A Survey on Modelling Historical Administrative Information on the Semantic Web

Jesper Zedlitz and Norbert Luttenberger
Christian-Albrechts-Universität
Kiel, Germany
Email: {j.zedlitz, 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. Let us give two examples:

(1) An obvious use case is the indication of a place's position on a map. This would not be possible having only the places' name.

(2) During a search, one can aggregate places. Imagine a data collection with census information. For each person a number of information including the occupation and the name of the place of residence is recorded. With a unique identification and administrative information linked to this identified place it becomes possible to search through all places within a province ("Show me all clock makers in Bavaria.") even if the places' names do not contain the province name.

However, this identification is not as simple as it might seem at a first glance. Usually, at the lowest level you have settlements such as villages, groups of houses, hamlets etc. These settlements 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. Also for the end-user changing structures are difficult understand. And, as a consequence, it is difficult for a user to formulate queries that include the expected results. Only a small example is given in Figure 1 showing only the affiliations for the late 19th and 20th century of the German village Suchsdorf as depicted by the GOV (see Section V. I).

Systems providing comprehensive historic administrative information help both, the content editors and the users, to navigate through these complex changing structures. 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).

This survey attempts to provide an overview of the different modelling approaches that are used to publish (historical) administrative information on the Semantic Web. It is the extended version of the paper published at WEB 2014 [1]. It is organized as follows. The difficulties involved with place identification (not only) on the Semantic Web are described in Section II. In Section III we discuss related work. In Section IV, a classification for systems providing (historical) administrative information is given. A sample of ten of such systems is presented in Section V and characterized according to this classification. Section VI concludes by summarizing which of the presented characteristics make a system universal and future-proof.

II. Problems of place identification

For common 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.

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.

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 example the German town "Wuppertal" carried the name "Barmen-Elberfeld" until 1930 (c.f. Section C.5). For a search one would like to see all results for one place, regardless of any name changes.

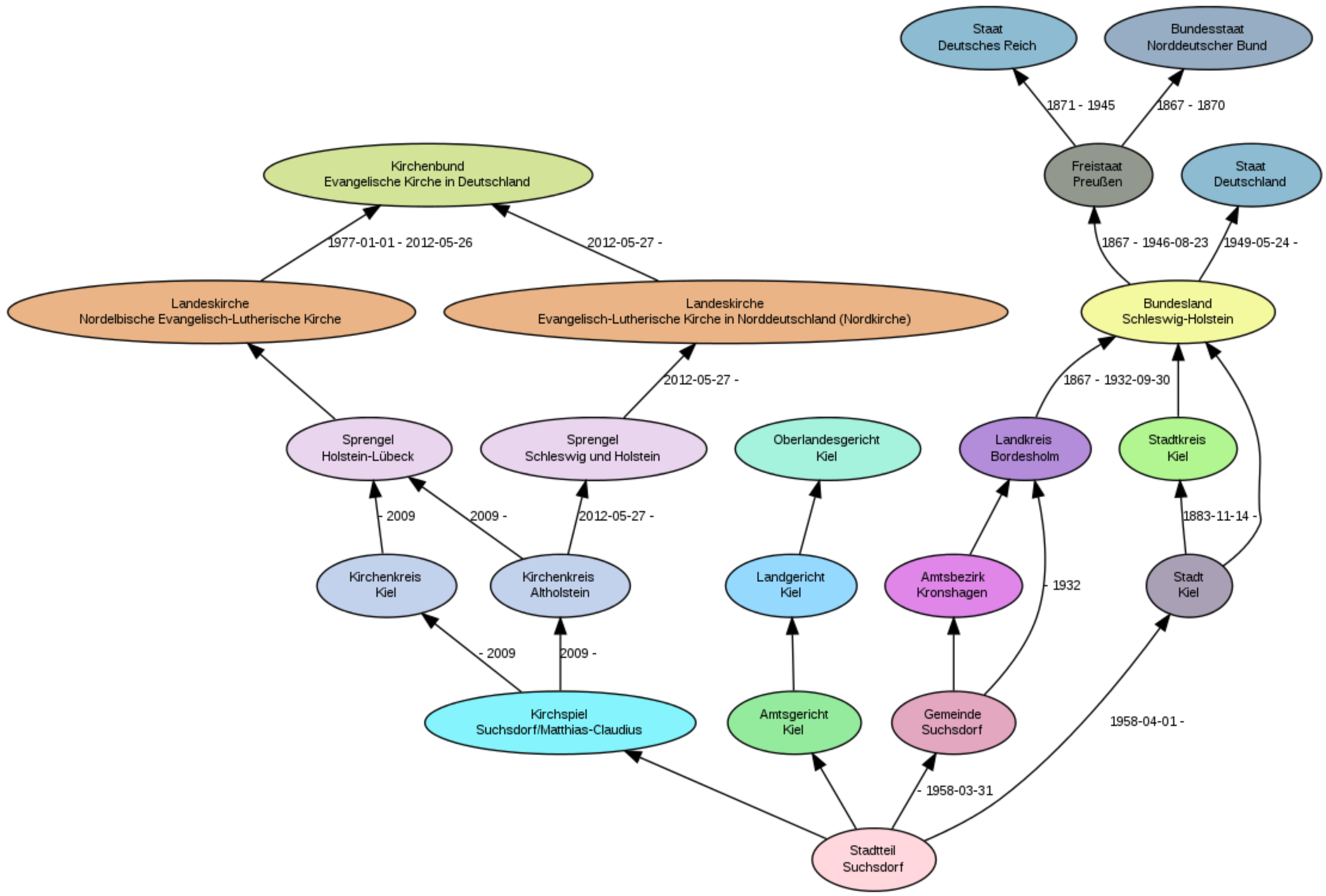

Fig. 1. Affiliations for the late 19th and 20th century of the German village Suchsdorf as depicted by the GOV.

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 such as a village, a group of houses, a hamlet etc. However, it could also be the name of municipality or a parish. In that case the place's 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. The correct solution in that case is to use a reference to the administrative object. Otherwise one would at least pretend accuracy that is not given by the source—or even create a reference to the wrong place.

We illustrate this problem with the help of the map in Figure 2. The three polygons show—from smallest to largest—the village Schönberg, the municipality Schönberg, and the parish Schönberg. If a source states that an event took place in the municipality of Schönberg, it would be wrong to use the identification for the village Schönberg: It is possible that the event took place in the village "Neu Schönberg". However, if a source states that a person came from the parish Schönberg, the parish has to be identified not the municipality or the village Schönberg. At least, the parish contains nine municipalities and even more different villages, which come into consideration.

An often proposed 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.

The map in Figure 2 illustrates this problem. Approximately in the middle of the map you can find the geographic coordinate 54.396°N, 10.370°E. However, what is identified by this coordinate? The Lutheran church of Schönberg is located there. However, also the village of Schönberg, the municipality of Schönberg and the parish Schönberg contain this coordinate.

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). For this reason, there are a number of projects that provide such kind of information on the Semantic Web with approaches that may be very different in other technical details.

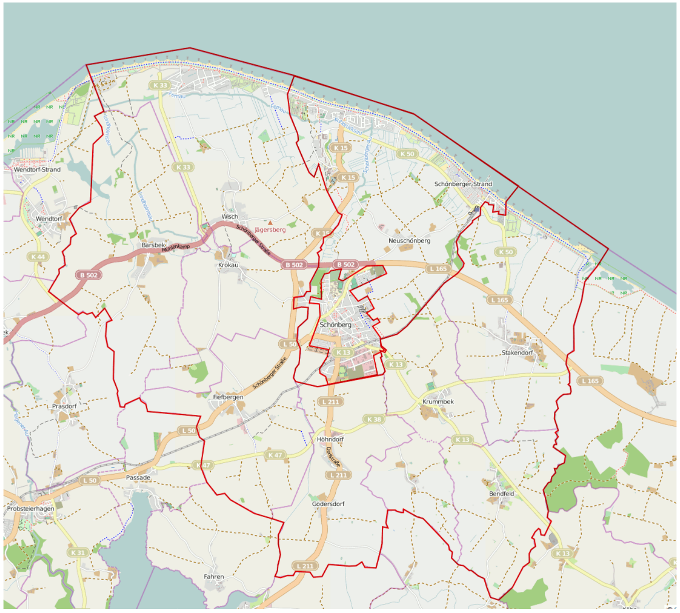

Fig. 2. Example for different administrative objects in the same region.

III. Related Work

The two technologies/languages that are typically used for publishing information on the Semantic Web are the Resource Description Framework (RDF) [2] and the Web Ontology Language (OWL) [3].

The representation of both spatial and temporal information with RDF or OWL is not trivial [4][5][6][7]. The problem of representing historical administrative information may be considered as a subproblem of modelling time in RDF and the Semantic Web. [8] contains a survey on approaches proposed to model temporal information, highlighting pros and cons of the different approaches.

Approaches such as named graphs [9] or contextual reasoning [10] 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.

IV. 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 systems 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 (see Section C.3), 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, population numbers for different year could not be distinguished.

A. *Basic characteristics relevant for all directories*

1) Source citations: 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. Figure 3 shows an example. 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.

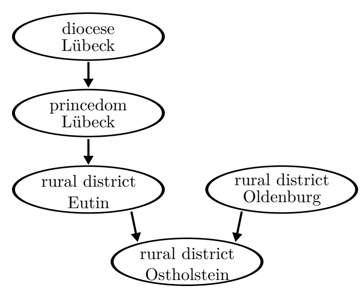

Fig. 3. Example for modelling changing administrative objects as a topology.

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 4 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.

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

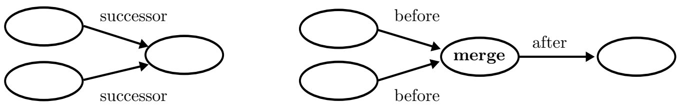

Fig. 4. Modelling a change as a result (left) or an an activity (right).

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. Figure 5 shows a sketch of this approach.

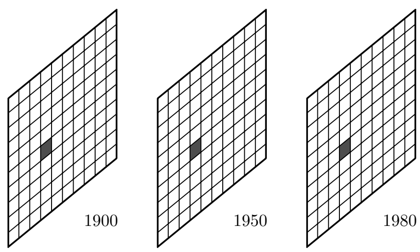

Fig. 5. Visualization of three "time-sliced" ontologies.

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.

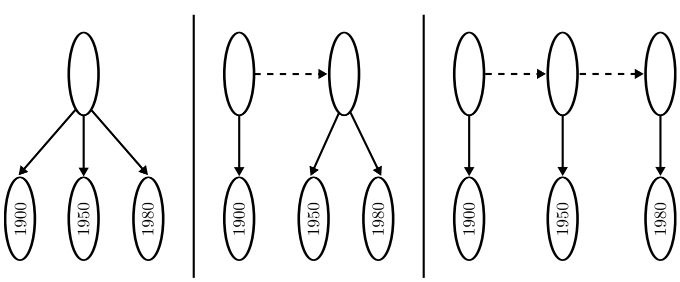

Fig. 6. Different approaches of using one or multiple individuals per administrative unit.

Figure 6 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 values 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, 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 7.

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.

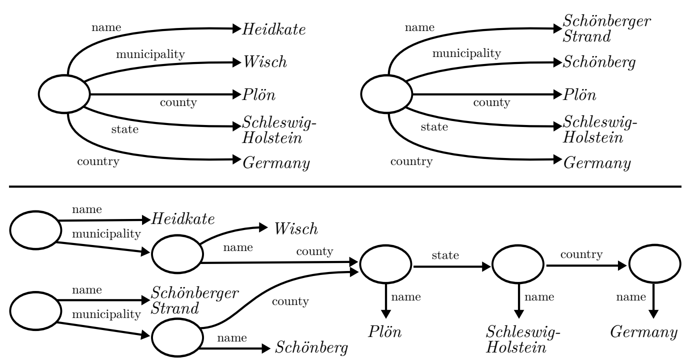

Fig. 7. Modeling with data properties only (top) and modelling with data and object properties (bottom).

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 7.

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. Figure 8 illustrates these different modelling approaches:

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

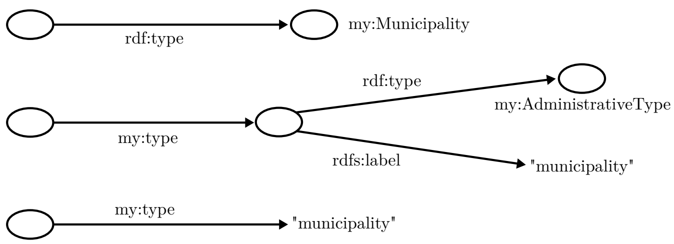

Fig. 8. Three different approaches to model the type of an administrative object.

The use of OWL classes (the first approach) has significant disadvantage when dealing with historic information: 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.

Using an individual (the second approach) has the obvious advantage that labels can be specified for different languages using RDF's language tags. In addition to the RDF/OWL-2 class-hierarchy an individual offers the possibility to create a hierarchy of types using a specialized object property. Since the relation between the administrative object and it's type object is an object property it is possible to use it for inference rules. Data properties (as used in the third approach) do not offer such a flexibility when defining inference rules.

3) Reification: In logic in general, and the Semantic Web in particular, the term "reification" has several meanings [11]:

a) an encoding of n-ary relations/properties as individuals
b) the possibility RDF offers to assign URIs to statements and treat them as resources
c) the use of classes as individuals
d) the usage of RDF as metalanguage for other logics

For the context of this article only the meanings (a) and (b) are relevant.

Chaning administrative affiliations can be modelled as n-ary relations. Not only the two individuals representing the involved administrative objects are relation members, but also temporal information, source citations, etc. are members of the relation. Instead of defining an object property between two individuals directly, the connection itself is represented by an individual. Figure 9 illustrates how an administrative affiliation can be enriched with a time period by using reification.

This encoding of n-ary relations into individuals works both for RDF and OWL-2 DL. However, RDF has a concept called

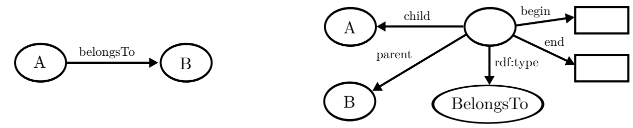

Fig. 9. Reification of an object property.

"higher-order statements" that can be used for making statements about other RDF statements. RDF offers the possibility to assign a URI to a statement and treat it as a resource. This resource can again be the subject of other statements. Here is an example. The first line and the last four lines have an equivalent meaning.

```
:A :belongsTo :B .

:s1 a rdf:Statement ;
   rdf:subject :A ;
   rdf:predicate :belongsTo ;
   rdf:object :B .
```

The main difference between these two approaches is the fact that in the latter case the reified statement (the object property connection) is still part of the model. However, these statements about other statements do not exist in OWL-2 DL.

4) Specification of names: To specify names there is the choice to use the existing `rdfs:label` property or to define a separate specialized property. The range of the `rdfs:label` property includes the data type `rdf:PlainLiteral`. In RDF, plain literals have an optional language tag as defined by [12]. Therefore, it is possible to specify names in different language variants, e.g., `"München"@de`, `"Munich"@en`. Using these language tags, even the specification of very specific languages or dialects is possible. For example, the language tag `sl-rozaj-biske` indicates the "San Giorgio dialect of Resian dialect of Slovenian" [12, p. 80].

Another observed possibility to specify names for different languages is the usage of a specialized data property for each language, e.g., `englishName` or `germanName`. This approach limits the number of supported languages, which might be an advantage for the implementation. However, the universality of the system is reduced.

In order to manage different name variants for one language, the relatively often used Simple Knowledge Organization System (SKOS) [13] 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 (see lines 1–4 in the example) or within literals—e.g., changing names of a German city from "Barmen-Elberfeld (-1930)" to "Wuppertal (1930-)" (see lines 6–9 in the example).

```
ex:Place1
  a ex:Municipality ;
  ex:name [ ex:value "Barmen-Elberfeld"; ex:end "1930"^^xsd
      :gYear ];
  ex:name [ ex:value "Wuppertal"; ex:begin "1930"^^xsd:
      gYear ] .

ex:Place2
  a ex:Municipality ;
  ex:name "Barmen-Elberfeld (-1930)" ;
  ex:name "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. For example it would not be possible to select the name of the place for a given year.

V. 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, e.g., the OpenGeoDB project (http://opengeodb.org). We also did not include ontologies or other data collections on the Semantic Web that only offer information for a single type or very few type of administrative objects, such as collections of statistical values for the countries of the world. The models for these kind of data are very simple and do not have to struggle with the problems discussed in this article.

We classify each modelling approach on the basis of the characteristics presented above. For most cases we show a visualisation of the project's model (or the relevant parts of the model) to provide the reader with a quick overview. Additionally, we give an example from the project's data written in Turtle[14]—a much more readable syntax than other RDF representations such as RDF/XML or NTriples.

Table I shows an overview of all ten projects that are discussed in this article. Each column of the table represents one of the characteristics listed above—in the same order as in this article. The table shows two rows for the SAPO (see Section V. H). The lower row represents an assumption about the ontologies used internally, based on the publications on SAPO.

A. schema.org

Schema.org [15] 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 10.

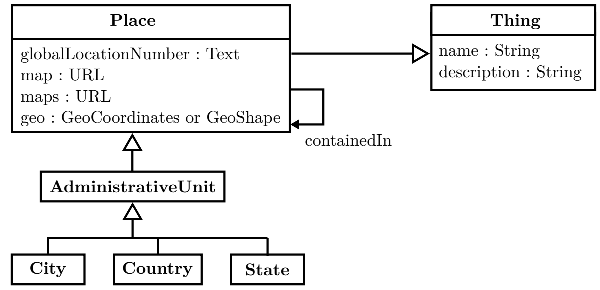

Fig. 10. Excerpt from the schema.org model that deals with administrative information.

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 [16][17] 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 [18]. The relevant sub-classes of `PopulatedPlace` and their connections via object properties are shown in Figure 11.

By taking a look a the ABox of the DBPedia ontology, it turned out that many of the classes in the `PopulatedPlace` class-hierarchy are currently not in use. The unused classes are depicted with gray rectangles in Figure 11.

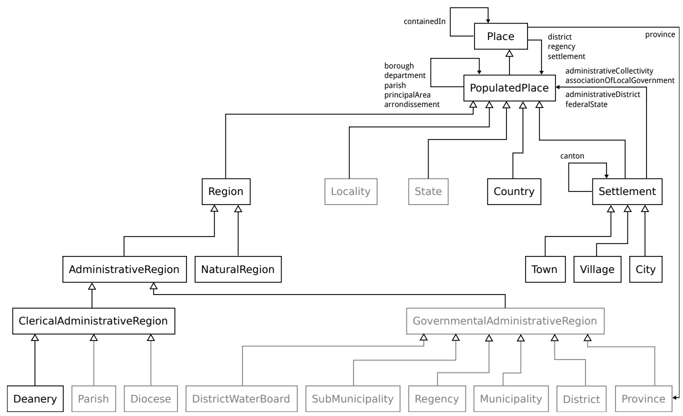

Fig. 11. Excerpt from the DBpedia ontology showing sub-classes of PopulatedPlace

The ontology also contains several other classes, which are not part of the `PopulatedPlace` class-hierarchy. However, the class names (e.g., "AustrianSettlement", "FrenchSettlement", "GermanSettlement") suggest that these classes should

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

	individuals	source citations	hierarchy levels	temporal information	time/topology	activity/result	time-slices/ individual times	one/multiple individuals per adm. object	administrative object type	property for names	time as property or within texts	adm. hierarchies as text or object	reification
		all systems			systems with time				RDF/OWL technology				
schema.org	-	-	∞	-	*These systems do not contain temporal information.*				class	own	-	obj.	-
DBPedia	+	-	∞	-						rdfs:label	-	obj.	-
GeoNames	+	-	pract. 5	-					indiv.	own	-	obj.	-
LinkedGeoData	+	-	-	-						rdfs:label	-	text	-
GB Ordnance Service	+	-	4	-					class	rdfs:label + SKOS	-	obj.	-
SHV	+	-	pract. 5	+	topol.	r	?	ind.	class	rdfs:label	-	obj.	-
GND	+	-	∞	+	topol.	r	-	-	indiv.	own	-	obj.	-
SAPO	+	-	3	+	time	r	ind.	mult.	class	rdfs:label	text	obj.	-
SAPO (interal)	+	-		+	time	a	t.s.	mult.	?	?	?	obj.	-
GOV	+	+	∞	+	time	r	ind.	one	indiv.	own	prop.	obj.	+
TGN	+	+	∞	+	time	r	ind.	one	indiv.	own	prop.	obj.	+

actually be subclasses of `Settlement`, too. The ontology also contains properties for these classes. However, currently neither the classes nor the properties are used.

In general, DBPedia contains no historical information. Only for a single population number per individual an indication of time can be given by using the data property `populationAsOf`. Source citations are rudimentary 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`.

The type of an administrative object is specified by using OWL classes (C.2), see Figure 11. In addition, however, there is the data property `settlementType`. It contains the type of the administrative object as literal (C.2).

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 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.

Administrative affiliations are represented by the object property `isPartOf`. Additionally, there are a couple of object properties (e.g., `country`, `federalState`, `municipality`), which allows a direct connection to higher level administrative objects. Due to the existence of the general `isPartOf` relation, the representation of an arbitrary number of hierarchy levels is therotically possible (A.2).

In practice, administrative affiliations are represented very differently. Figures 12–14 show similar administrative structures for three different countries, a settlement with four (Poland and Germany) resp. three (Denmark) administrative levels. As you can see, different object properties are used in each case.

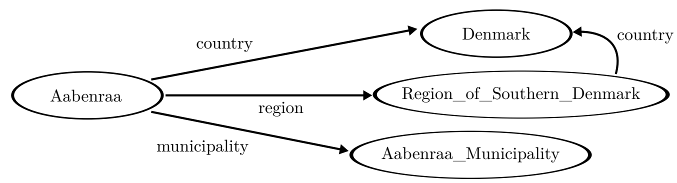

Fig. 12. Example from the DBPedia ontology showing a Danish settlement and its administrative objects.

For the Danish settlement (Figure 12) the object properties `municipality`, `region`, and `country` are used to connect it directly to all higher-level objects. However, the municipality is not connected to any superordinate objects.

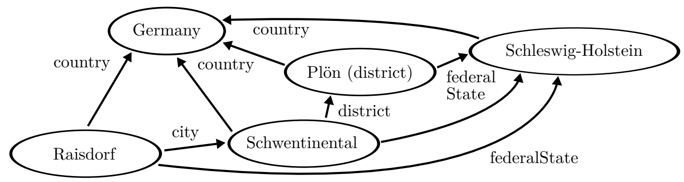

Fig. 13. Example from the DBPedia ontology showing a German settlement and its administrative objects.

The German settlement (Fig. 13) is also directly connected its higher-level objects, although not all of them. Different object properties are used: `city` and `federalState`. The municipality (Schwentinental) is directly connected to the rural district, the federal state and the country.

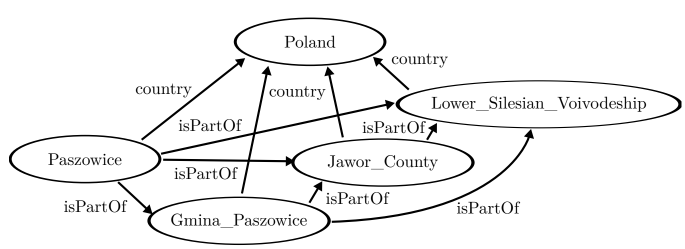

Fig. 14. Example from the DBPedia ontology showing a Polish settlement and its administrative objects.

For the Polish village (Figure 14) a more generic approach using the `isPartOf` object property is used. Only for the connection to the country a specialized object property `country` is used. However, `country` is not a sub-property of `isPartOf`. All individuals are directly connected with its higher-level objects.

These different modelling approaches within the same ontology makes it quite difficult to work with the DBPedia ontology.

```
@prefix schema: <http://schema.org/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix virtrdf: <http://www.openlinksw.com/schemas/virtrdf
    #> .
@prefix dbpo: <http://dbpedia.org/ontology/> .
@prefix dbpedia: <http://dbpedia.org/resource/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix georss: <http://www.georss.org/georss/> .

dbpedia:Raisdorf
 a dbpo:Settlement, dbpo:Place,
  dbpo:PopulatedPlace, schema:Place,
  <http://www.opengis.net/gml/_Feature> ;
 foaf:homepage <http://www.raisdorf.de/> ;
 foaf:isPrimaryTopicOf dbpedia:Raisdorf ;
 foaf:name "Raisdorf"@en ;
 georss:point "54.266666666666666 10.216666666666667"@en ;
 geo:geometry "POINT(10.216666 54.266666)"^^virtrdf:
     Geometry ;
 geo:lat "54.2667"^^xsd:float ;
 geo:long "10.2167"^^xsd:float ;
 dbpo:areaCode "04307, 04342"@en ;
 dbpo:areaTotal 1.129000e+7 ;
 dbpo:city dbpedia:Schwentinental ;
 dbpo:country dbpedia:Germany ;
 dbpo:elevation 36 ;
 dbpo:federalState dbpedia:Schleswig-Holstein ;
 dbpo:populationAsOf "2006-09-29+02:00"^^xsd:date ;
 dbpo:populationTotal 7675 ;
 dbpo:postalCode "24223"@en ;
 prov:wasDerivedFrom
  <http://en.wikipedia.org/wiki/Raisdorf?oldid=540674771>
     ;
 rdfs:label "Raisdorf"@nl, "Raisdorf"@en, "Raisdorf"@de .
```

C. GeoNames

GeoNames [19] 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.

As it can be seen in Figure 15 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 [20].

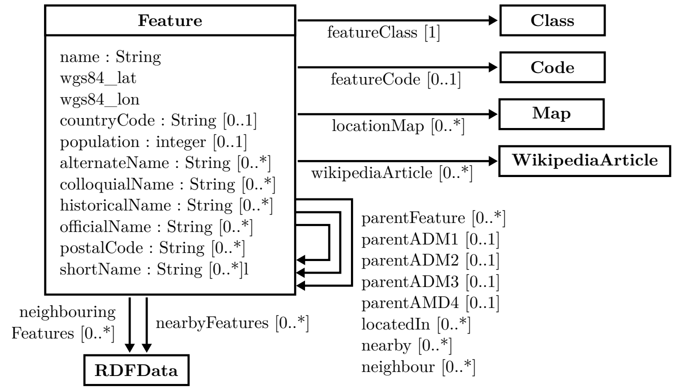

Fig. 15. GeoNames' main class `Feature`.

The model provides no indication of times (A.3). Dependencies are represented by using the object property `parentFeature` (C.1). Thus, the representation of an arbitrary number of hierarchy levels is possible (A.2). 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 (A.1).

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix gn: <http://www.geonames.org/ontology#> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .

<http://sws.geonames.org/2825253/>
 a gn:Feature ;
 rdfs:isDefinedBy <http://sws.geonames.org/2825253/about.
     rdf> ;
 gn:name "Suchsdorf" ;
 gn:featureClass gn:P ;
 gn:featureCode gn:P.PPLX ;
 gn:countryCode "DE" ;
 geo:lat "54.35937" ;
 geo:long "10.07947" ;
 gn:parentFeature <http://sws.geonames.org/2862623/> ;
 gn:parentCountry <http://sws.geonames.org/2921044/> ;
 gn:parentADM1 <http://sws.geonames.org/2838632/> ;
 gn:nearbyFeatures <http://sws.geonames.org/2825253/nearby
     .rdf> ;
 gn:locationMap <http://www.geonames.org/2825253/suchsdorf
     .html> .
```

D. LinkedGeoData

The aim of LinkedGeoData [21] is to make the information collected in the OpenStreetMap [22] 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 (C.1) 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 (A.3). The type of an administrative object is represented via OWL classes (C.2). The specification of names is done by using `rdfs:label` (C.4). Source citations are not possible (A.1).

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix lgd: <http://linkedgeodata.org/ontology/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix dc: <http://purl.org/dc/terms/> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix geom: <http://geovocab.org/geometry#> .
@prefix data: <http://linkedgeodata.org/triplify/> .

data:node1454999119
  a <http://geovocab.org/spatial#Feature>, lgd:Place,
   <http://linkedgeodata.org/meta/Node>, lgd:Suburb ;
  lgd:changeset "9470245"^^xsd:int ;
  lgd:version "1"^^xsd:int ;
  rdfs:label "Suchsdorf" ;
  geo:lat 5.435505e+1 ;
  geo:long 1.008036e+1 ;
  lgd:isIn "Kiel,Schleswig-Holstein,Bundesrepublik
       Deutschland,Europe" ;
  dc:contributor data:user472256 ;
  dc:modified "2011-10-04T17:16:49"^^xsd:dateTime ;
  geom:geometry <http://linkedgeodata.org/geometry/
       node1454999119> .
```

E. GB Ordnance Survey

Great Britain's national mapping agency publishes information about settlements and administrative objects in Great Britain as linked data. [23][24] In addition to geographical information, the published data also contain information on the administration.

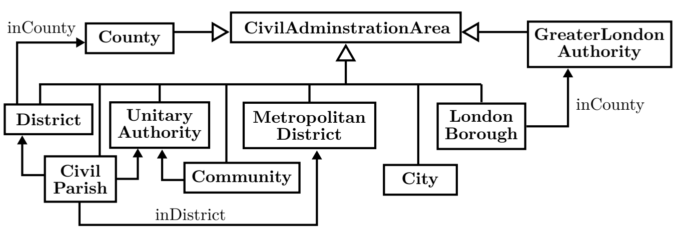

Fig. 16. Excerpt from the GB Ordnance Survey ontology.

Administrative objects are modelled as instances of the nine disjoint sub-classes of `CivilAdministrationArea` shown in Figure 16. 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 can be specified (A.2). For the specification of names `rdfs:label` and SKOS data properties are used (C.4). The model provides no indication of time (A.3). Source citations are not possible (A.1).

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dc: <http://purl.org/dc/terms/> .
@prefix spatial: <http://data.ordnancesurvey.co.uk/ontology
     /spatialrelations/> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix admingeo: <http://data.ordnancesurvey.co.uk/
     ontology/admingeo/> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix georss: <http://www.georss.org/georss/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix geom: <http://data.ordnancesurvey.co.uk/ontology/
     geometry/> .
@prefix gbos: <http://data.ordnancesurvey.co.uk/id/> .
@prefix gbstat: <http://statistics.data.gov.uk/id/
     statistical-geography/> .

gbos:7000000000015052
  a admingeo:County ;
  spatial:contains gbos:7000000000014797, [...] gbos
       :7000000000014892 ;
  spatial:touches gbos:7000000000023423, [...] gbos
       :7000000000043822 ;
  spatial:northing 325253 ;
  skos:prefLabel "Staffordshire" ;
  admingeo:district gbos:7000000000014727, [...] gbos
       :7000000000014869 ;
  geo:lat 52.8248 ;
  rdfs:label "Staffordshire" ;
  admingeo:hasUnitID "15052" ;
  admingeo:hasAreaCode "CTY" ;
  admingeo:inEuropeanRegion gbos:7000000000041426 ;
  admingeo:gssCode "E10000028" ;
  georss:point "52.824807 -2.006617" ;
  owl:sameAs gbstat:E10000028 ;
  spatial:easting 399651 ;
  geom:extent geom:15052-16 ;
  spatial:within gbos:7000000000041426 ;
  geo:long -2.00662 .
```

F. Spatial Hierarchy Vocabulary

The Spatial Hierarchy Vocabulary [25][26] 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 17.

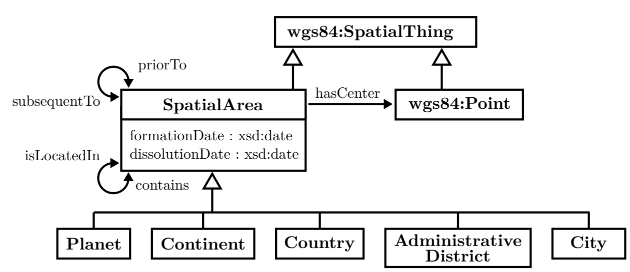

Fig. 17. The classes and relations of the Spatial Hierarchy Vocabulary

Dependencies are represented using the object property `isLocatedIn` and the inverse property `contains` (C.1). 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 (A.2).

The model provides basic indication of time: It is possible to specify the establishment and the termination date of an object using data properties. Topological relations (B.1) between administrative objects are given by using the inverse object properties `priorTo` and `formationDate`. The specification of names is done by using `rdfs:label` (C.4). Source citations are not possible (A.1).

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 18 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.

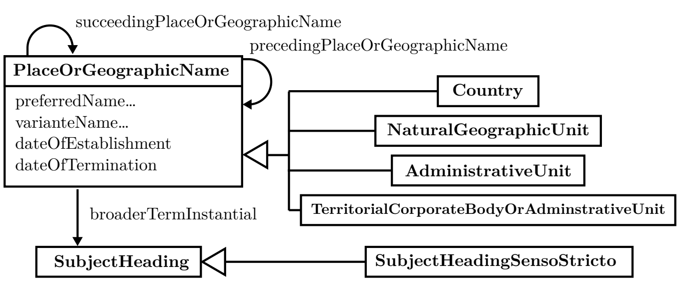

Fig. 18. Excerpt from the GND ontology that deals with administrative information.

The GND provides topological relations (B.1) between administrative objects by using the inverse object properties `succeedingPlaceOrGeographicName` and `preceedingPlaceOfGeographicName`. Additionally, establishment and the termination date of an object can be specified with data properties.

The type of an administrative object is represented by individuals that are members of the class `SubjectHeading` (C.2). 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 (C.4). In some cases the website contains source citations (A.1). However, they are currently not available as Linked Data.

```
@prefix gndo: <http://d-nb.info/standards/elementset/gnd#> .
@prefix gnd: <http://d-nb.info/gnd/> .
@prefix area: <http://d-nb.info/standards/vocab/gnd/geographic-area-code#> .

<http://d-nb.info/gnd/4091969-9>
  a gndo:TerritorialCorporateBodyOrAdministrativeUnit,
   gndo:AdministrativeUnit ;
  gndo:gndIdentifier "4091969-9" ;
  gndo:oldAuthorityNumber "(DE-588c)4091969-9" ;
  gndo:broaderTermInstantial gnd:4073976-4 ;
  gndo:geographicAreaCode area:XA-DE-SH ;
  gndo:variantNameForThePlaceOrGeographicName
     "Eutin <Kreis>" ;
  gndo:preferredNameForThePlaceOrGeographicName
     "Landkreis Eutin" ;
  gndo:precedingPlaceOrGeographicName gnd:4115317-0 ;
  gndo:succeedingPlaceOrGeographicName gnd:4044079-5 .
```

H. Suomen Ajallinen Paikka Ontologia (SAPO)

In [27] and [28] 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 (B.3) and the modelling of activities (B.2). 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) [29] these two ideas cannot be found.

For the lifetime of a administrative object, multiple individuals exist (B.4). 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 19 shows an example of such a union.

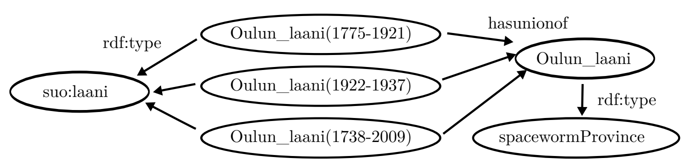

Fig. 19. Example for a “spaceworm” in the SAPO.

Indications of time are specified within `rdfs:label` values only (C.5). 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 (C.4). The type of an administrative object is specified by using OWL classes (C.2). The model contains three of these classes (`valtio`, `laani`, `kunta`) and hence three hierarchy levels (A.2). One can suspect that the internal ontologies of SAPO contain only these three classes, too.

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix sapo: <http://www.yso.fi/onto/sapo/> .
@prefix suo: <http://www.yso.fi/onto/suo/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

<http://www.yso.fi/onto/sapo/Ypaja(1910-)>
   a suo:kunta ;
   dc:isPartOf
     <http://www.yso.fi/onto/sapo/Etela-Suomen_laani(1997-2009)>,
     <http://www.yso.fi/onto/sapo/Hameen_laani(1870-1959)>,
     <http://www.yso.fi/onto/sapo/Hameen_laani(1969-1972)>,
     <http://www.yso.fi/onto/sapo/Hameen_laani(1993-1996)>;
   rdfs:label "Ypäjä(1910-)" .

sapo:Ypaja
   a sapo:spaceworm ;
   rdfs:label "Ypäjä" .
```

I. *Genealogisches Orts-Verzeichnis (GOV)*

One of the most extensive data models in this survey is provided by “Genealogisches Orts-Verzeichnis” (GOV) [30][31], 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 be found.

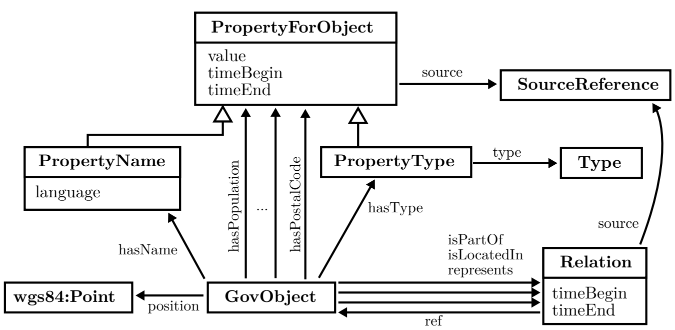

Fig. 20. Excerpt from the classes and properties of the Genealogisches Orts-Verzeichnis. Elements to handle references are left out.

In the GOV, the results of changes (B.2) are modelled. A single individual is used for the entire lifetime of an administrative object (B.4). Therefore, each administrative object is associated with exactly one URI. Both historical affiliations as well as time-dependent values are given (A.3).

Figure 20 shows an excerpt from the GOV ontology. In contrast to most systems presented in this article, reification—encoding of n-ary properties as individuals— (C.3) is used for relations and values to specify indications of time. The reification also allows to give source citations (A.1). These source citations are listed as object properties, not in text form only. That enables queries and reasoning over these source citations.

Dependencies are modelled with object properties `isPartOf`, `isLocatedIn`, and `represents` and the class `Relation` (C.1). Using the general `isPartOf` relation and the class `Relation`, the representation of parallel hierarchies with an arbitrary number of levels is possible (A.2).

Names are specified using the object property `hasName` and the class `PropertyName` (C.4), 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` (C.2). Again, the reification makes it possible to model type changes with an indication of time and source citations.

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix gov: <http://gov.genealogy.net/ontology.owl#> .
@prefix type: <http://gov.genealogy.net/types.owl#> .

<http://gov.genealogy.net/SUCORFJO54AI>
  a gov:GovObject ;
  rdfs:isDefinedBy <http://gov.genealogy.net/SUCORFJO54AI/about.rdf> ;
  gov:hasPopulation [ a gov:PropertyForObject ;
     gov:value "8441" ;
     gov:timeBegin "2004-12-31" ;
     gov:timeEnd "2004-12-31" ] ;
  gov:property [ a gov:Property ; gov:value "8.04" ] ;
  gov:hasName [ a gov:PropertyName ;
     gov:value "Suchsdorf" ;
     gov:language "deu" ] ;
  gov:hasType [ a gov:PropertyType ;
     gov:type type:55 ;
     gov:timeEnd "1958" ;
        gov:source [ a gov:SourceReference ;
             gov:sourceRef <http://gov.genealogy.net/source_387809> ;
             gov:note "S. 152" ]
       ],
     [ a gov:PropertyType ; gov:type type:54 ; gov:timeBegin "1958" ] ;
  gov:isPartOf
       [ a gov:Relation ;
        gov:ref <http://gov.genealogy.net/object_386988> ;
        gov:timeEnd "1958-03-31" ],
       [ a gov:Relation ; gov:ref <http://gov.genealogy.net/KIEIELJO54BI> ;
      gov:timeBegin "1958-04-01" ] ,
       [ a gov:Relation ;
        gov:ref <http://gov.genealogy.net/object_1042608> ],
       [ a gov:Relation ;
        gov:ref <http://gov.genealogy.net/object_285109> ;
        gov:source [a gov:SourceReference ;
             gov:sourceRef <http://gov.genealogy.net/source_387809> ;
             gov:note "S. 152" ]
       ] .
```

J. *Thesaurus of Geographic Names (TGN)*

The second extensive data model in this survey is provided by the “Thesaurus of Geographic Names” [32]. It is a project of the Getty Research Institute. TGN is part of the Getty Vocabulary Program (GVP) [33] ontology. Therefore, most of its properties have a broad domain and range to be compatible with other Getty vocabularies such as the Art & Architecture Thesaurus (AAT). The ontology extensively reuses components from many other ontologies such as SKOS [13], W3C’s PROV [34], Dublin Core (DC), schema.org and others.

Due to the aforementioned reuse of ontology components and the generic definitions of domains and ranges of object properties, it is hard to comprehend the key model design criteria for the TGN. We tried our best to compose the various elements from the different ontologies and observed the domain and range restrictions found in the data to make the structure visible. Figure 21 shows this structure.

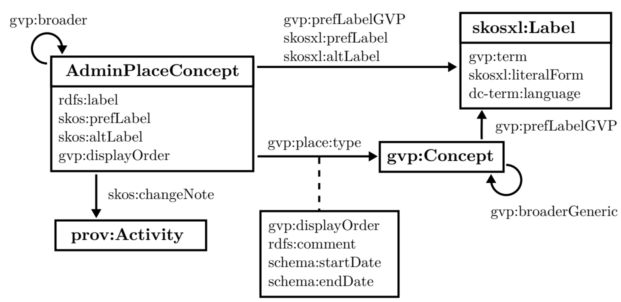

Fig. 21. Structure of an individual in the TGN.

In the TGN, the results of changes (B.2) are modelled. A single individual is used for the entire lifetime of an administrative object (B.4). Since the TGN uses a dual practice to separate between Concept and Thing [33, p. 48] the ontology also contains a second individual for each administrative object. A user of the TGN has to distinguish carefully, which individual (respective URI) to use. However, a discussion on the “Concept vs Place Duality” is beyond the scope of this article.

The TGN uses two different kinds of reification (C.3). Place names are modelled as encoded n-ary relations resulting in an individual of the class `skosxl:Label` (lines 42–48 in the example). TGN also uses RDF’s higher-order statements for making statements about other RDF statements. Types of administrative objects and dependencies are mostly modelled that way in the TGN (lines 50–54 in the example). Source citations, indications of time and display orders can be added to these named statements (line 54 in the example).

Dependencies are modelled with several object properties, which are sub-properties of each other (C.1). The most generic ones are `gvp:broader` and `skos:broader`. Using the general `gvp:broader` relation, the representation of parallel hierarchies with an arbitrary number of levels is theoretically possible (A.2). However, in practice—at least for central Europe—only a few fragmentary administrative affiliations are listed. In addition to the object properties one hierarchy of administrative objects is given as data property `gvp:parentString` (line 24 in the example). If a change at any position of the hierarchy structure occurs, the value of this data property has to be changed for all subordinate objects. There is a risk that this will be forgotten and the ontology will contain contradicting information.

The type of an administrative object is represented by individuals (C.2) that are connected via the object property `gvp:placeType` and sub-properties (lines 31 and 50–54 in the example). These properties links to an individual from the AAT. These individuals are put into a separate complex type hierarchy via object properties).

Names are specified using the object properties `gvp:prefLabelGVP` and `skosxl:prefLabel` and the class `skosxl:Label` (lines 42–48 in the example). The language of the name is indicated in two ways, a RDF language tag at the Label’s data property `gvp:term` and an object property from DC `dcterms:language`. 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.

One irritating fact in the context of names is the custom data property `displayOrder` (line 26 in the example). According to [33, p. 31] one can use this property to sort the places alphabetically. However, if a new place is added to the ontology the property values of all individuals following in alphabetical order have be to changed.

Source citations (A.1) as well as revision history play an important role in TGN and are modelled extensively (lines 37–40 in the example). Source citations are listed as individuals, not in text form only. That enables queries and reasoning over these source citations.

The TGN renders outstanding services to the modelling of provenance. However, the introduction of a semantic beyond OWL-2 is problematic.

```
1  @prefix dc: <http://purl.org/dc/elements/1.1/> .
2  @prefix dc-term: <http://purl.org/dc/terms/> .
3  @prefix foaf: <http://xmlns.com/foaf/0.1/> .
4  @prefix gvp: <http://vocab.getty.edu/ontology#> .
5  @prefix prov: <http://www.w3.org/ns/prov#> .
6  @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
      .
7  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
8  @prefix skos: <http://www.w3.org/2004/02/skos/core#> .
9  @prefix skosxl: <http://www.w3.org/2008/05/skos-xl#> .
10 @prefix tgn: <http://vocab.getty.edu/tgn/> .
11 @prefix tgn_rel: <http://vocab.getty.edu/tgn/rel/> .
12 @prefix tgn_rev: <http://vocab.getty.edu/tgn/rev/> .
13 @prefix tgn_term: <http://vocab.getty.edu/tgn/term/> .

15 tgn:7074121 a skos:Concept, gvp:Subject,
16            gvp:AdminPlaceConcept ;
17       rdfs:label "Suchsdorf" ;
18       rdfs:seeAlso <http://www.getty.edu/vow/TGNFullDisplay
            ?find=&place=&nation=&subjectid=7074121> ;
19       dc-term:contributor tgn_contrib:10000000 ;
20       skos:inScheme <http://vocab.getty.edu/tgn/> ;
21       skos:changeNote tgn_rev:5011369370 , tgn_rev
            :5011369369 , tgn_rev:5019010426 , tgn_rev
            :5019010427 ;
22       dc-term:created "2010-10-16T13:23:22"^^xsd:dateTime ;
```

```
      dc-term:modified "2010-10-16T13:23:22"^^xsd:dateTime , "2014-06-04T11:15:03"^^xsd:dateTime ;
      gvp:parentString "Schleswig-Holstein, Deutschland, Europe, World" ;
      gvp:parentStringAbbrev "Schleswig-Holstein, Deutschland, ... World" ;
      gvp:displayOrder "7314"^^xsd:positiveInteger ;
      gvp:broaderPartitive tgn:7003688 ;
      gvp:broaderPreferred tgn:7003688 ;
      gvp:prefLabelGVP tgn_term:1001287042 ;
      dc-term:source tgn_source:2009007144-subject-7074121 ;
      gvp:placeTypePreferred aat:300000745 ;
      dc:identifier "7074121" ;
      dc-term:issued "2014-06-04T11:15:03"^^xsd:dateTime ;
      prov:wasGeneratedBy tgn_rev:5011369370 ;
      foaf:focus tgn:7074121-place .

tgn_rev:5011369370 a prov:Activity , prov:Create ;
      dc:type "created" ;
      prov:startedAtTime "2010-10-16"^^xsd:date;
      dc:description "new:Suchsdorf" .

tgn_term:1001287042 a skosxl:Label ;
      skosxl:literalForm "Suchsdorf" ;
      gvp:term "Suchsdorf" ;
      gvp:displayOrder "1"^^xsd:positiveInteger ;
      gvp:termFlag <http://vocab.getty.edu/term/flag/Vernacular> ;
      gvp:termPOS <http://vocab.getty.edu/term/POS/Noun> ;
      dc:identifier "1001287042" .

tgn_rel:7074121-placeType-300000745 a rdf:Statement ;
      rdf:subject tgn:7074121 ;
      rdf:predicate gvp:placeTypePreferred ;
      rdf:object aat:300000745 ;
      gvp:displayOrder "1"^^xsd:positiveInteger .
```

VI. Conclusion

There is a variety of different approaches on how to model historical administrative structures on the Semantic Web. In this article, ten 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 groups, 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. The representation of the administrative structure should be done by using object properties—otherwise the key factor of unique identification will be lost. Indication of time should be modelled as separate properties to make them machine-interpretable.

Both, source citations and revision history allow quality control—usually difficult on the Semantic Web.

To provide information on time, sources, and the language used, the use of reification seems inevitable in the current state of RDF and OWL. Reification in the sense of making statements about other RDF statements is problematic because OWL-2 DL does not support these "higher-order statements" and a formal modelling is difficult using `rdf:Statement` individuals.

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.

References

[1] J. Zedlitz and N. Luttenberger, "Modelling (historical) administrative information on the semantic web," in *WEB 2014, The Second International Conference on Building and Exploring Web Based Environments*, 2014, pp. 33–39.

[2] (2014) RDF 1.1 Concepts and Abstract Syntax. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/

[3] (2012) OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/TR/2012/REC-owl2-syntax-20121211/

[4] 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.

[5] 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.

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

[7] 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.

[8] A. Rula, M. Palmonari, A. Harth, S. Stadtmüller, and A. Maurino, "On the diversity and availability of temporal information in linked open data," in *International Semantic Web Conference*, 2012, pp. 492–507.

[9] 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

[10] 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

[11] P. Severi, J. Fiadeiro, and D. Ekserdjian, "Guiding reification in owl through aggregation," in *23rd International Workshop on Description Logics DL2010*, 2010, p. 408.

[12] M. Davis, "Tags for the Identification of Language (BCP47)," 2009.

[13] (2009) SKOS Simple Knowledge Organization System Reference. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/TR/2009/REC-skos-reference-20090818/

[14] (2014) RDF 1.1 Turtle: Terse RDF Triple Language. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/TR/2014/REC-turtle-20140225/

[15] (2014) The schema.org website. (last access: 2014-11-27). [Online]. Available: http://schema.org

[16] (2014) The dbpedia.org website. (last access: 2014-11-27). [Online]. Available: http://www.dbpedia.org

[17] 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.

[18] (2014) DBpedia Ontology version 2014. (last access: 2014-11-27). [Online]. Available: http://downloads.dbpedia.org/2014/dbpedia_2014.owl

[19] (2014) The GeoNames Website. (last access: 2014-11-27). [Online]. Available: http://www.geonames.org/

[20] (2009) WGS84 Geo Positioning: an RDF vocabulary. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/2003/01/geo/wgs84_pos

[21] (2014) The LinkedGeoData Website. (last access: 2014-11-27). [Online]. Available: http://linkedgeodata.org

[22] (2014) The OpenStreetMap Website. (last access: 2014-11-27). [Online]. Available: http://www.openstreetmap.org

[23] (2014) Ordnance Survey Linked Data. (last access: 2014-11-27). [Online]. Available: http://data.ordnancesurvey.co.uk/datasets/os-linked-data

[24] 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.

[25] 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.

[26] T. Riechert, U. Morgenstern, S. Auer, S. Tramp, and M. Martin, "Knowledge engineering for historians on the example of the catalogus professorum lipsiensis," in *The Semantic Web–ISWC 2010*. Springer, 2010, pp. 225–240.

[27] 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.

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

[29] (2014) The Finnish Spatio-temporal Ontology. (last access: 2014-11-27). [Online]. Available: http://onki.fi/en/browser/search/sapo?&os=sapo

[30] (2014) The GOV Website. (last access: 2014-11-27). [Online]. Available: http://gov.genealogy.net

[31] J. Zedlitz and T. Kluttig, "Das Genealogische Ortsverzeichsnis (GOV)," *Der Archivar*, vol. 3/2014, pp. 289–292, 2014.

[32] (2014) Getty Thesaurus of Geographic Names. (last access: 2014-11-27). [Online]. Available: http://www.getty.edu/research/tools/vocabularies/tgn/

[33] (2014) Getty Vocabularies: Linked Open Data – Semantic Representation. (last access: 2014-11-27). [Online]. Available: http://vocab.getty.edu/doc/gvp-lod.pdf

[34] (2013) PROV-Overview: An Overview of the PROV Family of Documents. (last access: 2014-11-27). [Online]. Available: http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/