Architecting Digital Twins for Model-Centric Engineering: Semantic and Machine Learning Approach

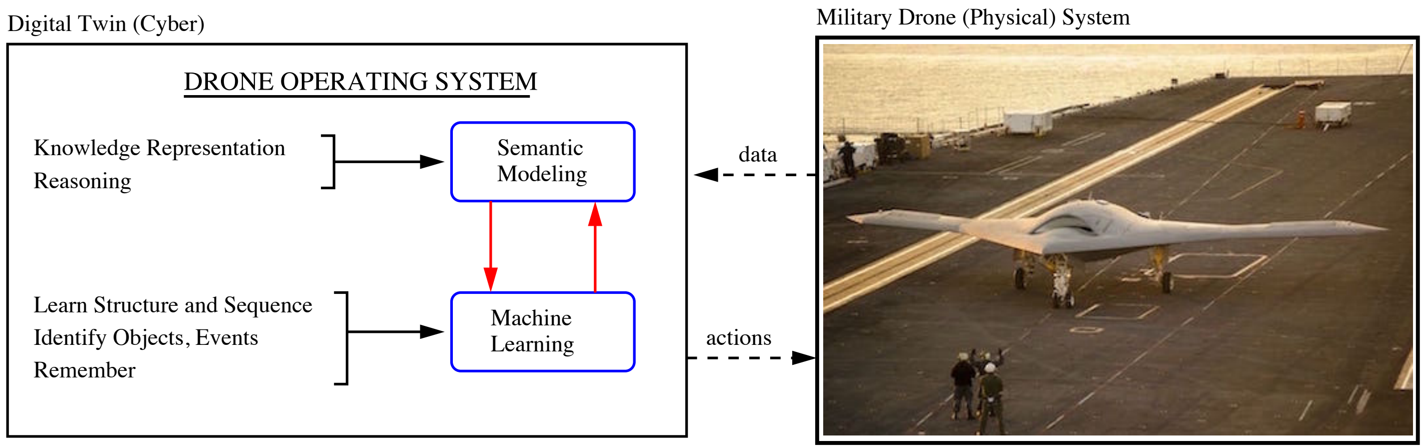
Sponsor: OUSD(R&E) | CCDC

By
Dr. Mark Austin (PI), Maria Coelho, Dr. Mark Blackburn
11th Annual SERC Sponsor Research Review
November 19, 2019
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

www.sercuarc.org

Incubator Project

Basic Idea: Explore design of digital twin architectures that support AI and ML formalisms working side-by-side as a team, providing complementary and supportive roles in collection of data, identification of events, and automated decision making.

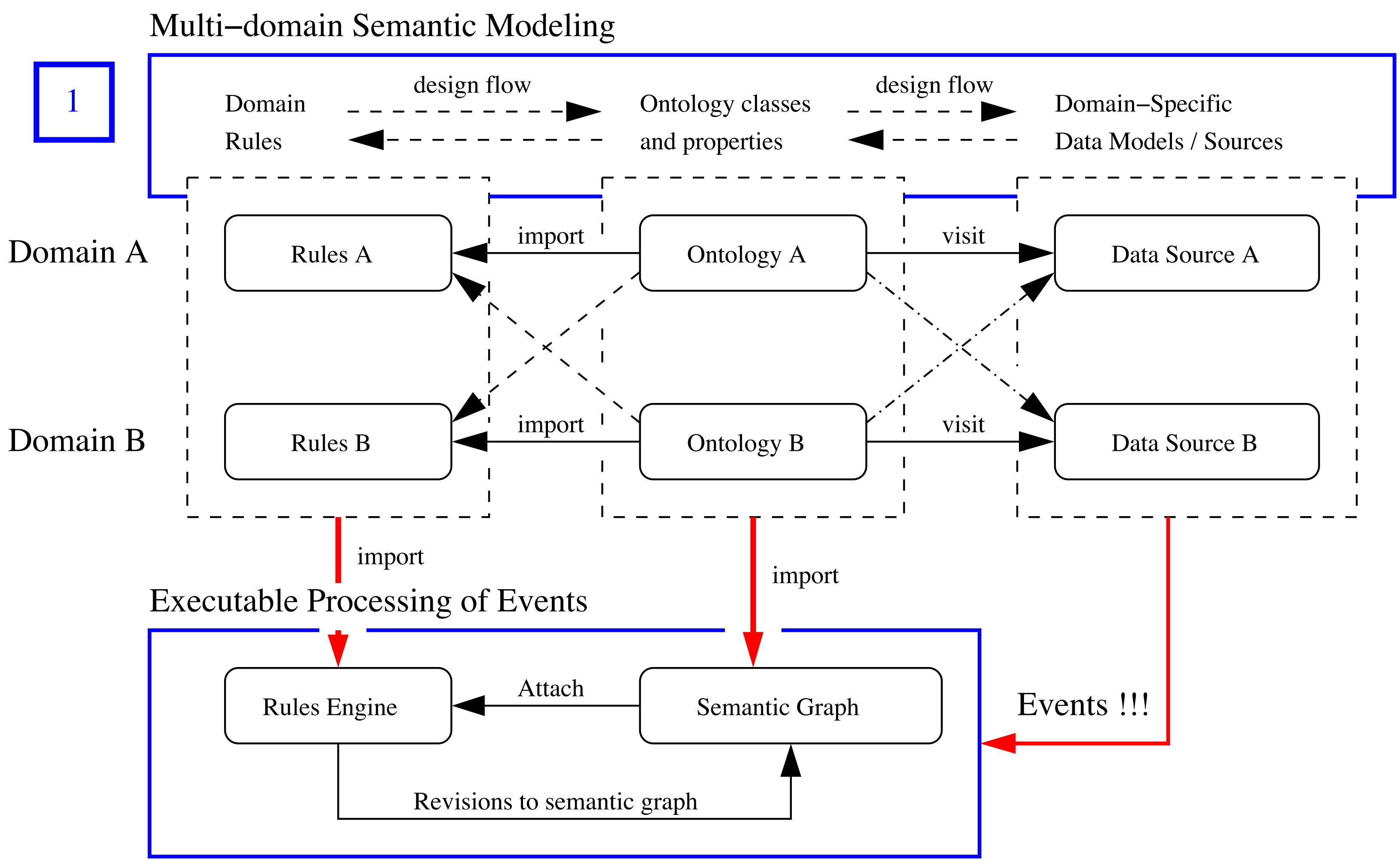


Research Challenge: How to design digital twin elements and their interactions so that collectively they can support a wide variety of systems engineering methods and processes?

Incubator Goals: Understand the range of possibilities for which machine learning of large-scale graphs and their attributes support activities in model-centric engineering.

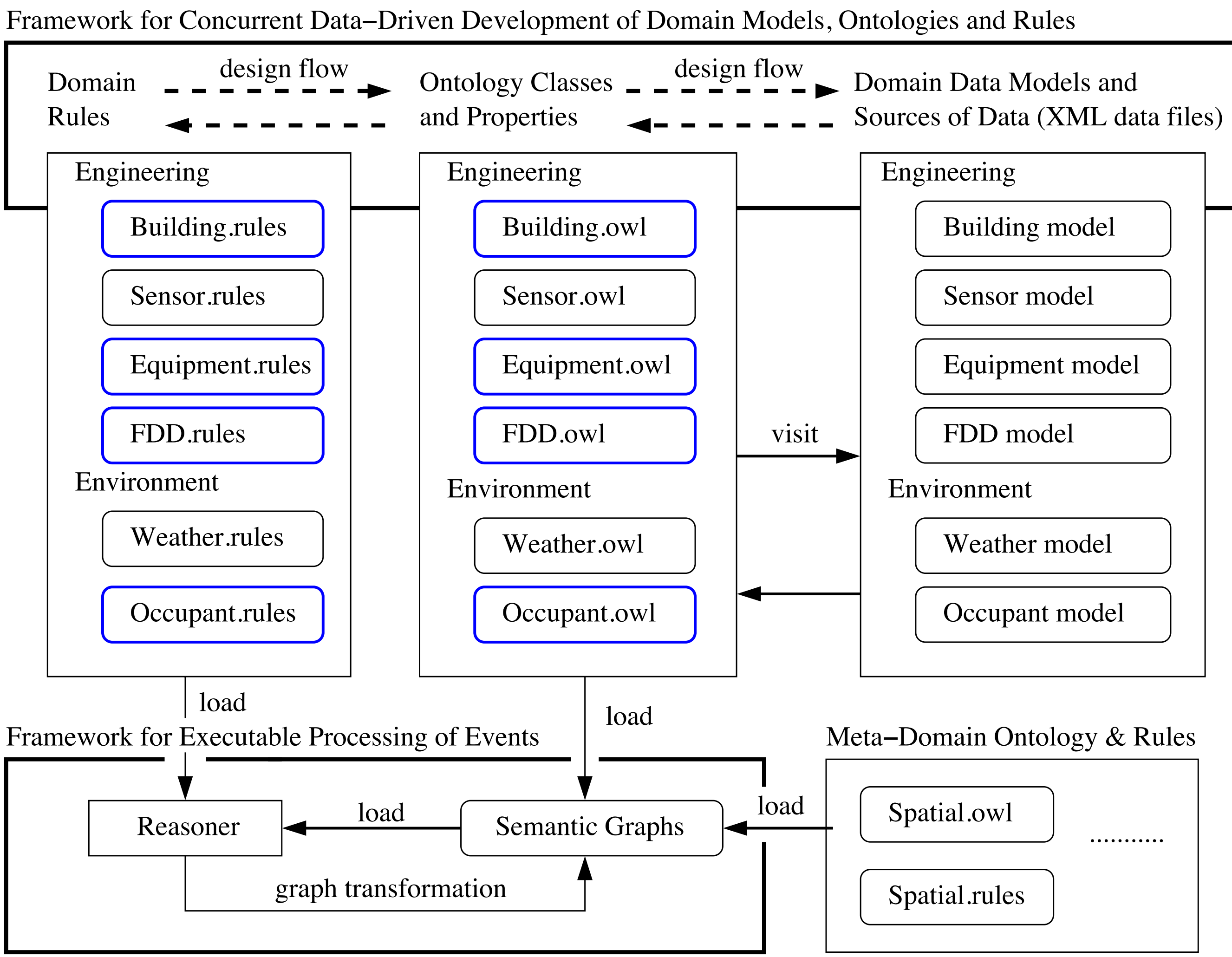
Architecture for Multi-Domain Semantic Modeling

Step 1: Data-Ontology-Rule Footing (Work at UMD / NIST / SERC in 2017).

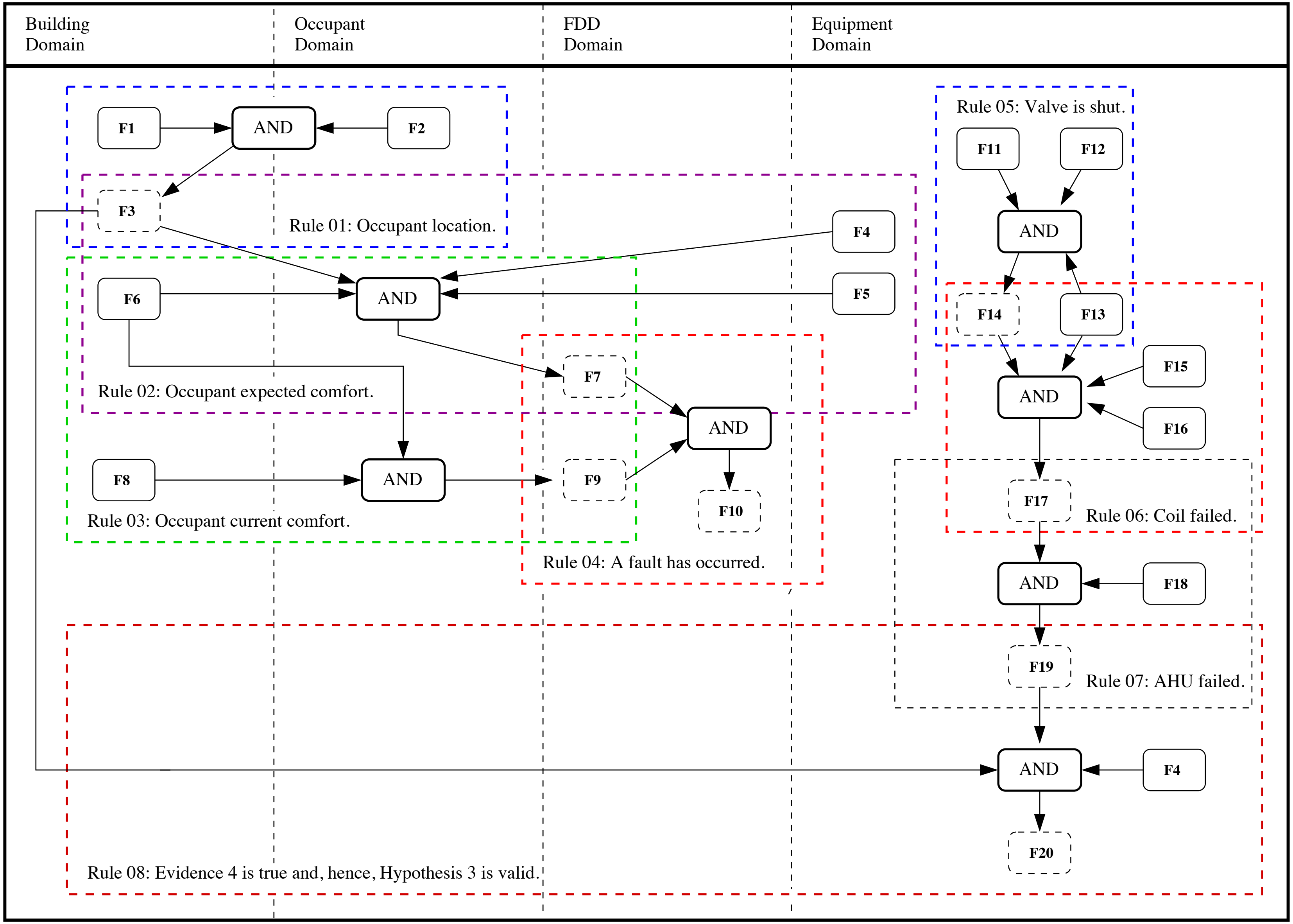


Multi-Domain Semantic Modeling

Example: Detection and Diagnostic Analysis of Faults in HVAC Equipment.

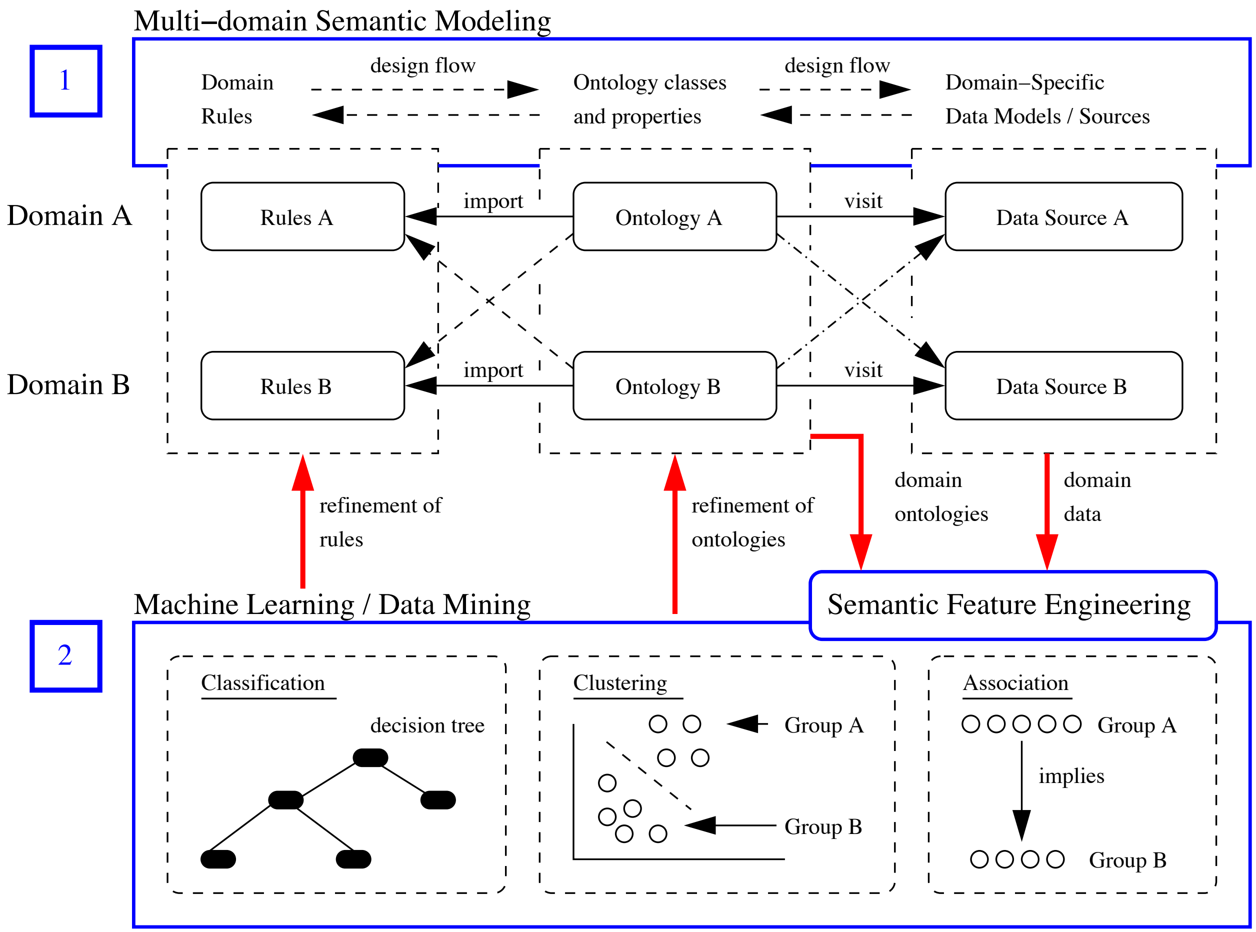


Multi-Domain Semantic Modeling



Combined Semantics + Data Mining

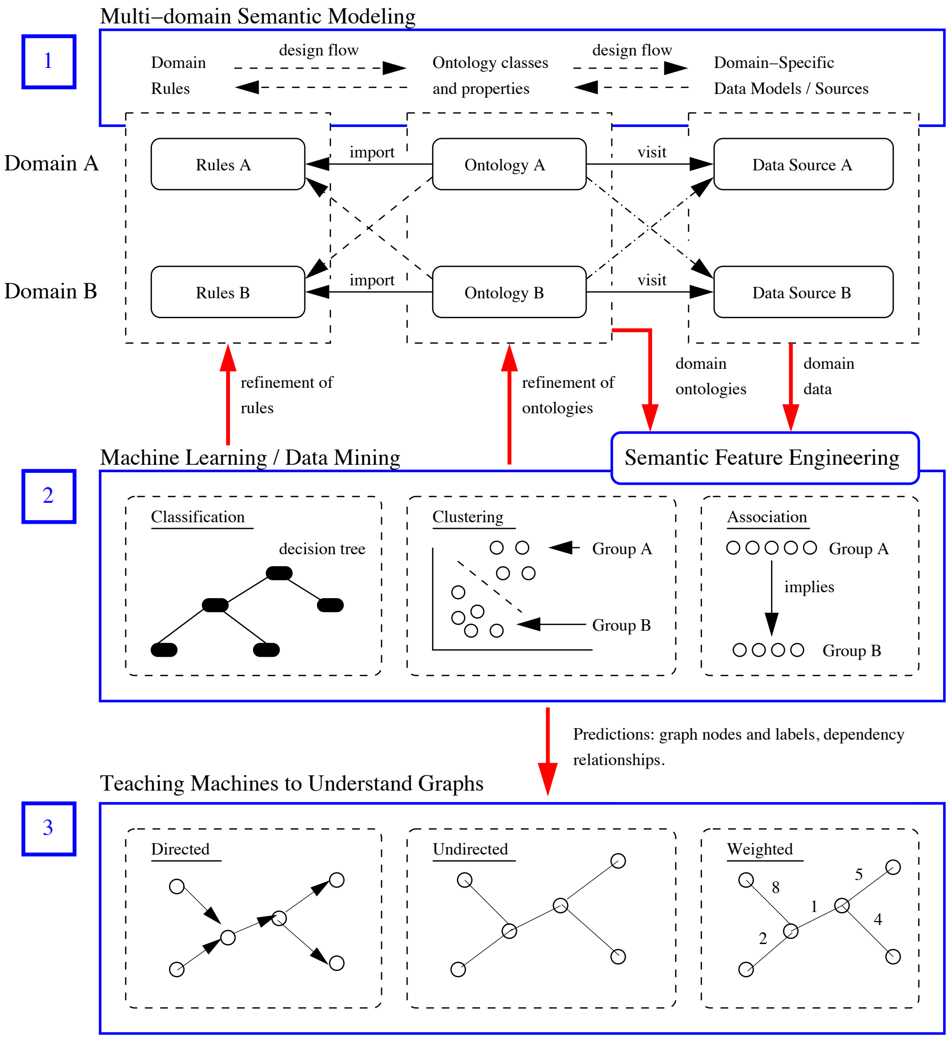
Step 2: Work at UMD / Building Energy Group at NIST / NCI, 2018-2019



Research Question: How can semantic modeling + machine learning / data mining work together as a team?

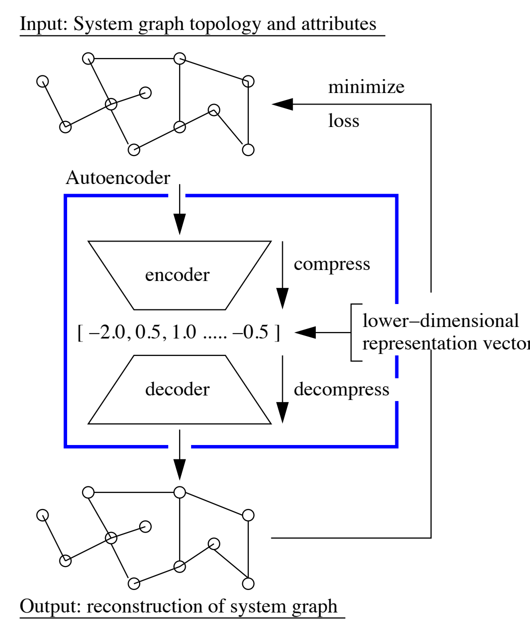
Incubator Project

Step 3: Focus on Machine Learning of Graphs and Model-Centric Engineering.



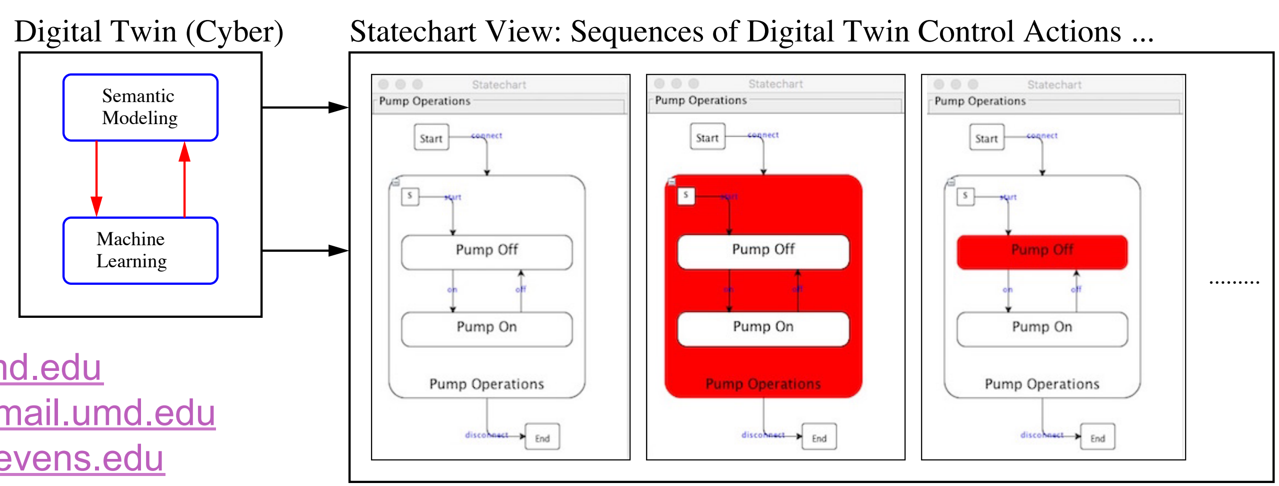
Observation: A lot of model-centric engineering boils down to representation of systems as graphs and sequences of graph transformations punctuated by decision making and work / actions.

Hence: Explore opportunities for teaching machines to understand graphs.



Incubator Research Questions

- What types of graphs (e.g., undirected, directed, weighted, multi-graph) are easy for the ML to learn?
- How well do these techniques work with graph topology and attributes that are dynamic?
- What can the ML do that is outside the capability of semantic modeling? And vice-versa?
- How can the ML improve the semantic modeling? And vice-versa?
- How to design the red arrows connecting layers 1, 2 and 3?
- How to represent and reason with uncertainties?
- How does the difficulty of these challenges increase with graph size?
- How to map AI-ML capability to state-of-the-art engineering views?



Contact Information

Mark Austin (PI): austin@isr.umd.edu
Maria Coelho: mecoelho@terpmail.umd.edu
Mark Blackburn: mblackbu@stevens.edu