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)
Brains for buildings: where to find all the relevant smart building data?

Buildings are to serve people’s needs:
- Occupants and FM: Health, comfort, ease of use & ease of operation, affordability
- Humankind: energy efficiency, renewables

Building operation is key (Energy & Indoor climate systems)
- Lots of occupants & FM dissatisfaction
- Lots of energy wastage

Operation data & data analytics, ML, AI are key to:
- Understand
- Steer & Control optimally
- Adapt to renewable energy
- Make better designs

April 1, 2021
Consortium Meeting
Presentation Outline

1. Stepping through Building Data Semantics: BIM, IFC, LBD, BRICK, Haystack, etc.
2. The ‘Scale of AI Methods’: ML and semantics
3. Towards scalable system integration that combines multiple AI fields
Building data, building data, and then more building data

Building Information Model (BIM)
3D representation enriched with semantic information

Digital Twin (DT)
Digital counterpart for a physically existing object

Linked Building Data (LBD)
Set of interlinked web-based data about the built environment

Gemini Digital Twin: combining building data with sensor data (A. Pelt-Thissen)
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels

- 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

Cloud of linked data

Tim Berners-Lee, the inventor of the Web and Linked Data Initiator, suggested a 5-star deployment scheme for Open Data. Here, we give examples for each step of the stars and explain costs and benefits that come along with it.

https://5stardata.info/en/
Cloud of linked building data

Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels
A building has different types of data associated.
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels
BIM data: Revit, modelling guidelines, agreements, 3D modelling, and IFC
Luckily we have IFC...

**STEP (ISO 10303)** can represent 3D objects in Computer-Aided design (CAD) and related information.

**IFC-SPF (STEP Physical File Format)** - Text format defined by ISO 10303-21 ("STEP-File"), where each line consists of a single object record.

**Extensible Markup Language (XML)** is a markup language designed to store and transport data.

**Resource Description Framework (RDF)** is a standard model for data interchange on the Web.

**EXPRESSION** is a data definition standard developed to enable a formal definition of industrial data.

**XSD (XML Schema Definition)** specifies how to formally describe the elements in a XML document.

**ifcOWL** provides a Web Ontology Language (OWL) representation of the IFC schema.

Conversions aiming at roundtripping and backwards compatibility.
Data in the Industry Foundation Classes (IFC)

- Overall building shape and topology easy
- Classification of elements possible, but not many classes => extension with bSDD classes and properties possible
- Difficult (not impossible) to include sensor data (timeseries data)
- Availability in STEP, XML, RDF, and JSON
The difficult parts – and ... more important findings

Challenging to convert from EXPRESS:
- Ordered collections: Lists and Arrays
- Geometry
- WHERE rules (WR)
- Functions
- SELECT data types
- PSETs

More important requirements:
1. Modularisation
2. Extensibility
3. Simplified access
W3C Linked Building Data
Emergence of W3C LBD Community Group: Mission Statement

Bring together experts in the area of Building Information Modeling (BIM) and Web of Data technologies to:

1. define existing and future **use cases and requirements** for Linked Data based applications across the life cycle of buildings.
2. discuss **best practices** for publishing building data on the Web propose ontology models to describe:
   1. Buildings and building elements (topology, associate values to properties)
   2. Products and product properties
3. discuss how they can be **used together with other specifications**:
   1. existing standards (IFC, GeoSPARQL, Semantic Sensor Network, ...)
   2. separate initiatives (schema.org, Haystack, BRICK, ...)
Scope of BOT: just the start, allowing to extend

- Limited set of classes
- Extensible and easy to combine with other ontologies and data sets
- Comprehensible
Product taxonomies are in separate vocabularies to increase reusability and modularity.
Modular ontology modelling advocated by LBD group

- Implemented using Semantic Web Technologies -> **Web-scale, queryable**
- Reuse of existing ontologies -> **Modular**
- **Linking at instance level** -> Multi-model method

Sample dataset available at:

Modular approach to building data

Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels
Reference ontologies

BOT: https://w3id.org/bot#
BEO: https://pi.pauwel.be/voc/buildingelement/
MEP: https://pi.pauwel.be/voc/distributionelement/
OMG: https://w3id.org/omg#
FOG: https://w3id.org/fog#
BPO: https://www.w3id.org/bpo#
OPM: https://www.w3id.org/opm#

Revit to LBD exporter: on demand
IFC to LBD converter: on demand
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels

BRICK, HTO, SAREF, and REC
Brains for buildings: where to find all the relevant smart building data?

- Developments are rather disconnected from any BIM-or building-related area
- Focus on systems, incl. operation and control
- Focus on the sensor Point and Equipment types
- Beware of biased overview tables

https://brickschema.org/
BRICK - A uniform metadata schema for buildings

https://brickschema.org/
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels
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/](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.
RDF encoding / mapping example

URI

def: ^lib:phIoT
doc: "Project Haystack definitions for Internet of Things"
version: "4.0"
baseUrl: `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"

Instances (data)

id:@24192ca1-0c85f75d "Headquarters" site
area: 140797 Ft²
tz: New York
dis: Headquarters
goAddr: "600 W Main St, Richmond, VA"
geoCoord: C(37.545826, -77.449188)
hq metro: Richmond
primaryFunction: Office
yearBuilt: 1999
### Extract from BMS (labels):

<table>
<thead>
<tr>
<th>Date</th>
<th>Item</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
<th>Value 6</th>
<th>Value 7</th>
<th>Value 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/01/2021</td>
<td>11NR008FT-011FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>280</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-031FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-012FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-032FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-011FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>280</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-031FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-012FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-032FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-011FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>280</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-031FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-012FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-032FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-011FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>280</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-031FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-012FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>11NR008FT-032FLW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Column</th>
<th>ItemName</th>
<th>ItemDescriptionDutch</th>
<th>ItemDescriptionEnglish</th>
<th>Spaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>6358</td>
<td>11NR009TE-033TRL</td>
<td>RUIMTETEMPERATUUR 9_Z01</td>
<td>ROOM TEMPERATURE 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>6359</td>
<td>11NR009TE-030TRL</td>
<td>RUIMTETEMPERATUUR 9_Z01</td>
<td>ROOM TEMPERATURE 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>2089</td>
<td>11NR009TE-030TRLBP</td>
<td>ACTUELLE SETPOINT REGELING 9_Z01</td>
<td>CURRENT SETPOINT SCHEME 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>6821</td>
<td>11NR009SCR030BNDRS</td>
<td>STAND ZONERING 9_Z01</td>
<td>STAND AWNINGS 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>5900</td>
<td>11NR009CV-030RWWRS</td>
<td>REGELSIGNAAL VERWARMING 9_Z01</td>
<td>CONTROL SIGNAL HEATING 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>5899</td>
<td>11NR009CV-033RWWRS</td>
<td>REGELSIGNAAL VERWARMING 9_Z01</td>
<td>CONTROL SIGNAL HEATING 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>4349</td>
<td>11NR009TE-033MAXBP</td>
<td>KOELINGSETPONT 9_Z01</td>
<td>COOLING SETPOINT 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>4350</td>
<td>11NR009TE-030MAXBP</td>
<td>KOELINGSETPONT 9_Z01</td>
<td>COOLING SETPOINT 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>3568</td>
<td>11NR009TE-030CPA</td>
<td>VIA WANDMODULE RUIMTE 9_Z01</td>
<td>VIA WALL MODULE ROOM 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>3567</td>
<td>11NR009TE-033CPA</td>
<td>VIA WANDMODULE RUIMTE 9_Z01</td>
<td>VIA WALL MODULE ROOM 9_Z01</td>
<td>9_Z01</td>
</tr>
<tr>
<td>6823</td>
<td>11NR009SCR033BNDRS</td>
<td>STAND ZONERING 9_Z01</td>
<td>STAND AWNINGS 9_Z01</td>
<td>9_Z01</td>
</tr>
</tbody>
</table>
Real Estate Core


Hartig et al., 2019
SAREF, SOSA and SSN

Example:

```turtle
@prefix cdt: <http://w3id.org/lindt/custom_datatypes#> .
BASE <http://example.org/> .
<observation/235715> a sosa:Observation ;
  sosa:hasFeatureOfInterest <house/134/kitchen> ;
  sosa:observedProperty <electricConsumption> ;
  sosa:madeBySensor <sensor/927> ;
  sosa:hasSimpleResult "22.4 kWh"^^cdt:ucum .

<observation/235715> a sosa:Observation ;
  sosa:resultTime "2017-11-15T14:35:13Z"^^xsd:dateTime ;
  sosa:hasSimpleResult "23.9 DEG"^^cdt:temperature .
```

Ontology Core:

Presentation Outline

1. Stepping through Building Data Semantics: BIM, IFC, LBD, BRICK, Haystack, etc.
2. The ‘Scale of AI Methods’: ML and semantics
3. Towards scalable system integration that combines multiple AI fields
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.
Strengths and methods in Artificial Intelligence

DATA STREAMS

- **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

**MACHINE LEARNING**

**STATISTICAL AI METHODS**

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

**SEMANTICS**

**SYMBOLIC AI METHODS**

Semantically rich and interconnected data
The Scale of AI Methods

STATISTICAL AI

Most Smart Buildings

SYMBOLIC AI
The Scale of AI Methods

Can we integrate these methods and use all of them?
Semantic encoding of buildings
The Scale of AI Methods

Statistical AI

Symbolic AI

Plain labels: SAREF, SOSA, SSN

Haystack: REC

IFC

BRICK: LBD

Tagging: META data

Most Smart Buildings

Metadata: LINKING DATA

Semantic: SEMANTICS
And all of this excludes control systems and control logic!!
Presentation Outline

1. Stepping through Building Data Semantics: BIM, IFC, LBD, BRICK, Haystack, etc.
2. The ‘Scale of AI Methods’: ML and semantics
3. Towards scalable system integration that combines multiple AI fields
Available options for integration in trustworthy manner

**OPTION 1: Transform all into semantic graphs** (e.g. R2RML or custom data transformers) and do data integration

- Plus: all in same format
- Plus: inference possible
- Minus: unfit storage
- Minus: disconnect from origin
- Minus: no ML algorithms nor procedural code possible
- Minus: how to handle privacy and security (trust?)
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels

- End user application development
- ETL procedure
- Linking data
- Native data format
- RDF stores
- RDF stores
- Data management in application back-end
- Native data format
- Native data format
- Native data format

45
Brains for buildings: where to find all the relevant smart building data? - Pieter Pauwels

Native data format → ETL procedure → RDF stores → Data management in application back-end

- Disconnect from source
- Specialised software and data handling lost

Linking data → RDF stores

Security and data access protocols?

End user application development
Available options for integration in trustworthy manner

**OPTION 2:** Store all in well-fit data stores (KV stores, graphDBs, relational DBs, timeseries stores, etc.) and perform data integration (also) on a system and API level (system integration)

- Plus: apt data storage
- Plus: data stays at source -> web-based connections needed
- Plus: ML algorithms and procedural algorithms not blocked
- Plus: Privacy and security can be easily handled at the gates of APIs and DBs.
- Minus: multitude of systems requires lots of diverse software and expertise
Brains for buildings: where to find all the relevant smart building data?

- Pieter Pauwels

Native data format

SQL

RDF

RDF

PCD

Data Access Layers (REST APIs)

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

Data management in application back-end

End user application back-end

Native data format

Native data format

Native data format

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

Data management in application back-end

End user application back-end

Standard web technologies (ACL)
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!

System Integrator
- API- and Web-based
- not semantics-based only
- not ML-based only
- human-accessible and -oriented

Semantically rich data store
Data lakes
Control system incl. control logic

SYMBOLIC AI
STATISTICAL AI
Presentation Outline

1. Stepping through Building Data Semantics: BIM, IFC, LBD, BRICK, Haystack, etc.
2. The ‘Scale of AI Methods’: ML and semantics
3. Towards scalable system integration that combines multiple AI fields
The Scale of AI Methods

Plain labels → STATISTICAL AI

Haystack → TAGGING

REC → METADATA

SAREF, SOSA, SSN → LINKING DATA

LBD → SEMANTICS

BRICK → IFC

IFC

Most Smart Buildings
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