Data- and Systems Paper: Museum Exhibitions in the Netherlands

R.A.M. Blankendaal and A.G. Lo

Vrije Universiteit Amsterdam

Abstract. Exhibitions and galleries can be found all over the Netherlands. Nowadays this is announced with the use of flyers, on social media and stated on the website of the museums. With the use of linking different data sets, this project describes an application made for visitors to find current exhibitions and galleries in the Netherlands based on their specific art interests and demands.

1 Introduction

The Netherlands counts numerous exhibitions and galleries that are presented to the public every year. Some of these collections are temporary and are being shared between countries. Others are permanent, and are visited by numerous visitors coming from different countries on a daily basis. The way art is experienced by visitors depends on the representation and on the way in which museums interact with their visitors [5]. Tourists often spend a short time in the Netherlands, and have to split their time over different sight seeings.

This is why it is thus important for them to be able to find an exhibition that meets their wishes and interests, without having to spend a lot of time browsing through different sites and brochures. Providing an overview of the existing exhibitions as well as indicating what they contain, could also help people in finding new, less popular exhibitions that are often overshadowed by the permanent collections displayed in large museums. In this research we aimed to solve this problem, by merging different data sets and querying over them.

First, the use case with the use of an interaction flow-diagram is discussed. This leads to two scenario's that are explained. In section 3, we address the necessary data for the application, the use of these data is then described in section 4, including made design choices. Next, semantic considerations including RDF and RDFS entailment is discussed. Finally, the discussion describes a first evaluation about the idea of the application.

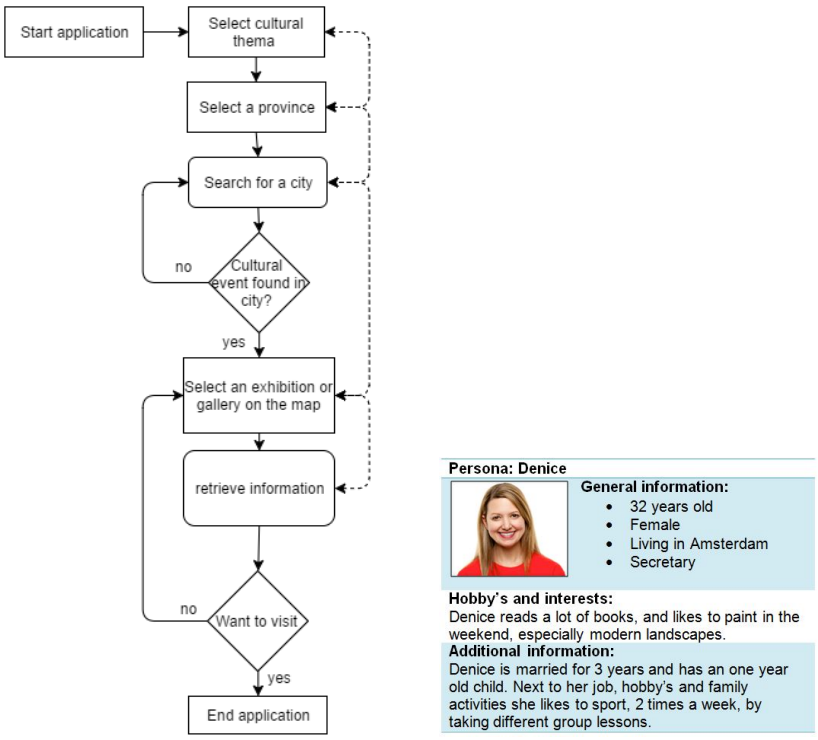
2 Scenario / Use Case

Based on the introduction above, a more detailed scenario is described. As aforementioned, interaction is an important aspect in contemplating and experiencing art. Being with a variety of people in a room that offer different perspectives

on art, is the added value of visiting physical exhibitions. However, as a visitor it is hard to find current exhibitions and galleries. A few Google searches point out that most exhibitions and galleries are indicated on the website of the museum itself or by corresponding organisations or sponsors. Additionally, posters, folders and articles in newspapers are often used for advertising.

However, there is no global overview of all the current exhibitions or galleries that are presented in the Netherlands. As use case, we describe, with use of available data, how a website or application can improve or spread the information about where current exhibitions and galleries take place, what kind of cultural exhibitions these are and what they represent.

The interaction flow-diagram is the left image represented in figure 1. Based on the diagram, we will discuss two use cases. We introduce a persona called Denice, who is introduced in the left image in figure 1.



(a) Interaction flow-diagram (b) Persona

Fig. 1: Scenario

2.1 Scenario 1

As explained above, the data is used to make a system which enables future visitors to find a exhibition or gallery in a particular city matching their interests. The persona used in this scenario, Denice, lives in Amsterdam and has an interest in modern landscapes. She want to see if their are any interesting exhibitions in Amsterdam. In order to use our application, she will first go to the website, and selects the modern art theme. Subsequently, she selects the province of Noord-Holland (a Dutch province) on the map and the city Amsterdam. Fortunately, there are multiple exhibitions containing modern art in her city of preference. Denice then selects the specific exhibition she wants to visit.

2.2 Scenario 2

Another scenario is that Denice wants to visit an exhibition together with her friend, who is living in Rotterdam. They differ in interests, Denice, as aforementioned, prefers modern art, and her friend likes art from the Renaissance. In order to minimize to overall travel time, they decide to select cities between Amsterdam and Rotterdam. Unfortunate, there is at this moment no exhibition or gallery which represents both themes. So, they made a compromise by going to two exhibitions during the same day in order to meet both their interests and wishes. As a result, they use the application multiple times, entering different cities, to see where the exhibitions are located to minimize travel time.

3 Data sources

In order to create an application that represents the current exhibitions and galleries, data needs to be collected. The two datasets [4, 9] that were used in this project were pulled from a website on which open datasets from Amsterdam have been published. The datasets are .csv files containing lots of information, we used the following data: a specific id for an exhibition or gallery called *Trcid*, the exhibition *title*, a *short description*, *longdescription*, a *media* image and a *thumbnail*. The data files also contain geographical information: the *city* of the exhibition or gallery with the *location name* of the building, the *latitude* and *longitude*, *zip code*, *address* and an *url* of the website.

In order to use these data sets about Dutch museums for our application, they first have to be converted to the .rdf format. In this way, it can be used as Linked Data. The two data sets used share the same column names. On the one hand this means that the conversion of the CSV files to RDF format can simply use one and the same Python script. On the other hand, it makes the data that is merged less varied than it would have been if two completely different data sets would have been used.

In order to incorporate and provide all the functions of the application, the following data sets could be added in the future:

- De data sets used only provide a description of the exhibition or galley in general. However, several museums have open data about their collection. This data could be incorporated to extend the information by adding data of the art objects presented. For example, three Dutch museums provide data sets about costume exhibitions [6]. These are XML files which have to be converted into RDF.
- The used data file also contains some information in English (the descriptions). To make the application accessible for a wider public, an option to switch to English can be included.
- Data should be collected about the dates on which an exhibition or gallery is presented. To extend that, data about the visiting hours could be included.

4 Building Knowledge Graphs

As aforementioned, the two data sets share the same column names and amount of columns. In order to execute the conversion successfully, these had to be divided into resources, literals, and properties. The predicates describe the relations that connect the different components that make the graph. This graph is represented in figure 2.

There exist many online tools that claim to enable CSV to RDF file conversion using a script. However, these tools are often insufficiently specific to recognize the ontological relationships that define the graph behind the data sets. This is why the CSV2RDF-tutorial [10] was used, which enabled us to customize an already available Python script to match the columns of the data sets and defined prefixes. The RDF schema describing all these different relations between subjects and objects was made using the ontology designing program Protégé [7].

As can be seen in the graph, different RDF prefixes are used. The main goal was to reuse as many as possible prefixes in order to connect to already available Semantic Data on the web. *RDFS* is used to define the subclasses and *rdf* to provide general links. *dpb* represents DBpedia [2] properties. We have used the *geo* prefix to emphasize the geographical data, such as longitude and latitude. *aat* refers to Art and Architecture Thesaurus [1] and *dc* refers to the Dublin Core Metadata Terms [3]. Finally, the prefix *co* is specially defined for the use of this RDF schema and stands for cultural ontology.Finally, the RDF files are placed into a tripplestore, Stardog [8], which gives the posibility to query over the data.

During the conversion of the data sets to RDF, some problems were encountered. The conversion uses utf-8 encoding by default. When working with cells containing a lot of information, as is the case for instance with the descriptions, an encoding error appeared, since this text is in Latin. In order to solve this error, latin8 decoding should thus be used.

Besides running into these technical issues, some design choices were made. Primarily, the CSV files contain more columns with information, but since many of them were empty or irrelevant for the application, we decided to not include

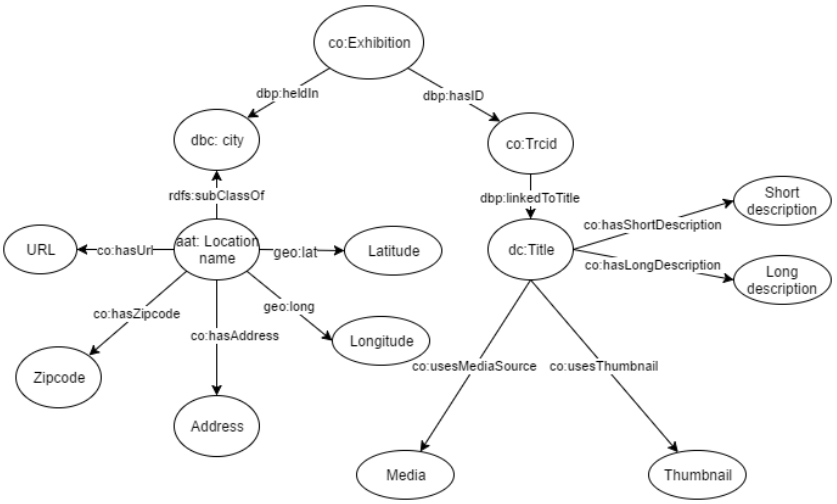


Fig. 2: RDFS

them in the conversion. Secondly, the data files are in Dutch, whereas some of the column names are in English. In order to make the data more reusable, we translated the Dutch data columns into English names in order to have more coherent data.

5 Semantic considerations

Entailment can be explained as logic reasoning based on the RDF graph. Some expressions and relations that may seem logical to humans, might be less obvious or simply illogical for machines. In figure 2 the used RDF scheme is shown, and the sub-graph in figure 3. This sub-graph shows that every exhibition takes place in a city. Each city consists of location names, which are the names of for instance public places and buildings.

An example of entailment given the data could be the following: Location name and exhibition (or gallery) are related. This is not explicitly stated in the graph, but since an exhibition (or gallery) is related to (held in a) city, and location name is a subclass of city, there is thus an implicit relation between the two classes. This means that all the different literals that are now linked to the location name, are also applicable to the exhibition (or gallery).

Using entailment in this example adds to our application in that it helps to filter according to the interests and demands of the user. It allows the user to skip the step of filtering on city, and instead gives him/her the opportunity to search on all the different literals that are now attached to city, but are thus implicitly connected to the variety of exhibitions (or galleries) as well. This also means that different combinations can be made to filter out those exhibitions (or galleries) that the user might want to visit.

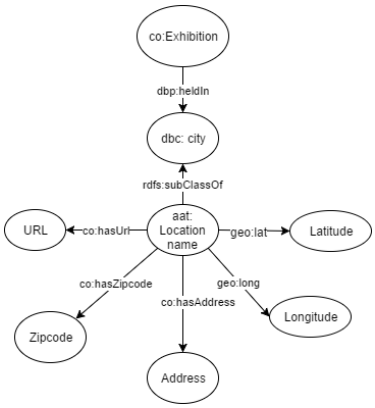


Fig. 3: Sub-graph

5.1 RDF and RDFS entailment

Figure 4 shows the entailed graph based on the graph discussed above. The dotted lines represent the entailed relations. As can be seen in the graph, more entailed relations could be added. All the different properties that are linked to the location name class are thus related to the exhibition (or gallery).

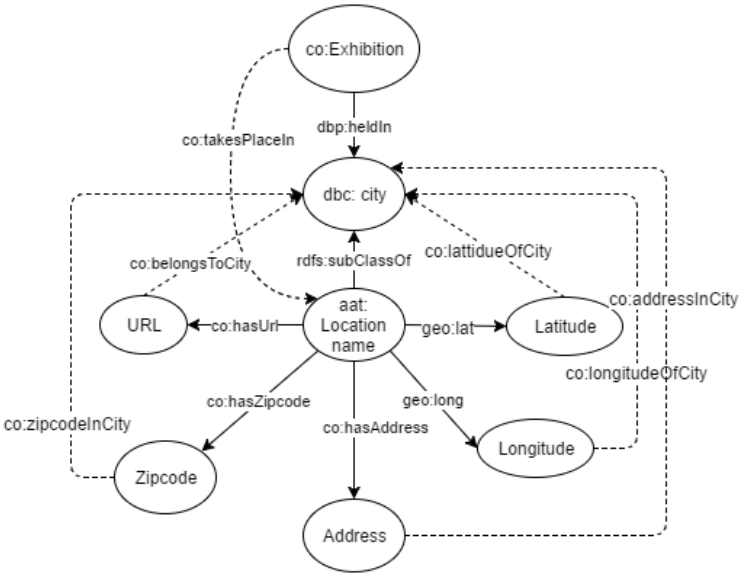


Fig. 4: Entailed sub-graph

5.2 SPARQL

The RDF query language SPARQL was used to query over the data. Using the FILTER regex method allowed us to filter out specific words from texts in the database. This made it possible to extract those exhibitions and galleries that are for instance given in a specific city, as shown in the query below.

```
SELECT DISTINCT ?exhibitions WHERE {
  ?exhibitions dbp:heldIn ?cities .
  FILTER regex(str(?cities), "amsterdam", "i")
}
```

The same method can also be used on the descriptions that are linked to exhibitions and galleries. If the user of the art application is interested in a specific art topic or period, using this filtering method allows to show merely the exhibitions and galleries that use this word in their description. The query below shows how this can be done on short descriptions for a user who is interested in modern art.

```
SELECT DISTINCT ?exhibition_name ?shortdescription WHERE {
  ?exhibitions dbp:heldIn ?cities .
  ?exhibition_name co:hasShortDescription ?shortdescription .
  FILTER regex(?shortdescription, "modern", "i")
}
```

In the same way, we could query and filter on those exhibitions and galleries that are located in a specific zip code range or that share a longitude or latitude.

If we want to query the entailment example that was described above, we use the following SPARQL query.

```
SELECT DISTINCT ?exhibitions ?locationname WHERE {
  ?exhibitions dbp:heldIn ?cities .
  ?locationname rdfs:subClassOf ?cities .
}
```

The reasoning button in the Stardog query panel enables to let the system reason over the queried data in the data sets. We expect the data results to be different using reasoning than when reasoning is turned off. This is the case, but the results are obtained when reasoning is turned on are formed based on the idea that the classes location name and exhibition are linked through the class city. What happens is that if we take the city of Amsterdam, it now reasons that all of the exhibitions that are located in Amsterdam, are automatically located in the specific location name of the museum location as well. This is of course not the case, and we consider the fine tuning of our graph future work.

6 Discussion

In conclusion, museum data is gathered and conversed to RDF. This is stored in a triple store which provides the possibility to query over the data. Figure 5 shows screenshots of a mock-up that can be built.

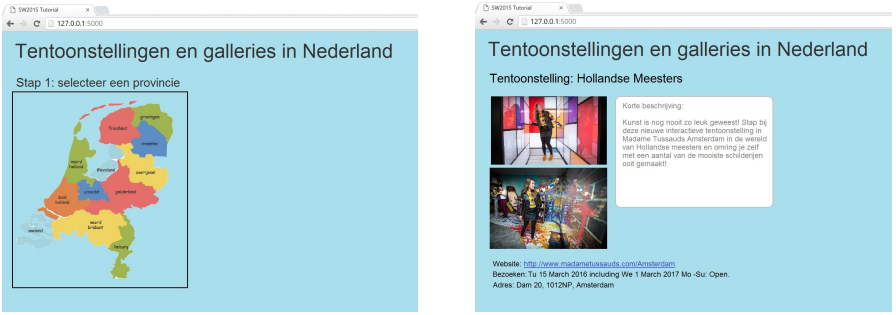


Fig. 5: The mock-up

During this project, we have learned where to find open, accessible data online and how to convert this data to the RDF format. Also, we have used SPARQL in order to extract the specific parts of data that we need given a specific goal.

There are many points for future work. First, the title is now used twice, as exhibition and simply as title. This is done based on the idea that this is logic. However, we do not want to use data twice under a different name, therefore this should be adjusted in the next version by having the Trcid as main class representing the unique ID of the exhibition or gallery. Second, more data should be incorporated to be able to provide more information and system possibilities to the user of the application.

This paper has explored the possibilities on how to use museum data in order to help future visitors by giving more and relevant information on current exhibitions and galleries. Future work based on the work described in this paper will demonstrate the extent to which this data can be reused in an application that assists visitors and tourists in finding the exhibition or gallery of their preference.

References

1. The Getty Research Institute Art and Architecture Thesaurus. http://www.getty.edu/research/tools/vocabularies/aat/. Accessed: 2016-04-02.
2. Dbpedia. Towards a public data infrastructure for a large, multilingual, semantic knowledge graph. http://wiki.dbpedia.org/. Accessed: 2016-04-02.
3. Dublin Core Metadata Initiative. http://dublincore.org/documents/dcmi-terms/. Accessed: 2016-04-02.
4. Gemeente Amsterdam Musea en galleries. http://data.amsterdam.nl/dataset/musea-en-galleries. Accessed: 2016-03-29.
5. Christian Heath and Dirk Vom Lehn. Interactivity and collaboration: new forms of participation in museums, galleries and science centres. *Museums in a digital age*, pages 266–280, 2010.
6. Open Cultuur Data. http://www.opencultuurdata.nl/datasets/. Accessed: 2016-03-29.
7. Protégé. http://protege.stanford.edu/. Accessed: 2016-04-02.
8. Stardog. http://stardog.com//. Accessed: 2016-04-04.
9. Gemeente Amsterdam Tentoonstellingen. http://data.amsterdam.nl/dataset/tentoonstellingen. Accessed: 2016-03-29.
10. Github: KRontheWeb CSV to RDF tutorial. https://github.com/KRontheWeb/csv2rdf-tutorial. Accessed: 2016-04-31.