BHoM.xyz

bhOWL: BHoM and Semantic Web Technologies

Alessio Lombardi

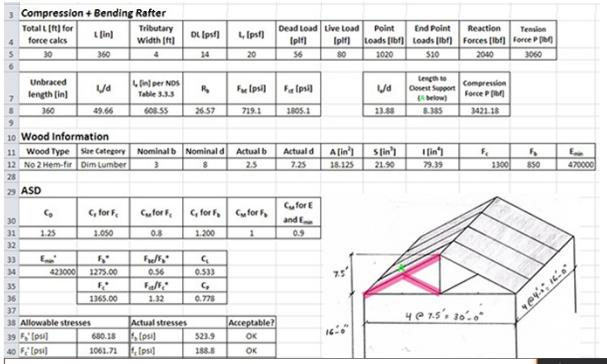
Interoperability and Machine Learning lead @ Burohappold, London UK

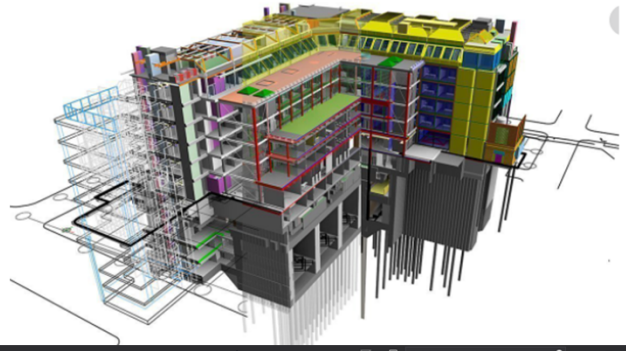
LDAC 2023

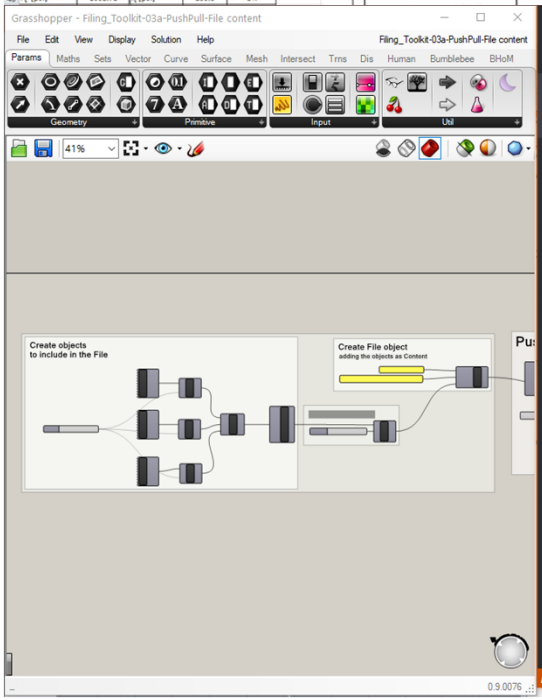
AEC design challenges

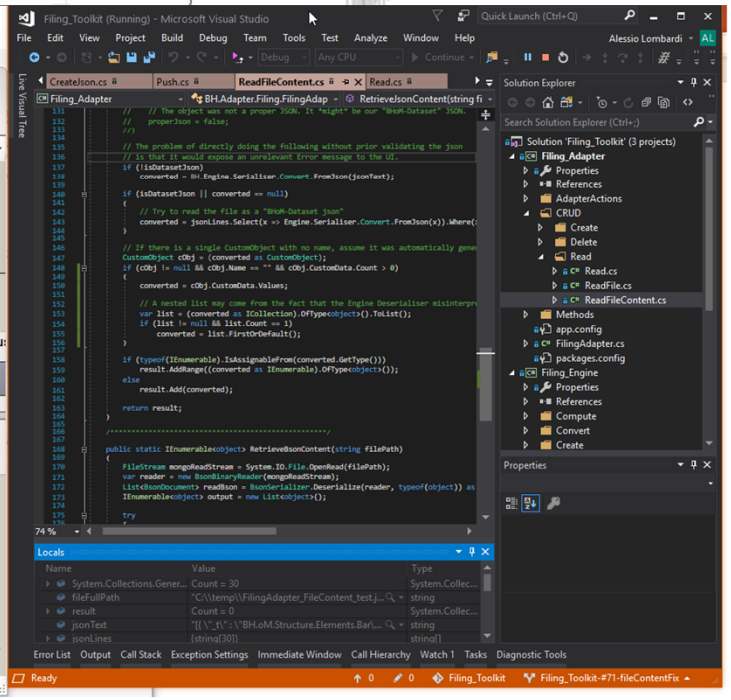
Complex projects and Co-creation using modern tools

BHoM.xyz





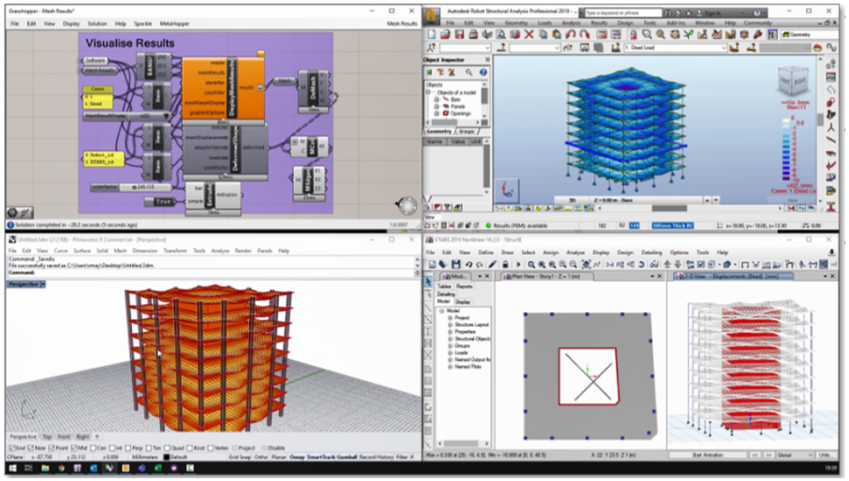


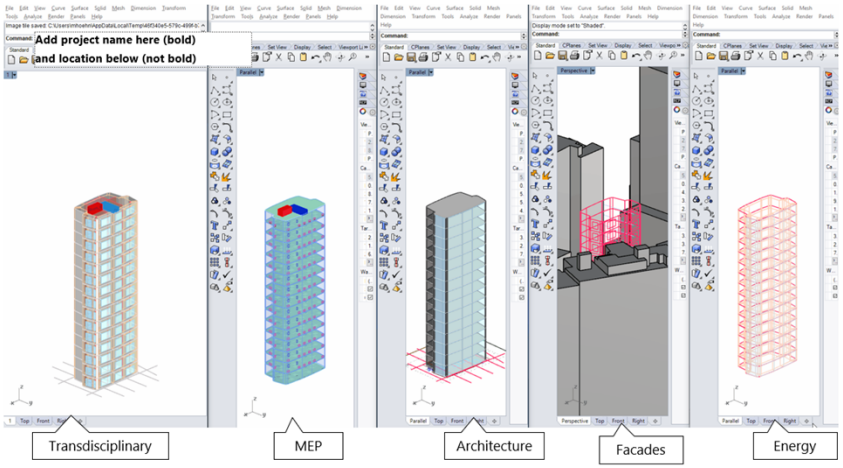


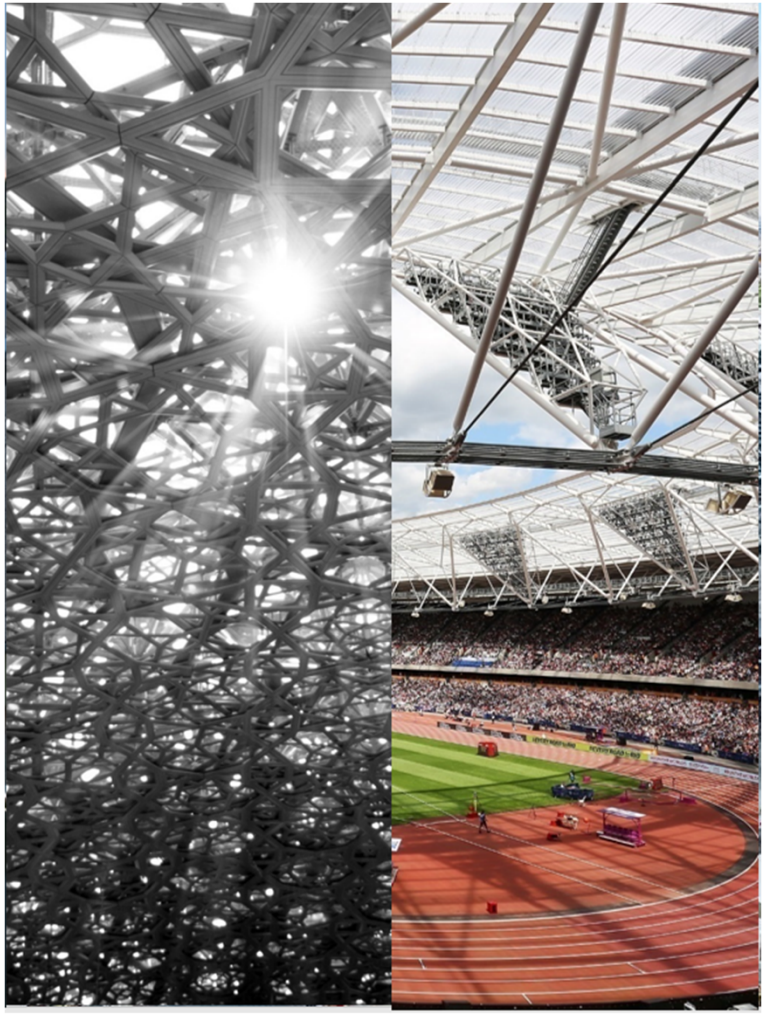
AEC design challenges

Enabling even more complex projects

BHoM.xyz





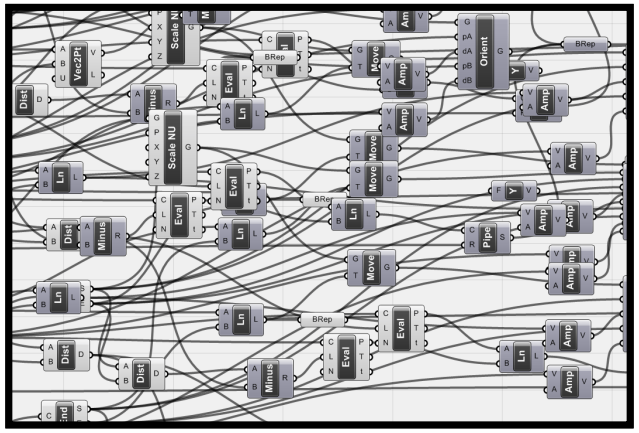


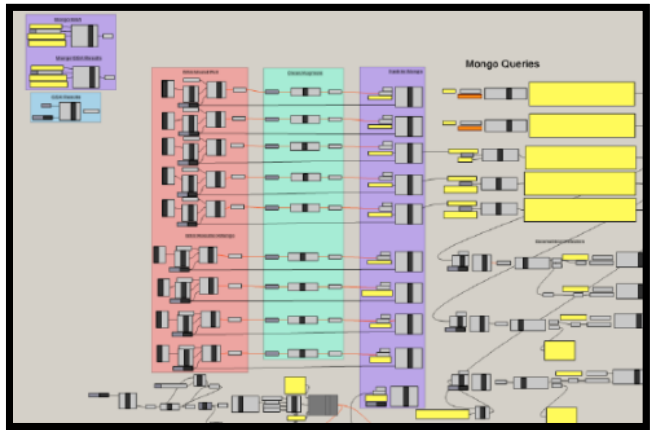
By co-designing and reusing previous work, we can handle more complex projects

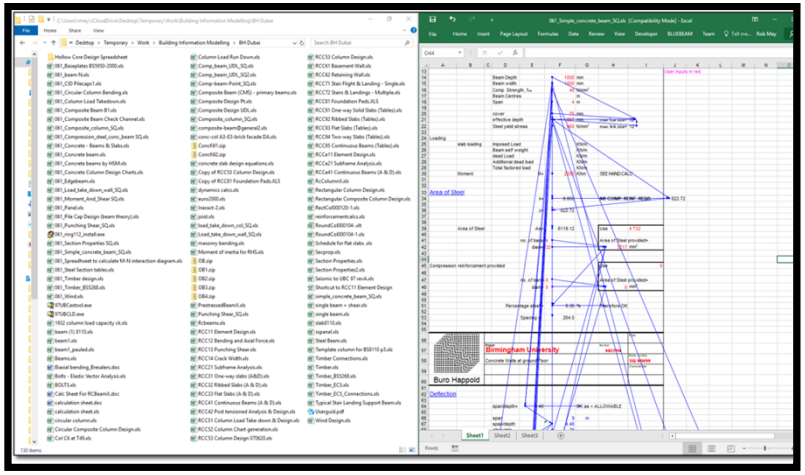
AEC design challenges

Reusing previous work

BHoM.xyz



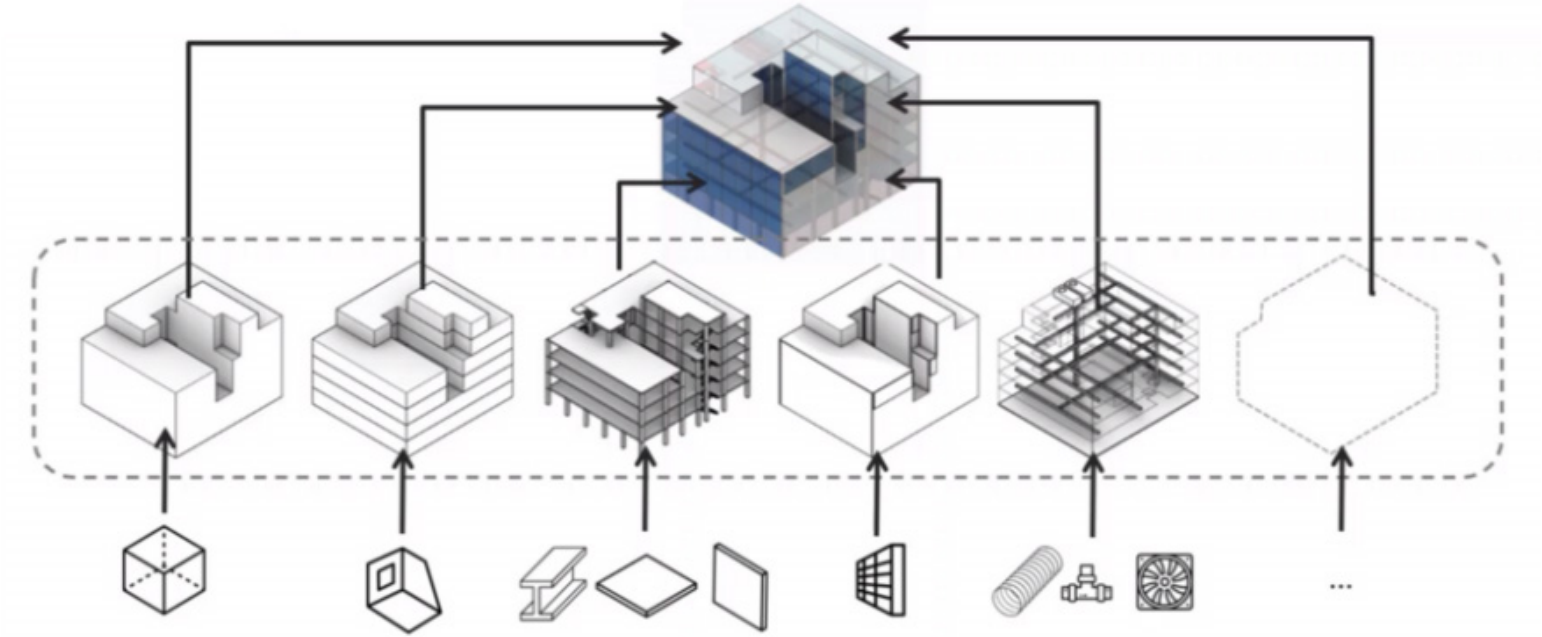




Can any of these
be reused
on future projects?

AEC design challenges

Multi-disciplinarity of AEC projects

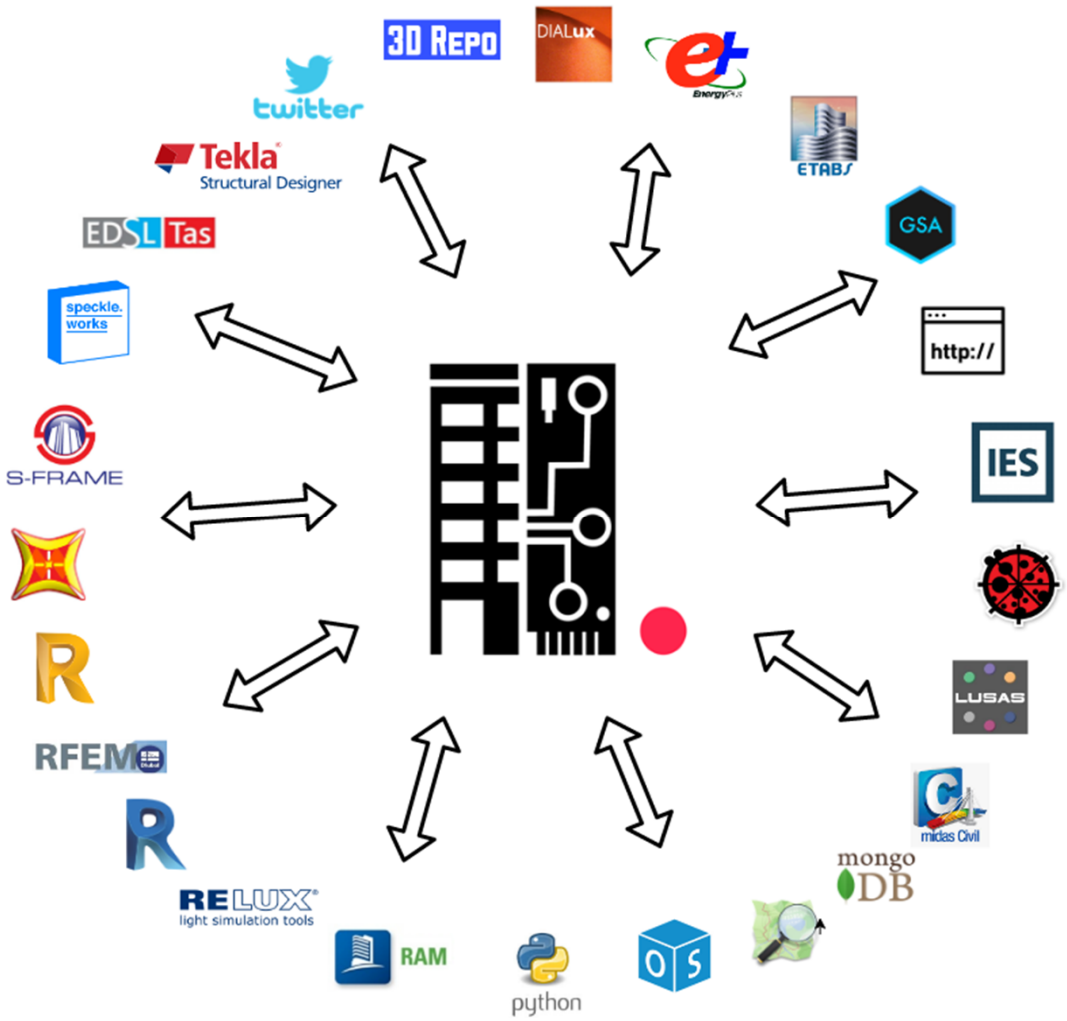


Multiple discrete representations of
a (“to-be”) physical asset

BHoM's approach:
domain experts-defined design language

BHoM ecosystem

BHoM.xyz



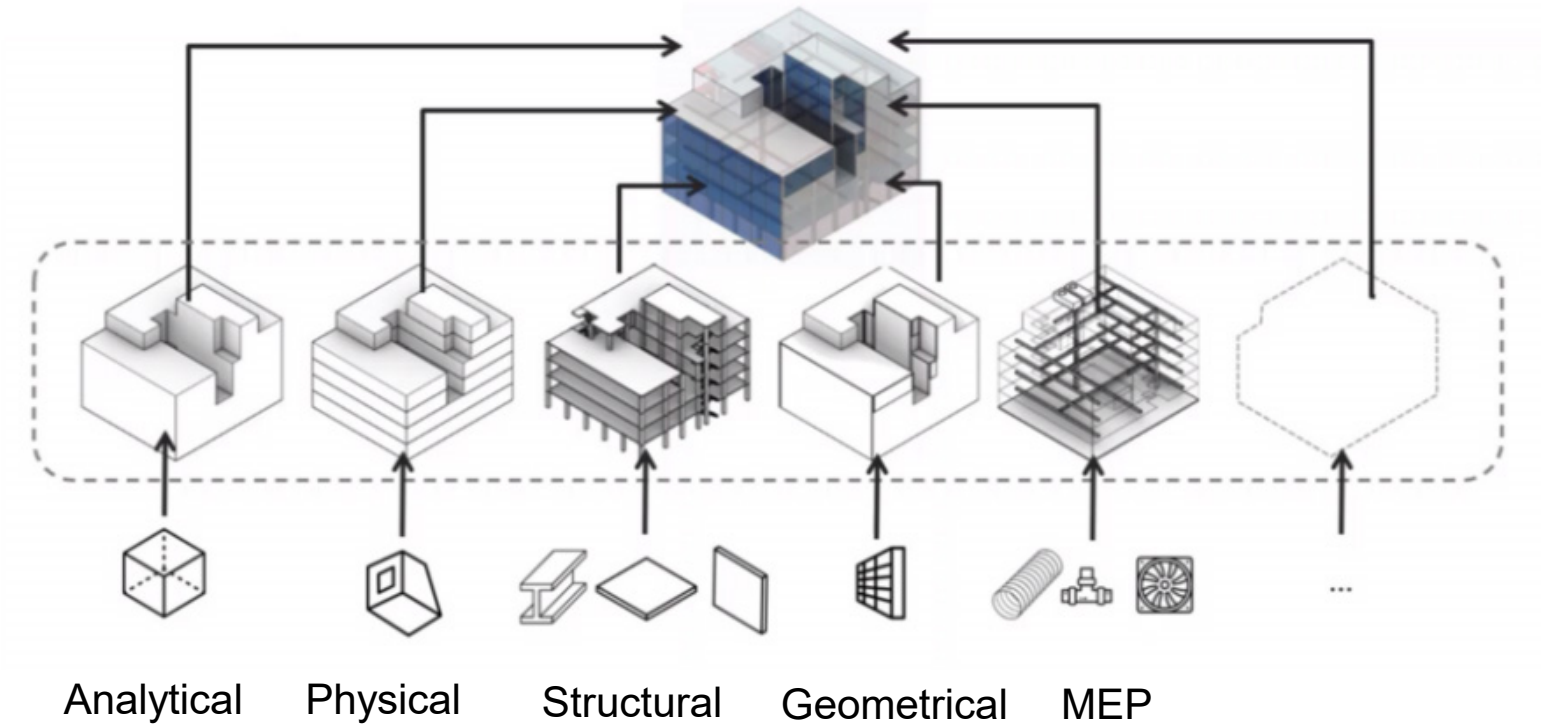
- A central, **federated** schema:
  the **object Model (oM)**
  (our main language)
  organised in "domains" of expertise
- "*Adapters*" that **convert *BHoM format*** to/from **external software format**
  (to understand every language)
- When an adapter is written for a software, it becomes part of the BHoM ecosystem
- **29 Open Source Adapters** at the moment (and growing)

BHoM's approach to multidisciplinarity

Let the industry drive categorisation

BHoM.xyz

The object model is *organically grown from the roots:*
domain experts define ***the concepts they need for work*** and ***in their own namespace***



BHoM's approach to multidisciplinarity

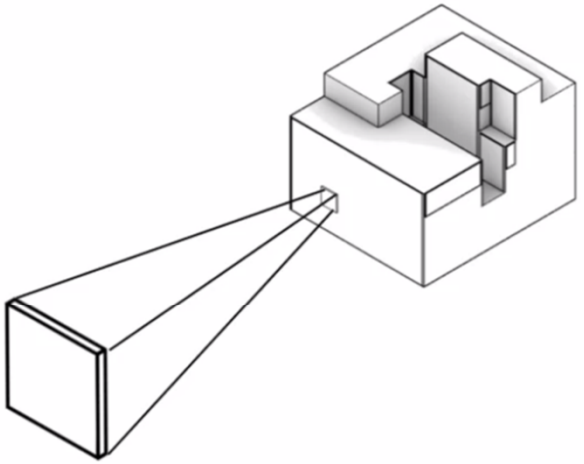
Let the industry drive categorisation

BHoM.xyz

As a result, there isn't a unique way to represent a concept.
The important thing is to acknowledge this!

Common object Model (oM)
How to represent a "Wall"

Physical	BH.oM.Physical.Elements.Wall
Analytical	BH.oM.Facade.Elements.Panel BH.oM.Structure.Elements.Panel BH.oM.Environment.Elements.Panel BH.oM.Acoustic.Elements.Panel
Geometrical	BH.oM.Geometry.PlanarSurface BH.oM.Geometry.NurbsSurface BH.oM.Geometry.NurbsCurve
Graphical	BH.oM.Graphics.RenderMesh



Differences with other products

Comparison with other common schemas/tools/frameworks

BHoM.xyz

IFC

- Standardized schema for AEC data. Not open-source.
- **Data represented in a monolithic approach. Focus on "as-built"/"physical" concepts (e.g. no force/energy concepts)**
- Powerful in representing/ documenting element classification and product properties, but it falls when representing dynamic data.
- Around 1000 classes focused on "as-built"/"physical" concepts. Does not allow for flexibility.
- **Most suitable to give a "snapshot" of a design rather than exchanging data during design.**

Speckle

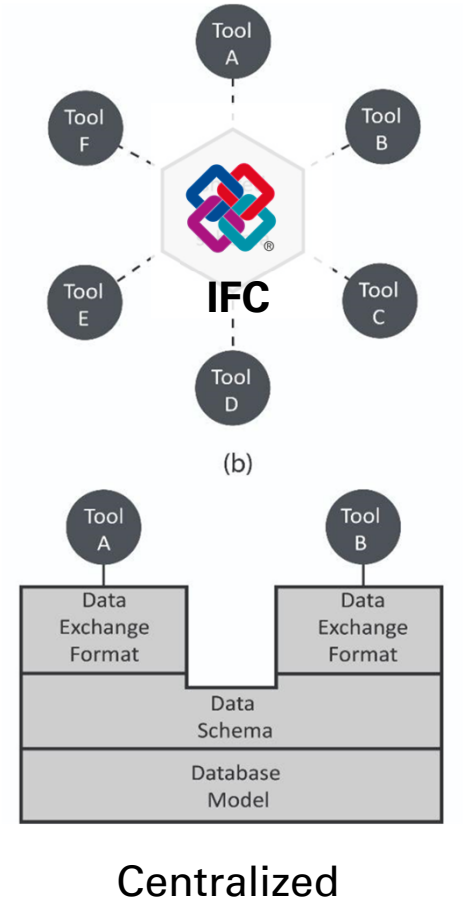
- Open-source, web and geometry-based AEC data exchange and versioning platform.
- **Does not focus on information modelling.** Offers a compact schema centered on geometry.
  There are few domain-specific classes spanning different domains.
- Around 150 object models, and numerous connectors to design software`.
- **Focuses on connecting software and exchanging data, rather than representing information.**

BHoM

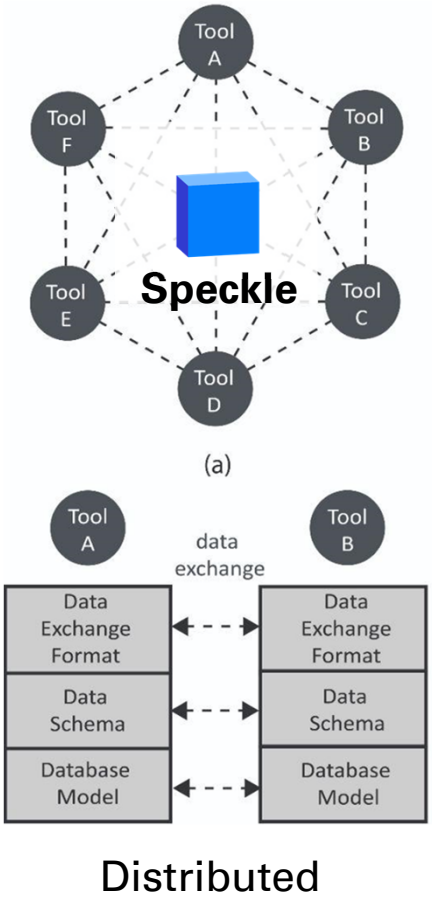
- Open-source framework for AEC information modelling and design tools.
- **Large schema that can include any concept useful in design**, from "physical" (e.g. columns) to "theoretical" (e.g. design forces) to geometrical (e.g. polylines).
- Over 1000 object models many different domains.
- **Also offers connectivity to exchange and convert data between domains and software.**

Differences with other products

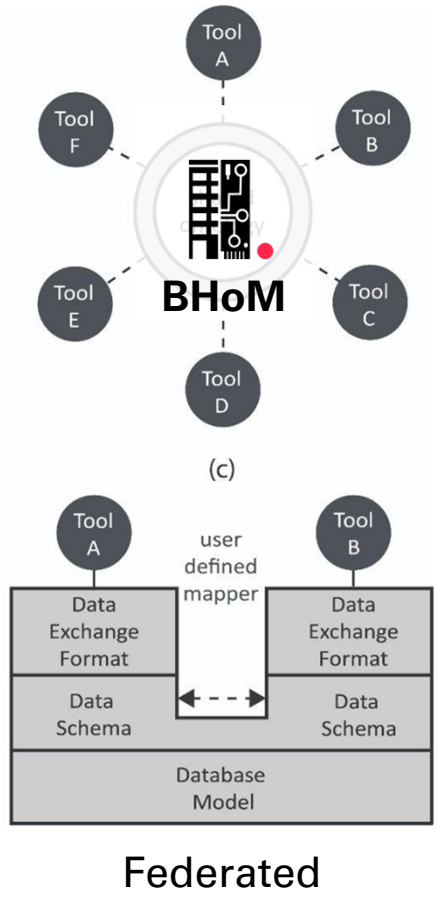
Comparison with other common schemas/tools/frameworks



Centralized



Distributed



Federated

There is a degree of centralisation in BHoM (base schemas), but the accent is on user-defined mappings rather than monolithically agreed-upon schema.

Toth, B., Janssen, P., Stouffs, R., Chaszar, A., & Boeykens, S. (2012). Custom digital workflows: a new framework for design analysis integration. International Journal of Architectural Computing, 481-500.

BHoM framework in a bit more detail

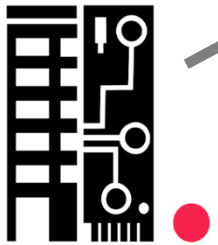
Separation of definitions and functionality

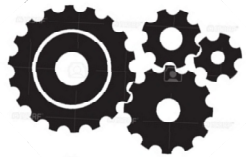
BHoM.xyz

The **object Model (oM)**

The central schema

contains all definitions as decided by domain experts





The **Engine**

Custom Functions

functionality built on the common schema (query, filter, calculate…)

The **Adapter**

Connects to/from other Software

A **Toolkit**

Packets an **oM,** an **Engine** and an **Adapter** **specific** to a **software** or **target analysis.**

All **exposed** in some ***User Interface software***

UI

e.g. Grasshopper, Excel, web applications, Revit plugin, etc…

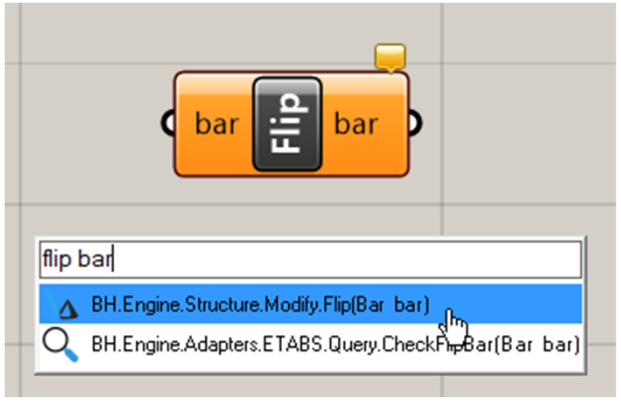
Using the BHoM

"Components" are always the same

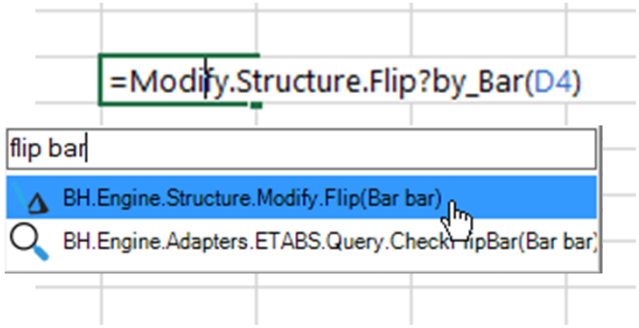
BHoM.xyz

Everything in BHoM can be used in the same way
no matter the UI:

A **Grasshopper component**



An **Excel formula**

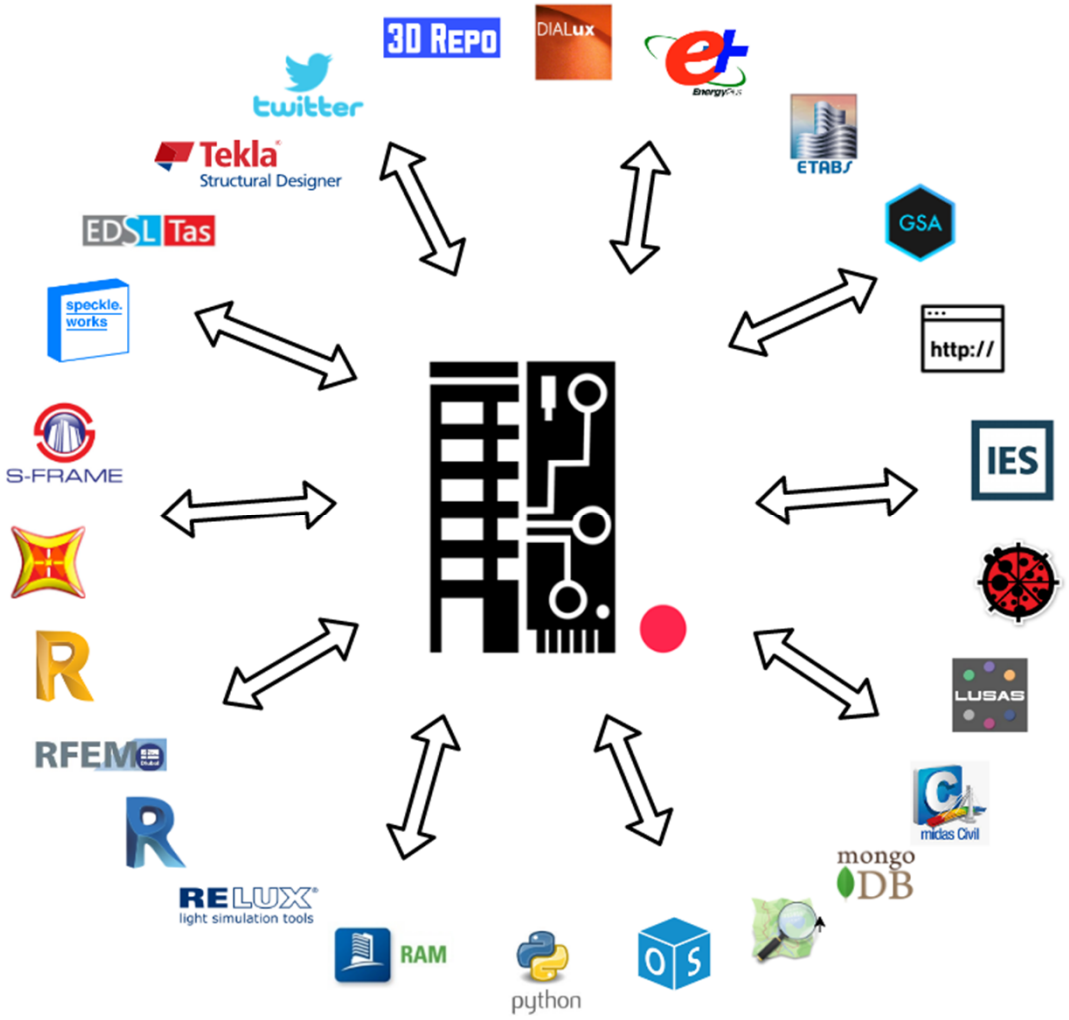


A **C# script**

```
Bar bar = new Bar();
BH.Engine.Structure.Modify.Flip(bar);
```

+ web apps, Revit plugins, etc...

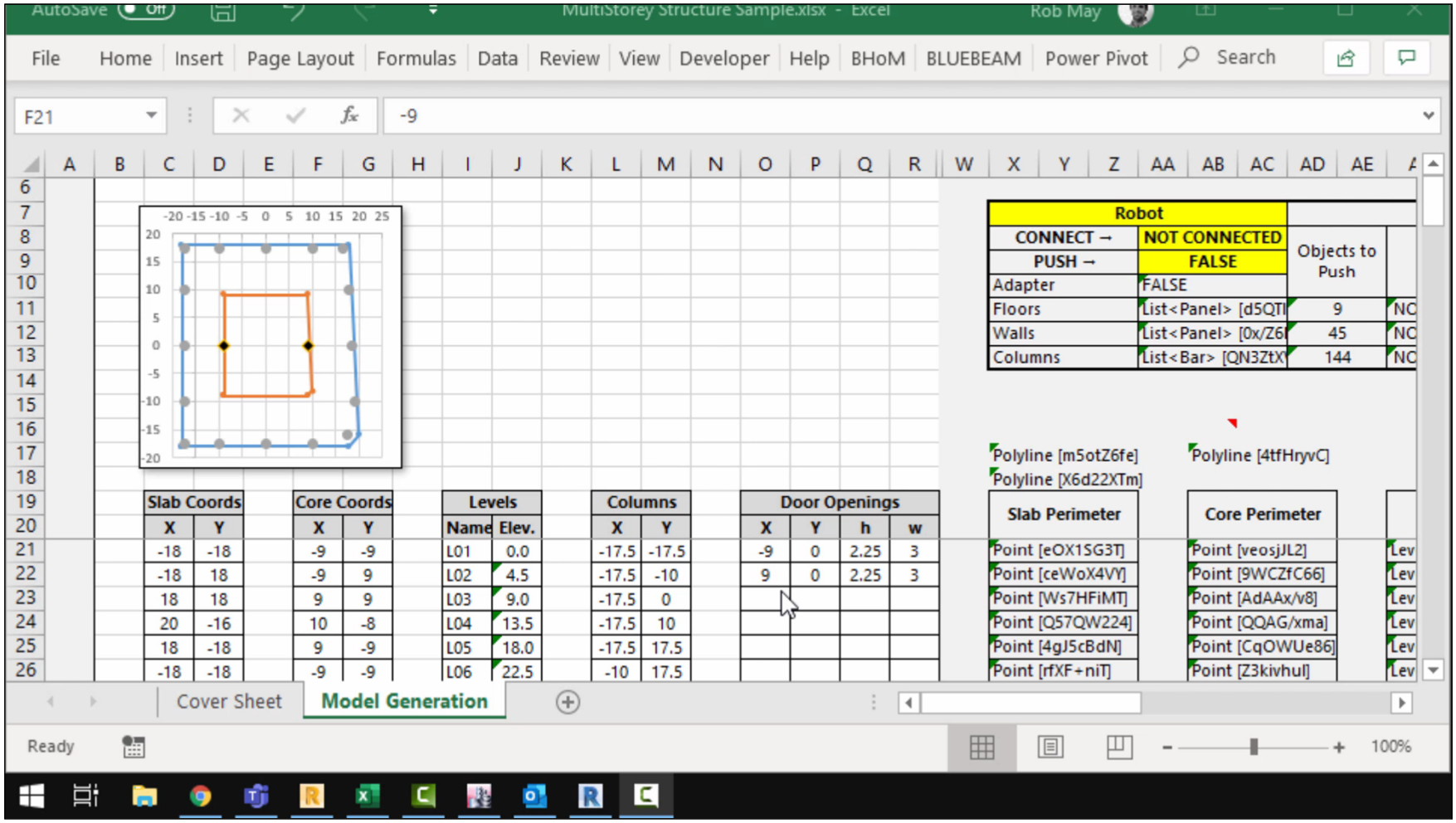
The Adapters



- Code that connects (**import/export**) BHoM models with other software
- Takes care of **converting** “BHoM format” towards the **external software format**
- **Over 30 Open Source Adapters** at the moment (and growing)

Examples:
Generative modelling beyond Grasshopper

BHoM.xyz

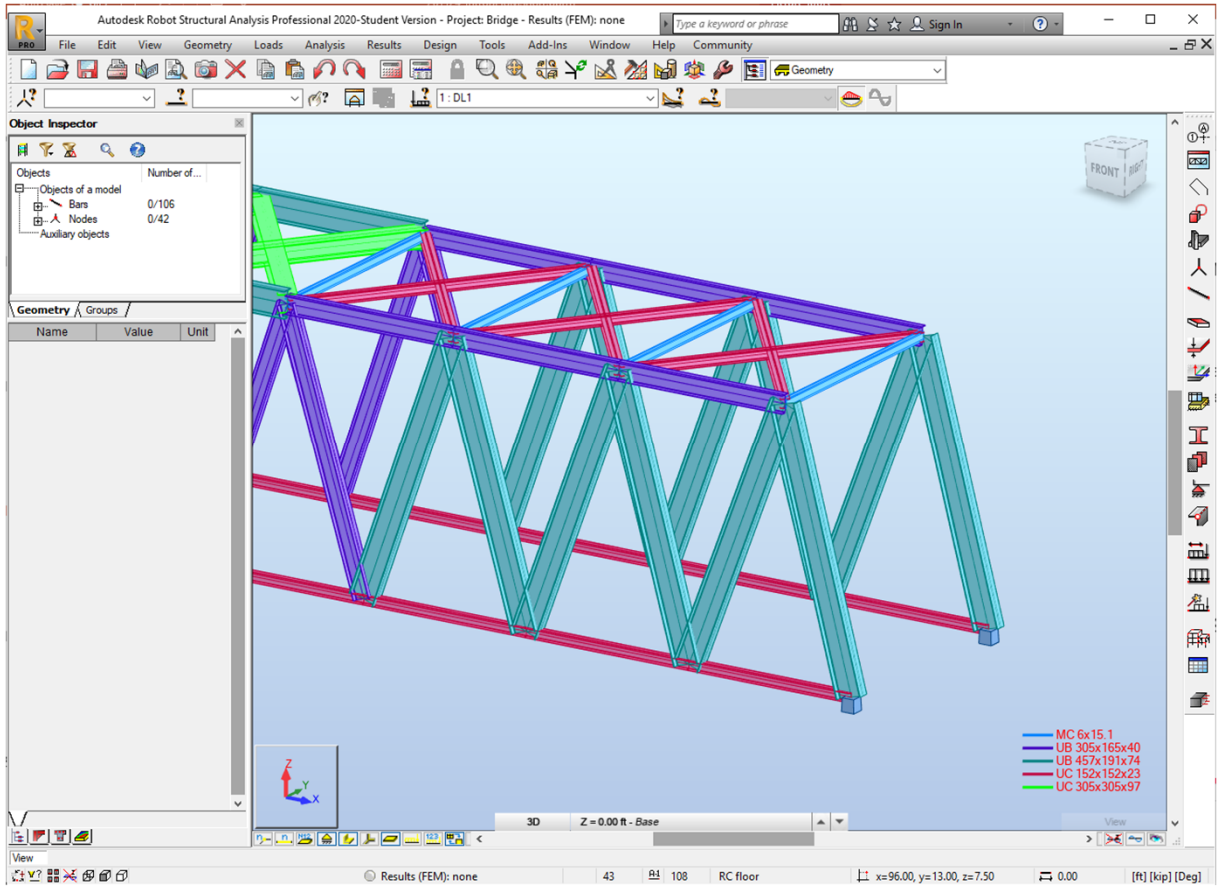


- All BHoM methods are available in Excel
- Can build geometry and objects using in-cell formulae and push out to any software

15

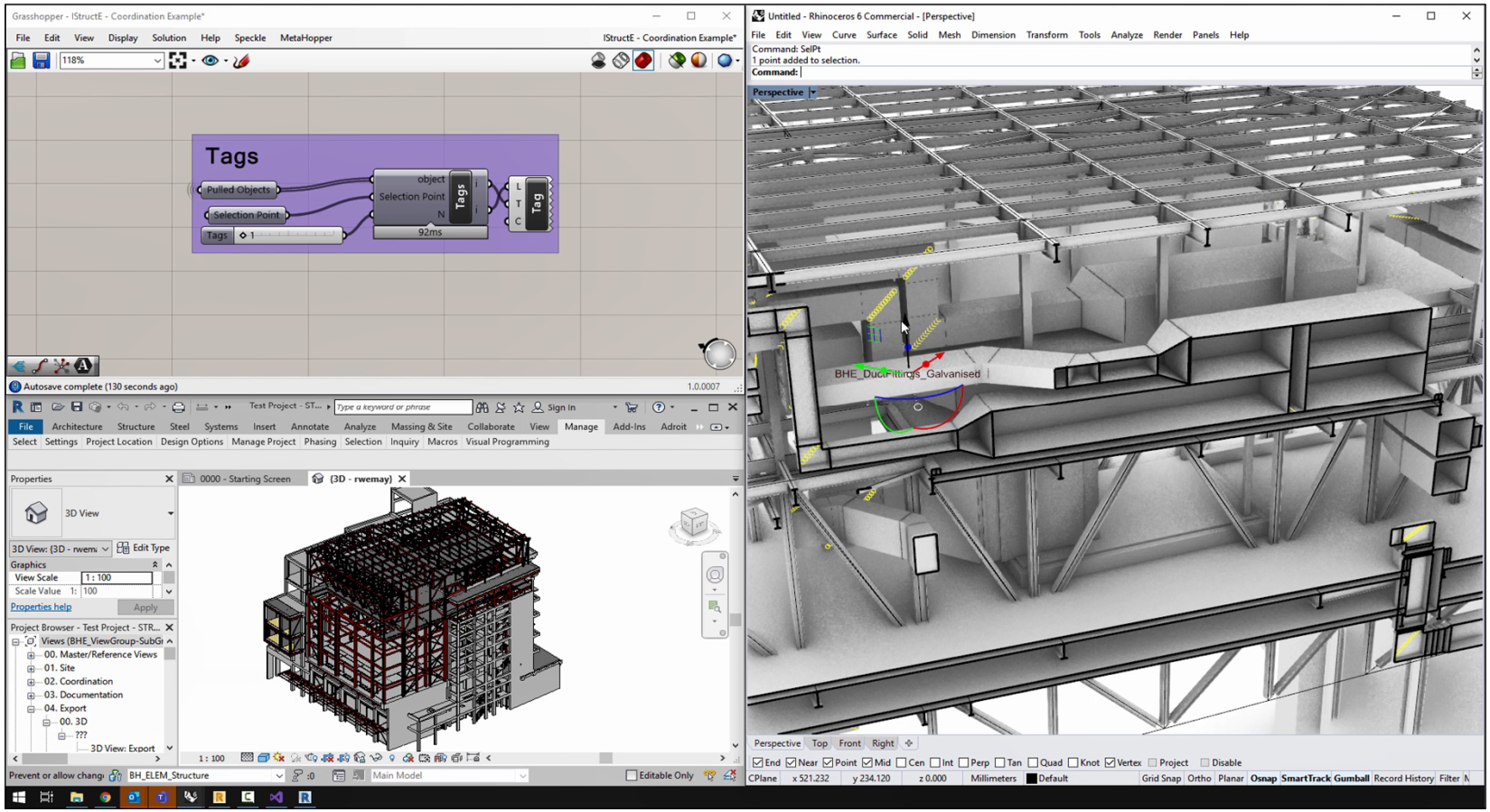
Examples:
Design to/from Robot

BHoM.xyz



- Once the Bar(s) have been pushed, we can define how the analysis should be performed.
- We can also define Loads, Load Cases, etc. as objects in Grasshopper, then push them.
- We can run the Analysis and check the results.
- We can Pull the results using the Robot Adapter:
  enabling iterative design and optimization.

Examples:
Interoperability Revit-Rhino

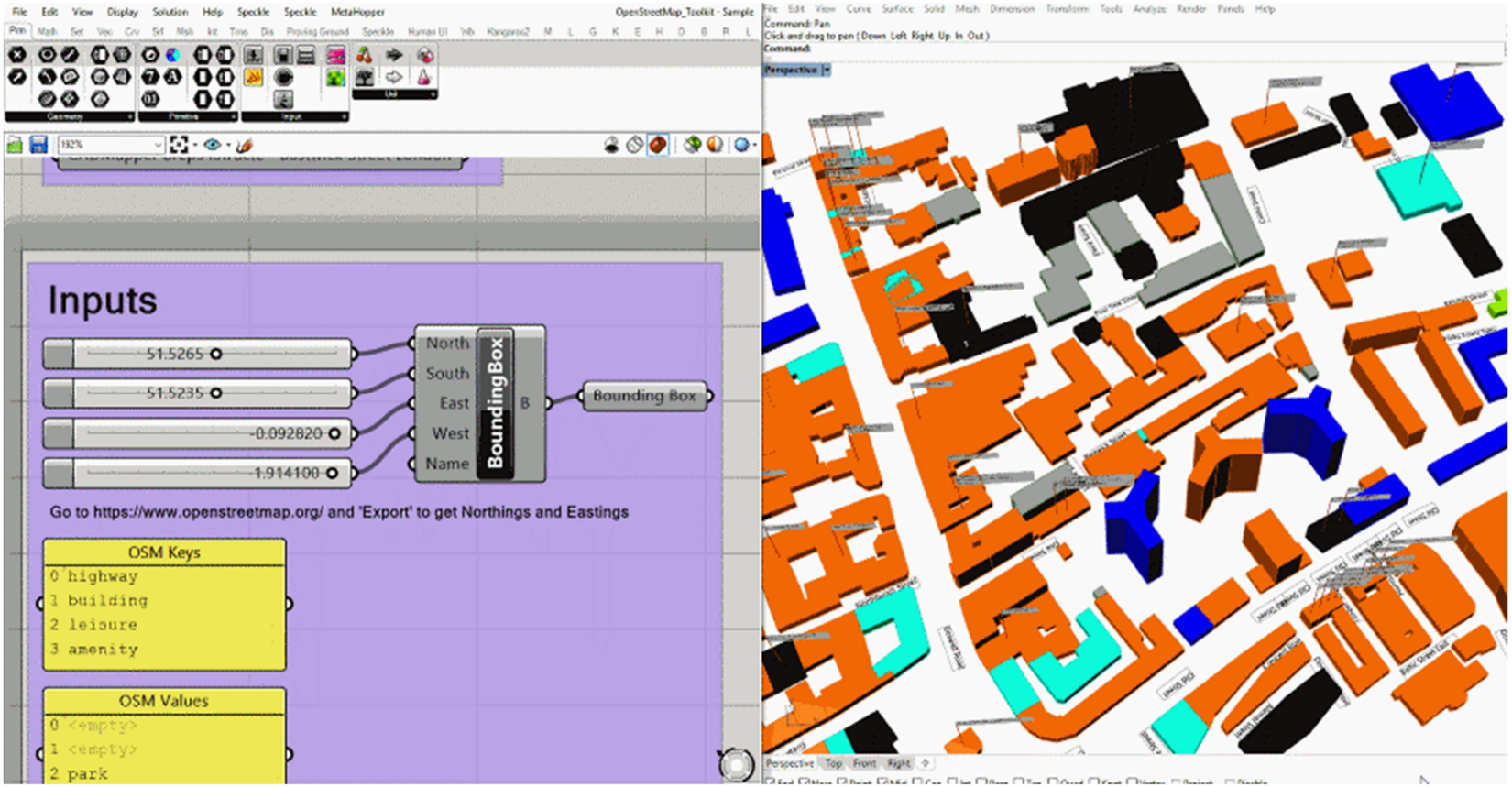


- Push/pull of Engineering/design /geometric elements with their properties
- Overlay objects which don't have formal definition in the BHoM

17

Open Street map

Download a building massing model



BHoM and Graphs

Current applications and future research

BHoM and graphs
BHoM lends itself to graph representation

BHoM.xyz

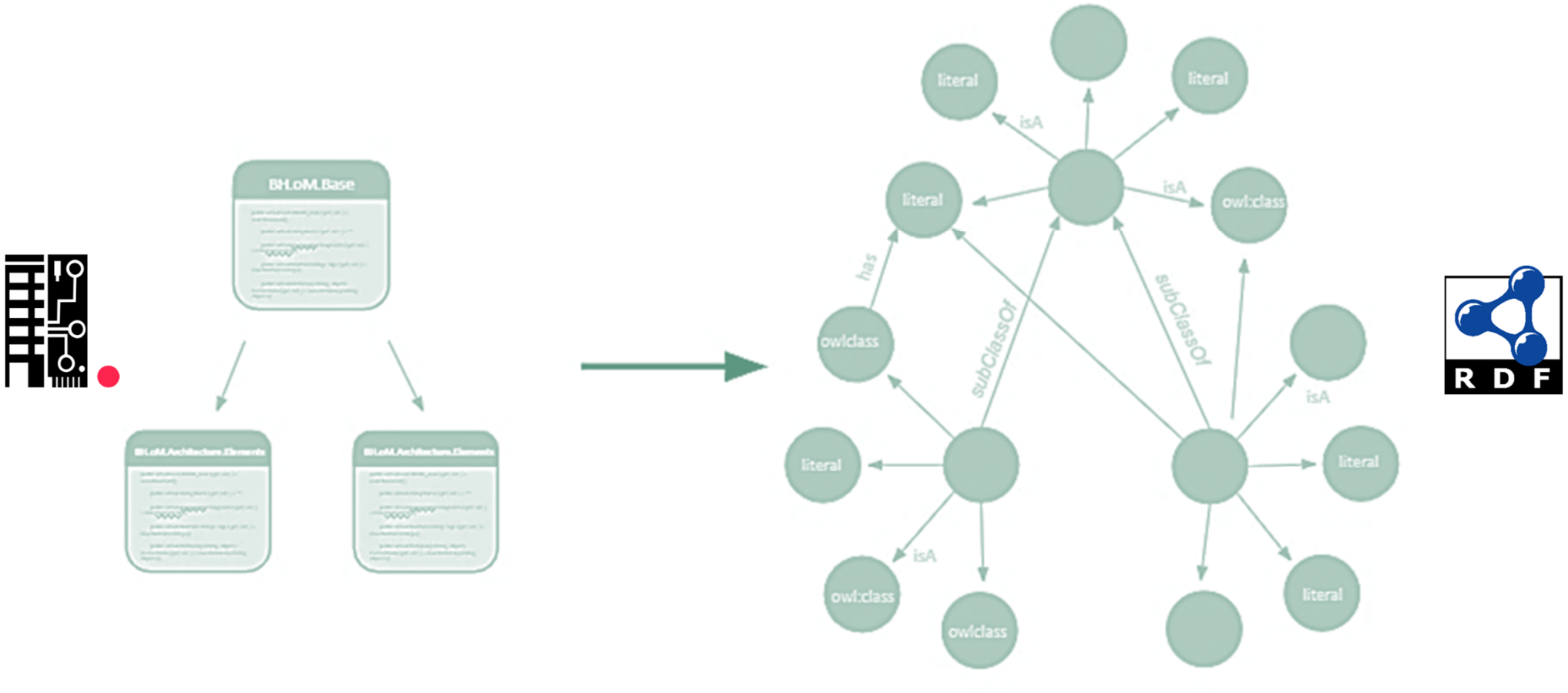
Some strict conventions followed in BHoM make it well suited for Graph representation:

- BHoM types are open to describe anything in the AEC world.
- BHoM types include classes, interfaces and enumerables. They can be instantiated into objects. Types and their object instances correspond to the facts on TBox and Abox in a knowledge graph.
- BHoM types define concepts avoiding redundant information: they only host *defining properties*. (e.g. a BHoM line is defined with two endpoint properties. The length of the Line is not included in its properties; it can be derived as a distance between its endpoints)
- Derived properties can be obtained via particular methods defined in BHoM_Engine (Query methods).
- All properties are public and accessible
- BHoM types do not own any functionality.

D. Elshani, T. Wortmann, and S. Staab, “Towards Better Co-Design with Disciplinary Ontologies: Review and Evaluation of Data Interoperability in the AEC Industry.,” in *CEUR-WS proceedings*, 2022.

BHoM and graphs

BHoM lends itself to graph representation

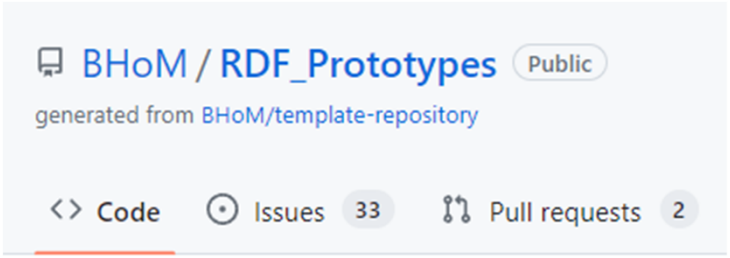


D. Elshani, T. Wortmann, and S. Staab, “Towards Better Co-Design with Disciplinary Ontologies: Review and Evaluation of Data Interoperability in the AEC Industry.,” in *CEUR-WS proceedings*, 2022.

BHoM and graphs

BHoM lends itself to graph representation

BHoM.xyz



README.md

BHoM to bhOWL Converter

"BHoM to bhOWL Converter" is part of ongoing research by University of Stuttgart and Buro Happold. It is developed as part of a research project of the Cluster of Excellence Integrative Computational Design and Construction for Architecture (IntCDC). The project's name is: Knowledge Representation for Multi-Disciplinary Co-Design of Buildings.

The developed tool helps convert BHoM data to a knowledge graph in any software BHoM supports. As a result, the converter returns a Resource Description Framework (RDF) graph serialized in Terse RDF Triple Language (TTL) format. The toolkit contains a live connector to GraphDB, a highly efficient and robust graph database with RDF and SPARQL support, compliant with W3C standards.

BHoM (The Buildings and Habitats object Model) is a collaborative framework that runs within several AEC design software, and it represents data in an object-oriented database model. OWL (Web Ontology Language) provides a standardized and expressive language for representing knowledge and relationships within a domain. It allows for creating ontologies, which are formal descriptions of the concepts and relationships within a domain. Additionally, OWL can support reasoning and inference over ontologies, allowing for automated reasoning about the relationships between different concepts and data elements. This can be particularly useful in identifying inconsistencies or gaps in data or suggesting additional data sources or mappings that might be needed to support integration efforts.

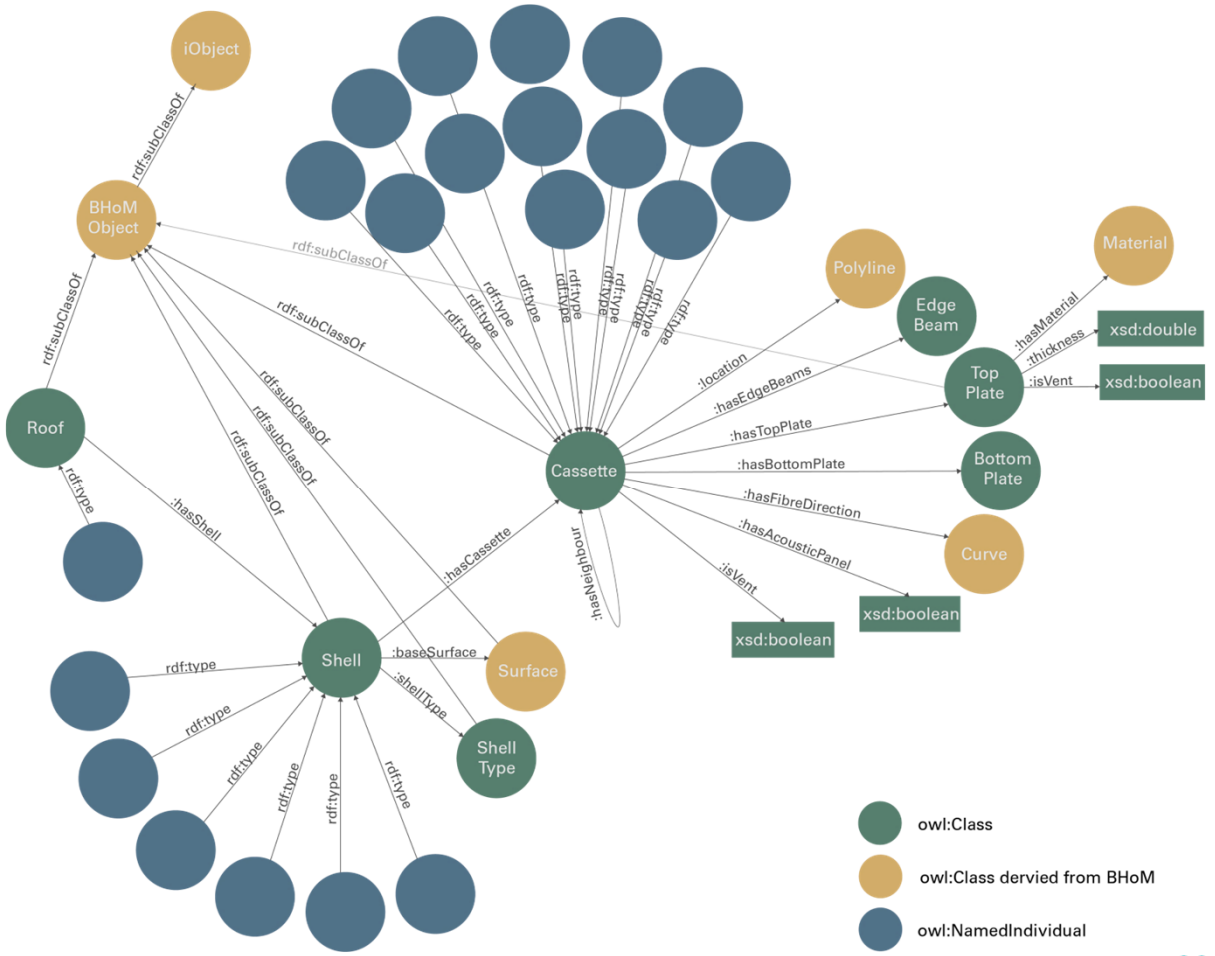
BHoM and graphs

BHoM lends itself to graph representation

BHoM.xyz

Converting to graph triples can give the following advantages:

- Convert the data to/from RDF
- Store data in a graph database (e.g. Ontotext GraphDB)
- Additional means to query info (e.g. SPARQL)
- Additional ways to validate/constraint (e.g. SHACL)
- Apply inferential reasoning
- Connect concepts with additional relations



BHoM and graphs

BHoM lends itself to graph representation

- **Connect concepts with additional relations**

Bar *= structural concept, for engineers, to represent a linear element for analysis and design (e.g. a column). Can hold/link to FEM data.*

```
namespace BH.oM.Structure.Elements
{

    [Description("1D finite element for structural analysis. Linear 2-noded element defined by a start and end node." )]
    public class Bar : BHoMObject, IElement1D, IElementM, ILink<Node>
    {
        /*************************************************/
        /**** Properties                              ****/
        /*************************************************/

        [Description("Defines the start position of the element." +
            "Note that Nodes can contain Supports which should not be confused with Releases.")]
        public virtual Node StartNode { get; set; }

        [Description("Defines the end position of the element." +
            "Note that Nodes can contain Supports which should not be confused with Releases.")]
        public virtual Node EndNode { get; set; }

        [Description("Section property of the bar, " +
            "containing all sectional constants and material as well as profile geometry and dimensions, where applicable.")]
        public virtual ISectionProperty SectionProperty { get; set; } = null;




        /*************************************************/
    }
}
```

Column *= physical world concept. Can represent an actual column in space.*

```
namespace BH.oM.Physical.Elements
{
    public class Column : BHoMObject, IFramingElement
    {
        /*************************************************/
        /**** Properties                              ****/
        /*************************************************/

        public virtual ICurve Location { get; set; }
        public virtual IFramingElementProperty Property { get; set; }

        /*************************************************/
    }
}
```

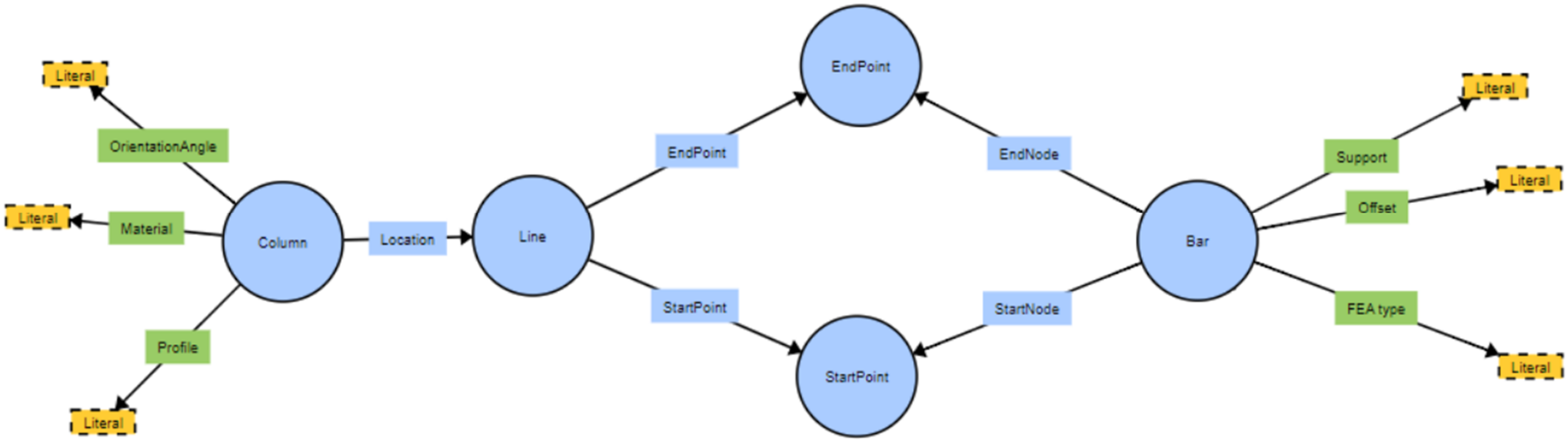
BHoM and graphs

BHoM lends itself to graph representation

BHoM.xyz

- **Connect concepts with additional relations**

*Combining a BHoM **Column** and a **Bar** in a graph-based data model*



D. Elshani, T. Wortmann, and S. Staab, “Towards Better Co-Design with Disciplinary Ontologies: Review and Evaluation of Data Interoperability in the AEC Industry.,” in *CEUR-WS proceedings*, 2022

Knowledge management in AEC consultancy:
Design VS Operational perspectives

BHoM.xyz

Design & construction perspective

Challenge: organise knowledge/data during design

Reason: better design & manufactory/assembly efficiency; cost cutting and efficiency

Applications: DfMA; modular construction; computational design; rapid prototyping

Existing data models: BHoM; BOT; IFC, IFCOWL;

Operational perspective

Challenge: organise and retrieve the operational (post-construction) data

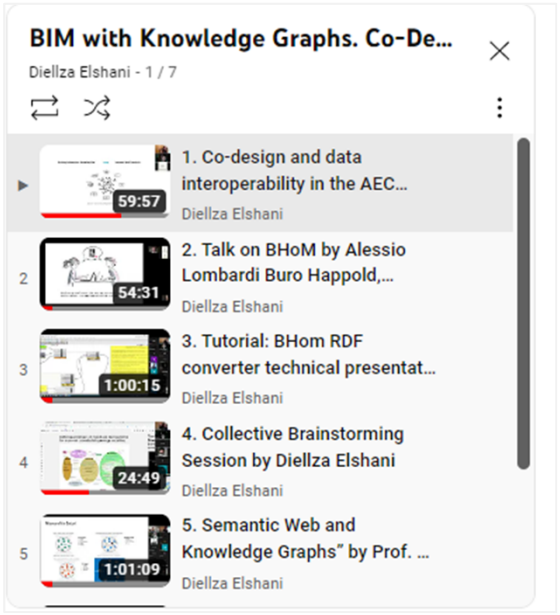
Reasons: performance monitoring and maintenance; stakeholders engagement; data ownership and control

Applications: Sensor data management; BIM data querying; validation

Example data models: Project Haystack, Brick Schema, RealEstateCore, Azure Digital Twins, Google Digital Building Ontology

Design perspective:
Digital Futures workshop

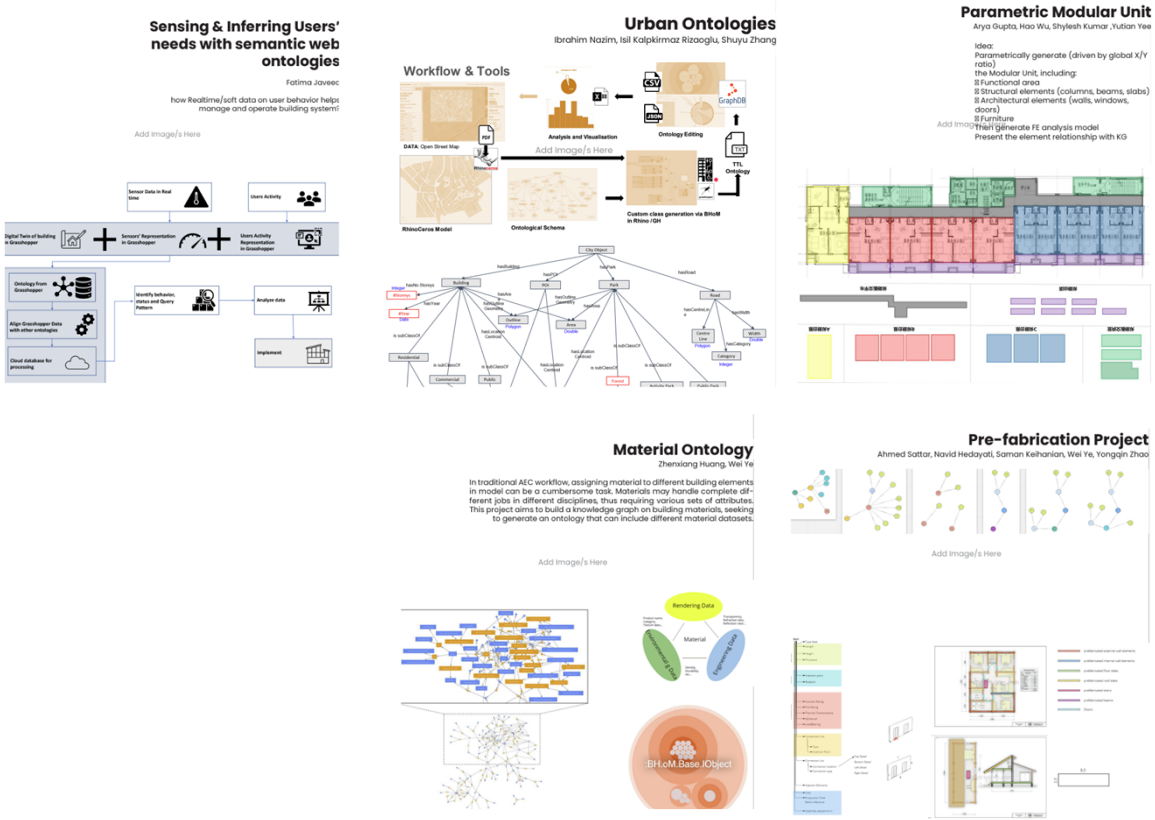
BHoM.xyz



https://vimeo.com/724817479

https://www.youtube.com/playlist?list=PLZ1uRtLJZbxAfkUXXxOgVRuMA8W52RBe2

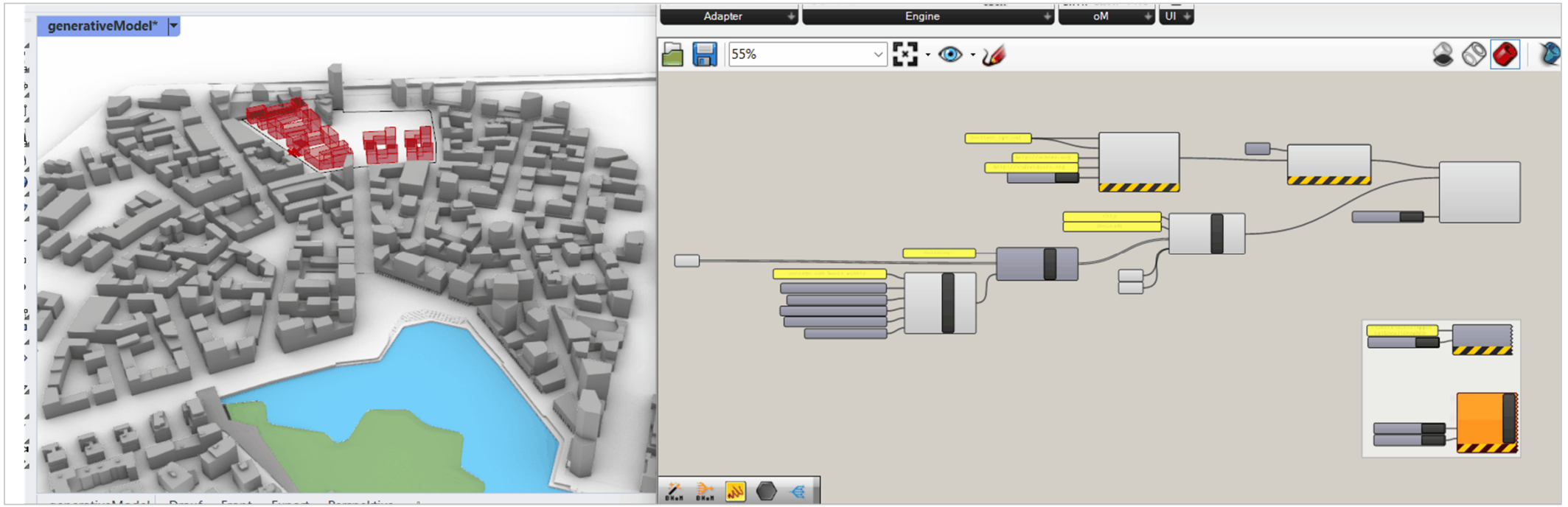
Digital Futures workshop, 100 applicants and dozens of active users



Design perspective:
CoDeC hackaton

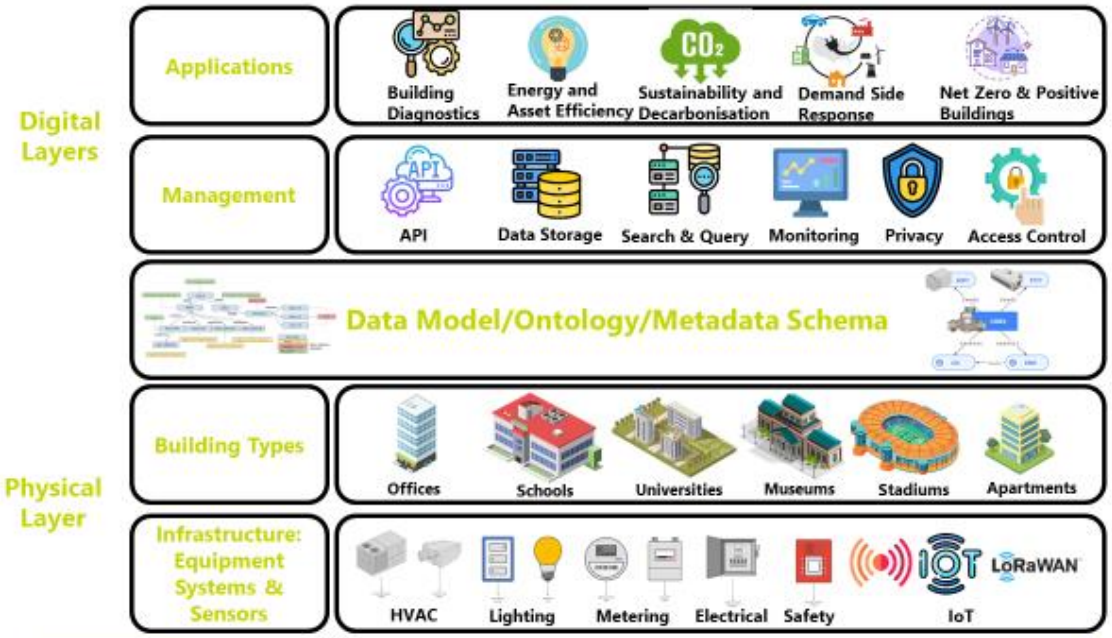
BHoM.xyz

Demonstrating the write to RDF (TTL) and export to GraphDB.



Operational perspective:
Connecting digital and physical layers – design to “as built”

BHoM.xyz

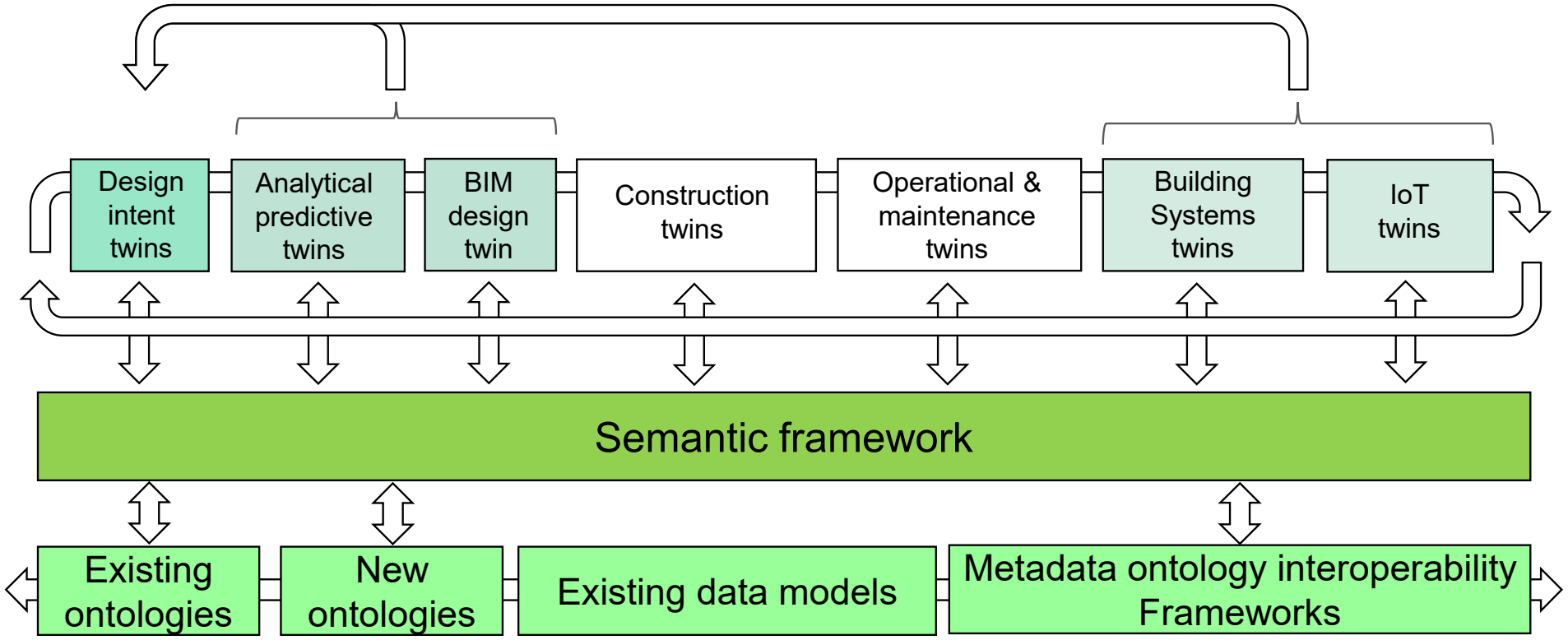


- Data comes from post-construction sources: sensors, measurements, time-series
- Need to connect to physical layer “as built”
- Little to no efforts to organise the data
- Hard to query information after construction
- Hard to connect existing concepts “as designed”

Operational perspective:
Connecting digital and physical layers – design to “as built”

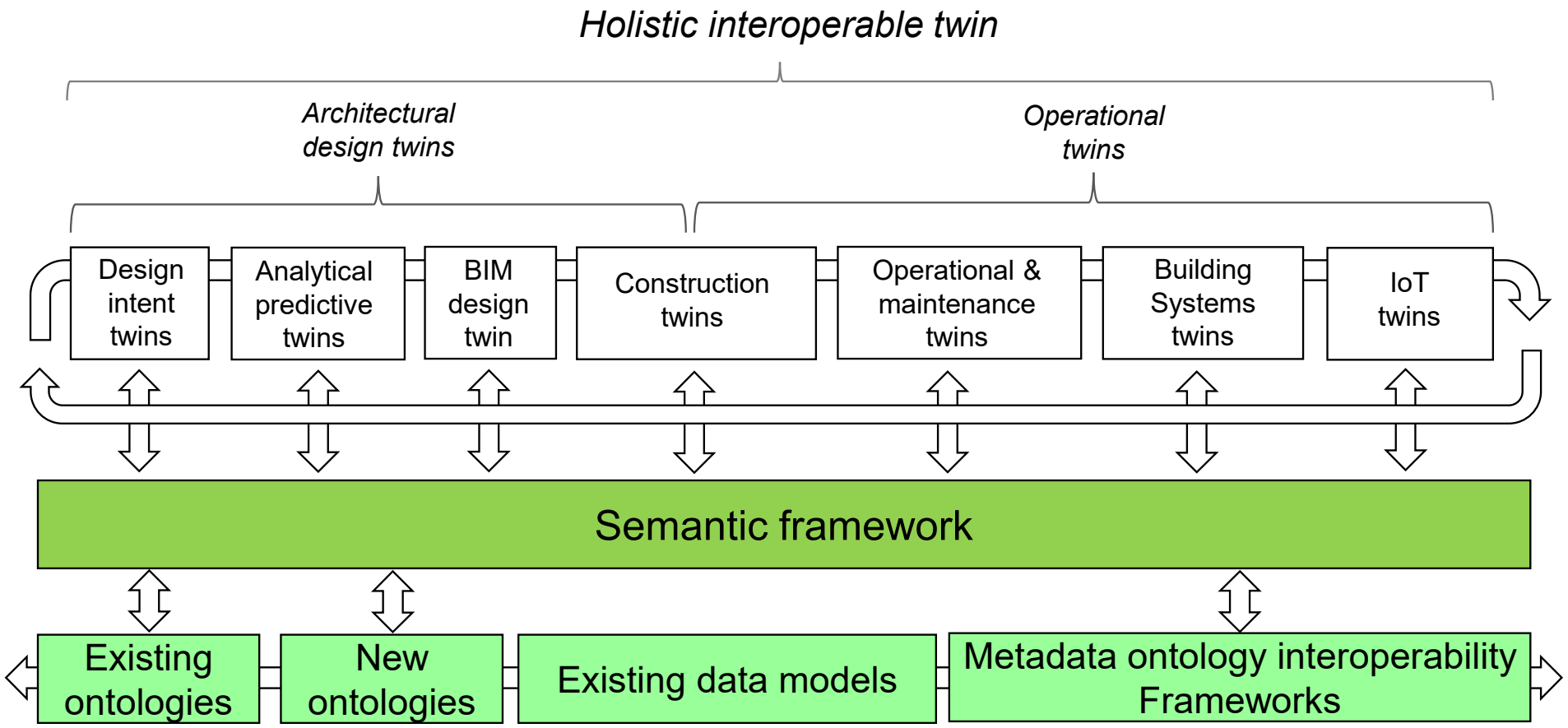
BHoM.xyz

Combine operational twins
with analytical and BIM design twins
in order to inform the Design Intent Twins



Operational perspective:
Connecting digital and physical layers – design to “as built”

BHoM.xyz



Future research
Graphs and Machine Learning applications

BHoM.xyz

- Neuro-symbolic approaches to extrapolate design information from existing, **unstructured** design data (e.g. CAD drawings)
- ML methods to identify “similar” concepts (classes) and potential ways to interconnect them
- Entity recognition/relation extraction
- Missing/incomplete entity or relation resolution; conflicts

Thank you!


University of Stuttgart

- Diellza Elshani, PhD Candidate, Research Associate
- Tenure-Track prof. Dr. Thomas Wortmann, Chair for Computing in Architecture ICD/CA


Buro Happold, London

- Alessio Lombardi, interop/ML lead
- Dr. Al Fisher, Head of Computational Development

Some refs

D. Elshani, T. Wortmann, and S. Staab, “Towards Better Co-Design with Disciplinary Ontologies: Review and Evaluation of Data Interoperability in the AEC Industry.,” in *CEUR-WS proceedings*, 2022

D. Elshani, A. Lombardi, A. Fisher, D. Hernandez, S. Staab, and T. Wortmann, “Knowledge Graphs for Multidisciplinary Co-Design: Introducing RDF to BHoM,” in *CEUR-WS proceedings*, 2022.

D. Elshani, A. Lombardi, A. Fisher, D. Hernandez, S. Staab, and T. Wortmann, “Inferential Reasoning in Co-Design Using Semantic Web Standards Alongside BHoM,” presented at the 33. Forum Bauinformatik 2022, Munich, 2022.

Toth, B., Janssen, P., Stouffs, R., Chaszar, A., & Boeykens, S. (2012). Custom digital workflows: a new framework for design analysis integration. International Journal of Architectural Computing, 481-500

Some links

http://bhom.xyz

github.com/BHoM/RDF_Prototypes