BITS, BRICKS, BEHAVIOUR
Data integration for smart communication

9 November 2021

Dr. Pieter Pauwels, Eindhoven University of Technology

This project received funding from Dutch Ministry of Economic Affairs and Climate Policy and Ministry of the Interior and Kingdom Relations under the MOOI program

Who am I?

Associate Professor TU Eindhoven (2019-...)

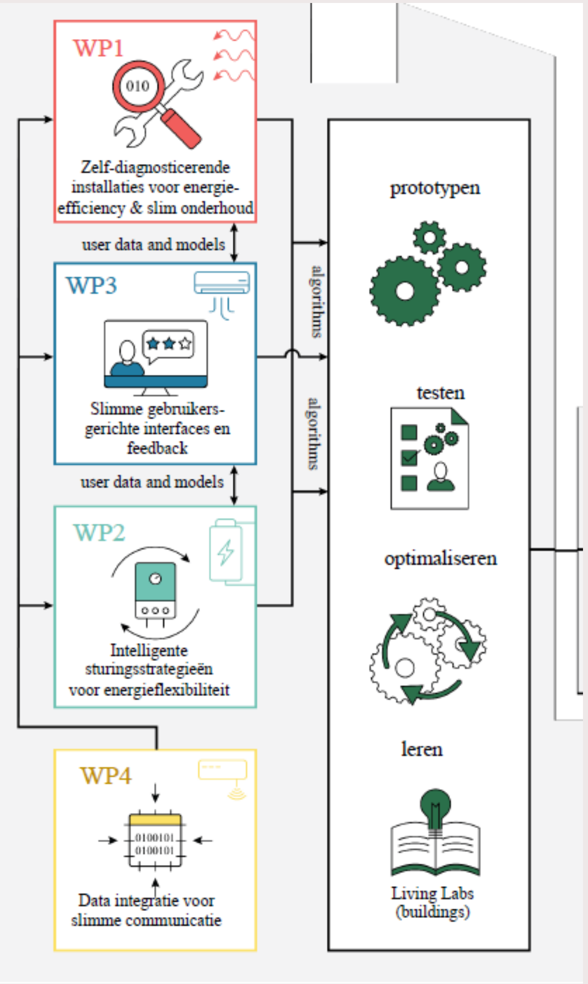
Assistant Professor Ghent University (2016-2019)

Postdoc Ghent University (2014-2016)

Postdoc University of Amsterdam (2012-2014)

Master & PhD in Civil Engineering – Architecture @Ghent University (2008, 2012)

Data Integration for Smart Buildings



To enable making our buildings smarter, advanced data integration is needed (among several other matters):

- Ensure **data connectivity** between applications
- Ensure security, **ethical use and privacy of data**
- **Standardise** data sets and approaches
- Aim for **system integration at API level**, between individual systems of diverse manufacturers

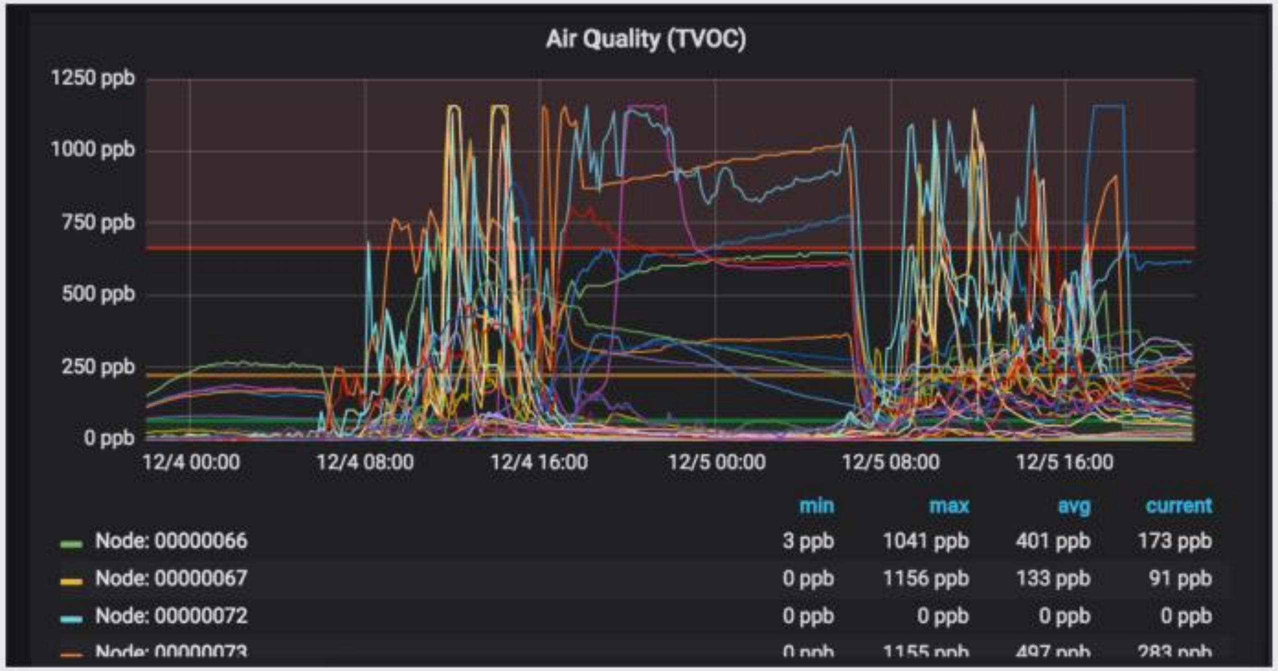
Presentation Outline

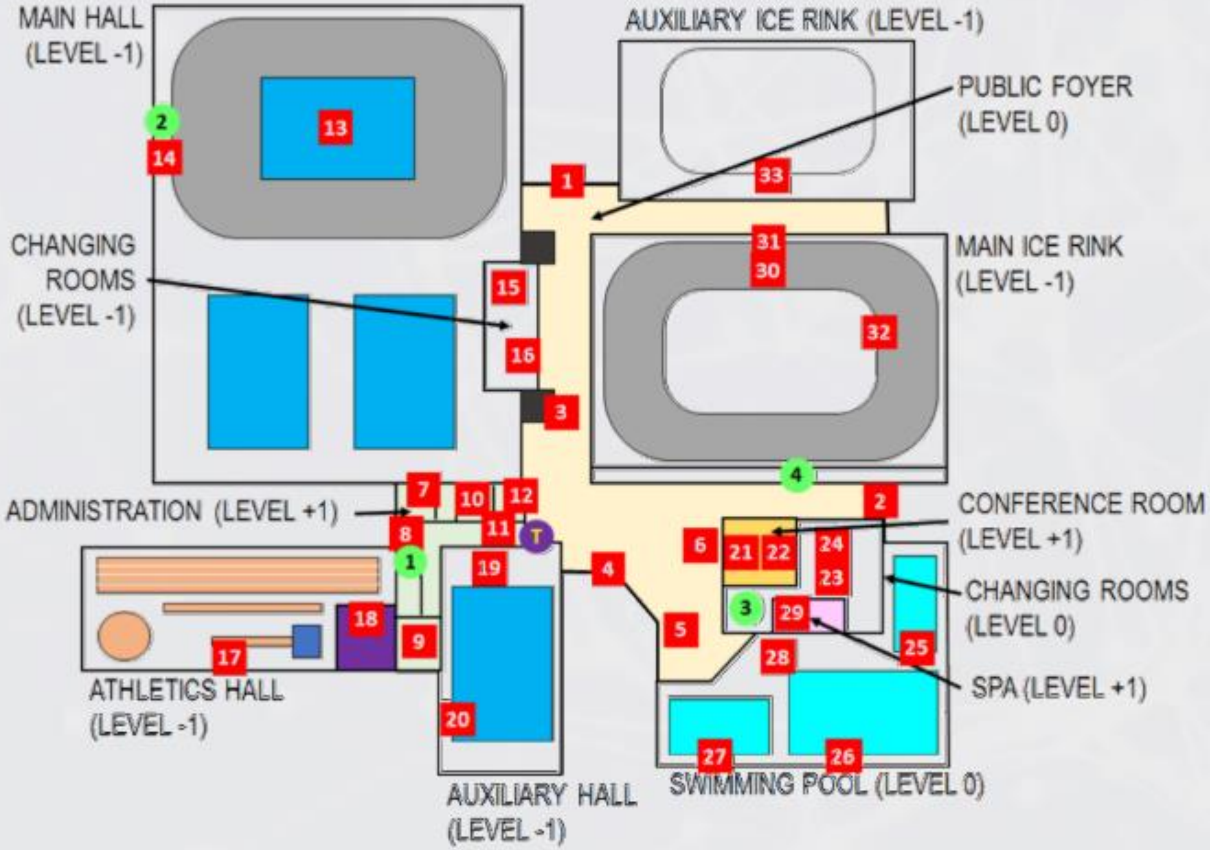
1. A reference system architecture
2. Core metadata schemas: BIM, IFC, LBD, BRICK, Haystack, REC, SSN
3. Connecting to streamed data, edge communication, 3D and control logic

SENSOR DATA EXAMPLE: GIGANTIUM IOT LIVING LAB AALBORG

- 35 sensor nodes monitoring Temperature (°C), Relative Humidity (%), Air Pressure (hPa), Indoor Air Quality (Total Volatile Organic Compounds ((TVOC), ppb) and CO2 (ppm)), illuminance (lux) and motion

- Data storage in SQL database
- Data monitoring and visualization in Grafana

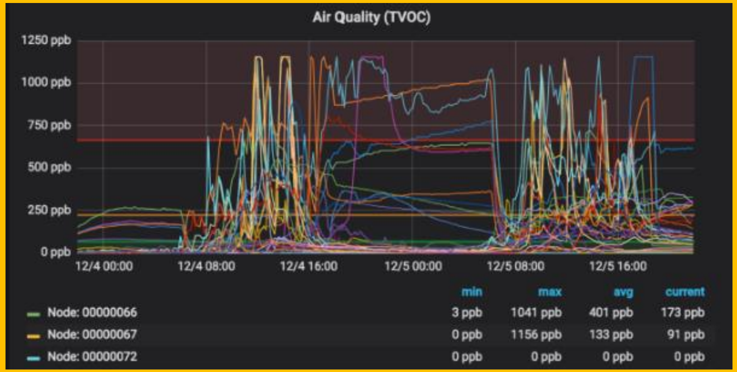


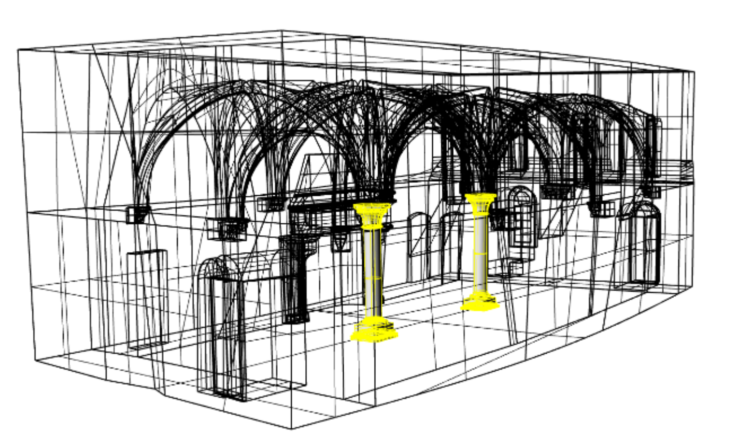


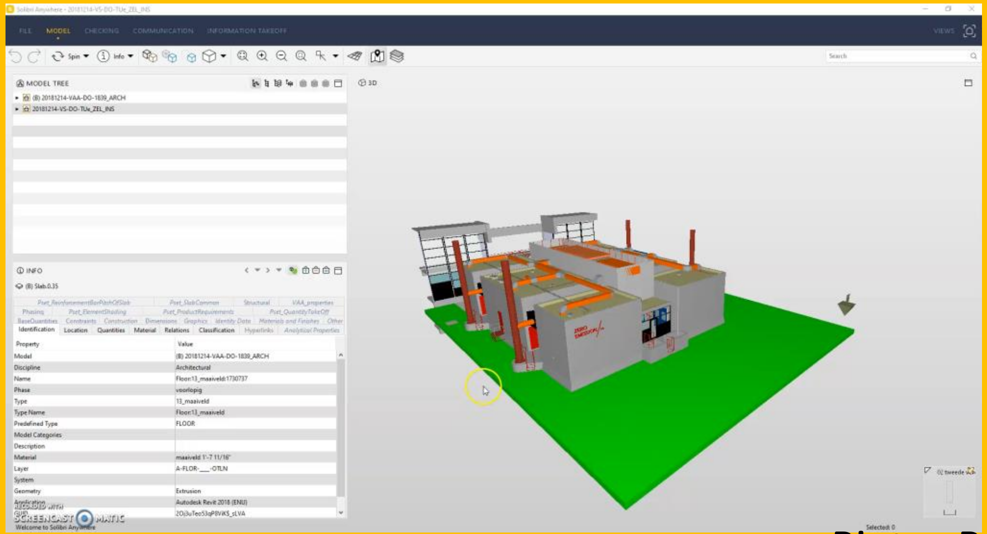
Source: Rodriguez et al. (2018)

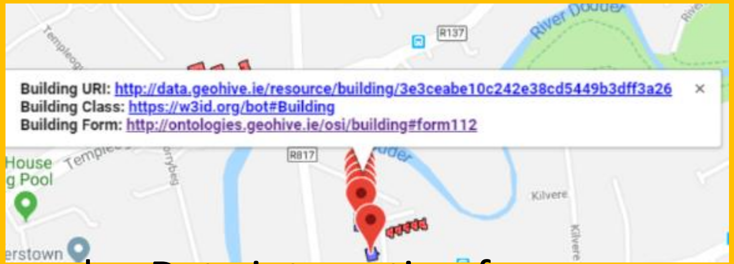
Petrova, E., Pauwels, P., Svidt, K., Jensen, R.L. (2018) From patterns to evidence: Enhancing sustainable building design with pattern recognition and information retrieval approaches. Proceedings of the 12th ECPPM conference, pp. 391-399.

A building has different types of data associated









Categories of data?

- **Data streams:**
  - Ordered lists of values, typically floating point values
  - Large amount of data -> data lakes
  - Almost no semantics, at best a few labels for categorization
  - E.g. temperature measurements, system logs (e.g. triggering of actuators), etc.
- **Semantically rich and interconnected data:**
  - Seldom including large data streams
  - Long debates about the semantics of things -> standardisation
  - Complex and brittle (breaks easily)
  - Small amount of very important data
  - Easy to combine with rule-based and/or logic-based technologies (inference and query)
  - E.g. BIM models, semantic web ontologies, taxonomies, OTLs, etc.
- **Control models:**
  - Algorithms for control, parametric functions
  - Communication system, signal processing, direct control, low latency
  - Typically located on the edge (devices with embedded functions)
  - E.g. Control Description Language (CDL), modelica models
- **User data:**
  - Outside of the system
  - Different privacy and security requirements
- **Files:**
  - No semantically rich encodings, no data streams
  - Highly valuable, and seldom machine-processable
  - E.g. PDFs, Images, Geometry ‘blobs’

Categories of data?

- Data streams:
  - Ordered lists of values, typically floating point values
  - Large amount of data -> data lakes
  - Almost no semantics, at best a few labels for categorization
  - E.g. temperature measurements, system logs (e.g. triggering of actuators), etc.
- Semantically rich and interconnected data:
  - Seldom including large data streams
  - Long debates about the semantics of things -> standardisation
  - Complex and brittle (breaks easily)
  - Small amount of very important data
  - Easy to combine with rule-based and/or logic-based technologies (inference and query)
  - E.g. BIM models, semantic web ontologies, taxonomies, OTLs, etc.

MACHINE LEARNING

STATISTICAL AI METHODS

- Control models
  - Algorithms for control, parametric fu ns
  - Communication system, signal processing, direct control, low latency
  - Typically located on the edge (devices with embedded functions)
  - E.g. Control Description Language (CDL), modelica models
- User data:
  - Outside of the system
  - Different privacy and s
- Files:
  - No semantically rich e ams
  - Highly valuable, and s sable
  - E.g. PDFs, Images, Geo

SEMANTICS

SYMBOLIC AI METHODS

Strengths and methods in Artificial Intelligence

Data streams

MACHINE LEARNING
STATISTICAL AI METHODS

- **Reinforcement learning**: data comes from an available experimentation environment
- **Neural networks**: ANNs, GNNs, CNNs, …
- **Traditional ML**:
  - **Regression**: predict next value
  - **Classification**: predict category / classification label
  - **Clustering**: group based on similarity
  - **Association**: identify sequences and combinations

SEMANTICS
SYMBOLIC AI METHODS

- **Semantic Web** technologies:
  - **Ontologies** and **formal** vocabularies
  - **Logics**: Description Logic, Defeasible Logic, etc.
- **Expert Systems**
- **Rule-based inference:** if-then rules

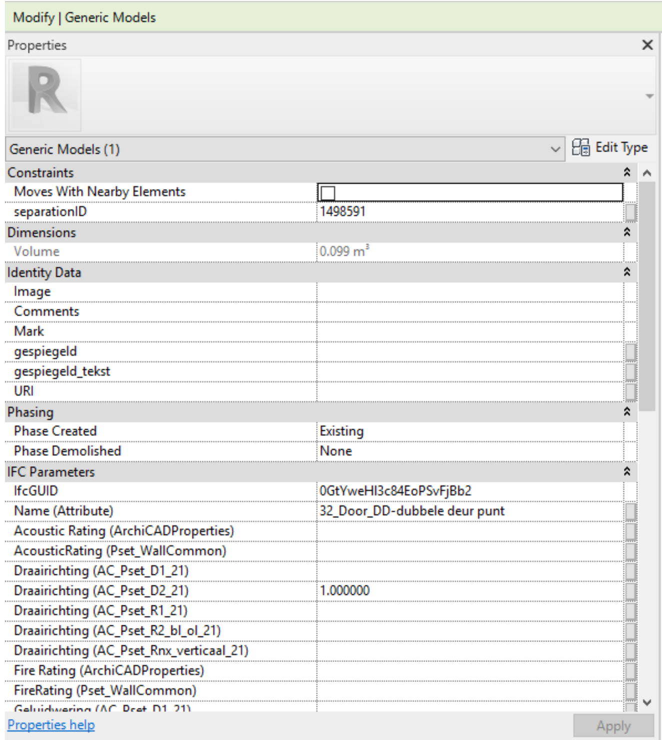
Semantically rich and interconnected data

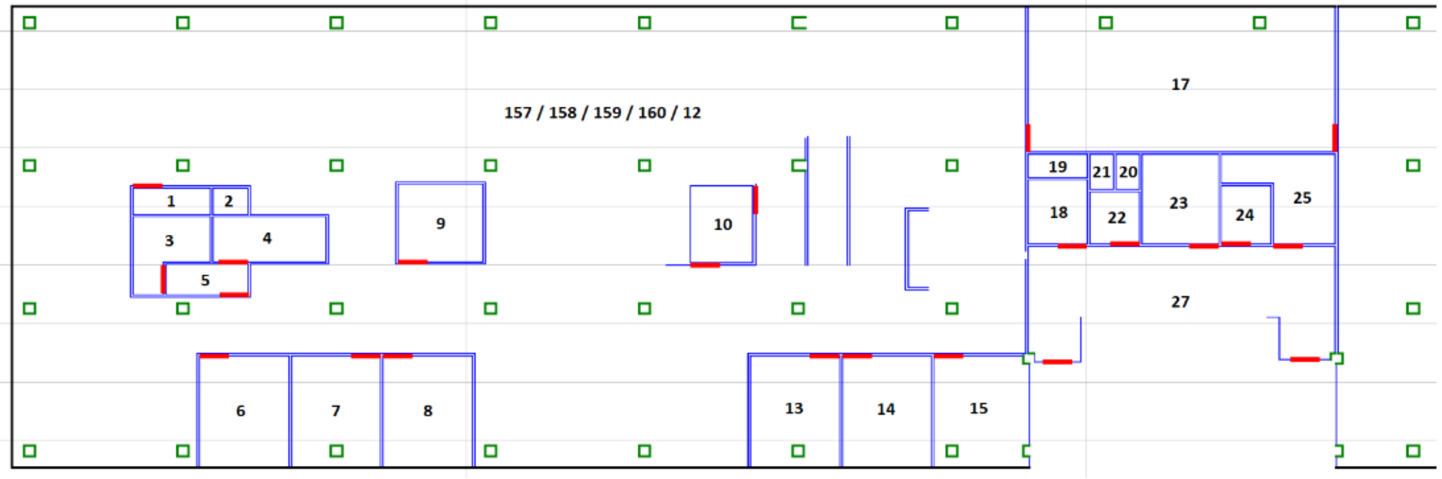
The Scale of AI Methods

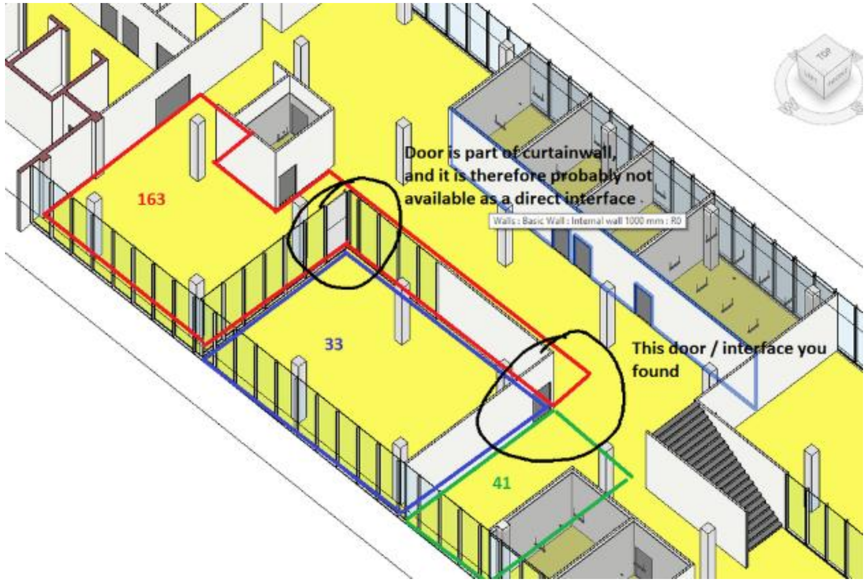
STATISTICAL AI **SYMBOLIC AI**

Can we integrate these methods and use all of them?

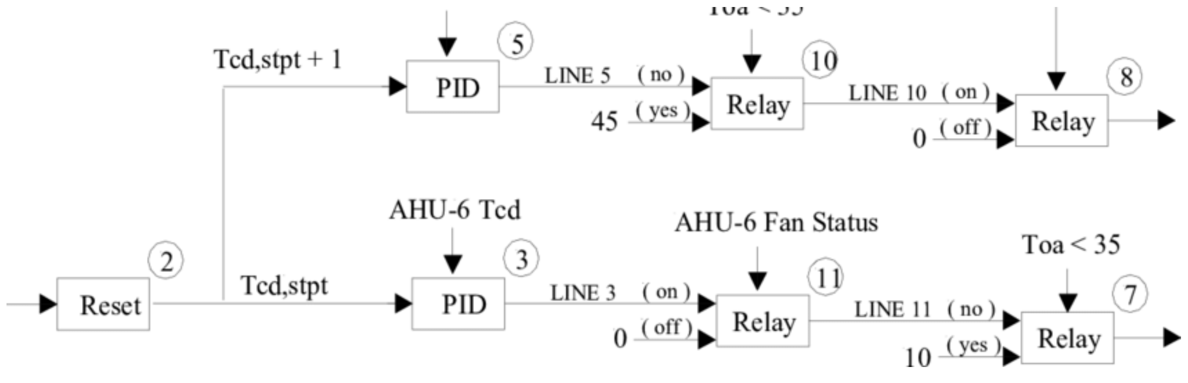
Semantic encoding of buildings

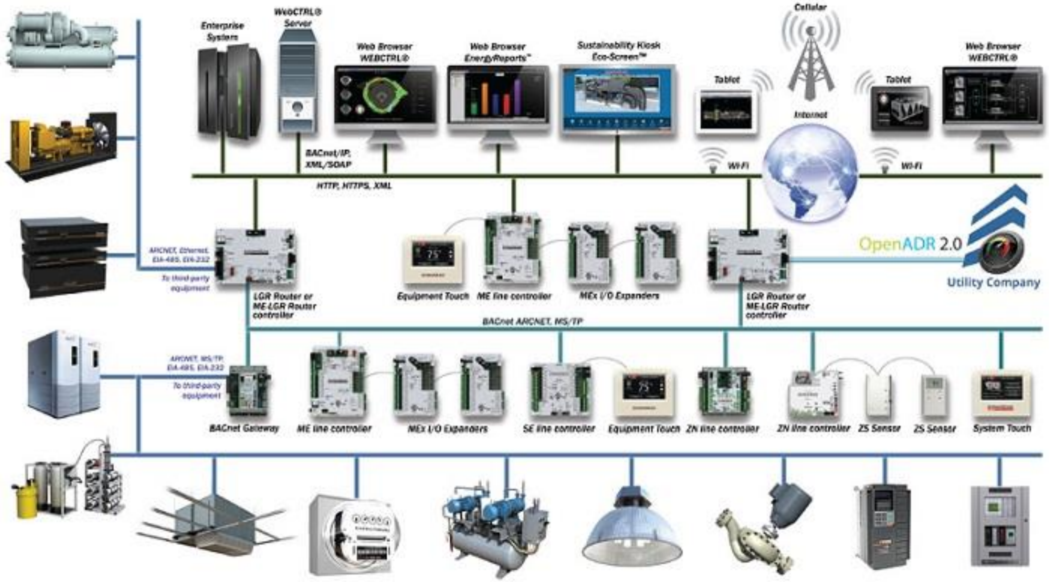


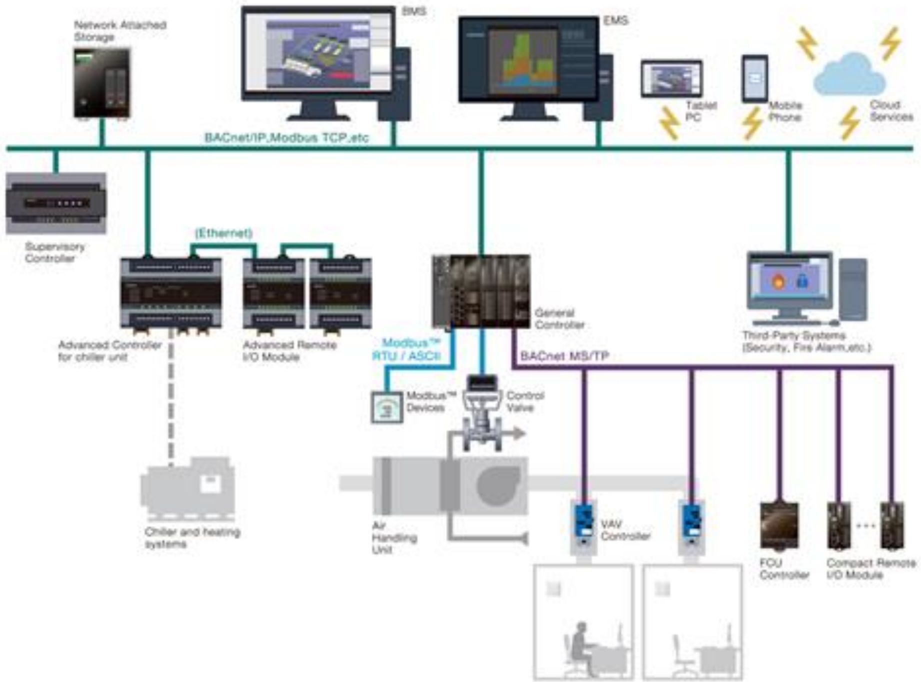


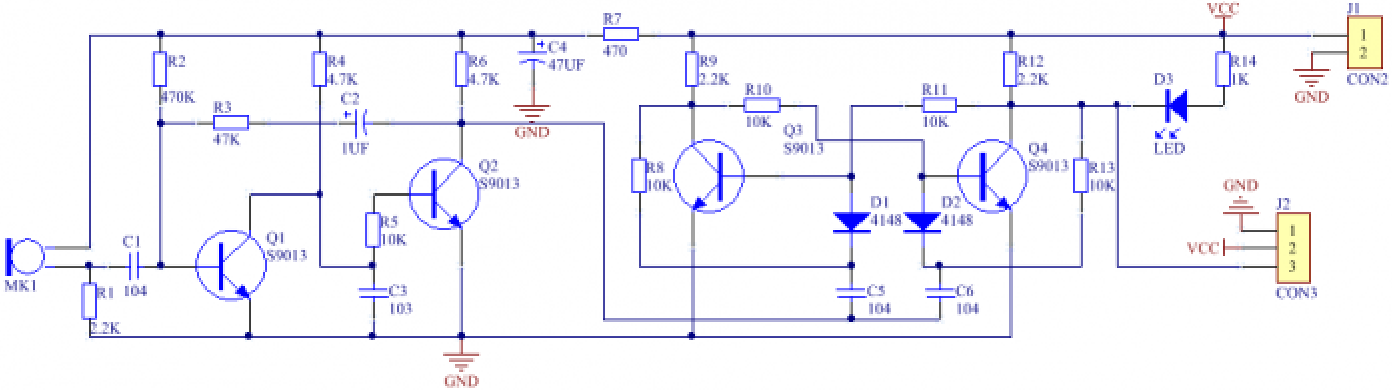


And all of this excludes control systems and control logic!!

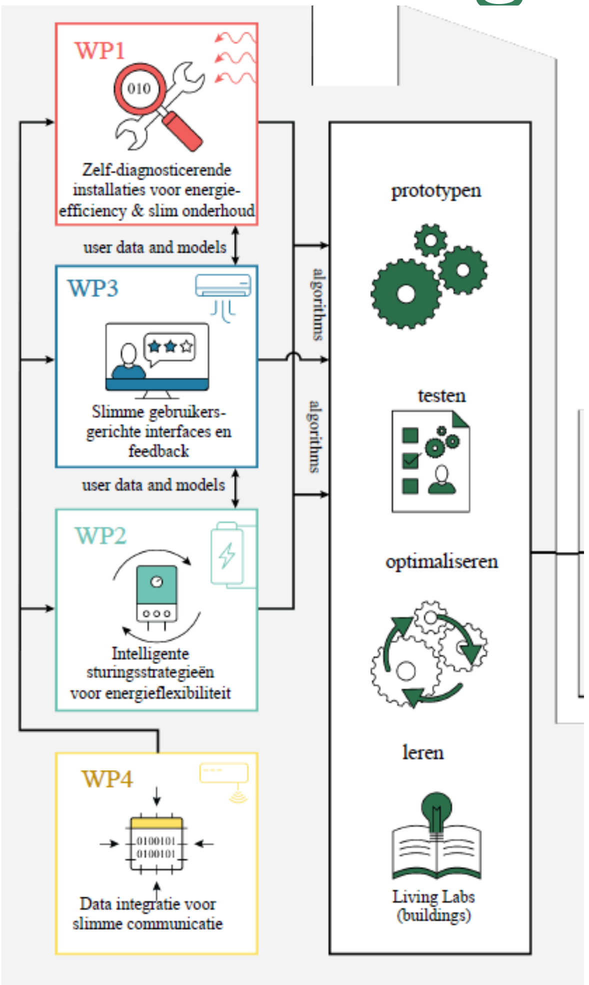








Data Integration for Smart Buildings

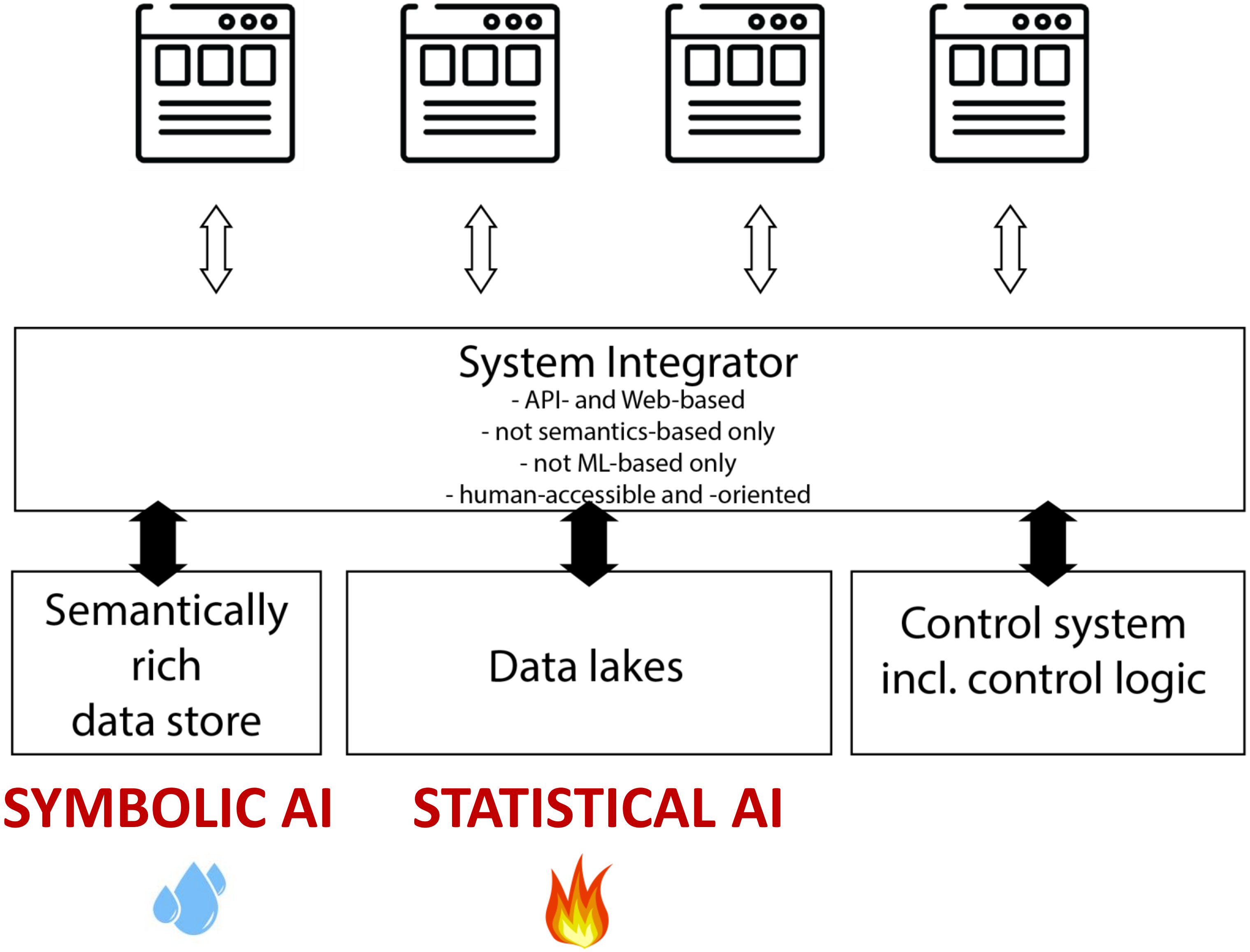


To enable making our buildings smarter, advanced data integration is needed (among several other matters):

- Ensure **data connectivity** between applications
- Ensure security, **ethical use and privacy of data**
- **Standardise** data sets and approaches
- Aim for **system integration at API level**, between individual systems of diverse manufacturers

Solution: combine inherently incompatible techniques using a system integration approach

Targeted framework for AI-based smart buildings



Important:

- Include access control (ACL)
- 'linking' of data on system integration level
- Agreement and standardization of labels and metadata tags
- Feed back into control systems!

Presentation Outline

1. A reference system architecture
2. Core metadata schemas: BIM, IFC, LBD, BRICK, Haystack, REC, SSN
3. Connecting to streamed data, edge communication, 3D and control logic

Labels and Tags

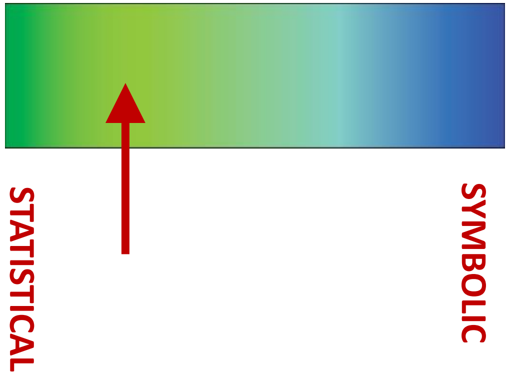
Extract from BMS (labels):

Timestamp	11NR008FT-013FLW	11NR008FT-038FLW	11NR008FT-039FLW	11NR008FT-040FLW	11NR008FT-301FLW	11NR008FT-302FLW	11NR008LT-001PIRTM	11NR008LT-003PIRTM
01/01/2021 00:02	0	0	0	280	0	0	20	20
01/01/2021 01:02	0	0	0	300	0	0	20	20
01/01/2021 02:02	0	0	0	300	0	0	20	20
01/01/2021 03:02	0	0	0	280	0	0	20	20
01/01/2021 04:02	0	0	0	300	0	0	20	20
01/01/2021 05:02	0	0	0	300	0	0	20	20
01/01/2021 06:02	0	0	0	300	0	0	30	30
01/01/2021 07:02	0	0	0	300	0	0	30	30
01/01/2021 08:02	0	0	0	300	0	0	30	30
01/01/2021 09:02	0	0	0	300	0	0	30	30
01/01/2021 10:02	0	0	0	300	0	0	30	30
01/01/2021 11:02	0	0	0	300	0	0	30	30
01/01/2021 12:02	0	0	0	300	0	0	30	30
01/01/2021 13:02	0	0	0	300	0	0	30	30
01/01/2021 14:02	0	0	0	280	0	0	30	30
01/01/2021 15:02	0	0	0	300	0	0	30	30

Mapping of sensors to rooms in BMS and then building model:

Column1	ItemName	ItemDescriptionDutch	ItemDescriptionEnglish	Spaces
6358	11NR009TE-033TRL	RUIMTETEMPERATUUR 9_Z01	ROOM TEMPERATURE 9_Z01	9_Z01
6359	11NR009TE-030TRL	RUIMTETEMPERATUUR 9_Z01	ROOM TEMPERATURE 9_Z01	9_Z01
2089	11NR009TE-030TRLBP	ACTUEEL SETPOINT REGELING 9_Z01	CURRENT SETPOINT SCHEME 9_Z01	9_Z01
6824	11NR009SCR030BNDRS	STAND ZONWERING 9_Z01	STAND AWNINGs 9_Z01	9_Z01
5900	11NR009CV-030RWWRS	REGELSIGNAAL VERWARMING 9_Z01	CONTROL SIGNAL HEATING 9_Z01	9_Z01
5899	11NR009CV-033RWWRS	REGELSIGNAAL VERWARMING 9_Z01	CONTROL SIGNAL HEATING 9_Z01	9_Z01
4349	11NR009TE-033MAXBP	KOELINGSETPOINT 9_Z01	COOLING SETPOINT 9_Z01	9_Z01
4350	11NR009TE-030MAXBP	KOELINGSETPOINT 9_Z01	COOLING SETPOINT 9_Z01	9_Z01
3568	11NR009TE-030CPA	CPA VIA WANDMODULE RUIMTE 9_Z01	CPA VIA WALL MODULE ROOM 9_Z01	9_Z01
3567	11NR009TE-033CPA	CPA VIA WANDMODULE RUIMTE 9_Z01	CPA VIA WALL MODULE ROOM 9_Z01	9_Z01
6823	11NR009SCR033BNDRS	STAND ZONWERING 9_Z01	STAND AWNINGs 9_Z01	9_Z01

Haystack Tags



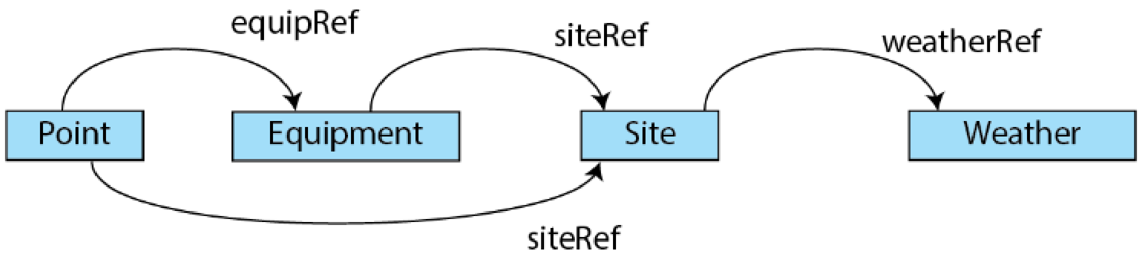
“We standardize semantic data models and web services with the goal of making it easier to unlock value from the vast quantity of data being generated by the smart devices that permeate our homes, buildings, factories, and cities.” (https://project-haystack.org/)

Haystack = stack of technologies:

- ***Data Types**: a fixed set of data types for modeling information*
- ***File Types**: a set of text formats to encode and exchange those data types*
- ***HTTP API**: a protocol to exchange data over HTTP using those file types*
- ***Ontology**: a standard way to model common concepts such as buildings, equipment, and sensors*
- ***Defs**: a standard way to define and extend the ontology*

Individual aspects of the technology stack can be used on their own. For example you can use the Haystack data types as an "enhanced JSON". Or you could use just the terms in the ontology without the Haystack data types.

Ontology Core:



RDF encoding / mapping example

URI

```
def: ^lib:phIoT
doc: "Project Haystack definitions for Internet of Things"
version: "4.0"
baseUri: `https://project-haystack.org/def/phIoT/`
depends: [^lib:ph, ^lib:phScience]
```

```
https://project-haystack.org/def/phIoT/4.0#site
```

Ontology (vocabulary)

```
def: ^site
is: [^entity, ^geoPlace]
mandatory
doc: "Site is a geographic location of the built environment"
```

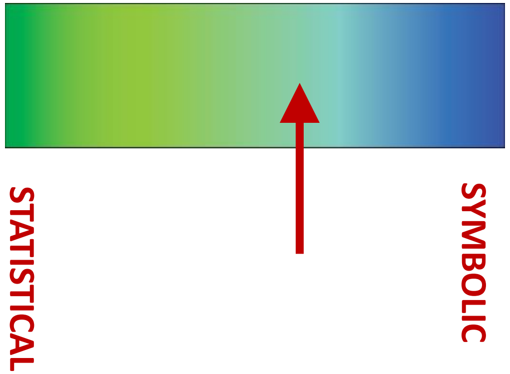
```
phIoT:site a owl:Class ;
    rdfs:subClassOf ph:entity,
        ph:geoPlace ;
    rdfs:comment "Site is a geographic location of the built environment" ;
    ph:is ph:entity,
        ph:geoPlace ;
    ph:lib phIoT:lib:phIoT ;
    ph:mandatory ph:marker .
```

Instances (data)

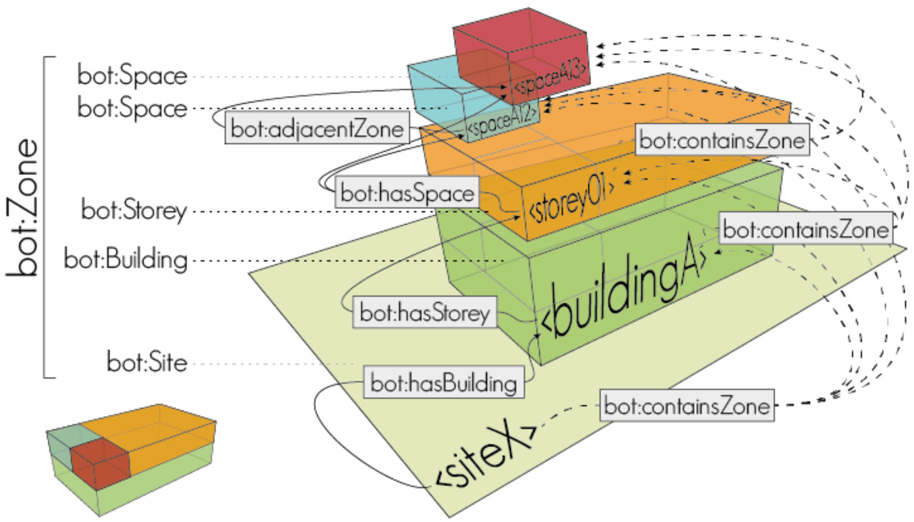
```
id:@24192ca1-0c85f75d "Headquarters"
site
area:140797ft²
tz:New_York
dis:Headquarters
geoAddr:"600 W Main St, Richmond, VA"
geoCoord:C(37.545826,-77.449188)
hq
metro:Richmond
primaryFunction:Office
yearBuilt:1999
```

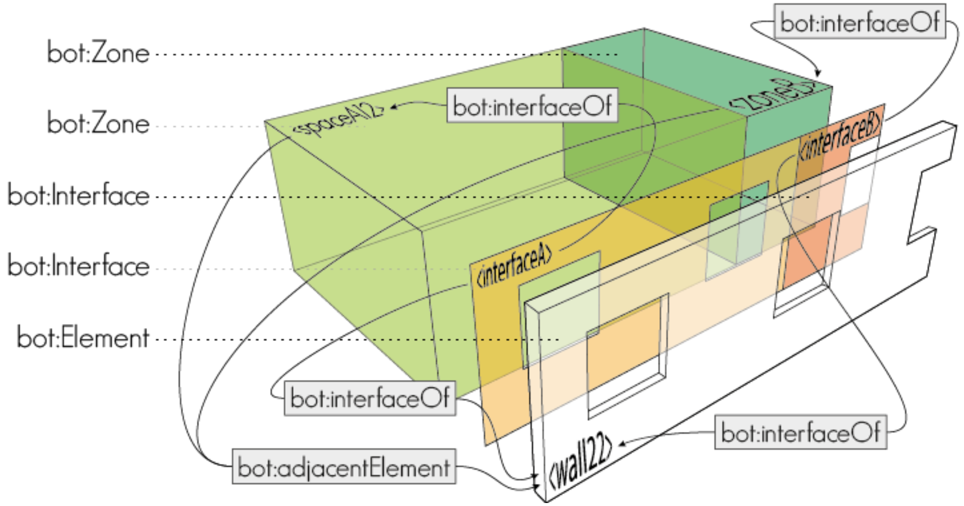
```
_:24192ca1-0c85f75d
    a phIoT:site ;
    ph:hasTag site,
    phIoT:area 140797 ;
    ph:tz "New_York" ;
    ph:dis "Headquarters" ;
    ph:geoAddr "600 W Main St, Richmond, VA" ;
    ph:geoCoord "C(37.545826,-77.449188)" ;
    phIoT:primaryFunction "Office" ;
    phIoT:yearBuilt 1999 .
```

Linked Building Data, incl. Brick

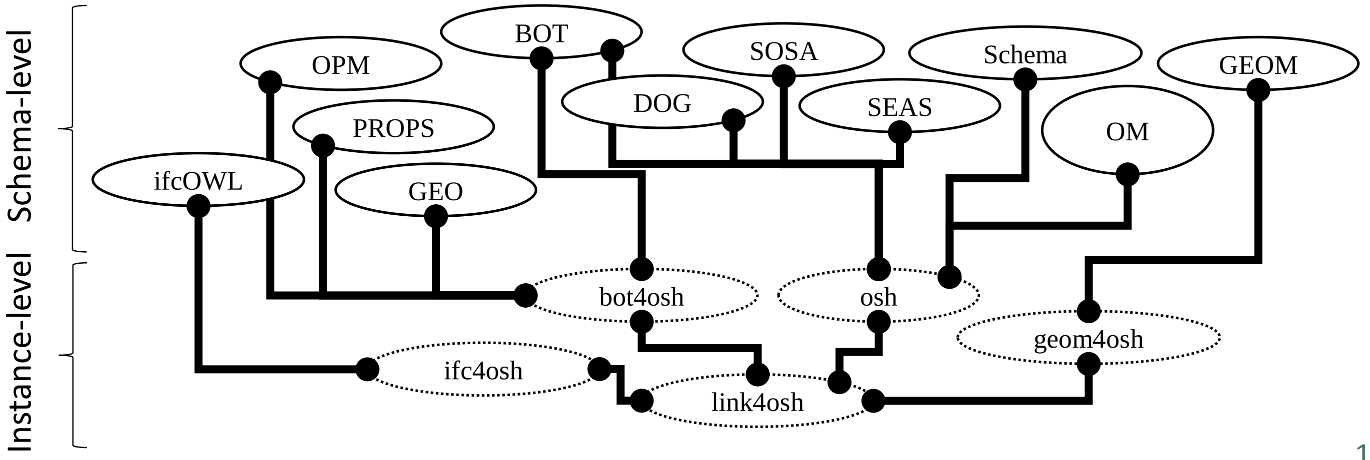


Ontology Core:

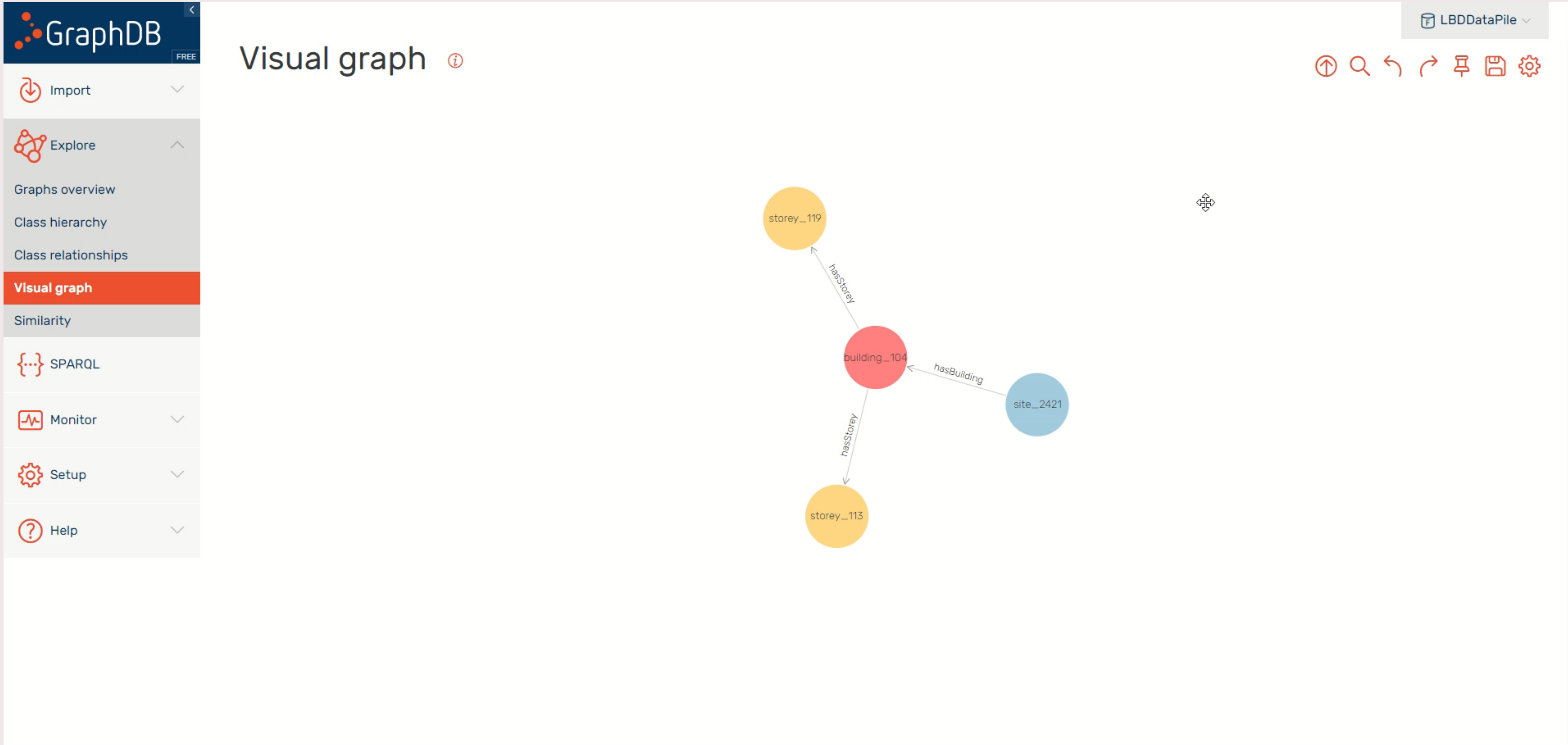




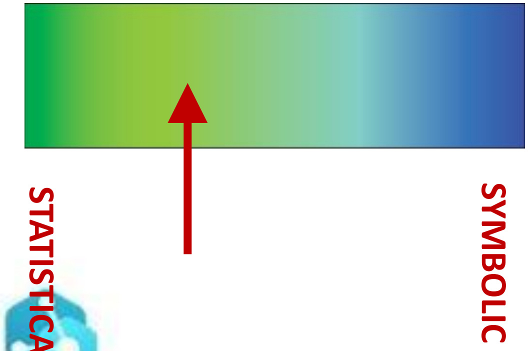
Possibility to extend with other vocabularies:



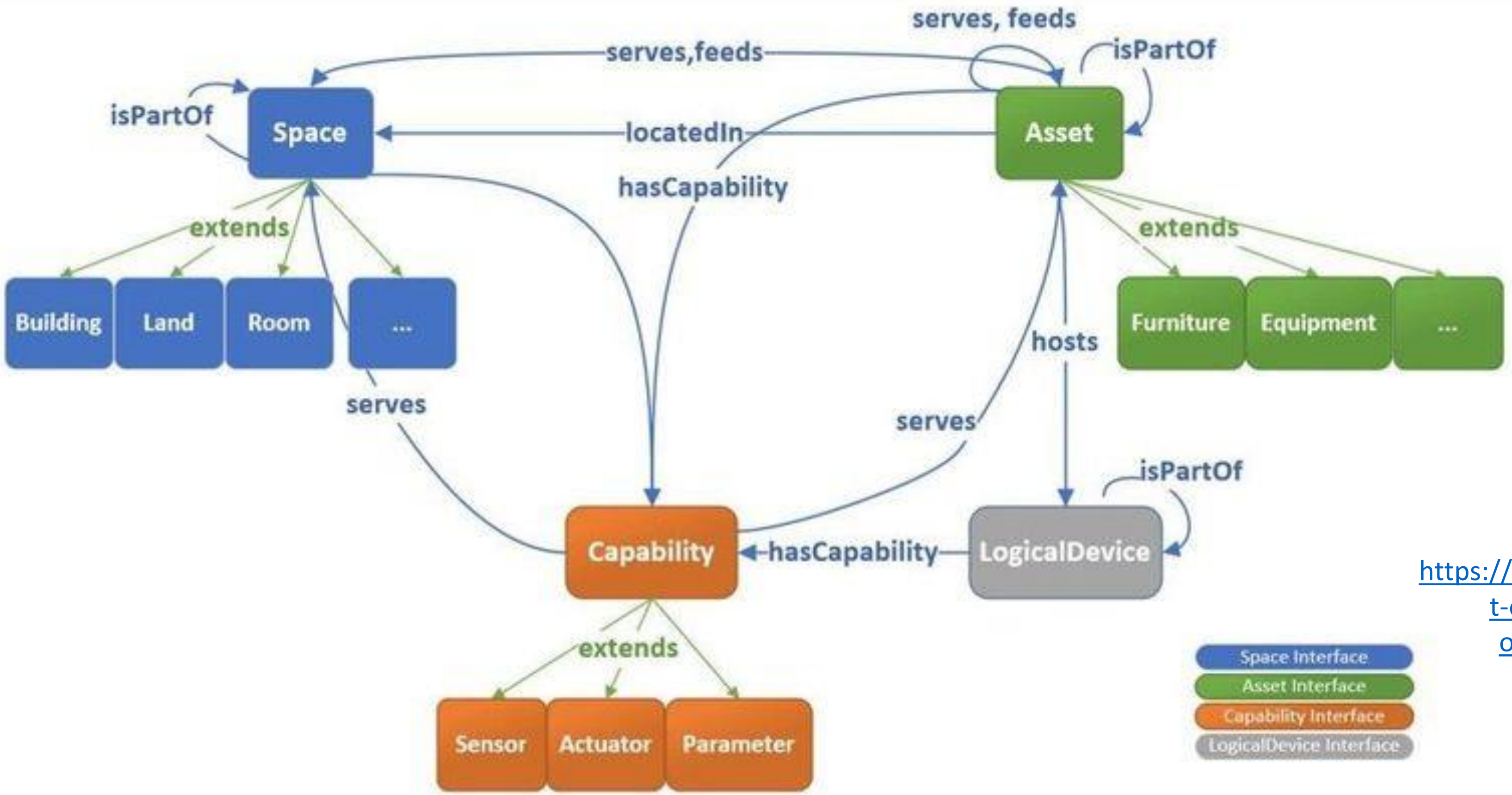
Cloud of linked building data



Real Estate Core

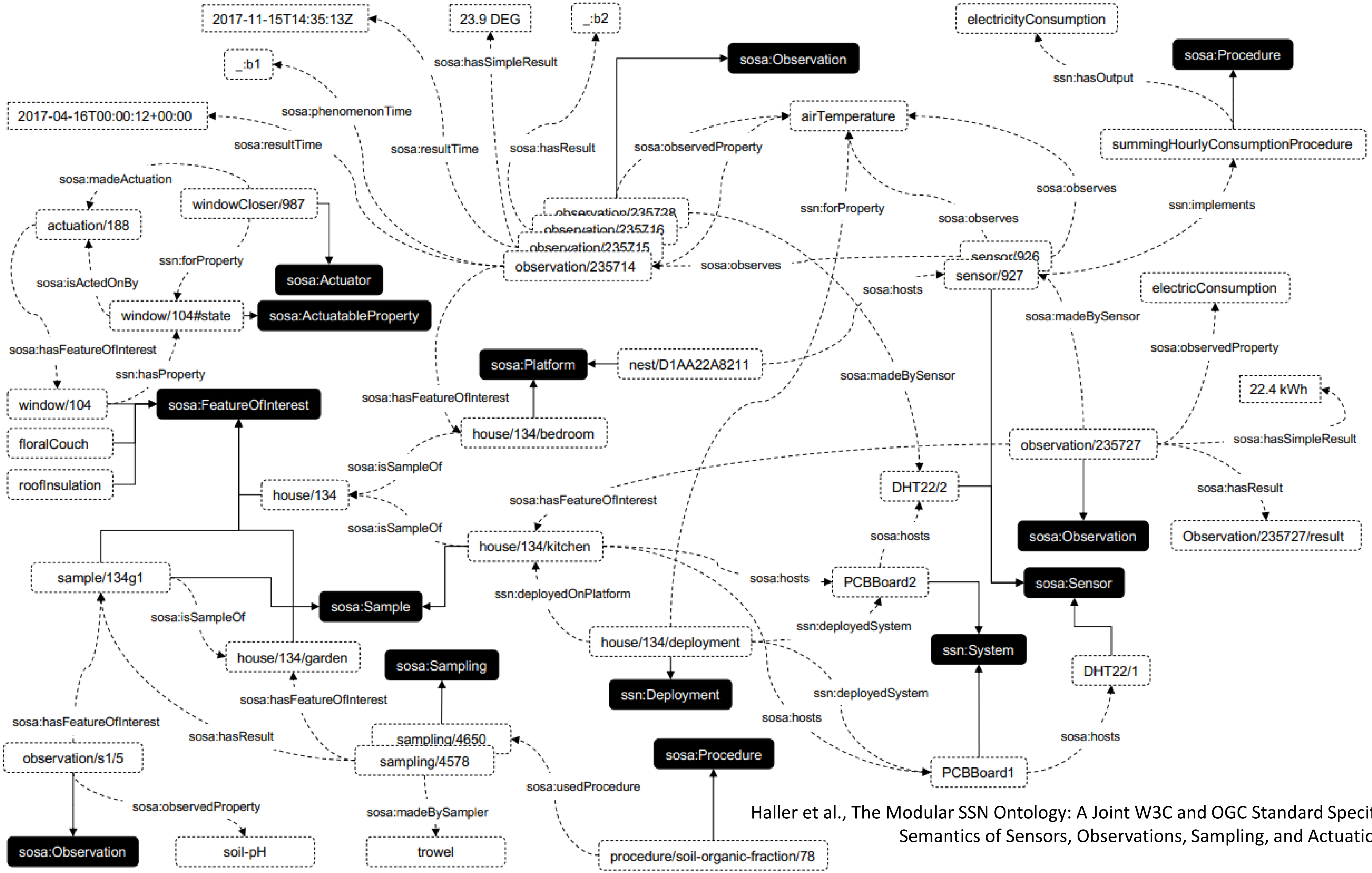


Microsoft RealEstateCore JÖNKÖPING UNIVERSITY willow know your world Azure Digital Twins



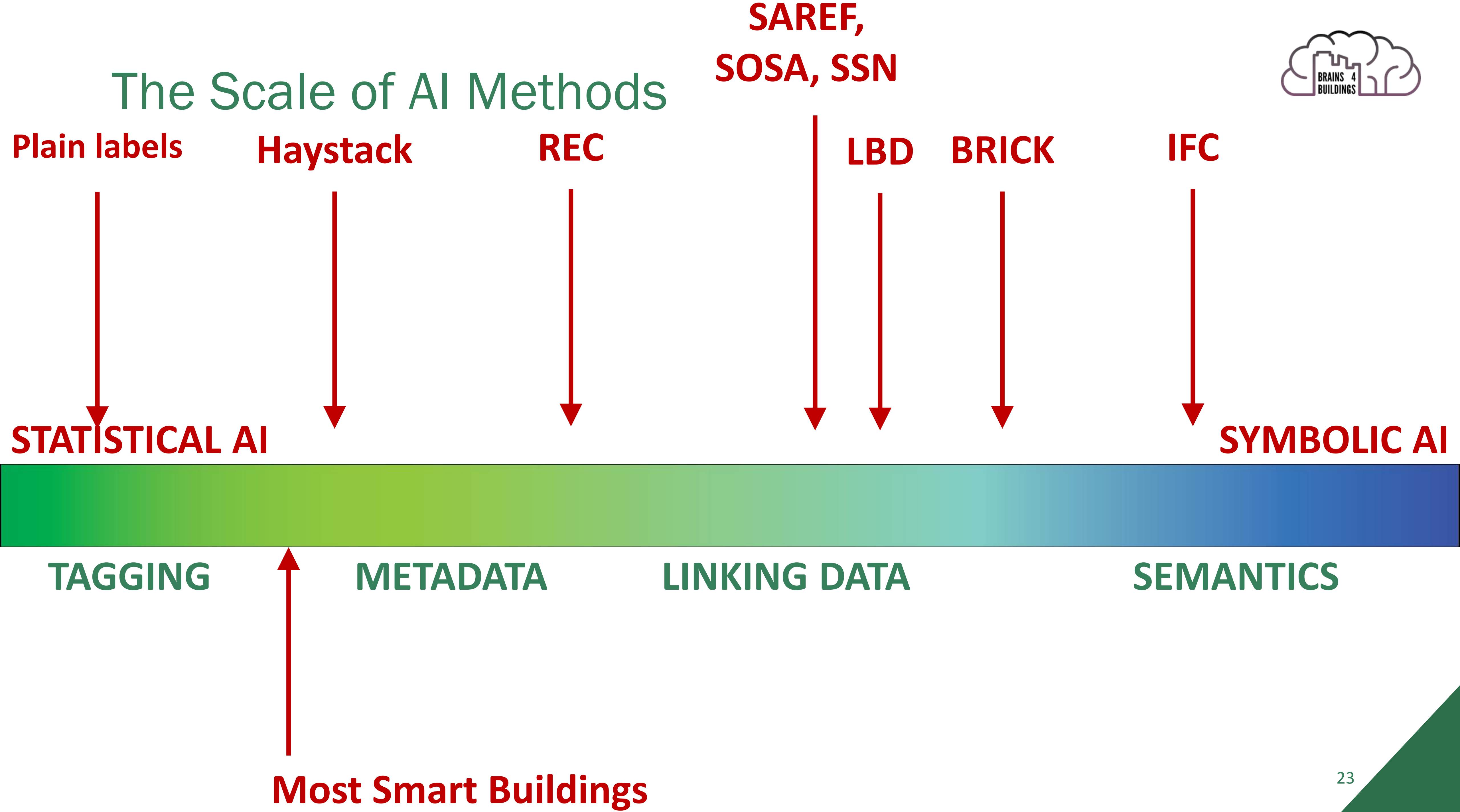
https://techcommunity.microsoft.com/t5/internet-of-things/realestatecore-a-smart-building-ontology-for-digital-twins-is/ba-p/1914794

Hartig et al., 2019



Haller et al., The Modular SSN Ontology: A Joint W3C and OGC Standard Specifying the Semantics of Sensors, Observations, Sampling, and Actuation, 2018.

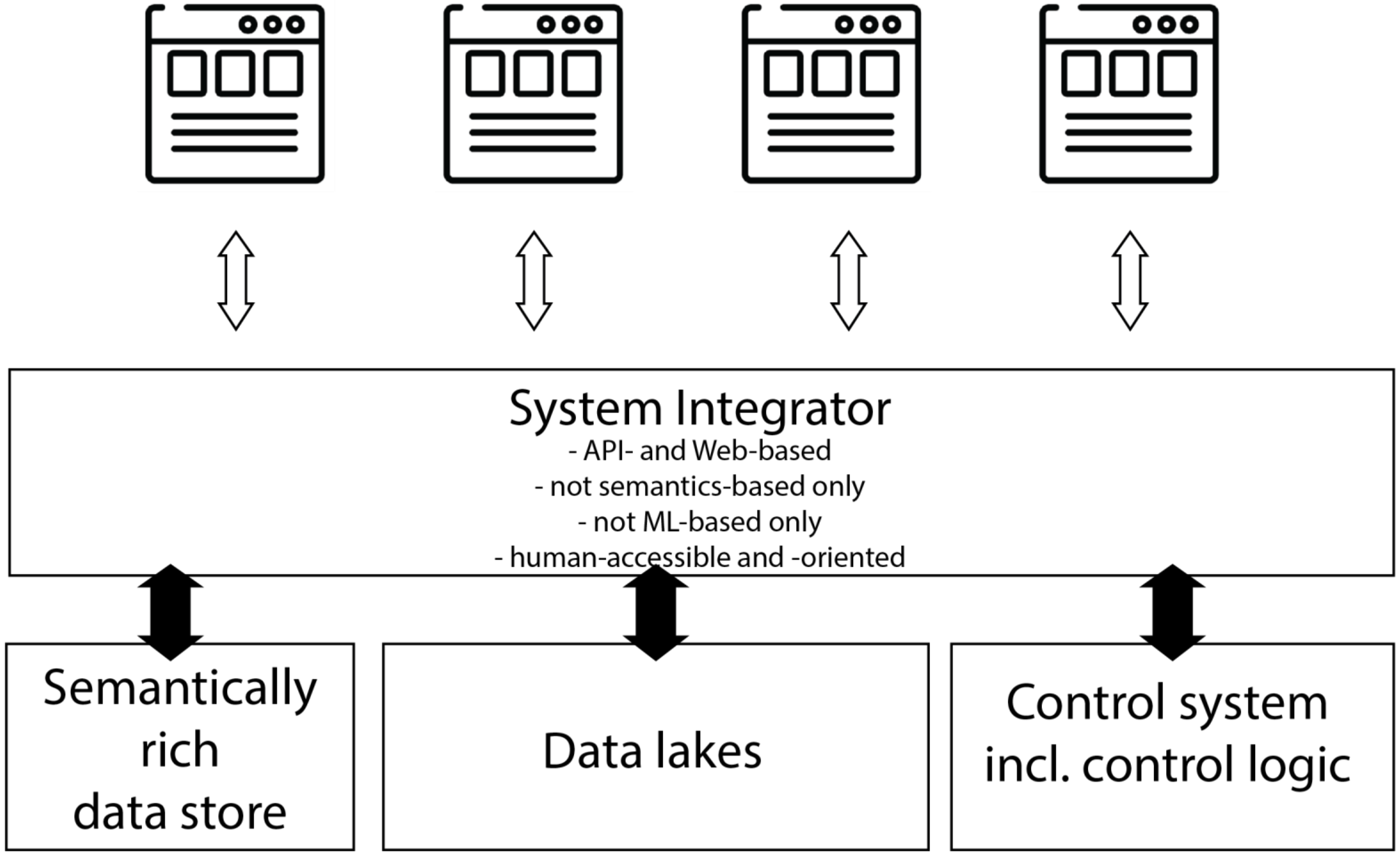
The Scale of AI Methods



Presentation Outline

1. A reference system architecture
2. Core metadata schemas: BIM, IFC, LBD, BRICK, Haystack, REC, SSN
3. Connecting to streamed data, edge communication, 3D and control logic

Targeted framework for AI-based smart buildings



SYMBOLIC AI **STATISTICAL AI**

Important:

- Include access control (ACL)
- 'linking' of data on system integration level
- Agreement and standardization of labels and metadata tags
- Feed back into control systems!

Native data format

SQL

Linking data in server back-end across different types of databases

Data management in application back-end

Native data format

RDF

End user application back-end

Native data format

RDF

Data Access Layers (REST APIs)

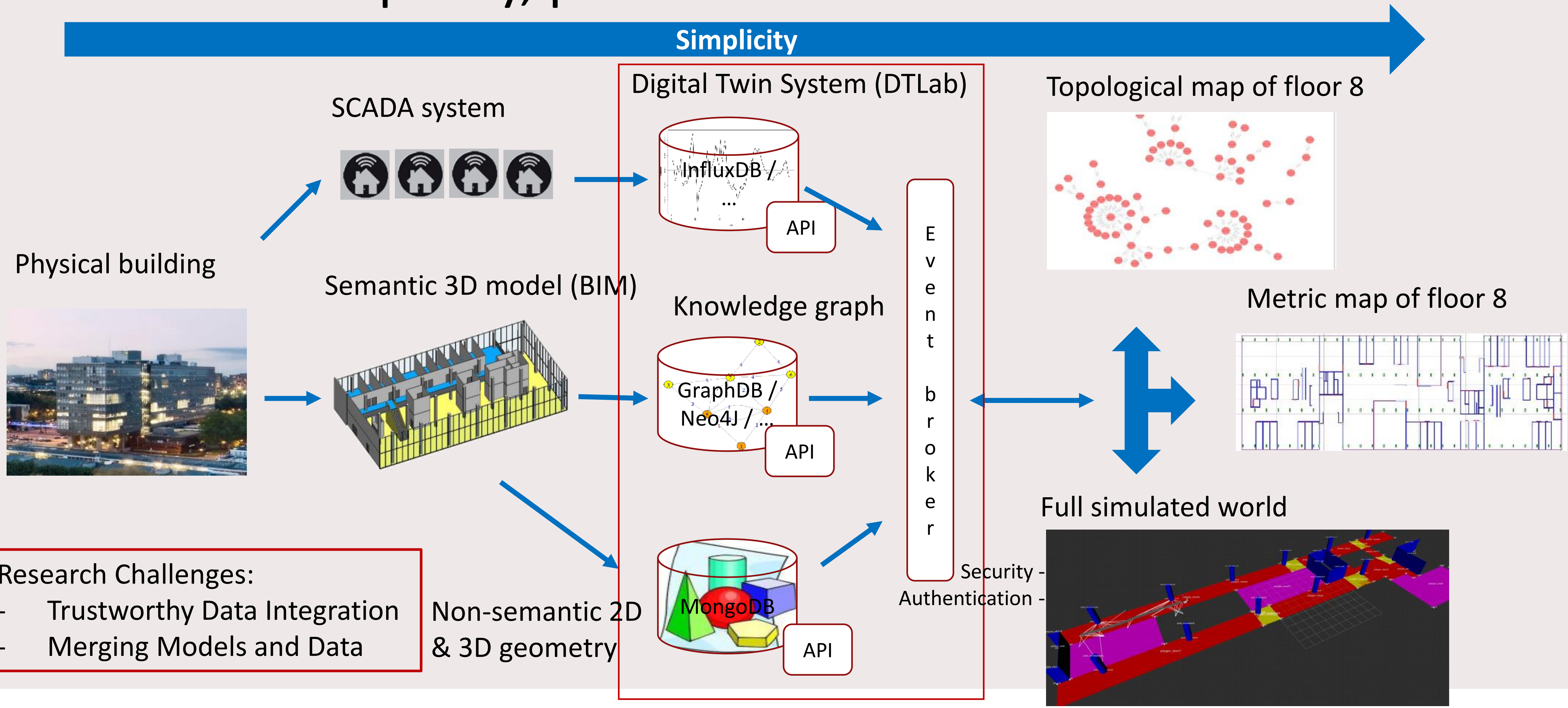
End user application back-end

Standard web technologies (ACL)

Native data format

PCD

Increase simplicity, preserve semantics



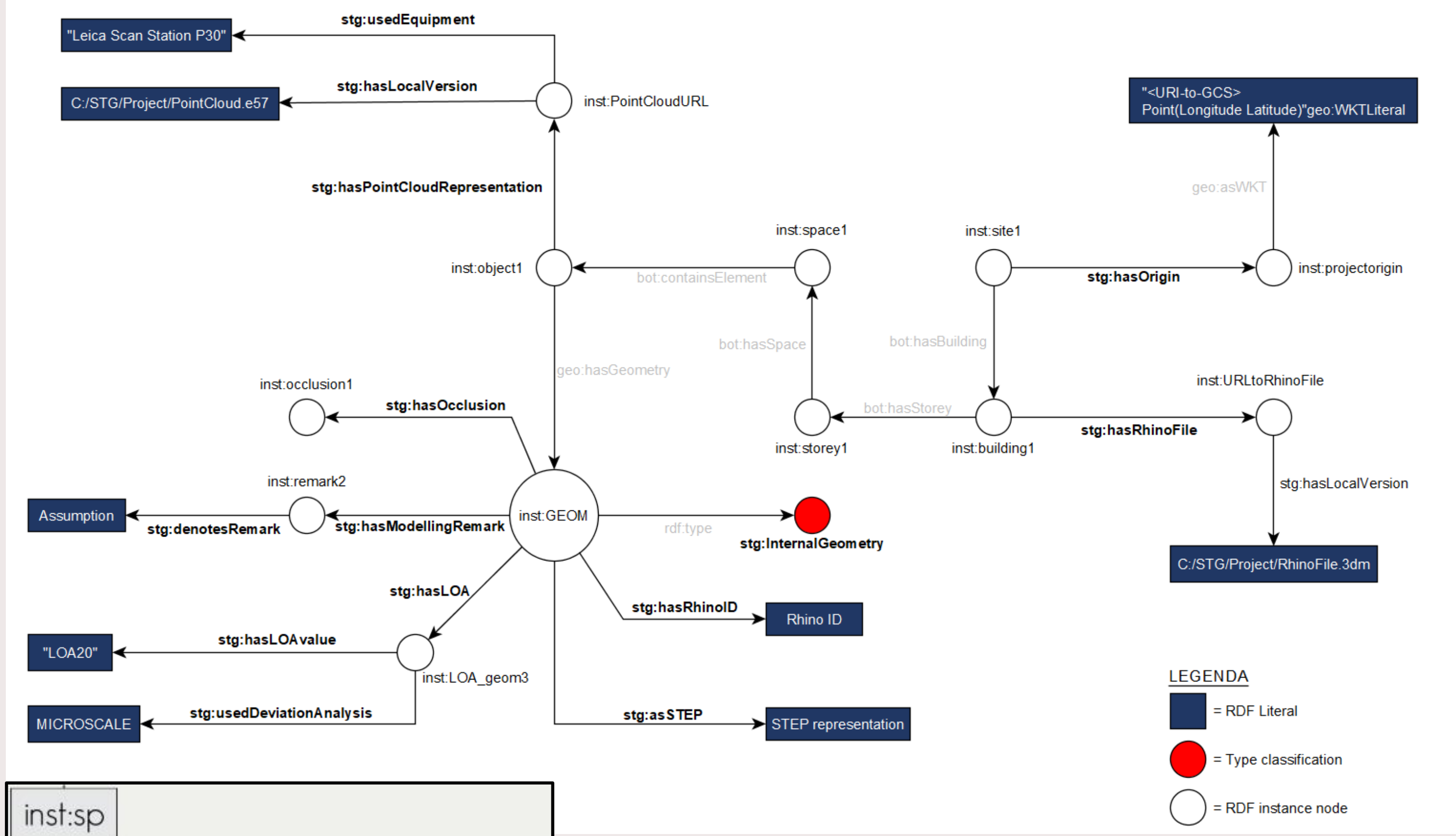
Research Challenges:
- Trustworthy Data Integration
- Merging Models and Data

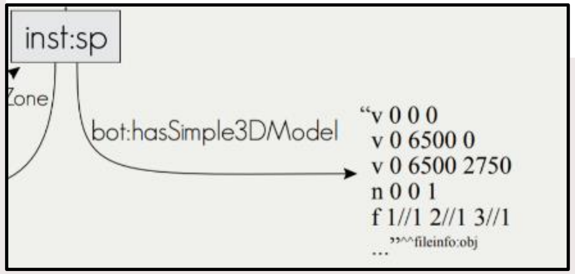
SEMANTIC GRAPH ENRICHED WITH PERFORMANCE PATTERNS AND WEB REFERENCE TO SENSOR DATA

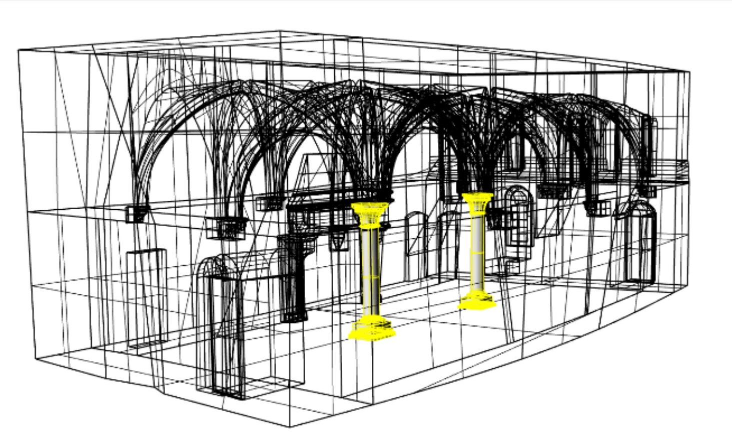
```
inst:room_16
	rdf:type bot:Space ;
	gig:hasSensorNode inst:sensorNode_0000014 ;
	gig:spaceType "Cafe" ;
	rdfs:label "Cafe" .

inst:sensorNode_00000014
	rdf:type gig:SensorNode ;
	rdfs:label "00000014" ;
	gig:observation "Indoor climate" ;
	gig:purpose "Thermal comfort in the lobby during big events when there is a gathering of a lot of people." ;
	sosa:hosts inst:sensor_00000014_1, inst:sensor_00000014_2, inst:sensor_00000014_3, inst:sensor_00000014_4, inst:sensor_00000014_5,
inst:sensor_00000014_6 ;
	gig:placement "Placed on a column in the cafe without direct sunlight." .

inst:sensor_00000014_1 ;
	rdf:type sosa:Sensor ;
	sosa:madeObservation inst:observation_1 ;
	sosa:observes inst:obsProperty_1 ;
	rdfs:label "00000014_1" .

inst:result_1 rdf:type sosa:Result ;
	rdfs:label "Result of observation of Relative Humidity";
	gig:values "https://gigantium.dk/Gigantium2018instances?orgId=1&datastream=true" .
```

Petrova, E., Pauwels, P., Svidt, K., Jensen, R.L. (2018) From patterns to evidence: Enhancing sustainable building design with pattern recognition and information retrieval approaches. Proceedings of the 12th ECPPM conference, pp. 391-399.

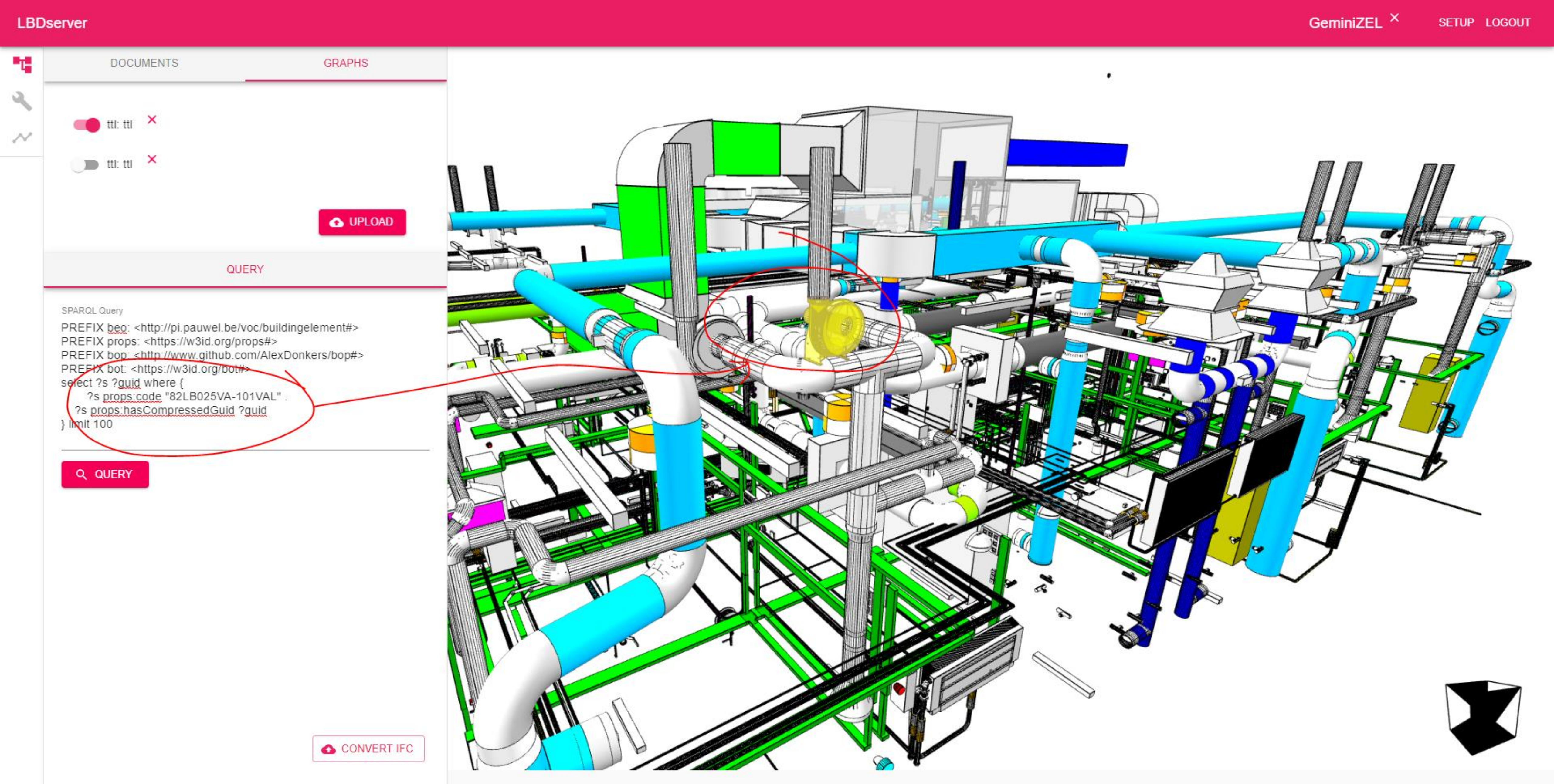
Abox linking to point cloud data and geometry







Jeroen Werbrouck et al., Scan-to-graph: Semantic enrichment of existing building geometry, Automation in Construction, 119 (2020). https://doi.org/10.1016/j.autcon.2020.103286.


LBDserver
GeminiZEL
SETUP
LOGOUT
DOCUMENTS
GRAPHS
ttl: ttl
ttl: ttl
UPLOAD
QUERY
SPARQL Query
PREFIX beo: <http://pi.pauwel.be/voc/buildingelement#>
PREFIX props: <https://w3id.org/props#>
PREFIX bop: <http://www.github.com/AlexDonkers/bop#>
PREFIX bot: <https://w3id.org/bot#>
select ?s ?guid where {
?s props:code "82LB025VA-101VAL" .
?s props:hasCompressedGuid ?guid
} limit 100
QUERY
CONVERT IFC

Presentation Outline

1. A reference system architecture
2. Core metadata schemas: BIM, IFC, LBD, BRICK, Haystack, REC, SSN
3. Connecting to streamed data, edge communication, 3D and control logic

Take-aways to conclude

- Structure your data
- Make agreements about how you structure your data
- Be critical towards acronyms and look beyond them
- Store your data in dedicated technologies
- Aim for (web-ready) system integration