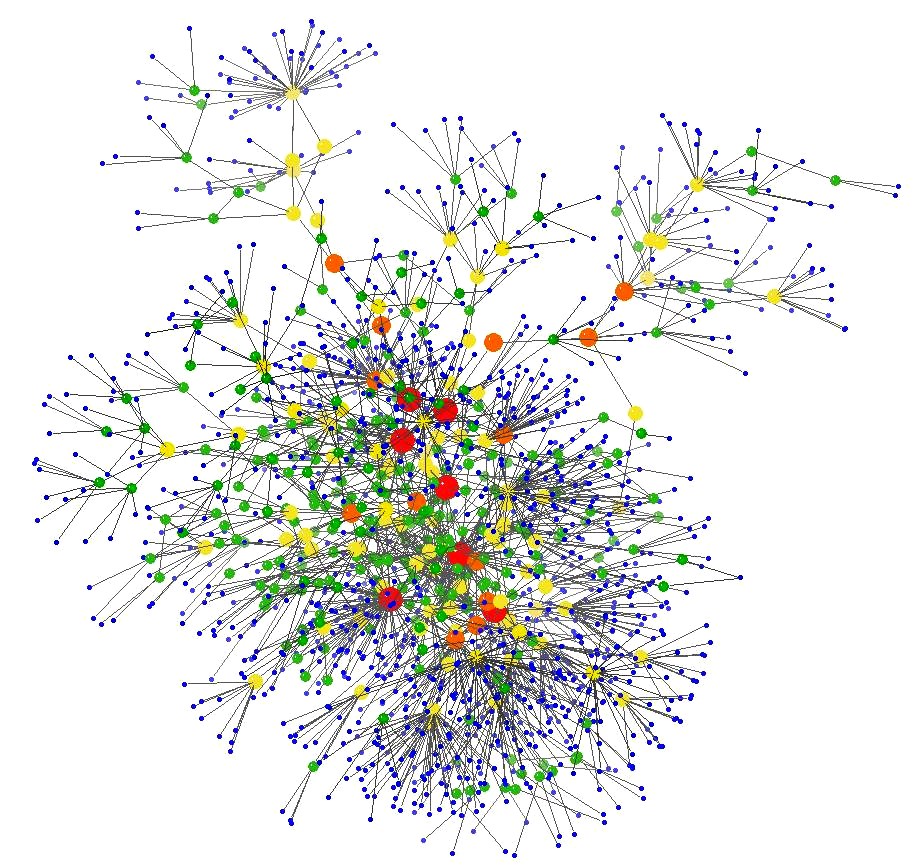
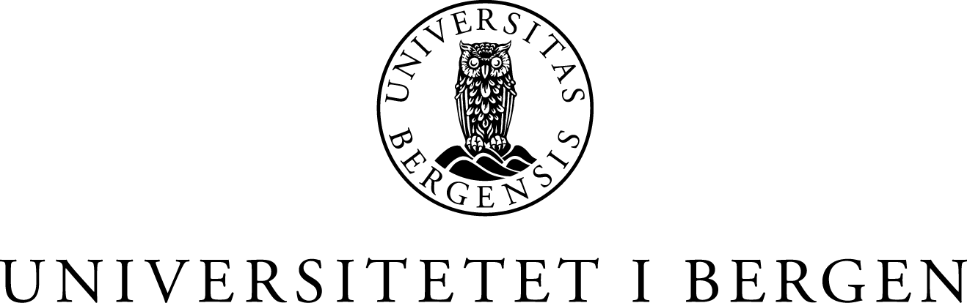
INFO116 ASSIGNMENT FALL 2018

MAKING SENSE OF HISTORY



Candidates: 135, 161, 204, 227



# Introduction

*“A single, fixed ontology cannot capture the relevant contextual knowledge for all of the documents in our collection, because the entities in the world and the relations among them—and consequently, the configuration of the ontology-differ depending on the date of the document”* (Ide & Woolner, 2007)

In the assignment we were given, the main task was to create meaning of concepts and events described in historical articles. We were given three articles and a commentary article about significant historical events throughout the 21th century. With this, we were meant to create a reasonable ontology, annotate one of the articles and create several SPARQL queries based on the ontology. Our approach to the assignment was to get an early start and acquire knowledge about the programs and tools necessary for the task, as well as truly understanding the content of the articles and the assignment as a whole. Looking in retrospect we acknowledge that there are several paths to a reasonable ontology, especially concerning an extensive topic like history.

We started by creating the ontology in Protege. As there is no clear way in how to structure the ontology we used substantial time on deciding what to include, and how to structure it properly. Our approach tries to capture both the historical events with information alone, and in combination with the articles. We always had in mind that we were supposed to use it later both in the SPARQL queries and the annotation part.

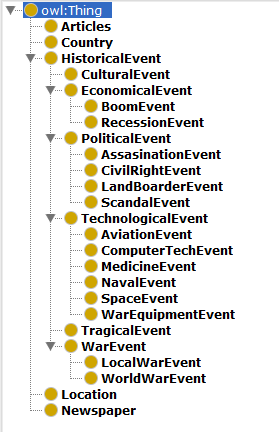
Following the ontology’s creation, we crafted SPARQL queries based on our own competency questions. We also annotated the website using RDFa-lite and JSON-LD, as well as annotating with the Facebook OGP.

Throughout the report we have discussed, explained and justified our approaches to the various parts of the assignment in greater detail. We will also reflect upon the added value semantics gives to web pages and give a conclusion on how our work is compatible with its purpose. We also discuss the group project, our experiences, contributions and challenges along the way in the very last part of the report.

# Ontology

Creating an ontology that encompasses all historical articles is not an easy task. This exercise shows the importance of well-thought ontologies for good user-experience and unbiased categorizing of data. To do so, we used a top-down approach trying to capture the bigger picture first and then specifying more details as we continued. We therefore pictured the ontology as a hierarchical drop-down menu of a web-page and structured our ontology according to this. One choice we had to make early on was what our ontology should depict. We chose to focus on the event occurrences and how to link the articles to those events. A central reason for that was the competency questions we wrote before and during the creation. Being able to answer complex questions and give valuable information about historical events, and the content of the articles in combination with these historical events, were our main goal.

### Classes

First, we sought inspiration through different history timelines and tried to find encompassing terms that would help create a reasonable ontology while staying unbiased. Since history is defined as past events, particularly in human affairs, we tried to subcategorize history into wide subclasses with significant importance to almost any human. These subclasses can be seen in our ontology as economical events, political events, war events, etc. The “Event” behind each subclass is to make the ontology more user-friendly and easy to understand for potential users. Also, having simplicity continuously on our minds, we tried to avoid making too many subclasses for each superclass. As shown in Figure 1, each class leads to a more specific part of history where it eventually ends up in well-defined historical happenings - in our ontology linked as individuals to its respective class.

Furthermore, a large part of the assignment was to combine such historical events with relevant news articles. Therefore, the class “Articles” was created adding each article as an individual of the class. We also did a bit of research and added extra articles to make the ontology more robust and exciting. This also opens for improved annotation later in the assignment. In addition, we added the classes “Country”, “Location” and “Publisher” to widen our opportunities with the queries.

### Object properties

As the foundation of the ontology started to take shape, we started to link the data together. We created object properties to link classes and instances with each other. This is shown in the picture below. As an example, “Germany” can be linked by the object property “isParticipant” to the individual “WW2” of the class WorldWarEvent.

Figure 1 Main ontology

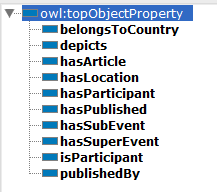
We created these object-properties shown above as means to give further depth to our own ontology, as well as giving more refined query results. This is shown in Figure 2. Our object-properties were also set to be quite general being able to connect many of the given subclasses and instances in the ontology. The object-property “hasParticipant” could for example be used in many different settings, as one can see in the ontology. We also added the inverse property to most our object properties. For instance, “hasParticipant” has the inverse property “isParticipant”. This way we could query the other way around and retrieve results about what event an individual has participated in. We could have added more specific object properties for each article, but since this is a protoptype we chose not to do so, and rather keep it more general.

Figure 2 Object properties

Some properties we found particularly useful and worth mentioning are the “hasSubEvent” and “hasSuperEvent” properties. All the instances in the ontology are specific events. Instead of adding several extra sub-classes to specify multiple and often quite intricate connections, we used “hasSubEvent” and “hasSuperEvent” to link individuals that intersected with each other. For example, “Holocaust” was a tragical event, and we therefore put it under “TragicalEvent”. Still, “Holocaust” was also a part of world war two and subsequently a part of the “WarEvent”. We could for example insert “Holocaust” in the class “WorldWarEvent” to illustrate that it is also a part of World War 2. But in our eyes, strictly speaking the class “WarEvent” should only include the *wars* themselves, and not the events that happened during them. We linked the properties together to declare that, even though they are not part of the same class, they are still related. This implementation is shown in Figure 3 and Figure 4.

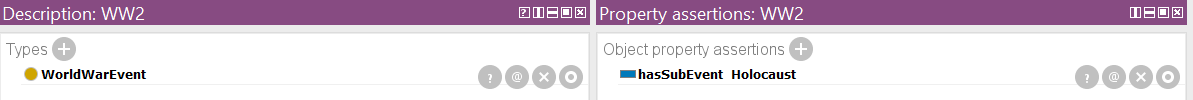
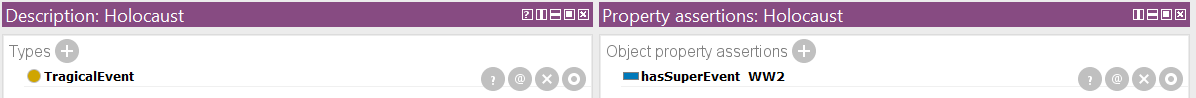


Figure 3 Holocaust event

Figure 4 World War 2 event

### Data properties

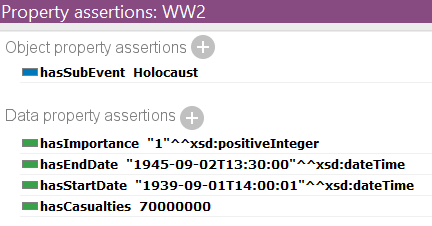
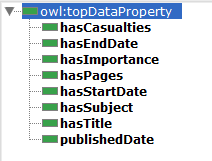
In addition to the object properties we also made data properties which give details about the individuals in our ontology. The difference between data properties and object properties is that a data property is linked to a given value, whilst an object property refers to another element in the ontology and creates a connection between them. In Figure 5 is a picture of our data properties and in Figure 6 is an example of a given value for “hasImportance”, “hasEndDate”, “hasStartDate” and “hasCasualties” for the Holocaust event.

Figure 5 Data properties

Figure 6 Properties of World War 2 event

One data property that might stand out more than the others is the “hasImportance” property. This is a property which gives a value to a historical event to illustrate its importance in comparison to others. These values are based upon the article “The Most Important Events of the Century From the Viewpoint of the People”. Though these cannot be stated as facts or empirical truths, they do give a good pinpoint on importance of different events. As well as it gives the ontology more preciseness. We also created an annotation property “description” to give short descriptions about individuals in the ontology.

### Limitations of the ontology

Because we wanted the ontology to be general but at the same time as specific as possible, we have chosen not to implement too many specific details. Our focus when building the ontology was to build a framework one could work with and implement details later. With the case of historical events, there are countless details and information one could bring into the ontology. In Figure 7 is an illustration of how it could be done in the case of the historical event of the ship Titanic and the article “Passengers on the titanic”:

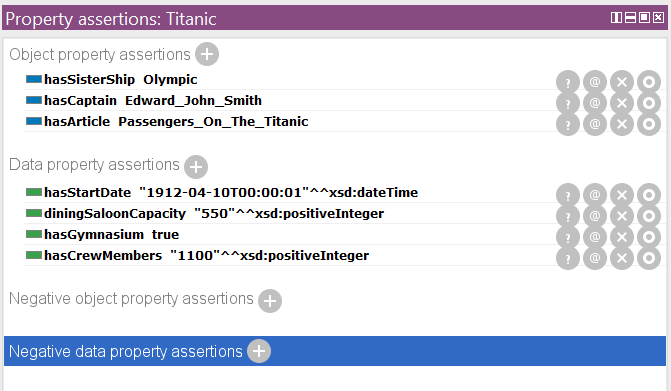


Figure 7 Property assertions for Titanic

We are also aware of the shortcoming of our ontology due to our biasedness. As Norwegians, we view certain types of events as important whilst other nations might not view them as important. We have tried to limit this biasedness to create an “international” ontology that would be applicable universally by viewing events from a global perspective. The list of the most important events has helped us in our attempt to reach this perspective.

# SPARQL

The purpose of the ontology is to enable users to search information about the different events. We have therefore created 6 questions that we will translate to a SPARQL-query. We should then find the answers through the query.

While finishing our ontology, we sat down to discuss possible competency questions. We discussed the different types of questions we could ask, and what output we wanted to obtain from the ontology. By creating a relatively wide ontology, we could now answer a wide array of questions. As our ontology is based on articles and historical events, we wanted to cover questions that could both give us information about the articles and the events. We wanted to create realistic questions that a user might find useful.

**Case 1** - **What are all the events that are related to space technology that occurred in the 60’s decade?**

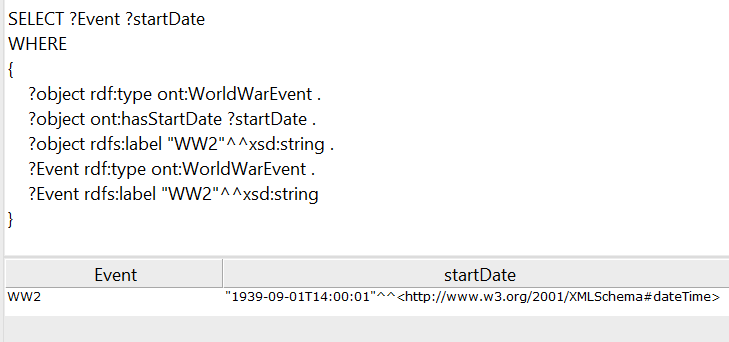
Our first query is about the space technology. The user wants to know all space events that happened between 1960 and 1970. Using SPARQL, we return the dates on which an event happened and its name. FILTER is used here to only retrieve the events that happened in the 1960’s. The query and result is shown below in Case 1.



Case 1 Space technology

**Case 2** - **When did World War 2 start?**

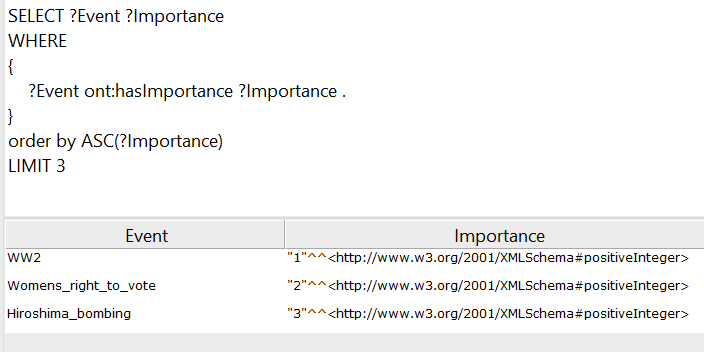
Our second query finds the WW2 event and returns its start date. This query shows that it is possible to extract specific information about a specific event. However, due to the relatively tedious query, it does not seem that SPARQL is ideal for such tasks. SPARQL is more useful when trying to compile a list of objects with specific criteria. The query and result is shown below in Case 2.



Case 2 World War 2 start date

**Case 3 -** **What are the 3 most important events of the XXth century?**

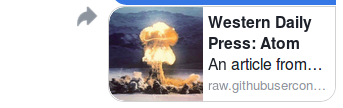
In the query, we return the 3 most important events of the XXth century from the perspective of the American population. We added the importance as a data property to mentioned events according to the article “The Most Important Events of the Century From the Viewpoint of the People”. We use the function “LIMIT” and “order by” to only get the top 3 events according to their importance. The query and result is shown below in Case 3.



Case 3 Three most important events of the XXth century

**Case 4 - What are all the articles related to the Titanic event sorted by publishing date?**

In this query, we return all the articles that are linked to the theme of the Titanic. To this end, we use the object property we created called “depicts”. This links the event of the Titanic to all articles that depicts it. Two articles are returned. The first is from April 10th 1912, just a few days before the sinking of the Titanic, and describes amongst other things what the Titanic is carrying. The other article is from 1997, and describes the musical production in New York from the same year. The query and result is shown below in Case 4.

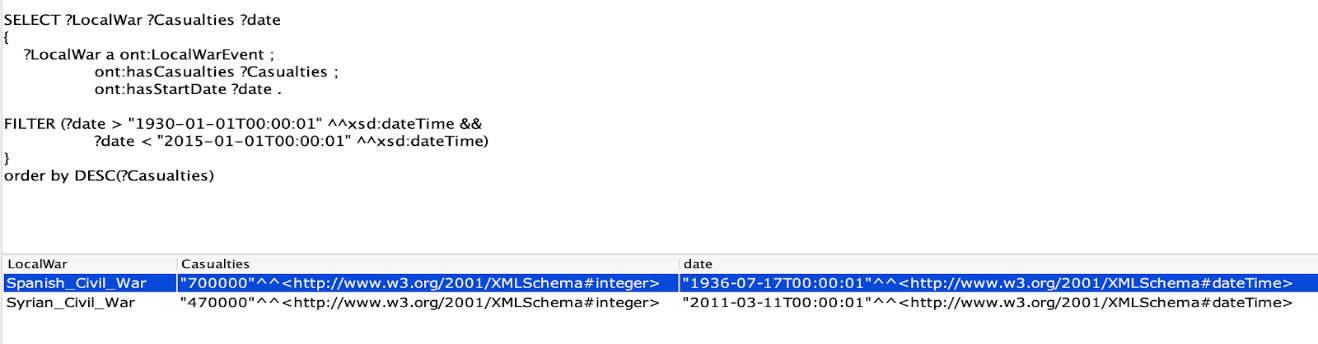




Case 4 Articles related to the Titanic

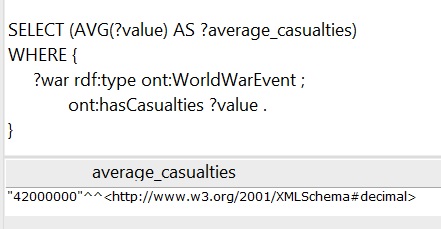
**Case 5 - What local wars have there been between 1930 and 2015 and how many casualties were there in each?**

In this query, we wish to list all the wars in the subclass “LocalWarEvent” and list them by descending casualties. To this end we use the function “FILTER” that allows us to limit the search to a specific timeframe and “order by” to sort the results in a descending order. The query and result is shown below in Case 5.



Case 5 Local wars and corresponding casualties 1930-2015

**Case 6 -**  **What is the average number of casualties of all wars?**



In this query, we wished to find all the wars present in our ontology. To this end we used the “AVG” function that calculates the average value of all casualties. This is an efficient function that can be very useful. The query and result is shown below in Case 6.

Case 6 Average casualties all wars

# Annotation

### Facebook OGP

Facebook OGP allows Facebook to create a little preview of any link that is shared on Facebook. Facebook will look for the meta property tag “og:” and use each instance of the tag to create the preview. Above we can see what our annotated web page looks like when it’s being shared on Facebook, and below can see what the actual HTML looks like.

The picture we use is not from the article itself, but it is selected as a placeholder because the web page didn’t contain any meaningful jpgs that Facebook require.



Figure 8 Metadata for a news article in “raw format”

In Figure 9 we can see the tags as Facebook would see them, using OGP check.

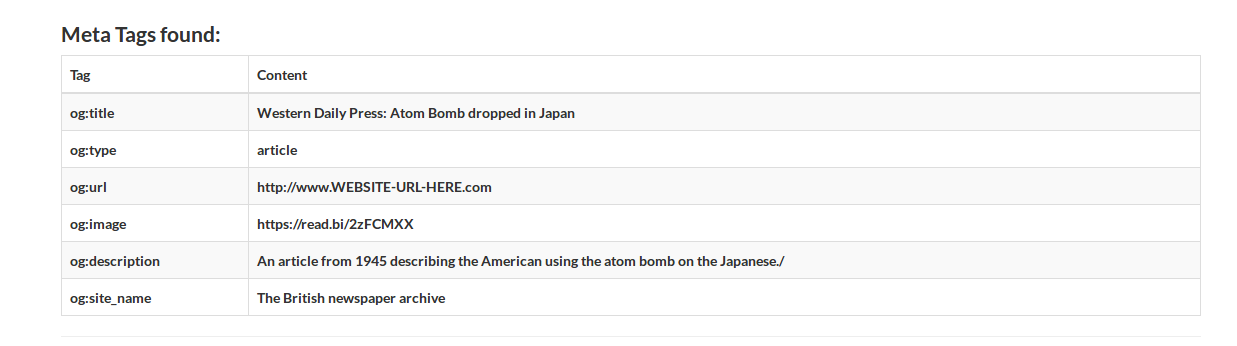


Figure 9 Metadata for a news article shown with OGP check

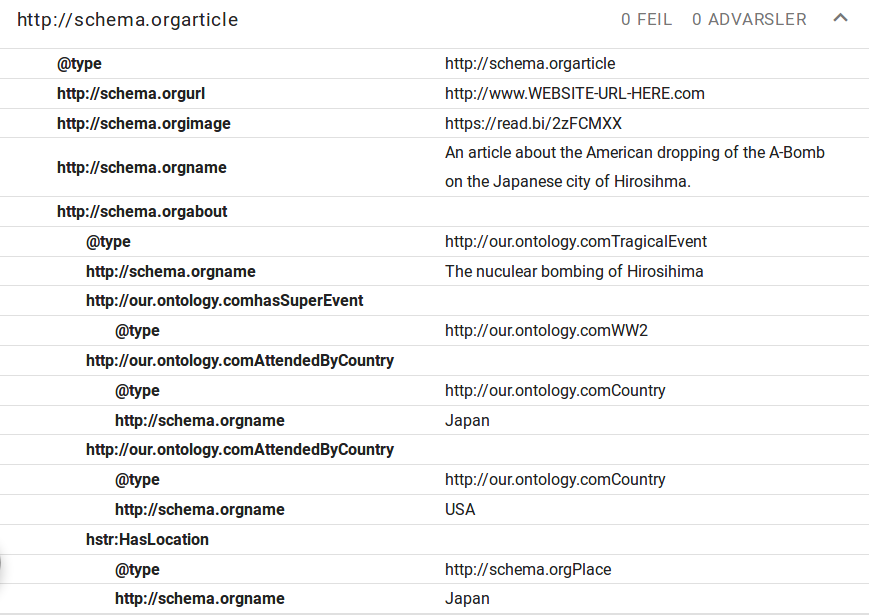
### JSON-LD

Annotating in JSON-LD was challenging because we weren’t sure if we should describe the web page as an article, or as the historical event it is describing. We eventually decided to use the JSON-LD to describe the article, and then fill in the relevant historical data in the “about tag”. This is shown above in Figure 10.

Figure 10 JSON-LD annotation

We gave our ontology a faux URL, with the prefix “hstry”. This prefix will be used later with the RDFa lite annotation as well.

Illustration on the next page shows what the Googles test tool sees:

As we can see the page is annotated as an article, and the article is annotated to be about the “TragicalEvent” Hiroshima bombing, and relevant historical information is added under “sch:about”. From this it is easy to see that the article is about a tragic event that happened in Japan, where Japan and USA were the two participating parties.

### RDFa Lite

This was the hardest annotation to do, partly because the syntax was slightly more complicated, but mostly because it was difficult to decide what to annotate and in what way. The web page we are working on is mostly filled with irrelevant non-historical data, with small nuggets of text about a very important historical event. But how should we describe this with RDFa?

We decided to do the markup in order to make it easy to understand *what* the article was talking about. It is not immediately apparent from the word “Chinese” that what we are talking about is the nation of China, not the language, and that “the race of discovery” in this context refers to the race to be the first to create the atom bomb.

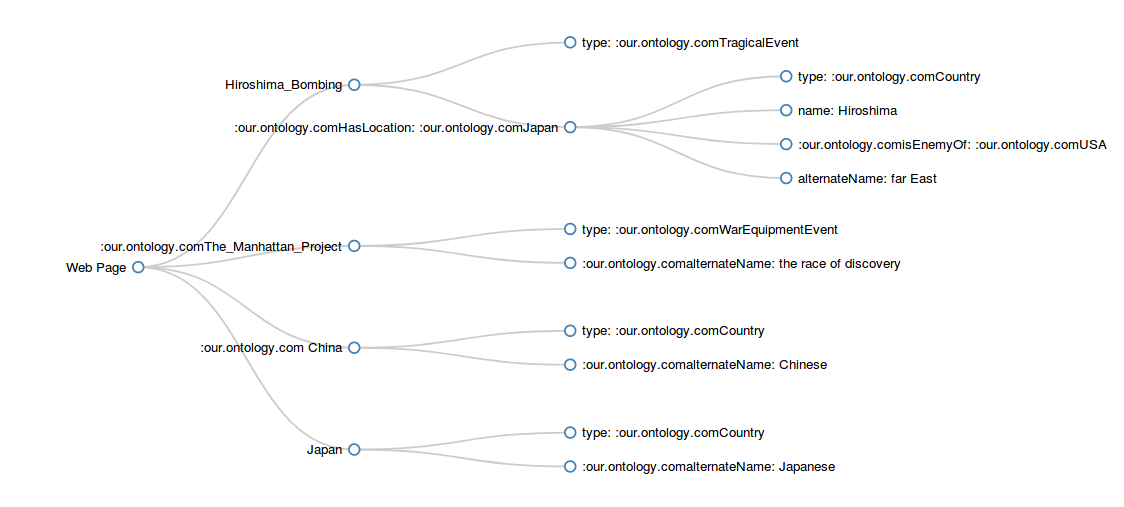
We looked for certain keywords in the text that give meaningful information on what the article is about. For example, in the first sentence of the article the context is set. The string “first atom bomb” is annotated as a TragicalEvent, with location Hiroshima which for this purpose is regarded as the same as the country Japan. This makes it easy to find this article in a search for “Hiroshima bombing” on the web.

Figure 12 Visualization of RDFa Lite annotation

In figure 12, the RDFa is visualized and we can see that the markup works the way that we want it to. The article describes the bombing of Hiroshima, and this information is located in the top node in figure 12. Further we see that the Hiroshima bombing is an instance of a TragicalEvent, and that the location of the event is the country Japan, and the name of the location is Hiroshima. An alternate name is set as “far east” because the article refers to Japan as “the far east” later in the article. It also picks up that Japan is, or at least was in WW2, the enemy of USA.

The graph in figure 12 is created with the tool <https://rdfa.info/play/> that creates visual representation of RDFa data.

# Applications

# One of the main reasons to annotate our website with metadata is to make it easier to find in a web search. Using JSON-LD to tag the page as an “article” that is “about” the “Hiroshima bombing” makes it easy to find this article in a search for articles about the Hiroshima bombing. It also allows search engines to create little preview, much like Facebook does with OGP. We can also use RDFa Lite to add properties like dates and participants, which makes it possible to find this article in an even narrower search, for example if we search for articles about tragic events with Japan and USA as participants during WW2. One great upside of using metadata on our websites is that we don’t have to use natural language processing (NLP) to extract meaningful data. NLP is processing intensive, while extracting the tags and the returning the websites with matching tags is much less work for the computer. This means we can search through a lot more information and still get as, if not more, accurate results. In addition, it removes ambiguity. A great part of our language understanding comes from context, and in certain languages more than others. This context is easy for us humans to understand, but very difficult for a computer. With RDFa and JSON-LD we can remove this ambiguity, by linking to a certain class or object in an ontology, so that when the computer reads “the Japanese” it understands that its talking about the collective Japanese people as a nation, and not, for example, a bar called “The Japanese”. This is obvious for humans but might not be obvious for a computer program trying to understand natural language.

# The downside of using metadata is that it takes quite a bit of work for the developer. It is important to use meaningful RDFa and JSON-LD so that we can extract the information we want and annotating a website can take a lot of valuable developer time to get right. Especially now that websites are very dynamic and new original content is being created all the time, content creators might not consider it important enough to annotate their websites.

# Group Dynamic

Each member had a specific area of the assignment he/she was responsible for. Two were assigned to creating the ontology, one was responsible for the SPARQL queries and one was responsible for the annotation. Due to this role division, everyone had to be updated on the ontology in order to fulfill their tasks. This allowed for great communication amongst group members. We met on a weekly basis and we discussed the different tasks. Those meetings were very helpful to get different perspectives on the issues we met along the way. In the end, we created an ontology that we deem useful.

The SPARQL queries were challenging to write. This was because none of the group members had previous experience with SPARQL or SQL. In addition, the web resources are more limited for the SPARQL language than for other more mainstream coding languages. We therefore spent a lot of time on these queries and received some help from the labs. We are satisfied with the queries we ended up with.

The annotation process was also challenging. However, with the help of web resources, the labs with the TA and, we managed to complete annotations we are satisfied with.

# Conclusion

Throughout our work with the assignment we have understood the powerful applications of semantics. We have become well acquainted with the difficulty of creating an all-encompassing ontology and the strenuousness of the choices to be made. We have also gained proficiency in the Protégé program that has enabled the ontology-building and the SPARQL querying. We have understood the strengths, and the shortcomings of Protégé. Furthermore, we have learned how to annotate web pages using OGP, RDFa Lite and JSON-LD. This gave us an understanding of how semantics can allow websites to become more machine readable and allow for enormous amounts of data to be collected efficiently.

All in all, this was a successful group effort and we learned a lot about the potential of semantics. It was motivating for us to apply the theoretical knowledge we received in lectures to a real-world case.

# Bibliography

Ide, N., & Woolner, D. (2007). Historical Ontologies. I B. C. Ahmad K., *Words and Intelligence II* (Vol. 36). Springer, Dordrecht.

Sheth, A., & Thirunarayan, K. (2013). *Semantics Empowered Web 3.0: Managing Enterprise, Social, Sensor, and Cloud-based Data and Services for Advanced Applications.* Morgan & Claypool.

|  |  |
| --- | --- |
| present an overview of what has been achieved | x |
| What work did each group member contribute? |  |
| Why was the ontology constructed in the way it was? | x |
| What kinds of questions can be answered by the ontology (the competency questions you used)? | x |
| What can the web site do with the added semantics (e.g. third party applications)? | … |
| The report should also include examples of the expected rich snippet from the markup (from the rich snippet tool). | x |
| The report should be no longer than 10 pages, including figures, example markup, queries, etc. | x |
| The deliverables should be bundled into a zip file and submitted. | x |