

Office of ENERGY EFFICIENCY & RENEWABLE ENERGY

# The Killer (Meta) App for Building Controls and Analytics

Amir Roth, PhD

US Department of Energy, Building Technologies Office

Haystack Connect 2023 June 6, 2023





## "Hello World Haystack Connect!"



#### **Amir Roth**

- Technology (R&D) Manager at DOE's Building Technologies Office
- Core program is Building Energy Modeling
- Also work in Controls program, mostly on standards and "platforms"
- Disclaimer: no training or professional experience in modeling or controls!

### **BTO's Building Controls Program**

- Some analytics/FDD
- Focused on (HVAC, some lighting) controls  $\leftarrow$  explains orientation
- Emphasis on grid-interaction and DER-integration
- "Emphasis" on small/medium buildings and homes (i.e., no BMS)
- Recently looking at neighborhood/district scale
- VOLTTRON<sup>™</sup> IoT platform (Python/MQTT)

### https://energy.gov/eere/buildings/



## **Supervisory HVAC Control Is Critical**

#### **18% of commercial buildings have BAS**

• 45% of floor area, about same fraction of energy use

#### **Control upgrades are a "no regrets" near-term play**

- High-performance
  - ASHRAE G36 reduces HVAC energy 29%, demand 21%
- Low embodied carbon, minimally disruptive to occupants
- Maximize heat-pumps, heat-recovery, storage
- Implement grid response and interactivity
- Enable "transitional" (hybrid) systems
- Retain value through HVAC upgrades

#### Applies to non-BAS as well



90.1 G36 FDD MPC ML

#### "Control delivers the value of the system to the customer" – Clas Jacobson, UTC

## Key to Scaling Up Control Upgrades is ... Semantic Models

### The artist formerly (currently?) known as "semantic metadata"

#### What makes metadata "semantic"?

• Explicitly defines meanings of data elements, values, and relationships

#### And what makes it "Semantic"?

- Semantic web technologies
- RDF (Resource Description Framework) schema for modeling graphs (i.e., nodes, edges, relationships)
  - Why graphs? HVAC systems are graphs (i.e., equipment, pipes, ducts, loops)
  - Several formats including XML and TTL (more compact)
- SPARQL (Standard Protocol and RDF Query Language) SQL or XQuery for RDF
- SHACL (Shape Constraint Language) XSD for RDF



## Why Is SPARQL So Important?

#### autoconf: configure build/install for different unix flavors (Solaris, HPUX, AIX, etc.)

- CMake: better autoconf with a worse name!
- Homebrew: game + name!





#### autoconf/CMake/homebrew are possible because

• Even if a file system is organized differently ... the tools to navigate it are the same, i.e., there is a "SPARQL"

## The Dream: "autoconf" for Building Automation Software

#### **Download software for a generic building/system**

- E.g., ASHRAE Guideline 36 "high performance sequences of operations" for multi-zone VAV air-handler
- Implemented for generic number of zones and terminal units

#### Run "autoconf" to configure for specific building/system

- Use SPARQL to systematically and iteratively query the building's Semantic model
- Instantiate for actual number of zones, with actual thermostat, air-handler, and terminal-box points
- (Generic) control software will include SPARQL queries for instantiation

Killer (Meta) App: ability to automatically configure and install apps "Plug-and-play" in the familiar/expected consumer electronics/software sense Inter-operability for installation/update, not for runtime

## What About SHACL?

#### Defines well-formed-ness and completeness criteria for Semantic models

- Completeness is defined relative to a set of applications or use cases
  - "Does the model contain the metadata to support this application? Is it unambiguous? Consistent?"
- Useful for procurement, i.e., "bring me a Semantic model! No, not that one!"



### What About "Analytics"? What About Haystack?

#### Analytics is simpler than control from a data/metadata standpoint

- In general, more "incremental" and "forgiving"
  - Can provide meaningful benefit with only a few data streams
  - Not a complete loss if some metadata is missing, wrong, or ambiguous
- Control is much more finicky, need more metadata to get started and tolerance for errors is lower

#### Haystack is semantic-ish (i.e., has semantic tags, can be queried ...

- Tag groups are human interpreted, proper use is not enforced by the language or system
- Query language does not "reason" about graphs, e.g., does not identify connections across multiple hops
- Obviously sufficient for analytics, what about for controls?

#### .. but not Semantic

#### **Could semantic modeling amplify Haystack's impact? What about Semantic modeling?**

### Where Are We?

#### Want autoconf/CMake for building automation software

- Need semantic model, query capability like SPARQL, and validation tool like SHACL
- Would like to get these capabilities connected with Haystack

#### There are the Brick and ASHRAE 223P Semantic models ... why are there two of them?

- Can we exploit them to give Haystack users access to SPARQL/SHACL and their capabilities?
- To make Haystack fully semantic even if it is not capital-S Semantic?
- What is the (desired) relationship between Haystack and (these) Semantic models?



### ASHRAE 223P

#### A "reference/foundational" standard supporting many use cases

- Aims for non-ambiguity, completeness (yes, I know), and flexibility
- Flat, but allows iterative addition of level of detail (as needed) without deleting existing objects
- Quite verbose!
- Semantic modeling "assembly language" (maybe C)



### **Brick**

#### An abstraction layer for common use cases

- Aims for productivity in application development
- Hides verbosity of 223P behind a class/type hierarchy
- Brick predates 223P so this "mapping" is retroactive
- Semantic modeling "C++"



## Haystack

### This makes Haystack (HaySTACK?) the Semantic modeling ...

- Jupyter Notebooks?
- chatGPT?
- "HaySTACK": software stack around data model

#### What really matters to the Haystack community?

- Validation/procurement functionality for analytics applications
- "autoconf" functionality for analytics applications?
- Expansion to controls applications??

### Where does semantic/Semantic modeling really help?

- Validation!
- Configuration!!
- Application development?
- Scripting during operations??



## What Are Some Options for Haystack?

#1: build out "native" SPARQL and SHACL

**#2: create RDF version** 

#3: map/translate to 223P

#4: map/translate to Brick

### #1 → #3

- Less duplicate effort
- Fewer (arbitrary) differences in schemas
- Easier expansion into controls

### What about #4?

- Easiest path? Brick is closest to Haystack
- Smallest marginal benefit relative to previous option?

### Multiple options – not exclusive!



## **BTO's Various Semantic Modeling Projects**

#### "Skewering the Silos", UC-Berkeley

- FY19-22 ← 2018 BENEFIT (competitive R&D solicitation) award
- Continued development of Brick schema and demonstrated applications

### "Develop and Disseminate Semantic Interoperability Standard", NREL, LBNL, PNNL, NIST

- FY23-25 AOP ← FY20-22 AOP
- Developing ASHRAE Standard 233P
- BuildingMOTIF translation tool ← demo this PM!
- Some demonstrations



#### "Meta-Data for the Masses", Project Haystack

- FY22-25 ← 2021 BENEFIT award
- You'll see ...





## **BTO's Modeling and Controls Standards & Tools Ecosystem**



## **ConStrain: Code-Compliance for Building Operation**



- Checks BAS data streams for compliance with control provisions of code (e.g., 90.1)
- Application to Building Performance Standards

Northwest

LABORATORY

## **Open Building Control**



- Portable control sequences (CDL) that can be both simulated and compiled/executed
- Digitizing the control design/optimization/implementation pipeline

BERKELEY I

Pacific

NATIONAL

ABORATORY

## CDL, Brick, and SPARQL in OpenBuildingControl



### **After-Image**

#### Semantic modeling already enables building automation software development



The "next step" is autoconf/Cmake

### AMA

Thanks for the invitation!

amir.roth@ee.doe.gov