Consolidation of Model Materials from: INCOSE Patterns WG Interface Project (2017-18) and ST4SE Interface Pattern Activities (2018-19)

January, 2020

V2.1.2

Purpose of consolidated materials

- Model content is Interface Pattern as discussed and consensus agreed by ST4SE team during Oct, 2018, Boston, face-to-face meeting (see slides 24-29).
- Combine several artifacts across the two INCOSE project teams: Patterns WG Interface Pattern Project (2017-2018) and ST4SE Project (2018-2019)
- Consolidates Jan, 2020, status of related Interface Pattern, represented in SysML, Protégé, and OWL DL (slides 42-49).
- Includes the original definitions of classes, relationships (slides 25-29).
- Prior to extraction and placement in public access media.
- Several historical sections are included, delineated, for perspective.

Discussion of Ontology Around “Interface”

ST4SE Core Team Meeting
Jan 29, 2018

Bill Schindel, ICTT System Sciences

schindel@ictt.com

V1.2.5

Contents

- Purpose of this material
- Brief background on S*Models, S*Patterns, INCOSE Patterns Working Group
- Focus of this discussion: Interface portion of S*Metamodel
- Translation to OWL, editing via Protégé, current status, related questions
- Discussion and plans

- References

Purpose of this material

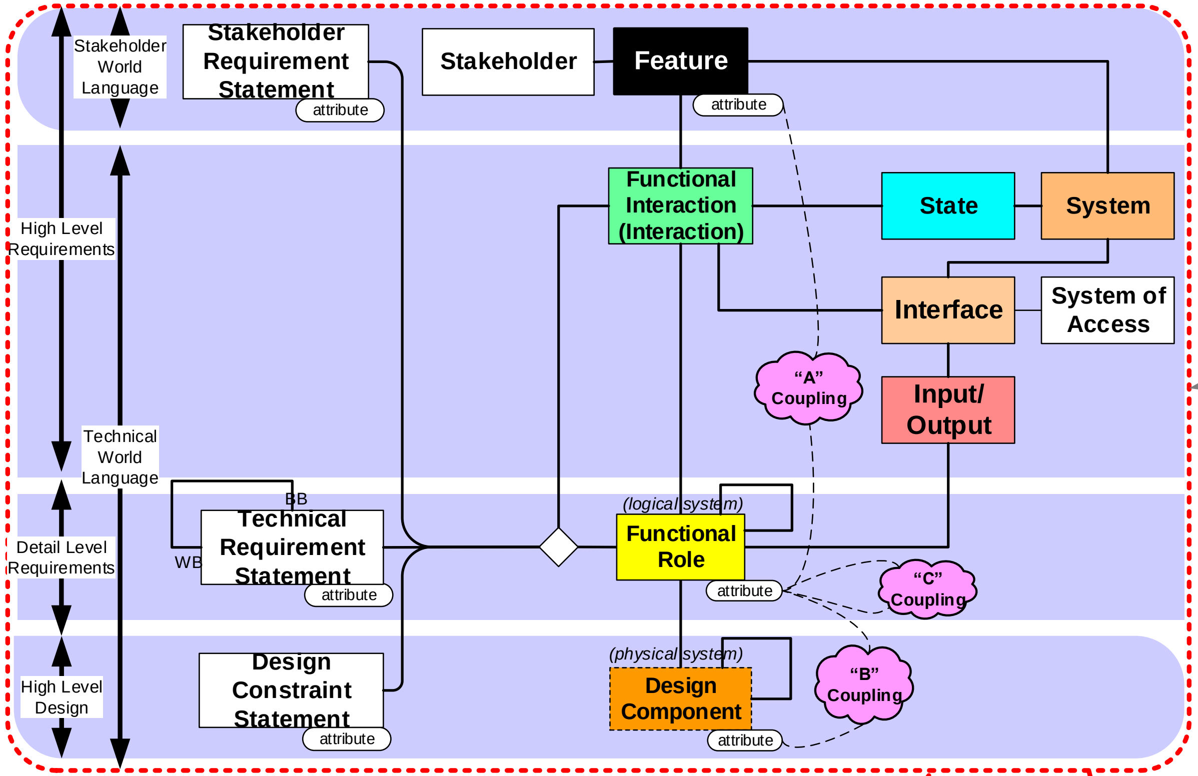
- Discuss SE ontology related to Interface
- Observe how this has been expressed in the past in the S*Metamodel
- Observe how the S*Metamodel is being expressed in OWL
- Solicit input on OWL / Protégé editing to continue this process

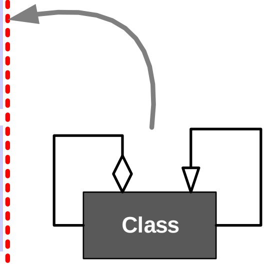
Brief background on S*Models, S*Patterns, and INCOSE Patterns Working Group

We are concerned with *configurable, re-usable system models*: “S*Patterns”

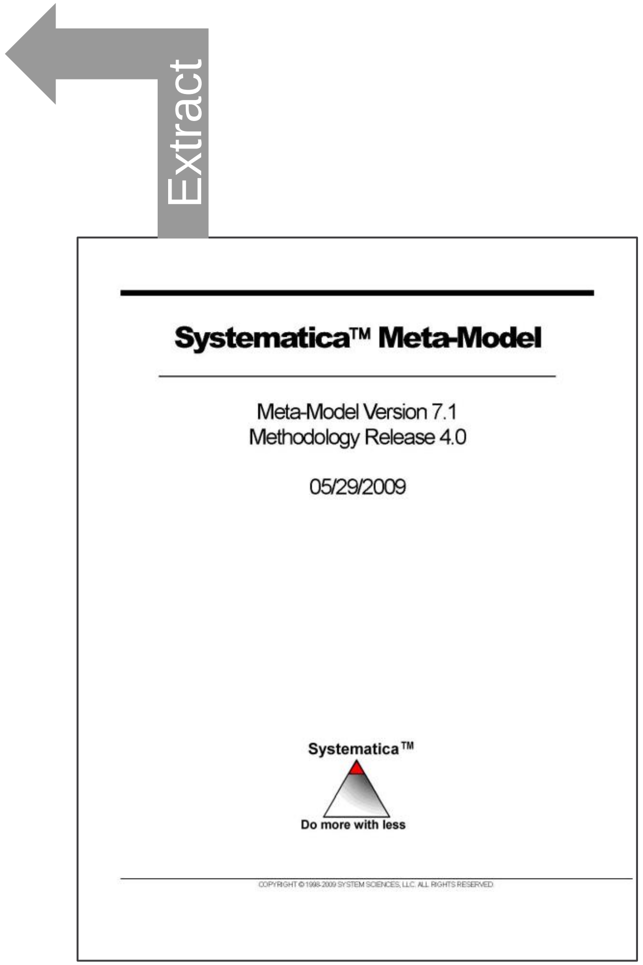
1. Models containing a certain minimal set of elements are called S*Models (S* is short for “Systematica”)
2. Those underlying elements are called the S*Metamodel, which was inspired by the physical sciences, seeking the smallest model necessary for life cycle engineering
3. S*Models using those elements may be (have been) expressed in any modeling language (e.g., SysML, or other languages)
4. S*Models can be (have been) created and managed in many different COTS modeling tools, engineering tools, requirements management and PLM systems.
5. Re-usable, configurable S*Models are called S*Patterns
6. By “Pattern-Based Systems Engineering” (PBSE) we mean MBSE enhanced by these generalized assets and utilizing the leverage of S*Patterns.
7. These are system-level patterns (models of whole managed platforms), not just smaller-scale component design patterns

S*Metamodel for Model-Based Systems Engineering (MBSE)





Every S*Metaclass shown is embedded in both a containment hierarchy and an abstraction (class) hierarchy.



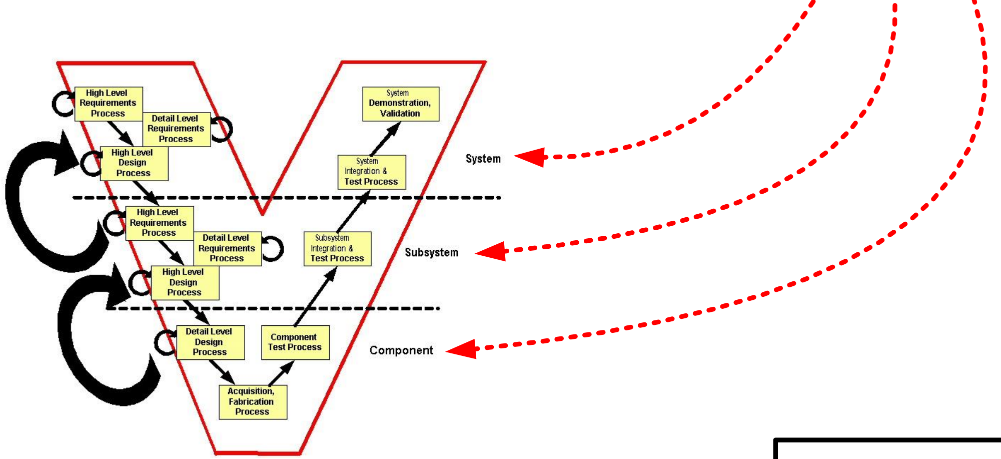
Systematica™ Meta-Model

Meta-Model Version 7.1
Methodology Release 4.0

05/29/2009

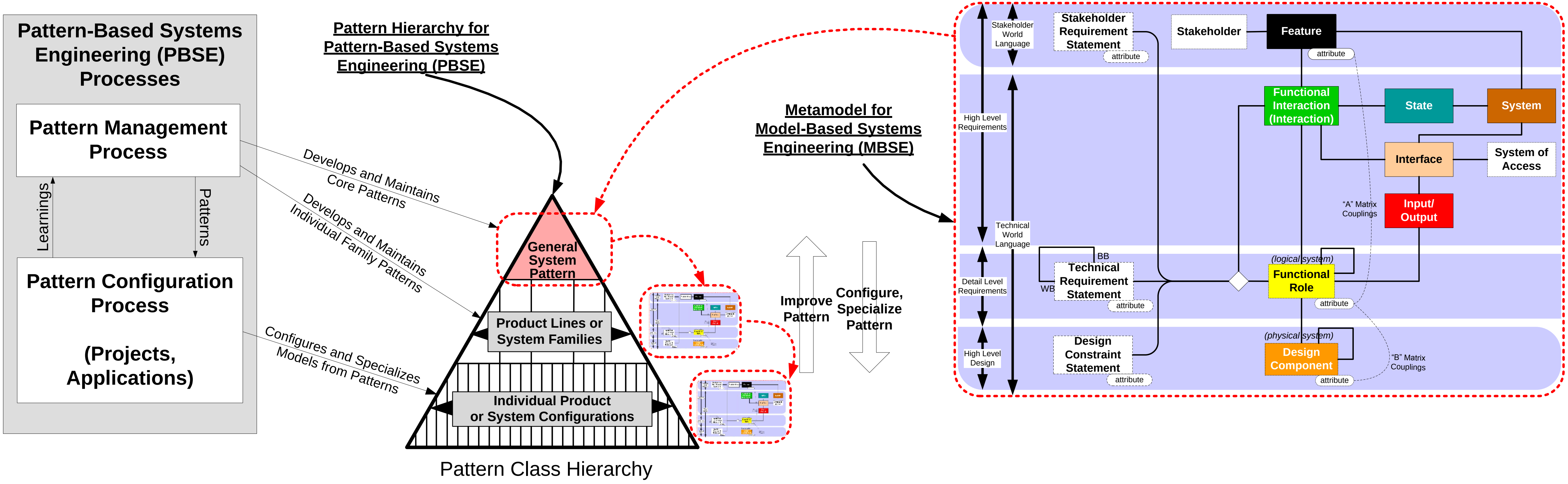
Systematica™
Do more with less

COPYRIGHT © 1998-2009 SYSTEM SCIENCES, LLC. ALL RIGHTS RESERVED.


High Level Requirements Process
Detail Level Requirements Process
High Level Design Process
System Demonstration, Validation
System Integration & Test Process
System
High Level Requirements Process
Detail Level Requirements Process
Subsystem Integration & Test Process
High Level Design Process
Subsystem
Detail Level Design Process
Component Test Process
Component
Acquisition, Fabrication Process

System Containment Hierarchy

Pattern-Based Systems Engineering:
Using Configurable S*Patterns to Create Configured S*Models



Pattern-Based Systems Engineering (PBSE) Processes

Pattern Management Process

Learnings

Patterns

Pattern Configuration Process

(Projects, Applications)

Develops and Maintains Core Patterns

Develops and Maintains Individual Family Patterns

Configures and Specializes Models from Patterns

Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)

General System Pattern

Product Lines or System Families

Individual Product or System Configurations

Pattern Class Hierarchy

Improve Pattern

Configure, Specialize Pattern

Metamodel for Model-Based Systems Engineering (MBSE)

Stakeholder World Language

High Level Requirements

Technical World Language

Detail Level Requirements

High Level Design

Stakeholder Requirement Statement

attribute

Stakeholder

Feature

attribute

Functional Interaction (Interaction)

State

System

Interface

System of Access

"A" Matrix Couplings

Input/ Output

BB

WB

Technical Requirement Statement

attribute

(logical system)

Functional Role

attribute

(physical system)

Design Constraint Statement

attribute

Design Component

attribute

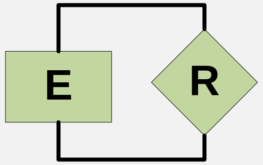
"B" Matrix Couplings

Emergence of Patterns from Patterns: S*Pattern Class Hierarchy

More General

Definition of **Relational Modeling Paradigm**

Entity-Relationship Paradigm

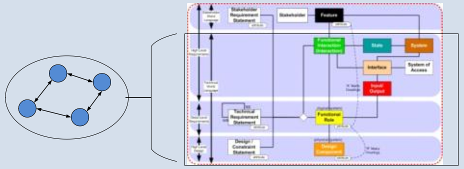


E=Entity
R= Relationship

Structured or unstructured semantic web

Minimal System S*Metamodel: Definition of (Elementary) System, Material Cause

S*Metamodel Core



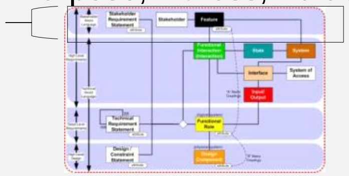
Core S*Metamodel

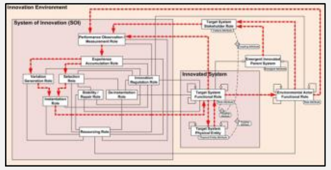
Smallest model of a system, for engineering or science

Emergence & Definition of **System of Innovation**, Fitness, Value, Purpose, Stakeholders, Agility, Final Cause, Formal Cause, Efficient Cause, Intelligence, Management, Science, Living Systems

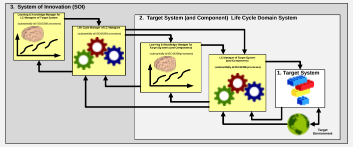
EI Pattern, SOI Pattern, Fitness, Value

S*Purpose, Fitness, Value

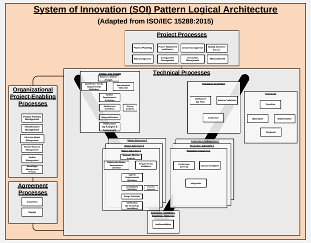




System of Innovation Pattern



Agile Sys Life Cycle Pattern


System of Innovation (SOI) Pattern Logical Architecture
(Adapted from ISO/IEC 15288:2015)
Project Processes
Technical Processes
Organizational Project-Enabling Processes
Agreement Processes

ISO 15288 System Life Cycle Mgmt Pattern

Emergence & Definition of **Domain Specific Systems**

Domain Specific Pattern

- **Terrestrial Vehicle Domain Pattern**
- **Manufacturing System Pattern**
- **Product Service System Pattern**
- **Space Tourism Pattern**
- **Electrical Power Domain Pattern**
- **Aircraft Flight Control Pattern**
- **Medical Device Pattern**
- **Distribution System Pattern**
- **Socio-Technical Pattern**
- **Critical Infrastructure Domain Pattern**

More Specific

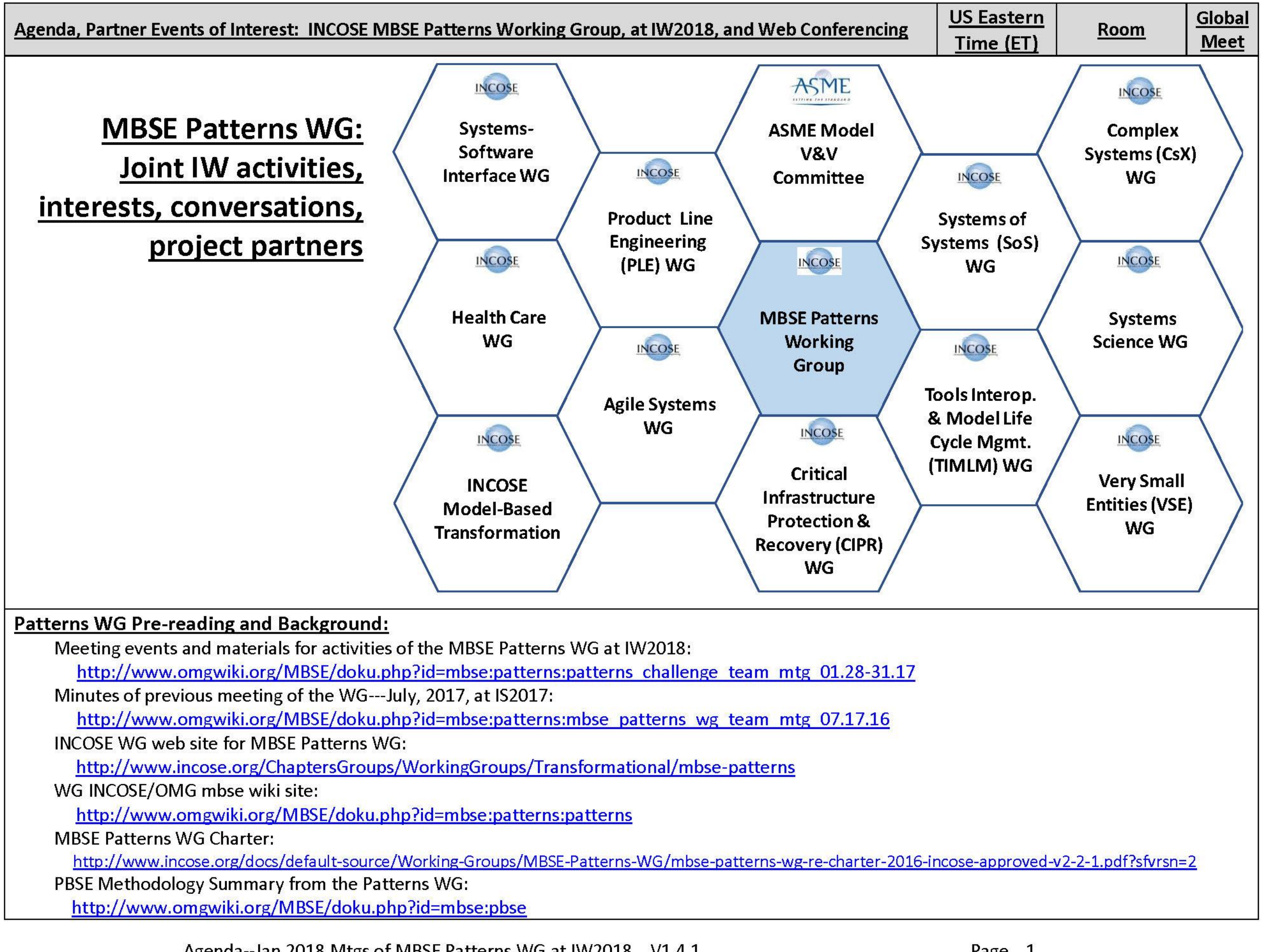
Example S*Pattern Content

- INCOSE PBSE Tutorial:
  - http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:patterns:pbse_tutorial_glrc_2016_v1.7.4.pdf
- More examples and materials on WG web wiki site:
  - http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns

The INCOSE Patterns Working Group began four years ago, as the MBSE Initiative Patterns Challenge Team:

- Part of the joint INCOSE/OMG MBSE Initiative, formed there initially as the MBSE Patterns Challenge Team.
- In 2016, our team formally became the INCOSE MBSE Patterns Working Group
- Because of our MBSE focus, and in order to continue to support the MBSE Initiative, we continue to also be listed as part of that INCOSE/MBSE Initiative
- WG web wiki site: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns

From Patterns WG Agenda at INCOSE IW2018:



Agenda, Partner Events of Interest: INCOSE MBSE Patterns Working Group, at IW2018, and Web Conferencing	US Eastern Time (ET)	Room	Global Meet

MBSE Patterns WG: Joint IW activities, interests, conversations, project partners

INCOSE Systems-Software Interface WG

INCOSE Product Line Engineering (PLE) WG

ASME ASME Model V&V Committee

INCOSE Systems of Systems (SoS) WG

INCOSE Complex Systems (CsX) WG

INCOSE Health Care WG

INCOSE MBSE Patterns Working Group

INCOSE Systems Science WG

INCOSE Agile Systems WG

INCOSE Tools Interop. & Model Life Cycle Mgmt. (TIMLM) WG

INCOSE INCOSE Model-Based Transformation

INCOSE Critical Infrastructure Protection & Recovery (CIPR) WG

INCOSE Very Small Entities (VSE) WG

Patterns WG Pre-reading and Background:

Meeting events and materials for activities of the MBSE Patterns WG at IW2018:
http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns_challenge_team_mtg_01.28-31.17

Minutes of previous meeting of the WG---July, 2017, at IS2017:
http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:mbse_patterns_wg_team_mtg_07.17.16

INCOSE WG web site for MBSE Patterns WG:
http://www.incose.org/ChaptersGroups/WorkingGroups/Transformational/mbse-patterns

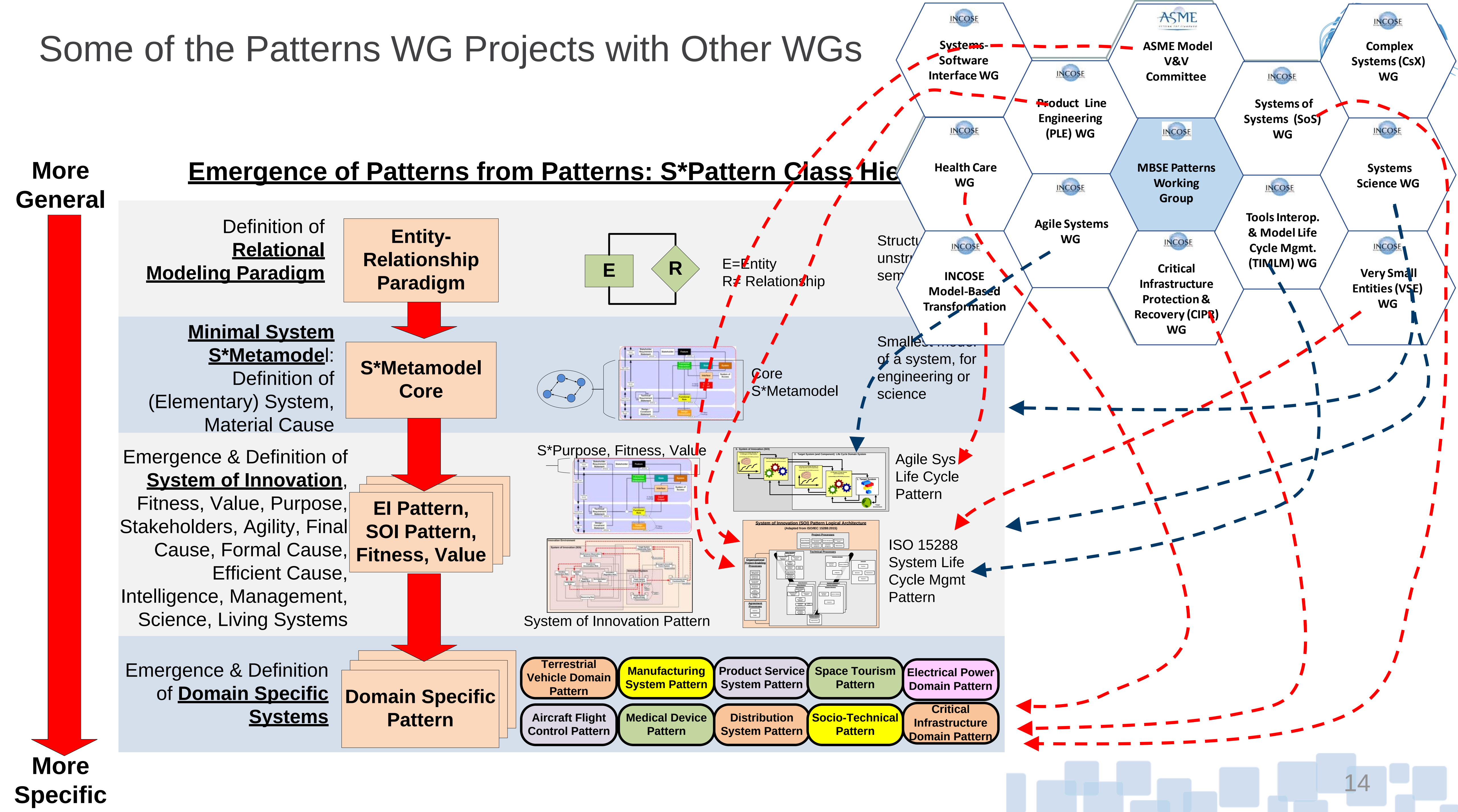
WG INCOSE/OMG mbse wiki site:
http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns

MBSE Patterns WG Charter:
http://www.incose.org/docs/default-source/Working-Groups/MBSE-Patterns-WG/mbse-patterns-wg-re-charter-2016-incose-approved-v2-2-1.pdf?sfvrsn=2

PBSE Methodology Summary from the Patterns WG:
http://www.omgwiki.org/MBSE/doku.php?id=mbse:pbse

Agenda--Jan 2018 Mtgs of MBSE Patterns WG at IW2018 -- V1.4.1 Page 1

Some of the Patterns WG Projects with Other WGs



More General

More Specific

Emergence of Patterns from Patterns: S*Pattern Class Hie

Definition of **Relational Modeling Paradigm**

Entity-Relationship Paradigm

E=Entity
R= Relationship

Structu unstr sem

Minimal System S*Metamodel: Definition of (Elementary) System, Material Cause

S*Metamodel Core

Core S*Metamodel

Smallest model of a system, for engineering or science

Emergence & Definition of **System of Innovation**, Fitness, Value, Purpose, Stakeholders, Agility, Final Cause, Formal Cause, Efficient Cause, Intelligence, Management, Science, Living Systems

EI Pattern, SOI Pattern, Fitness, Value

S*Purpose, Fitness, Value

System of Innovation Pattern

Agile Sys Life Cycle Pattern

ISO 15288 System Life Cycle Mgmt Pattern

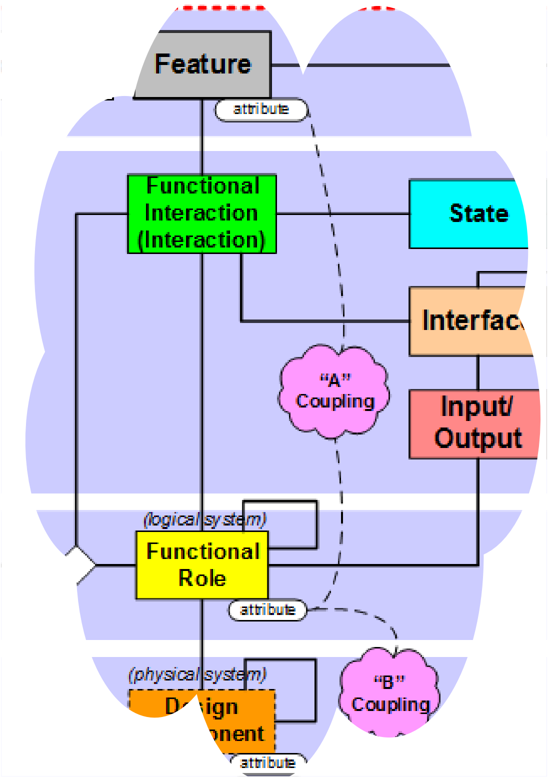
Emergence & Definition of **Domain Specific Systems**

Domain Specific Pattern

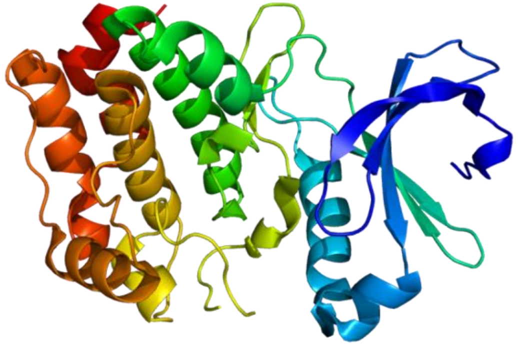
Terrestrial Vehicle Domain Pattern	Manufacturing System Pattern	Product Service System Pattern	Space Tourism Pattern	Electrical Power Domain Pattern
Aircraft Flight Control Pattern	Medical Device Pattern	Distribution System Pattern	Socio-Technical Pattern	Critical Infrastructure Domain Pattern

E
R
INCOSE
ASME
Systems-Software Interface WG
ASME Model V&V Committee
Complex Systems (CsX) WG
Product Line Engineering (PLE) WG
Systems of Systems (SoS) WG
Health Care WG
MBSE Patterns Working Group
Systems Science WG
Agile Systems WG
Tools Interop. & Model Life Cycle Mgmt. (TIMLM) WG
INCOSE Model-Based Transformation
Critical Infrastructure Protection & Recovery (CIPR) WG
Very Small Entities (VSE) WG

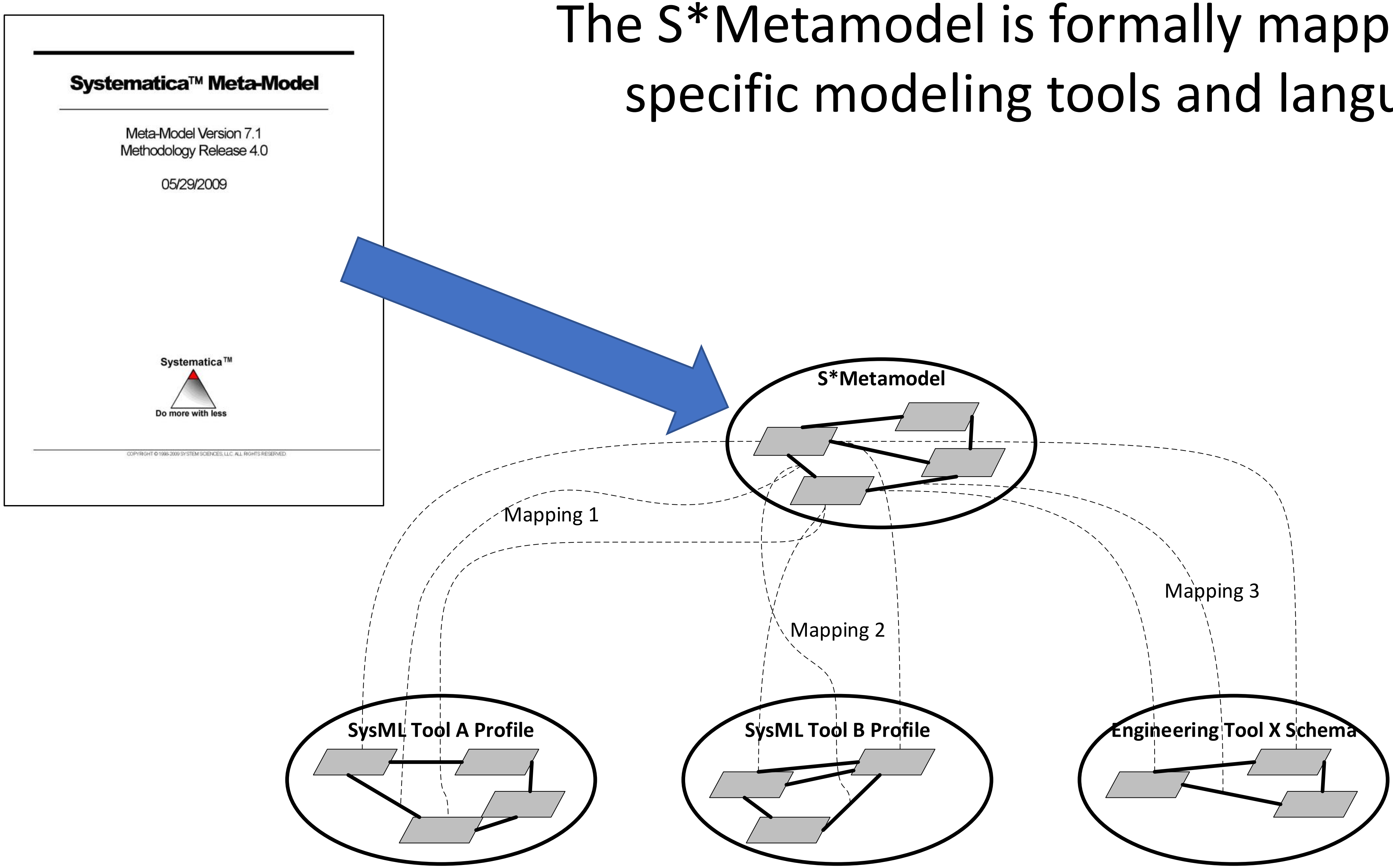
Status of WG Projects



- Interface Patterns Project
- ASME Model V&V Patterns Project, VVUQ Pattern
- VSE Patterns Project
- Agile Patterns Project
- S3 Pattern and INCOSE OCM
- Patterns in the Public Square
- TIMLM Patterns
- HC WG Collaboration
- PLE WG Project
- CIPR Patterns
- IFSR Conversation
- SysSciWG Patterns
- SoS WG Collaboration



The S*Metamodel is formally mapped to specific modeling tools and languages



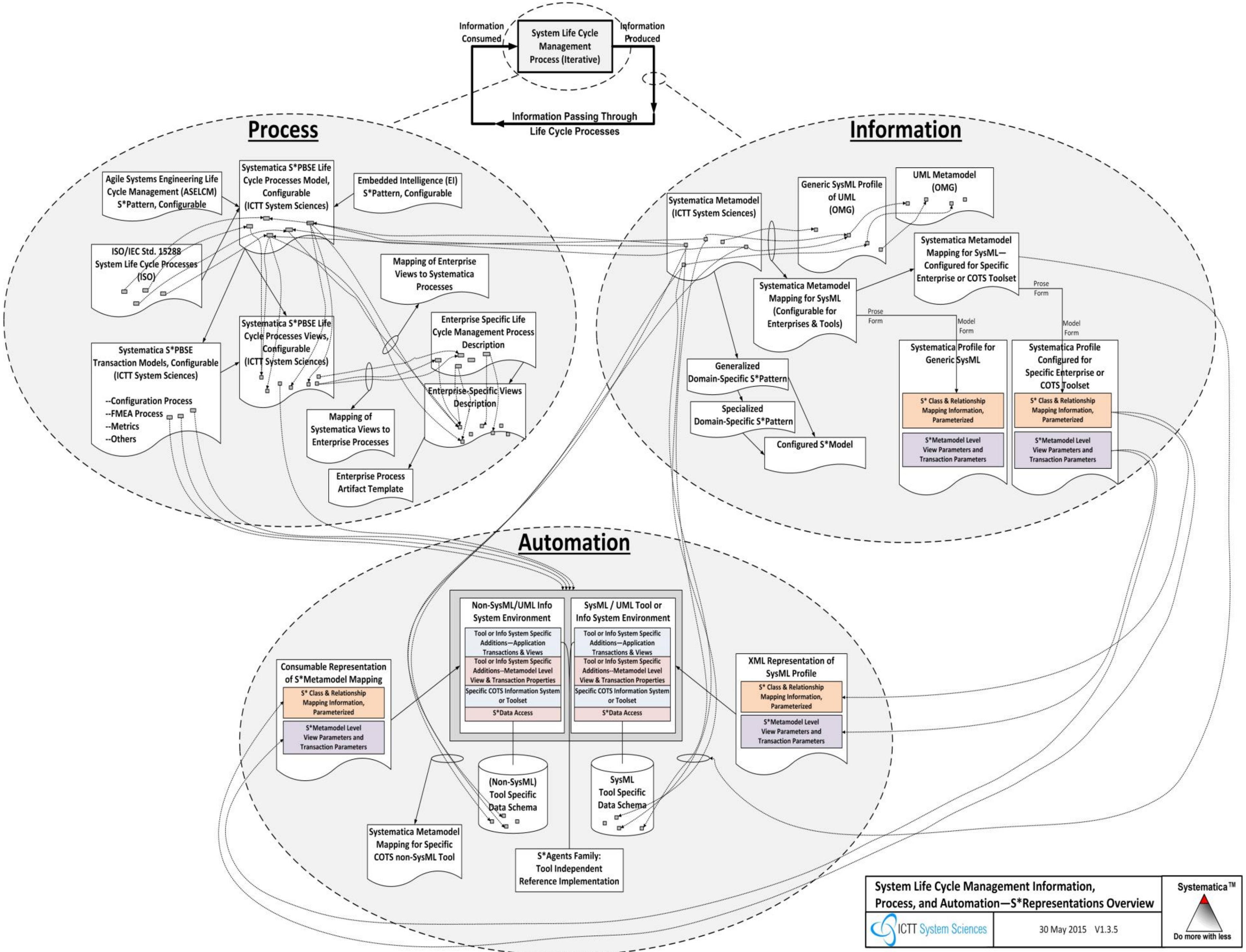
Systematica™ Meta-Model

Meta-Model Version 7.1
Methodology Release 4.0

05/29/2009

Systematica™
Do more with less

COPYRIGHT © 1998-2009 SYSTEM SCIENCES, LLC. ALL RIGHTS RESERVED.



Information Consumed

System Life Cycle Management Process (Iterative)

Information Produced

Information Passing Through Life Cycle Processes

Process

Agile Systems Engineering Life Cycle Management (ASELCM) S*Pattern, Configurable

Systematica S*PBSE Life Cycle Processes Model, Configurable (ICTT System Sciences)

Embedded Intelligence (EI) S*Pattern, Configurable

ISO/IEC Std. 15288 System Life Cycle Processes (ISO)

Mapping of Enterprise Views to Systematica Processes

Enterprise Specific Life Cycle Management Process Description

Systematica S*PBSE Life Cycle Processes Views, Configurable (ICTT System Sciences)

Systematica S*PBSE Transaction Models, Configurable (ICTT System Sciences)

--Configuration Process
--FMEA Process
--Metrics
--Others

Enterprise-Specific Views Description

Mapping of Systematica Views to Enterprise Processes

Enterprise Process Artifact Template

Information

Systematica Metamodel (ICTT System Sciences)

Generic SysML Profile of UML (OMG)

UML Metamodel (OMG)

Systematica Metamodel Mapping for SysML—Configured for Specific Enterprise or COTS Toolset

Systematica Metamodel Mapping for SysML (Configurable for Enterprises & Tools)

Prose Form

Model Form

Prose Form

Model Form

Generalized Domain-Specific S*Pattern

Specialized Domain-Specific S*Pattern

Configured S*Model

Systematica Profile for Generic SysML

S* Class & Relationship Mapping Information, Parameterized

S*Metamodel Level View Parameters and Transaction Parameters

Systematica Profile Configured for Specific Enterprise or COTS Toolset

S* Class & Relationship Mapping Information, Parameterized

S*Metamodel Level View Parameters and Transaction Parameters

Automation

Non-SysML/UML Info System Environment

Tool or Info System Specific Additions—Application Transactions & Views

Tool or Info System Specific Additions--Metamodel Level View & Transaction Properties

Specific COTS Information System or Toolset

S*Data Access

SysML / UML Tool or Info System Environment

Tool or Info System Specific Additions—Application Transactions & Views

Tool or Info System Specific Additions--Metamodel Level View & Transaction Properties

Specific COTS Information System or Toolset

S*Data Access

Consumable Representation of S*Metamodel Mapping

S* Class & Relationship Mapping Information, Parameterized

S*Metamodel Level View Parameters and Transaction Parameters

XML Representation of SysML Profile

S* Class & Relationship Mapping Information, Parameterized

S*Metamodel Level View Parameters and Transaction Parameters

(Non-SysML) Tool Specific Data Schema

SysML Tool Specific Data Schema

Systematica Metamodel Mapping for Specific COTS non-SysML Tool

S*Agents Family: Tool Independent Reference Implementation

System Life Cycle Management Information, Process, and Automation—S*Representations Overview

Systematica™

Do more with less

ICTT System Sciences

30 May 2015 V1.3.5

Discussion of S*Interface System of Access (SOA) Semantics

Interface Patterns Project Meeting

06.30.2017

B. Schindel (with help from J. Sherey)

Informal semantics of S*Interface

The Setting: Consider two interacting systems, exchanging at least one Input-Output (e.g., a Force, Energy Flow, Mass Flow, or Information), during Interaction D:

Interaction D

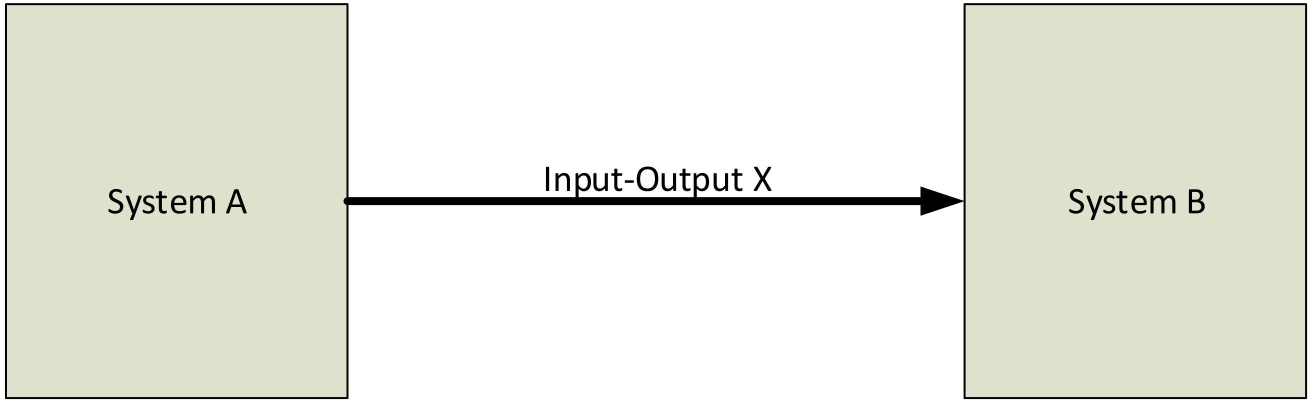


Figure 1: (Exact notation used not important to this discussion)

- In certain (important to identify) circumstances, we need to represent Interfaces involved in Interaction D.
- No matter what (graphical or other) modeling language or notation is used, the S*Metamodel tells us that an Interface is an association of:
  - A System, which “has” the Interface;
  - A (set of) Input-Output(s), which “pass through” the Interface;
  - A (set of) Interaction(s), which describe “behavior at the Interface”;
  - A System of Access (SOA), providing the interaction “medium”:

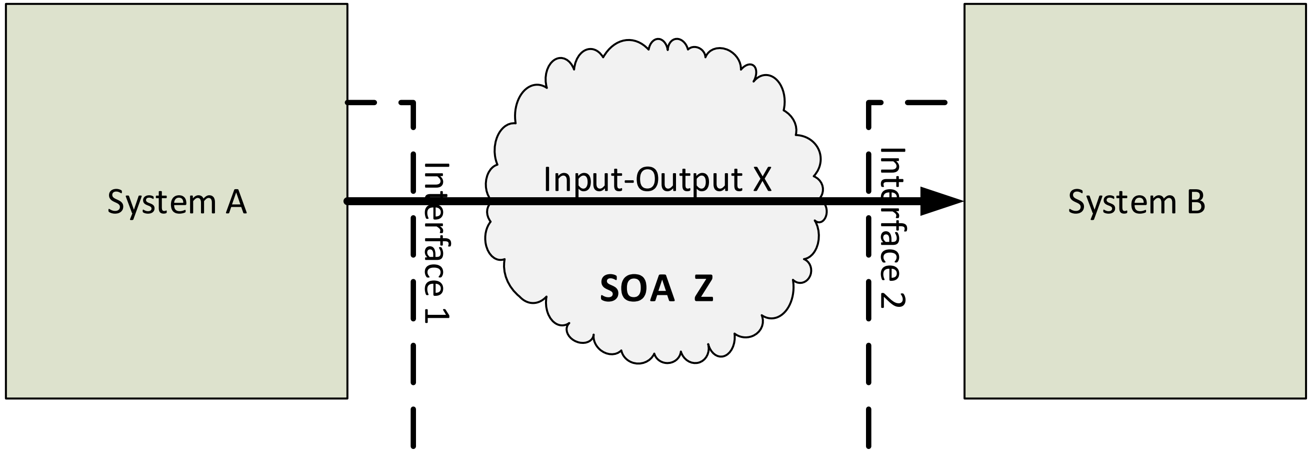


Figure 2: (Exact notation used not important to this discussion)

- However, there is a subtle inconsistency in the transition between Figure 1 and Figure 2 above:
  - Figure 1 and Figure 2 imply that the scope of “System A” must have changed between the two diagrams, . . .
  - Because, System A in Figure 2 can interact with an external-looking SOA Z, but . . . .
  - System A in Figure 1 implies that the scope of System A is such that it can interact directly with System B.

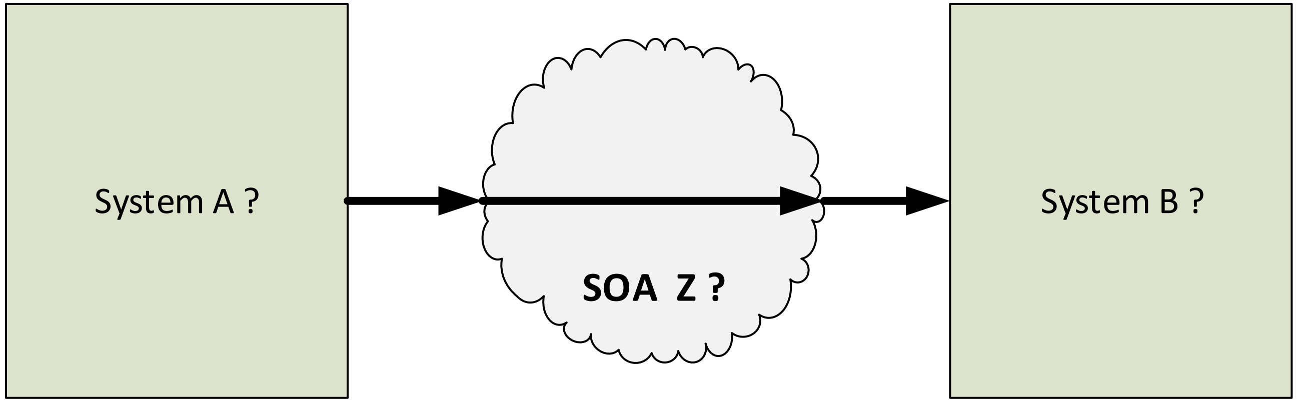


Figure 3: (Exact notation used not important to this discussion)

- The problem here is that even intended “neutral” notations can be specific enough to mislead us, or create ambiguities.
- The real problem is that, independent of notation, the System of Access by definition has larger scope than Figure 2 implied:

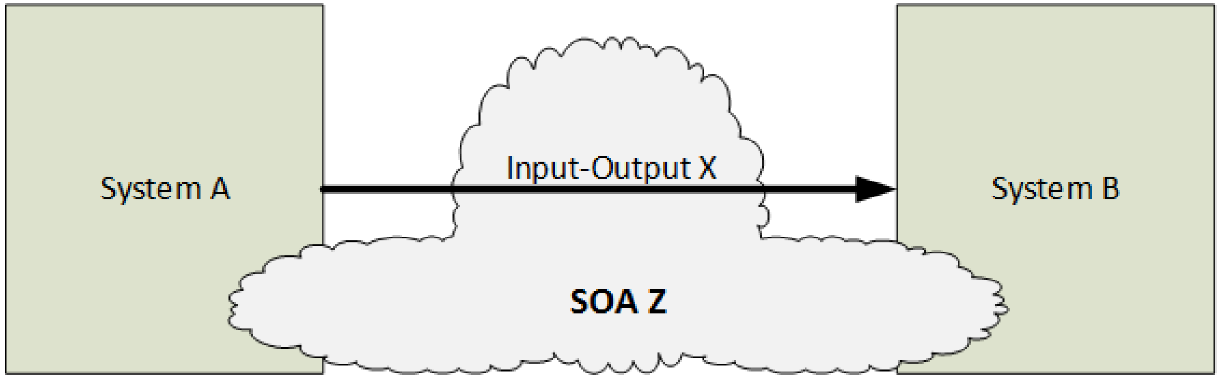


Figure 4: (Exact notation used not important to this discussion)

- Part of the scope of the System of Access for two interacting systems must necessarily be within the two interacting systems . . .

- So, to avoid conflicting or ambiguous definitions of the scope of System A, we have to recognize a slightly larger system, shown in Figure 5 as System A’
- The additional scope adds the SOA role shown here as SASOA:

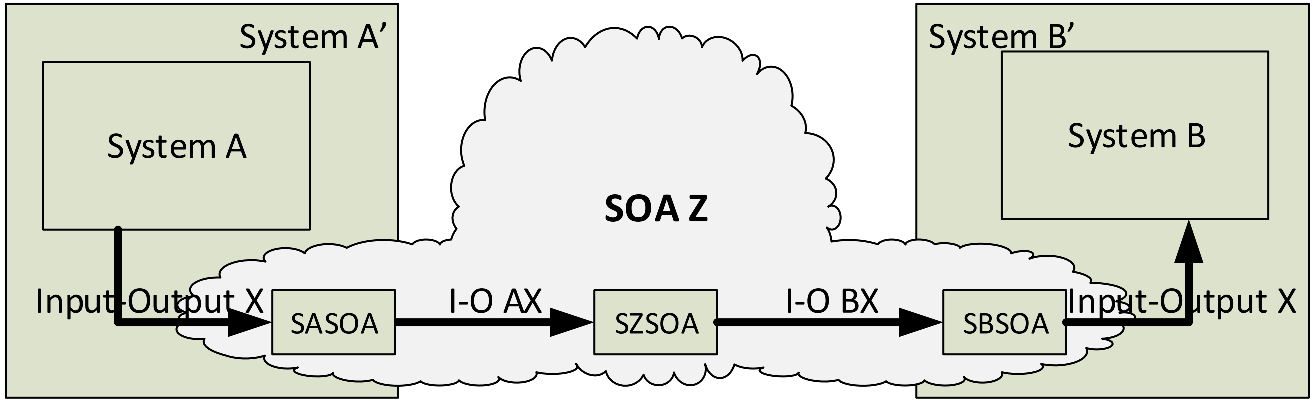
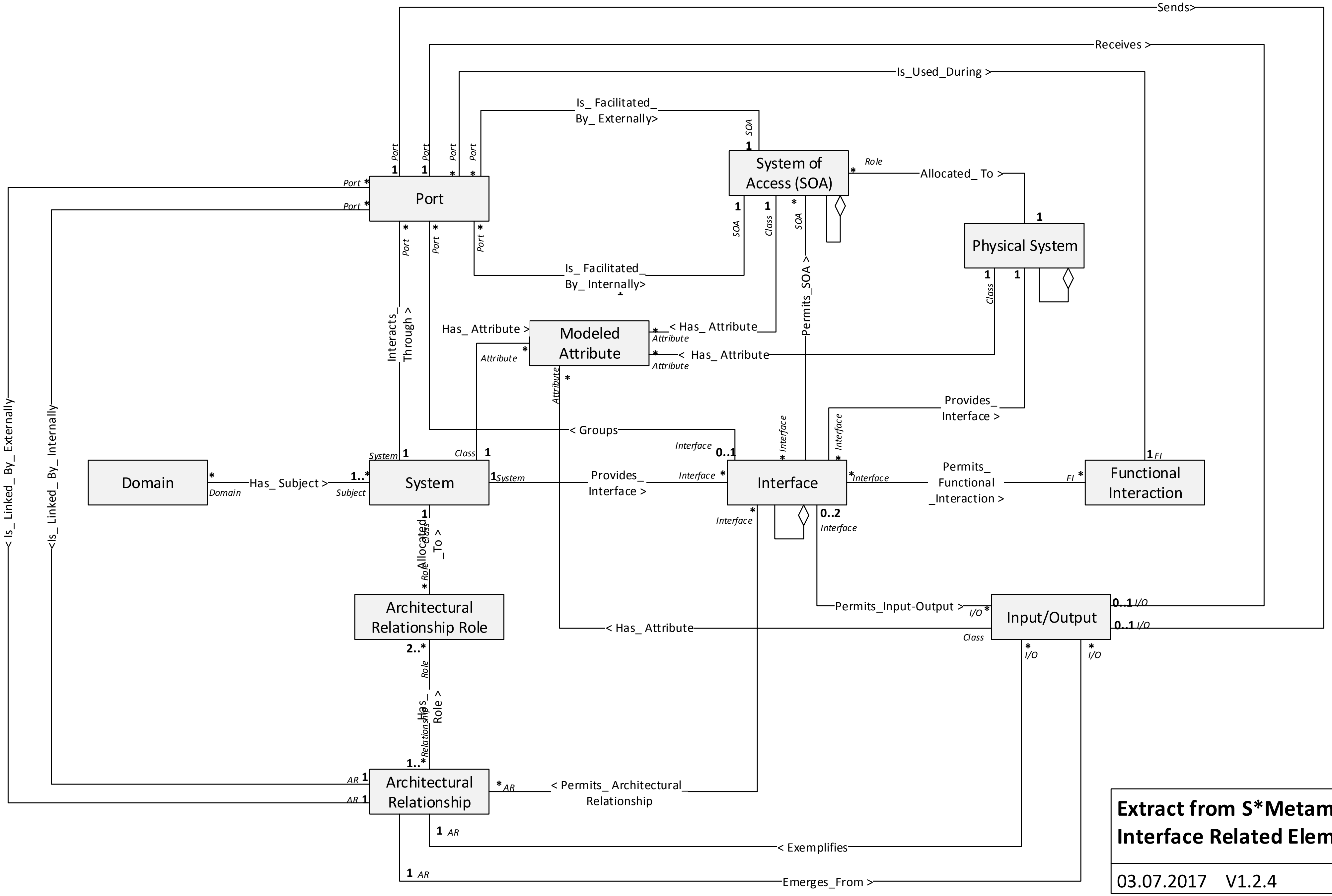


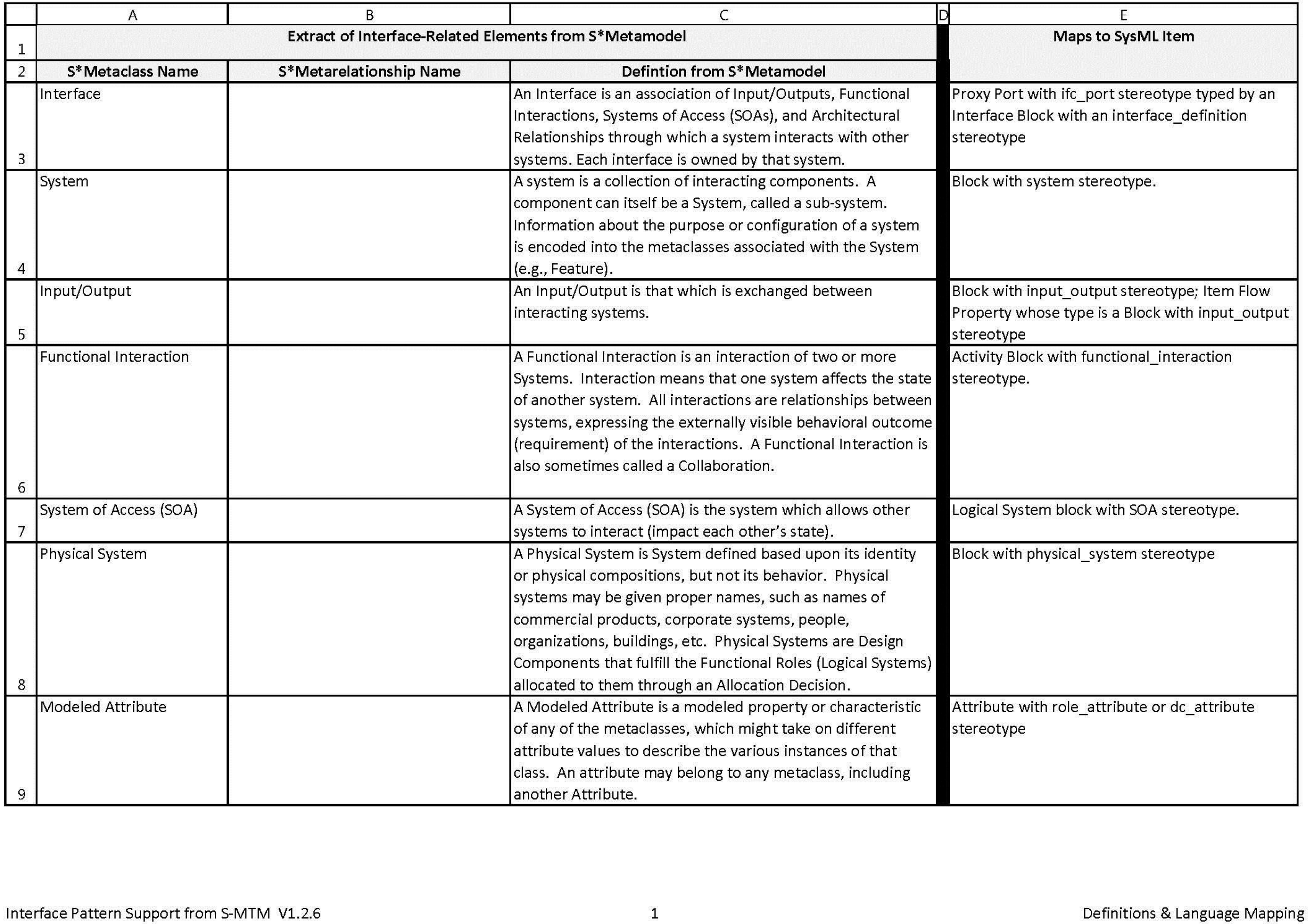
Figure 5: (Exact notation used not important to this discussion)

Interface portion of S*Metamodel



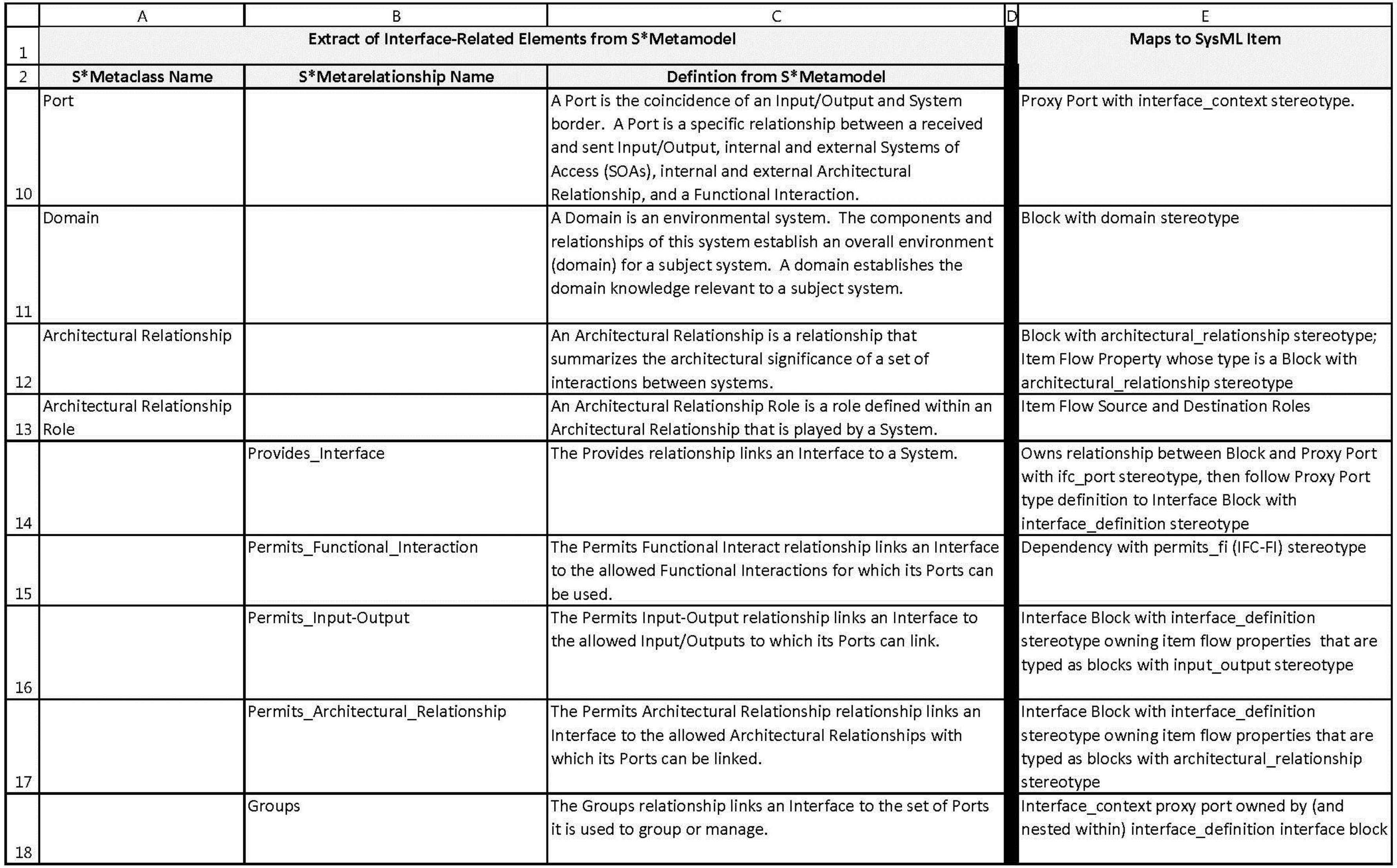
Extract from S*Metamodel: Interface Related Elements

03.07.2017 V1.2.4

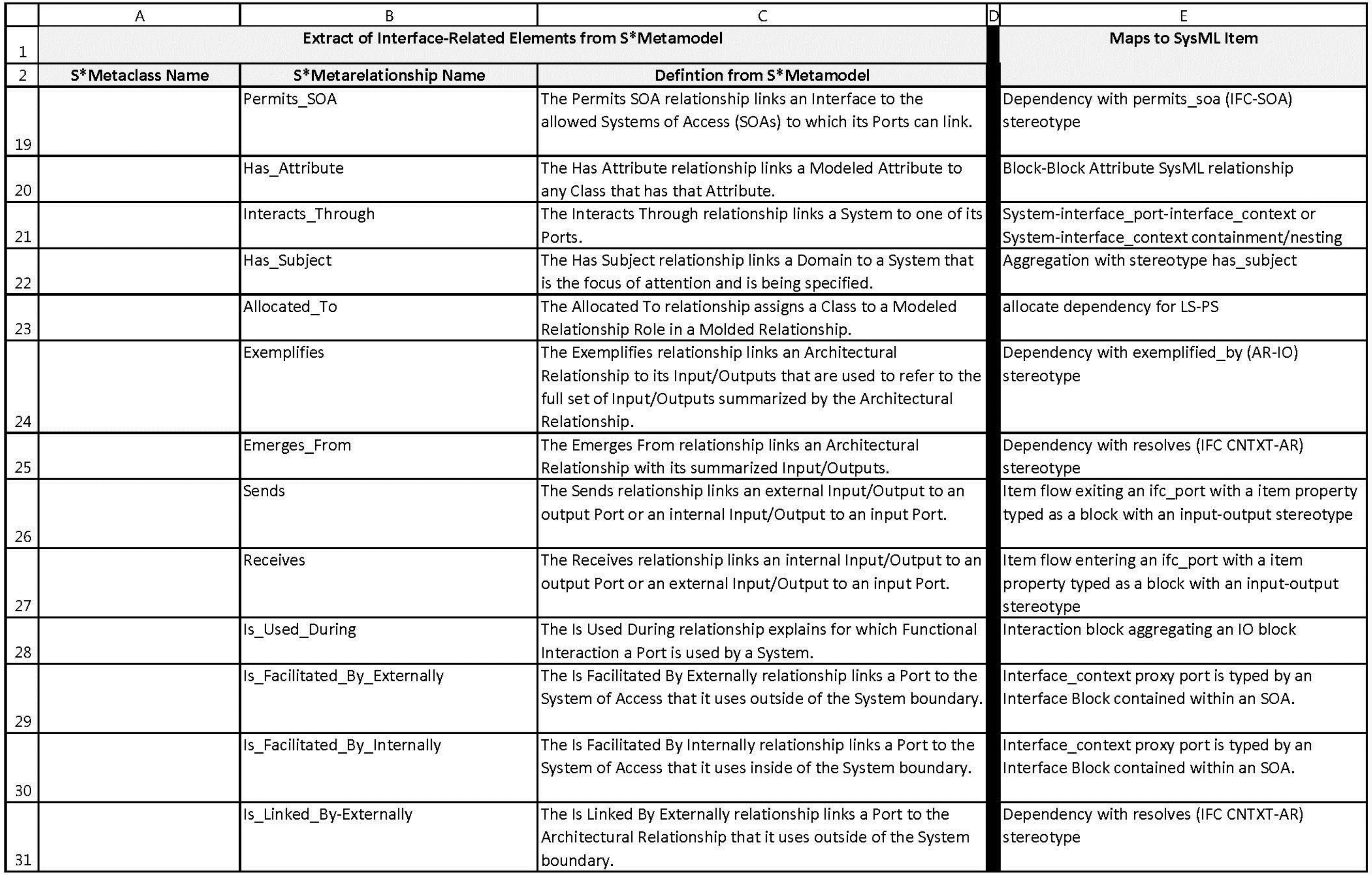


	A	B	C	D	E
1	Extract of Interface-Related Elements from S*Metamodel				Maps to SysML Item
2	S*Metaclass Name	S*Metarelationship Name	Defintion from S*Metamodel		
3	Interface		An Interface is an association of Input/Outputs, Functional Interactions, Systems of Access (SOAs), and Architectural Relationships through which a system interacts with other systems. Each interface is owned by that system.		Proxy Port with ifc_port stereotype typed by an Interface Block with an interface_definition stereotype
4	System		A system is a collection of interacting components. A component can itself be a System, called a sub-system. Information about the purpose or configuration of a system is encoded into the metaclasses associated with the System (e.g., Feature).		Block with system stereotype.
5	Input/Output		An Input/Output is that which is exchanged between interacting systems.		Block with input_output stereotype; Item Flow Property whose type is a Block with input_output stereotype
6	Functional Interaction		A Functional Interaction is an interaction of two or more Systems. Interaction means that one system affects the state of another system. All interactions are relationships between systems, expressing the externally visible behavioral outcome (requirement) of the interactions. A Functional Interaction is also sometimes called a Collaboration.		Activity Block with functional_interaction stereotype.
7	System of Access (SOA)		A System of Access (SOA) is the system which allows other systems to interact (impact each other's state).		Logical System block with SOA stereotype.
8	Physical System		A Physical System is System defined based upon its identity or physical compositions, but not its behavior. Physical systems may be given proper names, such as names of commercial products, corporate systems, people, organizations, buildings, etc. Physical Systems are Design Components that fulfill the Functional Roles (Logical Systems) allocated to them through an Allocation Decision.		Block with physical_system stereotype
9	Modeled Attribute		A Modeled Attribute is a modeled property or characteristic of any of the metaclasses, which might take on different attribute values to describe the various instances of that class. An attribute may belong to any metaclass, including another Attribute.		Attribute with role_attribute or dc_attribute stereotype

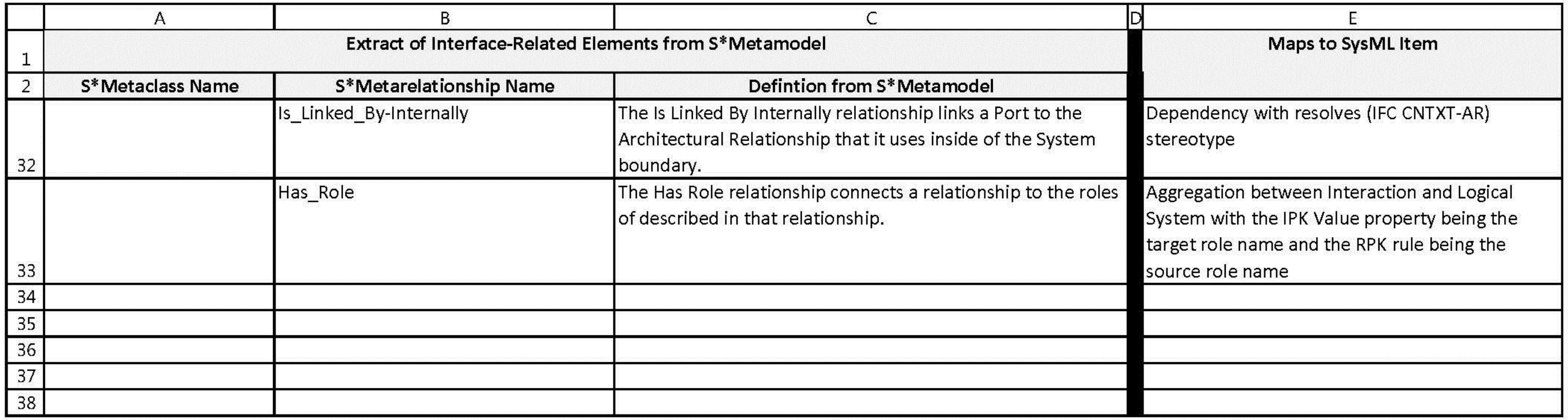
Interface Pattern Support from S-MTM V1.2.6 1 Definitions & Language Mapping



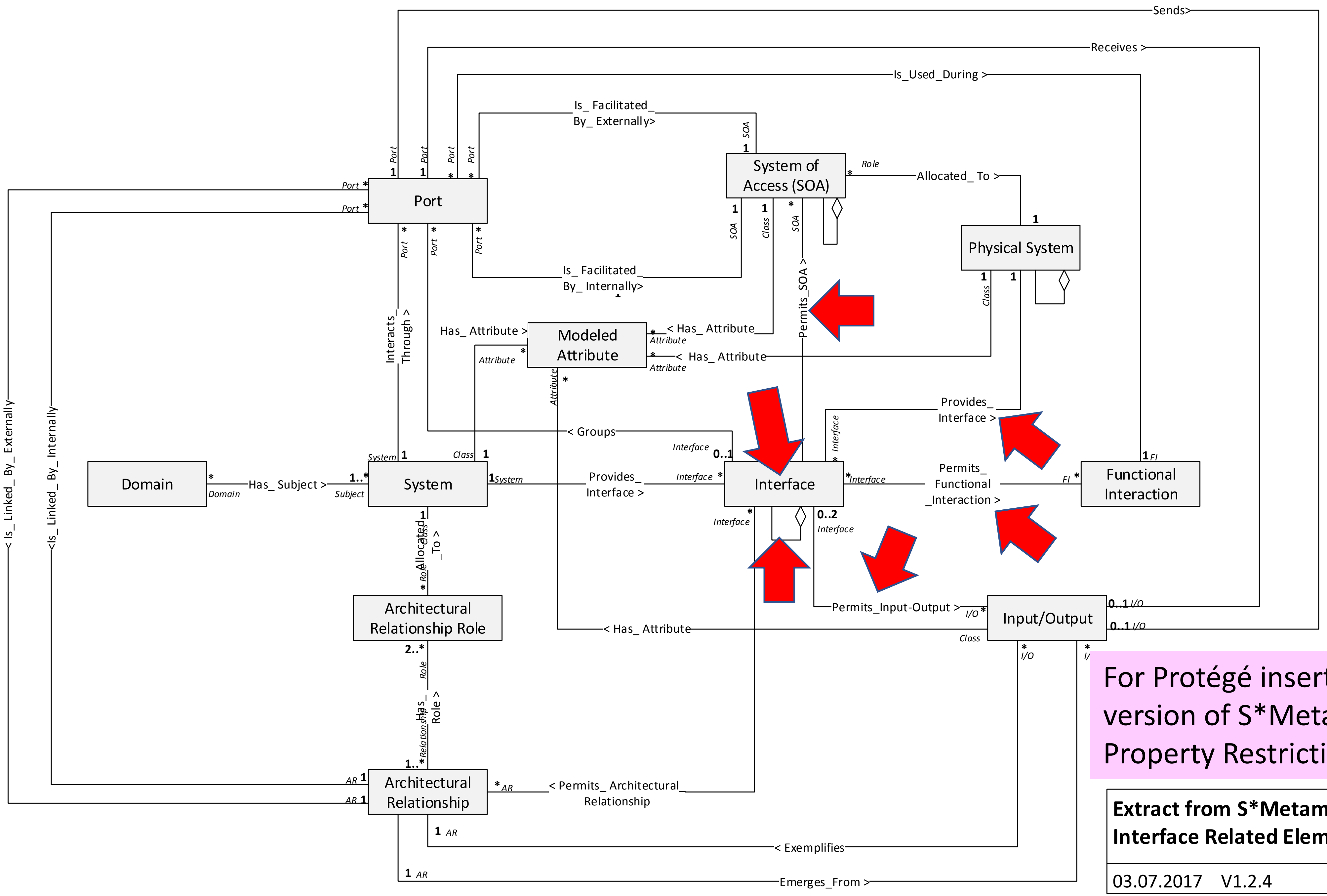
	A	B	C	D	E
1	Extract of Interface-Related Elements from S*Metamodel				Maps to SysML Item
2	S*Metaclass Name	S*Metarelationship Name	Defintion from S*Metamodel		
10	Port		A Port is the coincidence of an Input/Output and System border. A Port is a specific relationship between a received and sent Input/Output, internal and external Systems of Access (SOAs), internal and external Architectural Relationship, and a Functional Interaction.		Proxy Port with interface_context stereotype.
11	Domain		A Domain is an environmental system. The components and relationships of this system establish an overall environment (domain) for a subject system. A domain establishes the domain knowledge relevant to a subject system.		Block with domain stereotype
12	Architectural Relationship		An Architectural Relationship is a relationship that summarizes the architectural significance of a set of interactions between systems.		Block with architectural_relationship stereotype; Item Flow Property whose type is a Block with architectural_relationship stereotype
13	Architectural Relationship Role		An Architectural Relationship Role is a role defined within an Architectural Relationship that is played by a System.		Item Flow Source and Destination Roles
14		Provides_Interface	The Provides relationship links an Interface to a System.		Owns relationship between Block and Proxy Port with ifc_port stereotype, then follow Proxy Port type definition to Interface Block with interface_definition stereotype
15		Permits_Functional_Interaction	The Permits Functional Interact relationship links an Interface to the allowed Functional Interactions for which its Ports can be used.		Dependency with permits_fi (IFC-FI) stereotype
16		Permits_Input-Output	The Permits Input-Output relationship links an Interface to the allowed Input/Outputs to which its Ports can link.		Interface Block with interface_definition stereotype owning item flow properties that are typed as blocks with input_output stereotype
17		Permits_Architectural_Relationship	The Permits Architectural Relationship relationship links an Interface to the allowed Architectural Relationships with which its Ports can be linked.		Interface Block with interface_definition stereotype owning item flow properties that are typed as blocks with architectural_relationship stereotype
18		Groups	The Groups relationship links an Interface to the set of Ports it is used to group or manage.		Interface_context proxy port owned by (and nested within) interface_definition interface block



	A	B	C	D	E
1	Extract of Interface-Related Elements from S*Metamodel				Maps to SysML Item
2	S*Metaclass Name	S*Metarelationship Name	Defintion from S*Metamodel		
19		Permits_SOA	The Permits SOA relationship links an Interface to the allowed Systems of Access (SOAs) to which its Ports can link.		Dependency with permits_soa (IFC-SOA) stereotype
20		Has_Attribute	The Has Attribute relationship links a Modeled Attribute to any Class that has that Attribute.		Block-Block Attribute SysML relationship
21		Interacts_Through	The Interacts Through relationship links a System to one of its Ports.		System-interface_port-interface_context or System-interface_context containment/nesting
22		Has_Subject	The Has Subject relationship links a Domain to a System that is the focus of attention and is being specified.		Aggregation with stereotype has_subject
23		Allocated_To	The Allocated To relationship assigns a Class to a Modeled Relationship Role in a Molded Relationship.		allocate dependency for LS-PS
24		Exemplifies	The Exemplifies relationship links an Architectural Relationship to its Input/Outputs that are used to refer to the full set of Input/Outputs summarized by the Architectural Relationship.		Dependency with exemplified_by (AR-IO) stereotype
25		Emerges_From	The Emerges From relationship links an Architectural Relationship with its summarized Input/Outputs.		Dependency with resolves (IFC CNTXT-AR) stereotype
26		Sends	The Sends relationship links an external Input/Output to an output Port or an internal Input/Output to an input Port.		Item flow exiting an ifc_port with a item property typed as a block with an input-output stereotype
27		Receives	The Receives relationship links an internal Input/Output to an output Port or an external Input/Output to an input Port.		Item flow entering an ifc_port with a item property typed as a block with an input-output stereotype
28		Is_Used_During	The Is Used During relationship explains for which Functional Interaction a Port is used by a System.		Interaction block aggregating an IO block
29		Is_Facilitated_By_Externally	The Is Facilitated By Externally relationship links a Port to the System of Access that it uses outside of the System boundary.		Interface_context proxy port is typed by an Interface Block contained within an SOA.
30		Is_Facilitated_By_Internally	The Is Facilitated By Internally relationship links a Port to the System of Access that it uses inside of the System boundary.		Interface_context proxy port is typed by an Interface Block contained within an SOA.
31		Is_Linked_By-Externally	The Is Linked By Externally relationship links a Port to the Architectural Relationship that it uses outside of the System boundary.		Dependency with resolves (IFC CNTXT-AR) stereotype



	A	B	C	D	E
1	Extract of Interface-Related Elements from S*Metamodel				Maps to SysML Item
2	S*Metaclass Name	S*Metarelationship Name	Defintion from S*Metamodel		
32		Is_Linked_By-Internally	The Is Linked By Internally relationship links a Port to the Architectural Relationship that it uses inside of the System boundary.		Dependency with resolves (IFC CNTXT-AR) stereotype
33		Has_Role	The Has Role relationship connects a relationship to the roles of described in that relationship.		Aggregation between Interaction and Logical System with the IPK Value property being the target role name and the RPK rule being the source role name
34					
35					
36					
37					
38					



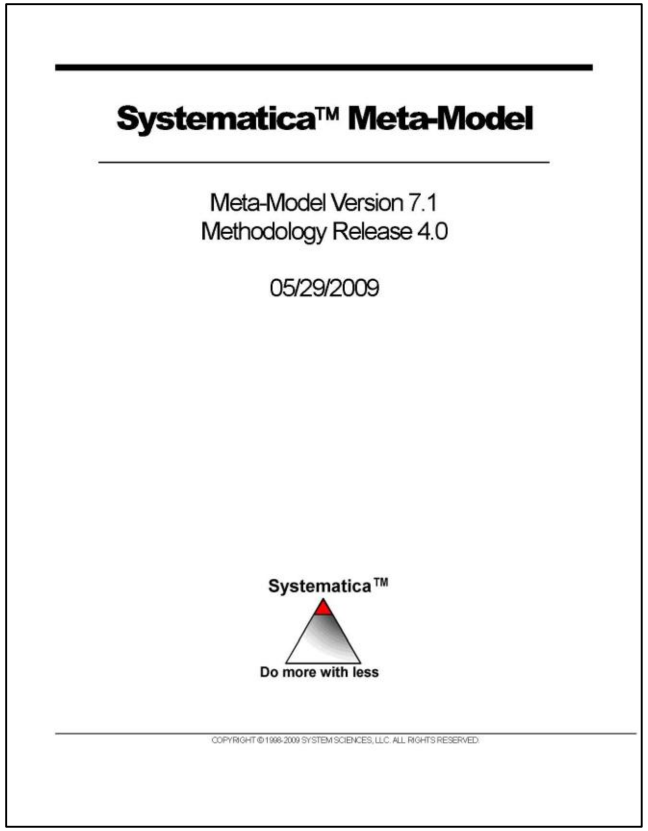
For Protégé insertion into OWL version of S*Metamodel, using Property Restrictions.

Extract from S*Metamodel: Interface Related Elements

03.07.2017 V1.2.4

Translation to OWL, editing via Protégé, current status, related questions

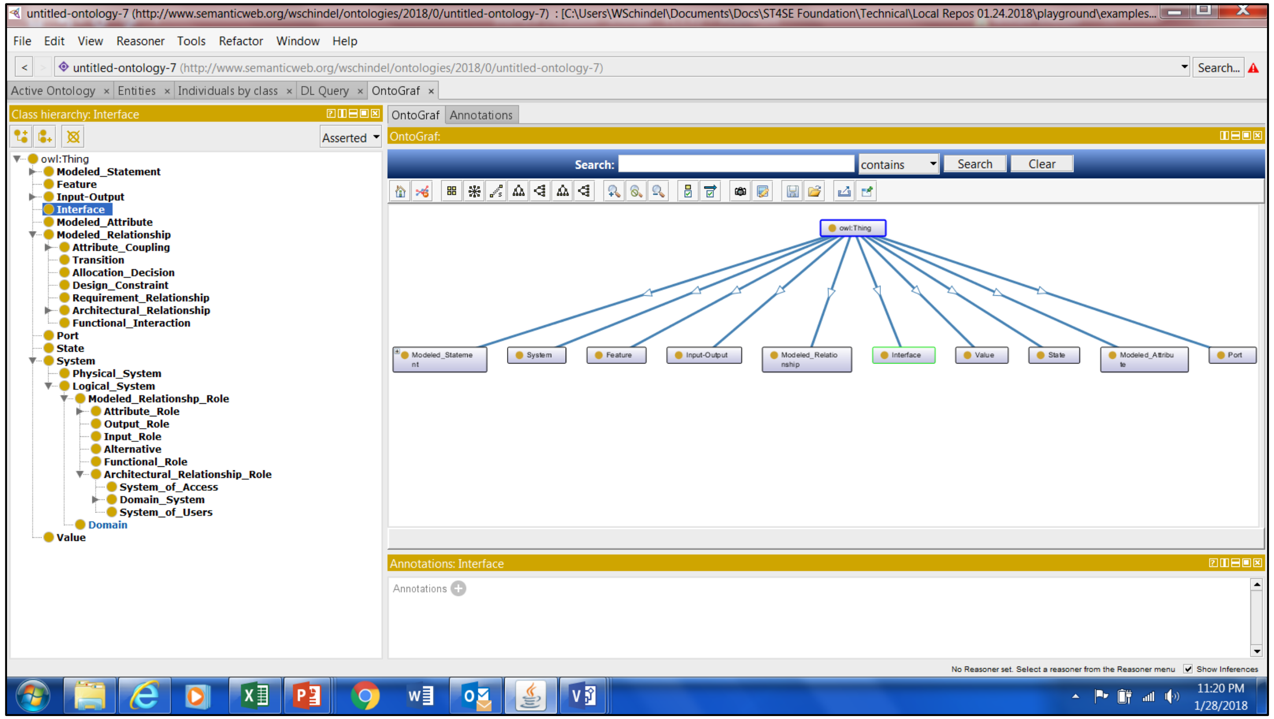
- S*Metamodel classes inserted to OWL using Protégé
- Working on Property restriction statements on Interface



Systematica™ Meta-Model

Meta-Model Version 7.1
Methodology Release 4.0

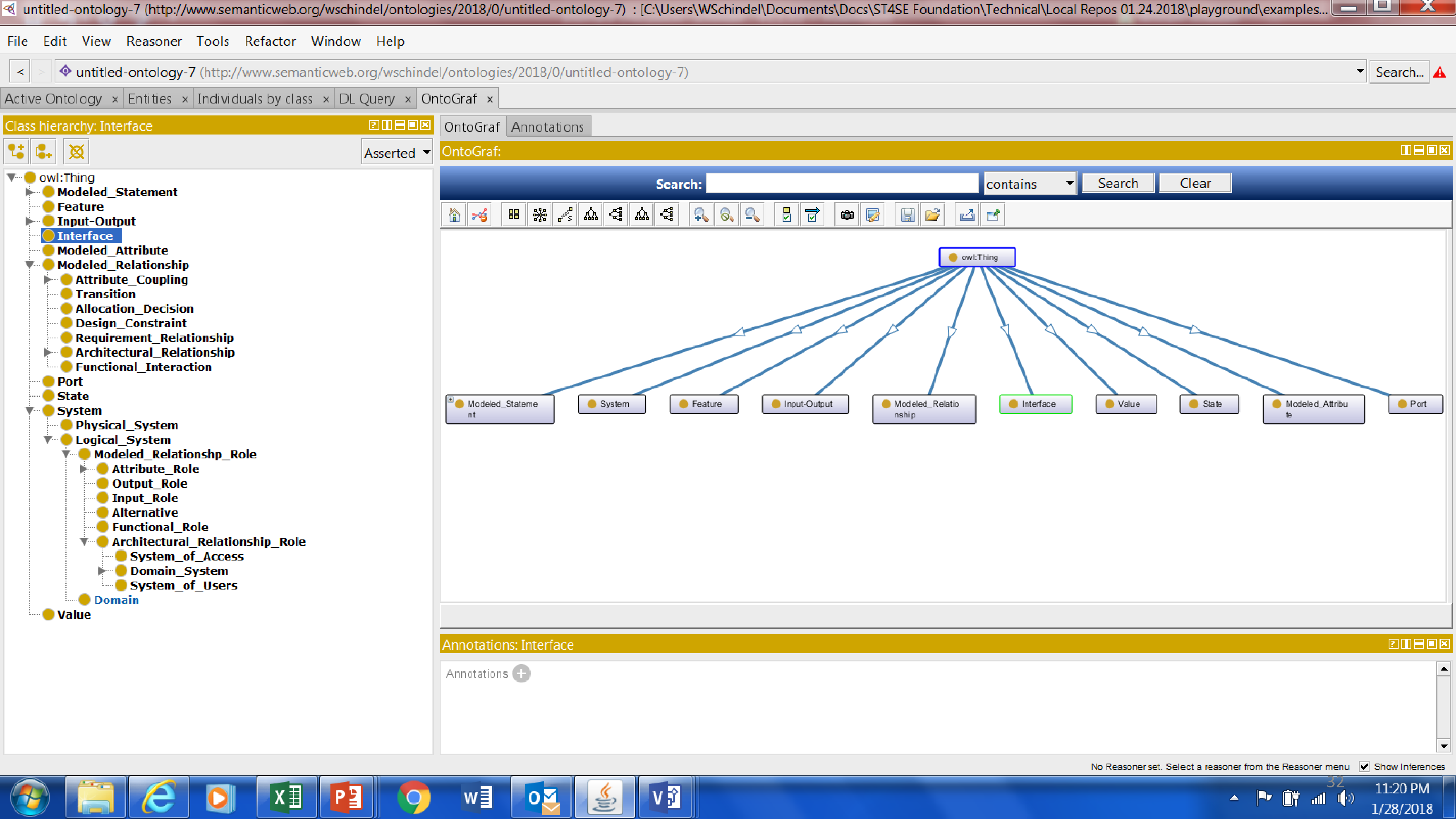
05/29/2009



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\examples...

```
owl:Thing
  Modeled_Statement
  Feature
  Input-Output
  Interface
  Modeled_Attribute
  Modeled_Relationship
    Attribute_Coupling
    Transition
    Allocation_Decision
    Design_Constraint
    Requirement_Relationship
    Architectural_Relationship
    Functional_Interaction
  Port
  State
  System
    Physical_System
    Logical_System
      Modeled_Relationshp_Role
        Attribute_Role
        Output_Role
        Input_Role
        Alternative
        Functional_Role
        Architectural_Relationship_Role
          System_of_Access
          Domain_System
          System_of_Users
      Domain
  Value
```

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences


untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\examples...
File
Edit
View
Reasoner
Tools
Refactor
Window
Help
untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)
Search...
Active Ontology
Entities
Individuals by class
DL Query
OntoGraf
Class hierarchy: Interface
Asserted
OntoGraf
Annotations
OntoGraf:
Search:
contains
Search
Clear
owl:Thing
Modeled_Statement
System
Feature
Input-Output
Modeled_Relationship
Interface
Value
State
Modeled_Attribute
Port
Annotations: Interface
Annotations
No Reasoner set. Select a reasoner from the Reasoner menu
☑ Show Inferences
11:20 PM
1/28/2018

- owl:Thing
  - Modeled_Statement
  - Feature
  - Input-Output
  - Interface
  - Modeled_Attribute
  - Modeled_Relationship
    - Attribute_Coupling
    - Transition
    - Allocation_Decision
    - Design_Constraint
    - Requirement_Relationship
    - Architectural_Relationship
    - Functional_Interaction
  - Port
  - State
  - System
    - Physical_System
    - Logical_System
      - Modeled_Relationshp_Role
        - Attribute_Role
        - Output_Role
        - Input_Role
        - Alternative
        - Functional_Role
        - Architectural_Relationship_Role
          - System_of_Access
          - Domain_System
          - System_of_Users
      - Domain
  - Value

Discussion and plans

-
-
-
-
-

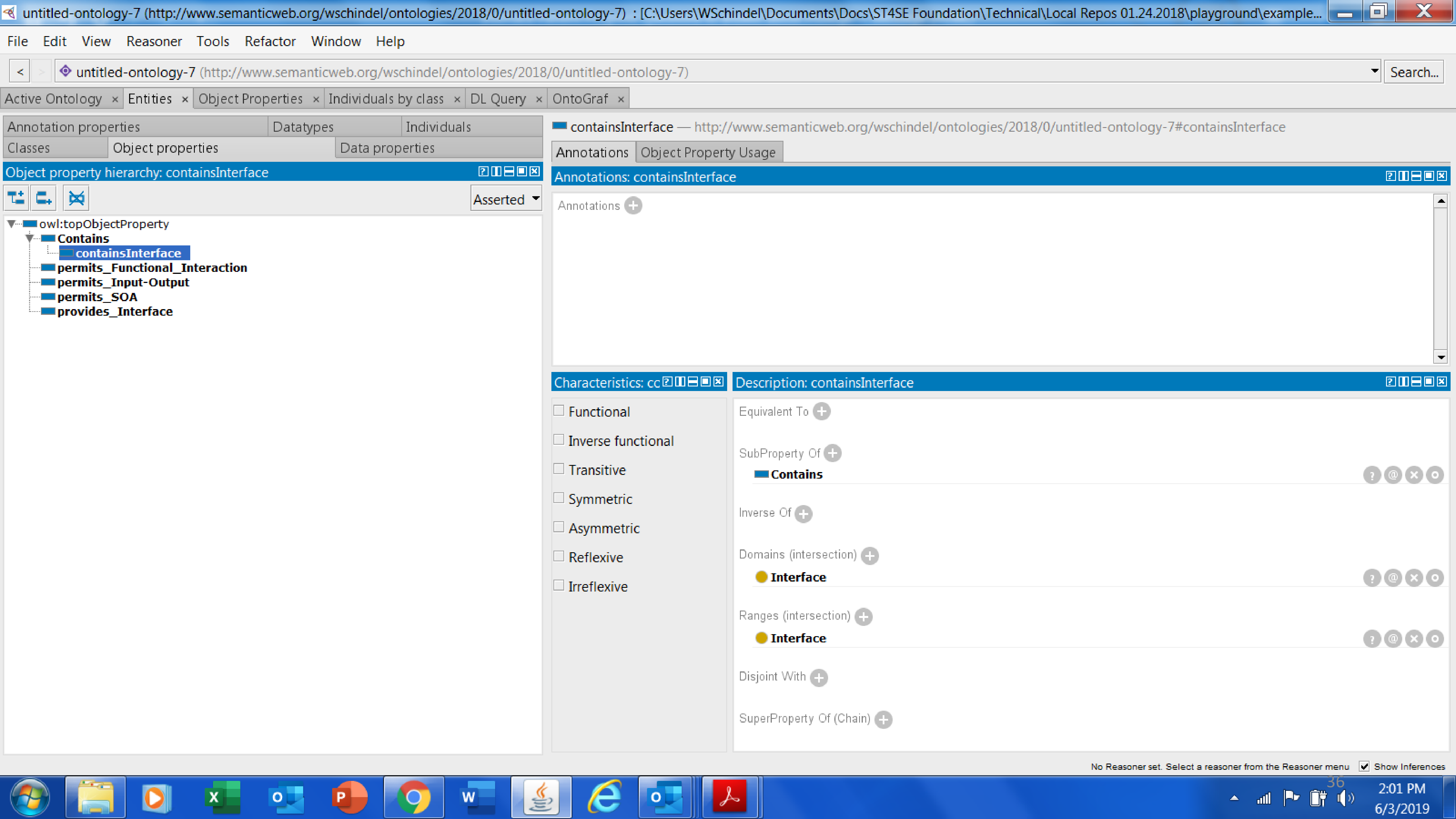
References

- “MBSE Methodology Summary: Pattern-Based Systems Engineering (PBSE), Based On S*MBSE Models”, INCOSE Patterns Working Group, 2015. That document contains a more complete list of related references, and can be retrieved from http://www.omgwiki.org/MBSE/doku.php?id=mbse:methodology#mbse_benchmarking_survey
- INCOSE PBSE Tutorial: http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:patterns:pbse_tutorial_glrc_2016_v1.7.4.pdf
- More examples and materials on WG web wiki site: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
- “Reference Model: Information, Processes, and Automation Associated with S*Representations for System Life Cycle Management”, ICTT System Sciences, 2015.

A little more progress getting Interface Pattern into OWL via Protege

First read the **January 2018** slides, with the basic set up and questions!

Then these slides, which are from **May, 2019**.



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...

untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)

Object property hierarchy: containsInterface

```
owl:topObjectProperty
    Contains
        containsInterface
    permits_Functional_Interaction
    permits_Input-Output
    permits_SOA
    provides_Interface
```

containsInterface — http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#containsInterface

Annotations: containsInterface

Annotations

Characteristics: cc

- □ Functional
- □ Inverse functional
- □ Transitive
- □ Symmetric
- □ Asymmetric
- □ Reflexive
- □ Irreflexive

Description: containsInterface

Equivalent To

SubProperty Of
Contains

Inverse Of

Domains (intersection)
● Interface

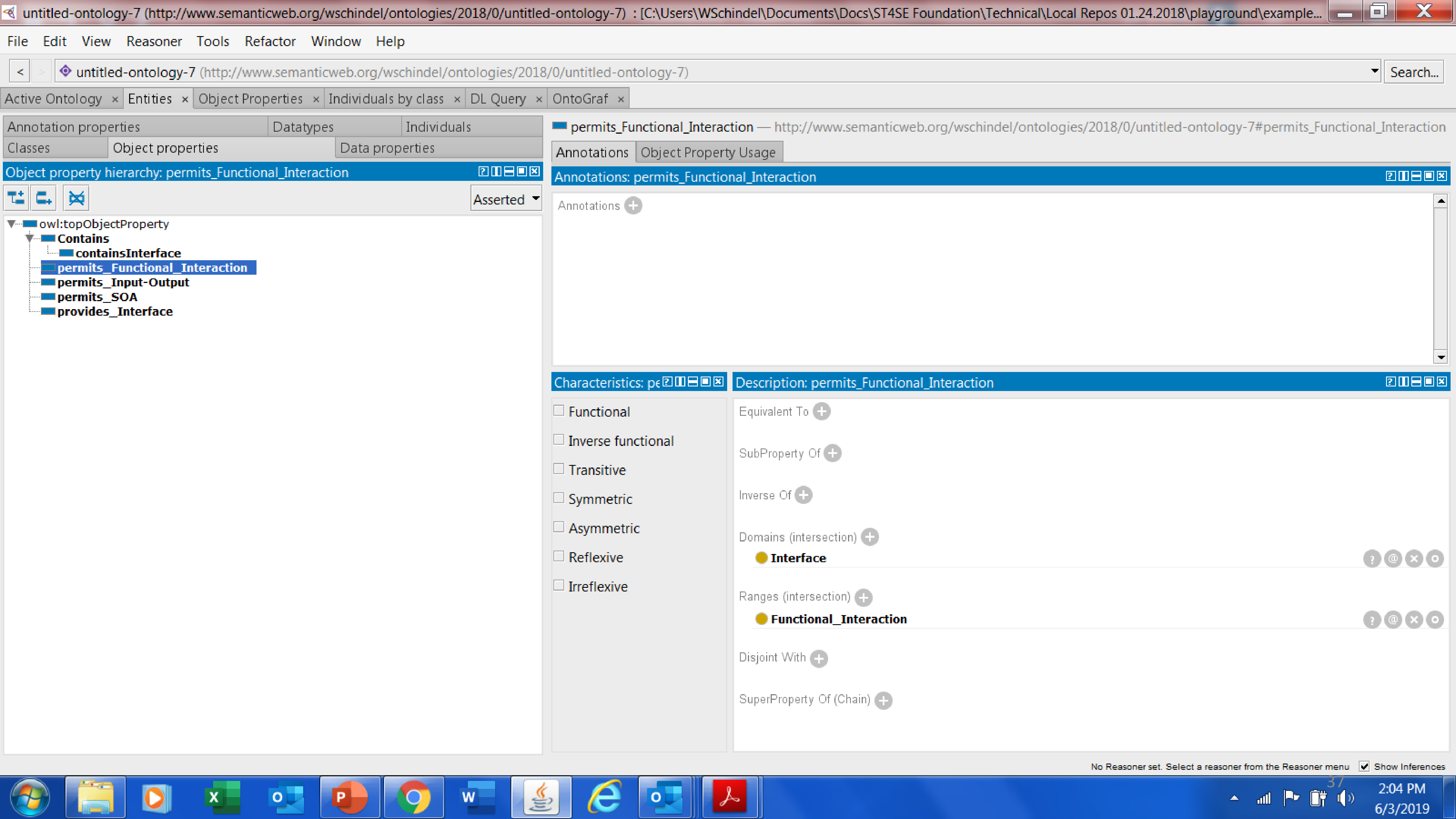
Ranges (intersection)
● Interface

Disjoint With

SuperProperty Of (Chain)

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2:01 PM
6/3/2019



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...
File
Edit
View
Reasoner
Tools
Refactor
Window
Help
untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)
Search...
Active Ontology
Entities
Object Properties
Individuals by class
DL Query
OntoGraf
Annotation properties
Datatypes
Individuals
Classes
Object properties
Data properties
Asserted
Annotations
Object Property Usage

Object property hierarchy: permits_Functional_Interaction

- owl:topObjectProperty
  - Contains
    - containsInterface
  - permits_Functional_Interaction
  - permits_Input-Output
  - permits_SOA
  - provides_Interface

permits_Functional_Interaction — http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#permits_Functional_Interaction

Annotations: permits_Functional_Interaction

Annotations

Characteristics: pe

□ Functional

□ Inverse functional

□ Transitive

□ Symmetric

□ Asymmetric

□ Reflexive

□ Irreflexive

Description: permits_Functional_Interaction

Equivalent To

SubProperty Of

Inverse Of

Domains (intersection)

- Interface

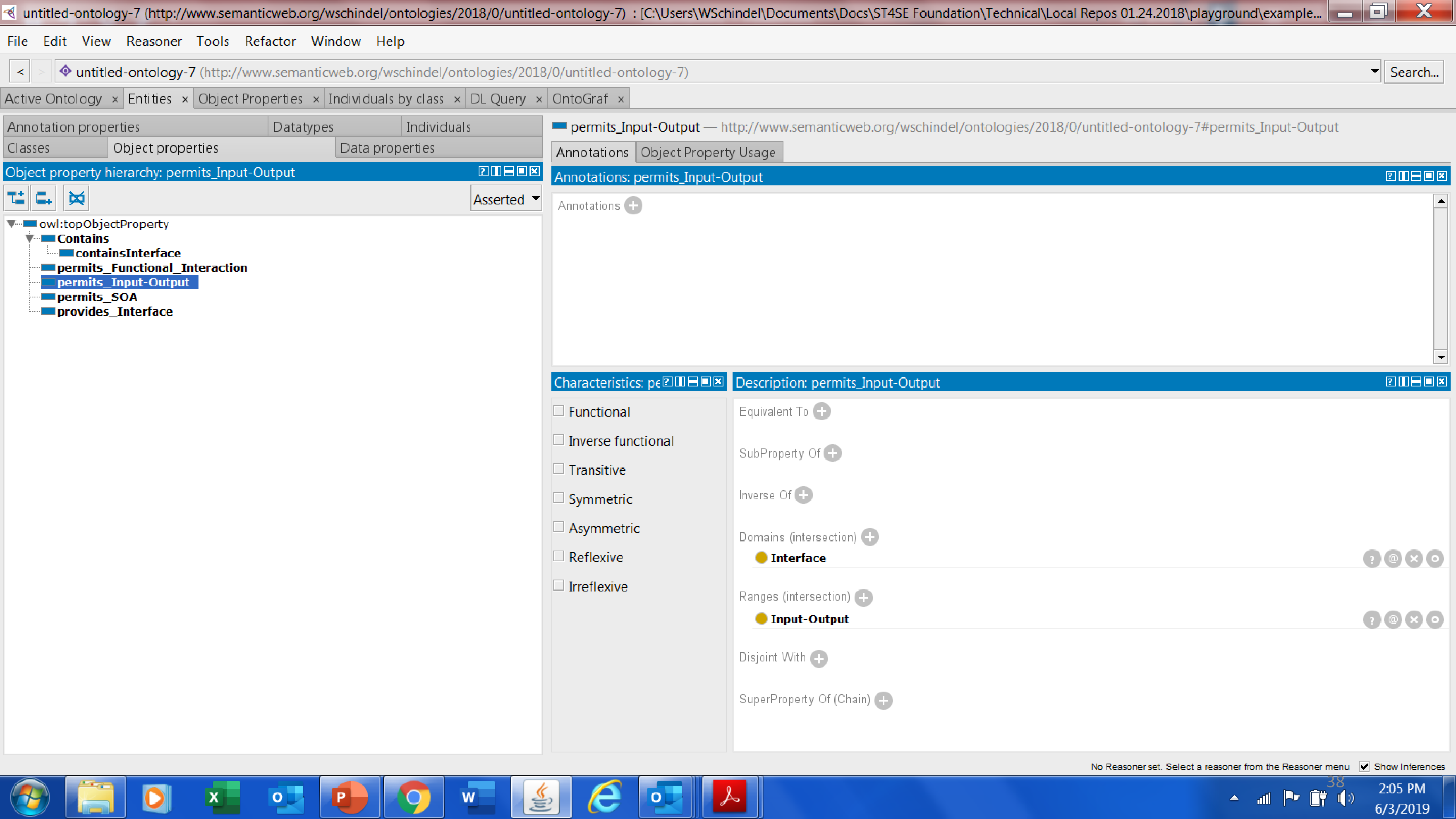
Ranges (intersection)

- Functional_Interaction

Disjoint With

SuperProperty Of (Chain)

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...

untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)

Object property hierarchy: permits_Input-Output

```
owl:topObjectProperty
    Contains
        containsInterface
    permits_Functional_Interaction
    permits_Input-Output
    permits_SOA
    provides_Interface
```

permits_Input-Output — http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#permits_Input-Output

Annotations: permits_Input-Output

Annotations

Characteristics: pe

□ Functional

□ Inverse functional

□ Transitive

□ Symmetric

□ Asymmetric

□ Reflexive

□ Irreflexive

Description: permits_Input-Output

Equivalent To

SubProperty Of

Inverse Of

Domains (intersection)

- Interface

Ranges (intersection)

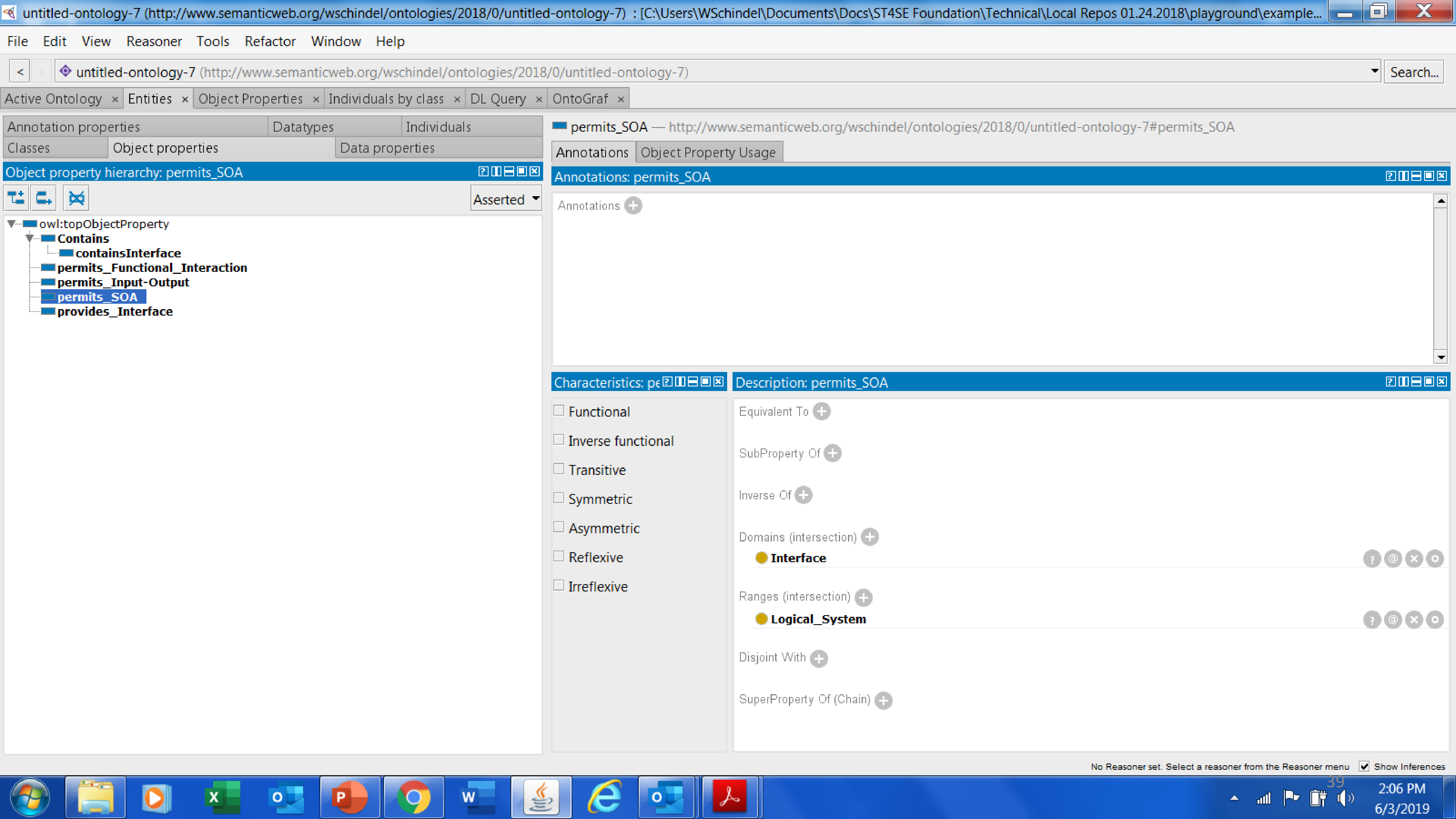
- Input-Output

Disjoint With

SuperProperty Of (Chain)

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2:05 PM
6/3/2019



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...

untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)

Object property hierarchy: permits_SOA

```
owl:topObjectProperty
    Contains
        containsInterface
    permits_Functional_Interaction
    permits_Input-Output
    permits_SOA
    provides_Interface
```

permits_SOA — http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#permits_SOA

Annotations: permits_SOA

Annotations

Characteristics: pe

- ☐ Functional
- ☐ Inverse functional
- ☐ Transitive
- ☐ Symmetric
- ☐ Asymmetric
- ☐ Reflexive
- ☐ Irreflexive

Description: permits_SOA

Equivalent To

SubProperty Of

Inverse Of

Domains (intersection)
- Interface

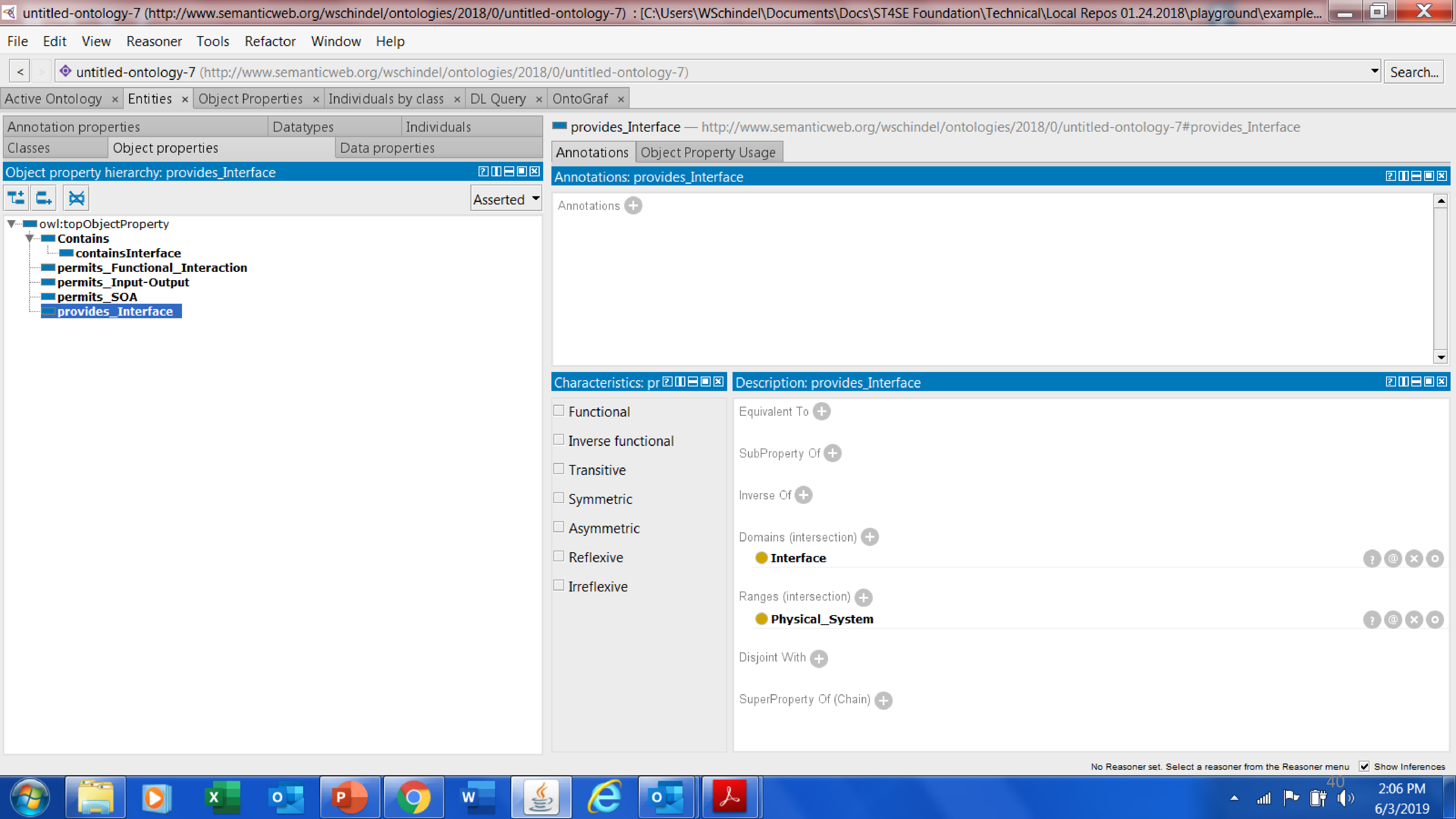
Ranges (intersection)
- Logical_System

Disjoint With

SuperProperty Of (Chain)

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2:06 PM
6/3/2019



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...

untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7)

Object property hierarchy: provides_Interface

- owl:topObjectProperty
  - **Contains**
    - **containsInterface**
  - **permits_Functional_Interaction**
  - **permits_Input-Output**
  - **permits_SOA**
  - **provides_Interface**

provides_Interface — http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#provides_Interface

Annotations: provides_Interface

Annotations

Characteristics: pr

□ Functional

□ Inverse functional

□ Transitive

□ Symmetric

□ Asymmetric

□ Reflexive

□ Irreflexive

Description: provides_Interface

Equivalent To

SubProperty Of

Inverse Of

Domains (intersection)

● **Interface**

Ranges (intersection)

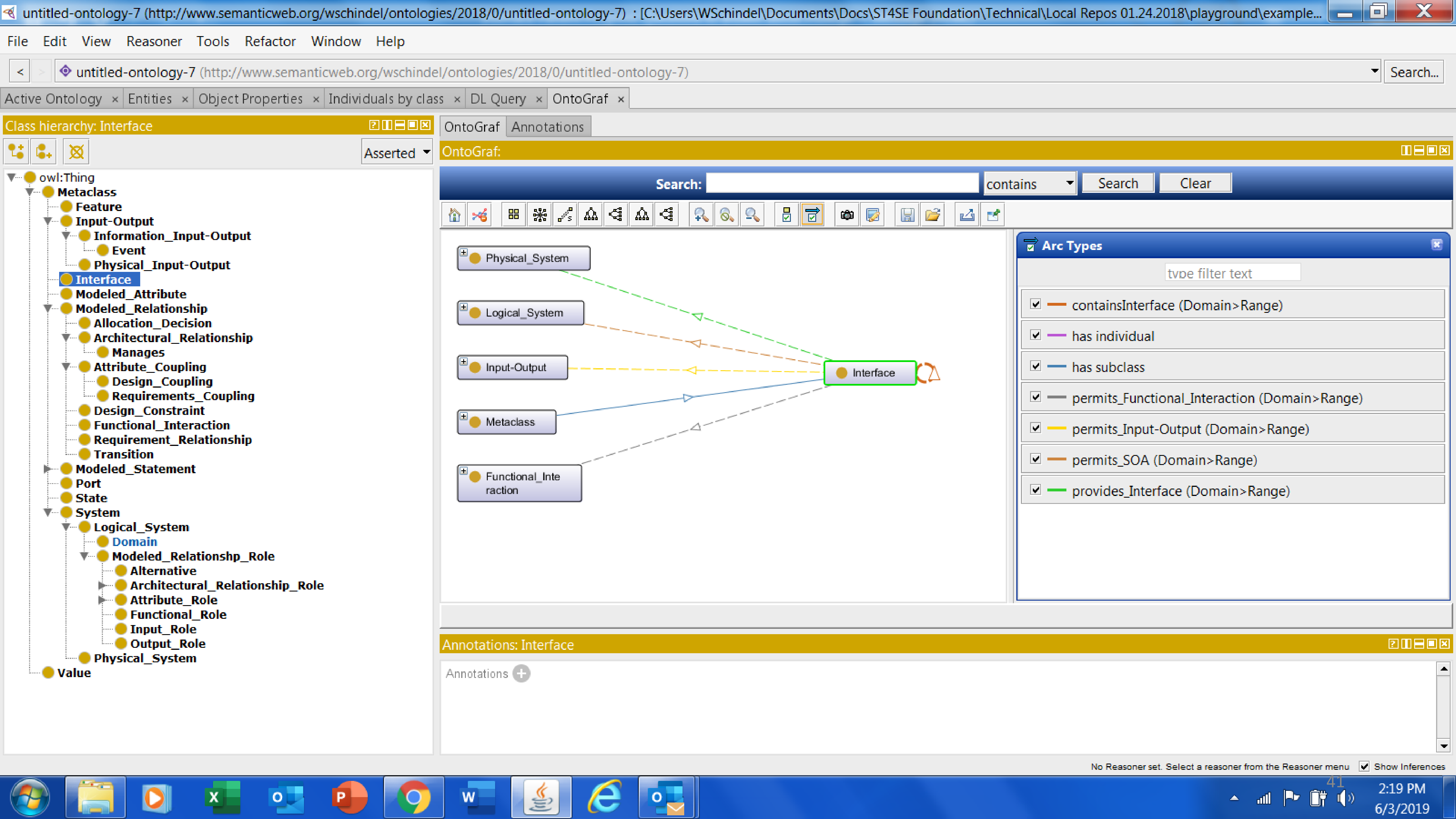
● **Physical_System**

Disjoint With

SuperProperty Of (Chain)

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2:06 PM
6/3/2019



untitled-ontology-7 (http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7) : [C:\Users\WSchindel\Documents\Docs\ST4SE Foundation\Technical\Local Repos 01.24.2018\playground\example...

Class hierarchy: Interface

- owl:Thing
  - Metaclass
    - Feature
    - Input-Output
      - Information_Input-Output
        - Event
      - Physical_Input-Output
    - Interface
    - Modeled_Attribute
    - Modeled_Relationship
      - Allocation_Decision
      - Architectural_Relationship
        - Manages
      - Attribute_Coupling
        - Design_Coupling
        - Requirements_Coupling
      - Design_Constraint
      - Functional_Interaction
      - Requirement_Relationship
      - Transition
    - Modeled_Statement
    - Port
    - State
    - System
      - Logical_System
        - Domain
        - Modeled_Relationshp_Role
          - Alternative
          - Architectural_Relationship_Role
          - Attribute_Role
          - Functional_Role
          - Input_Role
          - Output_Role
      - Physical_System
  - Value

OntoGraf:

Arc Types

- ☑ containsInterface (Domain>Range)
- ☑ has individual
- ☑ has subclass
- ☑ permits_Functional_Interaction (Domain>Range)
- ☑ permits_Input-Output (Domain>Range)
- ☑ permits_SOA (Domain>Range)
- ☑ provides_Interface (Domain>Range)

Annotations: Interface

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2:19 PM
6/3/2019

OWL DL Exported from Above Model in Protégé

06.17.2019
B. Schindel

```
Prefix(:=<http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7#>)
Prefix(owl:=<http://www.w3.org/2002/07/owl#>)
Prefix(rdf:=<http://www.w3.org/1999/02/22-rdf-syntax-ns#>)
Prefix(xml:=<http://www.w3.org/XML/1998/namespace>)
Prefix(xsd:=<http://www.w3.org/2001/XMLSchema#>)
Prefix(rdfs:=<http://www.w3.org/2000/01/rdf-schema#>)


Ontology(<http://www.semanticweb.org/wschindel/ontologies/2018/0/untitled-ontology-7>

Declaration(Class(:Allocation_Decision))
Declaration(Class(:Alternative))
Declaration(Class(:Architectural_Relationship))
Declaration(Class(:Architectural_Relationship_Role))
Declaration(Class(:Attribute_Coupling))
Declaration(Class(:Attribute_Coupling_Map))
Declaration(Class(:Attribute_Role))
Declaration(Class(:Design_Component_Attribute_Role))
Declaration(Class(:Design_Constraint))
Declaration(Class(:Design_Constraint_Statement))
Declaration(Class(:Design_Coupling))
Declaration(Class(:Design_Coupling_Map))
Declaration(Class(:Domain))
Declaration(Class(:Domain_System))
Declaration(Class(:Event))
Declaration(Class(:Feature))
Declaration(Class(:Feature_Attribute_Role))
Declaration(Class(:Functional_Interaction))
Declaration(Class(:Functional_Role))
Declaration(Class(:Information_Input-Output))
Declaration(Class(:Input-Output))
Declaration(Class(:Input_Role))
Declaration(Class(:Interface))
Declaration(Class(:Issue))
Declaration(Class(:Logical_System))
Declaration(Class(:Managed_System))
Declaration(Class(:Management_System))
Declaration(Class(:Manages))
Declaration(Class(:Metaclass))
Declaration(Class(:Modeled_Attribute))
Declaration(Class(:Modeled_Relationship))
Declaration(Class(:Modeled_Relationshp_Role))
Declaration(Class(:Modeled_Statement))
Declaration(Class(:Need))
Declaration(Class(:Output_Role))
Declaration(Class(:Physical_Input-Output))
Declaration(Class(:Physical_System))
Declaration(Class(:Port))
Declaration(Class(:Rationale))
Declaration(Class(:Requirement_Relationship))
Declaration(Class(:Requirement_Statement))
Declaration(Class(:Requirements_Coupling))
Declaration(Class(:Requirements_Coupling_Map))
Declaration(Class(:Role_Attribute_Role))
Declaration(Class(:State))
Declaration(Class(:System))
Declaration(Class(:System_of_Access))
Declaration(Class(:System_of_Users))
Declaration(Class(:Transition))
Declaration(Class(:Value))
Declaration(ObjectProperty(:Contains))
Declaration(ObjectProperty(:containsInterface))
Declaration(ObjectProperty(:permits_Functional_Interaction))
Declaration(ObjectProperty(:permits_Input-Output))
Declaration(ObjectProperty(:permits_SOA))
Declaration(ObjectProperty(:provides_Interface))
############################
#   Object Properties
############################

# Object Property: :containsInterface (:containsInterface)

SubObjectPropertyOf(:containsInterface :Contains)
ObjectPropertyDomain(:containsInterface :Interface)
ObjectPropertyRange(:containsInterface :Interface)

# Object Property: :permits_Functional_Interaction (:permits_Functional_Interaction)

ObjectPropertyDomain(:permits_Functional_Interaction :Interface)
ObjectPropertyRange(:permits_Functional_Interaction :Functional_Interaction)

# Object Property: :permits_Input-Output (:permits_Input-Output)

ObjectPropertyDomain(:permits_Input-Output :Interface)
ObjectPropertyRange(:permits_Input-Output :Input-Output)

# Object Property: :permits_SOA (:permits_SOA)
```

```
ObjectPropertyDomain(:permits_SOA :Interface)
ObjectPropertyRange(:permits_SOA :Logical_System)

# Object Property: :provides_Interface (:provides_Interface)

ObjectPropertyDomain(:provides_Interface :Interface)
ObjectPropertyRange(:provides_Interface :Physical_System)


############################
#   Classes
############################

# Class: :Allocation_Decision (:Allocation_Decision)

SubClassOf(:Allocation_Decision :Modeled_Relationship)

# Class: :Alternative (:Alternative)

SubClassOf(:Alternative :Modeled_Relationshp_Role)

# Class: :Architectural_Relationship (:Architectural_Relationship)

SubClassOf(:Architectural_Relationship :Modeled_Relationship)

# Class: :Architectural_Relationship_Role (:Architectural_Relationship_Role)

SubClassOf(:Architectural_Relationship_Role :Modeled_Relationshp_Role)

# Class: :Attribute_Coupling (:Attribute_Coupling)

SubClassOf(:Attribute_Coupling :Modeled_Relationship)

# Class: :Attribute_Coupling_Map (:Attribute_Coupling_Map)

SubClassOf(:Attribute_Coupling_Map :Modeled_Statement)

# Class: :Attribute_Role (:Attribute_Role)

SubClassOf(:Attribute_Role :Modeled_Relationshp_Role)

# Class: :Design_Component_Attribute_Role (:Design_Component_Attribute_Role)

SubClassOf(:Design_Component_Attribute_Role :Attribute_Role)
```

```
# Class: :Design_Constraint (:Design_Constraint)

SubClassOf(:Design_Constraint :Modeled_Relationship)

# Class: :Design_Constraint_Statement (:Design_Constraint_Statement)

SubClassOf(:Design_Constraint_Statement :Modeled_Statement)

# Class: :Design_Coupling (:Design_Coupling)

SubClassOf(:Design_Coupling :Attribute_Coupling)

# Class: :Design_Coupling_Map (:Design_Coupling_Map)

SubClassOf(:Design_Coupling_Map :Attribute_Coupling_Map)

# Class: :Domain (:Domain)

SubClassOf(:Domain :Logical_System)

# Class: :Domain_System (:Domain_System)

SubClassOf(:Domain_System :Architectural_Relationship_Role)

# Class: :Event (:Event)

SubClassOf(:Event :Information_Input-Output)

# Class: :Feature (:Feature)

SubClassOf(:Feature :Metaclass)

# Class: :Feature_Attribute_Role (:Feature_Attribute_Role)

SubClassOf(:Feature_Attribute_Role :Attribute_Role)

# Class: :Functional_Interaction (:Functional_Interaction)

SubClassOf(:Functional_Interaction :Modeled_Relationship)

# Class: :Functional_Role (:Functional_Role)

SubClassOf(:Functional_Role :Modeled_Relationshp_Role)
```

```
# Class: :Information_Input-Output (:Information_Input-Output)

SubClassOf(:Information_Input-Output :Input-Output)

# Class: :Input-Output (:Input-Output)

SubClassOf(:Input-Output :Metaclass)

# Class: :Input_Role (:Input_Role)

SubClassOf(:Input_Role :Modeled_Relationshp_Role)

# Class: :Interface (:Interface)

SubClassOf(:Interface :Metaclass)

# Class: :Issue (:Issue)

SubClassOf(:Issue :Modeled_Statement)

# Class: :Logical_System (:Logical_System)

SubClassOf(:Logical_System :System)

# Class: :Managed_System (:Managed_System)

SubClassOf(:Managed_System :Domain_System)

# Class: :Management_System (:Management_System)

SubClassOf(:Management_System :Domain_System)

# Class: :Manages (:Manages)

SubClassOf(:Manages :Architectural_Relationship)

# Class: :Modeled_Attribute (:Modeled_Attribute)

SubClassOf(:Modeled_Attribute :Metaclass)

# Class: :Modeled_Relationship (:Modeled_Relationship)

SubClassOf(:Modeled_Relationship :Metaclass)
```

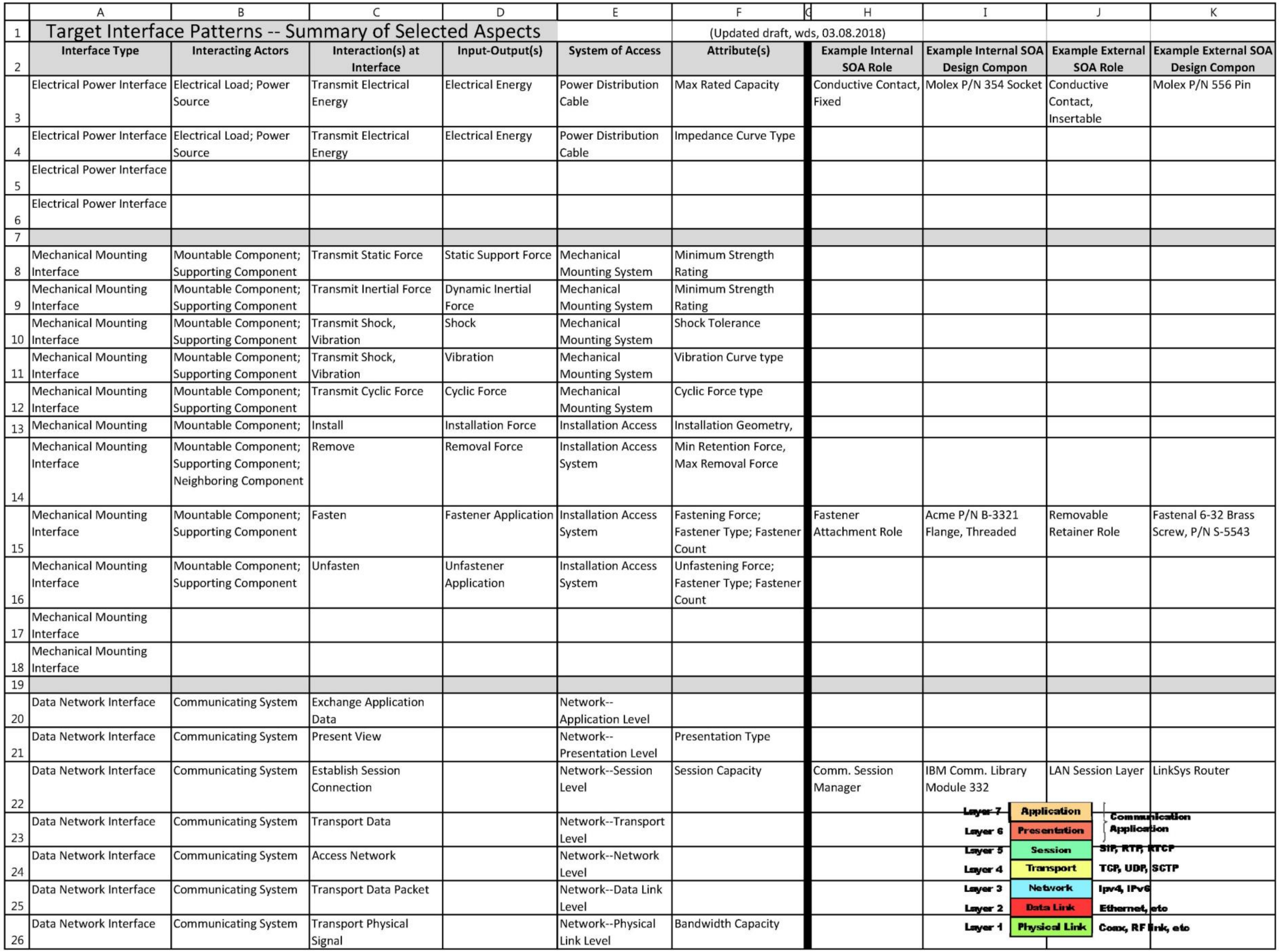
```
# Class: :Modeled_Relationshp_Role (:Modeled_Relationshp_Role)

SubClassOf(:Modeled_Relationshp_Role :Logical_System)

# Class: :Modeled_Statement (:Modeled_Statement)

SubClassOf(:Modeled_Statement :Metaclass)

# Class: :Need (:Need)

SubClassOf(:Need :Modeled_Statement)

# Class: :Output_Role (:Output_Role)

SubClassOf(:Output_Role :Modeled_Relationshp_Role)

# Class: :Physical_Input-Output (:Physical_Input-Output)

SubClassOf(:Physical_Input-Output :Input-Output)

# Class: :Physical_System (:Physical_System)

SubClassOf(:Physical_System :System)

# Class: :Port (:Port)

SubClassOf(:Port :Metaclass)

# Class: :Rationale (:Rationale)

SubClassOf(:Rationale :Modeled_Statement)

# Class: :Requirement_Relationship (:Requirement_Relationship)

SubClassOf(:Requirement_Relationship :Modeled_Relationship)

# Class: :Requirement_Statement (:Requirement_Statement)

SubClassOf(:Requirement_Statement :Modeled_Statement)

# Class: :Requirements_Coupling (:Requirements_Coupling)

SubClassOf(:Requirements_Coupling :Attribute_Coupling)
```

```
# Class: :Requirements_Coupling_Map (:Requirements_Coupling_Map)

SubClassOf(:Requirements_Coupling_Map :Attribute_Coupling_Map)

# Class: :Role_Attribute_Role (:Role_Attribute_Role)

SubClassOf(:Role_Attribute_Role :Attribute_Role)

# Class: :State (:State)

SubClassOf(:State :Metaclass)

# Class: :System (:System)

SubClassOf(:System :Metaclass)

# Class: :System_of_Access (:System_of_Access)

SubClassOf(:System_of_Access :Architectural_Relationship_Role)

# Class: :System_of_Users (:System_of_Users)

SubClassOf(:System_of_Users :Architectural_Relationship_Role)

# Class: :Transition (:Transition)

SubClassOf(:Transition :Modeled_Relationship)


)
```

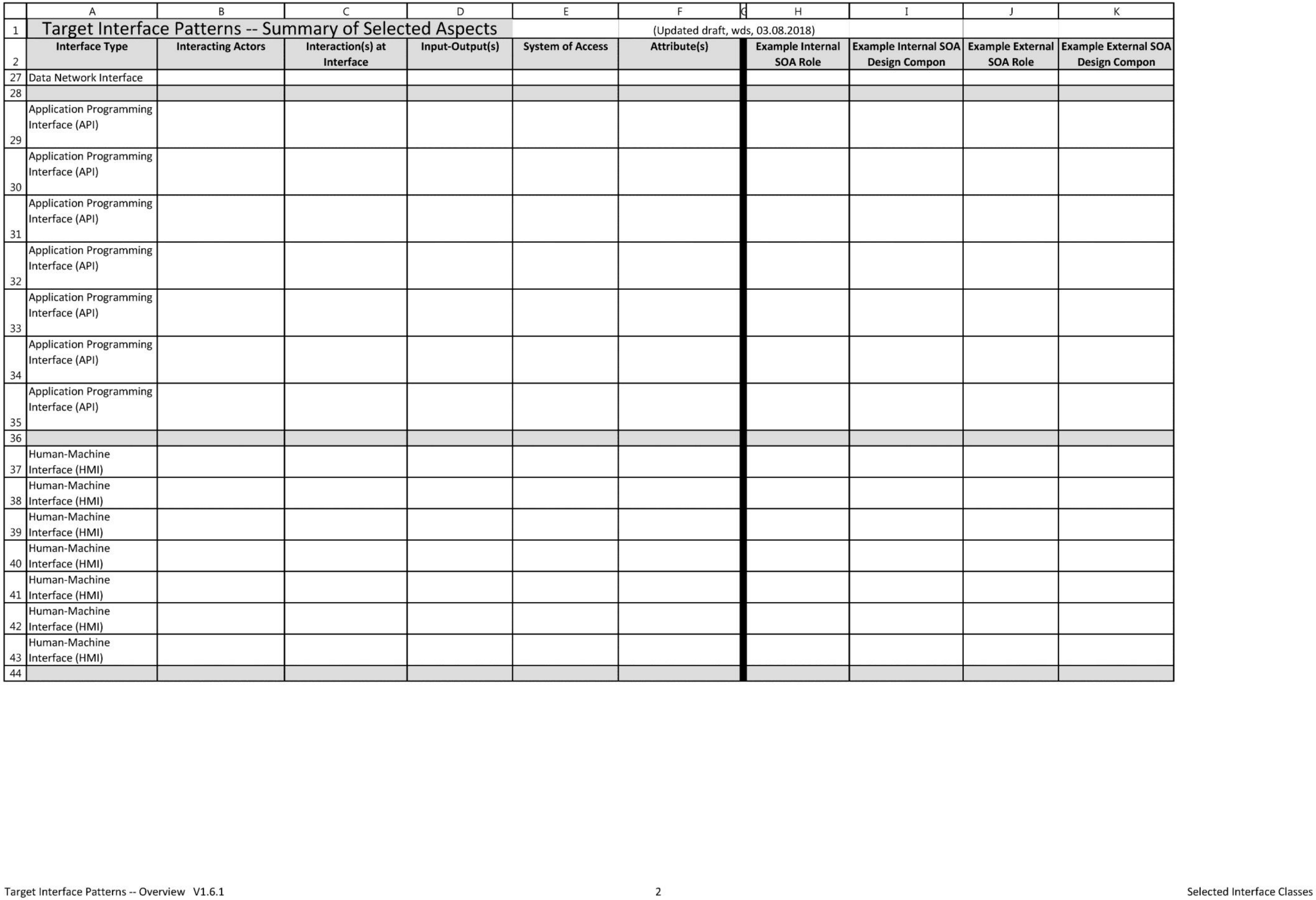
Specialization of Interface Pattern Types

Started by Interface Patterns Project Team,

2017-2018



	A	B	C	D	E	F	G	H	I	J	K
1	Target Interface Patterns -- Summary of Selected Aspects					(Updated draft, wds, 03.08.2018)					
2	**Interface Type**	**Interacting Actors**	**Interaction(s) at Interface**	**Input-Output(s)**	**System of Access**	**Attribute(s)**		**Example Internal SOA Role**	**Example Internal SOA Design Compon**	**Example External SOA Role**	**Example External SOA Design Compon**
3	Electrical Power Interface	Electrical Load; Power Source	Transmit Electrical Energy	Electrical Energy	Power Distribution Cable	Max Rated Capacity		Conductive Contact, Fixed	Molex P/N 354 Socket	Conductive Contact, Insertable	Molex P/N 556 Pin
4	Electrical Power Interface	Electrical Load; Power Source	Transmit Electrical Energy	Electrical Energy	Power Distribution Cable	Impedance Curve Type					
5	Electrical Power Interface										
6	Electrical Power Interface										
7											
8	Mechanical Mounting Interface	Mountable Component; Supporting Component	Transmit Static Force	Static Support Force	Mechanical Mounting System	Minimum Strength Rating					
9	Mechanical Mounting Interface	Mountable Component; Supporting Component	Transmit Inertial Force	Dynamic Inertial Force	Mechanical Mounting System	Minimum Strength Rating					
10	Mechanical Mounting Interface	Mountable Component; Supporting Component	Transmit Shock, Vibration	Shock	Mechanical Mounting System	Shock Tolerance					
11	Mechanical Mounting Interface	Mountable Component; Supporting Component	Transmit Shock, Vibration	Vibration	Mechanical Mounting System	Vibration Curve type					
12	Mechanical Mounting Interface	Mountable Component; Supporting Component	Transmit Cyclic Force	Cyclic Force	Mechanical Mounting System	Cyclic Force type					
13	Mechanical Mounting	Mountable Component;	Install	Installation Force	Installation Access	Installation Geometry,					
14	Mechanical Mounting Interface	Mountable Component; Supporting Component; Neighboring Component	Remove	Removal Force	Installation Access System	Min Retention Force, Max Removal Force					
15	Mechanical Mounting Interface	Mountable Component; Supporting Component	Fasten	Fastener Application	Installation Access System	Fastening Force; Fastener Type; Fastener Count		Fastener Attachment Role	Acme P/N B-3321 Flange, Threaded	Removable Retainer Role	Fastenal 6-32 Brass Screw, P/N S-5543
16	Mechanical Mounting Interface	Mountable Component; Supporting Component	Unfasten	Unfastener Application	Installation Access System	Unfastening Force; Fastener Type; Fastener Count					
17	Mechanical Mounting Interface										
18	Mechanical Mounting Interface										
19											
20	Data Network Interface	Communicating System	Exchange Application Data		Network--Application Level						
21	Data Network Interface	Communicating System	Present View		Network--Presentation Level	Presentation Type					
22	Data Network Interface	Communicating System	Establish Session Connection		Network--Session Level	Session Capacity		Comm. Session Manager	IBM Comm. Library Module 332	LAN Session Layer	LinkSys Router
23	Data Network Interface	Communicating System	Transport Data		Network--Transport Level						
24	Data Network Interface	Communicating System	Access Network		Network--Network Level						
25	Data Network Interface	Communicating System	Transport Data Packet		Network--Data Link Level						
26	Data Network Interface	Communicating System	Transport Physical Signal		Network--Physical Link Level	Bandwidth Capacity					



	A	B	C	D	E	F	H	I	J	K
1	Target Interface Patterns -- Summary of Selected Aspects					(Updated draft, wds, 03.08.2018)				
2	**Interface Type**	**Interacting Actors**	**Interaction(s) at Interface**	**Input-Output(s)**	**System of Access**	**Attribute(s)**	**Example Internal SOA Role**	**Example Internal SOA Design Compon**	**Example External SOA Role**	**Example External SOA Design Compon**
27	Data Network Interface									
28										
29	Application Programming Interface (API)									
30	Application Programming Interface (API)									
31	Application Programming Interface (API)									
32	Application Programming Interface (API)									
33	Application Programming Interface (API)									
34	Application Programming Interface (API)									
35	Application Programming Interface (API)									
36										
37	Human-Machine Interface (HMI)									
38	Human-Machine Interface (HMI)									
39	Human-Machine Interface (HMI)									
40	Human-Machine Interface (HMI)									
41	Human-Machine Interface (HMI)									
42	Human-Machine Interface (HMI)									
43	Human-Machine Interface (HMI)									
44										

Target Interface Patterns -- Overview V1.6.1 2 Selected Interface Classes