Modelling (Historical) Administrative Information on the Semantic Web

Jesper Zedlitz
Christian-Albrechts-Universität
Kiel, Germany
Email: j.zedlitz@email.uni-kiel.de

Norbert Luttenberger
Christian-Albrechts-Universität
Kiel, Germany
Email: n.luttenberger@email.uni-kiel.de

***Abstract*—Identifying and referencing places is important for many fields of research. Very different approaches of how to represent administrative structures on the Semantic Web can be found. This survey attempts to provide a broad overview of systems that work on (historic) administrative information. We present a classification for such systems, with special attention to the difference that arise from the processing of historic data. We also describe a sample of systems which approach the problem in very different ways. We conclude by evaluating which of the presented characteristics make a system universal and future-proof.**

***Keywords*—conceptual modelling; administrative affiliation; semantic web; linked open data; historical information**

I. Introduction

When working with location information, it is often not enough to provide only the name of a place. One would like to uniquely identify the place. This identification opens up new possibilities. For example, the location of the place can be indicated on a map. During a search, one can aggregate places (e.g., search through all places within a province in one go).

However, this identification is not as simple as it might seem at a first glance. Usually, settlements (such as villages, group of houses, hamlets etc.) are embedded in an administrative structure. At least in a large part of the world there are parallel political, ecclesiastical, and judicial administrative structures—“administrative objects” for the sake of brevity. It is easy to see that the affiliation of settlements with these administrative objects is needed to provide the desired functionality—such as the already mentioned aggregated search. Changing administrative structures makes it difficult for the content editor to specify correct references as well as for the end-user to understand the changing structure and, as a consequence, formulate queries that include the expected results. For this reason, there are a number of projects that provide such kind of information on the Semantic Web in the form of Linked Open Data (LOD). The two technologies/languages that are typically used for this purpose are the Resource Description Framework (RDF) [1] and the Web Ontology Language (OWL) [2].

Since the representation of both spatial and temporal information with RDF or OWL is not trivial [3][4][5][6], there are very different approaches on how to represent administrative structures on the Semantic Web. This survey attempts to provide a broad overview of systems that provide (historical) administrative information.

Approaches such as named graphs [7] or contextual reasoning [8] can be used to enhance information with a time dimension. However, as the survey will show, these techniques have not yet found their way into existing systems that provide administrative information.

The paper is organized as follows. The difficulties involved with place identification (not only) on the Semantic Web are described in Section II. In Section III, a classification for systems providing (historical) administrative information is given. A sample of nine of such systems is presented in Section IV and characterized according to this classification. Section V concludes by summarizing which of the presented characteristics make a system universal and future-proof.

II. Problems of place identification

For frequent place names, the name alone is obviously not sufficient for identification—just think of “Neustadt” in Germany. If only the names of places were available, it would be impossible to distinguish between entries from different places with identical names.

Especially in Central Europe with its eventful history, it has often been the case that the name of a place changed over time. The problem is exacerbated in cases where different historic sources mention different names for the same place. Considering only the place name, one might wrongly assume that events have taken place at different locations, when in reality only the name of the place had changed over time. For a search one would like to see all results for one place, regardless of any name changes.

A common place name such as “Berlin” quickly leads to a presumption (“Berlin=capital of Germany”) that may turn out to be wrong for the specific source. Not only is there a settlement called “Berlin” in the municipality of Seedorf in Schleswig-Holstein, Germany, but also numerous other places called “Berlin” exist in the United States.

You have to analyse a source closely to understand what is meant exactly by a place’s name. In the simplest case it is the name of a settlement. However, it could also be the name of municipality or a parish. In that case the name might refer to a settlement—maybe with a different name—within that municipality/parish. Today, the settlement might belong to another parish or another community.

Copyright (c) IARIA, 2014. **ISBN: 978-1-61208-333-9**

A frequently found solution for the identification of places is the usage of geographic coordinates. However, that does not solve the problem.

- Which coordinate do you use for a large city—the coordinate of the town hall or the coordinate of the church? There might be no church in the village or there might be several to choose from.
- Given two slightly different coordinates it is not possible to tell that they point to the same place.
- What does the coordinate point to? There are probably several objects at that position: the church, the village, the municipality, the parish, the county, etc.

Therefore, geographic coordinates do not solve the problem. A unique identifier for settlements and administrative objects is needed. Such a unique identifier is associated with each resource on the Semantic Web—its Uniform Resource Identifier (URI).

III. Classification of model approaches

We have discovered several characteristics by which systems providing information about (historical) administrative structures can be classified. These characteristics fall into three groups: (A) fundamental decisions which are made regardless of the technology of the Semantic Web and which affect all systems, (B) characteristics which affect only directories in which the time is a concern, i.e. which also contain historical information. (C) characteristics which are based on the use of the techniques of RDF or OWL.

To some extent, the characteristics influence each other. Here is an example: If on the one hand you cannot or do not want to use the technique of reification, but on the other hand you want to give population numbers for different years, you are almost forced to work with different individuals for different points in time. Otherwise, different population numbers could not be distinguished.

A. Basic characteristics relevant for all directories

1) Use of sources: A simple characterization is the fact of whether source citations are provided for the published information or not. Especially in cases of contradicting information (e.g. different population numbers for the same point in time), source citations allow to judge the quality of the data.

2) Number of hierarchy levels: Some models have a limitation in the number of possible hierarchy levels. A typical example for three levels in the civil administration is a county belonging to a state which in turn belongs to a country. The administrative affiliation can be represented more generally if there is no restriction on the levels of the hierarchy. This is usually achieved by the definition of a general `isPartOf` or a `belongsTo` relationship.

3) Only current values or complete history: Some projects make the decision to provide only the latest data and no historical information. Therefore, no time-dependent values are needed. When a value or a name changes, the previous value is overwritten. One problem is that overwriting the values makes it difficult to reproduce reasoning. The value valid at the time of reasoning might be no longer available.

B. Basic characteristics for historical information

The following characteristics are only relevant for systems that also contain historical information. For systems that only provide current information, these characteristic do not apply.

1) Topology vs. specification of time: Instead of making specific indications of time when an administrative structure was established or dissolved, one may state topological relationships (i.e., predecessors, successors, etc.) instead. When working with such a topology, one does not have to deal with all the problems that accompany the treatment of time on the Semantic Web.

2) Activities or results: Two approaches exist to model changes in the administrative structure. The activities, i.e. the processes of changing (incorporation of one local authority by another, renaming, etc.), can be modelled, or the results of these changes. Figure 1 shows an example for both approaches. The left half shows the results of an incorporation. In the right half, the process of the incorporation is represented by an additional individual in the ontology.

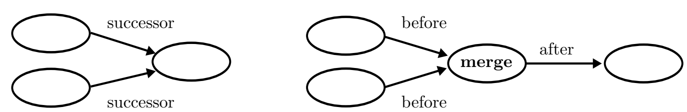

Fig. 1: Active vs. passive modelling.

3) Time-slices vs. individual times: To avoid problems that accompany the treatment of time on the Semantic Web, several ontologies can be used. Each of these ontologies contains information about just one single point in time. Within each of the ontologies, all relations and numerical values (e.g. population numbers) are specified, as they were valid at the selected point in time. One ontology represents one time-slice. An administrative object that exists across multiple time-slices will appear in each of the ontologies as an independent individual. It is possible to connect the different individuals belonging to the same administrative object by using additional object properties.

One disadvantage of the time-slice approach is that a completely new ontology for each considered point in time is needed. This solution is therefore only practical if one want to process a relatively small number of points in time. Throughout Germany changes at the municipal level occur about once a month. This would lead to a vast and hardly manageable number of ontologies.

4) One or multiple individuals: There are different approaches on how to model an administrative object during its entire lifespan. It can be represented by a single individual for its entire existence. In contrast, a new individual is created for every point in time—similar to time-slicing. As an intermediate form, a new individual is created only in cases where the administrative object is subject to changes.

Figure 2 sketches the different approaches. The variant when a single individual is used for the entire lifetime of an administrative object is shown on the left. All three time-dependent value are associated with that individual. At the center, the variant where a new individual is created only in case of the change of a value is illustrated. In this example,

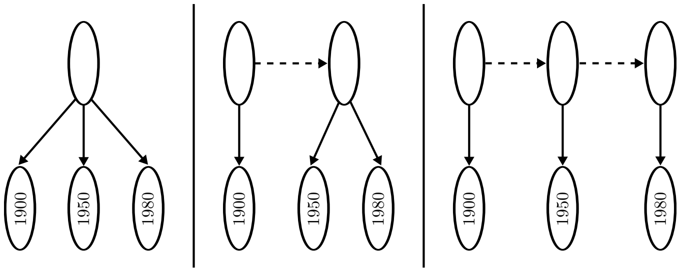

Fig. 2: Different approaches of using one or multiple individuals per administrative unit.

the value has changed between 1900 and 1950. Between 1950 and 1980, it has remained the same. Therefore, two different individuals are required. The dashed line shows a possible object property connection between the individuals that represent the same administrative object. The variant where a new individual is created for every point in time is shown on the right. Each individual is connected with exactly one value—regardless of whether the value has changed or not.

C. *RDF-/OWL-specific characteristics*

1) Text vs. objects: One way to provide information about administrative structures is to model only settlements as individuals. Information about affiliations is added to these individuals as literals (data properties) using the names of the administrative objects. An example of such modelling is shown in the upper half of Figure 3.

With this type of modelling, it is very easy to search for settlements which are subordinate to a higher administrative level (e.g., a state)—only simple comparisons of data properties are necessary. A navigation within the administrative hierarchy is not possible with this type of specification. Also, a clear identification of the administrative objects is not given: If two superordinate administrative objects have the same name they cannot be distinguished.

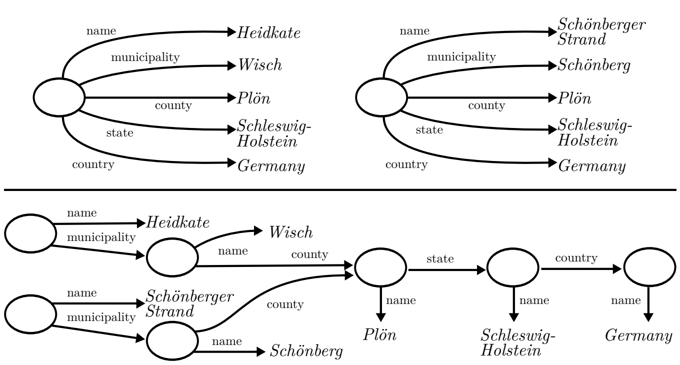

Fig. 3: Modeling with data properties only (top) and modelling with data and object properties (bottom).

This problem can be avoided by creating its own individual for each administrative object, which can be referenced by a URI—as usual on the Semantic Web. A link between the settlement and superordinate administrative object is created by using object properties. An example of this kind of modelling with individuals and object properties can be found in the lower half of Figure 3.

Since the objects of higher administrative levels (counties, states, etc.) occur only once and are named only once, one has to maintain less properties in total. The more objects are subordinate to another administrative object, the clearer this advantage becomes.

2) Types as classes, individuals, or literals: There are three ways to represent the type of an administrative object:

1) an OWL class—the individual representing the administrative object is instance of that class
2) a reference to an individual—the individual representing the administrative object has an object property that specifies the type
3) a literal—the individual representing the administrative object has a data property that contains the type as literal

In the first case, a municipality would be modelled as an instance of a class “Municipality”. If the municipality gets town privileges later, it also becomes instance of a class “City”. Here, a problem of this approach becomes evident: RDF or OWL have no time-dependent memberships in a class. Both class memberships are valid indefinitely. The change from municipality to city is no longer visible. Therefore, this approach is really only useful if you use multiple individuals for the representation of an administrative object over its lifetime.

3) Reification: Reification can be used in order to enrich a relation between two administrative objects with temporal information. Instead of defining an object property between two individuals directly, the connection itself is represented by an individual. Figure 4 illustrates how an administrative affiliation can be enriched with a time period by using reification.

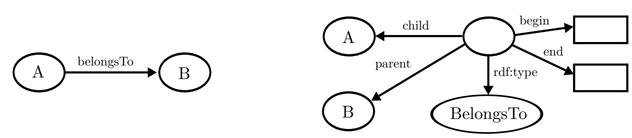

Fig. 4: Reification of an object property.

4) Specification of names: To specify names there is the choice to use the existing `rdfs:label` property or to define an own specialized property. The range of the `rdfs:label` property includes the data type `rdf:PlainLiteral`. In RDF, plain literals have an optional language tag. Therefore, it is possible to specify names in different language variants, e.g, `"München"@de`, `"Munich"@en`. In order to manage different name variants for one language, the relatively often used Simple Knowledge Organization System (SKOS) [9] provides several properties for names. To supply further information, such as a period of time or references for a name, it is necessary to define a specialized property. Of course, this property can be defined as sub-property of SKOS properties.

5) Specification of time as standalone property: Indications of time can be specified either as a standalone property or within literals—e.g., “Barmen-Elberfeld (-1930)”, “Wuppertal

(1930-)". A human user is likely to read and understand indications of time within literals. However, for machine processing (reasoning or SPARQL queries), they are not suitable.

IV. Projects in detail

For this article we selected projects that offer information about administrative structures on the Semantic Web. There are other approaches and projects for the publication of administrative structures. However, since they do not target the Semantic Web and its technologies, they are not covered in this article. Table I shows an overview of all nine projects that are discussed in this article. The table shows two rows for SAPO. The lower row represents an assumption about the ontologies used internally, based on the publications on SAPO. Each column of the table represents one of the characteristics listed above—in the same order as in this article.

TABLE I: Overview over the characteristics of the presented systems.

		all systems			systems with time				RDF/OWL technology				
	individuals	references	hierarchy levels	temporal inf.	time/topology	activity/result	time-slices/ ind. times	one/mult. indiv.	adm. obj. type	property for names	time as property	text/object	reification
schema.org	-	-	∞	-	*These systems do not contain temporal information.*				c	own	-	o	-
DBPedia	+	-	∞	-						rdfs:label	-	o	-
GeoNames	+	-	pr. 5	-					i	own	-	o	-
LinkedGeoData	+	-		-						rdfs:label	-	t	-
GB Ordnance Service	+	-	4	-					c	rdfs:label SKOS	-	o	-
SHV	+	-	pr. 5	-					c	rdfs:label	-	o	-
GND	+	-	∞	+	top.	r	-	-	i	own	-	o	-
SAPO	+	-	3	+	time	r	ind	mult	c	rdfs:label	t	o	-
SAPO (int.)	+	-		+	time	a	ts	mult	?	?	?	o	-
GOV	+	+	∞	+	time	r	ind	one	i	own	p	o	+

A. schema.org

Schema.org [10] is an initiative of several search engine operators. It provides vocabulary and the TBox of an ontology for semantic annotation of HTML pages. Thus, in contrasts to the other systems presented in this paper, the schema.org ontology does not contain any individuals. The ontology contains a part that deals with the description of administrative structures. The relevant classes and their relations are shown in Figure 5.

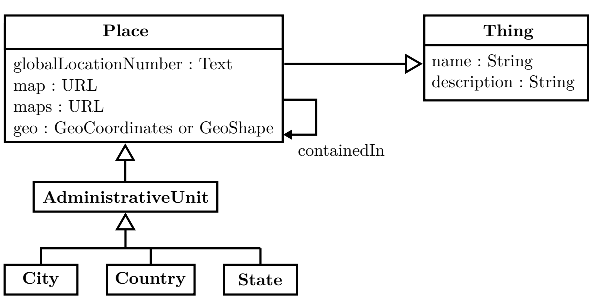

Fig. 5: Excerpt from the schema.org model that deals with administrative information.

The model is very simple. There are only three specialized types of administrative objects—cities, counties, and states. Additionally, there is a generic `AdministrativeUnit` class. However, it is not possible to specify the type of such a generic administrative object. In addition to the properties inherited from the class `Thing` to specify a name and a description, the geographic position can be specified by using two classes defined within the schema.org ontology.

The model provides no indication of time. Dependencies are represented using the object property `containedIn`. Due to the existence of the general `containedIn` relation, the representation of an arbitrary number of hierarchy levels is possible. Source citations are not possible.

B. DBPedia

The DBpedia project [11][12] extracts information from the various language variants of Wikipedia and publishes it as part of the LOD cloud as a RDF knowledge base. DBpedia's ontology contains a part that deals with the description of administrative structures. The currently most recent version of DBpedia's ontology can be found at [13]. The relevant sub-classes of `PopulatedPlace` are shown in Figure 6.

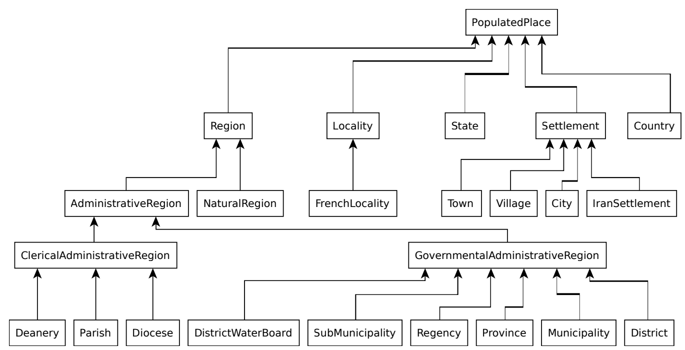

Fig. 6: Excerpt from the DBpedia ontology showing sub-classes of PopulatedPlace

In general, DBPedia contains no historical information. Only for one population number per individual an indication of time can be given by using the data property `populationAsOf`. Administrative affiliations are represented by the object property `isPartOf`. Additionally, there is an object property `country` which allows a direct connection to the containing country. Due to the existence of the general `isPartOf` relation, the representation of an arbitrary number of hierarchy levels is possible.

The type of an administrative object is specified by using OWL classes. In addition, however, there is the data property `settlementType`. It contains the type of the administrative object as literal. Source citations are rudimentarily possible. The object property `http://www.w3.org/ns/prov#wasDerivedFrom` is used to stated from which Wikipedia article the data was taken. For population numbers, the source citation can be specified as a note using the data property `http://dbpedia.org/property/populationNote`.

Basically, the ontology provides the distinction between settlements (`Settlement` and sub-classes) and administrative objects (`AdministrativeRegion` and sub-classes). The classes `Country`, `State`, and others show that this

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-333-9

distinction has not been carried out systematically. It can be observed that in some regions of the world, there is a fusion of settlements and administrative objects. This is problematic because they are actually different individuals—in the semantics of RDF. This is the case particularly with the information taken from German Wikipedia.

C. GeoNames

GeoNames [14] is a worldwide database containing information for more than 8 million settlements and administrative objects. It is probably the most commonly used gazetteer within the LOD cloud.

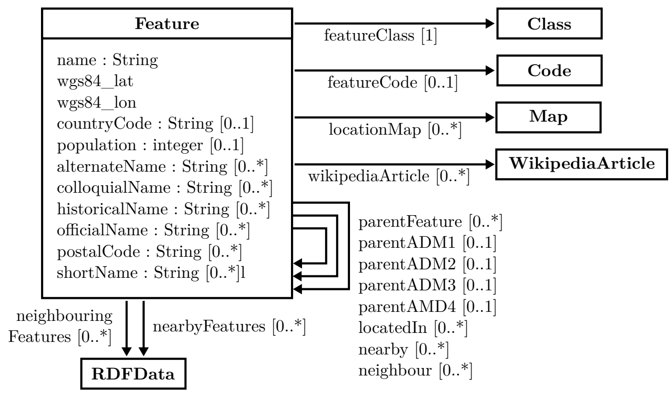

Fig. 7: GeoNames' main class `Feature`.

As it can be seen in Figure 7 the main class of GeoNames' data model is `Feature`. Using the properties `featureClass` and `featureCode` the type of the settlement or administrative object can be described in a very detailed way. Apart from a number of names, the geographical location can be specified. For this purpose, use is made of data properties from the WGS84 vocabulary [15].

The model provides no indication of times. Dependencies are represented by using the object property `parentFeature`. Thus, the representation of an arbitrary number of hierarchy levels is possible. Additionally, there are five specialized hierarchy levels which are represented by the object property `parentCountry`, `parentAdm1` ...`parentAdm4`. Therefore, the specification is practically limited to these five hierarchical levels. Source citations are not possible.

D. LinkedGeoData

The aim of LinkedGeoData [16] is to make the information collected in the OpenStreetMap [17] project available as a RDF knowledge base within the LOD cloud. The TBox of the ontology is very large, due to the types taken over from OpenStreetMap. However, the relevant part for the description of administrative structures is limited to the class `Place` and 16 sub-classes (`City`, `Continent`, `Country`, `County`, `Hamlet`, `Island`, `Islet`, `IsloatedDwelling`, `Locality`, `Municipality`, `Region`, `State`, `Subdivision`, `Suburb`, `Town`, and `Village`).

In contrast to all other systems presented in this paper, the specification of higher-level administrative objects is provided as literals using the data property `isIn`, e.g., `"Kiel,Schleswig-Holstein,Bundesrepublik Deutschland,Europe"`. Therefore, navigation within the administrative hierarchy is not possible.

The model provides no indication of time. The type of an administrative object is represented via OWL classes. The specification of names is done by using `rdfs:label`. Source citations are not possible.

E. Spatial Hierarchy Vocabulary

The Spatial Hierarchy Vocabulary [18] was created as part of the Leipzig professor catalogue. The structure of the model is very similar to the model of schema.org. Its classes and relations are shown in Figure 8.

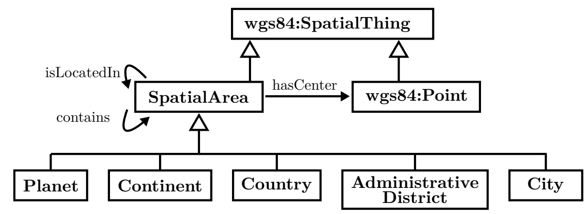

Fig. 8: The classes and relations of the Spatial Hierarchy Vocabulary

Dependencies are represented using the object property `isLocatedIn` and the inverse property `contains`. Thus, the representation of an arbitrary number of hierarchy levels is theoretically possible. The type of an administrative object is represented via OWL classes. However, since the model contains only five of these classes, the number of hierarchy levels is practically limited to five levels.

The model provides no indication of time. The specification of names is done by using `rdfs:label`. Source citations are not possible.

F. GB Ordnance Survey

Great Britain's national mapping agency publishes information about settlements and administrative objects in Great Britain as linked data. [19][20] In addition to geographical information, the published data also contain information on the administration.

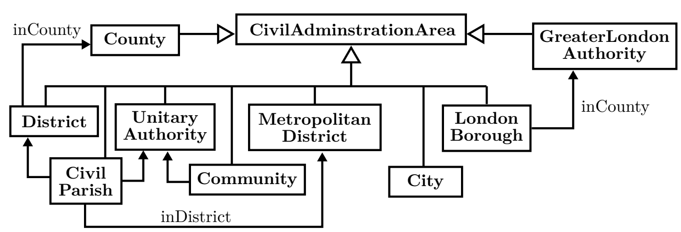

Fig. 9: Excerpt from the GB Ordnance Survey ontology.

Administrative objects are modelled as instances of the nine disjoint sub-classes of `CivilAdministrationArea` shown in Figure 9. The ontology specifies the relations between these classes very rigorously.

Via the `inDistrict`, the `inCounty` and the `inEuropeanRegion` object properties up to four hierarchy levels

Copyright (c) IARIA, 2014.
ISBN: 978-1-61208-333-9

can be specified. For the specification of names `rdfs:label` and SKOS data properties are used. The model provides no indication of time. Source citations are not possible.

G. Gemeinsame Normdatei (GND)

Within the “Gemeinsame Normdatei” (GND) the German National Library also publishes information about geographical objects. It lists both civil and ecclesiastical administrative structures. However, only in exceptional cases settlements are associated with them.

Figure 10 shows an excerpt of classes and properties from the GND ontology that are relevant for the modelling of administrative structures. Administrative objects are represented by individuals that are instances of the class `PlaceOrGeographicName` and its sub-classes.

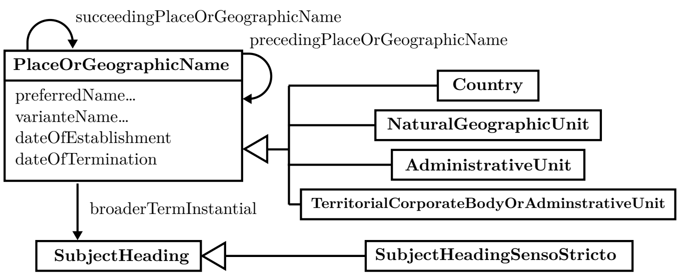

Fig. 10: Excerpt from the GND ontology that deals with administrative information.

In contrast to all other systems presented in this paper, topological relations between administrative objects are given by using the inverse object properties `succeeding-PlaceOrGeographicName` and `preceedingPlaceOf-GeographicName`. The type of an administrative object is represented by individuals that are members of the class `SubjectHeading`. They are connected to the individuals representing administrative objects via the object property `broaderTermInstantial`. For the specification of names two data properties have been defined. In some cases the website contains source citations. However, they are currently not available as Linked Data.

H. Suomen Ajallinen Paikka Ontologia (SAPO)

In [21] and [22] Kauppinen, Hyvönen et al. describe how data with time reference is stored in multiple ontologies as time-slices. In case of changes in the administrative structures between the times represented in the ontology, the process of change is modelled with the help of “change bridge classes”. Among the systems presented in this article, it is the only application of time-slices and the modelling of activities. Apparently, this modelling approach was used only internally in preparation of the published ontology. In today’s publicly accessible version of the “Suomen Ajallinen Paikka Ontologia” (SAPO) [23] these two ideas cannot be found.

For the lifetime of a administrative object, multiple individuals exist. After every change in the size (area) of the administrative object, a new individual is created. These individuals are combined into so-called “spaceworms”. Figure 11 shows an example of such a union.

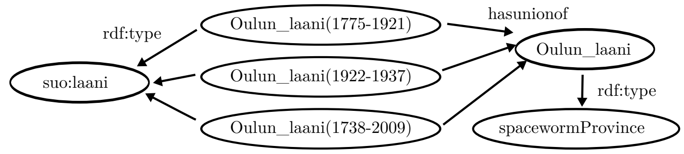

Fig. 11: Example for a “spaceworm” in the SAPO.

Indications of time are specified within `rdfs:label` values only. Therefore, they are not machine-interpretable and cannot be used for reasoning. There is no specialized data property for the specification of names—also `rdfs:label` is used. The type of an administrative object is specified by using OWL classes. The model contains three of these classes (`valtio`, `laani`, `kunta`) and hence three hierarchy levels. One can suspect that the internal ontologies of SAPO contain only these three classes, too.

I. Genealogisches Orts-Verzeichnis (GOV)

The most extensive data model is provided by “Genealogisches Orts-Verzeichnis” (GOV) [24]—a project of the German genealogical association “Verein für Computergenealogie”. The focus of the dataset is on Central Europe, but also data from the U.S. and Australia is included. In addition to structures of political administration, ecclesiastical and legal administrative structures can also be found.

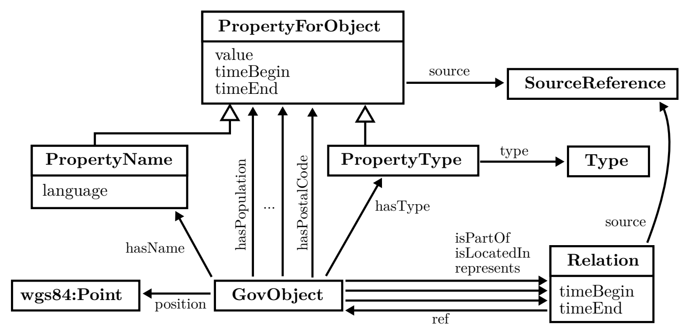

Fig. 12: Excerpt from the classes and properties of the Genealogisches Orts-Verzeichnis. Elements to handle references are left out.

In the GOV, the results of changes are modelled. A single individual is used for the entire lifetime of an administrative object. Therefore, each administrative object is associated with exactly one URI. Both historical affiliations as well as time-dependent values are given.

Figure 12 shows an excerpt of the classes and properties from the GOV ontology. In contrast to all other systems presented in this article, reification is used for both, relations and values, to specify indications of time. The reification also allows to give source citations. These source citations are listed as object properties, not in text form only.

Dependencies are modelled with object properties `is-PartOf`, `isLocatedIn`, and `represents` and the class `Relation`. Using the general `isPartOf` relation and the class `Relation`, the representation of an arbitrary number of parallel hierarchy levels is possible.

Names are specified using the object property `hasName` and the class `PropertyName`, which has a data property indicating the language as ISO-639-2 code. In combination with the aforementioned reification, it is possible to specify different language variants of the name as well as different names in the same language.

The type of an administrative object is represented by individuals that are connected via the object property `hasType` and the class `PropertyType`. Again, the reification makes it possible to model type changes with an indication of time and source citations.

V. Conclusion

There is a variety of different approaches on how to model historical administrative structures on the Semantic Web. In this article, nine systems were presented with their different approaches. In order to assess the differences better, a classification of systems has been developed which is divided into three main areas, each with different characteristics.

What characteristics make a system universal and future-proof? For some characteristics, this can be clearly stated. Especially for places from historical sources, time-dependent values are essential. An unlimited number of parallel affiliations enables the representation of the complex historical reality of administrative structures. Source citations allow quality control—usually difficult on the Semantic Web. Indication of time should be modelled as separate properties to make them machine-interpretable. The representation of the administrative structure should be done by using object properties—otherwise the key factor of unique identification will be lost. To provide information on time, sources, and the language used, the use of reification seems inevitable in the current state of RDF and OWL.

The use of time-slices is poorly scalable. However, during a preparation phase in the processing of sources (e.g. topographies which relate to a specific date), they might be useful. It is not future-proof to create a new individual for every point in time: There will be an unmanageable number of individuals (e.g. an annual addition of population numbers). The correct referencing of a single administrative object becomes difficult—but not impossible if additional union-individuals are defined.

Currently the modelling of results instead of activities is dominant. However, modelling activities could provide a better way to represent complex administrative processes. Particularly for processes that affect a multitude of objects, the correlation of the changes will be better understood.

Of the presented systems, the GOV seems to meet the requirements for a system that provides historical administrative structures best. Although GeoNames contains a significant larger number of settlements, the GOV contains the historical values and especially management objects which are essential for referencing places in a historic context.

References

[1] (2004) Resource Description Framework (RDF): Concepts and Abstract Syntax. (last access: 2014-02-14). [Online]. Available: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

[2] (2012) OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax. (last access: 2014-02-14). [Online]. Available: http://www.w3.org/TR/2012/REC-owl2-syntax-20121211/

[3] J. R. Hobbs and F. Pan, "An ontology of time for the semantic web," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 1, pp. 66–85, 2004.

[4] C. Gutierrez, C. A. Hurtado, and A. Vaisman, "Introducing time into rdf," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 2, pp. 207–218, 2007.

[5] F. Frasincar, V. Milea, and U. Kaymak, "towl: Integrating time in owl," in *Semantic Web Information Management*. Springer, 2010, pp. 225–246.

[6] B. Motik, "Representing and querying validity time in rdf and owl: A logic-based approach," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 12, pp. 3–21, 2012.

[7] J. J. Carroll, C. Bizer, P. Hayes, and P. Stickler, "Named graphs, provenance and trust," in *Proceedings of the 14th international conference on World Wide Web*. ACM, 2005, pp. 613–622. [Online]. Available: http://www2005.org/docs/p613.pdf

[8] M. Homola and L. Serafini, "Contextualized knowledge repositories for the semantic web," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 12, no. 0, 2012. [Online]. Available: http://www.websemanticsjournal.org/index.php/ps/article/view/231

[9] (2009) SKOS Simple Knowledge Organization System Reference. (last access: 2014-02-14). [Online]. Available: http://www.w3.org/TR/skos-reference/

[10] (2014) The schema.org website. (last access: 2014-02-14). [Online]. Available: http://schema.org

[11] (2014) The dbpedia.org website. (last access: 2014-02-14). [Online]. Available: http://www.dbpedia.org

[12] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "Dbpedia-a crystallization point for the web of data," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 3, pp. 154–165, 2009.

[13] (2013) DBpedia Ontology version 3.9. (last access: 2014-02-14). [Online]. Available: http://downloads.dbpedia.org/3.9/dbpedia_3.9.owl.bz2

[14] (2014) The GeoNames Website. (last access: 2014-02-14). [Online]. Available: http://www.geonames.org/

[15] (2009) WGS84 Geo Positioning: an RDF vocabulary. (last access: 2014-02-14). [Online]. Available: http://www.w3.org/2003/01/geo/wgs84_pos

[16] (2014) The LinkedGeoData Website. (last access: 2014-02-14). [Online]. Available: http://linkedgeodata.org

[17] (2014) The OpenStreetMap Website. (last access: 2014-02-14). [Online]. Available: http://www.openstreetmap.org

[18] M. Martin and T. Riechert, "Ortsbezogene navigation basierend auf einem vokabular zur erzeugung geographischer hierarchien," in *Catalogus Professorum Lipsiensis*. Universität Leipzig, 2010, pp. 107–116.

[19] (2014) Ordnance Survey Linked Data. (last access: 2014-02-14). [Online]. Available: http://data.ordnancesurvey.co.uk/datasets/os-linked-data

[20] J. Goodwin, C. Dolbear, and G. Hart, "Geographical linked data: The administrative geography of great britain on the semantic web," *Transactions in GIS*, vol. 12, no. s1, pp. 19–30, 2008.

[21] T. Kauppinen, R. Henriksson, J. Väätäinen, C. Deichstetter, and E. Hyvönen, "Ontology-based modeling and visualization of cultural spatio-temporal knowledge," in *Developments in Artificial Intelligence and the Semantic Web-Proceedings of the 12th Finnish AI Conference STeP 2006*, 2006, pp. 26–27.

[22] T. Kauppinen and E. Hyvönen, "Modeling and reasoning about changes in ontology time series," in *Ontologies*. Springer, 2007, pp. 319–338.

[23] (2014) The Finnish Spatio-temporal Ontology. (last access: 2014-02-14). [Online]. Available: http://onki.fi/en/browser/search/sapo?&os=sapo

[24] (2014) The GOV Website. (last access: 2014-02-14). [Online]. Available: http://gov.genealogy.net

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-333-9