

Bauhaus-Universität Weimar
Faculty of Media
Degree Programme Computer Science and Media

Visualizing and Editing the Evolution of Countries in Time and Space

Master's Thesis

Marcus Kossatz
b. 21.08.1989 in Spremberg, Germany

Matriculation Number 90487

1. First Referee: Univ.-Prof. Dr.-Ing. habil. Volker Rodehorst
2. Second Referee: Prof. Dr. rer. nat. Sven Bertel

Submission date: 6. June 2016

Contents

1	Introduction	10
1.1	Motivation	11
1.2	Research Questions	11
1.3	Overview	11
2	Basics	12
2.1	Historical Geographic Information Systems	12
2.1.1	History	13
2.1.2	Geography	14
2.1.3	Information	15
2.1.4	System	16
2.1.5	Motivation	16
2.1.6	Components	17
2.1.7	Applications	17
2.1.8	Data Sources	19
2.2	Time and Space	21
2.2.1	Model of Geographical Space	21
2.2.2	Presentation of Geographic Space	25
2.2.3	Model of Historical Time	26
2.2.4	Presentation of Historical Time	28
2.3	Spatio-Temporal Data Models	29
2.3.1	Snapshot Model	30
2.3.2	Simple Time-Stamping	30

2.3.3	Event-Based Spatio-Temporal Data Model	31
2.3.4	Three-Domain Model	32
2.3.5	History Graph Model	33
2.4	Database Management Systems	34
2.4.1	Relational Database Management Systems.	35
2.4.2	Object-Oriented Database Management Systems	35
2.4.3	Object-Relational Database Management Systems	36
2.4.4	Spatio-Temporal Databases	37
2.5	HistoGlobe	38
3	Development	39
3.1	Hivent Model.	40
3.1.1	Elements	41
3.1.2	Preconditions.	42
3.1.3	Hivent Operations	43
3.1.4	HistoGraph	45
3.2	Editing Hivent Data	47
3.2.1	Edit Operations	47
3.2.2	Edit Workflow	48
3.2.3	Retrospective Updates	51
3.2.4	Backward Operations	59
3.3	User Interface Design Process	60
3.3.1	Paper Prototype.	61
3.3.2	Mockup Prototype.	62
3.3.3	Web-based prototype	63
3.4	Application	66
3.4.1	System Architecture.	66
3.4.2	Server-Side Application	67
3.4.3	Client-Side Application	69
4	Evaluation	78

5	Uncertainty	83
5.1	Definition of a Country	83
5.1.1	Special Cases	84
5.1.2	Declaratory vs. Constitutive Theory	86
5.2	Types of Uncertainty.	87
5.3	Solution Approaches	90
5.3.1	Extension of the Edit Mode	92
5.3.2	Extension of the Data Model.	98
6	Summary	100
6.1	Results	100
6.2	Problems	100
6.3	Future Work	100

List of Figures

2.1	signs, data, information and knowledge	15
2.2	Napoleons Moscow Campaign ¹	18
2.3	Semi-automatic extraction of a border from a map of the Roman Empire ²	21
2.4	The geoid model, differences are exaggerated, [Bol08, Fig. 3-6, p. 75]	22
2.5	geographic coordinates using latitude and longitude	22
2.6	Comparison of the raster and the vector model	23
2.7	The basic geometric primitives point, polyline and polygon	24
2.8	Different properties of polygons	24
2.9	Comparison of equivalent and conformal map projections	26
2.10	Robinson projection ³	26
2.11	Comparison between a linear, a logarithmic and an irregular timeline	29
2.12	The Snapshot Model by [Lan88]	30
2.13	The Simple Time-Stamping method by [HW90]	31
2.14	The History Graph model	33
2.15	Types of changes in the History Graph model	34
3.1	Human Centered Design process with five project phases	39
3.2	Relevant models for an information system	40
3.3	Three event types that change Areas, resulting in two different area states	41
3.4	The initial state of the world map at time point t_0	43

3.5	The concept of the HistoGraph at the example of the history of Germany since 1945	45
3.6	Example for a simple conflict due to a retrospective update	52
3.7	Conflicts after a retrospective incorporation	54
3.8	Conflicts after a retrospective secession	55
3.9	Conflicts after a retrospective unification	57
3.10	Conflicts after a retrospective separation	58
3.11	The two iterations of the paper prototype for the Edit Mode	61
3.12	Two iteration stages of the mockup prototype for the Edit Mode	62
3.14	Initial state of the Browsing Mode	64
3.15	Initial state of the Edit Mode	64
3.16	1) Select old Areas	64
3.17	2) Set a new territory	64
3.18	3) Set a new name	65
3.19	4) Add the Operation to a Hivent	65
3.20	4) Create a new Hivent	65
3.21	The final state with Scotland	65
3.22	The system architecture of HistoGlobe	66
3.23	The Hivent Database Model	68
3.24	Visualization of the German Reunification in the Hivent Database Model	69
3.25	Class Diagram of HistoGlobe	70
3.26	Detecting the next Hivent that happens in the HiventController	71
3.27	The Within-Tree (left) for the set of polygons (right)	74
3.28	The resulting labels on the map in Europe 2016	76
5.1	The Declaratory Theory (left) and the Constitutive Theory (right) of Statehood	87
5.2	The difference between accuracy and precision	88
5.3	Newly designed and extended buttons for edit operations.	92

5.4	Three different methods to visualize uncertain courses of a border	93
5.5	Drawing historical borders instead of full areas and defining a level of certainty.	94
5.6	Getting suggestions for the name name from Wikipedia.	95
5.7	Defining a special status or relationship to a territory.	96
5.8	Creating a new Hivent and adding the newly created historical change.	97
5.9	New edit operation: Recognition – sets up the recognition of one area to another.	97
5.10	Changing the language in the user interface.	98
5.11	The new data model to support the developed approaches regarding uncertainty	99

List of Tables

2.1	differences between history and geography [KH08, pp. 2-4]	15
2.2	Temporal relations of time spans, based on [All84]	28
3.1	The five Hivent Operations	44
3.2	Historical events in German state history since 1945	46
3.3	The six Edit Operations	47
3.4	The requirements of each step for the Edit Operations	49
3.5	Translation from Edit Operations to Hivent Operations	51
3.6	All possible conflicts on retrospective updates regarding their resolvability	59

Selbstständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbstständig und nur unter Zuhilfenahme der angegebenen direkten und indirekten Quellen erstellt habe. Diese Arbeit wurde in gleicher oder ähnlicher Form noch bei keinem anderen Prüfer als Prüfungsleistung eingereicht und ist noch nicht veröffentlicht.

Statement of Authorship

Hereby I declare that I completed this Master's Thesis on my own and that information which has been directly or indirectly taken from other sources has been noted as such. Neither this, nor a similar work, has been published or presented to an examination committee.

Weimar, 6. June 2016

Marcus Kossatz

Acknowledgements

The work on this Master's Thesis began in October 2015 and ended in June 2016. I want to thank my supervisors Prof. Volker Rodehorst and Prof. Sven Bertel for the continuous advices, ideas and the motivation to work on this topic of my choice.

The majority of the time of this thesis (October-December 2015 and February-May 2016) I have spent working at the *Scholars' Lab* of *University of Virginia*. I was provided with an excellent workspace, very cooperative people and delicious oatmeal raising cookies that all contributed to a great work atmosphere in Charlottesville, Virginia. I especially want to thank all the members of the *Scholars' Lab* that were always eager to solve my problems regarding the concept and the implementation of this thesis, espacially Ammon, Eric, Scott, Purdom, Jeremy, Chris, Wayne, Ronda, Laura, Becca, Shane and Veronica. The perspective of humanity scholars on my project were very valuable for me personally and for my work. Thank you!

Since this thesis was an advancement of HistoGlobe, the project I have founded in 2010, I want to thank everybody that has ever been involved in this project. In the previous 6 years I have learned a lot about software engineering, history and geography, project management and entrepreneurship – and how to (not) make a project successful that almost everybody likes. Special thanks goes to Frej which whom I developed the first prototype in 2011. From 2012 on, my coworkers, fellow students and friends Simon and Felix deserve great credit for everything they have done to HistoGlobe and me. Without the software they have developed until 2014 the result of this thesis would be much different. I also want to thank the other developers that have spent their free time, ideas and coding skills on the HistoGlobe project: Sascha, Sebastian, Lene, Max, Chris and Sebastian. Thank you Christian for always burning for HistoGlobe and trying to make the project profitable. Im also thankful for Carola and Katrin who have spent significant free time on HistoGlobe in 2013. Last but not least a huge Thank You to my friend and colleague Tobias for your high efforts for the HistoGlobe project, great concepts and stunning designs. I have learned from you how valuable good design is.

Neither HistoGlobe nor this thesis, nor both of my academic degrees would be possible without *Bauhaus-Universität Weimar* and all the associated professors and academic staff at this great institution. Thank you for the support throughout the previous seven years. The history and the values of the Bauhaus have always inspired me and are a big part of how I think, develop and create. It has been a great time in Weimar.

Finally, I want to thank my family that has always been there for me in the first 26 years of my life. May it be my studies, my three foreign exchanges or my entrepreneurial idea – my parents, my brother, my sister and the rest of my family have always supported me. That feeling is invaluable.

Chapter 1

Introduction

*La Géographie n'est autre chose que l'Histoire dans l'espace,
de même que l'Histoire est la Géographie dans le temps.*

*Geography is nothing but History in space,
the same way as History is Geography over time.*

– Élisée Reclus: “L’Homme et la Terre” (1908)

long and aspiring text about human life, the question of where, when, what and how?

time and space are everywhere, highly related to our lives and objects we perceive time personal time points of major life events, ... -*i* events, can trigger other events periods of studying, working, ... -*i* collection of events with similar characteristics world: every major issue has a time scale climate change (decades) climate tipping points (years) climate tipping points (years) economic meltdown (months) infectious diseases (weeks) disasters (days) -*i* time not easy to scale and to grasp space location of major life events (static) travel routes (dynamic) not always easy to grasp (exact location of monument is simple, but exact location of problematic area with fremdenfeindlichen hintergrund or historic countries are hard to set) event locations are sometimes not related to their consequences (e.g. Conferences of Tehran or Casablanca (1943) discussed how to deal with Germany after planned victory in WWII) motivation for spatio-temporal queries exploration of German history using historical maps of 1800, 1850, 1900 and 1950 each map has both temporal and spatial information in it but how to tell a story with that? more realistically, maps from 1871, 1919, 1933, 1945 and 1949, because of major events (founding of German Reich, end of WWI, beginning of Nazi dictatorship, end of WWII, founding of two German nations -*i* for one country might be suitable exploration of European or history would need a world map for each year how to see what has changed? -*i* inefficient how to

know what is important? is that a reasonable way of storing information if one information set is with a high probability almost the same as the time point before? - \downarrow redundancy key problem model of historic maps at time points (- \downarrow snapshot model) given information at time point t1 and t2 How to know the status at time point t1 ; tm ; t2? - \downarrow It is impossible solution: away from snapshot based modeling of history to change-based modeling initial state t_i , changes at point t1 and t2 How to know the status at time point t1 ; tm ; t2? - \downarrow it is $t_i + \text{changes at } t_1 = \downarrow$ definition of each time point in history

research object of this thesis change over time of space of countries history of countries, their names and their borders and their relationships to each other visualize these changes and edit them (interface) Web-based historical geographic information system (WHGIS)

research questions of the thesis How to design and implement WHGIS? How to create an interface not just to explore historical changes? How to deal with uncertainty and fuzziness in history? Can researchers actually interact with such a system? How to design an interface that matches the mental model of a DH user of editing changes over time?

study of existing approaches, techniques and projects GIS: acquisition, management, analysis and presentation of spatial information handling of the spatial domain: extension to HGIS some systems allow presentations, but have very difficult interfaces no system that allows editing historical borders in time and space

domain: evolution of countries in time and space - \downarrow names and territories - \downarrow modelling, visualizing and !!! editing !!!

1.1 Motivation

research questions - \downarrow HGIS \downarrow - development of system historians / ME geographers (+) open source, direct manipulation, easy sharing and collaboration

1.2 Research Questions

1.3 Overview

Chapter 2

Basics

This chapter will lay the theoretical foundation of this Master's Thesis and will embed it into the context of current research. The title of this work is:

Visualizing and Editing the Evolution of Countries in Time and Space

It includes the domain (*history of countries*) and the system to acquire, model, manage and visualize data of the domain: *Historical Geographic Information Systems* (HGIS).

The first section of this chapter will define HGIS and related terms. Afterwards, concepts to model time and space in an information system are introduced. Data sources suitable for input into an HGIS are listed in the next section, followed by techniques to manage and analyse the data.

extension of Hivent model to actors

2.1 Historical Geographic Information Systems

"All human actions takes and makes place. The past is the set of places made by human action. History is a map of these places. The past thus exists not in time but in space."

– Philip J. Ethington in [Eth07, précis]

An Historical Geographic Information System helps to answer research questions about how geographical phenomena have developed over time. To understand how it works, it is important to understand

the four parts of the word: The research fields *history* and *geography* and the concepts of *information* and *systems*.

2.1.1 History

History is “an ideal field for thinking long and hard about important questions” [AHA]. The Greek word *Ιστορία / historia*, meaning “finding out, learning through research, narration of what is learned”, is the origin¹ and it signifies the two main modern usage forms of the term: To research about and learning something and to tell a story. There are many different definitions of the word *history*². The main goal of history is to study processes in the past to understand the situation in the present and make reasonable decisions for the future. The American Historical Association has developed the “five C’s of historical thinking [that] together describe the shared foundations of [the] discipline” [AHA]:

Change over time The lives of people, their languages and their cultures are continuously changing.

Describing these historical changes, triggered by historical events happened in the past, is a major goal of history. Snapshots in the form of historical maps or historical photography are used to tackle this task.

Context is an important element of historical thinking. The goal is to travel back in time to the moment of the event and recreate the world based on primary sources. The understanding of the historical context is crucial for the understanding of the event.

Causality The overall goal of each science is to answer the *why*-question concerning an event or a process. For historians that means to reasonably explain an historical event or process based on evidence. The problem is that history is not a science that can alter experimental conditions to extract new information, in a way that e.g. experiments in physics work. Historians have to focus on the interpretation of primary sources, which inherently yields multiple explanations for a single event.

Contingency is a derived aspect from this problem. Each event has a whole network of prior conditions, because the world is highly interconnected. A slight change in one prior condition could have led to a completely different outcome of the event and a different state of the world.

Complexity The intrinsic human need for order conflicts with the complexity of history and their events and processes, because of its contingency. It is questionable if all details about events in the world are scientifically explainable.

¹ *History*, Dictionary.com, based on Random House Dictionary, 2015, URL: <http://dictionary.reference.com/browse/history>, last access: 23.10.2015

² *History*, Merriam Webster – an Encyclopædia Britannica Company, URL: <http://www.merriam-webster.com/dictionary/history>, last access: 23.10.2015

Historical research is conducted by studying and interpreting primary sources, such as written documents, verbal texts, speeches, photographs, audio, video or historical maps. This signifies that most historical research is qualitative. The main organization principle in history is periodization: classifying events and processes to describe broader long-term changes and to explain complex phenomena [KH08, pp.4-7]. A special focus in this thesis is laid on historical maps as primary source to extract spatial information.

2.1.2 Geography

The term “geography” comes from Greek *γεωγραφία / geographia*, literally “describing the earth.”³ It is a science that studies the interplay between the landscapes and environments of the Earth (*physical geography*) on the one hand and the people, their cultures, societies and economies (*human geography*) on the other. That means geography is an interdisciplinary field between natural and social sciences [RGS].

Geographical research aims to understand where things are found, why they are there and how they developed over time. It focuses on the interconnectivity between elements of physical and human geography, which gets expressed in Tobler’s First Law of Geography: “Everything is related to everything else, but near things are more related than distant things.”⁴

Geographers use different technology and techniques to analyze geographic processes and to answer their research questions. The oldest and most important among those are maps. A map is a graphical expression of something that is not tangible: a part of the real world. A map shows the physical, environmental, political, economical or social properties of the Earth in order for the user of the map to get the most relevant information for his task, may it be orientation, learning or teaching. The “art and science of making maps” is the field of *cartography*⁵. Since maps visualize a model, they have a natural constraint: “No map can perfectly replicate the real world, since it inevitably generalizes, abstracts and approximates the complexity of the reality” [KH08, p. 181].

Comparison between geography and history Both research fields utilize maps for answering their research questions, which is the main commonality for the work of this thesis. However, the nature of both fields are also very different, illustrated in table 2.1.

Whereas geography answers the questions *where?*, history focuses on *when?* – but the ultimate goal

³ *Geography*, Dictionary.com, based on Random House Dictionary, 2015, URL: <http://dictionary.reference.com/browse/geography>, last access: 23.10.2015

⁴ “A computer movie simulating urban growth in the Detroit region”, Waldo Tobler, 1970 *Economic Geography*, 46(2): 234-240.

⁵ *History of maps and cartography*, James S. Aber, URL: http://academic.emporia.edu/aberjame/map/h_map/h_map.htm, last access: 24.10.2015

geography	difference	history
where	dimension	when
exact, statistical	character	complex, fuzzy
mainly quantitative	research	qualitative
spatial proximity of conditions	causal explanation	temporal sequence of events
spatial differentiation	explanation	temporal differentiation
clustering	ordering	periodization
mostly visual (maps)	expression	mostly verbal (texts)
high (GIS)	digitalization potential	low (digital humanities)

Table 2.1: differences between history and geography [KH08, pp. 2-4]

for both sciences is to answer the question *why?*

2.1.3 Information

The terms “signs”, “data”, “information” and “knowledge” are sometimes used interchangeably and there is no coherent definition for any of them. However, all describe different concepts. This explanation seen in figure 2.1 is based on the work of [Dra].

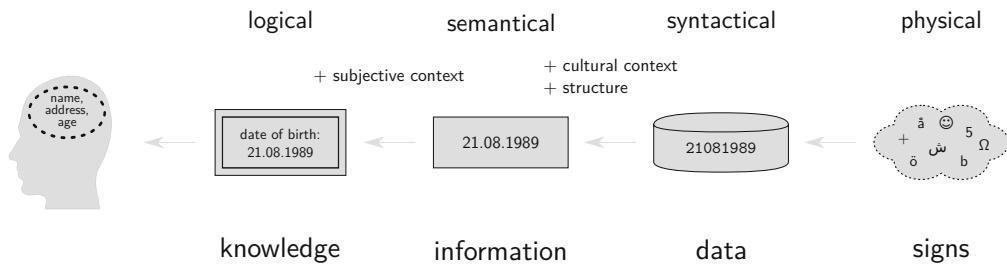


Figure 2.1: signs, data, information and knowledge

A *sign* is the physical representation of something in the real world. Since the real world is continuous, literally anything can be seen as a sign, so there are uncountably infinitely many different signs. *Data* is a subset of all possible signs and represents the syntactical level of what an information system deals with. Data itself does not have any meaning, but as soon as it is organized, it becomes *information*. However, information is sensitive to its cultural context. The string 14.07.1789 is useful and understandable for people in countries that use the date format DD.MM.YYYY. However, for people in Belize and the USA, that use the format MM.DD.YYYY, this might just be a random string of numbers without any meaning, and therefore no information – although it is the same data. If information is

visualized to and understood by a human and it can be integrated into his or her larger subjective context, it is *knowledge* [Nak]. The goal of a visualization is to present as much information as possible in a way that it can be transformed into knowledge by the viewer.

2.1.4 System

A *system* is an organized structure containing *elements* or *components* that are directly or indirectly related to and *interconnected* with each other. The elements and their relations form the whole of the structure. The surrounding of the system is its *environment*. There is an *internal state* at any point of the system's existence. This state only changes when it gets influenced by stimuli of its environment. *Emergent properties* characterize a system. They are independent from properties of the element of the system, e.g. water is liquid at room temperature, but the elements it consists of, hydrogen and oxygen, are a gas. Each system is both part of a larger system and can be decomposed into subsystems. Therefore, systems form a hierarchy.

A system has defined spatial and temporal boundaries. There are two types: *open systems* allow exchange of energy or information with their environment, whereas idealized *close systems* naturally do not interact with and are not influenced by its environment. Based on the black box principle the inner working of a closed system can not be seen from the outside [Bus].

2.1.5 Motivation

An *information system* (IS) is an application that is dealing with the acquisition, management, analysis and presentation of information. It is the unity of all its components and their interaction with each other [Zwa]. If the majority of the information in a system has a spatial relation to the Earth, its surface, its lithosphere, atmosphere or the social or economical structure of its habitation, it is a *geographic information system* (GIS). The data objects in the system are called *geo-objects* [Bol08]. If the information additionally has a temporal dimension, e.g. via time stamps or time spans, which enable to trace developments of geo-objects, it becomes an *Historical Geographic Information System* [GG14] or alternatively *Spatio-Temporal Information Systems* (STIS) [PTKT04].

HGIS react on the spatial turn of history: the integration of geographic methods in historical research. It aims to discover the power of cartographic representation: “The spatial turn in the humanities must [...] understand the role of space in human events” [BCH10]. At the same time, they are the product of the temporal run in GIS: the coexistence of space (where things are) and time (what has changed over time) [Sol14, p. 45]. With HGIS it is possible to analyze how “spatial patterns change over time in order to better understand large-scale Earth processes” [Peu99]. Since “the world never stands still”, but “the retention of information relating to past events [is] an important element of human representation of the world”, the dimension of time has to be integrated into a GIS [Peu99].

HGIS are rather recent tools and used mostly in *Digital Humanities* as a digital tool to answer research questions in the traditional fields of humanities: “situating history in its geographical context and using geographic information to illuminate the past” [KH08, p. 3]. Some interesting research questions that could be answered using HGIS could be:

- Did the European Union help to bring peace on the European continent? *(political)*
- Is there a coherence between life expectancy and fertility rate? *(social)*
- What is the effect of global warming on the melting of glaciers? *(physical)*
- What was the effect of Bismarck’s foreign policy on peace in Europe? *(historical)*

Or on a more abstract level: Where and When has something changed and why did it change?

2.1.6 Components

Information systems in general are based on a data model — HGIS in particular on a *spatio-temporal data model*, introduced in section 2.3. On top of that, there are different components. One way to classify them is using the four-component model:

1. **Input:** Primary acquisition of spatio-temporal data, i.e. historical events, historical and current countries and their territories.
2. **Management:** Physical storage and logical management of the data in a spatio-temporal database, using a structure that fits the spatio-temporal data model.
3. **Analysis:** Gaining spatio-temporal information by cleaning, transforming or combining the data in database.
4. **Presentation:** Visualization of information on different displays, e.g. a map and a timeline, transforming information into spatio-temporal knowledge.

2.1.7 Applications

“Today, operational temporal GIS does not exist”. This quote summarizes the state of the art in this field. The main reasons are “the complexity of integrating space and time and the lack of standards” [Raz12, p. 5].

However, there are numerous project that use HGIS for one specific research question. A large collection them can be found in [KH08] and [GG14]. A famous visualization combining time and space Napoleons Moscow Campaign by Charles Minard from 1869 (see figure 2.2). The “best statistical

graphic ever drawn”⁶ shows the number of men in Napoleon’s 1812 Russian campaign army, their movements, as well as the temperature they encountered on the return path [KH08, pp. 188-191].

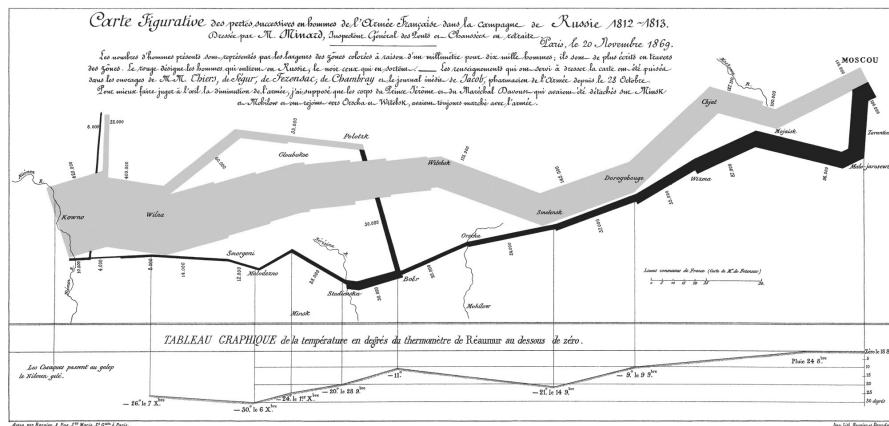


Figure 2.2: Napoleons Moscow Campaign⁷

The Great Britain Historical GIS Project (GBHGIS)⁸ combines statistical data with territorial units of the United Kingdom, e.g. to analyze net migration in the districts in UK. The data is collected by the British Ordnance Survey, who automatically detect spatial changes to the geography and land use of the United Kingdom using aerial photography⁹.

The National Historical Geographic Information System (NHGIS)¹⁰ provides the digital boundaries of the United States of America and census data for each year since 1790. While the data in the system is extensive, the interface to analyze and use the data is very frustrating to use: A tutorial is necessary to go through the selection process. To download data, a user has to register, receive an email with a link to download a compressed file which has to be decompressed and then loaded into a GIS software to be visualized.

HGIS are not widely accepted in the humanities. One reason is the nature of the qualitative historical research: historic sources are subjective and biased, their content may be fuzzy and they are definitely incomplete. So the knowledge that can be extracted from a source bears the integral problem of *uncertainty*. Information systems on the other hand have a logical architecture and, to be as precise and accurate as possible. Analysis is based on mathematical functions – an information system is

⁶ *The Visual Display of Quantitative Information* (p. 40), Edward R. Tufte, 2001

⁷ *Minard.png* Charles Minard, 1869, URL: <https://commons.wikimedia.org/wiki/File:Minard.png>, last access: 03.11.2015,

⁸ Great Britain Historical Geographical Information System (GBHGIS), Ian Gregory & Humphrey R. Southall, University of Portsmouth, since 1994, URL: <http://www.port.ac.uk/research/gbhgis/>, last access: 02.11.2015

⁹ British Ordnance Survey, URL: <https://www.ordnancesurvey.co.uk/education-research/research/automatic-change-detection.html>, last access: 02.11.2015

¹⁰ Welcome to NHGIS, Minnesota Population Center, University of Minnesota, since 2007, URL: , last access: 02.11.2015

quantitative in its entire nature [KH08, p. 2].

2.1.8 Data Sources

This HGIS needs data about historical countries, their names and borders and historical events that lead to historical changes of these countries. There are a lot of free and open sources for geographic data about the current countries, their names and borders. One of the most exhaustive collections of geographic data in public domain is hosted by Natural Earth¹¹. There is physical data (e.g. coastlines, rivers, or glacier areas) and cultural data (e.g. political borders, cities, roads, airports or timezones). OpenStreetMap also opens its database to the public¹².

However, data about historical countries and events are not as straightforward to acquire, because of the mostly qualitative nature of historical research (see section 2.1.1). The most exhaustive free and open source of historical is the *Wikipedia* and their article categories, e.g. armistices or treaties¹³. All sorts of historical events can be found, even translated into different languages. Some information is structured in information boxes, e.g. some historical treaties have a name, an image, a location, a signature and an effect date, an overview about treaty conditions and signatories. Particularly interesting for this thesis are articles about historical countries¹⁴, because they contain the name of the country and meta information, e.g. their historical successors and predecessors. Building an open-source Historical Geographic Information System on the basis of Wikipedia would be a huge project with significant impact on the world of free and open education — however, it would also be a big challenge: Wikipedia is incomplete, not all historical countries and events necessary to model the history of the world are available. It is also inconsistent, because not all articles about historical countries and events are structured, especially not to those who actually have an influence on a territorial change of a country, e.g. a border agreement. Retrieving, parsing and processing this information is a big challenge. Also the problem of accuracy and quality of information in the Wikipedia due to their open source nature has to be considered. Overall, using the Wikipedia as a data source for this thesis is not feasible, but is subject to further research.

Historical maps The most problematic data to acquire is about the territories and borders of historical countries. There is no primary data source for that, so the only way to retrieve a border is to extract it from an historical map.

They also can be found on Wikipedia, or in historical map collections, e.g. *OldMapsOnline*. The project

¹¹ *Natural Earth*, URL: <http://www.naturalearthdata.com/downloads/>, last access: 30.10.2015

¹² *Planet OSM*, URL: <http://planet.openstreetmap.org/>, last access: 30.10.2015

¹³ *Category:Treaties*, Wikipedia, the free encyclopedia,

URL: <https://en.wikipedia.org/wiki/Category:Treaties>, last access: 13.05.2016

¹⁴ *List of former sovereign states*, Wikipedia, the free encyclopedia, URL: https://en.wikipedia.org/wiki/List_of_former_sovereign_states, last access: 13.05.2016

is developed “out of a love of history and heritage of old maps” and currently stores about 400000 historical maps¹⁵. There are five steps to retrieve a border with points in geographic coordinates from an historical map.

1. **Digitization:** If the map is on paper, it has to be scanned in the best possible quality. The result is a raster graphic.
2. **Georeferencing:** The historical map has to fit as good possible on the reference map. This requires to manually define a set of reference points which are used to transform the map into the geographic coordinate system. This process is error-prone, especially if the projection of the historical map is not known and the map itself is not accurate [Kno02, pp. xvii]. The outcome is a raster graphic in which each pixel is assigned a geographic coordinate.
3. **Preprocessing:** The raster image has to be processed so that the desired border stands out and can be traced in the next step. This happens via greyscale conversion, thresholding or the Canny Edge Detector. This results in a monochrome graphic in which the desired border must be uninterrupted and clearly be seen.
4. **Line detection:** By selecting a start and an end point of the border, the line gets traced automatically. This step vectorizes one particular feature, a borderline, from the raster graphic and produces a polyline in geographic coordinates.
5. **Postprocessing:** In the last step, the polyline can be adapted: The line can be simplified to reduce unnatural artifacts and the position of border points can be manually edited. The final output of the whole process is a polyline whose points are expressed in the geographic coordinate system which can be used in the system as a border of an historic country.

This process was developed in a preceding *HiBo* project (see figure 2.3).

Manual data input For the domain of this HGIS, the evolution of countries over time, there is no complete dataset available. Therefore, the system developed in this thesis needs to have an interface to enter historical data. The user needs to have an interface to enter information about historical events that change territories and names of historical countries. This data has to be acquired either from primary historical sources directly, or from free online sources. Next to Wikipedia, there are other collections of historical events, e.g. *Correlates of War*¹⁷ for quantitative data about international relations.

¹⁵ Old Maps Online, URL: <http://www.oldmapsonline.org/>, last access: 13.05.2016

¹⁵ HiBo - semi-automatic extraction of borders from historical maps, Project of: B. Weber, N. K. Dankwa, K. Singh and T. Kashyappan, supervised by: Prof. Volker Rodehorst and Marcus Kossatz, Bauhaus-Universität Weimar, February 2015, URL: https://bitbucket.org/bastian_weber/hibo, last access: 29.10.2015

¹⁷ Data Sets, Correlates of War, URL: http://www.correlatesofwar.org/data-sets/folder_listing, last access: 13.05.2016

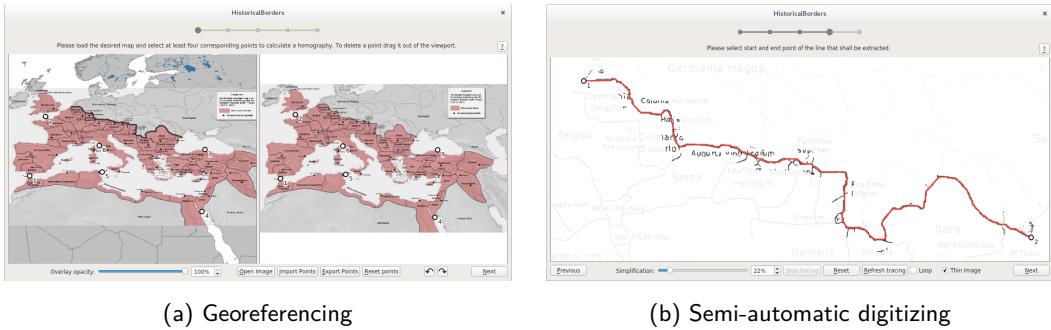


Figure 2.3: Semi-automatic extraction of a border from a map of the Roman Empire ¹⁶

2.2 Time and Space

This section will explain ways to separately represent time and space in an information system. It will first explain the geospatial data model used in traditional GIS and then introduce maps as the representation of spatial information. In the second part of the chapter, models to represent the temporal dimension are introduced.

2.2.1 Model of Geographical Space

HGIS need to unambiguously locate geo-objects on, underneath or close to the Earth's surface using *geographic coordinates*. They express an object directly in the coordinate system of the Earth. To understand that, a model of the Earth has to be developed, the *geodetic datum*, that needs to fit the real shape of the Earth as accurately as possible.

The shape of the Earth measured in the field of *geodesy* is very complicated. In the Babylonian Empire ($\approx 2000\text{-}539$ BC) the theory of the Earth being a flat disc surrounded by an infinite body of water evolved. The Greek scientists Pythagoras and Aristotle (340 BC) rejected this theory and proved the earth to be a three-dimensional spherical object. It took almost 2000 years until Sir Isaac Newton (1687) reasoned that due to the centrifugal forces of the rotating Earth the shape has to be flattened at the poles and is therefore better described as an *ellipsoid* with two radii: the polar radius (r_p) and the slightly larger equatorial radius (r_e) [Bol08, pp. 69-77].

However, the model disregards that the surface of the Earth is not flat but consists of deep oceanic trenches and high mountains. Therefore the gravitational field of the Earth is not homogeneous either: the actual *mean sea level*, the reference surface for the height of objects varies from 106 meter below to 85 meter above the uniform sea level of the ellipsoid model. These discoveries in the 20th century led to the complex *geoid* model (see figure 2.4). The latest and most accurate measurements for the shape of the Earth are the result of the GOCE satellite launched in March 2009 [Uot, Fra].

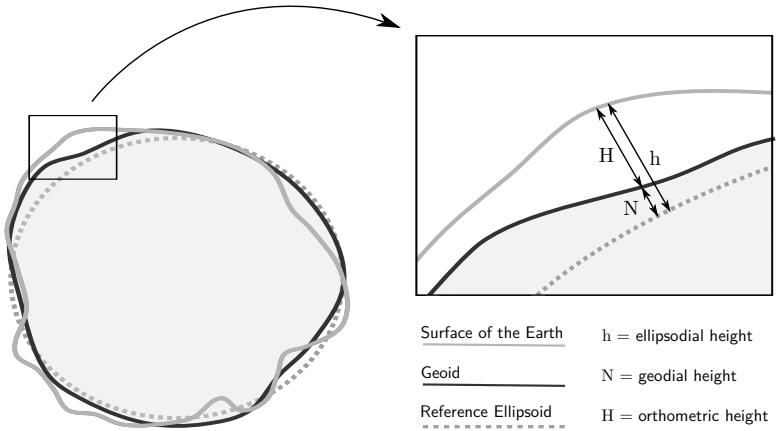


Figure 2.4: The geoid model, differences are exaggerated, [Bol08, Fig. 3-6, p. 75]

Geographic coordinate system The basis for the geospatial data model is the reference ellipsoid. It is represented in a three-dimensional *spherical coordinate system*. The *North* and the *South Pole* are defined as the two surface points closest to the Earth's center opposite to each other. The *Equator* is the line equidistant to the two poles and dividing the world in a *Northern* and *Southern Hemisphere*. Additionally, the *Prime Meridian* is defined as the line perpendicular to the Equator, running from the North to the South Pole. Since there are infinitely many lines like this, its definition is arbitrary, but by convention, the line running through Greenwich (London, United Kingdom) is used. Based on these two lines, each point in the spherical coordinate system can be unambiguously defined by [Bol08, pp. 26-28]:

1. The rotation angle along the Equator, defining its longitude: $\gamma = [-180^\circ \dots + 180^\circ]$
2. The rotation angle along the Prime Meridian, defining its latitude: $\phi = [-90^\circ \dots + 90^\circ]$
3. The distance to the origin: $r \in \mathbb{N}_0$

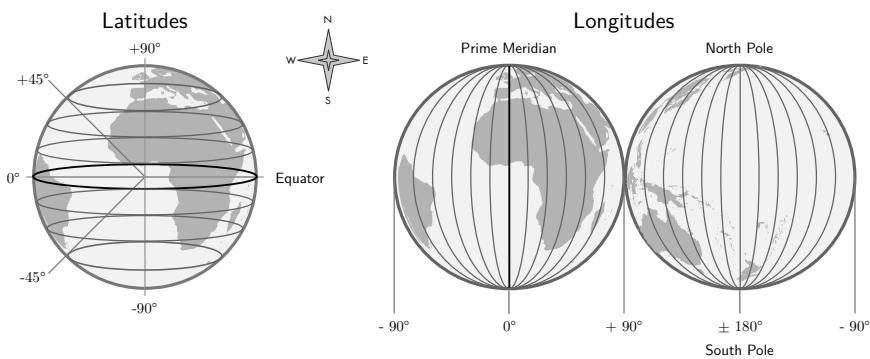


Figure 2.5: geographic coordinates using latitude and longitude

Lines of constant latitude are running horizontally and are called *parallels*, lines of constant longitude are *meridians* appearing in vertical direction. All parallels are circles with their center on the axis between the poles. No two parallels intersect. The longest parallel is the Equator (0° latitude). All meridians have the same length. Geographic coordinates are usually recorded either in degree-minutes-second (DMS, e.g. $50^\circ 58' 22''$) or in decimal degree (DD, e.g. 50.973) notation [Bol08, pp. 30, 79].

The Geodetic Datum is the digital model of the analogue Earth. It consists of two parts: The approximation of the Earth's surface in a the Cartesian coordinate system with the origin in the Earth's center and a set of reference points used to accurately locate a point.

Geodetic datums can be very accurate in one region of the world, i.e. the model fits the real geoid very well, but inaccurate in another region. This is the main reason why there are a lot of different geodetic datums used in the world. The same coordinates in two different geodetic datums define two different points on Earth. In order to be accurate is essential to know the geodetic datum of the coordinates [Bol08, p. 80]. The *World Geodetic System 1984 (WGS84)* is a model that found worldwide acceptance and is used in all major Web-based mapping services like *OpenStreetMap* and in the GPS unit of major mobile devices.

Raster and Vector Model The real world is infinite in detail, but storage in a computer system is finite. In order to model continuous geographical phenomena in an information system, a relevant subset of them has been sampled to create discrete spatial data. It can be represented in a raster or in a vector model.

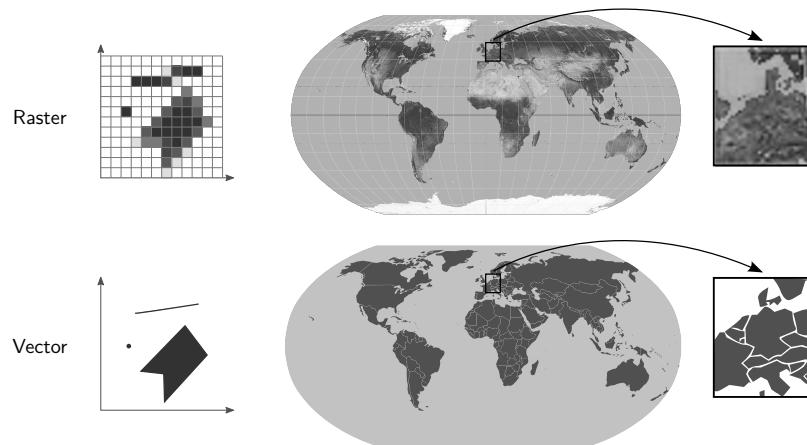


Figure 2.6: Comparison of the raster and the vector model

The *raster model* contains a regular grid with a fixed *cell dimension*. Each cell has a certain value, e.g.

a color value. The model is simple and allows straightforward rendering: only affine transformations have to be applied in order to project two raster map layers on top of each other. The main disadvantage of the raster model is its fixed resolution: it can not be scaled up without losing quality [Bol08, pp.42-48]. Raster graphics are used for map tiles by most map engines, e.g. in OpenStreetMap or the satellite image by NASA in Google Maps.

In the two-dimensional *vector model*, each object is a mathematically described geometric primitive. All of them can be expressed by three basic primitives (figure 2.7):

- 0D A *point* is the fundamental object in vector geometry. It has no dimension, no size and is only defined by its position, specified in geographic coordinates. One point is independent from all others. Points can be used to represent the location of an event.
- 1D A *polyline* is constructed by an ordered set of points with at least one start and one end point. A border line can be expressed by a polyline.
- 2D A *polygon* is an ordered set of polylines creating a closed area. A polygon can be *simple*, *weakly simple* or *complex* (see figure 2.8). The territory of a country without islands can be described by a polygon. If a country does have islands or overseas territories, a *polypolygon* represents multiple separate polygons belonging to one logical entity.

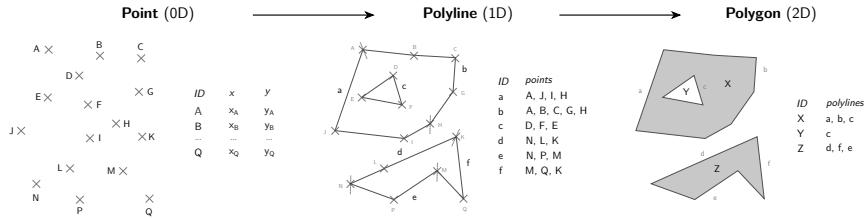


Figure 2.7: The basic geometric primitives point, polyline and polygon

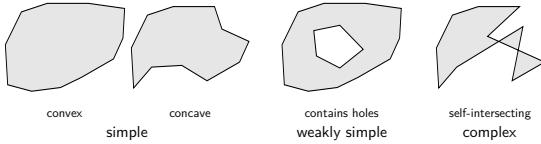


Figure 2.8: Different properties of polygons

Scale-independence is one of the biggest advantages of a vector model. The data model is more compact in comparison to the raster model. On the other hand, the model can become very complex. Since vector data has to be rasterized to be shown on the screen, the computational effort increases with complexity [Bol08, pp.33-42]. Vector models are suitable to represent phenomena that can

easily to discretized, e.g. the boundaries of a country. Common file types for vector data with spatial reference are the open file format GeoJSON (`.geojson`)¹⁸ or ESRI Shapefiles (`.shp`)¹⁹.

2.2.2 Presentation of Geographic Space

The most common ways to present geographic space are two-dimensional *maps* and three-dimensional *globes*. The HGIS in this thesis will use a map to show the evolution of countries over time. A map is a discrete graphical expression of the geographical features of the continuous real world. The creation of a map is not just a scientific, but also a creative process: The form, function and interaction methods shall follow the purpose of the usage of the map.

A map is typically structured according to the *layer* principle: Each layer is a transparent film showing one specific aspect, e.g. a physical layer showing coastlines, mountains or forests, a political layer showing international borders or a cultural layer showing cities or population densities. The layers are interchangeable, can be switched on and off and serve to serve a different visualization purpose. A *legend* including the scale bar and north arrow shall explain all symbols used on the map and give orientation. In interactive web based systems, there should be *menus* with different visualization options, e.g. panning and zooming on the map, switching map layers on and off or changing the color scheme of the map [Bol08, pp. 159-166].

Leaflet.js is “an open-source JavaScript library for mobile-friendly interactive maps”²⁰ that offers functionality to embed a map with a chosen projection in on the client-side of a web based information system, use own map tiles, symbols and markers on the map and tools for zooming and panning.

Map projections Since the Earth is three-dimensional, but the map on the computer screen only two-dimensional, the model of the Earth has to be projected onto the map. But as previously discussed in subsection 2.2.1, the Earth is an inhomogeneous spherical object with a curved surface whereas the map is flat [Bol08, p.79]. That is why some features of the Earth will be distorted on the map: An *equivalent projection* preserves the area sizes of features on the map, whereas a *conformal projection* preserves angles and the shapes of objects. Every map projection that is area-preserving distorts shapes at the same time, and each shape-preserving map distorts areas to some degree. There is no perfect map projection. [Geo].

¹⁸ *GeoJSON*, IETF GeoJSON Working Group, URL: <http://geojson.org/>, last access: 30.10.2015

¹⁹ *ESRI Shapefile Technical Description*, ESRI White Paper, July 1998, URL: <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>, last access: 30.10.2015

²⁰ *Leaflet - JavaScript library for interactive maps*, URL: <http://leafletjs.com/>, last access: 02.11.2015

²¹ *Tissot indicatrix world map equirectangular proj*, Eric Gaba / Sting (Wikimedia), June 2008 URL: https://commons.wikimedia.org/wiki/File:Tissot_indicatrix_world_map_equirectangular_proj.svg, last access: 28.10.2015

²² *Logo of the United Nations*, Shizhao (Wikimedia), 13.06.2007 URL: [https://commons.wikimedia.org/wiki/File:Logo_of_the_United_Nations_\(B%26W\).svg](https://commons.wikimedia.org/wiki/File:Logo_of_the_United_Nations_(B%26W).svg), last access: 28.10.2015, Comment: This work is

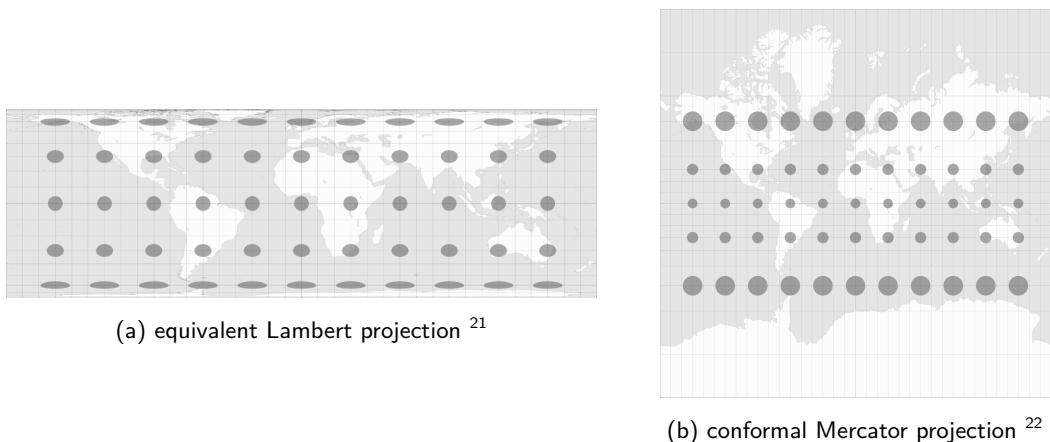


Figure 2.9: Comparison of equivalent and conformal map projections

A compromise between preserving areas and shapes is the *Robinson projection*. It is neither conformal, nor equivalent, but provides a reasonable trade-off between both properties.

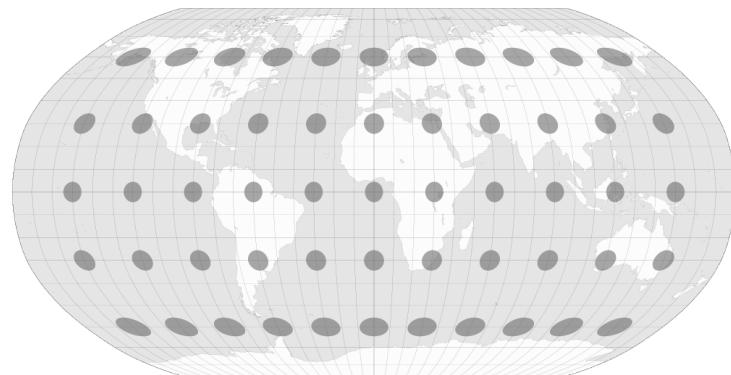


Figure 2.10: Robinson projection ²³

2.2.3 Model of Historical Time

Time is an abstract concept that “can be perceived only by its effects” [Lan89, p. 27]. Many philosophers and scientists have been developing models to work with time. In this case, the model needs to be appropriate to both represent time in an historical sense in interplay with geographical space.

excerpted from an official document of the United Nations prior to 17. September 1987.

²³ *Tissot indicatrix world map Robinson*, Eric Gaba / Sting (Wikimedia), June 2008 URL: https://commons.wikimedia.org/wiki/File:Tissot_indicatrix_world_map_Robinson_proj.svg, last access: 28.10.2015

A popular model is *Cartographic Time*, where time is seen as the “fourth cartographic dimension”, is suitable for spatio-temporal information systems [Lan89, p. 28]. Whereas space is represented by geo-objects on a map, time may be seen as versions or states on a timeline, separated by events that change one state to another state. Unlike space, time knows only one dimension. The position of an event on the timeline is described by its date using a reasonable sampling unit like century, year, day, hour or millisecond [Lan89, p. 32].

Types of Time The simplest categorization is between a discrete *event* and a continuous *process*. Events can happen at a certain *time point* or like processes in a *time interval* or *time period*, defined by two time points. An information system that stores events with a significant outcome regarding the geo-objects in the system, is an *event-based historical geographic information system*. On the other hand, a *process-based historical geographic information system* models mainly processes as a series of events of one kind regarding a small set of geo-objects [Sol14, chapter 2, pp. 47-49].

The Taxonomic Model of Time by [Fra98] classifies time not only into discrete and continuous, but also by the *nature of time* or *time order*: a consecutive development on the time axis, defined by start and end, defines *linear time*. In a contrary, *cyclic time* has no predefined order and events reoccur on a regular cyclic basis. The other two types, *branching time* and *multi-dimensional time*, are more complex and not relevant for this thesis.

Temporal Topological Relations The topological relationship between two time points t_1 and t_2 is straightforward. Since they are discrete elements and therefore isomorphic to the number space of integers, there are three different order relations:

1. $t_1 < t_2$: the first event happens before the second event
2. $t_1 > t_2$: the first event happens after the second event
3. $t_1 = t_2$: the first and the second event happen at the same time

For time spans, there are six possible temporal topological relations (table 2.2). Except for equals, each of them has an inverse, yielding a total of 13 different relations.

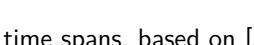
relation	symbol	visualization
X before Y	$X < Y$	
X meets Y	$X \text{ m } Y$	
X overlaps Y	$X \circ Y$	
X equals Y	$X = Y$	
X starts Y	$X \text{ s } Y$	
X during Y	$X \text{ d } Y$	
X ends Y	$X \text{ e } Y$	

Table 2.2: Temporal relations of time spans, based on [All84]

2.2.4 Presentation of Historical Time

In contrast to space, time does not have an intrinsic representation. However, the most common form to visualize cyclic time is on a cyclic display, e.g. a time wheel or a clock. Linear time is very often visualized on a *timeline*. The purpose of a timeline is to show events as time points or processes as time intervals in chronological order. A timeline additionally shows time markers showing a certain date to support orientation. A timeline uses a certain time scale:

- On a *linear* timeline, the distance between any two time points is directly proportional to their actual temporal distance.
- A *logarithmic* timeline uses a logarithmic function to scale the depicted time. Relative to a reference point on the timeline, e.g. the timeline center, the further away a time point, the further away its position on the timeline – however, the distance between the time point and the reference point does not increase linearly, but logarithmically. That means, events that are further away do not appear as far. This time scale accounts for logarithmic human perception: events that happened 20 years ago do not seem to be twice as long ago as events happening 10 years ago [DISP08].
- A timeline can also have an *irregular* scale, e.g. to have the same absolute distance of events on the timeline. This is useful if the distribution of the events on the timeline are far from homogeneous.

The visualization of time can be separate from the spatial dimension, according to the Triadic Framework, e.g. with a timeline. In another approach, space and time can also be coupled and displayed in the same presentation display, e.g. in a space-time cube [Häg70].

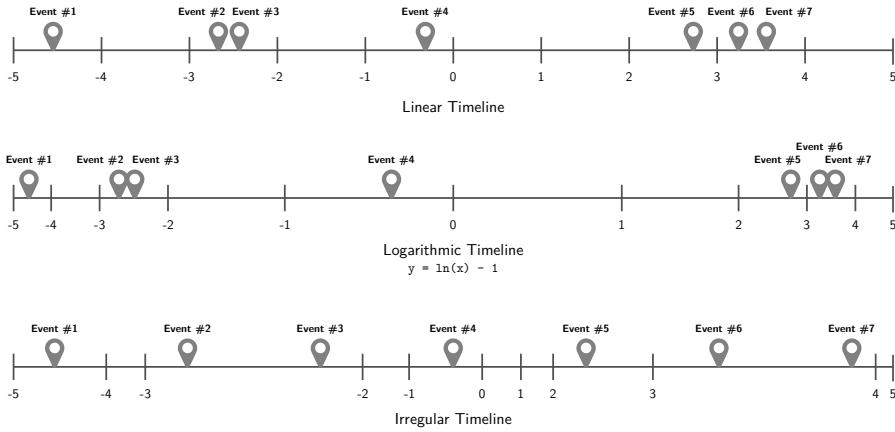


Figure 2.11: Comparison between a linear, a logarithmic and an irregular timeline

2.3 Spatio-Temporal Data Models

*“Geography differs from geometry because
in geography, space is indivisibly coupled with time”*

– Don Parkes & Nigel Thrift (1980)

A model tries to replicate a part of the real world. A data model abstracts a part of the world, identifies the most essential elements and their relation to each other. Historical Geographic Information Systems can be used to explain spatial-temporal phenomena in the real world. Therefore, it needs to handle the development of geo-objects and their attributes over time. Developments are driven by changes to the state of an object.

Based on the theory of the *Triadic Framework*, there are three components involved: space (3 dimensions), time (1 dimension) and attribute (multiple dimensions). All of these dimensions can change independently from each other [OS01, p. 53]. However, in order to trace spatial and attribute changes over time, the dimensions have to be related to each other.

Throughout the lifetime of a geo-object, it appears at some point, might undergo several changes and might disappear at some other time point. The data model has to be able to effectively and efficiently manage those changes. There are mainly two kinds: *Discrete changes* are based on the idea of a *state machine*: At any point in the lifetime, an object is in a certain state. It stays there until an event occurs that suddenly changes the object into a new state at a discrete time point. As an example, if an armistice agreement between two former war parties *A* and *B* contains a deal to cede parts of the territory of *A* to become territory of *B*, this territorial change is sudden. On a contrary, an object can gradually change according to a *continuous process*, e.g. the change of the coastlines of landmasses [Peu99].

The spatio-temporal data models developed in the previous 30 years differ mostly in the organizing dimension: In *location-based* models time is an attribute of a geo-object. On a contrary, *time-based* approaches handle events and processes that change objects suddenly or gradually. *Entity-based* models represent geo-objects as own entities. Spatial changes over time are related to these entities, but they are not attributes and therefore independent.

This section introduces different spatio-temporal data models to maintain relations between time and space of an entity.

2.3.1 Snapshot Model

One of the simplest, oldest and most frequently used models is based on the idea of *snapshots*: At a certain time point t_i , a new layer gets created. It stores the full picture of the current state of all geo-objects [Lan88].

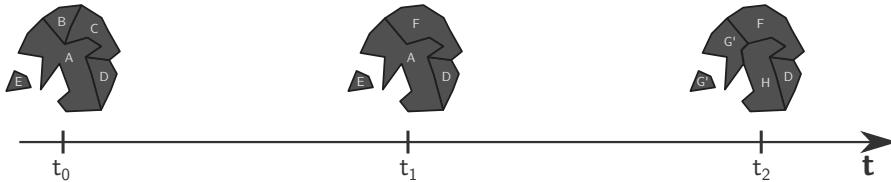


Figure 2.12: The Snapshot Model by [Lan88]

The model allows to retrieve the state of the system at a defined time point t_i . However, for all other time points $t \neq t_i$ that are not covered by a snapshot, it is impossible to retrieve the state of the system, because the data model does not record any changes. This is an integral problem of the model and can not be solved. The original model is also redundant, because objects that have not changed from one snapshot to the next one are duplicated. However, there have been improvements made, e.g. by [Arm92]. Historical maps are examples for snapshots: They show the state of the world at one point in history, e.g. Europe 1919 and Europe 1945. However, with no additional information, it is impossible to deduct how Europe looked like in 1939. Therefore, this model is not suitable for the domain of this thesis.

2.3.2 Simple Time-Stamping

This problem is solved by assigning a geo-object a period of existance by two additional attributes: at the *start date* t_{start} the object gets created and at the *end date* t_{end} it is ceased. If an object still exists its cessation date gets a special value, e.g. NOW [HW90].

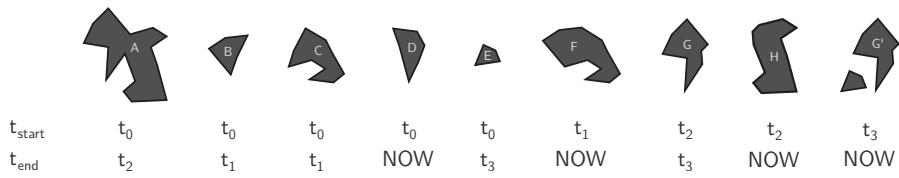


Figure 2.13: The Simple Time-Stamping method by [HW90]

The *Simple Time-Stamping* method is also location-based and tracks discrete changes of objects. Given full and integer information, the state of the system at each time point t_i can be retrieved: All geo-objects for which $t_{start} \leq t_i < t_{end}$ are active, all others are inactive. However, this retrieval is cumbersome, because without efficient data structures every time the date changes, it has to be checked for each geo-object if its state has changed.

Another problem of the model is that it does not allow for tracing the development of objects in different states. As an example, at time point t_1 the geo-objects B and C cease and G starts. Visually, G is a successor of B and C , but this historical relationship can not be deducted directly from the model. This shortcoming can be resolved by adding a reference to the predecessor and the successor of the object.

This model alone is not suitable for the domain of this thesis, because it is impossible to say what exactly has happened at a certain time point. Given the example above, it is unclear if two objects unified to a new one ($B + C \rightarrow G$) or if two are successors ($B \rightarrow G$) and one just stops to exist ($C \rightarrow \emptyset$). The model is also redundant: if a geo-object replaces another one ($B \rightarrow G$), then the end date of B is the same as the start date of G .

2.3.3 Event-Based Spatio-Temporal Data Model

A time-based approach addresses exactly those shortcomings: They explicitly represent events or processes in the data model and associate all objects that change according to them. One example of this approach is the *Event-Based Spatio-Temporal Data Model* (ESTDM) for geospatial raster data by [PD95].

At one defined time point t_b , a snapshot gets stored. This *base map* contains the current state of the map, i.e. the current value of each raster cell (x, y) . From that moment on, the system stores events that change the values of certain cells. Such an event has a time stamp (τ) and a list of components associated with it. A component represents a new value (v) and knows which raster cells (x, y) change their value to v .

The method uses the following data structures: a header file contains information about the thematic domain, a pointer to the base map and to the first and last element of the event list. This doubly-linked list stores all events chronologically. Therefore, each event knows its preceding and succeeding

event via a prev respectively next pointer.

If the time point of an event is reached, all its components are executed, i.e. the relevant raster cells change their value. The system follows the next pointer to know which event is waiting to be executed next. Since a change is relative to the previous change, not to the base map, change tracking is efficient.

The concept of the ESTDM suits the problem domain really well: An historical event changes the geometry of certain objects suddenly. The model explicitly represents these discrete changes. However, it does not work for vector data. The authors have explicitly stated that “the design of such a [vector-based] model is not seen as a straightforward task”, because of the problem “how to maintain the integrity of spatial topology as it changes [...] The solution will require a more complex definition of components within individual events” [PD95, p. 21].

2.3.4 Three-Domain Model

An event-based STDM for vector geometry including lines and polygons has to answer the following questions: What uniquely identifies a geo-object? What kind of spatial, topological and attribute changes can happen to an object? Which of these maintain the identity and which create a new object? This problem is addressed in the *Three-Domain Model* by [Yua96a, Yua96b]. The model is based on abstract entities that represent a spatio-temporal object. It handles the three domains identity, space and time separately:

- The *semantic domain* holds an entity uniquely identifiable. An object in this domain corresponds to a human concept, e.g. a “country”.
- The *spatial domain* represents geospatial objects in vector format, e.g. a polygon describing the territory of a country.
- The *temporal domain* stores all temporal objects, e.g. time points of an historical events, or time intervals of a war.

The model is not specific, but more a general abstract framework to handle space, time and identity. This makes the model very flexible, e.g. it can handle discrete and continuous changes, relative and absolute time, world and database time. One limitation of the model is that it only traces spatial attributes over time. In an alternative model by [CT95], the *thematic domain* is added to fully describe a spatio-temporal object and trace also aspatial attributes that can change over time, e.g. the name of an entity.

Since countries, their territories and attributes can change independently over time, the data model used in this thesis will be built on top of the Three Domain Model.

2.3.5 History Graph Model

Most of the data models introduced so far cover only static changes of geo-objects. [Ren96] identified three different types of temporal behaviour of changing objects:

- Dynamic objects that change continuously.
- Static objects that change according to events with duration (processes).
- Static objects that change according to sudden events.

Based on this observation he developed a data model that can handle all three kinds of temporal behaviour: The *History Graph Model*. It manages objects and events separately from each other. An object can only be in three different states:

1. An object is *static*, if it currently does not change. This is called an *object version*. The version has an interval associated to it representing the duration of the object version, until it changes the next time. If the object is dynamic and changes continuously, the duration is zero.
2. If an object is currently *changing*, it is in an *object transition*. The transition has an associated interval as well, whose duration is zero if it is a sudden change. Additionally, a transition links the relevant objects to each other creating a historical predecessor-successor-relationship.
3. An object that is currently not active, is *ceased* and not visible on the map.

The history of a geo-object is a chronologically ordered set of versions and transitions, that can be visualized in a graph (see figure 2.14).

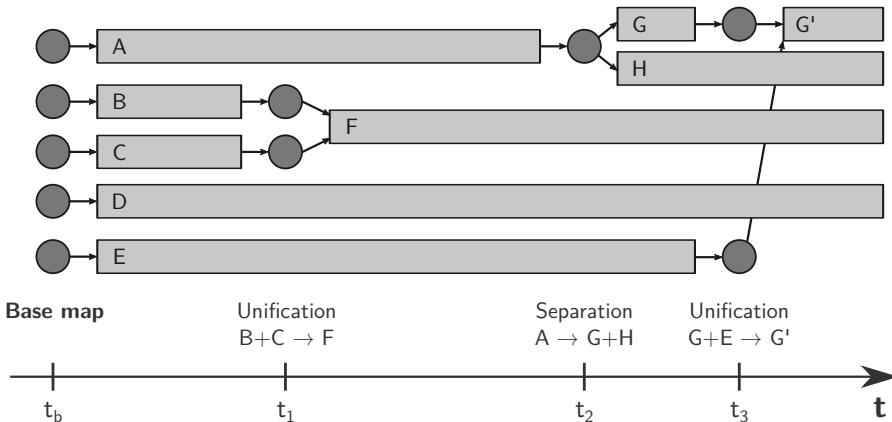


Figure 2.14: The History Graph model

The model defines six basic types of temporal changes that can happen (see figure 2.15):

- **Creation:** A new object is created.
- **Alteration:** A property of an object (e.g. geometry) changes.
- **Cessation:** An object is ceased.
- **Reincarnation:** An object that has previously been ceased is recreated.
- **Split/Deduction:** An object is divided into two or more new objects or one or more objects are deducted from an existing one.
- **Merge/Annexation:** Two or more objects are joined together to a new object or one or more objects are annexed to another object.

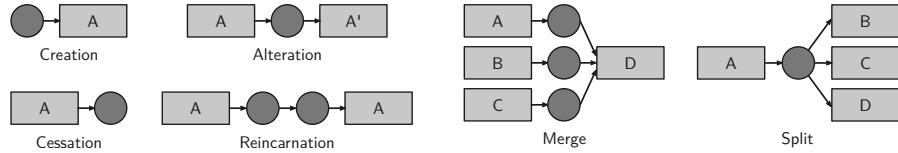


Figure 2.15: Types of changes in the History Graph model

The History Graph model can be seen as an extension to the ESTDM. It combines the advantages of event-based and entity-based spatio-temporal data models, supports discrete and continuous changes and relative and absolute time. The main improvement is that the historical development of a geo-object can directly be derived from the model, because objects are linked to their predecessors and successors — the History Graph Model can tell a story. This is the reason why this model is particularly suitable for the work of this thesis.

Other popular spatio-temporal data models that are not covered in this work, because they were not seen as relevant for the domain, include the *Space-Time Composite* model, the *Grid Model* and the *Amendment Vector Model*. Overviews about these and other spatio-temporal data models can be found in [Zha11], [PTKT04] and [Peu99].

2.4 Database Management Systems

Information systems use databases for managing the data. A *Database Management System* (DBMS) is a software system for the administration of data, mainly storage and retrieval. There are mainly two types of DBMS: the oldest and most common ones are *Relational DBMS*. *Object-Oriented DBMS* were developed to adapt concepts of object-oriented programming into the database world. The combination of both approaches are *Object-Relational DBMS*.

2.4.1 Relational Database Management Systems

RDBMS are built upon the concept of *entities*, e.g. an *HistoricalCountry*, with *attributes*, e.g. name and attribute values of a simple data type, e.g. the character string "Germany". Entities are represented in a table with one row for each *tuple* and one column for each attribute. An entity has one attribute that unambiguously identifies each tuple, the *primary key*, mostly a contiguous number.

Entities can be related to each other in three different kinds of *relations*:

- 1:1 Direct attributional relation, e.g. one country has one head of state and vice versa.
- 1:n One-to-many relation, e.g. one country can have many cities, but each city can belong to only one country.
- m:n Many-to-many relation, e.g. one country can have many rivers, but each river can also flow through multiple countries.

Entities and their relations are visualized in an *Entity-Relationship Model* (ER model). Data can be retrieved from and entered into a relational database using the *Structural Query Language* (SQL). The query to get the names of cities in Germany in alphabetical order is:

```
SELECT      city.id, city.name
FROM        (city JOIN country ON city.country = country.id)
WHERE       country.name = "Thüringen"
ORDER BY    city.name
```

The first RDBM developed was *Oracle*, released in 1979 [Ora]. Since then, the concept has been established as the state-of-the-art for databases. An example for a popular RDBMS used for Web-based systems is *MySQL*, the "world's most popular open source database" ²⁴.

2.4.2 Object-Oriented Database Management Systems

One problem with RDBMS is that attributes can only be assigned simple data types. Developers using object-oriented programming need to map the objects used in the application to tuples in the relational database – and vice versa data from the database needs to be transformed to objects in the application. This process can be cumbersome. OODBMS have been developed to address this problem and adopt the concepts of object-oriented programming for database management purposes [Dar].

- *Classes* are the structured representation of things in the real world of the same kind, with the same properties, e.g. a country, having a name and a territory. Classes in OODBMS relate to Entities in RDBMS.

²⁴ MySQL :: About MySQL, URL: <https://www.mysql.com/about/>, last access: 31.10.2015

- An *object* is an instance of a class, one specific thing with defined properties, e.g. the country of Germany with its territory. This correlates to a tuple in RDBMS.
 - The *attributes* of an object can not just be of a simple data type, but also instances of other classes, e.g. `country.territory` can be a `polypolygon` object. These complex data types are a major improvement compared to RDBMS.
 - Objects also have *methods* that can be called to do something with the object, e.g. `territory.getArea()` calculates the area size of a country.
- The internal state of an object can not be accessed from the outside. Methods are the only way to interact with an object. This is called *encapsulation* and maintains control over what can be done to and with an object and prevents corruption.
- According to the concept of *inheritance*, classes can be hierarchically structured, whereas the attributes and the methods of a *base class* are inherited to its *derived class*. As an example, an `Area` has a name, a `territory` and the method `getArea()` associated to it. A `Country` can be derived from the `Area`, inheriting both attributes and the method. Additionally, it can get an attribute `head_of_state`, which is specific to `Country`, but not to `Area`. The class `Ocean` can just as well be derived from `Area`.
- An associated concept is *polymorphism*: The same function can be called on different objects and the return value will be of the same type. However, internally it might be calculated differently. As an example, consider the classes `Polygon` and `Polypolygon`, both inherited the method `getArea()` from their base class `Geometry`. Whereas a polygon calculates its area directly based on its geometry, a polypolygon internally calls the function `getArea()` on all its associated polygons and sums up their areas.

OODBMS support all those concepts and allow to store the objects used in the application directly in the database. Additionally, objects from the database can be accessed directly, there is no need for an additional query language [Dar].

2.4.3 Object-Relational Database Management Systems

ORDBMS combine the advantages of both worlds. Internally, it uses the established and efficient relational database for the data storage. The database model and the interaction with the data happens in an object-oriented way while supporting all of the concepts mentioned in the previous subsection 2.4.2. The most popular ORDBMS example for Web-based systems is *PostgreSQL*, “the world’s most advanced open source database”²⁵.

²⁵ *PostgreSQL*: The world’s most advanced open source database, URL: <http://www.postgresql.org/>, last access: 31.10.2015

2.4.4 Spatio-Temporal Databases

Databases for HGIS have to deal with spatial, temporal and attribute data. According to the Triadic Frame, these aspects should be modeled separately from each other, as they can change independently.

Spatial data can easily become very large, because of the mass of very precise coordinate data. *Spatial databases* are specialized to work with spatial data: they process the data efficiently and to provide general data types, such as `Point` or `Polygon` and methods, e.g. to calculate the area or the distance between two points. *PostGIS*²⁶ is an extension for PostgreSQL that is especially utilized for handling spatial data.

Temporal data usually relates to events and processes. It is defined either by a time point or a time interval which is again defined by two time points. This is called a *bitemporal* element. A time point can be modeled in the database as an attribute of the complex `Date` type. For relational databases that only support simple data types, the date can be stored as a string or be expressed with a long integer (64 bit) $\forall n \in \mathbb{N} : n \in [-2^{63} \dots 2^{63} - 1]$ determining the number of milliseconds since 1st January 1970 (UNIX time) [Emi]. SQL was extended by features to handle time in a database, e.g. `SQL/MM` [Peu99, chapter 6].

Object-Oriented Spatio-Temporal Database Models The question is: How to implement the spatio-temporal data models introduced in section 2.3 in a relational, object-oriented or object-relational database management system introduced in this section? While the implementation depends on the data model, there are common concepts and issues that have to be addressed.

When storing time related data, it is important to distinguish between the time that was true in reality (*valid time* or *world time*) and the time it was stored in the database (*transaction time* or *database time*). A property of spatio-temporal database models is whether valid and transaction time are supported.

Object-oriented concepts are more appropriate than the concept of relational databases, because of the complex nature of spatio-temporal data [PTKT04, section 3.9]. One of the first concepts was the concept of a *spatio-temporal object* combining geometrical and bitemporal properties in one object [WHM90].

A similar approach by [Raz12] is the *Spatio-Temporal Data Type* (STT). Time is not considered an attribute of space, but a separate class. They are aggregated in the `Spatiotemporal` class, using both spatial and bitemporal attributes. The model also provides spatio-temporal operators, e.g.

²⁶ PostGIS, URL: <http://postgis.net/>, last access: 31.10.2015

`STT_intersects` returns `true` if two `SpatioTemporal` objects intersect in time and space, i.e. their geometries intersect and the time intervals in which they are active overlap. These operators are very helpful when analyzing spatio-temporal data or checking for data integrity.

Version management An issue is how to perform retrospective updates. Given a database model that stores objects that are created, updated and destroyed by events. Object X is created at time point t_x . At time point t_y , X gets destroyed and replaced by object Y . If a new change that updates X to X' gets inserted at time point t_u in between, i.e. $t_x < t_u < t_y$, then the event at time point t_y is not correct anymore, because object X does not exist. The question is how to maintain data integrity on insertion, update and deletion from a spatio-temporal database? This issue has to be addressed using formal logic for temporal reasoning [Peu99, section 6].

2.5 HistoGlobe

Application: HistoGlobe A distributed *Web Information System*, consists of a remote server side, on which the storage and management of the actual data happens, and the client side on which the user communicates with the system. It hosts the user interface that is rendered in a Web browser.

map for spatial domain (x, y) timeline for temporal domain (t) - \in 3D system

describe the components of the HG explicitly

ancestors successors layers of administrative units open to extension for additional attribute data (e.g. statistics)

requirements geographical knowledge contextualize / intersect historical sources accept imprecision prevent illusion of certainty

usable User Interface for both navigation and editing - \in problem: all interfaces are très horrible!

module system

transition to concept chapter

Chapter 3

Development

The aim of this thesis is to create a Historical Geographic Information System to visualize and edit the evolution of countries in time and space. This is a complicated task, because both the reality and the human using the system are complex. For such complex applications the methodologies of *Human Centered Design* are promising to create an interface that humans can easily understand.

The development process is iterative and divided in several phases. In each phase creates a prototype or the interface that gets closer to the final solution by increasing the fidelity of the prototype. A phase starts with an initial set of requirements. In multiple iterations, a prototype is developed that solves the problem. Each iteration has four steps: The requirements for the system are analyzed in the *planning* step. Afterwards, they translated into an abstract *design* which is realized in a specific *implementation* of the prototype. Finally, this prototype is tested with humans to find out how well it works. Based on the results of this *testing* step, the requirements are updated and the next iteration starts. This is repeated until a version of the interface is created that sufficiently solves the problem. Then the fidelity is increased, starting the next development phase. The five phases in this thesis are shown in figure 3.1.

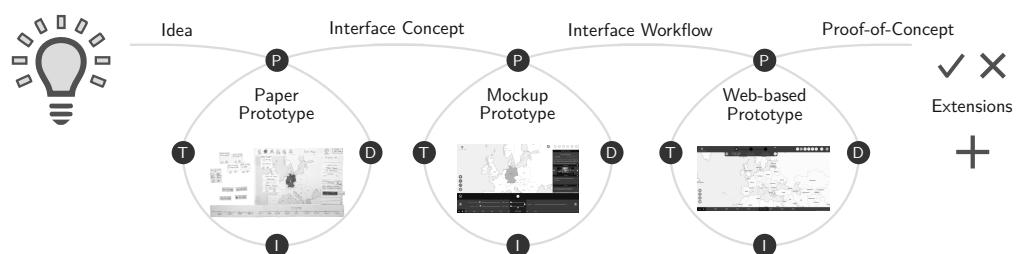


Figure 3.1: Human Centered Design process with five project phases

1. **Idea:** The initial idea how to edit and visualize the history of countries.
2. **Paper Prototype:** The concept of the interface realized and tested on paper.
3. **Mockup Prototype:** The concrete workflow developed in a slide-based presentation.
4. **Web-Based Prototype:** The final version developed in a Web application.
5. **Extensions:** Design approaches to account for the uncertain nature of history.

There are several models involved in the development of the software, each of them has to be developed or analyzed separately. The *data model* is an abstraction and simplification of the real world. The *Hivent Model* developed in this thesis is explained in the first section 3.1 of this chapter. It follows the method to *edit* the spatio-temporal data in the system in section 3.2. In interactive computer systems, the *mental model* is the representation in the mind of the human about how the interface should work – the *conceptual model* describes the way the interface actually works. The goal of Human Centered Design is to match the conceptual model to the mental model. Section 3.3 outlines the gradual design process to reach this goal. In the application, the data model is implemented in the *database model*. The task for the *computational model* is to translate between the database model and the conceptual model. The HistoGlobe application, including the database and computational model, is introduced in the last section 3.4 of this chapter.

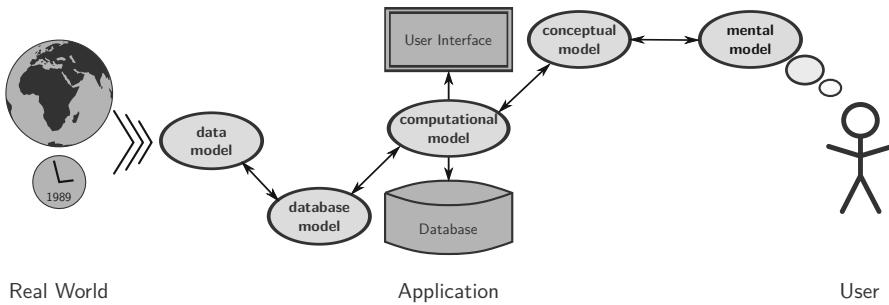


Figure 3.2: Relevant models for an information system

3.1 Hivent Model

This section proposes the spatio-temporal *Hivent model* to represent countries and their evolution in time and space. In section 2.3, different spatio-temporal data models were introduced. The *Snapshot Model* is unsuitable for the problem space. *Simple Time-Stamping* is helpful to link countries to their history, but it does not explicitly model historical changes, which is desireable. For that purpose, the idea of the *Event-Based Spatio-Temporal Data Model* was developed, but since it only works for raster data, it is also not suitable for this thesis. This problem is solved in the *History Graph Model*. Additionally, the introduced temporal changes allow to represent historical changes and their influences on geographic entities directly in the model. Finally, the *Three-Domain Model* introduces a helpful concept to separate the spatial, temporal and thematic dimension of a spatio-temporal entity.

The Hivent Model is constructed from components of some of these models: It is event-based and supports vector data. It is organized in four domains and allows to visualize data on a graph. The first section 3.1.1 introduces the main elements of the Hivent model. Afterwards, the preconditions are defined in section 3.1.2. One major contribution of this thesis is proposed in section 3.1.3: the set of five *Hivent Operations* that describe all possible changes of countries in time and space. This section closes with the *HistoGraph* (section 3.1.4), a visualization of the evolution of countries.

3.1.1 Elements

Hivents represent historically significant happenings, e.g. a treaty, bill or declaration. The word is an acronym for **Historical event**. The focus in this work is on events that influence the geopolitical situation on Earth. An Hivent happens at one particular point in time and space and is therefore the main organizing elements of the eponymous data model.

Areas represent one identical current or historical country. They are an abstract entity on the map with a *name* and a *territory*. The name consists of a common *short name*, e.g. “Germany” and a *formal name*, e.g. “Federal Republic of Germany”. The *territory* of the Area is described by a polypolygon, a set of weakly simple polygons to support enclaves and exclaves. The polylines of a polygon consist of an ordered set of points that represent a border of the country. They are either *interior*, i.e. bordering another country, or a *coastline*, bordering a body of water. Additionally, an Area keeps a reference to the historical changes creating, updating and ceasing it.

Historical Changes influence the evolution of Areas over time. Throughout the lifetime of an Area, it is created at some point t_s , then its territory and short name can change multiple times $t_i : t_s < t_i$ and at some point $t_e : t_s < t_i < t_e$ it ceases. Since all changes in this model are sudden, there are only two possible states an Area can be in: It is *active*, if at the current time point it is historically existing and it is *inactive* if it does not. Each Area is **uniquely identified by its formal name**. That means as soon as the formal name of an Area changes (e.g. “German Empire” to “Federal Republic of Germany”), it is considered a “new” Area. Each Historical Change belongs to exactly one Hivent, inheriting its time point at which the change happens. The change is described by a Hivent Operation introduced in section 3.1.3.

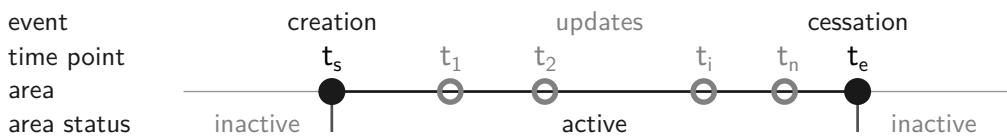


Figure 3.3: Three event types that change Areas, resulting in two different area states

3.1.2 Preconditions

*In the beginning God created the heavens and the Earth
 Now the Earth was formless and empty [...]
 And God said, "Let there be light" — and there was light.*

— Genesis 1:1, The First Book of Moses, Old Testament

There are five axioms and two assumptions the Hivent Model is based on. The theoretical foundation is the model of the Earth and its curved surface that can be projected on a two-dimensional map using a map projection, as introduced in sections 2.2.1.

Axiom 1 *The Earth's surface has an invariant area size, i.e. it does not change over time.*

Axiom 2 *Each Area in the spatio-temporal system is located directly on the surface of the Earth.*

These axiom sets the spatial foundation of the system: a constant dimension of the map and Areas covering the map. The basis of the temporal part of the system is content of the next three axioms:

Axiom 3 *The spatio-temporal system has an initial state at time point t_0 . At this initial state, there exists exactly one Area, denoted by Ω and referred to as the universe Area. It has no name and its territory covers the whole surface of the Earth.*

Axiom 4 *At each time point $t_i \geq t_0$ multiple historical changes can be introduced.*

Axiom 5 *At each time point $t_i \geq t_0$ each point on the surface of the Earth is covered by exactly one territory of exactly one Area.*

As it has been defined in section 3.1.1, an Historical Change can create, manipulate and cease Areas on the Earth's surface. According to axiom 5, each change introduced in the system must maintain the spatial integrity on the map: When an Area with a territory is created on the map, the Area claiming this territory before has to cease it. Formally, it can be said that each change consists of a set of old Areas A that are manipulated, a set of new Areas B that are created in the change, and an operation \rightarrow_C describing the change. Each Area $A_i \in A$ and $B_i \in B$ has a territory A_i^T respectively B_i^T . For each change introduced in the system, the territories of the old Areas must have the same size than the territories of the new Areas to maintain the spatial integrity of axiom 5:

$$\bigcup_{i=1}^n A_i^T = \bigcup_{i=1}^n B_i^T$$

The first changes introduced in the system at time point t_0 are the creation of all bodies of water, including the oceans and lakes, denoted as W . Each Area $W_i \in W$ is created with their name and territory cut out of Ω . The result is that after t_0 , the map is divided into water (W) and land (Ω).

Land can at any point in time be either *claimed*, i.e. it is currently occupied by the territory of exactly one active Area, or on a contrary be *unclaimed*, i.e. belonging to Ω . It is a subtractive data model, because each new Areas territory is cut out of Ω .

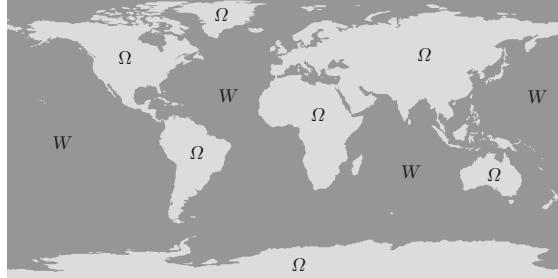


Figure 3.4: The initial state of the world map at time point t_0

In the real world, the name of a country changes according to sudden events, e.g. a declaration or a governmental bill. The territory can change either because of a geographical processes, e.g. the sea level rise influencing the change of the coastline, or according to a historical event, e.g. a treaty. The Hivent model is based on two assumptions that simplify the model and keep the problem space clear:

Assumption 6 *The territory of a country stops at the coastline.*

Assumption 7 *The spatial configuration of water and the coastlines have not changed over time.*

Both assumptions are obviously wrong: In line with [Uni82], the territory of a country extends in a range of 3 to 12 miles (5 to 20 kilometers) into international waters. They are constantly changing and so does the distribution of land and water on Earth. However, the assumptions allow the Hivent Model to focus only on discrete historical changes. It is subject to future work to extend the data model to account for long-term processes that change water and the coastlines. For now, the temporal behavior of an Area in the Hivent Model can be described as a *static object that changes according to sudden events*.

3.1.3 Hivent Operations

Respecting the preconditions, there are several different types of changes that transform a set of old Areas A to a set of new Areas B . All possible changes can be expressed with only five spatio-temporal operations that are called *Hivent Operations*. The first four change the identity of a set of Areas and therefore establish historical predecessor-successor-relationships. They are always symmetric, i.e. if one old Area is replaced by one new Area, the old Area is the historical predecessor of the new Area and vice versa the new Area is the successor of the old Area. The last operation changes an aspatial property of an Area.

UNI – Unification A set of old Areas unifies to one new Area. The old Areas cease, becoming the historical predecessors of the new Area. The territory of this new Area is the union of the territories of the old Areas. The new Area receives a new name.

In 1922, the Russian SFSR, the Transcaucasian SFSR, the Ukrainian SSR and the Byelorussian SSR unified and formed the Union of Soviet Socialist Republics (USSR).

INC – Incorporation One or more old Areas are incorporated into another Area that stays active. Its territory is enlarged by the union of the territories of the old Areas. The old Areas are historical predecessors of the Area that stays active.

In 1990, the territory of the German Democratic Republic (East Germany) became part of the Federal Republic of Germany (West Germany). Although this event is known as the *German Reunification*, it is historically an incorporation of East Germany into West Germany [Jan].

SEP – Separation As the inverse of unification, one old Area is separated into multiple new Areas. Each new Area gets a part of the territory of the old Area, receives a new name, and has the old Area as its only historical predecessor.

In 1993, the Czech and Slovak Federal Republic, commonly known as Czechoslovakia, dissolved into present-day Czech Republic and Slovak Republic, creating two new countries out of one old.

SEC – Secession As the inverse of incorporation, one or more new Areas are ceded from a previously existing Area that stays active. Each new Area gets a new name, receives the previously existing Area as the only historical predecessor and a part of its territory.

In 2008, the Republic of Kosovo declared independence from Serbia and has since then partially received international recognition. Serbia stays a country, keeping its name, but ceding a part of its territory to Kosovo.

NCH – Name Change An Area changes its short name but preserves its formal name and identity.

A recent change happened on 5. May 2016: The cabinet of Czech Republic approved that the country will now officially be called “Czechia”. However, the formal name stays “Czech Republic”, which preserves its identity.

UNI Unification	INC Incorporation	SEP Separation	SEC Secession	NCH Name Change

Table 3.1: The five Hivent Operations

3.1.4 HistoGraph

Based on the idea of the History Graph Model (section 2.14), the linguistically and conceptually related *HistoGraph* visualizes the evolution of countries in time, without any spatial relation. The edges of the graph represent an Area, the nodes a Hivent Operation. The graph shows the predecessor-successor-relationships between Areas. This is easily possible, because in the Hivent Model an Area keeps references to the historical changes creating, updating and ceasing the Area (section 3.1.1).

The two-dimensional HistoGraph has an horizontal orientation. The x-axis refers to one time point, the y-axis has no spatio-temporal relation and depends how much space it needs. The graph uses the visualization approach of the five Hivent Operations (table 3.1), including the following symbols:

—	Area
●	Identity-changing Hivent Operation UNI, INC, SEP, SEC
○	Property-changing Hivent Operation NCH
●○	A combination of both e.g. INC + NCH

Each uninterrupted horizontal line refers to exactly one Area. If an horizontal line leads straight through a circle, the identity of the Area is preserved in the operation. New Areas resulting from an identity-changing Hivent Operation emerge from the circle with a vertical line, indicating a sudden change with zero duration. From this line, the new Areas branch out right-angled. The HistoGraph is created from one particular reference Area. It visualizes historically related Areas in one direction: into the past, it recursively plots the predecessors on the graph, but not the predecessors successors. Into the future, the successors of the reference Area are plotted recursively, but not their predecessors.

The behavior of the HistoGraph is shown in figure 3.5 at the example of present-day Germany and its state history since the end of World War II. This history is driven by six historical events, which provide examples for all five Hivent Operations. They are listed in table 3.2.

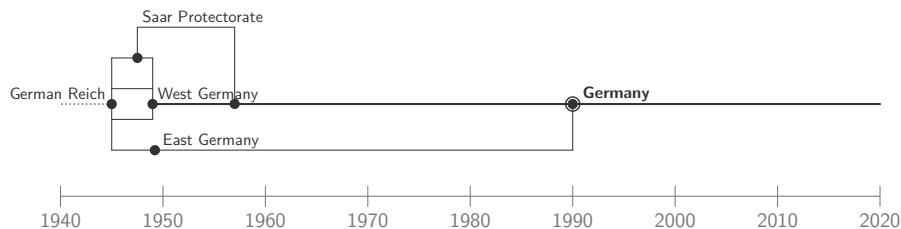


Figure 3.5: The concept of the HistoGraph at the example of the history of Germany since 1945

The example hosts a special case: in October 1949, East Germany was created from the Soviet Zone of Occupation. Both Areas have the same territory, but a different short and formal name. A NCH can not be performed, because the identity is not preserved: The German Democratic Republic is a new Area. However, the change can be described by a UNI of only one Area (Soviet zone), creating a new Area (East Germany) and establishing a historical relationship between both.

Hivent date	Hivent description	Hivent Operations
05.06.1945	In the Berlin Declaration the total dissolution of the Third Reich is confirmed. It separates into multiple parts, returning the territories annexed by the German Reich in World War II. The rest is controlled by the British, French, American and Soviet occupation zone.	SEP
16.02.1946	The Saar Protectorate is entangled from the French Zone of Occupation Germany, creating an own country.	SEC
28.05.1949	The Federal Republic of Germany (West Germany) is created from the British, American and French Zone of Occupation.	UNI
07.10.1949	The German Democratic Republic (East Germany) is created from the Soviet Zone of Occupation.	UNI
01.01.1957	The Saar Treaty ("Little Reunification") joins the Saar Protectorate as the Bundesland Saarland in West Germany.	INC
03.10.1990	In the German Reunification, East Germany joins West Germany. The Federal Republic of Germany is now just called "Germany".	INC + NCH

Table 3.2: Historical events in German state history since 1945

The graph plots Germany first. Since it does not have any successors, the plot goes only one way, historically backwards: East Germany and the Saar Protectorate were incorporated into Germany, so they are plotted. They emerged from the four post-war occupation zones, visualized next. All of the four occupation zones themselves originated from the German Reich. However, the Reich dissolved into many more Areas, e.g. the Memel territory. They are not included in the graph, because they are not predecessors of any Area that is a recursive predecessor of present-day Germany.

Many problems of the graph visualization are apparent in this example: Circles may overlap, if many operations happen in a short period of time – in this case between 1945 and 1949. The name "West Germany" collides with the vertical line indicating the incorporation of the Saar Protectorate, which should also be avoided. Additionally, the names of the Areas of the four post-war occupation zones can not be shown in the graph, because there is no space for them. One more important aspect can be seen in the creation of West Germany in 1949: A UNI operation unifies three old Areas to one new Area. This could be visualized symmetrically with a straight line from the midmost incoming Area line into the circle to the outgoing Area line of the new Area. This would give the wrong impression that this midmost Area has the same identity than the newly created Area. In general, the circle for UNI and SEP operations with an odd number of old respectively new Areas must be displaced off the center to emphasize that the identity has changed. All these issues are not in the scope of this thesis and subject to future work in the field of Information Visualization.

3.2 Editing Hivent Data

The previous section proposed the abstract Hivent Model, a set of Hivent operations and a visualization method. However, one purpose of the HGIS developed in this thesis is to add, alter and delete historical changes. This section presents the tools and methods to edit spatio-temporal data about the evolution of Areas in the Hivent Model. Whereas the Hivent Operations are well-defined and specific, user studies have shown that they are not well understood by humans to edit Areas. This thesis therefore introduces a different set of six *Edit Operations* in section 3.2.1. Afterwards, section 3.2.2 shows a *workflow* to perform an Edit Operation step by step. The Hivent Model needs to support editing historical changes in between other historical changes. The last section 3.2.3 explains the theoretical approach to *retrospective updates* of spatio-temporal data in the Hivent Model.

3.2.1 Edit Operations

The Hivent Operations are valuable, because they can describe all possible changes in the evolution of Areas in time and space. They are really well understood from the system point of view and form the basis for the Hivent Model. However, one purpose of the HGIS developed in this thesis is to provide a well understood user interface to edit historical changes to Areas.

Throughout the development process, interviews with researchers in humanities at University of Virginia were conducted to understand their mental model about the task. It turned out that the Hivent Operations are not suitable to be used for human edit purposes, because of their low-level nature. One example is that the operations do not provide a straightforward way to create a new Area on previously unclaimed land. The same is true for changing the formal name of an Area. Therefore, this thesis introduces a second set of operations: five high-level *Edit Operations* describe changes to countries on the map (see table 3.3). They have proven to be understandable in several user studies.

	CRE	Create	a new Area with a new name and territory on the map.
	MRG	Merge	two or more Areas to a new Area. The name has to be set manually, the territory is automatically unified.
	DIS	Dissolve	one Area into two or more new Areas, manually setting their new territory and name.
	CHB	Change Borders	between two neighboring Areas by defining the territory that changes sides.
	REN	Rename	an Area and set a new formal name, short name or both.
	CES	Cease	an Area by deleting it from the map, leaving unclaimed land.

Table 3.3: The six Edit Operations

Error correction Another possible use case for the HGIS created in this thesis is to correct wrong information in the system. For this purpose it is important to understand how correcting information in an event-based spatio-temporal system works: Given time point t_y and an Area A with the name X . If X happens to be wrong, it means that the historical change at time point t_x : $t_x < t_y$ that created the name X for Area A is erroneous and has to be corrected. Correcting a state means correcting the operation that created this state.

3.2.2 Edit Workflow

An Edit Operation describes an historical change that can be understood and performed by a user of the HGIS. This section shows that each Edit Operation can be internally expressed by a set of Hivent Operations. Therefore the Edit Operations are an abstraction layer in the Hivent Model between the Hivent and the Hivent Operations. To create an Edit Operation, four steps in a workflow need to be performed:

1. Select the Areas that will be changed in the Edit Operation.
2. For each new Area resulting from the Edit Operation, create a territory.
3. For each new Area create a name.
4. Add the Edit Operation to an Hivent to inherit the time point.

For each Edit Operation, the requirements for the steps are different. Not all operations need all steps, because some data can be processed automatically. Table 3.4 presents an overview about the behavior of the Edit Operations in the first three steps. The last step is necessary for all.

	<i>Select old Areas</i>	<i>Create new territories</i>	<i>Create new names</i>
CRE	–	create a territory of the new country	create a name for the new country
MRG	select the countries to be merged	– territories of selected countries are automatically unified	create a name for the new country
DIS	select a country to be dissolved	create a territory for each new country	create a name for each new country
CHB	select two neighboring countries to change their border	create the new border between both countries the territory for both countries will be created automatically	–
REN	select a country to rename it	–	create a new name of the country
CES	select a country to cease it	–	–

Table 3.4: The requirements of each step for the Edit Operations

Depending on the input of the user in the steps for an Edit Operation, there are different possibilities to express it by a set of Hivent Operations. Each Hivent Operation transforms a set of old Areas into a set of new Areas and can update the name or territory of one specific Area. All possibilities are introduced in table 3.2.2. Hivent Operations are combined when they happen at the same time. In the example of the German Reunification, East Germany was incorporated into West Germany which at the same time changed its short name to “Germany” (INC + NCH).

EditOp. (case)	old Areas	update Areas	new Areas	expression by Hivent Operations ¹	visualization
CRE (1)	Area B_1 is created with territory T . The part of T that is on previously unclaimed land (T_Ω) is seceded as B_1 from Ω . If T_Ω is empty, then B_1 is initialized with an empty territory. The rest of T covers some Areas A_p partially and some Areas A_f fully. For each A_p , the covered territory T_p is seceded and incorporated into B_1 . Each A_f is completely incorporated into B_1 .	n_f	n_p	SEC of B_1 from Ω SEC of T_p from A_p , INC of T_p into B_1 INC of A_f into B_1	
MRG (1)	Multiple Areas A_i are unified to B_1 . The new Area receives a name distinct from all the names of A_i .	$n \geq 2$	0	1	UNI of $\forall A_i$ to B_1

EditOp. (case)	old Areas	update Areas	new Areas	expression by Hivent Operations ¹	visualization
MRG (2)	Multiple Areas A_i are unified. The resulting Area reuses the short and formal name of one of the old Areas (A_0) and therefore preserves it. The remaining Areas A_i are incorporated into A_0 .	$n \geq 1$	1	1	INC of $\forall A_i$ into A_0
MRG (3)	The same as the previous case, just that A_0 receives a new short name and therefore an additional name change is required.	$n \geq 1$	1	1	INC of $\forall A_i$ into A_0 NCH of A_0
DIS (1)	Multiple Areas B_i are separated from one initial Area A_0 . Each B_i receives a part of the territory of A_0 and a name. Each name is distinct from the name of A_0 .	1	0	$n \geq 1$	SEP of A_1 into $\forall B_i$
DIS (2)	Multiple Areas B_i are separated from one initial Area A_0 . Each B_i receives a part of the territory of A_0 and a name. One of the separated Areas has the same short and formal name as A_0 , so it preserves its identity. The remaining new Areas secede from A_0 .	1	1	$n \geq 1$	SEC of $\forall B_i$ from A_0
DIS (3)	The same as the previous case, just that A_0 receives a new short name and therefore an additional name change is required.	1	1	$n \geq 1$	SEC of $\forall B_i$ from A_0 NCH of A_p
CHB (1)	One existing Area A_0 is selected and its territory changes. Relative to the old territory some parts of the territory expands (T_e) and some withdraws (T_w). The part of T_e that expands into unclaimed land ($T_\Omega : T_\Omega \in T_e$) is seceded from Ω and incorporated into A_0 . The Areas A_f fully covered by T_e are incorporated into A_0 , the Areas A_p partially covered by T_e secede this territory $T_p \in T_e$ to A_0 . T_w is be incorporated into Ω , resulting in unclaimed land.	n_f	$1 + n_p$	0	SEC of T_Ω from Ω , INC of T_Ω into A_0 SEC of T_p from A_p , INC of T_p into B_1 INC of A_f into B_1 SEC of T_w from B_1 , INC of T_w into Ω

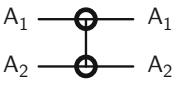
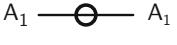
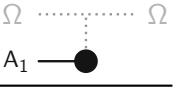
EditOp. (case)	old Areas	update Areas	new Areas	expression by Hivent Operations ¹	visualization
CHB (2)	Two existing Areas A_1 and A_2 are selected and their common border changes. This results in a symmetrical change of territories, made up by two sets of territories: T_2 that previously belonged to A_1 and is now part of A_2 and T_1 for which the opposite is true. T_2 is seceded by A_1 and incorporated into A_2 , the opposite happens to T_1 .	0	2	0 SEC of T_2 from A_1 , INC of T_2 into A_2 SEC of T_1 from A_2 , INC of T_1 into A_1	
REN (1)	One Area A_1 is selected and both its short and formal name is changed. Therefore, a new Area B_1 is created as a direct successor of A_1 . This is a special case of a unification with only one Area.	1	0	1 UNI of A_1 to B_1	
REN (2)	One Area A_1 is selected and receives a new short name, but the formal name and therefore the identity is preserved. A_1 is updated.	0	1	0 NCH of A_1	
CES (1)	One Area A_1 is selected and ceases by incorporating into the universe.	1	0	0 INC of A_1 into Ω	

Table 3.5: Translation from Edit Operations to Hivent Operations

3.2.3 Retrospective Updates

A straightforward use case of the Hivent Model is to change the current state of the system with a new Hivent Operation into the future. Given the initial start point t_0 , a current time point $t_{now} > t_0$ and a set of consecutively added Hivent Operations at $\forall t_i : t_0 \leq t_i < t_{now}$. The accumulation of all changes make up the current state of the system at t_{now} . To change this current state, a new Hivent Operation can be inserted at t_{now} into the future. This state is valid until the next change is inserted.

For historical research that use case alone is not sufficient, because the current state of the map at t_{now} (2016) is known to a large degree. The problem is to describe states and changes in the

¹multiple Hivent Operations in one row happen exactly at the same time point, so they are combined

past. Therefore the system needs to support entering Hivent Operations in between other existing operations.

Integrity Each Hivent Operation that is not entered to the end of the timeline must maintain the semantic, spatial and thematic integrity of the data, i.e. the changes to Areas, their territories and names must still work. The simple example in figure 3.6 shows the problem: Given time point t_1 with two Areas A and B and an UNI Hivent Operation at t_2 unifying A and B to C . If in retrospective a new Hivent Operation is inserted at $t_r : t_1 < t_r < t_2$ that cedes a part of A to a new Area X , the operation at t_2 is not consistent anymore, because the old territory of A is not the same. It is not simple to say how the remaining territory ? should be treated.

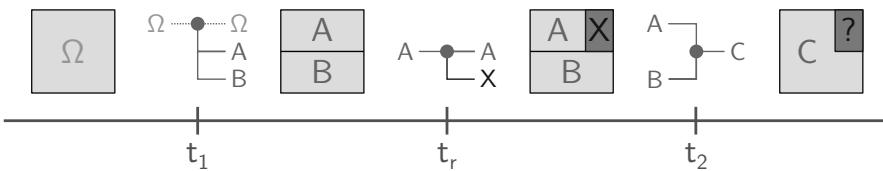


Figure 3.6: Example for a simple conflict due to a retrospective update

Conflicts The way the Hivent Model works is comparable to a version control system like *Git*². There are different kinds of conflicts that can occur on retrospective updates. In the Hivent Model, they are classified regarding their resolvability:

- A) The conflict can be resolved *automatically* without the interference of the user.
- S) The conflict requires the user to choose between two alternatives (*semi-automatic* resolution).
- M) The conflict is complex and the user needs to resolve it *manually*.

The remaining part of this section examines all possible cases of conflicts and their resolvability. Each inserted Hivent Operation transforms a set of old Areas $A = [A_i]$ to a set of new Areas $B = [B_i]$ or updates an update Area A_0 or both. Each consecutive Hivent Operation that manipulates $A_0, A_i \in A$ or $B_i \in B$ has to be checked regarding three aspects of integrity:

1. semantic: Does A_0 and $\forall A_i \in A$ still exist? If not, can it easily be replaced by another Area?
2. spatial: Is the territory of A_0 and $\forall A_i \in A$ still the same? If not, can it easily be updated?
3. thematic: Is the name of A_0 and $\forall A_i \in A$ still the same? If not, can it easily be updated?

All cases can be simulated in the following simple scenario: Given the system with only two states: an initial state at t_1 at which only three spatial entities are on the map (A_1, A_2, A_3) and an Hivent Operation at t_2 that manipulates some of these Areas with one of the five possible operations. This

² *Git*, - -everything-is-local, <https://git-scm.com/>, last access: 29.05.2016

is called the original Hivent Operation (H_o). Now, a retrospective update H_r is inserted in between the two states ($t_r : t_1 < t_r < t_2$). H_r manipulates the same set of Areas with an Hivent Operations. The question is: What happens regarding the semantic, spatial and thematic integrity of H_o ? Is there a conflict and if so, how can it be resolved? There are 25 possible cases, because for both H_o and H_r there are five possible Hivent Operations.

Retrospective Name Change The first five cases are straightforward: Given NCH is inserted in retrospective (H_r) to change the name of A_1 from X to Y . This has no effect on the identity or territory of A_1 . Therefore the system only needs to check for thematic integrity of H_o . If that operation is an INC or SEC, which both only change the territory of A_1 , it is not conflicting. If H_o is a UNI or SEP, then there is a conflict: A_1 is an old Area of the operation, but the name associated to A_1 is still X . This is not consistent, because H_r just changed the name to Y . This conflict can be resolved automatically by updating the name in the old Area from X to Y . The same is true if H_o is a NCH operation: A_1 is the update Area and the old name has to be updated from X to Y . To summarize what the system has to do if a NCH on A_1 is inserted in retrospective: find the next UNI, SEP or NCH operation that manipulates A_1 and update its old name.

Retrospective Incorporation An INC ceases a set of old Areas and changes the territory of one Area. In this scenario, H_r incorporates A_2 into A_1 . The question is what kind of conflicts can occur to the spatial integrity of H_o ? If H_o is a NCH, there is no conflict, because H_o changes the territory of A_1 and NCH the name.

Figure 3.7 shows the conflicts that occur in the remaining four possibilities of H_o . In the case of an original UNI operation, it can still be performed with the same Areas, because H_r did not change the identity of A_1 . However, the territory of A_1 has been enlarged. This new territory has to be taken into account for H_o : To maintain spatial integrity, the system has to update the territory of incoming A_1 . However, the territory of B_1 has to be updated as well, because it is enlarged in the same way as A_1 . This requires a recursive update into the future: the next Hivent Operation dealing with B_1 needs to take into account that the territory has changed. This case can be treated as if H_o would be H_r with an INC operation. The system has to repeat this process until all conflicts are solved. The same is true if H_o is an INC operation: The system needs to update the old and the new territory of A_1 in H_o and recursively update the territory of A_1 into the future.

If H_o is a SEP operation, there is a major conflict: originally, A_1 splitted into B_1 and B_2 . Due to H_r , the territory of A_1 is larger. H_o can still separate B_1 and B_2 from A_1 the same way as before, but it is unclear what should happen to the remaining territory of A_1 that has just been enlarged. This conflict has to be resolved manually, because the system can not derive a decision from any existing information. The remaining part could become Ω , it could be incorporated into B_1 or B_2 or stay A_1 . However, the user has to decide it. In case of an original SEC, the situation is slightly different: H_1 still exists like before, just with a larger territory. H_o can secede B_1 like originally, but the system

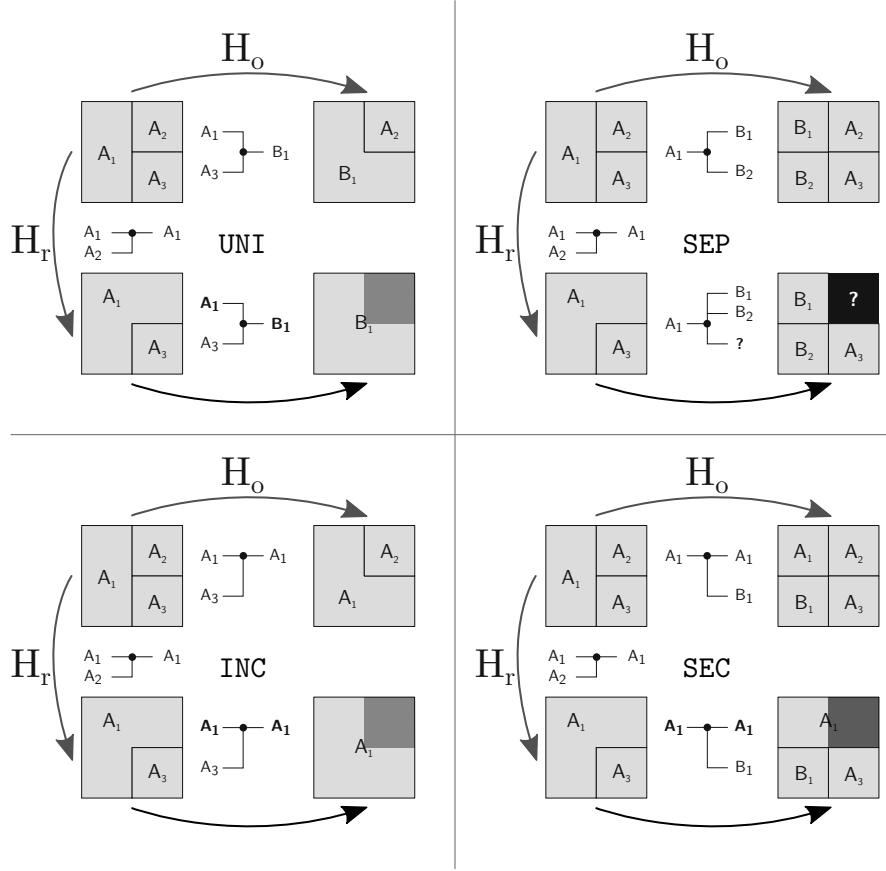


Figure 3.7: Conflicts after a retrospective incorporation

needs to update the old and new territory of A_1 in this operation and recursively into the future.

The foregoing investigation relates only to A_1 but not to A_2 that has been incorporated into H_r . From the perspective of A_2 , this change can be seen as a UNI, because its identity ceases and together with its territory it completely merges into A_1 . Therefore this is treated like a retrospective unification examined later in this section.

Retrospective Secession Retrospective secessions are comparable to incorporations: The identity and the name of A_1 does not change – but parts of its territory ceases to a new Area B_1 . This section examines how the system has to treat B_1 and the smaller territory of A_1 in the original operation H_o . Exactly like for retrospective incorporation, there is no conflict if H_o is a NCH.

The other four cases are visualized in figure 3.8. For an original UNI, there is a conflict: Originally, A_1 and A_3 unified to B_1 . Although H_r cedes parts of the territory of A_1 to R_1 , H_o can still merge

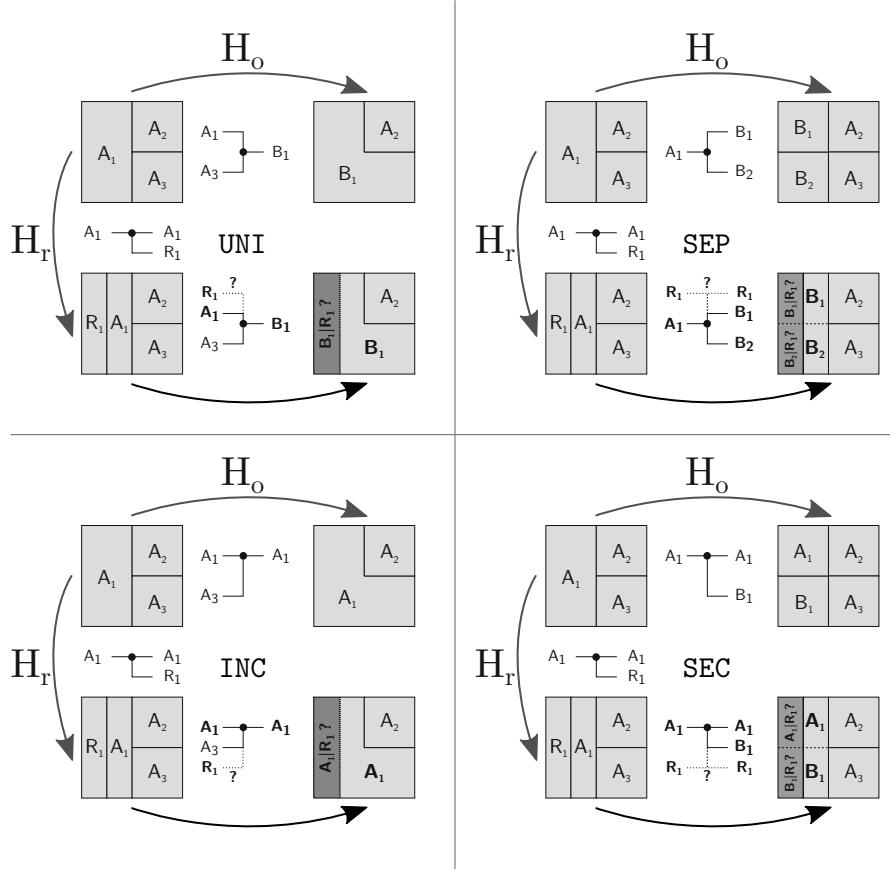


Figure 3.8: Conflicts after a retrospective secession

it with A_3 . The question is what happens to the remaining territory of R_1 ? There are two options:

1. *priority to H_r* : R_1 stays an own Area and is not affected by H_o .
2. *priority to H_o* : H_o unifies R_1 to B_1 as well.

For the system it is hard to decide this, because it is unclear what the intention of the user is when creating R_1 in H_r : Should the newly created Area be there for longer or is the territory of the united Area A_3 more important? In order for the system to not behave unexpectedly, it will ask the user which choice he or she prefers. In case of the first choice, the territory of R_1 has to be subtracted from the territory of the new Area B_1 in H_o . This update is again recursive, because the next Hivent Operation dealing with B_1 needs to operate on the correct territory as well. In the second case, R_1 simply has to be added to the old Areas of the UNI operation in H_o . No further recursive update is necessary. The same behavior is true if H_o is an INC. The user has to decide if he or she wants to incorporate R_1 into A_1 or keep it as a separate Area. In the latter case, the system needs to update the new territory of A_1 in H_o .

For an original SEP, the situation is comparable: Originally, A_1 splits into B_1 and B_2 , but after the retrospective secession of a part of A_1 to R_1 , the territory of R_1 is conflicting with B_1 and B_2 in H_o . An important observation is that each part of R_1 would be part of either B_1 or B_2 . There is no empty land, since both R_1 and $B_1 + B_2$ seceded from the same territory of A_1 . Just like the other two cases above, the system will give the user for both conflicting territories the choice if either R_1 should stay an Area or if B_1 respectively B_2 should incorporate R_1 into their territory. In the first case, the territory of R_1 has to be subtracted both from the incoming A_1 and recursively from the outgoing territory B_1 respectively B_2 . The latter case needs at least one additional Hivent Operation: the part of R_1 that shall be part of B_1 respectively B_2 needs to be seceded from R_1 and the same time incorporated into A_1 which then again in the same moment is separated into B_1 and B_2 ($H_o = SEC+INC+SEP$). If whole R_1 ceases in the operation, then it is incorporated into A_1 completely, so it is only a INC+SEP at H_o . The case of an orginal SEC the system behaves in exactly the same way, just with an update of the territory of A_1 and B_1 instead of B_1 and B_2 .

To summarize, the main difference between retrospective incorporation and secession is that in the latter case, the user needs to choose between two predefined options. For incorporations, this is seen as unnecessary, because the conflicting territory of A_2 has not been manipulated by H_o , so it can not be seen as a concious decision of the user to keep this Area.

Retrospective Unification If a UNI is inserted in retrospective, the semantic integrity of H_o is threatened, because in contrast to a INC all incoming Areas cease and unify to one completely new Area, in this example R_1 . For each Area $A_i \in A$ that was unified in H_r the system needs to find the next Hivent Operation H_o manipulating A_i and update it accordingly. In this example, A_1 is examined in place of each $A_i \in A$. If H_o is a NCH, there is a conflict: the name of A_1 can obviously not be updated anymore, because A does not exist anymore. The only way to resolve this conflict is to automatically delete the NCH operation. All the other four cases behave regarding spatial integrity in exactly the same way as for a retrospective incorporation – with the only difference that the Area A_1 is replaced by R_1 as an incoming Area in the operation. In all four cases, the territory has to be updated in the same way as for retrospective incorporation and the same conflict occurs for the original SEP operation.

Retrospective Separation In contrast to the previous example, retrospective separations behave slightly different from secessions. H_o has to be checked both for spatial and semantic integrity, since A_1 ceased in H_r to a set of new Areas R_i . In this scenario, R_1 and R_2 stay in place for $\forall r_i \in R$. If H_o is a NCH, the operation has to be automatically deleted, because A_1 does not exist anymore. Figure 3.2.3 shows the conflicts that arise for the remaining four cases.

In case H_o is a UNI, the arising conflict can be solved semi-automatically: originally, B_1 was created by unifying A_1 and A_3 . A_1 just got separated into R_1 and R_2 by H_r , so the system can not know if R_1 , R_2 or B_1 is preferred at H_o . In any case, the system needs to remove A_1 form the old Areas in

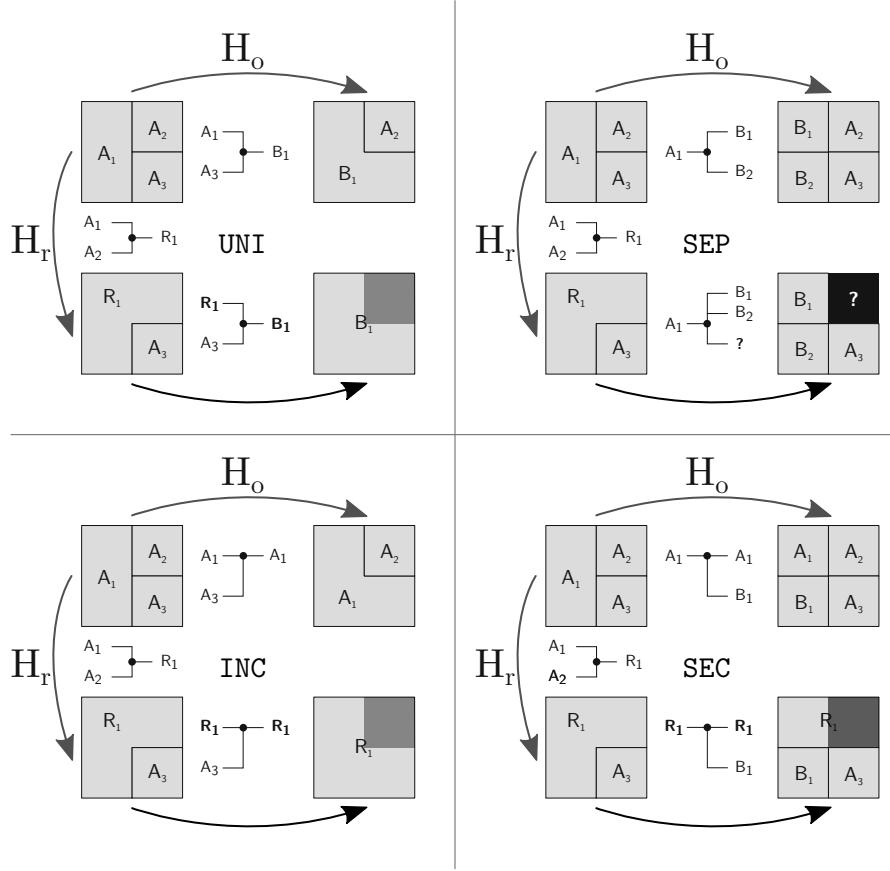


Figure 3.9: Conflicts after a retrospective unification

the **UNI** operation. For each created Area in H_r – in this case only R_1 and R_2 – the system asks the user if he or she prefers R_i or B_1 . If R_1 is preferred, then its territory has to be recursively subtracted from the outgoing Area B_1 in the **UNI** operation at H_o . Else R_i is added to the old Areas of H_o . If R_i is preferred every time, the remaining **UNI** operation only transforms A_3 to B_1 .

If H_o is an **INC**, the situation is more complex: The Area A_1 in which A_3 should be incorporated does not exist anymore. Moreover, it is replaced by a set of new Areas R_i and it is not straightforward to see where A_3 should be incorporated into – it is only clear that A_3 will cease in this operation. Since there is no information about what should be there instead, this conflict has to be solved manually. The situation is even more complicated, because on the one hand, in H_o the user intentionally incorporated A_3 into A_1 which means there is an obvious intention to keep A_1 . On the other hand, he or she separated A_1 in H_r to two new Areas R_1 and R_2 , so also they can be seen as intentionally created. To avoid an unexpected behavior of the system, the best approach is give him or her the choice to keep R_1 and R_2 , but in case of denial let the user resolve the whole conflict manually.

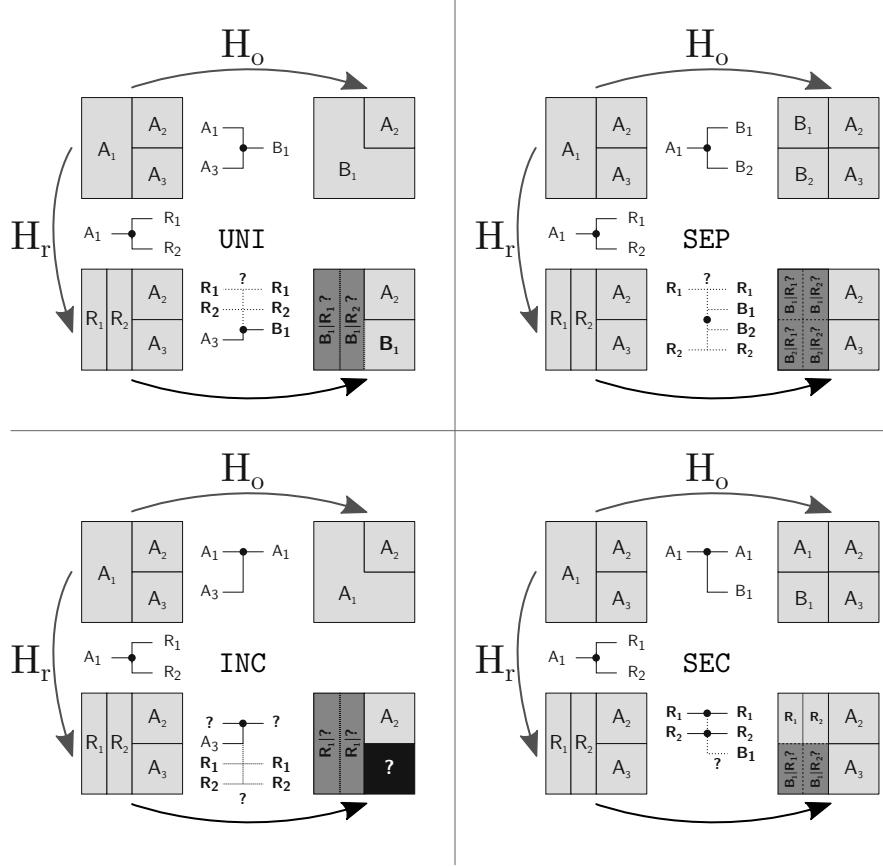


Figure 3.10: Conflicts after a retrospective separation

The case of an original SEP is likewise complex: A_1 does not exist anymore to be separated – instead there are two sets of Areas that can be seen as reasonable to be the outcome of the operation: On the one hand the Areas $B_i \in B$ created in H_o and on the other hand $R_j \in R$ created in H_r . Since it is impossible for the system to know which Areas to prefer, the user should decide for each possible combination $i \times j$ which Area it should be. If R_j is preferred, then its territory has to be recursively subtracted from each B_i that intersects with R_j . Else the territory of R_j intersecting with B_i has to secede from R_j and be incorporated into B_i with a combination of SEC+INC at H_o . The SEP operation itself is not necessary anymore, because there is not one simple old Area anymore.

The last case to examine is the influence of a retrospective SEP on an original SEC. Originally, $\forall B_i \in B$ would secede from A_1 , which has just been separated into $\forall R_j \in R$ in H_r . The part of the territory of each $R_j \in R$ not covered by any $B_i \in B$ will certainly stay R_j , because there is no other reasonable alternative. The conflict is at the overlapping territory between B_i and R_j . Just like for the previous example, the user has to resolve these conflicts semi-automatically by choosing either B_i or R_j for each possible alternative. If B_i is preferred, it is seceded from the related R_j at H_o . If

R_j is preferred, its territory has been recursively subtracted from the related B_i .

		original Hivent Operation H_o				
		UNI	SEP	INC	SEC	NCH
retrospective operation H_r	UNI	A	M	A	A	A
	SEP	S(A A)	S(A A)	M	S(A A)	A
	INC	A	M	A	A	X
	SEC	S(A A)	S(A A)	S(A A)	S(A A)	X
	NCH	A	A	X	X	A

Table 3.6: All possible conflicts on retrospective updates regarding their resolvability

X = no conflict, A = automatic, S = semi-automatic, **M** = manual resolution

For semi-automatic resolution, the resolvability of the two options is stated like S(1st|2nd)

All possible cases and their resolvability are visualized in the 5×5 matrix in figure 3.6. It became clear in the extensive examination of the conflicting cases that inserting a retrospective update into a spatio-temporal system is not straightforward at all. Especially if a separation or secession is added somewhere not to the end of the timeline, in almost all cases the user has to decide between two alternatives. In three cases there is even a manual resolution necessary. A lot of cases also require recursive updates, meaning the retrospective update can lead to a potentially high number of semi-automatic or even manual resolutions by the user, which can potentially be frustrating. On top of that the examination completely disregarded the situation of combined cases, e.g. if the original operation is a combination of SEC+INC resulting from a user-defined border change (BCH) between two Areas. This might become even more complex. In summary, the Hivent Model needs to be checked for potential adaptions that might simplify retrospective updates and make the data handling less frustrating.

3.2.4 Backward Operations

From a relative time point t_i there are two historical directions: forward into the future with a predefined end at the current time point t_{now} and backward into the past until the predefined start point of the system t_0 . Everything until now was focused on forward operations that change the current state of the system at the set time point t_i into the future, either until another operation changes it again or until t_{now} .

As it has been argued in the previous section 3.2.3 for retrospective updates, only forward operations to the end of the timeline are not sufficient for historical research. For a historian to edit a state in the past, a backward operation might be useful: A Hivent Operation is inserted at time point t_i , but into the past: $t_0 < t_i < t_{now}$. As an example: Given the initial state 10.06.2016

with present-day Germany created on 03.10.1990 on the map. The user wants to enter the German Reunification. The HGIS must support separating Germany into East and West, but indicating that this was the state *before* 1990 and the original state was *after* this date. This is complicated, because the conceptual, data and computational model have to adapt to this requirement.

The Hivent Operations themselves allow to be executed the other way, because each of them has an inverse operation: A `UNI` can be inverted with a `SEP` and a `INC` with a `SEC` operation. `NCH` can be inverted with itself by swapping the old and new name.

One problem is the conceptual model: The user interface has to provide a visual clue that inverting the direction of an operation is possible. Additionally, if an Hivent Operation is inserted backwards, another problems comes into play: all new Areas of the operation would now be active from t_i on backwards into the *past*. Each Area that is created in a backward operation has to be provided with another operation that ceases it in backward direction or creates it in forward direction, otherwise the Area would be active all the back to t_0 . This is probably not desirable.

3.3 User Interface Design Process

HistoGlobe is the application in which the work of this thesis is implemented. The Hivent Model presented in the section 3.1 serves as its data model and the methods to edit Hivent data in HistoGlobe (section 3.2) form parts of the computational model. However, developing the system bottom-up from the data model to the interface might not lead to usable system. Human Centered Design promotes a top-down process from the user via the interface into the core of the application. This section illustrates the iterative design process for this thesis seen in figure 3.1. The two main use cases for HistoGlobe that are focused in this thesis are:

1. **Understanding** the history of countries.
2. **Editing** the spatio-temporal evolution of countries with historical changes by inserting forward and backwards operations and correcting wrong information in the system.

The interviews with researchers in humanities at University of Virginia confirmed that the combination of a map and a timeline are a very appropriate and intuitive way to interactively visualize the history of countries. Therefore, the main concept of HistoGlobe introduced in section 2.5 does not need to be changed. This concept is extended by the *HistoGraph* introduced in section 3.1.4. This promising set of visualizations forms the *Browsing Mode* of HistoGlobe to understand the history of countries.

For editing purposes the idea of a second interface mode was developed: The *Edit Mode* is the main product of the iterative design process illustrated in this section. It is based on the Edit Operations, the workflow to edit the data and the concepts of retrospective updates and backward operations from section 3.2. The Edit Mode allows to intuitively edit Hivents, Areas and operations directly in HistoGlobe, without the need to write data into tables or forms.

Initial interviews Four researchers were asked about their opinions on the idea of HistoGlobe, use cases and the concept of the Edit Mode. The idea proved popular, especially for students and teachers in school, historically interested people in general and also for scholars in digital humanities. All researchers agreed that the key to a successful interface is usability, because editing data in time and space is a challenging task. A main concern is uncertainty: Almost all sorts of information in historical research – temporal, spatial and thematic – are potentially uncertain. A good user interface for researchers therefore has to support uploading historical sources and indicating uncertainty.

3.3.1 Paper Prototype

From the results of the initial interviews, an interface concept for the Edit Mode was developed in a paper prototype. It very fast to create and allows to identify flaws in the concept early in the design process. For this thesis, two paper prototype iterations were created that took about three full work days each: one day to create the prototype, half a day to conduct the study with three people, and one and a half days to analyze the results and rethink the concept.

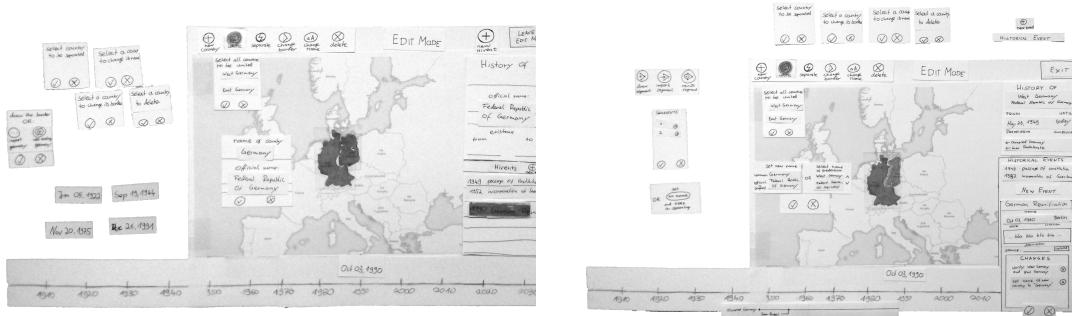


Figure 3.11: The two iterations of the paper prototype for the Edit Mode

The interface consists of a map of Europe, a timeline centered at 1975 and the buttons with a set of dialogs for the Edit Mode. Both prototypes were evaluated with three test subjects that had to solve four tasks covering different use cases and operations:

1. 1300: Rename incorrectly spelled name of Switzerland on the map (*correction*)
2. 1990: Unite East and West Germany (*forward change*)
3. 1993: Separate the Soviet Union into Russia, Estonia, Latvia, etc. (*forward change*)
4. 1944: Change the border between Finland and the Soviet Union before 1944 (*backward change*)

Most parts of the interface concept were understood and all subjects could solve the first three tasks. However, there were also problems:

1. There difference between Hivents, the history of a country and an historical change was unclear.

2. The border drawing dialoge was imagined to be very complex.
3. The backward change was not understood.
4. Correcting the name Switzerland by changing the event that created it in 1300 caused confusion.

The main finding of this step was that depending on the task, there is both an Hivent-based and an Area-based mental model of the task. This became apparent in the German Reunification Hivent: Some users started the unification operation first, and added West and East Germany afterwards – and some selected first West Germany, then initiated a unification operation and then added East Germany. From that finding arose that the interface has to support both an Hivent-based and an Area-based approach to introduce historical changes and correct information on the map.

3.3.2 Mockup Prototype

The main part of the design process was spent on the mockup prototypes. Their purpose is to rapidly develop an interface workflow that is understandable by the users. The prototypes were created in *LibreOffice Impress*, an open-source slide-based presentation tool. The interface is simulated on slides: the map is a background image, the timeline, the set of buttons and dialogs for the Edit Mode and HistoGraph are modelled with geometric elements: lines, circles and rectangles. Interactivity is simulated by linking a click on an element to a different slide that shows the effect of the operation. This allows to model sudden changes in the interface.

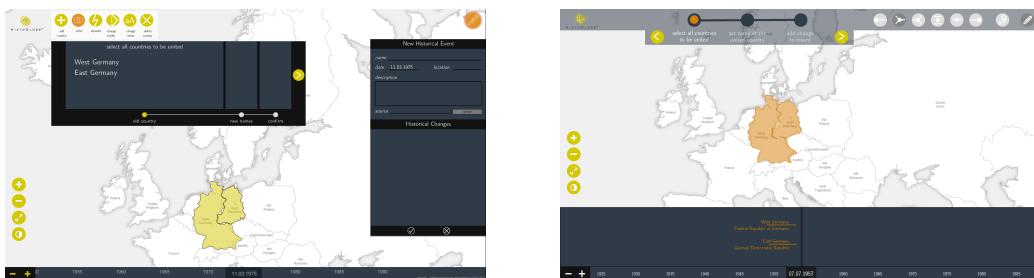


Figure 3.12: Two iteration stages of the mockup prototype for the Edit Mode

Each prototype iteration was tested with multiple subjects and similar tasks as for the paper prototype. From one test to the next one changes to the interfaces were made. A lot of design problems, e.g. position of buttons, font sizes or color schemes were solved, but also conceptual issues arose.

"this was much easier than I thought"

*"the interface is very clear
and graphically pleasing"*

"it's looking good"

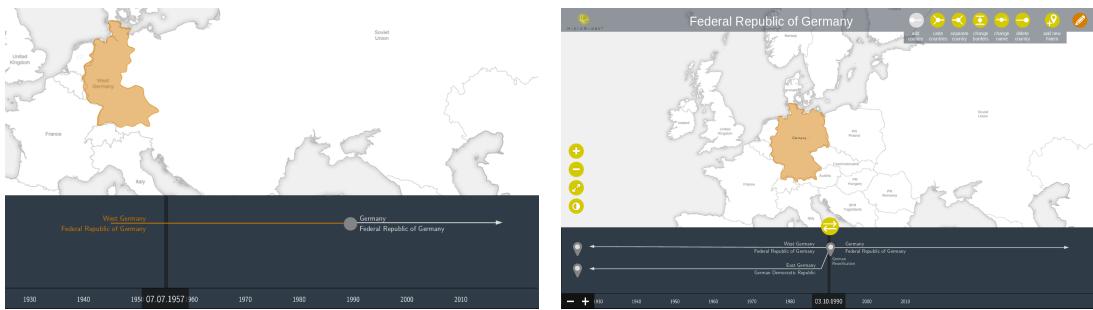
"there is a training session needed"

*"the logic makes sense,
it is just very complex"*

*"a nice tutorial and a good
documentation are necessary"*

– interesting quotes from the users of the mockup prototype were:

Especially the problem to initiate a retrospective update and a backward backward change proved to be very difficult. There was a design solutions developed for each problem: For retrospective updates, the interface needs to provide a visual clue where the next potential conflict arises: Figure 3.13b shows that West Germany can only be active until 1990, because then present-day Germany uses its territory. For backward operations a button that flippes an Edit Operation is introduced in figure 3.13b: the SEC operation introduced to secede East Germany from Germany will be flipped into an INC operation to incorporate East Germany into Germany. The two grey Hivent markers on the left side of the timeline indicate that West and East Germany need a creation event, otherwise they would be active backwards all the way to t_0 , the initial state of the system.



(a) Retrospective updates: Visualizing the next conflict (b) Backward operation: flipping the Edit Operation

The prototype was very valuable for the development process. In a total of two weeks, an interface concept and workflow was designed that was understandable by the users. Its creation took longer than the paper prototype, but was much faster than implementing an interactive Web-based interface.

3.3.3 Web-based prototype

The main advantage of the design process so far is that it prevents major redesigns of the final Web-based prototype. After three months of implementation of the final system, the interface looks very similar to the last version of the mockup prototype. The main elements of the interface are the map, the timeline with the Now Marker the control buttons the map and the timeline and the Edit Mode. However, not all desired features could be implemented: For the HistoGraph there were too many conceptual problems mentioned in section 3.1.4 that have to be solved first. Backward operations and retrospective updates are not supported as well, because of their complex nature. The HistoGlobe version developed in this thesis supports editing the evolution of countries with forward operations at the end of the timeline. The interaction and behavior is introduced in this section at the example of the fictional secession of Scotland from the United Kingdom in 2018.



Figure 3.14: Initial state of the Browsing Mode

The initial state of the user interface. Additional to the original elements, there is an edit button on the upper right corner. Clicking it enters the Edit Mode of the system.



Figure 3.15: Initial state of the Edit Mode

In the Edit Mode, a title bar and six buttons for the Edit Operations are revealed. Clicking a button starts the operation workflow introduced in section 3.2.2.



Figure 3.16: 1) Select old Areas

A *Workflow Window* guides the user step-by-step through the process of completing the Edit Operation. In the case of DIS, the user has to select the country to be dissolved by clicking it on the map. After the step is completed, clicking the next button in the workflow window proceeds to the next step. At each point in the workflow, clicking the back button reverts the previous action.



Figure 3.17: 2) Set a new territory

In step two, the user has to create a territory for each new Area. The *New Territory Tool* provides the functionality to create, manipulate and delete polygons directly on the map. The drawn polypolygon is intersected with the old territory to create the new Area. After at least one new territory is created successfully, the remaining old territory can be selected on the map to be used as the last territory. If the whole old territory is distributed among the new Areas, the workflow proceeds to the next step.

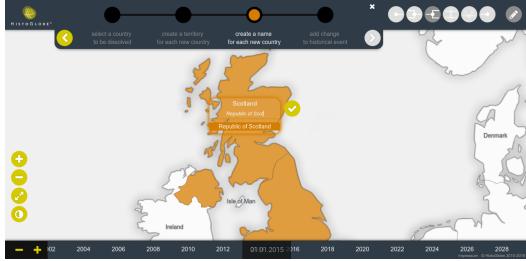


Figure 3.18: 3) Set a new name

In the next step, the user has to define a name for each Area that has just been created. The *New Name Tool* is a draggable box with two lines for the short and formal name. Via instant search, the user can select existing country names from the database to be put in. When clicking the confirm button, the short name is put directly on the map.



Figure 3.19: 4) Add the Operation to a Hivent

When all names are set, the Edit Operation is complete. In the last step of the workflow, it has to be added to an Hivent. The *New Hivent Box* offers two possibilities: search for an existing Hivent and add the operation to it or create a new Hivent.



Figure 3.20: 4) Create a new Hivent

The new Hivent created for that change is the “Scottish Independence” on 01.01.2018 with a description of the Hivent and possibly a location and a link to a wikipedia article. In the last line, the historical change “Secession of Scotland from United Kingdom” is noted. Clicking the confirm button finalizes the workflow.



Figure 3.21: The final state with Scotland

Clicking the edit button again leaves the Edit mode back to the Browsing Mode. Scotland and the United Kingdom are visible as separate Areas on the map after 2018. When moving the timeline before 2018, Scotland is still part of the UK.

3.4 Application

HistoGlobe is a Web-based Historical Geographic Information System. The Data model and the conceptual model of the user interface were introduced in the first sections of this chapter. This section introduces the underlying database model, a specific implementation of the data model, and the computational model that translates between the conceptual model and the database model. The first part provides an overview about the architecture of the system in section 3.4.1.

3.4.1 System Architecture

HistoGlobe uses a classical client-server architecture of a Web-based information system. The user opens the application and interacts with it through the user interface in a Web browser, the *client* side of the system. The Web *server* is a remote computer that hosts the database and the middleware. The user interacts with the interface and the client-side application sends a request to the Web server for new data. The middleware checks the request and queries the necessary data from the database. It transforms the data and sends it back to the client. The interface shows the new information.

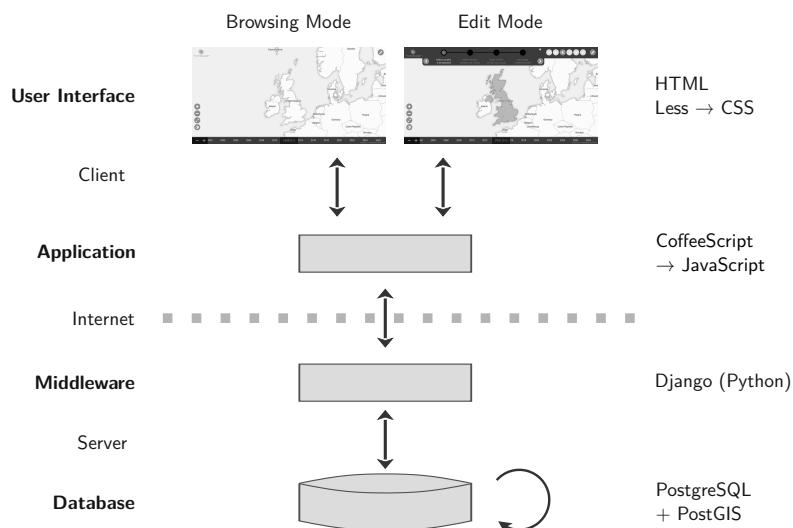


Figure 3.22: The system architecture of HistoGlobe

This clear separation between the data, the application and the user interaction in this chapter and in the system follows directly from the *model-view-controller* pattern: One part can be changed independently from the others parts: if the 2D map is replaced by a 3D globe, only the view changes, but the middleware and the database can stay untouched. Likewise, the implementation of a new database technology has no consequences to the view.

3.4.2 Server-Side Application

The underlying Hivent Model is implemented on the server-side part of the application. HistoGlobe uses *Django*, a free and open-source web framework³, combined with *PostgreSQL*⁴, one of the most popular Object-Relational Database Management Systems introduced in section 2.4.3, on the server-side of the system. This allows HistoGlobe to take advantage of object-oriented concepts in a stable and fast relational database. Since the database is using a lot of geospatial data, *PostGIS* is used as a spatial database extension for PostgreSQL⁵.

With these tools at hand, the Hivent Model from section 3.1 was implemented in a database model shown in figure 3.23. It is the final result of a highly iterative process that underwent many improvements and adaptations to new requirements introduced in the Human Centered Design process. The model is structured in two parts covering four different domains of the spatio-temporal model: The lower part describes the semantic, spatial and thematic domain of Areas and the upper part represents the temporal domain of Hivents that introduces changes to the Areas.

Semantic, Spatial and Thematic Domain In the Hivent Model, the entity visible on the map is an Area with a name and a territory, as introduced in section 3.1. In the database model, they are represented by three entities:

1. `Area`: semantic domain defining one identical Area with potentially changing name and territory. The `universe` attribute is true for Ω , for the other Areas it is false.
2. `AreaTerritory`: spatial domain. A polypolygon describes the geometry of the territory and a `representative_point` the position of the name label on the map.
3. `AreaName`: thematic domain. It is defined by a `short_name` and a `formal_name`.

Temporal Domain The main idea of the model is that the Areas can change over time. These changes are introduced by Hivents, the main entity of the eponymic model with five attributes: The name and a textual description of the Hivent, the point in time the Hivent happened (`date`), the Hivent location as a simple string and a link (URL) to the related article, serving as a historical source. Each Hivent can introduce a set of `EditOperations` introduced and understood by the user. They consist themselves of a set of low-level `HiventOperations`. They replace a set of `OldAreas` with a set of `NewAreas` and might update the name or the territory of one specific `UpdateArea`.

³ *Django*, The Web framework for perfectionists with deadlines, URL: <https://www.djangoproject.com/>, last access: 27.05.2016

⁴ *PostgreSQL*, The world's most advanced open source database, URL: <http://www.postgresql.org/>, last access: 31.10.2015

⁵ *PostGIS*, Spatial and Geographic Objects for PostgreSQL, URL: <http://postgis.net/>, last access: 27.05.2016

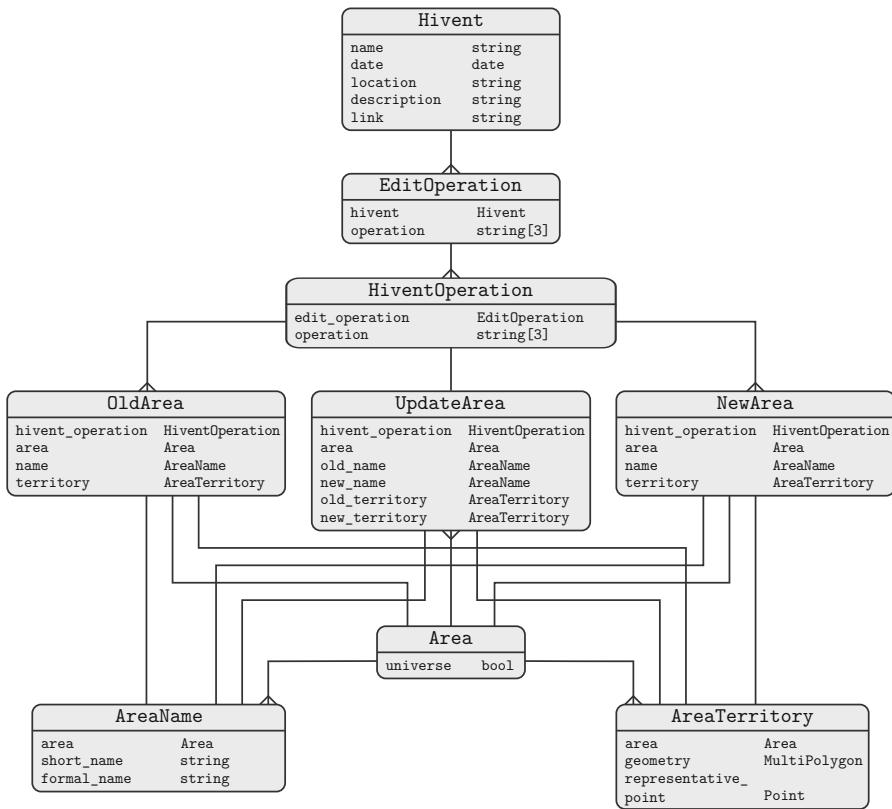


Figure 3.23: The Hivent Database Model

Each entity additionally has an `id` attribute, which is omitted for simplification purposes.

Example Figure 3.24 shows the Hivent Database Model at the example of the German Reunification on 3. October 1990. Before 1990, there were the Areas GDR (“German Democratic Republic”, East Germany) and FRG (“Federal Republic of Germany”, West Germany). A user introduced a Merge operation (MRG) in the Edit Mode between FRG and GDR. The new Area received the short name “Germany” and the same formal name “Federal Republic of Germany” as previous West Germany. Internally, the Edit Mode translates this to an INC of GBDR into FRG and a subsequent NCH of the FRG. One Area ceases, one Area is updated twice and no new Area is created.

Initial Dataset Section 2.1.8 explained the lack of data about historical countries. It is out of the scope of this thesis to create a large testing dataset with the historical countries in the world. The initial dataset consists of the following countries, their names and borders: the 193 UN members and 2 observer states (created by CRE operation) and seven countries with limited international recognition: Kosovo, Transnistria, South Ossetia, Abkhazia, Nagorno-Karabakh, Somaliland and Sahrawi Arab Democratic Republic, see section 5.1.1) (created by DIS operations from their homeland on the day of their declaration of independence).

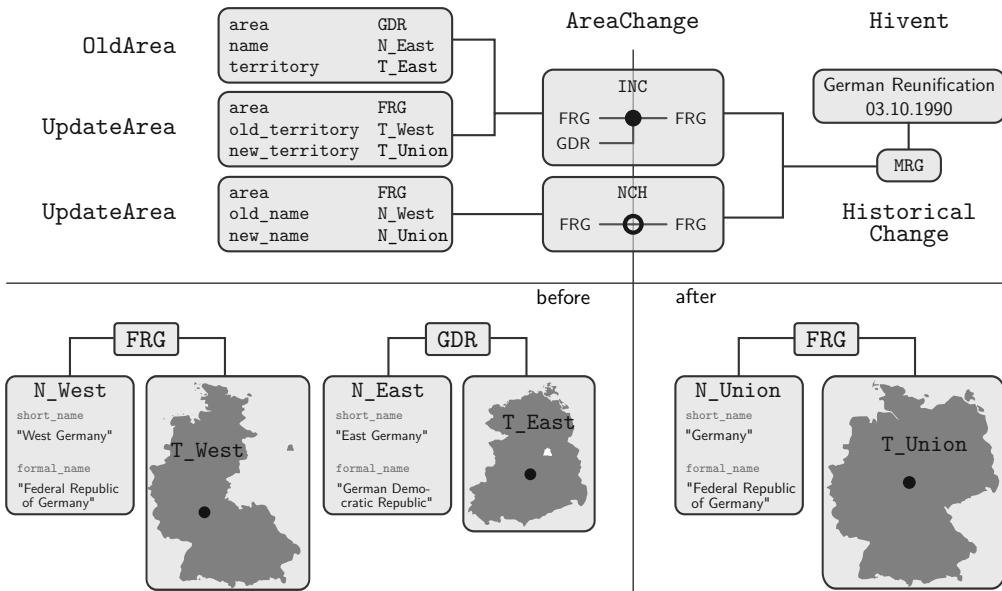


Figure 3.24: Visualization of the German Reunification in the Hivent Database Model

Middleware The Django web framework provides *view* classes as the middleware that receives requests from the client, processes them, queries the necessary data from the database and returns an *HttpResponse* back to the client. In the naive implementation of the system, the middleware provides only two views for the two use cases:

1. **get_all** is initially called by the client side on loading the web service. The server responds to this *HttpRequest* with all data from the database in one *JSON* object. While this behaviour is not scalable, for the initial dataset it was sufficient: The data was loaded in 3.5 seconds.
2. **save_edit_operation** is called by the client after an Edit Operation has been completely created in the Edit Mode. In the last step, the client assembles the relevant data: the associated *Hivent* and *HiventOperations*, data about each *OldArea*, *UpdateArea* and *NewArea*. The view checks the data for integrity and stores them in the database. The method returns to the client a confirmation and a set of final *ids* for the entities stored in the database.

3.4.3 Client-Side Application

The main application of HistoGlobe runs on the client. As introduced in section 2.5, the software is built upon a module system. The modules used in this this implementation of HistoGlobe are emphasised in **bold** in the class diagram in figure 3.25. The classes are structured by their functionality regarding the Model-View-Controller pattern.

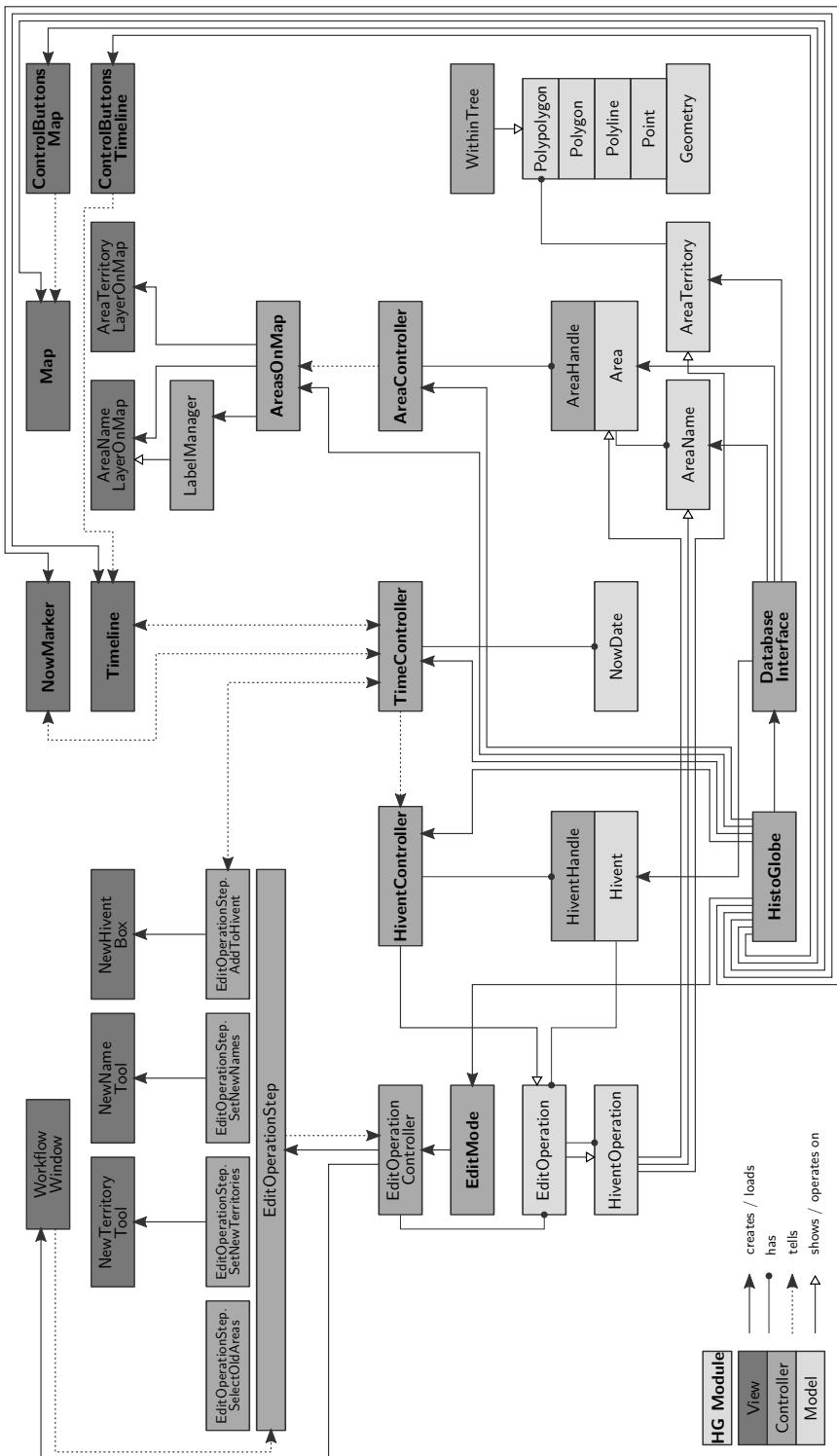


Figure 3.25: Class Diagram of HistoGlobe

Initialization The main HistoGlobe instance in the bottom initializes all modules, mainly the interface elements, the controllers and the DatabaseInterface. This class communicates with the middleware (section 3.4.2), loads data from and stores data in the database. Initially, each Hivent and the related EditOperations and HiventOperations are created. Each HiventOperation is assembled by its associated set of OldAreas, NewAreas and the UpdateArea from the database model in figure 3.23. Afterwards, each Area, AreaName and AreaTerritory is loaded. A double-link is established to their associated HiventOperations via the startOperation, updateOperation and endOperation. Therefore, each HiventOperation knows which Areas, names and territories it creates, updates and ceases – and vice versa each Area, AreaName and AreaTerritory knows which HiventOperation manipulates its evolution.

Executing temporal changes HistoGlobe visualizes time on the interactive Timeline and the static NowMarker showing the current date of the application: the NowDate. Both view classes can manipulate the current date by moving the Timeline or entering a date into the NowMarker. The TimeController stores the NowDate and tells all other modules if the current visualization has changed.

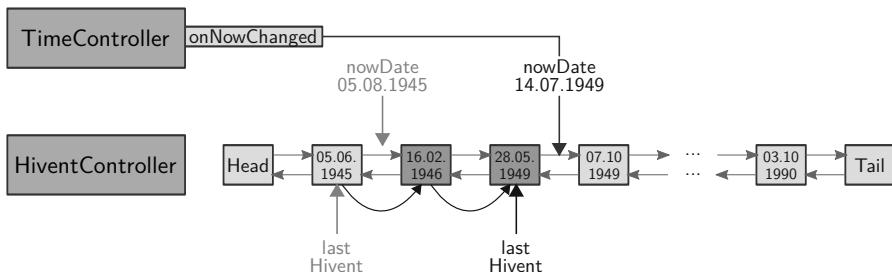


Figure 3.26: Detecting the next Hivent that happens in the HiventController

The core of the Hivent-based implementation is the HiventController. Figure 3.26 illustrates how it works: The controller stores a reference to each Hivent chronologically in a doubly-linked list, i.e. each Hivent knows the historically next and previous Hivent. Additionally, the controller stores a pointer to the last Hivent in the list that has happened and a copy of the current date. It listens to the TimeController – if the NowDate changes, the HiventController checks for the next Hivent if it has happened: The controller compares this new date with its current date checks if the Hivent.date is in between these two dates. If this is the case, the Hivent happens and all the EditOperations associated to this Hivent are executed on the map. The HiventController updates the pointer to the Hivent and checks for the next one until the next Hivent is outside this time span. If the nowDate from the TimeController is before the current date of the HiventController, it checks for Hivents backwards through the list and executes all the EditOperations backwards. This simple data structure allows to effectively and efficiently manage temporal changes of Areas on the map. On initialization of the system, after the DatabaseInterface loaded all the data, the HiventController starts this process: All

Hivents in the list from the beginning until the NowDate of the TimeController are happening one after the other.

An EditOperation diverts the execution to all its related HiventOperations. They are the integral objects that change the status on the map. The main part of its source code, the execution function is shown in listing 3.1. As mentioned above, there are two change directions: forward and backward. If an HiventOperation is to be executed forwards, the following three steps happen:

1. For each newArea, the AreaName and the AreaTerritory that the Area has in the moment it gets historically created are associated to the Area. Afterwards, the Area is shown on the map: The AreaHandle associated to the Area has a function that tells the AreaNameLayerOnMap and AreaTerritoryLayerOnMap to be shown.
2. For each oldArea, the opposite happens: the name and territory are detached from the Area and the AreaHandle hides the Area from the map.
3. In the updateArea the AreaName or AreaTerritory is replaced by the newName respectively newTerritory. The update method of the AreaHandle updates the respective layers on the map.

In case the operation happens backwards, oldAreas and newAreas are swapped and the updateArea uses the oldName respectively oldTerritory instead.

```

1  class HiventOperation
2
3  constructor: (data) ->
4      ...
5
6      @oldAreas    = [] # {area, name, territory}
7      @newAreas    = [] # {area, name, territory}
8      @updateArea = {} # {area, oldName, newName, oldTerritory, newTerritory}
9
10     ...
11
12     execute: (direction) ->
13
14         if direction is 1
15
16             for newArea in @newAreas
17                 newArea.area.name =      newArea.name
18                 newArea.area.territory = newArea.territory
19                 newArea.area.handle.show()
20
21             for oldArea in @oldAreas
22                 oldArea.area.name =      null
23                 oldArea.area.territory = null
24                 oldArea.area.handle.hide()
25
26             if @updateArea
27                 if @updateArea.newName
28                     @updateArea.area.name =      @updateArea.newName
29                 if @updateArea.newTerritory
30                     @updateArea.area.territory = @updateArea.newTerritory
31                     @updateArea.area.handle.update()
32
33         else # direction is -1 => backward change
34             # same as above, just each 'new' is replaced by 'old' and vice versa

```

Listing 3.1: Execution of an HiventOperation

EditMode The Edit Mode is the main contribution of this thesis to the HistoGlobe project. Its interface was introduced in section 3.3.3, this section shortly explains its implementation: When the user clicks a button of an Edit Operation in the upper right corner of the Edit Mode interface, internally a `EditOperationCreator` sets up the `WorkflowWindow` in the interface and the first `EditOperationStep` for this operation. Each of the four steps have different tasks. They are introduced in section 3.2.2. The `EditOperationCreator` stores the data for each step (selected Areas, newly defined territories and names) in an object. Each step can access this object and manipulate its content. It was especially difficult to design each action to be fully reversible. For that purpose, an associated inverse of the action was stored in an `UndoManager` that works like a stack. If the user clicks the back button in the Workflow Window, the last action of the stack gets executed. When the last stage of the workflow (`AddToHivent`) is completed, the `EditOperationCreator` assembles the `HiventOperations` from the data gathered in the task. It sends it along with the associated `Hivent` to the server and finishes the operation.

Within-Tree One particular problem of the territory of an Area is that the associated polypolygon can have holes to account for enclaves and exclaves. They can even be nested (second-order enclaves, third-order enclaves, etc.), as in the example of Baarle-Nassau and Baarle-Hertog at the border between the Netherlands and Belgium⁶. The NewTerritoryTool has to ensure that the drawn polygons are not self-intersecting and that no two polygons for one territory partially overlap each other. If they fully overlap, then they are holes in the polygon. A polygon consist of one *outer ring*, a closed polyline forming the boundary of the polygon, and a set of *inner rings*, closed polylines defining the holes in the polygon. Second-order enclaves are new polygons that happen to be positioned inside the inner rings of the other polygon. They can themselves have inner rings, which represent third-order enclaves, etc.

In order to supported nested holes, the *WithinTree* is introduced. The idea of the tree is to set up an hierarchical structure of polygons that contain each other. An example Within-Tree for a random set of polygons can be seen in figure 3.27.

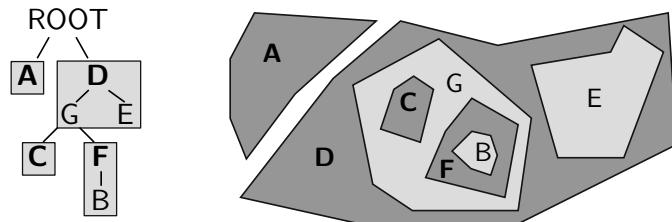


Figure 3.27: The Within-Tree (left) for the set of polygons (right)

The algorithm for inserting a polygon as a node into the tree is shown in listing 3.2. After the tree has been set up, the correct structure of polygons can be obtained by traversing the tree in the following custom order:

1. Remove the first child *F* of the root node and all its children *C* from the tree.
2. Insert each child of each *C* as a direct child of the root node.
3. Create a new polygon with *F* as the outer ring and each *C* as an inner ring.
4. Repeat until the tree is empty.

⁶ *The Curious Case of Baarle-Nassau and Baarle-Hertog*, kaushik, 06.11.2012, <http://www.amusingplanet.com/2012/11/the-curious-case-of-baarle-nassau-and.html>, last access: 31.05.2016

```

1  insert: (newNode, parentNode) ->
2
3      ## PREPARATION
4      # case 1) newNode also in 1 child of parentNode -> withinChild
5      # case 2) 1+ children of parentNode in newNode -> containChildren
6      # case 3) no hierarchical relation between newNode and any other node
7
8      withinChild = null
9      containChildren = []
10
11     for childNode in parentNode.children
12
13         if newNode.isWithin childNode                      # check if case 1)
14             withinChild = childNode
15             break # no other hierarchical relation to any other child possible
16
17         else if childNode.isWithin newNode                  # check if case 2)
18             containChildren.push childNode
19
20     ## EXECUTION
21
22     if withinChild                                      # case 1)
23         @insert newNode, childNode
24
25     else                                                 # cases 2 and 3)
26         # newNode is not in any child of parentNode => place it underneath
27         @_nodes.push newNode
28         newNode.setParent parentNode
29         parentNode.addChild newNode
30
31         for containChild in containChildren              # case 2)
32             # => detach from parent node and place them as children of newNode
33             containChild.setParent newNode
34             newNode.addChild containChild
35             parentNode.removeChild containChild

```

Listing 3.2: Insertion of a polygon node into the Within-Tree

LabelManager A major visualization problem that was sufficiently solved is the label collision problem: Each active Area has both a territory and a name that should be shown on the map. Since no territories can overlap (precondition 5), they can all be shown. This is not true for the names of the Areas: The short_name of the AreaName is placed as a label in the representative_point of the AreaTerritory. Labels can overlap, because they can extend beyond their territory. To avoid this, some labels have to be hidden. A LabelManager decides for each label if it is shown or hidden. Each label gets an additional set of attributes:

- `isVisible`: status variable if the label is shown or not
- `priority`: the “importance” of the label determined by the size of the territory
- `boundingBox`: width and height of the text plus 5 pixel padding
- `coveredBy`: a list of higher-priority labels that cover this label
- `covers`: a list of lower-priority labels that are covered by this label

Label *A* covers label *B* if their bounding boxes intersect and *A* has a higher priority. The labels are stored in a doubly-linked `labelList` in a descending order by priority. The algorithm is based on the following heuristic: *A label is shown unless it is covered by an higher-priority label.*

When a new label is supposed to be shown on the map, it is inserted into the `LabelManager` like this: The correct position of the label in the `labelList` is found by checking with each element in descending priority if they overlap and if the priority is still higher. As soon as the first label with a lower priority is found, the new label is inserted before this label in the list. If there was a higher-priority label before that covered it, the new label is hidden – else it is shown. In the latter case all lower-priority labels are checked if they are covered by the new label – if so, they are hidden.

If an Area ceases also its name is hidden from the map. Additionally, the old label is removed from the `labelList`. Each label that was previously covered by the old label is not covered by it anymore. If no other label is still covering it, the label can be shown now.

If the user zooms the map, the `LabelManager` has to update the visibility status of each label. Zooming in means that each label has more space to its neighbors. No label has to be hidden, but a hidden labels can be shown if it is not covered by any other label anymore. Vice versa, if the user zooms out, the labels have less space to their neighbors. No label can be shown now, but a visible labels needs to be hidden if is covered by at least one other label now.



Figure 3.28: The resulting labels on the map in Europe 2016

Figure 3.28 shows the result of the `LabelManager` on the map of Europe in 2016. It is obvious that no two labels collide which was the main motivation for the algorithm. Also, the labels of the large countries Ukraine, Poland, Germany, France and the United Kingdom are shown. However, the label “Czech Republic” is hidden, because its bounding box intersects with the label “Germany”. On the other hand the labels of Monaco and Andorra are shown, although they are rather insignificant. But since there is enough space around them, they are shown. The `LabelManager` sufficiently serves the purpose of this thesis.

transition to next chapter

Chapter 4

Evaluation

MECE principle The MECE principle – mutually exclusive and collectively exhaustive – is used by the consulting company McKinsey for organizing large amounts of data and as a strategy for effective problem solving. The advantages of a MECE model are [For]:

- Each possible case in the real world can be mapped to a case in the model, because the model covers all possibilities (*collectively exhaustive*).
- A case in the real world can be expressed by exactly one case in the model, because there is only one possibility (*mutually exclusive*).
- The model is logical and comprehensive, can easily be understood and followed.

Mutual exclusion are UNI \cup INC and SEP \cup SEC really mutually exclusive?

idea: express UNI as INC + SEC at the same time, just like border change, but leaving no territory for one of the areas $= \cup$ no SEC into Ω but directly into other Area. Would that work? If so, could we simplify everything by having only three Hivent Operations (INC, SEC, NCH) ?

First it is to be shown that one operation can not be equivalently expressed by a combination of any other operations. This is obviously true for CRE, because all other operations require at least one old area as an input to the operation. Vice versa, CES is unique, because it is the only operation without any new areas. ICH could geographically be represented by a combination of CES and CRE, but that would not create a historical relationship between both Areas. Since the other identity-changing operations require either multiple old or new areas and the last three operations are identity-preserving, ICH is also unique.

`UNI`, `INC`, `SEP` and `SEC` require either old or new areas and establish historical relationships by changing identities. That is why they can neither be replaced by `CRE`, `ICH` and `CES` (only one old and/or new area), nor by `NCH`, `BCH` and `TCH` (identity-preserving). It is trivial that no operation can be expressed by its inverse and an operation that requires one old area can not be replaced by one that requires multiple and vice versa. Therefore, the only possible combinations left are `UNI` \leftrightarrow `INC` and `SEP` \leftrightarrow `SEC`. While geographically, they are equivalent, because they unite respectively separate the territory in the same way, they are historically distinct: While in `UNI` and `SEP`, no Area is preserved in the operation, `INC` and `SEC` represent one Area that incorporate one Area into respectively cede one Area from its own territory. This shows the mutual exclusion of all identity-preserving operations.

It has already been argued that identity-preserving operations can not be expressed by a combination of any identity-changing ones. Also, `NCH` changes the name, whereas `TCH` and `BCH` manipulate the territory of an Area, so it is clear they can not replace each other. By intuition, `BCH` is the same as two `TCH` of both Areas affected by the `BCH`. Both operations do also not set up any historical relationship, so they are historically the same. However, geographically, two `TCH` of two neighboring countries would be redundant, since the territory ceded by one Area is exactly the same territory that is incorporated by the neighbor. Therefore it has been proofed that all operations are mutually exclusive.

Exhaustive collection Next it needs to be shown that all cases that can happen in the real world can be expressed using a combination of one of the ten HG Operations. The first aspect is the identity of an Area, representing a political entity in the real world. In the life cycle of an entity, it is established at one point t_s , its name and territory can change multiple times while being active $U : \forall t_u \in U : t_u > t_s$ and it ceases at some other point $t_e : t_s < \forall t_u \in U < t_e$.

In the real world, a political entity can be created in three ways:

1. If before the creation of the entity its initial territory was fully unclaimed, it does not have any historical predecessors and is created new. This is represented in the `CRE` operation.
2. If its initial territory was fully claimed by a set of entities, then all of these entities are historical predecessors.
 - (a) If the entity originates from itself by changing its formal name, the territory remains unchanged. The `ICH` operation reflects that case.
 - (b) An entity can also originate from one entity that has dissolved into several subsequent entities, which is represented in the `SEP` operation.
 - (c) Finally, an entity can originate from several entities unifying. The `UNI` operation models this case.
3. If the new entity's territory was partially claimed and partially unclaimed, this process of creating entity A can be expressed by a combination of three operations:

- (a) CRE creates the temporary entity A_T with a new name and its territory on all unclaimed land that shall be occupied by A later.
- (b) The rest is currently territory of a set of entities B . For each entity $B_i \in B$, the part that shall be territory of A gets ceded from B_i with a SEC operation, creating a set of entities A_R . This operation establishes a historical relationship between B_i and A_i .
- (c) A_T and all $A_i \in A_R$ are unified with UNI to the final entity A . A inherits its name from A_T and each Area $B_i \in B$ as a predecessor.

Throughout the lifetime of a political entity, the following changes can happen to it:

1. The entity can change its name. A change of the commonly known short name is represented by NCH and preserves its identity. A change of the long official or formal name creates a new Area (ICH).
2. The territory of the political entity can change.
 - (a) If it expands into land that is not claimed by any other entity at this time point or if it is shrinking without influencing the territory of potentially neighboring entities, the TCH operation can be used.
 - (b) If the entity incorporates a territory from or cedes a territory to one neighboring entity, then this change is modeled by a BCH operation.

is that historical relationships must always be established in both ways, i.e. $A \rightarrow_H B \Leftrightarrow A \leftarrow_H B$. There are five operations that set up an historical relationship and for all of them this is true. Regarding the Area name, it must be

name: no problem, can overlap territory: by precondition: can not overlap = \perp geometrical and topological integrity

investigate for each operation if it maintains integrity CRE ICH CES UNI, INC, SEP and SEC operate solely on NCH BCH TCH

compare 5 HG operations with temporal operations in History Graph Model

view

transition to extensions

-*i* Managing Vagueness, Uncertainty and Granularity in Spatial Information Systems (VUG) -*i* Karl Grasser (Diss. Santa Barbara) -*i* Fuzzy, Imprecise, probabilities vs. possibilities

big problem: why? intention and motivation of author? hard to find out... voice and perspective
medieval maps: natural landmarks as border points =*i* inaccurate and imprecise perspective: who is making the map? (illiterates?) different names: US Civil War (North) vs. WWI (West) vs. Germanic War (Russia) WWII (West) vs. Great Fatherland War (Russia)

accepted uncertainty: date != exact timepoint, only D.M.Y location != exact location, only name of place

[Sol14, chapter 2, p. 51]

think about how to represent historical knowledge in geographic context degree of certainty -*i* ironically: that has to be exact as well in a database table =*i* reason: careful conclusions from historical maps

country borders coastlines interior disputed territories situation: n fully recognized countries and m non or partially recognized entities claim sovereignty over 1 territory territory is surrounded by disputed border question: does this disputed area claim sovereignty? ¹<http://www.economist.com/blogs/economist-explains/2014/09/economist-explains-1> uncertain borders situation: n fully recognized countries commonly agree on a boundary between them, but the border is not clearly defined / fuzzy / uncertain states of borders planned agreed demarcated provisional valid vs. disputed

borders: complex model: different states of boundaries: draft -*i* proposal -*i* dispute -*i*

The model is open to future extensions to account also for geographic changes and international sea borders.

However, the first question arises regarding the relevance of the location: While the exact position of the battlefield of Verdun or the place where John F. Kennedy was assassinated might very relevant to the event itself, the location of a governmental bill, a declaration of independence or a border convention might not play an important role and usually happens in a representative place, e.g. the parliament or the office of a president. In a lot of cases, it is much more important which territories an event actually influences instead of where it happened.

Another constraint of the model is that it does only support coequal Areas and no hierarchies, e.g. a country consists of a set of states which consist of a set of counties. Also, independent Areas that overlap other areas, e.g. to visualize a disputed zone or the expansion of the rain forest, are not possible given the model.

The model can easily be extended to states, provinces or regions. Therefore, from now on the term

¹\unskip\penalty@M\vrulewidth{z@height}{z@depth}\dpff

political unit instead of country is used to describe the object in the real world that is modeled by an Area in the system.

Hierarchical Areas CTR -i STA -i CTY -i CIT each level one layer aggregate geometry upwards

Overlapping Areas e.g. war zone, independent layer

border change: manipulate border, not territory

For precision purposes it would be important to be able to import existing geometries from external sources or to import an historical map and extract the territory from there. However, this was not in the scope of this thesis and has to be integrated in the future.

Chapter 5

Uncertainty

Every aspect of the development chapter 3 of this work is based on the prerequisite of full certainty of the data. That means both the Historical-Geographic Operations and the Hivent-Based Spatio-Temporal Data Model assume that the dates of the historical events, the names and territories of the historical and current areas and the historical relations between events and areas are accurate and reasonably precise (definitions see 5.2).

However, this assumption is far from valid. In historical research, uncertainty is one of the major problems (see 2.1.1) a historian has to deal with on a daily basis: sources, even primary sources, can be biased towards the author of the source, information can be imprecise or even inaccurate and information can be conflicting with other sources. This chapter explains problems with uncertainty in the domain of evolution of countries in time and space and develops approaches to deal with these problems.

5.1 Definition of a Country

The problem begins with the definition of a term that almost everybody in the world is familiar with: a "country". Since countries are the domain of this historical geographic information system, it must be possible to decide for each current and historic territorial entity in the world if it is or was a country or not. Therefore a clear and non-conflicting definition of a country is necessary. However, this is impossible to do.

The Oxford Dictionary definition of a country reads as follows:

"The *territory* of a *nation*; a *region* constituting an *independent state*, or a *region*,

province, etc., which was once independent and is still distinct in institutions, language, etc.”¹

This definition includes many different concepts and terms: the territory or region that the country is on, a nation or state, a population and a culture of the territory in terms of institutions or languages. While nation and state are commonly used as synonyms for countries, their meaning varies from case to case, as it will be examined in this section.

To understand what a country really is, the United Nations as an intergovernmental organisation are a valuable source. It was founded after World War II (October 1945) and promotes international peace keeping, security, protection of human rights or humanitarian aid to all its member states which should coincide with all the countries in the world. The committee currently has 193 full member states and two permanent observers: The Holy See (Vatican City) and the State of Palestine [Unib]. But these 195 members in total do not cover all places in the world – and also a membership in the United Nations does not mean that the question of statehood can simply be answered.

5.1.1 Special Cases

Examining the list of the UN member states yields several interesting observations and special cases, which can be classified by their membership status in the United Nations and their degree of international recognition.

UN observer states The *Holy See* is the juridical and spiritual entity representing the territory of Vatican City. It is a fully recognized and sovereign state but is not a full member of the UN, because it has never applied for it. It is the by far smallest sovereign state in the world (0.44 m^2), is an enclave inside the city of Rome with a population of only 800 people, including 30 women [Vat].

The *State of Palestine* has a population of 4.8 million people [Pal, as of 2016] and is also an UN observer state. However, it is totally different in terms of sovereignty: While it consists of the territories of the West Bank, East Jerusalem and the Gaza Strip, their borders were drawn in the 1949 Green Line Armistice Agreement but were never intended to be used as international boundaries [Amn]. Since then, the ongoing and complex conflict with the State of Israel lead to a difficult situations regarding the sovereignty over the territories. Therefore, the state has no clearly defined territory. Moreover, while 114 states officially recognize the Palestinian state, almost all current main economic powers do not, including Canada, France, Germany, Italy, the United Kingdom and the United States. None of them even voted in favor of Palestine receiving an observing status

¹country, n. and adj., Oxford English Dictionary, URL: <http://www.oed.com/view/Entry/43085?>, last access: 2016-04-25

in the UN [Unia]. That means, unlike the Holy See, Palestine is not a fully sovereign and recognized state.

UN non-members with limited recognition Kosovo is a state Europe and declared independence from Serbia in 2008. It has a clearly defined territory and a permanent population and is recognized by 111 UN member states. In order for Kosovo to become a full member of the United Nations, all permanent members of the security council (United Kingdom, France, Russia, China and the United States) must agree. But since Russia and China strongly support the territorial integrity of Serbia, they would veto Kosovos membership in the United Nations. Therefore, Kosovo is not even an observer state of the United Nations, although having about the same degree of international recognition as Palestine [Peo].

The status of Taiwan is a very complicated issue. An overgeneralized description of the problem, which involves two territories and two political entities, is: There is the *People's Republic as China* (commonly known as China), with full control over mainland China, and the *Republic of China*, governing the island of Taiwan. However, both political entities claim each others land. That means, there are two states claiming the exact same territory. But, since 1971 the People's Republic of China is the representative of whole China in the United Nations, including the island of Taiwan. Because it is part of the Security Council, it successfully vetos membership requests of the Republic of China. Therefore, it can not be a member of the United Nations, although it operates like an independent country by international standards: They have an own jurisdiction, issue own passports and have unofficial diplomatic relations to most countries in the world. But officially, only 22 member states of the United Nations uphold diplomatic relations to Taiwan [Rep]. To all of these states the People's Republic of China does not have any diplomatic relations, which makes also them an only partially recognized state.

There are other non-member states of the United Nations which have not yet gained broad international recognition: the Sahrawi Arab Democratic Republic (recognized by 84 UN member states [Wes]), Abkhazia (6 [Glo]), South Ossetia (5 [BBCc]), the Turkish Republic of Northern Cyprus (1 [Leo15]), Nagorno-Karabakh Republic (0 [BBCa]), Transnistria (0 [Gut14]) and Somaliland (0 [BBCb]).

UN members with limited recognition In addition to the Republic of China, there are five other member states of the United Nations that are not fully recognized by all other UN members: Armenia (not recognized by Pakistan [Tod]), the Republic of Cyprus (not recognized by Turkey [Eur]), North and South Korea (officially Democratic People's Republic of Korea and Republic of Korea, mutual non-recognition [Dav]) and the State of Israel, which 32 UN member states do not recognize [Isr].

Special Territories Additionally to countries gaining for international recognition there are territories belonging to fully sovereign countries with a varying degree of sovereignty. For example Greenland

is an autonomous country within the Kingdom of Denmark, but not a sovereign state and therefore not a member of the United Nations. The same applies to the Faroe Islands (part of Denmark) and numerous overseas territories of the United Kingdom, the French Republic and the Kingdom of the Netherlands in the Caribbean, the Indian Ocean or the Southern Pacific Ocean. Moreover, there are five quasi-independent countries in a so called *Free Association*: Niue and Cook Islands are associated to New Zealand and not part of the United Nations. The Marshall Islands, the Federated States of Micronesia and Palau are associated to United States, but in contrast are full UN members [Won].

This incomplete and simplified list of special cases manifests the big problem that is associated with the terms "country", "state" or "nations": There is neither a *de jure* consistent definition nor a *de facto* consistent usage of these terms. Everything breaks down to two different concepts:

5.1.2 Declaratory vs. Constitutive Theory

The declaratory theory, established in the Montevideo Convention 1933 [Yal], gives each entity the right to declare a state if it matches all of the four requirements:

1. a clearly defined territory
2. a permanent population
3. a political representation / government
4. the *capacity* to enter diplomatic relations

These four requirements make sure that a state can exist physically and politically. However, it is worth noticing that this definition does not include any actual diplomatic relations to other states, but only the capacity to enter them. Therefore the existence of a state is independent from its recognition by other states. In other words: "A country is a country when it thinks it is a country."

In contrast, the constitutive theory requires exactly that: A state can only be considered as such if it is recognized by other states. However, it is not defined anywhere by how many other states [Law]. In short: "A country is a country when other countries think that country is a country." [CGP]

Both theories have advantages and disadvantages, but the two main problems are:

1. Following the declarative theory, countries are self-classifying and potentially conflicting entities. The application of this measure would grant Kosovo, the Republic of China, Abkhazia or the Sahrawi Arab Democratic Republic full statehood. However, since their territories are contested, this would lead to overlapping territories with Serbia, China, Georgia and Morocco, which is impossible.
2. There is no superior organization that can judge if a country is a country or not. Even the United Nations fail to do so, because their membership requirements prevent states like Kosovo

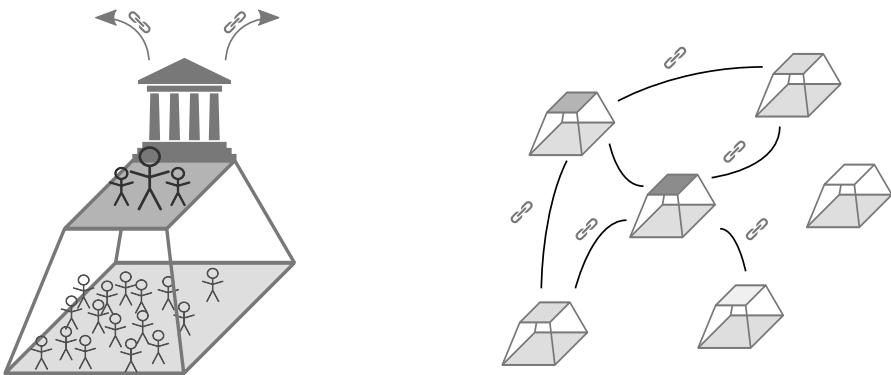


Figure 5.1: The Declaratory Theory (left) and the Constitutive Theory (right) of Statehood

or the Republic of China from becoming full members. They also have no power to rule out problems regarding the independence of Transnistria or Somaliland.

Therefore it is impossible to objectively classify an area as a country or not: nobody can say if Kosovo, the State of Palestine or Niue are countries or not. These theories have been introduced in the previous 80 years. For the time before that, a conflict-free decision of what is a country is not just impossible, but also not justifiable because of a lack of jurisdiction.

That means, an historical geographic information system with the goal to visualize the development of the countries on Earth in time and space inevitably deals with uncertain information that certain parties see as wrong. Its data model can not perfectly fit self-classifying data and can not rely on an objective data source. The system has to contain approaches that deal with this problem.

5.2 Types of Uncertainty

In order to understand different types of uncertainty it is important to understand the concepts of *disagreement*, *precision* and *accuracy*.

The model in an information system tries to resemble the real world as good as possible and necessary – in this case the history of countries. If there is already a conflict in the real world, e.g. the Kashmir region which is claimed by both India and Pakistan as part of their territory, then this is a *disagreement* which also has to be properly modeled as such in the system.

The better a model simulates the reality, the more *accurate* or correct it is. That means, the closer it gets to the target, the higher is the accuracy. *Precision* or exactness describes how similar the results are compared to each other, independent from the distance to the target. That means a precise model gets the same results over and over again (see figure 5.2).

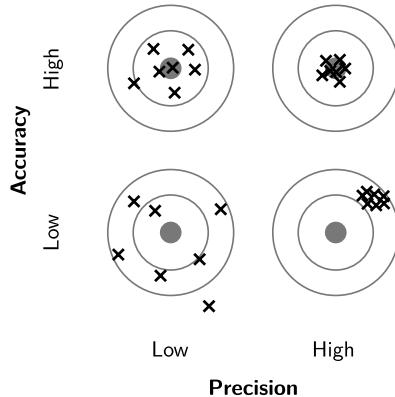


Figure 5.2: The difference between accuracy and precision

If the border between the Principalities of Transsylvania and Wallachia is deducted from an historical map of 1600, the course of the border is inaccurate to a certain degree, because the map does not show the real world correctly. However, it can be modelled in the system very precisely, because the coordinates of the border points are stored as floating point numbers in the data model. In contrast, there is currently no agreement upon territory of Palestine, although the different versions can be modelled very precisely. In order for the model to also be accurate in this case, it would need to support contested territories.

Hereafter the current data model introduced in section 3 is evaluated in terms of accuracy and precision.

Hivents The model for historically significant happenings contains only the following meta information: name, date and location of the event. This has several shortcomings in terms of precision:

The name of an historical event can have different versions: a long, official version and a short common version. The commonly known “Treaty of Versailles” (1919) is officially called the “Treaty of Peace between the Allied and Associated Powers and Germany”. Also the name is different in other languages affected by the treaty. Additionally, there can be different versions of the name from different perspectives, even within the same language, e.g. the “American Civil War” as it is known today was alternatively called “War Between the States”, “War for Southern Independence” or “War of Northern Aggression” depending on the perspective. The `Hivent` model does not account for different languages and versions and is therefore not very precise.

The `Hivent.date` is supposed to represent the temporal dimension of an historical event. While an historical change itself is discrete and happens at exactly one time point, the historical event yielding this change might not. The “Congress of Vienna” which reordered the empires on the European mainland was one of the main historical events in modern European history. While the changes of the congress came into effect on 9. June 1815, the congress itself took place in Vienna from September

1814 until June 1815 which is also a timespan of interest. Another phenomenon becomes apparent in the “Convention for the Extension of Hong Kong Territory” (1898) which had a predefined length of 99 years. The treaty therefore has two dates in which historical changes happened: the date the treaty came into effect (Hong Kong becomes part of the United Kingdom) and the date it stopped being in effect (Hong Kong is handed over to China). Other interesting aspects are different calendar systems used in different parts of the world throughout history: the October Revolution in Russia (1917) happened in November in their Gregorian Calendar system, but in October in the Julian Calendar. Also timezones can play a crucial role: The German Instrument of Surrender ending World War II in Europe came into effect on 8. May 1945 at 23:01 Central European Time, so the 8. May is celebrated as the Victory Day in Western Europe. But in the Soviet Union and nowadays Russia that happened at 1:01 Moscow Time on 9. May 1945 which is why the celebration of the Victory Day there happens one day later. While the `Hivent.date` field in the data model works with timezones, it does not support different calendar systems or multiple dates associated with one `Hivent` which limits its precision.

The event location is represented by the `Hivent.location` name of a place, which can e.g. be a city, a battlefield or a region. The model is not very precise, because the actual geospatial location or region in which an historical event happened is not stored in the system. Additionally, it does not support names in different languages.

The even larger problem is an integral lack of accuracy: The whole nature of historical research is based on subjective interpretation of supposedly objective primary sources. But it is questionable if a source can actually be objective. Each bill, treaty or speech is written by somebody, each map was drawn by someone and has therefore a subjective note. Information in a primary source can be (un)intentionally incomplete, imprecise or inaccurate. The source can be biased towards the author, can contain secret passages not open to the public or its geographic information might be wrong. There are many problems involved in historical sources which makes the acquisition of objective historical data almost impossible. The further documents go back in time, the lower is the expected accuracy. Since all the information in the historical geographic information system is based on primary sources, the data in the system inherits these problems.

Areas Also the model of an abstract area, consisting of a territory and a name, is problematic in terms of accuracy and precision. As it has been discussed in subsection 5.1 in detail, it is impossible to objectively model all areas free of conflicts. But the current model does not support the status of a territory as being contested. Also, countries can be part of other autonomous (constituent) countries, like England is part of the United Kingdom of Great Britain and Northern Ireland or Greenland is part of Denmark. However, the data model does not support different levels of sovereignty, autonomy or international recognition.

The `AreaName` has the same problem than the `Hivent.name`: it differs among the languages or even among cultures using the same language. The model does not support that. But in one aspect

it is more precise than for historical events, because it contains both the formal and the short name of a country.

More problematic is the `AreaTerritory`: Areas bordering international water have a constant coastline assuming that it has never changed. This is inaccurate, because coastlines gradually change all the time, therefore also the boundaries of the countries. The data model does support neither that nor international sea borders which are parts of a countries territory. The primary source for territories of countries are historical maps. They show the status of a country at one point in history or sometimes a territorial change. The process of extracting a boundary from an historical map is error-prone and yields to a loss of accuracy in each step on the way: digitizing, georeferencing and contour tracing. The level of inaccuracy depends on the resolution, the map projection and the colors used in the map. In the data model it is not possible to provide information about the expected accuracy of a territory.

Another problem is that the territory is stored as a whole polypolygon. Different parts of the border can have a different status, e.g. one part is a sea border, one is a well-established and demarcated border to neighboring country X and another part is a contested border to neighbour Y. The `AreaTerritory` data model does not account for these differences.

Accurately modelling contested territories is also problematic. It is based on the principle that there can not be overlapping territories at the same time. That means, a contested territory, for example China or occupied territories in the State of Palestine by the State of Israel can only exist once at the same time and therefore have to be treated specially. But the data model does not support contested areas. To go even further, it is questionable which areas should be included in the data model and which not. While it seems obvious to have Spain, Saudi-Arabia and Azerbaijan in the system, the question of whether or not to include the State of Palestine, Abkhazia, Somaliland or micronations like the Conch Republic in the Florida Keys is hard to answer.

Overall, the current data model poorly accounts for different levels of uncertainty in historical geographic information: imprecise and inaccurate sources, different viewpoints and interpretations, contested territories, changing coastlines or different languages. The question of the upcoming subsection is: How can the data model be extended in order to be more accurate and more precise?

5.3 Solution Approaches

In summary, the shortcomings of the current concept are:

1. General
 - (a) only one language (English)

(b) constant coastlines

2. Hivent

- (a) only one historical perspective on the Hivent name
- (b) only one discrete Hivent date
- (c) only one calendar system (Julian Calendar)
- (d) only location name, no connection to the map

3. Area

- (a) only one historical perspective on the Area name
- (b) all Areas on the same level (no dependencies)
- (c) no support for non-sovereign autonomous regions
- (d) no credibility of Areas existence (via international recognition)
- (e) only clear territories, no support for neutral zones or contested territories
- (f) no support for uncertain parts of a territory
- (g) no support for international sea borders

A higher accuracy in the data model usually leads to a higher complexity. This trade-off has to be thoroughly taken into consideration when supporting a new feature to make a model more accurate. This is why the following problems will be ignored in the rest of the thesis:

- 1b) Coastlines change continuously, therefore the Hivent-Based Spatio-Temporal Data Model is not suitable. A support for coastline changes would require another data model applied to coastlines. This is out of the scope of this thesis. One approach is to model international waters just like any other area with a name and a territory and change the boundaries according to an underlying continuous function. This way, the countries sharing that coastline as their international border would change likewise.
- 2a) The support for different historical perspectives on the same event, e.g. different names and descriptions or even different historical changes would create a research tool with great potential. It would enable the possibility for different versions of history based on alternative scenarios ("What if X would have (not) happened?"). However, this would significantly increase the complexity of the system and would also be very subjective.
- 2c) The introduction of different calendar systems would not increase the accuracy of the model significantly. The dates in the system must all stick to the Julian calendar, which is a reasonable requirement to avoid unnecessary complexity.
- 3a) see 2a)

- 3g) Currently each country's territory extends in a range of 3 to 12 miles (5 to 20 kilometers) [Uni82] into international waters. While this is important to accurately model a territory, it is complex, because not every country has signed the convention and each signing party can choose their range into international water. This would not just increase the complexity of the model but also create unfamiliar country territories.

In order to tackle the remaining shortcomings of the current concept, both the user interface and the data model have to be extended.

5.3.1 Extension of the Edit Mode

Two new operations (see figure 5.3) are introduced: SCH changes the status of an area and REC declares a new recognition, i.e. one country internationally recognizes another one.



Figure 5.3: Newly designed and extended buttons for edit operations.

Set New Territory Also the edit operation workflow gets changed. The second step (`SET_NEW_TERR`) defines the territory of the new area(s). Instead of drawing the whole territory as a set of polygons, the user draws one borderline at a time, geometrically as a polyline. This has the main advantage that each part of the border is treated separately.

The borderline is assigned a degree of certainty, in the interface controlled by a horizontal slider, in the model as a certainty value ($\text{certainty} \in]0..1]$). Absolute certainty (1.0) creates a sharp and crisp line on the map. In case of uncertainty ($\text{certainty} \in]0..1[$) three different visualization methods are introduced:

1. Blurred Border: The higher the uncertainty, the wider and more blurry the border.
2. Border Corridor: With increasing uncertainty, the offset around the actual border line extends. That creates a corridor in which the actual border is probably in.
3. Blurred Border Corridor: The combination of the first two approaches.

A simple model for the calculation of the blur factor, line width and offset distance is:

$$f(c) = -1 \cdot S \cdot \ln(c) + I$$

where c is the certainty factor, $S > 0$ is a scaling factor and I is the initial value (for width: 1 px, for blur: 0, for offset: 0 px). In the example in figure 5.4 the scaling factor $S = 4$. In the Blurred Border Corridor method, the scaling factor for line width and the blur factor was halved. Further analysis and user testing are required in order to decide for one of the three approaches to be used in the system.

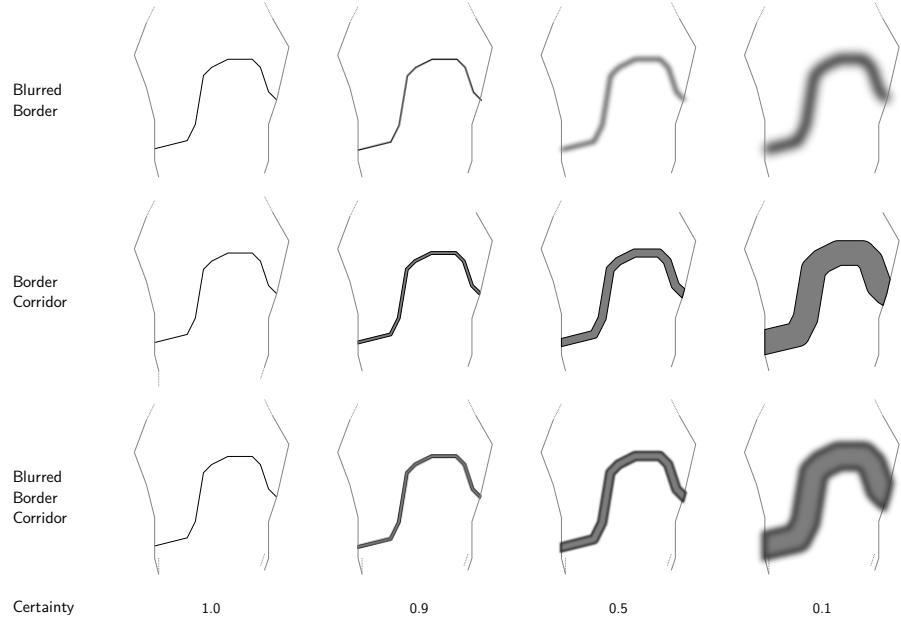


Figure 5.4: Three different methods to visualize uncertain courses of a border

Another advantage of the input of borderlines instead of territories is that once the model is further advanced, coastlines can be continuously changed according to an appropriate change model (see problem 1b). This can be applied solely to the coastlines without affecting the interior borders.

A new border point automatically snaps to an existing border point, if the mouse position is close enough to it (an appropriate threshold might be 5 px). This allows for a smooth workflow and is required to create closed polygons. In case the borderline is closed, it gets treated as a complete polygon and territory. When the user finished a territory by defining all surrounding polylines that create a closed ring, the polygon gets assembled. If a borderline meets another borderline at an interior node, the polyline gets split up into two parts so that each meeting point of borders is the start or end point of a polyline. This way integrity is maintained and each territory compounds of several polylines creating a set of closed polylines: a polypolygon.

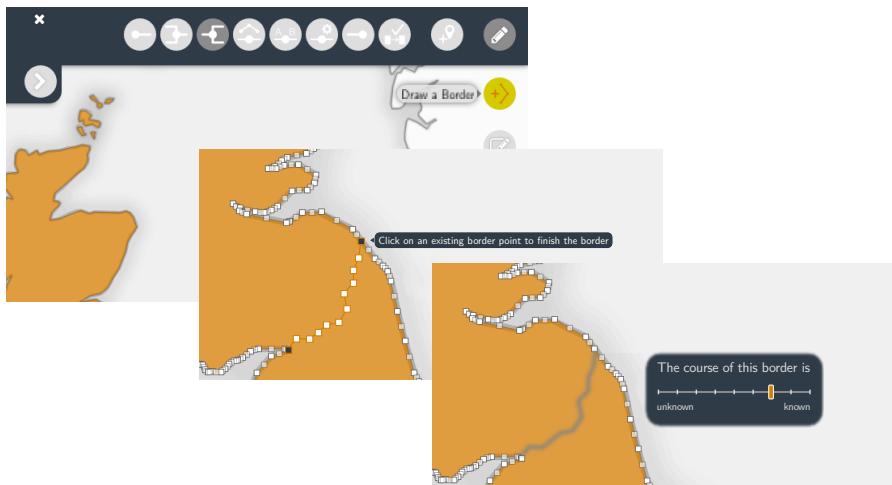


Figure 5.5: Drawing historical borders instead of full areas and defining a level of certainty.

If the created territory overlaps with an existing territory, its intersection will create a separate territory. In the next step, this territory can then be defined as a contested area or defined as a part of another area. If the step yields an empty territory that was claimed before, it can later be defined as a neutral zone or unclaimed land.

Set New Name When defining the name of an area, the user will get actual name suggestions. These result from a collection of current and historical countries from Wikipedia. That saves time for researching short and formal names of areas. In the long run, the system can be synchronized with Wikipedia or even be designed as an extension for Wikipedia articles about current or historical countries.

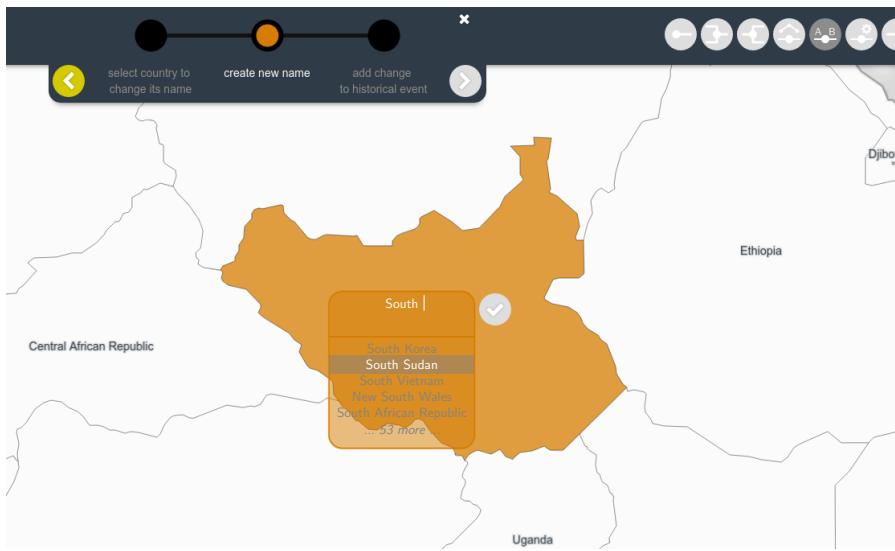


Figure 5.6: Getting suggestions for the name name from Wikipedia.

Set New Status To treat special areas differently, a new step in the edit operation workflow gets defined. After the territory and the name of a new area are defined, a special status can be assigned to it:

1. A *fully sovereign country* is a political entity with full sovereignty over its territory and people and significant international recognition, e.g. Estonia.
2. An *unclaimed land* is a territory that is not claimed by any political entity, e.g. currently Antarctica.
3. A *neutral zone* is often a buffer zone between two conflicting countries, e.g. the UN Buffer Zone in Cyprus.
4. A *contested territory* is claimed by at least two different political entities of the same hierarchical level, e.g. the Kashmir region between India and Pakistan. It is also suitable for areas that have claimed independence from a sovereign country but are not yet recognized as such, making their whole territory contested, e.g. Nagorno-Karabakh (see figure 5.7).
5. A territory can be a subordinate part of another country with a certain degree of autonomy ($\in [0..1]$). Fully subordinate parts of a country, like a US State or a German Bundesland have no autonomy (0). Autonomous countries within another country, like England to the United Kingdom or Greenland to Denmark, receive a certain degree of autonomy ($\in]0..1[$). Full autonomy (1) would mean the territory is a fully sovereign country and the value can therefore not be set in the options.

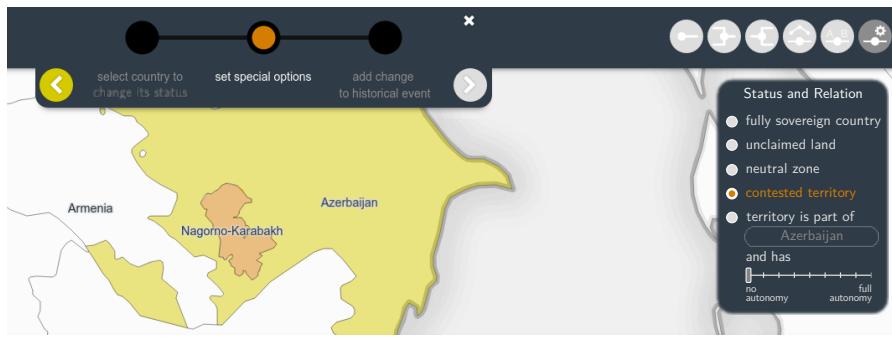


Figure 5.7: Defining a special status or relationship to a territory.

Add Historical Change The visualization of an Hivent gets split up into three parts:

1. An information section storing important meta data of the event location, the dates (timespan in which the event happened), a description and the link to the wikipedia article (if given).
2. A section storing all historical changes associated with that historical event. Each historical change is visualized and is assigned a date at which this event came into effect.
3. A multimedia section stores images, videos, audio files and documents and their sources associated to the historical event.

Similar to the extension of the area name step, also Hivent names can be chosen among a collection of Wikipedia articles describing historical events. Selecting a name from a wikipedia article automatically fills the information section and adds multimedia files from the wikipedia article. The historical change will automatically be entered in the section (see figure 5.8). With this separation, different historical changes at different dates can be associated with one historical events, largely increasing the Hivent data model.

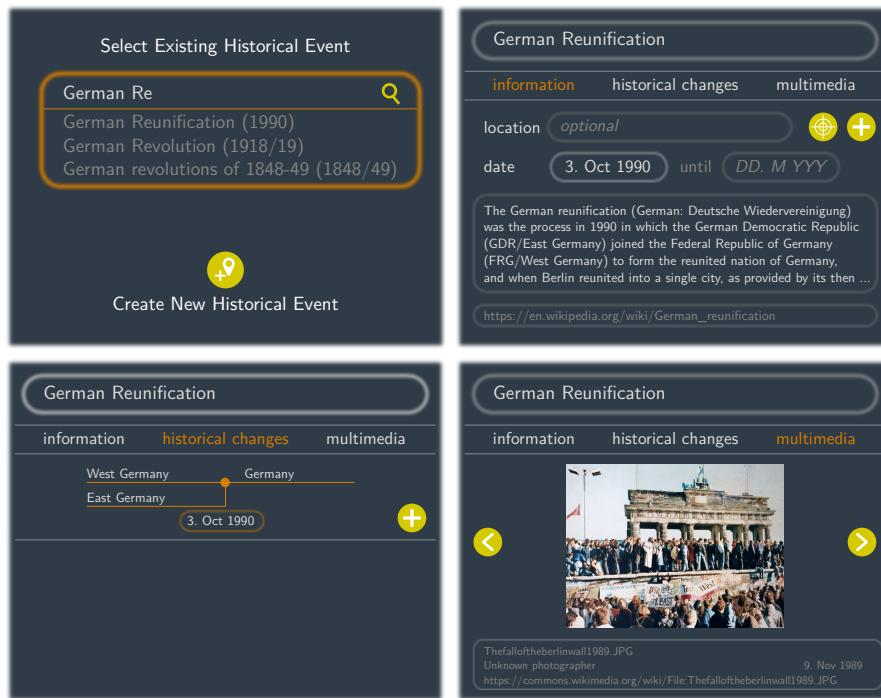


Figure 5.8: Creating a new Hivent and adding the newly created historical change.

New Area Recognition One new operation is to add the recognition of one country by another country. That is simply performed by selecting two areas on the map, whereas the first area recognized the second area. This is an historical change that can afterwards be attached to an Hivent.

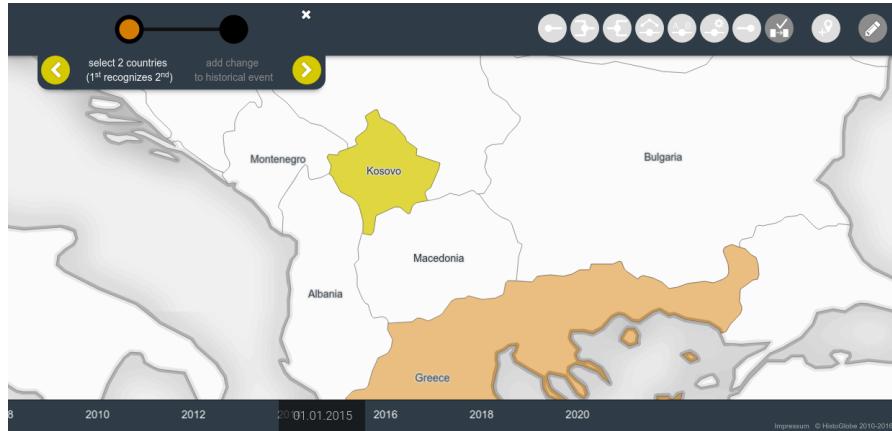


Figure 5.9: New edit operation: Recognition – sets up the recognition of one area to another.

Multi-language support In order to support different languages, a language selection is placed on the bottom right corner of the interface, on the timeline (see figure 5.10). This changes the language of the whole interface and loads the translations of the area names and the Hivent names, locations and descriptions in the newly created language. If a term is not defined in the language, the fallback language (English) is used instead.



Figure 5.10: Changing the language in the user interface.

5.3.2 Extension of the Data Model

To account for the changes in the interface, also the data model has to be adapted. The main changes to the original data model developed in section 3 are:

1. Creation of a `Multilang` entity to store a name of an Hivent, its location or an Area name in different languages.
2. Outsourcing of the `HiventLocation` into an own entity to identify a location with a name and a geospatial reference.
3. Creation of a `Multimedia` entity to manage multimedia files associated to an Hivent.
4. Attachment of a date to an `HistoricalChange`.
5. Inclusion of the `formal_name` into the `Area` model to emphasize it as the identifier of an area.
6. Creation of an `AreaBorder` with a `borderline`. A set of `AreaBorders` create one `AreaTerritory` which is associated to the `Area`. Each change of an `AreaBorder` creates one or two new `AreaTerritory`/ies.
7. Creation of an `AreaStatus` an an `AreaRelation` to account for special status of an area alone or in relation to another area with a certain level of autonomy.
8. Creation of an `AreaRecognition` to account for international recognition of one area to another one.

9. Adaption of the AreaChange entity to model a change of each possible property of an area.

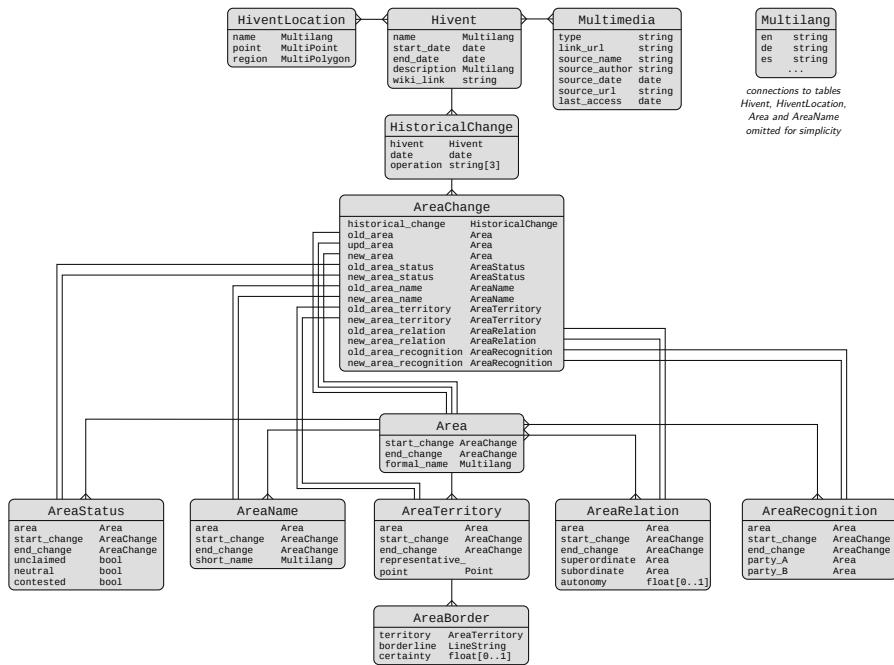


Figure 5.11: The new data model to support the developed approaches regarding uncertainty

Chapter 6

Summary

6.1 Results

Research Questions

6.2 Problems

6.3 Future Work

step further: temporal GIS to narrative GIS

idea: explain history with spatial narratives geographically contextualize events and interactions
organizing principle: time

extend the pure presentational purpose of st data to analytical purpose, e.g. where have most border changes take place in previous 200 years?

Another problem for historians is that they do not necessarily need a tool to better visualize existing knowledge (e.g. historical maps), but to generate new knowledge by analyzing spatio-temporal coherences or distributions in historical data. Spatio-temporal reasoning is still an open field and not easily possible with existing HGIS

[KH08, p. 268], [GG14, p. xii]. space-time premise by Gaddis 2002 time and space equal importance event what significantly has happened and by whom? (singularity!) process how something has

happened? (event+activity = i trigger of process) change driven by process spatiotemporal data defines all above three

extend area model to hierachies (country - i states - i counties/cities)

Bibliography

- [AHA] AHA - American Historical Association (Andrews, Thomas and Burke, Flannery). *What Does It Mean to Think Historically?* URL: <http://historians.org/publications-and-directories/perspectives-on-history/january-2007/what-does-it-mean-to-think-historically> [cited 23 10 2015].
- [All84] James F. Allen. Towards a general theory of action and time. *Artif. Intell.*, 23(2):123–154, 1984.
- [Amn] Amnon Sella. *Custodians and Redeemers: Israeli Leaders' Perceptions of Peace, 1967-79*. URL: http://www.jstor.org/stable/4283114?seq=1#page_scan_tab_contents [cited 26 04 2016].
- [Arm92] C. Armenakis. Estimation and organization of spatio-temporal data. *Proceedings of the Canadian Conference on GIS92*, 1992.
- [BBCa] BBC News. *Nagorno-Karabakh profile*. URL: <http://www.bbc.com/news/world-europe-18270325> [cited 26 04 2016].
- [BBCb] BBC News. *Somaliland profile*. URL: <http://www.bbc.com/news/world-africa-14115069> [cited 26 04 2016].
- [BBCc] BBC World News. *Russia recognises Georgian rebels*. URL: <http://news.bbc.co.uk/2/hi/7582181.stm> [cited 26 04 2016].
- [BCH10] D.J. Bodenhamer, J. Corrigan, and T.M. Harris. *The Spatial Humanities: GIS and the Future of Humanities Scholarship*. Spatial humanities. Indiana University Press, 2010. URL: <https://books.google.de/books?id=8VkhPkIRM4EC>.
- [Bol08] P. Bolstad. *GIS Fundamentals: A First Text on Geographic Information Systems*. Eider Press, 2008. URL: <https://books.google.com/books?id=2rjYPQAAQAAJ>.
- [Bus] BusinessDictionary. *What is a system? definition and meaning*. URL: <http://www.businessdictionary.com/definition/system.html> [cited 04 11 2015].
- [CGP] CGP Grey on YouTube. *How Many Countries Are There?* URL: <https://www.youtube.com/watch?v=4AivEQmfPpk> [cited 26 04 2016].
- [CT95] Christophe Claramunt and Marius Thériault. Managing time in gis: An event-oriented approach. In *Proceedings of the International Workshop on Temporal Databases: Recent Advances in Temporal Databases*, pages 23–42, London, UK, UK, 1995. Springer-Verlag. URL: <http://dl.acm.org/citation.cfm?id=647207.719698>.
- [Dar] Dare Obasanjo. *An Exploration of Object Oriented Database Management Systems*. URL: <https://www.cs.umd.edu/class/spring2002/cmsc434-0101/MUIseum/applications/odms.html> [cited 14 May 2016].
- [Dav] David Scofield, AsiaTimes. *Seoul's double-talk on reunification*. URL: <http://www.atimes.com/atimes/Korea/GA04Dg01.html> [cited 26 04 2016].

- [DISP08] Stanislas Dehaene, Véronique Izard, Elizabeth Spelke, and Pierre Pica. Log or Linear? Distinct Intuitions of the Number Scale in Western and Amazonian Indigene Cultures. *Science (New York)*, pages 1217–1220, 2008.
- [Dra] Nicolae Dragulanescu. Knowledge Map of Information Science: Data, Information, Knowledge. URL: <http://www.success.co.il/is/dik.html> [cited 05 11 2015].
- [Emi] Emil Mikulic, unix4lyfe. *Time*. URL: <https://unix4lyfe.org/time/?v=1> [cited 15 May 2016].
- [Eth07] Philip J. Ethington. Placing the Past: 'Groundwork' for a Spatial Theory of History. *Rethinking History*, 11:465–493, Dec 2007.
- [Eur] European Parliament. *Turkey and the problem of the recognition of Cyprus*. URL: http://www.europarl.europa.eu/meetdocs/2004_2009/documents/nt/553/553930/553930en.pdf [cited 26 04 2016].
- [For] Former McKinsey Consultant. *What Does It Mean to be "MECE" at McKinsey?* URL: <http://workingwithmckinsey.blogspot.de/2014/02/MECE-at-McKinsey.html> [cited 18 05 2016].
- [Fra] Fraczek, Witold (ESRI). *Mean Sea Level, GPS, and the Geoid*. URL: <http://www.esri.com/news/arcuser/0703/geoidlief3.html> [cited 27 10 2015].
- [Fra98] Andrew U. Frank. Different Types of Times in GIS. *Spatial and Temporal Reasoning in Geographic Information Systems*, pages 40–62, 1998.
- [Geo] Geokov. *Map projection - types and distortion*. URL: <http://geokov.com/education/map-projection.aspx> [cited 27 10 2015].
- [GG14] I.N. Gregory and A. Geddes. *Toward Spatial Humanities: Historical GIS and Spatial History*. The Spatial Humanities. Indiana University Press, 2014. URL: <https://books.google.com/books?id=JYQxAwAAQBAJ>.
- [Glo] Global Security. *Georgia mocks Nauru's recognition of Abkhazia*. URL: <http://www.globalsecurity.org/military/library/news/2009/12/mil-091215-rianovostill.htm> [cited 26 04 2016].
- [Gut14] Periparm Gutaj. The foreign policy of counter secession: Preventing the recognition of contested states by james ker-lindsay. oxford: Oxford university press, 2012. 224pp., £50.00, isbn 978 0 199 69839 4. *Political Studies Review*, 12(2):278–278, 2014. URL: <http://EconPapers.repec.org/RePEc:bla:pstrev:v:12:y:2014:i:2:p:278-278>.
- [Häg70] Torsten Hägerstrand. What about people in regional science? *Papers of the Regional Science Association*, 24(1):6–21, 1970. URL: <http://dx.doi.org/10.1007/BF01936872>, doi:10.1007/BF01936872.
- [HW90] Gary J. Hunter and Ian P. Williamson. The development of a historical digital cadastral database. *International journal of geographical information systems*, 4(2):169–179, 1990. doi:10.1080/02693799008941538.
- [Isr] Israel Ministry of Foreign Affairs. *Israel's Diplomatic Missions Abroad*. URL: <http://www.mfa.gov.il/mfa/abouttheminstriy/pages/israel-s%20diplomatic%20missions%20abroad.aspx#note> [cited 26 04 2016].
- [Jan] Jan Werner Müller. *East Germany: Incorporation, Tainted Truth, and the Double Division*. URL: <http://www.oxfordscholarship.com/view/10.1093/0199240906.001.0001/acprof-9780199240906-chapter-9> [cited 18 05 2016].
- [KH08] A.K. Knowles and A. Hillier. *Placing History: How Maps, Spatial Data, and GIS are Changing Historical Scholarship*. ESRI Press, 2008. URL: <https://books.google.com/books?id=VN1v7rzhsQEC>.
- [Kno02] A.K. Knowles. *Past Time, Past Place: GIS for History*. ESRI Press, 2002. URL: <https://books.google.com/books?id=UZy4WiWPepkC>.
- [Lan88] G. E. Langran. A framework for temporal geographic information systems. *Cartographica*, 25, 1988.
- [Lan89] G. E. Langran. Time in geographic information systems. 1989. UMI Order No: GAX90-00269.

- [Law] Law Teacher. *Declaratory And Constitutive Theories Of State*. URL: <http://www.lawteacher.net/free-law-essays/constitutional-law/declaratory-and-constitutive-theories-of-state.php> [cited 26 04 2016].
- [Leo15] Leon Hadar, AntiWar.com. *In Praise of 'Virtual States'*, 2015. URL: <http://antiwar.com/hadar/?articleid=8042> [cited 26 04 2016].
- [Nak] Frieder Nake. Data, Information and Knowledge. URL: <http://viola.informatik.uni-bremen.de/typo/fileadmin/media/DataInformKnowledge.pdf> [cited 05 11 2015].
- [Ora] Oracle. *1970s — Defying Conventional Wisdom*. URL: <http://www.oracle.com/us/corporate/profit/p27anniv-timeline-151918.pdf> [cited 14 May 2016].
- [OS01] T. Ott and F. Swiaczny. *Time-Integrative Geographic Information Systems: Management and Analysis of Spatio-Temporal Data*. Number v. 1 in Time-integrative Geographic Information Systems: Management and Analysis of Spatio-temporal Data. Springer Berlin Heidelberg, 2001. URL: https://books.google.com/books?id=0Yvg_PErYRkC.
- [Pal] Palestinian Central Bureau of Statistics. *Estimated Population in the Palestinian Territory*. URL: http://www.pcbs.gov.ps/Portals/_Rainbow/Documents/gover_e.htm [cited 26 04 2016].
- [PD95] Donna Peuquet and Niu Duan. An event-based spatiotemporal data model (estdm) for temporal analysis of geographical data. *International Journal of Geographical Information Systems*, 9(1):7–24, 1995. URL: <http://dblp.uni-trier.de/db/journals/gis/gis9.html#PeuquetD95>.
- [Peo] People of Kosova. *Who Recognizes Kosova?* URL: <http://www.kosovothankstyou.com> [cited 26 04 2016].
- [Peu99] D. J. Peuquet. Time in GIS and geographical databases. *Geographical information systems*, pages 91–102, 1999.
- [PTKT04] Nikos Pelekis, Babis Theodoulidis, Ioannis Kopanakis, and Yannis Theodoridis. Literature review of spatio-temporal database models. *Knowl. Eng. Rev.*, 19(3):235–274, September 2004. URL: <http://dx.doi.org/10.1017/S026988890400013X>, doi:10.1017/S026988890400013X.
- [Raz12] Ale Raza. Working with spatio-temporal data type. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIX-B2:5–10, 2012.
- [Ren96] A Renolen. History Graphs: Conceptual Modeling of Spatio-temporal Data. *GIS Frontiers in Business and Science*, 2, 1996.
- [Rep] Republic of China. *Ministry of Foreign Affairs*. URL: <http://www.mofa.gov.tw/en/default.html> [cited 26 04 2016].
- [RGS] RGS - Royal Geographical Society. *What is geography?* URL: <http://www.rgs.org/geographytoday/what+is+geography.htm> [cited 23 10 2015].
- [Sol14] A.C. Solana. *Spatio-Temporal Narratives: Historical GIS and the Study of Global Trading Networks (1500-1800)*. Cambridge Scholars Publisher, 2014. URL: <https://books.google.com/books?id=oqIxBwAAQBAJ>.
- [Tod] Today.az. *For Azerbaijan Pakistan does not recognize Armenia as a country*. URL: <http://www.today.az/news/politics/30102.html> [cited 26 04 2016].
- [Unia] United Nations. *General Assembly Votes Overwhelmingly to Accord Palestine 'Non-Member Observer State' Status in United Nations*. URL: <http://www.un.org/press/en/2012/ga11317.doc.htm> [cited 26 04 2016].
- [Unib] United Nations. *Member States*. URL: <http://www.un.org/en/member-states/index.html> [cited 26 04 2016].
- [Uni82] United Nations. *United Nations Convention on the Law of the Sea*, 1982. URL: http://www.un.org/depts/los/convention_agreements/texts/unclos/unclos_e.pdf [cited 28 04 2016].

- [Uot] Uotila, Urho A. (Britannica.com). *geoid — geology*. URL: <http://www.britannica.com/science/geoid> [cited 27 10 2015].
- [Vat] Vatican City State. *Population*. URL: <http://www.vaticanstate.va/content/vaticanstate/en/stato-e-governo/note-generali/popolazione.html> [cited 26 04 2016].
- [Wes] Western Sahara Online. *Proclamation Of The Saharawi Arab Democratic Republic*. URL: <https://wayback.archive.org/web/20080502035045/http://www.wsahara.net/sadr.html> [cited 26 04 2016].
- [WHM90] MICHAEL F. WORBOYS, HILARY M. HEARNSHAW, and DAVID J. MAGUIRE. Object-oriented data modelling for spatial databases. *International Journal of Geographical Information Systems*, 4(4):369–383, 1990.
- [Won] WonderWhy on YouTube. *How many countries are there in the world?* URL: https://www.youtube.com/watch?v=4vsPB_lbo94 [cited 26 04 2016].
- [Yal] Yale Law School. *Convention on Rights and Duties of States*. URL: http://avalon.law.yale.edu/20th_century/intam03.asp [cited 26 04 2016].
- [Yua96a] May Yuan. Modeling Semantical, Temporal, and Spatial Information in Geographic information Systems. *Geographic Information Research: Bridging the Atlantic*, pages 334–347, 1996.
- [Yua96b] May Yuan. Temporal gis and spatio-temporal modeling. In *Proceedings of Third International Conference Workshop on Integrating GIS and Environment Modeling, Santa Fe, NM*, 1996.
- [Zha11] Ziliang Zhao. Design and Implementation of an Object-Oriented Space-Time GIS Data Model. *Master's Thesis, University of Tennessee*, 2011. URL: http://trace.tennessee.edu/utk_gradthes/1043/ [cited 07 11 2015].
- [Zwa] Vladimir Zwass. information system. URL: <http://www.britannica.com/topic/information-system> [cited 05 11 2015].

Stuff