

# INM713 Semantic Web Technologies and Knowledge Graphs

*Konstantinos Meli & Quang Huy Bui*

*MSc Data Science*

*City University of London*

[konstantinos.meli@city.ac.uk](mailto:konstantinos.meli@city.ac.uk) & [quang.bui.2@city.ac.uk](mailto:quang.bui.2@city.ac.uk)

## Prefixes

The ontology contains default prefixes by Protégé and other added prefixes by the authors. The default prefixes which contains:

Prefix name	IRI
rdf	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
rdfs	<a href="http://www.w3.org/2000/01/rdf-schema#">http://www.w3.org/2000/01/rdf-schema#</a>
xml:	<a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>
xsd	<a href="http://www.w3.org/2001/XMLSchema#">http://www.w3.org/2001/XMLSchema#</a>
owl:	<a href="http://www.w3.org/2002/07/owl#">http://www.w3.org/2002/07/owl#</a>

Prefixes added by the authors are:

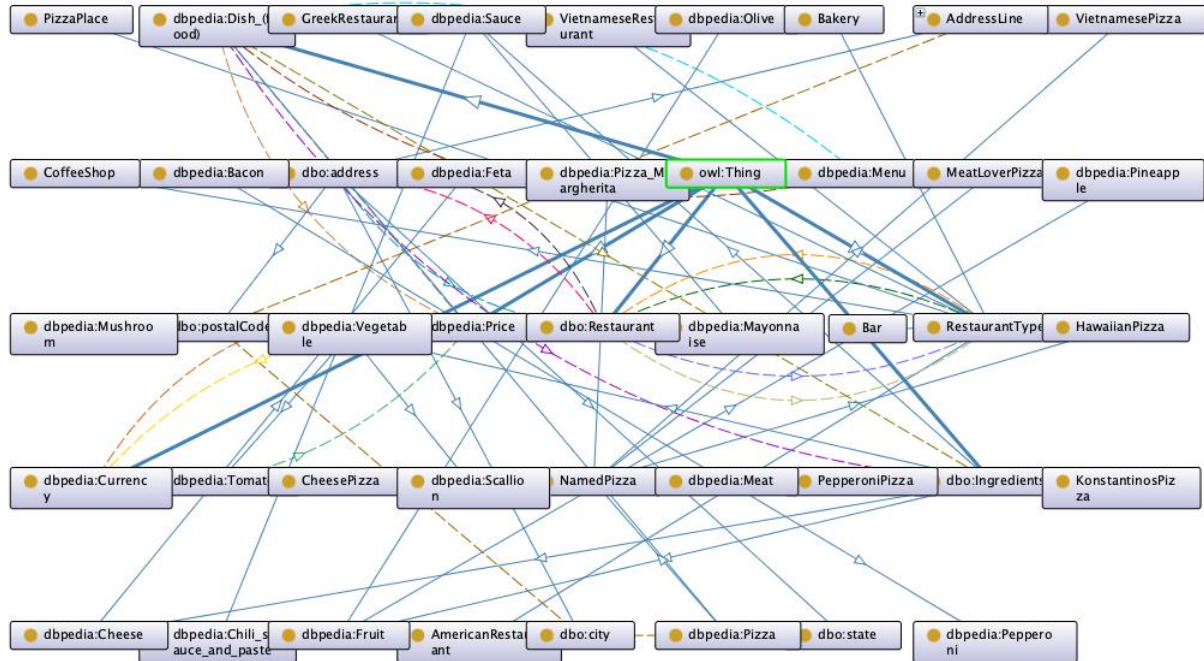
Prefix name	IRI
dbo	<a href="http://dbpedia.org/ontology/">http://dbpedia.org/ontology/</a>
dbp	<a href="http://dbpedia.org/property/">http://dbpedia.org/property/</a>
dbr	<a href="http://dbpedia.org/resource/">http://dbpedia.org/resource/</a>
dc	<a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>
rec	<a href="http://www.city.ac.uk/inm713-in3067/2023/recruitable/">http://www.city.ac.uk/inm713-in3067/2023/recruitable/</a>

- `rdfs` (RDF Schema) is a vocabulary that builds on top of rdf and is used to give additional modelling capabilities for describing and organising resources.
- `owl` (Web Ontology Language) It is an extension of rdf and is a semantic markup language used to share knowledge from ontologies on the world wide web. It also provides a rich vocabulary and syntax to create ontologies by describing relationships between resources.
- DBPedia (`dbo`, `dbp` and `dbr`) extracts information from wikipedia making it available for public use on the web. In this ontology dbo, dbp and dbr are prefixes describing ontology, properties and resources respectively. Each of these types of resources describe their own types of vocabularies and classes.
- `dc` (Dublin Core) is a prefix from rdf to represent the Dublin Core metadata vocabulary that has a set number of elements used to describe resources on the web. The element used in this ontology is dc:creator to describe the authors of this ontology.
- `rec` (Recruitable) is the ontology created by the authors, which is the acronym of the group name.

## Classes

The main focus of class modelling in this work is to maximise vocabulary re-usage from legitimate public knowledge graphs such as DBpedia by utilising reflexes. Therefore, the ontology can deliver high knowledge entropy while the user or computer follows IRIs.

To achieve this, after identifying a class, we search for it on current public knowledge graphs first and in the case that it is not available online, it will be created using our prefix.



The data was investigated by fields (columns) and rows (records) to obtain a class list from the dataset. Some column names contain enough information for a class construction (for example, the `Price` class was purely made from columns `item value` and `currency`). Meanwhile, investigating the `item description` column, we can see that some records show information about the dish ingredient, while others show random dish notes from the restaurant. Therefore, an iteration on the records was conducted to filter the subclasses for `Ingredient`.

## Properties & properties restriction

Object properties are the bridge between two individuals (subject and object). They define how the individual relates to other people in a given environment. Moreover, they provide each individual with unique attributes or characteristics. In essence, these properties work as verb phrases (predicate) in sentences (triples).

The properties list is obtained by analysing the relationship between identified classes. In order to build the properties, the same approach as the creation of a class is used: searching for the vocabulary online first and self-build using author's prefixes if not available.

Other aspects or properties have been added accordingly. Properties' inversion describes two opposite relationships between classes (`hasPrice` is the inversion of `isPriceOf`). Essential properties' characteristics were also added. For example, `dbp:locatedIn` is transitive, applied to the address and subclasses.

## Labels & comments

Labels are names that can be identified by both humans and machines to identify a resource in an ontology. Typically it is used to understand what a resource is and can be provided in multiple languages allowing ontologies to have multilingual labelling.

Comments provide additional information about resources expanding on labels. They explain the meaning and use of a resource or entity in order to be helpful to the users.