

Linked Building Data and Ontologies

Summer School of Linked Data in Architecture and Construction

June 10, 2024

Alex Donkers

a.j.a.donkers@tue.nl

Alex Donkers



Postdoctoral researcher

ITEA4 FireBIM

Information Systems in the Built Environment

Eindhoven University of Technology

Co-chair

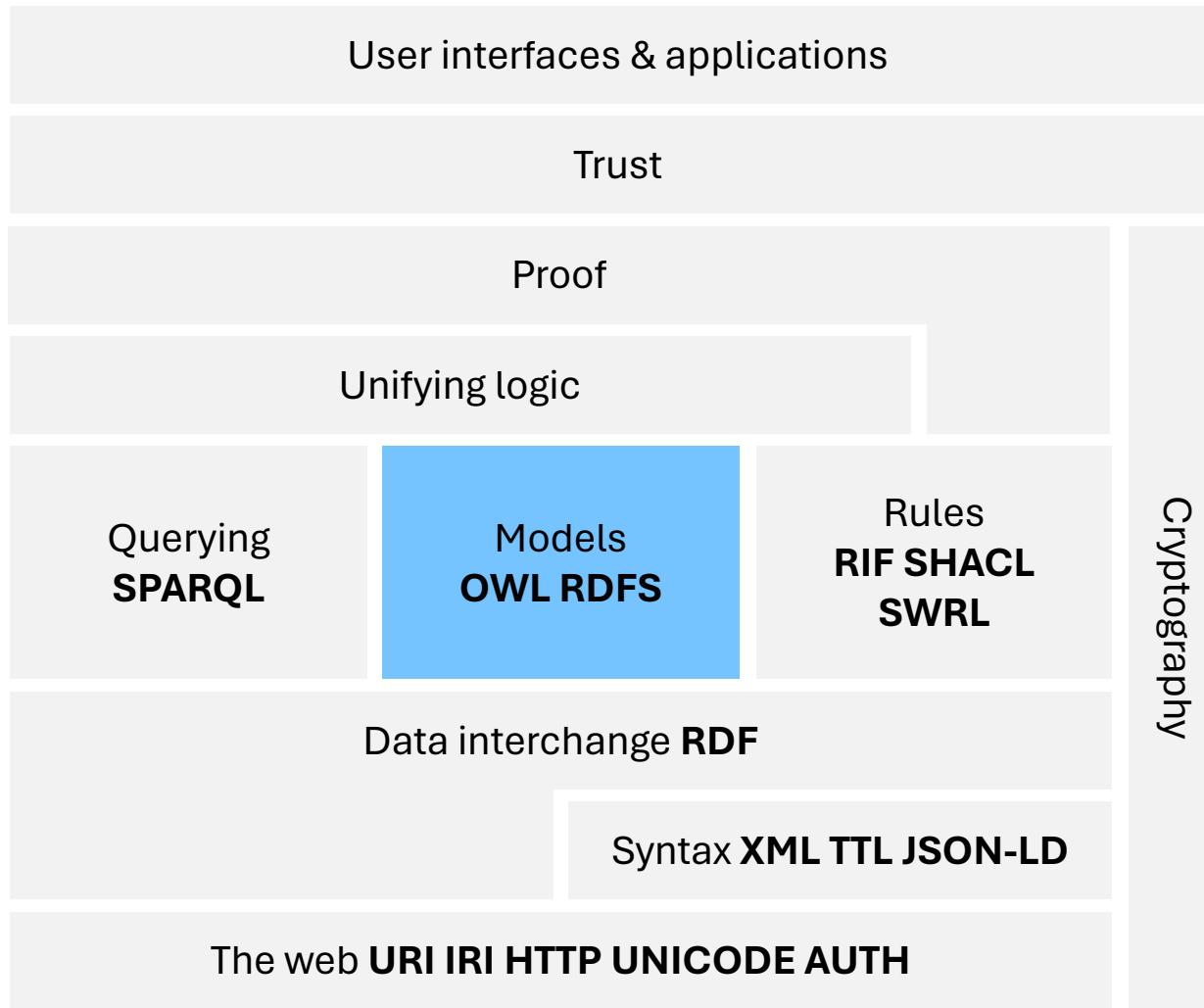
W3C Linked Building Data Community Group

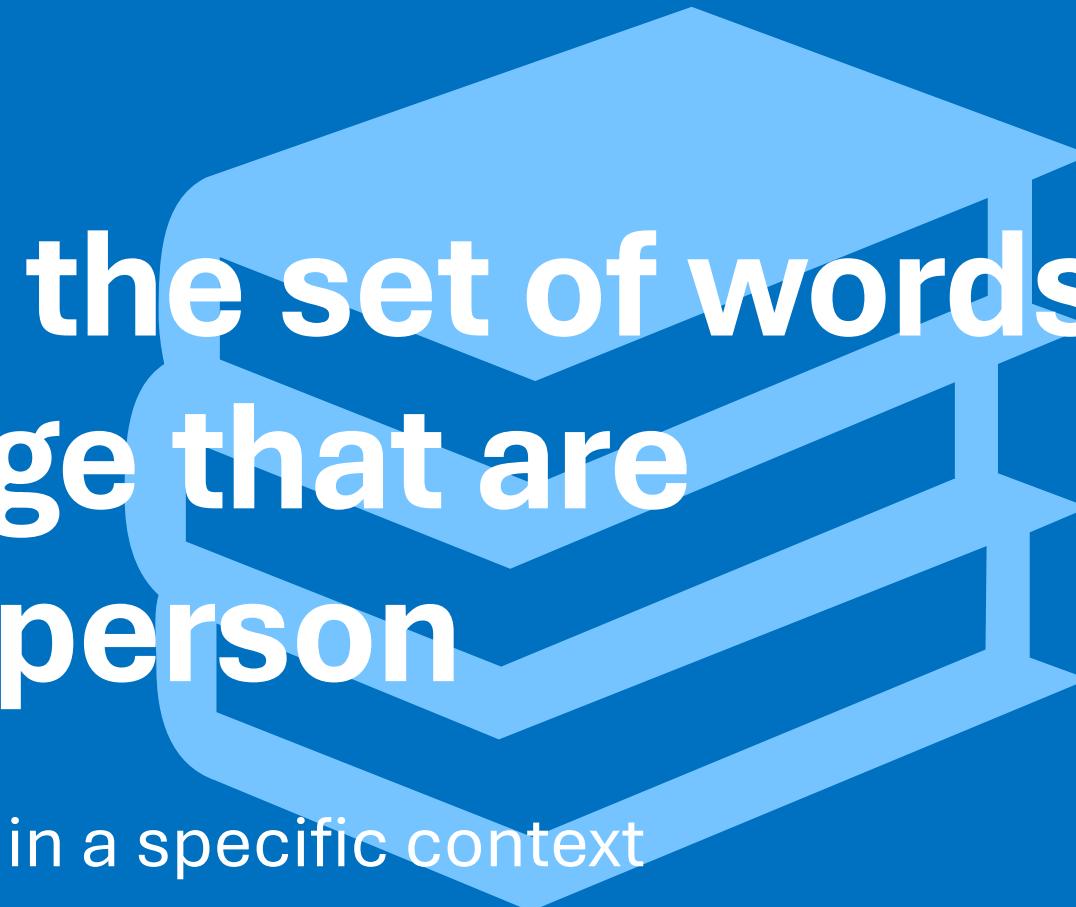


In this lecture, you'll learn:

- × What an ontology is
- × Why we use ontologies
- × Basics of ontology engineering
- × The difference between Abox and Tbox
- × Open World Assumption and the basics of reasoning
- × Common ontologies in the AEC industry
- × Practical tips and tools to kickstart your ontology development!

Ontology languages in semantic web stack





**A vocabulary is the set of words
within a language that are
familiar to that person**

Classify **terms** that can be used in a specific context

Classify **relationships** between terms

Define **constraints** on using those terms

H. Sack. "Linked Data Engineering: 3.1
Vocabularies and Ontologies"



An ontology is a formal, explicit specification of a shared conceptualization

R. Studer, V. R. Benjamins, and D. Fensel.
“Knowledge Engineering: Principles and methods”. In: Data & Knowledge Engineering
25.1-2 (1998), pp. 161–197. doi:
10.1016/S0169-023X(97)00056-6.

An ontology is a formal, explicit specification of a shared conceptualization

Conceptualization

An abstract model of something in the world that we want to represent

Explicit

All concepts in a domain are defined using formal logics supported by languages in the semantic web stack

**Formal
Shared**

Machine-understandable

Created for and agreed on by multiple users

Level 0

Level 1

Level 2

Level 3

CAD

2D BIM

3D BIM

Federated BIM

Integrated BIM

Semantics
× Ontologies
× RDF, OWL, SPARQL
× ICDD

International standards
(ISO)
× IFC
× IDM
× IFD

Integrated, interoperable
models

Proprietary formats and
the development of
domain-specific methods
and standards

Domain-specific models
× Architectural
× Structural
× HVAC
× Asset management

Central management of
files in Common Data
Environments

Cloud-based information
and coordination

Drawings

Geometric models

Paper

Files

1990s

2000s

2010s

2020s

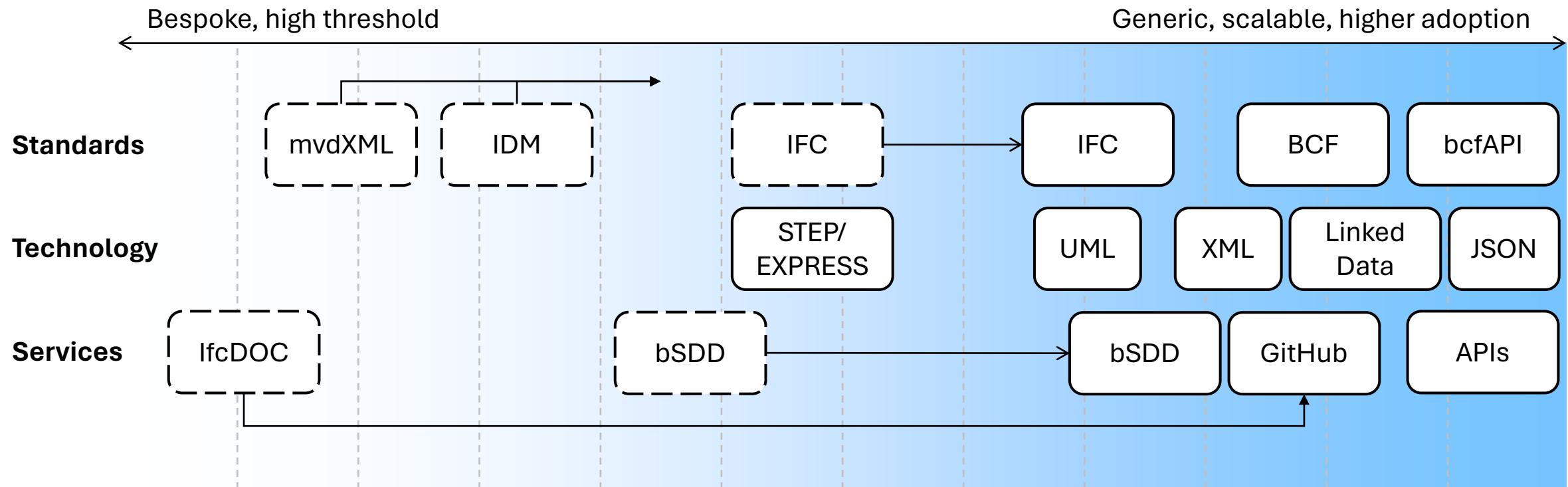
Semantics

Standards

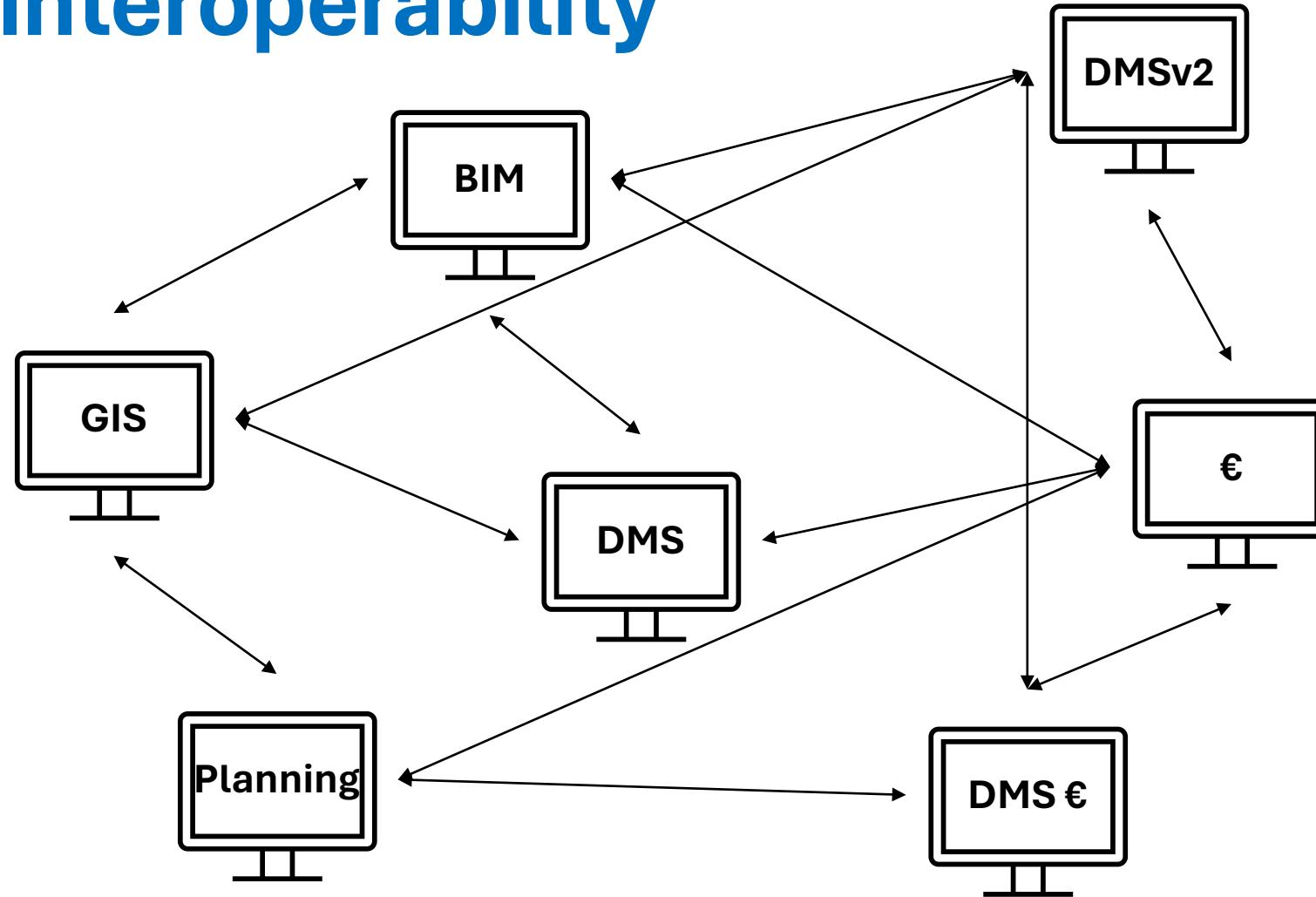
Information

Collaboration

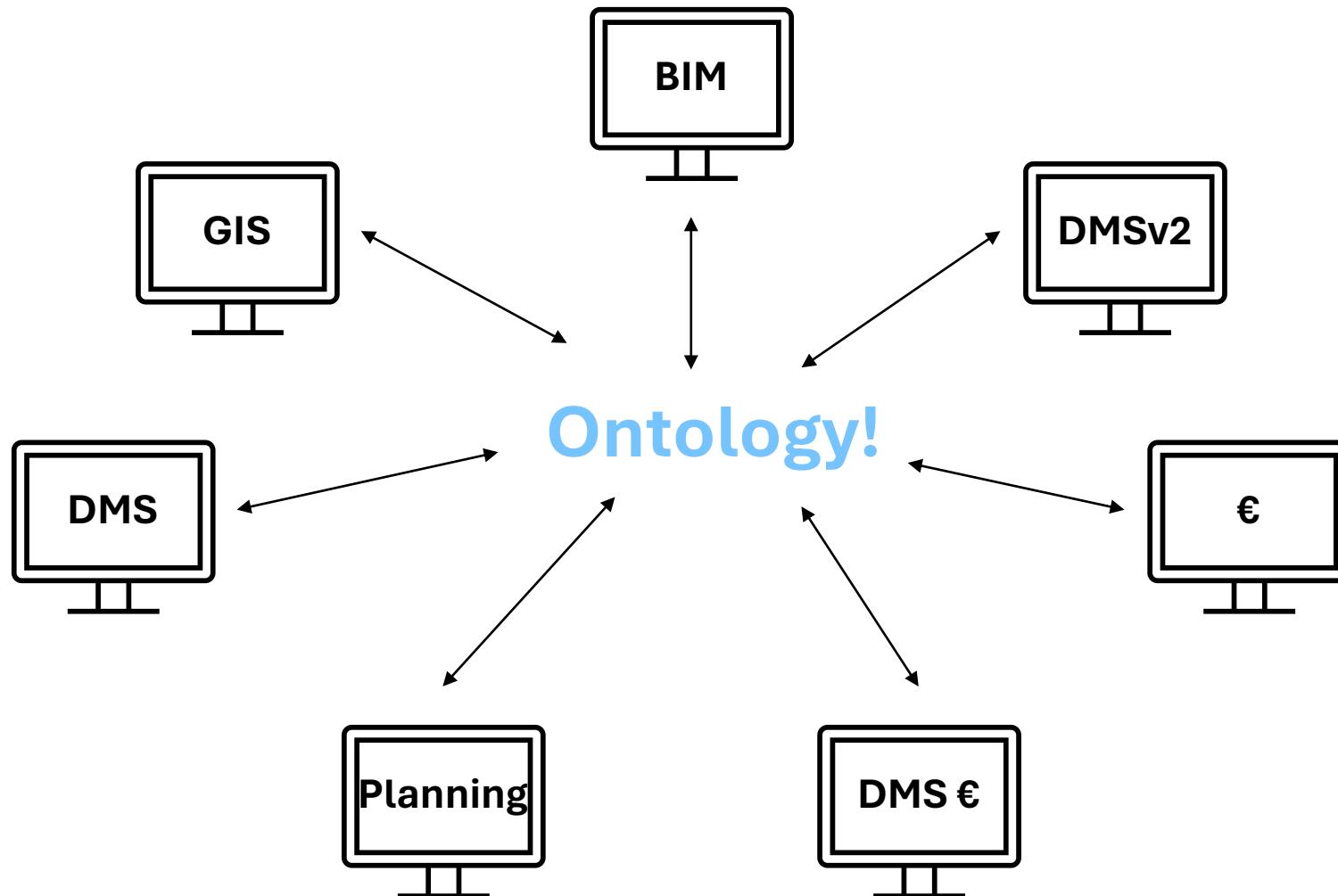
Developments in buildingSmart



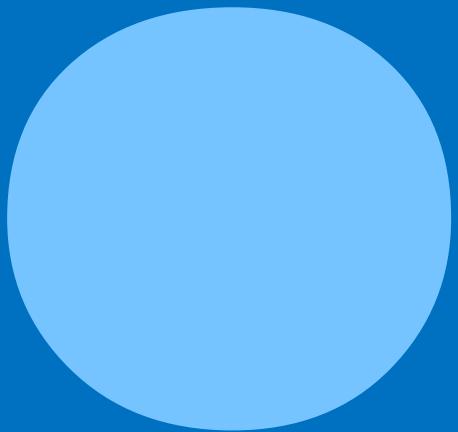
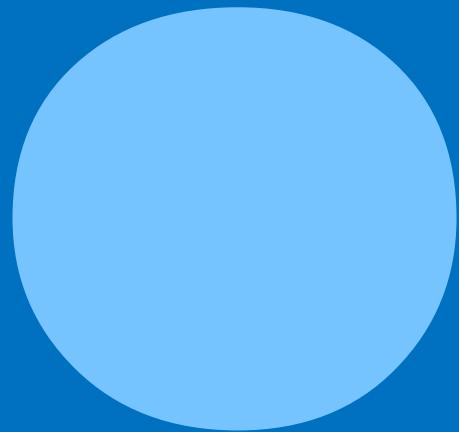
Semantic interoperability



Semantic interoperability



How to represent an ontology?



Ontologies are represented using

Classes

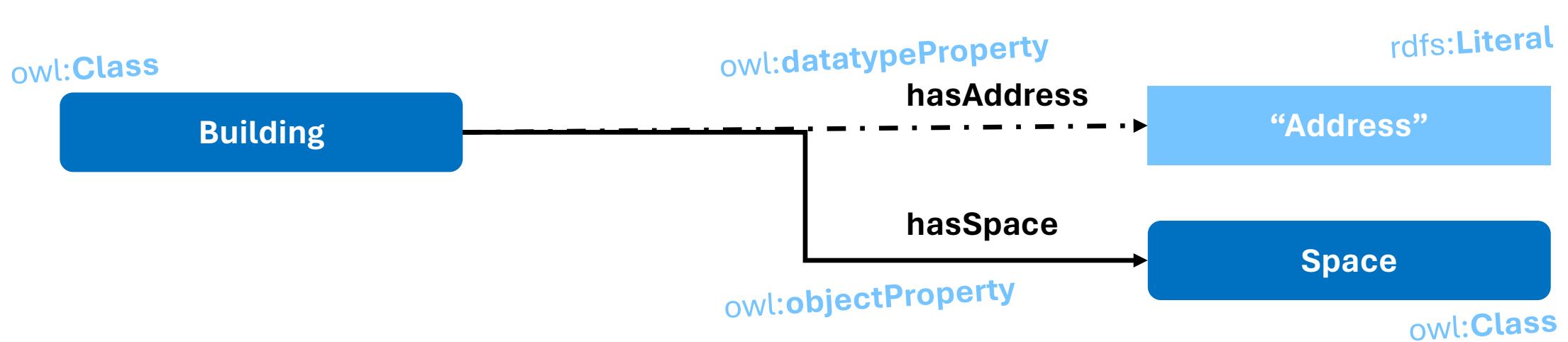
Abstract representation of a real-world concept

Relationships

Relationship between two classes or between a class and a literal

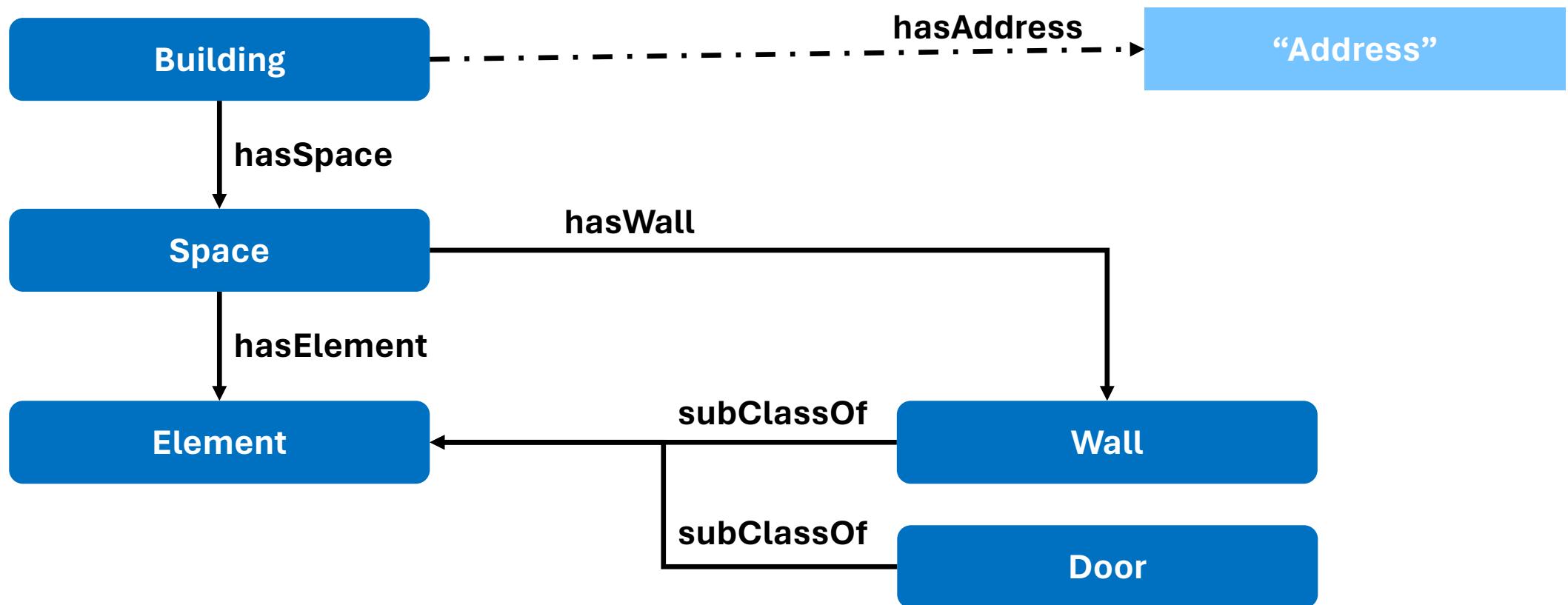
Literals

Data values, such as strings, booleans or integers



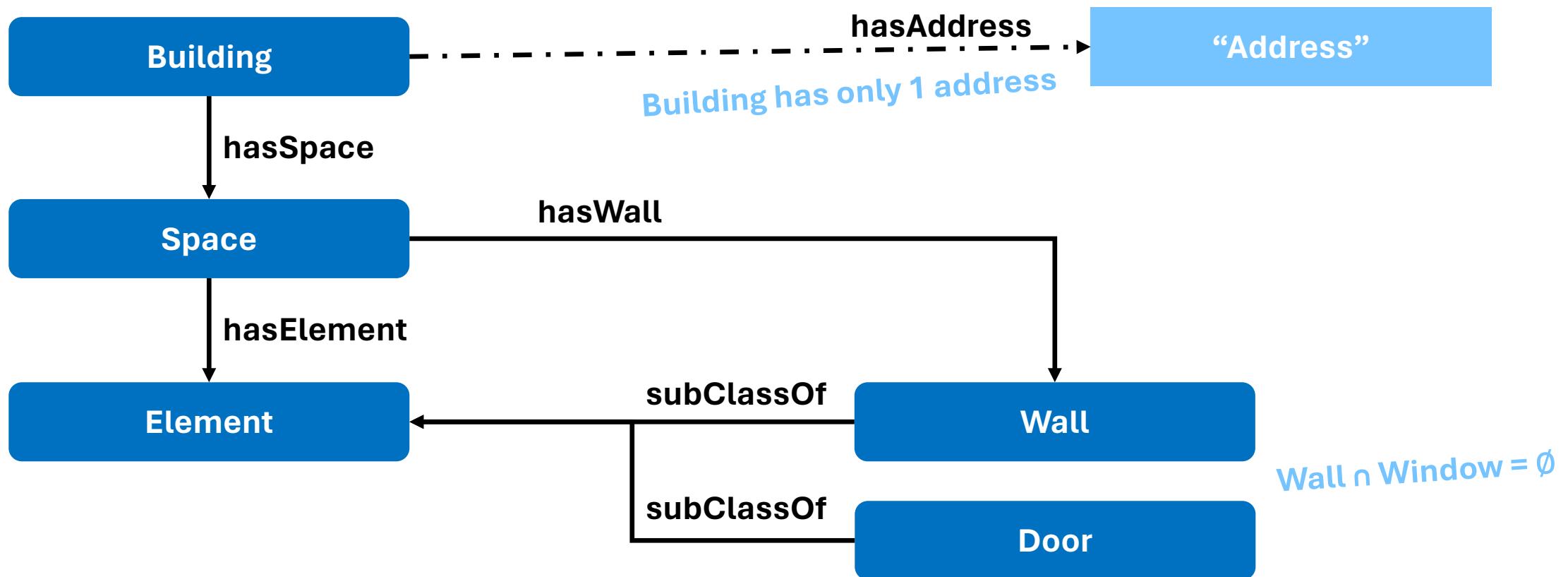
Slightly more complex ontology structures

Classes can have sub-classes



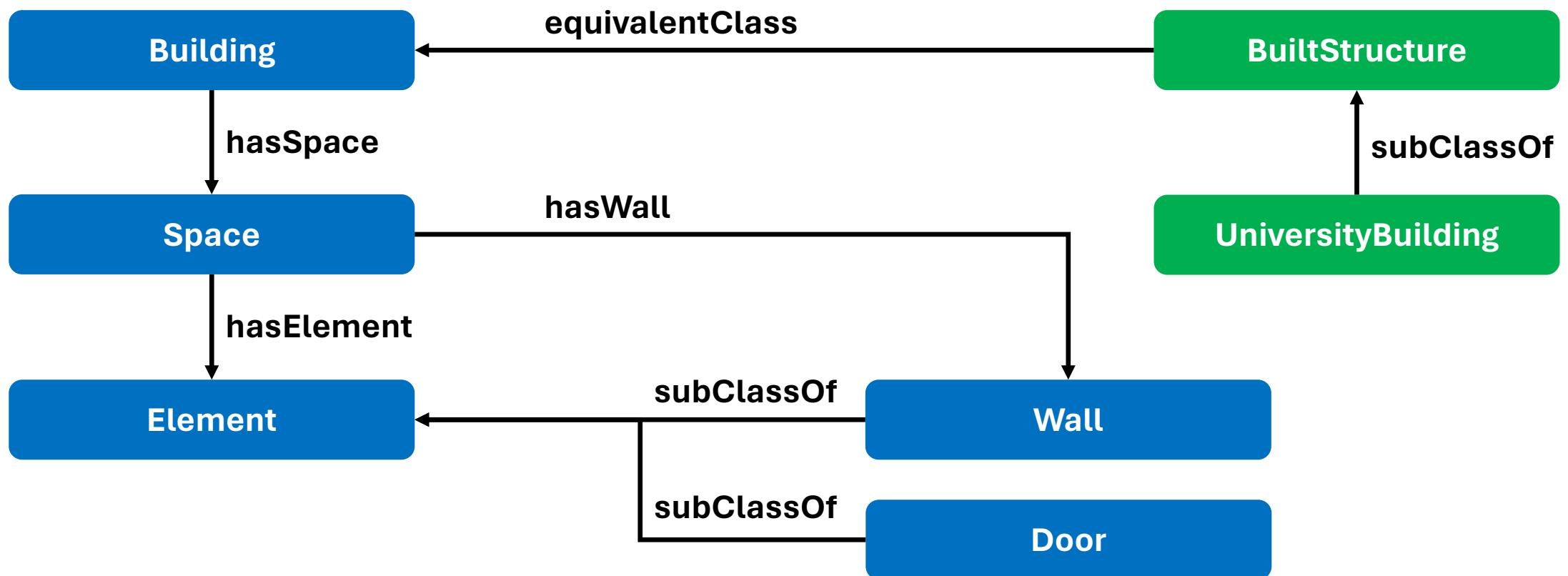
Slightly more complex ontology structures

Rules can be defined for classes and relationships



Connecting ontologies

Relationships can connect classes from two ontologies

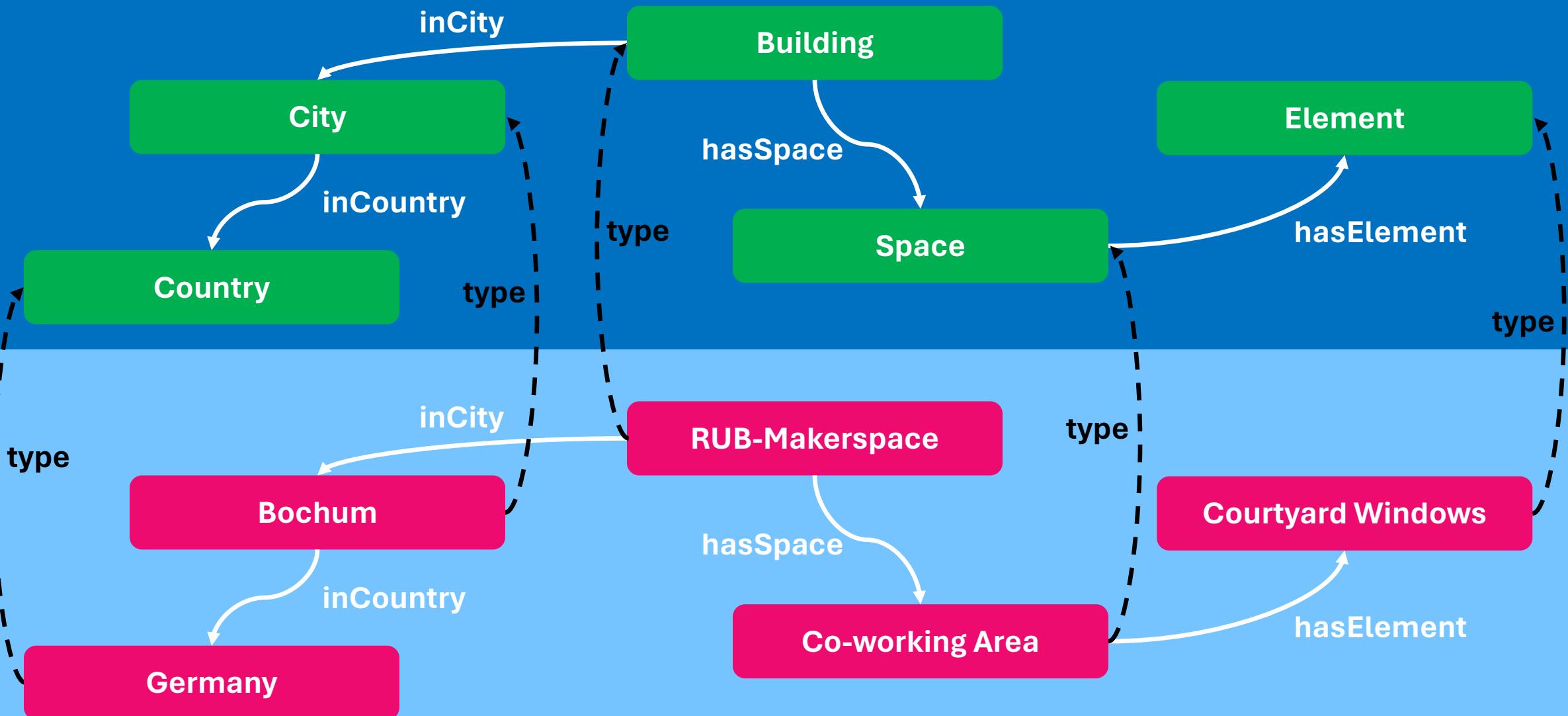


Ontologies

VS

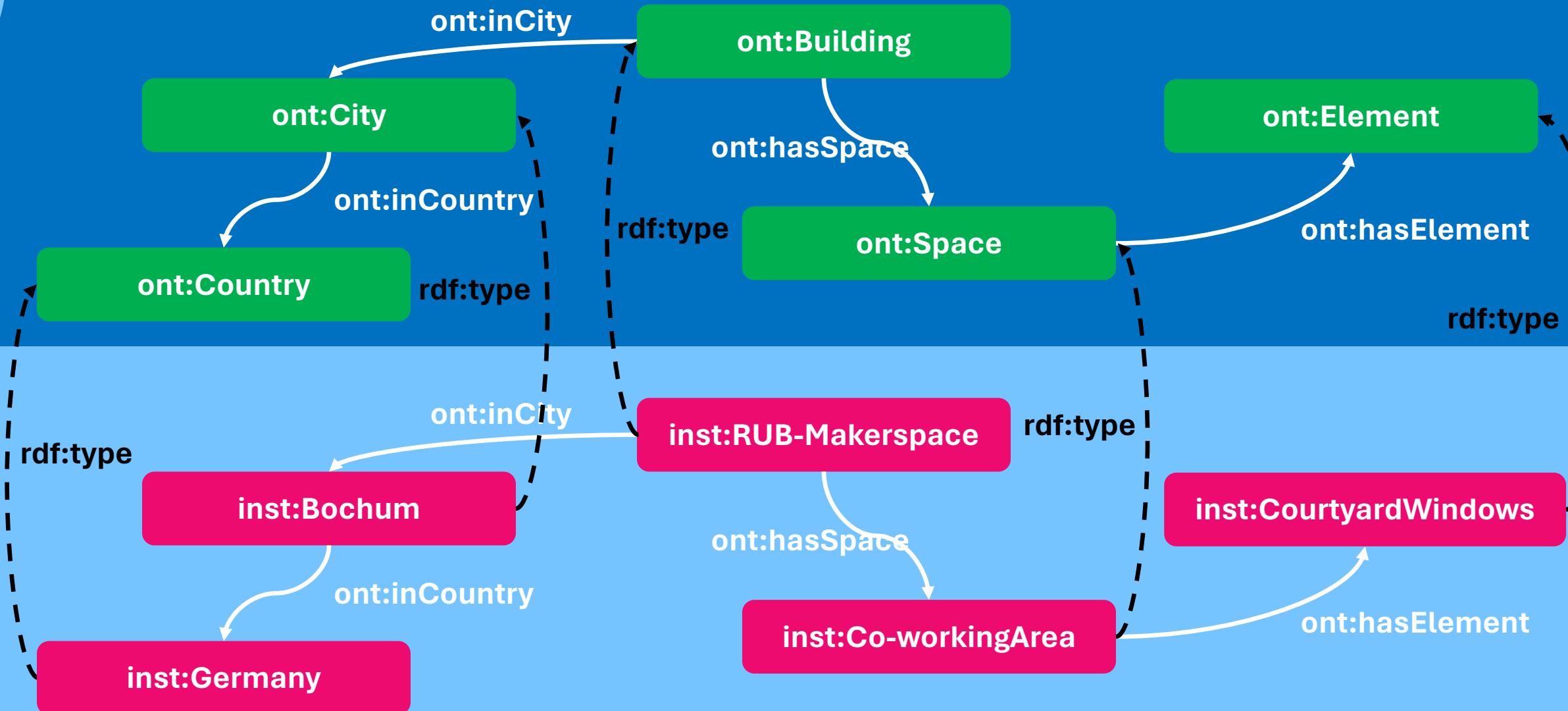
Data

Ontologies



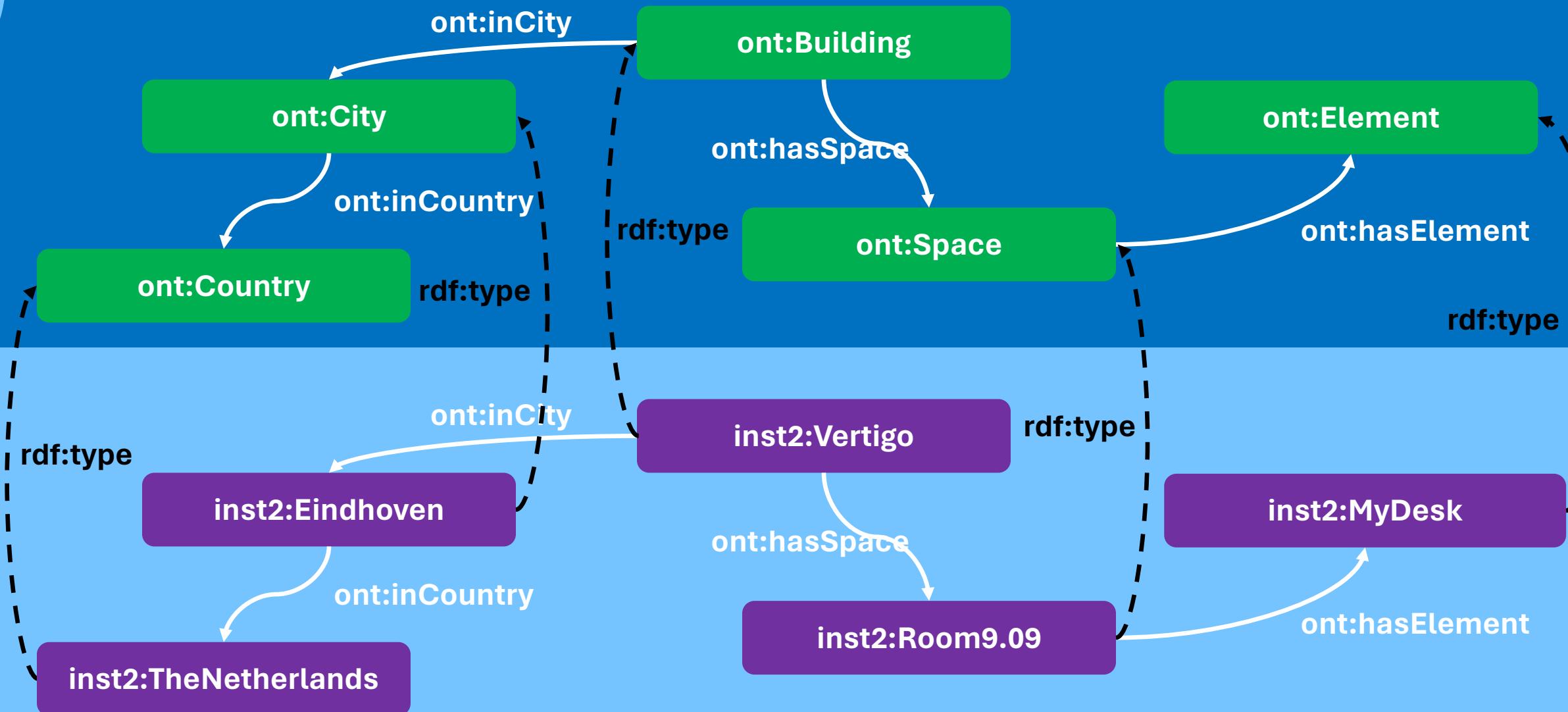
Data

Ontologies



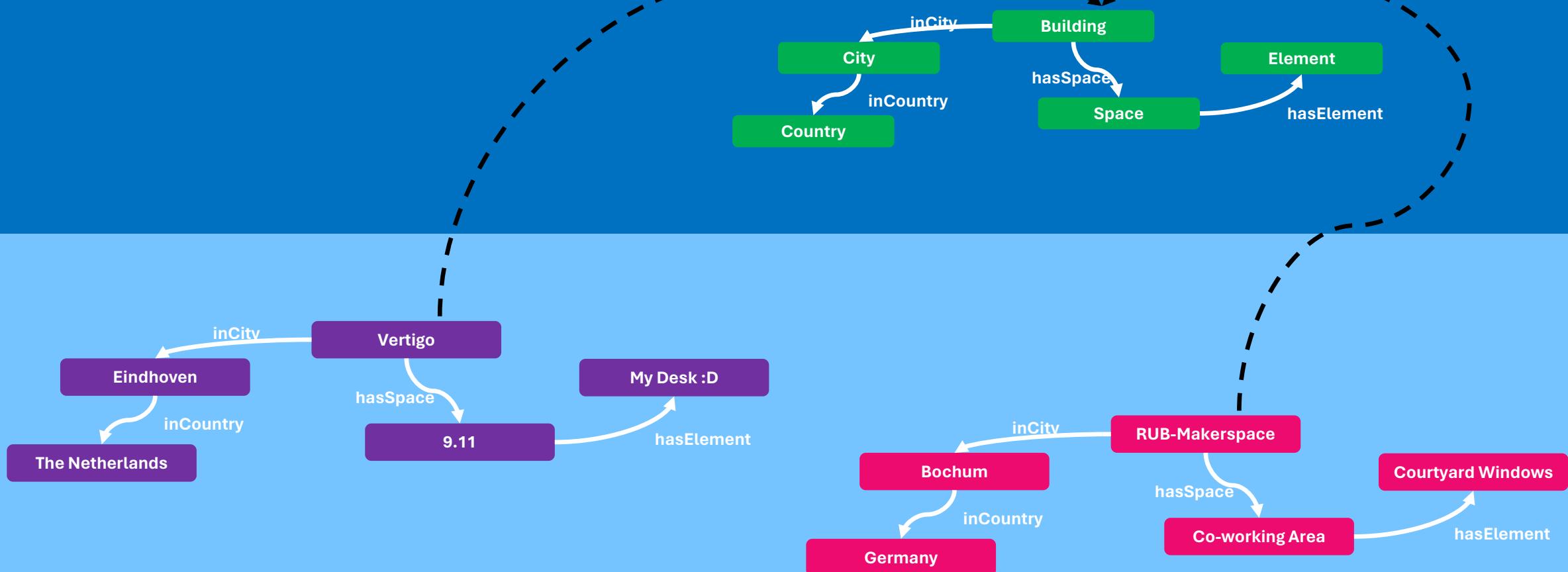
Data

Ontologies



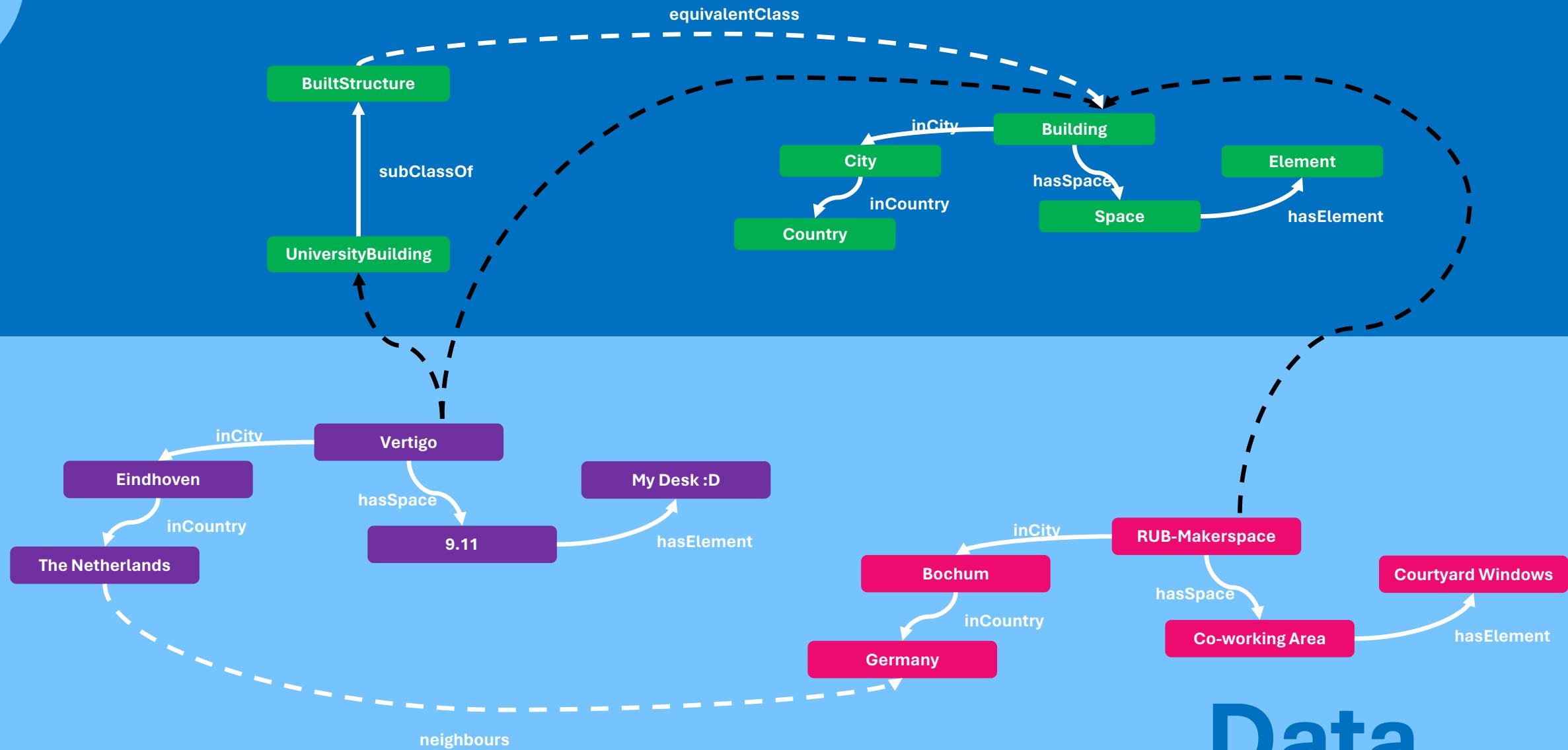
Data

Ontologies



Data

Ontologies



Data

Ontologies

- × **Abstract representation of concepts**
- × Terminology box (Tbox)
- × Ontologies, Vocabularies, Taxonomies, Object type libraries (OTL), dictionaries, masterdata
- × Happens in **ontology editors**, such as Stanford Protégé



- × **Unique real-world phenomena (or ‘things’)**
- × Assertion box (Abox)
- × Instances, or data
- × Ontologies and Data are represented in different files
- × Happens in **graph databases**, such as GraphDB



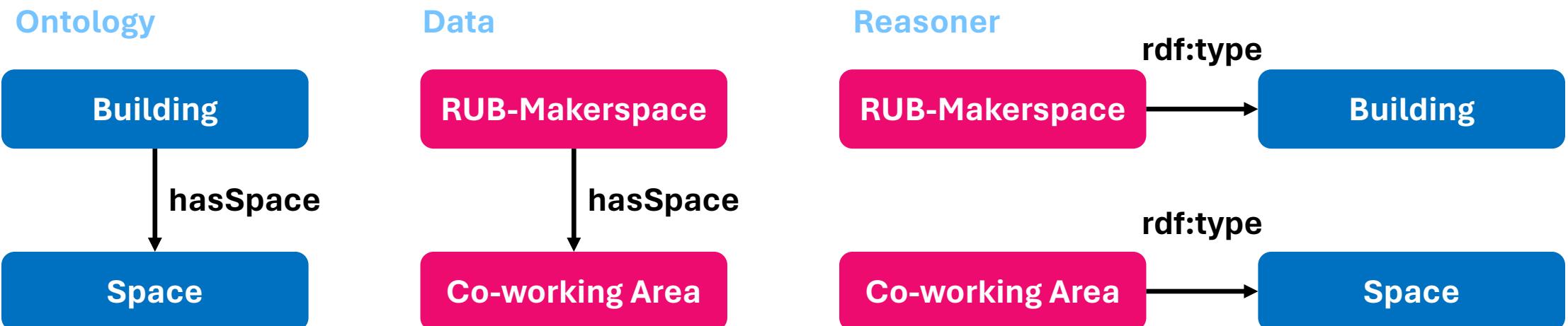
Data

Open world assumption

and a bit of reasoning

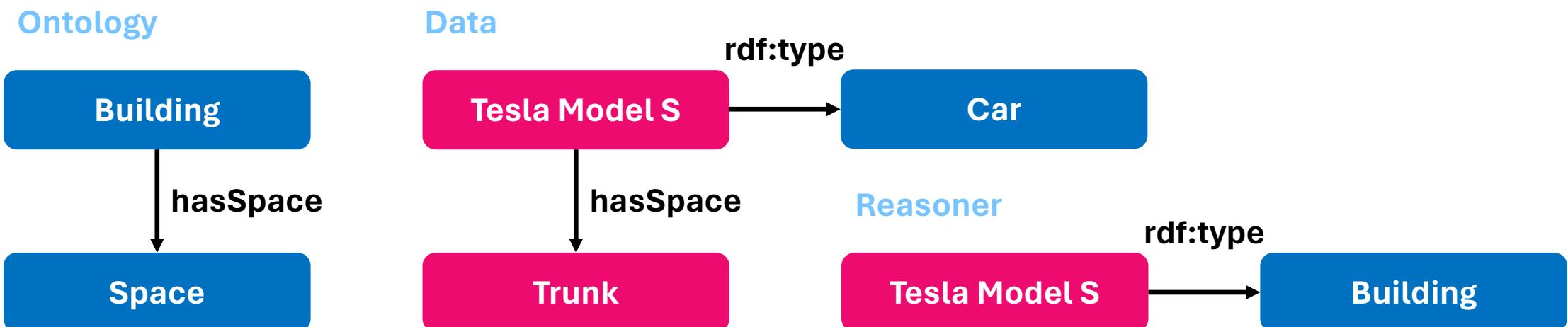
An example with rdfs:Domain

- × Relationships can have a **domain** and **range**, stating the **expected** class-membership of the subject and the object of a triple
- × If **hasSpace** has a **domain** Building and a **range** Space, a reasoner would do the following, even if you didn't explicitly specify this in the Data



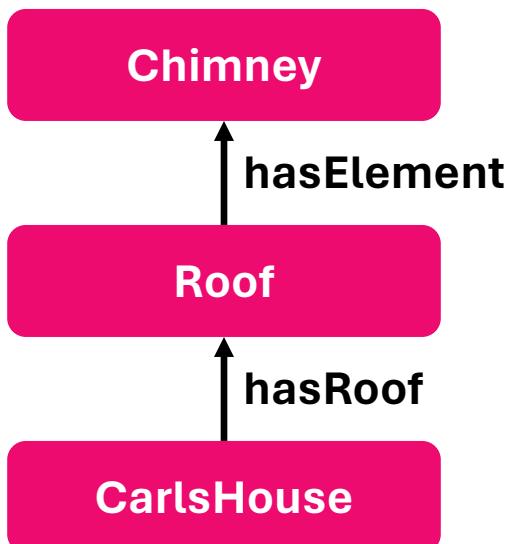
An example with rdfs:Domain

✗ Be careful!



The Open World Assumption (OWA)

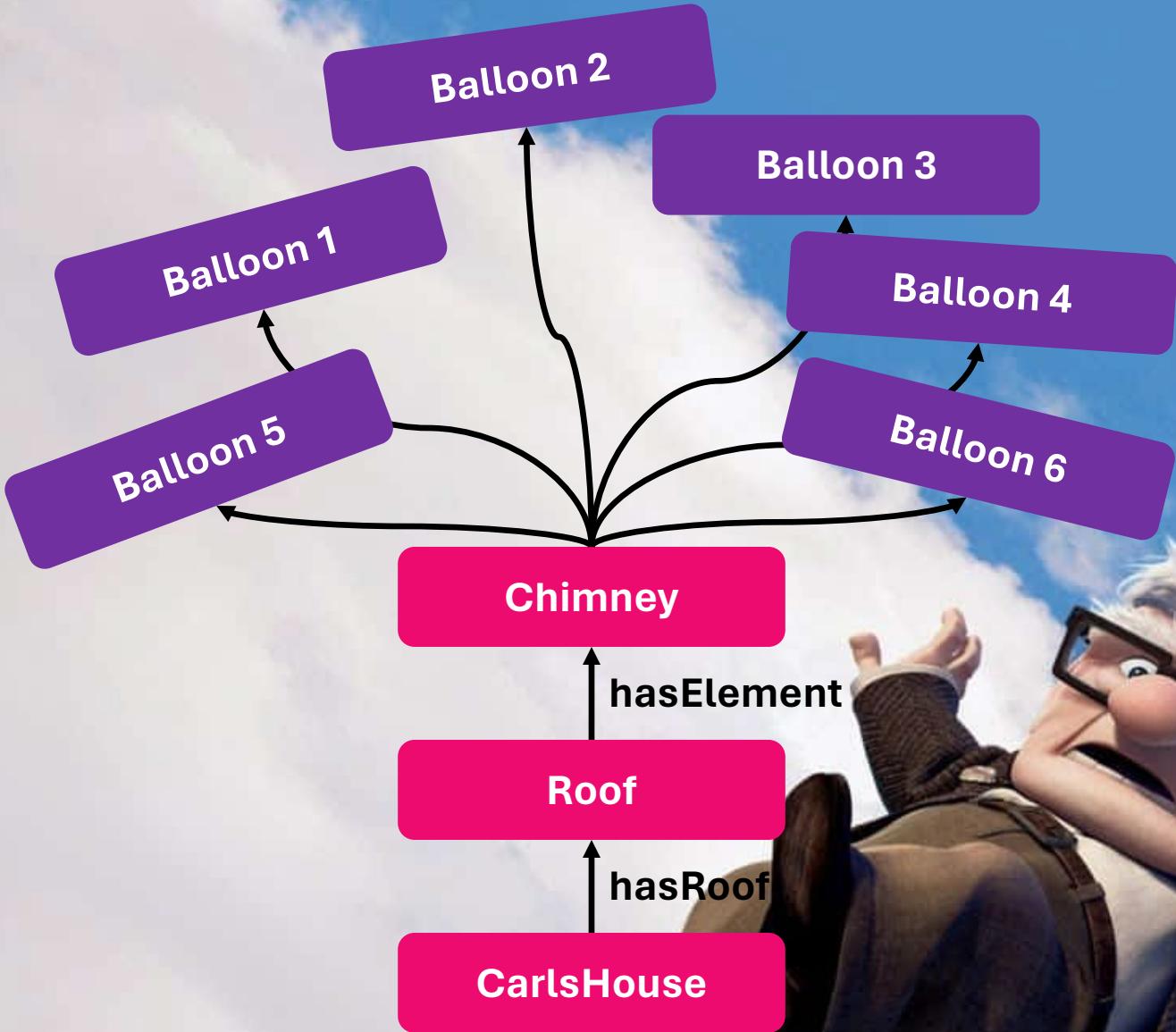
- × The assumption that a statement can be true, even if it is not known to be true.
- × Different from the **Closed World Assumption**: statements that are not known to be true are considered false.



Query
Can CarlsHouse fly?

CWA: No

OWA: We don't know, but possibly yes



The Open World Assumption (OWA)

- × Mimics the real world
- × We, in the AEC industry, very often deal with incomplete information
- × OWA assumes that information is incomplete by default
- × We allow others to extend our knowledge graphs
- × However, be careful, because houses can fly, and cars can be buildings...
- × Rule checking requires a CWA (SHACL)

Linked Building Data Ontologies



Where it all begun...

ifcOWL ontology (IFC4_ADD2_TC1)

language [en](#)

IRI: https://standards.buildingsmart.org/IFC/DEV/IFC4/ADD2_TC1/OWL

Date:
February 9, 2019

Authors:
Pieter Pauwels (pipauwel.pauwels@ugent.be)
Walter Terkaj (walter.terkaj@itia.cnr.it)

Contributors:
Jakob Beetz (j.beetz@tue.nl)
María Poveda Villalón (mpoveda@fi.upm.es)
Aleksandra Sojic (aleksandra.sojic@itia.cnr.it)

Imported Ontologies:
[express](#)
[list](#)

Download serialization:

[Format JSON LD](#) [Format RDF/XML](#) [Format N Triples](#) [Format TTL](#)

License:

[License <http://creativecommons.org/licenses/by/3.0/>](#)

Abstract

This ifcOWL ontology is automatically generated from the EXPRESS schema 'IFC4_ADD2_TC1' using the 'IFC-to-RDF' converter developed by Pieter Pauwels (pipauwel.pauwels@ugent.be), based on the earlier versions from Jyrki Oraskari (jyrki.oraskari@aalto.fi) and Davy Van Deursen (davy.vandeursen@ugent.be). The ontology is identical to the ontology that is generated from the EXPRESS schema 'IFC4_ADD2_TC1.exp' using the 'genOnto' converter developed by Walter Terkaj (ITIA-CNR, walter.terkaj@itia.cnr.it).

Web address: https://standards.buildingsmart.org/IFC/DEV/IFC4/ADD2_TC1/OWL Namespace: https://standards.buildingsmart.org/IFC/DEV/IFC4/ADD2_TC1/OWL

Table of contents

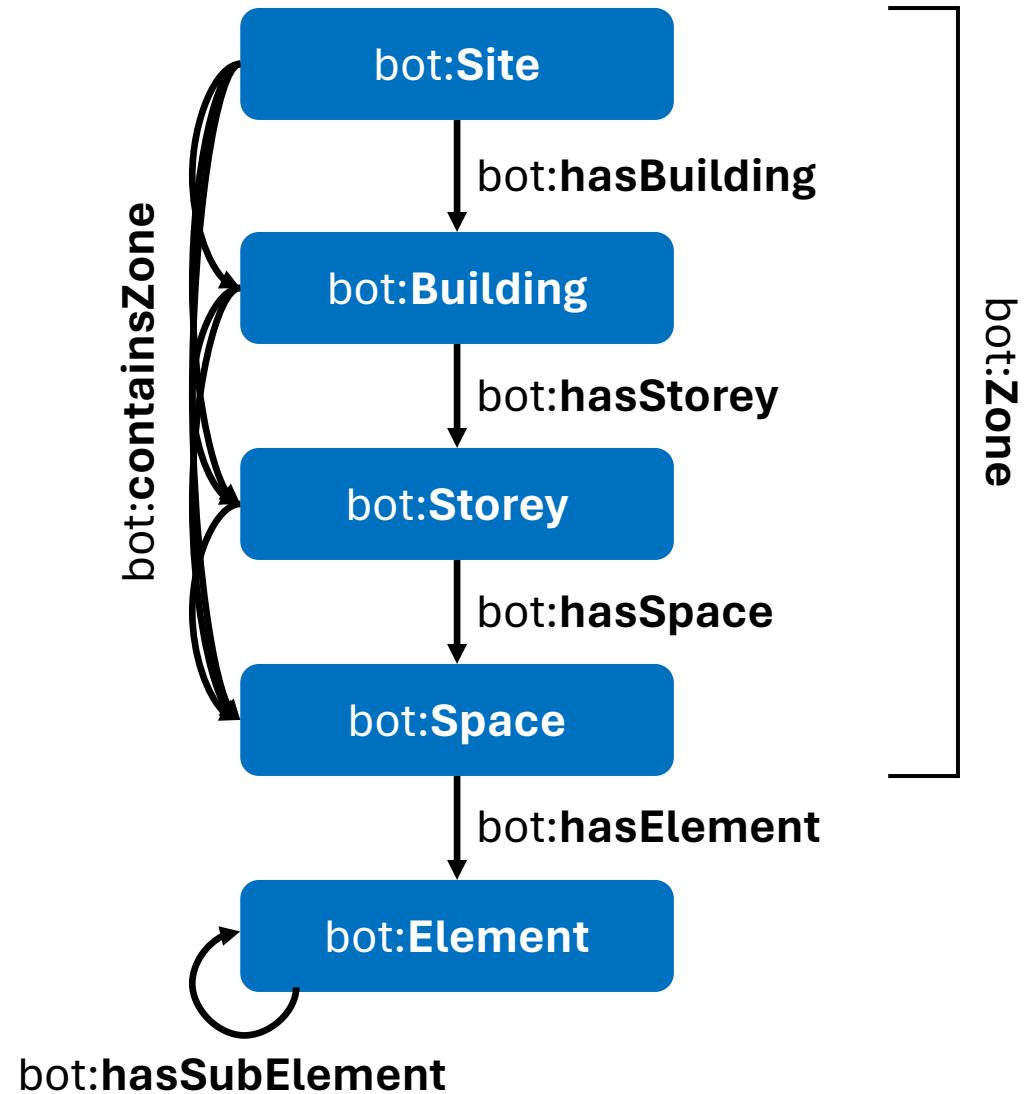
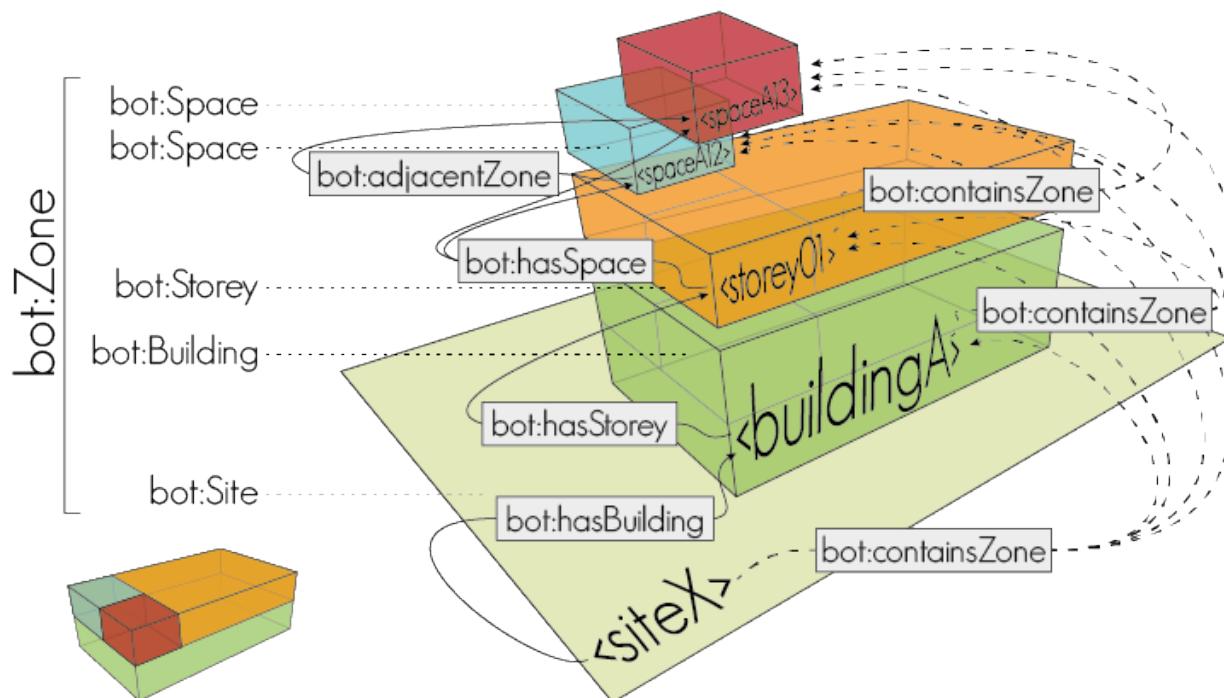
- 1. [Introduction](#)
 - 1.1. [Namespace declarations](#)
 - 1.2. [IFC4_ADD2_TC1: Overview](#)
- 2. [Cross reference for IFC4_ADD2_TC1 classes, properties and dataproperties](#)

... to where we are now

- × **Modular**, small, lightweight ontologies
- × Plug-and-play
- × Core ontologies representing basic building information
- × Domain-specific ontologies representing specific knowledge

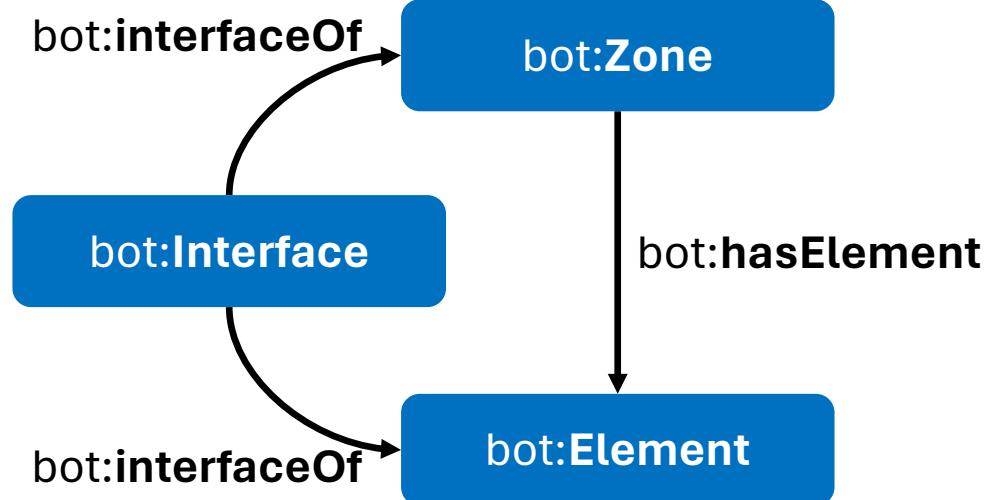
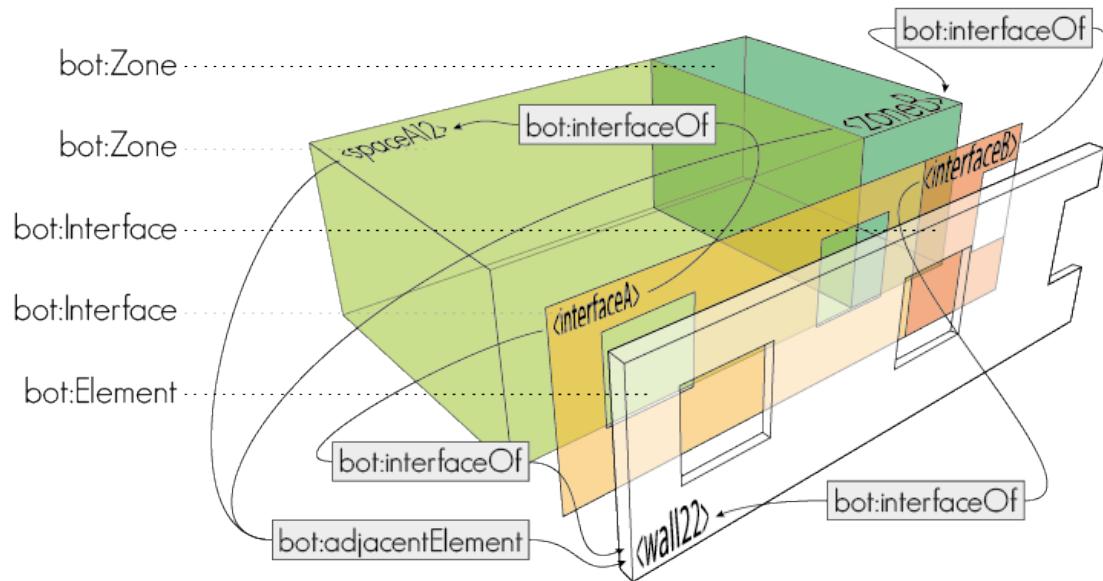
@prefix bot: <<https://w3id.org/bot#>>

Building Topology Ontology BOT

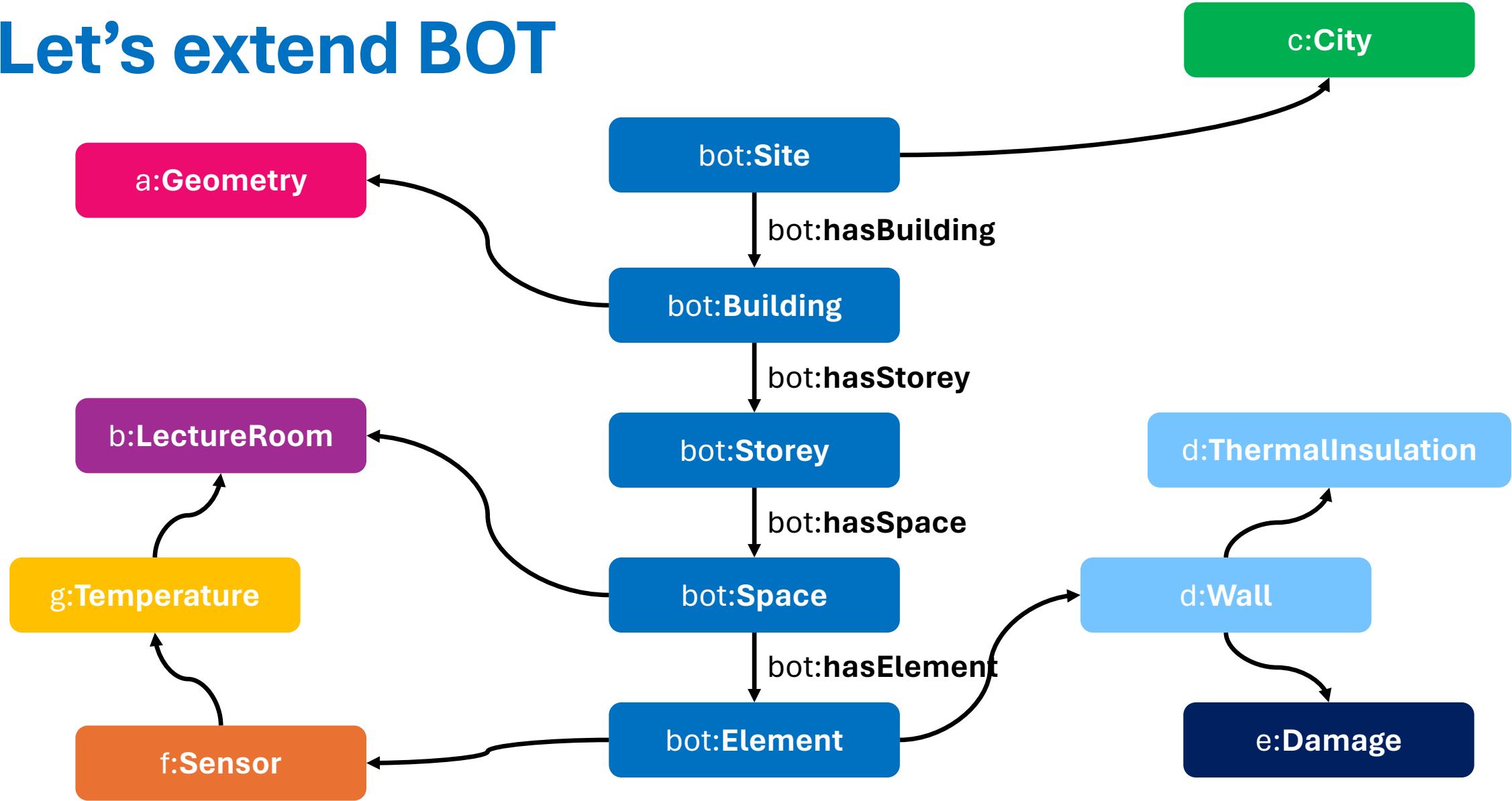


Building Topology Ontology BOT

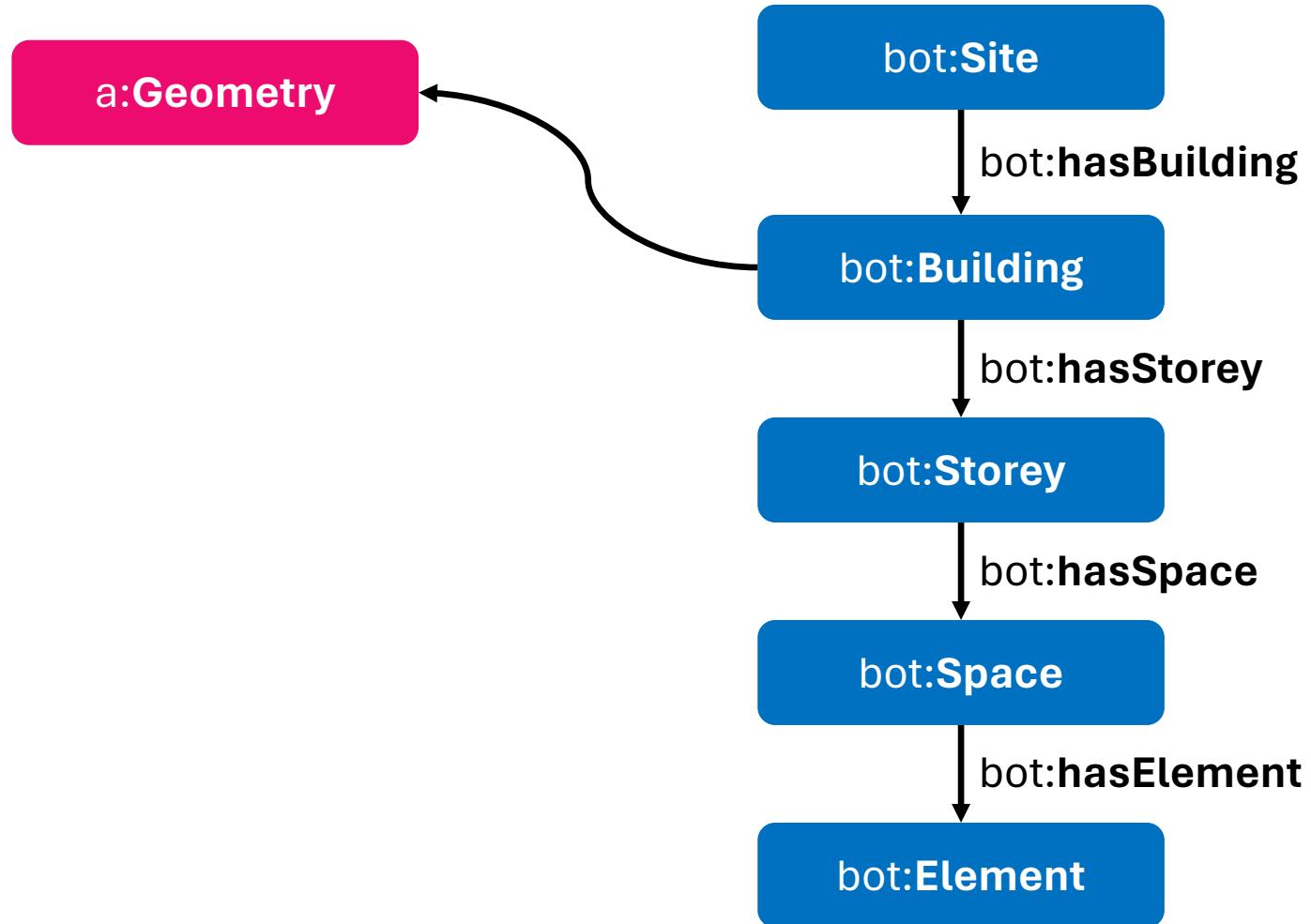
@prefix bot: <<https://w3id.org/bot#>>



Let's extend BOT

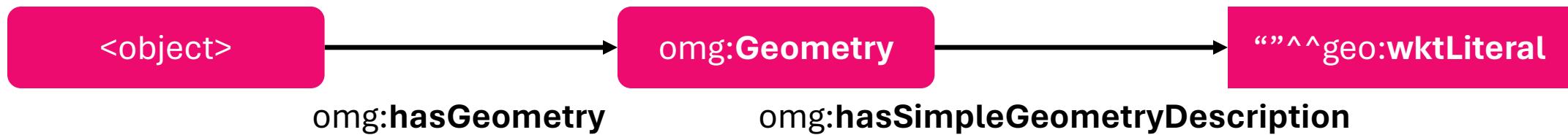


Let's extend BOT with geometry

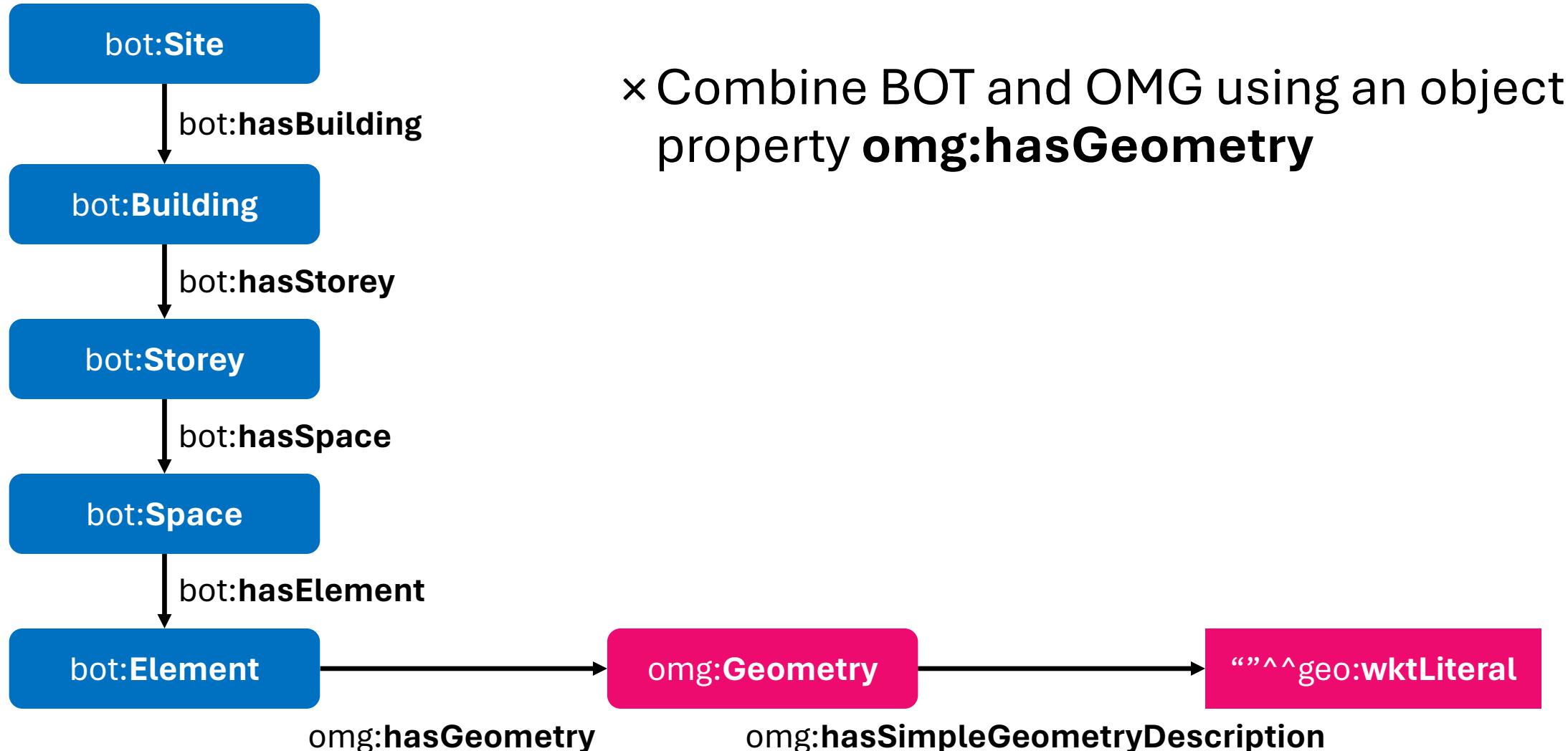


@prefix **omg**: <<https://w3id.org/omg#>>

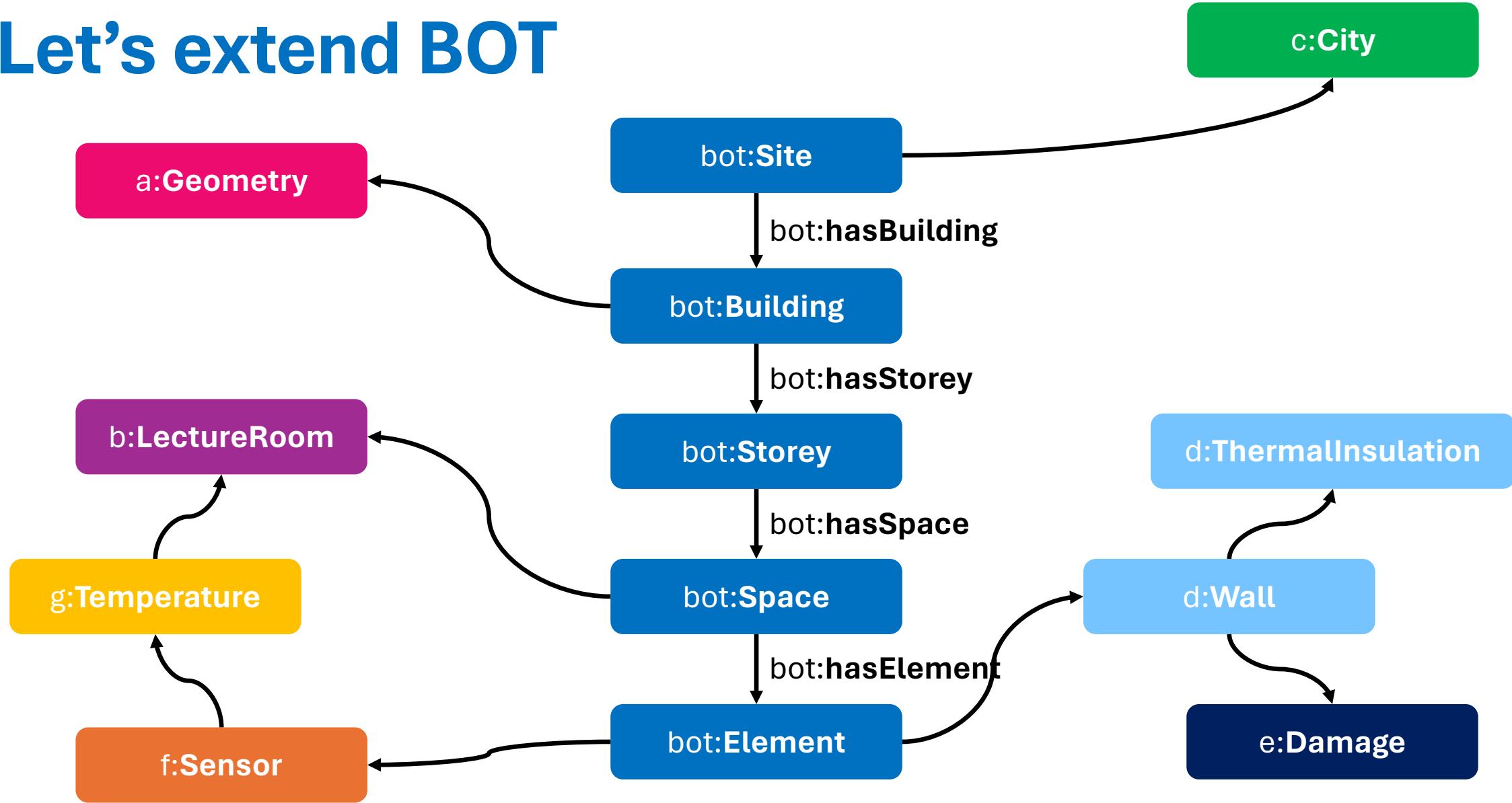
Ontology for Managing Geometry OMG



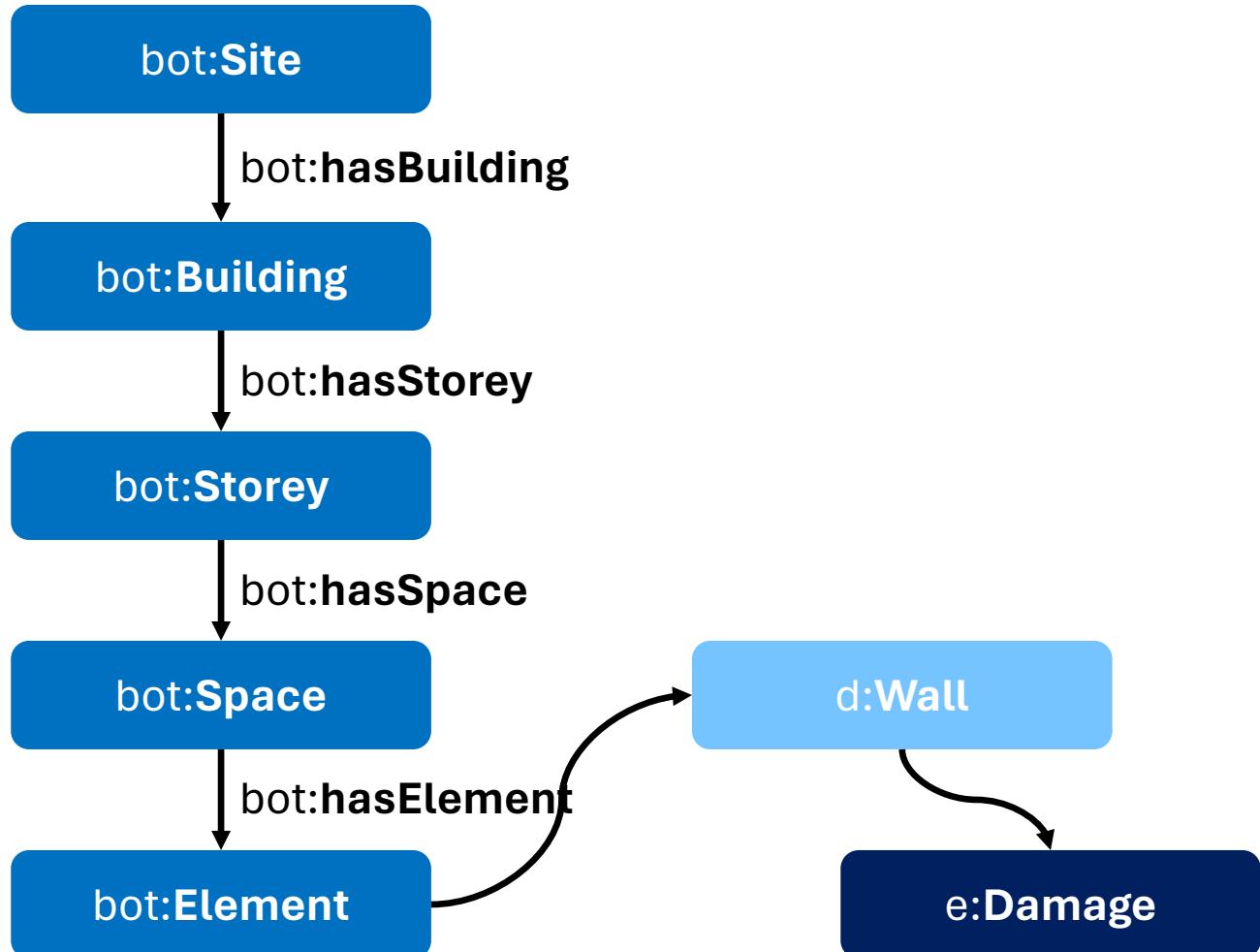
Ontology for Managing Geometry OMG



Let's extend BOT

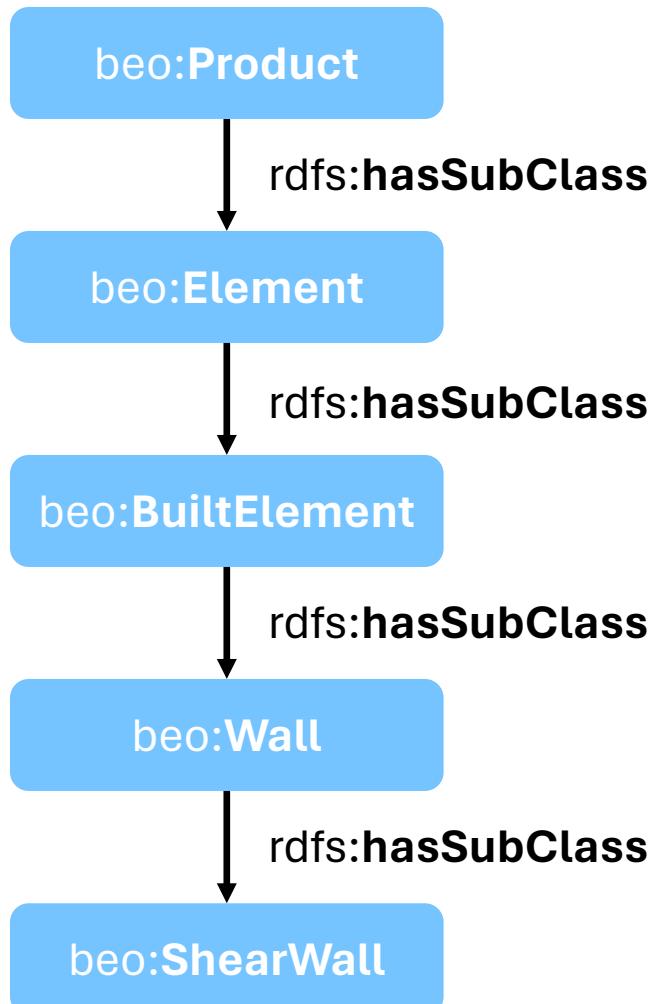


Let's extend BOT with specific elements



@prefix beo: <<https://w3id.org/beo#>>

Built Element Ontology BEO

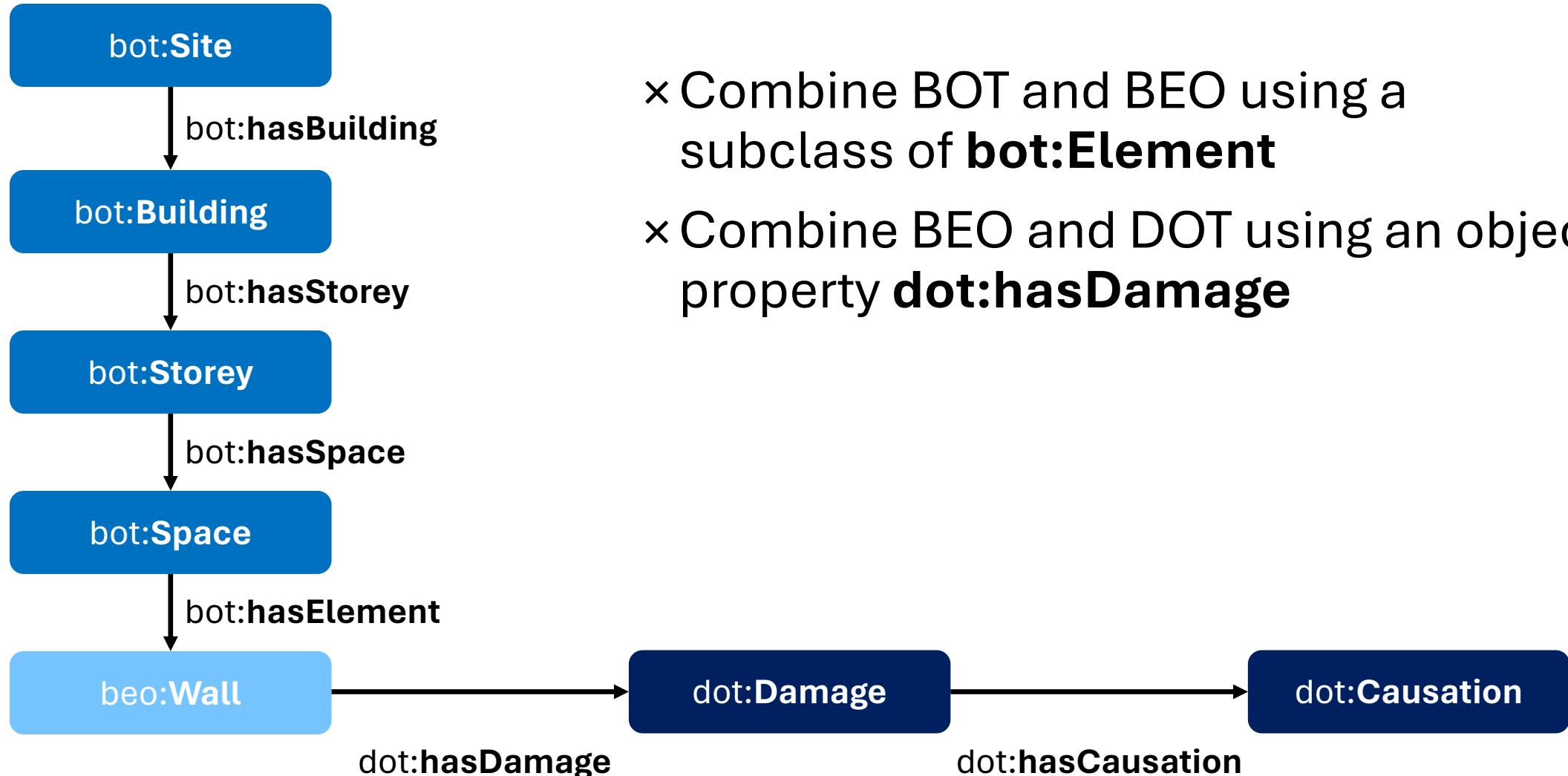


@prefix dot: <<https://w3id.org/dot#>>

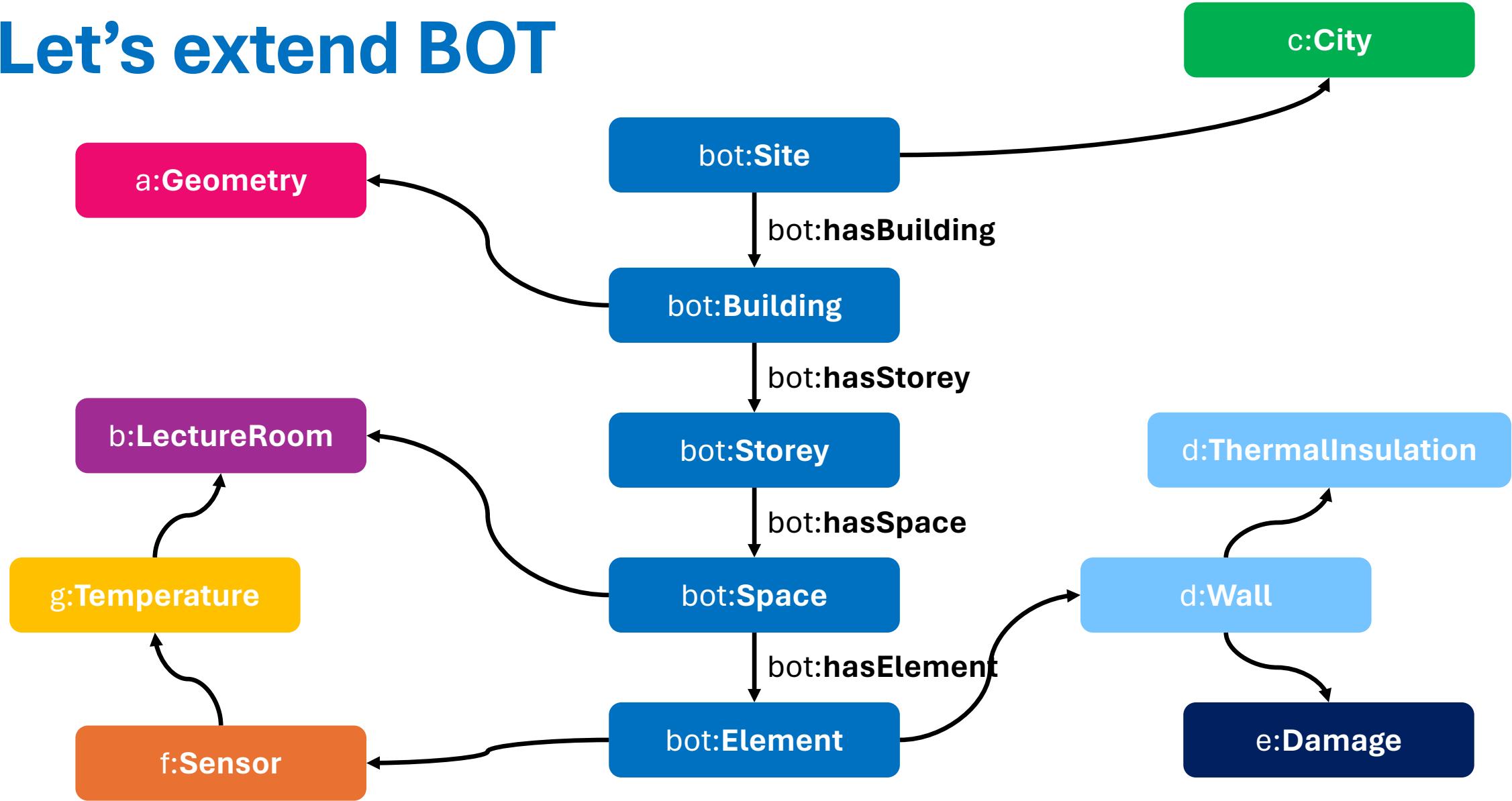
Damage Topology Ontology DOT



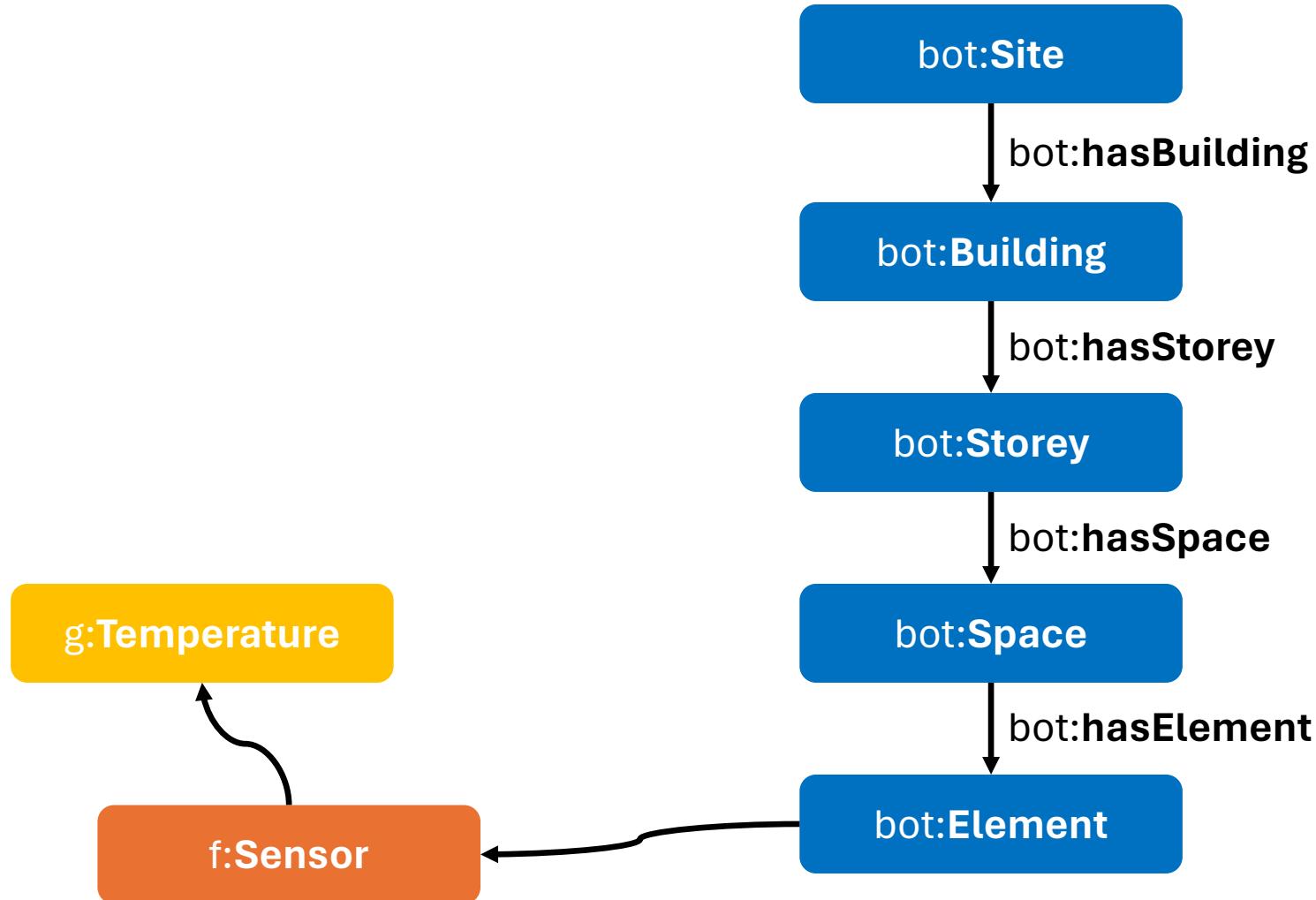
Built Element Ontology BEO



Let's extend BOT



Let's extend BOT with sensor data



@prefix ssn: <<https://w3.org/ns/ssn/>>

Semantic Sensor Network SSN

@prefix sosa: <<https://w3.org/ns/sosa/>>

Sensor, Observation, Sample, and Actuator

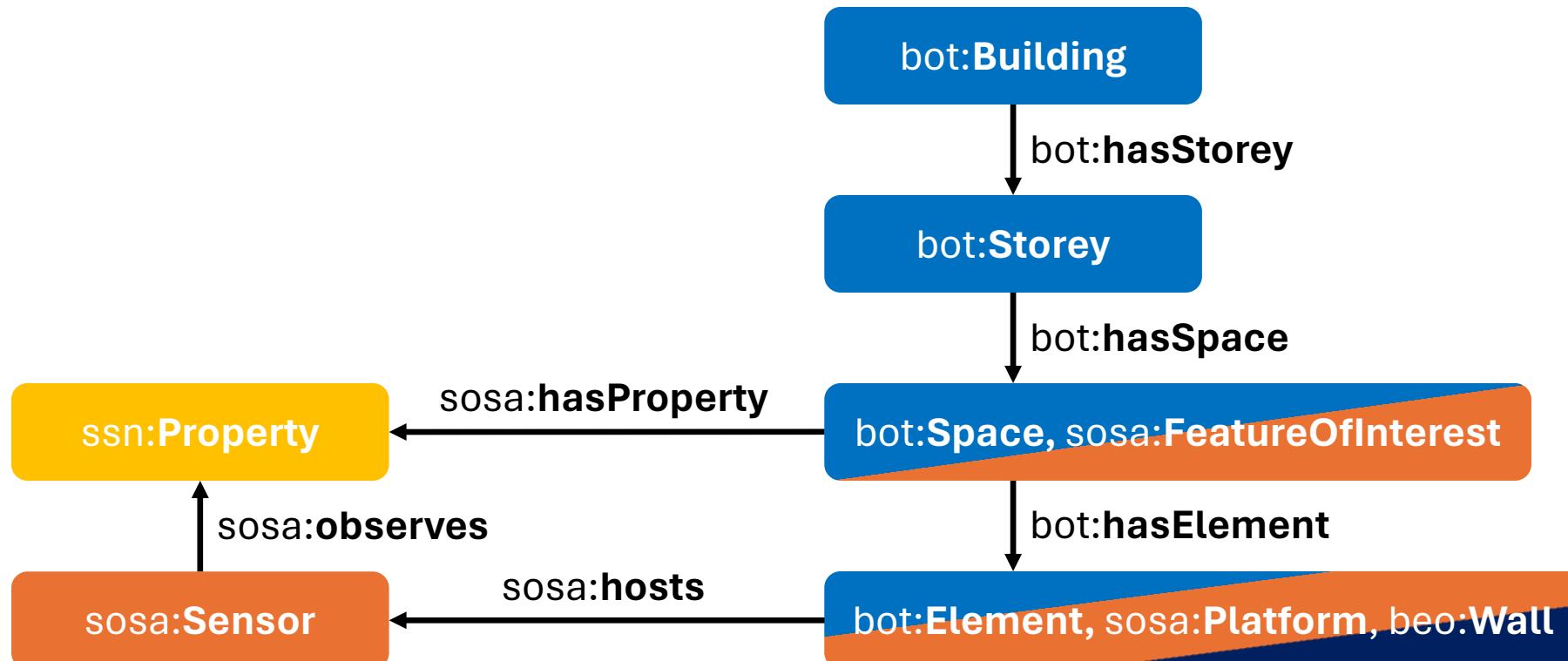


@prefix ssn: <<https://w3.org/ns/ssn/>>

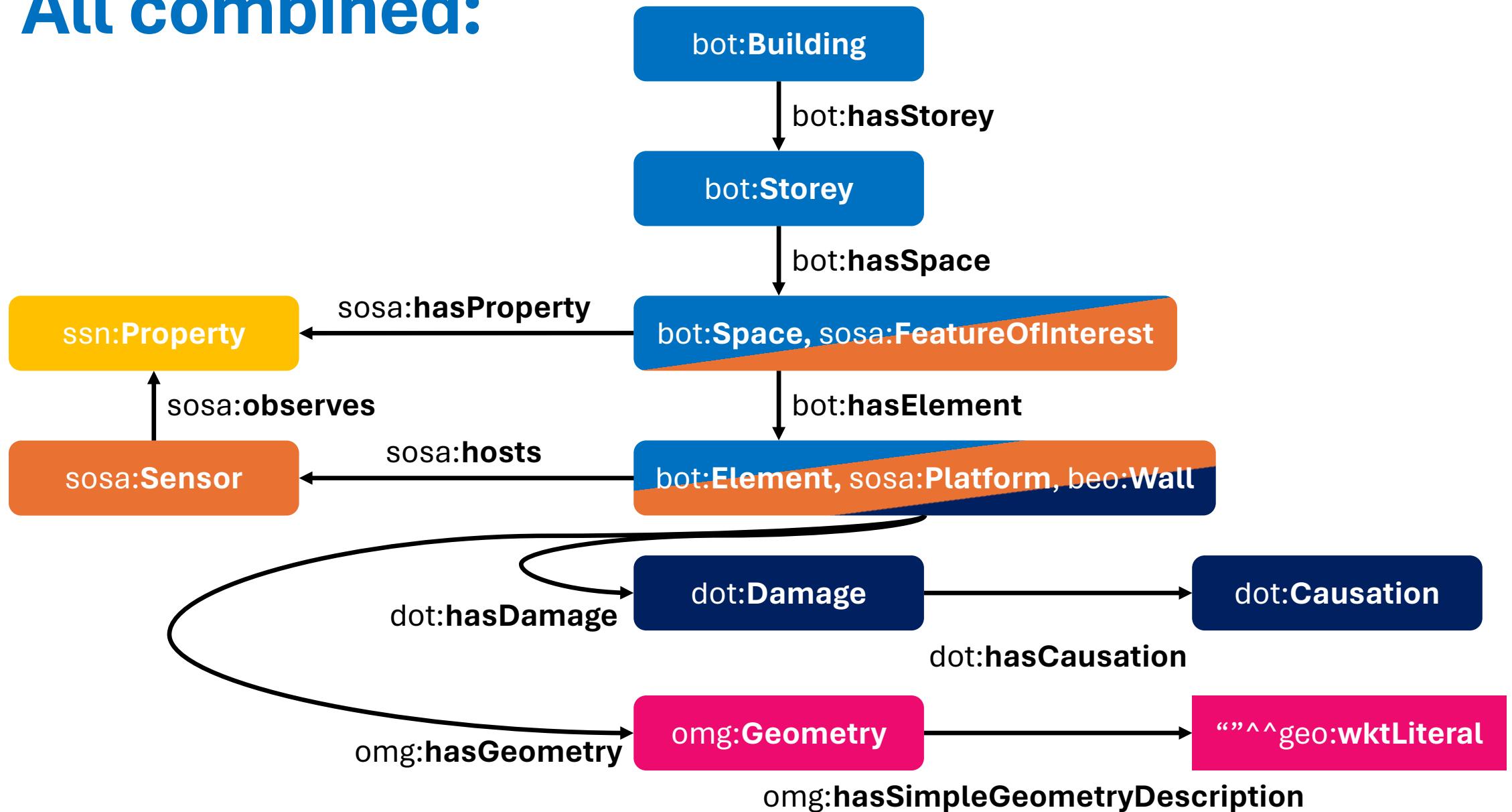
Semantic Sensor Network SSN

@prefix sosa: <<https://w3.org/ns/sosa/>>

Sensor, Observation, Sample, and Actuator



All combined:



Common LBD ontologies

Buildings: [BOT](#), [DogOnt](#), [SAREF4BLDG](#), [REC](#), [ifcOWL](#)

Element taxonomies: [BEO](#)

Properties: [OPM](#), [BOP](#)

Sensors: [SOSA/SSN](#), [SAREF](#), [SEAS](#), [BOP](#)

Geometry: [OMG](#), [GeoSPARQL](#), [GOM](#), [FOG](#)

HVAC: [Brick](#)

Provenance, metadata: [PROV-O](#), [DCAT](#), [DublinCore](#)

... and many, many more!

Ontology alignment

Tbox using formal alignment modules. E.g. Brick and Bot:

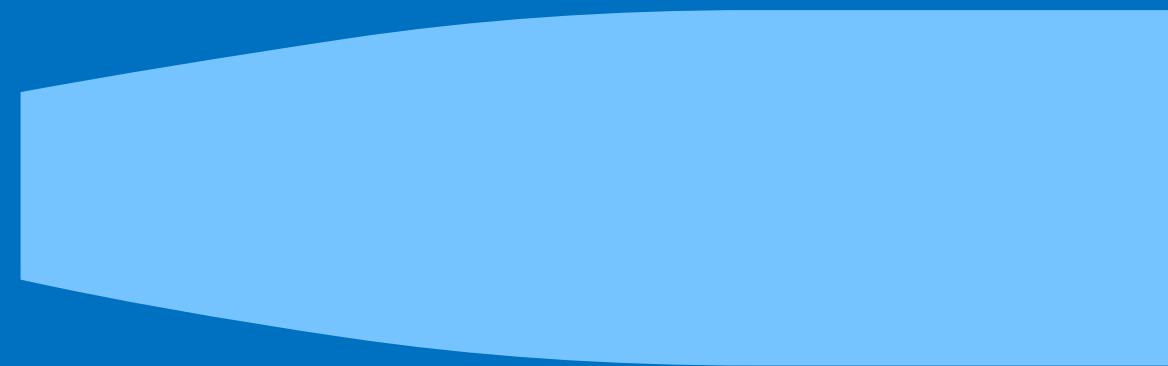
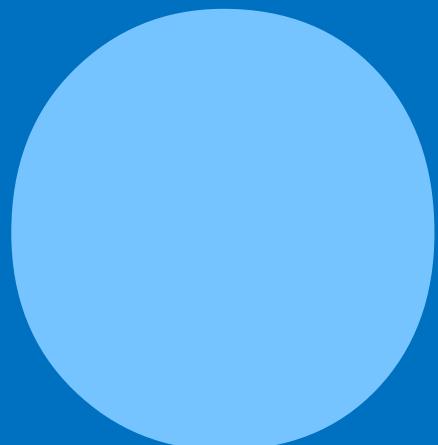
brick:Building owl:equivalentClass bot:Building .

Abox by simply using multiple rdf:type statements:

inst:RUB-Makerspace

rdf:type bot:Building, brick:Building .

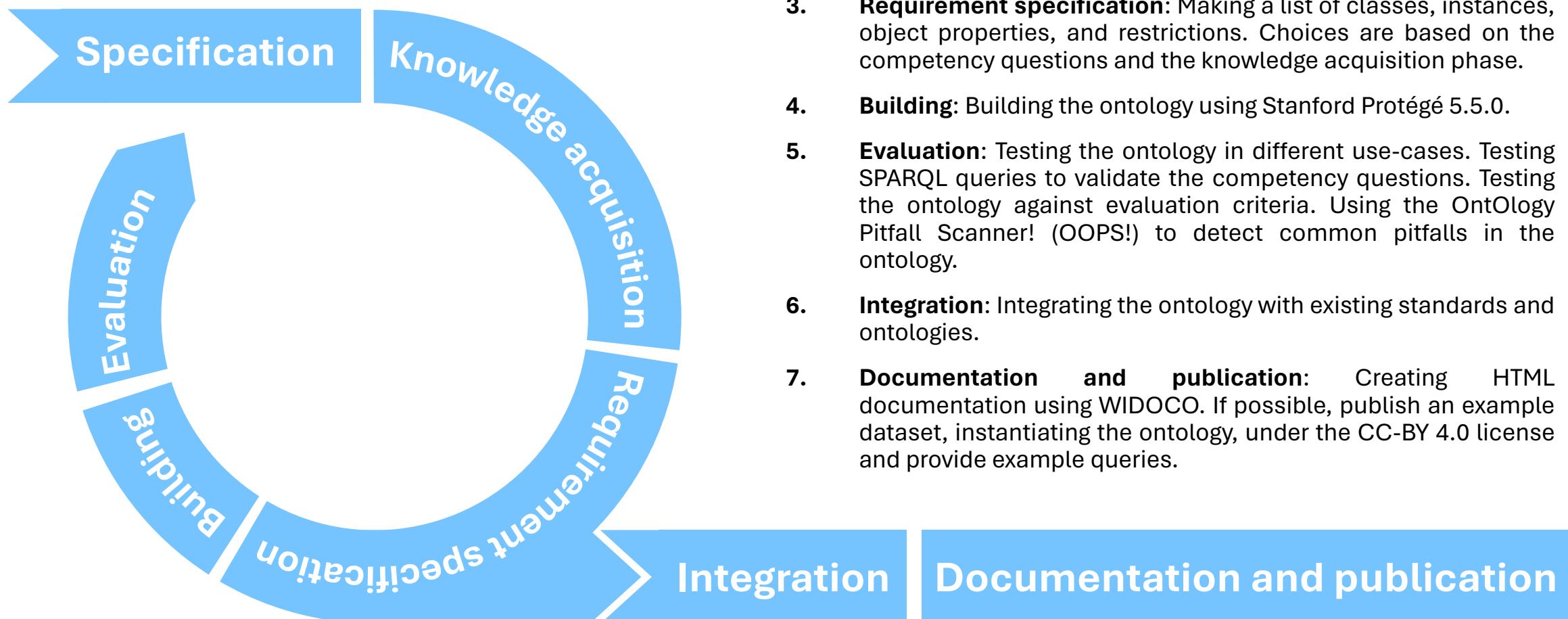
Practical tips and tools



Ontology design cycle

Donkers et al., 2022.

<https://doi.org/10.3390/buildings12101522>



Ontology editors

User interface:

- ✗ [Stanford Protégé](#)
- ✗ [TopBraid EDG](#)
- ✗ [Laces Library Manager](#)

Powerpoint:

- ✗ [Ontology Design Template](#)

UML to OWL:

- ✗ [Chowlk](#)
- ✗ [EA-to-RDF](#)

Tips:

- ✗ [W3C Best Practices](#)
- ✗ [W3C Cool URLs](#)
- ✗ [Other W3C standards](#)

Validators

Ontologies

✗ [OOPS!](#)

Instance data

✗ [RDF TTL syntax validator](#)

Publishing ontologies

Making HTML documentation

✗ [WIDOCO](#)

✗ [LODE](#)

Uploading to GitHub: <https://github.com/w3c-lbd-cg/bot>

Publish using GitHub Pages: <https://w3c-lbd-cg.github.io/bot>

Publishing ontologies (2)

Claiming a prefix: <http://prefix.cc>

Claim a w3 perma id: <https://github.com/perma-id/w3id.org>

Publish the ontology on [Linked Open Vocabularies](#)

Publish on bSDD ([buildingSmart Data Dictionary](#))

Talk about it!

Occupant Feedback Ontology

Latest version:
<https://w3id.org/ofo#>

Authors:
Alex Donkers

Contributors:
Bauke de Vries, ISBE, Eindhoven University of Technology
Dujuan Yang, ISBE, Eindhoven University of Technology

Downloads:
Format [JSON LD](#) Format [RDF/XML](#) Format [N Triples](#) Format [TTL](#)
License [CC BY 4.0](#) Visualize with [WebVowl](#)

Cite as:
Donkers, A.J.A., De Vries, B., Yang, D. (2022) Occupant Feedback Ontology Revision: 1.0.

language



Abstract

The aim of the Occupant Feedback Ontology is to semantically describe passive and active occupant feedback and to enable integration of this feedback with linked building data.

Namespace: <https://w3id.org/ofo#>

Suggested prefix: `ofo:`

A small extension module of the `ofo:Property` class has been created, as this enhances querying possibilities in indoor environmental quality use cases. The extension module is documented at:

Namespace: <https://alexdonkers.github.io/opt#>

The Occupant Feedback Ontology extends the Building Performance Ontology (BOP). Together, these ontologies can help creating rich semantic digital twins that combine sensor data, building properties and occupants and their opinions.

Namespace: <https://w3id.org/bop#>

Table of contents

- 1. Introduction
 - 1.1. Namespace declarations
- 2. Occupant Feedback Ontology: Overview
- 3. Cross reference for Occupant Feedback Ontology classes, properties and dataproPERTIES
 - 3.1. Classes
 - 3.2. Object Properties
 - 3.3. Data Properties
 - 3.4. Named Individuals
- 4. Alignment Modules
- 5. Acknowledgments

W3C Linked Building Data Group



If you like to contribute to those ontologies, discuss ongoing developments or learn from other academics and industry professionals, please join our W3C LBD CG.

Triweekly technical meetings with presentations on linked building data.

Linked Building Data and Ontologies

