	Block with «Design Component» stereotype	Attribute with «Physical System Attribute» stereotype
Has Attribute	Block-Block Attribute SysML relationship	Block with «Logical System» stereotype	Attribute with «Logical System Attribute» stereotype
Has Feature	Dependency with «Offered By (FT-SYS)» stereotype	Block with «Logical System» stereotype	Block with «Feature» stereotype
Has Previous	MagicDraw/CSM Configuration Management options	Any mapped S* class	Any mapped S* class
Has Role (Functional Role-Interaction)	Aggregation with «Has Role» stereotype between Interaction and Functional Role	Activity Block with «Functional Interaction» stereotype.	Block with «Logical System» stereotype
Has Role (Attribute Coupling-Attribute Role)	Constraint Block-Property composition	Constraint Block with «Attribute Coupling» stereotype	Parameter of Constraint Blocks
Has Role (Requirement Transfer Function-Input Role/Output Role)	Activity-Action Pin composition	Activity with «Requirement Transfer Function» stereotype	Action Pin on an Activity modeling a Requirement Relationship

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Has Role (Architectural Relationship-Architectural Relationship Role)	Connector-Connector End relationship	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype	Simple: Association Target and Source Roles; Reified: Dependency with «Has AR Role» stereotype
Has Stakeholder	Dependency with «Serves (LS-STK)» stereotype	Block with «Logical System» stereotype	Logical System Block with «Stakeholder» stereotype that only participates in stakeholder-type relationships
Has State	Dependency with «Chronicles (ST-LS)» stereotype	Block with «Logical System» stereotype	State with «State» stereotype
Has Subject	Aggregation with stereotype «Has Subject»	Block with «Domain» stereotype	Block with «Logical System» stereotype
Has Value	value property's-tag value relationship	Attribute with «Feature Attribute» or «Feature Primary Key» or «IO Attribute» or «Logical System Attribute» or «Physical System Attribute» stereotypes	Required, Capability, Baseline, Best In Class, or other value types of each Modeled Attribute are mapped to tags of those Modeled Attributes
Has View (Requirement Transfer Function-Requirement Statement)	Dependency with «Satisfy» stereotype	Activity with «Requirement Transfer Function» stereotype	Requirement «Requirement Statement» stereotype

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Has View (Requirement Transfer Function-IO Coupling)	Dependency with «Is Specified By (RTF-IO CPL)» between Requirement Transfer Function activity block and IO Attribute Coupling constraint block.	Activity with «Requirement Transfer Function» stereotype	Constraint Block with «IO Coupling» stereotype.
Impacts Feature	Dependency with «Impacts Feature» stereotype	Block with «Failure Impact» stereotype	Block with «Feature» stereotype
Interacts Through	«System»-«ifc port»-«Input Output» or «System»-«Input Output» containment/nesting	Block with «Logical System» stereotype	Flow Property with «Input Output» stereotype.
Is A Type Of	Generalization	Any mapped S* class	Any mapped S* class of same type
Is A Type Of (Pattern Element-Configured Element)	Dependency with «Is Configuration Of» stereotype indicates pattern sources of configured model classes	Any mapped S* Pattern class	Any mapped S* Configured Model class
Is Constrained By	«specifies» dependency between Design Constraint and other classes	Block with «Design Component» stereotype	Requirement with «Design Constraint» stereotype
Is Facilitated By Externally	«Facilitated By (Port-SOA)» dependency	Flow Property with «Input Output» stereotype.	Logical System block with «System of Access» stereotype.
Is Facilitated By Internally	«Facilitated By (Port-SOA)» dependency	Flow Property with «Input Output» stereotype.	Logical System block with «System of Access» stereotype.

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Is Linked By Externally	Dependency with «Resolves (Port-AR)» stereotype	Flow Property with «Input Output» stereotype.	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype
Is Linked By Internally	Dependency with «Resolves (Port-AR)» stereotype	Flow Property with «Input Output» stereotype.	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype
Is Specified By	«allocate» dependency from «Requirement Transfer Function» activity block to Logical System block	Block with «Logical System» stereotype	Activity with «Requirement Transfer Function» stereotype
Is Triggered By	SysML transition-trigger association	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype	Trigger with «Event» stereotype
Is Used During	Abstraction with «Is Used During (IO-FI)» stereotype	Flow Property with «Input Output» stereotype.	Activity Block with «Functional Interaction» stereotype.

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Perceives	Dependency with «Perceived By (N-STK)» stereotype	Logical System Block with «Stakeholder» stereotype that only participates in stakeholder-type relationships	Requirement with «Need» stereotype
Permits Architectural Relationship	Ownership between an Interface block with «Interface Definition» stereotype that owns a flow property with «Architectural Relationship» stereotype	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype
Permits Functional Interaction	Dependency with «Permits FI (IFC-FI)» stereotype	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype	Activity Block with «Functional Interaction» stereotype.
Permits Input-Output	Ownership between an Interface block with «Interface Definition» stereotype that owns a flow property with «Input Output» stereotype	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Permits SOA	Dependency with «Permits SOA (IFC-SOA») stereotype	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype	Logical System block with «System of Access» stereotype.
Provides Failure Context	Dependency with «Provides Failure Context» stereotype	Block with «Failure Mode Context Element» stereotype	Activity Block with «Functional Interaction» stereotype.

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Provides Failure Mode Context	Dependency with «Provides Failure Mode Context» stereotype	Block with «Failure Mode Context Element» stereotype	State with «Failure Mode» stereotype
Provides Context	Interaction activity owning a Requirement Transfer Function activity block	Activity Block with «Functional Interaction» stereotype.	Activity with «Requirement Transfer Function» stereotype
Provides Interface	Owns relationship between Block and Proxy Port with «ifc_port» stereotype, then follow Proxy Port type definition to Interface Block with «Interface Definition» stereotype	Block with «Logical System» stereotype	Proxy Port with «ifc port» stereotype typed by an Interface Block with an «Interface Definition» stereotype
Receives	The trace from a Flow Property with «Input Output» stereotype with an in setting to its owning proxy port with «ifc_port» stereotype to an itemFlow whose conveyed classifier is the same type as the Flow Property itself.	Flow Property with «Input Output» stereotype.	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Relates AR	Dependency with «AR-IE» Stereotype	Simple: Name of the Association relationship between systems with «architectural relationship» stereotype; Reified: Block with «Architectural Relationship Definition» stereotype	Block with «Interface Element» stereotype.
Relates FI	Dependency with «FI-IE» Stereotype	Activity Block with «Functional Interaction» stereotype.	Block with «Interface Element» stereotype.

Systematica Modeled Relationship Name	SysML Relationship	Source Class (mapped SysML element)	Target Class (mapped SysML element)
Relates IO	Dependency with «IO-IE» Stereotype	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype	Block with «Interface Element» stereotype.
Relates Sys	Dependency with «LS-IE» Stereotype	Block with «Logical System» stereotype	Block with «Interface Element» stereotype.
Replaces	Dependency with «Replaces» Stereotype	Requirement with «Counter Requirement» stereotype	Requirement «Requirement Statement» stereotype
Requires	«Abstraction» dependency from a State to Functional Interaction	State with «State» stereotype	Activity Block with «Functional Interaction» stereotype.
Satisfies	Dependency with «Satisfies (FT-N)» stereotype	Requirement with «Need» stereotype	Block with «Feature» stereotype
Sends	The trace from a Flow Property with an out setting with «Input Output» stereotype to its owning proxy port with «ifc_port» stereotype to an itemFlow whose conveyed classifier is the same type as the flow property itself.	Flow Property with «Input Output» stereotype.	itemFlow that conveys a Classifier with an «Input_Output Definition» stereotype
Transitions From	SysML state-transition relationship	SysML Transition with «Systematica Transition» stereotype.	State with «State» stereotype
Transitions To	SysML state-transition relationship	SysML Transition with «Systematica Transition» stereotype.	State with «State» stereotype
Uses_Functional_Interaction	Aggregation with «Uses Functional Interaction» stereotype	Block with «Feature» stereotype	Activity Block with «Functional Interaction» stereotype.

4.3 S* Attribute Mapping Details

This section of the mapping document maps Systematica Attributes of metaclasses and metarelationships, including specific pattern configuration rule attributes and common core attributes, into SysML properties, usually tags. Additionally, this section indicates which SysML element owns those properties. This section does not map the Systematica element to the SysML element. The details of those mappings are in Sections 1 and 2 and may differ from what is shown here, when an attribute is not mapped to the same owner that its metamodel owner is mapped to.

Systematica Attributes	Attribute Owner Systematica Metamodel Element	SysML Properties	Property Owner SysML Element
Interaction Population Rule	Uses Functional Interaction (Relationship)	"FPK Value" tag	Aggregation with «Uses Functional Interaction» stereotype
Interaction PK Value Rule		"IPK Rule" tag	
Role Population Rule	Has Role (Relationship)	"IPK Value" tag	Aggregation with «Has Role» stereotype
Role PK Value Rule		"RPK Rule" tag	
Design Component Population Rule	Allocated To (Relationship)	"Configuration Rule" tag	Aggregation with «Functional Role Allocation» stereotype
Design Component PK Value Rule		"IPPK Value" tag	
Requirement Population Rule-Interaction	Requirement Transfer Function (Class)	"IPK Rule" tag	Class with «Requirement Statement» stereotype
Requirement Population Rule-Role		"RPK Rule" tag	
Requirement Statement PK Value Rule		"RSPK Rule" tag	
State Population Rule-Interaction	Requires (Relationship)	"IPK" tag	Abstraction with «Requires» stereotype
State Population Rule-Role		"RPK" tag	
State PK Value Rule		"State PK Value Rule" tag	

Systematica Attributes	Attribute Owner Systematica Metamodel Element	SysML Properties	Property Owner SysML Element
State Type		"State Type" tag	
From State PK Matching Rule	Transition (Class)	"From State PK Matching Rule" tag	Transition with «Systematica Transition» stereotype
To State PK Matching Rule		"To State PK Matching Rule" tag	
Transition PK Value Rule		"Transition PK Value Rule" tag	
Transition Type		"Transition Type" tag	
From State PK Matching Rule	Provides Event Context (Relationship)	"From State PK Matching Rule" tag	Transition with «Systematica Transition» stereotype
To State PK Matching Rule		"To State PK Matching Rule" tag	
Event PK Value Rule		"Event PK Value Rule" tag	
Interface Primary Key Value Rule	Interface Element Relationship (Class)	"Interface PK Rule" tag	Flow Property with «Input Output» stereotype.
IO Primary Key Value Rule		"IO PK Rule" tag	
Port Primary Key Value Rule		"Port PK Rule" tag	
SOA Primary Key Value Rule		"SOA PK Rule" tag	
SOA Internal/External		"SOA Internal or External" tag	
IO Direction		"Direction" tag	SysML Flow Property
Reified Architectural Relationships			
AR Primary Key Value Rule	Interface Element Relationship (Class)	"AR Primary Key Value Rule" tag	Dependency with «Has AR Role» stereotype
AR Role Primary Key Value Rule		"AR Role Primary Key Value Rule" tag	
AR Internal/External		"AR Internal/External" tag	
AR Complexity		"AR Complexity" tag	
Simple Architectural Relationships			
AR Primary Key Value Rule	Interface Element Relationship (Class)	"AR Primary Key Value Rule" tag	

Systematica Attributes	Attribute Owner Systematica Metamodel Element	SysML Properties	Property Owner SysML Element
AR Internal/External		"AR Internal/External" tag	Item Flow with «AR Item Flow» stereotype
AR Complexity		"AR Complexity" tag	
Failure Impact Population Rule	Impacts Feature	"Feature PK Value Rule" tag	Dependency with «Impacts Feature» stereotype
Failure Impact Primary Key Value Rule		"Failure Impact PK Value Rule" tag	
Counter Requirement Population Rule	Replaces	"RSPK Value Rule" tag	Dependency with «Replaces» Stereotype
Counter Requirement Primary Key Value Rule		"Counter Requirement PK Value Rule" tag	
Failure Mode Population Rule	Abnormal State Of	"DCPK Value Rule" tag	Dependency with «Abnormal State Of» stereotype
Failure Mode Primary Key Value Rule		"FMPK Value Rule" tag	
Author	All metaclasses and metarelationships	Not mapped	
Change Date		Not mapped	
Change Description		Documentation/Comments	Element
Class Level		Not mapped	
Definition		Definition	Element
ID		Element ID	Element
Major Version		Not mapped	
Minor Version		Not mapped	
Name		Name	Element
Organization Owner		Not mapped	
Owner		Owner	Element
Status		Status	Element
Update Version		Not mapped	

