

Brick Ontology

Semantic Tiger Team

2025-04-11



Dr. Gabe Fierro

<https://gtf.fyi>

Asst. Prof, Computer Science @ Colorado School of Mines

Joint Appointment @ National Renewable Energy Laboratory

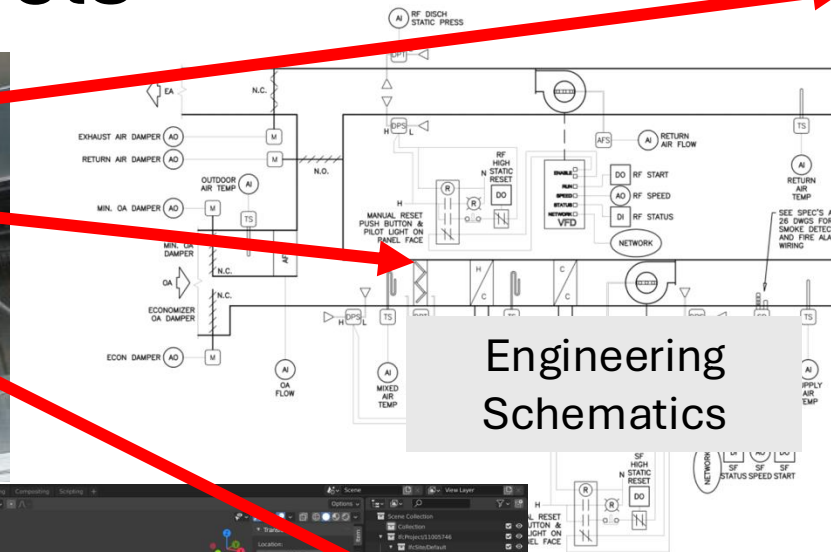
All views are my own and do not necessarily reflect institutional positions

Disconnected Digital Models

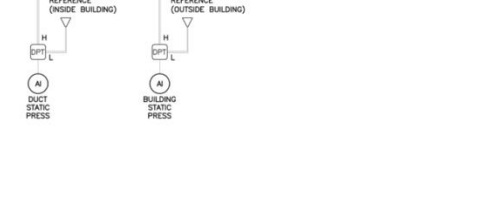
Field
Controllers / PLC



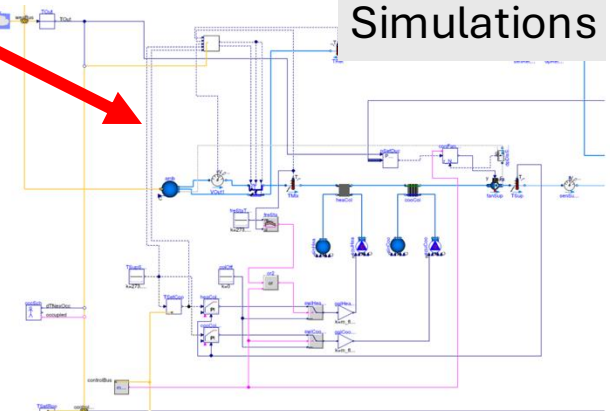
Physical system



Engineering Schematics



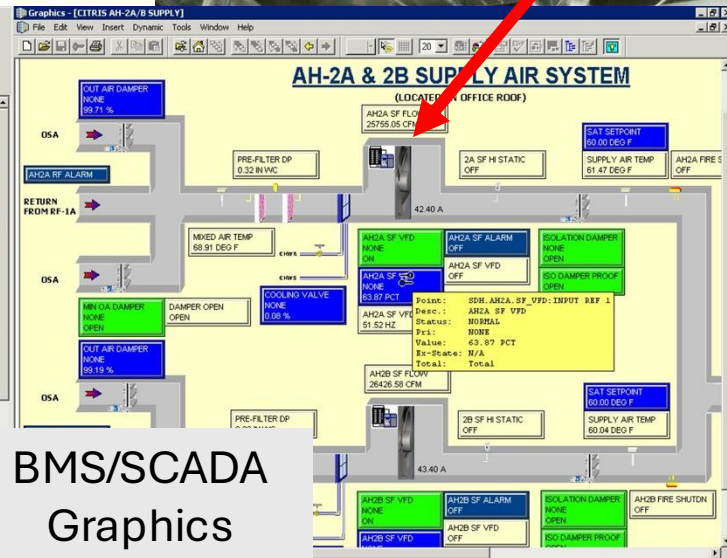
Simulations



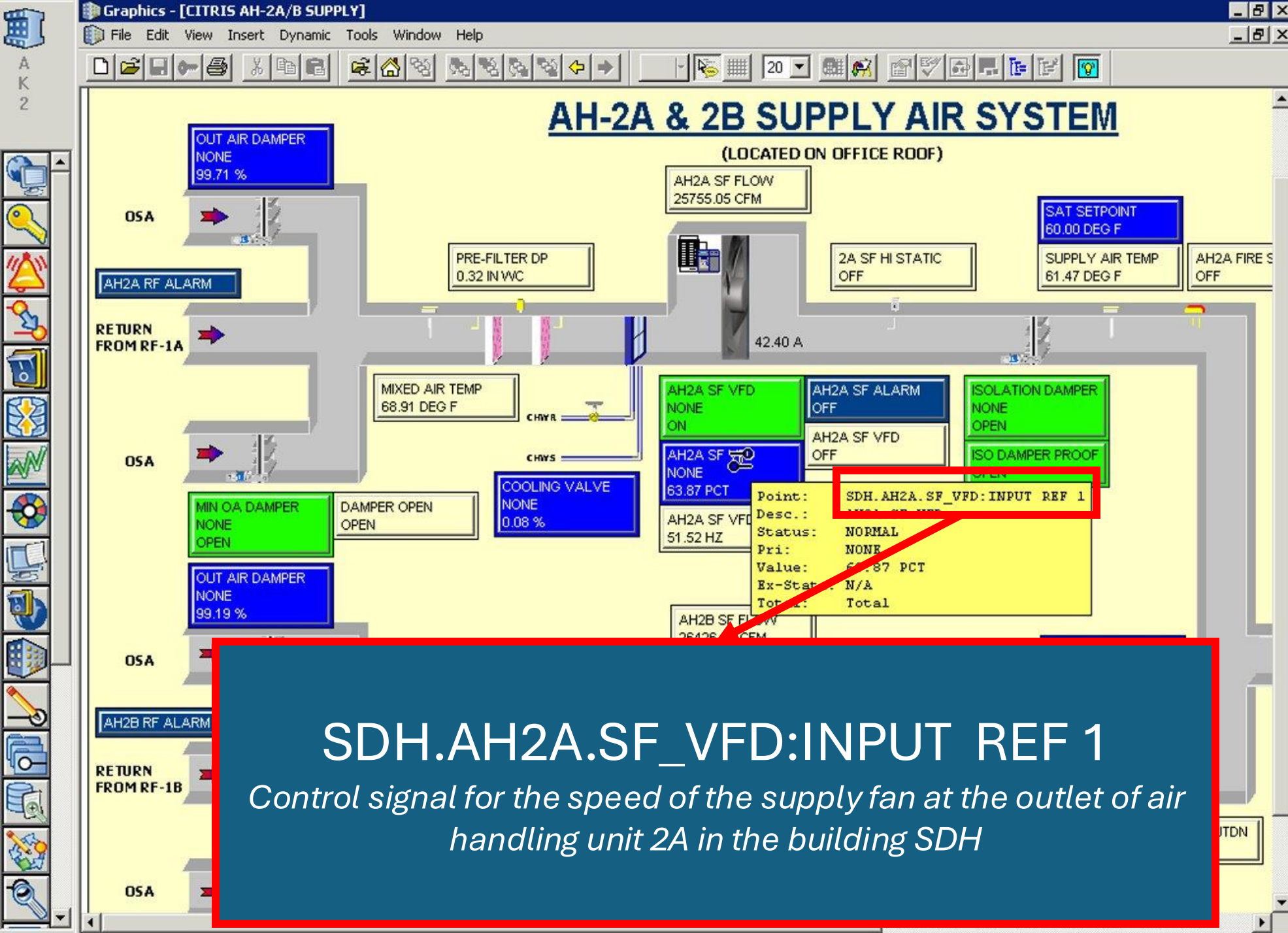
BIM/CAD Models



BMS/SCADA Graphics



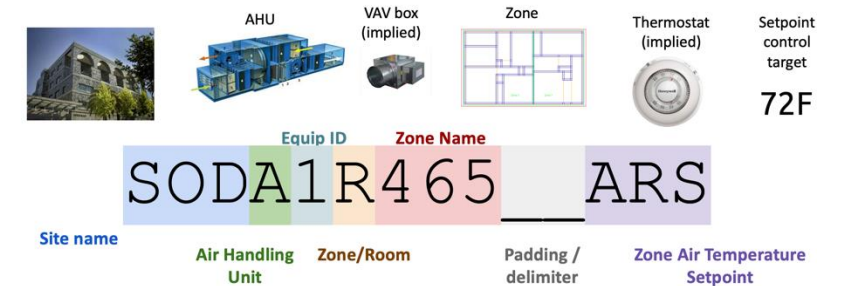
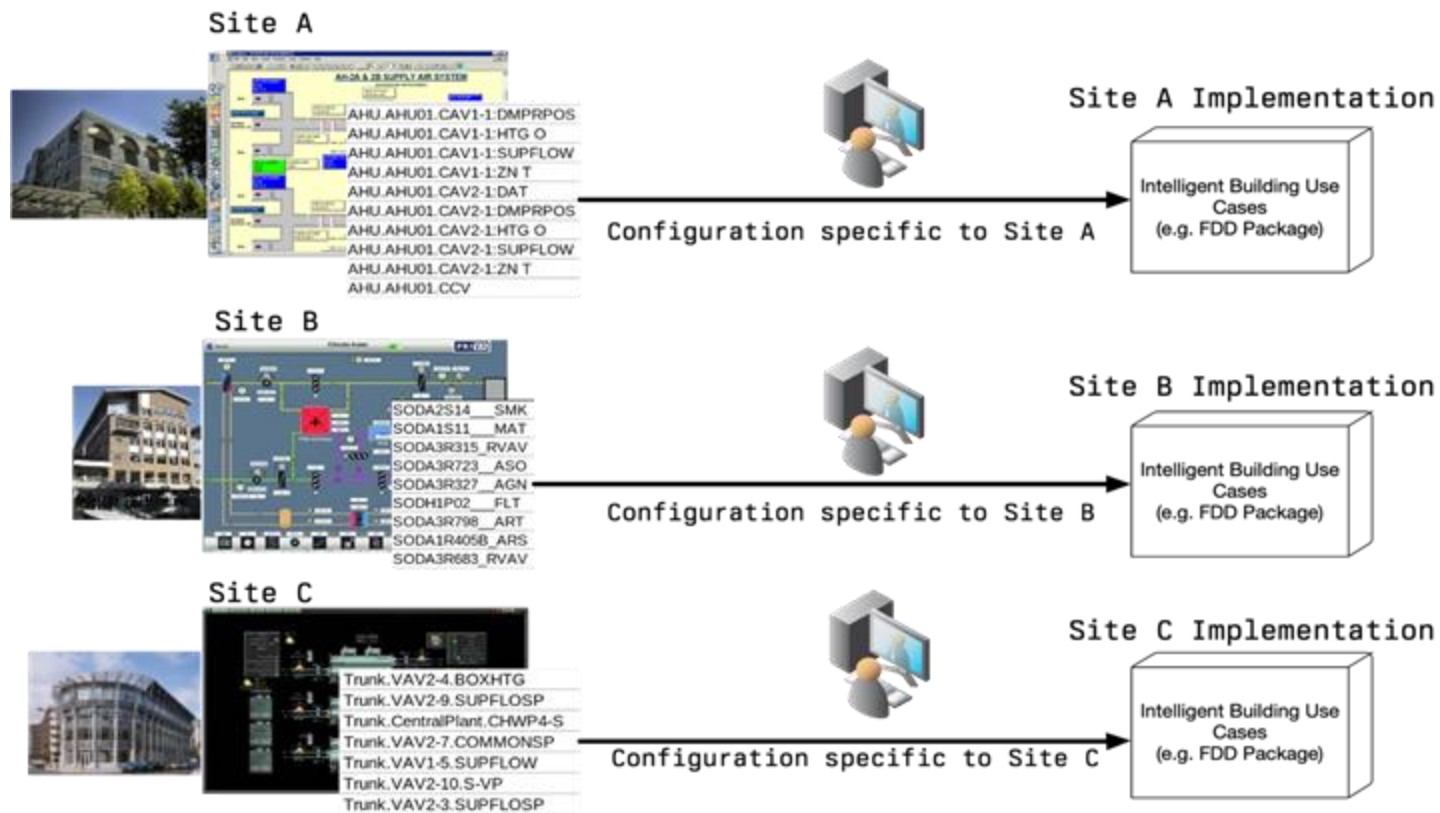
- Many useful building apps (FDD, controls) require data
- Finding data often requires cross-referencing between different representations of the same system



SCADA Tags and BMS Point Labels

- Availability and structure of information varies between buildings
- **Example:** BMS/SCADA tags for I/O points

Ending Bespoke Development



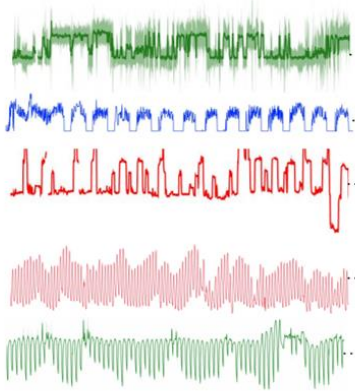
Naming conventions and protocol “soup”



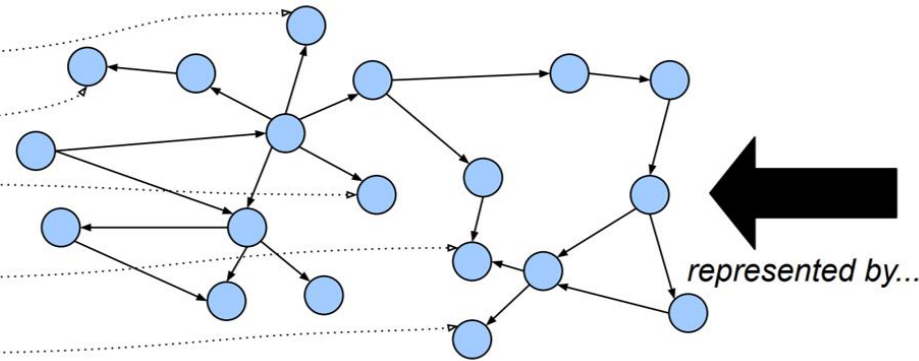
Lack of standardization, interoperability increases soft costs associated with developing and deploying data-driven solutions

Brick: “Data Twin” for Buildings

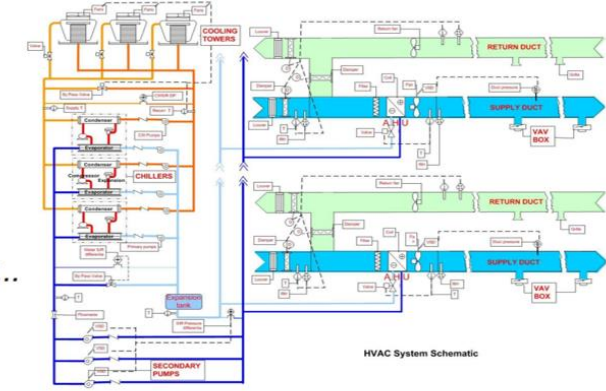
Live/Historical Facility Data



Metadata Graph Abstraction



Building Subsystems + Data Collection



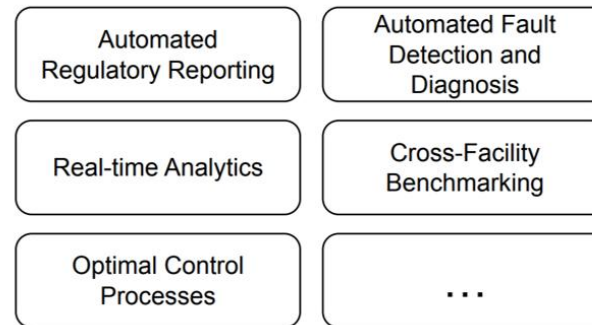
Founded in 2016

Collaborators: NIST, NREL, PNNL, LBNL, UC San Diego, UC Berkeley, Carnegie Mellon

Consortium Members: Siemens, Schneider Electric, JCI, Carrier, Mapped, Clockworks Analytics, + academic collaborators above

Backed by 501(c)(6) consortium

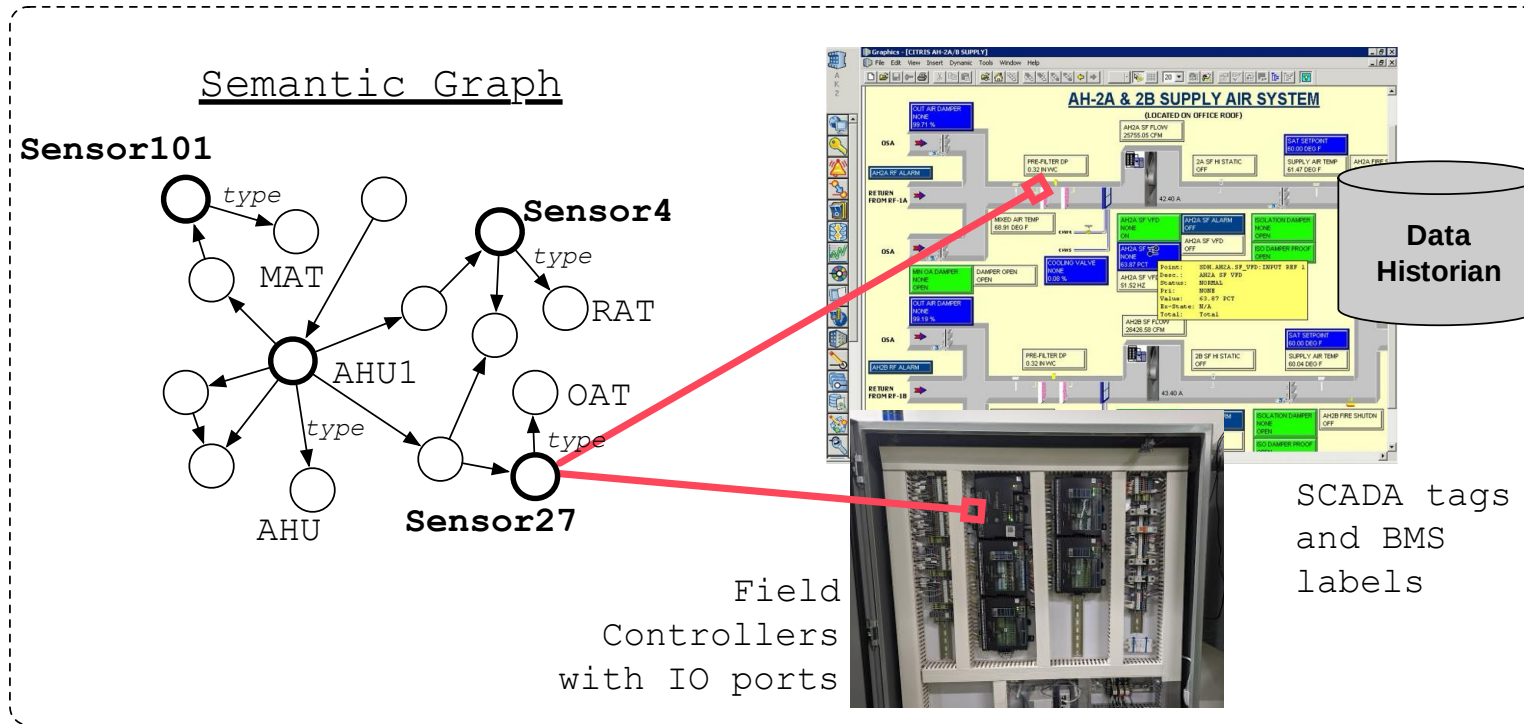
enables...



Portable Data-Driven Use Cases

- Over 1000 different equipment, point, location (REC) classes
- Organized into an extensible hierarchy
- Static “entity properties” for nameplate characteristics
- Maps to 223P, RealEstateCore, Haystack
- Links to other representations
 - BMS, BIM, Simulation, Timeseries
- Built on RDF, SHACL

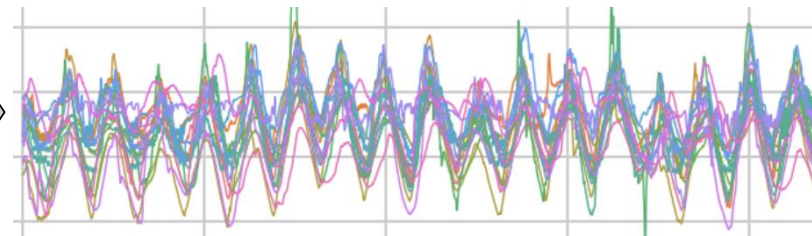
Query Brick Model to Configure Applications



- Brick model contains **points, equipment/assets**
- Use RealEstateCore for architectural elements
- Late binding between application data sources and the specific I/O points in buildings

AHU	Air Handling Unit (Equipment)
MAT	Mixed Air Temperature (Sensor)
RAT	Return Air Temperature (Sensor)
OAT	Outside Air Temperature (Sensor)

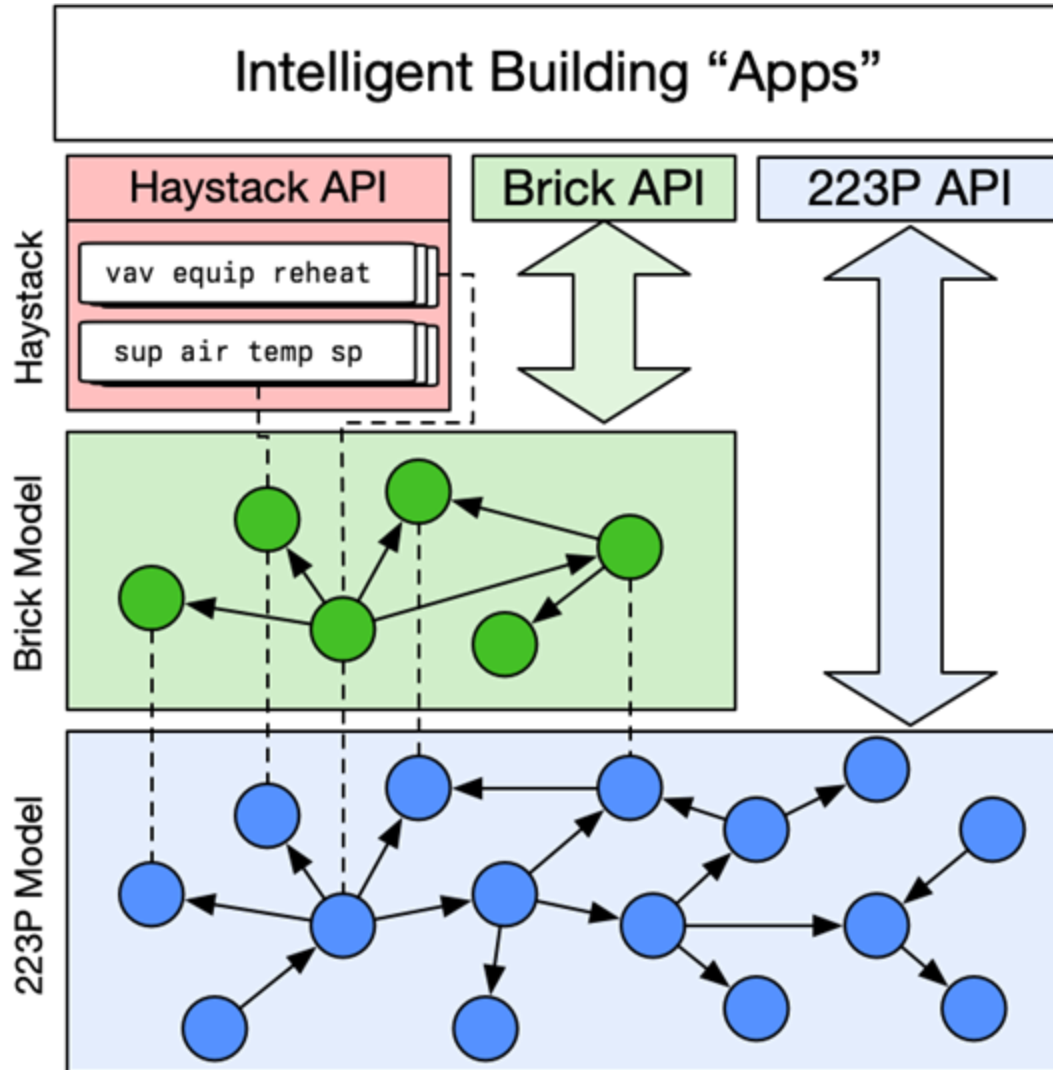
Graph-driven
Specialization



When $\{MAT\} < \min(\{RAT\}, \{OAT\})$
on an $\{AHU\}$ then raise **error**

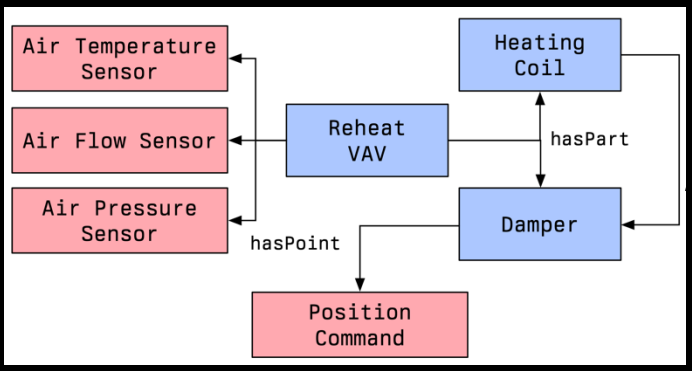
When $\{Sensor101\} < \min(\{Sensor4\}, \{Sensor27\})$
on $\{AHU1\}$ then raise **error**

Supporting Building Apps with a Semantic “Stack”

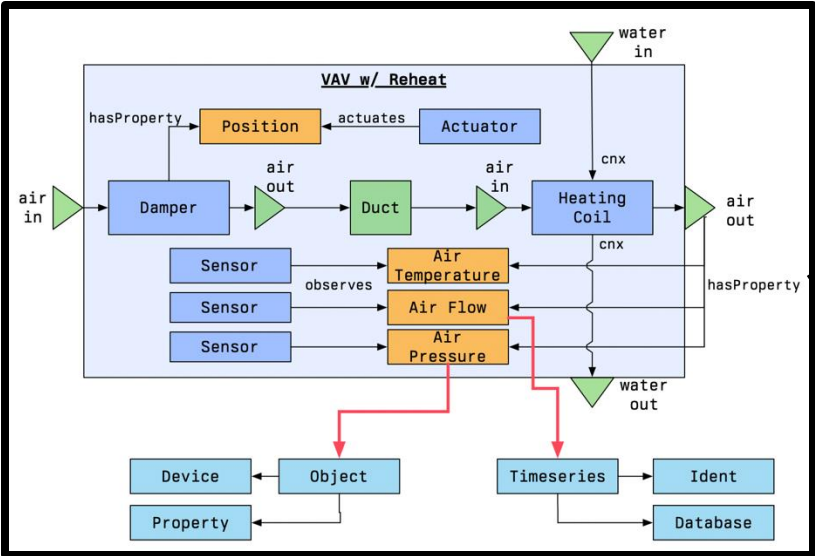


- “Best” model is relative to the applications’ needs
- Go “up” the stack
 - More abstracted
 - Easier queries, but less precision
- Go “down” the stack
 - More detail, more formal
 - Queries can be more precise, but may be harder to write
- Active research to automatically derive Haystack, Brick from 223P

Supporting Building Apps with a Semantic “Stack”



Brick model: app-focused types



223P model: detailed topology

Skyspark, Niagara, OSISoft, Johnson Controls, AutomatedLogic, others...

Application Platforms

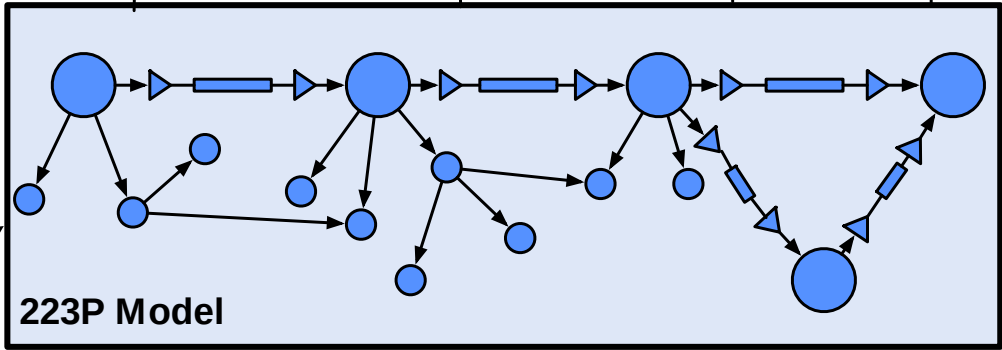
Project Haystack

Brick Schema

REC, BOT, others

“Native” 223P

Vendor / Consortium standards



Building Network Scan

OEM Cutsheets

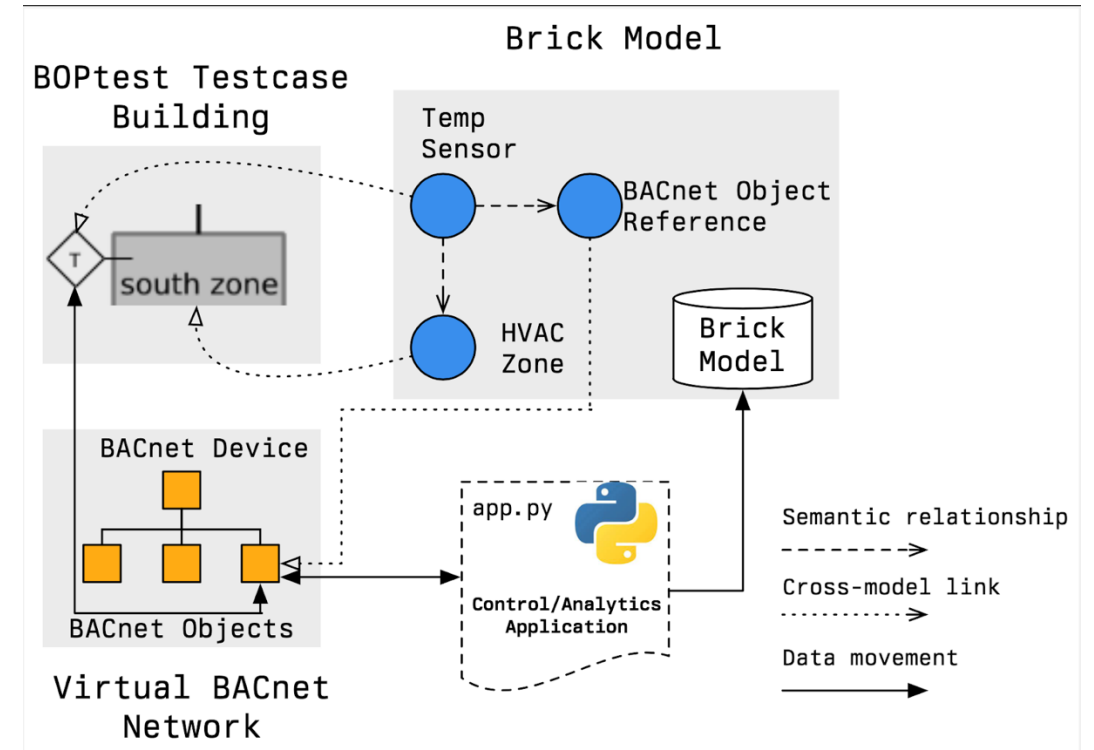
Equip, Point Schedules

BuildingMOTIF, other tools

Metadata Sources

Connecting Semantic Metadata with Simulations

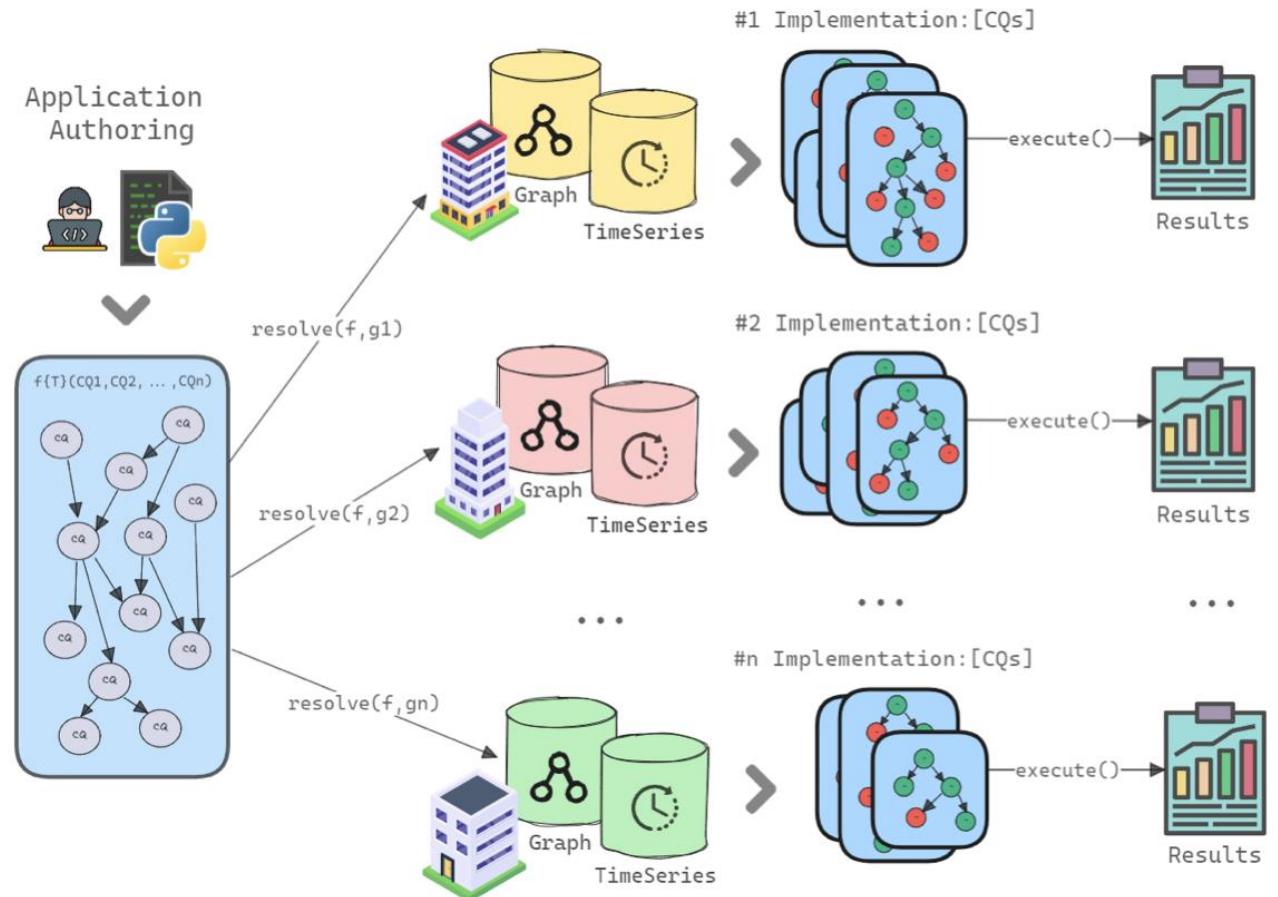
- **Challenge:** control testbed does not facilitate deployment of these algorithms in actual buildings
- **Solution:** layer virtual building network over the I/O points of simulation
- **Use Brick** (semantic metadata) to provide context over the simulation
- End result is an implementation-agnostic representation of building with realistic behavior



SeeQ: New Programming Model for Building Analytics

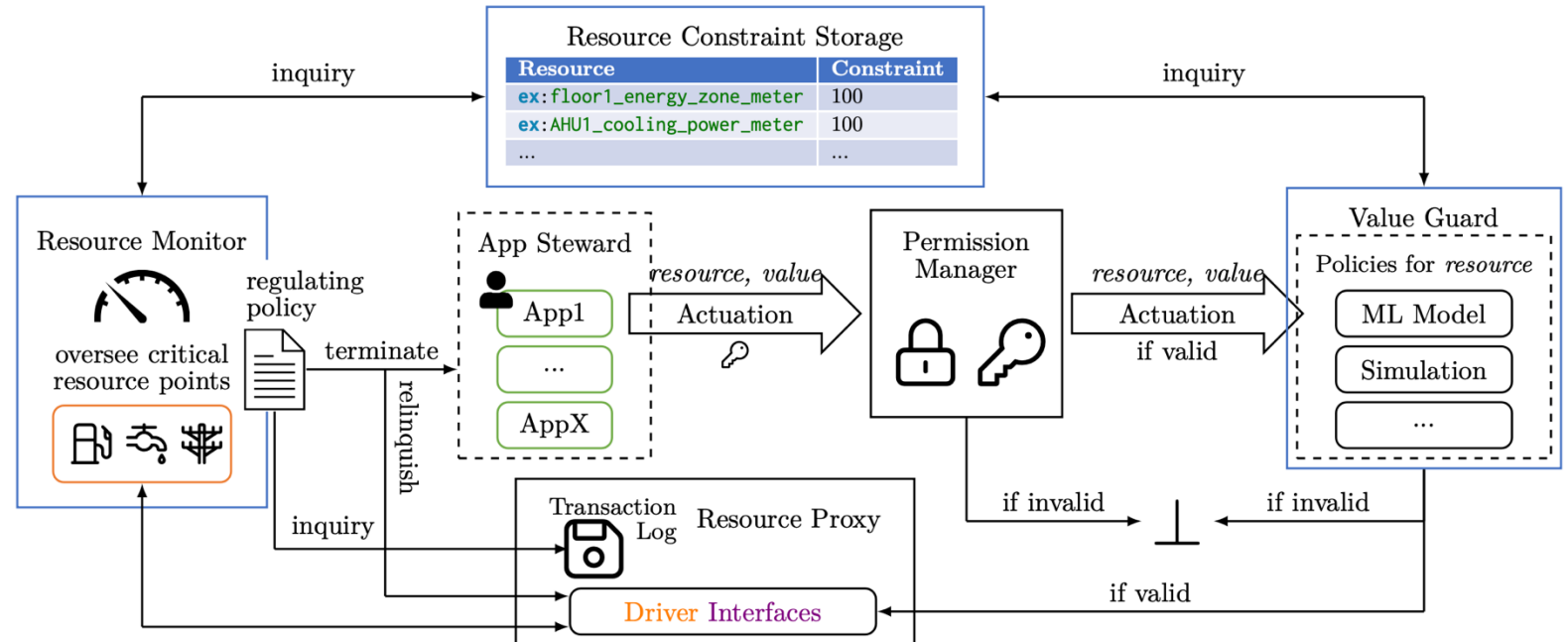
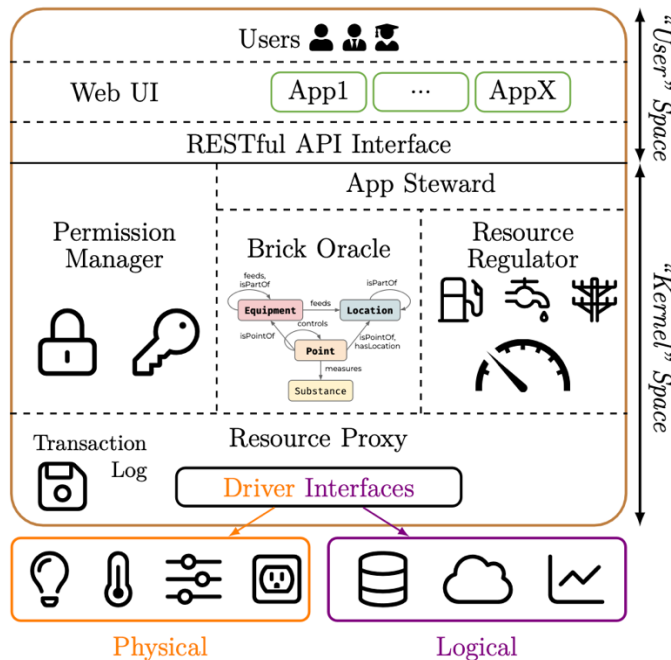
- Write Python applications against concepts defined by metadata ontology
- SeeQ “compiles” the Python code against the metadata model for each building
 - Generates building-specific impl.
- Demonstrated on FDD rules
- Step towards fully **portable** applications

```
1 from SeeQ import *
2 from pandas import DataFrame
3 from G36.CQs import Dmp_Pos, Fsa, Fsp_clg, Fan_s
4 from APAR.CQs import Tsa, Tma, DelTsf, Hc_pos, Epsilon_t
5
6 def APAR_R1(sup: Tsa, mix: Tma, drop: DelTsf, heat_coil: Hc_pos, e: Epsilon_t):
7     is_heating: DataFrame = heating_coil.df > 0
8     supply_air_low: DataFrame = sup.df < (mix.df + drop.df - error.df)
9     violating_records = is_heating & supply_air_low
10    # returns fault if more than 10 violating samples
11    if len(violating_records) > 10:
12        return "fault detected"
13
14 def G36_Dmp_Leaking(pos: Dmp_Pos, sup_flow: Fsa, cool_sp: Fsp_clg, fan: Fan_s):
15     if ((pos.df == 0) and (sup_flow.df > max([0.1*cool_sp.df, 50]) \
16         and (fan.df == "ON"))).for_time(600):
17         return "Level 4 alarm"
```

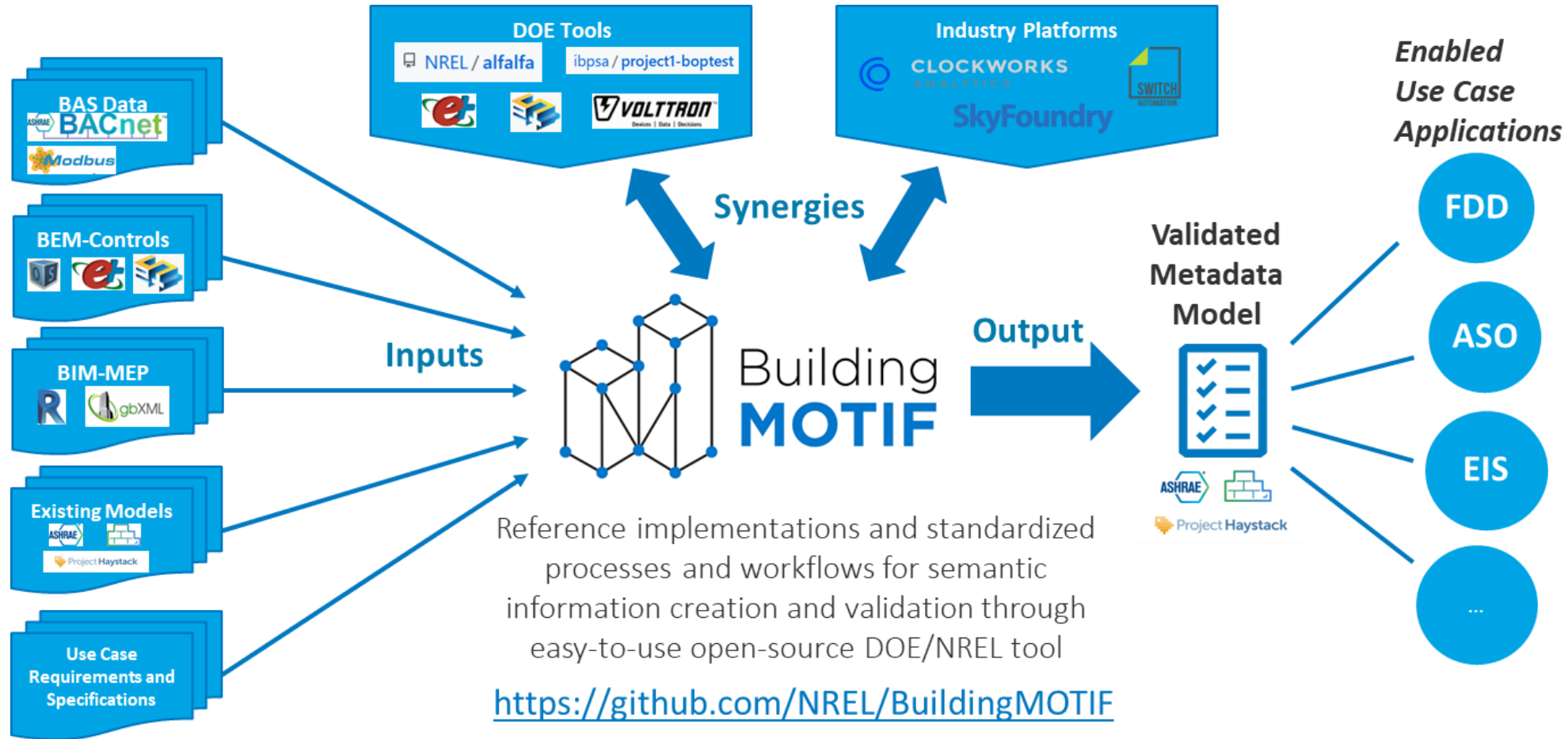


Playground: Untrusted Building Applications

- Research OS for executing untrusted building applications in a multi-tenant environment
 - Uses Brick models to grant access to resources using **principle of least privilege**
 - Provides **resource isolation** on critical building resources (total energy, peak power, etc)

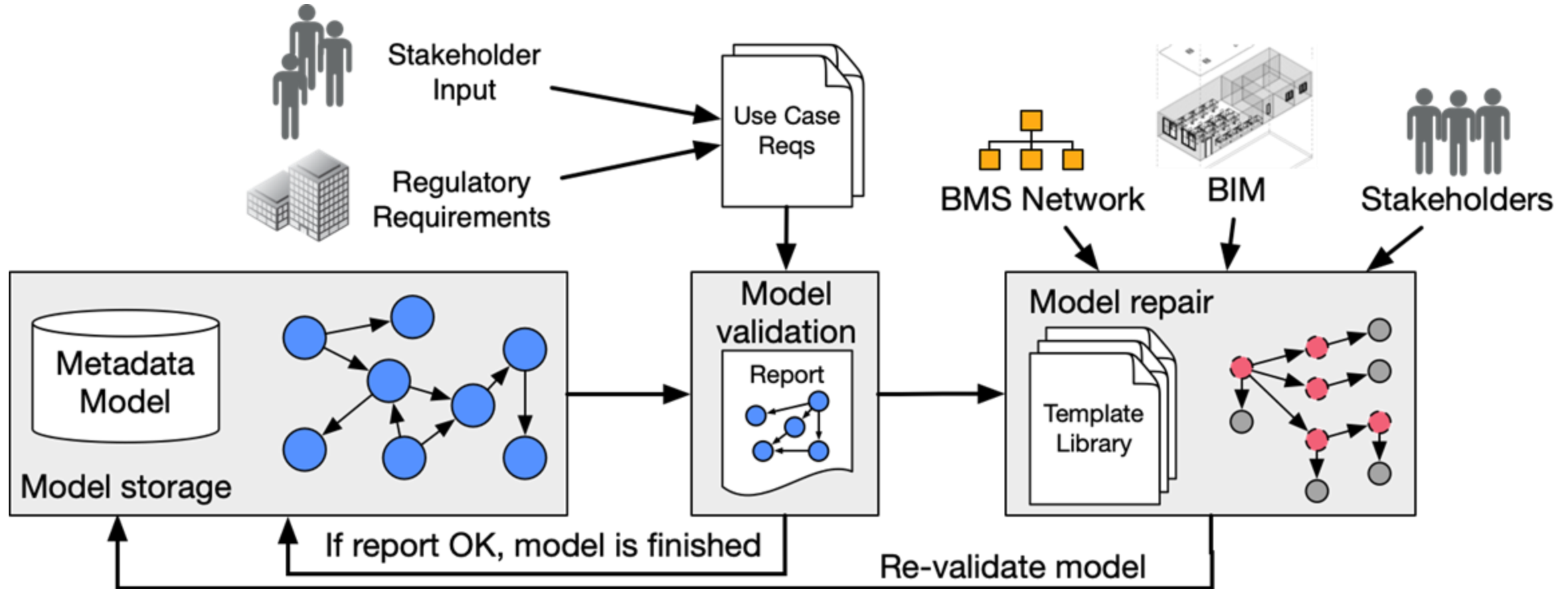


BuildingMOTIF: SDK to Support Semantic Metadata



- US Dept of Energy Building Technologies Office project, NREL led
- Use semantic metadata as “lingua franca” connecting existing tools for simulation, modeling, controls, AFDD, BIM, M&V, data science
- Libraries of templates and shapes for common systems and applications

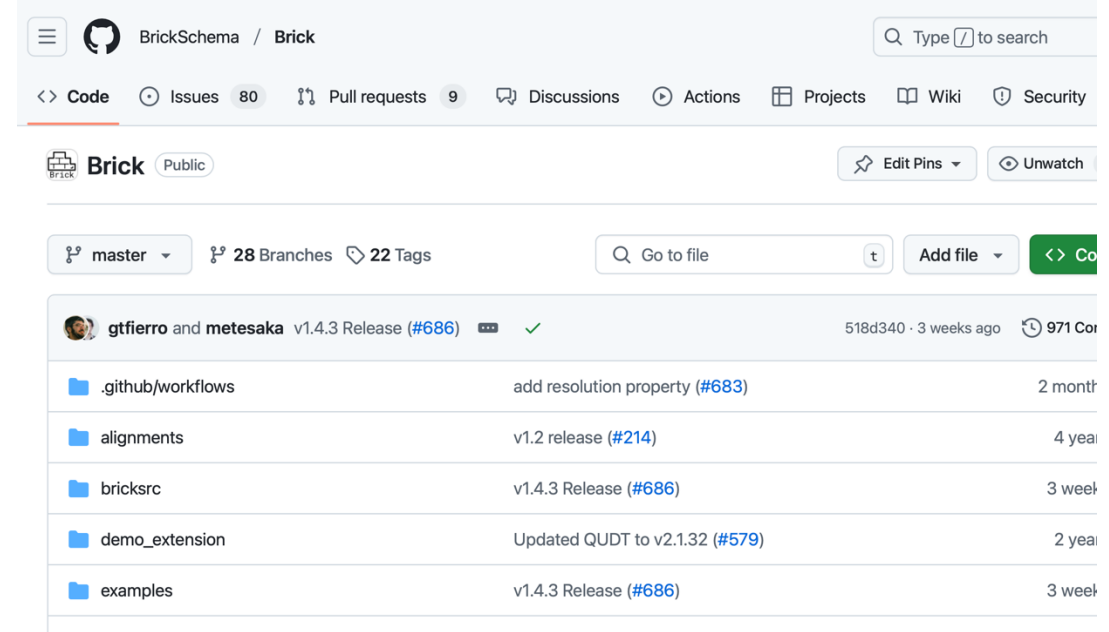
BuildingMOTIF: Create and Validate Metadata



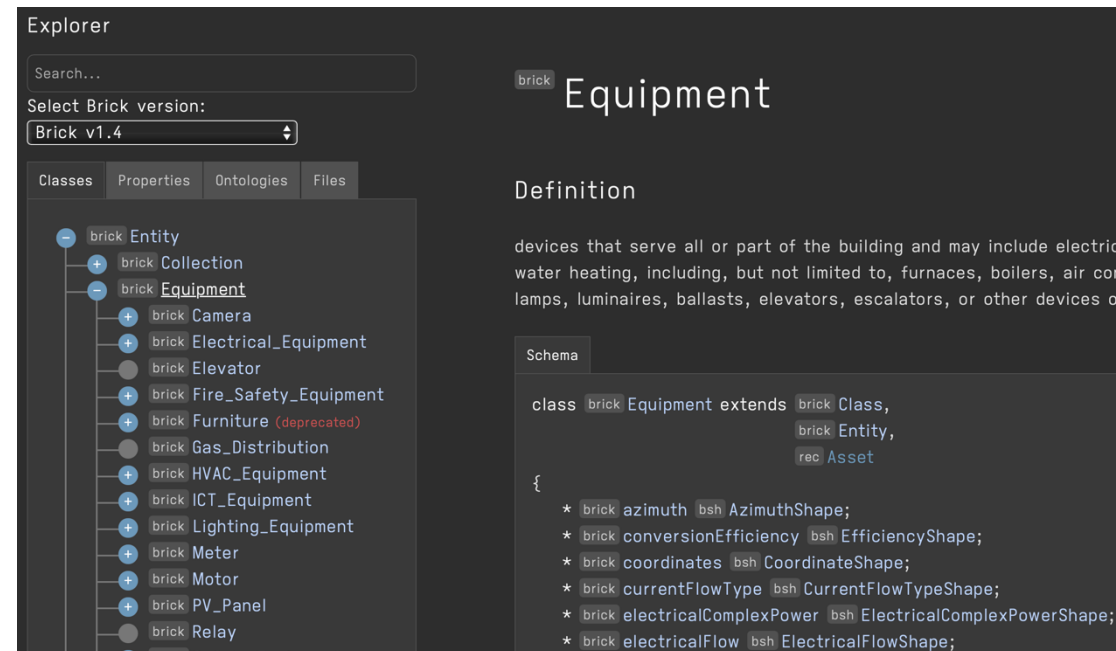
- Incorporate formal use case requirements into iterative workflow
- Ensure that delivered metadata model fulfills all use cases
- Automate / simplify authoring of models through templates, imports from other sources
- **Current work:** provide economic transparency on ROI for smart analytics

Resources

- Brick ontology:
 - <https://brickschema.org>
 - <http://github.com/BrickSchema/Brick>
- BuildingMOTIF SDK
 - <https://nrel.github.io/BuildingMOTIF>
 - <https://github.com/NREL/BuildingMOTIF>
- Papers, code, talks, tools:
 - <https://gtf.fyi>
- Everything is open-source under BSD 3-clause of similarly-permissible license



github.com/BrickSchema/Brick



Shape Docs: ontology.brickschema.org