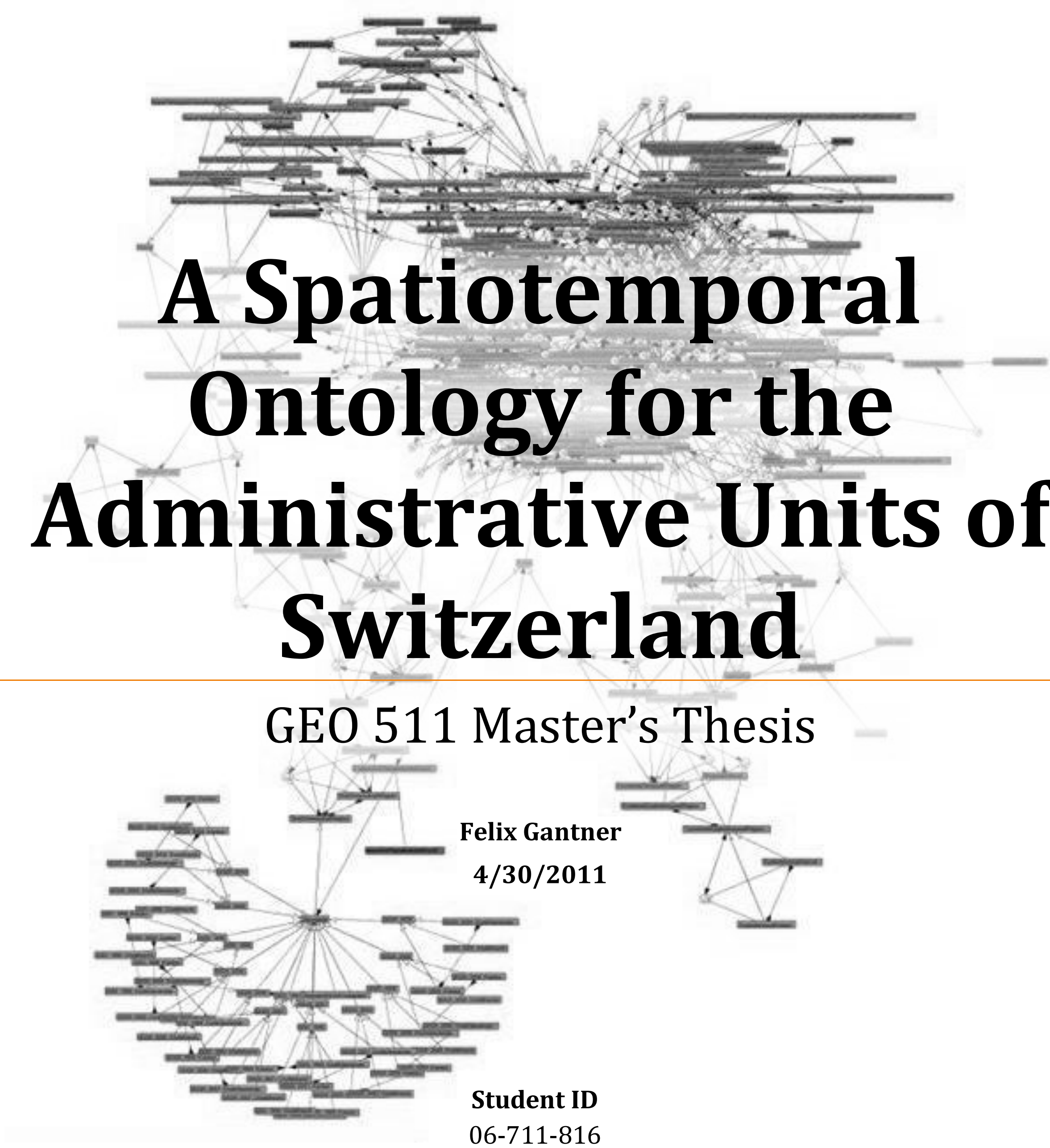
UNIVERSITY OF ZURICH, DEPT. OF GEOGRAPHY, GIS UNIT
IN COOPERATION WITH THE SWISS FEDERAL INSTITUTE FOR
FOREST, SNOW AND LANDSCAPE RESEARCH (WSL)



A Spatiotemporal Ontology for the Administrative Units of Switzerland

GEO 511 Master's Thesis

Felix Gantner

4/30/2011

Student ID
06-711-816

Supervisors at WSL
Dr. Bettina Waldvogel, Rolf Meile

Supervisor at UZH
Dr. Patrick Laube

Member of the Faculty
Prof. Dr. Robert Weibel

Contact
Kirchbühlstrasse 13, 5630 Muri AG, Switzerland
felix_gantner@hotmail.com

Cover picture

Visualization of the Spatiotemporal Ontology for the Administrative Units of Switzerland (SONADUS) with the Protégé plug-in SOVA (Kunowski 2010).

Acknowledgments

First of all, I would like to thank my supervisors Dr. Bettina Waldvogel, Rolf Meile, and Dr. Patrick Laube for their valuable advice, bright ideas, and numerous contributions.

Second, I wish to thank all other WSL staff members that helped me to complete this project. This particularly includes PD Dr. Rolf Grütter for various inputs on ontologies and proof-reading the thesis. Furthermore, I sincerely thank Dr. Thomas Scharrenbach for introducing me into Semantic Web software such as Pellet, Protégé, and Virtuoso.

Third, I am grateful to Dr. Tomi Kauppinen who explained SAPO to me at the GIScience Conference in September 2010.

Fourth, I would like to thank Elaine Schaefer for proof-reading my English writing.

Fifth, special thanks to my family for their moral and financial support.

Last, I want to thank my friends who also gave me great encouragement to pursue this study.

Summary

The evolution of the administrative units (AUs) of Switzerland has accelerated dramatically in the last 20 years. Between 1990 and 2010, the Federal Statistical Office (FSO) recorded a loss of 427 municipalities, a substantial decrease compared with the decline of 75 municipalities in the 30 years before. This trend implicates an elevated rate of spatial change and makes it difficult for the Swiss Federal Institute for Forest, Snow and Landscape Research (WSL) to correctly assign biotope objects to a municipality. Such an accurate reference is necessary because the cantons often entrust the municipalities with the responsibility to protect the biotopes within their boundaries. Hence, the location of a biotope has legal and financial consequences that may change as a result of boundary changes. Despite the relevance of spatial change, there is still a lack of an information system that supports queries on spatial change of municipalities. This stems from the fact that the thematic and the geometric data originate from different sources and do not match. Moreover, AUs can become unrecognizable over time because neither the Historicized Municipality Register (HMR+) nor the geometric datasets give AUs an identity.

In order to overcome these difficulties, this thesis set out to determine useful approaches to spatiotemporal databases and ontologies. Such approaches must facilitate the integration of different datasets and support queries on discrete change of objects with crisp boundaries. A review of the literature has found that ontologies fulfill these requirements best, in particular with respect to the integration of different datasets. Using an upper ontology as a common reference framework provides further advantages such as enhanced interoperability and knowledge sharing. While object-centered and event-oriented approaches to spatiotemporal databases focus on a world that comprises objects and events, respectively, the upper ontology Basic Formal Ontology (BFO) incorporates both views. BFO thereby provides a sound philosophical basis to formally define the entities of a domain and their mutual relationships. In contrast, traditional approaches fail to capture change because snapshots do not record phenomena that occur between two updates. Nevertheless, traditional approaches are still widely applied because their implementation is comparably easy and inexpensive. The opposite applies to formal ontologies of AUs of which only SAPO models a change history. Unlike many ontologies in the biomedical domain, however, SAPO neither extends BFO nor builds on any other upper ontology.

For the aforesaid reasons, it has been decided to construct a formal ontology that builds on BFO. The **S**patiotemporal **On**tology for the **Ad**ministrative **U**nits of **S**witzerland (SONADUS) traces the evolution of the AUs of Switzerland between 1960 and 2010. SONADUS endows AUs with an identity and takes account of the separate evolution of the HMR+ and the geometric datasets. Hence, SONADUS is able to assign each AU the geometries with which it temporally overlaps. Creating the knowledge base SONADUS-KB in a next step required extracting and converting the knowledge from a relational database into triples in RDF/XML format. The triplestore and SPARQL endpoint Virtuoso then served to efficiently store and query SONADUS-KB using the query language SPARQL. The evaluation of SONADUS-KB against predefined test queries showed that SONADUS represents a successful framework to query the change history of the AUs of Switzerland. Moreover, this work has confirmed that BFO is predestined to integrate the conceptualizations of different datasets. Comprising 241 classes and 167’843 individuals, however, SONADUS-KB has become comparably large and complex, mainly owing to the use of BFO. Consequently, this thesis not only proposes measures to alleviate this drawback, but also recommends further research to enhance query performance and to optimize the structure of ontologies.

Zusammenfassung

Der Wandel der Verwaltungseinheiten der Schweiz hat sich in den letzten 20 Jahren stark beschleunigt. Zwischen 1990 und 2010 hat das Bundesamt für Statistik eine Abnahme von 427 Gemeinden verzeichnet. Dies stellt ein starker Rückgang dar im Vergleich zum Verlust von 75 Gemeinden in den 30 Jahren zuvor. Die damit verbundenen Grenzänderungen erschweren es der Eidgenössischen Forschungsanstalt für Wald, Schnee und Landschaft (WSL), Biotopobjekte einer Gemeinde zuzuordnen. Eine genaue Zuordnung ist notwendig, da die Kantone oft den Gemeinden die Verantwortung zum Schutz der eigenen Biotope übertragen. Folglich hat die Lage eines Biotops rechtliche und finanzielle Auswirkungen, die durch Grenzänderungen beeinflusst werden können. Trotz gegebener Relevanz fehlt zur Zeit ein Informationssystem zur Abfrage von räumlichem Wandel von Verwaltungseinheiten. Das liegt daran, dass die thematischen und geometrischen Datensätze aus zwei verschiedenen Quellen stammen und nicht übereinstimmen. Zudem geben weder das Historisierte Gemeindeverzeichnis (HMR+) noch die geometrischen Datensätze den Verwaltungseinheiten eine Identität, so dass Verwaltungseinheiten über die Zeit unkenntlich werden können.

Um diese Probleme zu beheben, untersucht diese Arbeit bestehende Ansätze über raumzeitliche Datenbanken und Ontologien. Dabei müssen solche Ansätze sowohl die Integration unterschiedlicher Datensätze ermöglichen, als auch Abfragen über Objekte unterstützen, die abrupten Änderungen unterliegen und scharfe Grenzen haben. Das Literaturstudium hat aufgezeigt, dass Ontologien diese Anforderungen am besten erfüllen, speziell im Bezug auf die Integration verschiedener Datensätze. Beim Gebrauch einer allgemeinen Ontologie (upper ontology) ergeben sich zusätzliche Vorteile wie erhöhte Kompatibilität und vereinfachter Wissensaustausch. Während objekt- und ereignisorientierte Ansätze für raumzeitliche Datenbanken Objekte bzw. Ereignisse ins Zentrum stellen, vereint die allgemeine Ontologie Basic Formal Ontology (BFO) beide Sichten in sich. Dadurch verfügt die BFO über eine solide philosophische Grundlage, um die Einheiten eines Bereiches und deren Beziehungen untereinander zu definieren und formal zu beschreiben. Traditionelle Ansätze hingegen scheitern dabei, Änderungen festzuhalten, da sie Ereignisse zwischen zwei Updates nicht erfassen können. Dennoch finden traditionelle Ansätze immer noch verbreitet Anwendung, da deren Umsetzung vergleichsweise einfach und günstig ist. Das Gegenteil trifft auf Ontologien von Verwaltungseinheiten zu. Von denen modelliert nur die Raumzeitliche Ontologie Finnlands (SAPO) eine Änderungsgeschichte, wobei diese von keiner allgemeinen Ontologie Gebrauch macht.

Aus oben genannten Gründen wurde eine formale Ontologie erstellt, die auf der BFO aufbaut. Die so genannte Raumzeitliche Ontologie für die Verwaltungseinheiten der Schweiz (SONADUS) modelliert die Entwicklung der Verwaltungseinheiten der Schweiz zwischen 1960 und 2010. SONADUS gibt Verwaltungseinheiten eine Identität und berücksichtigt die eigenständige Entwicklung des HMR+ und der Geometrien. Daher kann SONADUS jeder Verwaltungseinheit diejenigen Geometrien zuordnen, mit denen sie sich zeitlich überschneidet. Bei der folgenden Erstellung der Wissensdatenbank (knowledge base) SONADUS-KB wurde das Wissen aus einer relationalen Datenbank abgefragt und in Triples im RDF/XML Format umgewandelt. Der Triplestore und SPARQL Endpoint Virtuoso diente danach zur effizienten Speicherung und Abfrage der Wissensdatenbank mit der Abfragesprache SPARQL. Diese Arbeit hat bestätigt, dass die BFO sich eignet, um die Konzeptualisierungen verschiedener Datensätze zu vereinen. Mit 241 Klassen und 167‘843 Instanzen ist SONADUS-KB vergleichsweise gross, was hauptsächlich vom Gebrauch der BFO herrührt. Diese Arbeit schlägt daher Massnahmen vor, um diese Nachteile zu verringern. Zugleich empfiehlt sie weitergehende Forschung zu betreiben, um die Abfrageeffizienz zu verbessern und die Struktur von Ontologien zu optimieren.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

Coalescing space and time has been a major challenge of GIScience over the past decades. Despite substantial progress in this field, there is still a lack of a comprehensive system that provides integrated functions to manage, analyze, and query spatiotemporal data (Yuan 2008). Since both the spatial and temporal dimension inhere in any real-world object (Gutiérrez et al. 2005), the need for a satisfactory solution will not cease to exist. That also applies to the evolution of the administrative units (AUs) of Switzerland which has accelerated in the last 20 years. The Historicized Municipality Register (HMR+) *(dt. Historisiertes Gemeindeverzeichnis der Schweiz)* of the Federal Statistical Office (FSO 2007b) records a loss of 427 municipalities between 01/01/1990 and 01/01/2010, a dramatic increase compared with the decline of 75 municipalities in the 30 years before. This trend implicates an elevated rate of spatial change, making it increasingly difficult to correctly reference and query objects in space and time.

Spatial change, as opposed to mere thematic change, directly affects the cognizance of an AU over a certain area. The Swiss Federal Institute for Forest, Snow and Landscape Research (WSL) maintains the Data Center Nature and Landscape (DNL) that collects spatial and thematic information about biotopes put under conservation by the Swiss Confederation (Bauer-Messmer et al. 2009). According to certain environmental indicators, the legislator along with a panel of experts defines a protective perimeter for each biotope. The Swiss law mandates the cantons with the protection of biotopes but these often entrust the municipalities with the responsibility, and thus the perimeter determines the respective municipality in charge (SR 451, Art. 18a+b). Hence, spatial change entails legal, political, and financial consequences, and therefore a complete documentation of current and historical states is of great interest for the parties involved.

1.2 Research context

The literature provides numerous approaches to store such knowledge of varying spatial and temporal dimensions. To facilitate the support of complex spatiotemporal queries, database research has yielded models that go beyond capturing change with regular time-slices or snapshots. Of these, the object-oriented approach (e.g., Worboys et al. 1990, Lohfink et al. 2007, Plumejeaud et al. 2009) and the event-oriented approach (e.g., Langran and Chrisman 1988, Peuquet and Duan 1995, Worboys 2005) have received the most attention. Both paradigms still prevail though Worboys (2005) and Yuan (2008) regard the event-oriented approach as being more suited for modeling dynamic spatiotemporal phenomena. This is because in contrast to the timestamping approaches, such as the snapshot model or the object-oriented paradigm, the event-oriented approach centers the basic entities of change, i.e., events and processes. However, one finds few implementations of both the object-oriented and the event-oriented approach. Gregory (2002), for example, presents five projects of historical GIS' that store the evolving pattern of AUs in different regions of Europe. None of them made use of a pure object-oriented or event-based model but relied on more simplistic approaches. The lack of experience with the rather sophisticated models increases the risk of unforeseeable results and expenditures and thus reduces the attractiveness of a practical implementation (Gregory 2002). Overall, database research has made considerable progress toward supporting spatiotemporal queries though the literature presents few approaches that found widespread use in practice.

In recent years, ontologies have exerted growing influence on database design and GIScience in general (Hornsby and Joshi 2010), as they provide an effective means to define the basic concepts and terms of a model (Agarwal 2005). Formal ontologies systematically describe the reality through a set of logical axioms (Guarino 1998). Ontologies thus offer enhanced interoperability and integration of different data sources (Agarwal 2005). For these reasons, ontologies play a key role in the Semantic Web where they facilitate interoperability by specifying a common vocabulary in a certain ontology language (Fu et al. 2005). The SPIRIT spatial search engine, for example, uses a place ontology for the disambiguation of place names from different gazetteers (Jones et al. 2002, Jones et al. 2004). Since most geographical ontologies still take the view of a static world, Grenon and Smith (2004) developed the Basic Formal Ontology (BFO), an upper ontology that accounts for both the static SNAP and the dynamic SPAN entities. On the level of application ontologies, Kauppinen et al. (2008) created an ontology time series that generates the different partition hierarchies of AUs in Finland between 1865 and 2007. By using an additional metadata schema of changes, the model can represent the complete change history of municipalities in Finland within that period. Despite being regarded as a successful application of the Semantic Web with respect to interoperability and information retrieval (Kauppinen et al. 2008), geographical ontologies become increasingly complex when integrating the temporal component (Yuan 2008).

1.3 Research gaps and shortcomings

The wealth of approaches in the field of database models and ontologies reflects the effort of this research area to develop a spatiotemporal information system with enhanced data management, analysis, and querying capabilities (Yuan 2008). The drawback of this diversity is that no generally accepted solution exists and that a thorough evaluation of all relevant approaches becomes a laborious task. Hence, the realization of such an information system for the AUs of Switzerland requires extensive research. Moreover, existing approaches concerning AUs have tended to focus on the implementation while not explicitly revealing their underlying conceptualization so far. There is thus a gap between research on general concepts and philosophical issues on the one hand, and on implementations bound to the actual data on the other hand. Despite widespread awareness of upper ontologies in the literature, no research has been found that implemented BFO in the geographic realm. For these reasons, a geographic information system is still lacking that 1) conforms with an upper ontology, 2) links the underlying conceptualizations of the source data with the upper ontology, and 3) supports spatiotemporal queries.

1.4 Research questions

Based on the present gaps in the research, this thesis examines the following research questions (RQs):

RQ1: Which approaches does the literature provide for an information system that supports queries about discrete change of spatial objects with crisp boundaries? Which of these approaches facilitate the integration of differing datasets?

RQ2: To what extent is SAPO (Kauppinen et al. 2008) applicable to the situation in Switzerland? In what way does the evolution of administrative units differ between Finland and Switzerland?

RQ3: Upper ontologies such as the Basic Formal Ontology (BFO) enhance interoperability and knowledge sharing with other ontologies (Noy 2004). Does BFO bring these advantages to an application ontology of administrative units? What are further advantages or disadvantages to such an application ontology that arise from using BFO?

RQ4: Having developed an information system that builds on an application ontology dealing with administrative units and extending BFO, what types of queries according to Yuan and McIntosh (2002) does such a system support?

RQ5: What are the challenges faced when building a conceptualization of the change history of the administrative units of Switzerland? What are the difficulties encountered when creating and querying a knowledge base based on this conceptualization?

1.5 Aims

To address the aforementioned research questions, this thesis has the following seven aims. In summary, the primary goal is to create an information system that builds on a knowledge base and supports spatiotemporal queries on the change history of administrative units.

Aim 1: Review the research conducted on spatiotemporal databases and ontologies.

Aim 2: Determine a suitable approach for an information system that supports queries about the evolution of the administrative units of Switzerland.

Aim 3: Build an application ontology for the administrative units of Switzerland that distinguishes between incremental and fundamental change, spatial and non-spatial change, as well as between spatial change and updates of the geometric data.

Aim 4: Link this application ontology with BFO to improve its interoperability and comparability as well as to determine the usefulness and practicability of an upper ontology.

Aim 5: Create a knowledge base that uses the application ontology as T-Box.

Aim 6: Identify the main obstacles for implementing an ontology and knowledge base of the change history of the administrative units of Switzerland.

Aim 7: Evaluate the information system against test queries to investigate the system's enhanced query capabilities.

1.6 Principal findings and products

This thesis presents SONADUS, a **S**patiotemporal **On**tology for the **Ad**ministrative **U**nits of **S**witzerland that builds on the upper ontology BFO. The application ontology SONADUS traces the evolution of the AUs of Switzerland between 1960 and 2010. In particular, SONADUS endows AUs with an identity and takes account of the separate evolution of the HMR+ and the geometric datasets. SONADUS is able to assign each AU the geometries with which it temporally overlaps. This work thereby confirms that BFO greatly facilitates the integration of different datasets. In addition to SONADUS, this thesis presents the knowledge base (SONADUS-KB). The triplestore and SPARQL endpoint Virtuoso serves to efficiently store and query SONADUS-KB using the query language SPARQL. The evaluation of SONADUS-KB against predefined test queries shows that SONADUS represents a successful framework to query the change history of the AUs of Switzerland. Comprising 241 classes, 43 properties, and 167'843 individuals, however, SONADUS-KB has become comparably large and complex, mainly owing to the use of BFO. Hence, this thesis not only proposes measures to alleviate this drawback, but also recommends further research to enhance query performance and to optimize the structure of ontologies.

1.7 Structure of the thesis

The remainder of this thesis has been organized as follows. Chapter 2 addresses the challenges that WSL faces when querying spatiotemporal changes of AUs. Based on that, this chapter also defines the use cases, that is, those types of queries that the information system must support. Chapter 3 provides the theoretical background to this thesis by considering research on databases, ontologies, and related subjects. Chapter 4 gives an overview of the subsequent four chapters that contain the methods applied and the results obtained of each step toward the realization of SONADUS-KB. Chapter 5 is thus concerned with the data analysis, Chapter 6 delineates the creation and structure of SONADUS, Chapter 7 explains the construction of the knowledge base, and Chapter 8 eventually describes its evaluation. In the following, Chapter 9 compares the research questions with the accomplishments of this research. This particularly includes a discussion of SONADUS and SONADUS-KB that reveals their benefits and shortcomings as an ontology and knowledge base, respectively. Chapter 10 finally summarizes the contributions of this thesis, highlights the conclusions drawn, and points at issues for further research. To improve the readability of this thesis, Appendix D lists the abbreviations used in the text. A table providing a complete overview of the abbreviations is to be found in the pocket on the back cover.

2 Current situation at WSL

2.1 Challenges faced by WSL

The Swiss Federal Institute for Forest, Snow and Landscape Research (WSL) maintains the Data Center Nature and Landscape (DNL). The DNL is a database that contains approximately 6'000 spatially and temporally referenced landscape and biotope objects in Switzerland (FOEN 2010). According to predefined environmental indicators, the legislator, consulted by an expert panel, defines the biotopes to be put under conservation and determines the corresponding protective perimeter. The Swiss law mandates the cantons with the conservation of biotopes that, however, often entrust the municipalities with that responsibility. The Federal Office for the Environment (FOEN) and private contractors regularly compile inventories of these biotopes to record border revisions proposed by public or private parties involved. WSL then stores these inventories on behalf of the FOEN. An inventory of a biotope always refers to a specific date and lists the municipalities in charge for its conservation at that time. Changes of administrative units (AUs) may cause a change of responsibility for a biotope. For example, at the annexation of Rohr (AG), Aarau inherited the responsibility of Rohr (AG) to conserve the biotopes in this area. In other cases, the correct identification of the responsible municipality can be difficult, even though the responsibility itself has not changed. Since the identification is made by the name or the FSO number of an AU, a change of one of the two may cause that a corresponding query returns no or false results. For these reasons, WSL requires an information system that supports queries about spatial and thematic change of AUs.

2.2 Test queries

Based on the current situation at WSL and research on spatiotemporal queries, a set of test queries was formulated that the information system must support. The evaluation eventually reveals whether the system is able to successfully answer these queries. The classification of the queries conforms to the typology of Yuan and McIntosh (2002) that Chapter 3.2 examines in detail. Note that certain query types of this typology also comprise more complex queries than those defined below. First, this is because AUs have crisp boundaries and undergo discrete change. Second, WSL does not require information about overlap ratios, probability calculations, or comparisons of patterns.

2.2.1 Attribute queries

Case 1: What is the name of the municipality with the FSO number 4001?

2.2.2 Spatial queries

Simple spatial query
Case 2: Where is the Municipality of Birmensdorf (ZH)?

Spatial range query
Case 3: What are the names of the municipalities of the District of Baden?

Spatial relationship query
Case 4: Where are the biotopes that are inside the boundaries of the City of Rapperswil-Jona?

2.2.3 Temporal queries

Simple temporal queries

Case 5: Which municipalities were abolished on December 31, 2009?

Case 6: Which municipality was responsible for the conservation of the bog of Johannisberg on January 3, 2011?

Case 7: When did the City of Rapperswil and the Municipality of Jona merge?

Temporal range query

Case 8: Has the Municipality of Birmensdorf (ZH) undergone change since 1960?

Temporal relationship query

Case 9: How many changes did the Municipality of Berg (TG) undergo after the annexation of Andhausen?

2.2.4 Spatiotemporal queries

Simple spatiotemporal query

Case 10: What were the borders of Berg (TG) on January 1, 1996?

Spatiotemporal range query

Case 11: Which municipalities existing on 01/01/1960 have been abolished by 12/31/2009 in the District of Baden?

Spatiotemporal behavior query

Case 12: Which mergers in the Canton of Ticino led to the abolition of more than two municipalities and occurred in the 2000s?

Spatiotemporal relationship query

The spatiotemporal relationship query represents the most complex query type (Yuan and McIntosh 2002) and is used for analyses and predictions. The envisaged information system does not have to support this type of query because there is no need for information that would require such computations.

3 Background

3.1 Broader research context

A comprehensive system to store, query, analyze, and manipulate spatiotemporal data is still lacking (Yuan 2008). Nonetheless, research in information science has achieved significant progress since the 1980s. This chapter reviews relevant work done on spatiotemporal databases and ontologies while also considering implementations involving administrative units (AUs). Due to the large amount of research, a complete review would go beyond the scope of this thesis that thus concentrates on the three predominant approaches. First of all, there is the traditional approach that either only stores the current state or takes snapshots at particular points in time (Abraham and Roddick 1999). Consequently, the detection of change requires the comparison of two snapshots or time-slices (Worboys 2005). Second, the object-oriented paradigm holds a view in which the world comprises objects that maintain an identity and can be described by attributes and behaviors (Worboys 2001). From a philosophical perspective, these two approaches comply with the SNAP ontology of Grenon and Smith (2004). Its counterpart, the SPAN ontology, corresponds philosophically with the event-oriented approach. This third category of approaches takes the view that the world consists of dynamic entities such as events and processes.

Since these three predominant approaches only provide a general framework, a variety of models have been proposed within each paradigm. Most of them extend the relational database model though some also build on alternative approaches such as the object-oriented database model. Yet, this does not automatically mean that the approaches proposed depend on one of either database models. Yuan's (1999) Three-Domain Model, for example, can be implemented in both a relational and object-oriented database. Even object-oriented approaches may be implemented into a relational database (Choi et al. 2008). With regard to modeling change of AUs, it can be observed that traditional approaches still find wide application because their implementation is comparably easy and inexpensive (Gregory 2002). Database solutions have also been deployed since the 1980s (e.g., Schreven 2000) whereas ontologies and knowledge bases have only recently been discovered as a means to store a change history of administrative units (Kauppinen et al. 2008).

This chapter has been divided into four parts. The first part gives an overview of the different types of queries and addresses the significance of efficient query processing of information systems. The second part presents the three kinds of temporal attributes in a database before describing approaches that integrate space and time into a database. Then, the third part focuses on ontologies and related technologies such as knowledge bases, the Semantic Web, and knowledge representation languages. The fourth part finally deals with implementations of spatiotemporal information systems that particularly concern the changes of administrative units.

3.2 Queries

The ability to process queries represents an elementary function of an information system (Yuan 1999). The types of queries that such a system can support depends on its design and structure. It is therefore critical to ascertain in advance what kind of information the prospective users aim to retrieve. This is last but not least due to the fact that inexperienced database users fail to extract the desired information if that requires the formulation of complex queries (Yuan and McIntosh 2002). Moreover, successful querying of spatiotemporal information presupposes that a database or knowledge base stores such information explicitly (Jiang and Worboys 2009). Another issue is the response time to a query that must remain moderate to maintain the efficiency of an information system (Kemp and Kowalczyk 1994). Due to the demanding requirements, further research is needed to create such a system that can process spatiotemporal queries (Yuan 2008).

Yuan and McIntosh (2002) differentiate between four main types of queries. Their classification corresponds with the one of Claramunt and Thériault (1995) and partly extends the taxonomy of Langran (1992). Note that the type of certain queries depends on the underlying data model. Retrieving explicitly stored information does not require the same query types as extracting implicitly stored knowledge. Furthermore, queries that begin with *where* always ask for a geometry.

3.2.1 Attribute queries

Attribute queries (Yuan and McIntosh 2002) or *thematic queries* (Claramunt and Thériault 1995) are the most simple query type as they only involve the extraction of explicitly stored information. Although temporal or spatial attributes may be concerned, this query type does not include computation.

Example
Attribute query: Which Swiss cantons have a population of more than 500'000 inhabitants?

3.2.2 Spatial queries

Spatial queries can be subdivided into simple spatial, spatial range, and spatial relationship queries. *Simple spatial queries* ask for spatial entities that meet the criteria specified in the query. Although this query type usually does not require spatial computation, it can be necessary if the database does not contain the requested geometries. *Spatial range queries* retrieve information about features or attributes within a spatial entity or user-defined boundaries. Finally, *spatial relationship queries* seek features that share a relationship of proximity or topology. In contrast to simple spatial queries, both spatial range and spatial relationship queries typically involve spatial computation (Yuan and McIntosh 2002).

Examples
Simple spatial query: Where are the Swiss cantons with fewer than 100'000 inhabitants?

Spatial range query: How many biotopes are within the City of Zurich?

Spatial relationship query: What are the neighboring municipalities of Birmensdorf (ZH)?

3.2.3 Temporal queries

Just as spatial queries, *temporal queries* consist of three subtypes, i.e., simple temporal, temporal range, and temporal relationship queries. While *simple temporal queries* retrieve a time-slice or snapshot of a certain time, *temporal range queries* search for the changes that a certain feature has undergone in a given interval. For both subtypes, the desired time or interval may not correspond to the time of the data collection which can make interpolations necessary. The simple temporal and the temporal range query are identical to Langran's (1992) temporal queries of the same name. *Temporal relationship queries*, finally, inquire how features and phenomena are temporally interrelated whereby proximity and topology relationships are of interest. Therefore, one may examine, for example, if two environmental phenomena coincide or if two events happen simultaneously or successively (Yuan and McIntosh 2002).

Examples

Simple temporal query: How many people lived in the City of Zurich on December 31, 2008?

Temporal range query: How many times did the City of Zurich have a budget deficit between 1990 and 2010?

Temporal relationship query: How does the number of municipality mergers correlate with the population development?

3.2.4 Spatiotemporal queries

While spatial and temporal queries focus on either the spatial or the temporal dimension, *spatiotemporal queries* search for answers in both, though one of these two dimensions may dominate. Spatiotemporal queries can be subdivided into simple spatiotemporal, spatiotemporal range, spatiotemporal behavior, and spatiotemporal relationship queries. *Simple spatiotemporal queries* ask for the location of an object at a given time, the existence of an object at a given location, or the situation at a given place to a given time. This query type may depend on spatial or temporal interpolation if the data does not exactly meet the specifications of the query. *Spatiotemporal range queries* inquire about the evolution of a spatial entity within a certain timeframe. Analogous to the temporal queries, the classification of Langran (1992) includes both the simple spatiotemporal and the spatiotemporal range query. Lastly, *spatiotemporal behavior queries* and *spatiotemporal relationship queries* represent the most challenging types of queries (Yuan 2008). At the same time, they create excellent opportunities to gain new insights into the data (Yuan and McIntosh 2002). The former aim to detect the evolution of patterns as well as to examine attribute change in the course of time. Spatiotemporal behavior queries thus allow users to draw conclusions about the agents of change. Spatiotemporal relationship queries, on the contrary, retrieve information about spatial and temporal relationships between entities of various kinds. They therefore represent the most complex query type and provide the means to use the full advantages of a spatiotemporal information system (Yuan and McIntosh 2002). However, such full-scale spatiotemporal information systems have not been implemented yet (Yuan 2008).

Examples

Simple spatiotemporal query: Where were the Swiss cities with more than 50’000 inhabitants in 2010?

Spatiotemporal range query: Where in Switzerland were wetlands placed under protection between 1990 and 2010?

Spatiotemporal behavior query: In which canton has the proportion of municipality mergers been highest since 1990?

Spatiotemporal relationship query: How does the pattern of municipality mergers correlate with the pattern of migration?

Spatiotemporal relationship query: Where and when in the next ten years will the population growth be higher than the national average?

3.3 Spatiotemporal databases

3.3.1 Temporal attributes in databases

A database supports up to three types of temporal attributes. Whereas *static* systems do not store any temporal information at all, *bitemporal* databases contain both transaction and valid time (Worboys

1994). In contrast, *historical* databases store only valid time and *rollback* systems restrict themselves to supporting transaction time (Worboys 1994). In certain cases, it makes sense to maintain an additional user-defined time. Since this requires the maintenance of valid time, only bitemporal and historical databases can implement a user-defined time (Snodgrass and Ahn 1985).

Transaction time

Transaction time (Pelekis et al. 2005), also *physical time* (Bubenko 1977) or *database time* (Langran and Chrisman 1988) corresponds to the time when a value or an entry is recorded in a database (Snodgrass and Ahn 1985, Langran and Chrisman 1988). When recording data, the transaction time is fixed and cannot be changed anymore (Snodgrass and Ahn 1985). Hence, the transaction time plays an important role for backtracking of previous states in systems with multi-user access (Wachowicz and Healey 1994).

Valid time

Valid time (Pelekis et al. 2005), also *logical time* (Bubenko 1977), *world time* (Langran and Chrisman 1988), or *event time* (Worboys 1994) corresponds to the time when the actual event takes place (Snodgrass and Ahn 1985, Langran and Chrisman 1988). In contrast to the transaction time, the valid time may be changed after an event was initially recorded in the database. This can be due to new insights about historical events or the detection of incorrect entries (Snodgrass and Ahn 1985).

User-defined time

A database may be amended with a *user-defined time* if transaction and valid time do not suffice to express the desired temporal information (Snodgrass and Ahn 1985).

Example

In case of a municipality merger, the date of the plebiscite could be stored as a user-defined time. The date when the merger came into force may be defined as the valid time. The transaction time consequently corresponds with the date when the merger was entered into the database (Snodgrass and Ahn 1985).

3.3.2 Traditional approaches

General characteristics

Traditional approaches build on the aspatial relational database model and are thus considered as being state-dominant instead of space-dominant (Langran 1992). Such databases either maintain an updated version of the current state or repeatedly take snapshots of the reality at particular points in time (Abraham and Roddick 1999). Whereas in the first case historical information is not retained at all, in the second one information is only provided about states at the snapshot times. Traditional approaches are thus able to capture discrete change, provided that snapshots are available for each state. Continuous change, on the contrary, also occurs between two recorded states, and therefore snapshots fail to track many geographic phenomena. In order to avoid overlooking such continuous processes, it is crucial to choose appropriate intervals for taking snapshots. The literature refers to this as granularity that in this context denotes the level of detail applied to study spatiotemporal processes (Hornsby and Egenhofer 2002).

Static databases

Static databases store only the present situation (Snodgrass and Ahn 1985). As soon as a change occurs, the current state is updated and overwritten, leading to a deletion of all historical information. For example, a database containing the names and the polygons of the Swiss municipalities would only store the most recent state. This makes it impossible to query a static database by historical municipality names or to compare different temporal states (Snodgrass and Ahn 1985, Worboys 2005).

Static rollback databases

Static rollback databases (rollback = time-slice) store a sequence of time-indexed static relations that represent states in the past (Armstrong 1988). Each relation contains the entire information at the recording time. Hence, this approach produces considerable redundancies because of the repetitive storage of unchanged data. Since rollbacks or time-slices are only indexed by the transaction time (Snodgrass and Ahn 1985), it is impossible to retrieve the time when an event occurred or a time-slice was recorded. The fact that errors cannot be removed afterwards is another drawback to this approach. For these reasons and, perhaps, due to its similarity with the snapshot model, static rollbacks have not found much attention in the recent literature.

Historical databases

Instead of transaction time, historical databases support valid time (Snodgrass and Ahn 1985). This allows to retroactively change erroneous entries but makes it impossible to keep track of the changes in the database. Since a modification of the valid time results in the loss of the previous value, historical databases resemble static databases with respect to the deletion of overwritten information. In contrast, historical databases may maintain a user-defined time if additional temporal information is necessary.

Snapshot approach

The snapshot model is the most commonly used approach to integrate space and time into a database (Worboys 2005). The idea to extend two-dimensional matrices to three-dimensional time series was first conceived by Berry (1964). Snapshots must not be confused with static rollbacks that also represent time-slices (Armstrong 1988). While the static rollback approach supports only transaction time, the snapshot model maintains both transaction and valid time. Systems that conform to the snapshot approach are thus referred to as bitemporal databases. As static rollback and historical databases, the snapshot model creates a new version of the entire dataset in case of an update (see Figure 3.1). This makes the snapshot approach very simple and easy to implement (Pelekis et al. 2005). On the contrary, the repetitive storage of all information leads to massive redundancies. A snapshot records everything regardless of whether an element has undergone change. As snapshots capture change only implicitly, the detection of change requires the comparison of multiple snapshots (Worboys 2005). Yuan (1999) therefore points out that the processing of complex spatiotemporal queries is only manageable as long as the number of snapshots does not surpass fifty layers. In contrast, Stock (2006) affirms that snapshots suit many real-world applications, even though being a relic of a static worldview that has its origins in printed maps (Grenon and Smith 2004).

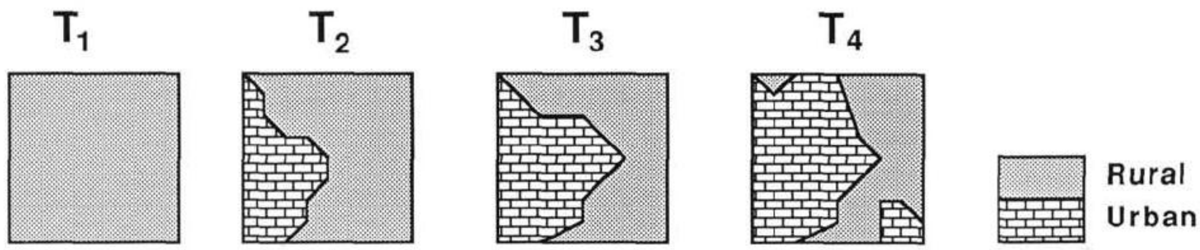


Figure 3.1: Snapshots representing urban expansion into a rural area. Intervals between snapshots are not necessarily equal (Langran and Chrisman 1988: 8)

Space-time composite

The space-time composite (Figure 3.2) consists of a base map composed of multiple objects that are characterized by a single attribute (Langran and Chrisman 1988). Each object has an identifier and a list with its own distinct attribute history. Every time a change occurs, an object is split into two or more parts that have to be assigned new identifiers and attributes. In order to maintain the consistency of the database, the identifiers of the old objects have to be adapted too. Objects that have ceased to exist are thus not omitted but retained in the attribute history and appended with a valid time interval. This process leads to an increasing fragmentation of the mosaic of spatiotemporal objects over time. Moreover, adjoining objects with identical attributes remain separate because the model disallows mergers of objects.

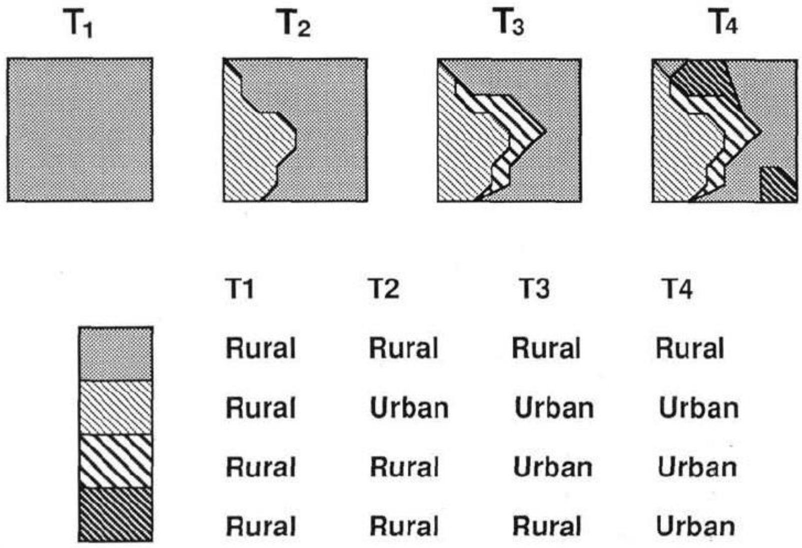


Figure 3.2: A space-time composite of urban encroachment (Langran and Chrisman 1988: 12)

The space-time composite makes it easy to restore a past state (Langran and Chrisman 1988). One has to search the history of every object and retrieve the appropriate attribute of the required time-slice. If two adjoining objects share the same attribute, the dividing line drops out. However, a major shortcoming of the space-time composite is the restriction to one attribute per object (Yuan 1999). An object might be a forest, a meadow, a municipality, or an orchard, but it cannot be a municipality with a population density and an unemployment rate. Moreover, the model is unable to store objects that consist of multiple polygons. Each spatial entity and its attribute history have to be stored separately which leads to the accumulation of redundancies. Last, the separate storage of every polygon makes processes occurring on multiple objects more difficult to capture.

3.3.3 Object-oriented approaches

General characteristics

Object-oriented approaches for modeling spatiotemporal changes build on models of object databases (Atkinson et al. 1989, Abraham and Roddick 1999). In the literature, one can find a plethora of object-oriented approaches that range from the beginnings in the early 1990s (Worboys et al. 1990) to present (e.g., Plumejeaud et al. 2009). The object-centered approach regards natural phenomena and man-made features as distinct objects that are characterized by thematic, topological, and geometric properties (Worboys 2001). These properties express the nature and behavior of an object which is reflected in its attributes and operators, respectively. Objects with the same types of attributes and operators together form an object class wherein each object represents a separate instance and features a unique and unchangeable identifier. Furthermore, the object-based paradigm has three essential characteristics, namely inheritance (Worboys 2001, Lohfink et al. 2007), composition (Worboys 2001), and encapsulation (Lohfink et al. 2007). First, inheritance requires a hierarchical class structure in which a subclass inherits all properties from its superclass. In addition, subclasses may also have their own attributes and operations (Lohfink et al. 2007). Second, composition denotes the formation of superclasses that comprise multiple subclasses (Worboys 2001). Third, encapsulation isolates object-specific properties from other objects to maintain the reusability of an object (Lohfink et al. 2007).

Versioning

As time elapses, objects undergo change which makes updates necessary. In the object-oriented approach, updates create new object versions that can be ordered hierarchically for the past, present and future (Wachowicz and Healey 1994). Basically, there are two alternatives to implement versioning. The first one timestamps each attribute separately and therefore duplicates only those attributes that have changed. The second one generates a new object instance with a unique identifier as soon as an object has changed (Lohfink et al. 2007). Becker et al. (1996) implemented the first alternative in the Geo Object-Oriented Database Core (GOODAC) for the realization of the Object-Oriented Geo-Data Model (OOGDM). OOGDM timestamps each attribute individually from which implicit object timestamps can be derived. The individual versioning of attributes thereby reduces redundancies because updates resulting from minor changes do not require the duplication of the entire object.

Geospatial Event Model (GEM)

GEM is a hybrid approach that integrates events into the object-oriented Geospatial Object Model (GOM) (Worboys and Hornsby 2004). The conceptualization of GEM rests on the Basic Formal Ontology (BFO) (Grenon and Smith 2004) that distinguishes between the two basic entities continuants and occurrents (see Chapter 3.4.4). GEM uses the terms object and event to refer to these two concepts and additionally considers settings as a third category of entities. Both objects and events are located in a setting that is either purely spatial, purely temporal, or mixed spatiotemporal. Settings have the same properties as objects and events, except that their identity ceases to exist if some attributes change. Based on Grenon and Smith (2004), Worboys and Hornsby (2004) identified several kinds of geospatial object-event and geospatial event-event relationships. On the one hand, a geospatial event-event relationship is given if the starting signal initiates a marathon. In object-event relationships, on the other hand, events either involve objects or objects participate in events. The former is the case if a plebiscite approves the merger of two municipalities. The latter applies if voters cast their ballots in a plebiscite. Depending on the event, both participations and involvements can be either optional or mandatory. Finally, Worboys and Hornsby (2004) implemented GEM in a case study using an extension of the Unified Modeling Language (UML). This extension differentiates between class diagrams for objects and events. Despite being differently shaped, both kinds of diagrams have the same structure that specifies the class name, the settings, attributes, and operations for a class.

Figure 3.3 depicts GEM where the gray part represents the temporal extension of the purely object-oriented GOM.

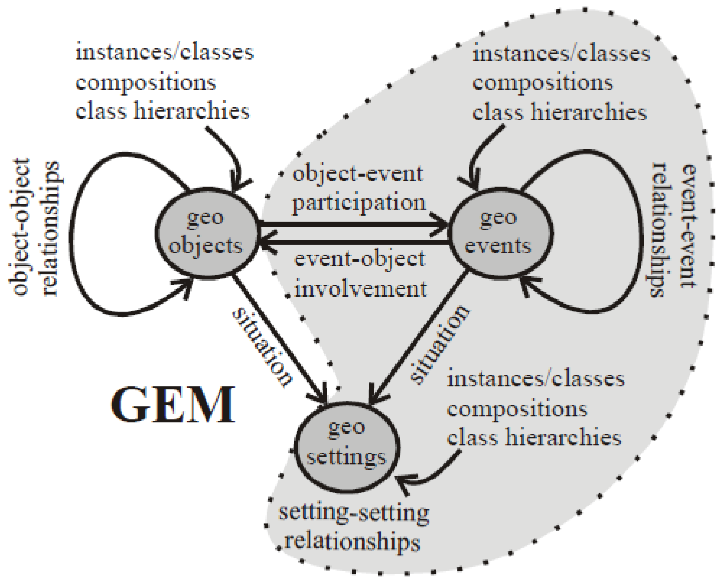


Figure 3.3: The Geospatial Event Model: objects, events and their interaction (Worboys and Hornsby 2004: 332)

Feature Evolution Model (FEM)

Lohfink et al. (2007) presented FEM, another object-oriented approach that refers to features as classifiable entities or areas, e.g., forestation or municipalities. FEM consists of only six object types: evolved feature, feature state, spatial object state, state descriptor, occurrent, and transition. Each of these object types represents a class that may be additionally subsumed by other classes. The evolved feature class stores the history of evolving objects by counting the number of states as well as the evolution path of a feature. In addition, the state descriptor class relates a state with the according transition object that has led to this state. The classes *feature state* and *spatial object state* record versions of the thematic and spatial attributes, respectively. While a single occurrent establishes a connection between states, the aggregation of multiple occurrents represents a transition of a feature from one state to another. FEM thus proves to be capable of processing the four types of temporal and spatiotemporal queries of Langran (1992). Furthermore, Lohfink et al. (2007) applied FEM to evolving transport network data and also took advantage of inheritance that is a prerequisite for polymorphic assignment. This provides more flexibility as polymorphic assignment means that a reference can point to any object of an inheritance hierarchy without specifying the class in it.

Figure 3.4 illustrates the relationships between the top-level object types of FEM.

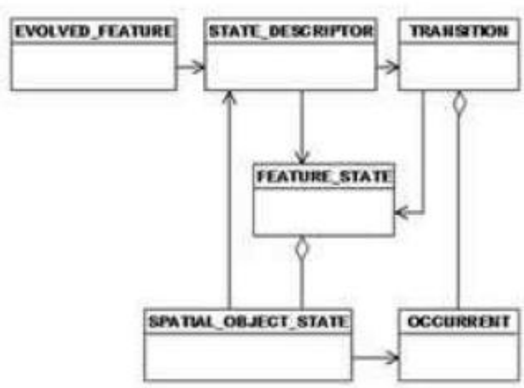


Figure 3.4: UML class diagram showing associations between FEM types (Lohfink et al. 2007: 5)

Framework for Evolutive SpatioTemporal Information (ESTI)

Plumejeaud et al. (2009) created ESTI to enhance the comparability of statistical data between different European countries. More precisely, ESTI aims to facilitate the standardization of heterogeneous geostatistical data. Such data are, for example, unemployment rates calculated at multiple hierarchical levels in different regions and at different points in time. However, the integration of different datasets is not a straightforward task. First, the geographic units (GUs) and spatial partition for the data collection vary depending on the variable. Second, the official EU definition of variables, such as unemployment, may differ from the definitions of the member states. Third, the composition of GUs as well as their names change over time.

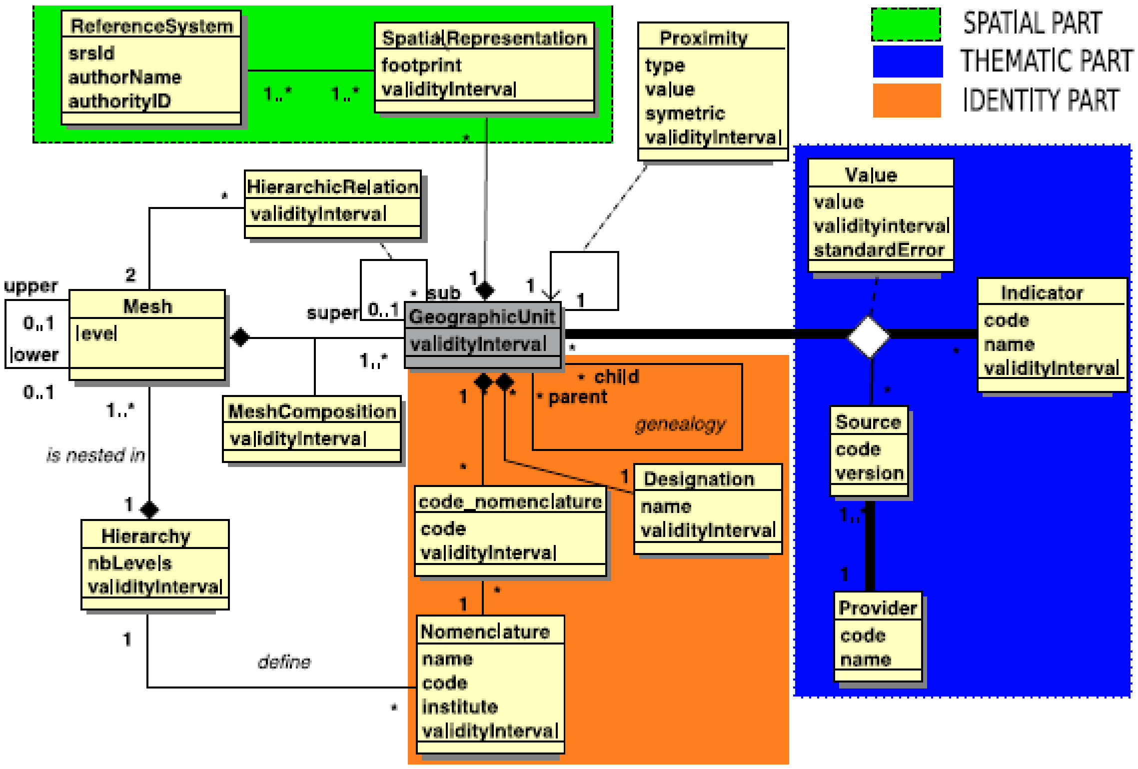


Figure 3.5: The object-oriented model overview, centered on the GU (Plumejeaud et al. 2009: 175)

For these reasons, ESTI adopts an object-oriented approach that considers GUs as basic entities with an identity, thematic properties, and a spatial extension (see Figure 3.5). The spatial part stores the geometries of GUs in a separate class. This allows the creation of multiple versions of the same GU representing different states. The same applies to the thematic part that records timestamped versions of statistical data. The identity part, finally, stores a unique identifier for every GU as well as the parent-child-relationships that hold between preceding and succeeding GUs. Aside from that, ESTI records the hierarchical and proximity relations that bear between two GUs. Moreover, each hierarchical level represents a mesh, that is, the composition of GUs at a given time. ESTI therefore records the mesh that a GU is part of, and its validity interval. Note that ESTI allows GUs to be part of more than a single hierarchical level as certain GUs exist on multiple levels. San Francisco, for example, is both a city and a county. Furthermore, ESTI supports queries to estimate missing values but fails to successfully manage metadata at the same time. Plumejeaud et al. (2009) thus proposed to put a metadata and ontology layer on top of the actual database.

3.3.4 Event-oriented approaches

General characteristics

In contrast to traditional and object-oriented approaches, event-oriented approaches store changes explicitly. In its most simple form, an event-oriented model consists of a base map that depicts the initial state which is amended by subsequent changes recorded in a transaction log. Due to the explicit storage of events, event-oriented approaches facilitate the formulation of queries to retrieve information about changes and to predict future conditions. Worboys (2005) thus considers event-oriented models as being superior to object-oriented approaches with respect to modeling geographic processes and phenomena.

Base map with overlays

The base map with overlays (Langran and Chrisman 1988), or base state with amendments (Langran 1992), represents the most basic form of the event-oriented models, and is essentially a vector-based approach (Peuquet and Duan 1995). At the time of its creation, the database captures and stores a base state (Langran and Chrisman 1988, Langran 1992). In contrast to snapshots that produce new versions of the entire dataset over and over again, amendment layers store only the changes that occurred between two updates. The base state itself is never subject to modifications and remains unchanged. The current state can be determined by putting all overlays on top of one another. The same applies to the retrieval of past states, even though those overlays have to be excluded that have been added afterward. Although a base map with overlays stores changes explicitly and produces minimal redundancies, the approach suffers from similar limitations than the snapshot model. The higher the number of amendments, the more layers must be compared which finally results in longer response times. In addition, confusion may arise because the temporal sequence of overlays does not necessarily correspond with the order of the valid time of the actual changes.

Figure 3.6 depicts a simple example of a base map with overlays.

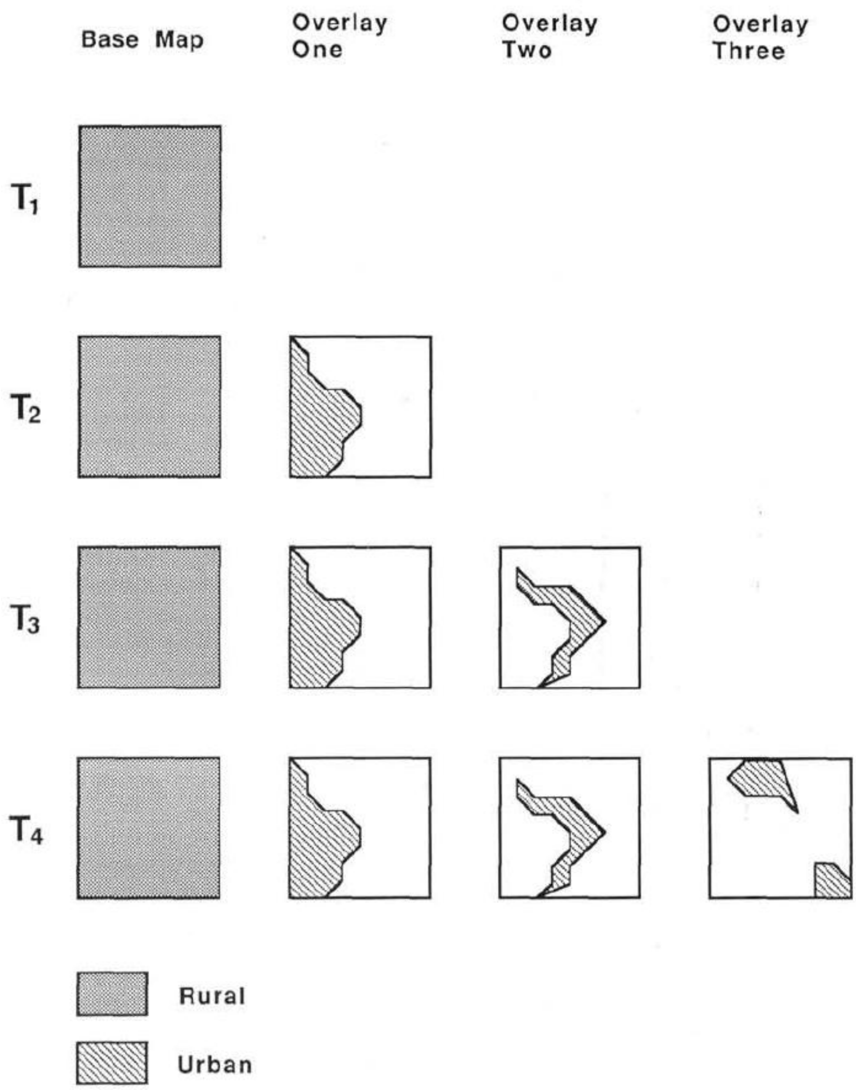


Figure 3.6: A base map with three amendments representing urban encroachment into a rural area (Langran and Chrisman 1988: 10)

Event-based Spatio Temporal Data Model (ESTDM)

ESTDM (Peuquet and Duan 1995) resembles the base map with overlays in broad terms but adopts the raster model instead of the vector representational approach (Abraham and Roddick 1999). Thus, ESTDM avoids the complexity of spatial topology in the vector model. The structure of ESTDM consists of a header, an event list, and a base map that depicts the initial state of the investigated area (Peuquet and Duan 1995). Besides pointing to the base map and the event list, the header contains the name of the base map, the date of its initial state, and the subject of the map. The event list stores all events that occur to an attribute in ascending order, including a timestamp and pointers to the previous and next event, respectively. Each event entry refers to the affected cells by declaring their coordinates and new attribute values. To achieve a reduction in storage costs, ESTDM forms clusters of cells termed components. Components collect those cells that change to the same value at an event. At each event, the storage of any value thus occurs only once.

Figure 3.7 illustrates the structure of ESTDM.

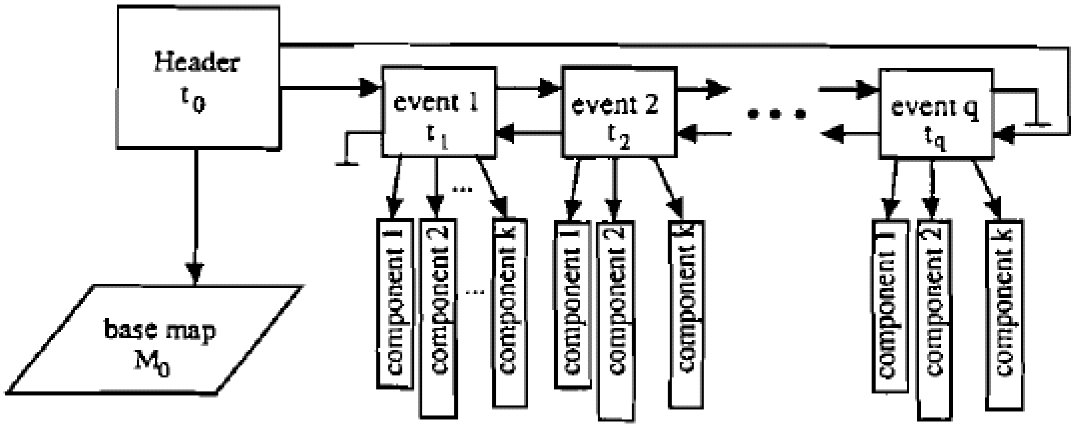


Figure 3.7: ESTDM showing all primary elements and the pointer structure (Peuquet and Duan 1995: 15)

The temporal ordering of events facilitates comparison of sequences of different thematic domains. Moreover, the explicit storage of events makes it obvious where spatial change occurs and does not require comparing two snapshot images. ESTDM also reduces storage costs if the variability of the phenomenon remains moderate. If no cell values can be grouped to components, the advantages of reduced storage over the snapshot model get lost. Ratio scaled variables such as temperatures are suited to field representations and may adopt a theoretically infinite number of values. Since these values also vary spatially and temporally, components can be scarcely formed, and therefore ratio scaled variables are not ideal for ESTDM. Categorical variables, such as land use, are described by a finite number of classes, cf., the areal statistics of Switzerland. This makes the formation of groups easier and leads to a reduction in storage. The same effect can be achieved with ratio scaled variables if they are subdivided into a number of intervals.

Process model of Worboys (2005)

The event-oriented approach of Worboys (2005) takes a view from which the world consists of processes. This view conceptually complies with an ontology of occurrents, e.g., the SPAN side of BFO (Grenon and Smith 2004) (see Chapters 3.4.4 and 3.4.5). Worboys (2005) defines time as a linear and finite sequence of instants referred to as ticks. Each tick has a channel named *tocc* that may contain temporally referenced entities. The existence of such entities begins and ends with their first and last reference in a *tocc* channel. The process-based view also includes space that is by definition a connected region composed of smaller units, termed locations. These locations are represented as processes that are connected with neighboring locations by two inversely directed adjacency relations. Just as a temporal sequence contains a finite number of ticks, a region comprises a limited number of location processes. In addition, there are *socc* channels that establish a connection between locations and their occupying entities and represent the spatial equivalent to the temporal *tocc* channels. While spatial entities have to maintain a relationship with at least one region through a *socc* channel, spatiotemporal entities occupy both a *socc* and a *tocc* channel. Unlike other models, this approach is described using process calculi that are suited to model the interaction of multiple processes. As a means to illustrate the conceptual framework, Worboys (2005) finally presents a case study in which a moving vehicle passes through three locations over a period of three ticks (Veh = vehicle in Figure 3.8).

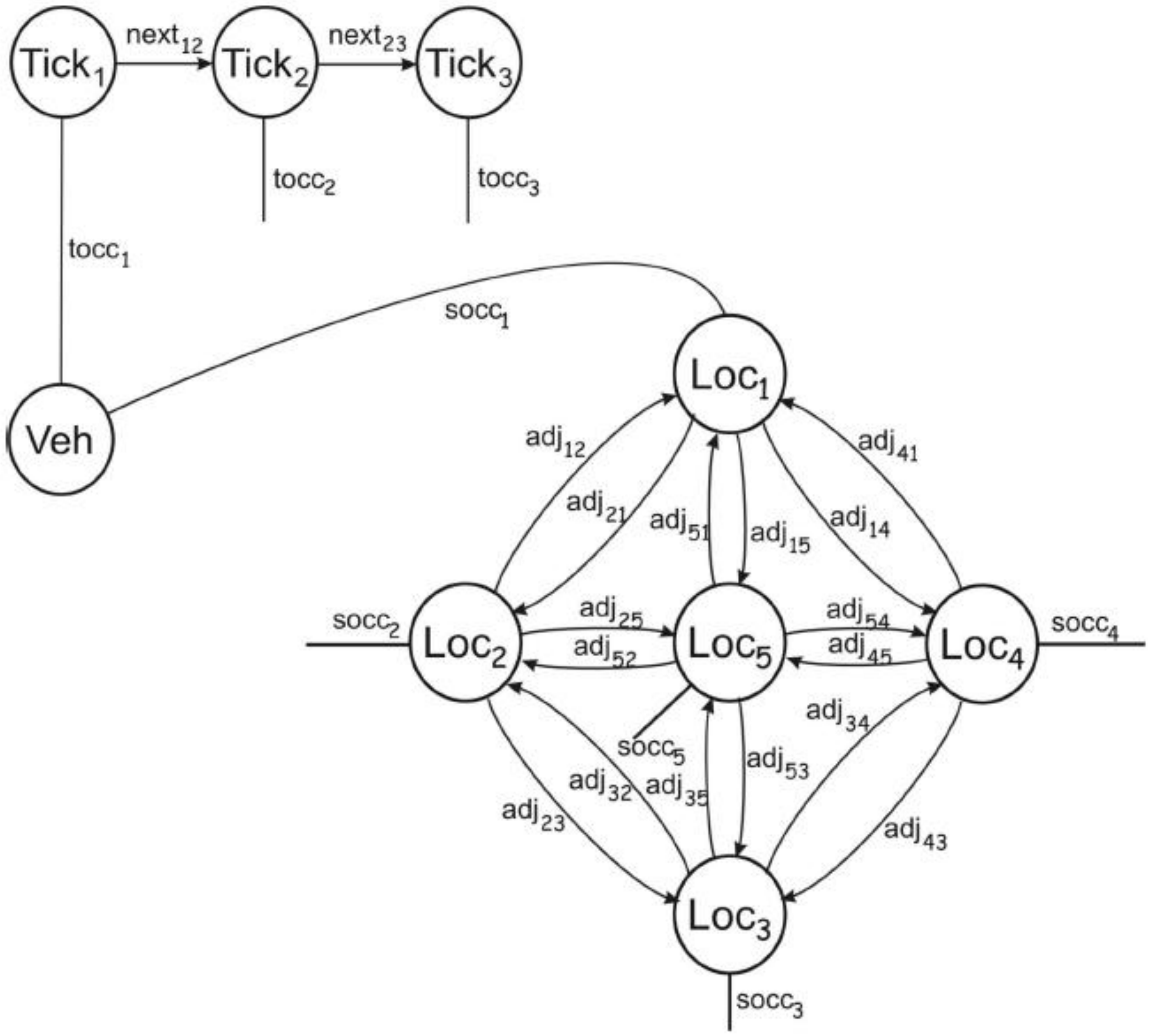


Figure 3.8: The initial stages of connection between processes for the motion example (Worboys 2005: 27)

SEST-Index

In contrast to the purely event-oriented models described above, the SEST-Index represents a hybrid approach that incorporates elements of the snapshot model (Gutiérrez et al. 2005). The SEST-Index provides an access method that can process queries about time-slices, time intervals, and events. Instead of producing another snapshot after each change or after a predefined period, the SEST-Index creates no snapshot until a certain number of changes is reached. A separate log captures and records changes that occur between two snapshots. Assume one queries the state of an object at instant t_1 between two consecutive snapshots at t_0 and t_2. The SEST-Index then accesses the snapshot at t_0. This snapshot is subsequently amended with the corresponding changes between t_0 and t_1 recorded in the change log. In order to assess the performance of the SEST-Index, Gutiérrez et al. (2005) evaluated it against the HR-Index that uses a pure snapshot approach. The results show that the SEST-Index requires less storage than the HR-Index, in particular if the number of objects or changes is low. The performance of the different query types varies between the two access methods. On the one hand, the HR-Index is always faster in answering time-slice queries because the SEST-Index has to read the log in addition to the snapshot. On the other hand, the SEST-Index outperforms the HR-Index in processing time interval queries and queries about events. Concerning the time interval query, this effect becomes even more pronounced as the interval increases. Whereas the HR-Index must read each

R-Tree (snapshot) within the interval, the SEST-Index merely reads one R-Tree and the change log. In the case of queries on events, the SEST-Index is faster than the HR-Index because accessing explicitly stored events requires less time than comparing two snapshots.

3.3.5 Three-Domain Model (3DM)

3DM classifies objects into a semantic, a temporal, and a spatial domain (Yuan 1999). Each of these domains is represented by a separate table whose mutual relations are established by the domain-link table. While the semantics table contains semantic objects such as real estate property, forest, or farmland, the spatial table stores the currently valid spatial objects (see Figure 3.9). An additional spatial graph records the relationships between the present spatial objects and their predecessors (see Figure 3.10). Thus, previous spatial objects can be retrieved without explicitly storing them. The time table lists the changes and their attributes in chronological order (see Figure 3.9). The domain-link table, finally, links the temporal, semantic, and spatial objects by joining the respective IDs of the objects of the three domain tables (see Figure 3.9). The separation of semantics, space, and time into three domains facilitates the processing of complex spatiotemporal queries. The structure of the vector-based 3DM, moreover, reduces storage demands because attributes shared by multiple spatial objects have to be recorded only once. 3DM is also flexible about the framework for implementation as it may be realized in either a relational or an object-oriented database.

a. Semantics Table

Sem. ID	Landcover	Management	Address
1	Old Growth	USFS	12 Forest Rd.
2	Clear-cut	A. Log Co.	3 Clear Dr.
3	Burn	USFS	12 Forest Rd.
4	Clear-cut	B. Log Co.	45 Pine Ave.

b. Time Table

Time ID	Time	Operator ID
1	1600	2439
2	1700	2439
3	1800	7473
4	1950	1029
5	1960	1029

c. Space Table

Space ID	Area	Perimeters
4	A_1	P_1
6	A_2	P_2
8	A_3	P_3
9	A_4	P_4
10	A_5	P_5
11	A_6	P_6
12	A_7	P_7
13	A_8	P_8
14	A_9	P_9
15	A_{10}	P_{10}

d. Domain Link Table (Links among temporal, semantic, and spatial objects)

Sem. ID	Time ID	Space ID List
1	1	1
1	2	2
2	2	3
3	2	4
1	3	5
2	3	3, 6
1	4	7, 10
4	4	8, 9
1	5	10, 11, 13
2	5	6, 12
3	5	4, 14, 15

Figure 3.9: The three domain tables and the domain link table (Yuan 1999: 147)

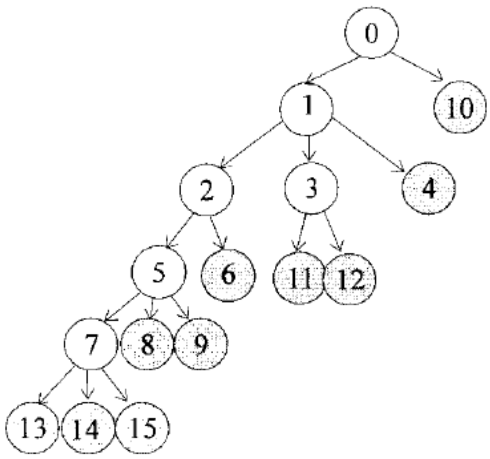


Figure 3.10: The spatial graph records parenthood transitions among spatial objects in the space table (Yuan 1999: 147)

3.4 Ontologies

In recent years, there has been an increasing amount of literature on ontologies. Not only GIScience, but also engineering and computer science in general have discovered ontologies as a significant area of research (Bittner et al. 2004, Agarwal 2005). To GIScience, the most important contributions of ontologies are enhanced interoperability and integration of different datasets. This is achieved by standardization processes and the definition of a common vocabulary. Furthermore, ontologies can serve as a systematic approach to define the general categories and relationships of a certain domain (Agarwal 2005). By including the temporal dimension, ontologies may also bring clarity to the effects of change on objects and properties. Moreover, an ontology can yield insight into the circumstances that induce a creation, a loss, or a preservation of an object's identity (Plumejeaud et al. 2009). For these reasons, ontologies make important contributions to fields such as knowledge representation, database design, information retrieval, Semantic Web applications, and reasoning (Hornsby and Joshi 2010). Nevertheless, Jean et al. (2007) point out that despite increasing use in research, ontologies have not found wide application in practice.

3.4.1 Ontology and ontologies

The term *Ontology* originates from philosophy (Agarwal 2005) where it designates "the metaphysical study of the nature of being and existence" (WordNet 2011). Since also disciplines such as artificial intelligence and computer science have done research on ontology, this term has experienced an extension of meaning (Agarwal 2005). Outside philosophy, an ontology is referred to as a "rigorous and exhaustive organization of some knowledge domain that is usually hierarchical and contains all the relevant entities and their relations" (WordNet 2011). Ontology with a capital "O" only exists in the singular form and denotes the concerning philosophical discipline. In contrast, ontology with a lower-case "o" may also be used in the plural form (Guarino and Giaretta 1995) and is a "specification of a conceptualization" Gruber (1993: 199). In this sense, an ontology corresponds to an abstraction of the world and its constituent entities defined in a particular ontology language such as OWL. A conceptualization, on the contrary, may be communicated in written, oral, or graphic form and is thus not tied to a specific language or a certain vocabulary (Guarino 1998). Hence, there may be different ontologies of the same conceptualization. In the following, this thesis refers to Gruber's (1993) definition of an ontology that differentiates between ontologies and knowledge bases and does therefore not include instances of classes.

3.4.2 Ontologies and knowledge bases

Ontologies and knowledge bases are two terms that often appear in the same context and need clarification. Just as the literature distinguishes between ontology as a philosophical discipline and an ontology as an engineering artifact, Guarino (1998) differentiates between a particular and a generic knowledge base. A particular knowledge base is an ontology and defines a hierarchically structured vocabulary of concepts that neither contains instances nor relationships of instances. A generic knowledge base consists of both the terminological box (T-Box), i.e., the particular knowledge base or ontology, and the assertional box (A-Box), i.e., the instances and their relationships with other instances (Guarino 1998, Swartout and Tate 1999) (see Chapter 3.4.10). This thesis consequently defines a knowledge base as a generic knowledge base according to Guarino (1998).

3.4.3 Universals and particulars

Philosophy makes a distinction between universals and particulars that are also referred to as kinds, species, and types and individuals, instances, and tokens, respectively (Agarwal 2005). Unlike universals, particulars do not have instances. In an ontology, classes and properties represent universals, and individuals are particulars (Gangemi et al. 2002). Country would therefore be a universal of type class which may have instances such as Switzerland, Germany, or the United States.

3.4.4 Continuants and occurrents

A critical ontological distinction can be made between continuant and occurrent entitites (Worboys 2005). Whereas one understands the former as things or objects, the latter are events, processes, or actions. Continuants, also termed endurants, are wholly present at every moment of their existence and thus do not have temporal parts (Johansson 2005). The identity of continuants remains the same when their qualities undergo change. A woman, for example, maintains her identity, even if some of her qualities such as the hair color or the last name change as a consequence of aging or marriage. Occurrents, also referred to as perdurants, extend in time and thus consist of one or multiple temporal parts. Hence, occurrents are only partially present at a certain time because of the absence of the preceding and succeeding temporal parts (Gangemi et al. 2002). The same woman’s life, for instance, comprises the temporal parts childhood, adolescence, adulthood, and retirement age. Due to the successive order of these phases, the woman’s life can never be wholly present in the sense that every temporal part existed at the same time.

3.4.5 Upper ontologies

Upper ontologies (Hornsby and Joshi 2010) or *top-level ontologies* (Guarino 1998) define the basic concepts and general categories that constitute the universe (Guarino 1998). Therefore, upper ontologies do not depend on a particular domain (Guarino 1998) but provide a comprehensive framework for extension by any domain ontology (Noy 2004). Domain ontologies can become reusable if they share the same upper ontology because a comparable structure facilitates interoperability between different domain ontologies (Noy 2004). Indeed, enhanced interoperability and knowledge sharing (Noy 2004) as well as facilitated integration and combination of different ontologies are the main goals of upper ontologies (Spear 2006) such as SUMO (Niles and Pease 2001), DOLCE (Gangemi et al. 2002), and BFO (Grenon and Smith 2004). Noy (2004) reported that though there have been both successful and unsuccessful applications of upper ontologies, it is too early to make a definitive statement about their utility.

Figure 3.11 depicts the hierarchy of ontology types according to Guarino (1998).

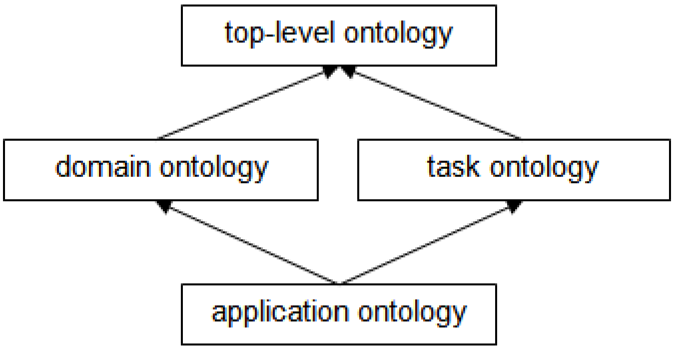


Figure 3.11: Kinds of ontologies (Guarino 1998: 9)

Suggested Upper Merged Ontology (SUMO)

SUMO combines a multitude of publicly available ontologies and is the candidate for the Standard Upper Ontology (SUO) of the Institute of Electrical and Electronics Engineers (IEEE) (Niles and Pease 2001). SUMO is currently one of the most extensive formal ontologies and covers a wide range of fields (Hornsby and Joshi 2010). Furthermore, SUMO applies pragmatic guidelines that avert distinctions due to purely philosophical reasons.

Figure 3.12 shows the three uppermost levels of SUMO.

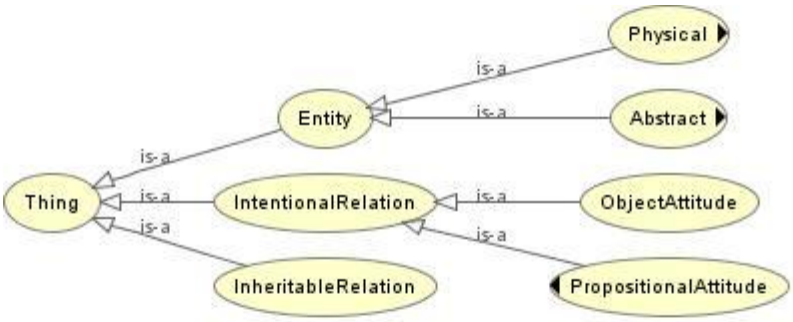


Figure 3.12: The three uppermost levels of SUMO

Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)

DOLCE (Gangemi et al. 2002) has been developed as part of the WonderWeb project to facilitate knowledge sharing among different ontologies (Noy 2004). As opposed to SUMO, DOLCE is not destined to serve as a standard upper ontology. Gangemi et al. (2002) rather intended to provide a reference framework for comparing and investigating ontologies. DOLCE shows obvious cognitive bias as its developers started from the premise that the concepts of the ontology must match human perception and cognition. In the geographic realm, López-Pellicer et al. (2008) used DOLCE to integrate multiple data sources containing the different types of administrative subdivisions in Europe. Their ontology, however, does not go beyond a snapshot of the administrative organization at a certain reference time. The authors rather attempted to improve the comparability of AUs of different EU countries.

Figure 3.13 depicts the four uppermost levels of DOLCE.

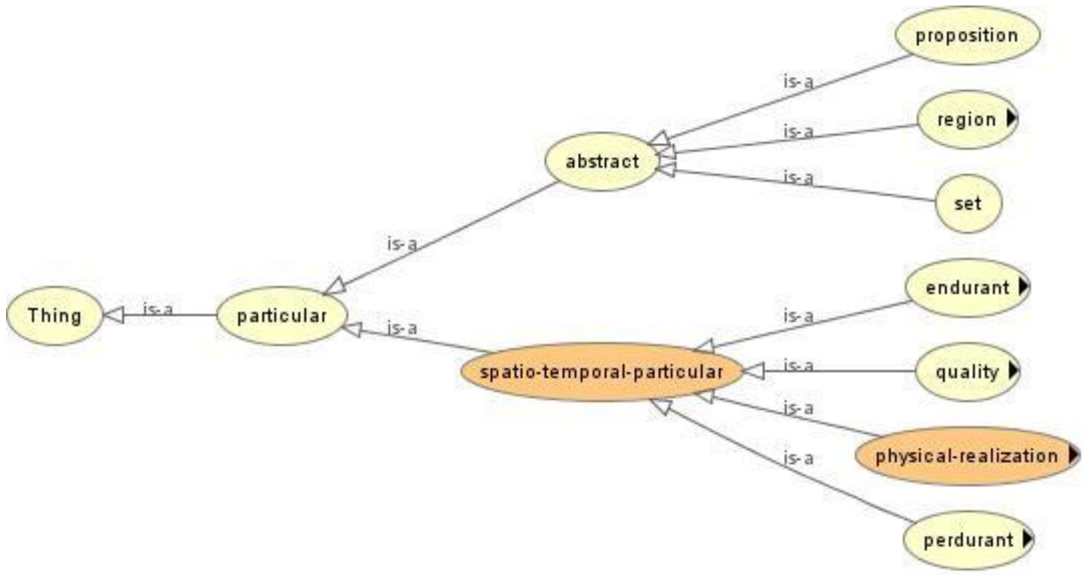


Figure 3.13: The four uppermost levels of DOLCE

Basic Formal Ontology (BFO)

BFO is an upper ontology that combines SNAP and SPAN, i.e., two distinct views of the world (Grenon and Smith 2004). Through the lens of SNAP, the world comprises entities whose existence endures and that also maintain their identity in case of change. A SNAP ontology thus records the state of the world at a given moment, e.g., the product range of Nestlé at a certain time. In BFO, SNAP entities are referred to as continuants and correspond to endurants of DOLCE (see Figure 3.14). Examples for continuants are a person, your heart, a lecture hall, or a municipality. Through the lens of SPAN, the world is composed of four-dimensional entities that extend both in space and time. A SPAN ontology therefore records the events and processes that take place over a certain period, e.g., wartimes and battles in European history. SPAN entities are occurrents in BFO and conform to the concept of perdurants in DOLCE (see Figure 3.14). By combining SNAP and SPAN, but also by making a distinction between a relational and an absolutist view of space, BFO provides concepts for multiple perspectives upon reality that an upper ontology must accommodate.

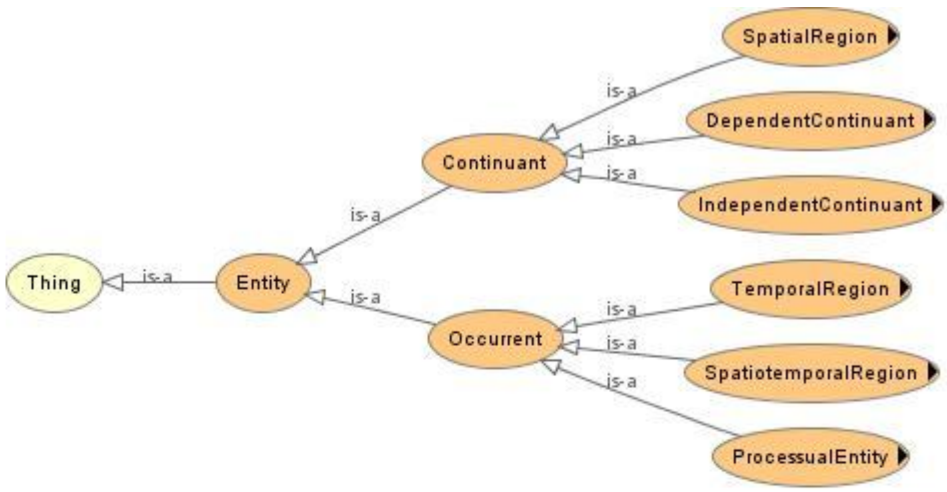


Figure 3.14: The four uppermost levels of BFO

BFO has been developed at the Institute for Formal Ontology and Medical Information Science (IFOMIS) (Grenon and Smith 2004). Since its release, application of BFO has been mainly found in the biomedical domain as the list[1] of users demonstrates. Many researchers now use BFO as a framework to structure biomedical data (Arp and Smith 2008). This success is because molecular biology is characterized by lots of data with well-defined categories that can be directly used for automatic processing (Smith et al. 2007). Geography, on the contrary, often has to cope with vagueness and ambiguity which makes it more difficult to create a formal ontology of its concepts. In geomorphology, for instance, many concepts are ill-defined (Straumann 2010) which is why ontologies have not become widely accepted in that domain. The same applies to the biotope data of the DNL (Bauer-Messmer et al. 2009) that often refer to vague concepts and regions. No wonder that despite limitations (Yu 2006), the biomedical Gene Ontology (GO) currently represents the most successful implementation of BFO in terms of user numbers, coverage of species, as well as granularities (Smith et al. 2007).

3.4.6 Domain and task ontologies

Domain ontologies and *task ontologies* are geared to a generic domain, such as geography, or a generic task, such as hiking (Guarino 1998). They represent a specification of an upper ontology inasmuch as their vocabulary extends the terms of the upper ontology. In spite of a higher level of detail, these ontology types are still application-independent. GIScience, and geography in general, lack a standard domain ontology because the meaning of geographic terms is heavily contingent upon the context of use (Agarwal 2005). The Dutch, for example, may have a different idea of a mountain than the Swiss.

3.4.7 Application ontologies

Application ontologies refine the concepts of both domain and task ontologies. This often implies the specification of roles adopted by domain entities during an activity (Guarino 1998). While domain ontologies attempt to comprehensively and objectively describe a certain domain, application ontologies focus on those elements relevant to a specific application. Hence, the latter are designed to attain a particular goal (Spear 2006).

3.4.8 Ontologies and the Semantic Web

The Semantic Web is an extension of the World Wide Web and enriches Web contents with machine-understandable meanings (Berners-Lee et al. 2001). Whereas the current Web has mostly provided human readable content, the Semantic Web adds structure and semantics to such content, and thus enables computers to perform certain tasks automatically. This includes reasoning, knowledge sharing, and the comparison of vocabulary. In order to accomplish such tasks, the Semantic Web requires an expressive language to describe resources and to define rules for reasoning. The two following languages form the basis for the Semantic Web. First, the Extensible Markup Language (XML) allows to structure Web contents by specifying self-defined tags. Second, the Resource Description Framework (RDF) adds meaning to the structure as it provides a vocabulary for describing resources (see Chapter 3.4.11). RDF stores knowledge in triples that consist of a subject, a predicate, and an object that each represent a resource with a Uniform Resource Identifier (URI). The triple tags are based on XML and enclose the RDF assertions that describe the resources. By adding additional expressiveness with RDF Schema (RDFS) and the Web Ontology Language (OWL), one can create an ontology composed of a taxonomy and a set of inference rules (see Chapter 3.4.11). The contributions of ontologies to the Semantic Web are manifold and include, for instance, enhanced interoperability,

[1] http://www.ifomis.org/bfo/users, accessed December 17, 2010

exchange and comparison of vocabulary, as well as improved hit rates of search engines. For example, ontologies facilitate the exchange of information by storing the equivalence of terms, such as zip code and postal code.

Spatially-Aware Information Retrieval on the Internet (SPIRIT)

An example of a Semantic Web application using ontologies is the SPIRIT spatial search engine (Jones et al. 2004). The aim of SPIRIT is to support spatial queries on the Web in the form < term, spatial relationship, place >, e.g., < hotels, in, London >. To answer such queries, SPIRIT maintains a geographical ontology of place names, including their spatial footprints and spatial relationships, such as containment or adjacency (Abdelmoty et al. 2005). Populated with data from digital maps, gazetteers, and thesauri, this geographical ontology disambiguates place names and expands queries by including alternative spellings. In the first case, SPIRIT requires the user to specify whether he wants Web pages containing information about hotels in London, Canada or London, UK. In the second case, SPIRIT not only expands the search for the place name, but also for the term. This means, for instance, that the geographical ontology provides alternative spellings of London, such as Londres or Londra. A domain ontology additionally feeds the search engine with similar concepts of the term hotel, such as guesthouse or hostel (Jones et al. 2004).

3.4.9 Reasoning

Reasoning is a central facility of ontologies and requires a set of inference rules and an expressive language with a well-defined syntax and semantics (Baader et al. 2005). According to the inference rules, a reasoner deduces implicit knowledge from an ontology. Imagine a class hierarchy where class A is a subclass of B that is again a subclass of C. A reasoner can then infer that A is also a subclass of C. Depending on the development phase of an ontology, reasoning serves different purposes. During the design phase, reasoning can be used to check the consistency of an ontology. Imagine an ontology in which two classes A and B are disjoint. A and B are both superclasses of C which is a class with instances. In this case, a reasoner would detect that the instances of C must not be instances of both A and B. At a subsequent stage of development, reasoning facilitates knowledge sharing and interoperability by computing relationships and comparing vocabularies between different ontologies. Furthermore, reasoning may also be performed on the instances of a knowledge base though this requires considerably more calculation capacity. Finally, the expressivity of an ontology language has a significant impact on reasoning performance (Fischer et al. 2010). Reasoners such as FaCT++ (Tsarkov and Horrocks 2006) face insoluble problems if the expressivity of the language increases as it is the case in OWL Full.

3.4.10 T-Box and A-Box

A knowledge base consists of a terminological box (T-Box) and an assertional box (A-Box) (De Giacomo and Lenzerini 1996). The T-Box contains the axioms on classes and relations and thus corresponds to an ontology. The A-Box, on the contrary, comprises the assertions on instances. While a typical axiom of the T-Box specifies the subclass relationship between two classes, the A-Box, for example, stores the class memberships of instances. In a knowledge base of AUs, therefore, an axiom of the T-Box would specify that municipality is a subclass of AU. In turn, an axiom of the A-Box would assign the instance Zurich to the class municipality. A reasoner could then deduce that Zurich is an AU.

3.4.11 Knowledge representation (KR) languages

KR languages are a family of machine-readable languages used to formally describe and represent knowledge (Stevens et al. 2000). This applies particularly when specifying and encoding a conceptualization in a KR language to create a formal ontology (see Figure 3.15). There is a plurality of KR languages with varying degrees of expressiveness, usability, and computability. Stevens et al. (2000) list three types of KR languages found in the biomedical domain. Besides vocabularies that still build upon natural language, biomedicine uses object-based KR languages such as UML, and logic-based ones such as those described below. Ontology languages such as OWL represent a subset of KR languages and provide additional semantics to express ontological relationships (Antoniou and van Harmelen 2009).

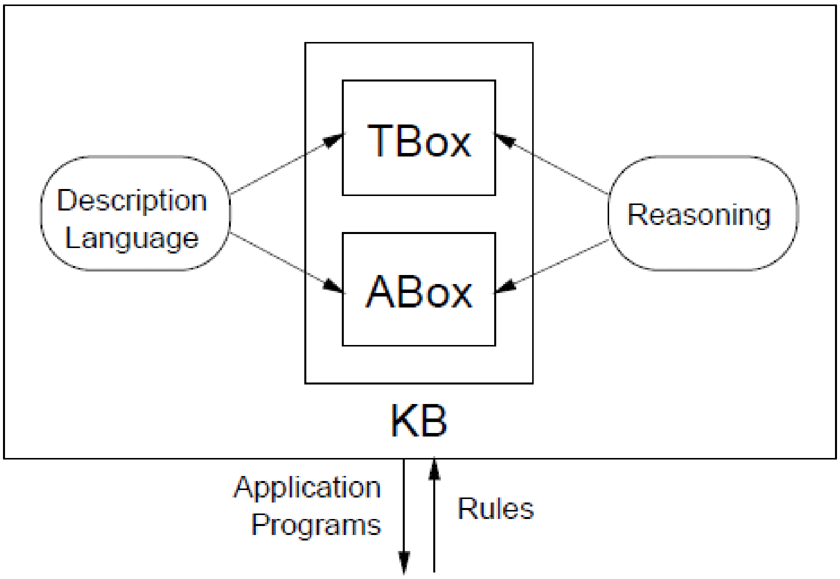


Figure 3.15: Architecture of a knowledge representation system based on description logics (KB = knowledge base) (Baader and Nutt 2003: 50)

Description logics (DLs)

DLs are a group of KR languages used to formally describe the knowledge of a certain domain by a set of logical axioms (Baader et al. 2005). Depending on the purpose, each DL has its own constructors for the formal expression of concepts and roles. Since DLs provide a well-defined semantics, they allow to perform reasoning. Due to their expressive power and reasoning capabilities, DLs form an ideal basis for ontology languages.

Resource Description Framework (RDF)

RDF is a machine-readable language to describe resources on the Web and represents an official recommendation of the World Wide Web Consortium (W3C) (Manola and Miller 2004). More specifically, RDF provides a vocabulary to express metadata about Web resources such as author, title, or creation date of a document. Such a resource does not have to be a Web document, but could also be a product on Amazon annotated with information about the price or availability. For the purpose of identification, each resource in RDF has a Uniform Resource Identifier (URI). A URI can also be a Uniform Resource Locator (URL), a special kind of URI commonly used to refer to a Web page. An exception to this are objects with constant values that have a literal instead of a URI reference. An RDF graph consists of statements termed triples that are composed of a subject, a predicate, and an object (see Figure 3.16). Whereas subjects and predicates are always URIs, objects may be either URI or literal. For the sake of clarity, namespace URIs are usually abbreviated by a prefix. For example, the prefix rdf: denotes the URI http://www.w3.org/1999/02/22-rdf-syntax-ns#. When referring to an RDF resource, one can therefore use the prefix instead of the whole URI.

Figure 3.16: An RDF graph describing Eric Miller (Manola and Miller 2004)

RDF Vocabulary Description Language: RDF Schema (RDFS)

RDFS is a W3C recommendation that semantically extends RDF inasmuch as it provides a vocabulary to describe application-specific classes and properties (Brickley and Guha 2004). Hence, typical RDFS properties are subClassOf to establish a class hierarchy and comment to enrich a resource with a textual description. RDFS, however, does not include terms of application-specific classes and properties such as author or date. The Dublin Core Metadata Initiative provides such a standardized terminology of the key properties to describe resources (Weibel and Koch 2000). Using a standardized terminology facilitates the exchange of information but does not increase the information content of an ontology otherwise. Although RDF and RDFS together allow the creation of simple ontologies including a class hierarchy and simple relationships, the expressive power does not suffice for useful reasoning (Antoniou and van Harmelen 2009).

Terse RDF Triple Language (Turtle)

Turtle is an RDF syntax that stores RDF triples in compact form. Like OWL and RDF, Turtle can be queried with SPARQL (see Chapter 3.4.12) (Beckett and Berners-Lee 2008). Currently, Turtle is not officially recommended by W3C.

Web Ontology Language (OWL)

OWL is currently the W3C standard language for Web ontologies and builds on XML, RDF, and RDFS (Antoniou and van Harmelen 2009). Partially based on DL, OWL provides means to express cardinality, relations between classes such as disjointness, more property types and characteristics, as well as enumerated classes (McGuinness and van Harmelen 2004). This additional expressive power allows to conceive ontologies with increased reasoning capabilities. Depending on the purpose of the ontology, OWL provides three specifications with varying degrees of expressiveness. Of these, OWL Lite is most limited with respect to modeling complex relationships. OWL Lite is thus only useful for simple taxonomies and classifications. On the contrary, OWL DL provides maximum expressivity but still imposes restrictions to maintain the computability of conclusions. OWL Full finally offers the syntactic freedom of RDF in addition to maximum expressiveness. However, computability cannot be guaranteed anymore and reasoning is unlikely to be performed on all elements. OWL Full, for example, allows classes to be a collection of individuals and an individual itself at the same time.

3.4.12 SPARQL Protocol and RDF Query Language (SPARQL)

The **S**PARQL **P**rotocol **a**nd **R**DF **Q**uery **L**anguage (SPARQL) (Prud'hommeaux and Seaborne 2008) is a W3C recommendation and the most frequently used language for querying the Semantic Web (Sirin et al. 2010). As its name says, SPARQL is specially adapted to query RDF graphs but may also be used to query OWL ontologies (Sirin et al. 2010). Since complex OWL expressions cannot be retrieved with standard SPARQL, there have been attempts to provide new query languages such as the **S**emantic **Q**uery-enhanced **W**eb **R**ule **L**anguage (SQWRL) (O'Connor and Das 2009) or OWL extensions of SPARQL such as Terp (Sirin et al. 2010). Although SPARQL resembles SQL in many ways, it also shows notable differences (Polleres 2007). These differences partially result from the open world assumption underpinning Semantic Web technologies such as RDF and OWL. This view assumes every statement to be true unless it is explicitly denied. Hence, there is no negation operation in SPARQL, as there is in SQL. SPARQL also lacks other functions such as AVERAGE or COUNT. This lack, however, has no bearing on the open world assumption. As opposed to SQL, SPARQL supports three other query forms than the SELECT type that sends back a set of resources matching the defined specifications. While the CONSTRUCT query form builds an RDF graph by the union of the selected triples, the ASK form returns a yes/no answer depending on whether there is a result set or not. The DESCRIBE form finally returns RDF triples that describe the resources specified in the query, e.g., the name, gender, date of birth, and salary of the resource employee no. 8638. For additional information about SPARQL and its syntax, consider Prud'hommeaux and Seaborne (2008) or the SPARQL tutorial provided by Matthews (2008).

Figure 3.17 depicts a SPARQL query of the SELECT form and the corresponding result set.

```
PREFIX foaf:   <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox
WHERE
  { ?x foaf:name ?name .
    ?x foaf:mbox ?mbox }
```

name	mbox
"Johnny Lee Outlaw"	<mailto:jlow@example.com>
"Peter Goodguy"	<mailto:peter@example.org>

Figure 3.17: A SPARQL query with the corresponding result set (Prud'hommeaux and Seaborne 2008)

3.5 Implementations

This chapter shifts the emphasis from the models themselves onto their practical implementation while focusing on research concerning AUs. Although work about implementation cannot be clearly separated from conceptual modeling of databases and ontologies, certain aspects come to the fore that are otherwise ignored. Stock (2006), for example, addresses the discrepancy between sophisticated spatiotemporal database models and the actual requirements for such a system. Whereas the former are designed to fully depict space and time, the latter are endowed with only those elements that are really needed. Additionally, implementations must fulfill economic requirements because the creation of a GIS database is an expensive venture. Traditional approaches such as the snapshot model are thus often preferred, which is due to experience with them on the one hand, and more foreseeable expenditures on the other hand (Gregory 2002). This is demonstrated by the fact that none of the examples reviewed by Gregory (2002) adopt an object-oriented approach, in spite of their popularity in database research. The following three paragraphs describe projects that were part of the workshop *Mapping Europe's Historic Boundaries and Borders (2000)*, and are largely based on Gregory (2002). Two additional paragraphs present an object lifecycles method as well as an ontology time series of the AUs of Finland.

3.5.1 Three implementations of a snapshot approach: Prussia, Belgium, and the Netherlands

Gregory (2002) presents three studies applying the snapshot approach (see Chapter 3.3.2) that in this case is also referred to as the key dates approach. The first of these studies created a historical atlas of Prussian counties between 1818 and 1925 (Winnige 2000). The atlas consists of several digitalized snapshots and thus represents a typical example of the key dates approach. Second, De Moor's (2000) quantitative database of Belgian municipalities between 1800 and 1961 reduces certain redundancies by taking separate snapshots of each Belgian province. A boundary change consequently causes the creation of a snapshot of only the concerned province. An additional concordance table links the temporally-indexed snapshots of all provinces. At last, Schreven's (2000) NL-KAART project about Dutch municipalities between 1811 and 1994 already started in 1984 when existing GIS know-how was scarce (Gregory 2002). The database consists of both a concordance and a coordinate file linked by an ID (see Figure 3.18). While the former contains the name and the validity interval of each municipality version, the latter records the corresponding coordinate pairs. Since NL-KAART stores each geometry separately, identical border segments of adjacent municipalities are recorded twice which also results in the omission of topological relationships. This data structure, however, could not be maintained when the database was migrated from SAS/Graph to MapInfo. For each possible combination, a separate snapshot had to be created leading to substantial redundancy. Nevertheless, the 280 snapshots provide a complete documentation of each state within the period of examination.

a. Concordance file

Name	Dates	ID
Assen	1830-1930	A1
Breda	1830-1880	B1
Cuijk	1830-1880	C1
Cuijk	1880-1994	C2
Delft	1830-1880	D1
Delft	1880-1930	D2
Delft	1930-1994	D3

b. Co-ordinate file

ID	Coordinates
A1	0,6 4,6 4,9 0,9
B1	0,3 4,3 4,6 0,6
C1	4,3 8,3 8,9 4,9
C2	4,6 8,3 8,9 4,9
D1	0,0 8,0, 8,3, 3,0
D2	0,0 8,0, 8,3 4,6 0,6
D3	0,0 8,0 8,3 4,6 4,9 0,9

Figure 3.18: Sample (a) concordance and (b) coordinate files used by NL-KAART for four simplified example municipalities (Gregory 2002: 169)

3.5.2 Implementation of a space-time composite: Sweden and the Palatinate

Both the Swedish and the Palatine project adopt approaches that come close to the space-time composite (see Chapter 3.3.2) (Gregory 2002). However, the purpose and the realization of these two projects are very different. The Swedish National Topographical Database (Kristiansson 2000) records the boundary changes of all 9’000 parishes, districts, municipalities, counties, and provinces of Sweden between 1620 and 1990. Due to their stability over centuries, the parishes serve as the least common geometry (LCG). Hence, a digital base map with modern parish boundaries represents the initial state that is retroactively amended with historical boundary changes. Hierarchical relationships among AUs are additionally recorded in a relational database management system (Gregory 2002). This differs markedly from the model of Swiaczny’s (2000) Palatine project. In this model, a boundary change leads to the creation of a new snapshot, and thus to a duplication of the entire base map. The old and the new version are then combined to an updated base map containing the geometries of the old state and the geometries resulting from the boundary change. An additional attribute database records detailed information about the AUs and related boundary changes. This database also contains newly created units that are related with their spatial representations by IDs. The state at a given time can be retrieved by the selection of the existing AUs including their geometries valid at that time. Although this data structure reduces redundancy, the implementation of such a database requires frequent updates and substantial know-how in GIS.

3.5.3 Implementation of a timestamping approach: Great Britain

The Great Britain Historical GIS (GBHGIS) adopts an approach that timestamps each version of an AU (Gregory 2002). The project covers a period from 1840 to 1974 and considers civil parishes, local government districts, as well as registration districts and poor law unions. The core of GBHGIS consists of a table that stores a label point for each AU and a table that contains an arc for each boundary segment. Whereas the former table timestamps the creation and abolition of AUs, the latter table records information about boundary changes. Furthermore, there is a separate master coverage for all three types of AUs. This master coverage stores information about label points and arcs, but not about topological relationships. Spatial topology can nevertheless be retrieved by creating a snapshot of a certain date. Information about temporal topology is obtained by comparing several of such snapshots.

3.5.4 Implementation of the object lifecycles method: Australian Capital Territory

Stock (2006) presents the Spatial Data Management System (SDMS) of the Australian Capital Territory (ACT), a spatiotemporal database and information system for cadastral and land information. Multiple departments of the ACT Planning and Land Authority use SDMS as a means for work, e.g., in planning, land administration, and as an instrument for decision support. Blocks, i.e., land parcels, are the primary dataset of SDMS. Depending on the block type, a change of a block entails a particular administrative procedure. SDMS consists of a data layer (database), an application layer (manipulation), as well as a presentation layer (visualization and interface). SDMS applies the object lifecycles method and represents a pragmatic approach as it does not model all aspects of time. Hence, the object lifecycles method does not correspond to one of the previously described spatiotemporal database models. Nevertheless, this method shares certain similarities with attribute timestamping of the object-oriented approach. As its name says, the object lifecycles method is adapted to the behavior of objects that go through a specific and consecutive sequence of stages throughout their life. Blocks, for example, pass through the stages *Proposed*, *Approved*, *Registered*, *Retired*, and *Deleted* but may also skip some of them. For each lifecycle stage, SDMS creates a new instance that stores the name of the object, the name of the stage, as well as the valid start time and valid end time of the stage. SDMS does the same with the geometries for which it creates a new instance after each boundary change. In order to keep the model simple, SDMS does not timestamp attributes other than the lifecycle stage because they rarely change. Despite its simple architecture, this system is still too complex for some users. Hence, some applications of the ACT Planning and Land Authority reduce the full representation of object lifecycles to a single snapshot view with which users feel more comfortable.

3.5.5 Finnish Spatio-Temporal Ontology (SAPO)

Kauppinen et al. (2008) describe the creation of the Suomen Ajallinen PaikkaOntologia (SAPO). SAPO is an application ontology and knowledge base to store the evolution of the AUs of Finland. The primary goal of SAPO is to overcome the difficulty of referencing historical information to spatiotemporal regions. In addition, SAPO aims to provide users with a facility to envision the extent of historical regions and to efficiently query the knowledge base. Kauppinen et al. (2008) refer to SAPO as an ontology time series that is able to automatically generate the 142 different partonomy hierarchies of the AUs of Finland between 1865 and 2007. SAPO consists of three metadata schemas, the one of current places, the one of historical places, and the one of changes. Due to their similarity, the former two could also be merged. When establishing the knowledge base, an instance was created for each place. This place was then assigned its temporal parts, including the corresponding validity intervals and polygons. The place Helsinki, for instance, is comprised of five temporal parts of which only for one a geometry is available (see CultureSampo 2011). Thereafter, instances were created for all changes and linked with the respective preceding and succeeding temporal parts. This step also included the specification of the type and date of a change. SAPO thus allows to reconstruct the entire change history of municipalities in Finland between 1865 and 2007. Furthermore, this ontology provides for the creation of partonomy hierarchies for any date during that period.

Kauppinen et al. (2008) identified the seven change types listed in Table 3.1. They reflect the specific history of Finland that is marked by several territorial changes with neighboring Russia.

Change types in Finland	Description	Quantity
Establishment	A region is established	508
Merge	Several regions are merged into one	144
Split	A region is split to several regions	94
Namechange	A region changes its name	33
Changepartof (type 1)	Annexed to a different country	66
Changepartof (type 2)	Annexed from a different country	1
Changepartof (type 3)	Region moved to another city or municipality	256
Total sum		1102

Table 3.1: Different types of regional changes in Finland between 1865 and 2007 (based on Kauppinen et al. 2008: 3)

4 Overview of the steps 1 to 4

This chapter gives a brief overview of the four steps toward a knowledge base of the entire change history of the AUs of Switzerland between 1960 and 2010. While the first step commences with analyzing the source data, the second describes the construction of the **S**patiotemporal **On**tology for the **Ad**ministrative **U**nits of **S**witzerland (SONADUS). SONADUS extends the upper ontology BFO and serves as T-Box to create the knowledge base SONADUS-KB in Step 3. The final step describes the evaluation of SONADUS-KB against predefined test queries. In the following, this page briefly explains the content of each step.

4.1 Step 1: Data analysis

This chapter describes the methods used to acquire and analyze the source data and presents the findings in form of figures, descriptions, and change maps. The data originate from two different sources, that is, the Federal Statistical Office (FSO) and the Federal Office of Topography (FOT). While the FSO issues the Historicized Municipality Register (HMR+) and the geometric datasets for the 1990s, the FOT has released yearly updated boundaries since 2000.

4.2 Step 2: Construction of an application ontology

Based on the insights from the literature and the data analysis, the second step describes the construction of an application ontology that models the change history of the AUs of Switzerland. First, this includes the creation of a draft ontology to identify the basic entities of the HMR+, before attaching the draft to the Basic Formal Ontology (BFO). Besides enhanced interoperability and knowledge sharing, upper ontologies such as BFO provide a sound philosophical basis for the creation of an application ontology. Step 2 explains the general structure of SONADUS and describes major contributions in detail. Furthermore, this chapter provides ontology visualizations using the SOVA plug-in and performs reasoning on SONADUS with Pellet to obtain the inferred axioms.

4.3 Step 3: Creation of a knowledge base

This chapter describes the creation of a knowledge base that uses SONADUS as T-Box, including the inferred axioms. The establishment of a knowledge base requires the definition of number intervals for groups of individuals to ensure the uniqueness of every instance in the knowledge base. Since RDF graphs consist of statements in the form < subject, predicate, object >, Step 3 subsequently explains the extraction of such triples from a relational database. Finally, this chapter describes the method to encode these triples into RDF/XML syntax, before presenting figures on the size of the knowledge base.

4.4 Step 4: Evaluation

Step 4 describes the method and the results of the evaluation of SONADUS-KB. This includes storing SONADUS-KB in the triplestore Virtuoso and querying it from a SPARQL endpoint to test whether the knowledge base supports the predefined test queries.

5 Step 1: Data analysis

This chapter describes the methods used to acquire and analyze the source data and presents the findings in form of figures, descriptions, and change maps. The data originate from two different sources, that is, the Federal Statistical Office (FSO) and the Federal Office of Topography (FOT). While the FSO issues the Historicized Municipality Register (HMR+) and the geometric datasets for the 1990s, the FOT has released yearly updated boundaries since 2000. As the source data have been collected by Swiss authorities, the official data descriptions use German, French, and Italian terms to refer to specifically Swiss concepts. Nevertheless, this chapter provides the reader with accurate translations of the Swiss terms used that attempt to come as close as possible to the original meaning.

5.1 Method

5.1.1 Data acquisition and storage

Historicized Municipality Register (HMR+)

Step 1 commenced with the acquisition of the Historicized Municipality Register of Switzerland (HMR+) *(dt. Historisiertes Gemeindeverzeichnis der Schweiz)*. The HMR+ records all current and historical municipalities, districts, and cantons (see Chapter 5.2). The version of April 2010 was downloaded that covers the time span between 01/01/1960 and 04/25/2010 (FSO 2010). This version contains three separate text files and XML documents, one each for municipalities, districts, and cantons, respectively. These text files were stored as excel spreadsheets. Thereafter, the official description of the HMR+ (FSO 2007b) was downloaded in order to understand the construction and design of this dataset. According to its entity-relationship model (FSO 2007b: 10), the excel spreadsheets were stored as three tables in an Oracle database which is the current database software at WSL.

Geometric data (GZG & GG25)

Although the HMR+ records boundary changes, it does not store the actual spatial representation and geometry of the administrative units (AUs). This required the acquisition of separate geometric data. Since the reference dataset had changed from the GZG non-generalized boundaries to the GG25 vector data in 2000, data from two sources had to be collected (see Chapter 5.3). While the FSO provides four versions of the former for the 1990s, the FOT has released yearly updated boundaries since 2000. The internal database of WSL already contained the GZG boundaries of the FSO. Therefore, it sufficed to make a copy of the four tables referring to 1990, 1994, 1996, and 1998. While the latest dataset (2009) of the GG25 boundaries was available online (FOT 2010), older versions (2000-2008) could be obtained on request. Then, the GG25 shapefiles and the GZG datasets were opened by using ESRI ArcMap Version 9.3.1. Since the geometric datasets only contain the municipality boundaries, the dissolve function of the ArcTool Box was used to generate the boundaries of districts and cantons of all 14 datasets. After this, the thematic data of the original GG25 shapefiles were stored in the Oracle database while keeping the actual geometries in a folder. Finally, corresponding documents describing the geometric data were downloaded. This was FOT (2010) concerning the GG25 boundaries and FSO (2007a) concerning the generalized municipality borders that, back in the 1990s, were based on the GZG boundaries.

5.1.2 Data description and analysis

In a first stage, the official data descriptions of the FSO and the FOT were studied to become acquainted with the data. To get an idea of the size and the composition of the dataset, key figures of the HMR+ were determined in a second stage. More precisely, SQL queries sent to the database returned the total number of changes as well as the incidence of each change type among the three hierarchical levels of AUs in Switzerland. Thereafter, the spatial distribution of changes between 01/01/1990 and 01/01/2010 was displayed on a map of 1990 to discover incompatibilities between the HMR+ and the geometries. Both the FSO number and the municipality name were used to establish a connection between these two datasets. In opposition to the HMR+ that goes back to 1960, the reference date of the first digitalized boundaries limits the period covered to the last twenty years.

5.2 Historicized Municipality Register (HMR+)

The HMR+ stores all current and historical AUs of Switzerland between 1960 and the present. Despite its name, the HMR+ records AUs on all hierarchical levels, except for the confederation. The HMR+ thus consists of the three tables *Historicized Municipality Register* (HMR), *Historicized District Register* (HDR), and *Canton Register* (CR). The HMR+ cannot be clearly assigned to any existing database model because it includes elements of both historical databases and object-oriented models. On the one hand, the HMR+ resembles a historical database as it builds on the relational model and only supports valid time. On the other hand, the HMR+ takes an object-centered view and creates versions of AUs with validity intervals. Nevertheless, the HMR+ lacks essential characteristics of object-oriented models, such as inheritance or the notion of an identity. Therefore, versions or entries of AUs do not have an identity that endures incremental change. The HMR+ just records these entries in chronological order, yet the order of the public announcement is decisive rather than the order of the dates of commencement (FSO 2007b). This results from the fact that the FSO updates these registers as soon as a change is published in the official gazette. Any attribute change automatically causes the creation of a new entry in the respective table, and in the hierarchically lower tables. The same applies to the HDR but less so to the CR that only lists the existing cantons. Since the CR does not record changes besides creation and abolition of cantons, the designation *historicized* is henceforth avoided. The pages that follow introduce the key elements of the HMR+ and are thus largely based on the detailed description provided by FSO (2007b).

5.2.1 The administrative units of Switzerland: hierarchical relationships and exceptions

The Swiss Confederation, or Switzerland, is federalist and made up of three hierarchical levels of AUs, that is, the confederation, the cantons, and the municipalities. Most of the 26 cantons additionally subdivide their territory into a number of districts or constituencies *(dt. Wahlkreise)* with functions differing from one canton to another. This means that any place in Switzerland belongs to a canton, and normally to a district and a municipality as well. Exceptions are the largest lakes and unincorporated areas. While the former fall into neither the territories of districts nor into the territories of municipalities, the latter are indeed part of a district but do not belong to any municipality.

Lakes

The HMR+ treats the lakes differently depending on the hierarchical level, which is shown by means of the following examples. On the municipality level, there are three entries of Lake Neuchâtel, one for each canton that makes up a portion of its surface. Consider that there is only one entry for each canton even if the portion consists of multiple parts. Look at Figure 5.1 that depicts the Seeland in

Western Switzerland. The Canton of Freiburg (light pink), for example, takes up a portion of Lake Neuchâtel that is separated into two parts by a small strip belonging to the Canton of Vaud (green). On the level of districts, different cantonal lake portions are subsumed into a *territory not assigned to a district (dt. Bezirksfreies Gebiet)*. Therefore, the *territory not assigned to a district of Freiburg (officially: Région sans district Fribourg / Freiburg)* consists of Freiburg's portions of Lake Neuchâtel, of Lake Murten (right half of the map), and of Lake of Gruyère (outside the map section). This rule does not apply to lake portions of cantons that are not subdivided into districts. Along with the municipalities of such cantons, these portions are part of an auxiliary construct termed *canton without districts (dt. Kanton ohne Bezirksunterteilung)*. An example of such a canton without districts is the Canton of Geneva. Hence, the corresponding HDR entry includes the canton's portion of Lake Geneva.

Unincorporated areas

In contrast to cantonal lake portions, unincorporated areas *(dt. Gemeindefreie Gebiete)* are always part of a district or a canton without districts. Thus, unincorporated areas only appear on the level of municipalities. On this level, unincorporated areas are treated as independent entities and are therefore assigned own FSO municipality numbers. In 2010, Switzerland had four unincorporated areas, one being the State Forest of Galm *(dt. Staatswald Galm)* in the Canton of Freiburg and the other three being *kommunanzen*. Kommunanzen *(singular: Kommunanz)* are relics of the Middle Ages and some type of common land that were once widespread in Alpine areas (Wikipedia 2011). A complete list of cantonal lake portions and unincorporated areas can be found in FSO (2007b: 24-25).

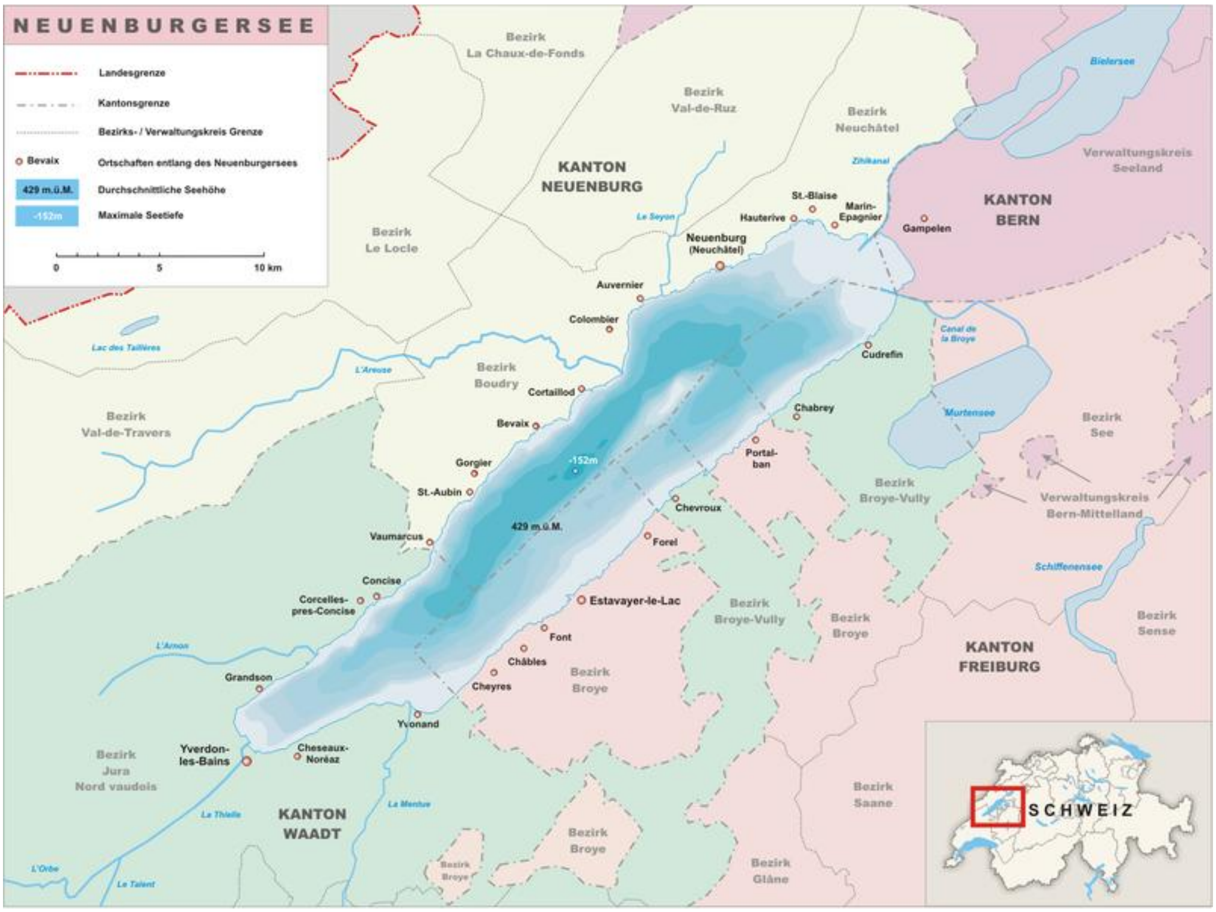


Figure 5.1: Lake Neuchâtel, Lake Murten, and their administrative subdivision (Wikipedia 2007)

5.2.2 Attributes

FSO (2007b: 10-13) provides a detailed description of the attributes of the HMR, HDR, and CR. Aside from briefly describing the most important attributes, this subchapter confines itself to addressing issues regarding the assignment of FSO numbers and the naming of municipalities and districts. This is because either the FSO number or the name is used to join the HMR with the geometric datasets. Joining two datasets on a common attribute requires a good knowledge of errors, special cases, and inconsistencies.

The three tables in Figure 5.2 show the entities, attributes, and relationships of the German version of the HMR+ and the respective English translations. The third and the fourth column list the corresponding names in the database and the data type of the attributes, respectively.

Historicized Municipality Register			
Attributes (German)	***Attributes (English)***	**Attribute Names in Database**	**Type**
Historisierungsnummer Gemeinde	*Historization Number Municipality*	HIST_NR_GDE	NUMBER
<u>*Historisierungsnummer Bezirk*</u>	*Historization Number District*	HIST_NR_BEZ	NUMBER
Kantonskürzel	*Cantonal Code*	KANTONSKUERZEL	VARCHAR2(2)
BFS-Gemeindenummer	*FSO Municipality Number*	BFS_GDE_NR	NUMBER(4)
Amtlicher Gemeindename	*Official Municipality Name*	AMTL_GDE_NAME	VARCHAR2(50)
Gemeindename kurz	*Municipality Name Short*	GDE_NAME_KURZ	VARCHAR2(25)
Art des Eintrags	*Type of Entry*	ART_EINTRAG	VARCHAR2(2)
Status	*State*	STATUS	NUMBER(1)
Mutationsnummer Aufnahme	*Change Number Registration*	MUT_NR_AUFN	NUMBER
Art der Aufnahme	*Type of Registration*	ART_AUFN	NUMBER(2)
Datum der Aufnahme	*Date of Registration*	DATUM_AUFN	DATE
Mutationsnummer Aufhebung	*Change Number Annulment*	MUT_NR_AUFH	NUMBER
Art der Aufhebung	*Type of Annulment*	ART_AUFH	NUMBER(2)
Datum der Aufhebung	*Date of Annulment*	DATUM_AUFH	DATE
Änderungsdatum - Hilfsmerkmal	*Date of Change - Auxiliary Attribute*	AENDERUNGSDATUM	DATE

Historicized District Register			
Attributes (German)	***Attributes (English)***	**Attribute Names in Database**	**Type**
Historisierungsnummer Bezirk	*Historization Number District*	HIST_NR_BEZ	NUMBER
<u>*Kantonsnummer*</u>	*Number of Canton*	KT_NR	NUMBER(2)
Bezirksnummer	*Number of District*	BEZ_NR	NUMBER(4)
Bezirksname	*District Name*	BEZ_NAME	VARCHAR2(70)
Bezirksname kurz	*District Name Short*	BEZ_NAME_KURZ	VARCHAR2(24)
Art des Eintrages	*Type of Entry*	ART_EINTRAG	NUMBER(2)
Mutationsnummer Aufnahme	*Change Number Registration*	MUT_NR_AUFN	NUMBER(3)
Art der Aufnahme	*Type of Registration*	ART_AUFN	NUMBER(2)
Datum der Aufnahme	*Date of Registration*	DATUM_AUFN	DATE
Mutationsnummer Aufhebung	*Change Number Annulment*	MUT_NR_AUFH	NUMBER(3)
Art der Aufhebung	*Type of Annulment*	ART_AUFH	NUMBER(2)
Datum der Aufhebung	*Date of Annulment*	DATUM_AUFH	DATE
Änderungsdatum - Hilfsmerkmal	*Date of Change - Auxiliary Attribute*	AENDERUNGSDATUM	DATE

Canton Register			
Attributes (German)	***Attributes (English)***	**Attribute Names in Database**	**Type**
Kantonsnummer	*Number of Canton*	KT_NR	NUMBER(2)
Kantonskürzel	*Cantonal Code*	KANTONSKUERZEL	VARCHAR2(2)
Kantonsname	*Name of Canton*	KT_NAME	VARCHAR2(50)
Änderungsdatum - Hilfsmerkmal	*Date of Change - Auxiliary Attribute*	AENDERUNGSDATUM	DATE

Legend
bold = primary key
<u>underlined</u> = foreign key

Figure 5.2: Entities, attributes, and relationships of the HMR+. Based on FSO (2007: 10)

Historization number

The historization numbers serve as the primary keys in both the HMR and the HDR (see Figure 5.2). Each entry is assigned a unique *historization number municipality* or *historization number district* that is also kept if an entry becomes invalid. In case of entries with a date of registration later than

01/01/1960, the order of historization numbers corresponds to the chronological order of the public announcements of the changes.

Canton number

The federal constitution determines the order of the cantons that begins with Zurich (1) and ends with Jura (26). The rank is equivalent to the canton number, i.e., the primary key of the CR.

Type of entry

The type of entry is a two-digit code that denotes the type of AU within a hierarchical level. On the level of municipalities, 11 stands for municipalities, 12 for unincorporated areas, and 13 for cantonal lake portions. On the level of districts, 15 is the code for districts, 16 for cantons without districts, and 17 for territories not assigned to a district (see Table 5.1).

Code	German Term	English Translation
11	Politische Gemeinde	Municipality
12	Gemeindefreies Gebiet	Unincorporated area
13	Kantonaler Seeanteil	Cantonal lake portion
15	Bezirk	District
16	Kanton ohne Bezirksunterteilung	Canton without districts
17	Bezirksfreies Gebiet	Territory not assigned to a district
20	Ersterfassung Gemeinde/Bezirk	Initial entry of a municipality/district
21	Neugründung Gemeinde/Bezirk	Establishment of a municipality/district
22	Namensänderung Bezirk	Change of the district name
23	Namensänderung Gemeinde	Change of the municipality name
24	Neue Bezirks-/Kantonszuteilung	New membership of a district/canton
26	Gebietsänderung Gemeinde	Area change of a municipality
27	Formale Neunummerierung Gemeinde/Bezirk	Formal renumbering of a municipality/district
29	Aufhebung Gemeinde/Bezirk	Abolition of a municipality/district
30	Mutation annulliert	Change annulled

Table 5.1: Types of entry and types of registration and annulment. Based on FSO (2007: 13)

FSO number

The FSO assigns a unique number to each entity on both the municipality and the district level. As opposed to the historization number, an FSO number is only unique at a certain point in time. Hence, the identification of an AU by the FSO number requires a specific date. When the FSO numbers were introduced, each canton and district was assigned a number interval within which the municipalities were arranged alphabetically (FSO 2007b). Due to an increasing number of changes, the FSO gave certain municipalities previously-used FSO numbers which made identification over time impossible. Moreover, the alphabetical order of municipalities within districts could not be maintained in every case. The current rules defined in 2003 (FSO 2007b: 6) say that the FSO shall not reuse FSO numbers as long as unused numbers are available. New numbers are only assigned to municipalities being created as a result of separations, secessions, and mergers (see Chapter 5.2.3). If there are free numbers in the concerning interval, municipalities also get assigned new FSO numbers when the membership of a district or canton changes. All other types of changes including annexations do generally not lead to the reassignment of FSO numbers. However, due to the reasons described above, several deviations from these rules exist that are listed in Chapter 5.2.3 and in Appendix A.1. There are several cases where the rules had already been broken in the 1960s and 1970s. This clearly shows that the system for assigning FSO numbers failed soon after its introduction. The number of exceptions has meanwhile reached a level that suggests a fundamental revision of the current model.

Names of municipalities
According to Article 10 of the Regulation on Geographic Names (SR 510.625, Verordnung über die geografischen Namen), a name of a municipality must be unique within Switzerland.

> Art. 10 Grundsätze (SR 510.625, Verordnung über die geografischen Namen)
>
> 1 Der Name einer Gemeinde muss im ganzen Gebiet der Schweiz eindeutig sein und darf zu keiner Verwechslung mit dem Namen einer anderen Gemeinde Anlass geben.
>
> 2 In folgenden Fällen muss dem Gemeindenamen ein Zusatz beigefügt werden:
>
> a. Der gleiche Name wird für mehrere Gemeinden verwendet.
>
> b. Der Name von mehreren Gemeinden wird zwar unterschiedlich geschrieben, aber gleich ausgesprochen.

If the spelling of two different names is identical, both municipalities have to add an adjunct. Frequent adjuncts are the cantonal code (e.g., Gossau (SG) / Gossau (ZH)), references to nearby municipalities (e.g., Wohlen bei Bern), or other toponyms (e.g., Stein am Rhein). This also applies to names with a different spelling but identical pronunciation (e.g., Birmensdorf (ZH) / Birmenstorf (AG)).

Consider that the uniqueness constraint does not apply to names of cantonal lake portions (FSO 2007b). Greifensee, for example, is a name of both a municipality and a lake portion of the Canton of Zurich.

Names of districts
The name of a district in the HDR corresponds with the name that the respective canton uses for a particular district (FSO 2007b). This also includes the prefixed designation of districts in that canton, i.e., either Bezirk, Amt, Amtsbezirk, Wahlkreis, District, or Distretto. In bilingual districts, the name consists of the designations in both languages (e.g., Bezirk Maloja / Distretto di Maloggia).

The names of territories not assigned to a district and cantons without districts are composed of the respective designation and the name of the canton (e.g., Bezirksfreies Gebiet Bern / Berne, Région sans district Neuchâtel, Territorio senza distretto Ticino, Kanton Basel-Stadt, Canton de Genève).

Names of cantons
The name of a canton corresponds with the name in the respective official language. In bilingual and trilingual cantons, the name is comprised of the names in all official languages (e.g., Graubünden / Grigioni / Grischun).

Change number
Changes lead to the registration and annulment of entries and have a unique number. Hence, each entity on both the district and the municipality level possesses the attributes *change number registration* and *change number annulment*. In case of a merger, the change numbers annulment of the abolished municipalities correspond to the change numbers registration of the newly created municipality. The change number registration of initial entries on 01/01/1960 is 1000 in the case of municipalities and 100 in the case of districts.

5.2.3 Changes and change types

The HMR+ does not store changes themselves but annuls and creates entries whenever a change occurs. The same applies to the change types whose classification requires the comparison of the types of annulment of the predecessors with the types of registration of the successors. Table 5.2 provides a list of the change types as defined by the FSO and the corresponding combination of codes. The FSO defined four change types for AUs on the district level and eight change types for AUs on the municipality level. There are no change types for cantons because the FSO does not historicize them. For the sake of completeness, this work considers initial entry also as a change type. Furthermore, it is clearly differentiated between *change of membership of a district* and *change of membership of a canton* as well as between *formal renumbering of districts* and *formal renumbering of municipalities*. A formal renumbering of a district produces a new entry of the concerning district and includes the assignment of a new FSO district number. Moreover, new entries of all municipalities are created that are located within that district. In contrast, a formal renumbering of a municipality concerns only the affected municipality and does not effect change on the district level. To know about the assignments of new FSO municipality numbers, however, is of paramount importance if one intends to join the HMR with geometric data. In case that the two datasets do not refer to the same date, the FSO numbers might not match exactly whereby errors may occur. Further explanations of every change type including examples will be provided in the next section.

Type No.	Type of Change	Type of Annulment	Type of Registration	Change Type of Municipalities	Change Type of Districts	*Comments*
1	Initial entry	none	20	Yes	Yes	
2	Annexation	26 (once) + 29 (once or more)	26 (once or more)	Yes	No	
3	Merger	29 (twice or more)	21 (once)	Yes	No	
4	Separation	29 (once)	21 (twice or more)	Yes	No	
5	Secession	26 (once)	26 (once) + 21 (once or more)	Yes	No	
6	Land exchange	26 (twice)	26 (twice)	Yes	No	
7	Change of the municipality name	23 (once)	23 (once)	Yes	No	
8	Change of the district name	22 (once)	22 (once)	Yes	Yes	
9	Change of membership of a canton	24 (once)	24 (once)	Yes	Yes	*Cantonal code changes*
10	Change of membership of a district	24 (once)	24 (once)	Yes	No	*Cantonal code does not change*
11	Formal renumbering of municipalities	27 (once)	27 (once)	Yes	No	*FSO number changes*
12	Formal renumbering of districts	27 (once)	27 (once)	Yes	Yes	*FSO number does not change*
13	Restructuring of districts in a canton	29 (twice or more)	21 (once or more)	No	Yes	

Table 5.2: Change types of municipalities and districts. Based on FSO (2007: 17)

Change types: examples

This section describes the characteristics of the 13 change types listed in Table 5.2 and illustrates them with representative examples. In certain cases, it requires multiple examples to demonstrate the different aspects of a change type.

1. **Initial entry**
   This change type registered all districts and municipalities at the time of the creation of the HMR+, i.e., on 01/01/1960. Initial entries have a change number registration that is 1000 in the case of municipalities and 100 in the case of districts.

   Example: Winterthur

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
10347	10074	Winterthur	1000	20	1960-01-01			

Table 5.3: Initial entry: Winterthur

2. **Annexation** [A] + [B] → [A+]
   A+ usually inherits the FSO number of A. Exceptions are listed below in the section *Change types: special cases*.

   Example: Aarau + Rohr → Aarau

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
11736	4001	Aarau	1000	20	1960-01-01	3167	26	2009-12-31
12936	4011	Rohr (AG)	1000	20	1960-01-01	3167	29	2009-12-31
15385	4001	Aarau	3167	26	2010-01-01			

Table 5.4: Annexation: Aarau

   Example: Lugano + Barbengo + Carabbia + Villa Luganese → Lugano

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
14493	5192	Lugano	2293	26	2004-04-04	2736	26	2008-04-19
11717	5147	Barbengo	1000	20	1960-01-01	2736	29	2008-04-19
11406	5168	Carabbia	1000	20	1960-01-01	2736	29	2008-04-19
10829	5235	Villa Luganese	1000	20	1960-01-01	2736	29	2008-04-19
14937	5192	Lugano	2736	26	2008-04-20			

Table 5.5: Annexation: Lugano

3. **Merger** [A] + [B] → [C]
   C is usually assigned a new FSO number. Exceptions are listed below in the section *Change types: special cases*.

   Example: Unterehrendingen + Oberehrendingen → Ehrendingen

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
10862	4043	Unter-ehrendingen	1000	20	1960-01-01	2333	29	2005-12-31
12460	4036	Oberehrendingen	1000	20	1960-01-01	2333	29	2005-12-31
14534	4049	Ehrendingen	2333	21	2006-01-01			

Table 5.6: Merger: Ehrendingen

Example: Nesslau + Krummenau → Nesslau-Krummenau

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
14436	3354	Krummenau	2236	24	2003-01-01	2736	26	2004-12-31
14437	3355	Nesslau	2237	24	2003-01-01	2736	29	2004-12-31
14516	3358	Nesslau-Krummenau	2315	21	2005-01-01			

Table 5.7: Merger: Nesslau-Krummenau

Example: Brunnadern + Mogelsberg + St. Peterzell → Neckertal

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
14440	3371	Brunnadern	2240	24	2003-01-01	2748	29	2008-12-31
14455	3406	Mogelsberg	2255	24	2003-01-01	2748	29	2008-12-31
14445	3376	St. Peterzell	2245	24	2003-01-01	2748	29	2008-12-31
14950	3378	Neckertal	2748	21	2009-01-01			

Table 5.8: Merger: Neckertal

Example: Sâles (Gruyère) + Maules + Rueyres-Treyfayes + Romanens → Sâles

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13021	2152	Sâles (Gruyère)	1000	20	1960-01-01	1938	29	2000-12-31
12364	2141	Maules	1000	20	1960-01-01	1938	29	2000-12-31
12839	2151	Rueyres-Treyfayes	1000	20	1960-01-01	1938	29	2000-12-31
12928	2150	Romanens	1000	20	1960-01-01	1938	29	2000-12-31
14138	2152	Sâles	1938	21	2001-01-01			

Table 5.9: Merger: Sâles

4. **Separation** [A] → [B] + [C]
   B and C are assigned new FSO numbers (FSO 2007b). However, this rule has been broken in the only example of a separation. Arni (AG) inherited the FSO number of Arni-Islisberg.

Example: Arni-Islisberg → Arni (AG) + Islisberg

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11320	4061	Arni-Islisberg	1000	20	1960-01-01	1481	29	1982-12-31
13668	4061	Arni (AG)	1481	21	1983-01-01			
13669	4084	Islisberg	1481	21	1983-01-01			

Table 5.10: Separation: Arni-Islisberg

5. **Secession** [A] → [A-] + [B]
A- inherits the FSO number of A while B receives a new FSO number.
There is only one example of that change type in the HMR.

Example: Rubigen → Rubigen + Allmendingen + Trimstein

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13445	623	Rubigen	1259	27	1979-01-01	1565	26	1992-12-31
13757	623	Rubigen	1565	26	1993-01-01	3033	24	2009-12-31
13756	630	Allmendingen	1565	21	1993-01-01	3040	24	2009-12-31
13755	631	Trimstein	1565	21	1993-01-01	3041	24	2009-12-31

Table 5.11: Secession: Rubigen

6. **Land exchange** [A] + [B] → [A+] + [B-]
FSO numbers do not change.
Consider that the HMR only contains land exchanges that concern permanently inhabited areas and that were published as part of change reports in the official municipality register (FSO 2007b). See Chapter 5.3 for an example of a land exchange that the HMR did not store.

Example: Frauenfeld + Gachnang → Frauenfeld + Gachnang

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
14105	4566	Frauenfeld	1907	26	1997-12-31	1909	26	1997-12-31
14106	4571	Gachnang	1908	26	1997-12-31	1909	26	1997-12-31
14108	4566	Frauenfeld	1909	26	1998-01-01			
14107	4571	Gachnang	1909	26	1998-01-01			

Table 5.12: Land exchange: Frauenfeld and Gachnang

7. **Change of the municipality name**
FSO number does not change.

Example: Zurzach → Bad Zurzach

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10328	4323	Zurzach	1000	20	1960-01-01	2719	23	2006-11-30
14920	4323	Bad Zurzach	2719	23	2006-12-01			

Table 5.13: Change of the municipality name: Bad Zurzach

Example: Schlatt → Schlatt (ZH)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13048	226	Schlatt	1000	20	1960-01-01	1920	23	1998-12-31
14120	226	Schlatt (ZH)	1920	23	1999-01-01			

Table 5.14: Change of the municipality name: Schlatt (ZH)

8. **Change of the district name**
Both the FSO municipality number and the FSO district number do not change.

Example: Deitingen (District of Kriegstetten) → Deitingen (District of Wasseramt)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11030	2516	Deitingen	1000	20	1960-01-01	1535	22	1990-12-31
13725	2516	Deitingen	1535	22	1991-01-01			

Table 5.15: Change of the district name: Deitingen

Example: District of Kriegstetten → District of Wasseramt

HIST_NR _BEZ	BEZ_NR	BEZ_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10051	1106	Bezirk Kriegstetten	100	20	1960-01-01	127	22	1990-12-31
10232	1106	Bezirk Wasseramt	127	22	1991-01-01			

Table 5.16: Change of the district name: Wasseramt

On 01/01/2004, the Municipality of **Küssnacht am Rigi** changed its name to **Küssnacht (SZ)**. Consequently, the District of Küssnacht simultaneously changed its name to Küssnacht (SZ). While the HDR recorded the change of the district name, the HMR only created a new entry for the change of the municipality name. Since the HMR did not store the change of the district name, it is impossible to retrieve the full set of changes of a particular change type, e.g., changes of the district name on the municipality level.

9. **Change of membership of a canton**
The municipality or district is assigned both a new FSO municipality number and FSO district number, respectively, if there are free numbers within the number interval of the concerning district. This has been always the case so far.

Example: Liesberg (Canton of Bern, District of Laufen) →
Liesberg (Canton of Basel-Landschaft, District of Laufen)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13458	648	Liesberg	1272	27	1979-01-01	1654	24	1993-12-31
13846	2788	Liesberg	1654	24	1994-01-01			

Table 5.17: Change of membership of a canton: Liesberg

Example: Vellerat (Canton of Bern, District of Moutier) →
Vellerat (Canton of Jura, District of Delémont)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13888	714	Vellerat	1696	27	1994-01-01	1890	24	1996-06-30
14087	6728	Vellerat	1890	24	1996-07-01			

Table 5.18: Change of membership of a canton: Vellerat

Example: District of Laufen (Canton of Bern) →
District of Laufen (Canton of Basel-Landschaft)

HIST_NR _BEZ	BEZ_NR	BEZ_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10222	213	Amtsbezirk Laufen	109	27	1979-01-01	132	24	1993-12-31
10236	1302	Bezirk Laufen	132	24	1994-01-01			

Table 5.19: Change of membership of a canton: District of Laufen

10. Change of membership of a district

The municipality is assigned a new FSO number if there are free numbers within the number interval of the concerning district. In case of restructurings of districts, the municipalities keep their FSO number. This is apparent from the restructurings in the cantons of Bern, St. Gallen, Vaud, Appenzell Innerrhoden, and Graubünden.

Example: Zollikon (Canton of Zurich, District of Zurich) →
Zollikon (Canton of Zurich, District of Meilen)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11347	252	Zollikon	1000	20	1960-01-01	1493	24	1985-12-31
13683	161	Zollikon	1493	24	1986-01-01			

Table 5.20: Change of membership of a district: Zollikon

Example: Burgdorf (Canton of Bern, District of Burgdorf) →
Burgdorf (Canton of Bern, District of Emmental)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11504	404	Burgdorf	1000	20	1960-01-01	2941	24	2009-12-31
15061	404	Burgdorf	2941	24	2010-01-01			

Table 5.21: Change of membership of a district: Burgdorf

On 12/31/2009, the Canton of Bern abolished all 26 districts and restructured its territory into 10 new districts that were established on 01/01/2010. This means that the HMR created a new entry for each Bernese municipality. On the same day, there were three annexations and one merger in the Canton of Bern which resulted in the creation of a new entry for **Langenthal**, **Oberdiessbach**, **Jegenstorf**, and **Twann-Tüscherz**. The FSO then erroneously forgot to record that these municipalities had also changed the district and did not create additional entries for this particular change.

In the 2000s, such restructurings of districts were also implemented in the cantons of St. Gallen, Vaud, and Graubünden. Appenzell Innerrhoden abolished its two districts without replacement at the end of 1996.

11. Formal renumbering of municipalities

The municipality obtains a new FSO number. Consider that all municipalities of the District of Arlesheim were formally renumbered when the District of Laufen changed from the Canton of Bern to the Canton of Basel-Landschaft.

Example: Birsfelden

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11893	2806	Birsfelden	1000	20	1960-01-01	1637	27	1993-12-31
13829	2766	Birsfelden	1637	27	1994-01-01			

Table 5.22: Formal renumbering of municipalities: Birsfelden

12. Formal renumbering of districts

Municipalities keep their FSO municipality numbers while the FSO district numbers change. Several districts of the Canton of Basel-Landschaft were formally renumbered when the District of Laufen passed from the Canton of Bern into the Canton of Basel-Landschaft. The District of Sissach, for example, was assigned the former FSO district number of the District of Waldenburg.

Example: Niederdorf

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10780	2891	Niederdorf	1000	20	1960-01-01	1627	27	1993-12-31
13819	2891	Niederdorf	1627	27	1994-01-01			

Table 5.23: Formal renumbering of districts: Niederdorf

Example: District of Sissach

HIST_NR _BEZ	BEZ_NR	BEZ_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10142	1303	Bezirk Sissach	100	20	1960-01-01	130	27	1993-12-31
10238	1304	Bezirk Sissach	130	27	1994-01-01			

Table 5.24: Formal renumbering of districts: District of Sissach

13. Restructuring of districts in a canton

The newly established districts are assigned a new FSO district number.

Example: Canton of Bern

On 12/31/2009, the Canton of Bern abolished all previous 26 districts (code 29) and restructured its territory into 10 new districts that were established on 01/01/2010 (code 21).

HIST_NR _BEZ	BEZ_NR	BEZ_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10071	203	Amtsbezirk Bern	100	20	1960-01-01	152	29	2009-12-31
10249	221	Amtsbezirk Schwarzenburg	139	27	1994-01-01	152	29	2009-12-31
all other districts of the Canton of Bern						152	29	2009-12-31
10287	245	Verwaltungskreis Emmental	152	21	2010-01-01			
10288	246	Verwaltungskreis Bern-Mittelland	152	21	2010-01-01			
all other districts of the Canton of Bern			152	21	2010-01-01			

Table 5.25: Restructuring of districts in a canton: Canton of Bern

Change types: special cases

I. Changes with an identical date of annulment and date of registration

At a change, the date of annulment of the old entry is usually the day before the date of registration of the new entry. However, if an AU undergoes various changes on the same date, an equal number of new entries has to be created. In such cases, the date of annulment of the old entry coincides with the date of registration and the date of annulment of the first new entry. The second new entry then becomes valid on the following day. A complete list of these cases can be found in FSO (2007b: 23).

II. Annexation: change #2174

In case of an annexation, the annexing municipality always keeps its name. Otherwise, the change would be considered a merger. The following example of an annexation is thus, strictly speaking, a merger because of the different spelling. Nevertheless, the FSO classified this change as an annexation, probably because the pronunciation remained the same.

FALSE: [A] + [B] → [A+]
Donath + Patzen-Fardün → Donat

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
14240	3705	Donath	2040	24	2001-01-01	2174	26	2002-12-31
14244	3709	Patzen-Fardün	2044	24	2001-01-01	2174	29	2002-12-31
14374	3705	Donat	2174	26	2003-01-01			

Table 5.26: Incorrect classification of a change: Donat

RIGHT: [A] + [B] → [C]
Donath + Patzen-Fardün → Donat

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
14240	3705	Donath	2040	24	2001-01-01	2174	29	2002-12-31
14244	3709	Patzen-Fardün	2044	24	2001-01-01	2174	29	2002-12-31
14374	37xx	Donat	2174	21	2003-01-01			

Table 5.27: Suggested solution: Donat

III. Annexation and land exchange: change #2299

According to the definitions of the FSO and the HMR query tool accessible on FSO (2010), change #2299 is an annexation. In fact, Castel San Pietro annexed Casima and Monte on 04/04/2004 but Caneggio did not annex land from these two abolished municipalities. Quite the contrary, Caneggio ceded the village of Campora to Castel San Pietro and thereby lost territory (Bianchi 2005). This, however, does not become apparent from the data because it cannot be reconstructed whether Castel San Pietro or Caneggio annexed Casima and Monte. Moreover, it would be even possible to conclude that both municipalities annexed a part of Casima and Monte. This ambiguity has devastating effects as erroneous inferences may be drawn at a later time. For example, Caneggio merged with five other municipalities to Breggia on 10/25/2009. A query to check the predecessors of Breggia would, inter alia, mistakenly return Casima and Monte. Hence, change #2299 should be split up into two separate changes, the first one being the annexation of Casima and Monte and the second one being the land exchange between Castel San Pietro and Caneggio.

FALSE: [A] + [B] + [C] + [D] → [A+] + [D-]
Castel San Pietro + Casima + Monte + Caneggio →
Castel San Pietro (with Campora) + Caneggio (without Campora)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11414	5249	Castel San Pietro	1000	20	1960-01-01	2299	26	2004-04-03
11412	5248	Casima	1000	20	1960-01-01	2299	29	2004-04-03
12346	5256	Monte	1000	20	1960-01-01	2299	29	2004-04-03
11403	5246	Caneggio	1000	20	1960-01-01	2299	26	2004-04-03
14499	5249	Castel San Pietro	2299	26	2004-04-04			
14500	5246	Caneggio	2299	26	2004-04-04	2772	29	2009-10-24

Table 5.28: Annexation and land exchange as one change: Castel San Pietro and Caneggio

RIGHT: [A] + [B] + [C] → [A] and [A] + [D] → [A+] + [D-]
Castel San Pietro + Casima + Monte → Castel San Pietro
Castel San Pietro + Caneggio →
Castel San Pietro (with Campora) + Caneggio (without Campora)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11414	5249	Castel San Pietro	1000	20	1960-01-01	2299	26	2004-04-03
11412	5248	Casima	1000	20	1960-01-01	2299	29	2004-04-03
12346	5256	Monte	1000	20	1960-01-01	2299	29	2004-04-03
14xxx	5249	Castel San Pietro	2299	26	2004-04-03	23xx	26	2004-04-03
11403	5246	Caneggio	1000	20	1960-01-01	23xx	26	2004-04-03
14499	5249	Castel San Pietro	23xx	26	2004-04-04			
14500	5246	Caneggio	23xx	26	2004-04-04	2772	29	2009-10-24

Table 5.29: Annexation and land exchange as two changes: Castel San Pietro and Caneggio

IV. Annexation by two municipalities: change #1077

As in the previous case, the FSO regards change #1077 as an annexation (FSO 2010). This change is indeed an annexation but with two annexing municipalities involved. Both Maienfeld and Fläsch annex a part of the abolished kommunanz Gemeinschaftsgebiet Maienfeld-Fläsch. In consideration of the fact that change #2299 combines an annexation with a land exchange, this could also apply to this case. It is thinkable that either Maienfeld or Fläsch annexes the Gemeinschaftsgebiet Maienfeld-Fläsch while the other one cedes land to the annexing municipality. The best way to remove this ambiguity is to avoid combinations of annexations and land exchanges in the future.

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10790	3953	Maienfeld	1000	20	1960-01-01	1077	26	1977-10-23
10792	3951	Fläsch	1000	20	1960-01-01	1077	26	1977-10-23
11252	3955	Gemeinschafts-gebiet Maienfeld-Fläsch	1000	20	1960-01-01	1077	29	1977-10-23
13262	3951	Fläsch	1077	26	1977-10-24	2140	24	2000-12-31
13263	3953	Maienfeld	1077	26	1977-10-24	2142	24	2000-12-31

Table 5.30: Annexation by two municipalities

V. Annexations where the annexing municipality received a new FSO number

a. Use of new numbers
The remaining cases of this type are listed in Appendix A.1.

Diessenhofen:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11562	4541	Diessenhofen	1000	20	1960-01-01	1931	26	1999-12-31
10343	4542	Willisdorf	1000	20	1960-01-01	1931	29	1999-12-31
14131	4545	Diessenhofen	1931	26	2000-01-01			

Table 5.31: Annexation with a new FSO number: Diessenhofen

b. Use of a number of a municipality that is annexed
The remaining cases of this type are listed in Appendix A.1.

Wagenhausen:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10456	4873	Wagenhausen	1000	20	1960-01-01	1875	26	1995-05-31
12648	4871	Kaltenbach	1000	20	1960-01-01	1875	29	1995-05-31
12910	4872	Rheinklingen	1000	20	1960-01-01	1875	29	1995-05-31
14069	4871	Wagenhausen	1875	26	1995-06-01			

Table 5.32: Annexation where the FSO number is taken from one of the municipalities that are annexed

c. Reuse of numbers of former municipalities

Berg (TG):
Berg (TG) + Andhausen → Berg (TG)
Berg (TG) + Guntershausen bei Birwinken + Graltshausen + Mauren → Berg (TG)

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11877	4892	Berg (TG)	1000	20	1960-01-01	1563	26	1992-12-31
11761	4891	Andhausen	1000	20	1960-01-01	1563	29	1992-12-31
13753	4892	Berg (TG)	1563	26	1993-01-01	1869	26	1994-12-31
12401	4894	Mauren	1000	20	1960-01-01	1869	29	1994-12-31
12041	4893	Graltshausen	1000	20	1960-01-01	1869	29	1994-12-31
11930	4903	Guntershausen bei Birwinken	1000	20	1960-01-01	1869	29	1994-12-31
14062	4891	Berg (TG)	1869	26	1994-12-31	1870	26	1994-12-31

Table 5.33: Annexation where a formerly used FSO number is taken

Sulgen/Zihlschlacht-Sitterdorf:
Sulgen + Bleiken (TG) → Sulgen
Sulgen + Opfershofen → Sulgen + Opfershofen
Donzhausen → Donzhausen
Hessenreuti → Hessenreuti
Sulgen + Götighofen + Donzhausen + Hessenreuti → Sulgen
Zihlschlacht + Sitterdorf → Zihlschlacht-Sitterdorf

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11279	4510	Sulgen	1000	20	1960-01-01	1014	26	1964-05-31
11241	4506	Bleiken (TG)	1000	20	1960-01-01	1014	29	1964-05-31
13199	4510	Sulgen	1014	26	1964-06-01	1858	26	1994-12-31
12481	4915	Opfershofen (TG)	1000	20	1960-01-01	1858	26	1994-12-31
14051	4510	Sulgen	1858	26	1995-01-01	1887	26	1995-12-31
14050	4915	Opfershofen (TG)	1858	26	1994-12-31	1859	29	1994-12-31
12177	4912	Donzhausen	1000	20	1960-01-01	1856	27	1994-12-31
14048	4511	Donzhausen	1856	27	1995-01-01	1887	29	1995-12-31
12315	4913	Hessenreuti	1000	20	1960-01-01	1857	27	1994-12-31
14049	4512	Hessenreuti	1857	27	1995-01-01	1887	29	1995-12-31
12045	4507	Götighofen	1000	20	1960-01-01	1887	29	1995-12-31
14084	4506	Sulgen	1887	26	1996-01-01			
12979	4517	Sitterdorf	1000	20	1960-01-01	1893	29	1996-12-31
10318	4518	Zihlschlacht	1000	20	1960-01-01	1893	29	1996-12-31
14090	4511	Zihlschlacht-Sitterdorf	1893	21	1997-01-01			

Table 5.34: Reuse of FSO numbers of former municipalities: Sulgen and Zihlschlacht-Sitterdorf

VI. Mergers without a new FSO number

There are several examples where the newly created municipality inherited an FSO number of one of its constituent municipalities. Since 2004, the FSO has stipulated the use of a new FSO number in case of a merger. However, if there are no unused FSO numbers within the number interval of the respective district, the FSO still reuses a number.
The remaining cases of this type are listed in Appendix A.1.

Pont-en-Ogoz:
Pont-en-Ogoz + Villars-d'Avry → Le Bry
Le Bry + Avry-devant-Pont + Gumefens → Pont-en-Ogoz

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
11194	2146	Pont-en-Ogoz	1000	20	1960-01-01	1031	29	1970-01-25
11195	2157	Villars-d'Avry	1000	20	1960-01-01	1031	29	1970-01-25
13216	2146	Le Bry	1031	21	1970-01-26	2168	29	2002-12-31
11662	2122	Avry-devant-Pont	1000	20	1960-01-01	2168	29	2002-12-31
11967	2136	Gumefens	1000	20	1960-01-01	2168	29	2002-12-31
14368	2122	Pont-en-Ogoz	2168	21	2003-01-01			

Table 5.35: Merger without a new FSO number: Pont-en-Ogoz

VII. Separations without a new FSO number

The only separation since 1960 represents a special case. This exception is described above in the section *Change types: examples* and is illustrated in Table 5.10.

5.2.4 Figures of the HMR+

The figures in Table 5.36 do not only provide an overview of the HMR+ as a dataset, but also yield insight into change of AUs as a socioeconomic phenomenon.

	HMR	HDR	CR
Number of entries on 01/01/1960	3143	197	25
Number of entries on 01/01/2010	5331	285	26
Number of valid entries on 01/01/2010	2641	163	26
Number of changes between 01/01/1960 and 01/01/2010	2172(+1*/+1**/+4***)	52	1
…of them annexations	167		
…of them mergers	109		
…of them separations	1		
…of them secessions	3		
…of them land exchanges	8(+1*)		
…of them changes of the municipality name	66		
…of them changes of the district name	47(+1**)	5	
…of them new memberships of a district	1094(+4***)		
…of them new memberships of a canton	96	4	
…of them formal renumberings of municipalities	27		
…of them formal renumberings of districts	554	37	
…of them restructurings of districts in a canton		6	

* land exchange and annexation of Castel San Pietro/Caneggio that is erroneously stored as one change

** name change from District of Küssnacht to District of Küssnacht (SZ) not modeled in the HMR

*** Langenthal, Twann-Tüscherz, Oberdiessbach, and Jegenstorf whose new memberships of a district were forgotten

Table 5.36: Figures of the HMR, HDR, and CR

Besides initial entries, there were 2172 changes between 1960 and 2010. Due to several errors, the actual number of changes is higher. First of all, the FSO classified change #2299 only as an annexation despite involving a land exchange as well. Furthermore, the FSO did not record the name change of the District of Küssnacht (SZ) on the municipality level and the new memberships of a district of four Bernese municipalities. These two types of cases have in common that on the very same day the municipalities involved underwent another change. It thus remains unclear how to proceed when two attributes of an AU change simultaneously.

5.2.5 Essential characteristics of the HMR+

Principle of creating new entries

The HMR and HDR create a new entry as soon as an attribute of an entry changes. In contrast, the CR only stores the current state and does therefore not deserve the designation *historicized*. This also means that the HMR and HDR record both the date of registration and the date of annulment while the CR stores no more than the date of the last change of the entry. Problems arise when multiple attributes change on the same day that would each be categorized as an individual change type. Examples of this are the simultaneous mergers and new memberships of a district of four municipalities in the Canton of Bern that are described in Chapter 5.2.3. The same section contains a comparable case, i.e., Küssnacht (SZ), where a change of the municipality name caused a change of the corresponding district name on the same date. The HMR lacks an additional entry to store the change of the district name. On the one hand, this practice makes it impossible to retrieve the full set of municipalities that underwent a change of a certain type. On the other hand, it prevents the HMR from creating nonsense entries, such as an HMR record with a new municipality name (Küssnacht (SZ)) and an old district name (Bezirk Küssnacht).

Entries with an identical date of registration and date of annulment

FSO (2007b: 23) lists 13 HMR entries with the same date of registration and annulment. These entries concern municipalities that underwent multiple changes that occurred on the same day and affected the courses of the borders. In reality, these coinciding changes cannot be regarded as being independent from each other. They rather represent an auxiliary construct of the HMR as there is no defined change type for large-scale territorial restructurings involving various municipalities. The following example illustrates that the HMR contains entries representing states that actually never existed.

Bischofszell:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11804	4471	Bischofszell	1000	20	1960-01-01	1881	26	1995-12-31
12036	4486	Gottshaus	1000	20	1960-01-01	1881	26	1995-12-31
14076	4486	Gottshaus	1881	26	1995-12-31	1882	29	1995-12-31
14077	4471	Bischofszell	1881	26	1995-12-31	1884	26	1995-12-31
12005	4497	Halden	1000	20	1960-01-01	1884	29	1995-12-31
12958	4500	Schweizersholz	1000	20	1960-01-01	1884	29	1995-12-31
14080	4471	Bischofszell	1884	26	1995-12-31	1886	26	1995-12-31
14081	4501	Kradolf-Schönenberg	1885	21	1995-12-31	1886	26	1995-12-31
14082	4501	Kradolf-Schönenberg	1886	26	1996-01-01			
14083	4471	Bischofszell	1886	26	1996-01-01			

Table 5.37: Three changes involving Bischofszell on the same day

No differentiation between incremental and fundamental change

The HMR+ treats all changes equally. Consequently, one does not know which change types cause incremental change and fundamental change, respectively. At an incremental change, the respective AU maintains its identity whereas a fundamental change leads to the creation or the loss of an identity. A reason for this, inter alia, is that the HMR+ does not possess an attribute that identifies an AU over time. As Table 5.38 reveals, certain entries of an AU may not share any attribute values with other entries of the same AU. Therefore, a retrieval of all entries of an AU is only possible by making joins on the change number. This is illustrated by means of the following example.

The HMR contains four entries of a kommunanz named Kommunanz Reckingen-Gluringen/Grafschaft since 01/01/2004. Two name changes and a formal renumbering sufficed to make the kommunanz unrecognizable in comparison to its first entry. Hence, it does not even require a change with border modifications to substantially alter attributes. The question is left open under which circumstances an identity is created, lost, or maintained.

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11249	6072	Kommunanz Gluringen/ Ritzingen	1000	20	1960-01-01	1934	23	2000-09-30
14134	6072	Kommunanz Gluringen/ Grafschaft	1934	23	2000-10-01	2291	27	2003-12-31
14491	6391	Kommunanz Gluringen/ Grafschaft	2291	27	2004-01-01	2304	23	2004-09-30
14505	6391	Kommunanz Reckingen-Gluringen/ Grafschaft	2304	23	2004-10-01			

Table 5.38: Two name changes and a formal renumbering make an AU unrecognizable over time

Unclear definition of the change type restructuring of districts in a canton

The figures in Table 5.36 show that there have been six restructurings since 1960. Four of them were large-scale restructurings where the cantons abolished the entire old structure and established districts with different names and areas. Although the restructurings in the cantons of St. Gallen, Bern, Vaud, and Graubünden affected all districts in every case, some districts were reestablished, and now correspond exactly with their predecessors. The District of Plessur in the Canton of Graubünden, for example, is spatially identical to the entry of the same name before the restructuring on 01/01/2001. As distinct from the prior cantons, the Canton of Appenzell-Innerrhoden abolished its two districts and became a canton without districts.

Exceptional is change #124 that led to the abolition of the District of Zurich at the end of 1989 and the ensuing establishment of a new District of Zurich and the District of Dietikon on January 1, 1990. This change amounts to a secession on the district level. The District of Dietikon was formed from municipalities of the District of Zurich that maintained its name at the same time. However, since the HDR does not regard secessions as a change type, change #124 is considered a restructuring of districts in a canton.

5.3 Geometric datasets

Before 2000, both the Federal Statistical Office (FSO) and the Federal Office of Topography (FOT) created vector datasets with the boundaries of municipalities, districts, and cantons. At that time, WSL used the non-generalized GZG boundaries to georeference biotope objects. The FSO published both non-generalized and various generalized versions of the years 1990, 1994, 1996, and 1998. Afterward, the FOT has been solely authorized to issue non-generalized datasets of Swiss administrative boundaries. Hence, WSL simultaneously changed the official reference to the GG25 boundaries. An information system must therefore use the GZG boundaries for the period between 1990 and 2000 and the GG25 boundaries for the time between 2000 and 2010. Note that the FSO still provides generalizations that are, however, based on GG25 now. Although there are only minor differences between the GZG and the GG25, the GG25 boundaries have a slightly higher level of detail. This becomes evident when comparing the two datasets. The GG25 vector data, for instance, depict the Brissago Islands in Lake Maggiore whereas the GZG boundaries do not. Despite small deviations, this fact has to be considered when GZG referenced biotope data are overlaid with current municipality boundaries. Moreover, the geometric datasets record smaller border changes in uninhabited areas that the HMR does not consider (FSO 2007b). According to the HMR, the Municipality of Monthey has not changed since 1960. In contrast, Figure 5.3 depicts the acquisition of an area of 1.8 km^2 from neighboring Collombey-Muraz. It cannot be excluded, however, that this land exchange is an error correction in reality.

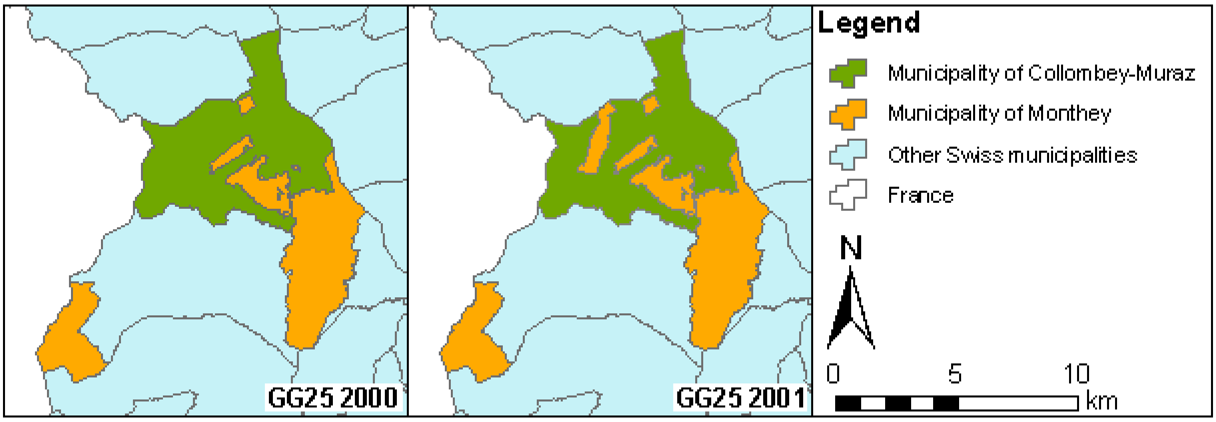


Figure 5.3: Land exchange between Monthey and Collombey-Muraz not recorded by the HMR

5.3.1 GZG

The GZG datasets of the FSO came out four times in the 1990s and refer to January 1 of the respective year. Since an official description lacks, Table 5.39 lists the attributes of the GZG datasets that slightly differ between the four versions. The OBJECTID serves as the primary key and uniquely identifies each polygon. This means that municipalities whose areas are comprised of multiple polygons have an analogical number of entries. Unlike the GG25, the GZG do not number the different parts of a municipality, and therefore the OBJECTID of polygons changes from one version to the other. The spelling of municipality names also differs between the single GZG datasets. In the version of 1990, for example, all names are spelled in upper case whereas names in later versions are spelled in lower case. Moreover, the spellings of municipality names differ between the GZG 1990 and the HMR (see Appendix A.2). Joins on the name thus require extensive modifications in advance. With respect to cantonal lake portions, joins with the HMR become even impossible since the GZG datasets contain the wrong FSO numbers.

GZG90	GZG94	GZG96	GZG98
OBJECTID	OBJECTID	OBJECTID	OBJECTID
AREA	AREA	FLAECHE	FLAECHE
PERIMETER	PERIMETER	UMFANG	UMFANG
GZG90_	GZG94_	GZG96_NEU_25_	GZG98_NEU_25_
GZG90_ID	GZG94_ID	GZG96_NEU_25_ID	GZG98_NEU_25_ID
KANTON	KT	KANTON	KANTON
BEZIRK	BEZIRK	BEZIRK	BEZIRK
GEMEINDE	GMDE	GEMEINDE	GEMEINDE
STATINF			
GEMEINDENAMEN	NAME	GEMEINDENAME	GEMEINDENAME
MUTATIONSCODE	MUT		MUT_1996-98
SHAPE	SHAPE	SHAPE	SHAPE
SHAPE.AREA	SHAPE.AREA	SHAPE.AREA	SHAPE.AREA
SHAPE.LEN	SHAPE.LEN	SHAPE.LEN	SHAPE.LEN

Table 5.39: Attributes of the GZG datasets

5.3.2 GG25

The FOT edits and annually updates the GG25 boundaries that refer to the state on January 1 of each year. An exception represents the dataset of 2001 whose reference date coincides with the reference date of the Federal Population Census 2000 which is December 5, 2000 (FSO 2007a). The GG25 datasets are based on three data sources, that is, the FOT itself, but also Cadastral Surveying and the FSO (FOT 2006). In contrast to the GZG datasets, FOT (2006) provides a detailed description of the GG25 and its attributes.

An error occurred when the Canton of St. Gallen abolished its districts at the end of 2002 and introduced constituencies *(dt. Wahlkreise)* on January 1, 2003. While the GG25 dataset of 2003 still contains the FSO district numbers of the former districts, the versions of 2004, 2005, and 2006 consider St. Gallen as a canton without districts. This means that all municipalities of St. Gallen and the canton's territory not assigned to a district erroneously have the same FSO district number.

5.4 Change maps

In total, there are six choropleth maps that visualize the occurrences of all change types. Layers with non-overlapping patterns of change types were overlaid in order to minimize the number of maps. An exception is the map in Figure 5.4 that only serves as a means of illustration. This map depicts those municipalities that were affected by mergers and/or annexations between 1990 and 2010, two change types that both lead to a reduction of municipalities. As the example of Montagny (FR) demonstrates, these two types spatially overlap, and therefore this map is not part of the six maps that can be found in Appendix B Change maps.

Montagny-les-Monts + Montagny-la-Ville → Montagny (FR) (01/01/2000)

Montagny (FR) + Mannens-Grandsivaz → Montagny (FR) (01/01/2004)

Regardless of the disproportionally large surfaces of municipalities in the mountainous parts of the country, the spatial distribution reveals which cantons proactively support the amalgamation of predominantly small municipalities. First of all, this concerns the Canton of Thurgau that conducted a large-scale restructuring of its municipalities in the 1990s (Schuler et al. 2005). Aside from the area along the shore of Lake Constance, there is hardly any municipality that was not affected by a merger or an annexation. Other cantons, such as Freiburg, Lucerne, or Ticino, have implemented similar projects that, likewise, primarily concern small municipalities in rural areas. Furthermore, the pattern of the map clearly shows the absence of amalgamations of municipalities in the large urban areas of Switzerland. The Canton of Zurich, for example, has experienced neither an annexation nor a merger for more than 50 years. However, this should not hide the fact that also in certain rural cantons, such as in the Canton of Uri, the number of municipalities has not changed at all since 1960.

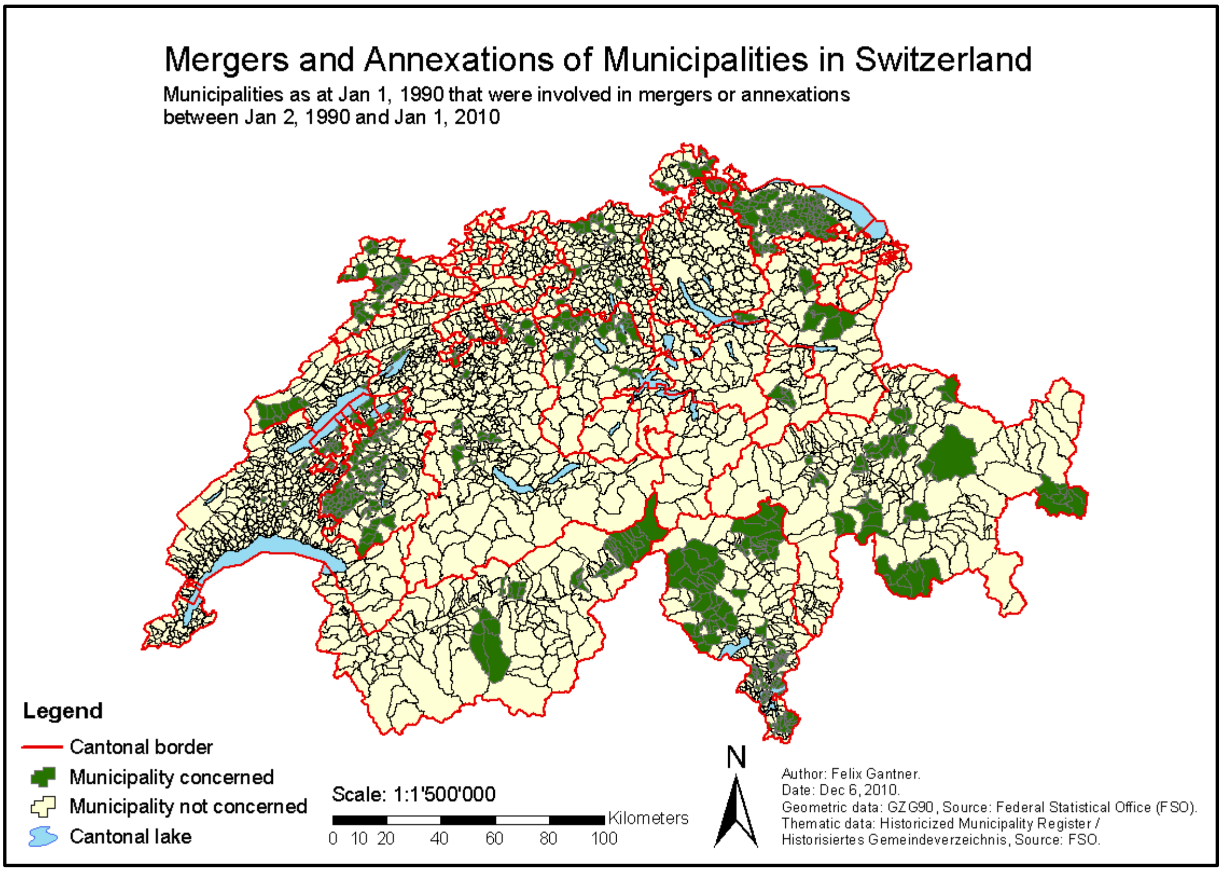


Figure 5.4: Mergers and annexations of municipalities in Switzerland between 1990 and 2010

6 Step 2: Construction of an application ontology

The prospective information system builds upon an ontology instead of a database. In contrast to databases, ontologies offer considerable advantages to information systems such as enhanced interoperability and integration of different datasets (Agarwal 2005). These were deciding factors since the data originate from two sources and use different conceptualizations. This chapter therefore describes the creation of the **S**patiotemporal **On**tology for the **Ad**ministrative **U**nits of **S**witzerland (SONADUS), an application ontology extending BFO. SONADUS integrates both the HMR+ and the geometric datasets and thus models the change history of the AUs of Switzerland between 1960 and 2010. Unlike the Finnish Spatio-Temporal Ontology (SAPO) (Kauppinen et al. 2008), SONADUS uses an upper ontology as a reference framework. Upper ontologies do not only facilitate interoperability and comparability between ontologies, but also provide a sound philosophical basis for the construction of an application ontology. Among the three upper ontologies described in Chapter 3.4.5, BFO has a relatively simple and symmetric structure. This is because BFO combines SNAP and SPAN and eschews multiple inheritance. The fact that BFO provides detailed descriptions of each class as comments in the OWL file facilitates its extension and represents another convincing reason for using this particular upper ontology. Since potential users are mostly Swiss, the concepts in the ontology were given German names. This also has the advantage that official Swiss terms do not have to be translated into the non-national language English. Nevertheless, this chapter provides accurate English translations for the names of the concepts used in this ontology.

6.1 Method

Step 2 describes the construction of an application ontology that traces the evolution of the AUs of Switzerland between 1960 and 2010. The design phase basically consisted of three stages. In a first stage, an ontology was drafted that is largely based on the data model of the HMR+. In a second stage, this draft was attached to BFO which required considerable modification of the class hierarchy and the relationships. Moreover, the second stage included the introduction of new concepts such as the creation of identities for AUs. In a final stage, the ontologies were visualized using the Protégé plug-in SOVA. The following paragraphs describe the procedure in detail as well as the materials used therein.

6.1.1 Materials

Protégé

The open source software Protégé (SCBIR 2010) is being developed at the Stanford Center for Biomedical Informatics Research and provides a development environment for the creation of OWL ontologies. This work used Protégé version 4.0.2 released on December 3, 2009. A careful study of the Protégé OWL Tutorial (Horridge 2009) was required to gain familiarity with the software.

Basic Formal Ontology (BFO)

BFO (Grenon and Smith 2004) is an upper ontology developed at the Institute for Formal Ontology and Medical Information Science (IFOMIS) (see Chapter 3.4.5). For this work, the freely available OWL file bfo-1.1.owl was downloaded from the IFOMIS Web page (IFOMIS 2010).

Simple Ontology Visualization API (SOVA)

SOVA (Kunowski 2010) is a Protégé plug-in that visualizes ontologies. Although there are a variety of tools for ontology visualization (for a review see Lanzenberger et al. 2010), the search for another plug-in compatible with Protégé 4.0 was unsuccessful.

Pellet

Pellet (Sirin et al. 2007) is an open source software to perform reasoning on OWL ontologies and to query the A-Box of a knowledge base. In contrast to other solutions, Pellet neither includes a triplestore nor provides a graphical interface. One must therefore use the command-line interface to send in queries or to perform reasoning. This work used Pellet version 2.2.2 that was downloaded from Clark & Parsia (2011).

6.1.2 Building a draft ontology

To obtain an overview of the basic entities that the HMR+ consists of, the conceptual data model of the HMR+ was transferred into an OWL ontology by using Protégé 4.0.2. Building the draft ontology included the creation of a class tree and the definition of object properties (see Figure 6.2). The draft ontology was then completed with entities that the HMR+ lacks. Examples of such entities are the geometries and the confederation as the uppermost AU in Switzerland. In a final step, object restrictions were imposed on each class in order to precisely define their properties.

6.1.3 Attaching the draft ontology to BFO

To enhance the comparability with other ontologies and to have a firm philosophical foundation to build on, the draft was linked to the upper ontology BFO. To be in line with the rules of BFO and to increase its expressivity, the ontology was subsequently extended with several classes and properties (see Figure 6.4). In doing so, it was defined that the ontology to be created must adhere to the rules of BFO in any case, even if this adds complexity to the structure. Due to the comprehensive rule set, it is referred to the BFO manual (Spear 2006) in this place. In addition, consider the comments on the classes in the OWL file of BFO.

Procedure

The procedure was composed of nine parts. First, the equivalent BFO categories of the root classes of the draft ontology were identified. This required substantial study of the comments on the classes of BFO. Second, these root classes including their branches were appended to the corresponding leaf classes of BFO. Subclasses, such as geometry, that BFO does not assign to the same branch were subsequently moved to the correct location within the class hierarchy. Since BFO accounts for both the SNAP and SPAN view of the world, certain entities must be modeled as continuants as well as occurrents. In this case, this concerned AUs that had already been classified as objects on the SNAP side. Hence, corresponding classes on the SPAN side had to be created for these objects in a third stage. Fourth, AUs were endowed with a unique identity. Whereas an AU maintains its identity in the case of incremental change, fundamental change causes either the creation or the loss of an identity. On the SNAP side, a separate branch was created for whole objects (WOs) that endure incremental change. On the SPAN side, the ontology was amended with an equivalent branch named life. The historicized objects of the draft that represent types of HMR+ entries were renamed partial objects administrative unit (POAUs) and subsumed under the class partial objects (POs). Instead of being solely classified as two-dimensional regions, geometries were also considered objects on the same level as POAUs. Fifth, partial objects geometry (POGs), a subclass of PO, were created to accommodate the fact that the geometries originate from independent data sources. To increase the expressiveness of the ontology, certain concepts of the HMR+ were subdivided into narrower terms in a sixth stage. For example, the change type formal renumbering was subdivided into formal renumbering of municipalities, formal renumbering of districts on the district level and formal renumbering of districts on the municipality level. Seventh, the ontology was amended with entities that the HMR+ lacks but that are necessary for the representation of the complete change history. This concerned, for instance, the change type final entry that represents the counterpart to initial entry and closes the evolution period. As soon as the final class hierarchy existed, object properties and restrictions were defined in an eighth stage to enable the establishment of relationships between classes. For example, it was stipulated that each POAU on the municipality level must have exactly one FSO municipality number. In a ninth stage, the only data property and restriction was defined to provide for temporal filtering in SPARQL queries. This restriction requires instances of date classes to have a data type property to store dates as XML data type date. Finally, the ontology was named SONADUS, an acronym standing for **S**patiotemporal **On**tology for the **Ad**ministrative **U**nits of **S**witzerland.

6.1.4 Visualizing the ontologies

To get an idea of their structure and extensiveness, both the draft ontology and SONADUS were visualized using the Protégé plug-in SOVA (see Figure 6.6 and Figure 6.7). When downloading and storing the .jar-file in the Protégé plug-in folder, SOVA automatically creates visualizations of OWL ontologies in a separate tab. Since ontologies consist of only the T-Box, individuals were omitted and not visualized. In the draft ontology, overlapping labels of classes and properties were manually moved. In case of SONADUS, the great number of classes made it impossible to avoid overlapping labels. Since SOVA 0.6.0 does not have a print function, a screen shot of the visualizations had to be taken in order to insert them into this file.

6.1.5 Reasoning

Queries to an ontology or a knowledge base only return explicitly stored information. Retrieving all instances of the class change (Mutationsprozess), for example, may require that indirect subclasses have a subClassOf-relationship with the class change. The asserted ontology that was built in Protégé, however, only comprises direct subClassOf-relationships. Hence, the result set of such a query would be incomplete. This stems from the fact that only one subclass of change (Mutationsprozess) is a leaf class, and thus contains individuals. Reasoning was therefore performed to infer the lacking indirect subClassOf-relationships. Having these indirect subClassOf-relationships enables users, for instance, to easily select all individuals of the children classes of change. To perform reasoning, this work used the OWL reasoner Pellet (Sirin et al. 2007). The command in Figure 6.1 was typed in the command-line interface. Note that this command caused Pellet to perform reasoning on all statements instead of only the classes. Consequently, Pellet also deduced statements such as subPropertyOf-relationships. Last, the resulting ontology was named SONADUSi, i.e., SONADUS including the inferred axioms (see Chapter 6.5).

Form of the command

pellet.bat extract --statements "AllStatements" OWLFILE 1> OUTPUTFILE

Figure 6.1: Example of a command to perform reasoning with Pellet

6.2 Draft ontology

6.2.1 Overview

The draft ontology consists of a class tree enthroned by the OWL class Thing. By definition, Thing collects all individuals and subsumes the entire set of subclasses. The class tree shown in Figure 6.2

can be subdivided into three major branches formed by the root classes object (Objekt), attribute (Attribut), and change (Mutationsprozess). Each of them subsumes abstract classes that again contain either abstract subclasses or leaf classes. The whole ontology consists of 50 classes, 21 object properties, and 0 data type properties and has the DL expressivity ALCRIQ. ALCRIQ is a composition of the four elements ALC, R, I, and Q. The acronym ALC refers to the **A**ttributive Concept **L**anguage with **C**omplements, a description logic that includes the constructors conjunction, disjunction, negation, existential restriction, and value restriction (Baader et al. 2007). R, I, and Q represent extensions to ALC to increase the expressivity of an ontology. Aside from ALC, the draft ontology thus additionally expresses the inversion of roles (I) (Baader 2003) and qualified number restrictions (Q) (Baader et al. 2007). Furthermore, R denotes the intersection of roles (Baader 2003), role disjointness (Horrocks et al. 2006), reflexivity and irreflexivity of roles, local reflexivity of roles, negated roles, complex roles, and universal roles.

6.2.2 Class structure and relationships

Figure 6.2 reveals that the conceptualization of the draft ontology still resembles the conceptual model of the HMR+, aside from the changes. The object branch reflects the division of the HMR+ into three tables and the historization of AUs on the level of districts and municipalities. Besides the confederation (Bund) that completes the set of AUs, a leaf class in the object branch represents an entity of the HMR+ classified according to the type of entry. Hence, the attribute branch contains the attributes of all three HMR+ tables and the geometry attached as an additional attribute. The change branch finally contains a class hierarchy of the different change types described in FSO (2007b). This differs fundamentally from the HMR+ that records changes only implicitly. Due to this implicit storage, the determination of the type of a change requires the comparison between the types of registration and the types of annulment. Apart from the class tree, the draft ontology includes object properties and restrictions though Figure 6.2 does not depict them. On the one hand, the restrictions define the range of attributes that an AU must have, and on the other hand ensure that an AU is related to its predecessor(s) and successor(s) via a change. In addition, object restrictions enforce that any AU, except for the confederation, is part of exactly one AU on the next higher level. Inverse restrictions impose that, aside from municipality-level entities, any AU comprises at least one unit on the next lower level.

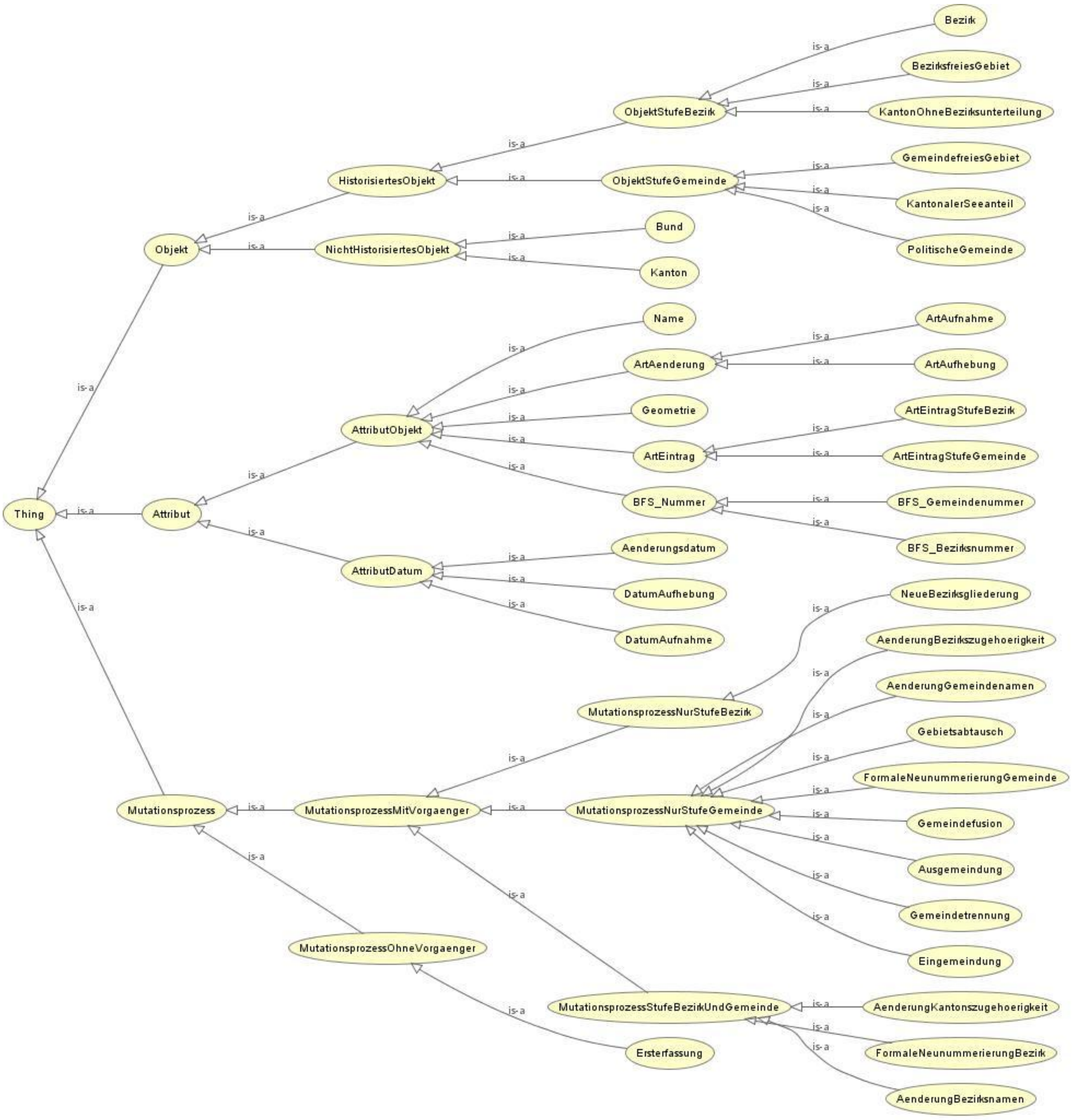


Figure 6.2: The class hierarchy of the draft ontology

6.3 Application ontology SONADUS

6.3.1 Overview

SONADUS is the result of appending the draft ontology to BFO and amendments such as the construction of identities and the assignment of geometries to AUs. As the figures in Table 6.1 demonstrate, using BFO led to an explosion of classes and object properties compared to the draft. While the number of object properties doubled, the number of classes increased by almost five times. In contrast to the draft, SONADUS also contains a data property whose purpose is to store dates in the XML format 'YYYY-MM-DD'. Due to this fact, the DL expressivity of SONADUS is ALCRIQ(D) where (D) means that the ontology uses data types properties. Otherwise, the DL expressivity has not changed in comparison with the draft ontology.

Ontology metrics and statistics	
Class count	241
...thereof OWL classes	1
...thereof original BFO classes	39
...thereof original SNAP classes	21
...thereof original SPAN classes	17
...thereof added SNAP classes	126
...thereof added SPAN classes	75
...thereof SNAP classes	147
...thereof SPAN classes	92
...thereof SNAP classes with instances	67
...thereof SPAN classes with instances	52
Object property count	42
...thereof object properties with instances	30
Data property count	1
DL expressivity	ALCRIQ(D)

Table 6.1: Metrics and statistics of SONADUS

6.3.2 Class structure and relationships

The application ontology SONADUS consists of a class tree, object and data type properties and restrictions, as well as a number of additional restrictions such as cardinality. Figure 6.4 and the poster in the pocket on the back cover depict the class tree of the ontology. Figure 6.5 presents an overview of the general structure and the extensiveness of SONADUS. The following paragraphs describe the structure of the ontology, its basic entities, and the underlying thoughts.

General class structure

BFO stipulates the distinction between continuants and occurrents. While continuants have qualities, occurrents extend in time and thus consist of multiple temporal parts. AUs as being modeled in the HMR+ both have characteristics of continuant and occurrent entities. Hence, SONADUS provides classes for AUs as objects on the one hand, and as lives on the other hand. Changes and attributes, on the contrary, do not have to be modeled both as SNAP and SPAN classes. While changes are subsumed under the SPAN class Process, attributes are attached to the SNAP class Quality. However, this does not apply to temporal attributes and the geometries. Temporal attributes must be SPAN entities because they refer to either an instant or period in time and are thus part of a temporal region that itself may be composed of temporal parts. Just as temporal attributes represent a section in time, geometrical attributes correspond to a region in space. More precisely, the geometries represent an absolutist two-dimensional region that is independent from the corresponding AU.

Identity and change

SONADUS adopts an approach that differentiates between fundamental and incremental change as well as between *whole objects* (WOs) (GanzesObjekt) and *partial objects* (POs) (Teilobjekt). From an object-oriented view, WOs are objects with an identity while POs correspond to the different versions of a WO. Due to two different data sources, one version cannot grasp the whole reality at a given time. Hence, there are two kinds of POs: *partial objects administrative unit* (POAUs) (TeilobjektVerwaltungseinheit) and *partial objects geometry* (POGs) (TeilobjektGeometrie). While a POAU represents an entity of the HMR+, a POG is a single polygon of one of the geometric datasets. On the contrary, WOs and *lives* (Leben) are novel concepts of SONADUS and accommodate the fact that AUs preserve their identity when undergoing incremental change. In fact, a life begins and ends with a fundamental change but may be divided into several *periods of life* (PLs) (Lebensabschnitt) evoked by incremental change. PLs are the SPAN counterparts to POs and represent the periods between two consecutive changes that both can be either fundamental or incremental. However, the class hierarchy does not reflect this distinction. Depending on the role of the AU, certain change types entail fundamental as well as incremental change. This is subsequently shown by using the example of an annexation.

Example: Zofingen annexes Mühlethal

The City of Zofingen annexed the neighboring Municipality of Mühlethal on 01/01/2002. As a consequence, the first PLAU, the sixth PLG, as well as the life of Mühlethal were ended, whereupon the WO of this municipality had to incur a loss of identity. In contrast, the same change did not end the life of Zofingen but just replaced its two PLs. The life of Zofingen thus moved to the next stage while the identity of the corresponding WO remained unchanged and has continued to exist (see Figure 6.3).

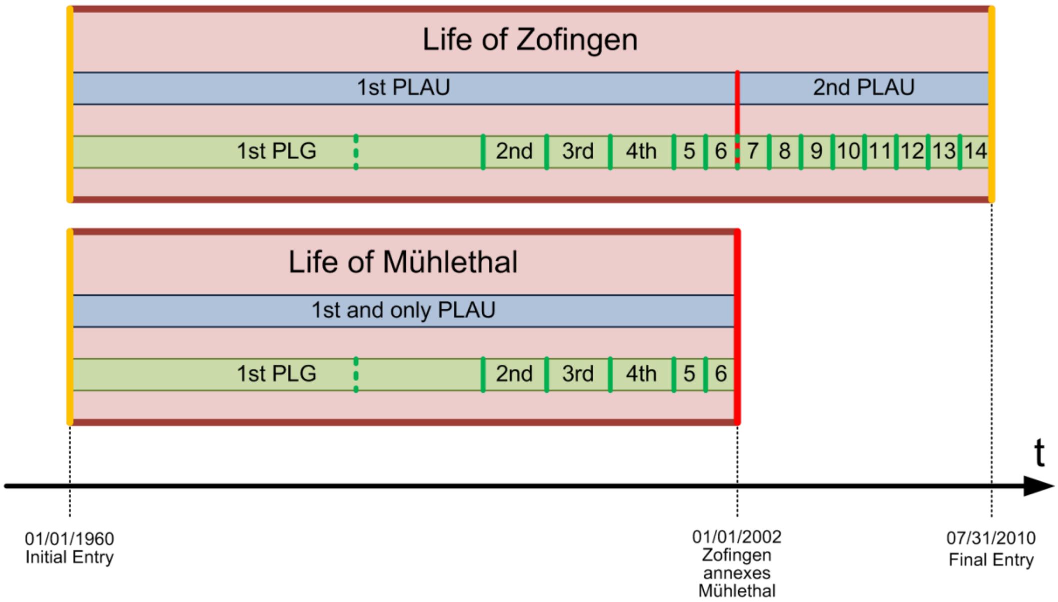


Figure 6.3: Annexation: maintenance and loss of an identity (see legend in Figure 6.5)

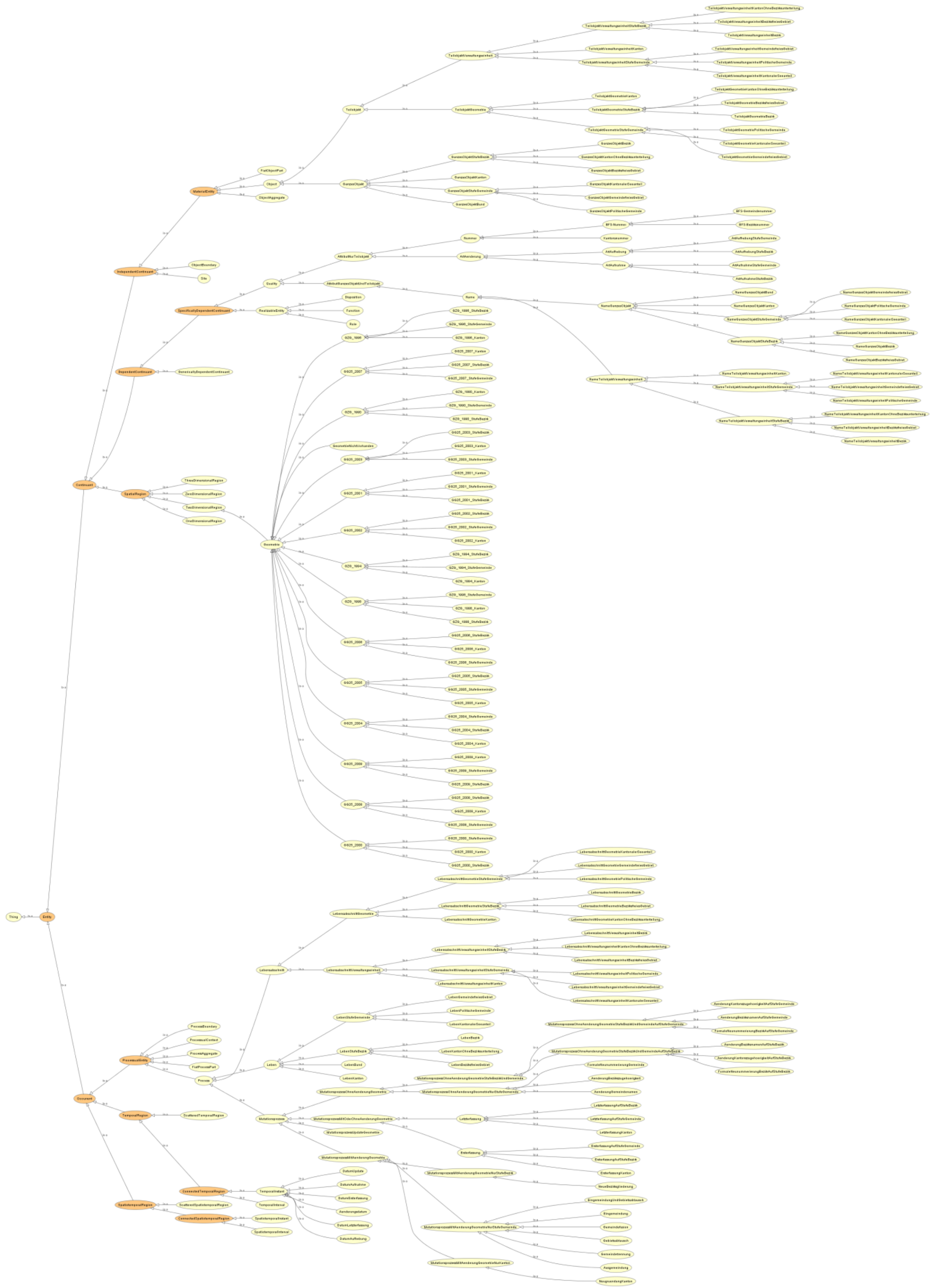


Figure 6.4: The class hierarchy of SONADUS

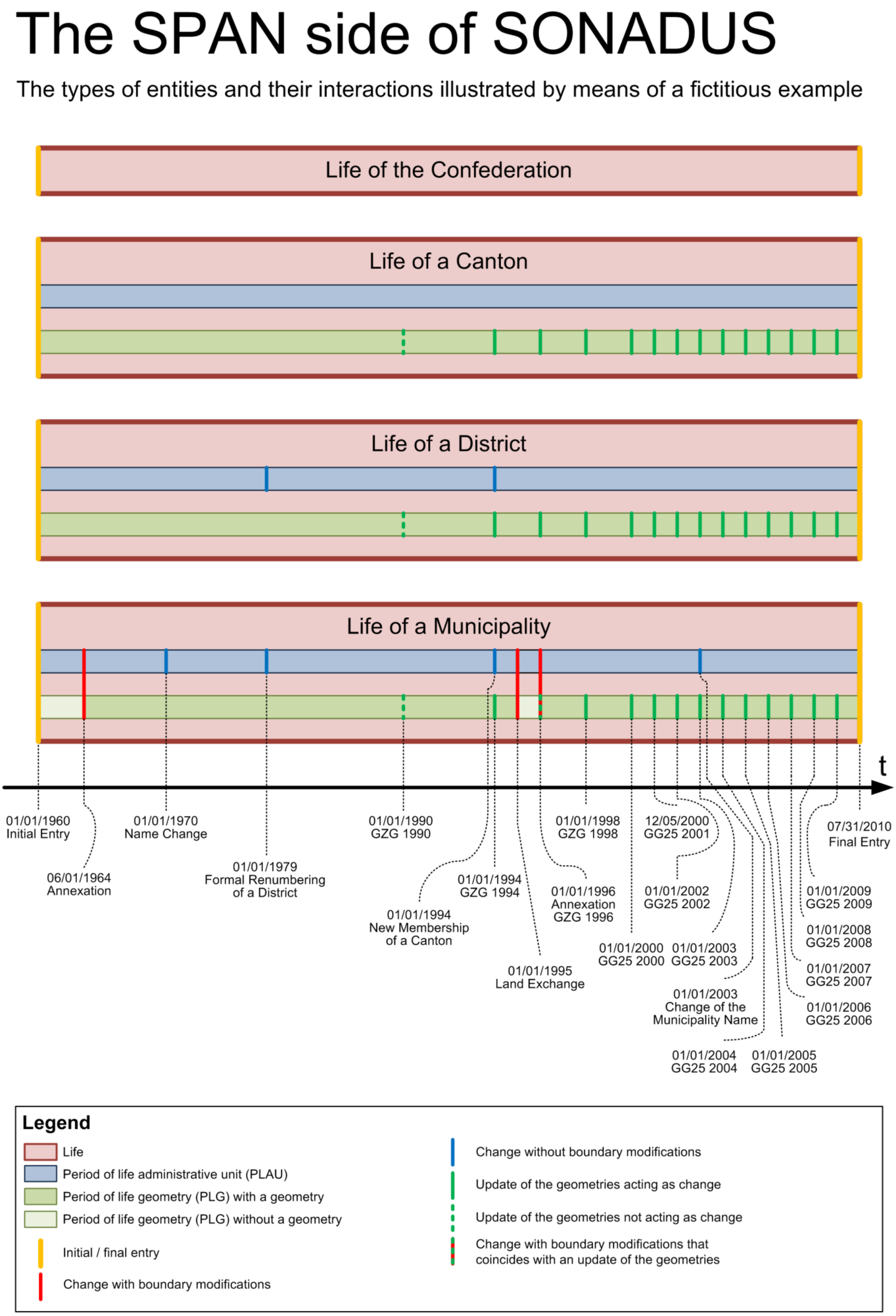


Figure 6.5: The SPAN side of SONADUS: the types of entities and their interactions by means of a fictitious example

Representation of the geometries

According to the philosophical criteria of BFO, geometries are a subcategory of the class TwoDimensionalRegion. Since the geometries originate from another data source than the historicized AUs, SONADUS accommodates their special status and independence from the HMR+. Consequently, geometries are not only two-dimensional regions, but also objects and processes. While geometries as objects are considered POGs, geometries as processes are deemed PLGs. This is not only done for philosophical reasons, but also because the validity intervals of PLGs and PLAUs do not match. To link a POAU with the corresponding POG, SONADUS establishes a relationship between those PLs of a life that temporally overlap. Therefore, a retrieval of the correct geometry or geometries of a POAU requires the selection of those PLGs that bear a temporallyOverlap-relationship with PLAUs. In Figure 6.3, for instance, the second PLAU of Zofingen has such relationships with the seventh to the fourteenth PLG. The incongruity between the PLAUs and the PLGs makes evident that the geometries cannot be merely appended to objects as attributes. This would only work if the HMR+ stored the valid geometry together with the thematic and the temporal data.

Periods of life geometry (PLGs)

While PLAUs represent the temporal region occupied by an entry of the HMR+, PLGs usually last from one update of the geometries to the next. As Figure 6.5 shows, there are several exceptions to this rule. First of all, a spatial change between two updates, such as the land exchange on 01/01/1995, ends a PLG because the geometry becomes invalid. If possible, the start date of the subsequent PLG is brought forward to the date of the spatial change. In case of the land exchange, an annexation coinciding with the next update makes this impossible. Hence, the period between the land exchange and the annexation is covered by a PLG for which no geometry exists. In other cases, however, advancing the start date of PLGs fills gaps in the change history. Figure 6.5, for example, reveals that bringing the start date forward to the annexation in 1964 extends the validity interval of the PLG by more than 25 years.

The extension of PLGs is based on the underlying assumption that the significance of geometric change is fairly low compared to spatial change. Therefore, SONADUS prioritizes closing gaps even though geometric change, such as boundary modifications in uninhabited areas, may occur during the period of extension. On the contrary, this lack of knowledge about geometric change requires the repeated storage of POGs, PLGs, and actual geometries every time a new update comes out. Since this also concerns AUs that do not undergo geometric change, SONADUS has to accept massive redundancy.

Change types

As Table 6.2 shows, SONADUS differentiates between 24 types of change, an increase of 11 types in comparison to the HMR+. This also contrasts sharply with the classification of Kauppinen et al. (2008) that includes seven change types of the AUs of Finland. Nevertheless, this difference should not obscure the fact that the large number of change types in SONADUS results from the increased level of detail rather than from wide disparities between these two countries. SONADUS, more precisely, makes a distinction between the hierarchical levels, expands the set of change types to the level of cantons, opens and ends the change history, corrects errors in the source data, and finally contains purely administrative change types such as formal renumbering. For these reasons, SONADUS contains six completely new change types that Table 6.2 depicts in orange and one new change type in red that corrects an error in the HMR+ .

Change types in SONADUS	English translation	Former change type in the HMR+	Difference
NeueBezirksgliederung	Restructuring of districts	Restructuring of districts	none
Ausgemeindung	Secession	Secession	none
Eingemeindung	Annexation	Annexation	none
Gebietsabtausch	Land exchange	Land exchange	none
Gemeindefusion	Merger	Merger	none
Gemeindetrennung	Separation	Separation	none
AenderungBezirkszugehoerigkeit	Change of membership of a district	Change of membership of a district	none
AenderungGemeindenamen	Change of municipality name	Change of municipality name	none
FormaleNeunummerierungGemeinde	Formal renumbering of a municipality	Formal renumbering of a municipality	none
ErsterfassungAufStufeBezirk	Initial entry on the district level	Initial entry	Breakdown according to the hierarchical level
ErsterfassungAufStufeGemeinde	Initial entry on the municipality level	Initial entry	Breakdown according to the hierarchical level
AenderungBezirksnamenAufStufeBezirk	Change of district name on the district level	Change of district name	Breakdown according to the hierarchical level
AenderungKantonszugehoerigkeitAufStufeBezirk	Change of membership of a canton on the district level	Change of membership of a canton	Breakdown according to the hierarchical level
FormaleNeunummerierungBezirkAufStufeBezirk	Formal renumbering of a district on the district level	Formal renumbering of a district	Breakdown according to the hierarchical level
AenderungBezirksnamenAufStufeGemeinde	Change of district name on the municipality level	Change of district name	Breakdown according to the hierarchical level
AenderungKantonszugehoerigkeitAufStufeGemeinde	Change of membership of a canton on the municipality level	Change of membership of a canton	Breakdown according to the hierarchical level
FormaleNeunummerierungBezirkAufStufeGemeinde	Formal renumbering of a district on the municipality level	Formal renumbering of a district	Breakdown according to the hierarchical level
NeugruendungKanton	Creation of a new canton	none	new
ErsterfassungKanton	Initial entry of a canton	none	new
LetzterfassungAufStufeBezirk	Final entry on the district level	none	new
LetzterfassungAufStufeGemeinde	Final entry on the municipality level	none	new
LetzterfassungKanton	Final entry of a canton	none	new
MutationsprozessUpdateGeometrie	Update of geometries	none	new
Eingemeindung und Gebietsabtausch	Annexation and land exchange	Annexation	new - error correction

Table 6.2: Change types of SONADUS

Spatial and geometric change

SONADUS draws a crucial distinction between spatial and geometric change. Spatial change, on the one hand, encompasses those change types of the HMR+ that evoke boundary modifications. This includes the change types merger, separation, annexation, secession, land exchange, and restructuring of districts in a canton, as well as the change type annexation and land exchange. Geometric change, on the other hand, refers to one of the 14 updates of the geometries between 1990 and 2009. As the data analysis has indicated, geometric change often coincides with spatial change, particularly since these updates have come out annually. Still, spatial change occurs between two updates and thus deserves special treatment as described above. Another aspect of geometric change is the ignorance of its form. On the one hand, geometric change may mirror spatial change. This is apparent from the example in Figure 6.5 where the GZG 1996 provide an updated geometry of the municipality after the annexation. On the other hand, geometric change includes smaller boundary modifications in uninhabited areas that the HMR+ does not record. Last, the vast majority of geometric changes simply replaces the old geometry with an identical copy of the new dataset. In contrast to spatial change that always refers to real events, geometric change can take multiple forms whose determination requires the comparison between two consecutive geometries.

Initial and final entry

Initial entry (Ersterfassung) is a change type of the HMR+ that stands for the registration of all AUs that existed on 01/01/1960. Since this date, the HMR+ records the change history of the AUs of Switzerland. Final entry (Letzterfassung) is a novel concept introduced with SONADUS and represents the counterpart to initial entry. Instead of leaving the end open for current AUs, last entry provides SONADUS with a change type that closes the history on 07/31/2010. This brings users the advantage that the knowledge base explicitly contains the end of the recording time. Users may otherwise automatically assume that AUs without a date of annulment still exist.

Annexation and land exchange

Annexation and land exchange (EingemeindungUndGebietsabtausch) is a change type to rectify the error of classifying change #2299 as an annexation (see Chapter 5.2.3). This auxiliary class is therefore subject to special restrictions that represent a combination of the restrictions imposed on the classes land exchange and annexation. These special restrictions allow multiple municipalities to endure the change. Since the FSO (2007b) placed the same restrictions upon annexations, SONADUS forbids annexations to have more than one successor in order to avoid ambiguity. However, this concerns only change #1077 (see Chapter 5.2.3) in which two municipalities annexed a kommunanz. Due to the error in change #2299, one may also interpret that the first of both municipalities annexed the kommunanz while the second ceded a part of its territory to the first.

Data type property

SONADUS uses its only data type property for the purpose of storing dates in the XML format 'YYYY-MM-DD'. Each instance of a date class is thus required to have a data type property date. This not only leads to a duplication of every date, but also of the corresponding properties. Object properties that connect instances of the classes Process and TemporalInstant do not suffice to temporally reference entities. Hence, SONADUS has an additional data type property that, besides creating redundancy, complicates querying the ontology and knowledge base.

6.4 Visualizations of the ontologies

The visualizations show the class structure, the properties, and the restrictions of SONADUS but completely omit individuals (see Figure 6.6 and Figure 6.7). Instead of listing the full set of elements, this section confines itself to describing the most important ones and refers to the complete legend that is available online (Kunowski 2010). SOVA depicts classes in teal, object properties in dark purple, data type properties in light green, and the restrictions used here in magenta. While empty arrows point to the respective superclass, converse arrows indicate disjointness, and inverse arrows show equivalence relationships between two classes. Since these visualizations represent screenshots, the readability of the labels is quite poor even when enlarging the image. To use the full power of visualization, one must therefore explore the visualized ontology in Protégé itself. Functions to zoom in and out or to hide certain elements greatly facilitate the analysis of an ontology.

6.4.1 Visualization of the draft ontology

Since the draft ontology consists of relatively few elements, the labels could be arranged in a way that they do not overlap (see Figure 6.6). On closer examination, one can still recognize the three branches that each form a cluster of classes. The change branch is situated in the upper half of the graphic while objects and attributes are located in the lower left and the lower right of the visualization, respectively.

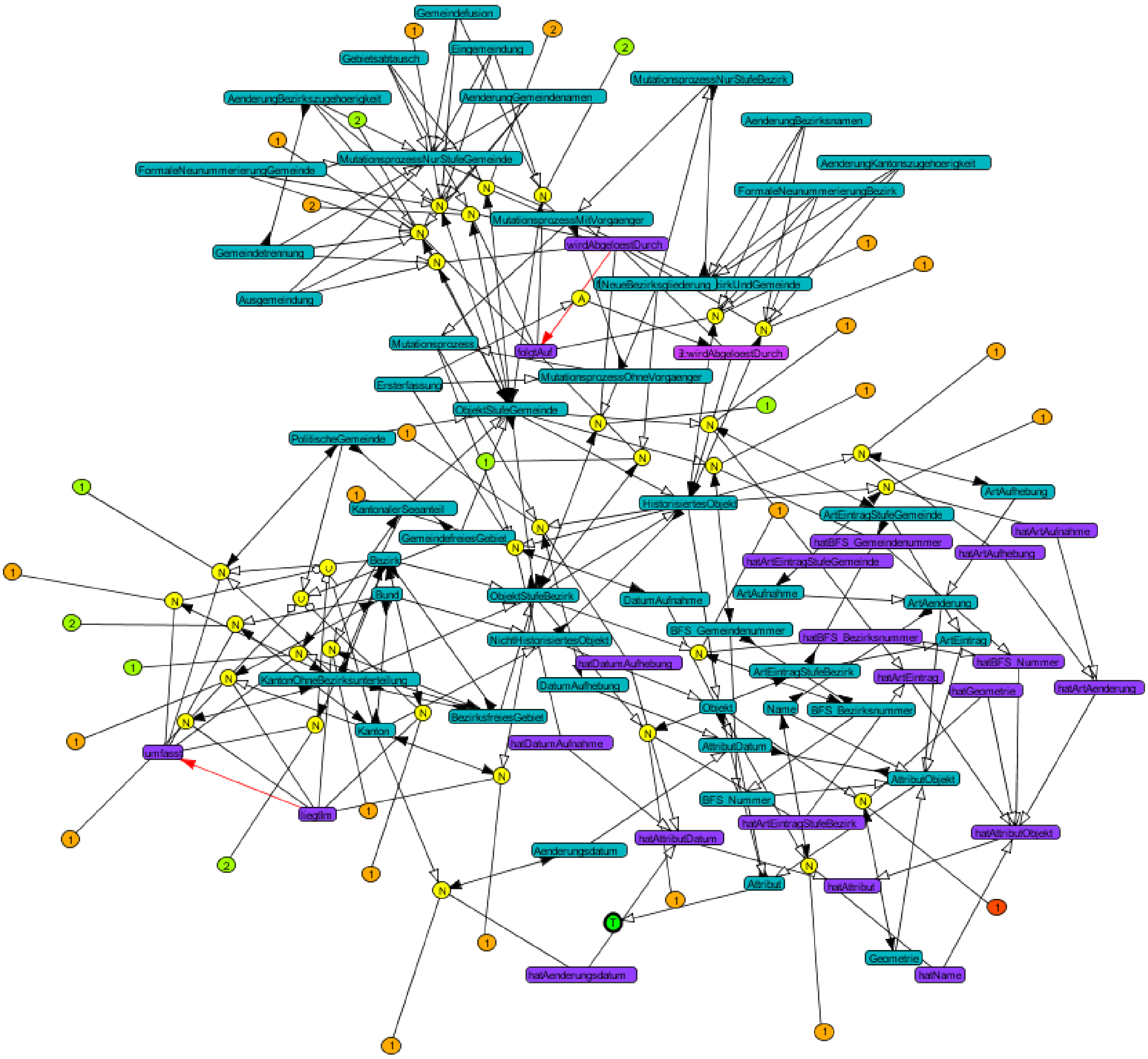


Figure 6.6: Visualization of the draft ontology

6.4.2 Visualization of SONADUS

The visualization of SONADUS (see Figure 6.7) reveals the actual size of this ontology that has considerably increased compared to the draft. Although overlaps of labels become inevitable, it is still possible to discover basic patterns in the ontology. First of all, there is a large compact cluster that contains both classes and properties as well as quite a number of restrictions. Second, the main cluster is surrounded by a belt of lower density and connected with an isolated crescent-shaped cluster in the lower left corner. When increasing the scale, one can see that the main cluster represents the core of SONADUS. This core comprises the classes, properties, and restrictions required for the separation of lives and objects, the construction of identities, and the connection of the HMR+ with the geometric datasets. The surrounding belt can be subdivided into two parts. While the upper left corner contains the attribute classes and the corresponding properties, the part below the main cluster primarily encompasses unused BFO classes. The teal crescent, finally, represents the class cluster of the geometries that are subclasses of TwoDimensionalRegion. The isolation from the rest reflects the fact that only one object restriction connects this group with the class POG (TeilobjektGeometrie) in the main cluster. Aside from patterns in the structure, it also becomes apparent that SONADUS uses significantly more restrictions compared to the draft, as the number of labels in magenta indicates. Eventually, it has to be said that SOVA fails to visualize all disjoint-relationships. In the outer crescent, SOVA depicts only a third of the mutual disjoint-relationships that hold between the classes. In the inner crescent, SOVA does not even visualize them at all though there are disjoint-relationships between all classes of the inner crescent.

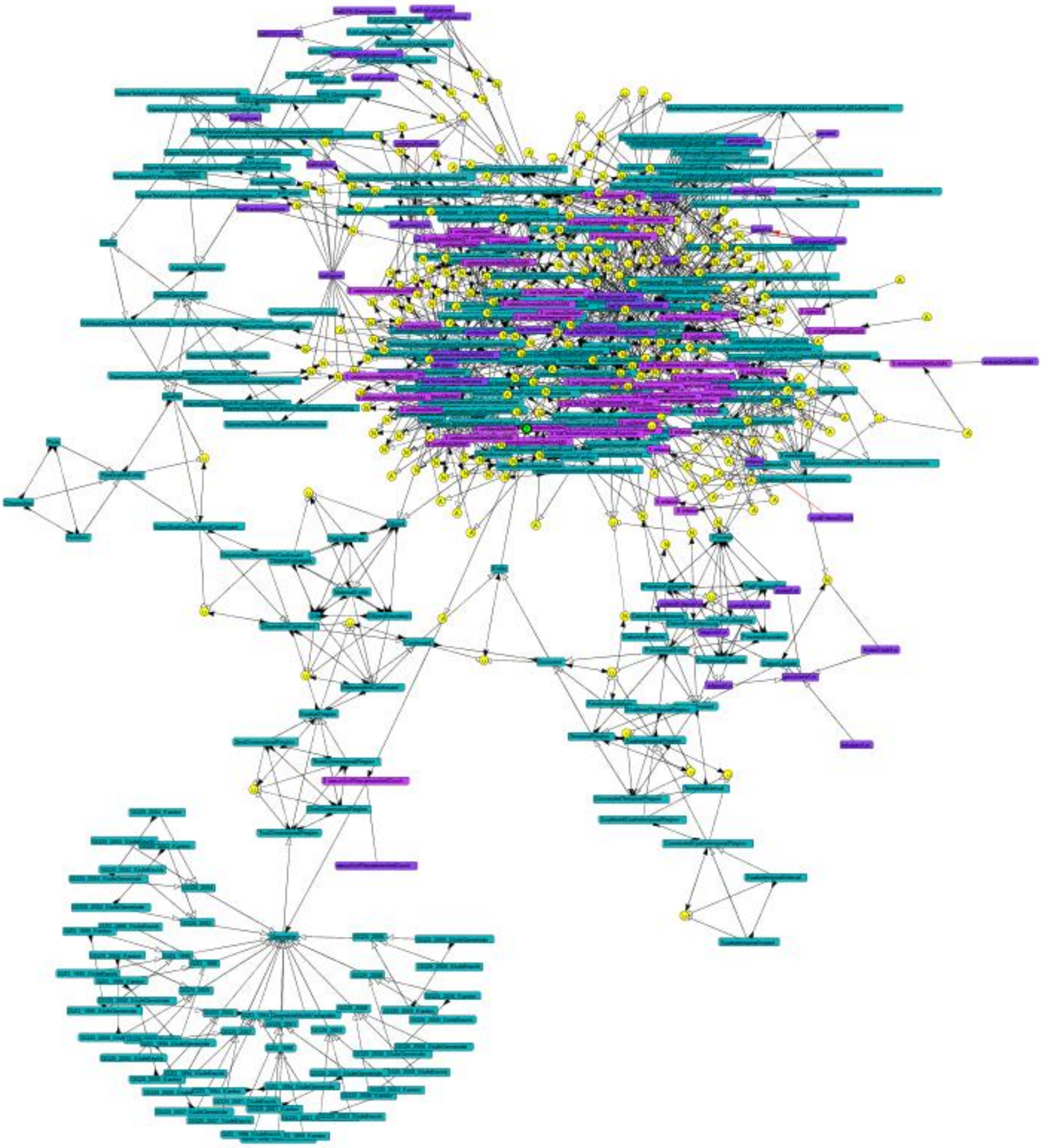


Figure 6.7: Visualization of SONADUS

6.5 Inferred ontology (SONADUSi)

In contrast to the original, asserted version of SONADUS, SONADUSi additionally contains the inferred axioms and is thus 20 times larger. While the storage of the asserted ontology requires 257 KB, the inferred ontology necessitates 5’198 KB. These figures show that reasoning (see Chapter 6.1.5) indeed generates a large amount of new explicitly-defined knowledge. At the same time, they also indicate increasing storage demands. Moreover, reasoning on SONADUS requires considerable computing power and becomes infeasible when performed on the entire knowledge base.

7 Step 3: Creation of a knowledge base

Until now, this work has largely focused on the establishment of a conceptual model for the integration of two different datasets. The application ontology SONADUS represents the result of this step. SONADUS lays the theoretical groundwork for a knowledge base that supports queries on the change history of the administrative units (AUs) of Switzerland. Hence, Step 3 describes the establishment of a knowledge base using SONADUSi as T-Box.

7.1 Method

Step 3 commenced with the definition of number intervals for groups of individuals to ensure the uniqueness of every instance in the knowledge base. Such groups of individuals often correspond to the set of instances of a class, such as in case of the attribute FSO municipality number. In other cases, such as the change number of districts, SONADUS does not have an equivalent class that encompasses this group of individuals, i.e., all change types on the level of districts. As soon as the number intervals were defined, SQL queries were sent to the database created in Chapter 5.1.1. Each query returned the required information in three columns. These columns were then exported into CSV files. At last, a Java program was written to encode the statements in the CSV files into RDF/XML syntax. The text files containing the knowledge in RDF/XML syntax were then merged with SONADUSi.

7.1.1 Definition of number intervals for individuals

Just as classes and properties, individuals are identified by a unique URI. Since all concepts and individuals share the same namespace (http://www.wsl.ch/sonadus), the URI must be made unambiguous through the assignment of a unique fragment identifier. This could have been both a name or a number. However, due to the kind of data, the structure of SONADUS, and the huge number of individuals, all but one group of instances were assigned numbers. More precisely, the data contained names of AUs with blank spaces (e.g., Avegno Gordevio) and forward slashes (e.g., Biel/Bienne). Such characters are not suited to be part of a URI and must be replaced. Furthermore, SONADUS differentiates between the life and the object of an entity while there is only one name of each entity. Hence, names would have had to be amended with a designation such as the class name, e.g., Basel_PLAU and Basel_POAU. Finally, it was easier to assign unique numbers to a huge number of individuals than to give them unique names. For the sake of simplicity, the names of POAUs and WOs were not additionally stored as numbers. In fact, these names were modified to be in accordance with the URI naming conventions.

In order to define unique number intervals for all groups of individuals, intersections of number intervals in the source data had to be eliminated in a first stage. The number 3, for example, is both the canton number of Lucerne and the FSO municipality number of Bonstetten. In order to disambiguate these two types of numbers, FSO numbers were added 70’000 and canton numbers were added 80’000. The same was done with other types of numbers until each of them had a unique interval with five-digit numbers. In a second stage, the disambiguated historization and canton numbers were defined as identifiers of POAUs. The corresponding PLAUs were assigned the same numbers raised by 100’000. Since WOs and lives represent new concepts of SONADUS and are thus not present in the data, unique numbers for both had to be generated in a third stage. To emphasize the mutual relationship between WOs and lives, the two intervals were made the same, except that the numbers of lives were put up by 100’000 again. This procedure is illustrated in Table 7.1 by means of the Municipality of Bargen (SH).

Example: the non-geometric entities of Bargen (SH)

Class	Prefix		Original code		Unique identifier
POAU	none		11688	→	11688
PLAU	1	+	11688	→	111688
FSO number	7	+	2931	→	72931
Class	**Prefix**		**Generated code**		**Unique identifier**
WO	none		40178	→	40178
Life	1	+	40178	→	140178

Table 7.1: Definition of number intervals for non-geometric entities

After the definition of non-overlapping intervals for HMR+ entities, every geometry-related entity was assigned a unique identifier in a fourth stage. In doing so, the primary keys of the geometric datasets, i.e., the object IDs, hereby served as a basis. Due to the repeated use of the same object IDs, these had to be prefixed with the respective year of publication. Then, a digit was added that indicates whether the identifier is a POG or a PLG. To ensure that the intervals between the three hierarchical levels do not overlap, the identifier was amended with a last digit. Since object IDs consist of up to eight digits, each POG and PLG has a 14-digit identifier that is comprised of the year of publication, the entity type, the hierarchical level, and the object ID. In contrast to POGs and PLGs, the geometries as two-dimensional regions must point to the actual object in the corresponding shapefile. Hence, the identifier is composed of the name of the shapefile, a colon as separator, and the object ID. Using the example of Bargen (SH) again, Table 7.2 illustrates the definition of number intervals for geometry-related entities. The number intervals of all groups of individuals are listed in Appendix C Number intervals.

Example: the geometric entities of Bargen (SH)

Class	Year		SNAP/SPAN		Hierarchical Level		Object ID		Unique identifier
POG	2009	+	0	+	0	+	248847	→	20090000248847
PLG	2009	+	1	+	0	+	248847	→	20091000248847
Class	**Dataset**		**Separator**				**Object ID**		**Unique identifier**
Geometry	gg25_09_a_ ch_gde.shp	+	:		+		248847	→	gg25_09_a_ch_gde.shp: 248847

Table 7.2: Definition of number intervals for geometric entities

7.1.2 Extraction of triples with SQL queries

To construct an RDF graph in the form < subject, predicate, object >, SQL queries were issued to the database created in Chapter 5.1.1. Each query returned three columns, so that every row of the result set represented a triple (see Figure 7.2). Besides simple queries, such as the example in Figure 7.1, the extraction of triples required a lot of preliminary work. In addition to the three HMR+ tables and the 14 tables of the geometric datasets, five auxiliary tables were created. The auxiliary tables store information that the source data only implicitly contain. Two of these five tables served to store the WOs and the lives of districts and municipalities as described below. The three others recorded the PLGs and, in chronological order, the corresponding PLAUs with which they bore a temporallyOverlap-relationship. In general, the SQL queries were aimed to be reusable over time. However, due to certain exceptions and errors in the source data (see Chapter 5.2.3), this was not possible in every case.

SQL Window - IndividuumTypTeilobjektVerwaltungseinheit.sql

SQL | Output | Statistics

```sql
-- Individuum
-- Typ
-- TeilobjektVerwaltungseinheit
-- Anzahl insgesamt: 5643

SELECT * FROM
(
-- Stufe Gemeinde, Anzahl: 5332
(SELECT G.HIST_NR_GDE AS SUBJECT,
'type' AS PREDICATE,
CASE WHEN G.ART_EINTRAG = 11
    THEN 'TeilobjektVerwaltungseinheitPolitischeGemeinde'
  WHEN G.ART_EINTRAG = 12
    THEN 'TeilobjektVerwaltungseinheitGemeindefreiesGebiet'
  WHEN G.ART_EINTRAG = 13
    THEN 'TeilobjektVerwaltungseinheitKantonalerSeeanteil'
    ELSE 'BUG' END AS OBJECT
FROM HIST_GDE_VZ G)

UNION
-- Stufe Bezirk, Anzahl: 285
(SELECT B.HIST_NR_BEZ+20000 AS SUBJECT,
'type' AS PREDICATE,
CASE WHEN B.ART_EINTRAG = 15
    THEN 'TeilobjektVerwaltungseinheitBezirk'
  WHEN B.ART_EINTRAG = 16
    THEN 'TeilobjektVerwaltungseinheitKantonOhneBezirksunterteilung'
  WHEN B.ART_EINTRAG = 17
    THEN 'TeilobjektVerwaltungseinheitBezirksfreiesGebiet'
    ELSE 'BUG' END AS OBJECT
FROM HIST_BEZ_VZ B)

UNION
-- Kanton, Anzahl: 26
(SELECT K.KT_NR+50100 AS SUBJECT,
'type' AS PREDICATE,
'TeilobjektVerwaltungseinheitKanton' AS OBJECT
FROM HIST_KT_VZ K)
)
```

Figure 7.1: SQL query to classify HMR+ entries according to their type of entry

	SUBJECT	PREDICATE	OBJECT
1	10001	type	TeilobjektVerwaltungseinheitPolitischeGemeinde
2	10002	type	TeilobjektVerwaltungseinheitPolitischeGemeinde
3	10003	type	TeilobjektVerwaltungseinheitPolitischeGemeinde
4	10004	type	TeilobjektVerwaltungseinheitPolitischeGemeinde
5	10005	type	TeilobjektVerwaltungseinheitPolitischeGemeinde

29:5 0:03 5643 rows selected in 3.751 seconds

Figure 7.2: A part of the result set in triple form

Construction of WOs and lives

Based on the types of registration and annulment in Table 5.1, type 21 leads to the creation and type 29 causes the loss of an identity. To construct lives in the HMR and the HDR, entries with a type of registration were selected first. This selection also included entries with a type of registration 20 to consider AUs that had already existed before the creation of the HMR+. This result set was subsequently joined with the entire dataset on the change number annulment and the change number registration, respectively. Where the type of annulment was 29, the change number annulment was set to null in order to avoid joins of abolished AUs with their successors. The result set was then joined with the whole dataset again, until no entry had a change number annulment other than 29 or null. To verify the results, it was checked whether the total number of entries had changed after a join.

Construction of POGs and PLGs

Just as WOs and lives, POGs and PLGs represent new concepts of SONADUS that are not explicitly present in the source data. Since SONADUS puts POGs and PLGs on the same level as POAUs and PLAUs, three tables were compiled that represent the geometric equivalents to the registers of the HMR+. Analogous to the HMR+, the three tables do not differentiate between POGs and PLGs but store them together as one entry. For each of these entries, however, the geometric equivalents record the corresponding HMR+ entries that they temporally overlap with. The actual creation of the POGs and PLGs was a difficult task that required the formulation of queries of more than 5’500 lines in case of the municipalities. Aside from the cantons, the process of building queries consisted of the six following stages.

First, each of the 14 geometric datasets was joined with the corresponding HMR+ table on the FSO number. In doing so, only those entries of the HMR+ were selected that were valid at the reference time of the respective geometric dataset. This result set comprised the HMR+ entries that temporally overlapped with the geometries. In the normal case, the period of validity of such an HMR+ entry begins before and ends after the default validity interval of a PLG. Such a default validity interval lasts from one update to the next. In several cases, however, the respective HMR+ entry undergoes spatial change before the next update. As already described in Chapter 6.3.2, the start date of the validity interval of the subsequent PLG is then brought forward to avoid PLGs without a geometry, if possible. The following five stages were necessary to determine whether the start date and/or the finish date of a PLG had to be advanced. In a first stage, the result sets were retrieved. In a second stage, those HMR+ entries thereof were selected that underwent spatial change before the present but after the previous update. The validity intervals of these entries were then extended to the date when the respective spatial change occurred. Third, those HMR+ entries were selected from the result set that underwent non-spatial change within the same interval. If the preceding entry underwent spatial change between the previous update and the non-spatial change, the validity interval was also extended to the date of the spatial change. If not, the default date was taken as the start date of the validity interval.

Performing the second and the third stage required joining the result set of the first stage with the respective HMR+ table on the *change number registration* and the *change number annulment*, respectively. Fourth, those HMR+ entries were selected from the result set that underwent spatial change before the next update. The finish dates of the PLGs concerned were consequently set to the dates of the spatial changes. Fifth, those HMR+ entries were selected from the result set that underwent non-spatial change before the next update. In cases where the superseding entry underwent spatial change, the validity interval of the respective PLG was also ended on that date. Otherwise, the default date was taken as the finish date of the validity interval. Just as the second and the third stage, the fourth and the fifth stage required joining the result set with the respective HMR+ table on the change numbers. After using this procedure for all 14 geometric datasets, the datasets were searched for PLGs without a geometry. In doing so, those entries of the entire HMR+ table were selected that underwent spatial change both at the beginning and the end, and within a period between two updates. This also concerned HMR+ entries whose validity interval was ended by a spatial change before 12/31/1989 or initiated after 01/01/2009.

Naming of WOs

SONADUS restricts WOs to having a name. Since the CR is not a historicized register and no canton has undergone a name change since 1960, WOs of cantons were given their full names according to the CR. In opposition to the cantons, name changes have occurred on both the district and the municipality level. WOs of these AUs were thus given a name that is composed of all names that they were known by in their life. As described in Chapter 6.3.2, a life consists of one or more PLAUs whose corresponding POAUs each bear a name. In most of the cases, this name remains the same from the beginning to the end of a life and thus becomes the name of the corresponding WO. In case of name changes, however, the different names during a life were concatenated in chronological order by using two underline separators (e.g., Zurzach__Bad Zurzach). If a name occurred various times in consecutive order, it was only considered once.

7.1.3 Encoding the triples into RDF/XML syntax

To store the knowledge base in a single text file, the triples in the CSV files were encoded into RDF/XML syntax and subsequently inserted into the text file with the inferred ontology, SONADUSi. In doing so, the 138 query results were first exported into an equal number of CSV files. To facilitate the conversion into RDF/XML syntax, these files were grouped according to the property and stored in separate folders. One folder, for instance, contained all files of the property `type`. A Java program then read all CSV files of a folder, encoded the triples into RDF/XML syntax and stored them in a single OWL file. Afterward, the contents of the 14 resulting OWL files were manually added to the OWL file with SONADUSi, i.e., SONADUS including the inferred axioms.

7.2 SONADUS knowledge base (SONADUS-KB)

As a knowledge base, SONADUS-KB comprises both a T-Box and an A-Box. The T-Box of SONADUS-KB corresponds to SONADUSi while the A-Box consists of 167’843 individuals retrieved from the source data. With 915’737 statements about individuals or 94 MB, the A-Box is significantly larger than the T-Box that requires only 5 MB. Therefore, the text file that contains both necessitates approximately 99 MB and consists of 3’721’610 lines.

The following triple in RDF/XML syntax represents a statement of the A-Box. This statement assigns an instance to the class *whole object municipality* (GanzesObjektPolitischeGemeinde). While rdf:about refers to the subject 42105, rdf: stands for the namespace of the RDF predicate type. Finally, rdf:resource describes the object of the triple, i.e., GanzesObjektPolitischeGemeinde.

```
<owl:Thing rdf:about="#42105">
<rdf:type rdf:resource="#GanzesObjektPolitischeGemeinde"/>
</owl:Thing>
```

8 Step 4: Evaluation

The last one of the four steps describes the method and the results of the evaluation of SONADUS-KB. This mainly includes storing SONADUS-KB in a triplestore and querying it from a SPARQL endpoint to test whether the knowledge base has the required querying capabilities.

8.1 Method

This section provides a description of the materials used and the methods applied to store and query SONADUS-KB.

8.1.1 Materials

Virtuoso

Virtuoso OpenLink Software (Erling and Mikhailov 2009) provides a triplestore for large RDF graphs, a SPARQL query interface, and an RDF mapper. When uploading a text file with an RDF graph, Virtuoso efficiently stores the triples of this graph in a relational database table. A single server is thus able to handle approximately 500 million triples (Virtuoso 2010a). The open source version of the software is downloadable on Virtuoso (2010b).

8.1.2 Loading SONADUS-KB into a triplestore

After downloading and installing Virtuoso, the text file with the entire knowledge base was loaded into the DAV RDF sink. For unknown reasons, Virtuoso did not store an RDF graph of the knowledge base even though the RDF sink contained the file. Since modifying the configurations of Virtuoso remained unsuccessful, this problem was eventually solved by splitting up SONADUS-KB into several files. Each of these OWL files had the same header and was less than 10 MB. When loading the files separately into the DAV RDF sink, Virtuoso created an RDF graph of each that could be queried.

8.1.3 Querying SONADUS-KB

The iSPARQL facility of Virtuoso was used to query SONADUS-KB. The test queries of Chapter 2.2 were expressed in SPARQL using the classes and properties of SONADUS. Since some queries exceeded the standard maximum response time of Virtuoso, certain SPARQL parameters had to be modified.

8.2 Results

This section lists all predefined test queries again and shows the corresponding SPARQL queries sent to the server. In addition, this subchapter provides a brief explanation of each SPARQL query and presents the result sets. In most cases, there is a variety of possibilities to formulate a SPARQL query in order to retrieve the desired information. The SPARQL queries used in this work thus represent just one possible solution. There might be many others that do the same job while being even more efficient.

8.2.1 Attribute queries

Case 1: What is the name of the municipality with the FSO number 4001?

The SPARQL query in Figure 8.1 first selects all POAUs with the FSO number 74001, and retrieves their names. Instead of the real FSO number 4001, 74001 had to be used. This is the corresponding number within the unique interval of FSO municipality numbers. Second, the query selects the corresponding PLAUs and their validity intervals before retrieving the life of these PLAUs in a third stage. Having the life allows the selection of the corresponding WO and its name in a final stage. This query pattern is very frequent since SONADUS lacks a direct relationship between POAUs and WOs. Table 8.1 eventually lists the stages in the life of Aarau, including the validity intervals and the names of the WO and the POAUs.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?POAU ?POAU_Name
                ?PLAU_ValidFrom as ?PLAU_VF
                ?PLAU_ValidUntil as ?PLAU_VU
                ?WO ?WO_Name
                ?Life_ValidFrom as ?Life_VF
                ?Life_ValidUntil as ?Life_VU
WHERE { ?POAU son:hatBFS-Gemeindenummer son:74001  .
        ?POAU son:hatName ?POAU_Name .
        ?POAU son:hatLeben ?PLAU .

        ?PLAU ?b ?PLAU_ValidFrom .
        FILTER (?b = son:beginntAm || ?b = son:zuerstErfasstAm)

        ?PLAU ?e ?PLAU_ValidUntil .
        FILTER (?e = son:endetAm || ?e = son:zuletztErfasstAm)

        ?PLAU son:istZeitteilVon ?Life .

        ?Life ?c ?Life_ValidFrom .
        FILTER (?c = son:beginntAm || ?c = son:zuerstErfasstAm)

        ?Life ?f ?Life_ValidUntil .
        FILTER (?f = son:endetAm || ?f = son:zuletztErfasstAm)

        ?Life son:hatTeilnehmerWaehrend ?WO .
        ?WO son:hatName ?WO_Name .

}
```

Figure 8.1: Case 1 SPARQL query

POAU	POAU_Name	PLAU_VF	PLAU_VU	WO	WO_Name	Life_VF	Life_VU
11736	Aarau	01.01.1960	31.12.2009	40002	Aarau	01.01.1960	31.07.2010
15385	Aarau	01.01.2010	31.07.2010	40002	Aarau	01.01.1960	31.07.2010

Table 8.1: Case 1 result

8.2.2 Spatial queries

Simple spatial query

Case 2: Where is the Municipality of Birmensdorf (ZH)?

The SPARQL query in Figure 8.2 first selects all WOs of the type municipality (GanzesObjektPolitischeGemeinde) before searching the names of these WOs for the substring Birmensdorf_ZH. Thereafter, the query selects the life of that particular WO and all temporal parts of this life that are PLGs. After extracting the start dates and the finish dates of these PLGs, the query retrieves the pointers to the geometries in the shapefiles. In contrast to Case 1, this query returns the dates in the XML format date which is necessary whenever a query includes temporal comparison or ordering. Hence, Table 8.2 lists all geometries of Birmensdorf (ZH) between initial entry and final entry, and the corresponding validity intervals. Since URIs must not contain brackets and blank spaces, these were replaced when encoding the triples into RDF/XML syntax. Thus, municipality names with cantonal codes have to be queried using an underscore between the actual name and the code. However, this SPARQL query cannot fully answer the question of Case 2 because the geometries in SONADUS-KB do not point to actual objects in the shapefiles yet. If that was the case, the information system could return the location of Birmensdorf (ZH) depicted on a map.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO ?WO_Name ?Geometry ?PLG_ValidFrom_Date ?PLG_ValidUntil_Date
WHERE { ?WO rdf:type son:GanzesObjektPolitischeGemeinde .
        ?WO son:hatName ?WO_Name .
             FILTER regex(?WO_Name, "Birmensdorf_ZH")
        ?WO son:hatLeben ?Life .
        ?Life son:umfasstZeitteil ?PLG .
        ?PLG rdf:type son:LebensabschnittGeometriePolitischeGemeinde .
        ?PLG ?b ?PLG_ValidFrom .
        FILTER (?b = son:beginntAm || ?b = son:zuerstErfasstAm)
        ?PLG_ValidFrom dc:date ?PLG_ValidFrom_Date .
        ?PLG ?e ?PLG_ValidUntil .
        FILTER (?e = son:endetAm || ?e = son:zuletztErfasstAm)
        ?PLG_ValidUntil dc:date ?PLG_ValidUntil_Date .
        ?PLG son:hatTeilnehmerWaehrend ?POG .
        ?POG son:raeumlichRepraesentiertDurch ?Geometry .

}

ORDER BY ?PLG_ValidFrom_Date
```

Figure 8.2: Case 2 SPARQL query

WO	WO_Name	Geometry	PLG_ValidFrom_Date	PLG_ValidUntil_Date
40288	Birmensdorf_ZH	gzg90_ch_gde.shp:754	1960-01-01	1993-12-31
40288	Birmensdorf_ZH	gzg94_ch_gde.shp:753	1994-01-01	1995-12-31
40288	Birmensdorf_ZH	gzg96_ch_gde.shp:686	1996-01-01	1997-12-31
40288	Birmensdorf_ZH	gzg98_ch_gde.shp:663	1998-01-01	1999-12-31
40288	Birmensdorf_ZH	gg25_00_a_ch_gde.shp:249500	2000-01-01	2000-12-04
40288	Birmensdorf_ZH	gg25_01_a_ch_gde.shp:249500	2000-12-05	2001-12-31
40288	Birmensdorf_ZH	gg25_02_a_ch_gde.shp:249500	2002-01-01	2002-12-31
40288	Birmensdorf_ZH	gg25_03_a_ch_gde.shp:249500	2003-01-01	2003-12-31
40288	Birmensdorf_ZH	gg25_04_a_ch_gde.shp:249500	2004-01-01	2004-12-31
40288	Birmensdorf_ZH	gg25_05_a_ch_gde.shp:249500	2005-01-01	2005-12-31
40288	Birmensdorf_ZH	gg25_06_a_ch_gde.shp:249500	2006-01-01	2006-12-31
40288	Birmensdorf_ZH	gg25_07_a_ch_gde.shp:249500	2007-01-01	2007-12-31
40288	Birmensdorf_ZH	gg25_08_a_ch_gde.shp:249500	2008-01-01	2008-12-31
40288	Birmensdorf_ZH	gg25_09_a_ch_gde.shp:249500	2009-01-01	2010-07-31

Table 8.2: Case 2 result

Spatial range query

Case 3: What are the names of the municipalities of the District of Baden?

First of all, the SPARQL query in Figure 8.3 searches the RDF graph for triples having the predicate hasName (hatName) and an object containing the substring Bezirk_Baden. Thereafter, the query selects the lives of these WOs and POs, and the temporal parts of these lives that are PLAUs of the type district (LebensabschnittVerwaltungseinheitBezirk). Next, the query selects the corresponding POAUs and the POAUs of the municipality level that these POAUs of the type district contain. After retrieving the names of the POAUs of the municipality level, the query selects the corresponding PLAUs including the respective validity intervals. Then, the query retrieves the lives of these PLAUs by using the predicate *isTemporalPartOf* (istZeitteilVon). After extracting the validity intervals of these lives, this SPARQL query eventually retrieves the names of their WOs. Hence, the result set in Table 8.3 contains the names of all current and historical municipalities in the District of Baden. The table additionally lists the WOs and the POAUs as well as the validity intervals of the corresponding lives and PLAUs. Since Baden annexed Dättwil in 1962, there are two POAUs named Baden.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO_Municipality_Name as ?WO_Name
                ?Life_Municipality_ValidFrom_Date as ?Life_ValidFrom
                ?Life_Municipality_ValidUntil_Date as ?Life_ValidUntil
                ?POAU_Municipality_Name as ?POAU_Name
                ?PLAU_Municipality_ValidFrom_Date as ?PLAU_ValidFrom
                ?PLAU_Municipality_ValidUntil_Date as ?PLAU_ValidUntil
WHERE{  ?WO son:hatName ?WO_District_Name .
             FILTER regex(?WO_District_Name, "Bezirk_Baden")
        ?WO son:hatLeben ?Life .
        ?Life son:umfasstZeitteil ?PLAU .
        ?PLAU rdf:type son:LebensabschnittVerwaltungseinheitBezirk .
        ?PLAU son:hatTeilnehmerWaehrend ?POAU .
        ?POAU_Municipality son:istRaumteilVon ?POAU .
        ?POAU_Municipality son:hatName ?POAU_Municipality_Name .
        ?POAU_Municipality son:hatLeben ?PLAU_Municipality .
        ?PLAU_Municipality ?b ?PLAU_Municipality_ValidFrom .
        FILTER (?b = son:beginntAm || ?b = son:zuerstErfasstAm)
        ?PLAU_Municipality_ValidFrom dc:date ?PLAU_Municipality_ValidFrom_Date .
        ?PLAU_Municipality ?e ?PLAU_Municipality_ValidUntil .
        FILTER (?e = son:endetAm || ?e = son:zuletztErfasstAm)
        ?PLAU_Municipality_ValidUntil dc:date ?PLAU_Municipality_ValidUntil_Date .
        ?PLAU_Municipality son:istZeitteilVon ?Life_Municipality .
        ?Life_Municipality ?c ?Life_Municipality_ValidFrom .
        FILTER (?c = son:beginntAm || ?c = son:zuerstErfasstAm)
        ?Life_Municipality_ValidFrom dc:date ?Life_Municipality_ValidFrom_Date .
        ?Life_Municipality ?f ?Life_Municipality_ValidUntil .
        FILTER (?f = son:endetAm || ?f = son:zuletztErfasstAm)
        ?Life_Municipality_ValidUntil dc:date ?Life_Municipality_ValidUntil_Date .
        ?Life_Municipality son:hatTeilnehmerWaehrend ?WO_Municipality .
        ?WO_Municipality son:hatName ?WO_Municipality_Name .

}
ORDER BY ?Life_Municipality_ValidUntil_Date
```

Figure 8.3: Case 3 SPARQL query

WO_Name	Life_ValidFrom	Life_ValidUntil	POAU_Name	PLAU_ValidFrom	PLAU_ValidUntil
Daettwil	1960-01-01	1961-12-31	Daettwil	1960-01-01	1961-12-31
Unterehrendingen	1960-01-01	2005-12-31	Unterehrendingen	1960-01-01	2005-12-31
Oberehrendingen	1960-01-01	2005-12-31	Oberehrendingen	1960-01-01	2005-12-31
Wettingen	1960-01-01	2010-07-31	Wettingen	1960-01-01	2010-07-31
Wuerenlos	1960-01-01	2010-07-31	Wuerenlos	1960-01-01	2010-07-31
Wuerenlingen	1960-01-01	2010-07-31	Wuerenlingen	1960-01-01	2010-07-31
Wohlenschwil	1960-01-01	2010-07-31	Wohlenschwil	1960-01-01	2010-07-31
Spreitenbach	1960-01-01	2010-07-31	Spreitenbach	1960-01-01	2010-07-31
Stetten_AG	1960-01-01	2010-07-31	Stetten_AG	1960-01-01	2010-07-31
Untersiggenthal	1960-01-01	2010-07-31	Untersiggenthal	1960-01-01	2010-07-31
Turgi	1960-01-01	2010-07-31	Turgi	1960-01-01	2010-07-31
Baden	1960-01-01	2010-07-31	Baden	1960-01-01	1961-12-31
Bergdietikon	1960-01-01	2010-07-31	Bergdietikon	1960-01-01	2010-07-31
Bellikon	1960-01-01	2010-07-31	Bellikon	1960-01-01	2010-07-31
Birmenstorf_AG	1960-01-01	2010-07-31	Birmenstorf_AG	1960-01-01	2010-07-31
Freienwil	1960-01-01	2010-07-31	Freienwil	1960-01-01	2010-07-31
Fislisbach	1960-01-01	2010-07-31	Fislisbach	1960-01-01	2010-07-31
Gebenstorf	1960-01-01	2010-07-31	Gebenstorf	1960-01-01	2010-07-31
Ennetbaden	1960-01-01	2010-07-31	Ennetbaden	1960-01-01	2010-07-31
Mellingen	1960-01-01	2010-07-31	Mellingen	1960-01-01	2010-07-31
Oberrohrdorf	1960-01-01	2010-07-31	Oberrohrdorf	1960-01-01	2010-07-31
Obersiggenthal	1960-01-01	2010-07-31	Obersiggenthal	1960-01-01	2010-07-31
Neuenhof	1960-01-01	2010-07-31	Neuenhof	1960-01-01	2010-07-31
Niederrohrdorf	1960-01-01	2010-07-31	Niederrohrdorf	1960-01-01	2010-07-31
Kuenten	1960-01-01	2010-07-31	Kuenten	1960-01-01	2010-07-31
Killwangen	1960-01-01	2010-07-31	Killwangen	1960-01-01	2010-07-31
Maegenwil	1960-01-01	2010-07-31	Maegenwil	1960-01-01	2010-07-31
Remetschwil	1960-01-01	2010-07-31	Remetschwil	1960-01-01	2010-07-31
Baden	1960-01-01	2010-07-31	Baden	1962-01-01	2010-07-31
Ehrendingen	2006-01-01	2010-07-31	Ehrendingen	2006-01-01	2010-07-31

Table 8.3: Case 3 result

Spatial relationship query

Case 4: Where are the biotopes that are inside the boundaries of the City of Rapperswil-Jona?

The SPARQL query in Figure 8.4 first selects all WOs of the municipality level, and then searches these WOs for objects containing the substring Rapperswil-Jona. Thereafter, the query selects the life of this WO and the PLGs that this life comprises. Before retrieving the validity intervals of these PLGs, the query extracts the pointers to the geometries in the shapefiles. Consequently, Table 8.4 lists all available geometries of Rapperswil-Jona as well as the respective validity intervals. It is nevertheless evident that this result set cannot answer the question of Case 4. This stems from the fact that SONADUS-KB neither stores information about biotopes nor is able to perform spatial overlays.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO ?WO_Name ?Geometry ?ValidFrom_Date ?ValidUntil_Date
WHERE{  ?WO rdf:type ?WO_MunicipalityLevelEntity .
        ?WO_MunicipalityLevelEntity rdfs:subClassOf son:GanzesObjektStufeGemeinde .
        ?WO son:hatName ?WO_Name .
            FILTER regex(?WO_Name, "Rapperswil-Jona")
        ?WO son:hatLeben ?Life .
        ?Life son:umfasstZeitteil ?PLG .
        ?PLG rdf:type ?PLG_MunicipalityLevelEntity .
        ?PLG_MunicipalityLevelEntity rdfs:subClassOf
             son:LebensabschnittGeometrieStufeGemeinde .
        ?PLG son:hatTeilnehmerWaehrend ?POG .
        ?POG son:raeumlichRepraesentiertDurch ?Geometry .
        ?PLG ?b ?ValidFrom .
        FILTER (?b = son:beginntAm || ?b = son:zuerstErfasstAm)
        ?ValidFrom dc:date ?ValidFrom_Date .
        ?PLG ?e ?ValidUntil .
        FILTER (?e = son:endetAm || ?e = son:zuletztErfasstAm)
        ?ValidUntil dc:date ?ValidUntil_Date .

}
ORDER BY ?ValidUntil_Date
```

Figure 8.4: Case 4 SPARQL query

WO	WO_Name	Geometry	ValidFrom_Date	ValidUntil_Date
42289	Rapperswil-Jona	gg25_07_a_ch_gde.shp:249793	2007-01-01	2007-12-31
42289	Rapperswil-Jona	gg25_08_a_ch_gde.shp:249793	2008-01-01	2008-12-31
42289	Rapperswil-Jona	gg25_09_a_ch_gde.shp:249793	2009-01-01	2010-07-31

Table 8.4: Case 4 result

8.2.3 Temporal queries

Simple temporal queries

Case 5:	Which municipalities were abolished on December 31, 2009?

The SPARQL query in Figure 8.5 selects all WOs of the municipality level before retrieving their names. This query then selects the lives of these WOs that ended on (endetAm) 12/31/2009. Accordingly, the result set in Table 8.5 contains the WOs that meet the query specifications.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO ?WO_Name son:31.12.2009 as ?AbolishedOn
WHERE{  ?WO rdf:type ?WO_MunicipalityLevelEntity .
        ?WO_MunicipalityLevelEntity rdfs:subClassOf son:GanzesObjektStufeGemeinde .
        ?WO son:hatName ?WO_Name .
        ?WO son:hatLeben ?Life .
        ?Life son:endetAm son:31.12.2009 .

}
ORDER BY ?WO_Name
```

Figure 8.5: Case 5 SPARQL query

WO	WO_Name	AbolishedOn
40023	Aeschlen	31.12.2009
40051	Alt St. Johann	31.12.2009
40164	Ballmoos	31.12.2009
40167	Balm bei Messen	31.12.2009
40331	Bondo	31.12.2009
40410	Brunnenthal	31.12.2009
40511	Castasegna	31.12.2009
40916	Etzgen	31.12.2009
41284	Hilfikon	31.12.2009
41324	Hottwil	31.12.2009
41373	Ittenthal	31.12.2009
41642	Littau	31.12.2009
41724	Malix	31.12.2009
41811	Mettau	31.12.2009
42074	Oberhofen AG	31.12.2009
42087	Oberramsern	31.12.2009
42175	Parpan	31.12.2009
42238	Portein	31.12.2009
42253	Praez	31.12.2009
42376	Rohr AG	31.12.2009
42528	Sarn	31.12.2009
42678	Soglio	31.12.2009
42730	Stampa	31.12.2009
42769	Sulz AG	31.12.2009
42792	Tartar	31.12.2009
42883	Tuescherz-Alfermee	31.12.2009
42885	Twann	31.12.2009
42903	Umiken	31.12.2009
42924	Untersteckholz	31.12.2009
42996	Vicosoprano	31.12.2009
43155	Wil AG	31.12.2009
43162	Wildhaus	31.12.2009

Table 8.5: Case 5 result

Case 6: Which municipality was responsible for the conservation of the bog of Johannisberg on January 3, 2011?

SONADUS-KB does not contain information about biotopes and can thus not answer this question. Supporting queries about biotopes would either require an extension of SONADUS with biotopes or a connection with a biotope ontology.

Case 7: When did the City of Rapperswil and the Municipality of Jona merge?

The SPARQL query in Figure 8.6 first selects all changes of the type merger (Gemeindefusion). The query subsequently retrieves the POAUs that were *participants beforehand* (hatTeilnehmerVorher) in one of these mergers. This means that the PLAUs of these POAUs did not endure the merger. Thereafter, the query retrieves the POAUs having a name that contains the substring Rapperswil or Jona. Last, the query extracts the start date of the new life that follows the merger. This start date corresponds to the date of commencement of the merger. Hence, Table 8.6 shows the change number of the merger and the date of commencement.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?Change ?DateOfMerger
WHERE{  ?Change rdf:type son:Gemeindefusion .
        ?Change son:hatTeilnehmerVorher ?POAU .
        ?POAU son:hatName ?POAU_Name .
             FILTER (?POAU_Name = son:Rapperswil || ?POAU_Name = son:Jona)
        ?Change son:erschafft ?Life .
        ?Life son:beginntAm ?DateOfMerger .

}
```

Figure 8.6: Case 7 SPARQL query

Change	DateOfMerger
52724	01.01.2007

Table 8.6: Case 7 result

Temporal range query

Case 8: Has the Municipality of Birmensdorf (ZH) undergone change since 1960?

First of all, the SPARQL query in Figure 8.7 selects all WOs of the type municipality having a name that contains the substring Birmensdorf_ZH. Second, the query retrieves the life of the concerning WO and the changes that this life endured. Third, in order to extract the dates of commencement of these changes, this query selects the start dates of the PLs that follow them. Table 8.7 shows a list of the changes, their types, and the dates of commencement. Note that instead of using the SELECT query form, one could alternatively use the ASK form. In SPARQL, an ASK query returns a Boolean value depending on whether there is a result set or not (see Chapter 3.4.12).

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO ?WO_Name ?TemporalParts as ?Change
                ?Change as ?ChangeType ?Date
WHERE{   ?WO rdf:type son:GanzesObjektPolitischeGemeinde .
         ?WO son:hatName ?WO_Name .
               FILTER regex(?WO_Name, "Birmensdorf_ZH")
         ?WO son:hatLeben ?Life .
         ?TemporalParts son:istZeitteilVon ?Life .
         ?TemporalParts rdf:type ?Change .
         ?Change rdfs:subClassOf son:Mutationsprozess .
         ?TemporalParts ?e ?PL .
               FILTER (?e = son:wirdAbgeloestDurch ||
                       ?e = son:erschafft ||
                       ?e = son:entsprichtZeitlichMit)
         ?PL ?b ?DateOfChange .
               FILTER (?b = son:beginntAm || ?b = son:findetStattAm)
         ?DateOfChange dc:date ?Date .

}
ORDER BY ?Date
```

Figure 8.7: Case 8 SPARQL query

WO	WO_Name	Change	ChangeType	Date
40288	Birmensdorf_ZH	51500	AenderungBezirkszugehoerigkeit	1990-01-01
40288	Birmensdorf_ZH	61994	MutationsprozessUpdateGeometrie	1994-01-01
40288	Birmensdorf_ZH	61996	MutationsprozessUpdateGeometrie	1996-01-01
40288	Birmensdorf_ZH	61998	MutationsprozessUpdateGeometrie	1998-01-01
40288	Birmensdorf_ZH	62000	MutationsprozessUpdateGeometrie	2000-01-01
40288	Birmensdorf_ZH	62001	MutationsprozessUpdateGeometrie	2000-12-05
40288	Birmensdorf_ZH	62002	MutationsprozessUpdateGeometrie	2002-01-01
40288	Birmensdorf_ZH	62003	MutationsprozessUpdateGeometrie	2003-01-01
40288	Birmensdorf_ZH	62004	MutationsprozessUpdateGeometrie	2004-01-01
40288	Birmensdorf_ZH	62005	MutationsprozessUpdateGeometrie	2005-01-01
40288	Birmensdorf_ZH	62006	MutationsprozessUpdateGeometrie	2006-01-01
40288	Birmensdorf_ZH	62007	MutationsprozessUpdateGeometrie	2007-01-01
40288	Birmensdorf_ZH	62008	MutationsprozessUpdateGeometrie	2008-01-01
40288	Birmensdorf_ZH	62009	MutationsprozessUpdateGeometrie	2009-01-01

Table 8.7: Case 8 result

Temporal relationship query

Case 9:	How many changes did the Municipality of Berg (TG) undergo after the annexation of Andhausen?

The SPARQL query in Figure 8.8 has a subquery in the WHERE clause that returns the date when Berg (TG) annexed Andhausen. The date of change is then used to filter later dates out of the entire set of dates. The query subsequently selects all PLs having a start date that corresponds to one of the dates of this subset. Thereafter, the POs of these PLs are retrieved to find out the types of the changes that occurred on these dates. Then, the query selects the life that comprises these changes (istZeitteilVon). This is done to be capable of retrieving the particular WO named Berg_TG. Hence, Table 8.8 shows the 13 changes that Berg (TG) underwent after the annexation of Andhausen. Note that the same SPARQL query does not produce any result set in a reasonable computation time if one replaces the statement <?WO, son:hatName, son:Berg_TG> with <?WO, son:hatName, ?WO_Name> and FILTER regex(?WO_Name, "Berg_TG"). This fact demonstrates that searching object literals for a certain substring requires significantly more computing power than simply retrieving triples where the object is given.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO son:Berg_TG as ?WO_Name
                ?Change ?ChangeType
                ?DatesOfChanges as ?Date

WHERE{   ?Dates dc:date ?DatesOfChanges .
              FILTER (?DatesOfChanges > xsd:date(?DateOfChange))
        ?PL son:beginntAm ?Dates .
        ?PO son:hatLeben ?PL .
        ?Change son:hatTeilnehmerNachher ?PO .
        ?Change rdf:type ?ChangeType .
        ?ChangeType rdfs:subClassOf son:Mutationsprozess .
        ?Change son:istZeitteilVon ?Life .
        ?WO son:hatLeben ?Life .
        ?WO son:hatName son:Berg_TG .

        {
        SELECT distinct ?DateOfChange

        WHERE{   ?Change rdf:type son:Eingemeindung .
                ?Change son:hatTeilnehmerVorher ?POAU .
                ?POAU rdf:type son:TeilobjektVerwaltungseinheitPolitischeGemeinde .
                ?POAU son:hatName son:Andhausen .
                ?Change son:wirdAbgeloestDurch ?PLAU .
                ?PLAU son:beginntAm ?Date .
                ?Date dc:date ?DateOfChange .

        }
        }
}
```

Figure 8.8: Case 9 SPARQL query

WO	WO_Name	Change	ChangeType	Date
40227	Berg_TG	51869	Eingemeindung	1994-12-31
40227	Berg_TG	51870	Gebietsabtausch	1995-01-01
40227	Berg_TG	51912	Gebietsabtausch	1998-01-01
40227	Berg_TG	62000	MutationsprozessUpdateGeometrie	2000-01-01
40227	Berg_TG	62001	MutationsprozessUpdateGeometrie	2000-12-05
40227	Berg_TG	62002	MutationsprozessUpdateGeometrie	2002-01-01
40227	Berg_TG	62003	MutationsprozessUpdateGeometrie	2003-01-01
40227	Berg_TG	62004	MutationsprozessUpdateGeometrie	2004-01-01
40227	Berg_TG	62005	MutationsprozessUpdateGeometrie	2005-01-01
40227	Berg_TG	62006	MutationsprozessUpdateGeometrie	2006-01-01
40227	Berg_TG	62007	MutationsprozessUpdateGeometrie	2007-01-01
40227	Berg_TG	62008	MutationsprozessUpdateGeometrie	2008-01-01
40227	Berg_TG	62009	MutationsprozessUpdateGeometrie	2009-01-01

Table 8.8: Case 9 result

8.2.4 Spatiotemporal queries

Simple spatiotemporal query

Case 10: What were the borders of Berg (TG) on January 1, 1996?

First of all, the SPARQL query in Figure 8.9 selects all WOs of the municipality level. Next, the query retrieves the WO whose name contains the substring `Berg_TG`. In order to extract the pointers to the geometries, this query then selects the life of this WO, the PLGs of that life, and the corresponding POGs. Finally, the query filters the PLGs to extract the particular geometry that was valid on 01/01/1996 (see Table 8.9). As in Case 2, this SPARQL query does not fully answer the question posed. This is because SONADUS-KB only returns a pointer to a geometry and does not depict the borders themselves.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?WO ?WO_Name ?Geometry ?ValidFrom_Date ?ValidUntil_Date
WHERE{   ?WO rdf:type ?WO_MunicipalityLevelEntity .
         ?WO_MunicipalityLevelEntity rdfs:subClassOf son:GanzesObjektStufeGemeinde .
         ?WO son:hatName ?WO_Name .
             FILTER regex(?WO_Name, "Berg_TG")
         ?WO son:hatLeben ?Life .
         ?Life son:umfasstZeitteil ?PLG .
         ?PLG rdf:type ?PLG_MunicipalityLevelEntity .
         ?PLG_MunicipalityLevelEntity rdfs:subClassOf
              son:LebensabschnittGeometrieStufeGemeinde .
         ?PLG son:hatTeilnehmerWaehrend ?POG .
         ?POG son:raeumlichRepraesentiertDurch ?Geometry .
         ?PLG ?b ?ValidFrom .
         FILTER (?b = son:beginntAm || ?b = son:zuerstErfasstAm)
         ?ValidFrom dc:date ?ValidFrom_Date .
         FILTER (?ValidFrom_Date < xsd:date("1996-01-02"))
         ?PLG ?e ?ValidUntil .
         FILTER (?e = son:endetAm || ?e = son:zuletztErfasstAm)
         ?ValidUntil dc:date ?ValidUntil_Date .
         FILTER (?ValidUntil_Date > xsd:date("1995-12-31"))

}
```

Figure 8.9: Case 10 SPARQL query

WO	WO_Name	Geometry	ValidFrom_Date	ValidUntil_Date
40227	Berg_TG	gzg96_ch_gde.shp:97	1995-01-01	1997-12-31

Table 8.9: Case 10 result

Spatiotemporal range query

Case 11: Which municipalities existing on 01/01/1960 have been abolished by 12/31/2009 in the District of Baden?

First of all, the SPARQL query in Figure 8.10 retrieves all POAUs of the municipality level, including the names of these POAUs. From this set, the query subsequently selects those POAUs that are part of the District of Baden (istRaumteilVon). After omitting POAUs that were not *participants afterward* (hatTeilnehmerNachher) in an initial entry, the query retrieves the corresponding PLAUs from the POAUs left. Thereafter, this SPARQL query selects the lives that comprise these PLAUs. Last, the query omits lives that still existed after 12/31/2009. Table 8.10 consequently shows the three municipalities that were abolished by this date. Consider that to simplify matters, this query uses the number 51000 to select POAUs that were *participants afterward* in an initial entry. This only works because in contrast to other change types, all initial entries share the same number. Alternatively, one could use a variable to select all changes with a *participant afterward* in a first step. From this set, one would then retrieve the changes of the type initial entry in a second step.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?POAU ?POAU_Name ?DateOfAbolition
WHERE{  ?POAU rdf:type ?POAU_M_Level .
        ?POAU_M_Level rdfs:subClassOf son:TeilobjektVerwaltungseinheitStufeGemeinde .
        ?POAU son:hatName ?POAU_Name .
        ?POAU son:istRaumteilVon ?POAU_D .
        ?POAU_D son:hatName son:Bezirk_Baden .
        son:51000 son:hatTeilnehmerNachher ?POAU .
        ?POAU son:hatLeben ?PLAU .
        ?PLAU son:istZeitteilVon ?Life .
        ?Life son:endetAm ?Date .
        ?Date dc:date ?DateOfAbolition .
              FILTER(?DateOfAbolition <= xsd:date("2009-12-31"))

}
ORDER BY ?DateOfAbolition
```

Figure 8.10: Case 11 SPARQL query

POAU	POAU_Name	DateOfAbolition
11227	Daettwil	1961-12-31
10862	Unterehrendingen	2005-12-31
12460	Oberehrendingen	2005-12-31

Table 8.10: Case 11 result

Spatiotemporal behavior query

Case 12:	Which mergers in the Canton of Ticino led to the abolition of more than two municipalities and occurred in the 2000s?

The SPARQL query in Figure 8.11 has two subqueries. The inner subquery selects all mergers and the participating municipalities, and passes this information on to the outer subquery. The latter counts the number of municipalities that were abolished by each merger. After omitting mergers that led to no more than two abolitions, the actual query retrieves the names of the municipalities left. Moreover, the query restricts these municipalities to being part of a district that is part of a canton whose name contains the substring Ticino. Finally, the query discards municipalities whose validity intervals did not start between 01/01/2000 and 12/31/2009. Table 8.11 therefore lists the change numbers of the mergers concerned. In addition, this table shows the municipalities that were abolished and those that were created, as well as the respective dates of commencement.

```
PREFIX son: <http://www.wsl.ch/sonadus#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?Change ?POAU_V_Name as ?Municipalities_Before
                ?POAU_Name as ?Municipalities_After ?DateOfMerger

WHERE{  FILTER(?Count > 2)
        ?Change son:hatTeilnehmerVorher ?POAU_V .
        ?POAU_V son:hatName ?POAU_V_Name .
        ?Change son:hatTeilnehmerNachher ?POAU .
        ?POAU son:hatName ?POAU_Name .
        ?POAU son:istRaumteilVon ?POAU_D .
        ?POAU_D son:istRaumteilVon ?POAU_C .
        ?POAU_C son:hatName ?POAU_C_Name .
                FILTER regex(?POAU_C_Name, "Ticino")
        ?POAU son:hatLeben ?PLAU .
        ?PLAU son:beginntAm ?Date .
        ?Date dc:date ?DateOfMerger .
                FILTER(?DateOfMerger >= xsd:date("2000-01-01") &&
                       ?DateOfMerger <= xsd:date("2009-12-31"))

        {
        SELECT COUNT(*) as ?Count ?Change

        WHERE{
                     {
                     SELECT distinct ?Change
                                     ?POAU_V_Name as ?Municipalities_Before
                                     ?POAU_Name as ?Municipalities_After
                                     ?DateOfMerger

                     WHERE{      ?Change rdf:type son:Gemeindefusion .
                                 ?Change son:hatTeilnehmerVorher ?POAU_V .
                                 ?POAU_V son:hatName ?POAU_V_Name .
                                 ?Change son:hatTeilnehmerNachher ?POAU .

                     }
                     }
        }
        GROUP BY ?Change
        }
}
```

Figure 8.11: Case 12 SPARQL query

Change	Municipalities_Before	Municipalities_After	DateOfMerger
52156	Auressio	Isorno	2001-04-13
52156	Berzona	Isorno	2001-04-13
52156	Loco	Isorno	2001-04-13
52157	Vaglio	Capriasca	2001-10-18
52157	Cagiallo	Capriasca	2001-10-18
52157	Lopagno	Capriasca	2001-10-18
52157	Roveredo-Capriasca	Capriasca	2001-10-18
52157	Sala_Capriasca	Capriasca	2001-10-18
52157	Tesserete	Capriasca	2001-10-18
52292	Castro	Acquarossa	2004-04-04
52292	Corzoneso	Acquarossa	2004-04-04
52292	Dongio	Acquarossa	2004-04-04
52292	Largario	Acquarossa	2004-04-04
52292	Marolta	Acquarossa	2004-04-04
52292	Leontica	Acquarossa	2004-04-04
52292	Lottigna	Acquarossa	2004-04-04
52292	Ponto_Valentino	Acquarossa	2004-04-04
52292	Prugiasco	Acquarossa	2004-04-04
52297	Brontallo	Lavizzara	2004-04-04
52297	Broglio	Lavizzara	2004-04-04
52297	Fusio	Lavizzara	2004-04-04
52297	Menzonio	Lavizzara	2004-04-04
52297	Prato-Sornico	Lavizzara	2004-04-04
52297	Peccia	Lavizzara	2004-04-04
52298	Agra	Collina_d_Oro	2004-04-04
52298	Gentilino	Collina_d_Oro	2004-04-04
52298	Montagnola	Collina_d_Oro	2004-04-04
52319	Vezio	Alto_Malcantone	2005-03-13
52319	Arosio	Alto_Malcantone	2005-03-13
52319	Breno	Alto_Malcantone	2005-03-13
52319	Fescoggia	Alto_Malcantone	2005-03-13
52319	Mugena	Alto_Malcantone	2005-03-13
52717	Torre	Blenio	2006-10-22
52717	Campo_Blenio	Blenio	2006-10-22
52717	Aquila	Blenio	2006-10-22
52717	Ghirone	Blenio	2006-10-22
52717	Olivone	Blenio	2006-10-22
52771	Borgnone	Centovalli	2009-10-25
52771	Palagnedra	Centovalli	2009-10-25
52771	Intragna	Centovalli	2009-10-25
52772	Bruzella	Breggia	2009-10-25
52772	Cabbio	Breggia	2009-10-25
52772	Morbio_Superiore	Breggia	2009-10-25
52772	Muggio	Breggia	2009-10-25
52772	Sagno	Breggia	2009-10-25
52772	Caneggio	Breggia	2009-10-25

Table 8.11: Case 12 result

9 Discussion

After the practical part of the thesis, the discussion returns to the research questions (RQs) posed at the beginning. Besides discussing the literature, this chapter points out shortcomings of the source data and highlights strengths and weaknesses of SONADUS. The chapter proposes measures to overcome these shortcomings and provides a critical review of using the upper ontology BFO as a reference framework.

9.1 Conclusions from the literature review

Since the 1980s, research in information science has made substantial progress toward the integration of time into a GIS (Yuan 2008). While Abraham and Roddick (1999) and Pelekis et al. (2005) provide comprehensive reviews of work done on spatiotemporal databases, this thesis has focused on the predominant approaches and has also included examples of applications. Aside from database models, Chapter 3 has considered ontologies and knowledge bases since they have received increasing attention in GIScience over the last years (Bittner et al. 2004, Agarwal 2005). This fact has also influenced the discussion of the following research question.

RQ1: Which approaches does the literature provide for an information system that supports queries about discrete change of spatial objects with crisp boundaries? Which of these approaches facilitate the integration of differing datasets?

General models

Most approaches adhere to a general model to organize knowledge (Abraham and Roddick 1999). On the one hand, there is the relational and the object model that the majority of the database models extend. On the other hand, there are ontologies that store knowledge in triple form < subject, predicate, object >. Databases and ontologies are not independent from each other, neither on the conceptual level nor with respect to their application. Ontologies have wielded significant influence on database design in recent years (Hornsby and Joshi 2010). In turn, databases in the form of triplestores may serve to efficiently store large ontologies and knowledge bases. Not only ontologies, but also the relational and the object model devise a particular structure. The relational model consists of two-dimensional tables that can be joined on corresponding attributes. The object model regards objects as the world’s basic entities. Objects have their own attributes and behaviors that can be inherited from superclasses and are protected from immediate access by encapsulation (Atkinson et al. 1989). Ontologies, finally, comprise a class tree as well as properties that establish relationships between the classes. The class tree, or taxonomy, has two implications that represent the crucial difference between the database models and ontologies. First, this structure allows to perform reasoning to infer implicit knowledge such as the subClassOf-relationships to indirect superclasses. Second, research has shown that such taxonomies are only suited to domains and applications with clear concepts. This applies, for instance, to the biomedical domain where ontologies are considered a great success (Smith et al. 2007). In geomorphology, on the contrary, many concepts are ill-defined (Straumann 2010), which is why ontologies have not become widely accepted in that domain.

Ontological approaches

Aside from certain exceptions, AUs represent clear concepts that undergo discrete change and have crisp boundaries. Building an ontology of AUs does not entail the same challenges as when creating an ontology of geomorphologic concepts. Hence, research has produced a few approaches concerning AUs on the domain and application level. This includes an ontology of the administrative subdivisions of EU countries (López-Pellicer et al. 2008) and SAPO, an ontology time series of the AUs of Finland (Kauppinen et al. 2008). The ontology of López-Pellicer et al. (2008) improves the comparability of AUs between EU countries, but its static nature does not allow queries on changes. In contrast, SAPO is able to deal with such queries. This stems from the fact that SAPO gives AUs an identity, stores the geometries of the temporal parts, and explicitly stores the changes that AUs undergo. Therefore, the domain ontology SAPO currently represents the most complete solution for supporting queries on change of AUs. On a higher level, however, upper ontologies such as SUMO (Niles and Pease 2001), DOLCE (Gangemi et al. 2002), or BFO (Grenon and Smith 2004) provide a reference framework and a firm philosophical basis to model change of any kind. BFO, for instance, combines SNAP and SPAN, two ontologies that each account for an opposing view of the world. By using both ontologies, the relationships and interactions between entities of any domain or application can be modeled. Moreover, the universality of upper ontologies facilitates the integration of different datasets, an issue that SAPO does not address.

Database approaches

Unlike upper ontologies, none of the three predominant types of database approaches emphasize on the integration of different datasets, even though all of them contain concepts of BFO. Traditional approaches that build on the relational model store a sequence of static relations. Aside from lacking features such as inheritance, each of these relations, or snapshots, conceptually corresponds to a SNAP ontology at a given time. Traditional approaches capture discrete change, provided that there are snapshots of each state (Abraham and Roddick 1999). Nevertheless, traditional approaches do not store changes explicitly. The detection of events thus requires the comparison of two or more snapshots which becomes unmanageable when there are more than fifty layers (Yuan 1999). Additionally, changes that occur between two updates remain undetected. Since successful querying requires the explicit storage of the desired information (Jiang and Worboys 2009), traditional approaches fail to adequately support queries on change. Less pronounced, the same applies to object-oriented approaches. According to these, the world is comprised of objects that have an identity and can be described by attributes and behaviors (Worboys 2001). Every object belongs to a class within a class hierarchy in which a subclass inherits all properties from its superclass. In addition, subclasses may also have their own attributes and operations (Lohfink et al. 2007). This conceptualization complies with the one of a SNAP ontology where subclasses inherit properties from their superclasses too. The object-centeredness usually leads to a demotion of occurrents in object-oriented approaches even though exceptions such as FEM (Lohfink et al. 2007) exist. By considering occurrents as an object type, FEM is able to support the four types of temporal and spatiotemporal queries of Langran (1992). While object-oriented approaches largely focus on objects, event-oriented approaches take the view that the world consists of dynamic entities such as events and processes. The explicit storage of events corresponds to the conceptualization of a SPAN ontology and facilitates the formulation of queries on change. Hence, Worboys (2005) regards event-oriented approaches as being superior to object-oriented approaches with respect to modeling geographic processes and phenomena. Besides these advantages, event-oriented approaches have their drawbacks. By focusing on events and processes, event-oriented approaches make it more difficult to formulate queries on objects. In general, none of the three predominant approaches place objects and events on the same level as BFO does. While object-oriented approaches are particularly suitable for queries on objects, event-oriented approaches are best for queries on events. Traditional approaches, at last, neither provide for sufficient support of queries on objects nor on events. In practice, however, the widespread use of traditional approaches demonstrates that also rather simplistic approaches offer sufficient query support for many real-world applications (Stock 2006).

9.2 Findings from the data analysis

The data originate from two different sources, that is, the Federal Statistical Office (FSO) and the Federal Office of Topography (FOT). While the FSO issues the Historicized Municipality Register (HMR+) and the geometric datasets for the 1990s, the FOT has released yearly updated boundaries since 2000. The separate data collection has several implications. Most importantly, the HMR+ and the geometric datasets use different approaches to modeling data which makes the integration of these two datasets a considerable challenge. The approach of the HMR+ that borrows some features of the object-oriented paradigm is clearly superior to the traditional approach of the geometric datasets. As opposed to the geometric datasets that fail to capture states between two updates, the HMR+ covers the entire change history of AUs between 1960 and present.

Shortcomings of the HMR+

Nonetheless, the HMR+ suffers from a serious shortcoming that has no bearing on the database model used. This shortcoming stems from the fact that the HMR only considers land exchanges that concern permanently inhabited areas and that were published as part of change reports in the official municipality register (FSO 2007b). Hence, the geometric datasets record border changes that the HMR ignores. The geometric datasets, however, only specify the last time that a change occurred, but not the respective change type. It is therefore impossible to automatically retrieve the municipalities whose border changes are recorded by the geometric datasets, but not by the HMR. Since WSL works with biotope data that also concern uninhabited areas, this thesis cannot neglect such border changes. Consequently, it does not suffice to store only a geometry of one dataset if there are several ones within the validity interval of an HMR entry. This is illustrated by means of the example in Figure 5.3.

A further shortcoming of the HMR+ is its lack of differentiation between fundamental and incremental change. While fundamental change leads to the creation or abolition of AUs, incremental change only induces an alteration of the properties such as the name. The reason for this lacking differentiation is that the HMR+ does not possess an attribute that identifies an AU over time. The attribute that should actually fulfill this function is the FSO number. However, formal renumberings and even reuse of numbers make this attribute unsuited for performing the function of an identity. As the example in Table 5.38 shows, incremental change can make AUs unrecognizable over time. An identity would increase the information content of the HMR+ and facilitate the retrieval of histories of AUs.

With an identity, the HMR+ would have a central feature of object models. If the HMR+ additionally timestamped each attribute individually instead of the whole entry (see Becker et al. 1996), the FSO could resolve the existing uncertainty about how to model two concurrent changes. This uncertainty is evident in the unequal treatment of such cases. When in the 1990s several municipalities exchanged land and annexed other municipalities on the same date, the HMR recorded both changes separately. The correct modeling of changes makes this solution preferable, even though that entails the creation of entries with no temporal extension and without any reflection of reality. In an identical case, the HMR nevertheless modeled an annexation and a land exchange as one change which makes it impossible to understand the roles of the municipalities involved. The same problem arises when a spatial and a non-spatial change coincide or two non-spatial changes concur. The former concerns municipalities that simultaneously participated in a merger or an annexation and changed the membership of the district. In these cases, the HMR only recorded the spatial change explicitly, while the new membership is declared in the newly created entry. The latter concerns Küssnacht (SZ), where a change of the municipality name caused a change of the corresponding district name on the same date. In this case, the HMR confined itself to storing the change of the municipality name explicitly. The declaration of the change of the district name happens indirectly by storing an updated *historization number district* in the new HMR entry.

Last, an obvious shortcoming of the HMR+ is the fact that the CR does not historicize the cantons. Since only one fundamental change has occurred since 1960, a historization of cantons would not incur considerable additional expenditure, but would establish the consistency of the HMR+. The creation of the Canton of Jura could then be modeled as the creation of the District of Dietikon in the HDR.

Shortcomings of the geometric datasets

Just as the HMR+, the geometric datasets have several shortcomings. On the conceptual level, the geometric datasets still use a traditional approach. Thus, they cannot capture states whose validity intervals do not overlap with the reference time of an update. There are a couple of such states in the 1990s because the FSO issued only four GZG datasets during that period. Since the FOT has released annual updates of GG25 boundaries, there has not been any state that could not be assigned a geometry. Despite a reduced probability, such states can still occur if AUs undergo spatial change twice within a calendar year.

Compared to missing geometries, errors in the data cause greater difficulties because they spread misinformation. Aside from incorrect spellings of municipality names and wrong FSO numbers, the gravest errors are obsolete entries. This means that certain geometric datasets contain entries that are not valid anymore according to the HMR. A typical example of this is Scherzingen and Landschlacht, two municipalities that merged to Münsterlingen on 01/01/1994. Instead of Münsterlingen, the GZG boundaries of 1994 still contain the two abolished municipalities. A grievous error occurred when the Canton of St. Gallen replaced its districts with constituencies *(dt. Wahlkreise)* at the beginning of 2003. While the GG25 dataset of 2003 still contains the FSO district numbers of the former districts, the versions of 2004, 2005, and 2006 consider St. Gallen as a canton without districts. This has serious repercussions on the reliability of all products that use these official datasets, be it an information system or a visualization. Hence, the FOT should either retroactively correct such faulty datasets or publish a file that lists those errors. As this was not the case, the creation of change maps proved to be an effective method to detect errors and inconsistencies both in and between the geometric datasets and the HMR+.

Change maps

Despite being a byproduct of this thesis, the change maps are of great value. In particular, the process of their creation turned out to be an excellent way to become acquainted with the HMR and the geometric data. Joining municipalities from the HMR with the corresponding geometric objects from the GZG 1990 facilitated the detection of errors in both datasets. This became apparent when a join on the FSO number produced different results than a join on the municipality name. Since neither the FSO number nor the municipality name is consistent over time, it emerged that both the FSO number and the municipality name created difficulties in some way. The high number of spelling differences between the HMR and the GZG 1990 swung the balance in favor of the FSO number. However, the change maps are not only suited to detect errors. They also provide an easy access to the subject for laymen by visualizing the quantity and type of change during the last 50 years. Furthermore, the change maps foster the awareness of change as a significant socioeconomic phenomenon of the last two decades. The visualizations allow the analysis of the distribution of change and the identification of clusters. In addition, the change maps make it possible to infer which cantons have undertaken municipal reorganizations. These advantages are in opposition to the drawback that the surface of many municipalities are too small to be seen at a scale that covers the entire area of Switzerland. A solution to this problem requires other visualization techniques that possibly include interaction functions, such as zooming or time lines. Such techniques already exist but their application would have gone beyond the scope of this thesis. In conclusion, the change maps provide a quick overview of the subject by collapsing twenty years into a single snapshot. At the same time, they reveal that static visualization techniques do not suffice to depict dynamic phenomena.

9.3 Applicability of SAPO to the situation in Switzerland

Before the creation of SONADUS-KB, SAPO has been the only ontology and knowledge base that has supported queries on the change history of AUs. Hence, it is a central issue to what extent the conceptualization of SAPO can be applied to the situation in Switzerland. The conclusions from the literature review and the data analysis now provide for the discussion of this research question.

RQ2: ***To what extent is SAPO (Kauppinen et al. 2008) applicable to the situation in Switzerland? In what way does the evolution of administrative units differ between Finland and Switzerland?***

The application ontology SAPO basically differentiates between the concepts place, temporal part, and change. A place consists of one or more temporal parts that each has a validity interval and a geometry, if available. As CultureSampo (2011) shows, the place Helsinki, for instance, is comprised of five temporal parts of which only for one a geometry is available. Consecutive temporal parts are connected by a change of one of seven Finnish change types. This classification (see Table 3.1) differs significantly from the set of Swiss change types. On the one hand, this results from the fact that the Finnish dataset covers a period from 1865 to 2007, in which the border with Russia changed several times. On the other hand, the FSO distinguishes between mergers and annexations as well as between separations and secessions that Kauppinen et al. (2008) each treat as one type of change. The FSO additionally considers formal renumberings as changes and thus takes an administration-centered view. The Finnish classification, on the contrary, includes only changes that affect the boundaries or the names of municipalities. Nonetheless, SAPO does not depend on particular change types. The differences in the classifications do not represent an obstacle to the application of SAPO in Switzerland. The main reason why SAPO is not directly applicable to the situation in Switzerland is the separate data collection of the HMR+ and the geometries. Hence, the two datasets adopt different approaches, whereby inconsistencies emerge. Since the geometric datasets are snapshots, the validity intervals of geometries do not correspond with the ones of the HMR+ entries. Moreover, the HMR+ and the geometric datasets evolve separately because the HMR does not record land exchanges in uninhabited areas. Although Kauppinen et al. (2008) may have encountered similar problems, SAPO neither models such discrepancies nor declares its level of abstraction. More precisely, it remains unknown whether there is information that SAPO does not contain. However, SAPO is relatively small since this ontology adopts a pragmatic approach that confines itself to modeling the basic elements. This is an important advantage because a simple conceptualization keeps the size and complexity of the knowledge base moderate, which, in turn, enhances query performance.

9.4 Using BFO as a reference framework

The upper ontology BFO takes a central role in the construction of SONADUS. This section confines itself to discussing the wider implications of BFO, while the subsequent one focuses on issues concerning only SONADUS.

RQ3: ***Upper ontologies such as the Basic Formal Ontology (BFO) enhance interoperability and knowledge sharing between ontologies (Noy 2004). Does BFO bring these advantages to an application ontology of administrative units? What are further advantages or disadvantages to such an application ontology that arise from using BFO?***

Aside from SONADUS, the literature currently provides only two domain or application ontologies that deal with AUs. By name, this is the ontology of López-Pellicer et al. (2008) and SAPO (Kauppinen et al. 2008). Since neither of them uses BFO, it is impossible to give a final assessment of interoperability and knowledge sharing with other ontologies. Such other ontologies either deal with biotope objects of WSL or capture the change history of another region. In the latter case, it remains questionable whether using the same upper ontology suffices for providing for interoperability and knowledge sharing. Even if the only comparable ontology, that is SAPO, used BFO, its structure would differ markedly from the one of SONADUS. First, it is unclear whether SAPO would grant the geometries the same status as SONADUS, since the issue of incongruent datasets possibly does not exist. Second, other application ontologies might choose another level of accuracy and expressivity. Third, the language of any ontology of AUs represents a central issue because many concepts are country-specify and cannot be easily translated. In addition to an ontology that contains the administrative subdivisions of different EU countries (López-Pellicer et al. 2008), there is also a need for an ontology that negotiates meaning between different change types. The enhancement of interoperability and knowledge sharing in the domain of AUs thus requires ontologies that establish a connection between the country-specific terminologies. This stands in stark contrast to the biomedical domain that has universal concepts and a standard terminology. Still, to make full use of interoperability and knowledge sharing likely necessitates additional design guidelines on the domain and application level.

Although SONADUS cannot use advantages such as interoperability and knowledge sharing, there are other benefits that arise from applying BFO. Most importantly, BFO provides concepts for multiple perspectives upon reality, and thus allows to incorporate different conceptualizations. When building SONADUS, for instance, BFO greatly facilitated integrating the different conceptualizations of the HMR+ and the geometric datasets. Moreover, SONADUS benefits from an improved comparability with other ontologies even if different terminologies make interoperability unrealistic. In such cases, SONADUS can serve as an example from which other ontology designers may borrow those elements that address their particular situation while omitting the rest. Aside from these considerable advantages, the application of BFO also has a major downside. Compared to the draft ontology, the number of classes in SONADUS is almost five times higher. Due to its relatively simple structure, this comparison applies to SAPO to a similar degree. Unlike the draft ontology and SAPO, SONADUS makes a distinction between WOs and lives, POAUs and PLAUs, as well as between POGs, PLGs, and geometries. Although the separation of continuant and occurrent entities is philosophically correct, it makes the ontology and knowledge base larger. This eventually complicates querying and leads to additional storage requirements.

9.5 Application ontology SONADUS

The structure of SONADUS adheres to the rules of BFO and thus inherits both its advantages and drawbacks. Whereas the previous section discussed the wider implications of using BFO, this one is concerned with the strengths and shortcomings of SONADUS. Drawing conclusions from the usage of BFO is of supreme importance since SONADUS is the first spatiotemporal ontology of AUs that extends an upper ontology. For that reason, the A-Box of SONADUS-KB comprises 167’843 individuals and 915’737 triples, which are high numbers compared to the number of entries in the source data. Although the results of the evaluation confirm that SONADUS-KB supports all types of test queries, SONADUS still has potential for improvement. Hence, the following section not only highlights the strengths of SONADUS, but also discusses measures to compensate for its shortcomings.

Strengths and shortcomings of SONADUS

A major strength of SONADUS is the aim to cover as much time as possible with geometries. If feasible, the start date of a validity interval is brought forward to the spatial change that ends the validity interval of the previous geometry ahead of time. The life of the municipality in Figure 6.5 includes such an example in which the validity interval of the GZG 1990 covers the time between 1964 and 1990. This example shows that the HMR did not record any spatial change of this municipality between 1964 and 1990. Hence, it can be assumed that the GZG boundaries of 1990 still reflect the state of 1964. Since the HMR does not record land exchanges in uninhabited areas, geometric change may have nevertheless occurred during that period. In such a case, SONADUS-KB would give a wrong answer to a query on the geometry between the spatial and the geometric change. However, the probability that this happens is more than outweighed by the benefit of extended validity intervals.

Another strength of SONADUS is that the ontology clearly differentiates between incremental and fundamental change. SONADUS reflects this distinction by having different object properties that relate a change with lives and PLs. As previously stated, changes cannot be classified into incremental and fundamental ones because certain types entail both. Instead of having different object properties, however, SONADUS could use the BFO class role to define what role an AU adopts at a change. At annexations, for example, there would be the role of the annexing municipality and the role of the municipalities that are annexed.

Aside from various strengths, SONADUS also suffers from a fundamental shortcoming. In contrast to SAPO, SONADUS lacks a simple structure which results from the use of BFO to a large extent. The elaborate structure of SONADUS makes it more difficult to understand the ontology and complicates querying. Moreover, due to the high number of classes and properties, the A-Box becomes large, and thus requires more storage capacity. To compensate for this shortcoming, there are several possible measures that would simplify the structure and reduce the size of SONADUS. The most effective measure is the detachment of SONADUS from BFO. This would allow WOs and lives, POAUs and PLAUs, as well as POGs and PLGs to merge to one class each. Such a knowledge base would contain the same information while having a simpler structure, requiring less storage, and providing for easier querying. At the same time, a detachment from BFO would not mean that SONADUS would lose all advantages of an upper ontology. Rather, the real benefits would still be obtained by using BFO as an example of an independent application ontology. On the contrary, it is questionable whether other advantages such as enhanced interoperability and knowledge sharing ever come into effect. A detachment from BFO would therefore not entail any serious drawbacks. SONADUS, however, would lose its status of being the only ontology of AUs that builds on both the static and dynamic part of an upper ontology.

In order to maintain this status, other measures must be imposed to compensate for the shortcomings of SONADUS. First of all, however, it is impossible to reduce the size of the A-Box by assigning individuals to both a SNAP class and their corresponding SPAN class. In BFO, continuant and occurrent are disjoint classes. A reduction of the knowledge base can therefore not be achieved without altering the ontology. Although the rules of BFO forbid extensive modifications, SONADUS contains redundant information that can be eschewed. On the one hand, this concerns inverse relationships, such as hasLife (hatLeben) and isParticipantIn (istTeilnehmerVon). Both establish a relationship between objects and their lives. Since successful querying does not depend on the direction of the relationship, one property suffices to connect two classes if there are no particular cardinality restrictions. Besides inverse relationships, SONADUS contains further superfluous properties and classes. This concerns the object properties hasTypeOfRegistration (hatArtAufnahme) and hasTypeOfAbolition (hatArtAufhebung), as well as the corresponding classes type of registration (ArtAufnahme) and type of abolition (ArtAufhebung). Each POAU on both the municipality and the district level is imposed with a restriction that enforces them to have these properties. SONADUS can dispense with these classes and properties without losing information. On the one hand, SONADUS classifies each change, which is why the types of registration and the types of annulment are not needed to identify the change type. On the other hand, different relationships exist between changes and PLs depending on whether the AU undergoes incremental or fundamental change. In conclusion, neither the deletion of inverse relationships nor the removal of the aforementioned classes and properties would exert a significant impact on the information content of SONADUS.

It has been suggested to remove redundant information beforehand. Now, this thesis proposes to introduce redundancy in order to facilitate the formulation of queries in a particular case. The formulation of queries for the evaluation has revealed that the establishment of a relationship between WOs and POs would shorten the length of many queries. In the current structure, selecting the POs of a WO requires first the selection of the life of the WO, then the PLs of the life, and finally the POs that participate in these PLs. Since this pattern is frequently found, the establishment of a direct relationship between WOs and POs would cause a reduction of complexity in many queries.

9.6 Insights from the evaluation

The evaluation has produced insights that provide for a discussion of the following research question. Furthermore, the evaluation has revealed structural shortcomings of SONADUS that slow down query processing.

RQ4: Having developed an information system that builds on an application ontology dealing with administrative units and extending BFO, what types of queries according to Yuan and McIntosh (2002) does such a system support?

The results demonstrate that SONADUS-KB supports test queries of all but one type according to Yuan and McIntosh (2002). The only exception is the spatiotemporal relationship query that, however, cannot be used to query a change history of AUs. By name, SONADUS-KB supports:

1. Attribute queries
2. Simple spatial queries
3. Spatial range queries
4. Spatial relationship queries
5. Simple temporal queries
6. Temporal range queries
7. Temporal relationship queries
8. Simple spatiotemporal queries
9. Spatiotemporal range queries
10. Spatiotemporal behavior queries

Since SONADUS-KB neither points to geometric data nor has a connection with a GIS, the actual polygons of AUs cannot be displayed. The same applies to spatial overlays required to retrieve biotopes within the boundaries of a certain municipality. Hence, SONADUS-KB does not yet have the full capabilities of a spatiotemporal information system. This knowledge base nevertheless has the potential to form the basis of such an information system since the underlying ontology contains all required concepts. In conclusion, the evaluation shows that SONADUS-KB is suited to process all kinds of queries on the change history of the AUs of Switzerland.

Query performance

In addition to insight about query support, the evaluation has exposed shortcomings of SONADUS, SPARQL, and ontologies in general. First of all, the rather elaborate structure of SONADUS complicates the formulation of queries. This increases the length of SPARQL queries in many cases and leads to a prolonged query response time. Query processing is thus likely to be less efficient in contrast to application ontologies with a simpler structure such as SAPO. Furthermore, ontologies require dates to be stored by a data type property to provide for temporal comparisons which adds further complexity to SONADUS. In order to oppose this, SONADUS does not store FSO numbers as integers and names as literals. At the same time, this implicates that users must know the number intervals and the naming convention of SONADUS, which eventually complicates querying again. Consequently, there is a tradeoff between exact modeling and efficient querying. This does not yet mean that a simple structure automatically improves efficiency. Efficient query processing not only depends on the structure of an ontology, but is also contingent upon the formulation of queries. Experiences from the evaluation suggest that FILTER functions slow down query processing, in particular if it requires scanning a high number of triples. This thesis, however, does not address issues about efficiency and therefore refers to research that specifically focuses on that.

9.7 Challenges to the creation of a knowledge base

The practical experiences gained by completing the four steps provide for a discussion of the last research question.

RQ5: What are the challenges faced when building a conceptualization of the change history of the administrative units of Switzerland? What are the difficulties encountered when creating and querying a knowledge base based on this conceptualization?

The process from building a conceptualization to creating a knowledge base included significant challenges. As described in Chapter 6.1, the conceptualization itself was designed in the ontology editor Protégé. The creation of the conceptualization thus concurred with its actual specification as an OWL ontology. Since Protégé provides for the easy creation and editing of ontologies, the real challenges lay in the conceptual work. The first of the four major challenges was to attach the draft ontology to BFO because this demands a complete understanding of the concepts of this upper ontology. As these concepts are highly abstract, it takes patience and time to obtain such a full comprehension. Second, the integration of the two different conceptualizations of the HMR+ and the geometric datasets required intensive work. This results from the fact that it is very difficult to develop a conceptualization that differs markedly from the prevailing paradigm. Geometries, for instance, are usually captured in a snapshot-like manner but are both objects, lives, and two-dimensional regions in SONADUS. Moreover, SONADUS reflects the insight that the geometries must represent independent objects and not simple attributes of HMR+ entries. Third, the development of a conceptualization confronts the creator with the tradeoff between maximum information and a simple structure. A conceptualization with fewer categories and relationships may both facilitate the formulation of queries and enhance query performance. Adding extra information to and placing further restriction on an ontology entails higher storage requirements and necessary computing power to perform reasoning. Hence, one has to discern the essential information whose advantages outweigh the aforesaid drawbacks. Fourth, inexperienced developers may have difficulties with assessing whether a conceptualization or ontology provides for efficient querying in practice. This not only requires great imagination, but also demands a profound knowledge of query languages and query interfaces. Aside from the determination of the relevant information, the formulation of a clear and concise conceptualization presupposes regular consistency checks, powers of imagination, and technological know-how.

The creation of a knowledge base likewise includes considerable challenges. In case of SONADUS-KB, particular difficulties occurred when extracting the triple statements and loading the knowledge base into a triplestore. In contrast, encoding the triples into RDF/XML syntax did not pose a serious challenge. The retrieval of triple statements from a relational database required the formulation of SQL queries of enormous length and complexity. This arises from the fact that certain concepts of SONADUS such as WOs and POGs do not exist in the source data. Among all SQL queries, the construction of POGs and PLGs presented the greatest difficulties. The main reason for this was that the validity intervals of PLGs were shortened or extended depending on whether a spatial change occurred. Furthermore, errors and special cases in the source data complicated querying in many cases. Aside from detecting and keeping track of errors, the modification of queries to counteract the effects of these errors required a fair amount of work. Due to explicit statements of exceptions, specific modifications also prevent the reuse of queries for other issues of the same dataset. After completing the knowledge base, loading the OWL text file into the triplestore, Virtuoso, presented a further challenge. Though nowhere stated, Virtuoso does not store RDF graphs that exceed a size of approximately 10 MB. The fact that larger files can nevertheless be loaded into the DAV RDF sink causes even more confusion. Moreover, a multitude of modifiable parameters is opposed to an incomplete description of Virtuoso, which makes it difficult to find out why something does not work. In this case, the solution was to split SONADUS-KB into several files that did not exceed 10 MB. Overall, the experiences of this work show that both creating and querying a knowledge base requires a lot of work. Challenges faced when formulating SQL queries depend heavily upon the model of the source data. Difficulties encountered when using Virtuoso or similar software also result from a lack of experience.

10 Conclusion

10.1 Summary

This thesis has presented an application ontology of AUs that endows AUs with an identity and integrates the conceptualizations of different datasets. The ontology provides an answer to the need of WSL for an information system that supports spatiotemporal queries on the change history of the AUs of Switzerland. Hence, this study set out to determine useful approaches in the literature on spatiotemporal databases and ontologies in a first stage. In a second stage, the source data were analyzed which included both the HMR+ and the geometries. Based on these insights, this work constructed an OWL ontology that builds on the upper ontology BFO. Using the ontology with the inferred axioms as T-Box, a knowledge base was created with 167’843 individuals and 915’737 triples in the A-Box. In a last stage, the triplestore and SPARQL endpoint Virtuoso served to store and evaluate the knowledge base against predefined test queries.

10.2 Contributions

This thesis makes the following contributions:

- Literature review of the relevant approaches that support queries on discrete change of objects with crisp boundaries while particularly considering work on AUs
- Detection and description of errors in the HMR+ and the geometric datasets
- Creation of maps that depict the occurrence of each change type in Switzerland since 1990
- Construction of an application ontology that builds on the upper ontology BFO, endows AUs with an identity, takes account of the separate evolution of the HMR+ and the geometric datasets, and is thus able to assign each AU the geometries with which it temporally overlaps
- Creation of a knowledge base in RDF/XML syntax that contains the entire change history of the AUs of Switzerland between 1960 and 2010
- Evaluation showing that the knowledge base supports a wide range of query types expressed in SPARQL

10.3 Findings

By reviewing the literature, this study has found that in practice many spatiotemporal databases adopt traditional approaches. Despite wide application, traditional approaches are not suited to capture change. While object-centered and event-oriented approaches focus on a world that comprises objects and events, respectively, the upper ontology BFO incorporates both views. Thus, BFO provides a sound philosophical basis to formally define the entities of a domain and their mutual relationships. Using the example of the HMR+ and the geometric datasets, this work has shown that BFO is predestined to integrate different conceptualizations. Aside from these benefits, using BFO as a reference framework entails major drawbacks. As the examples of SONADUS and SAPO demonstrate, comparable ontologies become significantly larger when building on an upper ontology. Hence, applying BFO leads to an elaborate structure that not only requires more storage capacity, but also complicates the formulation of queries. Both eventually impairs the efficiency of an information system to process queries. Although this thesis has proposed measures to alleviate this shortcoming, the potential to simplify the structure of SONADUS is limited as long as the ontology builds on BFO. A detachment from BFO would widen the scope of possible measures, but would also implicate that major advantages of upper ontologies are eschewed. Currently, there is no comparable ontology of AUs that also uses BFO as a reference framework. This thesis was therefore unable to determine whether or not SONADUS benefits from enhanced interoperability and knowledge sharing. Despite limitations, this work has proved that SONADUS represents a successful framework to query the change history of the AUs of Switzerland between 1960 and 2010.

Aside from findings on data modeling, this research has yielded insight into the application of Semantic Web software, such as Pellet, Protégé, and Virtuoso. In contrast to database programs such as Oracle, the former are relatively new and usually open source. Moreover, it has become evident that Semantic Web software, and in particular Virtuoso, is not very user-friendly. In general, therefore, it seems that besides research this software will not become widely applied in the near future. Not only Semantic Web software, but also the source data have to be improved. Both the HMR+ and the geometric datasets contain a significant number of errors which suggests uncertainties about and inaccuracies in updating the data. An interesting finding to emerge from this study is that the creation of maps represents a powerful means to detect errors and inconsistencies both in and between thematic and geometric datasets. At the same time, it has been shown that these change maps neither accurately depict municipalities with small areas nor show the chronological sequence of changes. This work thereby confirms that static maps do not suffice to visualize and explore dynamic phenomena such as the evolution of AUs.

10.4 Outlook

This research has thrown up many questions in need of further investigation. In a narrower context, further research has to determine the effects of the suggested measures intended to enhance the structure of SONADUS. This particularly concerns effects on query performance and storage requirements. Provided that no relevant information gets lost, a further study could quantify the potential to reduce the size of SONADUS that would result from a detachment from BFO. As soon as other ontologies of AUs build on BFO, it would be interesting to assess whether SONADUS benefits from enhanced interoperability and knowledge sharing. This would promote the vision to connect the change histories of Europe in the long run. Such a project would require an ontology that negotiates meaning between the different administrative subdivisions of Europe. The ontology of López-Pellicer et al. (2008) indeed represents such an approach. However, this ontology must first integrate the temporal dimension in order to be able to model a dynamic evolution of AUs. In a broader context, further research might identify which applications are suited to Semantic Web technologies. More specifically, a future study could assess in which applications ontologies outperform databases with respect to query support, query performance, storage requirements, and maintenance costs. Although the latter have not been addressed in this work, the maintenance costs are a crucial factor in deciding whether or not to implement an approach. Hence, a further study may quantify and compare the costs of keeping a database or knowledge base up-to-date. Besides that, future research could develop guidelines to avoid redundancy in ontologies. It would be useful to have guidelines that show how to enrich an ontology with maximum information while keeping its structure as simple as possible. Like a database normalization, such guidelines could avoid redundancy in ontologies and knowledge bases.

Regardless of these recommendations, SONADUS-KB has to be embedded into an information system with a user-friendly interface to query and display spatial data. The interface should offer a facility to specify the parameters of a query that does not require knowledge of SPARQL. Although intended to provide the DNL with additional query capabilities, SONADUS-KB can also be used in other applications. However, in order to completely fulfill the requirements of WSL, the information system must perform spatial overlays of biotope objects and municipality boundaries.

References

Abdelmoty, A.I., Smart, P.D., Jones, C.B., Fu, G., and D. Finch (2005): A critical evaluation of ontology languages for geographic information retrieval on the Internet. In *Journal of Visual Languages and Computing* 16: 331-358.

Abraham, T. and J.F. Roddick (1999): Survey of Spatio-Temporal Databases. In *GeoInformatica* 3(1): 61-99.

Agarwal, P. (2005): Ontological considerations in GIScience. In *International Journal of Geographical Information Science* 19(5): 501-536.

Antoniou, G. and F. van Harmelen (2009): Web Ontology Language: OWL. In *Handbook on Ontologies*, eds. S. Staab and R. Studer, 91-110. Springer.

Armstrong, M.P. (1988): Temporality in spatial databases. In *Proceedings of GIS/LIS'88, San Antonio, TX*: 880-889.

Arp, R. and B. Smith (2008): Function, Role, and Disposition in Basic Formal Ontology. In *Proceedings of Bio-Ontologies Workshop (ISMB2008)*: 45-48.

Atkinson, M., Bancilhon, F., DeWitt, D., Dittrich, K., Maier, D., and S. Zdonik (1989): The Object-Oriented Database System Manifesto. In *Proceedings of the International Conference on Deductive and Object-Oriented Databases (DOOD)*: 40-57.

Baader, F. (2003): Description Logic Terminology. In *The Description Logic Handbook: Theory, Implementation, and Applications*, eds. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P.F. Patel-Schneider, 495-505. Cambridge University Press.

Baader, F. and W. Nutt (2003): Basic Description Logics. In *The Description Logic Handbook: Theory, Implementation, and Applications*, eds. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P.F. Patel-Schneider, 47-100. Cambridge University Press.

Baader, F., Horrocks, I., and U. Sattler (2005): Description Logics as Ontology Languages for the Semantic Web. In *Mechanizing Mathematical Reasoning: Essays in Honor of Jörg H. Siekmann on the Occasion of his 60th Birthday, Lecture Notes in Artificial Intelligence* 2605: 228-248. Springer.

Baader, F., Horrocks, I., and U. Sattler (2007): Description Logics. In *Handbook of Knowledge Representation*, eds. F. van Harmelen, V. Lifschitz, and B. Porter, 135-180. Elsevier.

Bauer-Messmer, B., Wotruba, L., Müller, K., Bischof, S., Grütter, R., Scharrenbach, T., Meile, R., Hägeli, M., and J. Schenker (2009): The Data Centre Nature and Landscape (DNL): Service Oriented Architecture, Metadata Standards and Semantic Technologies in an Environmental Information System. In *Proceedings of the 23rd International Conference on Informatics for Environmental Protection (EnviroInfo 2009), Berlin, Germany*, eds. V. Wohlgemuth, B. Page, and K. Voigt, 101-112.

Becker, L., Voigtmann, A., and K.H. Hinrichs (1996): Temporal Support for Geo-Data in Object-Oriented Databases. In *Proceedings of Database and Expert Systems Applications*: 79-93.

Beckett, D. and T. Berners-Lee (2008): Turtle – Terse RDF Triple Language. W3C Team Submission 14 January 2008. http://www.w3.org/TeamSubmission/turtle/, accessed January 10, 2011.

Berners-Lee, T., Hendler, J., and O. Lassila (2001): The Semantic Web. In *Scientific American* 284(5): 34-43.

Berry, B.J.L. (1964): Approaches to regional analysis: a synthesis. In *Annals of the Association of American Geographers* 54(1): 2-11.

Bianchi, S. (2005): Historisches Lexikon der Schweiz: Castel San Pietro. http://www.hls-dhs-dss.ch/textes/d/D2229.php, accessed December 3, 2010.

Bittner, T., Donnelly M., and B. Smith (2004): Endurants and perdurants in directly depicting ontologies. In *AI Communications* 14(4): 247-258.

Brickley, D. and R.V. Guha (2004): RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-schema/, accessed January 12, 2011.

Bubenko, J.A. (1977): The Temporal Dimension in Information Modeling. In *Architecture and Models in Data Base Management Systems*, ed. G.M. Nijssen, 93-118. Amsterdam: North-Holland.

Claramunt, C. and M. Thériault (1995): Managing Time in GIS: An Event-Oriented Approach. In *Proceedings of the International Workshop on Temporal Databases: Recent Advances in Temporal Databases, Zurich, Switzerland*, eds. J. Clifford and A. Tuzhilin, 23-42. London, UK: Springer.

Clark & Parsia (2011): Pellet: OWL 2 Reasoner for Java. http://clarkparsia.com/pellet, accessed February 24, 2011.

CultureSampo (2011): Finnish Culture on the Semantic Web 2.0: Historical areas. http://www.kulttuurisampo.fi/historiallisetAlueet.shtml, accessed March 16, 2011.

De Giacomo, G. and M. Lenzerini (1996): TBox and ABox Reasoning in Expressive Description Logics. In *Proceedings of the 5th International Conference on Principles of Knowledge Representation and Reasoning (KR'96)*, eds. L.C. Aiello, J. Doyle, and S. Shapiro, 316-327.

De Moor, M. (2000): The Quantitative Database of Belgian Municipalities, 1796-1970. http://www.geog.port.ac.uk/hist-bound/papers/belgian_paper.htm, accessed September 7, 2010.

Erling, O. and I. Mikhailov (2009): RDF Support in the Virtuoso DBMS. In *Networked Knowledge – Networked Media, Studies in Computational Intelligence* 221, eds. S. Schaffert et al., 7-24. Springer.

Fischer, F., Unel, G., Bishop, B., and D. Fensel (2010): Towards a Scalable, Pragmatic Knowledge Representation Language for the Web. In *Perspectives of Systems Informatics, Lecture Notes in Computer Science* 5947: 124-134.

FOEN Federal Office for the Environment (2010): Thema Schutzgebiete und Biotopinventare. http://www.bafu.admin.ch/schutzgebiete-inventare/index.html?lang=de, accessed January 3, 2011.

FOT Federal Office of Topography (swisstopo) (2006): GG25: Die digitalen administrative Grenzen der Schweiz. http://www.swisstopo.admin.ch/internet/swisstopo/de/home/products/landscape/gg25.parsysrelated1.37450.downloadList.64853.DownloadFile.tmp/gg25infode.pdf, accessed October 8, 2010.

FOT Federal Office of Topography (swisstopo) (2010): GG25. http://www.toposhop.admin.ch/shop/products/landscape/gg25_1, accessed November 30, 2010.

Frank, A.U. (2001): Socio-Economic Units: Their Life and Motion. In *Life and Motion of Socio-Economic Units*, eds. J. P. Cheylan, A. U. Frank, and J. Raper, 21-34. London, UK: Taylor & Francis.

FSO Federal Statistical Office (2007a): Generalisierte Gemeindegrenzen der Schweiz: GEOSTAT-Datenbeschreibung. http://www.bfs.admin.ch/bfs/portal/de/index/dienstleistungen/geostat/datenbeschreibung/generalisierte_gemeindegrenzen.Document.97376.pdf, accessed October 8, 2010.

FSO Federal Statistical Office (2007b): Historisiertes Gemeindeverzeichnis der Schweiz: Erläuterungen und Anwendungen. http://www.bfs.admin.ch/bfs/portal/it/index/news/publikationen.Document.92104.pdf, accessed July 12, 2010.

FSO Federal Statistical Office (2010): Historisiertes Gemeindeverzeichnis der Schweiz. http://www.bfs.admin.ch/bfs/portal/de/index/infothek/nomenklaturen/blank/blank/gem_liste/02.html, accessed July 12, 2010.

Fu, G., Jones, C.B., and A.I. Abdelmoty (2005): Building a Geographical Ontology for Intelligent Spatial Search on the Web. In *Proceedings of IASTED International Conference on Databases and Applications*: 167-172.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., and L. Schneider (2002): Sweetening Ontologies with DOLCE. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, Lecture Notes in Computer Science* 2473: 223-233. Springer.

Gregory, I.N. (2002): Time-variant GIS Databases of Changing Historical Administrative Boundaries: A European Comparison. In *Transactions in GIS* 6(2): 161-178.

Grenon, P. and B. Smith (2004): SNAP and SPAN: Towards Dynamic Spatial Ontology. In *Spatial Cognition and Computation* 5(1): 69–104.

Gruber, T.R. (1993): A Translation Approach to Portable Ontology Specifications. In *Knowledge Acquisition* 5: 199-220.

Guarino, N. and P. Giaretta (1995): Ontologies and Knowledge Bases: Towards a Terminological Clarification. In *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, ed. N. Mars, 25-32. Amsterdam: IOS Press.

Guarino, N. (1998): Formal Ontology and Information Systems. In *Formal Ontology in Information Systems, Proceedings of FOIS'98, Trento, Italy,* ed. N. Guarino, 3-15. Amsterdam: IOS Press.

Gutiérrez, G.A., Navarro, G., Rodríguez, A. González, A., and J. Orellana (2005): A Spatio-Temporal Access Method based on Snapshots and Events. In *Proceedings of the13th Annual ACM International Workshop on Geographic Information Systems, Bremen, Germany*: 115-124.

Hornsby, K. and M.J. Egenhofer (2002): Modeling Moving Objects over Multiple Granularities. In *Annals of Mathematics and Artificial Intelligence*: 177-194.

Hornsby, K. and K. Joshi (2010): Combining ontologies to automatically generate temporal perspectives of geospatial domains. In *GeoInformatica* 14: 481-505.

Horridge, M. (2009): A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.2. http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/, accessed September 13, 2010.

Horrocks, I., Kutz, O., and U. Sattler (2006): The Even More Irresistible SROIQ. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006), Lake District, UK*: 57-67.

IFOMIS Institute for Formal Ontology and Medical Information Science (2010): Basic Formal Ontology (BFO). http://www.ifomis.org/bfo, accessed October 18, 2010.

Jean, S., Pierra, G., and Y. Ait-Ameur (2007): Domain Ontologies : A Database-Oriented Analysis. In *Web Information Systems and Technologies, Lecture Notes in Business Information Processing* 1, eds. J. Filipe, J. Cordeiro, and V. Pedrosa, 238-254. Springer.

Jiang, J. and M.F. Worboys (2009): Event-based topology for dynamic planar areal objects. In *International Journal of Geographical Information Science* 23(1): 33-60.

Johansson, I. (2005): Qualities, Quantities, and the Endurant-Perdurant Distinction in Top-Level Ontologies. In *WM 2005: Professional Knowledge Management Experiences and Vision, Kaiserslautern, Germany*, eds. K.-D. Althoff et al.: 543-550.

Jones, C.B., Purves, R.S., Ruas, A., Sanderson, M., Sester, M., van Kreveld, M., and R. Weibel (2002): Spatial Information Retrieval and Geographical Ontologies: An Overview of the SPIRIT Project. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland*: 387-388.

Jones, C.B., Abdelmoty, A.I., Finch, D., Fu., G., and S. Vaid (2004): The SPIRIT Spatial Search Engine: Architecture, Ontologies and Spatial Indexing. In *Geographic Information Science, Third International Conference, GIScience 2004, Adelphi, MD, Lecture Notes in Computer Science* 3234, eds. M.J. Egenhofer, C. Freksa, and H.J. Miller, 125-139. Springer.

Kauppinen, T., Väätäinen J., and E. Hyvönen (2008): Creating and Using Geospatial Ontology Time Series in a Semantic Cultural Heritage Portal. In *Proceedings of the 5th European Semantic Web Conference (2008), Tenerife, Spain.*

Kemp, Z. and A. Kowalczyk (1994): Incorporating the Temporal Dimension in a GIS. In *Innovations in GIS*, ed. M.F. Worboys, 89-103. London, UK: Taylor & Francis.

Kristiansson, G. (2000): Project: National Archival Database (Sweden). http://www.geog.port.ac.uk/hist-bound/project_rep/proj_sweden_NAD.htm, accessed September 7, 2010.

Kunowski, P. (2010): Simple Ontology Visualization API (SOVA). http://protegewiki.stanford.edu/wiki/SOVA, accessed February 1, 2011.

Langran, G. and N.R. Chrisman (1988): A Framework for Temporal Geographic Information. In *Cartographica* 25(3): 1-14.

Langran, G. (1992): *Time in Geographic Information Systems*. London, UK: Taylor & Francis.

Lanzenberger, M., Sampson, J., and M. Rester (2010): Ontology Visualization: Tools and Techniques for Visual Representation of Semi-Structured Meta-Data. In *Journal of Universal Computer Science* 16(7): 1036-1054.

Lohfink, A., Carnduff, T., Thomas, N., and M. Ware (2007): An Object-Oriented Approach to the Representation of Spatiotemporal Geographic Features. In *Proceedings of the 15th International Symposium on Advances in Geographic Information Systems (ACM GIS 2007), Seattle, WA.*

López-Pellicer, F., Florczyk, A.J., Lacasta, J., Zarazaga-Soria, F.J., and P.R. Muro-Medrano (2008): Administrative Units, an Ontological Perspective. In *Proceedings of ER Workshops 2008, Lecture Notes in Computer Science* 5232, eds. I.-Y. Song et al., 354-363. Springer.

Manola, F. and E. Miller (2004): RDF Primer. W3C Recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-primer-20040210/, accessed January 10, 2011.

Matthews, A. (2008): Understanding SPARQL: Create journaling micro-blogs with the semantic Web. http://www.ibm.com/developerworks/xml/tutorials/x-sparql/x-sparql-pdf.pdf, accessed October 27, 2010.

McGuinness, D.L. and F. van Harmelen (2004): OWL Web Ontology Language Overview. W3C Recommendation 10 February 2004. http://www.w3.org/TR/owl-features/, accessed January 10, 2011.

Niles, I. and A. Pease (2001): Towards a Standard Upper Ontology. In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001), Ogunquit, ME*, eds. C. Welty and B. Smith, 2-9.

Noy, N.F. (2004): Semantic Integration: A Survey of Ontology-Based Approaches. In *ACM SIGMOD Record* 33(4): 65-70.

O’Connor, M.J. and A.K. Das (2009): SQWRL: A Query Language for OWL. In *Proceedings of the Sixth OWL Experiences and Directions Workshop (OWLED 2009)*.

Pelekis, N., Theodoulidis, B., Kopanakis I., and Y. Theodoridis (2005): Literature Review of Spatio-Temporal Database Models. In *Knowledge Engineering Review* 19(3): 235-274.

Peuquet, D.J. and N. Duan (1995): An Event-Based Spatiotemporal Data Model (ESTDM) for Temporal Analysis of Geographical Data. In *International Journal of Geographical Information Science* 9 (1): 7-24.

Plumejeaud, C., Moisuc, B., Bimonte, S., Villanova, M., and J. Gensel (2009) : An Object-Oriented Model for the Sustainable Management of Evolving Spatio-temporal Information. In *Geocomputation & Urban Planning* 176, eds. B. Murgante, G. Borruso, and A. Lapucci, 167-186. Springer.

Polleres, A. (2007): From SPARQL to Rules (and back). In *Proceedings of the 16th International World Wide Web Conference (WWW), Banff, Canada*: 787-796.

Prud'hommeaux, E. and A. Seaborne (2008): SPARQL Query Language for RDF. W3C Recommendation 15 January 2008. http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/, accessed January 10, 2011.

SCBIR Stanford Center for Biomedical Informatics Research (2010): Protégé. http://protege.stanford.edu/, accessed December 7, 2010.

Schreven, L. (2000): Project: NL-KAART (Netherlands). http://www.geog.port.ac.uk/hist-bound/project_rep/proj_NLKAART.htm, accessed September 7, 2010.

Schuler, M., Dessemontet, P., and D. Joye (2005): Die Raumgliederungen der Schweiz. Eidgenössische Volkszählung 2000. Swiss Federal Statistical Office. http://www.bfs.admin.ch/bfs/portal/de/index/regionen/22/lexi.Document.64476.pdf, accessed February 7, 2011.

Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., and Y. Katz (2007): Pellet: A Practical OWL-DL Reasoner. In *Journal of Web Semantics* 5(2).

Sirin, E., Bulka, B., and M. Smith (2010): Terp: Syntax for OWL-friendly SPARQL queries. In *Proceedings of the Seventh OWL Experiences and Directions Workshop (OWLED 2010)*.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., the OBI Consortium, Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R.H., Shah, N., Whetzel, P.L., and S. Lewis (2007): The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. In *Nature Biotechnology* 25(11): 1251-1255.

Snodgrass, R. and I. Ahn (1985): A Taxonomy of Time in Databases. In *Proceedings of the SIGMOD'85 Conference*: 236-245.

Spear, A.D. (2006): Ontology for the Twenty First Century: An Introduction with Recommendations. Institute for Formal Ontology and Medical Information Science, Saarbrücken, Germany. http://www.ifomis.org/bfo/documents/manual.pdf, accessed January 7, 2011.

SR 451 (Systematische Rechtssammlung), Artikel 18a+b (2008): Bundesgesetz vom 1. Juli 1966 über den Natur- und Heimatschutz (NHG). http://www.admin.ch/ch/d/sr/4/451.de.pdf, accessed December 30, 2010.

SR 510.625 (Systematische Rechtssammlung), Artikel 10 (2008): Verordnung vom 21. Mai 2008 über die geografischen Namen (GeoNV). http://www.admin.ch/ch/d/sr/510_625/a10.html, accessed December 2, 2010.

Stevens, R., Goble, C.A., and S. Bechhofer (2000): Ontology-based knowledge representation for bioinformatics. In *Briefings in Bioinformatics* 1(4): 398-414.

Stock, K.M. (2006): Spatio-Temporal Data Management Using Object Lifecycles: A Case Study of the Australian Capital Territory Spatial Data Management System. In *Journal of Spatial Sciences* 51: 43-58.

Straumann, R. (2010): *Extraction and Characterisation of Landforms from Digital Elevation Models: Fiat Parsing the Elevation Field*. PhD Thesis, Department of Geography, University of Zurich.

Swartout, W. and A. Tate (1999): Ontologies. In *IEEE Intelligent Systems* 14(1): 18-19.

Swiaczny, F. (2000): Project: Jewish Population in C19 & C29 Pfalz. http://www.geog.port.ac.uk/hist-bound/project_rep/proj_judisch_pfalz.htm, accessed September 7, 2010.

Tsarkov, D. and I. Horrocks (2006): FaCT++ Description Logic Reasoner: System Description. In *Proceedings of the Third International Joint Conference on Automated Reasoning (IJCAR 2006), Lecture Notes in Computer Science* 4130, eds. U. Furbach and N. Shankar, 292-297. Springer.

Virtuoso FAQ (2010a): FAQ. http://virtuoso.openlinksw.com/virt_faq/, accessed March 4, 2011.

Virtuoso OpenLink Software (2010b): Product Downloads By Product Family. http://download.openlinksw.com/, accessed March 4, 2011.

Wachowicz, M. and R. Healey (1994): Towards temporality in GIS. In *Innovations in GIS*, ed. M.F. Worboys, 105-121. London, UK: Taylor & Francis.

Weibel, S.L. and T. Koch (2000): The Dublin Core Metadata Initiative: Mission, Current Activities, and Future Directions. In *D-Lib Magazine* 6(12).

Wikipedia (2007): Karte Neuenburgersee. http://de.wikipedia.org/w/index.php?title=Datei:Karte_Neuenburgersee.png&filetimestamp=20070107230104, accessed December 1, 2010.

Wikipedia (2011): Kommunanz. http://de.wikipedia.org/wiki/Kommunanz, accessed February 7, 2011.

Winnige, N. (2000): Project: Historical Atlas of Prussia. http://www.geog.port.ac.uk/hist-bound/project_rep/proj_prussia.htm, accessed September 7, 2010.

Worboys, M.F., Hearnshaw, H.M., and D.J. Maguire (1990): Object-Oriented Data Modelling for Spatial Databases. In *International Journal of Geographical Information Systems* 4(4): 369-383.

Worboys, M.F. (1994): A Unified Model for Spatial and Temporal Information. In *The Computer Journal* 37(1): 26-34.

Worboys, M.F. (2001): Modelling Changes and Events in Dynamic Spatial Systems with Reference to Socio-Economic Units. In *Life and Motion of Socio-Economic Units*, eds. J.P. Cheylan, A.U. Frank, and J. Raper, 129-137. London, UK: Taylor & Francis.

Worboys, M.F. and K. Hornsby (2004): From Objects to Events: GEM, the Geospatial Event Model. In *Geographic Information Science. Third International Conference, GIScience 2004*, eds. M.J. Egenhofer, C. Freksa, and H.J. Miller, 327-343. Springer.

Worboys, M.F. (2005): Event-Oriented Approaches to Geographic Phenomena. In *International Journal of Geographical Information Science* 19(1): 1-28.

WordNet (2011): WordNet: A lexical database for English. Department of Computer Science, Princeton University. http://wordnet.princeton.edu/, accessed January 4, 2011.

Yu, A.C. (2006): Methods in biomedical ontology. In *Journal of Biomedical Informatics* 39: 252-266.

Yuan, M. (1999): Use of a Three-Domain Representation to Enhance GIS Support for Complex Spatiotemporal Queries. In *Transactions in GIS* 3(2): 137-159.

Yuan, M. and J. McIntosh (2002): A Typology of Spatiotemporal Information Queries. In *Mining Spatiotemporal Information Systems*, eds. K. Shaw, R. Ladner, and M. Abdelguerfi, 63-82. Kuwer Academic Publishers.

Yuan, M. (2008): Adding Time into Geographic Information System Databases. In *The Handbook of Geographic Information Science*, eds. J.P. Wilson and A.S. Fotheringham, 169-184. Malden, MA: Blackwell Publishing.

Appendix A Source data

A.1 Special cases in the HMR+

1. Annexations where the annexing municipality received a new FSO number

a. Use of new numbers

Eschlikon:

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
12229	4762	Eschlikon	1000	20	1960-01-01	1894	26	1996-12-31
10465	4765	Wallenwil	1000	20	1960-01-01	1894	29	1996-12-31
14091	4724	Eschlikon	1894	26	1996-12-31	1896	26	1996-12-31

Table A.1: Annexation with a new FSO number: Eschlikon

Hohentannen:

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
12130	4492	Hohentannen	1000	20	1960-01-01	1918	26	1998-12-31
12222	4491	Heldswil	1000	20	1960-01-01	1918	29	1998-12-31
14118	4495	Hohentannen	1918	26	1999-01-01			

Table A.2: Annexation with a new FSO number: Hohentannen

Hüttlingen:

HIST_NR_GDE	BFS_GDE_NR	AMTL_GDE_NAME	MUT_NR_AUFN	ART_AUFN	DATUM_AUFN	MUT_NR_AUFH	ART_AUFH	DATUM_AUFH
12185	4583	Hüttlingen	1000	20	1960-01-01	1922	26	1998-12-31
12230	4581	Eschikofen	1000	20	1960-01-01	1922	29	1998-12-31
12014	4582	Harenwilen	1000	20	1960-01-01	1922	29	1998-12-31
12431	4584	Mettendorf	1000	20	1960-01-01	1922	29	1998-12-31
14122	4590	Hüttlingen	1922	26	1999-01-01			

Table A.3: Annexation with a new FSO number: Hüttlingen

Langrickenbach:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
12622	4678	Langrickenbach	1000	20	1960-01-01	1910	26	1997-12-31
12197	4676	Dünnershaus	1000	20	1960-01-01	1910	29	1997-12-31
12312	4677	Herrenhof	1000	20	1960-01-01	1910	29	1997-12-31
12961	4663	Schönenbaum-garten	1000	20	1960-01-01	1910	29	1997-12-31
10321	4679	Zuben	1000	20	1960-01-01	1910	29	1997-12-31
14109	4681	Langrickenbach	1910	26	1998-01-01			

Table A.4: Annexation with a new FSO number: Langrickenbach

Lommis:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
12802	4738	Lommis	1000	20	1960-01-01	1866	26	1994-12-31
10346	4739	Weingarten	1000	20	1960-01-01	1866	29	1994-12-31
12649	4737	Kalthäusern	1000	20	1960-01-01	1866	29	1994-12-31
14059	4741	Lommis	1866	26	1995-01-01			

Table A.5: Annexation with a new FSO number: Lommis

Pfyn:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
13144	4837	Pfyn	1000	20	1960-01-01	1916	26	1997-12-31
11557	4836	Dettighofen	1000	20	1960-01-01	1916	29	1997-12-31
14116	4841	Pfyn	1916	26	1998-01-01			

Table A.6: Annexation with a new FSO number: Pfyn

b. Use of a number of a municipality that is annexed

Birwinken:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11803	4902	Birwinken	1000	20	1960-01-01	1871	26	1994-12-31
11759	4901	Andwil (TG)	1000	20	1960-01-01	1871	29	1994-12-31
12013	4904	Happerswil-Buch	1000	20	1960-01-01	1871	29	1994-12-31
12698	4905	Klarsreuti	1000	20	1960-01-01	1871	29	1994-12-31
12746	4906	Mattwil	1000	20	1960-01-01	1871	29	1994-12-31
14065	4901	Birwinken	1871	26	1995-01-01			

Table A.7: The annexing municipality inherits an FSO number of an annexed municipality: Birwinken

Erlen:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
12302	4479	Erlen	1000	20	1960-01-01	1855	26	1994-12-31
11476	4476	Buchackern	1000	20	1960-01-01	1855	29	1994-12-31
12272	4477	Engishofen	1000	20	1960-01-01	1855	29	1994-12-31
12278	4478	Ennetaach	1000	20	1960-01-01	1855	29	1994-12-31
12588	4480	Kümmerts-hausen	1000	20	1960-01-01	1855	29	1994-12-31
12896	4509	Riedt	1000	20	1960-01-01	1855	29	1994-12-31
14047	4476	Erlen	1855	26	1995-01-01			

Table A.8: The annexing municipality inherits an FSO number of an annexed municipality: Erlen

Fischingen:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11178	4728	Fischingen	1000	20	1960-01-01	1037	26	1971-12-31
11179	4726	Au (TG)	1000	20	1960-01-01	1037	29	1971-12-31
11180	4727	Dussnang	1000	20	1960-01-01	1037	29	1971-12-31
11181	4729	Oberwangen	1000	20	1960-01-01	1037	29	1971-12-31
11182	4730	Tannegg	1000	20	1960-01-01	1037	29	1971-12-31
13222	4726	Fischingen	1037	26	1972-01-01			

Table A.9: The annexing municipality inherits an FSO number of an annexed municipality: Fischingen

Müllheim:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11270	4832	Müllheim	1000	20	1960-01-01	1020	26	1967-05-31
11271	4831	Langenhart	1000	20	1960-01-01	1020	29	1967-05-31
13205	4831	Müllheim	1020	26	1967-06-01			

Table A.10: The annexing municipality inherits an FSO number of an annexed municipality: Müllheim

Salenstein:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11371	4853	Salenstein	1000	20	1960-01-01	1471	26	1979-05-31
11372	4851	Fruthwilen	1000	20	1960-01-01	1471	29	1979-05-31
11383	4852	Mannenbach	1000	20	1960-01-01	1471	29	1979-05-31
13657	4851	Salenstein	1471	26	1979-06-01			

Table A.11: The annexing municipality inherits an FSO number of an annexed municipality: Salenstein

Sirnach:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
12982	4764	Sirnach	1000	20	1960-01-01	1895	26	1996-12-31
10280	4766	Wiezikon	1000	20	1960-01-01	1895	29	1996-12-31
11515	4761	Busswil (TG)	1000	20	1960-01-01	1895	29	1996-12-31
12202	4763	Horben	1000	20	1960-01-01	1895	29	1996-12-31
14092	4761	Sirnach	1895	26	1996-12-31			

Table A.12: The annexing municipality inherits an FSO number of an annexed municipality: Sirnach

Thundorf:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10482	4612	Thundorf	1000	20	1960-01-01	1860	26	1994-12-31
10276	4740	Wetzikon (TG)	1000	20	1960-01-01	1860	29	1994-12-31
12710	4611	Lustdorf	1000	20	1960-01-01	1860	29	1994-12-31
14053	4611	Thundorf	1860	26	1995-01-01			

Table A.13: The annexing municipality inherits an FSO number of an annexed municipality: Thundorf

Wäldi:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10460	4704	Wäldi	1000	20	1960-01-01	1863	26	1994-12-31
10410	4703	Sonterswil	1000	20	1960-01-01	1863	29	1994-12-31
12280	4701	Engwilen	1000	20	1960-01-01	1863	29	1994-12-31
12825	4702	Lipperswil	1000	20	1960-01-01	1863	29	1994-12-31
14056	4701	Wäldi	1863	26	1995-01-01			

Table A.14: The annexing municipality inherits an FSO number of an annexed municipality: Wäldi

Wängi:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11260	4784	Wängi	1000	20	1960-01-01	1024	26	1968-12-31
11261	4781	Anetswil	1000	20	1960-01-01	1024	29	1968-12-31
11262	4782	Krillberg	1000	20	1960-01-01	1024	29	1968-12-31
11263	4783	Tuttwil	1000	20	1960-01-01	1024	29	1968-12-31
13209	4781	Wängi	1024	26	1969-01-01	1876	26	1995-12-31

Table A.15: The annexing municipality inherits an FSO number of an annexed municipality: Wängi

Wigoltingen:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
10281	4954	Wigoltingen	1000	20	1960-01-01	1872	26	1994-12-31
11788	4951	Bonau	1000	20	1960-01-01	1872	29	1994-12-31
12193	4953	Illhart	1000	20	1960-01-01	1872	29	1994-12-31
12281	4952	Engwang	1000	20	1960-01-01	1872	29	1994-12-31
14066	4951	Wigoltingen	1872	26	1995-01-01			

Table A.16: The annexing municipality inherits an FSO number of an annexed municipality: Wigoltingen

Wuppenau:

HIST_NR _GDE	BFS_GDE _NR	AMTL_GDE_NAME	MUT_NR _AUFN	ART_ AUFN	DATUM_AUFN	MUT_NR _AUFH	ART_ AUFH	DATUM_AUFH
11189	4793	Wuppenau	1000	20	1960-01-01	1035	26	1971-05-31
11190	4791	Heiligkreuz	1000	20	1960-01-01	1035	29	1971-05-31
11191	4792	Hosenruck	1000	20	1960-01-01	1035	29	1971-05-31
13220	4791	Wuppenau	1035	26	1971-06-01			

Table A.17: The annexing municipality inherits an FSO number of an annexed municipality: Wuppenau

2. Mergers without a new FSO number

Former Municipality	New Municipality	FSO No.	Date of Commencement
Albeuve	Haut-Intyamon	2121	01.01.2002
Amlikon	Amlikon-Bissegg	4881	01.01.1995
Avry-devant-Pont	Pont-en-Ogoz	2122	01.01.2003
Avry-sur-Matran	Avry	2174	01.01.2001
Balterswil	Bichelsee-Balterswil	4721	01.01.1996
Biel (BL)	Biel-Benken	2805	01.01.1972
Billens	Billens-Hennens	2063	01.01.1998
Brig	Brig-Glis	6002	01.10.1972
Buch bei Frauenfeld	Uesslingen-Buch	4616	01.01.1995
Bussy-sur-Morges	Bussy-Chardonney	5625	01.04.1961
Corpataux	Corpataux-Magnedens	2184	01.01.1999
Croglio-Castelrotto	Croglio	5178	04.04.1976
Ebnat	Ebnat-Kappel	3352	01.01.1965
Farvagny-le-Grand	Farvagny	2192	01.01.1996
Felben	Felben-Wellhausen	4561	01.06.1983
Gottshaus	Hauptwil-Gottshaus	4486	01.01.1996
Grossgurmels	Gurmels	2262	01.01.1978
Grossguschelmuth	Guschelmuth	2263	01.01.1978
Heinrichswil	Heinrichswil-Winistorf	2521	01.01.1993
Lohn (SO)	Lohn-Ammannsegg	2526	01.01.1993
Lossy	Lossy-Formangueires	2203	01.01.1982
Lussery	Lussery-Villars	5487	01.01.1999
Lüterkofen	Lüterkofen-Ichertswil	2455	01.01.1961
Lüterswil	Lüterswil-Gächliwil	2456	01.01.1995
Marly-le-Grand	Marly	2206	14.03.1970
Martigny-Bourg	Martigny	6136	01.08.1964
Misery	Misery-Courtion	2272	01.01.1997
Montagny-les-Monts	Montagny (FR)	2029	01.01.2000
Montreux-Châtelard	Montreux	5886	01.01.1962
Niedersommeri	Sommeri	4446	01.06.1967
Pont-en-Ogoz	Le Bry	2146	26.01.1970
Praroman	Le Mouret	2220	01.01.2003
Riom	Riom-Parsonz	3536	01.01.1979
Romainmôtier	Romainmôtier-Envy	5761	01.01.1970
Rueyres-Saint-Laurent	Le Glèbe	2223	01.01.2003
St. Antönien Castels	St. Antönien	3893	01.01.1979
Surcasti	Suraua	3599	01.01.2002
Sâles (Gruyère)	Sâles	2152	01.01.2001
Tesserete	Capriasca	5226	18.10.2001
Tinizong	Tinizong-Rona	3541	01.07.1998
Uors (Lumnezia)	Uors-Peiden	3602	24.05.1963

Table A.18: Mergers since 1960 where the newly created municipality has inherited the FSO number of one of its constituent municipalities

A.2 Spelling differences between the HMR and the GZG 1990

1. Upper case

In contrast to the GZG 1990 that spells municipality names in upper case, the HMR capitalizes only the initial letter.

Example: Kreuzlingen

HMR	GZG 1990
Kreuzlingen	KREUZLINGEN

Table A.19: Lower-case spelling of the HMR vs. upper-case spelling of the GZG 1990

2. Umlauts, acute accents, grave accents, and circumflexes

The GZG 1990 does not consider these special characters and spells umlauts with an additional e. Accents and circumflexes are cut off of the respective letters.

Examples: Zurich and Geneva

HMR	GZG 1990
Zürich	ZUERICH
Genève	GENEVE

Table A.20: Removal of special characters in the GZG 1990

3. Abbreviations and spelling of kommunanzen

Both the HMR and the GZG 1990 abbreviate the designation Comunanza for kommunanzen in the Canton of Ticino. The GZG 1990 additionally abbreviates the designation Kommunanz of the only remaining kommunanz in German-speaking Switzerland and, sometimes, also the names of the kommunanzen itself. The spelling is wrong in certain cases. Furthermore, the GZG 1990 uses hyphens instead of obliques to separate the individual components of a kommunanz' name.

The absence of a general rule requires listing all exceptions.

HMR	GZG 1990
C'za Corticiasca/Valcolla	COMUN. CORTICIASCA-VALCOLLA
Kommunanz Gluringen/Ritzingen	COMUN. GLURINGEN-RITZINGEN
C'za Sala Capriasca/Vaglio/Lugaggia	COMUN. SALA CAPRIASCA-VAGLIO
C'za Medeglia/Robasacco	COMUN. MEDEGL.-ROBASSA.
C'za Bidogno/Sala Capriasca/Corticiasca	COMUN. SALA CAPRIASCA-BIDOGNO

Table A.21: Differences in spelling of kommunanzen between the HMR and the GZG 1990

4. Other differences in spelling

The HMR contains the correct name of the former Municipality of Roveredo-Capriasca. The spelling of the GZG 1990 is wrong and has never been an official municipality name.

HMR	GZG 1990
Roveredo-Capriasca	ROVEREDO (TI)

Table A.22: Other differences in spelling

Appendix B Change maps

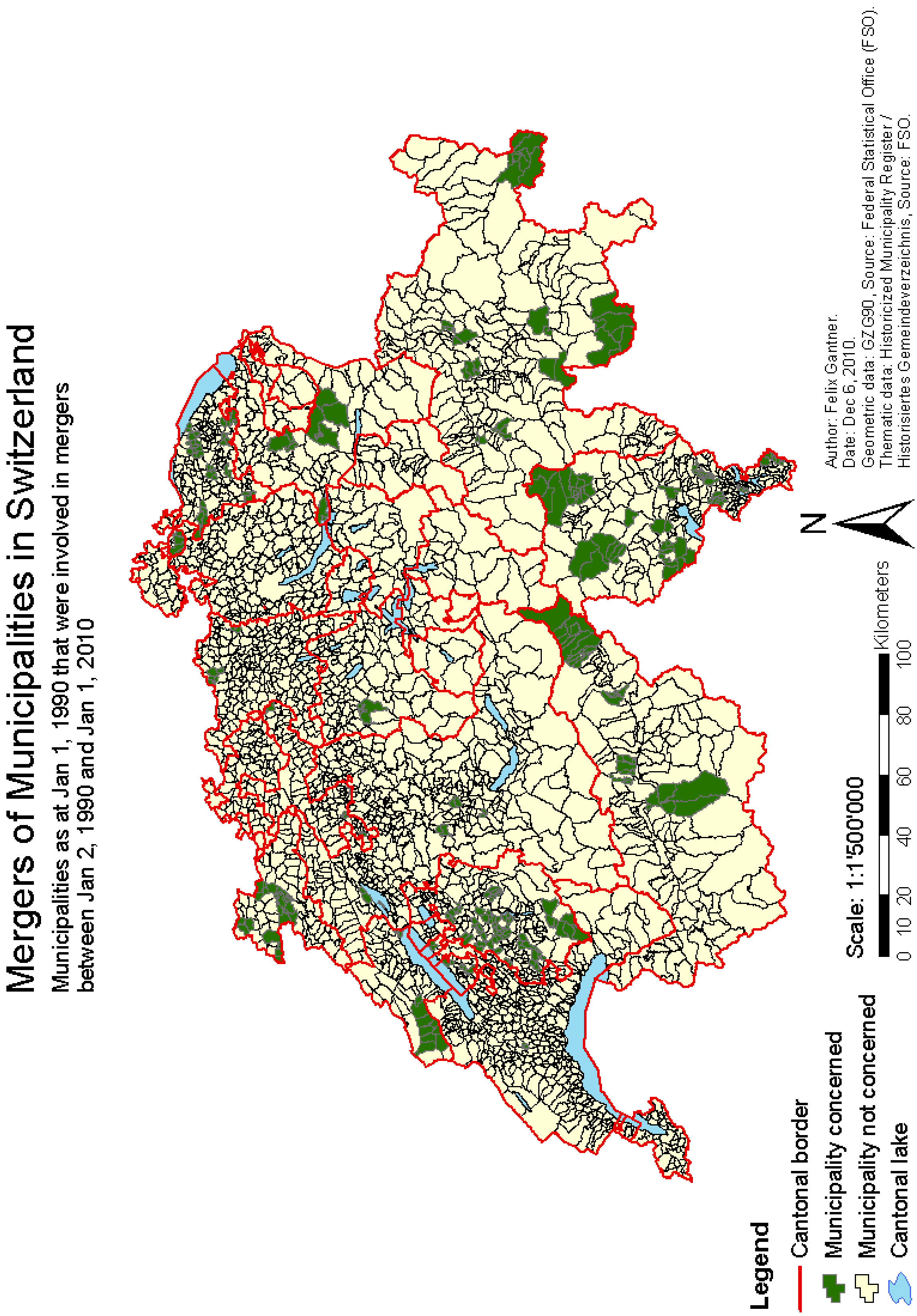


Figure B.1: Mergers of municipalities in Switzerland

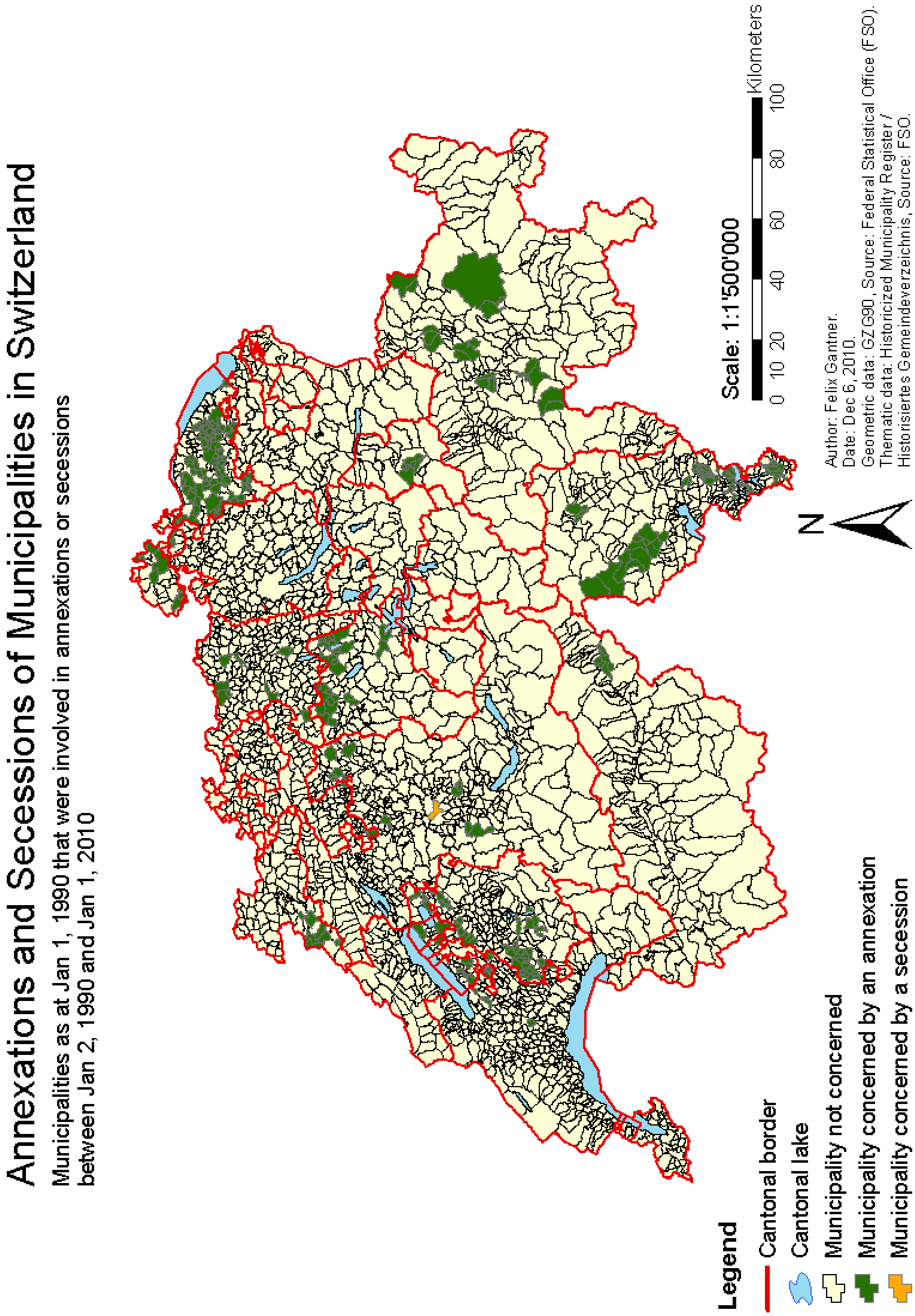


Figure B.2: Annexations and secessions of municipalities in Switzerland

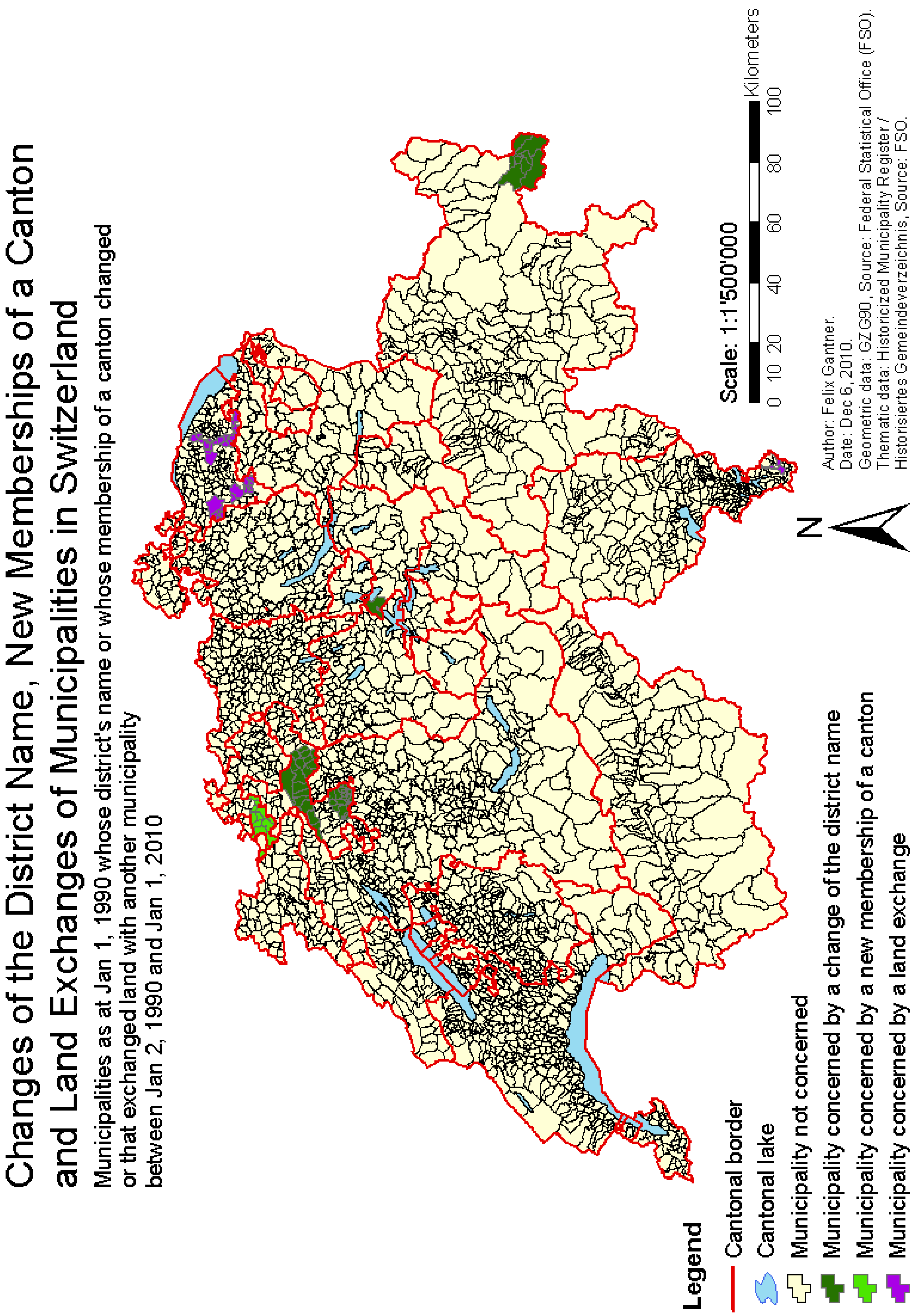


Figure B.3: Changes of the district name, new memberships of a canton, and land exchanges of municipalities in Switzerland

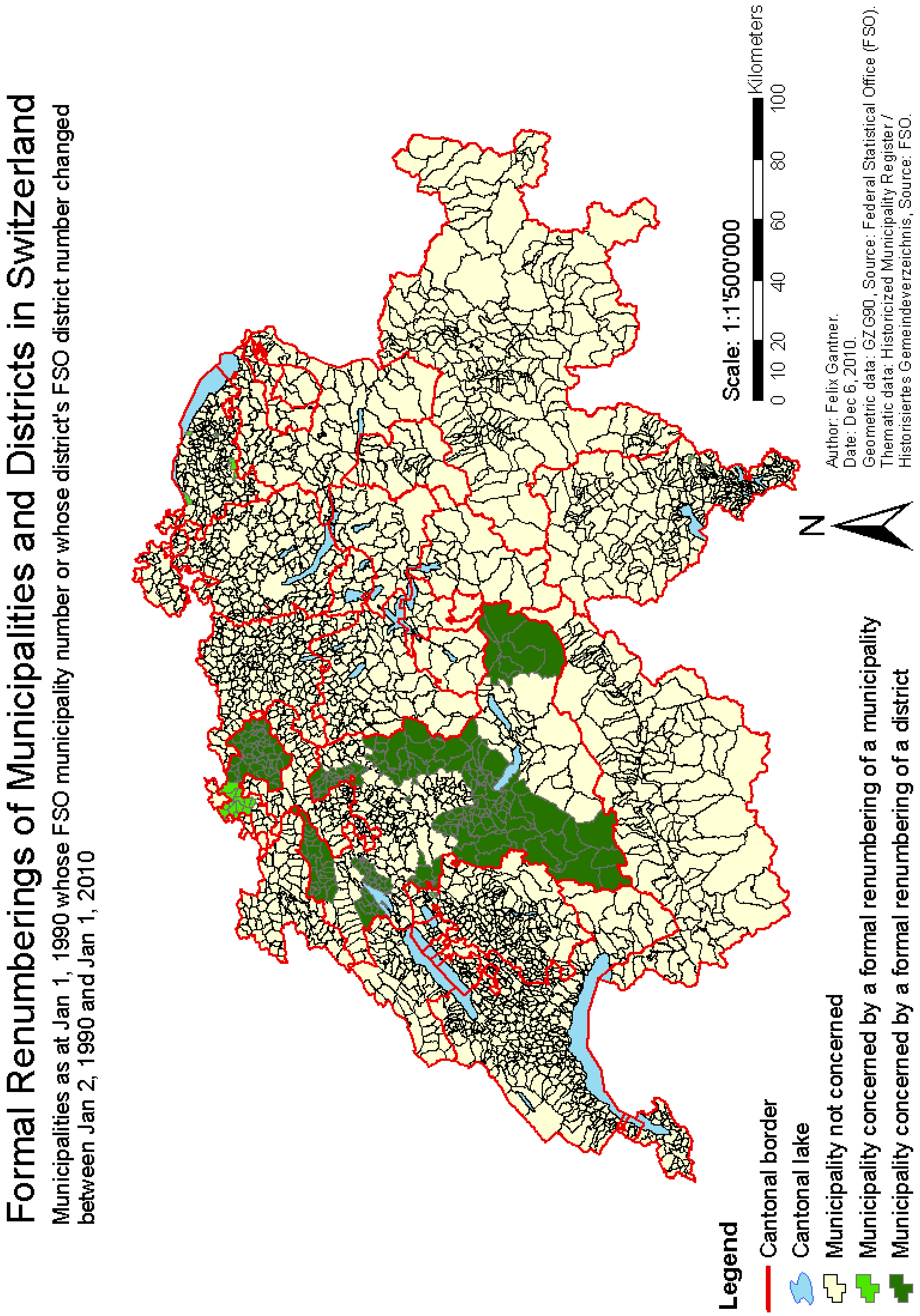


Figure B.4: Formal renumberings of municipalities and districts in Switzerland

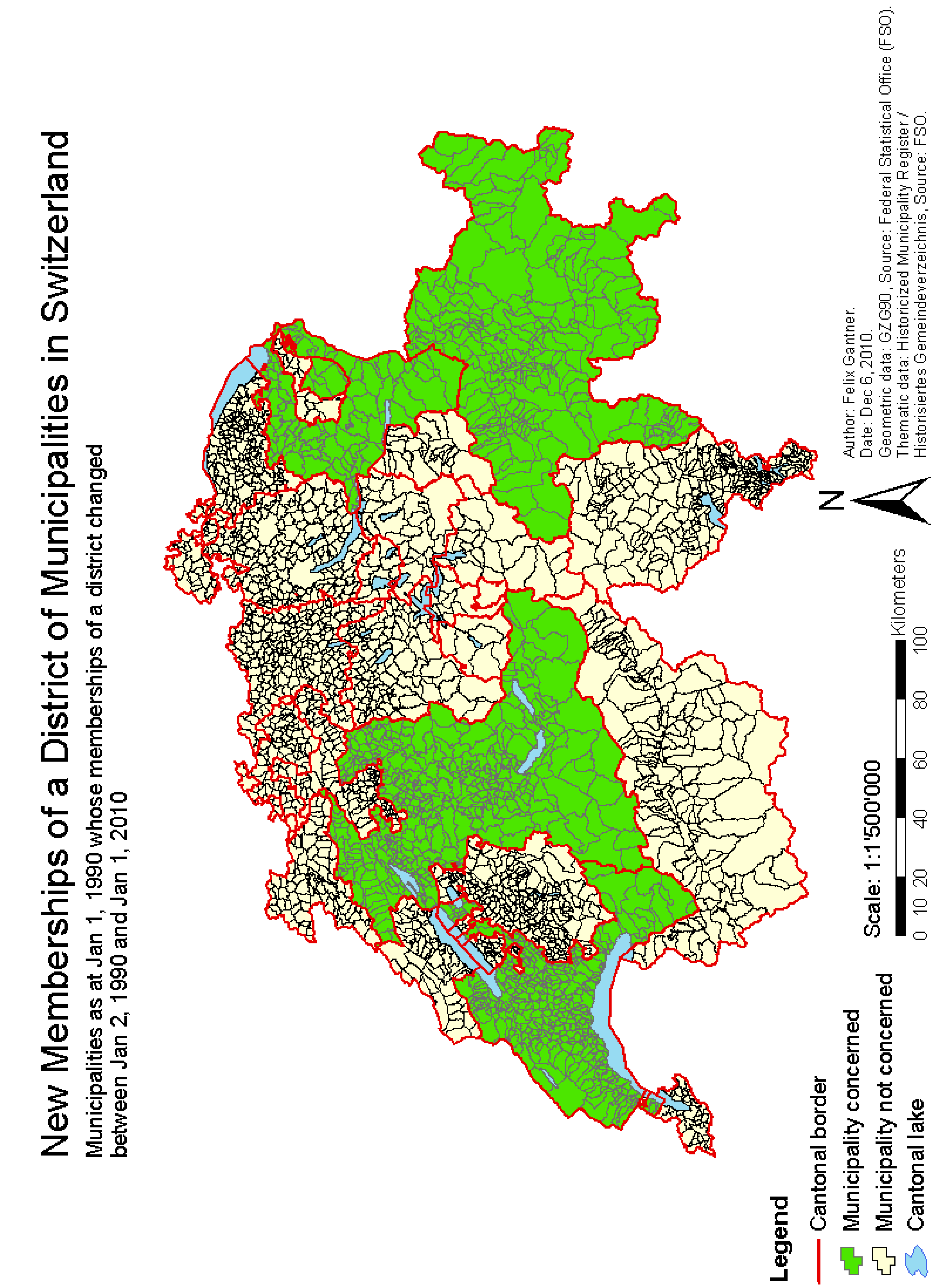


Figure B.5: New memberships of a district of municipalities in Switzerland

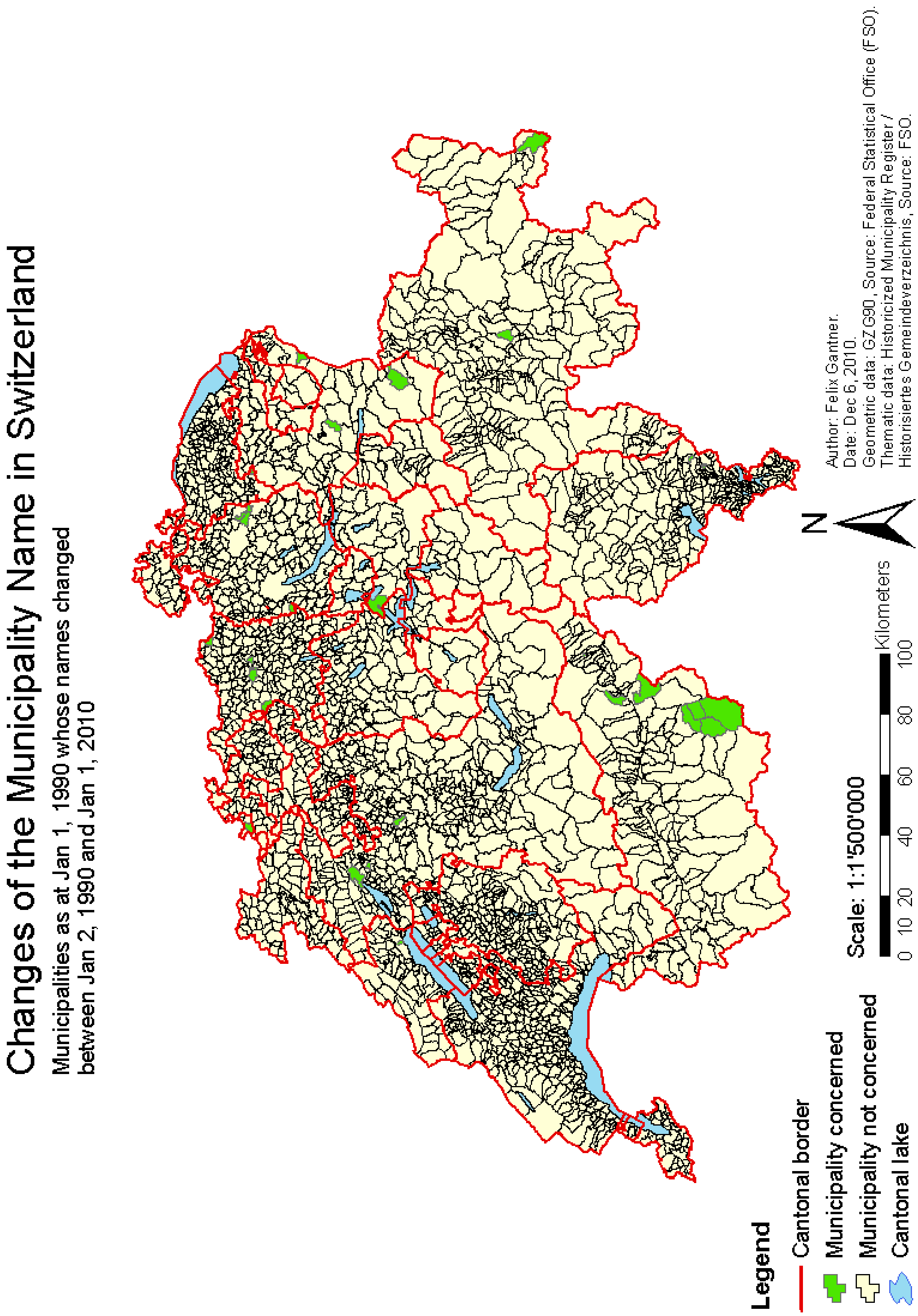


Figure B.6: Changes of the municipality name in Switzerland

Appendix C Number intervals

Number intervals / Nummernbereiche HMR+ entities / Einheiten aus dem Historisierten Gemeindeverzeichnis				
Klasse oder Teilklasse	**Minimum**	**Maximum**	**Spezial**	**Bemerkungen**
TeilobjektVerwaltungseinheitStufeGemeinde	10001	15390		
TeilobjektVerwaltungseinheitStufeBezirk	30001	30292		
TeilobjektVerwaltungseinheitKanton	50101	50126		
LebensabschnittVerwaltungseinheitStufeGemeinde	110001	115390		
LebensabschnittVerwaltungseinheitStufeBezirk	130001	130292		
LebensabschnittVerwaltungseinheitKanton	150101	150126		
GanzesObjektStufeGemeinde	40001	43260		
GanzesObjektStufeBezirk	48001	48239		
GanzesObjektKanton	50001	50026		
GanzesObjektBund	50000	50000		
LebenStufeGemeinde	140001	143260		
LebenStufeBezirk	148001	148239		
LebenKanton	150001	150026		
LebenBund	150000	150000		
Mutationsnummer Stufe Gemeinde	51000	53172	59000	Kode für Letzterfassung
Mutationsnummer Stufe Bezirk	60100	60152	60900	Kode für Letzterfassung
Mutationsnummer Kanton	61000	61001	69000	Kode für Letzterfassung
Mutationsnummern Geometrie	61990	62009		Kode gebildet durch 6 + Update-Jahr
BFS-Gemeindenummer	70001	79760		
BFS-Bezirksnummer	80100	82603		
Kantonsnummer	80001	80026		
ArtAenderung	90020	90031		

Table C.1: Number intervals of the HMR+ entities

Number intervals / Nummernbereiche Geometries on the municipality level / Geometrien Stufe Gemeinde		
Klasse oder Teilklasse	**Minimum**	**Maximum**
TeilobjektGeometrieStufeGemeinde 1960a	19600000011192	19600000011193
TeilobjektGeometrieStufeGemeinde 1960b	19600000010790	19700000013217
TeilobjektGeometrieStufeGemeinde 1990	19900000000001	19900000003251
TeilobjektGeometrieStufeGemeinde 1994	19940000000001	19940000003248
TeilobjektGeometrieStufeGemeinde 1996	19960000000001	19960000003174
TeilobjektGeometrieStufeGemeinde 1998	19980000000001	19980000003147
TeilobjektGeometrieStufeGemeinde 2000	20000000248847	20000004677156
TeilobjektGeometrieStufeGemeinde 2001	20010000248847	20010007749317
TeilobjektGeometrieStufeGemeinde 2002	20020000248847	20020007749317
TeilobjektGeometrieStufeGemeinde 2003	20030000248847	20030012938661
TeilobjektGeometrieStufeGemeinde 2004	20040000248847	20040012938661
TeilobjektGeometrieStufeGemeinde 2005	20050000248847	20050016489257
TeilobjektGeometrieStufeGemeinde 2006	20060000248847	20060017246587
TeilobjektGeometrieStufeGemeinde 2007	20070000248847	20070017246587
TeilobjektGeometrieStufeGemeinde 2008	20080000248847	20080018407004
TeilobjektGeometrieStufeGemeinde 2009	20090000248847	20090018407004
TeilobjektGeometrieStufeGemeinde 2010	20100000014975	20100000015391
TeilobjektGeometrieStufeGemeinde 1994/1995/1996/1997	19940000014050	19970000014110
LebensabschnittGeometrieStufeGemeinde 1960a	19601000011192	19601000011193
LebensabschnittGeometrieStufeGemeinde 1960b	19601000010790	19701000013217
LebensabschnittGeometrieStufeGemeinde 1990	19901000000001	19901000003251
LebensabschnittGeometrieStufeGemeinde 1994	19941000000001	19941000003248
LebensabschnittGeometrieStufeGemeinde 1996	19961000000001	19961000003174
LebensabschnittGeometrieStufeGemeinde 1998	19981000000001	19981000003147
LebensabschnittGeometrieStufeGemeinde 2000	20001000248847	20001004677156
LebensabschnittGeometrieStufeGemeinde 2001	20011000248847	20011007749317
LebensabschnittGeometrieStufeGemeinde 2002	20021000248847	20021007749317
LebensabschnittGeometrieStufeGemeinde 2003	20031000248847	20031012938661
LebensabschnittGeometrieStufeGemeinde 2004	20041000248847	20041012938661
LebensabschnittGeometrieStufeGemeinde 2005	20051000248847	20051016489257
LebensabschnittGeometrieStufeGemeinde 2006	20061000248847	20061017246587
LebensabschnittGeometrieStufeGemeinde 2007	20071000248847	20071017246587
LebensabschnittGeometrieStufeGemeinde 2008	20081000248847	20081018407004
LebensabschnittGeometrieStufeGemeinde 2009	20091000248847	20091018407004
LebensabschnittGeometrieStufeGemeinde 2010	20101000014975	20101000015391
LebensabschnittGeometrieStufeGemeinde 1994/1995/1996/1997	19941000014050	19971000014110

Table C.2: Number intervals of the geometries of the municipality level. Part 1

Number intervals / Nummernbereiche Geometries on the municipality level / Geometrien Stufe Gemeinde		
Klasse oder Teilklasse	**Minimum**	**Maximum**
GZG_1990_StufeGemeinde	gzg90_ch_gde.shp:1	gzg90_ch_gde.shp:3251
GZG_1994_StufeGemeinde	gzg94_ch_gde.shp:1	gzg94_ch_gde.shp:3248
GZG_1996_StufeGemeinde	gzg96_ch_gde.shp:1	gzg96_ch_gde.shp:3174
GZG_1998_StufeGemeinde	gzg98_ch_gde.shp:1	gzg98_ch_gde.shp:3147
GG25_2000_StufeGemeinde	gg25_00_a_ch_gde.shp:248847	gg25_00_a_ch_gde.shp:4677156
GG25_2001_StufeGemeinde	gg25_01_a_ch_gde.shp:248847	gg25_01_a_ch_gde.shp:7749317
GG25_2002_StufeGemeinde	gg25_02_a_ch_gde.shp:248847	gg25_02_a_ch_gde.shp:7749317
GG25_2003_StufeGemeinde	gg25_03_a_ch_gde.shp:248847	gg25_03_a_ch_gde.shp:12938661
GG25_2004_StufeGemeinde	gg25_04_a_ch_gde.shp:248847	gg25_04_a_ch_gde.shp:12938661
GG25_2005_StufeGemeinde	gg25_05_a_ch_gde.shp:248847	gg25_05_a_ch_gde.shp:16489257
GG25_2006_StufeGemeinde	gg25_06_a_ch_gde.shp:248847	gg25_06_a_ch_gde.shp:17246587
GG25_2007_StufeGemeinde	gg25_07_a_ch_gde.shp:248847	gg25_07_a_ch_gde.shp:17246587
GG25_2008_StufeGemeinde	gg25_08_a_ch_gde.shp:248847	gg25_08_a_ch_gde.shp:18407004
GG25_2009_StufeGemeinde	gg25_09_a_ch_gde.shp:248847	gg25_09_a_ch_gde.shp:18407004
GeometrieNichtVorhanden	GeometrieFuerDiesesTeilobjektNicht Vorhanden	GeometrieFuerDiesesTeilobjektNicht Vorhanden

Table C.3: Number intervals of the geometries of the municipality level. Part 2

Number intervals / Nummernbereiche Geometries on the district level / Geometrien Stufe Bezirk		
Klasse oder Teilklasse	**Minimum**	**Maximum**
TeilobjektGeometrieStufeBezirk 1990	19900100000001	19900100002603
TeilobjektGeometrieStufeBezirk 1994	19940100000001	19940100002603
TeilobjektGeometrieStufeBezirk 1996	19960100000001	19960100002603
TeilobjektGeometrieStufeBezirk 1998	19980100000001	19980100002603
TeilobjektGeometrieStufeBezirk 2000	20000100000001	20000100002603
TeilobjektGeometrieStufeBezirk 2001	20010100000001	20010100002603
TeilobjektGeometrieStufeBezirk 2002	20020100000001	20020100002603
TeilobjektGeometrieStufeBezirk 2003	20030100000001	20030100002603
TeilobjektGeometrieStufeBezirk 2004	20040100000001	20040100002603
TeilobjektGeometrieStufeBezirk 2005	20050100000001	20050100002603
TeilobjektGeometrieStufeBezirk 2006	20060100000001	20060100002603
TeilobjektGeometrieStufeBezirk 2007	20070100000001	20070100002603
TeilobjektGeometrieStufeBezirk 2008	20080100000001	20080100002603
TeilobjektGeometrieStufeBezirk 2009	20090100000001	20090100002603
TeilobjektGeometrieStufeBezirk 2010	20100100000241	20100100000250
LebensabschnittGeometrieStufeBezirk 1990	19901100000001	19901100002603
LebensabschnittGeometrieStufeBezirk 1994	19941100000001	19941100002603
LebensabschnittGeometrieStufeBezirk 1996	19961100000001	19961100002603
LebensabschnittGeometrieStufeBezirk 1998	19981100000001	19981100002603
LebensabschnittGeometrieStufeBezirk 2000	20001100000001	20001100002603
LebensabschnittGeometrieStufeBezirk 2001	20011100000001	20011100002603
LebensabschnittGeometrieStufeBezirk 2002	20021100000001	20021100002603
LebensabschnittGeometrieStufeBezirk 2003	20031100000001	20031100002603
LebensabschnittGeometrieStufeBezirk 2004	20041100000001	20041100002603
LebensabschnittGeometrieStufeBezirk 2005	20051100000001	20051100002603
LebensabschnittGeometrieStufeBezirk 2006	20061100000001	20061100002603
LebensabschnittGeometrieStufeBezirk 2007	20071100000001	20071100002603
LebensabschnittGeometrieStufeBezirk 2008	20081100000001	20081100002603
LebensabschnittGeometrieStufeBezirk 2009	20091100000001	20091100002603
LebensabschnittGeometrieStufeBezirk 2010	20101100000241	20101100000250
GZG_1990_StufeBezirk	gzg90_ch_bez.shp:100	gzg90_ch_bez.shp:2603
GZG_1994_StufeBezirk	gzg94_ch_bez.shp:100	gzg94_ch_bez.shp:2603
GZG_1996_StufeBezirk	gzg96_ch_bez.shp:100	gzg96_ch_bez.shp:2603
GZG_1998_StufeBezirk	gzg98_ch_bez.shp:100	gzg98_ch_bez.shp:2603
GG25_2000_StufeBezirk	gg25_00_a_ch_bez.shp:100	gg25_00_a_ch_bez.shp:2603
GG25_2001_StufeBezirk	gg25_01_a_ch_bez.shp:100	gg25_01_a_ch_bez.shp:2603
GG25_2002_StufeBezirk	gg25_02_a_ch_bez.shp:100	gg25_02_a_ch_bez.shp:2603
GG25_2003_StufeBezirk	gg25_03_a_ch_bez.shp:100	gg25_03_a_ch_bez.shp:2603
GG25_2004_StufeBezirk	gg25_04_a_ch_bez.shp:100	gg25_04_a_ch_bez.shp:2603
GG25_2005_StufeBezirk	gg25_05_a_ch_bez.shp:100	gg25_05_a_ch_bez.shp:2603
GG25_2006_StufeBezirk	gg25_06_a_ch_bez.shp:100	gg25_06_a_ch_bez.shp:2603
GG25_2007_StufeBezirk	gg25_07_a_ch_bez.shp:100	gg25_07_a_ch_bez.shp:2603
GG25_2008_StufeBezirk	gg25_08_a_ch_bez.shp:100	gg25_08_a_ch_bez.shp:2603
GG25_2009_StufeBezirk	gg25_09_a_ch_bez.shp:100	gg25_09_a_ch_bez.shp:2603
GeometrieNichtVorhanden	GeometrieFuerDiesesTeilobjekt NichtVorhanden	GeometrieFuerDiesesTeilobjekt NichtVorhanden

Table C.4: Number intervals of the geometries of the district level

Number intervals / Nummernbereiche Geometries of cantons / Geometrien Kanton		
Klasse oder Teilklasse	**Minimum**	**Maximum**
TeilobjektGeometrieKanton 1990	19900200000001	19900200000026
TeilobjektGeometrieKanton 1994	19940200000001	19940200000026
TeilobjektGeometrieKanton 1996	19960200000001	19960200000026
TeilobjektGeometrieKanton 1998	19980200000001	19980200000026
TeilobjektGeometrieKanton 2000	20000200000001	20000200000026
TeilobjektGeometrieKanton 2001	20010200000001	20010200000026
TeilobjektGeometrieKanton 2002	20020200000001	20020200000026
TeilobjektGeometrieKanton 2003	20030200000001	20030200000026
TeilobjektGeometrieKanton 2004	20040200000001	20040200000026
TeilobjektGeometrieKanton 2005	20050200000001	20050200000026
TeilobjektGeometrieKanton 2006	20060200000001	20060200000026
TeilobjektGeometrieKanton 2007	20070200000001	20070200000026
TeilobjektGeometrieKanton 2008	20080200000001	20080200000026
TeilobjektGeometrieKanton 2009	20090200000001	20090200000026
LebensabschnittGeometrieKanton 1990	19901200000001	19901200000026
LebensabschnittGeometrieKanton 1994	19941200000001	19941200000026
LebensabschnittGeometrieKanton 1996	19961200000001	19961200000026
LebensabschnittGeometrieKanton 1998	19981200000001	19981200000026
LebensabschnittGeometrieKanton 2000	20001200000001	20001200000026
LebensabschnittGeometrieKanton 2001	20011200000001	20011200000026
LebensabschnittGeometrieKanton 2002	20021200000001	20021200000026
LebensabschnittGeometrieKanton 2003	20031200000001	20031200000026
LebensabschnittGeometrieKanton 2004	20041200000001	20041200000026
LebensabschnittGeometrieKanton 2005	20051200000001	20051200000026
LebensabschnittGeometrieKanton 2006	20061200000001	20061200000026
LebensabschnittGeometrieKanton 2007	20071200000001	20071200000026
LebensabschnittGeometrieKanton 2008	20081200000001	20081200000026
LebensabschnittGeometrieKanton 2009	20091200000001	20091200000026
GZG_1990_Kanton	gzg90_ch_kt.shp:1	gzg90_ch_kt.shp:26
GZG_1994_Kanton	gzg94_ch_kt.shp:1	gzg94_ch_kt.shp:26
GZG_1996_Kanton	gzg96_ch_kt.shp:1	gzg96_ch_kt.shp:26
GZG_1998_Kanton	gzg98_ch_kt.shp:1	gzg98_ch_kt.shp:26
GG25_2000_Kanton	gg25_00_a_ch_kt.shp:1	gg25_00_a_ch_kt.shp:26
GG25_2001_Kanton	gg25_01_a_ch_kt.shp:1	gg25_01_a_ch_kt.shp:26
GG25_2002_Kanton	gg25_02_a_ch_kt.shp:1	gg25_02_a_ch_kt.shp:26
GG25_2003_Kanton	gg25_03_a_ch_kt.shp:1	gg25_03_a_ch_kt.shp:26
GG25_2004_Kanton	gg25_04_a_ch_kt.shp:1	gg25_04_a_ch_kt.shp:26
GG25_2005_Kanton	gg25_05_a_ch_kt.shp:1	gg25_05_a_ch_kt.shp:26
GG25_2006_Kanton	gg25_06_a_ch_kt.shp:1	gg25_06_a_ch_kt.shp:26
GG25_2007_Kanton	gg25_07_a_ch_kt.shp:1	gg25_07_a_ch_kt.shp:26
GG25_2008_Kanton	gg25_08_a_ch_kt.shp:1	gg25_08_a_ch_kt.shp:26
GG25_2009_Kanton	gg25_09_a_ch_kt.shp:1	gg25_09_a_ch_kt.shp:26

Table C.5: Number intervals of the geometries of the cantons

Appendix D Abbreviations

D.1 Abbreviations 3-P

Abbreviation	Meaning	German translation
3DM	Three-Domain Model	
A-Box	assertional box	
ACT	Australian Capital Territory	Australisches Hauptstadtterritorium
AU	administrative unit	Verwaltungseinheit
BFO	Basic Formal Ontology	
CR	Canton Register	Kantonsverzeichnis
DL	description logic	Beschreibungslogik
DNL	Data Center Nature and Landscape	Datenzentrum Natur und Landschaft
DOLCE	Descriptive Ontology for Linguistic and Cognitive Engineering	
ESTDM	Event-based Spatio Temporal Data Model	
ESTI	Framework for Evolutive SpatioTemporal Information	
EU	European Union	Europäische Union
FEM	Feature Evolution Model	
FOEN	Federal Office for the Environment	Bundesamt für Umwelt (BAFU)
FOT	Federal Office of Topography	Bundesamt für Landestopografie (swisstopo)
FSO	Federal Statistical Office	Bundesamt für Statistik (BFS)
GBHGIS	Great Britain Historical GIS	
GEM	Geospatial Event Model	
GIS	geographic information system	geografisches Informationssystem
GU	geographic unit	geografische Einheit
HDR	Historicized District Register	Historisiertes Bezirksverzeichnis
HMR	Historicized Municipality Register in the stricter sense	Historisiertes Gemeindeverzeichnis im engeren Sinne
HMR+	Historicized Municipality Register in the broader sense (consists of HMR, HDR, CR)	Historisiertes Gemeindeverzeichnis im weiteren Sinne (besteht aus HMR, HDR, CR)
KR	knowledge representation	Wissensrepräsentation
OWL	Web Ontology Language	
PL	period of life	Lebensabschnitt
PLAU	period of life administrative unit	Lebensabschnitt Verwaltungseinheit
PLG	period of life geometry	Lebensabschnitt Geometrie
PO	partial object	Teilobjekt
POAU	partial object administrative unit	Teilobjekt Verwaltungseinheit
POG	partial object geometry	Teilobjekt Geometrie

Table D.1: Abbreviations 3-P

D.2 Abbreviations R-X

Abbreviation	Meaning	German translation
RDF	Resource Description Framework	
RDFS	RDF Vocabulary Description Language (RDF Schema)	
RQ	research question	Forschungsfrage
SAPO	Suomen Ajallinen PaikkaOntologia (Finnish Spatio-Temporal Ontology)	
SDMS	Spatial Data Management System	
SONADUS	Spatiotemporal Ontology for the Administrative Units of Switzerland	Raumzeitliche Ontologie für die Verwaltungseinheiten der Schweiz
SONADUSi	SONADUS with inferred axioms	SONADUS mit abgeleiteten Axiomen
SONADUS-KB	SONADUS knowledge base	SONADUS Wissensdatenbank
SOVA	Simple Ontology Visualization API	
SPARQL	SPARQL Protocol and RDF Query Language	
SPIRIT	Spatially-Aware Information Retrieval on the Internet	
SQL	Structured Query Language	
SUMO	Suggested Upper Merged Ontology	
T-Box	terminological box	
Turtle	Terse RDF Triple Language	
UK	United Kingdom	Vereinigtes Königreich
UML	Unified Modeling Language	
URI	Uniform Resource Identifier	
URL	Uniform Resource Locator	
UZH	University of Zurich	Universität Zürich
W3C	World Wide Web Consortium	
WO	whole object	Ganzes Objekt
WSL	Swiss Federal Institute for Forest, Snow and Landscape Research	Eidgenössische Forschungsanstalt für Wald, Schnee und Landschaft
XML	Extensible Markup Language	

Table D.2: Abbreviations R-X

Appendix E Personal declaration

I hereby declare that the submitted thesis is the result of my own independent work. All external sources are explicitly acknowledged in the thesis.

Birmensdorf / April 26, 2011

Felix Gantner