General City Digital Twin User Manual

Anderson Wong

Urban Data Research Centre (urbandatacentre.ca)
School of Cities (schoolofcities.ca)
University of Toronto

22 April 2025

Table of Contents

Table of Contents 1
Introduction 2
Accessing the Dashboard 2
Updating the Graph Database 3
Representation of City Data 4
- Administrative Areas 4
- Address 5
- Geospatial Location 6
- Census Profiles and Characteristics 7
- Toronto Police Service Crime Data 8
- OpenStreetMap Tags Mapping 12
- Complete Community Walking Distance Indicator 15
- Parks 16
- Schools 17
- Kindergartens 18
- Universities 19
- Colleges 20
- Hospitals 21
- Clinics 23

Doctor’s Office 24
Pharmacy........ 25
Supermarket 26
Greengrocer 27
Restaurant 28
FastFood 29
Representation of Road Data 30
Junction 32
Road 34
RoadLink........ 35
Integration of ORN Data in TTL Generation........ 46
Custom Classes and Properties........ 48
....

Introduction

The City Digital Twin project at the University of Toronto’s Urban Data Research Centre aims to develop a digital twin of the City of Toronto. It builds on existing city data standards such as ISO/IEC 21972:2020, ISO/IEC 5087-1:2024, and ISO/IEC 5087-2:2024, and conforms to Linked Data requirements. The goal of this project is to demonstrate standards-based semantic interoperability of city data. These standard ontologies enable the integration of data from multiple sources, opening up new possibilities for the development of data analysis and visualization tools. Our City Digital Twin is represented as a knowledge graph containing a growing variety of city data (e.g. Canadian Census, Toronto Police Crime Data, City transportation infrastructure, parks, food stores, schools, …). Our City Digital Twin also supports a dashboard that can generate data visualizations using the city data from the graph database.

Our City Digital Twin is general in the sense the ontologies and standards being used to represent city data are city independent. Therefore, the CDT can be applied to any city with modification.

Accessing the Dashboard

The City Digital Twin Dashboard can be directly accessed using the following link: http://ec2-3-97-59-180.ca-central-1.compute.amazonaws.com:3001/.

Alternatively, it can be downloaded from GitHub (https://github.com/csse-uoft/city-digital-twin/tree/develop) and run locally. You can run it locally on your computer by completing the following steps:

1. Download the "develop" branch of the City Digital Twin from our GitHub: https://github.com/csse-uoft/city-digital-twin/tree/develop. If you downloaded the code as a .zip file, unzip the file before proceeding to the next step.
2. Open the Command Prompt. On Windows 10 or 11, you can do this by pressing the Windows key and typing "cmd". For Mac OS users, you can open Terminal instead by clicking the Launchpad icon and typing Terminal in the search field.
3. In the Command Prompt, navigate to the unzipped folder that you downloaded from GitHub using the "cd" command (e.g., "cd folder_name").
4. Enter and run "cd frontend" to navigate to the "frontend" folder
5. Enter and run "npm install" in the Command Prompt to install the dependencies.
6. Repeat steps 4 and 5 for the "backend" folder (you can return to the parent file directory by using "cd..")
7. While in the "backend" folder, enter and run "npx nodemon index.js" to start up the backend of the dashboard
8. Open a new Command Prompt window by repeating step 2
9. In this new Command Prompt window, navigate to the unzipped folder by repeating step 3
10. Enter and run "cd frontend" to navigate to the "frontend" folder
11. Enter and run "npm start" to start up the frontend of the dashboard
12. Once the above steps are completed, the dashboard should be up and running at a localhost address specified in the second Command Prompt window. You can enter this localhost address into a web browser to access the dashboard.

Updating the Graph Database

The graph database for the City Digital Twin project can be accessed here: http://ec2-3-97-59-180.ca-central-1.compute.amazonaws.com:7200/. To update the contents of the graph database, please refer to the steps below as a guide and more detailed information about how to use SPARQL to update graphs along with examples can be found here: https://docs.progress.com/bundle/marklogic-server-develop-with-semantic-graphs-11/page/topics/sparql-update.html. Official SPARQL Update documentation can also be found here: https://www.w3.org/TR/sparql12-update/.

1. If triples need to be removed from the graph database:
   a. If an entire graph needs to be deleted, you can run a "drop graph" query in SPARQL to remove the specified graph: DROP GRAPH <insert_graph_IRI>
   b. If specific triples need to be deleted from the graph database:
      i. You can run a "delete data" query if you are deleting individual triples explicitly or:
      ii. A "delete where" query if you are deleting patterns of triples
2. If triples need to be added to the graph database:
   a. If an entire RDF file (e.g. a TTL or OWL file) needs to be uploaded to the graph database, you can upload and import the file in GraphDB Workbench using its user interface:
      i. Click on the "Import" button on the left sidebar in GraphDB Workbench
      ii. Click on "Upload RDF files" button and upload your RDF file(s)

iii. After the file(s) have been uploaded, click on the “Import” button next to the uploaded file and a popup window will appear
iv. It is highly recommended that you specify the name of the target graph by clicking on “Named graph” under the “Target graphs” section and entering a name for the graph in the textbox
v. Click on the “Import” button in the bottom right corner of the popup window when you are ready to import the data
vi. An “Imported successfully” message should show if the update was successful

b. If specific triples need to be added to the graph database:
   i. You can run a “insert data” query if you are adding explicitly defined triples or:
   ii. A “insert where” query if you are adding patterns of triples

Representation of City Data

The following sections will provide an overview of how the city data in the graph database is ontologically represented.

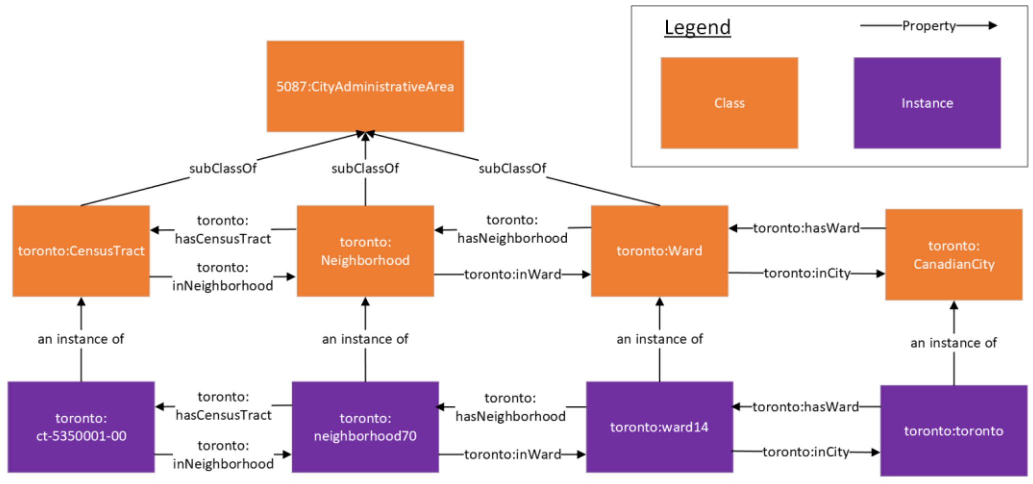
The following is a list of namespace prefixes:

- cdt: http://ontology.eil.utoronto.ca/CDT#
- code: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Code/
- contact: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Contact/
- crime: http://ontology.eil.utoronto.ca/CKGN/Crime#
- foaf: http://xmlns.com/foaf/0.1/
- gcie: http://ontology.eil.utoronto.ca/GCI/Education/GCI-Education.owl#
- gcir: http://ontology.eil.utoronto.ca/GCI/Recreation/GCIRecreation.owl#
- genprop: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/GenericProperties/
- geo: http://www.opengis.net/ont/geosparql#
- geof: http://www.opengis.net/def/function/geosparql/
- iso21972: http://ontology.eil.utoronto.ca/ISO21972/iso21972#
- iso50871: http://ontology.eil.utoronto.ca/5087/1/SpatialLoc/
- iso5087m: http://ontology.eil.utoronto.ca/5087/1/Mereology/
- loc: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/SpatialLoc/
- rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
- rdfs: http://www.w3.org/2000/01/rdf-schema#
- time: http://www.w3.org/2006/time#
- toronto: http://ontology.eil.utoronto.ca/Toronto/Toronto#
- uoft: http://ontology.eil.utoronto.ca/tove/cacensus#
- xsd: http://www.w3.org/2001/XMLSchema#

Administrative Areas

Administrative areas are geographic regions that are defined for governance and administrative purposes. Examples include areas like neighborhoods or city wards as seen in the below diagram and are expressed as a subclass of the CityAdministrativeArea class from ISO 5087-2. These administrative areas also have properties such as ”hasNeighborhood” or ”inWard” to describe the relationships between each other. These properties are useful for quickly finding, for example, a list of neighborhoods that can be found inside a given ward.

Formatted: Font: 11 pt

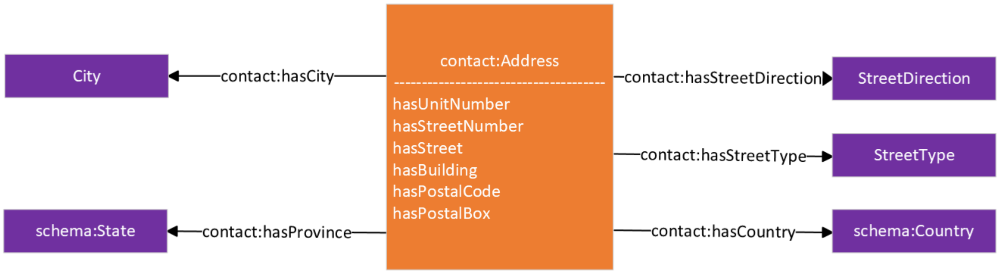


An example SPARQL query for finding a list of neighborhoods in ward 1 (Etobicoke North) can be found below:

```
PREFIX toronto: <http://ontology.eil.utoronto.ca/Toronto/Toronto#>

SELECT ?neighborhood

WHERE{

toronto:ward1 toronto:hasNeighborhood ?neighborhood

}
```

Address

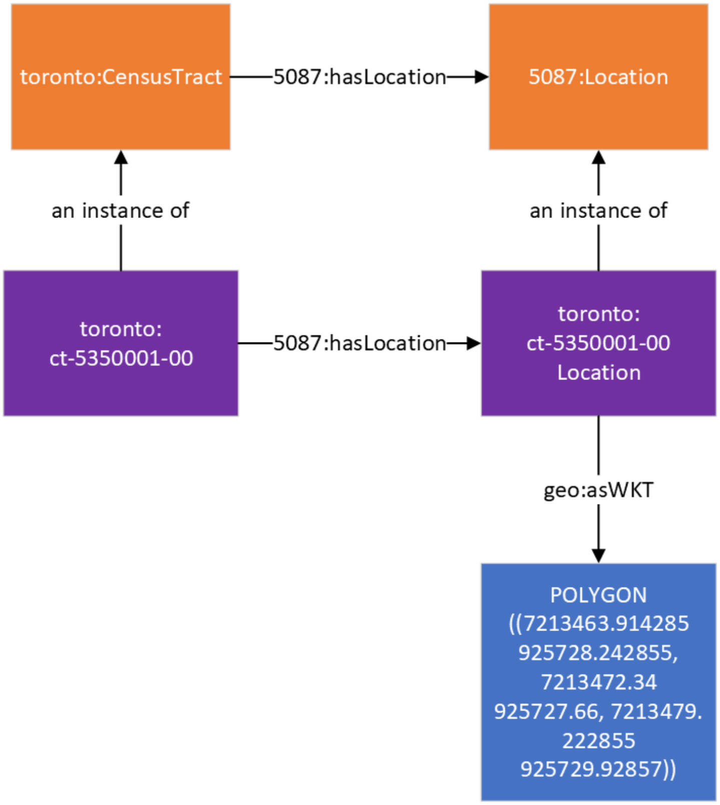
The address of an entity is represented using the Address representation from the Contact pattern from ISO 5087-2 where an entity is linked to an Address instance using the “hasAddress” property. This Address instance is then linked to individual components of an address using properties such as hasStreetNumber, hasStreet, hasPostalCode, etc.



Geospatial Location

The geospatial location of an entity is represented using the Location pattern from ISO 5087-1 where an entity is linked to a Location class instance using the "hasLocation" property. The location instance is then linked to the WKT polygon coordinates using the "asWKT" property.

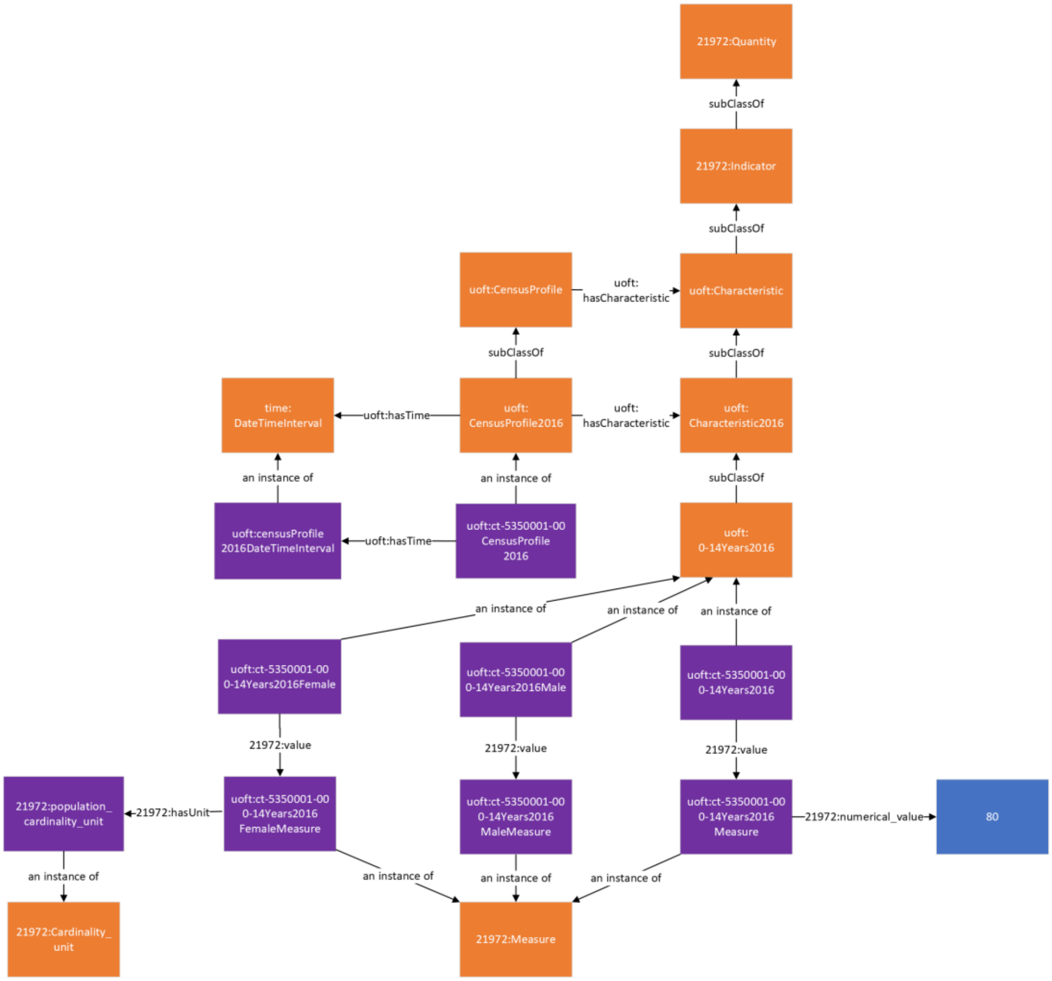
Formatted: Font: 12 pt, Font color: Auto



Census Profiles and Characteristics

Census data for geographic areas are organized as datasets called census profiles. Census profiles contain a collection of individual census characteristics, which are a subclass of the Indicator class from ISO 21972 and linked using the hasCharacteristic property. Each characteristic also has a class that is used to represent it, and this class is also a subclass of the Characteristic class. For example, the 0-14Years2016 class in the diagram above represents the characteristic for the number of people who were between the ages 0 to 14 during the 2016 Canadian Census. An instance of this class represents the characteristic for a given geographic area. For example, the “ct-5350001-000-14Years2016” instance in the diagram below represents the characteristic for the number of people in the census tract 5350001.00 who were between the ages 0 to 14 in 2016. These characteristic instances are linked to an instance of the Measure class from ISO 21972 using the ”value” property and this measure instance is linked to the value of the characteristic using the ”numerical_value” property.

Formatted: Font: 12 pt, Font color: Auto



A more in-depth explanation of the Canadian Census ontology can be found in our paper titled “Semantically Interoperable Census Data: Unlocking the Semantics of Census Data Using Ontologies and Linked Data” which can be found here: https://ijpds.org/article/view/2378#:~:text=Using%20census%20data%20as%20an,data%20points%20from%20large%20datasets.

Toronto Police Service Crime Data

This is a basic ontology for representing neighborhood crime indicators from the Toronto Police Service using the Indicator pattern from ISO/IEC 21972. An instance of a crime indicator (e.g.

AssualtCrime2016 which measures the number of assault crimes that occurred during the year 2016 in a given location) is defined as a subclass of the Indicator class from ISO/IEC 21972. Crime indicators can be linked to their location using the hasLocation property and its corresponding time period using the hasTime property. The value of a crime indicator is defined as an instance of the Measure class from ISO/IEC 21972 and can be linked to its numerical value using the numerical_value property. Additionally, crime indicators can be linked to its population instance using the cardinality_of property from ISO/IEC 21972.

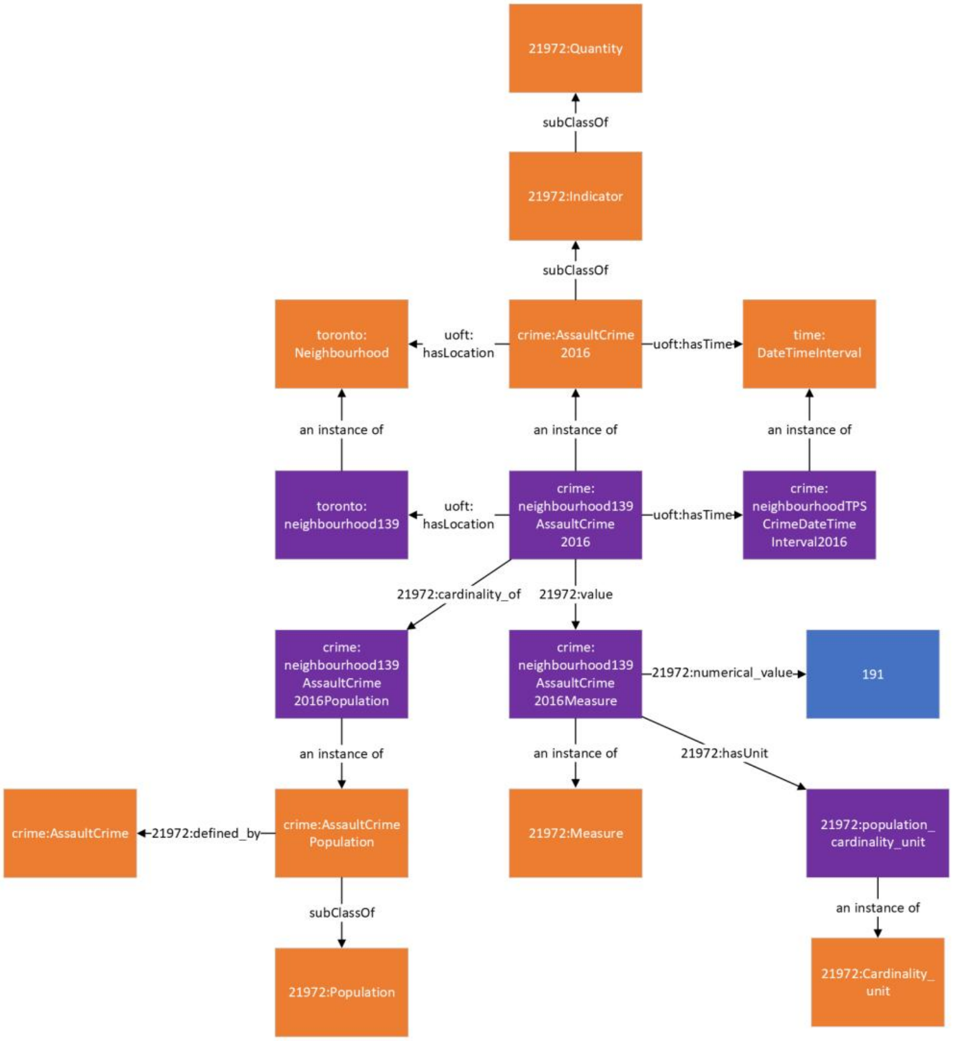
This section provides a brief description of the column labels found in the data spreadsheet published by the Toronto Police Service.

Toronto Police Service Label	Description
HoodName,C,35	Name of the neighborhood
HoodID,C,3	The neighborhood number
Assault_20,N,3,0	Number of assault crime incidents in 2014
Assault_1,N,3,0	Number of assault crime incidents in 2015
Assault_2,N,3,0	Number of assault crime incidents in 2016
Assault_3,N,3,0	Number of assault crime incidents in 2017
Assault_4,N,3,0	Number of assault crime incidents in 2018
Assault_5,N,3,0	Number of assault crime incidents in 2019
Assault_6,N,3,0	Number of assault crime incidents in 2020
Assault_7,N,3,0	Number of assault crime incidents in 2021
AutoTheft_,N,3,0	Number of auto theft crime incidents in 2014
AutoThef_1,N,3,0	Number of auto theft crime incidents in 2015
AutoThef_2,N,3,0	Number of auto theft crime incidents in 2016
AutoThef_3,N,3,0	Number of auto theft crime incidents in 2017
AutoThef_4,N,3,0	Number of auto theft crime incidents in 2018
AutoThef_5,N,3,0	Number of auto theft crime incidents in 2019
AutoThef_6,N,3,0	Number of auto theft crime incidents in 2020
AutoThef_7,N,3,0	Number of auto theft crime incidents in 2021
BreakAndEn,N,3,0	Number of break and enter crime incidents in 2014

BreakAnd_1,N,3,0	Number of break and enter crime incidents in 2015
BreakAnd_2,N,3,0	Number of break and enter crime incidents in 2016
BreakAnd_3,N,3,0	Number of break and enter crime incidents in 2017
BreakAnd_4,N,3,0	Number of break and enter crime incidents in 2018
BreakAnd_5,N,3,0	Number of break and enter crime incidents in 2019
BreakAnd_6,N,3,0	Number of break and enter crime incidents in 2020
BreakAnd_7,N,3,0	Number of break and enter crime incidents in 2021
Robbery_20,N,3,0	Number of robbery crime incidents in 2014
Robbery__1,N,3,0	Number of robbery crime incidents in 2015
Robbery__2,N,3,0	Number of robbery crime incidents in 2016
Robbery__3,N,3,0	Number of robbery crime incidents in 2017
Robbery__4,N,3,0	Number of robbery crime incidents in 2018
Robbery__5,N,3,0	Number of robbery crime incidents in 2019
Robbery__6,N,3,0	Number of robbery crime incidents in 2020
Robbery__7,N,3,0	Number of robbery crime incidents in 2021
TheftOver_,N,2,0	Number of thefts over $5000 crime incidents in 2014
TheftOve_1,N,2,0	Number of thefts over $5000 crime incidents in 2015
TheftOve_2,N,2,0	Number of thefts over $5000 crime incidents in 2016
TheftOve_3,N,2,0	Number of thefts over $5000 crime incidents in 2017
TheftOve_4,N,2,0	Number of thefts over $5000 crime incidents in 2018
TheftOve_5,N,2,0	Number of thefts over $5000 crime incidents in 2019
TheftOve_6,N,2,0	Number of thefts over $5000 crime incidents in 2020
TheftOve_7,N,2,0	Number of thefts over $5000 crime incidents in 2021
Homicide_2,N,1,0	Number of homicide crime incidents in 2014

Homicide_1,N,1,0	Number of homicide crime incidents in 2015
Homicide_3,N,1,0	Number of homicide crime incidents in 2016
Homicide_4,N,1,0	Number of homicide crime incidents in 2017
Homicide_5,N,1,0	Number of homicide crime incidents in 2018
Homicide_6,N,1,0	Number of homicide crime incidents in 2019
Homicide_7,N,1,0	Number of homicide crime incidents in 2020
Homicide_8,N,1,0	Number of homicide crime incidents in 2021
Shootings_,N,2,0	Number of shooting crime incidents in 2014
Shooting_1,N,2,0	Number of shooting crime incidents in 2015
Shooting_2,N,2,0	Number of shooting crime incidents in 2016
Shooting_3,N,2,0	Number of shooting crime incidents in 2017
Shooting_4,N,2,0	Number of shooting crime incidents in 2018
Shooting_5,N,2,0	Number of shooting crime incidents in 2019
Shooting_6,N,2,0	Number of shooting crime incidents in 2020
Shooting_7,N,2,0	Number of shooting crime incidents in 2021
TheftfromM,N,3,0	Number of thefts from motor vehicle crime incidents in 2014
Theftfro_1,N,3,0	Number of thefts from motor vehicle crime incidents in 2015
Theftfro_2,N,3,0	Number of thefts from motor vehicle crime incidents in 2016
Theftfro_3,N,3,0	Number of thefts from motor vehicle crime incidents in 2017
Theftfro_4,N,3,0	Number of thefts from motor vehicle crime incidents in 2018
Theftfro_5,N,3,0	Number of thefts from motor vehicle crime incidents in 2019
Theftfro_6,N,3,0	Number of thefts from motor vehicle crime incidents in 2020
Theftfro_7,N,3,0	Number of thefts from motor vehicle crime incidents in 2021

Formatted: Font: 12 pt, Font color: Auto



OpenStreetMap Tags Mapping

This section provides a brief summary of how the tags used in OpenStreetMap were mapped into the ontology used in the City Digital Twin.

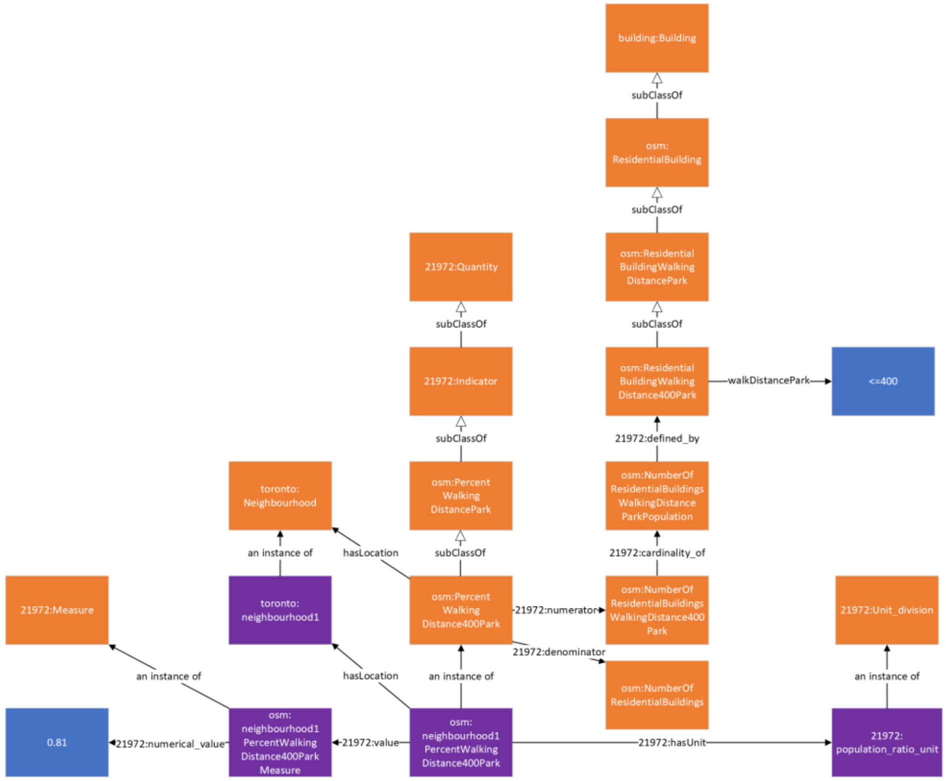
OpenStreetMap Tag	OpenStreetMap Description	CDT Equivalent
name	The primary name: in general, the most prominent signposted name or the most common name in the local language(s).	genprop:hasName
id	OSM ID is a numerical identifier that is assigned to every element in the OpenStreetMap (OSM) database.	cdt:osmID
addr:street	The street name that this address is (and any others in this location are) grouped by. This street name should match that of a nearby road, track or path.	contact:hasStreet (street name) contact:hasStreetType (street type) contact:hasStreetDirection(street direction
addr:housenumber	The house (or building) number that is included in the address. The number may contain non-digits and if recording multiple house numbers separate them by "," (e.g. "12b,12c").	contact:hasStreetNumber
addr:postcode	The postal code / zip code that is included in the address.	contact:hasPostalCode
operator	Company, corporation, person or any other entity who is directly in charge of the current operation of a map object	cdt:operator
operator:type	Defines the type of operator, e.g. "public", "private", "government"	cdt:operatorType gcie:hasOwnership (for schools)
website contact:website	Specifies the link to the official website for a feature.	cdt:website
surface	The surface key is used to provide additional information about the physical surface of roads/footpaths and some other features, particularly regarding material composition and/or structure.	cdt:surface
opening_hours	Describes when something is open or closed in a standard format	cdt:openingHours
phone	A telephone number associated with the object. Use +CC XXX	contact:hasTelephone

	XXX XXX format, where CC is a country code.	
email contact:email	An email address associated with the object	cdt:email
lit	The key lit=* indicates the presence of lighting and can be used on nodes, ways, areas, or relations.	cdt:lit
wheelchair	Indicate if a special place can be used with wheelchairs	cdt:wheelchairAccess
school:language	The main language of teaching and the administration of a school	cdt:languageOfInstruction
isced:level	Indicates a level of education on a numeric scale inspired by the International Standard Classification of Education (ISCED)	code:hasCode Note: the ISCED level of an educational facility is represented as an instance of the code:Code class with a name and description linked using the genprop:hasName and genprop:hasDescription properties respectively
religion	Defines the specific religion of a facility	cdt:religion
emergency	Used to indicate whether a hospital is equipped to deal with emergencies	cdt:emergencyServices
geometry	"Geometry" in this case means the shape and position of a node or way	geo:asWKT Note: the geometry is converted to WKT format using Shapely's to_wkt function if the geometry is not initially in WKT format
dispensing	Whether a pharmacy dispenses prescription drugs or not	cdt:dispensing
cuisine	Describes the type of food served at a place	cdt:cuisine
delivery	Indicates whether a restaurant or shop offers delivery of meals or goods	cdt:delivery
takeaway	Indicates whether the restaurant offers meals for pick-up that can be consumed elsewhere	cdt:takeaway
smoking	Used to describe whether smoking is allowed or not in a facility or on a property	cdt:smoking

outdoor_seating	To indicate if a feature offers outdoor seating	cdt:outdoorSeating
capacity	Describes the capacity a facility is suitable for	cdt:capacity
drive_through	To indicate whether an amenity offers a drive-through service	cdt:driveThrough

Complete Community Walking Distance Indicator

This is an ontological representation of a complete community walking distance indicator that shows the percentage of residential buildings in a geographic area that are within a specified distance of a complete community amenity. The diagram below shows the representation of a walking distance indicator for parks where individual instances of an indicator represent the walking distance indicator for a geographic area specified by the “hasLocation” property. This indicator is linked to a measure instance using the “value” property and the measure instance is linked to the value of the indicator using the “numerical_value” property. In the diagram below, we can see that the numerical value for the “neighborhood1PercentWalkingDistance400Park” indicator is 0.81 which means that 81% of the residential buildings in neighborhood1 are within a 400m distance of a park. This indicator is calculated by dividing the number of residential buildings that are within a 400m distance of a park by the total number of residential buildings in the area and this relationship is represented using the “numerator” and “denominator” properties, respectively. Similar indicators are also available for other Complete Community amenities.


building:Building
subClassOf
osm:ResidentialBuilding
subClassOf
osm:ResidentialBuildingWalkingDistancePark
subClassOf
21972:Quantity
subClassOf
21972:Indicator
subClassOf
osm:ResidentialBuildingWalkingDistance400Park
walkDistancePark
<=400
21972:defined_by
osm:NumberOfResidentialBuildingsWalkingDistanceParkPopulation
toronto:Neighbourhood
osm:PercentWalkingDistancePark
an instance of
hasLocation
subClassOf
21972:cardinality_of
21972:Measure
toronto:neighbourhood1
osm:PercentWalkingDistance400Park
21972:numerator
osm:NumberOfResidentialBuildingsWalkingDistance400Park
21972:Unit_division
21972:denominator
an instance of
hasLocation
an instance of
osm:NumberOfResidentialBuildings
an instance of
0.81
21972:numerical_value
osm:neighbourhood1PercentWalkingDistance400ParkMeasure
21972:value
osm:neighbourhood1PercentWalkingDistance400Park
21972:hasUnit
21972:population_ratio_unit

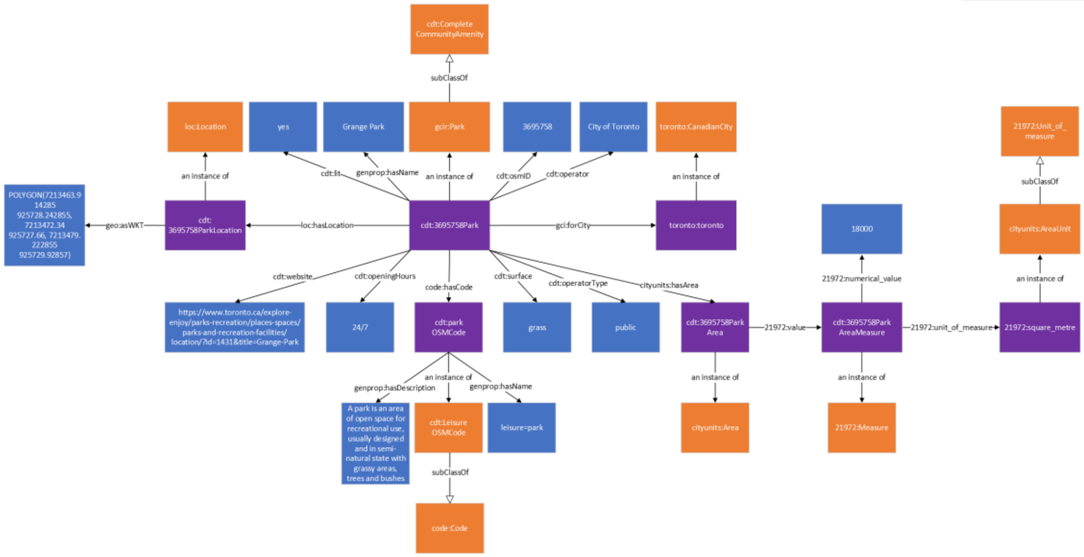
Parks

This is an ontological representation of park data. Parks are represented as instances of the Park class from the Recreation Ontology for Global City Indicators. Parks can have a name which is linked via the hasName property, an operator which is linked via the operator property, an operator type (e.g. public, private) which is linked via the operatorType property, a value describing the physical surface of the park which is linked via the surface property, opening hours which is linked via the openingHours property, a website which is linked via the website property, an indicator for whether the park is lit or not via the lit property, a hasArea property that links the park to its area instance and area value using the numerical_value property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively, and a code instance that can be used to link the park to a specific set list of values (e.g., classification systems).

Additionally, for OpenStreetMap park data, the instance can also have an osmID property that links the park to its OSM ID (a unique identifier that is used in OpenStreetMap). It can also be linked to

OpenStreetMap's "leisure=park" tag that is used to identify parks in OpenStreetMap using the hasCode property and Code class.

The value of a park's surface area is calculated using the Pyproj (a cartographic projections and coordinate transformations library for Python) and its polygon coordinates. The Shapely Python package is also used to convert the coordinates into a Shapely shape for use with Pyproj functions.

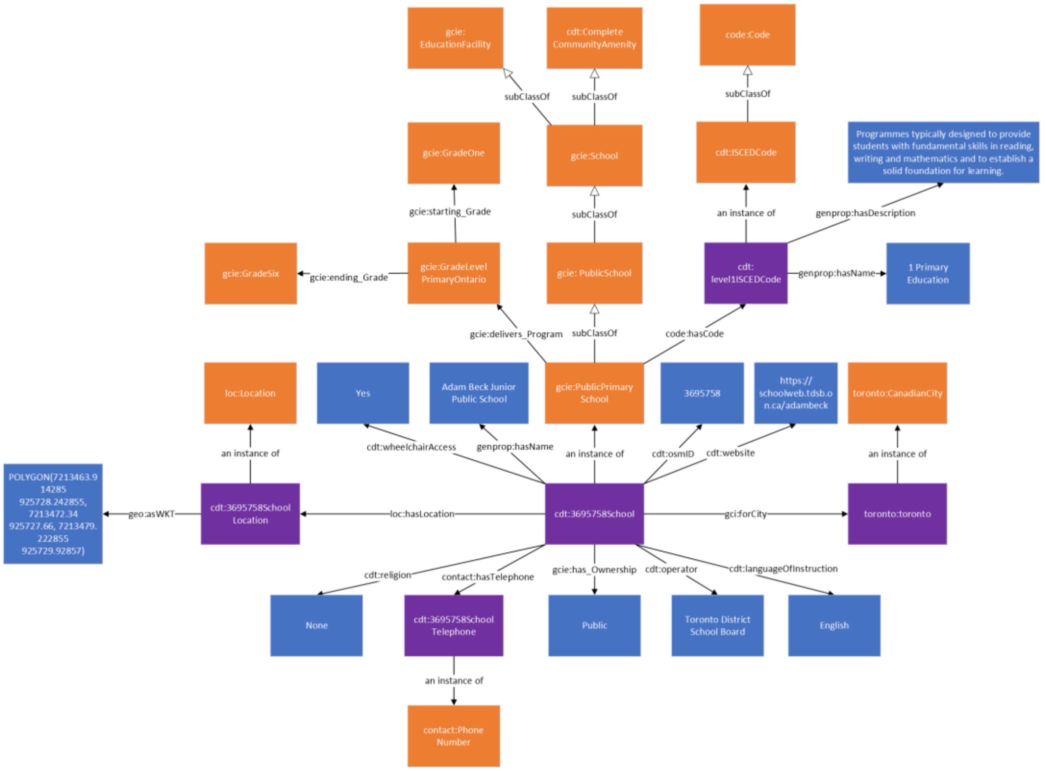


Schools

This is an ontological representation of school data. Schools are represented as instances of the School class from the GCI Education Ontology (GCIE) which is also a subclass of the EducationFacility class from GCIE and the CompleteCommunityAmenity class. Schools can have a name which is linked via the hasName property, a website which is linked via the website property, a religion which is linked via the religion property, a phone number which is linked via the hasPhoneNumber property, an ownership type (e.g. public, private) which is linked via the has_Ownership property, an operator which is linked via the operator property, a language of instruction (i.e. the language used to teach in a school) which is linked via the languageOfInstruction property, its wheelchair accessibility which is linked via the wheelchairAccess property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. School classes are linked to an ISCED code instance which describes the level of education that type of school provides. For example, in the diagram below, the PublicPrimarySchool class has an ISCED code of level 1 which represents Primary Education (i.e., programmes typically designed to provide students with fundamental skills in reading, writing and mathematics and to establish a solid foundation for learning.)

Additionally, for OpenStreetMap school data, the instance can also have an osmID property that links the school to its OSM ID (a unique identifier that is used in OpenStreetMap).

Schools with different operator types (e.g. public, private) and level of education are placed in the appropriate school subclass from the GCIE ontology (e.g. a primary school that is publicly operated would be an instance of the gcie:PublicPrimarySchool class).



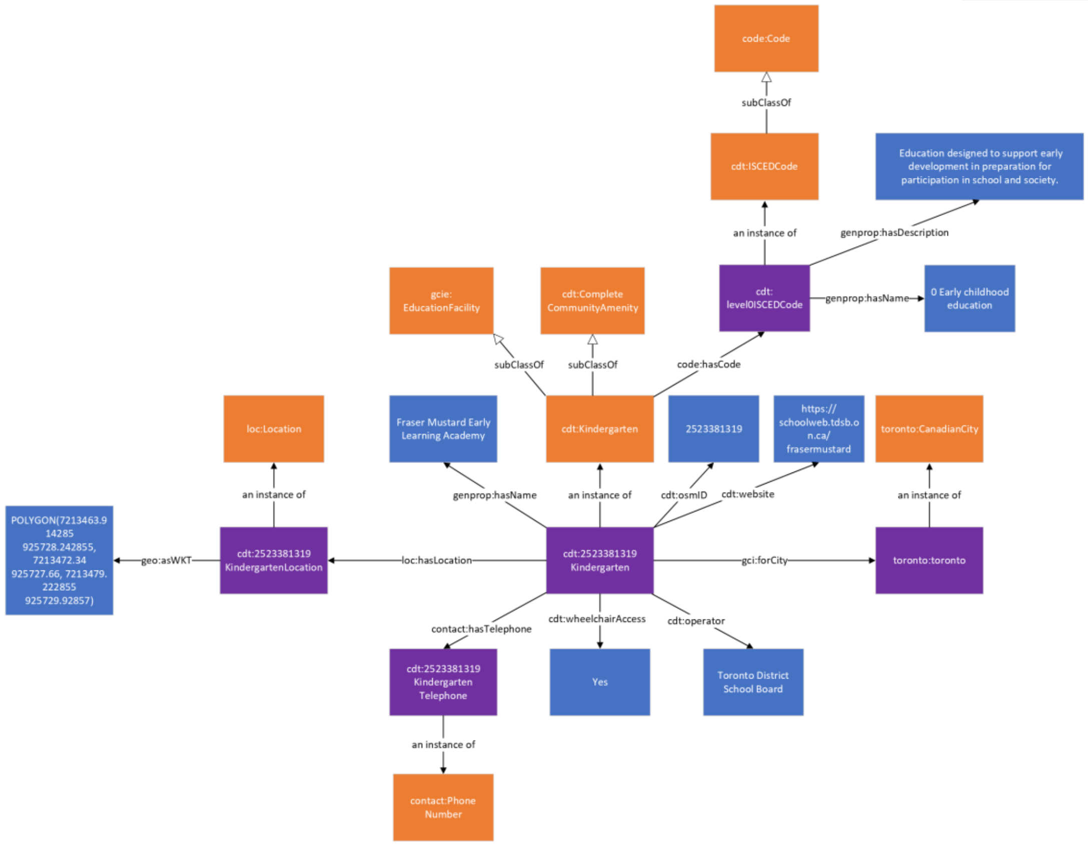
Kindergartens

This is an ontological representation of kindergarten data. Kindergartens are represented as instances of the Kindergarten class which is a subclass of the EducationFacility class from the GCI Education Ontology and the CompleteCommunityAmenity class. Kindergartens can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. The Kindergarten class is linked to an ISCED code instance which describes the level of education that a kindergarten provides using the hasCode property.

Kindergartens provide ISCED level 0 education (i.e., education designed to support early development in preparation for participation in school and society).

Additionally, for OpenStreetMap kindergarten data, the instance can also have an osmID property that links the kindergarten to its OSM ID (a unique identifier that is used in OpenStreetMap).

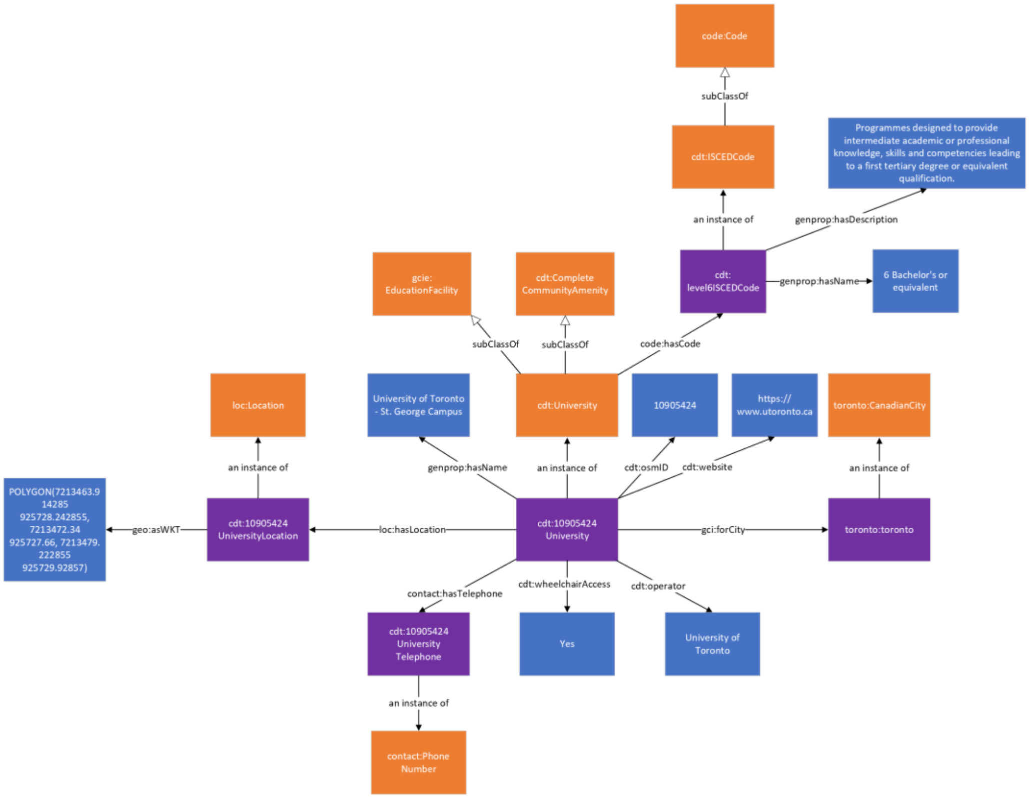
Formatted: Font: 12 pt, Font color: Auto


code:Code
subClassOf
cdt:ISCEDCode
Education designed to support early development in preparation for participation in school and society.
an instance of
genprop:hasDescription
gcie: EducationFacility
cdt:Complete CommunityAmenity
cdt: level0ISCEDCode
genprop:hasName
0 Early childhood education
subClassOf
subClassOf
code:hasCode
loc:Location
Fraser Mustard Early Learning Academy
cdt:Kindergarten
2523381319
https://schoolweb.tdsb.on.ca/frasermustard
toronto:CanadianCity
an instance of
genprop:hasName
an instance of
cdt:osmID
cdt:website
an instance of
POLYGON(7213463.9 14285 925728.242855, 7213472.34 925727.66, 7213479.222855 925729.92857)
geo:asWKT
cdt:2523381319 KindergartenLocation
loc:hasLocation
cdt:2523381319 Kindergarten
gci:forCity
toronto:toronto
contact:hasTelephone
cdt:wheelchairAccess
cdt:operator
cdt:2523381319 Kindergarten Telephone
Yes
Toronto District School Board
an instance of
contact:Phone Number

Universities

This is an ontological representation of university data. Universities are represented as instances of the University class which is a subclass of the EducationFacility class from the GCI Education Ontology and the CompleteCommunityAmenity class. Universities can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. The University class is linked to an ISCED code instance which describes the level of education that a university provides using the hasCode property. Universities provide ISCED level 6 education (i.e., programmes designed to provide intermediate academic or professional knowledge, skills and competencies leading to a first tertiary degree or equivalent qualification) and higher.

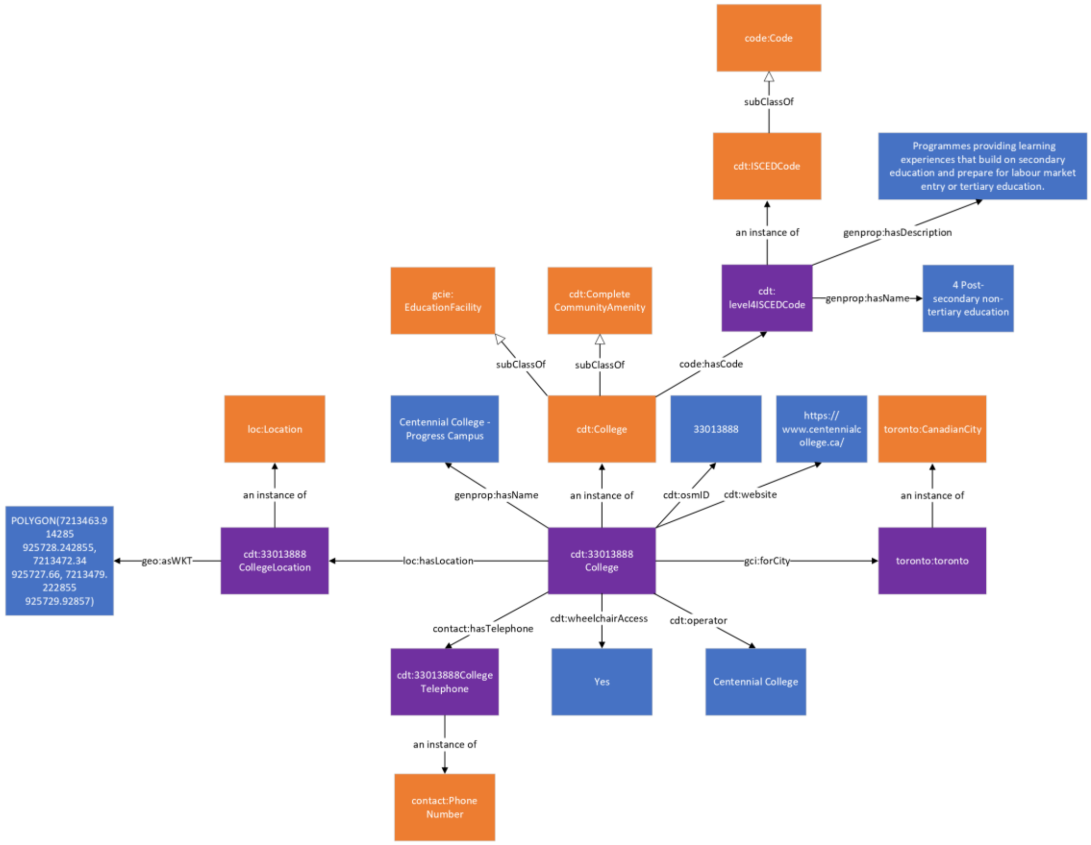
Additionally, for OpenStreetMap university data, the instance can also have an osmID property that links the university to its OSM ID (a unique identifier that is used in OpenStreetMap).



Colleges

This is an ontological representation of college data. Colleges are represented as instances of the College class which is a subclass of the EducationFacility class from the GCI Education Ontology and the CompleteCommunityAmenity class. Colleges can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. The College class is linked to an ISCED code instance which describes the level of education that a college provides using the hasCode property. Colleges provide ISCED level 4 education (i.e., programmes providing learning experiences that build on secondary education and prepare for labour market entry or tertiary education.) and higher.

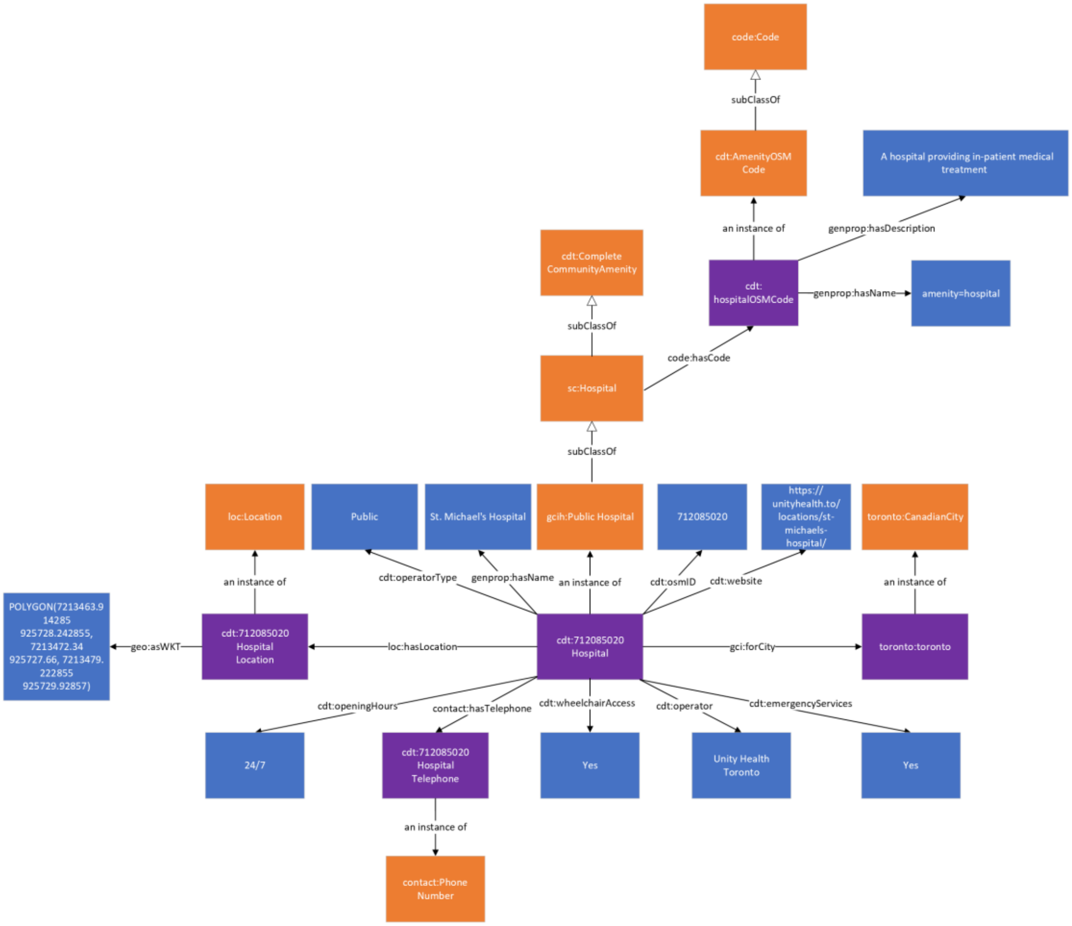
Additionally, for OpenStreetMap college data, the instance can also have an osmID property that links the college to its OSM ID (a unique identifier that is used in OpenStreetMap).



Hospitals

This is an ontological representation of hospital data. Hospitals are represented as instances of the Hospital class from Schema.org which has the subclasses PublicHospital and PrivateHospital from the GCI Healthcare Ontology to indicate whether they are publicly or privately operated and the superclass CompleteCommunityAmenity class. Hospitals can have a name which is linked via the hasName property, whether it provides emergency services via the emergencyServices property, a website which is linked via the website property, an operator which is linked via the operator property, the type of the operator which is linked via the operatorType property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. The Hospital class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

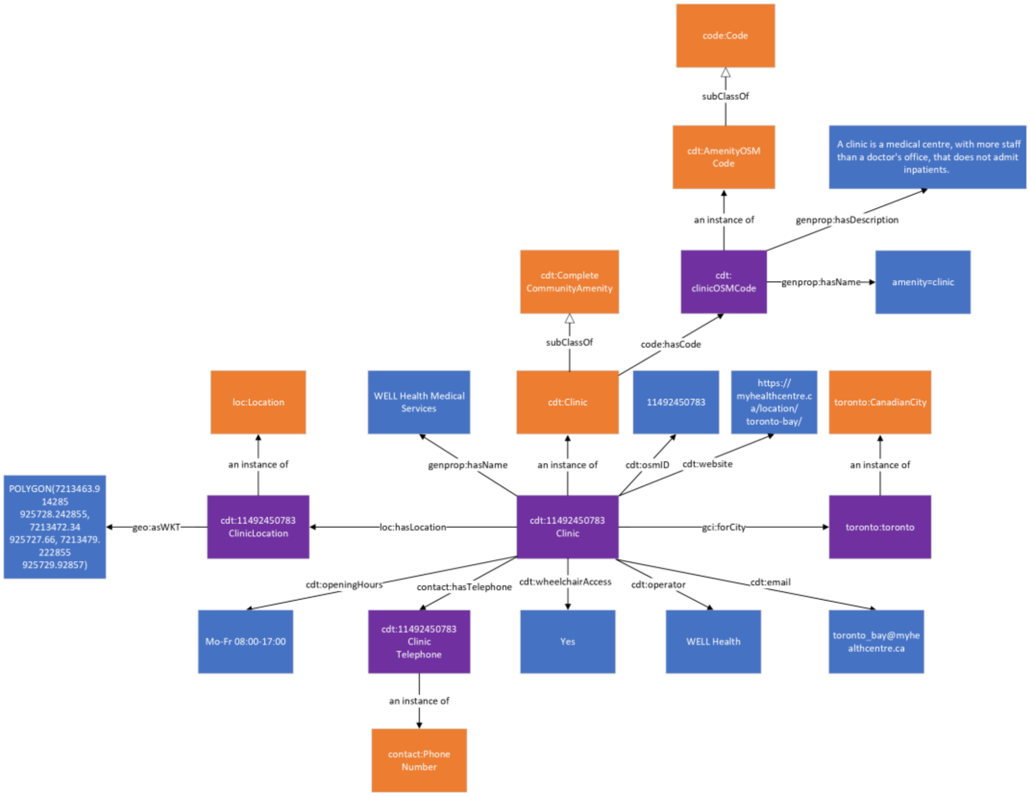
Additionally, for OpenStreetMap hospital data, the instance can also have an osmID property that links the hospital to its OSM ID (a unique identifier that is used in OpenStreetMap).



Clinics

This is an ontological representation of clinic (i.e. medical centre) data. Clinics are represented as instances of the Clinic class which is a subclass of the CompleteCommunityAmenity class. Clinics can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a contact email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property respectively. The Clinic class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

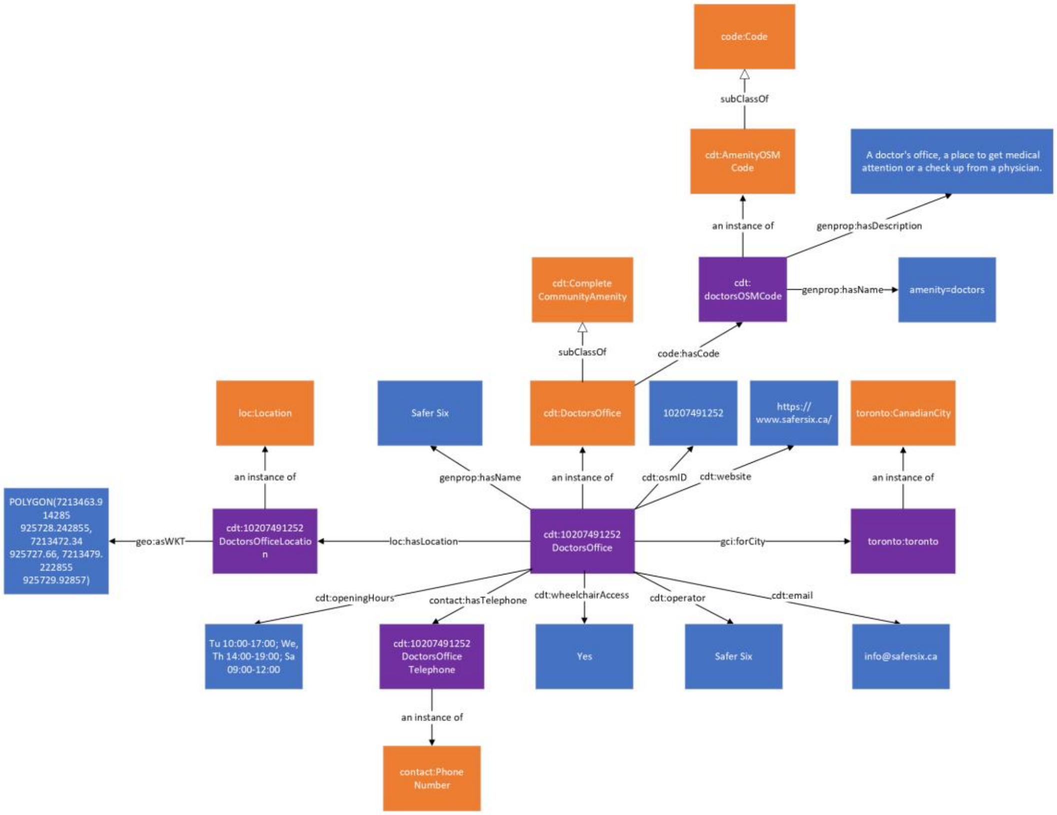
Additionally, for OpenStreetMap clinic data, the instance can also have an osmID property that links the clinic to its OSM ID (a unique identifier that is used in OpenStreetMap).



Doctor’s Office

This is an ontological representation of doctor’s office data. Doctor’s offices are represented as instances of the DoctorsOffice class which is a subclass of the CompleteCommunityAmenity class. Doctor’s offices can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a contact email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The DoctorsOffice class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

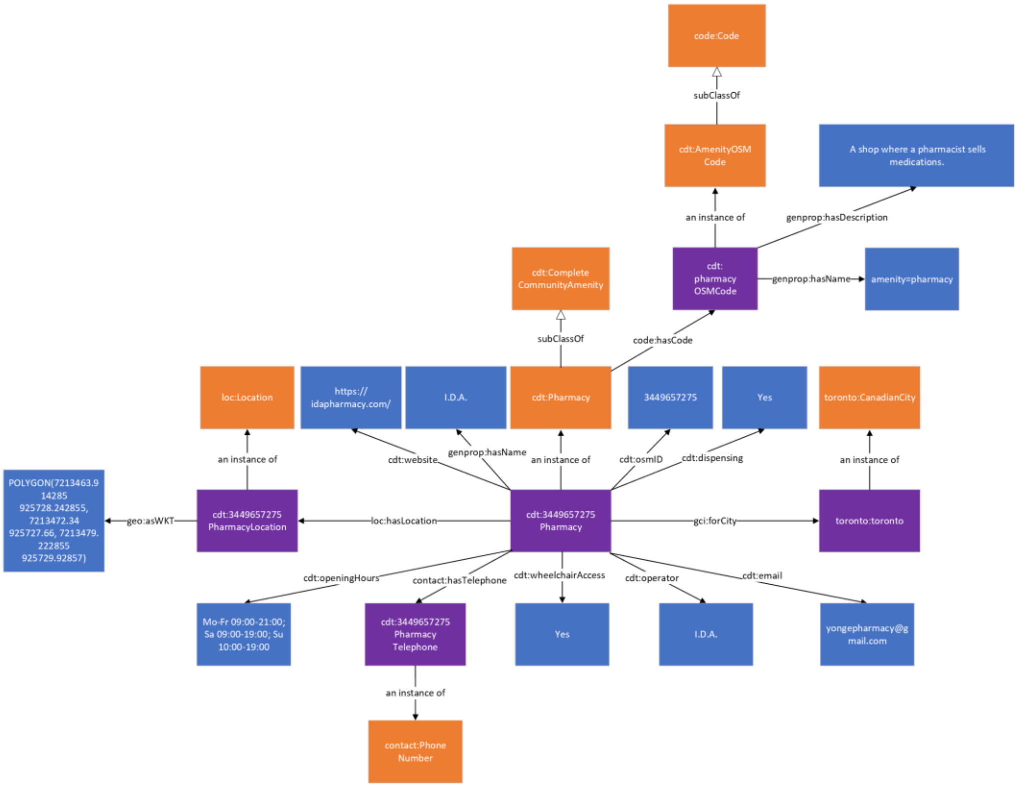
Additionally, for OpenStreetMap doctor’s office data, the instance can also have an osmID property that links the doctor’s office to its OSM ID (a unique identifier that is used in OpenStreetMap).



Pharmacy

This is an ontological representation of pharmacy data. Pharmacies are represented as instances of the Pharmacy class which is a subclass of the CompleteCommunityAmenity class. Pharmacies can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a contact email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, whether they provide drug dispensing services via the dispensing property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The Pharmacy class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

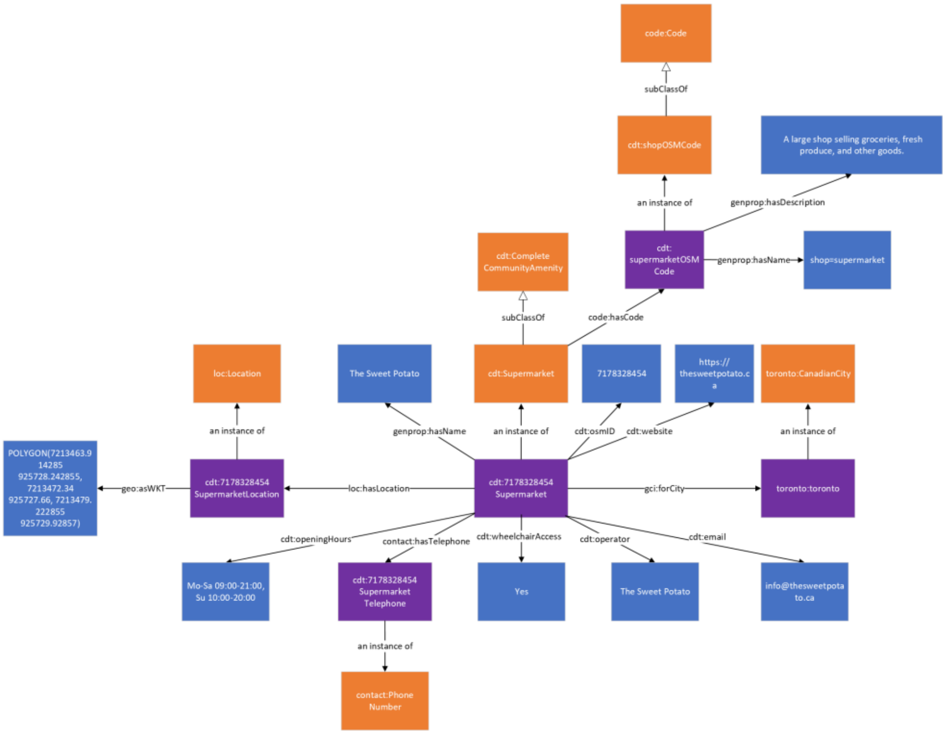
Additionally, for OpenStreetMap pharmacy data, the instance can also have an osmID property that links the pharmacy to its OSM ID (a unique identifier that is used in OpenStreetMap).



Supermarket

This is an ontological representation of supermarket data. Supermarkets are represented as instances of the Supermarket class which is a subclass of the CompleteCommunityAmenity class. Supermarkets can have a name which is linked via the hasName property, a website which is linked via the website property, an operator which is linked via the operator property, a contact email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The Supermarket class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

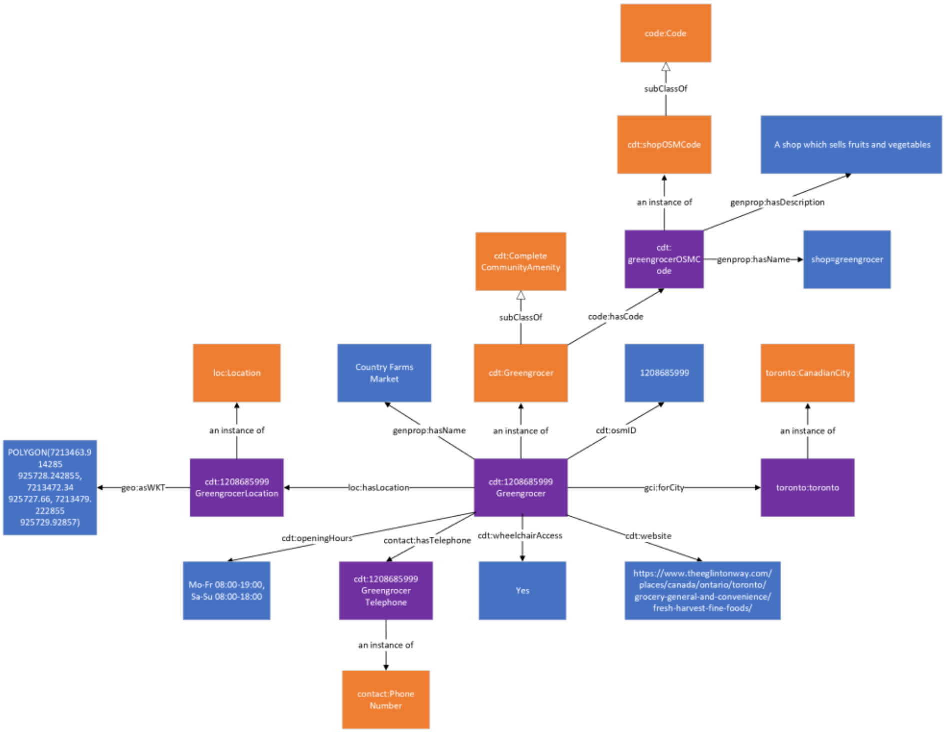
Additionally, for OpenStreetMap supermarket data, the instance can also have an osmID property that links the supermarket to its OSM ID (a unique identifier that is used in OpenStreetMap).



Greengrocer

This is an ontological representation of greengrocer data. Greengrocers are represented as instances of the Greengrocer class which is a subclass of the CompleteCommunityAmenity class. Greengrocers can have a name which is linked via the hasName property, a website which is linked via the website property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The Greengrocer class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

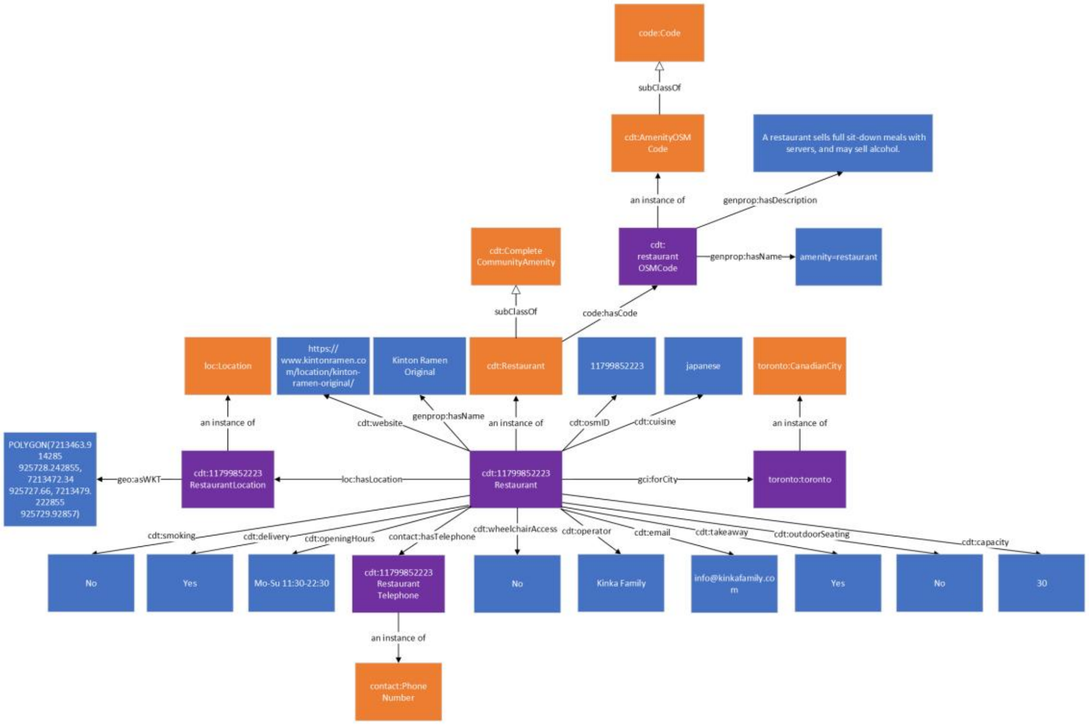
Additionally, for OpenStreetMap greengrocer data, the instance can also have an osmID property that links the greengrocer to its OSM ID (a unique identifier that is used in OpenStreetMap).



Restaurant

This is an ontological representation of restaurant data. Restaurants are represented as instances of the Restaurant class which is a subclass of the CompleteCommunityAmenity class. Restaurants can have a name which is linked via the hasName property, a website which is linked via the website property, an email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, the type of cuisine that it serves via the cuisine property, whether delivery is available via the delivery property, whether takeaway is offered via the takeaway property, whether smoking is allowed via the smoking property, whether outdoor seating is available via the outdoorSeating property, its capacity (i.e. number of people it can accomodate) via the capacity property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The Restaurant class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

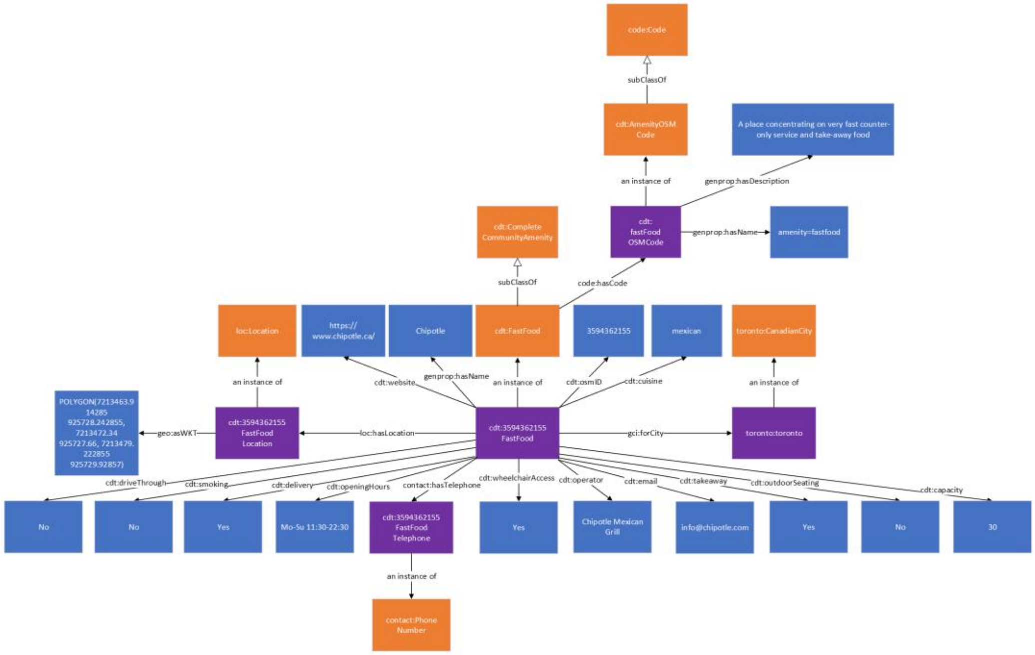
Additionally, for OpenStreetMap restaurant data, the instance can also have an osmID property that links the restaurant to its OSM ID (a unique identifier that is used in OpenStreetMap).



FastFood

This is an ontological representation of fast food vendor data. Fast food vendorsare represented as instances of the FastFood class which is a subclass of the CompleteCommunityAmenity class. Fast food vendors can have a name which is linked via the hasName property, a website which is linked via the website property, an email which is linked via the email property, a phone number which is linked via the hasTelephone property, its wheelchair accessibility which is linked via the wheelchairAccess property, its opening hours via the openingHours property, the type of cuisine that it serves via the cuisine property, whether delivery is available via the delivery property, whether takeaway is offered via the takeaway property, whether smoking is allowed via the smoking property, whether outdoor seating is available via the outdoorSeating property, its capacity (i.e. number of people it can accomodate) via the capacity property, whether it offers drive-through services via the driveThrough property, a Location instance and a set of geospatial coordinates linked using the hasLocation and asWKT property, respectively. The FastFood class can also use the hasCode property in order to link it to a specific set list of values (e.g., classification systems).

Additionally, for OpenStreetMap fast food vendor data, the instance can also have an osmID property that links the fast food vendorto its OSM ID (a unique identifier that is used in OpenStreetMap).



Representation of Road Data

The Ontario Road Network (ORN) dataset represents the provincial road infrastructure and is segmented at real-world intersections or junctions. The primary dataset consists of a shapefile (ORN_ROAD_NET_ELEMENT.shp) that describes road geometries and includes fields such as OGF_ID (a unique identifier for each road element), FROM_JCT, and TO_JCT, which indicate the junctions that bound each segment, and more. These junction references establish a topological connection between road elements. Supplementary CSV files (e.g., ORN_ROAD_CLASS.csv, ORN_SPEED_LIMIT.csv) enrich the dataset with additional attributes, including road names, speed limits, lane counts, and more. Example of Data in shp file for a road element:

FROM_JCT:1500091661	TO_JCT:1500045335
LENGTH:254.258	ACCURACY:3.0
NID:9bed6561cdbd438590abec7bf592d722	DIRECTION:Both
EXIT_NUM:18	ELEM_TYPE:ROAD ELEMENT
TOLL_ROAD:Yes	ACQTECH:VECTOR DATA
CREDATE:20020401000000	REVDATE:None
GEO_UPDATE_DT:None	EFF_DATE:20090123155815

In the example, this is a road element identified by OGF_ID = 1509876543, runs from junction 1501234567 to junction 1507654321, with a total length of 254.258 meters, has a positional accuracy of 3.0 meters, unique national identifier of 9bed6561cdbd438590abec7bf592d722, has traffic flowing in both directions, an exit number of 18, it's a type of road element, it is a toll road, it was created using vector data, had a creation date of April 1st 2002, a revision and geometry update date that is unknown, and effective date (when record was created) of January 23rd 2009 (15 hours, 58 minutes, and 15 seconds). More information about these attributes below.

Each road network element is bounded by a FROM_JCT and a TO_JCT, making it semantically aligned with the RoadLink class in the ISO/IEC 5087-3 ontology. Each RoadLink begins and ends at a TransportNode, which corresponds to a Junction in the ontology

Thus, in our City Digital Twin:

- **ORN road net elements** → transnet:RoadLink (subclass of TravelledWayLink)
- **ORN junctions** → transnet:Junction (subclass of TransportNode)
- A group of RoadLinks sharing the same street name → transnet:Road

The data was also filtered around only Toronto roads using the Toronto Bound Filter:

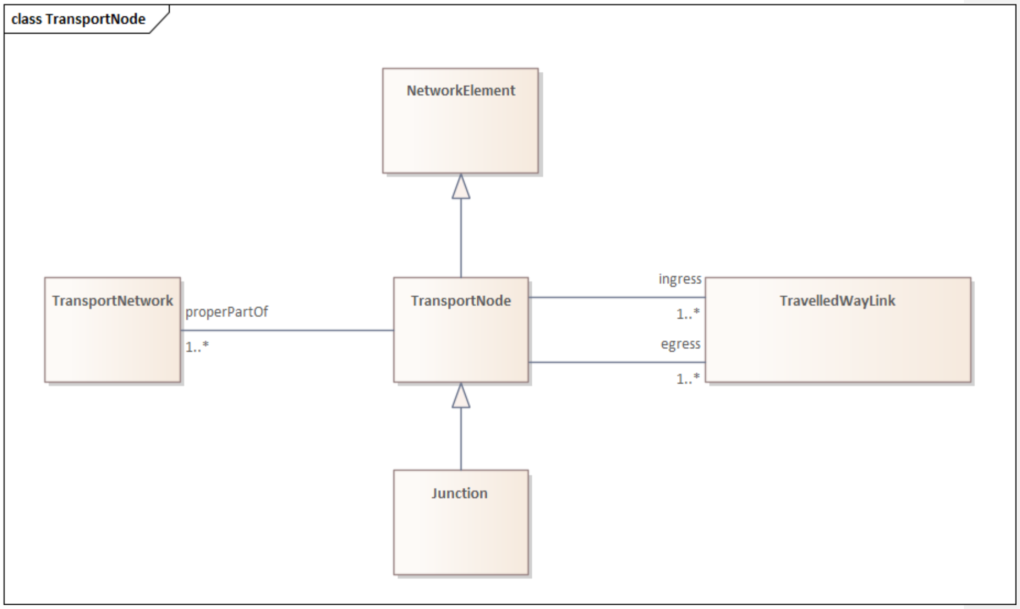
lat_min: 43.5810, lat_max: 43.8555, lon_min: -79.6393, lon_max: -79.1152

The following is a list of namespace prefixes:

- geo: http://www.opengis.net/ont/geosparql#
- transnet: https://standards.iso.org/isoiec/5087/3/ed1/en/ontology/TransportaionNetwork/
- transinfras: https://standards.iso.org/isoiec/5087/2/ed1/en/ontology/TransportationInfrastructure/
- loc: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/SpatialLoc/>
- partwhole: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/Mereology/
- cityunits: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/CityUnits/
- cdt: http://ontology.eil.utoronto.ca/CDT#
- rdfs: http://www.w3.org/2000/01/rdf-schema#
- i72: http://ontology.eil.utoronto.ca/5087/2/iso21972/
- genprop: https://standards.iso.org/iso-iec/5087/-1/ed-1/en/ontology/GenericProperties/
- rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
- contact: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Contact/
- code: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Code/
- infras: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Infrastructure/
- road: https://standards.iso.org/iso-iec/5087/-3/ed-1/en/ontology/RoadNetwork
- org_city: https://standards.iso.org/iso-iec/5087/-2/ed-1/en/ontology/Organization/
- xsd: http://www.w3.org/2001/XMLSchema#

Junction

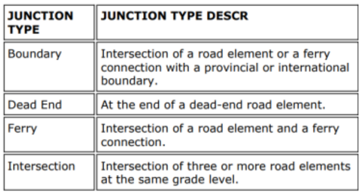
Junctions are an instance of a subclass/specialization of transnet:Junction and are a subclass of TransportNode. These entities connect travellers from one TravelledWayLink to another, in particular serving a connection between one or more RoadLinks and is uniquely identified using ORN-provided IDs. Geospatial coordinates are linked using geo:Geometry pointing to a geo:asWKT. Each Junction participates in one or more ingress and egress relationships with RoadLinks, ensuring accurate topological representation of the network.



Although the shp file does not have any information around the junctions beside which junctions bound the element, all the necessary information regarding all junctions in the dataset are in a special ORN_JUNCTIONS.csv file. And ID of the junction are specified in the csv along with its location, type, etc. (more on this below)

Note: All the CSV files had a data tag ORN_ROAD_NET_ELEMENT_ID, an Integer representing a system-generated identifier unique at the application level.

All the following properties are for associated with instances of the the cdt:Junction class, which is a subclass of the Junction class in the TransportationNetwork ontology.

<table>
<tr><th colspan="4">Data Provided by ORN_JUNCTION.csv</th></tr>
<tr><th>ORN Data Tag</th><th>ORN Data Description</th><th>Maps to CDT Property</th><th>Value</th></tr>
<tr><td>JUNCTION_ID</td><td>System-generated identifier, unique at the application level.</td><td>genprop:hasIdentifier</td><td>xsd:integer</td></tr>
<tr><td>LATITUDE_DECIMAL_DEGREES</td><td>The latitude in decimal degrees.</td><td rowspan="2">For each Junction individual, a new geo:Geometry individual is created and linked via
loc:hasLocation.
This geometry ~~uses~~ is defined with the geo:asWKT property using the LATITUDE_DECIMAL_DEGREES and LONGITUDE_DECIMAL_DEGREES with the POINT (lon lat) format.</td><td rowspan="2">geo:asWKT</td></tr>
<tr><td>LONGITUDE_DECIMAL_DEGREES</td><td>The longitude in negative decimal degrees.</td></tr>
<tr><td>JUNCTION_TYPE</td><td>The classification of a junction is based on the valency of the junction. The number of road elements or ferry connections joining at a junction is termed the valency of a junction.

<table>
<tr><th>JUNCTION TYPE</th><th>JUNCTION TYPE DESCR</th></tr>
<tr><td>Boundary</td><td>Intersection of a road element or a ferry connection with a provincial or international boundary.</td></tr>
<tr><td>Dead End</td><td>At the end of a dead-end road element.</td></tr>
<tr><td>Ferry</td><td>Intersection of a road element and a ferry connection.</td></tr>
<tr><td>Intersection</td><td>Intersection of three or more road elements at the same grade level.</td></tr>
</table></td><td>cdt:hasJunctionType</td><td>cdt:
JunctionType</td></tr>
<tr><td>EXIT_NUMBER</td><td>The number of an exit on or off a freeway, expressway or highway, assigned by an administrating body and</td><td>cdt:exitNumber</td><td>xsd:strin</td></tr>
</table>

Commented [MK1]: I think you can remove this column entirely

Commented [MK2]: These restrictions are required for the definition of the extension, but they don't need to be in the mapping; implicitly, you're defining the value to be whatever is in the row of data (which could be one value, multiple values, or nothing). If your mapping addresses cases of single, multiple, or null values then you can describe that in the procedure.

	is represented by a valid number or character		
NATIONAL_UUID	A unique national identifier assigned to a road net element, junction and selected event data such as Toll Point, Blocked Passage and Structure which are required to support the National Road Network (NRN).	cdt:nationUUID	xsd:string
EFFECTIVE_DATETIME	Date/time the record was created or last modified in the source database.	cdt:effectiveDate	xsd:Date

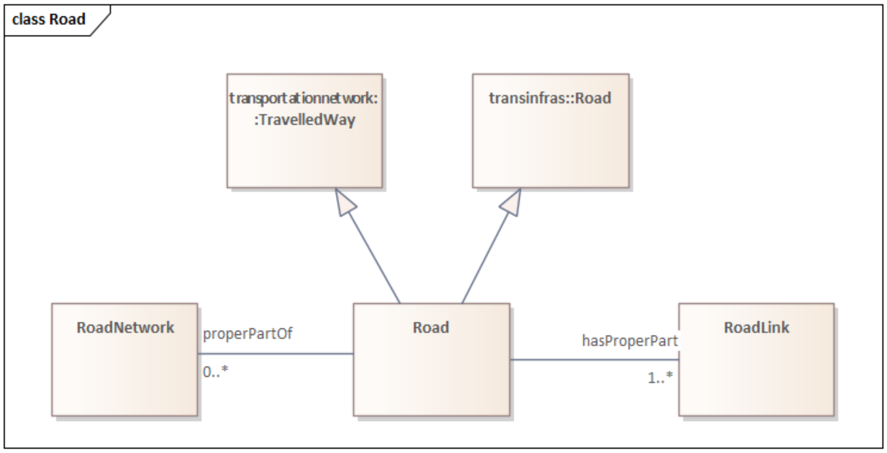
ORN_Junction.csv: A unique national identifier assigned to a road net element, junction and selected event data such as Toll Point, Blocked Passage and Structure which are required to support the National Road Network (NRN).

Data Provided by ORN_ROAD_NET_ELEMENT.shp:			
ORN Data Tag	ORN Data Description	Property	Value
TO_JCT	The end junction for a road element or ferry connection.	transnet:ingress	RoadLink
FROM_JCT	The beginning junction for a road element or ferry connection.	transnet:egress	RoadLink

Note: The TO_JCT and FROM_JCT; are both junction IDs that are within the ORN_JUNCTION.csv files. So, using the ID, we can find the URI for the junction that has already been created. When the junction entities are found, then we use the properties above on the corresponding RoadLinks.

Road

A Road is modeled as a transinfras:Road, a subclass of TravelledWay, and is defined as a continuous sequence of RoadLinks that share a common entity (e.g., Highway 401, Dundas Street). A single RoadLink may be part of multiple Roads to accommodate overlapping entities. Each Road is linked to a unique designator (road name or number), and its extent is defined by the collective geometry of its constituent RoadLinks.



Data Provided by ORN_OFFICIAL_STREET_NAME.csv:			
ORN Data Tag	ORN Data Description	Property	Value
FULL_STREET_NAME	This attribute is derived from the individual street name components where present, namely directional prefix, street type prefix, street name body, street type suffix and directional suffix and is stored in upper case text.	genprop:hasName	xsd:string

ORN_OFFICIAL_STREET_NAME.csv: An event identifying an official street name and may be associated with a bilingual name.

The ORN data is grouped using the road name to identify the collection of RoadLinks that form a single Road. A unique URI is generated for each Road entity, and all corresponding RoadLinks are created and linked to that Road using the partwhole:hasProperPart property.

RoadLink

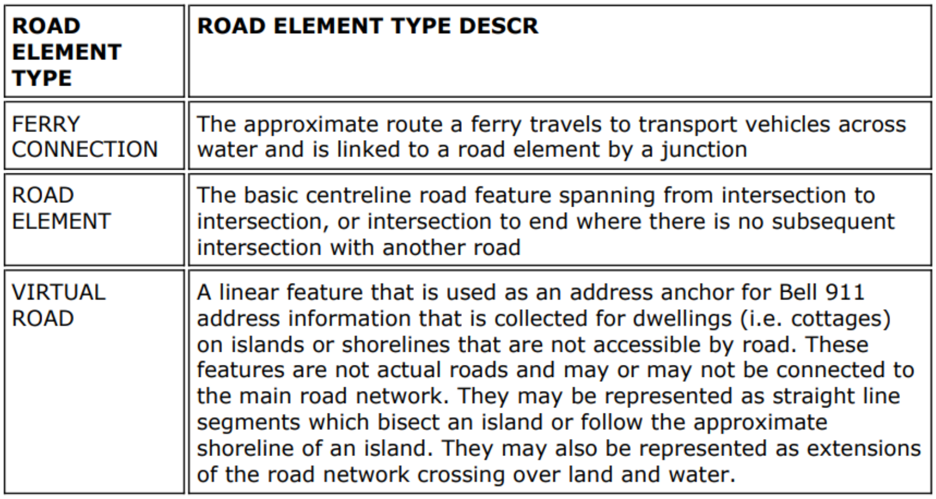
A RoadLink is the fundamental linear segment between two TransportNodes (to and from junctions) and is represented using the transinfras:RoadLink class. Roadlinks are grouped together to form a Road.

FROM_JCT and TO_JCT represent the identifiers for the start and end junctions, respectively.

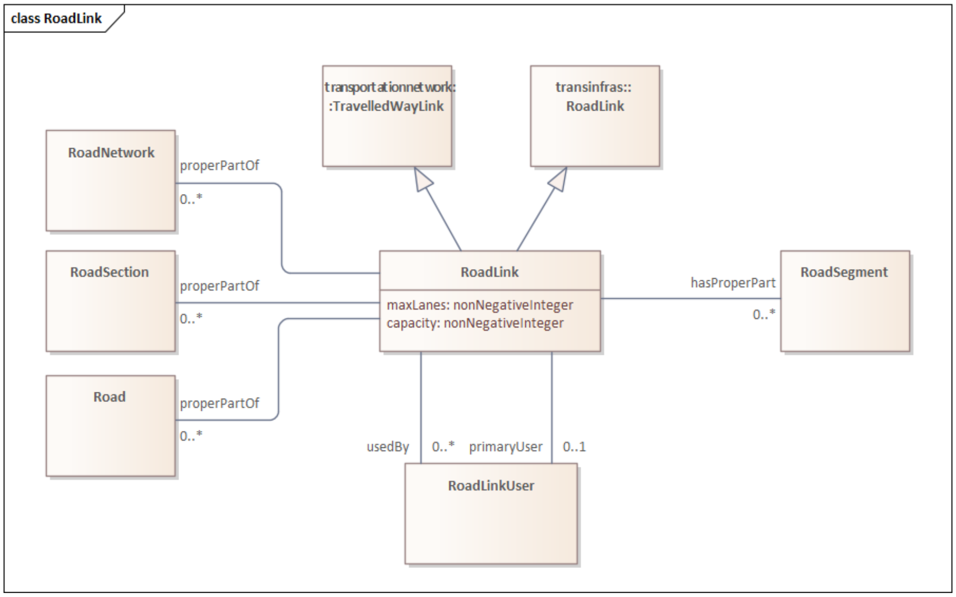
- In the RDF output, these are used to construct the transnet:forth and transnet:to properties for the corresponding RoadLink.

Each RoadLink instance is generated using the unique OGF_ID as an identifier (e.g. transnet:roadLink_12345). This ensures consistent referencing across other entities such as Roads and Junctions. The OGF_ID also serves as the subject for attaching additional metadata like speed limits, surface type, and geometry (WKT). All supplementary CSV data is joined to the shapefile using this key during pre-processing.

Each RoadLink can be one of three element types: Ferry Connection, Road Element, and Virtual Road. We use the ELEM_TYPE attribute in the shapefile to filter out all the roads that are of type "Virtual Road."



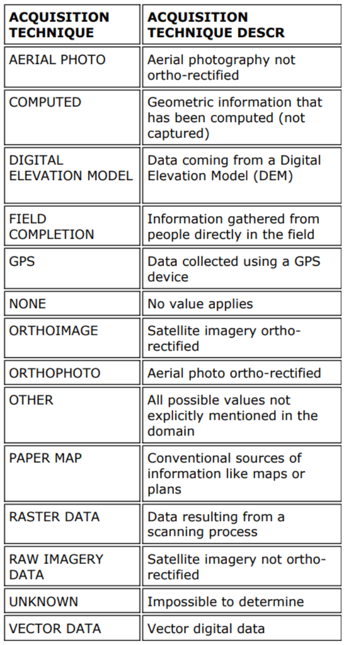
ROAD ELEMENT TYPE	**ROAD ELEMENT TYPE DESCR**
FERRY CONNECTION	The approximate route a ferry travels to transport vehicles across water and is linked to a road element by a junction
ROAD ELEMENT	The basic centreline road feature spanning from intersection to intersection, or intersection to end where there is no subsequent intersection with another road
VIRTUAL ROAD	A linear feature that is used as an address anchor for Bell 911 address information that is collected for dwellings (i.e. cottages) on islands or shorelines that are not accessible by road. These features are not actual roads and may or may not be connected to the main road network. They may be represented as straight line segments which bisect an island or follow the approximate shoreline of an island. They may also be represented as extensions of the road network crossing over land and water.



All the following properties are for associated with the cdt:RoadLink class, which is a subclass of the RoadLink class in the TransportationInfrastructure ontology.

ORN Data Tag	ORN Data Description	Property	Value
OGF_ID	A unique numeric provincial identifier assigned to each object.	genprop:hasIdentifier	xsd:integer
TO_JCT	The end junction for a road element or ferry connection.	transnet:to	Junction
FROM_JCT	The beginning junction for a road element or ferry connection.	transnet:from	Junction

NID	A unique national identifier assigned to a road net element, junction and selected event data such as Toll Point, Blocked Passage and Structure which are required to support the National Road Network (NRN).	cdt:nationUUID	xsd:string
DIRECTION	The direction(s) of vehicular or motor traffic flow. All road elements must have a direction of traffic flow assigned. Mapped using an enumeration class to capture semantic direction values (Positive, Negative, Both) in accordance with ISO 5087-3.	transnet:allowedDirections	transnet:LinkDirection
EXIT_NUM	The number of an exit on or off a freeway, expressway or highway, assigned by an administrating body and is represented by a valid number or character.	cdt:exitNum	xsd:string
TOLL_ROAD	Indicates if the road net element is a toll road.	cdt:tollRoadIND	xsd:boolean
ACQTECH	The type of data source or technique used to create or revise the road net element.	cdt:hasacquisitionTechnique	cdt:AcquisitionTechnique

<table>
<tr><td></td><td>
<table>
<tr><th>ACQUISITION TECHNIQUE</th><th>ACQUISITION TECHNIQUE DESCR</th></tr>
<tr><td>AERIAL PHOTO</td><td>Aerial photography not ortho-rectified</td></tr>
<tr><td>COMPUTED</td><td>Geometric information that has been computed (not captured)</td></tr>
<tr><td>DIGITAL ELEVATION MODEL</td><td>Data coming from a Digital Elevation Model (DEM)</td></tr>
<tr><td>FIELD COMPLETION</td><td>Information gathered from people directly in the field</td></tr>
<tr><td>GPS</td><td>Data collected using a GPS device</td></tr>
<tr><td>NONE</td><td>No value applies</td></tr>
<tr><td>ORTHOIMAGE</td><td>Satellite imagery ortho-rectified</td></tr>
<tr><td>ORTHOPHOTO</td><td>Aerial photo ortho-rectified</td></tr>
<tr><td>OTHER</td><td>All possible values not explicitly mentioned in the domain</td></tr>
<tr><td>PAPER MAP</td><td>Conventional sources of information like maps or plans</td></tr>
<tr><td>RASTER DATA</td><td>Data resulting from a scanning process</td></tr>
<tr><td>RAW IMAGERY DATA</td><td>Satellite imagery not ortho-rectified</td></tr>
<tr><td>UNKNOWN</td><td>Impossible to determine</td></tr>
<tr><td>VECTOR DATA</td><td>Vector digital data</td></tr>
</table></td><td></td><td></td></tr>
<tr><td>CREDATE</td><td>The date the road net element was originally created.</td><td>cdt:creationDate</td><td>xsd:date</td></tr>
<tr><td>REVDATE</td><td>The date the road net element was last revised or updated.</td><td>cdt:revisionDate</td><td>xsd:date</td></tr>
<tr><td>GEO_UPD_DT</td><td>Date/time the geometry was created or last modified in the source database.</td><td>cdt:geoUpdateDate</td><td>xsd:date</td></tr>
<tr><td>EFF_DATE</td><td>Date/time the record was created or last modified in the source database.</td><td>cdt:effectiveDate</td><td>xsd:date</td></tr>
<tr><td>LENGTH</td><td>The measured planimetric length of a road net element in meters.</td><td>cdt:length</td><td>cityunits:Length</td></tr>
</table>

ACCURACY	A statement that identifies the positional accuracy of the ORN road geometry, in metres.	cdt:roadAbsoluteAccuracy	cityunits:Length

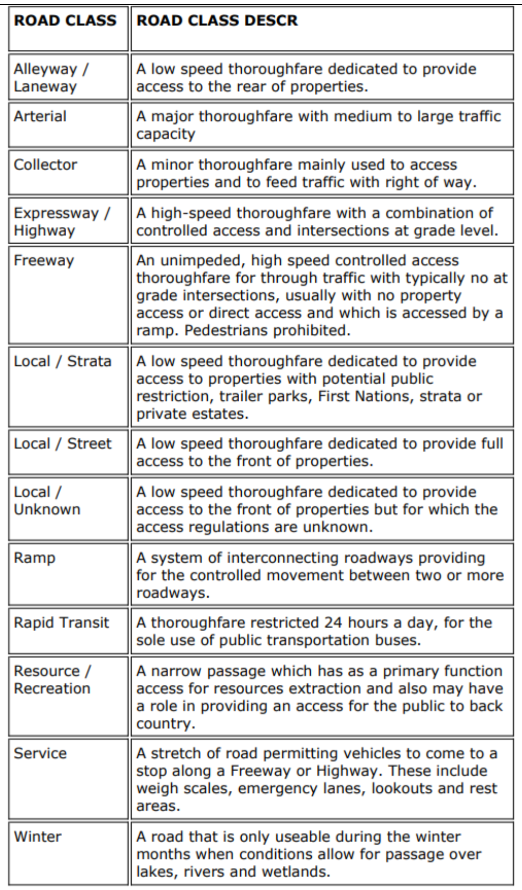
Data Provided by ORN csv Files:

Note: All the following CSV files had a data tag ORN_ROAD_NET_ELEMENT_ID, an Integer representing a system-generated identifier unique at the application level.

Data Provided by ORN_SPEED_LIMIT.csv:			
ORN Data Tag	ORN Data Description	Property	Value
SPEED_LIMIT	The maximum speed limit assigned to a road element in kilometres per hour in accordance with Municipal By-Laws or Provincial Law.	road:speedLimit Property of a RoadLinkUser, and is linked with corresponding RoadLink using road:usedBy and road:uses properties.	cityunits:Speed

ORN_SPEED_LIMIT.csv: The maximum speed limit assigned to a road element in kilometres per hour in accordance with Municipal By-Laws or Provincial Law. In cases where a road element has more than one speed limit value, the speed limit of the longest portion of the road element is supplied.

ORN Data Tag	ORN Data Description	Property	Value
ROAD_CLASS	The classification of a road.	cdt:hasRoadClass	cdt:RoadClass



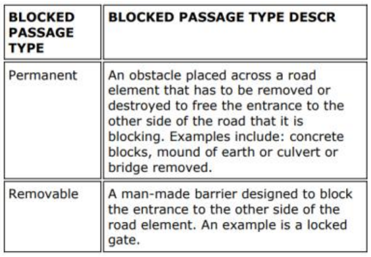
ROAD CLASS	ROAD CLASS DESCR
Alleyway / Laneway	A low speed thoroughfare dedicated to provide access to the rear of properties.
Arterial	A major thoroughfare with medium to large traffic capacity
Collector	A minor thoroughfare mainly used to access properties and to feed traffic with right of way.
Expressway / Highway	A high-speed thoroughfare with a combination of controlled access and intersections at grade level.
Freeway	An unimpeded, high speed controlled access thoroughfare for through traffic with typically no at grade intersections, usually with no property access or direct access and which is accessed by a ramp. Pedestrians prohibited.
Local / Strata	A low speed thoroughfare dedicated to provide access to properties with potential public restriction, trailer parks, First Nations, strata or private estates.
Local / Street	A low speed thoroughfare dedicated to provide full access to the front of properties.
Local / Unknown	A low speed thoroughfare dedicated to provide access to the front of properties but for which the access regulations are unknown.
Ramp	A system of interconnecting roadways providing for the controlled movement between two or more roadways.
Rapid Transit	A thoroughfare restricted 24 hours a day, for the sole use of public transportation buses.
Resource / Recreation	A narrow passage which has as a primary function access for resources extraction and also may have a role in providing an access for the public to back country.
Service	A stretch of road permitting vehicles to come to a stop along a Freeway or Highway. These include weigh scales, emergency lanes, lookouts and rest areas.
Winter	A road that is only useable during the winter months when conditions allow for passage over lakes, rivers and wetlands.

ORN_ROAD_CLASS.csv: A linear event identifying the class of road based on a functional classification schema.

Data Provided by ORN_OFFICIAL_STREET_NAME.csv:			
ORN Data Tag	ORN Data Description	Property	Value
FULL_STREET_NAME	This attribute is derived from the individual street name components where present, namely directional prefix, street type prefix, street name	genProp:hasName	xsd:string

	body, street type suffix and directional suffix and is stored in upper case text.		

ORN_OFFICIAL_STREET_NAME.csv: An event identifying an official street name and may be associated with a bilingual name.

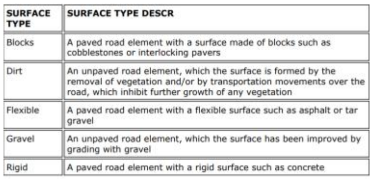
<table>
<tr><th>ORN Data Tag</th><th>ORN Data Description</th><th>Property</th><th>Value</th></tr>
<tr><td>BLOCKED_PASSAGE_TYPE</td><td>A man-made or natural barrier or access restriction placed on a road net element to control or limit access to a road net element.

<table>
<tr><th>BLOCKED PASSAGE TYPE</th><th>BLOCKED PASSAGE TYPE DESCR</th></tr>
<tr><td>Permanent</td><td>An obstacle placed across a road element that has to be removed or destroyed to free the entrance to the other side of the road that it is blocking. Examples include: concrete blocks, mound of earth or culvert or bridge removed.</td></tr>
<tr><td>Removable</td><td>A man-made barrier designed to block the entrance to the other side of the road element. An example is a locked gate.</td></tr>
</table></td><td>cdt:hasBlockedPassageType</td><td>cdt: BlockedPassageType</td></tr>
</table>

ORN_BLOCKED_PASSAGE.csv: A point event on a road element identifying the existence of an access barrier or an obstruction, either man-made or natural, which controls or limits access to a road element

<table>
<tr><th colspan="4">Data Provided by ORN_JURISDICTION.csv:</th></tr>
<tr><th>ORN Data Tag</th><th>ORN Data Description</th><th>Property</th><th>Value</th></tr>
<tr><td>JURISDICTION</td><td>An indication of who has the jurisdictional, or custodianship responsibility for a road net element. The custodian would have the responsibility to ensure maintenance</td><td>cdt:hasCustodian</td><td>org_city:GovernmentOrganization</td></tr>
</table>

	occurs, but is not necessarily the one who undertakes the maintenance directly.		

ORN_JURISDICTION.csv: Identifies jurisdictional, or custodianship responsibility of the road

<table>
<tr><th>ORN Data Tag</th><th>ORN Data Description</th><th>Property</th><th>Value</th></tr>
<tr><td>SURFACE_TYPE</td><td>A linear event indicating the surface type of a road element.

<table>
<tr><th>SURFACE TYPE</th><th>SURFACE TYPE DESCR</th></tr>
<tr><td>Blocks</td><td>A paved road element with a surface made of blocks such as cobblestones or interlocking pavers</td></tr>
<tr><td>Dirt</td><td>An unpaved road element, which the surface is formed by the removal of vegetation and/or by transportation movements over the road, which inhibit further growth of any vegetation</td></tr>
<tr><td>Flexible</td><td>A paved road element with a flexible surface such as asphalt or tar gravel</td></tr>
<tr><td>Gravel</td><td>An unpaved road element, which the surface has been improved by grading with gravel</td></tr>
<tr><td>Rigid</td><td>A paved road element with a rigid surface such as concrete</td></tr>
</table></td><td>cdt:hasSurfaceType</td><td>cdt:SurfaceType</td></tr>
<tr><td>PAVEMENT_STATUS</td><td>The surface type of a road element.</td><td>cdt:pavementStatus</td><td>xsd:boolean</td></tr>
</table>

ORN_ROAD_SURFACE.csv: The surface type of a road element.

ORN Data Tag	ORN Data Description	Property	Value
NUMBER_OF_LANES	The number of lanes of a road.	road:numLanes	xsd:integer

ORN_NUMBER_OF_LANES.csv: A linear event indicating the number of lanes.

ORN Data Tag	ORN Data Description	Property	Value

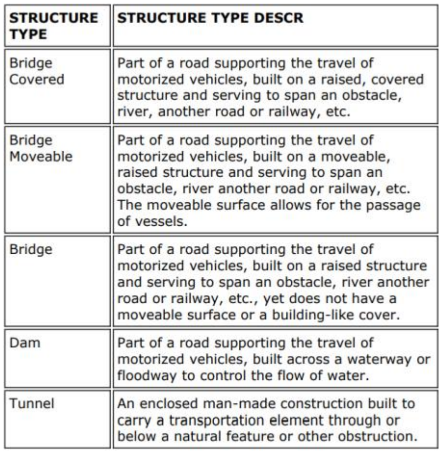
ROUTE_NAME_ENGLISH	The English name that is attached to a road net element as defined by a Municipality, Provincial Ministry, or Federal Agency and is associated to an established and/or maintained route.	cdt:routeName	xsd:string

ORN_ROUTE_NAME.csv: The name attached to a road net element as defined by a Municipality, Provincial Ministry, or Federal Agency and is associated to an established and/or maintained route.

ORN Data Tag	ORN Data Description	Property	Value
ROUTE_NUMBER	The route number assigned to a road typically associated with provincial highways, secondary highways, county roads and regional roads and is represented by a numeric and/or an alpha-numeric character. A road can be assigned multiple route numbers.	cdt:routeNumber	xsd:string Some are Alpha Numeric Values

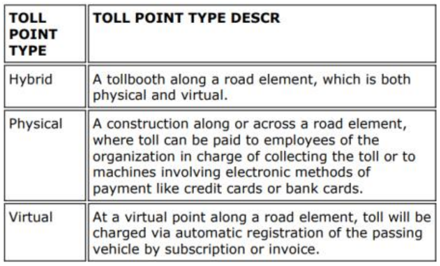
ORN_ROUTE_NUMBER.csv: The route number attached to a road net element as defined by a Municipality, Provincial Ministry, or Federal Agency and is typically associated with provincial highways, secondary highways, county roads and regional roads

ORN Data Tag	ORN Data Description	Property	Value

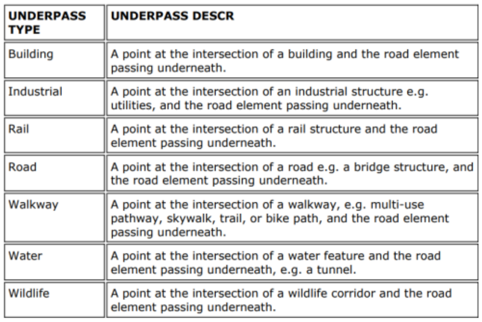
<table>
<tr>
<td>STRUCTURE_TYPE</td>
<td>The classification of a structure, that exists on a road element and is managed as a linear event.
<table><tr><th>STRUCTURE TYPE</th><th>STRUCTURE TYPE DESCR</th></tr><tr><td>Bridge Covered</td><td>Part of a road supporting the travel of motorized vehicles, built on a raised, covered structure and serving to span an obstacle, river, another road or railway, etc.</td></tr><tr><td>Bridge Moveable</td><td>Part of a road supporting the travel of motorized vehicles, built on a moveable, raised structure and serving to span an obstacle, river another road or railway, etc. The moveable surface allows for the passage of vessels.</td></tr><tr><td>Bridge</td><td>Part of a road supporting the travel of motorized vehicles, built on a raised structure and serving to span an obstacle, river another road or railway, etc., yet does not have a moveable surface or a building-like cover.</td></tr><tr><td>Dam</td><td>Part of a road supporting the travel of motorized vehicles, built across a waterway or floodway to control the flow of water.</td></tr><tr><td>Tunnel</td><td>An enclosed man-made construction built to carry a transportation element through or below a natural feature or other obstruction.</td></tr></table></td>
<td>cdt:hasStructureType</td>
<td>cdt:StructureType</td>
</tr>
</table>

ORN_STRUCTURE.csv: The classification of a structure, that exists on a road element and is managed as a linear event. The types are mutually exclusive.

ORN Data Tag	ORN Data Description	Property	Value
TOLL_POINT_TYPE	A point event on a road element identifying the existence of an underpass. An underpass occurs where the road element runs underneath a passage accommodating the movement of water, a building, road, rail, pedestrian or wildlife.	cdt:hasTollPointType	cdt:TollPointType

<table>
<tr><td></td><td>
<table>
<tr><th>TOLL POINT TYPE</th><th>TOLL POINT TYPE DESCR</th></tr>
<tr><td>Hybrid</td><td>A tollbooth along a road element, which is both physical and virtual.</td></tr>
<tr><td>Physical</td><td>A construction along or across a road element, where toll can be paid to employees of the organization in charge of collecting the toll or to machines involving electronic methods of payment like credit cards or bank cards.</td></tr>
<tr><td>Virtual</td><td>At a virtual point along a road element, toll will be charged via automatic registration of the passing vehicle by subscription or invoice.</td></tr>
</table></td><td></td><td></td></tr>
</table>

ORN_TOLL_POINT.csv: A point event along a road element indicating the presence of a toll point.

<table>
<tr><th>ORN Data Tag</th><th>ORN Data Description</th><th>Property</th><th>Value</th></tr>
<tr><td>UNDERPASS_TYPE</td><td>Identifies the type of underpass present at this road location.

<table>
<tr><th>UNDERPASS TYPE</th><th>UNDERPASS DESCR</th></tr>
<tr><td>Building</td><td>A point at the intersection of a building and the road element passing underneath.</td></tr>
<tr><td>Industrial</td><td>A point at the intersection of an industrial structure e.g. utilities, and the road element passing underneath.</td></tr>
<tr><td>Rail</td><td>A point at the intersection of a rail structure and the road element passing underneath.</td></tr>
<tr><td>Road</td><td>A point at the intersection of a road e.g. a bridge structure, and the road element passing underneath.</td></tr>
<tr><td>Walkway</td><td>A point at the intersection of a walkway, e.g. multi-use pathway, skywalk, trail, or bike path, and the road element passing underneath.</td></tr>
<tr><td>Water</td><td>A point at the intersection of a water feature and the road element passing underneath, e.g. a tunnel.</td></tr>
<tr><td>Wildlife</td><td>A point at the intersection of a wildlife corridor and the road element passing underneath.</td></tr>
</table></td><td>cdt:hasUnderpassType</td><td>cdt:UnderpassType</td></tr>
</table>

ORN_UNDERPASS.csv: A point event on a road element identifying the existence of an underpass. An underpass occurs where the road element runs underneath a passage accommodating the movement of water, a building, road, rail, pedestrian or wildlife.

~~Integration of~~Implementation of ORN Data ~~in~~ Mapping to TTL ~~Generation~~

All URIs are generated using a consistent base namespace (https://standards.iso.org/iso-iec/5087/-3/ed-1/en/ontology/TransportationNetwork/). Instances of RoadLink, Junction, and related classes are assigned URIs using their source dataset identifiers (e.g., OGF_ID, FROM_JCT) to ensure traceability. For example, RoadLinks use the format trans:roadLink_{OGF_ID} and Junctions use trans:junction_{JCT_ID}. This ensures each instance is uniquely and consistently defined in the RDF graph.

Inputs:

- ORN_ROAD_NET_ELEMENT.shp (road geometries and metadata)
- ORN_OFFICIAL_STREET_NAME.csv (maps road segments to names)
- ORN_JUNCTION.csv (contains coordinates and types of junctions)
- Other CSVs: speed limits, number of lanes, surface type, jurisdiction, etc.

Outputs

- Turtle (.ttl) RDF file that semantically represents Toronto's road network that instantiates:
    - transinfras:Road entities
    - transinfras:RoadLink entities
    - transinfras:Junction entities

Commented [MK3]: Output file(s)?

Step 1: Load and Prepare Data

- Load the shapefile and merge it with all relevant CSV files based on OGF_ID and ORN_ROAD_NET_ELEMENT_ID.
- Convert key identifiers to strings to ensure compatibility for merging and URI construction.

Step 2: Filter Geographic Bounds

- Define a bounding box for the City of Toronto.
- Ensure only roads and junctions within these bounds are processed.

Step 3: Create Junction Instances

- For each junction in ORN_JUNCTION.csv, create a transinfras:Junction (a subclass of transnet:TransportNode).
- Assign each junction a geospatial location using geo:asWKT within the geo:Geometry class.

Step 4: Group and Construct Road Entities

- Group road segments by FULL_STREET_NAME_road_names.

- For each unique group (i.e., road name), create a transinfras:Road entity.
- Assign metadata like road name, jurisdiction, and pavement status.

Step 5: Construct Road Links

- For each segment in the road group:
  - Instantiate a transinfras:RoadLink.
  - Set partwhole:properPartOf to point to the parent Road.
  - Set geo:asWKT location geometry.
  - Add attributes like speed limit, number of lanes, direction, road class, and others.
  - Set transnet:from and transnet:to relationships to link the RoadLink to its associated Junction entities.

Step 6: Associate RoadLinks with Roads

- After creating RoadLinks, collect and link them to their parent Road using partwhole:hasProperPart.

Step 7: Serialize TTL Output

- The final RDF graph is serialized to Turtle (.ttl) format for loading into graph systems.

Notes:

- The OGF_ID is used as the unique identifier for road links (ensuring traceability back to the source dataset).
- Dates (creation, revision, geo update, and effective) are formatted using a utility function to XSD-compatible 'YYYY-MM-DD' format.
- Optional attributes are conditionally added based on their availability.
- All custom or non-ISO properties use the CDT namespace.
- This structured pipeline ensures that each ORN road element is correctly modeled according to ISO 5087-3, respecting semantic constraints and spatial relationships.

Commented [MK4]: What namespace is used? What about the URIs for any other instances that you define in the mapping?

Suggest you include a brief note at the start of this section that describes how URIs are defined for all instances generated from the mapping.

~~Custom~~ CDT Classes and Properties

The following ~~custom~~ classes and properties ~~were~~ are introduced as part of the City Digital Twin ~~project~~ ontology, to extend the ISO/IEC 5087 ontologies.

Key Properties for the cdt:Junction class:

Class	Property	Value
cdt:Junction	rdfs:subClassOf	transnet:Junction
cdt:Junction	cdt:hasJunctionType	exactly 1 cdt:JunctionType
cdt:Junction	cdt:exitNumber	exactly 1 xsd:string
cdt:Junction	cdt:nationUUID	exactly 1 xsd:string
cdt:Junction	cdt:effectiveDate	exactly 1 xsd:Date
cdt:RoadLink	rdfs:subClassOf	transinfras:RoadLink
cdt:RoadLink	cdt:exitNumber	exactly 1 xsd:string
cdt:RoadLink	cdt:nationUUID	exactly 1 xsd:string
cdt:RoadLink	cdt:effectiveDate	exactly 1 xsd:Date
cdt:RoadLink	cdt:roadAbsoluteAccuracy	exactly 1 cityunits:Length
cdt:RoadLink	cdt:length	exactly 1 cityunits:Length
cdt:RoadLink	cdt:tollRoadIND	exactly 1 xsd:boolean
cdt:RoadLink	cdt:hasAquisitionTechnique	exactly 1 cdt:AquisitionTechnique

Commented [MK5]: Except for subclass definitions, all of these value restrictions should specify quantifiers. We can work through the appropriate definitions together.

cdt:RoadLink	cdt:creationDate	exactly 1 xsd:date
cdt:RoadLink	cdt:revisionDate	exactly 1 xsd:date
cdt:RoadLink	cdt:geoUpdateDate	exactly 1 xsd:date
cdt:RoadLink	cdt:hasBlockedPassageType	exactly 1 cdt:BlockedPassageType
cdt:RoadLink	cdt:hasSurfaceType	exactly 1 cdt:SurfaceType
cdt:RoadLink	cdt:pavementStatus	xsd:boolean
cdt:RoadLink	cdt:routeName	exactly 1 xsd:string
cdt:RoadLink	cdt:routeNumber	exactly 1 xsd:string Some are Alpha-Numeric Values
cdt:RoadLink	cdt:hasStructureType	exactly 1 cdt:StructureType
cdt:RoadLink	cdt:hasTollPointType	exactly 1 cdt:TollPointType
cdt:RoadLink	cdt:hasUnderpassType	exactly 1 cdt:UnderpassType

cdt:RoadLink	cdt:hasCustodian	city_org:GovernmentOrganization
city_org:GovernmentOrganization	cdt:responsibleFor	Infras:InfrastructureElement

Key Classes Created in CDT:

Commented [MK6]: I think this needs to be updated (based on our last discussion, you'll be using the Code class to define this classification, is that right?)

Description	Class	Property	Value
A class describing the Traffic is in the same direction as the geometry.	cdt:Forward	rdf:type	transnet:LinkDirection
A class describing the is opposite to the direction of the geometry.	cdt:Reverse	rdf:type	transnet:LinkDirection
A class describing the is allowed in both directions.	cdt:Bidirectional	rdf:type	transnet:LinkDirection
	cdt:StructureType	code:hasCode	code:Code
	cdt:TollPointType	code:hasCode	code:Code
	cdt:UnderpassType	code:hasCode	code:Code
	cdt:BlockedPassageType	code:hasCode	code:Code
	cdt:SurfaceType	code:hasCode	code:Code
	cdt:JunctionType	code:hasCode	code:Code
	cdt:AquisitionTechnique	code:hasCode	code:Code
	cdt:RoadClass	code:hasCode	code:Code