INFO 116

OBLIGATORY GROUP ASSIGNMENT

Candidate numbers:

219

169

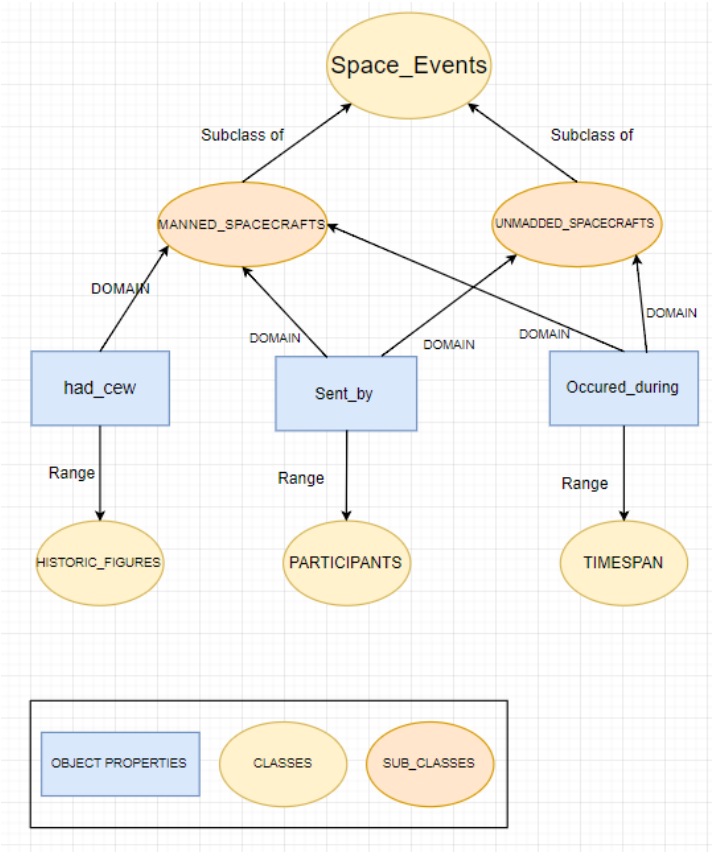
201

Task 1

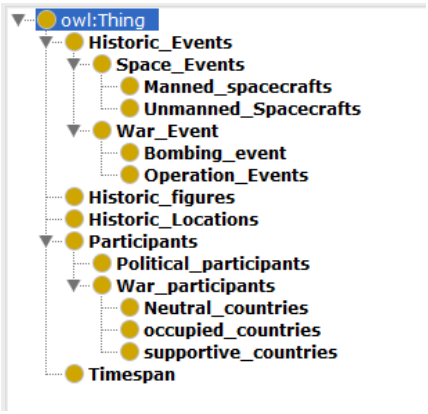
When we received our task and found our group, we were to enhance website design with the help of semantics, while also create new ontologies to help represent details of previous historic events given to us in the document "The Most Important Events of the Century from the Viewpoint of the People". We also decided to choose the floating titanic "mhtml" file.

To complete this assignment, we had to manage tasks between ourselves. We came to the agreement of having **candidatenr: 219** (me) write this report/document as well as to find and create ontologies in protégé as I felt most comfortable with the program out of the group. All ontologies in our ".owl" file was created by myself, as I found myself feeling that importing them was quite the hassle and not as fun. As we were to cover a wide variety of events, finding relevant information was time consuming, so to speed up the process I had **candidatenr: 201** help me find information of each topic, which we then discussed and planned on where to structure our ontologies. **candidatenr:** 201 was also in charge of working on "SPARQL" queries to show how a potential application could use our ontologies. We had one more group member who was supposed to work on "RDFa Lite", but he rarely responded/ignored all our Facebook messages, leaving them on read, and as of today (nov.24.18) writing this report, he has yet to start on his assignment. **candidatenr:** 169 volunteered to do the work on "JSON-LD" and OGP as he felt most comfortable with it. (comment: each person is supposed to explain and comments on how they completed their tasks in this report).

Before making my ontologies, I made some sketches in draw.io so I could get an overview of how I would organize my ontologies. Below is a picture of one of my figures I created:



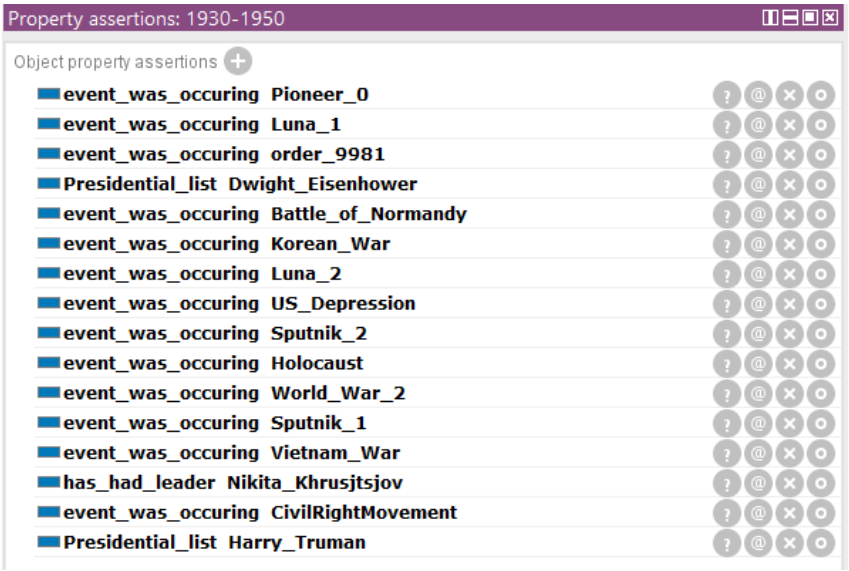
In protégé I found myself starting off by mapping out all classes I planned on using, but as I added more and more individuals from the historic periods, I found myself adding more classes/subclasses and modifying existing ones to have them coexist in the best way I was able to. All data can be found in the .owl file linked in the document submitted.



As I was adding the individuals, I made sure to link them with the classes by using object properties. I tried to recycle as many object properties as I could throughout my individuals, to avoid having an overflow of rarely used object properties. I also made some data properties to add metadata to some of the individuals. I did not add data properties to every single individual, as it is quite time consuming, but I did add to some, and I hope it helps show how we did it.

I linked every event to a timespan, as I thought it would make it easier to access data if it were to be used in a search engine. I created four time periods to represent all historic events, which I then linked to the class: “Timespan”. The picture below show us which centuries we used:

I found myself liking this format, as if we were to visit one of the time periods, the object property assertions easily show which events, happened in that exact time period. As an example we can easily see every event occurring in 1930-1950 period by looking at the object property assertions linked to the individuals.



Task 2.

In this task, my objective was to create some SPARQL queries that could prove useful and relevant for our ontology. At first, I tried working with the SPARQL query provided in Protégé. I had some trouble getting certain code to work, and the output I got seemed Inconsistent at times. At some points, code that had previously worked didn't return any output. A simple restart was all that was required, but this left me wondering if my code was wrong, or if I had to restart Protégé once again. This ordeal led me to use Apache Jena Fuseki, which I found to be more reliable and overall more pleasant to work with. The table result provided by Protégé was generally much easier to read. But it was sometimes faultier than the results provided by Fuseki, which led me to use the best output in each query.

```
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 vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>
PREFIX foaf:  <http://xmlns.com/foaf/0.1/>
SELECT ?object ?subject ?name

WHERE
{
?name rdfs:subClassOf ?subject .
?subject rdfs:subClassOf ?object .
}
```

Query figure 1.

Note: I didn't use all the prefixes in the query, but left in the ones that I found relevant for future use. The goal of this query, was to get a simple table overview of the classes in our ontology. Using an instance of rdfs called "subClassOf" I can say that name is a subclass of subject and subject is a subclass of object. The result of running this query looked like this:

object	subject	name
Historic_Events	War_Event	Bombing_event
Participants	War_participants	supportive_countries
Historic_Events	War_Event	Operation_Events
Participants	War_participants	occupied_countries
Historic_Events	Space_Events	Unmanned_Spacecrafts
Historic_Events	Space_Events	Manned_spacecrafts
Participants	War_participants	Neutral_countries

Query figure 2.

The output of this query shows all classes in order of which class is on top in the class hierarchy.

```
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 vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>
SELECT ?individual ?subject
WHERE {
    ?individual rdf:type owl:NamedIndividual .

    ?individual rdf:type ?subject .
}
```

Query figure 3.

If you look at figure 3, one can see that I implemented the owl prefix and an instance of it called “NamedIndividual”. This allows for declaration of named individuals in the ontology. Also used is the instance of the rdf prefix called: “type”, which states that something is an instance of a class.

1	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Alan_Bean>	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Historic_figures>
2	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Alan_Bean>	owl:NamedIndividual
3	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#United_States_of_America>	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#War_participants>
4	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#United_States_of_America>	owl:NamedIndividual
5	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Apollo_12>	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Manned_spacecrafts>
6	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Apollo_12>	owl:NamedIndividual
7	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#1960-1980>	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Timespan>
8	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#1960-1980>	owl:NamedIndividual
9	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Apollo_11>	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Manned_spacecrafts>
10	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Apollo_11>	owl:NamedIndividual

Query figure 4.

What I effectively ended up with here is a table which contains the subjects shown in figure 2, and what individuals the subjects contain. A problem I ran into with this query is that every Individual is printed twice. I tried working around this, and got different results, where everything was seemingly correct, but that query also gave me datatypes which I wasn’t looking for in this instance. Other than that, this query works fine, and proves useful to highlight under which class every individual is located. It’s Worth mentioning that I only included 10 rows due to the length of our ontology.

```
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 vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>
SELECT ?individual ?name
WHERE {

    ?name rdf:type owl:DatatypeProperty.

    ?name rdf:type ?individual .
}
```

Query figure 5.

Figure 5 is similar to figure 3 structurally. The information we receive is not similar however. In this query we still use the owl prefix but a with a different instance.

	individual	name
1	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#bomb_location_dropped>
2	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_bomb_dropped>
3	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_built>
4	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_destroyed>
5	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_of_death>
6	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_of_landing>
7	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_war_disestablished>
8	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#date_war_established>
9	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#death_by>
10	owl:DatatypeProperty	<http://www.semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#had_current_title>

Query figure 6.

Through using the “DatatypeProperty” instance, we get all the Data properties of individuals in the query. Unfortunately as seen in figure 6 below, I didn’t manage to make the data properties link up to their individuals. Other than that, this query serves to show the user all data properties used in the ontology.

Task 3:

Below is the code for some basic Open Graph Protocol Metadata for Facebook. This code is added in the start of the <head> - section. The first line of the code defines which kind of type this is, which in our case is an article. The next line then shows the url - address in where to find this article. The third line tells us the name of the article, and the last line is a url to the image of the page.

```
<!-- Group="0" -->

<head prefix="og: http://ogp.me/ns# article: http://ogp.me/ns/article#">
    <meta property="og:type" content="article" />
    <meta property="og:url" content="https://www.britishnewspaperarchive.co.uk/viewer/bl/0001715/19970329/553/0028" />
    <meta property="og:title" content="Floating Titanic sinking NY show" />
    <meta property="og:image" content="https://thumbs-api.britishnewspaperarchive.co.uk/API/thumb/BL/0001715/19970329/0028.jpg" />

    <script type="application/ld+json">
```

Below we can see a screenshot of a section of the JSON-LD code. This code should also preferably be put in the <head> - section of the document, but can in some cases also be put in the <body> - section. The first thing we had to do is to define that JSON-LD is a <script> - type of code, then we have to create an object which we can do by using curly brackets ({}). The next line then links to where to find this data we are defining, and here is a url to both schema.org and our own self-made ontology, and because there are two different sources we have to put them inside an array with []. The next thing to do is to define the type, which in our case is “Article”, and then the name of this article. In our case this article specifically is about the musical “Floating Titanic”, which we have already defined in our own ontology, as well as the actual ship “Titanic”, so we combined schema.org and our ontology to state what this article is about. The last lines of code are defining who published the article, and that it was used in a newspaper called Irish Independent on the 29th of March 1997.

```
<script type="application/ld+json">
    {
        "@context": [
          "https://schema.org",
          "http://semanticweb.org/hallv/ontologies/2018/10/untitled-ontology-18#Titanic"
        ],
        "@type": "Article",
        "name": "Floating Titanic sinking NY show",
        "about": [
          "Titanic",
          "Floating_Titanic"
        ],
        "publisher": "Irish Independent",
        "genre": "Newspaper",
        "datePublished": "1997-03-29",
        "story_based_on": "ocean_liner_disaster",
        "occured_during": "1990-2010",
        "event_located": "New_York",
        "incident_in_the_musical": "Floating_Titanic",
        "has_publisher": "Susanne_Tighe",
        "has_writer_and_composer_for": [
          "Maury_Yeston",
          "Peter_Stone"
        ],
    }
</script>
```

Comment: The last group member, said he would complete the “rdfa” assignment, as he had access to all out files, but we did not hear from him on Tuesday the 27th (due date) as we had minimal time left, we did not have time to complete that part and I hope you take it into consideration. We tried everything, calling, writing messages, it said he even was online but did not read our messages. It seems like his plan was to free load on our group and hope to get a grade, so therefore we did not include him/his name or his candidate nr in this report.

SOURCES:

Ding L. Khandelwal A. (2012) The World Wide Web Consortium [Internet] Available from: https://www.w3.org/TR/rdf-schema/#ch_type

The Apache Software Foundation(2011) SPARQL Tutorial [Internet] Available from: https://jena.apache.org/tutorials/sparql.html

https://stackoverflow.com/questions/44205661/how-to-import-specific-classes-and-object-properties-from-an-ontology-in-protege

https://protegewiki.stanford.edu/wiki/Importing_Ontologies_in_P41

https://schema.org

https://en.wikipedia.org/wiki/Wikipedia:List_of_online_newspaper_archives

http://sparql.ub.uib.no/

https://search.google.com/structured-data/testing-tool#

http://linter.structured-data.org/

https://en.wikipedia.org/wiki/World_War_II

https://www.britishnewspaperarchive.co.uk