

Towards a genealogical ontology for the Semantic Web

Ivo Zandhuis*

Genealogy is an interesting domain for demonstrating the possibilities of the Semantic Web. As a first step towards such a demonstrator this paper introduces an ontology for the genealogical domain. First the existing standards are investigated and a short introduction into the Semantic Web is presented. After that two ontologies are introduced. The first ontology models the capturing of personal information from an (archival) resource. The second ontology models the personal information itself and can be used to capture genealogical information. The most important aspects of the ontologies introduced are (1) the possibility to model the relationship with the source from which the information is obtained, (2) the possibility to model the relationship with the agent responsible for the information and (3) the way from nomenclature and time-representation in various cultures.

1 Introduction

The World Wide Web is intensively used for genealogical research. People from all over the world investigate their ancestry by searching databases and publish their genealogy on a home-made web site. The information on these platforms however is not structured to facilitate intelligent cross-platform searching.

Family constructions are often used to introduce computing scientists into the field of Artificial Intelligence. The Semantic Web combines the insights in Artificial Intelligence with the possibilities of internet. The automatic construction of genealogies is therefore an obvious demonstrator of the Semantic Web.

This paper is the first step towards this demonstrator discussing a standard for exchanging genealogical data in the form of an *ontology*.

1.1 Goals

A standard for exchanging genealogical data could be used in various situations. The data can be used for

finding a person in genealogical research or primary sources published on the Web. With enough data complete genealogies could be derived automatically. When complete reconstructions of populations have been derived, new demographic questions can emerge and be answered.

1.2 Requirements

To reach the goals mentioned above, the standard should have the following properties. Firstly, users should get correct data only. In a system based on the ontology, various integrity checks must be possible, like the check that a person has died after he has been born, or that he can not be over 120 years old. Secondly, the relations between the genealogical data and sources that prove the assertions must be stored, as well as the agent responsible for publishing the assertions. Finally, the ontology must facilitate extensibility for different cultural approaches to persons names and dates and research questions in the future.

*(ivo@zandhuis.nl) Ivo Zandhuis Research & Consultancy

1.3 The existing standards

This paper is not the first initiative in developing a technology for exchanging genealogical data. The most important existing standard in this field is GEDCOM [1]. The latest official version of GEDCOM dates back from 1995 and facilitates migration of data from one system to another.

Another important development in the field of genealogical data is the datamodel developed and published by GENTECH [2]. The datamodel functions as a reference model for programmers of genealogical software. GEDCOM correctly encodes genealogical information. However the technical implementation is outdated, the reference to a source is done with a single string and its oriented on the situation in the US. Like in GEDCOM and GENTECH, the ontology introduced here uses events to model information about birth, death and marriage.

2 The Semantic Web

2.1 Introducing the Semantic Web

This section gives a short introduction into the Semantic Web. A more extensive introduction can be found in [3]. The information on the Semantic Web is described in formal languages of different types and levels of expression: Resource Description Framework (RDF)/ RDF Schema [4] and the Web Ontology Language (OWL) [5]. Basis of all these languages is the concept of the *triple*. Take for example the sentence 'Joe has a mother called Sue.' On the Semantic Web this relation is modelled as the triple:

```
('Joe', hasMother, 'Sue')
```

The three parts of a triple are called Subject ('Joe'), Property ('hasMother') and Object ('Sue'). Subjects and Objects are instances of a certain Class. Classes, Properties, Subclasses can be defined in a similar way as in Object Oriented Programming. The declaration of all the Classes, Properties and their relations is called an *ontology*. More precisely: Guarino [6] states that an ontology refers to an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words. The vocabulary is defined in terms of Classes and Properties. For every domain an ontology can be developed; this paper introduces an ontology for the genealogical

domain. The ontology and data based on the scheme can be distributed to agents (human or automatic) in XML.

2.2 Why the Semantic Web?

Now why should we use SemanticWeb technology, when GEDCOM is both a correct and widely used standard for exchanging genealogical information?

The Semantic Web languages make extensions possible with concepts needed in all types of future research. Without changing the existing ontologies, researchers can add the extensions themselves. The second reason is that the Semantic Web adds the possibility to use standardized, existing SemanticWeb tools for reasoning about personal information and family relations. Semantic Web tools are built to derive information encoded in an language, i.e. a language not specially developed for genealogical information. Therefore more people are developing tools and better tools are available.

Finally the Semantic Web standards are developed solving known problems in Computing Science. So Computing Scientists discuss the typical problems of formal languages, computability and distribution. The users of the Semantic Web (for instance in the genealogical domain) can use the technology without worrying about these problems, knowing they are dealt with.

3 Modelling information

To reach our goals two ontologies are developed: *genont* and *srcont*. The first ontology models the personal information, the second ontology models the source with the personal information it contains. In actual practice three files are stored: one with the definition of the concepts for personal information in the *genont*-ontology, one with the definition of the concepts for resources in the *srcont* ontology and a datafile which contains the actual data. The actual data could be either the transcription of a genealogical resource or a digital publication of genealogical research.

3.1 Persons: *genont*

The *genont* ontology models personal information of interest for genealogical research. There are two types of persons: female and male. Furthermore a person has relations with names, events (birth, marriage, death), other persons (family-relations) and other attributes (occupations and addresses).

3.2 Sources: *srcont*

The *srcont* ontology models the reference to an original source in a repository, such as an archival item or a publication. To archival material is referred by stating the repository, name of the archive, a description of the referred item with a number, a title and a date. A publication is described with author, title, imprint and pages. Besides the reference to the source the containing personal information is captured. This is done by using the constructions of the *genont* ontology. The *genont* ontology is therefore imported into the *srcont*.

3.3 Datafile containing information of a person in a source

For capturing the personal information stored in the resource, a datafile is constructed based on the *srcont* ontology, extended with the *genont* ontology.

3.4 Datafile containing genealogical research

A datafile based on the *genont* ontology can be used to publish the result of genealogical research. In that case only the *genont* constructions are used. The relations with the sources can be maintained by stating that the persons mentioned in a datafile based on *srcont* (extended with *genont*) and those mentioned in the datafile based on *genont* are equivalent. This can be done with the standard OWL-constructor `equivalentClass`.

4 The ontologies

In this section the actual ontologies are defined. The OWL Syntax is used [5], which is easier to read than the XML-syntax.

4.1 Responsible agent

There are standard-language constructions for capturing the agent (human, institution, software agent) responsible for publishing the information. This could either be the agent that constructed a genealogy (in a datafile based on *genont*) or the agent that made the transcriptions of a source available (in a datafile based on *srcont*).

```
AnnotationProperty(<http://purl.org/dc/elements/1.1/creator>)
AnnotationProperty(<http://purl.org/dc/elements/1.1/date>)
AnnotationProperty(<http://purl.org/dc/elements/1.1/publisher>)
```

4.2 Description of the classes and its relations

4.2.1 Source (*srcont*)

The *srcont* ontology defines the concept *item* which has a relation with the description of the resource (repository, name of the archive etc.) and has a relation with the *content* of the resource. This content is stored according to the model defined in the *genont* ontology.

```
Class(item partial
      restriction(srcref cardinality(1))
ObjectProperty(srcref
      domain(item)
      range(archref))
ObjectProperty(content
      domain(item)
      range(person))
```

The reference to an archival item is made with four components: the repository, the title of the archive, the description of the archival unit and the date of this unit.

```
Class(ref partial)
Class(archref partial ref)
DatatypeProperty(repository
      domain(archref)
      range(xsd:string))
DatatypeProperty(archtitle
      domain(archref)
      range(xsd:string))
DatatypeProperty(desc
      domain(archref)
      range(xsd:string))
DatatypeProperty(unitdate
      domain(archref)
      range(xsd:string))
```

4.2.2 Person (*genont*)

The Class *person* is the basic-entity in the *genont* ontology. All other entities described below relate to this Class. The following declaration states that a *person* has a name, is born, and by definition has exactly one (biological) father and exactly one (biological) mother. The ontology distinguishes two subclasses of a person: *male* and *female*. Situations where the sex is unknown, the *person*-class itself can be used.

```

Class(person partial
  restriction(hasName cardinality(1))
  restriction(isBorn cardinality(1))
  restriction(hasFather cardinality(1))
  restriction(hasMother cardinality(1)))
Class(male partial person)
Class(female partial person)
DisjointClasses(female male)

```

4.2.3 Name (genont)

The main relation a *person* has, is the relation with a name. Various cultures have different ways of constructing a personsname. For every culture a subclass of the class *name* can be defined in the ontology to support cultural differences. In this paper the construction for Dutch names, mainly before 1800, is elaborated. A typical Dutch name from this era has four parts:

- Voornaam (forename, e.g. the English equivalent "Joseph")
- Patroniem (patronymic)
- Tussenvoegsel (e.g. "van" of "van der")
- Achternaam (surname)

All of these parts are properties of the class *dutchName*.

```

Class(name partial)
Class(dutchName partial name)
DatatypeProperty(voornaam
  domain(dutchName)
  range(xsd:string))
DatatypeProperty(patroniem
  domain(dutchName)
  range(xsd:string))
DatatypeProperty(tussenvoegsel
  domain(dutchName)
  range(xsd:string))
DatatypeProperty(achternaam
  domain(dutchName)
  range(xsd:string))

```

The class *dutchName* can be related to a person with the following relations; i.e. are properties of the class *person*.

```

ObjectProperty(hasName
  domain(person)
  range(name))
ObjectProperty(hasDutchName
  domain(person)
  range(dutchName))
SubPropertyOf(hasDutchName hasName)

```

4.2.4 Event (genont)

A person goes through various events in his life. Typical events of genealogical interest are birth, death, marriage, baptism, burial or cremation. Here only the birth-event is described. Other events have the same construction.

```

Class(event partial
  restriction(certainty cardinality(1))
  restriction(place cardinality(1)))
Class(birth partial event)
ObjectProperty(isBorn
  domain(person)
  range(birth))

```

An *event* has a spatial and a temporal dimension, both of which can be related to the event. The property that relates the event to a *place* refers to the name of the place. Eventually it should refer to a formal definition of a geographical unit, in a certain period, most likely defined in a specific geographical ontology. Until then the name of the place is a simple string.

```

DatatypeProperty(place
  domain(event)
  range(xsd:string))

```

Time is modelled with a reference to the name of a moment: a date. In various cultures and eras the name of a moment is constructed in another way. In this ontology the Dutch culture of the gregorian date is elaborated.

```

Class(date partial)
Class(gregorianDate partial date
  restriction(day cardinality(1))
  restriction(month cardinality(1))
  restriction(year cardinality(1)))
DatatypeProperty(day
  domain(gregorianDate)
  range(xsd:string))
DatatypeProperty(month
  domain(gregorianDate)
  range(xsd:string))
DatatypeProperty(year
  domain(gregorianDate)
  range(xsd:string))

```

A moment is always a time interval. Maybe small, like the interval between the start and end of a minute, sometimes longer, like the interval between the first of

January and the last of December of a year. The relation between an event and the time interval in which the event takes place are inspired by [7]. Mostly the *event* takes place during the specified time-interval (like ‘the birth took place on October 29th, 2003’ or ‘He died in 1985’), but sometimes the source only reveals that an event took place before or after a specific interval. In the ontology the following three relations are defined: during, before and after.

```
ObjectProperty(certainty
  domain(event)
  range(date) )
ObjectProperty(after
  domain(event)
  range(gregorianDate) )
ObjectProperty(before
  domain(event)
  range(gregorianDate) )
ObjectProperty(during
  domain(event)
  range(gregorianDate) )
SubPropertyOf(after certainty)
SubPropertyOf(before certainty)
SubPropertyOf(during certainty)
```

4.2.5 Family-relations (genont)

Obviously in genealogy the family-relations are very important. These relations are modelled as properties of a person. As we have seen before there are two family-relations needed in all situations: the hasFather property and the hasMother property. All other types of family-relations can be constructed, e.g. the hasChild property.

```
ObjectProperty(hasParent
  inverseOf(hasChild)
  domain(person)
  range(person) )
ObjectProperty(hasChild
  inverseOf(hasParent)
  domain(person)
  range(person) )
ObjectProperty(hasFather
  domain(person)
  range(male) )
ObjectProperty(hasMother
  domain(person)
  range(female) )
SubPropertyOf(hasFather hasParent)
SubPropertyOf(hasMother hasParent)
```

5 Example

At [<http://www.zandhuis.nl/sw/genealogy/>] an example can be found of datafiles containing genealogical information and its sources. Besides the ontologies in XML-syntax, there are two datafiles: one containing the source-material (source.owl.rdf) and one containing the conclusions of the research (conclusion.owl.rdf). In the first file transcriptions of data from the archive is captured. The last file refers to the sources that prove the assertions made.

Conclusion

The Semantic Web is very suitable for publishing genealogical data in an open and extensible way. In this paper a first attempt is presented for a Genealogical Ontology, that can start the discussion for a standardized ontology, that improves the exchange of genealogical data and facilitates automatic processing. With such an ontology, standard software tools can be used to encode integrity checks on the data and perform intelligent processing.

References

- [1] Family History Department of The Church of Jesus Christ of Latter-day Saints, *The GEDCOM Standard* (Release 5.5), 2 January 1996 (<http://www.familysearch.org/GEDCOM/GEDCOM55.exe>)
- [2] *GENTECH Genealogical Data Model: A Comprehensive Data Model for Genealogical Research and Analysis* (version 1.1), May 29, 2000 (<https://www.ngsgenealogy.org/ngsgentech/projects/Gdm/Gdm.htm>)
- [3] G. Antoniou and F. van Harmelen, *A Semantic Web Primer*, London and Cambridge 2004.
- [4] <http://www.w3.org/RDF/>
- [5] <http://www.w3.org/TR/owl-semantics/>
- [6] N. Guarino, ‘Formal Ontology in Information Systems’, in: *Proceedings of FOIS98*, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press, pp. 3-15.
- [7] J.F. Allen, ‘Maintaining knowledge about temporal intervals’, in: *Communications of the ACM* 26, 832-843