Ontology Summit

The Role of Ontologies in Building Automation

Matthew Giannini - SkyFoundry

What is a “Building”

- Residential home
- Fast-food restaurant
- Department stores
- Business parks
- College campuses
- Cities!

What's *in* a Building?

- Heating, Ventilation and Air Conditioning systems (HVAC)
- Metering (energy, gas, water)
- Lighting
- Security
- IP-connected "things" (IoT)

The Landscape

- Modern Building Automation Systems (BAS) make data collection (relatively) easy
  - Environmental conditions
  - Energy usage
  - Equipment operation

Data! Glorious Data!

The Problem

- Data is hard to utilize:
  - Different formats
  - Inconsistent naming conventions
  - Low-level
  - Lack of/Limited data descriptors
  - Unorganized
- It's expensive to "map" data into systems

If only…

- The data had “context”
- We had a common vocabulary
  - DA_TEMP
  - AHU1_TEMP_DA
- We had a common model

Project Haystack is the Solution

- Vision:
  - A standardized methodology for describing data will make it easier and more cost effective to analyze, visualize, and derive value from our operational data

Haystack is…

- Semantics through “tagging”
- Example: DA_TEMP

```
dis:"AHU1-SAT", sensor, discharge, air, temp, deg F, ahuRef -> AHU1

Point Name              descriptive tags                  association tag
```

Haystack is…

- **A set of standard equipment models** developed by consensus of the community
- An ongoing effort by "birds of a feather" to develop tagging models for equipment systems based on the Haystack tagging methodology

The following lists points commonly used with an AHU:

Discharge

- discharge air temp sensor
- discharge air humidity sensor
- discharge air pressure sensor
- discharge air flow sensor
- discharge air fan cmd
- discharge air fan sensor

Return

- return air temp sensor
- return air humidity sensor
- return air pressure sensor
- return air flow sensor
- return air co2 sensor
- return air fan cmd
- return air damper cmd

Mixed

- mixed air temp sensor

Haystack is…

- **A highly efficient REST API** that makes it easy to exchange Haystack tagged data among applications
- **A Java Reference implementation** of the Rest API that can be easily incorporated into applications and products to allow them to communicate via Haystack
- Collection of community tools
  - NHaystack – for "speaking" haystack with Niagara® systems
  - Tools to streamline tagging

How Haystack Solves the Problem

- Using Haystack, applications receive data **that includes the meta data** essential to describing the meaning of the data
- This enables automatic interpretation of the data by software applications
- **Dramatically reduces engineering effort**
- Machine-readable and people-readable!

Semantic Data! Glorious, Semantic Data!

Haystack Overview

- Tagging model
- Data formats
- HTTP protocol

Data Modeling – Domain

- Buildings
- Equipment and systems: HVAC, central plants, energy, lighting
- Points: sensors, actuators, commands, setpoints
- M2M / Internet of Things

Meta Model

- Meta model defines how we describe our model
- Relational databases: schema of tables and columns
- Java: classes, interfaces, members
- oBIX: contracts, prototype inheritance

Haystack Meta Model

- **Entities** model a physical or logical object
- **Tags** define a fact or attribute about an entity as a name/value pair
- Entity is a set of name/value pairs (the tags)
- Tag names are standardized or may be created by vendors or individual projects
- The 'id' tag is primary key

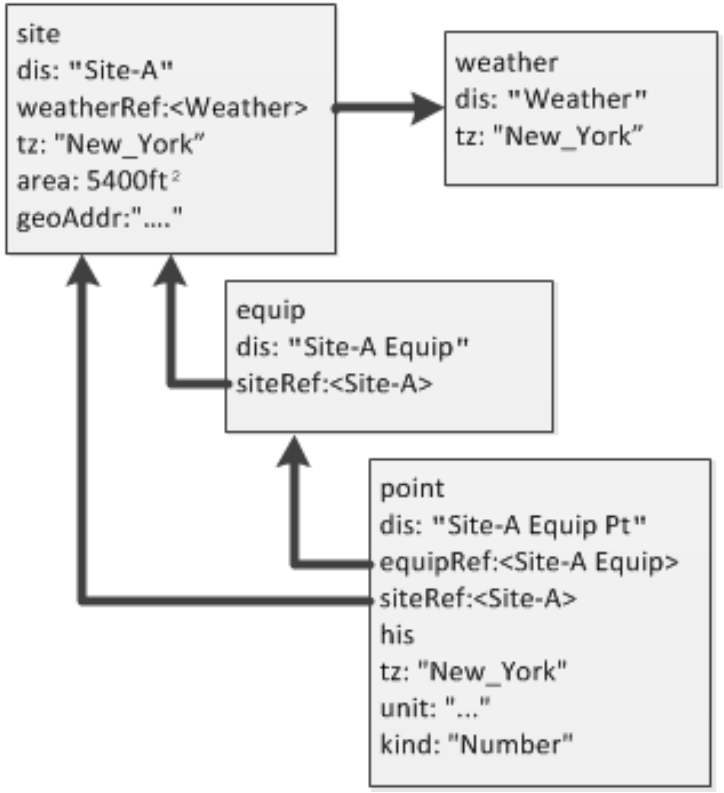
Haystack Tag Values

- **Bool**: true/false
- **Number**: integer or double with optional unit; 72.4°F, 6750ft²
- **Str**: Unicode, UTF-8; "100 Main St"
- **Uri**: `http://project-haystack.org`
- **Date**: 2013-04-31
- **Time**: 14:30:27.354
- **DateTime**: with timezone (tz db); 2013-04-31T14:30:00-04:00 New_York
- **Marker**: is-a, type-of
- **Ref**: cross reference other entities @ahu1
- **Coord**: C(lat,lng)
- **Bin**: binary file with MIME type; Bin(application/pdf)

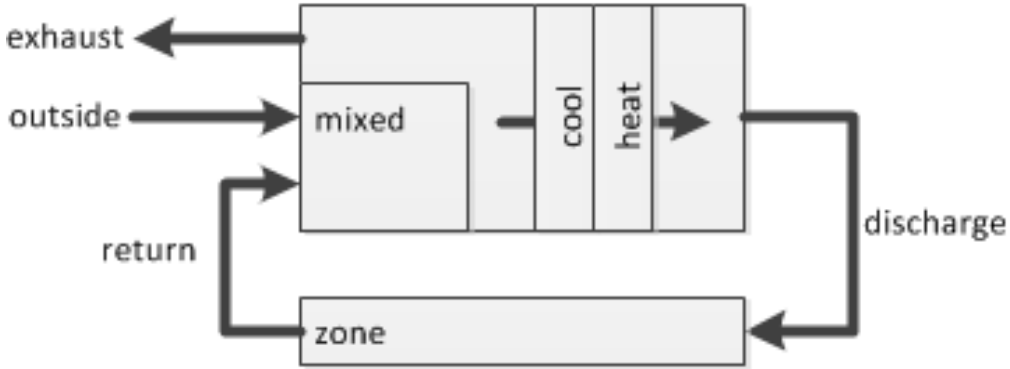
Point Example

```
id: @s1.ahu2.dat
dis: "Site-1 AHU-2 Discharge-Air-Temp"
discharge
air
temp
sensor
point
siteRef: @s1
equipRef: @s1.ahu2
kind: "Number"
unit: "°F"
tz: "New_York"
```

Site / Equip / Points



Air Handling Units (AHUs)




```
discharge air fan cmd
discharge air temp sensor
discharge air flow sensor
discharge air pressure sensor
zone air temp sensor
zone air temp sp
return air temp sensor
outside air temp sensor
outside air damper cmd
mixed air temp sensor
cool cmd
heat cmd
```

Data Formats

- Pass over network, store on file system
- Entities are map of tags (name/value pairs)
- A list of entities is represented as a grid (table)
- Standardized encodings:
  - Zinc: Zinc Is Not CSV
  - Json: Java Script Object Notation (RFC 4627)
  - CSV: Comma Separated Values (RFC 4180)

Grids

```
id: @site-a
dis: "Site A"
site
area: 4500ft²
```

```
id: @site-b
dis: "Site B"
site
phone: "555-1234"
```

id	dis	site	area	phone
@site-a	"Site-A"	✓	4500ft²	
@site-b	"Site-B"	✓		"555-1234"

Zinc Encoding

```
id       dis          site  area     phone
-------  --------     ----  -------  -----
@site-a  "Site-A"     ✓     4500ft²
@site-b  "Site-B"     ✓             "555-1234"
```

```
ver:"2.0"
id,dis,site,area,phone
@site-a,"Site-A",M,4500ft²,
@site-b,"Site-B",M,,"555-1234"
```

HTTP Protocol

- **Ops** are pluggable operations :
  - Standardized
  - Vendor specific
- Request **grid** ⇨ response **grid**
- HTTP GET, grid row specified in query parameters
- HTTP POST, grid encoded in body
- HTTP Content Negotiation via "Accept" Header
- `http://server/haystack/{op}`

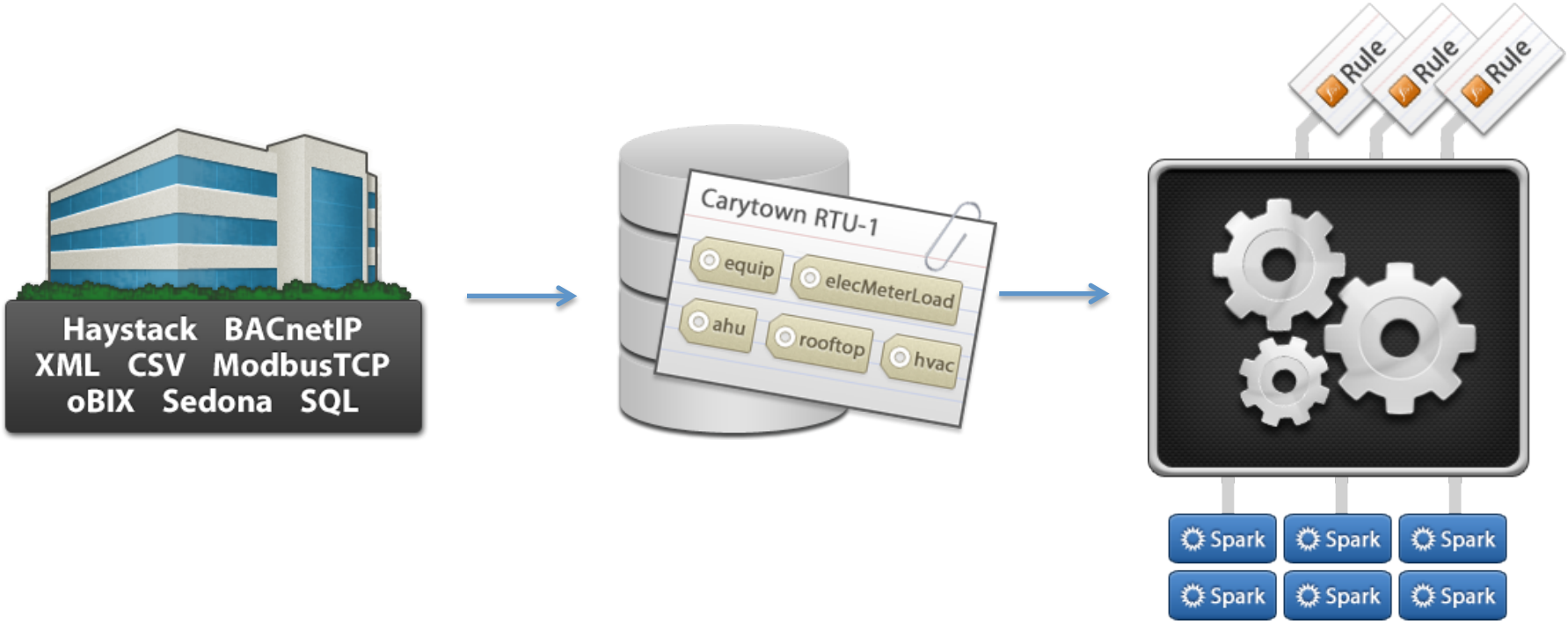
Haystack Read Op

```
GET /haystack/read?filter=point%20and%20sp HTTP/1.1
Host: localhost
Accept: text/csv

POST /haystack/read HTTP/1.1
Host: localhost
Accept: text/csv
Content-Type: text/zinc; charset=utf-8
Content-Length: 33

ver: "2.0"
filter
"point and sp"
```

Standard Ops

- about
- ops
- formats
- read (by ids or with filter)
- nav
- watchSub / watchUnsub / watchPoll
- pointWrite
- hisRead / hisWrite
- invokeAction

SkySpark: Haystack in Action

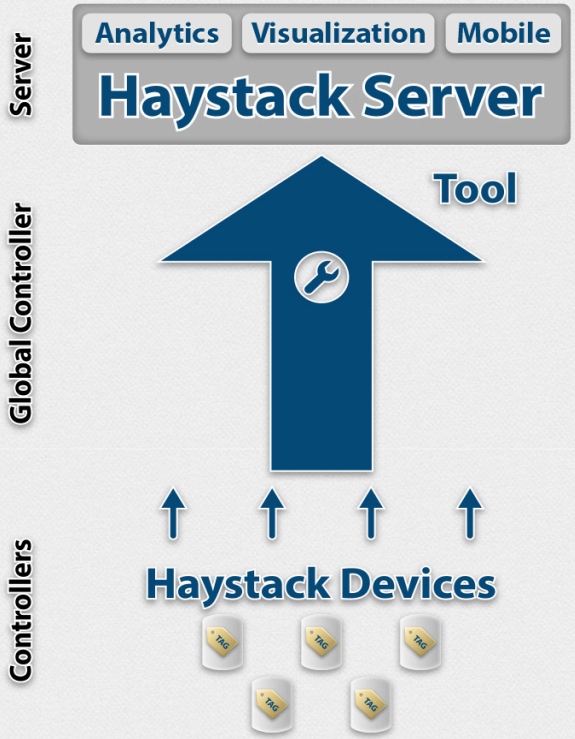


Example Spark

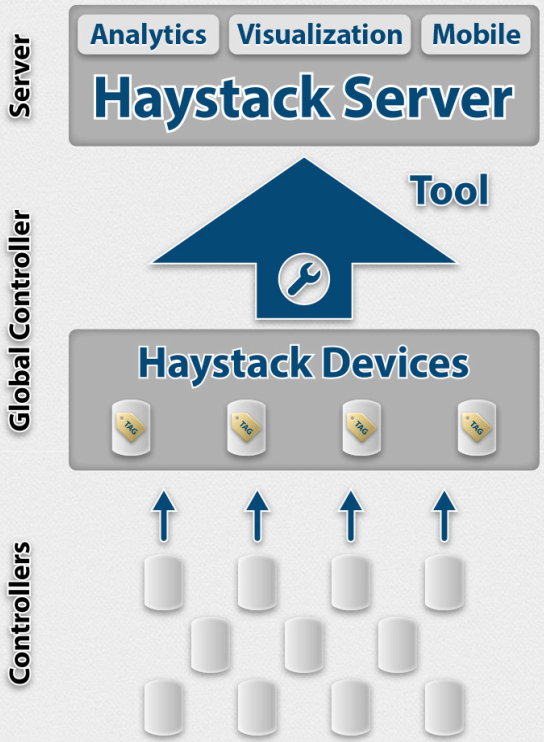
Resources

- www.project-haystack.org
  - Docs for tags, models, formats, protocols
  - Forum for all discussion
- Haystack Connect
  - http://haystackconnect.org/
  - May 18-20
- SkySpark by SkyFoundry
  - http://skyfoundry.com/skyspark

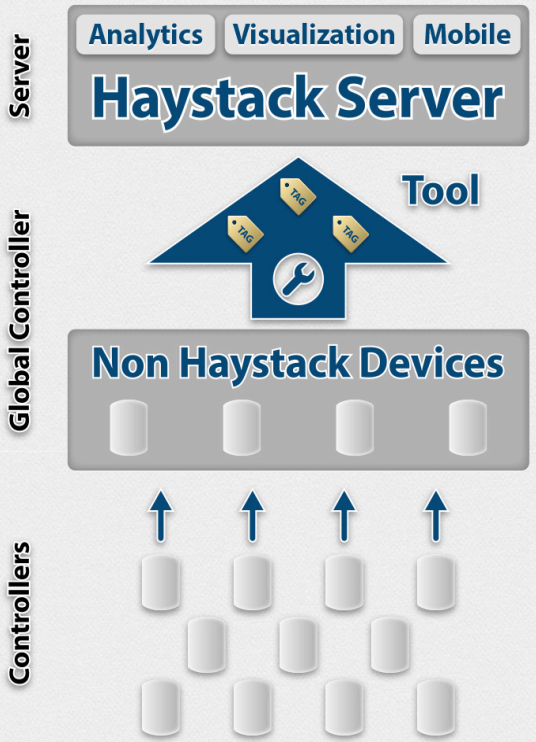
Architectures Using Haystack



Tags exist in end devices



Tags exist in network controllers



Tags applied in server level application