

# Building Ontology Documentation

**Anderson Wong**

Urban Data Centre ([urbandatacentre.ca](http://urbandatacentre.ca))

School of Cities ([schoolofcities.ca](http://schoolofcities.ca))

University of Toronto

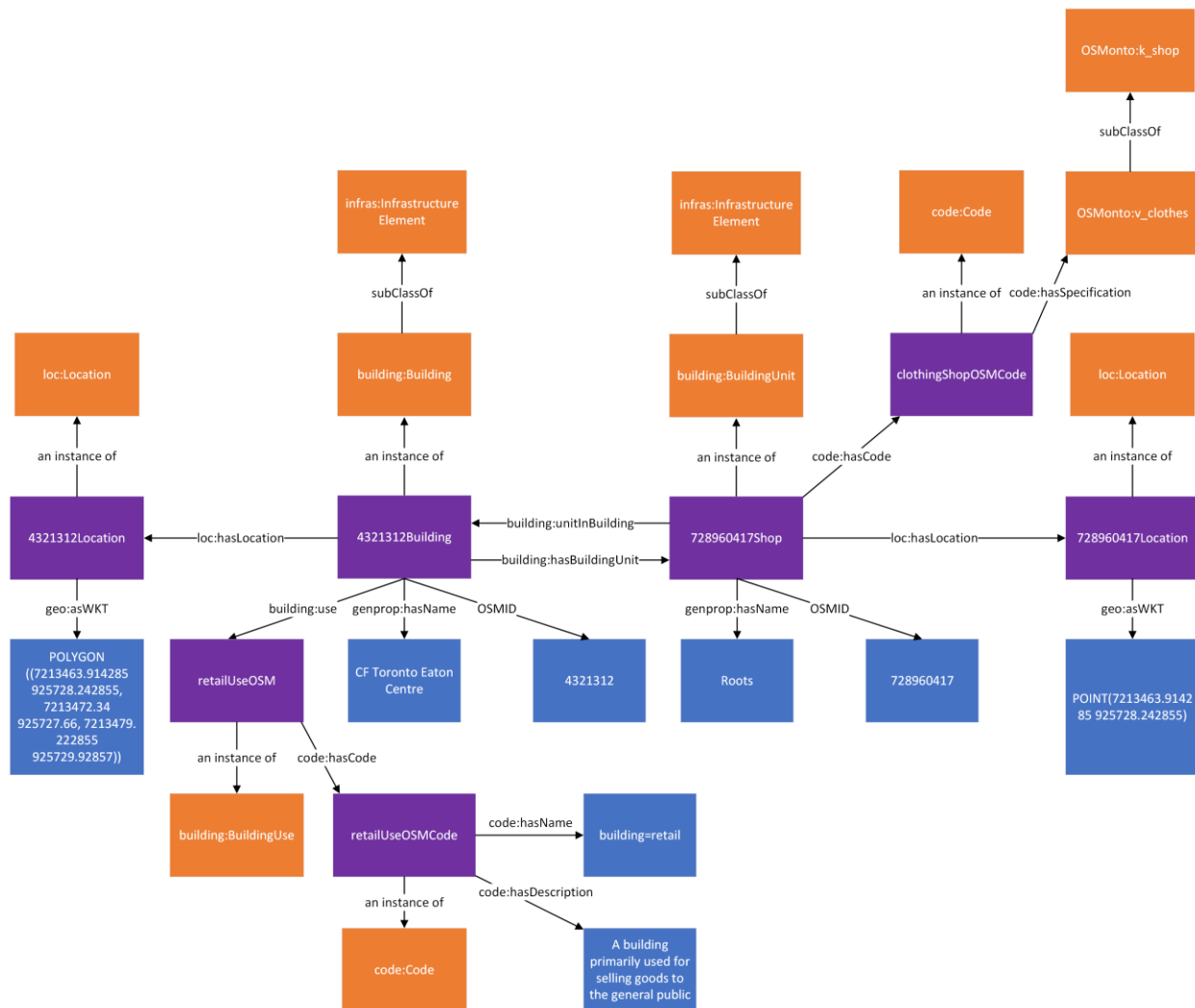
15 December 2023

## Introduction

This is a basic ontology for representing building data using the Building pattern from ISO/IEC 5087-2. An instance of a building (e.g. CF Toronto Eaton Centre as seen in the diagram below) is defined as an instance of the Building class from ISO/IEC 5087-2 and can be linked to the name of the building using the `hasName` property as well as its unique OpenStreetMap identifier using the `OSMID` property. Additionally, Buildings can be linked to their BuildingUse according to the BuildingUse pattern from ISO/IEC 5087-2. The `hasLocation` property is used to link Buildings to their Location instance which represents the geospatial location of the Building. The `asWKT` property shows the geospatial information as a WKT value. Buildings can also be linked to the individual BuildingUnits that may be found inside them using the `hasBuildingUnit` property. Conversely, BuildingUnits can be linked to the Buildings in which they are located in using the `unitInBuilding` property. A list of the key properties is shown below:

- **genProp:hasName:** Indicates the name of an entity
- **loc:hasLocation:** Identifies the Location instance that represents the entity's geospatial location
- **geo:asWKT:** Indicates the geospatial information of an entity using the WKT (well-known text) representation of geometry
- **code:hasCode:** Links to a Code instance that represents the classification system used by OpenStreetMap
- **building:hasBuildingUnit:** Identifies the BuildingUnit found within the Building
- **building:unitInBuilding:** Identifies the Building in which the BuildingUnit is located within
- **OSMID:** Indicates the unique identifier that is used by OpenStreetMap

A diagram of the ontology using the CF Toronto Eaton Centre as an example can be found below. Orange boxes represent classes, purple boxes represent instances, and blue boxes represent literals.



Namespace prefixes used:

- building: <https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Building/>
- code: <https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Code/>
- genprop: <https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/GenericProperties/>
- geo: <http://www.opengis.net/ont/geosparql#>
- infras: <https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Infrastructure/>
- loc: <https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/SpatialLoc/>
- OSMonto: <https://raw.github.com/doroam/planning-do-roam/master/Ontology/tags.owl>

The following table shows the key classes and properties:

Class	Property	Value Restriction
building:Building	rdfs:subClassOf	infras:InfrastructureElement
	building:hasBuildingUnit	building:BuildingUnit

	genProp:hasName	only xsd:string
	OSMID	only xsd:integer
	loc:hasLocation	only loc:Location
Building:BuildingUnit	loc:hasLocation	only loc:Location
	genProp:hasName	only xsd:string
	code:hasCode	only code:Code
	building:unitInBuilding	building:Building
	OSMID	only xsd:integer
loc:Location	geo:asWKT	only xsd:string
code:Code	code:hasName	only xsd:string
	code:hasDescription	only xsd:string
	code:hasSpecification	only xsd:anyURI