Ireland Schools
GROUP E

1. Approach to Ontology Modelling

This project uses distinct databases to create a separate ontology relating these two datasets.

a. GeoHive dataset defining area on various counties inside Ireland (http://data.geohive.ie/dumps/county/default.ttl)

b. List of Post Primary Schools for academic year 2016/2017 (https://data.gov.ie/dataset/post-primary-schools-list-2017/resource/8e42619c-768e-4366-94bb-77b2c5f089d0), which defines different schools with their attributes. The most important attributes among them are longitude & latitude, which would allow us to connect this dataset with the first one.

The attempt has been made to model a simple ontology which can further be modified and used to accommodate similar datasets.

i. Assumptions made:
   a. The county column present in the second dataset has been ignored and more dependent attributes for location i.e. latitudes and longitudes are used. The location is more dependable, as 1. in some cases (or in some datasets in future) the county column might be missing, or 2. some spelling mistakes due to human error might cause an unintended effect which may leave a record unlinked.
   b. The type of school (e.g. primary, secondary, community college etc.) is identified depending on its name as there is not any explicit 'type' column in the school dataset. In the present scenario, many schools are identifiable using this method; while other remaining schools have been marked under 'Others' category.

ii. References to sources used/reused e.g. SIOC, FOAF for people

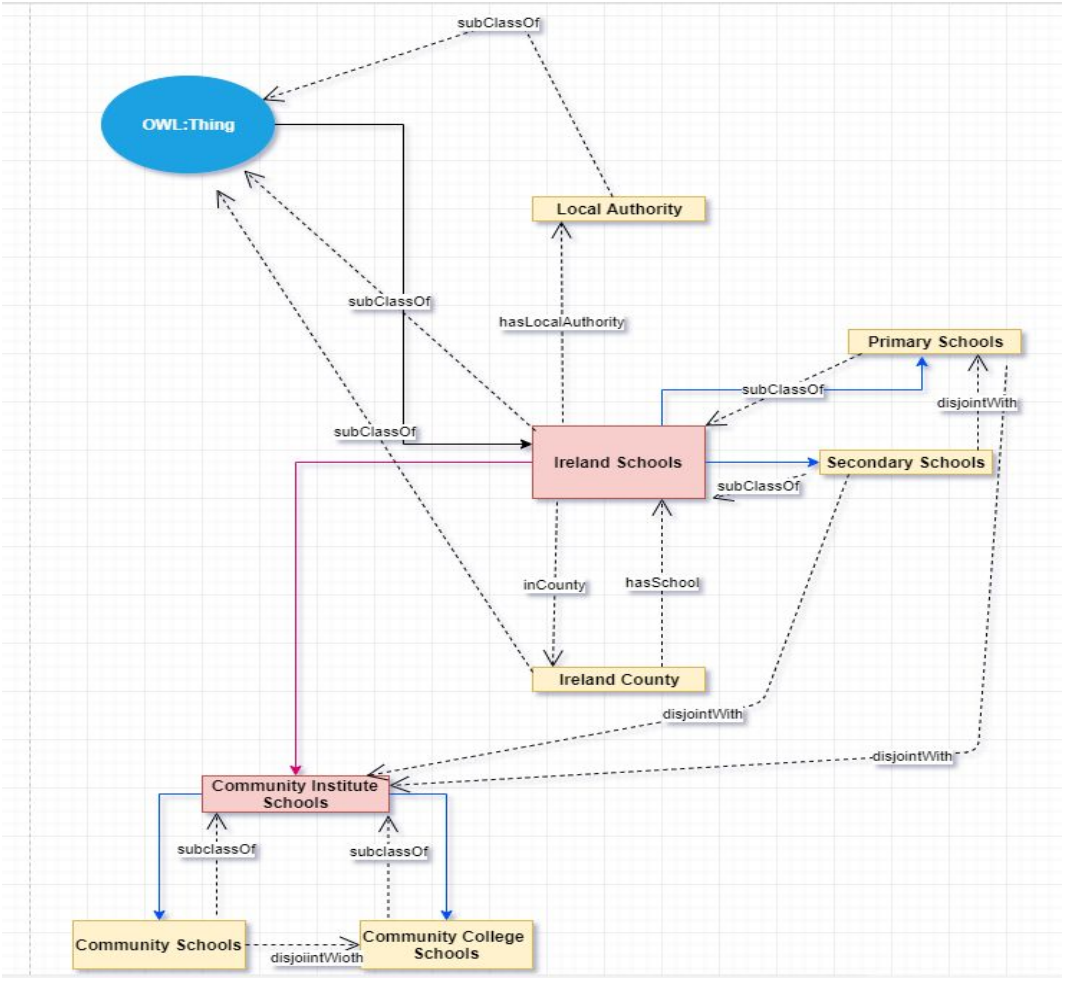
This ontology is designed to depend on well-known, lightweight and basic schemas to define different components of the design, as below:

a. @prefix owl: <http://www.w3.org/2002/07/owl#>
b. @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
c. @prefix xsd: <http://www.w3.org/2001/XMLSchema#>
d. @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>

iii. Your data conversion/input process

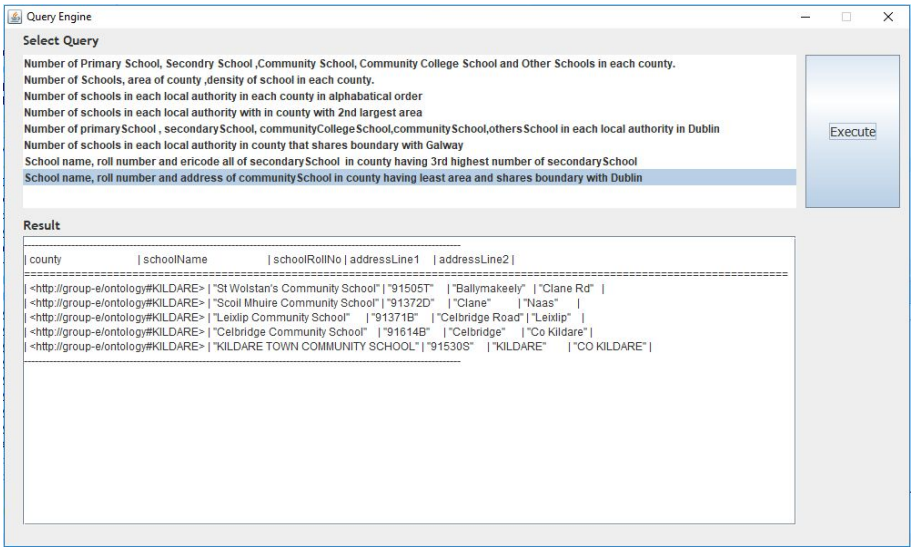
Mapping of CSV (tabular data) into OWL ontologies and RDF vocabularies has been done using the Juma Editor.Mapping definitions were created having subjects, predicates and objects.

2 Overview of Design



a. Description of the query interface

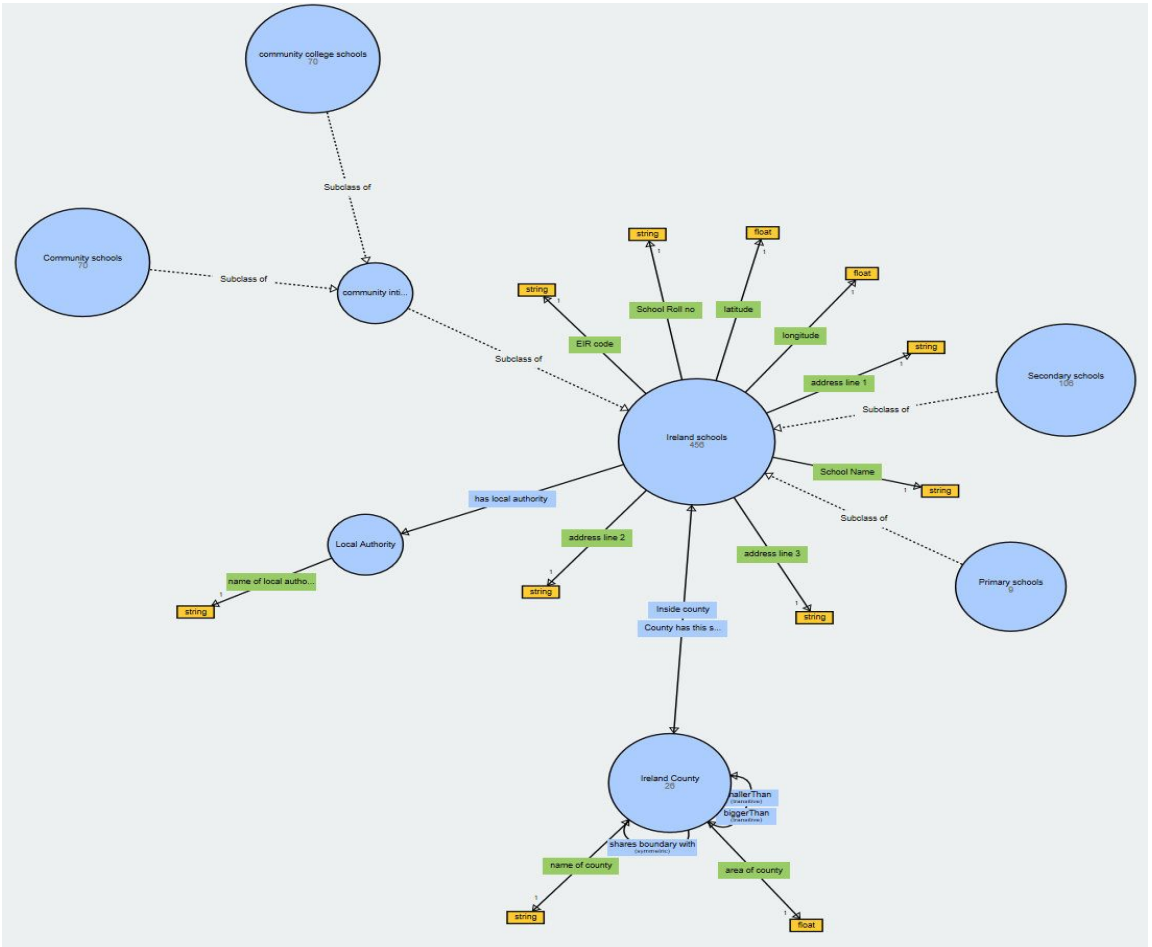
The Query Engine was created with the help of swing package in JAVA. The input to the application came from a number of query files containing the query description and SPARQL query. Based upon the selection of the description from the GUI, the appropriate SPARQL query mapped to it was fired and the result was formatted to text and displayed on the GUI.



b. Description of Queries and SPARQL– Listed in Appendix 1.

c. Documenting Ontology using LODE(Widoco):

Firstly, Metadata was added in the ontology model which provides the necessary information. We have used Widoco, that re-uses LODE internally and after executing it a Folder is generated consisting of HTML Files and WebOWL diagram representation.



3 Challenges faced while ontology modelling or creating queries:

Defining overlapping relationships between different type of schools was a crucial issue. e.g. community type may contain two different type under it *viz.* community college and community schools. To address this, a superclass 'community institute', that holds these two subclasses under it is defined. Also, primary and secondary schools might be under community institute or independent. To address this, these two are made disjoint with each other; but not made disjoint with community institutes. Local Authority is the class added later in the development (when we understood the concepts and domain in depth) to accommodate it as en entity rather than as a mere string value (as we had thought before).

Once we verified our ONTOLOGY with LODE we came to know that there are some critical changes that need to be corrected, but while doing those changes we faced another issue that most of our queries were not working, so finally we re-created all the queries from scratch based on our new ONTOLOGY.

4 Conclusions: Strengths/weakness of the ontology model, queries & interface

In order to meet time constraints and lay outline to develop for future work, we have assumed certain things (see Assumptions section.) which can further be improved on. For instance, instance creation is dependent on 'official name of school' column in the second dataset. This need to be changed or improved on in future to make it more dependable.

At the same time, the primary feature of this project is connecting school records with accurate location measures as given in GeoHive dataset. This gives more strength to ontology as previously impossible/troublesome information retrieval is now possible, as shown by queries. E.g. We can find the density of schools in different counties.

Self-Reflections:

Paras V. Prabhu (18305357):

Contributed towards ontology modelling at the initial phase of the project together with Mr Pavan. This involved understanding both the datasets and ontology modelling. In the second part work independently on programming side which involved individuals creation, establishing proper relationships and writing this to final ontology file (Turtle).

I think, we could have developed more proper modelling to accommodate this as well as future data, if we had some more time to work on this. As I had most of my experience working on Object-oriented terminologies, this first-hand experience with Linked Data concepts allowed me to get out of OOPs, to think and relate things in terms of the linked data manner().

Shanmukha Sai Ram Pavan Parvathina(18305688):

Contributed towards the selection of the initial dataset, selection and development of several SPARQL queries, editing and creating the final report. According to me, my biggest contributions is towards the selection of database and framing queried on it which is a very important task. My group has performed really well and within the limited time we came up with a very efficient ontology with all required features.

Shubham Pandey(18304352):

Contributed towards the selection of Initial dataset, selection and development of several SPARQL and also suggested changes in ONTOLOGY based on LODE result. Also, uplifted the second dataset. The strength of my contribution is the uplifting of the data and effective changes based the LODE results, I think we should have taken a reference of LODE throughout the development process of ontology. According to me, a better ONTOLOGY can be created if we would have given a bit more time as well as SPARQL queries can also be enhanced based on the new ONTOLOGY. This project taught me a new approach to handle data which is far more advanced than the relational database. Regarding the group, overall a better ontology and advanced GUI can be achieved using a reference to this project.

Aditya Misra(18302706):

Contributed towards the selection of Initial dataset, Uplifting the Dataset, Documenting the Ontology using WIDOCO, suggested changes in ONTOLOGY based on LODE result, creating visualizations using Draw.io and editing of the final report. To check whether a ontology created is correct or not is one of the most imported task of this project and I think I have well contributed towards it, using LODE.

Amogh Pandit(17313998):

Contributed towards the selection of Initial dataset by searching and analysing different datasets to match with the Geohive dataset, analysing mapping of how data can be mapped from one dataset to another dataset and also worked on formatting and editing of the final report. Previous experience of working in JAVA helped to develop the GUI for query execution within the given time. Many challenges came will developing GUI like the result of the query was not displaying in an appropriate format. For that developed a function to parse the result and display in a table format. But due to limited time, we were not able to test it properly. The queries created to test our Ontology could have been more complex so as to get a better understanding and use of the ontology created from the two datasets.

Appendix 1: Questions and SPARQL queries

Common Prefix:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX base: http://group-e/ontology#
```

1)Number of Primary School, Secondary School ,Community School, Community College School and Other Schools in each county.

```
SELECT ?county ((COUNT(?schoolS)) AS ?noOfSecondarySchool) ((COUNT(?schoolP)) AS
?noOfPrimarySchool) ((COUNT(?schoolCS)) AS ?noOfCommunitySchool)
((COUNT(?schoolCCS)) AS ?noOfCommunityCollegeSchool) ((COUNT(?schoolOS) -
?noOfSecondarySchool - ?noOfCommunityCollegeSchool - ?noOfCommunitySchool -
?noOfPrimarySchool) AS ?noOfOthersSchool)
WHERE {
{?schoolS a base:secondarySchool.
?schoolS base:inCounty ?county.
}
UNION
{?schoolP a base:primarySchool.
?schoolP base:inCounty ?county.}
UNION
{?schoolCS a base:communitySchool.
?schoolCS base:inCounty ?county.}
UNION
{?schoolCCS a base:communityCollegeSchool.
?schoolCCS base:inCounty ?county.}
UNION
{?schoolOS a base:school.
?schoolOS base:inCounty ?county.}
} GROUP BY?county
```

2)Number of Schools, area of county ,density of school in each county.

```
SELECT ?county  ?area ((COUNT(?schoolS)) AS ?numberOfSchools)
((COUNT(?schoolS)/?area) AS ?densityOfSchools)
WHERE
{?schoolS base:inCounty ?county.
?county base: area ?area
} GROUP BY?county ?area
```

3)Number of schools in each local authority in each county in alphabatical order

```
SELECT ?county ?localAuthorityName   ((COUNT(?schoolS)) AS ?numberOfSchools)
WHERE
{?schoolS base:inCounty ?county.
?schoolS base:hasLocalAuthority ?localAthority
?localAthority  base:localAuthorityName ?localAuthorityName
} GROUP BY?county ?localAuthorityName order by ?county
```

4)Number of schools in each local authority with in county with 2nd largest area

```
SELECT ?county ?localAuthorityName  ((COUNT(?schoolS)) AS ?numberOfSchool)
WHERE
{?schoolS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .
?schoolS base:inCounty ?county  .
{
SELECT  ?county  ?area
WHERE {
SELECT  ?county  ?area
WHERE {
 ?county base:area ?area
} ORDER BY DESC (?area) limit 2}
ORDER BY ?area
LIMIT 1
}} GROUP BY? LocalAuthorityName ?county
```

5)Number of primary School , secondary School, community College School, community School, others School in each local authority in Dublin

```
SELECT? LocalAuthorityName ((COUNT(?schoolS)) AS ?noOfSecondarySchool)
((COUNT(?schoolP)) AS ?noOfPrimarySchool) ((COUNT(?schoolCS)) AS
?noOfCommunitySchool) ((COUNT(?schoolCCS)) AS ?noOfCommunityCollegeSchool)
((COUNT(?schoolOS) -  ?noOfSecondarySchool - ?noOfCommunityCollegeSchool -
?noOfCommunitySchool - ?noOfPrimarySchool) AS ?noOfOthersSchool)
WHERE {
{
SELECT  ?localAuthorityName
                        WHERE
                        {?schoolS base:hasLocalAuthority ?localAthority.
                        ?localAthority  base:localAuthorityName ?localAuthorityName .
                        ?schoolS base:inCounty base: DUBLIN
                        } GROUP BY?localAuthorityName}
{?schoolS a base:secondarySchool.
```

```
?schoolS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .}
UNION
{?schoolP a base:primarySchool.
?schoolP base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .}
UNION
{?schoolCS a base:communitySchool.
?schoolCS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .}
UNION
{?schoolCCS a base:communityCollegeSchool.
?schoolCCS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .}
UNION
{?schoolOS a base:school.
?schoolOS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .}
} GROUP BY?localAuthorityName
```

6)Number of schools in each local authority in county that shares boundary with Galway

```
SELECT ?county ?localAuthorityName   ((COUNT(?schoolS)) AS ?numberOfSchools)
WHERE
{
{ SELECT ?county
WHERE {
base: GALWAY base:sharesBoundaryWith ?county.
}}
?schoolS base:inCounty ?county.
?schoolS base:hasLocalAuthority ?localAthority.
?localAthority  base:localAuthorityName ?localAuthorityName .
} GROUP BY ?county ?localAuthorityName order by ?county
```

7)School name, roll number and address of community School in county having least area and shares boundary with Dublin

```
SELECT ?county  ?schoolName  ?schoolRollNo ?addressLine1 ?addressLine2
WHERE
{
?schoolS a base:communitySchool.
?schoolS base:inCounty ?county.
?schoolS base:addressLine1 ?addressLine1.
?schoolS base:addressLine2 ?addressLine2.
?schoolS base:officialSchoolName ?schoolName.
?schoolS base:schoolRollno ?schoolRollNo.
```

```
{
SELECT  ?county
WHERE {
base: DUBLIN base:sharesBoundaryWith ?county.
?schoolS base:inCounty ?county.
?county base:area ?area.
} ORDER BY ?area LIMIT 1 }
}
```

8) School name, roll number and eircode all of secondary School in county having 3rd highest number of secondary School

```
SELECT ?county  ?schoolName  ?schoolRollNo ?eriCode
WHERE
{
?schoolS base:eirCode ?eriCode.
?schoolS a base:secondarySchool.
?schoolS base:inCounty ?county.
?schoolS base:officialSchoolName  ?schoolName.
?schoolS base:schoolRollno ?schoolRollNo.
{
SELECT  ?county   ?noOfSecondarySchool
WHERE {
SELECT  ?county ((COUNT(?schoolS)) AS ?noOfSecondarySchool)
WHERE {
?schoolS a base:secondarySchool.
?schoolS base:inCounty ?county.
} GROUP BY ?county ORDER BY DESC (?noOfSecondarySchool) limit 3}
ORDER BY  ?noOfSecondarySchool
LIMIT 1
}}
```