Model-Based Systems Engineering $\rightarrow$ Semantics + Data Mining

Mark A. Austin

University of Maryland

austin@umd.edu

ENCE 688P, Fall Semester 2020

September 14, 2020

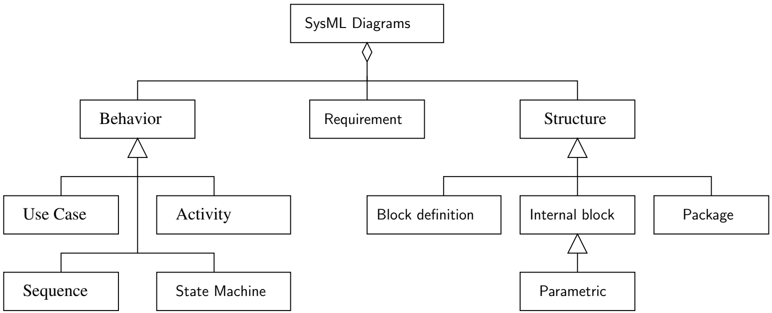
Overview

Part 2

Model-based Systems Engineering

System Modeling Techniques

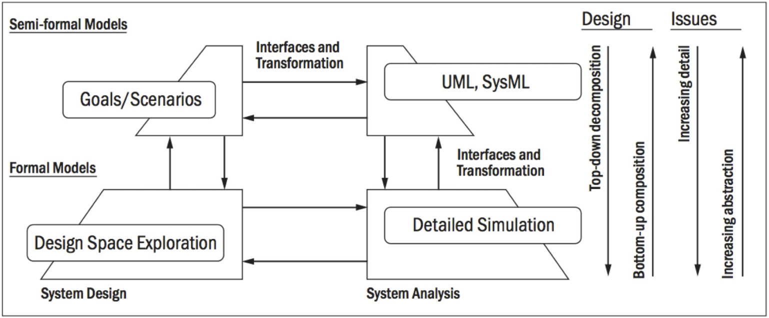
Taxonomy of diagrams in SysML:



Pillars of SysML: Structure, Behavior, Requirements, and Parametric Diagrams.

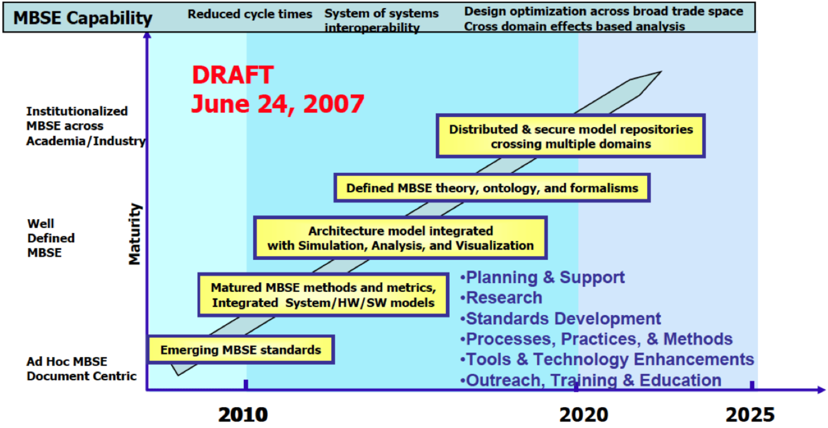
System Modeling Techniques

Use multi-scale approaches to system modeling:



- Semi-Formal Models: View the complete system (efficiency).
- Formal Models: Detailed view of the actual system (accuracy).

INCOSE: MBSE Capability 2020-2025



Ontologies and Ontology-Enabled Computing

Definition of an Ontology

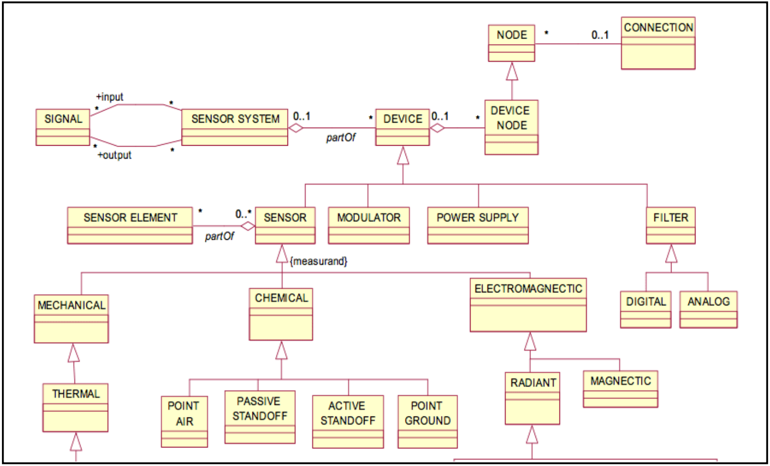
Definition (Ontology)

An ontology is a set of knowledge terms, including the vocabulary, the semantic interconnections, and some simple rules of inference and logic for some particular topic or domain.

Three Goals:

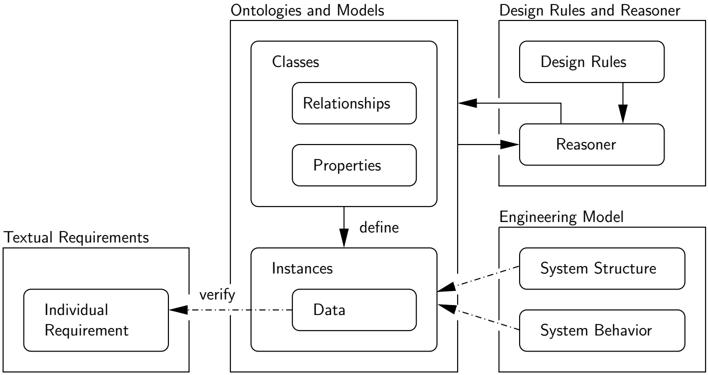
- Provide a semantic representation of each entity and its relationships to other entities;
- Provide constraints and rules that permit reasoning within the ontology;
- Describe behavior associated with stated or inferred facts.

High-Level Sensor Ontology



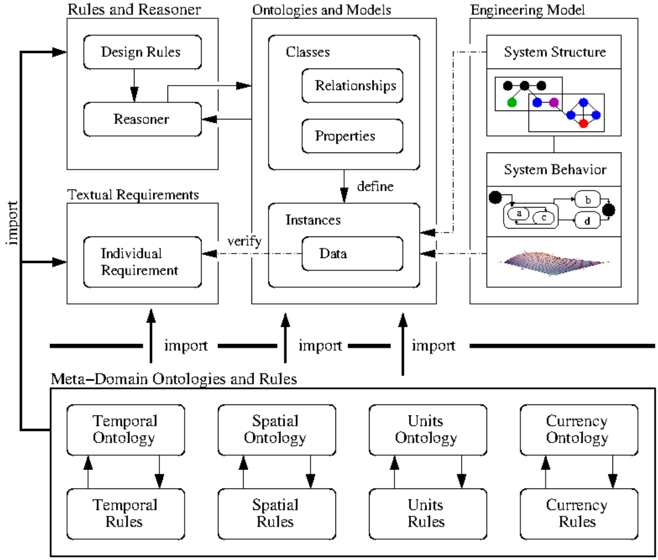
Ontologies and Rule Sets

Framework for Ontology-Enabled Design Assessment (Version 1):



Source: Parastoo Delgoshaei, MSSE Student, 2010-2012. Ph.D. Student in Civil Systems, 2013-2017.

Framework for Model-Based Design



Ontologies and Rule Sets

Benefits of Rule-Based Approaches to Problem Solving:

- Rules that represent policies are easily communicated and understood,
- Rules retain a higher level of independence than logic embedded in systems,
- Rules separate knowledge from its implementation logic, and
- Rules can be changed without changing source code or underlying model.

Benefits of Rules

A rule-based approach to problem solving is particularly beneficial when the application logic is dynamic.

Semantic Web Support for Ontologies

Goals of the WWW

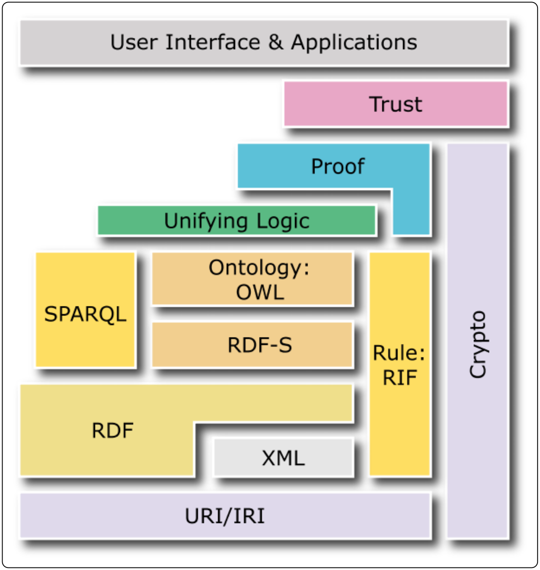
In his original vision for the World Wide Web, Tim Berners-Lee described two key objectives:

- To make the Web a collaborative medium, and
- To make the Web understandable and, thus, processable by machines.

Goals of the Semantic Web

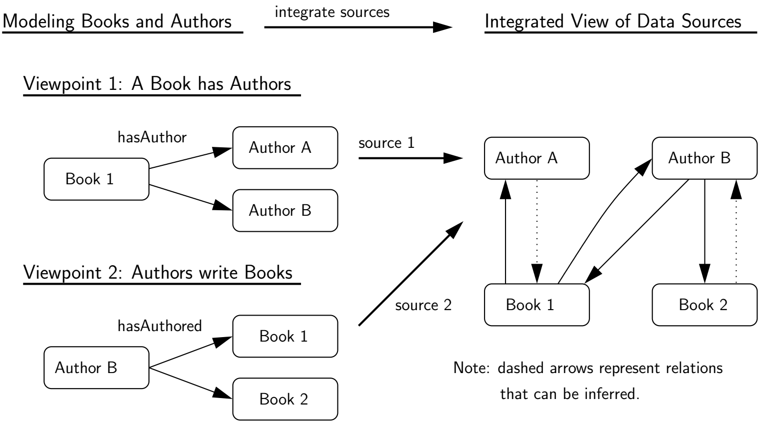
Give information a well-defined meaning, thereby creating a pathway for machine-to-machine communication and automated services based on descriptions of semantics.

Semantic Web Support for Ontologies



Semantic Web Support for Ontologies

Process for merging trees of data into graphs:



Note: dashed arrows represent relations that can be inferred.

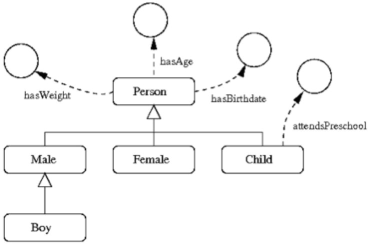
Example 1. A Simple Family Model

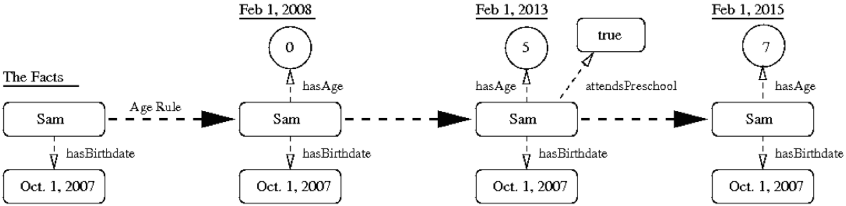
Fact. Sam is a boy. He was born October 1, 2007.

Rule 1: For a given date of birth, a built-in function getAge() computes a person's age.

Rule 2: A child is a person with age < 18.

Rule 3: Children who are age 5 attend preschool.





Example 1. Family Semantic Model

Create Family Individuals:

```
mark =   male.createIndividual(ns + "Mark");
sam  =    boy.createIndividual(ns + "Sam");
nina = female.createIndividual(ns + "Nina");

// Statements "Sam has birthdate 2007-10-01" and "Sam has weight 35"

Literal    dob01 = model.createTypedLiteral("2007-10-01", ...XSDdate );
Statement samdob = model.createStatement( sam, hasDOB, dob01 );
model.add ( samdob );

Literal weight35 = model.createTypedLiteral("35.0", ...XSDdouble );
Statement samw35 = model.createStatement( sam, hasWeight, weight35 );
model.add ( samw35 );
```

Facts in the Simple Family Model:

```
<rdf:Description rdf:about="http://austin.org/family#Sam">
   <j:hasWeight    rdf:datatype="http://www.w3.org/2001/XMLSchema#double"> 35.0 </j:hasWeight>
   <j:hasBirthDate rdf:datatype="http://www.w3.org/2001/XMLSchema#date"> 2007-10-01 </j:hasBirthDate>
   <rdf:type rdf:resource="http://austin.org/family#Boy"/>
</rdf:Description>
.........
```

Example 1. Family Rules (Apache Jena Rules)

Apache Jena Rules:

```
@prefix af:  <http://austin.org/family#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

// Rule 01: Propogate class hierarchy relationships ....

[ rdfs01: (?x rdfs:subClassOf ?y), notEqual(?x,?y) ->
        [ (?a rdf:type ?y) <- (?a rdf:type ?x)] ]

// Rule 02: Identify a person who is also a child ...

[ Child: (?x rdf:type af:Person) (?x af:hasAge ?y) lessThan(?y, 18) ->
         (?x rdf:type af:Child) ]

// Rule 03: See if a child attends preschool ...

[ Preschool: (?x rdf:type af:Child) (?x af:hasAge ?y)
             equal(?y, 5) -> (?x af:attendsPreSchool af:True) ]

// Rule 04: Compute and store the age of a person ....

[ GetAge: (?x rdf:type af:Person) (?x af:hasBirthDate ?y)
          getAge(?y,?z) -> (?x af:hasAge ?z) ]
```

Example 1. Query Transformed Semantic Model

```
Statements: Sam ...
======================================================================

Statement[1] Subject  : http://austin.org/family#Sam
             Predicate: http://austin.org/family#hasAge
             Object   : "5.0^^http://www.w3.org/2001/... #double"

Statement[2] Subject  : http://austin.org/family#Sam
             Predicate: http://www.w3.org/1999/02/... s#type
             Object   : http://austin.org/family#Child

Statement[3] Subject  : http://austin.org/family#Sam
             Predicate: http://austin.org/family#attendsPreSchool
             Object   : http://austin.org/family#True

Statement[4] Subject  : http://austin.org/family#Sam
             Predicate: http://austin.org/family#hasWeight
             Object   : "35.0^^http://www.w3.org/2001/... #double"

Statement[5] Subject  : http://austin.org/family#Sam
             Predicate: http://austin.org/family#hasBirthDate
             Object   : "2007-10-01^^http://www.w3.org/2001/... #date"

Statement[6] Subject  : http://austin.org/family#Sam
             Predicate: http://www.w3.org/1999/02/... #type
             Object   : http://austin.org/family#Boy

======================================================================
```

Distributed System Behavior Modeling

Small Networks of Semantic Graphs Employ Software Design Patterns

MSSE/Ph.D. (Civil Systems) Students

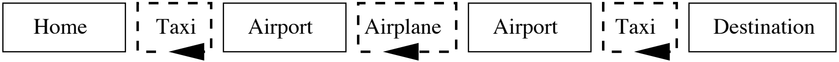
1. Parastoo Delgoshaei (2013-2017);
2. Maria Coelho (2015-present).

Motivation

ENCE 688P: Behaviors in the built environment are distributed and concurrent:

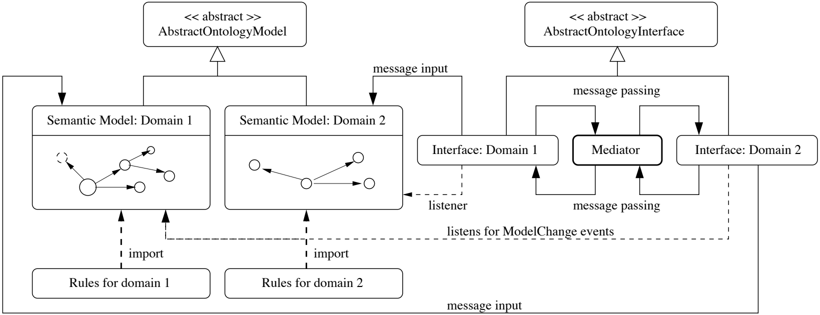
- Cities are system of systems.
- City subsystems may have a preference to operating as independently as possible from the other subsystems.
- Strategic collaboration among subsystems is often needed to either avoid cascading failures across systems and/or recover from a loss of functionality.

Systems-of-systems need not be complicated:

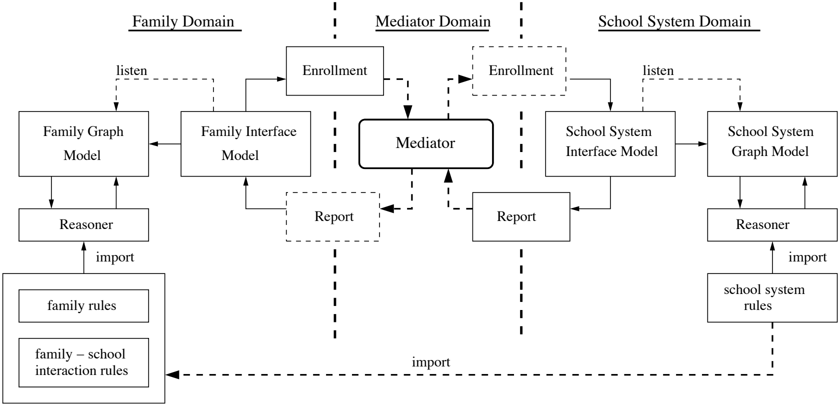


Distributed Behavior Modeling (Initial)

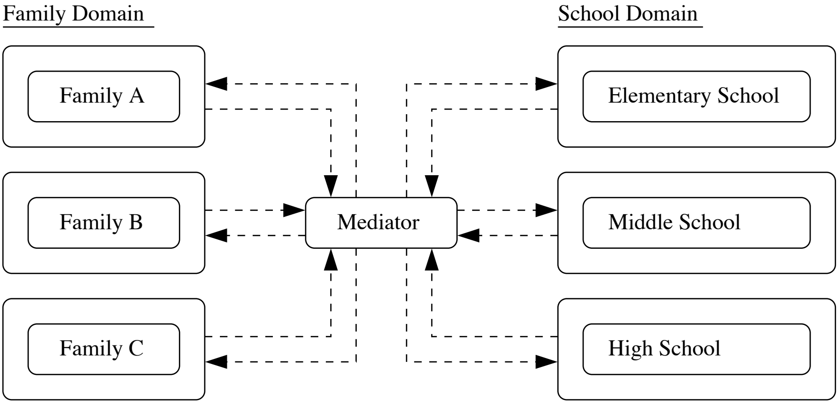
Initial Idea: Use semantic graphs to model behavior of individual entities (e.g., a family). Wrap entities with interfaces. Enable communication among entities with message passing.



Example 2. Family-School System Dynamics



Example 2. Framework for Communication



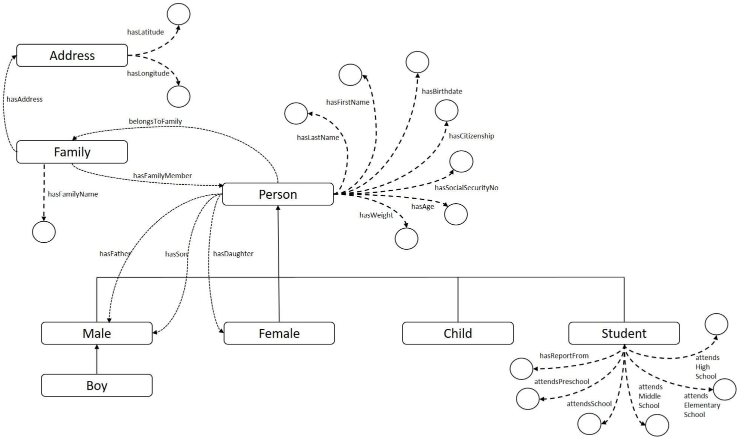
Example 2. Family Datafile (XML)

```xml
<?xml version="1.0" encoding="UTF-8"?>
<FamilyModel author="Maria Coelho" date="2017" source="UMD">
<Family>
    <attribute text="FamilyName" value="Austin"/>
    <attribute text="Address" value="6242 Heather Glen Way, Clarksville, MD 21029"/>
    <Person>
        <attribute text="Type" value="Male"/>
        <attribute text="FirstName" value="Mark"/>
        <attribute text="MiddleName" value="William"/>
        <attribute text="LastName" value="Austin"/>
        <attribute text="BirthDate" value="1704-06-10"/>
        <attribute text="Weight" value="170.0"/>
        <attribute text="Citizenship" value="New Zealand"/>
        <attribute text="SocialSecurity" value="111"/>
    </Person>
    <Person>
        ... description of other Austin family members ....
    </Person>
</Family>
<Family>
    <attribute text="FamilyName" value="Jones"/>
    <attribute text="Address" value="5807 Laurel Leaves Ln, Clarksville, MD 21029"/>
    <Person>
        ... description of Jones family members....
    </Person>
</Family>
</FamilyModel>
```

Example 2. School Datafile (XML)

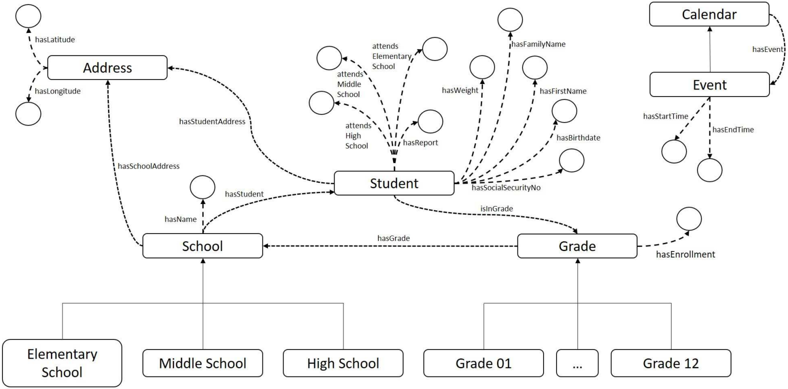
```
<?xml version="1.0" encoding="UTF-8"?>
<SchoolSystemModel author="Maria Coelho" date="2017" source="UMD">
   <School>
       <attribute text="Type" value="High School"/>
       <attribute text="Name" value="River Hill High School"/>
       <attribute text="Grade" value="Grade09"/>
       <attribute text="Grade" value="Grade10"/>
       <attribute text="Grade" value="Grade11"/>
       <attribute text="Grade" value="Grade12"/>
       <attribute text="Report Period Start Time" value="2016-09-01T00:00:00"/>
       <attribute text="Report Period End Time" value="2020-10-20T00:00:00"/>
   </School>
   <School>
       ... description of Clarksville Middle School ...
   </School>
   <School>
       ... description of Pointers Run Elementary School ...
   </School>
</SchoolSystemModel>
```

Example 2. Family and School Ontologies



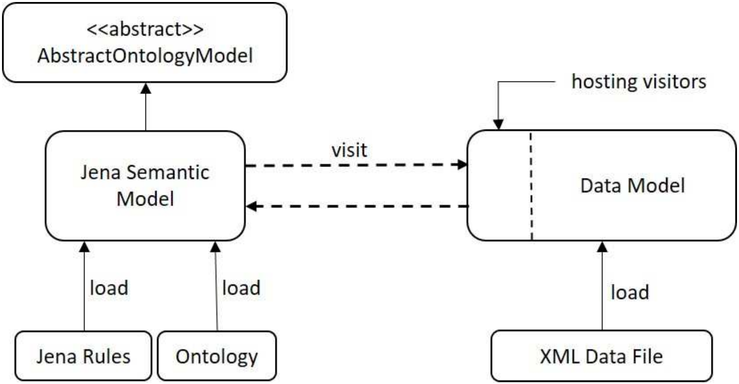
Source: Maria Coelho, MS Thesis, 2017.

Example 2. Family and School Ontologies



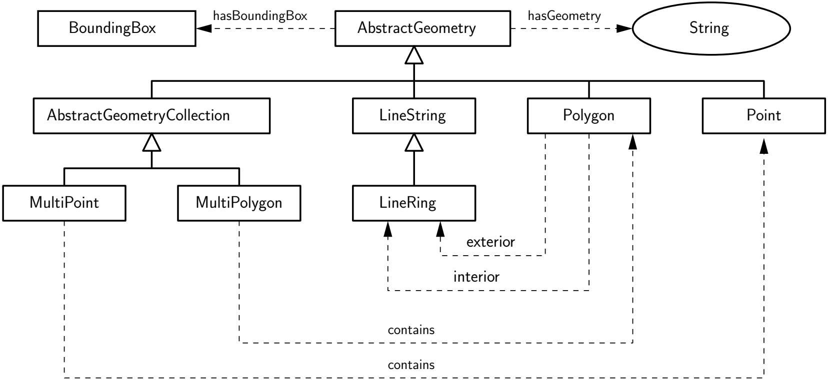
Source: Maria Coelho, MS Thesis, 2017.

Example 2. Populating Models with Data

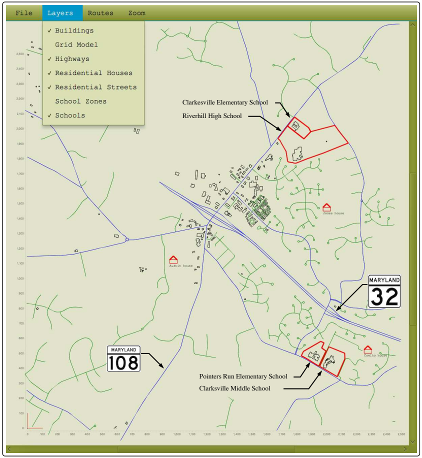


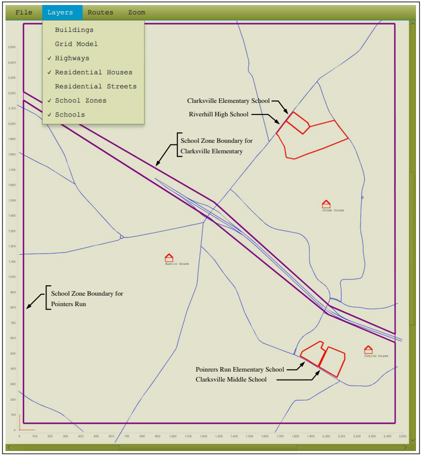
Example 3. Spatial Ontology

Abbreviated Spatial Ontology:



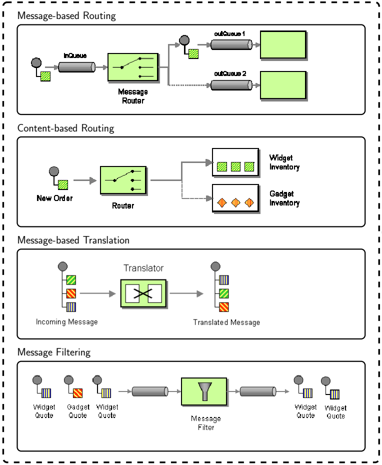
Example 3. Family-School-Urban-Geography Dynamics



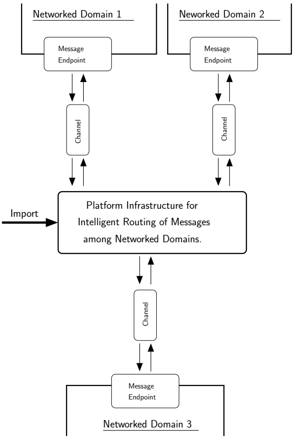


Future Work. Smart Messages with Apache Camel

Mechanisms for Message Transmisson and Processing in Apache Camel.



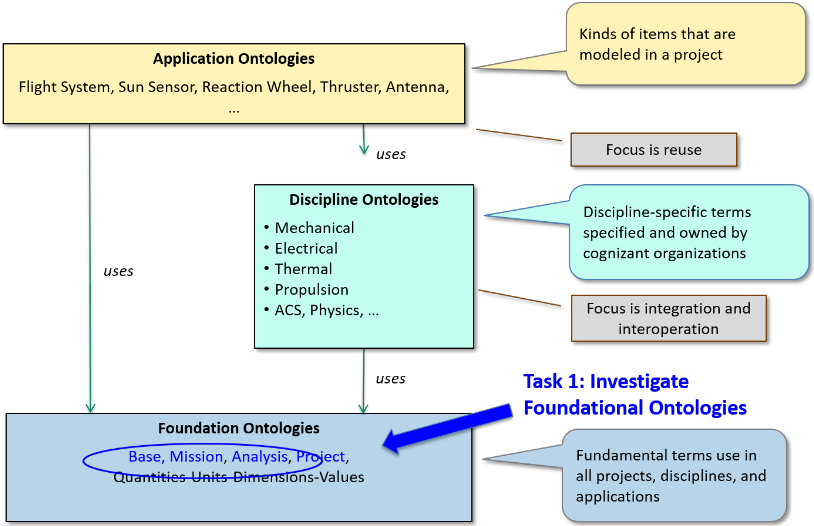
Distributed System Behavior Modeling



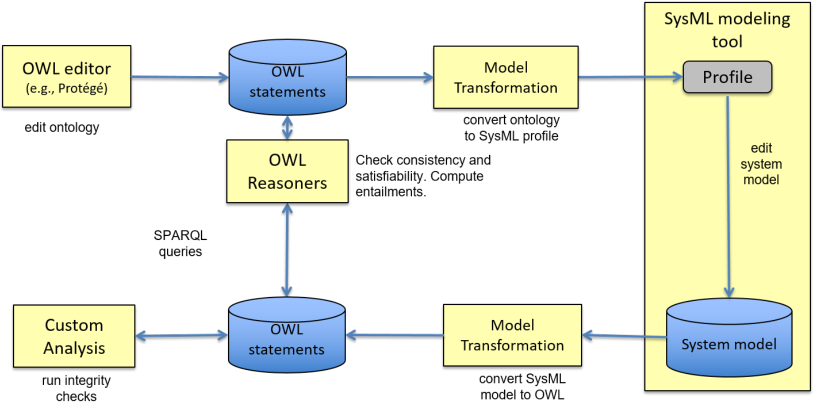
Ontology-Enabled Computing at JPL

Time frame: 2000-2006

Side-by-Side: Semantic/SysML Modeling at JPL



Side-by-Side: Semantic/SysML Modeling at JPL



Task 2: Investigate opportunities adding value to the MBSE process through integration of OWL ontologies and reasoning mechanisms with state-of-the-art SysML tools such as MagicDraw. How well does the proposed interaction of OWL and SysML actually work? What is actually be transformed in the model transformations? Is the model transformation process robust?

Analysis Procedure at UMD

1. Load the individual ontologies in Jena (e.g., base.owl, analysis.owl, mechanical.owl, etc, etc).
2. Systematically traverse the semantic graph.
3. For each class, print:
   1. Name of the class.
   2. The list of super classes.
   3. The list of subclasses classes.
   4. The list of data properties and object properties.
4. Record the number of classes and model size (i.e., number of statements in semantic graph).
5. Identify SWRL rules (if they exist).
6. Use VOWL to visualize the ontology (classes, data properties, object properties).

Note: At this point there are no individuals.

Analysis Procedure at UMD

Here's what a typical class looks like:

```
--- Full Name: http://imce.jpl.nasa.gov/foundation/analysis/analysis#Analysis

--- Superclass: http://imce.jpl.nasa.gov/foundation/analysis/analysis#Explanation ...

--- Subclass: http://imce.jpl.nasa.gov/foundation/analysis/analysis#TradeStudy ...
--- Subclass: http://imce.jpl.nasa.gov/foundation/analysis/analysis#KeyRequirementsExplanation ...
--- Subclass: http://imce.jpl.nasa.gov/foundation/analysis/analysis#DrivingRequirementsExplanation ...
--- Subclass: http://imce.jpl.nasa.gov/foundation/analysis/analysis#CostEstimate ...

--- Data Property Name: http://imce.jpl.nasa.gov/foundation/base/base#hasShortName ...
---             Domain: http://imce.jpl.nasa.gov/foundation/base/base#IdentifiedElement ...

... six data properties removed ...

--- Data Property Name: http://imce.jpl.nasa.gov/foundation/base/base#hasIndexEntry ...
---             Domain: http://imce.jpl.nasa.gov/foundation/base/base#IdentifiedElement ...

--- Object Property: http://imce.jpl.nasa.gov/foundation/analysis/analysis#isCharacterizedBy ...
---           Range: http://imce.jpl.nasa.gov/foundation/analysis/analysis#Characterization ...

... nine object properties removed ...

--- Object Property: http://imce.jpl.nasa.gov/foundation/analysis/analysis#isExplainedBy ...
---           Range: http://imce.jpl.nasa.gov/foundation/analysis/analysis#Explanation ...
```

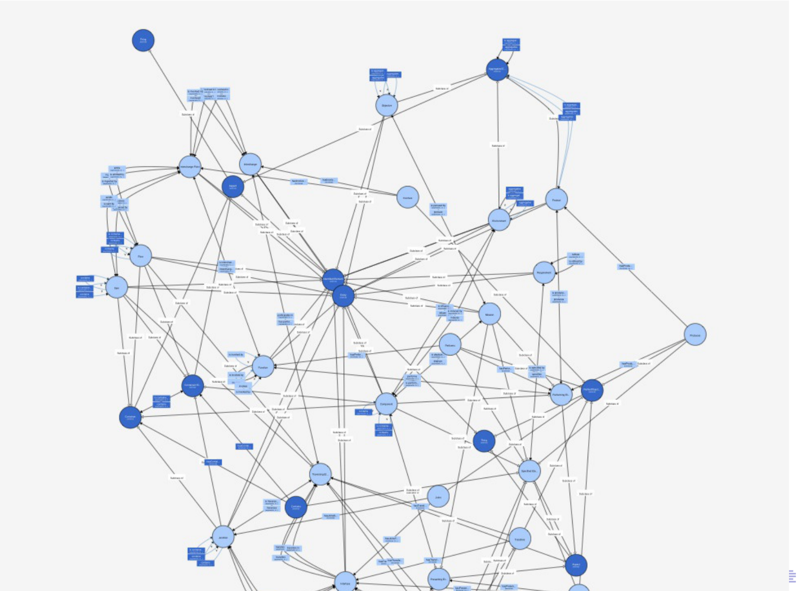
IMCE Ontologies (Number of Classes/Model Size)

Foundation Ontologies	Number of Classes	Model Size
Analysis.owl	101	2,769
Base.owl	13	–
Mission.owl	64	1,991
Project.owl	227	4,920
Time.owl	48	1,000

Discipline Ontologies	Number of Classes	Model Size
Mechanical.owl	105	–
Electrical.owl	243	5,074

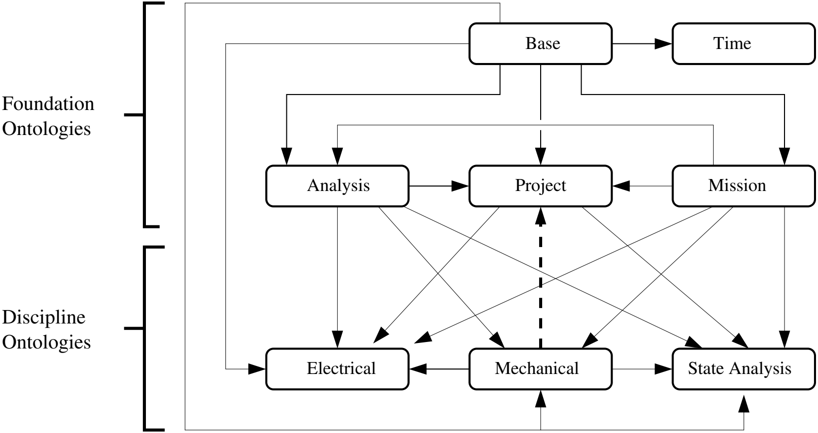
Miscellaneous Ontologies	Number of Classes	Model Size
SysML.owl	877	21,079

Panoramic View of Mission Ontology



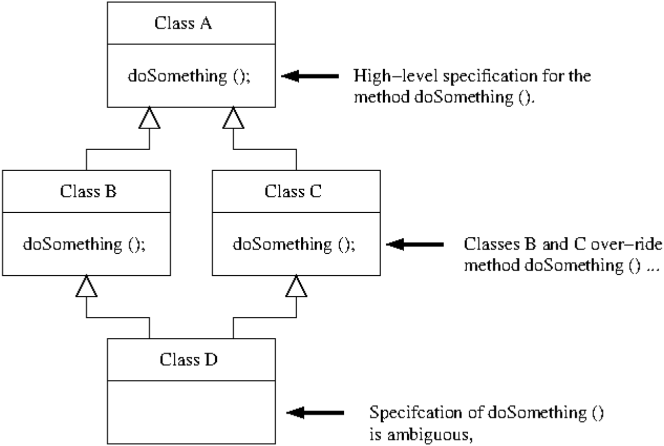
Concern 1: Dependencies Among Ontologies

What happened to notions of modularity?



Concern 2: Multiple Inheritance Relationships

Use of sultiple inheritance relationships in software:



Concern 2: Multiple Inheritance Relationships

Excessive use of multiple inheritance:

```
Named Class(79): Item

--- Full Name: http://imce.jpl.nasa.gov/foundation/mission/mission#Item

--- Superclass: http://imce.jpl.nasa.gov/backbone/imce.jpl.nasa.gov/foundation/mission/mission#Entity ...
--- Superclass: http://imce.jpl.nasa.gov/foundation/base/base#ContainedElement ...
--- Superclass: http://imce.jpl.nasa.gov/foundation/base/base#Container ...
--- Superclass: http://imce.jpl.nasa.gov/foundation/base/base#IdentifiedElement ...
--- Superclass: http://imce.jpl.nasa.gov/foundation/mission/mission#TraversingElement ...

--- Subclass: http://imce.jpl.nasa.gov/foundation/mission/mission#MaterialItem ...
--- Subclass: http://imce.jpl.nasa.gov/foundation/mission/mission#Message ...

--- Data Property Name: http://imce.jpl.nasa.gov/foundation/base/base#hasShortName ...
---             Domain: http://imce.jpl.nasa.gov/foundation/base/base#IdentifiedElement ...
--- Data Property Name: http://imce.jpl.nasa.gov/foundation/base/base#hasDescription ...

... etc ...
```

The Data-Ontology-Rule Footing

Building Block for Semantic Modeling and Event-driven Execution of Multi-Domain Systems

MSSE/Ph.D. (Civil Systems) Students

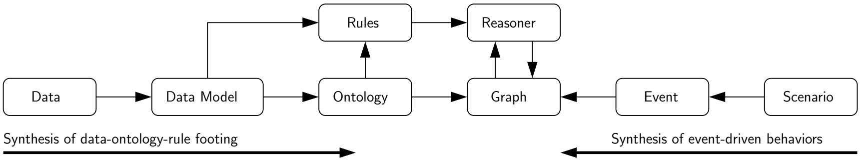
1. Parastoo Delgoshaei (2013-2017);
2. Maria Coelho (2015-present).

Data-Driven Approach

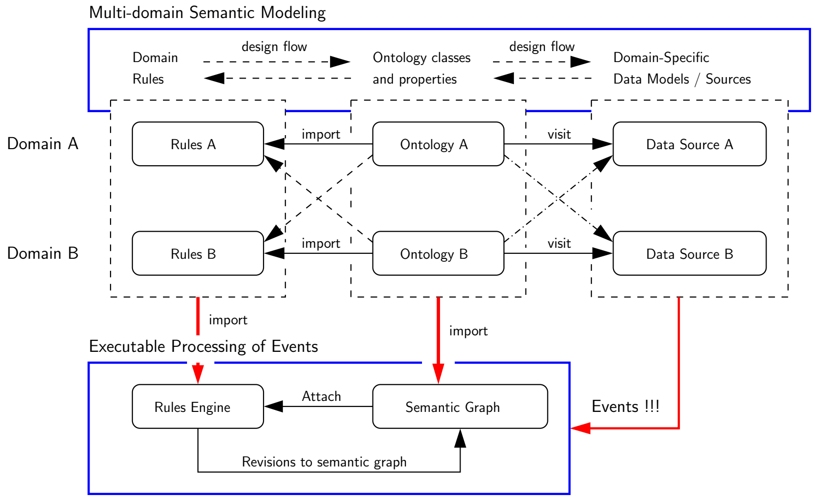
Guiding Principles:

1. One footing for ontologies, rules and data ...
2. Use (but do not extend) foundational level ontologies ...
3. Ontologies visit data models to get individuals ...
4. Semantic graph dynamically responds to incoming events ...
5. Enhance power of rules with backend functions ...

Preliminary Schematic:

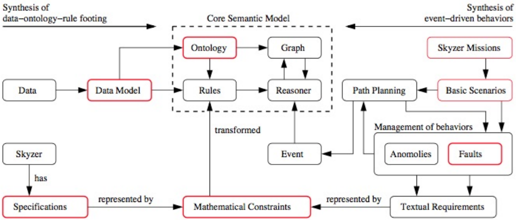


Template for Semantic Modeling + Processing of Events

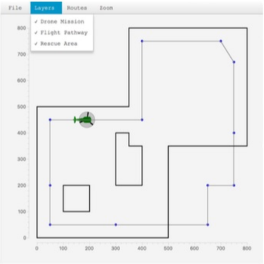


Data-Driven Approach (Synthesis of UAV Operations)

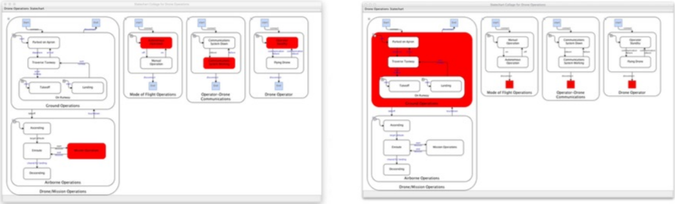
Synthesis of data-ontology-rule footing + event-driven behaviors.



Simulation in Whistle ...



Visualization of subsystem behaviors ....



Data-Driven Approach (Populating Models with Data)